



UNIVERSITY OF THE PHILIPPINES
OPEN UNIVERSITY

Master of Information Systems

BYRON V. MANIQUIS

INTEGRATED INFORMATION SYSTEMS FOR
ENHANCED ACCOUNTS MANAGEMENT

Thesis Adviser:

Assistant Professor Mari Anjeli Lubrica Crisanto

Faculty of Information and Communication Studies

Date of Submission

18 November 2017

Permission is given for the Following people to have access to this thesis:

Available to the General Public	YES
Available only after consultation with author/thesis/dissertation adviser	YES
Available only to those bound by confidentiality agreement	YES

Student's Signature:

Signature of Student's Adviser:

© 2017 By Byron V. Maniquis

All Rights Reserved

University Permission Page

"I hereby grant the University of the Philippines a non-exclusive, worldwide, royalty-free license to reproduce, publish and publicly distribute copies of this thesis or dissertation in whatever form subject to the provisions of applicable laws, the provisions of the UP IRR policy and any contractual obligations, as well as more specific permission marking on the Title Page"

"Specifically, I grant the following rights to the University"

- a) To upload a copy of the work in the theses database of the college/school/institute/ department and in any other databases available on the public internet.*
- b) To publish the work in the college/school/institute /department journal, both in print and electronic or digital format and online, and*
- c) To give open access to above-mentioned work, thus allowing "fair use" of the work in accordance with the provisions of the Intellectual Property Code of the Philippines (Republic Act No. 8293), especially for teaching, scholarly and research purposes."*

Byron V. Mañiquis 18 November 2017
Student Name over Signature and Date

This Special Project titled
Integrated Information Systems for Enhanced Accounts Management
is hereby accepted by the Faculty of Information and Communication Studies in
partial fulfillment of the requirements for the Master of Information Systems

Assistant Professor Mari Anjeli Crisanto
Adviser

Dr. Concepcion L. Khan
Program Chair

Date

Professor Alexander G. Flor, Ph.D.
Dean
Faculty of Information and
Communication Studies

Date

This thesis titled **Integrated Information Systems for Enhanced Accounts Management** is hereby accepted by the Faculty of Information and Communication Studies, U.P. Open University, in partial fulfillment of the requirements for the degree Master of Information Systems

Members of the Academic Advisory Committee

_____, Chair, Advisory Committee
(Signature over printed name)

(Date)

_____, Member, Advisory Committee
(Signature over printed name)

(Date)

_____, Member, Advisory Committee
(Signature over printed name)

(Date)

_____, Dean,
(Signature over printed name) Faculty of Information and
Communication Studies

(Date)

ACKNOWLEDGMENT

The completion of this project is not possible without the blessings of our Lord God. For that I'm truly grateful to Him. Glory be to God!

I wish to express my sincere thanks to Assistant Professor Mari Anjeli Lubrica Crisanto not only for the sincere and valuable guidance but also for the encouragement she extended to me. Her patience is also much appreciated.

I'm also thankful to the University Computer Center, headed by our Director, Professor Manuel Ramos Jr., and to all my officemates (previous and current ;-)) for being there to support and encourage me.

Last but not least, I thank my family for the unceasing encouragement and support. To my wife, it was a great comfort and relief to know that you were willing to provide management of our household activities while I completed my project, especially those times when you kept our beautiful, loveable and inspiring daughter Eistelle from playing with me.

ABSTRACT

The University Computer Center, an office under the office of the Chancellor of the University of the Philippines Diliman, utilizes an increasing number of information systems to manage the Diliman Network accounts and related services. To improve efficiency and raise productivity, this project integrated several information systems into one enhanced accounts management website

The new website consolidated the functionalities of several information systems into one with three main (3) modules: (1) accounts management module, (2) mail module, and (3) proxy module. It also enhanced the existing features of these modules. The development was divided into three(3) major phases covering each module.

Through the use of Technology Acceptance Model theory, the perceived usefulness and perceived ease of use of the integrated system was evaluated to determine if the project achieved its goal. The reported findings show there is a significant relationship between the perceived ease of use and perceived usefulness which affects the attitude of the administrators towards using the new website. There is a strong correlation between perceived ease of use and perceived usefulness. Perceived usefulness has stronger correlation compared to perceived ease of use on the attitude toward using. Based on these findings, the goal of the project to integrate and enhance the accounts management is achieved.

Keywords – database administration, information integration, information retrieval, innovation, log analysis, perceived ease of use, perceived usefulness, responsive design, technology acceptance model, web sites.

TABLE OF CONTENTS

Chapter I INTRODUCTION 1

Statement of the Problem 1

Background and Objectives of the Project 3

Significance and Scope of the Project 4

Documentation of Existence and Seriousness of the Problem..... 5

SWOT and GAP Analyses 5

SWOT Analysis 6

GAP Analysis 7

Chapter II REVIEW OF EXISTING ALTERNATIVES..... 11

Review of Existing Alternatives 11

QMAIL Suite..... 11

LDAP Account Manager..... 12

Chapter III APPROACH TAKEN IN THIS SUBJECT..... 13

Theoretical Framework 13

Rationale for the Framework 14

Technologies..... 15

Chapter IV METHODOLOGY 17

Output Project 17

Tech Tools 18

Enhanced Account Lookup..... 18

TABLE OF CONTENTS

Enhanced Account Modification	19
Enhanced Disable Account	20
Special Account Creation	20
Multiple Account Creation	20
Merge Accounts	20
Mail Tools	21
Search Mail Logs	21
Block/Unblock Email	21
Block/Unblock Host	22
Proxy Tools	23
Block/Unblock Client	23
Radius Profile	24
Admin Tools	25
Activate Student Account	25
Methods	25
User Testing and Assessment	26
Chapter V RESULTS AND DISCUSSION.....	30
Chapter VI CONCLUSION	32
Chapter VII RECOMMENDATIONS	33
REFERENCES.....	34

TABLE OF CONTENTS

Appendix A ACCOUNTSDB DATA MODEL 35

Appendix B MAILDB DATA MODEL..... 38

Appendix C PROXYDB DATA MODEL..... 39

Appendix D SOFTWARE REQUIREMENTS SPECIFICATION 40

Appendix E USE CASE SUITE 44

Appendix F USE CASES 46

Appendix G FEATURE SET 63

Appendix H FEATURES 66

Appendix I USER MANUAL 89

Appendix J TECHNICAL REFERENCE 124

Appendix K NETWORK ARCHITECTURE..... 152

Appendix L SURVEY QUESTIONNAIRE RESULTS 153

Appendix M PROCESS MODELS 161

Appendix N PROGRAM LISTING..... 165

TABLE OF FIGURES

Figure 1 Block User's Proxy Access 8

Figure 2 New Edit Account 8

Figure 3 New Disable Account 9

Figure 4 Account Lookup in the old system 9

Figure 5 Additional DilNet Account Tools 9

Figure 6 New DilNet Account Lookup 10

Figure 7 Merge Accounts Preparation 10

Figure 8 Technology Acceptance Model 13

Figure 9 Network Architecture 17

Figure 10 New View Account 19

Figure 11 Search Mail Logs 21

Figure 12 Blocked Email Address Status 22

Figure 13 Blocked Mail Host Status 23

Figure 14 Blocked Proxy User Status 24

Figure 15 Radius Profile 24

Figure 16 Technology Acceptance Model with Hypotheses 26

Figure 17 Respondents Gender 27

Figure 18 Respondents Age 28

LIST OF TABLES

Table 1 SWOT Analysis of the IS group 6

Table 2 GAP Analysis of the project 7

Table 3 Reliability Scores Cronbach Alpha 30

Table 4 PEOU and PU Correlations 30

Table 5 PEOU and ATU Correlations 31

Table 6 PU and ATU Correlations 31

Chapter I

INTRODUCTION

Information technology (IT) services are used by organizations to advance and enhance the IT knowledge and skills of their stakeholders. These IT services often employ the AAA framework (authentication, authorization, accounting) for privileged access. Management of the users and their accounts are done by utilizing information systems.

Organizations with thousands of users need to have an efficient information system for managing not only the IT services they are providing but also the users or clients they are catering for. This will not only be beneficial to the users but also to the IS group assigned in managing the said users and services.

An underutilized information system will not bring the efficiency or productivity aimed by the organization no matter how good it is. Even businesses are plagued with the problem of underutilized systems (Venkatesh, 2000). An example is the Internal Revenue System (IRS) who reverted to manual processing of tax returns after investing \$4B on a system aimed to simplify it (Johnston, 1997).

The perceived ease of use and perceived usefulness of an information system play a great role on the utilization of an information system (Davis, 1989).

Statement of the Problem

The University Computer Center (UCC) manages the Diliman Network (DilNet) account required by several online services of the University of the Philippines (UP) Diliman. These online services, mostly built in-house by their respective units include:

1. Computerized Registration System (CRS) by the Office of the University Registrar.
2. iLib (online library) by the University Library.
3. University Virtual Learning Environment (UVLE) by the Diliman Interactive Learning Center (DILC)
4. Dorm Application (campus dormitory application) by the Office of Student Housing.
5. UP Diliman Webmail (official webmail for UP Diliman) by UCC.
6. Proxy services (for internet access) by the UCC.

UCC utilizes several information systems to manage DilNet accounts along with UPD Webmail and the Proxy service. These information systems include:

1. Tech.accounts website for managing employee accounts.
2. TechaccountsV2 website for student accounts activation and password changes.
3. CRS website with administrative access for checking student or faculty module status.
4. Radius server for the proxy service.
5. Mailcleaner website for antivirus and anti-spam.
6. Mightyban website for blocking or unblocking user accounts on the proxy servers.
7. PHPLDAPAdmin site for advanced user information modification.
8. Miscellaneous Command Line Interface (CLI) tools and scripts used for server side administration.

Improved efficiency can be achieved if the utilized functionalities of these information systems are integrated in one place and run using a graphical user interface (GUI).

Centralizing the management of these integrated tools will reduce the number of steps required to perform a task. It will also reduce the complexity of the whole system aside from conserving time and other resources.

Background and Objectives of the Project

UCC is composed of eleven (11) employees divided into the administrative unit and the network and systems operation unit. Its main thrust is to provide and maintain high-speed connectivity to the internet and inter-unit data and voice connectivity within the university thru the Diliman Network (DilNet).

The project is inspired by the enhancement done by UCC in the student account signup process. In the enhanced process, the student only presents UP ID and Form 5. After verification, the UP ID will be scanned and a printout will be given to the student containing the newly created account.

Before, the student needs to type all the details which take time. The admin then validates the input data, create the account, and lastly, print the account details.

For changing account passwords, this simple process of scanning and printing is also utilized.

The project aims to replicate the efficiency in the enhanced accounts management which covers both student and employees.

In the old process, when creating accounts, the client must present his/her information. The administrator will search for the information to know if the client has

an existing account. However, the search process takes a long time because there are several forms used depending on the provided information of the client

If the client has an account, the process will then be a modification process and will entail several more steps. It might even include accessing other servers just to effect the changes on the client's account

The main objective of the project is to integrate and consolidate the information systems into one enhanced accounts management website. The website will serve as a conduit for all accounts management processes without having to access several systems separately.

With the integration, a UCC staff will be able to:

- handle larger quantities of support requests
- achieve higher accuracy due to fewer mistakes
- achieve higher work efficiency

Significance and Scope of the Project

UCC has been re-assigned as a front-line service unit which requires the unit to provide support and services in compliance with the Anti-Red Tape Act (ARTA). The integration and enhancement of the information systems and tools provide another way for UCC to improve its performance in support and services.

The scope of the project is the integration and enhancement of the commonly used functionalities of the following information systems and tools:

- Tech.accounts website
- Techaccountsv2 website
- Freeradius server
- Mailcleaner tools

- Mightyban tools
- PHPLDAPAdmin functionalities
- CLI tools and scripts used for accounts, mail, and proxy management

The interfacing of the CLI programs for easier access and execution was implemented using direct secure shell access

The project concentrated on developing an in-house software and not purchasing standard package software.

Documentation of Existence and Seriousness of the Problem

Currently, the Information System (IS) group of UCC is composed of five (5) technical staff. UCC workforce is augmented by around twenty (20) student assistants for support services.

Two (2) are permanent staff and the rest are Non-UP contractuels. One of the contractuels is also the security consultant.

The technical staff is further divided into 2 subgroups:

- network team, and
- systems and software development team

The network team is in-charge of the network infrastructure support while the systems and software development team is in-charge of managing the servers and services along with developing or creating new information systems.

SWOT and GAP Analyses

During the proposal stage of the project, SWOT and GAP analyses were performed to determine how to best approach the project. With our limited personnel

and resources, it's best to make sure our time are well spent when getting involved with the project

SWOT Analysis

The IS group were requested to provide their inputs not only as individuals but as a whole group. They were asked to share what they think are their own and the group's SWOT. Afterwards, the inputs were combined and an overall SWOT analysis is produced. Table 1 below shows the IS Group's SWOT analysis.

Table 1 SWOT Analysis of the IS Group

	POSITIVE	NEGATIVE
INTERNAL	Strengths - Ability to respond to unit/user queries/requests fast - We are few and the stakeholders are many, so we don't want to let tasks pile up so we tackle and finish them as soon as possible - The technical staff shares the same values of not putting off for tomorrow what we can do today - Great team dynamics help us solve problems faster - CFOT trained employees - Emergency repairs of damaged fiber optic connections due to constructions/renovations of new/old buildings are easily done because two personnel are trained for such - Experienced security consultant - We have a security consultant who has more than 15 years of experience in network security and management which helps a lot in deterring IT related security threats. - Optimistic staff - Optimism shields us from impatient and/or demanding stakeholders. - We work with patience and compassion in supporting our stakeholders. - Even though sometimes we are at the verge of meltdown, we keep our cool. - Non-UP staff is more willing to extend their work hours. - Non-UP personnel are willing to work until late hours, during weekends or even work overnights to finish an urgent task or resolve an urgent problem.	Weaknesses - Limited resources - Our own budget is dependent on Department of Budget and Management's (DBM) allotment to UPD and in turn UPD to us. This restricts our technological capabilities - Flexible work hours for the Non-UP contractuels and student assistants - Not all personnel are available within the 8am to 5pm period delaying some support responses - Our security consultant is also a consultant in other government branch and at the same time, a faculty in UPD - Requests that involve security scanning requires a lead time of one (1) - Vulnerable to vital staff leaving or getting sick. - Recently, we have three (3) staff who went abroad. When an employee leaves, our performance is degraded until the position is filled. Same thing happens when a staff is sick, absent or on leave for whatever reason. - Sometimes the stakeholders think we're delaying the resolution of their problems but most of the time, we just don't have time process them yet. - Vulnerable to natural disasters or man-made accidents. - Although we have offline backups, we can still be rendered inoperable by disasters and would require time to get back up or fully operational again. - Not Windows Server or Windows Mobile savvy. - Most of our devices/gadgets are Linux, Mac/iOS, or Android. Servers are mostly, if not all, Linux based.

EXTERNAL	Opportunities • Proliferation of low cost mobile devices • Increasing Wi-Fi adoption means more deployment for us • Authentication-less or minimal authentication for proxy service • More convenience for stakeholders • We could develop portal based site for one stop authentication and authorization • New buildings being constructed around the campus • Can help us in making network infrastructure or network connection a standard in building construction/renovation like other utilities (water and electricity) • More deployment for network and VOIP installations • Renovation of our own building • We will be able to provide training facilities • We will be able to provide rentals which could give us additional source of funds • A fresh and better environment that will boost our productivity	Threats • Cloud computing • Most of our services are hosted locally While local hosting has its benefits being online or fully in the cloud also has benefits. Not being in the cloud could affect our organization in the future • Since we're not 'in the cloud', any natural disasters or man made accidents can render the whole DilNet inoperable and would require time to get back up • Budget support change • Budget support from UP System thru the eUP project may soon change or even stop depending on the priorities of the new administration as we have a new President leading the UP System • Thru collaboration with eUP/ITDC we acquired equipment and solutions not possible if done alone. Losing this will greatly affect our buying power

GAP Analysis

A gap analysis was conducted on the target information systems primarily the old tech.accounts website. The goal of the analysis is to flesh out the specifications of the desired features for the enhanced accounts management website. The shortcomings of the target systems were enumerated in the process and the corresponding replacement or enhanced features to fulfill the shortcomings were formulated.

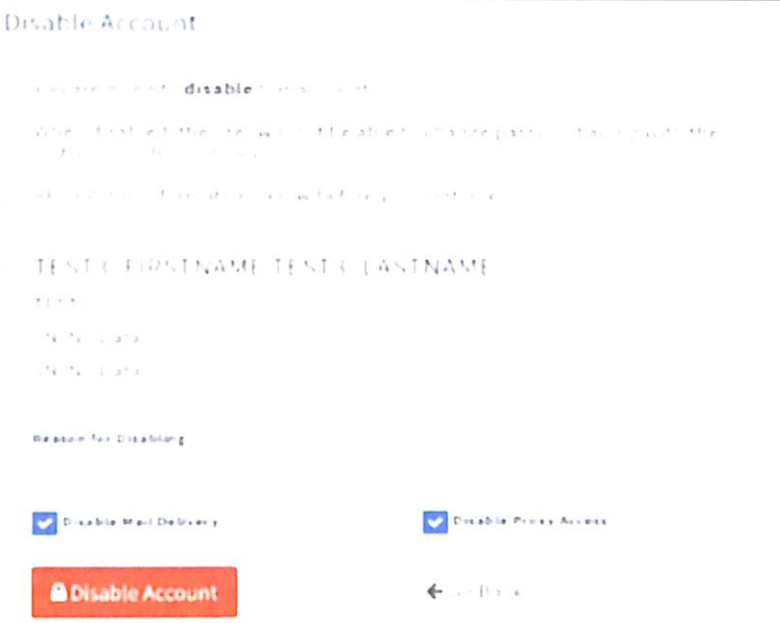
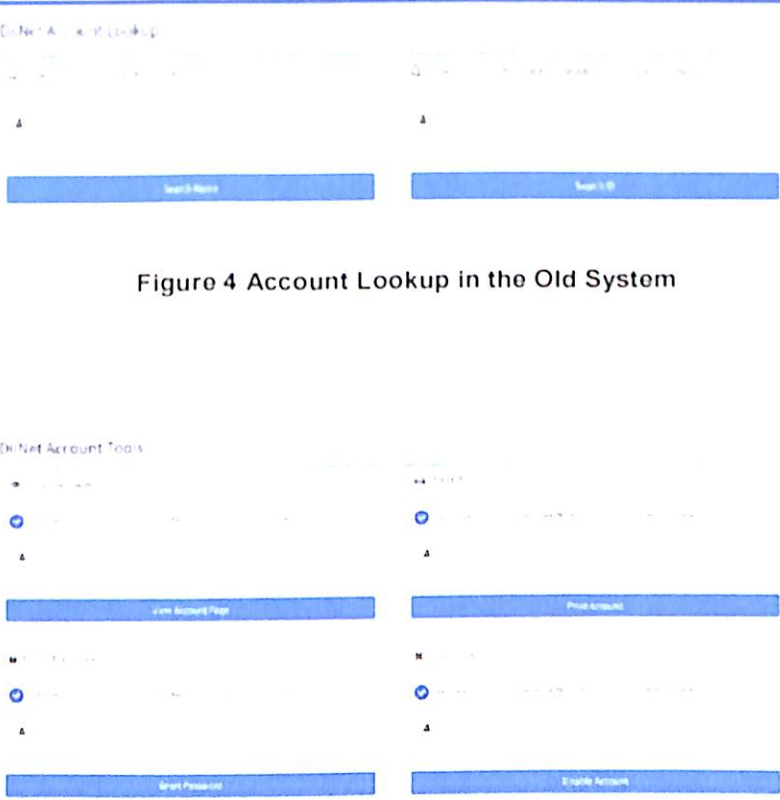
The IS Group was also asked for their biggest gripes and wish list for the target systems. These were incorporated in the analysis.

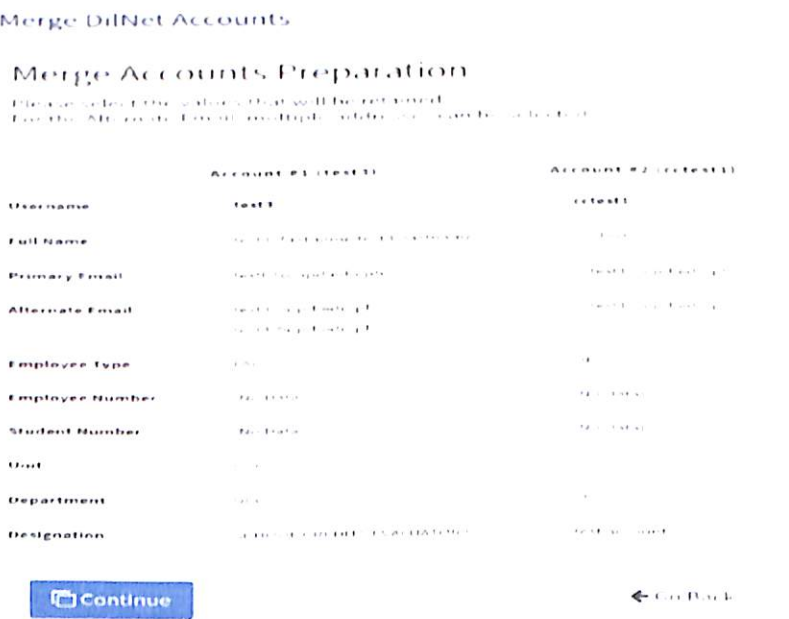
The GAP analysis of the project is shown below in Table 2.

Table 2 GAP Analysis of the project

Future Goal: One stop blocking/unblocking of DilNet accounts in the proxy service.		
CURRENT STRATEGIES	• Search for the username of the account on tech.accounts site. • Disable the account there by toggling the account status. • Login to all the proxy containers to add the username to the blacklist. • Reload all proxy hosts to effect the changes.	

ACTION PLAN	- The new Block/Unblock Client supports blocking of users on the proxy service without having to leave the website - Figure 1 on the right shows this feature	Block User Proxy Access To block proxying to block this DiINet account, when I block it, the user will not be able to use the proxy service until I click the information button before you continue. Block test1 Reason Block User Cancel Figure 1 Block User's Proxy Access
Future Goal: <i>One stop updating of accounts whose information changed for whatever reason (got married, legal name change, incorrect data, etc.).</i>		
CURRENT STRATEGIES	- In the old system search for the username of the target account - Then update the information - Next login to LDAP server to update uneditable details on the site - Lastly login to mail DB to synchronize the changes	
ACTION PLAN	- The new Edit Account page includes all these functionalities without leaving the website - The username, primary and alternate email addresses are now editable - Figure 2 on the right shows how it looks like	Edit DiINet Account TEST1 FIRSTNAME TEST1 LASTNAME Save Cancel Account Information Username: TEST1 FIRSTNAME TEST1 LASTNAME Primary Email Address: TEST1@TEST1.COM Alternate Email Address: TEST1@TEST1.COM Figure 2 New Edit Account
Future Goal: <i>Block or unblock an email using the same website. Useful when the target is compromised/malicious and sending phishing or scam emails.</i>		
CURRENT STRATEGIES	- To block a compromised DiINet account, a technical staff first searches for the email address used by the account for sending the spam/phishing emails. - The email address is then handed over to the mail admin who logs in to the mail relays to search and remove all the emails sent by the address. - While doing that, the mail admin adds the email to the blacklist to prevent it from sending another round of emails. - At the same time, the account is disabled and its password changed to prevent further damage.	

ACTION PLAN	- The new Disable Account page supports all the functionalities above in one place - The Admin can also disable the account's access to the proxy service in the same page along with the mail delivery and password change - Figure 3 on the right shows the new Disable Account feature	 Figure 3 New Disable Account
Future Goal: <i>Use only one search form instead of several to search for an account. It will be smart enough to determine what kind of input it has received and process it accordingly.</i>		
CURRENT STRATEGIES	- If the search criteria consist of a partial name or an email address, the admin must use the search by username form. See Figure 4 on the right - If the search criteria consist of numbers, the admin must use the search by student or employee number form - The email address must be a valid email address. Searching by full email address produces mixed results - The name must be a complete name. Partial names produce mixed results - Student number must be in the format (YYYY-NNNNN), otherwise it is considered as an employee number - Employee number must be in 10-digit format (0123456789). - If there is no DiI Net account yet and the student number must be checked against CRS, the format should be changed to YYYYNNNNN (without the dash). This results to confusion. - DiI Net Accounts tools (Figure 5 on the right) can be used for other actions.	 Figure 4 Account Lookup in the Old System Figure 5 Additional DiI Net Account Tools

ACTION PLAN	- We only search for a name, email, student number or employee number to find an account - The search functionality can distinguish between these types of information - Retaining just one search form and integrating all the functionalities in it achieved this goal - New account lookup is shown in Figure 6 on the right	 Figure 6 New DilNet Account Lookup
Future Goal: One stop page for merging duplicate accounts.		
CURRENT STRATEGIES	- If the client knows the usernames, the processing starts from there - Otherwise, the accounts are searched for a given criteria provided by the client - Once the accounts are found, usually, the older of the two accounts are retained while the other is removed - If the alternate email addresses of the duplicate account are being used, they are reassigned to the retained account - Lastly, the username of the duplicate account is deleted so it can be available again - Reassigning the alternate email addresses of the duplicate account requires another website for processing	
ACTION PLAN	- A Merge Accounts feature is introduced in the new system - The Admin can search for and merge accounts without leaving the website - The Admin is presented and asked to choose the attribute values that will be retained for the merged accounts - After review, the old account will be deleted and the chosen values reassigned to the retained account - Figure 7 on the right shows this new feature	 Figure 7 Merge Accounts Preparation

Chapter II

REVIEW OF EXISTING ALTERNATIVES

The project integrated the existing information systems into one centralized system, much like a command operations center. These information systems are either built in-house or collections of CLI programs run directly on the servers.

In searching for alternatives, only free suites or systems were considered. As such, commercial enterprise resource planning (ERP) solutions were disregarded.

The following are alternatives for each included subsystem. No alternatives were found for the new integrated system. These alternatives require separate utilization which has been solved through integration by the project.

QMAIL Suite

Qmail suite is a suite of tools for managing Qmail LDAP accounts either from command line interface (CLI) or thru web access.

Its features as listed on their site, www.qmailsuite.com:

- add, edit and delete email for his domain
- edit quotasize, quotacount and maxsize attributes
- change personal data and accountstatus for user
- add, edit and delete aliases and forward

The site also stated it can also do the following in CLI:

- add new domain in Qmail system and setup password for new domain postmaster
- delete entire domain, with optional safe mode to preserve maildir file (--safe)

- add new single email account or delete an existent account, with option safe mode
- change password for every account with qlpasswd binary

Though it has features for account management, QmailSuite is not sufficient for the advanced features UCC needs in account management. Neither can it be integrated with the other software considered in the following section.

LDAP Account Manager

From their site, www.ldap-account-manager.org

LDAP Account Manager (LAM) is a webfrontend for managing entries (e.g. users, groups, DHCP settings) stored in an LDAP directory. LAM was designed to make LDAP management as easy as possible for the user. It abstracts from the technical details of LDAP and allows persons without technical background to manage LDAP entries. If needed, power users may still directly edit LDAP entries via the integrated LDAP browser.

Advanced features require license subscription. The software supports several account types but the schema required by UCC is not supported.

Chapter III

APPROACH TAKEN IN THIS SUBJECT

Theoretical Framework

Technology Acceptance Model is the information system theory used in this project. The perceived usefulness and perceived ease of use of the new system were evaluated and found to be successful in achieving its goal of integrating the information systems for better accounts management.

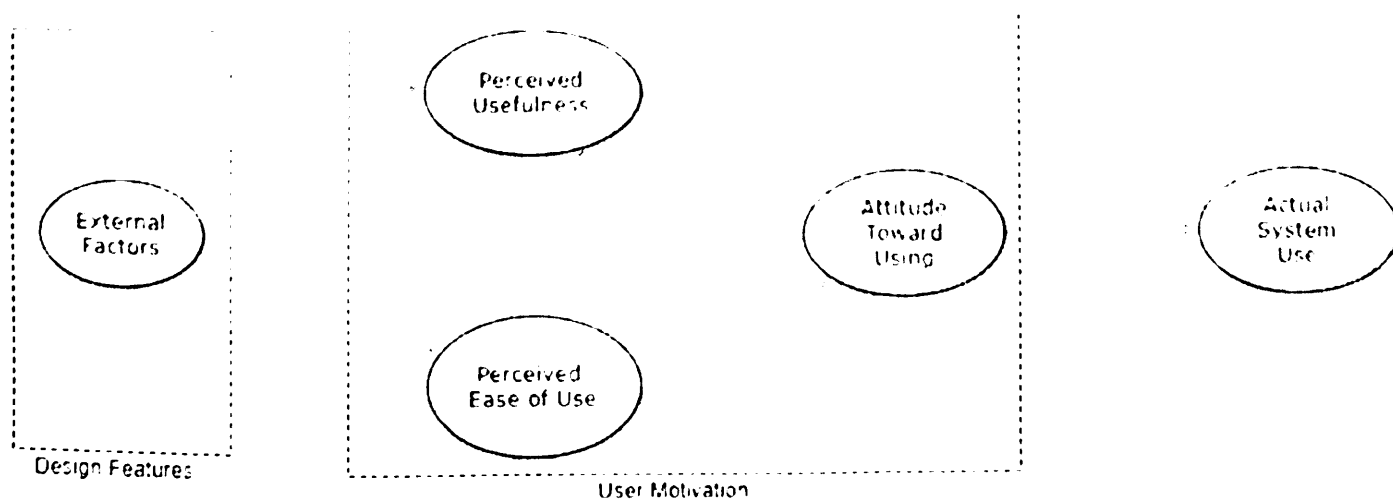


Figure 8 Technology Acceptance Model

The following system design principles also guided the project:

1. Prioritization

Due to the time constraint faced by the project, prioritization was exercised to maximize the allotted time schedule without compromising the system architecture, customer satisfaction and risk reduction (Botta and Bahill, 2008).

2. Work on high-risk items first

Working on high-risk items reduced the risks and the need to rework other items.

3. Do not optimize early

Optimization was done as late as possible in the project to maximize the time constraint and minimize re-optimization due to design changes

4. Develop stable intermediates

By developing intermediates that are stable, anytime the project is stopped, there will be a useful state or iteration of the project

5. Design for evolvability

Flexibility in system design allows for adapting to changes in available technologies.

6. Develop iteratively and test immediately

Testing of small completed parts are done before testing large complex systems.

7. Create modules

Making the system modular helped prevent changes cascade on the whole system. Pinpointing problems especially in complex systems requires less time if the system is modular.

8. Use open standards

They are publicly available and can be used to achieve specific tasks.

9. Group data and behavior

The framework used promoted this type of grouping. This is useful because data and the behavior that accesses that data often change together (Fowler, 2004).

Rationale for the Framework

Technology Acceptance Model (TAM) was used in the project to evaluate if the goal of the project was achieved. Specifically, measuring the perceived

usefulness and perceived ease of use was done to correlate their effects on the intention of the target users to utilize the new system

According to Davis (1986), "design features directly influence perceived usefulness and perceived ease of use. Perceived ease of use has a direct effect on perceived usefulness since a system which is easier to use will result in increased job performance. If the user becomes more productive in that fraction of time using the system which is easier to use, then he or she becomes productive overall

Using TAM, it's determined that the integration brought about the needed enhancements on accounts management

Technologies

The technologies used in the project are divided into the following categories:

- Front end
- Backend
- Command line script
- Version control

For the frontend the following frameworks or toolkits were used:

- Codeigniter
- Bootstrap for HTML, CSS and JS framework
- AdminLTE for the template and/or theme
- JQuery for JS Library

For the backend the following were used:

- MySQL is used for storing account related information.
- Open LDAP is used for the LDAP database. It's used for storing account credentials, mail and radius related information.

For CLI, PHP and BASH scripting were used. It's accessed via SSH by the webserver.

GIT was used for source code version control. This way, if there are changes that need to be rolled back, the author can utilize GIT to go back to a certain state or phase of the project. It is also used as a backup system for all the codes used in the project.

Chapter IV

METHODOLOGY

Output Project

The limitations and the shortcomings of the old system lead to the design and implementation of the new one. The name of the site remains the same: Tech.Accounts.

The new Tech.Accounts site has all the improved functionalities the old one needs. It also has the integration of the other information systems needed for effectively and efficiently managing DilNet accounts.

Figure 9 below shows the network architecture where the new website resides. The upper left side of the image shows the logical connections of the new site (indicated by blue lines) to the servers, namely: CRS, Mail Relays, LDAP and Radius servers. These connections give the administrator direct access without having to login on each one of them. Their services are all manageable within the new website. Appendix K shows a bigger version of the architecture.

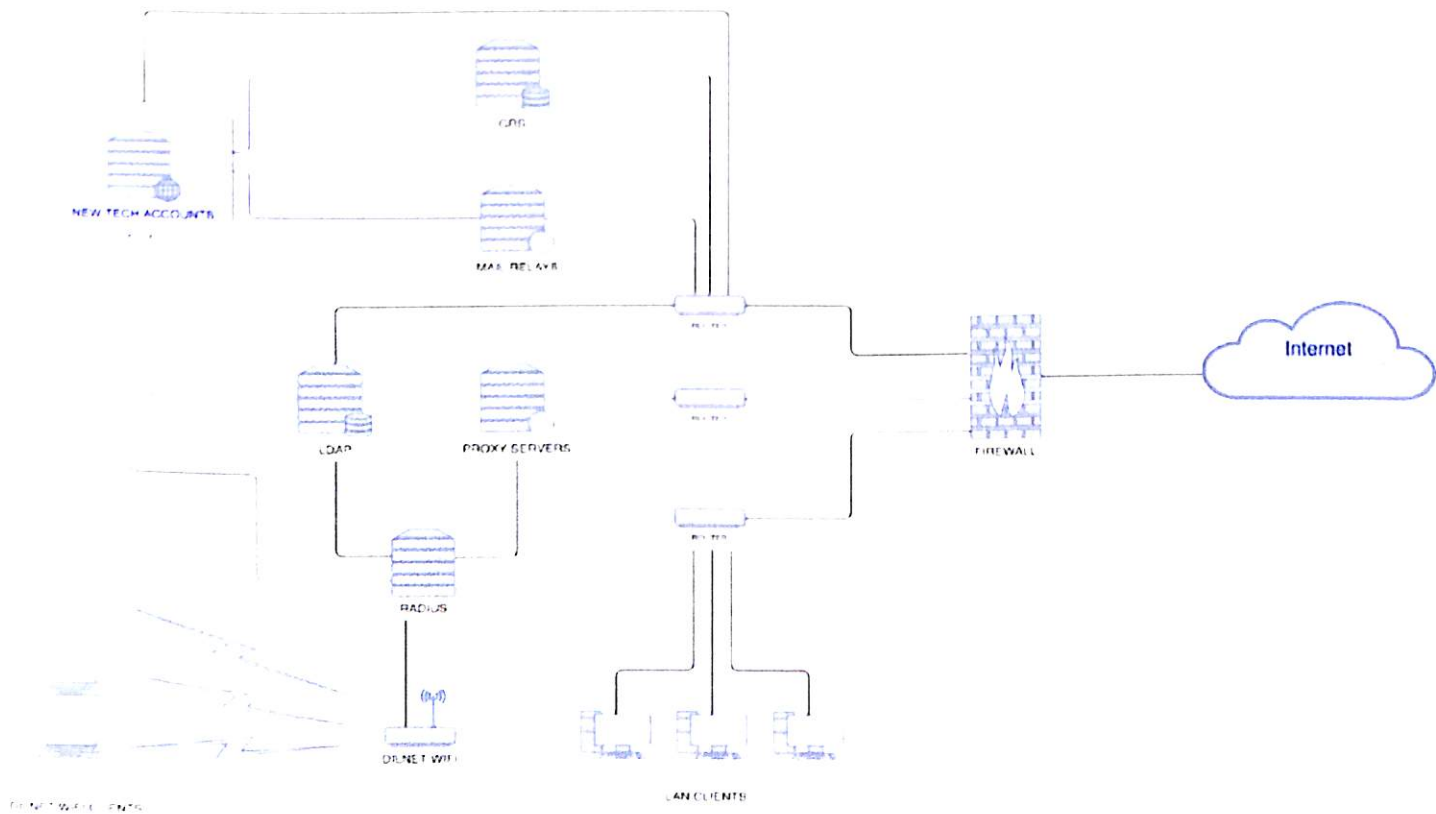


Figure 9 Network Architecture

Access to the new site is determined by the roles and permissions assigned to a DilNet account. These accounts are called Admins. Only Admins can login to the site.

There are two types of Admins: regular Tech staff or Supertech staff. Supertech staff has more responsibilities and is given additional tools to perform them. The features of the new website are detailed in Appendix H.

The following modules make up the new site:

- Tech Tools
- Mail Tools
- Proxy Tools
- Admin Tools

Tech Tools

This module is comprised of the tools used for managing DilNet account information. It includes the combined and enhanced features of tech.accounts, techaccountsv2, and accounts.upd. These features are:

Enhanced Account Lookup

In this version of Account lookup, the admin no longer needs to remember the formats of the student numbers or employee number. It can determine the format on its own and find the matching. It also improved the searching algorithm to minimize misses and increase the accuracy of the search.

If a single matching account is found, the admin is shown the View Account page of the match. This page also contains the actions possible for the account.

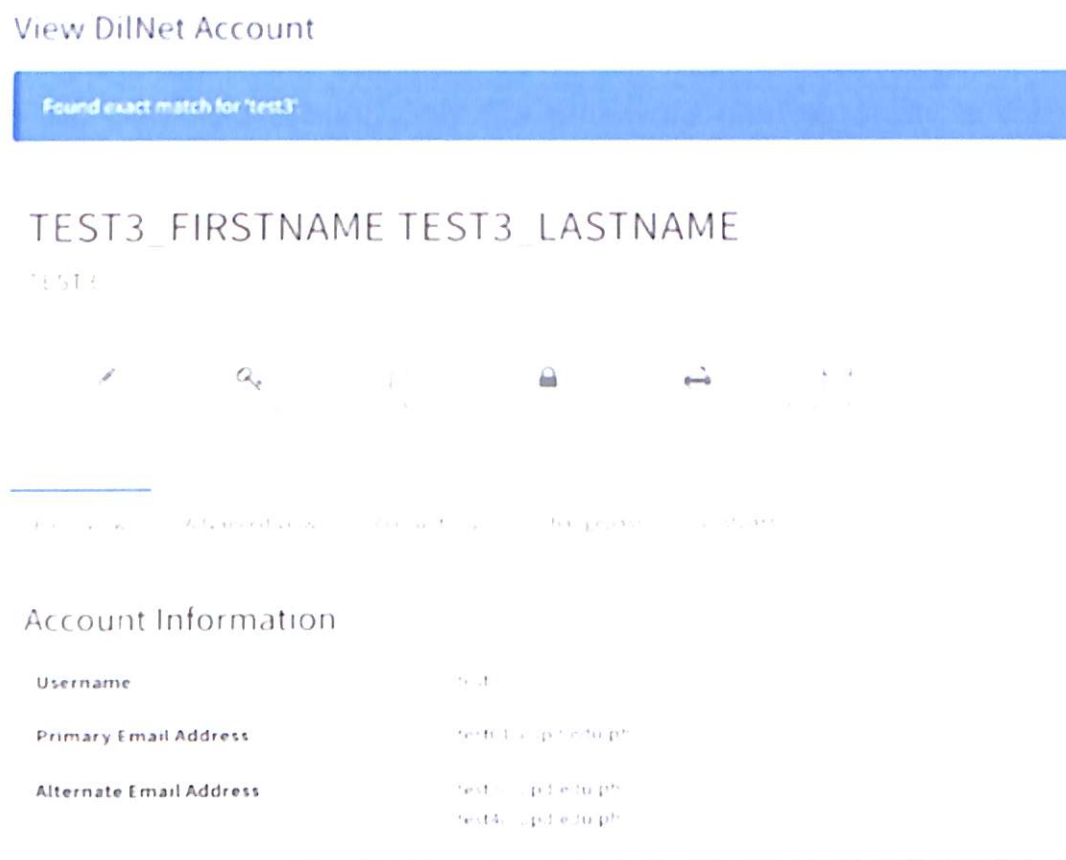


Figure 10 New View Account

Enhanced Account modification

In the new system, the Account modification has been enhanced to cater to the needs of those account holders whose names were legally changed and thus require updates on their names and email addresses. Usual cases are those who got married.

In the new system, the following fields can be edited (as shown in Figure 5) without going to another website:

- Username
- Primary Email
- Alternate Emails

Enhanced Disable Account

In the old Disable Account, only the password change ability is disabled. In the new version, the Admin can also disable the Mail Delivery and Proxy Service access of the account. Figure 3 shows the new Disable Account page.

Special Account Creation

Special Account creation is needed for cases where a specific email address must be granted to a Unit or Project. These accounts do not conform to the usual account generation. The username and email address are manually assigned by the Admin.

Multiple Account Creation

Multiple Account Creation is a new feature which allows the admin to create several employee accounts in one pass. A CSV file is required from the admin which contains the information of the accounts to be created. This feature can be accessed within the Create Employee Account page.

Merge Accounts

In the old site, there is no direct merge account feature. Merging accounts require accessing two additional servers to effect the change. Now, it can be done within the same site. The Admin only needs to find the two accounts to be merged and start the procedure.

The Admin chooses the username of the account plus all the attributes and values to be retained on the account. The non-chosen account is deleted afterwards. See Figure 7 for a screenshot.

Mail Tools

This module is comprised of tools used mainly for managing compromised accounts and spam cleaning. It can also be used to determine the status of a specific email. Its features are:

Search Mail Logs

This feature allows the Admin to search the current mail log file for a specific entry. Usual searches are the sender or recipient address, or the subject of the email. The Admin can't search for the content of the email because only transit information is logged. A screenshot is shown in Figure 11 below.

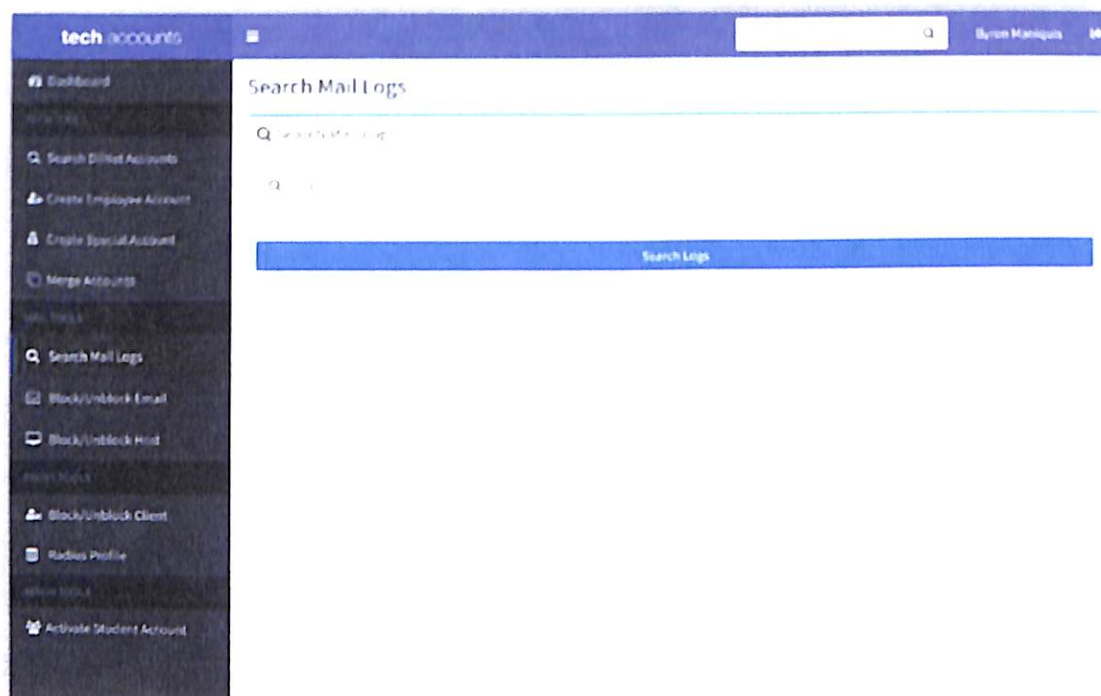


Figure 11 Search Mail Logs

Block/Unblock Email

When an account is compromised due to phishing scams or when an email address is spamming the Dilnet mail relays, this feature can be used to block the offending email address. The address only needs to be a valid email address regardless if it's a DilNet account or not.

Email addresses blocked by this feature can't send or received through the mail relays. Emails sent are rejected with a blacklisted error message.

A sample screenshot of a blocked email address is shown in Figure 12 below.



Figure 12 Blocked Email Address Status

Futhermore, the queued emails of the blocked address can also be removed by pressing the Remove Emails action button in the page. To unblock, the Unblock Email button is utilized.

Block/Unblock Host

There are times when the problem of spamming is not caused by a user but by a mail host server. This feature is designed for the scenario. The IP address of an

offending mail host server can be blocked or unblocked on this page. See Figure 13 below for a screenshot of a blocked mail host IP address.

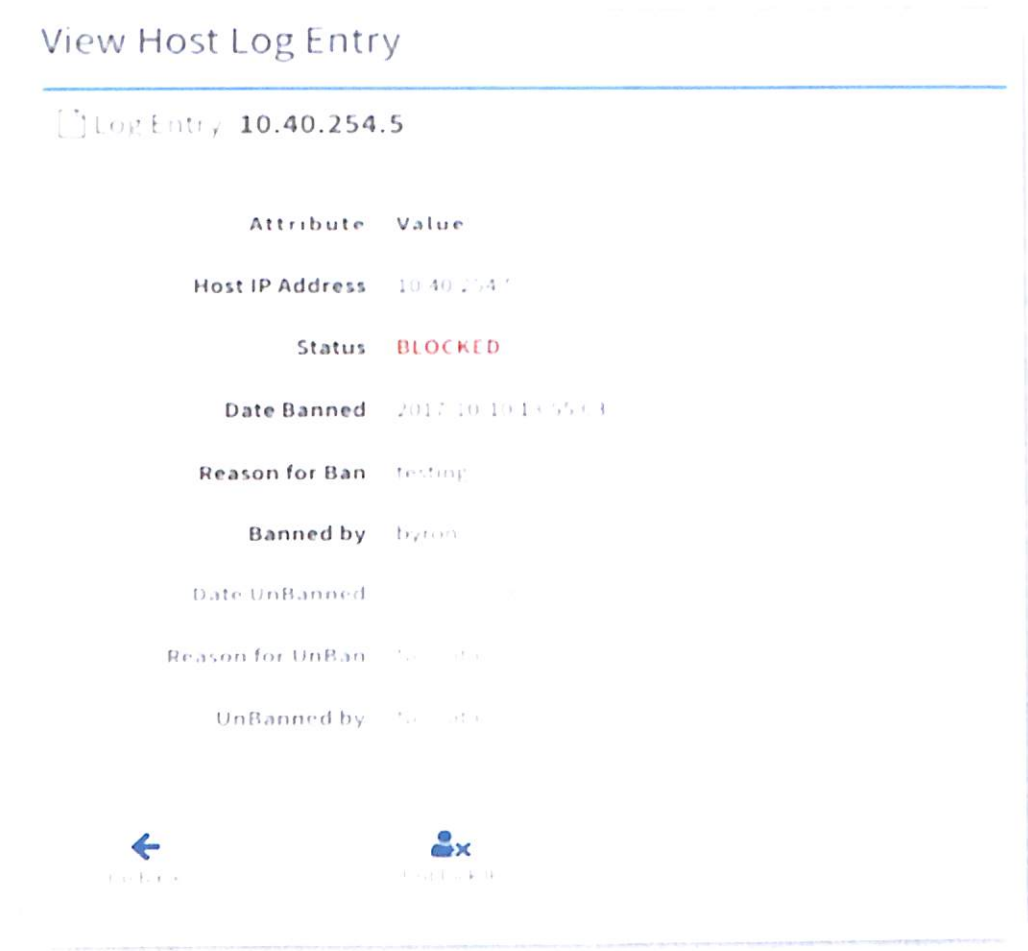


Figure 13 Blocked Mail Host Status

Blocked host IP addresses can't initiate mail transaction on the mail relays.

Proxy Tools

This module is used for managing the proxy service access of Dilnet accounts. The proxy service access is filtered by the freeRadius server. Only those authenticated and authorized by the server are allowed to use the proxy service.

Its features are:

Block/Unblock Client

A DilNet account's access to the proxy service can be disabled or re-enabled in this page. Blocking an account from the proxy service disables the account's

ability to login to the freeRadius server which serves as the gatekeeper for the proxy service.

This tool looks similar to the Mail Tools' email and host blocking features except this time, a Dilnet account is required. A blocked user will not be able to login to the radius server. A screenshot of a blocked user is shown in Figure 14 below.



Figure 14 Blocked Proxy User Status

Radius Profile

Radius profile summarizes the bandwidth usage in the proxy servers. A radius profile of a specific user can also be found in the same page.

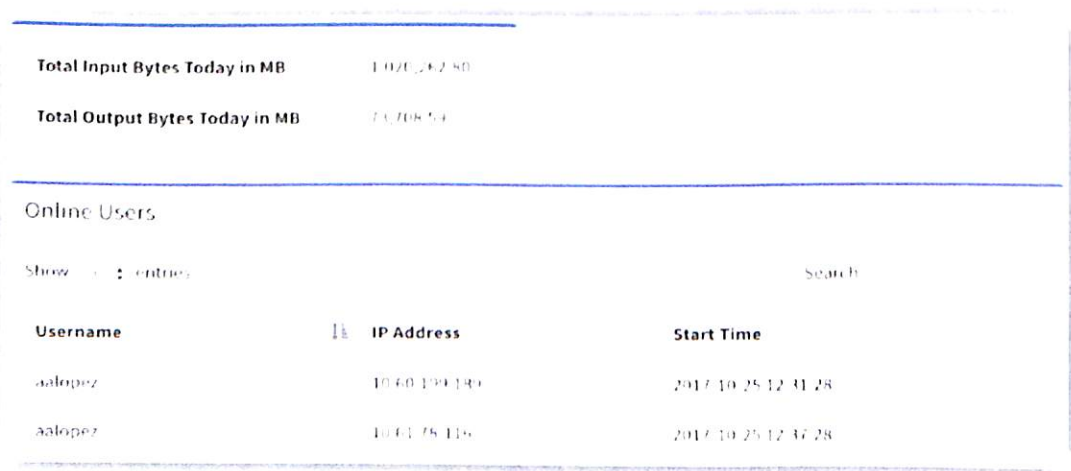


Figure 15 Radius Profile

Admin Tools

This module has the same features as techaccounts2. The techaccounts2 website is used by an administrative staff to activate and/or reset student accounts. The tool is adapted and integrated on the new Tech accounts in consideration of the enhanced features enumerated from the previous modules. Its sole feature is

Activate Student Account

An enrolled student who doesn't have a DilNet account can be granted one through this page. The Admin inputs the student number or the Form 5 barcode number of the student and the system will retrieve the user info from the CRS database. If there's no account associated, creation of account is initiated. Otherwise, password reset for the existing account is initiated.

Methods

To determine if the project goal was met, a targeted user testing was done. The users were sampled directly from the beneficiary of the project, the office of the University Computer Center.

The users were given activities to perform on the new site to let them experience the old website compared to the new one. The activities were designed to highlight the enhancements and new features of the new site.

After the activities, the users were asked to answer a questionnaire containing the TAM measurement scales. Each question is a 7-point Likert scale, with lowest score of 1 for Extremely UNLIKELY to the highest score of 7 for Extremely LIKELY.

Correlation analyses were done on the responses to determine the significance of the relationships between perceived ease of use and perceived usefulness of the new site. These relationships will answer if the goal was achieved.

User Testing and Assessment

According to Davis (1989), the perceived ease of use is the degree to which a person believes that using a particular system would be free of physical and mental effort. Effort in turn is a limited resource that one person may allocate to various activities for which the person is responsible (Radner and Rothschild, 1975).

On the other hand, perceived usefulness, still according to Davis, is the degree to which a person believes that using a particular system would enhance that person's job performance.

Lastly, attitude toward using is a function of the above beliefs: perceived ease of use and perceived usefulness (Davis, 1986).

To assess the effectiveness of the new website, the following hypotheses were formed:

H1: Perceived ease of use positively affects the perceived usefulness of the enhanced accounts management site.

H2: Perceived usefulness positively affects attitude toward using the enhanced accounts management site.

H3: Perceived ease of use positively affects attitude toward using the enhanced accounts management site.

All the hypotheses are drawn together with the TAM model below (Figure 16):

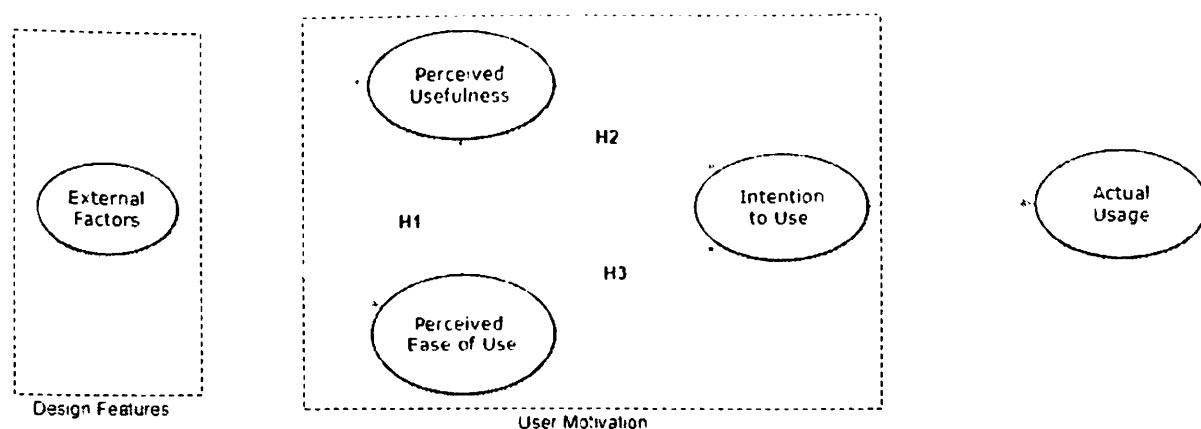


Figure 16 Technology Acceptance Model with Hypotheses

The target sample is composed of employees and student assistants. Of the possible 25, only 20 were able to complete the activities and answer the survey.

Of these 20 respondents, 11 were males and 9 were females. A pie chart (Figure 16 below) shows gender distribution. There is an almost equal distribution of males and females.

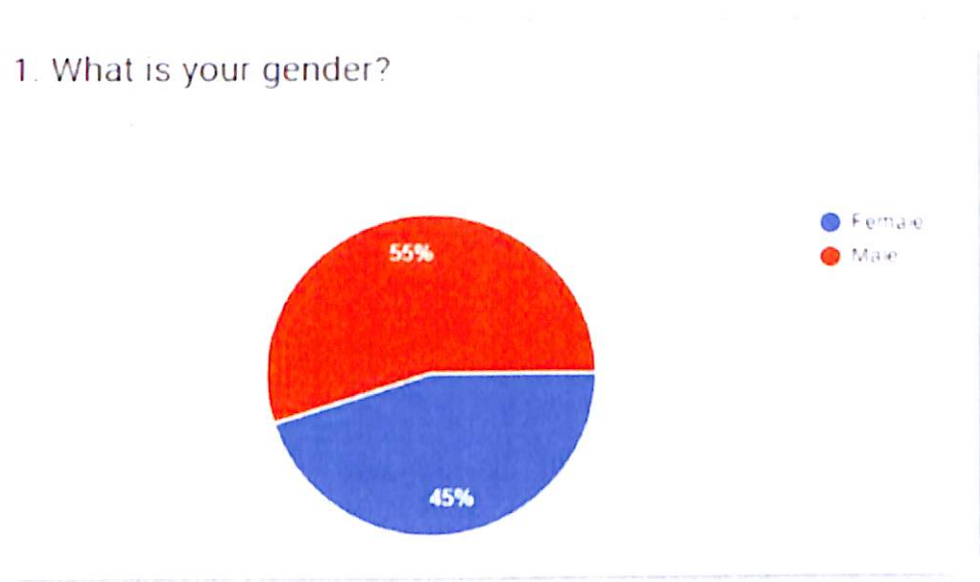


Figure 17 Respondents Gender

The age group is divided into following: 13 respondents is between 18 to 24 years old; 4 respondents are between 25 to 34 years old; and 3 are between 35 to 44 years old. See Figure 18 below.

Majority of the respondents are between 18 to 24 years old. The ratio of student assistants to employees is close to 2:1.

2 What is your age group?

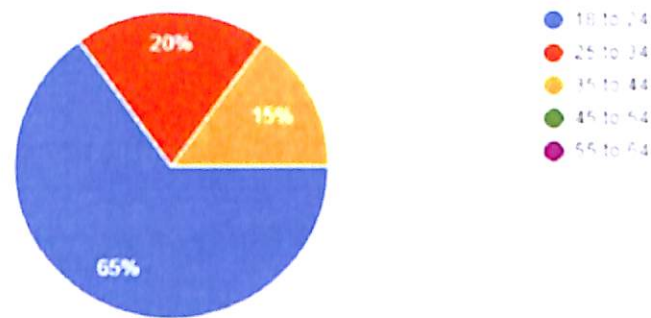


Figure 18 Respondents Age

The following are the survey questions employed. Appendix L shows the results gathered from the questionnaires.

Perceived Usefulness (PU):

1. Using the new Tech.Accounts in my job would enable me to accomplish tasks more quickly.
2. Using the new Tech.Accounts would improve my job performance.
3. Using the new Tech.Accounts in my job would increase my productivity.
4. Using the new Tech.Accounts would enhance my effectiveness on the job.
5. Using the new Tech.Accounts would make it easier to do my job.
6. I would find the new Tech.Accounts useful in my job.

Perceived Ease of Use (PEOU):

1. Learning to operate the new Tech.Accounts would be easy for me.
2. I would find it easy to get the new Tech.Accounts to do what I want it to do.
3. My interaction with the new Tech.Accounts would be clear and understandable.

- 4 I would find the new Tech Accounts to be flexible to interact with
- 5 It would be easy for me to become skillful at using the new Tech Accounts
- 6 I would find the new Tech Accounts easy to use

Attitude Toward Using (ATU):

1. I believe it is a good idea to use the new Tech Accounts.
2. I like the idea of using the new Tech Accounts
3. Using the new Tech Accounts is a positive idea.

Chapter V

RESULTS AND DISCUSSIONS

Using the TAM measurement scales (Davis, 1989) the perceived usefulness and perceived ease of use were measured using a 7-point Likert scale

Using the Cronbach Alpha coefficient test, the reliability and validity of each category (PU, PEOU, ATU) were tested Table 3 below shows the reliability scores.

Table 3. Reliability Scores Cronbach Alpha

Category	# of Items	Cronbach Alpha
Perceived Usefulness (PU)	6	0.875
Perceived Ease of Use (PEOU)	6	0.896
Attitude Toward Using (ATU)	3	0.813
Overall	15	0.951

All of the categories scored greater than the minimum 0.70 recommended by Nunally & Bernstein (1994). This suggests that the measurement scales used were valid.

From the correlation analysis result in Table 4, it can be observed there is a significant positive relationship between perceived ease of use and perceived usefulness of the enhanced accounts management site. Therefore, H1 is supported.

Table 4. PEOU and PU correlations

Correlations		
Factors		PU
PEOU	r-value	0.929***
	p-value (p<0.001)	3.24863E-09
	N	20

From the correlation analysis result in Table 5, it can be observed there is a significant positive relationship between perceived ease of use and attitude toward using the enhanced accounts management site. Therefore, H2 is supported.

Table 5. PEOU and ATU correlations

Correlations		
Factors		ATU
PEOU	r-value	0.750***
	p-value (p<0.001)	0.000140231
	N	20

From the correlation analysis result in Table 6, it can be observed there is a significant positive relationship between perceived usefulness and attitude toward using the enhanced accounts management site. Therefore, H3 is supported.

Table 6. PU and ATU correlations

Correlations		
Factors		ATU
PU	r-value	0.825***
	p-value (p<0.001)	7.58868E-06
	N	20

It can also be observed from Table 5 and Table 6 that PU-ATU has much lower significance value (p-value) compared to PEOU-ATU. Even PEOU-PU's p-value (Table 4) is much lower than PEOU-ATU.

Chapter VI

CONCLUSION

The results validate the relationships between perceived ease of use, perceived usefulness and attitude toward use. This in turn, validates the goal of enhancing the accounts management through the integrated systems.

From the results, it is seen that perceived usefulness has greater effect on the attitude toward using compared to the perceived ease of use. This means that user friendliness of a website doesn't necessarily mean the client is going to use the site already. The client must feel that the website is indeed useful for his job.

It is also seen that perceived ease of use has a significant effect on the perceived usefulness of the site. This means that making the site user friendly helps in emphasizing the usefulness of the site to the target user.

From these observations, the goal of enhancing the accounts management through the integrated information systems has been achieved.

Chapter VII

RECOMMENDATIONS

The implementation of the project has some limitations. First and most importantly, it was limited by time. Due to limited time, the statistical tests done are limited. Other statistical analyses might give additional results that might further confirm or deny the results. Furthermore, since the users sampled are small, increasing it could make the conclusion more general.

The evaluation through the use of the technology acceptance model was only utilized in the new website and not the old one.

The focus of the project was the implementation of an everyday tool used by the proponent office. The tool might or might not be that useful to other units.

Improvements on the output product are recommended. For example, in blocking/unblocking users/emails/hosts, the currently blocked users/emails/hosts could be listed in their own pages just like in the radius profile page. Also, the Go Back functionalities could further be improved to make it targeted depending on the referrer of the page.

Personalization might or might not improve the usability of the website but is still recommended to make each admin truly at home with the website.

Further research could also be made on the actual system usage of the new website to further determine the relationships between perceived ease of use, perceived usefulness and attitude toward using the new site.

REFERENCES

- Bahill, A. T. & Botta, R. (2008). Fundamental Principles of Good System Design. *Engineering Management Journal*, 20(4), 9–17.
- Davis, F. D. (1986). A technology acceptance model for empirically testing new end-user information systems: Theory and results. Doctoral dissertation. Cambridge, MA: MIT Sloan School of Management.
- Davis, F. D. (1989). Perceived usefulness, perceived ease of use, and user acceptance of information technology. *MIS Quarterly*, 13(3), 319–339.
- Fowler, Martin (2004). *UML Distilled: A Brief Guide to the Standard Object Modeling Language*, 3rd Edition, Addison-Wesley.
- Johnston, D.C. (1997). Computers clogged, IRS seeks to hire outside processors. *New York Times* (Jan. 31).
- LDAP Account Manager. (n.d.). Retrieved from <http://www.ldap-account-manager.org/>
- Nunally, J. C., Bernstein, I. H. (1994). *Psychometric Theory*. New York: McGraw-Hill.
- PHPLDAPAdmin Main Page. (2013, April 8). Retrieved from http://phpldapadmin.sourceforge.net/wiki/index.php/Main_Page
- QmailSuite Qmail Ldap Tools (n.d.). Retrieved from <http://www.qmailsuite.com/>
- Radner, R. and Rothschild, M. (1975). "On the Allocation of Effort," *Journal of Economic Theory* (10), 358-376.
- Venkatesh, Viswanath (2000). Determinants of Perceived Ease of Use: Integrating Control, Intrinsic Motivation, and Emotion into the Technology Acceptance Model. *Information Systems Research* 11(4), 342-365.
- Welcome to the MailCleaner OpenSource Edition web site. (2014, December 12). Retrieved from <http://www.mailcleaner.org/>

Appendix A

ACCOUNTSDB DATA MODEL

Admin_users Table

Field	Type	Null	Key	Default	Extra
userid	int(11)	NO	PRI	NULL	auto_increment
username	varchar(32)	NO		NULL	
empno	varchar(10)	YES		NULL	
role	tinyint(4)	NO		0	

4 rows in set (0.01 sec)

Aliases Table

Field	Type	Null	Key	Default	Extra
id	int(11)	NO	PRI	NULL	auto_increment
unit_id	int(11)	NO	MUL	NULL	
acronym	varchar(64)	NO		NULL	
name	varchar(64)	NO		NULL	

4 rows in set (0.01 sec)

Changepass Table

Field	Type	Null	Key	Default	Extra
username	varchar(16)	NO		NULL	
ip	varchar(40)	NO		NULL	
changed_by	varchar(40)	NO		NULL	
time	datetime	NO		NULL	
notes	text	YES		NULL	

5 rows in set (0.02 sec)

Appendix A (continued)

ACCOUNTSDB DATA MODEL

Ci_sessions_techv3 Table

Field	Type	Null	Key	Default	Extra
id	varchar(40)	NO	PRI	NULL	
ip_address	varchar(45)	NO	PRI	NULL	
timestamp	int(10) unsigned	NO	MUL	0	
data	mediumblob	NO		NULL	

4 rows in set (0.01 sec)

Logs Table

Field	Type	Null	Key	Default	Extra
id	int(11)	NO	PRI	NULL	auto_increment
time	datetime	YES		NULL	
account	varchar(32)	YES		NULL	
detail	text	YES		NULL	
service	varchar(32)	YES		NULL	
log_level	enum('debug','info','warn','error','severe')	YES		NULL	
changed_by	varchar(40)	YES		NULL	
ip	varchar(40)	YES		NULL	

8 rows in set (0.01 sec)

Tech_role_perms Table

Field	Type	Null	Key	Default	Extra
id	int(11)	NO	PRI	NULL	auto_increment
role	varchar(255)	NO		NULL	
permname	varchar(255)	NO		NULL	

3 rows in set (0.01 sec)

Tech_users Table

Field	Type	Null	Key	Default	Extra
userid	int(11)	NO	PRI	NULL	auto_increment
username	varchar(32)	NO		NULL	
empno	varchar(10)	NO		NULL	
roles	varchar(255)	NO		NULL	
last_login	datetime	YES		NULL	

5 rows in set (0.01 sec)

Appendix A (continued)

ACCOUNTSDB DATA MODEL

Units Table

Field	Type	Null	Key	Default	Extra
unit_id	int(11)	NO	PRI	NULL	auto_increment
parent_id	int(11)	NO	MUL	0	
unit_type	int(11)	NO	MUL	NULL	

3 rows in set (0.01 sec)

Users Table

Field	Type	Null	Key	Default	Extra
id	int(11)	NO	PRI	NULL	auto_increment
username	varchar(32)	YES	UNI	NULL	
email	text	YES		NULL	
lastname	varchar(32)	YES		NULL	
middlename	varchar(32)	YES		NULL	
firstname	varchar(32)	YES	MUL	NULL	
id_number	varchar(16)	YES		NULL	
campus_id	int(11)	NO	MUL	NULL	
unit_id	int(11)	NO	MUL	NULL	
designation	text	NO		NULL	
course_id	int(11)	NO	MUL	NULL	
hostname	varchar(32)	NO	MUL	NULL	
activated	int(11)	NO		0	
birthday	date	NO		NULL	
gender	int(11)	NO		NULL	
date_activated	timestamp	NO		CURRENT_TIMESTAMP	
activated_by	varchar(16)	NO		NULL	
ip_activated	varchar(40)	YES		NULL	
security_question	int(11)	YES		NULL	
security_answer	varchar(40)	YES		NULL	
recovery_email	varchar(255)	YES		NULL	
empno	varchar(16)	YES		NULL	
studno	varchar(16)	YES		NULL	
type	varchar(16)	YES		NULL	
recovery_count	int(11)	YES		0	
callcode	varchar(40)	YES		NULL	
security_lastupdated	datetime	YES		NULL	
force_changepass	int(11)	YES		1	
last_login	datetime	YES		NULL	
disabled	tinyint(1)	YES		0	
disabled_remarks	varchar(40)	YES		NULL	

31 rows in set (0.01 sec)

Appendix B

MAILDB DATA MODEL

Emails Table

Field	Type	Null	Key	Default	Extra
id	int(10) unsigned	NO	PRI	NULL	auto_increment
email_address	varchar(256)	NO		NULL	
disabled	tinyint(1)	NO		0	
date_banned	datetime	YES		NULL	
date_unbanned	datetime	NO		NULL	
reason_banned	text	NO		NULL	
reason_unbanned	text	NO		NULL	
tech_banner	varchar(128)	NO		NULL	
tech_unbanner	varchar(128)	NO		NULL	

9 rows in set (0.01 sec)

Hosts Table

Field	Type	Null	Key	Default	Extra
id	int(10) unsigned	NO	PRI	NULL	auto_increment
ip_address	varchar(16)	NO		NULL	
disabled	tinyint(1)	NO		0	
date_banned	datetime	YES		NULL	
date_unbanned	datetime	NO		NULL	
reason_banned	text	NO		NULL	
reason_unbanned	text	NO		NULL	
tech_banner	varchar(128)	NO		NULL	
tech_unbanner	varchar(128)	NO		NULL	

9 rows in set (0.01 sec)

Appendix C

PROXYDB DATA MODEL

Users Table

Field	Type	Null	Key	Default	Extra
id	int(10) unsigned	NO	PRI	NULL	auto_increment
username	varchar(256)	NO		NULL	
disabled	tinyint(1)	NO		0	
date_banned	datetime	YES		NULL	
date_unbanned	datetime	NO		NULL	
reason_banned	text	NO		NULL	
reason_unbanned	text	NO		NULL	
tech_banner	varchar(128)	NO		NULL	
tech_unbanner	varchar(128)	NO		NULL	

9 rows in set (0.01 sec)

Appendix D

SOFTWARE REQUIREMENTS SPECIFICATION

Release Information

Project:	Enhanced Accounts Management Website
Internal Release Number:	1 0 0
Attached worksheets:	Use Case Suite, Use Cases, Feature Set, Features
Related Documents:	Project Proposal

Introduction

The Enhanced Accounts Management website will be the new Tech.Accounts website used by the University Computer Center. It is an integration of several information systems used for managing the DilNet accounts.

The University Computer Center is an office under the Office of the Chancellor of UP Diliman in charge with managing the Diliman Network.

Use Cases

There are three (3) main use cases for this project:

- Accounts management comprising of:
 - Tech Tools
 - Admin Tools
- Mail management via Mail Tools
- Proxy management via Proxy Tools

Please see Appendix E for the **Use Case Suite** for more information.

Functional Requirements

Depending on the level of the technical staff or the admin level, an admin can perform several tasks on the Enhanced Accounts Management Website.

A **Supertech level staff** can utilize all the functionalities of the site while a **regular Tech staff** can use a finite set of functionalities. Non-tech staff are not allowed to use the site at all.

Please see Appendix G for the **Feature Set** of the project for more information.

Non-Functional Requirements

What are the usability requirements?

The main criterion for the usability of this project is the difficulty in attending to cases compared to the old system. Difficulty depends on how complex it is to manage a DilNet account given a desired operation. Essentially, reducing the number of steps it takes and reducing the number of websites to log on to perform tasks will help determine the usability of the new site. For example, does the number of reduction in steps or reduction in time used justify this project?

The user interface must conform to the modern standards in responsiveness utilizing the latest and most user-friendly frameworks in HTML, CSS, and JS.

Users must be able to perform tasks regardless of what device they are using. Hence, mobile first design is implemented to make sure the website is usable in most if not all screen sizes.

What are the security requirements?

Access will be controlled with DilNet account credentials. Only UCC Technical staff is allowed to use the site. The technical staff consists of the UCC director and employees belonging to the systems and network operations team.

What are the maintainability and upgradability requirements?

The website is using different frameworks designed for maintainability and upgradability. changes over time can easily be integrated

What are the supportability and operability requirements?

Supportability is our ability to provide cost effective technical support. Supportability is an inherent cost included in the daily or hourly rate of our staff.

Operability is our ability to host and operate the software. The different frameworks used should help us achieve our goal of 99.9% uptime. Key features supporting the ability to do hot data backups, and application restoration in case of hardware failures through a GIT repository.

What are the business life-cycle requirements?

Until there is a new technology that will surpass the features and advantages of the frameworks used in this project, we will use this enhanced website.

New features will be added at no cost to the customers or until our office maintains its mandate to provide support to our clients.

More user customizations will be developed to make the site truly personal for each administrator.

Environmental Requirements

What are the system software requirements?

PHP 5.4 and above are required for the CodeIgniter framework. MySQL 5.1 and above will be used for the database needs. Bootstrap 3.3.6 and above is required for the CSS framework and the latest version of JQuery for the JS library.

What are the data import and export requirements?

MySQL will be used for the data import and export. It must support Fulltext query to enable some search features. MyISAM will be used as the default database storage type.

OpenLDAP is used for the LDAP directory where the usernames and passwords are stored. LDAP over SSL is required for protection of the login operations.

Secure Shell (SSH) is required for some backend operations where web service is not applicable due to system resources constraint.

Exim is used by the mail relays for mail delivery operations.

Appendix E
USE CASE SUITE

Release Information

Project:	DilNet Helpdesk
Internal Release Number:	1.0.0
Related Documents:	Features, Feature Set, Use Cases

Use Cases by Functional Area

- Tech Tools
 - UC-00 DilNet Account Lookup
 - UC-01 View Account - Basic View
 - UC-02 View Account - Advanced View
 - UC-03 View Account - Account Log
 - UC-04 View Account - Change pass
 - UC-05 View Account - Lost pass
 - UC-06 Edit Account
 - UC-07 Reset Password
 - UC-08 Disable/Enable Account
 - UC-09 Print Account
 - UC-10 Resend Email
 - UC-11 Create Employee Account
 - UC-12 Create Special Account
 - UC-13 Create Multiple Accounts
 - UC-14 Merge Accounts
 - UC-15 Add/Remove to/from Merge
 - UC-16 Edit Account - Username
 - UC-17 Edit Account - Primary Email
 - UC-18 Edit Account - Alternate Mail
 - UC-19 Edit Account - Mail Forwarding
- Mail Tools
 - UC-20 Search Mail Logs
 - UC-21 View Mail Log Entry
 - UC-22 Block/Unblock Email
 - UC-23 View Email Status
 - UC-24 Block/Unblock Sender
 - UC-25 Block/Unblock Recipient
 - UC-26 Remove Emails
 - UC-27 Block/Unblock Host
 - UC-28 View Host Log Entry
- Proxy Tools
 - UC-30 Block/Unblock Client
 - UC-31 View User Proxy Status

- UC-32 Radius Profile
 - UC-33 View User Radius Profile
- Admin Tools
 - UC-40 Activate Student Account
- Dashboard
 - UC-51 Accounts Statistics

Use Cases by Stakeholder

- All Tech Staff
 - UC-00 DilNet Account Lookup
 - UC-01 View Account - Basic View
 - UC-02 View Account - Advanced View
 - UC-03 View Account - Account Log
 - UC-04 View Account - Changepass
 - UC-05 View Account - Lostpass
 - UC-06 Edit Account
 - UC-07 Reset Password
 - UC-08 Disable/Enable Account
 - UC-09 Print Account
 - UC-10 Resend Email
 - UC-11 Create Employee Account
 - UC-20 Search Mail Logs
 - UC-21 View Mail Log Entry
 - UC-22 Block/Unblock Email
 - UC-23 View Email Status
 - UC-24 Block/Unblock Sender
 - UC-25 Block/Unblock Recipient
 - UC-27 Block/Unblock Host
 - UC-28 View Host Log Entry
 - UC-30 Block/Unblock Client
 - UC-31 View User Proxy Status
 - UC-32 Radius Profile
 - UC-33 View User Radius Profile
 - UC-40 Activate Student Account
 - UC-51 Accounts Statistics
- Super Tech Staff
 - UC-13 Create Multiple Accounts
 - UC-14 Merge Accounts
 - UC-15 Add/Remove to/from Merge
 - UC-16 Edit Username
 - UC-17 Edit Primary Email
 - UC-18 Edit Alternate Mail
 - UC-19 Edit Mail Forwarding
 - UC-26 Remove Emails

Appendix F

USE CASES

Release Information

Project:	DilNet Helpdesk
Internal Release Number:	1.0.0
Related Documents:	Features, Feature Set, Use Case Suite

Default Aspects of All Use Cases

Direct Actors:	User: end-user in any role (tech, supertech) System: The system being built When actors are not listed, assume User is doing it. Items beginning with "see" indicate that System has presented a new screen.
Stakeholders:	Tech Staff
Prereq:	User is logged in

UC-00: DilNet Account Lookup

Summary:	The Administrator enters a keyword on the search page to find an account.
Priority:	Essential
Use Frequency:	Often
Direct Actors:	Admin: Tech Staff or Supertech Staff
Main Success Scenario:	1. Visit Search DilNet Accounts page as an admin 2. Enter a search keyword which can be: ○ a part of name, ○ a part of email, ○ student number, ○ or employee number 3. Submit form. 4. See search result.
Alternative Scenario	• Exact match found, redirect to View Account page.

Extensions:	Multiple results The Admin selects the appropriate account for next operation
--------------------	---

UC-01: View Account - Basic View

Summary:	The Administrator navigates to the Basic View tab in the View Account page to view basic information of the user.
Priority:	Essential
Use Frequency:	Often
Direct Actors:	Admin: Tech Staff or Supertech Staff
Main Success Scenario:	- Upon a successful search, the Admin clicks on the View Account link (view icon) of the user. - Select the Basic View tab (by default, it's selected) - See Basic Information of the user.
Alternative Scenario Extensions:	- From the Edit Account Page, the Admin clicks on the View Account. - On the Merge liist, the Admin clicks on the View Account.

UC-02: View Account - Advanced View

Summary:	The Administrator navigates to the Advanced View tab in the View Account page to view advanced information of the user.
Priority:	Optional
Use Frequency:	Rarely
Direct Actors:	Admin: Tech Staff or Supertech Staff
Main Success Scenario:	- Upon a successful search, the Admin clicks on the View Account link (view icon) of the user. - Select the Advanced View tab. - See Advanced Information of the user account.

UC-03: View Account - Account Log

Summary:	The Administrator navigates to the Account Log tab in the View Account page to view history of the account.
-----------------	---

Priority:	Optional
Use Frequency:	Rarely
Direct Actors:	Admin: Tech Staff or Supertech Staff
Main Success Scenario:	1. Upon a successful search, the Admin clicks on the View Account link (view icon) of the user. 2. Select the Account Log tab. 3. See history of the user account.

UC-04: View Account - Changepass

Summary:	The Administrator navigates to the Changepass tab in the View Account page to view password change history of the account.
Priority:	Optional
Use Frequency:	Rarely
Direct Actors:	Admin: Tech Staff or Supertech Staff
Main Success Scenario:	1. Upon a successful search, the Admin clicks on the View Account link (view icon) of the user. 2. Select the Changepass tab. 3. See history of the password changes of the user account.

UC-05: View Account - Lostpass

Summary:	The Administrator navigates to the Lostpass tab in the View Account page to view password recovery history of the account.
Priority:	Optional
Use Frequency:	Rarely
Direct Actors:	Admin: Tech Staff or Supertech Staff
Main Success Scenario:	1. Upon a successful search, the Admin clicks on the View Account link (view icon) of the user. 2. Select the Lostpass tab. 3. See history of the password recovery of the user

	account
--	---------

UC-06: Edit Account

Summary:	The Administrator navigates to the Edit Account Page to update user information
Priority:	Essential
Use Frequency:	Rarely
Direct Actors:	Admin: Tech Staff or Supertech Staff
Main Success Scenario:	1. Visit Edit Account page as Admin 2. Update information of the user. 3. Submit form to review changes. 4. Review Changes to be made. 5. Submit form to save changes.
Alternative Scenario Extensions:	• From the View Account Page, the Admin clicks on Edit Account button. • On the Merge liist, the Admin clicks on Edit Account icon.

UC-07: Reset Password

Summary:	The Administrator navigates to the Reset Password Page to change password of an account.
Priority:	Optional
Use Frequency:	Rarely
Direct Actors:	Admin: Tech Staff or Supertech Staff
Main Success Scenario:	1. Visit View Account page as Admin 2. Select Reset Password. 3. Provide a valid reason for reset. 4. Submit form to reset account's password.

UC-08: Disable/Enable Account

Summary:	The Administrator navigates to the Disable/Enable Account Page to ban an account.
----------	---

Priority:	Essential
Use Frequency:	Rarely
Direct Actors:	Admin: Tech Staff or Supertech Staff
Main Success Scenario:	1. Visit View Account page as Admin. 2. Select Disable/Enable Account. 3. Provide a valid reason for disable/enable. 4. Submit form to disable/enable account.
Alternative Scenario Extensions:	• If the account is not disabled, the operation will be to 'disable.' The Admin can select options for the disable operation. By default, all access will be disabled. • If the account is disabled, the operation will be to 'enable.' 'Enable' account will restore all access.

UC-09: Print Account

Summary:	The Administrator navigates to the Print Account Page to print account information.
Priority:	Expected
Use Frequency:	Rarely
Direct Actors:	Admin: Tech Staff or Supertech Staff
Main Success Scenario:	1. Visit Print Account page as Admin. 2. Select Print Options. 3. Provide a valid reason for disable/enable. 4. Submit form to print account information.
Alternative Scenario Extensions:	• After a password reset, the Admin is given the choice to print the new password.

UC-10: Resend Email

Summary:	The Administrator navigates to the View Account Page to press Resend Email button.
Priority:	Optional
Use Frequency:	Rarely

Direct Actors:	Admin: Tech Staff or Supertech Staff
Main Success Scenario:	1. Visit View Account page as Admin 2. Press Resend Email. 3. Welcome email will be resent to the user.

UC-11: Create employee Account

Summary:	The Administrator navigates to the Create Employee Account Page to create a new employee account.
Priority:	Essential
Use Frequency:	Rarely
Direct Actors:	Admin: Tech Staff or Supertech Staff
Main Success Scenario:	1. Visit Create Employee Account page as Admin. 2. Fill out the form with the required information. 3. Submit form to review information. 4. Review the information and go back if needed. 5. Submit form to create the account.

UC-12: Create Special Account

Summary:	The Administrator navigates to the Create Special Account Page to create a new special account.
Priority:	Essential
Use Frequency:	Rarely
Direct Actors:	Admin: Tech Staff or Supertech Staff
Main Success Scenario:	1. Visit Create Special Account page as Supertech Admin. 2. Fill out the form with the required information. 3. Submit form to review information. 4. Review the information and go back if needed. 5. Submit form to create the account.

UC-13: Create Multiple Accounts

Summary:	The Administrator navigates to the Create Multiple Accounts page to activate multiple accounts. A CSV
-----------------	---

	containing the accounts information is uploaded to process the creation
Priority:	Optional
Use Frequency:	Rarely
Direct Actors:	Admin: Tech Staff or Supertech Staff
Main Success Scenario:	1. Visit the Create Employee page as Supertech Admin. 2. Click the Choose File. 3. Navigate to the CSV file and select it. 4. Press the Upload and Process Multiple Accounts. 5. A review page is shown to present all the valid entries of the CSV. These entries comply with the account creation requirements. 6. Press the Activate Accountst to create and activate all listed account information. 7. After the activation of the accounts, the Print Information page will be shown just like the Print information page of single account creation. 8. Printing the account information will print it per account, containing all the selected Print Messages.

UC-14: Merge Accounts

Summary:	The Administrator navigates to the Merge Accounts Page to merge two accounts into one.
Priority:	Essential
Use Frequency:	Rarely
Direct Actors:	Admin: Tech Staff or Supertech Staff
Main Success Scenario:	1. Visit the Merge Accounts page as Supertech Admin. 2. Search the accounts to be merged. 3. Press the Add to Merge to add the account to merge list. 4. Repeat the above steps until two (2) accounts are on the merge list. 5. Press the Proceed with Merging to start the merge

	process 6. Select all attributes and values to be retained 7. Review final merge account info 8. Submit form to merge accounts.
Alternative Scenario Extensions:	From the View Account Page, the Admin clicks on the Add to Merge

UC-15: Add/Remove to/from Merge

Summary:	The Administrator navigates to the Merge Accounts Page to add or remove an account to or from the merge list.
Priority:	Essential
Use Frequency:	Rarely
Direct Actors:	Admin: Tech Staff or Supertech Staff
Main Success Scenario:	- Visit the Merge Accounts page as Supertech Admin. - To add to the list, search an account in the Merge Account Lookup search facility. Then press Add to Merge on the results for a specific account. - From the Merge Accounts List, press the Remove from Merge to remove the account from the list.

UC-16: Edit Account - Username

Summary:	The Administrator navigates to the Edit Account Page to update account username.
Priority:	Essential
Use Frequency:	Rarely
Direct Actors:	Admin: Tech Staff or Supertech Staff
Main Success Scenario:	- Visit Edit Account page as SuperTech admin. - Update username of the user. - Submit form to review changes. - Review Changes to be made. - Submit form to save changes.

Alternative Scenario Extensions:	• From the View Account Page, the Admin clicks on Edit Account button • On the Merge list, the Admin clicks on Edit Account icon
---	---

UC-17: Edit Account - Primary Email

Summary:	The Administrator navigates to the Edit Account Page to update account primary email address.
Priority:	Essential
Use Frequency:	Rarely
Direct Actors:	Admin: Tech Staff or Supertech Staff
Main Success Scenario:	1. Visit Edit Account page as SuperTech admin. 2. Update primary email of the user. 3. Submit form to review changes. 4. Review Changes to be made. 5. Submit form to save changes.
Alternative Scenario Extensions:	• From the View Account Page, the Admin clicks on Edit Account button. • On the Merge liist, the Admin clicks on Edit Account icon.

UC-18: Edit Account - Alternate Email

Summary:	The Administrator navigates to the Edit Account Page to update account alternate email address(es).
Priority:	Essential
Use Frequency:	Rarely
Direct Actors:	Admin: Tech Staff or Supertech Staff
Main Success Scenario:	1. Visit Edit Account page as SuperTech admin. 2. Update alternate email address(es) of the user. 3. Submit form to review changes. 4. Review Changes to be made. 5. Submit form to save changes.
Alternative Scenario	• From the View Account Page, the Admin clicks on

Extensions:	Edit Account button On the Merge liist, the Admin clicks on Edit Account icon
--------------------	---

UC-19: Edit Account - Mail Forwarding

Summary:	The Administrator navigates to the Edit Account Page to update account mail forwarding settings.
Priority:	Essential
Use Frequency:	Rarely
Direct Actors:	Admin: Tech Staff or Supertech Staff
Main Success Scenario:	- Visit Edit Account page as SuperTech admin. - Update mailforwarding of the user. - Submit form to review changes. - Review changes to be made. - Submit form to save changes.
Alternative Scenario Extensions:	- From the View Account Page, the Admin clicks on Edit Account button. - On the Merge liist, the Admin clicks on Edit Account icon.

UC-20: Search Mail Logs

Summary:	The Administrator navigates to the Search Mail Logs Page to search the current mail log file.
Priority:	Essential
Use Frequency:	Rarely
Direct Actors:	Admin: Tech Staff or Supertech Staff
Main Success Scenario:	- Visit Search Mail Logs page. - Enter query in the search field. - Submit form. - See search results.
Alternative Scenario Extensions:	- Exact match found, redirect to View Mail Log Entry. - Multiple results. The Admin selects the appropriate account for next operation.

UC-21: View Mail Log Entry

Summary:	The Administrator navigates to the Search Mail Logs Page to search the current mail log file
Priority:	Essential
Use Frequency:	Rarely
Direct Actors:	Admin: Tech Staff or Supertech Staff
Main Success Scenario:	1. Visit Search Mail Logs page. 2. Enter message ID. 3. Submit form. 4. View Mail Log Entry.
Alternative Scenario Extensions:	• From the search results of Search Mail Logs, redirect to View Mail Log Entry. • Multiple results. The Admin selects the appropriate account for next operation.

UC-22: Block/Unblock Email

Summary:	The Administrator navigates to the Block/Unblock Email Page to block/unblock access to the mail relay.
Priority:	Essential
Use Frequency:	Rarely
Direct Actors:	Admin: Tech Staff or Supertech Staff
Main Success Scenario:	1. Visit Block/Unblock Email page as Admin. 2. Type a query on the search field. 3. Select the account to be blocked/unblocked. 4. Provide a valid reason. 5. Submit form to block/unblock the email.
Alternative Scenario Extensions:	• If there are multiple search results, the Admin clicks on the block or unblock icon on the Action column of the list.

UC-23: View Email Status

Summary:	The Administrator navigates to the View Email Status page to view status of the email address.
-----------------	--

Priority:	Essential
Use Frequency:	Rarely
Direct Actors:	Admin: Tech Staff or Supertech Staff
Main Success Scenario:	1. Visit Search Mail Logs page. 2. Type the exact email address on the search field. 3. Submit form to view the status of the email address
Alternative Scenario Extensions:	• From the Block/Unblock Email results page, the Admin clicks on the View icon under the Action column.

UC-24: Block/Unblock Sender

Summary:	The Administrator navigates to the Search Mail Logs Page to block/unblock a sender of an email.
Priority:	Essential
Use Frequency:	Rarely
Direct Actors:	Admin: Tech Staff or Supertech Staff
Main Success Scenario:	1. Visit Search Mail Logs page. 2. Type a query on the search field. 3. See results page. 4. Select the view icon next to the matching result to view the email log entry. 5. Press the Block/Unblock Sender to block/unblock the sender of the email.
Alternative Scenario Extensions:	• If there are multiple search results, the Admin clicks on the block or unblock icon on the Action column of the list.

UC-25: Block/Unblock Recipient

Summary:	The Administrator navigates to the Search Mail Logs Page to block/unblock a recipient of an email.
Priority:	Essential

Use Frequency:	Rarely
Direct Actors:	Admin: Tech Staff or Supertech Staff
Main Success Scenario:	1. Visit Search Mail Logs page. 2. Type a query on the search field. 3. See results page. 4. Select the view icon next to the matching result to view the email log entry. 5. Press the Block/Unblock Recipient to block/unblock the Recipient of the email.

UC-26: Remove Emails

Summary:	The Administrator navigates to the View Email Status Page to remove emails of a blocked account.
Priority:	Essential
Use Frequency:	Rarely
Direct Actors:	Admin: Tech Staff or Supertech Staff
Main Success Scenario:	1. Visit the View Email Status page as Supertech Admin. 2. Check the status of the email address is Blocked. 3. Press Remove Emails to remove the emails of the blocked account from the mail queue.

UC-27: Block/Unblock Host

Summary:	The Administrator navigates to the Block/Unblock Host Page to block/unblock a mail host IP address.
Priority:	Essential
Use Frequency:	Rarely
Direct Actors:	Admin: Tech Staff or Supertech Staff
Main Success	1. Visit Block/Unblock Host page. 2. Type a query on the search field.

Scenario:	3. See results 4. Select the view icon next to the matching result to view the host log entry. 5. Press the Block/Unblock Recipient to block/unblock the host IP address
Alternative Scenario Extensions:	• From the View Host Log Entry page, the Admin clicks on the Block/Unblock button.

UC-28: View Host Log Entry

Summary:	The Administrator navigates to the Block/Unblock Host page to view status of the email address.
Priority:	Essential
Use Frequency:	Rarely
Direct Actors:	Admin: Tech Staff or Supertech Staff
Main Success Scenario:	1. Visit Block/Unblock Host page. 2. Type the exact IP address on the search field. 3. Submit form to view the status of the email address.
Alternative Scenario Extensions:	• From the Block/Unblock Host results page, the Admin clicks on the View icon under the Action column.

UC-30: Block/Unblock Client

Summary:	The Administrator navigates to the Block/Unblock Client page to block/unblock a Dilnet account on the proxy servers.
Priority:	Essential
Use Frequency:	Rarely
Direct Actors:	Admin: Tech Staff or Supertech Staff
Main Success Scenario:	1. Visit Block/Unblock Client page. 2. Type a query to search for the account. 3. See results. 4. If exact match, the Admin will be redirected to the View User Proxy Status page.

	5. Press block/unblock User. 6. Provide Reason 7. Submit form to block/unblock the account.
Alternative Scenario Extensions:	• From the Block/Unblock Client results page, the Admin clicks on the block/unblock icon under the Action column

UC-31: View User Proxy Status

Summary:	The Administrator navigates to the Block/Unblock Client page to block/unblock a Dilnet account on the radius server.
Priority:	Essential
Use Frequency:	Rarely
Direct Actors:	Admin: Tech Staff or Supertech Staff
Main Success Scenario:	1. Visit Block/Unblock Client page. 2. Type a query to search for the account. 3. See results. 4. Select the matching result.
Alternative Scenario Extensions:	• From the Block/Unblock Client results page, the Admin clicks on the view icon under the Action column.

UC-32: Radius Profile

Summary:	The Administrator navigates to the Radius Profile page to see the current usage of the proxy servers.
Priority:	Essential
Use Frequency:	Rarely
Direct Actors:	Admin: Tech Staff or Supertech Staff
Main Success Scenario:	1. Visit Radius Profile page. 2. See summary and other statistics.

UC-33: View User Radius Profile

Summary:	The Administrator navigates to the Radius Profile page to see the current usage of a specific user.
Priority:	Essential
Use Frequency:	Rarely
Direct Actors:	Admin: Tech Staff or Supertech Staff
Main Success Scenario:	1. Visit Radius Profile page. 2. Type the username on the search field. 3. See results page.

UC-40: Activate Student Account

Summary:	The Administrator navigates to the Activate Student Account page to activate a student account.
Priority:	Essential
Use Frequency:	Rarely
Direct Actors:	Admin: Tech Staff or Supertech Staff
Main Success Scenario:	1. Visit Activate Student Account page. 2. Type/Scan the student number or Form 5 barcode. 3. Confirm the information on the resulting page. 4. Submit form to activate account.
Alternative Scenario Extensions:	• If already activated, the Admin will be redirected to Reset Password Page. • If the student is also an employee, the Admin will be redirected to the View Account page.

UC-51: Accounts Statistics

Summary:	The Administrator navigates to the Accounts Statistic page to view statistics on DilNet accounts.
Priority:	Essential

Use Frequency:	Rarely
Direct Actors:	Admin: Tech Staff or Supertech Staff
Main Success Scenario:	1. Visit Dashboard page. 2. See current statistics on DilNet accounts.

Appendix G
FEATURE SET

Release Information

Project:	DilNet Helpdesk
Internal Release Number:	1.0.0
Related Documents:	Features, Use Cases, Use Case Suite

Features by Priority

- Release 1.0.0
 - Essential
 - F-00 Account Lookup
 - F-01 View Account - Basic
 - F-02 View Account - Advanced
 - F-03 View Account - Account Log
 - F-04 View Account - Changepass
 - F-05 View Account - Lostpass
 - F-06 Edit Account
 - F-07 Reset Password
 - F-08 Enable Account
 - F-09 Disable Account
 - F-11 Resend Email
 - F-12 Create Employee Account
 - F-13 Create Special Account
 - F-14 Merge Accounts
 - F-15 Add to Merge
 - F-16 Remove from Merge
 - F-17 Edit Account - Username
 - F-18 Edit Account - Primary Email
 - F-19 Edit Account - Alternate Mail
 - F-20 Edit Account - Mail Forwarding
 - F-21 Search Mail Logs
 - F-22 View Mail Log Entry
 - F-23 Block Email
 - F-24 Unblock Email
 - F-25 View Email Status
 - F-26 Block Sender
 - F-27 Unblock Sender
 - F-28 Block Recipient
 - F-29 Unblock Recipient
 - F-30 Remove Emails
 - F-31 Block Host
 - F-32 Unblock Host
 - F-33 View Host Log Entry
 - F-34 Block Client

- F-35 Unblock Client
- F-36 View User Proxy Status
- F-37 Radius Profile
- F-38 View User Radius Profile
- F-39 Activate Student Account
- F-40 Accounts Statistics
- F-41 Create Multiple Accounts
- Expected
 - F-10 Print Account

Features by Functional Area

- FUNCTIONAL AREA ONE
 - F-00 Account Lookup
 - F-01 View Account - Basic View
 - F-02 View Account - Advanced View
 - F-03 View Account - Account Log
 - F-04 View Account - Changepass
 - F-05 View Account - Lostpass
 - F-06 Edit Account
 - F-07 Reset Password
 - F-08 Enable Account
 - F-09 Disable Account
 - F-10 Print Account
 - F-11 Resend Email
 - F-17 Edit Account - Username
 - F-18 Edit Account - Primary Email
 - F-19 Edit Account - Alternate Mail
 - F-20 Edit Account - Mail Forwarding
- FUNCTIONAL AREA TWO
 - F-12 Create Employee Account
 - F-41 Create Multiple Accounts
- FUNCTIONAL AREA THREE
 - F-13 Create Special Account
- FUNCTIONAL AREA FOUR
 - F-14 Merge Accounts
 - F-15 Add to Merge
 - F-16 Remove from Merge
- FUNCTIONAL AREA FIVE
 - F-21 Search Mail Logs
 - F-22 View Mail Log Entry
 - F-23 Block Email
 - F-24 Unblock Email
 - F-25 View Email Status
 - F-26 Block Sender
 - F-27 Unblock Sender
 - F-28 Block Recipient
 - F-29 Unblock Recipient
 - F-30 Remove Emails
 - F-31 Block Host
 - F-32 Unblock Host

- F-33 View Host Log Entry
- FUNCTIONAL AREA SIX
 - F-34 Block Client
 - F-35 Unblock Client
 - F-36 View User Proxy Status
 - F-37 Radius Profile
 - F-38 View User Radius Profile
- FUNCTIONAL AREA SEVEN
 - F-39 Activate Student Account
- FUNCTIONAL AREA EIGHT
 - F-40 Accounts Statistics

Appendix H

FEATURES

Release Information

Project:	Enhanced Accounts Management
Internal Release Number:	1.0.0
Related Documents:	Feature Set, Use Cases, Use Case Suite

F-00: Account Lookup

Priority:	Essential
Effort:	Days
Risk:	Safe
Functional area(s):	Administration
Use case(s):	UC-00 UC-01 UC-02 UC-03 UC-04 UC-05 UC-06 UC-07 UC-08 UC-09 UC-10
Description:	Both admin types can search for accounts by entering a keyword in the search field and pressing Enter or the Search button. Details: • A minimum of three (3) characters is required for the search. • If the query is a valid DilNet account email, the Admin is automatically redirected to the View Account Page. • If the query is a valid student number, a student number search is performed. • If the query is a valid employee number, an employee number search is performed. • If the query is a valid Dilnet account username, the Admin is redirected to the View Account page for the account. • If all of the above fail, a fulltext search is done against the firstname, middlename, lastname, username, and email fields of the database. • Multiple search results is presented by a table with the following fields: username, name, email, id number and actions. • Exact match results or single matching result is automatically redirected to the View Account page of the account.

F-01: View Account - Basic View

Priority:	Essential
Effort:	Days
Risk:	Safe
Functional area(s):	Administration
Use case(s):	UC-00 UC-01 UC-13
Description:	Both admin types can view the basic information of the user. Details: • The following information are shown: ○ Account Information ■ Username ■ Primary Email ■ Alternate Email(s) ○ User Information ■ Full Name ■ College/Unit ■ Department/Division ■ Employee Number ■ Employee Type ■ Appointment Period ■ Student Number ■ Birthdate ■ Gender ○ Account Status ■ Last Login ■ Status ■ Date Activated ○ Mail Usage ■ Mail Host ■ Mail Forwarding Address ■ Mail Quota ○ Password Recovery ■ Security Last Updated ■ Recovery Email ■ Security Question Type ■ Recovery Feature Status ■ Recovery Count • Action Buttons above the information are also shown: ○ Edit Account ○ Reset Password ○ Add to Merge

	Disable Account Print Account Resend Email Go Back (depending on the page referer)
--	---

F-02: View Account - Advanced View

Priority:	Essential
Effort:	Hours
Risk:	Safe
Functional area(s):	Administration
Use case(s):	UC-00 UC-01 UC-13
Description:	Both admin types can view the advanced information of the user account. Details: • The information shown is a comparison table of the data stored in MySQL and LDAP database comparing the following fields: ○ Username ○ Fullname ○ Lastname ○ Firstname ○ Middle Name ○ Birthdate ○ ID Number ○ Student Number ○ Employee Number ○ Unit ○ Division ○ Course ○ Title ○ Mail • Action Buttons above the information are also shown: ○ Edit Account ○ Reset Password ○ Add to Merge ○ Disable Account ○ Print Account ○ Resend Email ○ Go Back (depending on the page referer)

F-03: View Account - Account Log

Priority:	Essential
Effort:	Hours

Risk:	Safe
Functional area(s):	Administration
Use case(s):	UC-00 UC-01 UC-13
Description:	Both admin types can view the Account log information of the user account. It shows the history of the processes or operations done on the account. Details: • The information shown is a formatted table with the following fields: ○ ID ○ Date/Time ○ Account ○ Detail ○ Service ○ Log Level ○ Changed by ○ IP • Action Buttons above the information are also shown: ○ Edit Account ○ Reset Password ○ Add to Merge ○ Disable Account ○ Print Account ○ Resend Email ○ Go Back (depending on the page referer)

F-04: View Account - Changepass

Priority:	Essential
Effort:	Hours
Risk:	Safe
Functional area(s):	Administration
Use case(s):	UC-00 UC-01 UC-13
Description:	Both admin types can view the Changepass information of the user account. It shows the history of password changes done on the account. Details: • The information shown is a formatted table with the following fields: ○ Username ○ IP

	○ Changed By ○ Date/Time ○ Notes ● Action Buttons above the information are also shown: ○ Edit Account ○ Reset Password ○ Add to Merge ○ Disable Account ○ Print Account ○ Resend Email ○ Go Back (depending on the page referer)
--	--

F-05: View Account - Lostpass

Priority:	Essential
Effort:	Hours
Risk:	Safe
Functional area(s):	Administration
Use case(s):	UC-00 UC-01 UC-13
Description:	Both admin types can view the Lostpass information of the user account. It shows the history password recovery done by the user. Details: ● The information shown is a formatted able with the following fields: ○ Username ○ Date Added ○ Active ● Action Buttons above the information are also shown: ○ Edit Account ○ Reset Password ○ Add to Merge ○ Disable Account ○ Print Account ○ Resend Email ○ Go Back (depending on the page referer)

F-06: Edit Account

Priority:	Essential
Effort:	Days

Risk:	Safe
Functional area(s):	Administration
Use case(s):	UC-00 UC-01 UC-02 UC-03 UC-04 UC-05 UC-13
Description:	Both admin types can edit account information. However, only Supertech types can edit certain fields. Details: • The information that can be edited by a regular Tech Staff: ○ Gender ○ Birthdate (required) ○ Unit/College ○ Division/Department ○ Designation ○ Student Number ○ Employee Number ○ Employee Type ○ Appointment Period ○ Mail Quota Size • Additionally, the SuperTech types can also edit the following: ○ Username ○ Primary email ○ Alternate emails ○ Mail forwarding address ○ First name ○ Middle Name ○ Last name • Action Buttons above the information are also shown: ○ View Account ○ Reset Password ○ Add to Merge ○ Disable Account ○ Print Account ○ Resend Email ○ Go Back (depending on the page referer)

F-07: Reset Password

Priority:	Essential
Effort:	Days
Risk:	Safe
Functional area(s):	Administration
Use case(s):	UC-01 UC-02 UC-03 UC-04 UC-05 UC-06

Description:	Both admin types can reset an account's password. Details: • Before an account password is reset, a reason is required from the Admin. • Also the full name of the user and his or her student number and/or employee number is shown to the Admin for review of accuracy. • After resetting the user's password, the Admin is present an option to print the new password receipt which is given to the account holder.
---------------------	---

F-08: Enable Account

Priority:	Essential
Effort:	Days
Risk:	Safe
Functional area(s):	Administration
Use case(s):	UC-01 UC-02 UC-03 UC-04 UC-05 UC-06
Description:	Both admin types can enable a disabled account. Details: • Enabling an account requires a reason to be filled by the Admin. • Enabling an account will restore the following: ○ Password Change ○ Mail Delivery ○ Proxy Access

F-09: Disable Account

Priority:	Essential
Effort:	Days
Risk:	Safe
Functional area(s):	Administration
Use case(s):	UC-01 UC-02 UC-03 UC-04 UC-05 UC-06
Description:	Both admin types can disable an account. Details: • Disabling an Account doesn't automatically change the password of the user. • By default access to the following will be disabled: ○ Password Change ○ Mail Delivery

	Proxy Access • The Admin may opt to not disable the following by deselecting their checkbox: Mail Delivery Proxy Access
--	--

F-10: Print Account

Priority:	Essential
Effort:	Days
Risk:	Safe
Functional area(s):	Administration
Use case(s):	UC-00 UC-01 UC-02 UC-03 UC-04 UC-05 UC-06
Description:	Both admin types can print an account information. Details: • The following field values will be printed: ○ Username ○ Full Name ○ Primary Mail ○ Alternate Email ○ Unit/College ○ Department/Division ○ Designation ○ Appointment Period ○ Employee Type ○ Employee Number ○ Student Number

F-11: Resend Email

Priority:	Essential
Effort:	Days
Risk:	Safe
Functional area(s):	Administration
Use case(s):	UC-00 UC-01 UC-02 UC-03 UC-04 UC-05
Description:	Both admin types can resend the welcome email of the user. Details: • Resending the welcome email

F-12: Create Employee Account

Priority:	Essential
-----------	-----------

Effort:	Days
Risk:	Safe
Functional area(s):	Administration
Use case(s):	UC-11
Description:	Both admin types can create an account for the employee. Details: • The Admin needs to provide the following information: ○ First Name (required) ○ Middle Name (required) ○ Last Name (required) ○ Account Type (required) ○ Gender (required) ○ Birthdate ○ Employee Number ○ Student Number ○ Unit/College (required) ○ Division/Department (required) ○ Designation (required) ○ Appointment Period ○ Mail Quota Size • For cases where the employee has no Middle Name, the No Middle Name checkbox must be checked to pass the checking of the Middle Name field. • For cases where the employee has no Last Name, the No Last Name checkbox must be checked to pass the checking of the Last Name field. • For cases where the employee has no Middle Name and No Last Name (yes, this can happen), both No Middle Name and No Last Name must be checked. The Bypass Special Case validation checkbox must also be checked to pass the double no name security check of the system.

F-13: Create Special Account

Priority:	Essential
Effort:	Days
Risk:	Safe
Functional area(s):	Administration
Use case(s):	UC-12
Description:	Both admin types can create a special account. Special accounts are usually created for official email of Units or Program that normally would not fit the normal account

	creation process. It can also be used for special cases where specific account details are needed to be created. Details: • The Admin needs to provide the following information: ○ First Name (required) ○ Middle Name (required) ○ Last Name (required) ○ Account Type (required) ○ Username (required) ○ Email (required) ○ Unit/College (required) ○ Division/Department (required) ○ Designation (required) ○ Mail Quota Size • For cases where the account has no Middle Name, the No Middle Name checkbox must be checked to pass the checking of the Middle Name field. • For cases where the account has no Last Name, the No Last Name checkbox must be checked to pass the checking of the Last Name field. • For cases where the account has no Middle Name and No Last Name, both No Middle Name and No Last Name must be checked. The Bypass Special Case validation checkbox must also be checked to pass the double no name check.
--	---

F-14: Merge Accounts

Priority:	Essential
Effort:	Days
Risk:	Safe
Functional area(s):	Administration
Use case(s):	UC-13 UC-14 UC-15
Description:	Merging accounts is the process of merging two accounts into one. Only a Supertech Admin type can merge two accounts. Details: • To Merge Accounts, the accounts must first be known and added to the Merge Accounts list. • The Merge Accounts page has the facility to do the merging of accounts. • The Merge Accounts page has an Account Lookup facility to look for the accounts. • Once an account has been found, the Add to

	Merge can be pressed to add the selected account to the Merge Accounts List. • Once two accounts are in the list, the Merging process can be started. • On the Merge Accounts Preparation page, the Admin is asked to select all the attribute values that will be retained on the final account. Once a complete selection is done, the Admin is redirected to the Merge Review page. • At the Merge Review page, the Admin may review the proposed changes. Once the Admin press the Save button, a final prompt is presented. • Once the final prompt is confirmed, the merging will proceed and the Admin will be redirected to the View Account page of the new account. • The Admin can only select one value for each attribute except for the Alternate Emails field.
--	--

F-15: Add to Merge

Priority:	Essential
Effort:	Hours
Risk:	Safe
Functional area(s):	Administration
Use case(s):	UC-00 UC-01 UC-02 UC-03 UC-04 UC-05 UC-13 UC-14
Description:	Add to Merge is the process where an account is selected for merging. Only a Supertech Admin type can use Add to Merge. Details: • An account added to merge will be listed in the Merge Accounts List. • On the Search DiINet Accounts results page an account can be added to the merge list by pressing the icon that says Add to Merge. • On the Merge Accounts Lookup results page an account can be added to the merge list by pressing the icon that says Add to Merge.

F-16: Remove from Merge

Priority:	Essential
Effort:	Hours
Risk:	Safe
Functional area(s):	Administration
Use case(s):	UC-13 UC-15
Description:	Remove from Merge is the process where an account is

	removed from the Merge Accounts list. Only a Supertech Admin type can use Remove from Merge. Details: In the Merge Accounts List, an account can be removed by pressing the Remove from Merge button link.
--	---

F-17: Edit Account - Username

Priority:	Essential
Effort:	Hours
Risk:	Safe
Functional area(s):	Administration
Use case(s):	UC-06 UC-16
Description:	In the Edit Account Page, a Supertech Admin can edit the username of the account. Details: The username can be changed to any username that is not currently owned by any account.

F-18: Edit Account - Primary Email

Priority:	Essential
Effort:	Hours
Risk:	Safe
Functional area(s):	Administration
Use case(s):	UC-06 UC-17
Description:	In the Edit Account Page, a Supertech Admin can edit the primary email of the account. Details: - The primary email address can be changed by a Supertech Admin. - The primary email address must be a valid DilNet account email address.

F-19: Edit Account - Alternate Email

Priority:	Essential
Effort:	Hours
Risk:	Safe
Functional area(s):	Administration
Use case(s):	UC-06 UC-18

Description:	In the Edit Account Page, a Supertech Admin can edit the alternate email of the account. Details: • The alternate email address can be changed by a Supertech Admin. • Alternate emails must be a valid DilNet account email address. • Each email must be separated by a new line.
---------------------	---

F-20: Edit Account - Mail Forwarding

Priority:	Essential
Effort:	Hours
Risk:	Safe
Functional area(s):	Administration
Use case(s):	UC-06 UC-19
Description:	In the Edit Account Page, a Supertech Admin can edit the mail forwarding settings of a user. Details: • When saved, the user will no longer receive email in his/her DilNet account inbox. Instead, the emails will be forwarded to the mail forwarding address set. • A valid email address must be set as a mail forwarding address. • The mail forwarding address must not be set to another DilNet account.

F-21: Search Mail Logs

Priority:	Essential
Effort:	Hours
Risk:	Safe
Functional area(s):	Administration
Use case(s):	UC-20
Description:	Both admin types can search the current mail log file of the relays. Details: • The search speed depends on the search query. • A valid email address must be set as a mail forwarding address. • Search queries can be anything from email address, message ID or even subject. • A minimum length of three (3) characters is

	required for the search query.
--	--------------------------------

F-22: View Mail Log Entry

Priority:	Essential
Effort:	Hours
Risk:	Safe
Functional area(s):	Administration
Use case(s):	UC-20 UC-21
Description:	Both admin types can view the log entry of a specific email. Details: • The log entry shows the following: ○ Message ID ○ Subject ○ Recipient ○ Sender ○ Sender Host ○ Start Time ○ End Time ○ Status • Action buttons below the information are also shown: ○ Block Sender ○ Block Sender IP ○ Block Recipient

F-23: Block Email

Priority:	Essential
Effort:	Hours
Risk:	Safe
Functional area(s):	Administration
Use case(s):	UC-08 UC-21 UC-22 UC-23
Description:	Both admin types can block an email address on the mail relays Details: • Blocked email address won't be able to send and receive emails within the mail relays facing the internet. • A reason is required before an email address can be blocked.

F-24: Unblock Email

Priority:	Essential
------------------	-----------

Effort:	Hours
Risk:	Safe
Functional area(s):	Administration
Use case(s):	UC-08 UC-21 UC-22 UC-23
Description:	Both admin types can unblock an email address on the mail relays. Details: • Unblocking an email will allow the email address to send/receive mails to/from the mail relays facing the internet. • A reason is required before an email address can be unblocked.

F-25: View Email Status

Priority:	Essential
Effort:	Hours
Risk:	Safe
Functional area(s):	Administration
Use case(s):	UC-22 UC-23
Description:	Both admin types can view the status of a specific email address whether or not it's blocked in the mail relays. Details: • The following fields will be shown: ○ Email Address ○ Status ○ Date Banned ○ Reason for Ban ○ Banned by ○ Date Unbanned ○ Reason for Unban ○ Unbanned by • Possible action buttons are also shown below ○ For blocked emails: ■ Unblock email ■ Remove emails ○ For unblocked emails: ■ Block Email

F-26: Block Sender

Priority:	Essential
Effort:	Hours
Risk:	Safe
Functional area(s):	Administration

Use case(s):	UC-20 UC-21 UC-24
Description:	Both admin types can block an email address from sending on the mail relays. Details: • Just like in Block email, the sender email address will be blocked from the mail relays. • The sender address won't be able to send nor receive from the mail relays. • A reason is required before a sender email address can be blocked.

F-27: Unblock Sender

Priority:	Essential
Effort:	Hours
Risk:	Safe
Functional area(s):	Administration
Use case(s):	UC-20 UC-21 UC-24
Description:	Both admin types can unblock an email address blocked on the mail relays. Details: • Just like in Unblock email, the sender email address will be unblocked on the mail relays. • The sender address will be able to send and receive again emails to and from the mail relays. • A reason is required before a sender email address can be unblocked.

F-28: Block Recipient

Priority:	Essential
Effort:	Hours
Risk:	Safe
Functional area(s):	Administration
Use case(s):	UC-20 UC-21 UC-25
Description:	Both admin types can block an email address from receiving on the mail relays. Details: • Just like in Block email, the recipient email address will be blocked on the mail relays. • The recipient address won't be able to receive from the mail relays. • A reason is required before a recipient address can be blocked.

F-29: Unblock Recipient

Priority:	Essential
Effort:	Hours
Risk:	Safe
Functional area(s):	Administration
Use case(s):	UC-20 UC-21 UC-25
Description:	Both admin types can unblock an email address blocked on the mail relays. Details: • Just like in Unblock email, the recipient email address will be unblocked on the mail relays. • The recipient address will be able to send and receive again emails to and from the mail relays. • A reason is required before a recipient address can be unblocked.

F-30: Remove Emails

Priority:	Essential
Effort:	Hours
Risk:	Safe
Functional area(s):	Administration
Use case(s):	UC-23 UC-26
Description:	A Supertech staff can remove the emails currently on queue of a blocked email address. Details: • All the emails from the blocked email address is removed from the incoming mail queue of the mail relays.

F-31: Block Host

Priority:	Essential
Effort:	Hours
Risk:	Safe
Functional area(s):	Administration
Use case(s):	UC-27 UC-28
Description:	Both admin types can block an email host ip address from sending/receiving on the mail relays. Details: • The IP address of the email host will be blocked on the mail relays. • Blocked Host IP addresses can't initiate mail

	transactions on the mail relays. • A reason is required before a host address can be blocked.
--	--

F-32: Unblock Host

Priority:	Essential
Effort:	Hours
Risk:	Safe
Functional area(s):	Administration
Use case(s):	UC-27 UC-28
Description:	Both admin types can unblock an email host ip address from sending/receiving on the mail relays. Details: • The IP address of the email host will be unblocked on the mail relays. • Unblocked hosts can once again initiate mail transactions on the mail relays. • A reason is required before a host address can be unblocked.

F-33: View Host Log Entry

Priority:	Essential
Effort:	Hours
Risk:	Safe
Functional area(s):	Administration
Use case(s):	UC-27 UC-28
Description:	Both admin types can view the current status of a mail host IP address on the mail relays. Details: • The following details are shown: ○ Host IP address ○ Status ○ Date Banned ○ Reason for Ban ○ Banned by ○ Date Unbanned ○ Reason for Unban ○ Unbanned by • Action button is shown below ○ Block IP for non-blocked IP addresses ○ Unblock IP for blocked IP addresses

F-34: Block Client

Priority:	Essential
Effort:	Hours
Risk:	Safe
Functional area(s):	Administration
Use case(s):	UC-30 UC-31
Description:	Both admin types can block a Dilnet account from accessing the proxy servers thru the radius server. Details: • The username must be a valid Dilnet account. • A reason must be provided.

F-35: Unblock Client

Priority:	Essential
Effort:	Hours
Risk:	Safe
Functional area(s):	Administration
Use case(s):	UC-30 UC-31
Description:	Both admin types can unblock a Dilnet account currently blocked on the proxy servers. Details: • The username must be a valid Dilnet account. • A reason must be provided.

F-36: View User Proxy Status

Priority:	Essential
Effort:	Hours
Risk:	Safe
Functional area(s):	Administration
Use case(s):	UC-31
Description:	Both admin types can view the current status of a DilNet account on the proxy servers. Details: • The following information are shown: ○ Username ○ Status ○ Date Banned ○ Reason for Ban ○ Banned by ○ Date Unbanned ○ Reason for Unban

	○ Unbanned by ● Action buttons are shown depending on the status of the user. ○ Block user for unblocked accounts. ○ Unblock user for currently blocked accounts.
--	--

F-37: Radius Profile

Priority:	Essential
Effort:	Hours
Risk:	Safe
Functional area(s):	Administration
Use case(s):	UC-32
Description:	Both admin types can view the current radius profile of all DilNet accounts who use or are currently using the radius server. Details: ● The following information are shown: ○ Total Input Bytes ○ Total Output Bytes ○ Online Users ■ Username ■ IP address ■ Start Time ○ Users Today ■ Username ■ IP address ■ Start Time ■ ● Action buttons are shown depending on the status of the user. ○ Block user for unblocked accounts. ○ Unblock user for currently blocked accounts.

F-38: View User Radius Profile

Priority:	Essential
Effort:	Hours
Risk:	Safe
Functional area(s):	Administration
Use case(s):	UC-33
Description:	Both admin types can view the current radius profile of a specific DilNet account.

	Details: • The following information are shown: ○ ID ○ IP Address ○ Start Time ○ Stop Time ○ Terminate Cause ○ Input bytes ○ Output bytes
--	--

F-39: Activate Student Account

Priority:	Essential
Effort:	Hours
Risk:	Safe
Functional area(s):	Administration
Use case(s):	UC-40
Description:	Both admin types can activate a new DilNet account. Details: • Either a valid student number or a valid Form 5 barcode number is required for activation. • Valid cases will show the details of the student ○ Full Name ○ Student Number ○ Unit ○ College ○ Course • Existing accounts for students will change to reset password. • Existing accounts for employees will redirect to View Account page.

F-40: Accounts Statistics

Priority:	Essential
Effort:	Hours
Risk:	Safe
Functional area(s):	Administration
Use case(s):	UC-51
Description:	Both admin types can view the Dashboard which shows accounts statistics. Details: • The page is a slow loading page due to calculations done. • The following are shown:

	New accounts statistics Reset password statistics Lostpass statistics Student accounts created Accounts type created
--	--

F-41: Create Multiple Accounts

Priority:	Essential
Effort:	Days
Risk:	Safe
Functional area(s):	Administration
Use case(s):	UC-13
Description:	A supertech staff can create multiple accounts for employees. Details: • A CSV file containing the information of the accounts to be created is uploaded using the file upload option in the Create Employee Account page. • The CSV file consists of the following fields: ○ Last Name ○ First Name ○ Middle Name ○ Unit ○ Division ○ Designation ○ Type ○ Employee Number • This feature assumes each account to be created is of permanent status. If this is not the case, the employee information must not be included in the CSV file as per policy. • If at least one of the above fields are missing or invalid, the line entry is discarded. A notice is shown how many entries were discarded. Only valid entries will be processed for account creation. • Cases where there are no lastnames or middle names must not be created through this facility. • The following are considered valid entries for each field: ○ Last name – alphabet type. Spaces, apostrophes, underscores are ignored. ○ First name – alphabet type. Spaces,

	apostrophes, underscores are ignored. Middle Name – alphabet type. Spaces, apostrophes, underscores are ignored. Unit – alphabet type. Spaces, apostrophes, underscores are ignored. • Division – alphabet type. Spaces, apostrophes, underscores are ignored. • Designation – must not be empty. • Type – alphabet characters. • Employee Number - must be numeric
--	--

Appendix I

USER MANUAL

Login

To login, one must go to the website and enter his or her DilNet account credentials in the login form as shown below:

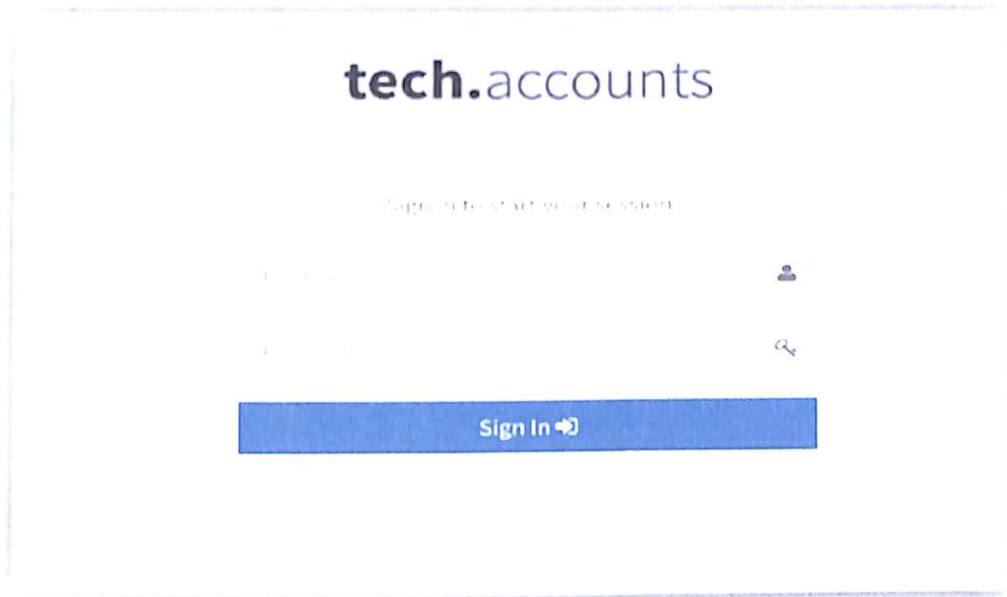


Figure I-1 Login Page

To log out of the system, an authenticated user can press the logout button in the upper right part of the site:

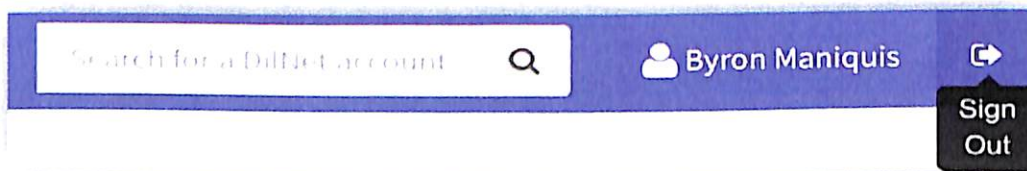


Figure I-2 Sign Out Link

On admin's successful login, he or she will be redirected to the **Search DilNet Accounts** page. This is the default behavior. A screenshot of this page is shown below:

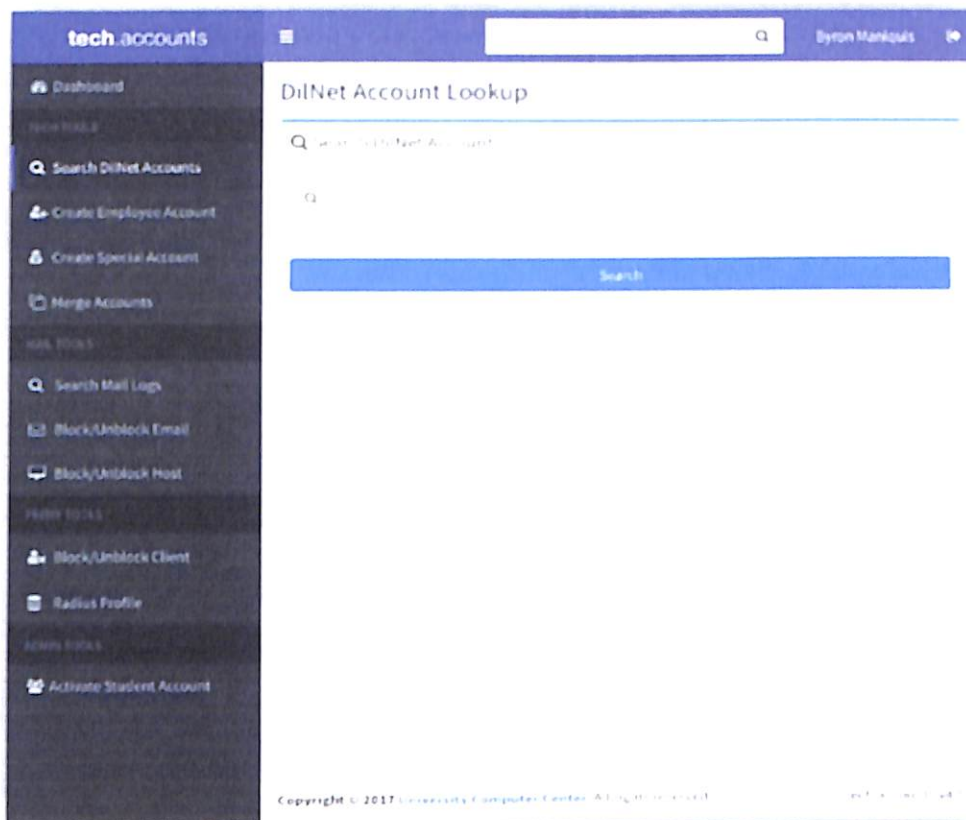


Figure I-3 DilNet Account Lookup

Tech Tools

Tech Tools is a collection of DilNet Account tools for creating, editing or merging DilNet Accounts.

Tech tools is also where a DilNet account can be disabled. Disabled accounts will lose their access on the password change, mail delivery and proxy service depending on the restrictions imposed by the admin.

The following are its main sections/features and how to use them.

Search DilNet Accounts

The Search DilNet Accounts page is the default page when logging in to the new Tech.Accounts website. To perform operations or actions on a Dilnet account, it must be first searched here.

The Search DilNet Accounts page is the enhanced version of the old DilNet account lookup. In this iteration, the admin no longer need to memorize formats or even search fields. Only one search field is used for all the search operations.

To use the Search DilNet Accounts page, the Admin must provide a search query which can be one of the following:

- Name or part of it (e.g. Juan Dela Cruz, Juan Dela)

A sample name search result for the above is shown below:

Figure I-4 Name Search Result

- Username or part of it (e.g. jadelacruz, jadelacr)

A sample username search result is shown below:

Figure I-5 Username Search Result

- An email address in the form **jadelacruz@upd.edu.ph**

The results page has the following columns:

- Username - username of the account
- Name - full name of the account holder
- Email - primary email of the account
- ID Number - employee number or student number of the account, whichever is available
- Actions - shortcuts for managing the account

In the Actions column, the following actions are available:

- View Account
- Edit Account
- Print Account
- Add to Merge

If only one matching result is found, the **View Account** page of the matching account is shown. A sample screenshot is shown below:

View DilNet Account

Found exact match for 'jadelacruz'.

JUAN DELA CRUZ

(J. D. Cruz)

[Edit](#) [Add](#) [Remove](#) [Share](#) [Print](#) [Export](#)

Account Information

Username	jadelacruz
Primary Email Address	juan.delacruz@upd.edu.ph
Alternate Email Address	jadelacruz@upd.edu.ph

User Information

Last First Middle	De la Cruz Juan D
Unit / College	BSA
Division / Department	BSA

Figure I-6 View DilNet Account page

- A student number in the form **2017-12345** or **201712345** or part of it.
- An employee number in the form **0123456789** or **012345678**.

In the **View Account** page, there are four (4) tabs:

- Basic View
 - Shows basic information of the user account.
- Advanced View
 - Shows LDAP and MySQL db information of the user account.
- Account Log
 - Shows the account operations log history usually from the moment the account was activated or created. These operations are either done by an Admin or the system itself. Account activities like recovery email setup and mail forwarding are also logged here.

- **Changepass**
 - Shows the password change history of the account. It includes password changes done by the Admin.
- **Lostpass**
 - Shows the account history on using the lostpass facility.

The following are the actions available when an account is viewed using the **View Account** tool:

- **Edit Account**
Allows the Admin to update or change information of the account.
- **Reset Password**
Allows the Admin to change the password of the user.
- **Add To Merge**
Allows the Admin to add the account to a merge list.
- **Disable Account**
Allows the Admin to disable the account on one or more DilNet services.
- **Print Account**
Allows the Admin to print an information receipt for the user.
- **Resend Email**
Allows the Admin to resend the Welcome email to the account holder.

Create Employee Account

The **Create Employee Account** tool is used for creating accounts for regular employees of the university. Regular employees are those with permanent employment status.

New Direct Account

⚠

First Name Middle Name Last Name No Middle Name
No Last Name

Account Type Gender Birth Date

Employee Number

Student Number

Unit

Division

Designation

Figure I-7 Create Employee Account page

To create an employee account all the required fields must be filled. The required fields are:

- First Name
- Middle Name
- Last Name
- Account Type – can be Faculty, REPS, or Admin for administrative personnel
- Gender
- Unit (or College)
- Division (or Department)
- Designation

The default Mail Quota for Employee Accounts is 400Mb.

The following are the cases where the Bypass special case validation checkbox must be checked in order to proceed with the account creation:

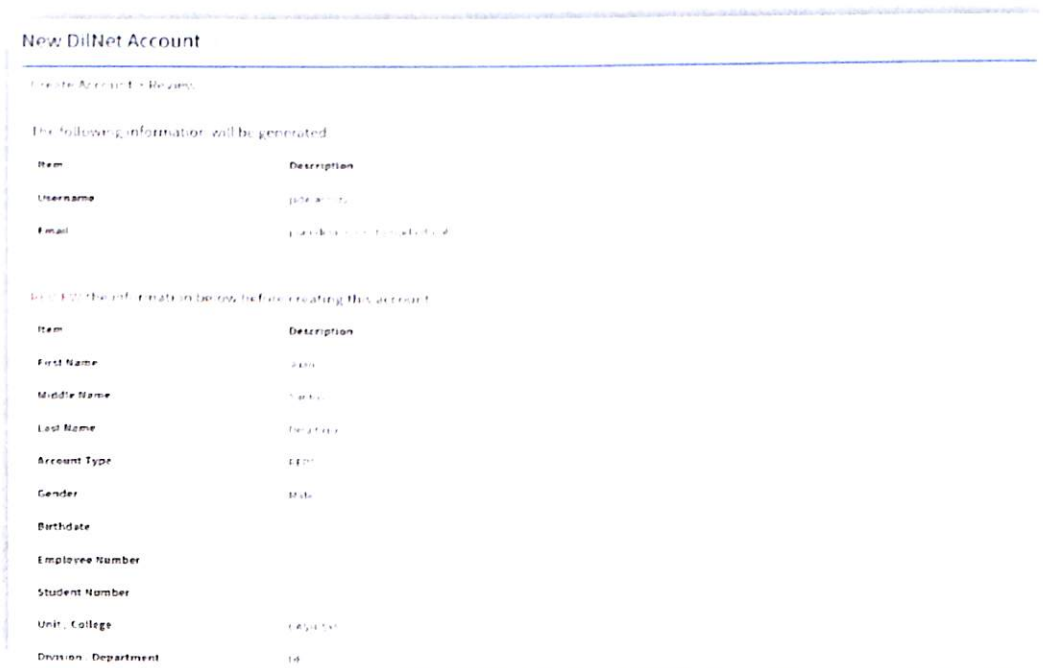


A screenshot of a web form. It features a single checkbox with the text "Bypass special case validation." to its right. Below the checkbox is a blue rectangular button with the word "Submit" in white text.

Figure I-8 Bypass special case validation checkbox

1. If there is already an account with the exact same name. The Admin must determine if it's a duplicate account or simply same names.
2. If both No Middle Name and No Last Name fields are checked.
3. If there's an account with the same employee number or student number.

When the account form passes the validation checks, the review page will be presented to the Admin for one last check:



A screenshot of a web page titled "New DilNet Account". Below the title is a link "Create Account > Review". The main content area is titled "The following information will be generated:" and contains two tables. The first table shows generated information: Username (johndoe123), Email (johndoe123@stateofcalifornia.gov), and Password (1234567890). The second table, preceded by the instruction "Be sure the information below is correct before creating this account.", shows account details: First Name (John), Middle Name (Michael), Last Name (Doe), Account Type (DIPN), Gender (Male), Birthdate (1980-01-01), Employee Number (123456789), Student Number (987654321), Unit / College (University of California), and Division / Department (State of California).

Item	Description
Username	johndoe123
Email	johndoe123@stateofcalifornia.gov

Be sure the information below is correct before creating this account.

Item	Description
First Name	John
Middle Name	Michael
Last Name	Doe
Account Type	DIPN
Gender	Male
Birthdate	1980-01-01
Employee Number	123456789
Student Number	987654321
Unit / College	University of California
Division / Department	State of California

Figure I-9 New DilNet account review page

The admin must then press the Create Account button to proceed with the actual creation.

After creation, the admin may print the account details on the connected thermal printer:

Confirm Print Information

DINET ACCOUNT : NEW ACCOUNT

Username

jsdelacruz2

Password

!4cEYHdJ

Name

Juan Delacruz

Mail

juan.delacruz1@upd.edu.ph

Mail Alt

jsdelacruz2@upd.edu.ph

Emp Type

REPS

Select Printer

Tech Printer IM-T82 Thermal

Print Messages

Change Password

Faculty Proxy

Online Services

Dinet Helpdesk

CRS Tagging

iLib Access

Print Receipt

Go Back

Figure I-10 Print Receipt page

The **Select Printer** dropdown menu allows the admin to select which printer to use for the receipt.

In the **Print Messages** section, several options are available which when selected will print the corresponding additional information on the receipt. The information are:

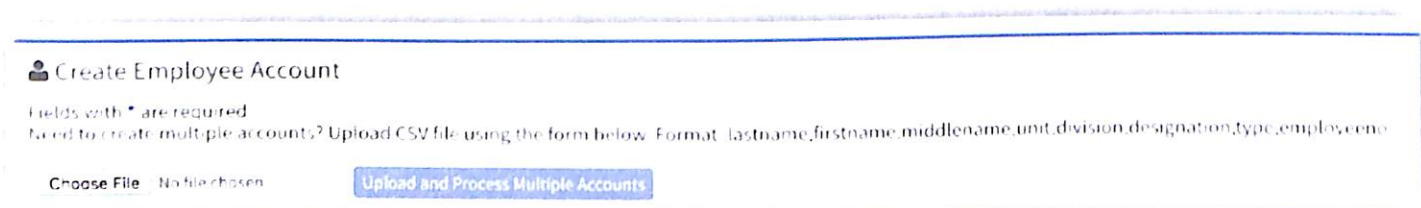
- Change Password - this prints the link where the user can changer his or her password.

- Faculty Proxy - information on the faculty proxy server that can be used for unrestricted access.
- Online Services - lists the online services where the DilNet account can be used.
- Dilnet Helpdesk - information on how to contact the DilNet helpdesk and what services they support.
- CRS Tagging - information on CRS Support for CRS related concerns.
- iLib Access - information on how to contact the iLib Support for library related concerns.

Create Multiple Employee Accounts

This tool allows the admin to create multiple employee accounts simultaneously. Just like the normal Create Employee Account, it must be used for creating accounts for regular employees of the university. Regular employees are those with permanent employment status.

To create multiple employee accounts, a CSV file containing their information must be uploaded using the upload file option.



The screenshot shows a web form titled "Create Employee Account" with a user icon. Below the title, it says "Fields with * are required". A question "Need to create multiple accounts?" is followed by the instruction "Upload CSV file using the form below. Format: lastname.firstname.middlename.unit.division.designation.type.employee.no.". There are two buttons: "Choose File" (which shows "No file chosen") and "Upload and Process Multiple Accounts".

Figure I-11 Upload CSV for multiple account creation

The CSV file requires each line of entry to have the following field values:

- Last Name
- First Name

- Middle Name
- Unit (or College)
- Division (or Department)
- Designation
- Type
- Employee Number

All the fields are required and one missing or invalid field value will render the whole line entry to be ignored for processing.

A review page is shown after uploading and validating the CSV file.

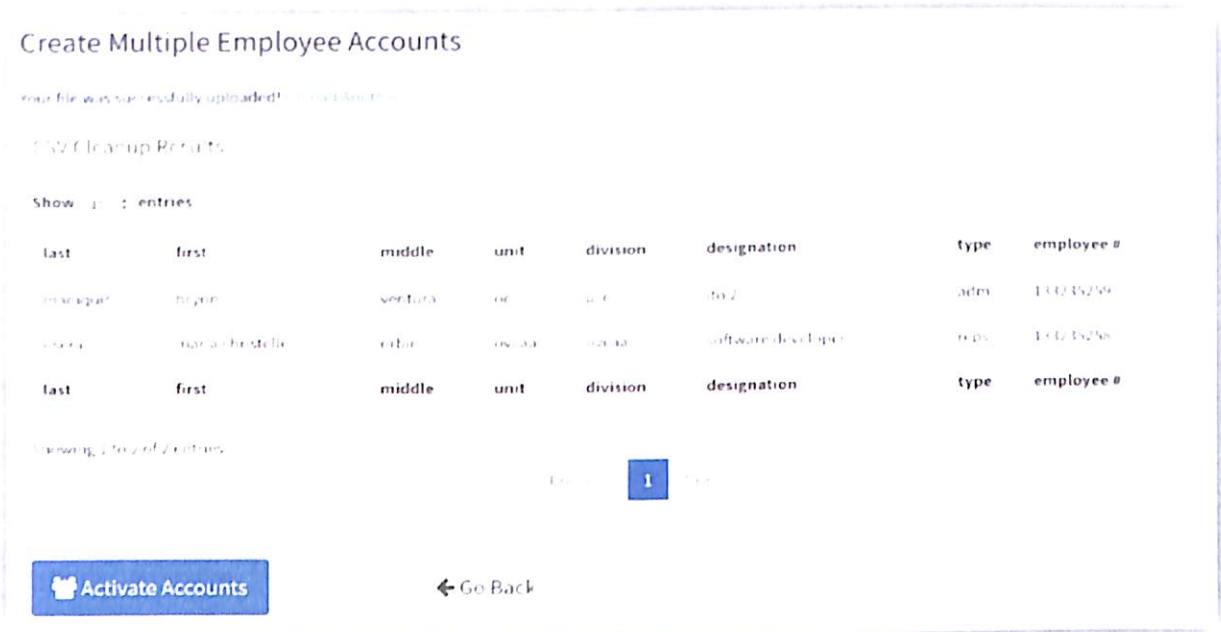


Figure I-12 CSV Cleanup Results for Multiple Account Creation showing valid entries

All the listed entries are valid ones. Once the Activate Accounts button is pressed by the Admin, the creation and activation of the accounts commences.

After creation, the admin may print the account details on the connected thermal printer:

Confirm Print Information

DILNET ACCOUNT: NEW ACCOUNTS

Username bvmaniquisl

Password drkrPQDt

Name Bryon Maniquis

Mail bryon.maniquis@upd.edu.ph

Mail_Alt bvmaniquisl@upd.edu.ph

Emp_Type ADM

Username mooseral

Password 6bpJUbXJ

Name Maria Christelle Osera

Mail maria_christelle.o.osera@upd.edu.ph

Mail_Alt mooseral@upd.edu.ph

Emp_Type REPS

Select Printer

Tech Printer TM-T82 (Thermal) ▼

Print Messages

Change Password 

Faculty Proxy 

Online Services 

Dilnet Helpdesk 

CRS Tagging 

iLib Access 

Figure I-13 Multiple Account Creation Print Receipt page

The **Select Printer** dropdown menu allows the admin to select which printer to use for the receipt. This is similar to Create Employee Account facility.

Just like in normal Create Employee Account, several options are available which when selected will print the corresponding additional information on the receipt. The information are:

- Change Password - this prints the link where the user can change his or her password.
- Faculty Proxy - information on the faculty proxy server that can be used for unrestricted access.
- Online Services - lists the online services where the DilNet account can be used.
- Dilnet Helpdesk - information on how to contact the DilNet helpdesk and what services they support.
- CRS Tagging - information on CRS Support for CRS related concerns.
- iLib Access - information on how to contact the iLib Support for library related concerns.

Printing the account information will print individual account information along with the selected print messages.

Create Special Account

The Create Special Account tool is used for creating accounts that normally don't follow the naming convention. A screenshot is shown below:

Create Special Account

8

First Name *	Middle Name *	Last Name *	No Middle Name No Last Name
Account Type *			
Username *			
Email *			
Unit *			
Division *			
Designation *			

Figure I-14 Create Special Account page

In the Create Special Account page, the Admin is required to designate the email and the username that will be assigned to the account.

The following fields are required in the **Create Special Account** page:

- First Name
- Middle Name
- Last Name
- Username
- Email
- Unit (or College)
- Division (or Department)
- Designation

The username field does not allow email format. A uniqueness check is done to prevent duplicates.

The email field accepts entries with or without @upd.edu.ph. A uniqueness check is performed to prevent duplicates.

The default Mail Quota for Special accounts is 800Mb.

Just like in Create Employee Account page, a review page is presented to the Admin for one final check.

The Bypass special case validation is also present here for cases similar to Create Employee Account page.

After the creation of the account, the Admin has the option to print an account receipt.

Merge Accounts

The Merge Accounts page is the tool used for merging two DilNet accounts into one. One of the two accounts will remain and the other will be deleted afterwards. The Admin is tasked to choose which attributes to retain and which to discard for the final merged account.

Two accounts are needed for the merge accounts process. In the Merge accounts page, the Admin can search for these accounts.

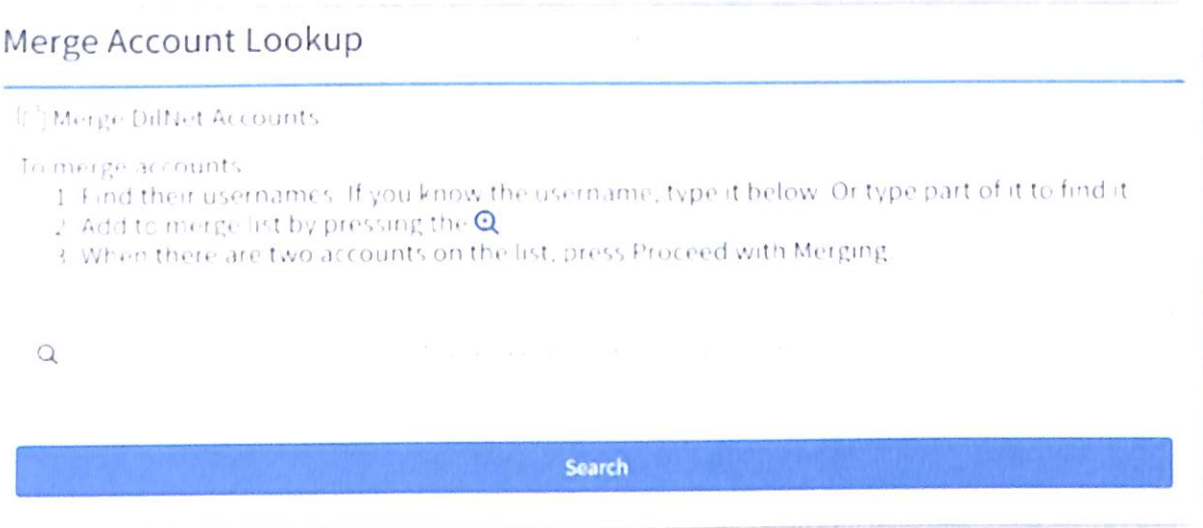


Figure I-15 Merge Account Lookup page

After finding the accounts, the Admin may press the Add to merge on each account to add them to the merge list.

Adding two accounts to the merge list will enable the Admin to merge the two accounts. The merge list looks like:

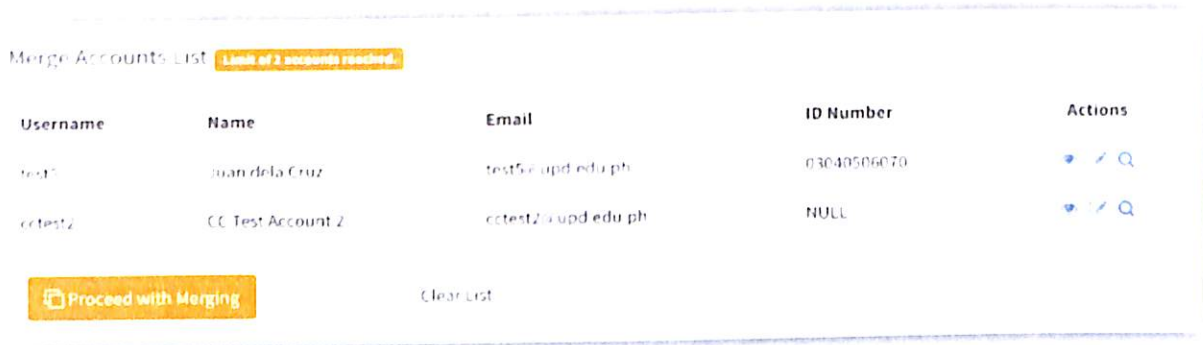


Figure I-16 Merge Accounts List

To proceed with the merging, the Proceed with Merging button must be pressed. The Admin can press the Clear List button to start over.

The Merge Accounts Preparation page is the next step in merging accounts. Here, the Admin is presented with all the details that can be merged. The Admin must choose which values to retain and which to discard. Except for the Alternate Emails, the Admin can only choose one value for each of the attributes available. A sample Merge Accounts Preparation is shown below:

Merge DilNet Accounts

Merge Accounts Preparation

Please select the values that will be retained.
For the Alternate Email, multiple addresses can be selected.
All fields are required

	Account #1 (test5)	Account #2 (cctest2)
Username	test5	cctest2
Full Name	Juan Delacruz	CC Test Account 2
Primary Email	test5@upf.edu.ph	cctest2@upf.edu.ph
Alternate Email	test5@upf.edu.ph	cctest2@upf.edu.ph
Employee Type	FAC	OFc
Employee Number	000454070	[No Data]
Student Number	2012-8999	[No Data]
Unit	EE	EE
Department	CC	CC
Designation	MAP's Test Account	test account
Continue Go Back		

Figure I-17 Merge Accounts Preparation page

After making the values selection, the Admin is presented with the Merge Accounts Review page showing the selected values one last time before merging the accounts.

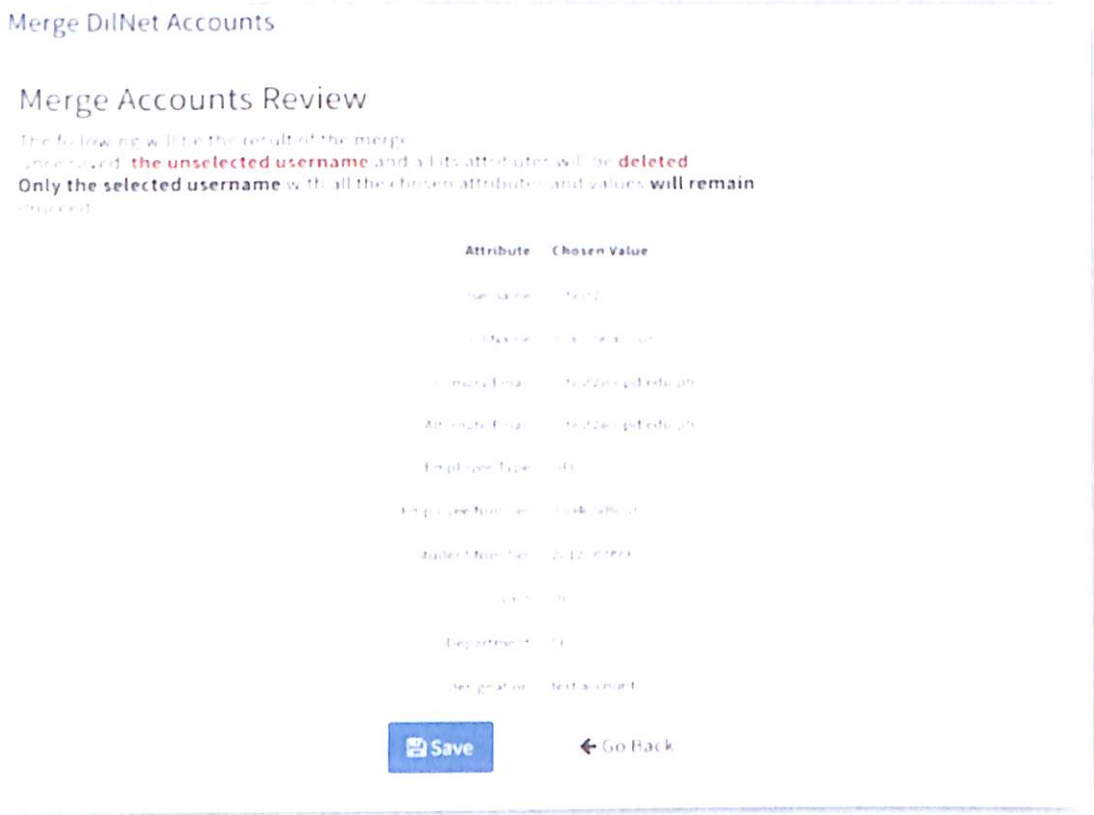


Figure I-18 Merge Accounts Review page

The Admin may then press the Save button and finalize the merging process. Afterwards, the Admin will be redirected to the View Account page of the newly merged accounts.

Mail Tools

Mail Tools is a collection of tools for managing the mail related services of DilNet. It has the following sections/features:

Search Mail Logs

The Search Mail Logs tool allows the Admin to find a specific email or emails in the current mail log file. A search query is checked against each line of the log file containing the subject, the recipient or the sender’s email address, the IP address, or even the domain host. A sample search results:

Search Mail Logs

Q Search Mail Logs

Search results for the keyword "facebook" in the mail logs. The results show 10 entries.

Search Logs

Showing 10 entries

Search

ID	Sender	Recipient	Status	Actions
1	web.facebook.com	mmtagab@updt.edu.ph	Completed	 
2	web.facebook.com	mmartraza@updt.edu.ph	Completed	 
3	web.facebook.com	mmartraza@updt.edu.ph	Completed	 
4	web.facebook.com	mmartraza@updt.edu.ph	Completed	 
5	web.facebook.com	mmartraza@updt.edu.ph	Completed	 
6	web.facebook.com	mmartraza@updt.edu.ph	Completed	 
7	web.facebook.com	mmartraza@updt.edu.ph	Completed	 
8	web.facebook.com	mmartraza@updt.edu.ph	Completed	 
9	web.facebook.com	mmartraza@updt.edu.ph	Completed	 
10	web.facebook.com	mmartraza@updt.edu.ph	Completed	 

Figure I-19 Search Mail Logs results page

The search results show the following fields:

- ID
 - Shows the Exim message ID assigned to the email log entry.
- Sender
 - Shows the sender email.
- Recipient
 - Shows the recipient email.
- Status
 - Shows the status of the email delivery. Since this is a log file, this is usually Completed.
- Actions
 - Available actions for the email. The Admin can:
 - View the email log entry

- Block the Sender

Block/Unblock Email

The Block/Unblock Email allows the Admin to block or unblock specific email addresses on the mail relays. Blocking an email address will prevent it from sending through the mail relays or receiving from them.

The default page for this tool is the search utility to know if an email address is already in the database either as blocked or unblocked entry. A sample screenshot with search results is shown below.

Search Mail DB

Q Search Mail DB

If you know the email address, type it below. Or type part of it to find it. Then press the  to **unblock**, or  to **block** the email address.

Q

Search Email Address

DB Search Results for 'test'

Email	Status	Date Banned	Admin	Actions
test4@upd.edu.ph	BLOCKED	2017-10-22 01:08:20	byron	 
test3@upd.edu.ph	BLOCKED	2017-10-25 12:07:30	byron	 
test7@upd.edu.ph	ALLOWED	2017-10-20 11:30:17	byron	 
test3_first1@upd.edu.ph	ALLOWED	2017-11-02 18:22:34	byron	 
Email	Status	Date Banned	Admin	Actions

Figure I-20 Search Mail DB page

The fields shown are:

- Email

Shows the email address currently blocked or unblocked on the mail relays.

- Status

Status of the email. Either Blocked or Allowed.

- Date Banned

Shows the Date when the email address was blocked from the system.

- Admin

Shows the username of the Admin who performed the blocking/unblocking.

- Actions

Available actions for the email address:

- View Email Entry shows a detailed info on the email address.
- Block Email address for those not yet blocked or currently allowed.
- Unblock Email Address for those that are currently blocked in the system.

View Email Status

When viewing an email entry, a similar page is shown:

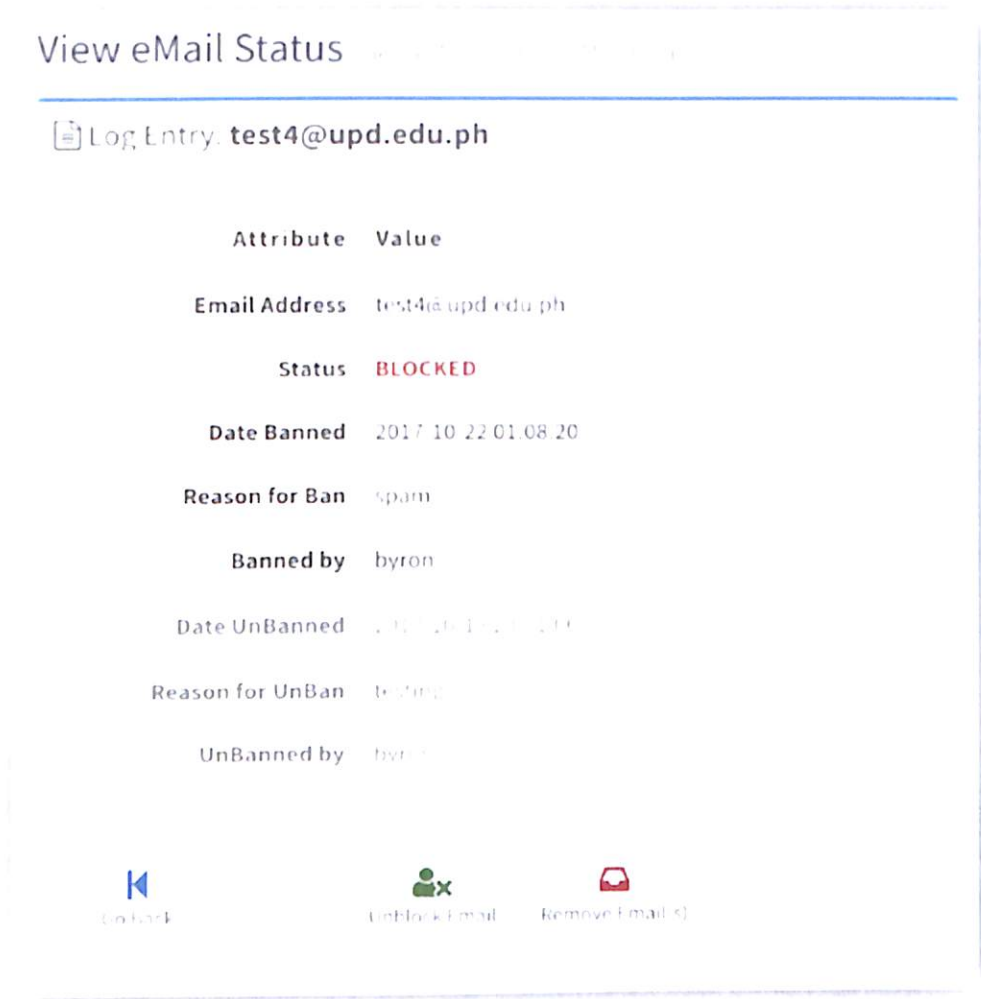


Figure I-21 View Email Status page

It contains the information when the email was blocked, the reason for blocking and the admin responsible for the action.

As can be seen from the screenshot, the Admin can also remove if any the currently queued emails of the address.

Block Email Address

This action is available for addresses currently allowed or is not yet blocked on the system. A sample screenshot is shown below:

Block Email Address

You are going to **block** this email address.

When blocked, mails originating from and destined to this address will be rejected.

REVIEW the information below before you continue.

Block: **test7@upd.edu.ph**

Reason

 **Block Email**

 Go Back

Figure I-22 Block Email Address page

A reason is required before an email address can be blocked. A blocked email address will not receive and will not be able to send emails through the mail relays.

Unblock Email Address

Unblocking an email address will allow it to send and receive emails to and from the mail relays. A reason is required before an email address can be unblocked.

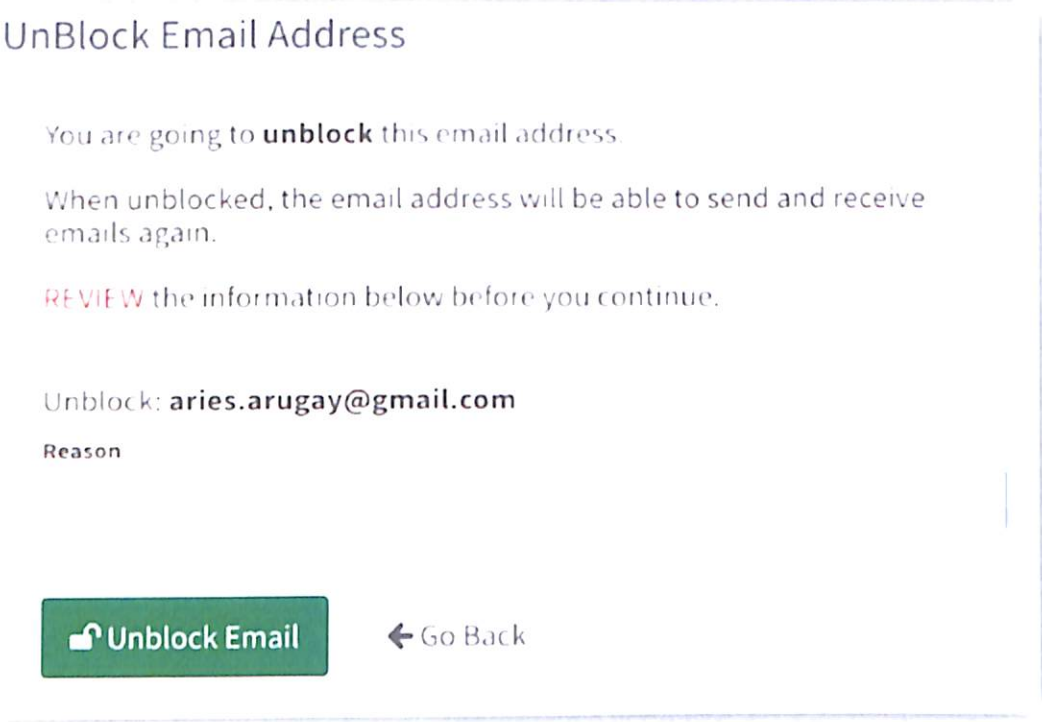


Figure I-23 Unblock Email Address page

Block/Unblock Host

The Block/Unblock Host allows the Admin to block or unblock specific Host IP addresses on the mail relays. Blocking a Host IP address will prevent the host from initiating mail transactions.

Like in the Block/Unblock Email, the Block/Unblock Host default page is the search utility. A screenshot with sample search results is shown below:

Search Blocked Hosts DB

Q Search Blocked Hosts DB

If you know the Host IP address, type it below. Or type part of it to find it. Then press the  to **unblock**, or  to **block** the Host IP address.



Search IP Address

DB Search Results for '10.'

Host	Status	Date Banned	Admin	Actions
10.32.1.7	ALLOWED	2017-10-10 13:48:57	byron	 
10.40.254.5	BLOCKED	2017-10-10 13:55:03	byron	 
Host	Status	Date Banned	Admin	Actions

Figure I-24 Search Blocked Hosts DB page

The fields shown are:

- Host

Shows the IP address of the mail host. Currently, only IPV4 addresses are supported.

- Status

Shows **BLOCKED** if the IP address is currently blocked or **ALLOWED** if it's unblocked.

- Date Banned

The date when the IP address was blocked.

- Admin

The Admin responsible for the blocking of the IP address.

- Actions

The actions available are:

- View Host Entry - opens the View Host Log Entry to view a more detailed info on the IP address.
- Block Host Address - to block the IP address.
- Unblock Host Address - unblock the currently blocked IP Address.

View Host Log Entry

In this page, information on when the IP address was blocked, the reason, and the Admin responsible is shown.

If the IP address is currently blocked, an Unblock IP button is available at the bottom. If the IP address is not blocked, a Block IP is shown below:

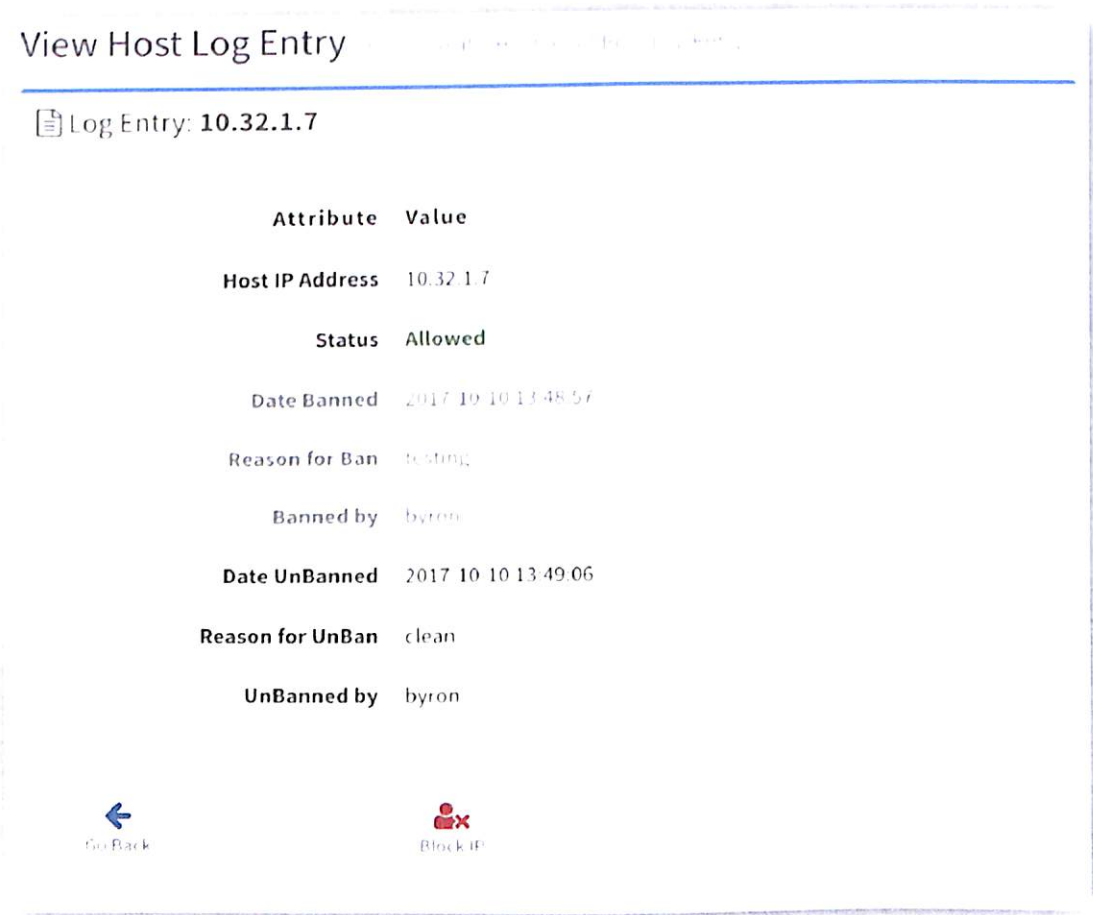


Figure I-25 View Host Log Entry page

Block IP Address

This action is available for IP addresses not yet blocked or is currently allowed on the mail relays. Blocking an IP address requires a reason. A sample screenshot is shown below:

Block IP Address

You are going to **block** this IP address.
When blocked, the IP address will be rejected from initiating any mail transfer operation.
REVIEW the information below before you continue.

Block IP: **10.32.1.7**

Reason

Block IP

Go Back

Figure I-26 Block IP Address page

Unblock IP Address

An IP address currently blocked can be allowed using this action. An unblocked IP address can once again initiate mail transactions on the mail relays. A reason is required to unblock an IP address. A screenshot is shown below:

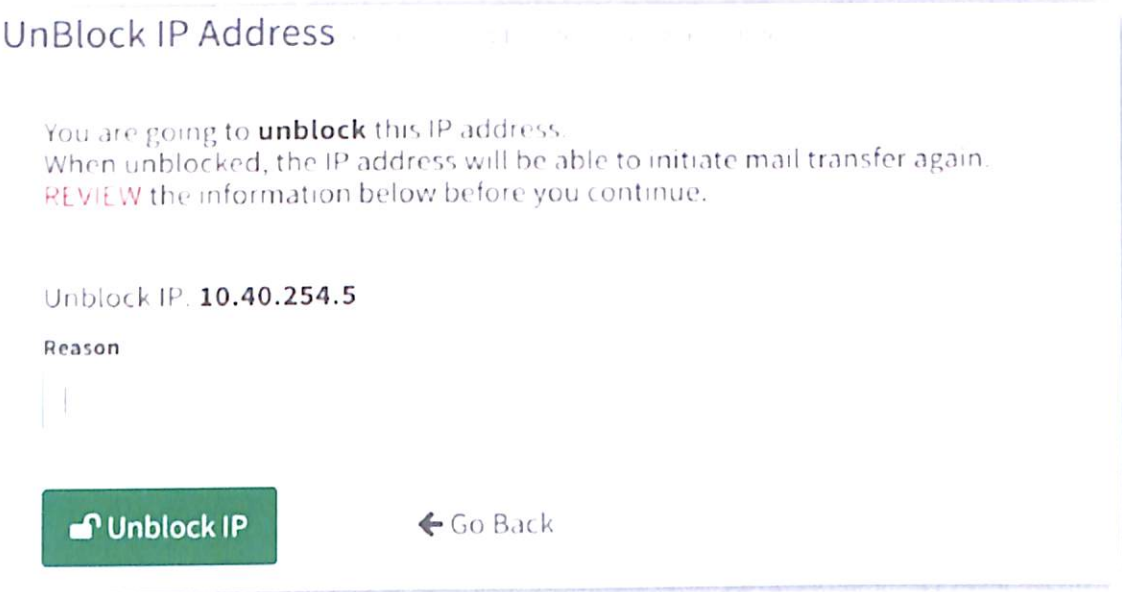


Figure I-27 Unblock IP address page

Proxy Tools

Proxy Tools is a collection of tools for managing the proxy service access of DilNet accounts. It is also where the Radius Profile or proxy activity of DilNet accounts can be seen.

It has the following sections/features:

Radius Profile

This page shows the current bandwidth consumption of all the users connected to the DilNet WiFi proxy service. Specific username profile can be generated using the search form.

Radius Profile Lookup

Get Radius Profile by Username



Search Profile

Total Input Bytes Today in MB	25,159.88
Total Output Bytes Today in MB	3,549.05

Online Users

Showing 10 entries

Search

Username	IP Address	Start Time
aaarroyo2	10.60.36.167	2017-11-06 00:59:30
aadalam	10.60.159.25	2017-11-06 00:09:24
aagonzales7	10.60.98.77	2017-11-06 00:52:57
aailagan	10.61.51.7	2017-11-06 00:36:09

Figure I-28 Radius Profile Lookup page

A sample radius profile of a specific user is shown below:

Radius profile for AAARROYO2

Showing 10 entries

Search

ID	IP Address	Start Time	Stop Time	Terminate Cause	Input (MB)	Output (MB)
218606	10.61.33.134	2016-12-14 10:38:01	2016-12-14 10:38:18	Lost Service	0.14	0.02
2186127	10.61.33.134	2016-12-14 10:40:03	2016-12-14 10:46:20	Lost Service	18.16	1.62
2186175	10.61.33.134	2016-12-14 11:23:29	2016-12-14 18:10:12	Lost Service	84.66	7.46
2186436	10.61.33.134	2016-12-14 16:10:17	2016-12-14 16:26:23	User Request	7.24	0.44
2186459	10.61.33.134	2016-12-14 19:42:10	2016-12-14 19:42:29	User Request	0.14	0.02
2186473	10.61.33.134	2016-12-14 19:42:46	2016-12-14 20:14:18	Lost Service	594.93	37.57
2186701	10.61.33.134	2016-12-14 20:14:18	2016-12-14 20:47:05	Lost Service	2.55	0.44
2194461	10.61.33.134	2016-12-15 09:10:42	2016-12-15 09:39:54	Lost Service	661.11	41.02
2195149	10.61.33.134	2016-12-15 09:39:54	2016-12-15 10:23:33	Lost Service	1.94	0.18
2205834	10.61.33.134	2016-12-15 15:20:56	2016-12-15 15:55:13	Lost Service	784.28	48.16

Showing 1 to 10 of 4,533 entries

1

Figure I-29 Radius Profile for specific user

Block/Unblock Client

In the Block/Unblock Client page, a DilNet account can be blocked or unblocked in the proxy service. A blocked DilNet account will not be able to login on the Radius server which filters the access on the proxy service of the DilNet WiFi. A sample screenshot with search results is shown below:

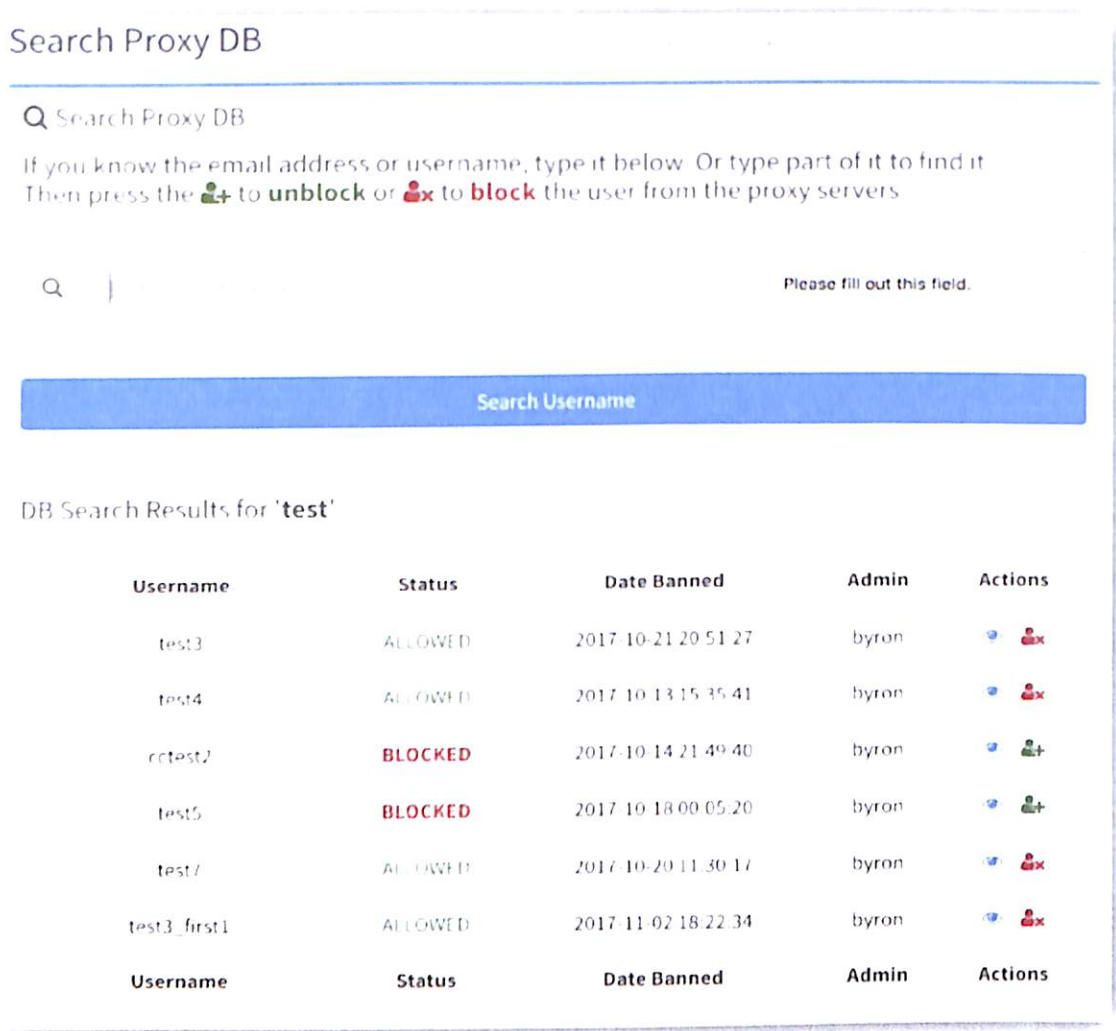


Figure I-30 Search Proxy DB page

The fields shown are:

- Username
Shows the username of the account currently listed in the database, either as blocked or unblocked entry.
- Status

Status of the account in the proxy service. Either Blocked or Allowed.

- **Date Banned**

The latest date the account is blocked on the proxy service.

- **Admin**

The Admin responsible for the blocking or unblocking.

- **Actions**

The actions available are:

- **View User Entry** - allows the admin to use the View User Proxy Status to see a more detailed information on the account.
- **Block User** - block the user from accessing the proxy service.
- **Unblock User** - unblock a currently blocked proxy user.

View User Proxy Status

In this page, information on when the user was blocked, the reason for blocking, and the admin responsible is shown. A sample screenshot is shown below:

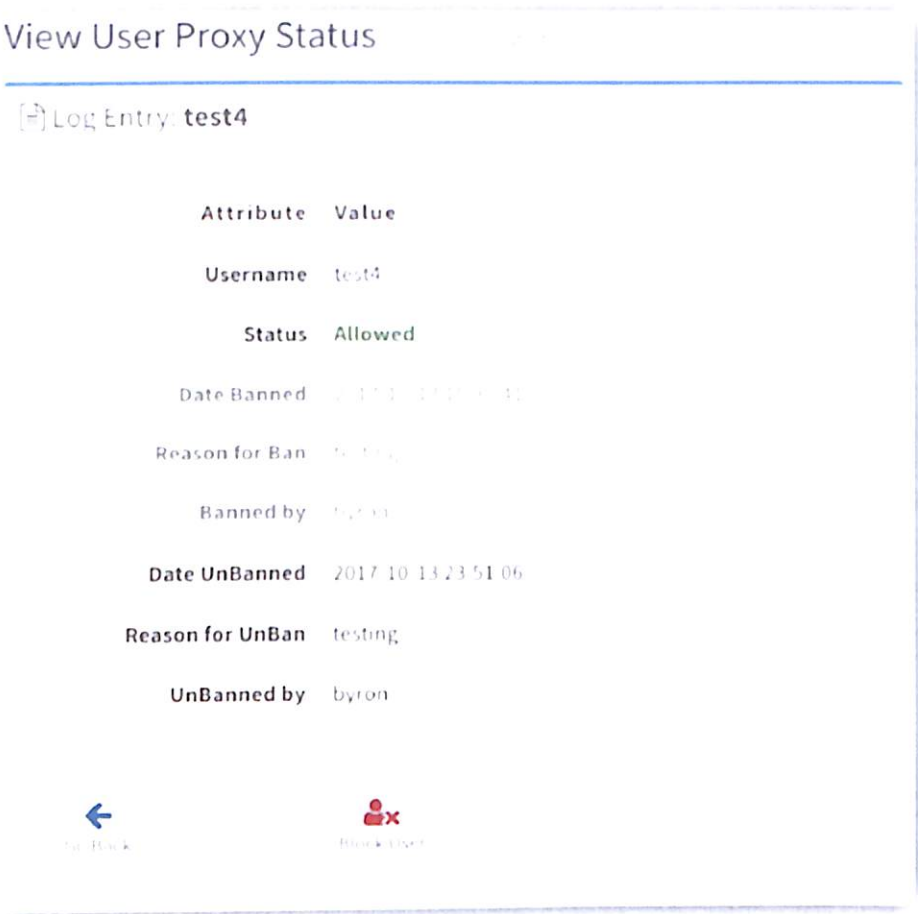


Figure I-31 View User Proxy Status page

Block User Proxy Access

This page allows the admin to block a user from using the proxy service. A blocked user won't be able to login to the Radius server controlling the access to the proxy service. A reason is required before an account can be blocked. A sample screenshot is shown below:

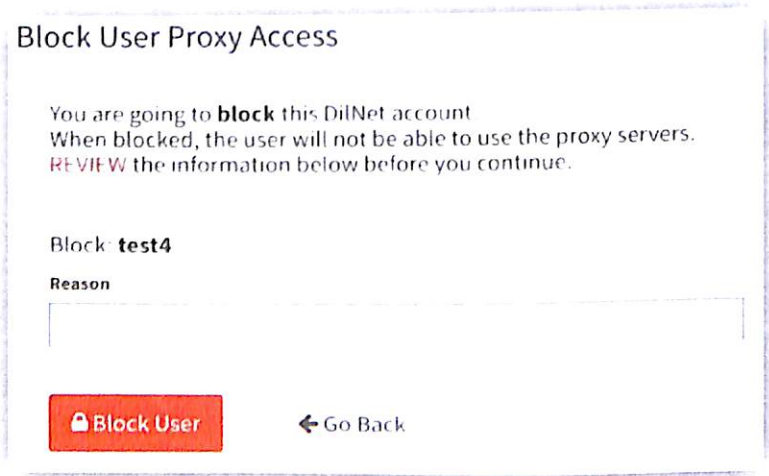


Figure I-32 Block User Proxy Access page

Unblock Proxy User

Blocked users can be allowed again to use the proxy service using this tool. To unblock a user, the admin must provide a reason. A sample screenshot is shown below.

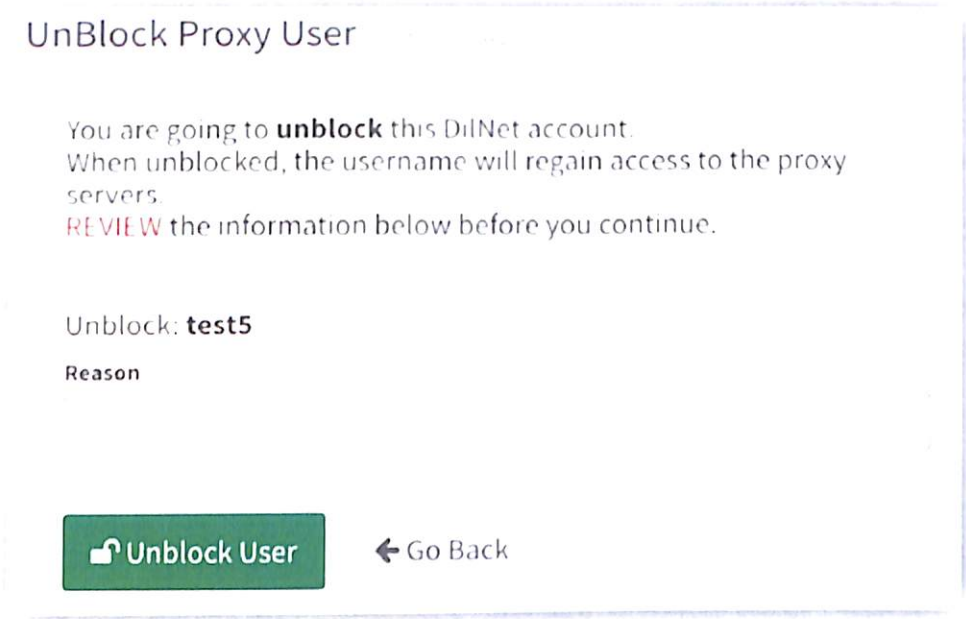


Figure I-33 Unblock Proxy User page

Admin Tools

Admin Tools is where a student account can be activated or if the account is existing, reset the account's password.

It has one section for activating student accounts.

Activate Student Account

To activate a student account, the student number or the form 5 barcode must be entered in the Activate Student Account form. A screenshot is shown below:

Activate Student Account

Scan Form 5 or UP ID

Check

Figure I-34 Activate Student Account input page

The admin will then be presented with the account information. Upon verification of the information, the admin may press the Create DilNet Account button. This page looks like:

Student Account Activation

Create DilNet account

VERIFY Student Information

If info does not match, **DO NOT CREATE**
Student should go to OUR-CRS to correct his/her details.

Name

OCAMPO-BRENZUELA, MYRA EVAN CONSTANTINO

Student Number

2001 54507

Gender

Female

College

National College of Public Administration and Governance

Department

Center for Public Administration and Governance Education

Degree Program

M Public Administration

Cancel

Create DilNet account

Figure I-35 Student Account Activation review page

In the case the user has an account, the reset password page is shown:

Integrated Information Systems for Enhanced Accounts Management 122

Student Account Manager - Reset Password

Reset Password

VERIFY

Student Information

If info does not match, do not reset.
Student should go to OUR CRS to correct his/her details.

DilNet Username	jcbasconcillo
Status	ENROLLED
Name	BASCONCILLO, JENNIFER CANILLAS
Student Number	201790333
Gender	Female
College	College of Social Work and Community Development
Department	Department of Social Work
Degree Program	M Social Work

Cancel

Reset Password

Figure I-36 Student Account Reset Password page

If the student is also an employee and already has an account, the admin is redirected to the View Account page of the user.

Appendix J

TECHNICAL REFERENCE

Hardware

LDAP Server

Processor Information

Socket Designation: LGA771/CPU1

Type: Central Processor

Family: Xeon

Manufacturer: GenuineIntel

ID: 7A 06 01 00 FF FB EB BF

Signature: Type 0, Family 6, Model 23, Stepping 10

Flags:

FPU (Floating-point unit on-chip)

VME (Virtual mode extension)

DE (Debugging extension)

PSE (Page size extension)

TSC (Time stamp counter)

MSR (Model specific registers)

PAE (Physical address extension)

MCE (Machine check exception)

CX8 (CMPXCHG8 instruction supported)

APIC (On-chip APIC hardware supported)

SEP (Fast system call)

MTRR (Memory type range registers)

PGE (Page global enable)

MCA (Machine check architecture)

CMOV (Conditional move instruction supported)

PAT (Page attribute table)

PSE-36 (36-bit page size extension)
CLFSH (CLFLUSH instruction supported)
DS (Debug store)
ACPI (ACPI supported)
MMX (MMX technology supported)
FXSR (FXSAVE and FXSTOR instructions supported)
SSE (Streaming SIMD extensions)
SSE2 (Streaming SIMD extensions 2)
SS (Self-snoop)
HTT (Multi-threading)
TM (Thermal monitor supported)
PBE (Pending break enabled)

Version: Intel(R) Xeon(TM) processor

Voltage: 1.4 V

External Clock: 1333 MHz

Max Speed: 5000 MHz

Current Speed: 2333 MHz

Status: Populated, Enabled

Upgrade: Socket LGA771

L1 Cache Handle: 0x0006

L2 Cache Handle: 0x0007

L3 Cache Handle: Not Provided

Serial Number: Not Specified

Asset Tag: Not Specified

Part Number: Not Specified

Core Count: 4

Core Enabled: 4

Thread Count: 4

Characteristics:

64-bit capable

Physical Memory Information

MemTotal: 32962192 kB
MemFree: 173792 kB
Buffers: 318344 kB
Cached: 31301760 kB
SwapCached: 0 kB
Active: 16000468 kB
Inactive: 15785868 kB
HighTotal: 0 kB
HighFree: 0 kB
LowTotal: 32962192 kB
LowFree: 173792 kB
SwapTotal: 8161012 kB
SwapFree: 8160896 kB
Dirty: 32 kB
Writeback: 0 kB
AnonPages: 166108 kB
Mapped: 27840 kB
Slab: 964540 kB
PageTables: 6632 kB
NFS_Unstable: 0 kB
Bounce: 0 kB
CommitLimit: 24642108 kB

Committed_AS: 859844 kB
VmallocTotal: 34359738367 kB
VmallocUsed: 267916 kB
VmallocChunk: 34359469243 kB
HugePages_Total: 0
HugePages_Free: 0
HugePages_Rsvd: 0
Hugepagesize: 2048 kB

Storage Information

Attached devices:

Host: scsi0 Channel: 00 Id: 00 Lun: 00

Vendor: ATA Model: ST3160813AS Rev: CC2J

Type: Direct-Access ANSI SCSI revision: 05

Host: scsi1 Channel: 00 Id: 00 Lun: 00

Vendor: ATA Model: ST31000528AS Rev: CC38

Type: Direct-Access ANSI SCSI revision: 05

Web and MySQL Server

Processor Information

Handle 0x0401, DMI type 4, 32 bytes

Processor Information

 Socket Designation: CPU 1

 Type: Central Processor

 Family: Other

 Manufacturer: Bochs

ID: D3 06 00 00 FD FB 8B 07

Version: Not Specified

Voltage: Unknown

External Clock: Unknown

Max Speed: 2000 MHz

Current Speed: 2000 MHz

Status: Populated, Enabled

Upgrade: Other

L1 Cache Handle: Not Provided

L2 Cache Handle: Not Provided

L3 Cache Handle: Not Provided

Handle 0x0402, DMI type 4, 32 bytes

Processor Information

Socket Designation: CPU 2

Type: Central Processor

Family: Other

Manufacturer: Bochs

ID: D3 06 00 00 FD FB 8B 07

Version: Not Specified

Voltage: Unknown

External Clock: Unknown

Max Speed: 2000 MHz

Current Speed: 2000 MHz

Status: Populated, Enabled

Upgrade: Other

L1 Cache Handle: Not Provided

L2 Cache Handle: Not Provided

L3 Cache Handle: Not Provided

Physical Memory Information

MemTotal: 1922316 kB

MemFree: 69680 kB

Buffers: 150316 kB

Cached: 1401388 kB

SwapCached: 844 kB

Active: 983148 kB

Inactive: 702360 kB

Active(anon): 69476 kB

Inactive(anon): 77388 kB

Active(file): 913672 kB

Inactive(file): 624972 kB

Unevictable: 0 kB

Mlocked: 0 kB

SwapTotal: 2097148 kB

SwapFree: 2090288 kB

Dirty: 136 kB

Writeback: 0 kB

AnonPages: 133284 kB

Mapped: 31572 kB

Shmem: 13056 kB

Slab: 132284 kB

SReclaimable: 75000 kB

SUnreclaim: 57284 kB

KernelStack: 1152 kB
PageTables: 9688 kB
NFS_Unstable: 0 kB
Bounce: 0 kB
WritebackTmp: 0 kB
CommitLimit: 3058304 kB
Committed_AS: 481932 kB
VmallocTotal: 34359738367 kB
VmallocUsed: 11476 kB
VmallocChunk: 34359706904 kB
HardwareCorrupted: 0 kB
AnonHugePages: 16384 kB
HugePages_Total: 0
HugePages_Free: 0
HugePages_Rsvd: 0
HugePages_Surp: 0
Hugepagesize: 2048 kB
DirectMap4k: 8180 kB
DirectMap2M: 2088960 kB

Storage Information

Filesystem	Size	Used	Avail	Use%	Mounted on
/dev/vda2	20G	2.6G	16G	14%	/
tmpfs	939M	0	939M	0%	/dev/shm
/dev/vda1	229M	68M	150M	32%	/boot

/dev/vda5 77G 66G 6.6G 91% /var

Mail Relay

Processor Information

Handle 0x0004, DMI type 4, 35 bytes

Processor Information

Socket Designation: WMT478/NWD

Type: Central Processor

Family: Pentium 4

Manufacturer: Intel

ID: 41 0F 00 00 FF FB EB BF

Signature: Type 0, Family 15, Model 4, Stepping 1

Flags:

FPU (Floating-point unit on-chip)

VME (Virtual mode extension)

DE (Debugging extension)

PSE (Page size extension)

TSC (Time stamp counter)

MSR (Model specific registers)

PAE (Physical address extension)

MCE (Machine check exception)

CX8 (CMPXCHG8 instruction supported)

APIC (On-chip APIC hardware supported)

SEP (Fast system call)

MTRR (Memory type range registers)

PGE (Page global enable)

MCA (Machine check architecture)

CMOV (Conditional move instruction supported)

PAT (Page attribute table)

PSE-36 (36-bit page size extension)
CLFSH (CLFLUSH instruction supported)
DS (Debug store)
ACPI (ACPI supported)
MMX (MMX technology supported)
FXSR (Fast floating-point save and restore)
SSE (Streaming SIMD extensions)
SSE2 (Streaming SIMD extensions 2)
SS (Self-snoop)
HTT (Hyper-threading technology)
TM (Thermal monitor supported)
PBE (Pending break enabled)

Version: E0

Voltage: 1.8 V

External Clock: 100 MHz

Max Speed: 3400 MHz

Current Speed: 1400 MHz

Status: Populated, Enabled

Upgrade: Slot 1

L1 Cache Handle: 0x000A

L2 Cache Handle: 0x000B

L3 Cache Handle: Not Provided

Serial Number: Not Specified

Asset Tag: Not Specified

Part Number: Not Specified

Physical Memory Information

MemTotal: 2072652 kB

MemFree: 129100 kB
Buffers: 48692 kB
Cached: 605728 kB
SwapCached: 0 kB
Active: 1785504 kB
Inactive: 100848 kB
HighTotal: 1179072 kB
HighFree: 18188 kB
LowTotal: 893580 kB
LowFree: 110912 kB
SwapTotal: 2048276 kB
SwapFree: 2048276 kB
Dirty: 2368 kB
Writeback: 0 kB
Mapped: 1244316 kB
Slab: 47800 kB
CommitLimit: 3084600 kB
Committed_AS: 2028268 kB
PageTables: 2656 kB
VmallocTotal: 114680 kB
VmallocUsed: 23968 kB
VmallocChunk: 90116 kB

Storage Information

Filesystem	Size	Used	Avail	Use%	Mounted on
------------	------	------	-------	------	------------

/dev/sda1	7.3G	5.2G	1.8G	76%	/
tmpfs	1013M	0	1013M	0%	/dev/shm
/dev/sda3	59G	39G	18G	70%	/var
/dev/hdb1	145G	25G	113G	19%	/mnt/mailc_ext

Radius Server

Processor Information

Handle 0x0401, DMI type 4, 32 bytes

Processor Information

- Socket Designation: CPU 1
- Type: Central Processor
- Family: Other
- Manufacturer: Bochs
- ID: A1 06 02 00 FD FB 8B 07
- Version: Not Specified
- Voltage: Unknown
- External Clock: Unknown
- Max Speed: 2000 MHz
- Current Speed: 2000 MHz
- Status: Populated, Enabled
- Upgrade: Other
- L1 Cache Handle: Not Provided
- L2 Cache Handle: Not Provided
- L3 Cache Handle: Not Provided

Handle 0x0402, DMI type 4, 32 bytes

Processor Information

- Socket Designation: CPU 2

Type: Central Processor
Family: Other
Manufacturer: Bochs
ID: A1 06 02 00 FD FB 8B 07
Version: Not Specified
Voltage: Unknown
External Clock: Unknown
Max Speed: 2000 MHz
Current Speed: 2000 MHz
Status: Populated, Enabled
Upgrade: Other
L1 Cache Handle: Not Provided
L2 Cache Handle: Not Provided
L3 Cache Handle: Not Provided

Handle 0x0403, DMI type 4, 32 bytes

Processor Information

Socket Designation: CPU 3
Type: Central Processor
Family: Other
Manufacturer: Bochs
ID: A1 06 02 00 FD FB 8B 07
Version: Not Specified
Voltage: Unknown
External Clock: Unknown
Max Speed: 2000 MHz
Current Speed: 2000 MHz
Status: Populated, Enabled
Upgrade: Other

L1 Cache Handle: Not Provided
L2 Cache Handle: Not Provided
L3 Cache Handle: Not Provided

Handle 0x0404, DMI type 4, 32 bytes

Processor Information

Socket Designation: CPU 4
Type: Central Processor
Family: Other
Manufacturer: Bochs
ID: A1 06 02 00 FD FB 8B 07
Version: Not Specified
Voltage: Unknown
External Clock: Unknown
Max Speed: 2000 MHz
Current Speed: 2000 MHz
Status: Populated, Enabled
Upgrade: Other
L1 Cache Handle: Not Provided
L2 Cache Handle: Not Provided
L3 Cache Handle: Not Provided

Physical Memory Information

MemTotal: 1883860 kB
MemFree: 70984 kB
MemAvailable: 1361384 kB
Buffers: 0 kB
Cached: 1468108 kB

SwapCached: 2608 kB
Active: 835384 kB
Inactive: 845624 kB
Active(anon): 153296 kB
Inactive(anon): 157972 kB
Active(file): 682088 kB
Inactive(file): 687652 kB
Unevictable: 0 kB
Mlocked: 0 kB
SwapTotal: 2097148 kB
SwapFree: 2001468 kB
Dirty: 60 kB
Writeback: 0 kB
AnonPages: 211376 kB
Mapped: 23148 kB
Shmem: 98368 kB
Slab: 97808 kB
SReclaimable: 66564 kB
SUnreclaim: 31244 kB
KernelStack: 2672 kB
PageTables: 4856 kB
NFS_Unstable: 0 kB
Bounce: 0 kB
WritebackTmp: 0 kB
CommitLimit: 3039076 kB

Committed_AS: 843956 kB
VmallocTotal: 34359738367 kB
VmallocUsed: 13136 kB
VmallocChunk: 34359722748 kB
HardwareCorrupted: 0 kB
AnonHugePages: 182272 kB
HugePages_Total: 0
HugePages_Free: 0
HugePages_Rsvd: 0
HugePages_Surp: 0
Hugepagesize: 2048 kB
DirectMap4k: 71668 kB
DirectMap2M: 2025472 kB

Storage Information

Filesystem	Size	Used	Avail	Use%	Mounted on
/dev/vda3	118G	106G	12G	90%	/
devtmpfs	911M	0	911M	0%	/dev
tmpfs	920M	0	920M	0%	/dev/shm
tmpfs	920M	97M	824M	11%	/run
tmpfs	920M	0	920M	0%	/sys/fs/cgroup
/dev/vda1	509M	117M	393M	23%	/boot
tmpfs	184M	0	184M	0%	/run/user/0

Software

LDAP Server

- CentOS release 5.11 (Final)
 - Linux 2.6.18-404.el5 #1 SMP Tue Apr 7 12:42:54 EDT 2015 x86_64
x86_64 x86_64 GNU/Linux
- OpenLDAP: slapd 2.3.43 (Sep 29 2015 06:22:05)
 - mockbuild@builder17.centos.org:/builddir/build/BUILD/openldap-2.3.43/openldap-2.3.43/build-servers/servers/slapd

Web and MySQL Server

- CentOS release 6.6 (Final)
 - Linux 2.6.32-504.8.1.el6.x86_64 #1 SMP Wed Jan 28 21:11:36 UTC 2015 x86_64 x86_64 x86_64 GNU/Linux
- Server version: Apache/2.2.15 (Unix)
 - Server built: Aug 24 2015 17:52:49
 - Server's Module Magic Number: 20051115:25
 - Server loaded: APR 1.3.9, APR-Util 1.3.9
 - Compiled using: APR 1.3.9, APR-Util 1.3.9
 - Architecture: 64-bit
 - Server MPM: Prefork
 - threaded: no
 - forked: yes (variable process count)
 - Server compiled with....

- -D APACHE_MPM_DIR="server/mpm/prefork"
- -D APR_HAS_SENDFILE
- -D APR_HAS_MMAP
- -D APR_HAVE_IPV6 (IPv4-mapped addresses enabled)
- -D APR_USE_SYSVSEM_SERIALIZE
- -D APR_USE_PTHREAD_SERIALIZE
- -D SINGLE_LISTEN_UNSERIALIZED_ACCEPT
- -D APR_HAS_OTHER_CHILD
- -D AP_HAVE_RELIABLE_PIPED_LOGS
- -D DYNAMIC_MODULE_LIMIT=128
- -D HTTPD_ROOT="/etc/httpd"
- -D SUEXEC_BIN="/usr/sbin/suexec"
- -D DEFAULT_PIDLOG="run/httpd.pid"
- -D DEFAULT_SCOREBOARD="logs/apache_runtime_status"
- -D DEFAULT_LOCKFILE="logs/accept.lock"
- -D DEFAULT_ERRORLOG="logs/error_log"
- -D AP_TYPES_CONFIG_FILE="conf/mime.types"
- -D SERVER_CONFIG_FILE="conf/httpd.conf"
- mysql Ver 14.14 Distrib 5.1.73, for redhat-linux-gnu (x86_64) using readline 5.1
- PHP 5.3.3 (cli) (built: Jul 9 2015 17:39:00)
 - Zend Engine v2.3.0, Copyright (c) 1998-2010 Zend Technologies
 - with XCache v3.0.4, Copyright (c) 2005-2013, by mOo
 - with XCache Optimizer v3.0.4, Copyright (c) 2005-2013, by mOo
 - with XCache Cacher v3.0.4, Copyright (c) 2005-2013, by mOo

- with XCache Coverager v3.0.4, Copyright (c) 2005-2013, by mOo
- OpenSSH_5.3p1, OpenSSL 1.0.1e-fips 11 Feb 2013
- Mike42 ESC/POS Print Driver for PHP

Mail Relay

- Debian 3.1
 - Linux 2.6.17.9-mc #1 SMP Tue Aug 22 14:28:31 CEST 2006 i686 GNU/Linux
- Exim version 4.66 #1 built 09-Jan-2007 18:03:23
- GNU bash, version 2.05b.0(1)-release (i386-pc-linux-gnu)
- OpenSSH_3.8.1p1 Debian-8.sarge.6, OpenSSL 0.9.7e 25 Oct 2004

Radius Server

- CentOS Linux release 7.2.1511 (Core)
 - Linux 3.10.0-327.el7.x86_64 #1 SMP Thu Nov 19 22:10:57 UTC 2015 x86_64 x86_64 x86_64 GNU/Linux
- mysql Ver 14.14 Distrib 5.6.31, for Linux (x86_64) using EditLine wrapper
- radiusd: FreeRADIUS Version 3.0.4, for host x86_64-redhat-linux-gnu, built on Mar 5 2015 at 23:41:36

Programming Language

These are the softwares required for running the new Tech.Accounts website along with the softwares mentioned on each servers above.

Language: PHP

Framework: Codeigniter 3.0.4

Additional Softwares/Frameworks/Kit Used:

- AdminLTE 2.3.0
- Bootstrap 3.3.5
- Chrome 61.0.3163.100 (64-bit)
- Font Awesome 4.5.0
- Git 2.7.1
- Ionicons 2.0.1
- jQuery 2.1.4
- Sparks 0.0.9
- Sublime Text 3.0 (3143)

Maintenance Plan

The following figures list all the used files and their locations relative to the base URL of the website. These do not include the frameworks or kits that must be installed separately.

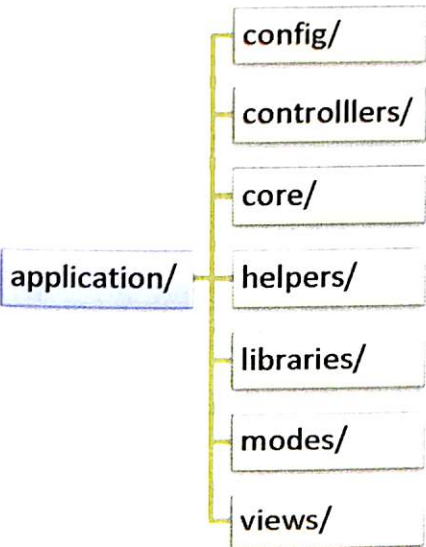


Figure J-1 application folder contents

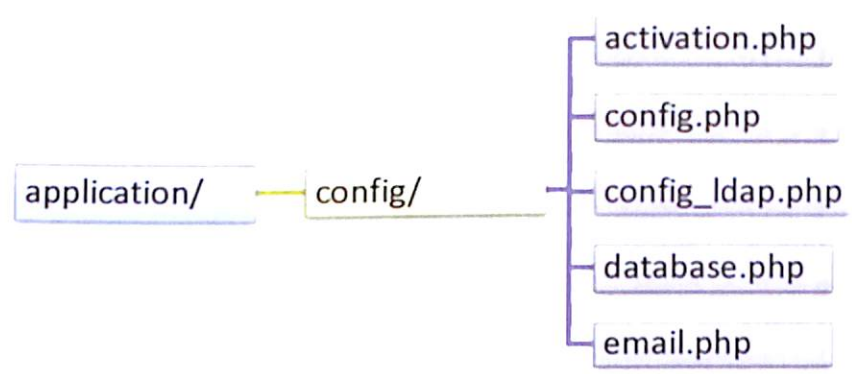


Figure J-2 config folder contents

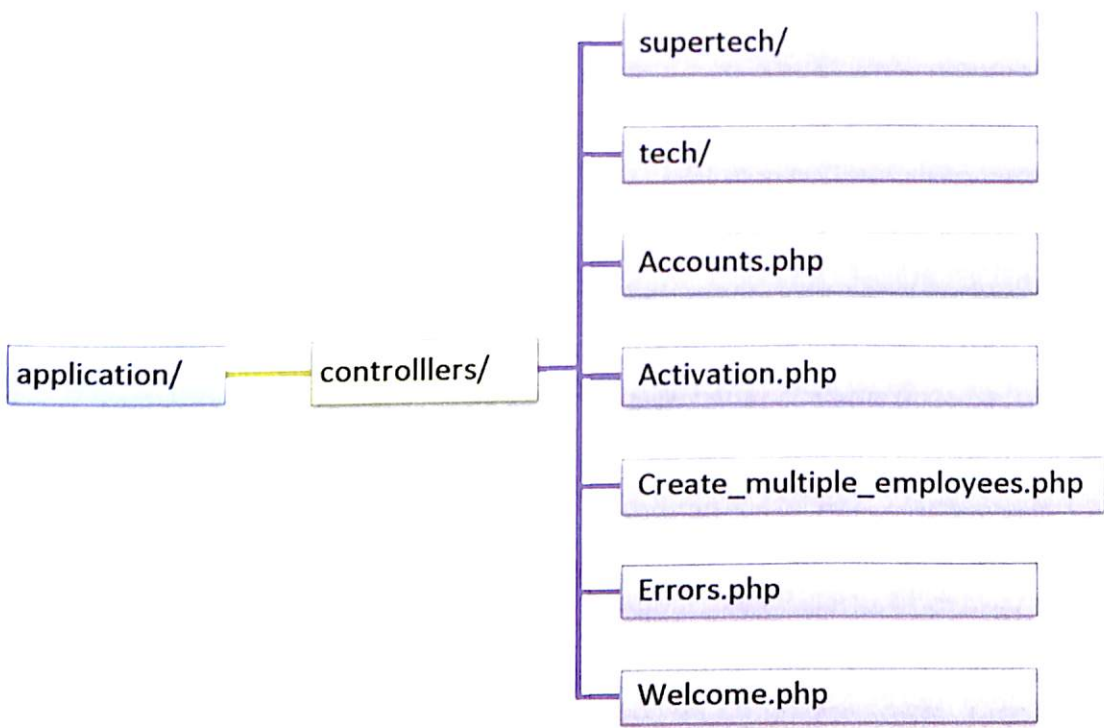


Figure J-3 controllers folder contents

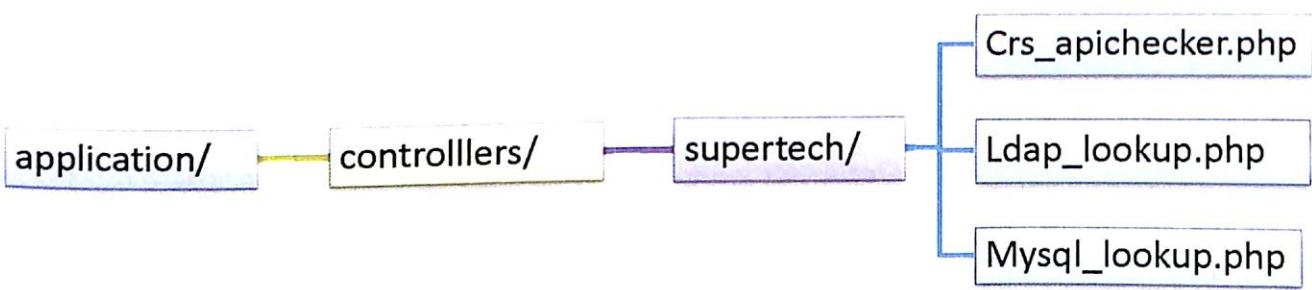


Figure J-4 supertech folder contents

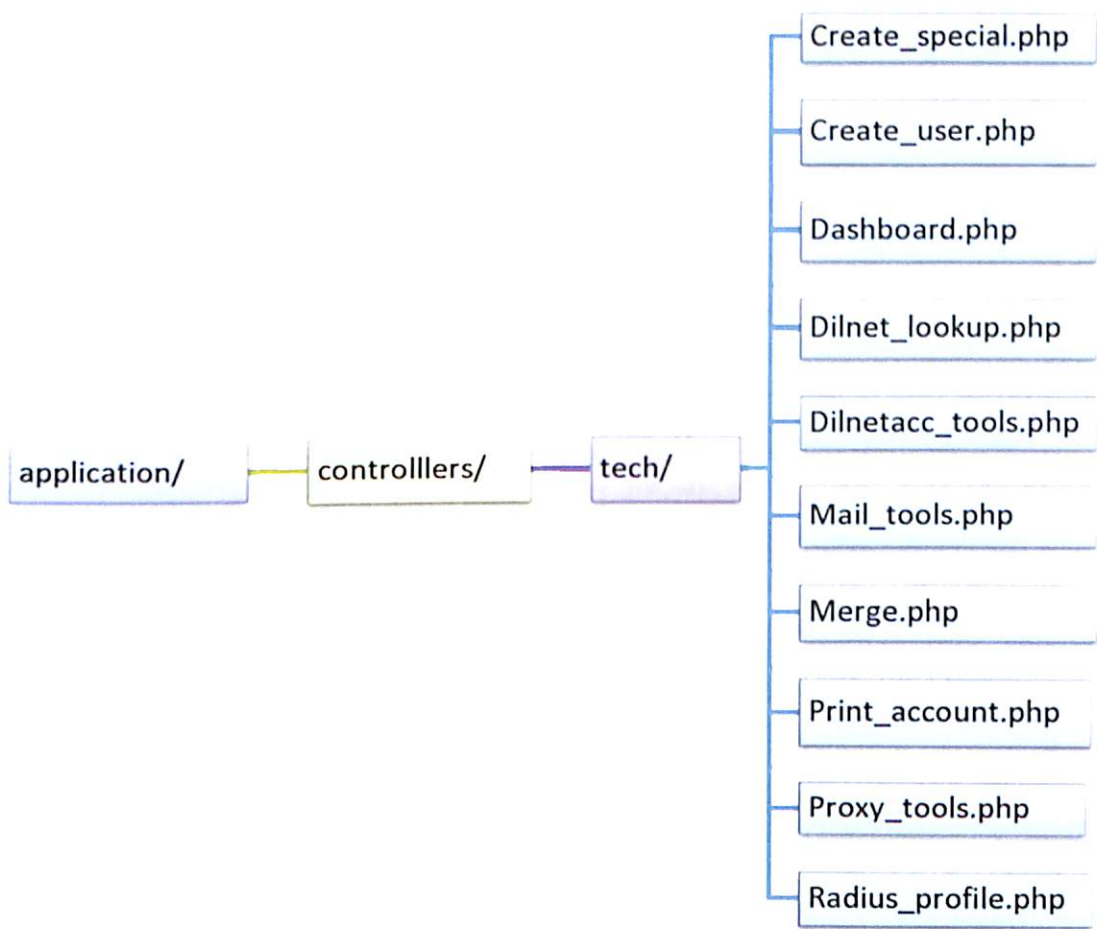


Figure J-5 tech folder contents

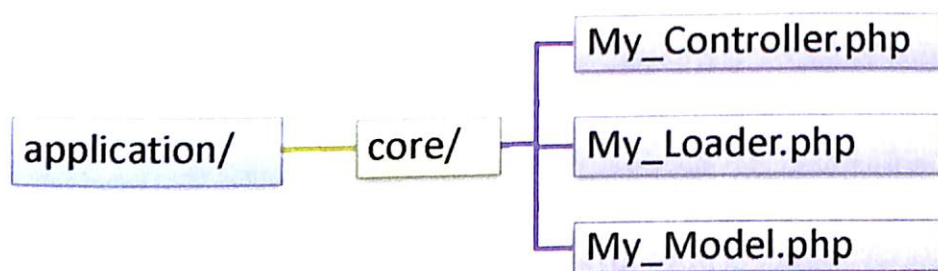


Figure J-6 core folder contents

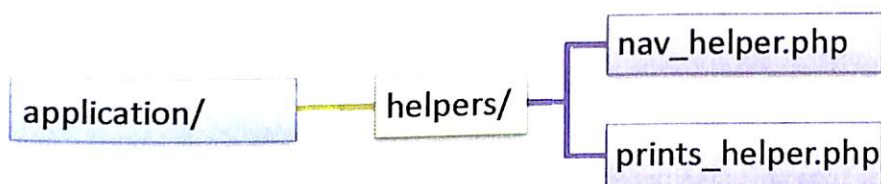


Figure J-7 helpers folder contents

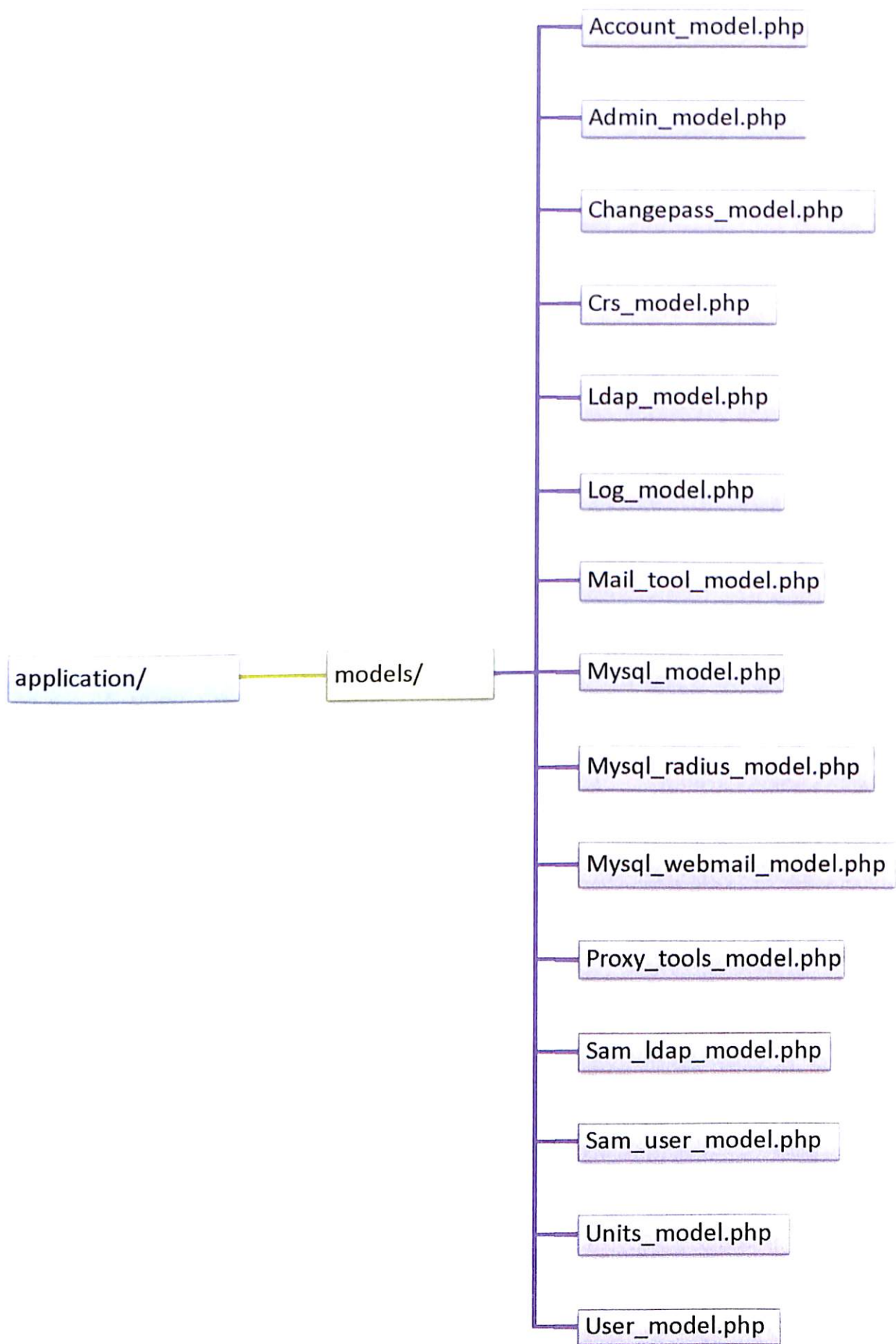


Figure J-8 models folder contents

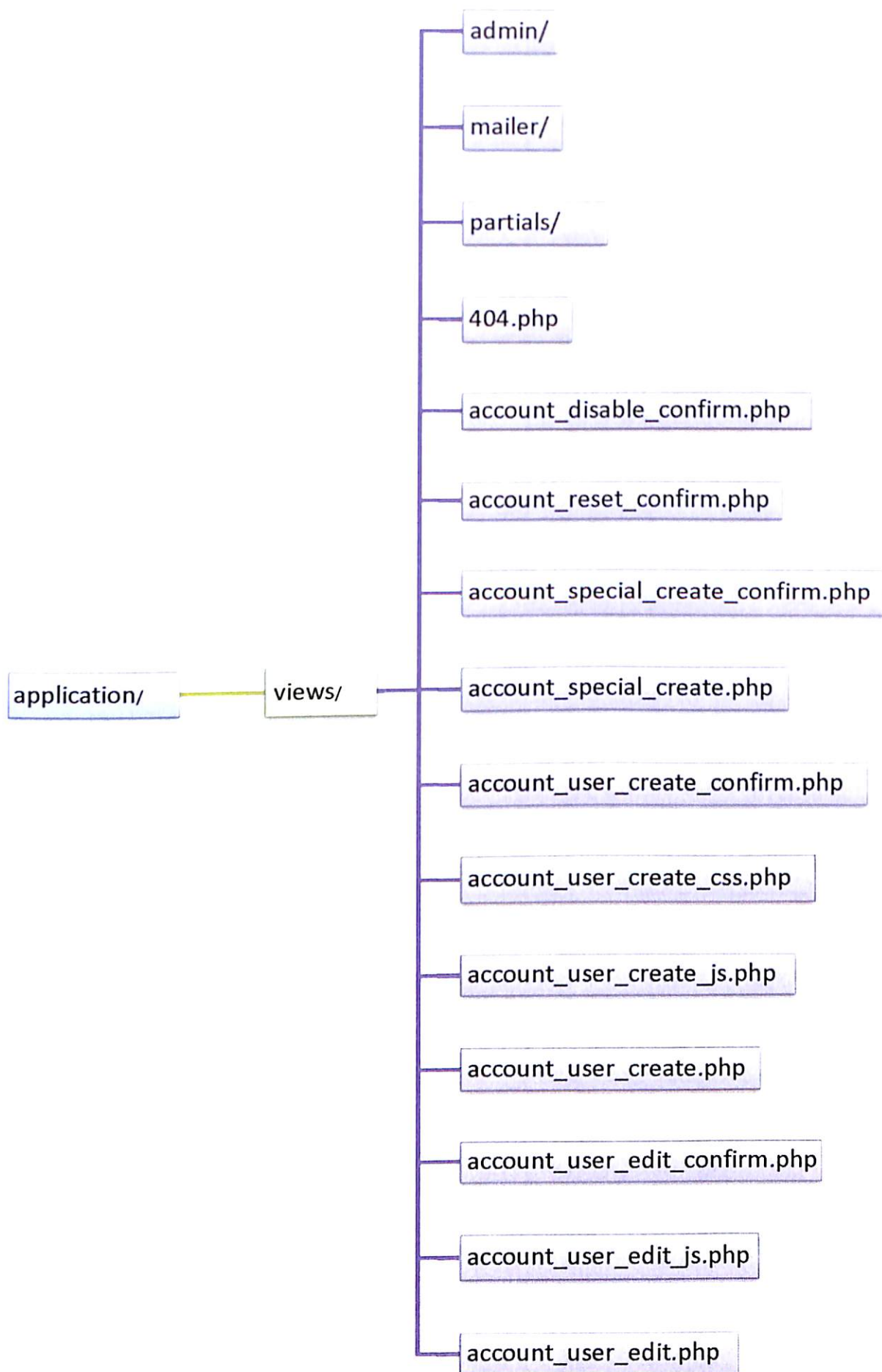


Figure J-9 view folder contents (1 of 4)

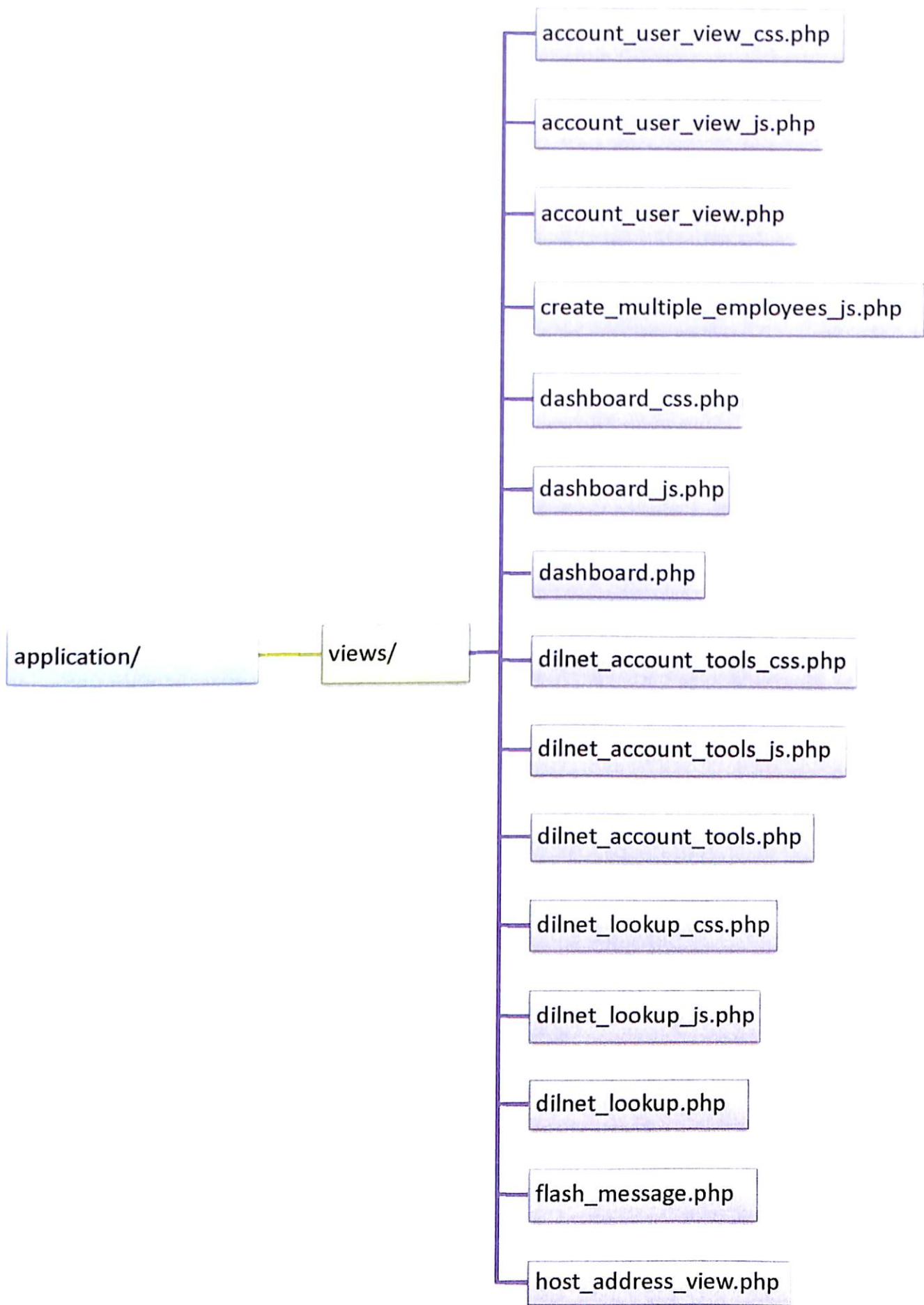


Figure J-10 views folder contents (2 of 4)

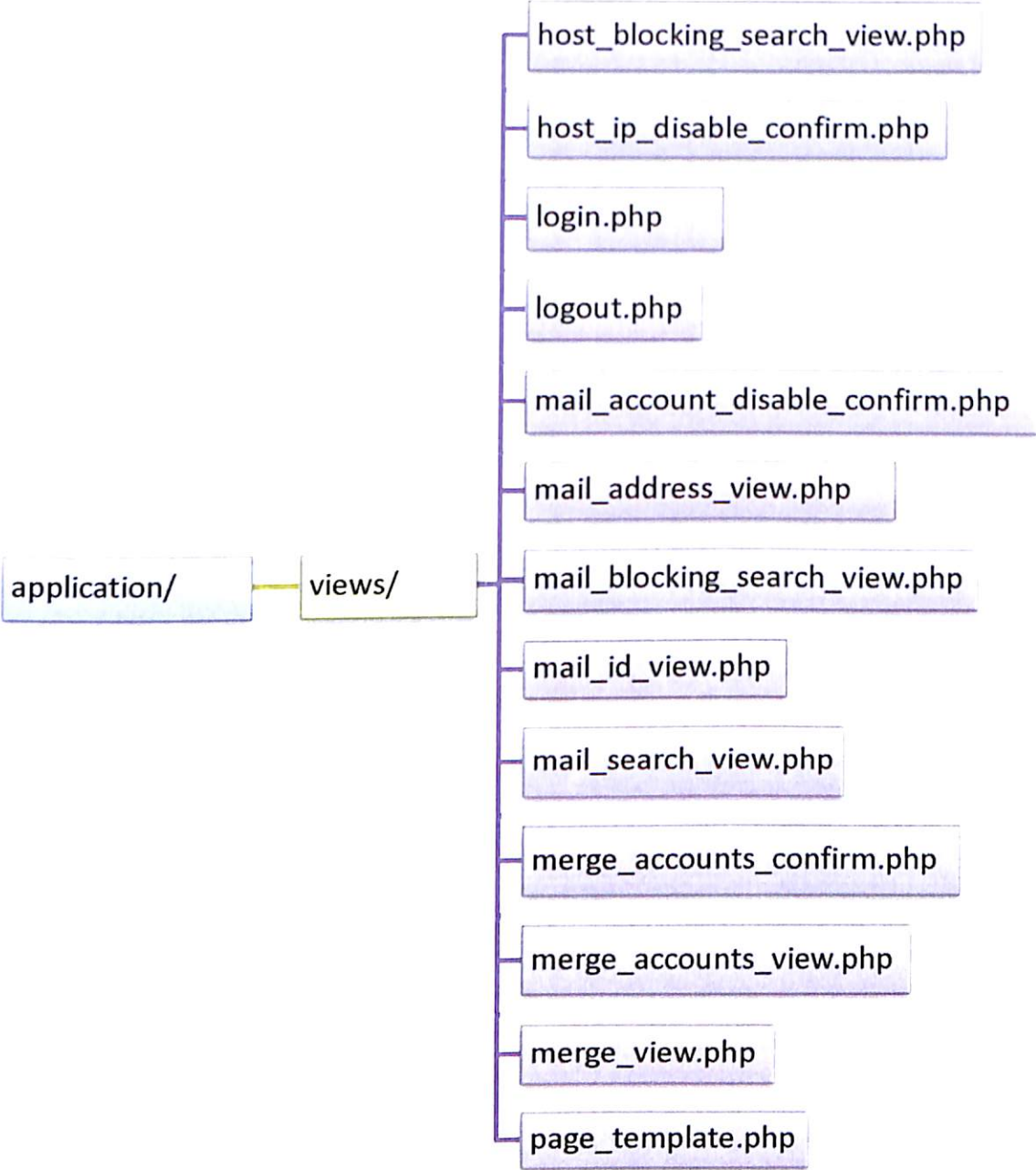


Figure J-11 views folder contents (3 of 4)

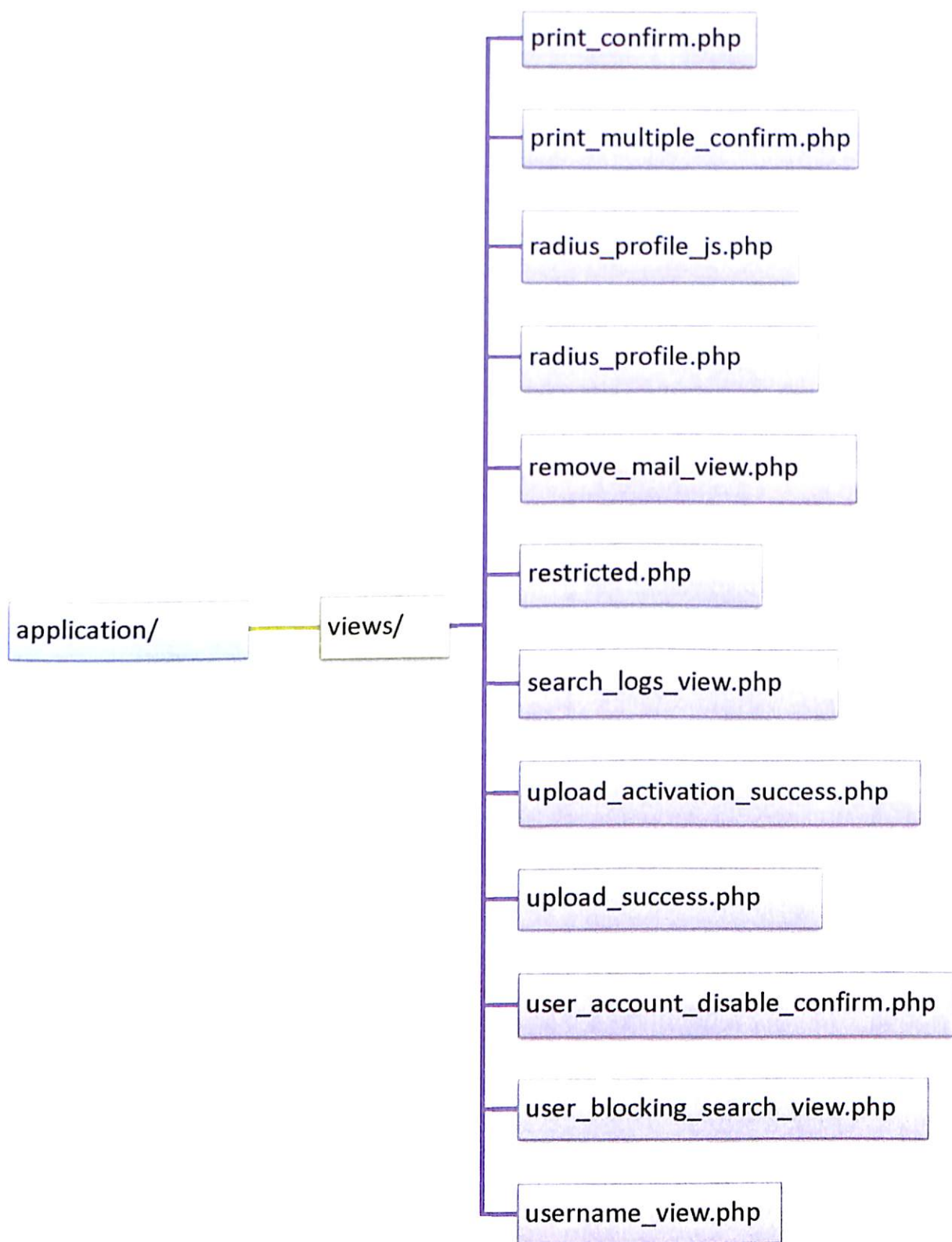


Figure J-12 views folder contents (4 of 4)

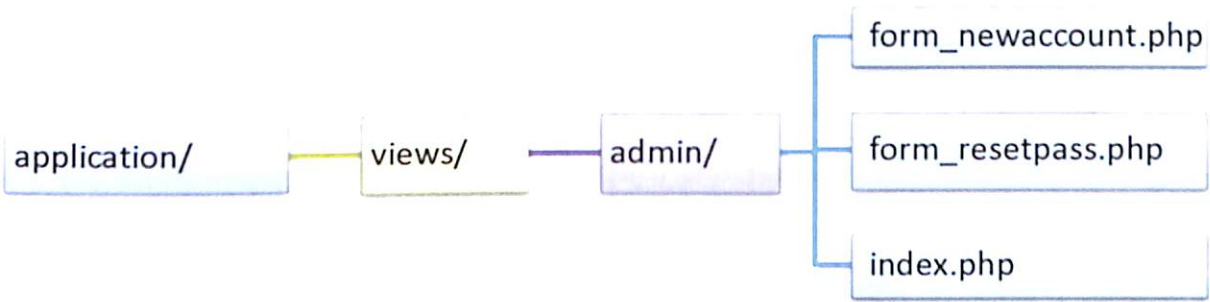


Figure J-13 admin folder contents

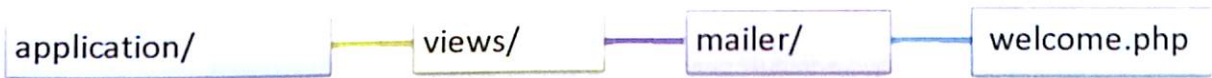


Figure J-14 mailer folder contents

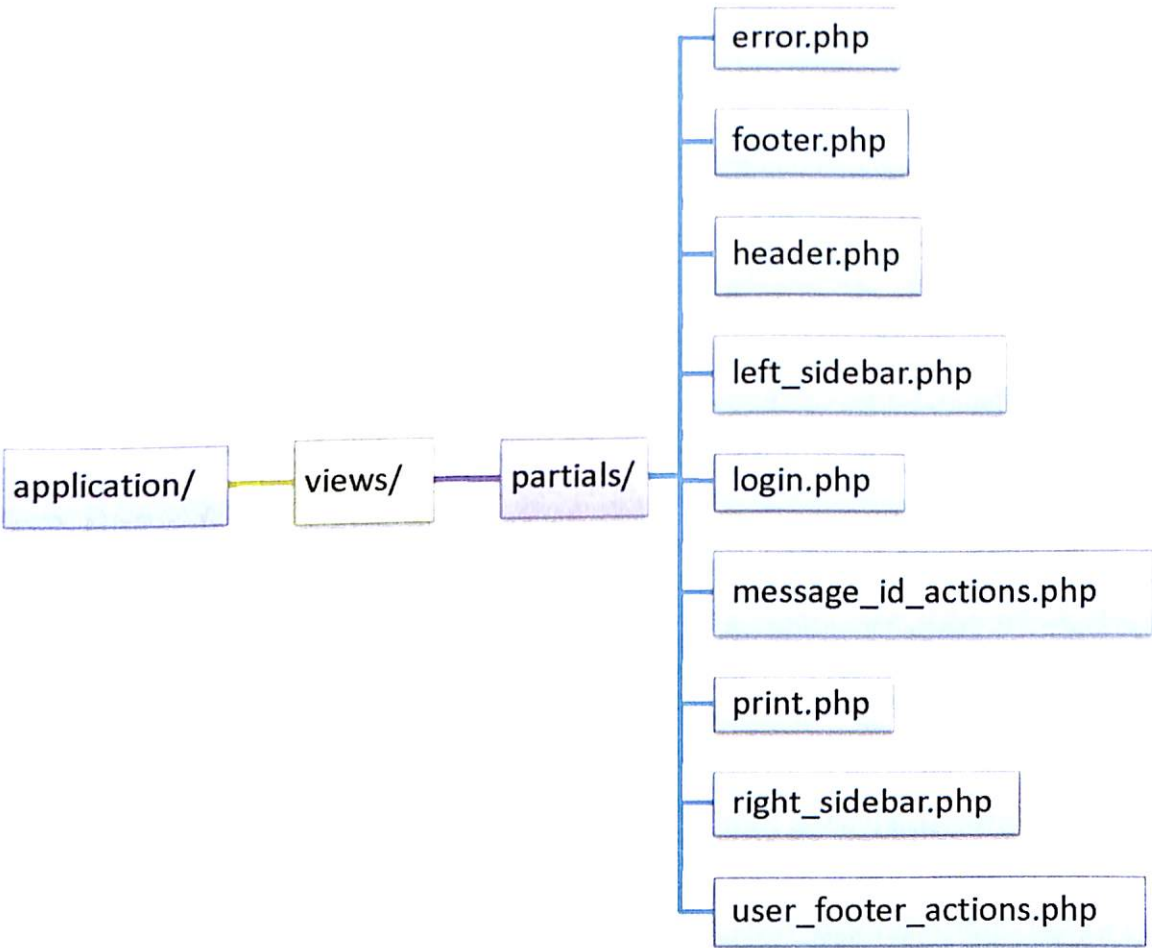


Figure J-15 partials folder contents

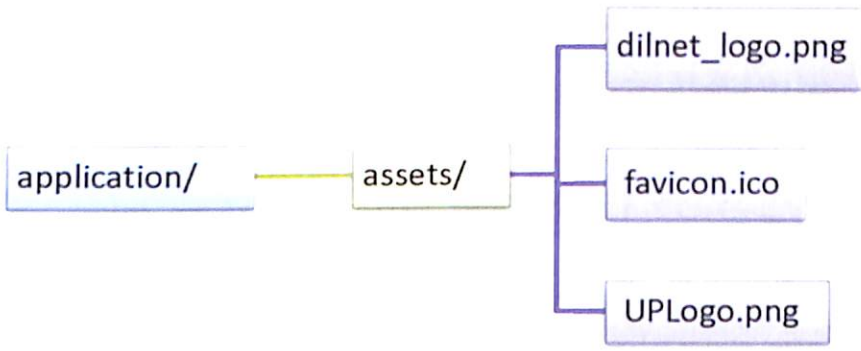
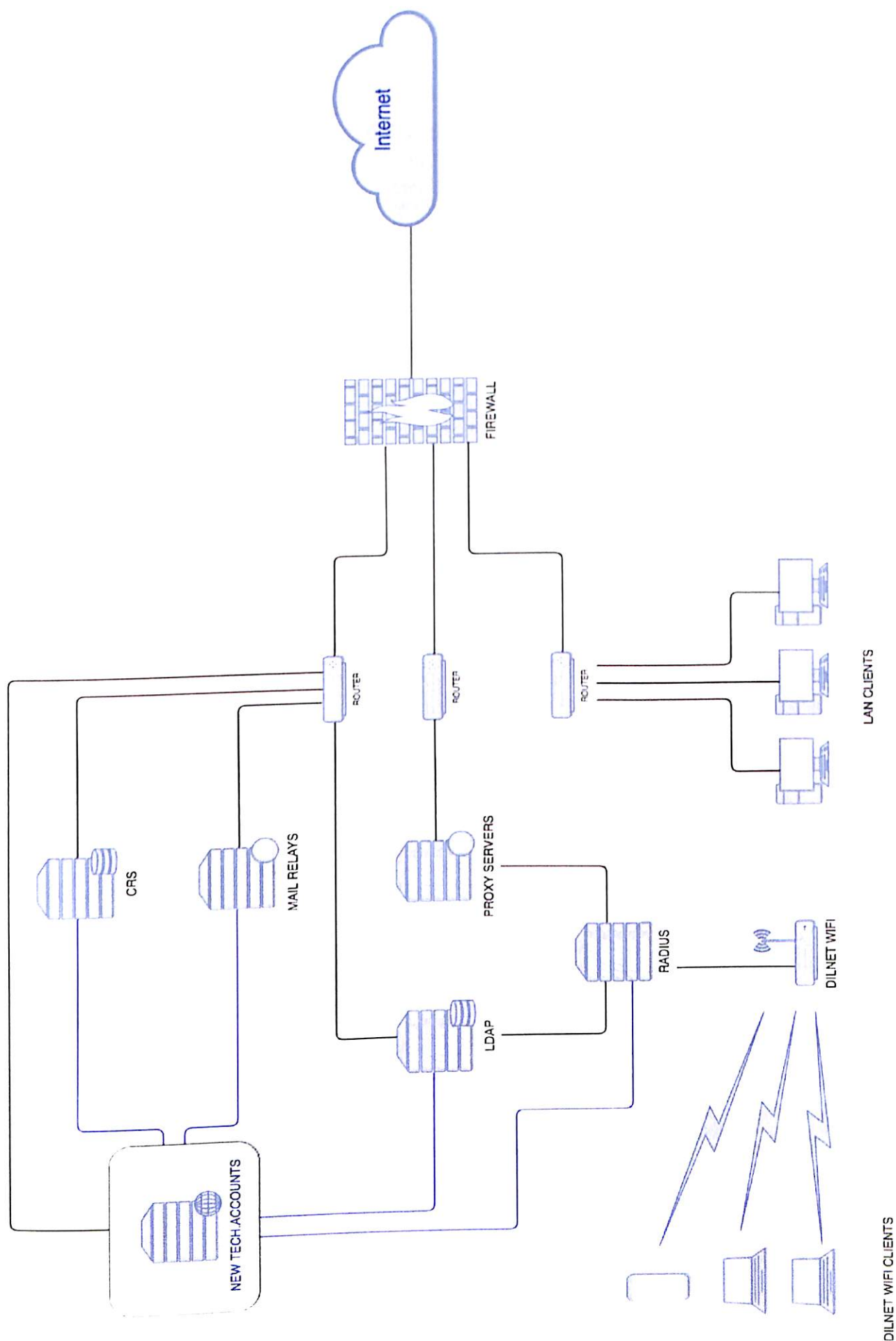


Figure J-16 assets folder contents

Appendix K

NETWORK ARCHITECTURE



Appendix L

SURVEY QUESTIONNAIRE RESULTS

Perceived Usefulness

1. Using the new Tech.Accounts in my job would enable me to accomplish tasks more quickly.

2 - Responses

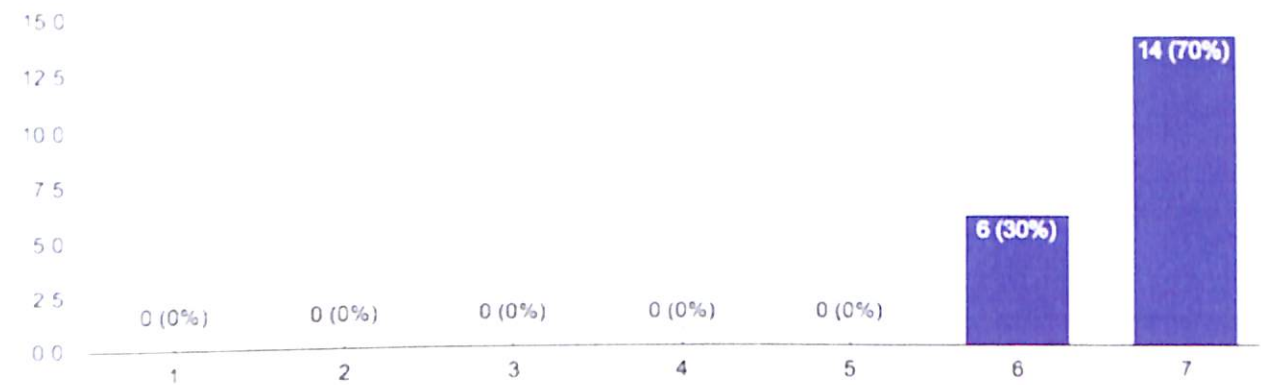


Figure L-1 Perceived Usefulness question #1 results

2. Using the new Tech.Accounts would improve my job performance.

2 - Responses

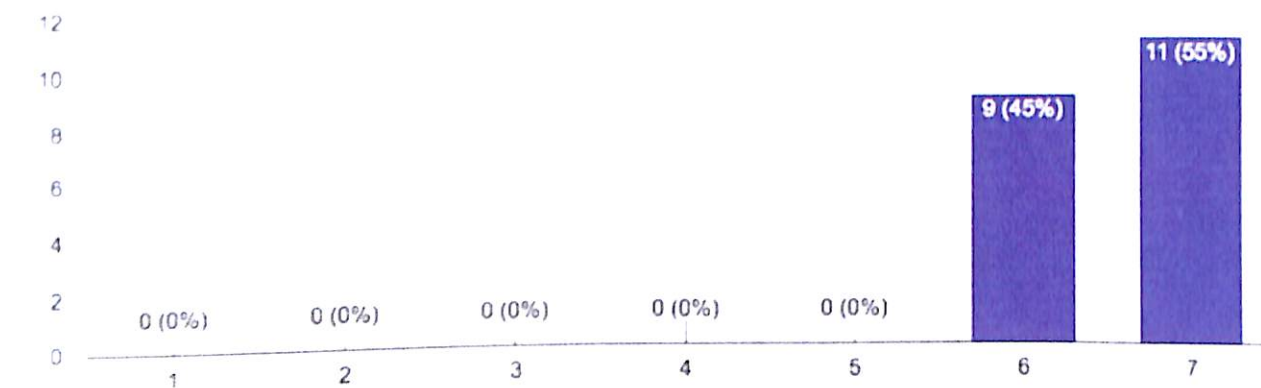


Figure L-2 Perceived Usefulness question #2 results

Appendix L (Continued)

SURVEY QUESTIONNAIRE RESULTS

3. Using the new Tech.Accounts in my job would increase my productivity.

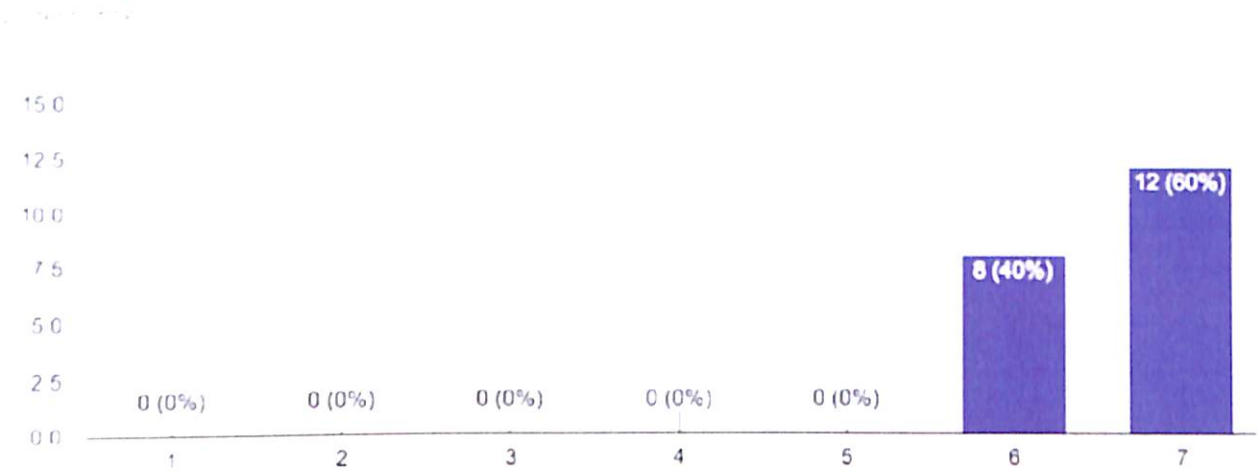


Figure L-3 Perceived Usefulness question #3 results

4. Using the new Tech.Accounts would enhance my effectiveness on the job.

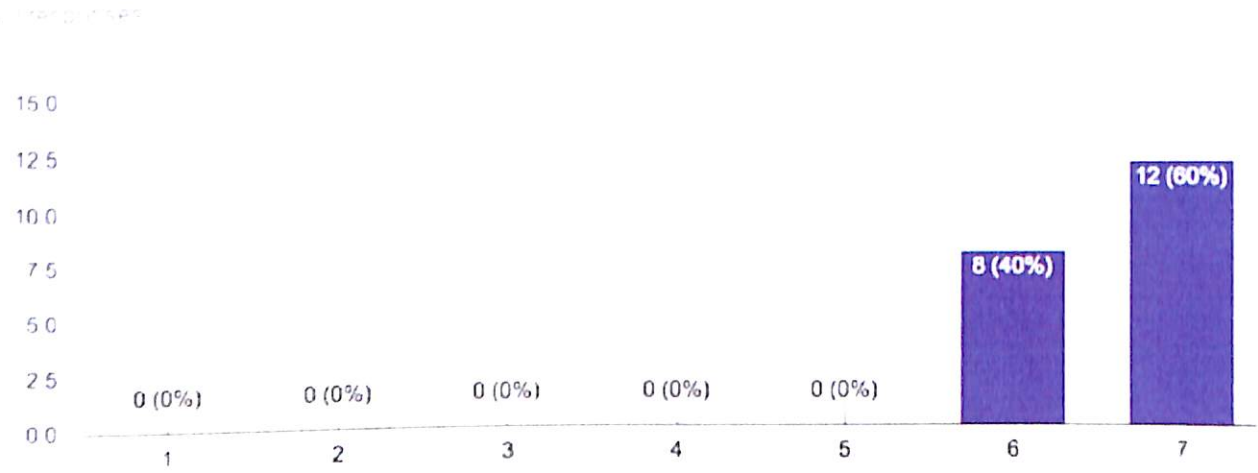


Figure L-4 Perceived Usefulness question #4 results

Appendix L (Continued)

SURVEY QUESTIONNAIRE RESULTS

5. Using the new Tech.Accounts would make it easier to do my job.

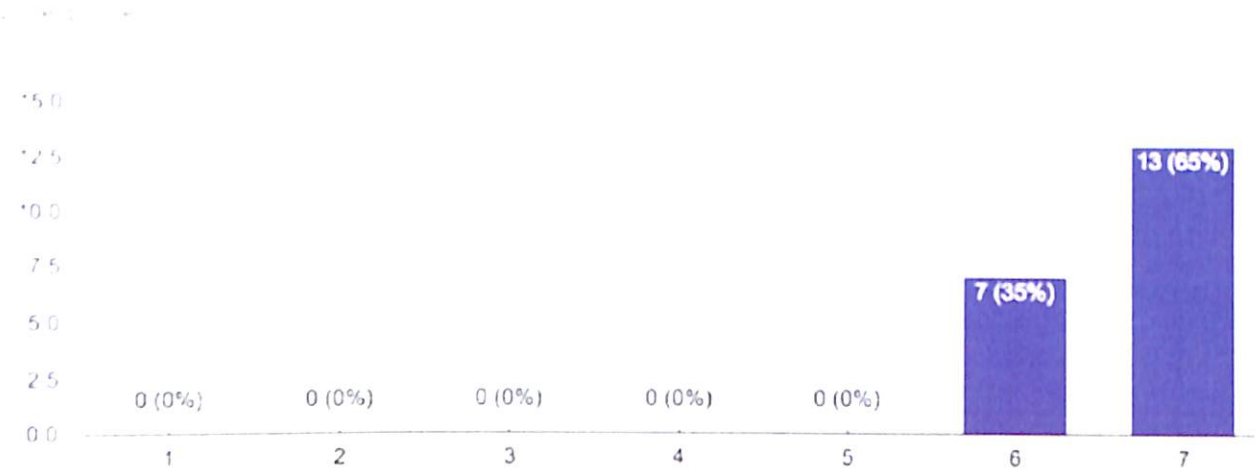


Figure L-5 Perceived Usefulness question #5 results

6. I would find the new Tech.Accounts useful in my job.

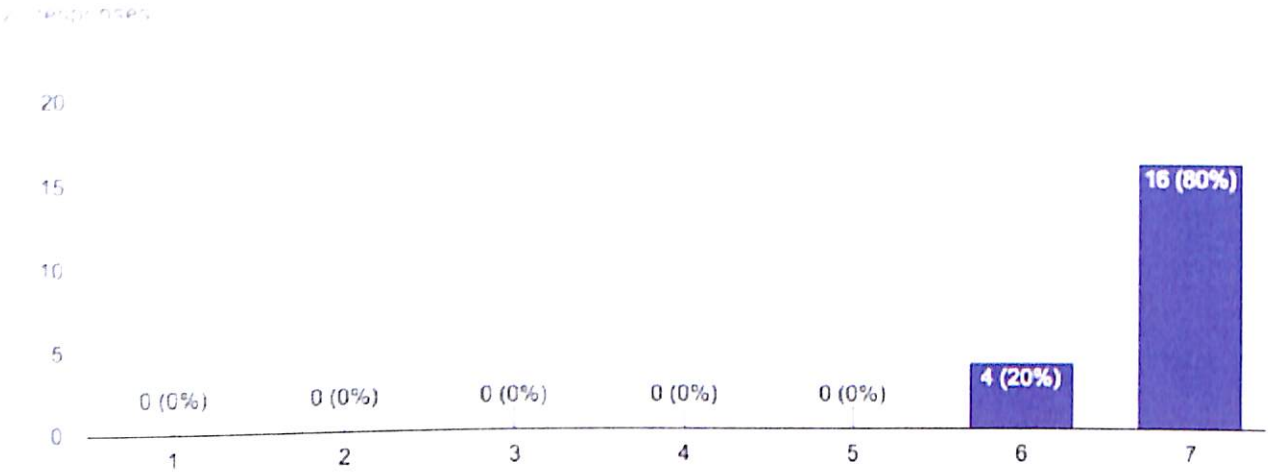


Figure L-6 Perceived Usefulness question #6 results

Appendix L (Continued)

SURVEY QUESTIONNAIRE RESULTS

Perceived Ease of Use

1. Learning to operate the new Tech.Accounts would be easy for me.

20 responses

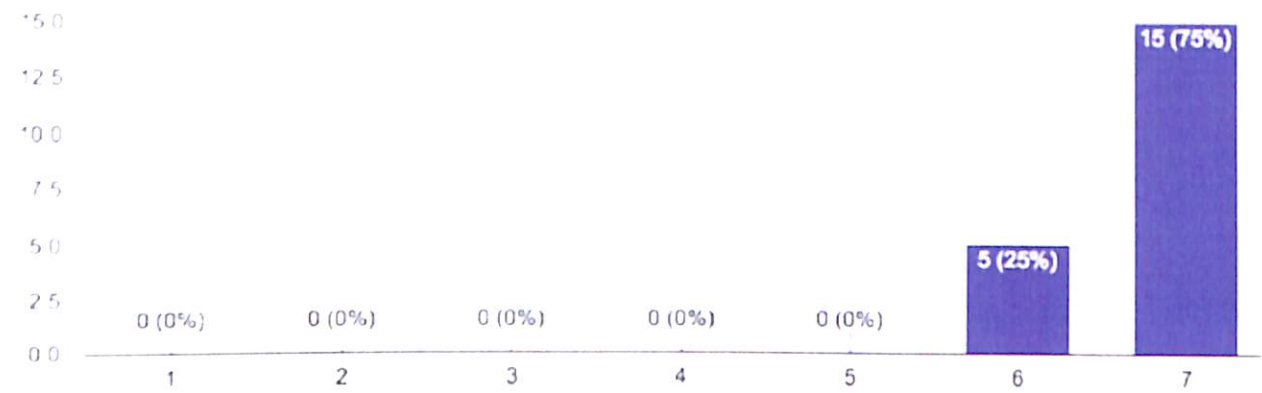


Figure L-7 Perceived Ease of Use question #1 results

2. I would find it easy to get the new Tech.Accounts to do what I want it to do.

20 responses

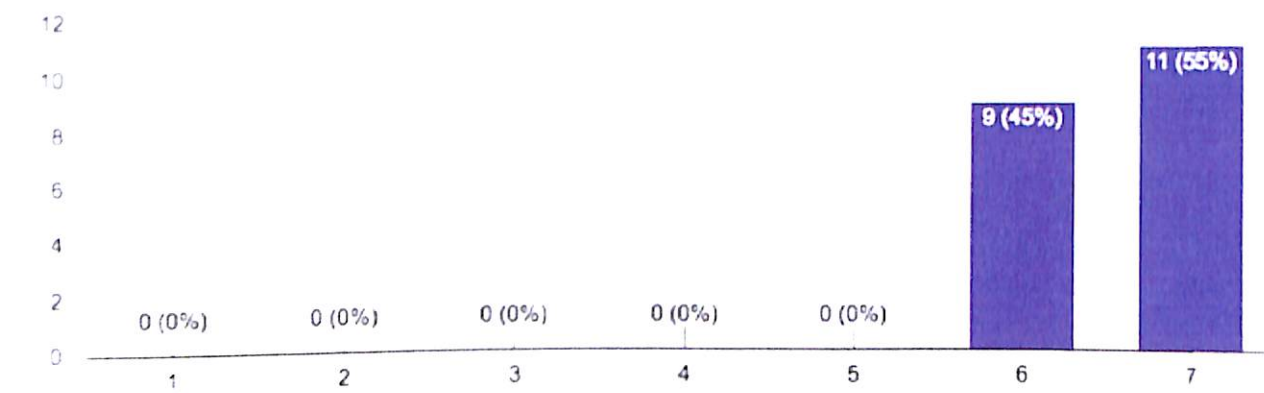


Figure L-8 Perceived Ease of Use question #2 results

Appendix L (Continued)

SURVEY QUESTIONNAIRE RESULTS

3. My interaction with the new Tech.Accounts would be clear and understandable.

RESPONSES

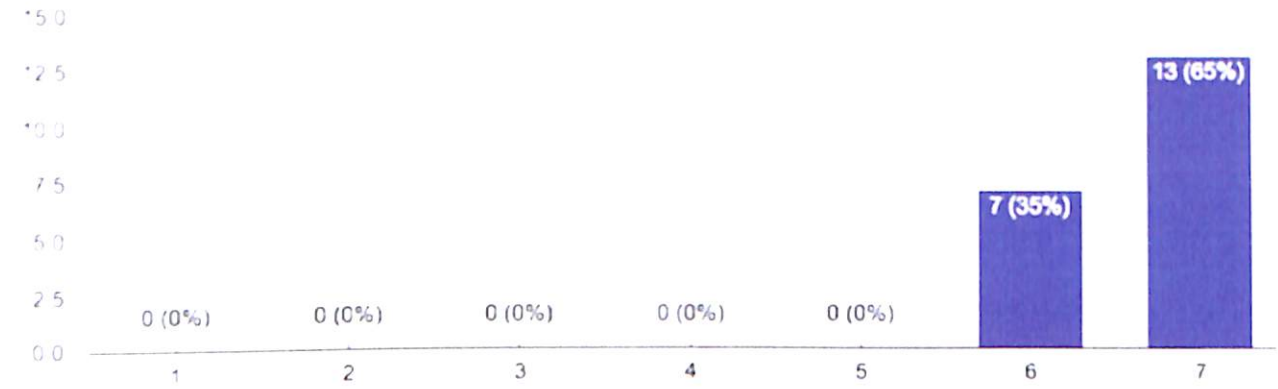


Figure L-9 Perceived Ease of Use question #3 results

4. I would find the new Tech.Accounts to be flexible to interact with.

RESPONSES

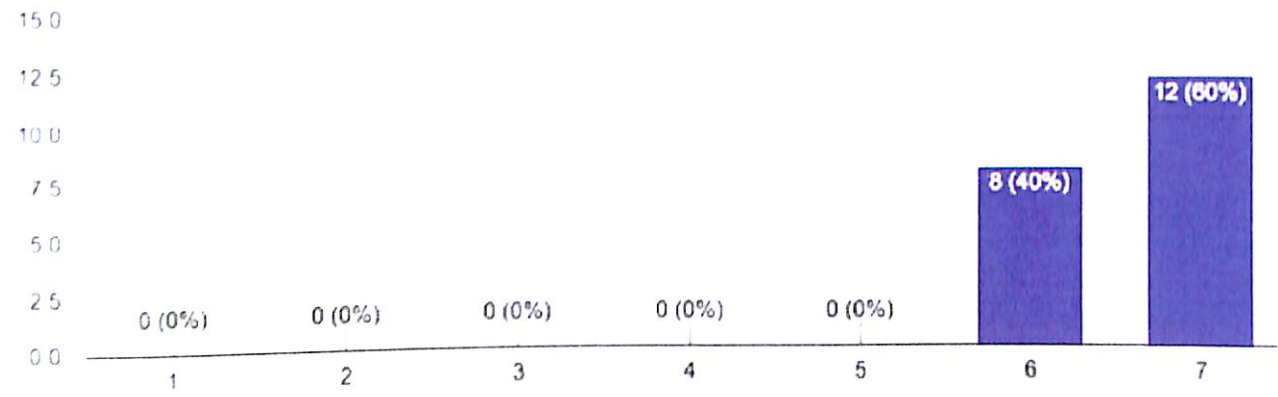


Figure L-10 Perceived Ease of Use question #4 results

Appendix L (Continued)

SURVEY QUESTIONNAIRE RESULTS

5. It would be easy for me to become skillful at using the new Tech.Accounts.

Responses

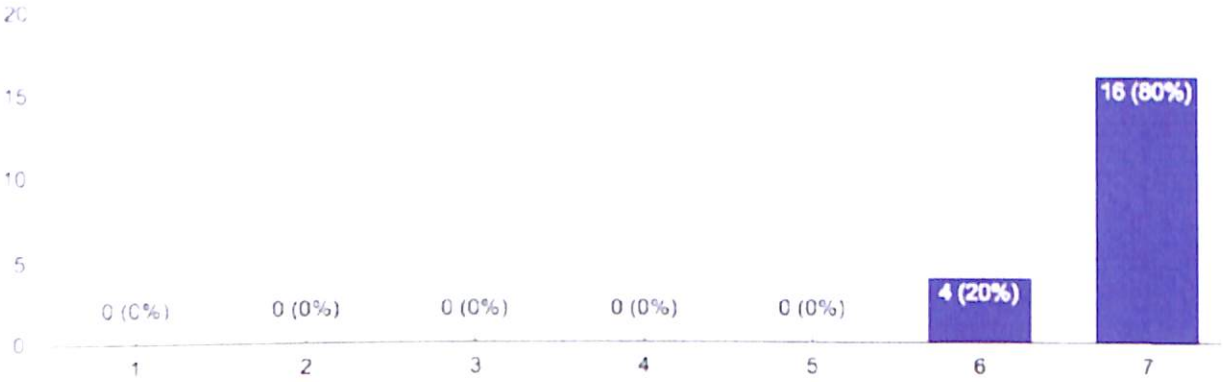


Figure L-11 Perceived Ease of Use question #5 results

6. I would find the new Tech.Accounts easy to use.

Responses

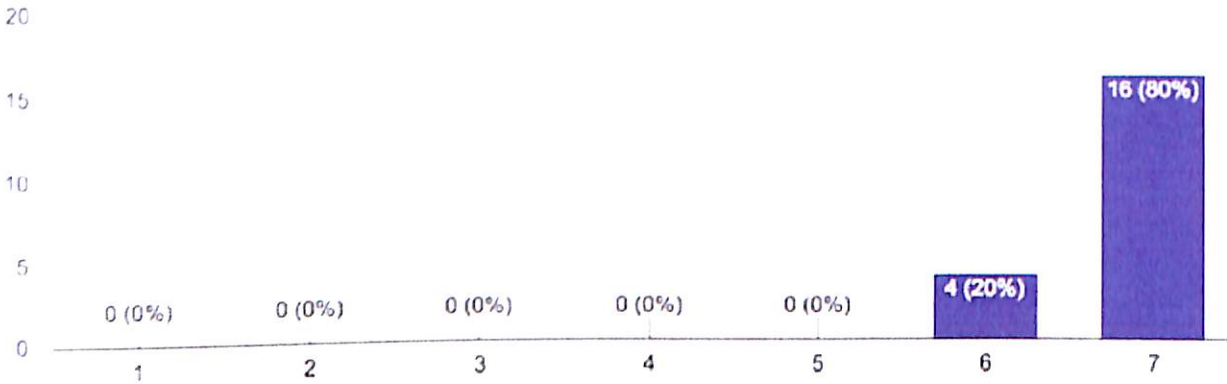


Figure L-12 Perceived Ease of Use question #6 results

Appendix L (Continued)

SURVEY QUESTIONNAIRE RESULTS

Attitude Toward Using

1. I believe it is a good idea to use the new Tech.Accounts.

4 responses

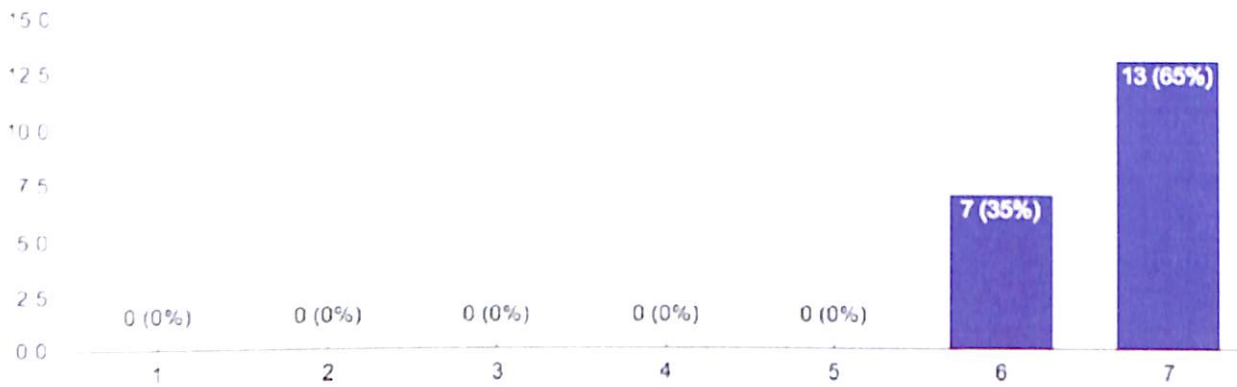


Figure L-13 Attitude Toward Using question #1 results

2. I like the idea of using the new Tech.Accounts.

4 responses

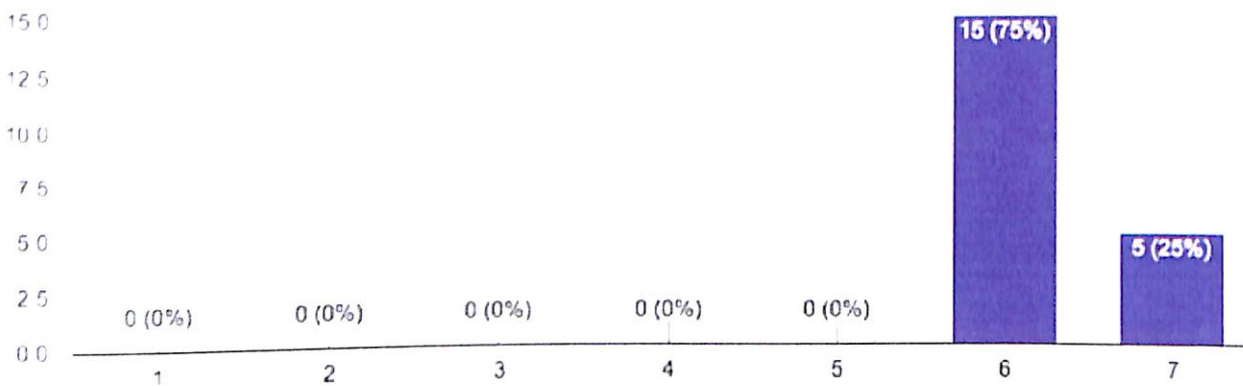


Figure L-14 Attitude Toward Using question #2 results

Appendix L (Continued)

SURVEY QUESTIONNAIRE RESULTS

3. Using the new Tech.Accounts is a positive idea.

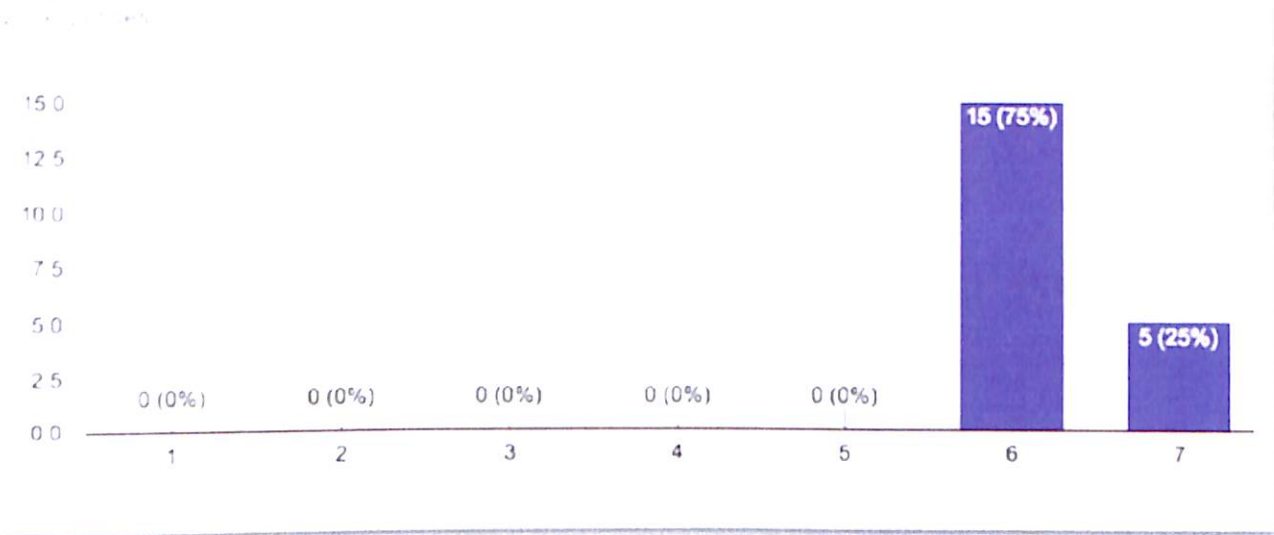


Figure L-15 Attitude Toward Using question #3 results

Appendix M

PROCESS MODELS

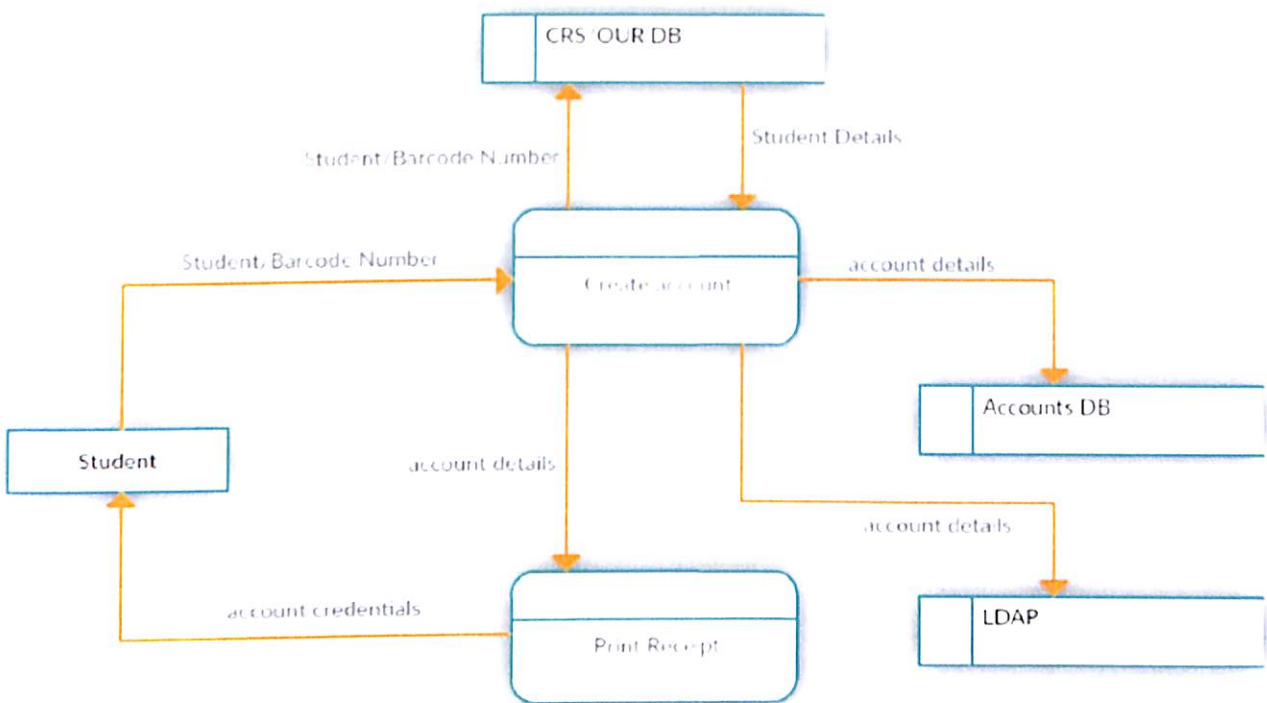


Figure M-1 Student Account Activation

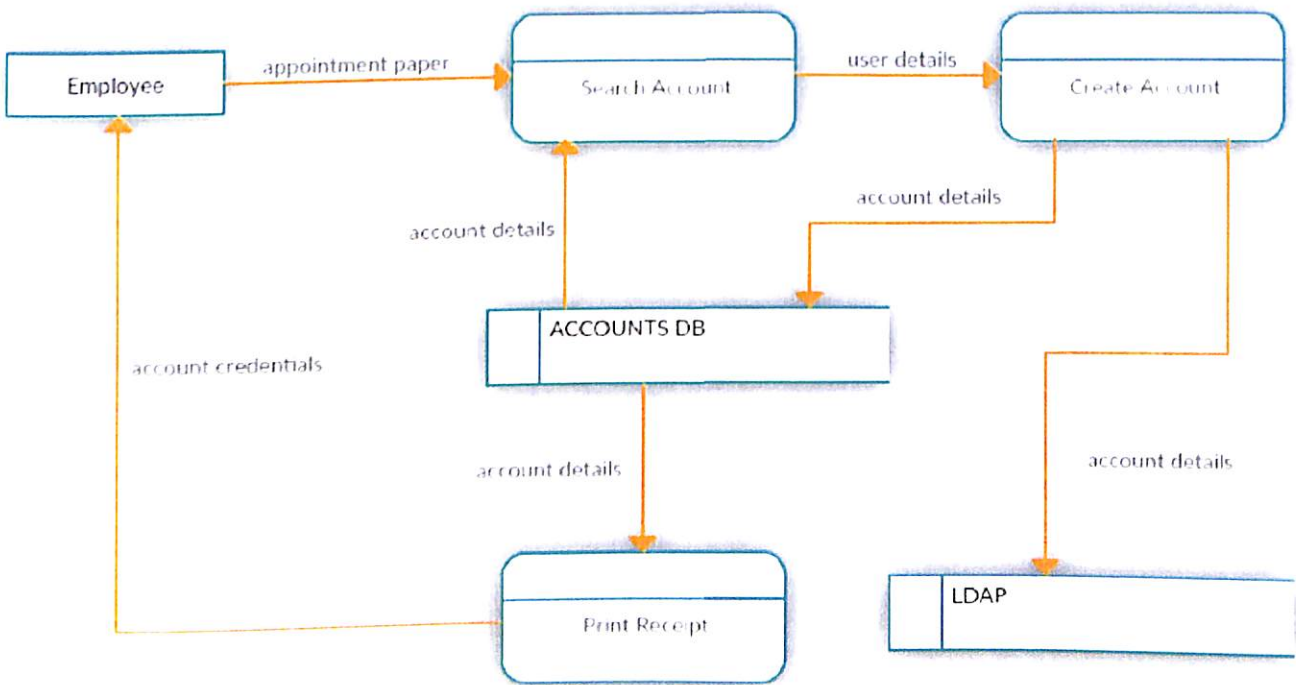


Figure M-2 Employee Account Creation

Appendix M (Continued)

PROCESS MODELS

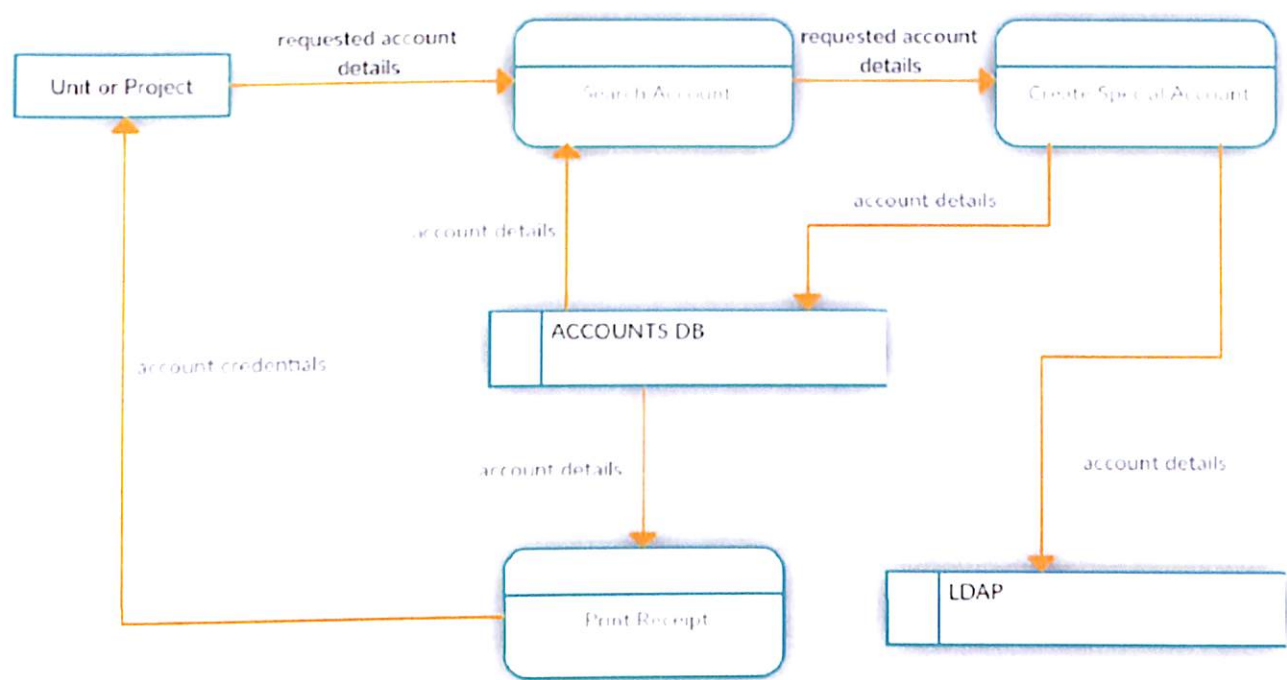


Figure M-3 Special Account Creation

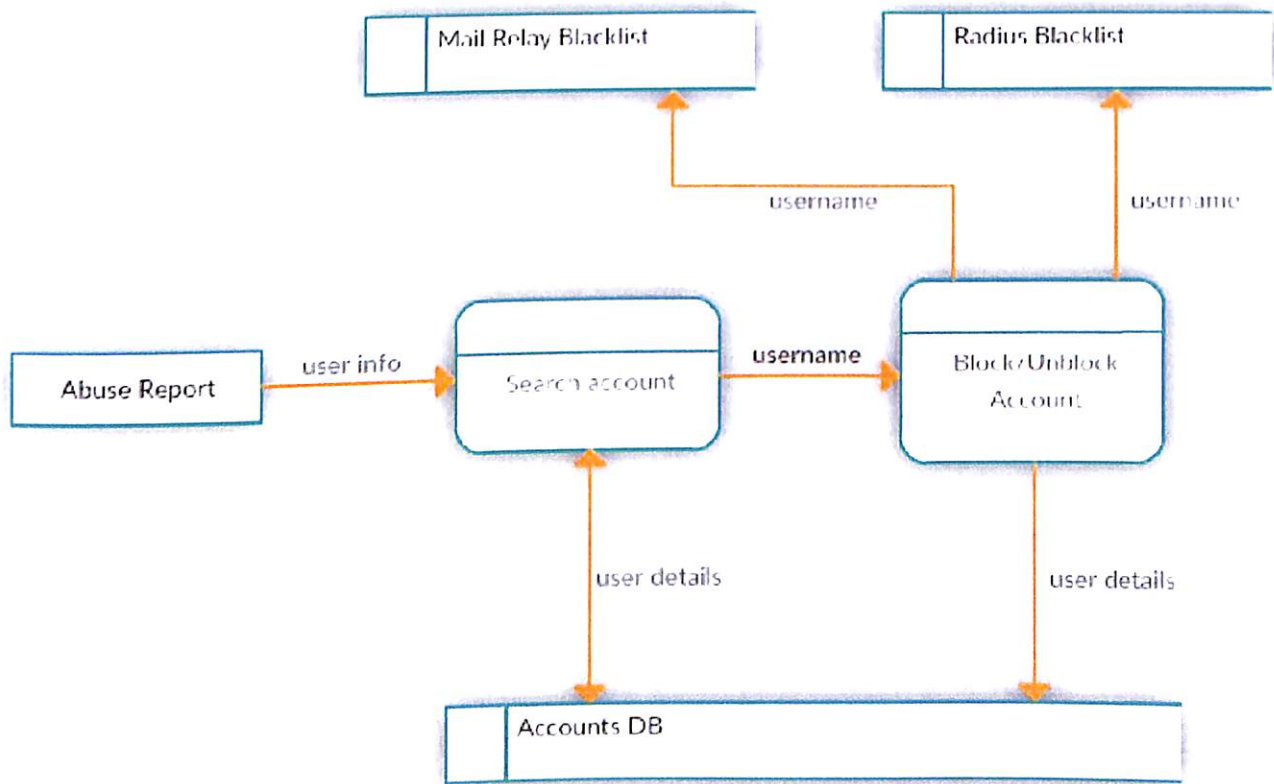


Figure M-4 Enable/Disable Account

Appendix M (Continued)

PROCESS MODELS

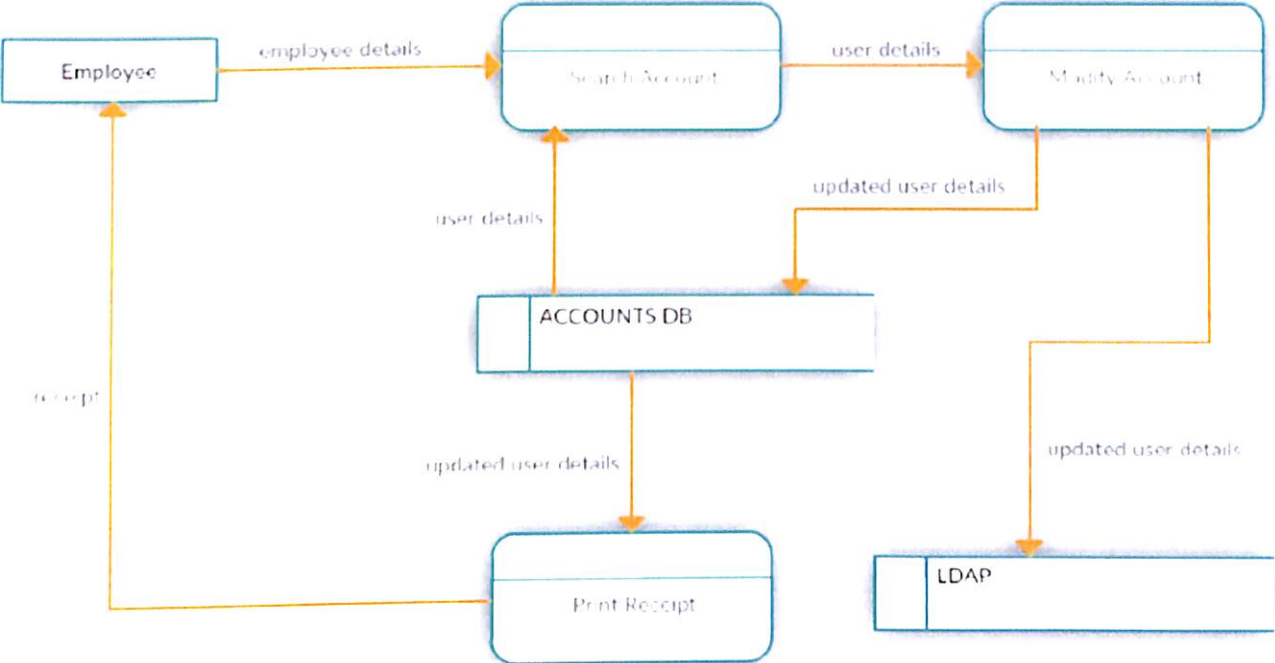


Figure M-5 Account Modification

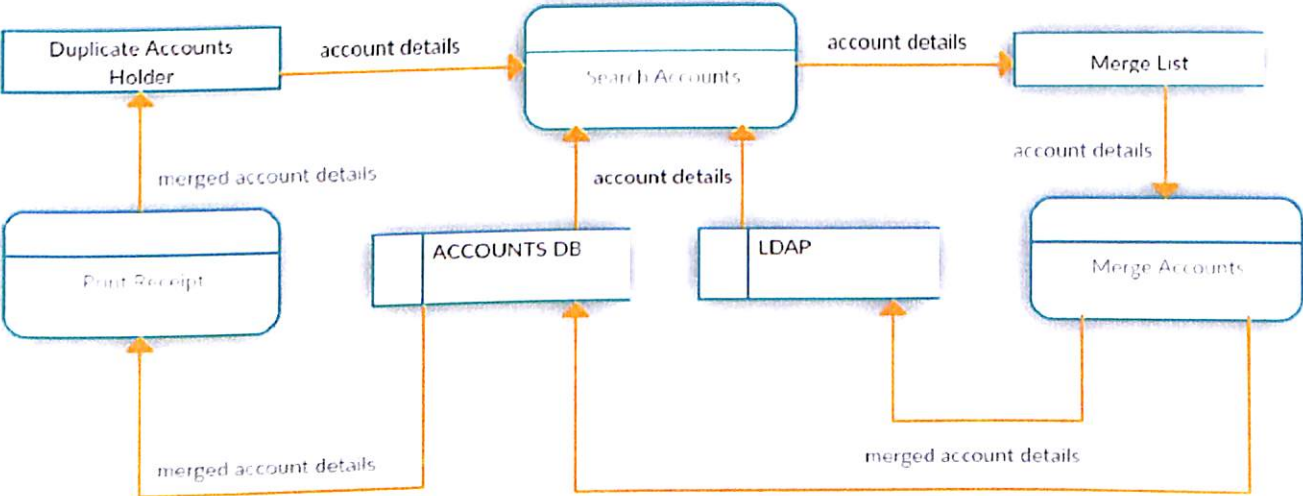


Figure M-6 Merge Duplicate Accounts

Appendix M (Continued)

PROCESS MODELS

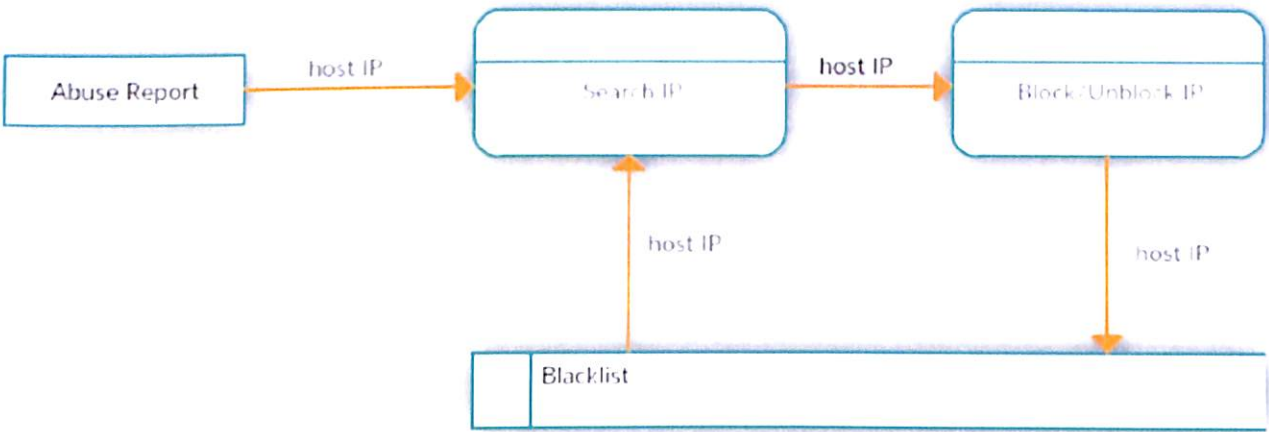


Figure M-7 Block/Unblock Host IP

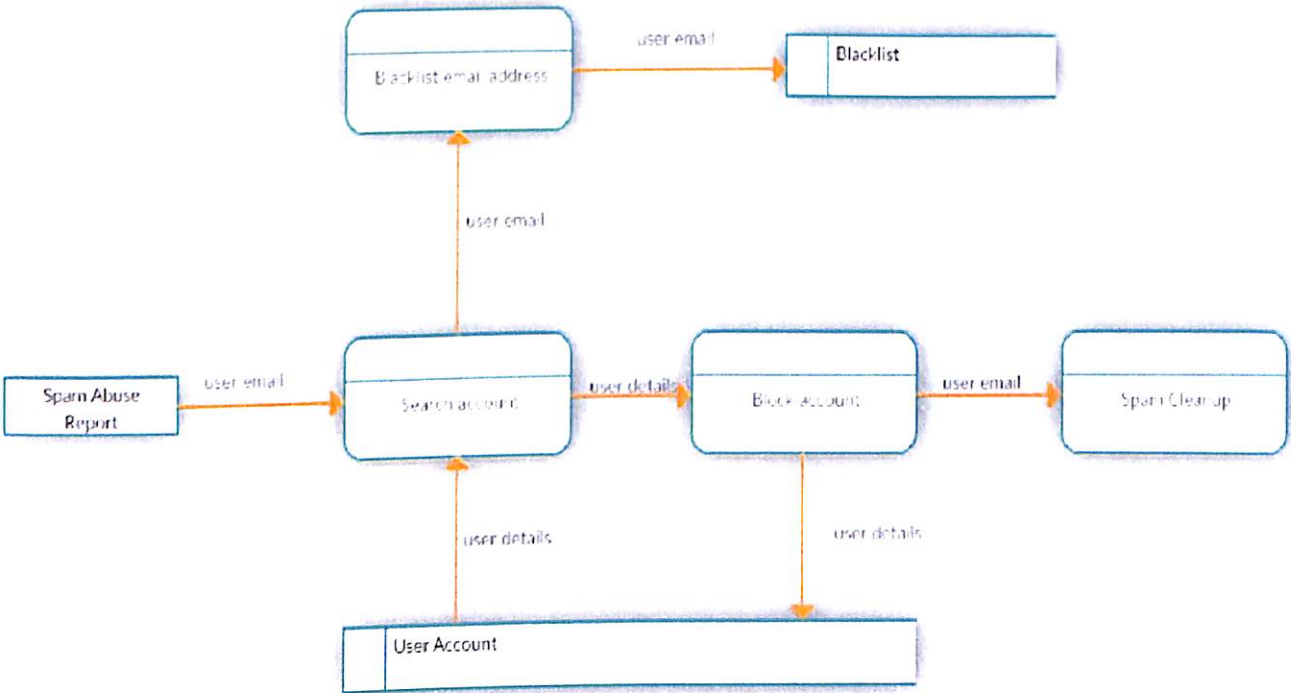


Figure M-8 Spam Cleanup

Appendix N

PROGRAM LISTING

application/config/activation.php

```
<?php  if ( ! defined('BASEPATH')) exit('No direct script access allowed');

$config['none'] = '';
$config['oldestYear'] = 1940;
$config['defaultMailQuota'] = "209715200";
$config['defaultMailHost'] = "smicro02.up.edu.ph";
$config['defaultMailHostUPD'] = "smicro02.upd.edu.ph";
$config['employeeMailQuota'] = "419430400";

$config["ay"] = "2015";
$config["termno"] = "1";

$config['notif_special_case'] = "<div class='alert alert-danger'><h3><span
style='font-size:35px'>&#9888; </span>";
$config['notif_special_case'] .= "Special case or possible duplicate webmail
account detected. ";
$config['notif_special_case'] .= "Proceed to Rm. 301 to continue with account
activation.</h3></div>";
```

application/config/config.php

```
<?php
defined('BASEPATH') OR exit('No direct script access allowed');

/*
|-----
| Base Site URL
|-----
|
| URL to your CodeIgniter root. Typically this will be your base URL,
| WITH a trailing slash:
|
| http://example.com/
|
| WARNING: You MUST set this value!
|
| If it is not set, then CodeIgniter will try guess the protocol and path
| your installation, but due to security concerns the hostname will be set
| to $_SERVER['SERVER_ADDR'] if available, or localhost otherwise.
| The auto-detection mechanism exists only for convenience during
| development and MUST NOT be used in production!
|
| If you need to allow multiple domains, remember that this file is still
| a PHP script and you can easily do that on your own.
|
*/
$config['base_url'] = '';
$config['theme_url'] = $config['base_url'].'/techaccounts/assets/AdminLTE-2.3.0';
```



```

/*
|-----|
| Index File |
|-----|

| Typically this will be your index.php file, unless you've renamed it to
| something else. If you are using mod_rewrite to remove the page set this
| variable so that it is blank.
|
*/
$config['index_page'] = '';

/*
|-----|
| URI PROTOCOL |
|-----|

| This item determines which server global should be used to retrieve the
| URI string. The default setting of 'REQUEST_URI' works for most servers.
| If your links do not seem to work, try one of the other delicious flavors:
|
| 'REQUEST_URI'    Uses $_SERVER['REQUEST_URI']
| 'QUERY_STRING'   Uses $_SERVER['QUERY_STRING']
| 'PATH_INFO'      Uses $_SERVER['PATH_INFO']
|
| WARNING: If you set this to 'PATH_INFO', URIs will always be URL-decoded!
|
*/
$config['uri_protocol'] = 'REQUEST_URI';

/*
|-----|
| URL suffix |
|-----|

| This option allows you to add a suffix to all URLs generated by CodeIgniter.
| For more information please see the user guide:
|
| https://codeigniter.com/user\_guide/general/urls.html
|
*/
$config['url_suffix'] = '';

/*
|-----|
| Default Language |
|-----|

| This determines which set of language files should be used. Make sure
| there is an available translation if you intend to use something other
| than english.
|
*/
$config['language'] = 'english';

/*
|-----|
| Default Character Set |
|-----|

```



```

| This determines which character set is used by default in various methods
| that require a character set to be provided.
|
| See http://php.net/htmlspecialchars for a list of supported charsets.
|
*/
$config['charset'] = 'UTF-8';

/*
|-----
| Enable/Disable System Hooks
|-----
|
| If you would like to use the 'hooks' feature you must enable it by
| setting this variable to TRUE (boolean).  See the user guide for details.
|
*/
$config['enable_hooks'] = FALSE;

/*
|-----
| Class Extension Prefix
|-----
|
| This item allows you to set the filename/classname prefix when extending
| native libraries.  For more information please see the user guide:
|
| https://codeigniter.com/user\_guide/general/core\_classes.html
| https://codeigniter.com/user\_guide/general/creating\_libraries.html
|
*/
$config['subclass_prefix'] = 'MY_';

/*
|-----
| Composer auto-loading
|-----
|
| Enabling this setting will tell CodeIgniter to look for a Composer
| package auto-loader script in application/vendor/autoload.php.
|
| $config['composer_autoload'] = TRUE;
|
| Or if you have your vendor/ directory located somewhere else, you
| can opt to set a specific path as well:
|
| $config['composer_autoload'] = '/path/to/vendor/autoload.php';
|
| For more information about Composer, please visit http://getcomposer.org/
|
| Note: This will NOT disable or override the CodeIgniter-specific
| autoloading (application/config/autoload.php)
|
*/
$config['composer_autoload'] = FALSE;

/*
|-----
| Allowed URL Characters
|-----

```

```
|
| This lets you specify which characters are permitted within your URLs.
| When someone tries to submit a URL with disallowed characters they will
| get a warning message.
|
| As a security measure you are STRONGLY encouraged to restrict URLs to
| as few characters as possible. By default only these are allowed: a-z 0-9~%.:_-
|
| Leave blank to allow all characters -- but only if you are insane.
|
| The configured value is actually a regular expression character group
| and it will be executed as: ! preg_match('/^[<permitted_uri_chars>]+$/i
|
| DO NOT CHANGE THIS UNLESS YOU FULLY UNDERSTAND THE REPERCUSSIONS!!
|
| */
$config['permitted_uri_chars'] = 'a-z 0-9~%.:_-';
|
| /*
| -----
| Enable Query Strings
| -----
|
| By default CodeIgniter uses search-engine friendly segment based URLs:
| example.com/who/what/where/
|
| By default CodeIgniter enables access to the $_GET array. If for some
| reason you would like to disable it, set 'allow_get_array' to FALSE.
|
| You can optionally enable standard query string based URLs:
| example.com?who=me&what=something&where=here
|
| Options are: TRUE or FALSE (boolean)
|
| The other items let you set the query string 'words' that will
| invoke your controllers and its functions:
| example.com/index.php?c=controller&m=function
|
| Please note that some of the helpers won't work as expected when
| this feature is enabled, since CodeIgniter is designed primarily to
| use segment based URLs.
|
| */
$config['allow_get_array'] = TRUE;
$config['enable_query_strings'] = FALSE;
$config['controller_trigger'] = 'c';
$config['function_trigger'] = 'm';
$config['directory_trigger'] = 'd';
|
| /*
| -----
| Error Logging Threshold
| -----
|
| You can enable error logging by setting a threshold over zero. The
| threshold determines what gets logged. Threshold options are:
|
| 0 = Disables logging, Error logging TURNED OFF
| 1 = Error Messages (including PHP errors)
```

```

| 2 = Debug Messages
| 3 = Informational Messages
| 4 = All Messages
|
| You can also pass an array with threshold levels to show individual error types
|
| array(2) = Debug Messages, without Error Messages
|
| For a live site you'll usually only enable Errors (1) to be logged otherwise
| your log files will fill up very fast.
|
| */
$config['log_threshold'] = 0;

/*
|-----
| Error Logging Directory Path
|-----
|
| Leave this BLANK unless you would like to set something other than the default
| application/logs/ directory. Use a full server path with trailing slash.
|
| */
$config['log_path'] = '/tmp/';

/*
|-----
| Log File Extension
|-----
|
| The default filename extension for log files. The default 'php' allows for
| protecting the log files via basic scripting, when they are to be stored
| under a publicly accessible directory.
|
| Note: Leaving it blank will default to 'php'.
|
| */
$config['log_file_extension'] = '';

/*
|-----
| Log File Permissions
|-----
|
| The file system permissions to be applied on newly created log files.
|
| IMPORTANT: This MUST be an integer (no quotes) and you MUST use octal
| integer notation (i.e. 0700, 0644, etc.)
|
| */
$config['log_file_permissions'] = 0644;

/*
|-----
| Date Format for Logs
|-----
|
| Each item that is logged has an associated date. You can use PHP date
| codes to set your own date formatting

```

```

*/
$config['log_date_format'] = 'Y-m-d H:i:s';

/*
|-----
| Error Views Directory Path
|-----
|
| Leave this BLANK unless you would like to set something other than the default
| application/views/errors/ directory. Use a full server path with trailing
| slash.
|
*/
$config['error_views_path'] = '';

/*
|-----
| Cache Directory Path
|-----
|
| Leave this BLANK unless you would like to set something other than the default
| application/cache/ directory. Use a full server path with trailing slash.
|
*/
$config['cache_path'] = '';

/*
|-----
| Cache Include Query String
|-----
|
| Whether to take the URL query string into consideration when generating
| output cache files. Valid options are:
|
| FALSE          = Disabled
| TRUE           = Enabled, take all query parameters into account.
|                  Please be aware that this may result in numerous cache
|                  files generated for the same page over and over again.
| array('q')     = Enabled, but only take into account the specified list
|                  of query parameters.
|
*/
$config['cache_query_string'] = FALSE;

/*
|-----
| Encryption Key
|-----
|
| If you use the Encryption class, you must set an encryption key.
| See the user guide for more info.
|
| https://codeigniter.com/user\_guide/libraries/encryption.html
|
*/
$config['encryption_key'] = 'ftLHiGoxII5iSMTtGrenCht75iykRg85';

/*
|-----

```

Session Variables

'sess_driver'

The storage driver to use: files, database, redis, memcached

'sess_cookie_name'

The session cookie name, must contain only [0-9a-z_-] characters

'sess_expiration'

The number of SECONDS you want the session to last.
Setting to 0 (zero) means expire when the browser is closed.

'sess_save_path'

The location to save sessions to, driver dependent.

For the 'files' driver, it's a path to a writable directory.
WARNING: Only absolute paths are supported!

For the 'database' driver, it's a table name.
Please read up the manual for the format with other session drivers.

IMPORTANT: You are REQUIRED to set a valid save path!

'sess_match_ip'

Whether to match the user's IP address when reading the session data.

WARNING: If you're using the database driver, don't forget to update
your session table's PRIMARY KEY when changing this setting.

'sess_time_to_update'

How many seconds between CI regenerating the session ID.

'sess_regenerate_destroy'

Whether to destroy session data associated with the old session ID
when auto-regenerating the session ID. When set to FALSE, the data
will be later deleted by the garbage collector.

Other session cookie settings are shared with the rest of the application,
except for 'cookie_prefix' and 'cookie_httponly', which are ignored here.

```
*/
$config['sess_driver'] = 'database';
$config['sess_cookie_name'] = 'ci_sess_tech';
$config['sess_expiration'] = 7200;
$config['sess_save_path'] = 'ci_sessions_techv3';
$config['sess_match_ip'] = TRUE;
$config['sess_time_to_update'] = 300;
$config['sess_regenerate_destroy'] = TRUE;
```

/*

```

-----
Cookie Related Variables
-----

'cookie_prefix'    = Set a cookie name prefix if you need to avoid collisions
'cookie_domain'    = Set to .your-domain.com for site-wide cookies
'cookie_path'      = Typically will be a forward slash
'cookie_secure'    = Cookie will only be set if a secure HTTPS connection exists.
'cookie_httponly'  = Cookie will only be accessible via HTTP(S) (no javascript)

Note: These settings (with the exception of 'cookie_prefix' and
      'cookie_httponly') will also affect sessions.

*/
$config['cookie_prefix']  = 'techaccounts';
$config['cookie_domain']  = '';
$config['cookie_path']    = '/';
$config['cookie_secure']  = FALSE;
$config['cookie_httponly'] = FALSE;

/*
-----
Standardize newlines
-----

Determines whether to standardize newline characters in input data,
meaning to replace \r\n, \r, \n occurrences with the PHP_EOL value.

This is particularly useful for portability between UNIX-based OSes,
(usually \n) and Windows (\r\n).

*/
$config['standardize_newlines'] = FALSE;

/*
-----
Global XSS Filtering
-----

Determines whether the XSS filter is always active when GET, POST or
COOKIE data is encountered

WARNING: This feature is DEPRECATED and currently available only
for backwards compatibility purposes!

*/
$config['global_xss_filtering'] = FALSE;

/*
-----
Cross Site Request Forgery
-----

Enables a CSRF cookie token to be set. When set to TRUE, token will be
checked on a submitted form. If you are accepting user data, it is strongly
recommended CSRF protection be enabled.

'csrf_token_name' = The token name
'csrf_cookie_name' = The cookie name
'csrf_expire'     = The number in seconds the token should expire.

```

```

| 'csrf_regenerate' = Regenerate token on every submission
| 'csrf_exclude_uris' = Array of URIs which ignore CSRF checks
*/
$config['csrf_protection'] = TRUE;
$config['csrf_token_name'] = 'csrf_ucc';
$config['csrf_cookie_name'] = 'csrf_cookie_ucc';
$config['csrf_expire'] = 7200;
$config['csrf_regenerate'] = FALSE;
$config['csrf_exclude_uris'] = array();

/*
|-----|
| Output Compression |
|-----|

| Enables Gzip output compression for faster page loads.  When enabled,
| the output class will test whether your server supports Gzip.
| Even if it does, however, not all browsers support compression
| so enable only if you are reasonably sure your visitors can handle it.

| Only used if zlib.output_compression is turned off in your php.ini.
| Please do not use it together with httpd-level output compression.

| VERY IMPORTANT:  If you are getting a blank page when compression is enabled it
| means you are prematurely outputting something to your browser. It could
| even be a line of whitespace at the end of one of your scripts.  For
| compression to work, nothing can be sent before the output buffer is called
| by the output class.  Do not 'echo' any values with compression enabled.

*/
$config['compress_output'] = FALSE;

/*
|-----|
| Master Time Reference |
|-----|

| Options are 'local' or any PHP supported timezone. This preference tells
| the system whether to use your server's local time as the master 'now'
| reference, or convert it to the configured one timezone. See the 'date
| helper' page of the user guide for information regarding date handling.

*/
$config['time_reference'] = 'local';

/*
|-----|
| Rewrite PHP Short Tags |
|-----|

| If your PHP installation does not have short tag support enabled CI
| can rewrite the tags on-the-fly, enabling you to utilize that syntax
| in your view files.  Options are TRUE or FALSE (boolean)

| Note: You need to have eval() enabled for this to work.

*/
$config['rewrite_short_tags'] = FALSE;

```

```

/*
-----
Reverse Proxy IPs
-----

If your server is behind a reverse proxy, you must whitelist the proxy
IP addresses from which CodeIgniter should trust headers such as
HTTP_X_FORWARDED_FOR and HTTP_CLIENT_IP in order to properly identify
the visitor's IP address.

You can use both an array or a comma-separated list of proxy addresses,
as well as specifying whole subnets. Here are a few examples:

Comma-separated: '10.0.1.200,192.168.5.0/24'
Array:           array('10.0.1.200', '192.168.5.0/24')
*/
$config['proxy_ips'] = '';

spl_autoload_extensions('.php'); // Only Autoload PHP Files

spl_autoload_register ( function ($class) {
    /*
     * PSR-4 autoloader, based on PHP Framework Interop Group snippet (Under MIT
     License.)
     * https://github.com/php-fig/fig-standards/blob/master/accepted/PSR-4-
     autoloader-examples.md
     */
    $prefix = "Mike42\\";
    $base_dir = APPPATH . "/libraries/Mike42/";

    /* Only continue for classes in this namespace */
    $len = strlen ( $prefix );
    if (strncmp ( $prefix, $class, $len ) !== 0) {
        return;
    }

    /* Require the file if it exists */
    $relative_class = substr ( $class, $len );
    $file = $base_dir . str_replace ( '\\', '/', $relative_class ) . '.php';
    if (file_exists ( $file )) {
        require $file;
    }
} );

```

application/config/config_ldap.php

```

<?php
defined('BASEPATH') OR exit('No direct script access allowed');

$config['ldap_basedn'] = 'o=University of the Philippines,c=PH';
$config['ldap_host'] = 'ldap://10.16.5.85';
$config['ldap_root'] = 'cn=root,o=University of the Philippines,c=PH';
$config['ldap_root_pass'] = 'abc123';
$config['ldap_port'] = '389';

```

```
<?php
defined('BASEPATH') OR exit('No direct script access allowed');

/*
|-----
| DATABASE CONNECTIVITY SETTINGS
|-----
| This file will contain the settings needed to access your database.
|
| For complete instructions please consult the 'Database Connection'
| page of the User Guide.
|
|-----
| EXPLANATION OF VARIABLES
|-----
|
| ['dsn']          The full DSN string describe a connection to the database.
| ['hostname']     The hostname of your database server.
| ['username']     The username used to connect to the database
| ['password']     The password used to connect to the database
| ['database']     The name of the database you want to connect to
| ['dbdriver']     The database driver. e.g.: mysqli.
|                  Currently supported:
|                      cubrid, ibase, mssql, mysql, mysqli, oci8,
|                      odbc, pdo, postgre, sqlite, sqlite3, sqlsrv
| ['dbprefix']     You can add an optional prefix, which will be added
|                  to the table name when using the Query Builder class
| ['pconnect']     TRUE/FALSE - Whether to use a persistent connection
| ['db_debug']     TRUE/FALSE - Whether database errors should be displayed.
| ['cache_on']     TRUE/FALSE - Enables/disables query caching
| ['cachedir']     The path to the folder where cache files should be stored
| ['char_set']     The character set used in communicating with the database
| ['dbcollat']     The character collation used in communicating with the database
|                  NOTE: For MySQL and MySQLi databases, this setting is only
used
|                  as a backup if your server is running PHP < 5.2.3 or MySQL <
5.0.7
|                  (and in table creation queries made with DB Forge).
|                  There is an incompatibility in PHP with
mysql_real_escape_string() which
|                  can make your site vulnerable to SQL injection if you are
using a
|                  multi-byte character set and are running versions lower than
these.
|                  Sites using Latin-1 or UTF-8 database character set and
collation are unaffected.
| ['swap_pre']     A default table prefix that should be swapped with the dbprefix
| ['encrypt']      Whether or not to use an encrypted connection.
|
|                  'mysql' (deprecated), 'sqlsrv' and 'pdo/sqlsrv' drivers accept
TRUE/FALSE
|                  'mysqli' and 'pdo/mysqli' drivers accept an array with the following
options:
|
|                  'ssl_key'    - Path to the private key file
|                  'ssl_cert'   - Path to the public key certificate file
|                  'ssl_ca'     - Path to the certificate authority file

```

```

|         'ssl_capath' - Path to a directory containing trusted CA
certificats in PEM format
|         'ssl_cipher' - List of 'allowed' ciphers to be used for the
encryption, separated by colons (':')
|         'ssl_verify' - TRUE/FALSE; Whether verify the server
certificate or not ('mysqli' only)
|
| ['compress'] Whether or not to use client compression (MySQL only)
| ['stricton'] TRUE/FALSE - forces 'Strict Mode' connections
|                 - good for ensuring strict SQL while
developing
| ['ssl_options'] Used to set various SSL options that can be used when making
SSL connections.
| ['failover'] array - A array with 0 or more data for connections if the main
should fail.
| ['save_queries'] TRUE/FALSE - Whether to "save" all executed queries.
|                 NOTE: Disabling this will also effectively disable both
|                 $this->db->last_query() and profiling of DB queries.
|                 When you run a query, with this setting set to TRUE (default),
|                 CodeIgniter will store the SQL statement for debugging
purposes.
|                 However, this may cause high memory usage, especially if you
run
|                 a lot of SQL queries ... disable this to avoid that problem.
|
| The $active_group variable lets you choose which connection group to
make active. By default there is only one group (the 'default' group).
|
| The $query_builder variables lets you determine whether or not to load
the query builder class.
*/
$active_group = 'default';
$query_builder = TRUE;

$db['default'] = array(
    'dsn' => '',
    'hostname' => 'localhost',
    'username' => 'accountsdb',
    'password' => 'abc123',
    'database' => 'accountsdb',
    'dbdriver' => 'mysqli',
    'dbprefix' => '',
    'pconnect' => FALSE,
    'db_debug' => (ENVIRONMENT !== 'production'),
    'cache_on' => FALSE,
    'cachedir' => '',
    'char_set' => 'utf8',
    'dbcollat' => 'utf8_general_ci',
    'swap_pre' => '',
    'encrypt' => FALSE,
    'compress' => FALSE,
    'stricton' => FALSE,
    'failover' => array(),
    'save_queries' => TRUE
);

$db['webmail'] = array(
    'dsn' => '',
    'hostname' => 'localhost',

```

```

'username' => 'accountsdb',
'password' => 'abc123',
'database' => 'rcmaildbv1',
'dbdriver' => 'mysqli',
'dbprefix' => '',
'pconnect' => FALSE,
'db_debug' => (ENVIRONMENT !== 'production'),
'cache_on' => FALSE,
'cachedir' => '',
'char_set' => 'utf8',
'dbcollat' => 'utf8_general_ci',
'swap_pre' => '',
'encrypt' => FALSE,
'compress' => FALSE,
'stricton' => FALSE,
'failover' => array(),
'save_queries' => TRUE

```

```
);
```

```

$db['radius'] = array(
    'dsn' => '',
    'hostname' => '10.16.5.135',
    'username' => 'radius',
    'password' => 'radpass',
    'database' => 'radius',
    'dbdriver' => 'mysqli',
    'dbprefix' => '',
    'pconnect' => FALSE,
    'db_debug' => (ENVIRONMENT !== 'production'),
    'cache_on' => FALSE,
    'cachedir' => '',
    'char_set' => 'utf8',
    'dbcollat' => 'utf8_general_ci',
    'swap_pre' => '',
    'encrypt' => FALSE,
    'compress' => FALSE,
    'stricton' => FALSE,
    'failover' => array(),
    'save_queries' => TRUE

```

```
);
```

```

$db['dilnetlog'] = array(
    'dsn' => '',
    'hostname' => 'localhost',
    'username' => 'root',
    'password' => 'abc123',
    'database' => 'dilnetlog',
    'dbdriver' => 'mysqli',
    'dbprefix' => '',
    'pconnect' => FALSE,
    'db_debug' => (ENVIRONMENT !== 'production'),
    'cache_on' => FALSE,
    'cachedir' => '',
    'char_set' => 'utf8',
    'dbcollat' => 'utf8_general_ci',
    'swap_pre' => '',
    'encrypt' => FALSE,
    'compress' => FALSE,

```

```
'stricton' => FALSE,  
'failover' => array(),  
'save_queries' => TRUE
```

```
);
```

```
$db['mailban'] = array(  
    'dsn' => '',  
    'hostname' => 'localhost',  
    'username' => 'root',  
    'password' => 'abc123',  
    'database' => 'maildb',  
    'dbdriver' => 'mysqli',  
    'dbprefix' => '',  
    'pconnect' => FALSE,  
    'db_debug' => (ENVIRONMENT !== 'production'),  
    'cache_on' => FALSE,  
    'cachedir' => '',  
    'char_set' => 'utf8',  
    'dbcollat' => 'utf8_general_ci',  
    'swap_pre' => '',  
    'encrypt' => FALSE,  
    'compress' => FALSE,  
    'stricton' => FALSE,  
    'failover' => array(),  
    'save_queries' => TRUE  
);
```

```
$db['proxydb'] = array(  
    'dsn' => '',  
    'hostname' => 'localhost',  
    'username' => 'root',  
    'password' => 'abc123',  
    'database' => 'proxydb',  
    'dbdriver' => 'mysqli',  
    'dbprefix' => '',  
    'pconnect' => FALSE,  
    'db_debug' => (ENVIRONMENT !== 'production'),  
    'cache_on' => FALSE,  
    'cachedir' => '',  
    'char_set' => 'utf8',  
    'dbcollat' => 'utf8_general_ci',  
    'swap_pre' => '',  
    'encrypt' => FALSE,  
    'compress' => FALSE,  
    'stricton' => FALSE,  
    'failover' => array(),  
    'save_queries' => TRUE  
);
```

application/config/email.php

```
<?php if ( ! defined('BASEPATH')) exit('No direct script access allowed');
```

```
$config['protocol'] = 'smtp';
$config['charset'] = 'iso-8859-1';
$config['wordwrap'] = TRUE;
$config['smtp_host'] = 'smtp.upd.edu.ph';
$config['protocol'] = 'smtp';
$config['mailtype'] = 'text';
$config['wrapchars'] = 76;
$config['smtp_port'] = 25;
$config['crlf'] = "\r\n";
$config['newline'] = "\r\n";
```

application/controllers/supertech/Crs_apichecker.php

```
<?php
defined('BASEPATH') OR exit('No direct script access allowed');

class Crs_apichecker extends MY_Controller {

    /**
     * Index Page for this controller.
     *
     * Maps to the following URL
     *      http://example.com/index.php/welcome
     * - or -
     *      http://example.com/index.php/welcome/index
     * - or -
     * Since this controller is set as the default controller in
     * config/routes.php, it's displayed at http://example.com/
     *
     * So any other public methods not prefixed with an underscore will
     * map to /index.php/welcome/<method_name>
     * @see https://codeigniter.com/user_guide/general/urls.html
     */

    function __construct(){
        parent::__construct();

        // Load Markdown
        $this->load->library('markdown');

        $this->form_validation->set_error_delimiters('<label>', '</label>');
        $this->load->model('crs_model');

        $this->page_vars = array(
            'content' => 'crs_apichecker',
            'page_title' => 'CRS API checker',
            'page_subtitle' => 'View values returned by CRS API',
            'include_stylesheets' => 'dilnet_lookup_css', //Reuse CSS and JS
calls from Dilnet account
            'include_scripts' => 'dilnet_lookup_js',
            'crs_apidoc' => $this->markdown->
>parse_file(APPPATH.'views/crs_apidoc.md')
        );
```

```

}

public function index()
{
    $data = $this->page_vars;
    $this->load->vars($data);
    $this->load->view('page_template');
}

public function getstudentinfo() {
    // Load page variables
    $data = $this->page_vars;

    $this->form_validation->set_rules('input_str', 'Input', 'required');

    if ($this->form_validation->run() == TRUE){
        $search_string = $this->input->post('input_str');
        #TODO: Sanitize search string
        if($search_string != "" || $search_string != NULL) {
            $data["results_crs"] = $this->crs_model-
>_getstudentinfo($search_string);
            $data["search_str"] = $search_string;
        }
    }

    $this->load->vars($data);
    $this->load->view('page_template');
}

public function getstudentnobyusername() {
    // Load page variables
    $data = $this->page_vars;

    $this->form_validation->set_rules('input_str', 'ID Number',
'required|alpha_dash');

    if ($this->form_validation->run() == TRUE){
        $search_string = $this->input->post('input_str');

        #TODO: Sanitize search string
        if($search_string != "" || $search_string != NULL) {
            $data["results_crs"] = $this->crs_model-
>_getstudentnobyusername($search_string);
            $data["search_str"] = $search_string;
        }
    }

    $this->load->vars($data);
    $this->load->view('page_template');
}

public function isenrolledbystudentno() {
    // Load page variables
    $data = $this->page_vars;

    $this->form_validation->set_rules('input_str', 'ID Number',
'required|alpha_dash');

    if ($this->form_validation->run() == TRUE){

```

```

        $search_string = $this->input->post('input_str');
        $year = $this->input->post('year');
        $term = $this->input->post('term');
        #TODO: Sanitize search string
        if($search_string != "" || $search_string != NULL) {
            $data["results_crs"] = $this->crs_model-
>_isenrolledbystudentno($search_string,$year,$term);
            $data["search_str"] = $search_string;
        }
    }

    $this->load->vars($data);
    $this->load->view('page_template');
}

public function isenrolledbybarcode() {
    // Load page variables
    $data = $this->page_vars;

    $this->form_validation->set_rules('input_str', 'ID Number',
'required|alpha_dash');

    if ($this->form_validation->run() == TRUE){
        $search_string = $this->input->post('input_str');

        #TODO: Sanitize search string
        if($search_string != "" || $search_string != NULL) {
            $data["results_crs"] = $this->crs_model-
>_isenrolledbybarcode($search_string);
            $data["search_str"] = $search_string;
        }
    }

    $this->load->vars($data);
    $this->load->view('page_template');
}
}
}

```

application/controllers/supertech/Ldap_lookup.php

```

<?php
defined('BASEPATH') OR exit('No direct script access allowed');

class Ldap_lookup extends MY_Controller {

    /**
     * Index Page for this controller.
     *
     * Maps to the following URL
     *      http://example.com/index.php/welcome
     *  - or -
     *      http://example.com/index.php/welcome/index
     *  - or -
     * Since this controller is set as the default controller in
     * config/routes.php, it's displayed at http://example.com/
     *
     */
}

```

```

* So any other public methods not prefixed with an underscore will
* map to /index.php/welcome/<method_name>
* @see https://codeigniter.com/user_guide/general/urls.html
*/
function __construct(){
    parent::__construct();

    $this->form_validation->set_error_delimiters('<label>', '</label>');
    $this->load->model('ldap_model');

    $this->page_vars = array(
        'content' => 'ldap_account_lookup',
        'page_title' => 'LDAP Lookup',
        'page_subtitle' => 'Get account details from LDAP',
        'include_stylesheets' => 'dilnet_lookup_css',
        'include_scripts' => 'dilnet_lookup_js'
    );
}

public function index()
{
    $data = $this->page_vars;
    $this->load->vars($data);
    $this->load->view('page_template');
}

public function name_search() {
    // Load page variables
    $data = $this->page_vars;

    $this->form_validation->set_rules('name_str', 'Name', 'required');

    if ($this->form_validation->run() == TRUE){
        $search_string = $this->input->post('name_str');
        #TODO: Sanitize search string
        if($search_string != "" || $search_string != NULL) {
            $data["results_ldap"] = $this->ldap_model-
>search_substring($search_string,
array('uid','cn','mail','studentnumber','employeenumber'));
            $data["search_str"] = $search_string;
        }
    }

    $this->load->vars($data);
    $this->load->view('page_template');
}

public function mail_search() {
    // Load page variables
    $data = $this->page_vars;

    $this->form_validation->set_rules('mail_str', 'Email address', 'required');

    if ($this->form_validation->run() == TRUE){
        $search_string = $this->input->post('mail_str');
        #TODO: Sanitize search string
        if($search_string != "" || $search_string != NULL) {

```



```

        $data["results_ldap"] = $this->ldap_model-
>search_mail_addresses($search_string,
array('uid','cn','mail','studentnumber','employeenumber'));
        $data["search_str"] = $search_string;
    }
}

$this->load->vars($data);
$this->load->view('page_template');
}

public function id_search() {
    // Load page variables
    $data = $this->page_vars;

    $this->form_validation->set_rules('id_str', 'ID Number',
'required|alpha_dash');

    if ($this->form_validation->run() == TRUE){
        $search_string = $this->input->post('id_str');

        #TODO: Sanitize search string
        if($search_string != "" || $search_string != NULL) {
            $data["results_ldap"] = $this->ldap_model-
>search_substring_number($search_string);
            $data["search_str"] = $search_string;
        }
    }

    $this->load->vars($data);
    $this->load->view('page_template');
}

public function view_account($username = NULL) {

    // Load page variables
    $data = array(
        'content' => 'ldap_account_view',
        'page_title' => 'View LDAP Entry',
        'page_subtitle' => 'Review this user\'s LDAP information',
        'include_stylesheets' => null,
        'include_scripts' => null
    );

    if (isset($username) || $username != NULL){
        $data['account'] = $this->ldap_model->get_user_account($username);
        $data['username'] = $username;
    }

    //TODO: Add ELSE statement

    $this->load->vars($data);
    $this->load->view('page_template');
}
}

```

application/controllers/supertech/Mysql_lookup.php

```
<?php
defined('BASEPATH') OR exit('No direct script access allowed');

class Mysql_lookup extends MY_Controller {

    /**
     * Index Page for this controller.
     *
     * Maps to the following URL
     *      http://example.com/index.php/welcome
     * - or -
     *      http://example.com/index.php/welcome/index
     * - or -
     * Since this controller is set as the default controller in
     * config/routes.php, it's displayed at http://example.com/
     *
     * So any other public methods not prefixed with an underscore will
     * map to /index.php/welcome/<method_name>
     * @see https://codeigniter.com/user_guide/general/urls.html
     */

    function __construct(){
        parent::__construct();

        $this->form_validation->set_error_delimiters('<label>', '</label>');
        $this->load->model('mysql_model');

        $this->page_vars = array(
            'content' => 'mysql_account_lookup',
            'page_title' => 'MySQL Lookup',
            'page_subtitle' => 'Get account details from database',
            'include_stylesheets' => 'dilnet_lookup_css', //Reuse CSS and JS
calls from Dilnet account
            'include_scripts' => 'dilnet_lookup_js'
        );
    }

    public function index()
    {
        $data = $this->page_vars;
        $this->load->vars($data);
        $this->load->view('page_template');
    }

    public function name_search() {
        // Load page variables
        $data = $this->page_vars;

        $this->form_validation->set_rules('name_str', 'Name', 'required');

        if ($this->form_validation->run() == TRUE){
            $search_string = $this->input->post('name_str');
            #TODO: Sanitize search string
        }
    }
}
```

```

        if($search_string != "" || $search_string != NULL) {
            $data["results_db"] = $this->mysql_model->search_substring($search_string);
            $data["search_str"] = $search_string;
        }
    }

    $this->load->vars($data);
    $this->load->view('page_template');
}

public function view_account($username = NULL) {

    // Load page variables
    $data = array(
        'content' => 'mysql_account_view',
        'page_title' => 'View MySQL Entry',
        'page_subtitle' => 'Review this user\'s MySQL information',
        'include_stylesheets' => null,
        'include_scripts' => null
    );

    if (isset($username) || $username != NULL){
        $data['account'] = $this->mysql_model->get_user_account($username);
        $data['username'] = $username;
    }

    //TODO: Add ELSE statement

    $this->load->vars($data);
    $this->load->view('page_template');
}

public function id_search() {
    // Load page variables
    $data = $this->page_vars;

    $this->form_validation->set_rules('id_str', 'ID Number',
'required|alpha_dash');

    if ($this->form_validation->run() == TRUE){
        $search_string = $this->input->post('id_str');

        #TODO: Sanitize search string
        if($search_string != "" || $search_string != NULL) {
            $data["results_db"] = $this->mysql_model->search_substring_number($search_string);
            $data["search_str"] = $search_string;
        }
    }

    $this->load->vars($data);
    $this->load->view('page_template');
}
}

```

application/controllers/tech/Create_special.php

```
<?php
defined('BASEPATH') OR exit('No direct script access allowed');

class Create_special extends MY_Controller {

    /**
     * Index Page for this controller.
     *
     * Maps to the following URL
     *      http://example.com/index.php/welcome
     * - or -
     *      http://example.com/index.php/welcome/index
     * - or -
     * Since this controller is set as the default controller in
     * config/routes.php, it's displayed at http://example.com/
     *
     * So any other public methods not prefixed with an underscore will
     * map to /index.php/welcome/<method_name>
     * @see https://codeigniter.com/user_guide/general/urls.html
     */

    function __construct(){
        parent::__construct();

        $this->form_validation->set_error_delimiters('<label>', '</label>');
        $this->load->model('ldap_model');
        $this->load->model('user_model');
        $this->load->model('mysql_model');
        $this->load->model('log_model');

        $this->page_vars = array(
            'content' => 'account_special_create',
            'page_title' => 'Create Special Account',
            'page_subtitle' => 'Manually create a DilNet / LDAP account with
custom username or email',
            'include_stylesheets' => 'account_user_create_css',
            'include_scripts' => 'account_user_create_js'
        );

        $this->tech_user = $_SESSION['tech_user'];
        $this->ip = $this->input->ip_address();
    }

    public function index() {

        $this->form_validation->set_rules('username_special', 'Username',
'required|xss_clean|trim|callback__check_username_special');
        $this->form_validation->set_rules('email_special', 'Email',
'required|callback__check_email_special');
        $this->form_validation->set_rules('id_number', 'Employee Number',
'min_length[9]');
```

```

        $this->form_validation->set_rules('studentno', 'Student Number',
'exact_length[10]|regex_match[/^[1-2][0-9][0-9][0-9]-[0-9][0-9][0-9][0-9][0-9]+$/'']);
        $this->form_validation->set_rules('firstname', 'First Name',
'required|xss_clean|min_length[2]|trim');
        $this->form_validation->set_rules('middlename', 'Middle Name',
'xss_clean|min_length[1]|trim|callback__check_blank_middlename');
        $this->form_validation->set_rules('lastname', 'Last Name',
'xss_clean|min_length[2]|trim|callback__check_blank_lastname');
        $this->form_validation->set_rules('unit', 'Unit', 'required|alpha_dash');
        $this->form_validation->set_rules('division', 'Division',
'required|alpha_dash');
        $this->form_validation->set_rules('designation', 'Designation',
'required');
        $this->form_validation->set_rules('type', 'Account Type', 'required');
        $this->form_validation->set_rules('mailquotasize', 'Mail Quota Size',
'required');

```

```

if ($this->form_validation->run() == TRUE){
    $account_form = $this->input->post(NULL, TRUE);

    $_SESSION['create_special'] = $account_form;

    $this->session->mark_as_flash('create_special');

    redirect('tech/create_special/review');
}

```

```

if(isset($_SESSION['create_special_back'])){
    $_POST = $_SESSION['create_special_back'];
    unset($_SESSION['create_special_back']);
}

```

```

$data = $this->page_vars;
$data['desig_list'] = $this->mysql_model->get_designation_list();
$data['unit_list'] = $this->mysql_model->get_unit_list();
$data['division_list'] = $this->mysql_model->get_division_list();

$this->load->vars($data);
$this->load->view('page_template');

```

```

}

```

```

##CREATE USER :: REVIEW
public function review() {

```

```

    if(isset($_SESSION['create_special'])){

```

```

        $data = $this->page_vars;

```

```

        $data['content'] = 'account_special_create_confirm';
        $data['account'] = $account_form = $_SESSION['create_special'];

```

```

        #'Account_fieldnames'

```

```

        //Add pretty output for input field names here.

```

```

        //If key is not found in this array, then it will display the

```

```

        default field name

```

```

        $data['account_fieldnames'] = array(

```

```

        'type' => 'Account Type',
        'firstname' => 'First Name',
        'middlename' => 'Middle Name',
        'lastname' => 'Last Name',
        'unit' => 'Unit / College',
        'division' => 'Division / Department',
        'id_number' => 'Employee Number',
        'studentno' => 'Student Number',
        'designation' => 'Designation',
        'bypass' => 'Special Case',
        'mn_none' => 'No Middle Name',
        'ln_none' => 'No Last Name',
        'mailquotasize' => 'Mail Quota Size',
        'username_special' => 'Username',
        'email_special' => 'Email'
    );

```

```

        //Change session key for next and previous step (so that the data
        will be destroyed if "review" page is refreshed)

```

```

        $_SESSION['create_special_back'] =
        $_SESSION['create_special_confirm'] = $account_form;

```

```

        $this->load->vars($data);
        $this->load->view('page_template');

```

```

    }

```

```

    else {

```

```

        // Store in session for the final step
        $this->session->mark_as_flash('create_special_confirm');
        $this->session->mark_as_flash('create_special_back');

```

```

        if(!isset($_SESSION['create_special_back'])){
            $_SESSION['error'] = "The page was refreshed so the form data
was lost. Fill up the form again.";
            $this->session->mark_as_flash('error');

```

```

        }

```

```

        redirect('tech/create_special');

```

```

    }

```

```

}

```

```

//CREATE USER :: FINAL STEP FOR ACCOUNT CREATION

```

```

public function activate(){
    #TODO catch isset
    $account = $_SESSION['create_special_confirm'];
    // print_r($account);

```

```

    # Get name
    $first = $name['firstname'] = isset($account['firstname']) ?
ucwords($account['firstname']) : NULL;
    $middle = $name['middlename'] = isset($account['middlename']) ?
ucwords($account['middlename']) : NULL;
    $last = $name['lastname'] = isset($account['lastname']) ?
ucwords($account['lastname']) : NULL;

```

```

    #If no last name, use first name as last name
    if($last == NULL) {
        $last = $first;

```

```

}

#TODO use first, middle, last

if($this->_is_special_case($first,$middle,$last) &&
!array_key_exists("bypass", $account)){
    #TODO Add link to duplicate user
    $_SESSION['create_special_back'] =
$_SESSION['create_special_confirm'];
    $_SESSION["warning"] = "Special case detected (fullname match or
double 'none'). <br/> Check bypass to proceed.";
    $this->session->mark_as_flash('warning');
    $this->session->mark_as_flash('create_special_back');
    redirect('tech/create_special');
}

#TODO use ready array
else if($this->_is_duplicate($account) && !array_key_exists("bypass",
$account)) {
    #TODO Add links to duplicate users
    $_SESSION['create_special_back'] =
$_SESSION['create_special_confirm'];
    $_SESSION["warning"] = "Duplicate student/employee number detected.
<br/> Check bypass if you want to proceed.";
    $this->session->mark_as_flash('warning');
    $this->session->mark_as_flash('create_special_back');
    redirect('tech/create_special');
}
else {
    # find available username & email
    // $this->Logs_model->create($user["id_number"], "creating
account", "creation", $ip, $currentuser);

    # Begin processing available email and username
    $username = $account['username_special'];
    #TODO Check this username against CRS data
    $account_email = $account['email_special'];

    #TODO Add isset $username, $account_email
    #TODO Add qmail uid and qmail gid

    if(filter_var($account_email, FILTER_VALIDATE_EMAIL))
    { // full email was provided)
        list($email_only, $domain_only) = explode("@",$account_email);
        if(strtolower($domain_only)=="upd.edu.ph")
        {
            $mail_full = $account_email;
        }
    }
    else
    {
        $mail_full = $account_email."@upd.edu.ph";
        $email_only = $account_email;
    }

    $ldap_data = array(

```

```

        'uid' => $username,
        'cn' => $first." ".$last,
        'sn' => $last,
        'mail' => $mail_full,
        'employeenumber' => $this->_format_array_ctrl2model($account,
'id_number'),
        'studentnumber' => $this->_format_array_ctrl2model($account,
'studno'),
        'ou' => $this->_format_array_ctrl2model($account, 'unit'),
        'employeetype' => $this->_format_array_ctrl2model($account,
'type'),
        'title' => $this->_format_array_ctrl2model($account,
'designation'),
        'mailquotasize' => $this-
>_format_array_ctrl2model($account,'mailquotasize'),
        'departmentnumber' => $this-
>_format_array_ctrl2model($account, 'division'),
        'description' => $this->_format_array_ctrl2model($account,
'description')
    );

    $ldap_data["objectclass"][0] = "top";
    $ldap_data["objectclass"][1] = "person";
    $ldap_data["objectclass"][2] = "inetOrgPerson";
    $ldap_data["objectclass"][3] = "qmailUser";
    $ldap_data["objectclass"][4] = "radiusprofile";

    $ldap_data["dialupAccess"] = 'TRUE';

    $ldap_data["mailHost"] = 'imap4.upd.edu.ph';
    $ldap_data["mailAlternateAddress"][0] = $username."@upd.edu.ph";
    $ldap_data['mailMessageStore'] =
'upd.edu.ph/' . $ldap_data['ou'] . '/' . $ldap_data['departmentnumber'] . '/' . $username;

    # generate password
    $this->load->helper('string');
    $password = random_password();
    mt_srand((double)microtime()*1000000);
    $salt = pack("CCCC", mt_rand(), mt_rand(), mt_rand(), mt_rand());
    $ldap_data["userPassword"] = "{SSHA}" . base64_encode(pack("H*",
sha1($password . $salt)) . $salt);

    # Callback function for array_filter
    function _is_not_null ($var) {
        if(!is_null($var) || $var != NULL || $var != '') { return
true; } else { return false; }
    }

    #Remove keys of NULL and False values
    $ldap_data = array_filter( $ldap_data, '_is_not_null' );
    // print_r($ldap_data);

    # Insert to ldap

    $result = $ldapresult = $this->ldap_model->add_ldapentry($username,
$ldap_data);

    if ($result) {

```



```

        $this->log_model->create($username, "Account added to LDAP",
"activation_special", $this->tech_user, $this->ip, 'info');

```

```

        # Insert to mysql
        #TODO Add Unit, Course, Dept
        $mysql_data = array(
            'username' => $username,
            'email' => $email_only,
            'activated_by'=>$_SESSION['tech_user'],
            'ip_activated'=>$this->input->ip_address(),
            'activated' => 1,
            'firstname' => $this-
>_format_array_ctrl2model($account, 'firstname'),
            'lastname' => $this-
>_format_array_ctrl2model($account, 'lastname'),
            'middlename' => $this-
>_format_array_ctrl2model($account, 'middlename'),
            'id_number' => $this-
>_format_array_ctrl2model($account, 'id_number'),
            'empno' => $this-
>_format_array_ctrl2model($account, 'id_number'),
            'studno' => $this-
>_format_array_ctrl2model($account, 'studno'),
            'designation' => $this-
>_format_array_ctrl2model($account, 'designation'),
        );

```

```

        #Remove keys with NULL and FALSE values
        $mysql_data = array_filter( $mysql_data, '_is_not_null' );

```

```

        # Send Welcome Email
        $welcome_data['user'] = array(
            'email' => $ldap_data['mail'],
            'firstname' => $mysql_data['firstname'],
            'username' => $username
        );

```

```

        $this->_send_welcome_mail($welcome_data);
        $this->log_model->create($username, "Sending welcome email",
"activation_special", $this->tech_user, $this->ip, 'debug');

```

```

        #TODO Check correct return values
        if($this->mysql_model->create($username, $mysql_data)){
            $this->log_model->create($username, "Account added to
MySQL", "activation_special", $this->tech_user, $this->ip, 'info');

```

```

        # Prepare details for printing, forward to printing
function
    $_SESSION['create_account_print']['username'] =
$username;
    $_SESSION['create_account_print']['password'] =
$password;
    $this->session->mark_as_flash('create_account_print');

    redirect('tech/print_account/confirm_print/'.$username);
    }
    }
}

```

```

function _check_username_special($str){
    if ($str != NULL){
        // check if the supplied username is an email
        if(filter_var($str, FILTER_VALIDATE_EMAIL)){
            $this->form_validation->set_message('_check_username_special',
'Username can\'t be of email form.');
```

return FALSE;

}

#Check MySQL existence

\$temp = \$this->mysql_model->search_username(\$str);

if(!empty(\$temp)){

 \$this->form_validation->set_message('_check_username_special',

'This username ('. \$str. ') exists in the MySQL database');

 return FALSE;

}

#Check LDAP existence

\$temp = \$this->ldap_model->search_exact(\$str);

if(\$temp){

 \$this->form_validation->set_message('_check_username_special',

'This username ('. \$str. ') exists in the LDAP database');

 return FALSE;

}

return TRUE;

}

else {

 \$this->form_validation->set_message('_check_username_special',

'Please provide a valid username.');

 return FALSE;

}

}

function _check_email_special(\$str){

 if (\$str != NULL){

 if(filter_var(\$str, FILTER_VALIDATE_EMAIL))

 {

 list(\$email_only, \$domain_only) = explode("@", \$str);

 if(strtolower(\$domain_only) != "upd.edu.ph")

 {

 \$this->form_validation->

>set_message('_check_email_special', 'Only @upd.edu.ph domain is allowed. Please

provide another one.');

 return FALSE;

 }

 } else

 {

 \$email_only_str = \$email_only;

 }

 } else

 {

 \$email_only_str = \$str;

 }

 #Check MySQL existence

```

        $temp = $this->mysql_model->search_email($email_only_str);
        if(!empty($temp)){
            $this->form_validation->set_message('_check_email_special',
'The email ('. $str. ') exists in the MySQL database');
            return FALSE;
        }

        #Check LDAP existence
        $temp = $this->ldap_model->search_exact($email_only_str);
        if($temp){
            $this->form_validation->set_message('_check_email_special',
'The email ('. $str. ') exists in the LDAP database');
            return FALSE;
        }

        return TRUE;
    }

    return FALSE;
}

function _check_blank_lastname($str){
    $ln_none = $this->input->post('ln_none');

    if ($ln_none && $str != NULL){
        $this->form_validation->set_message('_check_blank_lastname',
'<em>Last Name field</em> has a value but <em>No Last Name</em> is checked. ');
        return FALSE;
    }
    if (!$ln_none && $str == NULL){
        $this->form_validation->set_message('_check_blank_lastname', ' <em>No
Last Name</em> is unchecked but <em>Last Name</em> field has no value. ');
        return FALSE;
    }
    return TRUE;
}

function _check_blank_middlename($str){
    $mn_none = $this->input->post('mn_none');

    if ($mn_none && $str != NULL){
        $this->form_validation->set_message('_check_blank_middlename',
'<em>Middle Name field</em> has a value but <em>No Middle Name</em> is checked. ');
        return FALSE;
    }
    if (!$mn_none && $str == NULL){
        $this->form_validation->set_message('_check_blank_middlename',
'<em>No Middle Name</em> is unchecked but <em>Middle Name</em> field has no
value. ');
        return FALSE;
    }
    return TRUE;
}
}

```



```

        $this->form_validation->set_rules('unit', 'Unit',
'required|alpha_dash');
        $this->form_validation->set_rules('gender', 'Gender', 'required');
        $this->form_validation->set_rules('division', 'Division',
'required|alpha_dash');
        $this->form_validation->set_rules('designation', 'Designation',
'required');
        $this->form_validation->set_rules('type', 'Account Type',
'required');

        if ($this->form_validation->run() == TRUE) {
            $account_form = $this->input->post(NULL, TRUE);

            $_SESSION['create_user'] = $account_form;
            $this->session->mark_as_flash('create_user');

            redirect('tech/create_user/review');
        }

        if(isset($_SESSION['create_user_back'])){
            $_POST = $_SESSION['create_user_back'];
            unset($_SESSION['create_user_back']);
        }

        $data = $this->page_vars;
        $data['desig_list'] = $this->mysql_model->get_designation_list();
        $data['unit_list'] = $this->mysql_model->get_unit_list();
        $data['division_list'] = $this->mysql_model->get_division_list();

        $this->load->vars($data);
        $this->load->view('page_template');
    }

    ##CREATE USER :: REVIEW
    public function review() {

        if(isset($_SESSION['create_user'])){

            $data = $this->page_vars;

            $data['content'] = 'account_user_create_confirm';
            $data['account'] = $account_form = $_SESSION['create_user'];

            # Get name
            $data['account']['firstname'] =
isset($account_form['firstname']) ? ucwords($account_form['firstname']) : NULL;
            $data['account']['middlename'] =
isset($account_form['middlename']) ? ucwords($account_form['middlename']) : NULL;
            $data['account']['lastname'] =
isset($account_form['lastname']) ? ucwords($account_form['lastname']) : NULL;

            # Begin processing available email and username
            $data['account_gen']['username'] = $this-
>generate_username($data['account']['firstname'], $data['account']['middlename'],
$data['account']['lastname']);

            #TODO Check this username against CRS data

```

```

        $data['account_gen']['account_email'] = $this->
_generate_user_email($data['account']['firstname'],
$data['account']['middlename'], $data['account']['lastname']);

        $data['account_gen']['account_email'] .= '@upd.edu.ph';

        # 'Account_fieldnames'
        // Add pretty output for input field names here.
        // If key is not found in this array, then it will display the
default field name
        $data['account_fieldnames'] = array(
            'type' => 'Account Type',
            'firstname' => 'First Name',
            'middlename' => 'Middle Name',
            'lastname' => 'Last Name',
            'birthdate' => 'Birthdate',
            'unit' => 'Unit / College',
            'division' => 'Division / Department',
            'gender' => 'Gender',
            'id_number' => 'Employee Number',
            'studentno' => 'Student Number',
            'designation' => 'Designation',
            'description' => 'Appointment Period',
            'mailquotasize' => 'Mail Quota Size',
            'bypass' => 'Special Case',
            'mn_none' => 'No Middle Name',
            'ln_none' => 'No Last Name'
        );

        $data['account_gen_fieldnames'] = array(
            'username' => 'Username',
            'account_email' => 'Email'
        );

        // Change session key for next and previous step (so that the
data will be destroyed if "review" page is refreshed)
        $_SESSION['create_user_back'] =
$_SESSION['create_user_confirm'] = $account_form;

        $this->load->vars($data);
        $this->load->view('page_template');
    }

    else {
        // Store in session for the final step
        $this->session->mark_as_flash('create_user_confirm');
        $this->session->mark_as_flash('create_user_back');

        if(!isset($_SESSION['create_user_back'])){
            $_SESSION['error'] = "The page was refreshed so the
form data was lost. Kindly fill up the form again.";
            $this->session->mark_as_flash('error');
        }

        redirect('tech/create_user');
    }
}

```

```

//CREATE USER :: FINAL STEP FOR ACCOUNT CREATION
public function activate(){
    #TODO catch isset
    if(isset($_SESSION['create_user_confirm'])){
        $account = $_SESSION['create_user_confirm'];
    }

    else {
        $_SESSION['warning'] = "Session data lost. Redirecting you to
this page instead.";
        $this->session->mark_as_flash('warning');
        redirect('tech/create_user');
    }
    // print_r($account);

    # TODO Optimize
    $first= isset($account['firstname']) ?
ucwords($account['firstname']) : NULL;
    $middle = isset($account['middlename']) ?
ucwords($account['middlename']) : NULL;
    $last = isset($account['lastname']) ? ucwords($account['lastname'])
: NULL;

    # Begin processing available email and username
    $username = $this->_generate_username($first, $middle, $last);
    $account_email = $this->_generate_user_email($first, $middle,
$last);

    # Check username against CRS data
    $crs_check = $this->crs_model->_getstudentnobyusername($username);

    if(isset($crs_check) && !$crs_check['errcode'] &&
!array_key_exists("bypass", $account)){
        #TODO Add link to duplicate user
        $_SESSION['create_user_back'] =
$_SESSION['create_user_confirm'];
        $_SESSION["warning"] = "Username that will be generated exists
in the CRS database with student number:
<strong>".$crs_check['studentno']. "</strong><br />Double check with CRS. If there
is no problem with this account, check bypass to proceed";
        $this->session->mark_as_flash('warning');
        $this->session->mark_as_flash('create_user_back');
        redirect('tech/create_user');
    }

    if($this->_is_special_case($first, $middle, $last) &&
!array_key_exists("bypass", $account)){
        #TODO Add link to duplicate user
        $_SESSION['create_user_back'] =
$_SESSION['create_user_confirm'];
        $_SESSION["warning"] = "Special case detected (fullname match
or double 'none'). <br/> Check bypass to proceed.";
        $this->session->mark_as_flash('warning');
        $this->session->mark_as_flash('create_user_back');
        redirect('tech/create_user');
    }

    #TODO use name (first,middle,last) array

```

```

        else if($this->_is_duplicate($account) &&
!array_key_exists("bypass", $account)){
            #TODO Add links to duplicate users
            $_SESSION['create_user_back'] =
$_SESSION['create_user_confirm'];
            $_SESSION["warning"] = "Duplicate student/employee number
detected. <br/> Check bypass if you want to proceed.";
            $this->session->mark_as_flash('warning');
            $this->session->mark_as_flash('create_user_back');
            redirect('tech/create_user');

        } else {

            #TODO Add isset $username, $account_email
            $ldap_data = array(
                'uid' => $username,
                'cn' => $first." ".$last,
                'sn' => $this->_format_array_ctrl2model($account,
'lastname'),
                'mail' => $account_email."@upd.edu.ph",
                'employeenumber' => $this-
>_format_array_ctrl2model($account, 'id_number'),
                'studentnumber' => $this-
>_format_array_ctrl2model($account, 'studentno'),
                'ou' => strtoupper($this-
>_format_array_ctrl2model($account, 'unit')),
                'employeetype' => $this-
>_format_array_ctrl2model($account, 'type'),
                'title' => $this->_format_array_ctrl2model($account,
'designation'),
                'mailquotasize' => $this-
>_format_array_ctrl2model($account, 'mailquotasize'),
                'departmentnumber' => strtoupper($this-
>_format_array_ctrl2model($account, 'division')),
                'description' => $this-
>_format_array_ctrl2model($account, 'description')
            );

            if(empty($ldap_data['sn'])){
                $ldap_data['sn'] = "NO_LAST_NAME";
            }

            $ldap_data["objectclass"][0] = "top";
            $ldap_data["objectclass"][1] = "person";
            $ldap_data["objectclass"][2] = "inetOrgPerson";
            $ldap_data["objectclass"][3] = "qmailUser";
            $ldap_data["objectclass"][4] = "radiusprofile";

            $ldap_data["dialupAccess"] = 'TRUE';

            $ldap_data["mailHost"] = 'imap4.upd.edu.ph';
            $ldap_data["mailAlternateAddress"][0] =
$username."@upd.edu.ph";
            $ldap_data['mailMessageStore'] =
'upd.edu.ph/'. $ldap_data['ou']. '/' . $ldap_data['departmentnumber']. '/' . $username;

            # generate password
            $this->load->helper('string');
            $password = random_password();

```



```

        mt_srand((double)microtime()*1000000);
        $salt = pack("CCCC", mt_rand(), mt_rand(), mt_rand(),
mt_rand());
        $ldap_data["userPassword"] = "{SSHA}" .
base64_encode(pack("H*", sha1($password . $salt)) . $salt);

        # Callback function for array_filter
        function _is_not_null ($var) {
            if(!is_null($var) || $var != NULL || $var != '') {
return true; } else { return false; }
        }

        #Remove keys of NULL and False values
        $ldap_data = array_filter( $ldap_data, '_is_not_null' );

        # Insert to ldap
        $result = $ldapresult = $this->ldap_model-
>add_ldapentry($username, $ldap_data);

        if ($result) {

            $this->log_model->create($username, "Account added to
LDAP", "activation", $this->tech_user, $this->ip, 'info');

            # Insert to mysql
            #TODO Add Unit, Course, Dept
            $mysql_data = array(
                'username' => $username,
                'email' => $account_email,
                'activated_by'=>$_SESSION['tech_user'],
                'ip_activated'=>$this->input->ip_address(),
                'activated' => 1,
                'firstname' => $this-
>_format_array_ctrl2model($account, 'firstname'),
                'lastname' => $this-
>_format_array_ctrl2model($account, 'lastname'),
                'middlename' => $this-
>_format_array_ctrl2model($account, 'middlename'),
                'id_number' => $this-
>_format_array_ctrl2model($account, 'id_number'),
                'empno' => $this-
>_format_array_ctrl2model($account, 'id_number'),
                'studno' => $this-
>_format_array_ctrl2model($account, 'studentno'),
                'birthday' => $this-
>_format_array_ctrl2model($account, 'birthdate'),
                'gender' => $this-
>_format_array_ctrl2model($account, 'gender'),
                'designation' => $this-
>_format_array_ctrl2model($account, 'designation'),
            );

            # Remove keys with NULL and FALSE values
            $mysql_data = array_filter( $mysql_data, '_is_not_null'
);

            # Send Welcome Email
            $welcome_data['user'] = array(
                'email' => $ldap_data['mail'],

```

```

        'firstname' => $mysql_data['firstname'],
        'username' => $username
    );

    $this->_send_welcome_mail($welcome_data);
    $this->log_model->create($username, "Sending welcome
email", "activation", $this->tech_user, $this->ip, 'debug');

    #TODO Check correct return values
    if($this->mysql_model->create($username, $mysql_data)){
        $this->log_model->create($username, "Account
added to MySQL", "activation", $this->tech_user, $this->ip, 'info');

        # Prepare details for printing, forward to
printing function
        $_SESSION['create_account_print']['username'] =
$username;
        $_SESSION['create_account_print']['password'] =
$password;
        $this->session-
>mark_as_flash('create_account_print');

        redirect('tech/print_account/confirm_print/' . $username);
    }
}

function _check_blank_lastname($str){
    $ln_none = $this->input->post('ln_none');

    if ($ln_none && $str != NULL){
        $this->form_validation->set_message('_check_blank_lastname',
'<em>Last Name field</em> has a value but <em>No Last Name</em> is checked.');
```

'Last Name field has a value but No Last Name is checked.');

```

        return FALSE;
    }
    if (!$ln_none && $str == NULL){
        $this->form_validation->set_message('_check_blank_lastname',
'<em>No Last Name</em> is unchecked but <em>Last Name</em> field has no value.');
```

'No Last Name is unchecked but Last Name field has no value.');

```

        return FALSE;
    }
    return TRUE;
}

function _check_blank_middlename($str){
    $mn_none = $this->input->post('mn_none');

    if ($mn_none && $str != NULL){
        $this->form_validation->set_message('_check_blank_middlename',
'<em>Middle Name field</em> has a value but <em>No Middle Name</em> is checked.');
```

'Middle Name field has a value but No Middle Name is checked.');

```

        return FALSE;
    }
    if (!$mn_none && $str == NULL){
        $this->form_validation->set_message('_check_blank_middlename',
'<em>No Middle Name</em> is unchecked but <em>Middle Name</em> field has no
value.');
```

'No Middle Name is unchecked but Middle Name field has no value.');

```

        return FALSE;
    }
}

```

```
        return TRUE;
```

```
    }
```

```
}
```

```
application/controllers/tech/Dashboard.php
```

```
<?php
```

```
defined('BASEPATH') OR exit('No direct script access allowed');
```

```
class Dashboard extends MY_Controller {
```

```
    /**
```

```
     * Index Page for this controller.
```

```
     *
```

```
     * Maps to the following URL
```

```
     *      http://example.com/index.php/welcome
```

```
     *      - or -
```

```
     *      http://example.com/index.php/welcome/index
```

```
     *      - or -
```

```
     * Since this controller is set as the default controller in
```

```
     * config/routes.php, it's displayed at http://example.com/
```

```
     *
```

```
     * So any other public methods not prefixed with an underscore will
```

```
     * map to /index.php/welcome/<method_name>
```

```
     * @see https://codeigniter.com/user_guide/general/urls.html
```

```
    */
```

```
function __construct(){
```

```
    parent::__construct();
```

```
    $this->load->model('mysql_model');
```

```
}
```

```
public function index()
```

```
{
```

```
    $data = array(
```

```
        'content' => 'dashboard',
```

```
        'page_title' => 'Accounts Dashboard',
```

```
        'page_subtitle' => 'Overview and general statistics of DilNet
```

```
accounts',
```

```
        'include_stylesheets' => 'dashboard_css',
```

```
        'include_scripts' => 'dashboard_js'
```

```
    );
```

```
    $prev_year = date("Y",strtotime("-1 year"));
```

```
    $curr_year = date('Y');
```

```
    $curr_month = date('m');
```

```
    $prev_month = date('m',strtotime("-1 month"));
```

```
    /** NEW ACCOUNTS **/
```

```
    $curryear = $this->mysql_model-
```

```
>get_new_accounts_by_year($curr_year);
```

```

        $prevyear = $this->mysql_model-
>get_new_accounts_by_year($prev_year);

        $currmonth_curryear = $this->mysql_model-
>get_new_accounts_by_month($curr_year,$curr_month);
        $currmonth_prevyear = $this->mysql_model-
>get_new_accounts_by_month($prev_year,$curr_month);

        $prevmonth_curryear = $this->mysql_model-
>get_new_accounts_by_month($curr_year,$prev_month);
        $prevmonth_prevyear = $this->mysql_model-
>get_new_accounts_by_month($prev_year,$prev_month);

        $currmonth_curryear_reset = $this->mysql_model-
>get_reset_passwords_by_staff_month($curr_year,$curr_month);
        $currmonth_prevyear_reset = $this->mysql_model-
>get_reset_passwords_by_staff_month($prev_year,$curr_month);

        $prevmonth_curryear_reset = $this->mysql_model-
>get_reset_passwords_by_staff_month($curr_year,$prev_month);
        $prevmonth_prevyear_reset = $this->mysql_model-
>get_reset_passwords_by_staff_month($prev_year,$prev_month);

        if ($currmonth_prevyear == 0) { $currmonth_prevyear = 1; }
        if ($prevmonth_prevyear == 0) { $prevmonth_prevyear = 1; }
        if ($currmonth_prevyear_reset == 0) { $currmonth_prevyear_reset = 1;
}
        if ($prevmonth_prevyear_reset == 0) { $prevmonth_prevyear_reset = 1;
}

        $currmonth_rateval = (($currmonth_curryear/$currmonth_prevyear) -
1);
        $currmonth_rateval_class = $this-
>add_percentage_class($currmonth_rateval);
        $currmonth_rateval = round(abs($currmonth_rateval)*100);
        $currmonth_rateval_str =
$currmonth_rateval_class.$currmonth_rateval.'% </span>';

        $prevmonth_rateval = (($prevmonth_curryear/$prevmonth_prevyear) -
1);
        $prevmonth_rateval_class = $this-
>add_percentage_class($prevmonth_rateval);
        $prevmonth_rateval = round(abs($prevmonth_rateval)*100);
        $prevmonth_rateval_str =
$prevmonth_rateval_class.$prevmonth_rateval.'% </span>';

        $currmonth_rateval_reset =
(($currmonth_curryear_reset/$currmonth_prevyear_reset) - 1);
        $currmonth_rateval_reset_class = $this-
>add_percentage_class($currmonth_rateval_reset);
        $currmonth_rateval_reset = round(abs($currmonth_rateval_reset)*100);
        $currmonth_rateval_reset_str =
$currmonth_rateval_reset_class.$currmonth_rateval_reset.'% </span>';

        $prevmonth_rateval_reset =
(($prevmonth_curryear_reset/$prevmonth_prevyear_reset) - 1);
        $prevmonth_rateval_reset_class = $this-
>add_percentage_class($prevmonth_rateval_reset);
        $prevmonth_rateval_reset = round(abs($prevmonth_rateval_reset)*100);

```

```

        $prevmonth_rateval_reset_str =
$prevmonth_rateval_reset_class.$prevmonth_rateval_reset.'% </span>';

        $graph_prevyear = array();
        $graph_curryear = array();

        for($i = 1; $i <= 12; $i++){
            $graph_prevyear[$i] = $this->mysql_model-
>get_new_accounts_by_month($prev_year,$i);
        }

        for($i = 1; $i <= 12; $i++){
            $graph_curryear[$i] = $this->mysql_model-
>get_new_accounts_by_month($curr_year,$i);
        }

        $graph_prevyear_reset = array();
        $graph_curryear_reset = array();
        for($i = 1; $i <= 12; $i++){
            $graph_prevyear_reset[$i] = $this->mysql_model-
>get_reset_passwords_by_staff_month($prev_year,$i);
        }

        for($i = 1; $i <= 12; $i++){
            $graph_curryear_reset[$i] = $this->mysql_model-
>get_reset_passwords_by_staff_month($curr_year,$i);
        }

        $graph_prevyear_lp = array();
        $graph_curryear_lp = array();
        for($i = 1; $i <= 12; $i++){
            $graph_prevyear_lp[$i] = $this->mysql_model-
>get_lostpass_by_month($prev_year,$i);
        }

        for($i = 1; $i <= 12; $i++){
            $graph_curryear_lp[$i] = $this->mysql_model-
>get_lostpass_by_month($curr_year,$i);
        }

        #TODO Optimize
        $graph_tech_stats = array();
        $tech_staff_stats = $this->mysql_model-
>get_tech_staff_stats($curr_year);
        $colors =
array("#f56954", "#00a65a", "#f39c12", "#00c0ef", "#3c8dbc", "#d2d6de", "#f39c12", "#00c0
ef", "#3c8dbc", "#d2d6de");
        for($i = 0; $i < count($tech_staff_stats); $i++){
            if($tech_staff_stats[$i]['changed_by'] == NULL)
            { $tech_staff_stats[$i]['changed_by'] = 'Old data'; }
            $graph_tech_stats[$i] = array(
                'label' => $tech_staff_stats[$i]['changed_by'],
                'value' =>
intval($tech_staff_stats[$i]['total']),
                'color' => $colors[$i],
                'highlight' => $colors[$i]
            );
        }
        $graph_tech_stats_datastr = json_encode($graph_tech_stats);

```

```
##### END TODO #####
```

```
#TODO Optimize
```

```
$graph_acctype_stats = array();
$acctype_stats = $this->mysql_model->get_acctype_stats($curr_year);
for($i = 0; $i < count($acctype_stats); $i++){
    if($acctype_stats[$i]['type'] == NULL)
    { $acctype_stats[$i]['type'] = 'STUD'; }
    $graph_acctype_stats[$i] = array(
        'label' => $acctype_stats[$i]['type'],
        'value' => intval($acctype_stats[$i]['total']),
        'color' => $colors[$i],
        'highlight' => $colors[$i]
    );
}
$graph_acctype_datastr = json_encode($graph_acctype_stats);
##### END TODO #####
```

```
$graph_prevyear_datastr =
json_encode(array_values($graph_prevyear));
$graph_curryear_datastr =
json_encode(array_values($graph_curryear));
```

```
$graph_prevyear_datastr_reset =
json_encode(array_values($graph_prevyear_reset));
$graph_curryear_datastr_reset =
json_encode(array_values($graph_curryear_reset));
```

```
$graph_prevyear_datastr_lp =
json_encode(array_values($graph_prevyear_lp));
$graph_curryear_datastr_lp =
json_encode(array_values($graph_curryear_lp));
```

```
$statistics = array(
    'currmonth_rateval' => $currmonth_rateval_str,
    'prevmonth_rateval' => $prevmonth_rateval_str,
    'currmonth_rateval_reset' => $currmonth_rateval_reset_str,
    'prevmonth_rateval_reset' => $prevmonth_rateval_reset_str,
    'accounts_currmonth' => $currmonth_curryear,
    'accounts_prevmonth' => $prevmonth_curryear,
    'reset_currmonth' => $currmonth_curryear_reset,
    'reset_prevmonth' => $prevmonth_curryear_reset,
    'graph_prevyear_datastr' => $graph_prevyear_datastr,
    'graph_curryear_datastr' => $graph_curryear_datastr,
    'graph_prevyear_datastr_reset' =>
$graph_prevyear_datastr_reset,
    'graph_curryear_datastr_reset' =>
$graph_curryear_datastr_reset,
    'graph_prevyear_datastr_lp' => $graph_prevyear_datastr_lp,
    'graph_curryear_datastr_lp' => $graph_curryear_datastr_lp,
    'graph_prevyear_legend' => $prev_year,
    'graph_curryear_legend' => $curr_year,
    'pie_tech_data' => $graph_tech_stats_datastr,
    'pie_acctype_data' => $graph_acctype_datastr
);
$data['stats'] = $statistics;

$this->load->vars($data);
$this->load->view('page_template');
```

```

    }

    #TODO Improve please
    public function add_percentage_class($value = NULL){
        if (isset($value)){
            if($value < 0){ $caret = '<span class="description-percentage
text-red"><i class="fa fa-caret-down"></i> '; }
            else if ($value == 0){ $caret = '<span class="description-
percentage text-yellow"><i class="fa fa-caret-left"></i> '; }
            else if ($value > 0){ $caret = '<span class="description-
percentage text-green"><i class="fa fa-caret-up"></i> '; }
            else $value = '<span class="description-percentage">';

            return $caret;
        }

        else return FALSE;
    }
}

```

application/controllers/tech/Dilnet_lookup.php

```

<?php
defined('BASEPATH') OR exit('No direct script access allowed');

class Dilnet_lookup extends MY_Controller {

    function __construct(){
        parent::__construct();

        $this->form_validation->set_error_delimiters('<label>', '</label>');
        $this->load->model('ldap_model');
        $this->load->model('mysql_model');
        $this->load->model('proxy_tools_model');
        $this->load->model('mysql_webmail_model');
        $this->load->model('mail_tool_model');
        $this->load->model('log_model');

        $this->include_ldap = FALSE;
        $this->tech_user = $_SESSION['tech_user'];
        $this->ip = $this->input->ip_address();

        $this->page_vars = array(
            'content' => 'dilnet_lookup',
            'page_title' => 'DilNet Account Lookup',
            'page_subtitle' => 'Search DilNet Account Information',
            'include_stylesheets' => 'dilnet_lookup_css',
            'include_scripts' => 'dilnet_lookup_js'
        );
    }

    public function index()
    {
        // Load page variables
    }
}

```

```

        $data = $this->page_vars;

        $this->load->vars($data);
        $this->load->view('page_template');
    }

    public function name_search_form($new=NULL) {
        // Load page variables
        $data = $this->page_vars;

        if($new=="new"){

            if(isset($_SESSION['results_db'])){
                unset($_SESSION['results_db']);
            }
            if(isset($_SESSION['search_string'])){
                unset($_SESSION['search_string']);
            }
        }

        $temp_name_str = $this->input->post('name_str');

        if(!empty($_SESSION['results_db']) &&
!empty($_SESSION['search_string']) && empty($temp_name_str)){

            $data['results_db'] = $_SESSION['results_db'];
            $data['search_str'] = $_SESSION['search_string'];

        } else {
            $this->form_validation->set_rules('name_str', 'Search',
'required|trim|xss_clean|min_length[3]');

            if ($this->form_validation->run() == TRUE){

                if(isset($_SESSION['results_db'])){
                    unset($_SESSION['results_db']);
                }

                if(isset($_SESSION['search_string'])){
                    unset($_SESSION['search_string']);
                }

                $search_string = $this->input->post('name_str');
                #TODO: Sanitize search string
                if($search_string != "" || $search_string != NULL) {
                    $stringArray = explode(" ", $search_string);
                    $count_query = count($stringArray);
                    $data["search_str"] = $search_string;

                    if(1==$count_query){ // only one keyword
                        if(strpos($search_string,"@upd.edu.ph")){ //
might be a primary email
                                // email match
                                $pos =
strpos($search_string,"@upd.edu.ph");
                                $email = substr($search_string, 0, $pos);
                                if(strpos($email,".")){ // might be a
Primary email

```



```

// $data["results_db"] = $this->mysql_model->search_email($email);
// $temp = $this->ldap_model->search_mail_addresses($email);
// $data["results_db"] = $this->mysql_model->search_username($temp[0]["uid"][0]);
} else {
// username match
// $data["results_db"] = $this->mysql_model->search_username($email);
// $temp = $this->ldap_model->search_mail_addresses($email);
// $data["results_db"] = $this->mysql_model->search_username($temp[0]["uid"][0]);
}

// primary email match
else if(strpos($search_string, ".")){ // might be
a primary email
// $data["results_db"] = $this->mysql_model->search_email($search_string);

if(filter_var($search_string, FILTER_VALIDATE_EMAIL)){
$temp = $this->ldap_model->search_mail_addresses($search_string);
$data["results_db"] = $this->mysql_model->search_username($temp[0]["uid"][0]);
} else {
// let's try adding @upd.edu.ph
$expanded =
$search_string."@upd.edu.ph";
$temp = $this->ldap_model->search_mail_addresses($expanded);
$data["results_db"] = $this->mysql_model->search_username($temp[0]["uid"][0]);

if(empty($data['results_db'])){
$data["results_db"] = $this->mysql_model->search_substring($search_string);
}

}

}

// exact student number match
else if(strpos($search_string, "-")){ // might be
a student number
if($this->_is_idno_studentno($search_string)){
$search_result_sn = $this->mysql_model->search_substring_number($search_string);
if(empty($search_result_sn)){ //
might be stored in LDAP

```

```

                                $temp = $this->ldap_model-
>search_substring_number($search_string);
                                $search_result_sn = $this-
>mysql_model->search_username($temp[0]["uid"][0]);
                                }
                                $data["results_db"] =
$search_result_sn;
                                } else { // might be partial
                                $search_result_sn_partial = $this-
>mysql_model->search_substring_number($search_string);
                                if(!empty($search_result_sn_partial)){
                                $data["results_db"] =
$search_result_sn_partial;
                                }
                                }
                                }
                                else if(is_numeric($search_string)){
                                // employee number match
                                $len = strlen($search_string);
                                if($len<=9){
                                $exact = FALSE;
                                } else if($len==10){
                                $exact = TRUE;
                                }

                                // check MySQL for the employee number
                                $search_result = $this->mysql_model-
>search_substring_number($search_string,$exact);

                                // it's possible that what's stored in
MySQL is less than 10 chars
                                if(empty($search_result)){
                                if($len==10){ // maybe the value
stored is 9 digits or lower
                                $temp_search_string =
substr($search_string, 1);
                                $exact = FALSE;
                                $search_result = $this-
>mysql_model->search_substring_number($temp_search_string,$exact);
                                }
                                }
                                /**/

                                // it's possible that it's stored only in
LDAP
                                if(empty($search_result)){
                                $temp = $this->ldap_model-
>search_substring_number($search_string);
                                if(empty($temp)){
                                if($len==10){
                                $temp_search_string =
substr($search_string, 1);
                                $temp = $this-
>ldap_model->search_substring_number($temp_search_string);
                                $search_result = $this-
>mysql_model->search_username($temp[0]["uid"][0]);
                                }

```

```

        } else {
            $search_result = $this->mysql_model->search_username($temp[0]["uid"][0]);
        }
    }

    // print "still empty";
    // still empty, might be a student number

without the dash
    if(empty($search_result)){
        if($this->_is_idno_studentno($search_string)){
            $temp_student_number =
            "".substr($search_string,0, 4)."-".substr($search_string, 4, 9)."";

            $search_result_sn = $this->mysql_model->search_substring_number($temp_student_number);
            if(empty($search_result_sn)){
                // might be stored in LDAP
                $temp = $this->ldap_model->search_substring_number($temp_student_number);
                $search_result_sn =
                $this->mysql_model->search_username($temp[0]["uid"][0]);
            }
            $data["results_db"] =
            $search_result_sn;
        }
        } else {
            $data["results_db"] =
            $search_result;
        }

        // print_r($data['results_db']);
    }
    else {
        // try first a username search
        // $data["results_db"] = $this->mysql_model->search_username($search_string);
        $temp = $this->ldap_model->search_mail_addresses($search_string);
        $data["results_db"] = $this->mysql_model->search_username($temp[0]["uid"][0]);
        if(empty($data["results_db"])){
            $data["results_db"] = $this->mysql_model->search_substring($search_string);
        }
    }
} else if($count_query>1){ // more than one

search keywords
    // try connecting them first
    $connected = implode("_", $stringArray);
    $data["results_db"] = $this->mysql_model->search_substring($connected);
    if(empty($data['results_db'])){
        $data["results_db"] = $this->mysql_model->search_substring($search_string);
    }
}

```

```

        }
    }

    // if(empty($data['results_db'])){ // let's try one
more time, ldap search
        // $temp = $this->ldap_model-
>search_mail_addresses($email);
        // $data["results_db"] = $this->mysql_model-
>search_username($temp[0]["uid"][0]);
        // }

        // if($this->include_ldap) {
        // // $data["results_ldap"] = $this->ldap_model-
>search_substring($search_string);
        // $data["results_ldap"] = $this->ldap_model-
>search_mail_addresses($search_string);
        // }

    }

    if(isset($data['results_db']) && !empty($data['results_db'])){

        $_SESSION['results_db'] = $data['results_db'];
        $_SESSION['search_string'] = $data["search_str"];
    }

}

    // print "klasjdlfjasdfjkasfjdfkaskf Count:
".count($data['results_db']);

    if(isset($data['results_db']) && count($data['results_db']) == 1 ) {
// exact match
        $this->session->set_flashdata('info', "Found exact match for
'".$search_string."'");
        $username = $data["results_db"][0]['username'];
        redirect('tech/dilnetacc_tools/view_account/'.$username.'');
    }
    $this->load->vars($data);
    $this->load->view('page_template');
}

public function navbar_search_form($new=NULL) {
    // Load page variables
    $data = $this->page_vars;

    if($new=="new"){

        if(isset($_SESSION['results_db'])){
            unset($_SESSION['results_db']);
        }
        if(isset($_SESSION['search_string'])){
            unset($_SESSION['search_string']);
        }
    }

    $temp_navbar_search_str = $this->input->post('navbar_search_str');

```

```

        if(!empty($_SESSION['results_db']) &&
!empty($_SESSION['search_string']) && empty($temp_navbar_search_str)){

            $data['results_db'] = $_SESSION['results_db'];
            $data['search_str'] = $_SESSION['search_string'];

        } else {
            $this->form_validation->set_rules('navbar_search_str',
'Search', 'required|xss_clean|min_length[3]');

            if ($this->form_validation->run() == TRUE){

                if(isset($_SESSION['results_db'])){
                    unset($_SESSION['results_db']);
                }

                if(isset($_SESSION['search_string'])){
                    unset($_SESSION['search_string']);
                }

                $search_string = $this->input-
>post('navbar_search_str');
                #TODO: Sanitize search string
                if($search_string != "" || $search_string != NULL) {
                    $stringArray = explode(" ", $search_string);
                    $count_query = count($stringArray);
                    $data["search_str"] = $search_string;

                    if(1==$count_query){ // only one keyword
                        if(strpos($search_string,"@upd.edu.ph")){ //
might be a primary email
                                // email match
                                $pos =
strpos($search_string,"@upd.edu.ph");
                                $email = substr($search_string, 0, $pos);
                                if(strpos($email,".")){ // might be a
Primary email
                                        // $data["results_db"] = $this-
>mysql_model->search_email($email);
                                        $temp = $this->ldap_model-
>search_mail_addresses($email);
                                        $data["results_db"] = $this-
>mysql_model->search_username($temp[0]["uid"][0]);
                                        } else {
                                        // username match
                                        // $data["results_db"] = $this-
>mysql_model->search_username($email);
                                        $temp = $this->ldap_model-
>search_mail_addresses($email);
                                        $data["results_db"] = $this-
>mysql_model->search_username($temp[0]["uid"][0]);
                                        }
                                }

                                // primary email match
                                else if(strpos($search_string,".")){ // might be
a primary email

```

```

        if(!empty($_SESSION['results_db']) &&
!empty($_SESSION['search_string']) && empty($temp_navbar_search_str)){

            $data['results_db'] = $_SESSION['results_db'];
            $data['search_str'] = $_SESSION['search_string'];

        } else {
            $this->form_validation->set_rules('navbar_search_str',
'Search', 'required|xss_clean|min_length[3]');

            if ($this->form_validation->run() == TRUE){

                if(isset($_SESSION['results_db'])){
                    unset($_SESSION['results_db']);
                }

                if(isset($_SESSION['search_string'])){
                    unset($_SESSION['search_string']);
                }

                $search_string = $this->input-
>post('navbar_search_str');
                #TODO: Sanitize search string
                if($search_string != "" || $search_string != NULL) {
                    $stringArray = explode(" ", $search_string);
                    $count_query = count($stringArray);
                    $data["search_str"] = $search_string;

                    if(1==$count_query){ // only one keyword
                        if(strpos($search_string,"@upd.edu.ph")){ //
might be a primary email
                                // email match
                                $pos =
strpos($search_string,"@upd.edu.ph");
                                $email = substr($search_string, 0, $pos);
                                if(strpos($email,".")){ // might be a
primary email
                                        // $data["results_db"] = $this-
>mysql_model->search_email($email);
                                        $temp = $this->ldap_model-
>search_mail_addresses($email);
                                        $data["results_db"] = $this-
>mysql_model->search_username($temp[0]["uid"][0]);
                                        } else {
                                        // username match
                                        // $data["results_db"] = $this-
>mysql_model->search_username($email);
                                        $temp = $this->ldap_model-
>search_mail_addresses($email);
                                        $data["results_db"] = $this-
>mysql_model->search_username($temp[0]["uid"][0]);
                                        }

                                }

                                // primary email match
                                else if(strpos($search_string,".")){ // might be
a primary email

```

```

// $data["results_db"] = $this-
>mysql_model->search_email($search_string);

    if(filter_var($search_string,FILTER_VALIDATE_EMAIL)){
        $temp = $this->ldap_model-
>search_mail_addresses($search_string);
        $data["results_db"] = $this-
>mysql_model->search_username($temp[0]["uid"][0]);
        } else {
            // let's try adding @upd.edu.ph
            $expanded =
            $search_string."@upd.edu.ph";
            $temp = $this->ldap_model-
>search_mail_addresses($expanded);
            $data["results_db"] = $this-
>mysql_model->search_username($temp[0]["uid"][0]);

            if(empty($data['results_db'])){
                $data["results_db"] = $this-
>mysql_model->search_substring($search_string);
            }

        }

    }

    // exact student number match
    else if(strpos($search_string,"-")){ // might be
a student number
        if($this-
>_is_idno_studentno($search_string)){
            $search_result_sn = $this-
>mysql_model->search_substring_number($search_string);
            if(empty($search_result_sn)){ //
might be stored in LDAP
                $temp = $this->ldap_model-
>search_substring_number($search_string);
                $search_result_sn = $this-
>mysql_model->search_username($temp[0]["uid"][0]);
            }
            $data["results_db"] =
            $search_result_sn;
        } else { // might be partial
            $search_result_sn_partial = $this-
>mysql_model->search_substring_number($search_string);

            if(!empty($search_result_sn_partial)){
                $data["results_db"] =
            $search_result_sn_partial;
            }

        }

    }

    else if(is_numeric($search_string)){
        // employee number match
        $len = strlen($search_string);
        if($len<=9){
            $exact = FALSE;

```

```

        } else if($len==10){
            $exact = TRUE;
        }

        // check MySQL for the employee number
        $search_result = $this->mysql_model-
>search_substring_number($search_string,$exact);

        // it's possible that what's stored in
MySQL is less than 10 chars
        if(empty($search_result)){
            if($len==10){ // maybe the value
stored is 9 digits or lower
                $temp_search_string =
substr($search_string, 1);
                $exact = FALSE;
                $search_result = $this-
>mysql_model->search_substring_number($temp_search_string,$exact);
            }
        }
        /**/

        // it's possible that it's stored only in
LDAP
        if(empty($search_result)){
            $temp = $this->ldap_model-
>search_substring_number($search_string);
            if(empty($temp)){
                if($len==10){
                    $temp_search_string =
substr($search_string, 1);
                    $temp = $this-
>ldap_model->search_substring_number($temp_search_string);
                    $search_result = $this-
>mysql_model->search_username($temp[0]["uid"][0]);
                }
            } else {
                $search_result = $this-
>mysql_model->search_username($temp[0]["uid"][0]);
            }
        }

        // print "still empty";
        // still empty, might be a student number
without the dash
        if(empty($search_result)){
            if($this-
>_is_idno_studentno($search_string)){
                $temp_student_number =
"".substr($search_string,0, 4)."-".substr($search_string, 4, 9)."";
                $search_result_sn = $this-
>mysql_model->search_substring_number($temp_student_number);
                if(empty($search_result_sn)){
// might be stored in LDAP
                    $temp = $this-
>ldap_model->search_substring_number($temp_student_number);
                    $search_result_sn =
$this->mysql_model->search_username($temp[0]["uid"][0]);

```



```

        }
        $data["results_db"] =
$search_result_sn;
    }
    } else {
        $data["results_db"] =
$search_result;
    }

    // print_r($data['results_db']);
}
else {
    // try first a username search
    // $data["results_db"] = $this-
>mysql_model->search_username($search_string);
    $temp = $this->ldap_model-
>search_mail_addresses($search_string);
    $data["results_db"] = $this->mysql_model-
>search_username($temp[0]["uid"][0]);
    if(empty($data["results_db"])){
        $data["results_db"] = $this-
>mysql_model->search_substring($search_string);
    }
}

} else if($count_query>1){ // more than one
search keywords
    // try connecting them first
    $connected = implode("_", $stringArray);
    $data["results_db"] = $this->mysql_model-
>search_substring($connected);
    if(empty($data['results_db'])){
        $data["results_db"] = $this-
>mysql_model->search_substring($search_string);
    }
}

}

}

if(isset($data['results_db']) && !empty($data['results_db'])){
    $_SESSION['results_db'] = $data['results_db'];
    $_SESSION['search_string'] = $data["search_str"];
}
}

    // print "klasjdlfjasdfjkasfjdfaskf Count:
".count($data['results_db']);

    if(isset($data['results_db']) && count($data['results_db']) == 1 ) {
// exact match
        $this->session->set_flashdata('info', "Found exact match for
'".$search_string."'");
        $username = $data["results_db"][0]['username'];
        redirect('tech/dilnetacc_tools/view_account/'.$username.'');
    }
}

```

```

    }
    $this->load->vars($data);
    $this->load->view('page_template');
}

public function id_search_form() {
    // Load page variables
    $data = $this->page_vars;

    $this->form_validation->set_rules('id_str', 'ID Number',
    'required|alpha_dash');

    if ($this->form_validation->run() == TRUE){
        $search_string = $this->input->post('id_str');

        #TODO: Sanitize search string
        if($search_string != "" || $search_string != NULL) {
            $data["results_db"] = $this->mysql_model-
>search_substring_number($search_string);
            $data["search_str"] = $search_string;
        }
    }

    $this->load->vars($data);
    $this->load->view('page_template');
}
}

```

application/controllers/tech/Dilnetacc_tools.php

```

<?php
defined('BASEPATH') OR exit('No direct script access allowed');

class Dilnetacc_tools extends MY_Controller {

    function __construct() {
        parent::__construct();

        $this->form_validation->set_error_delimiters('<label>', '</label>');
        $this->load->model('ldap_model');
        $this->load->model('mysql_model');
        $this->load->model('proxy_tools_model');
        $this->load->model('mysql_webmail_model');
        $this->load->model('mail_tool_model');
        $this->load->model('log_model');

        $this->include_ldap = FALSE;
        $this->tech_user = $_SESSION['tech_user'];
        $this->ip = $this->input->ip_address();
        $this->page_vars = array(
            'content' => 'dilnet_account_tools',
            'page_title' => 'DilNet Account Tools',
            'page_subtitle' => 'Search and Print DilNet Account
Information',

```

```

        'include_stylesheets' => 'dilnet_account_tools_css',
        'include_scripts' => 'dilnet_account_tools_js'
    );
}

public function index() {
    // Load page variables
    $data = $this->page_vars;

    $this->load->vars($data);
    $this->load->view('page_template');
}

public function view_form() {
    // Load page variables
    $data = $this->page_vars;
    $type = $this->input->post('view_radio');

    $this->form_validation->set_rules('view_inp', 'Print Input Field',
'required|alpha_dash|callback__check_user_view');

    if ($this->form_validation->run() == TRUE){
        $inp = $this->input->post('view_inp');

        if($type != 'UN'){
            $user = $this->mysql_model->get_by_search_type($inp,
$type);

            $username = $user['username'];
        }
        else { $username = $inp; }

        #TODO Sanitize search string
        if($username != "" || $username != NULL) {
            redirect('tech/dilnetacc_tools/view_account/' . $username);
        }
    }

    $this->load->vars($data);
    $this->load->view('page_template');
}

public function print_form() {
    // Load page variables
    $data = $this->page_vars;
    $type = $this->input->post('print_radio');

    $this->form_validation->set_rules('print_inp', 'Print Input Field',
'required|alpha_dash|callback__check_user_print');

    if ($this->form_validation->run() == TRUE){
        $inp = $this->input->post('print_inp');

        if($type != 'UN'){

```

```

        $user = $this->mysql_model->get_by_search_type($inp,
$type);
        $username = $user['username'];
    }
    else { $username = $inp; }

    #TODO Sanitize search string
    if($username != "" || $username != NULL) {

redirect('tech/print_account/confirm_print/' . $username);
    }
    }

    $this->load->vars($data);
    $this->load->view('page_template');
}

public function reset_form($account = NULL) {
    #TODO make sure user is not in disabled table

    // Load page variables
    $data = $this->page_vars;
    $type = $this->input->post('reset_radio');

    $this->form_validation->set_rules('reset_inp', 'Reset Input Field',
'required|alpha_dash|callback__check_user_reset');

    if ($this->form_validation->run() == TRUE){

        $inp = $this->input->post('reset_inp');

        #TODO May use get_by_search_type with $key parameter
        $account = $this->mysql_model->get_by_search_type($inp,
$type);

        if(!empty($account)){
            $_SESSION['username'] = $account['username'];
            $this->session->mark_as_flash('username');
            redirect('tech/dilnetacc_tools/reset_confirm/');
        }
    }

    $this->load->vars($data);
    $this->load->view('page_template');
}

public function reset_confirm($username = NULL) {

    // Load page variables
    $data = array(
        'content' => 'account_reset_confirm',
        'page_title' => 'Reset Password',
        'page_subtitle' => null,
        'include_stylesheets' => 'dilnet_account_tools_css',
        'include_scripts' => 'dilnet_account_tools_js'
    );

    #Get username from previous screen (if coming from reset_form)
    if(isset($_SESSION['username'])) {

```

```

        $username = $_SESSION['username'];
    }

    #Get username from post input, if form is reloaded by
form_validation
    if($username == NULL){
        $username = $this->input->post('username');
    }

    if($username != NULL){
        if($username=='byron'){
            $_SESSION['error'] = "Can't reset superuser account
password.";
            $this->session->mark_as_flash('error');
            redirect('/tech/dilnet_lookup');
        }

        $account = $this->mysql_model->get_user_account($username);
        $account_ldap = $this->ldap_model-
>get_user_account($username);

        // print "<pre>";
        // print_r($account);
        // print_r($account_ldap["dn"]);
        // print "</pre>";

        if(isset($account['firstname']) &&
isset($account['lastname'])){
            $account['fullname'] = $account['firstname']."
".$account['lastname'];
        }

        else {

            $account['fullname'] = $account_ldap['cn'][0];
        }

        // GET ID NUMBERS
        $id_number_used = FALSE;
        if(!empty($account_ldap['employeenumber'])){
            if(is_array($account_ldap['employeenumber'])){
                $account['employeenumber'] =
$account_ldap['employeenumber'][0];
            } else {
                $account['employeenumber'] =
$account_ldap['employeenumber'];
            }
        } else {
            if(!empty($account['id_number'])){
                $account['employeenumber'] =
$account['id_number'];
                $id_number_used = TRUE;
            } else {
                $account['employeenumber'] = "No Data";
            }
        }

        // if by chance, the employee number is just a student number
        if($this->_is_idno_studentno($account['employeenumber'])) {

```

```

        // discard the number
        $account['employeenumber'] = "No Data";
    }

    if(!empty($account_ldap['studentnumber'])){
        if(is_array($account_ldap['studentnumber'])){
            $account['studentnumber'] =
$account_ldap['studentnumber'][0];
        } else {
            $account['studentnumber'] =
$account_ldap['studentnumber'];
        }
    } else {
        if(!$id_number_used && !empty($account['id_number'])){
            $account['studentnumber'] =
$account['id_number'];
        } else {
            $account['studentnumber'] = "No Data";
        }
    }

    //
    // End of GET ID NUMBERS
    //

    $this->form_validation->set_rules('reset_reason', 'Reason',
'xss_clean');

    if ($this->form_validation->run() == TRUE){

        $reason = $this->input->post('reset_reason');
        $_SESSION['passinfo']['reason'] = $reason;
        $this->session->mark_as_flash('passinfo');

        redirect('/tech/dilnetacc_tools/reset_final/' . $username);
    }

    $data['username'] = $username;
    $data['account'] = $account;

    $this->load->vars($data);
    $this->load->view('page_template');
}

else {
    #TODO Keep session data for next redirect
    if (isset($_SESSION['success'])){
        $_SESSION['success'] = $_SESSION['success'];
        $this->session->mark_as_flash('success');
    }
    else {
        $_SESSION['error'] = "Could not get username for
Password reset. Try again.";
        $this->session->mark_as_flash('error');
    }

    redirect('/tech/dilnet_lookup');
}
}

```

```

public function reset_final($username = NULL) {

    if ($username != NULL){

        $account_mysql = $this->mysql_model-
>get_user_account($username);
        $account_ldap = $this->ldap_model-
>get_user_account($username);

        $reason = $_SESSION['passinfo']['reason'];

        #TODO Allow reset of tech staff if role is supertech
        $data["ldap_print"]["username"] = $username;
        $data["ldap_print"]["password"] = random_password();

        mt_srand((double)microtime()*1000000);
        $salt = pack("CCCC", mt_rand(), mt_rand(), mt_rand(),
mt_rand());

        $newdata["userPassword"] = "{SHA}" .
base64_encode(pack("H*", sha1($data["ldap_print"]['password'] . $salt)) . $salt);

        $set_ldap_result = $this->ldap_model-
>set_ldapentry($username, $account_ldap["dn"], $newdata);
        if($set_ldap_result){
            $this->mysql_model->enable_force_changepass($username);

            $this->mysql_model-
>reset_recovery_count($username);

            # logging
            $message = $reason != NULL ? $reason : "reset
password";

            $log_result = $this->log_model-
>changepass_create($username, $this->ip, $this->tech_user, $message);
            // print "<pre>";
            // print "<br>username: $username";
            // print "<br>IP: ".$this->ip;
            // print "<br>TECH: ".$this->tech_user;
            // print "<br>Message: $message";
            // print_r($log_result);
            // print "</pre>";

            # render view and email details
            $data["ldap_print"]["email"] =
$account_ldap["mail"][0];
            $data["ldap_print"]["alternate_mail"] =
$account_ldap["mailalternateaddress"][0];
            $data["reason"] = $message;

            #TODO Enable sendmail
            // $this->_sendmail($data);

            $_SESSION['reset_print'] = $data["ldap_print"];
            $this->session->mark_as_flash('reset_print');
            redirect("/tech/print_account/confirm_print");

            # redirect to prevent resubmission
        }
        else {

```

```

        $_SESSION['error'] = "Reset password data lost.
Please try again.";
        $this->session->mark_as_flash('error');
        redirect("/tech/dilnet_lookup");
    }
}

else {
    $_SESSION['error'] = "Reset password data lost. Please try
again.";
    $this->session->mark_as_flash('error');
    redirect("/tech/dilnet_lookup");
}

}

public function reset_cancel($username = NULL) {
    if($username != NULL){
        $_SESSION['warning'] = "Cancelled Password reset for
<strong>".strtoupper($username)."</strong>.";
        $this->session->mark_as_flash('warning');
        redirect('tech/dilnetacc_tools/view_account/' . $username);
    }
    else { redirect('tech/dilnet_lookup'); }
}

public function disable_form($username = NULL) {

    // Load page variables
    $data = $this->page_vars;
    $type = $this->input->post('disable_radio');

    $this->form_validation->set_rules('disable_inp', 'Print Input
Field', 'required|alpha_dash|callback__check_user_disable');

    if ($this->form_validation->run() == TRUE){

        $inp = $this->input->post('disable_inp');

        $account = $this->mysql_model->get_by_search_type($inp,
$type);

        if(!empty($account)){
            $_SESSION['username'] = $account['username'];
            $this->session->mark_as_flash('username');
            redirect('tech/mail_tools/disable_confirm/');
        }
    }

    $this->load->vars($data);
    $this->load->view('page_template');
}

public function disable_confirm($username = NULL) {

    // Load page variables
    $data = array(

```



```

        'content' => 'account_disable_confirm',
        'page_title' => 'Disable Account',
        'page_subtitle' => null,
        'include_stylesheets' => 'dilnet_account_tools_css',
        'include_scripts' => 'dilnet_account_tools_js'
    );

    #Get username from previous page (if coming from disable_form)
    if(isset($_SESSION['username'])) {
        $username = $_SESSION['username'];
    }

    #Get username from post input, if form is reloaded by
form_validation
    if($username == NULL) {
        $username = $this->input->post('username');
    }

    if($username != NULL) {
        if($username == 'byron') {
            $_SESSION['error'] = "Can't disable superuser
account.";

            $this->session->mark_as_flash('error');
            redirect('/tech/dilnet_lookup');
        }

        $this->form_validation->set_rules('disable_reason', 'Reason',
'required|xss_clean');
        $this->form_validation->set_rules('mail_disable', 'Disable
Mail Delivery', 'trim');
        $this->form_validation->set_rules('proxy_disable', 'Disable
Proxy Access', 'trim');

        $account = $this->mysql_model->get_user_account($username);
        $account_ldap = $this->ldap_model-
>get_user_account($username);

        // CHANGE the Page Title
        if($account['disabled'] == TRUE) {
            $data['page_title'] = "Enable Account";
            $data['page_subtitle'] = "Restore DilNet Account
Access";
        } else if($account['disabled'] == FALSE) {
            $data['page_title'] = "Disable Account";
            $data['page_subtitle'] = "Restrict DilNet Account
Access";
        }

        // GET FULL NAME
        if(isset($account['firstname']) &&
isset($account['lastname'])) {
            $account['fullname'] = $account['firstname']."
".$account['lastname'];
        } else {
            $account['fullname'] = $account_ldap['cn'][0];
        }

        // GET ID NUMBERS
        $id_number_used = FALSE;

```

```

        if(!empty($account_ldap['employeenumber'])){
            if(is_array($account_ldap['employeenumber'])){
                $account['employeenumber'] =
$account_ldap['employeenumber'][0];
            } else {
                $account['employeenumber'] =
$account_ldap['employeenumber'];
            }
        } else {
            if(!empty($account['id_number'])){
                $account['employeenumber'] =
$account['id_number'];
                $id_number_used = TRUE;
            } else {
                $account['employeenumber'] = "No Data";
            }
        }

// if by chance, the employee number is just a student number
if($this->is_idno_studentno($account['employeenumber'])) {
    // discard the number
    $account['employeenumber'] = "No Data";
}

if(!empty($account_ldap['studentnumber'])){
    if(is_array($account_ldap['studentnumber'])){
        $account['studentnumber'] =
$account_ldap['studentnumber'][0];
    } else {
        $account['studentnumber'] =
$account_ldap['studentnumber'];
    }
} else {
    if(!$id_number_used && !empty($account['id_number'])){
        $account['studentnumber'] =
$account['id_number'];
    } else {
        $account['studentnumber'] = "No Data";
    }
}

//
// End of GET ID NUMBERS
//

$this->form_validation->set_rules('reset_reason', 'Reason',
'xss_clean');

if ($this->form_validation->run() == TRUE){
    if($this->input->post('mail_disable')){
        $_SESSION['mail_disable'] = TRUE;
    }

    if($this->input->post('proxy_disable')){
        $_SESSION['proxy_disable'] = TRUE;
    }
    $reason = $this->input->post('disable_reason');
    $_SESSION['disableinfo']['reason'] = $reason;
    $this->session->mark_as_flash('disableinfo');
}

```

```

    redirect('/tech/dilnetacc_tools/disable_final/' . $username);
    }

    $data['username'] = $username;
    $data['account'] = $account;

    $this->load->vars($data);
    $this->load->view('page_template');
}

else {
    #TODO Keep session data for next redirect
    if (isset($_SESSION['success'])) {
        $_SESSION['success'] = $_SESSION['success'];
        $this->session->mark_as_flash('success');
    }
    else {
        $_SESSION['error'] = "Could not get username for
Account disable operation. Try again.";
        $this->session->mark_as_flash('error');
    }

    redirect('/tech/dilnet_lookup');
}

}

public function disable_final($username = NULL) {

    if ($username != NULL) {

        $account_mysql = $this->mysql_model-
>get_user_account($username);
        $account_ldap = $this->ldap_model-
>get_user_account($username);

        $reason = $_SESSION['disableinfo']['reason'];

        $admin = $this->tech_user;
        $decoded_email = $username . "@upd.edu.ph";

        if($account_mysql['disabled']) {
            $reason = $reason != NULL ? $reason : NULL;
            $this->mysql_model->enable_account($username, $reason);
            $this->proxy_tools_model->enable_username($username,
$reason, $admin);
            $this->mail_tool_model->enable_email($decoded_email,
$reason, $admin);

            $this->ldap_model->enable_ldap_user($username);

            $this->log_model->create($username, "Account ENABLED.
". $reason, "update_mysql", $this->tech_user, $this->ip, 'info');

            $_SESSION['success'] = "Account
<strong>".strtoupper($username). "</strong> enabled.";
            $this->session->mark_as_flash("success");

```

```

    } else {
        $reason = $reason != NULL ? $reason : "";
        $this->mysql_model->disable_account($username,
$reason);
        if($_SESSION['mail_disable']){
            print "Disabling mail access ...<br>";
            $this->mail_tool_model-
>disable_email($decoded_email, $reason, $admin);
        }
        if($_SESSION['proxy_disable']){
            print "Disabling proxy access ...<br>";
            $this->proxy_tools_model-
>disable_username($username, $reason, $admin);
            $this->ldap_model->disable_ldap_user($username);
        }

        $this->log_model->create($username, "Account disabled.
".$reason, "update_mysql", $this->tech_user, $this->ip, 'info');
        $_SESSION['warning'] = "Account
<strong>".strtoupper($username)."</strong> disabled.";
        $this->session->mark_as_flash("warning");
    }

    redirect("/tech/dilnetacc_tools/view_account/".$username);
}

else {

    $_SESSION['error'] = "Disable account data lost. Please try
again.";
    $this->session->mark_as_flash('error');
    redirect("/tech/dilnet_lookup");
}

}

public function disable_cancel($username = NULL) {
    if($username != NULL){
        $_SESSION['warning'] = "Cancelled Account disable operation
for <strong>".strtoupper($username)."</strong>.";
        $this->session->mark_as_flash('warning');
        redirect('tech/dilnetacc_tools/view_account/'.$username);
    }
    else { redirect('tech/dilnet_lookup'); }
}

public function view_account($uid = NULL) {

    // Load page variables
    $data = array(
        'content' => 'account_user_view',
        'page_title' => 'View DilNet Account',
        'page_subtitle' => 'Review this user\'s account information',
        'include_stylesheets' => 'account_user_view_css',
        'include_scripts' => 'account_user_view_js'
    );

    if (isset($uid) && $uid != NULL){

```

```

        $mysql_account = $this->mysql_model->get_user_account($uid);
        $ldap_account = $this->ldap_model->get_user_account($uid);
        $webmaildb_account = $this->mysql_webmail_model-
>get_user_account($uid);

//FORMAT DATA
if (isset($ldap_account) && isset($mysql_account)){
    ##### BASIC VIEW #####

    $data['basic_view'] = array(
        'fullname' => $this-
>_format_array_model2ctrl($ldap_account,'cn','ldap'),
        'unit' => strtoupper($this-
>_format_array_model2ctrl($ldap_account,'ou','ldap')),
        'division' => $this-
>_format_array_model2ctrl($ldap_account,'departmentnumber','ldap'),
        'type' => $this-
>_format_array_model2ctrl($ldap_account,'employeetype','ldap'),
        'empno' => $this-
>_format_array_model2ctrl($ldap_account,'employeenumber','ldap'),
        'description' => $this-
>_format_array_model2ctrl($ldap_account,'description','ldap'),
        'studno' => $this-
>_format_array_model2ctrl($ldap_account,'studentnumber','ldap'),
        'designation' => $this-
>_format_array_model2ctrl($ldap_account,'title','ldap'),
        'firstname' => $this-
>_format_array_model2ctrl($mysql_account,'firstname','db'),
        'middlename' => $this-
>_format_array_model2ctrl($mysql_account,'middlename','db'),
        'lastname' => $this-
>_format_array_model2ctrl($mysql_account,'lastname','db'),
        'birthdate' => $this-
>_format_array_model2ctrl($mysql_account,'birthday','db'),
        'gender' => $this-
>_format_array_model2ctrl($mysql_account,'gender','db'),
        'username' => $this-
>_format_array_model2ctrl($ldap_account,'uid','ldap'),
        'mailquotasize' => $this-
>_format_array_model2ctrl($ldap_account,'mailquotasize','ldap')/1024/1024,
        'mailhost' => str_replace("up.edu.ph",
"upd.edu.ph", $this->_format_array_model2ctrl($ldap_account,'mailhost','ldap')),
        'mail' => $this-
>_format_array_model2ctrl($ldap_account,'mail','ldap'),
        'mailalternateaddress' => $this-
>_format_array_model2ctrl($ldap_account,'mailalternateaddress','ldap'),
        'mailforward' => $this-
>_format_array_model2ctrl($ldap_account,'mailforwardingaddress','ldap'),
        'webmail_last_login' => $this-
>_format_array_model2ctrl($webmaildb_account,'last_login','db'),
        'accounts_last_login' => $this-
>_format_array_model2ctrl($mysql_account,'last_login','db'),
        'recovery_email' => $this-
>_format_array_model2ctrl($mysql_account,'recovery_email','db'),
        'security_lastupdated' => $this-
>_format_array_model2ctrl($mysql_account,'security_lastupdated','db'),
        'date_activated' => $this-
>_format_array_model2ctrl($mysql_account,'date_activated','db'),

```

```

        'recovery_count' => $this->_format_array_model2ctrl($mysql_account,'recovery_count','db'),
        'activated_by' => $this->_format_array_model2ctrl($mysql_account,'activated_by','db'),
        'disabled' => $this->_format_array_model2ctrl($mysql_account,'disabled','db'),
        'disabled_remarks' => $this->_format_array_model2ctrl($mysql_account,'disabled_remarks','db'),
    );

    if(!isset($mysql_account['recovery_email']) ||
    empty($mysql_account['recovery_email']))
    {
        // check if there's pending recovery email
        $pending_recovery_email_check = $this->mysql_model->search_pending_recovery_email($uid);
        // print "<pre>";
        // print_r($pending_recovery_email_check);
        // print "</pre>";
        if(is_array($pending_recovery_email_check)
        && count($pending_recovery_email_check)>0)
        {
            $recovery_email_placeholder = "<div
class=\"label label-warning\" data-toggle=\"tooltip\" data-placement=\"auto\"
data-html=\"true\"
title=\"{\$pending_recovery_email_check[0]['date']}\">>PENDING</div>
\".\$pending_recovery_email_check[0]['email']];

            $data['basic_view']['recovery_email'] = $recovery_email_placeholder;
        }

        }
        if($mysql_account["date_activated"]==0){
            $data['basic_view']['date_activated'] =
"PRE-ACTIVATED";
        }

        # Security question type
        if ($mysql_account["security_question"] == "n/a")
        {
            $data['basic_view']['security_question_type'] = '<div class="label label-
danger">No recovery feature found</div>';
        }
        else if ($mysql_account["security_question"] ==
0) {

            $data['basic_view']['security_question_type'] = '<div class="label label-
primary">Custom question</div>';
        }
        else {

            $data['basic_view']['security_question_type'] = '<div class="label label-
primary">Preset Question</div>';
        }
        # Security Question Status
        if((($mysql_account["security_question"] < 6) &

```

```

0)) ||
"n/a") ||
NULL)
) {

    $data['basic_view']['recovery_feature_status'] = '<div class="label label-
danger">Recovery not set / outdated</div>';
    } else {

    $data['basic_view']['recovery_feature_status'] = '<div class="label label-
success">Updated recovery feature</div>';
    }

    if( !isset($data['basic_view']['birthdate'])
        || empty($data['basic_view']['birthdate'])
        || $data['basic_view']['birthdate']=="0000-00-
00")
    {
        $data['basic_view']['birthdate'] = "1908-06-18
[No Data]";
    }

    ##ADVANCED VIEW: To display LDAP vs MySQL information
    $data['account_keys'] = $account_keys =
array('Username', 'Full Name', 'Last Name', 'First Name', 'Middle Name',
'Birthdate', 'ID Number', 'Student Number', 'Employee Number', 'Employee Type',
'Unit', 'Division', 'Course', 'Title','Mail');

    foreach($account_keys as $key => $value){
        if($value == 'Username'){
            $ldap_view[$value] = $this-
>_format_array_model2ctrl($ldap_account, 'uid', 'ldap');
            $mysql_view[$value] = $this-
>_format_array_model2ctrl($mysql_account, 'username', 'db');
        }
        if($value == 'Full Name'){
            $ldap_view[$value] = $this-
>_format_array_model2ctrl($ldap_account, 'cn', 'ldap');
            $mysql_view[$value] = 'Not stored';
        }

        if($value == "Last Name"){
            $ldap_view[$value] = $this-
>_format_array_model2ctrl($ldap_account, 'sn', 'ldap');
            $mysql_view[$value] = $this-
>_format_array_model2ctrl($mysql_account, 'lastname', 'db');
        }

        if($value == 'First Name'){
            $ldap_view[$value] = 'Not stored';
            $mysql_view[$value] = $this-
>_format_array_model2ctrl($mysql_account, 'firstname', 'db');
        }

        if($value == "Middle Name"){
            $ldap_view[$value] = 'Not stored';

```

```

        $mysql_view[$value] = $this-
>_format_array_model2ctrl($mysql_account, 'middlename', 'db');
    }

    if($value == "Birthdate"){
        $ldap_view[$value] = 'Not stored';
        $mysql_view[$value] = $this-
>_format_array_model2ctrl($mysql_account, 'birthday', 'db');
    }

    if($value == "ID Number"){
        $ldap_view[$value] = 'Not stored';
        $mysql_view[$value] = $this-
>_format_array_model2ctrl($mysql_account, 'id_number', 'db');
    }

    if($value == "Student Number"){
        $ldap_view[$value] = $this-
>_format_array_model2ctrl($ldap_account, 'studentnumber', 'ldap');
        $mysql_view[$value] = 'Not stored';
    }

    if($value == "Employee Number"){
        $ldap_view[$value] = $this-
>_format_array_model2ctrl($ldap_account, 'employeenumber', 'ldap');
        $mysql_view[$value] = 'Not stored';
    }

    if($value == "Employee Type"){
        $ldap_view[$value] = $this-
>_format_array_model2ctrl($ldap_account, 'employeetype', 'ldap');
        $mysql_view[$value] = $this-
>_format_array_model2ctrl($mysql_account, 'type', 'db');
    }

    if($value == "Unit"){
        $ldap_view[$value] = $this-
>_format_array_model2ctrl($ldap_account, 'ou', 'ldap');
        $mysql_view[$value] = $this-
>_format_array_model2ctrl($mysql_account, 'unit_id', 'db');
    }

    if($value == "Division"){
        $ldap_view[$value] = $this-
>_format_array_model2ctrl($ldap_account, 'departmentnumber', 'ldap');
        $mysql_view[$value] = 'Not stored';
    }

    if($value == "Course"){
        $ldap_view[$value] = 'Not stored';
        $mysql_view[$value] = $this-
>_format_array_model2ctrl($mysql_account, 'course_id', 'db');
    }

    if($value == "Title"){
        $ldap_view[$value] = $this-
>_format_array_model2ctrl($ldap_account, 'title', 'ldap');
    }

```



```

        $mysql_view[$value] = $this->format_array_model2ctrl($mysql_account, 'designation', 'db');
    }
    if($value == "Mail"){
        $ldap_view[$value] = $this->format_array_model2ctrl($ldap_account, 'mail', 'ldap');
        $mysql_view[$value] = $this->format_array_model2ctrl($mysql_account, 'email', 'db');
        $mysql_view[$value] .= "@upd.edu.ph";
    }
}

$data['ldap_view'] = $ldap_view;
$data['uid'] = $uid;
$data['disabled'] = $this->format_array_model2ctrl($mysql_account, 'disabled', 'db');
$data['mysql_view'] = $mysql_view;
// print "<pre>";
// print_r($ldap_view);
// print_r($mysql_view);
// print "</pre>";
}

$data["results_accountlog"] = $this->mysql_model->get_accountlogs_by_user($uid);
$data["results_changepass"] = $this->mysql_model->get_changepass_logs_by_user($uid);
$data["results_lostpass"] = $this->mysql_model->get_lostpass_logs_by_user($uid);
}

else {
    #TODO Add proper redirect (or add an error message)
    redirect('tech/dilnet_lookup');
}

$this->load->vars($data);
$this->load->view('page_template');
}

public function edit_account_form($uid = NULL) {
    // Load page variables
    $data = array(
        'content' => 'account_user_edit',
        'page_title' => 'Edit DilNet Account',
        'page_subtitle' => 'Update this user\'s account information',
        'include_stylesheets' => 'account_user_create_css',
        'include_scripts' => 'account_user_edit_js'
    );

    if (!isset($uid) || $uid == NULL){
        $uid = $this->input->post('uid', TRUE);
    }

    if (isset($uid) || $uid != NULL){

```

```

// if($uid=='byron' && $this->tech_user!='byron'){
if($uid=='byron'){
    $_SESSION['error'] = "Can't Edit superuser account.";
    $this->session->mark_as_flash('error');
    redirect('tech/dilnetacc_tools/view_account/' . $uid);
}

$mysql_account = $this->mysql_model->get_user_account($uid);
$ldap_account = $this->ldap_model->get_user_account($uid);

$this->form_validation->set_rules('username',
'Username', 'trim|xss_clean|min_length[3]|callback_username_free_check');

$this->form_validation->set_rules('mail', 'Primary Email
Address', 'trim|valid_email|callback_email_free_check');

$this->form_validation->set_rules('mailalternateaddress',
'Alternate Email Address', 'trim|callback_alt_email_check');

$mailforward_test = $this->input->post('mailforward', TRUE);
if(isset($mailforward_test) && !empty($mailforward_test)){
    $this->form_validation->set_rules('mailforward', 'Mail
Forwarding Address', 'trim|valid_email|callback_email_not_dilnet');
}

$this->form_validation->set_rules('id_number', 'Employee
Number', 'numeric|min_length[9]');
$this->form_validation->set_rules('studentno', 'Student
Number', 'exact_length[10]|regex_match[/^[1-2][0-9][0-9][0-9]-[0-9][0-9][0-9][0-
9][0-9]+$/]');

#TODO Return validation, create callback
$this->form_validation->set_rules('firstname', 'First Name',
'xss_clean|min_length[2]|trim');
$this->form_validation->set_rules('middlename', 'Middle Name',
'xss_clean|min_length[1]|trim');
$this->form_validation->set_rules('lastname', 'Last Name',
'xss_clean|min_length[2]|trim');
$this->form_validation->set_rules('birthdate', 'Birthdate',
'xss_clean|regex_match[/^[0-9]{4}-[0-1][0-9]-[0-3][0-9]$/]');
$this->form_validation->set_rules('unit', 'Unit',
'required|alpha_dash');
$this->form_validation->set_rules('gender', 'Gender',
'required');
$this->form_validation->set_rules('division', 'Division',
'required|alpha_dash');
$this->form_validation->set_rules('designation',
'Designation', 'required');
$this->form_validation->set_rules('type', 'Type', 'required');

#Save account info from db and ldap and display in edit form
view
if (!empty($ldap_account) && !empty($mysql_account)){
    $info_post = array(
        'fullname' => $this-
>_format_array_model2ctrl($ldap_account, 'cn', 'ldap', 'form'),
        'mailquotasize' => $this-
>_format_array_model2ctrl($ldap_account, 'mailquotasize', 'ldap', 'form'),

```

```

        'unit' => strtoupper($this-
>_format_array_model2ctrl($ldap_account,'ou','ldap','form')),
        'division' => $this-
>_format_array_model2ctrl($ldap_account,'departmentnumber','ldap','form'),
        'type' => $this-
>_format_array_model2ctrl($ldap_account,'employeeetype','ldap','form'),
        'empno' => $this-
>_format_array_model2ctrl($ldap_account,'employeenumber','ldap','form'),
        'studno' => $this-
>_format_array_model2ctrl($ldap_account,'studentnumber','ldap','form'),
        'description' => $this-
>_format_array_model2ctrl($ldap_account,'description','ldap','form'),
        'designation' => $this-
>_format_array_model2ctrl($ldap_account,'title','ldap','form'),
        'firstname' => $this-
>_format_array_model2ctrl($mysql_account,'firstname','db','form'),
        'middlename' => $this-
>_format_array_model2ctrl($mysql_account,'middlename','db','form'),
        'lastname' => $this-
>_format_array_model2ctrl($mysql_account,'lastname','db','form'),
        'birthdate' => $this-
>_format_array_model2ctrl($mysql_account,'birthday','db','form'),
        'gender' => $this-
>_format_array_model2ctrl($mysql_account,'gender','db','form'),
        'mailforward' => $this-
>_format_array_model2ctrl($ldap_account,'mailforwardingaddress','ldap','form'),
        'mail' => $this-
>_format_array_model2ctrl($ldap_account,'mail','ldap','form'),
        'mailalternateaddress' => $this-
>_format_array_model2ctrl($ldap_account,'mailalternateaddress','ldap','form'),
        'uid' => $this-
>_format_array_model2ctrl($ldap_account,'uid','ldap','form'),
        'username' => $this-
>_format_array_model2ctrl($ldap_account,'uid','ldap','form')
    );

```

```

    $data['other_info'] = array(
        'username' => $this-
>_format_array_model2ctrl($ldap_account,'uid','ldap','form'),
        'mail' => $this-
>_format_array_model2ctrl($ldap_account,'mail','ldap','form'),
        'mailalternateaddress' => $this-
>_format_array_model2ctrl($ldap_account,'mailalternateaddress','ldap','form'),
        'firstname' => $this-
>_format_array_model2ctrl($mysql_account,'firstname','db','view'),
        'middlename' => $this-
>_format_array_model2ctrl($mysql_account,'middlename','db','view'),
        'lastname' => $this-
>_format_array_model2ctrl($mysql_account,'lastname','db','view')
    );

```

```

        if(!isset($info_post['birthdate']) ||
empty($info_post['birthdate'])) || $info_post['birthdate']=="0000-00-00"){
            print "<pre>";
            // print "birthdate: {$info_post['birthdate']}";
            // print "</pre>";
            $info_post['birthdate'] = "1908-06-18";

```

```

    }
    // print "<pre>";
    // print "birthdate: {$info_post['birthdate']}";
    // print "</pre>";

    #TODO Save to $_POST, check if secure
    if(empty($_POST)){
        $_POST = $info_post;
    }

    if($this->form_validation->run() == TRUE){
        $_SESSION['edit_form_new'] = $this->input-
>post(NULL,TRUE);
        $_SESSION['edit_form_orig'] = $info_post;
        $_SESSION['uid'] = $uid;
        $_SESSION['uid_dn'] = $ldap_account['dn'];
        $this->session->mark_as_flash('uid');
        $this->session->mark_as_flash('uid_dn');
        $this->session->mark_as_flash('edit_form_orig');
        $this->session->mark_as_flash('edit_form_new');

        redirect('/tech/dilnetacc_tools/edit_account_confirm');
    } //if.form_validation_run true

    $data['desig_list'] = $this->mysql_model-
>get_designation_list();
    $data['unit_list'] = $this->mysql_model-
>get_unit_list();
    $data['division_list'] = $this->mysql_model-
>get_division_list();
    $data['uid'] = $uid;
    $data['disabled'] = $this-
>_format_array_model2ctrl($mysql_account,'disabled','db','view');

    $this->load->vars($data);
    $this->load->view('page_template');

    } //if.isset ldap_account & mysql_account

    else {
        $_SESSION['error'] = "We cannot get the LDAP and/or
MySQL account of <strong>".strtoupper($username)."</strong>.";
        $this->session->mark_as_flash('error');
        redirect('/tech/dilnet_lookup');
    }

    } //if.isset.uid

    else {
        $_SESSION['error'] = "You are redirected to this page because
you were not supposed to access the edit account form page that way.";
        $this->session->mark_as_flash('error');
        redirect($_SESSION['homepage']);
    } //else.no.uid

```

```

}

public function edit_account_confirm() {

    if(isset($_SESSION['uid']) && $_SESSION['edit_form_orig'] &&
$_SESSION['edit_form_new']){

        $data = array(
            'content' => 'account_user_edit_confirm',
            'page_title' => 'Confirm user edit',
            'page_subtitle' => 'Search and Print DilNet Account
Information',
            'include_stylesheets' => 'dilnet_account_tools_css',
            'include_scripts' => 'dilnet_account_tools_js'
        );

        $data['uid'] = $uid = $_SESSION['uid'];
        $info_new = $_SESSION['edit_form_new'];
        $data['info_orig'] = $info_orig = $_SESSION['edit_form_orig'];
        $info_diff = array();

        // if($info_new['username']=='byron' && $this-
>tech_user!='byron'){
            if($info_new['username']=='byron'){
                $_SESSION['error'] = "Can't Edit superuser account.";
                $this->session->mark_as_flash('error');

                redirect('tech/dilnetacc_tools/view_account/' . $info_new['username']);
            }

            if(isset($info_new['username']) &&
!empty($info_new['username']) &&
(trim($info_new['username'])!=trim($info_orig['username']))) {

                if(filter_var($info_new['username'], FILTER_VALIDATE_EMAIL)){ // a valid
email was provided
                    list($email_only, $domain) = explode("@",
$info_new['username']);
                    $info_new['username'] = $email_only;
                }
            }

            if(!is_array($info_new['mailalternateaddress'])){
                foreach($info_new as $key => $value){
                    if ($key == 'mailalternateaddress') {
                        $tvalue = explode(PHP_EOL,
str_replace("\r", "", $value));
                        $info_new[$key] = $tvalue;
                    }
                }
            }

            if(isset($info_new['username']) &&
!empty($info_new['username']) &&
(trim($info_new['username'])!=trim($info_orig['username']))) {
                $new_uid_alt = $info_new['username']."@upd.edu.ph";
            }
        }
    }
}

```

```

        if(!in_array($new_uid_alt,$info_new['mailalternateaddress'])){
            $info_new['mailalternateaddress'][] =
trim($new_uid_alt);
        }
    }

    // restructure the mail alt address(es) as arrays
    if(!is_array($info_orig['mailalternateaddress'])){
        $temp_value[] = $info_orig['mailalternateaddress'];
        $info_orig['mailalternateaddress'] = $temp_value;
    }

    // this is a better version of getting diff fields
    $info_diff = arrayRecursiveDiff($info_new,$info_orig);

    $diff_mail_alt = array();

    if(isset($info_new['mailalternateaddress'])){
        foreach($info_orig['mailalternateaddress'] as
$t_address){
            if(in_array($t_address,$info_new['mailalternateaddress'])){
                $diff_mail_alt[] = trim($t_address);
            }
        }

        if(isset($info_new['mailalternateaddress'])){
            foreach($info_new['mailalternateaddress'] as
$t_address){
                if(!in_array($t_address,$info_orig['mailalternateaddress'])){
                    $diff_mail_alt[] = trim($t_address);
                }
            }
        }

        if($info_orig['mailalternateaddress'] != $diff_mail_alt){
            $info_diff['mailalternateaddress'] = $diff_mail_alt;
        } else {
            unset($info_diff['mailalternateaddress']);
        }

        if(!empty($info_diff)){
            $data['info_diff'] = $info_diff;
            $this->load->vars($data);
            $this->load->view('page_template');
        } else {
            $_SESSION['warning'] = "No changes were made.";
            $this->session->mark_as_flash('warning');
            redirect('tech/dilnetacc_tools/view_account/' . $uid);
        }
    } else {
        $_SESSION['error'] = "Session data lost. Try again.";
        $this->session->mark_as_flash('error');
        redirect('tech/dilnet_lookup');
    }

```

```

    }
}

public function edit_account_final() {

    $entry_diff = $this->input->post(NULL, TRUE);

    $uid = $this->input->post('uid');

    $new_uid = NULL;

    $account_ldap = $this->ldap_model->get_user_account($uid);
    $account_mysql = $this->mysql_model->get_user_account($uid);

    if(isset($entry_diff['username'])){
        $new_uid = $entry_diff['username'];
        $mysql_diff['username'] = trim($entry_diff['username']);
    }

    if(isset($entry_diff['mail'])){
        $ldap_diff['mail'] = $entry_diff['mail'];
        list($email_only, $domain) = explode("@",$entry_diff['mail']);
        $mysql_diff['email'] = $email_only;
    }

    if(isset($entry_diff['mailalternateaddress'])){
        $ldap_diff['mailalternateaddress'] =
$entry_diff['mailalternateaddress'];
    }

    if(isset($entry_diff['fullname'])){
        $ldap_diff['cn'] = $entry_diff['fullname'];
    }

    if(isset($entry_diff['mailquotasize'])){
        $ldap_diff['mailquotasize'] = $entry_diff['mailquotasize'];
    }
    if(isset($entry_diff['unit'])){
        $mysql_diff['unit_id'] = $ldap_diff['ou'] =
$entry_diff['unit'];
    }
    if(isset($entry_diff['division'])){
        $ldap_diff['departmentnumber'] = $entry_diff['division'];
    }
    if(isset($entry_diff['type'])){
        $mysql_diff['type'] = $ldap_diff['employeetype'] =
$entry_diff['type'];
        // if($entry_diff['type'] == "STUD")
        {$ldap_diff['employeetype'] = ""; }
    }
    #TODO Remove id_number
    if(isset($entry_diff['studno'])){
        $mysql_diff['id_number'] = $mysql_diff['studno'] =
$ldap_diff['studentnumber'] = $entry_diff['studno'];
    }

    if(isset($entry_diff['empno'])){

```

```

        $mysql_diff['id_number'] = $mysql_diff['empno'] =
$ldap_diff['employeenumber'] = $entry_diff['empno'];
    }

    if(isset($entry_diff['designation'])) {
        $mysql_diff['designation'] = $ldap_diff['title'] =
$entry_diff['designation'];
    }

    if(isset($entry_diff['firstname'])) {
        $mysql_diff['firstname'] = $entry_diff['firstname'];
        #Do not change ldap cn if name is incomplete
        if(isset($entry_diff['lastname'])) {
            $ldap_diff['cn'] = $entry_diff['firstname']."
".$entry_diff['lastname'];
        }
        else if(isset($mysql_account['lastname'])) {
            $ldap_diff['cn'] = $entry_diff['firstname']."
".$account_mysql['lastname'];
        }
    }

    if(isset($entry_diff['middlename'])) {
        if($entry_diff['middlename'] == 'none') {
$entry_diff['middlename'] = NULL; }
        $mysql_diff['middlename'] = $entry_diff['middlename'];
    }

    if(isset($entry_diff['lastname'])) {
        $mysql_diff['lastname'] = $entry_diff['lastname'];
        #TODO redundant function (same as what firstname is doing),
refactor
        #Do not change ldap cn if name is incomplete
        if(isset($entry_diff['firstname'])) {
            $ldap_diff['cn'] = $entry_diff['firstname']."
".$entry_diff['lastname'];
        }
        else if(isset($account_mysql['firstname'])) {
            $ldap_diff['cn'] = $account_mysql['firstname']."
".$entry_diff['lastname'];
        }
    }

    if(isset($entry_diff['birthdate'])) {
        $mysql_diff['birthday'] = $entry_diff['birthdate'];
    }

    if(isset($entry_diff['gender'])) {
        $mysql_diff['gender'] = $entry_diff['gender'];
    }

    if(isset($entry_diff['description'])) {
        $ldap_diff['description'] = $entry_diff['description'];
    }

    if(isset($entry_diff['mailforward'])) {
        $ldap_diff['mailforwardingaddress'] =
$entry_diff['mailforward'];
        if ($ldap_diff["mailforwardingaddress"] == NULL) {

```



```

        $ldap_diff["deliveryMode"] = NULL;
    } else {
        $ldap_diff["deliveryMode"][0] = "forwardonly";
        $ldap_diff["deliveryMode"][1] = "nombox";
    }
}
$update_ldap = $update_mysql = $rename_user = FALSE;

if(isset($ldap_diff)){
    print "<pre>";
    print_r($ldap_diff);
    print "</pre>";

    # Change null values to empty arrays
    array_walk($ldap_diff, function(&$n) { if(($n == NULL) || $n
== "") { $n = array(); } });
    $update_ldap = $this->ldap_model->set_ldapentry($uid,
$account_ldap['dn'], $ldap_diff);

    $ldap_keys = array_keys($ldap_diff);
    $ldap_keys = implode(",", $ldap_keys);

    // print_r($update_ldap);
    // print_r($ldap_keys);
}
if(isset($mysql_diff)){
    $update_mysql = $this->mysql_model-
>update_mysql_entry($mysql_diff, $uid);

    $mysql_keys = array_keys($mysql_diff);
    $mysql_keys = implode(",", $mysql_keys);
}

if(isset($new_uid) && !empty($new_uid)){
    $rename_user = $this->ldap_model-
>rename_ldap_user($uid, $new_uid);
    if(isset($_SESSION['merge_db'])){
        $_SESSION['merge_db'] = $this-
>removeElementWithValue($_SESSION['merge_db'], 'uid', $uid);
        array_push($_SESSION['merge_db'], $this->ldap_model-
>get_ldap_info($new_uid));
    }
}

if( $update_ldap || $update_mysql || $rename_user){
    if(isset($ldap_keys)){
        $this->log_model->create($uid, "LDAP fields updated:
".$ldap_keys, "update_ldap", $this->tech_user, $this->ip, 'info');
    }
    if(isset($mysql_keys)){
        $this->log_model->create($uid, "MySQL fields updated:
".$mysql_keys, "update_mysql", $this->tech_user, $this->ip, 'info');
    }

    if($rename_user==TRUE){
        $this->log_model->create($uid, "LDAP username renamed
from ".$uid." to ".$new_uid."", "update_ldap", $this->tech_user, $this->ip,
'info');
    }
}

```

```

    }

    $success_msg = "";

    if($update_ldap) { $success_msg .= "Account
<strong>".strtoupper($uid)."</strong> successfully updated in LDAP.<br>"; }
    if($update_mysql) { $success_msg .= "Account
<strong>".strtoupper($uid)."</strong> successfully updated in MySQL.<br>"; }
    if($rename_user) { $success_msg .= "Account ".$uid."
successfully renamed to <strong>".strtoupper($new_uid)."</strong>.<br>"; }

    $_SESSION['success'] = $success_msg;
    $this->session->mark_as_flash('success');
    if($rename_user){

        redirect('tech/dilnetacc_tools/view_account/' . $new_uid);
    } else {
        redirect('tech/dilnetacc_tools/view_account/' . $uid);
    }

}
else {
    $_SESSION['error'] = "Something went wrong. Account
<strong>".strtoupper($username)."</strong> was not updated.";
    $this->session->mark_as_flash('error');
    redirect('tech/dilnetacc_tools/view_account/' . $uid);
}
}

public function edit_account_cancel($uid = NULL) {
    if($uid != NULL){
        $_SESSION['warning'] = "Cancelled Edit for
<strong>".strtoupper($uid)."</strong>.";
        $this->session->mark_as_flash('warning');
        redirect('tech/dilnetacc_tools/view_account/' . $uid);
    } else {
        redirect('tech/dilnet_lookup');
    }
}

public function resend_welcome_email($username = NULL) {
    if($username != NULL){
        $ldap_account = $this->ldap_model-
>get_user_account($username);

        $welcome_data['user'] = array(
            'email' => $ldap_account['mail'][0],
            'firstname' => $ldap_account['cn'][0],
            'username' => $username
        );

        $this->_send_welcome_mail($welcome_data);
        $this->log_model->create($username, "Sending welcome email",
"account_update", $this->tech_user, $this->ip, 'debug');
        $_SESSION['success'] = "Welcome email sent to
<strong>".strtoupper($username)."</strong>.";
        $this->session->mark_as_flash('success');
        redirect('tech/dilnetacc_tools/view_account/' . $username);
    }
}

```

```

        else { redirect('tech/dilnetacc_tools'); }
    }

    /** Will be used for CALLBACKS */
    public function email_not_dilnet($query_email){
        if(filter_var($query_email, FILTER_VALIDATE_EMAIL)){ // valid email
            list($email_only, $domain) = explode("@",$query_email);
            if(isset($domain) && !empty($domain)){
                if(strtolower($domain) == "upd.edu.ph"){
                    $this->form_validation->
>set_message('email_not_dilnet', 'DilNet accounts (@upd.edu.ph) can\'t be used as
mail forwarding address. Please provide a non-DilNet email address.');
```

return FALSE;

```
                } else {
                    return TRUE;
                }
            } else {
                $this->form_validation->set_message('email_not_dilnet',
'Please provide a valid email.');
```

return FALSE;

```
            }
        } else {
            $this->form_validation->set_message('email_not_dilnet',
'Please provide a valid email.');
```

return FALSE;

```
        }
    }

    public function email_free_check($query_email){
        $query_uid = $this->input->post('uid');
        $query_email = explode(PHP_EOL, str_replace("\r", "",
$query_email));

        if(is_array($query_email)){

            foreach ($query_email as $value_temp) {
                $value = trim($value_temp);
                if(empty($value)){
                    $this->form_validation->
>set_message('email_free_check', 'Please provide a valid email.');
```

return FALSE;

```
                }

                list($email_only, $domain) = explode("@",$value);
                if(strtolower($domain) != "upd.edu.ph"){
                    $this->form_validation->
>set_message('email_free_check', '\''. $value. '\'' is not a valid DilNet account.
DilNet accounts end with @upd.edu.ph.');
```

return FALSE;

```
                }
            } else if (!$this->ldap_model->
>is_email_address_available($value,$query_uid)){
                $this->form_validation->set_message('email_free_check',
'\''.$value.'\' is already taken. Please choose another one.');
```

return FALSE;

```
            } else {
                return TRUE;
            }
        }
    }

```

```

        } else {
            $query_email = trim($query_email);
            if (!$this->ldap_model-
>is_email_address_available($query_email,$query_uid)){
                $this->form_validation->set_message('email_free_check',
                '\'.$query_email.'\' is already taken. Please choose another one.');
```

provided

```

                return FALSE;
            } else {
                return TRUE;
            }
        }
    }

    public function username_free_check($query){
        $query_uid = $this->input->post('uid');
        if(empty($query)){
            $this->form_validation->set_message('username_free_check',
            'Please provide a valid username.');
```

provided

```

            return FALSE;
        }
        if(filter_var($query, FILTER_VALIDATE_EMAIL)){ // full email was
            list($email_only, $domain_only) = explode("@",$query);
            if(strtolower($domain_only)!="upd.edu.ph"){
                $this->form_validation-
>set_message('username_free_check', '\'.$query.'\' only @upd.edu.ph domain is
allowed. Please provide another one.');
```

provided

```

            return FALSE;
        } else if (!$this->ldap_model-
>is_username_available($email_only,$query_uid)){
            $this->form_validation->set_message('username_free_check',
            '\'.$query.'\' is already taken. Please provide another one.');
```

provided

```

            return FALSE;
        } else {
            return TRUE;
        }
    }

    } else {
        $email_form = $query."@upd.edu.ph";
        if(filter_var($email_form, FILTER_VALIDATE_EMAIL)){
            if (!$this->ldap_model-
>is_username_available($query,$query_uid)){
                $this->form_validation-
>set_message('username_free_check', '\'.$query.'\' is already taken. Please
Provide another one.');
```

provided

```

                return FALSE;
            } else {
                return TRUE;
            }
        } else {
            $this->form_validation-
>set_message('username_free_check', '\'.$query.'\' is not a valid DilNet account.
Please provide another one.');
```

provided

```

            return FALSE;
        }
    }

    // if (!$this->ldap_model-
>is_username_available($query,$query_uid)){

```

```

        //          $this->form_validation-
>set_message('username_free_check', '\'.$query.\' is already taken. Please
provide another one.');
```

```

        //          return FALSE;
        //      } else {
        //          return TRUE;
        //      }
    }

    public function alt_email_check($query_alt){
        // print "<pre>query_alt<br>";
        // print_r($query_alt);
        // print "</pre>";

        $has_error = FALSE;
        $error_msg = "";

        if (empty($query_alt)) {
            $error_msg .= "At least one valid alternate email is
required.<br>";
            $has_error = TRUE;
        }

        $query_uid = $this->input->post('uid');
        $query_email = explode(PHP_EOL, str_replace("\r", "", $query_alt));

        if(is_array($query_email)){
            foreach ($query_email as $value_temp) {
                $value = trim($value_temp);
                if(!empty($value)){
                    @list($email_only, $domain) =
@explode("@",$value);

                    if(strtolower($domain) != "upd.edu.ph"){
                        $error_msg .= "'$value' is not a valid
DilNet account.<br>";

                        $has_error = TRUE;
                    }

                    if (!filter_var($value, FILTER_VALIDATE_EMAIL)) {
                        $error_msg .= "'$value' is not a valid
email.<br>";

                        $has_error = TRUE;
                    }

                    if (!$this->ldap_model-
>is_email_address_available($value,$query_uid)){
                        $error_msg .= "'$value' is already taken.<br>";
                        $has_error = TRUE;
                    }
                }
            }
        } else {
            $query_email = trim($query_email);

            if (!filter_var($query_email, FILTER_VALIDATE_EMAIL)) {
                $error_msg .= "'$value' is not a valid email.<br>";
                $has_error = TRUE;
            }
        }
    }
}

```

```

        if (!$this->ldap_model-
>is_email_address_available($query_email,$query_uid)){
            $error_msg .= "'$value' is already taken.<br>";
            $has_error = TRUE;
        }
    }
    if($has_error){
        $this->form_validation->set_message('alt_email_check',$error_msg);
        return FALSE;
    } else {
        return TRUE;
    }
}
}
/** HELPER FUNCTIONS **/

/** FORM FIELD CHECKERS **/

function _check_user_view($str) {
    $type = $this->input->post('view_radio');
    return $this->_check_user($str,$type,'_check_user_view');
}

function _check_user_print($str) {
    $type = $this->input->post('print_radio');
    return $this->_check_user($str,$type,'_check_user_print');
}

function _check_user_reset($str) {
    $type = $this->input->post('reset_radio');
    return $this->_check_user($str,$type,'_check_user_reset');
}

function _check_user_disable($str) {
    $type = $this->input->post('disable_radio');
    return $this->_check_user($str,$type,'_check_user_disable');
}

function _check_user($str, $type, $callback) {
    switch($type){
        case 'UN':
            if($this->mysql_model->get_by_search_type($str,'UN')) {
return true; }
            else { $this->form_validation->set_message($callback,
'Username does not exist.');
```

```

        if($this->mysql_model->get_by_search_type($str, 'EMP'))
    { return true; }
        else { $this->form_validation->set_message($callback,
'Employee number does not exist in DB.');
```

break;

```

        default:
            $this->form_validation->set_message($callback, 'User
does not exist in the database. Try again.');
```

}

```

        return false;
    }

    /*
    function _rule_valid_sn($sn) {
        if (substr($sn, 0, 4) > date("Y") || substr($sn, 0, 4) < 1940) {
            # use config for 1940
            return FALSE;
        }
        return TRUE;
    }
    /**/
    public function removeElementWithValue($array, $key, $value){
        foreach($array as $subKey => $subArray){
            if($subArray[0][$key][0] == $value){
                unset($array[$subKey]);
            }
        }
        return $array;
    }
}
}
}

```

application/controllers/tech/Mail_tools.php

```

<?php
defined('BASEPATH') OR exit('No direct script access allowed');

class Mail_tools extends MY_Controller {

    function __construct(){
        parent::__construct();

        $this->form_validation->set_error_delimiters('<label>', '</label>');
        $this->load->model('ldap_model');
        $this->load->model('mysql_model');
        $this->load->model('log_model');
        $this->load->model('mail_tool_model');

        $this->tech_user = $_SESSION['tech_user'];
        $this->ip = $this->input->ip_address();

        $this->page_vars = array(
            'content' => 'mail_search_view',
            'page_title' => 'Search Mail Logs',
            'page_subtitle' => 'Search the Mail Logs',
            'include_stylesheets' => 'account_user_view_css',

```

```

        'include_scripts' => 'account_user_view_js'
    );
}
public function index() {

```

```

    // Load page variables
    $data = $this->page_vars;

    $this->load->vars($data);
    $this->load->view('page_template');
}

```

```

public function mail_search($new=NULL) {

```

```

    $data = array(
        'content' => 'mail_search_view',
        'page_title' => 'Search Mail Logs',
        'page_subtitle' => 'Search the Mail Logs',
        'include_stylesheets' => 'account_user_view_css',
        'include_scripts' => 'account_user_view_js'
    );

```

```

    if($new=="new"){
        if(isset($_SESSION['mail_search_results_db'])){
            unset($_SESSION['mail_search_results_db']);
        }

        if(isset($_SESSION['mail_search_string'])){
            unset($_SESSION['mail_search_string']);
        }

        if(isset($_SESSION['link_referer'])){
            unset($_SESSION['link_referer']);
        }
    }

```

```

    $temp_mail_search_str = $this->input->post('mail_search_str');

```

```

    if(!empty($_SESSION['mail_search_results_db']) &&
!empty($_SESSION['mail_search_string']) && empty($temp_mail_search_str)){

        $data['mail_search_results_db'] =
$_SESSION['mail_search_results_db'];
        $data['mail_search_string'] = $_SESSION['mail_search_string'];
        $data['link_referer'] = $_SESSION['link_referer'];
    }

```

```

    } else {
        $this->form_validation->set_rules('mail_search_str', 'Search',
'required|trim|xss_clean|min_length[3]');
    }

```

```

    if ($this->form_validation->run() == TRUE){

```

```

        $temp = $this->input->post('mail_search_str');

```

```

        if(isset($_SESSION['mail_search_results_db'])){
            unset($_SESSION['mail_search_results_db']);
        }
    }

```



```

    }

    if(isset($_SESSION['mail_search_string'])){
        unset($_SESSION['mail_search_string']);
    }

    if(isset($_SESSION['link_referer'])){
        unset($_SESSION['link_referer']);
    }

    if(!empty($temp)){
        $search_string = $this->input-
>post('mail_search_str');
        if(isset($_SESSION['mail_search_results_db'])){
            unset($_SESSION['mail_search_results_db']);
        }
        } else {
            $this->session->set_flashdata('error',
"Invalid search query.");
            redirect('tech/mail_tools/mail_search/new');
        }

        // exact match for a message ID
        if(preg_match('/^[A-Za-z_0-9]{6}-[A-Za-z_0-9]{6}-[A-Za-
z_0-9]{2}$/', $search_string, $match)){
            redirect('tech/mail_tools/view_message_id/'.$search_string.'');
        }

        $mail_log = array();
        $temp = $this->mail_tool_model->search($search_string);

        for($i=0;$i<count($temp);$i++){
            $exploded = explode(" ", $temp[$i]);
            if(preg_match('/^[A-Za-z_0-9]{6}-[A-Za-z_0-9]{6}-
[A-Za-z_0-9]{2}$/', $exploded[2], $match)){
                $col_IDs[] = $exploded[2];
            }
        }
        $tempIDs_result = array();

        foreach($col_IDs as $col_ID){
            $tempIDs_result[] = $this->mail_tool_model-
>search($col_ID);
        }

        foreach($tempIDs_result as $temp_items){
            foreach($temp_items as $temp_each){
                $exploded = explode(" ", $temp_each);
                foreach($col_IDs as $col_ID){
                    if($col_ID == $exploded[2]){ // same
mail ID
sender log line
                    if($exploded[3] == "<="){ //

```

```

if(!empty($exploded[4])){
$mail_log[$col_ID]["sender"] = trim($exploded[4]);
} else {

$mail_log[$col_ID]["sender"] = "NULL";
}

$mail_log[$col_ID]["start_time"] = "".trim($exploded[0])."
.trim($exploded[1])."";

if(empty($mail_log[$col_ID]["start_time"])){
$mail_log[$col_ID]["start_time"] = "NULL";
}

if(preg_match('/\[.*?\]/', $exploded[6], $match)){
$mail_log[$col_ID]["sender_ip"] = trim(trim($exploded[6]),"[]");
} else {

if(preg_match('/\[.*?\]/', $exploded[7], $match)){
$mail_log[$col_ID]["sender_ip"] = trim(trim($exploded[7]),"[]");
} else {

$mail_log[$col_ID]["sender_ip"] = "NULL";
}
}
}

if($exploded[3] == ">"){ //
recipient line
$mail_log[$col_ID]["recipient"] = trim($exploded[4]);

if(empty($mail_log[$col_ID]["recipient"])){
$mail_log[$col_ID]["recipient"] = "NULL";
}
} else if($exploded[3] ==
"***"){ // returned email
if(!empty($exploded[4])){
$mail_log[$col_ID]["recipient"] = trim($exploded[4]);
} else {

$mail_log[$col_ID]["recipient"] = "NULL";
}
}

if(count($exploded) == 4){ //

```

this must be the Completed line, ideally

```

        $mail_log[$col_ID]["end_time"] = "".trim($exploded[0])."
.trim($exploded[1])."";

        if(empty($mail_log[$col_ID]["end_time"])){

            $mail_log[$col_ID]["end_time"] = "NULL";

        }

        $mail_log["$col_ID"]["status"] = trim($exploded[3]);

        if(empty($mail_log["$col_ID"]["status"])){

            $mail_log["$col_ID"]["status"] = "NULL";

        }

    }

    // all else fails and empty
recipient

    if(empty($mail_log[$col_ID]['recipient'])){

        $mail_log[$col_ID]["recipient"] = "NULL";

    }

}

}

    if(!empty($mail_log)){
        $data['mail_search_results_db'] = $mail_log;
        $data['mail_search_string'] = $search_string;
        $data['link_referer'] = 'mail_search';

        $_SESSION['mail_search_results_db'] = $mail_log;
        $_SESSION['mail_search_string'] = $search_string;
        $_SESSION['link_referer'] = 'mail_search';
    } else {
        $this->session->set_flashdata('warning', "No
match found.");

        redirect('tech/mail_tools/mail_search/');

    }

}

// Load page variables
$this->load->vars($data);
$this->load->view('page_template');
}

public function view_message_id($ID) {

    // Load page variables
    $data = array(
        'content' => 'mail_id_view',
        'page_title' => 'View Mail Log Entry',
        'page_subtitle' => 'View Specific Log Entry',
        'include_stylesheets' => 'account_user_view_css',
        'include_scripts' => 'account_user_view_js'
    );

```

```

    );

    $data['raw_entry'] = $this->mail_tool_model->view_log_entry($ID);
    $data['mail_ID'] = $ID;

    // if(isset($_SESSION['link_referer']) &&
!empty($_SESSION['link_referer'])) {
    //     $data['link_referer'] = $_SESSION['link_referer'];
    // } else {
    //     $data['link_referer'] = 'view_message_id' . $ID;
    // }
    // $data['link_referer'] = 'view_message_id/' . $ID;

    $temp = $data['raw_entry'];
    // print "<pre>";
    // print_r($temp);
    // print "</pre>";

    for($i=0;$i<count($temp);$i++){
        $exploded = explode(" ", $temp[$i]);
        foreach($temp as $temp_each){
            $exploded = explode(" ", $temp_each);
            $col_ID = $exploded[2];
            if($ID == $col_ID){ // same mail ID
                if($exploded[3] == "<="){ // sender log line
                    if(!empty($exploded[4])){
                        $mail_log[$col_ID]["sender"] =
$exploded[4];
                    } else {
                        $mail_log[$col_ID]["sender"] =
"NULL";
                    }

                    $mail_log[$col_ID]["sender_disabled"] =
$this->mail_tool_model->is_email_disabled($mail_log[$col_ID]["sender"]);

                    $mail_log[$col_ID]["start_time"] =
"". $exploded[0]. " " . $exploded[1]. "";

                    if(preg_match('/"([^\"]+)"\/', $temp_each,
$match)){
                        $mail_log[$col_ID]["subject"] =
$match[1];
                    } else {
                        $mail_log[$col_ID]["subject"] = "No
Data";
                    }

                    if($mail_log[$col_ID]["sender"]=="<>"){
                        if(preg_match('/\[.*?\]/',
$temp_each, $match)){
                            // print "<pre>";
                            // print_r($match);
                            // print "</pre>";

                            $mail_log[$col_ID]["sender_ip"] = trim(trim($match[0]),"[]");

```

```

        $mail_log[$col_ID]["sender_ip_disabled"] = $this->mail_tool_model-
>is_host_disabled($mail_log[$col_ID]["sender_ip"]);

        $mail_log[$col_ID]['sender_host'] = trim($exploded[5]);
                                } else {

        $mail_log[$col_ID]["sender_ip"] = "NULL";

        $mail_log[$col_ID]["sender_ip_disabled"] = NULL;

        $mail_log[$col_ID]['sender_host'] = "NULL";
                                }

                                } else {
                                        if(preg_match('/\[.*?\]/',
$temp_each, $match)){

        $mail_log[$col_ID]["sender_ip"] = trim(trim($match[0]),"[]");
                                } else {

        $mail_log[$col_ID]["sender_ip"] = "NULL";
                                }

        $mail_log[$col_ID]["sender_ip_disabled"] = $this->mail_tool_model-
>is_host_disabled($mail_log[$col_ID]["sender_ip"]);

                                $mail_log[$col_ID]['sender_host'] =
trim($exploded[5]);

                                }
        }

        if($exploded[3] == ">"){ // recipient line

                if(!empty($exploded[4])){
                        $mail_log["$col_ID"]["recipient"] =
trim($exploded[4]);

                } else {
                        $mail_log["$col_ID"]["recipient"] =
"NULL";

                }

                $mail_log[$col_ID]["recipient_disabled"] =
$this->mail_tool_model->is_email_disabled($mail_log[$col_ID]["recipient"]);

        } else if($exploded[3] == "***"){ // returned
email

                if(!empty($exploded[4])){
                        $mail_log["$col_ID"]["recipient"] =
trim($exploded[4]);

                } else {
                        $mail_log["$col_ID"]["recipient"] =
"NULL";

                }

```

```

    }

    if(count($exploded) == 4){ // this must be the
Completed line, ideally
        $mail_log[$col_ID]["end_time"] =
        "".trim($exploded[0])." ".trim($exploded[1])."";
        if(empty($mail_log[$col_ID]["end_time"])){
            $mail_log[$col_ID]["end_time"] =
"NULL";
        }
        if(!empty($exploded[3])){
            $mail_log["$col_ID"]["status"] =
trim($exploded[3]);
        } else {
            $mail_log["$col_ID"]["status"] =
"NULL";
        }
    }
}

if(empty($mail_log[$col_ID]["recipient_disabled"])){
    $mail_log[$col_ID]["recipient_disabled"] = NULL;
}

if(empty($mail_log[$col_ID]['recipient'])){
    $mail_log["$col_ID"]["recipient"] = "NULL";
}

if(empty($mail_log[$col_ID]['sender_host'])){
    $mail_log[$col_ID]['sender_host'] = "NULL";
}
}

if(!empty($mail_log)){
    $data['results_db'] = $mail_log;
    // _SESSION['link_referer'] = $data['link_referer'];
} else {
    $this->session->set_flashdata('error', "Message ID not
found.");
    redirect('tech/mail_tools/mail_search/');
}

$this->load->vars($data);
$this->load->view('page_template');
}

public function block_email_confirm($block_email = NULL) {

    // Load page variables
    $data = array(
        'content' => 'mail_account_disable_confirm',
        'page_title' => 'Block Email Address',
        'page_subtitle' => null,
        'include_stylesheets' => 'dilnet_account_tools_css',
        'include_scripts' => 'dilnet_account_tools_js'
    );

```

```

        #Get username from previous page (if coming from disable_form)
        if(!empty($_SESSION['block_email']))){
            $block_email = $_SESSION['block_email'];
        }

        #Get username from post input, if form is reloaded by
form_validation
        if(empty($block_email)){
            $block_email = $this->input->post('block_email',TRUE);
        }

        if(!empty($block_email)){
            $email = $this->mail_tool_model->
>get_block_email_info(base64_decode(urldecode($block_email)));

            $this->form_validation->set_rules('disable_reason', 'Reason',
'required|xss_clean');

            if ($this->form_validation->run() == TRUE){

                $reason = $this->input->post('disable_reason');
                $_SESSION['disableinfo']['reason'] = $reason;
                $this->session->mark_as_flash('disableinfo');

                redirect('/tech/mail_tools/block_email_final/'.urlencode($block_email));
            }

            $data['block_email'] = trim($block_email);
            $data['email'] = $email;

            if(isset($_SESSION['link_referer']) &&
!empty($_SESSION['link_referer']))){
                $data['link_referer'] = $_SESSION['link_referer'];
            } else {
                $data['link_referer'] = 'email_blocking';
            }

            $this->load->vars($data);
            $this->load->view('page_template');
        }

        else {
            #TODO Keep session data for next redirect
            if (isset($_SESSION['success'])){
                $_SESSION['success'] = $_SESSION['success'];
                $this->session->mark_as_flash('success');
            }
            else {
                $_SESSION['error'] = "Could not get username for
Account disable operation. Try again.";
                $this->session->mark_as_flash('error');
            }

            redirect('/tech/mail_tools/email_blocking');
        }
    }

    public function block_email_final($block_email = NULL) {

```

```

        if (!empty($block_email)){

            // print "block_email: $block_email<br>";
            $decoded_email = base64_decode(urldecode($block_email));
            // print "decoded_email: $decoded_email";

            $account_mysql = $this->mail_tool_model->get_block_email_info($decoded_email);

            $reason = $_SESSION['disableinfo']['reason'];

            $admin = $this->tech_user;

            if($account_mysql['disabled']) {
                $this->session->set_flashdata('info', "Account
<strong>". $decoded_email. "</strong> already blocked. No action taken.");
                // $this->session->mark_as_flash("success");

            } else {
                $reason = $reason != NULL ? $reason : "";
                $this->mail_tool_model->disable_email($decoded_email,
$reason, $admin);
                $this->log_model->create($decoded_email, "Account
disabled. ". $reason, "update_mysql", $this->tech_user, $this->ip, 'info');
                $this->session->set_flashdata('warning', "Account
<strong>". $decoded_email. "</strong> blocked.");
            }

            redirect('tech/mail_tools/view_email_status/'.urlencode(base64_encode($deco
ded_email)));
        }

        else {

            $_SESSION['error'] = "Disable account data lost. Please try
again.";
            $this->session->mark_as_flash('error');
            redirect("/tech/mail_tools/email_blocking/");

        }

    }

    public function unblock_email_confirm($block_email = NULL) {

        // Load page variables
        $data = array(
            'content' => 'mail_account_disable_confirm',
            'page_title' => 'UnBlock Email Address',
            'page_subtitle' => null,
            'include_stylesheets' => 'dilnet_account_tools_css',
            'include_scripts' => 'dilnet_account_tools_js'
        );

        if(!empty($_SESSION['block_email'])){
            $block_email = $_SESSION['block_email'];
        }
    }

```



```

        #Get username from post input, if form is reloaded by
form_validation
        if(empty($block_email)){
            $block_email = $this->input->post('block_email',TRUE);
        }

        if(!empty($block_email)){
            $email = $this->mail_tool_model-
>get_block_email_info(base64_decode(urldecode($block_email)));

            $this->form_validation->set_rules('disable_reason', 'Reason',
'required|xss_clean');

            if ($this->form_validation->run() == TRUE){

                $reason = $this->input->post('disable_reason');
                $_SESSION['disableinfo']['reason'] = $reason;
                $this->session->mark_as_flash('disableinfo');

                redirect('/tech/mail_tools/unblock_email_final/'.urlencode($block_email));
            }

            $data['block_email'] = trim($block_email);
            $data['email'] = $email;

            if(isset($_SESSION['link_referer']) &&
!empty($_SESSION['link_referer']))){
                $data['link_referer'] = $_SESSION['link_referer'];
            } else {
                $data['link_referer'] = 'email_blocking';
            }

            $this->load->vars($data);
            $this->load->view('page_template');
        }

        else {
            #TODO Keep session data for next redirect
            if (isset($_SESSION['success'])){
                $_SESSION['success'] = $_SESSION['success'];
                $this->session->mark_as_flash('success');
            }
            else {
                $_SESSION['error'] = "Could not get username for
Account disable operation. Try again.";
                $this->session->mark_as_flash('error');
            }

            redirect('/tech/mail_tools/email_blocking');
        }
    }

    public function unblock_email_final($block_email = NULL) {
        if (!empty($block_email)){
            // print "block_email: $block_email<br>";
            $decoded_email = base64_decode(urldecode($block_email));
            // print "decoded_email: $decoded_email";

```

```

        $account_mysql = $this->mail_tool_model-
>get_block_email_info($decoded_email);

        $reason = $_SESSION['disableinfo']['reason'];

        $admin = $this->tech_user;

        if($account_mysql['disabled']) {
            $reason = $reason != NULL ? $reason : "";
            $this->mail_tool_model->enable_email($decoded_email,
$reason, $admin);
            $this->log_model->create($decoded_email, "Account
enabled. ".$reason, "update_mysql", $this->tech_user, $this->ip, 'info');
            $this->session->set_flashdata('info', "Account
<strong>".$decoded_email."</strong> unblocked.");

        } else {
            $this->session->set_flashdata('info', "Account
<strong>".$decoded_email."</strong> is not blocked. No action taken.");
            // $this->session->mark_as_flash("success");
        }

        redirect('tech/mail_tools/view_email_status/'.urlencode(base64_encode($deco
ed_email)));
    }

    else {

        $_SESSION['error'] = "Disable account data lost. Please try
gain.";

        $this->session->mark_as_flash('error');
        redirect("/tech/mail_tools/email_blocking/");

    }

}

public function remove_queued_emails($block_email = NULL) {
    // Load page variables
    $data = array(
        'content' => 'remove_mail_view',
        'page_title' => 'Remove Blocked Emails',
        'page_subtitle' => 'Remove Blocked Emails',
        'include_stylesheets' => 'account_user_view_css',
        'include_scripts' => 'account_user_view_js'
    );

    if (!empty($block_email)){
        // print "block_email: $block_email<br>";
        $decoded_email = base64_decode(urldecode($block_email));
        // print "decoded_email: $decoded_email";

        $account_mysql = $this->mail_tool_model-
get_block_email_info($decoded_email);

        // $reason = $_SESSION['disableinfo']['reason'];

```

```

        $admin = $this->tech_user;

        if($account_mysql['disabled']) {
            $reason = "Remove queued emails.";

            $mail_log = $this->mail_tool_model->remove_emails($decoded_email);
            $this->log_model->create($decoded_email, "Removed
queued emails. ".$reason, "update_mysql", $this->tech_user, $this->ip, 'info');
            $this->session->set_flashdata('info', "Queued emails
from <strong>".$decoded_email."</strong> removed. <br>Removed
".$mail_log[0]."</strong> email(s) from Primary Relay. <br> Removed
".$mail_log[1]."</strong> email(s) from Secondary Relay.");

        } else {
            $this->session->set_flashdata('info', "Account
<strong>".$decoded_email."</strong> is not blocked. No action taken. <br>Can only
remove emails from blocked accounts.");
            // $this->session->mark_as_flash("success");
        }

        redirect('tech/mail_tools/view_email_status/'.urlencode(base64_encode($decoded_email)));
    }

    else {

        $_SESSION['error'] = "Disable account data lost. Please try
again.";
        $this->session->mark_as_flash('error');
        redirect("/tech/mail_tools/email_blocking/");
    }
}

public function email_blocking($new=NULL) {

    $data = array(
        'content' => 'mail_blocking_search_view',
        'page_title' => 'Search Mail DB',
        'page_subtitle' => 'Search the Mail DB for blocked/unblocked
email address(es)',
        'include_stylesheets' => 'account_user_view_css',
        'include_scripts' => 'account_user_view_js'
    );

    if($new=="new"){
        if(isset($_SESSION['email_blocking_results_db'])){
            unset($_SESSION['email_blocking_results_db']);
        }
        if(isset($_SESSION['email_blocking_search_string'])){
            unset($_SESSION['email_blocking_search_string']);
        }

        if(isset($_SESSION['link_referer'])){
            unset($_SESSION['link_referer']);
        }
    }

```

```

    }
}

$temp_mail_block_search_str = $this->input-
>post('mail_block_search_str');
    if(!empty($_SESSION['email_blocking_results_db']) &&
!empty($_SESSION['email_blocking_search_string']) &&
empty($temp_mail_block_search_str)){

        $data['email_blocking_results_db'] =
$_SESSION['email_blocking_results_db'];
        $data['email_blocking_search_string'] =
$_SESSION['email_blocking_search_string'];
        $data['link_referer'] = $_SESSION['link_referer'];

    } else {
        $this->form_validation->set_rules('mail_block_search_str',
'Search', 'required|trim|min_length[3]|xss_clean');

        if ($this->form_validation->run() == TRUE){

            $temp = $this->input->post('mail_block_search_str');

            if(isset($_SESSION['email_blocking_results_db'])){
                unset($_SESSION['email_blocking_results_db']);
            }

            if(isset($_SESSION['email_blocking_search_string'])){
                unset($_SESSION['email_blocking_search_string']);
            }

            if(isset($_SESSION['link_referer'])){
                unset($_SESSION['link_referer']);
            }

            if(!empty($temp)){
                $search_string = $this->input-
>post('mail_block_search_str', TRUE);
                if (filter_var($search_string,
FILTER_VALIDATE_EMAIL)) {

                    redirect('tech/mail_tools/view_email_status/'.urlencode(base64_encode($search_stri
ng)));

                } else {

                    $search_results = $this->mail_tool_model-
>search_email($search_string);

                    if(!empty($search_results)){

                        $data["email_blocking_results_db"] =
$search_results;

                        $data['email_blocking_search_string'] = $search_string;
                        $data['link_referer'] =
'email_blocking';

```

```

$_SESSION["email_blocking_results_db"] = $search_results;

$_SESSION['email_blocking_search_string'] = $search_string;

$_SESSION['link_referer'] =
'email_blocking';

        } else {
            $this->session->
>set_flashdata('warning', "No match found.");
            redirect('tech/mail_tools/email_blocking/');
        }

    } else {
        $this->session->set_flashdata('error', "Invalid
search query.");
        redirect('tech/mail_tools/email_blocking/');
    }
}
// Load page variables
$this->load->vars($data);
$this->load->view('page_template');
}

public function view_email_status($email) {
    // Load page variables
    $data = array(
        'content' => 'mail_address_view',
        'page_title' => 'View eMail Status',
        'page_subtitle' => 'View eMail Status on Mail Relays',
        'include_stylesheets' => 'account_user_view_css',
        'include_scripts' => 'account_user_view_js'
    );

    $decoded_email = base64_decode(urldecode($email));
    $mail_log = $this->mail_tool_model->
>get_block_email_info($decoded_email);

    // if(isset($_SESSION['link_referer']) &&
!empty($_SESSION['link_referer'])) {
        // $data['link_referer'] = $_SESSION['link_referer'];
    // } else {
        // $data['link_referer'] = 'email_blocking';
    // }

    $data['block_email'] = $decoded_email;

    if(!empty($mail_log)) {
        $data['results_db'] = $mail_log;
        // $_SESSION['link_referer'] = 'view_email_status/'.$email;
    } else {
        $this->session->set_flashdata('error', "Email address not in
database.");
    }
}

```

```

        redirect('tech/mail_tools/block_email_confirm/'.urlencode(base64_encode($de
coded_email)));
    }

    $this->load->vars($data);
    $this->load->view('page_template');
}

public function host_blocking($new=NULL) {

    // Load page variables
    $data = array(
        'content' => 'host_blocking_search_view',
        'page_title' => 'Search Blocked Hosts DB',
        'page_subtitle' => 'Search the Hosts DB for blocked/unblocked
IP address(es)',
        'include_stylesheets' => 'account_user_view_css',
        'include_scripts' => 'account_user_view_js'
    );

    if($new=="new"){
        if(isset($_SESSION['host_blocking_results_db'])){
            unset($_SESSION['host_blocking_results_db']);
        }

        if(isset($_SESSION['host_blocking_search_string'])){
            unset($_SESSION['host_blocking_search_string']);
        }

        if(isset($_SESSION['link_referer'])){
            unset($_SESSION['link_referer']);
        }
    }

    $temp_host_block_search_str = $this->input-
>post('host_block_search_str');
    if(!empty($_SESSION['host_blocking_results_db']) &&
!empty($_SESSION['host_blocking_search_string']) &&
empty($temp_host_block_search_str)){

        $data['host_blocking_results_db'] =
$_SESSION['host_blocking_results_db'];
        $data['host_blocking_search_string'] =
$_SESSION['host_blocking_search_string'];
        $data['link_referer'] = $_SESSION['link_referer'];

    } else {

        $this->form_validation->set_rules('host_block_search_str',
'Search', 'required|trim|xss_clean|min_length[3]');

        if ($this->form_validation->run() == TRUE){
            $temp = $this->input->post('host_block_search_str');

            if(isset($_SESSION['host_blocking_results_db'])){
                unset($_SESSION['host_blocking_results_db']);
            }
        }
    }
}

```

```

        if(isset($_SESSION['host_blocking_search_string'])){
            unset($_SESSION['host_blocking_search_string']);
        }

        if(isset($_SESSION['link_referer'])){
            unset($_SESSION['link_referer']);
        }

        if(!empty($temp)){
            $search_string = $this->input-
>post('host_block_search_str', TRUE);

            if (filter_var($search_string,
FILTER_VALIDATE_IP)) {

redirect('tech/mail_tools/view_host_status/'.urlencode(base64_encode($search_string)));

        } else {

            $search_results = $this->mail_tool_model-
>search_host($search_string);

            if(!empty($search_results)){
                $data["host_blocking_results_db"] =
                $data['host_blocking_search_string']
                $data['link_referer'] =
                'host_blocking';

                $_SESSION["host_blocking_results_db"] = $search_results;
                $_SESSION['host_blocking_search_string'] = $search_string;
                $_SESSION['link_referer'] =
                'host_blocking';

            } else {
                $this->session-
>set_flashdata('warning', "No match found.");
                redirect('tech/mail_tools/host_blocking/');
            }

        } else {
            $this->session->set_flashdata('error', "Invalid
search query.");
            redirect('tech/mail_tools/host_blocking/');
        }
    }
    // Load page variables
    $this->load->vars($data);
    $this->load->view('page_template');
}

```

```

public function view_host_status($host) {

    // Load page variables
    $data = array(
        'content' => 'host_address_view',
        'page_title' => 'View Host Log Entry',
        'page_subtitle' => 'View Specific Host IP address Log Entry',
        'include_stylesheets' => 'account_user_view_css',
        'include_scripts' => 'account_user_view_js'
    );

    $decoded_host = base64_decode(urldecode($host));
    $host_log = $this->mail_tool_model-
>get_block_host_info($decoded_host);

    // $_SESSION['link_referer'] = 'view_host_status/'.$host;
    $data['block_host'] = $decoded_host;

    if(!empty($host_log)){
        $data['results_db'] = $host_log;
    } else {
        $this->session->set_flashdata('error', "Host IP address not in
database.");

        redirect('tech/mail_tools/block_host_confirm/'.urlencode(base64_encode($dec
oded_host)));
    }

    $this->load->vars($data);
    $this->load->view('page_template');
}

public function block_host_confirm($block_host = NULL) {

    // Load page variables
    $data = array(
        'content' => 'host_ip_disable_confirm',
        'page_title' => 'Block IP Address',
        'page_subtitle' => 'Block an IP address from initiating mail
transfer',
        'include_stylesheets' => 'dilnet_account_tools_css',
        'include_scripts' => 'dilnet_account_tools_js'
    );

    #Get username from previous page (if coming from disable_form)
    if(!empty($_SESSION['block_host'])){
        $block_host = $_SESSION['block_host'];
    }

    #Get username from post input, if form is reloaded by
form_validation
    if(empty($block_host)){
        $block_host = $this->input->post('block_host',TRUE);
    }

    if(!empty($block_host)){
        $host = $this->mail_tool_model-
>get_block_host_info(base64_decode(urldecode($block_host)));
    }
}

```



```

        $this->form_validation->set_rules('disable_reason', 'Reason',
'required|xss_clean');

        if ($this->form_validation->run() == TRUE){

            $reason = $this->input->post('disable_reason');
            $_SESSION['disableinfo']['reason'] = $reason;
            $this->session->mark_as_flash('disableinfo');

            redirect('/tech/mail_tools/block_host_final/'.urlencode($block_host));
        }

        $data['block_host'] = trim($block_host);
        $data['host'] = $host;

        if(isset($_SESSION['link_referer']) &&
!empty($_SESSION['link_referer'])){
            $data['link_referer'] = $_SESSION['link_referer'];
        } else {
            $data['link_referer'] = 'host_blocking';
        }

        $this->load->vars($data);
        $this->load->view('page_template');
    }

    else {
        #TODO Keep session data for next redirect
        if (isset($_SESSION['success'])){
            $_SESSION['success'] = $_SESSION['success'];
            $this->session->mark_as_flash('success');
        }
        else {
            $_SESSION['error'] = "Could not get info of IP for
disable operation. Try again.";
            $this->session->mark_as_flash('error');
        }

        redirect('/tech/mail_tools/host_blocking');
    }
}

public function block_host_final($block_host = NULL) {
    if (!empty($block_host)){

        $decoded_host = base64_decode(urldecode($block_host));

        $account_mysql = $this->mail_tool_model-
>get_block_host_info($decoded_host);

        $reason = $_SESSION['disableinfo']['reason'];

        $admin = $this->tech_user;

        if($account_mysql['disabled']) {
            $this->session->set_flashdata('info', "IP Address
<strong>".$decoded_email."</strong> already blocked. No action taken.");

```

```

        } else {
            $reason = $reason != NULL ? $reason : "";
            $this->mail_tool_model->disable_ip($decoded_host,
$reason, $admin);
            $this->log_model->create($decoded_host, "IP Address
disabled. ".$reason, "update_mysql", $this->tech_user, $this->ip, 'info');
            $this->session->set_flashdata('warning', "IP Address
<strong>".$decoded_host."</strong> blocked.");
        }

        redirect('tech/mail_tools/view_host_status/'.urlencode(base64_encode($decod
ed_host)));
    }

    else {

        $_SESSION['error'] = "Disable IP data lost. Please try
again.";
        $this->session->mark_as_flash('error');
        redirect("/tech/mail_tools/host_blocking/");
    }
}

public function unblock_host_confirm($block_host = NULL) {

    // Load page variables
    $data = array(
        'content' => 'host_ip_disable_confirm',
        'page_title' => 'UnBlock IP Address',
        'page_subtitle' => 'Allow an IP address to initiate mail
transfer',
        'include_stylesheets' => 'dilnet_account_tools_css',
        'include_scripts' => 'dilnet_account_tools_js'
    );

    if(!empty($_SESSION['block_host'])) {
        $block_host = $_SESSION['block_host'];
    }

    if(empty($block_host)) {
        $block_host = $this->input->post('block_host', TRUE);
    }

    if(!empty($block_host)) {
        $host = $this->mail_tool_model-
>get_block_host_info(base64_decode(urldecode($block_host)));
        $this->form_validation->set_rules('disable_reason', 'Reason',
'required|xss_clean');
        if ($this->form_validation->run() == TRUE) {
            $reason = $this->input->post('disable_reason');
            $_SESSION['disableinfo']['reason'] = $reason;
            $this->session->mark_as_flash('disableinfo');
            redirect('/tech/mail_tools/unblock_host_final/'.urlencode($block_host));
        }
    }
}

```

```

    }

    $data['block_host'] = trim($block_host);
    $data['host'] = $host;

    if(isset($_SESSION['link_referer']) &&
!empty($_SESSION['link_referer'])) {
        $data['link_referer'] = $_SESSION['link_referer'];
    } else {
        $data['link_referer'] = 'host_blocking';
    }

    $this->load->vars($data);
    $this->load->view('page_template');
}

else {
    #TODO Keep session data for next redirect
    if (isset($_SESSION['success'])) {
        $_SESSION['success'] = $_SESSION['success'];
        $this->session->mark_as_flash('success');
    }
    else {
        $_SESSION['error'] = "Could not get info of IP for
disable operation. Try again.";
        $this->session->mark_as_flash('error');
    }

    redirect('/tech/mail_tools/host_blocking');
}
}

```

```

public function unblock_host_final($block_host = NULL) {
    if (!empty($block_host)) {

        $decoded_host = base64_decode(urldecode($block_host));

        $account_mysql = $this->mail_tool_model->
>get_block_host_info($decoded_host);

        $reason = $_SESSION['disableinfo']['reason'];
        $admin = $this->tech_user;

        if($account_mysql['disabled']) {
            $reason = $reason != NULL ? $reason : "";
            $this->mail_tool_model->enable_ip($decoded_host,
$reason, $admin);
            $this->log_model->create($decoded_host, "IP address
disabled. ".$reason, "update_mysql", $this->tech_user, $this->ip, 'info');
            $this->session->set_flashdata('info', "IP address
<strong>".$decoded_host."</strong> unblocked.");
        } else {
            $this->session->set_flashdata('info', "IP Address
<strong>".$decoded_host."</strong> is not blocked. No action taken.");
        }
    }
}

```

```

    }

    redirect('tech/mail_tools/view_host_status/'.urlencode(base64_encode($decoded_host)));
    }

    else {

        $_SESSION['error'] = "Disable IP data lost. Please try
again.";

        $this->session->mark_as_flash('error');
        redirect("/tech/mail_tools/host_blocking/");
    }
}
}

```

application/controllers/tech/Merge.php

```

<?php
defined('BASEPATH') OR exit('No direct script access allowed');

class Merge extends MY_Controller {

    function __construct(){
        parent::__construct();

        $this->form_validation->set_error_delimiters('<label>', '</label>');
        $this->load->model('ldap_model');
        $this->load->model('mysql_model');
        $this->load->model('log_model');

        $this->include_ldap = FALSE;

        $this->tech_user = $_SESSION['tech_user'];
        $this->ip = $this->input->ip_address();

        $this->page_vars = array(
            'content' => 'merge_view',
            'page_title' => 'Merge Account Lookup',
            'page_subtitle' => 'Search DilNet Accounts for Merging',
            'include_stylesheets' => 'merge_css',
            'include_scripts' => 'dilnet_lookup_js'
        );
    }

    public function index()
    {
        // Load page variables
        $data = $this->page_vars;

        $this->load->vars($data);
        $this->load->view('page_template');
    }
}

```

```

public function merge_confirm (){

    if(isset($_SESSION['merge_account_info'])){

        $data = array(
            'content' => 'merge_accounts_confirm',
            'page_title' => 'Merge DilNet Accounts',
            'page_subtitle' => 'Confirm selected merge attributes',
            'include_stylesheets' => 'merge_css',
            'include_scripts' => 'dilnet_account_tools_js'
        );

        $merge_info = $_SESSION['merge_account_info'];

        $info_diff = array();

        $info_diff['username'] = $merge_info['opt_username'];

        if($info_diff['username'] == $merge_info['username1']) {
            $info_diff['delete_username'] =
$merge_info['username2'];
        } else if($info_diff['username'] == $merge_info['username2'])
        {
            $info_diff['delete_username'] =
$merge_info['username1'];
        }

        $account_mysql = $this->mysql_model-
>get_user_account($merge_info['opt_fullname']);

        $info_diff['firstname'] = $account_mysql['firstname'];
        $info_diff['middlename'] = $account_mysql['middlename'];
        $info_diff['lastname'] = $account_mysql['lastname'];
        $info_diff['fullname'] = ".$info_diff['firstname'].
".$info_diff['lastname'];

        $info_diff['mail'] = $merge_info['opt_mail'];
        $info_diff['mailalternateaddress'] = $merge_info['mail_alt'];

        $info_diff['employeetype'] = $merge_info['opt_employeetype'];
        $info_diff['employeenumber'] =
$merge_info['opt_employeenumber'];
        $info_diff['studentnumber'] =
$merge_info['opt_studentnumber'];
        $info_diff['ou'] = $merge_info['opt_unit'];
        $info_diff['departmentnumber'] =
$merge_info['opt_department'];
        $info_diff['designation'] = $merge_info['opt_designation'];

        if(!empty($info_diff)){
            $data['info_diff'] = $info_diff;
            $this->load->vars($data);
            $this->load->view('page_template');
        } else {
            $_SESSION['warning'] = "No changes were made.";
            $this->session->mark_as_flash('warning');
            redirect('tech/merge');
        }

    }
}

```

```

    } else {
        $_SESSION['error'] = "Session data lost. Try again.";
        $this->session->mark_as_flash('error');
        redirect('tech/merge');
    }
}

public function merge_accounts_final() {

    $entry_diff = $this->input->post(NULL, TRUE);
    /*
    print "<pre>";
    print_r($entry_diff);
    print "</pre>";
    /**/

    if(isset($entry_diff['username']))){
        // $new_uid = $entry_diff['username'];
        $mysql_diff['username'] = $entry_diff['username'];
        $account_ldap = $this->ldap_model-
>get_user_account($entry_diff['username']);
        $account_ldap_delete = $this->ldap_model-
>get_user_account($entry_diff['delete_username']);
        // $account_mysql = $this->mysql_model-
>get_user_account($entry_diff['username']);
    }

    if(isset($entry_diff['mail']))){
        $ldap_diff['mail'] = $entry_diff['mail'];
        $mysql_diff['email'] = $entry_diff['mail'];
    }

    if(isset($entry_diff['mailalternateaddress']))){
        $ldap_diff['mailalternateaddress'] =
$entry_diff['mailalternateaddress'];
    }

    if(isset($entry_diff['fullname']))){
        $ldap_diff['cn'] = $entry_diff['fullname'];
    }

    if(isset($entry_diff['firstname']))){
        $mysql_diff['firstname'] = $entry_diff['firstname'];
    }

    if(isset($entry_diff['middlename']))){
        if($entry_diff['middlename'] == 'none') {
            $mysql_diff['middlename'] = NULL;
        } else {
            $mysql_diff['middlename'] = $entry_diff['middlename'];
        }
    }

    if(isset($entry_diff['lastname']))){
        $ldap_diff['sn'] = $entry_diff['lastname'];
        $mysql_diff['lastname'] = $entry_diff['lastname'];
    }

    /**/

```

```

        if($account_ldap['mailquotasize'] >=
$account_ldap_delete['mailquotasize']){
            $ldap_diff['mailquotasize'] = $account_ldap['mailquotasize'];
        } else {
            $ldap_diff['mailquotasize'] =
$account_ldap_delete['mailquotasize'];
        }
        /**/

        if(isset($entry_diff['ou'])){
            $ldap_diff['ou'] = $entry_diff['ou'];
            $mysql_diff['unit_id'] = $entry_diff['ou'];
        }

        if(isset($entry_diff['departmentnumber'])){
            $ldap_diff['departmentnumber'] =
$entry_diff['departmentnumber'];
        }

        if(isset($entry_diff['employeeetype'])){
            if($entry_diff['employeeetype'] == "STUD") {
                $ldap_diff['employeeetype'] = "";
            } else {
                $ldap_diff['employeeetype'] =
$entry_diff['employeeetype'];
            }
            $mysql_diff['type'] = $entry_diff['employeeetype'];
        }

        if(isset($entry_diff['studentnumber'])){
            $ldap_diff['studentnumber'] = $entry_diff['studentnumber'];
            $mysql_diff['id_number'] = $mysql_diff['studno'] =
$entry_diff['studentnumber'];
        }

        if(isset($entry_diff['employeenumber'])){
            $ldap_diff['employeenumber'] = $entry_diff['employeenumber'];
            $mysql_diff['id_number'] = $mysql_diff['empno'] =
$entry_diff['employeenumber'];
        }

        if(isset($entry_diff['designation'])){
            $ldap_diff['title'] = $entry_diff['designation'];
            $mysql_diff['designation'] = $entry_diff['designation'];
        }

        $update_ldap = $update_mysql = $deleted_ldap_entry =
$deleted_mysql_entry = FALSE;
        /**/

        /*
        print "<pre>";
        print_r($ldap_diff);
        print_r($mysql_diff);
        print "</pre>";
        /**/

        /**/

```

```

        if(isset($ldap_diff)){
            # Change null values to empty arrays
            array_walk($ldap_diff, function(&$n) { if(($n == NULL) || $n
== "") { $n = array(); } });
            $update_ldap = $this->ldap_model-
>set_ldapentry($entry_diff['username'], $account_ldap['dn'], $ldap_diff);

            $ldap_keys = array_keys($ldap_diff);
            $ldap_keys = implode(",", $ldap_keys);

            // print_r($update_ldap);
            // print_r($ldap_keys);
        }

        if(isset($mysql_diff)){
            $update_mysql = $this->mysql_model-
>update_mysql_entry($mysql_diff, $entry_diff['username']);

            $mysql_keys = array_keys($mysql_diff);
            $mysql_keys = implode(",", $mysql_keys);
        }
        /**/
        /**/
        if(isset($entry_diff['delete_username']) &&
!empty($entry_diff['delete_username'])){
            $deleted_ldap_entry = $this->ldap_model-
>delete_ldap_user($entry_diff['delete_username']);

            $where = array('username' => $entry_diff['delete_username']);
            $deleted_mysql_entry = $this->mysql_model-
>delete_mysql_entry($where);
        }
        /**/

        if( $update_ldap || $update_mysql || $deleted_ldap_entry ||
$deleted_mysql_entry){
            if(isset($ldap_keys)){
                $this->log_model->create($entry_diff['username'], "LDAP
fields updated: ".$ldap_keys, "update_ldap", $this->tech_user, $this->ip,
'info');
            }
            if(isset($mysql_keys)){
                $this->log_model->create($entry_diff['username'],
"MySQL fields updated: ".$mysql_keys, "update_mysql", $this->tech_user, $this-
>ip, 'info');
            }

            if(isset($deleted_user)){
                $this->log_model-
>create($entry_diff['delete_username'], "LDAP account deleted.", "update_ldap",
$this->tech_user, $this->ip, 'info');
            }

            $success_msg = "";

            if($update_ldap) {
                $success_msg .= "Account
<strong>".strtoupper($uid)."</strong> successfullly updated in LDAP.<br>";

```



```

    }

    if($update_mysql) {
        $success_msg .= "Account
<strong>".strtoupper($uid)."</strong> successfullly updated in MySQL.<br>";
    }

    if($deleted_ldap_entry) {
        $success_msg .= "Account
".$entry_diff['delete_username']." successfullly deleted in LDAP.<br>";
    }

    if($deleted_mysql_entry) {
        $success_msg .= "Account
".$entry_diff['delete_username']." successfullly deleted in MySQL.<br>";
    }

    $_SESSION['success'] = $success_msg;
    unset($_SESSION['merge_db']);
    unset($_SESSION['merge_account_info']);

    $this->session->mark_as_flash('success');

    redirect('tech/dilnetacc_tools/view_account/'.$entry_diff['username']);

}
else {
    $_SESSION['error'] = "Something went wrong. Account
<strong>".strtoupper($entry_diff['username'])."</strong> was not updated.";
    $this->session->mark_as_flash('error');
    redirect('tech/merge');
}
/**/
}

public function merge_accounts_form() {

    // Load page variables
    $data = array(
        'content' => 'merge_accounts_view',
        'page_title' => 'Merge DilNet Accounts',
        'page_subtitle' => 'Merge two DilNet Accounts into one',
        'include_stylesheets' => 'account_user_view_css',
        'include_scripts' => 'account_user_view_js'
    );

    $this->form_validation->set_message('required', 'Please select an
option');
    $this->form_validation->set_rules('opt_username',
'Username','required');
    $this->form_validation->set_rules('opt_fullname', 'Full
Name','required');
    $this->form_validation->set_rules('opt_mail', 'Primary
Email','required');
    $this->form_validation->set_rules('mail_alt[]', 'Alternate
Email','required');
    $this->form_validation->set_rules('opt_employeetype', 'Employee
Type','required');

```

```

        $this->form_validation->set_rules('opt_employeenumber', 'Employee
Number','required');
        $this->form_validation->set_rules('opt_studentnumber', 'Student
Number','required');
        $this->form_validation->set_rules('opt_unit', 'Unit','required');
        $this->form_validation->set_rules('opt_department',
'Department','required');
        $this->form_validation->set_rules('opt_designation',
'Designation','required');

        if (isset($_SESSION['merge_db']) && count($_SESSION['merge_db']) ==
2){
            ##ADVANCED VIEW: To display LDAP vs MySQL information
            $data['account_keys'] = $account_keys = array('Username',
'Full Name', 'Mail', 'Mail Alt', 'Employee Number', 'Employee Type', 'Student
Number', 'Unit', 'Department', 'Designation');

            $_SESSION['merge_db'] = array_values($_SESSION['merge_db']);

            /*
            print "<pre>";
            print_r($_SESSION['merge_db']);
            print "</pre>";
            /**/

            $first_account = $this->ldap_model-
>get_ldap_info($_SESSION['merge_db'][0][0]["uid"][0]);
            $second_account = $this->ldap_model-
>get_ldap_info($_SESSION['merge_db'][1][0]["uid"][0]);

            foreach($account_keys as $key => $value){
                if($value == 'Username'){
                    $first_account_view[$value] =
$first_account[0]['uid'][0];
                    $second_account_view[$value] =
$second_account[0]['uid'][0];
                }

                if($value == "Full Name"){
                    $first_account_view[$value] =
"".@$first_account[0]['cn'][0]."";
                    $second_account_view[$value] =
"".@$second_account[0]['cn'][0]."";
                }

                if($value == "Unit"){
                    $first_account_view[$value] =
"".@$first_account[0]['ou'][0]."";
                    $second_account_view[$value] =
"".@$second_account[0]['ou'][0]."";
                }

                if($value == "Student Number"){
                    $first_account_view[$value] =
@$first_account[0]['studentnumber'][0];

```

```

        $second_account_view[$value] =
@$second_account[0]['studentnumber'][0];
    }

    if($value == "Employee Number"){
        $first_account_view[$value] =
@$first_account[0]['employeenumber'][0];
        $second_account_view[$value] =
@$second_account[0]['employeenumber'][0];
    }

    if($value == "Title"){
        $first_account_view[$value] =
@$first_account[0]['title'][0];
        $second_account_view[$value] =
@$second_account[0]['title'][0];
    }

    if($value == "Mail"){
        $first_account_view[$value] =
$first_account[0]['mail'][0];
        $second_account_view[$value] =
$second_account[0]['mail'][0];
    }

    if($value == "Mail Alt"){
        $first_account_view[$value] =
$first_account[0]['mailalternateaddress'];
        $second_account_view[$value] =
$second_account[0]['mailalternateaddress'];
    }

    if($value == "Employee Type"){
        $first_account_view[$value] =
@$first_account[0]['employeetype'][0];
        $second_account_view[$value] =
@$second_account[0]['employeetype'][0];
    }

    if($value == "Department"){
        $first_account_view[$value] =
@$first_account[0]['departmentnumber'][0];
        $second_account_view[$value] =
@$second_account[0]['departmentnumber'][0];
    }

    if($value == "Designation"){
        $first_account_view[$value] =
@$first_account[0]['title'][0];
        $second_account_view[$value] =
@$second_account[0]['title'][0];
    }

}
$data['first_account_view'] = $first_account_view;
$data['second_account_view'] = $second_account_view;
if($this->form_validation->run() == TRUE) {

```

```

        $_SESSION['merge_account_info'] = $this->input-
>post(NULL,TRUE);
        redirect('tech/merge/merge_confirm');

    }
    } else {
        redirect('tech/dilnet_lookup');
    }

    $this->load->vars($data);
    $this->load->view('page_template');
}

public function merge_add($username){
    // Load page variables
    $data = $this->page_vars;

    if($username=='byron'){
        $_SESSION['error'] = "Can't add superuser account for
merging.";
        $this->session->mark_as_flash('error');
        redirect('/tech/dilnet_lookup');
    }

    if(!empty($_SESSION['merge_search_results_db']) &&
!empty($_SESSION['merge_search_string'])){

        $data['merge_search_results_db'] =
$_SESSION['merge_search_results_db'];
        $data['merge_search_string'] =
$_SESSION['merge_search_string'];
        // $data['link_referer'] = $_SESSION['link_referer'];

    }

    if(!isset($_SESSION['merge_db'])){
        $_SESSION['merge_db'] = array();
        array_push($_SESSION['merge_db'], $this->ldap_model-
>get_ldap_info($username));
        // $this->session->set_flashdata('info', "Added
"". $username. "" to merge list.", "refresh");
    } else {
        if(count($_SESSION['merge_db']) == 2){
            $this->session->set_flashdata('danger', "Maximum
number of accounts reached. Can't add anymore.", "refresh");
        } else {
            if(!$this->findElementWithValue($_SESSION['merge_db'],
'uid', $username)){
                array_push($_SESSION['merge_db'], $this-
>ldap_model->get_ldap_info($username));
                // $this->session->set_flashdata('info', "Added
"". $username. "" to merge list.", "refresh");
            }
        }
    }
    $this->load->vars($data);
    $this->load->view('page_template');
}

```

```

public function merge_remove($username){
    // Load page variables
    $data = $this->page_vars;

    if(!empty($_SESSION['merge_search_results_db']) &&
    !empty($_SESSION['merge_search_string'])){

        $data['merge_search_results_db'] =
$_SESSION['merge_search_results_db'];
        $data['merge_search_string'] =
$_SESSION['merge_search_string'];
        // $data['link_referer'] = $_SESSION['link_referer'];

    }

    if(!isset($_SESSION['merge_db'])){
        //
        // show error here
        //
    } else {
        $_SESSION['merge_db'] = $this-
>removeElementWithValue($_SESSION['merge_db'],'uid',$username);
        // $this->session->set_flashdata('default', "Removed
'".$username.'" from merge list.", "refresh");
    }
    /*
    print "<pre>";
    print_r($_SESSION['merge_db']);
    print "</pre>";
    /**/
    $this->load->vars($data);
    $this->load->view('page_template');
}

public function merge_clear_list(){
    // Load page variables
    $data = $this->page_vars;
    if(!isset($_SESSION['merge_db'])){
        //
        // show error here
        //
    } else {
        unset($_SESSION['merge_db']);
        // $this->session->set_flashdata('default', "Removed
'".$username.'" from merge list.", "refresh");
    }
    /*
    print "<pre>";
    print_r($_SESSION['merge_db']);
    print "</pre>";
    /**/
    $this->load->vars($data);
    $this->load->view('page_template');
}

public function merge_search_form($new=NULL) {
    // Load page variables
    $data = $this->page_vars;

```

```

if($new=="new"){
    if(isset($_SESSION['merge_search_results_db'])){
        unset($_SESSION['merge_search_results_db']);
    }

    if(isset($_SESSION['merge_search_string'])){
        unset($_SESSION['merge_search_string']);
    }

    if(isset($_SESSION['merge_db'])){
        unset($_SESSION['merge_db']);
    }
}

$temp_merge_search_str = $this->input->post('merge_str');
if(!empty($_SESSION['merge_search_results_db']) &&
!empty($_SESSION['merge_search_string']) && empty($temp_merge_search_str)){

    $data['merge_search_results_db'] =
$_SESSION['merge_search_results_db'];
    $data['merge_search_string'] =
$_SESSION['merge_search_string'];
    // $data['link_referer'] = $_SESSION['link_referer'];

} else {
    $this->form_validation->set_rules('merge_str', 'Search',
'required|trim|xss_clean|min_length[3]');

    if ($this->form_validation->run() == TRUE){
        $search_string = $this->input->post('merge_str');

        if(isset($_SESSION['merge_search_results_db'])){
            unset($_SESSION['merge_search_results_db']);
        }

        if(isset($_SESSION['merge_search_string'])){
            unset($_SESSION['merge_search_string']);
        }

        #TODO: Sanitize search string
        if($search_string != "" || $search_string != NULL) {
            $stringArray = explode(" ", $search_string);
            $count_query = count($stringArray);
            $data["merge_search_string"] = $search_string;
            if(1==$count_query){ // only one keyword
                if(strpos($search_string,"@upd.edu.ph")){ //
                    // email match
                    $pos =
                    strpos($search_string,"@upd.edu.ph");
                    $email = substr($search_string, 0, $pos);
                    if(strpos($email,".)){ // might be a
                        // $data["results_db"] = $this-
                        $temp = $this->ldap_model-
                        >mysql_model->search_email($email);
                        $data["merge_search_results_db"] =
                        >search_mail_addresses($email);
                        $this->mysql_model->search_username($temp[0]["uid"][0]);
                    }
                }
            }
        }
    }
}

```

```

        } else {
            // username match
            // $data["results_db"] = $this-
>mysql_model->search_username($email);
            $temp = $this->ldap_model-
>search_mail_addresses($email);
            $data["merge_search_results_db"] =
$this->mysql_model->search_username($temp[0]["uid"][0]);
        }
    }

    // primary email match
    else if(strpos($search_string, ".")){ // might be
a primary email
        // $data["results_db"] = $this-
>mysql_model->search_email($search_string);
        $temp = $this->ldap_model-
>search_mail_addresses($search_string);
        $data["merge_search_results_db"] = $this-
>mysql_model->search_username($temp[0]["uid"][0]);
    }

    else if(strpos($search_string, "-")){ // might be
a student number
        // student number match
        $data["merge_search_results_db"] = $this-
>mysql_model->search_substring_number($search_string);
    }

    // exact student number match
    else if(strpos($search_string, "-")){ // might be
a student number
        if($this-
>_is_idno_studentno($search_string)){
            $search_result_sn = $this-
>mysql_model->search_substring_number($search_string);
            if(empty($search_result_sn)){ //
might be stored in LDAP
                $temp = $this->ldap_model-
>search_substring_number($search_string);
                $search_result_sn = $this-
>mysql_model->search_username($temp[0]["uid"][0]);
            }
            $data["merge_search_results_db"] =
$search_result_sn;
        } else { // might be partial
            $search_result_sn_partial = $this-
>mysql_model->search_substring_number($search_string);
            if(!empty($search_result_sn_partial)){
                $data["merge_search_results_db"] = $search_result_sn_partial;
            }
        }
    }
}

```

```

        else if(is_numeric($search_string)){
            // employee number match
            $len = strlen($search_string);
            if($len==9){
                $employee_number_search_string =
"0".$search_string;
            } else if($len==10){
                $employee_number_search_string =
$search_string;
            }
            $data["merge_search_results_db"] = $this->mysql_model->search_substring_number($employee_number_search_string);
        }
        else {
            // try first a username search
            // $data["results_db"] = $this->mysql_model->search_username($search_string);
            $temp = $this->ldap_model->search_mail_addresses($search_string);
            $data["merge_search_results_db"] = $this->mysql_model->search_username($temp[0]["uid"][0]);
            if(empty($data["merge_search_results_db"])){
                $data["merge_search_results_db"] = $this->mysql_model->search_substring($search_string);
            }
        }
        } else if($count_query>1){ // more than one
            search keywords
            $data["merge_search_results_db"] = $this->mysql_model->search_substring($search_string);
        }
        if(isset($data['merge_search_results_db']) &&
!empty($data['merge_search_results_db'])){
            $_SESSION['merge_search_results_db'] =
$data['merge_search_results_db'];
            $_SESSION['merge_search_string'] =
$data['merge_search_string'];
        } else {
            $this->session->set_flashdata('warning', "No
match found.");
            redirect('tech/merge/');
        }
    }

}

$this->load->vars($data);
$this->load->view('page_template');
}

```



```

    public function removeElementWithValue($array, $key, $value){
        foreach($array as $subKey => $subArray){
            if($subArray[0][$key][0] == $value){
                unset($array[$subKey]);
            }
        }
        return $array;
    }

    public function findElementWithValue($array, $key, $value){
        foreach($array as $subKey => $subArray){
            if($subArray[0][$key][0] == $value){
                return TRUE;
            }
        }
        return FALSE;
    }
}

```

application/controllers/tech/Print_account.php

```

<?php
defined('BASEPATH') OR exit('No direct script access allowed');

use Mike42\Escpos\EscposImage;
use Mike42\Escpos\PrintConnectors\NetworkPrintConnector;
use Mike42\Escpos\PrintConnectors\CupsPrintConnector;
use Mike42\Escpos\Printer;

class Print_account extends MY_Controller {

    function __construct() {
        parent::__construct();

        $this->load->model('ldap_model');
        $this->load->model('log_model');

    }
    function index(){
        redirect('tech/dilnet_lookup/name_search_form/new');
    }

    function confirm_print($username = NULL){

        $print_type=null;

        $account_session_array = array();
        $data = array(
            'content' => 'print_confirm',
            'page_title' => 'Confirm Print Information',
            'page_subtitle' => null,
            'include_stylesheets' => 'print_confirm_css',

```

```

        'include_scripts' => 'print_confirm_js'
    );

    $data['ldap_print'] = array();

    // print "<pre>";
    // print_r($_SESSION['reset_print']);
    // print "</pre>";

    # Get session data
    if(isset($_SESSION['reset_print'])){
        $reset_user = $_SESSION['reset_print'];
        $username = $reset_user['username'];
        $print_type = 'reset';
    }

    else if (isset($_SESSION['create_account_print'])){
        $print_type='new';
        $new_account = $_SESSION['create_account_print'];
        $username = $new_account['username'];
    }

    if($username != NULL){
        if($print_type == NULL){ $print_type = 'info'; }

        $ldap_account = $this->ldap_model-
>get_user_account($username);
        if(isset($ldap_account)){
            $type = 'ldap';
            $page_type = 'print';
            $data['print_type'] = $print_type;

            switch($print_type){
                case 'reset':
                    $data['ldap_print'] = $reset_user;
                    break;
                case 'new':
                    $data['ldap_print'] = array(
                        'Username' => $this-
>_format_array_model2ctrl($ldap_account, 'uid', $type, $page_type),
                        'Password' => $new_account['password'],
                        'Name' => $this-
>_format_array_model2ctrl($ldap_account, 'cn', $type, $page_type),
                        'Mail' => $this-
>_format_array_model2ctrl($ldap_account, 'mail', $type, $page_type),
                        'Mail Alt' => $this-
>_format_array_model2ctrl($ldap_account, 'mailalternateaddress', $type,
$page_type),
                        'Emp Type' => $this-
>_format_array_model2ctrl($ldap_account, 'employeetype', $type, $page_type)
                    );
                    break;
                case 'info':
                    $data['ldap_print'] = array(
                        'Username' => $this-
>_format_array_model2ctrl($ldap_account, 'uid', $type, $page_type),
                        'Name' => $this-
>_format_array_model2ctrl($ldap_account, 'cn', $type, $page_type),

```

```

        'Mail' => $this->_format_array_model2ctrl($ldap_account, 'mail', $type, $page_type),
        'Mail Alt' => $this->_format_array_model2ctrl($ldap_account, 'mailalternateaddress', $type, $page_type),
        'Unit' => $this->_format_array_model2ctrl($ldap_account, 'ou', $type, $page_type),
        'Dept' => $this->_format_array_model2ctrl($ldap_account, 'departmentnumber', $type, $page_type),
        'Designation' => $this->_format_array_model2ctrl($ldap_account, 'title', $type, $page_type),
        'Appt Period' => $this->_format_array_model2ctrl($ldap_account, 'description', $type, $page_type),
        'Emp Type' => $this->_format_array_model2ctrl($ldap_account, 'employeetype', $type, $page_type),
        'Emp No' => $this->_format_array_model2ctrl($ldap_account, 'employeenumber', $type, $page_type),
        'Stud No' => $this->_format_array_model2ctrl($ldap_account, 'studentnumber', $type, $page_type)
    );
    break;
}

$data['username'] = $username;
$this->load->vars($data);
$this->load->view('page_template');
}
#ELSE Walang laman ang ldap account

}

else {
    $this->session->mark_as_flash('error');
    redirect('tech/dilnet_lookup');
}

}

function confirm_multiple_print(){
    $print_type=null;
    $account_session_array = array();
    $data = array(
        'content' => 'print_multiple_confirm',
        'page_title' => 'Confirm Print Information',
        'page_subtitle' => null,
        'include_stylesheets' => 'print_confirm_css',
        'include_scripts' => 'print_confirm_js'
    );

    $data['ldap_prints'] = array();
    $ldap_prints = array();

    // print "<pre>";
    // print_r($this->session->mark_as_flash('error'));
    // print "</pre>";

    # Get session data

```

```

        if (isset($_SESSION['activated_accounts'])) {
            $print_type = 'new';
            $new_accounts = $_SESSION['activated_accounts'];

            // print "<pre>";
            // print_r($new_accounts);
            // print "</pre>";

            foreach($new_accounts as $new_account) {
                $username = $new_account[0];
                if($username != NULL)
                {
                    $ldap_account = $this->ldap_model->
>get_user_account($username);
                    if(isset($ldap_account)) {
                        $type = 'ldap';
                        $page_type = 'print';
                        $data['print_type'] = $print_type;

                        $data['ldap_prints'][] = array(
                            'Username' => $this->
>_format_array_model2ctrl($ldap_account, 'uid', $type, $page_type),
                            'Password' => $new_account[1],
                            'Name' => $this->
>_format_array_model2ctrl($ldap_account, 'cn', $type, $page_type),
                            'Mail' => $this->
>_format_array_model2ctrl($ldap_account, 'mail', $type, $page_type),
                            'Mail_Alt' => $this->
>_format_array_model2ctrl($ldap_account, 'mailalternateaddress', $type,
$page_type),
                            'Emp_Type' => $this->
>_format_array_model2ctrl($ldap_account, 'employeetype', $type, $page_type)
                        );
                    }
                }
            }
            // print "<pre>";
            // print_r($data['ldap_prints']);
            // print "</pre>";
            $this->load->vars($data);
            $this->load->view('page_template');
        }
        else {
            $_SESSION['error'] = "You tried to print something but the
session was lost.";
            $this->session->mark_as_flash('error');
            // redirect('tech/create_user');
        }
    }

    # $printer = tech, staff
    function send_to_printer() {

        #TODO Add print type

        $ldap_print = array();
        $opt_changepass = $this->input->post('opt_changepass');
    }

```

```

$opt_facproxy = $this->input->post('opt_facproxy');
$opt_services = $this->input->post('opt_services');
$opt_helpdesk = $this->input->post('opt_helpdesk');
$opt_crs = $this->input->post('opt_crs');
$opt_library = $this->input->post('opt_library');

$printer_type = $this->input->post('printer_opt');
$print_type = $this->input->post('print_type');
$username = $this->input->post('username');
$ldap_print = $this->input->post(NULL, TRUE);

$arrlist =
array('printer_opt', 'print_type', 'opt_changepass', 'opt_facproxy', 'opt_services', 'opt_helpdesk', 'opt_crs', 'opt_library');
foreach($arrlist as $arr){
    unset($ldap_print[$arr]);
}

if($print_type!='reset'){
    unset($ldap_print['username']);
}

#Sanitize ldap_print
#TODO Optimize
foreach ($ldap_print as $key => $value){
    if(strpos($key, '_') !== false){
        $newkey = str_replace("_", " ", $key);
        $ldap_print[$newkey] = $value;
        unset($ldap_print[$key]);
    }
}

// print_r($ldap_print);

#Default printer port for network-enabled printers
$printer_port = 9100;

switch($printer_type){
    case 'tech-thermal':
        $printer_ip = "10.40.2.207";
        break;
    case 'tech-ribbon':
        $printer_ip = "10.40.3.203";
        $printer_port = 80;
        break;
    case 'admin-thermal':
        $printer_ip = "10.40.253.207";
        break;
    default:
        $_SESSION['error'] = "Printer not set.";
        $this->session->mark_as_flash('error');
        redirect('tech/dilnet_lookup/');
}

if(!empty($ldap_print)){
    // $this->log_model->create($username, "Printing account
    credentials on ".$printer_ip, "activation", $_SESSION['tech_user'], $this->input-
    >ip_address(), 'debug');
}

```

```

        # Check printer if up
        $check_printer = @fsockopen($printer_ip, $printer_port,
$errno, $errstr, 3);
        if ($check_printer === false) {
            $_SESSION['error'] = "Cannot connect to printer.
".ucfirst($printer_type)." ".$printer_ip." Printer may be offline.";
            $this->session->mark_as_flash('error');

            redirect('tech/dilnetacc_tools/view_account/' . $username);
        }

        else {
            if($printer_type == 'tech-ribbon'){
                $connector = new CupsPrintConnector('TMU220');
            }
            else { $connector = new
NetworkPrintConnector($printer_ip, $printer_port); }
            // $connector = new NetworkPrintConnector($printer_ip,
80);

            $printer = new Printer($connector);

            # Do not print image for TMU220, not supported
            if($printer_type != 'tech-ribbon'){
                $printer-
>setJustification(Printer::JUSTIFY_CENTER);
                $img =
EscposImage::Load($_SERVER['DOCUMENT_ROOT']. "/techaccounts/assets/dilnetlogo.png")
;

                $printer->graphics($img);
                $printer->feed(1);
            }
            $printer->setEmphasis(TRUE);
            $printer->setTextSize(2,3);
            $printer->text("DILNET ACCOUNT \n");
            $printer->setTextSize(2,2);

            switch($print_type){
                case 'reset':
                    $printer->text("Reset Password \n");
                    break;
                case 'new':
                    $printer->text("New Account \n");
                    break;
                case 'info':
                    $printer->text("Account Information\n");
                    break;
            }

            $printer->initialize();
            $printer->feed(2);

            foreach ($ldap_print as $key => $value){
                if($key == 'password' || $key == 'username'){
                    $printer->setEmphasis(TRUE);
                    $printer->text($key.": ");
                    $printer->setEmphasis(FALSE);
                    $printer->setTextSize(2,1);
                    $printer->text($value."\n");
                }
            }
        }
    }
}

```

```

        $printer->initialize();
    }

    else if(is_array($value)){
        foreach(array_slice($value, 1) as $a) {
            $printer->setEmphasis(TRUE);
            $printer->text($key);
            $printer->setEmphasis(FALSE);
            $printer->text(": ".$a."\n");
        }
    }
    else {
        $printer->setEmphasis(TRUE);
        $printer->text($key);
        $printer->setEmphasis(FALSE);
        $printer->text(": ".$value."\n");
    }
}
$printer->feed(2);
if($printer_type == 'reset' || $opt_changepass){
    $printer->text("Please change your password
immediately at \n");

    $printer->setEmphasis(TRUE);
    $printer->setTextSize(1,2);
    $printer->text('https://accounts.upd.edu.ph');
    $printer->feed(2);
    $printer->setTextSize(1,1);
    $printer->text("Or scan this code to go to the
website: ");

    $printer->feed(2);
    $printer-
>setJustification(Printer::JUSTIFY_CENTER);
    $printer-
>qrCode("https://accounts.upd.edu.ph",Printer::QR_ECLEVEL_M,5);
    $printer-
>setJustification(Printer::JUSTIFY_LEFT);
    $printer->feed(2);
}

#Print if Faculty
if($opt_facproxy){
    $printer->setEmphasis(TRUE);
    $printer->text("FACULTY PROXY SETTINGS \n");
    $printer->setEmphasis(FALSE);
    $printer->text("You may access the unrestricted
proxy for the faculty by using the following settings:\n");
    $printer->text("Host:
facultyproxy.upd.edu.ph\nPort:8080 \n");
    $printer->feed(1);
}
if($opt_crs){
    $printer->setEmphasis(TRUE);
    $printer->text("CRS FACULTY ROLE \n");
    $printer->setEmphasis(FALSE);
    $printer->text("We have forwarded your
information to CRS for tagging. Average processing time is 1-2 days. You may
follow up this request by calling \n");
    $printer->setEmphasis(TRUE);
    $printer->text("981-8500 local 4560. \n");
}

```



```

        $printer->setEmphasis(FALSE);
        $printer->text("helpdesk@upd.edu.ph\n");
        $printer->setEmphasis(TRUE);
        $printer->text("Phone: ");
        $printer->setEmphasis(FALSE);
        $printer->text("981-8500 local 2050\n");
        $printer->setEmphasis(TRUE);
        $printer->text("Website: ");
        $printer->setEmphasis(FALSE);
        $printer->text("http://dilnet.upd.edu.ph\n");
        $printer->setEmphasis(TRUE);
        $printer->text("Facebook: ");
        $printer->setEmphasis(FALSE);
        $printer-
>text("http://www.facebook.com/updilnet\n");
        $printer->setEmphasis(TRUE);
        $printer->text("Twitter: ");
        $printer->setEmphasis(FALSE);
        $printer-
>text("http://www.twitter.com/dilnet\n");
        $printer->feed(2);
    }

    $date = date("Y-m-d H:i:s");
    $printer->text("Date/Time Generated: ".$date."\n");
    $printer->feed(1);
    $printer -> cut();
    $printer -> close();

    switch($print_type){
        case 'reset':
            $_SESSION['success'] = "Password reset successful
for <strong>".strtoupper($username)."</strong>.";
            break;
        case 'new':
            $_SESSION['success'] = "New user created:
<strong>".strtoupper($username)."</strong>.";
            break;
        case 'info':
            $_SESSION['success'] = "Account information
Printed for <strong>".strtoupper($username)."</strong>.";
            break;
    }

    $this->session->mark_as_flash('success');

    redirect('tech/dilnetacc_tools/view_account/'.$username);
    } //else.if printer is up
    } //else.if ldap_print is empty
else {
    $_SESSION['error'] = "You tried to print something but the data was
lost. Sorry.";
    $this->session->mark_as_flash('error');
    redirect('tech/dilnet_lookup');
}
}

function send_multiple_to_printer(){

```

#TODO Add print type

```
$ldap_prints = array();
$opt_changepass = $this->input->post('opt_changepass');
$opt_facproxy = $this->input->post('opt_facproxy');
$opt_services = $this->input->post('opt_services');
$opt_helpdesk = $this->input->post('opt_helpdesk');
$opt_crs = $this->input->post('opt_crs');
$opt_library = $this->input->post('opt_library');
```

```
$printer_type = $this->input->post('printer_opt');
$print_type = $this->input->post('print_type');
$ldap_prints = $this->input->post(null, TRUE);
```

```
// print "<pre>";
// print_r($ldap_prints);
// print "</pre>";
```

```
$arrlist =
array('printer_opt', 'print_type', 'opt_changepass', 'opt_facproxy', 'opt_services', 'opt_helpdesk', 'opt_crs', 'opt_library');
foreach($arrlist as $arr){
    unset($ldap_prints[$arr]);
}
```

```
/**
```

```
foreach($ldap_prints as $ldap_print){
    #Sanitize ldap_print
    #TODO Optimize
    foreach ($ldap_print as $key => $value){
        if(strpos($key, '_') !== false){
            $newkey = str_replace("_", " ", $key);
            $ldap_print[$newkey] = $value;
            unset($ldap_print[$key]);
        }
    }
}
```

```
// print_r($ldap_print);
```

```
#Default printer port for network-enabled printers
$printer_port = 9100;
```

```
switch($printer_type){
    case 'tech-thermal':
        $printer_ip = "10.40.2.207";
        break;
    case 'tech-ribbon':
        $printer_ip = "10.40.3.203";
        $printer_port = 80;
        break;
    case 'admin-thermal':
        $printer_ip = "10.40.253.207";
        break;
    default:
        $_SESSION['error'] = "Printer not set.";
        $this->session->mark_as_flash('error');
        redirect('tech/dilnet_lookup/');
}
```

```

        if(!empty($ldap_print)){

            //      $this->log_model->create($username, "Printing account
credentials on ".$printer_ip, "activation", $_SESSION['tech_user'], $this->input-
>ip_address(), 'debug');

            # Check printer if up
            $check_printer = @fsockopen($printer_ip, $printer_port,
$errorno, $errstr, 3);
            if ($check_printer === false) {
                $_SESSION['error'] = "Cannot connect to printer.
".ucfirst($printer_type)." ".$printer_ip." Printer may be offline.";
                $this->session->mark_as_flash('error');
                redirect('tech/dilnet_lookup');
            }

            else {
                if($printer_type == 'tech-ribbon'){
                    $connector = new CupsPrintConnector('TMU220');
                }
                else { $connector = new
NetworkPrintConnector($printer_ip, $printer_port); }
                // $connector = new NetworkPrintConnector($printer_ip,
80);

                $printer = new Printer($connector);

                # Do not print image for TMU220, not supported
                if($printer_type != 'tech-ribbon'){
                    $printer-
>setJustification(Printer::JUSTIFY_CENTER);
                    $img =
EscposImage::Load($_SERVER['DOCUMENT_ROOT']."/techaccounts/assets/dilnetlogo.png")
;
                    $printer->graphics($img);
                    $printer->feed(1);
                }
                $printer->setEmphasis(TRUE);
                $printer->setTextSize(2,3);
                $printer->text("DILNET ACCOUNT \n");
                $printer->setTextSize(2,2);

                switch($print_type){
                    case 'reset':
                        $printer->text("Reset Password \n");
                        break;
                    case 'new':
                        $printer->text("New Account \n");
                        break;
                    case 'info':
                        $printer->text("Account Information\n");
                        break;
                }

                $printer->initialize();
                $printer->feed(2);

                foreach ($ldap_print as $key => $value){
                    if($key == 'password' || $key == 'username'){

```

```

        $printer->setEmphasis(TRUE);
        $printer->text($key.": ");
        $printer->setEmphasis(FALSE);
        $printer->setTextSize(2,1);
        $printer->text($value."\n");
        $printer->initialize();
    }

    else if(is_array($value)){
        foreach(array_slice($value, 1) as $a) {
            $printer->setEmphasis(TRUE);
            $printer->text($key);
            $printer->setEmphasis(FALSE);
            $printer->text(": ".$a."\n");
        }
    }
    else {
        $printer->setEmphasis(TRUE);
        $printer->text($key);
        $printer->setEmphasis(FALSE);
        $printer->text(": ".$value."\n");
    }
}
$printer->feed(2);
if($printer_type == 'reset' || $opt_changepass){
    $printer->text("Please change your password
immediately at \n");

    $printer->setEmphasis(TRUE);
    $printer->setTextSize(1,2);
    $printer->text('https://accounts.upd.edu.ph');
    $printer->feed(2);
    $printer->setTextSize(1,1);
    $printer->text("Or scan this code to go to the
website: ");

    $printer->feed(2);
    $printer-
>setJustification(Printer::JUSTIFY_CENTER);
    $printer-
>qrCode("https://accounts.upd.edu.ph",Printer::QR_ECLEVEL_M,5);
    $printer-
>setJustification(Printer::JUSTIFY_LEFT);
    $printer->feed(2);
}

#Print if Faculty
if($opt_facproxy){
    $printer->setEmphasis(TRUE);
    $printer->text("FACULTY PROXY SETTINGS \n");
    $printer->setEmphasis(FALSE);
    $printer->text("You may access the unrestricted
proxy for the faculty by using the following settings:\n");
    $printer->text("Host:
facultyproxy.upd.edu.ph\nPort:8080 \n");
    $printer->feed(1);
}
if($opt_crs){
    $printer->setEmphasis(TRUE);
    $printer->text("CRS FACULTY ROLE \n");
    $printer->setEmphasis(FALSE);

```

```

        $printer->text("We have forwarded your
information to CRS for tagging. Average processing time is 1-2 days. You may
follow up this request by calling \n");
        $printer->setEmphasis(TRUE);
        $printer->text("981-8500 local 4560. \n");
        $printer->feed(1);
    }

    if($opt_library){
        $printer->text("LIBRARY (iLib) FACULTY ACCESS
\n");

        $printer->setEmphasis(FALSE);
        $printer->text("If you encounter problems with
ilib using your faculty ID (usual reason is an existing iLib student profile),
call ");

        $printer->setEmphasis(TRUE);
        $printer->text("981-8500 local 2864.");
        $printer->setEmphasis(FALSE);
        $printer->text("Request your iLib account to be
changed from student to faculty. \n");
        $printer->feed(2);
    }

    if($opt_services){
        $printer->setTextSize(2,1);
        $printer->text("ONLINE SERVICES\n");
        $printer->initialize();
        $printer->setEmphasis(TRUE);
        $printer->text("DILNET ACCOUNT: ");
        $printer->setEmphasis(FALSE);
        $printer->text("https://accounts.upd.edu.ph\n");

        $printer->setEmphasis(TRUE);
        $printer->text("UPD WEBMAIL: ");
        $printer->setEmphasis(FALSE);
        $printer->text("https://mail.upd.edu.ph\n");

        $printer->setEmphasis(TRUE);
        $printer->text("CRS: ");
        $printer->setEmphasis(FALSE);
        $printer->text("https://crs.upd.edu.ph\n");

        $printer->setEmphasis(TRUE);
        $printer->text("UVLE: ");
        $printer->setEmphasis(FALSE);
        $printer->text("http://uvle.upd.edu.ph\n");

        $printer->setEmphasis(TRUE);
        $printer->text("Library (iLib): ");
        $printer->setEmphasis(FALSE);
        $printer->text("http://ilib.upd.edu.ph\n");
        $printer->feed(2);
    }

    if($opt_helpdesk){
        $printer->initialize();
        $printer->setTextSize(2,1);
        $printer->text("DILNET HELPDESK\n");
        $printer->setTextSize(1,1);
        $printer->text("Contact us if you have problems
with the ff: \n");
    }

```

```

        $printer->text("- Connecting to DILNET Wifi\n-
UPD Webmail (mail.upd.edu.ph)\n- Dilnet Accounts (accounts.upd.edu.ph)\n");
        $printer->feed(1);
        $printer->setEmphasis(TRUE);
        $printer->text("Email: ");
        $printer->setEmphasis(FALSE);
        $printer->text("helpdesk@upd.edu.ph\n");
        $printer->setEmphasis(TRUE);
        $printer->text("Phone: ");
        $printer->setEmphasis(FALSE);
        $printer->text("981-8500 local 2050\n");
        $printer->setEmphasis(TRUE);
        $printer->text("Website: ");
        $printer->setEmphasis(FALSE);
        $printer->text("http://dilnet.upd.edu.ph\n");
        $printer->setEmphasis(TRUE);
        $printer->text("Facebook: ");
        $printer->setEmphasis(FALSE);
        $printer->
>text("http://www.facebook.com/updilnet\n");
        $printer->setEmphasis(TRUE);
        $printer->text("Twitter: ");
        $printer->setEmphasis(FALSE);
        $printer->
>text("http://www.twitter.com/dilnet\n");
        $printer->feed(2);
    }

    $date = date("Y-m-d H:i:s");
    $printer->text("Date/Time Generated: ".$date."\n");
    $printer->feed(1);
    $printer -> cut();
    $printer -> close();

    // switch($print_type){
    //     case 'reset':
    //         $_SESSION['success'] = "Password reset successful
for <strong>".strtoupper($username)."</strong>.";
    //         break;
    //     case 'new':
    //         $_SESSION['success'] = "New user created:
<strong>".strtoupper($username)."</strong>.";
    //         break;
    //     case 'info':
    //         $_SESSION['success'] = "Account information
printed for <strong>".strtoupper($username)."</strong>.";
    //         break;
    //     }

    // $this->session->mark_as_flash('success');
    $_SESSION['success'] = "New accounts printed.";
    $this->session->mark_as_flash('success');

    } //else.if printer is up
} //else.if ldap_print is empty
else {
    $_SESSION['error'] = "You tried to print something but the
data was lost. Sorry.";
    $this->session->mark_as_flash('error');

```

```

        redirect('tech/dilnet_lookup');
    }

}
/**/

redirect('tech/create_user');
}

public function cancel($username = NULL) {
    if($username != NULL){
        $_SESSION['warning'] = "Cancelled Printing for
<strong>".strtoupper($username)."</strong>.";
        $this->session->mark_as_flash('warning');
        redirect('tech/dilnetacc_tools/view_account/' . $username);
    }
    else { redirect('tech/dilnet_lookup'); }
}
}

```

application/controllers/tech/Proxy_tools.php

```

<?php
defined('BASEPATH') OR exit('No direct script access allowed');

class Proxy_tools extends MY_Controller {

    function __construct(){
        parent::__construct();

        $this->form_validation->set_error_delimiters('<label>', '</label>');
        $this->load->model('ldap_model');
        $this->load->model('mysql_model');
        $this->load->model('log_model');
        $this->load->model('proxy_tools_model');

        $this->tech_user = $_SESSION['tech_user'];
        $this->ip = $this->input->ip_address();

        $this->page_vars = array(
            'content' => 'user_blocking_search_view',
            'page_title' => 'Search Proxy DB',
            'page_subtitle' => 'Search the Proxy DB for blocked/unblocked
username(s)',
            'include_stylesheets' => 'account_user_view_css',
            'include_scripts' => 'account_user_view_js'
        );
    }

    public function index() {

        // Load page variables
        $data = $this->page_vars;

        $this->load->vars($data);
    }
}

```

```

        $this->load->view('page_template');
    }

    public function block_user_confirm($block_email = NULL) {

        // Load page variables
        $data = array(
            'content' => 'user_account_disable_confirm',
            'page_title' => 'Block User Proxy Access',
            'page_subtitle' => null,
            'include_stylesheets' => 'dilnet_account_tools_css',
            'include_scripts' => 'dilnet_account_tools_js'
        );

        #Get username from previous page (if coming from disable_form)
        if(!empty($_SESSION['block_email'])) {
            $block_email = $_SESSION['block_email'];
        }

        #Get username from post input, if form is reloaded by
form_validation
        if(empty($block_email)) {
            $block_email = $this->input->post('block_email', TRUE);
        }

        if(!empty($block_email)) {
            $email = $this->proxy_tools_model->
get_block_user_info(base64_decode(urldecode($block_email)));

            $this->form_validation->set_rules('disable_reason', 'Reason',
'required|xss_clean');

            if ($this->form_validation->run() == TRUE) {

                $reason = $this->input->post('disable_reason');
                $_SESSION['disableinfo']['reason'] = $reason;
                $this->session->mark_as_flash('disableinfo');

                redirect('/tech/proxy_tools/block_user_final/'.urlencode($block_email));
            }

            $data['block_email'] = trim($block_email);
            $data['email'] = $email;

            $this->load->vars($data);
            $this->load->view('page_template');
        }

        else {
            #TODO Keep session data for next redirect
            if (isset($_SESSION['success'])) {
                $_SESSION['success'] = $_SESSION['success'];
                $this->session->mark_as_flash('success');
            }
            else {
                $_SESSION['error'] = "Could not get username for
Account disable operation. Try again.";
                $this->session->mark_as_flash('error');
            }
        }
    }

```



```

        redirect('/tech/proxy_tools/user_blocking');
    }
}

public function block_user_final($block_email = NULL) {
    if (!empty($block_email)){
        // print "block_email: $block_email<br>";
        $decoded_email = base64_decode(urldecode($block_email));
        // print "decoded_email: $decoded_email";

        $account_mysql = $this->proxy_tools_model-
>get_block_user_info($decoded_email);

        $reason = $_SESSION['disableinfo']['reason'];

        $admin = $this->tech_user;

        if($account_mysql['disabled']) {
            $this->session->set_flashdata('info', "Account
<strong>".$decoded_email."</strong> already blocked. No action taken.");
            // $this->session->mark_as_flash("success");

        } else {
            $reason = $reason != NULL ? $reason : "";
            $this->proxy_tools_model-
>disable_username($decoded_email, $reason, $admin);

            $this->log_model->create($decoded_email, "Account
disabled. ".$reason, "update_mysql", $this->tech_user, $this->ip, 'info');
            $this->session->set_flashdata('warning', "Account
<strong>".$decoded_email."</strong> blocked.");
        }

        redirect('tech/proxy_tools/view_user_proxy_status/'.urlencode(base64_encode
($decoded_email)));
    }

    else {
        $_SESSION['error'] = "Disable account data lost. Please try
again.";
        $this->session->mark_as_flash('error');
        redirect("/tech/mail_tools/email_blocking/");
    }
}

public function unblock_user_confirm($block_email = NULL) {
    // Load page variables
    $data = array(
        'content' => 'user_account_disable_confirm',
        'page_title' => 'UnBlock Proxy User',
        'page_subtitle' => 'Restore Proxy Access',
        'include_stylesheets' => 'dilnet_account_tools_css',

```

```

        'include_scripts' => 'dilnet_account_tools_js'
    );

    if(!empty($_SESSION['block_email'])) {
        $block_email = $_SESSION['block_email'];
    }

    #Get username from post input, if form is reloaded by
form_validation
    if(empty($block_email)) {
        $block_email = $this->input->post('block_email', TRUE);
    }

    if(!empty($block_email)) {
        $email = $this->proxy_tools_model-
>get_block_user_info(base64_decode(urldecode($block_email)));

        $this->form_validation->set_rules('disable_reason', 'Reason',
'required|xss_clean');

        if ($this->form_validation->run() == TRUE) {

            $reason = $this->input->post('disable_reason');
            $_SESSION['disableinfo']['reason'] = $reason;
            $this->session->mark_as_flash('disableinfo');

            redirect('/tech/proxy_tools/unblock_user_final/'.urlencode($block_email));
        }

        $data['block_email'] = trim($block_email);
        $data['email'] = $email;

        $this->load->vars($data);
        $this->load->view('page_template');
    }

    else {
        #TODO Keep session data for next redirect
        if (isset($_SESSION['success'])) {
            $_SESSION['success'] = $_SESSION['success'];
            $this->session->mark_as_flash('success');
        }
        else {
            $_SESSION['error'] = "Could not get username for
Account disable operation. Try again.";
            $this->session->mark_as_flash('error');
        }

        redirect('/tech/proxy_tools/user_blocking');
    }
}

public function unblock_user_final($block_email = NULL) {
    if (!empty($block_email)) {

        // print "block_email: $block_email<br>";
        $decoded_email = base64_decode(urldecode($block_email));
        // print "decoded_email: $decoded_email";
    }
}

```

```

        $account_mysql = $this->proxy_tools_model-
>get_block_user_info($decoded_email);

        $reason = $_SESSION['disableinfo']['reason'];

        $admin = $this->tech_user;

        if($account_mysql['disabled']) {
            $reason = $reason != NULL ? $reason : "";
            $this->proxy_tools_model-
>enable_username($decoded_email, $reason, $admin);
            $this->log_model->create($decoded_email, "Account
enabled. ".$reason, "update_mysql", $this->tech_user, $this->ip, 'info');
            $this->session->set_flashdata('info', "Account
<strong>".$decoded_email."</strong> unblocked.");

        } else {
            $this->session->set_flashdata('info', "Account
<strong>".$decoded_email."</strong> is not blocked. No action taken.");
            // $this->session->mark_as_flash("success");
        }

        redirect('tech/proxy_tools/view_user_proxy_status/' .urlencode(base64_encode
($decoded_email)));
    }

    else {

        $_SESSION['error'] = "Disable account data lost. Please try
again.";

        $this->session->mark_as_flash('error');
        redirect("/tech/proxy_tools/user_blocking/");

    }

}

public function user_blocking($new=NULL) {

    $data = array(
        'content' => 'user_blocking_search_view',
        'page_title' => 'Search Proxy DB',
        'page_subtitle' => 'Search the Proxy DB for blocked/unblocked
username(s)',
        'include_stylesheets' => 'account_user_view_css',
        'include_scripts' => 'account_user_view_js'
    );

    if($new=="new"){
        if(isset($_SESSION['user_blocking_results_db'])){
            unset($_SESSION['user_blocking_results_db']);
        }
        if(isset($_SESSION['user_blocking_search_string'])){
            unset($_SESSION['user_blocking_search_string']);
        }
    }

}

```

```

        $temp_user_block_search_str = $this->input-
>post('user_block_search_str');
        if(!empty($_SESSION['user_blocking_results_db']) &&
!empty($_SESSION['user_blocking_search_string']) &&
empty($temp_user_block_search_str)){

            $data['user_blocking_results_db'] =
$_SESSION['user_blocking_results_db'];
            $data['user_blocking_search_string'] =
$_SESSION['user_blocking_search_string'];

        } else {

            $this->form_validation->set_rules('user_block_search_str',
'Search', 'required|trim|xss_clean|min_length[3]');

            if ($this->form_validation->run() == TRUE){
                $temp = $this->input->post('user_block_search_str');

                if(isset($_SESSION['user_blocking_results_db'])){
                    unset($_SESSION['user_blocking_results_db']);
                }

                if(isset($_SESSION['user_blocking_search_string'])){
                    unset($_SESSION['user_blocking_search_string']);
                }

                if(!empty($temp)){
                    $search_string = $this->input-
>post('user_block_search_str', TRUE);
                    // print "search_string: $search_string";
                    $ldap_user = $this->ldap_model-
>get_user_account($search_string);
                    if (!empty($ldap_user)) {

                        redirect('tech/proxy_tools/view_user_proxy_status/'.urlencode(base64_encode($search_string)));

                    } else {
                        $search_results = $this-
>proxy_tools_model->search_email($search_string);
                        if(!empty($search_results)){
                            $data["user_blocking_results_db"] =
$search_results;
                            $data['user_blocking_search_string']
= $search_string;

                            $_SESSION["user_blocking_results_db"] = $search_results;
                            $_SESSION['user_blocking_search_string'] = $search_string;
                        } else {
                            $this->session-
>set_flashdata('warning', "No match found.");
                            redirect('tech/proxy_tools/user_blocking/');
                        }
                    }
                }
            }
        }
    }
}

```

```

        }
    } else {
        $this->session->set_flashdata('error', "Invalid
search query.");
        redirect('tech/proxy_tools/user_blocking/');
    }
}
// Load page variables
$this->load->vars($data);
$this->load->view('page_template');
}

public function view_user_proxy_status($email) {
    // Load page variables
    $data = array(
        'content' => 'username_view',
        'page_title' => 'View User Proxy Status',
        'page_subtitle' => 'View Account Status in Proxy Servers',
        'include_stylesheets' => 'account_user_view_css',
        'include_scripts' => 'account_user_view_js'
    );

    $decoded_email = base64_decode(urldecode($email));
    $mail_log = $this->proxy_tools_model->get_block_user_info($decoded_email);

    $data['block_email'] = $decoded_email;

    if(!empty($mail_log)){
        $data['results_db'] = $mail_log;
    } else {
        $this->session->set_flashdata('info', "Username not in blocked
list.");
        redirect('tech/proxy_tools/block_user_confirm/'.urlencode(base64_encode($de
coded_email)));
    }

    $this->load->vars($data);
    $this->load->view('page_template');
}
}

```

application/controllers/tech/Radius_profile.php

```

<?php
defined('BASEPATH') OR exit('No direct script access allowed');

class Radius_profile extends MY_Controller {

```

```

function __construct(){
    parent::__construct();

    $this->form_validation->set_error_delimiters('<label>', '</label>');
    $this->load->model('mysql_radius_model');

    $this->page_vars = array(
        'content' => 'radius_profile',
        'page_title' => 'Radius Profile Lookup',
        'page_subtitle' => 'Get Radius profile by Username',
        'include_stylesheets' => 'radius_profile_css',
        'include_scripts' => 'radius_profile_js'
    );
}

public function index()
{
    // Load page variables
    $data = $this->page_vars;
    $data['users_today'] = $this->mysql_radius_model->get_users_today();
    $data['online_users'] = $this->mysql_radius_model-
>get_online_users();
    $data['today_input_total'] = $this->mysql_radius_model-
>get_total_input_usage();
    $data['today_output_total'] = $this->mysql_radius_model-
>get_total_output_usage();
    $this->load->vars($data);
    $this->load->view('page_template');
}

public function name_search($username=NULL) {
    // Load page variables
    $data = $this->page_vars;

    #TODO Check LDAP
    if($username != NULL){
        $search_string = $username;
        $data["results_db"] = $this->mysql_radius_model-
>get_username_profile($search_string);
        $data["username"] = $search_string;
    }
    else {
        $this->form_validation->set_rules('username', 'Username',
'required|xss_clean|min_length[3]');
        if ($this->form_validation->run() == TRUE){
            $search_string = $this->input->post('username');
            #TODO: Sanitize search string
            if($search_string != "" || $search_string != NULL) {
                $data["results_db"] = $this->mysql_radius_model-
>get_username_profile($search_string);
                $data["username"] = $search_string;
            }
        }
    }

    $data['users_today'] = $this->mysql_radius_model->get_users_today();
}

```

```

        $data['online_users'] = $this->mysql_radius_model-
>get_online_users();
        $data['today_input_total'] = $this->mysql_radius_model-
>get_total_input_usage();
        $data['today_output_total'] = $this->mysql_radius_model-
>get_total_output_usage();
        $this->load->vars($data);
        $this->load->view('page_template');
    }
}

```

application/controllers/Accounts.php

```

<?php if ( ! defined('BASEPATH')) exit('No direct script access allowed');

use Mike42\Escpos\EscposImage;
use Mike42\Escpos\PrintConnectors\NetworkPrintConnector;
use Mike42\Escpos\Printer;

class Accounts extends MY_Controller {

    function __construct(){
        parent::__construct();

        $this->load->model('account_model');
        $this->load->model('crs_model');
        $this->load->model('admin_model');
        $this->load->model('sam_user_model');
        $this->load->model('sam_ldap_model');
        $this->load->model('changepass_model');

        $this->page_vars = array(
            'content' => 'admin/index',
            'page_title' => 'Activate Student Account',
            'page_subtitle' => 'Sign Up/Reset Student Accounts',
            'include_stylesheets' => 'merge_css',
            'include_scripts' => 'dilnet_lookup_js'
        );
    }

    public function index()
    {
        $data = $this->page_vars;

        $studentno = $this->session->flashdata('studentno');

        $this->form_validation->set_rules('idno', 'Student/Barcode number',
'trim|required|xss_clean|max_length[12]|callback__check_if_valid_idno');

        if($this->form_validation->run()){
            $form_data = $this->input->post(NULL,TRUE);

```

```

        // Format student number
        if($this->_is_idno_barcode($form_data['idno'])){
            $studno_from_barcode = $this->account_model-
>is_enrolled_by_barcode($form_data['idno']);

            // Get studentno from barcode
            if($studno_from_barcode['success']){
                $studentno =
$studno_from_barcode['details']['studentno'];
                $studentno = $this-
>_format_studentno($studentno,TRUE);
            }
            else {
                $this->session->set_flashdata('error',
$studno_from_barcode['details']['errmsg']);
                redirect('accounts');
                //print "redirecting to accounts/error ...";
            }
        }

        else if($this->_is_idno_studentno($form_data['idno'])){
            $studentno = $this-
>_format_studentno($form_data['idno'],TRUE);
        }

        // Check if student exists in UCC database and LDAP
        $is_eligible_for_creation = $this->account_model-
>is_eligible_for_creation($studentno);
        if($is_eligible_for_creation["success"]){
            $this->session->set_flashdata('studentno', $studentno);
            redirect('activation');
            // print "redirecting to activation ...";
        }

        else {
            $this->session->set_flashdata('form_resetpass_load',
TRUE);

            $this->session->set_flashdata('studentno', $studentno);
            // print "redirecting to accounts/resetpass ...";

            $sn = $studentno;
            $mysql_user = $this->sam_user_model-
>get_by_student_number($sn);
            $username = $mysql_user["username"];
            $ldap_user = $this->sam_ldap_model-
>get_ldapentry($username);
            # what if mysql user not found

            # ADMIN USER
            if ($this->admin_model->admin_user_exists($username)) {
                $this->session->set_flashdata('warning', "Admin
staff account detected.");
                redirect('tech/dilnetacc_tools/view_account/' . $username . '');
            }

            # OFFICE ACCOUNT

```



```

        if (!empty($ldap_user["employeetype"]) && !$this->is_tech()) {
            $this->session->set_flashdata('warning',
"Employee account detected.");
            redirect('tech/dilnetacc_tools/view_account/'.$username.'');
        }

        # DISABLED ACCOUNT
        if (($mysql_user["disabled"] == 1) && !$this->is_tech()) {
            $this->session->set_flashdata('warning',
"Disabled account detected.");
            redirect('tech/dilnetacc_tools/view_account/'.$username.'');
        }

        redirect('accounts/resetpass');
    }

}

else {
    $data['content'] = 'admin/index';
    $this->load->vars($data);
    $this->load->view('page_template');
}

}

# TODO: accept other inputs other than studentnumber?
public function resetpass(){

    $this->page_vars = array(
        'content' => 'admin/form_resetpass',
        'page_title' => 'Student Account Manager - Reset Password',
        'page_subtitle' => 'Reset Password of Dilnet Account',
        'include_stylesheets' => 'merge_css',
        'include_scripts' => 'dilnet_lookup_js'
    );

    $data = $this->page_vars;
    // From accounts controller
    $studentno = $this->session->flashdata('studentno');
    // From resetpass form (if there are form validation errors)
    if(empty($studentno)) { $studentno = $this->input->post("studentno",
TRUE); }

    if($this->session->flashdata('form_resetpass_load') == FALSE ||
empty($studentno)){
        redirect("accounts");
    }

    // $this->form_validation->set_rules('empno', 'Employee Number',
'required|max_length[10]|callback__is_admin_user');

```

```

        $this->form_validation->set_rules('reason', 'Password Change
Reason', 'required');

        if ( $this->form_validation->run() ) {
            //TOFIX: is_enrolled
            $data["is_enrolled"] = $this->input->post("is_enrolled",
TRUE);

            $sn = $this->input->post("studentno", TRUE);
            $mysql_user = $this->sam_user_model-
>get_by_student_number($sn);
            $username = $mysql_user["username"];
            $ldap_user = $this->sam_ldap_model->get_ldapentry($username);
            # what if mysql user not found

            # ADMIN USER
            if ($this->admin_model->admin_user_exists($username)) {
                $this->session->set_flashdata('warning', "Admin staff
account detected.");

                redirect('tech/dilnetacc_tools/view_account/' . $username . '');
            }

            # OFFICE ACCOUNT
            if (!empty($ldap_user["employee_type"]) && !$this->_is_tech())
{
                $this->session->set_flashdata('warning', "Employee
account detected.");

                redirect('tech/dilnetacc_tools/view_account/' . $username . '');
            }

            # DISABLED ACCOUNT
            if (($mysql_user["disabled"] == 1) && !$this->_is_tech()) {
                $this->session->set_flashdata('warning', "Disabled
account detected.");

                redirect('tech/dilnetacc_tools/view_account/' . $username . '');
            }

            # ELIGIBLE FOR RESET
            $data["account"]["password"] = random_password();
            $data["account"]["microtime"] = microtime()*1000000;
            mt_srand((double)microtime()*1000000);
            $salt = pack("CCCC", mt_rand(), mt_rand(), mt_rand(),
mt_rand());
            $newdata["userPassword"] = "{SSHA}" . base64_encode(pack("H*",
sha1($data["account"]["password"] . $salt)) . $salt);

            if($this->sam_ldap_model->set_ldapentry($username,
$ldap_user["entrydn"], $newdata)){
                $this->sam_user_model-
>enable_force_changepass($username);
                $this->changepass_model-
>reset_recovery_count($username);

                # logging
                $userip = $this->input->ip_address();

                //$currentuser = $this->session->userdata("user");

```

```

        $currentuser = $this->tech_user =
$_SESSION['tech_user'];
        $message = $this->input->post('reason') != NULL ?
$this->input->post('reason') : "reset password";
        $this->changepass_model->create($username, $userip,
$currentuser, $message);
    }

    $data["account"]["user"] = $ldap_user;
    $data["printer"] = $this->input->post("printer");
    $this->send_reset_email($data["account"]);
    $this->session->set_flashdata('passinfo', $data);
    redirect('accounts/account_print_receipt');
}
else {
    $this->load_resetpass_form($studentno);
}
}

function _load_resetpass_form($studentno = NULL){

    $data = $this->page_vars;

    $is_eligible_for_reset = $this->account_model-
>is_eligible_for_reset($studentno);

    $data['studentno'] = $studentno;

    if($is_eligible_for_reset["success"]){

        if(isset($is_eligible_for_reset["student"]["details"])){

            $student =
$is_eligible_for_reset["student"]["details"];

            $data['student_crs_display'] = array(
                'Name' =>
strtoupper($student['lastname'].", ".$student['firstname'].
".$student['middlename']),
                'Student Number' => $studentno,
                'Gender' => ucwords($student['gender']),
                'College' => $student['mainunit'],
                'Department' => $student['subunit'],
                'Degree Program' => $student['program']
            );

        }

        $mysql_user = $this->sam_user_model-
>get_by_student_number($studentno);

        $data['student_ucc_display'] = array(
            'DilNet Username' => '<span class="text-
info" style="font-weight: bold;">'.$mysql_user["username"].'</span>'
        );

        $data['is_enrolled'] =
$is_eligible_for_reset['is_enrolled'];
    }
}

```

```

        if($data['is_enrolled']){
            $data['status'] = '<span class="text-
success">ENROLLED</span>';
        }
        else {
            $data['status'] = '<span class="text-danger">NOT
ENROLLED</span>';
        }

        $this->output->set_header("Expires: Tue, 01 Jan 2000
00:00:00 GMT");
        $this->output->set_header("Last-Modified: " .
gmdate("D, d M Y H:i:s") . " GMT");
        $this->output->set_header("Cache-Control: no-store, no-
cache, must-revalidate, max-age=0");
        $this->output->set_header("Cache-Control: post-check=0,
pre-check=0", false);
        $this->output->set_header("Pragma: no-cache");

        $this->session->set_flashdata('form_resetpass_load',
TRUE);

        $data['content'] = 'admin/form_resetpass';
        $this->load->vars($data);
        $this->load->view('page_template');
    }

    else {
        $this->session->set_flashdata('error',
$is_eligible_for_reset['message']);
        redirect('accounts');
    }
}

# Used to avoid refresh/reload on reset password
function reset_print() {
    $data = $this->session->flashdata('passinfo');

    // Prevent reload of page when back button is pressed
    $this->session->set_flashdata('form_resetpass_load', FALSE);

    if (!empty($data)) {
        $data["print_title"] = "RESET PASSWORD";
        $alternate_emails = NULL;

        foreach(array_slice($data["account"]["user"]["mailalternateaddress"], 1) as
$a) {
            $alternate_emails = $alternate_emails.$a."<br>";
        }

        $data["print_array"] = array(
            'Username' => $data["account"]["user"]["username"],
            'Password' => $data["account"]["password"]
        );

        if ($data["is_enrolled"]){
            $data["print_array"]["Primary Email"] =
$data["account"]["user"]["mail"];

```

```

        $data["print_array"]['Alternate Email'] =
$alternate_emails;
    }

    $this->load->vars($data);
    $this->load->view('partials/print');
}
else { redirect("accounts"); }
}

function account_print_receipt() {
    $print_data = $this->session->flashdata('passinfo');

    // Reload page wehn back button is pressed
    $this->session->set_flashdata('form_resetpass_load', FALSE);

    // Prevent reload of page when back button is pressed
    $this->session->set_flashdata('form_resetpass_load', FALSE);

    if (!empty($print_data)) {

        $print_array = array(
            'Username' =>
$print_data["account"]["user"]["username"],
            'Password' => $print_data["account"]["password"]
        );

        if ($print_data["is_enrolled"]){
            $print_array['Primary Email'] =
$print_data["account"]["user"]["mail"];
            $print_array['Alternate Email'] =
$print_data["account"]["user"]["mailalternateaddress"];
        }

        //admin-thermal by default
        $printer_ip = "10.40.253.207";

        switch ($print_data["printer"]){
            case 'tech-thermal':
                $printer_ip = "10.40.2.207";
                break;
            case 'tech-ribbon':
                $printer_ip = "10.40.3.203";
                break;
            default:
                $printer_ip = "10.40.253.207";
                break;
        }
        // bypass printer IP selection, sets to tech-thermal
        $printer_ip = "10.40.2.207";

        $connector = new NetworkPrintConnector($printer_ip, 9100);
        $printer = new Printer($connector);
        $printer->setJustification(Printer::JUSTIFY_CENTER);
        $img =
EscposImage::Load($_SERVER['DOCUMENT_ROOT']."/techaccounts/assets/dilnetlogo.png")
;
    }

```

```

$printer->graphics($img);
$printer->feed(1);
$printer->setEmphasis(TRUE);
$printer->setTextSize(2,3);
$printer->text("DILNET ACCOUNT \n");
$printer->setTextSize(2,2);

$printer->text("Reset Password \n");

$printer->initialize();
$printer->feed(2);

foreach ($print_array as $key => $value){
    if($key == 'Password' || $key == 'Username'){
        $printer->setEmphasis(TRUE);
        $printer->text($key.": ");
        $printer->setEmphasis(FALSE);
        $printer->setTextSize(2,1);
        $printer->text($value."\n");
        $printer->initialize();
    }

    else if(is_array($value)){
        foreach(array_slice($value, 1) as $a) {
            $printer->setEmphasis(TRUE);
            $printer->text($key);
            $printer->setEmphasis(FALSE);
            $printer->text(": ".$a."\n");
        }
    }
    else {
        $printer->setEmphasis(TRUE);
        $printer->text($key);
        $printer->setEmphasis(FALSE);
        $printer->text(": ".$value."\n");
    }
}
$printer->feed(2);

$printer->text("Please change your password immediately at
\n");

$printer->setEmphasis(TRUE);
$printer->setTextSize(1,2);
$printer->text('https://accounts.upd.edu.ph');
$printer->feed(2);
$printer->setTextSize(1,1);
$printer->text("Or scan this code to go to the website: ");
$printer->feed(2);
$printer->setJustification(Printer::JUSTIFY_CENTER);
$printer->
>qrCode("https://accounts.upd.edu.ph",Printer::QR_ECLEVEL_M,5);
$printer->setJustification(Printer::JUSTIFY_LEFT);
$printer->feed(2);

$printer->setTextSize(2,1);
$printer->text("ONLINE SERVICES\n");
$printer->initialize();
$printer->setEmphasis(TRUE);
$printer->text("DILNET ACCOUNT: ");

```

```

$printer->setEmphasis(FALSE);
$printer->text("https://accounts.upd.edu.ph\n");

$printer->setEmphasis(TRUE);
$printer->text("UPD WEBMAIL: ");
$printer->setEmphasis(FALSE);
$printer->text("https://mail.upd.edu.ph\n");

$printer->setEmphasis(TRUE);
$printer->text("CRS: ");
$printer->setEmphasis(FALSE);
$printer->text("https://crs.upd.edu.ph\n");

$printer->setEmphasis(TRUE);
$printer->text("UVLE: ");
$printer->setEmphasis(FALSE);
$printer->text("http://uvle.upd.edu.ph\n");

$printer->setEmphasis(TRUE);
$printer->text("Library (ilib): ");
$printer->setEmphasis(FALSE);
$printer->text("http://ilib.upd.edu.ph\n");
$printer->feed(2);

$date = date("Y-m-d H:i:s");
$printer->text("Date/Time Generated: ".$date."\n");
$printer->feed(1);
$printer -> cut();
$printer -> close();

```

```

}

```

```

$this->session->set_flashdata('info', "New password printed.");
redirect("accounts");

```

```

}

```

```

/* Callback functions */

```

```

function _check_if_valid_idno($idno){
    if($this->_is_idno_barcode($idno) || $this->_is_idno_studentno($idno)){
        return TRUE;
    }

```

```

    $this->form_validation->set_message('_check_if_valid_idno', 'Invalid student or barcode number!');
    return FALSE;
}

```

```

/* Checker functions */

```

```

function _is_admin_user($empno){
    $this->form_validation->set_message('_is_admin_user', 'You do not have admin privileges. ');
    $result = $this->admin_model->get_by_employee_number($empno);

```

```

    if ( $result != NULL ) {
        if(isset($result["role"]) && $result["role"] != 0) { return TRUE; }
        else { return FALSE; }
    }

```

```

    return FALSE;
}

```

```

} // function.is_admin_user

public function _send_reset_email($details) {

    $this->load->library('email');

    $message = 'Username: '.$details["user"]['username']."\n";
    $message .= 'Password: '.$random_string('alnum',
3).$details['password']."\n";
    $message .= 'Email: '.$details["user"]['mail']."\n";

    $message .= "\nPlease change your password immediately\n";
    // $message .= 'Admin: '.$this->session->userdata("user")."\n";
    $message .= date('Y-m-d H:i:s');

    $this->email->from('support@upd.edu.ph', 'Password Changer');
    $this->email->to('postmaster@upd.edu.ph');
    $this->email->subject('passwd ['.$details["user"]['username'].']");
    $this->email->message($message);

    $this->email->send();
    // echo $this->email->print_debugger();
}
}

```

application/controllers/Activation.php

```

<?php if ( ! defined('BASEPATH')) exit('No direct script access allowed');

use Mike42\Escpos\EscposImage;
use Mike42\Escpos\PrintConnectors\NetworkPrintConnector;
use Mike42\Escpos\Printer;

class Activation extends MY_Controller {

    /**
     * Index Page for this controller.
     *
     * Maps to the following URL
     *      http://example.com/index.php/welcome
     *  - or -
     *      http://example.com/index.php/welcome/index
     *  - or -
     * Since this controller is set as the default controller in
     * config/routes.php, it's displayed at http://example.com/
     *
     * So any other public methods not prefixed with an underscore will
     * map to /index.php/welcome/<method_name>
     * @see http://codeigniter.com/user_guide/general/urls.html
     */
    function __construct(){
        parent::__construct();

        $this->load->model('crs_model');
    }
}

```



```

$this->load->model('sam_user_model');
$this->load->model('sam_ldap_model');
$this->load->model('account_model');
$this->load->model('admin_model');
$this->load->model('units_model');
$this->load->model('log_model');

$this->page_vars = array(
    'content' => 'activation/index',
    'page_title' => 'Student Account Activation',
    'page_subtitle' => 'Verify',
    'include_stylesheets' => 'merge_css',
    'include_scripts' => 'dilnet_lookup_js'
);
}

public function index(){

    $data = $this->page_vars;

    // Display form
    $studno = $this->session->flashdata('studentno');

    if(!empty($studno)){
        $this->session->set_flashdata('form_newaccount_load', TRUE);
        $this->session->set_flashdata('studentno', $studno);

        redirect('activation/verify');
    } else {
        redirect("accounts");
    }
}

/**
 * Check student_number && name eligibility in:
 * CRS, Mysql, and ldap
 * Create Dilnet Account if eligible
 * Else return Error
 * Restrict user from refresh / resubmit
 */

public function verify(){

    $this->page_vars = array(
        'content' => 'admin/form_newaccount',
        'page_title' => 'Student Account Activation',
        'page_subtitle' => 'Verify',
        'include_stylesheets' => 'merge_css',
        'include_scripts' => 'dilnet_lookup_js'
    );

    $data = $this->page_vars;

    // From accounts controller
    $studentno = $this->session->flashdata('studentno');

    // From activation/verify form (if there are form validation errors)

```

```

        if(empty($studentno)) { $studentno = $this->input->post("studentno",
TRUE); }

        if($this->session->flashdata('form_newaccount_load') == FALSE ||
empty($studentno)){
            redirect("accounts");
        }

        // $this->form_validation->set_rules('empno', 'Employee number',
'required|callback__is_admin_user');

        // TODO: ADD OU validation
        if ( $this->form_validation->run() ) {
            // $employee = $this->admin_model-
>get_username_from_employee_number($this->input->post("empno", TRUE));
            $ip = $this->input->ip_address();

            $student_number = $this->input->post("studentno", TRUE);
            $student = $this->account_model-
>get_signupinfo_from_crs($student_number);
            $employee = $_SESSION['tech_user'];
            $this->log_model->create_activation($student_number,
"Verifying account for creation", 'debug', $employee, $ip);

            # ELIGIBILITY FAIL
            $eligibility = $this->account_model-
>is_eligible_for_creation($student_number);
            if ( !$eligibility["success"] ) {
                $this->log_model->create_activation($student_number,
                "Ineligible for creation: ".$eligibility["message"], 'warn', $employee, $ip);
                $this->session->set_flashdata('error',
$eligibility["message"]);
                redirect('accounts');
            }

            # ELIGIBILITY SUCCESS: every thing below
            # ELIGIBLE BUT PERFECT MATCH
            $perfect_match = $this->sam_user_model-
>get_by_name($student["details"]);
            if ( !empty($perfect_match) && !$this->_is_tech() ) {
                $this->session->set_flashdata('error', "Perfect name
match or special case found. Proceed to Rm 301 to activate account.");
                redirect('accounts');
            }

            $create_result = $this->account_model-
>create_student($student_number, $employee);

            # CREATION SUCCESS
            if ( $create_result["success"] ) {
                $this->log_model->create_activation($student_number,
                "DilNet account created : ".$create_result["user"]["uid"], 'info', $employee,
                $ip);
                $create_result["printer"] = $this->input-
>post("printer");
                $this->session->set_flashdata('details',
                $create_result);
                $this->_send_welcome_mail($create_result);
                redirect('activation/account_print_receipt');
            }

```

```

    }

    # CREATION FAIL
    else {
        $this->log_model->create_activation($student_number,
"Account creation failed.", 'error', $employee, $ip);
        $this->session->set_flashdata('error',
$create_result["message"]);
        redirect('accounts');
    }

    $this->session->set_flashdata('details', $create_result);
    redirect('activation/account_print');

} // form_validation.run
else {
    $this->_load_activation_form($studentno);
}
} // function.verify

#TODO: Set studno default value NULL
function _load_activation_form($studno){
    $data = $this->page_vars;

    $student = $this->account_model->get_signupinfo_from_crs($studno);

    // print_r($student);
    if($student["success"]){
        $data['student_display'] = array(
            'Name' => strtoupper($student['details']['lastname'].",
".$student['details']['firstname']. " ".$student['details']['middlename']),
            'Student Number' => $studno,
            'Gender' => ucwords($student['details']['gender']),
            'College' => $student['details']['mainunit'],
            'Department' => $student['details']['subunit'],
            'Degree Program' => $student['details']['program']
        );

        $data['studentno'] = $studno;

        $this->output->set_header("Expires: Tue, 01 Jan 2000 00:00:00
GMT");
        $this->output->set_header("Last-Modified: " . gmdate("D, d M Y
H:i:s") . " GMT");
        $this->output->set_header("Cache-Control: no-store, no-cache,
must-revalidate, max-age=0");
        $this->output->set_header("Cache-Control: post-check=0, pre-
check=0", false);
        $this->output->set_header("Pragma: no-cache");

        $this->session->set_flashdata('form_newaccount_load', TRUE);

        $data['content'] = 'admin/form_newaccount';
        $this->load->vars($data);
        $this->load->view('page_template');
    }

    else {

```

```

        // Can't get info from CRS for student
        $this->session->set_flashdata('error', $student['message']);
        redirect('accounts');
    }
}

function account_print() {
    $print_data = $this->session->flashdata('details');

    // Reload page wehn back button is pressed
    $this->session->set_flashdata('form_resetpass_load', FALSE);

    if (!empty($print_data)) {
        $data["print_title"] = "NEW ACCOUNT";

        $data["print_array"] = array(
            'Username' => $print_data["user"]["uid"],
            'Password' => $print_data["password"],
            'Primary Email' => $print_data["user"]["mail"],
            'Alternate Email' =>
$print_data["user"]["mail_alternate"][0],
            'CRS Status' => $print_data["crs_status"]
        );

        $this->load->vars($data);
        $this->load->view('partials/print');
    }

    else { redirect("accounts"); }
}

function account_print_receipt() {
    $print_data = $this->session->flashdata('details');

    // Reload page wehn back button is pressed
    $this->session->set_flashdata('form_resetpass_load', FALSE);

    if (!empty($print_data)) {
        $print_array = array(
            'Username' => $print_data["user"]["uid"],
            'Password' => $print_data["password"],
            'Primary Email' => $print_data["user"]["mail"],
            'Alternate Email' =>
$print_data["user"]["mail_alternate"],
            'CRS Status' => $print_data["crs_status"]
        );

        //admin-thermal by default
        $printer_ip = "10.40.253.207";

        switch ($print_data["printer"]){
            case 'tech-thermal':
                $printer_ip = "10.40.2.207";
                break;
            case 'tech-ribbon':
                $printer_ip = "10.40.3.203";
                break;
        }
    }
}

```

```

        default:
            $printer_ip = "10.40.253.207";
            break;
    }

    $printer_ip = "10.40.2.207";

    $connector = new NetworkPrintConnector($printer_ip, 9100);
    $printer = new Printer($connector);
    $printer->setJustification(Printer::JUSTIFY_CENTER);
    $img =
EscposImage::Load($_SERVER['DOCUMENT_ROOT']."/techaccounts/assets/dilnetlogo.png")
;

    $printer->graphics($img);
    $printer->feed(1);
    $printer->setEmphasis(TRUE);
    $printer->setTextSize(2,3);
    $printer->text("DILNET ACCOUNT \n");
    $printer->setTextSize(2,2);

    $printer->text("New Account \n");

    $printer->initialize();
    $printer->feed(2);

    foreach ($print_array as $key => $value){
        if($key == 'Password' || $key == 'Username'){
            $printer->setEmphasis(TRUE);
            $printer->text($key.": ");
            $printer->setEmphasis(FALSE);
            $printer->setTextSize(2,1);
            $printer->text($value."\n");
            $printer->initialize();
        }

        else if(is_array($value)){
            foreach(array_slice($value, 1) as $a) {
                $printer->setEmphasis(TRUE);
                $printer->text($key);
                $printer->setEmphasis(FALSE);
                $printer->text(": ".$a."\n");
            }
        }
        else {
            $printer->setEmphasis(TRUE);
            $printer->text($key);
            $printer->setEmphasis(FALSE);
            $printer->text(": ".$value."\n");
        }
    }
    $printer->feed(2);

    $printer->text("Please change your password immediately at
\n");

    $printer->setEmphasis(TRUE);
    $printer->setTextSize(1,2);
    $printer->text('https://accounts.upd.edu.ph');
    $printer->feed(2);
    $printer->setTextSize(1,1);

```

```

$printer->text("Or scan this code to go to the website: ");
$printer->feed(2);
$printer->setJustification(Printer::JUSTIFY_CENTER);
$printer-
>qrCode("https://accounts.upd.edu.ph",Printer::QR_ECLEVEL_M,5);
$printer->setJustification(Printer::JUSTIFY_LEFT);
$printer->feed(2);

$printer->initialize();
$printer->setTextSize(2,1);
$printer->text("CRS ACCESS\n");
$printer->setTextSize(1,1);
$printer->text("If CRS status is ");
$printer->setEmphasis(TRUE);
$printer->text("TAGGED: ");
$printer->setEmphasis(FALSE);
$printer->text("\n-You can now log in to the CRS website using
your DilNet account.\nIf status is ");
$printer->setEmphasis(TRUE);
$printer->text("UNTAGGED: ");
$printer->setEmphasis(FALSE);
$printer->text("\n-Please go to the OUR-CRS section to request
linking of your DilNet account to your CRS profile. Or you may contact them
through the ff:");
$printer->feed(1);
$printer->setEmphasis(TRUE);
$printer->text("Email: ");
$printer->setEmphasis(FALSE);
$printer->text("crssupport@list.upd.edu.ph\n");
$printer->setEmphasis(TRUE);
$printer->text("Phone: ");
$printer->setEmphasis(FALSE);
$printer->text("981-8500 local 4560\n");
$printer->feed(2);

$printer->setTextSize(2,1);
$printer->text("ONLINE SERVICES\n");
$printer->initialize();
$printer->setEmphasis(TRUE);
$printer->text("DILNET ACCOUNT: ");
$printer->setEmphasis(FALSE);
$printer->text("https://accounts.upd.edu.ph\n");

$printer->setEmphasis(TRUE);
$printer->text("UPD WEBMAIL: ");
$printer->setEmphasis(FALSE);
$printer->text("https://mail.upd.edu.ph\n");

$printer->setEmphasis(TRUE);
$printer->text("CRS: ");
$printer->setEmphasis(FALSE);
$printer->text("https://crs.upd.edu.ph\n");

$printer->setEmphasis(TRUE);
$printer->text("UVLE: ");
$printer->setEmphasis(FALSE);
$printer->text("http://uvle.upd.edu.ph\n");

$printer->setEmphasis(TRUE);

```

```

$printer->text("Library (iLib): ");
$printer->setEmphasis(FALSE);
$printer->text("http://ilib.upd.edu.ph\n");
$printer->feed(2);

$printer->initialize();
$printer->setTextSize(2,1);
$printer->text("DILNET HELPDESK\n");
$printer->setTextSize(1,1);
$printer->text("Contact us if you have problems with the ff:
\n");

$printer->text("- Connecting to DILNET Wifi\n- UPD Webmail
(mail.upd.edu.ph)\n- Dilnet Accounts (accounts.upd.edu.ph)\n");
$printer->feed(1);
$printer->setEmphasis(TRUE);
$printer->text("Email: ");
$printer->setEmphasis(FALSE);
$printer->text("helpdesk@upd.edu.ph\n");
$printer->setEmphasis(TRUE);
$printer->text("Phone: ");
$printer->setEmphasis(FALSE);
$printer->text("981-8500 local 2050\n");
$printer->setEmphasis(TRUE);
$printer->text("Website: ");
$printer->setEmphasis(FALSE);
$printer->text("http://dilnet.upd.edu.ph\n");
$printer->setEmphasis(TRUE);
$printer->text("Facebook: ");
$printer->setEmphasis(FALSE);
$printer->text("http://www.facebook.com/updilnet\n");
$printer->setEmphasis(TRUE);
$printer->text("Twitter: ");
$printer->setEmphasis(FALSE);
$printer->text("http://www.twitter.com/dilnet\n");
$printer->feed(2);

$date = date("Y-m-d H:i:s");
$printer->text("Date/Time Generated: ".$date."\n");
$printer->feed(1);
$printer -> cut();
$printer -> close();

}

redirect("accounts");
}

/* Checker functions */
function _is_admin_user($empno)
{
    $this->form_validation->set_message('_is_admin_user', 'You do not
have admin privileges. ');
    $result = $this->admin_model->get_by_employee_number($empno);

    if ( $result != NULL ) {
        if(isset($result["role"]) && $result["role"] != 0) { return
TRUE; }
        else { return FALSE; }
    }
}

```

```

        return FALSE;
    } // function.is_admin_user

    public function _send_welcome_mail($details) {

        $this->load->library('email');

        $this->email->from('support@upd.edu.ph', 'UPD Webmail Team');
        $this->email->to($details["user"]["mail"]);
        // $this->email->to("postmaster@upd.edu.ph");
        $this->email->subject('Welcome to UP Diliman Webmail!');
        $this->email->set_mailtype('html');

        $contents = $this->load->view('mailer/welcome', $details, TRUE);
        $this->email->message($contents);

        $this->email->send();
    } // function._send_welcome_mail

} // class.Activation
/* End of file welcome.php */
/* Location: ./application/controllers/welcome.php */

```

application/controllers/Create_multiple_employees.php

```

<?php if ( ! defined('BASEPATH')) exit('No direct script access allowed');

class Create_multiple_employees extends MY_Controller {

    public function __construct()
    {
        parent::__construct();
        $this->load->helper(array('form', 'url'));
        $this->form_validation->set_error_delimiters('<label>', '</label>');
        $this->load->model('ldap_model');
        $this->load->model('mysql_model');
        $this->load->model('crs_model');
        $this->load->model('log_model');

        $this->page_vars = array(
            'content' => 'upload_form',
            'page_title' => 'Create Multiple Employee Accounts',
            'page_subtitle' => 'Create multiple employee accounts from a CSV',
            'include_stylesheets' => 'account_user_create_css',
            'include_scripts' => 'create_multiple_employees_js'
        );
        $this->tech_user = $_SESSION['tech_user'];
        $this->ip = $this->input->ip_address();
    }

    public function index()
    {
        // $this->load->view('upload_form', array('error' => ' ' ));
        redirect('tech/create_user');
    }
}

```



```

}

public function do_upload()
{
    $data = $this->page_vars;

    $config['upload_path']      = './uploads/';
    $config['allowed_types']    = 'csv';
    // $config['max_size']      = 100;

    $this->load->library('upload', $config);

    if ( ! $this->upload->do_upload('userfile'))
    {
        $error = array('error' => $this->upload->display_errors());

        // $this->load->view('upload_form', $error);
        $data['content'] = 'upload_form';
        $this->load->vars($data);
        $this->load->view('page_template', $error);
    }
    else
    {
        $upload_data = array('upload_data' => $this->upload->data());
        $csv_filtered = array();
        $incomplete_details_count = 0;
        $csv = array_map('str_getcsv',
            file('./uploads/'. $upload_data['upload_data']['file_name']));
        // print "<pre>";
        // print_r($csv);
        // print "</pre>";

        foreach($csv as $line_entries)
        {
            $incomplete_details = false;
            if($line_entries[0]=="lastname")
            {
                // this is a header
                continue;
            }
            else
            {
                if(isset($line_entries[0]) &&
                    ctype_alpha(str_replace(array(' ', '"', '-'), '', $line_entries[0])))
                {
                    $lastname = $line_entries[0];
                }
                else
                {
                    $incomplete_details = true;
                }

                if(isset($line_entries[1]) &&
                    ctype_alpha(str_replace(array(' ', '"', '-'), '', $line_entries[1])))
                {
                    $firstname = $line_entries[1];
                }
            }
        }
    }
}

```

```

else
{
    $incomplete_details = true;
}

ctype_alpha(str_replace(array(' ', '"', '-'), '', $line_entries[2]))
{
    $middlename = $line_entries[2];
}
else
{
    $incomplete_details = true;
}

ctype_alpha(str_replace(array(' ', '"', '-'), '', $line_entries[3]))
{
    $unit = $line_entries[3];
}
else
{
    $incomplete_details = true;
}

ctype_alpha(str_replace(array(' ', '"', '-'), '', $line_entries[4]))
{
    $division = $line_entries[4];
}
else
{
    $incomplete_details = true;
}

if(isset($line_entries[5]) &&
!empty($line_entries[5]))
{
    $designation = $line_entries[5];
}
else
{
    $incomplete_details = true;
}

ctype_alpha($line_entries[6])
{
    $type = $line_entries[6];
}
else
{
    $incomplete_details = true;
}

ctype_digit($line_entries[7])
{
    $employeeno = $line_entries[7];
}

```

```

        }
        else
        {
            $incomplete_details = true;
        }
        if(!$incomplete_details)
        {
            $csv_filtered[] = array($lastname,

$firstname,

$middlename,

$unit,

$division,

$designation,

$type,

$employeenno
);

        }
        else
        {
            $incomplete_details_count++;
        }
    }
}

$data['incomplete_details_count'] = $incomplete_details_count;
$data['csv_filtered'] = $csv_filtered;
$data['content'] = 'upload_success';
$_SESSION['multiple_accounts_info'] = $csv_filtered;
$this->load->vars($data);
$this->load->view('page_template');
}

}

public function activate_multiple()
{
    $data = $this->page_vars;
    if(!isset($_SESSION['multiple_accounts_info']) ||
count($_SESSION['multiple_accounts_info'])<=0)
    {
        $_SESSION['error'] = "Invalid data for multiple account
creation.";
        $this->session->mark_as_flash('error');
        redirect('tech/create_user');
    }
    else
    {
        function _is_not_null ($var) {

```

```

        if(!is_null($var) || $var != NULL || $var != '') {
return true; } else { return false; }
    }

    // print "Creating and activating accounts ...";
    $accounts = $_SESSION['multiple_accounts_info'];
    $activated_accounts = array();
    foreach($accounts as $account)
    {
        $first = isset($account[1]) ? ucwords($account[1]) :
NULL;
        $middle = isset($account[2]) ? ucwords($account[2]) :
NULL;
        $last = isset($account[0]) ? ucwords($account[0]) :
NULL;

        $username = $this->_generate_username($first, $middle,
        $last);
        $account_email = $this->_generate_user_email($first,
        $middle, $last);

        $ldap_data = array(
            'uid' => $username,
            'cn' => $first." ".$last,
            'sn' => $last,
            'mail' => $account_email."@upd.edu.ph",
            'employeenumber' => $account[7],
            'ou' => strtoupper($account[3]),
            'employeetype' => strtoupper($account[6]),
            'title' => ucwords($account[5]),
            'mailquotasize' => 419430400,
            'departmentnumber' => strtoupper($account[4])
        );

        if(empty($ldap_data['sn'])){
            $ldap_data['sn'] = "NO_LAST_NAME";
        }

        $ldap_data["objectclass"][0] = "top";
        $ldap_data["objectclass"][1] = "person";
        $ldap_data["objectclass"][2] = "inetOrgPerson";
        $ldap_data["objectclass"][3] = "qmailUser";
        $ldap_data["objectclass"][4] = "radiusprofile";

        $ldap_data["dialupAccess"] = 'TRUE';

        $ldap_data["mailHost"] = 'imap4.upd.edu.ph';
        $ldap_data["mailAlternateAddress"][0] =
$username."@upd.edu.ph";
        $ldap_data['mailMessageStore'] =
'upd.edu.ph/'.$ldap_data['ou'].'/'.$ldap_data['departmentnumber'].'/'.$username;

        # generate password
        $this->load->helper('string');
        $password = random_password();
        mt_srand((double)microtime()*1000000);
        $salt = pack("CCCC", mt_rand(), mt_rand(), mt_rand(),
        mt_rand());
    }

```

```

        $ldap_data["userPassword"] = "{SSHA}" .
base64_encode(pack("H*", sha1($password . $salt)) . $salt);

        # Callback function for array_filter

        #Remove keys of NULL and False values
        $ldap_data = array_filter( $ldap_data, '_is_not_null'
);

        # Insert to ldap
        $result = $ldapresult = $this->ldap_model-
>add_ldapentry($username, $ldap_data);

        if ($result) {

            $this->log_model->create($username, "Account
added to LDAP", "activation", $this->tech_user, $this->ip, 'info');

            # Insert to mysql
            #TODO Add Unit, Course, Dept
            $mysql_data = array(
                'username' => $username,
                'email' => $account_email,
                'activated_by'=>$_SESSION['tech_user'],
                'ip_activated'=>$this->input-
>ip_address(),

                'activated' => 1,
                'firstname' => $first,
                'lastname' => $last,
                'middlename' => $middle,
                'id_number' => $account[7],
                'empno' => $account[7],
                'designation' => ucwords($account[5]),
            );

            # Remove keys with NULL and FALSE values
            $mysql_data = array_filter( $mysql_data,
'_is_not_null' );

            # Send Welcome Email
            $welcome_data['user'] = array(
                'email' => $ldap_data['mail'],
                'firstname' => $mysql_data['firstname'],
                'username' => $username
            );

            $this->_send_welcome_mail($welcome_data);
            $this->log_model->create($username, "Sending
welcome email", "activation", $this->tech_user, $this->ip, 'debug');

            #TODO Check correct return values
            if($this->mysql_model->create($username,
$mysql_data)){
                $this->log_model->create($username,
                $this->tech_user, $this->ip, 'info');

                # Prepare details for printing, forward to
printing function

```

```

        $activated_accounts[] = array($username,
$password);

        //
$_SESSION['create_account_print']['username'] = $username;
        //
$_SESSION['create_account_print']['password'] = $password;
        // $this->session-
>mark_as_flash('create_account_print');
    }
}
}
$_SESSION['activated_accounts'] = $activated_accounts;
redirect('tech/print_account/confirm_multiple_print/');
}
$this->load->vars($data);
$this->load->view('page_template');
}
}
}

```

application/controllers/Errors.php

```

<?php
defined('BASEPATH') OR exit('No direct script access allowed');

class Errors extends MY_Controller {

    /**
     * Index Page for this controller.
     *
     * Maps to the following URL
     *      http://example.com/index.php/welcome
     *    - or -
     *      http://example.com/index.php/welcome/index
     *    - or -
     * Since this controller is set as the default controller in
     * config/routes.php, it's displayed at http://example.com/
     *
     * So any other public methods not prefixed with an underscore will
     * map to /index.php/welcome/<method_name>
     * @see https://codeigniter.com/user_guide/general/urls.html
     */
    public function index()
    {
        $this->page_not_found();
    }

    public function page_not_found()
    {
        $data = array(
            'content' => '404',
            'page_title' => 'Page Not Found',

```

```

        'page_subtitle' => null,
        'include_stylesheets' => null,
        'include_scripts' => null
    );

    $this->load->vars($data);
    $this->load->view('page_template');
}

public function restricted() {
    $data = array(
        'content' => 'restricted',
        'page_title' => 'Restricted Access',
        'page_subtitle' => null,
        'include_stylesheets' => null,
        'include_scripts' => null
    );

    $this->load->vars($data);
    $this->load->view('page_template');
}
}

```

application/controllers/Welcome.php

```

<?php if ( ! defined('BASEPATH')) exit('No direct script access allowed');

```

```

class Welcome extends MY_Controller {

```

```

/**
 * Index Page for this controller.
 *
 * Maps to the following URL
 *      http://example.com/index.php/welcome
 * - or -
 *      http://example.com/index.php/welcome/index
 * - or -
 * Since this controller is set as the default controller in
 * config/routes.php, it's displayed at http://example.com/
 *
 * So any other public methods not prefixed with an underscore will
 * map to /index.php/welcome/<method_name>
 * @see http://codeigniter.com/user_guide/general/urls.html
 */
function __construct(){
    parent::__construct();

    $this->form_validation->set_error_delimiters('<label>', '</label>');
    $this->load->model('ldap_model');
    $this->load->model('user_model');
}

public function index()
{
    #TOFIX: Log Messages

```

```

Log_message('debug', "trying to log in");
$this->session->set_userdata('logged_in',TRUE);

if(!$this->session->userdata('logged_in')){
    Log_message('debug', "trying to log in");
    redirect('welcome/login/');
}

else {
    #Redirect to Homepage
    #TOFIX: Should change depending on role
    redirect('tech/dilnet_lookup');
}

}

public function login(){
    if($this->session->logged_in == TRUE){
        $this->session->set_flashdata('warning', 'User is already
logged in. ');
        redirect($this->session->homepage);
    }

    else {
        $this->form_validation->set_rules('username', 'Username',
'required|alpha_numeric|callback__check_user_in_db');
        $this->form_validation->set_rules('password', 'Password',
'required|callback__ldaplogin');

        if ($this->form_validation->run() == TRUE){
            $username = $this->input->post('username');

            #TODO This line should be in admin_model

            // Set the session data
            $_SESSION['tech_user'] = $username;
            $_SESSION['logged_in'] = TRUE;
            $_SESSION['homepage'] = 'tech/dashboard';
            $this->user_model->admin_last_login($username);

            if ($this->session->next)
                redirect($this->session->next);
            else
                redirect('/');
        }
        else{
            $this->load->view('login');
        }
    }

}

}

public function logout()
{
    // Just set logged_in to FALSE and then destroy everything for good
    unset($_SESSION['tech_user']);
    unset($_SESSION['logged_in']);
}

```

measure


```
unset($_SESSION);
session_destroy();

$this->load->view('logout');
```

```
}
```

```
function _ldaplogin($password){
    $username = $this->input->post('username');

    if(isset($username) || $username != NULL){
        $ldapauth = $this->ldap_model->_authenticate($username,$password);

        if( $ldapauth == "Success"){
            return TRUE;
        }
        else {
            $this->form_validation->set_message('username', NULL);
            $this->form_validation->set_message('_ldaplogin',
'Cannot log in. <br>Reason: '.$ldapauth);
            return FALSE;
        }
    }
    else {
        $this->form_validation->set_message('username', 'The Username
field is required.');
```

```
        return FALSE;
    }
}
```

```
function _check_user_in_db($username)
{
    if($this->user_model->admin_user_exists($username)){
        return TRUE;
    }

    $this->form_validation->set_message('_check_user_in_db', 'This user
is not allowed to access this system.');
```

```
    return FALSE;
}
```

```
}
```

```
application/core/MY_Controller.php
```

```
<?php if ( ! defined('BASEPATH')) exit('No direct script access allowed');
```

```
class MY_Controller extends CI_Controller {
```

```
    public function __construct()
    {
```

```
        parent::__construct();
        date_default_timezone_set ( 'Asia/Manila' );
        $this->_check_auth();
    }
```

```

}

private function _check_auth()
{
    $logged_in = $this->session->userdata('logged_in');

    //If user is not logged in, check IP
    //Added total segments as condition so that error flashdata will not show when
base url is accessed
    if(!isset($_SESSION['tech_user']) || !isset($logged_in) || $logged_in != true
&& $this->uri->total_segments() != 0)
    {
        if($this->uri->uri_string != 'welcome/login'){

            if($this->uri->uri_string != 'welcome/logout'){

                // Save the page we are on now to redirect if the user
                // successfully authenticates
                $_SESSION['next'] = $this->uri->uri_string;
                $_SESSION['error'] = 'You need to log in to access this page.';
                $this->session->mark_as_flash('error');

            }

            redirect('welcome/login');
        }
    }
    //Check permissions
    else {
        // Get username and pass to the config array
        $config = array('userID'=>$_SESSION['tech_user']);

        $this->load->library('acl',$config);

        // Get current controller
        $current_controller = $this->uri->segment(2);
        //TODO: Redirect to 404 if not in the list of permissions

        if($this->uri->segment(1) === 'welcome') { return; }
        if($current_controller != NULL){
            // If the user does not have permission, redirect to restricted page
            if ( !$this->acl->hasPermission($current_controller) && $this->uri-
>segment(2) != 'restricted') {
                redirect('errors/restricted');
            }
        }
    }
    return;
}

```

```

function _format_array_model2ctrl($array, $arr_index, $type, $page_type =
'view') {

```

```

    $value = NULL;

    if ($type == 'ldap'){

```

```

    if(isset($array[$arr_index])){
        if (count($array[$arr_index]) == 1){
            return $array[$arr_index][0];
        }
        else return $array[$arr_index];
    }
    else {
        if ($page_type == 'form'){
            $value = NULL;
        }
        else if ($page_type == 'view'){
            $value = '<strong>No data</strong>';
        }
    }
}

if ($type == 'db'){
    if(isset($array[$arr_index])){
        return $array[$arr_index];
    }
    else {
        if ($page_type == 'form'){
            $value = NULL;
        }
        if ($page_type == 'print'){
            $value = 'NO DATA';
        }
        else if ($page_type == 'view'){
            $value = '<strong>No data<strong>';
        }
    }
}

return $value;
}

function _format_array_ctrl2model($array, $arr_index) {

    $value = NULL;

    if(isset($array[$arr_index])){
        if ($array[$arr_index] != NULL){
            return $array[$arr_index];
        }
        return NULL;
    }
}

```

```

# searches if $user exists in
# mysql(olddb & user table) - as username, userid, or email
# ldap - as mail or mailalternateaddress
# Todo after migration:
#   remove "freshmen" check
#   remove "users_old" check
#   add check on deleted_table

```

```

function _is_duplicate($account) {
    $this->load->model('mysql_model');
}

```

```

$this->load->model('ldap_model');
# check employee number duplicate in ldap
if(isset($account['id_number']) && $account['id_number'] != NULL){
    $id['id_number'] = $id['empno'] = $account['id_number'];
    $ldap_result = $this->ldap_model->get_by_id_number($account['id_number']);
    if ($ldap_result["count"] != 0) { return TRUE; }
}
# check student number duplicate in ldap
if(isset($account['studentno']) && $account['studentno'] != NULL){
    $id['studno'] = $account['studentno'];
    $ldap_result = $this->ldap_model->get_by_id_number($account['studentno']);
    if ($ldap_result["count"] != 0) { return TRUE; }
}
# After checking ldap (above), check mysql
if (!empty($id)) {
    $mysql_result = $this->mysql_model->get_by_id_number($id);
    if($mysql_result){
        return TRUE;
    }
}
return FALSE;
}

# checks if user is a special case, returns true or false
# special case = perfect match of fullname and no last and middle
function _is_special_case($first, $middle, $last) {
    # TODO optimize
    if ((!isset($middle)) && (!isset($last))) {
        return TRUE;
    }
    else if (($middle == NULL) && ($last == NULL)) {
        return TRUE;
    }
}

$this->load->model('mysql_model');

$test = $this->mysql_model->get_by_fullname($first, $middle, $last,
"users_old");
if (is_array($test) && !empty($test)) { return TRUE; }
$test = $this->mysql_model->get_by_fullname($first, $middle, $last, "users");
if (is_array($test) && !empty($test)) { return TRUE; }
#Middle Initial same
if(isset($middle) && $middle != NULL){
    $middle = $middle[0];
    $test = $this->mysql_model->get_by_fullname_dup($first, $middle, $last);
    if (is_array($test) && !empty($test)) { return TRUE; }
}
}

# assume input parameters are already in lowercase
# returns available username for user
function _generate_username($first, $middle, $last) {
    $middle = ($middle == NULL) ? "" : strtolower($middle[0]); # get middle
    initial or set to blank

    if ($last == NULL) { # special case
        $username = $this->_format_username($first);
    } else {

```

```

$username = $this->_format_username($first[0].$middle.$last); # fmlastname
}

$test = $this->_check_user_existence($username); # check existence
if ($test) {
    for ($i = 1; ; $i++) {
        $usernameX = $username.$i; # fmlastnameX
        $test = $this->_check_user_existence($usernameX); # check existence
        if (!$test) {
            $username = $usernameX;
            break;
        }
    }
}
// $this->Logs_model->create($username, "generated username", "activation");
return $username;
}

# returns available email address for user
function _generate_user_email($first, $middle, $last) {
    $middle = ($middle == NULL) ? "" : ".$middle[0]; # get
    $last = ($last == NULL) ? "" : ".$last; # get .lastname or set
    $user_email = $this->_format_user_email($first.$last); #
    $test = $this->_check_user_existence($user_email); # check
    if ($test) {
        $user_email = $this->_format_user_email($first.$middle.$last);
        $test = $this->_check_user_existence($user_email); #
    }
    if (!$test) {
        for ($i = 1; ; $i++) {
            $user_email = $this->_format_user_email($first.$last."_".$i); #
            $test = $this->_check_user_existence($user_email); # check
        }
    }
    return $user_email;
}

function _format_user_email($user_email) {
    $user_email = str_replace(" ", "_", $user_email);
    $user_email = str_replace("-", "_", $user_email);
    return strtolower($user_email);
}

function _format_username($username) {
    $username = trim($username, "\t\n\r\0\x0B");
    $username = str_replace(" ", "", $username);
    $username = str_replace("-", "", $username);
    $username = str_replace(".", "", $username);
    return strtolower($username);
}

```

```

function _check_user_existence($user) {
    // $this->load->model('mysql_model');
    $this->load->model('ldap_model');
    #TODO Check both mysql and ldap! (Baka di na kailangan kung sync lagi ang LDAP
at MySQL)
    #$test = $this->mysql_model->get_by_email_or_username($user);

    $test = $this->ldap_model->search_exact($user);
    # Check LDAP if $test is 1 (mysql might not have a copy but LDAP might)
    if ($test) {
        return $test;
    }
    return FALSE;
}

public function _send_welcome_mail($details) {

    // $config = Array(
    //     'protocol' => 'smtp',
    //     'smtp_host' => 'smtp.upd.edu.ph',
    //     'smtp_port' => 25,
    //     'charset'   => 'iso-8859-1'
    // );

    // $this->load->library('email',$config);
    $this->load->library('email');

    $this->email->from('support-NO-REPLY@upd.edu.ph', 'UPD Webmail Team');
    $this->email->to($details["user"]["email"]);
    $this->email->subject('Welcome to UP Diliman Webmail!');
    $this->email->set_mailtype('html');

    $contents = $this->load->view('mailer/welcome', $details, TRUE);
    $this->email->message($contents);

    $this->email->send();
}

// from MY_Controller of SAM
# check if current user is in admin_users
# check if current user's role is included in $val
# else redirect
function _check_admin($val = null) {
    $username = $this->session->userdata("user");
    if ($val == null) {
        if(!$this->admin_model->admin_user_exists($username)){
            if(!isset($this->session->flashdata('error', 'Unauthorized access.'))){
                $this->session->set_flashdata('error', 'Unauthorized access.');
```



```

function _format_studentno($studno, $dash = TRUE){
    // maglalagay ng dash
    if($dash && !(strpos($studno,"-") == 4)){
        $year = substr($studno, 0, 4);
        $num = substr($studno, 4, 5);

        $studno = $year."-".$num;
    }

    //magtatanggal ng dash
    else if(!$dash && (strpos($studno,"-")==4)){
        $studno = str_replace("-", "", $studno);
    }

    return $studno;
}

/**
 * This is the Error Page
 * Set Message through error flashdata
 */
function error() {
    $data = $this->page_vars;

    $data['errmsg'] = 'Oops, something went wrong.';

    if ($this->session->flashdata('error') != null) {
        $data['errmsg'] = $this->session->flashdata('error');
    }
    $data['errmsg'] = '<h2 class="text-danger"><i class="glyphicon glyphicon-
exclamation-sign"></i></h2>
    <h5 class="text-danger">'. $data['errmsg']. '</h5>
    <p>Click OK to go back to the home page.</p>';
    $data['content'] = 'partials/error';
    $this->load->vars($data);
    $this->load->view('page_template');
}
}

```

application/core/MY_Loader.php

```

<?php if (! defined('BASEPATH')) exit('No direct script access allowed');
/**
 * Sparks
 *
 * An open source application development framework for PHP 5.1.6 or newer
 *
 * @package      CodeIgniter
 * @author       CodeIgniter Reactor Dev Team
 * @author       Kenny Katzgrau <katzgrau@gmail.com>
 * @since        CodeIgniter Version 1.0
 * @filesource
 */

```



```

* Loader Class
*
* Loads views and files
*
* @package      CodeIgniter
* @subpackage   Libraries
* @author       CodeIgniter Reactor Dev Team
* @author       Kenny Katzgrau <katzgrau@gmail.com>
* @category     Loader
* @link         http://codeigniter.com/user_guide/libraries/loader.html
*/
class MY_Loader extends CI_Loader
{
    /**
     * Keep track of which sparks are loaded. This will come in handy for being
     * speedy about loading files later.
     *
     * @var array
     */
    var $_ci_loaded_sparks = array();

    /**
     * Is this version less than CI 2.1.0? If so, accomodate
     * @bubbafoley's world-destroying change at: http://bit.ly/sIqR7H
     * @var bool
     */
    var $_is_lt_210 = false;

    /**
     * Constructor. Define SPARKPATH if it doesn't exist, initialize parent
     */
    function __construct()
    {
        if(!defined('SPARKPATH'))
        {
            define('SPARKPATH', 'sparks/');
        }

        $this->_is_lt_210 = (is_callable(array('CI_Loader', 'ci_autoloader'))
            || is_callable(array('CI_Loader',
'_ci_autoloader'))));

        parent::__construct();
    }

    /**
     * To accomodate CI 2.1.0, we override the initialize() method instead of
     * the ci_autoloader() method. Once sparks is integrated into CI, we
     * can avoid the awkward version-specific logic.
     * @return Loader
     */
    function initialize()
    {
        parent::initialize();

        if(!$this->_is_lt_210)
        {
            $this->ci_autoloader();
        }
    }
}

```

```

    return $this;
}

/**
 * Load a spark by it's path within the sparks directory defined by
 * SPARKPATH, such as 'markdown/1.0'
 * @param string $spark The spark path within the sparks directory
 * @param <type> $autoload An optional array of items to autoload
 * in the format of:
 *   array (
 *     'helper' => array('somehelper')
 *   )
 * @return <type>
 */
function spark($spark, $autoload = array())
{
    if(is_array($spark))
    {
        foreach($spark as $s)
        {
            $this->spark($s);
        }
    }

    $spark = ltrim($spark, '/');
    $spark = rtrim($spark, '/');

    $spark_path = SPARKPATH . $spark . '/';
    $parts      = explode('/', $spark);
    $spark_slug = strtolower($parts[0]);

    # If we've already loaded this spark, bail
    if(array_key_exists($spark_slug, $this->_ci_loaded_sparks))
    {
        return true;
    }

    # Check that it exists. CI Doesn't check package existence by itself
    if(!file_exists($spark_path))
    {
        show_error("Cannot find spark path at $spark_path");
    }

    if(count($parts) == 2)
    {
        $this->_ci_loaded_sparks[$spark_slug] = $spark;
    }

    $this->add_package_path($spark_path);

    foreach($autoload as $type => $read)
    {
        if($type == 'library')
            $this->library($read);
        elseif($type == 'model')
            $this->model($read);
        elseif($type == 'config')
            $this->config($read);
    }
}

```

```

        elseif($type == 'helper')
            $this->helper($read);
        elseif($type == 'view')
            $this->view($read);
        else
            show_error ("Could not autoload object of type '$type' ($read) for
spark $spark");
    }

```

```

    // Looks for a spark's specific autoloader
    $this->ci_autoloader($spark_path);

    return true;
}

```

```

/**
 * Pre-CI 2.0.3 method for backward compatibility.
 *
 * @param null $basepath
 * @return void
 */

```

```

function _ci_autoloader($basepath = NULL)
{
    $this->ci_autoloader($basepath);
}

```

```

/**
 * Specific Autoloader (99% ripped from the parent)
 *
 * The config/autoload.php file contains an array that permits sub-systems,
 * libraries, and helpers to be loaded automatically.
 *
 * @param array|null $basepath
 * @return void
 */

```

```

function ci_autoloader($basepath = NULL)
{
    if($basepath !== NULL)
    {
        $autoload_path = $basepath.'config/autoload'.EXT;
    }
    else
    {
        $autoload_path = APPPATH.'config/autoload'.EXT;
    }

    if(! file_exists($autoload_path))
    {
        return FALSE;
    }

    include($autoload_path);

    if ( ! isset($autoload))
    {
        return FALSE;
    }

    if($this->_is_lt_210 || $basepath !== NULL)

```

```

{
    // Autoload packages
    if (isset($autoload['packages']))
    {
        foreach ($autoload['packages'] as $package_path)
        {
            $this->add_package_path($package_path);
        }
    }
}

// Autoload sparks
if (isset($autoload['sparks']))
{
    foreach ($autoload['sparks'] as $spark)
    {
        $this->spark($spark);
    }
}

if($this->_is_lt_210 || $basepath !== NULL)
{
    if (isset($autoload['config']))
    {
        // Load any custom config file
        if (count($autoload['config']) > 0)
        {
            $CI =& get_instance();
            foreach ($autoload['config'] as $key => $val)
            {
                $CI->config->load($val);
            }
        }
    }

    // Autoload helpers and languages
    foreach (array('helper', 'language') as $type)
    {
        if (isset($autoload[$type]) AND count($autoload[$type]) > 0)
        {
            $this->$type($autoload[$type]);
        }
    }

    // A little tweak to remain backward compatible
    // The $autoload['core'] item was deprecated
    if ( ! isset($autoload['libraries']) AND isset($autoload['core']))
    {
        $autoload['libraries'] = $autoload['core'];
    }

    // Load libraries
    if (isset($autoload['libraries']) AND count($autoload['libraries']) >
0)
    {
        // Load the database driver.
        if (in_array('database', $autoload['libraries']))
        {
            $this->database();
        }
    }
}

```

```

        $autoload['libraries'] = array_diff($autoload['libraries'],
array('database'));
    }

    // Load all other libraries
    foreach ($autoload['libraries'] as $item)
    {
        $this->library($item);
    }
}

// Autoload models
if (isset($autoload['model']))
{
    $this->model($autoload['model']);
}
}
}
}
}

```

application/core/MY_Model.php

```

<?php if ( ! defined('BASEPATH')) exit('No direct script access allowed');

class MY_Model extends CI_Model {

    public function __construct()
    {
        parent::__construct();
        date_default_timezone_set ( 'Asia/Manila' );
    }

    public function _audit($msg){
        $date = date('Y/m/d H:i:s');
        if( ! file_put_contents($this->auditlog, $date." : ".$msg."\n",FILE_APPEND))
        {
            log_message('info', 'Error opening audit log '.$this->auditlog);
            return FALSE;
        }
        return TRUE;
    }

    //
    // from SAM My_Model
    function _trim_array($var_array, $info_array){
        $prep_array = array();
        foreach ($var_array as $key => $var){
            foreach ($var_array as $key => $var){
                if(isset($info_array[$var])){
                    $prep_array[$var] = $info_array[$var];
                }
                else {
                    $prep_array[$var] = "";
                }
            }
        }
        return $prep_array;
    }
}

```

```

}

function _stripdash($str){
    $str_strip = str_replace("-", "", $str);
    return $str_strip;
}
// end of SAM My_Model
}

```

application/helpers/nav_helper.php

```

<?php if ( ! defined('BASEPATH')) exit('No direct script access allowed');
if ( ! function_exists('active_link'))
{
    function active_Link($controller)
    {
        $CI =& get_instance();

        $class = $CI->router->fetch_class();

        return ($class == $controller) ? 'active' : '';
    }
}

```

application/helpers/prints_helper.php

```

<?php if ( ! defined('BASEPATH')) exit('No direct script access allowed');

function print_s ($stuff) {
    echo '<pre>' . print_r($stuff, true) . ' -- ';
    var_dump($stuff);
    echo '</pre>';
}

function object_to_array($object) {
    return json_decode(json_encode($object), true);
}

function create_dropdown_array($key_name, $value_name, $object_array) {
    $object_array = object_to_array($object_array);
    foreach ($object_array as $object) {
        foreach ($object as $key => $value) {
            foreach ($object as $key => $value) {
                if ($key_name == $key) $keys[] = $value;
                if ($value_name == $key) $values[] = $value;
            }
        }
    }
    return array_combine($keys,$values);
}

function gender($value) {
    if ($value == '1') { return "Female"; }
    return "Male";
}

```

```

function random_password($length = 8) {
    $characters = '23456789abcdefghijklmnopqrstuvwxyzACDEFGHJKMNPQRSTUVWXYZ';
    $string = "";
    for ($p = 0; $p < $length; $p++) {
        $string .= $characters[mt_rand(0, strlen($characters)-1)];
    }

    return $string;
}

function arrayRecursiveDiff($aArray1, $aArray2) {
    $aReturn = array();

    foreach ($aArray1 as $mKey => $mValue) {
        if (array_key_exists($mKey, $aArray2)) {
            if (is_array($mValue)) {
                $aRecursiveDiff = arrayRecursiveDiff($mValue, $aArray2[$mKey]);
                if (count($aRecursiveDiff)) { $aReturn[$mKey] = $aRecursiveDiff; }
            } else {
                if ($mValue != $aArray2[$mKey]) {
                    $aReturn[$mKey] = $mValue;
                }
            }
        } else {
            $aReturn[$mKey] = $mValue;
        }
    }
    return $aReturn;
}

function myArrayDiff($A, $B) {
    $intersect = array_intersect($A, $B);
    return array_merge(array_diff($A, $intersect), array_diff($B, $intersect));
}

```

application/models/Account_model.php

```

<?php if ( ! defined('BASEPATH')) exit('No direct script access allowed');

class Account_model extends MY_Model {

    function __construct(){
        parent::__construct();
        $this->table = 'users';

        $this->load->model('crs_model');
        $this->load->model('sam_user_model');
        $this->load->model('sam_ldap_model');
        $this->load->model('units_model');
        $this->load->model('log_model');
        $this->load->model('log_model');
        $this->auditlog = '/tmp/activation.log';
    }
}

```

```

function is_enrolled_by_barcode($barcode) {
    $data["details"] = $this->crs_model->_isenrolledbybarcode($barcode);
    if( isset($data["details"]["isEnrolled"]) && $data["details"]["isEnrolled"] ==
TRUE ) {
        $data["success"] = TRUE;
    } else {
        $data["details"]["errmsg"] = "Student not enrolled by barcode.";
        $data["success"] = FALSE;
    }
    return $data;
}

function is_enrolled_by_studentno($student_number) {
    $student_number = $this->_stripdash($student_number);
    $acad_year = date("Y");
    $month_number = date("n");
    if($month_number >=1 && $month_number <=5){ // 2nd sem
        $termno = 2;
    } else if($month_number >= 6 && $month_number <=7 ){
        $termno = 3;
    } else if($month_number >=8 && $month_number <=12 ){
        $termno = 1;
    }
    $data["details"] = $this->crs_model->_isenrolledbystudentno($student_number,
$acad_year, $termno);
    // print_r($data["details"]);
    if( isset($data["details"]["isEnrolled"]) && $data["details"]["isEnrolled"] ==
TRUE) {
        $data["success"] = TRUE;
    } else {
        $data["details"]["errmsg"] = "Student not enrolled by student number.";
        $data["success"] = FALSE;
    }
    return $data;
}

function is_eligible_for_creation($student_number) {
    $eligible_crs = TRUE;
    $eligible_crs = $this->is_enrolled_by_studentno($student_number);
    $eligible_mysql = $this->sam_user_model->
    $eligible_mysql = $this->sam_user_model->is_eligible_for_creation($student_number);
    $eligible_ldap = $this->sam_ldap_model->
    $eligible_ldap = $this->sam_ldap_model->is_eligible_for_creation($student_number);
    $eligible_ldap = $this->sam_ldap_model->is_eligible_for_creation($student_number);

    /*
    if($student_number=="2012-99999") { // test case
        $eligible_crs = $eligible_mysql = $eligible_ldap = TRUE;
    }
    /**/

    $result['success'] = FALSE;
    if ( !$eligible_mysql ) {
        $result['message'] = "Student number already has a Dilnet Account (MYSQL)";
    }
    else if ( !$eligible_ldap ) {
        $result['message'] = "Student number already has a Dilnet Account (LDAP)";
    }
    else if ( !$eligible_crs ) {

```



```

        $result['message'] = "Student not enrolled for current term. Please resolve
with CRS.";
    }
    else if ( $eligible_crs && $eligible_mysql && $eligible_ldap ) {
        $result['message'] = "Student Number is eligible for new account.";
        $result['success'] = TRUE;
    }
    return $result;
}

function is_eligible_for_reset($student_number) {
    #Should check if this student has an existing DilNet account
    $eligible_crs = $this->is_enrolled_by_studentno($student_number);
    $eligible_ldap_creation = $this->sam_ldap_model-
>is_eligible_for_creation($student_number);

    #if account exists in ldap, and currently enrolled
    if ( !$eligible_ldap_creation["success"]) {
        $result['success'] = TRUE;
        $result['message'] = "Student eligible for reset.";

        $result['student'] = $this->account_model-
>get_signupinfo_from_crs($student_number);

        if ( $eligible_crs["success"] )
            { $result["is_enrolled"] = TRUE; }
        else
            { $result["is_enrolled"] = FALSE; }
    }

    #if account does not exist in ldap
    else if ( $eligible_ldap_creation["success"] ) {
        $result['success'] = FALSE;
        $result['message'] = "Student does not exist.";
    }

    return $result;
}

public function get_signupinfo_from_crs($studentno){
    $studentno = $this->_stripdash($studentno);
    $studentno = $this->_isenrolledbystudentno($studentno, 2014,
1);
    // $is_enrolled = $this->crs_model->_isenrolledbystudentno($studentno, 2014,
    // var_dump($is_enrolled);
    $is_enrolled = TRUE;
    if($is_enrolled == TRUE){
        $signupinfo_vars =
array('firstname','middlename','lastname','birthdate','gender',
'mainunit','subunit','program','abbrevmainunit',
'abbrevsubunit','abbrevprogram');
        $studentinfo = $this->crs_model->_getstudentinfo($studentno);
        if($studentinfo['errcode'] == 0) {
            if($studentinfo['success'] = TRUE;
                $signupinfo['success'] = $this->_trim_array($signupinfo_vars,
                $studentinfo["details"]);
            }
        }
    }
}

```

```

        $signupinfo["details"]["firstname"] =
trim($signupinfo["details"]["firstname"], "\t\n\r\0\x0B\0\x60");
        $signupinfo["details"]["middlename"] =
trim($signupinfo["details"]["middlename"], "\t\n\r\0\x0B\0\x60");
        $signupinfo["details"]["lastname"] =
trim($signupinfo["details"]["lastname"], "\t\n\r\0\x0B\0\x60");
        $signupinfo["details"]["abbrevmainunit"] =
trim($signupinfo["details"]["abbrevmainunit"], "\t\n\r\0\x0B");
        $signupinfo["details"]["abbrevsubunit"] =
trim($signupinfo["details"]["abbrevsubunit"], "\t\n\r\0\x0B");

        $signupinfo["details"]["mainunit_cc"] = $this->units_model-
>get_acronym_from_name($signupinfo["details"]["mainunit"]);
        $signupinfo["details"]["subunit_cc"] = $this->units_model-
>get_acronym_from_name($signupinfo["details"]["subunit"]);
    }
    else {
        $signupinfo['success'] = FALSE;
        $signupinfo['message'] = $studentinfo['errmsg'];
    }
} else {
    $signupinfo['success'] = FALSE;
    $signupinfo['message'] = "Student is not enrolled for this semester.";
}

return $signupinfo;

}

```

```

// Creates Student User Account (Mail, LDAP, MySQL)
// WARNING: function assumes that student number is already eligible
// WARNING: function assumes that student number is already eligible
public function create_student($student_number, $employee) {
    $ip = $this->input->ip_address();
    // Param: Student Number
    // Returns: Array of form shown below
    $crs_student = $this->get_signupinfo_from_crs($student_number);
    $this->_audit($student_number." : Fetching details from CRS");

    // testing purposes
    /*
    if($student_number == "2012-99999"){
        $crs_student["success"] = TRUE;
    }
    /**/

    if ( !$crs_student["success"] ) {
        $data["success"] = false;
        $data["message"] = "No data found from CRS.";
        $this->_audit($student_number." : No data found form CRS");
        $this->log_model->create_activation($student_number, "No data found from
CRS", 'debug', $employee, $ip);

        return $data;
    }

    // Generating Credentials
    $this->_audit($student_number." : Generating credentials");
    $crs_student = $crs_student["details"];

```

```

$name = $this->_sanitize_name($crs_student["firstname"],$crs_student["middlename"],$crs_student["lastname"]);
$credentials = $this->_generate_credentials($name);
$this->_audit($credentials["username"]." : Credentials generated");
$this->log_model->create_activation($student_number, "Credentials generated : ".$credentials["username"], 'debug', $employee, $ip);

// // Prepare Arrays
$unit = array( "main" => $crs_student["abbrevmainunit"], "sub" => $crs_student["abbrevsubunit"]);
if ($unit["sub"] == "") { $unit["sub"] = $unit["main"]; }

$mysql_data = $this->_prepare_student_data_mysql($student_number, $credentials, $name, $employee, $crs_student['birthdate']);
$ldap_data = $this->_prepare_student_data_ldap($student_number, $credentials, $name, $unit);

// Create accounts: Mysql, Ldap, Mail
$mysql_result = $this->sam_user_model->create($mysql_data);

if ($mysql_result) {
    $this->_audit($credentials["username"]." : Mysql user created");
    $this->log_model->create_activation($credentials["username"], "Mysql user created", 'debug', $employee, $ip);
    $ldap_result = $this->sam_ldap_model->add_ldapentry($ldap_data, $mail_result["details"], "student");
    if($ldap_result["success"]) {
        $this->_audit($credentials["username"]." : LDAP user created");
        $this->log_model->create_activation($credentials["username"], "LDAP user created", 'debug', $employee, $ip);
        // CRS TAG
        $crs_tag_result= $this->crs_model->_assignusernamestudent($credentials["username"],$student_number);
        if($crs_tag_result['success']){
            $this->_audit($credentials["username"]." : CRS linked to ".$student_number);
            $this->log_model->create($credentials["username"], "Dilnet account linked to ".$student_number, 'crs_tag', 'info', $employee, $ip);
            $data["crs_tag"]["success"] = TRUE;
            $data["crs_status"] = "Tagged";
            $this->_send_crs_email($credentials["username"], $student_number, FALSE);
        } else { #CRS TAG FAIL
            $this->_audit($credentials["username"]." : CRS Tag Error : Failed to link ".$credentials["username"]." to ".$student_number);
            $this->_audit($credentials["username"]." : CRS Tag Error : ".$crs_tag_result["errmsg"]);
            $this->log_model->create($credentials["username"], "Failed to link ".$credentials["username"]." to ".$student_number, 'crs_tag', 'error', $employee, $ip);
        }
    }
}

```

```

        $this->log_model->create($credentials["username"],
$crs_tag_result["errmsg"], 'crs_tag', 'error', $employee, $ip);

        $this->_send_crs_email($credentials["username"], $student_number,
TRUE, $crs_tag_result["errmsg"]);
        $data["crs_tag"]["success"] = FALSE;
        $data["crs_status"] = "Untagged";

    }

    $data["success"] = TRUE;
    $data["message"] = "Account successfully created.";
    $data["user"] = $ldap_result["details"];
    $data["user"]["firstname"] = $name["firstname"];
    $data["password"] = $credentials["password"];

    } else { # LDAP FAIL
        $this->_audit($credentials["username"]." : LDAP add failed");
        $this->log_model->create_activation($credentials["username"], "LDAP add
failed", 'error', $employee, $ip);
        $data["success"] = FALSE;
        $data["message"] = "LDAP insert fail.";
    }
    // } else { # MAIL FAIL
    // $this->_audit($credentials["username"]." : Mail user creation failed");
    // $this->log_model->create_activation($credentials["username"], "Mail
user creation failed", 'error', $employee, $ip);
    // $data["success"] = FALSE;
    // $data["message"] = "Mail server insert fail.";
    // }
    } else { # MYSQL FAIL
        $this->_audit($credentials["username"]." : Mysql insert failed");
        $this->log_model->create_activation($credentials["username"], "Mysql insert
failed", 'error', $employee, $ip);
        $data["success"] = FALSE;
        $data["message"] = "Mysql insert fail.";
    }
    return $data;
}

```

```

/*****
* Helper Functions:
* Generators
* Username, Email
*****/

```

```

private function _generate_credentials($name) {
    private function _generate_username($name["firstname"],
        $user["username"] = $this->_generate_username($name["firstname"],
$name["middlename"], $name["lastname"]);
        $user["email"] = $this->_generate_email($name["firstname"],
$name["middlename"], $name["lastname"]);
        $user["password"] = random_password();
    return $user;
}

```

```

// Returns available username for user
private function _generate_username($first, $middle, $last) {
    // Get middle initial or set to blank

```

```

$none = $this->config->item("none");
$middle = strtolower($middle) == $none ? "" : $middle[0];

if (strtolower($last) == $none) {
    // Special Case
    $username = $this->_format_username($first);
} else {
    // fmlastname
    $username = $this->_format_username($first[0].$middle.$last);
}

$username = $this->_remove_special_chars($username);

if ( $this->_credentials_exists($username) ) {
    for ($i = 1; ; $i++) {
        // fmlastnameX
        $usernameX = $username.$i;
        if ( !$this->_credentials_exists($usernameX) ) {
            $username = $usernameX;
            break;
        }
    }
}
return $username;
}

// # Returns available email address for user
private function _generate_email($first, $middle, $last) {
    $none = $this->config->item("none");
    $middle = strtolower($middle) == $none ? "" : ".$middle[0]; #
    Get .middleinitial or set to blank #
    $last = strtolower($last) == $none ? "" : ".$last; #
    Get .lastname or set to blank #

    $user_email = $this->_format_user_email($first.$last);
    first.last@up(d).edu.ph

    if ( $this->_credentials_exists($user_email) ) { #
        $user_email = $this->_format_user_email($first.$middle.$last);
        first.middleinitial.last@up(d).edu.ph
        if ( $this->_credentials_exists($user_email) ) {
            for ($i = 1; ; $i++) {
                $user_email = $this->_format_user_email($first.$last."_".$i); #
                first.last_X@up(d).edu.ph
                if ( !$this->_credentials_exists($user_email) ) {
                    break;
                }
            }
        }
    }
    return $user_email;
}

/*****
* Helper Functions:
* Checking Existance
* Checking Availbility
* Send CRS email
*****/

```

```

private function _credentials_exists($username_or_email) {
    $test = $this->sam_user_model->get_by_email_or_username($username_or_email);
    if (empty($test)) {
        $test = $this->sam_ldap_model->search($username_or_email);
        if (empty($test)) { return FALSE; }
        return TRUE;
    } else {
        return TRUE;
    }
}

public function _send_crs_email($username,$student_number,$error,$errmsg) {

    $this->load->library('email');

    if($error) {

        $subj = "ERROR: CRS Tag Failed [ ".$username." ]";

        $message = 'Failed to tag CRS to DilNet account.'."\n";
        $message .= '-----'."\n";
        $message .= 'Username: '.$username."\n";
        $message .= 'Student Number: '.$student_number."\n";
        $message .= 'Error: '.$errmsg."\n";

    }

    else {

        $subj = "CRS Tag Success [ ".$username." ]";

        $message = 'CRS-DilNet Account tag successful.'."\n";
        $message .= '-----'."\n";
        $message .= 'Username: '.$username."\n";
        $message .= 'Student Number: '.$student_number."\n";

    }

    // $message .= 'Admin: '.$this->session->userdata("user")."\n";
    $message .= "\nDate tagged: ";
    $message .= date('Y-m-d H:i:s');

    $this->email->from('postmaster@upd.edu.ph', 'DilNet Accounts');
    $this->email->to('postmaster@upd.edu.ph');
    $this->email->subject($subj);
    $this->email->message($message);

    $this->email->send();
    // echo $this->email->print_debugger();
}

/*****
 * Helper Functions:
 * Formatting
 * Sanitation
 *****/

// Trim name and remove .
Private function _sanitize_name($firstname, $middlename, $lastname) {

```

```

    $firstname = preg_replace( '@\x{00F1}@u', "n", mb_strtolower($firstname, 'UTF-8'));
    $middlename = preg_replace( '@\x{00F1}@u', "n", mb_strtolower($middlename, 'UTF-8'));
    $lastname = preg_replace( '@\x{00F1}@u', "n", mb_strtolower($lastname, 'UTF-8'));

    $user["firstname"] = str_replace(".", "", trim($firstname));
    $user["middlename"] = str_replace(".", "", trim($middlename));
    $user["lastname"] = str_replace(".", "", trim($lastname));

    return $user;
}

private function _remove_special_chars($name) {
    $name = trim($name, "\t\n\r\0\x0B");
    $name = str_replace("-", "", $name);
    $name = str_replace(".", "", $name);
    $name = str_replace("'", "", $name);
    return $name;
}

// Converts spaces and -s
private function _format_user_email($user_email) {
    $user_email = str_replace(" ", "_", $user_email);
    $user_email = str_replace("-", "_", $user_email);
    $user_email = str_replace("'", "", $user_email);
    return strtolower($user_email);
}

// Remove special characters
private function _format_username($username) {
    $username = trim($username, "\t\n\r\0\x0B");
    $username = str_replace(" ", "", $username);
    $username = str_replace("-", "", $username);
    $username = str_replace(".", "", $username);
    return strtolower($username);
}

private function _prepare_student_data_mysql($student_number, $credentials, $name, $employee_username, $birthdate) {
    $student = array (
        "username" => $credentials["username"],
        "email" => $credentials["email"],

        "firstname" => $name["firstname"],
        "middlename" => $name["middlename"],
        "lastname" => $name["lastname"],

        "id_number" => $student_number,
        "studno" => $student_number,

        "hostname" => "imap4.upd.edu.ph",
        "designation" => "student",
        "activated_by" => $employee_username,
        "gender" => 0,
        "campus_id" => 1,
        "unit_id" => 0,
        "course_id" => 0,
        "birthday" => $birthdate,
    );
}

```

```

        "activated"    => TRUE,
    );
    return $student;
}

private function _prepare_student_data_ldap($student_number, $credentials,
$name, $unit) {
    $student = array (
        "username"    => $credentials["username"],
        "email"       => $credentials["email"],
        "password"    => $credentials["password"],

        "firstname"   => $name["firstname"],
        "lastname"    => $name["lastname"],

        "id_number"   => $student_number,

        "ou"          => $unit["main"],
        "department"  => $unit["sub"],

    );

    $ldap_data["objectclass"][0] = "top";
    $ldap_data["objectclass"][1] = "person";
    $ldap_data["objectclass"][2] = "inetOrgPerson";
    $ldap_data["objectclass"][3] = "qmailUser";

    $ldap_data["mailHost"] = 'imap4.upd.edu.ph';
    $ldap_data["mailAlternateAddress"][0] = $username."@upd.edu.ph";
    $ldap_data["mailMessageStore"] =
'upd.edu.ph/'.$ldap_data['ou'].'/'.$ldap_data['departmentnumber'].'/'.$username;

    return $student;
}
}

```

application/models/Admin_model.php

```

<?php if ( ! defined('BASEPATH')) exit('No direct script access allowed');

class Admin_model extends MY_Model {

    function __construct(){
        parent::__construct();
        $this->table = 'admin_users';
    }

    function get_by_employee_number($empno) {
        return $this->db->where('empno', $empno)                ->get($this->table)
                                ->row_array();
    }

    function get_username_from_employee_number($empno){
        $result = $this->db->select('username')

```



```

        ->where('empno',$empno)
        ->get($this->table)
        ->row_array();

    if ($result != NULL) {
        return $result["username"];
    }
    return NULL;
}

function get_admin_type($username) {
    $user = $this->db->where("username", $username)
        ->get("admin_users")->row_array();
    return $user["role"];
}

function admin_user_exists($username)
{
    $q = "SELECT * FROM admin_users WHERE `username` = ?";
    $r = $this->db->query($q, array($username));

    if($r->num_rows() > 0){
        return true;
    }
    else
    {
        return false;
    }
}
}

```

application/models/Changepass_model.php

```

<?php if ( ! defined('BASEPATH')) exit('No direct script access allowed');

class Changepass_model extends CI_Model {

    function __construct() {
        parent::__construct();
        $this->table = 'changepass';
    }

    // function get_all() {
    //     return $this->db->get($this->table)->result_array();
    // }

    function get_by_username($username, $latest = FALSE) {
        $result = $this->db->where('username', $username);
        if ($latest) {
            $result = $result->order_by("time", "desc")
                ->get($this->table)
                ->row_array();
        } else {
            $result = $result->get($this->table)->result_array();
        }
    }
}

```

```

    }
    return $result;
}

function create($username, $ip, $changed_by, $notes) {
    $log = array( "username" => $username,
                  "ip"       => $ip,
                  "time"     => date('Y-m-d H:i:s'),
                  "changed_by" => $changed_by,
                  "notes"    => $notes);
    return $this->db->insert($this->table, $log);
}

function reset_recovery_count($username) {
    $this->db->set("recovery_count", 0)
        ->where("username", $username)
        ->update("users");
}
}

```

application/models/Crs_model.php

```

<?php if ( ! defined('BASEPATH')) exit('No direct script access allowed');

class Crs_model extends MY_Model {

    /**
     * Index Page for this controller.
     *
     * Maps to the following URL
     *      http://example.com/index.php/welcome
     *  - or -
     *      http://example.com/index.php/welcome/index
     *  - or -
     * Since this controller is set as the default controller in
     * config/routes.php, it's displayed at http://example.com/
     *
     * So any other public methods not prefixed with an underscore will
     * map to /index.php/welcome/<method_name>
     * @see http://codeigniter.com/user_guide/general/urls.html
     */

    function __construct(){
        parent::__construct();
        // Load the rest client spark
        $this->load->spark('curl/1.2.1');
        $this->load->spark('restclient/2.1.0');

        // Load the library
        $this->load->library('rest');

        // Run some setup
        $config = array(
            'server' => 'http://10.16.9.122/webservices/upcc'
        );
    }
}

```

```

        $this->rest->initialize($config);

        $this->key = 'pYuOdsF5cqOstdQR';
    }

    /* CRS API */

    function _getstudentinfo($studentno){

        $params = array(
            'studentno' => $studentno,
            'key' => $this->key,
        );

        $result = $this->rest->post('getstudentinfo', $params, 'json');

        $result = json_decode(json_encode($result), TRUE);

        if(isset($result["errcode"]) && $result["errcode"] == 40) {
            $this->_enable_epp();
            $result = $this->rest->post('getstudentinfo', $params, 'json');
            $result = json_decode(json_encode($result), TRUE);
            $this->_revert_to_upd();
        }

        return $result;
    }

    function _getstudentnobyusername($username){

        $params = array(
            'username' => $username,
            'key' => $this->key,
        );

        $result = $this->rest->post('getstudentnobyusername',
$params, 'json');

        $result = json_decode(json_encode($result), TRUE);

        if(isset($result["errcode"]) && $result["errcode"] == 40) {
            $this->_enable_epp();
            $result = $this->rest->post('getstudentnobyusername',
$params, 'json');

            $result = json_decode(json_encode($result), TRUE);
            $this->_revert_to_upd();
        }

        return $result;
    }

    function _isenrolledbystudentno($studentno,$ay,$termno){
        $params = array(
            'studentno' => $studentno,
            'ay' => $ay,
            'termno' => $termno,
            'key' => $this->key,
        );
    }

```

```

        // print_r($params);
        $result = $this->rest->post('isenrolledbystudentno', $params,
'json');

        $result = json_decode(json_encode($result), TRUE);

        if(isset($result["errmsg"]) && $result["errmsg"] == "Student does
not exist.") {
            $this->_enable_epp();
            $result = $this->rest->post('isenrolledbystudentno', $params,
'json');

            $result = json_decode(json_encode($result), TRUE);
            $this->_revert_to_upd();

        }

        return $result;
    }

    function _isenrolledbybarcode($f5barcode){
        $params =
            array(
                'f5barcode' => $f5barcode,
                'key' => $this->key,
            );

        $result = $this->rest->post('isenrolledbybarcode', $params, 'json');

        $result = json_decode(json_encode($result), TRUE);

        if(isset($result["errcode"]) && $result["errcode"] == 30) {
            $this->_enable_epp();
            $result = $this->rest->post('isenrolledbybarcode', $params,
'json');

            $result = json_decode(json_encode($result), TRUE);
            $this->_revert_to_upd();

        }

        return $result;
    }

    function _assignusernamestudent($username,$studentno){
        $studentno = $this->_stripdash($studentno);

        $result = $this->rest->post('assignusernamestudent',
            array(
                'username' => $username,
                'studentno' => $studentno,
                'key' => $this->key,
            ), 'json');

        $result = json_decode(json_encode($result), TRUE);

        return $result;
    }

    function _revert_to_upd() {

```

```

$config = array('server' => 'http://10.16.9.122/webservices/upcc' );
$this->rest->initialize($config);
}
function _enable_epp() {
$config = array('server' => 'http://10.16.9.122/webservices/upcc/epp' );
$this->rest->initialize($config);
}
}

```

application/models/Ldap_model.php

```

<?php if ( ! defined('BASEPATH')) exit('No direct script access allowed');

class Ldap_model extends MY_Model{

function __construct() {
    parent::__construct();

    log_message('debug', 'Auth_Ldap initialization commencing...');

    $this->_init();
}

private function _init() {
    #TODO Removed show_error for the meantime, should be added to logs

    // Verify that the LDAP extension has been loaded/built-in
    // No sense continuing if we can't
    if (! function_exists('ldap_connect')) {
        show_error('LDAP functionality not present. Either load the module
ldap php module or use a php with ldap support compiled in.');
```

log_message('error', 'LDAP functionality not present in php.');

```

    }

    // $this->host = 'ldaps://ldap.upd.edu.ph/';
    $this->config->load('config_ldap',TRUE);
    $this->host = $this->config->item('ldap_host','config_ldap');
    $this->basedn = $this->config->item('ldap_basedn','config_ldap');
    $this->ldap_port = $this->config->item('ldap_port','config_ldap');
    $this->login_attribute = 'uid';
    $this->login_attribute = 'application/logs/audit.log';
    // $this->auditlog = 'application/logs/audit.log';
    $this->auditlog = '/tmp/audit.log';
}

public function _ldapconnect(){
    $this->ldapconn = ldap_connect($this->host, $this->ldap_port);

    // At this point, $this->ldapconn should be set. If not... DOOM!
    if(! $this->ldapconn) {
        log_message('error', "Couldn't connect to any LDAP servers.
Bailing...");
        show_error('Error connecting to your LDAP server(s). Please check the
connection and try again.');
```

}

```

        ldap_set_option($this->ldapconn, LDAP_OPT_PROTOCOL_VERSION, 3);

        return $this->ldapconn;
    }

    public function _authenticate($username, $password){

        $this->ldapconn = $this->_ldapconnect();
        $this->bind = ldap_bind($this->ldapconn);

        if(!$this->bind){
            log_message('error', 'Unable to perform anonymous/proxy bind');
            show_error('Unable to bind for user id lookup');
            $errmsg = ldap_error($this->ldapconn);
            return $errmsg;
        }

        log_message('debug', 'Successfully bound to directory. Performing dn
lookup for '.$username);
        $filter = '(uid='.$username.')';
        $search = ldap_search($this->ldapconn, $this->basedn, $filter);

        $entries = ldap_get_entries($this->ldapconn, $search);
        /*
        print "<pre>";
        print_r($entries);
        print "</pre>";
        /**/

        if(isset($entries[0]['dn'])){
            $binddn = $entries[0]['dn'];
        }

        //Bind as the user
        //print "<pre>";
        //print_r($binddn);
        //print "</pre>";
        if(isset($binddn)){
            $bind = @ldap_bind($this->ldapconn, $binddn, $password);
        }

        if(!isset($bind) || !$bind) {
            $this->_audit("Failed login attempt: ".$username." from
".$_SERVER['REMOTE_ADDR']);
            ldap_close($this->ldapconn);
            $errmsg = "Incorrect username or password";
            return $errmsg;
        }

        else {
            $cn = $entries[0]['cn'][0];
            $dn = stripslashes($entries[0]['dn']);
            $id = $entries[0][$this->login_attribute][0];
            $ldapinfo = $entries[0];

            $this->session->set_userdata('ldap_name', $cn);

            $this->ldap_model->_audit("Successful login: ".$cn."(".$username.")
from ".$this->input->ip_address());

```

```

        ldap_close($this->ldapconn);
        return "Success";
    }
}

public function search_mail_addresses($query,$attr=null){
    $this->ldapconn = $this->_ldapconnect();
    $this->bind = ldap_bind($this->ldapconn);

    if(!$this->bind){
        log_message('error', 'Unable to bind.');
```

show_error('Unable to bind for mail addresses lookup.');

```
    }

    log_message('debug', 'Successfully bound to directory. Performing lookup
for '.$query);

    if(strpos($query,"@upd.edu.ph")){ // might be a primary email
        $query_mail = $query;
    } else {
        if (filter_var($query, FILTER_VALIDATE_EMAIL)) {
            $query_mail = $query;
        } else {
            $query_mail = $query."@upd.edu.ph";
        }
    }

    $filter =
'(|(mail='.$query_mail.')(mailAlternateAddress='.$query_mail.'))';
    if ($attr == NULL) {
        $search = ldap_search($this->ldapconn, $this->basedn, $filter);
    } else {
        $search = ldap_search($this->ldapconn, $this->basedn, $filter, $attr);
    }
    $entries = ldap_get_entries($this->ldapconn, $search);
    if( !empty($entries) && $entries['count'] > 0 ){

        ldap_close($this->ldapconn);
        $entries = $this->rCountRemover($entries);
        return $entries;
    }

    else {
        ldap_close($this->ldapconn);
        return FALSE;
    }
}

public function is_username_available($query,$uid,$attr=null){
    $this->ldapconn = $this->_ldapconnect();
    $this->bind = ldap_bind($this->ldapconn);

    if(!$this->bind){
        log_message('error', 'Unable to bind.');
```

show_error('Unable to bind for mail addresses lookup.');

```
    }

    log_message('debug', 'Successfully bound to directory. Performing lookup
for '.$query);

```

```

$filter = '(! (uid= '.$query.'))';
if ($attr == NULL) {
    $search = ldap_search($this->ldapconn, $this->basedn, $filter);
} else {
    $search = ldap_search($this->ldapconn, $this->basedn, $filter, $attr);
}

$entries = ldap_get_entries($this->ldapconn, $search);
if( !empty($entries) && $entries['count'] > 0 ){

    ldap_close($this->ldapconn);
    $entries = $this->remove($entries);
    if($entries[0]["uid"][0]==$uid){
        return TRUE; // available because same owner
    } else {
        return FALSE; // owned by another user
    }

} else {
    ldap_close($this->ldapconn);
    return TRUE; // available
}
}

public function is_email_address_available($query,$uid,$attr=null){
    $this->ldapconn = $this->ldapconnect();
    $this->bind = ldap_bind($this->ldapconn);

    if(!$this->bind){
        log_message('error', 'Unable to bind.');
```

show_error('Unable to bind for mail addresses lookup.');

```

    }

    log_message('debug', 'Successfully bound to directory. Performing lookup
for '.$query);

    if(strpos($query,"@upd.edu.ph")){ // might be a primary email
        $query_mail = $query;
    } else {
        if (filter_var($query, FILTER_VALIDATE_EMAIL)) {
            $query_mail = $query;
        } else {
            $query_mail = $query."@upd.edu.ph";
        }
    }

    $filter =
'(! (mail= '.$query_mail.' ) (mailAlternateAddress= '.$query_mail.'))';
    if ($attr == NULL) {
        $search = ldap_search($this->ldapconn, $this->basedn, $filter);
    } else {
        $search = ldap_search($this->ldapconn, $this->basedn, $filter, $attr);
    }

    $entries = ldap_get_entries($this->ldapconn, $search);
    if( !empty($entries) && $entries['count'] > 0 ){

        ldap_close($this->ldapconn);

```



```

        $entries = $this->rCountRemover($entries);

        if($entries[0]["uid"][0]==$uid){
            return TRUE; // available because same owner
        } else {
            return FALSE; // owned by another user
        }
    } else {
        ldap_close($this->ldapconn);
        return TRUE; // available
    }
}

public function search_substring($id,$attr=null){

    $this->ldapconn = $this->_ldapconnect();
    $this->bind = ldap_bind($this->ldapconn);

    if(!$this->bind){
        log_message('error', 'Unable to perform anonymous/proxy bind');
        show_error('Unable to bind for user id lookup');
    }

    log_message('debug', 'Successfully bound to directory. Performing dn
lookup for '.$id);

    // Search exact phrase
    $filter = '(|(uid='.$id.')(cn='.$id.'))';
    if ($attr == NULL) {
        $search = ldap_search($this->ldapconn, $this->basedn, $filter);
    } else {
        $search = ldap_search($this->ldapconn, $this->basedn, $filter, $attr);
    }
    $entries = ldap_get_entries($this->ldapconn, $search);

    if( !empty($entries) && $entries['count'] > 0 ){

        ldap_close($this->ldapconn);
        $entries = $this->rCountRemover($entries);
        return $entries;
    }

    // If no results, search using wildcard
    // TODO: Reevaluate nested if-else statements
    else {
        $filter =
        '(|(uid='.$id.'*)(cn='.$id.'*)(mail='.$id.'@up.edu.ph)(mail='.$id.'@upd.edu.ph)(
mailAlternateAddress='.$id.'@up.edu.ph)(mailAlternateAddress='.$id.'@upd.edu.ph))'
        ;
        if ($attr == NULL) { $search = ldap_search($this->ldapconn, $this-
>basedn, $filter); }
        else { $search = ldap_search($this->ldapconn, $this->basedn, $filter,
$attr); }
        $entries = ldap_get_entries($this->ldapconn, $search);
    }
}

```

```

        if( !empty($entries) && $entries['count'] > 0 ){

            ldap_close($this->ldapconn);
            $entries = $this->rCountRemover($entries);
            return $entries;
        }

        else {
            ldap_close($this->ldapconn);
            return FALSE;
        }
    }
}

public function search_substring_number($id,$attr=null){

    $this->ldapconn = $this->_ldapconnect();
    $this->bind = ldap_bind($this->ldapconn);

    if(!$this->bind){
        log_message('error', 'Unable to perform anonymous/proxy bind');
        show_error('Unable to bind for user id lookup');
    }

    log_message('debug', 'Successfully bound to directory. Performing dn
lookup for '.$id);
    $filter = '(|(studentnumber='.$id.')(employeenumber='.$id.))';
    if ($attr == NULL) { $search = ldap_search($this->ldapconn, $this->basedn,
$filter); }
    else { $search = ldap_search($this->ldapconn, $this->basedn, $filter,
$attr); }
    $entries = ldap_get_entries($this->ldapconn, $search);

    if( !empty($entries) && $entries['count'] > 0 ){
        ldap_close($this->ldapconn);
        $entries = $this->rCountRemover($entries);
        return $entries;
    }

    else {
        ldap_close($this->ldapconn);
        return FALSE;
    }
}

public function get_ldap_info($id) {
    $this->ldapconn = $this->_ldapconnect();

    $binddn = $this->config->item('ldap_root','config_ldap');
    $bindp = $this->config->item('ldap_root_pass','config_ldap');

    $bind = ldap_bind($this->ldapconn, $binddn, $bindp);
    if(!$bind){
        log_message('error', 'Unable to perform anonymous/proxy bind');
        show_error('Unable to bind for user id lookup');
        echo "bind fail";
        ldap_unbind($this->ldapconn);
    }
}

```

```

    } else {
        $filter = '(|(uid= '.$id.'))';
        $attributes =
array('uid','cn','ou','mail','mailalternateAddress','studentnumber','employeeNumbe
r','departmentNumber','title','employeeType','mailquotasize');
        // $attributes = "";

        $search = ldap_search($this->ldapconn, $this->basedn,
$filter,$attributes);
        $entries = $this->rCountRemover(ldap_get_entries($this->ldapconn,
$search));

        /*
        print "<pre>";
        print_r($entries);
        print "</pre>";
        /**/

        if(count($entries)>0){
            return $entries;
        } else {
            ldap_close($this->ldapconn);
            return FALSE;
        }
    }
}

public function get_user_account($id){

    $this->ldapconn = $this->_ldapconnect();
    $this->bind = ldap_bind($this->ldapconn);

    if(!$this->bind){
        log_message('error', 'Unable to perform anonymous/proxy bind');
        show_error('Unable to bind for user id lookup');
    }

    log_message('debug', 'Successfully bound to directory. Performing dn
lookup for '.$id);
    $filter = '(|(uid= '.$id.'))';

    #Suppress LDAP error messages
    $search = @ldap_read($this->ldapconn, "uid=".$id.", ".$this->basedn,
$filter);
    if(!$search) { return FALSE; }

    $entries = @ldap_get_entries($this->ldapconn, $search);

    if( !empty($entries) && $entries['count'] > 0 ){
        ldap_close($this->ldapconn);
        $entries = $this->rCountRemover($entries);
        return $entries[0];
    }

    else {
        ldap_close($this->ldapconn);
        return FALSE;
    }
}
}

```

```

public function rename_ldap_user($old_uid,$new_uid){
    $this->ldapconn = $this->_ldapconnect();

    $binddn = $this->config->item('ldap_root','config_ldap');
    $bindp = $this->config->item('ldap_root_pass','config_ldap');

    $bind = ldap_bind($this->ldapconn, $binddn, $bindp);

    if(!$bind){
        log_message('error', 'Unable to bind.');
```

rename

```

        show_error('Unable to bind for mail addresses lookup.');
```

deletion on

```

    } else {
        log_message('debug', 'Successfully bound to directory. Performing
        $.old_uid.' -> '$new_uid.');
```

```

        $full_old_dn = "uid=$old_uid,".$this->basedn;
        $new_rdn = "uid=".$new_uid;

        return ldap_rename($this->ldapconn, $full_old_dn, $new_rdn, $this->
        basedn, TRUE);
    }

}

public function delete_ldap_user($delete_uid){
    $this->ldapconn = $this->_ldapconnect();

    $binddn = $this->config->item('ldap_root','config_ldap');
    $bindp = $this->config->item('ldap_root_pass','config_ldap');

    $bind = ldap_bind($this->ldapconn, $binddn, $bindp);

    if(!$bind){
        log_message('error', 'Unable to bind.');
```

```

        show_error('Unable to bind for mail addresses lookup.');
```

```

    } else {
        log_message('debug', 'Successfully bound to directory. Performing
        $.delete_uid.');
```

```

        $delete_rdn = "uid=$delete_uid,".$this->basedn;
        return ldap_delete($this->ldapconn, $delete_rdn);
    }

}

public function add_ldapentry($username, $data)
{
    // $this->ldapconn = $this->_ldapconnect();
    // $this->bind = ldap_bind($this->ldapconn);

    $this->ldapconn = $this->_ldapconnect();

    $binddn = $this->config->item('ldap_root','config_ldap');
    $bindp = $this->config->item('ldap_root_pass','config_ldap');

    $bind = ldap_bind($this->ldapconn, $binddn, $bindp);

    if(!$bind){
        log_message('error', 'Unable to perform anonymous/proxy bind');
```

```

        show_error('Unable to bind for user id lookup');
```

```

    } else {

```

```

        $dn = "uid=".$username.",o=University of the Philippines,c=PH";
        // print "<pre>";
        // print_r($data);
        // print_r($dn);
        // print "</pre>";
        $r = ldap_add($this->ldapconn, $dn, $data);
        ldap_close($this->ldapconn);
        return $r;
    }
}

# WARNING: modify with caution
# being used in both updating ldap entry and resetting password
public function set_ldapentry($id, $entrydn, $entry)
{
    $this->ldapconn = $this->_ldapconnect();

    $binddn = $this->config->item('ldap_root','config_ldap');
    $bindp = $this->config->item('ldap_root_pass','config_ldap');

    $bind = ldap_bind($this->ldapconn, $binddn, $bindp);

    if(ldap_mod_replace($this->ldapconn, $entrydn, $entry)){
        ldap_unbind($this->ldapconn);
        return true;
    }

    else{ echo ldap_error($this->ldapconn); return false; }

}

public function get_by_id_number($id) {
    $this->ldapconn = $this->_ldapconnect();

    $binddn = $this->config->item('ldap_root','config_ldap');
    $bindp = $this->config->item('ldap_root_pass','config_ldap');

    $bind = ldap_bind($this->ldapconn, $binddn, $bindp);
    if(!$bind){
        log_message('error', 'Unable to perform anonymous/proxy bind');
        show_error('Unable to bind for user id lookup');
        echo "bind fail";
        ldap_unbind($this->ldapconn);
    } else {
        $filter = '(!(|(studentNumber='.$id.')(employeeNumber='.$id.)))';
        $attributes=array('uid','cn','ou');
        $search = ldap_search($this->ldapconn, $this->basedn, $filter,
        $attributes);
        $entries = ldap_get_entries($this->ldapconn, $search);

        ldap_unbind($this->ldapconn);
        return $entries;
    }
}

# search for exact match of uid | studentnumber | employeenumber |
mailaddresses

```

```

# returns boolean
public function search_exact($id)
{
    $this->ldapconn = $this->_ldapconnect();
    $this->bind = ldap_bind($this->ldapconn);

    if(!$this->bind){
        log_message('error', 'Unable to perform anonymous/proxy bind');
        show_error('Unable to bind for user id lookup');
    }

    log_message('debug', 'Successfully bound to directory. Performing dn
lookup for '.$id);
    $filter =
'|(|(uid='.$id.')(studentNumber='.$id.')(employeeNumber='.$id.')(mail='.$id.'@up.ed
u.ph)(mail='.$id.'@upd.edu.ph)(mailAlternateAddress='.$id.'@up.edu.ph)(mailAlterna
teAddress='.$id.'@upd.edu.ph))';
    $search = ldap_search($this->ldapconn, $this->basedn, $filter,
array('uid'));
    $entries = ldap_get_entries($this->ldapconn, $search);

    if( !empty($entries) && $entries['count'] > 0 ){
        ldap_unbind($this->ldapconn);
        return TRUE;
    }
    else {
        ldap_unbind($this->ldapconn);
        return FALSE;
    }
}

#Remove "count key" on LDAP entries
function rCountRemover($arr) {
    foreach($arr as $key=>$val) {
        # (int)0 == "count", so we need to use ===
        if($key === "count")
            unset($arr[$key]);
        elseif(is_array($val))
            $arr[$key] = $this->rCountRemover($arr[$key]);
    }
    return $arr;
}

function disable_ldap_user($disable_uid){
    $ldap_diff['dialupAccess'] = "FALSE";
    $update_ldap = $this->ldap_model->set_ldapentry($disable_uid,
"uid=$disable_uid,o=University of the Philippines,c=PH", $ldap_diff);
    return $update_ldap;
}

function enable_ldap_user($enable_uid){
    $ldap_diff['dialupAccess'] = "TRUE";
    $update_ldap = $this->ldap_model->set_ldapentry($enable_uid,
"uid=$enable_uid,o=University of the Philippines,c=PH", $ldap_diff);
    return $update_ldap;
}
}

```

application/models/Log_model.php

```
<?php if ( ! defined('BASEPATH')) exit('No direct script access allowed');

class Log_model extends MY_Model {

    function __construct(){
        parent::__construct();
        $this->logs_table = 'logs';
        $this->changepass_table = 'changepass';
    }

    function changepass_create($username, $ip, $changed_by, $notes) {
        $log = array( "username" => $username,
            "ip" => $ip,
            "time" => date('Y-m-d H:i:s'),
            "changed_by" => $changed_by,
            "notes" => $notes);
        return $this->db->insert($this->changepass_table, $log);
    }

    function create($account, $details, $service, $changed_by=null, $ip=null,
$type='debug') {
        $log = array( "account" => $account,
            "time" => date('Y-m-d H:i:s'),
            "detail" => $details,
            "service" => $service,
            "log_level" => $type,
            "ip" => $ip,
            "changed_by"=> $changed_by);
        return $this->db->insert($this->logs_table, $log);
    }

    function create_activation($account, $details, $type='debug',
$changed_by=null, $ip=null) {
        $log = array( "account" => $account,
            "time" => date('Y-m-d H:i:s'),
            "detail" => $details,
            "service" => "activation",
            "log_level" => $type,
            "ip" => $ip,
            "changed_by"=> $changed_by);
        return $this->db->insert($this->logs_table, $log);
    }
}
```

application/models/Mail_tool_model.php

```
<?php if ( ! defined('BASEPATH')) exit('No direct script access allowed');

class Mail_tool_model extends MY_Model {
```

```

function __construct(){
    parent::__construct();
    log_message('debug', 'SSH CMDs initialization commencing...');
    $this->db = $this->load->database('mailban', TRUE);
    $this->emails_table = 'emails';
    $this->hosts_table = 'hosts';
    $this->logs_table = 'logs';
}

function search($query_str) {
    // $cmd = "ssh root@10.16.3.12 \"grep $query_str
/var/mailcleaner/log/exim_stage1/mainlog
/var/mailcleaner/log/exim_stage1/mainlog.0\"";
    $cmd = "ssh root@10.16.3.12 /bin/grep -i \\\"$query_str\\\"
/var/mailcleaner/log/exim_stage1/mainlog";
    // $this->session->set_flashdata('error', $cmd);
    exec($cmd, $results);

    // $cmd = "ssh root@10.16.3.11 grep \\\"$query_str\\\"
/var/mailcleaner/log/exim_stage1/mainlog";
    // exec($cmd, $results);

    return $results;
}

function view_log_entry($ID){
    // TODO: validate mail ID
    // $cmd = "ssh root@10.16.3.12 \"grep $ID
/var/mailcleaner/log/exim_stage1/mainlog
/var/mailcleaner/log/exim_stage1/mainlog.0\"";
    $cmd = "ssh root@10.16.3.12 /bin/grep -i $ID
/var/mailcleaner/log/exim_stage1/mainlog";
    exec($cmd, $results);

    // $cmd = "ssh root@10.16.3.11 grep $ID
/var/mailcleaner/log/exim_stage1/mainlog";
    // exec($cmd, $results);
    return $results;
}

function get_block_email_info($email){
    $this->db->where('email_address', $email);
    $sql_res = $this->db->get($this->emails_table)->row_array();

    return $sql_res;
}

##### DISABLE/ENABLE ACCOUNT #####
function enable_email($email, $reason, $admin){
    $this->db->where('email_address', $email);
    $this->db->where('email_address', $email);
    $this->db->update($this->emails_table, array('disabled' => 0,
    $this->db->update($this->emails_table, array('date_unbanned' => date('Y-m-d H:i:s'),
'reason_unbanned' => $reason, 'date_unbanned' => date('Y-m-d H:i:s'),
'tech_unbanned' => $admin));

    $cmd = "ssh root@10.16.3.12 \"echo $email > /root/unphisher.inc;
/root/./removePhisher.sh\"";
    // uncomment below to execute

```



```

    // exec($cmd, $results);

    $cmd = "ssh root@10.16.3.11 \"echo $email > /root/unphisher.inc;
/root/./removePhisher.sh\"";
    // uncomment below to execute
    // exec($cmd, $results);

    return TRUE;
}

function disable_email($email, $reason, $admin) {
    $temp = $this->get_block_email_info($email);
    if(!empty($temp)){
        $this->db->where('email_address', $email);
        $this->db->update($this->emails_table, array('disabled' => 1,
'reason_banned' => $reason, 'date_banned' => date('Y-m-d H:i:s'), 'tech_banner' =>
$admin));
    } else {
        $this->db->set('email_address', $email);
        $this->db->insert($this->emails_table, array('disabled' => 1,
'reason_banned' => $reason, 'date_banned' => date('Y-m-d H:i:s'), 'tech_banner' =>
$admin));
    }

    $cmd = "ssh root@10.16.3.12 \"echo $email > /root/phisher.inc;
/root/./appendPhisher.sh\"";
    // uncomment below to execute
    // exec($cmd, $results);

    $cmd = "ssh root@10.16.3.11 \"echo $email > /root/phisher.inc;
/root/./appendPhisher.sh\"";
    // uncomment below to execute
    // exec($cmd, $results);

    return TRUE;
}

function remove_emails($email) {
    $results_array = array();
    $cmd = "ssh root@10.16.3.12 /root/./remFromEximStage1Spool.sh $email";
    // uncomment below to execute on live
    // exec($cmd, $results);

    $cmd = "ssh root@10.16.3.11 /root/./remFromEximStage1Spool.sh $email";
    // uncomment below to execute on live
    // exec($cmd, $results);

    return $results;
}

function is_email_disabled($email){
    $email_info = $this->get_block_email_info($email);
    if(!empty($email_info['disabled']) && $email_info['disabled'] == TRUE){
        return TRUE;
    } else {
        return FALSE;
    }
}

```

```

function get_block_host_info($ip){

    $this->db->where('ip_address', $ip);
    $sql_res = $this->db->get($this->hosts_table)->row_array();

    return $sql_res;
}

function enable_ip($ip, $reason, $admin){
    $this->db->where('ip_address', $ip);
    $this->db->update($this->hosts_table, array('disabled' => 0, 'reason_unbanned'
=> $reason, 'date_unbanned' => date('Y-m-d H:i:s'), 'tech_unbanner' => $admin));

    $cmd = "ssh root@10.16.3.12 \"echo $ip > /root/unphisherhost.inc;
/root/./removePhisherHost.sh\"";
    // uncomment below to execute
    // exec($cmd, $results);

    $cmd = "ssh root@10.16.3.11 \"echo $ip > /root/unphisherhost.inc;
/root/./removePhisherHost.sh\"";
    // uncomment below to execute
    // exec($cmd, $results);

    return TRUE;
}

function disable_ip($ip, $reason, $admin) {
    $temp = $this->get_block_host_info($ip);
    if(!empty($temp)){
        $this->db->where('ip_address', $ip);
        $this->db->update($this->hosts_table, array('disabled' => 1, 'reason_banned'
=> $reason, 'date_banned' => date('Y-m-d H:i:s'), 'tech_banner' => $admin));
    } else {
        $this->db->set('ip_address', $ip);
        $this->db->insert($this->hosts_table, array('disabled' => 1, 'reason_banned'
=> $reason, 'date_banned' => date('Y-m-d H:i:s'), 'tech_banner' => $admin));
    }

    $cmd = "ssh root@10.16.3.12 \"echo $ip > /root/phisherhost.inc;
/root/./appendPhisherHost.sh\"";
    // uncomment below to execute
    // exec($cmd, $results);

    $cmd = "ssh root@10.16.3.11 \"echo $ip > /root/phisherhost.inc;
/root/./appendPhisherHost.sh\"";
    // uncomment below to execute
    // exec($cmd, $results);

    return TRUE;
}

function is_host_disabled($ip){
    $host_info = $this->get_block_host_info($ip);
    if(!empty($host_info['disabled']) && $host_info['disabled'] == TRUE){
        return TRUE;
    } else {
        return FALSE;
    }
}

```

```

    }
}

function search_email($string) {
    $sql = "select * from ".$this->emails_table." where email_address like
'%" . $string . "%' order by 'disabled' desc";
    return $this->db->query($sql)->result_array();
}

// hosts blocking functions
function search_host($string) {
    $sql = "select * from ".$this->hosts_table." where ip_address like
'%" . $string . "%' order by 'disabled' desc";
    return $this->db->query($sql)->result_array();
}
}

```

application/models/Mysql_model.php

```

<?php if ( ! defined('BASEPATH')) exit('No direct script access allowed');

class Mysql_model extends MY_Model {

    function __construct(){
        parent::__construct();
        $this->webmaildb = 'default';

        $this->accounts_table = 'users';
        $this->changepass_table = 'changepass';
        $this->logs_table = 'logs';
        $this->pending_recovery_table = 'pending_recovery_email';
    }

    /*** USER FUNCTIONS ***/

    function get_user_account($username){

        $this->db->where('username', $username);
        $sql_res = $this->db->get($this->accounts_table)->row_array();

        return $sql_res;
    }

    /*** GET TABLE FUNCTIONS ***/
    function get_designation_list(){
        $sql = "SELECT DISTINCT designation from ".$this->accounts_table;
        $sql_res = $this->db->query($sql)->result_array();

        return $sql_res;
    }

    function get_unit_list(){
        $sql = "SELECT DISTINCT acronym, unitname as name from unit";
        $sql_res = $this->db->query($sql)->result_array();
    }
}

```

```

    return $sql_res;
}

function get_division_list(){
    $sql = "SELECT DISTINCT acronym, unitname from ldap_division";
    $sql_res = $this->db->query($sql)->result_array();

    return $sql_res;
}

/** "Get by" functions: Used by input field checkers */

function get_by_search_type($string, $type) {

    switch($type){
        case "STUD":
            $sql_res = $this->db->where('id_number', $string)
                ->or_where("studno", $string)
                ->get($this->accounts_table)
                ->row_array();

            break;

        case "EMP":
            $sql_res = $this->db->where('id_number', $string)
                ->or_where("empno", $string)
                ->get($this->accounts_table)
                ->row_array();

            break;

        case "UN":
            $sql_res = $this->db->where('username', $string)
                ->get($this->accounts_table)
                ->row_array();

            break;

        default:
            $sql_res = NULL;
    }

    return $sql_res;
}

/** GET STATISTICS */
function get_new_accounts_by_year($year){
    // NOTE: Also consider service type "activation" for earlier years
    // NOTE: Also consider service type "activation" for earlier years
    $sql = "SELECT COUNT(*) as total FROM logs WHERE YEAR(time)=". $year. " AND
service='activation' AND log_level='info'";
    // $sql = "SELECT COUNT(*) as total FROM logs WHERE YEAR(time)=". $year. " AND
service='signup' AND log_level='info'";
    $sql_res = $this->db->query($sql)->row_array();
    return $sql_res['total'];
}

function get_new_accounts_by_month($year,$month){
    // NOTE: Also consider service type "activation" for earlier years
    // NOTE: Also consider service type "activation" for earlier years
    $sql = "SELECT COUNT(*) as total FROM logs WHERE YEAR(time)=". $year. " AND
MONTH(time)=". $month. " AND service='activation' AND log_level='info'";
    // $sql = "SELECT COUNT(*) as total FROM logs WHERE YEAR(time)=". $year. " AND
MONTH(time)=". $month. " AND service='signup' AND log_level='info'";

```

```

    $sql_res = $this->db->query($sql)->row_array();
    return $sql_res['total'];
}

function get_reset_passwords_by_staff_month($year,$month){
    $sql = "SELECT COUNT(*) as total FROM ".$this->changepass_table." WHERE
YEAR(time)=".$year." AND MONTH(time)=".$month." AND changed_by != username AND
changed_by != 'lostpass feature'";
    $sql_res = $this->db->query($sql)->row_array();
    return $sql_res['total'];
}

function get_lostpass_by_month($year,$month){
    $sql = "SELECT COUNT(*) as total FROM ".$this->changepass_table." WHERE
YEAR(time)=".$year." AND MONTH(time)=".$month." AND changed_by = 'lostpass
feature'";
    $sql_res = $this->db->query($sql)->row_array();
    return $sql_res['total'];
}

function get_tech_staff_stats($year){
    $sql = "SELECT changed_by,COUNT(*) as total FROM ".$this->logs_table." WHERE
YEAR(time)=".$year." AND service='activation' AND log_level='info' GROUP BY
changed_by";
    $sql_res = $this->db->query($sql)->result_array();
    return $sql_res;
}

function get_acctype_stats($year){
    $sql = "SELECT type,COUNT(*) as total FROM ".$this->accounts_table." WHERE
YEAR(date_activated)=".$year." GROUP BY type";
    $sql_res = $this->db->query($sql)->result_array();
    return $sql_res;
}

/** SEARCH FUNCTIONS */

function search_substring($string) {
    // $sql = "select * from users where firstname like '%" . $string . "%' or
    lastname like '%" . $string . "%' or middlename like '%" . $string . "%' or username like
    '%" . $string . "%'";

    //Search partial words -- workaround for names with underscores
    $stringArray = explode(" ", $string);
    $match_str = "MATCH (firstname,middlename,lastname,username,email) AGAINST
('";
    $i=0;
    $j=count($stringArray);
    foreach ($stringArray as $word) {
        if ($i == 0) {
            $match_str .= " +". $word ."* ";
        } else if ($i == $j-1) {
            $match_str .= " +". $word;
        } else {
            $match_str .= " +*". $word ."* ";
        }
        $i++;
    }
}

```

```

$match_str .= "' IN BOOLEAN MODE)";

$sql = "SELECT
username,firstname,middlename,lastname,email,id_number, ".$match_str." AS score
FROM ".$this->accounts_table." where ".$match_str." ORDER BY score DESC;";
return $this->db->query($sql)->result_array();
}

function search_substring_number($string,$exact=false) {
    if($exact){
        $sql = "select * from ".$this->accounts_table." where id_number like
'".$string."'";
    } else {
        $sql = "select * from ".$this->accounts_table." where id_number like
'%"$string.%'";
    }
    return $this->db->query($sql)->result_array();
}

function search_username($string) {
    $sql = "select * from ".$this->accounts_table." where username =
'".$string."'";
    return $this->db->query($sql)->result_array();
}

function search_email($string) {
    $sql = "select * from ".$this->accounts_table." where email = '".$string."' OR
email = '".$string."@upd.edu.ph'";
    return $this->db->query($sql)->result_array();
}

function search_pending_recovery_email($string) {
    $sql = "select * from ".$this->pending_recovery_table." where username =
'".$string."'";
    return $this->db->query($sql)->result_array();
}

/** LOG FUNCTIONS */
function get_accountlogs(){
    $sql = "SELECT * FROM logs WHERE (DATE(time) >= date_sub(CURDATE(), INTERVAL 3
MONTH)) ORDER BY time DESC;";
    $sql_res = $this->db->query($sql)->result_array();
    return $sql_res;
}

function get_changepass_logs(){
    $sql = "SELECT * FROM changepass WHERE (DATE(time) >= date_sub(CURDATE(),
INTERVAL 3 MONTH)) ORDER BY time DESC;";
    $sql_res = $this->db->query($sql)->result_array();
    return $sql_res;
}

function get_lostpass_logs(){
    $sql = "SELECT * FROM lostpass WHERE (DATE(date_added) >= date_sub(CURDATE(),
INTERVAL 3 MONTH)) ORDER BY date_added DESC;";
    $sql_res = $this->db->query($sql)->result_array();
    return $sql_res;
}

```

```

function get_accountlogs_by_user($username){
    $sql = "SELECT * FROM logs WHERE account = '". $username."' ORDER BY time
DESC;";
    $sql_res = $this->db->query($sql)->result_array();
    return $sql_res;
}

function get_changepass_logs_by_user($username){
    $sql = "SELECT * FROM changepass WHERE username = '". $username."' ORDER BY
time DESC;";
    $sql_res = $this->db->query($sql)->result_array();
    return $sql_res;
}

function get_lostpass_logs_by_user($username){
    $sql = "SELECT * FROM lostpass WHERE username = '". $username."' ORDER BY
date_added DESC;";
    $sql_res = $this->db->query($sql)->result_array();
    return $sql_res;
}

##### PASSWORD FUNCTION #####
function enable_force_changepass($username){
    $this->db->where('username', $username);
    $this->db->update($this->accounts_table, array('force_changepass' => 1));

    return TRUE;
}

function reset_recovery_count($username) {
    $this->db->set("recovery_count", 0)
        ->where("username", $username)
        ->update($this->accounts_table);
}

##### DISABLE ACCOUNT #####
function enable_account($username, $reason){
    $this->db->where('username', $username);
    $this->db->update($this->accounts_table, array('disabled' => 0,
'disabled_remarks' => $reason));

    return TRUE;
}

function disable_account($username, $reason) {
    $this->db->where('username', $username);
    $this->db->update($this->accounts_table, array('disabled' => 1,
'disabled_remarks' => $reason));

    return TRUE;
}

function update_mysql_entry ($data, $username) {
    $this->db->where('username', $username);
    if($this->db->update($this->accounts_table, $data)){ return true; }
    else { return false; }
}

function delete_mysql_entry ($where = array()) {

```

```

        $this->db->where($where);
        $res = $this->db->delete($this->accounts_table);
        if($res)
            return TRUE;
        else
            return FALSE;
    }

    ##### CREATE USER ACCOUNT #####
    function get_by_fullname($first, $middle, $last, $table) {
        $sql = "SELECT * FROM ".$table." WHERE firstname='".$first."' AND
        middlename LIKE '".$middle."' AND lastname='".$last.'";
        if ($table == "users") {
            $sql .= " AND activated=1";
        }
        return $this->db->query($sql)->row_array();
    }

    #Checks if the new account has the same name and same middle initial in the db
    function get_by_fullname_dup($first, $middle, $last) {
        $sql = "SELECT username FROM ".$this->accounts_table." WHERE
        firstname='".$first."' AND middlename LIKE '".$middle."' AND lastname='".$last.'"
        AND activated=1";
        return $this->db->query($sql)->result_array();
    }

    function get_by_id_number($id) {
        #TODO Merge table
        $table = $this->accounts_table;
        $this->db->or_where($id);
        if ($table == "users") {
            $this->db->where("activated", 1);
        }
        return $this->db->get($table)
            ->row_array();
    }

    function get_by_email_or_username($user) {
        return $this->db->where('email', $user)
            ->or_where('email', $user.'@upd.edu.ph')
            ->or_where('username', $user)
            ->get($this->accounts_table)
            ->row_array();
    }

    function create($username, $data) {
        # assume correct format
        if (!is_array($this->get_user_account($username))) {
            $this->db->insert($this->accounts_table, $data);
            return TRUE;
        } else { return FALSE; }
    }
}

```



```
<?php if ( ! defined('BASEPATH')) exit('No direct script access allowed');

class Mysql_radius_model extends MY_Model {

    function __construct(){
        parent::__construct();

        $this->radius_db = $this->load->database('radius', TRUE);
        $this->table = 'radacct';

    }

    function get_username_profile($string) {
        $sql = "select * from ".$this->table." where username = '".$string."'";
        return $this->radius_db->query($sql)->result_array();
    }

    function get_total_input_usage() {
        $today_start = date('Y-m-d 00:00:00');
        $today_end = date('Y-m-d 23:59:00');
        $sql = "select SUM(acctinputoctets) as today_input_total from ".$this->table."
where acctstarttime >= '".$today_start."' AND (acctstoptime <= '".$today_end."' OR
acctstoptime = NULL)";
        $sql_res = $this->radius_db->query($sql)->row_array();
        return $sql_res['today_input_total'];
    }

    function get_total_output_usage() {
        $today_start = date('Y-m-d 00:00:00');
        $today_end = date('Y-m-d 23:59:00');
        $sql = "select SUM(acctoutputoctets) as today_output_total from ".$this->
>table." where acctstarttime >= '".$today_start."' AND (acctstoptime <=
'".$today_end."' OR acctstoptime is NULL)";
        $sql_res = $this->radius_db->query($sql)->row_array();
        return $sql_res['today_output_total'];
    }

    function get_online_users() {
        $today_start = date('Y-m-d 00:00:00');
        $today_end = date('Y-m-d 23:59:00');
        $sql = "select username, framedipaddress, acctstarttime from ".$this->table."
where acctstarttime >= '".$today_start."' AND acctstoptime is NULL";
        $sql_res = $this->radius_db->query($sql)->result_array();
        return $sql_res;
    }

    function get_users_today() {
        $today_start = date('Y-m-d 00:00:00');
        $today_end = date('Y-m-d 23:59:00');
        $sql = "select distinct username, framedipaddress from ".$this->table." where
acctstarttime >= '".$today_start."' AND (acctstoptime <= '".$today_end."' OR
acctstoptime is NULL)";
        $sql_res = $this->radius_db->query($sql)->result_array();
        return $sql_res;
    }
}
```

```
}
```

application/models/Mysql_webmail_model.php

```
<?php if ( ! defined('BASEPATH')) exit('No direct script access allowed');
```

```
class Mysql_webmail_model extends MY_Model {
```

```
function __construct(){
    parent::__construct();
```

```
    $this->webmaildb = 'webmail';
    $this->accounts_table = 'users';
    $this->changepass_table = 'changepass';
    $this->logs_table = 'logs';
    $this->current_db = $this->load->database($this->webmaildb, TRUE);
}
```

```
    #TODO Optimize
function get_user_account($username){
    $this->current_db->where('username', $username);
    $sql_res = $this->current_db->get($this->accounts_table)->row_array();
```

```
    return $sql_res;
}
```

```
}
```

application/models/Proxy_tools_model.php

```
<?php if ( ! defined('BASEPATH')) exit('No direct script access allowed');
```

```
class Proxy_tools_model extends MY_Model {
```

```
function __construct(){
    parent::__construct();
    log_message('debug', 'SSH CMDs initialization commencing...');
    $this->db = $this->load->database('proxydb', TRUE);
    $this->users_table = 'users';
    $this->domains_table = 'domains';
    $this->expr_table = 'expressions';
    $this->logs_table = 'logs';
    $this->load->model("ldap_model");
}
```

```
function get_block_user_info($email){
    $this->db->where('username', $email);
    $sql_res = $this->db->get($this->users_table)->row_array();

    return $sql_res;
}
```

```

function enable_username($email, $reason, $admin){
    $this->db->where('username', $email);
    $this->db->update($this->users_table, array('disabled' => 0, 'reason_unbanned'
=> $reason, 'date_unbanned' => date('Y-m-d H:i:s'), 'tech_unbanner' => $admin));
    $this->ldap_model->enable_ldap_user($email);

    return TRUE;
}

function disable_username($email, $reason, $admin) {
    $temp = $this->get_block_user_info($email);
    if(!empty($temp)){
        $this->db->where('username', $email);
        $this->db->update($this->users_table, array('disabled' => 1, 'reason_banned'
=> $reason, 'date_banned' => date('Y-m-d H:i:s'), 'tech_banner' => $admin));
    } else {
        $this->db->set('username', $email);
        $this->db->insert($this->users_table, array('disabled' => 1, 'reason_banned'
=> $reason, 'date_banned' => date('Y-m-d H:i:s'), 'tech_banner' => $admin));
    }

    $this->ldap_model->disable_ldap_user($email);

    // $cmd = "ssh root@10.16.3.12 \"echo $email > /root/phisher.inc;
/root/./appendPhisher.sh\"";
    // exec($cmd, $results);

    // $cmd = "ssh root@10.16.3.11 \"echo $email > /root/phisher.inc;
/root/./appendPhisher.sh\"";
    // exec($cmd, $results);

    print "Initiating backend operation ...";
    return TRUE;
}

function remove_emails($email) {

    $results_array = array();
    // $cmd = "ssh root@10.16.3.12 /root/./remFromEximStage1Spool.sh $email";
    // exec($cmd, $results);

    // $cmd = "ssh root@10.16.3.11 /root/./remFromEximStage1Spool.sh $email";
    // exec($cmd, $results);

    return $results;
}

function is_email_disabled($email){
    $email_info = $this->get_block_email_info($email);
    if(!empty($email_info['disabled']) && $email_info['disabled'] == TRUE){
        return TRUE;
    } else {
        return FALSE;
    }
}

function get_block_host_info($ip){

```

```

$this->db->where('ip_address', $ip);
$sql_res = $this->db->get($this->hosts_table)->row_array();

return $sql_res;
}

function enable_ip($ip, $reason, $admin){
    $this->db->where('ip_address', $ip);
    $this->db->update($this->hosts_table, array('disabled' => 0, 'reason_unbanned'
=> $reason, 'date_unbanned' => date('Y-m-d H:i:s'), 'tech_unbanner' => $admin));

    // $cmd = "ssh root@10.16.3.12 \"echo $ip > /root/unphisherhost.inc;
/root/./removePhisherHost.sh\"";
    // print $cmd;
    // exec($cmd, $results);

    // $cmd = "ssh root@10.16.3.11 \"echo $ip > /root/unphisherhost.inc;
/root/./removePhisherHost.sh\"";
    // print $cmd;
    // exec($cmd, $results);

    return TRUE;
}

function disable_ip($ip, $reason, $admin) {
    $temp = $this->get_block_host_info($ip);
    if(!empty($temp)){
        $this->db->where('ip_address', $ip);
        $this->db->update($this->hosts_table, array('disabled' => 1, 'reason_banned'
=> $reason, 'date_banned' => date('Y-m-d H:i:s'), 'tech_banner' => $admin));
    } else {
        $this->db->set('ip_address', $ip);
        $this->db->insert($this->hosts_table, array('disabled' => 1, 'reason_banned'
=> $reason, 'date_banned' => date('Y-m-d H:i:s'), 'tech_banner' => $admin));
    }

    // $cmd = "ssh root@10.16.3.12 \"echo $ip > /root/phisherhost.inc;
/root/./appendPhisherHost.sh\"";
    // print $cmd;
    // exec($cmd, $results);

    // $cmd = "ssh root@10.16.3.11 \"echo $ip > /root/phisherhost.inc;
/root/./appendPhisherHost.sh\"";
    // print $cmd;
    // exec($cmd, $results);

    return TRUE;
}

function is_host_disabled($ip){
    $host_info = $this->get_block_host_info($ip);
    if(!empty($host_info['disabled']) && $host_info['disabled'] == TRUE){
        return TRUE;
    } else {
        return FALSE;
    }
}

```

```

function search_email($string) {
    $sql = "select * from ".$this->users_table." where username like
'%" . $string . "%' order by 'disabled' desc";
    return $this->db->query($sql)->result_array();
}

// hosts blocking functions
function search_domain($string) {
    $sql = "select * from ".$this->domains_table." where domain like
'%" . $string . "%' order by 'disabled' desc";
    return $this->db->query($sql)->result_array();
}
}

```

application/models/Sam_ldap_model.php

```

<?php if ( ! defined('BASEPATH')) exit('No direct script access allowed');

class Sam_ldap_model extends CI_Model{

    function __construct() {
        parent::__construct();

        log_message('debug', 'Auth_Ldap initialization commencing...');

        $this->_init();
    }

    private function _init() {

        // Verify that the LDAP extension has been loaded/built-in
        // No sense continuing if we can't
        if ( ! function_exists('ldap_connect')) {
            show_error('LDAP functionality not present. Either load the module
ldap php module or use a php with ldap support compiled in.');
```

log_message('error', 'LDAP functionality not present in php.');

```

        }

        $this->config->load('config_ldap',TRUE);
        $this->config->item('ldap_host','config_ldap');
        $this->host = $this->config->item('ldap_basedn','config_ldap');
        $this->basedn = $this->config->item('ldap_port','config_ldap');
        $this->ldap_port = $this->config->item('ldap_port','config_ldap');
        $this->ldap_attribute = 'uid';
        $this->login_attribute = 'uid';
        $this->auditlog = '/tmp/audit.log';
    }

    public function _ldapconnect(){
        $this->ldapconn = ldap_connect($this->host);

        // At this point, $this->ldapconn should be set. If not... DOOM!
        if(! $this->ldapconn) {
            log_message('error', "Couldn't connect to any LDAP servers.
Bailing...");
            show_error('Error connecting to your LDAP server(s). Please check the
connection and try again.');
```

```

    }

    ldap_set_option($this->ldapconn, LDAP_OPT_PROTOCOL_VERSION, 3);

    return $this->ldapconn;
}

# search for exact match of uid | studentnumber | employee number |
# mailaddresses
# returns boolean
public function search($id)
{
    $this->ldapconn = $this->_ldapconnect();
    $this->bind = ldap_bind($this->ldapconn);

    if(!$this->bind){
        log_message('error', 'Unable to perform anonymous/proxy bind');
        show_error('Unable to bind for user id lookup');
    }

    log_message('debug', 'Successfully bound to directory. Performing dn
lookup for '.$id);
    $filter =
'(|(uid='.$id.')(studentNumber='.$id.')(employeeNumber='.$id.')(mail='.$id.'@up.ed
u.ph)(mail='.$id.'@upd.edu.ph)(mailAlternateAddress='.$id.'@up.edu.ph)(mailAlter
nateAddress='.$id.'@upd.edu.ph))';
    $search = ldap_search($this->ldapconn, $this->basedn, $filter,
array('uid'));
    $entries = ldap_get_entries($this->ldapconn, $search);

    if( !empty($entries) && $entries['count'] > 0 ){
        ldap_unbind($this->ldapconn);
        return TRUE;
    }
    else return FALSE;
}

# search substring of uid
# return all match
public function search_substring($id)
{
    $this->ldapconn = $this->_ldapconnect();
    $this->bind = ldap_bind($this->ldapconn);

    if(!$this->bind){
        log_message('error', 'Unable to perform anonymous/proxy bind');
        show_error('Unable to bind for user id lookup');
    }

    log_message('debug', 'Successfully bound to directory. Performing dn
lookup for '.$id);
    $filter =
'(|(uid=*'.$id.'*)(studentNumber='.$id.')(employeeNumber='.$id.')(mail='.$id.'@up.
edu.ph)(mail='.$id.'@upd.edu.ph)(mailAlternateAddress='.$id.'@up.edu.ph)(mailAlter
nateAddress='.$id.'@upd.edu.ph))';
    $attributes=array('uid','cn','ou');
    $search = ldap_search($this->ldapconn, $this->basedn, $filter,
$attributes);
    $entries = ldap_get_entries($this->ldapconn, $search);

```

```

    return $entries;
}

public function add_ldapentry($user, $mail, $type)
{
    $this->ldapconn = $this->_ldapconnect();

    $binddn = $this->config->item('ldap_root','config_ldap');
    $bindp = $this->config->item('ldap_root_pass','config_ldap');

    $bind = ldap_bind($this->ldapconn, $binddn, $bindp);

    if(!$this->bind){
        log_message('error', 'Unable to perform anonymous/proxy bind');
        show_error('Unable to bind for user id lookup');
    } else {
        $info["cn"] = $user['firstname']." ".$user['lastname'];
        $info["sn"] = $user['lastname'];

        $info["objectclass"][0] = "top";
        $info["objectclass"][1] = "person";
        $info["objectclass"][2] = "inetOrgPerson";
        $info["objectclass"][3] = "qmailUser";
        $info["objectclass"][4] = "radiusprofile";

        $info["dialupAccess"] = 'TRUE';

        if ($type == "student") {
            $info["title"] = "student";
            $info["mailQuotaSize"] = $this->config->item("defaultMailQuota");
            $info["studentNumber"] = $user['id_number'];
        } else {
            $info["title"] = $user['designation'];
            $info["mailQuotaSize"] = $this->config->item("employeeMailQuota");
            $info["employeeType"] = $user['type'];
            $info["employeeNumber"] = $user['id_number'];
            if($user["studno"] != NULL) {
                $info["studentNumber"] = $user['studno'];
            }
        }

        $info["dialupAccess"] = 'TRUE';
        $info["mailHost"] = $this->config->item("defaultMailHost");
        $info["mail"] = $user['email']."@upd.edu.ph";
        $info["mailAlternateAddress"][0] = $user['username']."@upd.edu.ph";

        $info["uid"] = $user['username'];
        $info["ou"] = $user["department"];
        $info["departmentNumber"] = $user["departmentNumber"];

        # Remove old method of creating directories
        // $info["homeDirectory"] = $mail["homedir"];
        // $info["qmailUID"] = $mail["uid"];
        // $info["qmailGID"] = "508";
    }
}

```

```

        $info["mailHost"] = 'imap4.upd.edu.ph';
        $info['mailMessageStore'] =
'upd.edu.ph/' . $user['ou'] . '/' . $user['department'] . '/' . $user['username'];

        mt_srand((double)microtime()*1000000);
        $salt = pack("CCCC", mt_rand(), mt_rand(), mt_rand(), mt_rand());
        $info["userPassword"] = "{SSHA}" . base64_encode(pack("H*",
sha1($user['password'] . $salt)) . $salt);

        $dn = "uid=" . $user['username'] . ",o=University of the
Philippines,c=PH";

        $r = ldap_add($this->ldapconn, $dn, $info);
        ldap_unbind($this->ldapconn);

        if ($r) {
            $data["success"] = TRUE;
            $data["details"] = array (
                "mail" => $info["mail"],
                "mail_alternate" =>
$info["mailAlternateAddress"],
                "uid" => $info["uid"]
            );
        } else {
            $data["success"] = FALSE;
        }
        return $data;
    }
}

public function get_ldapentry($id)
{
    $this->ldapconn = $this->_ldapconnect();
    $this->bind = ldap_bind($this->ldapconn);

    if(!$this->bind){
        log_message('error', 'Unable to perform anonymous/proxy bind');
        show_error('Unable to bind for user id lookup');
    }

    log_message('debug', 'Successfully bound to directory. Performing dn
lookup for '.$id);
    $filter =
'(|(uid='.$id.')(employeeNumber='.$id.')(studentNumber='.$id.))';
    $search = ldap_search($this->ldapconn, $this->basedn, $filter);
    $entries = ldap_get_entries($this->ldapconn, $search);

    // echo "ECHO ENTRIES"; print_r($entries);
    if( !empty($entries) && $entries['count'] > 0 ){

        $ldapentry = array(
            'entrydn' => stripslashes($entries[0]['dn']),
            'name' => $entries[0]['cn'][0],
            'mail' => $entries[0]['mail'][0],
            'mailalternateaddress' => NULL,
            'username' => $entries[0]['uid'][0],
            'mailquotasize' => NULL,
            'ou' => NULL,

```



```

        'departmentnumber' => NULL,
        'employeetype' => NULL,
        'title' => NULL,
        'studentnumber' => NULL,
        'employeenumber' => NULL,
        'mailforwardingaddress' => NULL,
        'deliverymode' => NULL,
        'mailhost' => $entries[0]['mailhost'][0],
    );

    if(isset($entries[0]['mailalternateaddress'])) {
        $ldapentry['mailalternateaddress'] =
$entries[0]['mailalternateaddress'];
    }

    if(isset($entries[0]['mailquotasize'][0])) {
        $ldapentry['mailquotasize'] = $entries[0]['mailquotasize'][0];
    }

    if(isset($entries[0]['ou'][0])) {
        $ldapentry['ou'] = $entries[0]['ou'][0];
    }

    if(isset($entries[0]['departmentnumber'][0])) {
        $ldapentry['departmentnumber'] =
$entries[0]['departmentnumber'][0];
    }

    if(isset($entries[0]['employeetype'][0])) {
        $ldapentry['employeetype'] = $entries[0]['employeetype'][0];
    }

    if(isset($entries[0]['title'][0])) {
        $ldapentry['title'] = $entries[0]['title'][0];
    }

    if(isset($entries[0]['studentnumber'][0])) {
        $ldapentry['studentnumber'] = $entries[0]['studentnumber'][0];
    }

    if(isset($entries[0]['employeenumber'][0])) {
        $ldapentry['employeenumber'] = $entries[0]['employeenumber'][0];
    }

    if(isset($entries[0]['mailforwardingaddress'][0])) {
        $ldapentry['mailforwardingaddress'] =
$entries[0]['mailforwardingaddress'][0];
    }

    // echo "LDAP ENTRY LDAP MODEL"; print_r($ldapentry);
    ldap_unbind($this->ldapconn);
    return $ldapentry;
}
}

```

```

# WARNING: modify with caution
# being used in both updating ldap entry and resetting password
public function set_ldapentry($id, $entrydn, $entry)
{

```

```

$this->ldapconn = $this->_ldapconnect();

$binddn = $this->config->item('ldap_root','config_ldap');
$bindp = $this->config->item('ldap_root_pass','config_ldap');

$bind = ldap_bind($this->ldapconn, $binddn, $bindp);

if(ldap_mod_replace($this->ldapconn, $entrydn, $entry)){
    ldap_unbind($this->ldapconn);
    return true;
}

else{ echo ldap_error($this->ldapconn); return false; }

}

public function _authenticate($username,$password){

$this->ldapconn = $this->_ldapconnect();
$this->bind = ldap_bind($this->ldapconn);

if(!$this->bind){
    log_message('error', 'Unable to perform anonymous/proxy bind');
    show_error('Unable to bind for user id lookup');
}

log_message('debug', 'Successfully bound to directory. Performing dn
lookup for '.$username);
$filter = '(uid='.$username.')';
// $search = ldap_search($this->ldapconn, $this->basedn, $filter,
array('dn','uid','cn'));
$search = ldap_search($this->ldapconn, $this->basedn, $filter);
$entries = ldap_get_entries($this->ldapconn, $search);
$binddn = $entries[0]['dn'];

// Now actually try to bind as the user
$bind = @ldap_bind($this->ldapconn, $binddn, $password);
if(! $bind) {
    $this->_audit("Failed login attempt: ".$username." from
".$_SERVER['REMOTE_ADDR']);
    return FALSE;
}

else {
    $cn = $entries[0]['cn'][0];
    $dn = stripslashes($entries[0]['dn']);
    $id = $entries[0][$this->login_attribute][0];
    $ldapinfo = $entries[0];

    $this->ldap_model->_audit("Successful login: ".$cn."(".$username.")
from ".$this->input->ip_address());

    $admin = $this->db->where("username", $username)
        ->get("admin_users")->row_array();

    if (empty($admin) || $admin == NULL) { return FALSE; }

    // Set the session data

```

```

        $admindata = array('user' => $username,
                           'admin_name' => $cn,
                           'logged_in' => TRUE,
                           'admin_type' => $admin["role"]);

        $this->session->set_userdata($admindata);

        return TRUE;
    }
}

public function _audit($msg){
    $date = date('Y/m/d H:i:s');
    if( ! file_put_contents($this->auditlog, $date." :
".$msg."\n", FILE_APPEND)) {
        log_message('info', 'Error opening audit log '.$this->auditlog);
        return FALSE;
    }
    return TRUE;
}

public function get_by_id_number($sn) {
    $this->ldapconn = $this->_ldapconnect();

    $binddn = $this->config->item('ldap_root','config_ldap');
    $bindp = $this->config->item('ldap_root_pass','config_ldap');

    $bind = ldap_bind($this->ldapconn, $binddn, $bindp);
    if(!$bind){
        log_message('error', 'Unable to perform anonymous/proxy bind');
        show_error('Unable to bind for user id lookup');
        echo "bind fail";
    } else {
        $filter = '(|(studentNumber='.$sn.')(employeeNumber='.$sn.'))';
        $attributes=array('uid','cn','ou');
        $search = ldap_search($this->ldapconn, $this->basedn, $filter,
$attributes);
        $entries = ldap_get_entries($this->ldapconn, $search);

        return $entries;
    }
}

public function is_eligible_for_creation($student_no) {
    $user = $this->get_by_id_number($student_no);
    if ( $user["count"] != 0) {
        return FALSE;
    }
    return TRUE;
}
}

```

application/models/Sam_user_model.php

```
<?php if ( ! defined('BASEPATH')) exit('No direct script access allowed');
```

```
class Sam_user_model extends CI_Model {
```

```
function __construct(){
    parent::__construct();
    $this->table = 'users';
}
```

```
/**
 * Getter Functions
 */
```

```
function get_by_name($user) {
    $this->db->where('firstname', $user["firstname"])
        ->where('middlename', $user["middlename"])
        ->where('lastname', $user["lastname"])
        ->where('activated', 1);
    return $this->db->get($this->table)
        ->row_array();
}
```

```
function get_by_student_number($student_no) {
    return $this->db->where('id_number', $student_no)
        ->or_where("studno", $student_no)
        ->join('courses', 'users.course_id = courses.id', "left")
        ->join('aliases', 'aliases.unit_id = users.unit_id', "left")
        ->get($this->table)
        ->row_array();
}
```

```
function get_by_email_or_username($user) {
    return $this->db->where('email', $user)
        ->or_where('username', $user)
        ->get($this->table)
        ->row_array();
}
```

```
/**
 * Checker Functions
 */
```

```
function is_eligible_for_creation($student_no) {
    if ($this->get_by_student_number($student_no) == NULL) {
        return TRUE;
    }
    return FALSE;
}
```

```
/**
 * Create Functions
 */
```

```
function create($user) {
    # assume correct format
    if ($this->get_by_student_number($user['studno']) == NULL) {
        $this->db->insert($this->table, $user);
        return TRUE;
    } else {
        return FALSE;
    }
}
```

```

}

}

function create_home_directories($user) {
    $ssshpath = "/usr/bin/ssh";
    $host = $this->config->item("defaultMailHostUPD");

    $campus = "diliman";
    $homedir="/home/smicro02/".$campus;

    $cmd = $ssshpath." -l root $host '/bin/mkdir -p ".$homedir."'";
    system($cmd,$exitstatus);

    $hostcmd = "/bin/cat $cmd >> /tmp/ldap/hostcmd";
    system($hostcmd,$exitstatus);
    $homedir="/home/smicro02/$campus/".$user["ou"];

    $cmd = $ssshpath." -l root $host '/bin/mkdir -p ".$homedir."'";
    system($cmd,$exitstatus);
    $hostcmd = "/bin/cat $cmd >> /tmp/ldap/hostcmd";
    system($hostcmd,$exitstatus);

    // create system user account
    $fullname = $user["firstname"]." ".$user['lastname'];
    $fullname = trim($fullname, "\t\n\r\0\x0B");
    $fullname = str_replace(" ", "-", $fullname);
    $fullname = str_replace("-", "_", $fullname);
    $fullname = str_replace(".", "_", $fullname);
    $fullname = str_replace("'", "_", $fullname);
    $fullname = str_replace(" ", "_", $fullname);

    $loc_cmd="/usr/sbin/useradd -c ' ".$fullname."' -d
    ".$homedir."/ ".$user["username"]." -m -k /var/webmail/users/skel -g mailuser -s
    /sbin/nologin ".$user["username"]." &";
    $cmd= $ssshpath." -l root $host.\" ".$loc_cmd.\"";
    system($cmd,$exitstatus);

    $hostcmd = "/bin/cat $cmd >> /tmp/ldap/hostcmd";
    system($hostcmd,$exitstatus);

    // set quota //
    $loc_cmd= "/usr/sbin/edquota -p test0 ".$user["username"];
    $cmd= $ssshpath." -l root $host \" ".$loc_cmd\"";
    system ($cmd,$exitstatus);

    $hostcmd = "/bin/cat $cmd >> /tmp/ldap/hostcmd";
    system($hostcmd,$exitstatus);

    // get uid //
    $fp = popen ($ssshpath." -l root $host \"grep \"^\" ".$user["username"].":\"
    /etc/passwd | cut -d: -f3\"", "r");
    $id = fgets($fp, 4096);
    pclose($fp);

    $id = chop($id);
    if(!is_numeric($id)){
        printf("ERROR: Invalid UID, please contact the administrator.");
    }
}

```

```

        die();
    }

    $fp = popen ($sshpath." -l root $host \"grep \"^\".$user[\"username\"].\" :\n
/etc/passwd | cut -d: -f4\\\"\", \"r\");
    $gid = fgets($fp, 4096);
    pclose($fp);
    $gid = chop($gid);

    $result = array (
        "homedir"    => $homedir."/\".$user[\"username\"],
        "uid"        => $id,
        "gid"        => $gid,
    );

    $data["details"] = $result;
    $data["success"] = true;
    return $data;
}

/**
 * Update Functions
 */

function enable_force_changepass($username){
    $this->db->where('username', $username);
    $this->db->update('users', array('force_changepass' => 1));

    return TRUE;
}
}

```

application/models/Units_model.php

```

<?php if ( ! defined('BASEPATH')) exit('No direct script access allowed');

class Units_model extends CI_Model {

    function __construct() {
        parent::__construct();
        $this->table = 'units';
    }

    function get_subunits($parent = null) {
        if ($parent == null) { $parent = 0; }
        return $this->db->where('parent_id', $parent)
            ->join('aliases', 'aliases.unit_id = units.unit_id')
            ->group_by('aliases.unit_id')
            ->order_by("aliases.name", "asc")
            ->get($this->table)->result_array();
    }

    function get_all() {
        return $this->db->join('aliases', 'aliases.unit_id = units.unit_id')

```

```

        ->group_by('aliases.unit_id')
        ->order_by("aliases.name", "asc")
        ->get($this->table)->result_array();
    }

    function get_acronym_from_name($name) {
        $unit = $this->db->where('name', $name)
            ->get("aliases")->row_array();
        if ($unit == NULL) { return ""; }
        return $unit["acronym"];
    }
}

```

application/models/User_model.php

```

<?php if ( ! defined('BASEPATH')) exit('No direct script access allowed');

class User_model extends MY_Model {

    function __construct(){
        parent::__construct();
        $this->tech_table = 'tech_users';
    }

    #TOFIX: Improve this function
    function admin_user_exists($username)
    {
        $q = "SELECT * FROM tech_users WHERE `username` = ?";
        $r = $this->db->query($q, array($username));

        if($r->num_rows() > 0){
            return true;
        }
        else
        {
            return false;
        }
    }

    function admin_last_login($username){
        $this->db->where('username', $username);
        $this->db->update($this->tech_table, array('last_login' => date('Y-m-d
H:i:s')));

        return TRUE;
    }
}

```

application/views/admin/form_newaccount.php

```

<div class="content-header container">

```

```

        <span class="fa-stack">
            <i class="fa fa-square-o fa-stack-2x"></i>
            <i class="fa fa-user fa-stack-1x"></i>
        </span>
        Create DilNet account
    </div>
    <?php if(validation_errors() != false): ?>
        <div class="container">
            <div class='col-xs-12 alert alert-danger'>
                <?php echo validation_errors(); ?>
            </div>
        </div>
    <?php endif; ?>
    <div class="content container">
        <?php echo form_open('activation/verify'); ?>
        <table class="table table-centered table-striped table-bordered">
            <tr>
                <td class="col-md-4"><h6 style="padding-left:10px;"></i><span
class="text-info">VERIFY</span> Student Information</h6></td>
                <td class="col-md-8">
                    <p>If info does not match, <span class="text-danger">DO NOT
CREATE.</span><br />
                    Student should go to OUR-CRS to correct his/her details.</p>
                    <?php if(isset($student_display)): ?>
                        <table class="table table-bordered table-centered table-min">
                            <?php foreach($student_display as $key => $value): ?>
                                <tr>
                                    <td><?php echo $key; ?></td>
                                    <td>
                                        <?php echo $student_display[$key]; ?>
                                    </td>
                                </tr>
                            <?php endforeach; ?>
                        </table>
                    <?php else: ?>
                        <hr />
                        <p class="text-danger">Student has no CRS record.</p>
                    <?php endif; ?>
                </td>
            </tr>
        </table>
        <!--
            <tr>
                <td><h6 style="padding-left:10px;"><span class="text-
info">CONFIRM</span> Account Creation</h6></td>
                <td>
                    <p>Type or scan your <span class="text-info">employee number</span> to
create account.</p>
                    <div class="row">
                        <div class="col-xs-9">
                            <div class="text" name="empno" class="form-control"
placeholder="Employee Number" autocomplete="off">
                                </div>
                            </div>
                        </td>
                    </tr>
                    <?php if($this->input->ip_address() == '10.40.2.47'): ?>
                        <tr>
                            <td><h4 style="padding-left:10px;"><span class="text-
info">PRINTER</span></h4><h6 style="padding-left:10px;">Select Printer</h6></td>

```



```

        <div class="row">
        <div class="col-xs-9">
            <p>
                <select name="printer" placeholder="Printer" value="-- Select
Reason --" class="select-block">
                    <option value="tech-thermal">Tech Printer TM-T82
(Thermal)</option>
                    <option value="tech-ribbon">Tech Printer TM-U220
(Ribbon)</option>
                    <option value="admin-thermal">Admin Printer TM-T82
(Thermal)</option>
                </select>
            </p>
        </div>
        </div>
    </td>
</tr>
<?php endif; ?>
<tr>
    <td colspan="2"></td>
</tr>
<tr>
    <td colspan="2" style="width: 100%">
        <input type="hidden" name="studentno" value=<?php echo $studentno;
?>>
        <div class="row">
        <div class="col-xs-4">
            <?php echo anchor("/", "Cancel", 'class="btn btn-lg btn-danger col-
xs-12"'); ?>
        </div>
        <div class="col-xs-8">
            <button type="submit" class="btn btn-lg btn-primary col-xs-
12">Create DilNet account</button>
        </div>
        </div>
    </td>
</tr>
</table>

<?php echo form_close(); ?>
</div> <!-- div.container -->

```

```

application/views/admin/form_resetpass.php

```

```

<div class="content-header container">
    <span class="fa-stack">
        <i class="fa fa-square-o fa-stack-2x"></i>
        <i class="fa fa-lock fa-stack-1x"></i>
    </span>
    Reset Password
</div>
<?php if(validation_errors() != false): ?>
    <div class="container">

```

```

        <div class='col-xs-12 alert alert-danger'>
        <?php echo validation_errors(); ?>
        </div>
    </div>
<?php endif; ?>

<div class="content container">
    <?php echo form_open('accounts/resetpass'); ?>
    <table class="table table-centered table-striped table-bordered">
    <tr>
        <td class="col-md-4"><h4 style="padding-left:10px;"></i><span
class="text-info">VERIFY</span></h4><h6 style="padding-left:10px;">Student
Information</h6></td>
        <td class="col-md-8">
            <p>If info does not match, <span class="text-danger">do not
reset.</span>
            <br />Student should go to OUR-CRS to correct his/her details.</p>
            <table class="table table-bordered table-nested">
            <?php if(isset($student_ucc_display)): ?>
            <?php foreach($student_ucc_display as $key => $value): ?>
            <?php if(!empty($value)): ?>
            <tr>
                <th><?php echo $key; ?></th>
                <td>
                    <?php echo $student_ucc_display[$key]; ?>
                </td>
            </tr>
            <?php endif; ?>
            <?php endforeach; ?>
            <?php endif; ?>
            </table>
            <table class="table table-bordered table-centered table-min">
            <?php if(isset($status)): ?>
            <tr>
                <th>Status</th>
                <td><span style="font-weight: bold;"><?php echo $status;
?></span></td>
            </tr>
            <?php endif; ?>
            <?php if(isset($student_crs_display)): ?>
            <?php if(isset($student_crs_display as $key => $value): ?>
            <?php foreach($student_crs_display as $key => $value): ?>
            <?php if(!empty($value)): ?>
            <tr>
                <th><?php echo $key; ?></th>
                <td>
                    <?php echo $student_crs_display[$key]; ?>
                </td>
            </tr>
            <?php endif; ?>
            <?php endforeach; ?>
            <?php else: ?>
            <hr />
            <p class="text-danger">Student has no CRS record.</p>
            <?php endif; ?>
            </table>
        </td>
    </tr>
    <?php if(!$is_enrolled): ?>
    <tr>

```

```

        <td><h4 style="padding-left:10px;"><span class="text-
info">REASON</span></h4><h6 style="padding-left:10px;">for Password
Reset</h6></td>
        <td>
        <div class="row">
        <div class="col-xs-9">
        <p><span class="text-danger">DO NOT RESET</span> if reason is not in
the list.</p>
        <select name="reason" placeholder="Reason" value="-- Select Reason --"
class="select-block">
        <option value="">-- Select Reason --</option>
        <option value="NOT ENROLLED: Returning Student">Returning Student /
Readmission</option>
        <option value="NOT ENROLLED: University Clearance">University
Clearance</option>
        <option value="NOT ENROLLED: View CRS Grades">View CRS
Grades</option>
        <option value="NOT ENROLLED: Applying for LOA">Applying for
LOA</option>
        </select>
        </div>
        </div>
        </td>
    </tr>
    <?php elseif($is_enrolled): ?>
    <input type="hidden" name="reason" value="Enrolled student for AY[[]]">
    <?php endif; ?>
    <?php if($this->input->ip_address() == '10.40.2.47'): ?>
    <tr>
        <td><h4 style="padding-left:10px;"><span class="text-
info">PRINTER</span></h4><h6 style="padding-left:10px;">Select Printer</h6></td>
        <td>
        <div class="row">
        <div class="col-xs-9">
        <p>
        <select name="printer" placeholder="Printer" value="-- Select
Reason --" class="select-block">
        <option value="tech-thermal">Tech Printer TM-T82
(Termal)</option>
        <option value="tech-ribbon">Tech Printer TM-U220
(Ribbon)</option>
        <option value="admin-thermal">Admin Printer TM-T82
(Thermal)</option>
        </select>
        </p>
        </div>
        </div>
        </td>
    </tr>
    <?php endif; ?>
    <tr>
        <td colspan="2"></td>
    </tr>
    <tr>
        <td colspan="2" style="width: 100%">
        <td colspan="2" style="width: 100%">
        <input type="hidden" name="studentno" value=<?php echo $studentno;
?>>
        <input type="hidden" name="is_enrolled" value=<?php echo
$is_enrolled ? 1 : 0; ?>>
    </td>
    </tr>

```

```

        <div class="row">
        <div class="col-xs-4">
        <?php echo anchor("accounts", "Cancel", 'class="btn btn-lg btn-
danger col-xs-12"'); ?>
        </div>
        <div class="col-xs-8">
        <button type="submit" class="btn btn-lg btn-primary col-xs-12">Reset
Password</button>
        </div>
        </div>
    </td>
</tr>
</table>
<?php echo form_close(); ?>
</div> <!-- div.container -->

```

application/views/admin/index.php

```

<div class="row">
    <div class="col-xs-12">
        <!-- Horizontal Form -->
        <div class="box box-info">
            <div class="box-header with-border">
                <h3 class="box-title"><i class="fa fa-barcode"></i> Scan Form 5 or UP
ID</h3>
            </div><!-- /.box-header -->
            <?php echo validation_errors('<p class="text-danger">', '</p>'); ?>
            <!-- form start -->
            <?php echo form_open('accounts', 'class="form-card"'); ?>
            <div class="box-body">
                <div class="col-xs-12">
                    <?php echo form_error('merge_str'); ?>
                    <div class="input-group">
                        <div class="input-group-addon"><i class="glyphicon glyphicon-
barcode"></i></div>
                        <input type="text" class="form-control input-lg" name="idno"
placeholder="Enter SN xxxx-xxxxx or barcode number" required autofocus>
                    </div><!-- /.input-group -->
                </div>
            </div><!-- /.box-body -->
            <div class="box-footer">
                <button type="submit" class="btn btn-info col-xs-12">Check</button>
                <script>
                    $('input-group').on('focus', '.form-control', function () {
                        $(this).closest('input-group').addClass('focus');
                    });
                </script>
            </div><!-- /.box-footer -->
        </div><!-- /.row -->
    </div><!-- /.box -->
</div><!-- /.col-md-6 -->

```

application/views/mailer/welcome.php

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD XHTML 1.0 Transitional //EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-
transitional.dtd"><html><head><title></title><meta http-equiv="Content-Type"
content="text/html; charset=utf-8"><meta name="viewport" content="width=320,
target-densitydpi=device-dpi">
<style type="text/css">
/* Client-specific Styles */
#outlook a { padding: 0; }/* Force Outlook to provide a "view in browser" button.
*/
body { width: 100% !important; }
.ReadMsgBody { width: 100%; }
.ExternalClass { width: 100%; display: block !important; } /* Force Hotmail to
display emails at full width */
/* Reset Styles */
/* Add 100px so mobile switch bar doesn't cover street address. */
body { background-color: #ecec; margin: 0; padding: 0; }
img { outline: none; text-decoration: none; display: block; }
br, b br, b br, em br, i br { line-height: 100%; }
h1, h2, h3, h4, h5, h6 { line-height: 100% !important; -webkit-font-smoothing:
antialiased; }
h1 a, h2 a, h3 a, h4 a, h5 a, h6 a { color: blue !important; }
h1 a:active, h2 a:active, h3 a:active, h4 a:active, h5 a:active, h6 a:active {
color: red !important; }
/* Preferably not the same color as the normal header link color. There is
limited support for psuedo classes in email clients, this was added just for good
measure. */
h1 a:visited, h2 a:visited, h3 a:visited, h4 a:visited, h5 a:visited, h6
a:visited { color: purple !important; }
/* Preferably not the same color as the normal header link color. There is limited
support for psuedo classes in email clients, this was added just for good measure.
*/
table td, table tr { border-collapse: collapse; }
.yshortcuts, .yshortcuts a, .yshortcuts a:link, .yshortcuts a:visited, .yshortcuts
a:hover, .yshortcuts a span {
color: black; text-decoration: none !important; border-bottom: none !important;
background: none !important;
} /* Body text color for the New Yahoo. This example sets the font of
Yahoo's Shortcuts to black. */
/* This most probably won't work in all email clients. Don't include code blocks
in email. */
code {
white-space: normal;
word-break: break-all;
}
#background-table { background-color: #ecec; }
/* Webkit Elements */
#top-bar { border-radius: 6px 6px 0px 0px; -moz-border-radius: 6px 6px 0px 0px; -
webkit-border-radius: 6px 6px 0px 0px; -webkit-font-smoothing: antialiased;
background-color: #4b4044; color: #72a6a6; }
#top-bar a { font-weight: bold; color: #72a6a6; text-decoration: none; -
webkit-border-radius: 0px 0px 6px 6px; -moz-border-radius: 0px 0px 6px 6px; -
#footer { border-radius: 0px 0px 6px 6px; -webkit-font-smoothing: antialiased; }
/* Fonts and Content */
```

```

body, td { font-family: HelveticaNeue, Arial, sans-serif; }
.header-content, .footer-content-left, .footer-content-right { -webkit-text-
size-adjust: none; -ms-text-size-adjust: none; }
/* Prevent Webkit and Windows Mobile platforms from changing default font sizes
on header and footer. */
.header-content { font-size: 12px; color: #72a6a6; }
.header-content a { font-weight: bold; color: #72a6a6; text-decoration: none; }
#headline p { color: #F3F3F3; font-family: HelveticaNeue, Arial, sans-serif;
font-size: 36px; text-align: center; margin-top:0px; margin-bottom:30px; }
#headline p a { color: #F3F3F3; text-decoration: none; }
.article-title { font-size: 18px; line-height:24px; color: #fe6461; font-
weight:bold; margin-top:0px; margin-bottom:18px; font-family: HelveticaNeue,
Arial, sans-serif; }
.article-title a { background-color: #55bcda; color: white; text-decoration:
none; border-radius:6px; -moz-border-radius: 6px; -webkit-border-radius:6px; -
webkit-font-smoothing: antialiased; padding: 10px; margin: 10px 0; font-size:
14px;}
.article-title.with-meta {margin-bottom: 0;}
.article-meta { font-size: 13px; line-height: 20px; color: #ccc; font-weight:
bold; margin-top: 0;}
.article-content { font-size: 13px; line-height: 18px; color: #444444; margin-
top: 0px; margin-bottom: 18px; font-family: HelveticaNeue, Arial, sans-serif; }
.article-content a { color: #7f8c4f; font-weight:bold; text-decoration:none; }
.article-content img { max-width: 100% }
.article-content ol, .article-content ul { margin-top:0px; margin-bottom:18px;
margin-left:19px; padding:0; }
.article-content li { font-size: 13px; line-height: 18px; color: #444444; }
.article-content li a { color: #7f8c4f; text-decoration:underline; }
.article-content p {margin-bottom: 15px;}
.footer-content-left { font-size: 12px; line-height: 15px; color: #72a6a6;
margin-top: 0px; margin-bottom: 15px; }
.footer-content-left a { color: #d5e9e7; font-weight: bold; text-decoration:
none; }
.footer-content-right { font-size: 11px; line-height: 16px; color: #72a6a6;
margin-top: 0px; margin-bottom: 15px; }
.footer-content-right a { color: #d5e9e7; font-weight: bold; text-decoration:
none; }
#footer { background-color: #4b4c44; color: #72a6a6; }
#footer a { color: #d5e9e7; text-decoration: normal; }
#permission-reminder { white-space: normal; }
#street-address { color: #d5e9e7; white-space: normal; }
</style>
<!--[if gte mso 9]>
<style _tmplitem="157" >
.article-content ol, .article-content ul {
margin: 0 0 0 24px;
padding: 0;
list-style-position: inside;
}
</style>
<![endif]-->
</head>
<body style="background-color: #ececce;margin: 0;padding: 0;font-family:
HelveticaNeue, Arial, sans-serif;width: 100% !important;">
<table width="100%" cellpadding="0" cellspacing="0" border="0" id="background-
table" style="background-color: #ececce;">
<tbody>
<tr style="border-collapse: collapse;">

```

```

        <td align="center" bgcolor="#ecec" style="font-family: HelveticaNeue,
Arial, sans-serif;border-collapse: collapse;">
        <table class="w640" style="margin:0 10px;" width="640" cellpadding="0"
cellspacing="0" border="0">
            <tbody>
                <tr style="border-collapse: collapse;"><td class="w640"
width="640" height="20" style="font-family: HelveticaNeue, Arial, sans-
serif;border-collapse: collapse;"></td></tr>
                <tr style="border-collapse: collapse;">
                    <td class="w640" width="640" style="font-family: HelveticaNeue,
Arial, sans-serif;border-collapse: collapse;">
                        <table id="top-bar" class="w640" width="640" cellpadding="0"
cellspacing="0" border="0" bgcolor="#d5e9e7" style="border-radius: 6px 6px 0px
0px;-moz-border-radius: 6px 6px 0px 0px;-webkit-border-radius: 6px 6px 0px 0px;-
webkit-font-smoothing: antialiased;background-color: #4b4044;color: #72a6a6;">
                            <tbody>
                                <tr style="border-collapse: collapse;">
                                    <td class="w15" width="15" style="font-family:
HelveticaNeue, Arial, sans-serif;border-collapse: collapse;"></td>
                                    <td class="w325" width="350" valign="middle" align="left"
style="font-family: HelveticaNeue, Arial, sans-serif;border-collapse: collapse;">
                                        <table class="w325" width="350" cellpadding="0"
cellspacing="0" border="0">
                                            <tbody>
                                                <tr style="border-collapse: collapse;"><td
class="w325" width="350" height="8" style="font-family: HelveticaNeue, Arial,
sans-serif;border-collapse: collapse;"></td></tr>
                                                </tbody>
                                            </table>
                                            <div class="header-content" style="-webkit-text-size-
adjust: none;-ms-text-size-adjust: none;font-size: 12px;color: #72a6a6;"></div>
                                        <!-- Unsubscribe link -->
                                        <table class="w325" width="350" cellpadding="0"
cellspacing="0" border="0">
                                            <tbody>
                                                <tr style="border-collapse: collapse;"><td
class="w325" width="350" height="8" style="font-family: HelveticaNeue, Arial,
sans-serif;border-collapse: collapse;"></td></tr>
                                                </tbody>
                                            </table>
                                            <td>
                                                <td class="w30" width="30" style="font-family:
HelveticaNeue, Arial, sans-serif;border-collapse: collapse;"></td>
                                                <td class="w255" width="255" valign="middle" align="right"
style="font-family: HelveticaNeue, Arial, sans-serif;border-collapse: collapse;">
                                                    <table class="w255" width="255" cellpadding="0"
cellspacing="0" border="0">
                                                        <tbody>
                                                            <tr style="border-collapse: collapse;"><td
class="w255" width="255" height="8" style="font-family: HelveticaNeue, Arial,
sans-serif;border-collapse: collapse;"></td></tr>
                                                            </tbody>
                                                        </table>
                                                        <table class="w255" width="255" cellpadding="0"
cellspacing="0" border="0">
                                                            <tbody>
                                                                <tr style="border-collapse: collapse;"><td
class="w255" width="255" height="8" style="font-family: HelveticaNeue, Arial,
sans-serif;border-collapse: collapse;"></td></tr>
                                                                </tbody>
                                                            </table>
                                                        </td>
                                                    </table>
                                                </td>
                                            </table>
                                        </td>
                                    </tr>
                                </tbody>
                            </table>
                    </td>
                </tr>
            </tbody>
        </table>

```

```

        </tbody>
    </table>
</td>
    <td class="w15" width="15" style="font-family:
HelveticaNeue, Arial, sans-serif;border-collapse: collapse;"></td>
    </tr>
</tbody>
</table>
</td>
</tr>
<tr style="border-collapse: collapse;">
    <td id="header" class="w640" width="640" align="center"
bgcolor="#55bcda" style="font-family: HelveticaNeue, Arial, sans-serif;border-
collapse: collapse;">
        <table class="w640" width="640" cellpadding="0"
cellspacing="0" border="0">
            <tbody>
                <tr style="border-collapse: collapse;">
                    <td class="w30" width="30" style="font-family:
HelveticaNeue, Arial, sans-serif;border-collapse: collapse;"></td>
                    <td class="w580" width="580" height="30" style="font-
family: HelveticaNeue, Arial, sans-serif;border-collapse: collapse;"></td>
                    <td class="w30" width="30" style="font-family:
HelveticaNeue, Arial, sans-serif;border-collapse: collapse;"></td>
                </tr>
                <tr style="border-collapse: collapse;">
                    <td class="w30" width="30" style="font-family:
HelveticaNeue, Arial, sans-serif;border-collapse: collapse;"></td>
                    <td class="w580" width="580" style="font-family:
HelveticaNeue, Arial, sans-serif;border-collapse: collapse;">
                        <div align="center" id="headline">
                            <p style="color: #F3F3F3;font-family: HelveticaNeue,
Arial, sans-serif;font-size: 36px;text-align: center;margin-top: 0px;margin-
bottom: 30px;"><singleline label="Title">Welcome to UP Diliman
Webmail</singleline></p>
                        </div>
                    </td>
                    <td class="w30" width="30" style="font-family:
HelveticaNeue, Arial, sans-serif;border-collapse: collapse;"></td>
                </tr>
            </tbody>
        </table>
    </td>
    <td class="w640" width="640" align="center" style="border-collapse: collapse;">
        <tr style="border-collapse: collapse;">
            <td class="w640" width="640" height="30" bgcolor="#ffffff" style="font-family: HelveticaNeue,
Arial, sans-serif;border-collapse: collapse;"></td></tr>
            <tr id="simple-content-row" style="border-collapse: collapse;">
                <td class="w640" width="640" bgcolor="#ffffff" style="font-
family: HelveticaNeue, Arial, sans-serif;border-collapse: collapse;">
                    <table align="left" class="w640" width="640" cellpadding="0"
cellspacing="0" border="0">
                        <tbody>
                            <tr style="border-collapse: collapse;">
                                <td class="w30" width="30" style="font-family:
HelveticaNeue, Arial, sans-serif;border-collapse: collapse;"></td>
                                <td class="w580" width="580" style="font-family:
HelveticaNeue, Arial, sans-serif;border-collapse: collapse;">
                                    <repeater>

```



```

<layout label="Text only">
  <table class="w580" width="580" cellpadding="0"
cellspacing="0" border="0">
    <tbody>
      <tr style="border-collapse: collapse;">
        <td class="w580" width="580" style="font-
family: HelveticaNeue, Arial, sans-serif;border-collapse: collapse;">
          <div align="left" class="article-content"
style="font-size: 13px;line-height: 18px;color: #444444;margin-top: 0px;margin-
bottom: 18px;font-family: HelveticaNeue, Arial, sans-serif;">
            <p align="left" style="font-size:
18px;margin-bottom: 15px;">Dear <?php echo $user['firstname']; ?>,</p>
            <multiline label="Description">
              </multiline><p style="margin-bottom:
15px;">You can now use your UPD Webmail addresses for official academic
correspondences. You can also use it to contact us for support queries.</p>
              <p style="margin-bottom: 15px;">Not
sure where to start? Below are some tips on using your new UPD Webmail.</p>
            </div>
          </td>
        </tr>
      <tr style="border-collapse: collapse;"><td
class="w580" width="580" height="10" style="font-family: HelveticaNeue, Arial,
sans-serif;border-collapse: collapse;"></td></tr>
    </tbody>
  </table>
</layout>
<layout label="Text only">
  <table class="w580" width="580" cellpadding="0"
cellspacing="0" border="0">
    <tbody>
      <tr style="border-collapse: collapse;">
        <td class="w580" width="580" style="font-
family: HelveticaNeue, Arial, sans-serif;border-collapse: collapse;">
          <p align="left" class="article-title"
style="font-size: 18px;line-height: 24px;color: #fe6461;font-weight: bold;margin-
top: 0px;margin-bottom: 18px;font-family: HelveticaNeue, Arial, sans-
serif;"><singleline label="Title">Your UPD Webmail Address</singleline></p>
          <div align="left" class="article-content"
style="font-size: 13px;line-height: 18px;color: #444444;margin-top: 0px;margin-
bottom: 18px;font-family: HelveticaNeue, Arial, sans-serif;">
            <multiline label="Description">
              </multiline><p style="margin-bottom:
15px;">Upon signing up for a DilNet account, you are given the following
addresses:</p>
              <p style="margin-bottom: 15px;">
                <b>Primary email address:</b> <?php
echo $user['email']; ?><br style="line-height: 100%;">
                <b>Alternate email
address(es):</b><?php echo $user['username']. "@upd.edu.ph"; ?>
              </p>
              <p style="margin-bottom: 15px;">You
can send and receive emails through these addresses which we call as
<b>identities.</b> You can give either of them for email correspondences.</p>

```

`<p style="margin-bottom: 15px;">To`
select which one to use for sending emails, you can manage your identities with
the link provided below:`</p>`

`</div>`
`<div class="article-title" style="font-size: 18px;line-height: 24px;color: #fe6461;font-weight: bold;margin-top: 0px;margin-bottom: 18px;font-family: HelveticaNeue, Arial, sans-serif;">Manage my identities</div>`

`</td>`
`</tr>`
`<tr style="border-collapse: collapse;"><td class="w580" width="580" height="10" style="font-family: HelveticaNeue, Arial, sans-serif;border-collapse: collapse;"></td></tr>`
`</tbody>`
`</table>`
`</layout>`
`<layout label="Text only">`
`<table class="w580" width="580" cellpadding="0"`

`cellspacing="0" border="0">`

`<tbody>`
`<tr style="border-collapse: collapse;">`
`<td class="w580" width="580" style="font-family: HelveticaNeue, Arial, sans-serif;border-collapse: collapse;">`
`<p align="left" class="article-title" style="font-size: 18px;line-height: 24px;color: #fe6461;font-weight: bold;margin-top: 0px;margin-bottom: 18px;font-family: HelveticaNeue, Arial, sans-serif;">`
`<singleline label="Title">Mailbox Size`
`and Mail Forwarding</singleline>`

`</p>`
`</td>`
`</tr>`
`<tr style="border-collapse: collapse;">`
`<td class="w580" width="580" style="font-family: HelveticaNeue, Arial, sans-serif;border-collapse: collapse;">`
`<div align="left" class="article-content" style="font-size: 13px;line-height: 18px;color: #444444;margin-top: 0px;margin-bottom: 18px;font-family: HelveticaNeue, Arial, sans-serif;">`
`<multiline label="Description">`
`</multiline><p style="margin-bottom: 15px;">Mailbox size is a feature we're continuously improving as users require more storage for their emails. To view your current allocation, look at the icon on the lower left part of your Inbox.</p>`

`<p style="margin-bottom: 15px;">UPD`
Webmail supports mail forwarding so you can read your emails from another mail provider (like Gmail or Yahoo). Just remember that when you turn on your mail forwarding, `<em>you will not receive any messages in your UPD Webmail inbox until you turn your mail forwarding off again.</em></p>`

`</div>`
`<div class="article-title" style="font-size: 18px;line-height: 24px;color: #fe6461;font-weight: bold;margin-top: 0px;margin-bottom: 18px;font-family: HelveticaNeue, Arial, sans-serif;">`
`<a href="https://mail.upd.edu.ph/?_task=settings&_action=plugin.upd_mailforward"`

```
style="background-color: #55bcda;color: white;text-decoration: none;border-radius:
6px;-moz-border-radius: 6px;-webkit-border-radius: 6px;-webkit-font-smoothing:
antialiased;padding: 10px;margin: 10px 0;font-size: 14px;">Turn on mail
forwarding</a>
</div>
</td>
</tr>
<tr style="border-collapse: collapse;"><td
class="w580" width="580" height="10" style="font-family: HelveticaNeue, Arial,
sans-serif;border-collapse: collapse;"></td></tr>
</tbody>
</table>
</layout>
<layout label="Text only">
<table class="w580" width="580" cellpadding="0"
cellspacing="0" border="0">
<tbody>
<tr style="border-collapse: collapse;">
<td class="w580" width="580" style="font-
family: HelveticaNeue, Arial, sans-serif;border-collapse: collapse;">
<p align="left" class="article-title"
style="font-size: 18px;line-height: 24px;color: #fe6461;font-weight: bold;margin-
top: 0px;margin-bottom: 18px;font-family: HelveticaNeue, Arial, sans-
serif;"><singleline label="Title">DilNet Account Security</singleline></p>
</td>
</tr>
<tr style="border-collapse: collapse;">
<td class="w580" width="580" style="font-
family: HelveticaNeue, Arial, sans-serif;border-collapse: collapse;">
<div align="left" class="article-content"
style="font-size: 13px;line-height: 18px;color: #444444;margin-top: 0px;margin-
bottom: 18px;font-family: HelveticaNeue, Arial, sans-serif;">
<multiline label="Description">
</multiline><p style="margin-bottom:
15px;">UPD Webmail is just one of the services you can access using your DilNet
account. It's important to make sure your account is secure.</p>
<p style="margin-bottom: 15px;">You
can change your password only at <b>https://accounts.upd.edu.ph</b>. Remember that
if your DilNet account password is compromised, your access to UPD Webmail and
other UP Diliman services (CRS, UVLE, iLib, DILNET Wifi) will most likely be
compromised as well.</p>
<p style="margin-bottom:
15px;">Setting up your account recovery is important too. It will allow you to
reset your password in case you forget it in the future.</p>
</div>
<div class="article-title" style="font-
size: 18px;line-height: 24px;color: #fe6461;font-weight: bold;margin-top:
0px;margin-bottom: 18px;font-family: HelveticaNeue, Arial, sans-serif;"><a
href="http://accounts.upd.edu.ph/" style="background-color: #55bcda;color:
white;text-decoration: none;border-radius: 6px;-moz-border-radius: 6px;-webkit-
border-radius: 6px;-webkit-font-smoothing: antialiased;padding: 10px;margin: 10px
0;font-size: 14px;">Secure my DilNet account</a></div>
</td>
</tr>
<tr style="border-collapse: collapse;">
```

```
 <multiline label="Description"> </multiline><p style="margin-bottom: 15px;">We emphasize this to all of our users -- <b>do not give your DilNet account password to anyone, even to us!</b> The University Computer Center will never ask for your password through email, phone or in person.</p>  <p style="margin-bottom: 15px;">Do not click links that you think are suspicious, and do not give any personal information to an unknown website. If you are unsure whether the email came from us, contact our DilNet Helpdesk.</p> |
```

```


```

```


```

```


```

```


```

```


```

```

white;text-decoration: none;border-radius: 6px;-moz-border-radius: 6px;-webkit-
border-radius: 6px;-webkit-font-smoothing: antialiased;padding: 10px;margin: 10px
0;font-size: 14px;">Visit the DilNet Helpdesk website</a></div>
</td>
</tr>
<tr style="border-collapse: collapse;"><td
class="w580" width="580" height="10" style="font-family: HelveticaNeue, Arial,
sans-serif;border-collapse: collapse;"></td></tr>
</tbody>
</table>
</layout>
<layout label="Text only">
<table class="w580" width="580" cellpadding="0"
cellspacing="0" border="0">
<tbody>
<tr style="border-collapse: collapse;">
<td class="w580" width="580" style="font-
family: HelveticaNeue, Arial, sans-serif;border-collapse: collapse;">
<div align="left" class="article-content"
style="font-size: 13px;line-height: 18px;color: #444444;margin-top: 0px;margin-
bottom: 18px;font-family: HelveticaNeue, Arial, sans-serif;">
<multiline label="Description">
Thanks, <br style="line-height: 100%;">
<br style="line-height: 100%;">
UPD Webmail Team <br style="line-height:
100%;">
University Computer Center <br
University of the Philippines Diliman
</multiline>
</div>
</td>
</tr>
<tr style="border-collapse: collapse;"><td
class="w580" width="580" height="10" style="font-family: HelveticaNeue, Arial,
sans-serif;border-collapse: collapse;"></td></tr>
</tbody>
</table>
</layout>
</repeater>
</td>
<td class="w30" width="30" style="font-family:
HelveticaNeue, Arial, sans-serif;border-collapse: collapse;"></td>
</tr>
</tbody>
</table>
</td>
<tr>
<tr style="border-collapse: collapse;"><td class="w640"
width="640" height="15" bgcolor="#ffffff" style="font-family: HelveticaNeue,
Arial, sans-serif;border-collapse: collapse;"></td></tr>
<tr style="border-collapse: collapse;">
<td class="w640" width="640" style="font-family: HelveticaNeue,
Arial, sans-serif;border-collapse: collapse;">
<table id="footer" class="w640" width="640" cellpadding="0"
cellspacing="0" border="0" bgcolor="#4b4c44" style="border-radius: 0px 0px 6px
6px;-moz-border-radius: 0px 0px 6px 6px;-webkit-border-radius: 0px 0px 6px 6px;-
webkit-font-smoothing: antialiased;background-color: #4b4c44;color: #72a6a6;">

```

white;text-decoration: none;border-radius: 6px;-moz-border-radius: 6px;-webkit-border-radius: 6px;-webkit-font-smoothing: antialiased;padding: 10px;margin: 10px 0;font-size: 14px;">Visit the DilNet Helpdesk website</div>

</td>
</tr>
<tr style="border-collapse: collapse;"><td
class="w580" width="580" height="10" style="font-family: HelveticaNeue, Arial,
sans-serif;border-collapse: collapse;"></td></tr>
</tbody>
</table>
</layout>
<layout label="Text only">
<table class="w580" width="580" cellpadding="0"

cellspacing="0" border="0">
<tbody>
<tr style="border-collapse: collapse;">
<td class="w580" width="580" style="font-
family: HelveticaNeue, Arial, sans-serif;border-collapse: collapse;">
<div align="left" class="article-content"
style="font-size: 13px;line-height: 18px;color: #444444;margin-top: 0px;margin-
bottom: 18px;font-family: HelveticaNeue, Arial, sans-serif;">
<multiline label="Description">
Thanks, <br style="line-height: 100%;">

<br style="line-height: 100%;">
UPD Webmail Team <br style="line-height:
100%;">
University Computer Center <br

style="line-height: 100%;">
University of the Philippines Diliman
</multiline>
</div>
</td>

</tr>
<tr style="border-collapse: collapse;"><td
class="w580" width="580" height="10" style="font-family: HelveticaNeue, Arial,
sans-serif;border-collapse: collapse;"></td></tr>
</tbody>
</table>
</layout>
</repeater>

</td>
<td class="w30" width="30" style="font-family:
HelveticaNeue, Arial, sans-serif;border-collapse: collapse;"></td>
</tr>
</tbody>
</table>
</td>

</tr>
<tr style="border-collapse: collapse;"><td class="w640"
width="640" height="15" bgcolor="#ffffff" style="font-family: HelveticaNeue,
Arial, sans-serif;border-collapse: collapse;"></td></tr>
<tr style="border-collapse: collapse;">
<td class="w640" width="640" style="font-family: HelveticaNeue,
Arial, sans-serif;border-collapse: collapse;">
<table id="footer" class="w640" width="640" cellpadding="0"
cellspacing="0" border="0" bgcolor="#4b4c44" style="border-radius: 0px 0px 6px
6px;-moz-border-radius: 0px 0px 6px 6px;-webkit-border-radius: 0px 0px 6px 6px;
-moz-border-radius: 0px 0px 6px 6px;-webkit-border-radius: 0px 0px 6px 6px;
background-color: #4b4c44;color: #72a6a6;">

```

        <tbody>
        <tr style="border-collapse: collapse;">
            <td class="w30" width="30" style="font-family:
HelveticaNeue, Arial, sans-serif;border-collapse: collapse;"></td>
            <td class="w580 h0" width="360" height="30" style="font-
family: HelveticaNeue, Arial, sans-serif;border-collapse: collapse;"></td>
            <td class="w0" width="60" style="font-family:
HelveticaNeue, Arial, sans-serif;border-collapse: collapse;"></td><td class="w0"
width="160" style="font-family: HelveticaNeue, Arial, sans-serif;border-collapse:
collapse;"></td>
            <td class="w30" width="30" style="font-family:
HelveticaNeue, Arial, sans-serif;border-collapse: collapse;"></td>
        </tr>
        <tr style="border-collapse: collapse;">
            <td class="w30" width="30" style="font-family:
HelveticaNeue, Arial, sans-serif;border-collapse: collapse;"></td>
            <td class="w580 h0" width="360" height="15" style="font-
family: HelveticaNeue, Arial, sans-serif;border-collapse: collapse;"></td>
            <td class="w0" width="60" style="font-family:
HelveticaNeue, Arial, sans-serif;border-collapse: collapse;"></td>
            <td class="w0" width="160" style="font-family:
HelveticaNeue, Arial, sans-serif;border-collapse: collapse;"></td>
            <td class="w30" width="30" style="font-family:
HelveticaNeue, Arial, sans-serif;border-collapse: collapse;"></td>
        </tr>
        </tbody>
    </table>
</td>
</tr>
<tr style="border-collapse: collapse;"><td class="w640"
width="640" height="60" style="font-family: HelveticaNeue, Arial, sans-
serif;border-collapse: collapse;"></td></tr>
</tbody>
</table>
</td>
</tr>
</tbody>
</table>
</body>
</html>

```

application/views/partials/error.php

```

<div class="block-card clearfix">
<span class="block-card-text"><?php echo $errmsg; ?></span>
<?php echo anchor('accounts', 'OK', 'class="col-md-4 col-md-offset-4 btn btn-primary
btn-lg" role="button"') ?>
</div>

```

application/views/partials/footer.php

```

<!-- Main Footer -->
<footer class="main-footer">

```

```

<!-- To the right -->
<div class="pull-right hidden-xs">
    tech.accounts enhanced v1.0
</div>
<!-- Default to the left -->
<strong>Copyright &copy; 2017 <a href="http://ucc.upd.edu.ph">University
Computer Center</a>.</strong> All rights reserved.
</footer>

```

application/views/partials/header.php

```

<!-- Main Header -->
<header class="main-header">

<!-- Logo -->
<a href="<?php echo base_url('tech/dilnet_lookup'); ?>" class="logo">
    <!-- mini logo for sidebar mini 50x50 pixels -->
    <span class="logo-mini">UCC</span>
    <!-- logo for regular state and mobile devices -->
    <span class="logo-lg"><b>tech</b>.accounts</span>
</a>

<!-- Header Navbar -->
<nav class="navbar navbar-static-top" role="navigation">
    <!-- Sidebar toggle button-->
    <a href="#" class="sidebar-toggle" data-toggle="offcanvas"
role="button">
        <span class="sr-only">Toggle navigation</span>
    </a>
    <!-- Navbar Right Menu -->
    <div class="navbar-custom-menu">
        <ul class="nav navbar-nav">
            <li class="dropdown user user-menu hidden-xs">
                <?php
                    $attributes = array('class' => 'navbar-form', 'role' =>
'search', 'id' => 'searchForm');
                    echo form_open('tech/dilnet_lookup/navbar_search_form',
$attributes);
                ?>
                <div class="input-group <?php echo
(form_error('navbar_search_str')) == TRUE ? 'has-error' : ''; ?>">
                    <input type="text" class="form-control" size="25"
placeholder="Search for a DilNet account" name="navbar_search_str">
                    <span class="input-group-btn">
                        <button class="btn btn-default" type="submit" data-
toggle="tooltip" data-placement="auto" data-html="true" title="Search"><i
class="fa fa-search"></i></button>
                    </span>
                </div><!-- /input-group -->
                <?php echo form_close(); ?>
            </li>
            <li class="dropdown user user-menu">
                <!-- Menu Toggle Button -->
                <a href="#" class="dropdown-toggle" data-toggle="dropdown">
                    <!-- The user image in the navbar-->

```



```

        <!-- hidden-xs hides the username on small devices so only the
image appears. -->
        <span data-toggle="tooltip" data-placement="auto" data-
html="true" title="<?php if($this->session->userdata('ldap_name') != NULL) { echo
ucwords($this->session->userdata('ldap_name')); } else { echo "No Name"; }
?>"><?php if($this->session->userdata('ldap_name') != NULL) { echo "<i class='fa
fa-user fa-lg hidden-xs'></i><i class='fa fa-user hidden-lg hidden-md hidden-
sm'></i> ".ucwords($this->session->userdata('ldap_name')); } else { echo "No
Name"; } ?> </span>
        </a>
    </li>
    <li>
        <!-- Log Out Button -->
        <a href="<?php echo base_url('welcome/logout'); ?>"
        <i class="fa fa-sign-out" data-toggle="tooltip" data-
placement="auto" data-html="true" title="Sign Out"></i>
        </a>
    </li>
    <!-- Control Sidebar Toggle Button -->
    <!--
    <li>
        <a href="#" data-toggle="control-sidebar"><i class="fa fa-
gears"></i></a>
    </li>
    -->
</ul>
</div>
</nav>
</header>

```

application/views/partials/left_sidebar.php

```

<!-- Left side column. contains the logo and sidebar -->
<aside class="main-sidebar">
    <!-- sidebar: style can be found in sidebar.less -->
    <section class="sidebar">
        <!-- Sidebar Menu -->
        <ul class="sidebar-menu">
            <?php if($this->acl->hasPermission('dashboard')): ?>
                <?php if($this->acl->hasPermission('dashboard')): ?><a href="<?php
                <li class="<?php echo active_link('dashboard'); ?>"><i class="fa fa-
echo site_url('tech/dashboard'); ?>"></i><span>Dashboard</span></a></li>
dashboard"></i><span>Dashboard</span></a></li>
                <?php endif; ?>
            <?php if($this->acl->hasRole('tech')): ?>
                <!-- TECH TOOLS -->
                <li class="header">TECH TOOLS</li>
                <?php endif; ?>
            <?php if($this->acl->hasPermission('dilnet_lookup')): ?>
                <?php if($this->acl->hasPermission('dilnet_lookup')): ?><a href="<?php
                <li class="<?php echo active_link('dilnet_lookup'); ?>"><i class="fa fa-
echo site_url('tech/dilnet_lookup/name_search_form/new'); ?>"></i><span>Search DilNet Accounts</span></a></li>
search"></i><span>Search DilNet Accounts</span></a></li>
                <?php endif; ?>
            <?php if($this->acl->hasPermission('create_user')): ?>

```

```

        <li class="<?php echo active_link('create_user'); ?>"><a href="<?php
echo site_url('tech/create_user'); ?>"><i class="fa fa-user-plus"></i><span>Create
Employee Account</span></a></li>
        <?php endif; ?>
        <?php if($this->acl->hasPermission('create_special')): ?>
        <li class="<?php echo active_link('create_special'); ?>"><a
href="<?php echo site_url('tech/create_special'); ?>"><i class="fa fa-user-
secret"></i><span>Create Special Account</span></a></li>
        <?php endif; ?>
        <?php if($this->acl->hasPermission('merge')): ?>
        <li class="<?php echo active_link('merge'); ?>"><a href="<?php echo
site_url('tech/merge/merge_search_form/new'); ?>"><i class="fa fa-
clone"></i><span>Merge Accounts</span></a></li>
        <?php endif; ?>

        <?php if($this->acl->hasRole('tech')): ?>
        <li class="header">MAIL TOOLS</li>
        <?php endif; ?>
        <?php if($this->acl->hasPermission('mail_tools')): ?>
        <li class="<?php if($this->uri->segment(3)=='mail_search'){echo
"active";}}?>"><a href="<?php echo site_url('tech/mail_tools/mail_search/new');
?>"><i class="fa fa-search"></i> <span>Search Mail Logs</span></a></li>
        <?php endif; ?>
        <?php if($this->acl->hasPermission('mail_blocking')): ?>
        <li class="<?php if($this->uri->segment(3)=='email_blocking'){echo
"active";}}?>"><a href="<?php echo site_url('tech/mail_tools/email_blocking/new');
?>"><i class="fa fa-envelope-o"></i> <span>Block/Unblock Email</span></a></li>
        <?php endif; ?>
        <?php if($this->acl->hasPermission('host_blocking')): ?>
        <li class="<?php if($this->uri->segment(3)=='host_blocking'){echo
"active";}}?>"><a href="<?php echo site_url('tech/mail_tools/host_blocking/new');
?>"><i class="fa fa-desktop"></i> <span>Block/Unblock Host</span></a></li>
        <?php endif; ?>

        <?php if($this->acl->hasRole('tech')): ?>
        <li class="header">PROXY TOOLS</li>
        <?php endif; ?>

        <?php if($this->acl->hasPermission('radius_profile')): ?>
        <li class="<?php echo active_link('radius_profile'); ?>"><a
href="<?php echo site_url('tech/radius_profile'); ?>"><i class="fa fa-
database"></i> <span>Radius Profile</span></a></li>
        <?php if($this->acl->hasPermission('user_blocking')): ?>
        <li class="<?php if($this->uri->segment(3)=='user_blocking'){echo
"active";}}?>"><a href="<?php echo site_url('tech/proxy_tools/user_blocking/new');
?>"><i class="fa fa-user-times"></i> <span>Block/Unblock Client</span></a></li>
        <?php endif; ?>
        <!--
        <?php if($this->acl->hasPermission('domain_blocking')): ?>
        <li class="<?php if($this->uri->segment(3)=='domain_blocking'){echo
"active";}}?>"><a href="<?php echo site_url('tech/proxy_tools/domain_blocking');
?>"><i class="fa fa-bank"></i> <span>Block/Unblock Domain</span></a></li>
        <?php endif; ?>
        <?php if($this->acl->hasPermission('expr_blocking')): ?>
        <li class="<?php if($this->uri->segment(3)=='expr_blocking'){echo
"active";}}?>"><a href="<?php echo site_url('tech/proxy_tools/expr_blocking');
?>"><i class="fa fa-bank"></i> <span>Block/Unblock Domain</span></a></li>
        <?php endif; ?>

```

```

?>"<i class="fa fa-commenting"></i> <span>Block/Unblock
Expressions</span></a></li>
    <?php endif; ?>
    -->

    <?php endif; ?>

    <?php if($this->acl->hasRole('admin')): ?>
    <!-- ADMIN TOOLS -->
    <li class="header">ADMIN TOOLS</li>
    <?php endif; ?>

    <?php if($this->acl->hasPermission('accounts')): ?>
    <li class="<?php echo active_link('accounts'); ?>"><a href="<?php echo
site_url('accounts'); ?>"><i class="fa fa-users"></i><span>Activate Student
Account</span></a></li>
    <?php endif; ?>

    <?php if($this->acl->hasRole('tech')): ?>
    <!-- ADVANCED TOOLS -->
    <!--
    <li class="header">MISC TOOLS</li>
    <?php endif; ?>

    <?php if($this->acl->hasPermission('crs_apichecker')): ?>
    <li class="<?php echo active_link('crs_apichecker'); ?>"><a href="<?php echo site_url('supertech/crs_apichecker'); ?>"><i class="fa fa-
database"></i><span>CRS API Checker</span></a></li>
    <?php endif; ?>
    <?php if($this->acl->hasPermission('mysql_lookup')): ?>
    <li class="<?php echo active_link('mysql_lookup'); ?>"><a href="<?php
echo site_url('supertech/mysql_lookup'); ?>"><i class="fa fa-database"></i>
<span>MySQL Lookup</span></a></li>
    <?php endif; ?>
    <?php if($this->acl->hasPermission('ldap_lookup')): ?>
    <li class="<?php echo active_link('ldap_lookup'); ?>"><a href="<?php
echo site_url('supertech/ldap_lookup'); ?>"><i class="fa fa-book"></i> <span>LDAP
Lookup</span></a></li>
    <?php endif; ?>
    <?php if($this->acl->hasPermission('webmail_lookup')): ?>
    <li class="<?php echo active_link('webmail_lookup'); ?>"><a href="<?php
echo site_url('supertech/webmail_lookup'); ?>"><i class="fa fa-
envelope-o"></i><span>Roundcube Mail Lookup</span></a></li>
    <?php endif; ?>
    <?php if($this->acl->hasPermission('search_logs')): ?>
    <li class="<?php echo active_link('search_logs'); ?>"><a href="<?php
echo site_url('supertech/search_logs'); ?>"><i class="fa fa-file-text-
o"></i><span>View Logs</span></a></li>
    <?php endif; ?>
    -->

</ul><!-- /.sidebar-menu -->

```

```

    </section>
    <!-- /.sidebar -->
</aside>

```

application/views/partials/login.php

```

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="utf-8">
    <title>TECH.accounts</title>
    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <!-- Loading Bootstrap -->
    <link href="<?php echo base_url();>assets/bootstrap/css/bootstrap.css"
rel="stylesheet">

    <!-- Loading Login CSS -->
    <link href="<?php echo base_url();>assets/css/Login.css" rel="stylesheet">

    <!-- HTML5 shim, for IE6-8 support of HTML5 elements. All other JS at the
end of file. -->
    <!--[if lt IE 9]>
    <script src="js/html5shiv.js"></script>
    <script src="js/respond.min.js"></script>
    <![endif]-->
</head>
<body>
    <!-- Main content -->
    <div class="container">
        <form method="post" action="<?php echo site_url('welcome/login');
?>" id="form-signin">
            <div class="form-signin-glyph"><i class="glyphicon glyphicon-
cog"></i></div>
            <h3 class="form-signin-heading">Sign in to tech.accounts</h3>
            <?php echo validation_errors('<h5 class="text-danger form-
signin-text">', '</h5>'); ?>
            <input type="text" class="form-control" name="un"
placeholder="Username" required autofocus><br />
            <input type="password" class="form-control" name="pwd"
placeholder="Password" required><br />
            <button type="submit" class="btn btn-primary btn-block">Sign
in</button>
        </form>
    </div>

    <!-- Load JS here for greater good =====>
    <script src="<?php echo base_url();>assets/js/jquery-
1.8.3.min.js"></script>
    <script src="<?php echo base_url();>assets/js/jquery-ui-
1.10.3.custom.min.js"></script>
    <script src="<?php echo base_url();>assets/js/jquery.ui.touch-
punch.min.js"></script>
    <script src="<?php echo base_url();>assets/js/bootstrap.min.js"></script>
    <script>
        $(function(){ // $(document).ready shorthand

```

```

        $('#form-signin').hide().fadeIn(1000);
    });
</script>
</body>
</html>

```

application/views/partials/message_id_actions.php

```

<div class="box-footer">
    <div class="row">
        <div class="col-xs-12">
            <?php if ($this->uri->segment(3) == 'view_message_id'): ?>
                <a class="btn btn-app" href="<?php echo
site_url('tech/mail_tools/block_sender/' . $mail_ID); ?>" data-toggle="tooltip"
data-placement="auto" data-html="true" title="Block Sender"><i class="fa fa-user-
times text-warning"></i> Block Sender</a>
                <a class="btn btn-app" href="<?php echo
site_url('tech/mail_tools/block_sender/' . $mail_ID); ?>" data-toggle="tooltip"
data-placement="auto" data-html="true" title="Block Sender IP"><i class="fa fa-
lock text-warning"></i> Block Sender IP</a>
                <a class="btn btn-app" href="<?php echo
site_url('tech/mail_tools/block_recipient/' . $mail_ID); ?>" data-toggle="tooltip"
data-placement="auto" data-html="true" title="Block Recipient"><i class="fa fa-
user-times text-warning"></i> Block Recipient</a>
            <?endif; ?>
        </div>
    </div><!-- /.row -->
</div><!-- div.box-footer -->

```

application/views/partials/print.php

```

<!DOCTYPE html>
<html lang="en">
    <head>
        <meta charset="utf-8">
        <title>DilNet Account</title>
        <meta name="viewport" content="width=device-width, initial-scale=1.0">
        <!-- Loading Flat UI -->
        <link href="<?php echo base_url();?>assets/css/print.css" rel="stylesheet">
        <link rel="shortcut icon" href="images/favicon.ico">

        <!-- HTML5 shim, for IE6-8 support of HTML5 elements. All other JS at the end
of file. -->
        <!--[if lt IE 9]>
            <script src="js/html5shiv.js"></script>
            <script src="js/respond.min.js"></script>
        <![endif]-->
    </head>
    <body>
        <div class="print-container">
            <div class="print-heading" style="letter-spacing: 3px;">DILNET ACCOUNT
            <div class="print-heading"><?php if (isset($print_title)): ?>
                <?php echo " : ".$print_title; ?>
            <?php endif; ?>
        </div>
    </body>

```

```

<table class="print-main">
  <tbody>
    <?php foreach($print_array as $key => $value): ?>
      <tr>
        <th><?php echo strtoupper($key); ?></th>
        <td><?php echo $print_array[$key]; ?></td>
      </tr>
    <?php endforeach; ?>
  </tbody>
</table>
<table class="print-info">
  <tr>
    <td colspan="2">CHANGE YOUR PASSWORD at
<strong>https://accounts.upd.edu.ph</strong></td>
  </tr>
  <tr>
    <th><b>Dilnet Website:</b></th><td>https://dilnet.upd.edu.ph</td>
  </tr>
  <tr>
    <th><b>Mail Site:</b></th><td>https://mail.upd.edu.ph</td>
  </tr>
  <tr>
    <th><b>CRS Site:</b></th><td>https://crs.upd.edu.ph</td>
  </tr>
</table>
</div>
<script>window.print();</script>
</body>
</html>

```

application/views/partials/right_sidebar.php

```

<!-- Control Sidebar -->
<aside class="control-sidebar control-sidebar-dark">
  <!-- Create the tabs -->
  <ul class="nav nav-tabs nav-justified control-sidebar-tabs">
    <li class="active"><a href="#control-sidebar-home-tab" data-
toggle="tab"><i class="fa fa-home"></i></a></li>
    <li><a href="#control-sidebar-settings-tab" data-toggle="tab"><i
class="fa fa-gears"></i></a></li>
  </ul>
  <!-- Tab panes -->
  <div class="tab-content">
    <!-- Home tab content -->
    <div class="tab-pane active" id="control-sidebar-home-tab">
      <h3 class="control-sidebar-heading">Recent Activity</h3>
      <ul class="control-sidebar-menu">
        <li>
          </li>
      </ul><!-- /.control-sidebar-menu -->
    </div><!-- /.tab-pane -->
    <!-- Stats tab content -->
    <div class="tab-pane" id="control-sidebar-stats-tab">Stats Tab
Content</div><!-- /.tab-pane -->
    <!-- Settings tab content -->
    <div class="tab-pane" id="control-sidebar-settings-tab">
      <form method="post">

```



```

    </div>
</div><!-- /.row -->
</div><!-- div.box-footer -->

```

application/views/404.php

```

<div class="error-page clearfix">
    <h2 class="headline text-yellow">404</h2>
    <div class="error-content" style="margin-top: 40px;">
        <h3>Oops! Page not found.</h3>
        <p>We could not find the page you were looking for. Contact the dev
team if you feel this is incorrect.</p>
    </div>
</div><!-- /.error-page -->

```

application/views/account_disable_confirm.php

```

<div class="box box-solid" style="padding-left: 15px;">
    <div class="box-body">
        <?php // print "<pre>"; print_r($account); print "</pre>"; ?>
        <?php // print "<pre>"; print_r($account); print "</pre>"; ?>
        <h4>You are going to <strong><?php echo $account['disabled'] ? "enable" :
"disable" ?></strong> this account. <br /><br><?php echo $account['disabled'] ?
"When enabled, the following will be restored: <br><li><i>Change Password,
<li>Mail Delivery, <li>Proxy Access</i><br><br>" : "When disabled, the user will
not be able to change password along with the restrictions chosen below. <br><br>
?>
        <span class="text-danger">REVIEW</span> the information below before you
continue.</h4>
        <hr>
        <div>
            <h3 style="margin-top: 10px;"><?php echo strtoupper($account['fullname']);
?></h3>
            <h4><?php echo strtoupper($username); ?></h4>
            <h4><?php echo strtoupper($employee_number); ?></h4>
            <h4><?php echo "EN: ".$account['employeenumber']; ?></h4>
            <h4><?php echo "SN: ".$account['studentnumber']; ?></h4>
            <?php echo form_open('tech/dilnetacc_tools/disable_confirm'); ?>

            <hr>
            <div class="form-group <?php echo (form_error('disable_reason') == '') ? '' :
'has-error'; ?>">
                <label for="disable_reason">Reason for <?php echo $account['disabled'] ?
"Re-enabling" : "Disabling" ?></label>
                <input type="text" data-toggle="tooltip" data-placement="auto" data-
html="true" title="Type reason for disabling the account here." class="form-
control" name="disable_reason" value="<?php echo set_value('disable_reason');
?>">
                <input type="hidden" class="form-control" name="username" value="<?php echo
$username; ?>">
                <?php echo form_error('disable_reason'); ?>
            </div>
            <?php echo form_error('disable_reason')==FALSE){ ?>
            <?php if($account['disabled']==FALSE){ ?>

<div class="row">
    <div class="col-xs-6 col-sm-6 col-md-4 col-lg-3">

```



```

        <label style="font-weight: 400">
            <input type="checkbox" name="mail_disable" value="Disable Mail
Delivery" <?php echo set_checkbox('mail_disable', 'Disable Mail Delivery', TRUE);
?> /> <strong>Disable Mail Delivery</strong>
        </label>
    </div>
    <div class="col-xs-6 col-sm-6 col-md-4 col-lg-3">
        <label style="font-weight: 400">
            <input type="checkbox" name="proxy_disable" value="Disable Proxy Access"
<?php echo set_checkbox('proxy_disable', 'Disable Proxy Access', TRUE); ?> />
        <strong>Disable Proxy Access</strong>
    </label>
    </div>
</div>
<?php } ?>
</div>
</div><!-- /.box-body -->
<div class="box-footer">
    <div class="row">
        <div class="col-xs-5 col-sm-4 col-md-4 col-lg-4 col-xl-4">
            <button type="submit" class="btn btn-lg <?php echo $account['disabled']
? 'btn-success' : 'btn-danger'?>"><?php echo $account['disabled'] ? "<i class='fa
fa-unlock'></i> Enable" : "<i class='fa fa-lock'></i> Disable" ?> Account</button>
        </div>
        <div class="col-xs-offset-7 col-sm-offset-6 col-md-offset-4">
            <a class="btn btn-lg btn-default" href="<?php echo
site_url('tech/dilnetacc_tools/disable_cancel/' . $username); ?>" data-
toggle="tooltip" data-placement="auto" data-html="true" title="Go Back"><i
class="fa fa-arrow-left"></i> Go Back</a>
            <?php echo form_close(); ?>
        </div>
    </div><!-- /.row -->
</div><!-- div.box-footer -->
</div><!-- div.box -->

```

application/views/account_reset_confirm.php

```

<div class="box box-solid" style="padding-left: 15px;">
    <div class="box-body">
        <h4>You are going to <strong>reset</strong> the password of this account. <br
/><br />
        <span class="text-danger">REVIEW</span> the information below before you
        continue.</h4>
        <hr>
        <div>
            <h3 style="margin-top: 10px;"><?php echo strtoupper($account['fullname']);
?></h3>
            <h4><?php echo strtoupper($username); ?></h4>
            <h4><?php echo "EN: " . $account['employeenumber']; ?></h4>
            <h4><?php echo "SN: " . $account['studentnumber']; ?></h4>
            <h4><?php echo "SN: " . $account['studentnumber']; ?></h4>
            <?php echo form_open('tech/dilnetacc_tools/reset_confirm'); ?>
            <?php echo form_error('reset_reason') == '' ? '' :
            <div class="form-group"><?php echo form_error('reset_reason') == '' ? '' :
            'has-error'; ?>>
            <label for="reset_reason">Reason</label>
            <input type="text" class="form-control" name="reset_reason" value="<?php
echo set_value('reset_reason'); ?>">
            <input type="hidden" class="form-control" name="username" value="<?php echo
$username; ?>">

```

```

        <?php echo form_error('reset_reason'); ?>
    </div>
</div>
</div><!-- /.box-body -->
<div class="box-footer">
    <div class="row">
        <div class="col-xs-5 col-sm-4 col-md-4">
            <button type="submit" class="btn btn-lg btn-primary" onclick="return
confirm('This will reset the user\'s password. Continue?');"><i class="fa fa-
key"></i> Reset Password</button>
        </div>
        <div class="col-xs-offset-7 col-sm-offset-6 col-md-offset-4">
            <a class="btn btn-lg btn-default" href="<?php echo
site_url('tech/dilnetacc_tools/reset_cancel/' . $username); ?>" data-
toggle="tooltip" data-placement="auto" data-html="true" title="Go Back"><i
class="fa fa-arrow-left"></i> Go Back</a>
            <?php echo form_close(); ?>
        </div>
    </div><!-- /.row -->
</div><!-- div.box-footer -->
</div><!-- div.box -->

```

application/views/account_special_create_confirm.php

```

<div class="row">
    <div class="col-xs-12">
        <div class="box box-primary">
            <div class="box-header with-border">
                <div class="box-title">Create <strong>Special</strong>
                <h3 class="box-title">Create <strong>Special</strong>
Account > Review</h3>
            </div><!-- /.box-header -->
            <div class="box-body">
                <div class="table-responsive no-padding">
                    <div class="table-danger">REVIEW</span> the
                    <h4><span class="text-danger">REVIEW</span> the
information below before creating this account:</h4>
                    <table class="table table-hover">
                        <thead><tr>
                            <th>Item</th>
                            <th>Description</th>
                        </tr>
                    </thead>
                    <tbody>
                        <?php
                        foreach($account as $key => $value) {
                            if(!is_numeric($key)){
                                if
(array_key_exists($key,$account_fieldnames) == TRUE){
                                    echo '<tr><th
class="col-md-3">' . $account_fieldnames[$key] . "</th>";
                                }
                                else { echo '<tr><th
class="col-md-3">' . $key . "</th>"; }
                                echo '<td class="col-md-
9">' . $value . "</td></tr>';
                            }
                        }
                    }
                }
            </div>
        </div>
    </div>
</div>

```

```

        ?>

        <tr>
            <td colspan="2">
                <?php echo
anchor("tech/create_special/activate", "Create DilNet Account", 'class="btn btn-lg
btn-primary col-xs-12 col-md-8"'); ?>
                <div class="col-md-1"></div>
                <?php echo
anchor("tech/create_special", "Go Back", 'class="btn btn-lg btn-danger col-xs-12
col-md-3"'); ?>
            </td>
        </tr>
    </table>
</div><!-- /div.box-body -->
</div><!-- /.box-primary -->
</div><!-- /div.col-xs-12 -->
</div>

```

application/views/account_special_create.php

```

<div class="row">
    <div class="col-xs-12">
        <div class="box box-primary">
            <div class="box-header with-border">
                <h3 class="box-title"><i class="fa fa-user-secret"></i> Create Special
Account</h3>
                <br><h5> Fields with <strong><span class="text-info">*</span></strong> are
required.</h5>
            </div><!-- /.box-header -->
            <!-- form start -->
            <?php echo form_open('tech/create_special'); ?>
            <div class="box-body">

                <div class="form-group row">
                    <div class="col-md-4 col-xs-12 <?php echo (form_error('firstname') ==
'') ? '' : 'has-error form-group'; ?>">
                        <label for="name">First Name <span class="text-info">*</span></label>
                        <input type="text" class="form-control" name="firstname" value="<?php
echo set_value('firstname'); ?>" placeholder="Juan">
                        <?php echo form_error('firstname'); ?>
                    </div>
                    <div class="col-md-3 col-xs-12 <?php echo (form_error('middlename') ==
'') ? '' : 'has-error form-group'; ?>">
                        <label for="name">Middle Name <span class="text-info">*</span></label>
                        <input type="text" class="form-control" name="middlename"
id="middlename" value="<?php echo set_value('middlename'); ?>"
placeholder="Santos">
                        <?php echo form_error('middlename'); ?>
                    </div>
                    <div class="col-md-3 col-xs-12 <?php echo (form_error('lastname') == '')
? '' : 'has-error form-group'; ?>">
                        <label for="name">Last Name <span class="text-info">*</span></label>
                        <input type="text" class="form-control" id="lastname" name="lastname"
placeholder="de la Cruz" value="<?php echo set_value('lastname'); ?>"
                        <?php echo form_error('lastname'); ?>
                    </div><!-- /.row -->
                <div class="col-md-2 col-xs-12" style="margin-top: 5px;">

```

```

        <div><label><input type="checkbox" id="mn_none" name="mn_none"
class="minimal" <?php if($this->input->post('mn_none') === 'on') { echo "checked";
} ?>> No Middle Name</label></div>
        <div><label> <input type="checkbox" id="ln_none" name="ln_none"
class="minimal" <?php if($this->input->post('ln_none') === 'on') { echo "checked";
} ?>> No Last Name</label></div>
    </div>
</div>
<hr>

<div class="form-group <?php echo (form_error('type') == '') ? '' : 'has-
error'; ?>" >
    <label>Account Type <span class="text-info">*</span></label>
    <div>
        <label class="checkbox-inline">
            <input type="radio" name="type" value="OFC" class="minimal" <?php
if($this->input->post('type') === 'OFC') { echo "checked"; } ?>>
            Office/Unit
        </label>
        <label class="checkbox-inline">
            <input type="radio" name="type" value="PROJ" class="minimal" <?php
if($this->input->post('type') === 'PROJ') { echo "checked"; } ?>>
            Project/Program
        </label>
    </div>
    <?php echo form_error('type'); ?>
</div><!-- /.form-group -->
<hr>

<!-- USERNAME -->
<div class="form-group <?php echo (form_error('username_special') == '') ? ''
: 'has-error'; ?>">
    <label for="username_special">Username <span class="text-
info">*</span></label>
    <input type="text" class="form-control" name="username_special"
placeholder="secretary.law" value="<?php echo set_value('username_special'); ?>">
    <?php echo form_error('username_special'); ?>
</div>

<div class="form-group <?php echo (form_error('email_special') == '') ? ''
: 'has-error'; ?>">
    <label for="email_special">Email <span class="text-
info">*</span></label>
    <input type="text" class="form-control" name="email_special"
placeholder="secretary.law" value="<?php echo set_value('email_special'); ?>">
    <?php echo form_error('email_special'); ?>
</div>
<hr>

<div class="form-group <?php echo (form_error('unit') == '') ? '' : 'has-
error'; ?>">
    <label for="unit">Unit <span class="text-info">*</span></label>
    <select class="form-control select2-unit select2-hidden-
accessible" style="width: 100%;" tabindex="-1" aria-hidden="true" name="unit">
        <option></option>
        <?php foreach($unit_list as $key => $value){
            echo '<option value="'. $value['acronym']. ' '
            . $value['name']. ' '
            . $value['acronym']. ' "></option>';
        }
    </select>
    <?php set_select('unit', $value['acronym']);
    </div>

```

```

    }
    ?>
</select>
<?php echo form_error('unit'); ?>
</div>
<div class="form-group <?php echo (form_error('division') == '') ? '' :
'has-error'; ?>">
    <label for="division">Division <span class="text-
info">*</span></label>
    <select class="form-control select2-division select2-hidden-
accessible" style="width: 100%;" tabindex="-1" aria-hidden="true" name="division">
        <option></option>
        <?php foreach($division_list as $key => $value){
            echo '<option value="'. $value['acronym']. ' '
'.set_select('division', $value['acronym']).'>'. $value['unitname']. '
('.$value['acronym'].')</option>';
        }
        ?>
    </select>
    <?php echo form_error('division'); ?>
</div>
<div class="form-group <?php echo (form_error('designation') == '') ? '' :
'has-error'; ?>">
    <label for="designation">Designation <span class="text-
info">*</span></label>
    <select class="form-control select2-desig select2-hidden-
accessible" style="width: 100%;" tabindex="-1" aria-hidden="true"
name="designation">
        <option></option>
        <?php foreach($desig_list as $key => $value){
            echo '<option value="'. $value['designation']. ' '
'.set_select('designation',
$value['designation']).'>'. $value['designation']. '</option>';
        } ?>
    </select>
    <?php echo form_error('designation'); ?>
</div>
<div class="form-group <?php echo (form_error('mailquotasize') == '') ? ''
: 'has-error'; ?>">
    <label for="mailquotasize">Mail Quota Size</label>
    <select class="form-control select2-mailquota select2-hidden-
accessible" style="width: 100%;" tabindex="-1" aria-hidden="true"
name="mailquotasize">
        <option value="838860800" <?php echo
set_select('mailquotasize', '838860800', TRUE); ?> >800 MB (special
account)</option>
        <option value="419430400" <?php echo
set_select('mailquotasize', '419430400'); ?> >400 MB (employee)</option>
    </select>
    <?php echo form_error('mailquotasize'); ?>
</div>
<div class="form-group icheck">
    <label>
        <input type="checkbox" name="bypass" class="minimal" <?php if($this-
>input->post('bypass') === 'on') { echo "checked"; } ?>>
        Bypass special case validation.
    </label>
</div>
</div><!-- /.box-body -->

```

```

<div class="box-footer">
    <button type="submit" class="btn btn-primary">Submit</button>
</div>
<?php echo form_close(); ?>
</div><!-- /.box -->
</div>
</div>

```

application/views/account_user_create_confirm.php

```

<div class="row">
    <div class="col-xs-12">
        <div class="box box-primary">
            <div class="box-header with-border">
                <h3 class="box-title">Create Account > Review</h3>
            </div><!-- /.box-header -->
            <div class="box-body">
                <div class="table-responsive no-padding">
                    <h4>The following information will be
generated:</h4>
                    <table class="table table-hover">
                        <thead><tr>
                            <th>Item</th>
                            <th>Description</th>
                        </tr>
                        </thead>
                        <tbody>
                            <?php
                                foreach($account_gen as $key => $value) {
                                    if(!is_numeric($key)){
                                        if
(array_key_exists($key,$account_gen_fieldnames) == TRUE){
                                            echo '<tr><th
class="col-md-3">'.$account_gen_fieldnames[$key]."</th>";
                                            }
                                            else { echo '<tr><th
class="col-md-3">'.$key."</th>"; }
                                            echo '<td class="col-md-
9">'.$value.'"</td></tr>';
                                        }
                                    }
                                ?>
                            </tbody>
                        </table>
                    </div>
                </div><!-- /div.box-body -->
            <div class="box-body">
                <div class="table-responsive no-padding">
                    <h4><span class="text-danger">REVIEW</span> the
information below before creating this account:</h4>
                    <table class="table table-hover">
                        <thead><tr>
                            <th>Item</th>
                            <th>Description</th>
                        </tr>
                        </thead>
                        <tbody>

```

```

        <?php
        foreach($account as $key => $value) {
            if ($key == 'gender') { $value =
gender($value); }
            if(!is_numeric($key)){
                if
(array_key_exists($key,$account_fieldnames) == TRUE){
                    echo '<tr><th
class="col-md-3">'. $account_fieldnames[$key]. "</th>";
                }
                else { echo '<tr><th
class="col-md-3">'. $key. "</th>"; }
                    echo '<td class="col-md-
9">'. $value. '</td></tr>';
                }
            }
        }
    </tbody>
</table>
</div><!-- /div.box-body -->
<div class="box-footer">
<div class="row">
    <div class="col-xs-2 col-sm-2 col-md-3
col-lg-2">
        <a href="<?php echo
base_url('tech/create_user/activate'); ?>" class="btn btn-lg btn-primary"><i
class="fa fa-save"></i> Create Account</a>
    </div>
    <div class="col-xs-offset-7 col-sm-offset-
6 col-md-offset-4 col-lg-offset-3">
        <a href="<?php echo
base_url('tech/create_user'); ?>" class="btn btn-lg btn-default"><i class="fa fa-
arrow-left"></i> Go Back</a>
    </div>
</div>
</div>
</div><!-- /.box-primary -->
</div><!-- /div.col-xs-12 -->
</div>

```

application/views/account_user_create_css.php

```

<!-- iCheck -->
<link rel="stylesheet" href="<?php echo $this->config->item('theme_url');
?>/plugins/iCheck/square/blue.css">
<!-- Select2 -->
<link rel="stylesheet" href="<?php echo $this->config->item('theme_url');
?>/plugins/select2/select2.min.css">

```

application/views/account_user_create_js.php

```

<!-- InputMask -->
<script src="<?php echo $this->config->item('theme_url'); ?>/plugins/input-
mask/jquery.inputmask.js"></script>
<script src="<?php echo $this->config->item('theme_url'); ?>/plugins/input-
mask/jquery.inputmask.date.extensions.js"></script>

```

```

<script src="<?php echo $this->config->item('theme_url'); ?>/plugins/input-
mask/jquery.inputmask.extensions.js"></script>
<script src="<?php echo $this->config->item('theme_url');
?>/plugins/iCheck/ichack.min.js"></script>
<script src="<?php echo $this->config->item('theme_url');
?>/plugins/select2/select2.full.min.js"></script>

<script>
$('input[type=file]').change(function(){
    if($('input[type=file]').val()==''){
        $('button').attr('disabled',true)
    }
    else{
        $('button').attr('disabled',false);
    }
})
$(function(){
    $("#bd_mask").inputmask("yyyy-mm-dd", {"placeholder": "yyyy-mm-dd"});
    $("#sn_mask").inputmask("9999-99999", {"placeholder": "xxxx-xxxxx"});
    $("#id_mask").inputmask("9999999999", {"placeholder": "xxxxxxxxxxx"});

    $('input').iCheck({
        checkboxClass: 'icheckbox_square-blue',
        radioClass: 'iradio_square-blue',
        increaseArea: '20%' // optional
    });

    $('.select2-mailquota').select2(
    {
        tags: true,
        maximumSelectionLength: 2
    });

    $('.select2-desig').select2(
    {
        placeholder: "Select or type in designation. Ex: Adjunct Professor",
        tags: true,
        maximumSelectionLength: 2
    });

    $('.select2-unit').select2(
    {
        placeholder: "If unit doesn't exist, type in the acronym of the new unit.
Ex: Engg",
        tags: true,
        maximumSelectionLength: 2
    });

    $('.select2-division').select2(
    {
        placeholder: "If division doesn't exist, type in the acronym of the new
division. Ex: EEEI",
        tags: true,
        maximumSelectionLength: 2
    });

    $('#ln_none').on('ifChecked', function () {
    $('#lastname').prop("disabled", true);
    $('#lastname').removeAttr('value');
    });

```



```

$('#ln_none').on('ifUnchecked', function () {
    $('#lastname').prop("disabled", false);
});
$('#mn_none').on('ifChecked', function () {
    $('#middlename').prop("disabled", true);
    $('#middlename').removeAttr('value');
});
$('#mn_none').on('ifUnchecked', function () {
    $('#middlename').prop("disabled", false);
});
});
</script>

```

application/views/account_user_create.php

```

<div class="row">
    <div class="col-xs-12">
        <div class="box box-primary">
            <div class="box-header with-border">
                <h3 class="box-title"><i class="fa fa-user"></i> Create Employee
Account</h3>
                <br>
                <h5> Fields with <strong><span class="text-info">*</span></strong> are
required. <br>Need to create multiple accounts? Upload CSV file using the form
below. Format:
lastname,firstname,middlename,unit,division,designation,type,employeeeno
                <br><br><?php echo
form_open_multipart('create_multiple_employees/do_upload');?>
                <input type="file" class="btn btn-xs btn-default btn-flat"
name="userfile" id='userfile' size="20" id="file" style="display: inline;">
                <button type="submit" class="btn btn-xs btn-primary" disabled>Upload
and Process Multiple Accounts</button>
                <?php echo form_close(); ?>
            </h5>
        </div><!-- /.box-header -->
        <!-- form start -->
        <?php echo form_open('tech/create_user'); ?>
        <div class="box-body">

            <!-- NAME -->
            <div class="form-group row">
                <div class="col-md-4 col-xs-12 <?php echo (form_error('firstname') ==
'' ) ? '' : 'has-error form-group'; ?>">
                    <label for="name">First Name <span class="text-info">*</span></label>
                    <input type="text" class="form-control" name="firstname" value="<?php
echo set_value('firstname'); ?>" placeholder="Juan">
                    <?php echo form_error('firstname'); ?>
                </div>
                <div class="col-md-3 col-xs-12 <?php echo (form_error('middlename') ==
'' ) ? '' : 'has-error form-group'; ?>">
                    <label for="name">Middle Name <span class="text-info">*</span></label>
                    <input type="text" class="form-control" name="middlename"
id="middlename" placeholder="Santos" value="<?php echo set_value('middlename');
?>">
                    <?php echo form_error('middlename'); ?>
                </div>
                <div class="col-md-3 col-xs-12 <?php echo (form_error('lastname') == '' )
? '' : 'has-error form-group'; ?>">

```

```

        <label for="name">Last Name <span class="text-info">*</span></label>
        <input type="text" class="form-control" id="Lastname" name="lastname"
placeholder="de la Cruz" value="<?php echo set_value('lastname'); ?>"
        <?php echo form_error('lastname'); ?>
    </div><!-- /.row -->
    <div class="col-md-2 col-xs-12" style="margin-top: 5px;">
        <div><label><input type="checkbox" id="mn_none" name="mn_none"
class="minimal" <?php if($this->input->post('mn_none') === 'on') { echo "checked";
} ?>> No Middle Name</label></div>
        <div><label> <input type="checkbox" id="ln_none" name="ln_none"
class="minimal" <?php if($this->input->post('ln_none') === 'on') { echo "checked";
} ?>> No Last Name</label></div>
    </div>
</div>
<hr>

```

```

    <div class="form-group row">
    <!-- EMPLOYEE TYPE -->
    <div class="col-md-4 col-sm-12 <?php echo (form_error('type') == '') ?
'' : 'has-error form-group'; ?>">
        <label>Account Type <span class="text-info">*</span></label>
        <div>
            <label class="checkbox-inline">
                <input type="radio" name="type" value="FAC" class="minimal" <?php
if($this->input->post('type') === 'FAC') { echo "checked"; } ?>>
                Faculty
            </label>
            <label class="checkbox-inline">
                <input type="radio" name="type" value="REPS" class="minimal" <?php
if($this->input->post('type') === 'REPS') { echo "checked"; } ?>>
                REPS
            </label>
            <label class="checkbox-inline">
                <input type="radio" name="type" value="ADM" class="minimal" <?php
if($this->input->post('type') === 'ADM') { echo "checked"; } ?>>
                Admin
            </label>
        </div>
        <?php echo form_error('type'); ?>
    </div>

```

```

    <!-- GENDER -->
    <div class="col-md-3 col-sm-12 <?php echo (form_error('gender') == '') ?
'' : 'has-error form-group'; ?>">
        <label>Gender <span class="text-info">*</span></label>
        <div>
            <label class="checkbox-inline">
                <input type="radio" name="gender" value="0" class="minimal" <?php
if($this->input->post('gender') === '0') { echo "checked"; } ?>>
                <i class="fa fa-male"></i> Male
            </label>
            <label class="checkbox-inline">
                <input type="radio" name="gender" value="1" class="minimal" <?php
if($this->input->post('gender') === '1') { echo "checked"; } ?>>
                <i class="fa fa-female"></i> Female
            </label>
        </div>
        <?php echo form_error('gender'); ?>
    </div>

```

```
</div><!-- /.col-md-6 col-sm-12 -->
```

```
<!-- Birthdate -->
```

```
<div class="col-md-3 col-sm-12 <?php echo (form_error('birthdate') ==  
'') ? '' : 'has-error form-group'; ?>">
```

```
<label>Birthdate</label>
```

```
<div class="input-group">
```

```
<div class="input-group-addon">
```

```
<i class="fa fa-calendar"></i>
```

```
</div>
```

```
<input type="text" id="bd_mask" class="form-control" data-
```

```
inputmask="'alias': 'yyyy-mm-dd'" data-mask placeholder="yyyy-mm-dd"
```

```
name="birthdate" value="<?php echo set_value('birthdate'); ?>">
```

```
</div><!-- /.input group -->
```

```
<?php echo form_error('birthdate'); ?>
```

```
</div><!-- /.col-md-6 col-sm-12 -->
```

```
</div><!-- /.form-group -->
```

```
<hr>
```

```
<!-- EMPLOYEE NUMBER -->
```

```
<div class="form-group <?php echo (form_error('id_number') == '') ? '' :
```

```
'has-error'; ?>">
```

```
<label for="empno">Employee Number</label>
```

```
<input type="text" class="form-control" id="id_mask" data-
```

```
inputmask="'mask': '0999999999'" data-mask maxlength="10" name="id_number"
```

```
placeholder="0123456789" value="<?php echo set_value('id_number'); ?>">
```

```
<?php echo form_error('id_number'); ?>
```

```
</div>
```

```
<div class="form-group <?php echo (form_error('studentno') == '') ? '' :
```

```
'has-error'; ?>">
```

```
<label for="studentno">Student Number</label>
```

```
<input type="text" class="form-control" id="sn_mask" data-
```

```
inputmask="'mask': '9999-9999'" data-mask maxlength="10" name="studentno"
```

```
placeholder="2008-12345" value="<?php echo set_value('studentno'); ?>">
```

```
<?php echo form_error('studentno'); ?>
```

```
</div>
```

```
<hr>
```

```
<div class="form-group <?php echo (form_error('unit') == '') ? '' : 'has-
```

```
error'; ?>">
```

```
<label for="unit">Unit <span class="text-info">*</span></label>
```

```
<select class="form-control select2-unit select2-hidden-
```

```
accessible" style="width: 100%;" tabindex="-1" aria-hidden="true" name="unit">
```

```
<option></option></select>
```

```
<?php foreach($unit_list as $key => $value){
```

```
<?php echo 'option value="' . $value['acronym'] . '"
```

```
<?php echo 'option value="' . $value['name'] . '"
```

```
<?php echo 'option value="' . $value['name'] . '"
```

```
<?php echo form_error('unit'); ?>
```

```
</div>
```

```
<div class="form-group <?php echo (form_error('division') == '') ? '' :
```

```
'has-error'; ?>">
```

```
<label for="division">Division <span class="text-
```

```
info">*</span></label>
```

```
<select class="form-control select2-division select2-hidden-
```

```
accessible" style="width: 100%;" tabindex="-1" aria-hidden="true" name="division">
```

```

        <option></option>
        <?php foreach($division_list as $key => $value){
            echo ' <option value="'. $value['acronym']. ' "'
'.set_select('division', $value['acronym']).'>'. $value['unitname']. '
('.$value['acronym'].')</option>';
        }
        ?>
    </select>
    <?php echo form_error('division'); ?>
</div>
<div class="form-group <?php echo (form_error('designation') == '') ? '' :
'has-error'; ?>">
    <label for="designation">Designation <span class="text-
info">*</span></label>
    <select class="form-control select2-desig select2-hidden-
accessible" style="width: 100%;" tabindex="-1" aria-hidden="true"
name="designation">
        <option></option>
        <?php foreach($desig_list as $key => $value){
            echo ' <option value="'. $value['designation']. ' "'
'.set_select('designation',
$value['designation']).'>'. $value['designation']. </option>';
        } ?>
    </select>
    <?php echo form_error('designation'); ?>
</div>
<div class="form-group <?php echo (form_error('description') == '') ? '' :
'has-error'; ?>">
    <label for="description">Appointment Period</label>
    <input type="text" class="form-control" name="description"
placeholder="01 Sep 2017 - 31 Jul 2018">
    <?php echo form_error('description'); ?>
</div>
<div class="form-group <?php echo (form_error('mailquotasize') == '') ? ''
: 'has-error'; ?>">
    <label for="mailquotasize">Mail Quota Size</label>
    <select class="form-control select2-mailquota select2-hidden-
accessible" style="width: 100%;" tabindex="-1" aria-hidden="true"
name="mailquotasize">
        <option value="838860800" <?php echo
set_select('mailquotasize', '838860800'); ?> >800 MB (special account)</option>
        <option value="419430400" <?php echo
set_select('mailquotasize', '419430400', TRUE); ?> >400 MB (employee)</option>
    </select>
    <?php echo form_error('mailquotasize'); ?>
</div>
<!--
<div class="form-group <?php echo (form_error('mailquotasize') == '') ? ''
: 'has-error'; ?>">
    <label for="mailquotasize">Mail Quota Size</label>
    <select class="form-control select2-mailquota select2-hidden-
accessible" style="width: 100%;" tabindex="-1" aria-hidden="true"
name="mailquotasize">
        <?php
        $options = array(
            838860800 => '800 MB (special)',
            419430400 => '400 MB (employee)'
        );

```

```

        ?>
        <option></option>
        <?php foreach($options as $key => $option){
            echo 'option value='.$key.'>'.$option.'</option>';
        }?>
    </select>
    <?php echo form_error('mailquotasize'); ?>
</div>
-->
<div class="form-group icheck">
    <label>
        <input type="checkbox" name="bypass" class="minimal" <?php if($this->input->post('bypass') === 'on') { echo "checked"; } ?>>
        Bypass special case validation.
    </label>
</div>
</div><!-- /.box-body -->

<div class="box-footer">
    <button type="submit" class="btn btn-primary">Submit</button>
</div>
<?php echo form_close(); ?>
</div><!-- /.box -->
</div>
</div>

```

application/views/account_user_edit_confirm.php

```

<div class="box box-solid">
    <div class="box-body">
        <?php if(!empty($info_diff)): ?>
            <h3> Are you sure you want to change the following
            attribute(s)? </h3>
            <table class="table table-bordered table-striped">
                <thead>
                    <tr>
                        <th>Attribute</th>
                        <th>Old Value</th>
                        <th>New Value</th>
                    </tr>
                </thead>
                <?php foreach($info_diff as $key =>
                    $value): ?>
                    <?php
                        if ($key == 'gender') {
                            $value = gender($value);
                            $info_orig[$key] =
                                gender($info_orig[$key]);
                        }
                    <?>
                    <tr>
                        <td><?php echo $key; ?></td>
                        <td><?php
                            if($info_orig[$key] == '' ||
                                echo "NULL";
                            } else {

```

```

        if(is_array($info_orig[$key])){
            $tvalue =
            implode("<br>", $info_orig[$key]);
            echo $tvalue;
        } else {
            echo
            $info_orig[$key];
        }
    } ?></td>
    <td><?php
        if($value == '' || $value ==
        NULL) {
            echo "NULL";
        } else {
            if(is_array($value)){
                $tvalue =
                implode("<br>", $value);
                print_r($tvalue);
            } else {
                echo $value;
            }
        } ?></td>
    </tr>
    <?php endforeach; ?>
    <?php endif; ?>
</table>
<div class="box-footer" style="margin-top: 10px;">
    <?php echo
    form_open('tech/dilnetacc_tools/edit_account_final'); ?>
    <?php echo form_hidden($info_diff); ?>
    <input type="hidden" class="form-control" name="uid"
    value="<?php echo $uid; ?>">
    <div class="row">
        <div class="col-xs-5 col-sm-4 col-md-4 col-lg-2">
            <button type="submit" class="btn btn-lg btn-
            primary"><i class="fa fa-save"></i> Save</button>
            <div class="col-xs-offset-7 col-sm-offset-6 col-md-offset-4
            col-lg-offset-3">
                <a class="btn btn-lg btn-default" href="<?php echo
                site_url('tech/dilnetacc_tools/edit_account_cancel/' . $uid); ?>" data-
                toggle="tooltip" data-placement="auto" data-html="true" title="Go Back"><i
                class="fa fa-arrow-left"></i> Go Back</a>
            </div>
        </div>
        <?php echo form_close(); ?>
    </div>
</div>
</div>

```

```

application/views/account_user_edit_js.php

```

```

<!-- InputMask -->
<script src="<?php echo $this->config->item('theme_url'); ?>/plugins/input-
mask/jquery.inputmask.js"></script>

```

```

<script src="<?php echo $this->config->item('theme_url'); ?>/plugins/input-
mask/jquery.inputmask.date.extensions.js"></script>
<script src="<?php echo $this->config->item('theme_url'); ?>/plugins/input-
mask/jquery.inputmask.extensions.js"></script>
<script src="<?php echo $this->config->item('theme_url');
?>/plugins/iCheck/ichack.min.js"></script>
<script src="<?php echo $this->config->item('theme_url');
?>/plugins/select2/select2.full.min.js"></script>

<script>
$(function(){
    $("#bd2_mask").inputmask("yyyy-mm-dd", {"placeholder": "yyyy-mm-dd"});
    $("#sn_mask").inputmask("9999-99999", {"placeholder": "xxxx-xxxxx"});
    // $("#id_mask").inputmask("9999999999", {"placeholder": "xxxxxxxxxxx"});

    $('input').iCheck({
        checkboxClass: 'icheckbox_square-blue',
        radioClass: 'iradio_square-blue',
        increaseArea: '20%' // optional
    });

    $('.select2-desig').select2(
    {
        placeholder: "Select or type in designation. Ex: Adjunct Professor",
        tags: true,
        maximumSelectionLength: 2
    });

    $('.select2-unit').select2(
    {
        placeholder: "If unit doesn't exist, type in the ACRONYM of the new unit",
        tags: true,
        maximumSelectionLength: 2
    });
    $('.select2-division').select2(
    {
        placeholder: "If division doesn't exist, type in the ACRONYM of the new
division",
        tags: true,
        maximumSelectionLength: 2
    });
    $('.select2-mailquotasize').select2(
    {
        placeholder: "Choose mail quota size from list",
        maximumSelectionLength: 2
    });
});
</script>

```

```

application/views/account_user_edit.php

```

```

<?php $uid = $this->input->post('uid'); ?>
<div class="box box-solid">
    <div class="box-body">
        <div style="margin-top: 10px;"><?php echo "".strtoupper($this->input-
        <h2 style="margin-top: 10px;"><?php echo $this->input->post('lastname')).""; ?></h2>
        <h2 style="margin-top: 10px;"><?php echo $this->input->post('firstname')).""; ?></h2>
        <h4><?php echo strtoupper($uid); ?></h4>
        <h4><?php echo strtoupper($uid); ?></h4>
    </div><!-- /.box-body -->

```

```

    <?php $this->load->view('partials/user_footer_actions'); ?>
</div><!-- div.box -->
<div class="row">
    <div class="col-xs-12">
        <div class="box box-solid">
            <div class="box-body">
                <br><h5> NOTE: Fields with <strong><span class="text-
info">*</span></strong> are required.</h5>
                <div class="table-responsive no-padding">
                    <?php echo form_open('tech/dilnetacc_tools/edit_account_form'); ?>
                    <h3>Account Information</h3>
                    <table class="table table-hover">
                        <tr>
                            <th class="col-md-3">Username <strong><span class="text-
info">*</span></strong>
                            <?php if($this->acl->hasRole('supertech')): ?>
                                <br><small class="bg-primary">SUPERTECH ACCESS ONLY</small>
                            <?php endif; ?>
                            </th>

                            <td class="col-md-9 <?php echo (form_error('username') == '') ? '' :
'text-danger'; ?>">
                                <?php if($this->acl->hasRole('supertech')): ?>
                                    <input type="text" class="form-control" name="username"
placeholder="username" value="<?php echo set_value('username', $uid); ?>">
                                <?php else: ?>
                                    <input type="text" class="form-control" name="username"
placeholder="username" value="<?php echo set_value('username', $uid); ?>"
disabled>
                                <input type="hidden" class="form-control" name="username"
placeholder="username" value="<?php echo $uid; ?>">
                                <?php endif; ?>
                                <?php echo form_error('username'); ?>
                            </td>
                        </tr>
                        <tr>
                            <th class="col-md-3">Primary Email Address <strong><span
class="text-info">*</span></strong>
                            <?php if($this->acl->hasRole('supertech')): ?>
                                <br><small class="bg-primary">SUPERTECH ACCESS ONLY</small>
                            <?php endif; ?>
                            </th>
                            <td class="col-md-9 <?php echo (form_error('mail') == '') ? '' :
'has-error text-danger'; ?>">
                                <?php if($this->acl->hasRole('supertech')): ?>
                                    <input type="text" class="form-control " name="mail"
placeholder="primary email address" value="<?php echo set_value('mail',
$other_info['mail']); ?>">
                                <?php else: ?>
                                    <input type="text" class="form-control " name="mail"
placeholder="primary email address" value="<?php echo set_value('mail',
$other_info['mail']); ?>" disabled>
                                <input type="hidden" class="form-control " name="mail"
placeholder="primary email address" value="<?php echo $other_info['mail']; ?>">
                                <?php endif; ?>
                                <?php echo form_error('mail'); ?>
                            </td>
                        </tr>
                    </table>
                </div>
            </div>
        </div>
    </div>
</tr>
</div>

```



```

        <?php if(sizeof($other_info['mailalternateaddress'])>1){ ?>
            <th class="col-md-3">Alternate Email Addresses <strong><span
class="text-info">*</span></strong>
            <?php if($this->acl->hasRole('supertech')): ?>
            <br><small class="bg-primary">SUPERTech ACCESS ONLY</small>
            <?php endif; ?>
        </th>
        <?php } else { ?>
            <th class="col-md-3">Alternate Email Address <strong><span
class="text-info">*</span></strong>
            <?php if($this->acl->hasRole('supertech')): ?>
            <br><small class="bg-primary">SUPERTech ACCESS ONLY</small>
            <?php endif; ?>
        </th>
        <?php }; ?>
        <td class="col-md-9 <?php echo (form_error('mailalternateaddress')
== '') ? '' : 'has-error text-danger'; ?>">
            <?php
            if($this->acl->hasRole('supertech')) {
                $data = array(
                    'name'          => 'mailalternateaddress',
                    'id'            => 'mailalternateaddress',
                    'rows'          => '5',
                    'class'         => 'form-control'
                );
            } else {
                $data = array(
                    'name'          => 'mailalternateaddress',
                    'id'            => 'mailalternateaddress',
                    'rows'          => '5',
                    'class'         => 'form-control',
                    'readonly'     => 'readonly'
                );
            }
            if(is_array($other_info['mailalternateaddress'])) {
                $alt_addresses =
implode("\n",$other_info['mailalternateaddress']);
                $set_value = set_value('mailalternateaddress');
                if(is_array($set_value)){
                    $data['value'] =
implode("\n",set_value('mailalternateaddress'));
                } else {
                    $data['value'] =
set_value('mailalternateaddress',$alt_addresses);
                }
            } else {
                $data['value'] =
set_value('mailalternateaddress',$other_info['mailalternateaddress']);
            }
            echo form_textarea($data);
            echo form_error('mailalternateaddress'); ?>
        </td>
    </tr>
</table>
<h3 style="margin-top: 5px;">User Information</h3>
<table class="table table-hover">
    <tr>
        <th class="col-md-3">Full Name

```



```

        Female
    </label>
    <label class="checkbox-inline">
        <input type="radio" name="gender" value="0" class="minimal"
<?php if($this->input->post('gender') === '0') { echo "checked"; } ?>>
        Male
    </label>
</div>
<?php echo form_error('gender'); ?>
</td>
</tr>
<tr>
    <th class="col-md-3">Birthdate <strong><span class="text-
info">*</span></strong></th>
    <td class="col-md-9 <?php echo (form_error('birthdate') == '') ? '' :
: 'text-danger'; ?>">
        <div class="input-group">
            <div class="input-group-addon">
                <i class="fa fa-calendar"></i>
            </div>
            <input type="text" id="bd2_mask" class="form-control" data-
inputmask="'alias': 'yyyy-mm-dd'" data-mask placeholder="yyyy-mm-dd"
name="birthdate" value="<?php echo set_value('birthdate'); ?>">
        </div><!-- /.input group -->
        <?php echo form_error('birthdate'); ?>
    </td>
</tr>
<tr>
    <th class="col-md-3">Unit / College <strong><span class="text-
info">*</span></strong></th>
    <td class="col-md-9 <?php echo (form_error('unit') == '') ? '' :
'text-danger'; ?>">
        <select class="form-control select2-unit select2-hidden-
accessible" style="width: 100%;" tabindex="-1" aria-hidden="true" name="unit">
            <?php if(isset($_POST['unit'])): ?>
                <option value="<?php echo $this->input->post('unit');
?>"><?php echo $this->input->post('unit'); ?></option>
            <?php else: ?>
                <option></option>
            <?php endforeach($unit_list as $key => $value){
                echo '<option value="' . $value['acronym'] . ' "
                . $value['acronym'] . ' ">' . $value['name'] . ' '
                . $value['acronym'] . ' "></option>';
            }
            ?>
        </select>
        <?php echo form_error('unit'); ?>
    </td>
</tr>
<tr>
    <th class="col-md-3">Division / Department <strong><span
class="text-info">*</span></strong></th>
    <td class="col-md-9 <?php echo (form_error('division') == '') ? '' :
'text-danger'; ?>">
        <select class="form-control select2-division select2-hidden-
accessible" style="width: 100%;" tabindex="-1" aria-hidden="true" name="division">
            <?php if(isset($_POST['division'])): ?>

```

```

        <option value="<?php echo $_POST['division']; ?>"><?php echo
$_POST['division']; ?></option>
        <?php else: ?>
            <option></option>
        <?php endif; ?>
        <?php foreach($division_list as $key => $value){
            echo '<option value="'. $value['acronym']. ' "'
            '.set_select('division', $value['acronym']).'>'. $value['unitname']. '
('.$value['acronym'].')</option>';
        }
        ?>
    </select>
    <?php echo form_error('division'); ?>
</td>
</tr>
<tr>
    <th class="col-md-3">Designation <strong><span class="text-
info">*</span></strong></th>
    <td class="col-md-9 <?php echo (form_error('designation') == '') ?
'' : 'text-danger'; ?>">
        <select class="form-control select2-desig select2-hidden-
accessible" style="width: 100%;" tabindex="-1" aria-hidden="true"
name="designation">
            <?php if(isset($_POST['designation'])): ?>
                <option value="<?php echo $_POST['designation']; ?>"><?php
echo $_POST['designation']; ?></option>
            <?php else: ?>
                <option></option>
            <?php endif; ?>
            <?php foreach($desig_list as $key => $value){
                echo '<option value="'. $value['designation']. ' "'
                '.set_select('designation',
$value['designation']).'>'. $value['designation']. '</option>';
            }
            ?>
        </select>
        <?php echo form_error('designation'); ?>
    </td>
</tr>
<tr>
    <th class="col-md-3">Student Number</th>
    <td class="col-md-9 <?php echo (form_error('studno') == '') ? '' :
'text-danger'; ?>">
        <input type="text" class="form-control" id="sn_mask" data-
inputmask="'mask': '9999-9999'" data-mask maxlength="10" name="studno"
placeholder="2008-12345" value="<?php echo set_value('studno'); ?>"
        <?php echo form_error('studno'); ?>
    </td>
</tr>
<tr>
    <th class="col-md-3">Employee Number</th>
    <td class="col-md-9 <?php echo (form_error('empno') == '') ? '' :
'text-danger'; ?>">
        <input type="text" class="form-control" id="id_mask" data-
inputmask="'mask': '0999999999'" data-mask maxlength="10" name="empno"
placeholder="(0)123456789" value="<?php echo set_value('empno'); ?>"
        <?php echo form_error('empno'); ?>
    </td>
</tr>

```

```
|  |  |  |  | | | |
|---|---|---|---|---|---|---|
| Employee Type <strong><span class="text- info">*</span></strong></th>  <label class="checkbox-inline"> <input type="radio" name="type" value="FAC" class="minimal" <?php if($this->input->post('type') === 'FAC') { echo "checked"; } ?> Faculty </label> <label class="checkbox-inline"> <input type="radio" name="type" value="REPS" class="minimal" <?php if($this->input->post('type') === 'REPS') { echo "checked"; } ?> REPS </label> <label class="checkbox-inline"> <input type="radio" name="type" value="ADM" class="minimal" <?php if($this->input->post('type') === 'ADM') { echo "checked"; } ?> Admin </label> <label class="checkbox-inline"> <input type="radio" name="type" value="OFC" class="minimal" <?php if($this->input->post('type') === 'OFC') { echo "checked"; } ?> Office/Unit </label> <label class="checkbox-inline"> <input type="radio" name="type" value="PROJ" class="minimal" <?php if($this->input->post('type') === 'PROJ') { echo "checked"; } ?> Project/Program </label> <label class="checkbox-inline"> <input type="radio" name="type" value="STUD" class="minimal" <?php if($this->input->post('type') === NULL || $this->input->post('type') === 'STUD') { echo "checked"; } ?> Student </label> </div> <?php echo form_error('type'); ?> </td> </tr> | Appointment Period</th>   |placeholder="1 Jan 2016 - 31 Jul 2016" value="<?php echo set_value('description'); ?>"/>   appointment.</p> <?php echo form_error('description'); ?> </td> </tr> </table> <h3>Mail Usage</h3> <table class="table table-hover"> <tr> | | |

```

```

        <th class="col-md-3">Mail Forwarding Address <br><small class="bg-
primary">SUPERTECH ACCESS ONLY</small></th>
        <td class="col-md-9 <?php echo (form_error('mailforward') == '') ?
'' : 'has-error text-danger'; ?>">
            <input type="text" class="form-control <?php echo
(form_error('mailforward') == '') ? '' : 'text-danger'; ?>" name="mailforward"
placeholder="emailaddress@gmail.com" value="<?php echo set_value('mailforward');
?>">
            <?php echo form_error('mailforward'); ?>
        </td>
    </tr>
    <?php endif; ?>
    <tr>
        <th class="col-md-3">Mail Quota Size</th>
        <td class="col-md-9 <?php echo (form_error('mailquotasize') == '')
? '' : 'text-danger'; ?>">
            <select class="form-control select2-mailquota" style="width:
100%;" tabindex="-1" aria-hidden="true" name="mailquotasize">
                <option value="838860800" <?php echo
set_select('mailquotasize', '838860800'); ?> >800 MB (special account)</option>
                <option value="419430400" <?php echo
set_select('mailquotasize', '419430400'); ?> >400 MB (employee)</option>
                <option value="209715200" <?php echo
set_select('mailquotasize', '209715200'); ?> >200 MB (student)</option>
            </select>
            <?php echo form_error('mailquotasize'); ?>
        </td>
    </tr>
</table>
</table>
<input type="hidden" class="form-control" name="uid" value="<?php echo
set_value('uid',$uid); ?>">

</div><!-- /div.box-body -->
<div class="box-footer">
    <button type="submit" class="btn btn-primary">Submit</button>
</div>
<?php echo form_close(); ?>
</div><!-- /.box-primary -->
</div><!-- /div.col-xs-12 -->
</div>

```

```

application/views/account_user_view_css.php

```

```

<style>
.dataTables_wrapper .dataTables_filter {
float: right;
text-align: left;
}
</style>

```

```

application/views/account_user_view_js.php

```

```

<!-- DataTables -->

```

```

<script src="<?php echo $this->config->item('theme_url');
?>/plugins/datatables/jquery.dataTables.min.js"></script>
<script src="<?php echo $this->config->item('theme_url');
?>/plugins/datatables/dataTables.bootstrap.min.js"></script>

<!-- page script -->
<script>
$(function () {
    $("#accountlog_results").DataTable();
    $("#changepass_results").DataTable();
    $("#lostpass_results").DataTable();
    $("#mail_search_results").DataTable();
});
var timeout = 5000; // in milliseconds (3*1000)
$('.callout').delay(timeout).fadeOut(300);

function changeText(){
    if ($('#mail_search_str').val().length >= 3) {
        // value is empty or is default value
        // value is empty or is default value
        $("#submitbutton").text("Searching ... Please wait ...");
        $("#submitbutton").removeClass('btn-info').addClass('btn-warning');
    }
};

function changeMailBlockSearchText(){
    if ($('#mail_block_search_str').val().length >= 3) {
        // value is empty or is default value
        // value is empty or is default value
        $("#mailBlockSubmitButton").text("Searching ... Please wait ...");
        $("#mailBlockSubmitButton").removeClass('btn-info').addClass('btn-
warning');
    }
};

function changeHostBlockSearchText(){
    if ($('#host_block_search_str').val().length >= 3) {
        // value is empty or is default value
        // value is empty or is default value
        $("#hostBlockSubmitButton").text("Searching ... Please wait ...");
        $("#hostBlockSubmitButton").removeClass('btn-info').addClass('btn-
warning');
    }
};

function changeUserBlockSearchText(){
    if ($('#user_block_search_str').val().length >= 3) {
        // value is empty or is default value
        // value is empty or is default value
        $("#userBlockSubmitButton").text("Searching ... Please wait ...");
        $("#userBlockSubmitButton").removeClass('btn-info').addClass('btn-
warning');
    }
};
</script>

```

```

application/views/account_user_view.php

```

```

<div class="box box-solid <?php echo $basic_view['disabled'] ? 'bg-red' : '' ?>">
Integrated Information Systems for Enhanced Accounts Management 435

```

```

<div class="box-body">
    <h2 style="margin-top: 10px;"><?php echo strtoupper($ldap_view['Full Name']);
?></h2>
    <h4><?php echo strtoupper($ldap_view['Username']); ?></h4>
</div><!-- /.box-body -->
<?php $this->load->view('partials/user_footer_actions'); ?>
</div><!-- div.box -->
<?php // print "<pre>"; print_r($_SESSION); print "</pre>"; ?>
<div class="row">
    <div class="col-xs-12">
        <div class="nav-tabs-custom">
            <ul class="nav nav-tabs">
                <li class="active"><a href="#userview" data-toggle="tab" aria-
expanded="true">Basic View</a></li>
                <li class=""><a href="#advancedview" data-toggle="tab" aria-
expanded="false">Advanced View</a></li>
                <li class=""><a href="#accountlogs" data-toggle="tab" aria-
expanded="false">Account Log</a></li>
                <li class=""><a href="#changepass" data-toggle="tab" aria-
expanded="false">Changepass</a></li>
                <li class=""><a href="#lostpass" data-toggle="tab" aria-
expanded="false">Lostpass</a></li>
            </ul>
            <!-- Basic View #userview -->
            <div class="tab-content">
                <div class="tab-pane active" id="userview">
                    <div class="table-responsive no-padding style="overflow: auto;">
                        <h3>Account Information</h3>
                        <table class="table table-hover">
                            <tr>
                                <th class="col-md-3">Username</th>
                                <td class="col-md-9"><?php echo $basic_view['username']; ?></td>
                            </tr>
                            <tr>
                                <th class="col-md-3">Primary Email Address</th>
                                <td class="col-md-9"><?php echo $basic_view['mail']; ?></td>
                            </tr>
                            <tr>
                                <?php if(sizeof($basic_view['mailalternateaddress'])>2){
?>
                                <th class="col-md-3">Alternate Email Addresses</th>
                                <?php } else { ?>
                                <th class="col-md-3">Alternate Email Address</th>
                                <?php }; ?>
                                <td class="col-md-9"><?php
if(is_array($basic_view['mailalternateaddress']))){
foreach(array_slice($basic_view['mailalternateaddress'], 0) as
$a) { echo $a."<br>"; }
}
else { echo $basic_view['mailalternateaddress']; }
?>
                                </td>
                            </tr>
                        </table>
                        <h3>User Information</h3>
                        <table class="table table-hover">
                            <tr>
                                <th class="col-md-3">Last, First [Middle]</th>

```



```

        <td class="col-md-9"><?php echo ucwords($basic_view['lastname'])."
, ".ucwords($basic_view['firstname'])." [ ".ucwords($basic_view['middlename'])." ]
"; ?></td>
    </tr>
    <tr>
        <th class="col-md-3">Unit / College </th>
        <td class="col-md-9"><?php echo $basic_view['unit']; ?></td>
    </tr>
    <tr>
        <th class="col-md-3">Division / Department</th>
        <td class="col-md-9"><?php echo $basic_view['division']; ?></td>
    </tr>
    <tr>
        <th class="col-md-3">Employee Number</th>
        <td class="col-md-9"><?php echo $basic_view['empno']; ?></td>
    </tr>
    <tr>
        <th class="col-md-3">Employee Type</th>
        <td class="col-md-9"><?php echo $basic_view['type']=='<strong>No
data</strong>' ? 'STUD' : $basic_view['type']; ?></td>
    </tr>
    <tr>
        <th class="col-md-3">Appointment Period</th>
        <td class="col-md-9"><?php echo $basic_view['description'];
?></td>
    </tr>
    <tr>
        <th class="col-md-3">Student Number</th>
        <td class="col-md-9"><?php echo $basic_view['studno']; ?></td>
    </tr>
    <tr>
        <th class="col-md-3">Birthdate</th>
        <td class="col-md-9"><?php echo $basic_view['birthdate']; ?></td>
    </tr>
    <tr>
        <th class="col-md-3">Gender</th>
        <td class="col-md-9"><?php echo ($basic_view['gender']) ? "Female"
: "Male"; ?></td>
    </tr>
</table>
<h3>Account Status</h3>
<table class="table table-hover">
    <tr>
        <th class="col-md-3">Last login (Webmail)</th>
        <td class="col-md-9"><?php echo $basic_view['webmail_last_login'];
?></td>
    </tr>
    <tr>
        <th class="col-md-3">Last login (Accounts.upd)</th>
        <td class="col-md-9"><?php echo
$basic_view['accounts_last_login']; ?></td>
    </tr>
    <tr>
        <th class="col-md-3">Status</th>
        <td class="col-md-9"><?php echo ($basic_view['disabled']) ? "<div
class='label label-danger'>ACCOUNT DISABLED</div>" : "<div class='label label-
success'>ACTIVE</div>"; ?></td>
    </tr>
    <?php if($basic_view['disabled']): ?>

```

```

        <tr>
            <th class="col-md-3">Reason for Deactivation</th>
            <td class="col-md-9"><?php echo $basic_view['disabled_remarks'];
?></td>

        </tr>
        <?php endif; ?>
        <tr>
            <th class="col-md-3">Date Activated</th>
            <td class="col-md-9"><?php echo $basic_view['date_activated'].
("$basic_view['activated_by']."); ?></td>
        </tr>
    </table>
    <h3>Mail Usage</h3>
    <table class="table table-hover">
        <tr>
            <th class="col-md-3">Mail Host</th>
            <td class="col-md-9"><?php echo $basic_view['mailhost']; ?></td>
        </tr>
        <tr>
            <th class="col-md-3">Mail Forwarding Address</th>
            <td class="col-md-9"><?php echo $basic_view['mailforward'];
?></td>

        </tr>
        <tr>
            <th class="col-md-3">Mail Quota</th>
            <td class="col-md-9"><?php echo $basic_view['mailquotasize'].
"MB"; ?></td>

        </tr>
    </table>
    <h3>Password Recovery</h3>
    <table class="table table-hover">
        <tr>
            <th class="col-md-3">Security Last Updated</th>
            <td class="col-md-9"><?php echo
$basic_view['security_lastupdated']; ?></td>
        </tr>
        <tr>
            <th class="col-md-3">Recovery Email</th>
            <td class="col-md-9"><?php echo $basic_view['recovery_email'];
?></td>

        </tr>
        <tr>
            <th class="col-md-3">Security Question Type</th>
            <td class="col-md-9"><?php echo
$basic_view['security_question_type']; ?></td>
        </tr>
        <tr>
            <th class="col-md-3">Recovery Feature Status</th>
            <td class="col-md-9"><?php echo
$basic_view['recovery_feature_status']; ?></td>
        </tr>
        <tr>
            <th class="col-md-3">Recovery Count</th>
            <td class="col-md-9"><?php echo $basic_view['recovery_count'];
?></td>

        </tr>
    </table>
</div> <!-- /div.table-responsive -->

```

```

</div> <!-- /div.tab-pane#userview -->

<!-- Advanced View #advancedview -->
<div class="tab-pane" id="advancedview">
  <div class="table-responsive no-padding">
    <table class="table table-hover">
      <thead style="letter-spacing:1px;">
        <tr>
          <th></th>
          <th>LDAP</th>
          <th>MySQL</th>
        </tr>
      </thead>
      <tbody>
        <?php foreach ($account_keys as $key => $value): ?>
          <?php if($ldap_view["Employee Type"] == "<strong>No
data</strong>") {$ldap_view["Employee Type"] = 'STUD'; }?>
          <?php if($ldap_view[$value] != "Not stored" &&
$mysql_view[$value] != "Not stored"){
            if($ldap_view[$value] != $mysql_view[$value]){
              echo '<tr class="bg-red color-palette">';
            }
          }
          else { echo '<tr>'; }
          ?>
          <th class="col-md-2"><?php echo $value; ?></th>
          <td class="col-md-4 <?php if($ldap_view[$value] == "Not
stored"){ echo 'text-muted'; } ?> ">
            <?php echo $ldap_view[$value]; ?></td>
            <td class="col-md-4 <?php if($mysql_view[$value] == "Not
stored"){ echo 'text-muted'; } ?>"><?php echo $mysql_view[$value]; ?></td>
          </tr>
        <?php endforeach; ?>
      </tbody>
    </table>
  </div> <!-- /div.table-responsive -->
</div> <!-- /div.tab-pane#advancedview -->

<!-- Change Password Logs #changepass -->
<div class="tab-pane" id="changepass">
  <div class="table-responsive no-padding">
    <table id="changepass_results" class="table table-bordered table-
hover">
      <thead>
        <tr>
          <th>Username</th>
          <th>IP</th>
          <th>Changed by</th>
          <th>Date / Time</th>
          <th>Notes</th>
        </tr>
      </thead>
      <tbody>
        <?php foreach($results_changepass as $r): ?>
          <tr>
            <td><?php echo $r["username"]; ?></td>
            <td><?php echo $r["ip"]; ?></td>
            <td><?php echo $r["changed_by"]; ?></td>
            <td><?php echo $r["time"]; ?></td>
            <td><?php echo $r["notes"]; ?></td>
          </tr>
        </tbody>
      </table>
    </div>
  </div>

```

```

        <td><?php echo $r["notes"]; ?></td>
    </tr>
    <?php endforeach; //.results ?>
</tbody>
<tfoot>
    <tr>
        <th>Username</th>
        <th>IP</th>
        <th>Changed by</th>
        <th>Date / Time</th>
        <th>Notes</th>
    </tr>
</tfoot>
</table>
</div> <!-- /div.table-responsive -->
</div> <!-- /div.tab-pane#changepass -->

<!-- Account Logs #accountlogs -->
<div class="tab-pane" id="accountlogs">
    <div class="table-responsive no-padding">
        <table id="accountlog_results" class="table table-bordered table-

```

hover">

```

        <thead>
            <tr>
                <th>ID</th>
                <th>Date/Time</th>
                <th>Account</th>
                <th>Detail</th>
                <th>Service</th>
                <th>Log Level</th>
                <th>Changed by</th>
                <th>IP</th>
            </tr>
        </thead>
        <tbody>
            <?php foreach($results_accountlog as $r): ?>
                <tr>
                    <td><?php echo $r["id"]; ?></td>
                    <td><?php echo $r["time"]; ?></td>
                    <td><?php echo $r["account"]; ?></td>
                    <td><?php echo $r["detail"]; ?></td>
                    <td><?php echo $r["service"]; ?></td>
                    <td><?php echo $r["log_level"]; ?></td>
                    <td><?php echo $r["changed_by"]; ?></td>
                    <td><?php echo $r["ip"]; ?></td>
                </tr>
            <?php endforeach; //.results ?>
        </tbody>
    </div>
</div>
<tfoot>
    <tr>
        <th>ID</th>
        <th>Date/Time</th>
        <th>Account</th>
        <th>Detail</th>
        <th>Service</th>
        <th>Log Level</th>
        <th>Changed by</th>
        <th>IP</th>
    </tr>

```

```

        </tfoot>
    </table>
</div> <!-- /div.table-responsive -->
</div> <!-- /div.tab-pane#accountlogs -->

<!-- Lost Password Logs #lostpass -->
<div class="tab-pane" id="lostpass">
    <div class="table-responsive no-padding">
        <table id="lostpass_results" class="table table-bordered table-hover">
            <thead>
                <tr>
                    <th>Username</th>
                    <th>Date Added</th>
                    <th>Active</th>
                </tr>
            </thead>
            <tbody>
                <?php foreach($results_lostpass as $r): ?>
                    <tr>
                        <td><?php echo $r["username"]; ?></td>
                        <td><?php echo $r["date_added"]; ?></td>
                        <td><?php echo $r["active"]; ?></td>
                    </tr>
                <?php endforeach; //.results ?>
            </tbody>
            <tfoot>
                <tr>
                    <th>Username</th>
                    <th>Date Added</th>
                    <th>Active</th>
                </tr>
            </tfoot>
        </table>
    </div> <!-- /div.table-responsive -->
</div> <!-- /div.tab-pane#lostpass -->
</div> <!-- /div.tab-content -->
</div><!-- /div.nav-tabs-custom -->
</div><!-- /div.col-xs-12 -->
</div><!-- /div.row -->

```

application/views/create_multiple_employees_js.php

```

<!-- DataTables -->
<script src="<?php echo $this->config->item('theme_url');
?>/plugins/datatables/jquery.dataTables.min.js"></script>
<script src="<?php echo $this->config->item('theme_url');
?>/plugins/datatables/dataTables.bootstrap.min.js"></script>

<!-- page script -->
<script>

$(function () {
    $("#csv_upload_results").DataTable({
        "bSort" : false,
        "ordering": false,
        "searching": false,
        "info": true
    });

```

```

    });

    var timeout = 5000; // in milliseconds (3*1000)

    $('.callout').delay(timeout).fadeOut(300);

    function changeText(){
        $('#create_activate_multiple').text("Activating ... Please wait.");
        $('#create_activate_multiple').removeClass('btn-info').addClass('btn-
warning');
    };
</script>

```

application/views/dashboard_css.php

```

<style type="text/css">
    .chart-legend li span{
        display: inline-block;
        width: 12px;
        height: 12px;
        margin-right: 5px;
    }

    .chart-legend li{
        display: inline;
        margin-right: 15px;
    }

    .chart-legend ul{
        list-style: none;
    }

    .pie ul li {
        display: block;
    }

    .pie ul {
        padding-left: 0;
    }

</style>

```

application/views/dashboard_js.php

```

<!-- ChartJS 1.0.1 -->
<script src="<?php echo $this->config->item('theme_url');
?>/plugins/chartjs/Chart.min.js"></script>
<script>
    $(function () {

        'use strict';

        /* ChartJS
         * -----
         * Here we will create a few charts using ChartJS

```

```

*/

//-----
// - NEW ACCOUNTS CHART -
//-----

// Get context with jQuery - using jQuery's .get() method.
var dilnetChartCanvas = $("#dilnetChart").get(0).getContext("2d");
// This will get the first returned node in the jQuery collection.
var dilnetChart = new Chart(dilnetChartCanvas);

var dilnetChartData = {
    labels: ["January", "February", "March", "April", "May", "June", "July",
"August", "September", "October", "November", "December"],
    datasets: [
        {
            label: <?php echo $stats['graph_prevyear_legend']; ?>,
            fillColor: "rgba(220,220,220,0.2)",
            strokeColor: "rgba(220,220,220,1)",
            pointColor: "rgba(220,220,220,1)",
            pointStrokeColor: "#fff",
            pointHighlightFill: "#fff",
            pointHighlightStroke: "rgba(220,220,220,1)",
            data: <?php echo $stats['graph_prevyear_datastr']; ?>
        },
        {
            label: <?php echo $stats['graph_curryear_legend']; ?>,
            fillColor: "rgba(151,187,205,0.2)",
            strokeColor: "rgba(151,187,205,1)",
            pointColor: "rgba(151,187,205,1)",
            pointStrokeColor: "#fff",
            pointHighlightFill: "#fff",
            pointHighlightStroke: "rgba(151,187,205,1)",
            data: <?php echo $stats['graph_curryear_datastr']; ?>
        }
    ]
};

var dilnetChartOptions = {
    //Boolean - If we should show the scale at all
    showScale: true,
    //Boolean - Whether grid lines are shown across the chart
    scaleShowGridLines: false,
    //String - Colour of the grid lines
    scaleGridLineColor: "rgba(0,0,0,.05)",
    //Number - Width of the grid lines
    scaleGridLineWidth: 1,
    //Boolean - Whether to show horizontal lines (except X axis)
    scaleShowHorizontalLines: true,
    //Boolean - Whether to show vertical lines (except Y axis)
    scaleShowVerticalLines: true,
    //Boolean - Whether the line is curved between points
    bezierCurve: true,
    //Number - Tension of the bezier curve between points
    bezierCurveTension: 0.3,
    //Boolean - Whether to show a dot for each point
    pointDot: true,
    //Number - Radius of each point dot in pixels
    pointDotRadius: 4,

```

```

//Number - Pixel width of point dot stroke
pointDotStrokeWidth: 1,
//Number - amount extra to add to the radius to cater for hit detection
outside the drawn point
pointHitDetectionRadius: 20,
//Boolean - Whether to show a stroke for datasets
datasetStroke: true,
//Number - Pixel width of dataset stroke
datasetStrokeWidth: 2,
//Boolean - Whether to fill the dataset with a color
datasetFill: true,
//String - A legend template
legendTemplate : "<ul class=\"%name.toLowerCase()%>-legend\"><% for (var
i=0; i<datasets.length; i++){%><li><span style=\"background-
color: \"%=datasets[i].strokeColor%\"></span><%if(datasets[i].label){%><%=datasets[
i].label%><%}%></li><%}%></ul>",
//Boolean - whether to maintain the starting aspect ratio or not when
responsive, if set to false, will take up entire container
maintainAspectRatio: true,
//Boolean - whether to make the chart responsive to window resizing
responsive: true,
scaleShowLabels: false,
scaleLineColor: 'transparent'
};

```

```

//Create the line chart
var dilnetChartLine = dilnetChart.Line(dilnetChartData, dilnetChartOptions);
//then you just need to generate the legend
var legend = dilnetChartLine.generateLegend();

//and append it to your page somewhere
$('##js-legend').append(legend);
//-----
// - END NEW ACCOUNTS CHART -
//-----

```

```

//-----
// - RESET PASSWORD CHART -
//-----
// Get context with jQuery - using jQuery's .get() method.
var dilnetChartCanvas = $("#dilnetChartReset").get(0).getContext("2d");
// This will get the first returned node in the jQuery collection.
var dilnetChart = new Chart(dilnetChartCanvas);

var dilnetChartData = {
  labels: ["January", "February", "March", "April", "May", "June", "July",
"August", "September", "October", "November", "December"],
  datasets: [
    {
      label: <?php echo $stats['graph_prevyear_legend']; ?>,
      fillColor: "rgba(220,220,220,0.2)",
      strokeColor: "rgba(220,220,220,1)",
      pointColor: "rgba(220,220,220,1)",
      pointStrokeColor: "#fff",
      pointHighlightFill: "#fff",
      pointHighlightStroke: "rgba(220,220,220,1)",
      data: <?php echo $stats['graph_prevyear_datastr_reset']; ?>
    },
    {

```



```

    label: <?php echo $stats['graph_curryear_legend']; ?>,
    fillColor: "rgba(151,187,205,0.2)",
    strokeColor: "rgba(151,187,205,1)",
    pointColor: "rgba(151,187,205,1)",
    pointStrokeColor: "#fff",
    pointHighlightFill: "#fff",
    pointHighlightStroke: "rgba(151,187,205,1)",
    data: <?php echo $stats['graph_curryear_datastr_reset']; ?>
  }
];
};

```

```

var dilnetChartOptions = {
  //Boolean - If we should show the scale at all
  showScale: true,
  //Boolean - Whether grid lines are shown across the chart
  scaleShowGridLines: false,
  //String - Colour of the grid lines
  scaleGridLineColor: "rgba(0,0,0,.05)",
  //Number - Width of the grid lines
  scaleGridLineWidth: 1,
  //Boolean - Whether to show horizontal lines (except X axis)
  scaleShowHorizontalLines: true,
  //Boolean - Whether to show vertical lines (except Y axis)
  scaleShowVerticalLines: true,
  //Boolean - Whether the line is curved between points
  bezierCurve: true,
  //Number - Tension of the bezier curve between points
  bezierCurveTension: 0.3,
  //Boolean - Whether to show a dot for each point
  pointDot: true,
  //Number - Radius of each point dot in pixels
  pointDotRadius: 4,
  //Number - Pixel width of point dot stroke
  pointDotStrokeWidth: 1,
  //Number - amount extra to add to the radius to cater for hit detection
  //outside the drawn point
  pointHitDetectionRadius: 20,
  //Boolean - Whether to show a stroke for datasets
  datasetStroke: true,
  //Number - Pixel width of dataset stroke
  datasetStrokeWidth: 2,
  //Boolean - Whether to fill the dataset with a color
  datasetFill: true,
  //String - A legend template
  legendTemplate : "<ul class=\"%<%=name.toLowerCase()%>-legend\"><% for (var i=0; i<datasets.length; i++){%<li><span style=\"%background-color: \"%=datasets[i].strokeColor%\"></span><%if(datasets[i].label){%<%=datasets[i].label%><%}%></li><%}%></ul>",
  //Boolean - whether to maintain the starting aspect ratio or not when
  //responsive, if set to false, will take up entire container
  responsive: true,
  maintainAspectRatio: true,
  //Boolean - whether to make the chart responsive to window resizing
  responsive: true,
  scaleShowLabels: false,
  scaleLineColor: 'transparent'
};

```

```

//Create the line chart

```

```

var dilnetChartLine = dilnetChart.Line(dilnetChartData, dilnetChartOptions);
//then you just need to generate the legend
var legend = dilnetChartLine.generateLegend();

//and append it to your page somewhere
$('#js-legend-reset').append(legend);
//-----
// - END RESET PASSWORD CHART -
//-----

//-----
// - LOSTPASS FEATURE CHART -
//-----
// Get context with jQuery - using jQuery's .get() method.
var dilnetChartCanvas = $("#dilnetChartLostpass").get(0).getContext("2d");
// This will get the first returned node in the jQuery collection.
var dilnetChart = new Chart(dilnetChartCanvas);

var dilnetChartData = {
  labels: ["January", "February", "March", "April", "May", "June", "July",
"August", "September", "October", "November", "December"],
  datasets: [
    {
      label: <?php echo $stats['graph_prevyear_legend'];?>,
      fillColor: "rgba(220,220,220,0.2)",
      strokeColor: "rgba(220,220,220,1)",
      pointColor: "rgba(220,220,220,1)",
      pointStrokeColor: "#fff",
      pointHighlightFill: "#fff",
      pointHighlightStroke: "rgba(220,220,220,1)",
      data: <?php echo $stats['graph_prevyear_datastr_lp'];?>
    },
    {
      label: <?php echo $stats['graph_curryear_legend'];?>,
      fillColor: "rgba(151,187,205,0.2)",
      strokeColor: "rgba(151,187,205,1)",
      pointColor: "rgba(151,187,205,1)",
      pointStrokeColor: "#fff",
      pointHighlightFill: "#fff",
      pointHighlightStroke: "rgba(151,187,205,1)",
      data: <?php echo $stats['graph_curryear_datastr_lp'];?>
    }
  ]
};

var dilnetChartOptions = {
  //Boolean - If we should show the scale at all
  showScale: true,
  //Boolean - Whether grid lines are shown across the chart
  scaleShowGridLines: false,
  //String - Colour of the grid lines
  scaleGridLineColor: "rgba(0,0,0,.05)",
  //Number - Width of the grid lines
  scaleGridLineWidth: 1,
  //Boolean - Whether to show horizontal lines (except X axis)
  scaleShowHorizontalLines: true,
  //Boolean - Whether to show vertical lines (except Y axis)
  scaleShowVerticalLines: true,
  //Boolean - Whether the line is curved between points

```

```

bezierCurve: true,
//Number - Tension of the bezier curve between points
bezierCurveTension: 0.3,
//Boolean - Whether to show a dot for each point
pointDot: true,
//Number - Radius of each point dot in pixels
pointDotRadius: 4,
//Number - Pixel width of point dot stroke
pointDotStrokeWidth: 1,
//Number - amount extra to add to the radius to cater for hit detection
outside the drawn point
pointHitDetectionRadius: 20,
//Boolean - Whether to show a stroke for datasets
datasetStroke: true,
//Number - Pixel width of dataset stroke
datasetStrokeWidth: 2,
//Boolean - Whether to fill the dataset with a color
datasetFill: true,
//String - A legend template
legendTemplate : "<ul class=\"%>name.toLowerCase()%-legend\"><% for (var
i=0; i<datasets.length; i++){%><li><span style=\"%background-
color:=%>datasets[i].strokeColor%\"></span><%if(datasets[i].label){%><%=datasets[
i].label%><%}%></li><%}%></ul>",
//Boolean - whether to maintain the starting aspect ratio or not when
responsive, if set to false, will take up entire container
maintainAspectRatio: true,
//Boolean - whether to make the chart responsive to window resizing
responsive: true,
scaleShowLabels: false,
scaleLineColor: 'transparent'
};

```

```

//Create the line chart
var dilnetChartLine = dilnetChart.Line(dilnetChartData, dilnetChartOptions);
//then you just need to generate the legend
var legend = dilnetChartLine.generateLegend();

```

```

//and append it to your page somewhere
$('#js-legend-lp').append(legend);

```

```

//-----
// - END LOSTPASS CHART -
//-----

```

```

//-----

```

```

// - PIE CHART -
//-----
// Get context with jQuery - using jQuery's .get() method.
var pieChartCanvas = $("#techPieChart").get(0).getContext("2d");
var pieChart = new Chart(pieChartCanvas);

```

```

var PieData = <?php echo $stats['pie_tech_data']; ?>;
var pieOptions = {
//Boolean - Whether we should show a stroke on each segment
segmentShowStroke: true,
//String - The colour of each segment stroke
segmentStrokeColor: "#fff",
//Number - The width of each segment stroke
segmentStrokeWidth: 1,
//Number - The percentage of the chart that we cut out of the middle

```

```

percentageInnerCutout: 50, // This is 0 for Pie charts
//Number - Amount of animation steps
animationSteps: 100,
//String - Animation easing effect
animationEasing: "easeOutBounce",
//Boolean - Whether we animate the rotation of the Doughnut
animateRotate: true,
//Boolean - Whether we animate scaling the Doughnut from the centre
animateScale: false,
//Boolean - whether to make the chart responsive to window resizing
responsive: true,
// Boolean - whether to maintain the starting aspect ratio or not when
responsive, if set to false, will take up entire container
maintainAspectRatio: false,
//String - A legend template
legendTemplate: "<ul class=\"<%=name.toLowerCase()%>-legend\"><% for (var i=0;
i<segments.length; i++){%><li><span style=\"background-
color:<%=segments[i].fillColor%>\"></span><%if(segments[i].label){%><%=segments[i]
.label%> [ <%=segments[i].value%> ] <%}%></li><%}%></ul>",
//String - A tooltip template
tooltipTemplate: "<%=value %> <%=label%>"
};
//Create pie or doughnut chart
// You can switch between pie and doughnut using the method below.
var techStaffPie = pieChart.Doughnut(PieData, pieOptions);

//then you just need to generate the legend
var pie_legend = techStaffPie.generateLegend();

//and append it to your page somewhere
$('#js-legend-pie').append(pie_legend);
//-----
// - END PIE CHART -
//-----
//-----
// - PIE CHART -
//-----
// Get context with jQuery - using jQuery's .get() method.
var pieChartCanvas = $("#acctypePieChart").get(0).getContext("2d");
var pieChart = new Chart(pieChartCanvas);

var PieData = <?php echo $stats['pie_acctype_data']; ?>;
var pieOptions = {
//Boolean - Whether we should show a stroke on each segment
segmentShowStroke: true,
//String - The colour of each segment stroke
segmentStrokeColor: "#fff",
//Number - The width of each segment stroke
segmentStrokeWidth: 1,
//Number - The percentage of the chart that we cut out of the middle
percentageInnerCutout: 50, // This is 0 for Pie charts
//Number - Amount of animation steps
animationSteps: 100,
//String - Animation easing effect
animationEasing: "easeOutBounce",
//Boolean - Whether we animate the rotation of the Doughnut
animateRotate: true,
//Boolean - Whether we animate scaling the Doughnut from the centre

```

```

animateScale: false,
//Boolean - whether to make the chart responsive to window resizing
responsive: true,
// Boolean - whether to maintain the starting aspect ratio or not when
responsive, if set to false, will take up entire container
maintainAspectRatio: false,
//String - A legend template
legendTemplate: "<ul class=\"<%=name.toLowerCase()%>-legend\"><% for (var i=0;
i<segments.length; i++){%><li><span style=\"background-
color:<%=segments[i].fillColor%>\"></span><%if(segments[i].label){%><%=segments[i]
.label%> [ <%=segments[i].value%> ] <%}%></li><%}%></ul>",
//String - A tooltip template
tooltipTemplate: "<%=value %> <%=label%>"
};
//Create pie or doughnut chart
// You can switch between pie and doughnut using the method below.
var accTypePie = pieChart.Doughnut(PieData, pieOptions);

//then you just need to generate the legend
var pie_legend = accTypePie.generateLegend();

//and append it to your page somewhere
$('##js-legend-pie-type').append(pie_legend);
//-----
// - END PIE CHART -
//-----
});
</script>

```

application/views/dashboard.php

```

<div class="row">
  <div class="col-md-12">
    <div class="box">
      <div class="box-footer">
        <div class="row">
          <div class="col-sm-3 col-xs-6">
            <div class="description-block border-right">
              <div class="description-block header">
                <?php echo $stats['prevmonth_rateval']; ?>
                <h5 class="description-header"><?php echo
$stats['accounts_prevmonth']; ?></h5>
                <span class="description-text">NEW ACCOUNTS<br><?php echo date('M
Y',strtotime("-1 month")); ?></span>
              </div><!-- /.description-block -->
            </div><!-- /.col -->
          <div class="col-sm-3 col-xs-6">
            <div class="description-block border-right">
              <div class="description-block header">
                <?php echo $stats['currmonth_rateval']; ?>
                <h5 class="description-header"><?php echo
$stats['accounts_currmonth']; ?></h5>
                <span class="description-text">NEW ACCOUNTS<br><strong><?php echo
date('M Y'); ?></strong></span>
              </div><!-- /.description-block -->
            </div><!-- /.col -->
          <div class="col-sm-3 col-xs-6">

```

```

        <div class="description-block border-right">
            <?php echo $stats['prevmonth_rateval_reset']; ?>
            <h5 class="description-header"><?php echo $stats['reset_prevmonth'];
?></h5>
            <span class="description-text">PASSWORD RESET BY TECH STAFF<br><?php
echo date('M Y',strtotime("-1 month")); ?></span>
        </div><!-- /.description-block -->
    </div><!-- /.col -->
    <div class="col-sm-3 col-xs-6">
        <div class="description-block border-right">
            <?php echo $stats['currmonth_rateval_reset']; ?>
            <h5 class="description-header"><?php echo $stats['reset_currmonth'];
?></h5>
            <span class="description-text">PASSWORD RESET BY TECH
STAFF<br><strong><?php echo date('M Y'); ?></strong></span>
        </div><!-- /.description-block -->
    </div>
</div><!-- /.row -->
</div><!-- /.box-footer -->
</div><!-- /.box -->
</div><!-- /.col-md-12 -->
</div><!-- /.row -->
<div class="row">
    <div class="col-md-4">
        <div class="box box-success">
            <div class="box-header with-border">
                <h3 class="box-title">New DilNet Accounts Statistics</h3>
            </div>
            <div class="box-body">
                <div class="row">
                    <div class="col-md-12">
                        <div id="js-legend" class="chart-legend text-center"></div>
                        <div class="chart">
                            <canvas id="dilnetChart" style="height: 180px;"></canvas>
                        </div><!-- /.chart-responsive -->
                    </div><!-- /.col -->
                </div><!-- /.row -->
            </div><!-- /.box-body -->
        </div><!-- /.box -->
    </div><!-- /.col-md-8 -->
    <div class="col-md-4">
        <div class="box box-danger">
            <div class="box-header with-border">
                <h3 class="box-title">Reset Password Statistics</h3>
            </div>
            <div class="box-body">
                <div class="row">
                    <div class="col-md-12">
                        <div id="js-legend-reset" class="chart-legend text-center"></div>
                        <div class="chart">
                            <canvas id="dilnetChartReset" style="height: 180px;"></canvas>
                        </div><!-- /.chart-responsive -->
                    </div><!-- /.col -->
                </div><!-- /.row -->
            </div><!-- /.box-body -->
        </div><!-- /.box -->
    </div><!-- /.col-md-12 -->
    <div class="col-md-4">
        <div class="box box-warning">

```

```

<div class="box-header with-border">
  <h3 class="box-title">Lostpass Statistics</h3>
</div>
<div class="box-body">
  <div class="row">
    <div class="col-md-12">
      <div id="js-Legend-Lp" class="chart-Legend text-center"></div>
      <div class="chart">
        <canvas id="dilnetChartLostpass" style="height: 180px;"></canvas>
      </div><!-- /.chart-responsive -->
    </div><!-- /.col -->
  </div><!-- /.row -->
</div><!-- /.box-body -->
</div><!-- /.box -->
</div><!-- /.col-md-12 -->
</div><!-- /.row -->
<div class="row">
  <div class="col-md-6">
    <div class="box box-default">
      <div class="box-header with-border">
        <h3 class="box-title">Tech Staff Stats: Student Accounts created in <?php
echo $stats['graph_curryear_legend']; ?></h3>
      <div class="box-tools pull-right">
        <button type="button" class="btn btn-box-tool" data-widget="collapse"><i
class="fa fa-minus"></i>
      </button>
    </div>
  </div>
<!-- /.box-header -->
<div class="box-body" style="display: block;">
  <div class="row">
    <div class="col-md-6">
      <div class="chart-responsive">
        <div class="chartPieChart" height="320" width="444" style="width:
222px; height: 200px;"></div><!-- /.chart-responsive -->
      </div><!-- /.col -->
    <div class="col-md-6">
      <div id="js-Legend-pie" class="pie chart-Legend clearfix"></div>
    </div><!-- /.col -->
  </div><!-- /.row -->
</div><!-- /.box-body -->
</div><!-- /.box -->
</div><!-- /.col-md-5 -->
<div class="col-md-6">
  <div class="box box-default">
    <div class="box-header with-border">
      <h3 class="box-title">Accounts Type created in <?php echo
$stats['graph_curryear_legend']; ?></h3>
    <div class="box-tools pull-right">
      <button type="button" class="btn btn-box-tool" data-widget="collapse"><i
class="fa fa-minus"></i>
    </button>
  </div>
  </div>
</div>
<!-- /.box-header -->
<div class="box-body" style="display: block;">
  <div class="row">
    <div class="col-md-6">

```

```

        <div class="chart-responsive">
        <canvas id="acctypePieChart" height="320" width="444" style="width:
222px; height: 200px;"></canvas>
        </div><!-- /.chart-responsive -->
    </div><!-- /.col -->
    <div class="col-md-6">
        <div id="js-legend-pie-type" class="pie chart-legend clearfix"></div>
    </div><!-- /.col -->
</div> <!-- /.row -->
</div><!-- /.box-body -->
</div><!-- /.box -->
</div><!-- /.col-md-5 -->
</div><!-- /.row -->

```

application/views/dilnet_account_tools_css.php

```

<!-- iCheck -->
<link rel="stylesheet" href="<?php echo $this->config->item('theme_url');
?>/plugins/iCheck/square/blue.css">

<style>
    .dilnetacc-radio {
        margin-left: -15px;
    }

</style>

```

application/views/dilnet_account_tools_js.php

```

<script src="<?php echo $this->config->item('theme_url');
?>/plugins/iCheck/ichack.min.js"></script>
<script>
$(function(){
    $('input').iCheck({
        checkboxClass: 'icheckbox_square-blue',
        radioClass: 'iradio_square-blue',
    });
    var timeout = 3000; // in milliseconds (3*1000)
    $('.callout').delay(timeout).fadeOut(300);
</script>

```

application/views/dilnet_account_tools.php

```

<div class="row">
    <div class="col-md-6">
        <!-- Horizontal Form -->
        <div class="box box-info">
            <div class="box-header with-border">
                <h3 class="box-title"><i class="fa fa-folder-open-o"></i>View Account</h3>
            </div><!-- /.box-header -->
            <!-- form start -->

```



```

        Employee Number
    </label>
    <label class="checkbox-inline">
        <input type="radio" name="print_radio" value="STUD"
class="minimal">
        Student Number
    </label>
</div>
</div><!--div.form-group -->
<div class="form-group <?php echo (form_error('print_inp') == '') ? '' :
'has-error'; ?>">
    <div class="col-xs-12">
        <?php echo form_error('print_inp'); ?>
        <div class="input-group">
            <div class="input-group-addon"><i class="fa fa-user"></i></div>
            <div class="input-group-addon"><i class="fa fa-user"></i></div>
            <input type="text" class="form-control input-lg" name="print_inp">
        </div><!-- /.input-group -->
    </div>
</div>
</div><!-- /.box-body -->
<div class="box-footer">
    <button type="submit" class="btn btn-info col-xs-12">Print
Account</button>
</div><!-- /.box-footer -->
<?php echo form_close(); ?>
</div><!-- /.box -->
</div><!-- /.col-md-6 -->
</div><!-- /.row -->
<div class="row">
    <div class="col-md-6">
        <!-- Horizontal Form -->
        <div class="box box-info">
            <div class="box-header with-border">
                <div class="box-header with-border"><i class="fa fa-lock"></i>Reset Password</h3>
                <h3 class="box-title"><i class="fa fa-lock"></i>Reset Password</h3>
            </div><!-- /.box-header -->
            <!-- form start -->
            <?php echo form_open('tech/dilnetacc_tools/reset_form','class="form-
horizontal"'); ?>
            <div class="box-body">
                <div class="form-group">
                    <div class="col-xs-12 dilnetacc-radio">
                        <div class="checkbox-inline">
                            <label class="checkbox-inline" name="reset_radio" value="UN" class="minimal"
checked>
                                Username
                            </label>
                            <label class="checkbox-inline">
                                <input type="radio" name="reset_radio" value="EMP"
class="minimal">
                                Employee Number
                            </label>
                            <label class="checkbox-inline">
                                <input type="radio" name="reset_radio" value="STUD"
class="minimal">
                                Student Number
                            </label>
                        </div>
                    </div><!--div.form-group -->

```

```

        <div class="form-group <?php echo (form_error('reset_inp') == '') ? '' :
'has-error'; ?>">
            <div class="col-xs-12">
                <?php echo form_error('reset_inp'); ?>
                <div class="input-group">
                    <div class="input-group-addon"><i class="fa fa-user"></i></div>
                    <input type="text" class="form-control input-lg" name="reset_inp">
                </div><!-- /.input-group -->
            </div>
        </div><!-- /.box-body -->
        <div class="box-footer">
            <button type="submit" class="btn btn-info col-xs-12">Reset
Password</button>
        </div><!-- /.box-footer -->
        <?php echo form_close(); ?>
    </div><!-- /.box -->
</div><!-- /.col-md-6 -->
<div class="col-md-6">
    <!-- Horizontal Form -->
    <div class="box box-info">
        <div class="box-header with-border">
            <h3 class="box-title"><i class="fa fa-remove"></i>Disable Account</h3>
        </div><!-- /.box-header -->
        <!-- form start -->
        <?php echo form_open('tech/dilnetacc_tools/disable_form','class="form-
horizontal"'); ?>
        <div class="box-body">
            <div class="form-group">
                <div class="col-xs-12 dilnetacc-radio">
                    <label class="checkbox-inline">
                        <input type="radio" name="disable_radio" value="UN"
class="minimal" checked>
                        Username
                    </label>
                    <label class="checkbox-inline">
                        <input type="radio" name="disable_radio" value="EMP"
class="minimal">
                        Employee Number
                    </label>
                    <label class="checkbox-inline">
                        <input type="radio" name="disable_radio" value="STUD"
class="minimal">
                        Student Number
                    </label>
                </div>
            </div><!--div.form-group -->
            <div class="form-group <?php echo (form_error('disable_inp') == '') ? ''
: 'has-error'; ?>">
                <div class="col-xs-12">
                    <?php echo form_error('disable_inp'); ?>
                    <div class="input-group">
                        <div class="input-group-addon"><i class="fa fa-user"></i></div>
                        <div class="input-group-addon">
                            <input type="text" class="form-control input-lg"
name="disable_inp">
                        </div><!-- /.input-group -->
                    </div>
                </div>
            </div><!-- /.box-body -->
        </div><!-- /.box-body -->

```

```

        <div class="box-footer">
            <button type="submit" class="btn btn-info col-xs-12">Disable
Account</button>
        </div><!-- /.box-footer -->
        <?php echo form_close(); ?>
    </div><!-- /.box -->
</div><!-- /.col-md-6 -->
</div><!-- /.row -->

```

application/views/dilnet_lookup_css.php

```

<!-- DataTables -->
<link rel="stylesheet" href="<?php echo $this->config->item('theme_url');
?>/plugins/datatables/dataTables.bootstrap.css">

<style>

    /*
    i {
        margin-right: 5px;
    }
    */

</style>

```

application/views/dilnet_lookup_js.php

```

<!-- DataTables -->
<script src="<?php echo $this->config->item('theme_url');
?>/plugins/datatables/jquery.dataTables.min.js"></script>
<script src="<?php echo $this->config->item('theme_url');
?>/plugins/datatables/dataTables.bootstrap.min.js"></script>

<!-- page script -->
<script>

    $(function () {
        $("#acct_search_results").DataTable({
            "bSort" : false
        });
    });

    $(function () {
        $("#merge_search_results").DataTable({
            "bSort" : false,
            "paging": false,
            "ordering": false,
            "searching": false,
            "info": false
        });
        $("#merge_search_results tfoot").remove();
    });

    $(function () {
        $("#log_search_results").DataTable({
            "bSort" : true,

```

```

        "paging": true,
        "ordering": true,
        "searching": true,
        "info": true
    });
});

$(function () {
    $("#mail_search_results").DataTable({
        "bSort": true,
        "paging": true,
        "ordering": true,
        "searching": true,
        "info": true
    });
});
var timeout = 5000; // in milliseconds (3*1000)

$('.callout').delay(timeout).fadeOut(300);

function changeText(){
    if ($('#name_str_dilnet_lookup').val().length >= 3) {
        // value is empty or is default value
        $("#submitButton_dilnet_lookup").text("Searching ... Please wait
...");
        $('#submitButton_dilnet_lookup').removeClass('btn-
info').addClass('btn-warning');
    }
};

function changeMergeSearchText(){
    if ($('#merge_search_field').val().length >= 3) {
        // value is empty or is default value
        $("#mergeSubmitButton").text("Searching ... Please wait ...");
        $('#mergeSubmitButton').removeClass('btn-info').addClass('btn-
warning');
    }
};
</script>

```

application/views/dilnet_lookup.php

```

<div class="row">
    <div class="col-xs-12">
        <!-- Horizontal Form -->
        <div class="box box-info">
            <div class="box-header with-border">
                <h3 class="box-title"><i class="fa fa-search"></i> Search DilNet
Account</h3>
            </div><!-- /.box-header -->
            <!-- form start -->
            <?php echo form_open('tech/dilnet_lookup/name_search_form', 'class="form-
horizontal"> ?>
            <div class="box-body">
                <div class="form-group"><?php echo (form_error('name_str') == '') ? '' :
'has-error'; ?>>

```

```

<div class="col-xs-12">
    <?php echo form_error('name_str'); ?>
    <div class="input-group">
        <div class="input-group-addon"><i class="fa fa-search"></i></div>
        <input type="text" id="name_str_dilnet_lookup" class="form-control
input-lg" name="name_str" placeholder="Name, Username, Email Address, SN xxxx-
xxxxx, or Employee number" required autofocus>
    </div><!-- /.input-group -->
    </div>
</div>
<div><!-- /.box-body -->
<div class="box-footer">
    <button type="submit" id="submitButton_dilnet_lookup" class="btn btn-
info col-xs-12" onclick="return changeText();">Search</button>
</div><!-- /.box-footer -->
<?php echo form_close(); ?>
</div><!-- /.box -->
</div><!-- /.col-md-6 -->
</div><!-- /.row -->

```

```

<?php if(isset($results_db)): ?>
<?php // print "<pre>"; print_r($_SESSION); print "</pre>"; ?>
<div class="row">
    <div class="col-xs-12">
        <div class="box">
            <div class="box-header">
                <h3 class="box-title">DB Search Results <?php if(isset($search_str)){ echo
"for '<strong>".$search_str."</strong>'"; } ?></h3>
            </div><!-- /.box-header -->
            <div class="box-body hidden-xs hidden-sm hidden-md">
                <table id="acct_search_results" class="table table-bordered table-hover
table-responsive">
                    <thead>
                        <tr>
                            <th class="text-center">Username</th>
                            <th class="text-center">Name</th>
                            <th class="text-center">Email</th>
                            <th class="text-center">ID Number</th>
                            <th class="text-center">Actions</th>
                        </tr>
                    </thead>
                    <tbody>
                        <?php foreach($results_db as $r): ?>
                            <tr>
                                <td><?php echo $r["username"]; ?></td>
                                <td><?php echo $r["firstname"]." ".$r["middlename"]."
".$r["lastname"]; ?></td>
                                <td><?php echo $r["email"]; ?></td>
                                <td><?php echo $r["id_number"]; ?></td>
                                <td class="text-center"><a href="<?php echo
base_url('tech/dilnetacc_tools/view_account/' . $r['username']); ?>"><i class="fa
fa-file-o fa-eye" data-toggle="tooltip" data-placement="auto" data-html="true"
title="View Account"></i></a>
                                <a href="<?php echo
base_url('tech/dilnetacc_tools/edit_account_form/' . $r['username']); ?>"
style="margin-left: 5px;"><i class="fa fa-edit" data-toggle="tooltip" data-
placement="auto" data-html="true" title="Edit Account"></i></a>
                                <a href="<?php echo
base_url('tech/print_account/confirm_print/' . $r['username']); ?>" style="margin-

```

```

left: 5px;"><i class="fa fa-print hidden-xs" data-toggle="tooltip" data-
placement="auto" data-html="true" title="Print Account"></i></a>
    <a href="<?php echo
base_url('tech/merge/merge_add/'. $r['username']); ?>" style="margin-left: 5px;"><i
class="fa fa-search-plus" data-toggle="tooltip" data-placement="auto" data-
html="true" title="Add to Merge"></i></a>
    </td>
</tr>
<?php endforeach; //.results_db ?>

</tbody>
<tfoot>
<tr>
<th class="text-center">Username</th>
<th class="text-center">Name</th>
<th class="text-center">Email</th>
<th class="text-center">ID Number</th>
<th class="text-center">Actions</th>
</tr>
</tfoot>
</table>
</div><!-- /.box-body -->
<div class="box-body hidden-lg hidden-xl" style="overflow: auto;">
<table id="acct_search_results" class="table table-bordered table-hover
table-responsive">
<thead>
<tr>
<th class="text-center">Username</th>
<th class="text-center">Name</th>
<th class="text-center">Email</th>
<th class="text-center">ID Number</th>
<th class="text-center">Actions</th>
</tr>
</thead>
<tbody>
<?php foreach($results_db as $r): ?>
<tr>
<td><?php echo $r["username"]; ?></td>
<td><?php echo $r["firstname"]. " ".$r["middlename"]. "
".$r["lastname"]; ?></td>
<td><?php echo $r["email"]; ?></td>
<td><?php echo $r["id_number"]; ?></td>
<td class="text-center"><?php echo $r["id_number"]; ?></td>
<td class="text-center"><a href="<?php echo
base_url('tech/dilnetacc_tools/view_account/'. $r['username']); ?>"><i class="fa
fa-file-o fa-eye" data-toggle="tooltip" data-placement="auto" data-html="true"
title="View Account"></i></a>
    <a href="<?php echo
base_url('tech/dilnetacc_tools/edit_account_form/'. $r['username']); ?>"
style="margin-left: 5px;"><i class="fa fa-edit" data-toggle="tooltip" data-
placement="auto" data-html="true" title="Edit Account"></i></a>
    <a href="<?php echo
base_url('tech/print_account/confirm_print/'. $r['username']); ?>" style="margin-
left: 5px;"><i class="fa fa-print hidden-xs" data-toggle="tooltip" data-
placement="auto" data-html="true" title="Print Account"></i></a>
    <a href="<?php echo
base_url('tech/merge/merge_add/'. $r['username']); ?>" style="margin-left: 5px;"><i
class="fa fa-search-plus" data-toggle="tooltip" data-placement="auto" data-
html="true" title="Add to Merge"></i></a>
    </td>

```

```

        </tr>
        <?php endforeach; //.results_db ?>

    </tbody>
    <tfoot>
        <tr>
            <th class="text-center">Username</th>
            <th class="text-center">Name</th>
            <th class="text-center">Email</th>
            <th class="text-center">ID Number</th>
            <th class="text-center">Actions</th>
        </tr>
    </tfoot>
</table>
</div><!-- /.box-body -->
</div><!-- /.box -->
</div><!-- /.row -->
<?php endif; //isset.results ?>

<?php if(empty($results_db) && isset($results_ldap) && !empty($results_ldap)):
?>
    <div class="row">
        <div class="col-xs-12">
            <div class="box">
                <div class="box-header">
                    <h3 class="box-title">LDAP Search Results <?php if(isset($search_str)){
echo "for '<strong>".$search_str."</strong>'"; } ?></h3>
                </div><!-- /.box-header -->
                <div class="box-body">
                    <table id="acct_search_results" class="table table-bordered table-
hover">
                        <thead>
                            <tr>
                                <th>Username</th>
                                <th>Name</th>
                                <th>Email</th>
                                <th>Student Number</th>
                                <th>Employee Number</th>
                                <th class="text-center">Actions</th>
                            </tr>
                        </thead>
                        <tbody>
                            <?php if (is_array($results_ldap) || is_object($results_ldap)): ?>
                                <?php foreach($results_ldap as $key => $r): ?>
                                    <tr>
                                        <td><?php echo $r["uid"][0]; ?></td>
                                        <td><?php echo $r["cn"][0] ?></td>
                                        <td><?php echo $r["mail"][0]; ?></td>
                                        <td><?php echo $r["studentnumber"][0]); { echo
                                        <td><?php if (isset($r["studentnumber"][0])) { echo
                                        <td><?php if (isset($r["employeenumber"][0])) { echo
                                        <td><?php if (isset($r["employeenumber"][0])) { echo
                                        <td class="text-center"><a href="<?php echo
                                        base_url('supertech/ldap_lookup/view_account/'.$r['uid'][0]); ?>"><i class="fa fa-
                                        eye"></i></a>
                                        <a href="<?php echo base_url('supertech/ldap_lookup/'); ?>
                                        style="margin-left: 30px;"><i class="fa fa-pencil"></i></a>
                                    </td>
                                </tr>
                            </tbody>
                        </table>
                    </div>
                </div>
            </div>
        </div>
    </div>

```



```

        <?php endforeach; //.results_ldap ?>
    <?php endif; //.isarray.results_ldap ?>
    </tbody>
    <tfoot>
        <tr>
            <th>Username</th>
            <th>Name</th>
            <th>Email</th>
            <th>Student Number</th>
            <th>Employee Number</th>
            <th>Actions</th>
        </tr>
    </tfoot>
</table>
</div><!-- /.box-body -->
</div><!-- /.box -->
</div><!-- /.row -->
<?php endif; //isset.results_ldap?>

```

application/views/flash_message.php

```

<?php if(isset($_SESSION['error'])): ?>
<div class="callout callout-danger">
    <p><?php echo $_SESSION['error']; ?></p>
</div>
<?php endif; ?>

<?php if(isset($_SESSION['info'])): ?>
<div class="callout callout-info">
    <p><?php echo $_SESSION['info']; ?></p>
</div>
<?php endif; ?>

<?php if(isset($_SESSION['warning'])): ?>
<div class="callout callout-warning">
    <p><?php echo $_SESSION['warning']; ?></p>
</div>
<?php endif; ?>

<?php if(isset($_SESSION['success'])): ?>
<div class="callout callout-success">
    <p><?php echo $_SESSION['success']; ?></p>
</div>
<?php endif; ?>

```

application/views/host_address_view.php

```

<div class="row">
    <div class="col-xs-12">
        <!-- Horizontal Form -->
        <div class="box box-info">
            <div class="box-header with-border">
                <h3 class="box-title"><i class="fa fa-file-text-o"></i> Log Entry:
            </div><!-- /.box-header -->
            <div class="box-body">
                <strong><?php echo $block_host ?></strong>
            </div>
        </div>
    </div>

```

```

<?php if(isset($results_db)): ?>
<div class="row">
    <div class="col-xs-12">
        <div class="nav-tabs-custom">
            <?php
                $attributes = array('class' => 'form-horizontal', 'role' => 'form',
'id' => 'mergeForm');
                echo form_open('tech/mail_tools/view_host_id', $attributes);
            ?>

            <table class="table table-hover table-striped">
                <thead style="letter-spacing:1px;">
                    <th style="text-align: right">Attribute</th>
                    <th width="65%">Value</th>
                </tr>
            </thead>
            <tbody>
                <!-- Address -->
                <tr>
                    <th style="text-align: right">Host IP Address</th>
                    <td class="col-md-4" align="left">
                        <?php echo $results_db['ip_address']; ?>
                    </td>
                </tr>

                <!-- Status -->
                <tr>
                    <th style="text-align: right">Status</th>
                    <td class="col-md-4" align="left">
                        <?php echo $results_db['disabled'] == TRUE ? "<strong><span
class=\"text-danger\">BLOCKED</span></strong>" : "<span class=\"text-
success\"><strong>Allowed</strong></span>" ?>
                    </td>
                </tr>

                <!-- Date Banned -->
                <tr>
                    <th style="text-align: right" class="<?php echo
$results_db['disabled'] == FALSE ? 'text-muted': ''>">Date Banned</th>
                    <td class="col-md-4" align="left">
                        <?php if(!$results_db['disabled']){ ?>
                            <?php if(!$results_db['disabled']){ ?>
                                <span class="text-muted"><?php echo
$results_db['date_banned'] ?></span>
                                <?php } else {
                                    echo $results_db['date_banned']!="0000-00-00 00:00:00" ?
                                    <span class="text-muted\">No Data</span>;
                                } ?>
                            </td>
                        </tr>

                        <!-- Reason Banned -->
                        <tr>
                            <th style="text-align: right" class="<?php echo
$results_db['disabled'] == FALSE ? 'text-muted': ''>">Reason for Ban</th>
                            <td class="col-md-4" align="left">
                                <?php if(!$results_db['disabled']){ ?>
                                    <span class="text-muted"><?php echo
$results_db['reason_banned'] ?></span>
                                } ?>
                            </td>
                        </tr>
                    </tbody>
                </table>
            </div>
        </div>
    </div>
</div>

```

```

                echo $results_db['reason_banned']!=NULL ?
$results_db['reason_banned'] : "<span class=\"text-muted\">No Data</span>";
            } ??
        </td>
    </tr>

    <!-- Admin Banner -->
    <tr>
        <th style="text-align: right" class="<?php echo
$results_db['disabled'] == FALSE ? 'text-muted': ''?>">Banned by</th>
        <td class="col-md-4" align="left">
            <?php if(!$results_db['disabled']){ ??
                <span class="text-muted"><?php echo
$results_db['tech_banner'] ?></span>
            <?php } else {
                echo $results_db['tech_banner']!=NULL ?
$results_db['tech_banner'] : "<span class=\"text-muted\">No Data</span>";
            } ??
        </td>
    </tr>

    <!-- Date UnBanned -->
    <tr>
        <th style="text-align: right" class="<?php echo
$results_db['disabled'] == TRUE ? 'text-muted': ''?>">Date UnBanned</th>
        <td class="col-md-4" align="left" >
            <?php if($results_db['disabled']){ ??
                <span class="text-muted"><?php echo
!empty($results_db['date_unbanned']) ? $results_db['date_unbanned'] : "<span
class=\"text-muted\">Never</span>"; ??</span>
            <?php } else {
                echo !empty($results_db['date_unbanned']) ?
$results_db['date_unbanned'] : "<span class=\"text-muted\">Never</span>";
            } ??
        </td>
    </tr>

    <!-- Reason UnBanned -->
    <tr>
        <th style="text-align: right" class="<?php echo
$results_db['disabled'] == TRUE ? 'text-muted': ''?>">Reason for UnBan</th>
        <td class="col-md-4" align="left">
            <?php if($results_db['disabled']){ ??
                <span class="text-muted"><?php echo
$results_db['reason_unbanned']!=NULL ? $results_db['reason_unbanned'] : "<span
class=\"text-muted\">No Data</span>"; ??</span>
            <?php } else {
                echo $results_db['reason_unbanned']!=NULL ?
$results_db['reason_unbanned'] : "<span class=\"text-muted\">No Data</span>";
            } ??
        </td>
    </tr>

    <!-- Admin Banner -->
    <tr>
        <th style="text-align: right" class="<?php echo
$results_db['disabled'] == TRUE ? 'text-muted': ''?>">UnBanned by</th>
        <td class="col-md-4" align="left">
            <?php if($results_db['disabled']){ ??

```

```

        <span class="text-muted"><?php echo
$results_db['tech_unbanner']!=NULL ? $results_db['tech_unbanner'] : "<span
class=\"text-muted\">No Data</span>"; ?></span>
        <?php } else {
            echo $results_db['tech_unbanner']!=NULL ?
$results_db['tech_unbanner'] : "<span class=\"text-muted\">No Data</span>";
        } ?>
    </td>
</tr>
</tbody>
</table>
<div class="box-footer">
    <div class="row">
        <div class="col-xs-4 col-sm-4">
            <a class="btn btn-app" href="<?php echo
site_url('tech/mail_tools/host_blocking'); ?>" data-toggle="tooltip" data-
placement="auto" data-html="true" title="Go Back"><i class="fa fa-arrow-left text-
info"></i> Go Back</a>
        </div>
        <div class="col-xs-offset-2 col-sm-offset-4">
            <a class="btn btn-app" href="<?php echo
$results_db['disabled'] ?
site_url('tech/mail_tools/unblock_host_confirm/'.urlencode(base64_encode($results_
db['ip_address']))) :
site_url('tech/mail_tools/block_host_confirm/'.urlencode(base64_encode($results_db
['ip_address']))) ?>" data-toggle="tooltip" data-placement="auto" data-html="true"
title="<?php echo $results_db['disabled'] ? "Unblock" : "Block" ?> IP Address"><i
class="fa fa-user-times <?php echo $results_db['disabled'] ? "text-info" : "text-
danger" ?>"></i> <?php echo $results_db['disabled'] ? "Unblock" : "Block" ?>
IP</a>
        </div>
    </div>
    <?php echo form_close(); ?>
</div>
</div> <!-- /div.table-responsive -->
</div><!-- /div.nav-tabs-custom -->
</div><!-- /div.col-xs-12 -->
</div><!-- /div.row -->
<?php endif; //isset.results ?>

```

application/views/host_blocking_search_view.php

```

<div class="row">
    <div class="col-xs-12">
        <!-- Horizontal Form -->
        <div class="box box-info">
            <div class="box-header with-border">
                <h3 class="box-title"><i class="fa fa-search"></i> Search Blocked Hosts
DB</h3>
                <br><h4>If you know the Host IP address, type it below. Or type part of it
to find it.
                <br>Then press the <i class="fa fa-user-plus text-success"></i> to <span
class="text-success"><strong>unblock</strong></span>, or <i class="fa fa-user-
times text-danger"></i> to <span class="text-danger"><strong>block</strong></span>
the Host IP address.</h4>
            </div><!-- /.box-header -->

```

```

<!-- form start -->
<?php echo form_open('tech/mail_tools/host_blocking','class="form-
horizontal"'); ?>
<div class="box-body">
<div class="form-group <?php echo (form_error('host_block_search_str')
== '') ? '' : 'has-error'; ?>">
<div class="col-xs-12">
<?php echo form_error('host_block_search_str'); ?>
<div class="input-group">
<div class="input-group-addon"><i class="glyphicon glyphicon-
search"></i></div>
<input type="text" id="host_block_search_str" class="form-control
input-lg" name="host_block_search_str" placeholder="Enter IP address X.X.X.X"
required autofocus>
</div><!-- /.input-group -->
</div>
</div>
</div><!-- /.box-body -->
<div class="box-footer">
<button type="submit" id="hostBlockSubmitButton" onclick="return
changeHostBlockSearchText();" class="btn btn-info col-xs-12">Search IP
Address</button>
</div><!-- /.box-footer -->
<?php echo form_close(); ?>
</div><!-- /.row -->
</div><!-- /.box -->
</div><!-- /.col-md-6 -->

<?php if(!empty($host_blocking_results_db)): ?>
<?php /* print "<pre>"; print_r($results_db); print "</pre>"; */ ?>
<div class="row">
<div class="col-xs-12">
<div class="box">
<div class="box-header">
<h3 class="box-title">DB Search Results <?php
if(isset($host_blocking_search_string)){ echo "for
'<strong>".$host_blocking_search_string."</strong>"; } ?></h3>
</div><!-- /.box-header -->
<div class="box-body">
<table id="log_search_results" class="table table-bordered table-hover">
<thead>
<tr>
<th class="text-center">Host</th>
<th class="text-center">Status</th>
<th class="text-center hidden-xs">Date Banned</th>
<th class="text-center">Admin</th>
<th class="text-center">Actions</th>
</tr>
</thead>
<tbody>
<?php foreach($host_blocking_results_db as $key => $r): ?>
<tr>
<td width="30%" class="text-center"><?php echo $r["ip_address"]
?></td>
<td class="text-center"><?php echo $r["disabled"]==TRUE ? "<span
class='text-danger'><strong>BLOCKED</strong></span>": "<span class='text-
success'>ALLOWED</span>" ?></td>
<td class="text-center hidden-xs"><?php echo $r["date_banned"]
?></td>

```



```

        <input type="hidden" class="form-control" name="block_host" value="<?php
echo $block_host; ?>"
        <?php echo form_error('disable_reason'); ?>
    </div>
</div>
</div><!-- /.box-body -->
<div class="box-footer">
    <div class="row">
        <div class="col-xs-5 col-sm-4 col-md-3">
            <button type="submit" class="btn btn-lg <?php echo $host['disabled'] ?
"btn-success" : "btn-danger" ?>"><i class="fa <?php echo $host['disabled'] ? 'fa-
unlock' : 'fa-lock' ?>"></i> <?php echo $host['disabled'] ? "Unblock" : "Block"?>
IP</button>
        </div>
        <div class="col-xs-offset-7 col-sm-offset-4 col-md-offset-2">
            <a class="btn btn-lg btn-default" href="<?php echo
site_url('tech/mail_tools/'. $link_referer); ?>" data-toggle="tooltip" data-
placement="auto" data-html="true" title="Go Back"><i class="fa fa-arrow-left"></i>
Go Back</a>
            <?php echo form_close(); ?>
        </div>
    </div><!-- /.row -->
</div><!-- div.box-footer -->
</div><!-- div.box -->

```

application/views/login.php

```

<!DOCTYPE html>
<html>
    <head>
        <meta charset="utf-8">
        <meta http-equiv="X-UA-Compatible" content="IE=edge">
        <meta http-equiv="X-UA-Compatible" content="IE=edge">
        <title>tech.accounts | Log in</title>
        <!-- Tell the browser to be responsive to screen width -->
        <meta content="width=device-width, initial-scale=1, maximum-scale=1, user-
scalable=no" name="viewport">
        <!-- Bootstrap 3.3.5 -->
        <link rel="stylesheet" href="<?php echo $this->config->item('theme_url');
?>/bootstrap/css/bootstrap.min.css">
        <!-- Font Awesome -->
        <link rel="stylesheet" href="<?php echo $this->config->item('theme_url');
?>/font-awesome-4.5.0/css/font-awesome.min.css">
        <!-- Ionicons -->
        <link rel="stylesheet" href="<?php echo $this->config->item('theme_url');
?>/ionicons-2.0.1/css/ionicons.min.css">
        <!-- Theme style -->
        <link rel="stylesheet" href="<?php echo $this->config->item('theme_url');
?>/dist/css/AdminLTE.min.css">
        <!-- iCheck -->
        <link rel="stylesheet" href="<?php echo $this->config->item('theme_url');
?>/plugins/iCheck/square/blue.css">
        <!-- HTML5 Shim and Respond.js IE8 support of HTML5 elements and media queries
-->
        <!-- WARNING: Respond.js doesn't work if you view the page via file:// -->
        <!--[if lt IE 9]>
            <script
src="https://oss.maxcdn.com/html5shiv/3.7.3/html5shiv.min.js"></script>

```

```

<script
src="https://oss.maxcdn.com/respond/1.4.2/respond.min.js"></script>
<![endif]-->
</head>
<body class="hold-transition login-page">
<div class="login-box">
<div class="login-logo">
<b>tech.</b>accounts</a>
</div><!-- /.login-logo -->
<div class="login-box-body">
<?php if($this->session->flashdata('logout') == TRUE): ?>
<div class="alert alert-success">
<h4><i class="icon fa fa-success"></i>Log Out Successful.</h4>
</div>
<?php endif; //$this->session->flashdata('logout') ?>
<p class="login-box-msg">Sign in to start your session
</p>
<?php echo form_open('welcome/login'); ?>
<div class="form-group has-feedback <?php echo (form_error('username')
== '') ? '' : 'has-error'; ?>">
<input type="text" name="username" class="form-control"
placeholder="Username">
<span class="fa fa-user form-control-feedback"></span>
<?php echo form_error('username'); ?>
</div>
<div class="form-group has-feedback <?php echo (form_error('password')
== '') ? '' : 'has-error'; ?>">
<input type="password" name="password" class="form-control"
placeholder="Password">
<span class="fa fa-key form-control-feedback"></span>
<?php echo form_error('password'); ?>
</div>
<div class="row">
<div class="col-xs-12">
<button type="submit" class="btn btn-primary btn-block btn-
flat">Sign In <i class="fa fa-sign-in"></i></button>
</div><!-- /.col -->
</div>
<?php echo form_close(); ?>
</div><!-- /.login-box-body -->
</div><!-- /.login-box -->

<!-- jQuery 2.1.4 -->
<script src="<?php echo $this->config->item('theme_url');
?>/plugins/jquery/jquery-2.1.4.min.js"></script>
<!-- Bootstrap 3.3.5 -->
<script src="<?php echo $this->config->item('theme_url');
?>/bootstrap/js/bootstrap.min.js"></script>
<!-- iCheck -->
<script src="<?php echo $this->config->item('theme_url');
?>/plugins/iCheck/ichack.min.js"></script>
<script>
$(function () {
$('input').iCheck({
checkboxClass: 'icheckbox_square-blue',
radioClass: 'iradio_square-blue',
increaseArea: '20%' // optional
});
$('.login-page').hide().fadeIn(1500);

```



```

    });
  </script>
</body>
</html>

```

application/views/logout.php

```

<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <title>tech.accounts | Log in</title>
    <!-- Tell the browser to be responsive to screen width -->
    <meta content="width=device-width, initial-scale=1, maximum-scale=1, user-
scalable=no" name="viewport">
    <!-- Bootstrap 3.3.5 -->
    <link rel="stylesheet" href="<?php echo $this->config->item('theme_url');
?>/bootstrap/css/bootstrap.min.css">
    <!-- Font Awesome -->
    <link rel="stylesheet" href="<?php echo $this->config->item('theme_url');
?>/font-awesome-4.5.0/css/font-awesome.min.css">
    <!-- Ionicons -->
    <link rel="stylesheet" href="<?php echo $this->config->item('theme_url');
?>/ionicons-2.0.1/css/ionicons.min.css">
    <!-- Theme style -->
    <link rel="stylesheet" href="<?php echo $this->config->item('theme_url');
?>/dist/css/AdminLTE.min.css">
    <!-- iCheck -->
    <link rel="stylesheet" href="<?php echo $this->config->item('theme_url');
?>/plugins/iCheck/square/blue.css">

    <!-- HTML5 Shim and Respond.js IE8 support of HTML5 elements and media queries
-->
    <!-- WARNING: Respond.js doesn't work if you view the page via file:// -->
    <!--[if lt IE 9]>
      <script
src="https://oss.maxcdn.com/html5shiv/3.7.3/html5shiv.min.js"></script>
      <script
src="https://oss.maxcdn.com/respond/1.4.2/respond.min.js"></script>
    <![endif]>
  </head>
  <body class="hold-transition login-page">
    <div class="login-box">
      <div class="login-logo">
        <b>tech.</b>accounts</a>
      </div><!-- /.login-logo -->
      <div class="login-box-msg text-green"><i class="icon fa fa-check"
<h4 class="login-box-msg">Log Out Successful.</h4>
style="margin-right: 5px;"></i>Log Out Successful.</h4>
      <p class="login-box-msg">
        Click <a href="<?php echo base_url('welcome/login'); ?>">here</a> to sign
in again.</p>
    </div><!-- /.login-box-body -->
  </div><!-- /.login-box -->

  <!-- jQuery 2.1.4 -->

```

```

<script src="<?php echo $this->config->item('theme_url');
?>/plugins/jquery/jquery-2.1.4.min.js"></script>
<!-- Bootstrap 3.3.5 -->
<script src="<?php echo $this->config->item('theme_url');
?>/bootstrap/js/bootstrap.min.js"></script>
<!-- iCheck -->
<script src="<?php echo $this->config->item('theme_url');
?>/plugins/iCheck/ichack.min.js"></script>
<script>
    $(function () {
        $('.login-page').hide().fadeIn(1500);
    });
</script>
</body>
</html>

```

application/views/mail_account_disable_confirm.php

```

<div class="box box-solid" style="padding-left: 15px;">
<div class="box-body">
    <h4>You are going to <strong><?php echo $email['disabled'] ? "unblock" :
"block" ?></strong> this email address. <br /><br><?php echo $email['disabled'] ?
"When unblocked, the email address will be able to send and receive emails again.
When blocked, mails originating from and destined to this address
will be rejected. <br><br>" : "When blocked, mails originating from and destined to this address
will be rejected. <br><br>" ?>
    <span class="text-danger">REVIEW</span> the information below before you
continue.</h4>
    <hr>
    <div>
        <h4><?php echo $email['disabled'] ? "Unblock" : "Block" ?>: <strong><?php
echo base64_decode(urldecode($block_email)); ?></strong></h4>
        <?php echo $email['disabled'] ?
form_open('tech/mail_tools/unblock_email_confirm') :
form_open('tech/mail_tools/block_email_confirm'); ?>
        <div class="form-group"><?php echo (form_error('disable_reason') == '') ? ''
: 'has-error'; ?>>
            <label for="disable_reason">Reason</label>
            <input type="text" class="form-control" name="disable_reason"
            value="<?php echo set_value('disable_reason'); ?>" required autofocus>
            <input type="hidden" class="form-control" name="block_email" value="<?php
echo $block_email; ?>">
            <?php echo form_error('disable_reason'); ?>
        </div>
    </div>
</div><!-- /.box-body -->
<div class="box-footer">
    <div class="row">
        <div class="col-xs-5 col-sm-4 col-md-3">
            <button type="submit" class="btn btn-lg"><?php echo $email['disabled'] ?
            "btn-success" : "btn-danger" ?>><i class="fa"><?php echo $email['disabled'] ? "Unblock" : "Block"
            unlock' : 'fa-lock' ?>></i> <?php echo $email['disabled'] ? "Unblock" : "Block"
            ?> Email</button>
        </div>
        <div class="col-xs-offset-7 col-sm-offset-4 col-md-offset-2">
            <a class="btn btn-lg btn-default" href="<?php echo
            site_url('tech/mail_tools/'. $link_referer); ?>" data-toggle="tooltip" data-
            placement="auto" data-html="true" title="Go Back"><i class="fa fa-arrow-left"></i>
            Go Back</a>
        </div>
    </div>

```

```

        <?php echo form_close(); ?>
    </div>
</div><!-- /.row -->
</div><!-- div.box-footer -->
</div><!-- div.box -->

```

```

application/views/mail_address_view.php

```

```

<div class="row">
    <div class="col-xs-12">
        <!-- Horizontal Form -->
        <div class="box box-info">
            <div class="box-header with-border">
                <h3 class="box-title"><i class="fa fa-file-text-o"></i> Log Entry:
            </div><!-- /.box-header -->
            <div class="box-body">
                <strong><?php echo $block_email ?></strong></div>
            </div>
        </div>
        <?php if(isset($results_db)): ?>
        <div class="row">
            <div class="col-xs-12">
                <div class="nav-tabs-custom">
                    <?php
                        $attributes = array('class' => 'form-horizontal', 'role' => 'form',
                        'id' => 'mergeForm');
                        echo form_open('tech/mail_tools/view_message_id', $attributes);
                    ?>

                    <table class="table table-hover table-striped table-responsive">
                        <thead style="letter-spacing:1px;">
                            <th style="text-align: right">Attribute</th>
                            <th width="65%">Value</th>
                        </thead>
                        <tbody>
                            <!-- Address -->
                            <tr>
                                <th style="text-align: right">Email Address</th>
                                <td class="col-md-4" align="left">
                                    <?php echo $results_db['email_address']; ?>
                                </td>
                            </tr>

                            <!-- Status -->
                            <tr>
                                <th style="text-align: right">Status</th>
                                <td class="col-md-4" align="left">
                                    <?php echo $results_db['disabled'] == TRUE ? "<strong><span
                                    class=\"text-danger\">BLOCKED</span></strong>" : "<span class=\"text-
                                    success\"><strong>Allowed</strong></span>" ?>
                                </td>
                            </tr>

                            <!-- Date Banned -->
                            <tr>
                                <th style="text-align: right" class="<?php echo
                                $results_db['disabled'] == FALSE ? 'text-muted': ''>">Date Banned</th>
                                <td class="col-md-4" align="left">
                                    <?php if(!$results_db['disabled']){ ?>

```

```

        <span class="text-muted"><?php echo
$results_db['date_banned'] ?></span>
        <?php } else {
            echo $results_db['date_banned']!="0000-00-00 00:00:00" ?
$results_db['date_banned'] : "<span class=\"text-muted\">No Data</span>";
        } ?>
    </td>
</tr>

<!-- Reason Banned -->
<tr>
    <th style="text-align: right" class="<?php echo
$results_db['disabled'] == FALSE ? 'text-muted': ''?>">Reason for Ban</th>
    <td class="col-md-4" align="left">
        <?php if(!$results_db['disabled']){ ?>
            <span class="text-muted"><?php echo
$results_db['reason_banned'] ?></span>
            <?php } else {
                echo $results_db['reason_banned']!=NULL ?
$results_db['reason_banned'] : "<span class=\"text-muted\">No Data</span>";
            } ?>
        </td>
</tr>

<!-- Admin Banner -->
<tr>
    <th style="text-align: right" class="<?php echo
$results_db['disabled'] == FALSE ? 'text-muted': ''?>">Banned by</th>
    <td class="col-md-4" align="left">
        <?php if(!$results_db['disabled']){ ?>
            <span class="text-muted"><?php echo
$results_db['tech_banner'] ?></span>
            <?php } else {
                echo $results_db['tech_banner']!=NULL ?
$results_db['tech_banner'] : "<span class=\"text-muted\">No Data</span>";
            } ?>
        </td>
</tr>

<!-- Date UnBanned -->
<tr>
    <th style="text-align: right" class="<?php echo
$results_db['disabled'] == TRUE ? 'text-muted': ''?>">Date UnBanned</th>
    <td class="col-md-4" align="left">
        <?php if($results_db['disabled']){ ?>
            <span class="text-muted"><?php echo
$results_db['date_unbanned']!="0000-00-00 00:00:00" ? $results_db['date_unbanned']
: "<span class=\"text-muted\">Never</span>"; ?></span>
            <?php } else {
                echo $results_db['date_unbanned']!="0000-00-00 00:00:00" ?
$results_db['date_unbanned'] : "<span class=\"text-muted\">Never</span>";
            } ?>
        </td>
</tr>

<!-- Reason UnBanned -->
<tr>
    <th style="text-align: right" class="<?php echo
$results_db['disabled'] == TRUE ? 'text-muted': ''?>">Reason for UnBan</th>

```

```
 <?php if($results_db['disabled']){ ?>         <span class="text-muted"><?php echo $results_db['reason_unbanned']!=NULL ? $results_db['reason_unbanned'] : "<span class=\"text-muted\">No Data</span>"; ?></span>         <?php } else {         echo $results_db['reason_unbanned']!=NULL ? $results_db['reason_unbanned'] : "<span class=\"text-muted\">No Data</span>";         } ?>     </td> </tr> <!-- Admin Banner --> <tr>     <th style="text-align: right" class="<?php echo $results_db['disabled'] == TRUE ? 'text-muted': ''>">UnBanned by</th>     <td class="col-md-4" align="left">         <?php if($results_db['disabled']){ ?>         <span class="text-muted"><?php echo $results_db['tech_unbanner']!=NULL ? $results_db['tech_unbanner'] : "<span class=\"text-muted\">No Data</span>"; ?></span>         <?php } else {         echo $results_db['tech_unbanner']!=NULL ? $results_db['tech_unbanner'] : "<span class=\"text-muted\">No Data</span>";         } ?>     </td> </tr> </tbody> </table> <div class="box-footer">     <div class="row">         <div class="col-xs-4 col-sm-4">             <a class="btn btn-app" href="<?php echo site_url('tech/mail_tools/email_blocking'); ?>" data-toggle="tooltip" data- placement="auto" data-html="true" title="Go Back"><i class="fa fa-step-backward text-primary"></i> Go Back</a>         </div>         <div class="col-xs-offset-2 col-sm-offset-4">             <a class="btn btn-app" href="<?php echo $results_db['disabled'] ? site_url('tech/mail_tools/unblock_email_confirm/'.urlencode(base64_encode($results _db['email_address']))) : site_url('tech/mail_tools/block_email_confirm/'.urlencode(base64_encode($results_d b['email_address']))) ?>" data-toggle="tooltip" data-placement="auto" data- html="true" title="<?php echo $results_db['disabled'] ? "Unblock" : "Block" ?> Email Address"><i class="fa fa-user-times"></i> <?php echo $results_db['disabled'] ? "text-success" : "text-danger" ?>></i> <?php echo $results_db['disabled'] ? "Unblock" : "Block" ?> Email</a>             <?php if ($results_db['disabled'] == TRUE): ?>             <a class="btn btn-app" href="<?php echo site_url('tech/mail_tools/remove_queued_emails/'.urlencode(base64_encode($results_ _db['email_address']))) ?>" data-toggle="tooltip" data-placement="auto" data- html="true" title="Remove Queued Emails"><i class="fa fa-inbox text-danger"></i> Remove Email(s)</a>             <?php endif; ?>         </div>     </div> <?php echo form_close(); ?> </div> </div> <!-- /div.table-responsive --> |
```

```

        </div><!-- /div.nav-tabs-custom -->
    </div><!-- /div.col-xs-12 -->
</div><!-- /div.row -->
<?php endif; //isset.results ?>

```

application/views/mail_blocking_search_view.php

```

<div class="row">
    <div class="col-xs-12">
        <!-- Horizontal Form -->
        <div class="box box-info">
            <div class="box-header with-border">
                <h3 class="box-title"><i class="fa fa-search"></i> Search Mail DB</h3>
                <br><h4>If you know the email address, type it below. Or type part of it
to find it.
                <br>Then press the <i class="fa fa-user-plus text-success"></i> to <span
class="text-success"><strong>unblock</strong></span>, or <i class="fa fa-user-
times text-danger"></i> to <span class="text-danger"><strong>block</strong></span>
the email address.</h4>
            </div><!-- /.box-header -->
            <!-- form start -->
            <?php echo form_open('tech/mail_tools/email_blocking', 'class="form-
horizontal"'); ?>
            <div class="box-body">
                <div class="form-group"><?php echo (form_error('mail_block_search_str')
== '') ? '' : 'has-error'; ?>>
                <div class="col-xs-12">
                    <?php echo form_error('mail_block_search_str'); ?>
                    <div class="input-group">
                        <div class="input-group-addon"><i class="glyphicon glyphicon-
search"></i></div>
                        <input type="text" id="mail_block_search_str" class="form-control
input-lg" name="mail_block_search_str" placeholder="Type email or part of it"
required autofocus>
                    </div><!-- /.input-group -->
                </div>
            </div><!-- /.box-body -->
            <div class="box-footer">
                <button type="submit" id="mailBlockSubmitButton" onclick="return
changeMailBlockSearchText();" class="btn btn-info col-xs-12">Search Email
Address</button>
            </div><!-- /.box-footer -->
            <?php echo form_close(); ?>
        </div><!-- /.row -->
    </div><!-- /.box -->
</div><!-- /.col-md-6 -->

<?php if(!empty($email_blocking_results_db)): ?>
<?php /* print "<pre>"; print_r($results_db); print "</pre>"; */ ?>
<div class="row">
    <div class="col-xs-12">
        <div class="box">
            <div class="box-header">
                <h3 class="box-title">DB Search Results <?php
if(isset($email_blocking_search_string)){ echo "for
'<strong>".$email_blocking_search_string."</strong>"; } ?></h3>
            </div><!-- /.box-header -->

```

```

<div class="row">
  <div class="col-xs-12">
    <!-- Horizontal Form -->
    <div class="box box-info">
      <div class="box-header with-border">
        <h3 class="box-title"><i class="fa fa-file-text-o"></i> Log Entry:
<strong><?php echo $mail_ID ?></strong></h3>
      </div><!-- /.box-header -->
    </div>
<?php if(isset($results_db)): ?>
<?php // print "<pre>"; print_r($results_db); print "</pre>"; ?>
    <div class="row">
      <div class="col-xs-12">
        <div class="nav-tabs-custom">
          <?php
            $attributes = array('class' => 'form-horizontal', 'role' => 'form',
id' => 'mergeForm');
            echo form_open('tech/mail_tools/view_message_id', $attributes);
          ?>

          <table class="table table-hover table-striped">
            <thead style="letter-spacing:1px;">
              <th style="text-align: right">Attribute</th>
              <th width="85%">Value</th>
            </tr>
          </thead>
          <tbody>
            <?php foreach($results_db as $key => $r): ?>
              <!-- Mail ID -->
              <tr>
                <th style="text-align: right" class="col-md-4">ID</th>
                <td class="col-md-4">
                  <?php echo "$key"; ?>
                </td>
              </tr>
              <!-- Subject -->
              <tr>
                <th style="text-align: right">Subject</th>
                <td class="col-md-4" align="left">
                  <?php echo $r['subject']; ?>
                </td>
              </tr>
              <!-- Recipient Address -->
              <tr>
                <th style="text-align: right">Recipient</th>
                <td class="col-md-4" align="left">
                  <?php echo $r['recipient']; ?>
                </td>
              </tr>
              <!-- Sender Address -->
              <tr>
                <th style="text-align: right">Sender</th>
                <td class="col-md-4" align="left">
                  <?php echo $r['sender']; ?>
                </td>
              </tr>
            </tbody>
          </table>
        </div>
      </div>
    </div>
  </div>
</div>

```



```

<!-- Sender Address -->
|  |  |  |  |  |  |  |  | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Sender Host</th>  <?php echo ''.str_replace("H=", "", $r['sender_host']).' ['.$r['sender_ip'].']'; ?>     </td> </tr>  <!-- Start TIME --> |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | | Start Time</th>  <?php echo $r['start_time']; ?>     </td> </tr>  <!-- End TIME --> |  |  |  |  | | --- | --- | --- | --- | | End Time</th>  <?php echo $r['end_time']; ?>     </td> </tr>  <!-- Status --> |  |  | | --- | --- | | Status</th>  <?php echo $r['status']; ?>     </td> </tr>  </tbody> </table> <div class="box-footer">     <div class="col-xs-8 col-sm-9 col-md-8 pull-right">         <?php if(filter_var($r['sender'], FILTER_VALIDATE_EMAIL)){ ?>             <a class="btn btn-app" href="<?php echo $r['sender_disabled'] ? site_url('tech/mail_tools/unblock_email_confirm/'.urlencode(base64_encode($r['send er']))) : site_url('tech/mail_tools/block_email_confirm/'.urlencode(base64_encode($r['sender ']))) ?>" data-toggle="tooltip" data-placement="auto" data-html="true" '')) ?>" data-toggle="tooltip" data-placement="auto" data-html="true" title="<?php echo $r['sender_disabled'] ? "Unblock" : "Block" ?> Sender"><i class="fa fa-user-times <?php echo $r['sender_disabled'] ? "Unblock" : "Block" ?> danger" ?>"></i> <?php echo $r['sender_disabled'] ? "Unblock" : "Block" ?> Sender</a>         <?php } ?>         <?php if(filter_var(trim($r['sender_ip'], "["), FILTER_VALIDATE_IP)){ ?>             <a class="btn btn-app" href="<?php echo $r['sender_ip_disabled'] ? site_url('tech/mail_tools/unblock_host_confirm/'.urlencode(base64_encode($r['sende r_ip']))) : site_url('tech/mail_tools/block_host_confirm/'.urlencode(base64_encode($r['sender_ ip']))) ?>" data-toggle="tooltip" data-placement="auto" data-html="true" ip']))) ?>" data-toggle="tooltip" data-placement="auto" data-html="true" | | | | | | | |

```

```

title="<?php echo $r['sender_ip_disabled'] ? "Unblock" : "Block" ?> Sender IP"><i
class="fa fa-user-times <?php echo $r['sender_ip_disabled'] ? "text-info" : "text-
danger" ?>"></i> <?php echo $r['sender_ip_disabled'] ? "Unblock" : "Block" ?>
Sender IP</a>
        <?php } ?>
        <?php if(filter_var($r['recipient'], FILTER_VALIDATE_EMAIL)){ ?>
            <a class="btn btn-app" href="<?php echo
$r['recipient_disabled'] ?
site_url('tech/mail_tools/unblock_email_confirm/'.urlencode(base64_encode($r['reci
ipient']))) :
site_url('tech/mail_tools/block_email_confirm/'.urlencode(base64_encode($r['recipi
ent']))) ?>" data-toggle="tooltip" data-placement="auto" data-html="true"
title="<?php echo $r['recipient_disabled'] ? "Unblock" : "Block" ?> Recipient"><i
class="fa fa-user-times <?php echo $r['recipient_disabled'] ? "text-info" : "text-
danger" ?>"></i> <?php echo $r['recipient_disabled'] ? "Unblock" : "Block" ?>
Recipient</a>
            <?php } ?>
        </div>
        <div class="col-xs-4 col-sm-2 col-md-4">
            <a class="btn btn-app" href="<?php echo
site_url('tech/mail_tools/mail_search'); ?>" data-toggle="tooltip" data-
placement="auto" data-html="true" title="Go Back"><i class="fa fa-arrow-left text-
primary"></i> Go Back</a>
        </div>
        <?php endforeach; //.results_db ?>
        <?php echo form_close(); ?>
    </div>
</div> <!-- /div.table-responsive -->
</div><!-- /div.nav-tabs-custom -->
</div><!-- /div.col-xs-12 -->
</div><!-- /div.row -->
<?php endif; //isset.results ?>

```

application/views/mail_search_view.php

```

<div class="row">
    <div class="col-xs-12">
        <!-- Horizontal Form -->
        <div class="box box-info">
            <div class="box-header with-border">
                <h3 class="box-title"><i class="fa fa-search"></i> Search Mail Logs</h3>
                <br><h4>You may type the sender, recipient, or the subject if you know it.
Or type part of it to find it.</h4>
            </div><!-- /.box-header -->
            <!-- form start -->
            <?php echo form_open('tech/mail_tools/mail_search', 'class="form-
horizontal"> ?>
            <div class="box-body">
                <div class="form-group <?php echo (form_error('mail_search_str') == '')
? '' : 'has-error'; ?>">
                    <div class="col-xs-12">
                        <?php echo form_error('mail_search_str'); ?>
                        <div class="input-group">
                            <div class="input-group-addon"><i class="glyphicon glyphicon-
search"></i></div>
                            <input type="text" class="form-control input-lg"
id="mail_search_str" name="mail_search_str" placeholder="Enter query" required
autofocus>

```

```

        </div><!-- /.input-group -->
    </div>
</div>
</div><!-- /.box-body -->
<div class="box-footer">
    <button type="submit" id="submitbutton" class="btn btn-info col-xs-12"
onclick="return changeText();">Search Logs</button>
</div><!-- /.box-footer -->
<?php echo form_close(); ?>
</div><!-- /.row -->
</div><!-- /.box -->
</div><!-- /.col-md-6 -->

<?php if(!empty($mail_search_results_db)): ?>
<div class="row">
    <div class="col-xs-12">
        <div class="box">
            <div class="box-header">
                <h3 class="box-title">DB Search Results <?php
if(isset($mail_search_string)){ echo "for
'<strong>".$mail_search_string."</strong>"; } ?></h3>
            </div><!-- /.box-header -->
            <!-- for big screens -->
            <div class="box-body hidden-xs hidden-sm hidden-md">
                <table id="mail_search_results" class="table table-bordered table-hover
table-responsive">
                    <thead>
                        <tr>
                            <th>ID</th>
                            <th>Sender</th>
                            <th>Recipient</th>
                            <th>Status</th>
                            <th class="text-center">Actions</th>
                        </tr>
                    </thead>
                    <tbody>
                        <?php foreach($mail_search_results_db as $key => $r): ?>
                            <tr>
                                <td><a href="<?php echo
base_url('tech/mail_tools/view_message_id/'.$key.''); ?>"><?php echo
$key;?></a></td>
                                <td title="<?php echo $r["sender"] ?>"><?php echo $r["sender"]
?></td>
                                <td title="<?php echo $r["recipient"] ?>"><?php echo $r["recipient"]
?></td>
                                <td><?php echo $r["status"] ?></td>
                                <td class="text-center"><a href="<?php echo
base_url('tech/mail_tools/view_message_id/'.$key.''); ?>"><i class="fa fa-file-o
fa-eye" data-toggle="tooltip" data-placement="auto" data-html="true" title="View
Log Entry"></i></a>
                                <?php if(filter_var($r['sender'],FILTER_VALIDATE_EMAIL)){ ?>
                                    <a href="<?php echo
base_url('tech/mail_tools/block_email_confirm/'.urlencode(base64_encode($r['sender']
''))); ?>" style="margin-left: 3px;" data-toggle="tooltip" data-placement="auto"
data-html="true" title="Block Sender"><i class="fa fa-user-times text-
danger"></i></a>
                                <?php } ?>
                                <!--
                                <?php if(filter_var($r['sender_ip'],FILTER_VALIDATE_IP)){ ?>

```

```

        <a href="<?php echo
base_url('tech/mail_tools/block_host_confirm/'.urlencode(base64_encode($r['sender_
ip'])))"; ?>" style="margin-left: 3px;" data-toggle="tooltip" data-placement="auto"
data-html="true" title="Block Sender IP"><i class="fa fa-lock text-
danger"></i></a>
        <?php } ?>
    -->
    </td>
</tr>
<?php endforeach; //.results_db ?>
</tbody>
<tfoot>
    <tr>
        <th>ID</th>
        <th>Sender</th>
        <th>Recipient</th>
        <th>Status</th>
        <th class="text-center">Actions</th>
    </tr>
</tfoot>
</table>
</div><!-- /.box-body -->
<!-- for small screens -->
<div class="box-body hidden-lg hidden-xl" style="overflow: auto;">
    <table id="mail_search_results" class="table table-bordered table-hover
table-responsive">
        <thead>
            <tr>
                <th>ID</th>
                <th>Sender</th>
                <th>Recipient</th>
                <th>Status</th>
                <th class="text-center">Actions</th>
            </tr>
        </thead>
        <tbody>
            <?php foreach($mail_search_results_db as $key => $r): ?>
                <tr>
                    <td><?php echo $key; ?></td>
                    <td title="<?php echo $r["sender"] ?>"><?php echo $r["sender"]
?></td>
                    <td title="<?php echo $r["recipient"] ?>"><?php echo $r["recipient"]
?></td>
                    <td><?php echo $r["status"] ?></td>
                    <td class="text-center"><a href="<?php echo
base_url('tech/mail_tools/view_message_id/'.$key.''); ?>"><i class="fa fa-file-o
fa-eye" data-toggle="tooltip" data-placement="auto" data-html="true" title="View
Log Entry"></i></a>
                        <?php if(filter_var($r['sender'],FILTER_VALIDATE_EMAIL)){ ?>
                            <a href="<?php echo
base_url('tech/mail_tools/block_email_confirm/'.urlencode(base64_encode($r['sender_
ip'])))"; ?>" style="margin-left: 3px;" data-toggle="tooltip" data-placement="auto"
data-html="true" title="Block Sender"><i class="fa fa-user-times text-
danger"></i></a>
                        <?php }?>
                        <!--
                        <?php if(filter_var($r['sender_ip'],FILTER_VALIDATE_IP)){ ?>
                            <a href="<?php echo
base_url('tech/mail_tools/block_host_confirm/'.urlencode(base64_encode($r['sender_

```



```

        <?php
        switch($key){
            case "username": echo "Username";
            case "fullname": echo "Full Name";
            case "mail": echo "Primary Email";
            case "mailalternateaddress": echo
            case "employeeetype": echo "Employee
            case "employeenumber": echo
            case "studentnumber": echo "Student
            case "ou": echo "Unit"; break;
            case "departmentnumber": echo
            case "designation": echo
            default: echo $key; break;
        } ?>
    </td>
    <?php
    } ?>

    <td align="left"><?php
        if($value == '' || $value == NULL) {
            echo "NULL";
        } else if($key=="firstname" ||
$key=="middlename" || $key=="lastname"){
            continue;
        } else {
            if(is_array($value)){
                $tvalue =
                print_r($tvalue);
            } else {
                echo $value;
            }
        } ?>
    </td>
</tr>
<?php endforeach; ?>

</tbody>
</table>
<div class="box-footer">
    <?php echo form_open('tech/merge/merge_accounts_final'); ?>
    <?php echo form_hidden($info_diff); ?>
    <div class="col-sm-2 col-xs-3 col-lg-push-5 col-md-push-4 col-
sm-push-3 clearfix">
        <button type="submit" class="btn btn-lg btn-primary"
onclick="return confirm('This operation is irreversible. Continue?');"><i
class="fa fa-save"></i> Save</button>
    </div>
    <div class="col-sm-6 col-xs-6 col-lg-5 pull-right ">

```

```

        <a href="<?php echo
base_url('tech/merge/merge_accounts_form/'); ?>" class="btn btn-lg btn-default"><i
class="fa fa-arrow-left"></i> Go Back</a>
    </div>
    <?php echo form_close(); ?>
</div>
<?php endif; ?>
</div>
</div>

```

application/views/merge_accounts_view.php

```

<div class="box box-solid">
    <div class="box-body">
        <h2 style="margin-top: 10px;">Merge Accounts Preparation</h2>
        <h4 style="margin-top: 10px;">Please select the values that will be retained.
        <br>For the Alternate Email, multiple addresses can be
        selected.<br><strong>All</strong> fields are <strong>required</strong>.</h4>
    </div><!-- /.box-body -->
</div><!-- div.box -->
<div class="row">
    <div class="col-xs-12">
        <div class="nav-tabs-custom">
            <?php
                $attributes = array('class' => 'form-horizontal', 'role' => 'form',
'id' => 'mergeForm');
                echo form_open('tech/merge/merge_accounts_form', $attributes);
            ?>
            <table class="table table-hover table-striped">
                <thead style="letter-spacing:1px;">
                    <tr>
                        <th></th> <!-- spacer -->
                        <th>Account #1 (<?php echo $first_account_view['Username'];
?></th>
                        <th>Account #2 (<?php echo $second_account_view['Username'];
?></th>
                    </tr>
                </thead>
                <tbody>
                    <!-- USERNAME -->
                    <tr class="<?php echo (form_error('opt_username') == TRUE) ?
'danger' : ''; ?>">
                        <th class="col-md-4">Username <?php echo "<span class='label
label-danger'" . form_error('opt_username')."</span>"; ?></th>
                        <td class="col-md-4">
                            <label>
                                <?php
                                    echo form_radio('opt_username',
$first_account_view["Username"], set_radio('opt_username',
$first_account_view["Username"]));
                                    echo "&nbsp;" . $first_account_view["Username"];
                                ?>
                            </label>
                        </td>
                    </tr>
                </tbody>
            </table>
        </div>
    </div>

```

```

        <?php
            echo form_radio('opt_username',
$second_account_view["Username"], set_radio('opt_username',
$second_account_view["Username"]));
            echo "&nbsp;";.$second_account_view["Username"];
        ?>
    </label>
</td>
</tr>

<!-- FULLNAME -->
<tr class="<?php echo (form_error('opt_fullname') == TRUE) ?
'danger' : ''; ?>"
    <th class="col-md-4">Full Name <?php echo "<span class='label
label-danger'>".form_error('opt_fullname')."</span>";?></th>
    <td class="col-md-4">
        <label style="font-weight: 400">
            <?php
                echo form_radio('opt_fullname',
$first_account_view["Username"], set_radio('opt_fullname',
$first_account_view["Username"]));
                echo "&nbsp;";.$first_account_view["Full Name"];
            ?>
        </label>
    </td>
    <td class="col-md-4">
        <label style="font-weight: 400">
            <?php
                echo form_radio('opt_fullname',
$second_account_view["Username"], set_radio('opt_fullname',
$second_account_view["Username"]));
                echo "&nbsp;";.$second_account_view["Full Name"];
            ?>
        </label>
    </td>
</tr>

<!-- PRIMARY EMAIL -->
<tr class="<?php echo (form_error('opt_mail') == TRUE) ? 'danger'
: ''; ?>"
    <th class="col-md-4">Primary Email <?php echo "<span
class='label label-danger'>".form_error('opt_mail')."</span>";?></th>
    <td class="col-md-4">
        <label style="font-weight: 400">
            <?php
                echo form_radio('opt_mail', $first_account_view["Mail"],
set_radio('opt_mail', $first_account_view["Mail"]));
                echo "&nbsp;";.$first_account_view["Mail"];
            ?>
        </label>
    </td>
    <td class="col-md-4">
        <label style="font-weight: 400">
            <?php
                echo form_radio('opt_mail', $second_account_view["Mail"],
set_radio('opt_mail', $second_account_view["Mail"]));
                echo "&nbsp;";.$second_account_view["Mail"];
            ?>
        </label>
    </td>

```



```

        </td>
    </tr>

    <!-- ALTERNATE EMAILS -->
    <tr class="<?php echo (form_error('mail_alt[]') == TRUE) ?
'danger' : ''; ?>">
        <th class="col-md-4">Alternate Email <?php echo "<span
class='label label-danger'>".form_error('mail_alt[]')."</span>";?></th>
        <td class="col-md-4">
            <?php
                if (is_array($first_account_view["Mail Alt"])){
                    foreach ($first_account_view["Mail Alt"] as
$mail_alt_value) { ?>
                        <label style="font-weight: 400">
                            <?php
                                echo form_checkbox('mail_alt[]', $mail_alt_value,
set_checkbox('mail_alt', $mail_alt_value));
                                echo "&nbsp;".$mail_alt_value;
                            ?>
                        </label>
                        <br>
                    <?php }
                    } ?>
                </td>
                <td class="col-md-4">
                    <?php
                        if (is_array($second_account_view["Mail Alt"])){
                            foreach ($second_account_view["Mail Alt"] as
$mail_alt_value) { ?>
                                <label style="font-weight: 400">
                                    <?php
                                        echo form_checkbox('mail_alt[]', $mail_alt_value,
set_checkbox('mail_alt', $mail_alt_value));
                                        echo "&nbsp;".$mail_alt_value;
                                    ?>
                                </label>
                                <br>
                            <?php }
                            } ?>
                        </td>
                    </tr>

    <!-- EMPLOYEE TYPE -->
    <tr class="<?php echo (form_error('opt_employeetype') == TRUE) ?
'danger' : ''; ?>">
        <th class="col-md-4">Employee Type <?php echo "<span
class='label label-danger'>".form_error('opt_employeetype')."</span>";?></th>
        <td class="col-md-4">
            <label style="font-weight: 400">
                <?php
                    if(empty($first_account_view["Employee Type"])){
                        $display_emp_type1 = "[No Data]";
                    } else {
                        $display_emp_type1 = $first_account_view["Employee
Type"];
                    }
                    echo form_radio('opt_employeetype', $display_emp_type1,
set_radio('opt_employeetype', $display_emp_type1);
                    echo "&nbsp;".$display_emp_type1;

```

```

        ?>
    </label>
</td>
<td class="col-md-4">
    <label style="font-weight: 400">
        <?php
            if(empty($second_account_view["Employee Type"])){
                $display_emp_type2 = "[No Data]";
            } else {
                $display_emp_type2 = $second_account_view["Employee
Type"];
            }
            echo form_radio('opt_employeetype', $display_emp_type2,
set_radio('opt_employeetype', $display_emp_type2));
            echo "&nbsp;".$display_emp_type2;
        ?>
    </label>
</td>
</tr>

<!-- EMPLOYEE NUMBER -->
<tr class="<?php echo (form_error('opt_employeenumber') == TRUE) ?
'danger' : ''; ?>">
    <th class="col-md-4">Employee Number <?php echo "<span
class='label label-danger'>".form_error('opt_employeenumber')."</span>";?></th>
    <td class="col-md-4">
        <label style="font-weight: 400">
            <?php
                if(empty($first_account_view["Employee Number"])){
                    $display_value = "[No Data]";
                } else {
                    $display_value = $first_account_view["Employee Number"];
                }
                echo form_radio('opt_employeenumber', $display_value,
set_radio('opt_employeenumber', $display_value));
                echo "&nbsp;".$display_value;
            ?>
        </label>
    </td>
    <td class="col-md-4">
        <label style="font-weight: 400">
            <?php
                if(empty($second_account_view["Employee Number"])){
                    $display_value = "[No Data]";
                } else {
                    $display_value = $second_account_view["Employee
Number"];
                }
                echo form_radio('opt_employeenumber', $display_value,
set_radio('opt_employeenumber', $display_value));
                echo "&nbsp;".$display_value;
            ?>
        </label>
    </td>
</tr>

<!-- STUDENT NUMBER -->
<tr class="<?php echo (form_error('opt_studentnumber') == TRUE) ?
'danger' : ''; ?>">

```

```

        <th class="col-md-4">Student Number <?php echo "<span
class='label label-danger'">.form_error('opt_studentnumber')."</span>";?></th>
        <td class="col-md-4">
            <label style="font-weight: 400">
                <?php
                    if(empty($first_account_view["Student Number"])){
                        $display_value = "[No Data]";
                    } else {
                        $display_value = $first_account_view["Student Number"];
                    }
                    echo form_radio('opt_studentnumber', $display_value,
set_radio('opt_studentnumber', $display_value));
                    echo "&nbsp;";$.display_value;
                ?>
            </label>
        </td>
        <td class="col-md-4">
            <label style="font-weight: 400">
                <?php
                    if(empty($second_account_view["Student Number"])){
                        $display_value = "[No Data]";
                    } else {
                        $display_value = $second_account_view["Student Number"];
                    }
                    echo form_radio('opt_studentnumber', $display_value,
set_radio('opt_studentnumber', $display_value));
                    echo "&nbsp;";$.display_value;
                ?>
            </label>
        </td>
    </tr>

    <!-- COLLEGE/UNIT -->
    <tr class="<?php echo (form_error('opt_unit') == TRUE) ? 'danger'
: '' ; ?>">
        <th class="col-md-4">Unit <?php echo "<span class='label label-
danger'">.form_error('opt_unit')."</span>";?></th>
        <td class="col-md-4">
            <label style="font-weight: 400">
                <?php
                    if(empty($first_account_view["Unit"])){
                        $display_value = "[No Data]";
                    } else {
                        $display_value = $first_account_view["Unit"];
                    }
                    echo form_radio('opt_unit', $display_value,
set_radio('opt_unit', $display_value));
                    echo "&nbsp;";$.display_value;
                ?>
            </label>
        </td>
        <td class="col-md-4">
            <label style="font-weight: 400">
                <?php
                    if(empty($second_account_view["Unit"])){
                        $display_value = "[No Data]";
                    } else {
                        $display_value = $second_account_view["Unit"];
                    }
                ?>
            </label>
        </td>
    </tr>

```

```

        echo form_radio('opt_unit', $display_value,
set_radio('opt_unit', $display_value));
        echo "&nbsp;". $display_value;
    ?>
</label>
</td>
</tr>

<!-- DEPARTMENT -->
<tr class="<?php echo (form_error('opt_department') == TRUE) ?
'danger' : ''; ?>"
    <th class="col-md-4">Department <?php echo "<span class='label
label-danger'>".form_error('opt_department')."</span>";?></th>
    <td class="col-md-4">
        <label style="font-weight: 400">
            <?php
                if(empty($first_account_view["Department"])){
                    $display_value = "[No Data]";
                } else {
                    $display_value = $first_account_view["Department"];
                }
            echo form_radio('opt_department', $display_value,
set_radio('opt_department', $display_value));
            echo "&nbsp;". $display_value;
        ?>
        </label>
    </td>
    <td class="col-md-4">
        <label style="font-weight: 400">
            <?php
                if(empty($second_account_view["Department"])){
                    $display_value = "[No Data]";
                } else {
                    $display_value = $second_account_view["Department"];
                }
            echo form_radio('opt_department', $display_value,
set_radio('opt_department', $display_value));
            echo "&nbsp;". $display_value;
        ?>
        </label>
    </td>
</tr>

<!-- DESIGNATION -->
<tr class="<?php echo (form_error('opt_designation') == TRUE) ?
'danger' : ''; ?>"
    <th class="col-md-4">Designation <?php echo "<span class='label
label-danger'>".form_error('opt_designation')."</span>";?></th>
    <td class="col-md-4">
        <label style="font-weight: 400">
            <?php
                if(empty($first_account_view["Designation"])){
                    $display_value = "[No Data]";
                } else {
                    $display_value = $first_account_view["Designation"];
                }
            echo form_radio('opt_designation', $display_value,
set_radio('opt_designation', $display_value));
            echo "&nbsp;". $display_value;
        ?>
        </label>
    </td>
</tr>

```

```

        ?>
    </label>
</td>
<td class="col-md-4">
    <label style="font-weight: 400">
        <?php
            if(empty($second_account_view["Designation"])){
                $display_value = "[No Data]";
            } else {
                $display_value = $second_account_view["Designation"];
            }
            echo form_radio('opt_designation', $display_value,
set_radio('opt_designation', $display_value));
            echo "&nbsp;". $display_value;
        ?>
    </label>
</td>
</tr>
</tbody>
</table>
<div class="box-footer">
    <div class="col-sm-6 pull-right">
        <!--
        <button type="reset" id="resetMemberForm" name="reset"
class="btn btn-default top-padding"><i class="fa fa-eraser"></i> Reset</button>
        -->
        <a href="<?php echo base_url('tech/merge'); ?>" class="btn btn-
default btn-lg top-padding"><i class="fa fa-arrow-left"></i> Go Back</a>
    </div>
    <div class="col-sm-2 col-xs-1 col-lg-push-4 col-md-push-4 col-sm-
push-4 clearfix">
        <button type="submit" id="bsave" name="save" class="btn btn-
primary btn-lg top-padding" value="merge"><i class="fa fa-clone"></i>
Continue</button>
        <?php echo form_hidden("username1",
$first_account_view['Username']); ?>
        <?php echo form_hidden("username2",
$second_account_view['Username']); ?>
        <?php echo form_close(); ?>
    </div>
</div>
</div> <!-- /div.table-responsive -->
</div><!-- /div.nav-tabs-custom -->
</div><!-- /div.col-xs-12 -->
</div><!-- /div.row -->

```

application/views/merge_view.php

```

<div class="row">
    <div class="col-xs-12">
        <!-- Horizontal Form -->
        <div class="box box-info">
            <div class="box-header with-border">
                <h3 class="box-title"><i class="fa fa-clone"></i> Merge DilNet
Accounts</h3>
                <br><h4>To merge accounts:
                <ol>

```

```

        <li>Find their usernames. If you know the username, type it below. Or
type part of it to find it.</li>
        <li>Add to merge list by pressing the <i class="fa fa-search-plus
text-info"></i>.</li>
        <li>When there are two accounts on the list, press Proceed with
Merging.</li>
    </ol></h4>
</div><!-- /.box-header -->
<!-- form start -->
    <?php echo form_open('tech/merge/merge_search_form','class="form-
horizontal"'); ?>
    <div class="box-body">
        <div class="form-group <?php echo (form_error('merge_str') == '') ? '' :
'has-error'; ?>">
            <div class="col-xs-12">
                <?php echo form_error('merge_str'); ?>
                <div class="input-group">
                    <div class="input-group-addon"><i class="glyphicon glyphicon-
search"></i></div>
                    <input type="text" id="merge_search_field" class="form-control
input-lg" name="merge_str" placeholder="Name, Username, Email, SN xxxx-xxxxx, or
Employee number" required autofocus>
                </div><!-- /.input-group -->
            </div>
        </div>
    </div><!-- /.box-body -->
    <div class="box-footer">
        <button type="submit" id="mergeSubmitButton" class="btn btn-info col-xs-
12" onclick="return changeMergeSearchText();">Search</button>
    </div><!-- /.box-footer -->
    <?php echo form_close(); ?>
</div><!-- /.row -->
</div><!-- /.box -->
</div><!-- /.col-md-6 -->
<?php // print "<pre>"; print_r($_SESSION['merge_db']); print "</pre>"; ?>
<?php if(isset($_SESSION['merge_db']) && !empty($_SESSION['merge_db'])): ?>
<div class="row">
    <div class="col-xs-12">
        <div class="box">
            <div class="box-header">
                <h3 class="box-title">Merge Accounts List </h3>
                <?php if(isset($_SESSION['merge_db']) && !empty($_SESSION['merge_db']) &&
count($_SESSION['merge_db'])==2){ ?>
                    &nbsp;<div class="label label-warning">Limit of 2 accounts
reached.</div>
                <?php } else { ?>
                    &nbsp;<div class="label label-default">Need 1 more.</div>
                <?php } ?>
            </div><!-- /.box-header -->
            <div class="box-body hidden-xs hidden-sm hidden-md">
                <table id="merge_search_results" class="table table-bordered table-hover
table-responsive">
                    <thead>
                        <tr>
                            <th>Username</th>
                            <th>Name</th>
                            <th>Email</th>
                            <th>ID Number</th>

```

```

        <th class="text-center">Actions</th>
    </tr>
</thead>
<tbody>
    <?php foreach($_SESSION['merge_db'] as $r): ?>
        <tr>
            <td><?php echo $r[0]["uid"][0]; ?></td>
            <td><?php echo $r[0]["cn"][0]; ?></td>
            <td><?php echo $r[0]["mail"][0]; ?></td>
            <td><?php if(isset($r[0]["employeenumber"]) &&
!empty($r[0]["employeenumber"])){
                if(!empty($r[0]["employeenumber"][0])){
                    echo $r[0]["employeenumber"][0];
                } else {
                    echo "NULL";
                }
            } else if(isset($r[0]["studentnumber"]) &&
!empty($r[0]["studentnumber"])){
                if(!empty($r[0]["studentnumber"][0])){
                    echo $r[0]["studentnumber"][0];
                } else {
                    echo "NULL";
                }
            } else {
                echo "NULL";
            }
        ?>
            </td>
            <td class="text-center">
                <a href="<?php echo
base_url('tech/dilnetacc_tools/view_account/'. $r[0]['uid'][0]); ?>"><i class="fa
fa-file-o fa-eye" data-toggle="tooltip" data-placement="auto" data-html="true"
title="View Account"></i></a>
                <a href="<?php echo
base_url('tech/dilnetacc_tools/edit_account_form/'. $r[0]['uid'][0]); ?>"
style="margin-left: 5px;"><i class="fa fa-edit" data-toggle="tooltip" data-
placement="auto" data-html="true" title="Edit Account"></i></a>
                <a href="<?php echo
base_url('tech/merge/merge_remove/'. $r[0]['uid'][0]); ?>" style="margin-left:
5px;"><i class="fa fa-search-minus" data-toggle="tooltip" data-placement="auto"
data-html="true" title="Remove from merge"></i></a>
            </td>
        </tr>
    <?php endforeach; //.results_db ?>
</tbody>
</table>
</div>
<div class="box-body hidden-lg hidden-xl" style="overflow: auto;">
    <table id="merge_search_results" class="table table-bordered table-hover
table-responsive">
        <thead>
            <tr>
                <th>Username</th>
                <th>Name</th>
                <th>Email</th>
                <th>ID Number</th>
                <th class="text-center">Actions</th>
            </tr>

```

```

</thead>
<tbody>
    <?php foreach($_SESSION['merge_db'] as $r): ?>
        <tr>
            <td><?php echo $r[0]["uid"][0]; ?></td>
            <td><?php echo $r[0]["cn"][0]; ?></td>
            <td><?php echo $r[0]["mail"][0]; ?></td>
            <td><?php if(isset($r[0]["employeenumber"]) &&
!empty($r[0]["employeenumber"])){
                if(!empty($r[0]["employeenumber"][0])){
                    echo $r[0]["employeenumber"][0];
                } else {
                    echo "NULL";
                }
            } else if(isset($r[0]["studentnumber"]) &&
!empty($r[0]["studentnumber"])){
                if(!empty($r[0]["studentnumber"][0])){
                    echo $r[0]["studentnumber"][0];
                } else {
                    echo "NULL";
                }
            } else {
                echo "NULL";
            }
        ?>
    </td>
    <td class="text-center">
        <a href="<?php echo
base_url('tech/dilnetacc_tools/view_account/' . $r[0]['uid'][0]); ?>"><i class="fa
fa-file-o fa-eye" data-toggle="tooltip" data-placement="auto" data-html="true"
title="View Account"></i></a>
        <a href="<?php echo
base_url('tech/dilnetacc_tools/edit_account_form/' . $r[0]['uid'][0]); ?>"
style="margin-left: 5px;"><i class="fa fa-edit" data-toggle="tooltip" data-
placement="auto" data-html="true" title="Edit Account"></i></a>
        <a href="<?php echo
base_url('tech/merge/merge_remove/' . $r[0]['uid'][0]); ?>" style="margin-left:
5px;"><i class="fa fa-search-minus" data-toggle="tooltip" data-placement="auto"
data-html="true" title="Remove from merge"></i></a>
    </td>
</tr>
<?php endforeach; //.results_db ?>
</tbody>
</table>
</div>
<div class="box-footer">
    <div class="col-xs-1 col-sm-2 col-md-3 col-lg-2">
        <div class="col-xs-1 col-sm-2 col-md-3 col-lg-2">
            <?php if(count($_SESSION['merge_db'])==2){ ?>
                <?php echo base_url('tech/merge/merge_accounts_form'); ?>
                <a href="<?php echo base_url('tech/merge/merge_accounts_form'); ?>"
role="button" class="btn btn-warning" data-toggle="tooltip" data-placement="auto"
data-html="true" title="Review merge options" ><i class="fa fa-clone"></i> Proceed
With Merging</a>
            <?php } else { ?>
                <a href="#" role="button" class="btn btn-warning" data-
toggle="tooltip" data-placement="auto" data-html="true" title="Add 1 more to
enable" disabled><i class="fa fa-clone"></i> Proceed with Merging</a>
            <?php } ?>
        </div>
    </div>

```



```

<div class="col-xs-offset-8 col-sm-offset-6 col-md-offset-4 col-lg-
offset-3">
    <a href="<?php echo base_url('tech/merge/merge_clear_list'); ?>"
    role="button" class="btn btn-default" data-toggle="tooltip" data-placement="auto"
    data-html="true" title="Clear Merge List">Clear List</a>
</div>
</div><!-- /.box-footer -->

</div><!-- /.box-body -->
</div><!-- /.box -->
</div><!-- /.row -->
<?php endif; //isset.results ?>

<?php if(isset($merge_search_results_db)): ?>
<div class="row">
    <div class="col-xs-12">
        <div class="box">
            <div class="box-header">
                <h3 class="box-title">DB Search Results <?php
if(isset($merge_search_string)){ echo "for
'<strong>".$merge_search_string."</strong>'; } ?></h3>
</div><!-- /.box-header -->
            <div class="box-body hidden-xs hidden-sm hidden-md">
                <table id="acct_search_results" class="table table-bordered table-hover
table-responsive">
                    <thead>
                        <tr>
                            <th>Username</th>
                            <th>Name</th>
                            <th>Email</th>
                            <th>ID Number</th>
                            <th class="text-center">Actions</th>
                        </tr>
                    </thead>
                    <tbody>
                        <?php foreach($merge_search_results_db as $r): ?>
                            <tr>
                                <td><?php echo $r["username"]; ?></td>
                                <td><?php echo $r["firstname"]." ".$r["middlename"]."
                                <td><?php echo $r["lastname"]; ?></td>
                                <td><?php echo $r["email"]; ?></td>
                                <td><?php echo $r["id_number"]; ?></td>
                                <td class="text-center"><a href="<?php echo
base_url('tech/dilnetacc_tools/view_account/' . $r['username']); ?>" style="margin-
left: 5px;"><i class="fa fa-file-o fa-eye" data-toggle="tooltip" data-
placement="auto" data-html="true" title="View Account"></i></a>
                                <a href="<?php echo
base_url('tech/dilnetacc_tools/edit_account_form/' . $r['username']); ?>"
style="margin-left: 5px;"><i class="fa fa-edit" data-toggle="tooltip" data-
placement="auto" data-html="true" title="Edit Account"></i></a>
                                <a href="<?php echo
base_url('tech/print_account/confirm_print/' . $r['username']); ?>" style="margin-
left: 5px;"><i class="fa fa-print hidden-xs" data-toggle="tooltip" data-
placement="auto" data-html="true" title="Print Account"></i></a>
                                <?php if(isset($_SESSION['merge_db']) &&
count($_SESSION['merge_db'])==2){ ?>
                                    &nbsp; <i class="fa fa-search-plus" style="margin-left: 5px; color:
grey" data-toggle="tooltip" data-placement="auto" data-html="true" title=""></i>

```

```

        <?php } else { ?>
            <a href="<?php echo
base_url('tech/merge/merge_add/'. $r['username']); ?>" style="margin-left: 5px;"><i
class="fa fa-search-plus" data-toggle="tooltip" data-placement="auto" data-
html="true" title="Add to Merge"></i></a>
        <?php } ?>
    </td>
</tr>
<?php endforeach; //.results_db ?>
</tbody>
<tfoot>
    <tr>
        <th>Username</th>
        <th>Name</th>
        <th>Email</th>
        <th>ID Number</th>
        <th>Actions</th>
    </tr>
</tfoot>
</table>
</div><!-- /.box-body -->
<div class="box-body hidden-lg hidden-xl" style="overflow: auto;">
    <table id="acct_search_results" class="table table-bordered table-hover
table-responsive">
        <thead>
            <tr>
                <th>Username</th>
                <th>Name</th>
                <th>Email</th>
                <th>ID Number</th>
                <th class="text-center">Actions</th>
            </tr>
        </thead>
        <tbody>
            <?php foreach($merge_search_results_db as $r): ?>
                <tr>
                    <td><?php echo $r["username"]; ?></td>
                    <td><?php echo $r["firstname"]. " ".$r["middlename"]. "
".$r["lastname"]; ?></td>
                    <td><?php echo $r["email"]; ?></td>
                    <td><?php echo $r["id_number"]; ?></td>
                    <td class="text-center"><a href="<?php echo
base_url('tech/dilnetacc_tools/view_account/'. $r['username']); ?>" style="margin-
left: 10px;"><i class="fa fa-file-o fa-eye" data-toggle="tooltip" data-
placement="auto" data-html="true" title="View Account"></i></a>
                    <a href="<?php echo
base_url('tech/dilnetacc_tools/edit_account_form/'. $r['username']); ?>"
style="margin-left: 10px;"><i class="fa fa-edit" data-toggle="tooltip" data-
placement="auto" data-html="true" title="Edit Account"></i></a>
                    <a href="<?php echo
base_url('tech/print_account/confirm_print/'. $r['username']); ?>" style="margin-
left: 5px;"><i class="fa fa-print hidden-xs" data-toggle="tooltip" data-
placement="auto" data-html="true" title="Print Account"></i></a>
                    <?php if(isset($_SESSION['merge_db'])) &&
count($_SESSION['merge_db'])==2){ ?>
                        &nbsp;<i class="fa fa-search-plus" style="margin-left: 5px; color:
grey" data-toggle="tooltip" data-placement="auto" data-html="true" title=""></i>
                    <?php } else { ?>

```

```

        <a href="<?php echo
base_url('tech/merge/merge_add/' . $r['username']); ?>" style="margin-left: 5px;"><i
class="fa fa-search-plus" data-toggle="tooltip" data-placement="auto" data-
html="true" title="Select for Merging"></i></a>
        <?php } ?>
    </td>
</tr>
<?php endforeach; //.results_db ?>
</tbody>
<tfoot>
<tr>
    <th>Username</th>
    <th>Name</th>
    <th>Email</th>
    <th>ID Number</th>
    <th>Actions</th>
</tr>
</tfoot>
</table>
</div><!-- /.box-body -->
</div><!-- /.box -->
</div><!-- /.row -->
</div><!-- /.row -->
<?php endif; //isset.results ?>

```

application/views/page_template.php

```

<!DOCTYPE html>
<html>
    <head>
        <meta charset="utf-8">
        <meta http-equiv="X-UA-Compatible" content="IE=edge">
        <meta http-equiv="X-UA-Compatible" content="IE=edge">
        <title>tech.accounts | <?php echo $page_title; ?></title>
        <link rel="icon" type="image/x-icon" href="<?php echo $this->config-
>item('base_url'); ?>/assets/favicon.ico">
        <!-- Tell the browser to be responsive to screen width -->
        <meta content="width=device-width, initial-scale=1, maximum-scale=1, user-
scalable=no" name="viewport">
        <!-- Bootstrap 3.3.5 -->
        <link rel="stylesheet" href="<?php echo $this->config->item('theme_url');
?>/bootstrap/css/bootstrap.min.css">
        <!-- Font Awesome -->
        <link rel="stylesheet" href="<?php echo $this->config->item('theme_url');
?>/font-awesome-4.5.0/css/font-awesome.min.css">
        <!-- Ionicons -->
        <link rel="stylesheet" href="<?php echo $this->config->item('theme_url');
?>/ionicons-2.0.1/css/ionicons.min.css">

        <!-- Additional Page Stylesheets -->
        <?php if(isset($include_stylesheets)) { $this->load-
view($include_stylesheets); } ?>

        <!-- Theme style -->
        <link rel="stylesheet" href="<?php echo $this->config->item('theme_url');
?>/dist/css/AdminLTE.min.css">
        <!-- AdminLTE Skins. We have chosen the skin-blue for this starter
page. However, you can choose any other skin. Make sure you
apply the skin class to the body tag so the changes take effect.

```

```

-->
<link rel="stylesheet" href="<?php echo $this->config->item('theme_url');
?>/dist/css/skins/skin-purple.min.css">

<!-- HTML5 Shim and Respond.js IE8 support of HTML5 elements and media queries
-->
<!-- WARNING: Respond.js doesn't work if you view the page via file:// -->
<!--[if lt IE 9]>
    <script
src="https://oss.maxcdn.com/html5shiv/3.7.3/html5shiv.min.js"></script>
    <script
src="https://oss.maxcdn.com/respond/1.4.2/respond.min.js"></script>
<![endif]-->
</head>
<!--
BODY TAG OPTIONS:
=====
Apply one or more of the following classes to get the
desired effect

```

SKINS	skin-blue skin-black skin-purple skin-yellow skin-red skin-green
LAYOUT OPTIONS	fixed layout-boxed layout-top-nav sidebar-collapse sidebar-mini

```

-->
<body class="hold-transition skin-purple sidebar-mini">
  <div class="wrapper">

    <?php $this->load->view('partials/header'); ?>

    <?php $this->load->view('partials/left_sidebar'); ?>

    <!-- Content Wrapper. Contains page content -->
    <div class="content-wrapper">
      <!-- Content Header (Page header) -->
      <section class="content-header">
        <h1>
          <?php echo $page_title; ?>
          <small><?php if(isset($page_subtitle)) { echo $page_subtitle; }
?></small>
        </h1>
      </section>

      <!-- Main content -->
      <section class="content">
        <?php $this->load->view('flash_message'); ?>

        <?php $this->load->view($content); ?>

      </section><!-- /.content -->

```

```

</div><!-- /.content-wrapper -->

<?php $this->load->view('partials/footer'); ?>

<?php // $this->load->view('partials/right_sidebar'); ?>
</div><!-- ./wrapper -->
<!-- REQUIRED JS SCRIPTS -->

<!-- jQuery 2.1.4 -->
<script src="<?php echo $this->config->item('theme_url');
?>/plugins/jquery/jquery-2.1.4.min.js"></script>
<!-- Bootstrap 3.3.5 -->
<script src="<?php echo $this->config->item('theme_url');
?>/bootstrap/js/bootstrap.min.js"></script>

<!-- Additional Page Scripts -->
<?php if(isset($include_scripts)) { $this->load->view($include_scripts); } ?>

<!-- AdminLTE App -->
<script src="<?php echo $this->config->item('theme_url');
?>/dist/js/app.min.js"></script>
<script>
$.AdminLTESidebarTweak = {};

$.AdminLTESidebarTweak.options = {
    EnableRemember: true,
    NoTransitionAfterReload: false
    //Removes the transition after page reload.
};

$(function () {
    "use strict";

    $("body").on("collapsed.pushMenu", function(){
        if($.AdminLTESidebarTweak.options.EnableRemember){
            document.cookie = "toggleState=closed";
        }
    });
    $("body").on("expanded.pushMenu", function(){
        if($.AdminLTESidebarTweak.options.EnableRemember){
            document.cookie = "toggleState=opened";
        }
    });

    if($.AdminLTESidebarTweak.options.EnableRemember){
        var re = new RegExp('toggleState' + "=[^;]+)");
        var value = re.exec(document.cookie);
        var toggleState = (value != null) ? unescape(value[1]) : null;
        if(toggleState == 'closed'){
            if($.AdminLTESidebarTweak.options.NoTransitionAfterReload){
                $("body").addClass('sidebar-collapse hold-
transition').delay(100).queue(function(){
                    $(this).removeClass('hold-transition');
                });
            }else{
                $("body").addClass('sidebar-collapse');
            }
        }
    }
}

```

```
});
</script>
```

```
<!-- Optionally, you can add Slimscroll and FastClick plugins.
Both of these plugins are recommended to enhance the
user experience. Slimscroll is required when using the
fixed layout. -->
```

```
</body>
</html>
```

```
application/views/print_confirm.php
```

```
<div class="box">
  <div class="box-body">
    <div id="print" style="font-family: Monospace;">
      <div id="print" style="letter-spacing: 1px;">DILNET ACCOUNT
      <span style="letter-spacing: 1px;">
        <?php if(isset($print_type)) {
          switch($print_type){
            case 'reset':
              echo " : RESET PASSWORD";
              break;
            case 'new':
              echo " : NEW ACCOUNT";
              break;
            case 'info':
              echo " : ACCOUNT INFORMATION";
              break;
          }
        }
      <?>
      <br>-----<br>
    </span>
  </div>
  <table>
    <tbody>
      <?php foreach ($ldap_print as $key => $value): ?>
        <tr>
          <td style="border-bottom: 1px solid gray;
vertical-align: top; padding-right: 5px; "><strong><?php echo $key;
?></strong></td>
          <td style="margin-left: 5px; border-
bottom: 1px solid gray;">
            <?php
              if(is_array($value)){
                foreach($value as $a) { echo
$a."<br>"; }
              }
              else { echo $value; }
            <?>
          </td>
        </tr>
      <?php endforeach; ?>
    </tbody>
  </table>
</div>
<?php echo form_open('tech/print_account/send_to_printer'); ?>
<hr>
<div class="col-xs-12 col-sm-12 col-md-12 col-lg-6 col-xl-6 form-
group <?php echo (form_error('printer_opt') == '') ? '' : 'has-error'; ?>">
```

```

        <h3>Select Printer</h3>
        <select class="form-control select2-printer select2-hidden-
accessible" style="width: 100%;" tabindex="-1" aria-hidden="true"
name="printer_opt">
            <option value="tech-thermal">Tech Printer TM-T82
(Thermal)</option>
            <option value="tech-ribbon">Tech Printer TM-U220
(Ribbon)</option>
            <option value="admin-thermal">Admin Printer TM-T82
(Thermal)</option>
        </select>
        <?php echo form_error('printer_opt'); ?>
    </div>
    <div class="col-xs-8 form-group">
        <h3>Print Messages</h3>
        <div class="checkbox">
            <label>
                <input type="checkbox" name="opt_changepass"
class="minimal" <?php if($print_type == 'reset') { echo "checked"; } ?>>Change
Password<a data-toggle="collapse" href=#collapseChangepass aria-
controls="collapseChangepass" aria-expanded="false"><i class="text-expand fa fa-
plus-square-o" style="margin-left:5px;"></i></a>
            </label>
            <div id="collapseChangepass" class="collapse">
                <p>Please change your password immediately
at</p><h4>https://accounts.upd.edu.ph</h4><p>Or scan this code to go to the
website: [QR Code]</p>
            </div>
        </div>
        <div class="checkbox">
            <label>
                <input type="checkbox" name="opt_facproxy"
class="minimal" <?php if(isset($ldap_print["Emp Type"]) && $ldap_print["Emp
Type"]=='FAC') { echo "checked"; } ?>>Faculty Proxy<a data-toggle="collapse"
href=#collapseFacproxy aria-controls="collapseFacproxy" aria-expanded="false"><i
class="text-expand fa fa-plus-square-o" style="margin-left:5px;"></i></a>
            </label>
            <div id="collapseFacproxy" class="collapse">
                <h4>FACULTY PROXY SETTINGS</h4>
                <p>You may access the unrestricted proxy for the
faculty by using the following settings:<br>
                <strong>Host:</strong>
facultyproxy.upd.edu.ph<br>
                <strong>Port:</strong> 8080<br>
                </p>
            </div>
        </div>
        <div class="checkbox">
            <label>
                <input type="checkbox" name="opt_services"
class="minimal" <?php if($print_type=='reset') { echo "checked"; } ?>>Online
Services<a data-toggle="collapse" href=#collapseServices aria-
controls="collapseServices" aria-expanded="false"><i class="text-expand fa fa-
plus-square-o" style="margin-left:5px;"></i></a>
            </label>
            <div id="collapseServices" class="collapse">
                <h4>ONLINE SERVICES</h4>
                <p>

```

```

        <strong>DILNET ACCOUNT:</strong>
https://accounts.upd.edu.ph<br>
        <strong>UPD WEBMAIL:</strong>
https://mail.upd.edu.ph<br>
        <strong>CRS:</strong> https://crs.upd.edu.ph<br>
        <strong>UVLE:</strong> http://uvle.upd.edu.ph<br>
        <strong>Library (iLib):</strong>
https://ilib.upd.edu.ph
    </p>
</div>
</div>
<div class="checkbox">
    <label><input type="checkbox" name="opt_helpdesk"
class="minimal" <?php if($print_type=='reset') { echo "checked"; } ?>>Dilnet
Helpdesk<a data-toggle="collapse" href=#collapseHelpdesk aria-
controls="collapseHelpdesk" aria-expanded = "false"><i class="text-expand fa fa-
plus-square-o" style="margin-left:5px;"></i></a>
    </label>
    <div id="collapseHelpdesk" class="collapse">
        <h4>DILNET HELPDESK</h4>
        <p>Contact us if you have problems with the ff:<br>
        - Connecting to DILNET Wifi<br>
        - UPD Webmail (mail.upd.edu.ph)<br>
        - Dilnet Account (accounts.upd.edu.ph)<br>
        <strong>Email:</strong> helpdesk@upd.edu.ph<br>
        <strong>Phone:</strong> 981-8500 local 2050<br>
        <strong>Website:</strong>
http://dilnet.upd.edu.ph<br>
        <strong>Facebook:</strong>
http://www.facebook.com/updilnet<br>
        <strong>Twitter:</strong>
http://www.twitter.com/dilnet<br>
    </p>
    </div>
</div>
<div class="checkbox">
    <label><input type="checkbox" name="opt_crs" class="minimal"
    <?php if(isset($ldap_print["Emp Type"]) && $ldap_print["Emp Type"]=='FAC') { echo
"checked"; } ?>>CRS Tagging<a data-toggle="collapse" href=#collapseCrs aria-
controls="collapseCrs" aria-expanded = "false"><i class="text-expand fa fa-plus-
square-o" style="margin-left:5px;"></i></a>
    </label>
    <div id="collapseCrs" class="collapse">
        <h4>CRS FACULTY ROLE</h4>
        <p>We have forwarded your information to CRS for
tagging. Average processing time is 1-2 days. You may follow up this request by
calling <strong>981-8500 local 4560</strong></p>
    </div>
</div>
<div class="checkbox">
    <label><input type="checkbox" name="opt_library"
class="minimal" <?php if(isset($ldap_print["Emp Type"]) && $ldap_print["Emp
Type"]=='FAC') { echo "checked"; } ?>>iLib Access<a data-toggle="collapse"
href=#collapseLibrary aria-controls="collapseLibrary" aria-expanded = "false"><i
class="text-expand fa fa-plus-square-o" style="margin-left:5px;"></i></a>
    </label>
    <div id="collapseLibrary" class="collapse">

```


<h4>LIBRARY (iLib) FACULTY ACCESS</h4>

<p>If you encounter problems with iLib using your
faculty ID (usual reason is an existing iLib student profile), call 981-
8500 local 2864. Request your iLib account to be changed from student to
faculty.</p>

```

    </div>
  </div>
</div>
<div class="box-footer" style="margin-top: 10px;">
  <?php echo form_hidden($ldap_print); ?>
  <input type="hidden" class="form-control" name="username"
value="<?php echo $username; ?>">
  <input type="hidden" class="form-control" name="print_type"
value="<?php echo $print_type; ?>">
  <div class="row">
    <div class="col-xs-5 col-sm-4 col-md-4 col-lg-4 col-xl-4">
      <button type="submit" class="btn btn-lg btn-primary"><i
class="fa fa-print"></i> Print Receipt</button>
    </div>
    <div class="col-xs-offset-7 col-sm-offset-6 col-md-offset-4">
      <a class="btn btn-lg btn-default" href="<?php echo
site_url('tech/print_account/cancel/'. $username); ?>" data-toggle="tooltip" data-
placement="auto" data-html="true" title="Go Back"><i class="fa fa-arrow-left"></i>
Go Back</a>
    </div>
  </div>
  <?php echo form_close(); ?>
</div>

```

application/views/print_multiple_confirm.php

```

<div class="box">
  <div class="box-body">
    <div id="print" style="font-family: Monospace;">
      <span style="letter-spacing: 1px;">DILNET ACCOUNT: NEW ACCOUNTS
      <br>-----<br>
    </span>
    <table>
      <tbody>
        <?php
          // print "<pre>";
          // print_r($ldap_prints);
          // print "</pre>";
          foreach ($ldap_prints as $ldap_print):
            // print "<pre>";
            // print_r($ldap_print);
            // print "</pre>";

          ?>
          <?php foreach ($ldap_print as $key => $value): ?>
            <tr>
              <td style="border-bottom: 1px solid gray;
vertical-align: top; padding-right: 5px; "><strong><?php echo $key; ?></strong>
            </td>
              <td style="margin-left: 5px; border-
bottom: 1px solid gray;">
              <?php

```

```

        if(is_array($value)){
            foreach($value as $a) { echo
$a."</br>"; }
        }
        else { echo $value; }
        ?>
    </td>
</tr>
<?php endforeach; ?>
<?php endforeach; ?>
</tbody>
</table>
</div>
<?php echo form_open('tech/print_account/send_multiple_to_printer');
?>
<hr>
<div class="col-xs-12 col-sm-12 col-md-12 col-lg-6 col-xl-6 form-
group <?php echo (form_error('printer_opt') == '') ? '' : 'has-error'; ?>">
    <h3>Select Printer</h3>
    <select class="form-control select2-printer select2-hidden-
accessible" style="width: 100%;" tabindex="-1" aria-hidden="true"
name="printer_opt">
        <option value="tech-thermal">Tech Printer TM-T82
(Thermal)</option>
        <option value="tech-ribbon">Tech Printer TM-U220
(Ribbon)</option>
        <option value="admin-thermal">Admin Printer TM-T82
(Thermal)</option>
    </select>
    <?php echo form_error('printer_opt'); ?>
</div>
<div class="col-xs-8 form-group">
    <h3>Print Messages</h3>
    <div class="checkbox">
        <label>
            <input type="checkbox" name="opt_changepass"
class="minimal" <?php if($print_type == 'reset') { echo "checked"; } ?>>Change
Password<a data-toggle="collapse" href=#collapseChangepass aria-
controls="collapseChangepass" aria-expanded="false"><i class="text-expand fa fa-
plus-square-o" style="margin-left:5px;"></i></a>
        </label>
        <div id="collapseChangepass" class="collapse">
            <p>Please change your password immediately
            <p>at</p><h4>https://accounts.upd.edu.ph</h4><p>Or scan this code to go to the
website: [QR Code]</p>
        </div>
    </div>
    <div class="checkbox">
        <label>
            <input type="checkbox" name="opt_facproxy"
class="minimal" <?php if(isset($ldap_print["Emp Type"]) && $ldap_print["Emp
Type"]=='FAC') { echo "checked"; } ?>>Faculty Proxy<a data-toggle="collapse"
href=#collapseFacproxy aria-controls="collapseFacproxy" aria-expanded="false"><i
class="text-expand fa fa-plus-square-o" style="margin-left:5px;"></i></a>
        </label>
        <div id="collapseFacproxy" class="collapse">
            <h4>FACULTY PROXY SETTINGS</h4>
            <p>You may access the unrestricted proxy for the
faculty by using the following settings:<br>

```

```

        <strong>Host:</strong>
facultyproxy.upd.edu.ph<br>
        <strong>Port:</strong> 8080<br>
    </p>
</div>
</div>
<div class="checkbox">
    <label>
        <input type="checkbox" name="opt_services"
class="minimal" <?php if($print_type=='reset') { echo "checked"; } ?>>Online
Services<a data-toggle="collapse" href=#collapseServices aria-
controls="collapseServices" aria-expanded = "false"><i class="text-expand fa fa-
plus-square-o" style="margin-left:5px;"></i></a>
    </label>
    <div id="collapseServices" class="collapse">
        <h4>ONLINE SERVICES</h4>
        <p>
            <strong>DILNET ACCOUNT:</strong>
https://accounts.upd.edu.ph<br>
            <strong>UPD WEBMAIL:</strong>
https://mail.upd.edu.ph<br>
            <strong>CRS:</strong> https://crs.upd.edu.ph<br>
            <strong>UVLE:</strong> http://uvle.upd.edu.ph<br>
            <strong>Library (iLib):</strong>
https://ilib.upd.edu.ph
        </p>
    </div>
</div>
<div class="checkbox">
    <label><input type="checkbox" name="opt_helpdesk"
class="minimal" <?php if($print_type=='reset') { echo "checked"; } ?>>Dilnet
Helpdesk<a data-toggle="collapse" href=#collapseHelpdesk aria-
controls="collapseHelpdesk" aria-expanded = "false"><i class="text-expand fa fa-
plus-square-o" style="margin-left:5px;"></i></a>
    </label>
    <div id="collapseHelpdesk" class="collapse">
        <h4>DILNET HELPDESK</h4>
        <p>Contact us if you have problems with the ff:<br>
        - Connecting to DILNET Wifi<br>
        - UPD Webmail (mail.upd.edu.ph)<br>
        - Dilnet Account (accounts.upd.edu.ph)<br>
            <strong>Email:</strong> helpdesk@upd.edu.ph<br>
            <strong>Phone:</strong> 981-8500 local 2050<br>
            <strong>Website:</strong>
http://dilnet.upd.edu.ph<br>
            <strong>Facebook:</strong>
http://www.facebook.com/updilnet<br>
            <strong>Twitter:</strong>
http://www.twitter.com/dilnet<br>
        </p>
    </div>
</div>
<div class="checkbox">
    <label><input type="checkbox" name="opt_crs" class="minimal"
<?php if(isset($ldap_print["Emp Type"]) && $ldap_print["Emp Type"]=='FAC') { echo
"checked"; } ?>>CRS Tagging<a data-toggle="collapse" href=#collapseCrs aria-

```

```

controls="collapseCrs" aria-expanded = "false"><i class="text-expand fa fa-plus-square-o" style="margin-left:5px;"></i></a>
</label>
<div id="collapseCrs" class="collapse">
    <h4>CRS FACULTY ROLE</h4>
    <p>We have forwarded your information to CRS for
tagging. Average processing time is 1-2 days. You may follow up this request by
calling <strong>981-8500 local 4560</strong></p>
</div>
</div>
<div class="checkbox">
    <label><input type="checkbox" name="opt_library"
class="minimal" <?php if(isset($ldap_print["Emp Type"])) && $ldap_print["Emp
Type"]=='FAC') { echo "checked"; } ?>>ilib Access<a data-toggle="collapse"
href=#collapseLibrary aria-controls="collapseLibrary" aria-expanded = "false"><i
class="text-expand fa fa-plus-square-o" style="margin-left:5px;"></i></a>
</label>
<div id="collapseLibrary" class="collapse">
    <h4>LIBRARY (ilib) FACULTY ACCESS</h4>
    <p>If you encounter problems with ilib using your
faculty ID (usual reason is an existing ilib student profile), call <strong>981-
8500 local 2864</strong>. Request your ilib account to be changed from student to
faculty.</p>
</div>
</div>
</div>
</div>
<div class="box-footer" style="margin-top: 10px;">
    <?php echo form_hidden($ldap_print); ?>
    <input type="hidden" class="form-control" name="print_type"
value="<?php echo $print_type; ?>">
    <div class="row">
        <div class="col-xs-5 col-sm-4 col-md-4 col-lg-4 col-xl-4">
            <button type="submit" class="btn btn-lg btn-primary"><i
class="fa fa-print"></i> Print Receipt</button>
        </div>
        <div class="col-xs-offset-7 col-sm-offset-6 col-md-offset-4">
            <a class="btn btn-lg btn-default" href="<?php echo
site_url('tech/create_user'); ?>" data-toggle="tooltip" data-placement="auto"
data-html="true" title="Go Back"><i class="fa fa-arrow-left"></i> Go Back</a>
        </div>
    </div>
    <?php echo form_close(); ?>
</div>

```

```

application/views/radius_profile_js.php

```

```

<!-- DataTables -->
<script src="<?php echo $this->config->item('theme_url');
?>/plugins/datatables/jquery.dataTables.min.js"></script>
<script src="<?php echo $this->config->item('theme_url');
?>/plugins/datatables/dataTables.bootstrap.min.js"></script>

```

```

<!-- page script -->

```

```

<script>
$(function () {
    $("#acct_search_results").DataTable({

```

```

    });
    $("#radius_search_results").DataTable({
    });
    $("#online_users_search_results").DataTable({
    });
    $("#users_today_search_results").DataTable({
    });
});

function changeRadiusSearchText(){
    if ($('#username_field').val().length >= 3) {
        $("#radiusSubmitButton").text("Searching ... Please wait ...");
        $("#radiusSubmitButton").removeClass('btn-info').addClass('btn-
warning');
    }

};
</script>

```

application/views/radius_profile.php

```

<div class="row">
    <div class="col-md-12">
        <!-- Horizontal Form -->
        <div class="box box-info">
            <div class="box-header with-border">
                <h3 class="box-title">Get Radius Profile by Username</h3>
            </div><!-- /.box-header -->
            <!-- form start -->
            <?php echo form_open('tech/radius_profile/name_search', 'class="form-
horizontal"'); ?>
            <div class="box-body">
                <div class="form-group <?php echo (form_error('username') == '') ? '' :
'has-error'; ?>">
                    <div class="col-xs-12">
                        <?php echo form_error('username'); ?>
                    <div class="input-group">
                        <div class="input-group-addon"><i class="fa fa-user"></i></div>
                        <div class="input-group-text" id="username_field" class="form-control input-lg"
                        <input type="text" id="username" required autofocus>
                        name="username" placeholder="Type username" required autofocus>
                    </div><!-- /.input-group -->
                </div><!-- /.col-xs-12 -->
            </div>
        </div><!-- /.box-body -->
        <div class="box-footer">
            <button type="submit" id="radiusSubmitButton" onclick="return
changeRadiusSearchText();" class="btn btn-info col-xs-12">Search Profile</button>
        </div><!-- /.box-footer -->
    </div><!-- /.col-md-12 -->
</div><!-- /.row -->

```

```

<div class="row">
    <div class="col-md-6">
        <div class="box box-primary">

```

```

<div class="box-body" style='overflow: auto;'>
  <table class="table table-hover table-responsive">
    <tr>
      <th>Total Input Bytes Today in MB</th>
      <td><?php $input_total = $today_input_total/1048576; echo
number_format($input_total, 2, '.', ','); ?></td>
    </tr>
    <tr>
      <th>Total Output Bytes Today in MB</th>
      <td><?php $output_total = $today_output_total/1048576; echo
number_format($output_total, 2, '.', ','); ?></td>
    </tr>
  </table>
</div><!-- ./box-body -->
</div><!-- ./box -->
</div><!-- ./col-xs-6 -->
</div><!-- ./row -->

<?php if(isset($username)): ?>
<div class="row">
  <div class="col-md-12">
    <div class="box box-primary">
      <div class="box-header with-border">
        <h3 class="box-title">Radius result for <strong><?php echo
strtoupper($username); ?></strong></h3>
      </div><!-- /.box-header with-border -->

      <?php if(isset($results_db)): ?>
      <div class="box-body" style='overflow: auto;'>
        <table id="radius_search_results" class="table table-hover table-
responsive">
          <thead>
            <tr>
              <th>ID</th>
              <th>IP Address</th>
              <th>Start Time</th>
              <th>Stop Time</th>
              <th>Terminate Cause</th>
              <th>Input (MB)</th>
              <th>Output (MB)</th>
            </tr>
          </thead>
          <tbody>
            <?php foreach($results_db as $r): ?>
            <tr>
              <td><?php echo $r["radacctid"]; ?></td>
              <td><?php echo $r["framedipaddress"]; ?></td>
              <td><?php echo $r["acctstarttime"]; ?></td>
              <td><?php echo $r["acctstoptime"]; ?></td>
              <td><?php echo $r["acctterminatecause"]; ?></td>
              <td><?php echo $r["acctinputoctets"]/1048576; echo
number_format($input_mb, 2, '.', ','); ?></td>
              <td><?php echo $r["acctoutputoctets"]/1048576; echo
number_format($output_mb, 2, '.', ','); ?></td>
            </tr>
            <?php endforeach; //.results_db ?>
          </tbody>
        </table>
      </div><!-- /.div.box-body.table-responsive -->
    </div>
  </div>
</div>

```

```

        <?php endif; ?>
    </div><!-- /.box box-primary -->
</div><!-- /.col-xs-12 -->
</div><!-- /.row -->
<?php endif; ?>

<?php if(isset($online_users)): ?>
<div class="row">
    <div class="col-xs-12">
        <div class="box box-primary">
            <div class="box-header with-border">
                <h3 class="box-title">Online Users</h3>
            </div><!-- /.box-header -->
            <div class="box-body" style='overflow: auto;'>
                <table id="online_users_search_results" class="table table-hover table-
responsive">
                    <thead>
                        <tr>
                            <th>Username</th>
                            <th>IP Address</th>
                            <th>Start Time</th>
                        </tr>
                    </thead>
                    <tbody>
                        <?php foreach($online_users as $r): ?>
                        <tr>
                            <td><?php echo $r["username"]; ?></td>
                            <td><?php echo $r["framedipaddress"]; ?></td>
                            <td><?php echo $r["acctstarttime"]; ?></td>
                        </tr>
                        <?php endforeach; //.results_db ?>
                    </tbody>
                </table>
            </div> <!-- /.div.box-body.table-responsive -->
        </div><!-- /.box -->
    </div><!-- /.col-xs-12 -->
</div><!-- /.row -->
<?php endif; ?>

<?php if(isset($users_today)): ?>
<div class="row">
    <div class="col-xs-12">
        <div class="box box-primary">
            <div class="box-header with-border">
                <h3 class="box-title">Users Today</h3>
            </div><!-- /.box-header -->
            <div class="box-body">
                <table id="users_today_search_results" class="table table-hover">
                    <thead>
                        <tr>
                            <th>Username</th>
                            <th>IP Address</th>
                            <th>Action</th>
                        </tr>
                    </thead>
                    <tbody>
                        <?php foreach($users_today as $r): ?>

```

```
|  |  |
| --- | --- |
| <?php echo $r["username"]; ?></td> | <?php echo $r["framedipaddress"]; ?></td> |
|  | |


<?php endforeach; //.results_db ?>
</tbody>
</table>
</div> <!--/.div.box-body.table-responsive -->
</div><!-- /.box -->
</div><!-- /.col-xs-12 -->
</div><!-- /.row -->
<?php endif; ?>

```

application/views/remove_mail_view.php

```

<?php // print "<pre>"; print_r($results_db); print "</pre>"; ?>
<?php if(isset($results_db)): ?>
<div class="row">
<div class="col-xs-12">
<div class="box">
<div class="box-header">
<h3 class="box-title">Remove Mail Results</h3>
</div><!-- /.box-header -->
<div class="box-body">
<table class="table table-hover table-striped">
<thead style="letter-spacing:1px;">
<th style="text-align: right">Relay</th>
<th width="65%"># of Mail(s) Removed</th>
</tr>
</thead>
<tbody>
<!-- Address -->
<tr>
<th style="text-align: right">Primary</th>
<td class="col-md-4" align="left">
<?php echo $results_db[0]; ?>
</td>
</tr>
<tr>
<th style="text-align: right">Secondary</th>
<td class="col-md-4" align="left">
<?php echo $results_db[1]; ?>
</td>
</tr>
</tbody>
</table>
<div class="box-footer">
<div class="col-md-8 pull-right">
<a class="btn btn-app" href="<?php echo
site_url('tech/mail_tools/view_email_status/' . urlencode(base64_encode($search_string))) ?>" data-toggle="tooltip" data-placement="auto" data-html="true" title="View
Email Status"><i class="fa fa-eye text-info"></i>View Email Status</a>
</div>
<?php echo form_close(); ?>

```



```

        </div>
    </div> <!-- /div.table-responsive -->
</div><!-- /div.nav-tabs-custom -->
</div><!-- /div.col-xs-12 -->
</div><!-- /div.row -->
<?php endif; //isset.results ?>

```

application/views/restricted.php

```

<div class="error-page clearfix">
    <h2 class="headline text-red"><i class="fa fa-warning text-
red"></i></h2>
    <div class="error-content" style="margin-left: 120px;">
        <h3>Sorry! you are not allowed to access this page.</h3>
        <p>Contact the dev team if you feel this is incorrect.</p>
    </div>
</div><!-- /.error-page -->

```

application/views/search_logs_view.php

```

<div class="row">
    <div class="col-xs-12">
        <div class="nav-tabs-custom">
            <ul class="nav nav-tabs">
                <li class="active"><a href="#accountlog" data-toggle="tab" aria-
expanded="true">Account Log</a></li>
                <li class=""><a href="#changepass" data-toggle="tab" aria-
expanded="false">Changepass</a></li>
                <li class=""><a href="#lostpass" data-toggle="tab" aria-
expanded="false">Lostpass</a></li>
            </ul>
            <div class="tab-content">
                <div class="tab-pane active" id="accountlog">
                    <h3 class="box-title">Displaying ALL account logs from the last 3
months</h3>
                    <table id="accountlog_results" class="table table-bordered table-hover">
                        <thead>
                            <tr>
                                <th>ID</th>
                                <th>Date/Time</th>
                                <th>Account</th>
                                <th>Detail</th>
                                <th>Service</th>
                                <th>Log Level</th>
                                <th>Changed by</th>
                                <th>IP</th>
                            </tr>
                        </thead>
                        <tbody>
                            <?php foreach($results_accountlog as $r): ?>
                                <tr>
                                    <td><?php echo $r["id"]; ?></td>
                                    <td><?php echo $r["time"]; ?></td>
                                    <td><?php echo $r["account"]; ?></td>
                                    <td><?php echo $r["detail"]; ?></td>
                                    <td><?php echo $r["service"]; ?></td>
                                    <td><?php echo $r["log_level"]; ?></td>
                                    <td><?php echo $r["changed_by"]; ?></td>
                                    <td><?php echo $r["ip"]; ?></td>
                                </tr>
                            </tbody>
                        </table>
                    </div>
                </div>
            </div>
        </div>
    </div>
</div>

```

```

<td><?php echo $r["log_level"]; ?></td>
<td><?php echo $r["changed_by"]; ?></td>
<td><?php echo $r["ip"]; ?></td>
</tr>
<?php endforeach; //.results ?>
</tbody>
<tfoot>
<tr>
<th>ID</th>
<th>Date/Time</th>
<th>Account</th>
<th>Detail</th>
<th>Service</th>
<th>Log Level</th>
<th>Changed by</th>
<th>IP</th>
</tr>
</tfoot>
</table>
</div><!-- /.tab-pane#accountlog -->
<div class="tab-pane" id="changepass">
<h3 class="box-title">Displaying changepass from the last 3 months</h3>
<table id="changepass_results" class="table table-bordered table-hover">
<thead>
<tr>
<th>Username</th>
<th>IP</th>
<th>Changed by</th>
<th>Date / Time</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<?php foreach($results_changepass as $r): ?>
<tr>
<td><?php echo $r["username"]; ?></td>
<td><?php echo $r["ip"]; ?></td>
<td><?php echo $r["changed_by"]; ?></td>
<td><?php echo $r["time"]; ?></td>
<td><?php echo $r["notes"]; ?></td>
</tr>
<?php endforeach; //.results ?>
</tbody>
<tfoot>
<tr>
<th>Username</th>
<th>IP</th>
<th>Changed by</th>
<th>Date / Time</th>
<th>Notes</th>
</tr>
</tfoot>
</table>
</div><!-- /.tab-pane#changepass -->
<div class="tab-pane" id="lostpass">
<h3 class="box-title">Displaying lostpass logs from the last 3 months</h3>
<table id="lostpass_results" class="table table-bordered table-hover">
<thead>
<tr>

```

```

        <th>Username</th>
        <th>Date Added</th>
        <th>Active</th>
    </tr>
</thead>
<tbody>
<?php foreach($results_lostpass as $r): ?>
<tr>
<td><?php echo $r["username"]; ?></td>
<td><?php echo $r["date_added"]; ?></td>
<td><?php echo $r["active"]; ?></td>
</tr>
<?php endforeach; //.results ?>
</tbody>
<tfoot>
    <tr>
        <th>Username</th>
        <th>Date Added</th>
        <th>Active</th>
    </tr>
</tfoot>
</table>
</div><!-- /.tab-pane#lostpass -->
</div><!-- /.tab-content -->
</div><!-- /.nav-tabs-custom -->
</div><!-- /.col-xs-12 -->
</div><!-- /.row -->

```

application/views/upload_activation_success.php

```

<html>
<head>
<title>Upload Form</title>
</head>
<body>
<h5>Your file was successfully uploaded! <?php echo anchor('tech/create_user',
'Upload Another. '); ?></h5>

<!-- <ul>
<?php foreach ($upload_data as $item => $value):?>
<li><?php echo $item;?>: <?php echo $value;?></li>
<?php endforeach; ?>
</ul> -->
<?php if($incomplete_details_count>0) { ?>
    <h4> Only the following will be created.
    <br><strong><?php echo $incomplete_details_count ?></strong> <?php echo
($incomplete_details_count > 1) ? 'entries were' : 'entry was'; ?> discarded due
to incomplete or invalid details.</h3>
<?php
} ?>
<div class="row">
    <div class="col-xs-12">
        <div class="box">
            <div class="box-header">
                <div class="box-title">CSV Cleanup Results</div>
            <h3 class="box-title">
        </div><!-- /.box-header -->

```

```

<!-- for big screens -->
<div class="box-body hidden-xs hidden-sm hidden-md">
  <table id="csv_upload_results" class="table table-bordered table-hover
table-responsive">
    <thead>
      <tr>
        <th>last</th>
        <th>first</th>
        <th>middle</th>
        <th>unit</th>
        <th>division</th>
        <th>designation</th>
        <th>type</th>
        <th>employee #</th>
      </tr>
    </thead>
    <tbody>
      <?php foreach($csv_filtered as $line_entries): ?>
        <tr>
          <td><?php echo $line_entries[0] ?></td>
          <td><?php echo $line_entries[1] ?></td>
          <td><?php echo $line_entries[2] ?></td>
          <td><?php echo $line_entries[3] ?></td>
          <td><?php echo $line_entries[4] ?></td>
          <td><?php echo $line_entries[5] ?></td>
          <td><?php echo $line_entries[6] ?></td>
          <td><?php echo $line_entries[7] ?></td>
        </tr>
      <?php endforeach; //.results_db ?>
    </tbody>
    <tfoot>
      <tr>
        <th>last</th>
        <th>first</th>
        <th>middle</th>
        <th>unit</th>
        <th>division</th>
        <th>designation</th>
        <th>type</th>
        <th>employee #</th>
      </tr>
    </tfoot>
  </table>
</div><!-- /.box-body -->
<?php echo form_open('create_multiple_employees/activate_multiple'); ?>
<div class="box-footer">
  <div class="row">
    <div class="col-xs-6 col-sm-5 col-md-5">
      <button type="submit" class="btn btn-lg btn-primary"
id='create_activate_multiple' onclick="return confirm('This will create and
activate the user accounts. Continue?');"><i class="fa fa-users"></i> Create and
Activate Multiple Accounts</button>
    </div>
    <div class="col-xs-offset-8 col-sm-offset-7 col-md-offset-5">
      <div class="col-xs-offset-8 col-sm-offset-7 col-md-offset-5">
        <a class="btn btn-lg btn-default" href="<?php echo
site_url('tech/create_user'); ?>" data-toggle="tooltip" data-placement="auto"
data-html="true" title="Go Back"><i class="fa fa-arrow-left"></i> Go Back</a>
      <?php echo form_close(); ?>
    </div>
  </div>

```

```

    </div><!-- /.row -->
</div><!-- div.box-footer -->
    </div><!-- /.box -->
</div><!-- /.row -->
</div><!-- /.row -->
</body>
</html>

```

application/views/upload_success.php

```

<html>
<head>
<title>Upload Form</title>
</head>
<body>
<h5>Your file was successfully uploaded! <?php echo anchor('tech/create_user',
'Upload Another. '); ?></h5>

<!-- <ul>
<?php foreach ($upload_data as $item => $value):?>
<li><?php echo $item;?>: <?php echo $value;?></li>
<?php endforeach; ?>
</ul> -->
<?php if($incomplete_details_count>0) { ?>
    <h4> Only the following will be created.
    <br><strong><?php echo $incomplete_details_count ?></strong> <?php echo
($incomplete_details_count > 1) ? 'entries were' : 'entry was'; ?> discarded due
to incomplete or invalid details.</h3>
<?php
} ?>
<div class="row">
    <div class="col-xs-12">
        <div class="box">
            <div class="box-header">
                <h3 class="box-title">CSV Cleanup Results</h3>

                </div><!-- /.box-header -->
                <!-- for big screens -->
                <div class="box-body hidden-xs hidden-sm hidden-md">
                    <table id="csv_upload_results" class="table table-bordered table-hover
table-responsive">
                        <thead>
                            <tr>
                                <th>last</th>
                                <th>first</th>
                                <th>middle</th>
                                <th>unit</th>
                                <th>division</th>
                                <th>designation</th>
                                <th>type</th>
                                <th>employee #</th>
                            </tr>
                        </thead>
                        <tbody>
                            <?php foreach($csv_filtered as $line_entries): ?>
                                <tr>
                                    <td><?php echo $line_entries[0] ?></td>

```

```

        <td><?php echo $line_entries[1] ?></td>
        <td><?php echo $line_entries[2] ?></td>
        <td><?php echo $line_entries[3] ?></td>
        <td><?php echo $line_entries[4] ?></td>
        <td><?php echo $line_entries[5] ?></td>
        <td><?php echo $line_entries[6] ?></td>
        <td><?php echo $line_entries[7] ?></td>
    </tr>

```

```

    <?php endforeach; //.results_db ?>
</tbody>

```

```

<tfoot>
    <tr>

```

```

        <th>last</th>
        <th>first</th>
        <th>middle</th>
        <th>unit</th>
        <th>division</th>
        <th>designation</th>
        <th>type</th>
        <th>employee #</th>
    </tr>

```

```

    </tfoot>
</table>

```

```

</div><!-- /.box-body -->

```

```

<div class="box-body hidden-lg hidden-xl" style="overflow: auto;">
    <table id="csv_upload_results" class="table table-bordered table-hover

```

```

table-responsive">
    <thead>

```

```

        <tr>
            <th>last</th>
            <th>first</th>
            <th>middle</th>
            <th>unit</th>
            <th>division</th>
            <th>designation</th>
            <th>type</th>
            <th>employee #</th>
        </tr>
    </thead>

```

```

    <tbody>
        <?php foreach($csv_filtered as $line_entries): ?>
            <tr>

```

```

                <td><?php echo $line_entries[0] ?></td>
                <td><?php echo $line_entries[1] ?></td>
                <td><?php echo $line_entries[2] ?></td>
                <td><?php echo $line_entries[3] ?></td>
                <td><?php echo $line_entries[4] ?></td>
                <td><?php echo $line_entries[5] ?></td>
                <td><?php echo $line_entries[6] ?></td>
                <td><?php echo $line_entries[7] ?></td>
            </tr>

```

```

        <?php endforeach; //.results_db ?>
    </tbody>

```

```

<tfoot>
    <tr>

```

```

        <th>last</th>
        <th>first</th>
        <th>middle</th>
        <th>unit</th>
    </tr>

```

```

        <th>division</th>
        <th>designation</th>
        <th>type</th>
        <th>employee #</th>
    </tr>
</tfoot>
</table>
</div><!-- /.box-body -->
<?php echo form_open('create_multiple_employees/activate_multiple'); ?>
<div class="box-footer">
    <div class="row">
        <div class="col-xs-3 col-sm-5 col-md-4">
            <button type="submit" class="btn btn-lg btn-primary"
id='create_activate_multiple' onclick="return changeText();" ><i class="fa fa-
users"></i> Activate Accounts</button>
        </div>
        <div class="col-xs-offset-7 col-sm-offset-6 col-md-offset-4">
            <a class="btn btn-lg btn-default" href="<?php echo
site_url('tech/create_user'); ?>" data-toggle="tooltip" data-placement="auto"
data-html="true" title="Go Back"><i class="fa fa-arrow-left"></i> Go Back</a>
            <?php echo form_close(); ?>
        </div>
    </div><!-- /.row -->
</div><!-- div.box-footer -->
</div><!-- /.box -->
</div><!-- /.row -->
</div><!-- /.row -->
</body>
</html>

```

application/views/user_account_disable_confirm.php

```

<div class="box box-solid" style="padding-left: 15px;">
    <div class="box-body">
        <h4>You are going to <strong><?php echo $email['disabled'] ? "unblock" :
"block" ?></strong> this DilNet account. <br /><?php echo $email['disabled'] ?
"When unblocked, the username will regain access to the proxy servers. <br>" ?>
"When blocked, the user will not be able to use the proxy servers. <br>" ?>
        <span class="text-danger">REVIEW</span> the information below before you
continue.</h4>
        <hr>
        <div>
            <h4><?php echo $email['disabled'] ? "Unblock" : "Block" ?>: <strong><?php echo
base64_decode(urldecode($block_email)); ?></strong></h4>
            <?php echo $email['disabled'] ?
form_open('tech/proxy_tools/unblock_user_confirm') :
form_open('tech/proxy_tools/block_user_confirm'); ?>
            <div class="form-group <?php echo (form_error('disable_reason') == '') ? '' :
'has-error'; ?>">
                <label for="disable_reason">Reason</label>
                <input type="text" class="form-control" name="disable_reason" value="<?php
echo set_value('disable_reason'); ?>" required autofocus>
                <input type="hidden" class="form-control" name="block_email" value="<?php
echo $block_email; ?>">
                <?php echo form_error('disable_reason'); ?>
            </div>
        </div>
    </div><!-- /.box-body -->
</div>

```

```

<div class="box-footer">
  <div class="row">
    <div class="col-xs-5 col-sm-4 col-md-3">
      <button type="submit" class="btn btn-lg <?php echo $email['disabled'] ?
"btn-success" : "btn-danger" ?>"><i class="fa <?php echo $email['disabled'] ? 'fa-
unlock' : 'fa-lock' ?>"></i> <?php echo $email['disabled'] ? "Unblock" : "Block"
?> User</button>
    </div>
    <div class="col-xs-offset-7 col-sm-offset-4 col-md-offset-2">
      <a class="btn btn-lg btn-default" href="<?php echo
site_url('tech/proxy_tools/user_blocking'); ?>" data-toggle="tooltip" data-
placement="auto" data-html="true" title="Go Back"><i class="fa fa-arrow-left"></i>
Go Back</a>
      <?php echo form_close(); ?>
    </div>
  </div><!-- /.row -->
</div><!-- div.box-footer -->
</div><!-- div.box -->

```

application/views/user_blocking_search_view.php

```

<div class="row">
  <div class="col-xs-12">
    <!-- Horizontal Form -->
    <div class="box box-info">
      <div class="box-header with-border">
        <div class="box-header"><i class="fa fa-search"></i> Search Proxy DB</h3>
        <h3 class="box-title"><i class="fa fa-search"></i> Search Proxy DB</h3>
        <br><h4>If you know the email address or username, type it below. Or type
part of it to find it.
        <br>Then press the <i class="fa fa-user-plus text-success"></i> to <span
class="text-success"><strong>unlock</strong></span> or <i class="fa fa-user-times
text-danger"></i> to <span class="text-danger"><strong>block</strong></span> the
user from the proxy servers.</h4>
      </div><!-- /.box-header -->
      <!-- form start -->
      <?php echo form_open('tech/proxy_tools/user_blocking', 'class="form-
horizontal"'); ?>
      <div class="box-body">
        <div class="form-group <?php echo (form_error('user_block_search_str'))
== '' ? '' : 'has-error'; ?>">
          <div class="col-xs-12">
            <?php echo form_error('user_block_search_str'); ?>
            <div class="input-group">
              <div class="input-group-addon"><i class="glyphicon glyphicon-
search"></i></div>
              <input type="text" id="user_block_search_str" class="form-control
input-lg" name="user_block_search_str" placeholder="Enter username" required
autofocus>
            </div><!-- /.input-group -->
          </div>
        </div><!-- /.box-body -->
      </div><!-- /.box-footer">
      <div class="box-footer">
        <button type="submit" id="userBlockSubmitButton" onclick="return
changeUserBlockSearchText();" class="btn btn-info col-xs-12">Search
Username</button>
      </div><!-- /.box-footer -->
      <?php echo form_close(); ?>

```



```
<?php if(!empty($user_blocking_results_db)) {?>  
    <div class="row">  
        <div class="col-xs-12">  
            <div class="box">  
                <div class="box-header">  
                    <h3 class="box-title">DB Search Results <?php  
if(isset($user_blocking_search_string)){ echo "for  
'<strong>" . $user_blocking_search_string . "</strong>'"; } ?></h3>  
                </div><!-- /.box-header -->  
                <div class="box-body">  
                    <table id="log_search_results" class="table table-bordered table-hover">  
                        <thead>  
                            <tr>  
                                <th class="text-center">Username</th>  
                                <th class="text-center">Status</th>  
                                <th class="text-center hidden-xs">Date Banned</th>  
                                <th class="text-center">Admin</th>  
                                <th class="text-center">Actions</th>  
                            </tr>  
                        </thead>  
                        <tbody>  
                            <?php foreach($user_blocking_results_db as $key => $r): ?>  
                                <tr>  
                                    <td width="30%" class="text-center"><?php echo $r["username"]  
                                    <td class="text-center"><?php echo $r["disabled"]==TRUE ? "<span  
class='text-danger'><strong>BLOCKED</strong></span>": "<span class='text-  
success'>ALLOWED</span>" ?></td>  
                                    <td class="text-center hidden-xs"><?php echo $r["date_banned"]  
                                    <td class="text-center"><?php echo $r["tech_banner"] ?></td>  
                                    <td class="text-center">  
                                        <a href="<?php echo  
base_url('tech/proxy_tools/view_user_proxy_status/' . urlencode(base64_encode($r['us  
ername']))) . '' ); ?>">i class="fa fa-file-o fa-eye" data-toggle="tooltip" data-  
placement="auto" data-html="true" title="View User  
Entry"></i></a>&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&~  
                                        <a href="<?php echo $r['disabled'] ?  
site_url('tech/proxy_tools/unblock_user_confirm/' . urlencode(base64_encode($r['userna  
me']))) :  
site_url('tech/proxy_tools/block_user_confirm/' . urlencode(base64_encode($r['userna  
me']))) ?>" data-toggle="tooltip" data-placement="auto" data-html="true"  
title="<?php echo $r['disabled'] ? "Unblock" : "Block" ?> User">i class="fa <?php  
echo $r['disabled'] ? "fa-user-plus" : "fa-user-times" ?> <?php echo  
$r['disabled'] ? "text-success" : "text-danger" ?>"></i></a>  
                                    </td>  
                                </tr>  
                            <?php endforeach; //.results_db ?>  
                        </tbody>  
                    </tfoot>  
                </div>  
            </div>  
        </div>
```

```

        <th class="text-center">Actions</th>
    </tr>
</tfoot>
</table>
</div><!-- /.box-body -->
</div><!-- /.box -->
</div><!-- /.row -->
</div><!-- /.row -->
<?php } ?>

```

application/views/username_view.php

```

<div class="row">
    <div class="col-xs-12">
        <!-- Horizontal Form -->
        <div class="box box-info">
            <div class="box-header with-border">
                <h3 class="box-title"><i class="fa fa-file-text-o"></i> Log Entry:
            </div><!-- /.box-header -->
            <div class="box-body">
                <strong><?php echo $block_email ?></strong></div>
            </div>
        </div>
        <?php if(isset($results_db)): ?>
        <div class="row">
            <div class="col-xs-12">
                <div class="nav-tabs-custom">
                    <?php
                        $attributes = array('class' => 'form-horizontal', 'role' => 'form',
                        'id' => 'mergeForm');
                        echo form_open('tech/proxy_tools/view_message_id', $attributes);
                    <?>

                    <table class="table table-hover table-striped">
                        <thead style="letter-spacing:1px;">
                            <th style="text-align: right">Attribute</th>
                            <th style="width:65%">Value</th>
                        </thead>
                        <tbody>
                            <!-- Address -->
                            <tr>
                                <th style="text-align: right">Username</th>
                                <td class="col-md-4" align="left">
                                    <?php echo $results_db['username']; ?>
                                </td>
                            </tr>

                            <!-- Status -->
                            <tr>
                                <th style="text-align: right">Status</th>
                                <td class="col-md-4" align="left">
                                    <?php echo $results_db['disabled'] == TRUE ? "<strong><span
                                    class=\"text-danger\">BLOCKED</span></strong>" : "<span class=\"text-
                                    success\"><strong>Allowed</strong></span>" ?>
                                </td>
                            </tr>

                            <!-- Date Banned -->
                            <tr>

```

```

        <th style="text-align: right" class="<?php echo
$results_db['disabled'] == FALSE ? 'text-muted': ''?">Date Banned</th>
        <td class="col-md-4" align="left">
        <?php if(!$results_db['disabled']){ ?>
        <span class="text-muted"><?php echo
$results_db['date_banned'] ?></span>
        <?php } else {
        echo $results_db['date_banned']!="0000-00-00 00:00:00" ?
$results_db['date_banned'] : "<span class=\"text-muted\">No Data</span>";
        } ?>
        </td>
    </tr>

    <!-- Reason Banned -->
    <tr>
        <th style="text-align: right" class="<?php echo
$results_db['disabled'] == FALSE ? 'text-muted': ''?">Reason for Ban</th>
        <td class="col-md-4" align="left">
        <?php if(!$results_db['disabled']){ ?>
        <span class="text-muted"><?php echo
$results_db['reason_banned'] ?></span>
        <?php } else {
        echo $results_db['reason_banned']!=NULL ?
$results_db['reason_banned'] : "<span class=\"text-muted\">No Data</span>";
        } ?>
        </td>
    </tr>

    <!-- Admin Banner -->
    <tr>
        <th style="text-align: right" class="<?php echo
$results_db['disabled'] == FALSE ? 'text-muted': ''?">Banned by</th>
        <td class="col-md-4" align="left">
        <?php if(!$results_db['disabled']){ ?>
        <span class="text-muted"><?php echo
$results_db['tech_banner'] ?></span>
        <?php } else {
        echo $results_db['tech_banner']!=NULL ?
$results_db['tech_banner'] : "<span class=\"text-muted\">No Data</span>";
        } ?>
        </td>
    </tr>

    <!-- Date UnBanned -->
    <tr>
        <th style="text-align: right" class="<?php echo
$results_db['disabled'] == TRUE ? 'text-muted': ''?">Date UnBanned</th>
        <td class="col-md-4" align="left">
        <?php if($results_db['disabled']){ ?>
        <span class="text-muted"><?php echo
empty($results_db['date_unbanned']) ? $results_db['date_unbanned'] : "<span
class=\"text-muted\">Never</span>"; ?></span>
        <?php } else {
        echo empty($results_db['date_unbanned']) ?
$results_db['date_unbanned'] : "<span class=\"text-muted\">Never</span>";
        } ?>
        </td>
    </tr>

```

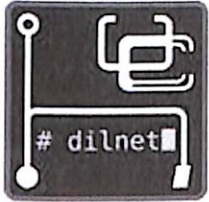
```

<!-- Reason UnBanned -->
<tr>
  <th style="text-align: right" class="<?php echo
$results_db['disabled'] == TRUE ? 'text-muted': ''?">Reason for UnBan</th>
  <td class="col-md-4" align="left">
    <?php if($results_db['disabled']){ ?>
      <span class="text-muted"><?php echo
$results_db['reason_unbanned']!=NULL ? $results_db['reason_unbanned'] : "<span
class=\"text-muted\">No Data</span>"; ?></span>
    <?php } else {
      echo $results_db['reason_unbanned']!=NULL ?
$results_db['reason_unbanned'] : "<span class=\"text-muted\">No Data</span>";
    } ?>
  </td>
</tr>
<!-- Admin Banner -->
<tr>
  <th style="text-align: right" class="<?php echo
$results_db['disabled'] == TRUE ? 'text-muted': ''?">UnBanned by</th>
  <td class="col-md-4" align="left">
    <?php if($results_db['disabled']){ ?>
      <span class="text-muted"><?php echo
$results_db['tech_unbanner']!=NULL ? $results_db['tech_unbanner'] : "<span
class=\"text-muted\">No Data</span>"; ?></span>
    <?php } else {
      echo $results_db['tech_unbanner']!=NULL ?
$results_db['tech_unbanner'] : "<span class=\"text-muted\">No Data</span>";
    } ?>
  </td>
</tr>
</tbody>
</table>
<div class="box-footer">
  <div class="row">
    <div class="col-xs-4 col-sm-4">
      <a class="btn btn-app" href="<?php echo
site_url('tech/proxy_tools/user_blocking'); ?>" data-toggle="tooltip" data-
placement="auto" data-html="true" title="Go Back"><i class="fa fa-arrow-left text-
info"></i> Go Back</a>
    </div>
    <div class="col-xs-offset-2 col-sm-offset-4">
      <a class="btn btn-app" href="<?php echo
$results_db['disabled'] ?
site_url('tech/proxy_tools/unblock_user_confirm/'.urlencode(base64_encode($results
_db['username']))) :
site_url('tech/proxy_tools/block_user_confirm/'.urlencode(base64_encode($results_d
_db['username']))) ?>" data-toggle="tooltip" data-placement="auto" data-html="true"
b['username']))) ?>" data-toggle="tooltip" data-placement="auto" data-html="true"
title="<?php echo $results_db['disabled'] ? "fa-user-plus" : "fa-user-times" ?>
class="fa <?php echo $results_db['disabled'] ? "text-success" : "text-danger" ?>"></i> <?php
echo $results_db['disabled'] ? "Unblock" : "Block" ?> User</a>
    </div>
  </div>
  <?php echo form_close(); ?>
</div>
</div>
</div> <!-- /div.table-responsive -->
</div><!-- /div.nav-tabs-custom -->
</div><!-- /div.col-xs-12 -->
</div><!-- /div.row -->

```

```
<?php endif; //isset.results ?>
```

application/assets/dilnetlogo.png



UNIVERSITY COMPUTER CENTER DILNET HELPDESK

application/assets/favicon.ico



application/assets/UPLogo.png

