Rochester Institute of Technology

B. Thomas Golisano College

of

Computing and Information Sciences

Master of Science in Information Technology

Project Approval Form

Student Name: Melissa A. Hofer

Project Title: A Website and Corresponding Database to Support the Digital Imaging and Remote Sensing (DIRS) Lab in the Chester F. Carlson Center for Imaging Science at the Rochester Institute of Technology

Project Committee

Name Signature Date

Professor Daniel Kennedy____________________________________
Chair

Dr. Carl Salvaggio____
Committee Member

A Website and Corresponding Database to Support the Digital Imaging and Remote Sensing (DIRS) Lab in the Chester F. Carlson Center for Imaging Science at the Rochester Institute of Technology

By

Melissa A. Hofer

Project submitted in partial fulfillment of the requirements for the degree of Master of Science in Information Technology

Rochester Institute of Technology

B. Thomas Golisano College
of
Computing and Information Sciences

June 30, 2005

Abstract:

The Digital Imaging and Remote Sensing (DIRS) lab in the Chester F. Carlson Center for Imaging Science at the Rochester Institute of Technology has developed a software package known as DIRSIG (Digital Imaging and Remote Sensing Image Generator), which performs image modeling in multiple regions of the spectrum (i.e. infrared, visible, etc.). The DIRS group required an efficient means to distribute software to new customers, provide software updates to existing customers, perform analysis on user information, and provide communications, such as training notices and upgrade notifications, to their user community.

In response to the issues identified above, a website and corresponding database was created using PHP and MySQL called "myDIRSIG". The website allows customers to create user accounts, containing contact, computer, and software information, and provides a vehicle to download new software products and updates to existing products. Every account can be enabled or disabled by the DIRS lab employees to help mitigate security concerns, by ensuring only authorized users have access to the website. Any enabled user account has access to download any version of the DIRSIG software that is available on the website. The DIRS lab employees who have additional admin privileges can send email announcements to the various mailing lists, based on user selected preferences, and maintain the operating system, platform, software and versions database tables via the website. The DIRS lab employees also have the ability to edit user accounts and the software available to the users.

This solution offers several benefits to the DIRS lab. The distribution of the DIRSIG software and software upgrades will become significantly more efficient with the elimination of having to create and mail CD-ROM media. Software upgrades and bug fixes will be accessible by the customers immediately as released by the DIRS lab, avoiding time spent previously on shipping new media. Customer information will be more accurate as customers will be able to easily update their contact and computer information. And finally, the DIRS lab employees will be able to perform analysis on customer information quickly and easily through the use of the customer database.

Table of Contents

Introduction

The Digital Imaging and Remote Sensing (DIRS) lab at the Chester F. Carlson Center for Imaging Science at the Rochester Institute of Technology has developed a software package known as DIRSIG (Digital Imaging and Remote Sensing Image Generator), which performs image modeling in multiple portions of the spectrum (i.e. infrared, visible, etc.). The user community of the software is located across the country, making an electronic software distribution solution desirable. Software upgrades have become problematical, as there is no tracking or distribution system in place to manage software upgrades and users can become out of synch with the latest version. The DIRS group would like to utilize a website to help manage the distribution of the software, software updates, and communications to the user community. The site would also help to ensure that only registered and authorized users, are able to download the software and related updates.

By implementing this website, the DIRS group will gain productivity in several areas. Software upgrades will be available for download as soon as they are created; no longer will CD-ROM's need to be burned and shipped out. The group will be able to tell at a glance for any given customer what operating systems and computers are running the software, and what software version each machine is currently running, which will aid in the troubleshooting processes. Mass e-mail announcements for training, software upgrades, and general administration will be able to be sent out quickly via mailing lists created by selections that the customers make when they create their accounts. Each customer account will have an enabled/disabled field that will be controlled by the website.

The website will benefit customers that use the software by allowing them real time access to upgrades and bug fixes, and allow them to create and modify their own user account with contact information, including a current e-mail address. The customer will be responsible for identifying and updating how many computers are running the software. For each computer, information on the operating system, platform, and the versions of DIRSIG currently installed will need to be entered. Each time a customer views their current software "subscriptions", they will be shown if they have the most current version of each software product, or if a new version is available for download.

1.1 Project Details

1.1.1 System Capabilities

- Provide a vehicle for the distribution of the DIRSIG software
- Create a database of customer contact and computer data
- Provide a communication vehicle for announcing new software releases, training opportunities, and other general information
- Provide the DIRS Lab with the ability to analyze customer information, i.e. software versions being utilized, types of computers and operating systems used, etc.
- Provide a means for the DIRS Lab to enable/disable customer accounts, thereby allowing or disallowing access to the myDIRSIG website

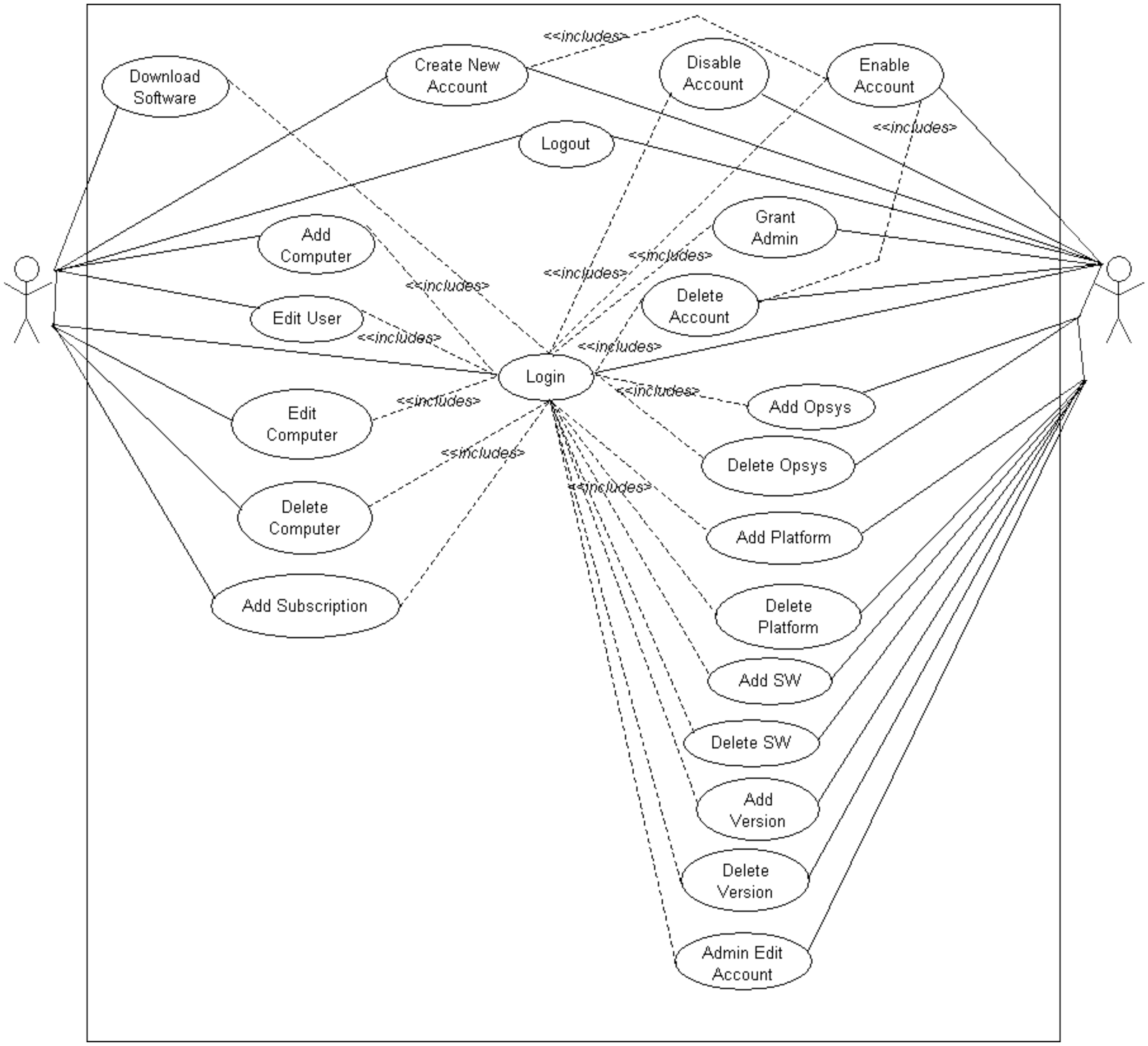
1.1.2 Project Objectives

The objective of this project is to create a working website using PHP, MySQL, and HTML to support the Digital Imaging and Remote Sensing (DIRS) group within the Chester Carlson Center for Imaging Science at the Rochester Institute of Technology.

The myDIRSIG website will be used for the tracking of customer information and the distribution of the Digital Imaging and Remote Sensing Image Generator (DIRSIG) software.

2 Documentation

2.1 Context Diagram



2.2 Use Cases

2.2.1 Log On to System

Use Case Number: **1**

Use Case Name: **Log On To System (Website)**

Primary Actor: User
Secondary Actor: n/a

Description:
When a user wishes to utilize the DIRSIG Website, they must go to the proper URL and login to the system.

Pre-Condition:
1. The user must have an enabled account.

Post-Condition:
1. The user is logged into the system.

Assumptions:
1. The user knows his/her email address and password
2. The user's account is enabled

Normal Course:
1. User types the correct URL into his/her web browser (dirsweb.php)
2. User inputs email address and password into web form
3. User clicks the "Login" button
4. System confirms user login information
5. System checks if user is "Enabled"
6. System checks if user is an "Admin User"
7. System displays either "User Options" or "Admin Options" page based on the user's account

Alternate Course:
4a. System is unable to validate login information
4a1. System displays error message to user
4a2. System prompts user to return to login screen
4a3. User inputs email address and password into the web form
4a4. User clicks the "Login" button
4a3a. If system is able to confirm login information goto #5
4a3b. If system is unable to confirm login information repeat 4a two additional times
4a3b1. If login information is not confirmed, system checks email address agaist valid email addresses in the database
4a3b2. System displays error to user
4a3b1a. System disables the account if the email address is valid
4a3b1a1. System generates an e-mail to the user with notification of account disablement
4a3b1a2. System generates an e-mail to a DIRS administrator with notification of account disablement
4a3b1b. System generates an e-mail to a DIRS administrator with notification of login faliure with invalid email address

5a. System displays an error message if the User is not "Enabled"

Includes: n/a

Files: dirsweb.php, useroptions.php, adminoptions.php, loginfail.php

2.2.2 Create New User Account

Use Case Number: **2**

Use Case Name: **Create New User Account**

Primary Actor: User
Secondary Actor: n/a

Description:
When a new user wishes to utilize the DIRSIG Website, they must create a user account.

Pre-Condition:

1. The user must know the correct URL (dirsweb.php).

Post-Condition:

1. The user creates an account ready to be enabled by a DIRS Staff member.

Assumptions:

1. The user is an individual that should have an account.

Normal Course:

1. User types the correct URL into his/her web browser (URL TBD).
2. User clicks the "Create a New Account" link.
3. System displays new account screen/form with required fields indicated in red.
4. User inputs all required information into the webform.
5. User clicks the "Create Account" button.
6. System validates that all required fields are entered.
7. System adds user information to the Database with "Enabled" field set to "N".
8. System sends validation e-mail to user based on e-mail address entered.
9. System sends e-mail notification to DIRS Staff indicating a new account has been created.

Alternate Course:

6a. System is not able to validate that all required fields are entered
6a1. System prompts user to fill in all required data fields
6a2. User inputs missing filed data
6a3. Goto #5

Includes: n/a

Note: Required input fields are email address, password, first name, last name, company name, street address, city, state, zip code, and phone number.

Files: dirsweb.php, newcust2.php

2.2.3 Enable User Account

Use Case Number: **3**

Use Case Name: **Enable User Account**

Primary Actor: user (DIRS Staff Member)
Secondary Actor: n/a

Description:
When a new user account is created, it must be enabled by a DIRS staff member to become a functional account.

Pre-Condition:
1. A user account needs to be enabled

Post-Condition:
1. The user account is enabled.

Assumptions:
1. The user is a DIRS staff member, with an admin enabled account, who needs to enable a user account.

Normal Course:
1. A Notifcaiton e-mail from the system, identifying a new user account has been created, is received by the DIRS Staff *<<include>> Create New User Account*
2. DIRS Staff user logs onto the system *<<include>> Log On To System*
3. User clicks on the "Display All Users" link
4. User locates the user account to enable on the displayed list
5. User selects the "Enable" link
6. System updates databse
7. System displays confirmation message to user
8. System generates e-mail to customer account with notifcation of account enablement
9. System provides a links to return to the user display page and to the admin options page

Alternate Course:
2a. The DIRS Staff user determines that the user account should not be enabled.
2a1. The DIRS Staff user may choose to manually send an e-mail to the user account denying account enablement, or take no action, or delete the account *<<include>> Delete Account*

Includes: Logon to System, Create New User Account, Delete Account

Files: adminoptions.php, usersdisplay.php, enable.php

2.2.4 Disable User Account

Use Case Number: **4**

Use Case Name: **Disable User Account**

Primary Actor: user (DIRS Staff Member)
Secondary Actor: n/a

Description:
If a user account should no longer be enabled, a DIRS staff member can disable the account.

Pre-Condition:
1. A user account must exist and be enabled

Post-Condition:
1. The user account is disabled.

Assumptions:
1. The user is a DIRS staff member, with an admin enabled account, who needs to disable a user account.

Normal Course:
1. A DIRS Staff user logs onto the system. *<<include>> Log On To System*
2. User clicks on the "Display All Users" link
3. User locates the user account to disable on the displayed list
4. User selects the "Edit" link
5. User unchecks the "Enabled" checkbox
6. User clicks the "Edit Account" button
7. System updates database
8. System displays confirmation message to user

Alternate Course:
7a. System displays error message if a database update error occurs.

Includes: Log On To System

Files: adminoptions.php, usersdisplay.php, edituser.php

2.2.5 Download Software

Use Case Number: **5**

Use Case Name: **Download Software**

Primary Actor: user
Secondary Actor: n/a

Description:
When a user needs to obtain new software they must log on to the website and download the software.

Pre-Condition:
1. A user account must exist and be enabled

Post-Condition:
1. The user successfully downloads the desired software.

Assumptions:
1. The user has a valid and enabled user account.

Normal Course:
1. The user logs onto the website *<<include>> Log On To System*
2. User selects the "Edit your computers and software" link
3. System displays the users list of computers
4. User selects the "Subscriptions" link for a selected computer
5. System displays the current list of software assigned to that computer
6. If the user's version of any product is up to date, the system displays an "Up to Date" message for that product
7. If the user's version of any product is not up to date, the system displays a "Download" link for the new version
8. The user clicks the "Download" link
9. The system downloads the software

Alternate Course:
none

Includes: Log On To System

Files: useroptions.php, computeredit.php, managesw.php, subscriptionadd.php, download.php

2.2.6 Add Computer to User Account

Use Case Number: **6**

Use Case Name: **Add Computer to User Account**

Primary Actor: user
Secondary Actor: n/a

Description:
A user can add additional computers to their user account.

Pre-Condition:
1. A user account must exist and be enabled

Post-Condition:
1. The user sucessfully adds a new computer to their account

Assumptions:
1. The user has a valid and enabled user account.

Normal Course:
1. The user logs onto the website *<<include>> Log On To System*
2. User selects the "Add a Computer to your Account" link
3. User selects the operating system and platform for the computer from the drop down boxes in the webform
4. User enters a computer name into the web form
5. User selects the "Add Computer" button
6. System updates database with new computer information

Alternate Course:

5a. System is unable to update database

5a1. System displays error message to user

Includes: Log On To System

Files: useroptions.php, computeradd.php

2.2.7 Edit User Account

Use Case Number: **7**

Use Case Name: **Edit User Account**

Primary Actor: user
Secondary Actor: n/a

Description:
A user can update their contact information in their user account.

Pre-Condition:
1. A user account must exist and be enabled

Post-Condition:
1. The user sucessfully updates their account contact information

Assumptions:
1. The user has a valid and enabled account.

Normal Course:
1. The user logs onto the website *<<include>> Log On To System*
2. User selects the "Edit you account information" link
3. User inputs changes into the webform
4. User selects the "Update Account" button
5. System validates that all required fields are entered.
6. System updates database with updated contact information

Alternate Course:
5a. System is not able to validate that all required fields are entered
5a1. System prompts user to fill in all required data fields
5a2. User inputs missing filed data
5a3. Goto #3

Includes: Log On To System

Files: useroptions.php, useredit.php

2.2.8 Log Off Website

Use Case Number: **8**

Use Case Name: **Log Off Website**

Primary Actor: user
Secondary Actor: n/a

Description:
When a user is finished using the DIRS website, they need to logoff the system.

Pre-Condition:

1. A user account must exist and be enabled
2. The user is logged on to the website

Post-Condition:

1. The user sucessfully logs off the website

Assumptions:

1. The user has a valid and enabled user account.

Normal Course:

1. The user clicks the "Logout" link
2. The system ends the current session
3. The system displays a confirmation message to the user

Alternate Course:
N/A

Includes: N/A

Files: logout.php

2.2.9 Delete User Account

Use Case Number: **9**

Use Case Name: **Delete User Account**

Primary Actor: user (DIRS Staff Member)
Secondary Actor: n/a

Description:
If a user account should no longer exist, a DIRS staff member can delete the account.

Pre-Condition:
1. A user account must exist.

Post-Condition:
1. The user account is deleted.

Assumptions:
1. The user is a DIRS staff member, with an admin enabled account, who needs to delete a user account.

Normal Course:
1. A DIRS Staff user logs onto the system. *<<include>> Log On To System*
2. User clicks on the "Display All Users" link
3. User locates the user account to delete on the displayed list
4. User selects the "Remove" link
5. System updates database
6. System displays confirmation message to user

Alternate Course:
5a. System displays error message if a database update error occurs.

Includes: Log On To System

Files: adminoptions.php, usersdisplay.php, removeuser.php

2.2.10 Grant Admin Privileges to Account

Use Case Number: **10**

Use Case Name: **Grant Admin Privilege to Account**

Primary Actor: user (DIRS Staff Member)
Secondary Actor: n/a

Description:
If an account should be granted admin privileges, a DIRS staff member can update the account.

Pre-Condition:

1. A user account must exist.

Post-Condition:

1. The user account is given admin privileges.

Assumptions:

1. The user is a DIRS staff member, with an admin enabled account, who needs to grant admin privileges to another account.

Normal Course:

1. A DIRS Staff user logs onto the system. *<<include>> Log On To System*
2. User clicks on the "Display All Users" link
3. User locates the user account to delete on the displayed list
4. User selects the "Grant Admin" link
5. System updates database
6. System displays confirmation message to user

Alternate Course:
5a. System displays error message if a database update error occurs.

Includes: Log On To System

Files: adminoptions.php, usersdisplay.php, removeuser.php

2.2.11 Add Operating System

Use Case Number: **11**

Use Case Name: **Add Operating System**

Primary Actor: user (DIRS Staff Member)
Secondary Actor: n/a

Description:
When a new Operating System should be added to the database

Pre-Condition:

1. A new operating system should be added to the existing list in the database

Post-Condition:

1. The operating system is added to the database

Assumptions:

1. The user is a DIRS staff member, with an admin enabled account.

Normal Course:

1. A DIRS Staff user logs onto the system. *<<include>> Log On To System*
2. User clicks on the "Maintain Operating Systems in the Database" link
3. User clicks on the "Add a New Operating System to the Database" link
4. User enters the Operating System name into the web form
5. User clicks the "Add Operating System" button
6. System updates the database
7. System displays confirmation message to user

Alternate Course:

6a. System displays error message if a database update error occurs.

Includes: Log On To System

Files: adminoptions.php, osview.php, osadd.php

2.2.12 Delete Operating System

Use Case Number: **12**

Use Case Name: **Delete Operating System**

Primary Actor: user (DIRS Staff Member)
Secondary Actor: n/a

Description:
When an Operating System in the database should be deleted

Pre-Condition:
1. An operating system should be deleted from the database

Post-Condition:
1. The operating system is removed from the database

Assumptions:
1. The user is a DIRS staff member, with an admin enabled account.

Normal Course:
1. A DIRS Staff user logs onto the system. *<<include>> Log On To System*
2. User clicks on the "Maintain Operating Systems in the Database" link
3. User locates the operating system to delete on the displayed list
4. User clicks on the "Delete" link
5. System updates the database
6. System displays confirmation message to user

Alternate Course:
6a. System displays error message if a database update error occurs.

Includes: Log On To System

Files: adminoptions.php, osview.php, osdelete.php

2.2.13 Add Platform

Use Case Number:	**13**
Use Case Name:	**Add Platform**
Primary Actor: **Secondary Actor:**	user (DIRS Staff Member) n/a
Description:	When a new Platform should be added to the database
Pre-Condition:	**1.** A new platform should be added to the existing list in the database
Post-Condition:	**1.** The platform is added to the database
Assumptions:	**1.** The user is a DIRS staff member, with an admin enabled account.
Normal Course:	**1.** A DIRS Staff user logs onto the system. *<<include>> Log On To System* **2.** User clicks on the "Maintain Platforms in the Database" link **3.** User clicks on the "Add a New Platform to the Database" link **4.** User enters the Platform name into the web form **5.** User clicks the "Add Platform" button **6.** System updates the database **7.** System displays confirmation message to user
Alternate Course:	**6a.** System displays error message if a database update error occurs.
Includes:	Log On To System

Files: adminoptions.php, platformview.php, platformadd.php

2.2.14 Delete Platform

Use Case Number: **14**

Use Case Name: **Delete Platform**

Primary Actor: user (DIRS Staff Member)
Secondary Actor: n/a

Description:
When a Platform in the database should be deleted

Pre-Condition:
1. A platform should be deleted from the database

Post-Condition:
1. The platform is removed from the database

Assumptions:
1. The user is a DIRS staff member, with an admin enabled account.

Normal Course:
1. A DIRS Staff user logs onto the system. *<<include>> Log On To System*
2. User clicks on the "Maintain Platforms in the Database" link
3. User locates the platform to delete on the displayed list
4. User clicks on the "Delete" link
5. System updates the database
6. System displays confirmation message to user

Alternate Course:
6a. System displays error message if a database update error occurs.

Includes: Log On To System

Files: adminoptions.php, platformview.php, platformdelete.php

2.2.15 Add Software Information

Use Case Number: **15**

Use Case Name: **Add Software Information**

Primary Actor: user (DIRS Staff Member)
Secondary Actor: n/a

Description:
When new Software information should be added to the database

Pre-Condition:
1. New software information should be added to the existing list in the database

Post-Condition:
1. The software information is added to the database

Assumptions:
1. The user is a DIRS staff member, with an admin enabled account.

Normal Course:
1. A DIRS Staff user logs onto the system. *<<include>> Log On To System*
2. User clicks on the "Maintain Software and/or Version Information in the Database" link
3. User clicks on the "Add New Software to the Database" link
4. User enters the Software Information into the web form
5. User clicks the "Add Software" button
6. System updates the database
7. System displays confirmation message to user

Alternate Course:
6a. System displays error message if a database update error occurs.

Includes: Log On To System

Files: adminoptions.php, softwareview.php, softwareadd.php

2.2.16 Delete Software Information

Use Case Number: **16**

Use Case Name: **Delete Software Information**

Primary Actor: user (DIRS Staff Member)
Secondary Actor: n/a

Description:
When Software Information in the database should be deleted

Pre-Condition:
1. A Software selection should be deleted from the database

Post-Condition:
1. The software information is removed from the database

Assumptions:
1. The user is a DIRS staff member, with an admin enabled account.

Normal Course:
1. A DIRS Staff user logs onto the system. *<<include>> Log On To System*
2. User clicks on the "Maintain Software and/or Version Information in the Database" link
3. User locates the software to delete on the displayed list
4. User clicks on the "Delete" link
5. System updates the database
6. System displays confirmation message to user

Alternate Course:
6a. System displays error message if a database update error occurs.

Includes: Log On To System

Files: adminoptions.php, softwareview.php, softwaredelete.php

2.2.17 Add Software Version

Use Case Number: **17**

Use Case Name: **Add Software Version**

Primary Actor: user (DIRS Staff Member)
Secondary Actor: n/a

Description:
When a new Software version should be added to the database

Pre-Condition:

1. A new software version should be added to the existing list in the database

Post-Condition:

1. The software version is added to the database

Assumptions:

1. The user is a DIRS staff member, with an admin enabled account.

Normal Course:

1. A DIRS Staff user logs onto the system. *<<include>> Log On To System*
2. User clicks on the "Maintain Software and/or Version Information in the Database" link
3. User locates the product requiring a new version on the displayed list
4. User clicks the "Version Info" link
5. User clicks the "Add a New Version to this Product" link
6. User enters the Version Information into the web form
7. User clicks the "Add Version" button
8. System updates the database
9. System displays confirmation message to user

Alternate Course:

8a. System displays error message if a database update error occurs.

Includes: Log On To System

Files: adminoptions.php, softwareview.php, versionview.php, versionadd.php

2.2.18 Delete Software Version

Use Case Number: **18**

Use Case Name: **Delete Software Version**

Primary Actor: user (DIRS Staff Member)
Secondary Actor: n/a

Description:
When a Software version should be deleted from the database

Pre-Condition:
1. A software version should be deleted from the existing list in the database

Post-Condition:
1. The software version is deketed from the database

Assumptions:
1. The user is a DIRS staff member, with an admin enabled account.

Normal Course:
1. A DIRS Staff user logs onto the system. *<<include>> Log On To System*
2. User clicks on the "Maintain Software and/or Version Information in the Database" link
3. User locates the product requiring a new version on the displayed list
4. User clicks the "Version Info" link
5. User locates the version that needs to be deleted on the displayed list
6. User clicks the "Delete" link
7. System updates the database
8. System displays confirmation message to user

Alternate Course:
7a. System displays error message if a database update error occurs.

Includes: Log On To System

Files: adminoptions.php, softwareview.php, versionview.php, versiondelete.php

2.2.19 Admin Edit User Account

Use Case Number: **19**

Use Case Name: **Admin Edit User Account**

Primary Actor: user (DIRS Staff Member)
Secondary Actor: n/a

Description:
If a user account needs to be edited (i.e. for a password reset), a DIRS staff member can edit the account.

Pre-Condition:

1. A user account must exist

Post-Condition:

1. The user account is edited

Assumptions:

1. The user is a DIRS staff member, with an admin enabled account, who needs to edit a user account.

Normal Course:

1. A DIRS Staff user logs onto the system. *<<include>> Log On To System*
2. User clicks on the "Display All Users" link
3. User locates the user account to disable on the displayed list
4. User selects the "Edit" link
5. User makes any edits necessary in the web form
6. User clicks the "Edit Account" button
7. System updates database
8. System displays confirmation message to user

Alternate Course:

7a. System displays error message if a database update error occurs.

Includes: Log On To System

Files: adminoptions.php, usersdisplay.php, edituser.php

2.2.20 Edit a Computer in a User Account

Use Case Number: **20**

Use Case Name: **Edit a Computer in a User Account**

Primary Actor: user
Secondary Actor: n/a

Description:
A user can edit the computers in their user account.

Pre-Condition:
1. A user account must exist and be enabled
2. A user account must have at least one computer assigned to it

Post-Condition:
1. The user sucessfully edits an exiting computer in their account

Assumptions:
1. The user has a valid and enabled user account.

Normal Course:
1. The user logs onto the website *<<include>> Log On To System*
2. User selects the "Edit your Computers and Software" link
3. User locates the computer they want to edit on the list of their computers
4. User selects the "Edit" link
5. User makes any changes via the webform
6. User selects the "Edit Computer" button
7. System updates the database
8. System displays confirmation message to user

Alternate Course:

7a. System is unable to update database

7a1. System displays error message to user

Includes: Log On To System

Files: useroptions.php, computeredit.php, editcomp.php

2.2.21 Delete a Computer in a User Account

Use Case Number: **21**

Use Case Name: **Delete a Computer in a User Account**

Primary Actor: user
Secondary Actor: n/a

Description:
A user can delete a computer in their user account.

Pre-Condition:
1. A user account must exist and be enabled
2. A user account must have at least one computer assigned to it

Post-Condition:
1. The user sucessfully deletes an exiting computer in their account

Assumptions:
1. The user has a valid and enabled user account.

Normal Course:
1. The user logs onto the website *<<include>> Log On To System*
2. User selects the "Edit your Computers and Software" link
3. User locates the computer they want to edit on the list of their computers
4. User selects the "Remove" link
5. System updates the database
6. System displays confirmation message to user

Alternate Course:
5a. System is unable to update database
5a1. System displays error message to user

Includes: Log On To System

Files: useroptions.php, computeredit.php, removecomp.php

2.2.22 Add New Subscription to Computer

Use Case Number: **22**

Use Case Name: **Add new subscription to computer**

Primary Actor: user
Secondary Actor: n/a

Description:
A user can add a software version (subscription) to the computers in their account

Pre-Condition:
1. A user account must exist and be enabled
2. A user account must have at least one computer assigned to it

Post-Condition:
1. The user sucessfully adds a subscription to their account

Assumptions:
1. The user has a valid and enabled user account.

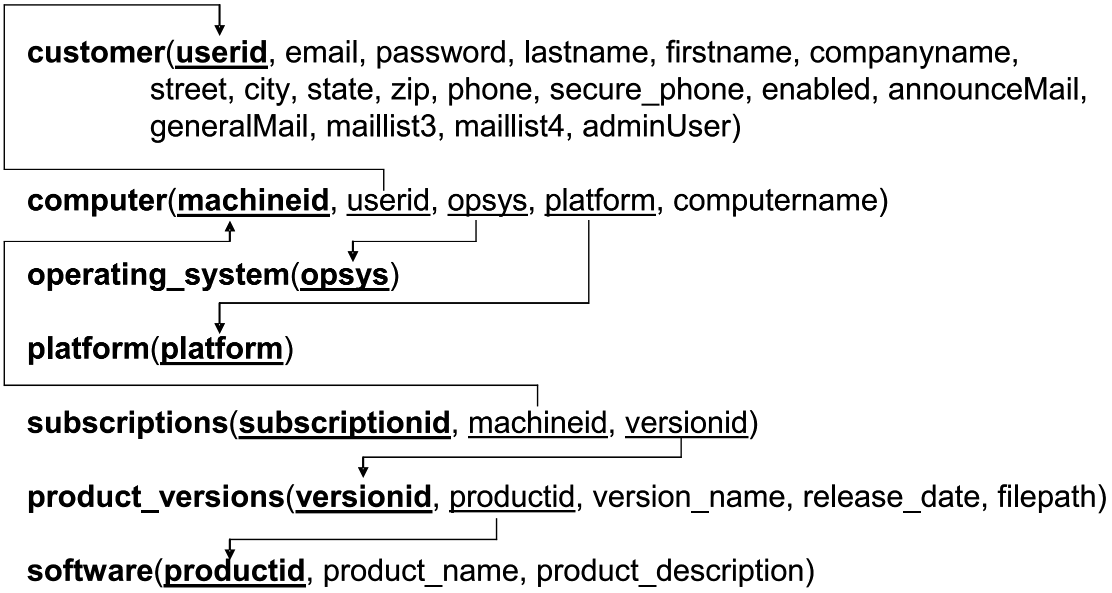
Normal Course:
1. The user logs onto the website *<<include>> Log On To System*
2. User selects the "Edit your Computers and Software" link
3. User locates the computer they want to edit on the list of their computers
4. User selects the "Subscriptions" link for a selected computer
5. System displays the current list of software assigned to that computer
6. User clicks on the "Add a Subscription" link
7. User fills out webform
8. User clicks on the "Add Subscription" button
9. System updates database
10. System displays confirmation message to user

Alternate Course:
9a. System is unable to update database
9a1. System displays error message to user

Includes: Log On To System

Files: useroptions.php, computeredit.php, managesw.php, subscriptionadd.php, addsub.php

2.3 Relation Mapping



customer(**userid**, email, password, lastname, firstname, companyname, street, city, state, zip, phone, secure_phone, enabled, announceMail, generalMail, maillist3, maillist4, adminUser)

computer(**machineid**, userid, opsys, platform, computername)

operating_system(**opsys**)

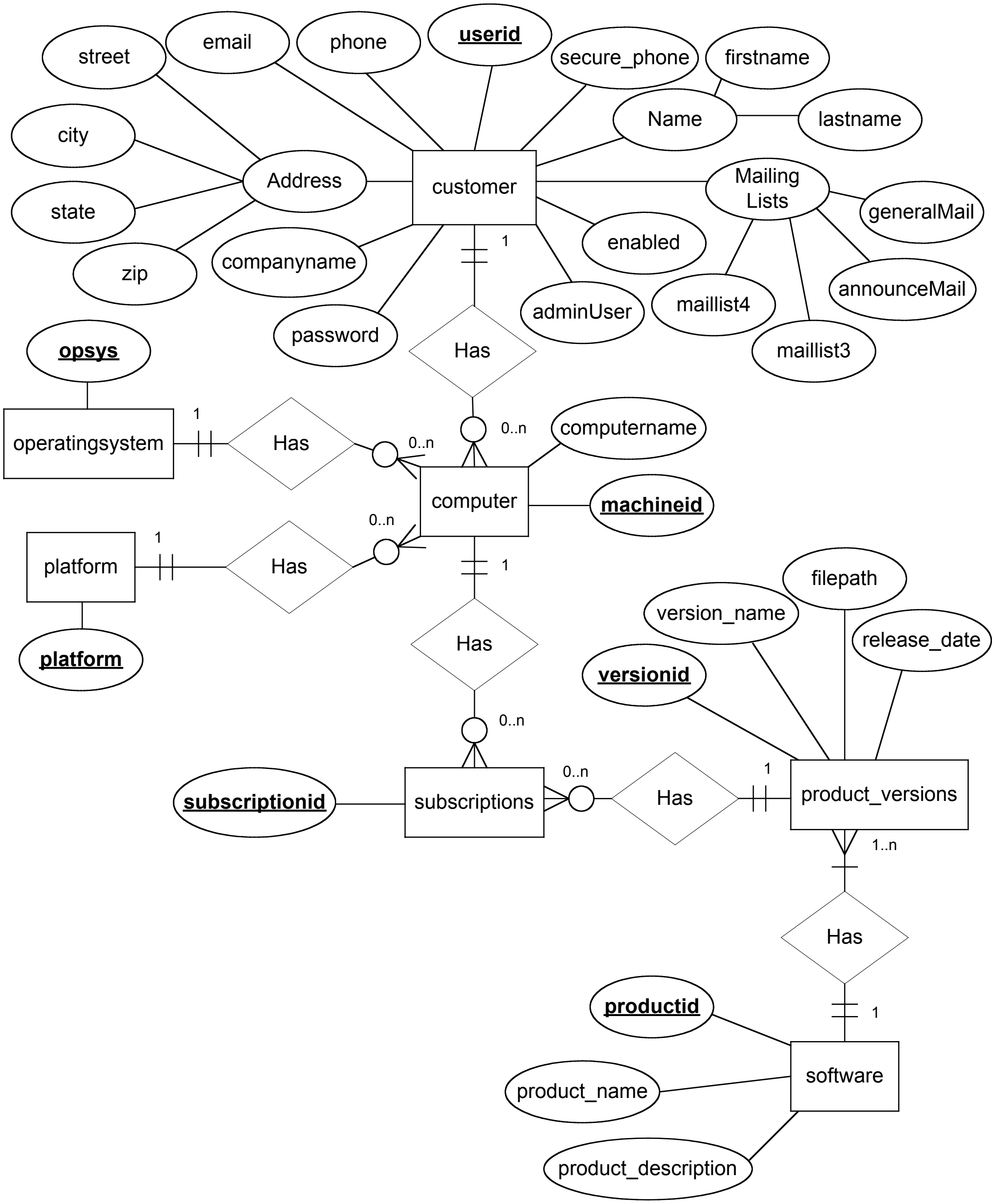
platform(**platform**)

subscriptions(**subscriptionid**, machineid, versionid)

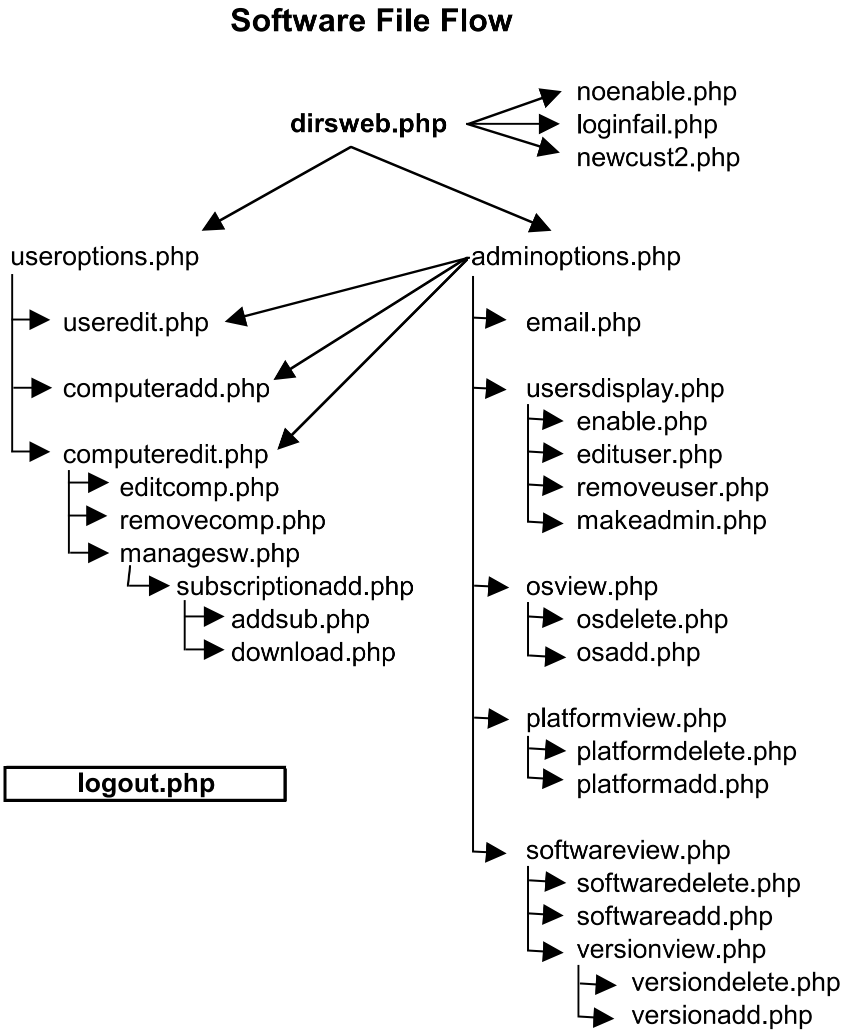
product_versions(**versionid**, productid, version_name, release_date, filepath)

software(**productid**, product_name, product_description)

2.4 Entity Relationship Diagram



2.5 Software File Flow



2.6 File Descriptions

dirsweb.php – This is the opening page for the site. A user can create a new account or log into their existing account. The system checks for a valid email address and password and gives the user three attempts to make a successful login. If the login attempts fail on an email address within the database, the system will disable the account and send an email to a myDIRSIG staff member and the user. If the login fails because the email address is not in the system, an email is sent only to the myDIRSIG staff to alert them that an unsuccessful login was attempted.

noenable.php – If a user attempts to log into the site and their account has not yet been enabled by a myDIRSIG staff member, then this page will display informing the user of their account status.

loginfail.php – If a user fails to successfully login three times, they are directed to this page informing them that they have exceeded the number of login attempts allowed and that their account has been disabled.

newcust2.php – This page allows a user to create a new account. The user is presented with a web form to enter their user information into and the account is then created, but not yet enabled. An email is then sent to the user and a myDIRSIG staff member notifying them about the new account creation. The myDIRSIG staff member would then have to login and enable the user's account for the user to be able to access the website.

useroptions.php – This file presents the available options that an enabled user can perform as a series of links.

useredit.php – A user can edit his/her own account information, including their email address, password and mailing list preferences, using the presented web form.

computeradd.php – A user can add a computer to his/her account by using this form. It allows the user to select the computer's platform and operating system, and to provide a name for the computer for reference when adding/updating software.

computeredit.php – This file displays all of the current user's computers in a table format and provides a links to edit or remove each computer, or view the subscriptions associated with each computer.

editcomp.php – This file allows the user to update the operating system, platform and/or computer name for the computer selected via the "computeredit.php" page.

removecomp.php – This file deletes the computer selected from the "computeredit.php" page.

managesw.php – This file displays all the subscriptions for the computer selected from the "computeredit.php" page. The table displays the software version that the user currently has and the most recent version of the software product and provides a download link if the two versions are not synchronized. A link to add a new subscription is also present on this page to serve two purposes, first, if the user already has software and just wants to update their account on myDIRSIG, and second, if the user wants to download a new software product that they do not have.

addsub.php – This file adds a subscription to the database for a user who already has software on their computer, but would like their myDIRSIG account to reflect that information. This page does not download software, it only does a database update.

download.php – This file allows the user to download new software to their computer and update their account with the subscription information associated with the software version being downloaded.

adminoptions.php – This file presents the available options that an enabled user with an admin account can perform as a series of links.

email.php – This file presents a simple web form for an admin user to send an email to all users, general mail list subscribers, or admin mail list subscribers. The email will be sent "from" the admin user logged in.

usersdisplay.php – This file displays all the myDIRSIG users in a table format to an admin user. Each displayed user account can be enabled, edited, removed, or granted admin privileges via links within the table.

enable.php – This file enables a user account when an admin user clicks the "Enable" link for a user account listed on the "usersdisplay.php" page. It also displays a confirmation (or error) message when the database is updated.

edituser.php – This file enables an admin user to edit the user account selected from the "usersdisplay.php" page. A web form is used and any field in the customer table can be updated.

removeuser.php – This file deletes a user account when an admin user clicks the "Remove" link for a user account listed on the "usersdisplay.php" page. When a user account is deleted the user's computers and subscriptions are also deleted. A confirmation (or error) message is displayed when the database is updated.

makeadmin.php –This file grants admin privileges to a user account when an admin user clicks the "Grant Admin" link for a user account listed on the "usersdisplay.php" page. It also displays a confirmation (or error) message when the database is updated.

osview.php – This file displays the list of operating systems available in the database in table form. From this page, an admin user can add a new operating system or remove an existing operating system.

osdelete.php – This file removes the operating system selected by an admin user from the "osview.php" page by clicking on the "Remove" link. A confirmation (or error) message is displayed when the database is updated.

osadd.php – This file provides a web form for an admin user to add a new operating system to the list of valid operating systems in the database. A confirmation (or error) message is displayed when the database is updated.

platformview.php – This file displays the list of platforms available in the database in table form. From this page, an admin user can add a new platform or remove an existing platform.

platformdelete.php – This file removes the platform selected by an admin user from the "platformview.php" page by clicking on the "Remove" link. A confirmation (or error) message is displayed when the database is updated.

platformadd.php – This file provides a web form for an admin user to add a new platform to the list of valid platforms in the database. A confirmation (or error) message is displayed when the database is updated.

softwareview.php – This file displays the current list of software products and product descriptions in a table to an admin user. The admin user can remove any product, view the versions of each product and add a new product from this page.

softwaredelete.php – This file removes the software product selected by an admin user from the "softwareview.php" page by clicking on the "Remove" link. A confirmation (or error) message is displayed when the database is updated.

softwareadd.php – This file provides a web form for an admin user to add a new software product to the list of valid products in the database. A confirmation (or error) message is displayed when the database is updated.

versionview.php – This file displays the current list of software versions and version descriptions in a table to an admin user. The admin user can remove any version or add a new version from this page.

versiondelete.php – This file removes the version selected by an admin user from the "versionview.php" page by clicking on the "Remove" link. A confirmation (or error) message is displayed when the database is updated.

versionadd.php – This file provides a web form for an admin user to add a new version to the list of valid product versions in the database. A confirmation (or error) message is displayed when the database is updated.

logout.php – This file ends the current web session and logs the user out of the myDIRSIG site. A link to the "dirsweb.php" login page is provided along with a confirmation message that the user has been logged out.

3 Future Enhancements

In the course of developing the myDIRSIG website, several items were determined to be beyond the scope of this project and were added to a list of possible future enhancements. The main focus of this project was the database creation and website functionality. The following list examines each of these future enhancement possibilities.

- The use of Cascading Style Sheets (CSS) to create a uniform toolbar throughout the myDIRSIG site, customized for admin and regular users. CSS could also be used to update or replace the existing HTML tables which are currently used throughout the site.

- Two blank mailing list fields were included in the "customer" table for future mailing list expansion without a disruption to the database structure.

- A download counter to be added to the "product_versions" table to keep track of the number of downloads and database updates for users that already have the software for each version.

- A future download page to update all the computers assigned to a user's account so that a user can download the software once, distribute the file to their other computers, and then update their computer information within the database.

- On the software version page, where the user can view if their current software versions are up to date or if new versions are available, an "Update All" link to bring all versions up to date. Currently the user must perform the updates one at a time for each version.

- The creation, assigning and tracking of a software license key. This was originally part of the project proposal, but in developing the product for the customer, it was decided to have the customer handle this issue separately after the initial implementation as the requirements have not yet been fully defined.

- A session timeout so that a user account is automatically logged off after a certain set amount of inactivity.

4 References

Hoffer, Jeffrey A., Mary B. Prescott and Fred R. McFadden. Modern Database Management. Upper Saddle River: Prentice Hall, 2002.

Lecky-Thompson, Ed, Heow Edie-Goodmand, Steven D. Nowicki, and Alec Cove. Professional PHP 5. Indianapolis: Wiley Publishing, Inc., 2005.

Spainhour, Stephen and Robert Eckstein. Webmaster in a Nutshell. Sebastopol: O'Reilly Media, Inc., 1999.

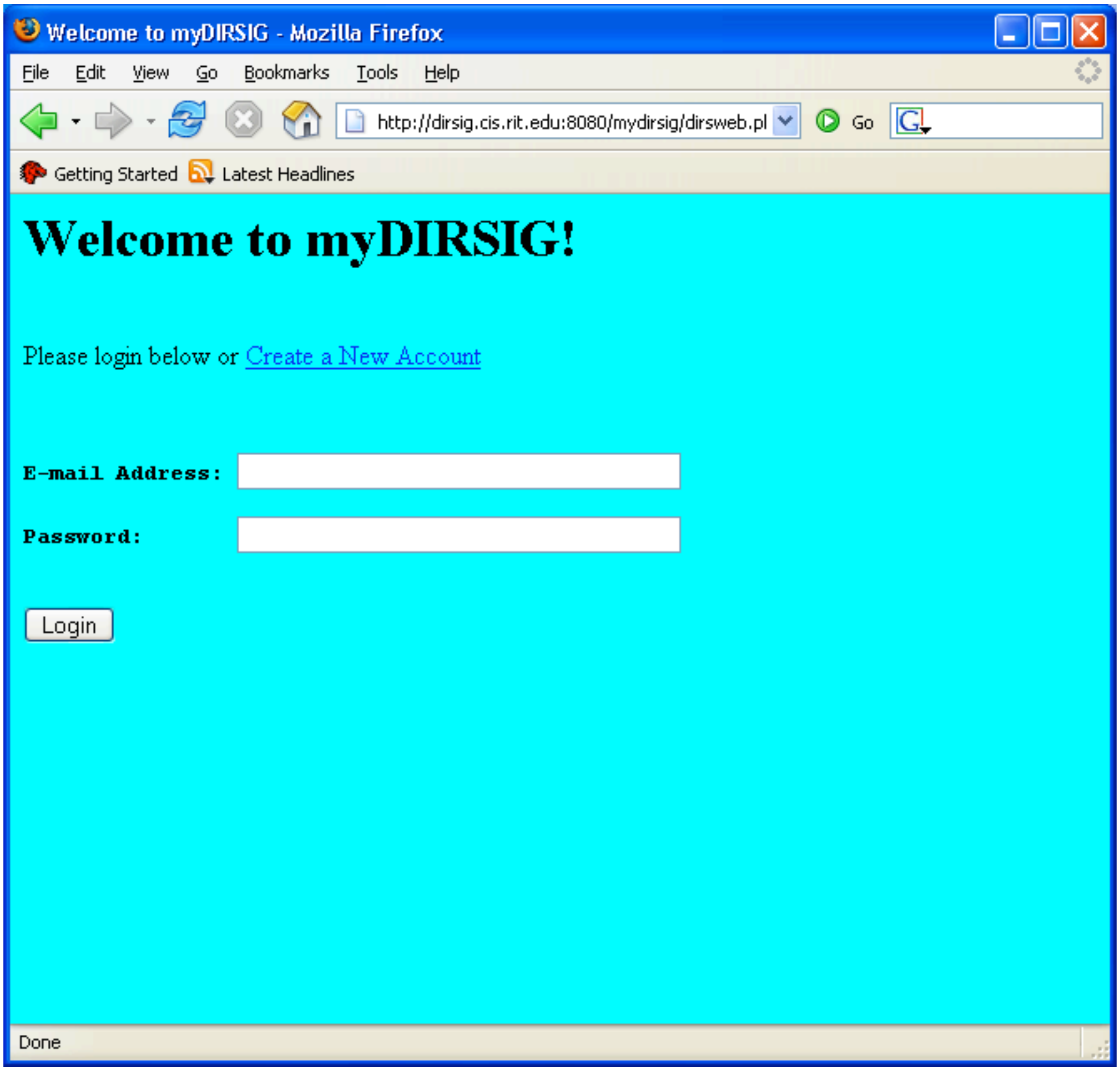
Trachtenberg, Adam. Upgrading to PHP 5. Sebastopol: O'Reilly Media, Inc., 2004.

Williams, Hugh E. and David Lane. Web Database Applications with PHP and MySQL. Sebastopol: O'Reilly Media, Inc., 2002.

5 Appendices

5.1 Screen Shots

5.1.1 Login Screen (dirsweb.php)


Welcome to myDIRSIG - Mozilla Firefox
File Edit View Go Bookmarks Tools Help
http://dirsig.cis.rit.edu:8080/mydirsig/dirsweb.pl
Go
Getting Started
Latest Headlines

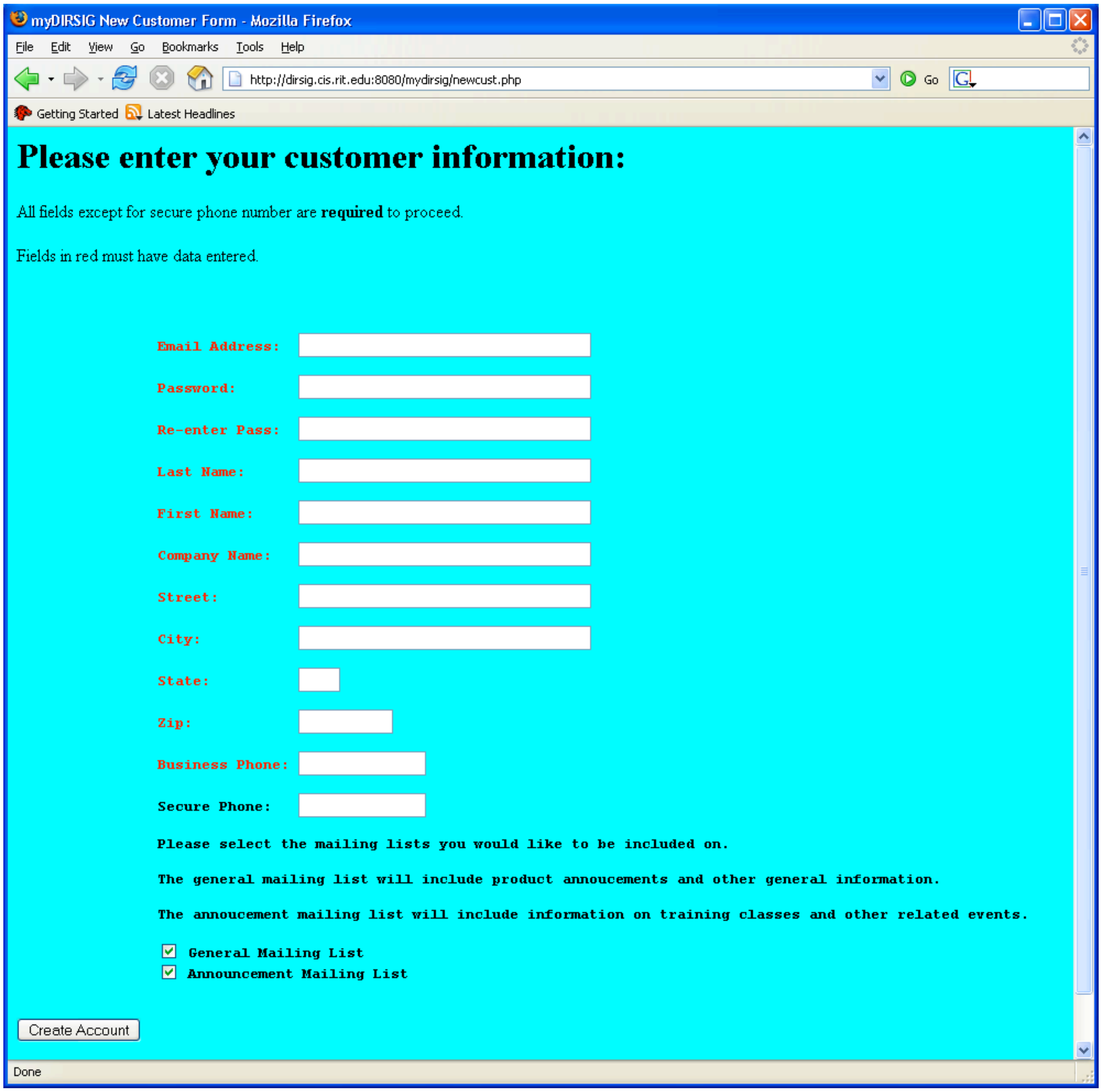
Welcome to myDIRSIG!

Please login below or Create a New Account

E-mail Address:

Password:

5.1.2 Create New Account (newcust.php)



myDIRSIG New Customer Form - Mozilla Firefox
File Edit View Go Bookmarks Tools Help
http://dirsig.cis.rit.edu:8080/mydirsig/newcust.php
Go
Getting Started
Latest Headlines
Create Account
Done

Please enter your customer information:

All fields except for secure phone number are **required** to proceed.

Fields in red must have data entered.

Email Address:	
Password:	
Re-enter Pass:	
Last Name:	
First Name:	
Company Name:	
Street:	
City:	
State:	
Zip:	
Business Phone:	
Secure Phone:	

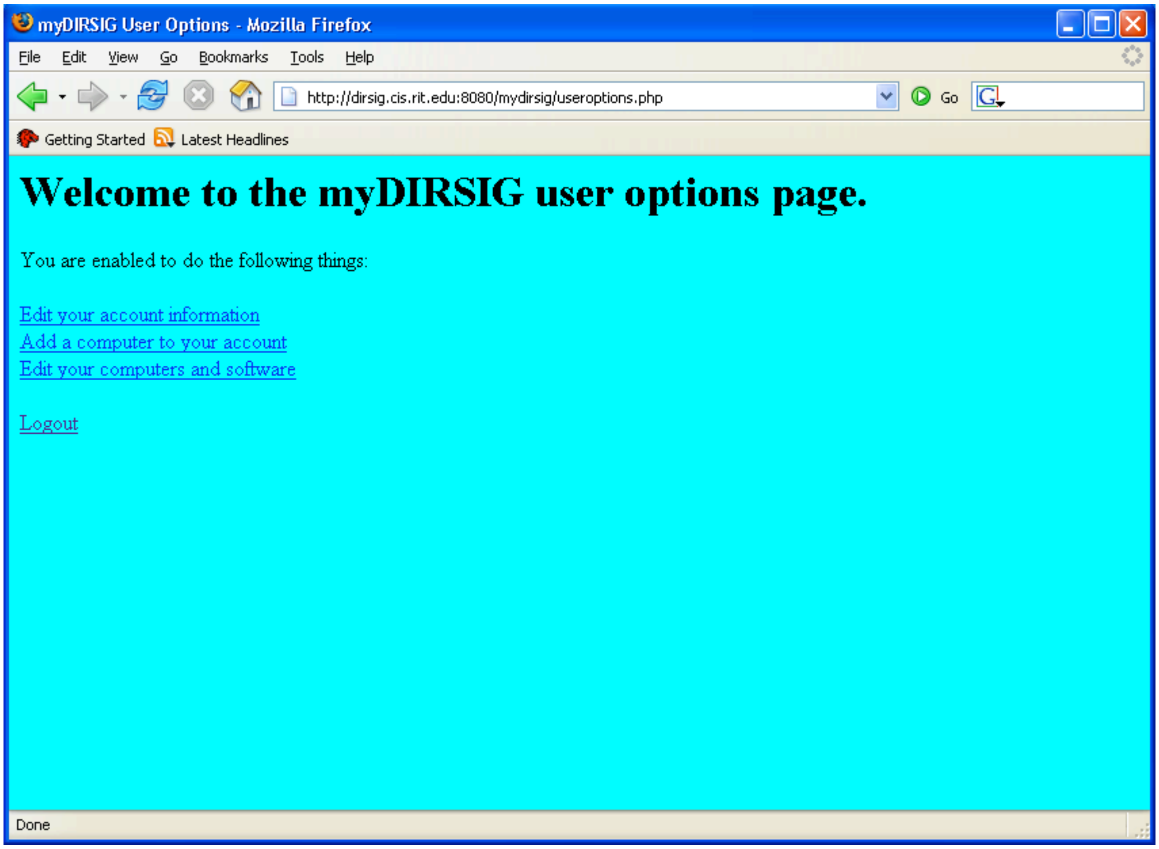
Please select the mailing lists you would like to be included on.

The general mailing list will include product annoucements and other general information.

The annoucement mailing list will include information on training classes and other related events.

☑ General Mailing List
☑ Announcement Mailing List

5.1.3 User Options Page (useroptions.php)


myDIRSIG User Options - Mozilla Firefox
File Edit View Go Bookmarks Tools Help
http://dirsig.cis.rit.edu:8080/mydirsig/useroptions.php
Go
Getting Started
Latest Headlines
Done

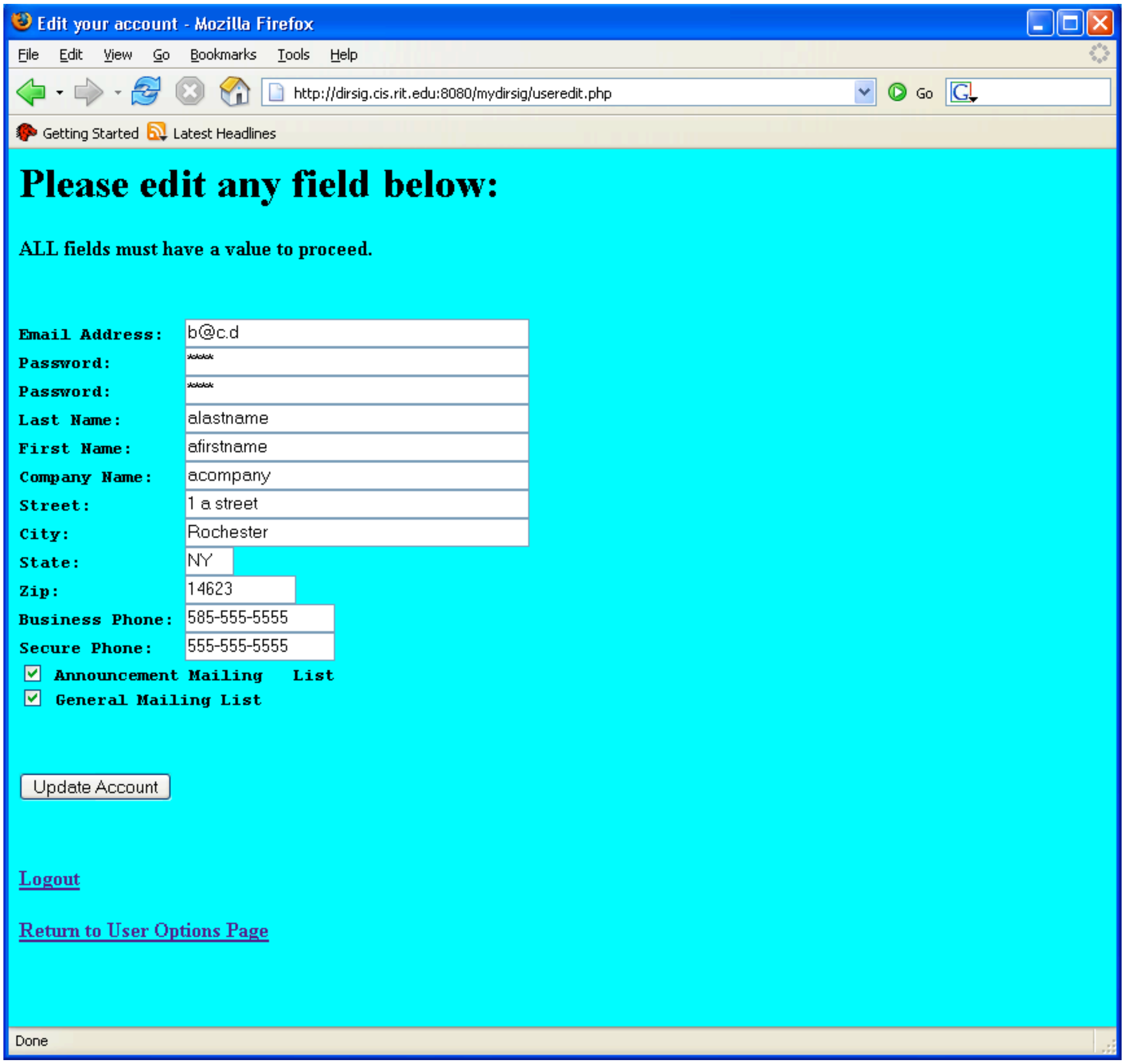
Welcome to the myDIRSIG user options page.

You are enabled to do the following things:

Edit your account information
Add a computer to your account
Edit your computers and software

Logout

5.1.4 Edit User Account (useredit.php)



Edit your account - Mozilla Firefox
File Edit View Go Bookmarks Tools Help
http://dirsig.cis.rit.edu:8080/mydirsig/useredit.php
Go
Getting Started
Latest Headlines

Please edit any field below:

ALL fields must have a value to proceed.

Email Address:	b@c.d
Password:	****
Password:	****
Last Name:	alastname
First Name:	afirstname
Company Name:	acompany
Street:	1 a street
City:	Rochester
State:	NY
Zip:	14623
Business Phone:	585-555-5555
Secure Phone:	555-555-5555

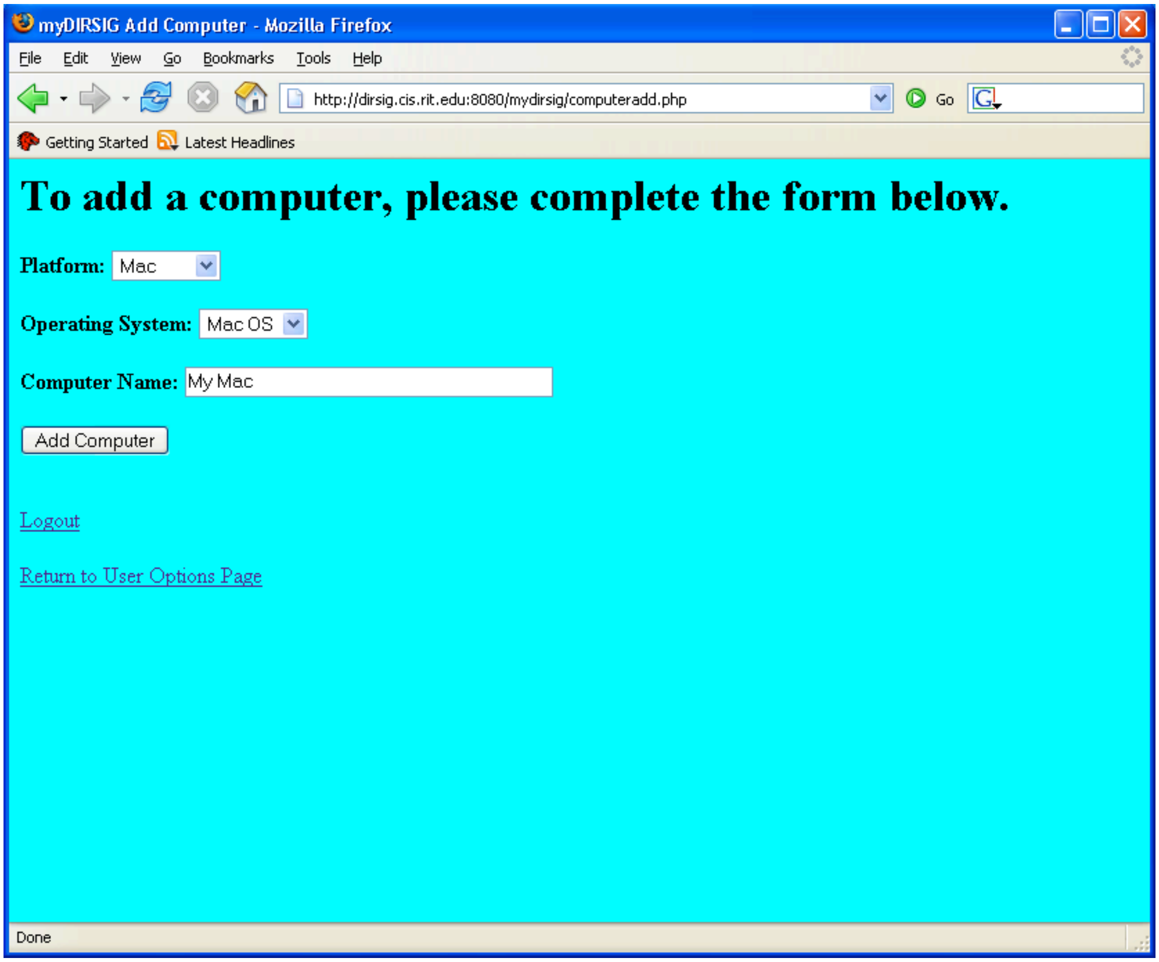
☑ Announcement Mailing List

☑ General Mailing List

Logout

Return to User Options Page

5.1.5 Add Computer to Account (computeradd.php)


myDIRSIG Add Computer - Mozilla Firefox
File Edit View Go Bookmarks Tools Help
http://dirsig.cis.rit.edu:8080/mydirsig/computeradd.php
Go
Getting Started
Latest Headlines
Done

To add a computer, please complete the form below.

Platform: Mac

Operating System: Mac OS

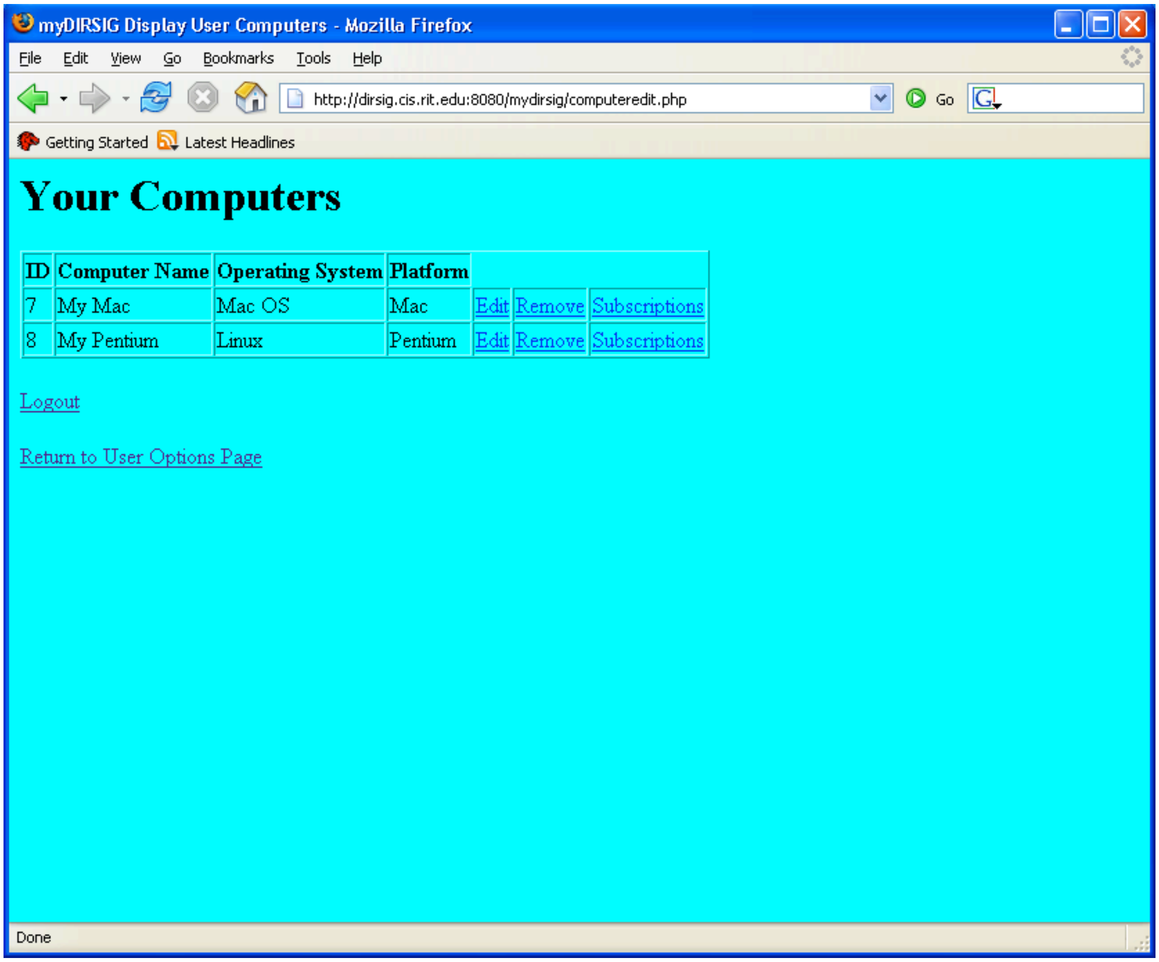
Computer Name: My Mac

Add Computer

Logout

Return to User Options Page

5.1.6 Manage Computers and Software (computeredit.php)


myDIRSIG Display User Computers - Mozilla Firefox
File Edit View Go Bookmarks Tools Help
http://dirsig.cis.rit.edu:8080/mydirsig/computeredit.php
Go
Getting Started
Latest Headlines
Done

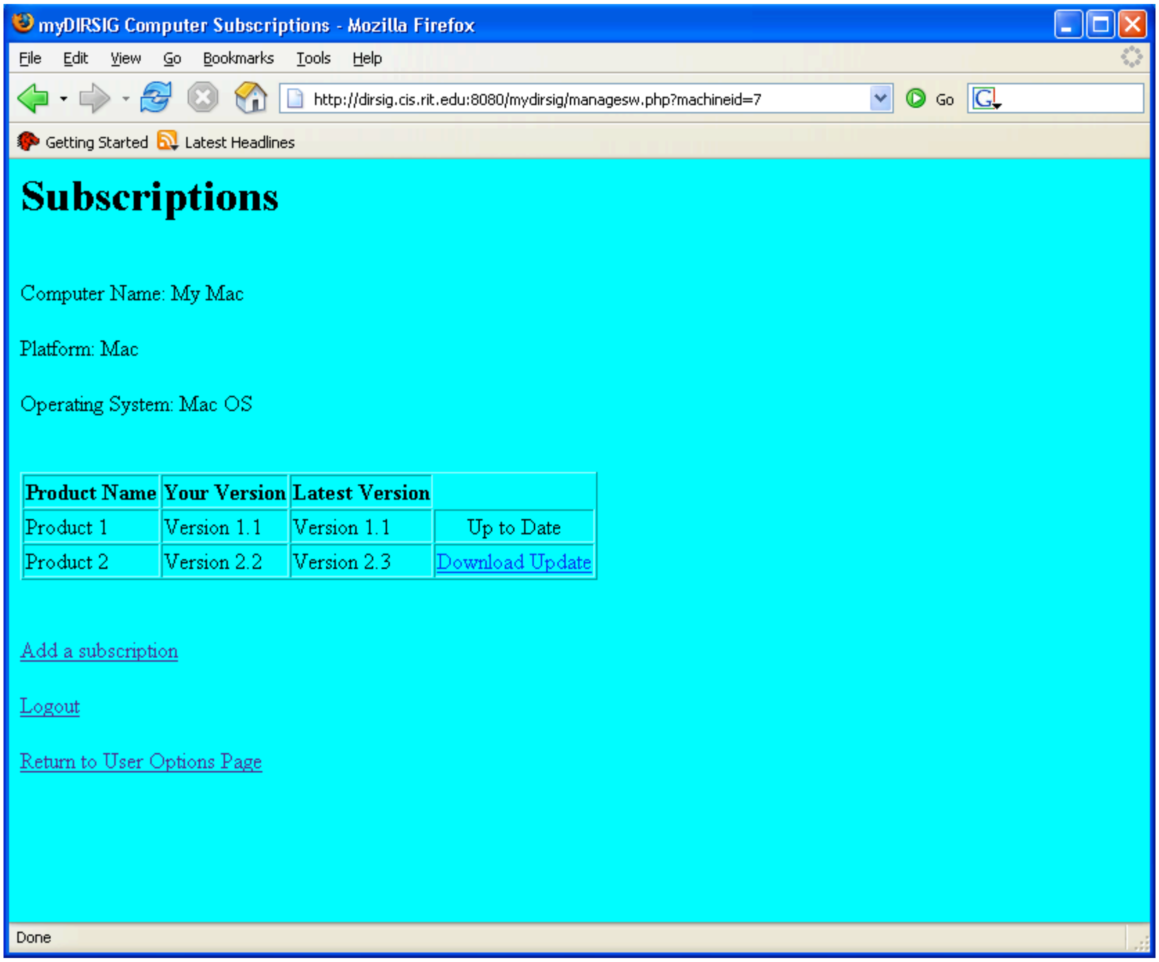
Your Computers

ID	Computer Name	Operating System	Platform			
7	My Mac	Mac OS	Mac	Edit	Remove	Subscriptions
8	My Pentium	Linux	Pentium	Edit	Remove	Subscriptions

Logout

Return to User Options Page

5.1.7 User Subscriptions (managesw.php)



myDIRSIG Computer Subscriptions - Mozilla Firefox
File Edit View Go Bookmarks Tools Help
http://dirsig.cis.rit.edu:8080/mydirsig/managesw.php?machineid=7
Go
Getting Started
Latest Headlines
Done

Subscriptions

Computer Name: My Mac

Platform: Mac

Operating System: Mac OS

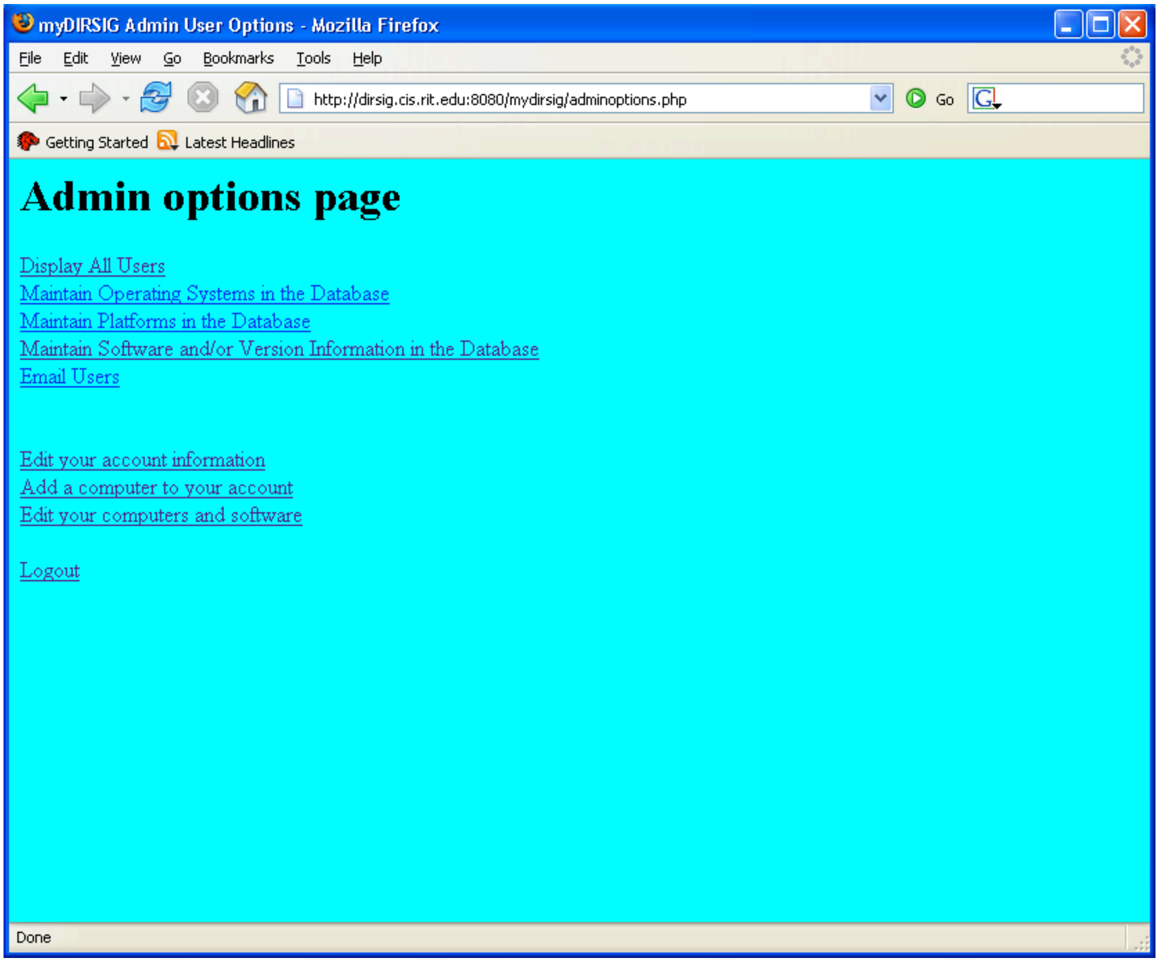
Product Name	Your Version	Latest Version	
Product 1	Version 1.1	Version 1.1	Up to Date
Product 2	Version 2.2	Version 2.3	Download Update

Add a subscription

Logout

Return to User Options Page

5.1.8 Admin Options Page (adminoptions.php)


myDIRSIG Admin User Options - Mozilla Firefox
File
Edit
View
Go
Bookmarks
Tools
Help
http://dirsig.cis.rit.edu:8080/mydirsig/adminoptions.php
Go
Getting Started
Latest Headlines
Done

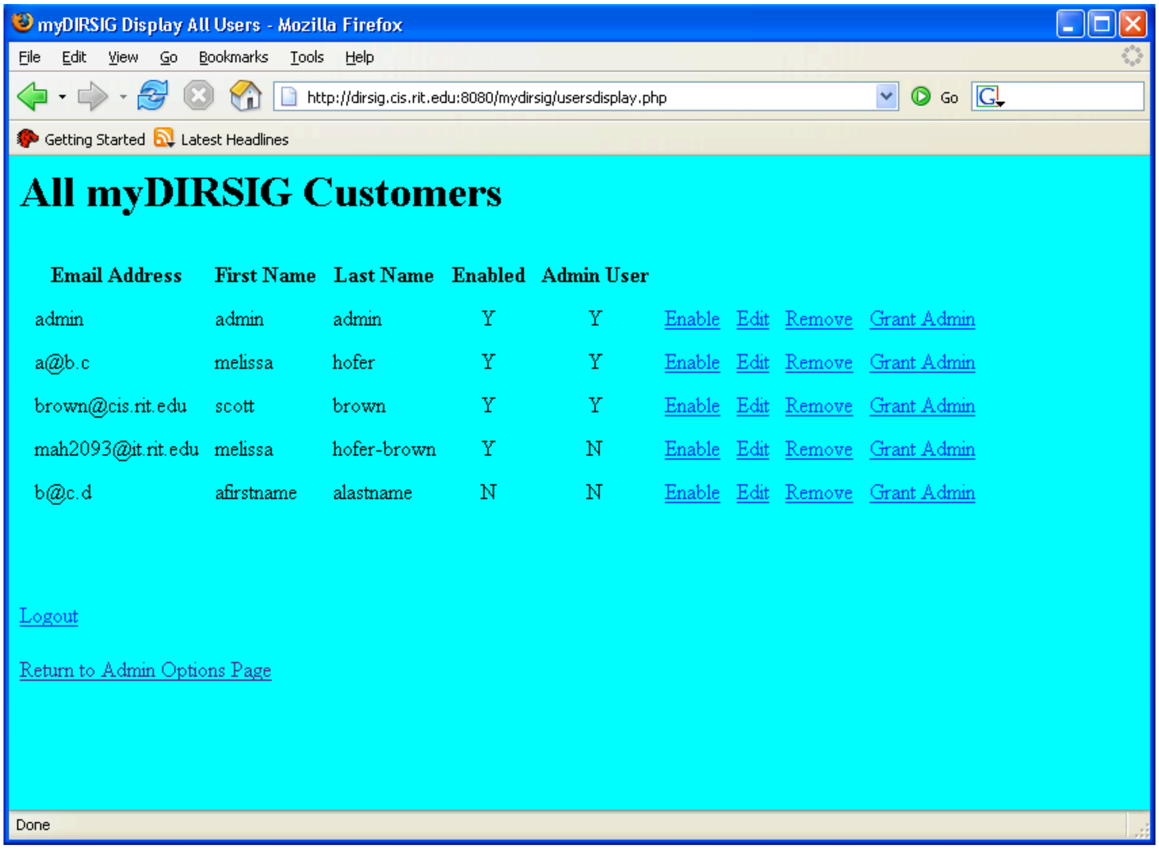
Admin options page

Display All Users
Maintain Operating Systems in the Database
Maintain Platforms in the Database
Maintain Software and/or Version Information in the Database
Email Users

Edit your account information
Add a computer to your account
Edit your computers and software

Logout

5.1.9 Display All Users (usersdisplay.php)



myDIRSIG Display All Users - Mozilla Firefox
File
Edit
View
Go
Bookmarks
Tools
Help
http://dirsig.cis.rit.edu:8080/mydirsig/usersdisplay.php
Go
Getting Started
Latest Headlines
Done

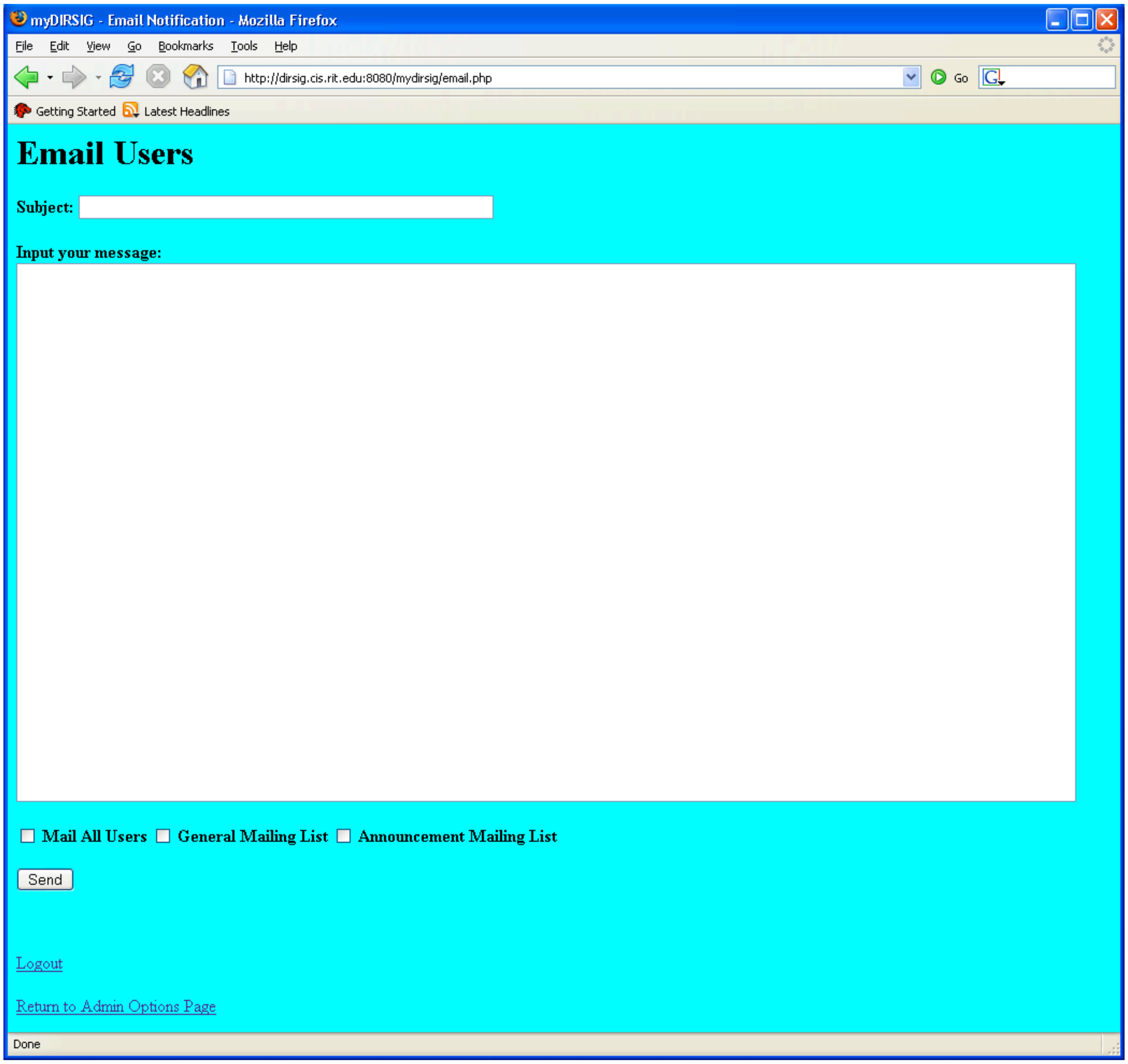
All myDIRSIG Customers

Email Address	First Name	Last Name	Enabled	Admin User				
admin	admin	admin	Y	Y	Enable	Edit	Remove	Grant Admin
a@b.c	melissa	hofer	Y	Y	Enable	Edit	Remove	Grant Admin
brown@cis.rit.edu	scott	brown	Y	Y	Enable	Edit	Remove	Grant Admin
mah2093@it.rit.edu	melissa	hofer-brown	Y	N	Enable	Edit	Remove	Grant Admin
b@c.d	afirstname	alastname	N	N	Enable	Edit	Remove	Grant Admin

Logout

Return to Admin Options Page

5.1.10 Email Users (email.php)



myDIRSIG - Email Notification - Mozilla Firefox
File Edit View Go Bookmarks Tools Help
http://dirsig.cis.rit.edu:8080/mydirsig/email.php
Go
Getting Started
Latest Headlines
Done

Email Users

Subject:

Input your message:

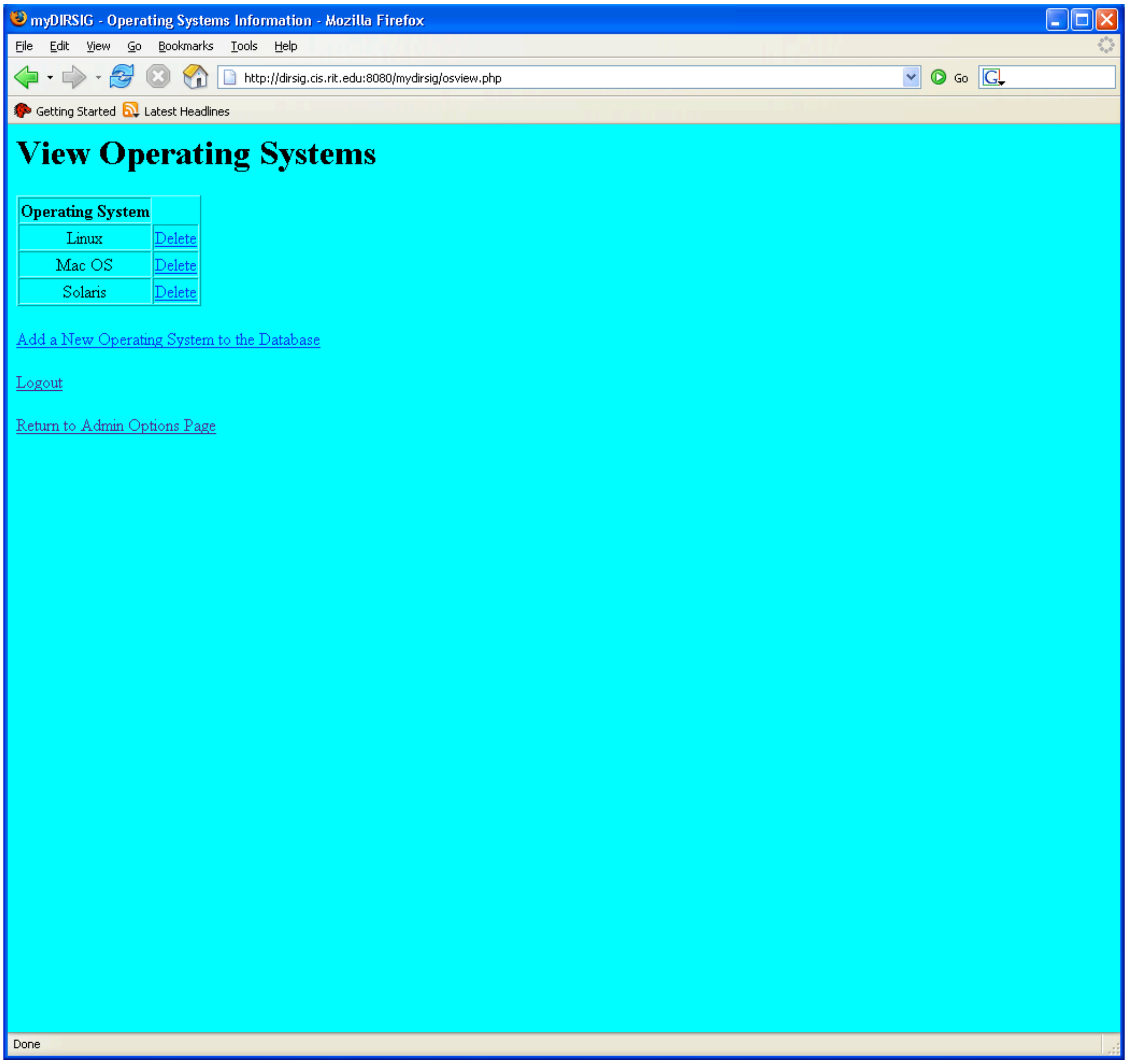
□ **Mail All Users** □ **General Mailing List** □ **Announcement Mailing List**

Send

Logout

Return to Admin Options Page

5.1.11 View Operating Systems (osview.php)


myDIRSIG - Operating Systems Information - Mozilla Firefox
File Edit View Go Bookmarks Tools Help
http://dirsig.cis.rit.edu:8080/mydirsig/osview.php
Go
Getting Started
Latest Headlines
Done

View Operating Systems

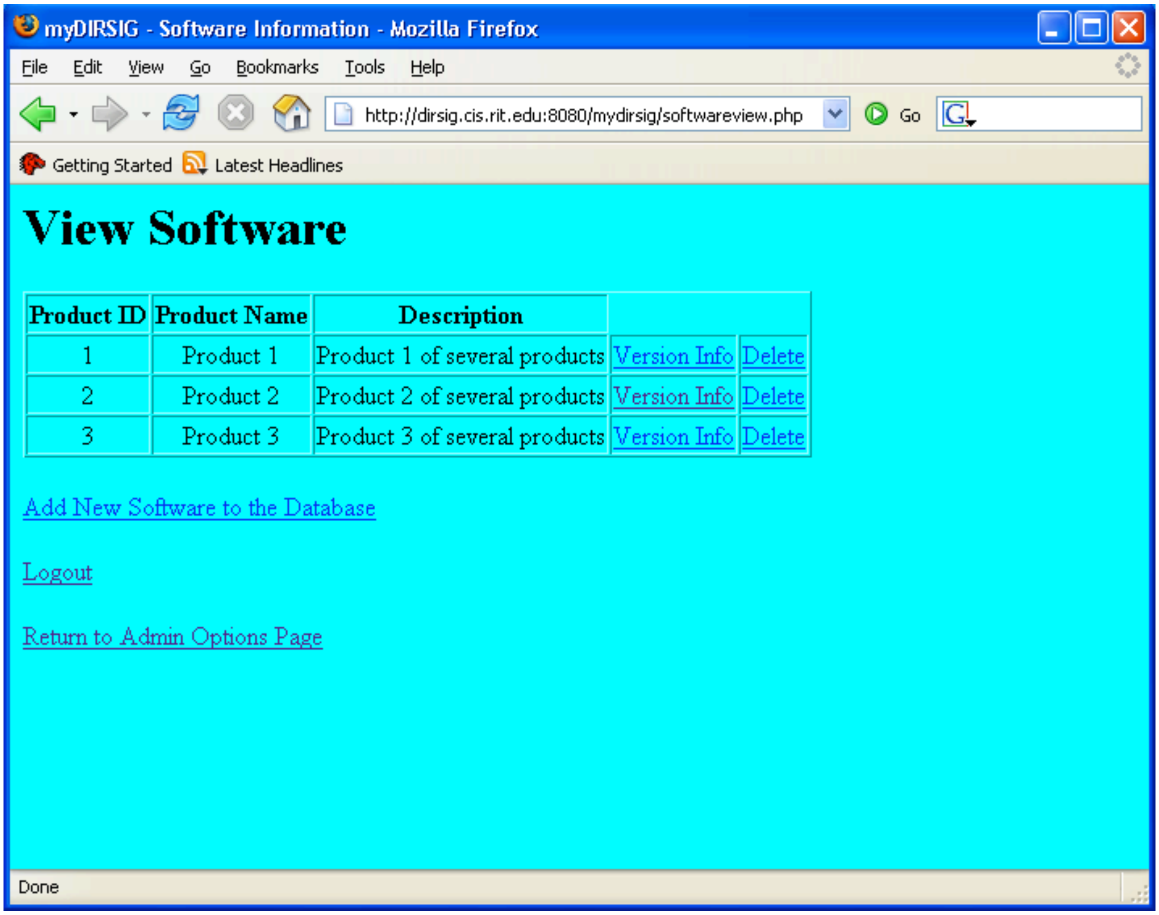
Operating System	
Linux	Delete
Mac OS	Delete
Solaris	Delete

Add a New Operating System to the Database

Logout

Return to Admin Options Page

5.1.12 View Software Products (softwareview.php)


myDIRSIG - Software Information - Mozilla Firefox
File Edit View Go Bookmarks Tools Help
http://dirsig.cis.rit.edu:8080/mydirsig/softwareview.php
Go
Getting Started
Latest Headlines
Done

View Software

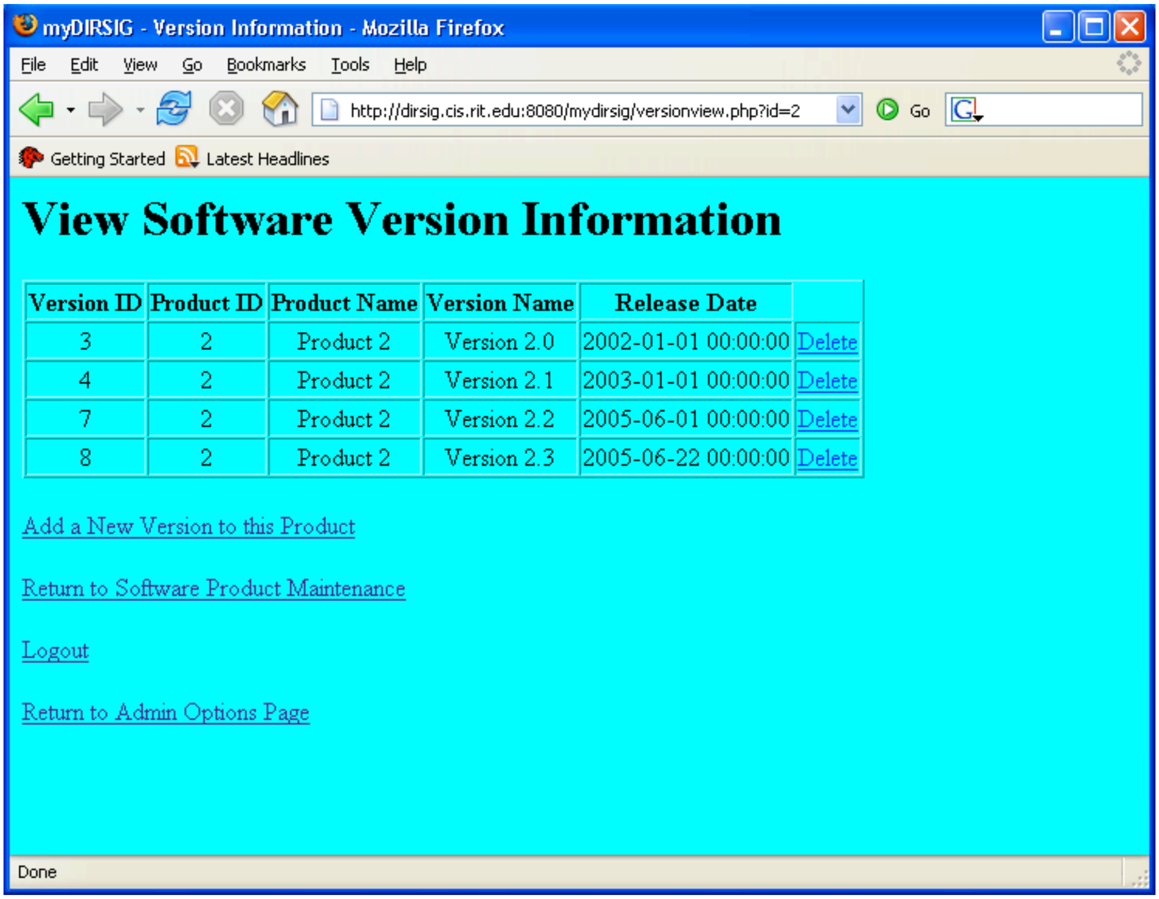
Product ID	Product Name	Description		
1	Product 1	Product 1 of several products	Version Info	Delete
2	Product 2	Product 2 of several products	Version Info	Delete
3	Product 3	Product 3 of several products	Version Info	Delete

Add New Software to the Database

Logout

Return to Admin Options Page

5.1.13 View Software Product Versions (versionview.php)


myDIRSIG - Version Information - Mozilla Firefox
File Edit View Go Bookmarks Tools Help
http://dirsig.cis.rit.edu:8080/mydirsig/versionview.php?id=2
Go
Getting Started
Latest Headlines
Done

View Software Version Information

Version ID	Product ID	Product Name	Version Name	Release Date	
3	2	Product 2	Version 2.0	2002-01-01 00:00:00	Delete
4	2	Product 2	Version 2.1	2003-01-01 00:00:00	Delete
7	2	Product 2	Version 2.2	2005-06-01 00:00:00	Delete
8	2	Product 2	Version 2.3	2005-06-22 00:00:00	Delete

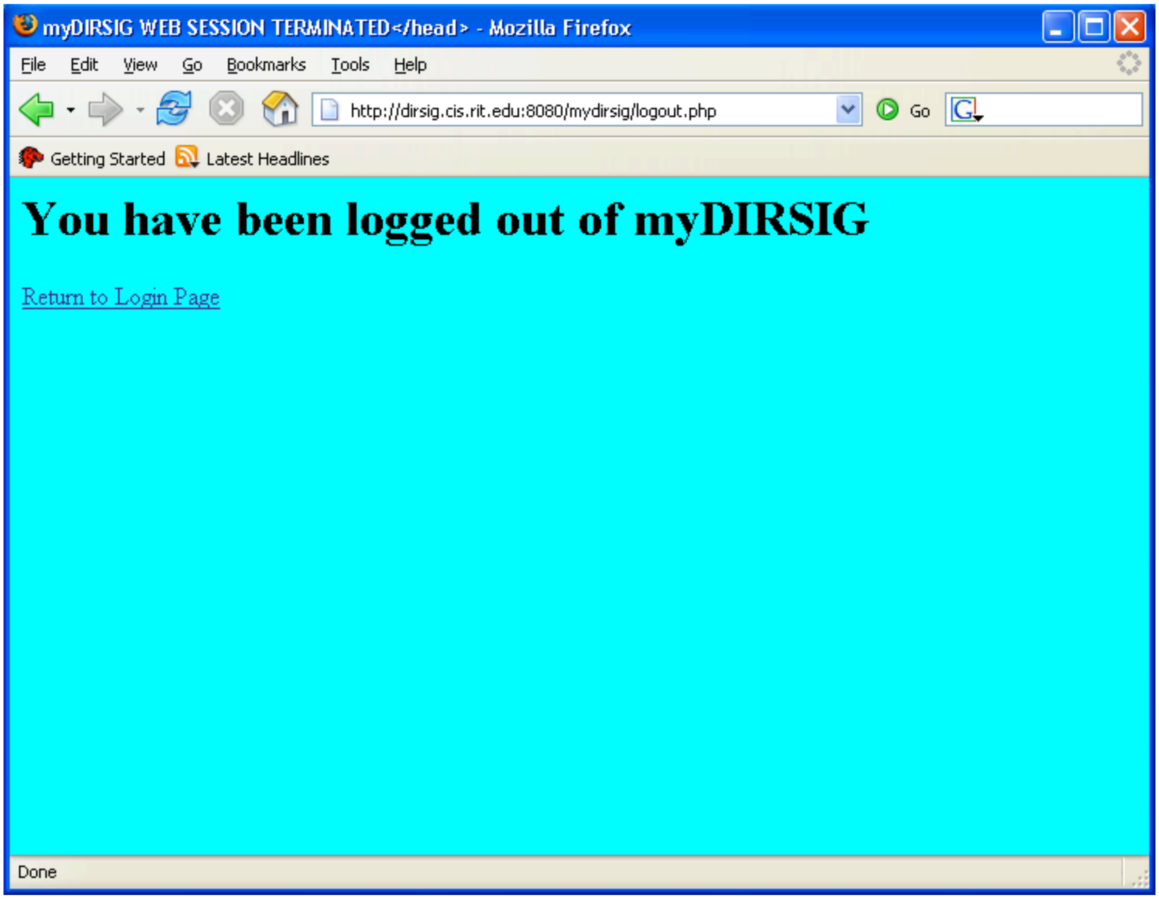
Add a New Version to this Product

Return to Software Product Maintenance

Logout

Return to Admin Options Page

5.1.14 Logout (logout.php)



myDIRSIG WEB SESSION TERMINATED</head> - Mozilla Firefox
File Edit View Go Bookmarks Tools Help
http://dirsig.cis.rit.edu:8080/mydirsig/logout.php
Go
Getting Started
Latest Headlines

You have been logged out of myDIRSIG

Return to Login Page

Done

5.2 Create Tables Script

```
CREATE TABLE customer(
      userid int(5) NOT NULL auto_increment,
      email varchar(40) NOT NULL,
      password varchar(40) NOT NULL,
      lastname varchar(20) NOT NULL,
      firstname varchar(20) NOT NULL,
      companyname varchar(50) NOT NULL,
      street varchar(50) NOT NULL,
      city varchar(20) NOT NULL,
      state varchar(2) NOT NULL,
      zip varchar(10) NOT NULL,
      phone varchar(12) NOT NULL,
      secure_phone varchar(12),
      enabled varchar(1) NOT NULL,
      announceMail varchar(1) NOT NULL,
      generalMail varchar(1) NOT NULL,
      maillist3 varchar(1) NOT NULL,
      maillist4 varchar(1) NOT NULL,
      adminUser varchar(1) NOT NULL,
      PRIMARY KEY(userid));

CREATE TABLE operating_system(
      opsys varchar(20) NOT NULL,
      PRIMARY KEY(opsys));

CREATE TABLE platform(
      platform varchar(20) NOT NULL,
      PRIMARY KEY(platform));

CREATE TABLE software(
      productid int(5) NOT NULL auto_increment,
      product_name varchar(40) NOT NULL,
      product_description varchar(80) NOT NULL,
      PRIMARY KEY(productid));

CREATE TABLE product_versions(
      versionid int(5) NOT NULL auto_increment,
      productid int(5) NOT NULL,
      version_name varchar(40) NOT NULL,
      release_date timestamp(8) NOT NULL,
      filepath varchar(60) NOT NULL,
      PRIMARY KEY(versionid),
      FOREIGN KEY(productid) REFERENCES software(productid));

CREATE TABLE computer(
      machineid int(5) NOT NULL auto_increment,
      userid int(5) NOT NULL,
      opsys varchar(20) NOT NULL,
      platform varchar(20) NOT NULL,
      computername varchar(40) NOT NULL,
      PRIMARY KEY(machineid),
      FOREIGN KEY(userid) REFERENCES customer(userid),
      FOREIGN KEY(opsys) REFERENCES operatingsystem(opsys),
      FOREIGN KEY(platform) REFERENCES platform(platform));
```

```
CREATE TABLE subscriptions(
      subscriptionid int(5) NOT NULL auto_increment,
      machineid int(5) NOT NULL,
      versionid int(5) NOT NULL,
      PRIMARY KEY(subscriptionid),
      FOREIGN KEY(machineid) REFERENCES computer(machineid),
      FOREIGN KEY(versionid) REFERENCES product_versions(versionid));

INSERT INTO customer VALUES(NULL, 'admin', 'admin', 'admin', 'admin',
'RIT', 'Center for Imaging Science', 'Rochester', 'NY', '14623', '585-
475-7194', 'n/a', 'Y', 'Y', 'Y', 'N', 'N', 'Y');
```