Florida International University

School of Computing and Information Sciences

Virtual Roll Call

Software Engineering Focus

Final Deliverable

**Team Members:**

Shalisha Witherspoon

Shonda Witherspoon

Ivana Rodriguez

**Product Owner:**

Jason Cohen

**Mentor(s):**

Masoud Sadjadi

Mohsen Taheri

**Instructor:**

Masoud Sadjadi

The MIT License (MIT)

Copyright (c) *2016 Florida International University*

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

***Abstract***

Police Officers use a roll call system as a means to relay important information at the beginning of each shift. Unfortunately, many officers are sometimes unable to attend roll due to various circumstances such as traffic, court obligations, and early assignments. Virtual Roll Call was a solution deemed fit by the Pinecrest Police Department in order to allow officers to view information provided at roll call at their earliest convenience, without missing any vital information.

Virtual Roll Call is a web application that allows Supervisors to upload documents to the system, and Officers to view/download these documents. The application is completely customizable by Administrators, who can add/edit/delete users from the system, add categories to upload documents under, and change logo and the department name, so that other police departments can use the system in the future.

By having the roll call system virtualized, officers are always up to date on the latest briefings, and can carry out their tasks efficiently, which will be a great service to the community as a whole.

**Table of Contents**

[Introduction](#_dajgq5qt1xha)

[Current System](#_5k0ozg86ihhm)

[Purpose of New System](#_5k0ozg86ihhm)

[User Stories](#_2et92p0)

[Implemented User Stories](#_1dkdkvh99qvl)

[Pending User Stories](#_3dy6vkm)

[Project Plan](#_1t3h5sf)

[Hardware and Software Resources](#_4d34og8)

[Sprint Plan](#_2s8eyo1)

[Sprint 1](#_17dp8vu)

[User Story 117: Learning, installation, and preparation](#_pjeb1mo29qhi)

[Sprint 2](#_ifx5diltoa53)

[User Story 121: Create User Login](#_xsi4c3swrdkr)

[Sprint 3](#_fdm7ndeqe77e)

[User Story 129: Add User](#_ksuycqwjsve9)

[User Story 141: Change Password](#_hm4qecmlzn0t)

[Sprint 4](#_oxkxvbsg3drk)

[User Story 168: Edit User](#_bz1f92ropv65)

[User Story 130: Remove User](#_e7q6pfu9fjca)

[User Story 140: Reset Password](#_hopcxvc9i51r)

[Sprint 5](#_gh9fcz886xck)

[User Story 132: Add Category](#_2h9svoxo9t0q)

[User Story 227: Add Document](#_fogles3rfkti)

[User Story 218: Edit Category](#_dc1zptqk5jwo)

[User Story 226: Remove Category](#_b2psxs27wukx)

[User Story 133: Upload Document](#_it2vdy1bp4el)

[Sprint 6](#_h39qdyqv56dh)

[User Story 131: Customize Site Settings](#_51bj07nfaqaq)

[User Story 229: List Pinned Documents](#_ysm9cquekhr4)

[User Story 230: List Unpinned Documents](#_9ioh5jelu1yv)

[User Story 228: View Categories](#_qxfy4as9yvy9)

[User Story 232: View Document](#_k8wbi5x23pag)

[Acceptance Criteria:](#_s6aphyq6oksz)

[System Design](#_qds3vr7os256)

[Architectural Patterns](#_26in1rg)

[System and Subsystem Decomposition](#_lnxbz9)

[Deployment Diagram](#_35nkun2)

[System Validation](#_44sinio)

[Glossary](#_2jxsxqh)

[Appendix](#_z337ya)

[Appendix A - UML Diagrams](#_3j2qqm3)

[Appendix B - User Interface Design](#_2xcytpi)

[Appendix C - Sprint Review Reports](#_wyi39pikxmq)

[Appendix D - User Manuals, Installation/Maintenance Document, Shortcomings/Wishlist Document and other documents](#_6w97znsp1i6t)

[References](#_2bn6wsx)

# 

# 

# Introduction

Traditionally, the first 15 minutes of a police officer’s shift is spent at the station in a briefing room where the shift supervisor, commonly of sergeant rank, goes over a clipboard of information with the officers. This information ranges from internal department matters, crime trends, be on the look outs (BOLOs), as well as any other information that needs to be passed on to the officers at the commencement of their daily tour of duty.

# Current System

The current roll call system is conducted with officers physically present at the police station. However, due to many factors, officers cannot always be present for roll call briefings; Officers may be in court, assigned to an overlapping shift, or may have to start their tour early due to call volume. The solution to help solve this problem was to invest in a virtualized version of the roll call system, where officers do not have to be physically present to receive the latest updates provided at roll call.

# Purpose of New System

The new roll call system, named Virtual Roll Call, is a web-based application where officers can login to view the documents and information that supervisors would go over at a traditional roll call. The purpose of this new system is to provide an additional avenue for supervisors to communicate and share important information with their officers. Virtual Roll Call will allow for officers to be briefed on all the necessary information, and give both the supervisor and officer the flexibility that is expected in the world we live in today.

**User Stories**

The following section provides the detailed user stories that were implemented in this iteration of the Virtual Roll Call project. These user stories served as the basis for the implementation of the project’s features. This section also shows the user stories that are to be considered for future development.

## Implemented User Stories

**User Story 117: Learning, installation, and preparation**

* This user story will be dedicated to learning the relevant technologies that will be used in our project, setting up our development environments, and preparing for the next sprint.

**User Story 121: Create User Login**

* As a user, I would like the ability to login of my role-based view of the application so that I can access the system.

**User Story 129: Add User**

* As an administrator, I would like to be able to add other users to the database, so that they will have authentication to login to the system.

**User Story 141: Change Password**

* As a user, I would like the ability to change my password so that I can protect the privacy of my credentials.

**User Story 168: Edit User**

* As an administrator, I would like the ability to edit user data so that I can update any changes for a given user.

**User Story 130: Remove User**

* As an administrator, I would like to be able to remove other users from the database, so that they may no longer access the system.

**User Story 140: Reset Password**

* As a supervisor, I would like the ability to reset passwords for locked users so that they can regain access to the system.

**User Story 132: Add Category**

* As an administrator, I would like the ability to add new categories to the officer view, so that documents are able to be sorted in a more appropriate category.

**User Story 227: Add Document**

* As a supervisor, I would like to add Document metadata to the database, so I can view detailed information about each document.

**User Story 218: Edit Category**

* As an administrator, I want the ability to edit the category names that documents will be uploaded into so that I can customize the application for my department.

**User Story 226: Remove Category**

* As an Administrator, I would like the ability to remove categories from the system, so that documents can no longer be associated with the chosen category.

**User Story 133: Upload Document**

* As a supervisor, I would like to be able to upload documents to a category folder, so that officers can view important material related to the category

**User Story 131: Customize Site Settings**

* As an administrator, I would like the ability to customize the department logo, application’s name, and the department’s name so that I can tailor the application to my police department’s needs.

**User Story 229: List Pinned Documents**

* As an Officer, I would like to see the list of pinned documents, so that I can know the most important documents that my supervisor would like to view

**User Story 230: List Unpinned Documents**

* As an officer, I would like to view the list of unpinned documents separately from pinned documents, so that I know the documents of least importance.

**User Story 228: View Categories**

* As an officer, I want to be able to view all of the document categories in one page so that I can drill down to a specific category.

**User Story 232: View Document**

* As an officer, I would like to view documents in a given category, so that I can be informed of important and relevant documents for each specific category

## 

## 

## Pending User Stories

**User Story 233: Add Document Confirmation**

* As a supervisor, I would like to add a document to the database, so that I can view the metadata about the document in the system

**User Story 231: Change Role View**

* As an administrator, I would like to change my role based view to Supervisor or Officer, so that I can check that customization additions and documents are being added properly

**User Story 147: Lock User**

* As an administrator, I want users to be locked out of the system after 5 unsuccessful login attempts so that I can strengthen the security of the system.

**User Story 241: Batch Add User**

* As an administrator, I would like to be able to add multiple users in one batch so that I don’t have to manually add each one at a time.

**User Story 242: Change Display Mode**

* As an officer, I would like to switch the display to either night or day mode so that I can view the site in lighter colors during the day and darker colors during the night.

**User Story 243: Add Document Quiz**

* As a supervisor, I want to be able to create quizzes that must be completed by the officer after viewing the document associated with the quiz so that I can ensure that the officer has understood the information in the document.

**User Story 244: Add User Logs**

* As a supervisor, I would like to view and export user activity logs that record whether a user has viewed a document and associated quiz attempt data (ex. score) so that I can hold users accountable for the information I share with them.

**User Story 245: Archive Documents**

* As an administrator, I want to be able to set the timeframe for which unpinned documents are automatically archived so that I can efficiently manage the storage on my department's server.

**User Story 246: Add Watch Orders**

* As a supervisor, I would like to add watch orders into the system so that they can later be viewed by officers in the app.

**User Story 247: View Watch Orders**

* As an officer, I would like to view all the watch orders relative to my current location on a map so that I can plan my visitation routes accordingly.

**User Story 248: Share Text Information**

* As a supervisor, I would like to be able to input updates into a free form text box so that I can share information with officers that doesn’t necessarily exist in a document form.

**User Story 249: View Document in App**

* As an officer, I would like to view documents within the app versus downloading it to my computer so that I can quickly traverse the information my supervisor is sharing with me.

**User Story 250: Add Password Encryption**

* As an administrator, I would like to add password encryption to user passwords so that I can improve the security of the site.

# 

# Project Plan

This section describes the planning that went into the realization of this project. This project incorporated the agile development techniques and as such required the sprints to be planned. These sprint plannings are detailed in the section. This section also describes the components, both software and hardware, chosen for this project.

## Hardware and Software Resources

The following is a list of all hardware and software resources that were used in this project:

* LAMP: Acronym for Linux, Apache, MySQL, and PHP, we used the following versions for the development of the VRC project:
  + Ubuntu 14.04.5 Linux as the OS of the development Virtual Machine
  + Apache2 as the web server
  + MySQL as the database for hosting system data and metadata
  + PHP 5.5.9 for backend development
* Angular JS framework for dynamic front-end web application development
* HTML5 and CSS3 for front-end web development
* BootStrap framework for front end web design
* Testing done on latest versions of FireFox, Internet Explorer, and Safari web browsers

## 

## Sprint Plan

### Sprint 1

#### User Story 117: Learning, installation, and preparation

*This user story will be dedicated to learning the relevant technologies that will be used in our project, setting up our development environments, and preparing for the next sprint.*

**Related Tasks:**

* Learn HTML5, CSS3, Javascript, AngularJS, PHP, and MySQL.
* Discuss production environment requirements with product owner and prepare identical development environments.
* Familiarize ourselves with SCRUM process, Mingle, Git/GitHub, and requirements for forthcoming sprints.

### Sprint 2

#### User Story 121: Create User Login

*As a user, I would like the ability to login of my role-based view of the application so that I can access the system.*  
   
**Acceptance Criteria:**

* user with credentials are granted access to the system
* user is redirected to their appropriate role-based view based on their role in the system
* unauthenticated users are unable to access the system
* users can exit the session by selecting logout; user is then redirected to login page

**Entrance Condition:**

* user is authenticated by their credentials and able to access the system

**Exit Condition:**

* user is on their appropriate role-based view

**Related Tasks:**

* Create a database
* Create login page
* Authenticate user
* Create user views (Administrator, Supervisor, Officer)
* Redirect user to appropriate view

### Sprint 3

#### User Story 129: Add User

*As an administrator, I would like to be able to add other users to the database, so that they will have authentication to login to the system.*

**Acceptance Criteria:**

* User is added to database
* User is able to login to system
* User is redirected to role-based page
* Password must satisfy the password requirements as specified in user story [#141](https://fiu-scis-seniorproject.mingle.thoughtworks.com/projects/virtual_roll_call/cards/141).

**Entrance condition:**

* User has credentials to login to system
* User role is administrator

**Exit condition:**

* User is successfully added to the database and provided login credentials

---

#### User Story 141: Change Password

*As a user, I would like the ability to change my password so that I can protect the privacy of my credentials.*

**Acceptance Criteria:**

* Password must be a minimum of 8 characters long and may include alphanumeric characters.
* If the user logs out after changing their password, the user will need to enter their new credentials to access the system.

**Entrance condition:**

* User has credentials to log into the system

**Exit condition:**

* User can log back in using the revise credentials

### Sprint 4

#### User Story 168: Edit User

*As an administrator, I would like the ability to edit user data so that I can update any changes for a given user.*

**Acceptance Criteria:**

* User data is modified in the database when edited in the UI
* The UI is updated to reflect changes

**Entrance condition:**

* User role is Administrator
* User to edit is selected from user table (via radio button)
* User data adheres to DB constraints

**Exit condition:**

* User is successfully added to the database and provided login credentials

---

#### User Story 130: Remove User

*As an administrator, I would like to be able to remove other users from the database, so that they may no longer access the system.*

**Acceptance Criteria:**

* User is removed from the database
* Removed user is no longer able to login to the system

**Entrance condition:**

* User role is administrator
* User to be removed has login credentials

**Exit condition:**

* User is no longer authenticated to login to the system

---

#### User Story 140: Reset Password

*As a supervisor, I would like the ability to reset passwords for locked users so that they can regain access to the system.*

**Acceptance Criteria:**

* The supervisor can reset the password for all users.
* If the supervisor resets the officer’s password, the officer regains access to the system.
* Password must satisfy the password requirements as specified in user story #141.

**Entrance condition:**

* User is a supervisor
* The officer that has been locked out must exist in the system

**Exit condition:**

* The officer’s password is reset

### Sprint 5

#### User Story 132: Add Category

*As an administrator, I would like the ability to add new categories to the officer view, so that documents are able to be sorted in a more appropriate category.*

**Acceptance Criteria:**

* New category is available for selection in the supervisor and officer view
* Documents are able to be uploaded to new category

**Entrance condition:**

* User is administrator and logged into the system

**Exit condition:**

* New category is added to the list of existing categories

---

#### User Story 227: Add Document

*As a supervisor, I would like to add Document metadata to the database, so I can view detailed information about each document.*

**Acceptance Criteria:**

* Document metadata is added to the Database
* Uploaded Document metadata is displayed in supervisor view Document table

**Entrance Condition:**

* User role is Supervisor
* Supervisor enters Document metadata in the form

**Exit Condition**

* Document metadata is added to the system

**---**

#### User Story 218: Edit Category

*As an administrator, I want the ability to edit the category names that documents will be uploaded into so that I can customize the application for my department.*

**Acceptance Criteria:**

* If I’m an admin, I should have access to view and edit categories in the database.
* When I change the name of a category, the changes are automatically reflected on the site.

**Entrance condition:**

* User is administrator and logged in

**Exit condition:**

* Existing category edits are saved to the database

---

#### User Story 226: Remove Category

*As an Administrator, I would like the ability to remove categories from the system, so that documents can no longer be associated with the chosen category.*

**Acceptance Criteria:**

* Category listing is removed from admin view
* Category is removed from Database

**Entrance condition:**

* User role is Supervisor
* Supervisor navigates to Site Settings view

**Exit condition:**

* Category is removed from system

**---**

#### User Story 133: Upload Document

*As a supervisor, I would like to be able to upload documents to a category folder, so that officers can view important material related to the category*

**Acceptance Criteria:**

* Document is added to category folder
* Document is able to be viewed by officer

**Entrance condition:**

* User role is supervisor or administrator
* User has document to upload

**Exit condition:**

* Document is added to category folder, and officer is able to view uploaded document

### Sprint 6

#### User Story 131: Customize Site Settings

*As an administrator, I would like the ability to customize the department logo, application’s name, and the department’s name so that I can tailor the application to my police department’s needs.*

**Acceptance Criteria:**

* Department logo image is changed to admin’s selected image
* Application name is changed accordingly and reflected on the rest of the site
* Department name is changed accordingly and reflected on the rest of the site
* All site setting changes persist after refreshing the page

**Entrance condition:**

* User is administrator, is logged in, and on the site settings page

**Exit condition:**

* Any changes an administrator makes in terms of site settings is reflected on the rest of the site

---

#### User Story 229: List Pinned Documents

*As an Officer, I would like to see the list of pinned documents, so that I can know the most important documents that my supervisor would like to view*

**Acceptance Criteria:**

* Officer is able to see a list of documents on a separate panel from unpinned documents

**Entrance condition:**

* User is logged in as officer role
* User selects Category to view documents under

**Exit condition:**

* User is able to view list of pinned documents

---

#### User Story 230: List Unpinned Documents

*As an officer, I would like to view the list of unpinned documents separately from pinned documents, so that I know the documents of least importance.*

**Acceptance Criteria:**

* Officer is able to view list of unpinned documents separately from pinned documents

**Entrance condition:**

* User is logged on to officer role
* Officer navigates to chosen category to view documents under

**Exit condition:**

* Officer is able to view list list of unpinned documents

---

#### User Story 228: View Categories

*As an officer, I want to be able to view all of the document categories in one page so that I can drill down to a specific category.*

**Acceptance Criteria:**

* The officer can view all the categories in the system
* When the officer clicks on a category, they can view all the documents that have been uploaded in that category

**Entrance condition:**

* User is an officer and has credentials to access the system

**Exit condition:**

* Officer is able to view all the document categories in the system

---

#### User Story 232: View Document

*As an officer, I would like to view documents in a given category, so that I can be informed of important and relevant documents for each specific category*

### Acceptance Criteria:

* Document metadata is added to the Database
* Document exists in the uploads directory
* Link to document exists in documents pane

**Entrance Condition:**

* User role is Officer
* Officer clicks on a category that has reference to documents

**Exit Condition:**

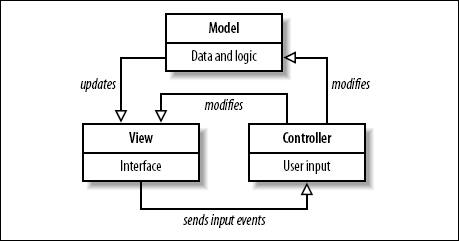
* Officer is able to view or download the file after clicking file icon

### System Design

This section contains information on the design decisions that went into this project. The architecture patterns are outlined and explained. The entire system is shown in a package diagram and the subsystems are explained.

## Architectural Patterns

**Model-View-Controller**



Model-View-Controller is an architectural pattern commonly used for designing web applications. As the name implies, there are 3 major components in the architecture:

1. Model – manages the data and logic of the application

2. View – The User Interface of the application

3. Controller – Receives user input and modifies data in the model

As the image shows, these components are separate, but interact to maintain the application state, and displays updates to the view. Our system uses Angular JS, which is a framework based on the Model-View-Controller architecture to achieve its functionality, and allows the application to be implemented more efficiently.

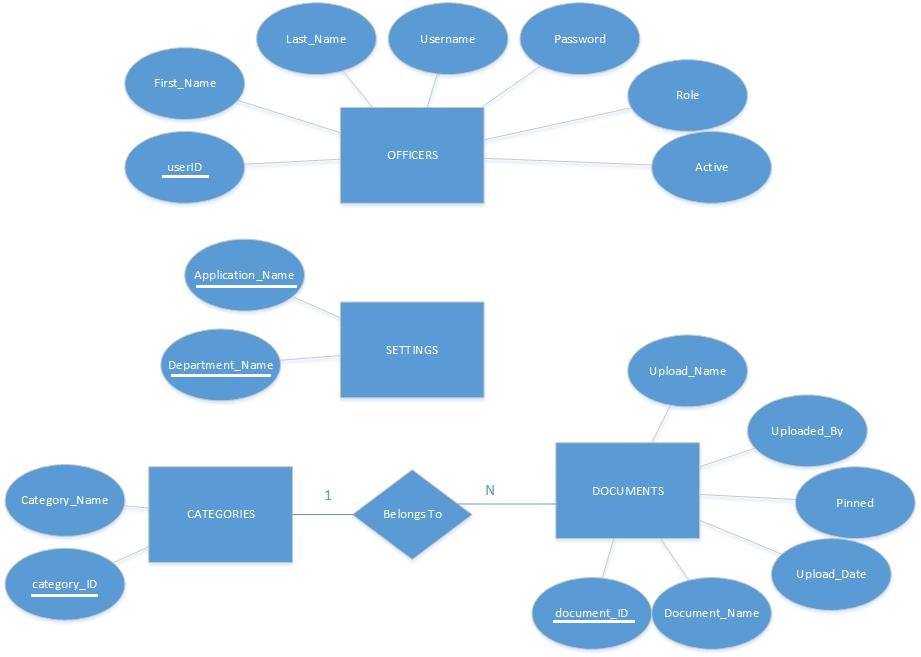
--

**Client-Server**



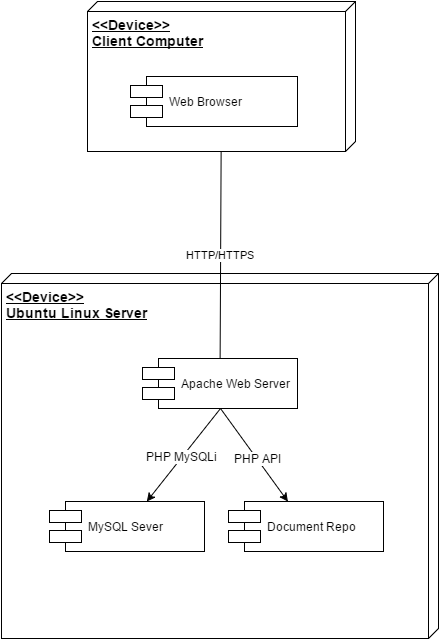
Our system uses the Client-Server model, which is an application structure used in distributed applications. As the image shows, the client and server communicate via the internet. The client makes requests to the server, and the server responds to the client with the requested function. This allows the client to not have to rely on their own computer’s resources, and to conveniently access applications and resources hosted on the server from any machine. Virtual Roll Call is hosted on a Linux Ubuntu virtual machine, and clients are able to use the application and perform actions through the server.

**ER Model**



## 

## Deployment Diagram



# System Validation

This section details the unit test cases that were used to validate the features completed during each sprint. Each use case unit test consist of a sunny day and rainy day scenario, as well as an integration test to ensure the functionality of previous implementations remains intact.

...

**User Story #1: Login**

**Sunny Day:**

|  |  |
| --- | --- |
| **Purpose** | **Verify that user with valid credentials can access the system** |
| **Entrance Condition** | **1. User role is one of three roles (officer, supervisor, or administrator)**  **2. User enters username and password in the form** |
| **Input** | **Username = test001 password = test1234** |
| **Expected outcome** | **Credentials are accepted, and user is routed to Officer role view** |

**Rainy Day:**

|  |  |
| --- | --- |
| **Purpose** | **Verify that user with invalid credentials cannot access the system** |
| **Entrance Condition** | **1. User role is one of three roles (officer, supervisor, or administrator)**  **2. User enters username and password in the form** |
| **Input** | **Username = test002 password = test1234** |
| **Expected output** | **Error Message: “Invalid credentials. Please try again.”** |

**Integration Test**

|  |  |
| --- | --- |
| **Purpose** | **Verify that user with changed password cannot access system with previous password** |
| **Entrance Condition** | **1. User role is one of three roles (officer, supervisor, or administrator)**  **2. User enters username and password in the form** |
| **Input** | **Username = test001 password = user’s old password** |
| **Expected output** | **Error Message: “Invalid credentials. Please try again.”** |

**User Story #129: Add User**

**Sunny Day:**

|  |  |
| --- | --- |
| Purpose | Verify user is successfully added to system |
| Entrance Condition | 1. User is logged in as Administrator |
| Input | User to be added’s first name, last name, username, password, and role in the system |
| Expected Output | Success message informing user was successfully added to system; updated row in edit user table |

**Rainy Day**

|  |  |
| --- | --- |
| Purpose | Verify user with username already in system cannot be added |
| Entrance Condition | 1. User with username about to added to system already exist in the database |
| Input | Username of user already added to the system |
| Expected Output | Error message informing user they cannot add the “new” user to the database |

**Integration Test**

|  |  |
| --- | --- |
| Purpose | Verify newly added user can now login to system |
| Entrance Condition | 1. Administrator provided credentials to user attempting to login |
| Input | Username and password |
| Expected Output | User is able to login to system and redirected to user view based on role |

**User Story #141: Change Password**

**Sunny Day**

|  |  |
| --- | --- |
| **Test Case ID** | Change\_Password\_UT\_01 (Sunny Day) |
| **Purpose** | Verify that when the input fields are valid, the system updates the user’s password permanently |
| **Preconditions** | User is logged in and on the change password page |
| **Input** | 1. The user enters their current password 2. The user enters a new password 3. The user repeats their new password in the confirmation field 4. The user selects the change password button |
| **Expected Output** | If the input fields are valid (see UC definition), change the user’s password |

**Integration Test**

|  |  |
| --- | --- |
| **Test Case ID** | Change\_Password\_IT\_01 (Sunny Day) |
| **Purpose** | Verify that if the user changes their password, they are able to log into the site again with their new password and not their old password |
| **Preconditions** | User is logged in and on the change password page |
| **Input** | 1. The user enters their current password 2. The user enters a new password 3. The user repeats their new password in the confirmation field 4. The user selects the change password button 5. The user selects logout in the menu 6. The user logs into the system again using their new password |
| **Expected Output** | The user is able to access the system |

**Rainy Day**

|  |  |
| --- | --- |
| **Test Case ID** | Change\_Password\_IT\_02 (Rainy Day) |
| **Purpose** | Verify that if the user changes their password, they can only log in with the new password and not the old one. |
| **Preconditions** | User is logged in and on the change password page |
| **Input** | 1. The user enters their current password 2. The user enters a new password 3. The user repeats their new password in the confirmation field 4. The user selects the change password button 5. The user selects logout in the menu 6. The user tries to log into the system again using their old password |
| **Expected Output** | The user is not able to access the system |

**User Story #130: Remove User**

**Sunny Day:**

|  |  |
| --- | --- |
| Purpose | Verify that user is removed from the data base and can longer access the system |
| Entrance Condition | 1. User is logged in as Admin role  2. User to be removed has credentials to access the system  3. Admin selects user they wish to delete in the Edit User panel |
| Input | The user’s ID in the system |
| Expected Output | Success Message. Selected user’s credentials and profile information are removed from the Database and Edit User panel table |

**Rainy Day**

|  |  |
| --- | --- |
| Purpose | Verify deleted user can no longer access the system |
| Entrance Condition | 1. User previously able to access the system  2. User deleted from system by Admin role  3. User attempts to login with previous credentials |
| Input | User enters previously authenticated username and password into login form |
| Expected Output | Error message informing user their credentials are not found in the system |

**Integration Test**

|  |  |
| --- | --- |
| Purpose | Verify that system is deleting correct user from system |
| Entrance Condition | 1. User is logged in as Admin role  2. User to be removed has credentials to access the system  3. Admin selects use they wish to delete in the Edit User panel |
| Input | 1. Admin clicks Delete User Button on Edit User Modal  2. ID from selected user is passed from the table by the controller to the DB  3. DB removes the user with the inputted ID from the Database, removing them from the system |
| Expected Output | DB returns “Deleted” JSON message and a confirmation is displayed to the user on the Edit User Panel |

**User Story #140: Reset Password**

**Sunny Day**

|  |  |
| --- | --- |
| **Test Case ID** | Reset\_Password\_UT\_01 (Sunny Day) |
| **Purpose** | Verify that a supervisor resets the password for the selected user |
| **Preconditions** | Supervisor is logged in and on the reset password page |
| **Input** | 1. The supervisor selects a user by marking the associated the checkbox 2. The supervisor enters a new password and password confirmation for the selected user that match and meet the minimum password requirements 3. The supervisor clicks on the reset button |
| **Expected Output** | The selected user’s password is reset |

**Rainy Day**

|  |  |
| --- | --- |
| **Test Case ID** | Reset\_Password\_UT\_02 (Rainy Day) |
| **Purpose** | Verify that if a supervisor selects a user, the password reset functions for other users is disabled |
| **Preconditions** | Supervisor is logged in and on the reset password page |
| **Input** | 1. The supervisor selects a user by marking the associated the checkbox 2. The supervisor clicks on the wrong reset password fields (for a different user in the table |
| **Expected Output** | Fields and functions for other users is disabled; user can’t enter a value in fields other than the selected user |

**Integration Test**

|  |  |
| --- | --- |
| **Test Case ID** | Reset\_Password\_IT\_01 (Rainy Day) |
| **Purpose** | Verify that if a supervisor changes a password for a user, they can only log in with the new password and not the old one. |
| **Preconditions** | Supervisor is logged in and on the reset password page |
| **Input** | 1. The supervisor selects a user by marking the associated the checkbox 2. The supervisor enters a new password and password confirmation for the selected user that match and meet the minimum password requirements 3. The supervisor clicks on the reset button 4. The user tries to log into the system again using their old password |
| **Expected Output** | The user is not able to access the system |

**User Story #168: Edit user**

**Sunny Day**

|  |  |
| --- | --- |
| Purpose | Verify that user’s attributes are edited in the database, and reflects the data the Administrator inputted in the edit modal |
| Entrance Condition | 1. User role is Administrator  2. User to edit is selected from user table (via button)  3. User data adheres to DB constraints |
| Input | The selected user’s first name, last name, username, and role in the system |
| Expected Output | Success Message. Selected user’s attributes edited in the Database and changes are reflected in UI (Edit User panel table) |

**Rainy Day**

|  |  |
| --- | --- |
| Purpose | Verify that Administrator cannot successfully data that violates DB constraints (such as duplicate username) and receive a success message |
| Entrance Condition | 1. User role is Administrator  2. User to edit is selected from user table (via button)  3. User data adheres to DB constraints |
| Input | User enters a username already in the database |
| Expected Output | Edit is denied, and user’s attributes do not update |

**Integration Test**

|  |  |
| --- | --- |
| Purpose | Verify that edited fields (such as first and/or last name) are reflected in user log in |
| Entrance Condition | User role is Administrator, Supervisor, or Officer |
| Expected Outcome | User login has updated first and/or last name |

**User Story #132: Add Category**

**Sunny Day**

|  |  |
| --- | --- |
| **Test Case ID** | Add\_Category\_UT\_01 (Sunny Day) |
| **Purpose** | Verify that a valid category name is added to the database |
| **Preconditions** | Administrator is logged in and on the document categories page |
| **Input** | 1. The admin enters a name in the category field 2. The admin clicks the ‘Add’ button |
| **Expected Output** | The category name is added to the database and the admin is notified accordingly |

**Integration Test**

|  |  |
| --- | --- |
| **Test Case ID** | Add\_Category\_IT\_01 (Sunny Day) |
| **Purpose** | Verify that if an admin adds a new category to the database, a supervisor can later view and select that category in their view |
| **Preconditions** | Administrator is logged in and on the manage categories page |
| **Input** | 1. The admin enters a name in the category field 2. The admin clicks the ‘Add’ button 3. A supervisor logs in and navigates to the manage documents page |
| **Expected Output** | The category name is now available for selection in the categories drop down menu |

**Rainy Day**

|  |  |
| --- | --- |
| **Test Case ID** | Add\_Category\_UT\_02 (Rainy Day) |
| **Purpose** | Verify that if an admin enters an invalid name, the category isn’t added to the database |
| **Preconditions** | Administrator is logged in and on the document categories page |
| **Input** | 1. The admin enters a name that already exists in the category field 2. The admin clicks the ‘Add’ button |
| **Expected Output** | The category name is not added to the database and the admin is notified accordingly |

**Rainy Day**

|  |  |
| --- | --- |
| **Test Case ID** | Add\_Category\_UT\_03 (Rainy Day) |
| **Purpose** | Verify that if an admin enters no name, the category isn’t added to the database |
| **Preconditions** | Administrator is logged in and on the document categories page |
| **Input** | 1. The admin doesn’t enter anything in the category field 2. The admin clicks the ‘Add’ button |
| **Expected Output** | The category name is not added to the database and the admin is notified accordingly |

**User Story #218: Edit Category**

**Sunny Day**

|  |  |
| --- | --- |
| **Test Case ID** | Edit\_Category\_UT\_01 (Sunny Day) |
| **Purpose** | Verify that category name is updated on the database |
| **Preconditions** | Administrator is logged in and on the document categories page |
| **Input** | 1. The admin selects the ‘Edit’ button for the category he/she would like to edit 2. The admin enters a valid name in the modal category field and hits ‘Update’ |
| **Expected Output** | The category name is updated to the database and the admin is notified accordingly |

**Integration Test**

|  |  |
| --- | --- |
| **Test Case ID** | Edit\_Category\_IT\_01 (Sunny Day) |
| **Purpose** | Verify that if an admin edits a category to the database, a supervisor can later view the edited category on their view |
| **Preconditions** | Administrator is logged in and on the manage categories page |
| **Input** | 1. The admin enters a name in the category field 2. The admin clicks the ‘Add’ button 3. A supervisor logs in and navigates to the manage documents page |
| **Expected Output** | The modified category name is now available for selection in the categories drop down menu |

**Rainy Day**

|  |  |
| --- | --- |
| **Test Case ID** | Edit\_Category\_UT\_02 (Rainy Day) |
| **Purpose** | Verify that if an admin enters an invalid name, the category isn’t updated on the database |
| **Preconditions** | Administrator is logged in and on the document categories page |
| **Input** | 1. The admin selects the ‘Edit’ button for the category he/she would like to edit 2. The admin enters a name that already exists in the category field 3. The admin clicks the ‘Update’ button |
| **Expected Output** | The category name is not updated on the database and the admin is notified accordingly |

**Rainy Day**

|  |  |
| --- | --- |
| **Test Case ID** | Edit\_Category\_UT\_03 (Rainy Day) |
| **Purpose** | Verify that if an admin enters no name, the category isn’t added to the database |
| **Preconditions** | Administrator is logged in and on the document categories page |
| **Input** | 1. The admin selects the ‘Edit’ button for the category he/she would like to edit 2. The admin doesn’t enter anything in the category field 3. The admin clicks the ‘Update’ button |
| **Expected Output** | The category name is not updated on the database and the admin is notified accordingly |

**User Story #226: Remove Category**

**Sunny Day**

|  |  |
| --- | --- |
| Purpose | Verify that category is removed from the database |
| Entrance Condition | 1. User is logged in as Admin role  2. Category to be deleted is inserted in Database  3. Admin selects category they wish to delete in the Edit User panel |
| Input | The category’s ID in the system |
| Expected Output | Success Message. Selected category is removed from the Database and Edit Category panel table |

**Rainy Day**

|  |  |
| --- | --- |
| Purpose | Verify Documents associated with removed category is removed from System |
| Entrance Condition | 1. Category previously entered in DB  2. Document added to DB and associated with category previously entered in DB  3. Admin deletes the selected category from the system  4. Supervisor logins to system  5. Supervisor navigates to Manage Documents view |
| Input | none |
| Expected Output | Edit documents table with all documents associated with removed category removed from table |

**Integration Test**

|  |  |
| --- | --- |
| Purpose | Verify that system is deleting correct category and documents from system |
| Entrance Condition | 1. User is logged in as Admin role  2. Category to be deleted is inserted in DB  3. Admin selects category they wish to delete in the Edit Category panel |
| Input | 1. Admin clicks Delete Category Button on Edit Category Modal  2. ID from selected category is passed from the table by the controller to the DB  3. DB removes the category with the inputted ID from the Database, removing them from the system |
| Expected Output | DB returns “Deleted” JSON message and a confirmation is displayed to the user on the Edit Category Panel |

# User Story #133: Upload Document

**Sunny Day:**

|  |  |
| --- | --- |
| Purpose | Verify that document is uploaded to the server |
| Entrance Condition | 1. User role is Supervisor 2. Supervisor has document to load to server |
| Input | Document from Supervisor’s computer files, name of document, category, and whether document should be pinned, |
| Expected Output | Document added to uploads directory in server; input data displayed on UI |

**Rainy Day**

|  |  |
| --- | --- |
| Purpose | Verify that Supervisor cannot successfully add document that violates DB constraints (such as duplicate document name) |
| Entrance Condition | 1. User role is Supervisor 2. Supervisor has document to load to server |
| Input | User enters a document already in the database |
| Expected Output | Document cannot be added |

**Integration Test**

|  |  |
| --- | --- |
| Purpose | Verify that document metadata appears in the officer view |
| Entrance Condition | User role is Officer |
| Expected Outcome | Uploaded document appears in the appropriate category |

# User Story #227: Add Document

**Sunny Day**

|  |  |
| --- | --- |
| Purpose | Verify that document metadata is added to DB |
| Entrance Condition | 1. User is logged in as Supervisor role  2. User selects file they wish to upload to system |
| Input | The document name, category, pinned status (date and uploaded by is added from session) |
| Expected Output | Document metadata displayed in Edit Documents panel table |

**Rainy Day**

|  |  |
| --- | --- |
| Purpose | Verify documents cannot be added with the same name |
| Entrance Condition | 1. Supervisor uploaded previous document and metadata  2. Supervisor chooses new file to upload |
| Input | User enters previously added filename to new uploaded file metadata |
| Expected Output | New file metadata is not added to Edit Documents panel table |

**Integration Test**

|  |  |
| --- | --- |
| Purpose | Verify that correct metadata is being added to DB |
| Entrance Condition | 1. User is logged in as Supervisor role  2. User navigates to Manage Documents view |
| Input | 1. Supervisor browses for document to upload  2. Supervisor inputs file name, category, and whether document is pinned in Upload Documents form  3. DB adds the inputted document metadata to the form, along with the logged in Supervisor’s name and date from the current session |
| Expected Output | Metadata added from form is displayed in Edit Documents panel table to be viewed |

# 

# User Story #131: Customize Site Settings

**Sunny Day**

|  |  |
| --- | --- |
| **Test Case ID** | Site\_Settings\_UT\_01 (Sunny Day) |
| **Purpose** | Verify that a valid logo can be uploaded to the site |
| **Preconditions** | Administrator is logged in and on the site settings page |
| **Input** | 1. The admin clicks on the ‘Select Image’ button 2. The admin selects a valid image (either PNG, JPG, or GIF) from their file system 3. The admin previews the image they just selected 4. The admin clicks on ‘Save’ to replace the logo on the site |
| **Expected Output** | The logo image file is replaced with the new uploaded image |

**Rainy Day**

|  |  |
| --- | --- |
| **Test Case ID** | Site\_Settings\_UT\_02 (Rainy Day) |
| **Purpose** | Verify that invalid file types are not uploaded to the site |
| **Preconditions** | Administrator is logged in and on the site settings page |
| **Input** | 1. The admin clicks on the ‘Select Image’ button 2. The admin then selects an invalid image (NOT PNG, JPG, or GIF) from their file system |
| **Expected Output** | Either the admin is not able to select a non-image type when prompted, or if they select a file with an unknown file extension (ex. .ico), they’re not able to move forward in uploading an image |

**Integration Test**

|  |  |
| --- | --- |
| **Test Case ID** | Site\_Settings\_IT\_01 (Sunny Day) |
| **Purpose** | Verify that if an admin uploads a new logo, the login page is updated accordingly |
| **Preconditions** | Administrator is logged in and on the site settings page |
| **Input** | 1. The admin clicks on the ‘Select Image’ button 2. The admin selects a valid image (either PNG, JPG, or GIF) from their file system 3. The admin previews the image they just selected 4. The admin clicks on ‘Save’ to replace the logo on the site |
| **Expected Output** | The logo on the login site is replaced with the newly uploaded image |

# User Story #228: View Categories

**Sunny Day**

|  |  |
| --- | --- |
| **Test Case ID** | View\_Categories\_UT\_01 (Sunny Day) |
| **Purpose** | Verify that all categories can be viewed and selected when an officer logs in |
| **Preconditions** | The user is an officer |
| **Input** | 1. The officer logs in and views all the categories in the system 2. The officer selects a category name and can the view any documents that have been added/uploaded into that category |
| **Expected Output** | The officer is able to view all the categories and is routed according to selected category |

# User Story #232: View Documents

**Sunny Day**

|  |  |
| --- | --- |
| Purpose | Verify that open or download document in the appropriate folder |
| Entrance Condition | 1. User role is Officer  2. Officer Category has a referenced file  3. Officer clicks on a category that has a document reference |
| Input | The name of the file |
| Expected outcome | User is able to open or download a document that has appropriate metadata for the selected category |

**Integration Test**

|  |  |
| --- | --- |
| Purpose | Verify that documents with category metadata for one category cannot be viewed in another category |
| Entrance Condition | 1. User role is Officer  2. User clicks on a category (such as BOLO) |
| Expected Outcome | User is only able to view documents that are of category “BOLO” in their metadata entry |

# User Story #229: List Pinned Documents

**Sunny Day:**

|  |  |
| --- | --- |
| Purpose | Verify that documents on top panel are pinned and belong to correct category listing |
| Entrance Condition | 1. User is logged in as Officer role  2. User selects category they wish to filter documents by |
| Input | The category to list documents by |
| Expected Output | List of document icons with documents name that belong to the selected category, and pinned by Supervisor upon uploading to system |

**Rainy Day**

|  |  |
| --- | --- |
| Purpose | Verify only pinned documents appear in top document panel |
| Entrance Condition | 1. Supervisor uploaded previous document and selected pinned checkbox  2. Officer selects category to view categories by |
| Input | The category to list documents by |
| Expected Output | List of document icons with documents name that belong to the selected category, and pinned by Supervisor upon uploading to system |

**Integration Test**

|  |  |
| --- | --- |
| Purpose | Verify only pinned documents under the selected category appear in top document panel |
| Entrance Condition | 1. Supervisor uploaded previous document and selected pinned checkbox and category from dropdown list  2. Officer selects category to view categories by |
| Input | The user selects Category from list to view documents |
| Expected Output | List of document icons with documents name that belong to the selected category, and pinned by Supervisor upon uploading to system |

# User Story #230: List Unpinned Documents

**Sunny Day:**

|  |  |
| --- | --- |
| Purpose | Verify that documents on bottom panel are not pinned and belong to correct category listing |
| Entrance Condition | 1. User is logged in as Officer role  2. User selects category they wish to filter documents by |
| Input | The category to list documents by |
| Expected Output | List of document icons with documents name that belong to the selected category, and unpinned by Supervisor upon uploading to system |

**Rainy Day**

|  |  |
| --- | --- |
| Purpose | Verify only unpinned documents appear in bottom document panel |
| Entrance Condition | 1. Supervisor uploaded previous document and did not check pinned checkbox  2. Officer selects category to view categories by |
| Input | The category to list documents by |
| Expected Output | List of document icons with documents name that belong to the selected category, and not pinned by Supervisor upon uploading to system |

**Integration Test**

|  |  |
| --- | --- |
| Purpose | Verify only unpinned documents under the selected category appear in bottom document panel |
| Entrance Condition | 1. Supervisor uploaded previous document, did not check pinned checkbox, and selected category from dropdown list  2. Officer selects category to view categories by |
| Input | The user selects Category from list to view documents |
| Expected Output | List of document icons with documents name that belong to the selected category, and unpinned by Supervisor upon uploading to system |

# Glossary

Administrator: A user role with the ability to add/edit/delete other users, change the internal logo and name of the application, add categories for document type assignment, and can proxy the roles of supervisors and officer.

Categories: The classification that a document falls under. Is used to collect documents into appropriate groups.

Document: A file loaded into the system for officers to view and/or download. Belongs to a category, which is created by an administrator.

Officer: Basic user role with the ability to view uploaded documents, which grouped by categories.

Pinned: A special document marked as “important” and remains at the top of the documents list view, until marked as “unpinned” in the system.

Supervisor: A user role with the ability to add documents assign document categories to the system, resets the password of users, and proxy the role of an officer

VRC: Initialism of Virtual Roll Call, which is the web-based application where officers can login to view the documents and information that supervisors would go over at a traditional roll call

# Appendix

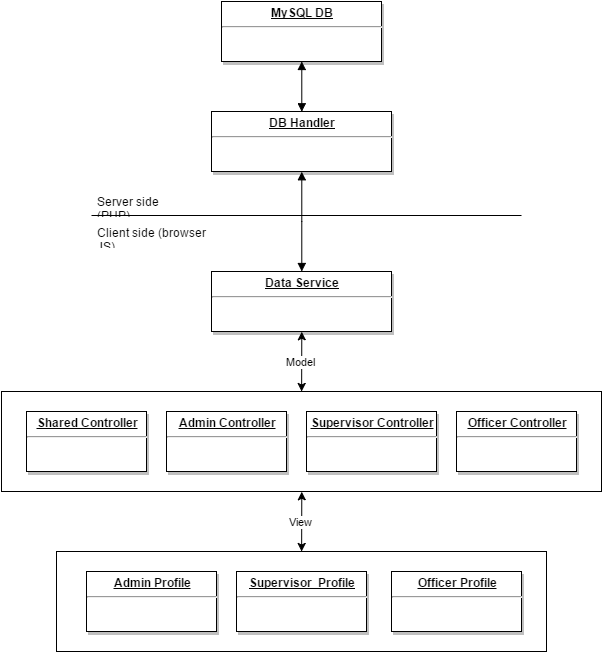
## Appendix A - UML Diagrams

### Figure 1: Use Case Diagram

## 

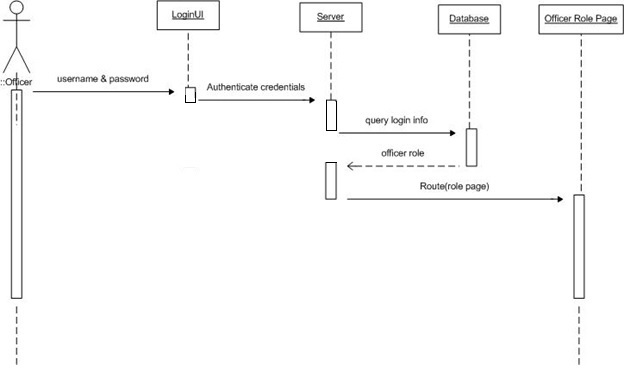
### 

### Figure 2: Class Diagram



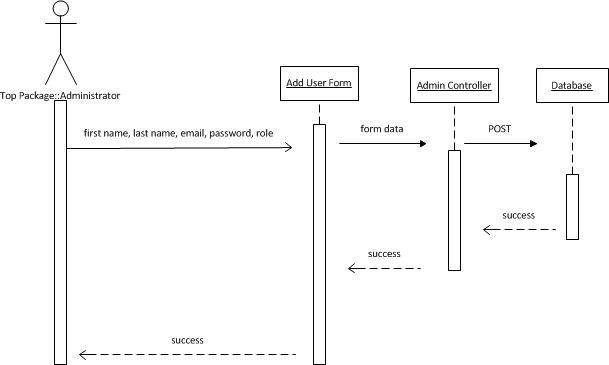
## 

### Figure 3: Login sequence diagram

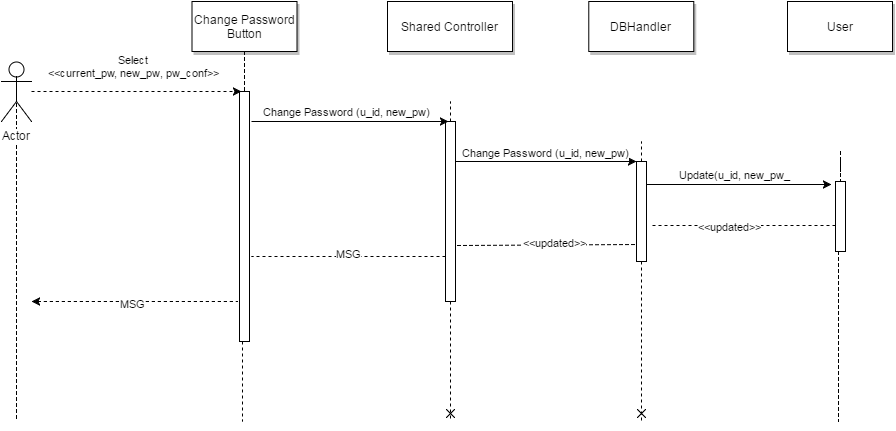


## 

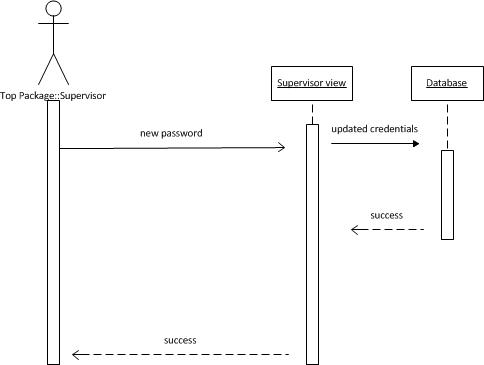
### Figure 4: Add user sequence diagram



### Figure 5: Change Password sequence diagram

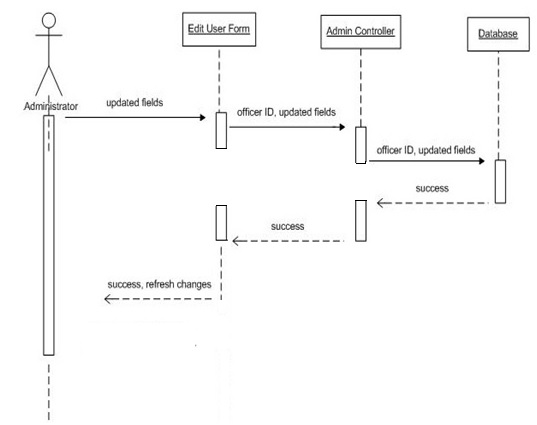


### Figure 6: Reset Password sequence diagram

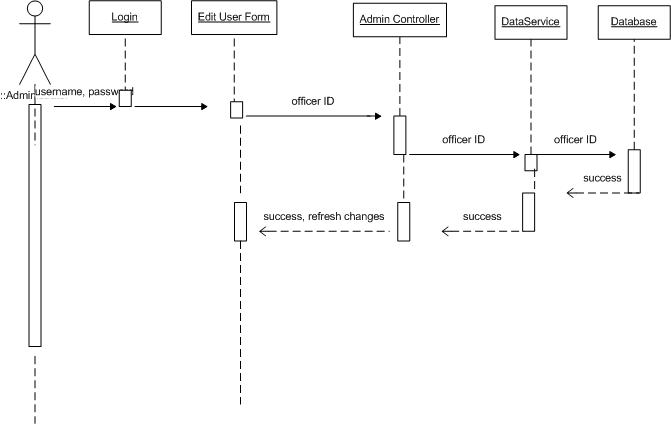


### 

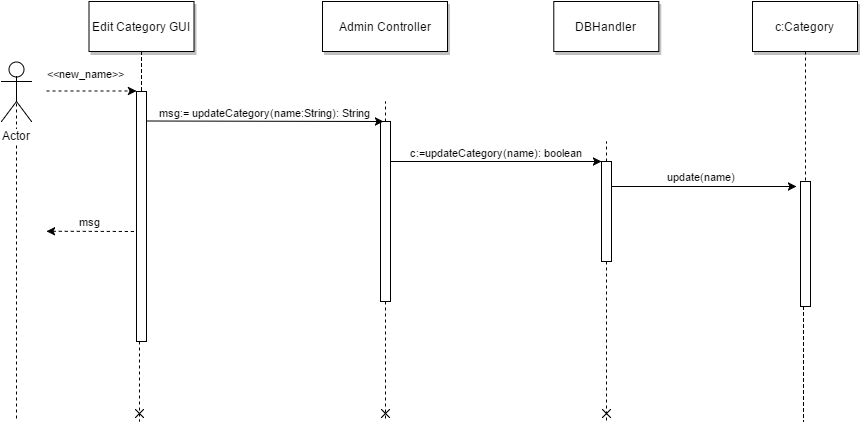
### Figure 7: Edit User sequence diagram



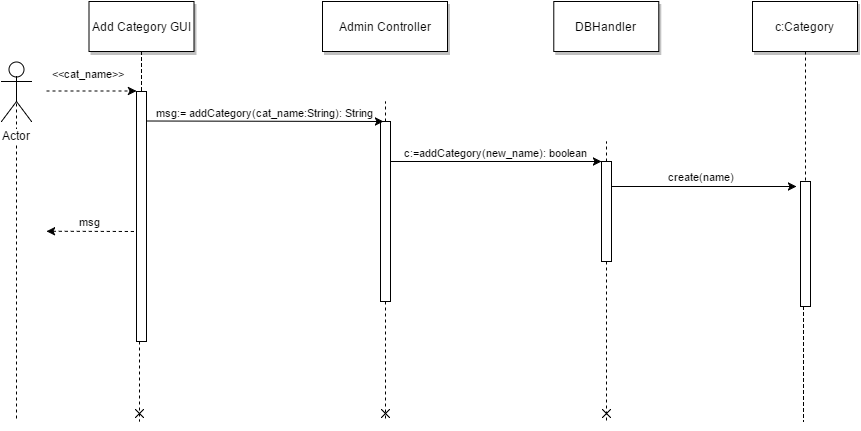
### Figure 8: Delete User sequence diagram



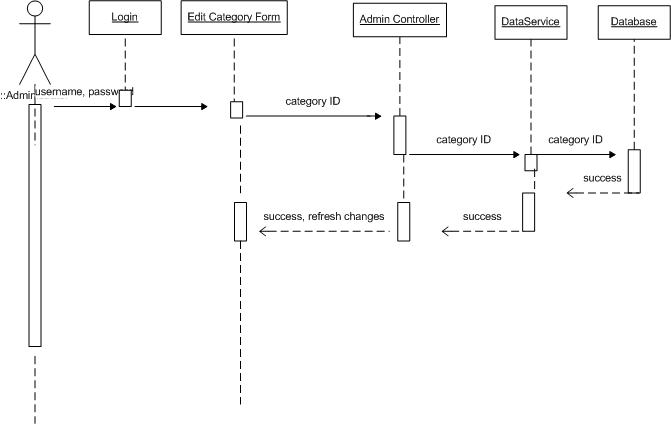
### Figure 9: Edit Category sequence diagram



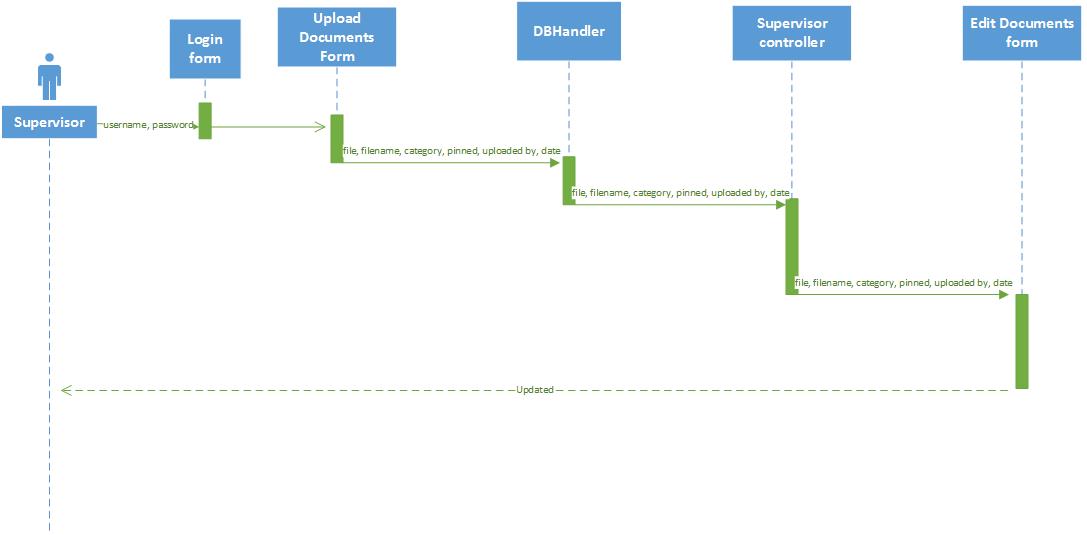
### Figure 10: Add Category sequence diagram



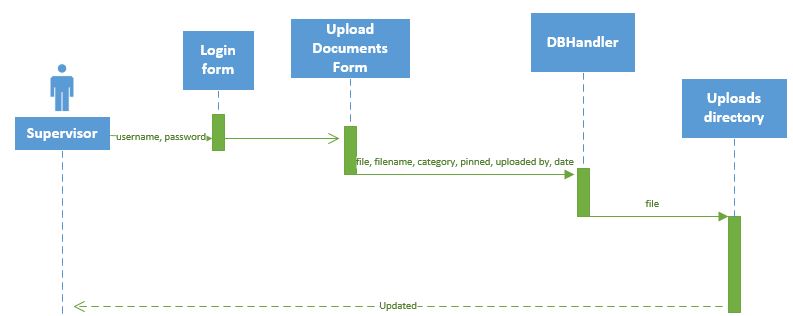
### Figure 11: Remove Category sequence diagram



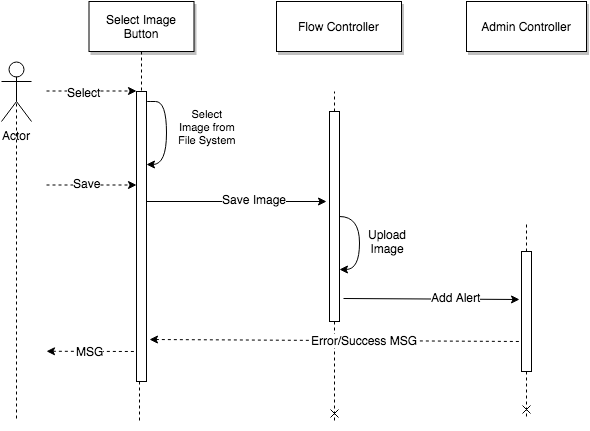
### Figure 12: Add Documents sequence diagram



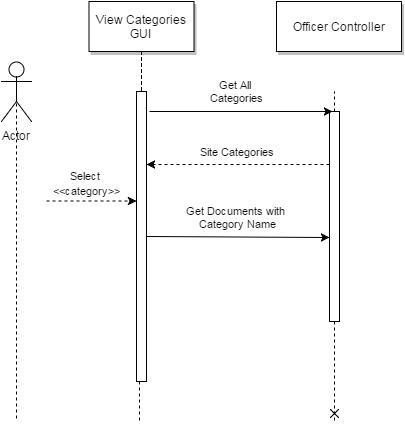
### Figure 13: Upload Document sequence diagram



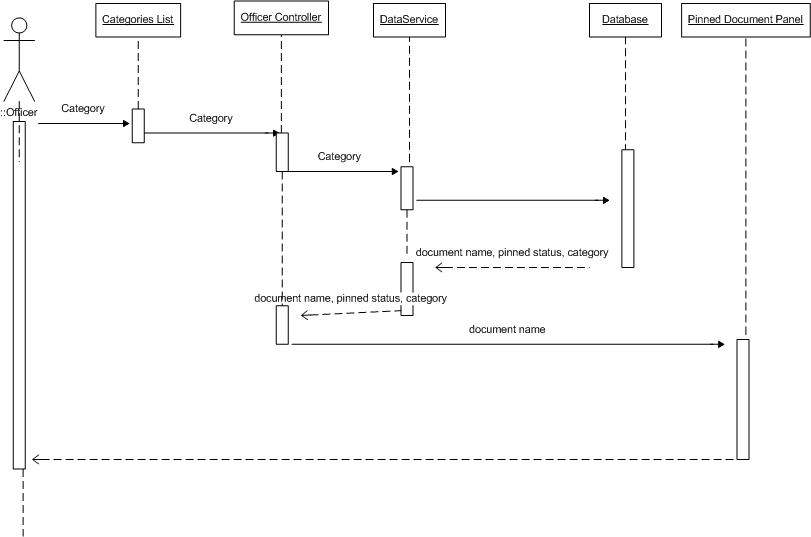
### Figure 14: Customize Site Settings sequence diagram



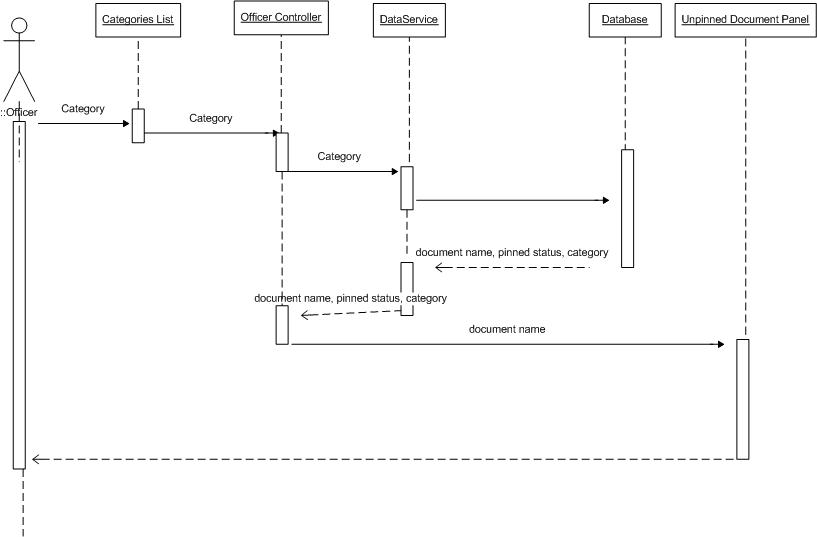
### Figure 15: View Categories sequence diagram



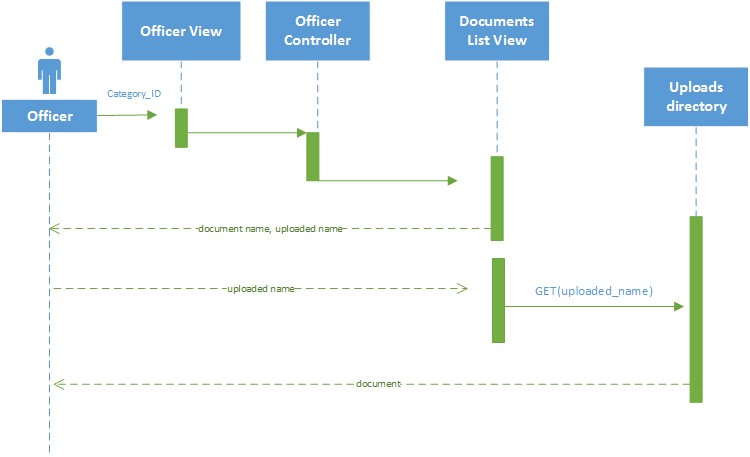
### Figure 16: List Pinned Documents sequence diagram



### Figure 17: List Unpinned Documents sequence diagram



### Figure 18: View Documents sequence diagram



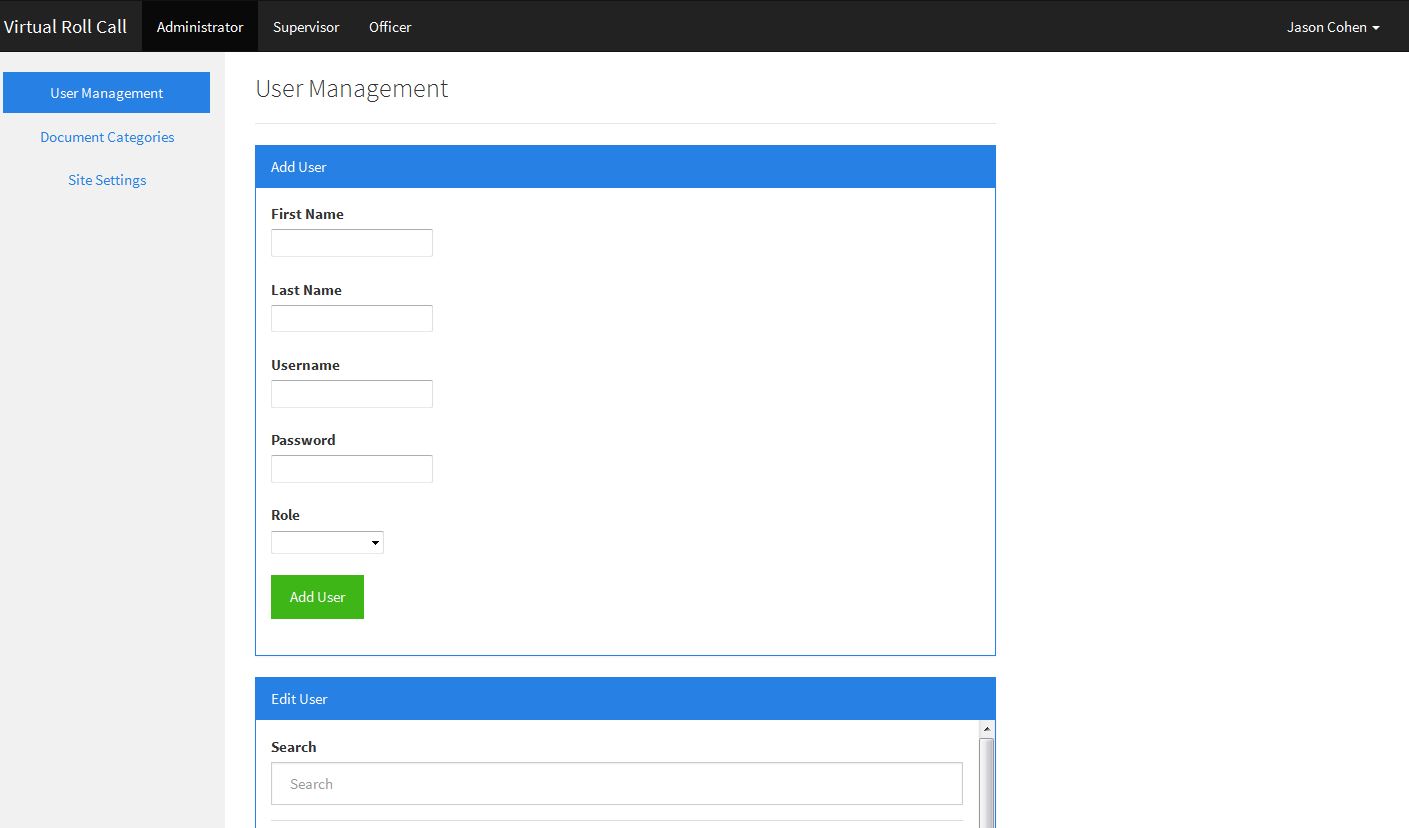
## 

## 

## Appendix B - User Interface Design

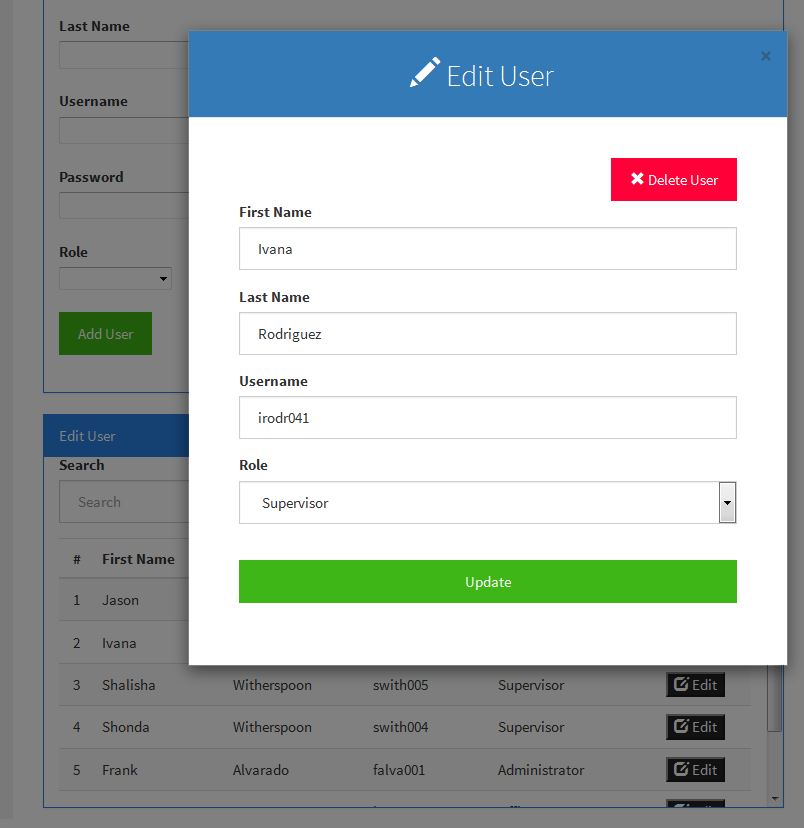
## 

**(VRC login page)**

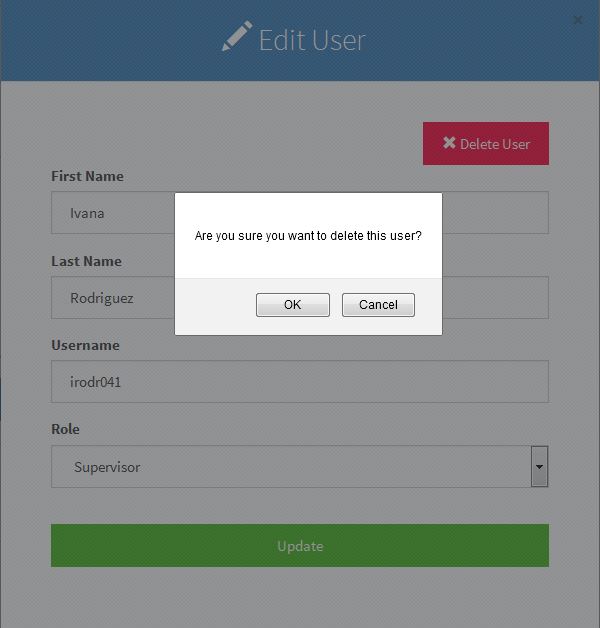
**(Administrator Dashboard - Defaults to User Management)**

## 

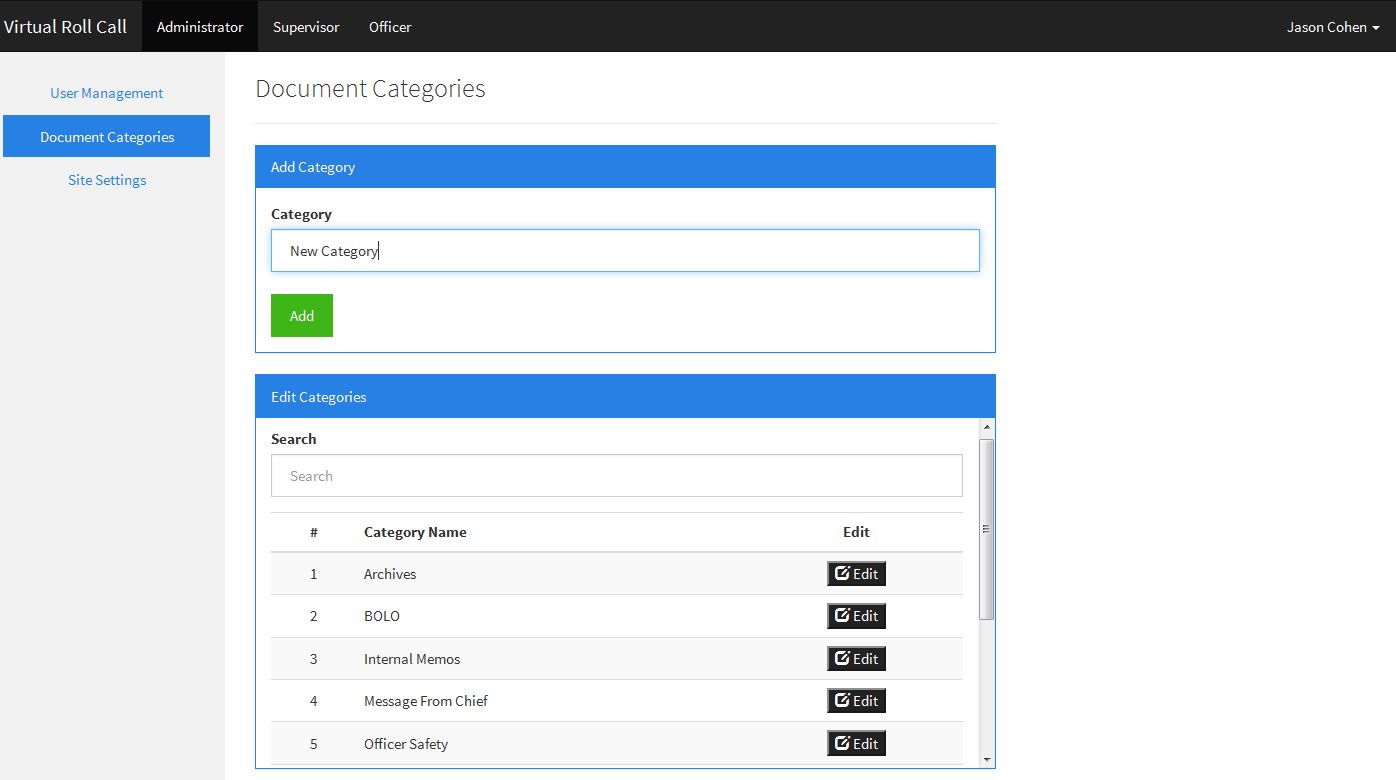
**(Edit User Pane - List of users in the systems are displayed and can be filtered by attribute)**



**(Edit User Module)**



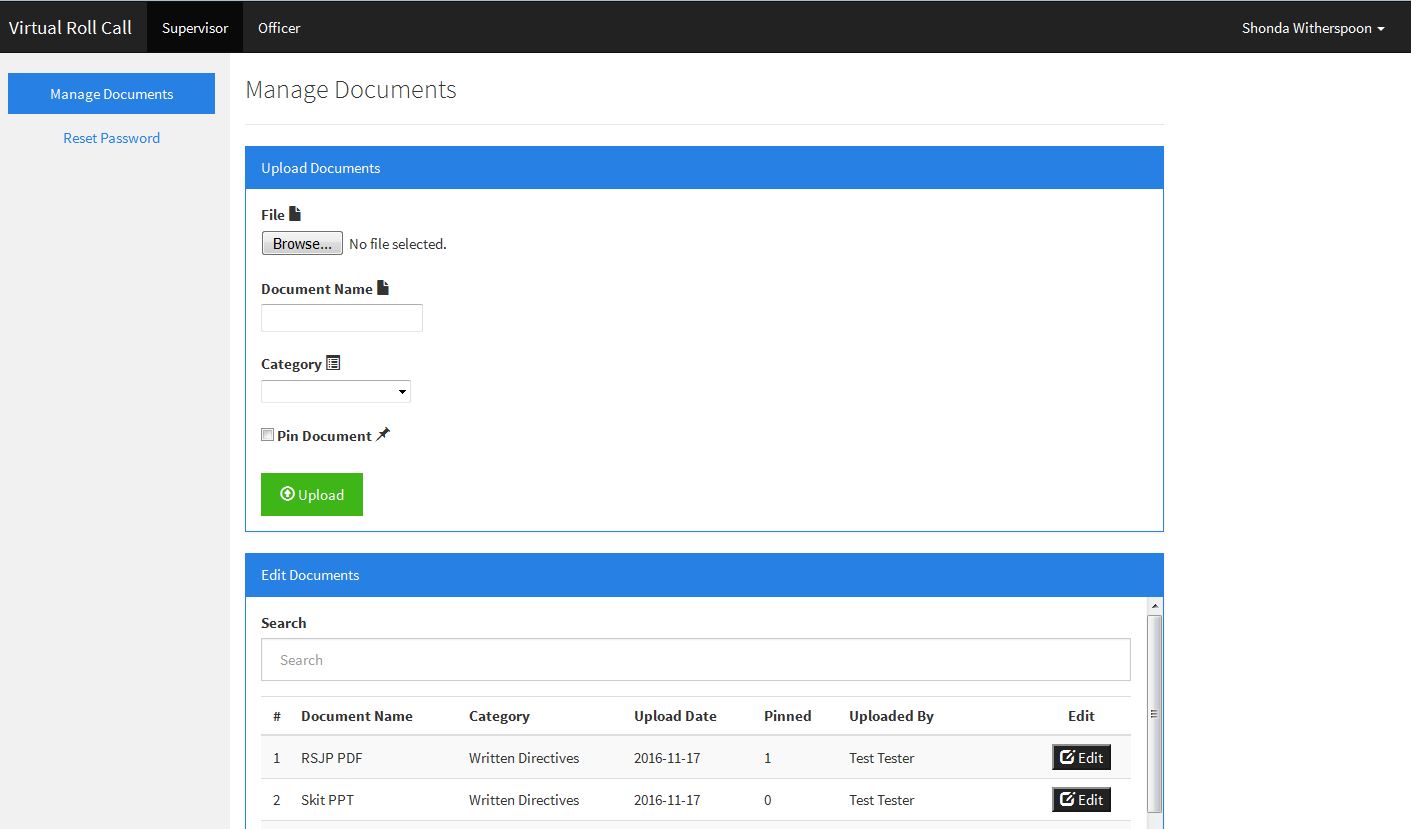
**(Delete User Button Selected)**



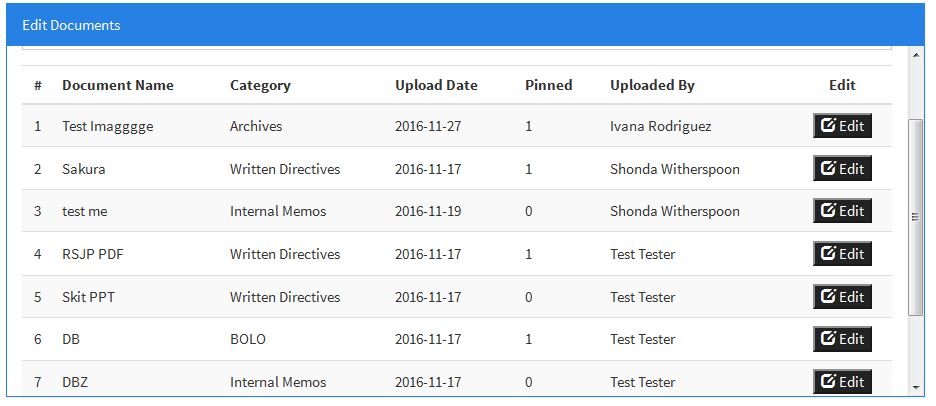
**(Add Categories Pane - available categories shown in second pane)**

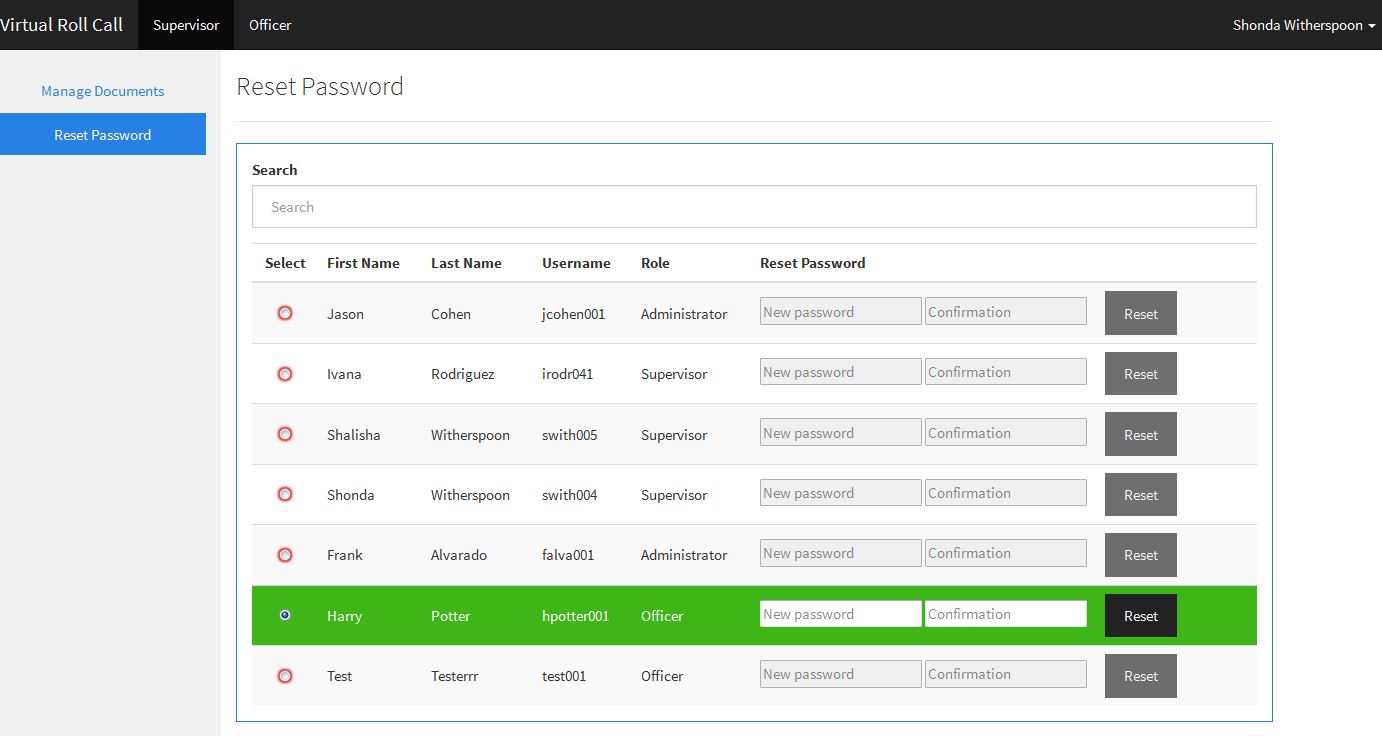
## 

**(Site Settings Page)**

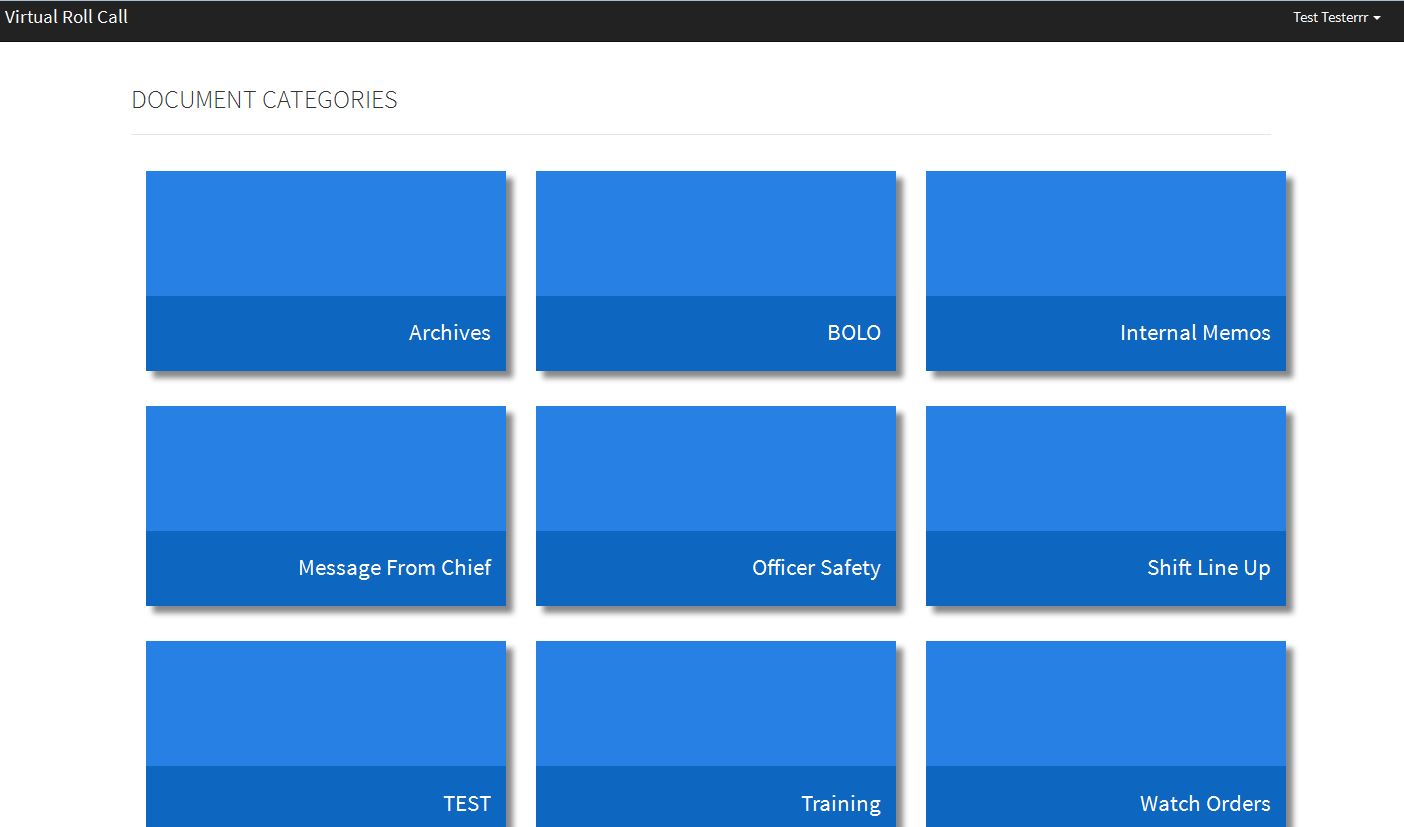


**(Supervisor Dashboard - Defaults to Manage Documents)**

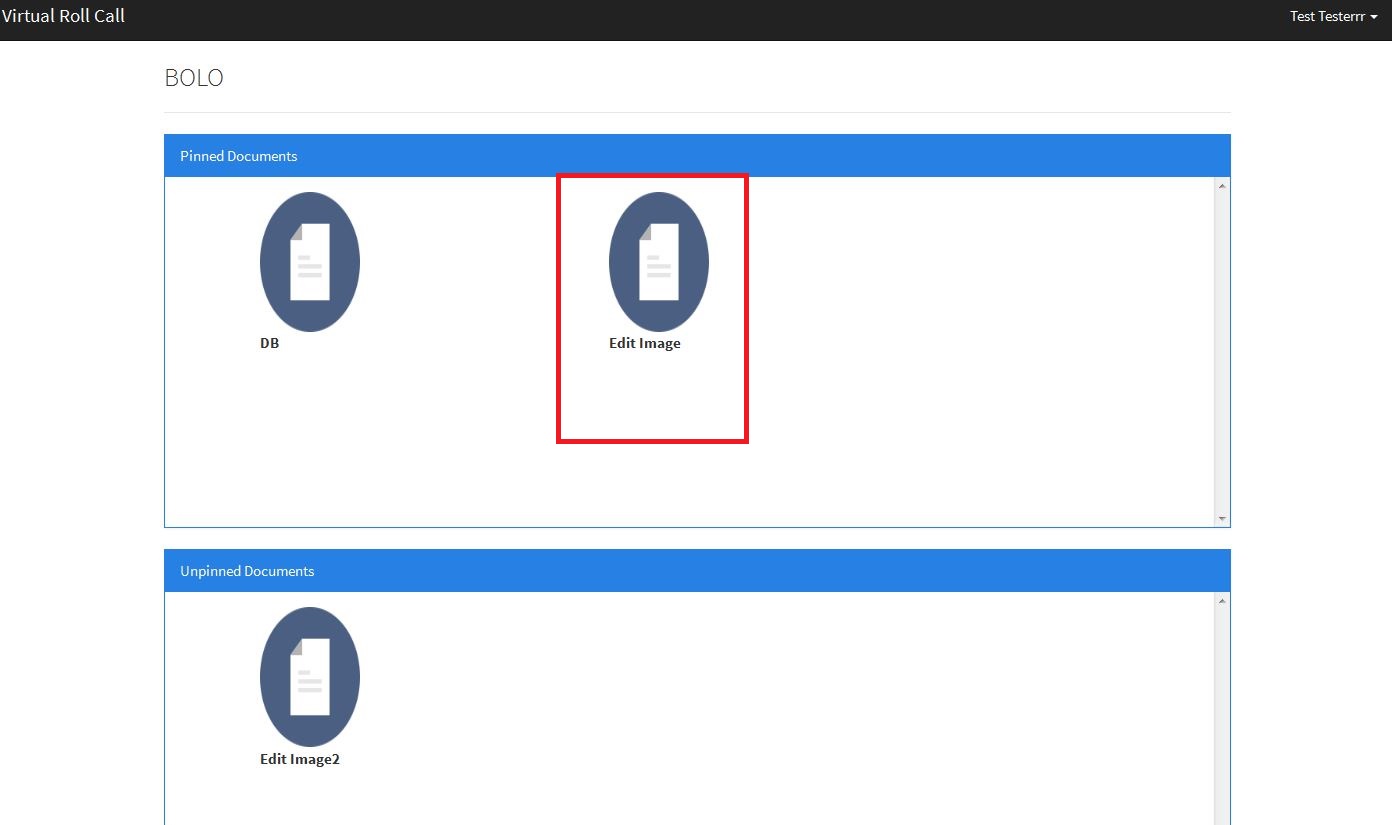
**(Edit Documents Pane - lists metadata of all documents uploaded in the system)**



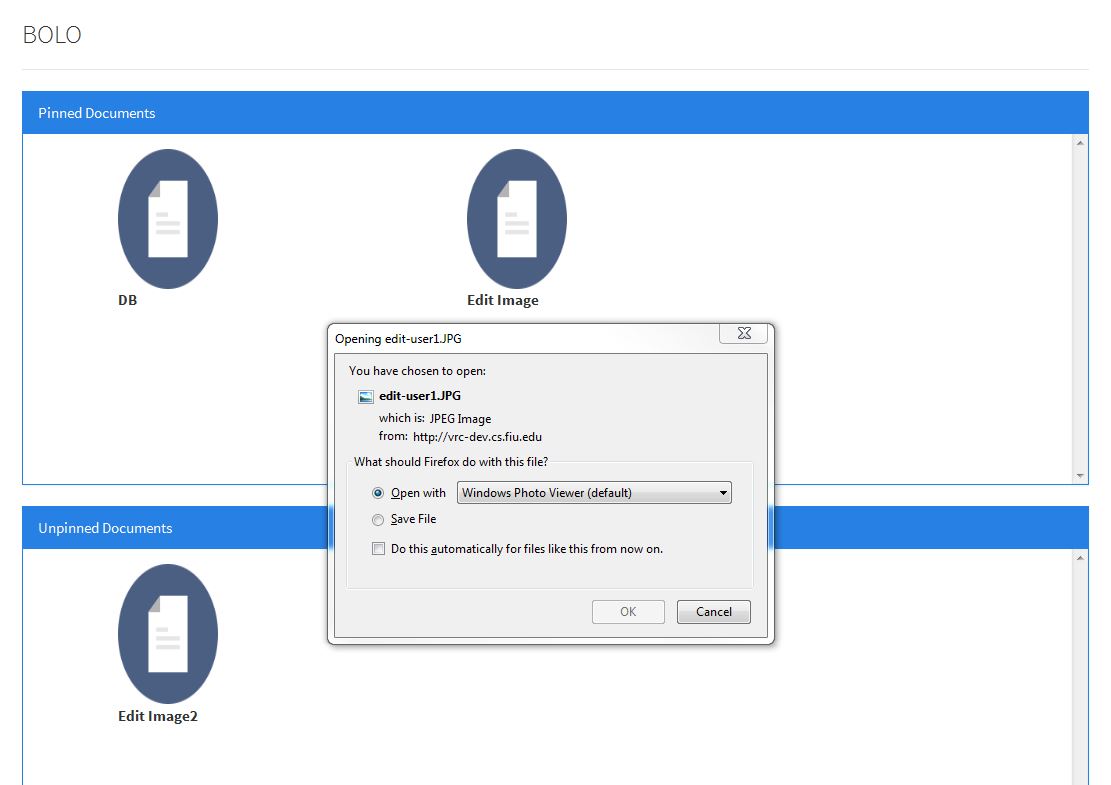
**(Reset Password Page)**



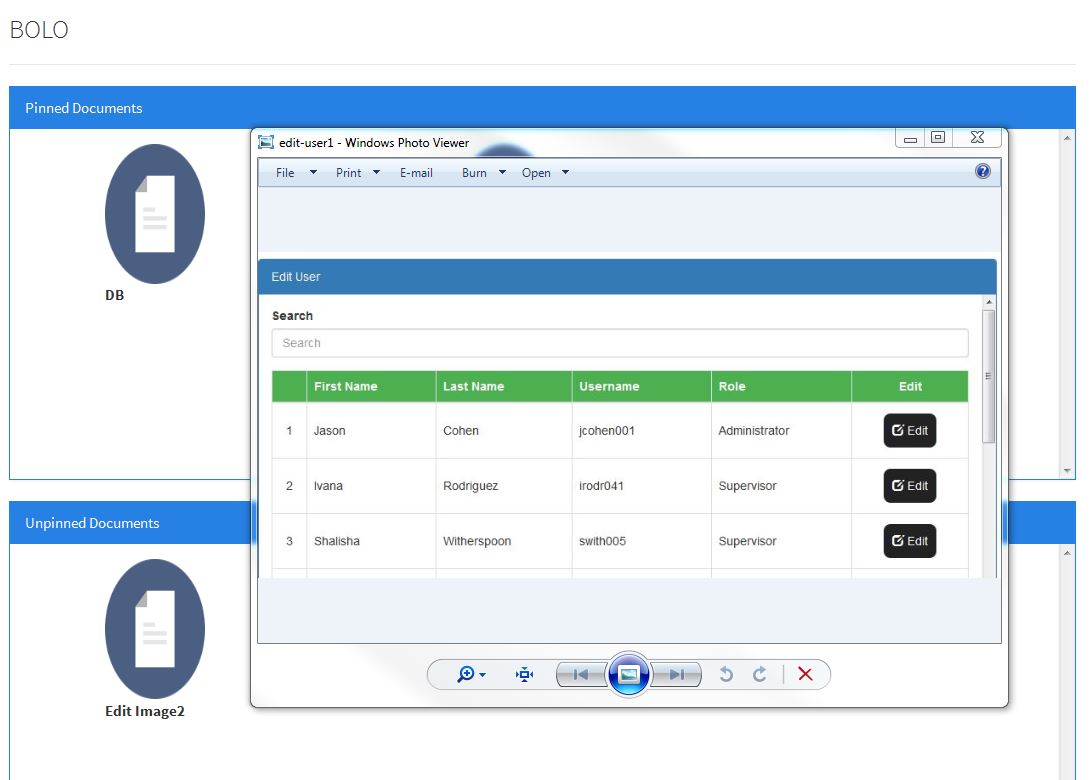
**(Officer Dashboard)**



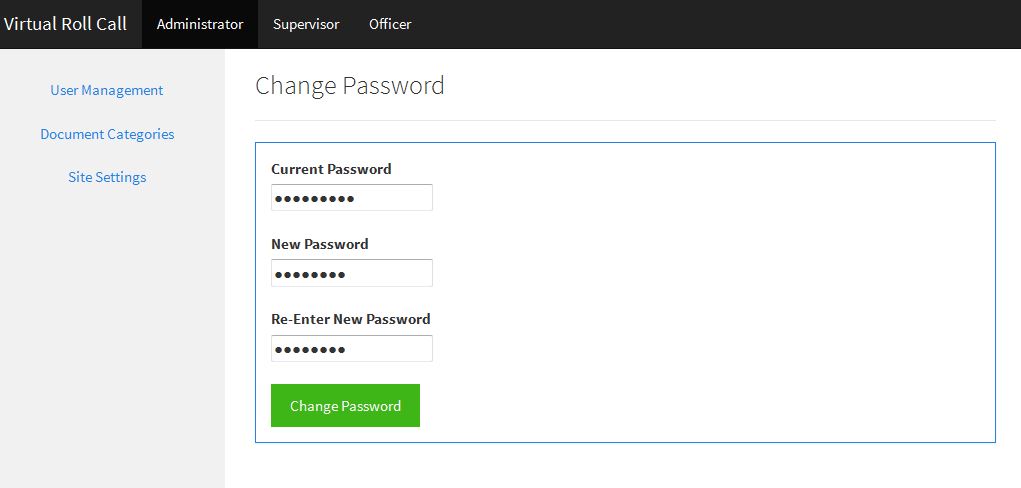
**(Viewing documents in a category - BOLO in this example. Image highlighted will be selected for view/download)**



**(After electing “Edit Image” file icon - user is prompted to open or download the file)**



**(Officer viewing the image)**

**(Change Password Page)**

## 

## 

## Appendix C - Sprint Review Reports

**Date: September 9, 2016**

Attendees: Ivana Rodriguez, Shalisha Witherspoon, Shonda Witherspoon, Jason Cohen

Start time: 1:00PM

End time: 1:10PM

After a show and tell presentation, the implementation of the following user stories were accepted by the product owners:

* He accepted our mockup for the administrative panel

The following ones were rejected and moved back to the product backlog to be assigned to a future sprint at a future Spring Planning meeting.

* Not applicable to this sprint
  + How this should be reflected on the user story definition in Mingle:
    - Not applicable

**Date: September 22, 2016**

Attendees: Ivana Rodriguez, Shalisha Witherspoon, Shonda Witherspoon, Jason Cohen

Start time: 2:00PM

End time: 2:42PM

After a show and tell presentation, the implementation of the following user stories were accepted by the product owners:

* None

The following ones were rejected and moved back to the product backlog to be assigned to a future sprint at a future Spring Planning meeting.

* User Story 120: Create User Login/Logout
  + How this should be reflected on the user story definition in Mingle:
    - The product owner added additional acceptance criteria items/tasks to this user stories (listed here):
      * Adding requirements to password length and the type of characters allowed
      * Locking the user from the system after 5 failed incorrect password attempts
        + Include a message to contact supervisor to reset password
      * Notifying the user if the username is not found in the database (via message on login page)
      * Allowing the supervisor to reset passwords
      * Having user create a new password after the password is reset

**Date: October 13, 2016**

Attendees: Ivana Rodriguez, Shalisha Witherspoon, Shonda Witherspoon, Jason Cohen

Start time: 2:00PM

End time: 2:35PM

After a show and tell of the presentation, the implementation of the following user stories were accepted by the product owners:

* Add User
* Change Password

The following ones were rejected and moved back to the product backlog to be assigned to a future sprint at a future Spring Planning meeting.

None

**Date: October 20, 2016**

Attendees: Ivana Rodriguez, Shalisha Witherspoon, Shonda Witherspoon, Jason Cohen

Start time: 2:00PM

End time: 2:43PM

After a show and tell of the presentation, the implementation of the following user stories were accepted by the product owners:

* Edit User
* Delete User
* Reset Password

The following ones were rejected and moved back to the product backlog to be assigned to a future sprint at a future Spring Planning meeting.

None

**Date: November 3, 2016**

Attendees: Ivana Rodriguez, Shalisha Witherspoon, Shonda Witherspoon, Jason Cohen

Start time: 2:00PM

End time: 2:37PM

After a show and tell of the presentation, the implementation of the following user stories were accepted by the product owners:

* Remove Categories
* Add Categories
* Edit Categories
* Add Document
* Upload Document

The following ones were rejected and moved back to the product backlog to be assigned to a future sprint at a future Spring Planning meeting.

None

**Date: November 17, 2016**

Attendees: Ivana Rodriguez, Shalisha Witherspoon, Shonda Witherspoon, Jason Cohen

Start time: 2:00PM

End time: 2:40PM

After a show and tell of the presentation, the implementation of the following user stories were accepted by the product owners:

* Customize Site Settings
* Upload Document
* View Categories
* View Pinned/Unpinned Documents

The following ones were rejected and moved back to the product backlog to be assigned to a future sprint at a future Spring Planning meeting.

None

## Appendix D - User Manuals, Installation/Maintenance Document, Shortcomings/Wish list Document and other documents

# Links to videos:

**Project Introduction**:

<https://www.youtube.com/watch?v=UmPRpwVW7lA&t=5s>

**User Manuals:**

Administrator

<https://www.youtube.com/watch?v=yUgOqZ28K8k&list=PLG3lylzOg8FgjQudiyxgwPXEfpkN5xOIT&index=2>

Supervisor

<https://www.youtube.com/watch?v=EQKR1bkqgrw&index=3&list=PLG3lylzOg8FgjQudiyxgwPXEfpkN5xOIT>

Officer

<https://www.youtube.com/watch?v=h7Ge2EJo_C0&list=PLG3lylzOg8FgjQudiyxgwPXEfpkN5xOIT&index=4>

**Installation:**

<https://youtu.be/Ecl8LQ_maiQ>

**Shortcomings/Wish list**

<https://youtu.be/2m3T_7iZHJQ>

# 

# References

**Overall, Web Technologies**

* W3Schools
* Google
* Github

**HTML/CSS**

* Bootstrap CSS
* Bootswatch
* Angular UI

**AngularJS**

* Core AngularJS & Additional Modules
* Ng-Flow
* Code School
* Team Treehouse
* Stack Overflow
* AngularJS Documentation

**PHP/MySQL**

* Team Treehouse
* Stack Overflow
* PHP Documentation