Florida International University

School of Computing and Information Sciences

Virtual Roll Call 3.0

Senior Project Fall 2017

Final Deliverable

**Team Members:**

Juan Lopez

Jean Faustin

**Product Owner:**

Jason Cohen

Frank Alvarado

**Mentor(s):**

Masoud Sadjadi

Leila Zahedi

**Instructor:**

Masoud Sadjadi

The MIT License (MIT)

Copyright (c) *2017 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.

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Description of Change** | **Authors** | **Date** |
| 1.0 | First Version, Initial System Design. | Shalisha Witherspoon, Shonda Witherspoon, Ivana Rodriguez | 11/27/2016 |
| 2.0 | The second release of the VRC app and the main objective was to fix some bugs from previous version as well as develop some new critical features that could make the application usable for the first time in production.  New Features:  -Document Status and Confirmation  -Document Log and Log Report.  -Archive Documents | Eric Fernandez, Oscar Yannelli | 7/17/2017 |
| 3.0 | The third version of the project, which focuses on adding new features to enhance the usability and utility of the VRC app. | Juan Lopez, Jean Faustin | 12/1/2017 |

**Abstract**

Police officers use a roll call system 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 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 and watch orders to the system and Officers to view/download these documents and orders. 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.

Table of Contents

[1. INTRODUCTION 7](#_Toc499831261)

[1.1 Current System 7](#_Toc499831262)

[1.2 Purpose of New System 7](#_Toc499831263)

[2. USER STORIES 8](#_Toc499831264)

[2.1 Implemented User Stories 8](#_Toc499831265)

[2.2 Pending User Stories 9](#_Toc499831266)

[3. PROJECT PLAN 10](#_Toc499831267)

[3.1 Hardware and Software Requirements 10](#_Toc499831268)

[3.2 Sprint Plan 11](#_Toc499831269)

[3.2.1 Version 1.0 11](#_Toc499831270)

[3.2.2 Version 2.0 18](#_Toc499831271)

[3.2.3 Version 3.0 21](#_Toc499831272)

[3.2.3 Not Yet Implemented User Stories 30](#_Toc499831273)

[4. SYSTEM DESIGN 31](#_Toc499831274)

[4.1 Architectural Patterns 31](#_Toc499831275)

[4.1.1 Model-View-Controller 31](#_Toc499831276)

[4.1.2 Client-Server 32](#_Toc499831277)

[4.2 ER Model 33](#_Toc499831278)

[4.3 Deployment Diagram 34](#_Toc499831279)

[4.4 Use Case Diagram 35](#_Toc499831280)

[4.5 Object Interaction 36](#_Toc499831281)

[5. SYSTEM VALIDATION AND TESTING PROCESS 55](#_Toc499831282)

[5.1 Version 1.0 55](#_Toc499831283)

[5.2 Version 2.0 71](#_Toc499831284)

[5.3 Version 3.0 74](#_Toc499831285)

[6. GLOSSARY 102](#_Toc499831286)

[7. APPENDIX 104](#_Toc499831287)

[Appendix A - UML Diagrams 104](#_Toc499831288)

[Appendix B - Sequence Diagrams Versions 1 and 2 105](#_Toc499831289)

[Appendix C - User Interface Design 115](#_Toc499831290)

[Appendix D - Sprint Review Reports 143](#_Toc499831291)

[Version 1.0 143](#_Toc499831292)

[Version 2.0 145](#_Toc499831293)

[Version 3.0 147](#_Toc499831294)

[Appendix E - User Manuals, Installation/Maintenance Document, Shortcomings/Wish list Document and other documents 150](#_Toc499831295)

[Appendix F - References 151](#_Toc499831296)

# 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.

## 1.1 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.

## 1.2 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 describes the user stories that were implemented in the previous and current 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 describes the user stories that will be considered for future development.

## Implemented User Stories

**Version 1.0**

* **User Story 117: Learning, installation, and preparation**
* **User Story 121: Create User Login**
* **User Story 129: Add User**
* **User Story 130: Remove User**
* **User Story 131: Customize Site Settings**
* **User Story 132: Add Category**
* **User Story 133: Upload Document**
* **User Story 140: Reset Password**
* **User Story 141: Change Password**
* **User Story 168: Edit User**
* **User Story 218: Edit Category**
* **User Story 226: Remove Category**
* **User Story 227: Add Document**
* **User Story 228: View Categories**
* **User Story 229: List Pinned Documents**
* **User Story 230: List Unpinned Documents**
* **User Story 232: View Document**

**Version 2.0**

* **User Story 231: Change Role View**
* **User Story 233: Add Document Confirmation**
* **User Story 244: Add User Logs**
* **User Story 245: Archive Documents**
* **User Story 249: View Document in App**
* **User Story 273: Add description to documents table and upload document screen**
* **User Story 274: Add document upload confirmation**
* **User Story 275: Add column to Supervisor Documents grid that shows  
  whether the document is active or archived**
* **User Story 277: Display all documents on Logs grid**
* **User Story 276: Fix Supervisor document edit screen that is not working**

**Version 3.0**

* **User Story 147: Lock User**
* **User Story 300: Improve Document List Presentation**
* **User Story 241: Add Users with CSV File**
* **User Story 242: Change Display Mode**
* **User Story 243: Add Document Quiz**
* **User Story 246: Add Watch Orders with CSV File**
* **User Story 247: View Watch Orders on Map**
* **User Story 248: Add Text Information**
* **User Story 250: Add Password Encryption**
* **User Story 295: Display Documents within Website**
* **User Story 296: Display Pending Documents Notification**
* **User Story 301: Application Distribution Package**
* **User Story 300: Improve Document List Presentation**
* **User Story 302: Change Navigation Bar Based on User Role**
* **User Story 304: Add Watch Order**
* **User Story 306: Delete Watch Order**
* **User Story 307: Edit Watch Order**
* **User Story 308: Edit Free Message**
* **User Story 309: Automatically Delete Watch Orders**
* **User Story 310: Display Watch Order List Table**
* **User Story 312: Take Quiz**

## 2.2 Pending User Stories

* **User Story 311: Change Map Based on Selected Watch Order**
* **User Story 313: Highlight Selected Watch Order**
* **User Story 314: Change Document Order Based on Officer Preference**
* **User Story 315: Add Real-time Chat Messages**
* **User Story 316: Add Voice Messages**
* **User Story 317: Improve Password Security**
* **User Story 318: Close Window Session**

# PROJECT PLAN

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

## Hardware and Software Requirements

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.

## 3.2 Sprint Plan

### 3.2.1 Version 1.0

#### 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 metadata displayed in supervisor view Document table

**Entrance Condition:**

* Supervisor role only to enter Document metadata in the form

**Exit Condition**

* Document metadata is added to the system

#### User Story 218: Edit Category

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

**Acceptance Criteria:**

* As an admin, I should have access to view and edit document categories.
* Name of a category changes to automatically reflect 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 and department names are 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 to view list of unpinned documents separately from pinned ones.

**Entrance condition:**

* Officer (role) 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

### 3.2.2 Version 2.0

#### Sprint 2

#### User Story 249: View Document

*Instead of download the files, the project owner requested that the officer could see the document in the same app (browser) the files accepted for this task are images (JPG,PNG) and PDF files, documents in the word format, would have to be downloaded and seeing in the computer.*

#### Acceptance Criteria:

* All the documents PDF can be seeing in the browser without the need of downloading the document.
* All the Document type Image can be seeing in the browser without having to download the document.
* Other Document type has to be downloaded and open with any available program in the computer.

#### Sprint 3

#### User Story 233: Add Document Confirmation

*As a Supervisor, I would like that newly added documents to show with status “Pending”, then when the officer clicks on the document for the first time change the status to “Reviewed” and finally I would also like to have a check mark with a disclaimer that specify that officer have read and understood the content of the document and the when this check mark is clicked the status must change to “Done”.*

**Acceptance Criteria:**

* Newly added documents to show with status “Pending”.
* Change status to “Reviewed” when the officer clicks on the document for the first time.
* Have a check mark with a disclaimer that specify that officer have read and understood the content of the document and the when this check mark is clicked the status must change to “Done”.
* When status change to “Reviewed”, it must record the read start date time.
* When status changes to “Done”, it must record the read end date time.
* Documents cannot be returned to a previous status.
* These functionalities must be available to the officer regardless of document being on the “Current” or “Archived” view.

#### User Story 244: Add User Log

*Every time an officer accesses a document the log has to save this action, by recording the user the date and the document acceded. There is need of showing this information in the supervisor profile*

**Acceptance Criteria:**

* The data will be inserted every time an officer views any of the documents
* The logs cannot be deleted, by no one.
* Only the supervisor will see the logs.

#### 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*

**Acceptance Criteria:**

* Administrator should be able to view site as Officer
* Administrator should be able to view site as Supervisor
* Administrator should be able to return their own view

**Entrance Condition:**

* User is logged in as administrator

**Exit Condition:**

* Administrator is redirected to appropriate role based view

#### Sprint 4

#### User Story 245: Archived Documents

*As an officer I will like to have all documents older than 7 days moved into another view called Archived Documents within each of the categories. As an administrator, I like the option to physically delete documents based on a date range. Pinned document cannot be either deleted or moved into the archived view.*

**Acceptance Criteria:**

* All documents older than 7 days moved into another view called Archived Documents within each of the categories.
* All documents on a date range entered will be physically deleted.
* Pinned document cannot be either deleted or moved into the archived view.

#### User Story 275: Column on the documents grid that displays whether the document is active or archived

*As a Supervisor, I will like to have a column on the documents grid that displays whether the document is active or archived.*

**Acceptance Criteria:**

* Have a column on the documents grid that displays whether the document is active or archived.

#### User Story 273: Add description field to documents table and to upload document screen

*As a Supervisor, I would like to have available an additional field on the document upload screen. This field will be used as a short description of the document content and will be displayed to officers on the documents card’s view.*

**Acceptance Criteria:**

* New field must be available to be fill from the document upload screen.
* New field must be displayed to officers on the documents card’s view.

#### User Story 274: Add document upload confirmation

*As a Supervisor, I will like to have confirmation or have the page refresh after a document is uploaded. Now when I click on the upload document button looks like nothing happened, but is actually uploading the documents.*

**Acceptance Criteria:**

* See visual confirmation or have the page refresh after a document is uploaded.

#### User Story 277: Display all documents on Logs grid. Add columns status, start date/time, end date/time and duration time

*As a Supervisor, I will like to be able to pull a report from the logs grid that can show me all document with its status (Pending, Reviewed, Completed) for each officer. For example, if I want to pull a list of all pending documents for an officer “OfficerA”, I will type the officer name and the document status “Pending” and the outcome should be a document with status “Pending” for “OfficerA “. Also, I will like to have available the columns for Start Date/Time, End Date/Time and Duration Time.*

**Acceptance Criteria:**

* Documents will have to be included on the grid, even when not yet logged.
* For documents that doesn’t have any entry on the logs table the status will be “Pending”, otherwise whatever status is set.
* Add columns Start Date/Time, End Date/Time and Duration Time.

#### User Story 276: Fix Supervisor document edit screen that is not working.

*As a Supervisor I will like to be able to use the edit document feature that it is supposed to be already implemented from previous release but is not working.*

**Acceptance Criteria:**

* Be to edit document information. (e.g. change document from Pinned to Unpinned).

### 3.2.3 Version 3.0

#### Sprint 2

#### 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.*

#### Acceptance Criteria:

1. There must be an option in the website header dropdown to manually change the display mode.
2. When night-mode is activated, only dark colors should be used throughout the site.
3. When day-mode is activated, the normal colors (lighter) should be used throughout the site.
4. The user’s display mode preference should be saved so that the display mode is consistent in all pages.
5. When the users logs off, the display mode will be reset to day-mode.

#### User Story 295:

#### Display Documents within Website

*As an officer, I would like to view documents (images and PDFs) inside the boundaries of the web application and easily return to main document page.*

#### Acceptance Criteria:

1. The user has logged into the system.
2. The user has selected the document category to view corresponding documents.
3. If there are no pending documents for the category, no window should open.
4. Documents PDF or Image (.jpg, png) versions are stored and available from the database and saved on the “uploads” folder.

#### Entrance Condition:

1. The Officer has successfully logged into the system.

#### Exit Condition:

1. The officers clicks on the “Close” button or outside the modal window to return to the documents category.

#### User Story 296:

#### Display Pending Documents Notification

*As an officer, I would like to view a notification badge in the main document page that shows the number of pending document per document category so that I can quickly identify which categories contain pending documents.*

#### Acceptance Criteria:

1. When the user views the main document page, a notification badge will be displayed for each document category that shows the total number of pending documents per category
2. When the status of a document changes, it must be reflected in the notification badge.
3. If there are no pending documents for the category, there must be no notification badge displayed.
4. The notification badge must be easily readable and in accordance with the rest of site’s style.
5. The user’s display mode preference should be saved so that the display mode is consistent in all pages.
6. When the users logs off, the display mode will be reset to day-mode.

#### Sprint 3

#### User Story 147:

#### Lock User

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

#### Acceptance Criteria:

1. If the username is not found in the system, notify the user (ex. “Username does not exist in system”).
2. If user enters their password incorrectly 5 times, the system locks the user out of the system.
3. If the user is locked out of the system, they should see a message to contact their supervisor.

#### Entrance Condition:

1. The Officer has successfully logged into the system.

#### Exit Condition:

1. The officers clicks on the “Close” button or outside the modal window to return to the documents category.

#### User Story 241:

#### Add Users with CSV File

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

#### Acceptance Criteria:

1. The result of parsing a CSV file must be displayed in a table of imported users.
2. The system must only accept CSV files, other files must be rejected with a warning.
3. If no CSV file is selected, the system will warn the user.
4. The admin must be able to edit the information of any imported user before the user is added to the database.
5. The admin must be able to delete any imported user before it is added.
6. If a user cannot be added to the database, the system must display an error message and the admin will be able to edit any rejected user and try again.

#### 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.*

#### Acceptance Criteria:

1. The database should not allow anyone to guess how long and the format for passwords.
2. Passwords are alpha-numeric only.
3. All passwords must be encrypted.
4. If user has forgotten or account has been locked, must have Supervisor to reset password.

#### Entrance Condition:

1. The user has entered his/her credentials at login.

#### Exit Condition:

1. The system validates the password against the stored hashed password in the database.

#### User Story 300:

#### Improve Document List Presentation

*As an officer, I would like to view an improved document list where documents have been ordered by their status and have different color labels so that I can easily distinguish between the different types of documents.*

#### Acceptance Criteria:

1. In the document list view, documents must be grouped by their status.
2. Pending documents must appear first, followed by Reviewed documents, and lastly Done documents.
3. Pending documents must have a red status label.
4. Reviewed document must have an orange status label.
5. Done documents must have a green status label.
6. The color of the document status label must update when the status of the document changes.

#### Sprint 4

#### User Story 246:

#### Add Watch Orders with CSV File

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

#### Acceptance Criteria:

1. Each watch order must include an address and the date when it was added.
2. The result of parsing a CSV file must be displayed in a table of imported watch orders.
3. The system must only accept CSV files, other files must be rejected with a warning.
4. If no CSV file is selected, the system will warn the user.
5. The supervisor must be able to edit the information of any imported watch order before the user is added to the database.
6. The supervisor must be able to delete any imported watch before it is added.
7. If a watch order cannot be added to the database, the system must display an error message, and the supervisor will be able to edit any rejected watch order and try again.
8. When the supervisor adds a CSV file, it will overwrite any existing watch orders.

#### User Story 247:

#### View Watch Orders on Map

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

#### Acceptance Criteria:

1. Each watch order should be displayed on a map as a marker.
2. The location of each marker will be determined by longitude and latitude properties of each watch order.
3. When the user clicks a marker, the description and date will be displayed in an information window.
4. If there are no watch orders, no markers will be displayed.
5. If a watch order cannot be added to the database, the system must display an error message, and the supervisor will be able to edit any rejected watch order and try again.
6. When the supervisor adds a CSV file, it will overwrite any existing watch orders.

#### User Story 302:

#### Change Navigation Bar Based on User Role

*As an administrator or supervisor, I want to navigate between roles without opening a new browser to help better and faster accessing information.*

#### Acceptance Criteria:

1. The system should allow an administrator to jump between administrator, supervisor and officer role through the same navigation bar.
2. The system should allow a supervisor to navigate between supervisor and officer roles.
3. The system should allow an officer not to navigate between document categories.
4. The system should allow non-administrator or non-supervisor to only view officer role.

#### Entrance Condition:

1. The user has entered his/her credentials at login.

#### Exit Condition:

1. The system still remains in highest role.
2. The system allows the user to navigate to previous role.

#### Sprint 5

#### User Story 248:

#### Add 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.*

#### Acceptance Criteria:

1. The plain-message must be short.
2. The message must be archived after seven (7) days.
3. An officer should be allowed to pin the message.
4. A notification should be sent upon new messages.

#### User Story 304:

#### Add Watch Order

*As a supervisor, I would like to a single watch order into the system instead of using a CSV file, so that it can later be viewed by officers.*

#### Acceptance Criteria:

1. A watch order consists of a description, address, longitude, latitude, and the date when it was added.
2. The add watch order form must have two fields: description and address.
3. If the user submits the form with empty fields, an error message should be displayed.
4. A notification should be displayed after attempting to add watch order, indicating if the watch order was added to the database.

#### User Story 306:

#### Delete Watch Order

*As a supervisor, I would like to delete watch orders, so that invalid or old watch orders can be removed from the system.*

#### Acceptance Criteria:

1. A watch order consists of a description, address, longitude, latitude, and the date when it was added.
2. The add watch order form must have two fields: description and address.
3. If the user submits the form with empty fields, an error message should be displayed.
4. A notification should be displayed after attempting to add watch order, indicating if the watch order was added to the database.

#### User Story 307:

#### Edit Watch Order

*As a supervisor, I would like to edit specific watch orders because certain properties may be incorrect or outdated.*

#### Acceptance Criteria:

1. The supervisor must be able to edit any watch order.
2. If a watch order is successfully edited, the system must display a success message.
3. If a watch order cannot be edited, the system must display an error message.

#### User Story 308:

#### Edit Free-Text Messages

*As a supervisor, I would like to be able to edit free-form text box so that I can update or add additional information or delete the entire message.*

#### Acceptance Criteria:

1. The edited message should have a title.
2. The edited message should have a description
3. The edited message cannot have an empty body.
4. The user should be able to delete the message from the list.

#### Entrance Condition:

1. The user has successfully logged in.
2. The user has selected the Supervisor tab.
3. The user has clicked on ‘Edit’ button for the message to edit.

#### Exit Condition:

1. The system closes the modal dialog box.
2. The system refreshes the ‘Free Message Upload’ page.
3. The system displays the newly updated Edit Messages list.

#### Sprint 6

#### User Story 243:

#### Add Document Quiz

*As a Supervisor, I should be able to set pop quizzes to help center key points in documents and messages, or to verify the document was read*

#### Acceptance Criteria:

1. There should me at most five (5) questions per quiz.
2. The quiz should not be timed.
3. If a document has a quiz, the document status cannot be updated Done not until the quiz is taken.
4. A scoring message should be displayed after taking the quiz.
5. There is no repeat for the quiz.

#### Entrance Condition:

1. The user has successfully logged in.
2. The user has selected the Supervisor tab.
3. The user has started the process to upload a new document.
4. The Quiz Available checkbox is checked.

#### Exit Condition:

1. The system closes the modal dialog box.
2. The system adds the new document to the Edit Documents list.
3. The multiple choice questions are saved into the database.

#### User Story 309:

#### Automatically Delete Watch Orders

*As a supervisor, I would like for watch orders to be automatically deleted based on their expiration date, so that outdated watch orders are not present in the system.*

#### Acceptance Criteria:

1. If the current date is before the expiration date, the watch order should not be deleted.
2. If the current date is past the expiration date, the watch order should be deleted.
3. Watch orders past their expiration date should never be visible to either supervisors or officers.

#### User Story 310:

#### Display Watch Order List Table

*As an officer, I would like to see a table list of watch orders so that I can quickly search for watch orders instead of having to examine the map.*

#### Acceptance Criteria:

1. All watch orders visible in the map should also be listed in the table.
2. The table must display the following fields for each watch order: address, description, add date, and expiration date.
3. The table must have a search bar.
4. The watch orders listed in the table can only be viewed, not modified.

#### User Story 312:

#### Take Quiz

*As a supervisor, I would like to be able to edit free-form text box so that I can update or add additional information or delete the entire message.*

#### Acceptance Criteria:

1. The quiz is of multiple choice questions

2. The quiz is available and can only be taken once.

3. The document status is updated after taking the quiz.

4. The quiz can only be taken only after the document has been reviewed.

#### Entrance Condition:

1. The user has successfully logged in.

2. The user has selected the Officer tab.

3. The user has reviewed the document.

#### Exit Condition:

1. The system logs the user’s answers to the database  
2. The system updates the document status to ‘Done’.  
3. The system closes the quiz modal window.

### 3.2.3 Not Yet Implemented User Stories

#### User Story 311:

**Change Map Based on Selected Watch Order**

*As an officer, I would like for the watch order map position to change based on the selected watch order from the list table.*

#### User Story 313:

**Highlight Selected Watch Order**

*As an officer, I would like for the watch order map position to change based on the selected watch order from the list table.*

#### User Story 314:

**Change Document Order Based on Officer Preference**

*As an officer, I would like to several options to change the order in which documents are displayed, so I that I can choose the best ordering for my needs.*

#### User Story 315:

**Add Real-time Chat Messages**

*As a supervisor, I would like to send a chat message to all officers, so I can get immediate feedback from any officer.*

#### User Story 316:

**Add Voice Messages**

*As a supervisor, I would like to upload voice messages, so that I can send audio information to officers.*

#### User Story 317:

**Improve Password Security**

*As an administrator, I would like to have stronger password requirements to improve the general security of the application.*

#### User Story 318:

**Close Window Session**

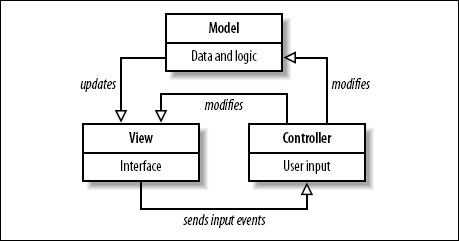
*As an officer, I want to close the window session after I log out so that I don’t preserve access across multiple pages.*

# 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.

## 4.1 Architectural Patterns

### 4.1.1 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:

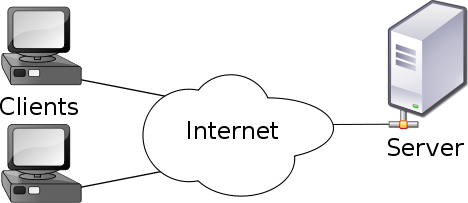
1. Model – manages the data and logic of the application

2. View – The User Interface of the application

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

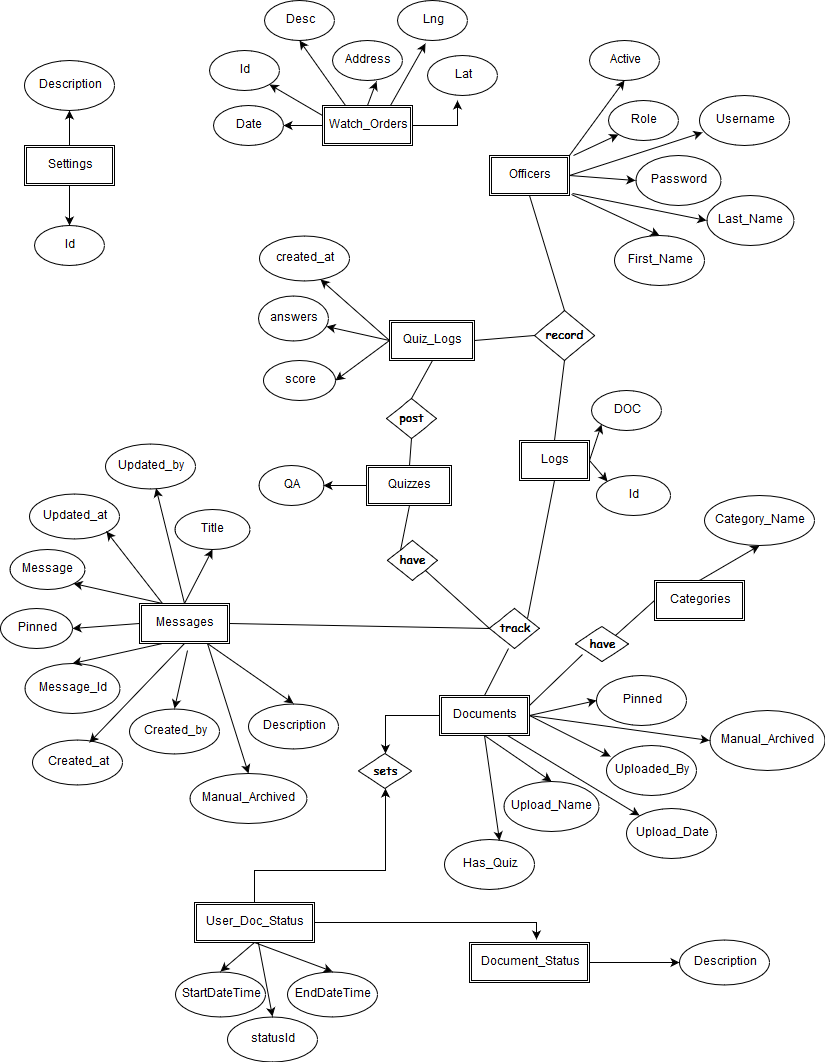
As the image shows, these components are separate, but interact to maintain the application state, and displays updates to the view. Our system uses Angular JS, which is a framework based on the Model-View-Controller architecture to achieve its functionality, and allows the application to be implemented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

### 4.1.2 Client-Server

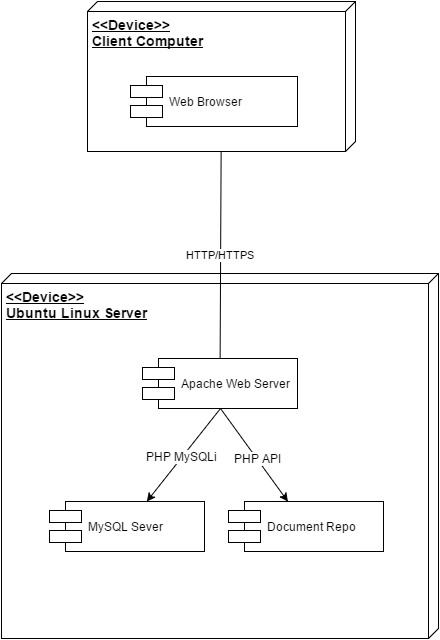


Our system uses the Client-Server model, which is an application structure used in distributed applications. As the image shows, the client and server communicate via the internet. The client makes requests to the server, and the server responds to the client with the requested function. This allows 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.

## 4.2 ER Model

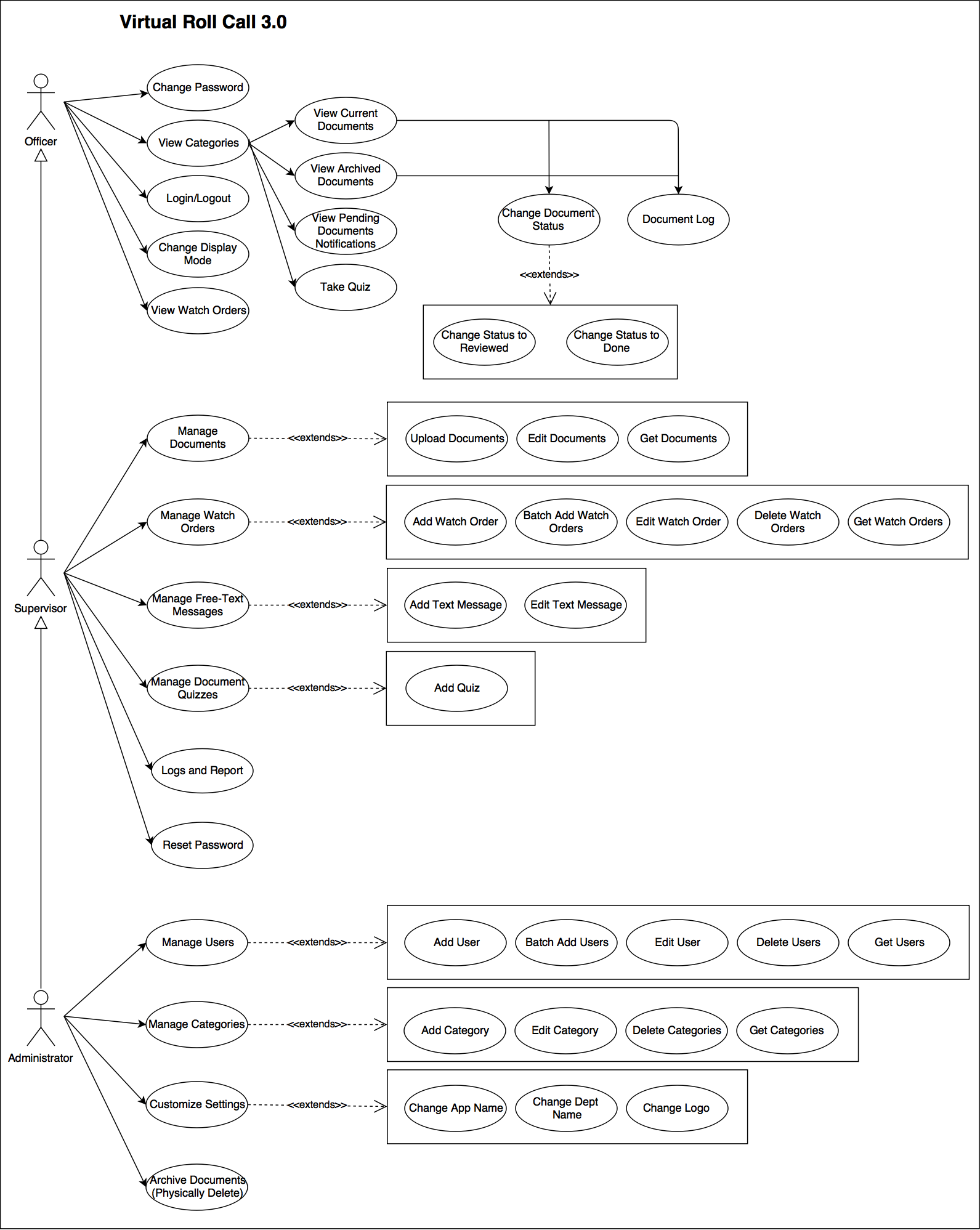


## 4.3 Deployment Diagram



# 

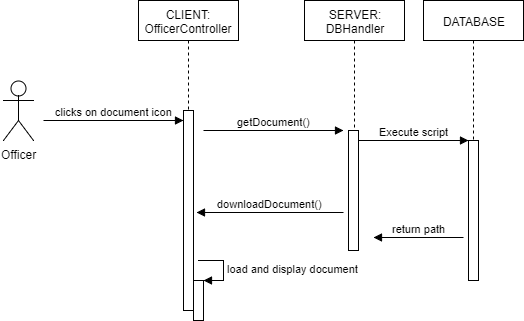
## 4.4 Use Case Diagram



**Fig 4.4 Virtual Roll Call Use Case Diagram**

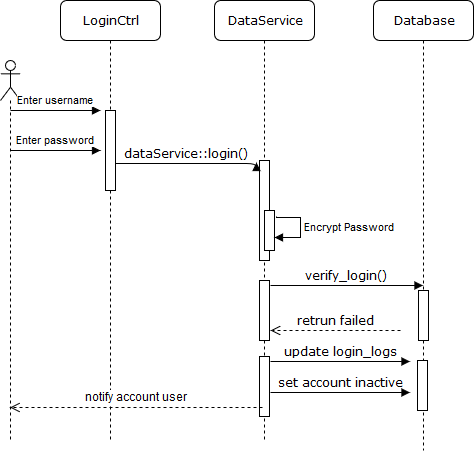
## 4.5 Object Interaction

In the section below, we will introduce the different interactions in our system using sequence diagrams. These will be the use cases that we will be implemented in our system.



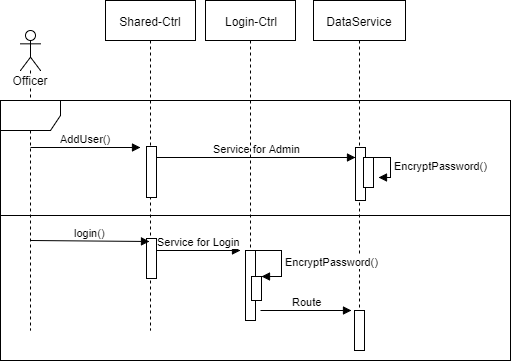
**Fig 4.5.1 View Document Diagram**

**Description:** This sequence diagram represents how an Officer views documents. This sequence diagrams begins when the Officer clicks on a document category. The System calls the getDocument() function to the database. The database returns the list of all documents (including name and path) and displays the list of documents along with their statuses (pending, reviewed or done). A Quiz link is visible for any document with quiz.



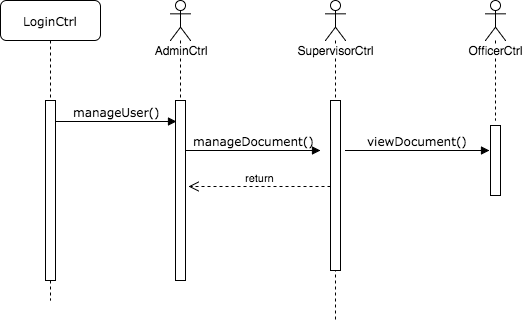
**Fig 4.5.2 Lock User after Five Consecutive Attempts Diagram**

**Description:** This sequence diagram represents how the system locks an officer account have five consecutive failed attempts to log in. This sequence diagrams begins when the Officer enters incorrect credentials on the fifth attempt. The system verifies the login credentials and returns a failed attempt. The system then updates the Login\_Logs table to increase the failed counter by 1. The system then updates the Officers table and set the user to inactive. The sequence ends when the system displays a warning message to the user that account has been blocked.



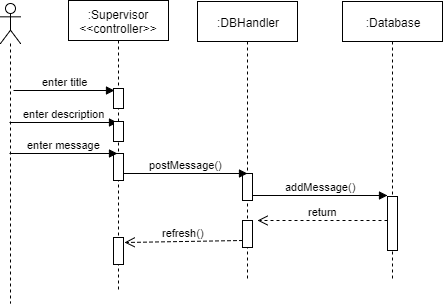
**Fig 4.5.3 Encrypt Passwords Diagram**

**Description:** This sequence diagram represents how the system encrypts password to make account more secured. This sequence diagram begins when a user logs in or a Supervisor adds a new user. The EncryptPassword() function is called to hash the password. At log in, the system grants access to the user when the encrypted password verifies the hash password saved for the respective account.

****

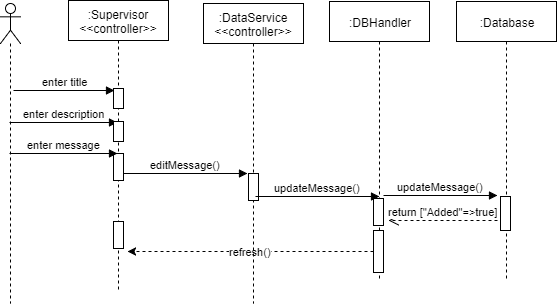
**Fig 4.5.4 Change Navigation Bar Based on User Role Diagram**

**Description:** This sequence diagram represents how the navigation bar changes based on user roles. The sequence diagram begins when the user logs into the system. If the user is an Administration, manageDocument() and viewDocument() can be invoked to switch role to Supervisor and Officer respectively. An Officer is restricted to an Officer role ONLY.

****

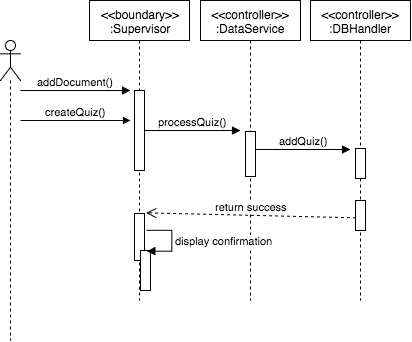
**Fig 4.5.5 Add Free-Text Message Diagram.**

**Description:** This sequence diagram represents how a Supervisor adds a new Free-Form Text Message. This sequence diagrams begins when the user enters the message details (title, description, message body). A option to pin the message is also available. When the form is submitted, the postMessage() function is invoked to the DBHandler controller; consequently the addMessage() function is invoked to add the details the to database. The system refreshes the page to update it with the newly added message.



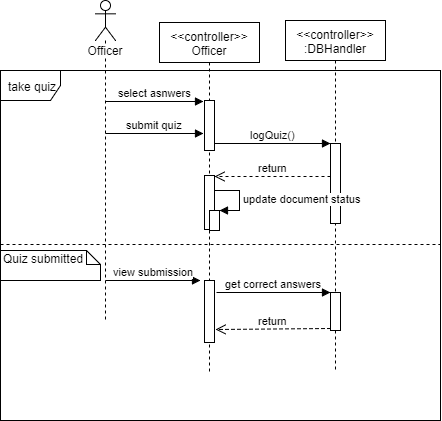
**Fig 4.5.6 Edit Free-Text Messages Diagram.**

**Description:** This sequence diagram represents how the user edits a free-form text message. The sequence diagram begins when the user clicks on the Edit button of the selected message. The user enters the edit details (updated title, description and body message). Upon submitting the form, the editMessage() and updateMessage() functions are called by the DataService and DBHandler controllers respectively. Once the message successfully updated in the database, the system refreshes the page to reflect the new updated message.



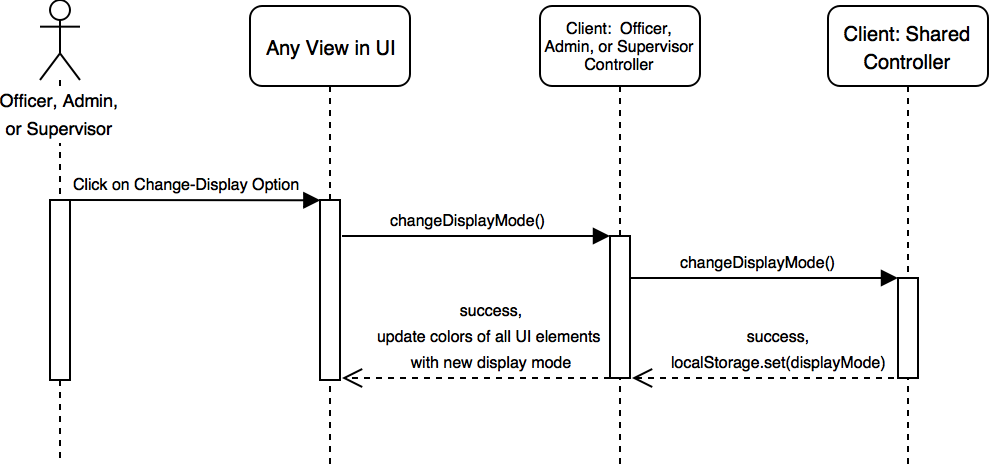
**Fig 4.5.7 Add Quiz to Document Diagram**

**Description:** This sequence diagram represents how a Supervisor add a quiz to a document. The sequence diagram begins when the user uploads a new document. The createQuiz() function is invoked the Open Quiz Maker is checked. Consequently the processQuiz() function is called to process and validate the quiz multiple choice questions and added to the database. A confirmation is displayed upon success.



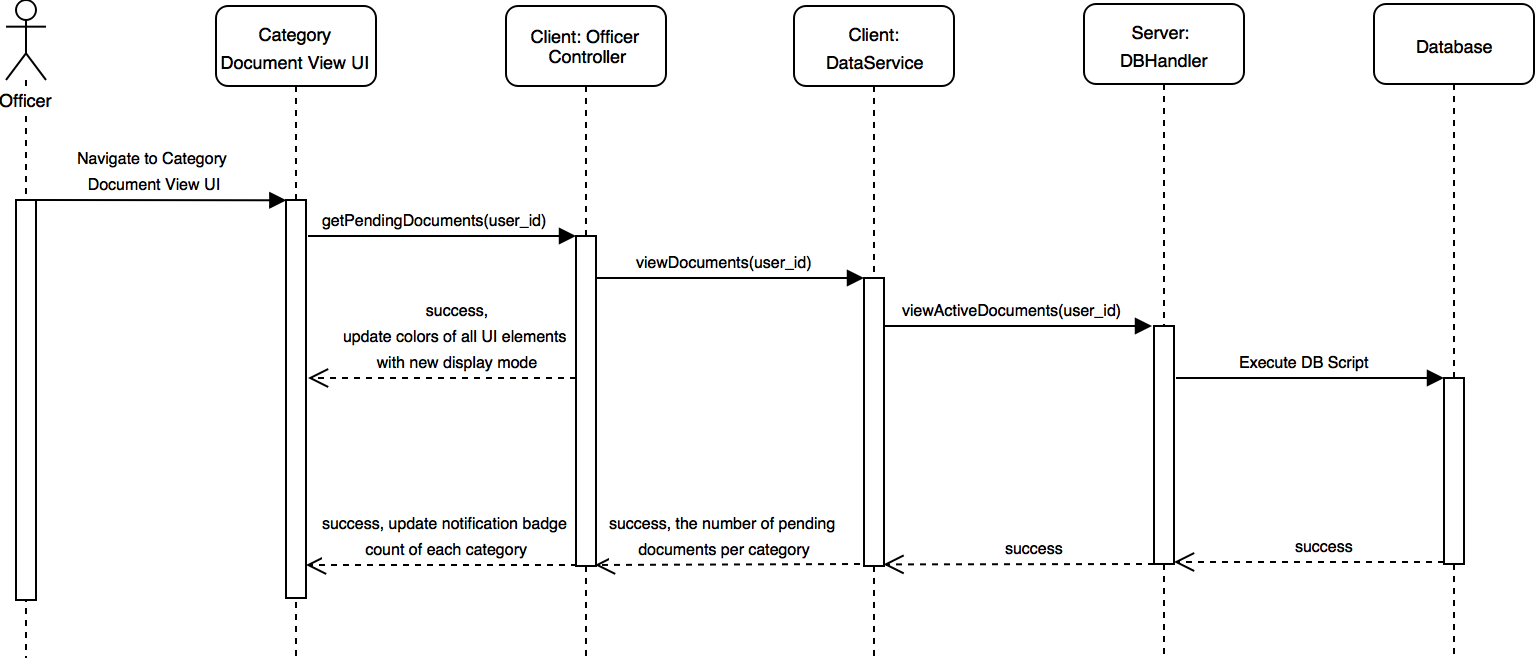
**Fig 4.5.8 Take Quiz Diagram**

**Description:** This sequence diagram represents how an officer takes a quiz. The sequence diagram begins when the user clicks on the Quiz link to start the quiz. The user select up to one answer per question and submits the quiz. The logQuiz() function is invoked to log the user’s answers to the database. Then, the documents status is updated to ‘Done’. If the quiz is already taken, the correct and submitted answers are displayed.

****

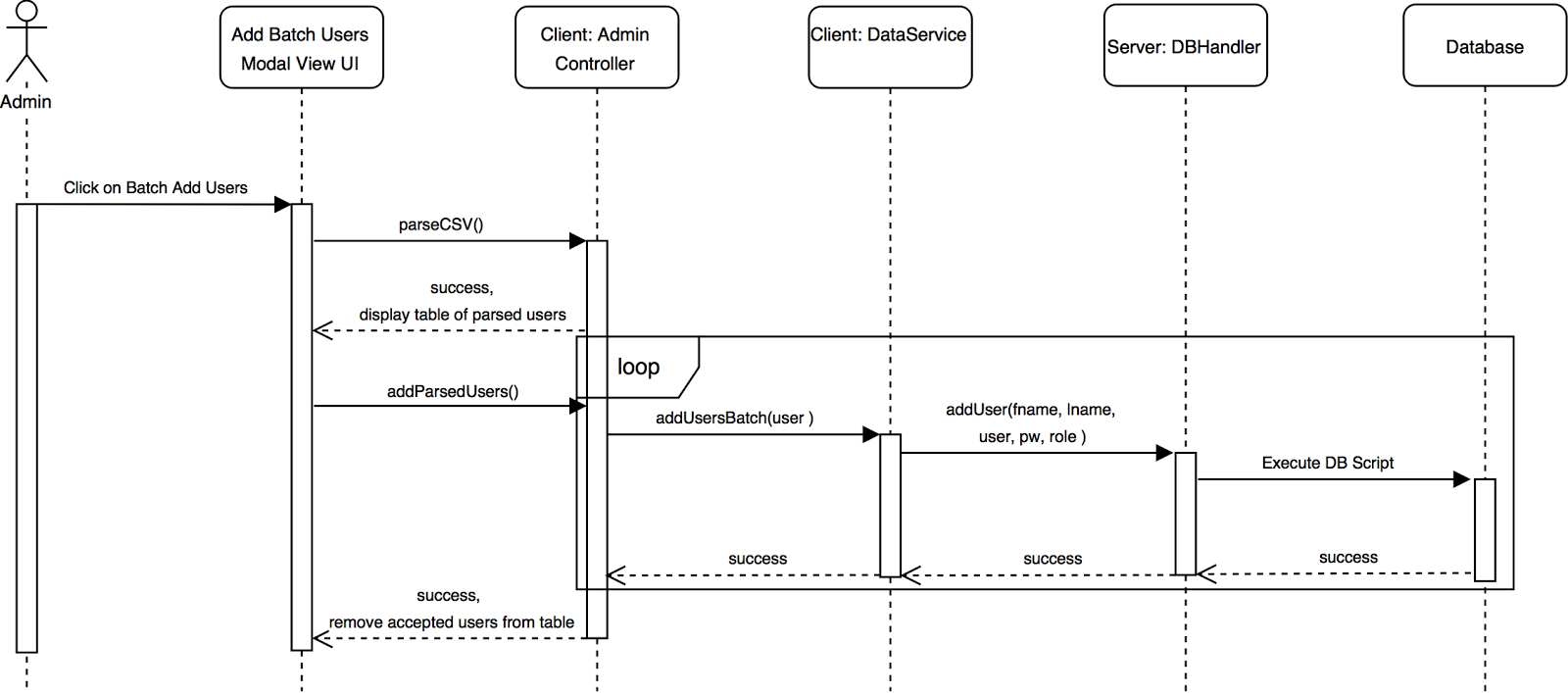
**Fig 4.5.9 Change Display Mode Diagram**

**Description:** This sequence diagram represents how an officer, supervisor, or admin changes the display mode. The sequence diagram begins when the user clicks on the change-display-mode option. The system identifies current display mode and changes it to the opposite display mode, going from day-mode to night-mode, or vice versa. The colors of all UI elements are reversed, going from light to dark, or vice versa. The text in the change-display-mode option is modified to include the previous display-mode (“Change to Day-Mode” or “Change to Night-Mode”)

****

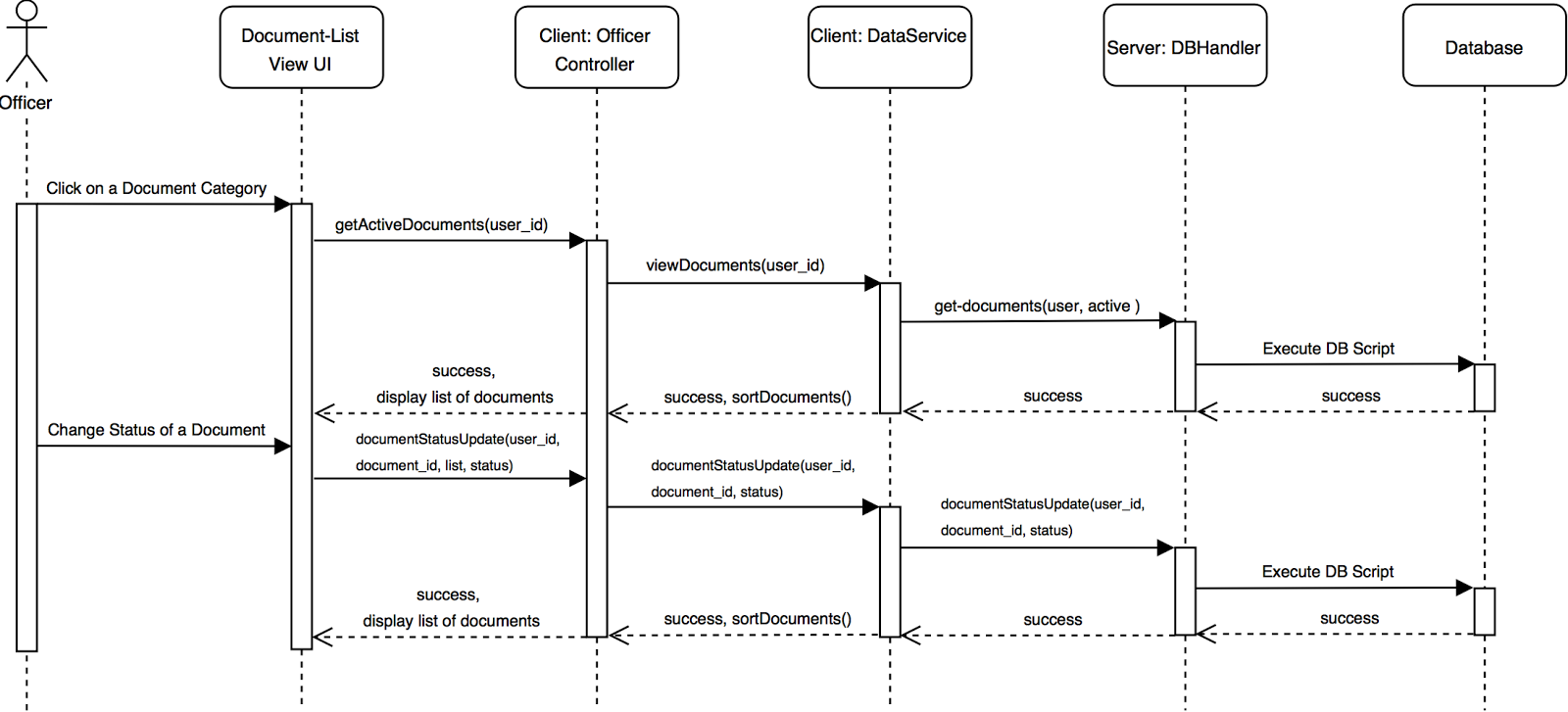
**Fig 4.5.10 View Pending Documents Notification Diagram**

**Description:** This sequence diagram represents how an officer views pending document notifications. The sequence diagram begins when the user navigates to the main category document view page. The system counts the number of pending documents per category, and it updates the notification badge UI of each category with this count. If there are no pending documents in a category, no badge is displayed.

****

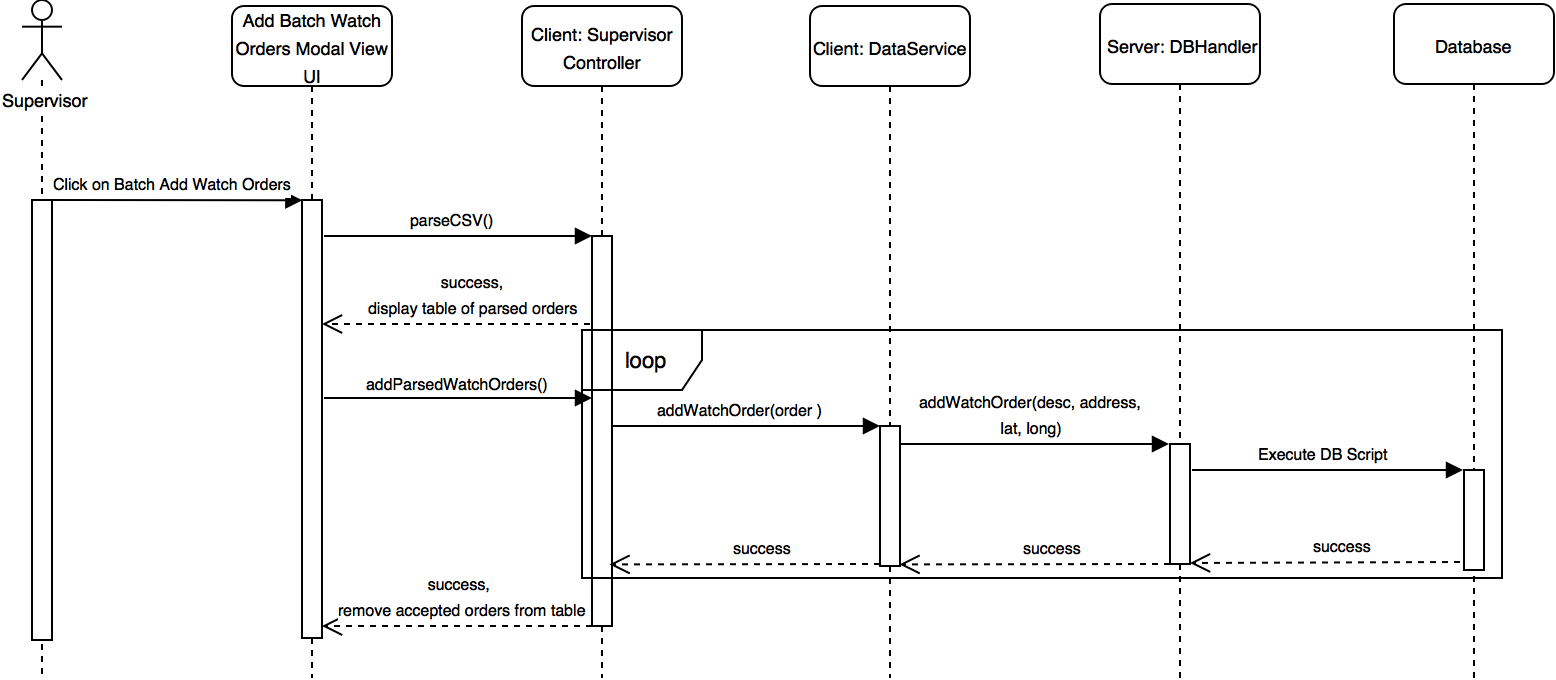
**Fig 4.5.11 Add Users with CSV File Diagram**

**Description:** This sequence diagram represents how an admin adds new users with a CSV file. The sequence diagram begins when the user clicks on batch-add-users option.

The user selects a CSV file to upload from the local computer and clicks ‘Parse CSV’ button. The system converts all rows in the CSV file into a table of users that are displayed in the current window. After the user clicks ‘Add All’ button, the system attempts to add all users in the table to the database of users.

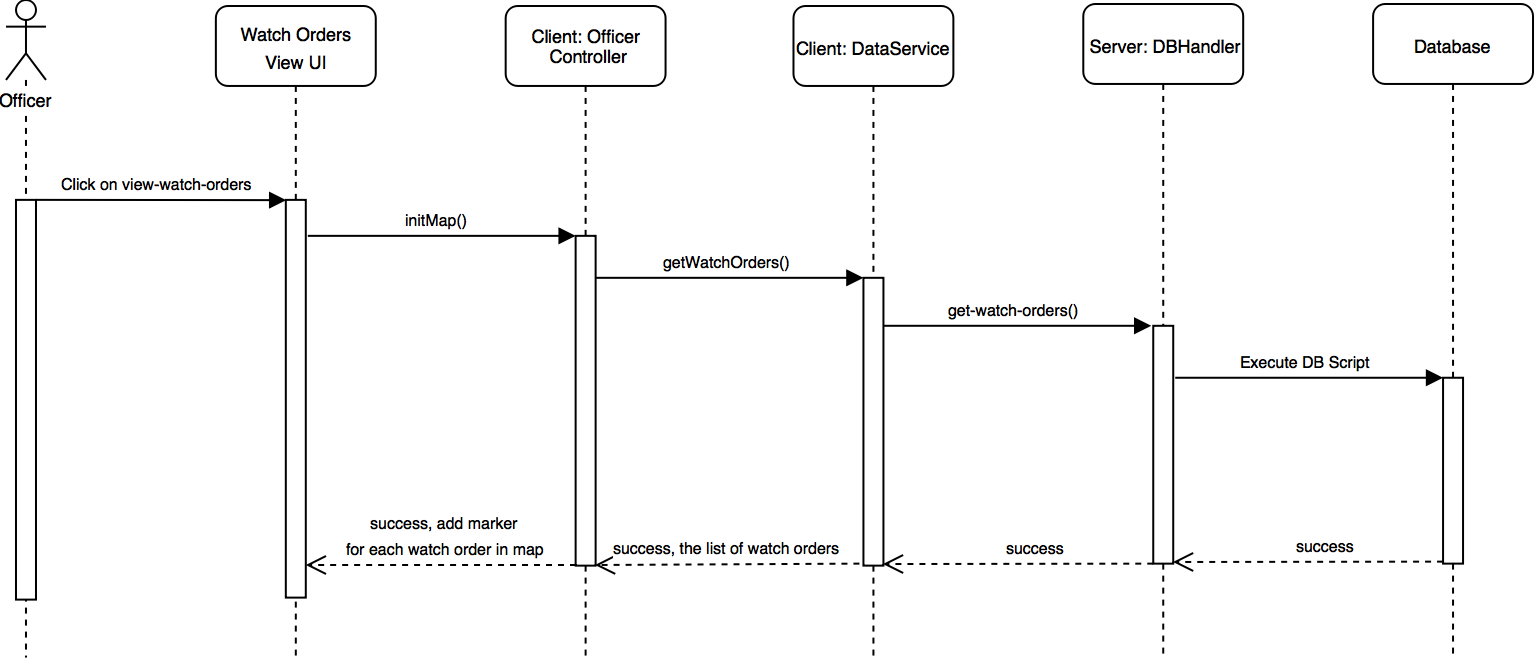
**Fig 4.5.12 Improve Document List Presentation Diagram**

**Description:** This sequence diagram represents how an officer interacts with the system with improved document list presentation. The sequence diagram begins when the user clicks on a document category. The system opens the document list view for the selected category, and it displays all active documents, ordered by their status and with the status label color corresponding to each document’s status. After the user, changes the status of a document, the system displays all active documents, changing the order and the label color of the modified documents according to their new status.

****

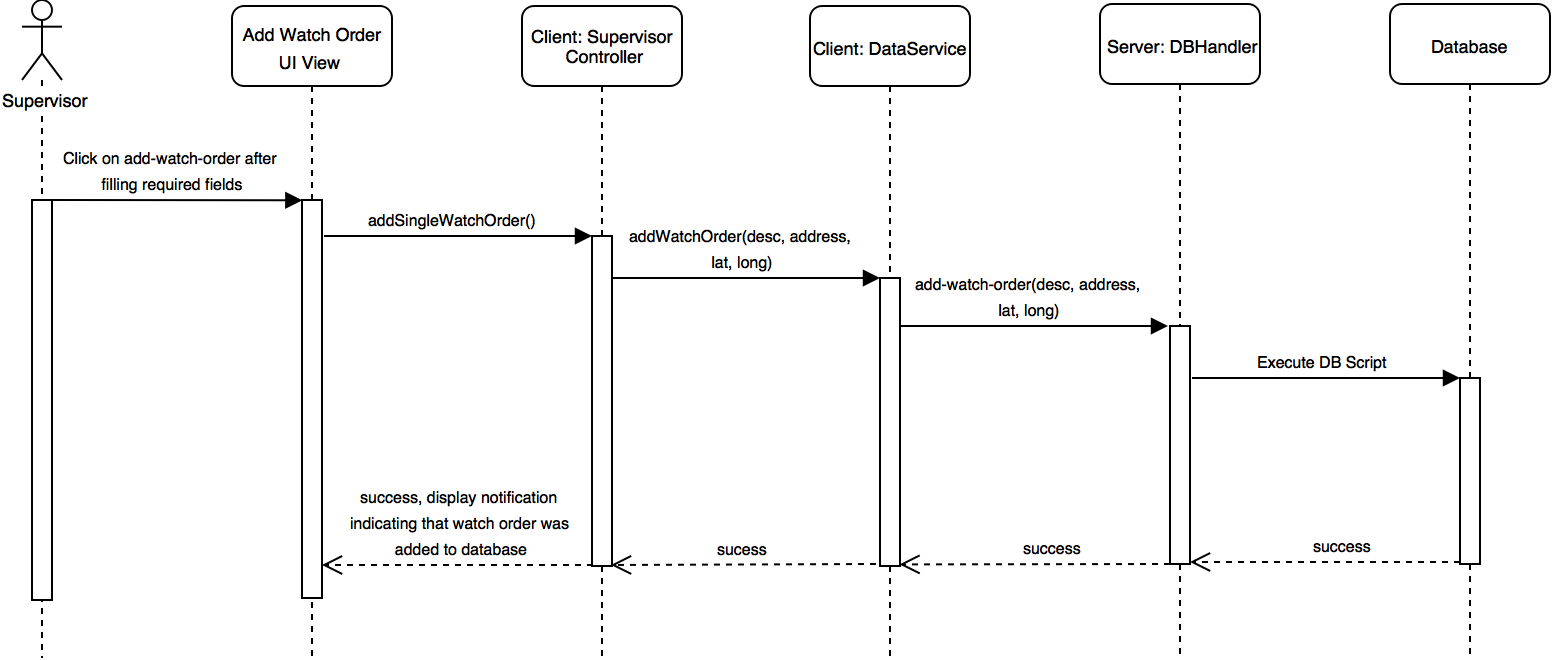
**Fig 4.5.13 Add Watch Orders with CSV File Diagram**

**Description:** This sequence diagram represents how a supervisor adds watch orders with a CSV file. The sequence diagram begins when the user clicks on the batch-watch-orders option. The user selects a CSV file to upload from the local computer and clicks ‘Parse CSV’ button. The system converts all rows in the CSV file into a table of watch orders that are displayed in the current window. After the user clicks ‘Add All’ button, the system attempts to add all watch orders in the table to the database of watch orders.

****

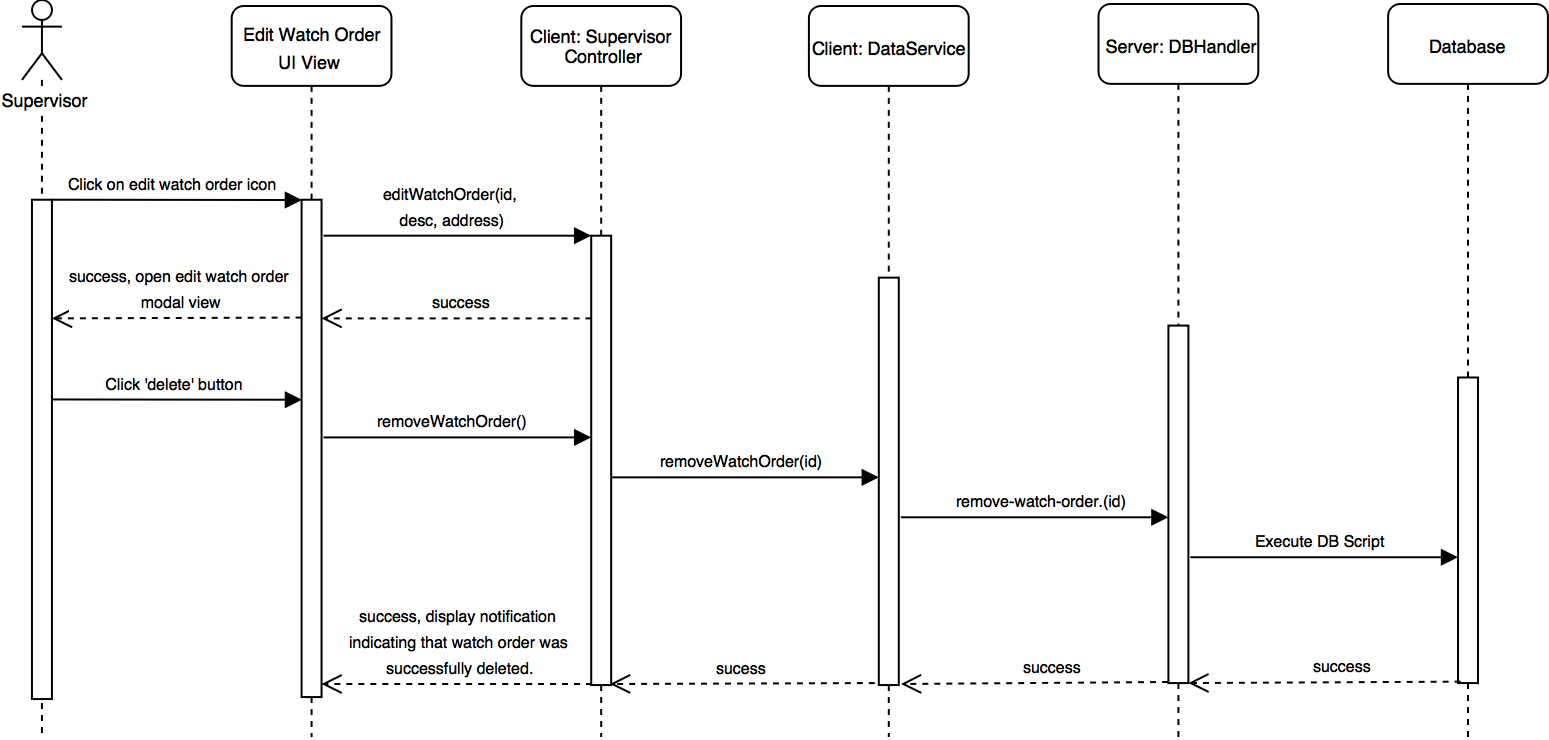
**Fig 4.5.14 View Watch Orders in a Map Diagram**

**Description:** This sequence diagram represents how an officer views watch orders on a map. The sequence diagram begins when the user clicks on the view-watch-orders option. The system opens the watch-orders-view and loads the map with watch order markers. If the user clicks on any watch order marker, the system displays an information window with address, description, and date for selected watch order.

****

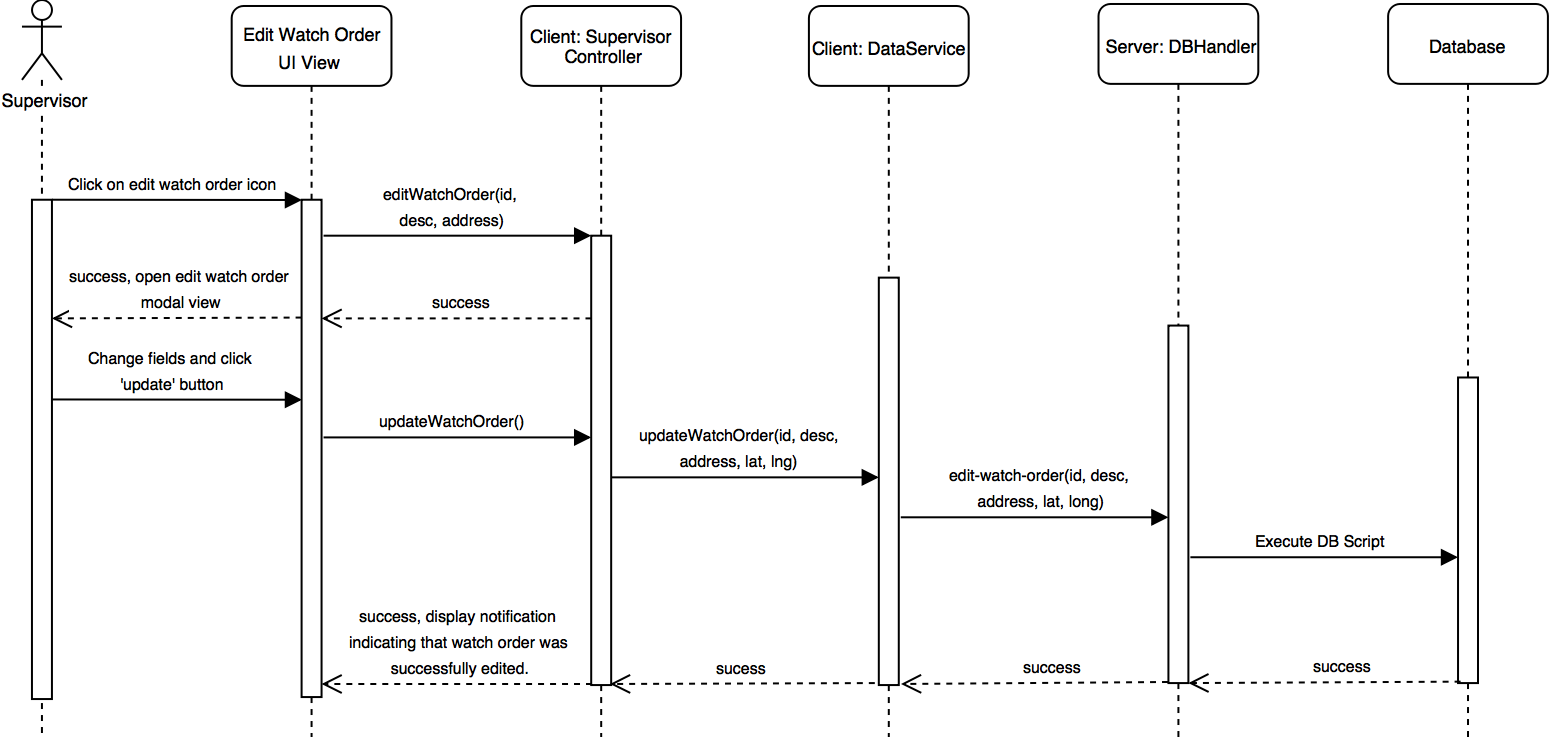
**Fig 4.5.15 Add Watch Order Diagram**

**Description:** This sequence diagram represents how a supervisor adds a watch order. The sequence diagram begins when the user clicks on the add-watch-orders option. The user must fill in the required fields and click the ‘Add Watch Order’ button. The system attempts to add a new watch order to database and displays a notification indicating that the watch order was successfully added.

****

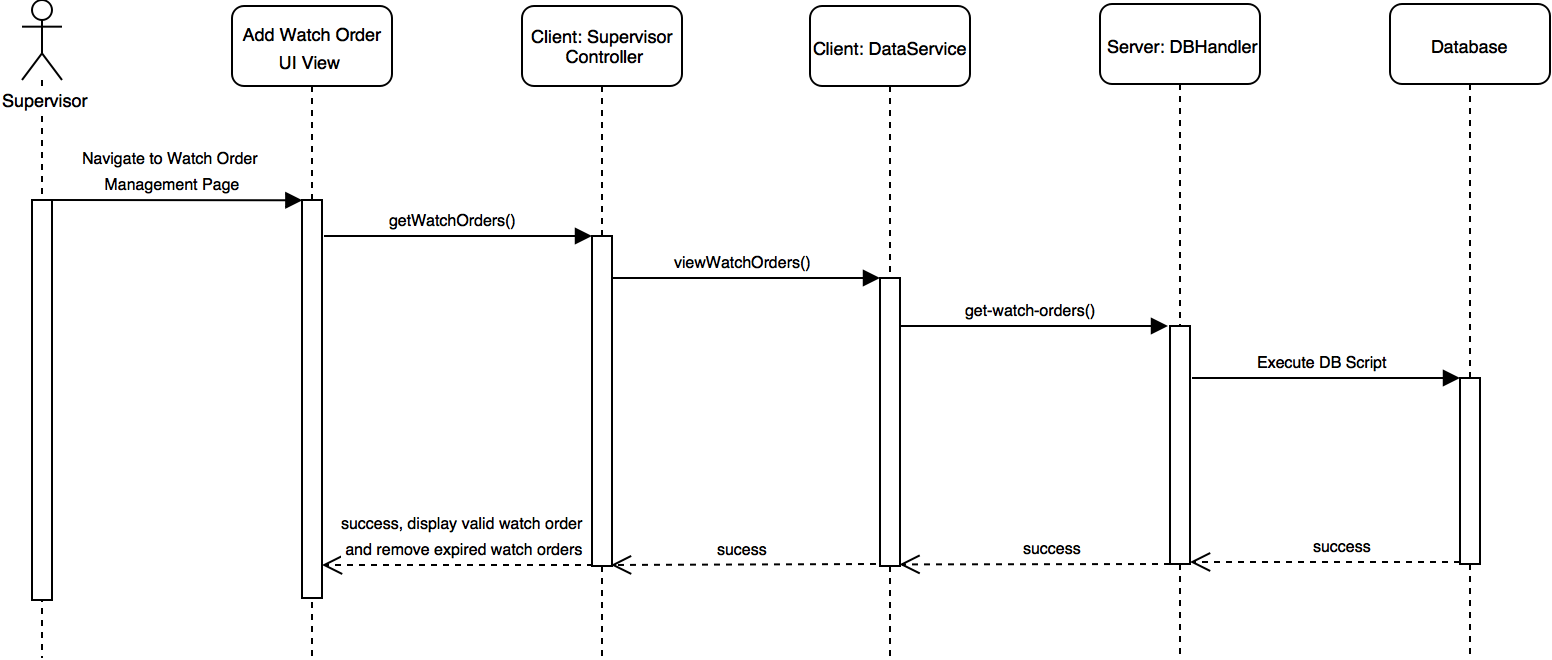
**Fig 4.5.16 Delete Watch Order Diagram**

**Description:** This sequence diagram represents how a supervisor deletes a watch order. The sequence diagram begins when the user clicks on the edit-watch-orders option. The user must select the delete option. The system attempts to remove the watch order from database, and it displays a success message indicating that watch order was removed.

****

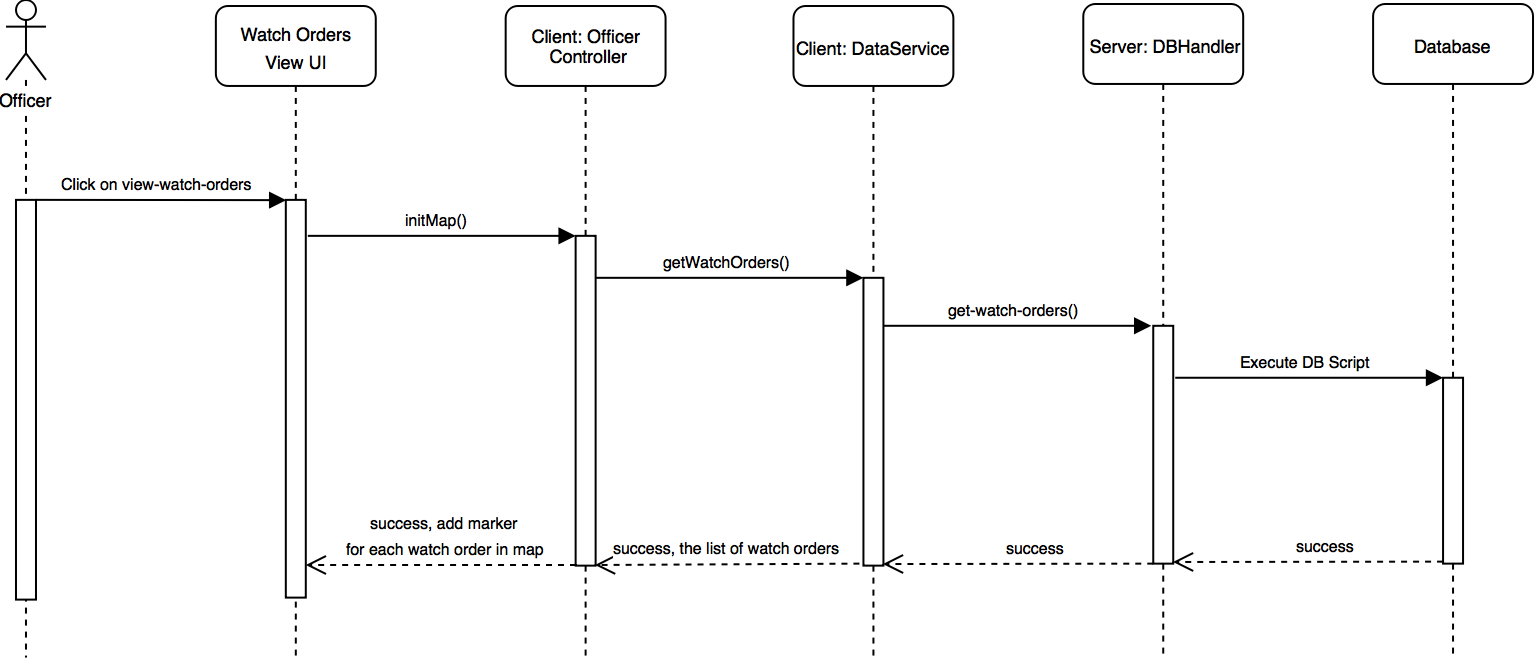
**Fig 4.5.17 Edit Watch Order Diagram**

**Description:** This sequence diagram represents how a supervisor edits a watch order. The sequence diagram begins when the user clicks on the edit-watch-orders option. The user updates the target fields with new values for the selected watch order and selects the update option. The system attempts to update the selected watch order with new field values, and it displays a success message indicating that the watch order was edited.

****

**Fig 4.5.18 Automatically Delete Watch Orders Diagram**

**Description:** This sequence diagram represents how watch orders are automatically deleted. The sequence diagram begins when the user navigates to the watch order management page. The system retrieves all watch orders from the database, checks the expiration date of all watch orders, and removes all expired watch orders from database.

** Fig 4.5.19 Display Watch Order List Table Diagram**

**Description:** This sequence diagram represents how an officer views the watch order list table. The sequence diagram begins when the user clicks on view-watch-order. The system retrieves all watch orders from the database and displays all watch orders in a list table and in the watch order map.

# SYSTEM VALIDATION AND TESTING PROCESS

This section details the unit test cases that were used to validate the features completed during each sprint. Each user story has one or more unit tests, as well as an integration test to ensure the functionality of previous implementations remains intact.

## 5.1 Version 1.0

**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** | Pass: 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 |
| **Pre-Conditions** | 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 admin is notified |

**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. Admin enters a name that already exists in the category field 2. The admin clicks the ‘Add’ button |
| **Expected Output** | Category name is not added to the database and admin is notified |

**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 to edit 2. The admin enters a valid name in the modal category field and hits ‘Update’ |
| **Expected Output** | Category name is updated to the database and admin is notified |

**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. Supervisor logs in and selects 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 to edit 2. 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 to edit 2. The admin doesn’t enter anything in the category field 3. The admin clicks the ‘Update’ button |
| **Expected Output** | Category name is not updated on the database and 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 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 are removed from System |
| **Pre-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 |
| **Pre-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 |
| **Pre-Condition** | User role is Officer |
| **Expected Output** | 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). 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 login page is updated when an admin uploads a new logo. |
| **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). 3. The admin previews the image they just selected 4. The admin clicks on ‘Save’ to replace the logo on the site |
| **Expected Output** | 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 open or download document in 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 Output** | 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 Output** | 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 of 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 document icons with documents name of selected category and unpinned by Supervisor upon uploading to system |

## 5.2 Version 2.0

**User Story #233: Add Document Confirmation (Release 2.0)**

**Sunny Day:**

|  |  |
| --- | --- |
| **Purpose** | Verify that document status change from default status “Pending” into “Reviewed” after officer clicks on document. for the first time. |
| **Entrance Condition** | 1. User is logged in as Officer role.  2. User selects category they wish to filter documents by. |
| **Input** | The category will list the documents. Documents pending to review for this officer will have status “Pending”. Officer will click on document for the first time. |
| **Expected Output** | Document will display and it will change its status on the list from “Pending” to “Reviewed”. |

**Rainy Day**

|  |  |
| --- | --- |
| **Purpose** | Verify only documents with status “Reviewed” can be changed to status “Done” by clicking on disclaimer checkmark. |
| **Entrance Condition** | 1. User is logged in as Officer role.  2. User selects category they wish to filter documents by. |
| **Input** | Officer attempts to click on the disclaimer checkmark for a document that has status “Pending”. |
| **Expected Output** | The system will not allow to click on the disclaimer check mark because the status is “Pending”. It will only allow to click on the check mark for documents with status “Reviewed”. |

**Integration Test**

|  |  |
| --- | --- |
| **Purpose** | Verify documents with status “Reviewed” can be changed to status “Done” by clicking on disclaimer checkmark. |
| **Entrance Condition** | 1. User is logged in as Officer role  2. User selects category they wish to filter documents by |
| **Input** | Officer clicks on the disclaimer check mark for a document with status “Reviewed”. |
| **Expected Output** | Document will change from status “Reviewed” to status “Done”, the disclaimer check mark will show as checked and will get disable so that it cannot be uncheck. |

**User Story #244: Add User Logs (Release 2.0)**

**Sunny Day:**

|  |  |
| --- | --- |
| **Purpose** | Verify that all the trace of the officer is recorded in the database, every time the officer click in a document. |
| **Entrance Condition** | 1. User is logged in as Supervisor role.  2. User selects logs and filter by the condition they desire. |
| **Input** | The Supervisor enter the word he/she desire to filter, it can be a document name, an officer or a date. |
| **Expected Output** | The Logs screen will show the Logs of the document(s) that the Officer just click. |

**User Story #245: Archive Document (Release 2.0)**

**Sunny Day:**

|  |  |
| --- | --- |
| **Purpose** | Delete all the documents from a determine date range |
| **Entrance Condition** | 1. User is logged in as Administrator role.  2. User selects Archive Document |
| **Input** | Administrator enters the range date he/she wish to delete |
| **Expected Output** | The Screen will show the friendly name and the file system name of the file that have been deleted. |

**Rainy Day**

|  |  |
| --- | --- |
| **Purpose** | Delete all the documents from a determine date range |
| **Entrance Condition** | 1. User is logged in as Administrator role.  2. User selects Archive Document |
| **Input** | The Administrator enters the range date he/she wish to delete |
| **Expected Output** | The system will not show any file, so no file was entered in that time range. |

**User Story #249: View Document in App (Release 2.0)**

**Sunny Day:**

|  |  |
| --- | --- |
| **Purpose** | Show The document in the App |
| **Entrance Condition** | 1. User is logged in as Officer role.  2. User selects a document |
| **Input** | The Officer click on a document. |
| **Expected Output** | The browser open a new tab with the content of the document |

**Rainy Day**

|  |  |
| --- | --- |
| **Purpose** | Show The document in the App |
| **Entrance Condition** | 1. User is logged in as Officer role.  2. User selects a document |
| **Input** | The Officer click on a document. |
| **Expected Output** | The browser does not support the document for opening and give the Officer the option to open the document with a specific application or to download the document. |

## 5.3 Version 3.0

**User Story #295: Display Document within Window (Release 3.0)**

|  |  |
| --- | --- |
| **Test Case ID** | VRC-3002-VD-001 |
| **Purpose** | Verify that an officer can view a PDF document. |
| **Preconditions** | Officer is logged in.  User navigates to main Document Category View page. |
| **Input** | The officer clicks on the document icon labeled Resume |
| **Expected Output** | A modal window shows the Resume PDF document |
| **Actual Output** | A 13-page resume is uploaded into the modal popup frame. |
| **Status** | Pass |

|  |  |
| --- | --- |
| **Test Case ID** | VRC-3002-VD-002 |
| **Purpose** | Verify that an officer can view an image document. |
| **Preconditions** | Officer is logged in. and selected the ‘BOLO’ documents category. |
| **Input** | The officer clicks on the document icon labeled face. |
| **Expected Output** | A window opens and shows an image. |
| **Actual Output** | A modal windows opens and loads the face.png image for display. |
| **Status** | Pass |

|  |  |
| --- | --- |
| **Test Case ID** | VRC-3002-VD-003 |
| **Purpose** | Assert that no document is shown for a missing file. |
| **Preconditions** | Supervisor and Officer are logged in.  The officer has selected the documents category. |
| **Input** | The officer clicks on the document icon. |
| **Expected Output** | A blank window shows up. |
| **Actual Output** | A modal window pops up with a broken image link. |
| **Status** | Pass |

**User Story #147: Lock User after Consecutive Failed Attempts (Release 3.0)**

|  |  |
| --- | --- |
| **Test Case ID** | VRC-3003 -Lock-147001 |
| **Purpose** | Verify that the Login\_Logs relation is updated whenever an Officer’s login attempt fails. |
| **Preconditions** | Officer has launched the application.  The system is ready for input. |
| **Input** | The officer enters username: ‘officer’  The officer enters an invalid password: ‘rest’ |
| **Expected Output** | ~ A message alert that invalid credentials were entered.  ~ A log has been updated with count increase of 1. |
| **Actual Output** | "Warning: Invalid credentials: Please try again!" |
| **Status** | Pass |

|  |  |
| --- | --- |
| **Test Case ID** | VRC-3003 -Lock-147002 |
| **Purpose** | Verify that an account is locked after the fifth attempt. |
| **Preconditions** | Officer has launched the application  The system is ready for input.  The officer has entered the wrong password four times already. |
| **Input** | The officer enters username: ‘officer’  The officer enters an invalid password: ‘rest’ |
| **Expected Output** | A message alert that account has been locked. |
| **Actual Output** | "This account has been locked: Please contact your Supervisor!" |
| **Status** | Pass |

**User Story #250: Encrypt Passwords** **(Release 3.0)**

|  |  |
| --- | --- |
| **Test Case ID** | VRC-3003 -Encrypt-250001 |
| **Purpose** | Verify that an account password is encrypted when new user is added. |
| **Preconditions** | An Administrator has successfully logged into the system.  The user select the User Management tab. |
| **Input** | The Administrator enters the new user information:  ~ First Name: Terrell  ~ Last Name: Wheaton  ~ Username: twheaton  ~ Password: test1234  ~ Role: Officer |
| **Expected Output** | ~ A new user has been added.  ~ Password is encrypted in the database |
| **Actual Output** | "User successfully added!" |
| **Status** | Pass |

|  |  |
| --- | --- |
| **Test Case ID** | VRC-3003 -Encrypt-250002 |
| **Purpose** | Verify that a user can access account with an encrypted password. |
| **Preconditions** | A user has launched the system and the system awaits user inputs |
| **Input** | The user enters credentials:  ~ Username: twheaton  ~ Password: test1234 |
| **Expected Output** | The system displays the dashboard. |
| **Actual Output** | The system displays document categories. |
| **Status** | Pass |

**User Story #302: Change Navigation Bar Based on User Role** **(Release 3.0)**

|  |  |
| --- | --- |
| **Test Case ID** | VRC-3004 -NavigateSupervisor-302001 |
| **Purpose** | Verify that a user can switch between an Administrator and Supervisor roles. |
| **Preconditions** | Officer has launched the application  The user has successfully logged in as an Administrator.  The system displays the User Management window. |
| **Input** | The user hits on the “Supervisor” tab. |
| **Expected Output** | The user is redirected to the Supervisor controller.  The system allows the user to hit the “Administrator” tab. |
| **Actual Output** | The system displays the “Manage Documents” window.  The system shows the “Administrator” tab to return to Administrator Controller. |
| **Status** | Pass |

|  |  |
| --- | --- |
| **Test Case ID** | VRC-3004 -NavigateSupervisor-302002 |
| **Purpose** | Verify a user can switch between a Supervisor and Officer roles. |
| **Preconditions** | Officer has launched the application  The user has successfully logged in as a Supervisor.  The system displays the “Manage Documents” window. |
| **Input** | The user hits on the “Officer” tab. |
| **Expected Output** | The user is redirected to the Officer Controller.  The system allows the user to hit the “Supervisor” tab. |
| **Actual Output** | The system displays the “Document Categories” window.  The system shows the “Supervisor” tab to return to Supervisor Controller |
| **Status** | Pass |

|  |  |
| --- | --- |
| **Test Case ID** | VRC-3004 -NavigateSupervisor-302003 |
| **Purpose** | Verify that an officer can only have the Officer Controller view. |
| **Preconditions** | Officer has launched the application  The user has successfully logged in as an Officer.  The system displays the Document Categories window. |
| **Input** | The user has one available tab. |
| **Expected Output** | The user has only the Officer controller for view. |
| **Actual Output** | The system displays the “Document Categories” window.  The user can only view the Officer Controller. |
| **Status** | Pass |

**User Story #248: Add Free-Text Messages** **(Release 3.0)**

|  |  |
| --- | --- |
| **Test Case ID** | VRC-3005 -AddFreeText-248001 |
| **Purpose** | Share free-text (plain-text) message. |
| **Preconditions** | An Administrator has successfully logged into the system.  The user select the Manage Free Message tab. |
| **Input** | The user enters the new user information:  ~ Title: Sample Title  ~ Message Description: A very short detail  ~ Message: “All the best messages for boss are sent by employees to express how they feel about the management and if this benefits everyone who have signed the terms of agreement and policies of a company, most importantly wishing your boss all the best in the company’s progress. |
| **Expected Output** | Message is saved to the database. |
| **Actual Output** | Message title, description and posted dated are added to Edit Message list. |
| **Status** | Pass |

**User Story #308: Edit Free-Text Messages** **(Release 3.0)**

|  |  |
| --- | --- |
| **Test Case ID** | VRC-3005 -EditFreeText-308001 |
| **Purpose** | Edit free-text (plain-text) message. |
| **Preconditions** | An Administrator has successfully logged into the system.  The user selects the Manage Free Message tab.  The user clicks on ‘Edit’ button |
| **Input** | The user enters the updated information:  ~ Title: Sample Title Edited  ~ Message Description: A very short detail Edited  ~ Message: This message has been modified. |
| **Expected Output** | Message is saved to the database. |
| **Actual Output** | New message title, description and posted dated are added to Edit Message list. |
| **Status** | Pass |

|  |  |
| --- | --- |
| **Test Case ID** | VRC-3005 -EditFreeText-308002 |
| **Purpose** | Delete free-text (plain-text) message from the list. |
| **Preconditions** | An Administrator has successfully logged into the system.  The user selects the Manage Free Message tab.  The user clicks on ‘Edit’ button |
| **Input** | The user clicks on the ‘Delete Message’ button. |
| **Expected Output** | The message is removed from the list. |
| **Actual Output** | Edit Messages list has been updated with message #3 removed. |
| **Status** | Pass |

**User Story #243: Add Document Quiz** **(Release 3.0)**

|  |  |
| --- | --- |
| **Test Case ID** | VRC-3006 -AddQuiz-243001 |
| **Purpose** | Create a quiz for a new uploaded document. |
| **Preconditions** | A Supervisor has successfully logged into the system.  User visits Manage Documents to upload a new document.  The user has entered document details.  The user clicks on Open Quiz Maker to open the quiz window. |
| **Input** | The user clicks on Open Quiz Maker to open:  ~ Question 1: When is the Village Address?  ~ Correct Answer: 12/06/2017  ~ Answer 2: 12/01/2017 7:00 PM  ~ Answer 3: 12/16/2017 7:00 PM  ~ Answer 4: 12/06/2017 7:00 PM  The user clicks on Save Questionnaire button |
| **Expected Output** | ~ Message is saved and added to the document.  ~ The modal window closes |
| **Actual Output** | ~ Question and answers are saved in memory.  ~ The Document Quiz Maker window is closed. |
| **Status** | Pass |

**User Story #312: Take Quiz** **(Release 3.0)**

|  |  |
| --- | --- |
| **Test Case ID** | VRC-3006 -TakeQuiz-312001 |
| **Purpose** | Take quiz to acknowledge reviewing document. |
| **Preconditions** | An Officer has successfully logged into the system.  The user has selected the Message From Chief category.  The user clicks on the Village Calendar ‘Quiz’ link to activate it. |
| **Input** | The user selects the following choices  ~ Question 1: 12/01/2017 7:00 PM  ~ Question 2: The People  ~ Question 3: 9000 Blue Road  ~ Question 4: Finance  The user clicks on ‘Submit Quiz’ button. |
| **Expected Output** | ~ The user answers are submitted and saved to the database.  ~ The modal window closes and the document status updated. |
| **Actual Output** | ~ Quiz answered are logged to the database.  ~ Document status updated to Done. |
| **Status** | Pass |

|  |  |
| --- | --- |
| **Test Case ID** | VRC-3006 -TakeQuiz-312002 |
| **Purpose** | Disable quiz for pending documents. |
| **Preconditions** | An Officer has successfully logged into the system.  The user has selected the Message From Chief category.  The document is pending for review. |
| **Input** | The user clicks on the Village Calendar ‘Quiz’ link to activate it. |
| **Expected Output** | The system displays a warning message. |
| **Actual Output** | A warning dialog box displays:  “Quiz available Only after reviewing document!” |
| **Status** | Pass |

|  |  |
| --- | --- |
| **Test Case ID** | VRC-3006 -TakeQuiz-312002 |
| **Purpose** | View Quiz correct answers. |
| **Preconditions** | An Officer has successfully logged into the system.  The user has selected the Message From Chief category.  The document is pending for review. |
| **Input** | The user clicks on the Village Calendar ‘Quiz’ link to activate it. |
| **Expected Output** | System displays quiz with user answers and the correct answers. |
| **Actual Output** | The system displays the submitted quiz and score. |
| **Status** | Pass |

**User Story #242: Change Display Mode** **(Release 3.0)**

|  |  |
| --- | --- |
| **Test Case ID** | 242-U1 |
| **Purpose** | Verify that current display mode setting in browser local storage is updated with alternate display mode. |
| **Preconditions** | User is logged in. |
| **Input** | User selects option to toggle display mode in navigation bar. |
| **Expected Output** | Display mode in local storage is updated with alternate display mode:   * if current display mode was day-mode, update with night-mode preference * if current display mode was night-mode, update with day-mode preference |
| **Actual Output** | Display mode in local storage is updated with correct display mode |
| **Status** | Pass |

|  |  |
| --- | --- |
| **Test Case ID** | 242-I1 |
| **Purpose** | Verify that all UI elements on page are updated with corresponding colors and text for the current display mode |
| **Preconditions** | User is logged in. |
| **Input** | User selects option to toggle display mode in navigation bar. |
| **Expected Output** | All UI elements on page will change when the display mode is modified:   * If night-mode is activated, all UI elements will have dark theme. * If day-mode is activated, all UI elements will have light theme   The text in the change-display-mode option is modified to include the previous display-mode:   * If night-mode is activated, option will read: “Change to Day-mode” * If day-mode is activated, option will read: “Change to Night-mode” |
| **Actual Output** | The properties of all UI elements including change-display-mode option correspond to current display mode |
| **Status** | Pass |

|  |  |
| --- | --- |
| **Test Case ID** | 242-I2 |
| **Purpose** | Verify that display mode is persistent throughout all pages. |
| **Preconditions** | User is logged in. |
| **Input** | 1. User selects option to toggle display mode in navigation bar. 2. Users navigates to any another page. |
| **Expected Output** | The properties of all UI elements including the change-display-mode option in the new page correspond to the new display mode. |
| **Actual Output** | All UI elements in new page correspond to the new display mode. |
| **Status** | Pass |

**User Story #296: View Pending Documents Notification (Release 3.0)**

|  |  |
| --- | --- |
| **Test Case ID** | 296-U1 |
| **Purpose** | Verify that correct pending documents count is displayed in notification badge for each category |
| **Preconditions** | Officer is logged in. |
| **Input** | User navigates to main Document Category View page. |
| **Expected Output** | The number displayed in the notification badge of each category reflects the actual amount of documents with “Pending” status in the category. If there are no pending documents in the category, no notification badge will be displayed. |
| **Actual Output** | The correct pending documents number is displayed for each category. If the number of pending document is zero, no notification badge is displayed for that category. |
| **Status** | Pass |
| **Test Case ID** | 296-I1 |
| **Purpose** | Verify that the number displayed in the pending documents notification badge is properly updated after the user opens pending documents. |
| **Preconditions** | Officer is logged in. |
| **Input** | 1. User navigates to main Document Category View page. 2. User opens any category with pending documents. 3. User opens one or more documents with “Pending” status. 4. User returns to main Document Category View page. |
| **Expected Output** | The pending documents count in the notification badge for the specific category is updated (decremented) to reflect the new number of pending documents. If the pending documents count is now zero, the notification badge disappears. |
| **Actual Output** | The notification badge for the specific category either reflects the correct number of pending documents, or it disappears (when there are no pending documents) |
| **Status** | Pass |

|  |  |
| --- | --- |
| **Test Case ID** | 296-I2 |
| **Purpose** | Verify that the number displayed in the pending documents notification badge is properly updated after a supervisor adds a new document. |
| **Preconditions** | Supervisor and Officer are logged in. |
| **Input** | 1. Supervisor uploads a new document in specific category 2. Officer navigates to main Document Category View page. |
| **Expected Output** | The pending documents count in the notification badge of the specific category is updated(incremented by one) to reflect the new number of the pending documents. |
| **Actual Output** | The pending document count is updated correctly after the new document is uploaded. |
| **Status** | Pass |

**User Story #241: Add Users with CSV File** **(Release 3.0)**

|  |  |
| --- | --- |
| **Test Case ID** | 241-U1 |
| **Purpose** | Verify that each row in CSV file is parsed into a table of users |
| **Preconditions** | User is logged in and is in the batch add users view. |
| **Input** | User selects a valid CSV file and selects the parse option. |
| **Expected Output** | Each row in the CSV files is added to a table of users, which is displayed in the current view. |
| **Actual Output** | The table of users populated from the CSV file is correctly displayed. |
| **Status** | Pass |

|  |  |
| --- | --- |
| **Test Case ID** | 241-U2 |
| **Purpose** | Verify that only CSV files can be parsed. |
| **Preconditions** | User is logged in and is in the batch add users view. |
| **Input** | User selects a file that is not of type CSV and selects the parse option. |
| **Expected Output** | A message is displayed warning the user that only CSV files can be parsed. |
| **Actual Output** | The warning message is displayed. |
| **Status** | Pass |

|  |  |
| --- | --- |
| **Test Case ID** | 241-U3 |
| **Purpose** | Verify that a file has been selected to be parsed. |
| **Preconditions** | User is logged in and is in the batch add users view. |
| **Input** | User clicks parse option without selecting a file. |
| **Expected Output** | A message is displayed warning the user that a file must be selected. |
| **Actual Output** | The warning message is displayed. |
| **Status** | Pass |

|  |  |
| --- | --- |
| **Test Case ID** | 241-I1 |
| **Purpose** | Verify that all valid users from the CSV file are added to the user database table. |
| **Preconditions** | User is logged in and is in the batch add users view. |
| **Input** | 1. User selects a CSV file containing valid users and selects the parse option. 2. User selects the add-all option. 3. User closes batch add users view 4. User navigates to the edit user view in the user management page |
| **Expected Output** | All users will be removed from the parsed user table in the batch add users view after the add-all option is clicked.  All users from the CSV file will be present in the edit user table, indicating that they were successfully added to the database. |
| **Actual Output** | All users from the CSV file were deleted from parsed user table after the add-all option was selected. These users are now visible in the edit user table in the user management page. |
| **Status** | Pass |

|  |  |
| --- | --- |
| **Test Case ID** | 241-I2 |
| **Purpose** | Verify that invalid users from the CSV file are not added to the user database table. |
| **Preconditions** | User is logged in and is in the batch add users view. |
| **Input** | 1. User selects a CSV file containing invalid users and selects the parse option. 2. User selects the add all option. 3. User closes batch add users view 4. User navigates to the edit user view in the user management page |
| **Expected Output** | No users are removed from the parsed user table in the batch add users view. An error message is displayed.  No users from the CSV file are present in the edit user table, indicating that they were rejected by the database. |
| **Actual Output** | After the add-all option was selected, all users from the CSV file are still present parsed user table and an error message is displayed.  These users are not visible in the edit user table in the user management page. |
| **Status** | Pass |

**User Story #300: Improve Document List Presentation** **(Release 3.0)**

|  |  |
| --- | --- |
| **Test Case ID** | 300-U1 |
| **Purpose** | Verify that pending documents are displayed in correct order and with correct status color label. |
| **Preconditions** | User is logged in and is in document category view. |
| **Input** | User selects a document category that contains pending documents. |
| **Expected Output** | Pending documents will appear first before other types of documents and will have red status label color. |
| **Actual Output** | All pending documents are ordered correctly and have appropriate status label colors. |
| **Status** | Pass |

|  |  |
| --- | --- |
| **Test Case ID** | 300-U2 |
| **Purpose** | Verify that reviewed documents are displayed in correct order and with correct status color label. |
| **Preconditions** | User is logged in and is in document category view. |
| **Input** | User selects a document category that contains reviewed documents. |
| **Expected Output** | Reviewed documents will appear second after pending documents and before done documents, and they will have yellow status label color. |
| **Actual Output** | All reviewed documents are ordered correctly and have appropriate status label colors. |
| **Status** | Pass |

|  |  |
| --- | --- |
| **Test Case ID** | 300-I1 |
| **Purpose** | Verify that documents are displayed in correct order and with correct status color label, after one or more document status changes |
| **Preconditions** | User is logged in and is in document category view. |
| **Input** | 1. User selects a document category that contains pending, reviewed and done documents. 2. Users selects one or more pending documents and accepts one or more reviewed documents |
| **Expected Output** | Documents will be ordered by status in the following way:   1. Pending documents 2. Reviewed documents 3. Done documents.   Document status label colors will change as follows:   * Pending to Reviewed: Red to Yellow * Reviewed to Done: Yellow to Green   Any document that was not modified, keeps same status color. |
| **Actual Output** | All documents are ordered correctly and have appropriate status label colors after the status changes of one or more documents. |
| **Status** | Pass |

**User Story #246: Add Watch Orders with CSV File** **(Release 3.0)**

|  |  |
| --- | --- |
| **Test Case ID** | 246-U1 |
| **Purpose** | Verify that each row in CSV file is parsed into a table of watch orders |
| **Preconditions** | User is logged in and is in the batch add watch orders view. |
| **Input** | User selects a valid CSV file and selects the parse option. |
| **Expected Output** | Each row in the CSV files is added to a table of watch orders, which is displayed in the current view. |
| **Actual Output** | The table of watch orders populated from the CSV file is correctly displayed. |
| **Status** | Pass |

|  |  |
| --- | --- |
| **Test Case ID** | 246-U2 |
| **Purpose** | Verify that only CSV files can be parsed. |
| **Preconditions** | User is logged in and is in the batch add watch orders view. |
| **Input** | User selects a file that is not of type CSV and selects the parse option. |
| **Expected Output** | A message is displayed warning the user that only CSV files can be parsed. |
| **Actual Output** | The warning message is displayed. |
| **Status** | Pass |

|  |  |
| --- | --- |
| **Test Case ID** | 246-U3 |
| **Purpose** | Verify that a file has been selected to be parsed. |
| **Preconditions** | User is logged in and is in the batch add watch orders view. |
| **Input** | User clicks parse option without selecting a file. |
| **Expected Output** | A message is displayed warning the user that a file must be selected. |
| **Actual Output** | The warning message is displayed. |
| **Status** | Pass |

|  |  |
| --- | --- |
| **Test Case ID** | 246-U4 |
| **Purpose** | Verify that only rejected watch orders are removed from the table after attempting to add watch orders to the database. |
| **Preconditions** | User is logged in and is in the batch add watch orders view. |
| **Input** | 1. User selects a valid CSV file and selects the parse option. 2. User selects the add all option. |
| **Expected Output** | If a watch order from the CSV was successfully added to the database, it will disappear from the parsed order table. If an order was rejected, it will remain in the table. A message will show how many orders were successfully added and how many orders rejected. |
| **Actual Output** | All accepted watch orders are removed from the parsed order table. If there are any rejected orders, they remain in the table. The correct counts of accepted and failed orders are displayed. |
| **Status** | Pass |

|  |  |
| --- | --- |
| **Test Case ID** | 246-I1 |
| **Purpose** | Verify that all valid watch orders from the CSV file are added to the watch-orders database table. |
| **Preconditions** | User is logged in and is in the batch add watch orders view. |
| **Input** | 1. User selects a CSV file containing valid orders and selects the parse option. 2. User selects the add-all option. 3. User closes batch add watch orders view 4. User examines database, checking for new watch orders |
| **Expected Output** | All watch orders will be removed from the parsed order table in the batch add watch orders view after the add-all option is clicked.  All watch orders from the CSV file will be present in the watch-orders database table, indicating that they were successfully added. |
| **Actual Output** | All watch orders from the CSV file were deleted from parsed order table after the add-all option was selected. These orders are now present in the database. |
| **Status** | Pass |

**User Story #247: View Watch Orders in a Map** **(Release 3.0)**

|  |  |
| --- | --- |
| **Test Case ID** | 247-U1 |
| **Purpose** | Verify that map marker position corresponds to watch order address. |
| **Preconditions** | User is logged in and is in the view-watch-orders view. There is at least one watch order in the system. |
| **Input** | User clicks on one or more map markers. |
| **Expected Output** | An information window for the marker is displayed which shows the address that should correspond to the correct position on the map. |
| **Actual Output** | The position of each map marker is correct, in the accordance with the watch order address. |
| **Status** | Pass |

|  |  |
| --- | --- |
| **Test Case ID** | 247-U2 |
| **Purpose** | Verify that the correct marker description and date is displayed. |
| **Preconditions** | User is logged in and is in the view-watch-orders view. There is at least one watch order in the system. |
| **Input** | User clicks on one or more map markers. |
| **Expected Output** | The information window of the marker contains the description and date of the watch order that is in the database. |
| **Actual Output** | The correct attributes, description and date, are displayed for each marker. |
| **Status** | Pass |

|  |  |
| --- | --- |
| **Test Case ID** | 247-U3 |
| **Purpose** | Verify that no markers are displayed when there are no watch orders in the database. |
| **Preconditions** | User is logged in and is in the category-view page. There are no watch orders in the system. |
| **Input** | User clicks on the view-watch-orders option. |
| **Expected Output** | The map will displayed with no markers. |
| **Actual Output** | The map is rendered without any markers. |
| **Status** | Pass |

|  |  |
| --- | --- |
| **Test Case ID** | 247-I1 |
| **Purpose** | Verify that an officer’s watch order map is updated after a supervisor modifies the watch order list. |
| **Preconditions** | Supervisor is logged in and is in the batch add watch orders view. Officer is logged in and is in the view-watch-orders page. |
| **Input** | Supervisor uploads a new list of one or more valid watch orders |
| **Expected Output** | The map in the officer’s watch order view will be updated with the new watch order markers. The previous watch orders should no longer be visible. |
| **Actual Output** | The officer watch order map only shows the markers belonging to the new watch orders. |
| **Status** | Pass |

**User Story #304: Add Watch Order** **(Release 3.0)**

|  |  |
| --- | --- |
| **Test Case ID** | 304-U1 |
| **Purpose** | Verify that an incomplete watch order cannot be added. |
| **Preconditions** | User is logged in and is in the add watch orders view. |
| **Input** | User clicks add-watch-order button without filling in all fields |
| **Expected Output** | The system should display an error message indicating that all fields must be completed. |
| **Actual Output** | An error message is shown. |
| **Status** | Pass |

|  |  |
| --- | --- |
| **Test Case ID** | 304-U2 |
| **Purpose** | Verify that a success message is shown after attempting to add a valid watch order. |
| **Preconditions** | User is logged in and is in the add watch orders view. |
| **Input** | 1. User fills out fields with valid data. 2. User selects ‘add-watch-order’ option |
| **Expected Output** | The watch order will be accepted by the database and a success message will displayed. |
| **Actual Output** | The correct status message is displayed. |
| **Status** | Pass |

|  |  |
| --- | --- |
| **Test Case ID** | 304-U3 |
| **Purpose** | Verify that a failure message is shown after attempting to add a invalid watch order. |
| **Preconditions** | User is logged in and is in the add watch orders view. |
| **Input** | 1. User fills out fields with invalid data. 2. User selects ‘add-watch-order’ option |
| **Expected Output** | The watch order will not be accepted by the database and a failure message will displayed. |
| **Actual Output** | The correct status message is displayed. |
| **Status** | Pass |

|  |  |
| --- | --- |
| **Test Case ID** | 304-I1 |
| **Purpose** | Verify that all valid watch orders are added to the watch-orders database table. |
| **Preconditions** | User is logged in and is in the add watch orders view. |
| **Input** | 1. User fills out fields with valid data. 2. User selects ‘add-watch-order’ option 3. User examines database, checking for the new watch order |
| **Expected Output** | The new watch order will be present in the watch-orders database table, indicating that it was successfully added. |
| **Actual Output** | The database contains the new watch order. |
| **Status** | Pass |

**User Story #306: Delete Watch Order** **(Release 3.0)**

|  |  |
| --- | --- |
| **Test Case ID** | 306-U1 |
| **Purpose** | Verify that a success message is shown after attempting to add a remove a watch order. |
| **Preconditions** | User is logged in and is in the edit watch orders view. |
| **Input** | User clicks ‘delete’ button. |
| **Expected Output** | The watch order will be removed from the database and a success message will displayed. |
| **Actual Output** | The correct status message is displayed. |
| **Status** | Pass |

|  |  |
| --- | --- |
| **Test Case ID** | 306-I1 |
| **Purpose** | Verify that the selected watch order is removed watch-orders database table. |
| **Preconditions** | User is logged in and is in the edit watch orders view. |
| **Input** | 1. User clicks ‘delete’ button. 2. User examines database, checking for the selected watch order |
| **Expected Output** | The selected watch order will be removed from the watch-orders database table, indicating that it was successfully deleted. |
| **Actual Output** | The database no longer contains the selected watch order. |
| **Status** | Pass |

**User Story #307: Edit Watch Order** **(Release 3.0)**

|  |  |
| --- | --- |
| **Test Case ID** | 307-U1 |
| **Purpose** | Verify that a success message is shown after attempting to edit a watch order. |
| **Preconditions** | User is logged in and is in the edit watch orders view. |
| **Input** | 1. User clicks edit button for selected watch order 2. User fills watch order fields with new values. 3. User clicks update button |
| **Expected Output** | The watch order will be updated in the database with the new values, and a success message will displayed. |
| **Actual Output** | The correct status message is displayed. |
| **Status** | Pass |

|  |  |
| --- | --- |
| **Test Case ID** | 307-I1 |
| **Purpose** | Verify that the selected watch order is updated in watch-orders database table. |
| **Preconditions** | User is logged in and is in the edit watch orders view. |
| **Input** | 1. User clicks edit button for selected watch order 2. User fills watch order fields with new values. 3. User clicks update button 4. User examines database, checking for the selected watch order |
| **Expected Output** | The selected watch order will be updated in the watch-orders database table with the new field values, indicating that it was successfully edited. |
| **Actual Output** | The database contains the updated watch order. |
| **Status** | Pass |

**User Story #309: Automatically Delete Watch Orders** **(Release 3.0)**

|  |  |
| --- | --- |
| **Test Case ID** | 309-U1 |
| **Purpose** | Verify that expired watch orders are removed from system. |
| **Preconditions** | User is logged in and is in the add watch orders view. |
| **Input** | 1. User fills watch order fields with new values, adding an expiration date that has already passed. 2. User clicks add watch order button 3. User examines database, checking for the selected watch order |
| **Expected Output** | The new watch order will not be in the watch-orders database table indicating that it was successfully removed. |
| **Actual Output** | The database does not contain the expired watch order. |
| **Status** | Pass |

|  |  |
| --- | --- |
| **Test Case ID** | 309-I1 |
| **Purpose** | Verify that watch orders are removed from the system after they have expired. |
| **Preconditions** | User is logged in and is in the watch orders view. |
| **Input** | 1. User fills add watch order fields with valid data. 2. User clicks add watch order button |
| **Expected Output** | After the current date is past the expiration date of the watch order, the watch order will not be visible. |
| **Actual Output** | The expired watch order is no longer in the system. |
| **Status** | Pass |

**User Story #310: Display Watch Order List Table** **(Release 3.0)**

|  |  |
| --- | --- |
| **Test Case ID** | 310-U1 |
| **Purpose** | Verify that no watch orders are displayed in the table when there are no watch orders in the database. |
| **Preconditions** | User is logged in and is in the category-view page. There are no watch orders in the system. |
| **Input** | User clicks on the view-watch-orders option. |
| **Expected Output** | The table will displayed with no watch orders. |
| **Actual Output** | The table is rendered without any watch orders. |
| **Status** | Pass |

|  |  |
| --- | --- |
| **Test Case ID** | 310-U2 |
| **Purpose** | Verify that all watch orders are displayed in the table when there are watch orders in the database. |
| **Preconditions** | User is logged in and is in the category-view page. There are is at least one watch orders in the system. |
| **Input** | User clicks on the view-watch-orders option. |
| **Expected Output** | The table will displayed with all watch orders. |
| **Actual Output** | The table is rendered with all watch orders. |
| **Status** | Pass |

|  |  |
| --- | --- |
| **Test Case ID** | 310-I1 |
| **Purpose** | Verify that an officer’s watch order table is updated after a supervisor modifies the watch order list. |
| **Preconditions** | Supervisor is logged in and is in the batch add watch orders view. Officer is logged in and is in the view-watch-orders page. |
| **Input** | Supervisor uploads a new list of one or more valid watch orders |
| **Expected Output** | The watch order table in the officer’s watch order view will be updated with the new watch orders. The previous watch orders should no longer be visible. |
| **Actual Output** | The officer watch order table only shows the new watch orders. |
| **Status** | Pass |

# GLOSSARY

**Actor:** Anyone that can use, or misuse, the system.

**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. It used to segregate documents into appropriate groups.

**Class Diagram**: A model representing the different classes within a software system and how they interact with each other.

**Database**: A collection of related data.

**Database Management System**: A software package to facilitate the creation and maintenance of a computerized database.

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

**Message:** A free-form data entry.

**Model**: An abstract representation of a system that enables us to answer questions about the system.

**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.

**Server**: A computer that provides client stations with access to files as a shared resource to a computer network.

**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.

**System:** Underlying reality.

**UML:** See Unified Modeling Language.

**Unified Modeling Language \*:** A standard set of notations for representing models.

**Use Case:** A general sequence of events that defines all possible actions between one or many actors and the system for a given piece of functionality.

**User Interface** – the way through which a user interacts with the computer system.

**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

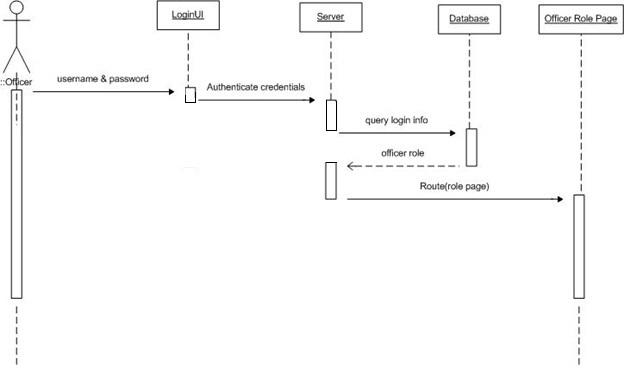
###### Figure 2: Class Diagram



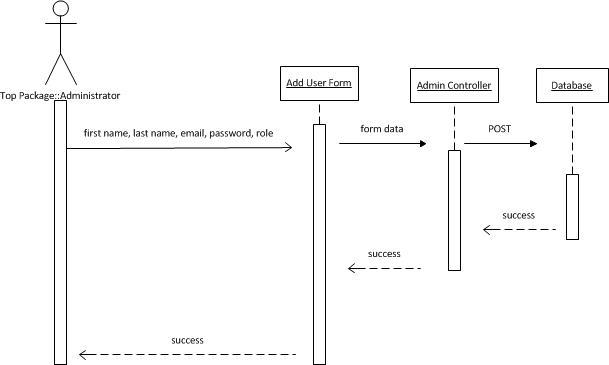
## Appendix B - Sequence Diagrams Versions 1 and 2

### 

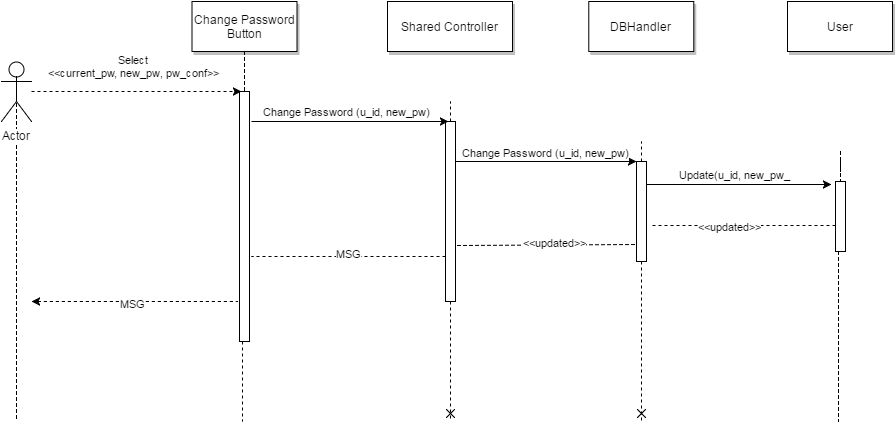
###### Figure 3: Login sequence diagram



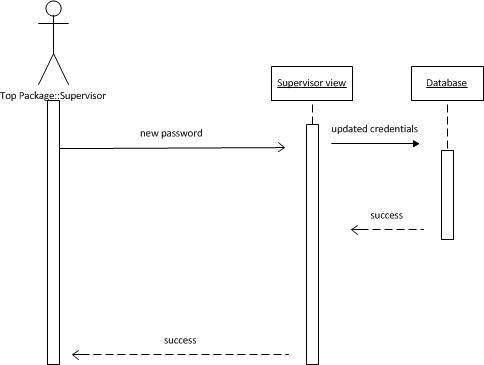
###### Figure 4: Add user sequence diagram



###### Figure 5: Change Password sequence diagram

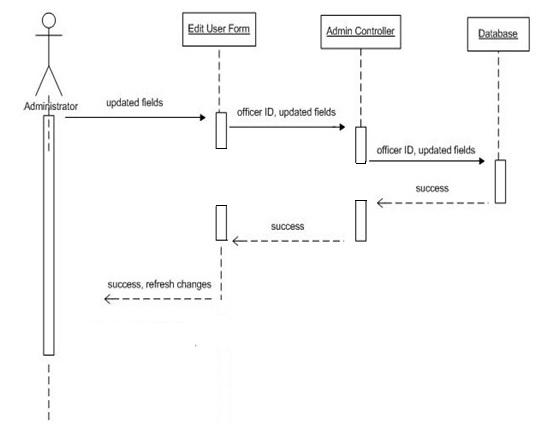


###### Figure 6: Reset Password sequence diagram

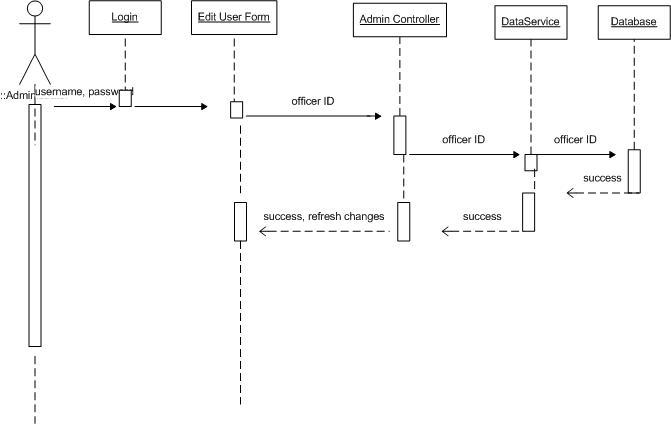


### 

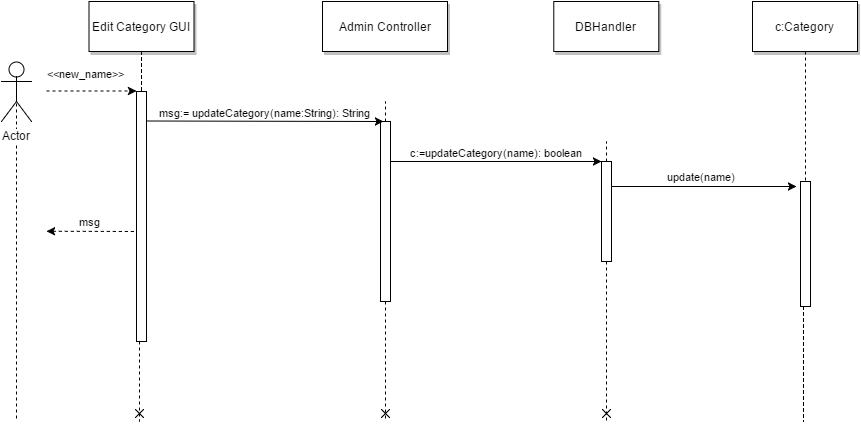
###### Figure 7: Edit User sequence diagram



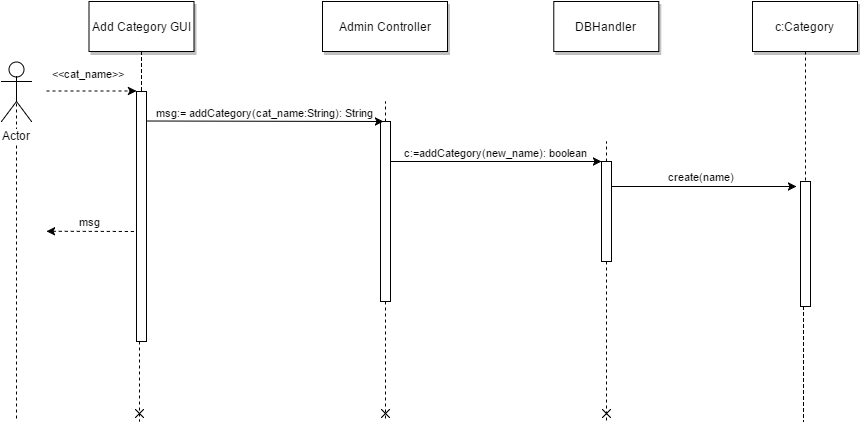
###### Figure 8: Delete User sequence diagram



###### Figure 9: Edit Category sequence diagram

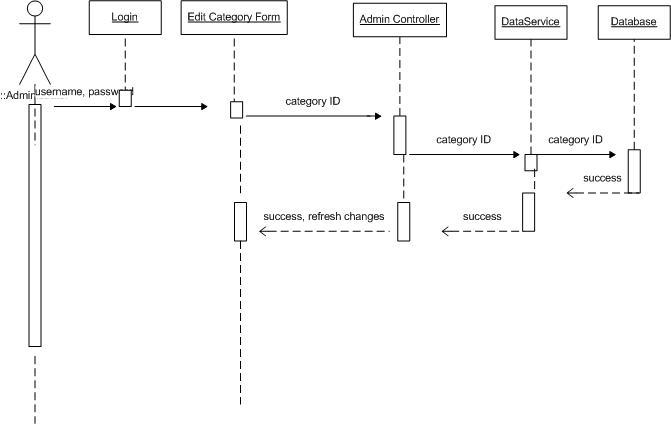


###### Figure 10: Add Category sequence diagram

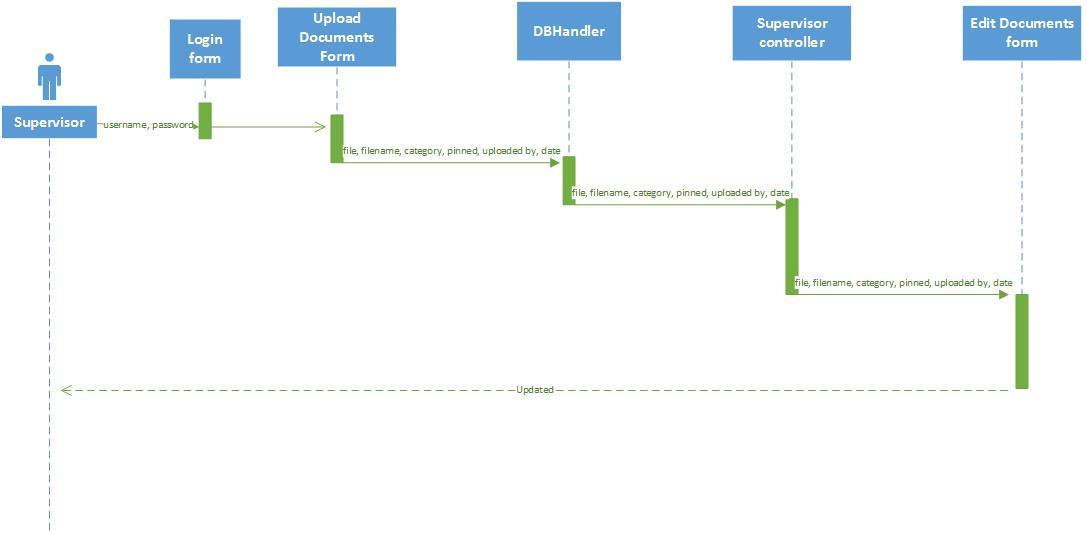


###### 

###### Figure 11: Remove Category sequence diagram

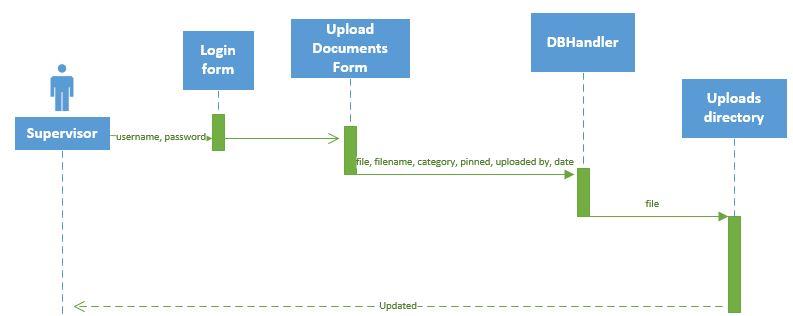


###### Figure 12: Add Documents sequence diagram

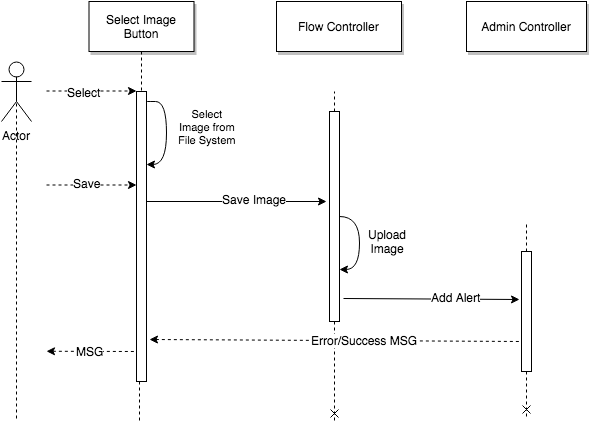


###### 

###### Figure 13: Upload Document sequence diagram

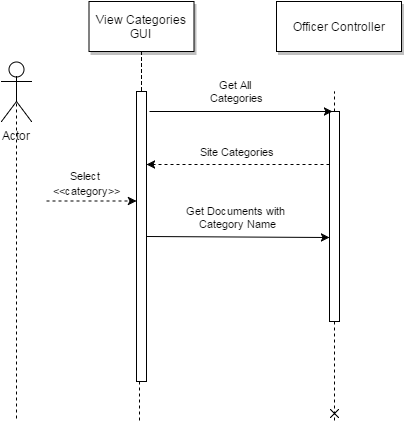


###### Figure 14: Customize Site Settings sequence diagram

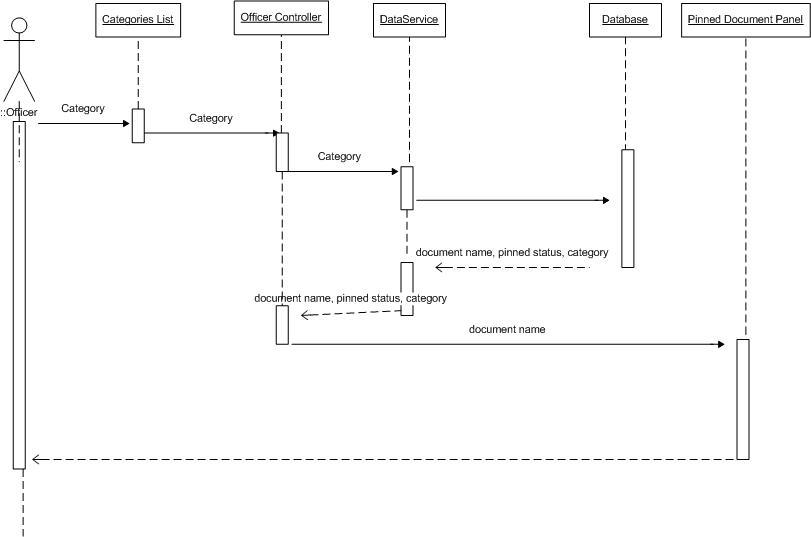


###### 

###### Figure 15: View Categories sequence diagram

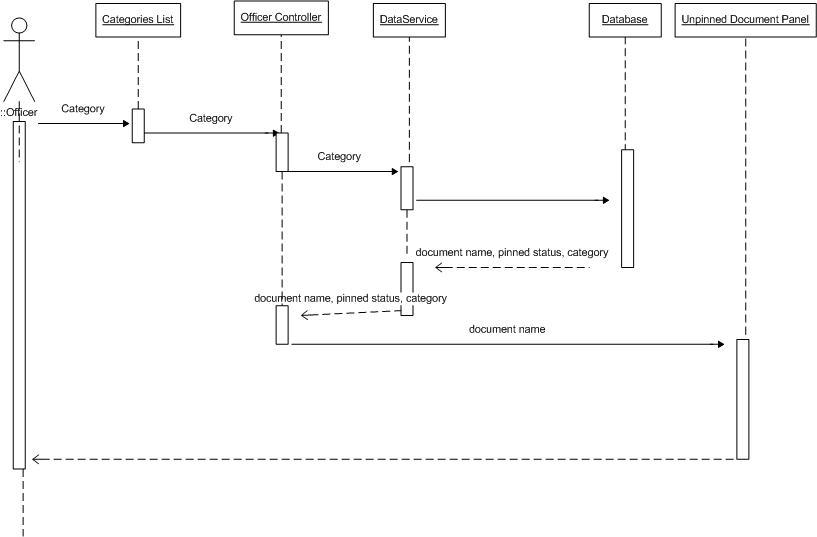


###### Figure 16: List Pinned Documents sequence diagram

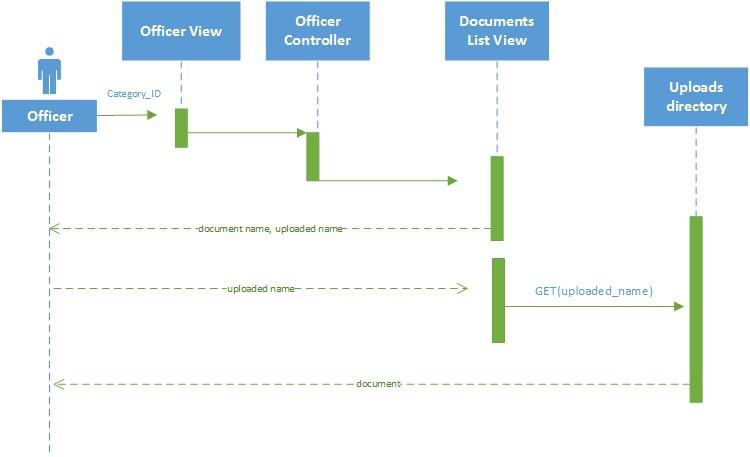


###### 

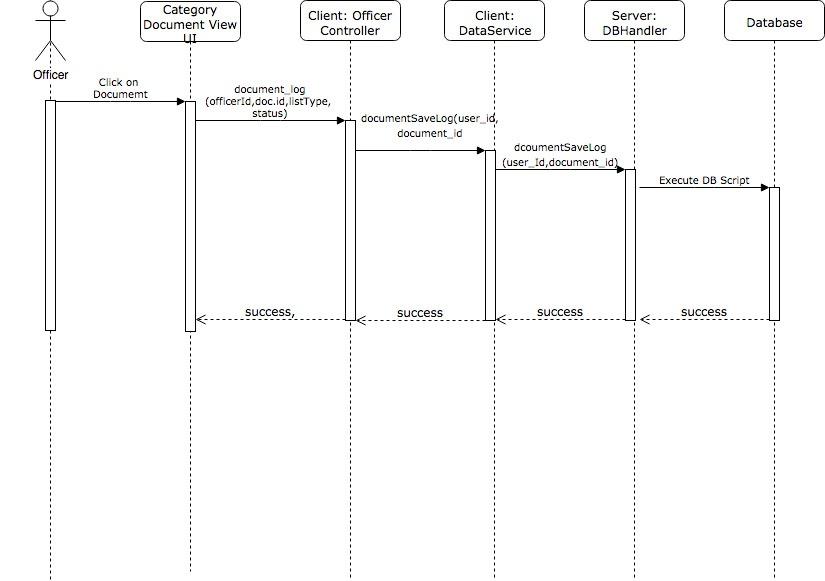
###### Figure 17: List Unpinned Documents sequence diagram



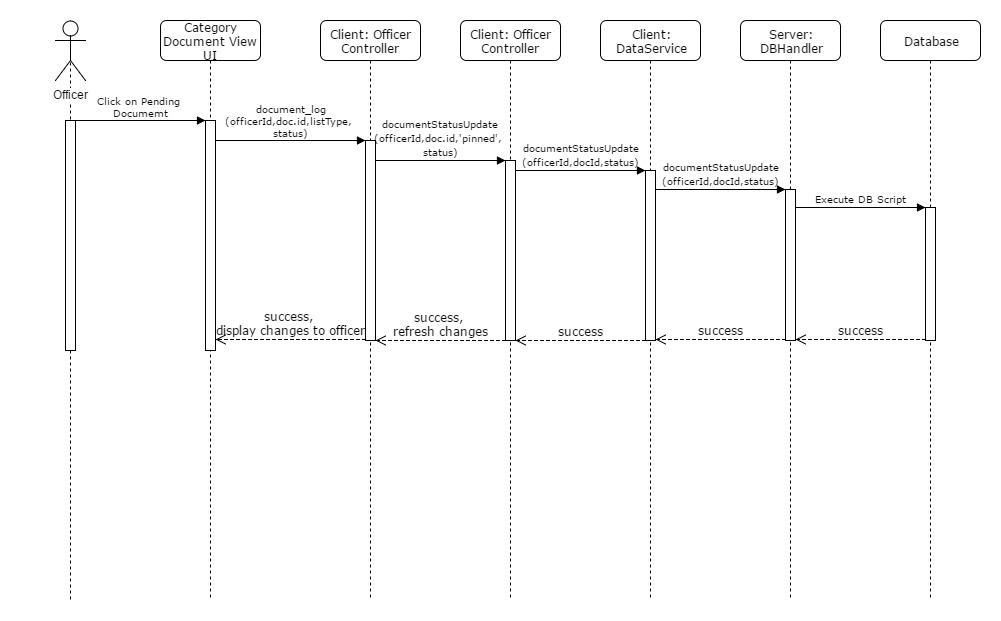
###### Figure 18: View Documents sequence diagram



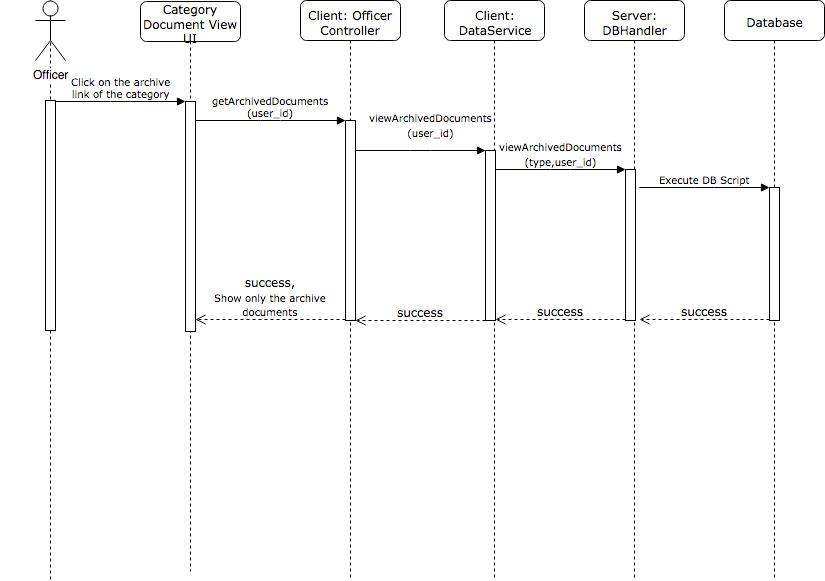
###### Figure 19: Document Log (Version 2.0)



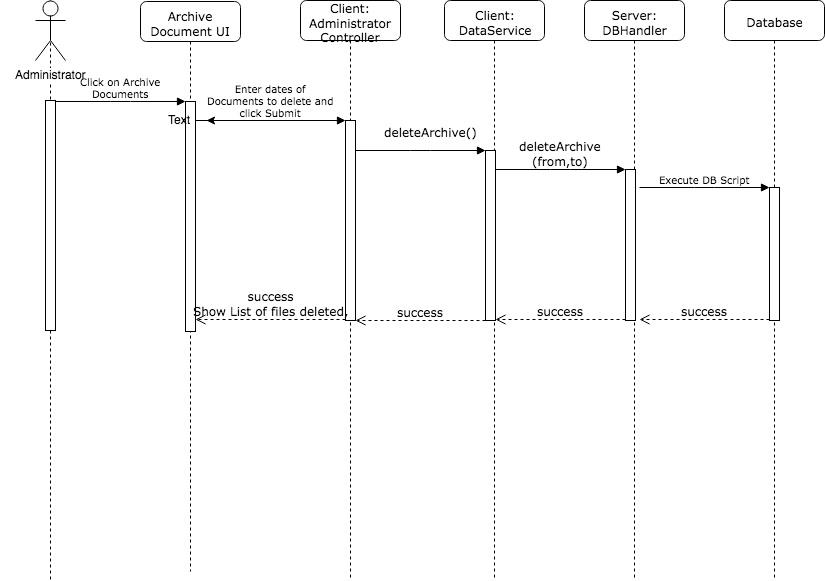
###### Figure 20: Change Document Status from “Pending” to “Reviewed” (Version 2.0)



###### Figure 21: Archived View for documents older than 7 days (Version 2.0)

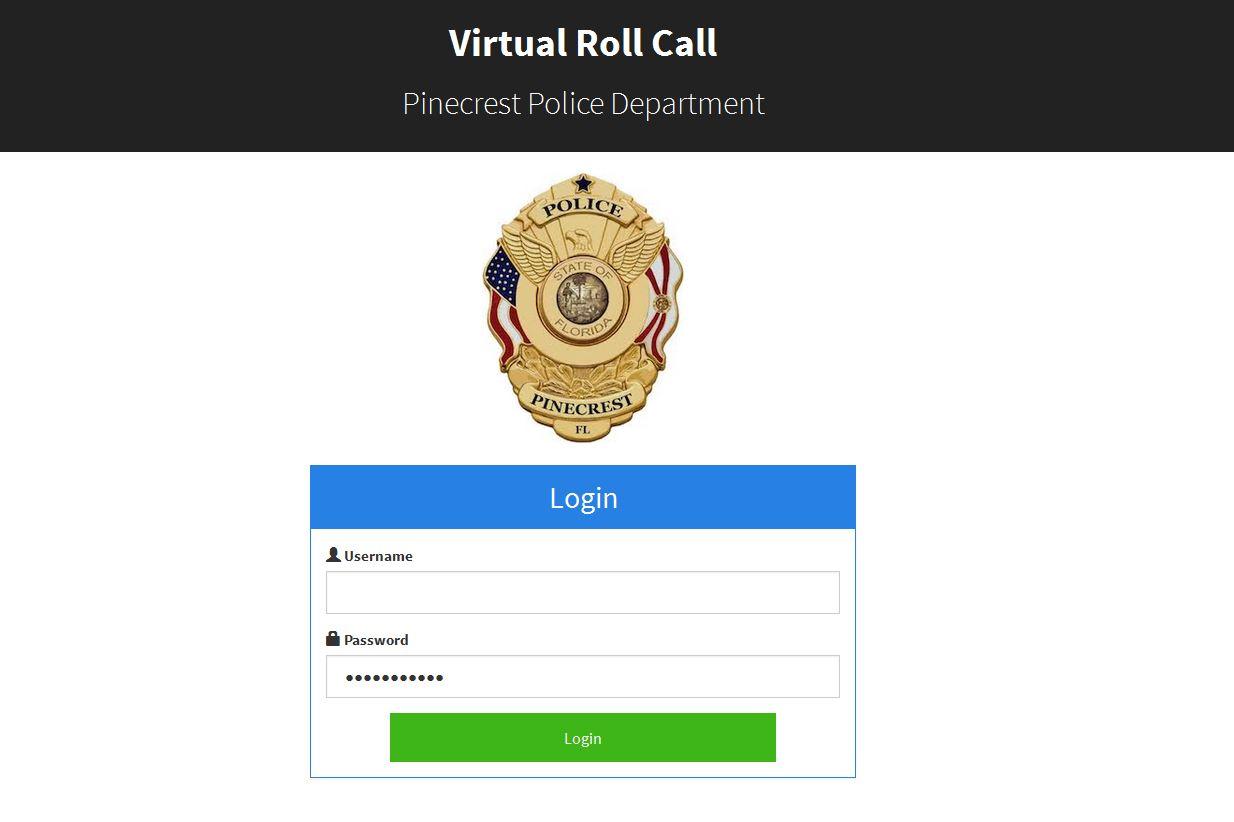


###### Figure 22: Delete old archived Documents using a date range (Version 2.0)

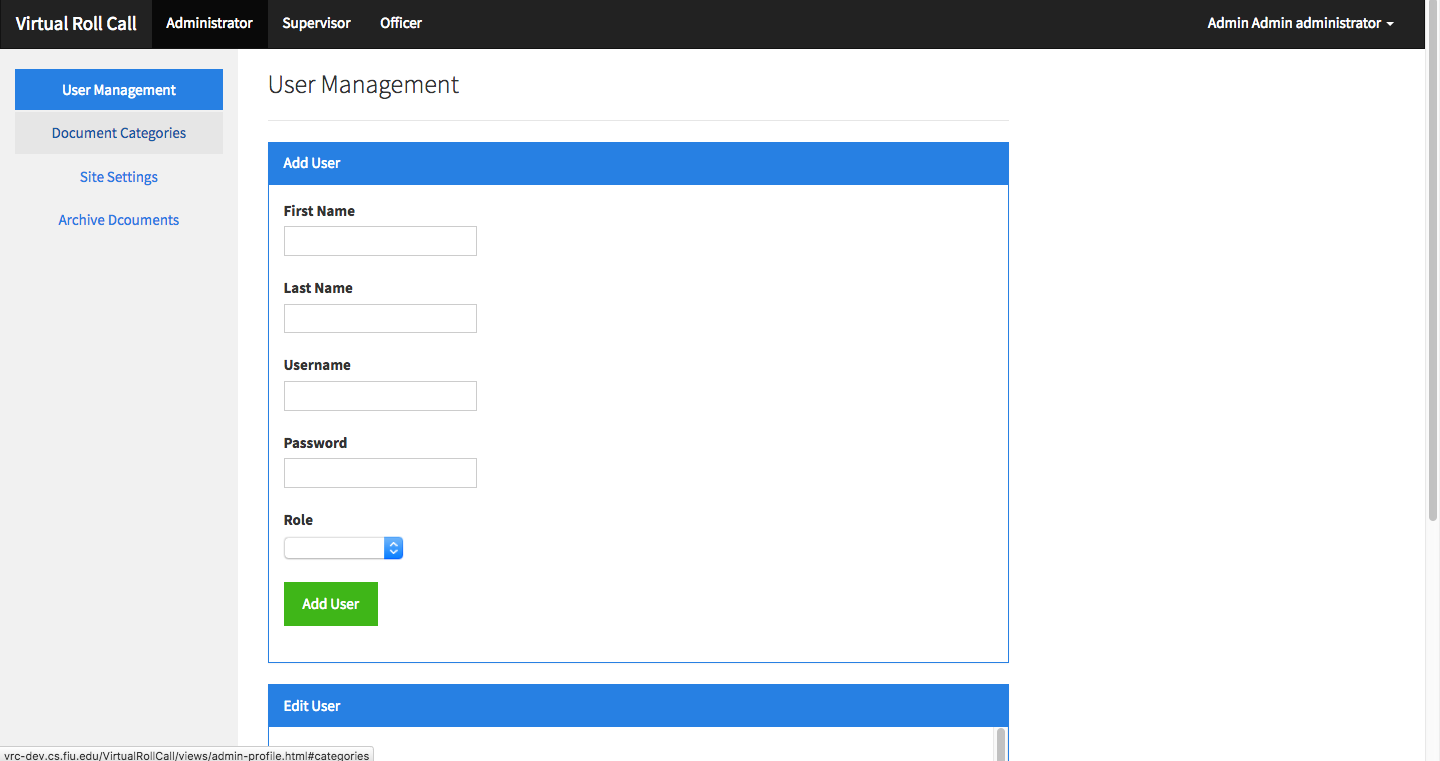


## 

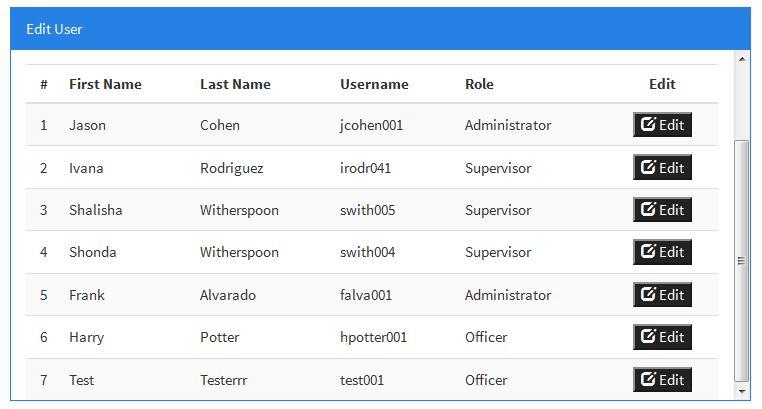
## Appendix C - User Interface Design



**Figure C.1 – VRC Login Page**

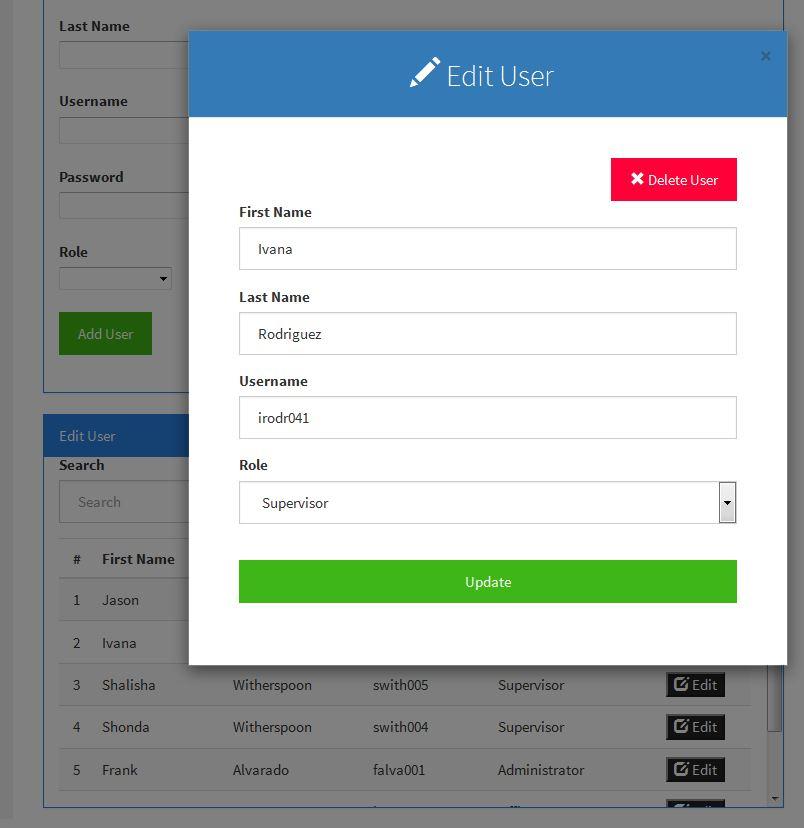


**Figure C.2.1 – Administrator Dashboard - Defaults to User Management**

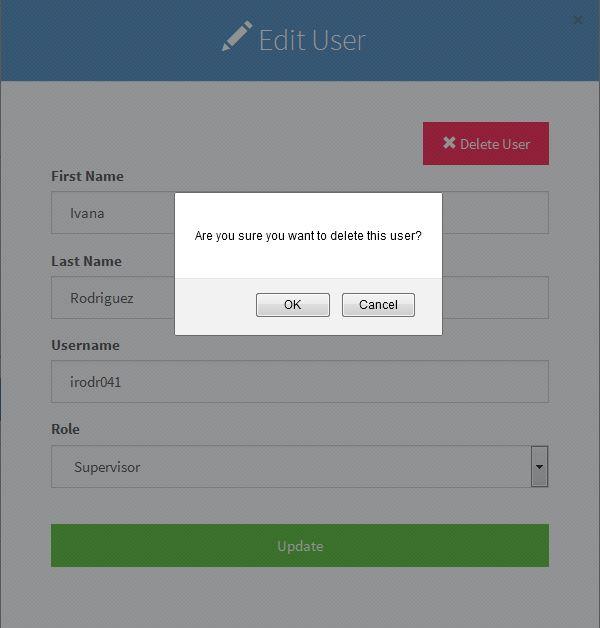


**Figure C.2.2 – Edit User Pane –**

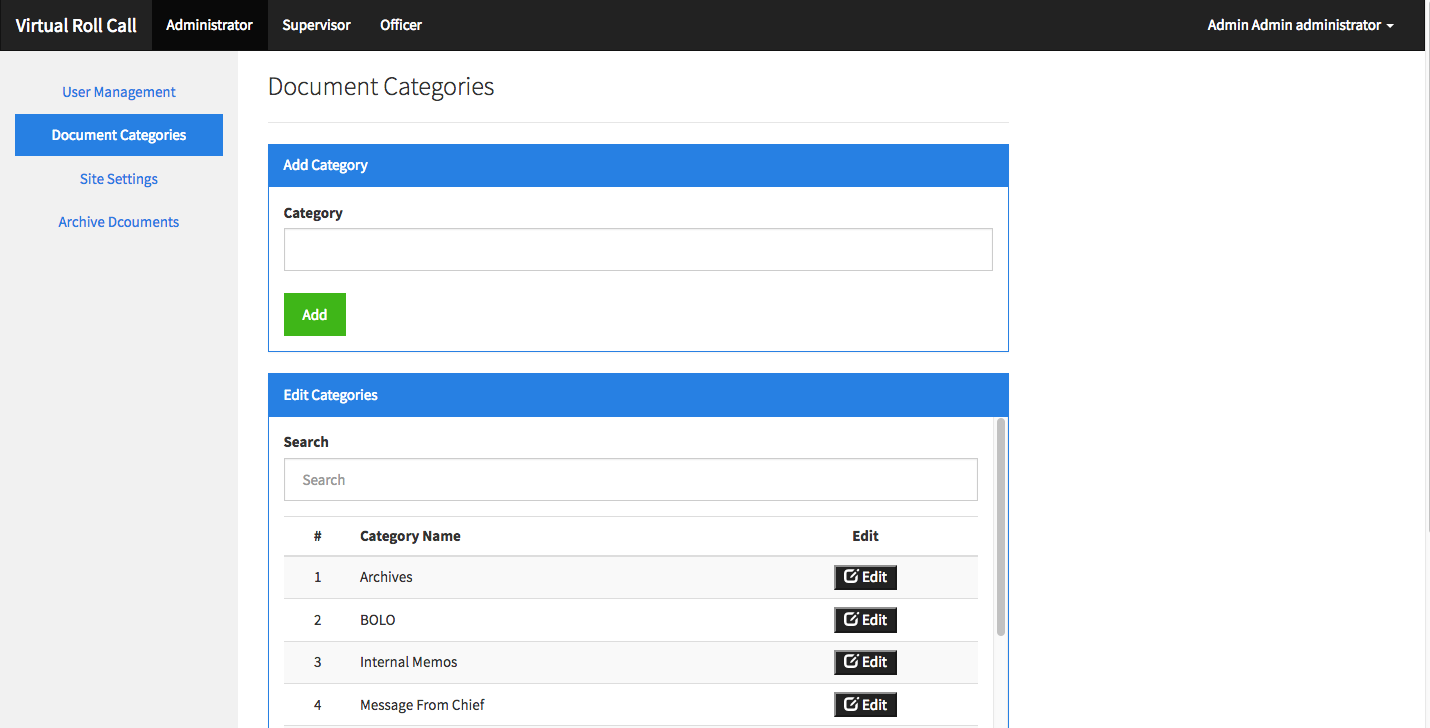
**List of users in the systems are displayed and can be filtered by attribute**



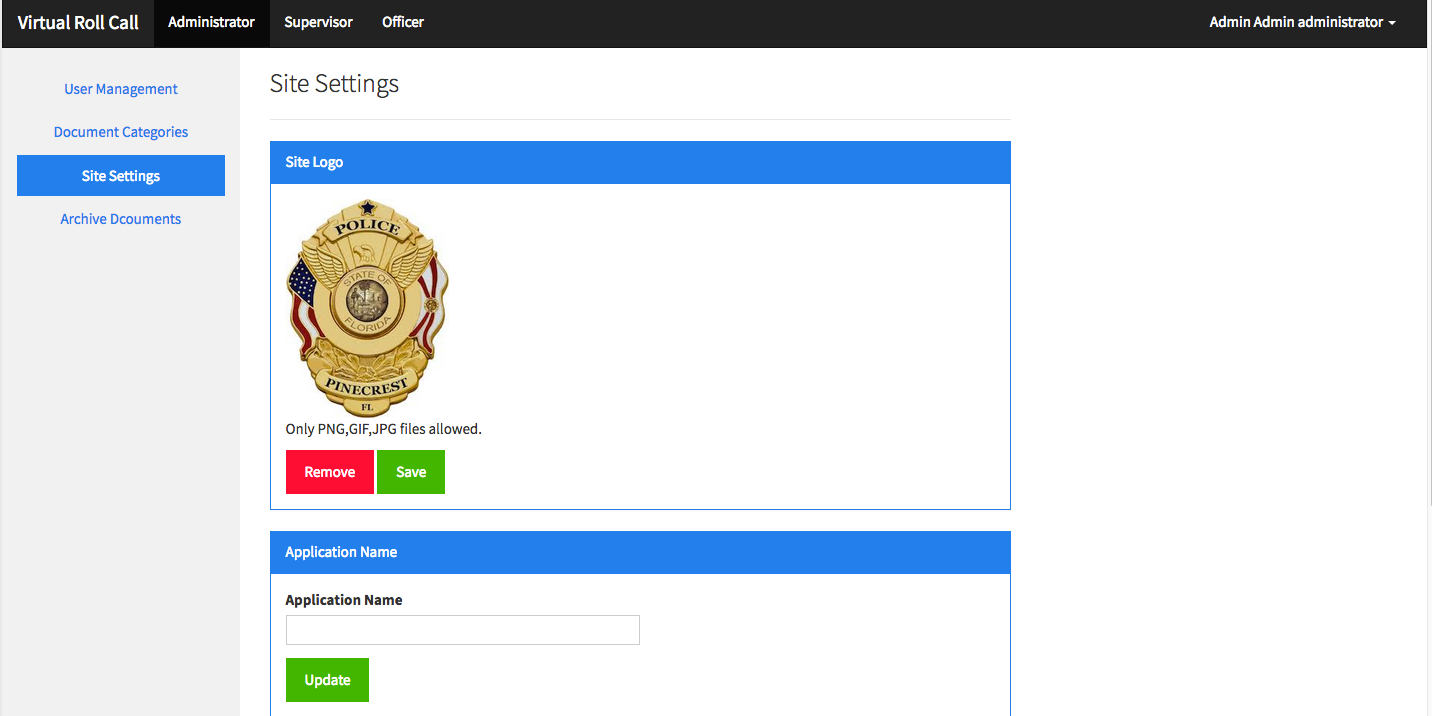
**Figure C.3 – Edit User Module**



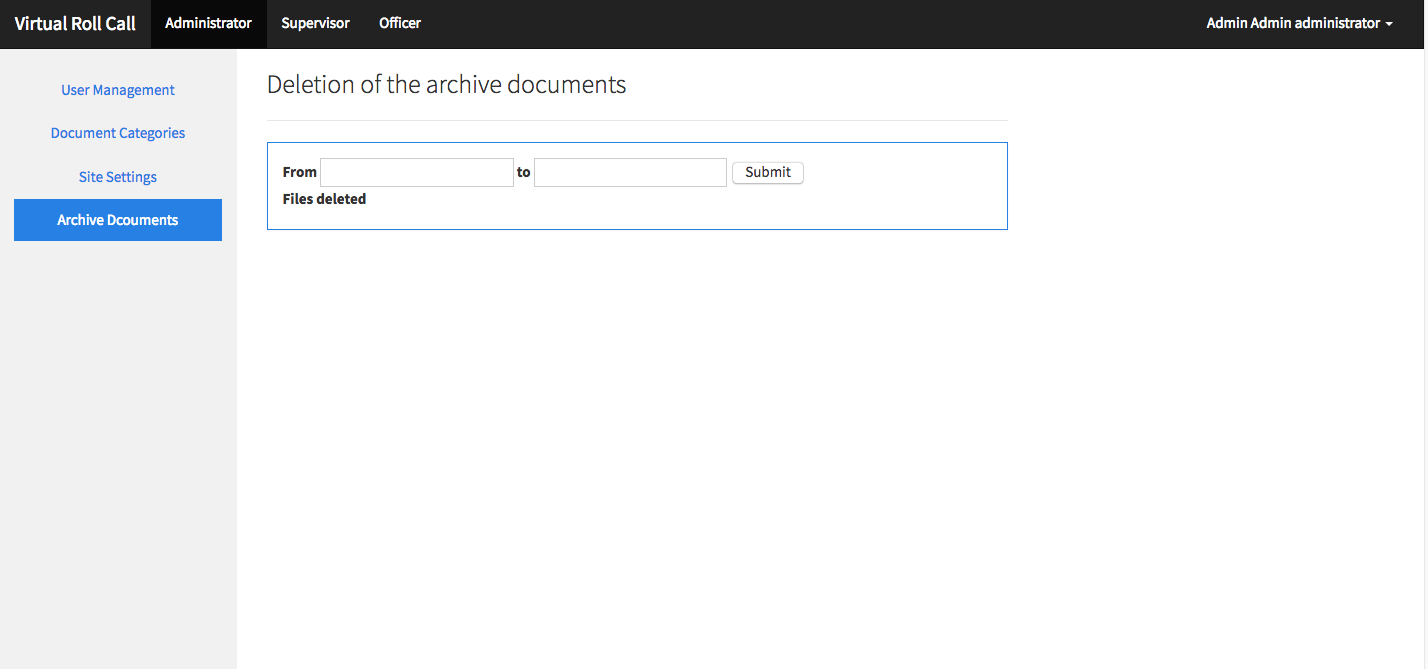
**Figure C.4 – Delete User Button Selected**



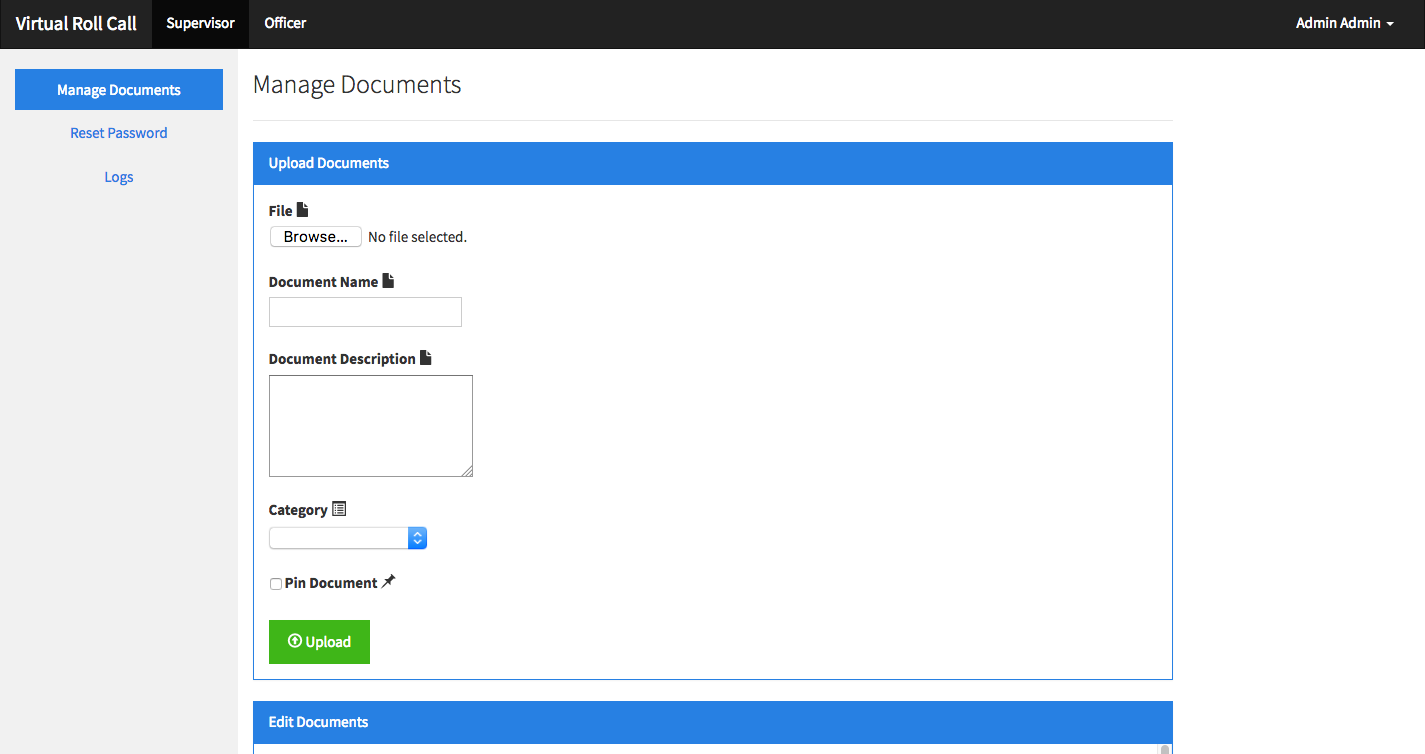
**Figure C.5 – Add Categories Pane - available categories shown in second pane**

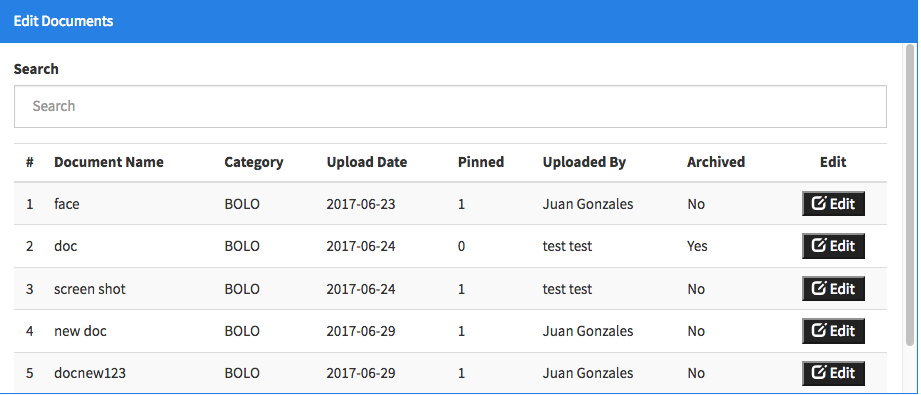


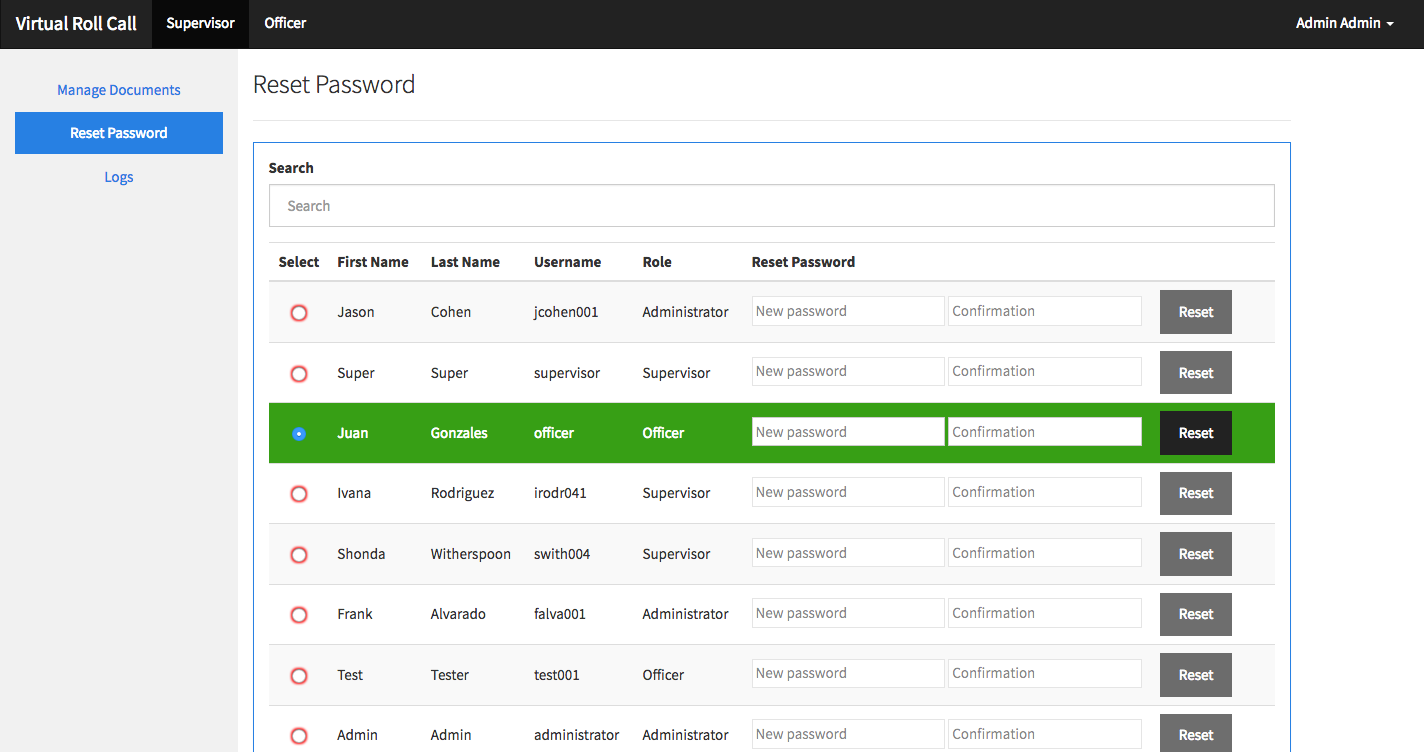
**Figure C.6 – Site Settings Page**



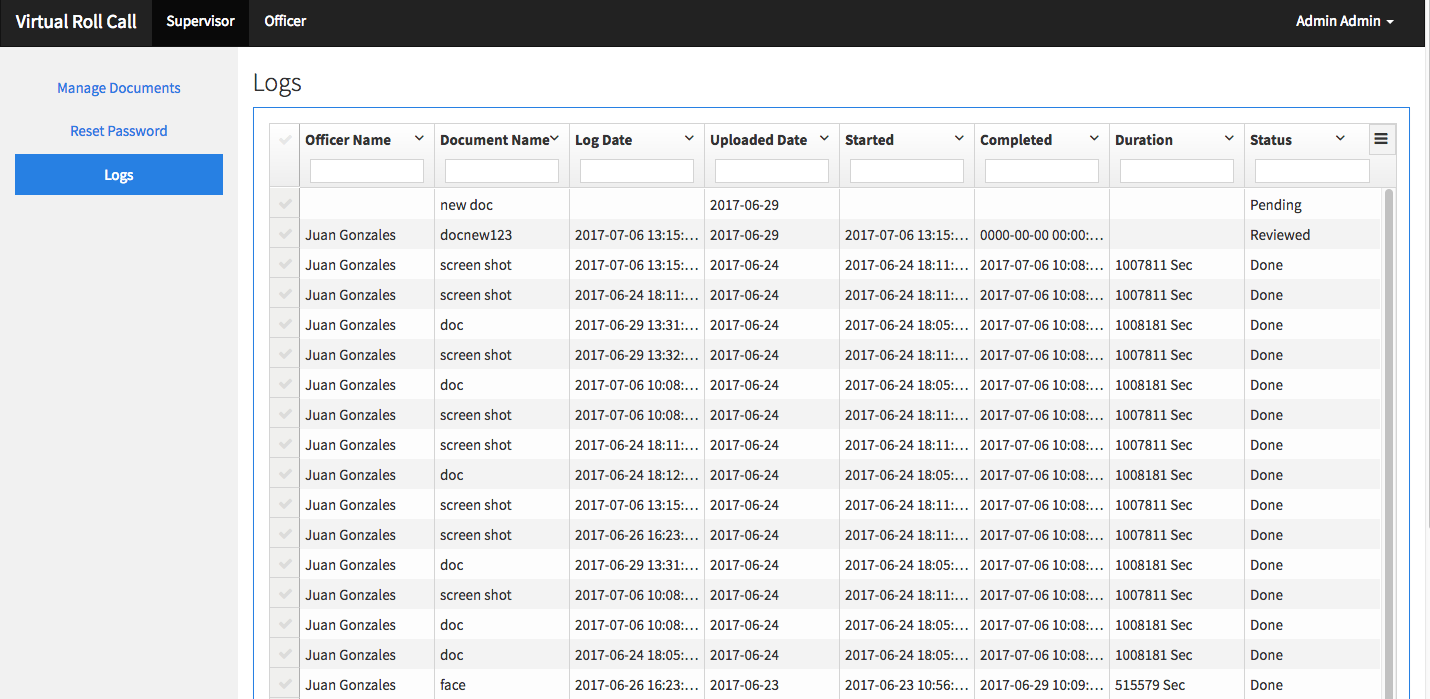
**Figure C.7 – Physical** **Archive Documents**

 **Figure C.8 – Supervisor Dashboard - Defaults to Manage Documents**

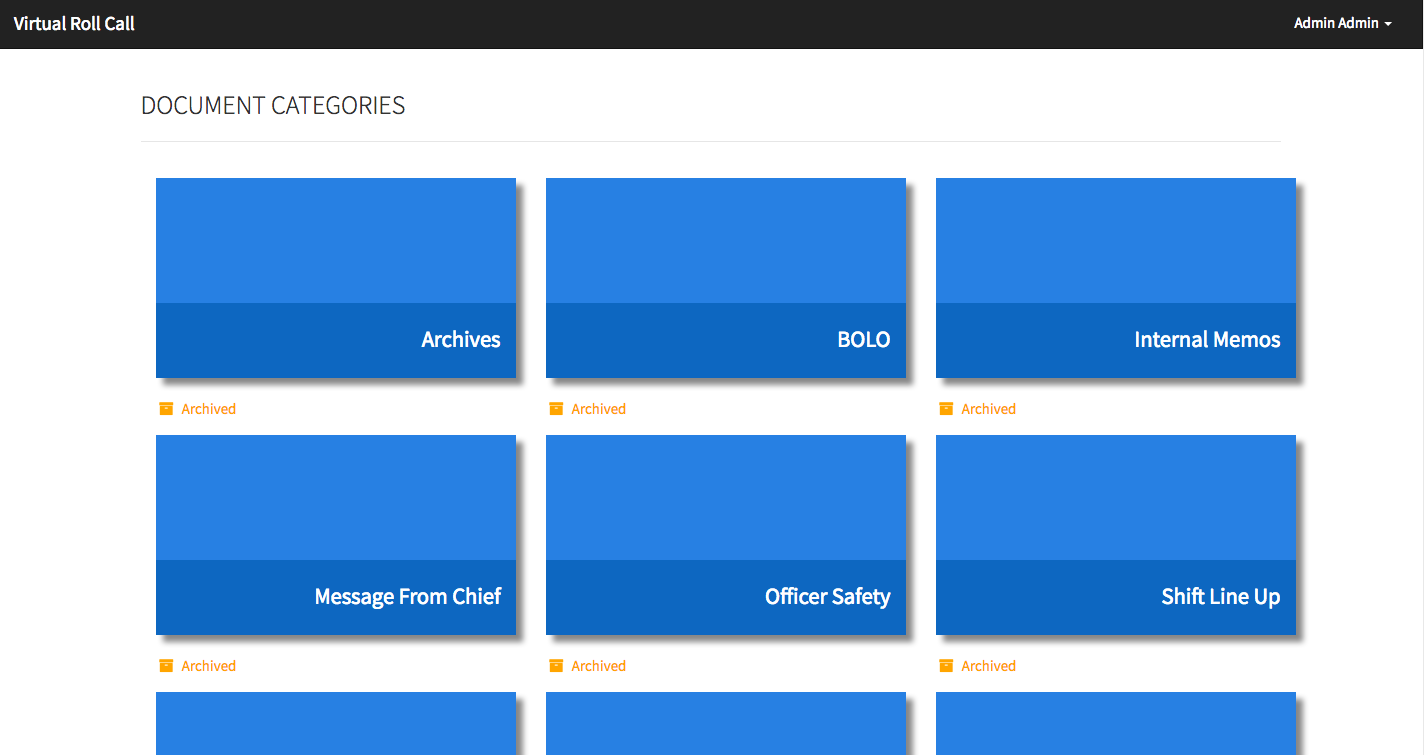
** Figure C.9 – Edit Documents Pane - lists metadata of all documents uploaded in the system**

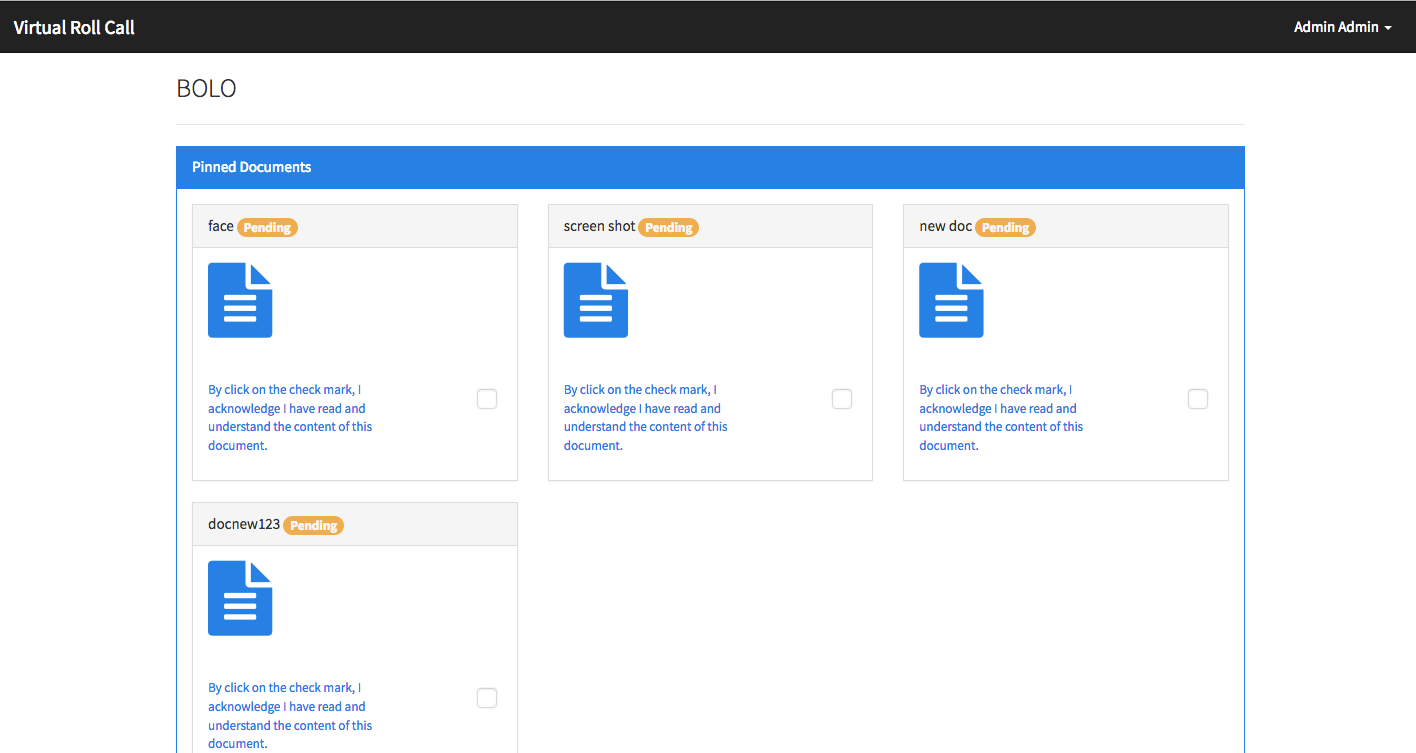


**Figure C.10 – Reset Password Page**

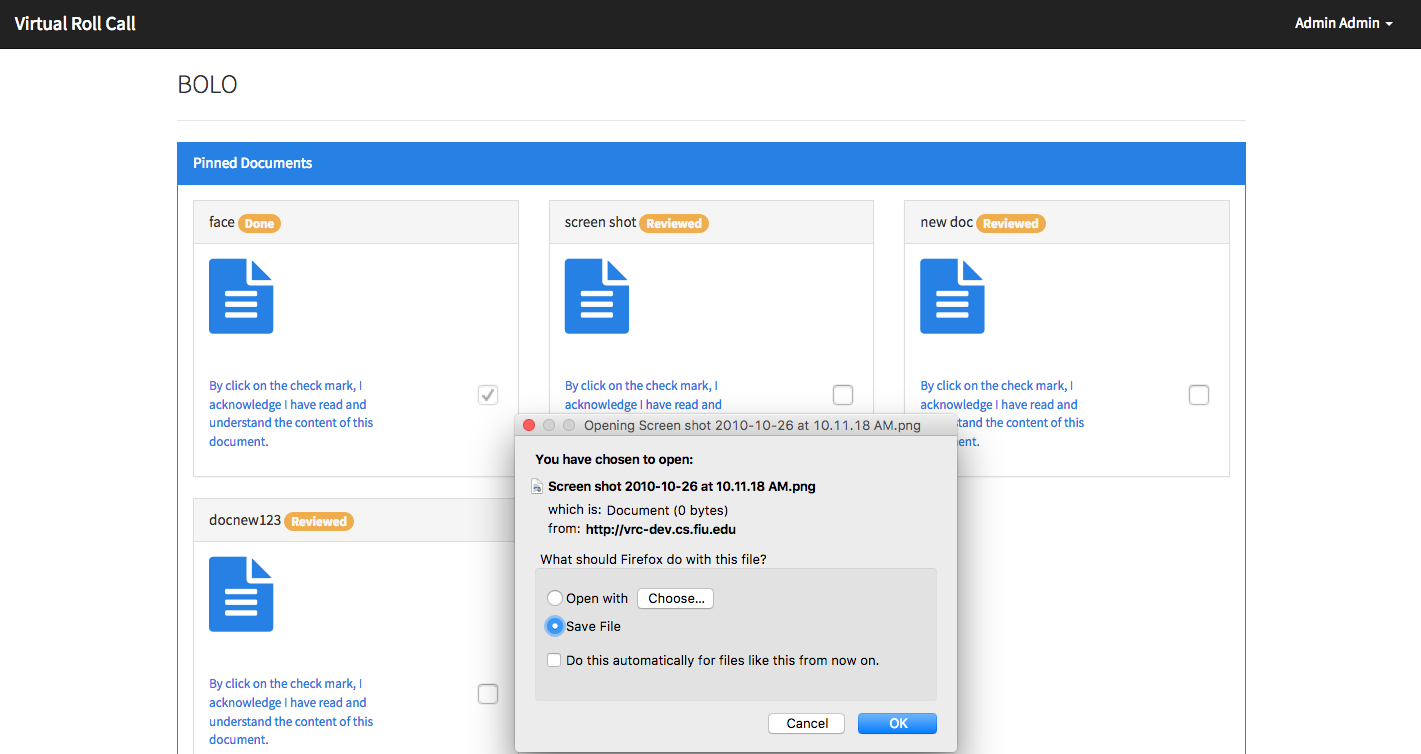


**Figure C.11 – Logs Page**

 **Figure C.12 – Officer Dashboard**



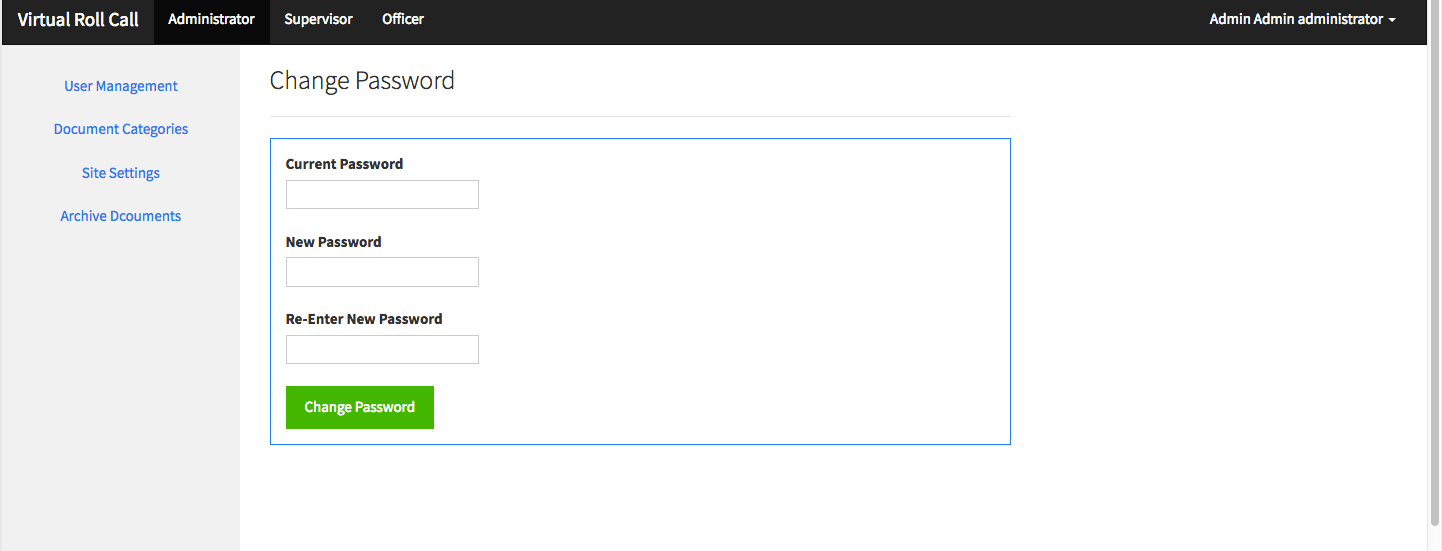
**Figure C.13 – Viewing documents in a category - BOLO in this example.**

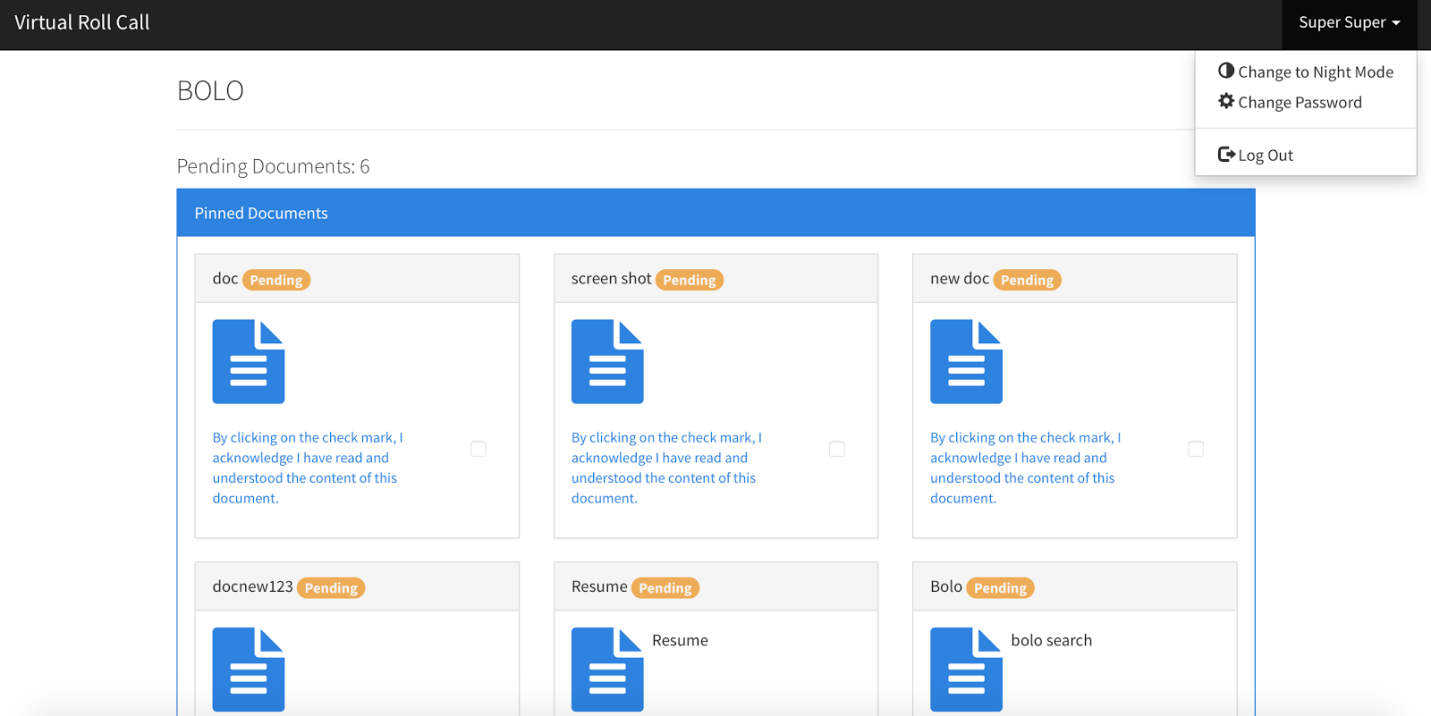


**Figure C.14 – After selecting a “Document” file icon - user is shown with the document or ask to download the document depending if the browser support the document**

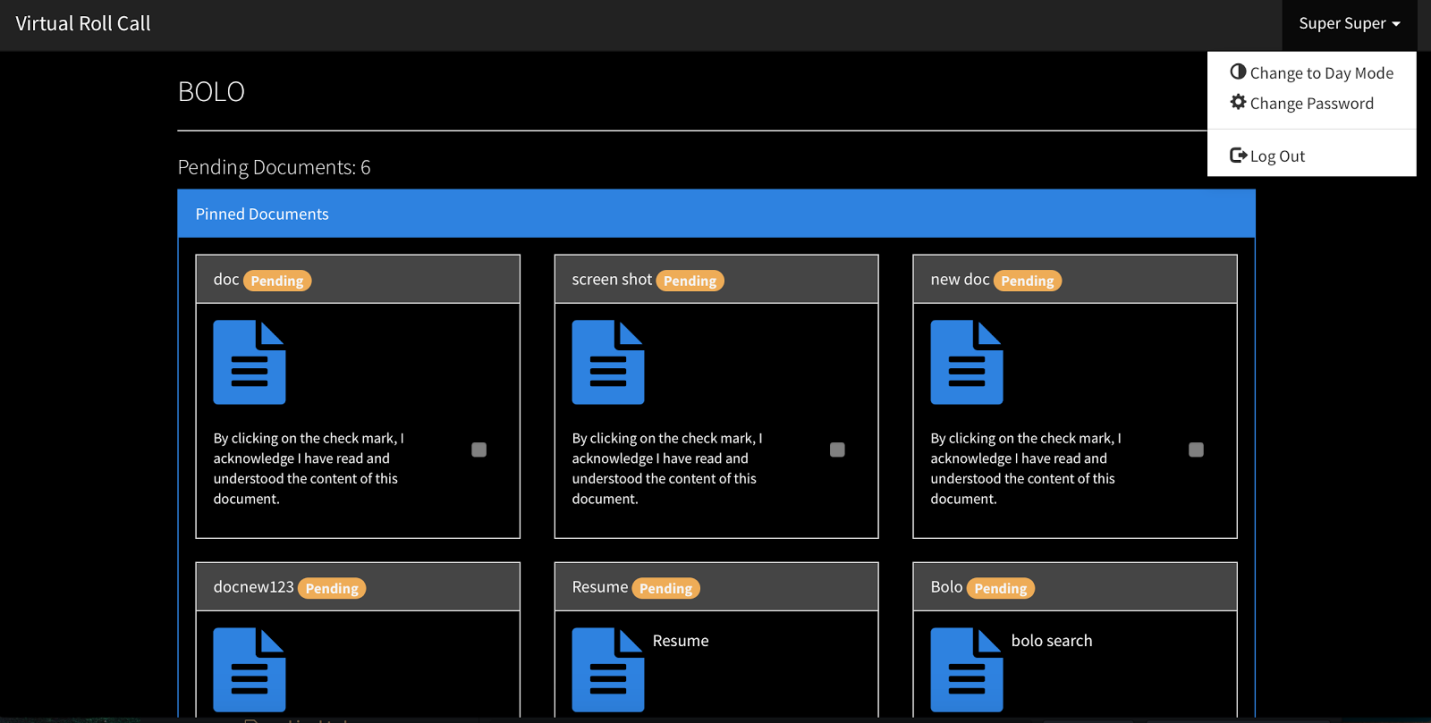


**Figure C.15 – Officer viewing the image in a new tab**

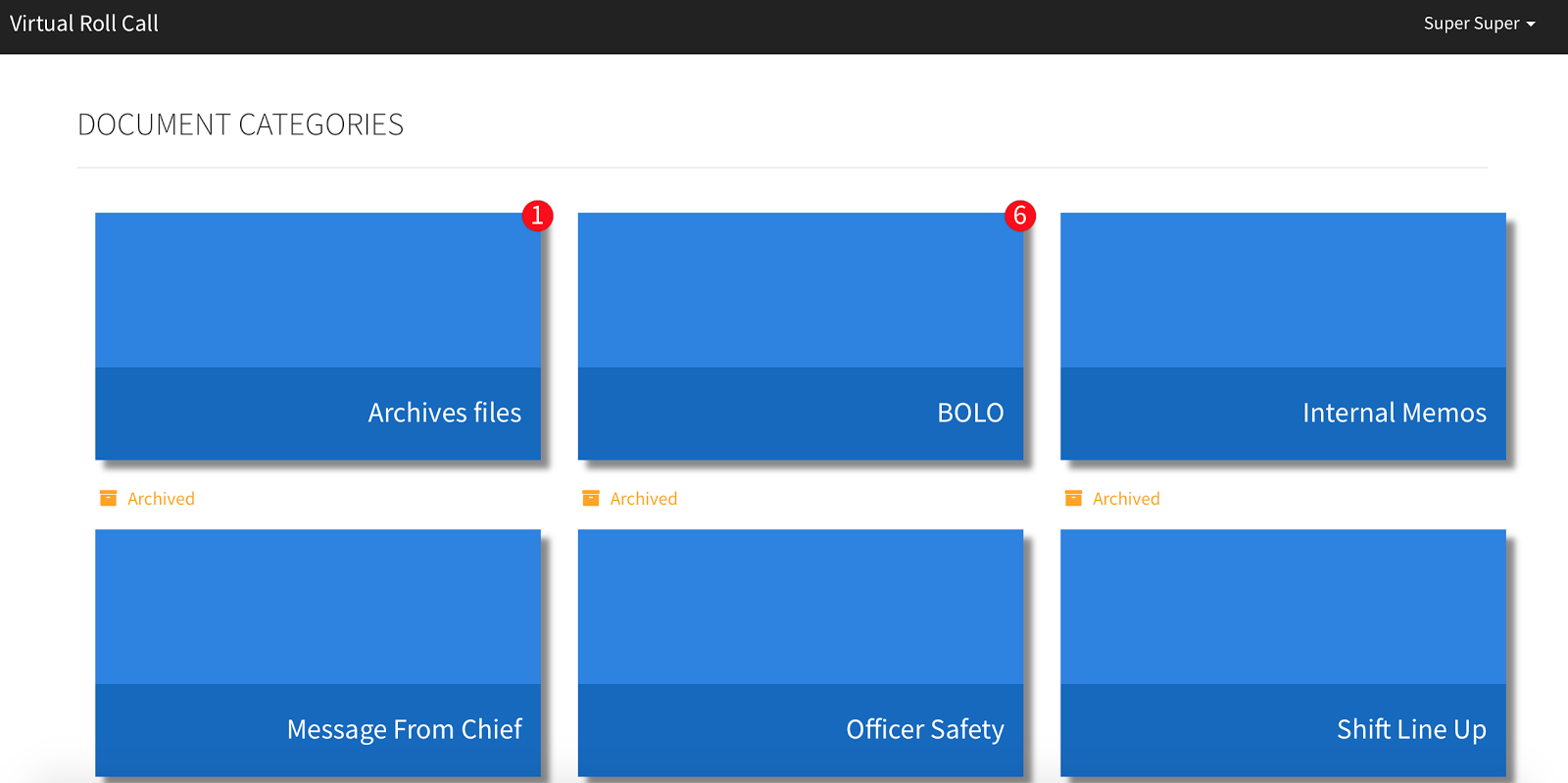
** Figure C.16 – Change Password Page**

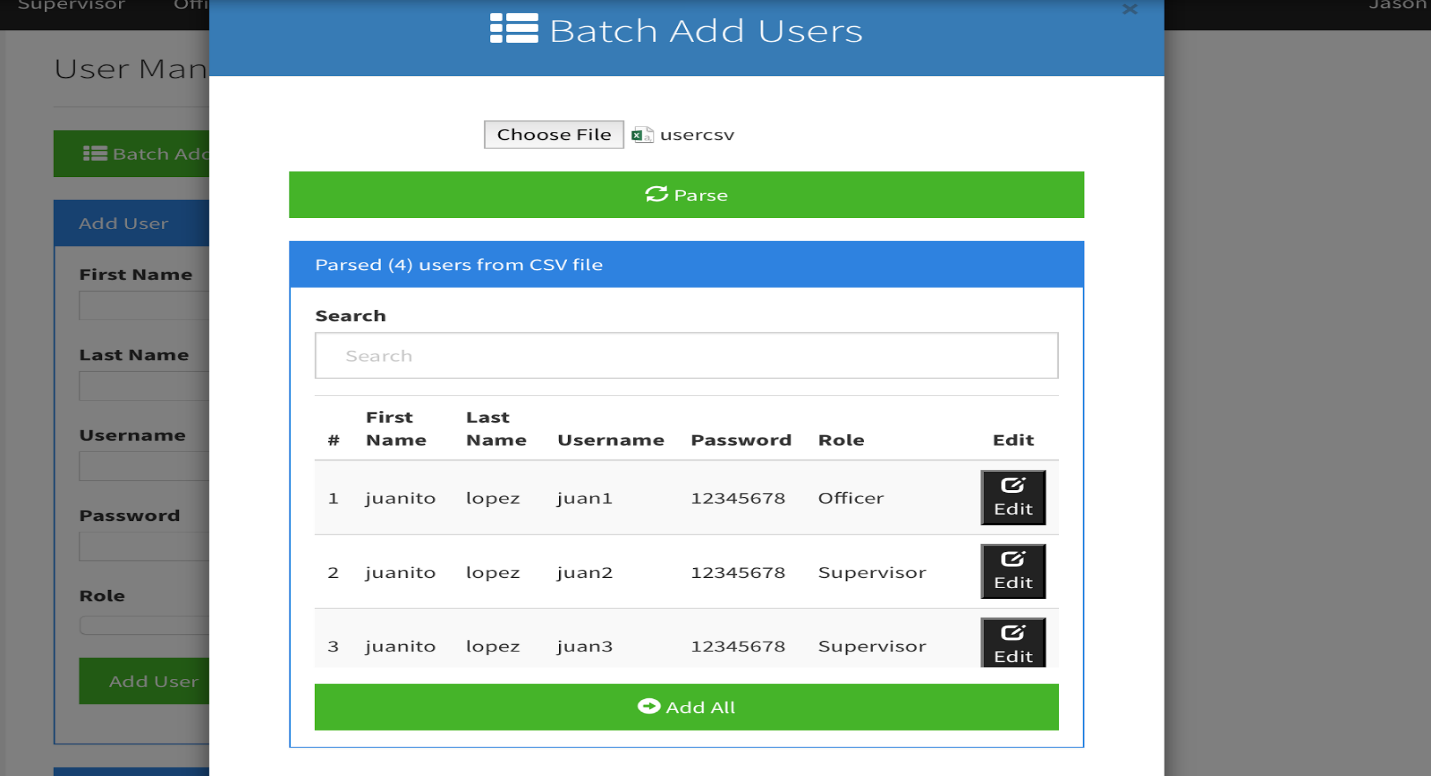
****

**Figure C.17.1 – Day-mode**



**Figure C.17.2 – Night-mode**

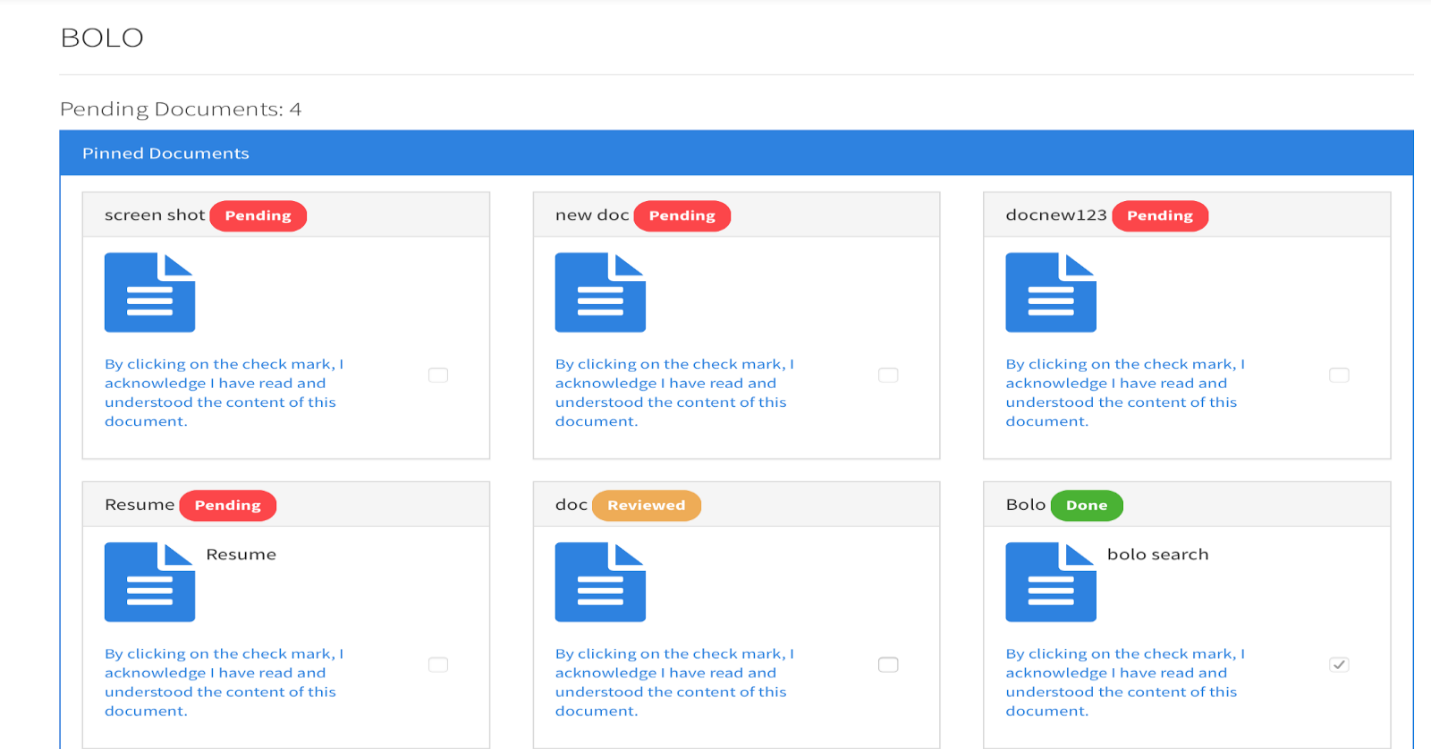
**Figure C.18 – Main Document View showing the number of pending documents per category in the notification badge**



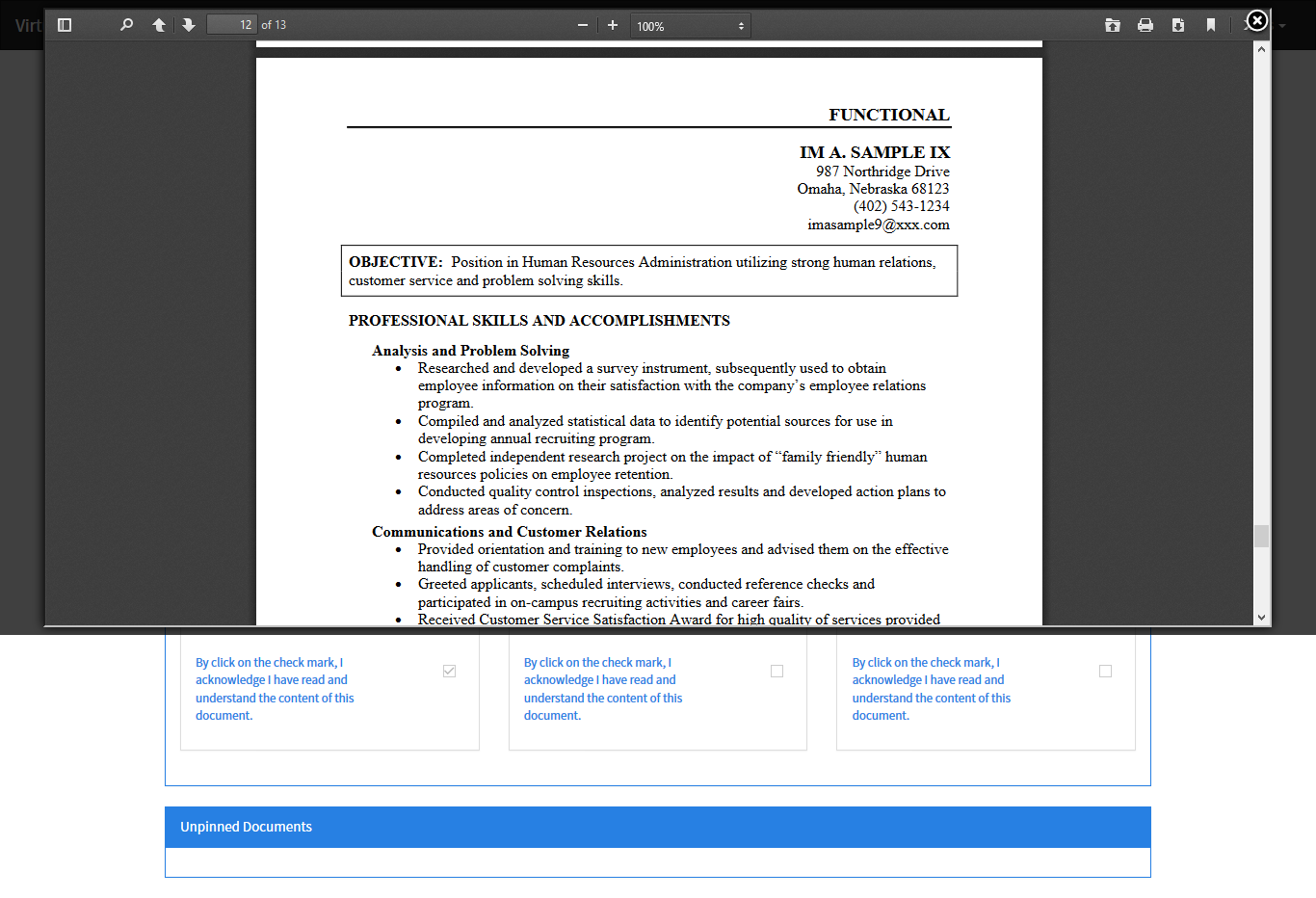
**Figure C.19 – Parsed user table after selecting CSV file and clicking Parse**



**Figure C.20 – After clicking Add All, valid users will be successfully added to database, invalid users will be rejected and will remain in table**

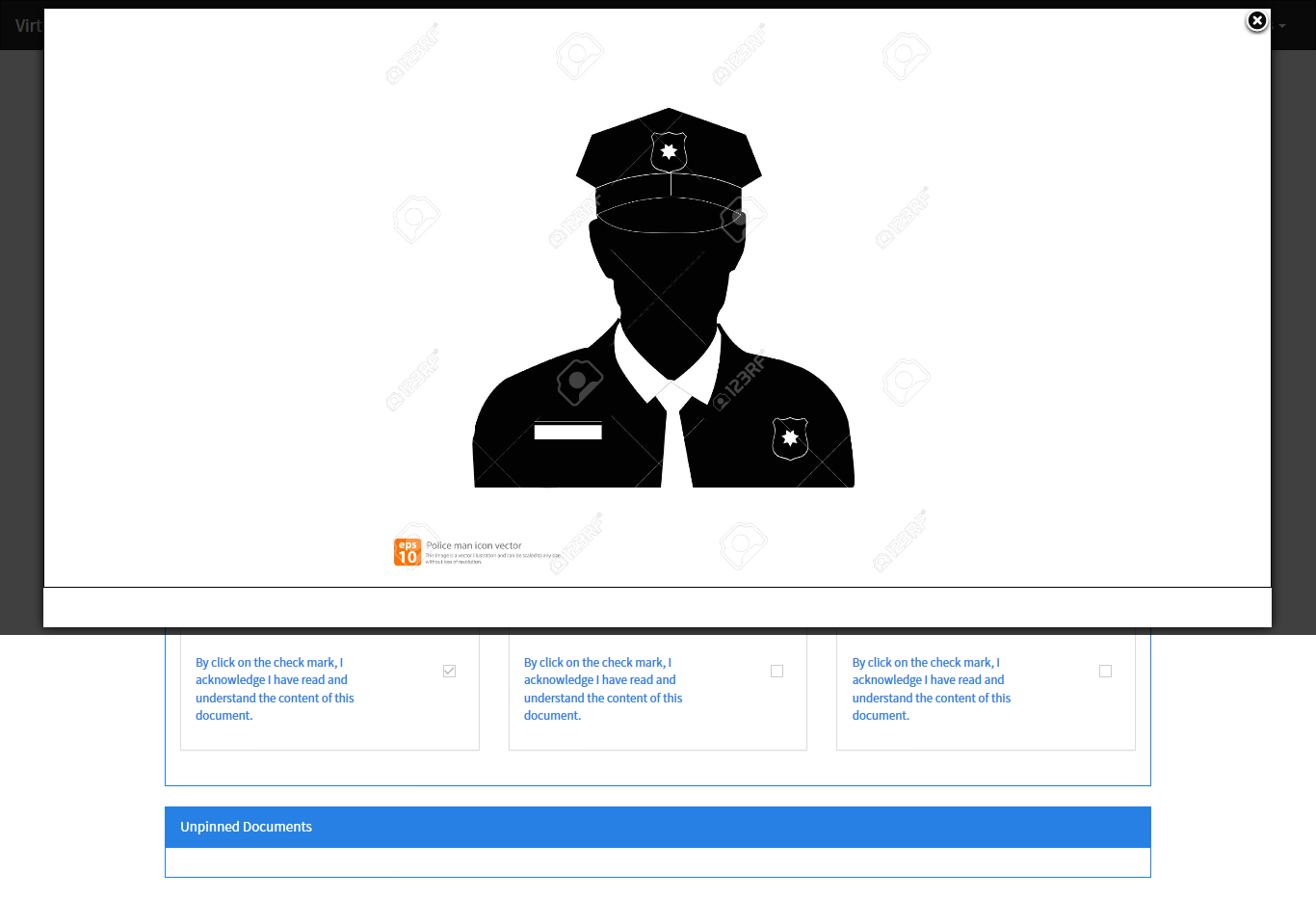


**Figure C.21 – Document list view showing documents ordered by status and with correct status label colors**

****

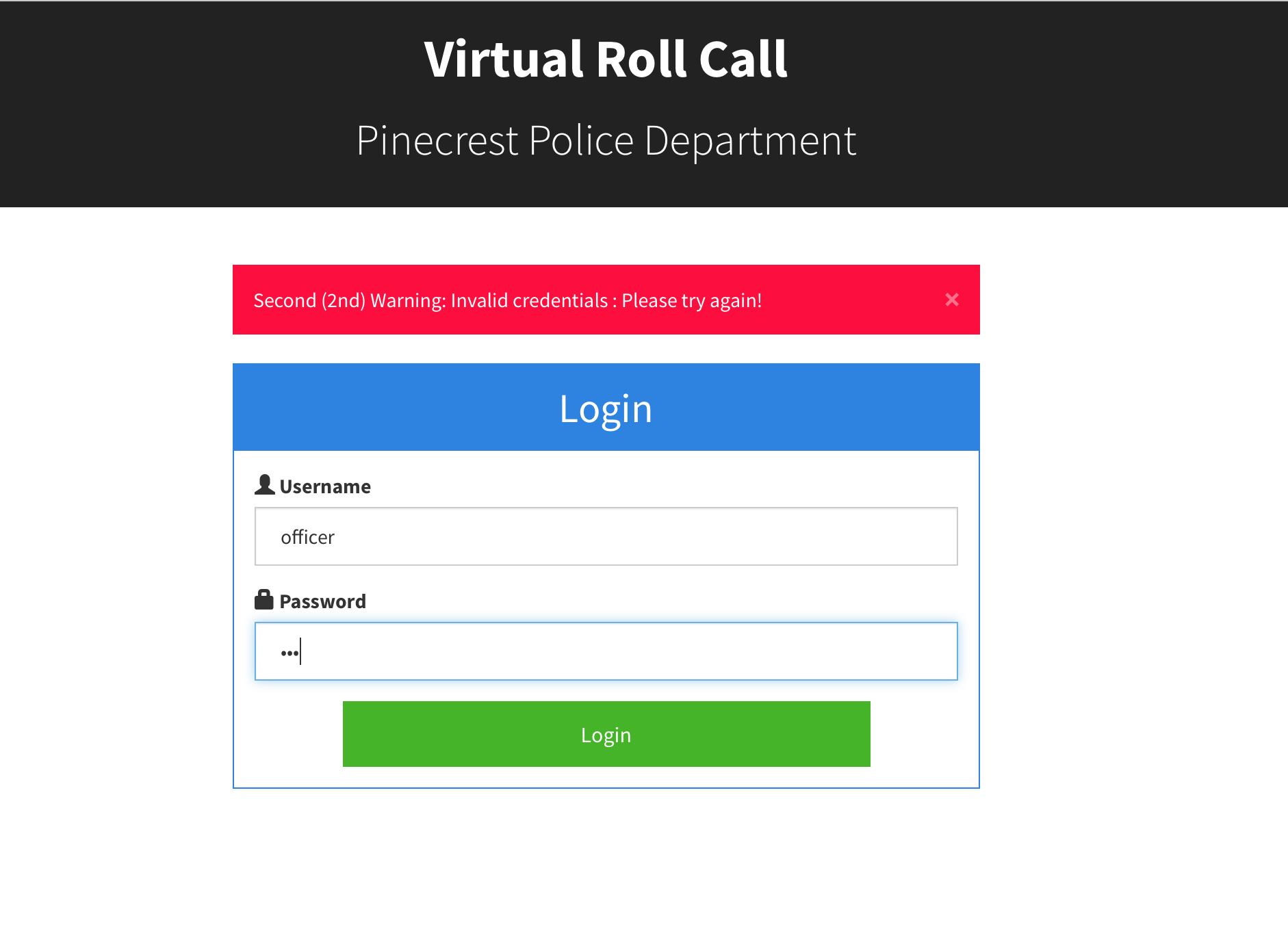
**Figure C.22: View Document (VRC-3002-VD-001)**

**Description:** This UI displays document within a modal window.

****

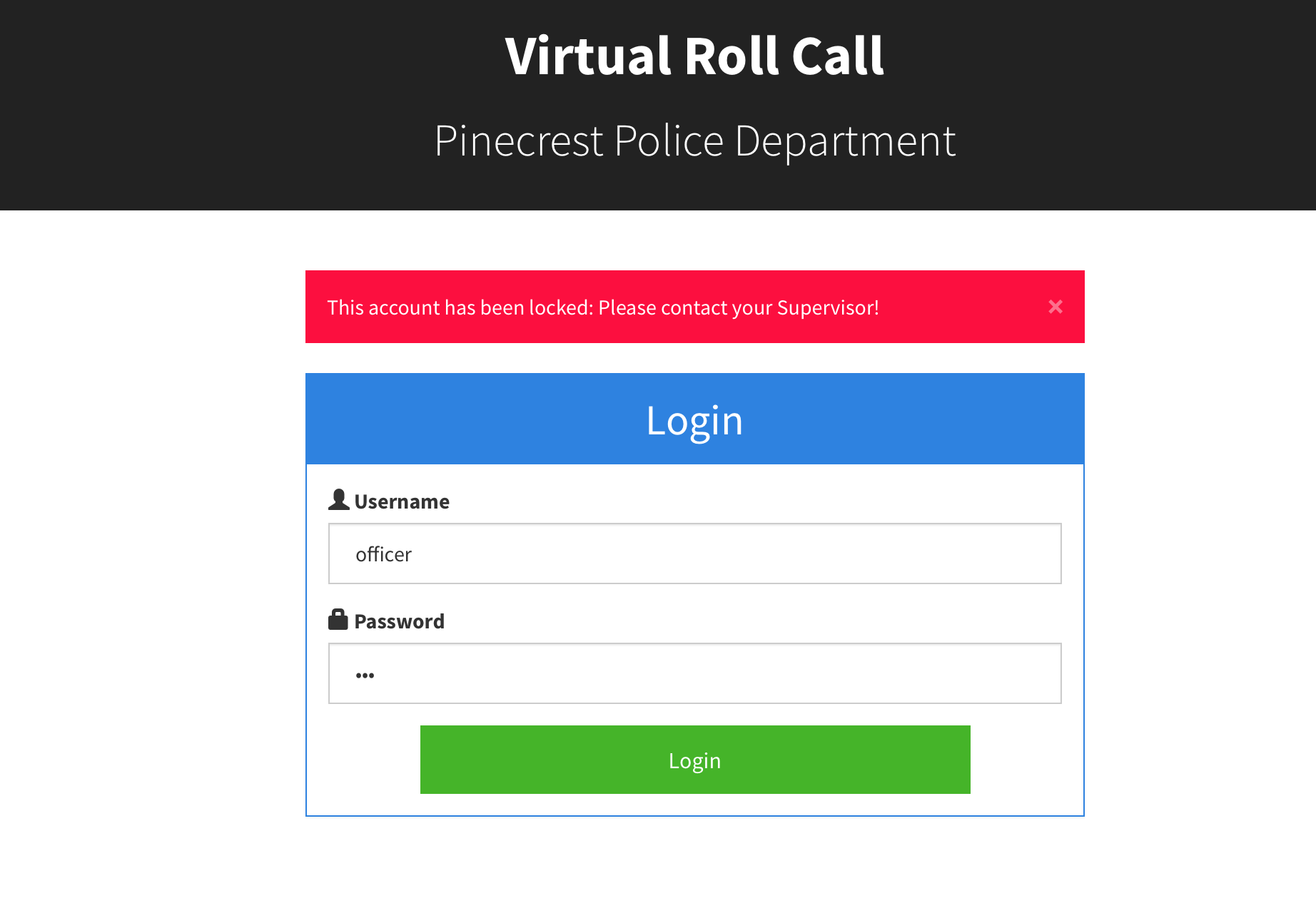
**Figure C.23: View Document (VRC-3002-VD-002)**

**Description:** This UI displays an image within a modal window.



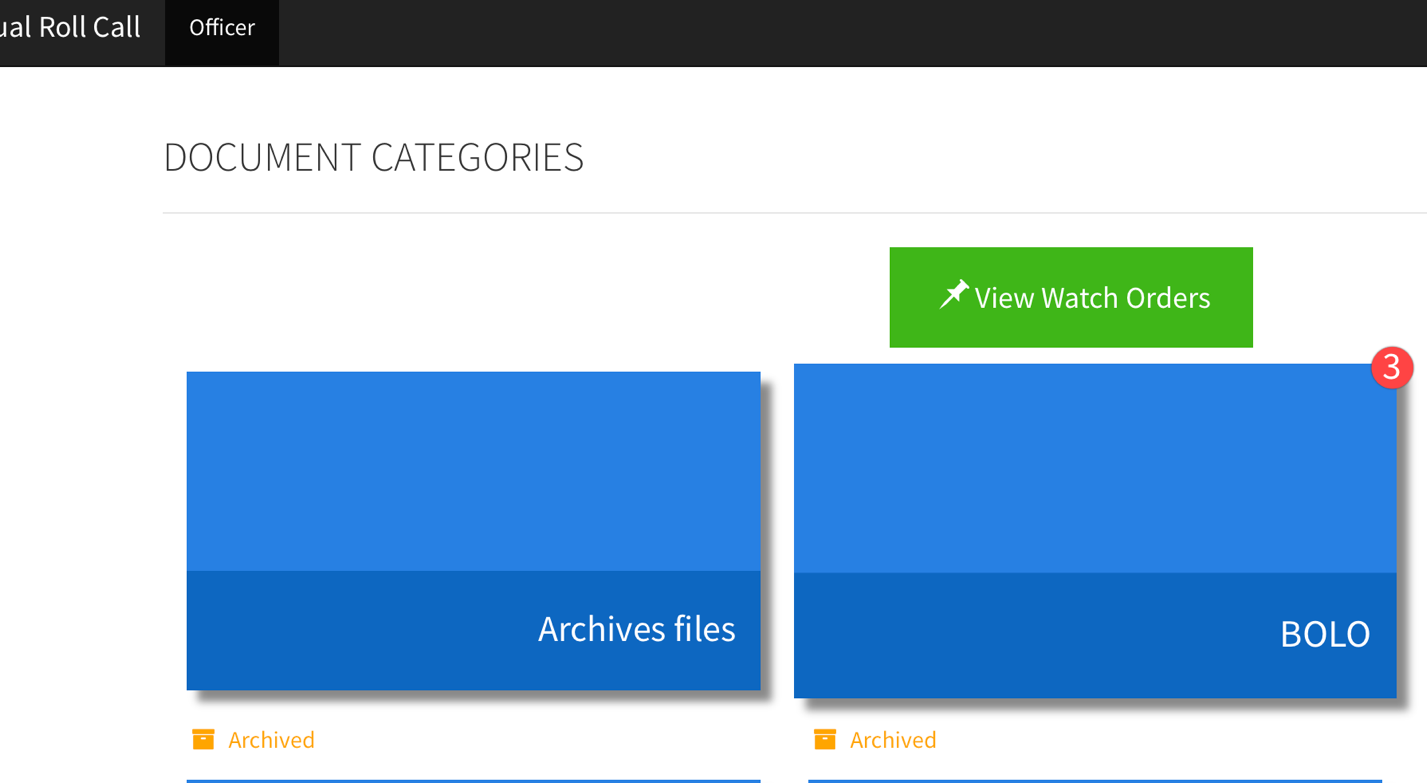
**Figure C.24: Warning for Failed Attempt Screen (VRC-3003 -Lock-147001)**

**Description:** This UI displays a warning message when a user login fails.



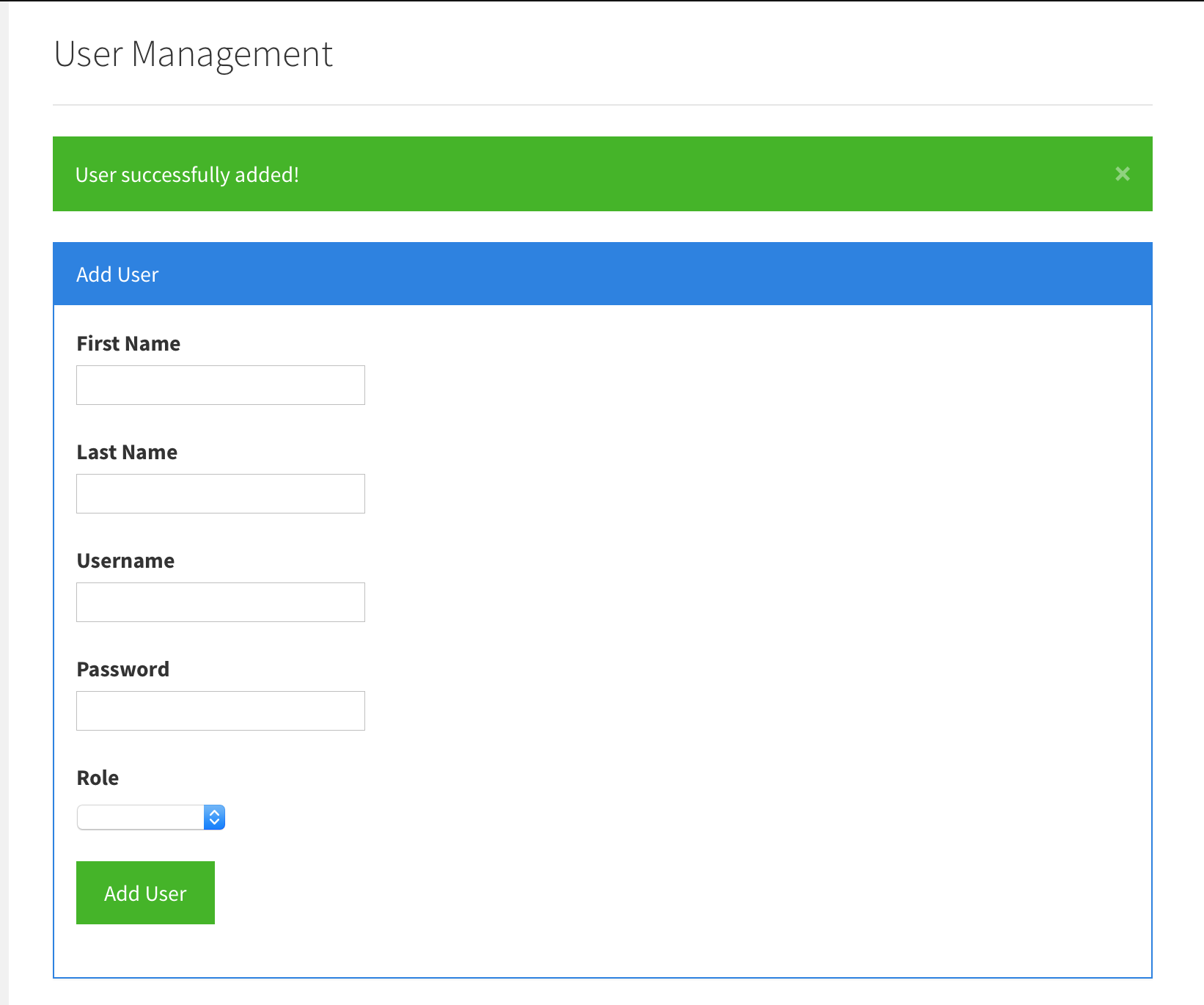
**Figure C.25: Locked Account Screen (VRC-3003 -Lock-147002)**

**Description:** This UI displays after a system locks a user account for failed attempts.



**Figure C.26: Navigate Based on User Role (VRC-4302 – NavigateUserRole)**

**Description:** This UI displays the navigation for Officer role.



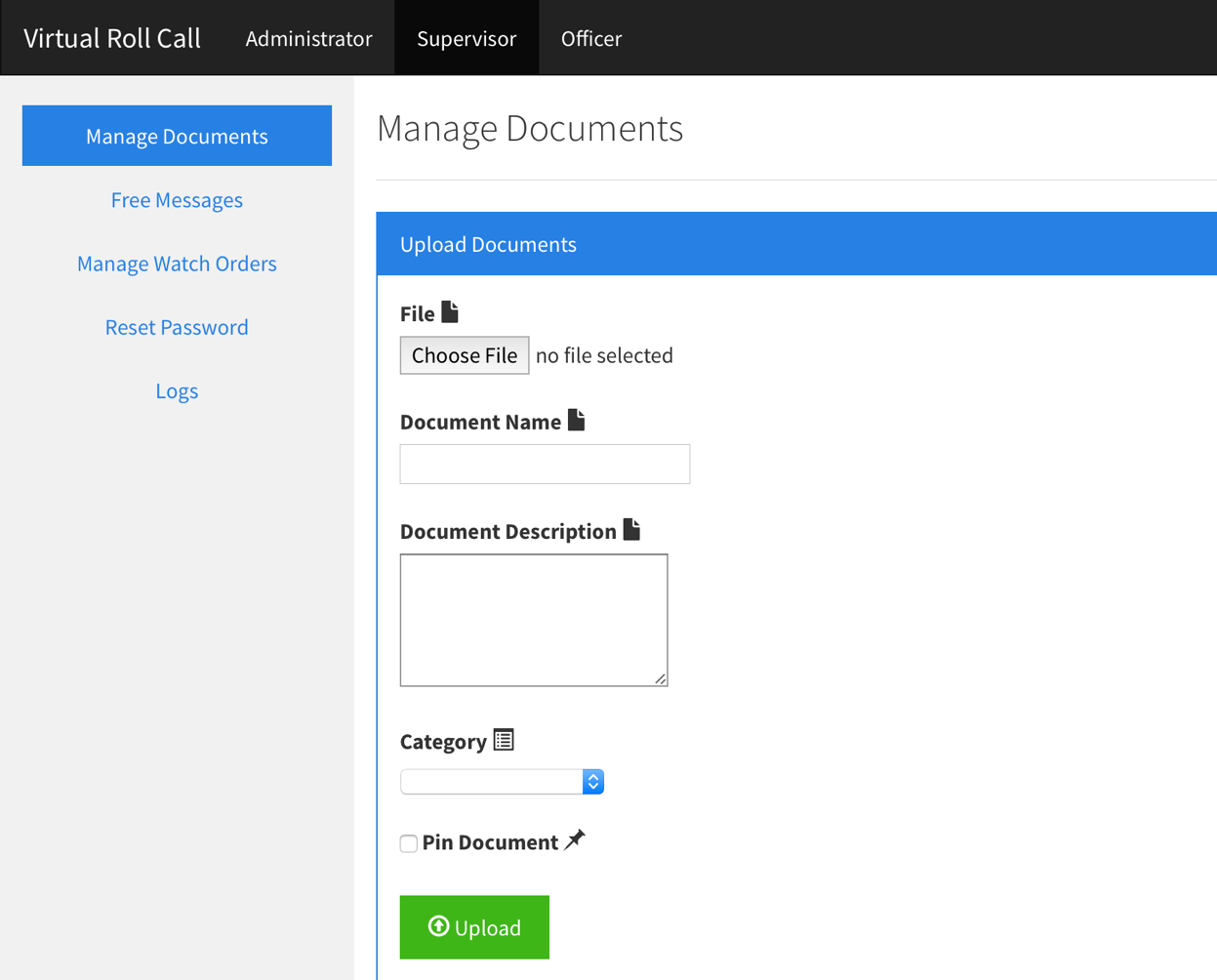
**Figure C.27.1: New User Screen (VRC-3003 – EncryptPassword-001)**

**Description:** This UI allows to add a new user with encrypted password.



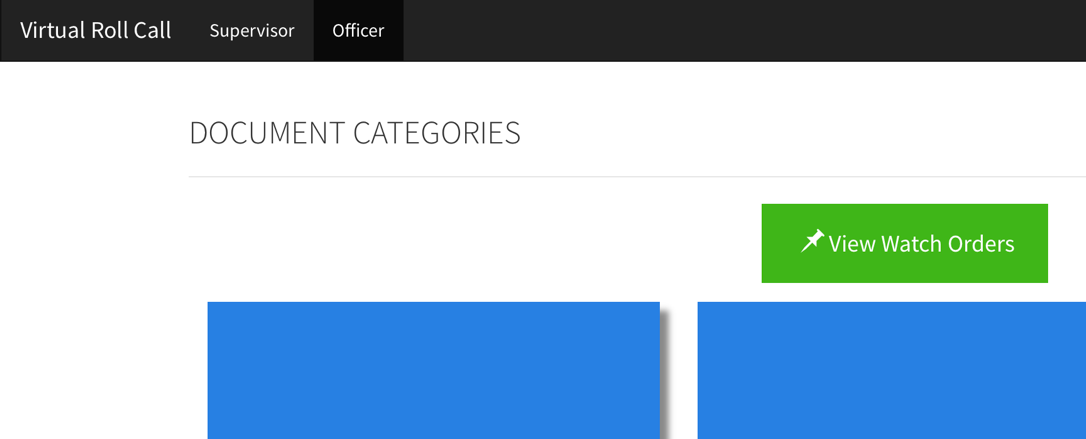
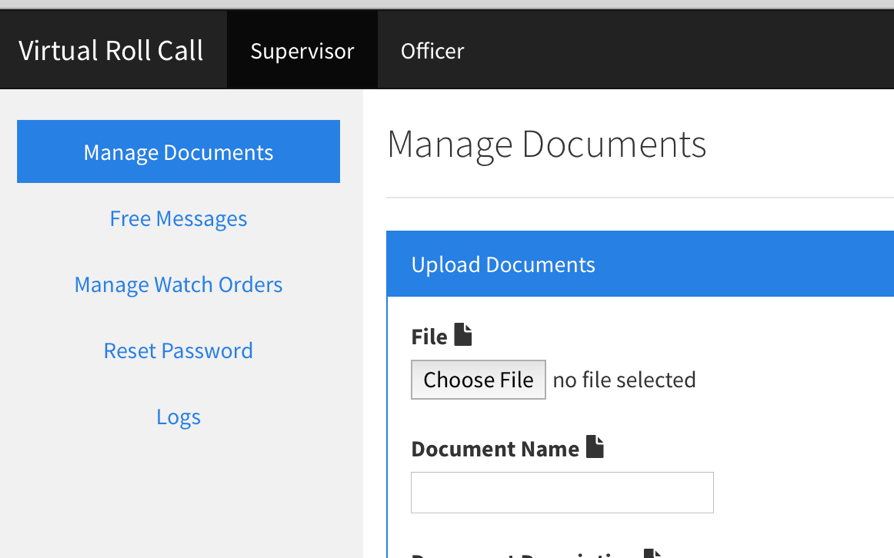
**Figure C.27.2: Database Screenshot (VRC-3003 – EncryptPassword-002)**

**Description:** This UI displays encrypted password in the database.



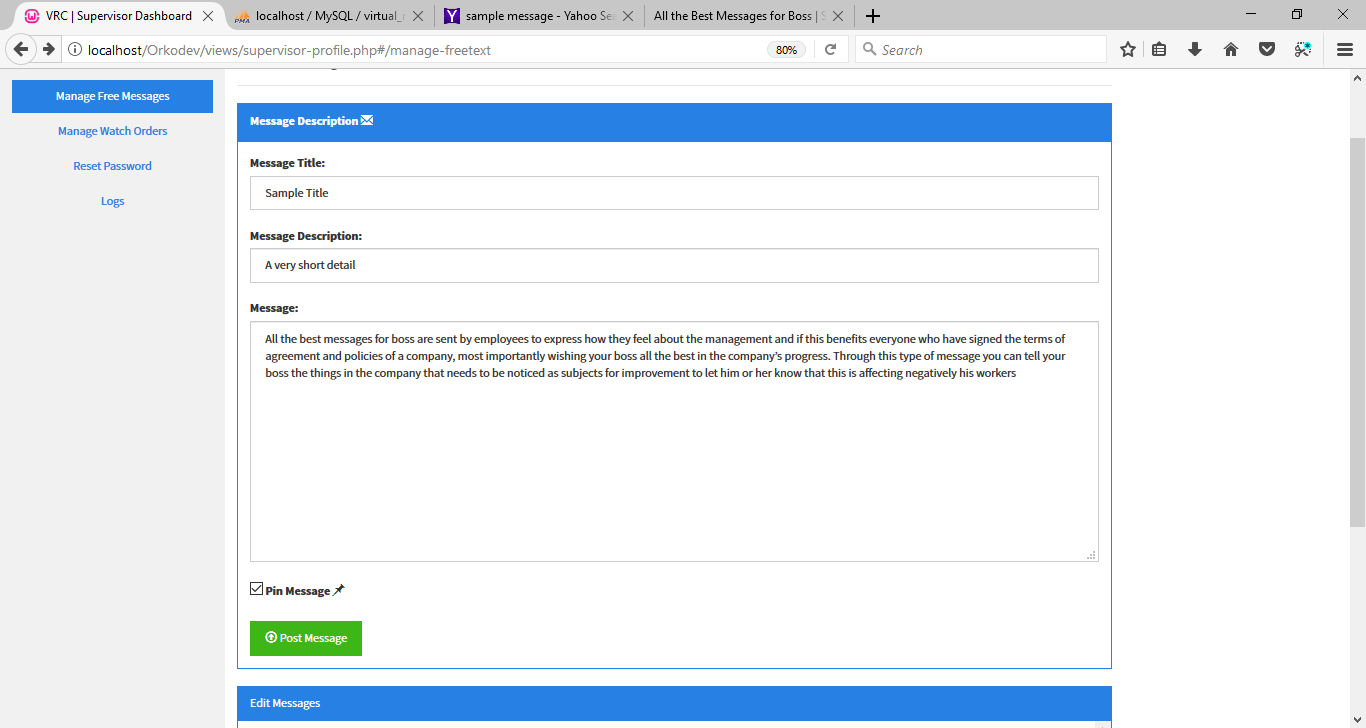
**Figure C.28: Navigate Based on User Role (VRC-4302 – NavigateUserRole)**

**Description:** This UI displays the navigation for Administrator role with option to switch to Supervisor and Officer roles.



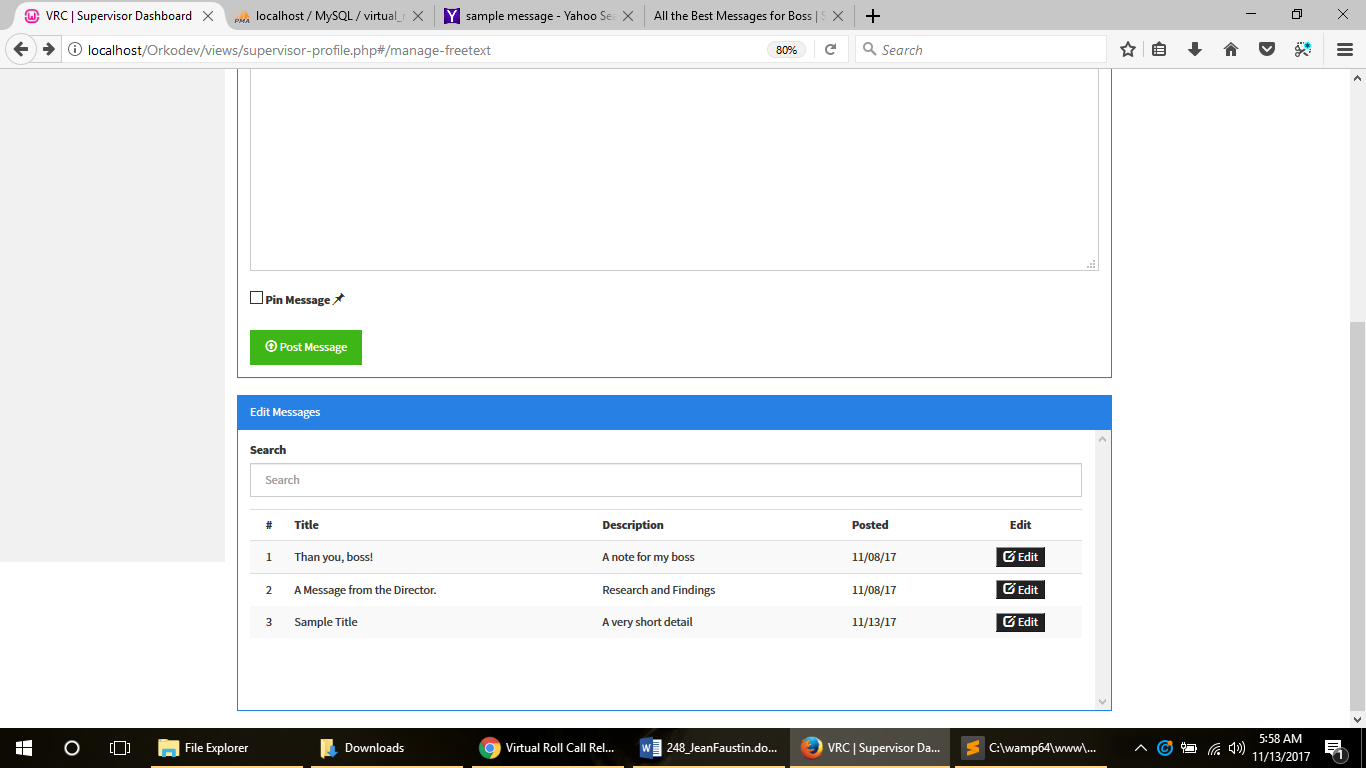
**Figure C.29: Navigate Based on User Role (VRC-4302 – NavigateUserRole)**

**Description:** This UI displays the navigation for Supervisor role with option to switch to an Officer role.



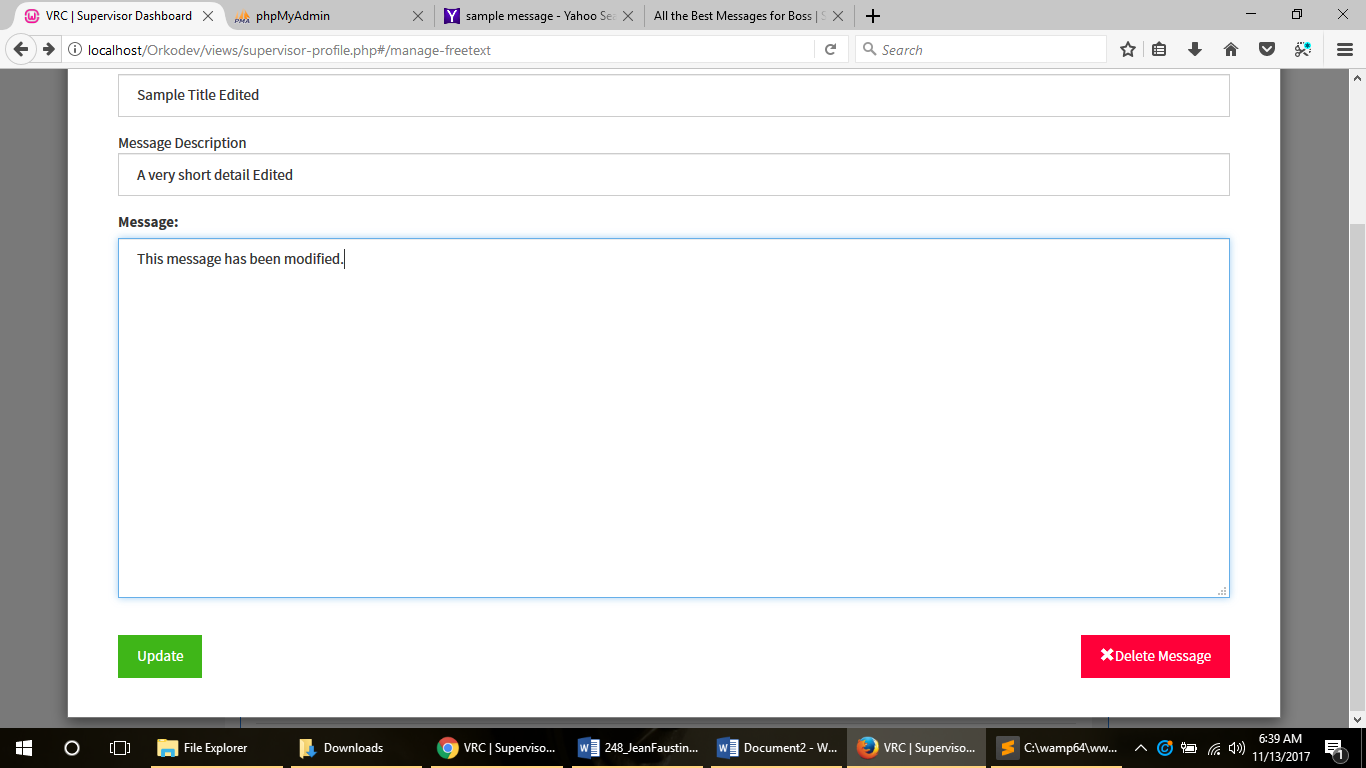
**Figure C.30.1: Add Free-Form Text (VRC-3005 – AddFreeText)**

**Description:** This UI allows a Supervisor to add free-form text messages.



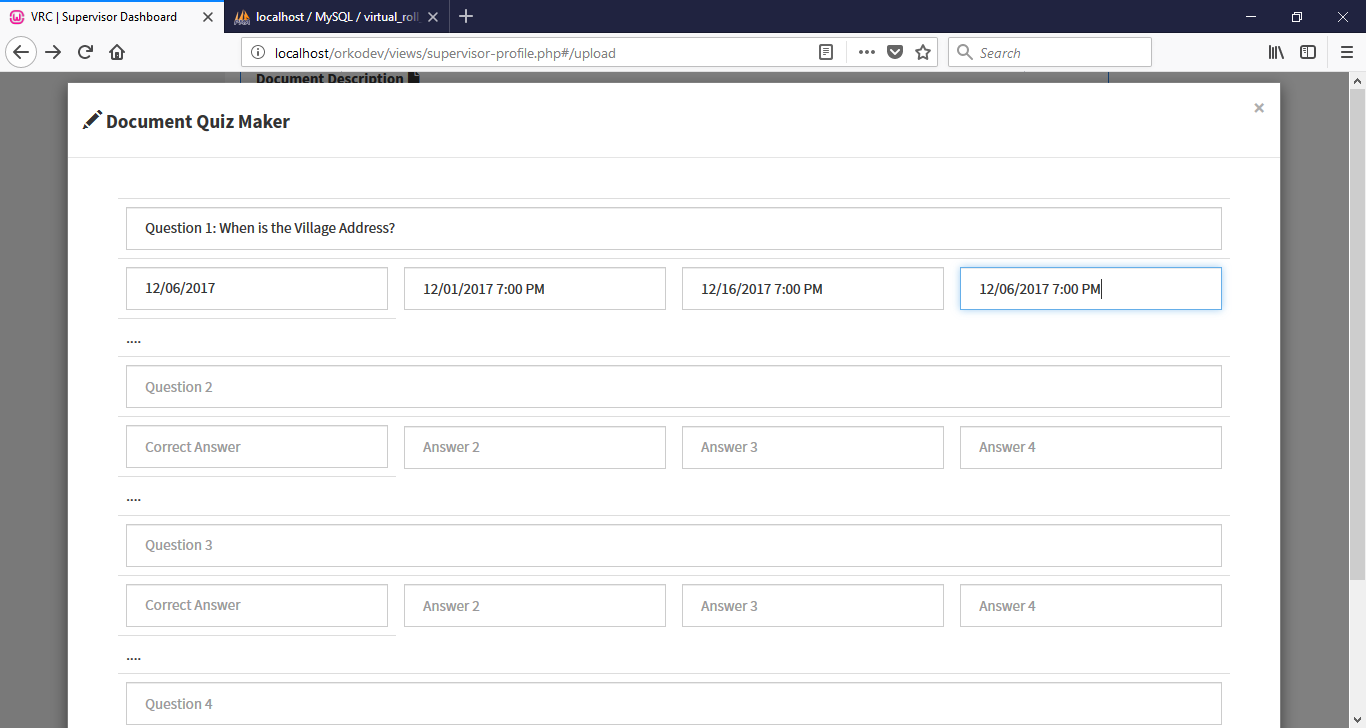
**Figure C.30.2: Messages List (VRC-3005 – AddFreeText)**

**Description:** This UI displays the added messages to the table list.



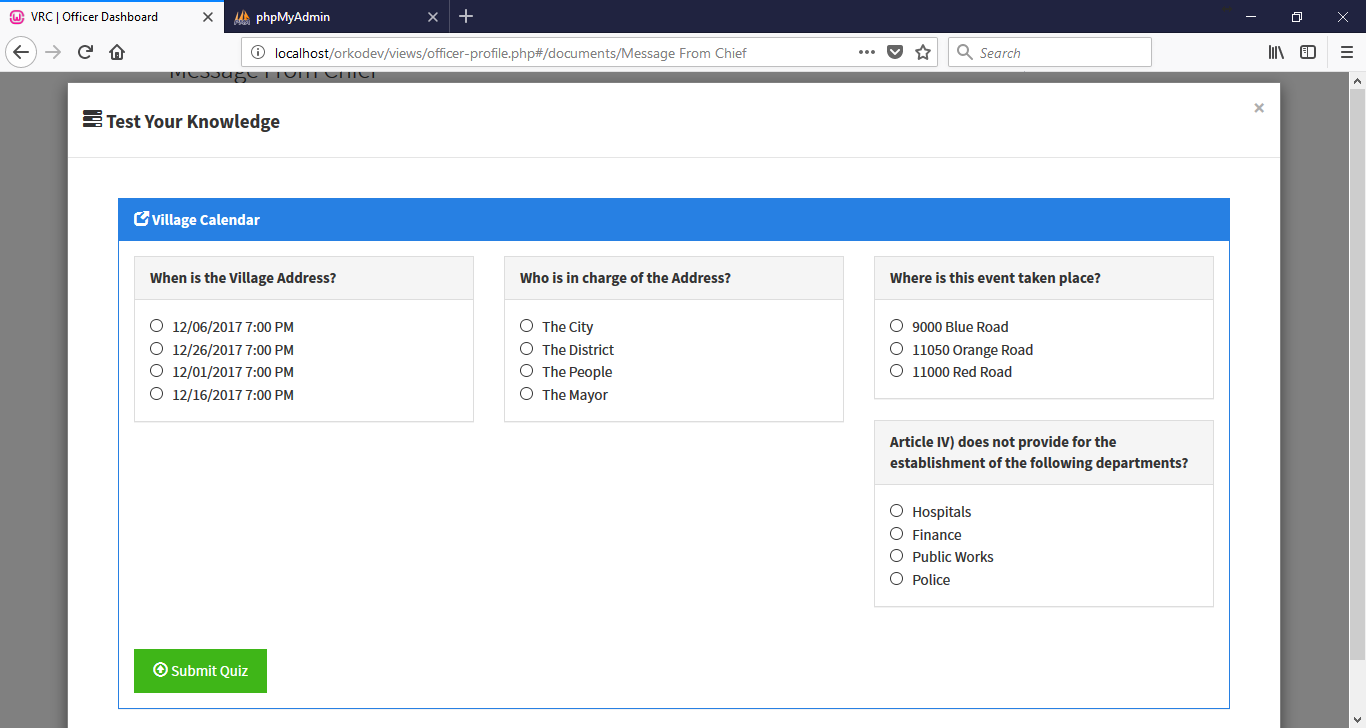
**Figure C.31: Edit Fee Text (VRC-3005 – EditFreeText)**

**Description:** This UI will be used to edit or delete a free-form text message.



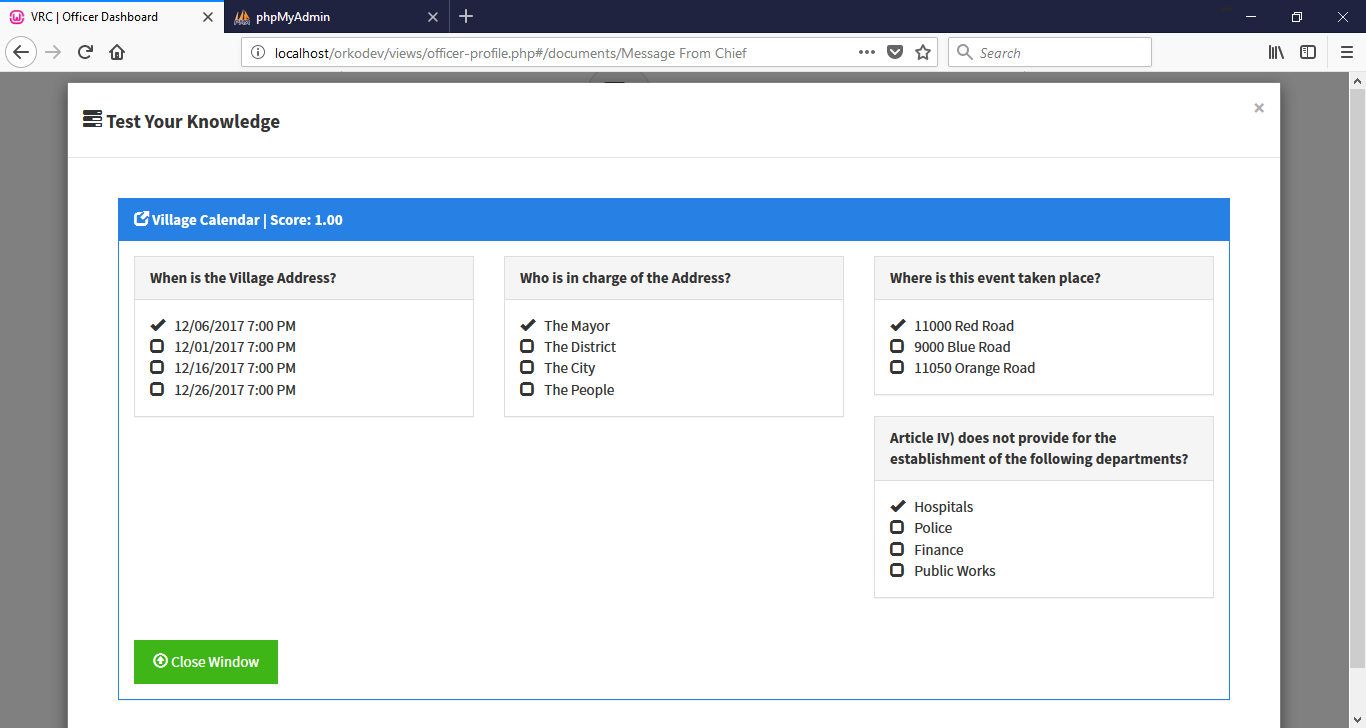
**Figure C.32: Quiz Maker UI (VRC-3005 – EditFreeText)**

**Description:** This UI allows a quiz to be created up to five questions.



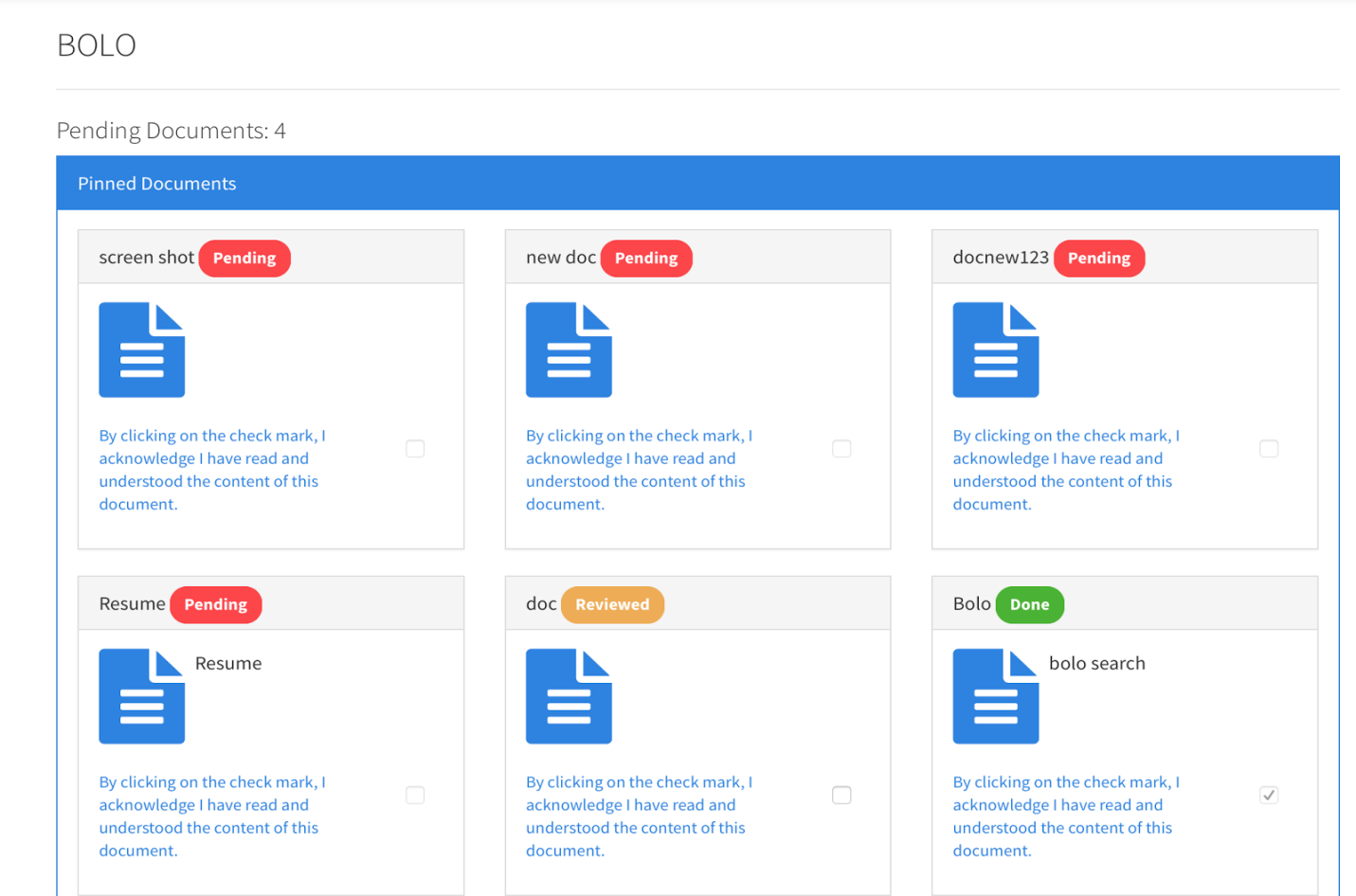
**Figure C.33.1: Take Quiz (VRC-3005 – TakeQuiz)**

**Description:** This UI allows a user to take a quiz to review a document.



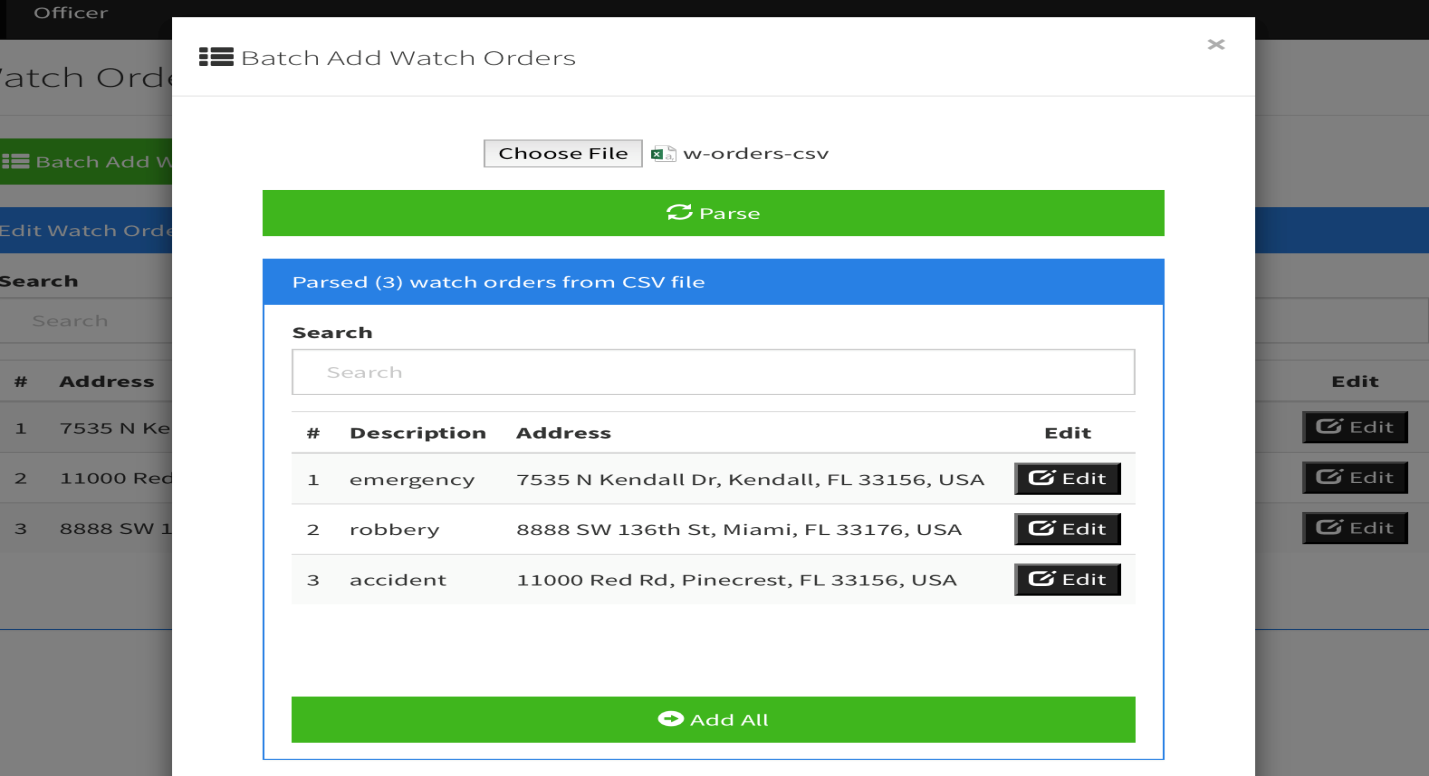
**Figure C.33.2: Take Quiz (VRC-3005 – TakeQuiz)**

**Description:** This UI displays the answers to a quiz.



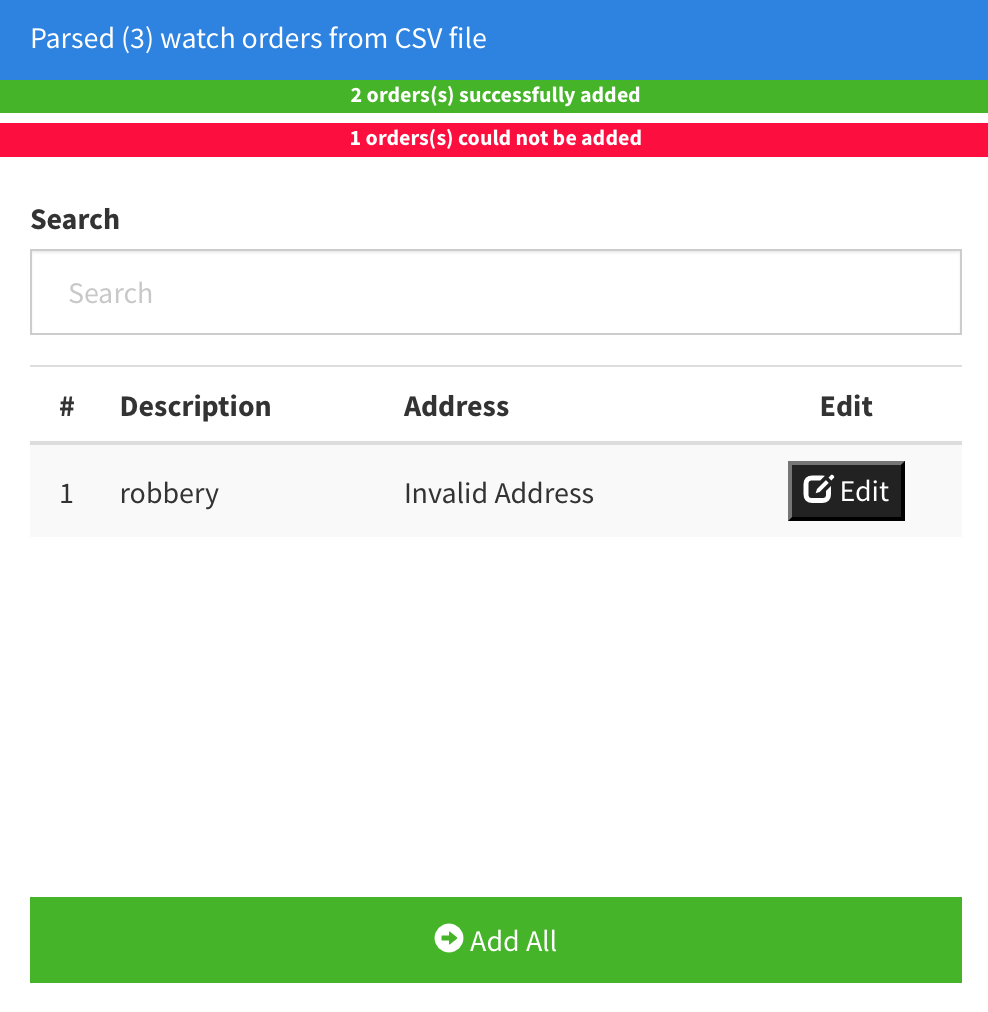
**Figure C.34: Improve Document List Presentation (#300)**

**Description:** Document list view showing documents ordered by status and with correct status label colors.



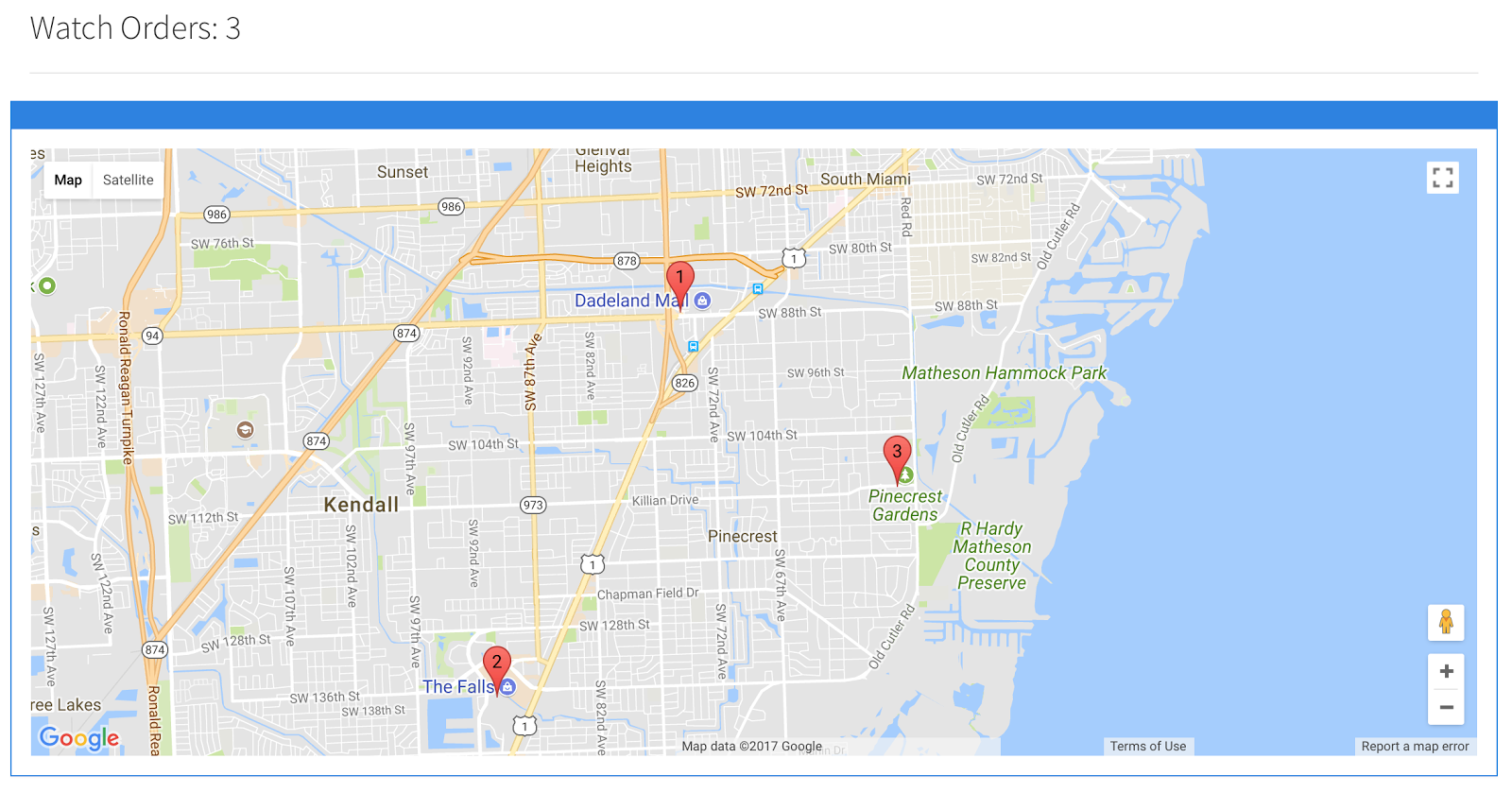
**Figure C.35.1: Add Watch Orders with CSV File (#246)**

**Description:** Parsed watch order table after selecting CSV file and clicking Parse

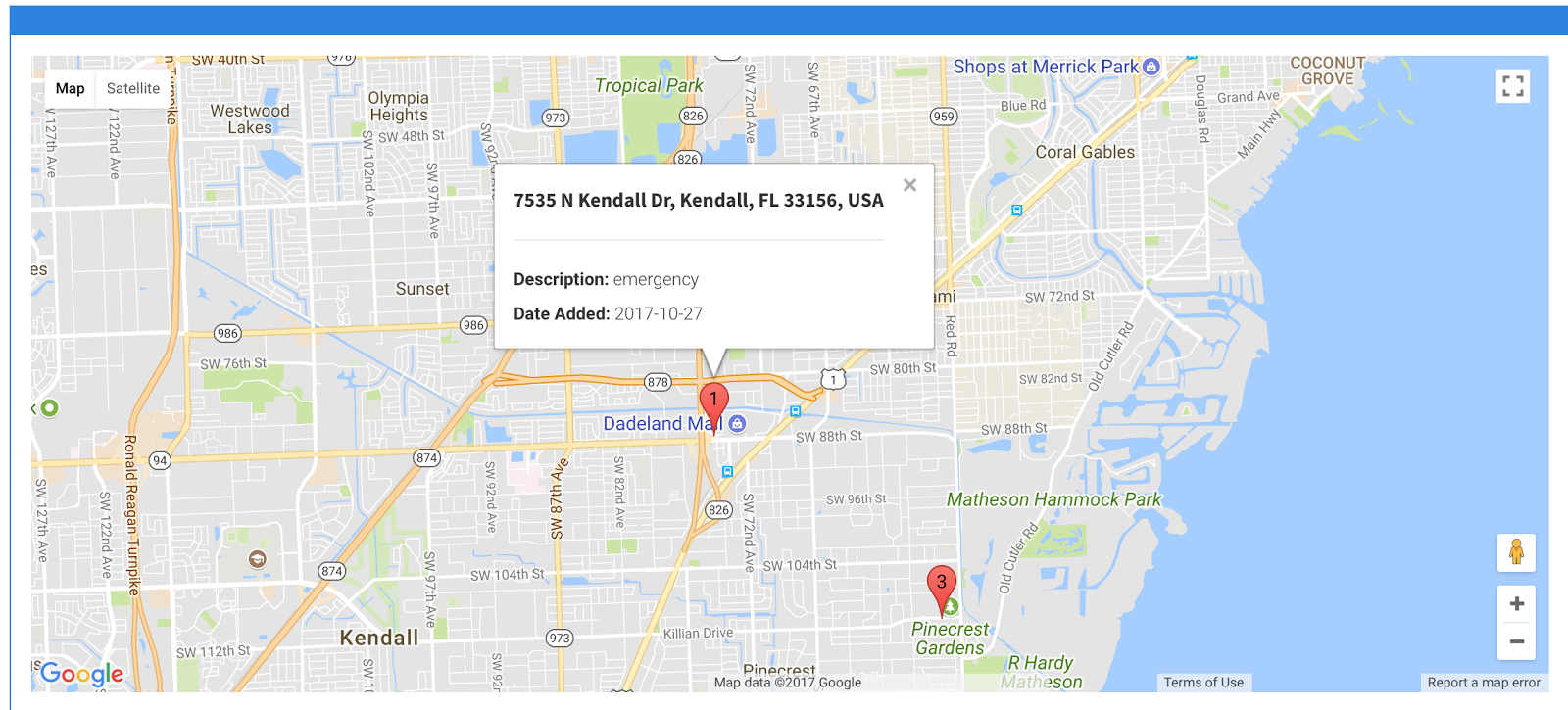


**Figure C.35.2: Add Watch Orders with CSV File (#246)**

**Description:** After clicking Add All, valid watch orders will be successfully added to database. Invalid orders will be rejected and will remain in the table.

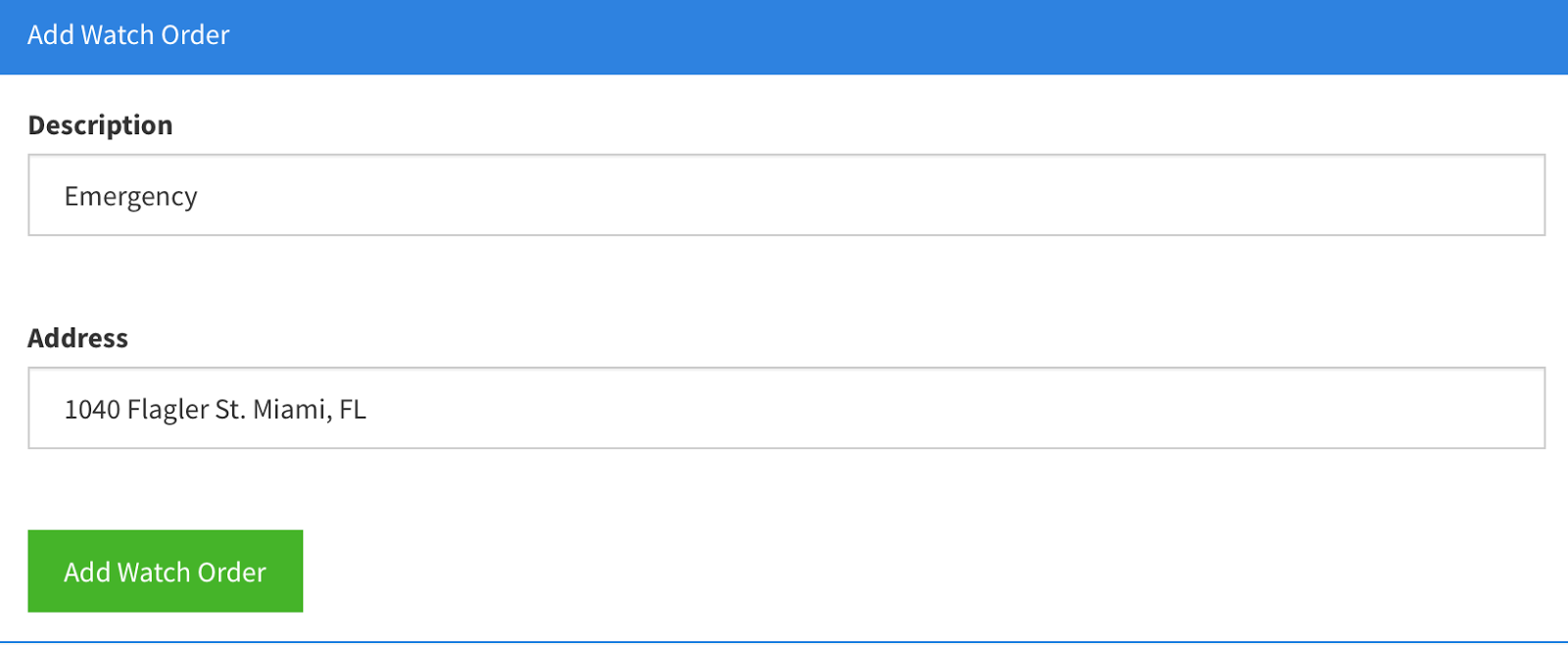
**Figure C.36.1: View Watch Orders in a Map (#247)**

**Description:** The watch order map showing the markers for the watch orders in the database.



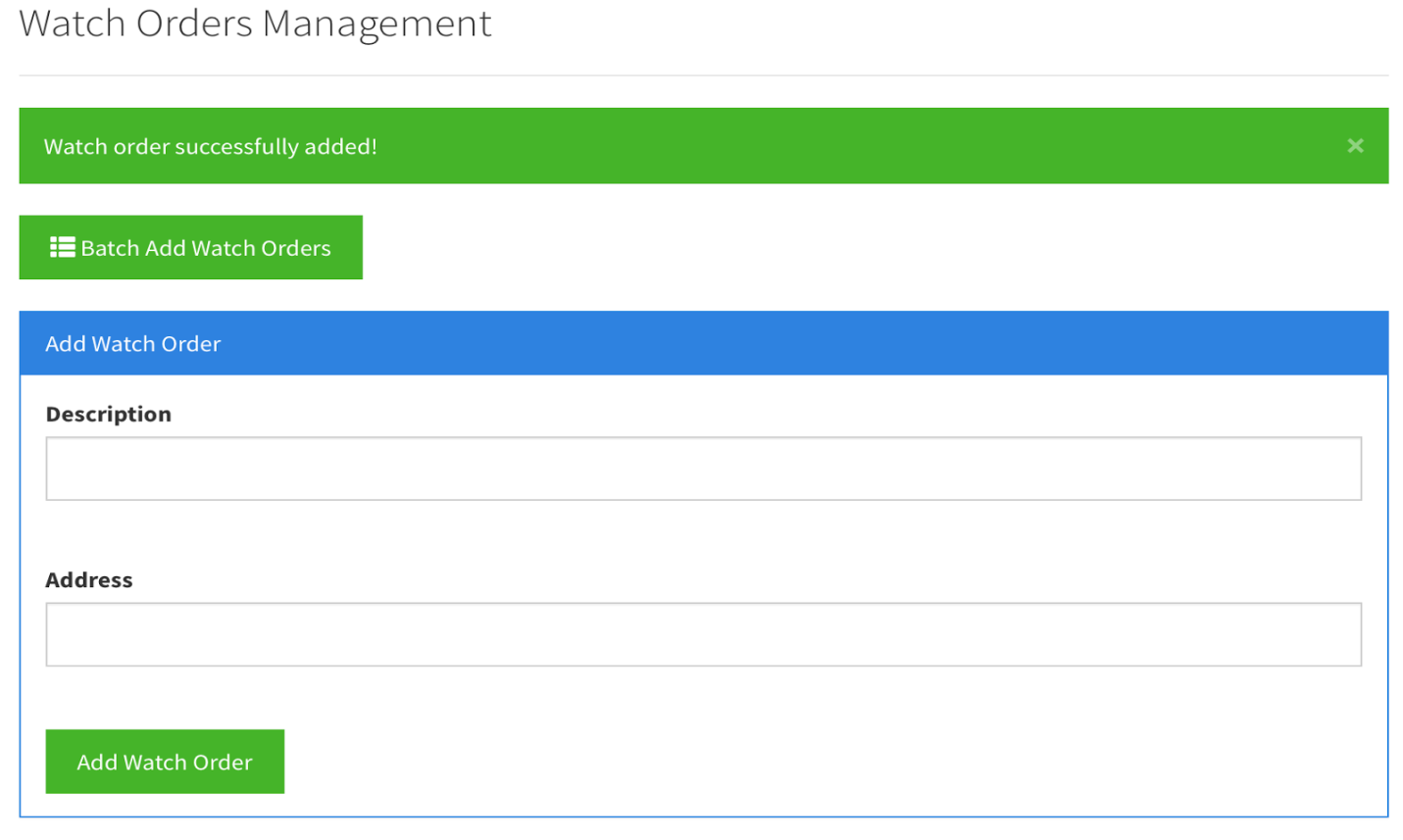
**Figure C.36.2: View Watch Orders in a Map (#247)**

**Description:** The watch order map showing the information window for a selected marker.

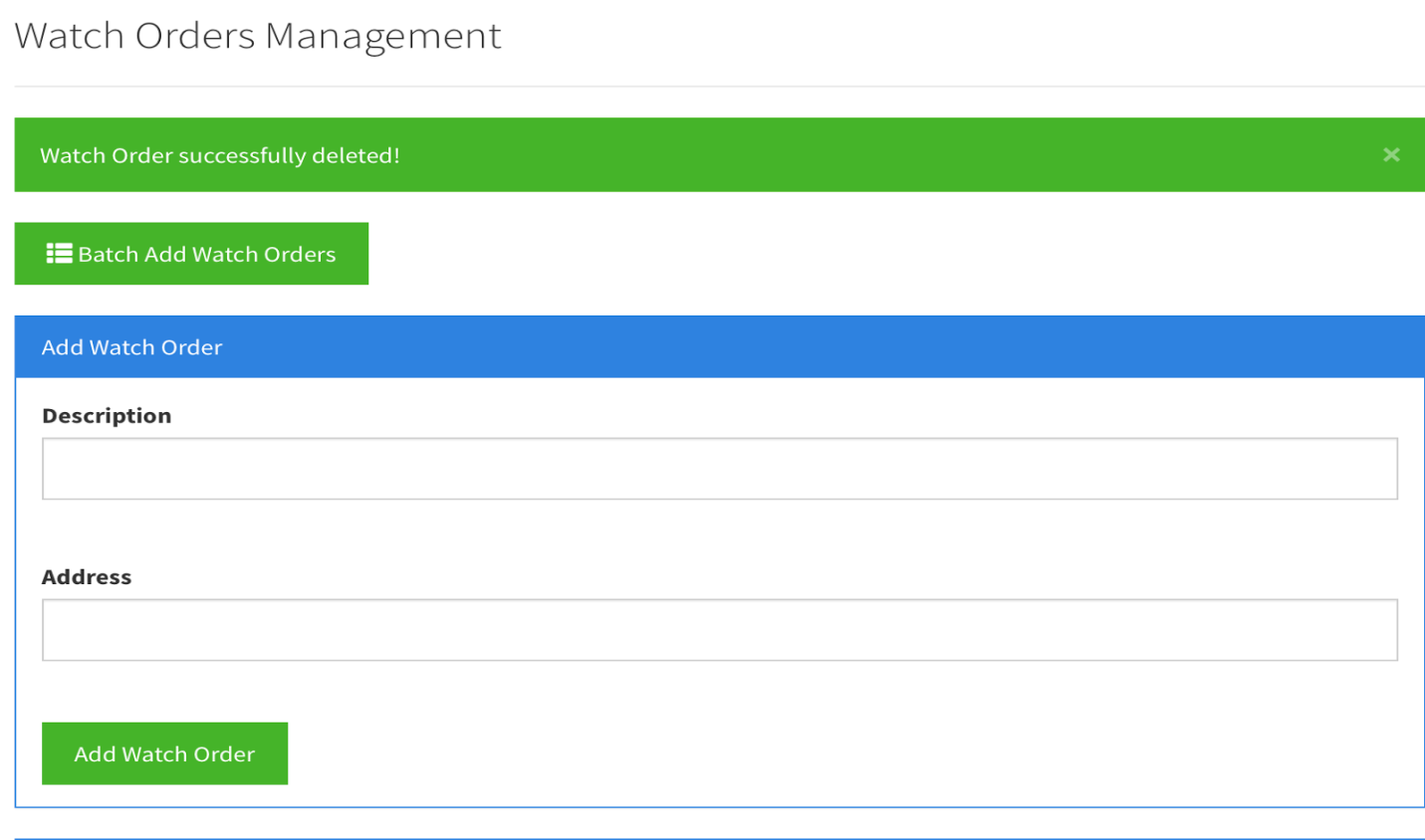


**Figure C.37.1: Add Watch Order (#304)**

**Description:** Add Watch Order Panel

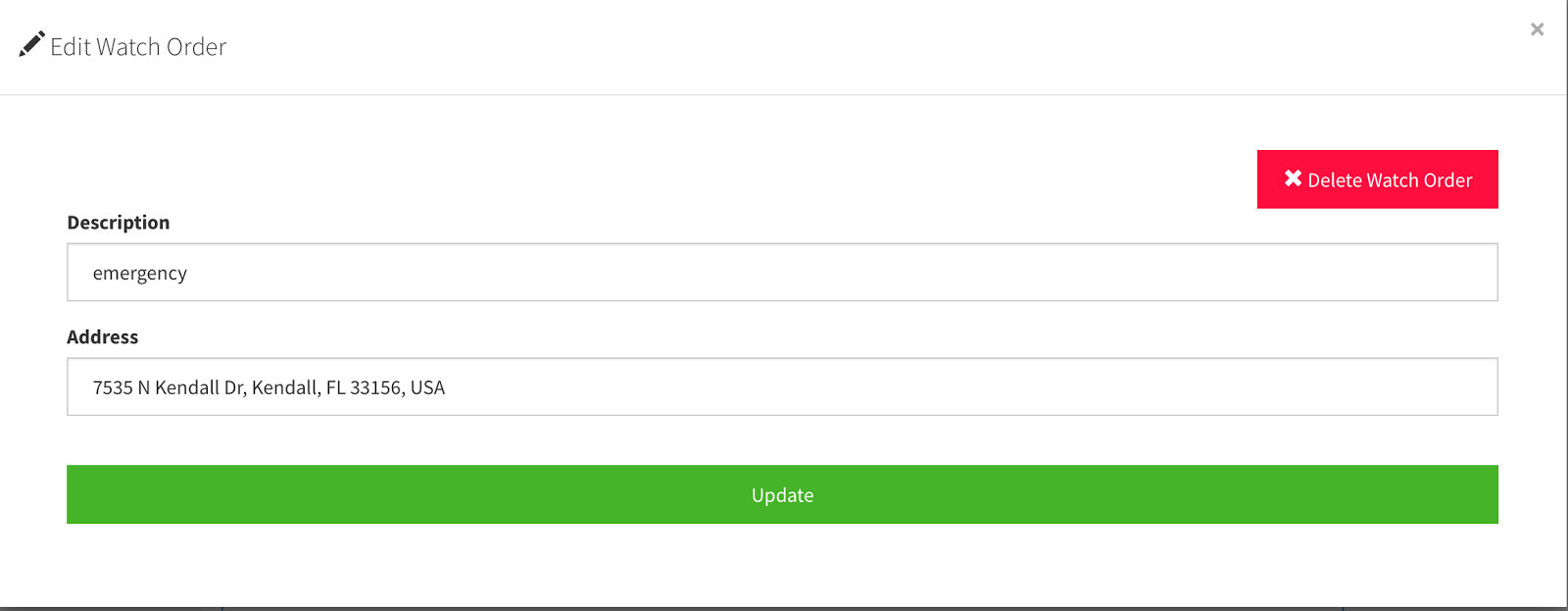
**Figure C.37.2: Add Watch Order (#304)**

**Description:** Success message after adding valid watch order.



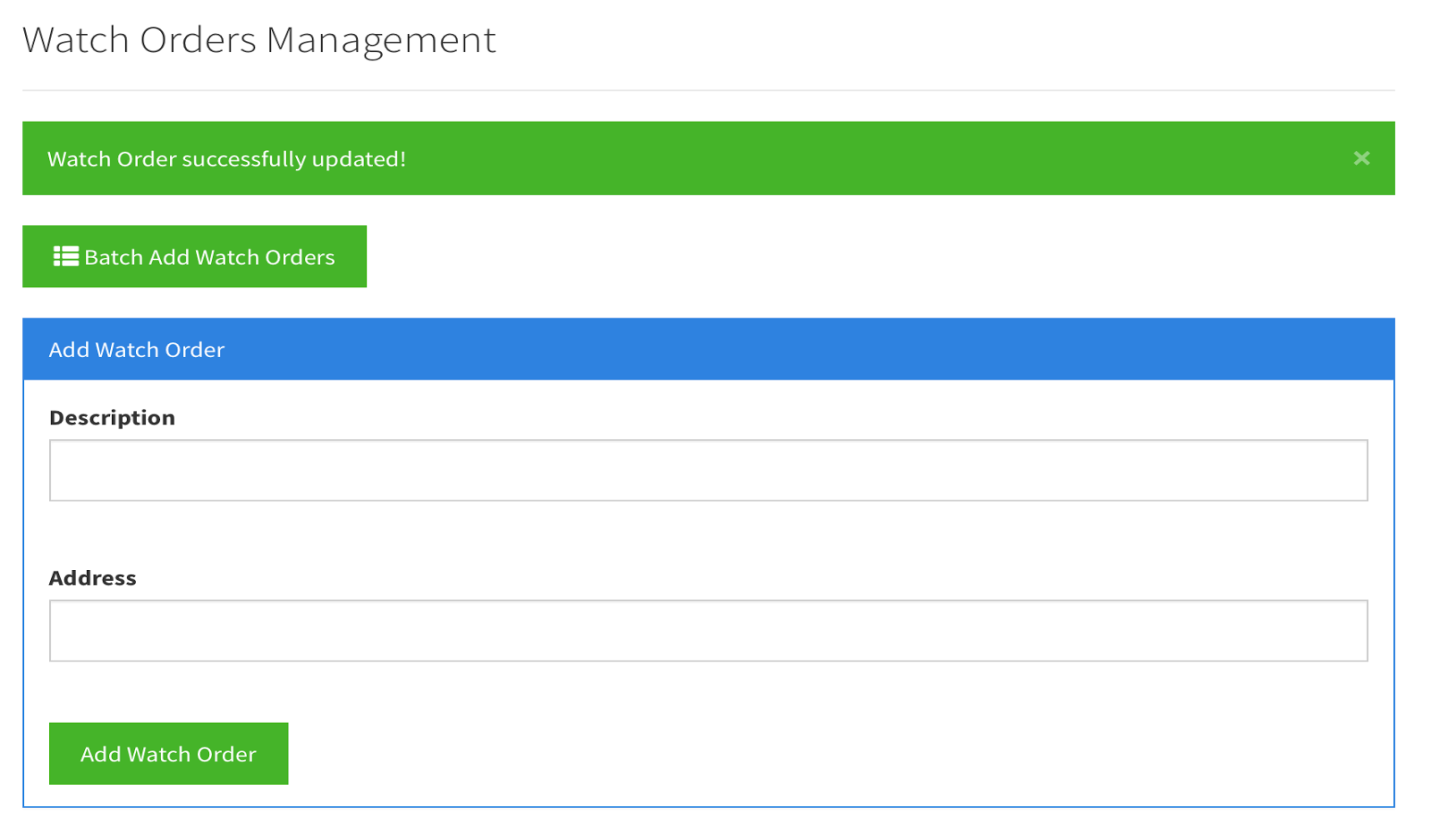
**Figure C.38: Delete Watch Order (#306)**

**Description:** Success message after deleting watch order.

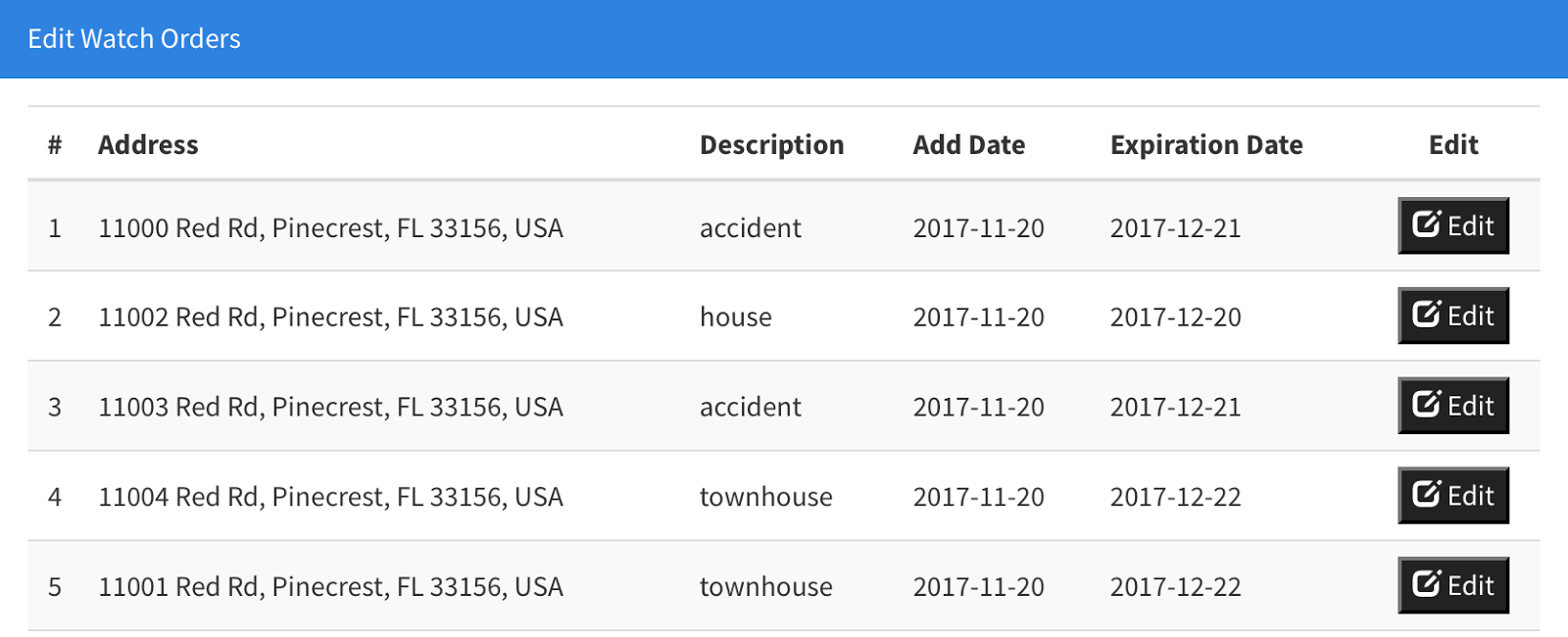


**Figure C.39.1: Edit Watch Order (#307)**

**Description:** Edit Watch Orders Modal View.

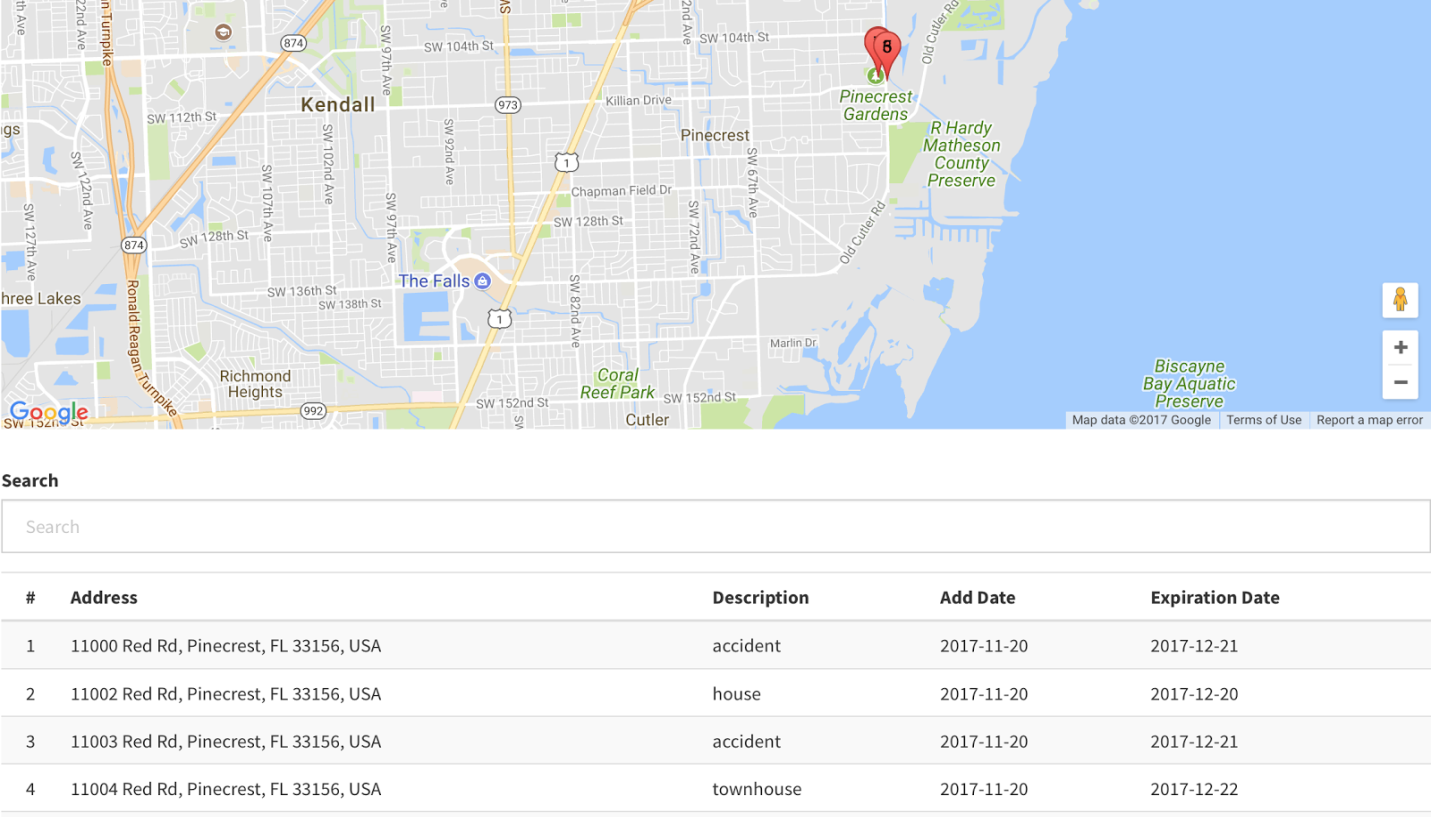
**Figure C.39.2: Edit Watch Order (#307)**

**Description:** Success message after editing watch order.



**Figure C.40: Automatically Delete Watch Orders (#309)**

**Description:** Table Displaying Only Unexpired Watch Orders:



**Figure C.41: Display Watch Order List Table (#310)**

**Description:** Table Displaying Watch Orders.

## Appendix D - Sprint Review Reports

### Version 1.0

**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 was 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 was 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 was 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 was 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 was 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 was 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

### Version 2.0

**Sprint Review Meeting Minutes - Sprint 1 (Version 2.0)**

Attendees: Eric Fernandez, Oscar A Yannelli

Date: 05/26/2017

Start time: 6:30 PM

End time: 7:00 PM

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

* User Story (Not user stories accepted so far, still setting up everything and getting familiar with documentation and current system)

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 (Not user stories rejected so far, still setting up everything and getting familiar with documentation and current system)

**Sprint Review Meeting Minutes - Sprint 2 (Version 2.0)**

Attendees: Eric Fernandez, Oscar A Yannelli

Date: 06/09/2017

Start time: 6:30 PM

End time: 7:00 PM

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

* User Story (Not user stories accepted so far, we have already started with development but we haven’t been able to complete any user story so far)

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 (Not user stories rejected so far, we have already started with development but we haven’t been able to complete any user story so far)

**Sprint Review Meeting Minutes - Sprint 3 (Version 2.0)**

Attendees: Eric Fernandez, Oscar A Yannelli

Date: 06/23/2017

Start time: 6:30 PM

End time: 7:00 PM

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

* User Story 244: Add User Logs
* User Story 233: Add Document Confirmation

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.

The two user stories completed were approved. None story was rejected. But we were unable to complete one of the user stories.

* User Story 245: Archive Documents
* How this should be reflected on the user story definition in Mingle:
  + The user story has been moved to the next sprint.

**Sprint Review Meeting Minutes - Sprint 4 (Version 2.0)**

Attendees: Eric Fernandez,Oscar A Yannelli

Date: 07/07/2017

Start time: 6:30 PM

End time: 7:00 PM

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

* User Story 245: Archive Documents.
* User Story 273: Add Description field to document upload and officer’s document view.
* User Story 274: Add upload document confirmation.
* User Story 275: Add column to Supervisor Documents grid for active or archived.
* User Story 276: Fix Supervisor document edit screen that is not working.
* User Story 277: Display all documents on Logs grid. Add columns status, start date/time, end date/time and duration time.

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 of the stories were rejected, all stories satisfy our customer requirements.

### Version 3.0

##### Sprint Review Meeting Minutes – Sprint 2 (Version 3.0)

September 29, 2017

Attendees: Juan Lopez, Jean Faustin, Frank Alvarado

Start time: 1:00PM

End time: 1:20PM

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

* **User Story 242:** Change Display Mode
* **User Story 295:** Display Documents within Website
* **User Story 296:** Display Unread Documents Notification

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

##### Sprint Review Meeting Minutes Sprint 3 (Version –3.0)

October 13, 2017

Attendees: Juan Lopez, Jean Faustin, Jason Cohen, Frank Alvarado

Start time: 1:00PM

End time: 1:20PM

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

* **User Story 147:** Lock User
* **User Story 241:** Add Users with CSV File
* **User Story 250:** Add Password Encryption
* **User Story 300:** Improve Document List Presentation

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

##### Sprint Review Meeting Minutes – Sprint 4 (Version 3.0)

October 27, 2017

Attendees: Juan Lopez, Jean Faustin, Frank Alvarado

Start time: 1:00PM

End time: 1:20PM

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

* **User Story 246**: Add Watch Orders with CSV File
* **User Story 247**: View Watch Orders on Map
* **User Story 301**: Prepare App for Distribution
* **User Story 302**: Change Navigation Bar Based on User Role

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

##### Sprint Review Meeting Minutes – Sprint 5 (Version 3.0)

November 09, 2017

Attendees: Juan Lopez, Jean Faustin, Frank Alvarado

Start time: 3:30PM

End time: 3:50PM

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

* **User Story 248**: Add Text Information
* **User Story 308**: Edit Text Information
* **User Story 304**: Add Watch Order
* **User Story 306**: Delete Watch Order
* **User Story 307**: Edit Watch Order

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

##### Sprint Review Meeting Minutes – Sprint 6 (Version 3.0)

November 23, 2017

Attendees: Juan Lopez, Jean Faustin, Frank Alvarado

Start time: 3:30PM

End time: 3:50PM

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

* **User Story 243**: Add Quiz
* **User Story 309:** Automatically Delete Watch Orders
* **User Story 310:** Display Watch Order List Table
* **User Story 312**: Take Quiz

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 E - User Manuals, Installation/Maintenance Document, Shortcomings/Wish list Document and other documents

# 

**Links to videos:**

**Project Introduction**:

<https://www.youtube.com/watch?v=dB9PjnYQAbY&list=PLG3lylzOg8FiXAWSZO6Hs9bpBO4Aqn2Mo&index=1>

**User Manuals:**

**Administrator**

<https://www.youtube.com/watch?v=uL7qJInT9F4&list=PLG3lylzOg8FiXAWSZO6Hs9bpBO4Aqn2Mo&index=3>

**Supervisor**

<https://www.youtube.com/watch?v=olqzz-cdjOE&t=5s&list=PLG3lylzOg8FiXAWSZO6Hs9bpBO4Aqn2Mo&index=4>

**Officer**

<https://www.youtube.com/watch?v=B5lzA9G8nCM&list=PLG3lylzOg8FiXAWSZO6Hs9bpBO4Aqn2Mo&index=5>

**Installation:**

<https://www.youtube.com/watch?v=rIQnAzJAYWc&list=PLG3lylzOg8FiXAWSZO6Hs9bpBO4Aqn2Mo&index=2>

**Shortcomings/Wish list**

<https://www.youtube.com/watch?v=Ft1yK0WYS6A&list=PLG3lylzOg8FiXAWSZO6Hs9bpBO4Aqn2Mo&index=6>

## Appendix F - 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