

*Florida International University
School of Computing and Information Sciences*

CIS 4911 - Senior Capstone Project
Software Engineering Focus

Final Deliverable

BOLO Flier Creator Version 6

Team Members:

Dominick Martelly
Brian Zamora

Product Owner(s):

Samuel Ceballos
Jason Cohen

Instructor: Masoud Sadjadi

The MIT License (MIT)
Copyright (c) 2016

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

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

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

Abstract

The Pinecrest Police Department wish to bring policing into the future by combining technology with their vast domain knowledge. The current system for the distribution of BOLO's is insecure. For these reasons the Pinecrest Police department has employed the assistance of the college of engineering and computer science at Florida International University implement their innovative idea.

This document is a repertoire of the BOLO Version 6.0 system, a user-driven approach that allows law enforcement agents to instantly publish BOLOs. This document contains an extensive list of implemented user stories. It also provides a detailed plan of execution that includes hardware and software specifications. System design is explained through specific diagrams. System validation is also discussed. A glossary with essential domain terminology can also be found. Finally, the last chapter contains a series of appendices with UML samples, screenshots, and progress reports.

Table of Contents

Introduction.....	6
Current System.....	6
Purpose of New System.....	6
Implemented User Stories.....	7
Sprint 1.....	7
Sprint 2.....	7
Sprint 3.....	8
Sprint 4.....	8
Sprint 5.....	9
Sprint 6.....	10
Project Plan	11
Hardware and Software Resources	11
Development Environment	11
Deployment Requirements.....	11
Sprints Plan	12
Sprint 1.....	12
Sprint 2.....	14
Sprint 3.....	16
Sprint 4.....	17
Sprint 5.....	21
Sprint 6.....	24
System Design	26
Architectural Patterns.....	26
System and Subsystem Decomposition	26
Deployment Diagram.....	27
Design Patterns	28
System Validation.....	29
Glossary	34
Appendix.....	35
Appendix A – UML Diagrams	35
Static UML Diagrams	35
Dynamic UML Diagrams	62
Appendix B – User Interface Design	88

Appendix C – Sprint Review Reports.....	107
Appendix D – Other Documents.....	119
References.....	124

Introduction

“BOLO Flyer Creator” is a web application that allows police officials of Pinecrest police agency and others to distribute and receive BOLOs (Be on the Lookout) flyers on the web. This is so that other officials in the surrounding areas can share local information quickly and securely to each other. This is meant to enhance the current system, which uses the following strategies to create and disseminate BOLOs: alerts over police radio, computer-aided dispatch that is dependent on antiquated hardware, and paper printed posters for manual distribution.

Current System

The current approach used by Pinecrest Police Department to create BOLO flyers is insecure. The web server is hosted on IBM’s Bluemix originally due to their uptime and largely expandable servers. Unfortunately, due to the confidentiality that cannot be guaranteed from the passing of confidential information to non-police-conformant servers, IBM’s servers can no longer be used to host the application.

Excluding the confidentiality, the system is too dependent on the use of programmers to add functionality instead of supervisors to the ever evolving world of crime. There needs to be a way for officers and administrators to create new fields when a new type of crime occurs.

Purpose of New System

BOLO Flyer Creator 6.0 was designed to change the database used for the application, and implement expandable categories for BOLOs.

The change of the database will allow the web application to be installed on a private server where the system operator will have full control of the system. This will ensure security and confidentiality if the information passed between other officers.

The expandable categories will now free the limits on what gets reported on the field in the future. If a crime or report needs to be created, but the form is not on the system, administrators can create new category forms so that the user can share valuable information. The system is essentially future proof since the new categories are stored on the database.

Other implementations were also completed on the system, including, but not limited to Agency removal, password conformation, and webpage editing.

Implemented User Stories

This is a list of the user stories that were implemented for version 6 of BOLO. All the user stories compiled here were requested by the product owner, Jason Cohen.

Sprint 1

(08/27/2016 - 09/09/2016)

User Story #1177 All pictures are viewed on the bolo-details view

- As a user, I can view all the pictures for a selected BOLO on a ViewSelectedBOLO Page

User Story #1179 Allow batch upload of excel document for new users

- As an admin, I can upload an excel file of new users, so that they can be added into the system

User Story #1186 New User Button

- As an admin, I want the create new user button to always be readily available, so that I don't have to scroll all the way to the bottom of the page to create a new user.

User Story #1187 Generated watermark on completed BOLOs

- As an officer, I want a watermark to be rendered when a BOLO is marked as completed, so that it is made clear which BOLOs are still pending and which ones are completed.

Sprint 2

(09/10/2016 - 09/23/2016)

User Story #1182 Allow root user to edit the “About Us” Page

- As a root user, I want to be able to edit the “About Us” page, so that changes can be made in the future if necessary

User Story #1183 Allow root to edit the “Login” page

- As a root user, I want to be able to edit the “Login” page, so that changes can be made in the future if it is necessary

User Story #1184 Allow root to edit the “User Guide” page

- As a root user, I want to be able to edit the “User Guide” page, so that changes can be made in the future if it is necessary

User Story #1189 Agency Specific BOLO

- As an officer I want to be able to view BOLO fliers specific to my department, so that I can see what BOLO fliers are active in my area

User Story #1190 Agency Specific View

- As an officer I want to be able to create an internal BOLO flier, specific to my department so that other officers in my department can see what is going on in our local area

Sprint 3

(09/24/2016 – 10/07/2016)

User Story #1191 Email PDF Privacy

- As an officer I want the details of a BOLO flier to be kept to a minimum when being emailed, so that no one except those that have access to the BOLO application can have specifics about the flier

User Story #1194 BOLO Category Template Creation

- As an administrator or root user, I want to be able to create a new template for a BOLO, so that the BOLO system can adapt based on the needs of the police department.

User Story #1196 BOLO Template Rendering

- As a user, I want to be able to select the category of the BOLO template that I want to fill out, so that from one single page I can load any template that I need, instead of having a separate static view for each category.

Sprint 4

(10/08/2016 – 10/21/2016)

User Story #700 Delete Agency

- As root, I can delete an Agency from the database as well as all users associated with the agency, so that I can better manage the BOLO web application.

User Story #1188 Convert Web Application Database to MongoDB

- As a root, I can install the BOLO web application on a server that can run MongoDB, so that the web application can be run on a secure server.

User Story #1192 Delete User

- As a root user or an administrative user, I want to be able to delete a user from the database, so that I can better manage the BOLO web application.

User Story #1195 Adjust CSV Uploading to use MongoDB

- As a root/administrator user, I want to be able to upload a csv user, so that I can upload more than one user at a time into the BOLO system.

User Story #1197 Email Confirmation for BOLO creation

- As a user, I want to receive an email notification when a new BOLO is created so that I can confirm that I want the BOLO to be posted.

User Story #1198 Adjust BOLO View

- As a user I want to have a central location to view all BOLO fliers that were created, so that I can view them at any time.

User Story #1199 Adjust PDF Rendering for MongoDB

- As a user I want to be able to create a PDF of a BOLO, so that I can easily print and distribute them if needed.

User Story #1200 Adjust PDF Email Privacy for MongoDB

- As an officer I only want to see a portion of the BOLO in my email, so that eavesdroppers can't get full BOLO details from my email.

Sprint 5

(10/22/2016 – 11/04/2016)

User Story #1180 Set Password

- As a new user, I can immediately change my password when I first login to my BOLO account, so that no one knows my password except me.

User Story #1193 BOLO Image Compression

- As a user, all photos uploaded as an image to my BOLO flier is compressed so the database will not fill its storage quickly

User Story #1202 BOLO Image Resizing

- As a user I want the images to be resized based on how many images are present on the BOLO, so that the pictures are always very prominent and as large as they can be.

User Story #1204 Data Analysis

- As a root/administrator user, I want to be able to pull data for my agency in a CSV form, so that I can quickly see what kind of bolas are created by agency as well as the details of the BOLOs.

User Story #1206 New User Notification

- As a root/administrator, I want that a new user gets an email notification when a BOLO account is made for them, so that they can begin using the BOLO system right away.

Sprint 6

(11/05/2016 – 11/11/2016)

User Story #1201 BOLO Data Subscriber

- As a user, I can subscribe to a particular BOLO so that I will be updated if the status of a subscribed BOLO changes

User Story #1205 BOLO Password Verification

- As a root/administrator, I want to make sure that all users are following the password strength requirements, so that the BOLO web application can only be accessed to those that are authorized to access it.

Project Plan

The project was done using agile development where project members had sprint meeting five days of the week. A meeting with the product owner occurs every Friday and sprints last bi-weekly.

Below are the recommended hardware specifications as well as the required software needed to run the application from a developer's point of view.

The plans for each sprint is also listed. The details for each sprint includes the requirements, related tasks, and references.

Hardware and Software Resources

In order to develop and deploy the application some hardware and software is required. The following sections will describe the environments needed.

Development Environment

In order to develop the application, the follow hardware specs are recommended:

- Windows/Apple Computer
- Min of 4Gb of RAM
- At least 30 GB of disk space
- At least 2.0 GHZ processor

For software development, developers are recommended to have working knowledge of the following:

- HTML 5 / CSS 3
- JavaScript (ECMAScript 6)
- Pug (previously known as Jade) templating for HTML
- Node.js 4.2+
- NPM (NPM is the package manager for node.js. It is used to install project dependencies as well as to automate the execution of the application testing and miscellaneous scripts)
- Express Framework (Express is a minimal and flexible Node.js dependency that provides a robust set of features for web and mobile applications such as routing)
- Favorite Text Editor (Atom) / Preferred IDE (Webstorm)
- MongoDB, A NoSQL database

Deployment Requirements

- The project is targeted for deployment on a Linux or Windows based server.
- MongoDB must be running on the same server, or hosted on another server with low latency (< 50ms)
- Access tokens are required to be set as environment variables for the application to work in both deployment and development.
- An active Sendgrid account. This is the service the application uses for sending notifications. An API key is required from SendGrid in order to successfully run the application locally and in production.

Sprints Plan

The primary goal of version 6 of the BOLO application was to change the database from IBM's Cloudant to MongoDB, adding to the previous iteration of the application. This feature will allow the BOLO application to be hosted on a server chosen by the product owner; The product owner will no longer be limited by only being able to run the application on IBM servers.

These are the user stories that were released per sprint. Listed for each user story is their respective story, functional/non-functional requirements, and related tasks.

Sprint 1

(08/27/2016 - 09/09/2016)

User Story #1177

As a user, I can view all the pictures for a selected BOLO on a ViewSelectedBOLO Page

Acceptance Criteria

Functional Requirements

- All users will view all the bolo pictures for a bolo when selected
- All pictures for a bolo, if any, are viewed on the same row
- Pictures are laid out horizontal

Non-Functional Requirements

- Pictures load in no longer than 3 seconds
- Pictures are vertical if viewed on a device smaller than 500px in width

Related Tasks / Stories

None

Modeling

- Refer to UML diagrams in Appendix A: Static UML Diagrams, Figure sp1-1 and Dynamic UML Diagrams, Figure sp1-1

User Story #1179

As an admin or root, I can upload an excel file of new users to add into the system so that new users can be created in batches

Acceptance Criteria

Functional Requirements

- Root and administrators can batch upload an excel document containing users
- Root and administrators can download a sample csv file to view the format

Non-Functional Requirements

- The process should take no longer than 15 seconds after upload
- The maximum upload file size is 100MB

Related Tasks / Stories

None

Modeling

- Refer to UML diagrams in Appendix A: Static UML Diagrams, Figure sp1-1 and Dynamic UML Diagrams, Figure sp1-1

User Story #1186

As an admin or root, the create new user button is always readily available so that I can quickly create a user without scrolling the page

Acceptance Criteria**Functional Requirements**

- The admin won't have to scroll down the list to add a new user into the system
- The add new user button will be at the top of the page
- The button will be immediately visible when the page loads

Non-Functional Requirements

- None

Related Tasks / Stories

None

Modeling

- Refer to UML diagrams in Appendix A: Static UML Diagrams, Figure sp1-1 and Dynamic UML Diagrams, Figure sp1-1

User Story #1187

As a user, I can mark a BOLO as located/solved/completed and have a generated watermark on the BOLO page so that other users will clearly know the status of a BOLO

Acceptance Criteria**Functional Requirements**

- Watermarks appear on BOLOs that are not still ongoing
- Officers can change their own watermarks on their own BOLOs

Non-Functional Requirements

- All watermarks have a 45-degree angle
- All watermarks are red in color

Related Tasks / Stories

None

Modeling

- Refer to UML diagrams in Appendix A: Static UML Diagrams, Figure sp1-1 and Dynamic UML Diagrams, Figure sp1-1

Sprint 2

(09/10/2016 - 09/23/2016)

User Story #1182

As a root user, I want to be able to edit the “About Us” page, so that changes can be made in the future if necessary

Acceptance Criteria

Functional Requirements

- Only root users will see the edit “About Us” link
- The page will be editable in Markdown
- The user will have a guide that shows how to edit in Markdown

Non-Functional Requirements

- The markdown should save in less than 5 seconds

Related Tasks / Stories

User Story #1183, User Story #1184

Modeling

- Refer to UML diagrams in Appendix A: Static UML Diagrams, Figure sp1-1 and Dynamic UML Diagrams, Figure sp1-1

User Story #1183

As a root user, I want to be able to edit the “Login” page, so that changes can be made in the future if necessary

Acceptance Criteria

Functional Requirements

- Only root users will see the edit “Login” link
- The page will be editable in Markdown
- The user will have a guide that shows how to edit in Markdown

Non-Functional Requirements

- The markdown should save in less than 5 seconds

Related Tasks / Stories

User Story #1183, User Story #1184

Modeling

- Refer to UML diagrams in Appendix A: Static UML Diagrams, Figure sp1-1 and Dynamic UML Diagrams, Figure sp1-1

User Story #1184

As a root user, I want to be able to edit the “User Guide” page, so that changes can be made in the future if necessary

Acceptance Criteria**Functional Requirements**

- Only root users will see the edit “User Guide” link
- The page will be editable in Markdown
- The user will have a guide that shows how to edit in Markdown

Non-Functional Requirements

- The markdown should save in less than 5 seconds

Related Tasks / Stories

User Story #1182, User Story #1183

Modeling

- Refer to UML diagrams in Appendix A: Static UML Diagrams, Figure sp1-1 and Dynamic UML Diagrams, Figure sp1-1

User Story #1189

As an officer I want to be able to create BOLO fliers specific only to my department, so that I can share what BOLO fliers are active in my area.

Acceptance Criteria**Functional Requirements**

- The officer can switch between creating a general BOLO and a department specific BOLO
- The department specific BOLO container contains only BOLO fliers that belong to that officer's police department

Non-Functional Requirements

- BOLOs from an agency should be viewed after 3 seconds after selecting

Related Tasks / Stories

None

Modeling

- Refer to UML diagrams in Appendix A: Static UML Diagrams, Figure sp1-1 and Dynamic UML Diagrams, Figure sp1-1

User Story #1190

As an officer I want to be able to view BOLO fliers specific only to my department, so that I can see what BOLO fliers are active in my area.

Acceptance Criteria

Functional Requirements

- The officer can switch between the general BOLO container and a department specific BOLO container
- The department specific BOLO container contains only BOLO fliers that belong to that officer's police department

Non-Functional Requirements

- BOLOs from an agency should be viewed after 3 seconds after selecting

Related Tasks / Stories

None

Modeling

- Refer to UML diagrams in Appendix A: Static UML Diagrams, Figure sp1-1 and Dynamic UML Diagrams, Figure sp1-1

Sprint 3

(09/24/2016 – 10/07/2016)

User Story #1191

As an officer, I want the details of a BOLO flier to be kept to a minimum when being emailed, so that no one except those that have access to the BOLO application can have specifics about the flier

Acceptance Criteria

Functional Requirements

- The PDF when emailed must contain only the featured images, the name of the department, and the appropriate logos
- The Email contains a link so that the officer can be redirected to the BOLO application to view full details of the BOLO

Non-Functional Requirements

- The email should arrive to the user within 15 minutes

Related Tasks / Stories

None

Modeling

- Refer to UML diagrams in Appendix A: Static UML Diagrams, Figure sp1-1 and Dynamic UML Diagrams, Figure sp1-1

User Story #1194

As an admin or root, I want to be able to create a new template for a BOLO, so that the BOLO system can adapt based on the needs of the Police Department.

Acceptance Criteria**Functional Requirements**

- The root/administrator user can visit the administrator portal
- The root/administrator user can visit the category management view
- The root/administrator user can create a new category
- The root/administrator user can edit a category only if the category isn't being used

Non-Functional Requirements

- The category is saved on the database within 5 seconds

Related Tasks / Stories

None

Modeling

- Refer to UML diagrams in Appendix A: Static UML Diagrams, Figure sp1-1 and Dynamic UML Diagrams, Figure sp1-1

User Story #1196

As a user, I want to be able to select the category of the BOLO template that I want to fill out, so that from one single page I can load any template that I need instead of having a separate static view for each category.

Acceptance Criteria**Functional Requirements**

- Any user can click "Create BOLO" and is presented with only one view
- The user can select the template that they want from a drop down menu
- The template fields are rendered on the fly without having to switch any pages

Non-Functional Requirements

- The form fields render within one second

Related Tasks / Stories

None

Modeling

- Refer to UML diagrams in Appendix A: Static UML Diagrams, Figure sp1-1 and Dynamic UML Diagrams, Figure sp1-1

Sprint 4

(10/08/2016 – 10/21/2016)

User Story #1188

As the product owner, I can run the BOLO program on a server that can run MongoDB, so that the web application can be ran on a secure server

Acceptance Criteria**Functional Requirements**

- BOLO can run only on MongoDB, not Cloudant
- The functions that was used with Cloudant are still usable with MongoDB
- The database can be running with the server, or hosted elsewhere

Non-Functional Requirements

- MongoDB queries must be of equal speed or faster than Cloudant

Related Tasks / Stories

None

Modeling

- Refer to UML diagrams in Appendix A: Static UML Diagrams, Figure sp1-1 and Dynamic UML Diagrams, Figure sp1-1

User Story #1195

As a root/administrator user, I want to be able to upload a csv of users on the database, so that I can upload more than one user at a time.

Acceptance Criteria**Functional Requirements**

- The root/administrator user can access the user management console
- The root/administrator user can upload a list of users in a properly formatted CSV
- The users are created on MongoDB
- The users are successfully registered on the BOLO system.

Non-Functional Requirements

- All users are uploaded in under 20 seconds

Related Tasks / Stories**User Story #1179****Modeling**

- Refer to UML diagrams in Appendix A: Static UML Diagrams, Figure sp1-1 and Dynamic UML Diagrams, Figure sp1-1

User Story #1197

As a user, I want to receive an email notification when a new BOLO is created so that I can confirm I want the BOLO posted.

Acceptance Criteria

Functional Requirements

- After a BOLO is created successfully, it is assigned a token.
- An email is sent to the author on BOLO creation to confirm the BOLO.

Non-Functional Requirements

- The email must not take longer than 15 seconds to send to the user.

Related Tasks / Stories

None

Modeling

- Refer to UML diagrams in Appendix A: Static UML Diagrams, Figure sp1-1 and Dynamic UML Diagrams, Figure sp1-1

User Story #1198

As a user I want to have a central location to view all BOLO fliers that were created, so can view them at any time.

Acceptance Criteria

Functional Requirements

- When the user logs in they can visually see all the BOLO fliers that have been created
- The user can see the status of the BOLO across the thumbnail of the picture if it isn't new or updated

Non-Functional Requirements

- The BOLOs must not take longer than 10 seconds to load

Related Tasks / Stories

None

Modeling

- Refer to UML diagrams in Appendix A: Static UML Diagrams, Figure sp1-1 and Dynamic UML Diagrams, Figure sp1-1

User Story #1199

As a user, I want to be able to create a PDF of a BOLO, so that I can easily print it and distribute it if needed

Acceptance Criteria

Functional Requirements

- The user can view a PDF from the BOLO details page

Non-Functional Requirements

- The PDF must not take longer than 10 seconds to load

Related Tasks / Stories**None**Modeling

- Refer to UML diagrams in Appendix A: Static UML Diagrams, Figure sp1-1 and Dynamic UML Diagrams, Figure sp1-1

User Story #1200

As a user, I only want to see a portion of the BOLO in my email, so that eavesdroppers can't get full BOLO details from my email.

Acceptance Criteria**Functional Requirements**

- The user receives an email notification of a new BOLO
- The PDF attached to the notification shows only the category, status, pictures, and the department.

Non-Functional Requirements

- The PDF email is set to the user within 15 minutes

Related Tasks / Stories**None**Modeling

- Refer to UML diagrams in Appendix A: Static UML Diagrams, Figure sp1-1 and Dynamic UML Diagrams, Figure sp1-1

User Story #1192

As an admin or root, I can delete a user from the database, so that the deleted user's information does not exist

Acceptance Criteria**Functional Requirements**

- The user I am deleting is not the user 'root'
- I have a confirmation before deleting a user
- If I am an administrator, I can only delete users from my agency, including myself
- The current user must enter a password to confirm deletion

Non-Functional Requirements

- The user is deleted within 5 seconds after password confirmation

Related Tasks / StoriesUser Story #700Modeling

- Refer to UML diagrams in Appendix A: Static UML Diagrams, Figure sp1-1 and Dynamic UML Diagrams, Figure sp1-1

User Story #700

As a root, I can delete an Agency from the database along with all users associated with the agency, so that the agency is inaccessible

Acceptance Criteria**Functional Requirements**

- Agency is deleted only after a confirmation page
- Only a root user will have access to this command
- All users from that agency are also deleted
- The current user must enter a password to confirm deletion

Non-Functional Requirements

- Agency deletion takes no longer than 5 seconds

Related Tasks / Stories**User Story #1192****Modeling**

- Refer to UML diagrams in Appendix A: Static UML Diagrams, Figure sp1-1 and Dynamic UML Diagrams, Figure sp1-1

Sprint 5

(10/22/2016 – 11/04/2016)

User Story #1193

As a user, all photos uploaded as an image to my bolo flyer is compressed, so the database will not fill its storage quickly

Acceptance Criteria**Functional Requirements**

- The user will choose an image from their device when creating a BOLO
- The image ratio will be kept the same after compression

Non-Functional Requirements

- The maximum image size of the compressed image is 100KB
- The maximum image resolution of the compressed image is 500x500px
- Accepted images are .jpeg, .png, and .bmp
- The time it takes from upload to compression should be less than 15 seconds

Related Tasks / Stories**User Story #1177, User Story #1201****Modeling**

- Refer to UML diagrams in Appendix A: Static UML Diagrams, Figure sp1-1 and Dynamic UML Diagrams, Figure sp1-1

User Story #1180

As a new user, I can immediately change my password when I first login to my BOLO account, so that no one knows my password except me.

Acceptance Criteria

Functional Requirements

- When a new user first logs into the system, they must change their password
- When a new user is created, they will have a randomly generated password

Non-Functional Requirements

- The password expires if the account is not activated within a day
- The system does not take longer than 1 second to register the new password

Related Tasks / Stories

None

Modeling

- Refer to UML diagrams in Appendix A: Static UML Diagrams, Figure sp1-1 and Dynamic UML Diagrams, Figure sp1-1

User Story #1202

As a user, I want the images to be resized based on how many images are present on the BOLO, so that the pictures are always very prominent and as large as they can be.

Acceptance Criteria

Functional Requirements

- When there is one image on the BOLO, it is displayed at 75% of the page width
- When there are two images on the BOLO, they both have to fit evenly within 75% of the page width
- When there are three images on the BOLO, they all have to fit evenly within 90% of the page width

Non-Functional Requirements

- The image is rendered in the correct size in less than 1 second
- The image resizing should take less than 1 second

Related Tasks / Stories

None

Modeling

- Refer to UML diagrams in Appendix A: Static UML Diagrams, Figure sp1-1 and Dynamic UML Diagrams, Figure sp1-1

User Story #1206

As a root/administrator, I want that a new user gets an email notification when a BOLO account is made for them, so that they can begin using the BOLO system right away.

Acceptance Criteria**Functional Requirements**

- Email notification is sent when a new user is created
- The notification has a link so that the User can quickly access the BOLO System.

Non-Functional Requirements

- The user receives the email notification within 1 hour of account creation

Related Tasks / Stories**None****Modeling**

- Refer to UML diagrams in Appendix A: Static UML Diagrams, Figure sp1-1 and Dynamic UML Diagrams, Figure sp1-1

User Story #1204

As a root user, I want to be able to pull analysis for my agency in a CSV form, so that I can quickly see what kind of bolos are created for agency as well as the details of the BOLOs.

Acceptance Criteria**Functional Requirements**

- When a root user clicks on data analysis they can select which agency they want the report for
- When a root user selects the agency, a CSV report is downloaded

Non-Functional Requirements

- The CSV is not downloaded if no agency is selected
- The CSV does not take longer than 10 seconds to download

Related Tasks / Stories**None****Modeling**

- Refer to UML diagrams in Appendix A: Static UML Diagrams, Figure sp1-1 and Dynamic UML Diagrams, Figure sp1-1

Sprint 6

(11/05/2016 – 11/11/2016)

User Story #1201

As a user, I can subscribe to a particular BOLO so that I will be updated if the status of a subscribed BOLO changes

Acceptance Criteria

Functional Requirements

- Users can subscribe to BOLOs from the BOLO details page
- Users are alerted when a BOLO has been updated

Non-Functional Requirements

- Emails are sent within 10 second after a bolo has been edited

Related Tasks / Stories

None

Modeling

- Refer to UML diagrams in Appendix A: Static UML Diagrams, Figure sp1-1 and Dynamic UML Diagrams, Figure sp1-1

User Story #1205

As the product owner, I want to make sure that all users are following the password strength requirements, so that the BOLO web application can only be accessed to those that are authorized to access it.

Acceptance Criteria

Functional Requirements

- Users are notified of the password requirements
- User is prompted for a new password if the entered password does not meet the requirements
- When a password is rejected, the user is notified

Non-Functional Requirements

- The password is checked and verified for strength within 2 seconds
- The password is saved and confirmed within 3 seconds

Related Tasks / Stories

None

Modeling

- Refer to UML diagrams in Appendix A: Static UML Diagrams, Figure sp1-1 and Dynamic UML Diagrams, Figure sp1-1

System Design

The system uses the hexagonal architecture, a variation of the model-view-controller architecture. The domain layer determines the persistence of the system since each domain is associated with services and repositories from the application layer. The hexagonal architecture will allow the design to expand the ports and adapters into micro-services, such as email and authentication.

The application is currently planned to be hosted on Pinecrest Police Department's servers. Outside of the application layer there are other micro-services that interact with the application. The selected structure facilitates the scalability and maintainability of the system. It makes easy to add features without affecting essential functionalities.

This chapter provides an overview of the Architectural patterns used to design the application. It describes the different subsystems in which the application was divided into. A Deployment diagram is included to depict how the system was implemented. The different Design patterns used is also included.

Architectural Patterns

BOLO 6.0 uses a Hexagonal architecture. There are four main elements that determine the system's functionality; users, categories, agencies, and bolas. These subsystems determine the domain, from which all the system's features are derived. One advantage of this architectural pattern is the ability to have several layers for the different subsystems that are interchangeable if necessary. All features within the application are determined through routes which transfer the requests a service, which send the correct call to a repository, which is the entity allowed to read and write from the MongoDB database. The multilayered approach provides advantages in terms of security, testing and scalability, since it makes easy to add new services without affecting core functionalities.

System and Subsystem Decomposition

BOLO 6.0 is divided into six major subsystems; agencies, authentication, edit, users, bolas. All subsystems are composed by views, a router, a service, and a repository. The proper functionality of the system relies on the interaction among all 6 subsystems:

Agency Controller: It handles agency creation, edition, deactivation, and deletion. Without an agency, users can not exist in the system.

Authentication Controller: It handles the login cookies of different users accessing the system. Every call to the server will pass though this controller to ensure that the user is currently logged in. It also handles password expiration.

Bolo Controller: Being the most extensive one, it manages all bolo-related functionality such as BOLO creation, previewing, search, PDF, Email, deletion, archiving and others.

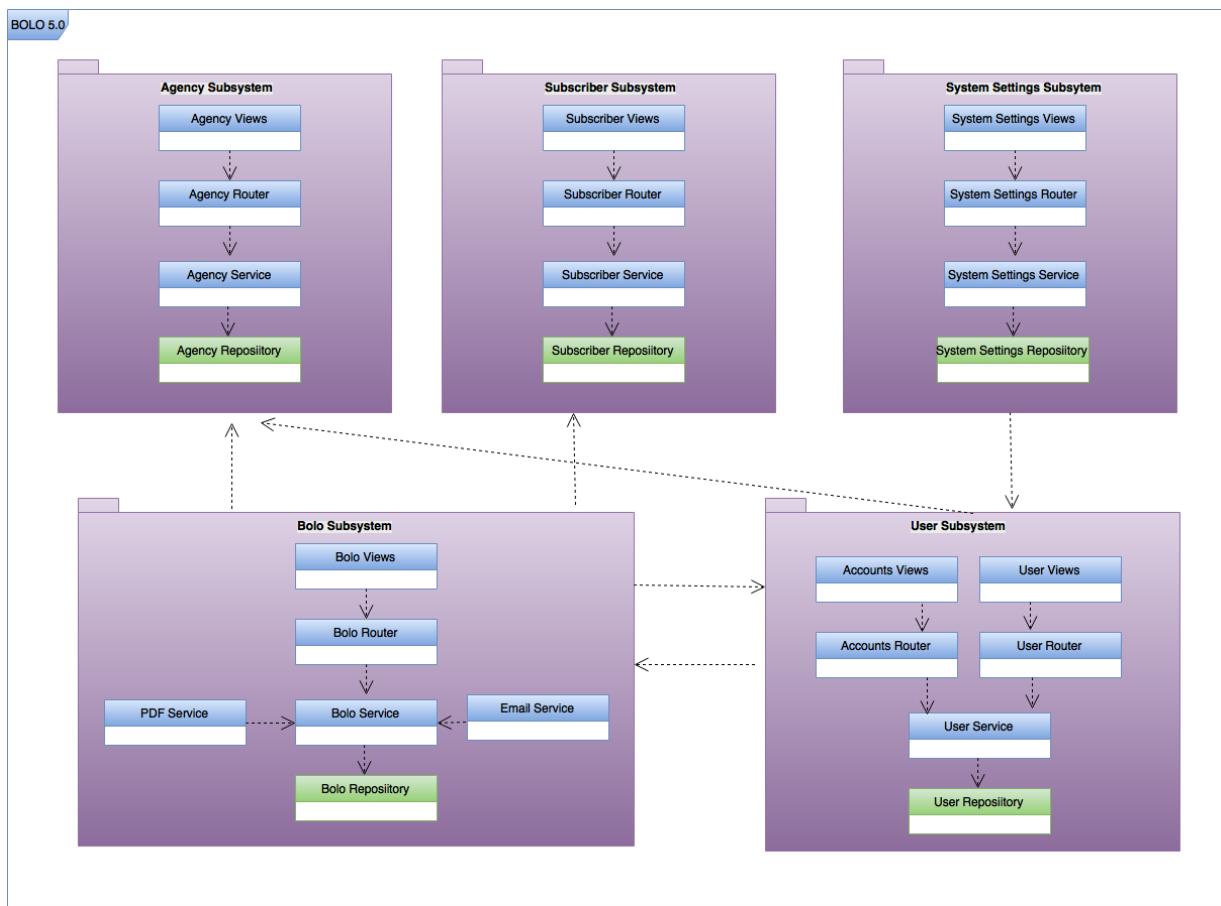
Category Controller: It handles the categories and form data for new bolo. BOLOs can only be created after at least one category is on the system.

Edit Controller: It allows the root user to edit the website. Editable pages include the user guide and the about us page.

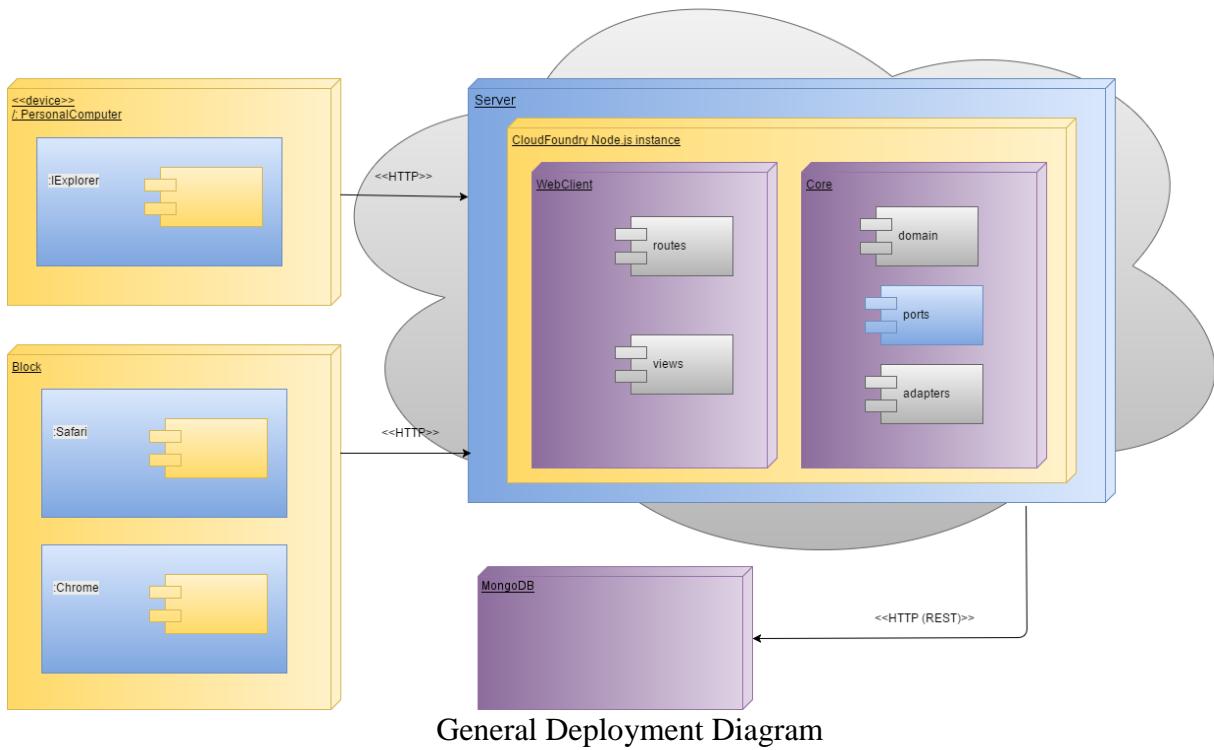
User Controller: It handles user accounts and user management. It allows users to be created, edited, and deleted. It also oversees account functionality such as passwords, login, and logout.

Deployment Diagram

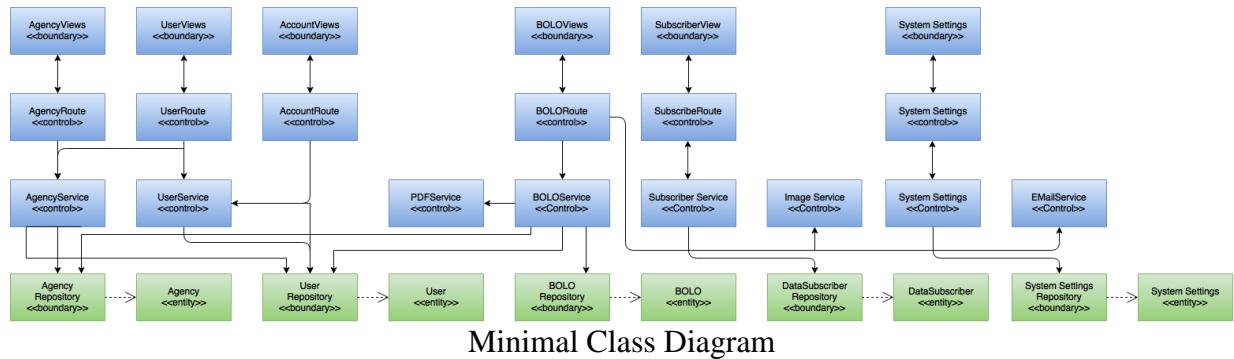
Bolo 6.0 runs on the MongoDB database for document, image, attachment and user storage.



Package diagram



General Deployment Diagram



Minimal Class Diagram

Design Patterns

The following design patterns were used during version 6 iteration of the BOLO Flyer Creator system:

- Observer: This is essential to maintain an organized execution of asynchronous calls in order to prevent the system from malfunctioning.
- Singleton: Only one instance of the server will be allowed to run at a time
- Repository: This abstraction is used to mediate between the domain and the persistent data storage so that if the database service was to be changed, it can be done without altering the entire code.

System Validation

Unit test is essential to test all the layers within the application. All test cases were used to help validate user stories within the project plan section.

ID	BOLO_ImageCompression_1
Purpose	Sunny-day test to check if a large .jpeg image (>1 megabyte) is compressed to less than 500kb
Precondition	The user is signed in as an administrator The node app is running The node app is connected to database
Input	*.jpeg image of size 1.38 megabytes
Expected Output	“Image is of size [<500KB]”

ID	BOLO_ImageCompression_2
Purpose	Sunny-day test to check if a small .jpeg image (<1 megabyte) is compressed to less than 500kb
Precondition	The user is signed in as an administrator The node app is running The node app is connected to database
Input	*.jpeg image of size 741 kilobytes
Expected Output	“Image is of size [<500KB]”

ID	BOLO_ImageCompression_3
Purpose	Rainy-day test to check if a small .jpeg image (<512 kilobytes) is not compressed
Precondition	The user is signed in as an administrator The node app is running The node app is connected to database
Input	*.jpeg image of size 198 kilobytes

Expected Output	“Image is of size 198KB”
------------------------	---------------------------------

ID	BOLO_CreateUser_1
Purpose	Sunny-day test to create a new officer user
Precondition	<p>A user is signed in as an administrator</p> <p>The node app is running</p> <p>The node app is connected to database</p> <p>The user does not exist on the database</p> <p>Agency Pinecrest exists on the server</p>
Input	<p>Username: abc123</p> <p>First Name: bob</p> <p>Last Name: Test</p> <p>Agency: Pinecrest</p> <p>Email: abc123(@pine.gov)</p> <p>Role: Officer</p> <p>Badge No.: 1234</p> <p>Section / Unit: 1234</p> <p>Rank / Title: Test</p>
Expected Output	“User abc123 has been registered!”

ID	BOLO_CreateUser_2
Purpose	Sunny-day test to create a new root user
Precondition	<p>The user is signed in as a root user</p> <p>The node app is running</p> <p>The node app is connected to database</p> <p>The user does not exist on the database</p> <p>Agency Raymond exists on the server</p>
Input	Username: domin1

	First Name: Dom Last Name: Mar Agency: Raymond Email: abc-123(@gmail.com) Role: Root Badge No.: 1234 Section / Unit: 1234 Rank / Title: Test
Expected Output	“User domin1 has been registered!”

ID	BOLO_CreateUser_3
Purpose	Rainy-day test to check if the system will catch a validation error of an incorrect email
Precondition	The user is signed in as a root user The node app is running The node app is connected to database The user does not exist on the database Agency Raymond exists on the server
Input	Username: do@#m First Name: Dom Last Name: Mar Agency: Raymond Domain: ab^c-123(@gmail.com) Role: Supervisor Badge No.: 1234 Section / Unit: 1234 Rank / Title: Test
Expected Output	“Email is not Valid”

ID	BOLO_DeleteUser_1
Purpose	Sunny-day test to check if an existing officer on the database is removed after password conformation
Precondition	The user is signed in as a root user The node app is running The node app is connected to database The user ‘deleteMe’ exists on the database
Input	Select ‘Delete User’ on ‘deleteMe’ user details page Enter correct password
Expected Output	“User ‘deleteMe’ has been deleted”

ID	BOLO_DeleteUser_2
Purpose	Rainy-day test to check if an existing officer on the database fails to be removed after password is entered incorrectly
Precondition	The user is signed in as a root user The node app is running The node app is connected to database The user ‘deleteMe’ exists on the database
Input	Select ‘Delete User’ on ‘deleteMe’ user details page Enter incorrect user password
Expected Output	“Password was incorrect. Please try again”

ID	BOLO_CreateCategory_1
Purpose	Sunny-day test to check if the system will save a new category correctly
Precondition	The user is signed in as a root user The node app is running The node app is connected to database The category Car already exists on the database
Input	Enter A Name For The New BOLO Category: Tester

	Enter Fields {abc, qwe, asd, zxc, xyz}
Expected Output	“Category Tester has been created”

ID	BOLO_CreateCategory_1
Purpose	Rainy-day test to check if the system will not save a new category with the same name
Precondition	The user is signed in as a root user The node app is running The node app is connected to database The category Car already exists on the database
Input	Enter A Name For The New BOLO Category: Car Enter Fields {abc, qwe, asd, zxc, xyz}
Expected Output	“Category Car already exists on the database”

Glossary

Administrator: A police agent that operates from a desk coordinating and overviewing tasks.

Agency: A district-based police department.

BOLO: A “Be On the Look Out” flyer that contains information about a specific crime.

Data Subscriber: A user that is notified when a BOLO has been updated

Officer: A police agent assigned to a crime scene.

Supervisor: A police field operator in charge of multiple cases.

Root Administrator: An officer that has universal access to the system. They can delete users, agencies, and bolas. They can also edit selected webpages on the website.

Appendix

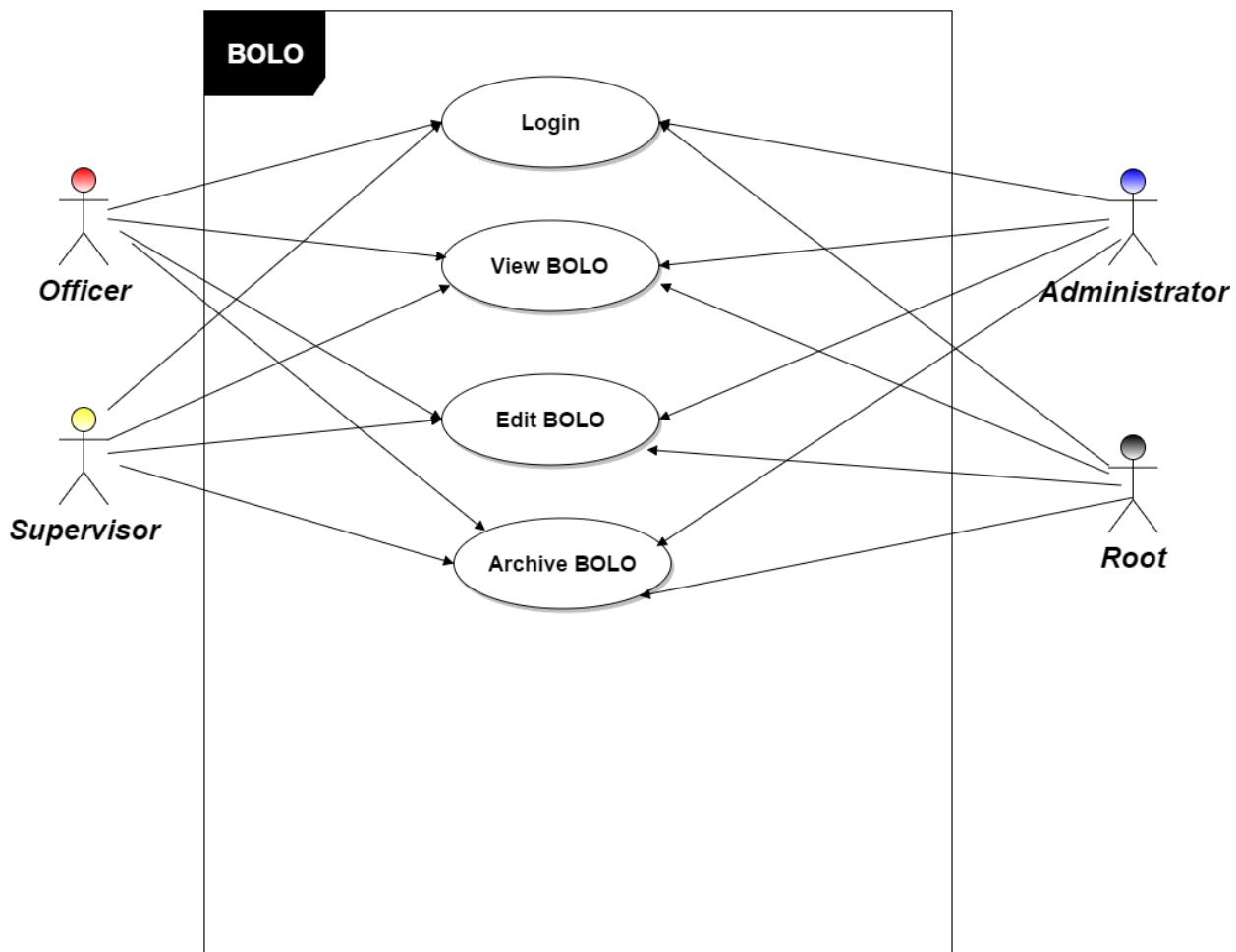
Appendix A – UML Diagrams

Static UML Diagrams

Sprint 1
(08/27/2016 - 09/09/2016)

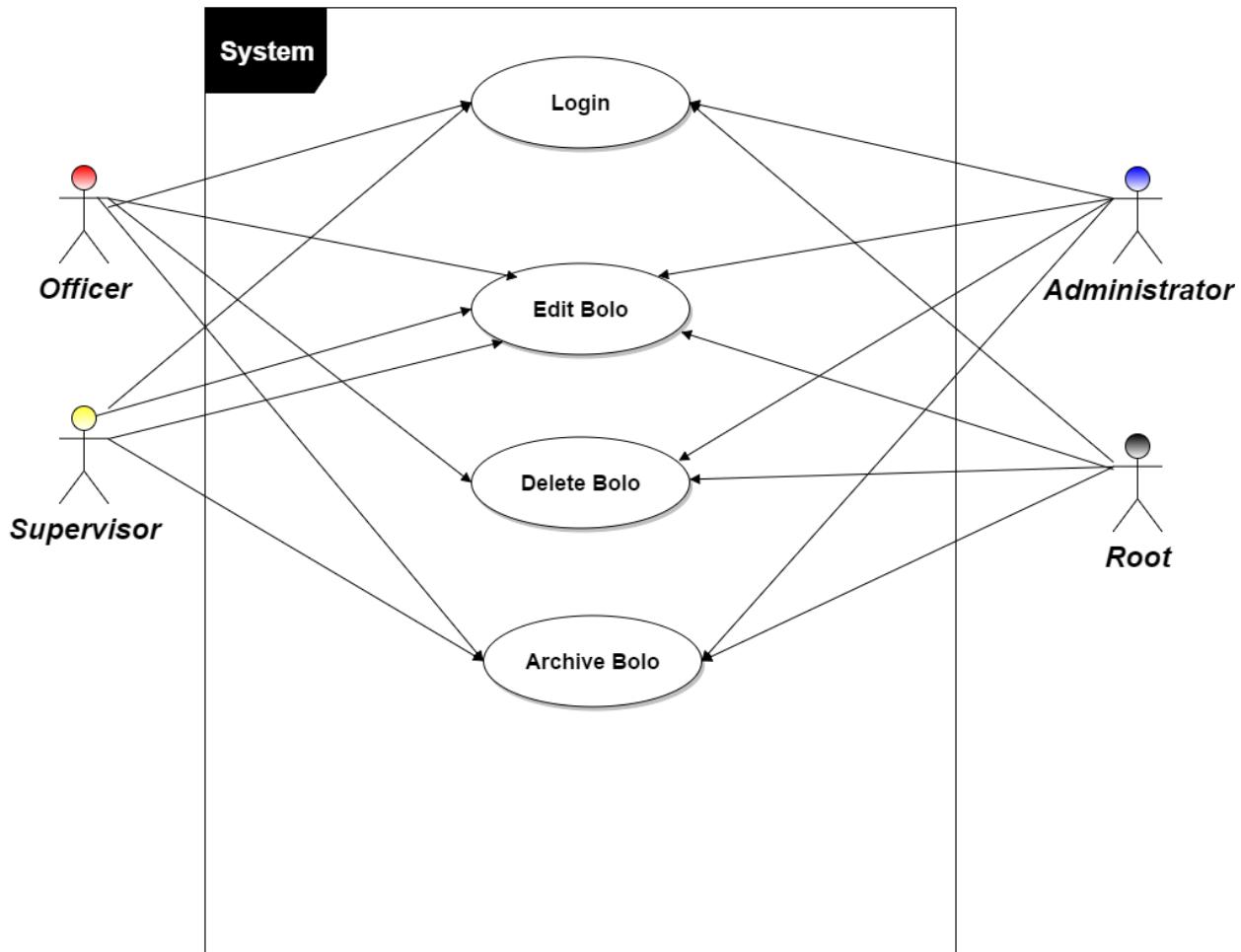
User Story #1177 View All Pictures on BOLO Details View

- Figure sp1-1

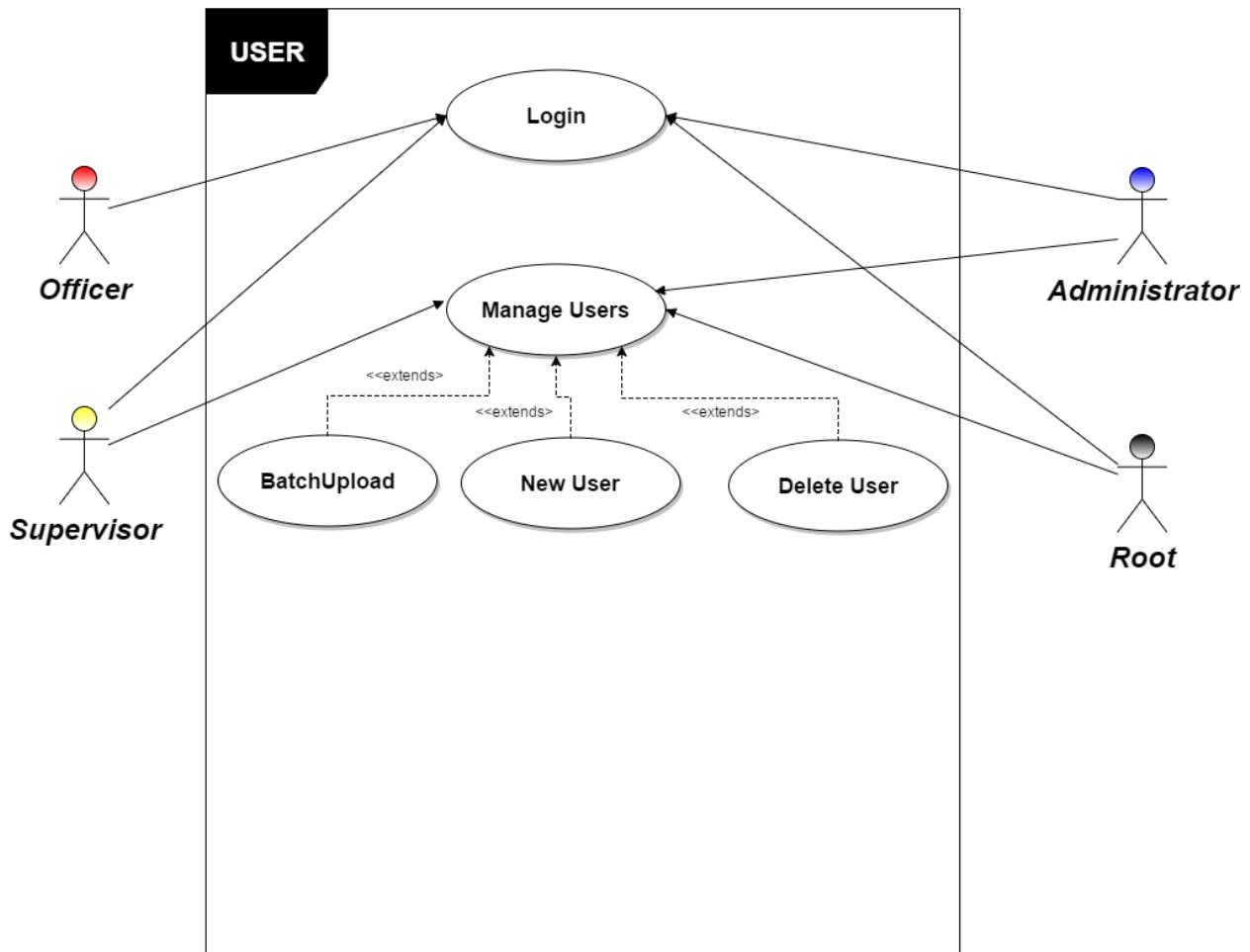


User Story #1187 Generate Watermark on BOLOs

- Figure sp1-2

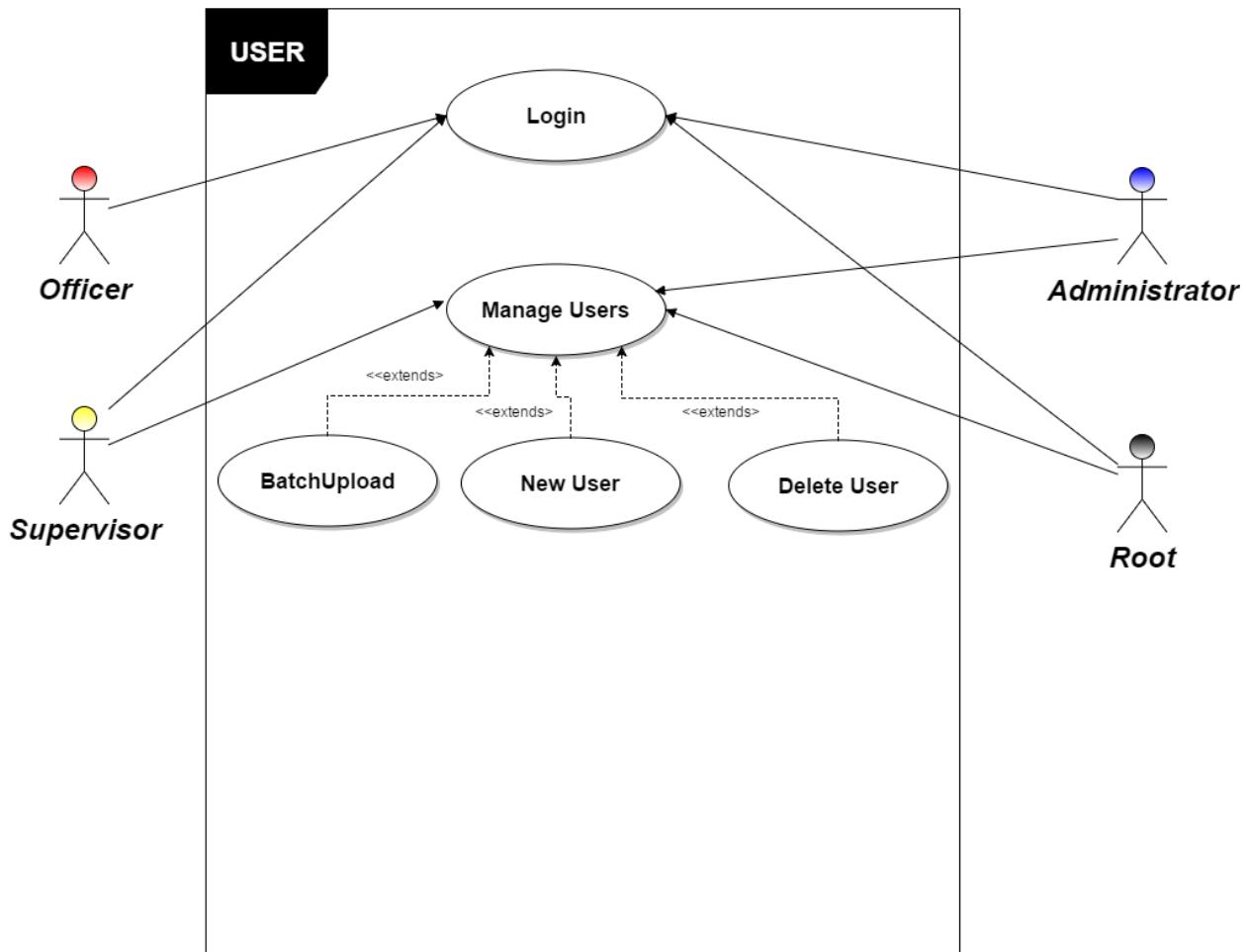
**User Story #1179** Batch Upload of New Users

- Figure sp1-3



User Story #1186 New User Button

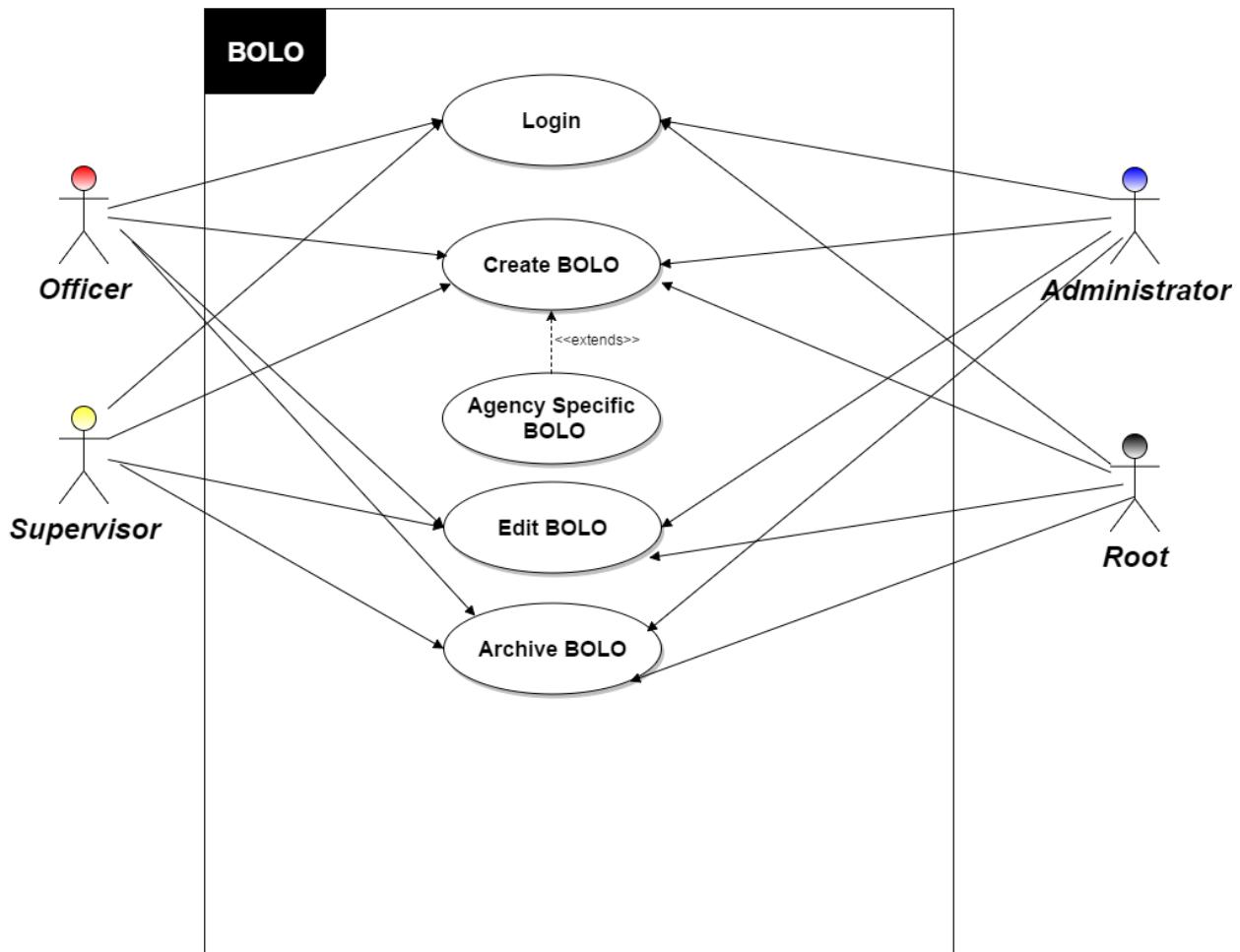
- Figure sp1-4



Sprint 2
(09/10/2016 - 09/23/2016)

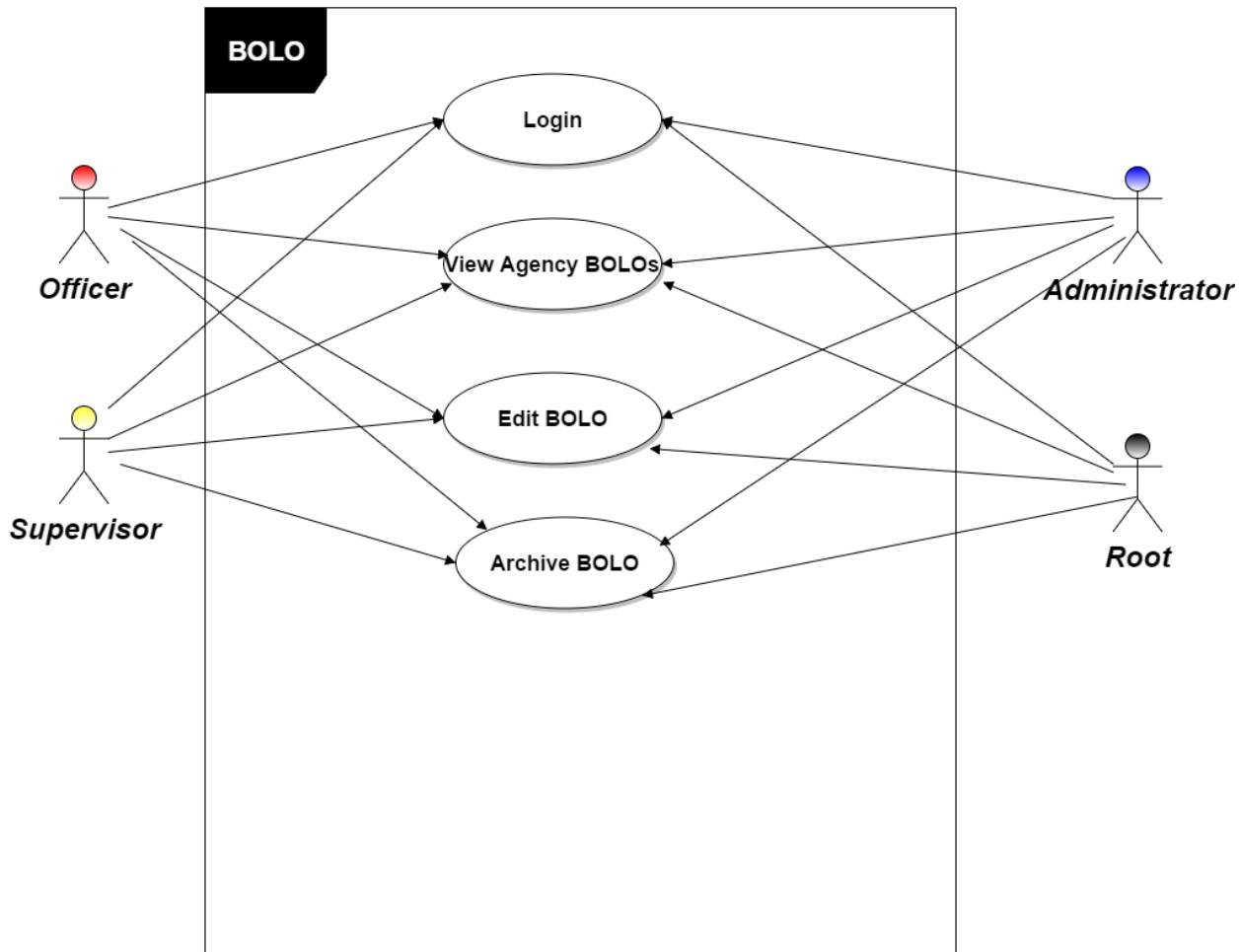
User Story #1189 Agency Specific BOLO

- Figure sp2-1



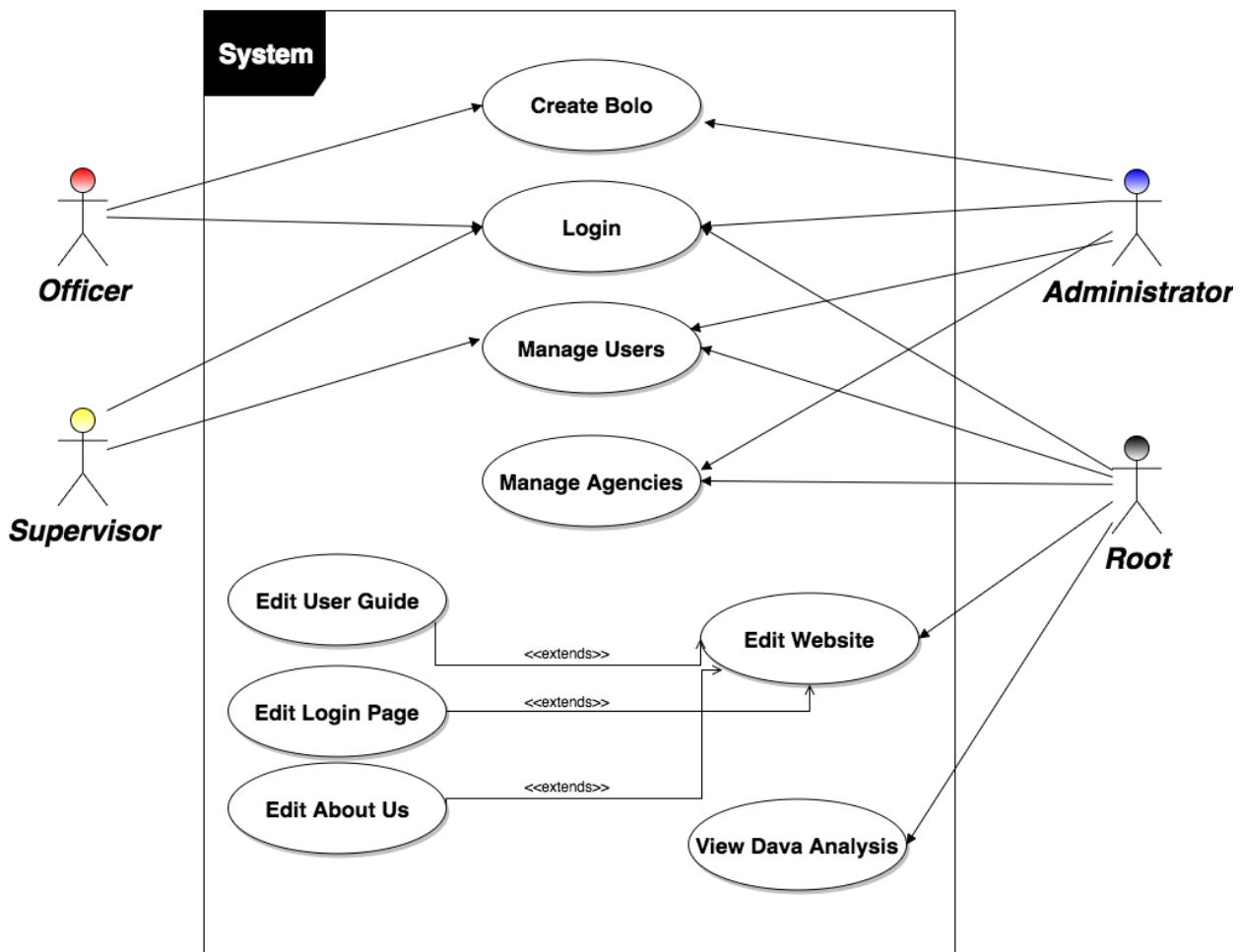
User Story #1190 Agency Specific View

- Figure sp2-2



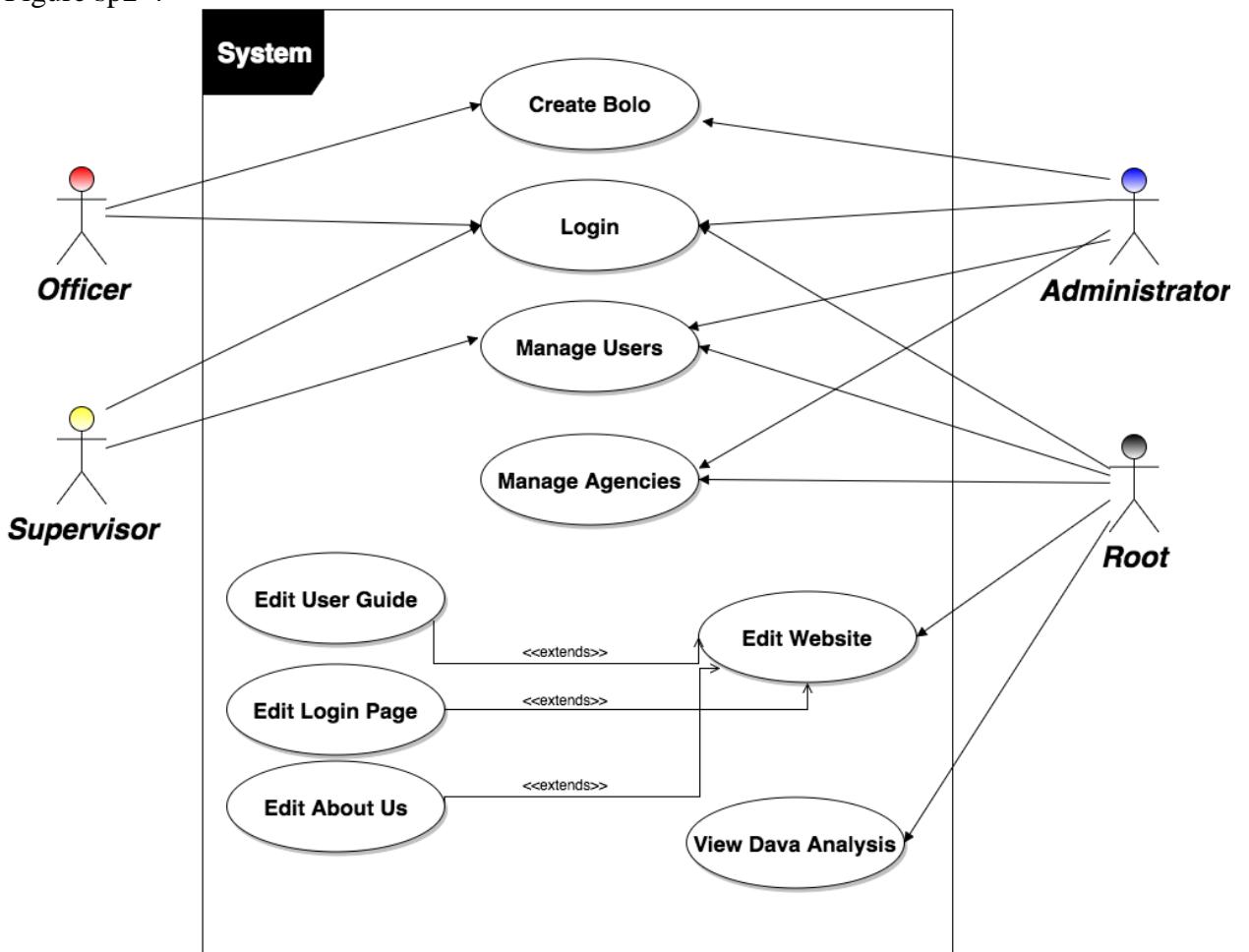
User Story 1182 Edit About Us Page

- Figure sp2-3



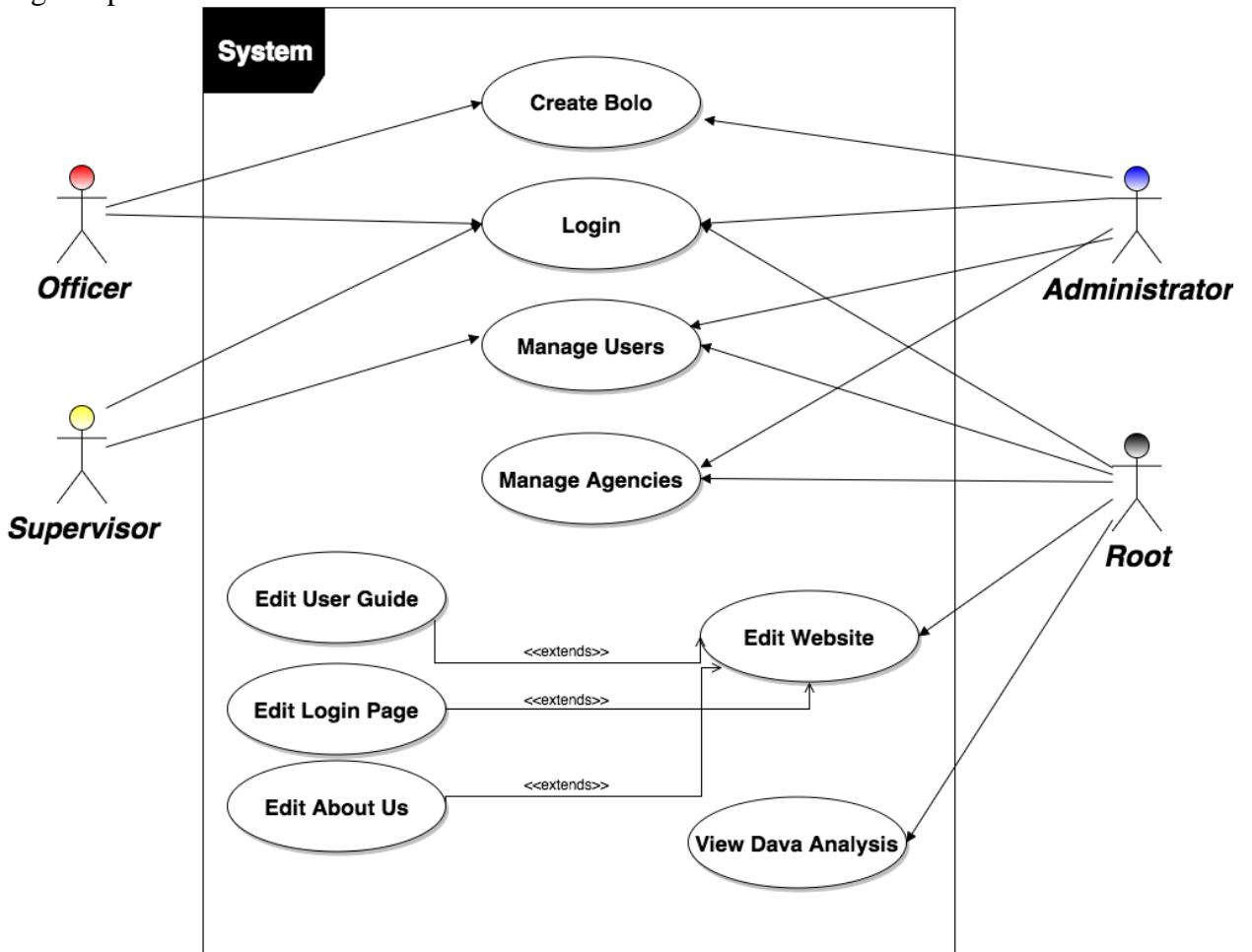
User Story 1183 Edit Login Page

- Figure sp2-4



User Story 1184 Edit User Guide Page

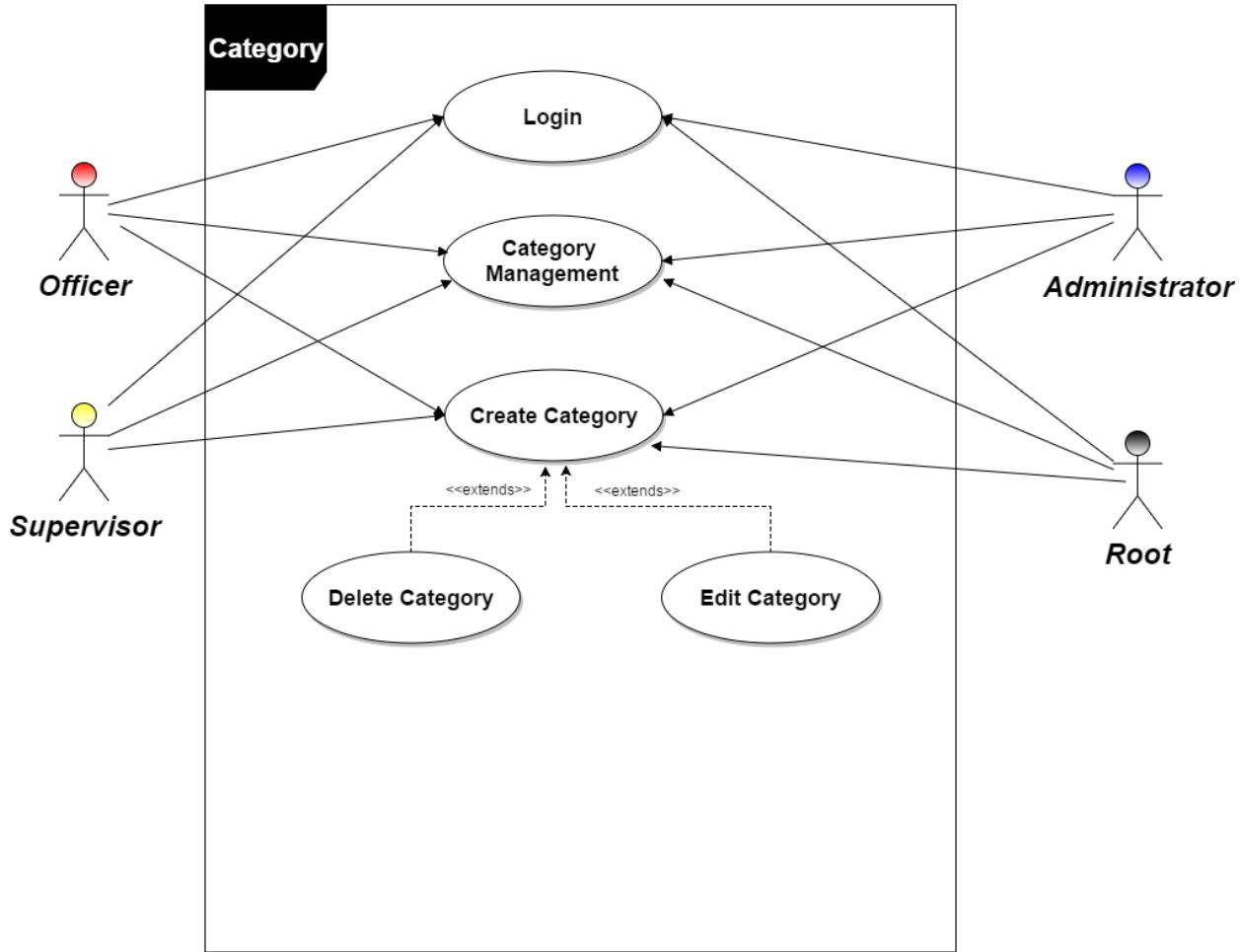
- Figure sp2-5



Sprint 3
(09/24/2016 – 10/07/2016)

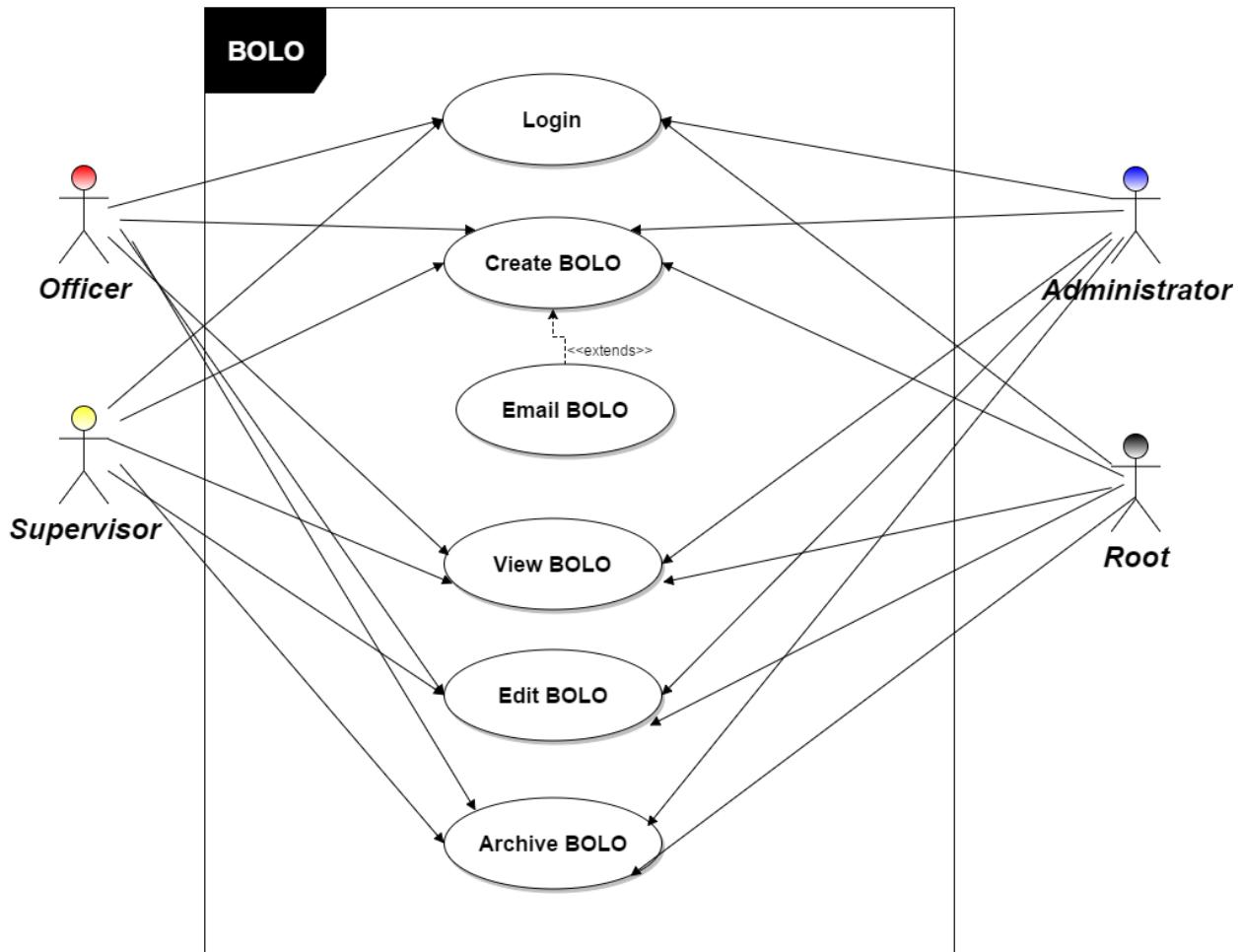
User Story #1194 BOLO Category Template Creation

- Figure sp3-1



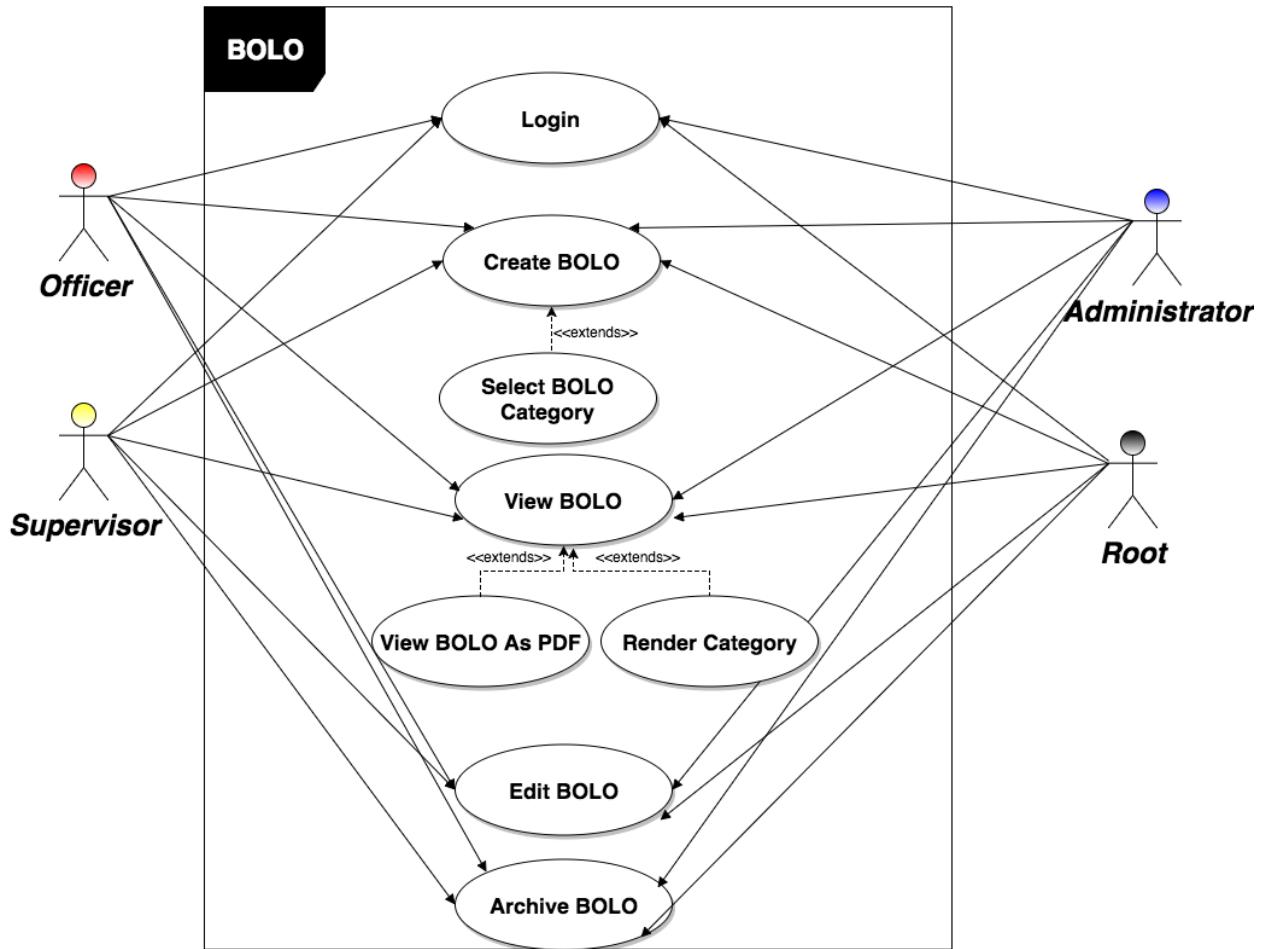
User Story #1191 Email PDF Privacy

- Figure sp3-2



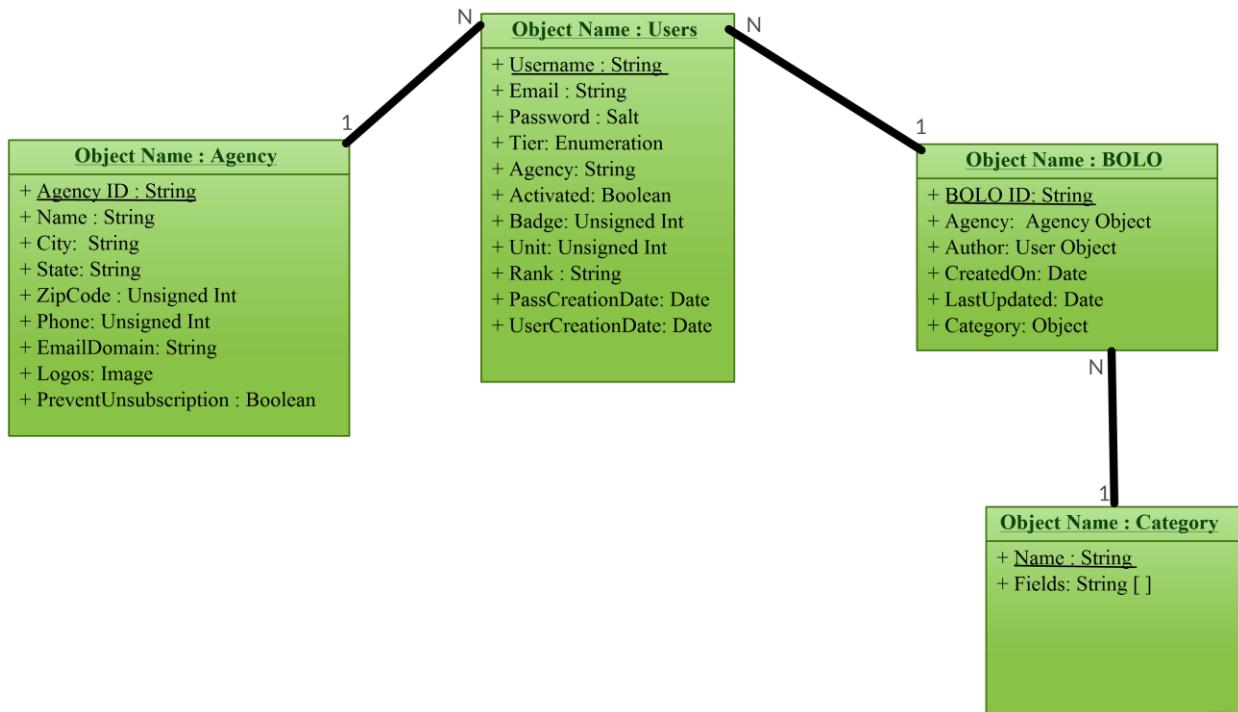
User Story #1196 Template Rendering

- Figure sp3-3



User Story #1188 Convert Database to MongoDB

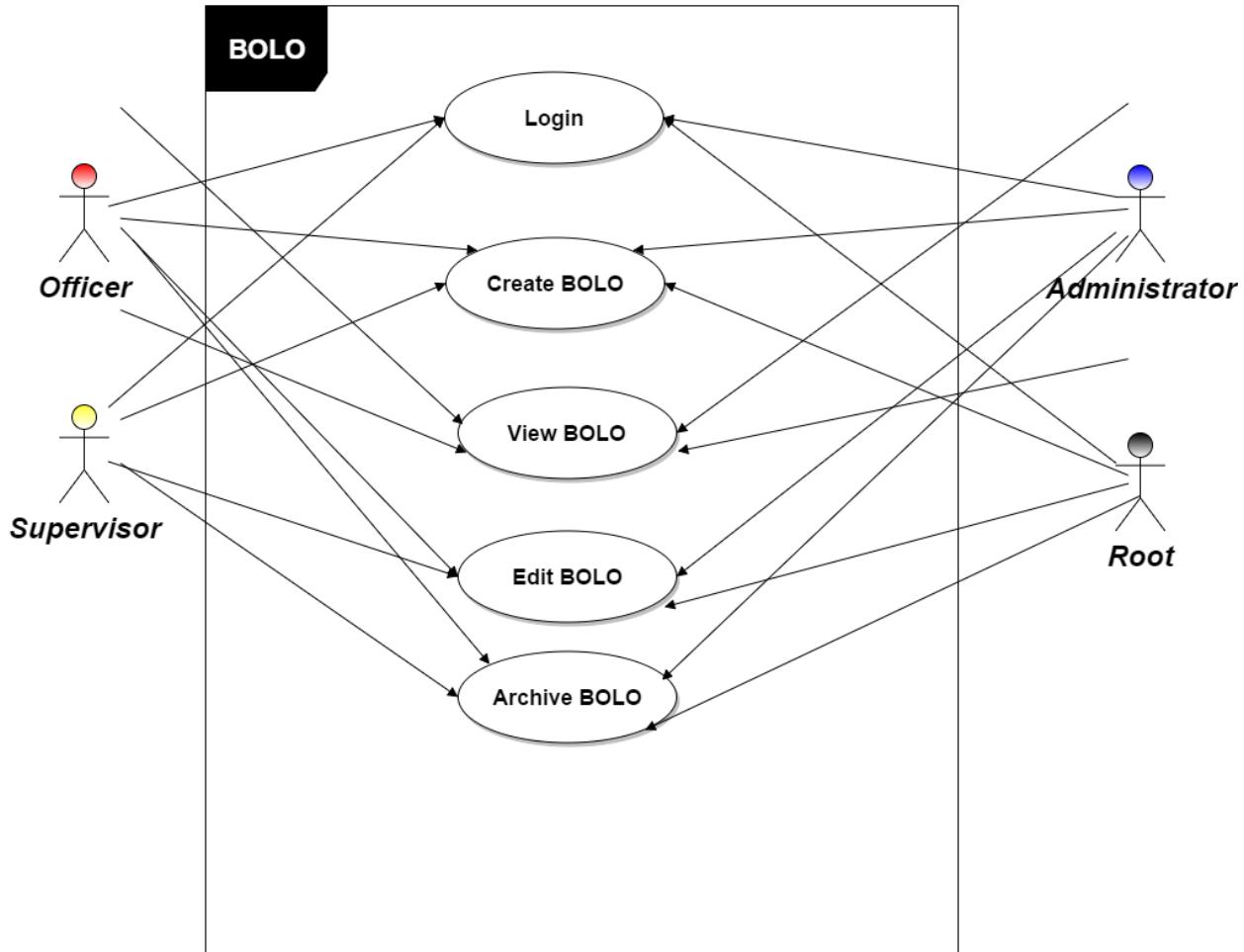
- Figure sp3-4

BOLO6 -- MongoDB ER Diagram

Sprint 4
(10/08/2016 – 10/21/2016)

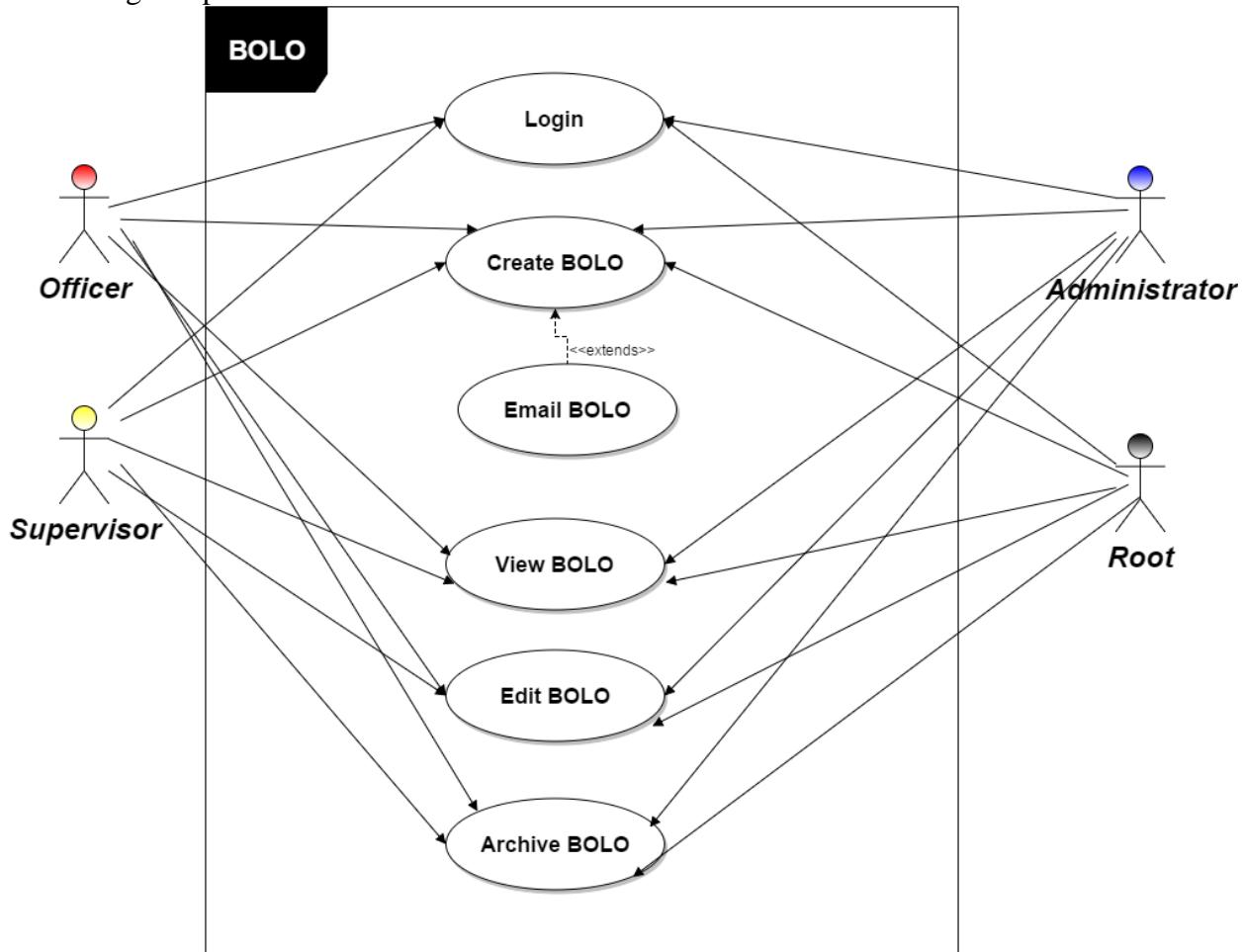
User Story #1198 Adjust BOLO View

- Figure sp4-1



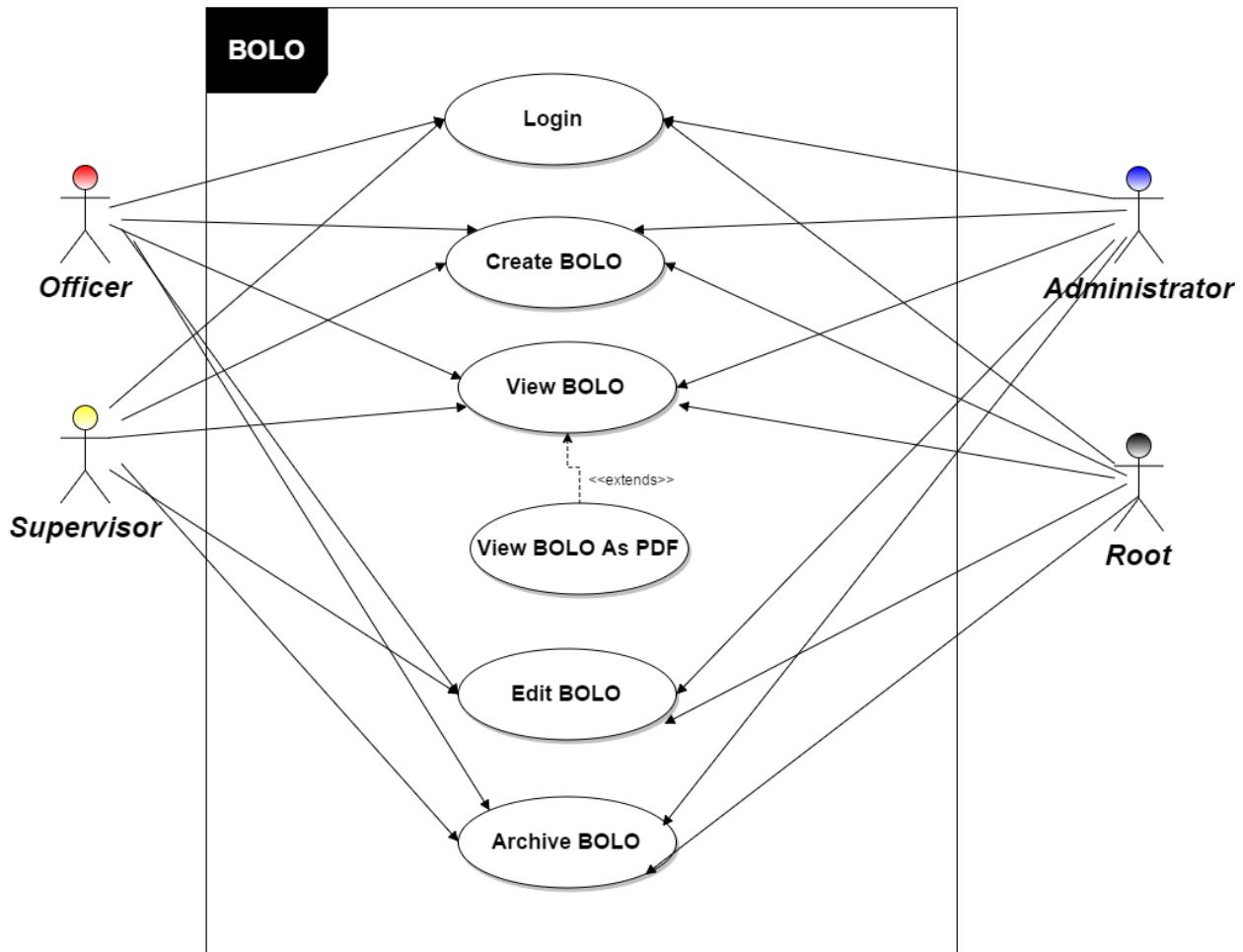
User Story #1200 Adjust Email PDF Privacy for MongoDB

- Figure sp4-2



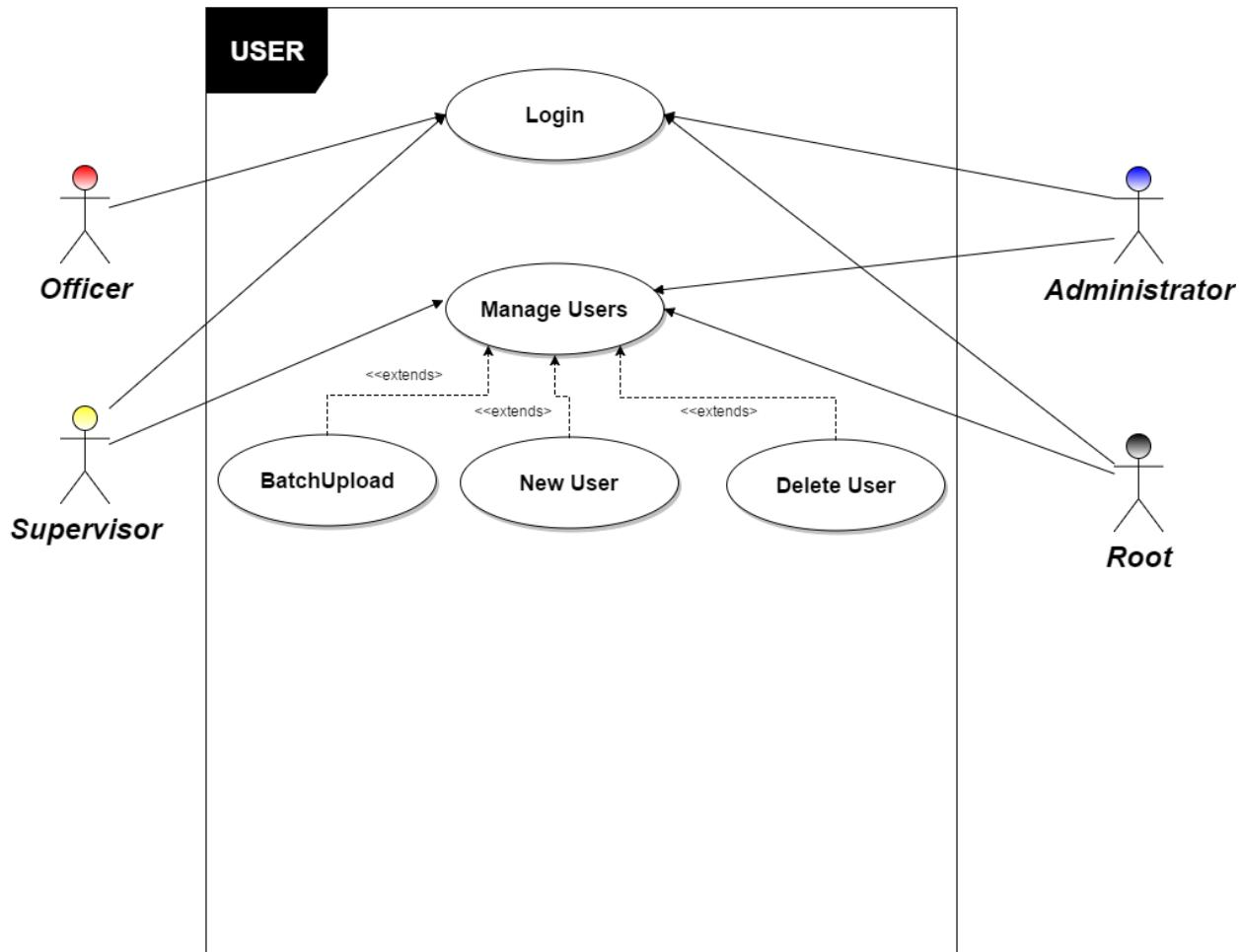
User Story #1199 Adjust PDF Rendering for MongoDB

- Figure sp4-3



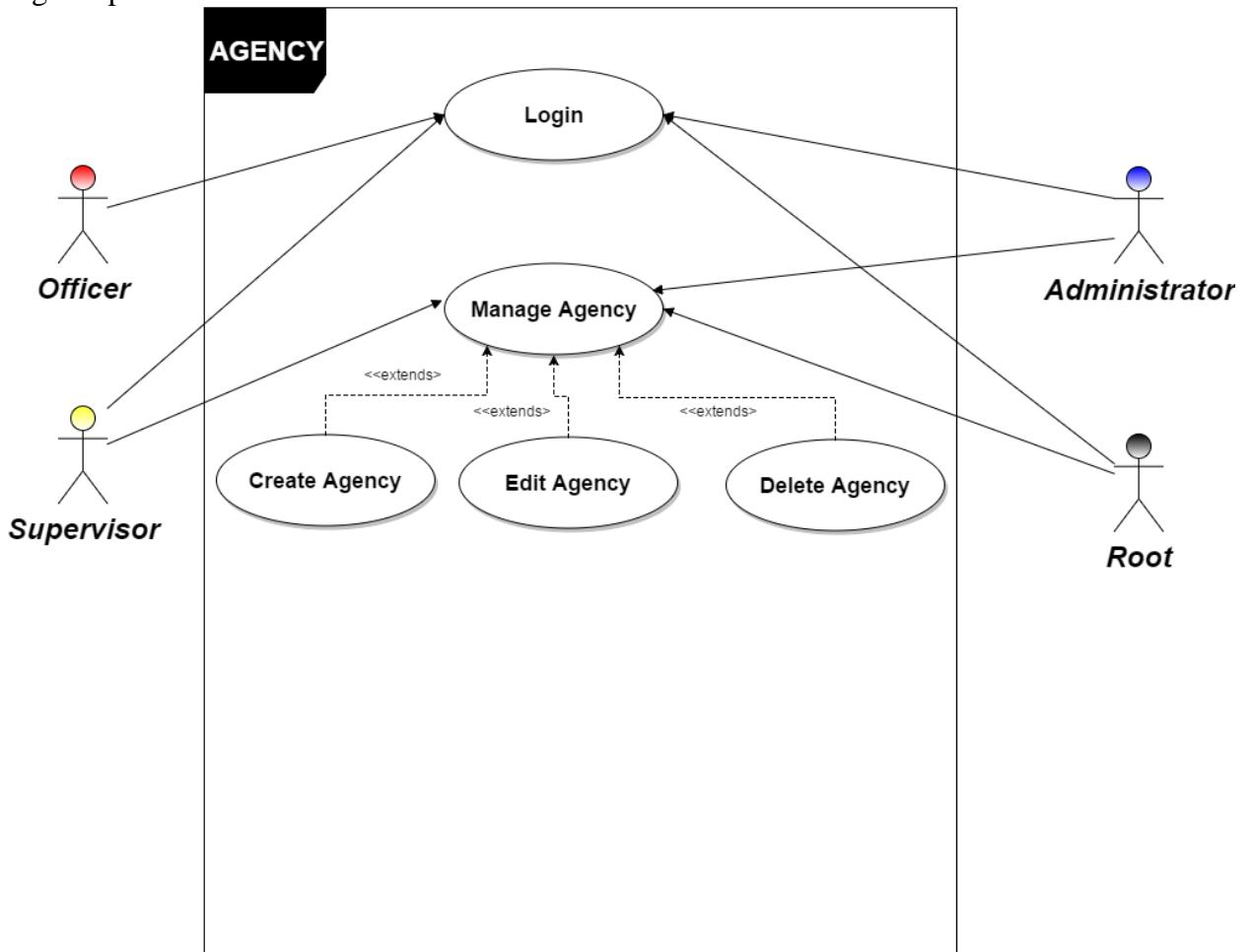
User Story #1195 CSV Upload

- Figure sp4-4



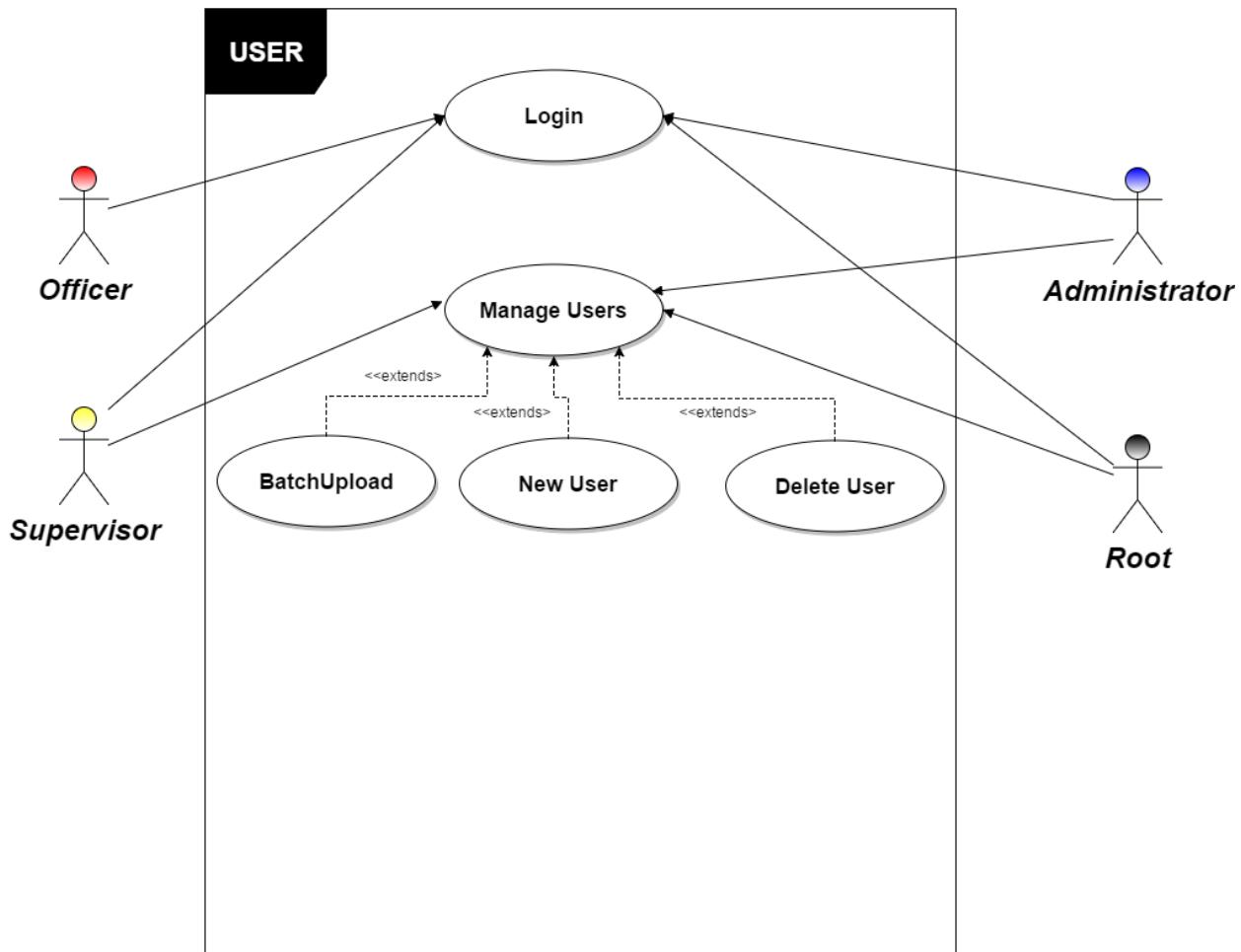
User Story #700 Delete Agency

- Figure sp4-5



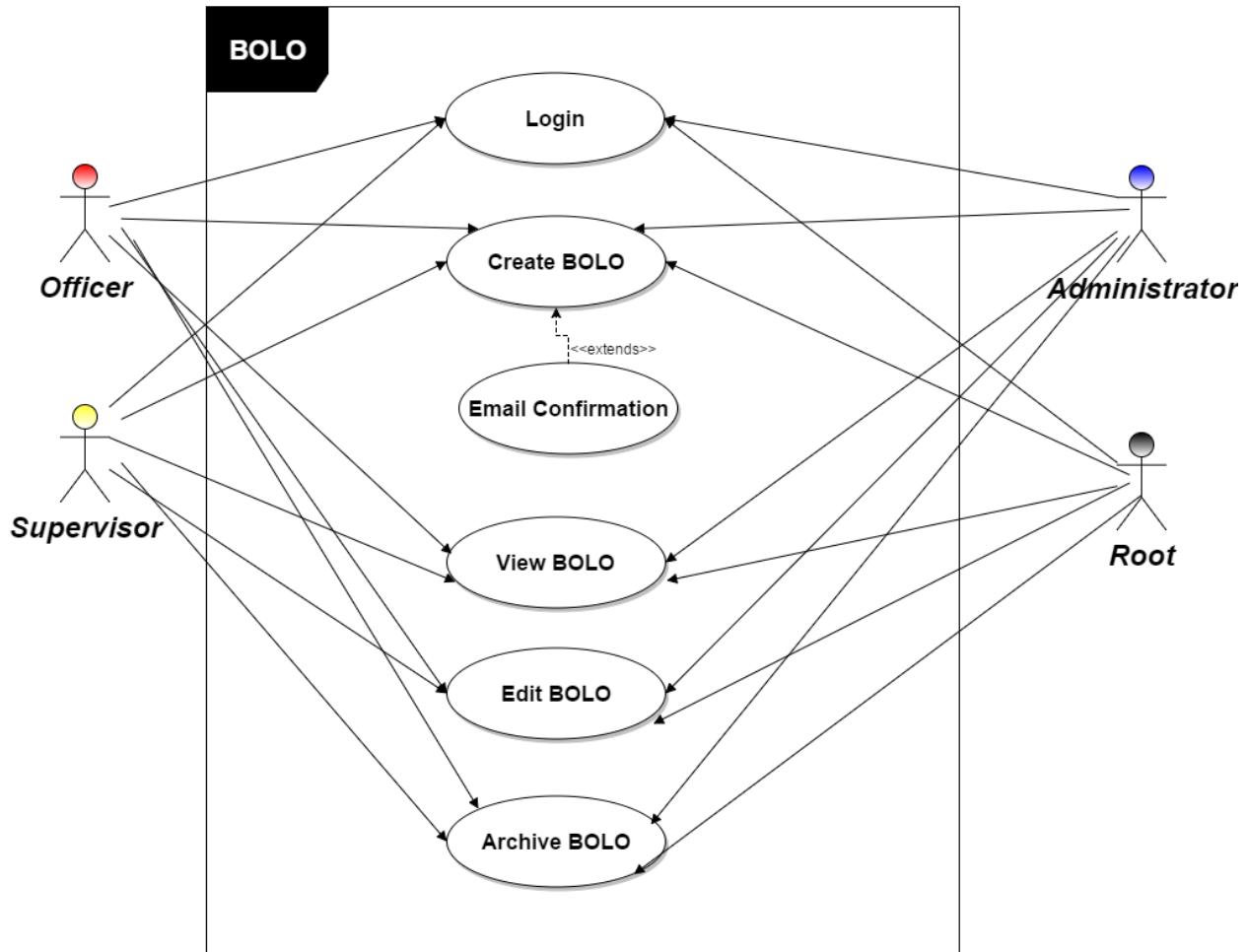
User Story #1192 Delete User

- Figure sp4-6



User Story #1197 Email Confirmation

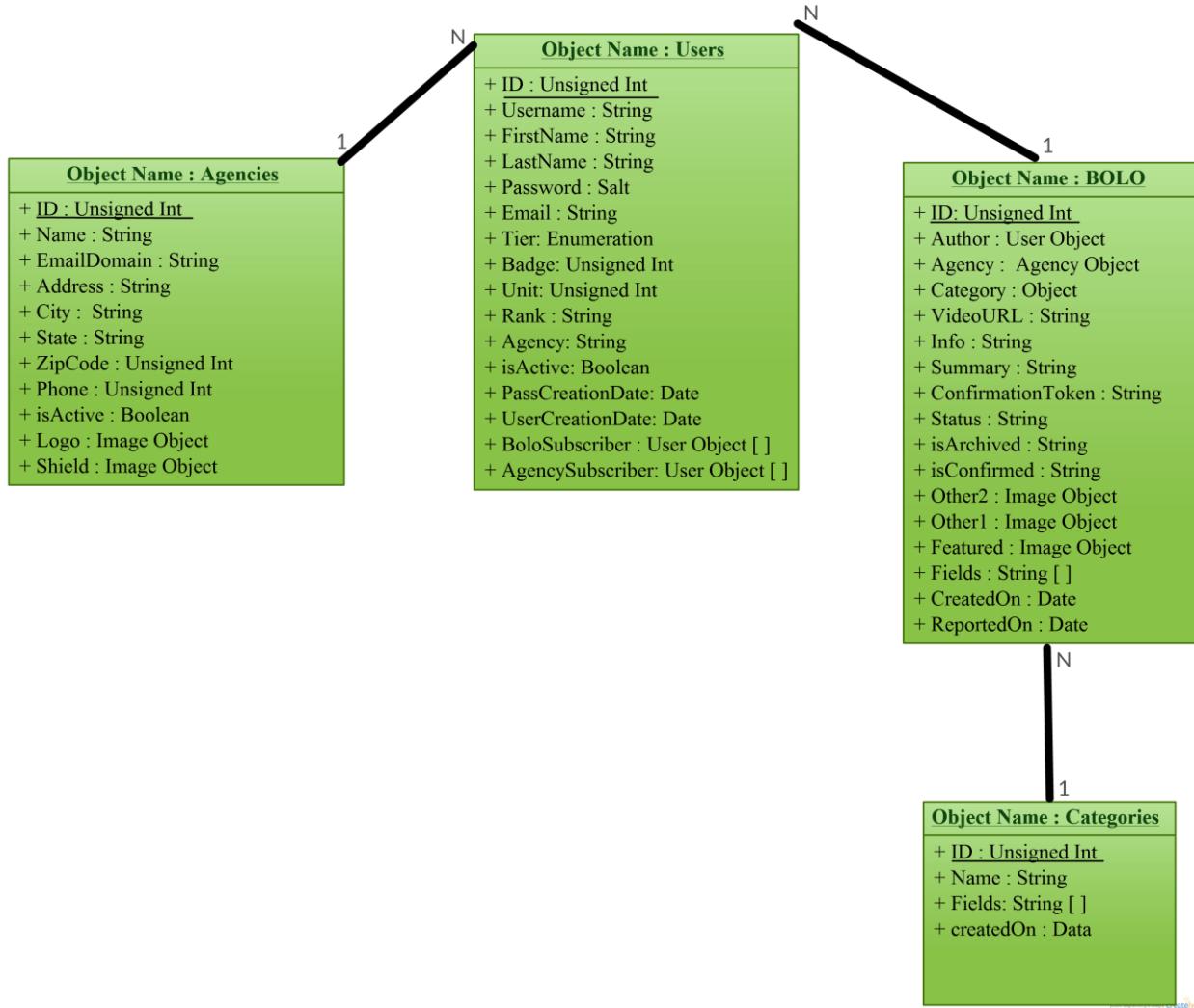
- Figure sp4-7



User Story #1188 Convert Database to MongoDB

- Figure sp4-9

BOLO6 -- MongoDB ER Diagram Version 2



Sprint 5
 (10/22/2016 – 11/04/2016)

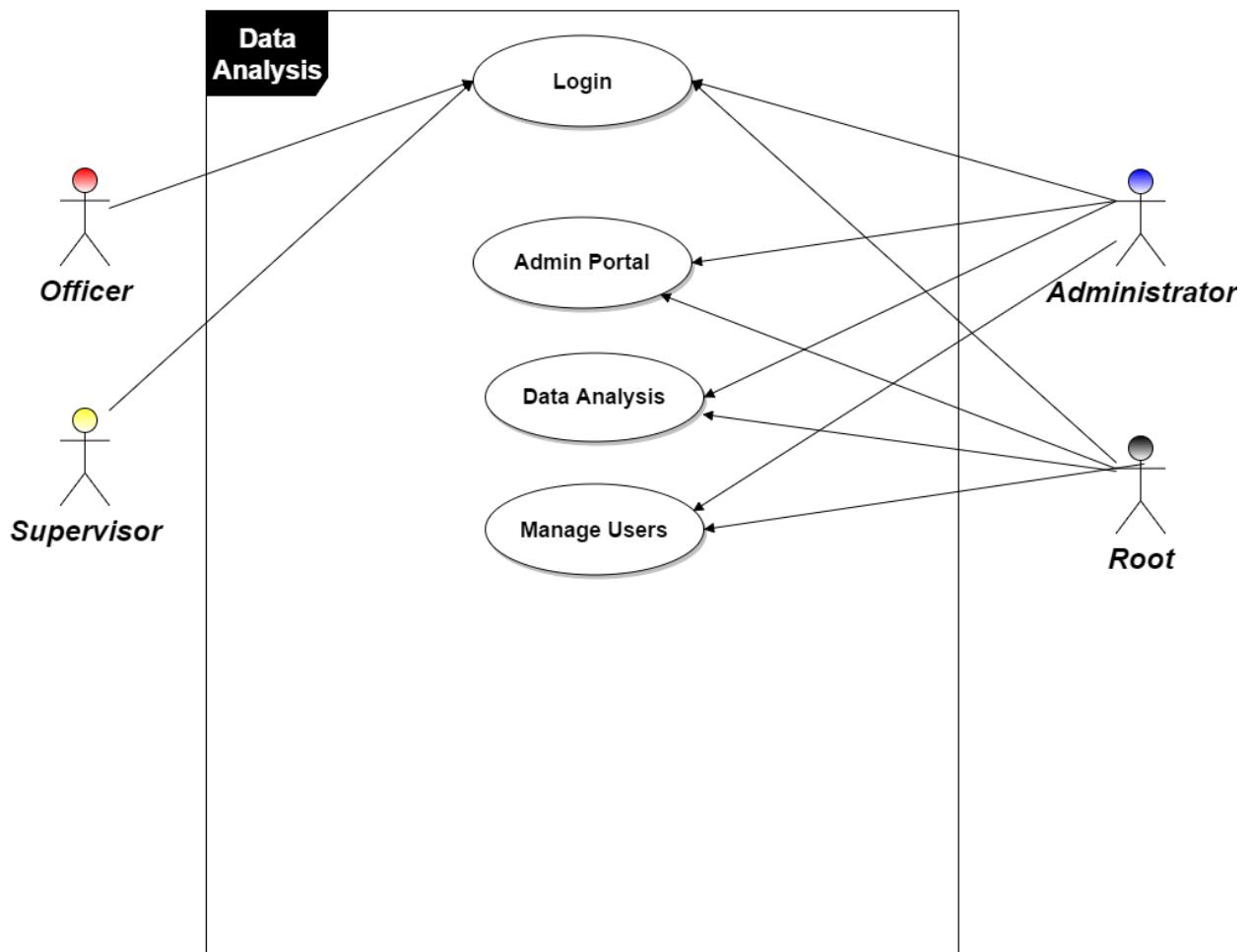
User Story #1193 Image Compression

- Figure sp5-1



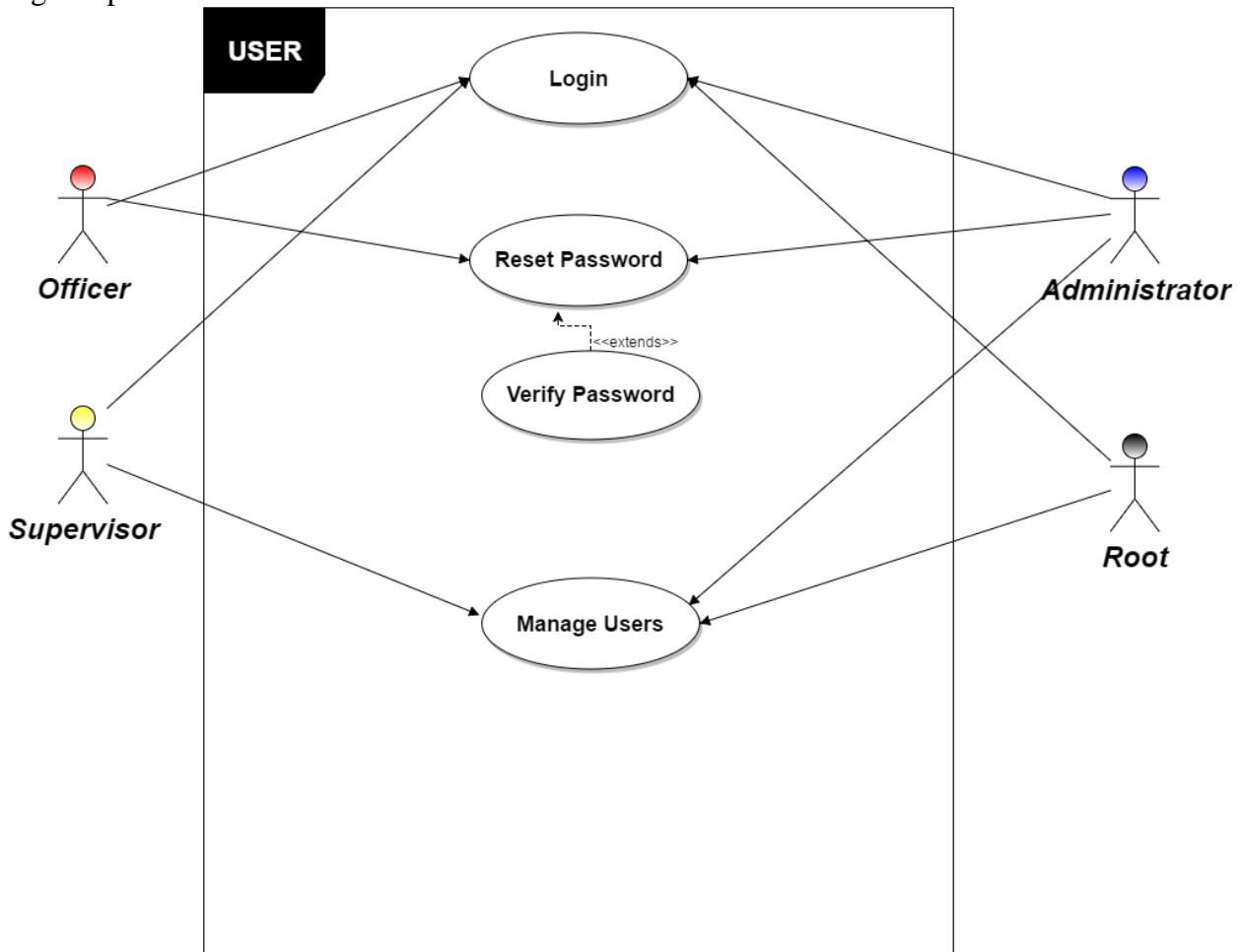
User Story #1204 Data Analysis

- Figure sp5-2



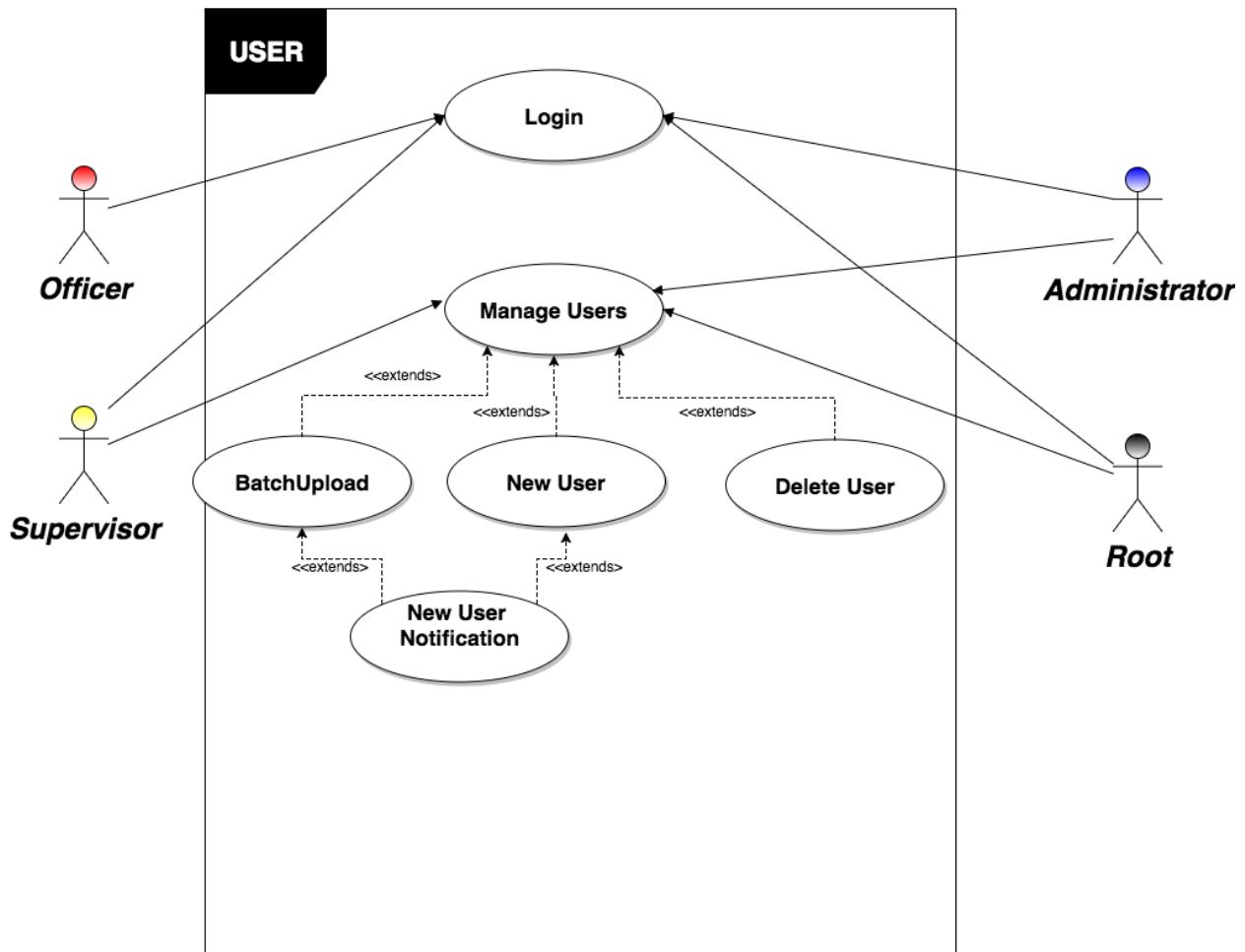
User Story #1180 Set Password

- Figure sp5-3



User Story #1206 New User Notification

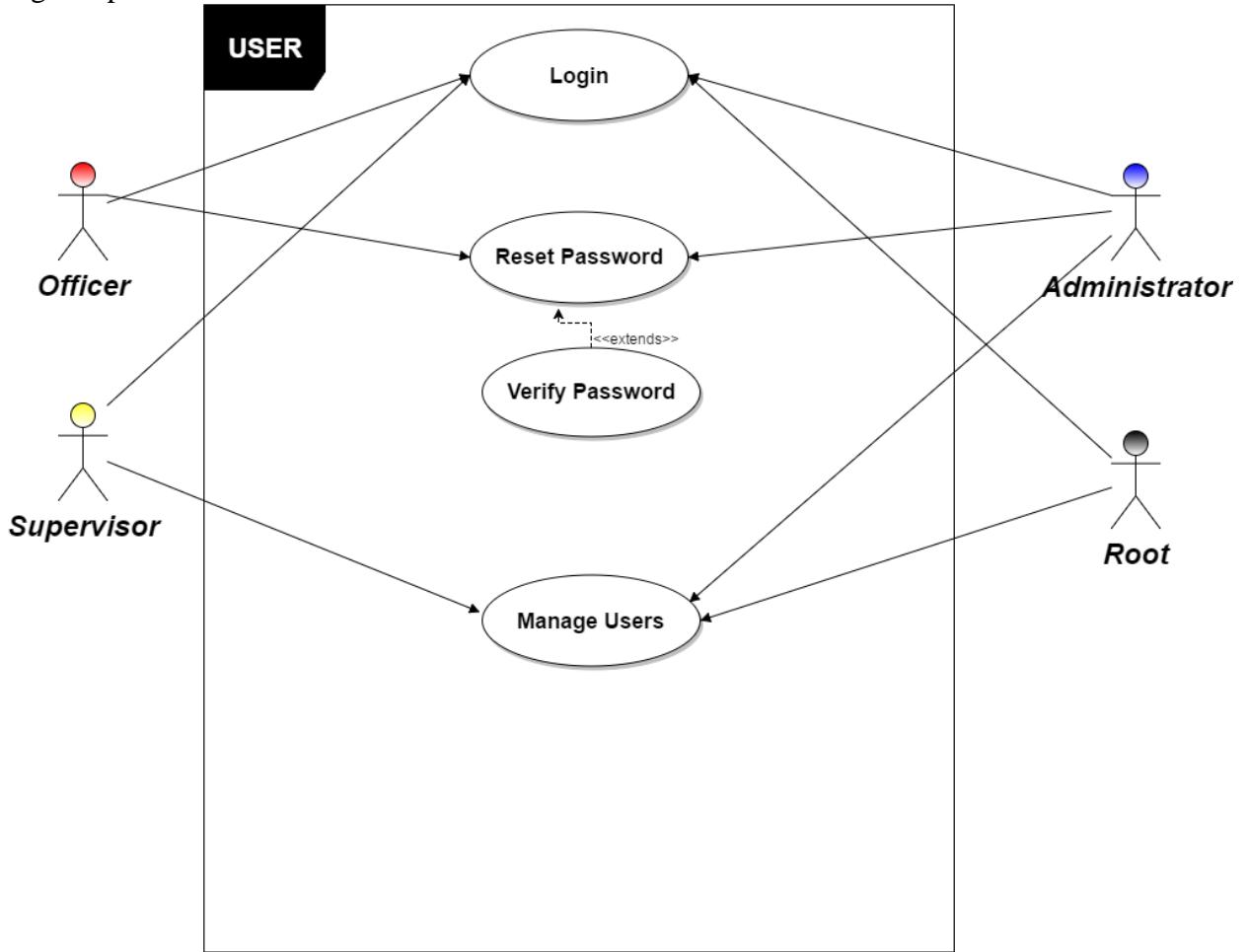
- Figure sp5-4



Sprint 6
(11/05/2016 – 11/11/2016)

User Story #1205 Password Verification

- Figure sp6-1



User Story #1201 BOLO Data Subscriber

- Figure sp6-2

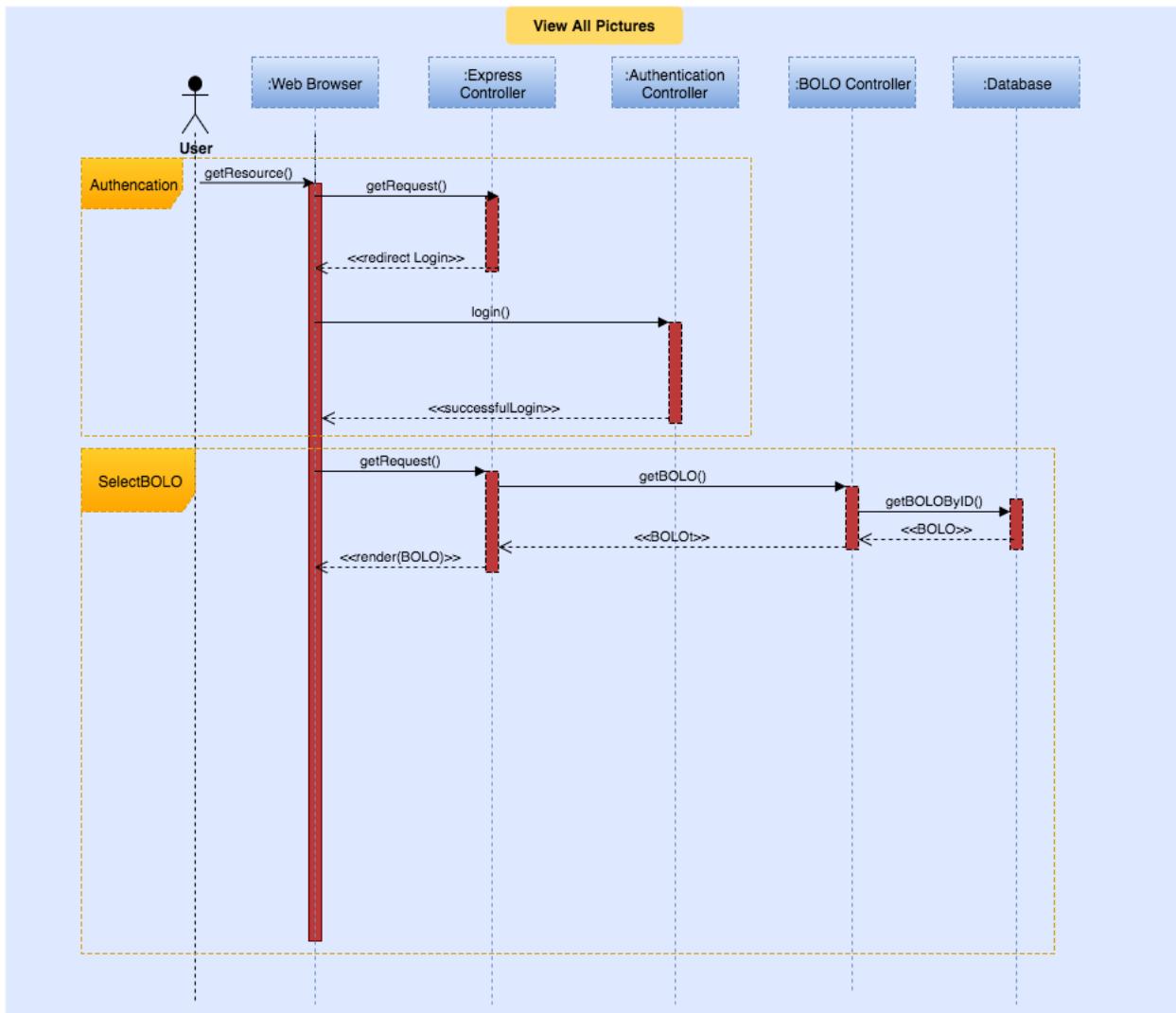


Dynamic UML Diagrams

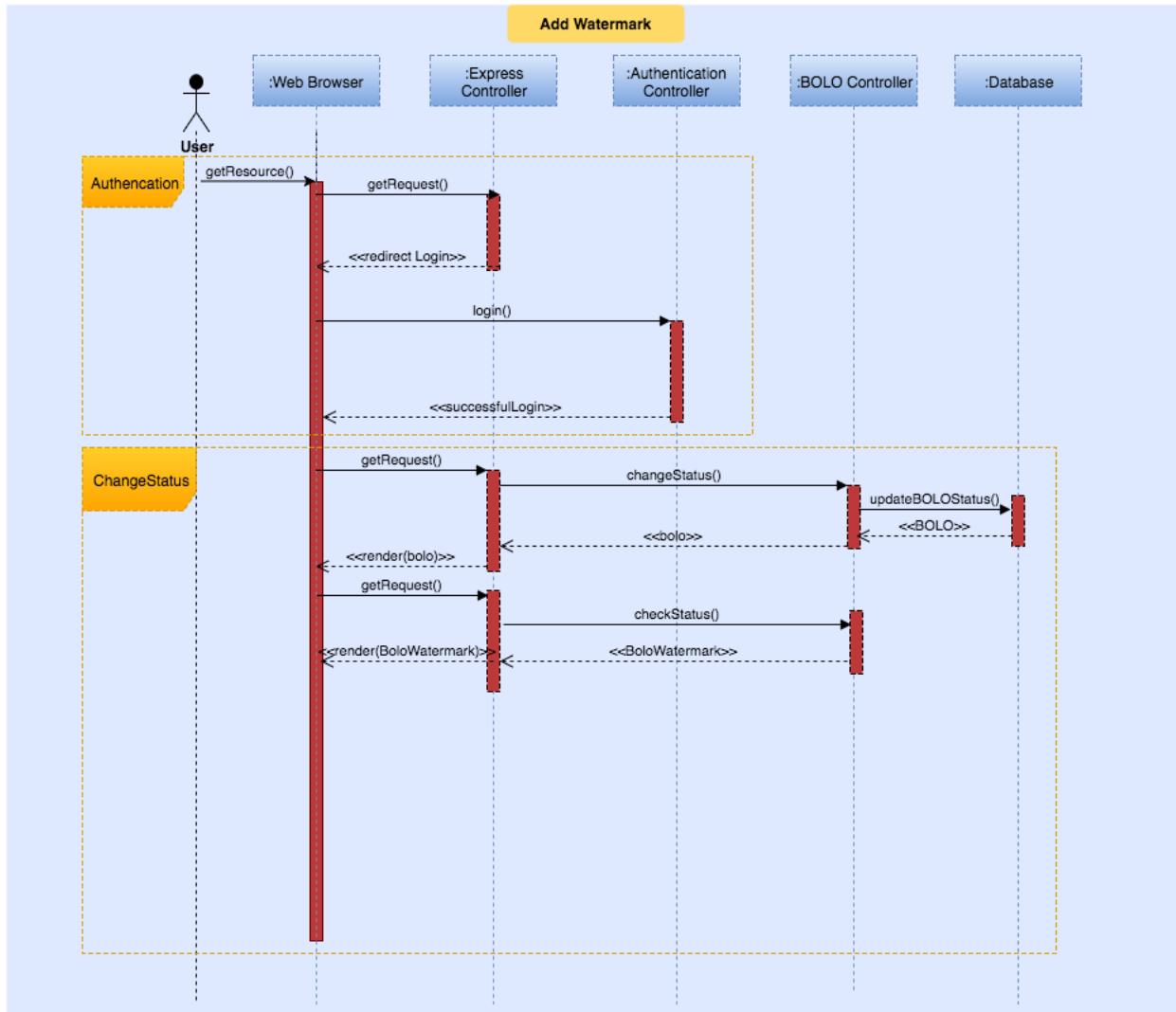
Sprint 1
(08/27/2016 - 09/09/2016)

User Story #1177 View All Pictures on BOLO Details View

- Figure sp1-1

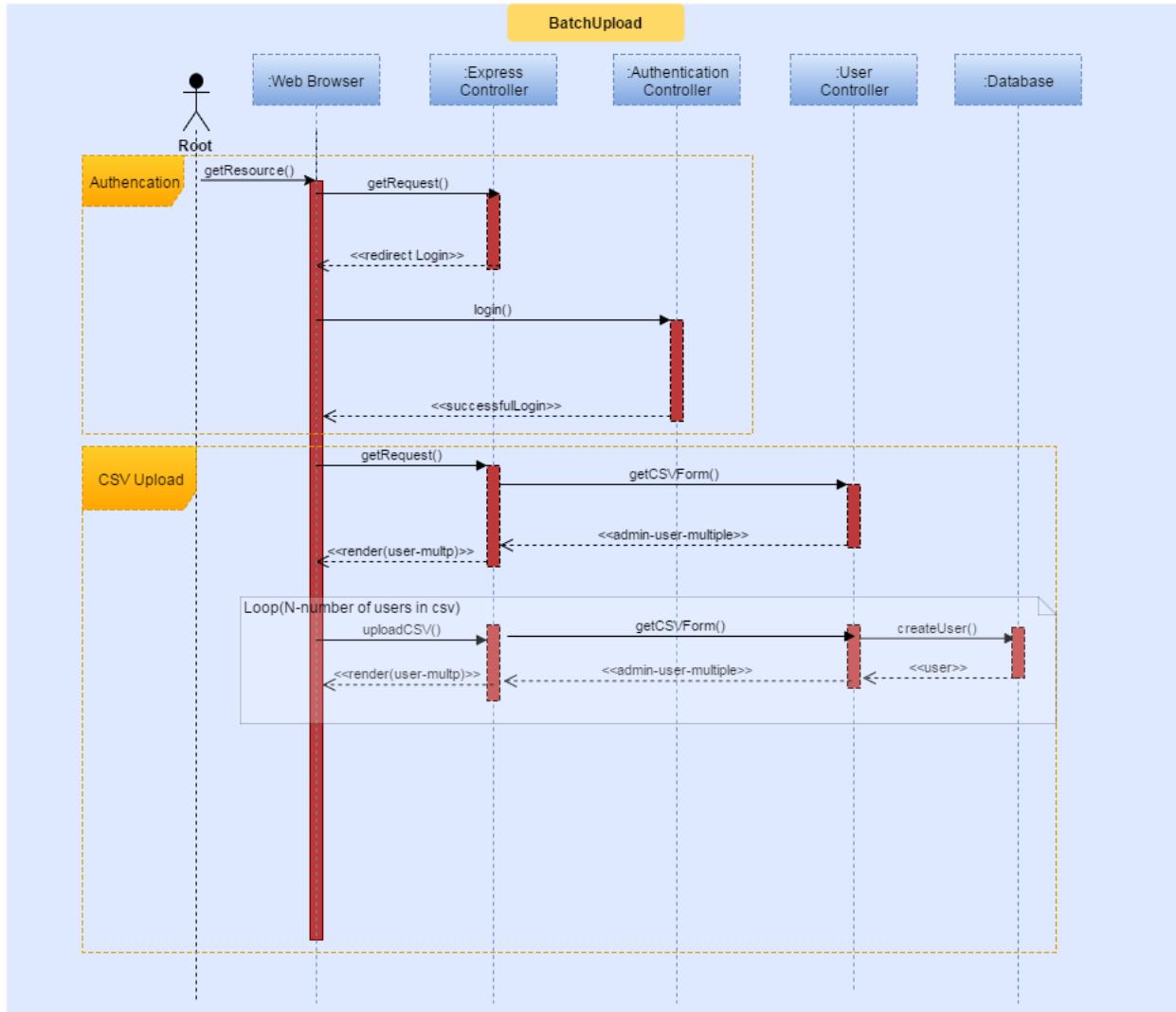
**User Story #1187** Generate Watermark on BOLOs

- Figure sp1-2



User Story #1179 Batch Upload of New Users

- Figure sp1-3



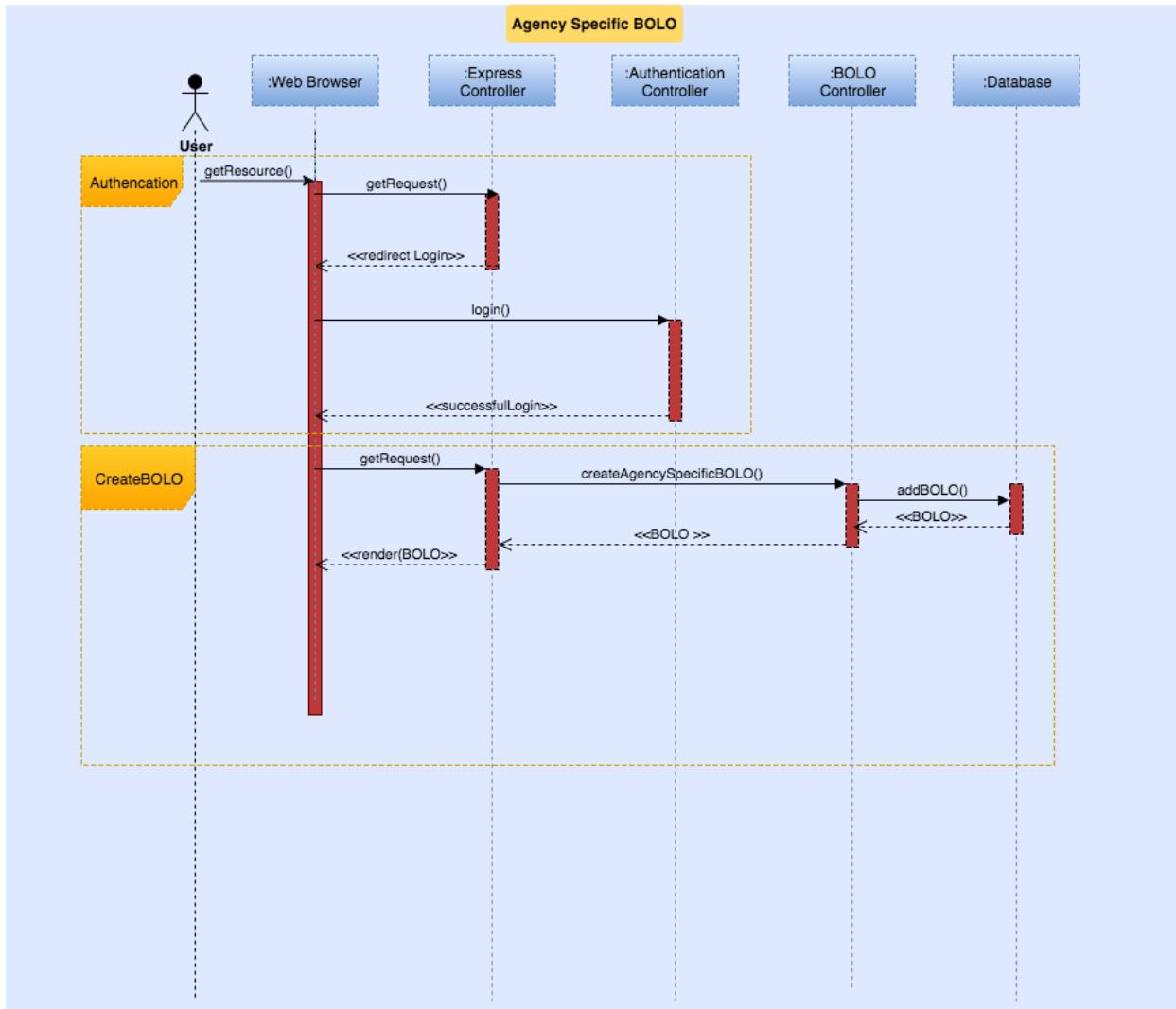
User Story #1186 New User Button

- No diagram for this user story

Sprint 2
(09/10/2016 - 09/23/2016)

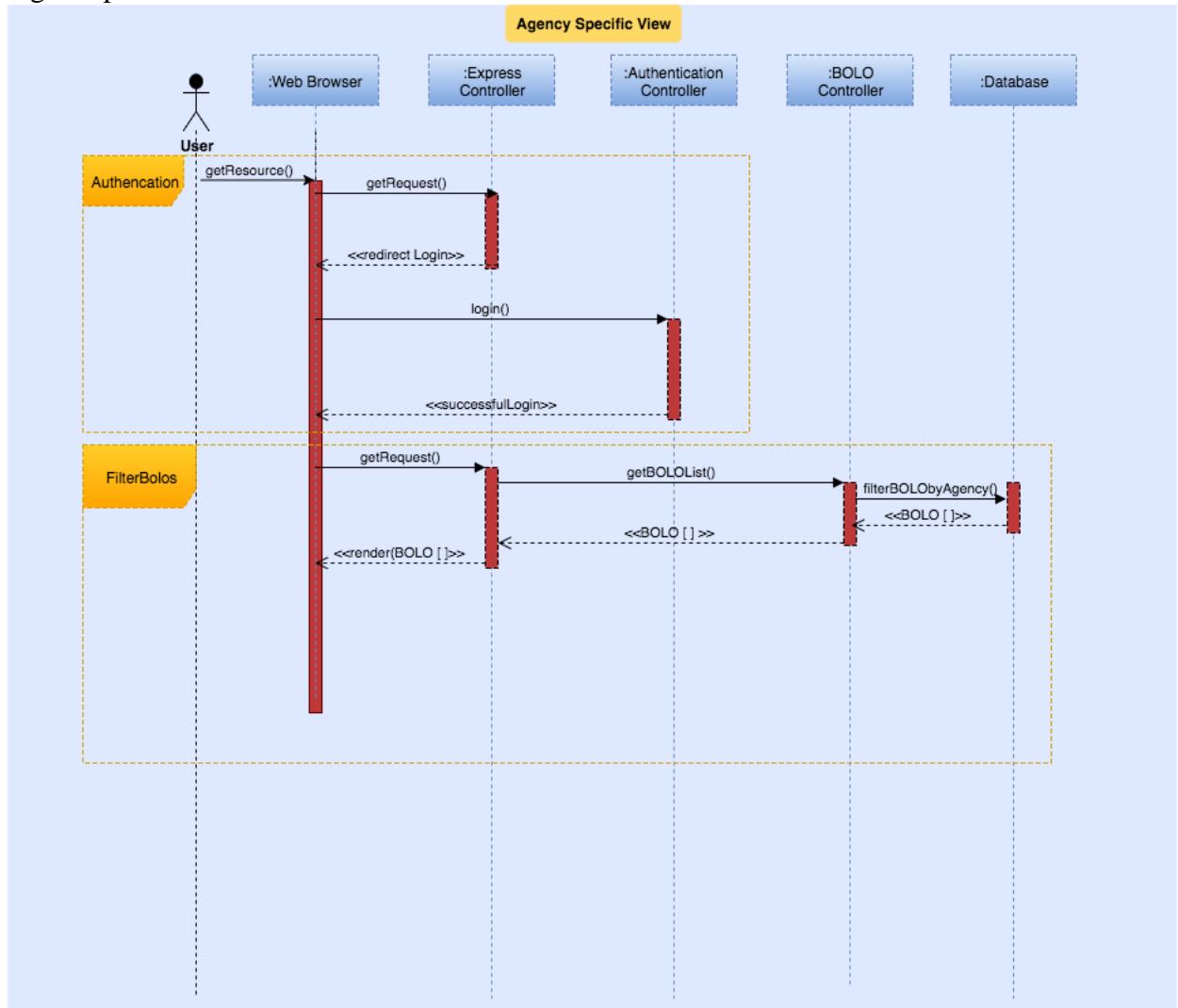
User Story #1189 Agency Specific BOLO

- Figure sp2-1



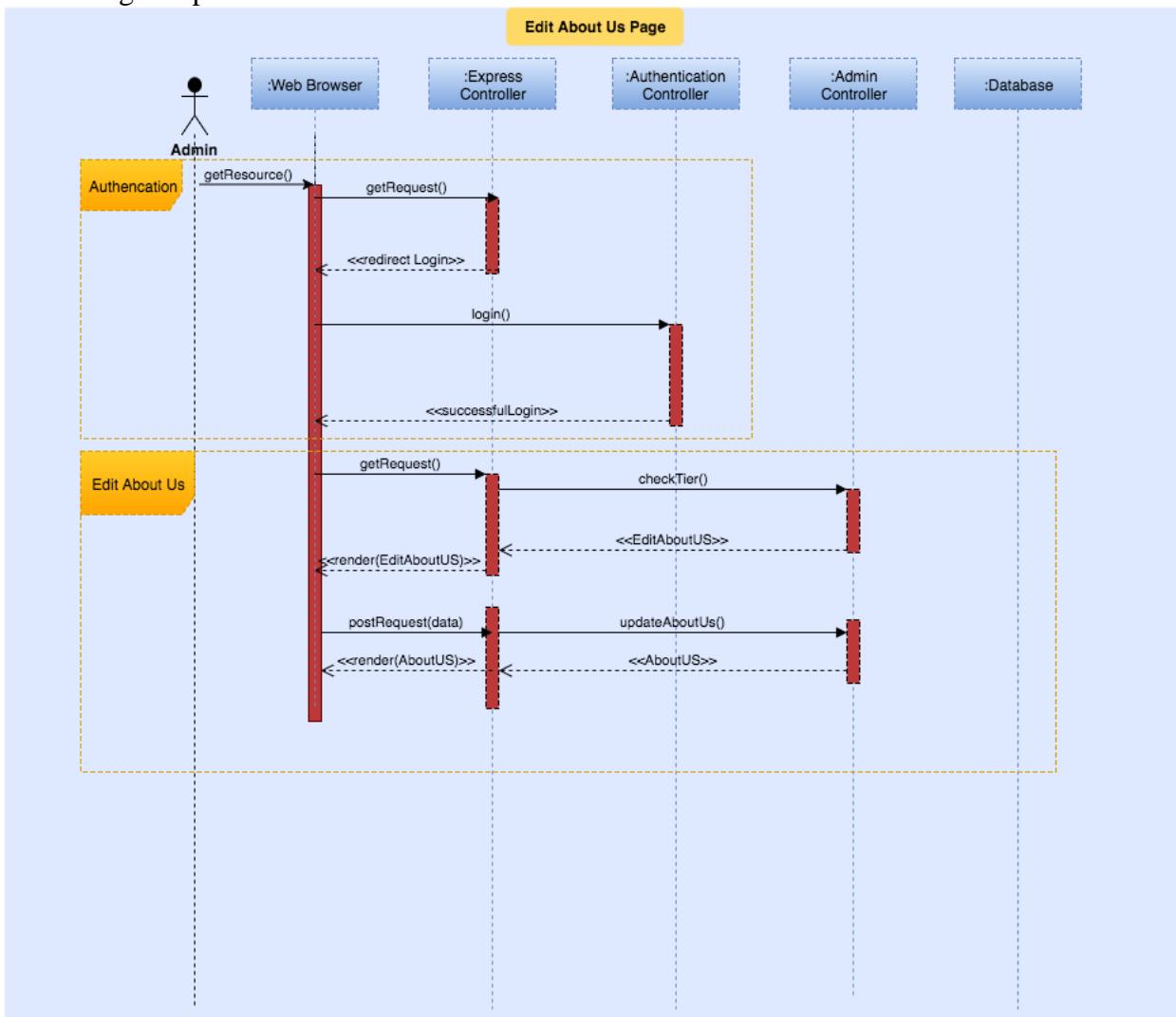
User Story #1190 Agency Specific View

- Figure sp2-2



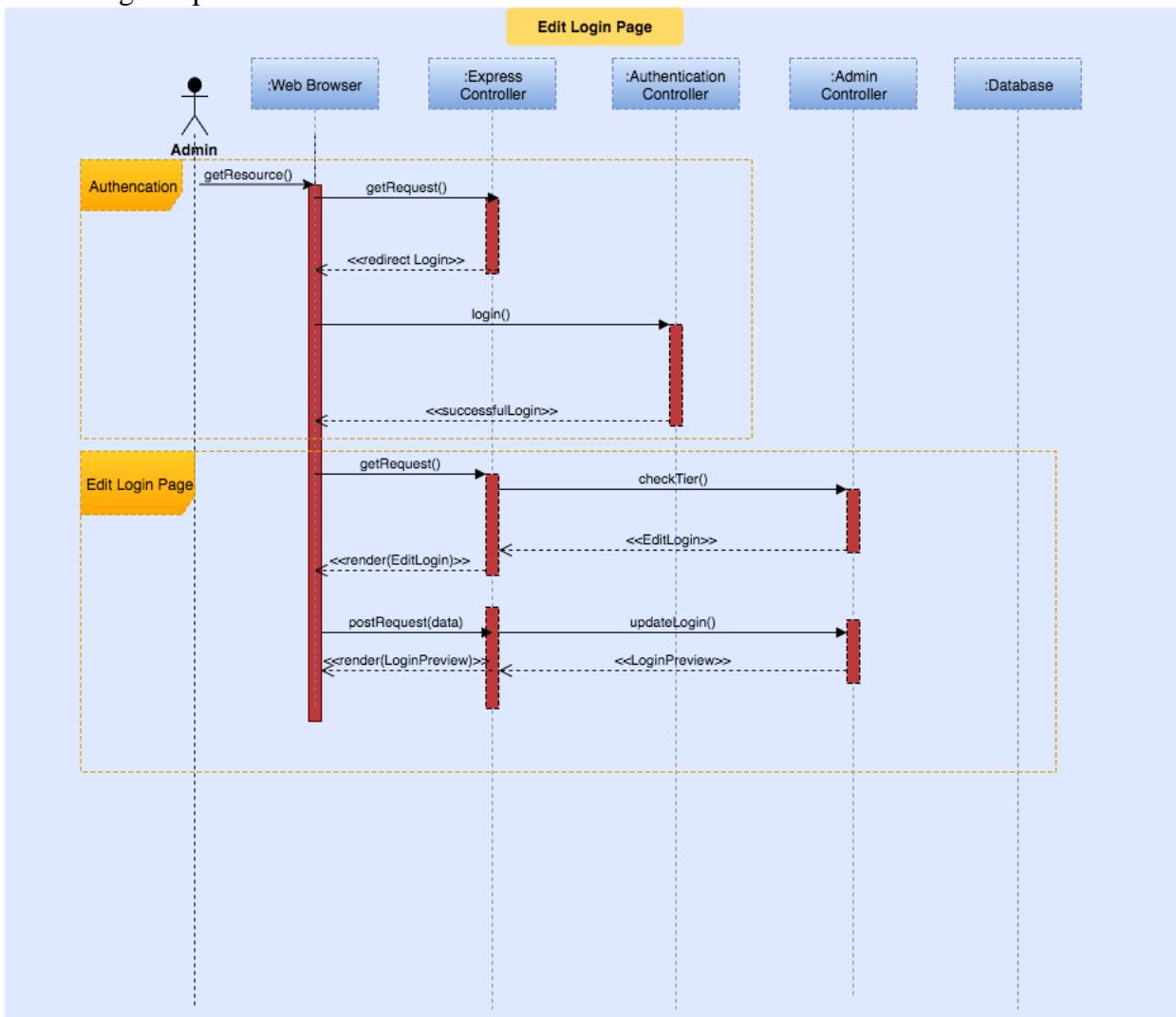
User Story 1182 Edit About Us Page

- Figure sp2-3



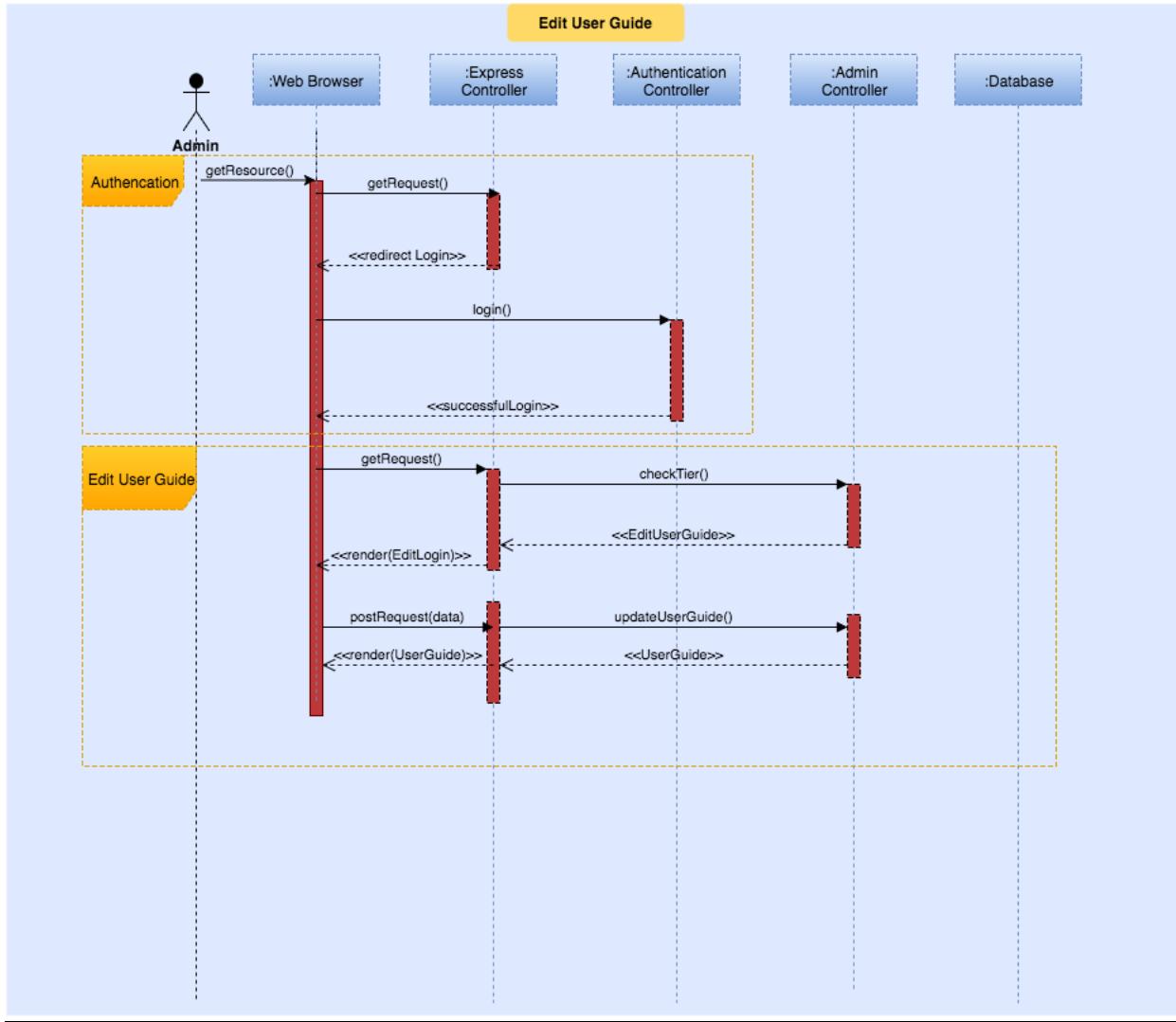
User Story 1183 Edit Login Page

- Figure sp2-4



User Story 1184 Edit User Guide Page

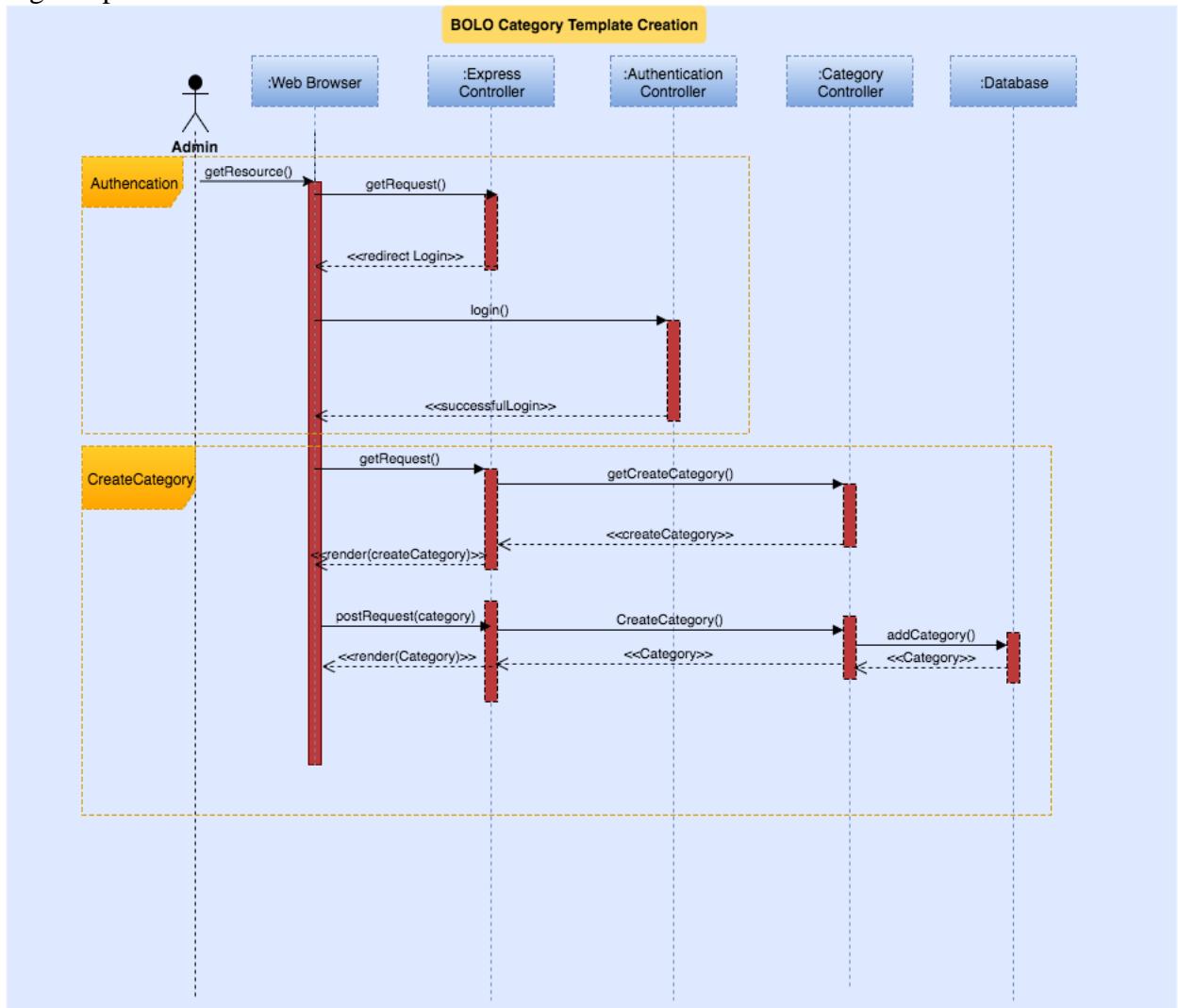
- Figure sp2-5



Sprint 3
(09/24/2016 – 10/07/2016)

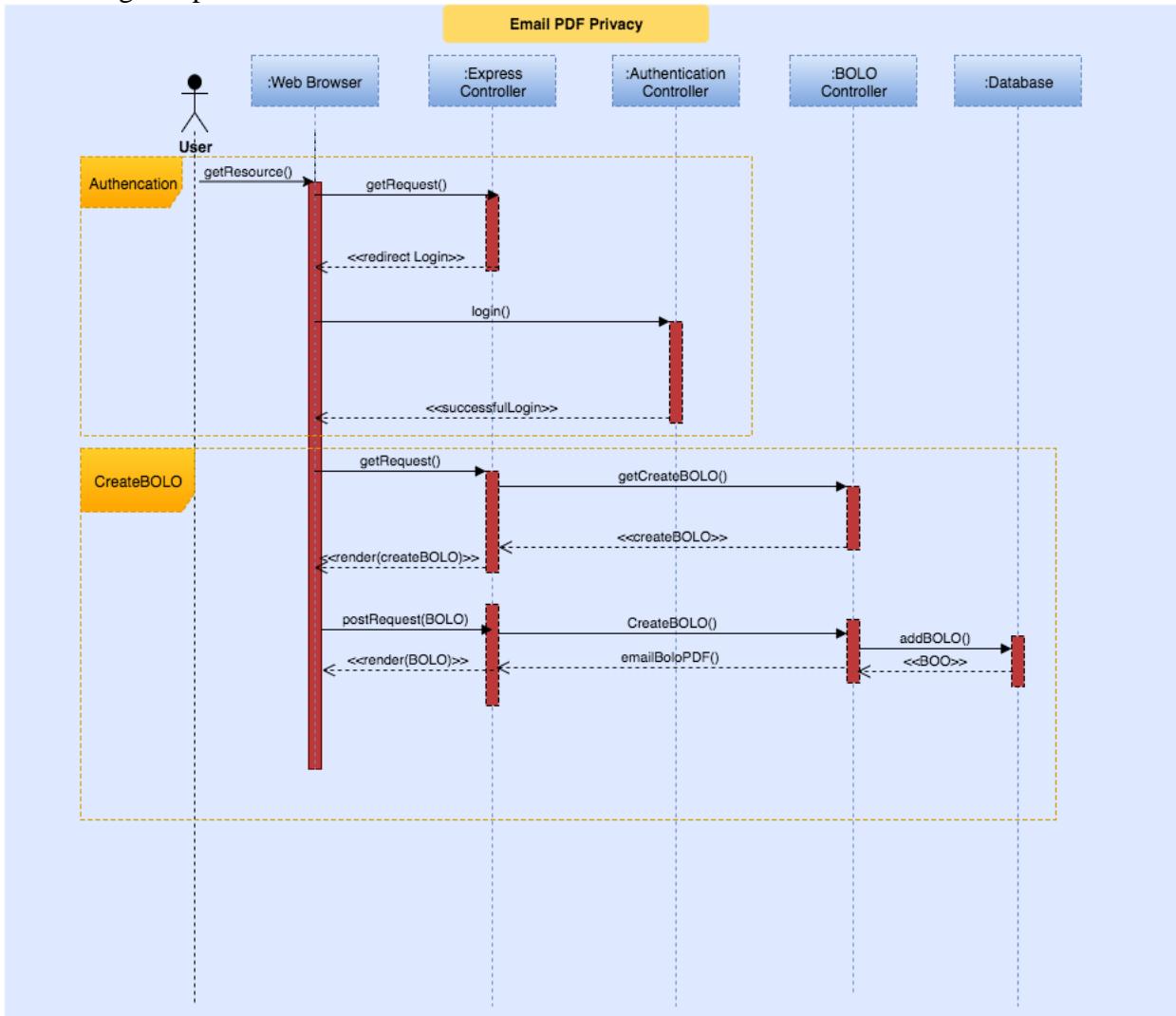
User Story #1194 BOLO Category Template Creation

- Figure sp3-1



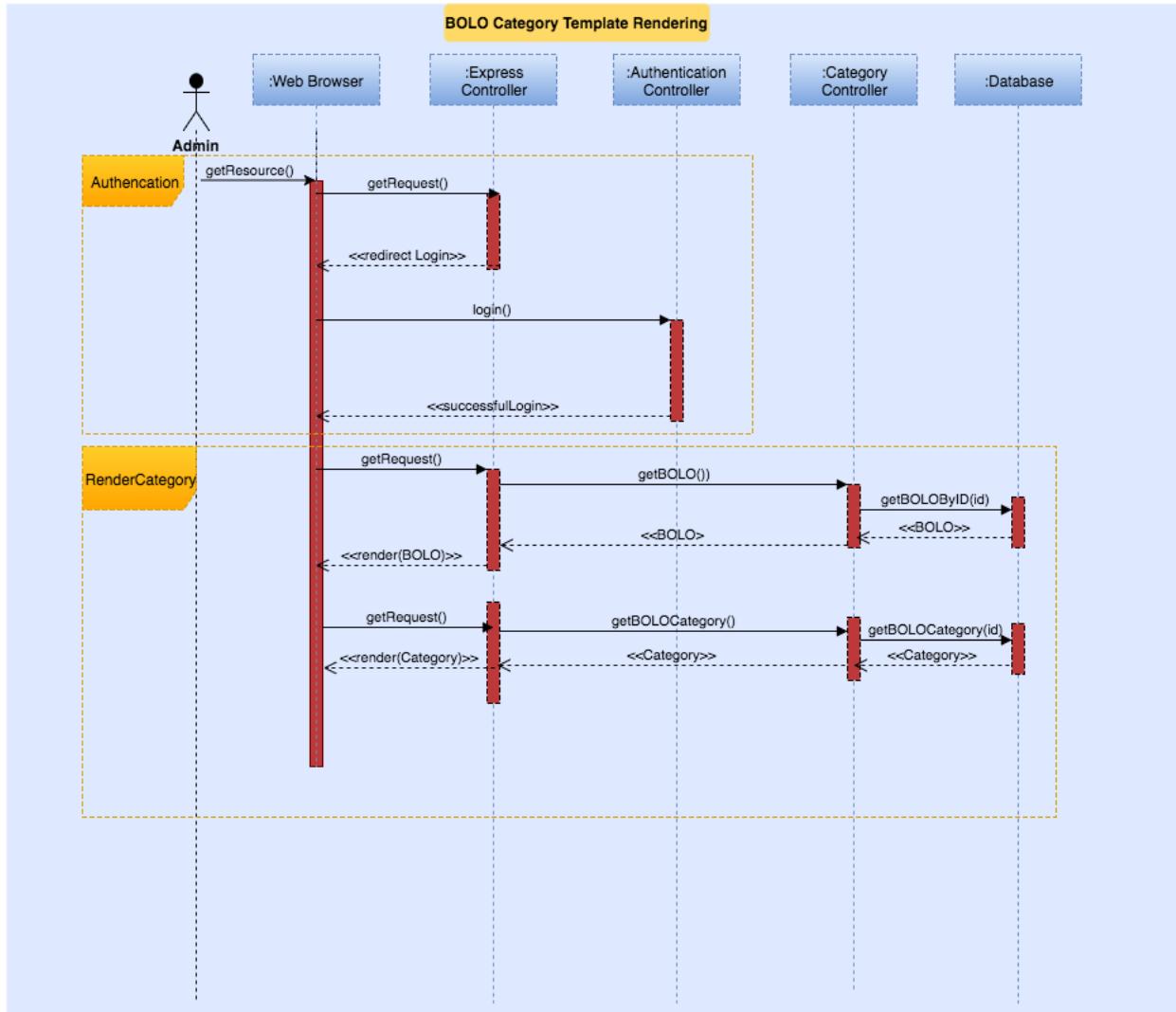
User Story #1191 Email PDF Privacy

- Figure sp3-2



User Story #1196 Template Rendering

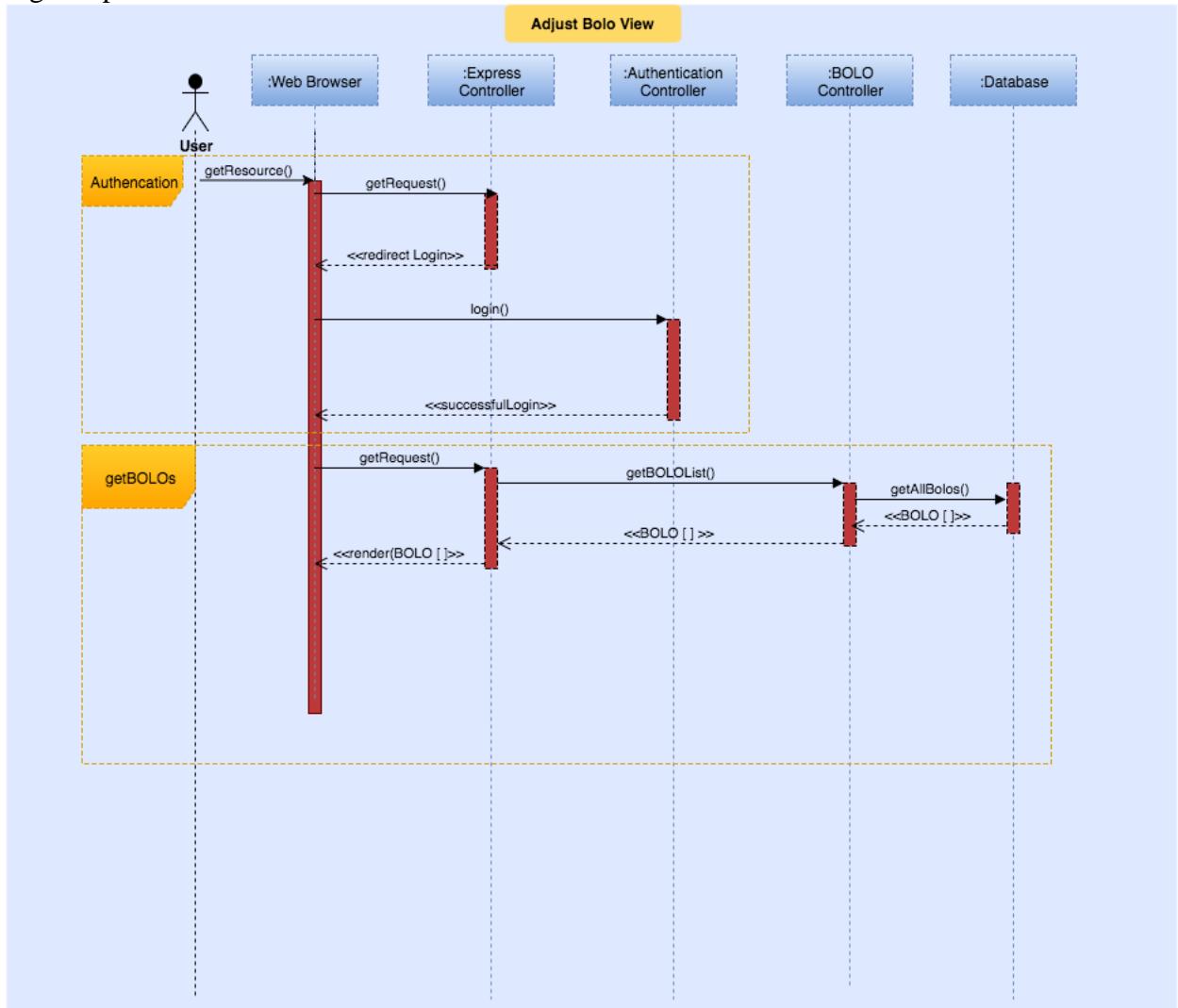
- Figure sp3-3



Sprint 4
(10/08/2016 – 10/21/2016)

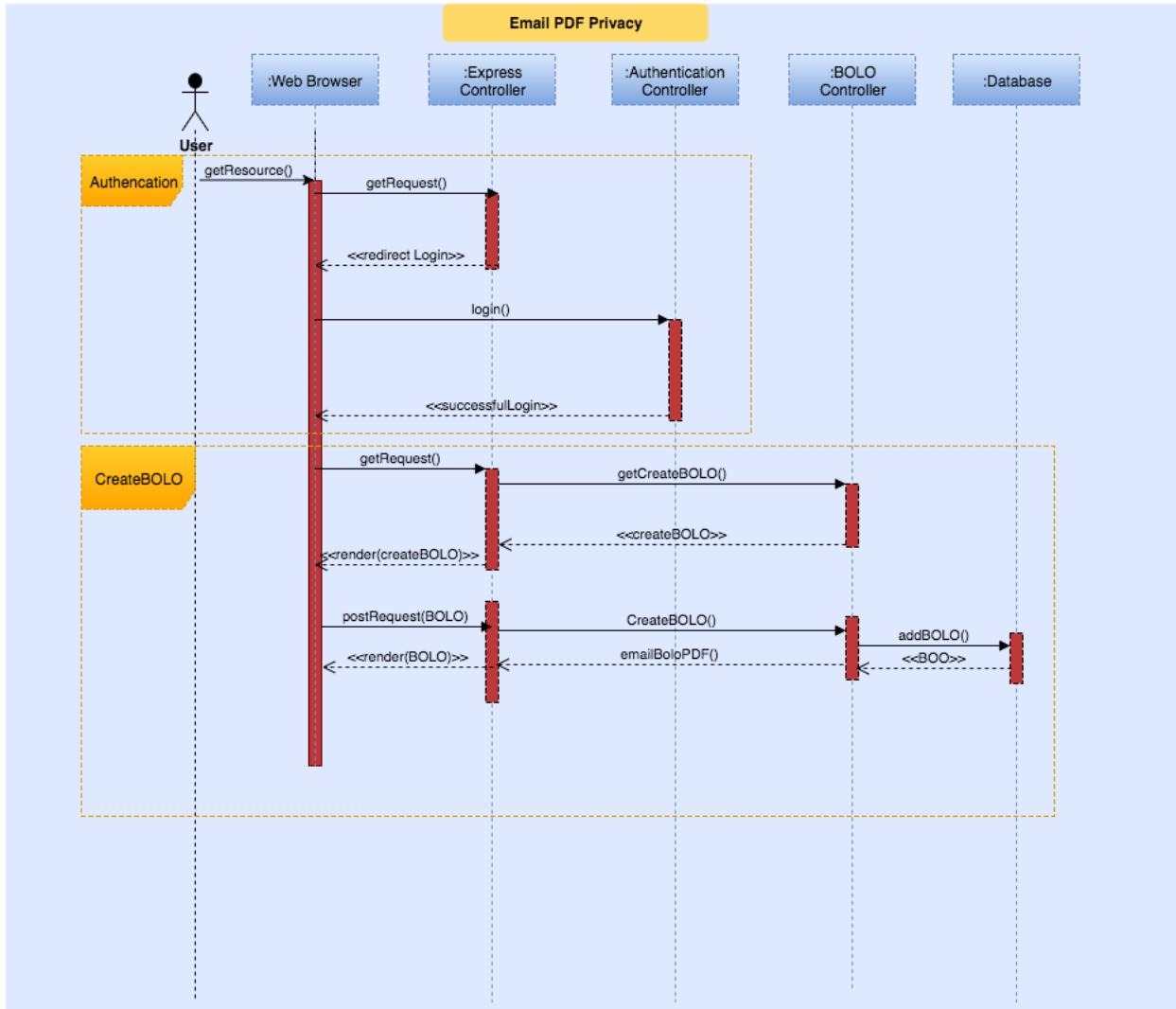
User Story #1198 Adjust BOLO View

- Figure sp4-1



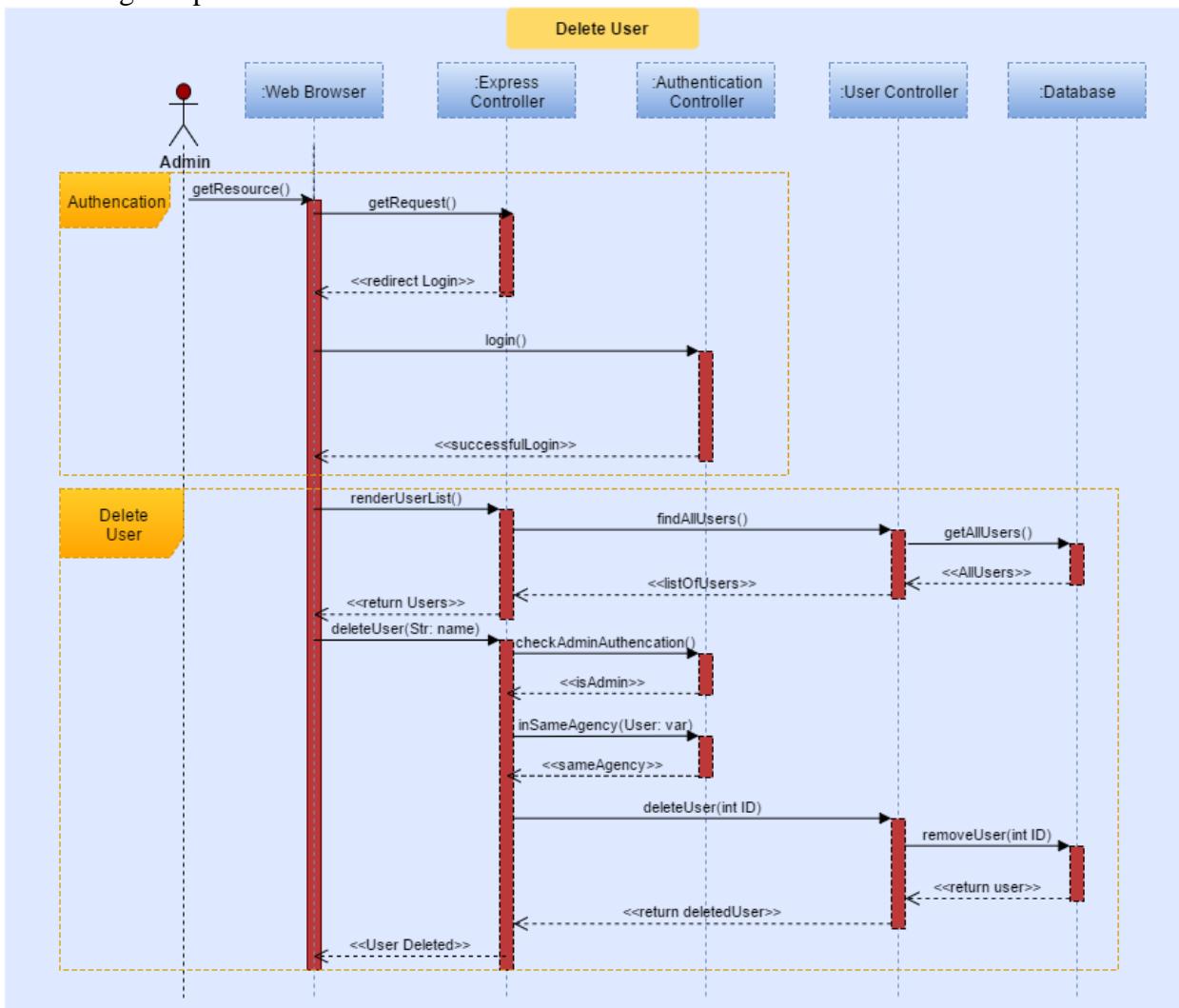
User Story #1200 Adjust Email PDF Privacy for MongoDB

- Figure sp4-2



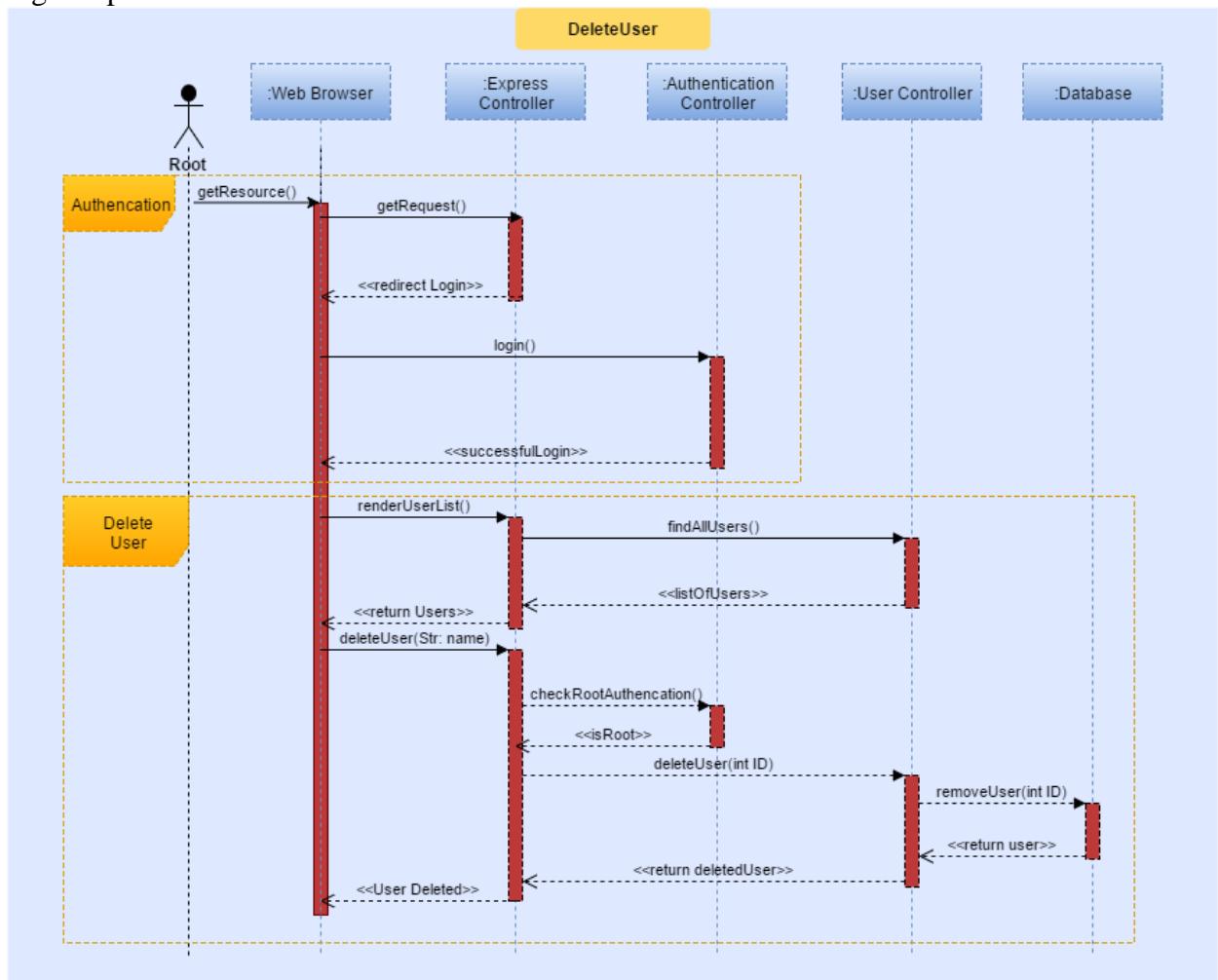
User Story #1192 Delete User (performed by admin user)

- Figure sp4-3



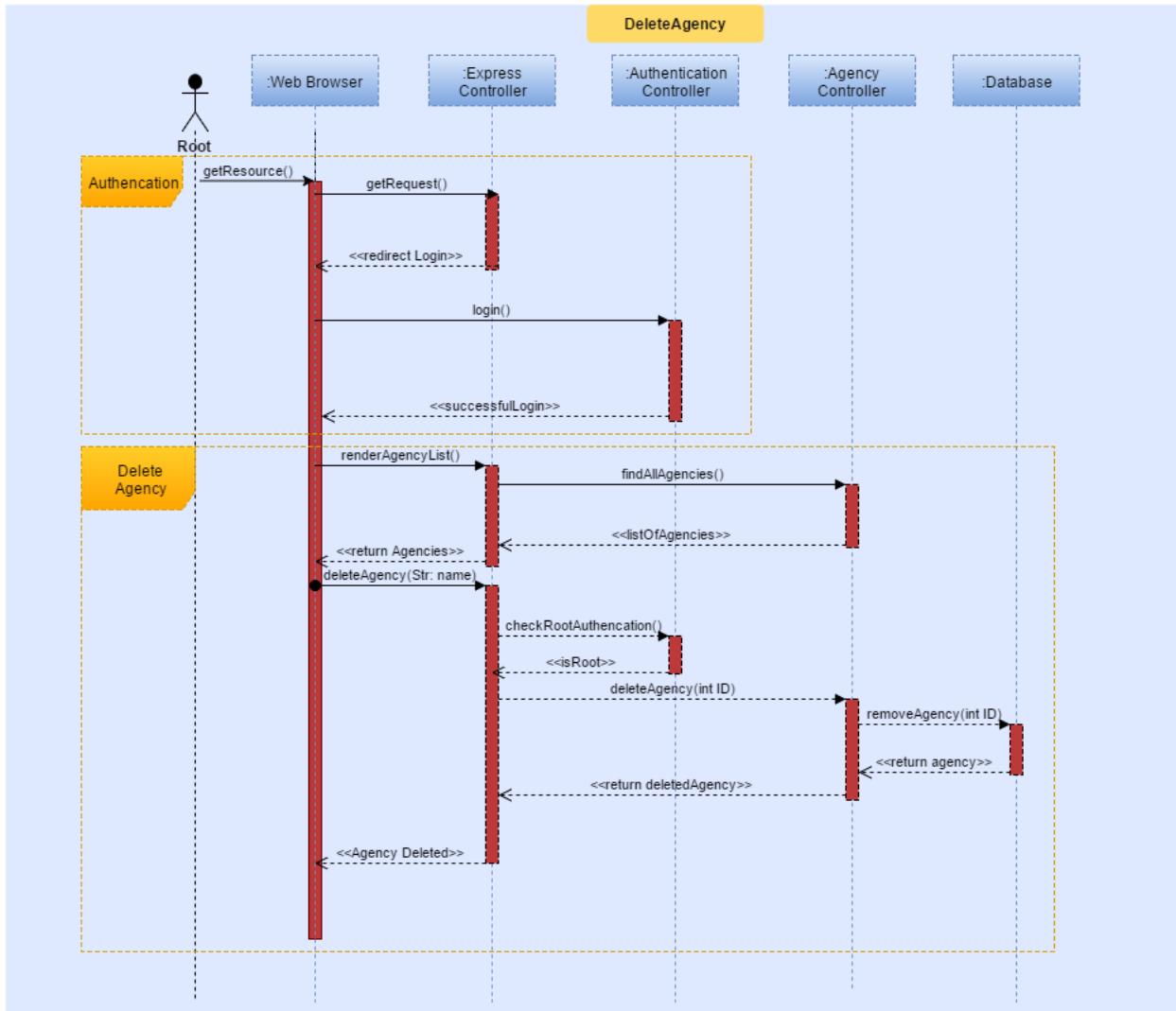
User Story #1192 Delete User (performed by root user)

- Figure sp4-4



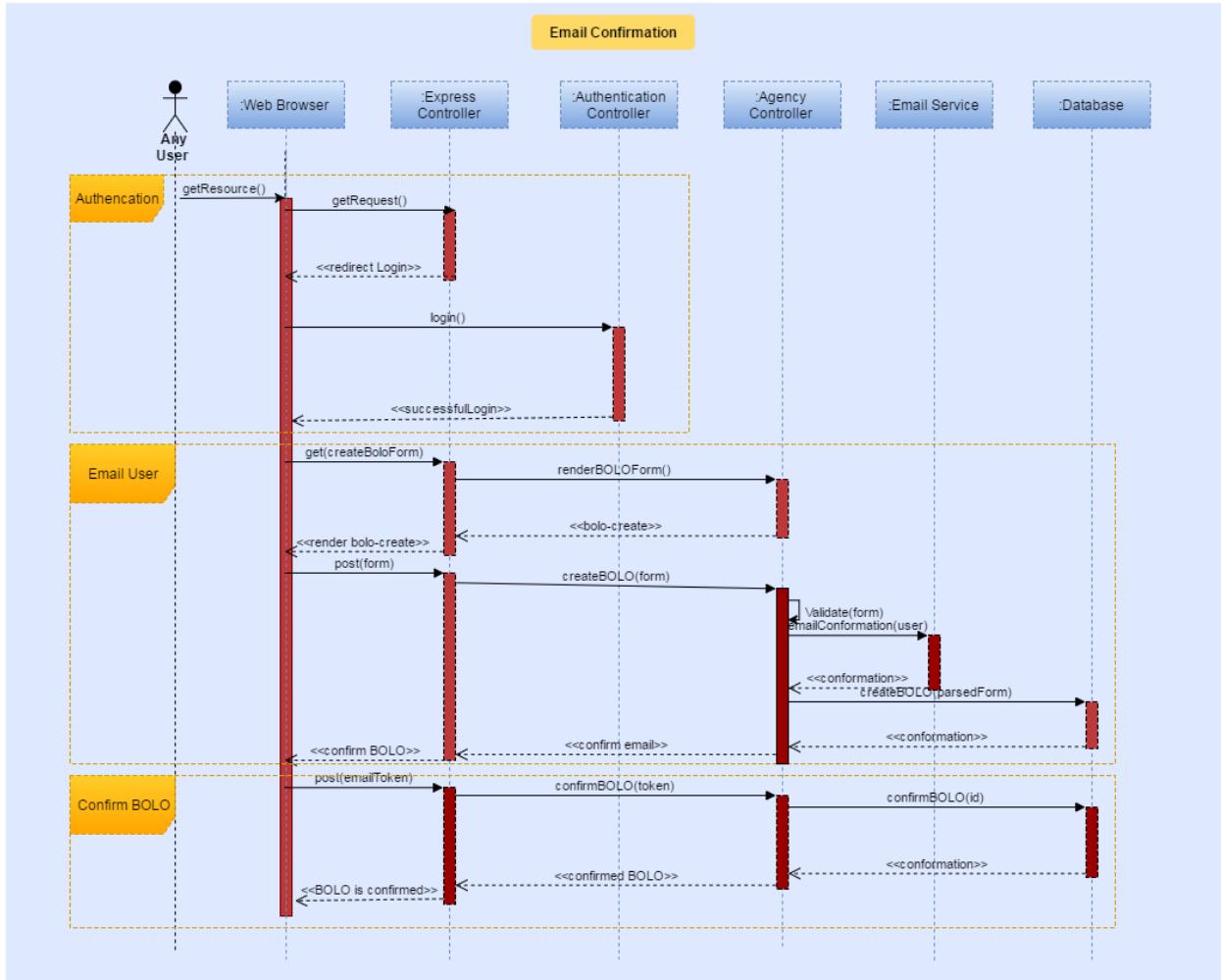
User Story #700 Delete Agency

- Figure sp4-5



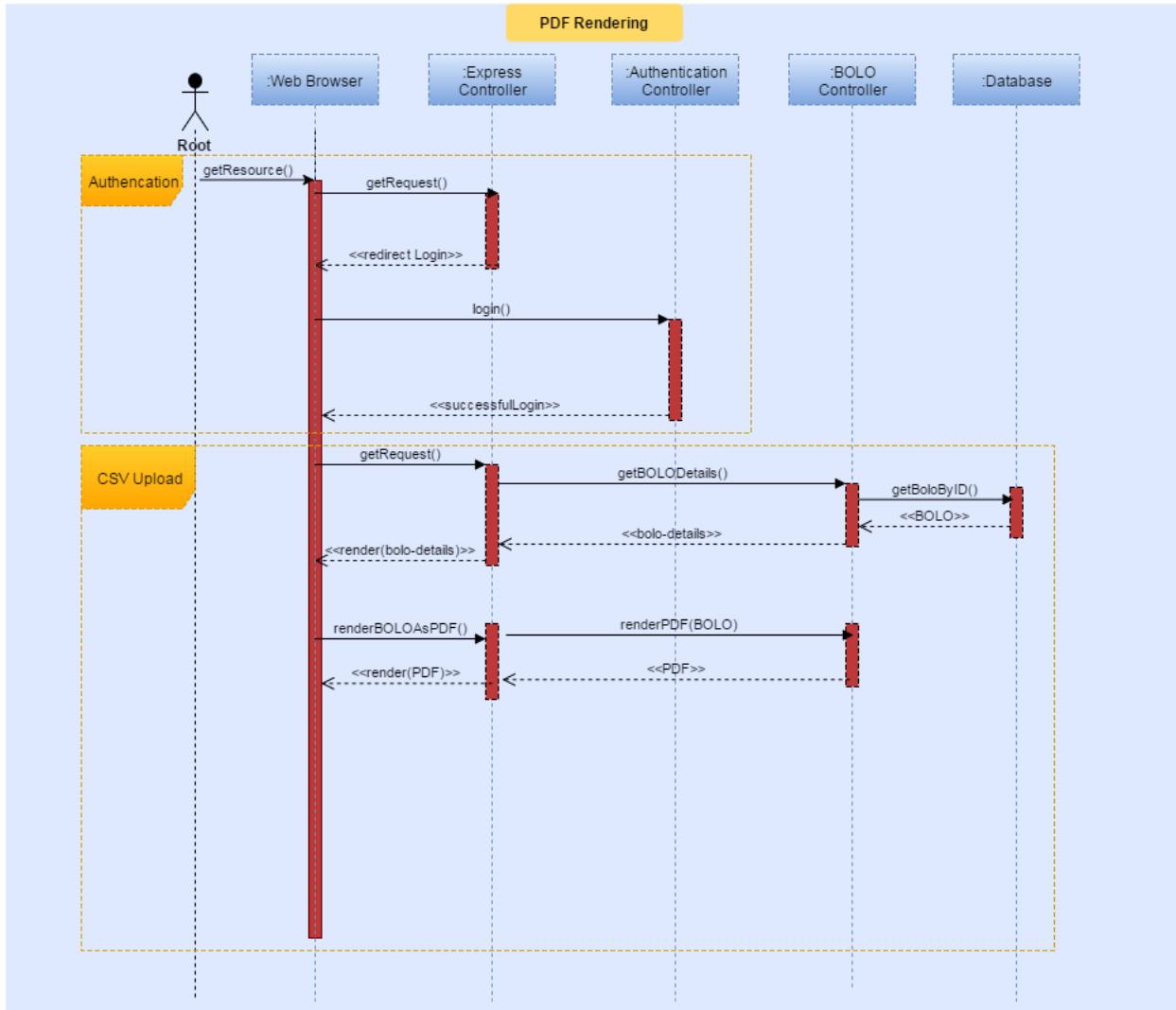
User Story #1197 Email Confirmation

- Figure sp4-6



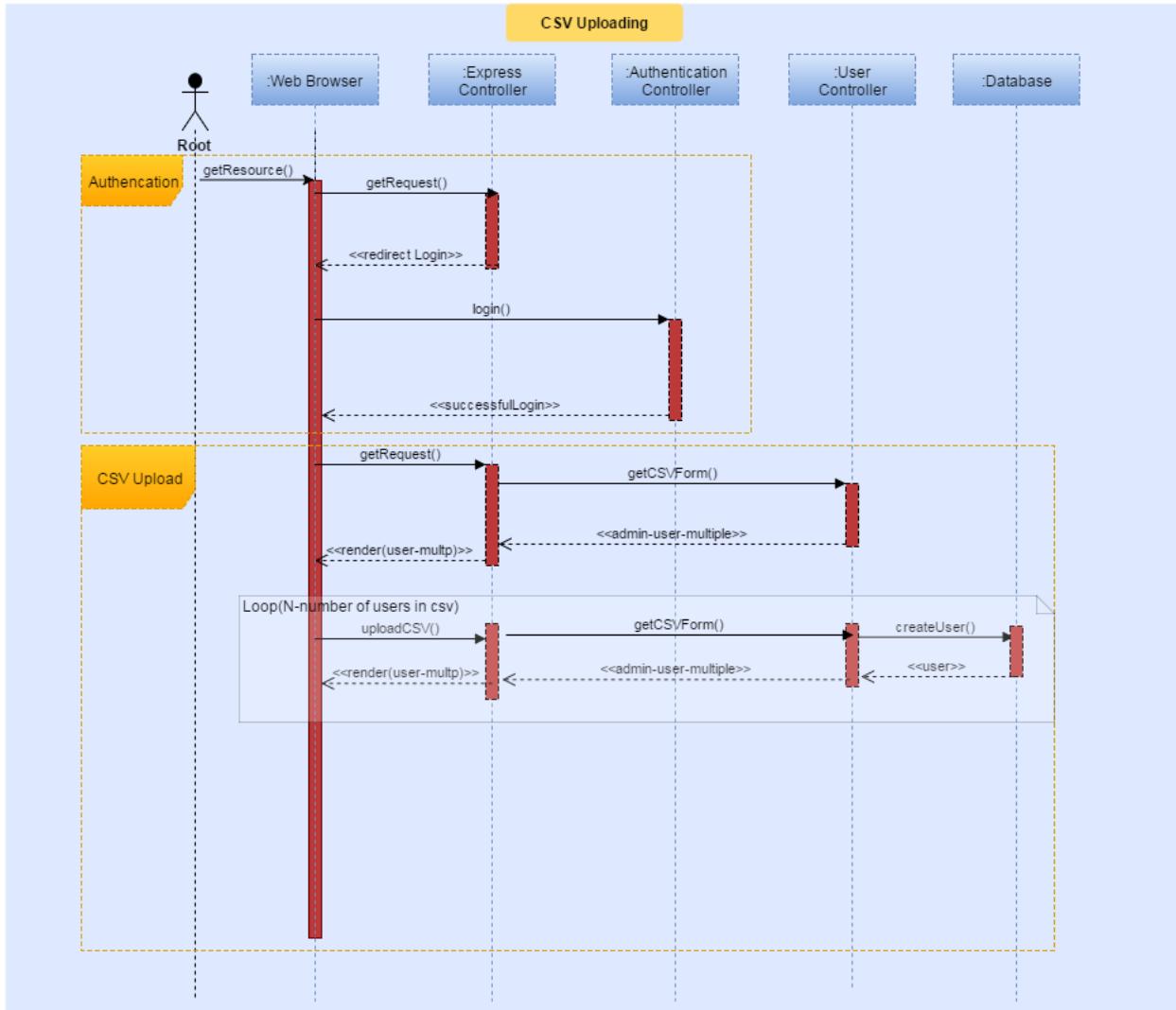
User Story #1199 Adjust PDF Rendering for MongoDB

- Figure sp4-7



User Story #1195 CSV Upload

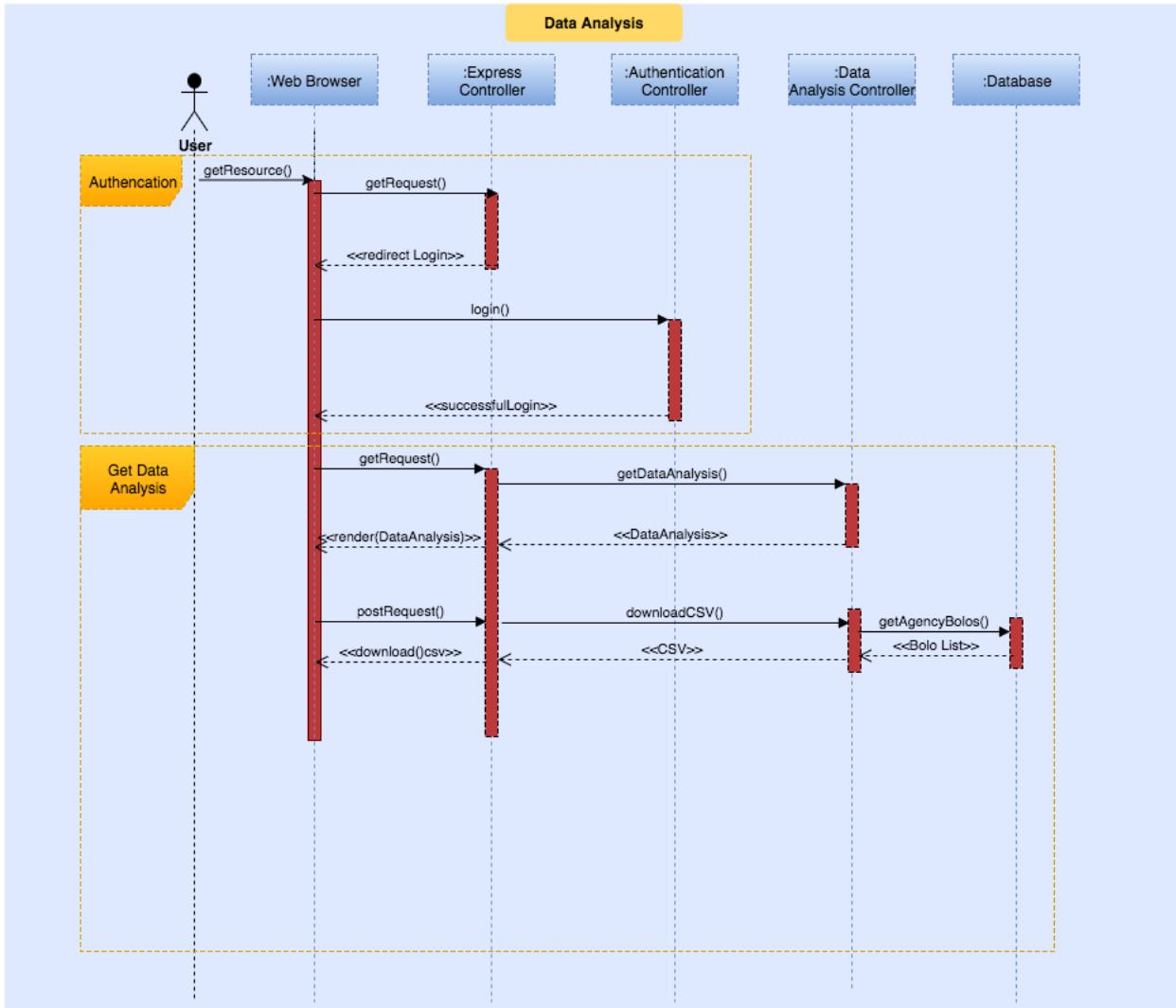
- Figure sp4-8



Sprint 5
(10/22/2016 – 11/04/2016)

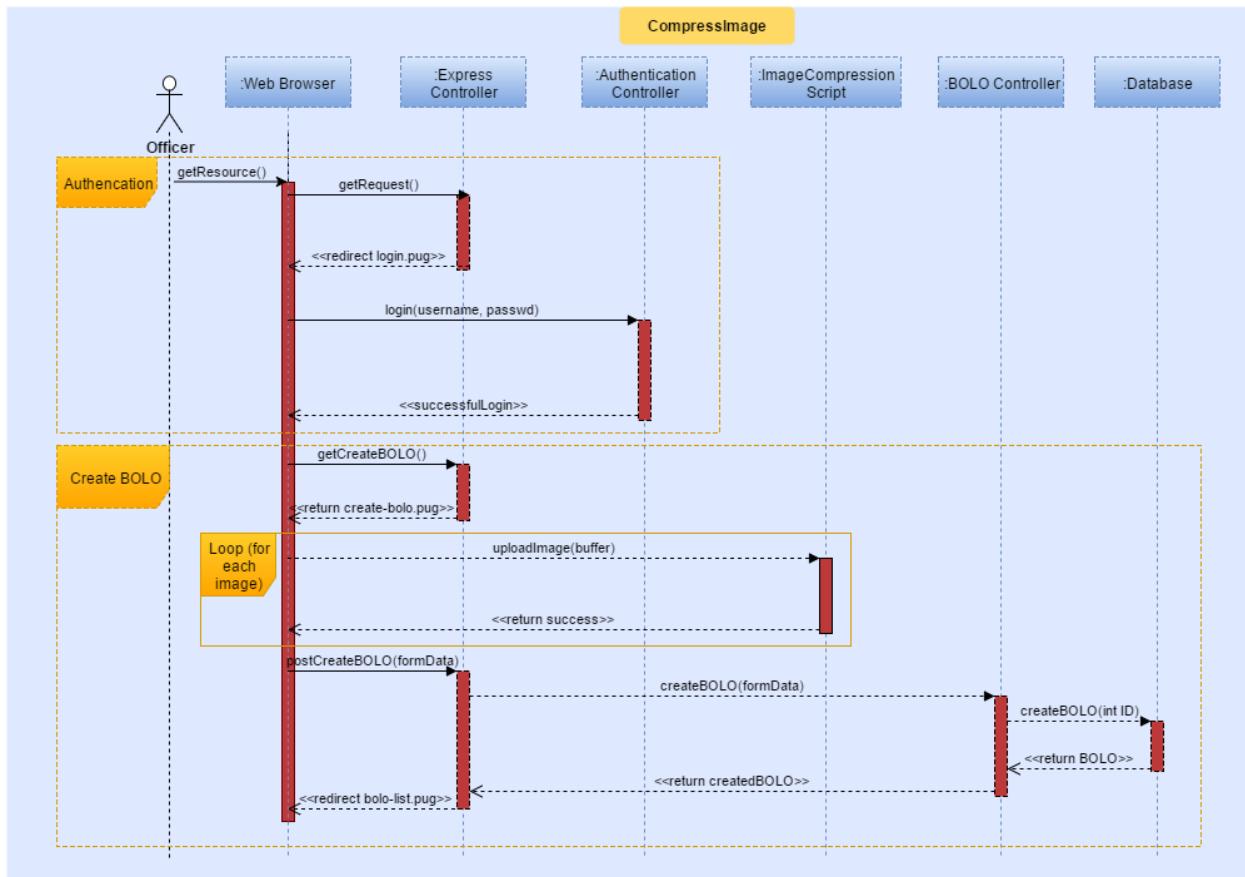
User Story #1204 Data Analysis

- Figure sp5-1



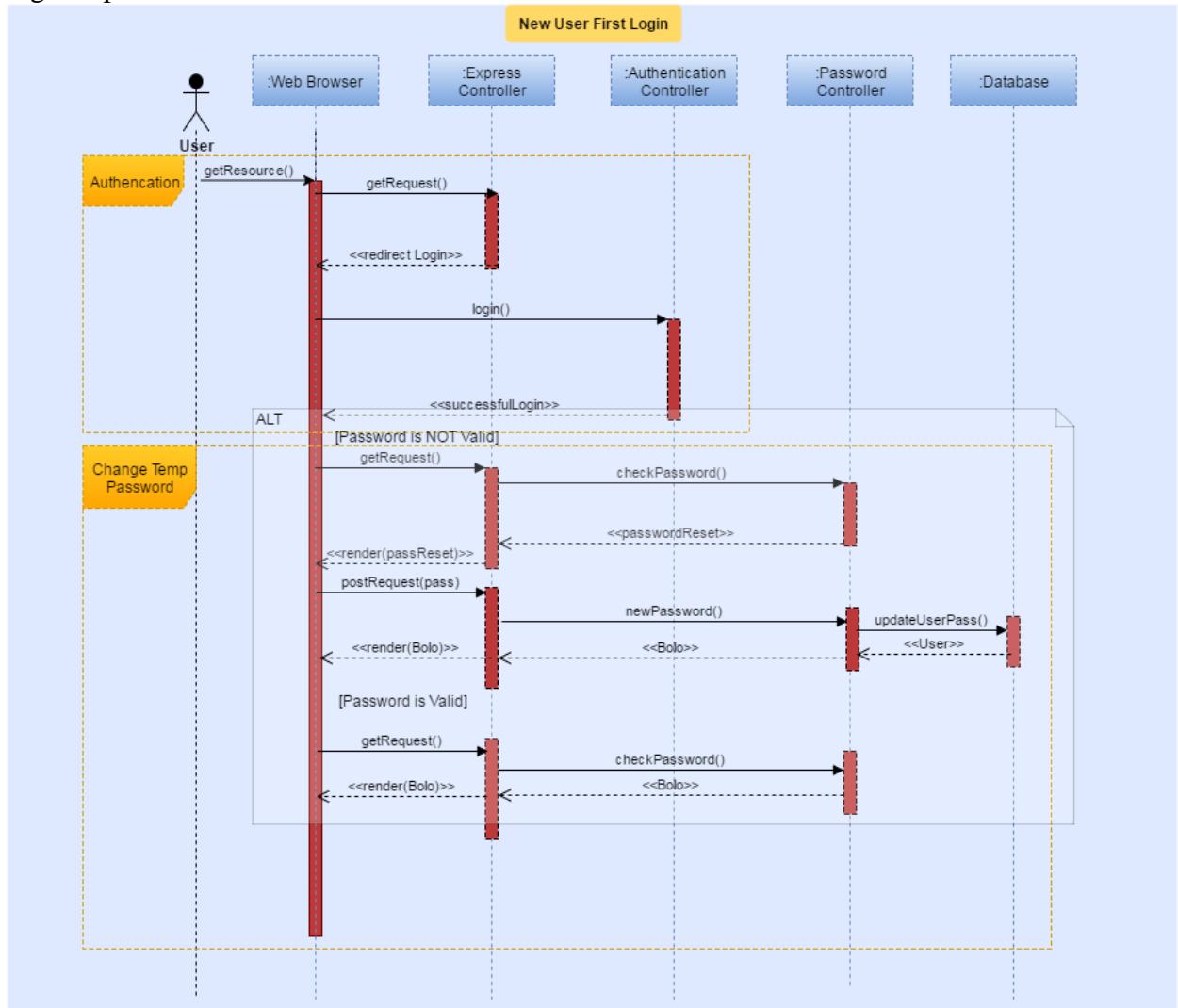
User Story #1193 Image Compression

- Figure sp5-2



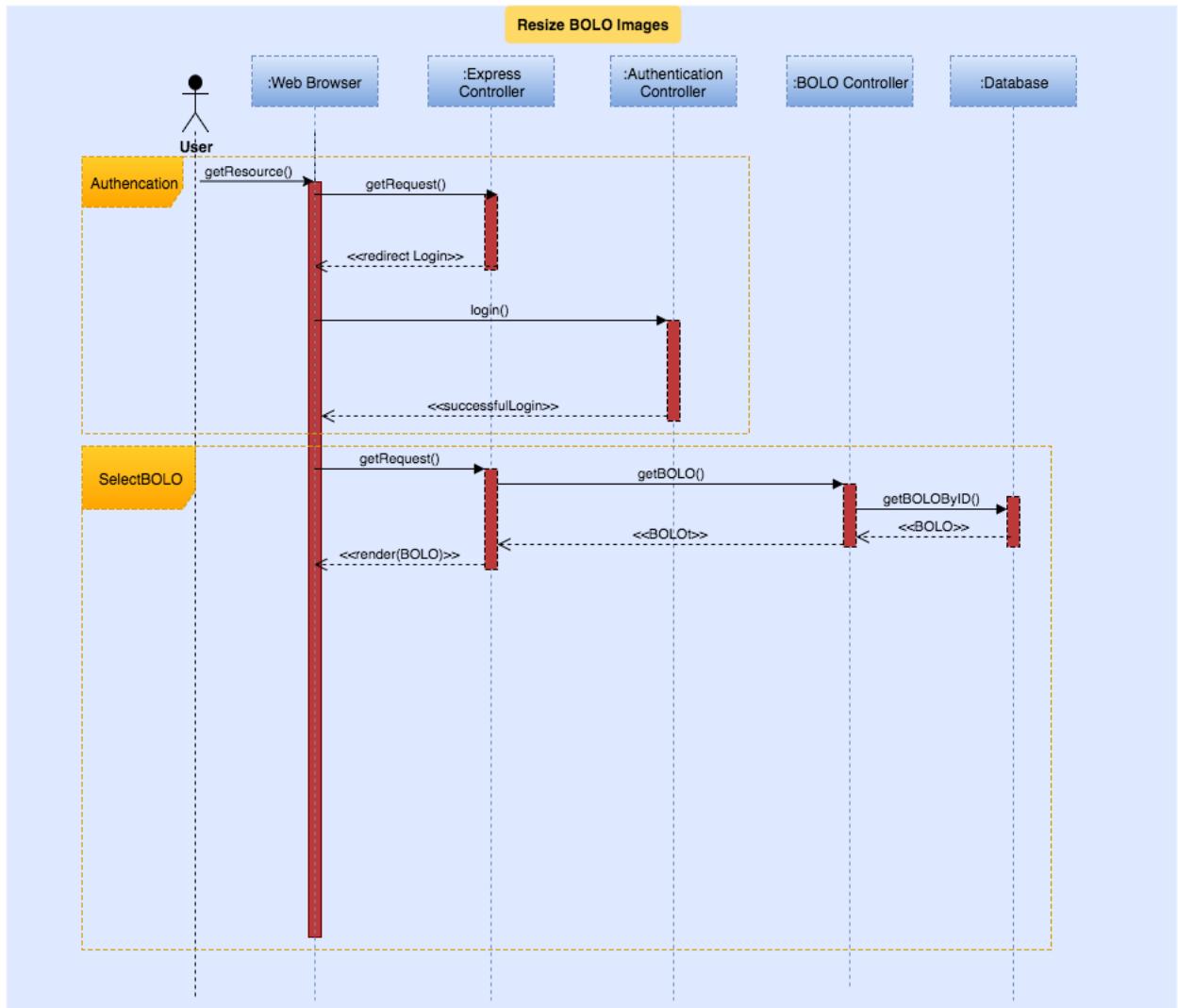
User Story #1180 Set Password

- Figure sp5-3



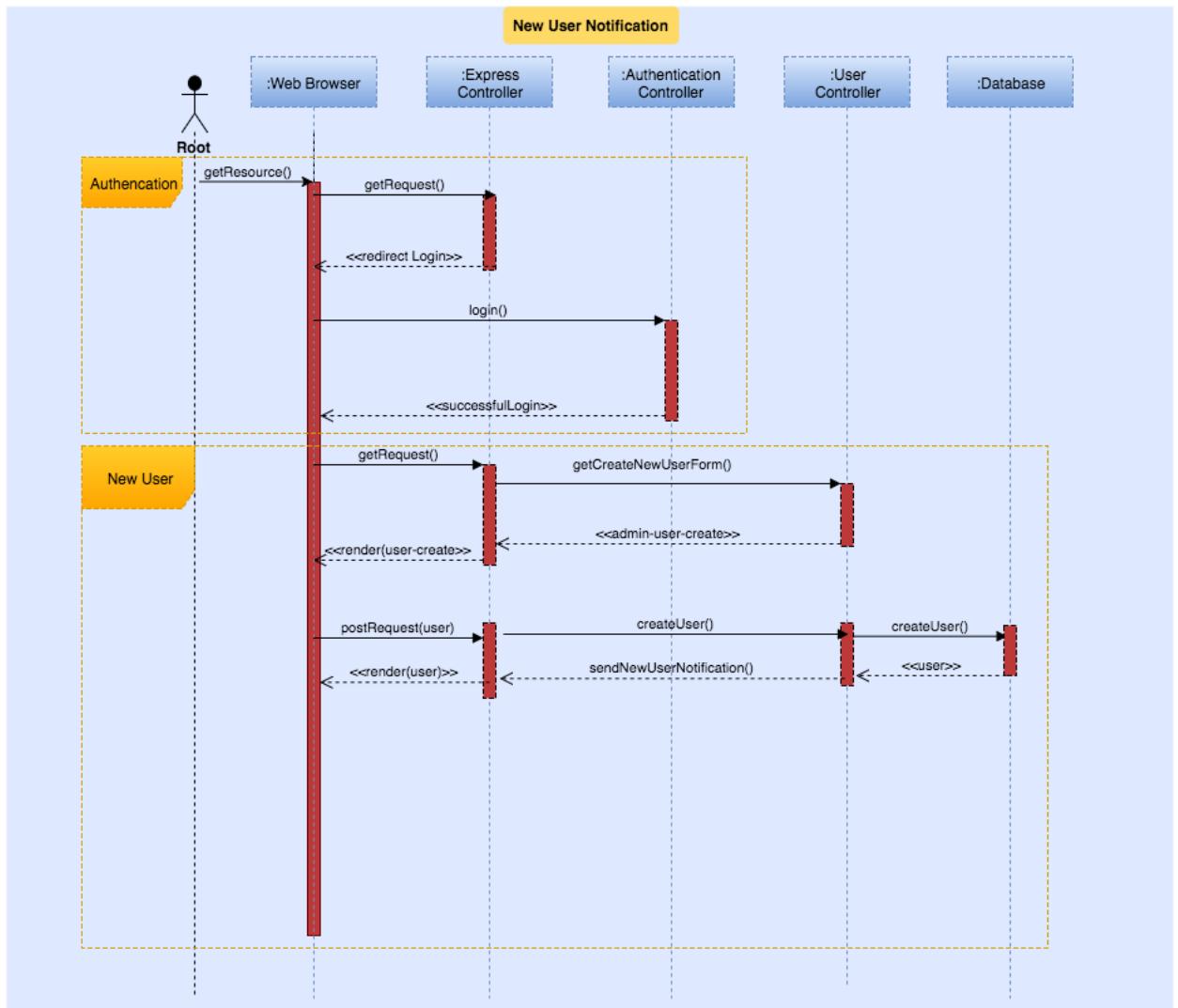
User Story #1202 BOLO Image Resizing

- Figure sp5-4



User Story #1202 New User Notification

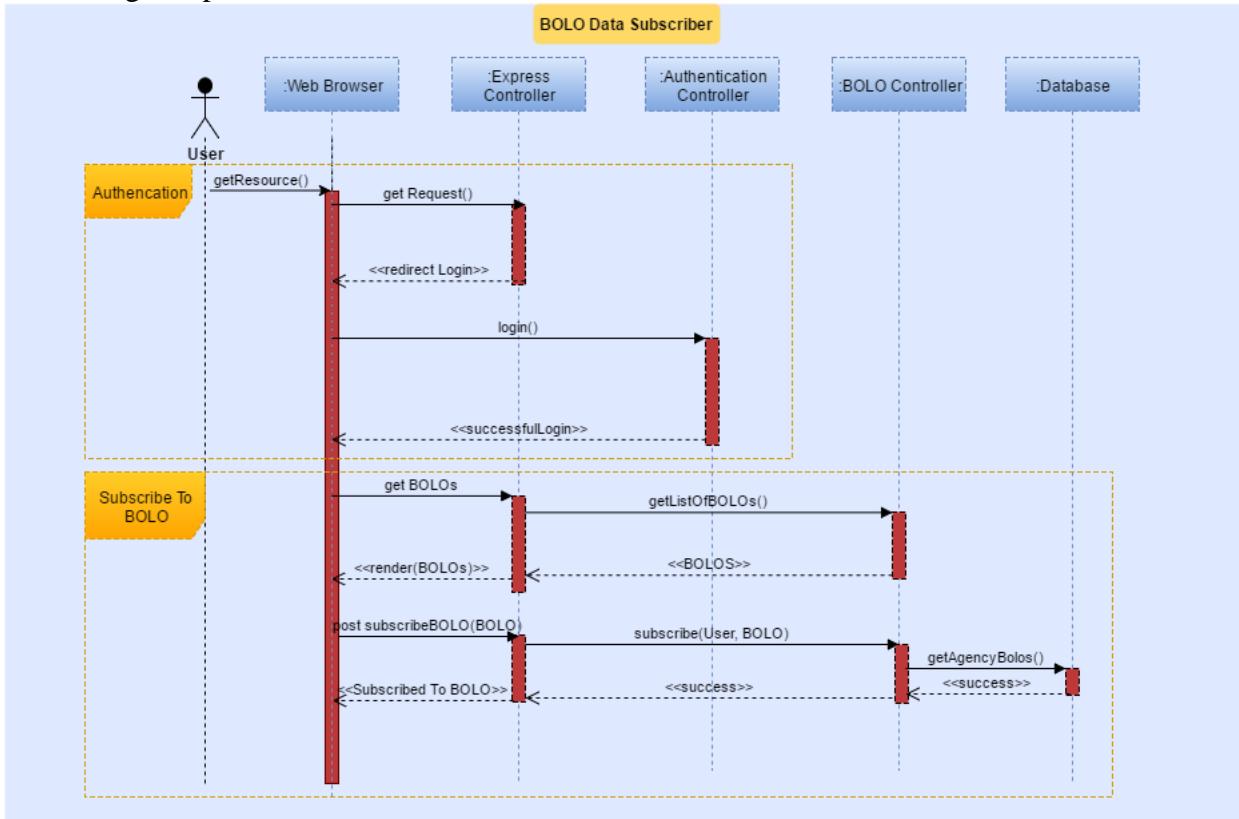
- Figure sp5-5



Sprint 6
(11/05/2016 – 11/11/2016)

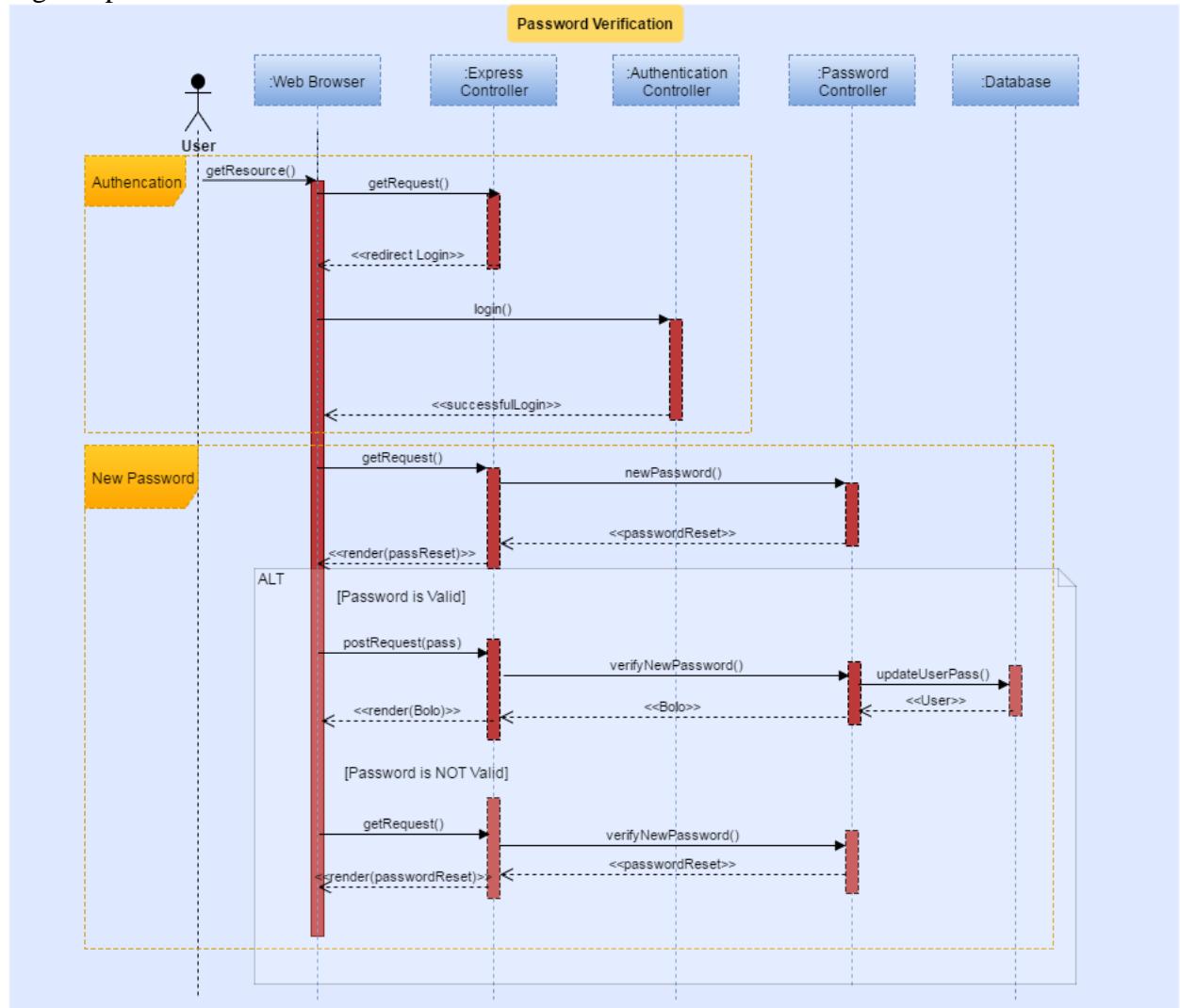
User Story #1201 BOLO Data Subscriber

- Figure sp6-1



User Story #1205 Password Verification

- Figure sp6-2



Appendix B – User Interface Design

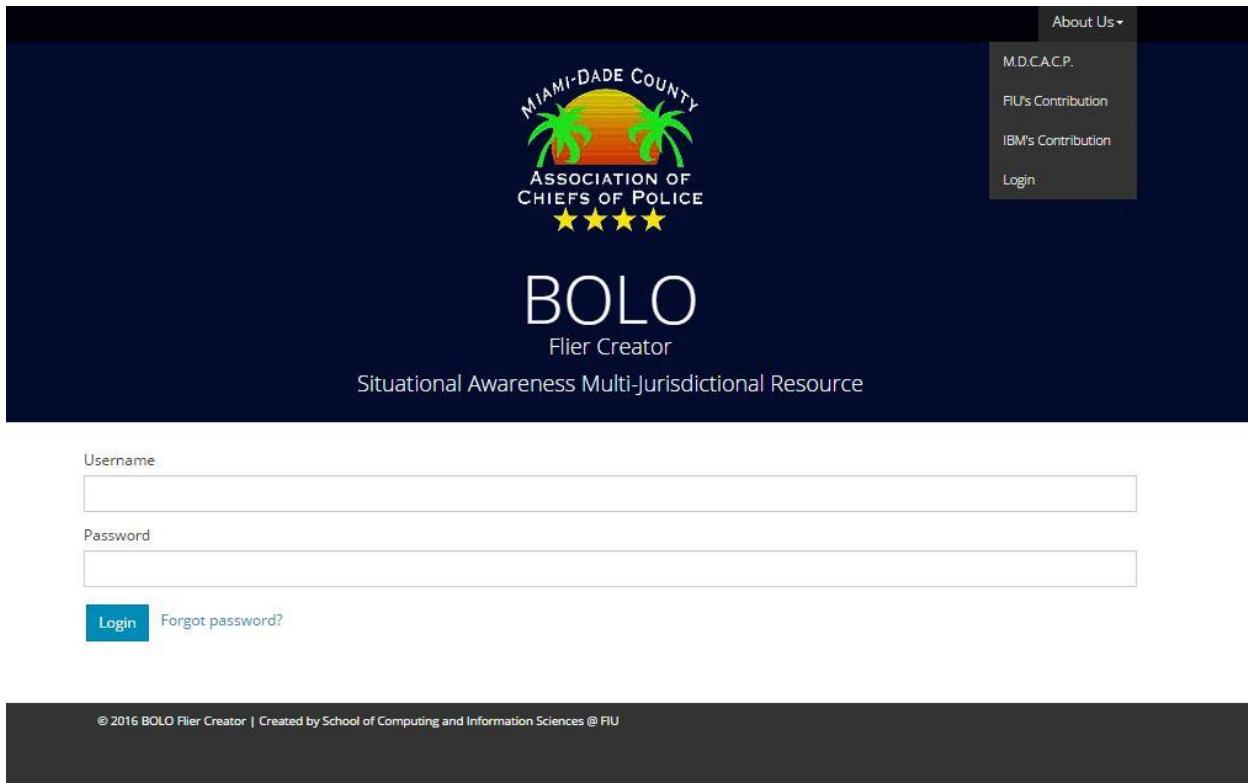


Figure UI01 – About us



The Miami-Dade County Association of Chiefs of Police is a premier member organization; the purpose of which is to promote efficient law enforcement services through the cooperation and coordination of all law enforcement agencies within Miami-Dade County.

The Miami-Dade County Association of Chiefs of Police is the second largest Police Chiefs Association in the State of Florida. It is comprised of more than 250 of the County's top law enforcement executives and another 300 associate members. The Association was originally organized in 1937 to promote the administration and professionalism of law enforcement that would enhance public security by providing superior police protection for the residents of Miami-Dade County and its many visitors. Today, with the same ultimate goal, its role has expanded to provide better communication, education, and training for the county's various police agencies. MDCACP facilitates the dissemination of information, provides continuing police training, and promotes a better understanding of the police profession in general. Membership in the Miami-Dade County Association of Chiefs of Police supports and maintains the highest standards of public safety and police protection for the millions of people who live, work and play in Miami-Dade County.

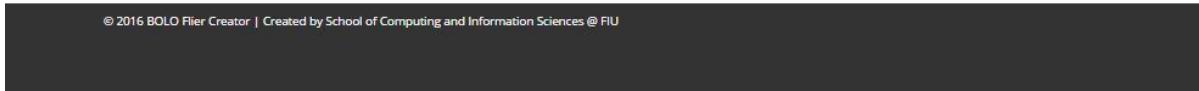


Figure UI02 – About us Miami Dade Association of Chiefs of Police



For the seniors in FIU's Department of Electrical and Computer Engineering, everything they've learned in four years culminates with a single project. Working together in teams, they take on projects that test their knowledge, ingenuity, patience and imagination. Finally, on Senior Design Day, the students put on their best suits and showcase their projects to a host of industry advisers who pick their brains and give them feedback on their work. The Senior Project course was created to test a potential graduate's ability to work in a real software development environment. Potential graduates' success in this course will enable them to provide a significant software project that may help them land a job in the real-world.

The Professor



Masoud Sadjadi, PhD

Dr. Sadjadi holds a B.S. degree in Hardware Engineering, an M.S. degree in Software Engineering, and a Ph.D. degree in Computer Science. He is currently an Associate Professor in the School of Computing and Information Sciences at Florida International University, where he has been on the faculty since 2004. He serves as the Director of the VIP project at FIU, the Director of the Professional Master in Science for Information Technology at FIU, and is in charge of Computer Science Senior Projects. He has extensive experience in software development and leading large-scale software engineering projects both in industry and in academia. During the past three years, Dr. Sadjadi has successfully transformed senior projects at FIU from a traditional waterfall approach to an agile/scrum approach. Dr. Sadjadi's research is focused on Agile Software Development in Distributed Systems, Autonomic Computing, High-Performance Computing, Grid Computing, Cloud Computing, Pervasive Systems, and Mobile Computing. He is the Director of the Autonomic Cloud Research Laboratory (ACRL) at FIU and leads several projects under the Latin American Grid initiative. Currently, he is collaborating with top researchers in 8 countries and is leading several international collaborative research projects. He has served as the General Chair, Program Chair, Co-Chair, and Committee Member of several top-tier international conferences and workshops in his field. He has served as a referee for several IEEE and SP&E journals and as a referee and panelist.

Figure UI03 – About us FIU



IBM played a major role in bringing BOLO Flyer Creator to life with their software engineers that mentored the students and their product Bluemix. Bluemix is a cloud platform as a service (PaaS) developed by IBM. It supports several programming languages and services as well as integrated DevOps to build, run, deploy and manage applications on the cloud. Bluemix is based on Cloud Foundry open technology and runs on SoftLayer infrastructure. Bluemix supports several programming languages including Java, Node.js, Go, PHP, Python, Ruby Sinatra, Ruby on Rails and can be extended to support other languages such as Scala through the use of buildpacks.

**Juan Caraballo**

Student Mentor

IBM Software Engineer

**Robert Loredo**

Student Mentor

Lead Developer, IBM Support,

Powered by Watson

Figure UI04 – About us IBM



Username

Password

[Forgot password?](#)

© 2016 BOLO Flier Creator | Created by School of Computing and Information Sciences @ FIU

Figure UI05 – Login

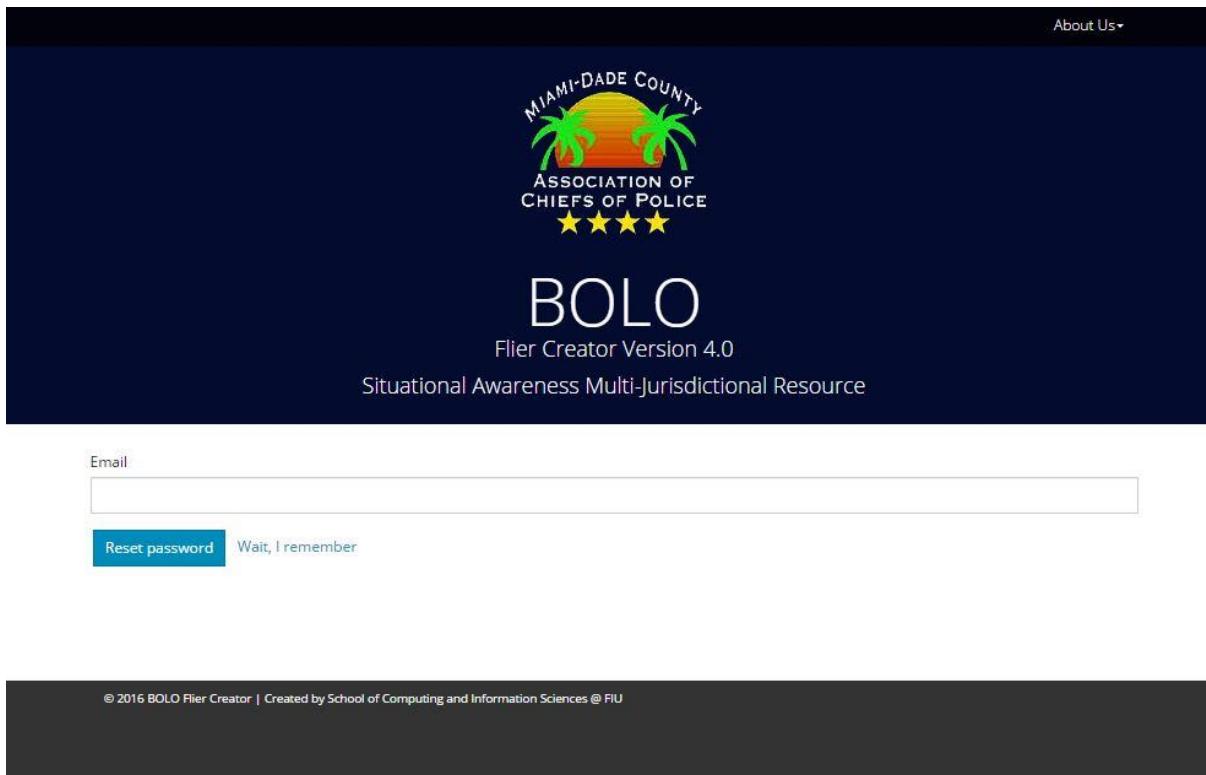


Figure UI06 – Forgot Password

Figure UI07 – Filter Bolos

The screenshot shows the 'Create a BOLO' form. At the top, there is a navigation bar with links for Home, Create, Archive, Agency Directory, Search, Admin, and a user dropdown labeled 'brian~'. Below the navigation bar, the title 'Create a BOLO' is displayed. The form fields include:

- Date Reported: DD/MM/YYYY input field.
- Time Reported: HH:MM input field.
- Category: A dropdown menu showing 'Select a Category' and several options: Testing, HIYA, Murder (which is selected and highlighted in blue), BOAT, and CHOOSE FILE.
- Image Upload: A file input field showing 'no file selected' and a 'Choose File' button.
- Video Link: An input field for a video link.
- Additional Info: A large input field for additional information.
- Summary: A large input field for a summary.
- Action Buttons: 'SUBMIT' and 'PREVIEW' buttons at the bottom.

Figure UI09 – Create Bolo Form

This screenshot shows the same 'Create a BOLO' form, but with a different category selection. The 'Category' dropdown now only displays 'Murder'. The other fields and layout are identical to Figure UI09.

Figure UI10 – BOLO Category Rendering



UNCLASSIFIED// FOR OFFICIAL USE ONLY// LAW ENFORCEMENT SENSITIVE

FIU Police Department

11200 SW 8TH ST

Miami, FL

3053053055



Murder -- ACTIVE



BOLO ID: 5839fb283263d39e03ccf6e2

Figure UI11 – BOLO Details View

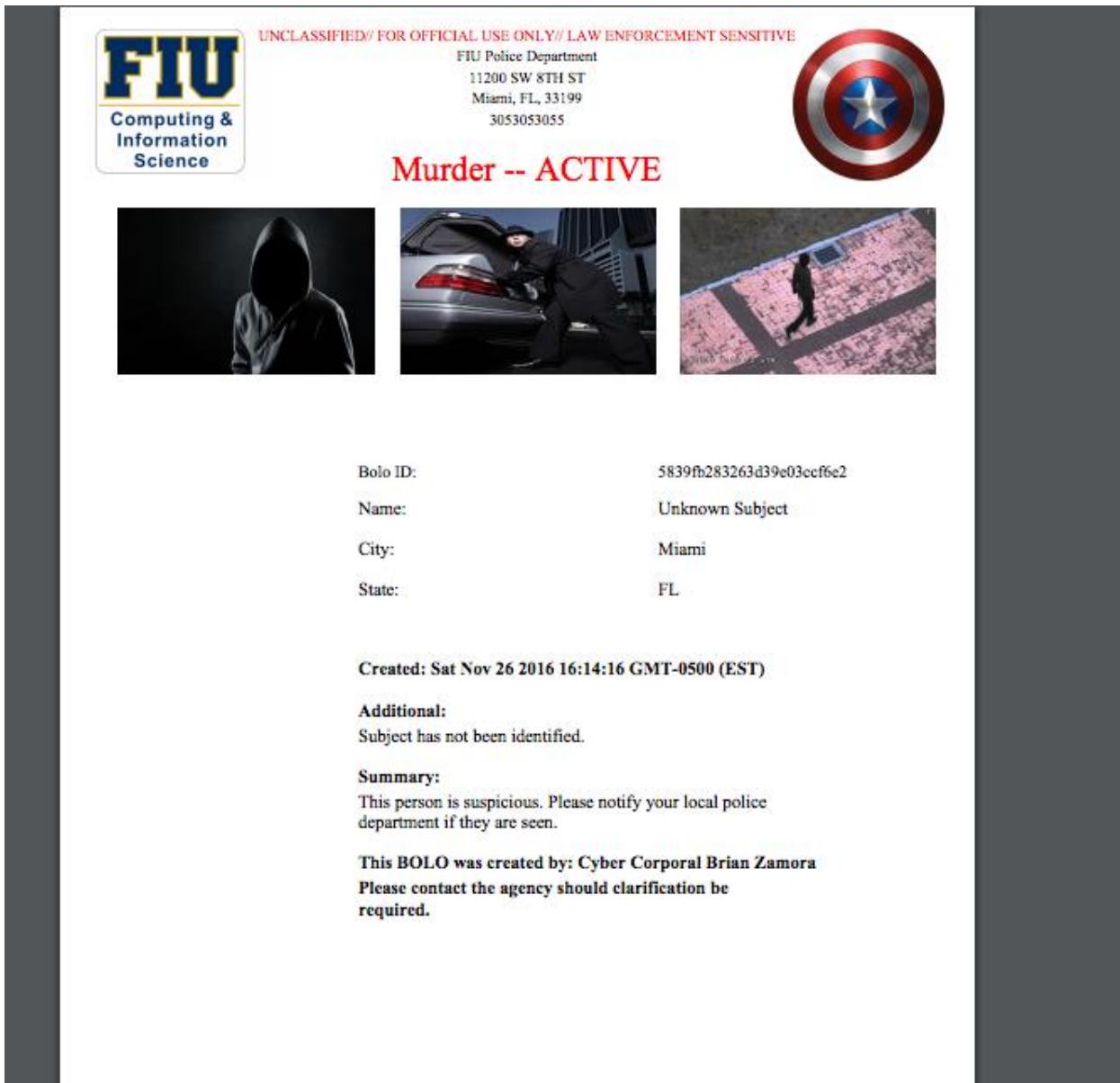


Figure UI12 – BOLO PDF Rendering

Update BOLO Details
[Go Back](#)

Bolo Details	
ID:	5835cee6a647061bc81d128b
Author:	Dominick Martelly
Agency:	Pinterest
Created:	Wed Nov 23 2016 12:16:22 GMT-0500 (Eastern Standard Time)
Date Reported (DD/MM/YYYY)	Time Reported (HH:MM)
<input type="text"/>	<input type="text"/>
Category	
Car	<input type="text"/>
Make	
asdff	<input type="text"/>
Model	
123	<input type="text"/>
Year	
123	<input type="text"/>
Color	
123	<input type="text"/>
Upload New Featured Image	Current Featured Image
<input type="button" value="Choose File"/> No file chosen	
Upload New Other Picture 1	Current Other Picture 1
<input type="button" value="Choose File"/> No file chosen	No Image Found
Upload New Other Picture 2	Current Other Picture 2
<input type="button" value="Choose File"/> No file chosen	No Image Found
Video Link	
<input type="text"/>	
Additional Info	
<input type="text"/>	
Summary	
<input type="text"/>	
<input type="button" value="Update"/>	

Figure UI13 – Edit Bolo

Archived BOLOS

Filter BOLOs By: All Bolos ▾

BOLO 5809263b... has been deleted X

Purge

 No Photo Available Murder Last Updated: 10/21/2016, 10:48:47 AM Reported On: Posted By: Corporal Zamora Details Restore Delete	 Murder Last Updated: 10/20/2016, 4:26:28 PM Reported On: Posted By: Corporal Zamora Details Restore Delete	 Murder Last Updated: 10/20/2016, 3:54:53 PM Reported On: Posted By: Corporal Zamora Details Restore Delete
--	--	---

Figure UI14 – Archived Bolos

Home Create ▾ Archive Agency Directory Agency Directory Search ▾ Admin jason4 ▾

Agency Directory

Name	Address	City	Zip	State	Phone
FIU		Miami		Florida	
Pinecrest	110 NW	Miami	33331	FM	305305305

© 2016 BOLO Flier Creator | Created by School of Computing and Information Sciences @ FIU

Figure UI15 – Agency Directory

The screenshot shows the 'Search BOLOs' page. At the top, there is a navigation bar with links for Home, Create, Archive, Agency Directory, Search, and Admin. On the far right of the navigation bar is a user dropdown labeled 'brian'. Below the navigation bar, the main title 'Search BOLOs' is displayed. Underneath the title are two dropdown menus: 'Agency' (set to 'N/A') and 'Category' (set to 'Select a Category'). A blue 'Search' button is located below these dropdowns.

Figure UI16 – Search Bolo

The screenshot shows the 'Admin Portal: Dashboard' page. At the top, the title 'Admin Portal: Dashboard' is displayed. Below the title, there is a sidebar with links: 'Dashboard', 'Agency Management', 'User Management', and 'Bolo Category Management'. To the right of the sidebar, a welcome message reads: 'Welcome to the BOLO Flier Creator Administrator Dashboard. Please use the links to the left (or above on mobile devices) to navigate the Dashboard.' Below the sidebar, there are three more links: 'Data Analysis', 'Edit Login Page', 'Edit User Guide Page', and 'Edit About Us Page'. At the bottom of the page, a footer bar contains the text '© 2016 BOLO Flier Creator | Created by School of Computing and Information Sciences @FIU'.

Figure UI17 – Admin Dashboard

The screenshot shows the Admin Portal interface for the Agency Management Directory. At the top, there is a navigation bar with links: Home, Create, Archive, Agency Directory, Search, Admin, and a user account section labeled 'jason4'. Below the navigation bar, the title 'Admin Portal: Agency Management Directory' is displayed. On the left, a sidebar menu includes Dashboard, Agency Management (which is selected and highlighted in blue), Data Analysis, Data Subscribers, System Setting, and User Management. The main content area displays a table titled 'Total Agencies: 2'. The table has columns for ID, Name, City, State, Zip, and Phone. It lists two entries: FIU (ID 456) and Pinecrest (ID 123). Both rows have a gear icon at the end. A button labeled 'Add New Agency' is located at the bottom right of the table. At the very bottom of the page, there is a footer note: '© 2016 BOLO Flier Creator | Created by School of Computing and Information Sciences @ FIU'.

ID	Name	City	State	Zip	Phone
456	FIU	Miami	Florida		
123	Pinecrest	Miami	FL	33331	305305305

Add New Agency

© 2016 BOLO Flier Creator | Created by School of Computing and Information Sciences @ FIU

Figure UI18 – Agency Management

Admin Portal: New Agency Registration

Agency Management

Restore Previous Form
Fill out the form and submit to create a new agency.

Agency ID: _____

Agency Name: _____

Agency Domain: @example.fl.gov

Address: _____

City: _____

Sate: AL

Zip Code: _____

Phone Number: _____

Logo Image Upload: Choose File No file chosen

Shield Image Upload: Choose File No file chosen

Watermark Image Upload: Choose File No file chosen

Submit

Figure UI19 – Create New Agency

Home Create Archive Agency Directory Search Admin brian▼

Admin Portal: Data Analysis

Agency Management

Dashboard

Retrieve Data For Analysis

FIU
 Pinecrest

User Management

Bolo Category Management

Data Analysis

Download **Select All** **Deselect All**

Edit Login Page
Edit User Guide Page
Edit About Us Page

© 2016 BOLO Flier Creator | Created by School of Computing and Information Sciences @FIU

Figure UI20 – Data Analysis

The screenshot shows the Admin Portal interface with a dark header bar containing navigation links: Home, Create, Archive, Agency Directory, Search, and Admin. A user profile 'jason4' is visible on the right. The main content area has a title 'Admin Portal: Data Subscriber Management Directory'. On the left is a sidebar with links: Dashboard, Agency Management, Data Analysis, Data Subscribers (which is highlighted in blue), System Setting, and User Management. The main panel displays a table titled 'Total Subscribers: 3' with three rows of data:

ID	Name	Email	Status	Action
123	County	jmedi116@fiu.edu	disabled	
213	fred	nmuno013@fiu.edu		
311	sdesb	sds@sd.com		

Below the table is a button labeled 'Add New Data Subscriber'.

At the bottom of the page, a footer bar contains the text '© 2016 BOLO Flier Creator | Created by School of Computing and Information Sciences @ FIU'.

Figure UI21 – Data Subscribers

The screenshot shows the Admin Portal interface with a dark header bar containing navigation links: Home, Create, Archive, Agency Directory, Search, and Admin. A user profile 'jason4' is visible on the right. The main content area has a title 'Admin Portal: New Data Subscriber Registration'. On the left is a sidebar with links: Dashboard, Agency Management, Data Analysis, Data Subscribers (which is highlighted in blue), System Setting, and User Management. The main panel contains a form with instructions: 'Fill out the form and submit to create a new Subscriber.' It includes three input fields: 'Data Subscriber ID' (empty), 'Data Subscriber Name' (empty), and 'Email' (empty). Below the fields is a blue 'Submit' button.

At the bottom of the page, a footer bar contains the text '© 2016 BOLO Flier Creator | Created by School of Computing and Information Sciences @ FIU'.

Figure UI22 – Create Data Subscriber

Home Create Archive Agency Directory Search Admin

Admin Portal: System Settings

Dashboard

Agency Management

Data Analysis

Data Subscribers

System Setting

User Management

Set the system settings

Brute Force Login (3 - 10) Attempts

System Timeout (10 - 240) Minutes

Submit

© 2016 BOLO Flier Creator | Created by School of Computing and Information Sciences @ FIU

Figure UI23 – System Settings

Home Create Archive Agency Directory Search Admin brian

User Management Sort

Dashboard

Agency Management

User Management

Bolo Category Management

Data Analysis

Edit Login Page

Edit User Guide Page

Edit About Us Page

Total Users: 2

Username	Agency	Name	Email	Role	Active
brian	FIU	Brian Zamora	bzamo007@fiu.edu	ROOT	Active
pbrian	Pinecrest	Brian Zamora	brian410zamora@hotmail.com	ROOT	Active

© 2016 BOLO Flier Creator | Created by School of Computing and Information Sciences @ FIU

Figure UI24 – User Management

User bolouser has been registered! ×

Dashboard Agency Management User Management Bolo Category Management Data Analysis Edit Login Page Edit User Guide Page Edit About Us Page	<p>Fill out the form and submit to create a new user.</p> <table border="0"> <tr> <td>Username</td> <td><input type="text"/></td> </tr> <tr> <td>First Name</td> <td><input type="text"/></td> </tr> <tr> <td>Last Name</td> <td><input type="text"/></td> </tr> <tr> <td>Agency</td> <td><input type="text" value="Select an Agency"/></td> </tr> <tr> <td>Email</td> <td><input type="text"/> N/A</td> </tr> <tr> <td>Role</td> <td><input type="text" value="OFFICER"/></td> </tr> </table>	Username	<input type="text"/>	First Name	<input type="text"/>	Last Name	<input type="text"/>	Agency	<input type="text" value="Select an Agency"/>	Email	<input type="text"/> N/A	Role	<input type="text" value="OFFICER"/>
Username	<input type="text"/>												
First Name	<input type="text"/>												
Last Name	<input type="text"/>												
Agency	<input type="text" value="Select an Agency"/>												
Email	<input type="text"/> N/A												
Role	<input type="text" value="OFFICER"/>												

Figure UI25 – Create New User

Account Management

Account Details Change Password Notifications	<p>brian</p> <table border="0"> <tr> <td>Name</td> <td>Brian Zamora</td> </tr> <tr> <td>Email</td> <td>bzamo007@fiu.edu</td> </tr> <tr> <td>Badge Number</td> <td>4128999</td> </tr> <tr> <td>Section / Unit</td> <td>Cyber</td> </tr> <tr> <td>Rank / Title</td> <td>Corporal</td> </tr> <tr> <td>Role</td> <td>ROOT</td> </tr> <tr> <td>Agency</td> <td>FIU</td> </tr> </table> <p>Update Details</p>	Name	Brian Zamora	Email	bzamo007@fiu.edu	Badge Number	4128999	Section / Unit	Cyber	Rank / Title	Corporal	Role	ROOT	Agency	FIU
Name	Brian Zamora														
Email	bzamo007@fiu.edu														
Badge Number	4128999														
Section / Unit	Cyber														
Rank / Title	Corporal														
Role	ROOT														
Agency	FIU														

© 2016 BOLO Flier Creator | Created by School of Computing and Information Sciences @FIU

Figure UI26 – Account Details

The screenshot shows a dark-themed web application interface. At the top, there is a navigation bar with links for Home, Create, Archive, Agency Directory, Search, Admin, and a user icon labeled "User". Below the navigation bar, the title "User Password Reset Page" is displayed. The main content area contains a "Password Reset" section with fields for "Password" and "Repeat Password", both represented by empty input boxes. A blue "Submit" button is located below these fields. At the bottom of the page, a dark footer bar displays the copyright notice: "© 2016 BOLO Flier Creator | Created by School of Computing and Information Sciences @FIU".

Password Reset

The new password must contain(uppercase and lowercase letters, numbers and special characters) and no more than 2 equal characters in a row. Valid special characters: ! ~ < > , ; : _ = ? * + # . " & % () | [] - \$ ^ @ /

Password

Repeat Password

Submit

© 2016 BOLO Flier Creator | Created by School of Computing and Information Sciences @FIU

Figure UI27 – Change Password

The screenshot shows a dark-themed web application interface. At the top, there is a navigation bar with links for Home, Create, Archive, Agency Directory, Search, Admin, and a user icon labeled "User". Below the navigation bar, the title "User Password Reset Page" is displayed. The main content area contains a "Password Reset" section with fields for "Password" and "Repeat Password", both represented by empty input boxes. A red error message "Password is not valid" is displayed above the "Password" field. A small "X" icon is located to the right of the message. A blue "Submit" button is located below these fields. At the bottom of the page, a dark footer bar displays the copyright notice: "© 2016 BOLO Flier Creator | Created by School of Computing and Information Sciences @FIU".

Password Reset

The new password must contain(uppercase and lowercase letters, numbers and special characters) and no more than 2 equal characters in a row. Valid special characters: ! ~ < > , ; : _ = ? * + # . " & % () | [] - \$ ^ @ /

Password

Repeat Password

Submit

© 2016 BOLO Flier Creator | Created by School of Computing and Information Sciences @FIU

Figure UI28 – Password Verification

The screenshot shows the 'Category Management' section of the Bolo Flier Creator interface. On the left, there's a sidebar with links like 'Dashboard', 'Agency Management', 'User Management', 'Bolo Category Management' (which is currently selected), and 'Data Analysis'. The main area displays a table titled 'Total Categories: 4' with four entries: 'Testing' (Created On: Fri Oct 07 2016 16:38:04 GMT-0400 (EDT)), 'HIYA' (Created On: Sat Oct 08 2016 13:21:16 GMT-0400 (EDT)), 'Murder' (Created On: Tue Oct 11 2016 21:07:19 GMT-0400 (EDT)), and 'BOAT' (Created On: Fri Oct 14 2016 15:33:51 GMT-0400 (EDT)). Below the table is a button labeled 'Add New Category'. At the bottom of the page, there's a footer with the copyright notice: '© 2016 BOLO Flier Creator | Created by School of Computing and Information Sciences @FIU'.

Figure UI29 – Category Management

The screenshot shows the 'New Category Creation' page. The sidebar on the left includes 'Dashboard', 'Agency Management', 'User Management', 'Bolo Category Management' (selected), and 'Data Analysis'. The main content area has a form for creating a new category. It starts with a text input field labeled 'Enter A Name For The New BOLO Category' containing 'Robbery Suspect'. Below it is a section titled 'Enter Fields' with three input fields: 'Name' (with a red minus sign icon to its right), 'Last Known Location' (with a red minus sign icon to its right), and 'Phone Number' (with a green plus sign icon to its right). At the bottom of the form are two buttons: 'Submit' and 'Preview BOLO Template'.

Figure UI30 – Category Creation

[Back to Category](#)

Robbery Suspect BOLO

The current Date and Time is Tue Nov 29 2016 11:17:17 GMT-0500 (EST)

Date Reported 29/10/2016	Time Reported 11:17	
Name <input type="text"/>		
Last Known Location <input type="text"/>		
Phone Number <input type="text"/>		
Featured Image <input type="file"/> no file selected Choose File	Image Upload <input type="file"/> no file selected Choose File	Image Upload <input type="file"/> no file selected Choose File
Video Link <input type="text"/>		
Additional Info <input type="text"/>		

Figure UI31 – Category Preview

Edit About Us

[Dashboard](#) [AboutUs Markdown Editor](#)
[Agency Management](#) [Write text here...](#)
[User Management](#)
[Bolo Category Management](#)

[Data Analysis](#)

[Edit Login Page](#)
[Edit User Guide Page](#)
[Edit About Us Page](#)

© 2016 BOLO Flier Creator | Created by School of Computing and Information Sciences @FIU

Figure UI32 – Edit About Us

Home Create Archive Agency Directory Search Admin brian▼

Edit User Guide

Dashboard Agency Management User Management **Bolo Category Management** Preview Data Analysis

[Save](#)

UserGuide Markdown Editor
Write text here...

Edit Login Page Edit User Guide Page Edit About Us Page

© 2016 BOLO Flier Creator | Created by School of Computing and Information Sciences @FIU

Figure UI33 – Edit User Guide

The screenshot shows a web browser window with multiple tabs open, including "BOLO Ver 6.0 - Google Chrome". The main content area displays four cards arranged horizontally:

- NEW THEFT - AUTO**: Features an image of a Batmobile. Details: Updated: 09-06-16 11:11:22, Created On: 09-06-16 11:11:22, Posted By: Pinecrest. Buttons: Edit, Details, Archive.
- NEW THEFT - BOAT**: Features an image of an F/A-18 Hornet. Details: Updated: 09-06-16 11:10:24, Created On: 09-06-16 11:10:24, Posted By: Pinecrest. Buttons: Edit, Details, Archive.
- IN CUSTODY SITUATIONAL AWARENESS**: Features an image of Captain America. A large red watermark "In Custody" is overlaid. Details: Updated: 09-06-16 10:35:06, Created On: 09-05-16 20:26:34, Posted By: Pinecrest. Buttons: Edit, Details, Archive.
- LOCATED WANTED**: Features an image of the Hulk. A large red watermark "Located" is overlaid. Details: Updated: 09-06-16 10:34:34, Created On: 09-05-16 15:07:07, Posted By: Pinecrest. Buttons: Edit, Details, Archive.

Figure UI34 – BOLO Watermark

Appendix C – Sprint Reports

Sprint Review

Sprint 1 Review

August 29 - September 9

Date: September 9th, 2016

Attendees: Dominick Martelly, Brian Zamora, Jason Cohen, Frank Jagger

Start time: 2:00pm

End time: 2:30pm

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

- #1777_ViewAllPictures
- #1187_AddWatermark
- #1186_AddNewUserButton
- #1179_BatchUploadUsers

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.

- #1179_BatchUploadUsers
 - The product owner wanted a different design for the batch upload
- #1182_EditAboutUs
 - We were not able to get to this user story during this sprint
- #1183_LoginHomePage
 - We were not able to get to this user story during this sprint
- #1184>EditUserGuide
 - We were not able to get to this user story during this sprint
- #1189_AgencySpecificBOLO
 - We were not able to get to this user story during this sprint
- #1180_SetPassword
 - This user story will be implemented at a later time, the product owner has expressed that it is not a priority for this iteration.

Sprint 2 Review

September 10 - September 23

Date: September 23rd, 2016

Attendees: Dominick Martelly, Brian Zamora, Jason Cohen, Frank Jagger

Start time: 2:00pm

End time: 2:30pm

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

- #1179_BatchUploadUsers
- #1182_EditAboutUs
- #1183_LoginHomePage
- #1184_EditUserGuide
- #1189_AgencySpecificBOLO
- #1180_SetPassword
- #1190_AgencySpecificView

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.

- #1180_SetPassword
 - This user story will be implemented at a later time, the product owner has expressed that it is not a priority for this iteration.

Sprint 3 Review

September 24 - October 7

Date: October 7th, 2016

Attendees: Dominick Martelly, Brian Zamora, Jason Cohen

Start time: 1:50pm

End time: 2:10pm

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

- #1191_EmailPDFPrivacy
- #1194_BOLOCategoryCreation
- #1196_BOLOCategoryRendering

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.

- #1198_AdjustBOLOViewForMongoDB
 - The product owner wants the view of bolo details to use more of the user space for pictures and details of the bolo
- #1197_Email Confirmation
 - The email confirmation system is going to be changed, because of an upcoming change in the database. As a result, this user story will be moved to sprint 4

Sprint 4 Review

October 7 - September 21

Date: September 21st, 2016

Attendees: Dominick Martelly, Brian Zamora, Jason Cohen

Start time: 1:50pm

End time: 2:10pm

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

- #1188_ChangeToMongoDB
- #1192_DeleteUser
- #700_DeleteAgency
- #1195_AdjustCSVUploadingMongoDB
- #1199_PDFRenderingMongoDB
- #1197_Email Confirmation
- #1200_PDF Email Privacy
- #1198_AdjustBOLOViewMongoDB

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.

- #1193_BOLOImage Compression
 - The product owner wanted us to look for a new image compression tool or algorithm so that generated PDFs would be below 5MB.
- #1202_BOLOImageResizing
 - The product owner wanted images to be as big as they possibly can be based on how many images are going to be displayed

Sprint 5 Review

October 22 - November 4

Date: November 4th, 2016

Attendees: Dominick Martelly, Brian Zamora, Jason Cohen, Frank Jagger

Start time: 1:35pm

End time: 1:45pm

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

- #1202_BOLOImageResizing
- #1193_BOLOImageCompression
- #1180_SetPassword
- #1206_NewUserNotification
- #1180_SetPassword

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.

- No Implementations were rejected

Sprint 6 Review

November 5 - November 11

Date: November 11th, 2016

Attendees: Dominick Martelly, Brian Zamora, Frank Jagger

Start time: 1:35pm

End time: 1:45pm

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

- #1201_BOLODataSubscriber
- #1205_BOLOPasswordVerification

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.

- No Implementations were rejected

Sprint Retrospective

Sprint 1 Retrospective

August 29 - September 9

Date: September 9th, 2016

Attendees: Dominick Martelly, Joseph Casal, Brian Zamora

Start time: 3:00pm

End time: 3:30pm

What went wrong?

- Did we do a good job estimating our team's velocity?
 - No
- Did we do a good job estimating the points (time required) for each user story?
 - Yes, the points reflects the amount of work needed to be done for each user story
- Did each team member work as scheduled?
 - Yes, but the schedule needs to be filled out more

What went right?

- Implementing at least one of the assigned user case for that sprint
- Good communication with the product owner

How to address the issues in the next sprint?

- How to improve the process?
 - Everyone comes to the meeting on time
 - Increase the amount of work hours for each sprint
- How to improve the product?
 - By making the GUI more user friendly and informative
 - Improving ranking algorithms
 - Implement functionality that will allow for root/admin user to dynamically create their own templates and customize the homepage

Sprint 2 Retrospective

September 10 - September 23

Date: September 23rd, 2016

Attendees: Dominick Martelly, Brian Zamora

Start time: 3:00pm

End time: 3:30pm

What went wrong?

- Did we do a good job estimating our team's velocity?
 - No
- Did we do a good job estimating the points (time required) for each user story?
 - Yes, the points reflects the amount of work needed to be done for each user story
- Did each team member work as scheduled?
 - No, Joseph has not been communicating very well with the team.
 -

What went right?

- We were able to implement several highly desired features
- Good communication with the product owner

How to address the issues in the next sprint?

- How to improve the process?
 - Everyone comes to the meeting on time
 - Increase the amount of work hours for each sprint
- How to improve the product?
 - Add more functionality
 - Clean up PDF Rendering
 - Improve back end method calls
 - communication with the product owners, the mentors and each other.

Sprint 3 Retrospective

September 24th - October 7th

Date: October 7th, 2016

Attendees: Dominick Martelly, Brian Zamora

Start time: 3:00pm

End time: 3:30pm

What went wrong?

- Did we do a good job estimating our team's velocity?
 - Yes
- Did we do a good job estimating the points (time required) for each user story?
 - Yes, the points reflects the amount of work needed to be done for each user story
- Did each team member work as scheduled?
 - No, Joseph has not been communicating very well with the team. Is not attending meetings
 - We have requested that Joseph be removed from the team

What went right?

- We began to transition the application from IBM Cloudant to MongoDB
- Good communication with the product owner
- Gained great understanding of the system

How to address the issues in the next sprint?

- How to improve the process?
 - Everyone comes to the meeting on time
 - Increase the amount of work hours for each sprint
- How to improve the product?
 - Keep improving back end method calls
 - Improve communication with the persistence
 - Improve back end method calls

Sprint 4 Retrospective

October 8th - October 21st

Date: September 21st, 2016

Attendees: Dominick Martelly, Brian Zamora

Start time: 3:00pm

End time: 3:30pm

What went wrong?

- Did we do a good job estimating our team's velocity?
 - Yes we believe we did a good job
- Did we do a good job estimating the points (time required) for each user story?
 - This sprint we did a decent job, but we could have done better. Some user stories proved more difficult to implement than we anticipated
- Did each team member work as scheduled?
 - Yes, the team met together for daily scrums and accomplished their user stories in timely manner.

What went right?

- We were able to complete the transition into MongoDB
- We were able to rebuild the basic system functionality using MongoDB
- Good teamwork

How to address the issues in the next sprint?

- How to improve the process?
 - Improve our foresight when planning our user stories
 - Manage our time more carefully
- How to improve the product?
 - Improve the overall user interface
 - Add more filters for the main BOLO page, to allow for sorting by category, agency

Sprint 5 Retrospective

October 22th - November 4th

Date: November 4th, 2016

Attendees: Dominick Martelly, Brian Zamora

Start time: 3:00pm

End time: 3:30pm

What went wrong?

- Did we do a good job estimating our team's velocity?
 - Yes we did a good job but it could have been better.
- Did we do a good job estimating the points (time required) for each user story?
 - Yes this sprint we did a really good job. We were able to accomplish all of our user stories
- Did each team member work as scheduled?
 - Yes all team members worked as scheduled and put in the required work to accomplish the user stories

What went right?

- Accomplished all user stories for the sprint
- Good team communication
- Overcame small challenges by working together

How to address the issues in the next sprint?

- How to improve the process?
 - Maintain good team communication
 - Don't take on more user stories than possible
- How to improve the product?
 - Focus on error handling
 - Focus on ensuring BOLO6 functions the same as BOLO5

Sprint 6 Retrospective

November 5 - November 11

Date: November 11th, 2016

Attendees: Dominick Martelly, Brian Zamora

Start time: 4:00pm

End time: 4:30pm

What went wrong?

- Did we do a good job estimating our team's velocity?
 - This sprint our velocity estimate was highly accurate
- Did we do a good job estimating the points (time required) for each user story?
 - Yes this sprint we did a really good job. We were able to accomplish all of our user stories
- Did each team member work as scheduled?
 - Yes, all team members worked above and beyond to accomplish all user stories

What went right?

- Accomplished all user stories for the sprint
- Good team communication
- Product owner was happy with the work that was done

How to address the issues in the next sprint?

- How to improve the process?
 - Maintain good team communication
 - Maintain good velocity estimate
- How to improve the product?
 - Focus on error handling
 - Focus on ensuring BOLO6 functions the same as BOLO5

Appendix D – Other Documents

User Guide

Officers

Supervisors

Administrators

Agency management

- 1) Click on 'Admin'
- 2) Click on 'Agency Management'
- 3) Click on the 'Blue Asterisk' to the right on the agency
- 4) Enter or Modify any fields (required)
- 5) Enter or update 'Logo' image (required)
- 6) Enter or update 'Shield' image (required)
- 7) Enter or update 'Water Mark' image(optional)
- 8) Choose to allow officers to opt out of viewing BOLOs from their agency
- 9) Click 'Submit'

Category management

- 1) Click on 'Admin'
- 2) Click on "BOLO Category Management"
- 3) To create a new Category:
 1. Click on "Add New Category"
 2. Fill out the Category Name
 3. Add as many fields as you need by clicking on the green "+" symbol
 4. Delete fields by clicking on the red "-" symbol
 5. Once you have finished creating the fields of the new Category click Preview
 6. Review the Category and go back if you want to make any changes
 7. Once you are satisfied with the Category click Submit and the Category will be created
- 4) Please Note that once a Category is created it can't be deleted because other BOLOs may already be using it.
- 5) Editing Categories should only be used if no BOLOs are using the category
 1. Select on the Category you want to edit
 2. This will open up Category details
 3. Click Edit
 4. Make changes as you see fit and then click submit
 5. The Category is now updated

User Management

- 1) Click on 'Admin'
- 2) Click on 'User Management'
* * Add a new user's profile * *
- 3) Click on 'Add New User'
- 4) Fill in the empty fields(required)
- 5) Click 'Submit'
* * Add a new user's profile * *
- 3) Click on the 'Blue Asterisk' on the right of the user
- 4) Update required fields
- 5) Click 'Submit'

Create a BOLO**Click 'Create'****Fill In the Date Reported and Time Reported Fields**

- 1) Select a Category
- 2) The Form will now be updated with the proper fields
- 3) Fill out all the required fields for that category
- 4) Upload any images that you might have, max is 3
- 5) Fill out the Video Link, Additional Info and Summary fields, if applicable
- 6) Select Preview
- 7) Go over the BOLO to ensure it looks exactly how you want it to
- 8) Repeat steps 1-8 until you are satisfied with the BOLO
- 9) Click Submit
- 10) Go to your registered email and click on the confirmation to finish creating your BOLO flier

Edit a BOLO*** * You can only edit your BOLOs * ***

- 1) Click 'Edit'
- 2) Select any field you want to edit
- 3) Enter any information required
- 4) Click 'Submit'

View Details of a BOLO

- 1) Click 'Details'
- 2) Review details of a selected BOLO
- 3) Select 'Generate PDF' to download
- * * View a detailed record of who has edit this BOLO * *
- 5) Select 'View Record'
- 6) Click 'Back' to go back to the BOLO details

Filter by (Me)

- 1) Click 'Filter By'
- 2) Select filter:
 - * My BOLO
 - * My Agency
 - * All BOLOs
- 3) Results BOLOs will display

Filter by Agency

- 1) Click 'Select Agency'
- 2) Select one or more agencies
- 3) Click 'Submit'
- 4) Results BOLOs will display

Archive

- 1) Click 'Archive'
* * At the top of the page* *
- 2) Select a BOLO
- 3) Click:
 - * 'View' to view details
 - * 'Restore' to restore the BOLO
 - * 'Delete' to delete a BOLO
* * On the BOLO * *

Send the BOLO to the archive files

Agency Directory

- 1) Click on 'Agency Directory' to view a list of partnering agencies.
- 2) Select an agency by clicking their name
 - * View contact information
 - * Verify an agency's 'Logo' and 'Shield'

Search BOLO by field(s)

- 1) Click 'Search'
- 2) Select any field(s) you wish to search.
- 3) Enter information value of field(s).
* * Wild card search is for something you can not search with an optional search * *
- 4) Click 'Search'

Get Notifications

- 1) Click on your 'Username'
- 2) Click on 'Settings' from the dropdown
- 3) Click on 'Notifications'
- 4) Click on 'Subscribe to other Agencies at the bottom left'
- 5) Select agencies to receive notifications
* * You may also unsubscribe from an agency * *
- 6) Select agencies to remove from your list of Agencies
- 7) Click 'Unsubscribe Selected'

Reset Password

- 1) Click on your 'Username'
- 2) Click on 'Settings' from the dropdown
- 3) Click 'Change Password'
- 4) Enter new password
- 5) enter confirmation password
- 6) Click 'Submit'

Installation / Maintenance

1. You are going to need a GitHub account, NodeJS, MongoDB, Git tools, a Sendgrid API key and an IDE. Our IDE of choice is Webstorm, but feel free to use whichever one you feel comfortable with.
2. Install NodeJs, MongoDB, your IDE and Git tools.
3. Go to GitHub and go to the project repository and get the clone link
4. Open up a bash shell, and type in “*git clone*” followed by the link from the repository.
5. While the project is downloading, set up MongoDB so that it points to the local host.
6. Once the repository is cloned, use a CLI, either a command prompt or a bash shell, to navigate to the project folder. Once there navigate to the “Code” folder, and type in “*npm install*”. This will install all the dependencies onto your machine.
7. While that is installing you need to create your Sendgrid API key. Go to the Sendgrid website and make an account, it is free.
8. Once your account is setup, go to settings and find the option that says “Generate API Key”.
9. Follow the steps and generate the API Key. DO NOT NAVIGATE AWAY FROM THE WEBPAGE, unless you have copied the API key to a text document or your clipboard. It will disappear and you won’t be able to view it again.
10. Open up your IDE and then open up the project folder. Once it is loaded up, you are going to look for a file named .env.example. Make a copy of this file and name it .env. This file has a line that dedicated to the Sendgrid API Key, locate that line and past your Sendgrid API key into it.
11. Your dependencies should be finished installing by now, but if they haven’t just wait. Then use either a bash shell or a command prompt and navigate to the MongoDB folder.
12. Ensure that you have root privileges or administrative rights, then navigate to the bin folder. Here type the command “*mongod*”, depending on your operating system you might need to use the command “*./mongod*” instead. This will start the database.
13. Next navigate back to your IDE and start the web application. Your application should start with no issues and now you are ready to begin using and or coding for the BOLO Flier Creator web application.
14. If you get an error please ensure that MongoDB is running, that no other service is using port 3000, and also that all your dependencies are installed correctly.

Shortcomings and Wish list

During the 6th iteration of BOLO Flier Creator, we were able to add more functionality to an already usable piece of software however there were still some things left to me desired. In future iterations of the BOLO Flier Creator web application we would like to see the following additions / improvements:

- Transition away from using Sendgrid or any other third party services. This is due to security risks and policies that are being observed by the product owner.
- A more user friendly and intuitive interface for editing the Login, User Guide and About Us pages.
- A more intuitive interface for creating and editing BOLO Categories.
- When rendering the BOLO as a PDF the document looks a little lackluster, although a watermark can be uploaded, maybe a border will help the document look like a proper law enforcement document
- It would also be very useful to add more system settings that the root or administrator can utilize.
- A system wide log that can be seen by the root users would also be useful. It would contain information such as which users have been deleted/deactivated and by whom.
- A system wide data collection portal would also be useful for collecting statistics. For example, it can show how many BOLOs each user has created, which agency/department creates the most BOLOs, and when is the most common time that BOLOs are created.

References

Govett, Devon (2016) pdfkit source code (0.7.2) [Source code].
<https://www.npmjs.com/package/pdfkit>,

Mueller, Patrick (2016) cfenv source code (Version 1.0.3) [Source code].
<https://www.npmjs.com/package/body-parser>,

Smith, J (2016) cloudant source code (Version 1.4.1) [Source code].
<https://www.npmjs.com/package/cloudant>,

Smith, J (2016) cfenv source code (Version 1.0.3) [Source code].
<https://www.npmjs.com/package/cfenv>,

Wilson, Doug (2016) body-parser source code (Version 1.13.2) [Source code].
<https://www.npmjs.com/package/body-parser>

Libraries:

```
"bcrypt-nodejs": "^0.0.3",
"bluebird": "latest",
"body-parser": "^1.15.2",
"brfs": "^1.4.3",
"browserify": "^13.1.0",
"cfenv": "^1.0.3",
"connect-flash": "^0.1.1",
"cookie-parser": "^1.4.3",
"csurf": "^1.9.0",
"csv": "^1.1.0",
"csvtojson": "^1.0.3",
"dotenv": "^1.2.0",
"errorhandler": "1.4.1",
"express": "4.12.x",
"express-messages": "*",
"express-session": "1.7.6",
"express-validator": "*",
"fast-csv": "^2.1.0",
"gm": "^1.21.1",
"hat": "0.0.3",
"iframe": "^0.3.1",
"imagemin": "^4.0.0",
"j-i-c": "^2.0.2",
"jimp": "^0.2.9",
"json2csv": "^3.6.2",
"jstransformer-markdown-it": "^1.0.0",
"lodash": "^3.10.1",
"marked": "latest",
"method-override": "2.3.3",
"moment": "*",
"moment-timezone": "^0.5.5",
"mongodb": "latest",
"mongoose": "^4.6.3",
"morgan": "1.6.1",
"multer": "^1.2.0",
"multiparty": "^4.1.2",
"node-markdown": "^0.1.1",
"node-phantom": "^0.2.5",
"node-uuid": "^1.4.3",
"node-wkhtml": "^0.1.0-alpha1",
"nodepdf": "1.2.1",
"passport": "^0.3.0",
"passport-local": "^1.0.0",
"pdfkit": "^0.7.2",
"phantomjs": "^2.1.3",
```

```
"promise": "^7.0.4",
"pug": "^2.0.0-beta6",
"request": "^2.64.0",
"sendgrid": "^2.0.0",
"serve-favicon": "^2.3.0",
"socket.io": "^1.0.0",
"socket.io-express-session": "^0.1.3",
"validate.js": "^0.9.0",
"wkhtmltopdf": "^0.1.5",
"node": "0.12.x",
"chai": "^3.3.0",
"gulp": "^3.9.1",
"gulp-confirm": "latest",
"gulp-exec": "latest",
"istanbul": "^0.3.22",
"jsdoc": "^3.3.3",
"jshint": "^2.8.0",
"mocha": "^2.3.3",
"mockery": "^1.4.0",
"nock": "^2.13.0",
"sinon": "^1.17.0"
```