*Florida International University*

*School of Computing and Information Sciences*

CIS 4911 - Senior Capstone Project

Software Engineering Focus

Final Deliverable

Senior Project Portal 1.0

**Team Members**

Christopher Jones

**Product Owner**: Masoud Sadjadi Ph.D.

**Instructor**: Masoud Sadjadi Ph.D.

Copyright and trademark notices, restrictions on copying or distributing the documentation, information for contacting the issuing organization (reader’s comments), warranties, contractual obligations or disclaimers, and general warnings and cautions.

***Abstract***

*The Senior Project Tools system is a powerful tool which enables Senior Project students to have a quick and simple method for accessing the web applications currently being used to drive Senior Project. It will greatly improve the transition for new Senior Project students into the course while not affecting the independence of the other Senior Project sites. The Final Deliverable document includes a detailed analysis of the Senior Project Portal system as of version 1.0. The main purpose of this document is to facilitate, in a well-documented manner, all of the information about the system.*

*This document includes extensive details about the current system’s user stories, the newly developed user stories implemented during the last seven sprints of version 1.0, and in-depth explanation of the software architecture used, the design patterns implemented to enhance the solution, testing criteria, analysis, UML diagrams, pictures of the system’s features, a breakdown of all sprint review meetings, and sprint retrospective reviews. Ultimately this document is a powerful guide that documents all the work done as of version 1.0. It also, serves as a structured-guide for future developers so that they can quickly become familiar with the system.*

Contents

[Introduction 7](#_Toc437528630)

[Current System 8](#_Toc437528631)

[Purpose of New System 8](#_Toc437528632)

[User Stories 10](#_Toc437528633)

[Implemented User Stories 10](#_Toc437528634)

[Pending User Stories 16](#_Toc437528635)

[Project Plan 17](#_Toc437528636)

[Hardware and Software Resources 17](#_Toc437528637)

[Feasibility Study 20](#_Toc437528638)

[Sprints Plan 21](#_Toc437528639)

[Sprint 1 21](#_Toc437528640)

[Sprint 2 22](#_Toc437528641)

[Sprint 3 22](#_Toc437528642)

[Sprint 4 23](#_Toc437528643)

[Sprint 5 23](#_Toc437528644)

[Sprint 6 24](#_Toc437528645)

[Sprint 7 25](#_Toc437528646)

[GANTT CHART 28](#_Toc437528647)

[System Design 30](#_Toc437528648)

[Architectural Patterns 30](#_Toc437528649)

[System and Subsystem Decomposition 31](#_Toc437528650)

[Deployment Diagram 32](#_Toc437528651)

[System Validation 33](#_Toc437528652)

[Glossary 34](#_Toc437528653)

[Appendix 35](#_Toc437528654)

[Appendix A - UML Diagrams 35](#_Toc437528655)

[Static UML Diagrams 35](#_Toc437528656)

[36](#_Toc437528657)

[Dynamic UML Diagrams 39](#_Toc437528658)

[40](#_Toc437528659)

[41](#_Toc437528660)

[Appendix B - User Interface Design 47](#_Toc437528661)

[Appendix C - Sprint Review Reports 51](#_Toc437528662)

[Appendix D - Sprint Retrospective Reports 55](#_Toc437528663)

[References 58](#_Toc437528664)

# Introduction

This chapter introduces the current version of Senior Project Portal. The document is divided into the following sections. User Stories which includes information about the user stories that were generated per demand of the product owners. Project Plan which includes details about how the work of the project was planned. System Design which provides detailed information about the architectural patterns, system, subsystems, deployment diagram, and design patterns. The System Validation section which describes the tests used to validate the system. The Glossary, which covers key terms used in the document. The Appendix section which has essential diagrams, UI pictures, sprint review reports, and retrospective reports. The References section, which has all of this document’s references

## Current System

The current system exists as the system that was created during this development cycle, seeing that it is Senior Project Portal Version 1.0. It was developed by Christopher Jones with the overview and guidance of Dr. Sadjadi.

The Senior Project Portal project was introduced by Dr. Masoud Sadjadi during the start of the Senior Project class in the fall of 2015. It was originally introduced as “a portal site, using the authentication system of Virtual Job Fair, which will combine the login systems of the four Senior Project Sites: Virtual Job Fair, Collaborative Platform, Mobile Judge, Senior Project Website”. It was initially developed by Christopher Jones with the overview and guidance of Dr. Sadjadi.

From now on, every semester, students are assigned with the task of improving past features and adding new features to the system.

## Purpose of New System

The Senior Project Portal system was established during this semester. The purpose of the new Senior Project Portal system is to centralize the access to the Senior Project sites in order to greatly improve convenience for current and future Senior Project students. The site should allow users to log in with their FIU Gmail credentials and be taken to a home page where they can select from a list of links to any of the Senior Project sites. When a link is activated, the user should be redirected to the site of their choice and logged in automatically in order to avoid entering credentials multiple times for each site they choose. This property of access control is known as single sign-on. In addition, the portal system should have administrator features that allow a registered administrator to modify the current list of sites on the portal (add, modify and delete sites). In the following system we will give a more thorough list of implemented features.

**New System Features and Enhancements**

1. User Subsystem
2. **Student Login**

FIU Students can access the portal by logging in with their FIU Gmail credentials. After entering their credentials, the system redirects them to Google in order to verify the FIU Gmail email which was entered. Afterwards, the user is redirected to the portal home page where they are provided with the list of Senior Project sites.

1. **Admin Login**

System administrators can access the portal by logging in with their administrator account credentials. The system detects that the username is that of an administrator account, and validates the login credentials. If successful, the user is sent to the administrator home page where they are immediately presented with the current list of Senior Project sites.

1. **Site Add**

Admins have the ability to add a new site to the portal by entering a new site name and site url.

1. **Site Delete**

Admins have the ability to delete a site from the portal by selecting the “delete” button next to a specific Senior Portal site on the list presented to admins on the admin home page.

1. **Site Update**

Admins have the ability to update site info by selecting the “update” button next to a specific Senior Portal site on the list presented to admins on the admin home page.

1. **OAuth Daemon for Google**

OAuth Daemon is a stand-alone web Background API server which was used for my Google authentication feature (FIU Gmail authentication). It completely abstracted away from the complexity of OAuth 2.0 integration and can be used to seamlessly add additional API providers in future releases. In addition to using it for authentication on the Senior Project Portal, I also implemented it on Virtual Job Fair to replace the previous OAuth 2.0 solution for LinkedIn and Google. With the implementation of OAuth Daemon, we are greatly improving the ease of utilizing API for authentication by allowing the server to handle API connections in a manner where a web application can simply ask for data and receive data (ex. Facebook email, Facebook status) without needing to understand how the data was retrieved.

1. **Single Sign-On**

Single sign-on capability is a property which allows the user to log in with a single ID and password and gain access to the other Senior Project sites without using different credentials for each site. After logging in to the portal site, a student user can select from a list of Senior Project site links and be logged in automatically to which ever site they are redirected to. This feature is the main feature of the portal and is the central reason that the portal will be a necessity for future Senior Project classes.

# User Stories

This chapter presents Senior Project Portal 1.0 user stories. In the following sections, we will refer to all the user stories implemented during the Then we will list all pending user stories available for implementation in the next product release.

## Implemented User Stories

**User Story #1 – Prepare Development Environment**

**Description:**

* The development environment needs to be set up and configured on my machine in order to begin development tasks for the project

**Acceptance Criteria:**

* WAMP server installed/configured
* VJF site is hosted locally and functional
* DBMS is configured

**User Story #2 – Pull latest version of Github repository onto development site**

**Description:**

* The current version of the VJF application on the development virtual machine must be the most recent version of the project, so it is necessary to pull from the repository to ensure the site is updated

**Acceptance Criteria:**

* Development branch of repository must have the latest commit of the VJF application

**User Story #3– Set up testing environment**

**Description:**

* The previous VJF group created Selenium IDE automated tests, these tests must be functional on my development environment

**Acceptance Criteria:**

* Selenium IDE is installed and operational
* Test suites can be loaded and run successfully with no errors

**User Story #4– Create feasibility study for OAuth Daemon**

**Description:**

* To aide in implementing single sign-on by the end of the semester, OAuth Daemon can be used to unify the API authentication for all Senior Project sites. Before it is implemented, a feasibility test is necessary to determine if it is a good decision.

**Acceptance Criteria:**

* Create document showing advantages and disadvantages of OAuth Daemon and current OAuth 2.0 authentication
* Select the best option based on the findings from the produced document

**User Story #5–** **Install OAuth Daemon on local virtual machine**

**Description:**

* Install OAuth Daemon instance on a local Ubuntu virtual machine to set it up and briefly test it before installing it on the VJF development virtual machine

**Acceptance Criteria:**

* OAuth Daemon instance is installed and running
* LinkedIn and Google connection tests on the server are successful

**User Story #6–** **Repair bug causing OAuth Daemon failure to start**

**Description:**

* A bug needs to be repaired, which is causing OAuth Daemon instance to fail when attempting to start it

**Acceptance Criteria:**

* System is troubleshooting is performed
* Bug is found and resolved

**User Story #7–** **Install OAuth Daemon instance on Virtual Job Fair**

**Description:**

* As a developer, I wish to modify VJF authentication classes so that API authentication can be routed through the OAuth Daemon server

**Acceptance Criteria:**

* LinkedIn and Google authentication classes on VJF authenticate user accounts through a connection with the OAuth Daemon server
* OAuth PHP SDK is used to write code which interacts with the OAuth Daemon instance
* User is able to log in to the system through LinkedIn or Google with no visible changes to the log in process

**User Story #8–** **Fix bug where OAuth Daemon’s Redis database randomly clears user account**

**Description:**

* Fix bug which is causing OAuth Daemon’s Redis database to randomly reset the main user account

**Acceptance Criteria:**

* System is troubleshooting is performed
* Redis database is updated to the latest stable release
* Bug is found and resolved

**User Story #9–** **Create Senior Project Portal as a subsystem of Virtual Job Fair**

**Description:**

* Develop a Senior Project Portal as a subsystem of the current Virtual Job Fair system.

**Acceptance Criteria:**

* Portal has its own login page
* Portal has add/update/delete site features for admin users
* Portal authenticates user’s with Google
* Portal implements single sign-on and links to the four other Senior Project sites

**User Story #10–** **Create Senior Project Portal as a subsystem of Virtual Job Fair**

**Description:**

* Develop a Senior Project Portal as a subsystem of the current Virtual Job Fair system.

**Acceptance Criteria:**

* Portal has its own login page
* Portal authenticates user’s with Google
* Portal implements single sign-on and links to the four other Senior Project sites

**User Story #11–** **Add system administrator functionality**

**Description:**

* Add administrator features to portal site so that the list of portal sites can be managed

**Acceptance Criteria:**

* Administrator can log into the system with an admin account
* Administrator can add/update/delete portal sites from the portal

**User Story #12–** **Create model for list of Portal Sites**

**Description:**

* Create a model and a new corresponding database table to store the current list of portal sites so that it can be loaded when a student actor logs into the portal

**Acceptance Criteria:**

* Portal Sites table is created in the database
* Corresponding model is created on Yii project

**User Story #13–** **Implement single sign-on with Collaborative Platform**

**Description:**

* Implement single sign-on with Collaborative Platform site by working with the Collaborative Platform lead developer

**Acceptance Criteria:**

* Controller actions are created on both VJF and CP which will act as endpoints for sending user credentials from VJF to CP
* Portal should be able to log a user into Collaborative Platform from the portal

**User Story #14–** **Create new web application for Senior Project Portal**

**Description:**

* Implement a unique web application only for Senior Project portal which will have nothing in common with the Virtual Job Fair system

**Acceptance Criteria:**

* New Yii application for the Senior Project Portal created
* Application is hosted on newly made apache server on the new virtual machine

**User Story #15–** **Establish separate database for Senior Project Portal site**

**Description:**

* Create a new phpmyadmin database on the new Senior Project Portal virtual machine

**Acceptance Criteria:**

* Phpmyadmin database is created on new virtual machine
* Necessary database tables for portal sites and users are created

**User Story #16–** **Implement FIU Gmail Login on Senior Project Portal site with OAuth D**

**Description:**

* Set up FIU Gmail Login with OAuth D for student actor authentication

**Acceptance Criteria:**

* After login credentials are entered on the login page by the student actor, they are directed to Google’s authentication page in order to authenticate the email address the entered on the login page

**User Story #17–** **Implement single sign-on for Collaborative Platform, Virtual Job Fair and Senior Project site**

**Description:**

* Implement single sign-on functionality for Collaborative Platform, Virtual Job Fair and Senior Project site

**Acceptance Criteria:**

* When a student actor logs in to the portal site successfully, and clicks a link to one of the Senior Project sites on the portal, they are redirected to the site which they chose and logged in using the credentials they entered when logging in to the portal

**User Story #18– Test Admin Features with Selenium IDE**

**Description:**

* Create Selenium test suite for Admin actor

**Acceptance Criteria:**

* Test cases should test all of the basic Admin functionality such as adding a site, deleting a site and updating a site

**User Story #19– Fix bug were browser doesn’t redirect after a portal site is added on admin dashboard**

**Description:**

* Fix bug which prevents the browser from redirecting after admin actor adds a new site but the browser does not redirect afterwards

**Acceptance Criteria:**

* System troubleshooting performed
* Redirect bug is fixed and browser redirects normally

## Pending User Stories

**User Story #20– Implement feature where single sign-on system prompts user for new credentials after sign on failure**

**Description:**

* Implement feature where single sign-on system prompts user for new credentials after sign on failure

**Acceptance Criteria:**

* If user credentials stored on portal site are not accepted by the site which the portal attempts to log in to, then the portal prompts the user to enter correct credentials, and if the credentials work then they are saved in the database

**User Story #21– Implement LinkedIn authentication on portal site**

**Description:**

* Implement LinkedIn authentication option. Student actor can optionally log in through LinkedIn instead of Gmail, but the system will still verify the account to ensure that it is an fiu.edu account

**Acceptance Criteria:**

* User is able to log in and access the portal by authenticating their LinkedIn account

# Project Plan

In this section we are going to introduce Senior Project Portal 1.0 project planning. First, we shall list all software and hardware requirements needed to develop the Senior Project Portal system. Then, we will provide a breakdown of all user stories, tasks, and mile stones created for each sprint of the project.

## Hardware and Software Resources

This subsection includes a comprehensive list of the hardware and software that was used to develop the system.

**Hardware**:

* Memory: 1 GB 133MHz of SDRAM or more.
* CPU Speed: Pentium IV 2.0 GHz processor or better.
* CPU Architecture: AMD & Intel 32 | 64-bit compatible
* Disk Space: 5 GB or more of HDD

**I/O Devices:**

* Compatible with PS2, USB wired | USB wireless keyboard and mouse device.
* VGA | HDMI | DVI | DS | capable monitor.

**Software:**

* PHP 5.5 or later: very popular server side scripting programming language used to develop dynamic web pages. PHP is free software widely used by developers for its stability, community support, security and cross-platform compatibility.
* MySQL 5.6.24: free database software used to control, manage, and store system data. MySQL is very stable, secure and scalable.
* PHPMyAdmin | MySQL Workbench 6.1: are two compatible DBMS to manage MySQL

database; these two software provide a friendly graphic user interface for managing MySQLdatabases, tables, configurations, etc.

* Apache Web Server 2.4 or later: reliable, stable and cross platform web server used to server web pages to all compatible client web browsers.
* WAMP/XAMP: software bundle that includes MySQL, Apache, PHP and Perl, that can be used to setup your local development environment.
* Yii PHP Framework 1.14: is an open-source Web application development framework written in PHP5. Used for a clear design and fast development specifically for MVC based web applications.
* Twitter Bootstrap: is one of the most popular frontend frameworks and open source projects for web development.
* Internet Explorer 10 or later | Mozilla Firefox | Chrome 43.0.2357 used for testing the site and also to test browser compatibility.
* NetBeans IDE 8.0.2: this is the development environment we used to develop the new features for the release 6.0.
* Google Docs: web based collaborative tool that allows multiple users to work on the same document. This tool provided us an amazing flexibility to work remotely on the documentation of this project and shared ideal with our team mates very quick.
* Github Collaborative Tool Platform: this is a web based collaborative tool that allows multiple developer to work in a project and track the changes made by each individual. The software allows you to check conflicts between team members code and allows you to roll back changes if necessary.
* Mingle Project Management: this is a web based project management application tool that allows you to keep track of the project
* Ganttproject 2.7: is a free open source software used to manage projects. It provides graphic representation of the project progress and cost. It is easy to use and very customizable in comparison to other similar software.
* Microsoft Visio Professional 2013: software provided by Microsoft featuring templates for architectural pattern, user stories, classes and sequence diagrams among many others functionalities in order to successfully document the software procedures.
* Selenium IDE Plugin 2.45.0: a Mozilla Firefox plugin to run the automated test cases.
* Skype: a software that will allows you to communicate with the project owners, team members and instructor. Also, very useful during the scrubbing session, since you can chat, share screen and documents.
* Virtual Box: virtualization software that will be used for the deployment of the software in a virtual environment.

## Feasibility Study

Before OAuth Daemon was implemented on Virtual Job Fair to replace the existing OAuth 2.0 authentication system for Google and LinkedIn, a feasibility study was performed during Sprint 2 in order to determine if OAuth Daemon’s advantages outweighed the disadvantages.

Comparison of Virtual Job Fair’s Authentication and OAuth Daemon Authentication

**Virtual Job Fair’s Current Authentication** **OAuth Daemon Authentication Solution**

* Open source
* Background API Server
* Allows developer to focus on product development instead of losing time on API integration through OAuth
* Authenticate and integrates many common API providers (LinkedIn, Facebook, Twitter…)
* Minimal lines of code and quick installation
* VJF will not be responsible for initiation of authentication, token requests, token refreshes, API interactions
* Will make it possible to combine the login of VJF and other sites by allowing OAuth D to handle all token and authentication work
* OAuth 2.0
* Individual authentication classes
  + Google
  + LinkedIn
  + FIU CS
* Different tokens required for each service
* VJF application handles the initiation of authentication, token requests, token refreshes, API interactions and any other tasks

Overall, the OAuth Daemon approach appears to be the best approach for integrating the authentication/login of multiple sites. It simplifies the process and will involve far less code and development time.

## 

## Sprints Plan

For each sprint, list the user stories selected for implementation in descending order of priority.

### Sprint 1

**User Story #1 – Prepare Development Environment**

**Description:**

* The development environment needs to be set up and configured on my machine in order to begin development tasks for the project

**Acceptance Criteria:**

* WAMP server installed/configured
* VJF site is hosted locally and functional
* DBMS is configured

**User Story #2 – Pull latest version of Github repository onto development site**

**Description:**

* The current version of the VJF application on the development virtual machine must be the most recent version of the project, so it is necessary to pull from the repository to ensure the site is updated

**Acceptance Criteria:**

* Development branch of repository must have the latest commit of the VJF application

**User Story #3– Set up testing environment**

**Description:**

* The previous VJF group created Selenium IDE automated tests, these tests must be functional on my development environment

**Acceptance Criteria:**

* Selenium IDE is installed and operational
* Test suites can be loaded and run successfully with no errors

### Sprint 2

**User Story #4– Create feasibility study for OAuth Daemon**

**Description:**

* To aide in implementing single sign-on by the end of the semester, OAuth Daemon can be used to unify the API authentication for all Senior Project sites. Before it is implemented, a feasibility test is necessary to determine if it is a good decision.

**Acceptance Criteria:**

* Create document showing advantages and disadvantages of OAuth Daemon and current OAuth 2.0 authentication
* Select the best option based on the findings from the produced document

### Sprint 3

**User Story #5–** **Install OAuth Daemon on local virtual machine**

**Description:**

* Install OAuth Daemon instance on a local Ubuntu virtual machine to set it up and briefly test it before installing it on the VJF development virtual machine

**Acceptance Criteria:**

* OAuth Daemon instance is installed and running
* LinkedIn and Google connection tests on the server are successful

**User Story #6–** **Repair bug causing OAuth Daemon failure to start**

**Description:**

* A bug needs to be repaired, which is causing OAuth Daemon instance to fail when attempting to start it

**Acceptance Criteria:**

* System is troubleshooting is performed
* Bug is found and resolved

### Sprint 4

**User Story #7–** **Install OAuth Daemon instance on Virtual Job Fair**

**Description:**

* As a developer, I wish to modify VJF authentication classes so that API authentication can be routed through the OAuth Daemon server

**Acceptance Criteria:**

* LinkedIn and Google authentication classes on VJF authenticate user accounts through a connection with the OAuth Daemon server
* OAuth PHP SDK is used to write code which interacts with the OAuth Daemon instance
* User is able to log in to the system through LinkedIn or Google with no visible changes to the log in process

### Sprint 5

**User Story #8–** **Fix bug where OAuth Daemon’s Redis database randomly clears user account**

**Description:**

* Fix bug which is causing OAuth Daemon’s Redis database to randomly reset the main user account

**Acceptance Criteria:**

* System is troubleshooting is performed
* Redis database is updated to the latest stable release
* Bug is found and resolved

**User Story #9–** **Create Senior Project Portal as a subsystem of Virtual Job Fair**

**Description:**

* Develop a Senior Project Portal as a subsystem of the current Virtual Job Fair system.

**Acceptance Criteria:**

* Portal has its own login page
* Portal has add/update/delete site features for admin users
* Portal authenticates user’s with Google
* Portal implements single sign-on and links to the four other Senior Project sites

### Sprint 6

**User Story #10–** **Create Senior Project Portal as a subsystem of Virtual Job Fair**

**Description:**

* Develop a Senior Project Portal as a subsystem of the current Virtual Job Fair system.

**Acceptance Criteria:**

* Portal has its own login page
* Portal authenticates user’s with Google
* Portal implements single sign-on and links to the four other Senior Project sites

**User Story #11–** **Add system administrator functionality**

**Description:**

* Add administrator features to portal site so that the list of portal sites can be managed

**Acceptance Criteria:**

* Administrator can log into the system with an admin account
* Administrator can add/update/delete portal sites from the portal

**User Story #12–** **Create model for list of Portal Sites**

**Description:**

* Create a model and a new corresponding database table to store the current list of portal sites so that it can be loaded when a student actor logs into the portal

**Acceptance Criteria:**

* Portal Sites table is created in the database
* Corresponding model is created on Yii project

**User Story #13–** **Implement single sign-on with Collaborative Platform**

**Description:**

* Implement single sign-on with Collaborative Platform site by working with the Collaborative Platform lead developer

**Acceptance Criteria:**

* Controller actions are created on both VJF and CP which will act as endpoints for sending user credentials from VJF to CP
* Portal should be able to log a user into Collaborative Platform from the portal

### Sprint 7

**User Story #14–** **Create new web application for Senior Project Portal**

**Description:**

* Implement a unique web application only for Senior Project portal which will have nothing in common with the Virtual Job Fair system

**Acceptance Criteria:**

* New Yii application for the Senior Project Portal created
* Application is hosted on newly made apache server on the new virtual machine

**User Story #15–** **Establish separate database for Senior Project Portal site**

**Description:**

* Create a new phpmyadmin database on the new Senior Project Portal virtual machine

**Acceptance Criteria:**

* Phpmyadmin database is created on new virtual machine
* Necessary database tables for portal sites and users are created

**User Story #16–** **Implement FIU Gmail Login on Senior Project Portal site with OAuth D**

**Description:**

* Set up FIU Gmail Login with OAuth D for student actor authentication

**Acceptance Criteria:**

* After login credentials are entered on the login page by the student actor, they are directed to Google’s authentication page in order to authenticate the email address the entered on the login page

**User Story #17–** **Implement single sign-on for Collaborative Platform, Virtual Job Fair and Senior Project site**

**Description:**

* Implement single sign-on functionality for Collaborative Platform, Virtual Job Fair and Senior Project site

**Acceptance Criteria:**

* When a student actor logs in to the portal site successfully, and clicks a link to one of the Senior Project sites on the portal, they are redirected to the site which they chose and logged in using the credentials they entered when logging in to the portal

**User Story #18– Test Admin Features with Selenium IDE**

**Description:**

* Create Selenium test suite for Admin actor

**Acceptance Criteria:**

* Test cases should test all of the basic Admin functionality such as adding a site, deleting a site and updating a site

**User Story #19– Fix bug were browser doesn’t redirect after a portal site is added on admin dashboard**

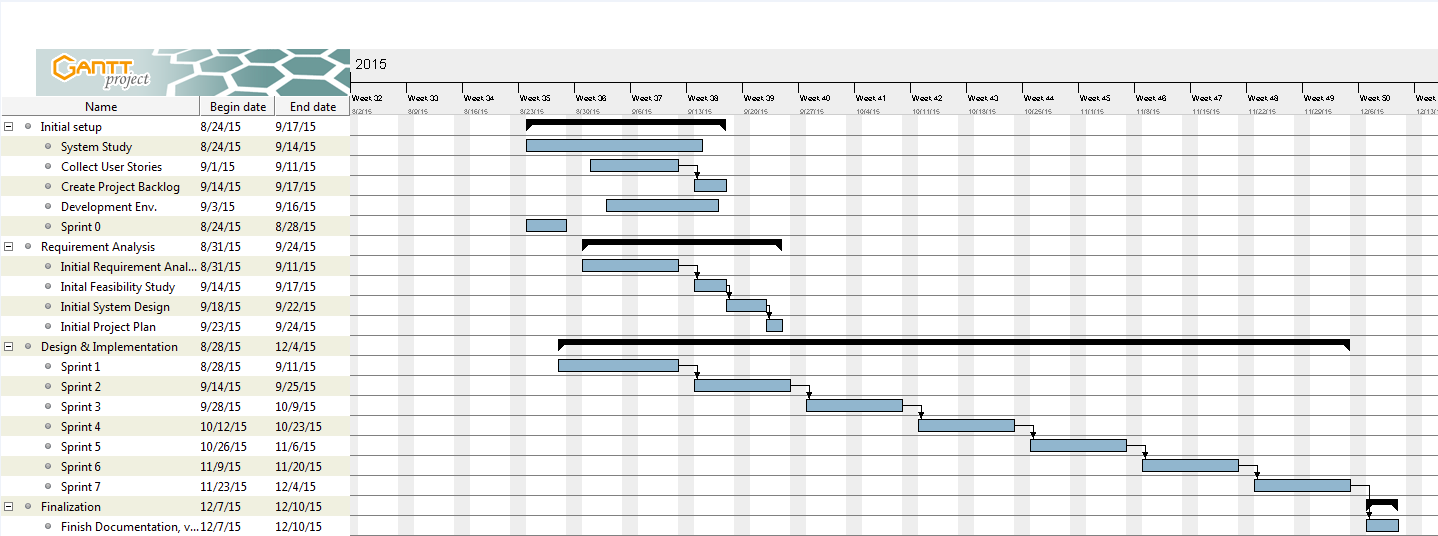
**Description:**

* Fix bug which prevents the browser from redirecting after admin actor adds a new site but the browser does not redirect afterwards

**Acceptance Criteria:**

* System troubleshooting performed
* Redirect bug is fixed and browser redirects normally

# GANTT CHART



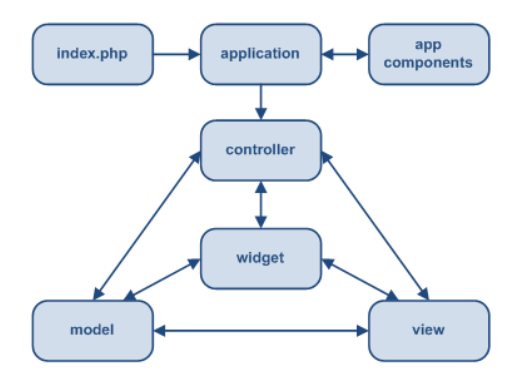
# 

# System Design

This chapter includes essential information about the system design. That is, it contains information about architectural patterns that are being used in the system, information about the system and subsystem decomposition, the deployment diagram, and the design patterns that were used to solve problems. This is fundamental chapter that provides very technical information that is required for further software engineering of the Senior Project Portal system.

## Architectural Patterns

This new implementation of Senior Project Portal was developed using the Model View Controller (MVC) software architectural pattern in the Yii Framework. It is a rich, reusable, and well structure software. The benefits of MVC in regards to this software solution are that it separates logic controllers and models from graphical user interfaces, so that developers can easily change each part of the system without affecting the other. For example, changing the way a button looks on a view will not affect the logical functionalities of that button in the controller nor will it affect a model related to that logic. The following diagram depicts the architecture in the Yii framework:



## System and Subsystem Decomposition

Currently the system is composed of two subsystems, each with its own unique purpose, each making the Senior Project Portal website richer in content. We shall describe the newly added subsystems, which are the Site subsystem and the Admin subsystem.

**Site Subsystem**

This subsystem is responsible for handling security, login validation, the navigation bar, and the index of the website. Its main goal is to handle the initial logic once the user arrives at the index, to determine if the user has valid credentials, and where in the system is the validate user should be directed to. It is also responsible for the creation of new user accounts, which is linked to login validation for student users (whenever a student user attempts to log in, a new user is created by default, or if an existing user exists, they are replaced, for record keeping purposes). Basically the site subsystem controls all interactions for student users.

**Admin Subsystem**

This subsystem is responsible for handling all system administrator interactions with the system. After an admin user is authenticated by the site subsystem, they are directed to the admin home page on the admin subsystem where they can perform the basic admin features (adding a portal site, deleting a portal site and updating a portal site.

## Deployment Diagram

The previous team set up a Scientific Linux, Apache, MySQL, and PHP (LAMP) server and the Yii framework on the FIU SCIS network. It is also important to note that we are still utilizing the Model View Controller architecture pattern. And the previous team developed a deployment diagram which indicated the mapping between hard and software. Therefore, we can reuse the same concept to generate our own diagram, the previous concept remains the same, only the graphical representation of the diagram has been updated, see (Figure DD001).

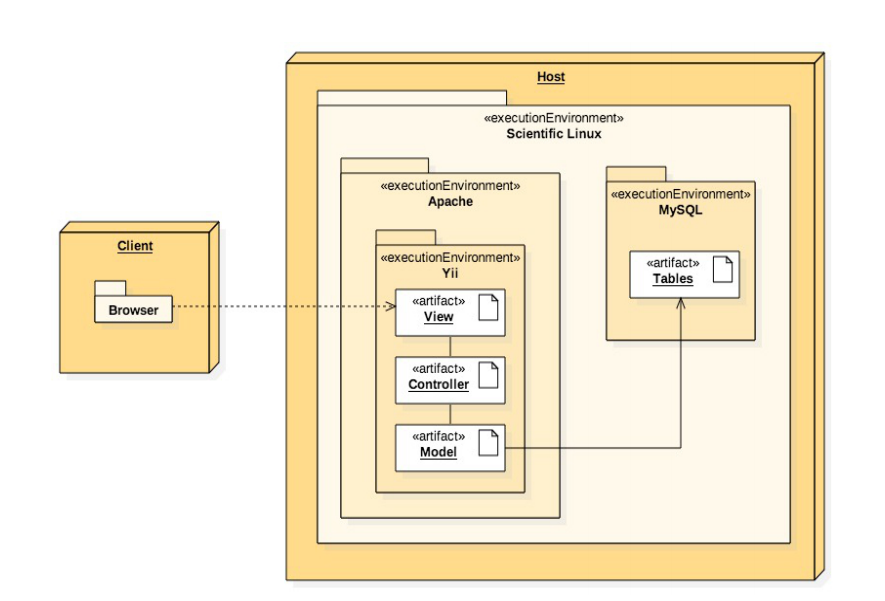


Figure DD-001 – Deployment Diagram

# System Validation

This chapter introduces the system validation of SPP project release 1.0. In this section we will provide a list of all test cases implemented during the semester. Furthermore, we will show with a brief description of each test purpose.

**User Story # 18 - Test Admin Features with Selenium IDE**

Subsystem Tests

* 001 Site Delete – Tests logging in as an admin actor and deleting a specific site
* 002 Site Add – Tests logging in as an admin actor adding a specific site to the portal
* 003 Site Update – Tests logging in as an admin actor and updating the name of a specific portal site
* 004 Login – Tests logging in to the portal as an admin actor
* 005 Login Failure – Tests a login failure event when attempting to log in to the portal as an admin actor

# Glossary

# Appendix

## Appendix A - UML Diagrams

### Static UML Diagrams

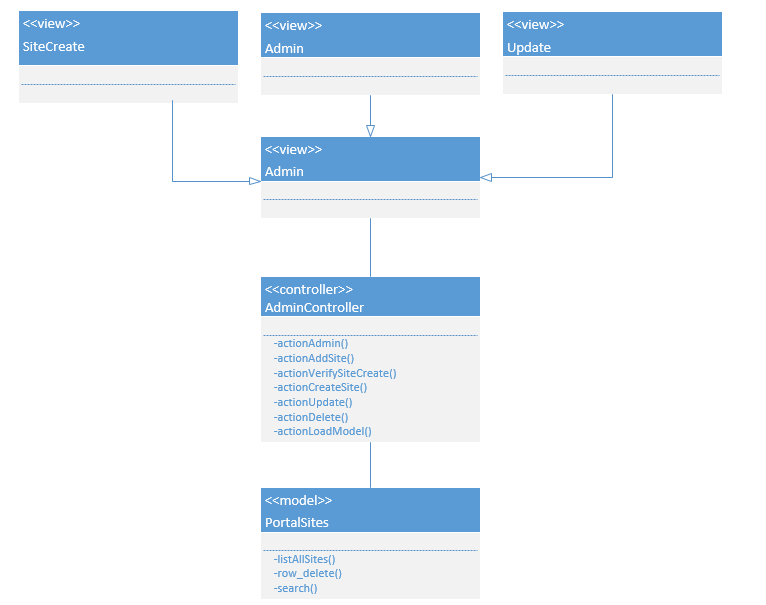


Figure S-001 - Admin Dashboard Class Diagram

### 

Figure S-002 – User Login Class Diagram

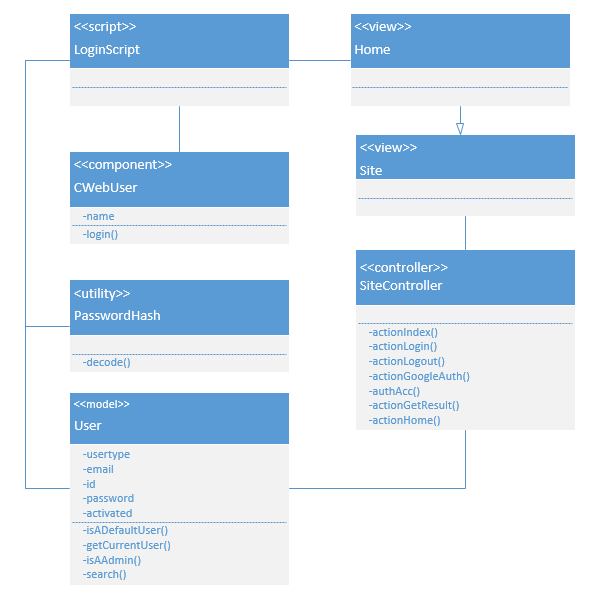


Figure S-003 Single Sign-On Class Diagram

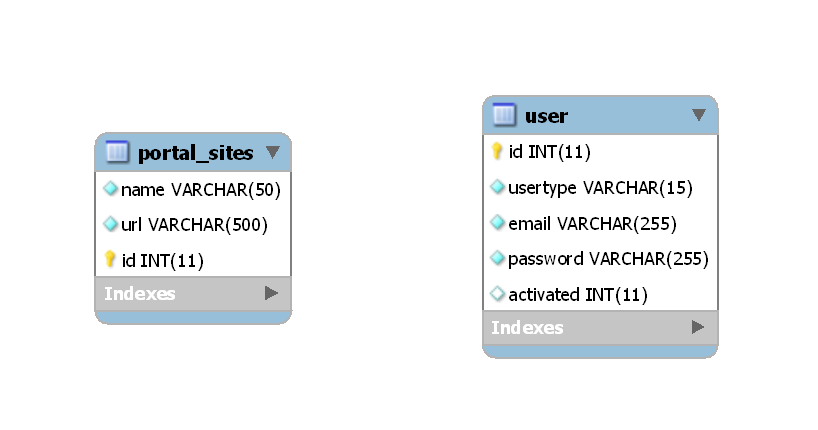


Figure S-004 – ERD of the System

### Dynamic UML Diagrams

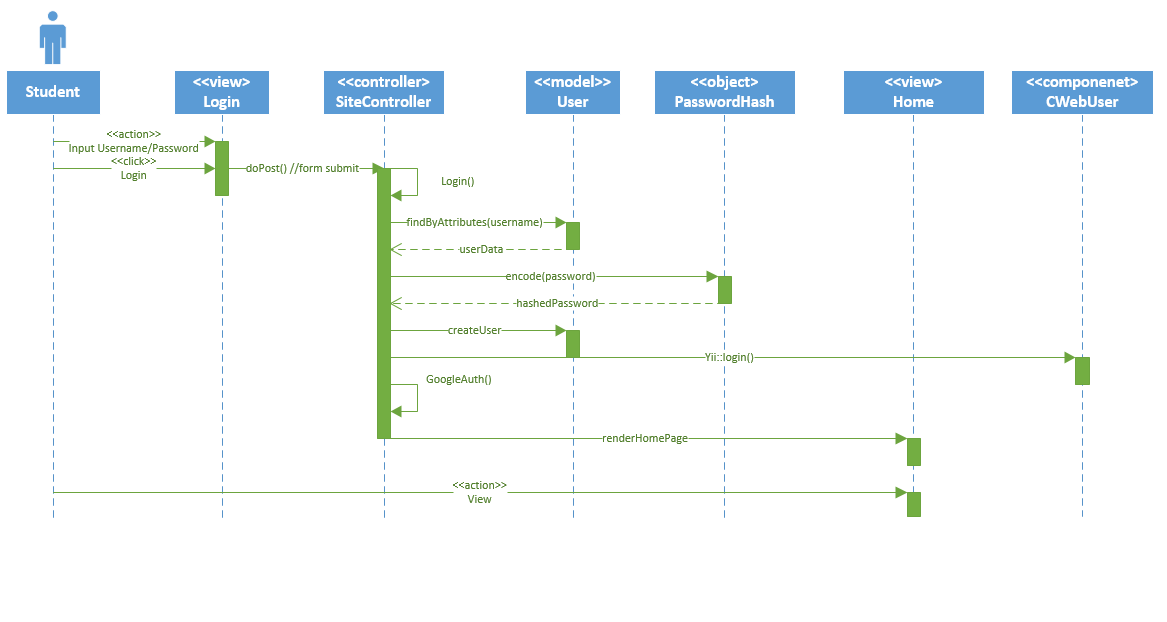


Figure D-001 – Student Actor Login

## 

Figure D-002 – Admin Actor Login

## 

Figure D-003 – Log Out

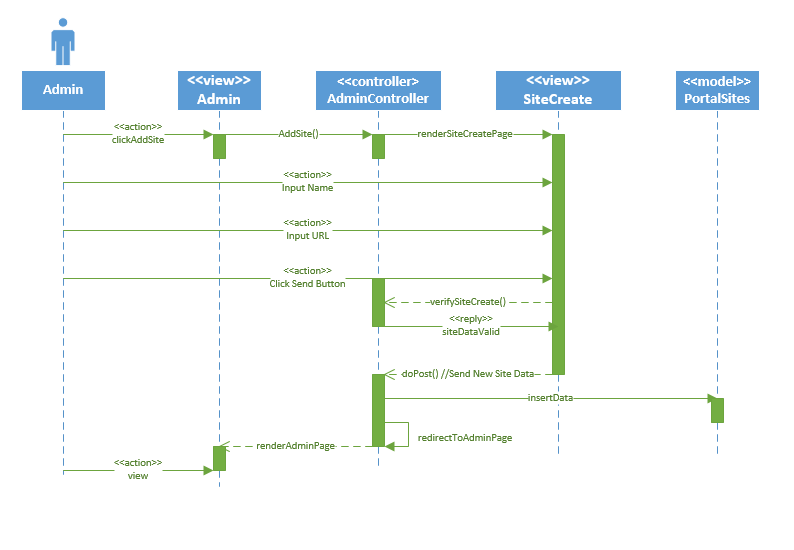


Figure D-004 – Add Site

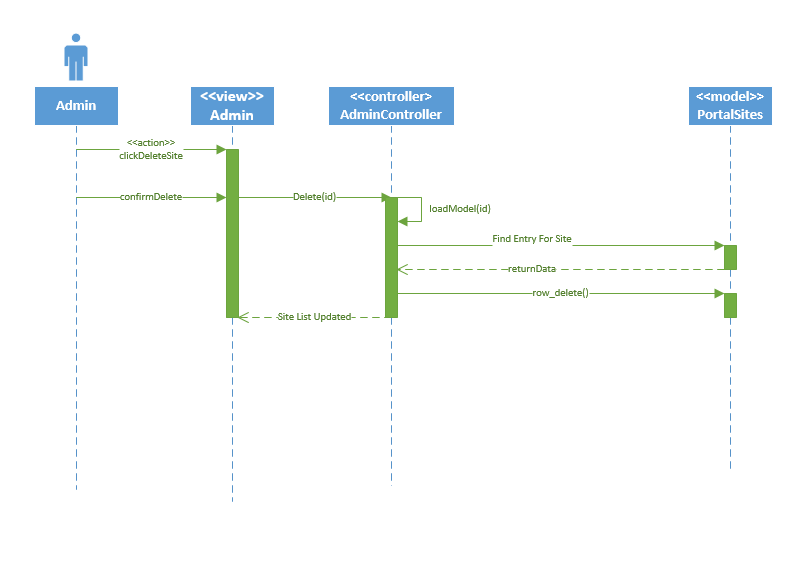


Figure D-005 – Delete Site

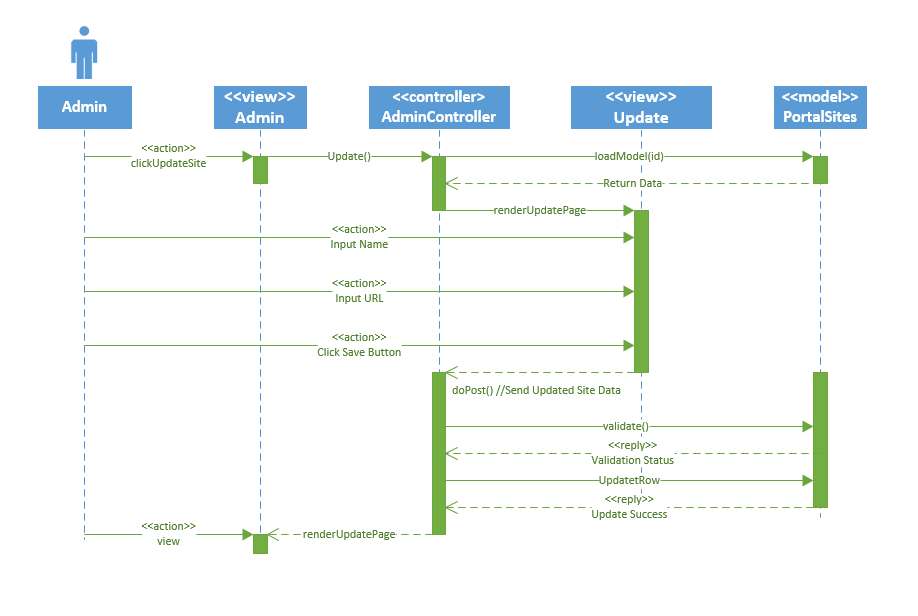


Figure D-006 Update Site

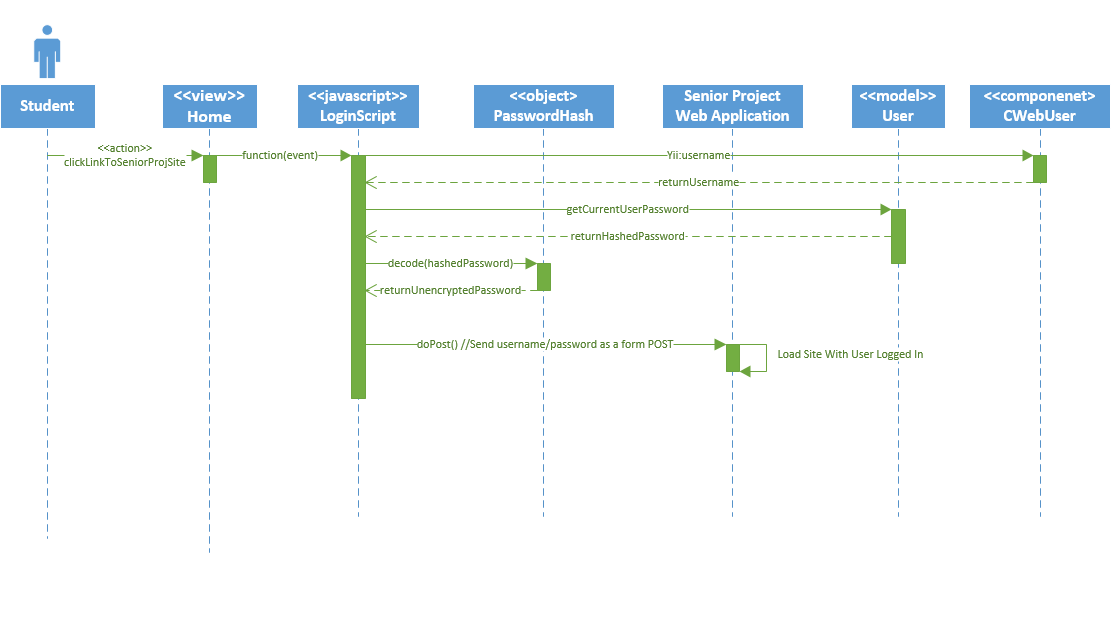


Figure D-007 Sign In to Senior Project Sites

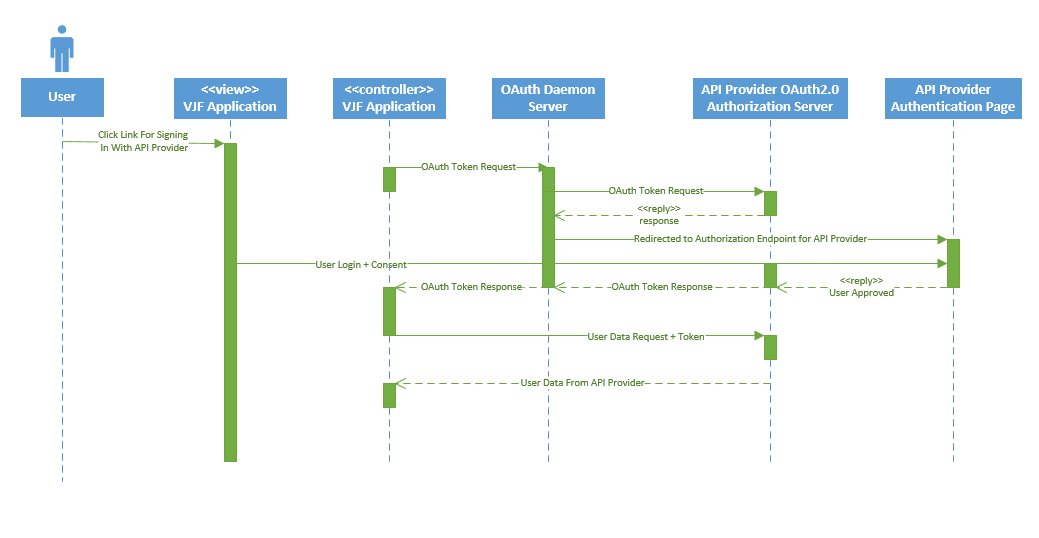


Figure D-008 Authenticate Account on VJF Using OAuth Daemon

## Appendix B - User Interface Design

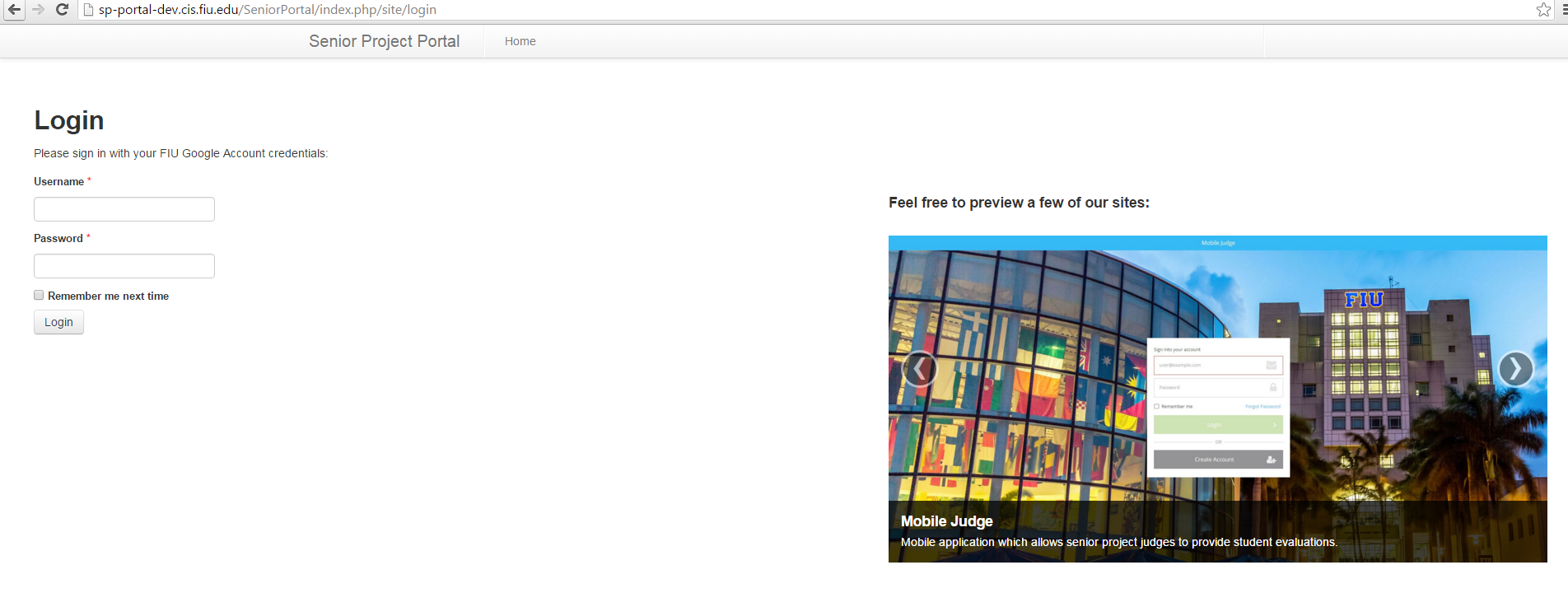


Figure UI-001 Login Page

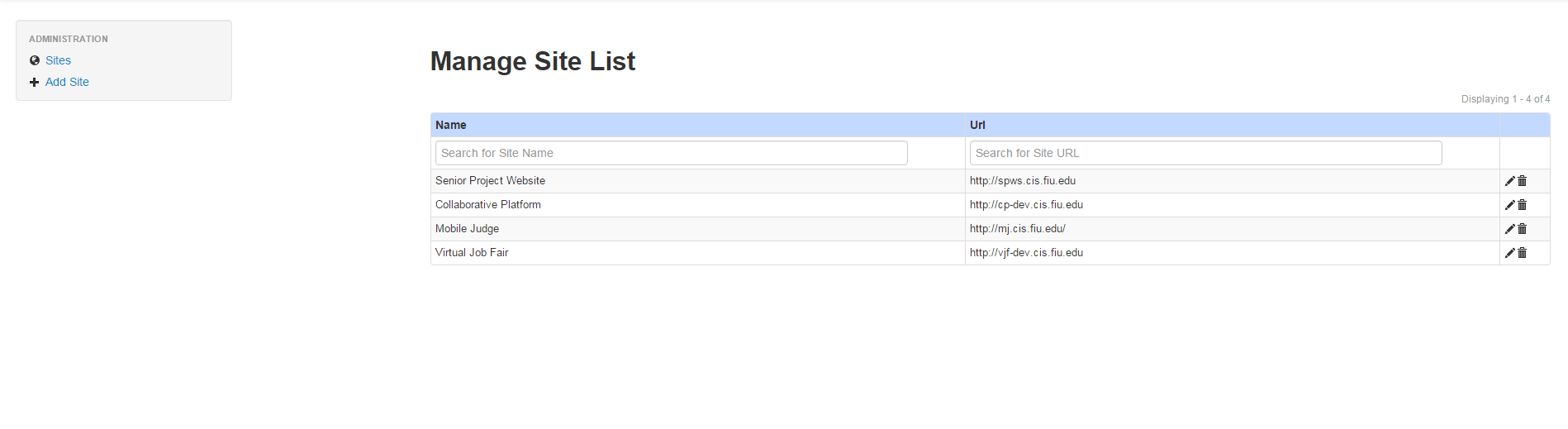


Figure UI-002 Admin Home

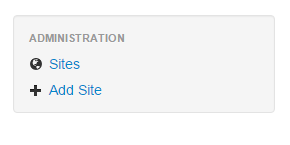


Figure UI-003 Admin Sidebar

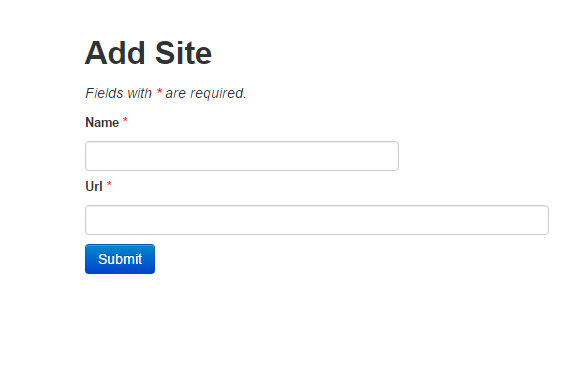


Figure UI-004 Add Site

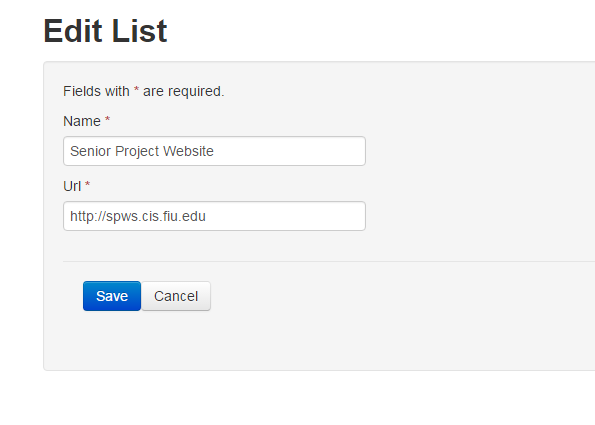


Figure UI-005 Update Site

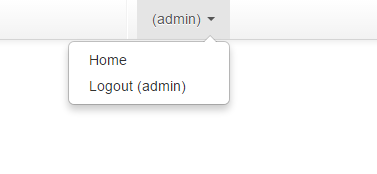


Figure UI-006 Pull Down Menu

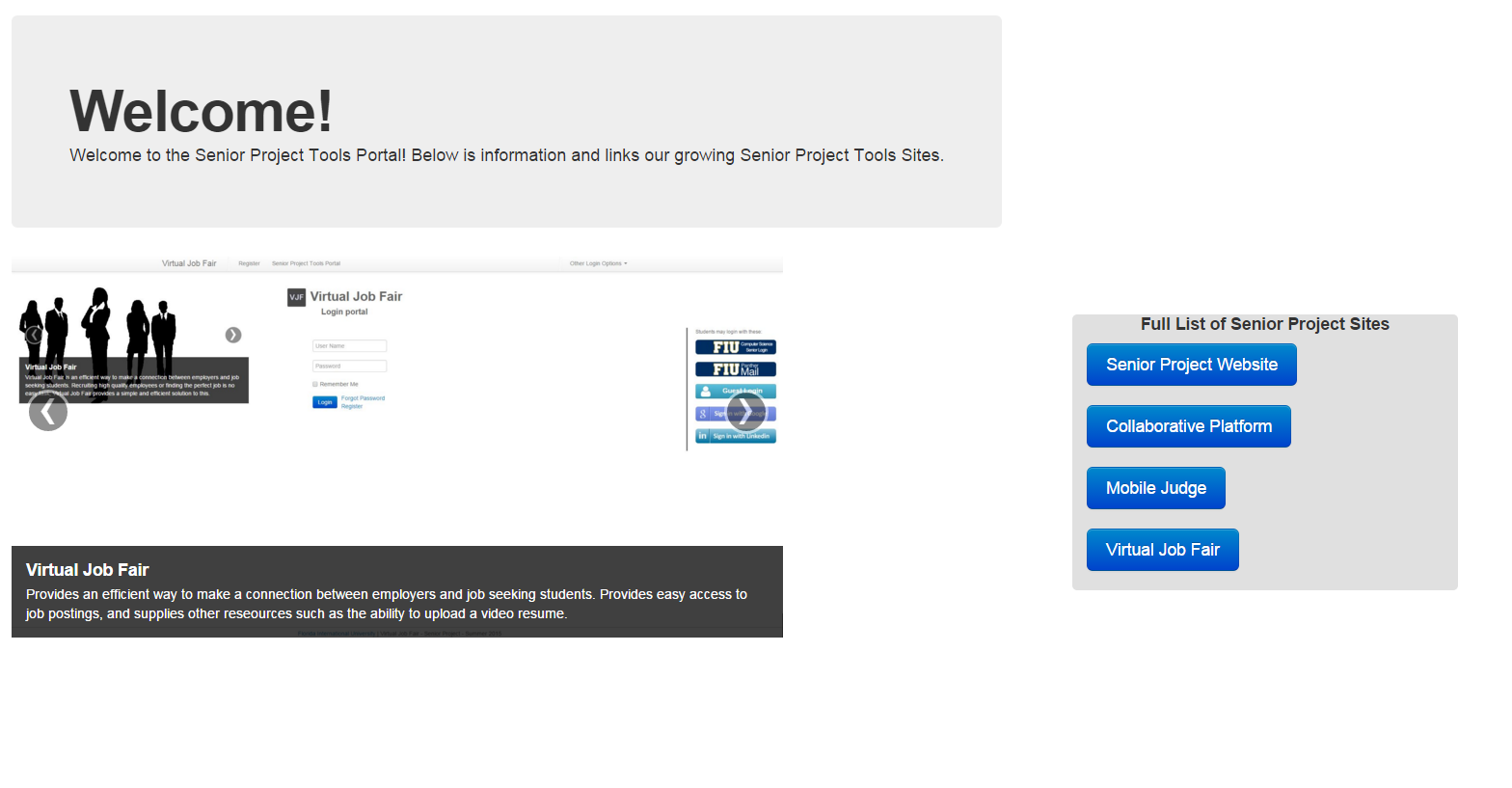


Figure UI-007 Student Home Page

## Appendix C - Sprint Review Reports

**Sprint 1:**

**Date:** September 11, 2015

**Attendees:** Christopher Jones, Ricardo Dominguez, Masoud Sadjadi, Frank Hernandez, Anthony Amador

**Discussed Topics:**

The initial goals were: to learn about the existing system which I would be working with, Virtual Job Fair, and also setting up the development environment to prepare for my development tasks during the rest of the semester. At the end of the sprint, the entire development environment was successfully set up and a fair amount of research was done, relating to the Yii Framework which the site was running off of. The product owner was fine with what was accomplished during this sprint and assigned the task for Sprint 2, which involved research on finding a way to better manage API authentication for Virtual Job Fair.

**Sprint 2 Report**

**Date:** September 25, 2015

**Attendees:** Christopher Jones, Ricardo Dominguez, Masoud Sadjadi, Frank Hernandez, Anthony Amador

**Discussed Topics:**

The initial goals were: determining the feasibility of using OAuth Daemon as an API authentication tool for use with integrating the login of the Senior Project sites. OAuth Daemon was compared with the usage of OAuth 2.0 and a list of advantages/disadvantages was compiled. After researching for a few days, it was determined that OAuth Daemon was the better approach for integrating API because it handles the majority of the API work and minimal code is necessary on the web application which utilizes it (such as Virtual Job Fair). The product owner was fine with my findings and gave the approval to start working on created an OAuth Daemon implementation during Sprint 3.

**Sprint 3 Report**

**Date:** October 9, 2015

**Attendees:** Christopher Jones, Ricardo Dominguez, Frank Hernandez, Anthony Amador

**Discussed Topics:**

The initial goals were: hosting an OAuth Daemon server on the development site for Virtual Job Fair. I was unable to get the OAuth Daemon server to operate on my local virtual machine because of several dependency related errors and other errors. A majority of the sprint was spent on troubleshooting the issues I was facing with the server. This incomplete work was moved to Sprint 4 with a much higher priority, since it was the second sprint in a row where I was delayed by this task. The product owner was not present during this entire sprint, so I was unable to update him on my progress.

**Sprint 4 Report**

**Date:** October 23, 2015

**Attendees:** Christopher Jones, Ricardo Dominguez, Masoud Sadjadi, Frank Hernandez, Anthony Amador

**Discussed Topics:**

The initial goals were: to host an OAuth Daemon server on the development site for Virtual Job Fair and have Virtual Job Fair’s Google and LinkedIn authentication systems modified to operate through the OAuth Daemon server. I was able to successfully setup OAuth D on a local virtual machine and test it on my own home network. Afterwards, a small test site (Yii application) was created with the purpose of testing basic OAuth Daemon features and learning the PHP SDK for OAuth D. The product owner asked for faster progress on the implementation of OAuth Daemon with Virtual Job Fair so that implementation of other features could begin.

**Sprint 5 Report**

**Date:** November 6, 2015

**Attendees:** Christopher Jones, Ricardo Dominguez, Masoud Sadjadi, Anthony Amador

**Discussed Topics:**

The initial goals were: to host an OAuth Daemon server on the development site for Virtual Job Fair and have Virtual Job Fair’s Google and LinkedIn authentication systems modified to operate through the OAuth Daemon server. Also the product owner asked for the Senior Project Portal site to be created. All OAuth Daemon related tasks were completed and Google/LinkedIn authentication for Virtual Job Fair were fully functional through OAuth Daemon. However, due to disagreements between myself and the rest of the group, we were unable to decide on an implementation strategy for the portal and it was unable to be created during this sprint and a new plan for implementation was created during the sprint review.

**Sprint 6 Report**

**Date:** November 20, 2015

**Attendees:** Christopher Jones, Ricardo Dominguez, Masoud Sadjadi, Frank Hernandez

**Discussed Topics:**

The initial goals were: to create the Senior Project Portal, create an administrator account with the ability to manage (delete, edit, add etc.) a list of Senior Project sites to be displayed on the portal and implement single sign-on through the portal so that when a user logs in to the portal, their credentials are saved and used when the portal attempts to log a user into other Senior Project sites. Most of these features were implemented, but the product owner asked for certain things to be done differently, such as having the portal utilize a separate database from the Virtual Job Fair database which it was currently using, along with hosting the portal on its own separate server on a separate web application from Virtual Job Fair. Also the product owner wanted single-sign on implemented in a way where the portal has no interactions with the business logic of the sites which it attempts to log into.

**Sprint 7 Report**

**Date:** December 4, 2015

**Attendees:** Christopher Jones, Ricardo Dominguez, Masoud Sadjadi, Frank Hernandez

**Discussed Topics:**

The initial goals were: to establish the Senior Project Portal as a stand-alone web application hosted separately from Virtual Job Fair. SPP needed to be hosted from a separate web address, operate with its own unique database, use its own user accounts and use code which is completely independent of Virtual Job Fair. Also a few features had to be implemented/improved, such as Google authentication for students. Also single sign-on for the Senior Project Sites needed to be implemented in a manner where the portal has no interactions with the business logic of the sites which it attempts to log into. At the end of this sprint, all of these requirements and features were implemented and a demo of the system was performed during the review. The product owner was happy and acknowledged that all of the features were included in the release.

## Appendix D - Sprint Retrospective Reports

**Sprint 1 Retrospective**

**Date:** September 11, 2015

**Attendees:** Christopher Jones, Ricardo Dominguez, Masoud Sadjadi, Frank Hernandez, Anthony Amador

**Discussed Topics:**

During our retrospective meeting, we agreed that we all made good progress by setting up our development environments. All of us but Anthony were able to have our environments up and running and out locally hosted sites to work properly. Anthony and Dr. Sadjadi made it a point to get his environment running by the end of the weekend after the end of this current sprint.

**Sprint 2 Retrospective**

**Date:** September 25, 2015

**Attendees:** Christopher Jones, Ricardo Dominguez, Masoud Sadjadi, Frank Hernandez, Anthony Amador

**Discussed Topics:**

We discussed that Ricardo had an issue with his GitHub repository. All of us tried giving our input on how to solve the problem, but we realized it was an issue that had to be resolved by the FIU School of Computing and Information Sciences. Otherwise, our individual tasks proceeded smoothly and out progress was well received by Dr. Sadjadi.

**Sprint 3 Retrospective**

**Date:** October 9, 2015

**Attendees:** Christopher Jones, Ricardo Dominguez, Masoud Sadjadi, Frank Hernandez, Anthony Amador

**Discussed Topics:**

During our retrospective meeting, the work of all other group members besides myself progressed normally. I spent the majority of the sprint fixing issues I had with setting up an OAuth Daemon server instance on a local virtual machine. I spent over 20 hours researching the numerous errors and issues I had and 15 – 20 hours doing troubleshooting on my own. Occasionally I asked Frank and Ricardo for any helpful advice that they might have. Dr. Sadjadi was not present for this sprint, so I was unable to discuss my issues with him until a few days after the end of this sprint.

**Sprint 4 Retrospective**

**Date:** October 23, 2015

**Attendees:** Christopher Jones, Ricardo Dominguez, Masoud Sadjadi, Frank Hernandez, Anthony Amador

**Discussed Topics:**

During our retrospective meeting, I mentioned that I continued to have issues with successfully setting up an OAuth Daemon instance. Even after exhausting every option for finding assistance (peers, StackOverflow.com, README, skimming code etc.). I was able to complete the majority of the steps for the installation but I continued to have issues with getting the server to start. Eventually, after repeating steps several times in different ways I was able to get an OAuth Daemon server instance running on my local virtual machine. However, I only had a few days left until the end of the sprint, which only left me with enough time to test OAuth Daemon with a small site. Dr. Sadjadi stressed that this was a huge setback and future issues should not halt progress for such a long period of time; I agreed and prepared to put forth more effort in the remaining sprints.

**Sprint 5 Retrospective**

**Date:** November 6, 2015

**Attendees:** Christopher Jones, Ricardo Dominguez, Masoud Sadjadi, Frank Hernandez, Anthony Amador

**Discussed Topics:**

It was discussed that overall we realized we were each making progress on our respective tasks. Each of us were working diligently yet efficiently. No major issues occurred. Any minor issue we discussed during daily scrum meetings and we were able to resolve fairly quickly. I was able to implement OAuth Daemon on Virtual Job Fair but had a small setback when as a team we couldn’t reach an agreement on how to combine the login functionality of our four separate Senior Project sites. We resolved this during our sprint review, along with Dr. Sadjadi.

**Sprint 6 Retrospective**

**Date:** November 20, 2015

**Attendees:** Christopher Jones, Ricardo Dominguez, Masoud Sadjadi, Frank Hernandez

**Discussed Topics:**

After discussion, we noticed that we were all occupied with our individual tasks for the most part. All of us did our work separately and none of us brought up major issues during daily scrum meetings. Ricardo and I worked together for a few days on creating a method for passing user login credentials from my site to his site, which involved the creation of controller action endpoints on my site and his site. We were able to get that working successfully, but Dr. Sadjadi mentioned a few issues with that method, among other things (detailed in Sprint 6 Review). Overall I was proud of my progress and I acknowledged that Dr. Sadjadi’s requested changes needed to be heavily focused on.

**Sprint 7 Retrospective**

**Date:** December 4, 2015

**Attendees:** Christopher Jones, Ricardo Dominguez, Masoud Sadjadi, Frank Hernandez

**Discussed Topics:**

It was discussed that as a group, our individual work was complete and the majority of the features requested by the product owner were implemented successfully. We discussed that we must now focus on our project presentation, posters and videos for future developers.

# 

# References

Virtual Job Fair Ver6.0 <https://github.com/FIU-SCIS-Senior-Projects/Virtual-Job-Fair>