*Florida International University*

*School of Computing and Information Sciences*

Software Engineering Focus

Final Deliverable

Project Title:

ADVISOR AVAILABILITY NOTIFICATION SYSTEM

**Team Members:**

Jean Tovar

Hector Borges

**Product Owner(s)**:

Mario Sanchez

**Mentor(s)**:

Mario Sanchez,Mohsen Taheri

**Instructor**: Masoud Sadjadi

The MIT License (MIT)

Copyright (c) *2016 Florida International University*

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

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

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

***Abstract***

*This document presents the information necessary to gain a good understanding of the Advisor Availability System.The Advisor Availability system is a website that helps advisors get notified by the front desk when a student is on the way to their office and the advisor is also able to notify front desk when they are able to receive students.Within the next pages, information such as diagrams , use cases and user stories will be presented to describe the project in detail.*

**Table of Contents**

**Introduction** ……………………………………………………………………………………………………………………………….. 5

Current System ………………………………………………………………………………………………………………………... 5

Purpose of New System ……………………………………………………………………………………………………………... 5

**User Stories**

Implemented User Stories …………………………………………………………………………………………………………..6

Pending User Stories …………………………………………………………………………………………………………..….. 10

**Project Plan**

Hardware and Software Resources ………………………………………………………………………………………….… 12

Sprints Plan ……………………………………………………………………………………………………………………... 13

*Sprint 1*  …………………………………………………………………………………………………………………………... 13

*Sprint 2*  …………………………………………………………………………………………………………………………... 14

*Sprint 3*  …………………………………………………………………………………………………………………………... 15

*Sprint 4*  …………………………………………………………………………………………………………………………... 16

*Sprint 5*  …………………………………………………………………………………………………………………………... 17

*Sprint 6*  …………………………………………………………………………………………………………………………... 18

*Sprint 7*  …………………………………………………………………………………………………………………………... 18

**System Design**

Architectural Patterns ………………………………………………………………………………………………………….. 19

System and Subsystem Decomposition ……………………………………………………………………………………….… 20

Deployment Diagram ………………………………………………………………………………………………………….…... 21

**System Validation**  …………………………………………………………………………………………………………………….22

**Glossary**  ………………………………………………………………………………………………………………………………….26

**Appendix**  ………………………………………………………………………………………………………………………………….27

Appendix A - UML Diagrams ……………………………………………………………………………………………………. 27

*Static UML Diagrams*  ……………………………………………………………………………………………………….27

*Dynamic UML Diagrams*  …………………………………………………………………………………………………..28

Appendix B - User Interface Design ……………………………………………………………………………………….…... 39

Appendix C - Sprint Review Reports ……………………………………………………………………………………...…… 41

Appendix D - User Manuals, Installation/Maintenance Document, Shortcomings/Wishlist Document and other documents …………………………………………………………………….…………… 45

**References** ……………………………………………………………………………………………………………………...………...49

# Introduction

Communication between the front desk and advisors at their office is very inefficient currently. The front desk attendant has to walk over to each advisor in order to alert them of an incoming student. This is very time consuming, especially if the number of advisors increases.

## Current System

The current system is not digital, so the only way to notify advisors is by phone or by walking to the office of each advisor.

## Purpose of New System

Create a website in conjunction with a database to store information about an advisor's current status of availability and for the front desk to be able to view such information and send students from a queue into the advisor's office.

# User Stories

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

## Implemented User Stories:

# [#130](https://fiu-scis-seniorproject.mingle.thoughtworks.com/projects/advisor_availability_notificat/cards/130) Front desk Login

### Description:

* As the front desk i want to log in to view the appropriate front desk information.

### Acceptance Criteria:

1. Front desk must have a login account
2. Table of advisors must be displayed in front desk

**Modeling:** Refer to UML diagrams

# [#128](https://fiu-scis-seniorproject.mingle.thoughtworks.com/projects/advisor_availability_notificat/cards/128) Multiple user Login

### Description:

* As a user i want to be able to log in with different advisor accounts

### Acceptance Criteria:

1. Multiply advisors must be able to log in simultaneously and post their statuses.
2. All users must also be able to log out
3. Users must be able to change their own passwords.

**Modeling:** Refer to UML diagrams

# [#123](https://fiu-scis-seniorproject.mingle.thoughtworks.com/projects/advisor_availability_notificat/cards/123) Advisor signals front desk

### Description:

* As an Advisor i would like to signal signal front desk of my status.

### Acceptance Criteria:

1. Front desk must display the status of the advisor.
2. Advisor must change statuses when desired.

**Modeling:** Refer to UML diagrams

# [#127](https://fiu-scis-seniorproject.mingle.thoughtworks.com/projects/advisor_availability_notificat/cards/127) Notification alert

### Description:

* Create a notification alert so show availability of advisor

### Acceptance Criteria:

1. Front Desk is Notified when an advisor changes status to available

**Modeling:** Refer to UML diagrams

# [#127](https://fiu-scis-seniorproject.mingle.thoughtworks.com/projects/advisor_availability_notificat/cards/127) Graphical User Interface

### Description:

* Creating of the Graphical User interface using boostrap and HTML code

### Acceptance Criteria:

1. Create visually appealing Login interface for users
2. Visually appealing Menus
3. Visually appealing Tables

No UML representation of this story.

# 

# 

# 

# 

# 

# 

# 

# 

# 

# 

# 

# [#124](https://fiu-scis-seniorproject.mingle.thoughtworks.com/projects/advisor_availability_notificat/cards/124) Student Queue

### Description:

* Second Split of the first features Story.
* A queue with a sign in of walk in students

### Acceptance Criteria:

1. A queue displaying walk in students at the front desk computer
2. The same queue should be displayed in another page, to be displayed for everyone to see.

**Modeling:** Refer to UML diagrams

# [#129](https://fiu-scis-seniorproject.mingle.thoughtworks.com/projects/advisor_availability_notificat/cards/129) Admin Login

### Description:

* As an admin i would like to add/remove/ or edit user accounts

### Acceptance Criteria:

1. Able to add/ remove or edit user accounts.
2. Cannot remove itself

**Modeling:** Refer to UML diagrams

# 

# 

# 

# 

# 

# 

# 

# 

# [#131](https://fiu-scis-seniorproject.mingle.thoughtworks.com/projects/advisor_availability_notificat/cards/131) Notify advisor when student is on the way

### Description:

* As an advisor I want to be notified when a student is sent my way.

### Acceptance Criteria:

1. A pop up with sound should be displayed as a notification.

**Modeling:** Refer to UML diagrams

# [#132](https://fiu-scis-seniorproject.mingle.thoughtworks.com/projects/advisor_availability_notificat/cards/132) Appointment Queue

### Description:

* As the front desk I would like to have a queue of appointment students that arrive and sign in.

### Acceptance Criteria:

1. Student arrival time and appointment time must be displayed
2. Student id or name must be displayed

**Modeling:** Refer to UML diagram

# [#134](https://fiu-scis-seniorproject.mingle.thoughtworks.com/projects/advisor_availability_notificat/cards/134) Notify advisor when appointment student is on the way

### Description:

* As an advisor I want to be notified when an appointment student is sent my way.

### Acceptance Criteria:

1. A pop up with sound should be displayed as a notification.
2. Name of student should be displayed in the notification. **Modeling:** Refer to UML diagram

## Pending User Stories

# [#133](https://fiu-scis-seniorproject.mingle.thoughtworks.com/projects/advisor_availability_notificat/cards/133) Deployment

### Description:

* The system needs to be deployed as a FIU engineering center website

### Acceptance Criteria:

1. System must work online or locally with all of the advisor’s computers

**Modeling:** Refer to UML diagrams

# Project Plan

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

## Hardware and Software Resources

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

**Mingle:**

Mingle was used to plan and keep track of our scrum development progress.

**SQL Database:**

An SQL database was used to store vital information for the function of the program.

**PhpMyAdmin:**

Tool used to handle the database.

**Netbeans 8.2 :**

IDE used to develop the required php and javascript code.

**Xampp control panel v3.2.2:**

Tool used to set up an **apache 2.0** server with **php 5.6.24** as the server language.

This local server was used for testing.

**Git:**

A version control repository used to store finished or modified code.

## 

## Sprints Plan

### Sprint 1

Attendees: <Hector Borges, Mario Sanchez,Jean Tovar>

Start time: <2:00 PM 9/19/2016>

End time: <3:00 PM 9/19/2016>

After discussion, the velocity of the team were estimated to be <7>.

The product owner chose the following user stories to be done during the next sprint. They are ordered based on their priority.

* User Story <[**#**](https://fiu-scis-seniorproject.mingle.thoughtworks.com/projects/advisor_availability_notificat/cards/89)**89 First Features Story 1**>

The team members indicated their willingness to work on the following user stories.

* <Hector Borges>
* User Story <[**#**](https://fiu-scis-seniorproject.mingle.thoughtworks.com/projects/advisor_availability_notificat/cards/89)**89 First Features Story 1**>
* <Jean Tovar>
* User Story <[**#**](https://fiu-scis-seniorproject.mingle.thoughtworks.com/projects/advisor_availability_notificat/cards/89)**89 First Features Story 1**>

### Modeling: Future Stories Offer better models

### 

### 

### 

### Sprint 2

Attendees: < Mario Sanchez,Jean Tovar>

Start time: <2:00 PM 9/262016>

End time: <3:00 PM 9/26/2016>

Note: Hector was Unable to make it to the (Planning,Retrospective and Review) meeting from 2 to 3. Because of that Jean and Hector met before and after the meeting time in order for Hector to provide his input and to be briefed.

After discussion, the velocity of the team were estimated to be <14 days>.

The product owner chose the following user stories to be done during the next sprint. They are ordered based on their priority.

* User Story <[**#89**](https://fiu-scis-seniorproject.mingle.thoughtworks.com/projects/advisor_availability_notificat/cards/89) **First Features Story 1**>( this story was later divided into other stories, due to its size)

The team members indicated their willingness to work on the following user stories.

* <Hector Borges>
* User Story <[**#89**](https://fiu-scis-seniorproject.mingle.thoughtworks.com/projects/advisor_availability_notificat/cards/89) **First Features Story 1**>
* <Jean Tovar>
* User Story <[**#89**](https://fiu-scis-seniorproject.mingle.thoughtworks.com/projects/advisor_availability_notificat/cards/89) **First Features Story 1**>

**Modeling:** Refer to UML diagrams

### Sprint 3

Attendees: < Mario Sanchez,Jean Tovar,Hector Borges>

Start time: <2:00 PM 10/10/2016>

End time: <3:00 PM 10/10/2016>

After discussion, the velocity of the team were estimated to be <14 days>.

The product owner chose the following user stories to be done during the next sprint. They are ordered based on their priority.

* User Story <#**128 Multiple user Login**>
* User Story <**#130 Front desk Login**>
* User Story <**#127 Notification Alert**>
* User Story <**#123 Advisor signals front desk**>
* User Story <**#124 Student Queue**>

The team members indicated their willingness to work on the following user stories.

* <Hector Borges>
* User Story <**#127 Notification Alert**>
* User Story <**#123 Advisor signals front desk**>
* <Jean Tovar>
* User Story <#**128 Multiple user Login**>
* User Story <**#130 Front desk Login**>
* User Story <**#124 Student Queue**>

**Modeling:** Refer to UML diagrams

### Sprint 4

Attendees: < Mario Sanchez,Jean Tovar,Hector Borges>

Start time: <2:00 PM 10/24/2016>

End time: <3:00 PM 10/24/2016>

After discussion, the velocity of the team were estimated to be <14 days>.

The product owner chose the following user stories to be done during the next sprint. They are ordered based on their priority.

* User Story < **#124 Student Queue**>
* User Story < **# 126 Graphical User Interface** >
* User Story < **# 129 Admin Login**>
* User Story < **# 129 Notify advisor when student is on the way**>

The team members indicated their willingness to work on the following user stories.

* <Hector Borges>
* User Story < **# 129 Admin Login**>
* User Story < **# 129 Notify advisor when student is on the way**>
* <Jean Tovar>
* User Story < **#124 Student Queue**>
* User Story < **# 126 Graphical User Interface** >

### Sprint 5

Attendees: < Mario Sanchez,Jean Tovar,Hector Borges>

Start time: <2:00 PM 11/7/2016>

End time: <3:00 PM 11/7/2016>

After discussion, the velocity of the team were estimated to be <14 days>.

The product owner chose the following user stories to be done during the next sprint. They are ordered based on their priority.

* User Story < **#133 Deployment**>
* User Story < **# 126 Graphical User Interface** >
* User Story <**#132 Appointment Queue**>
* User Story < **# 134 Notify advisor when appointment student is on the way**>

The team members indicated their willingness to work on the following user stories.

* <Hector Borges>
* User Story < **# 134 Notify advisor when appointment student is on the way**>
* User Story <**#132 Appointment Queue**>
* <Jean Tovar>
* User Story < **#133 Deployment**>
* User Story < **# 126 Graphical User Interface** >

### 

***Sprint 6 and 7***

Attendees: < Mario Sanchez,Jean Tovar,Hector Borges>

Start time: <2:00 PM 11/7/2016>

End time: <3:00 PM 11/7/2016>

< No future stories, documentation work only>

# System Design

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

## Architectural Patterns

The architectural pattern used is a 2-tier client/server architecture. The application layer contains UI and application logic while the data layer contains the database with user information.

Advisor Logic

Admin Logic

Front Desk Logic

Database

## 

## System and Subsystem Decomposition:

The system is made up of a login interface which is then split into front desk, advisor or administrator after the credentials have been entered. If the credentials are for front desk there are some subsystems related to front desk. One is walk-in handle another is appointment handle and lastly advisor display.For advisor there is the advisorTable subsystem that fetches the status and the advisor systems handles notifications and status change,there is also a subsystem that handles password change.For admin there is a table fetching subsystem that retrieves the whole table of users, other functions are handled the admin subsystem.

Walk-in

Handle

Login

Appmnt. Handle

FetchTable

Admin

Front Desk

Advisor

Get Stat.Tbl.

DataBase

DatBse. Con.

Fetch Status

Change Password

## Deployment Diagram

All components of the website are in the same device, the database is located locally in the same server hosting the site.

# System Validation

**Story #130 Front Desk Login**

**System Test:**

ST-1S1: Front Desk logs in correctly with correct credentials:Pass

ST-2S2:Front Desk prevent log in with incorrect password: Pass

**Subsystem Test:**

SUBT-1S1: Correct credentials are passed to the next page upon submission: Pass

**Story #128 Multiple user Login**

**System Test:**

ST-3S1: Different advisors have access to the system at the same time:Pass

**Subsystem Test:**

SUBT-2S1: Correct credentials are passed to the next page upon submission: Pass

**Story #123 Advisor Signals front desk**

**System Test:**

ST-4S1: Advisor sets status to RFW, front desk plays a sound:Pass

ST-5S2: Advisor sets status to RFA, front desk plays a sound:Pass

ST-6S2:Advisor sets status to anything else,front desk does not play a sound:Pass

**Subsystem Test:**

SUBT-3S1: Advisor sends the correct status to database upon status change: Pass

SUBT-4S2;Front Desk plays a sound when a status change to select specific has been made:Pass

**Story #127 Notification alert**

**System Test:**

**Tested with story #123**

**Subsystem Test:**

**Tested with story #123**

**Story #124 Student Queue**

**System Test:**

ST-7S1: Front Desk can add a student to the walk-in queue:Pass

ST-8S2: Front Desk can send a walk-in student to a specific advisor:Pass

ST-9S3:Front Desk can delete a walk-n student:Pass

**Subsystem Test:**

SUBT-5S1: Walk-in handle correctly adds/removes/sends required students: Pass

**Story #132 Appointment Queue**

**System Test:**

ST-10S1: Front Desk can add a student to the appointment queue:Pass

ST-11S2: Front Desk can send an appointment student to a specific advisor:Pass

ST-12S3: Front Desk can delete an appointment student:Pass

**Subsystem Test:**

SUBT-6S1:Appointment handle correctly adds/removes/sends required students:Pass

**Story #129 Admin Login**

**System Test:**

ST-13S1 :An admin account with the correct credentials is able to login:Pass

ST-14S2: Admin accounts can add new users :Pass

ST-15S3: Admin accounts can remove users: Pass

ST-16S4:Admin accounts can change passwords:Pass

**Subsystem Test:**

SUBT-7S1:Appointment handle correctly adds/removes/sends required students:Pass

**Story #131 Notify advisor when student is on the way**

**System Test:**

ST-16S1 : Advisor screen displays an alert when front desk sends a new walk\_in student:Pass

**Subsystem Test:**

SUBT-8S1:Walk\_in handle correctly marks the required section of the database when sending a student:Pass

SUBT-9S1:Walk\_in handle correctly marks the required section of the database when sending a student:Pass

**Story #134 Notify advisor when appointment student is on the way**

**System Test:**

ST-17S1 : Advisor screen displays an alert when front desk sends a new appointment student:Pass

**Subsystem Test:**

SUBT-9S1:Appointment handle correctly marks the required section of the database when sending a student:Pass

# 

# 

# 

# Glossary

GLOSSARY - USERS

● Front Desk - a user that is in charge of managing the student and walk in queue as well as signaling advisors when they send a student to the advisor's office.

● Advisor -a user that is in charge of advising student and requires front desk to send students his or her way.

● Admin- a user that is in charge of adding removing and creating accounts, and also changing account passwords.

● Display- An output screen to display both queues.

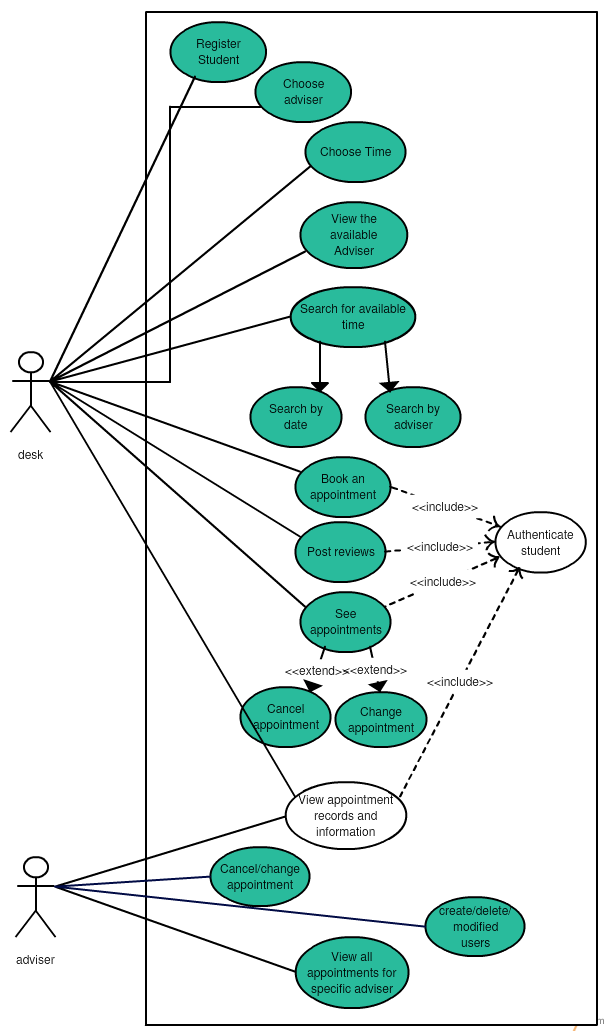
GLOSSARY -Technical Terms

* **UI** - User Interface
* **Mingle:**Mingle was used to plan and keep track of our scrum development progress.
* **SQL Database:** An SQL database was used to store vital information for the function of the program.
* **PhpMyAdmin:**Tool used to handle the database.
* **Netbeans 8.2 :**IDE used to develop the required php and javascript code.
* **Xampp control panel v3.2.2:**Tool used to set up an **apache 2.0** server with **php 5.6.24** as the server language.This local server was used for testing.
* **Git:**A version control repository used to store finished or modified code.

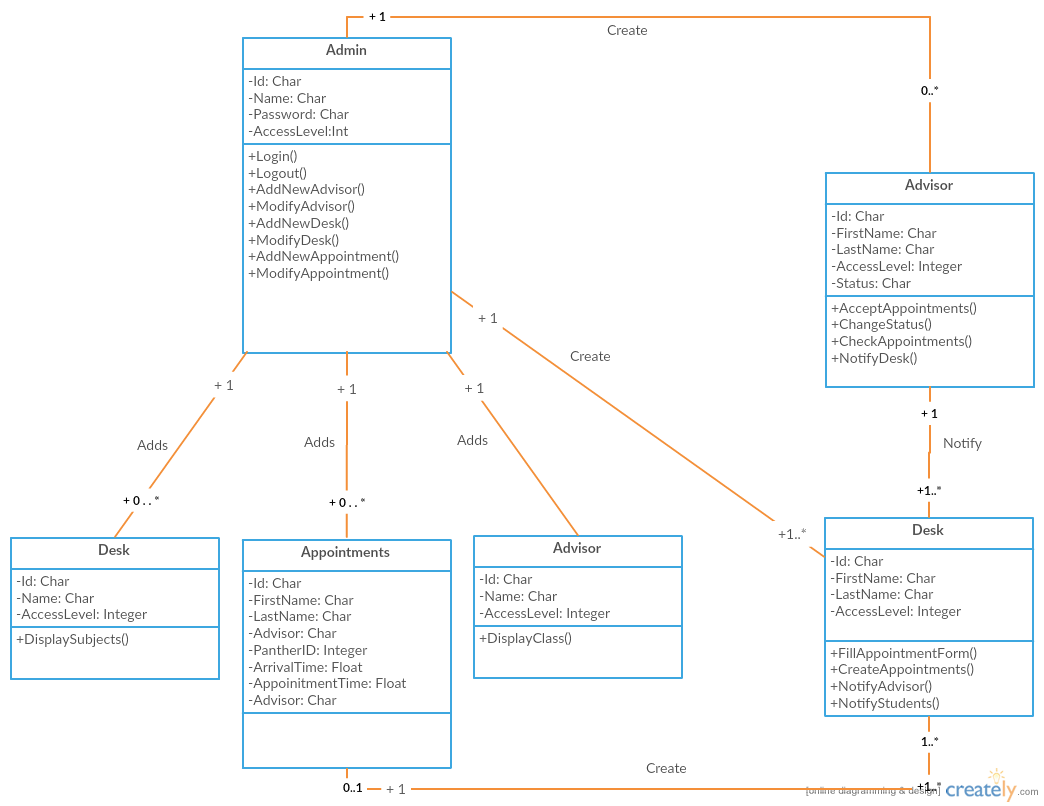
# Appendix

## Appendix A - UML Diagrams

### Static UML Diagrams



**Class Diagram**



## [#130](https://fiu-scis-seniorproject.mingle.thoughtworks.com/projects/advisor_availability_notificat/cards/130) Front desk Login

## 

## 

## 

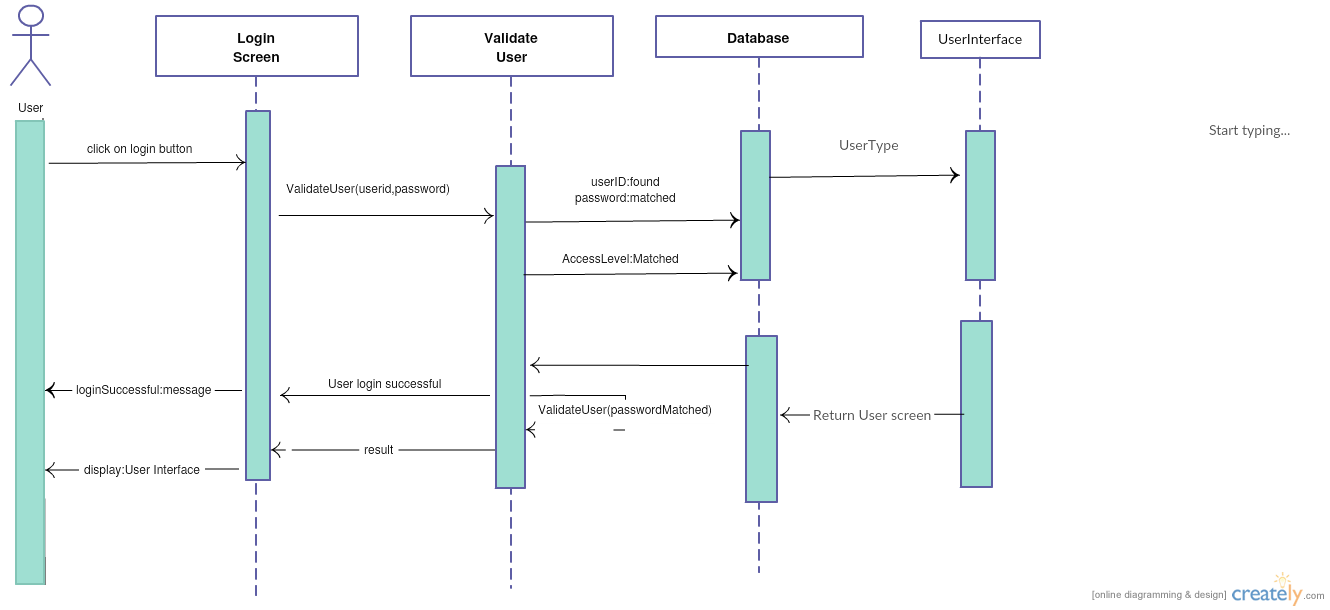
## 

## 

## 

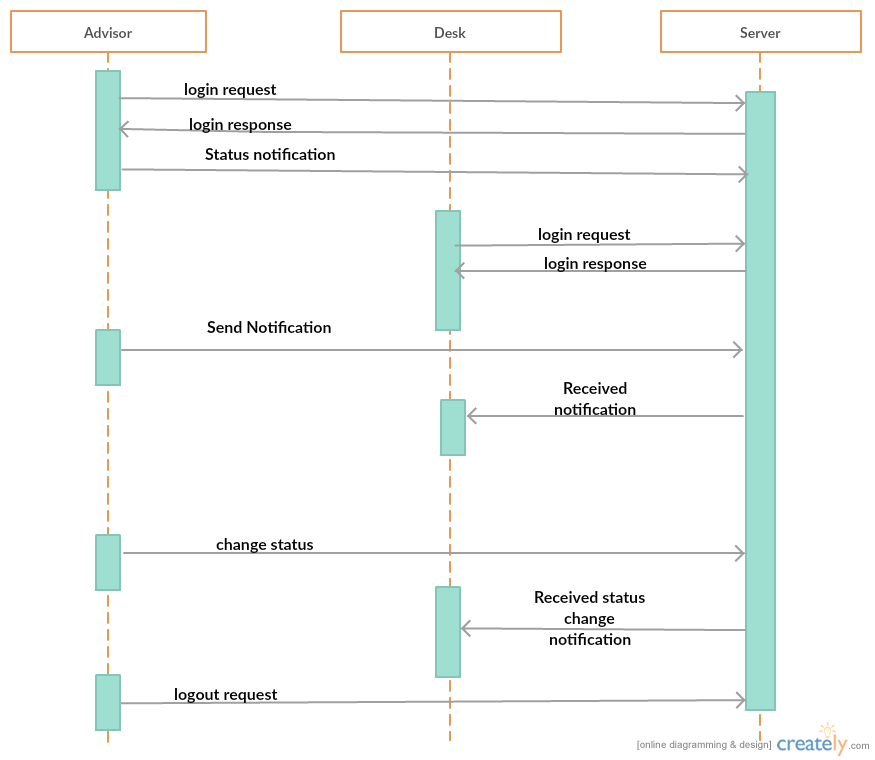
## 

## [#128](https://fiu-scis-seniorproject.mingle.thoughtworks.com/projects/advisor_availability_notificat/cards/128) Multiple user Login



## 

# [#123](https://fiu-scis-seniorproject.mingle.thoughtworks.com/projects/advisor_availability_notificat/cards/123) Advisor signals front desk

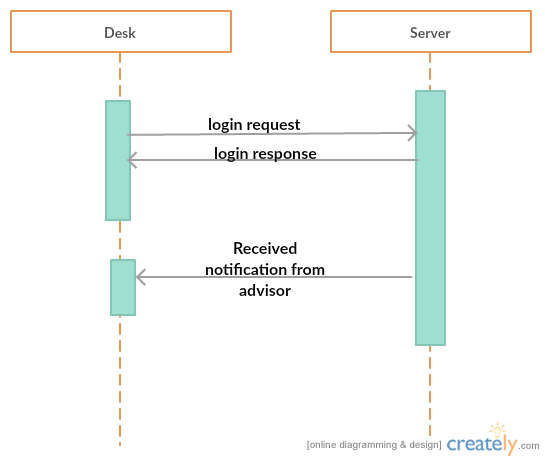


# 

# 

# 

# [#127](https://fiu-scis-seniorproject.mingle.thoughtworks.com/projects/advisor_availability_notificat/cards/127) Notification alert



# 

# 

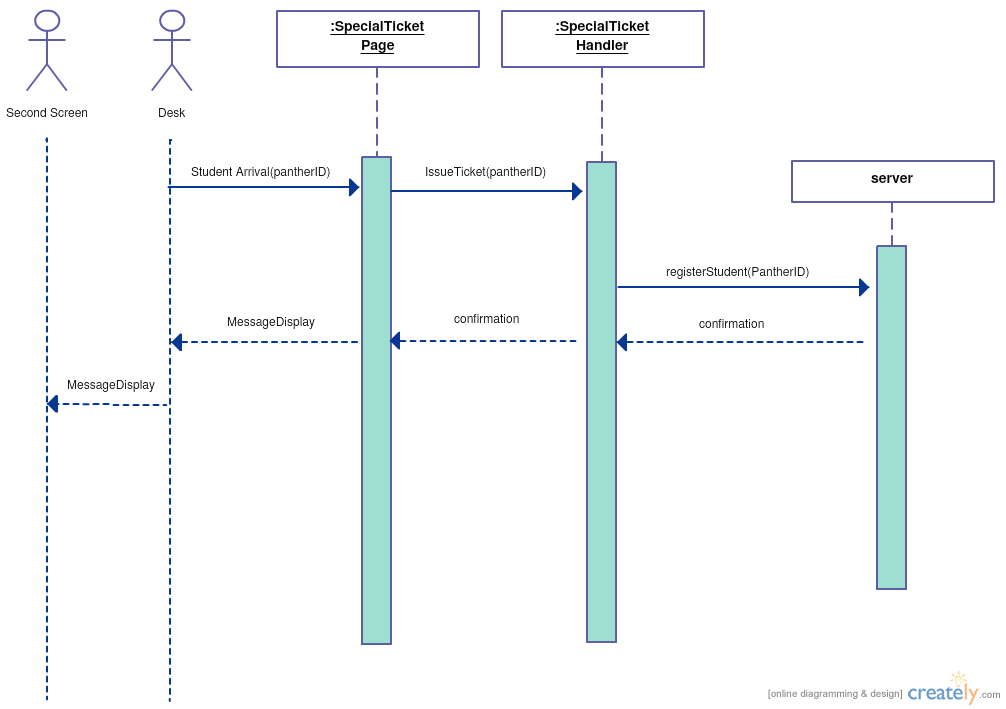
# 

# 

# 

# 

# [#124](https://fiu-scis-seniorproject.mingle.thoughtworks.com/projects/advisor_availability_notificat/cards/124) Student Queue



# 

# 

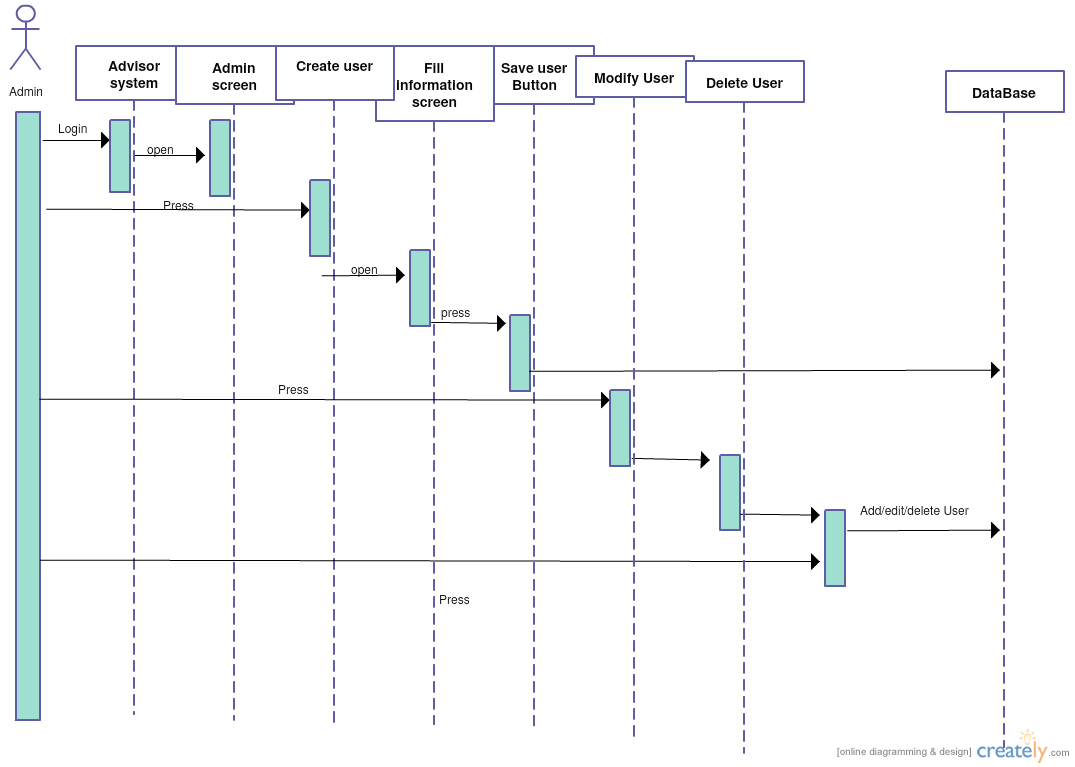
# 

# 

# 

# 

# [#129](https://fiu-scis-seniorproject.mingle.thoughtworks.com/projects/advisor_availability_notificat/cards/129) Admin Login



# 

# 

# 

# 

# 

# 

# [#131](https://fiu-scis-seniorproject.mingle.thoughtworks.com/projects/advisor_availability_notificat/cards/131) Notify advisor when student is on the way

# New advisor appointment system sequence.png

# 

# 

# 

# 

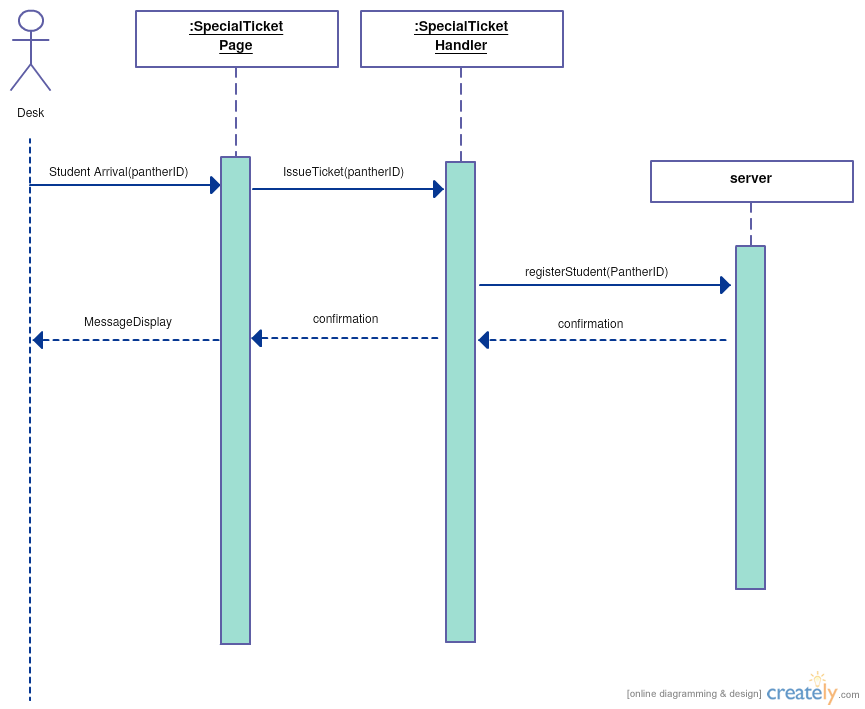
# 

# 

# 

# 

# [#132](https://fiu-scis-seniorproject.mingle.thoughtworks.com/projects/advisor_availability_notificat/cards/132) Appointment Queue



# 

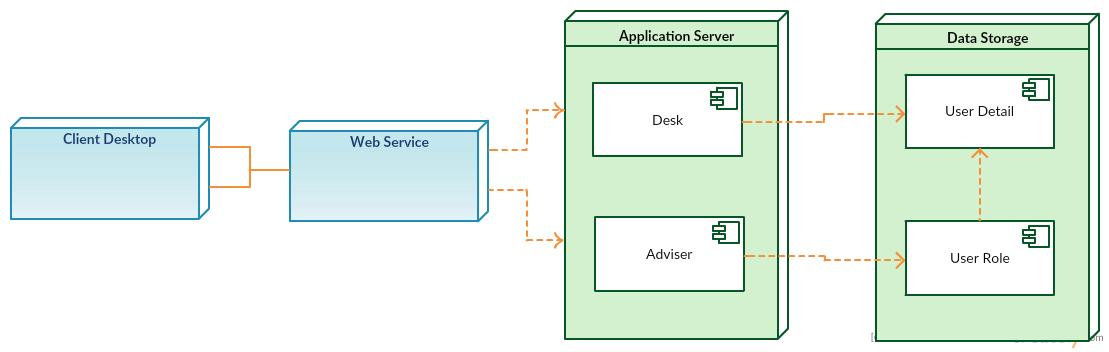
# 

# 

# 

# 

# [#133](https://fiu-scis-seniorproject.mingle.thoughtworks.com/projects/advisor_availability_notificat/cards/133) Deployment



# 

# 

# 

# 

# 

# 

# 

# 

# 

# 

# 

# 

# 

# 

# [#134](https://fiu-scis-seniorproject.mingle.thoughtworks.com/projects/advisor_availability_notificat/cards/134) Notify advisor when appointment student is on the way

# New advisor appointment system sequence.png

# 

# 

# 

# 

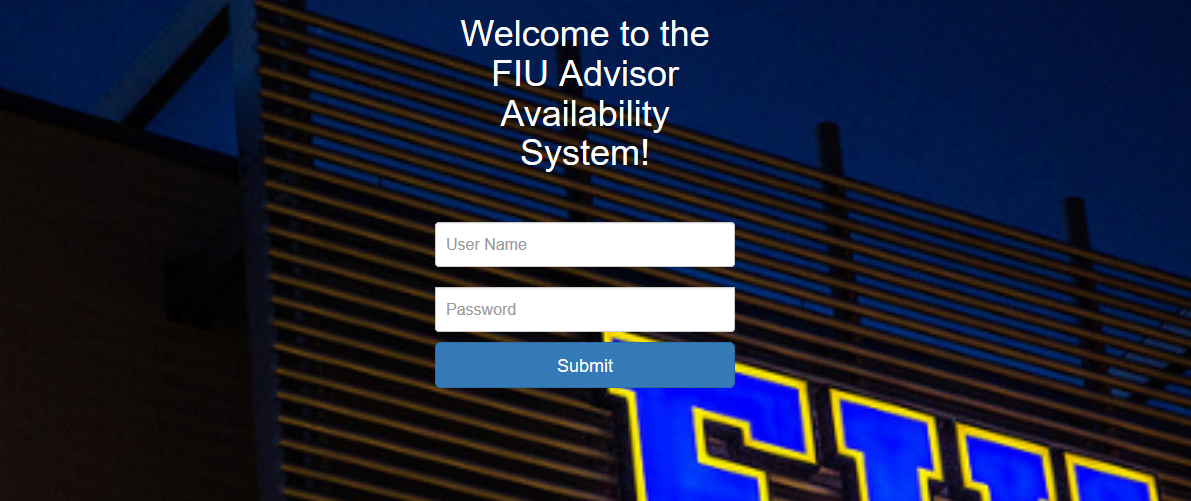
# 

# 

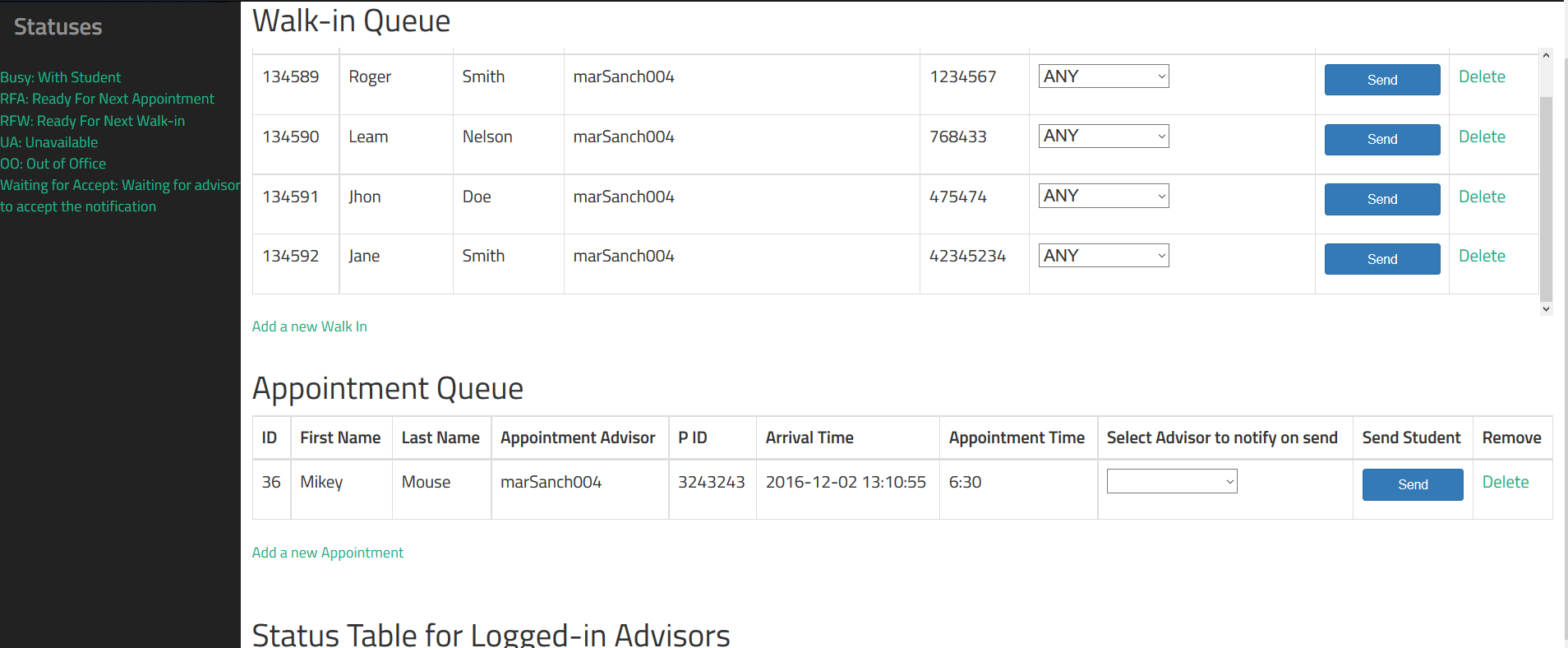
# 

# Appendix B - User Interface Design

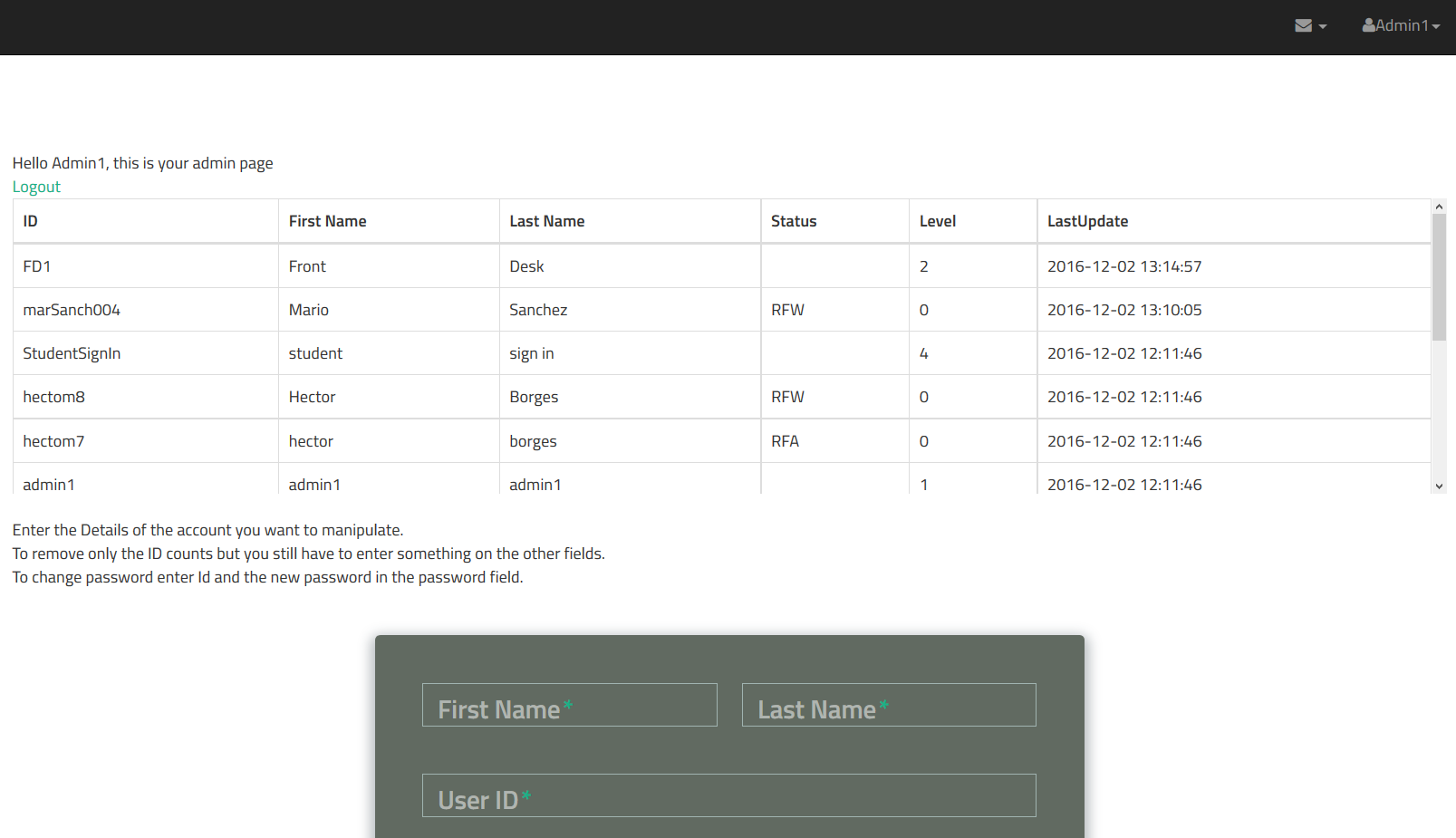
**Login Page:**



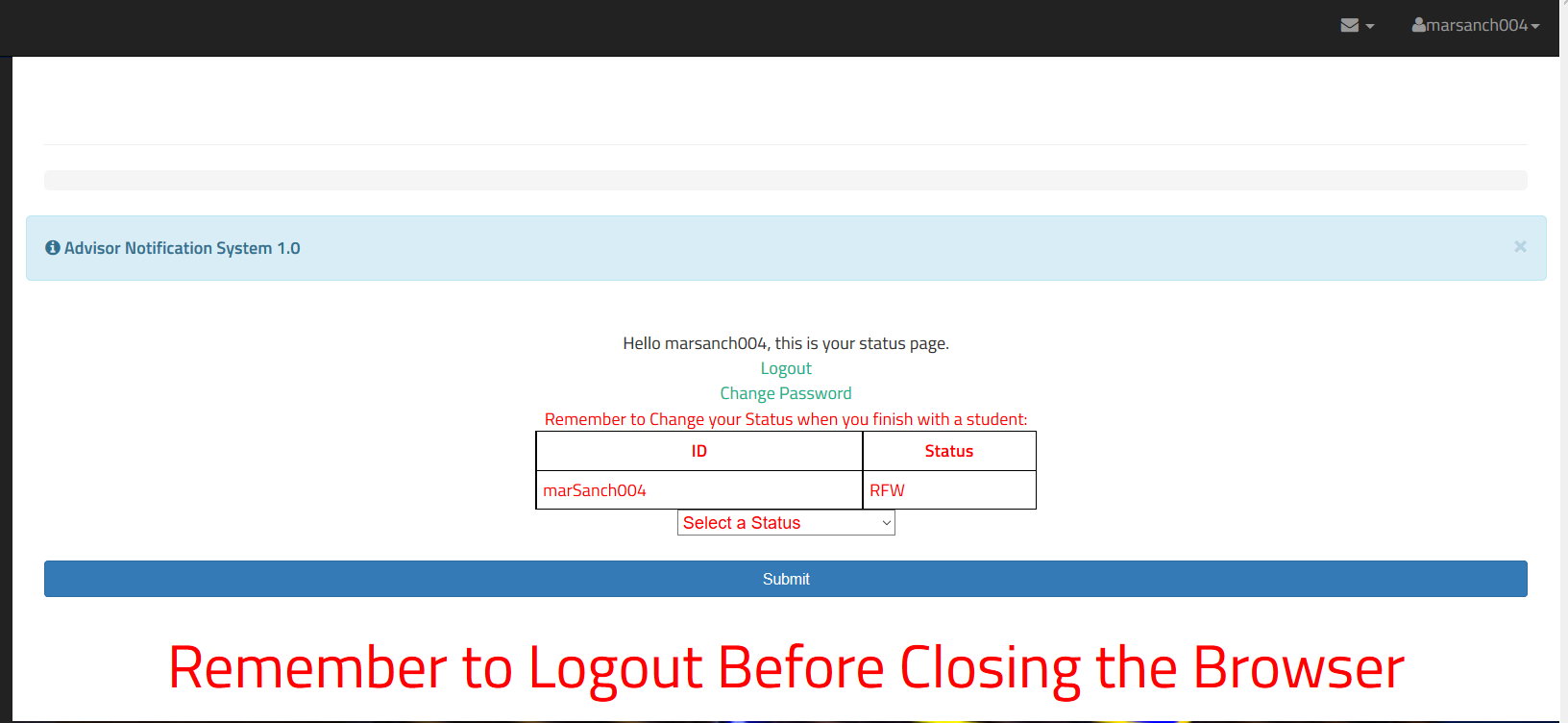
**Front Desk:**



**Admin Page:**



**Advisor Page:**



## Appendix C - Sprint Review Reports

Attendees: <Jean Tovar,Mario Sanchez>

Start time: <2:00 9/26/2016>

End time: <2:20 9/26/2016>

Note: Hector was Unable to make it to the (Planning,Retrospective and Review) meeting from 2 to 3. Because of that Jean and Hector met before and after the meeting time in order for Hector to provide his input and to be briefed.

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

* User Story <None>

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

* User Story <# 89 First Feature Story 1>
* How this should be reflected on the user story definition in Mingle:
  + This story is Described Under Sprint 2

Attendees: <Jean Tovar,Mario Sanchez, Hector Borges>

Start time: <10/10/2016 2:00 >

End time: < 10/10/2016 2:20>

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

* User Story <None>

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

* User Story <#123 Advisor signals front desk>
* How this should be reflected on the user story definition in Mingle:
  + This story is Described Under Sprint 3

## 

Attendees: <Jean Tovar,Mario Sanchez, Hector Borges>

Start time: <10/24/2016 2:00 >

End time: < 10/24/2016 2:20>

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

* User Story <#**128 Multiple user Login**>
* User Story <**#130 Front desk Login**>
* User Story <**#127 Notification Alert**>
* User Story <**#123 Advisor signals front desk**>

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

* User Story <**#124 Student Queue** >
* How this should be reflected on the user story definition in Mingle:
  + This story is Described Under Sprint 5

## 

## 

## 

## 

Attendees: <Jean Tovar,Mario Sanchez, Hector Borges>

Start time: <11/07/2016 2:00 >

End time: < 11/07/2016 2:20>

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

* User Story <**#124 Student queue**>
* User Story <**#129 Admin Login**>
* User Story <**#131 Notify advisor when student is on the way**>

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

* User Story <**#126 Graphical User interface**>
* How this should be reflected on the user story definition in Mingle:
  + This story is Described Under Sprint 6

## 

## 

## 

Attendees: <Jean Tovar,Mario Sanchez, Hector Borges>

Start time: <11/19/2016 2:00 >

End time: < 11/19/2016 2:20>

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

* User Story <**#124 Appointment Student queue**>
* User Story <**#126 Graphical User interface**>
* User Story <**##134 Notify advisor when appointment student is on the way**>

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.

Finished.

## 

## 

## 

## 

## 

## 

## 

## 

## 

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

**User Manual:**

**Note: The advisor page is only correctly working on Chrome Browsers**

User: Admin

When the website is setup, the SQL script provided will offer two admin accounts:admin1 and admin2 each with password 1234.

The person in charge of adding or removing advisors in the system should be given access to these two admin accounts.

Admin accounts have the ability to change any account password to the desired password, however they are unable to view any passwords. If a user or another admin forgets their password another admin account has the ability to make a new password and give it to the user, later the user can change the password to any other desired password.

* How can an admin change a password? In the Id field entering the user id of the desired account and in the password field entering the new password of the desired account.Everything else can be left blank except for the last drop down menu, there you can select the change password option and then press submit. The password should be changed for the desired account.
* How can an admin add an account? An account can be added by filling in all the desired fields, and choosing the account type from the drop down menu. After all fields have been filled and selected, on the last drop down menu, select the add option and click submit, the new account should be added.
* How can an admin remove an account?Account can be added by typing the User Id of the account to be removed and selecting remove on the last drop down menu and pressing submit.

User: Advisor

A sample advisor account is provided in the setup of the program within the sql script.The sample username is advisor1 and password 1234. Advisors can change their status and their passwords. After receiving an alert that a student is on the way the advisors should acknowledge it by pressing the ok button.

* How can an advisor change their status? Selecting a new status from the drop down menu and then clicking the submit button.
* How can an advisor change password? Click the change password link and enter the new password in each of the two fields, then press submit.

User: Front Desk

When the website is setup, the SQL script provided will offer a sample front desk account: FD1 with password 1234. Front desk is able to add or remove students into both student queues and is able to send students to advisors.

* How can front desk add students to the queue?Students can be added by clicking either the add walk-in or add appointment links and filling out the information then pressing the submit button.
* How can students be sent to advisors?A student can be sent if he has been added to the queue, the front desk attendant will select what advisor to send the student to and then press the send button.
* How can a student be deleted? Press the delete link on the row that the student is located in.

User: Display

Display will display both student queues for everyone to see.Username provided is Display and 1234 for password.

**Installation:**

**Note: The advisor page is only correctly working on Chrome Browsers**

A person with IT knowledge is required to setup this program, a person with knowledge of Databases and web server setups.

I will provide two options for setting up the website:

Option 1:

1. Install XAMPP v3.2.2 ( look into how to install XAMPP).
2. Start the apache and the SQL Server that comes with XAMPP.
3. In the XAMPP installation directory insert the file provided from github(<https://github.com/FIU-SCIS-Senior-Projects/ADVISOR-AVAILABILITY-NOTIFICATION-SYSTEM-Ver-1.0>) named ADVISOR-AVAILABILITY-NOTIFICATION-SYSTEM-Ver-1.0 ( this folder is inside the ADVISOR-AVAILABILITY-NOTIFICATION-SYSTEM-Ver-1.0-master/websitecode/website folder) into htdocs.
4. In the MySQL row in XAMPP click admin and phpMyAdmin should come up in a browser. This is to set up the Database.
5. On the top right side there should be an import button, click it and import the .sql script that is provided in the file named databaseSCRIPT
6. Now find a file named sqliConnect.php in ADVISOR-AVAILABILITY-NOTIFICATION-SYSTEM-Ver-1.0 and edit it to add the required database information. Originally the XAMPP database comes without a password, if you want your database to be secure you should add a password to it.
7. After importing the database, since the passwords were encrypted in a different machine they need to be re encrypted for the machine you wish to install it on. You can do this by running a small piece of code that re encrypts all passwords in the database, after running it every user and admin will have the password 1234. ( this can later be changed by admin or users themselves.
8. Now you should visit <http://localhost/ADVISOR-AVAILABILITY-NOTIFICATION-SYSTEM-Ver-1.0/> on your browser and a login page should appear, enter fd1 as username and 1234 as password and you should be logged in to the front desk.
9. To make the website live, configure the XAMPP server for internet access, you can also configure it for Local Network Access only.

Option 2:

1. Take the source code provided and configure it into a php server ( such as apache). The code was created using php 5.6.24 but it should work in php 5.3.7 and up ( but it has not been tested on php 5.3.7)
2. Create a MySQL database and import the SQL script provided.
3. Enter the database credentials into the sqliConnect.php file.
4. Do step 7 from above.
5. The website should run.

**Shortcomings:**

* Students can not sign themselves in, this can be implemented in future versions.
* Currently the advisor page only works correctly on chrome

# References

XAMPP Home Page:

* https://www.apachefriends.org/index.html

NetBeans Home Page:

* <https://netbeans.org/>

Bootstrap Home Page:

* http://getbootstrap.com/