Student Companion

Fourth Iteration Report

COMP-SCI 5551 Advanced Software Engineering

Group 7

Sri Harsha Chennavajjala (7)

Teja Garidepally (12)

Raj Kiran Reddy Munnangi (32)

Suhas Sai Raparthi (51)

# Introduction

This document is intended to provide an overall description of the project named “Student Companion” in detail. The project schedule and the plan of action is also discussed. The proposal document will give an insight on the project. The outcome of the fourth increment is the final web and mobile application with end to end user functionality.

# Project Goal and Objectives

The goal of this project is to provide various functionalities that a student uses regularly such as updating the profile, checking for computer lab availability, library study room reservation etc. The student details will already present in the database. The student has to login before he uses these functionalities. Main objectives of this application are:

* To reduce the student’s stress and to save the student’s time by providing the latest availability of the computer labs.
* To develop an application that helps the students in taking the decision on to which laboratory the students have to go.
* To secure the information by providing a login form to the end user.
* To provide a tool with which the students will be able to reserve the library study rooms.
* To ensure that the student will never miss his schedules by setting reminders.
* To enable the Student Assistants to view their shifts, post and take substitutions.
* To provide the students with the option to update their address or mobile phone number etc.

# Project Background and Related Work

Some functionalities of this application are already exist. We are creating a new android application which integrates (mash up) all the available and new functionalities under one hood, thus making the application a viable one. Some functionalities will be developed by importing the existing APIs into our application like Google Calendar API, Google Maps API etc. We are inspired by the problems that the Students are currently facing in reserving the study rooms, problems related to their working shifts and we came up with a solution which can resolve the existing challenges.

**Significance:**

The major significance of the application lies in mashing up of all the useful services under one system. This will save the student’s time and increases the productivity. The application will prevent the fraudulent usage by restricting the resources access to only the students who successfully logged in to the system. As of now, the student assistants has to go through a lengthy process in order to post or take shifts. The proposed application will make it easier for the student to perform such tasks by providing on-the-go support.

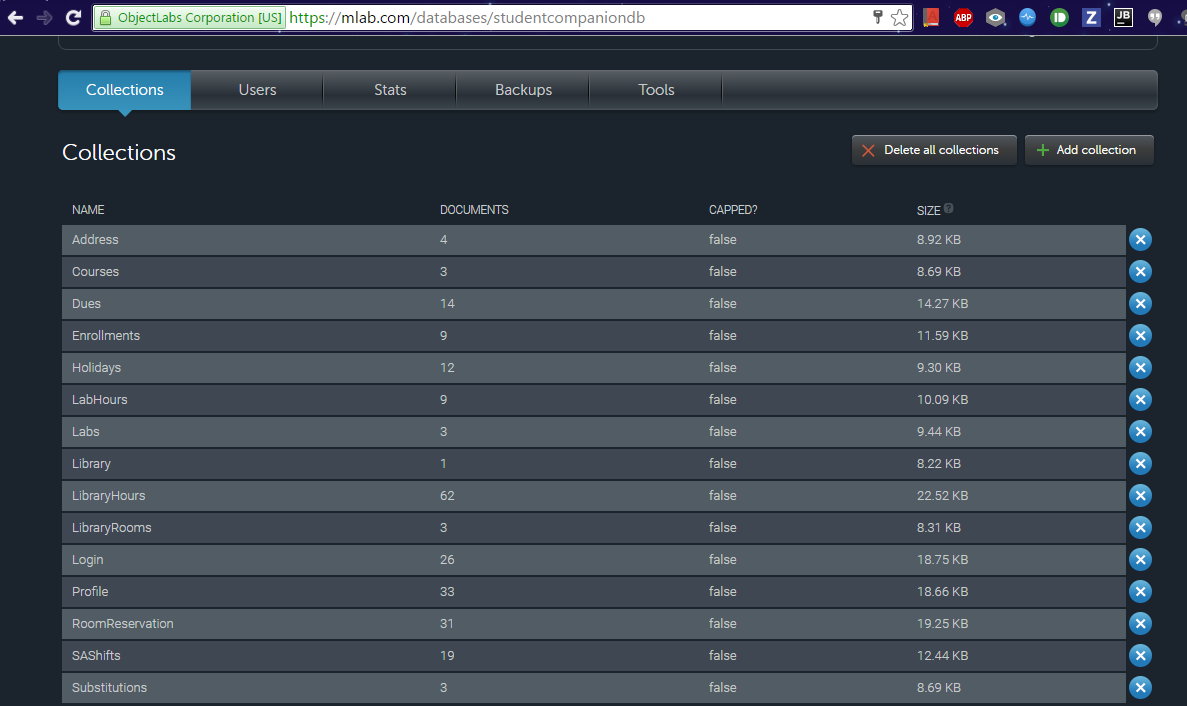
# Work flow

The first workflow of the application is that of the designing the User Interface for the system. Our application goal is to provide instant access to the user regarding lab availability and library room availability. So, for this we are developing a mobile application. We are using ionic framework so that we can develop hybrid application.

Second, we have created the login page and database using MongoLab and we have successfully authenticated the user based on his details which are stored in database in mongodb.

The third workflow deals with the creating a mongodb database to store and retrieve the user profile and information about IS labs, library study rooms etc. The work flow of the each and every feature of the application is explained through sequence diagrams and state chart diagram presented in the first increment report.

Fourth workflow deals with the end to end functional development of the application. Development of each and every feature of the application. Successfully running the application on web browser and in mobile phone.



**Fig: Collections in Mongo Database**

# Technological and architecture requirements

In the application system need to interact with the database for retrieving the data to do this interaction between the system and database is done through REST technology. We use CRUD API calls of the REST for accessing the database.

Ionic framework, it is a powerful SDK used to develop hybrid applications using web technologies like HTML, CSS and JavaScript. It is also a core in providing better UI to the user.

The architectural requirement aims at the development of flexible architecture for the better interaction between the system and the database.

# Existing Services/API

We have so far used the MongoDB in our application. MongoDB is a cross-platform document-oriented database and it is a NoSQL database, MongoDB avoids the traditional table-based relational database structure in favor of JSON-like documents with dynamic schemas making the integration of data in certain types of applications easier and faster.

We have used mongoDB to store the information of the users and data about the IS labs, library study rooms, student assistant information etc. Using REST API calls data is retrieved from the database and displayed to the user.

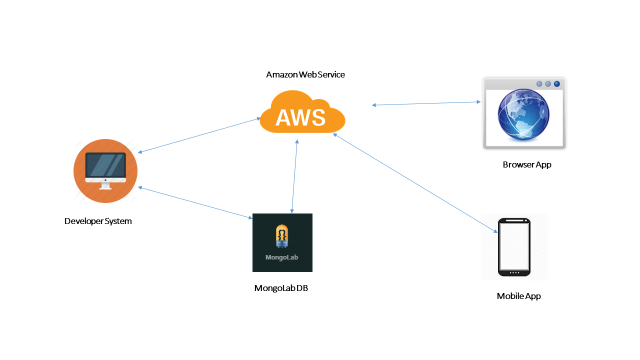
mLab is a fully managed cloud database service that hosts and provide featuring automated provisioning and scaling of MongoDB databases. Data can be accessed from mongoDB using two ways. Fist, by using the DATA APT URL in which it contains database name, collection name and API key.

Example URL: **[https://api.mongolab.com/api/1/databases/studentcorner/collections//Ase\_project/'](https://api.mongolab.com/api/1/databases/studentcorner/collections/Ase_project/'%20+%20id%20+%20'?apiKey=Q_u73BV4oOdMGpnu3WFGmJ8YH_lxHDHO)** [+ id +](https://api.mongolab.com/api/1/databases/studentcorner/collections/Ase_project/'%20+%20id%20+%20'?apiKey=Q_u73BV4oOdMGpnu3WFGmJ8YH_lxHDHO) **['?apiKey=Q\_u73BV4oOdMGpnu3WFGmJ8YH\_lxHDHO](https://api.mongolab.com/api/1/databases/studentcorner/collections/Ase_project/'%20+%20id%20+%20'?apiKey=Q_u73BV4oOdMGpnu3WFGmJ8YH_lxHDHO)**

Second, accessing the mongoDB using a driver to connect the database. An example driver:

mongodb://<dbuser>:<dbpassword>@ds011399.mlab.com:11399/studentcompaniondb

**REST Services:**

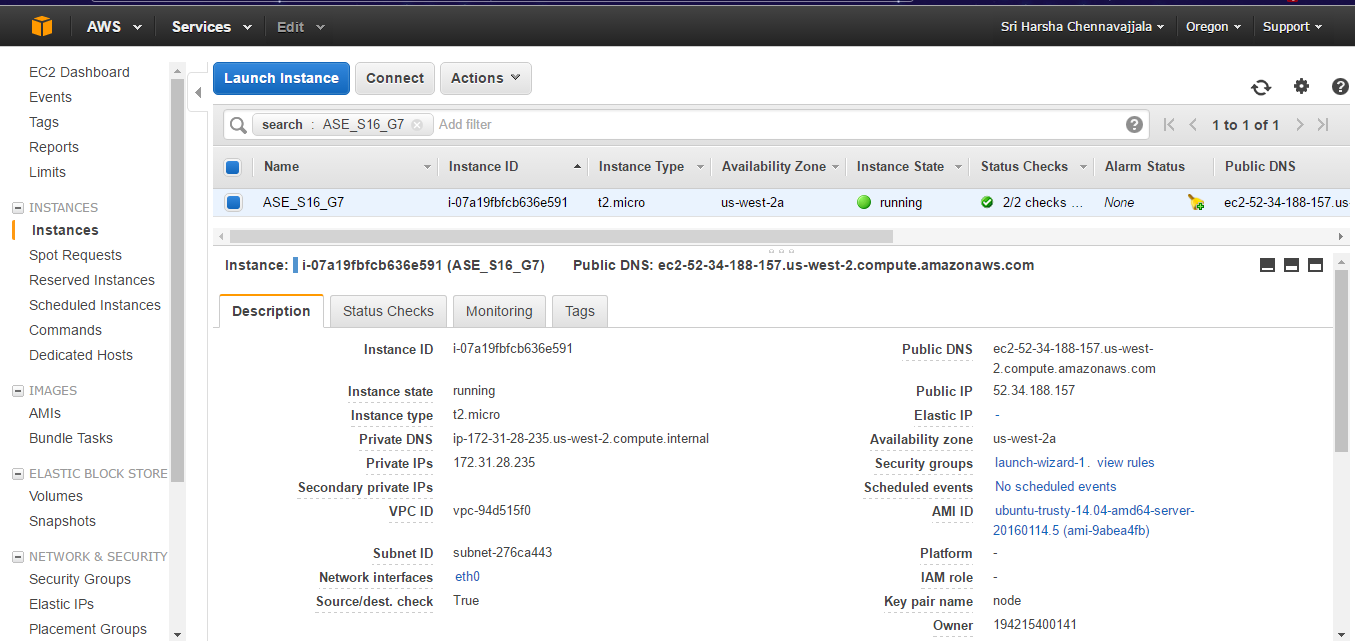


**Fig: REST API Communication with mongoDB and Application**

We have developed an API through which we perform the REST services for our application. As in the second increment we have developed database in the mongoLab and the data is fetched from the mongoLab through REST services.

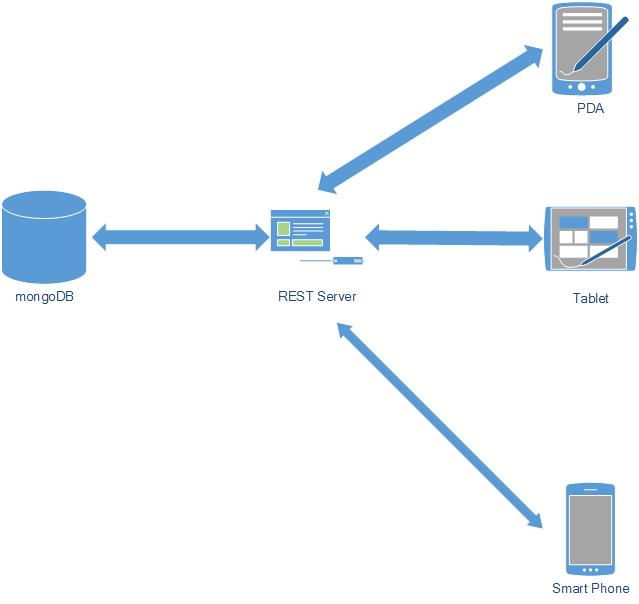
# Amazon Web Services (EC2):

We have used Amazon Web Services as a server to host our REST API. Our REST API will run in background and listen to port 9000. Our application send data request to the REST API hosted on Amazon EC2 cloud server. REST API will fetch the request data from mongoDB collections and perform the filtering of data according to the criteria and sends responds to the client application. This enables us complex queries on the server instead on the client thereby increasing the performance of client application.



**Fig: Amazon Web Services (EC2) Dashboard - Hosting our REST API**

# Architecture Diagram

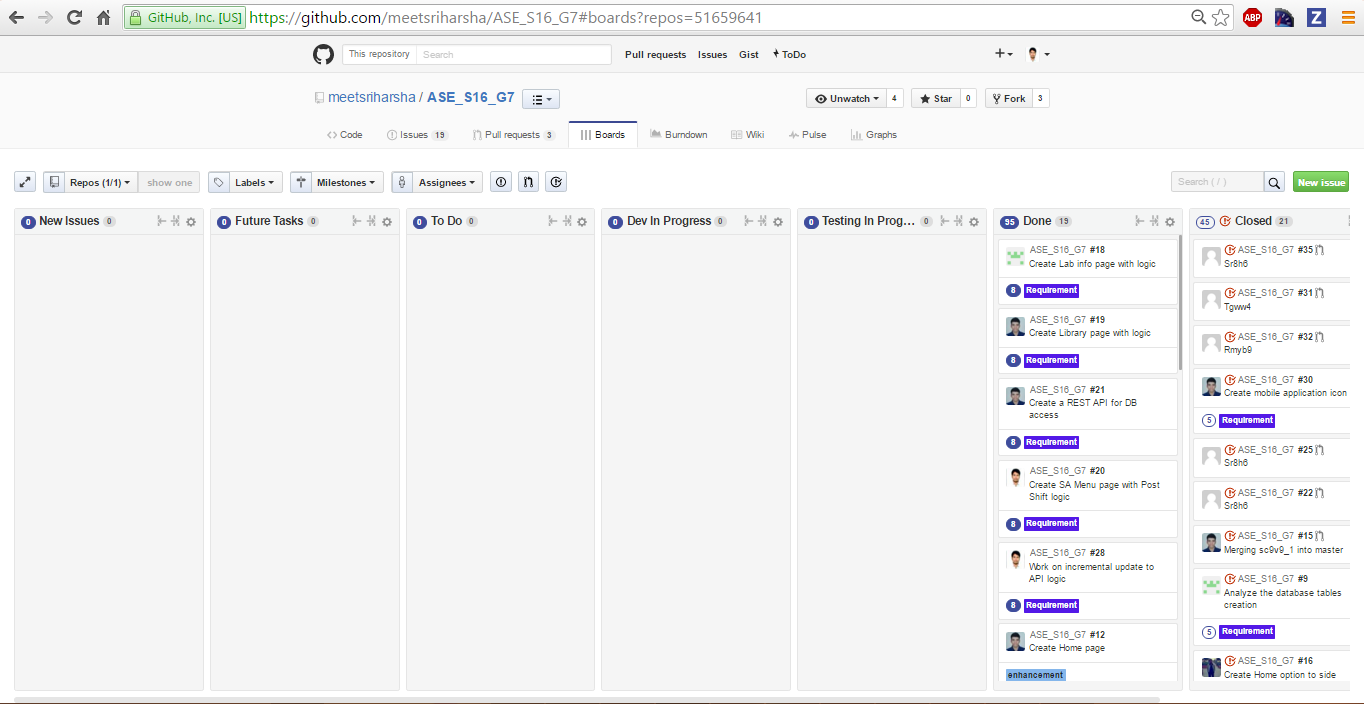


**Fig: Architecture Diagram for StudentCompanion application**

The architecture diagram for our application shows that it is 3-tier architecture. Application which deployed in smart phones, PDA’s and Tablet uses the REST services for interacting with the database for storing and retrieval of required information from the mongoDB.

# Project plan

The proposed project plan is outlined the by the screenshot from the ZenHub tool. The project is divided into four milestones. Each iteration has several states namely future tasks, new issues, to do, development in progress, testing in progress, done and closed.



**Fig: ZenHub Board showing the project plan and current tasks.**

**Milestone 4:**

This milestone mainly deals with the designing the end to end functionality of the system. The tasks of this phase mainly focuses six tasks.

First, the development of REST API which performs the communication between Client application and database.

Second, development of Profile page and display the profile of the user which is stored in the mongoDB. Developed the functionalities for register user, edit profile and edit password.

Third, development of Lab Information page and display the Lab name and lab details.

Fourth, development of Reserve Study Room page and display the Library information, study rooms which are reserved by the user and available study rooms and the details of resources present in it and cancellation of reserved study room.

Fifth, development SA Menu page and display the SA Shifts of the user and available shifts on a selected date. Developed the functionality for taking the available substitutions.

Sixth. Implemented the toast functionality for the final application.

The results of this milestone contains the screenshots of the application and the related class, sequence and state chart diagrams.

# Fourth Increment Report

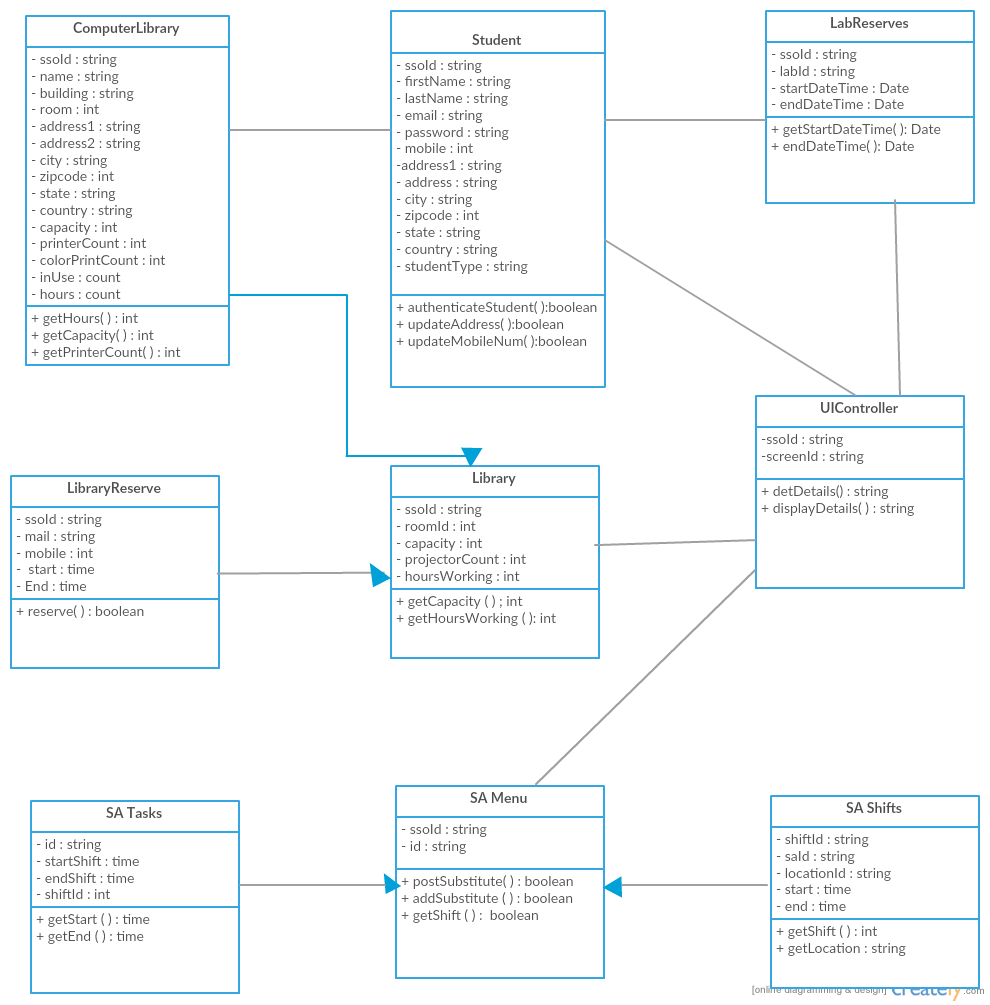
This document contains the report of fourth increment work done on the Student Companion application. This document emphasizes the final end to end implementation of the application using different technologies such as REST API, Amazon Web Services, Ionic framework and mongoDB. We’re using the HTML5, CSS ,Node.js, , ExpressJS and Angular JS for the end to end implementation.

We have created REST API which host on the Amazon Web Services through which communication between client application and database.

We have created the Profile, Register, Edit Profile, Lab Information, Reserve Study Room, SA Menu, Contact Us pages. We have also checked the work so far by deploying the application in the mobile and manual testing is done.

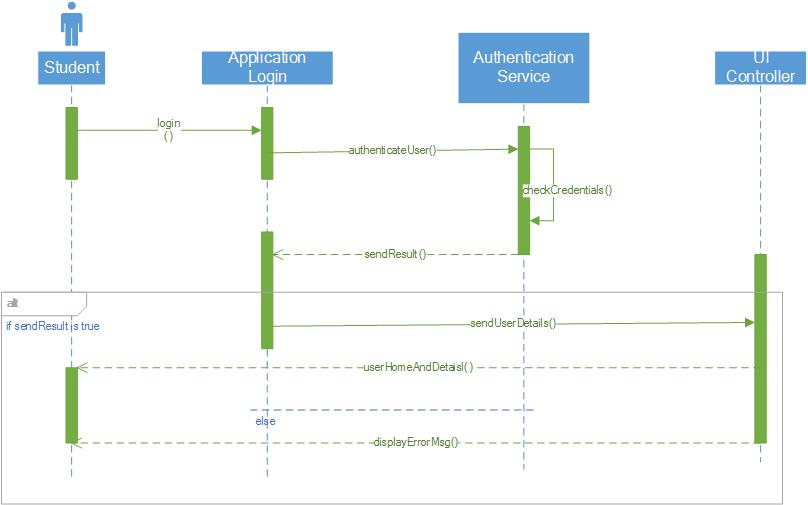
The detail development in fourth increment include the enhancement of Profile, Lab Information, Reserve Study Room, SA Menu, Contact Us pages and development of Register and Edit Profile pages. Developed an icon for the application.

**Class diagram for the high level design of the application:**

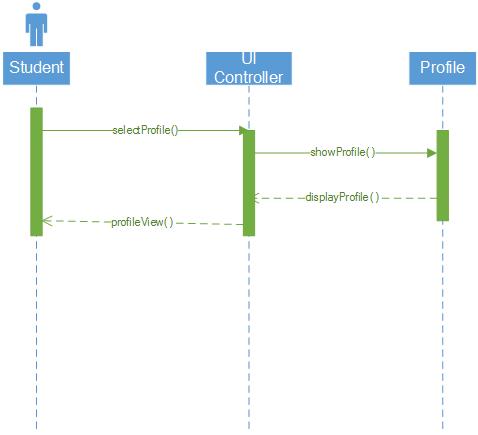


**Fig: Demonstrates a class diagram of high level design of application**

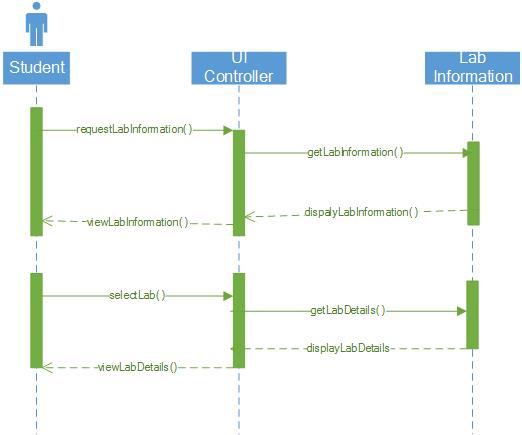
**Sequence diagrams for the high level design of the application:**



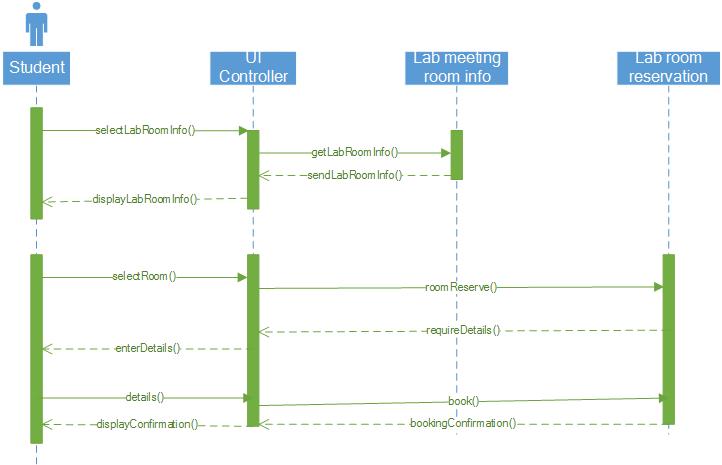
**Fig: Sequence diagram for student login activity**



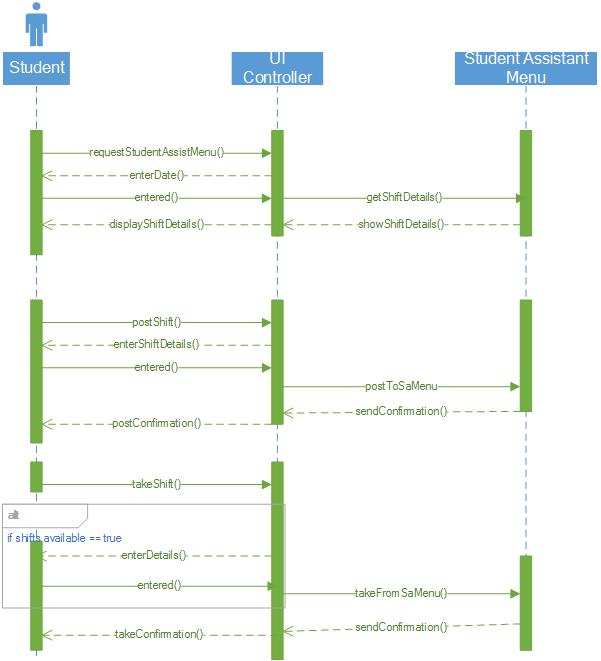
**Fig: Sequence diagram shows the control flow for “view user profile” task**



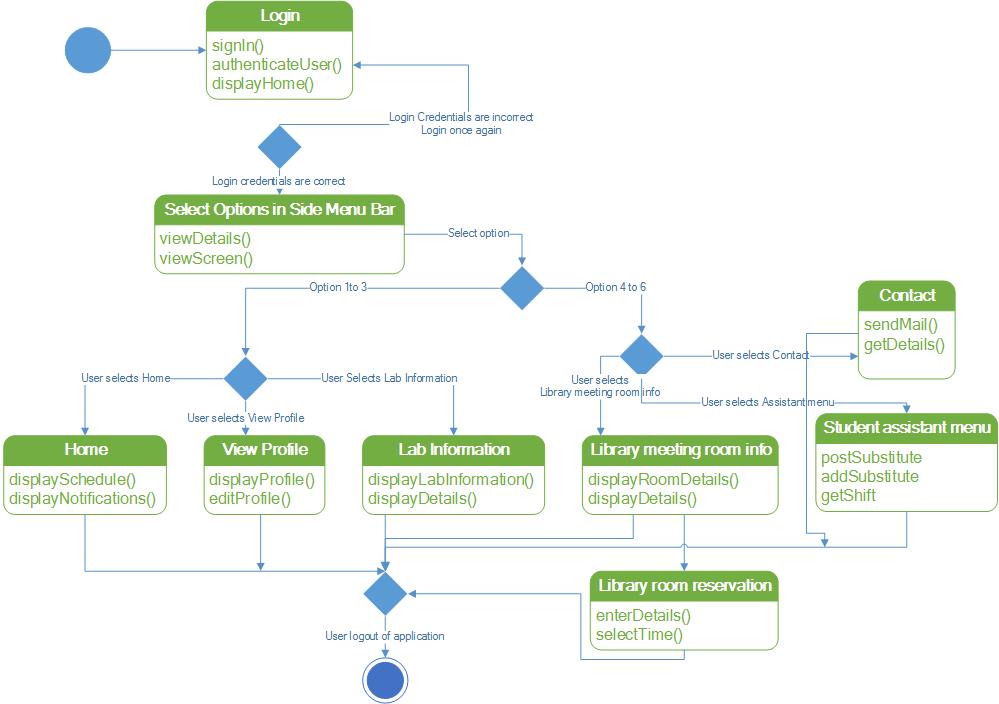
**Fig: Sequence diagram for “View lab information” activity**



**Fig: Sequence diagram for reserving library study room activity**

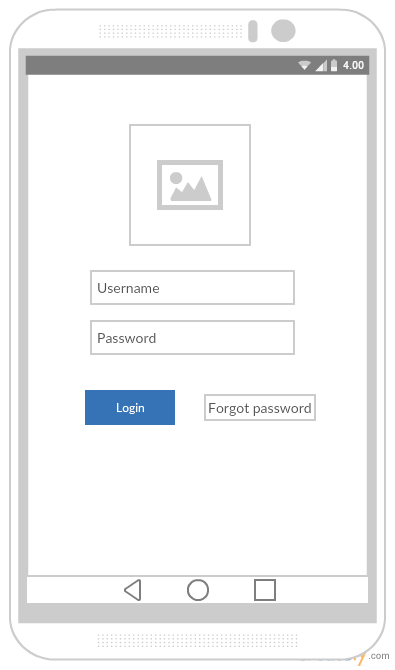


**Fig: Sequence diagram for student assistant activities like post or take shifts.**

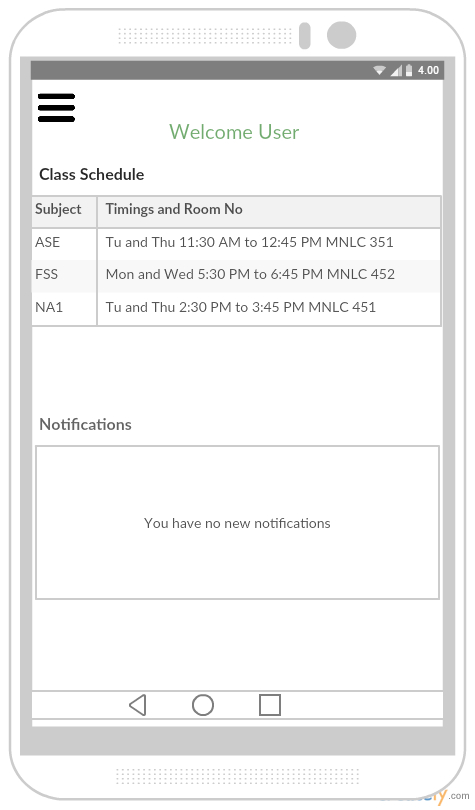


**Fig: State chart diagram for the application modules.**

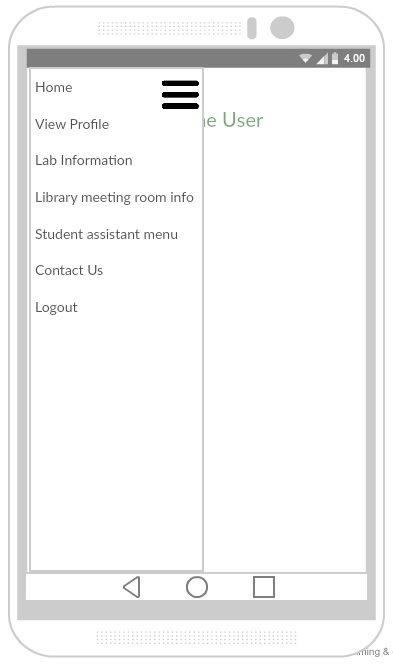
## Wireframes of the application:



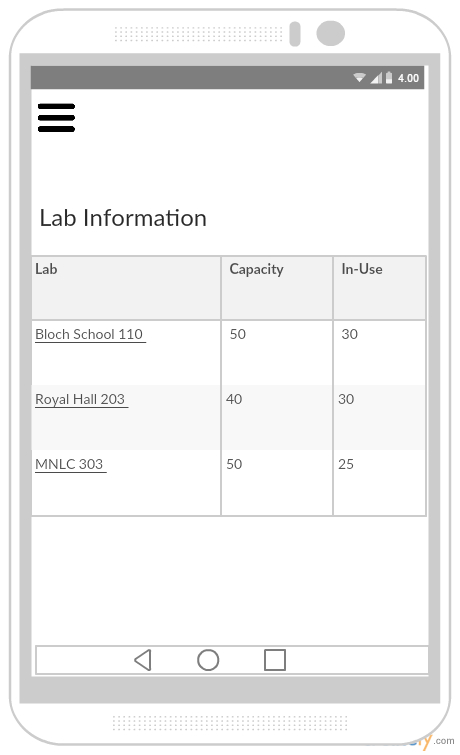
**Fig: Login screen**



**Fig: Main Home page**



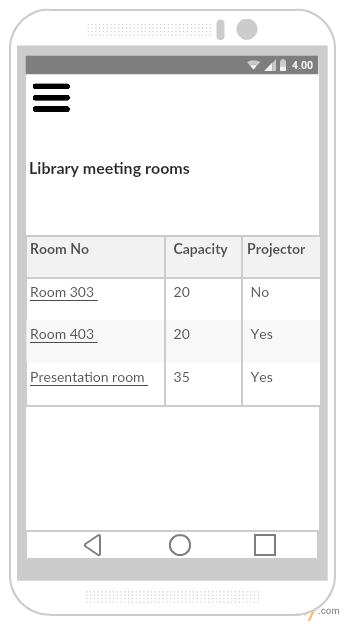
**Fig: Side menu bar**



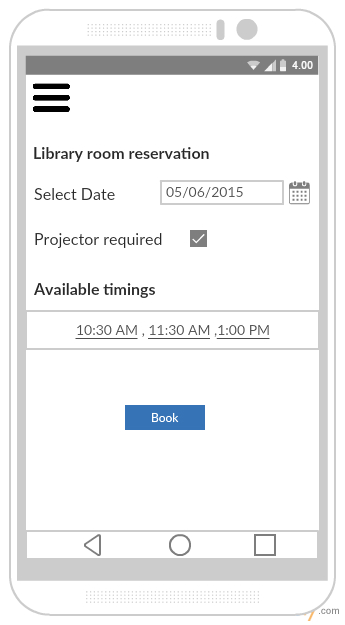
**Fig: Computer Labs information page**



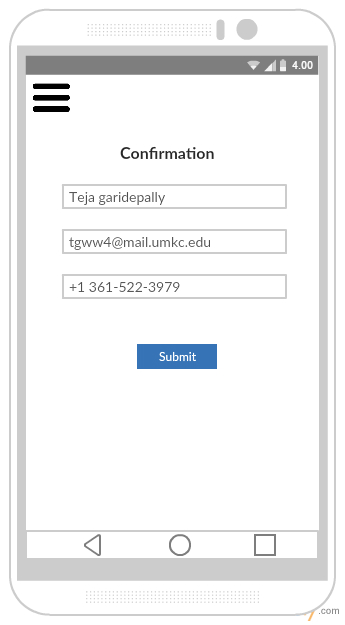
**Fig: Detailed information of a computer lab**



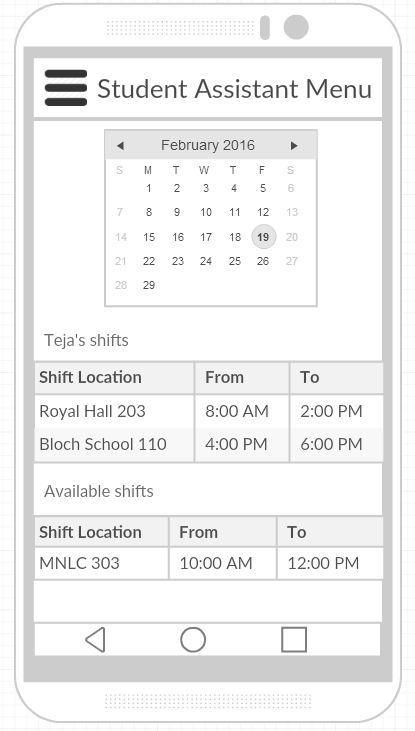
**Fig: Library study room information**



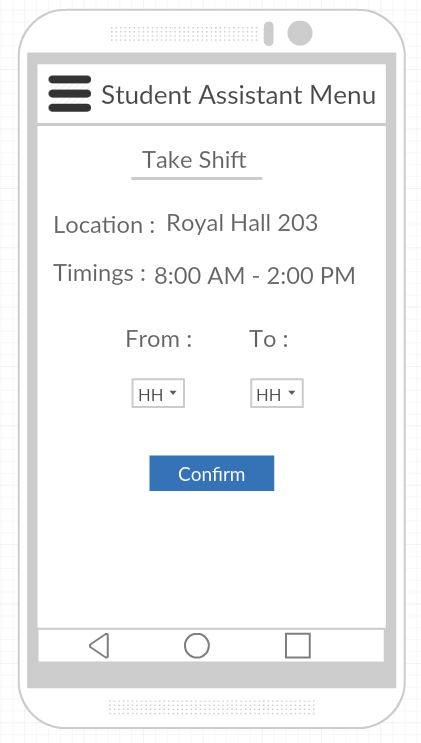
**Fig: Library room reservation page**



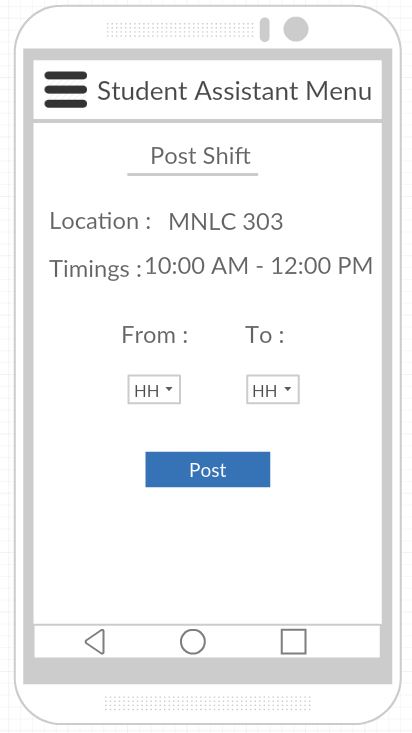
**Fig: Library room reservation form**



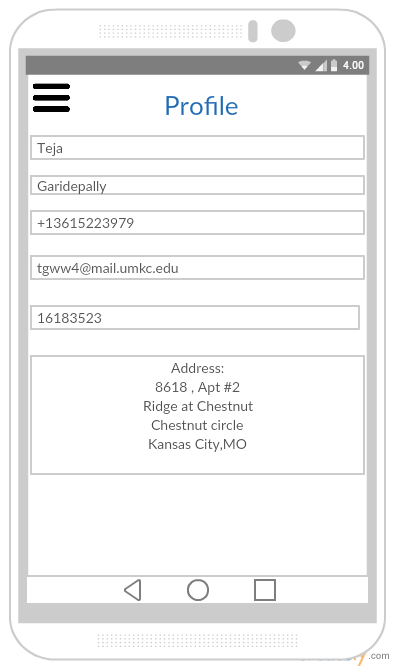
**Fig: Student Assistant menu page**



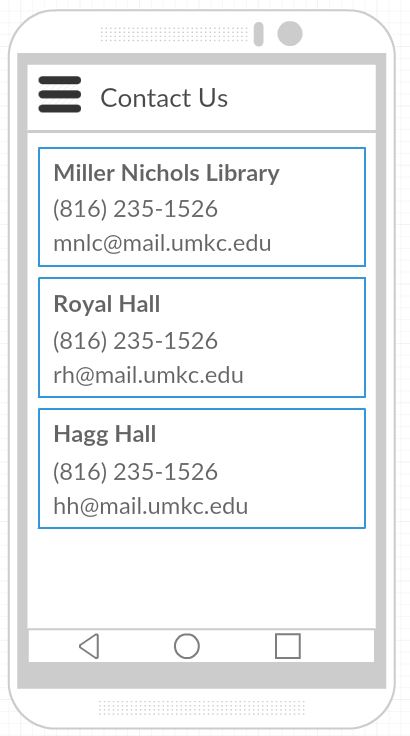
**Fig: Taking a Student Assistant shift**



**Fig: Posting a Student Assistant shift**



**Fig: Student profile page**



**Fig: Contact Us information page**

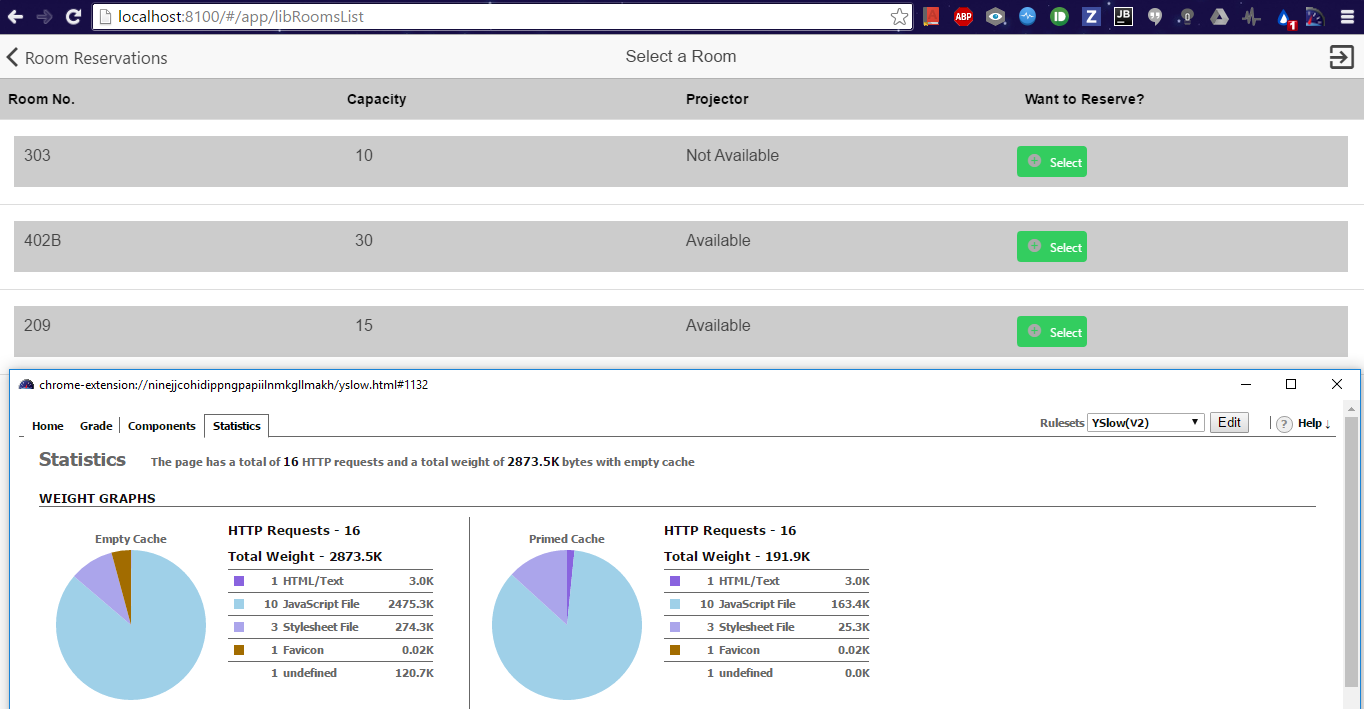
**User Stories:**

* As a User, I can see the UI of StudentCompanion App.
* As a user, I can see the Login Screen.
* As a user, I can enter my Details and Login to the App.
* As a user, I can see the Side menu bar on Home Page with side menu options like Profile, Lab Information, Reserve Study Room, and SA Menu.
* As a User, I can see my Profile.
* As a User, I can see the Lab Information of specified lab.
* As a User, I can Reserve Study Room.
* As a User, I can choose the SA Menu option
* As a User, I can utilize the options in SA Menu.

As a user, I can Sign-out from the Application.

# Testing

We’ve used *YSlow* - a Google Chrome extension to test the performance of our application. It shows the requests and their weights in a graphical manner. This helped us to tweak our code to improve the application overall performance.

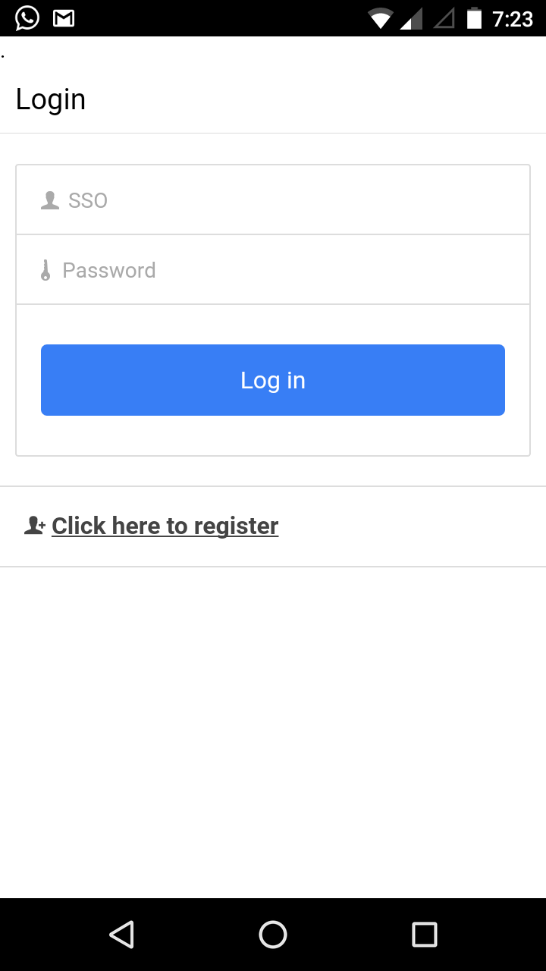
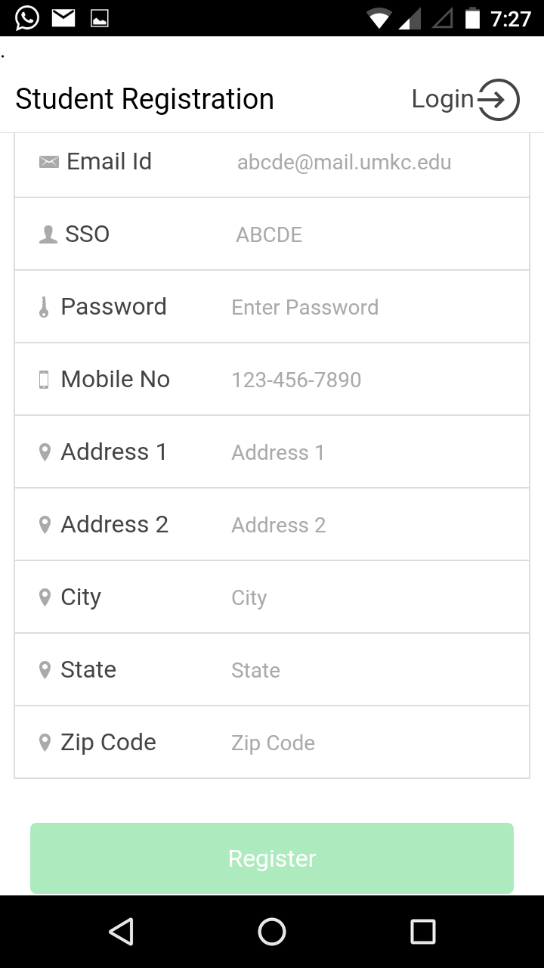


**Fig: YSlow statistic of “Room Reservation” page**

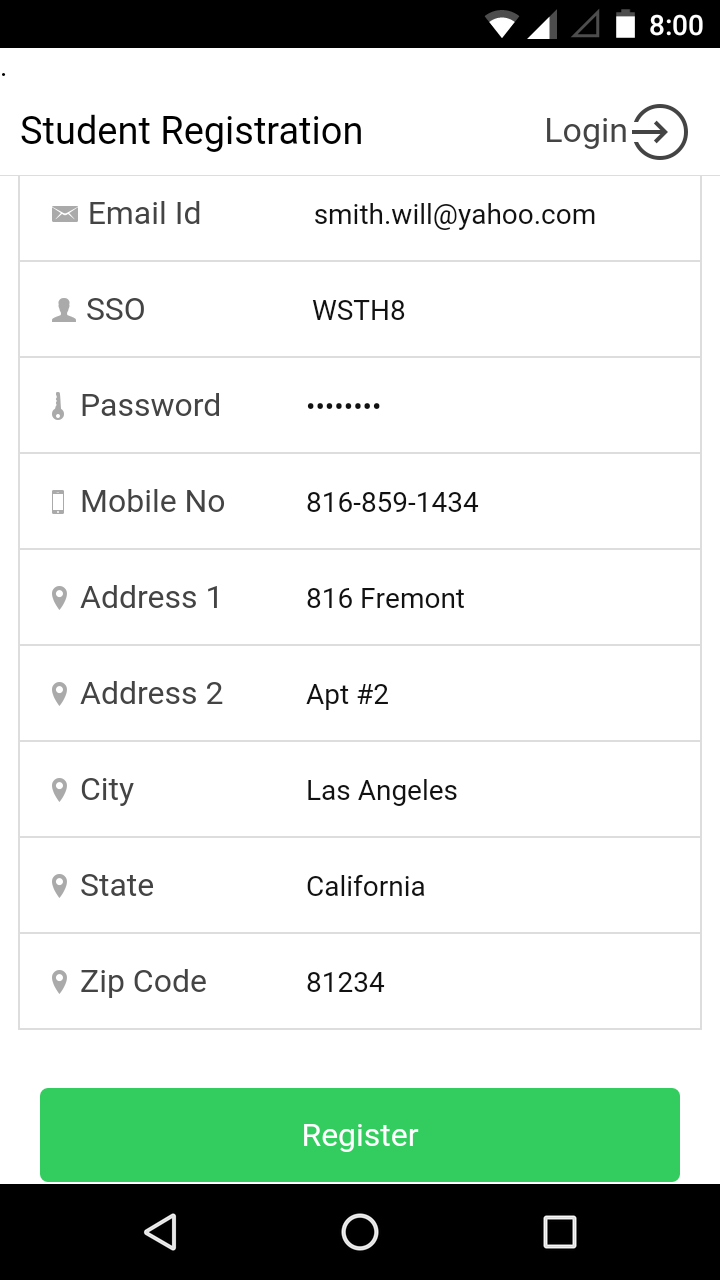
# Deployment

Deployment of application to Mobile phone.

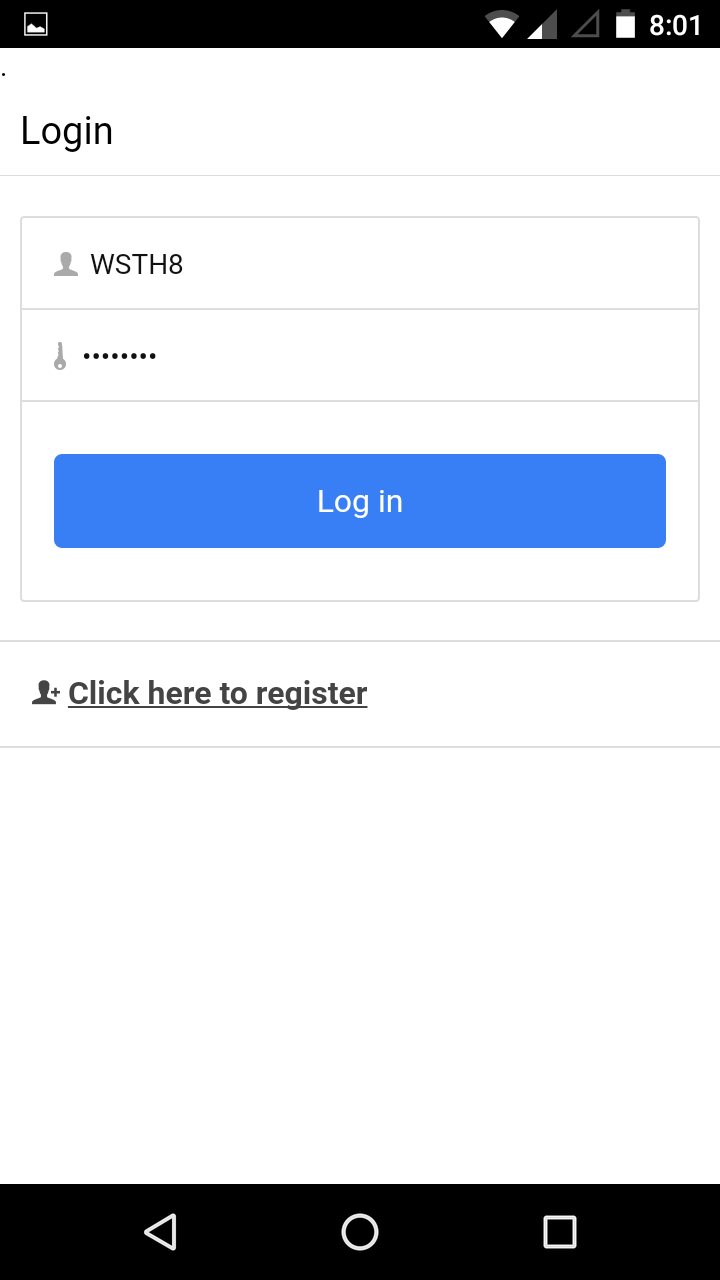
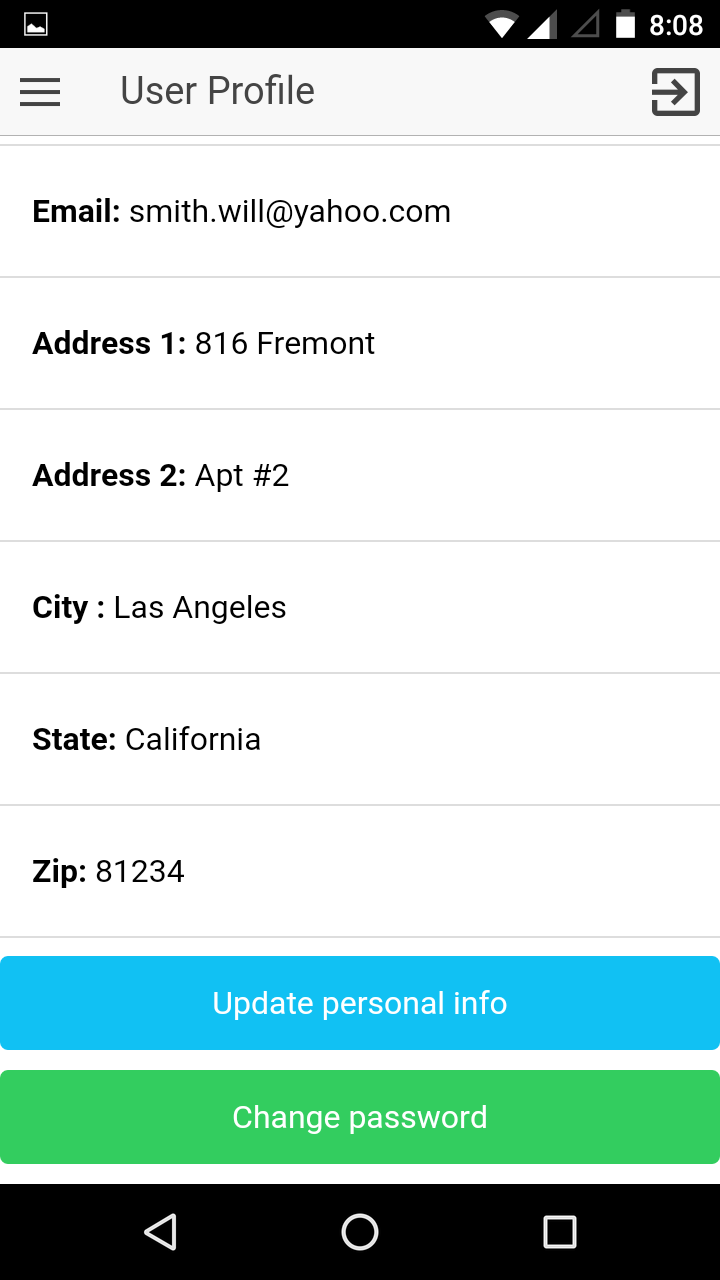
Screen Shots:

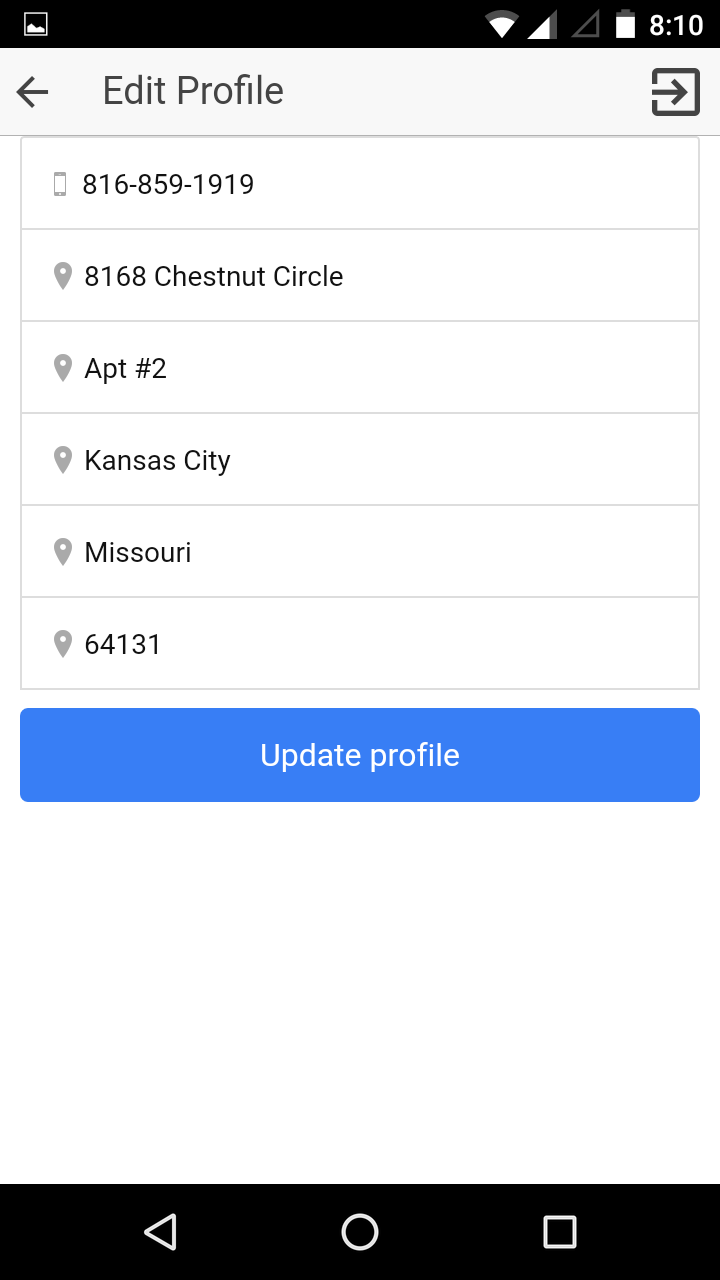
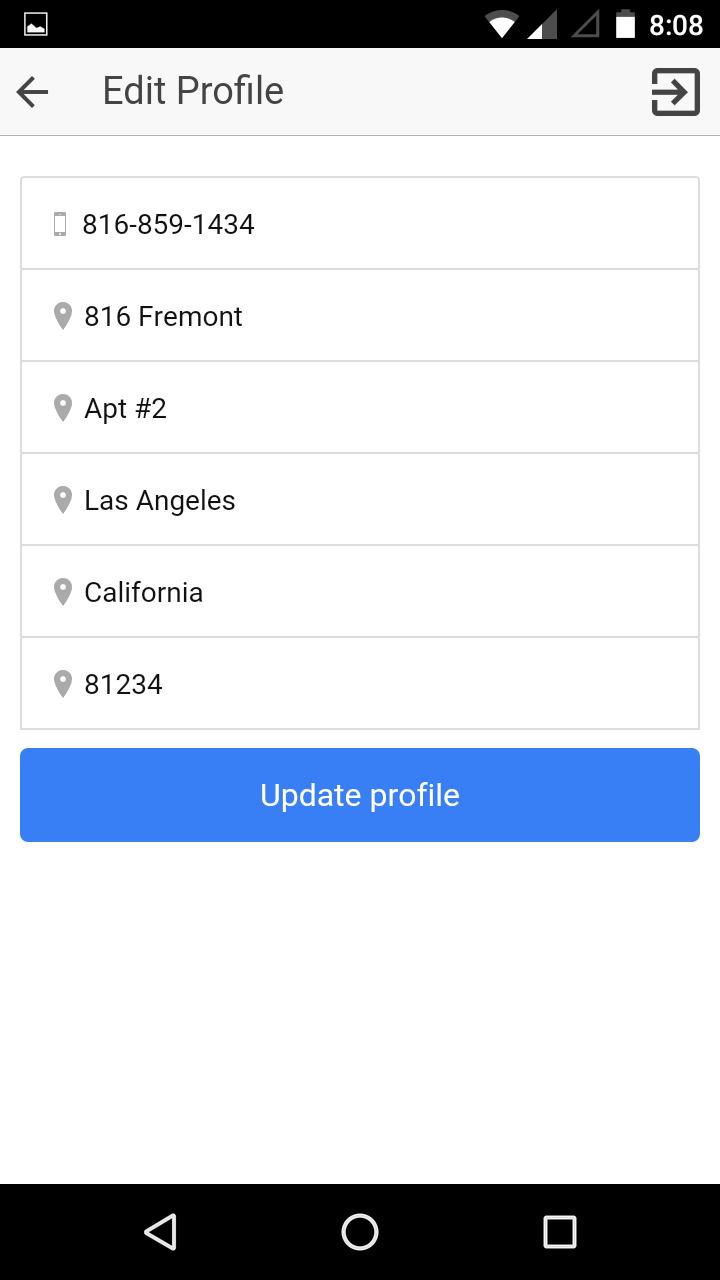
**Fig: Login Fig: Register User**

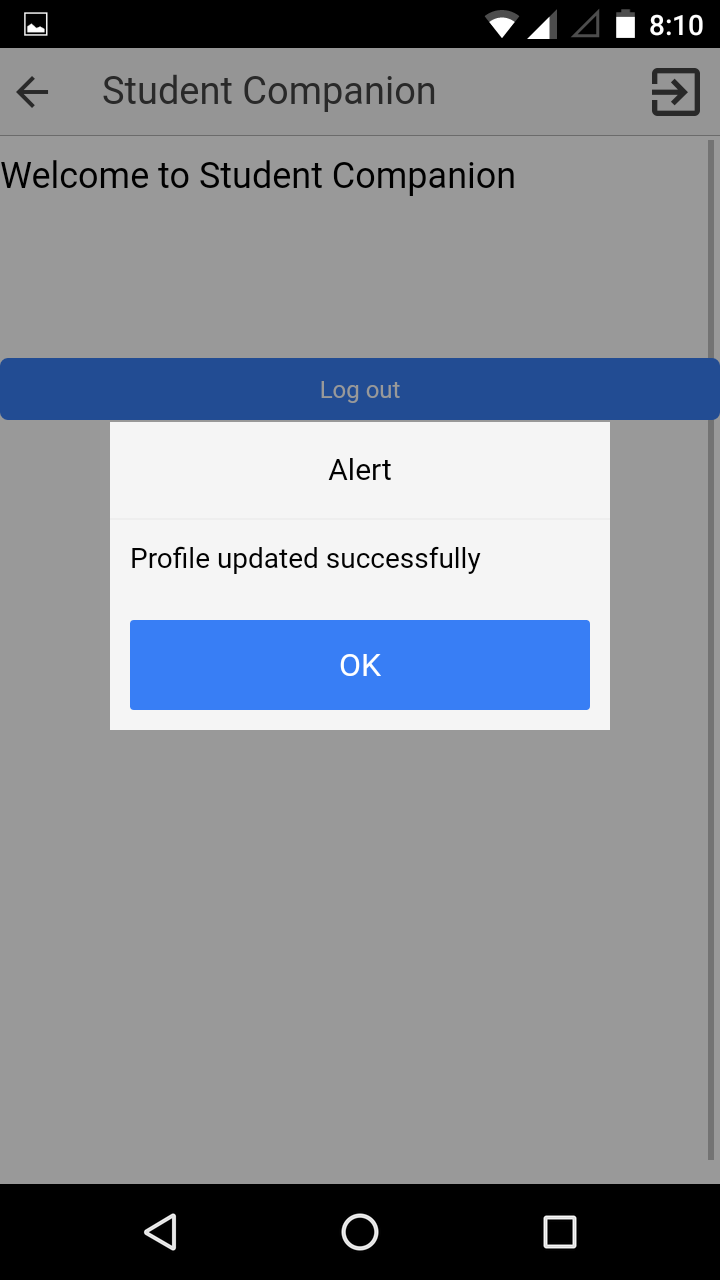
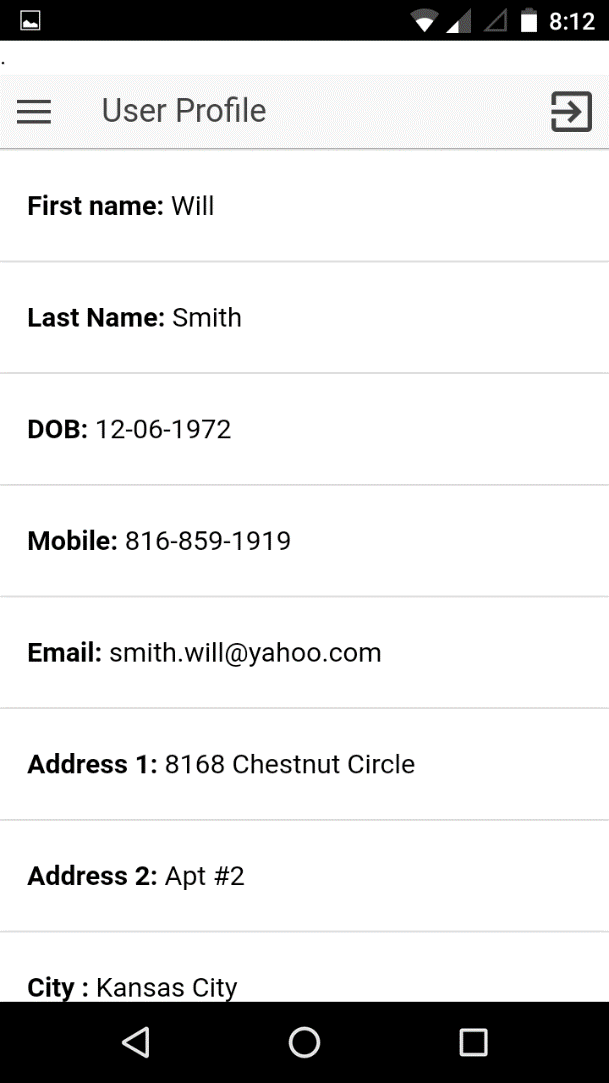
**Fig: User Details Fig: User Registered**

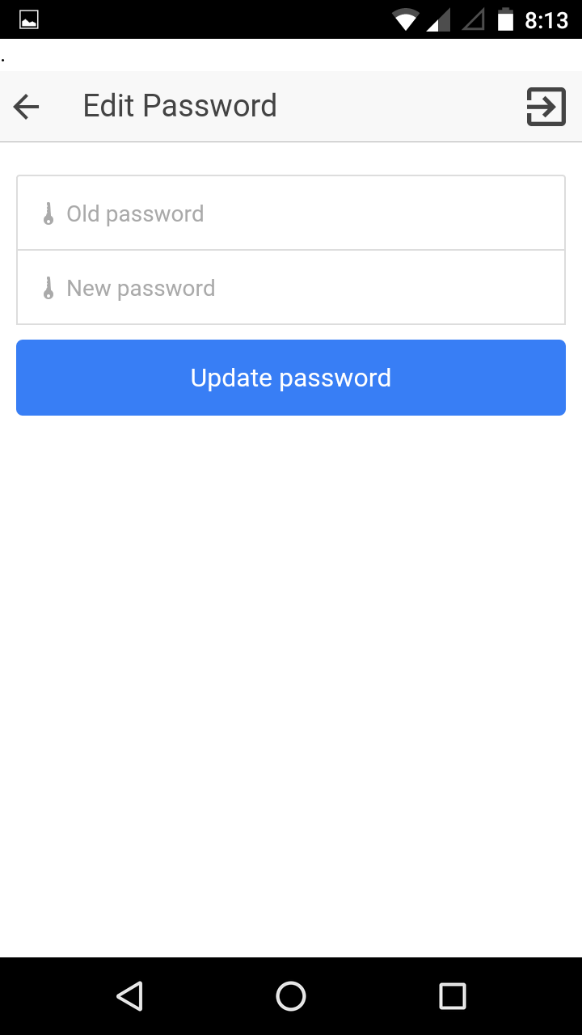
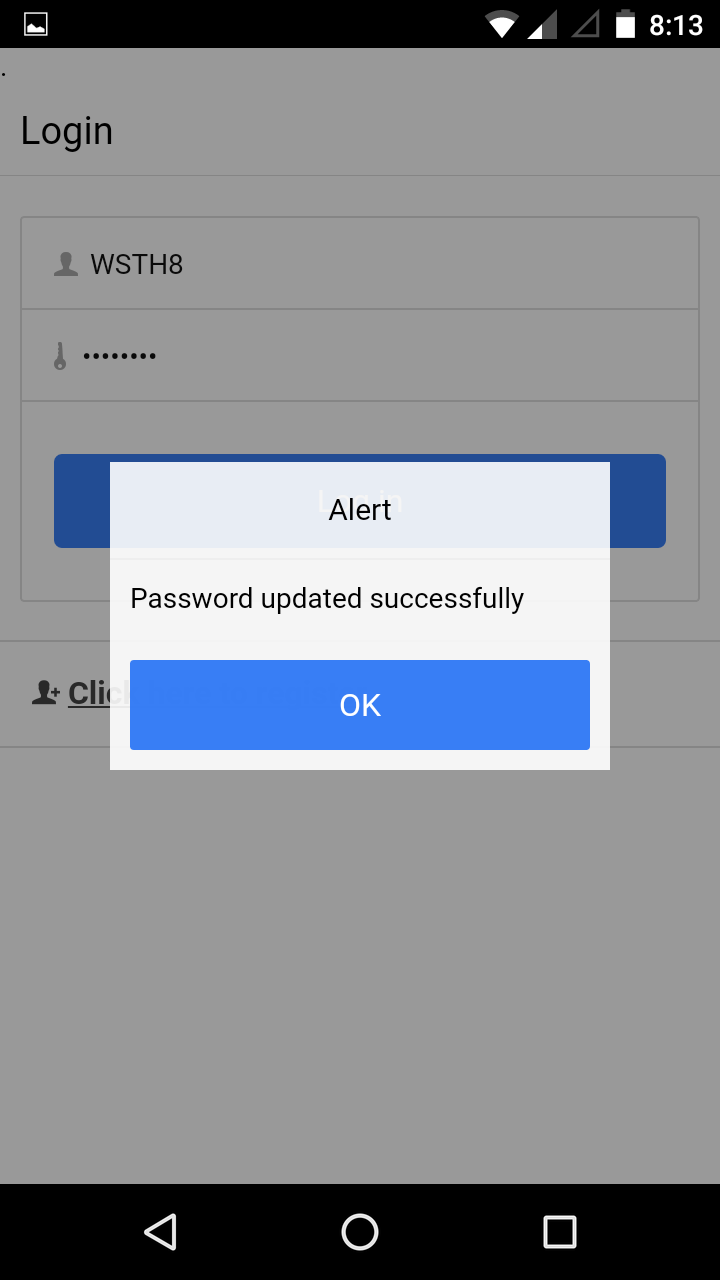
**Fig: Login Success Fig: User Profile**

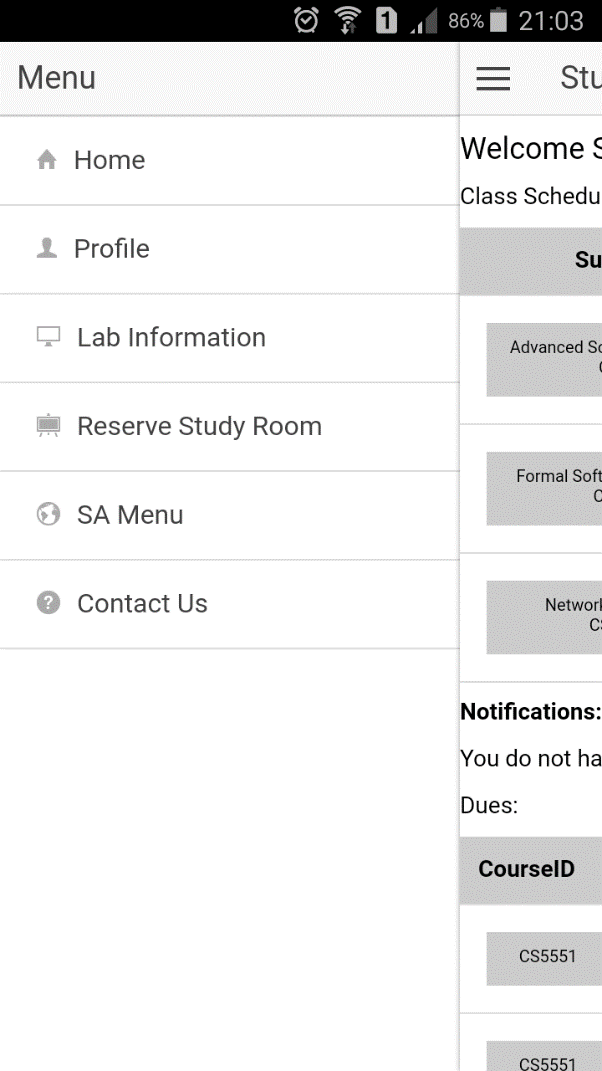
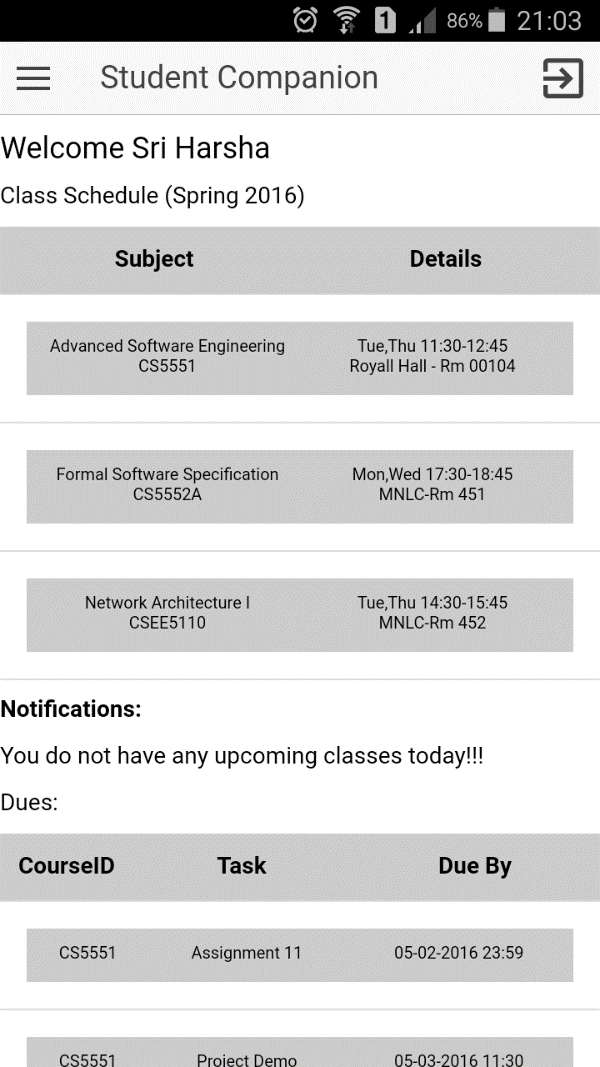
**Fig: Initial Details Fig: Updated Fields**

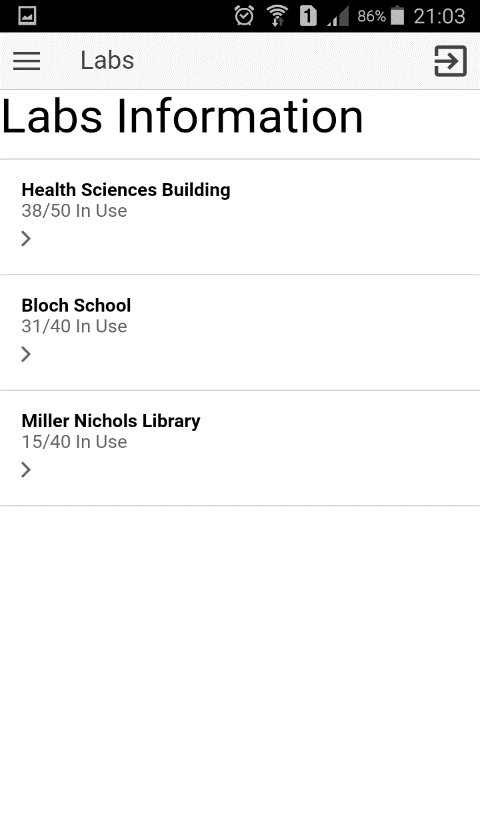
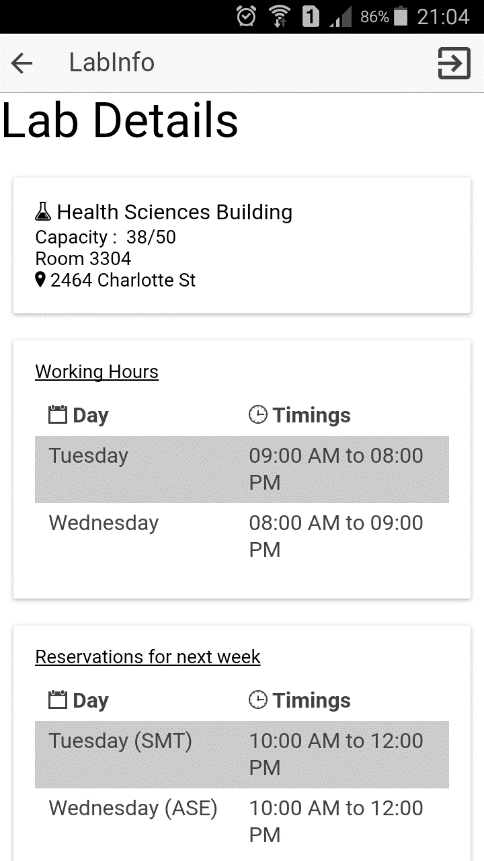
**Fig: Profile Updated Fig: Updated Profile**

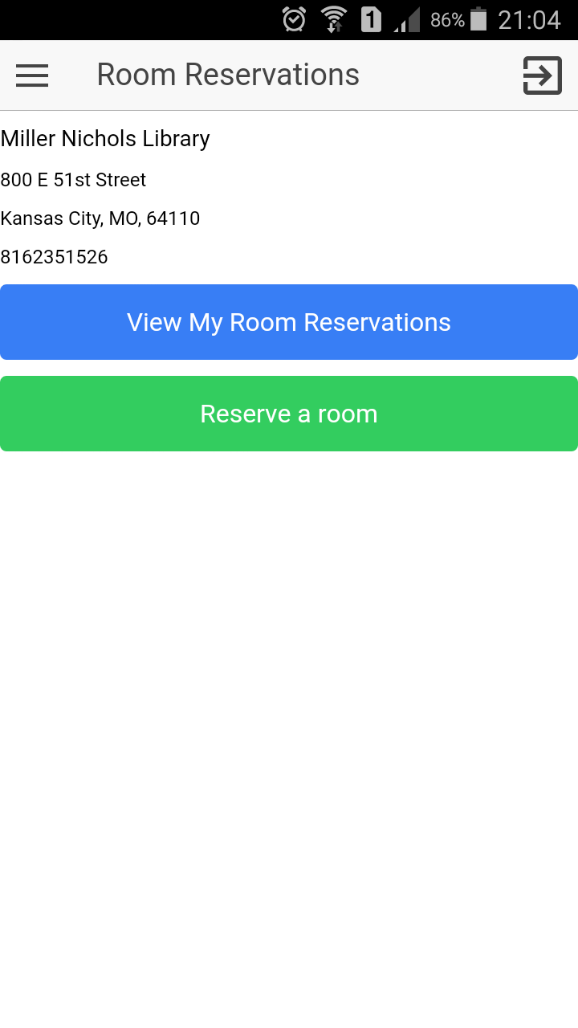
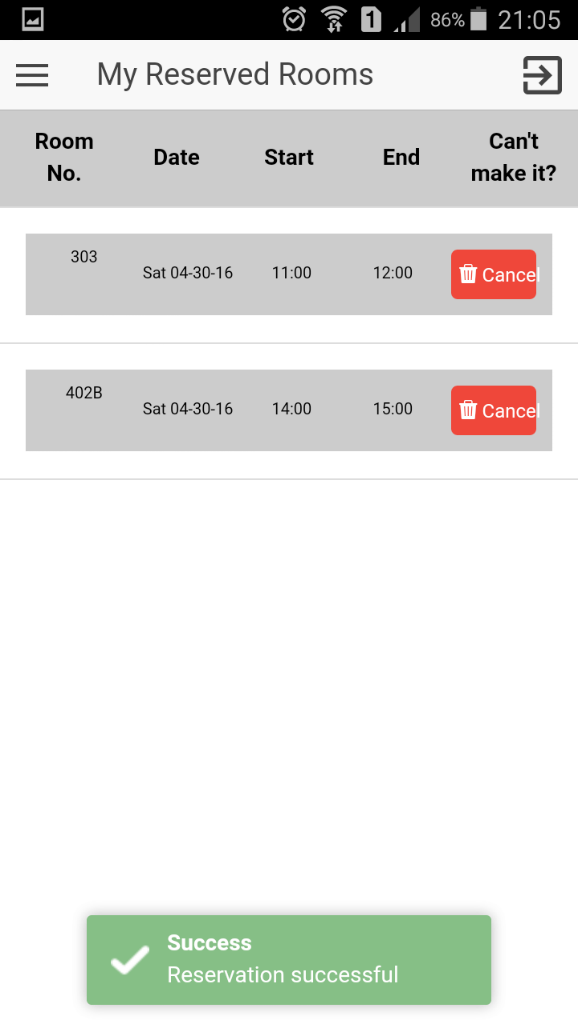
**Fig Edit Password Fig: Password Updated**

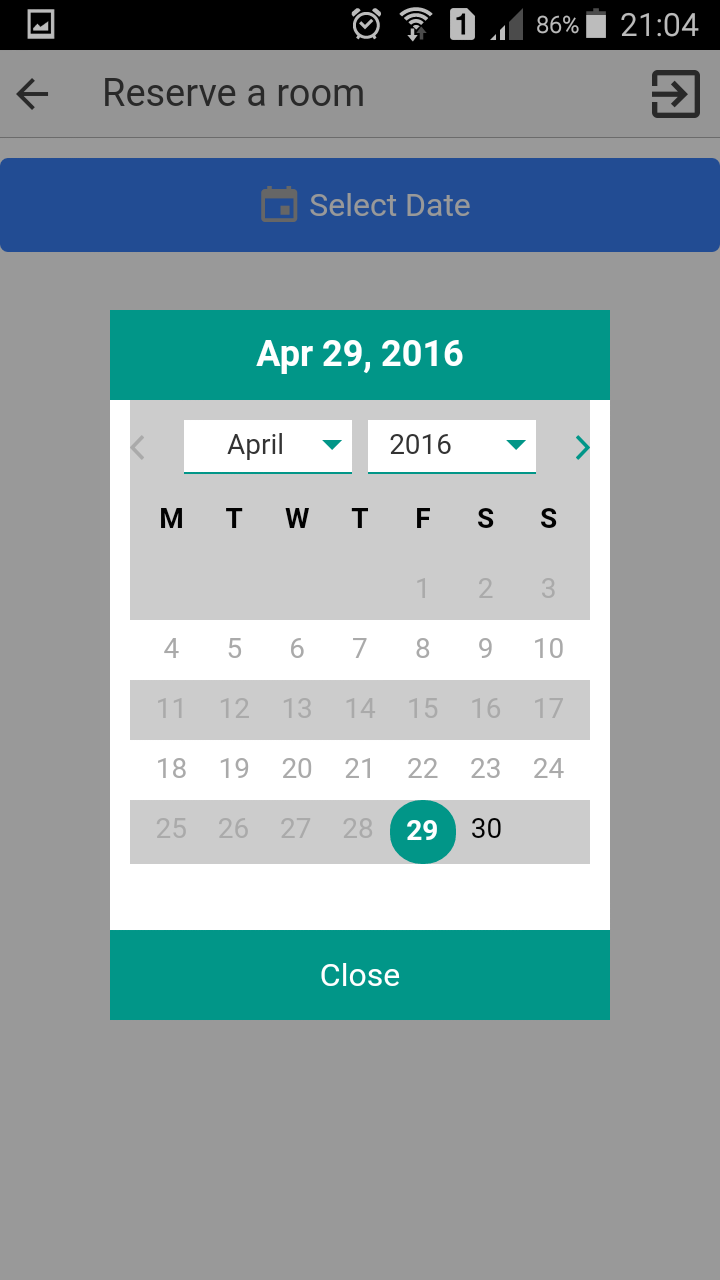
**Fig: Side Menu Fig: Home Page**

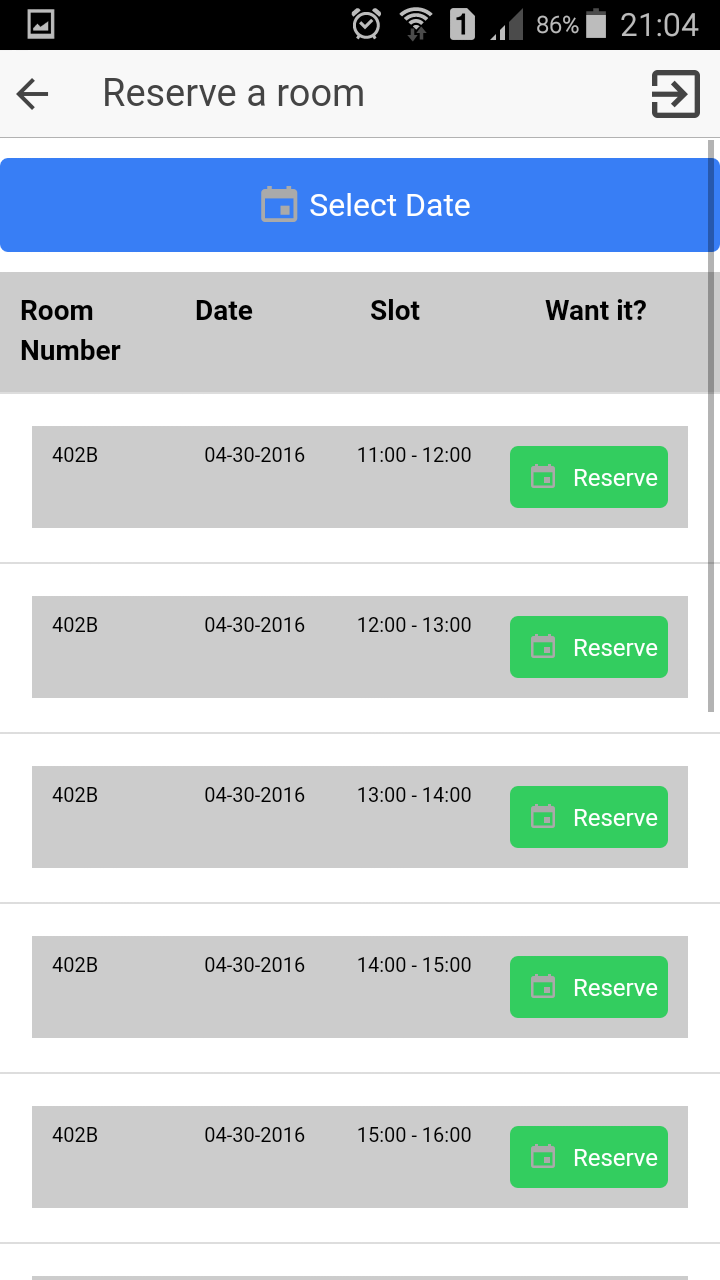
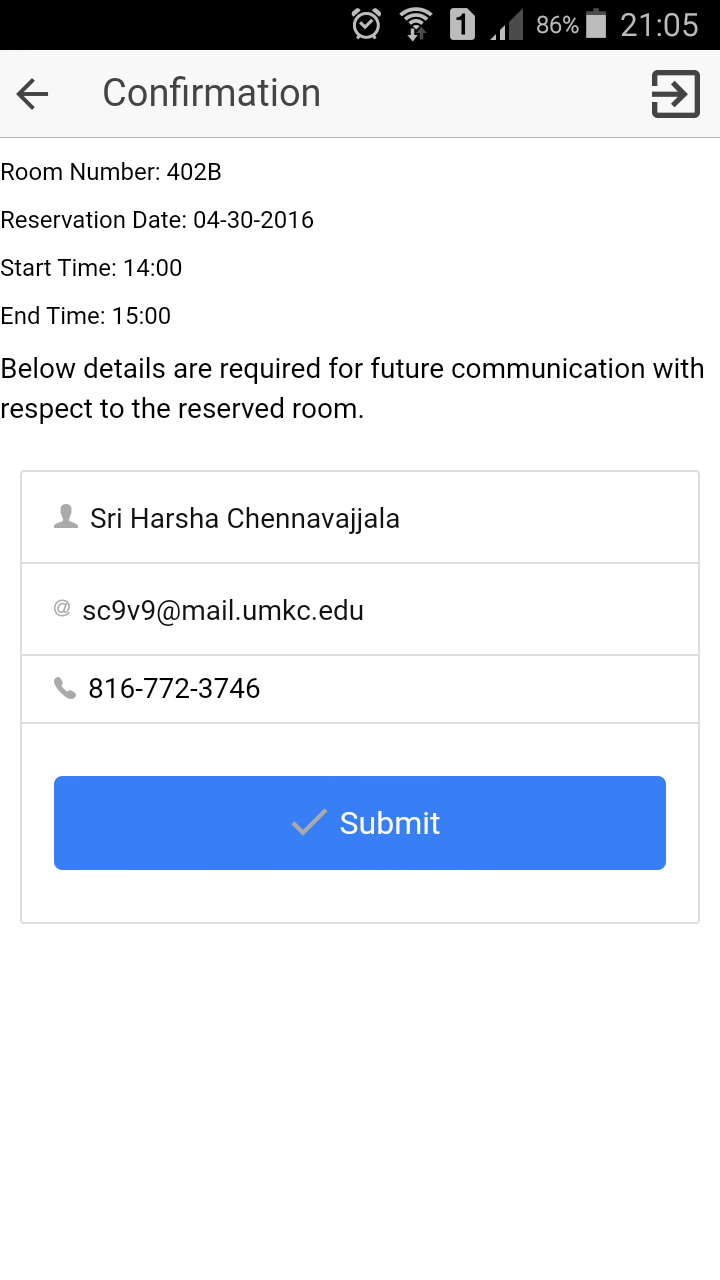
**Fig: Lab Information Fig: Lab Details**

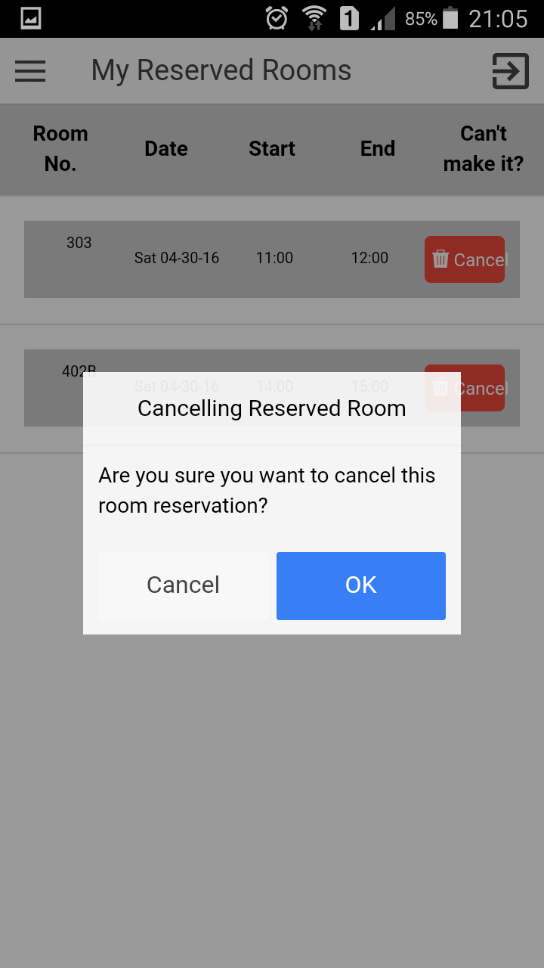
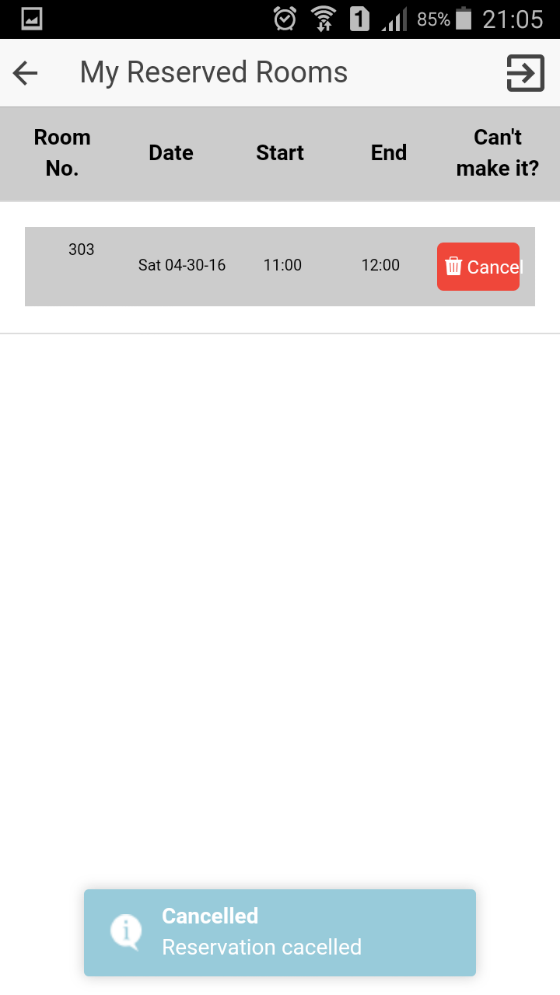
**Fig: Room Reservation Fig: View My Reservations**

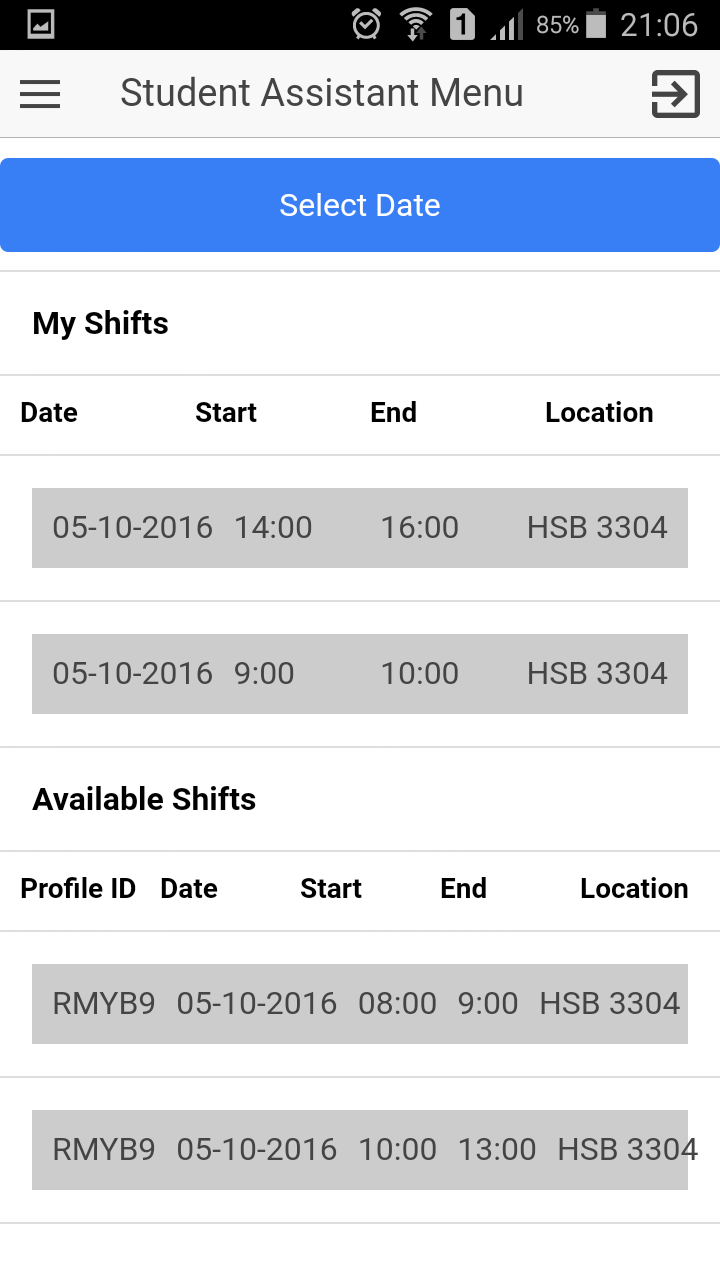
**Fig: Select a Room Fig: Select Date**

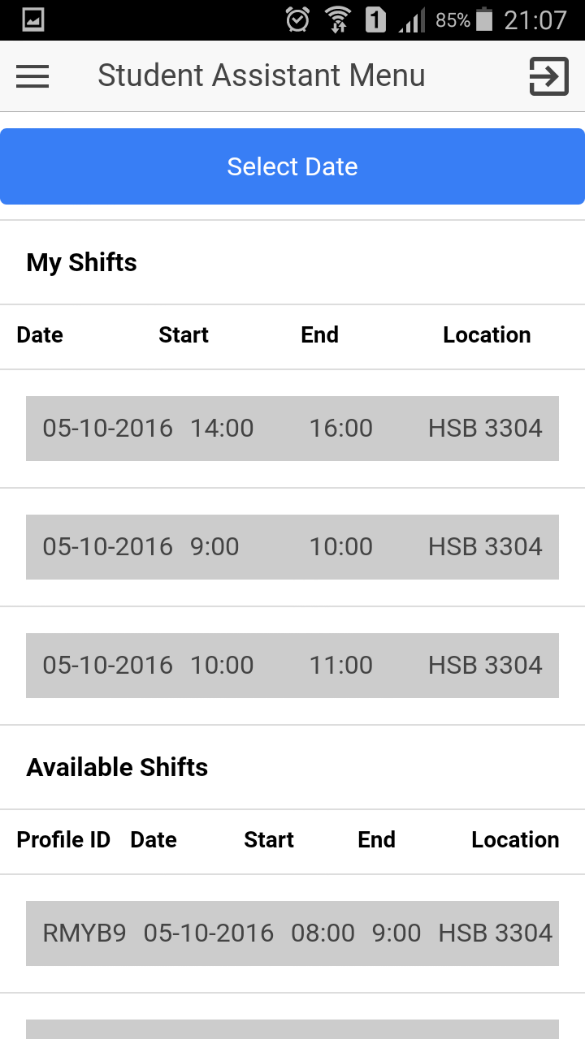
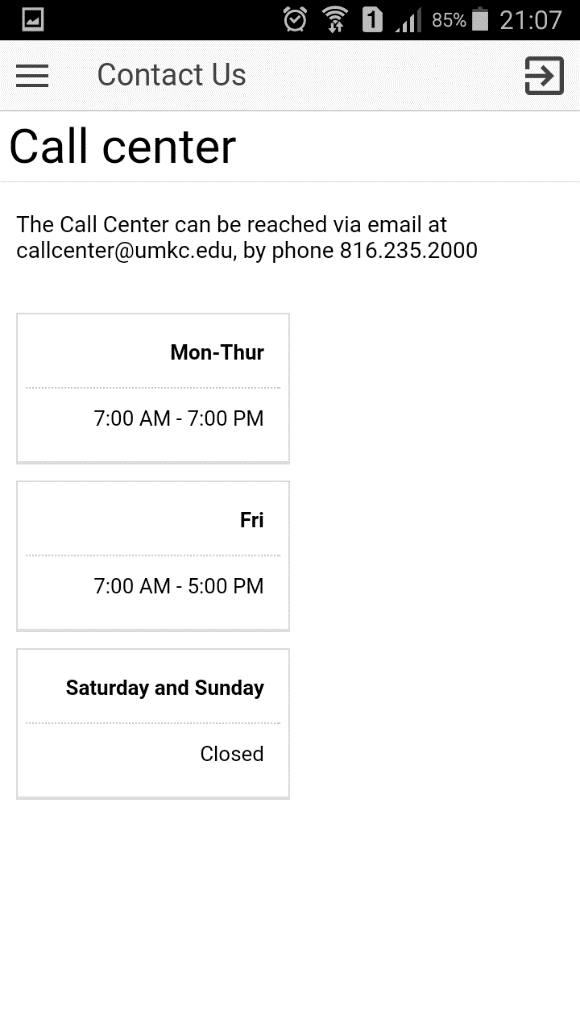
**Fig: Select a Reservation Slot Fig: Room Reserve Confirmation**

**Fig: Room Cancellation Fig: Room Cancellation**

**Fig: View Shifts Fig: Take Shift**

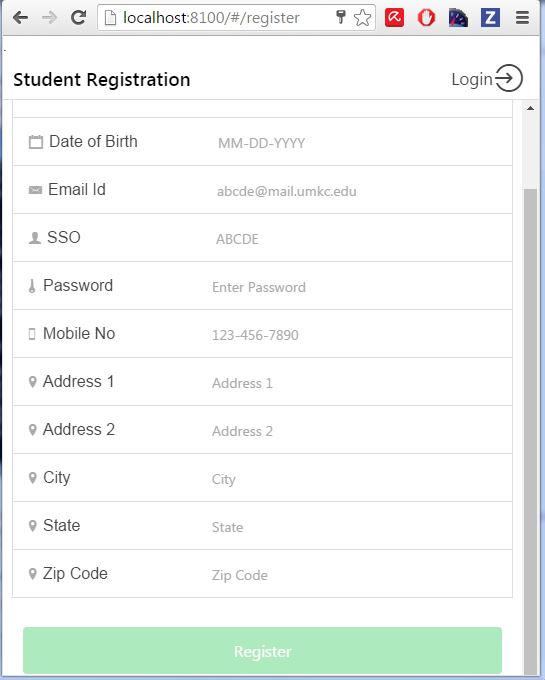
 

**Fig: After Take Shift Fig: Contact Us**

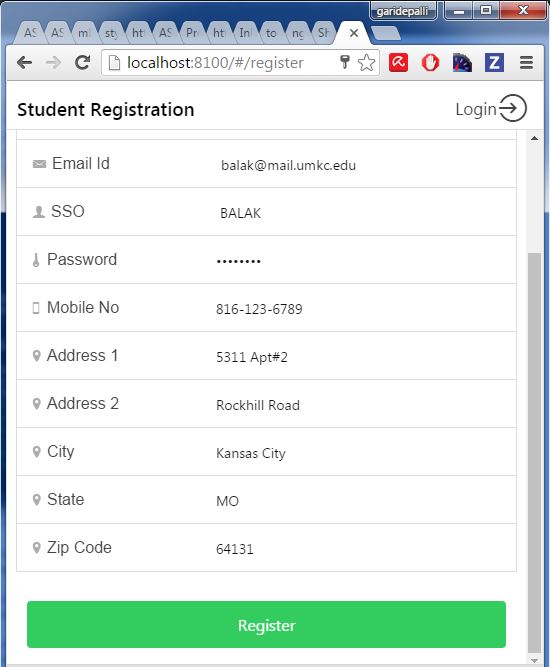
**Web Application Views:**



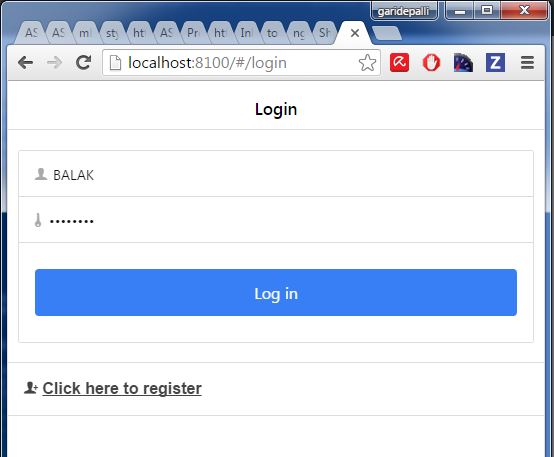
**Fig: Login**



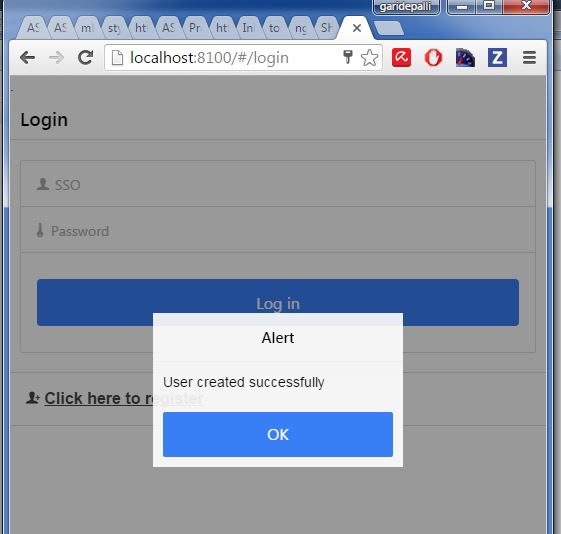
**Fig: Registration**



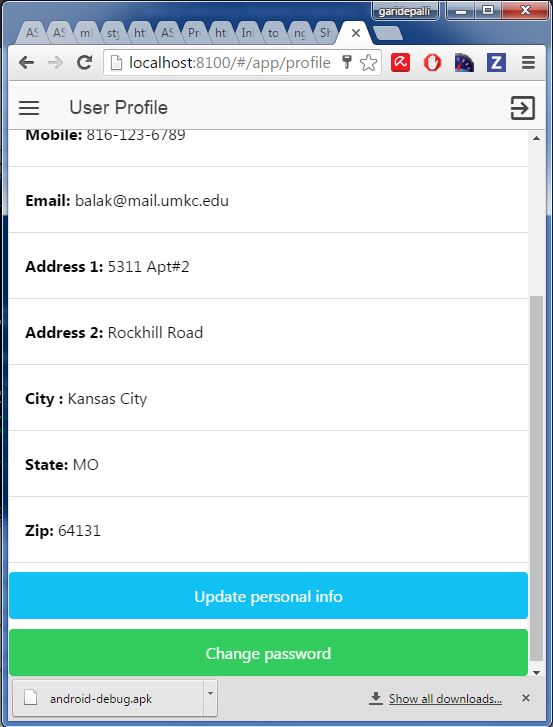
**Fig: Student Registration**



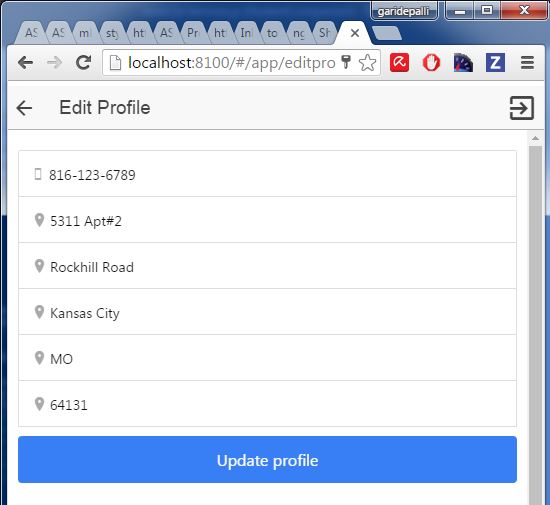
**Fig: Registered User Login**



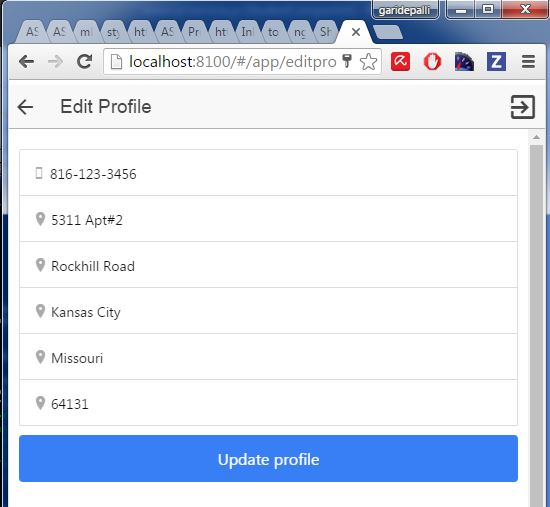
**Fig: Login Success**



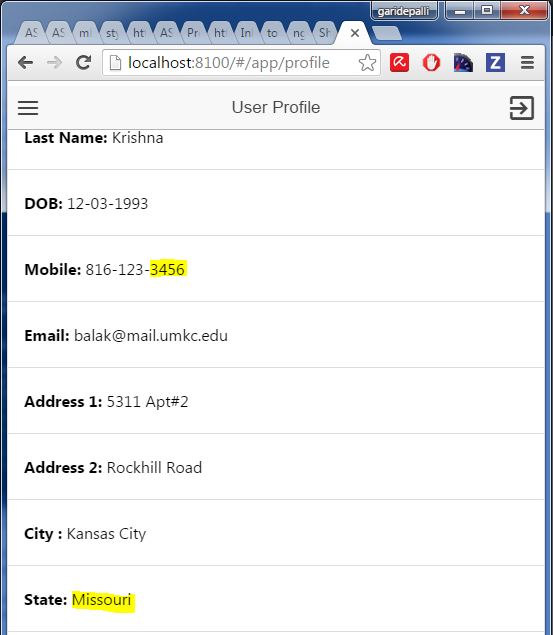
**Fig: View Profile**



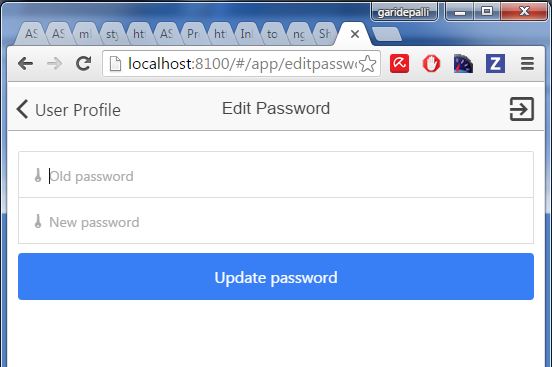
**Fig: Edit Profile**



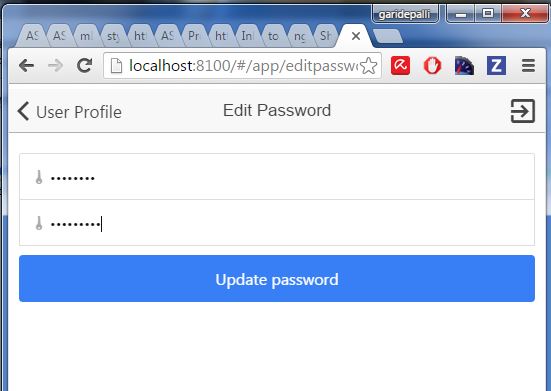
**Fig: Profile Updated**



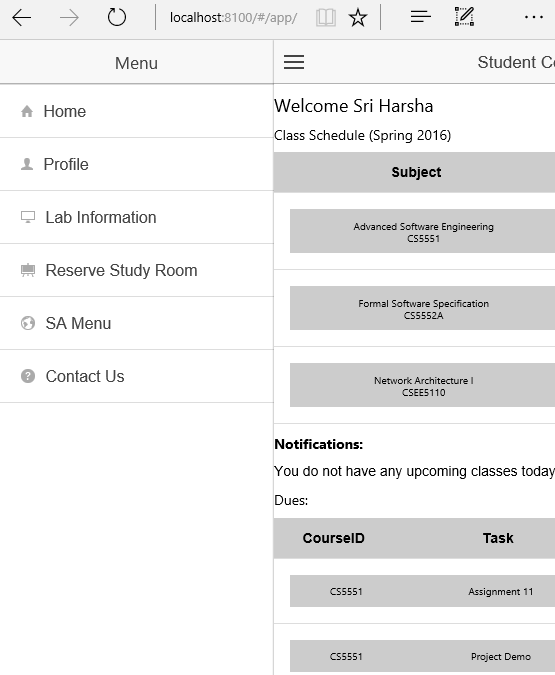
**Fig: Updated View**



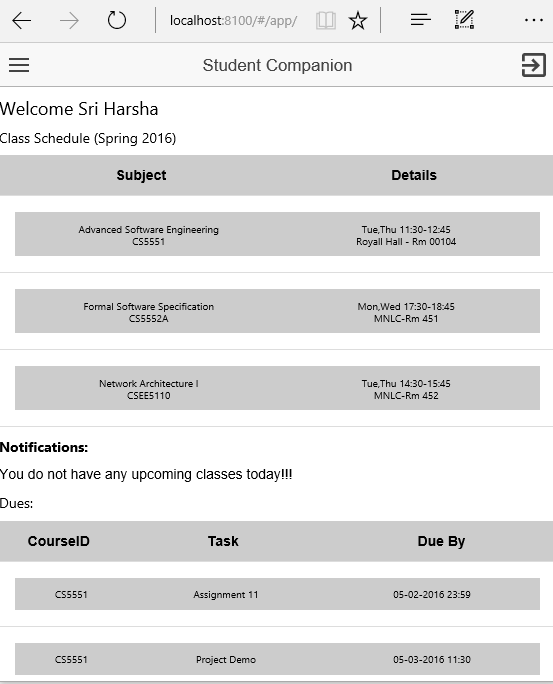
**Fig: Edit Password**



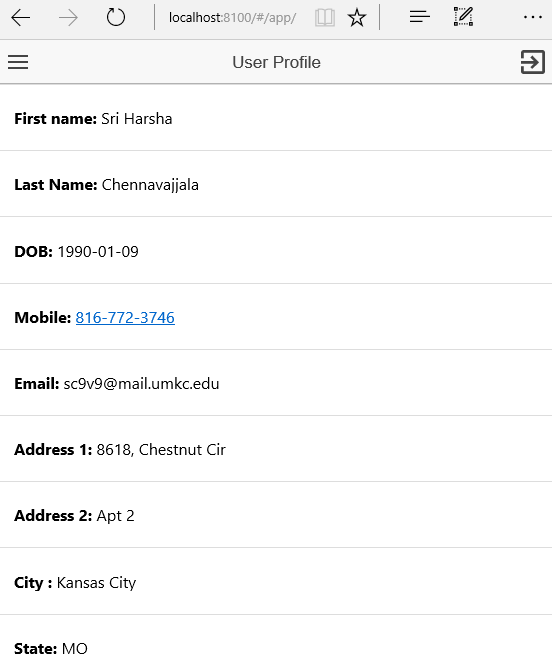
**Fig: Updated Password**



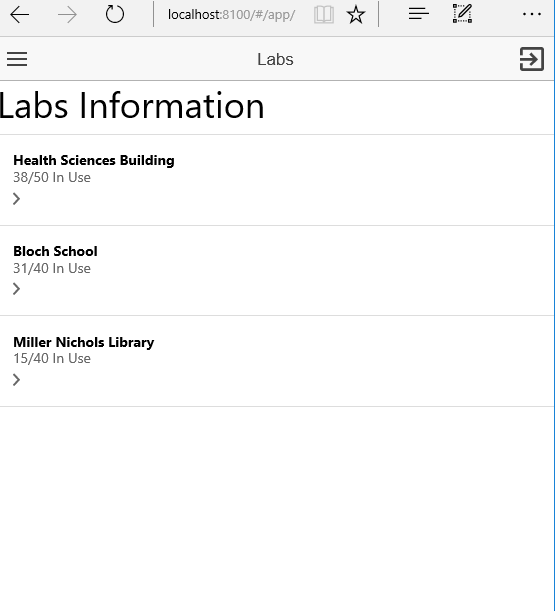
**Fig: Side Menu**



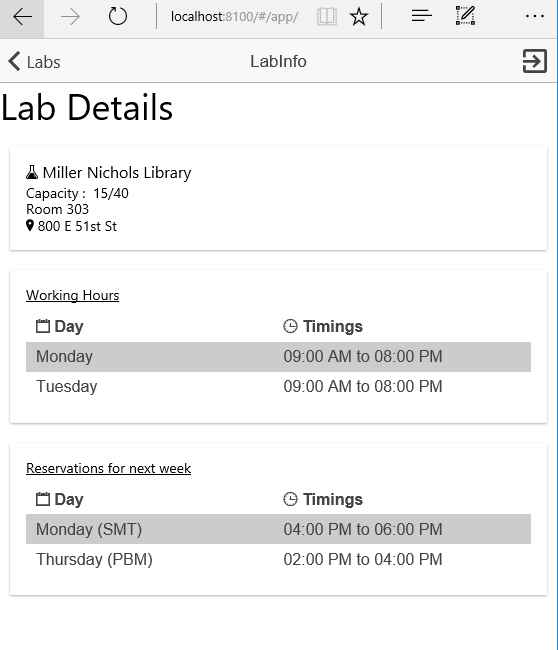
**Fig: Home Page**



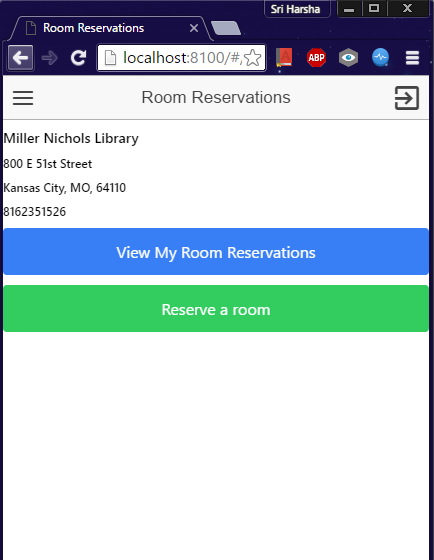
**Fig: Profile**



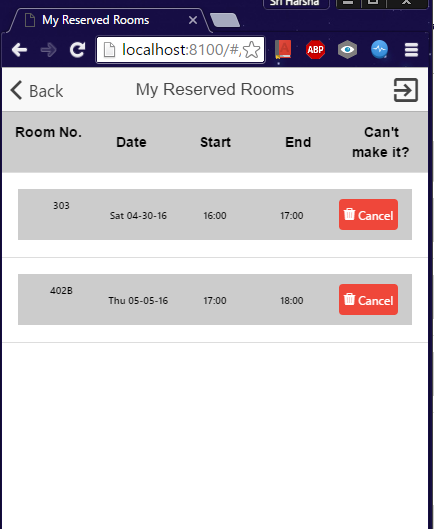
**Fig: Lab Information**



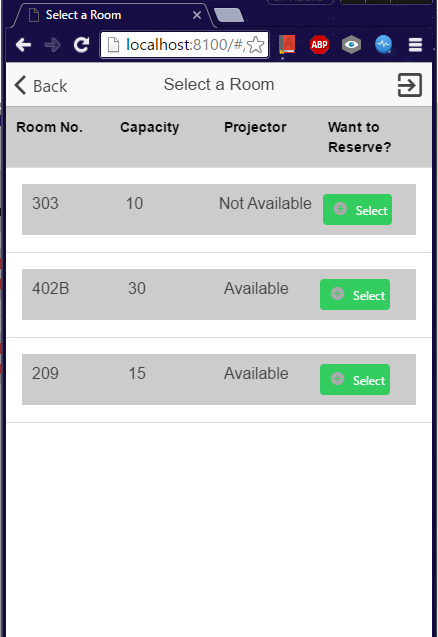
**Fig: Lab Details**



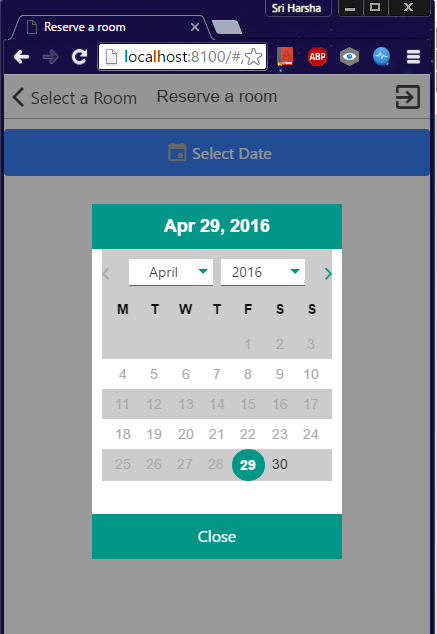
**Fig: Room Reservation**



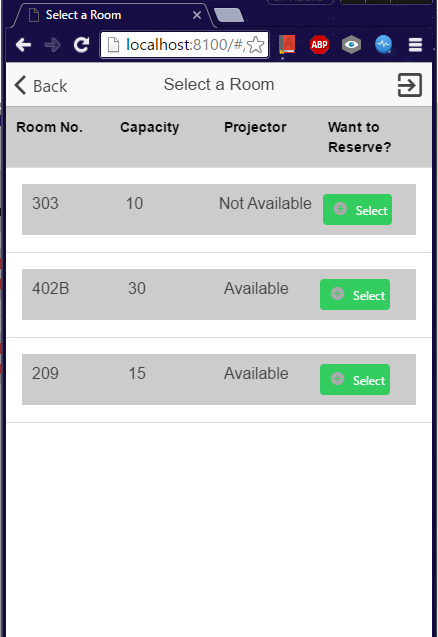
**Fig: View Reserved Rooms**



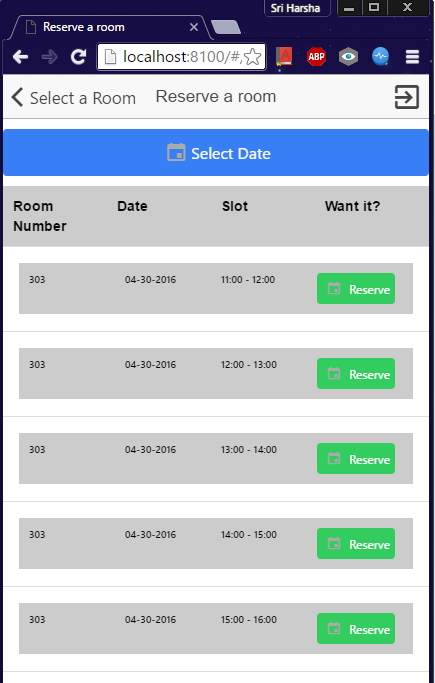
**Fig: Available Rooms**



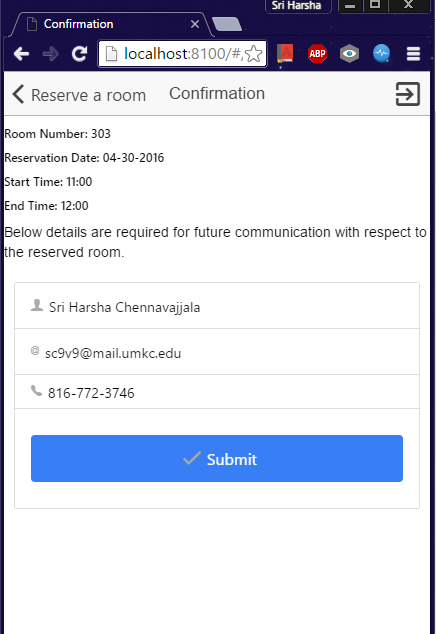
**Fig: Select Date**



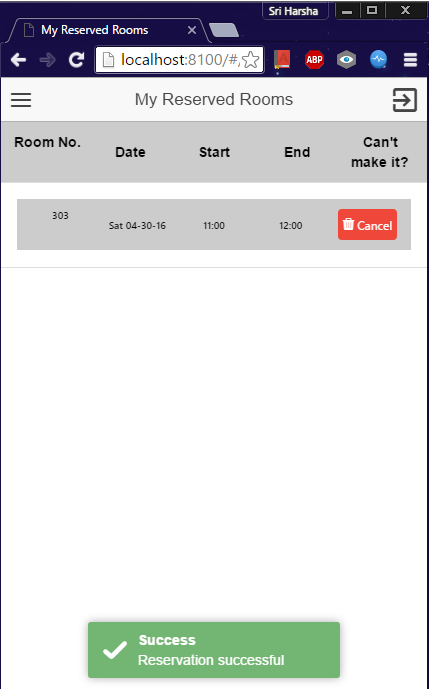
**Fig: Available Rooms**



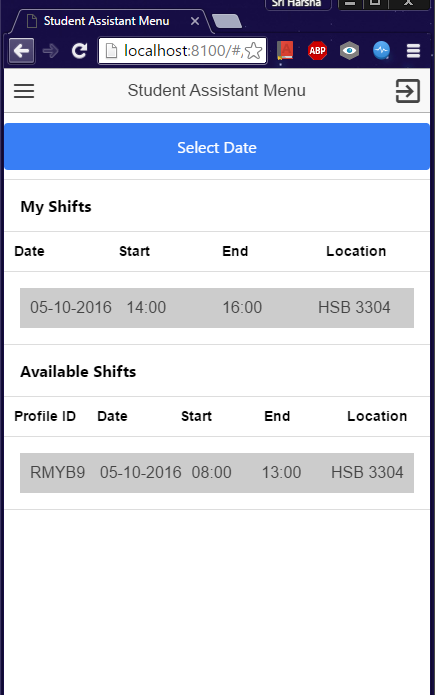
**Fig: Available Slots**



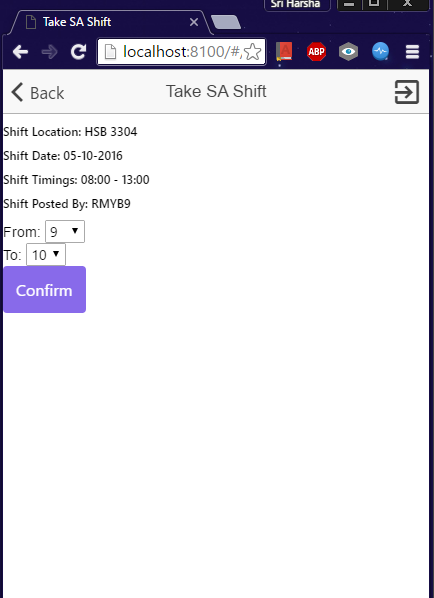
**Fig: Confirm Reservation**



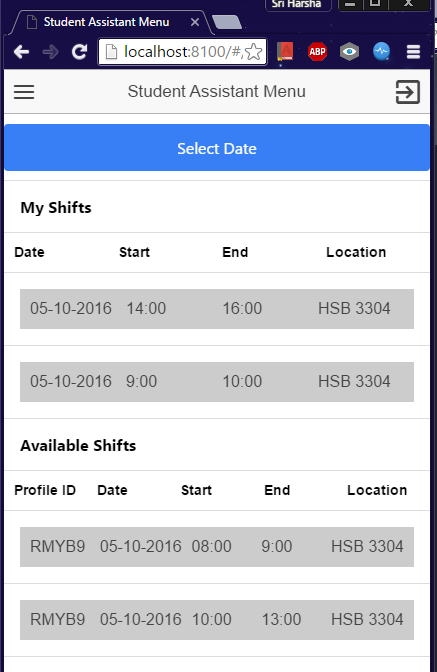
**Fig: Room Confirmation Toast**



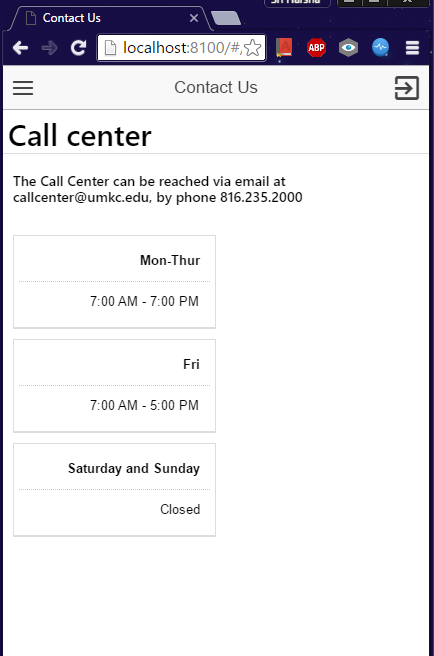
**Fig: SA Menu**



**Fig: Take Shift**



**Fig: After Take Shift**



**Fig: Contact Us**

**Project URL:**

<https://github.com/meetsriharsha/ASE_S16_G7/>

**Document URL:**

<https://github.com/meetsriharsha/ASE_S16_G7/Documentation>

# Project Management

## Work Completed

In detailed analysis of the system, environment and technical requirements for the application development. Project proposal documentation, Project tasks created in ZenHub and assigned the tasks to team members. Created the Project increment 4 document. All team members are involved in this task. Total time taken is 30 hours per person. Contributions: Harsha 30%, Teja 25%, Raj 25%, Suhas 20%.

Limitations:

The data which we worked on is a static database so the test cases that we performed on the application may not be accurate or sufficient. As we used ionic frame work for the development of application we have some limitations on the core functionalities of the mobile phones.

## Issues/Concerns:

Sometimes we are getting black background for the login page due to some unknown big in ionic framework .Sometimes we are getting asynchronous results from API in Home page.

Burndown Chart for Increment 4:

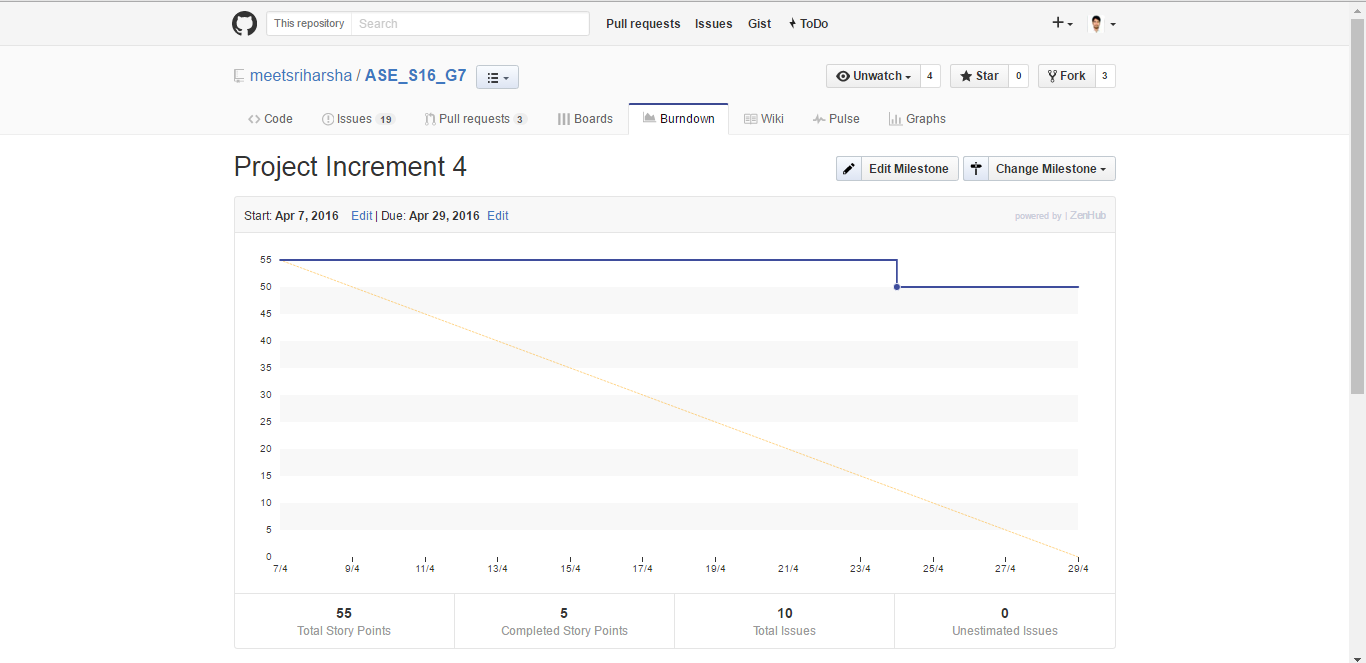


Fig: Burndown Chart

# Bibliography

* + University of Missouri Library -<http://library.umkc.edu/>
  + UMKC IS Labs - <http://www.umkc.edu/is/labs>
  + Top Library management android applications - [http://appcrawlr.com/androidapps/best-apps-library-manager](http://appcrawlr.com/android-apps/best-apps-library-manager)
  + SFU: Student Companion - <https://play.google.com/store/apps/details?id=com.teamzeta.sfu&hl=en>
  + ZenHub - Agile project management tool for GitHub - <https://www.zenhub.io>
  + Ionic Framework Documentation - <http://ionicframework.com/docs/>
  + AngularJS ng directives - <https://docs.angularjs.org/api/ng>
  + Bootstrap CSS - <http://getbootstrap.com/css/>
  + Learn AngularJS - <http://www.w3schools.com/angular/>
  + Ion Icons cheat sheet - <http://ionicons.com/cheatsheet.html>