ADVANCED SOFTWARE ENGINEERING

PROJECT: STUDENT ACCOMMODATION

REPORT ON INCREMENT-1

Submitted by:

Manusha Reddy

Sneha Lagandula

Dheeraj Reddy Bethi

Project Group: 11

IMPORT EXISTING SERVICES:

In the first increment, user interface for log-in form, registration form with validations has been created for student and administrator access. Hence, no existing web services are used so far in the first increment.

However, for the further increments, we are to implement the following APIs:

User Feedback API: http://api.sandbox.freelancer.com/User/getUserFeedback.{xml|json}

Geo Location API: https://developers.google.com/maps/documentation/business/geolocation/

Direction Service API: https://developers.google.com/maps/documentation/directions/

Facebook API: https://facebook.com/share/

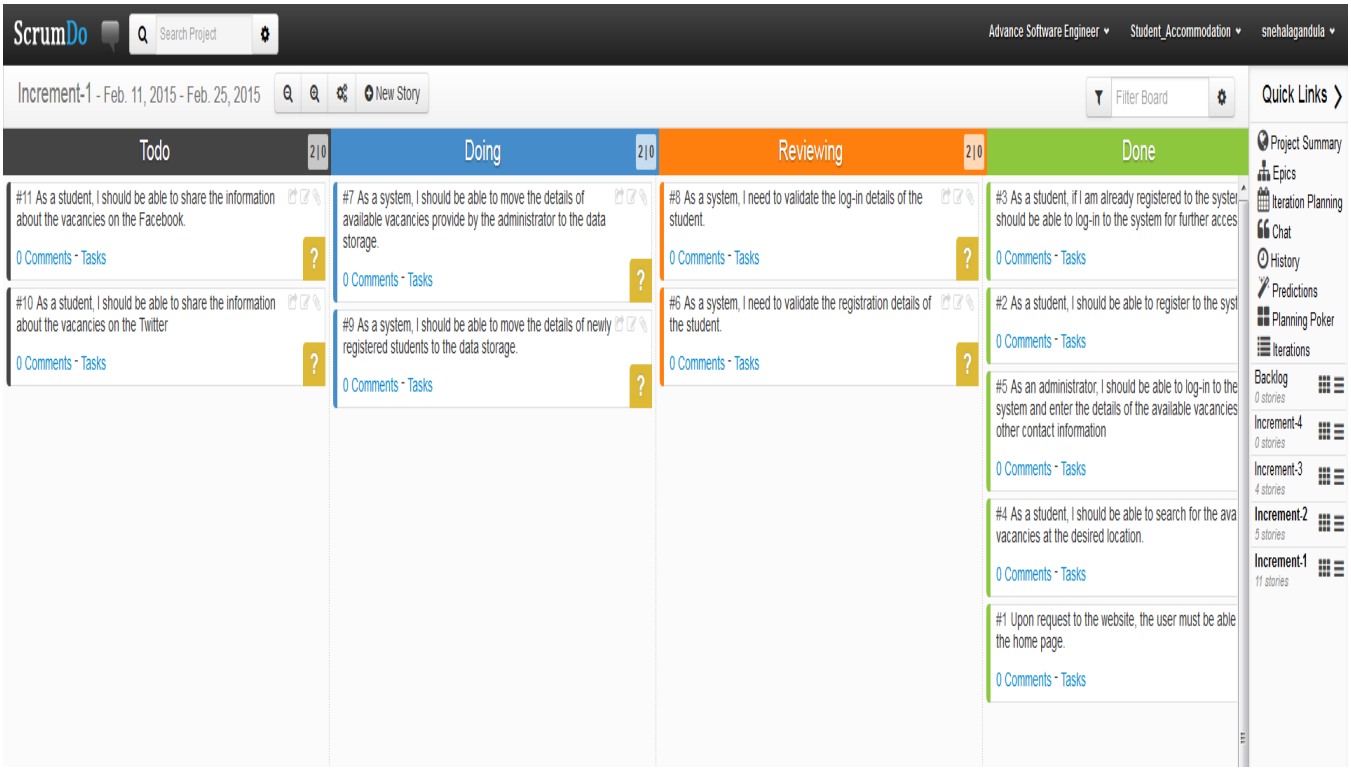
Twitter API: https://twitter.com/share/

Source code for the so far developed application is uploaded in the GitHub.

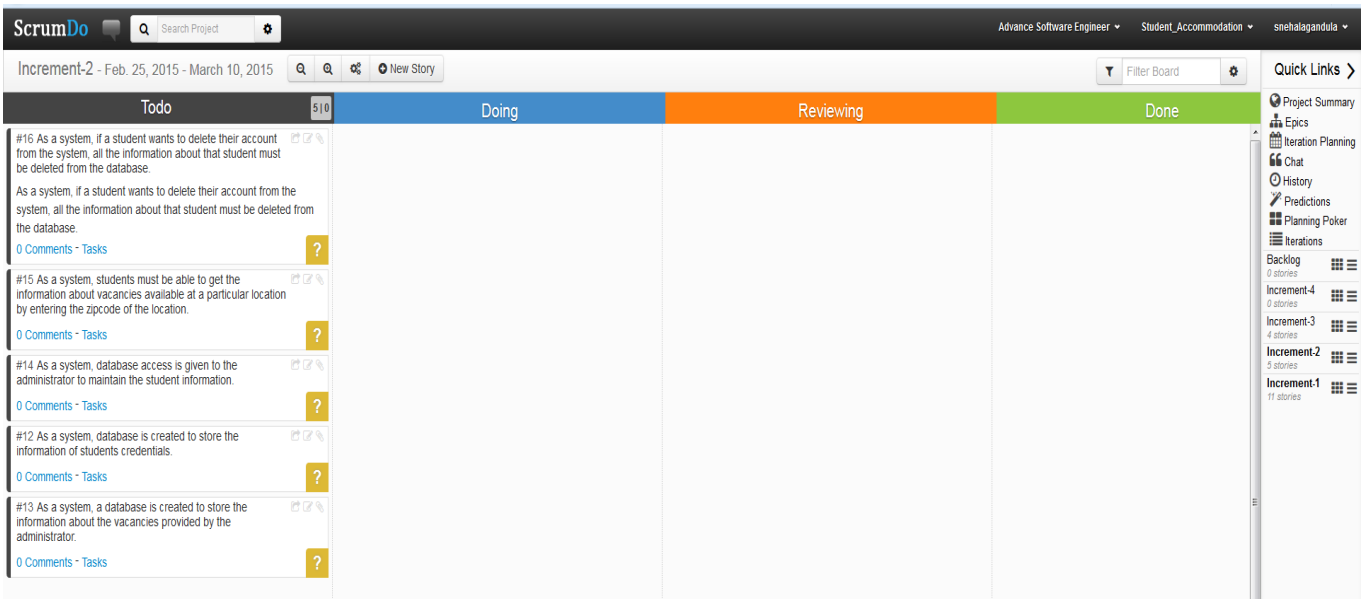
DETAIL DESIGN OF SERVICES:

- **User Stories for first increment**

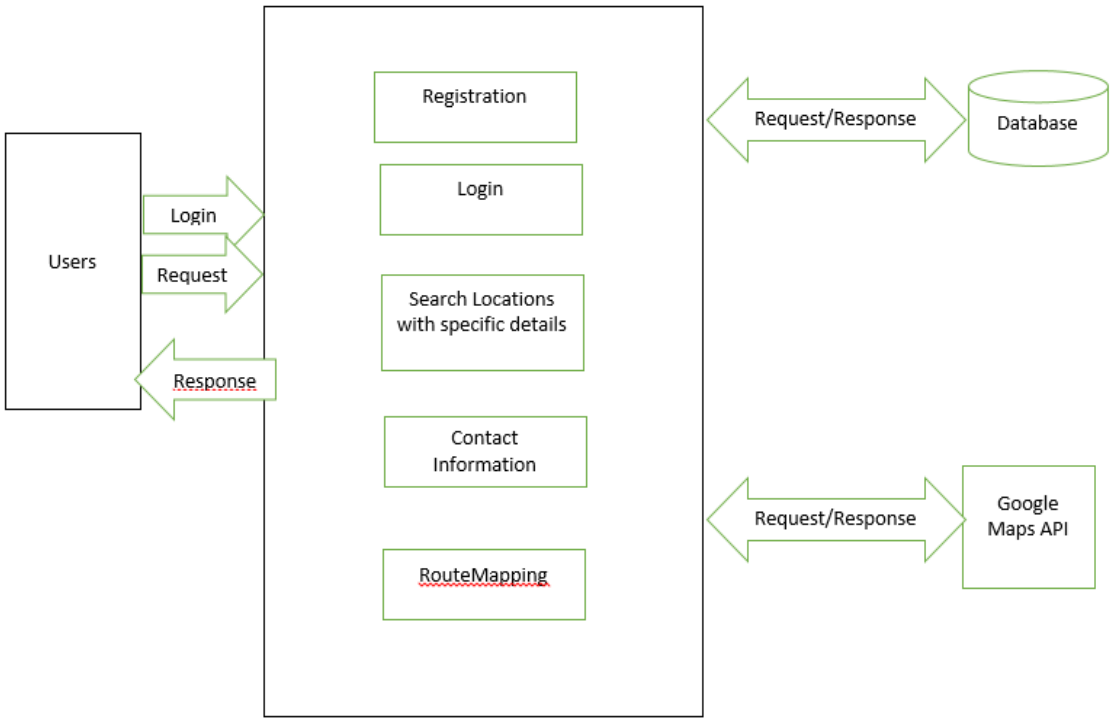
  The first increment includes development of log-in form and registration form for students and administrators with validations. The student details provided in the registration form will be stored in a database for further access. The log-in details of a student will be validated to avoid unauthorized access to the system.



Stories for increment 2:

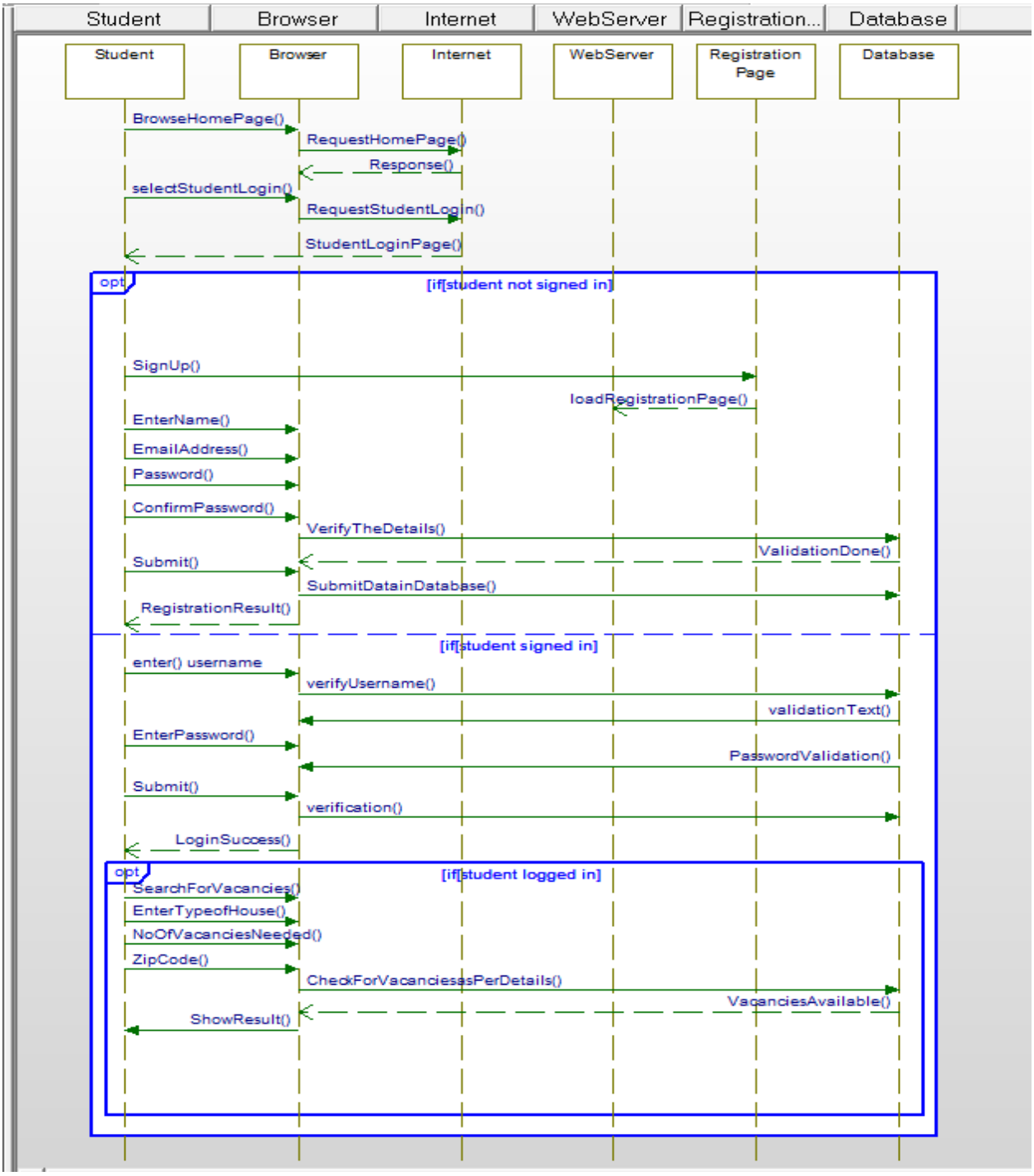


- **Design of Mobile Client Interface:**



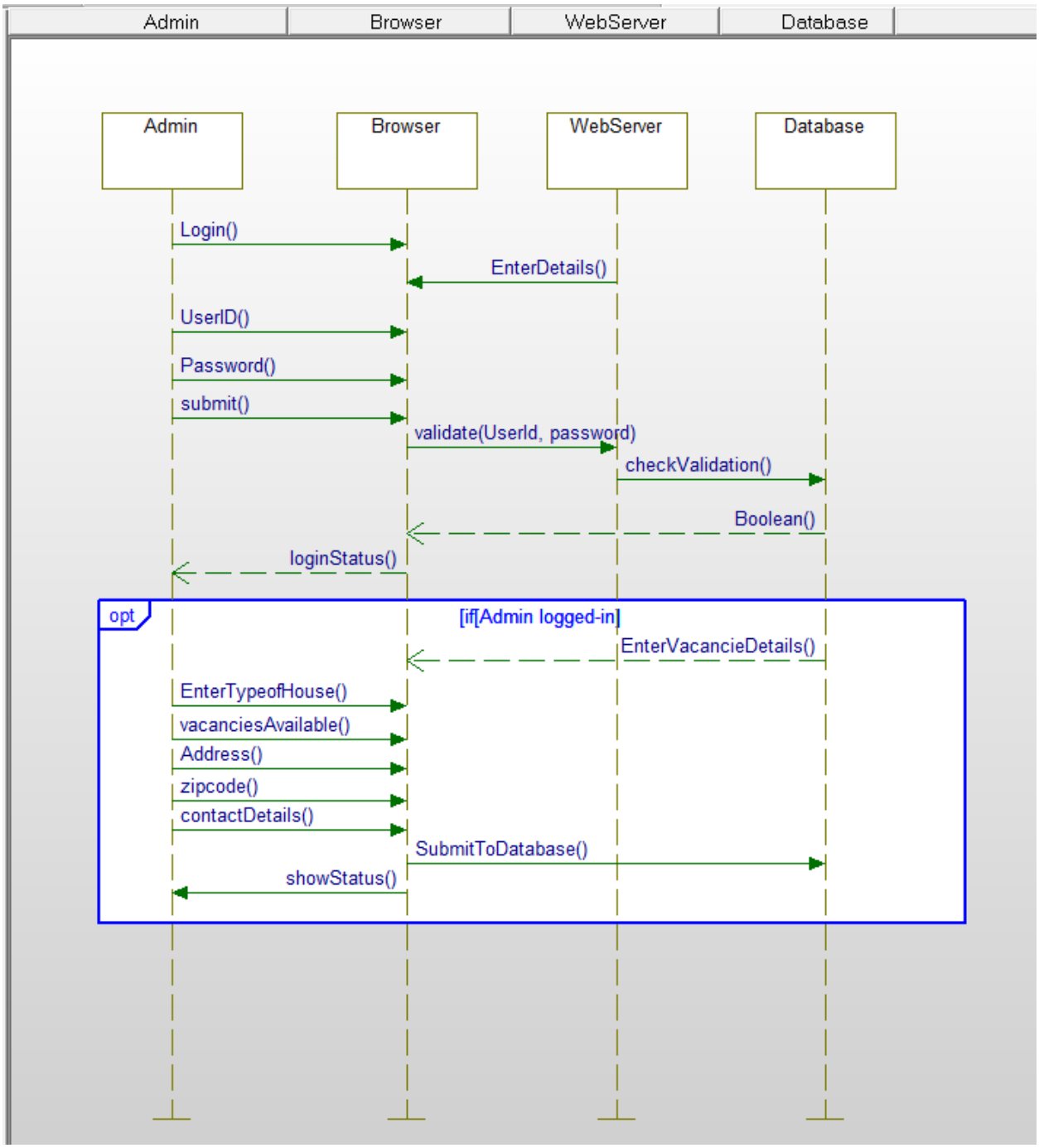
- **Sequence Diagram**

  A sequence diagram is an interaction diagram that provides the collaboration of objects depending on a particular time sequence. The following is the sequence diagram that shows the flow of processes that operates one after the other in a certain order. The sequence diagram is to show the flow of student registration and login form with certain validations.



Sequence Diagram for AdminLogin:

The following sequence diagram explains the log-in action of the administrator. It shows the flow with which administrator can store the details about the vacancies available at a particular location.

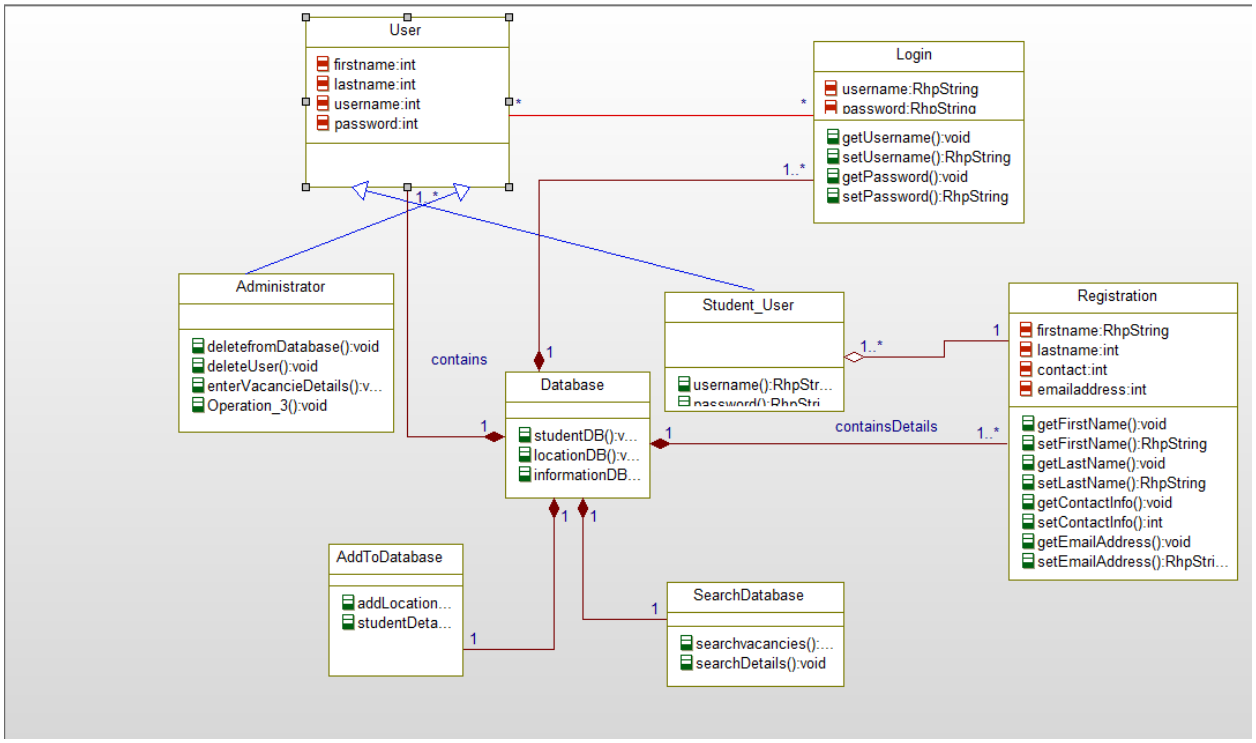


- **Class Diagram**

A class diagram identifies the structure of the application by describing the relationship between the objects and classes of the system.

The following class diagram describes a student or an administrator log-in to the website.

The system has two types of users, student user and the administrator. The student logs-in to the account only if the student is already registered in to the system. The student or admin logs-in to the system by entering the username and password, the system then validates the provided log-in details, and if successful, the system connects the user to the account for further access. If not successful, the system allows repeated attempts to the user and prevents the entry until the log-in is successful.



IMPLEMENTATION:

a) Implementation of User Interface:

The first increment covers the design of the basic user interface for our login and registration form for students and administrator.

Upon request to the URI, a home page is displayed with two buttons, Student Login and Admin Login.

On request for Student Login, a new page will be opened that asks for registration. A new user must register to the application for further access by entering their details. All the registration details are validated for proper data entry.

After completion of the registration, a student has to log-in with the provided userid and password. On successful login, the student is provided with specific fields to search for the information about the available vacancies.

On request for Admin Login, a new page is triggered with log-in details. The log-in details are validated through reusable validation functions. If the admin is successfully logged into the account, he is given privileges to add the housing details such as, available vacancies, address and other contact information.

For all the above log-in and registration forms, validation is done for text fields, passwords and other contact information.

b) Implementation of Test Cases:

Test Cases:

Test-Case	Check Item	Test case objective	Execution Steps	Test Data	Expected Output
1	Log-in page	Click log-in button by leaving all the fields empty/blank	Click log-in button		If all the fields are empty, a mandatory (*) symbol will appear to the username and password
2	Username	Enter invalid username		Username: Sne23a	An error message appears as “Enter valid username”
3	Username	Enter valid username		Username: Sneha	It should allow the student to proceed to

					enter further details.
4	Password	Enter correct password			The entered password should be displayed in encrypted format.
5	Password	Enter wrong password		Password: ***	An error message “Please enter valid password” is displayed
6	Log-in	Correct inputs	Click Log-in		On successful log-in, it should trigger to respective new page
7	Forgot Password		Click forgot password		Password will be recovered by using the link given
8	Registration		Click registration button		On click to registration button a new page will be directed to a new page to enter registration details

Test Requirements:

Username:

- should contain numbers, letters and period

- should not be more than 15 characters

- should not start with any symbols

-should not contain any symbol, period or space

-should not be blank

Test Scenarios:

- Verify if all the specific fields on the registration form are available
- Verify that all the mandatory fields on the registration form are marked with symbol (*). If the user fails to enter the mandatory fields, the web page will not redirect to a new page.
- Verify that the page has submit button after all the fields
- Verify that the validation of user must be done at client side.
- Verify the email address of the students
- Check for validation error if the user enters blank space in mandatory fields
- Check for validation error if the user submit the log-in without entering all the required fields.

User Interface:

- The user name and password text fields should be in order
- New users should be provided with register link that redirects to a new page where registration form is triggered.
- User interface should match the specifications of the application
- On click to the submit button, the page must be redirected to new page where search can be performed.

DEPLOYMENT

1) ScrumDo Link:

   https://www.scrumdo.com/projects/project/student_accommodation/iteration/122907

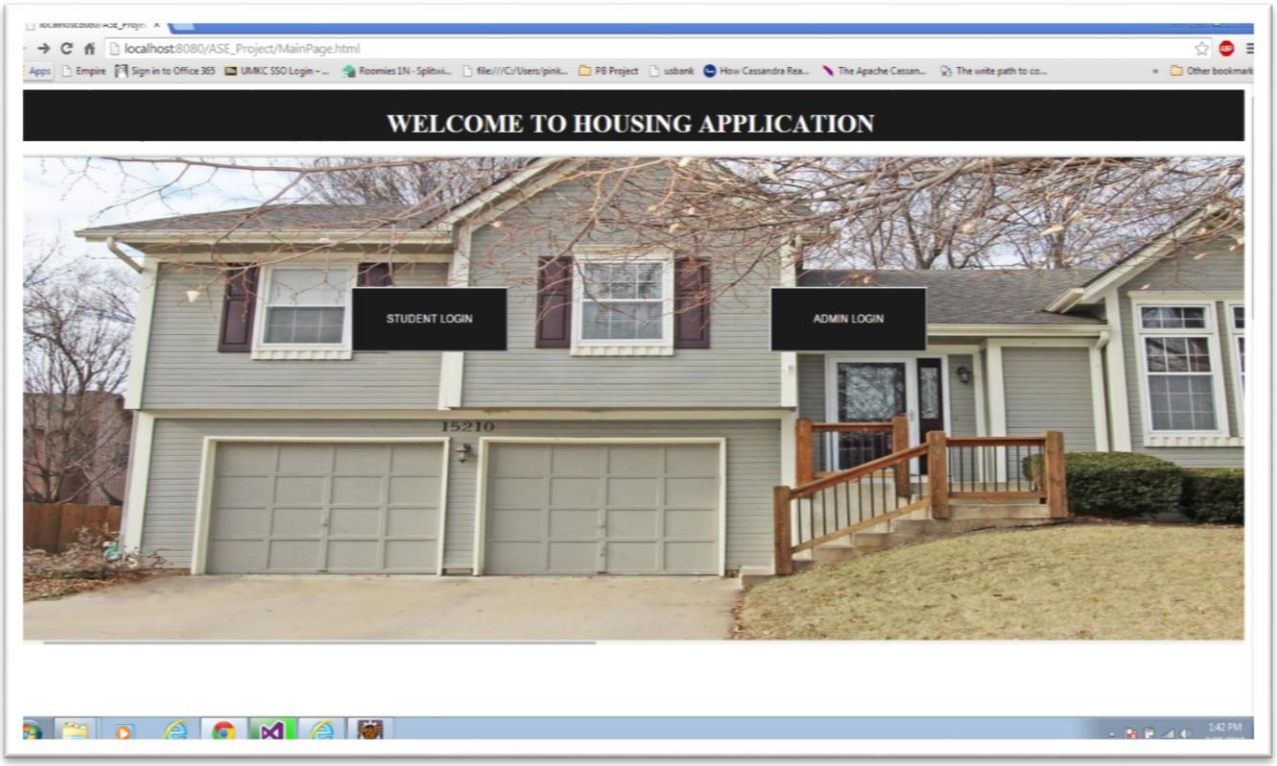
2) GitHub (Source Code)
   https://github.com/ManushaReddy/GitHub/blob/master/ASE_Project.zip

3) GitHub (Report)
   https://github.com/ManushaReddy/GitHub/blob/master/Project%20Report%20Increment%201.docx

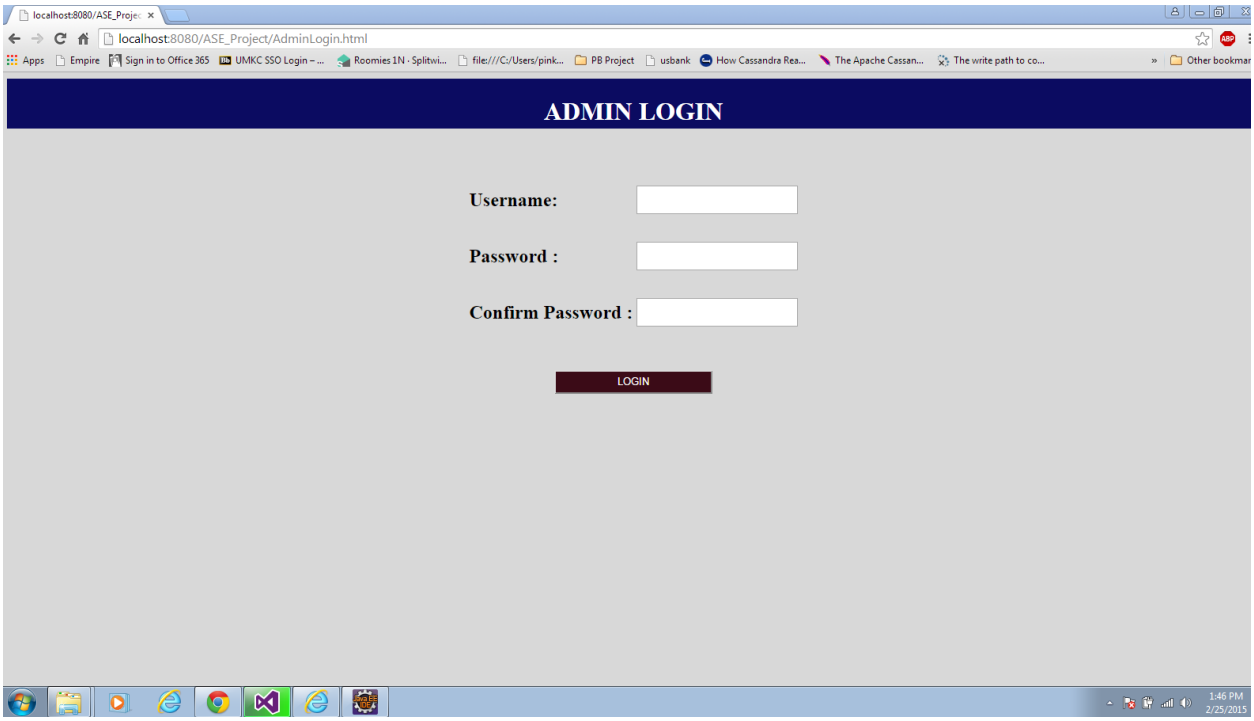
REPORT

Implementation of services:

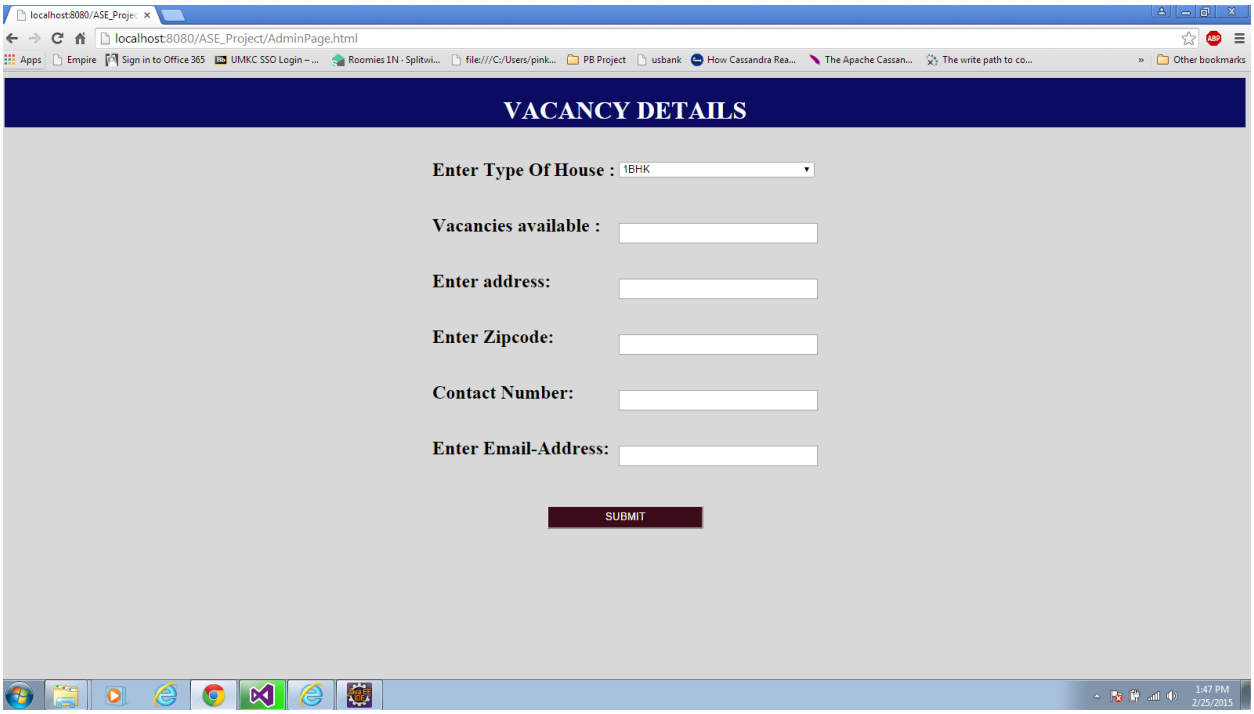
The following page allows the users access the home page of our website.



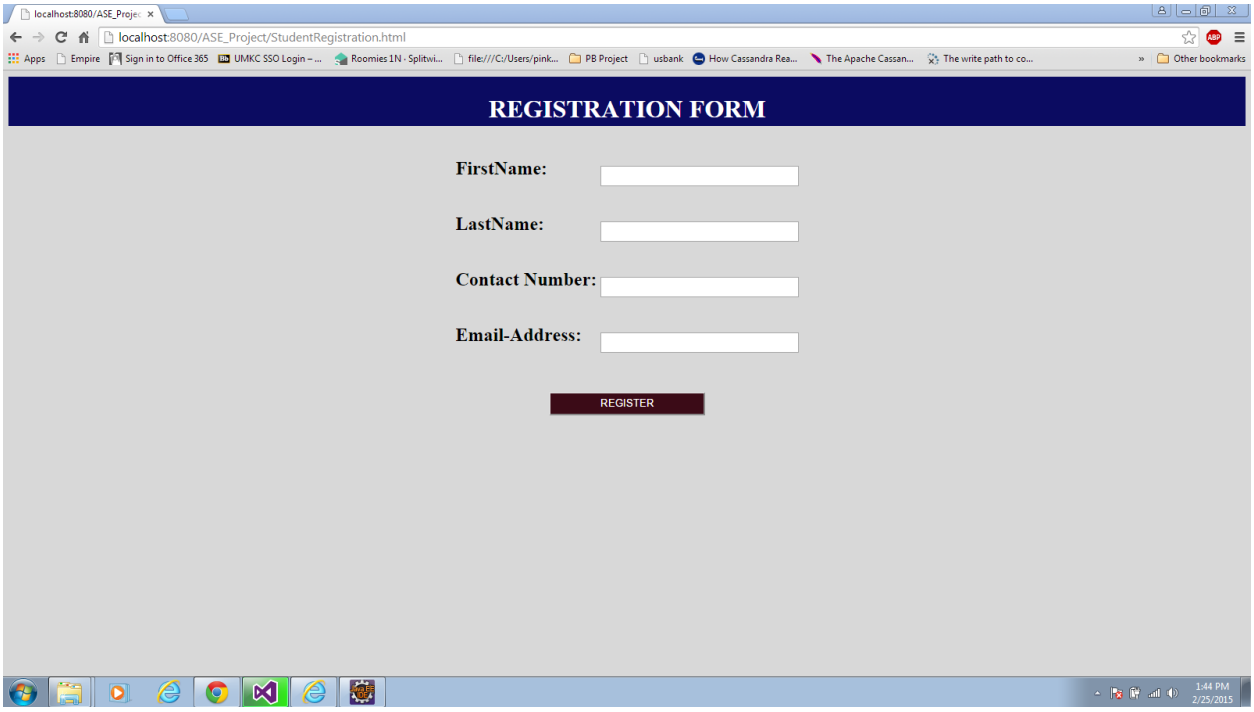
The following log-in page allows administrator to log-in into his account by providing his username and password.



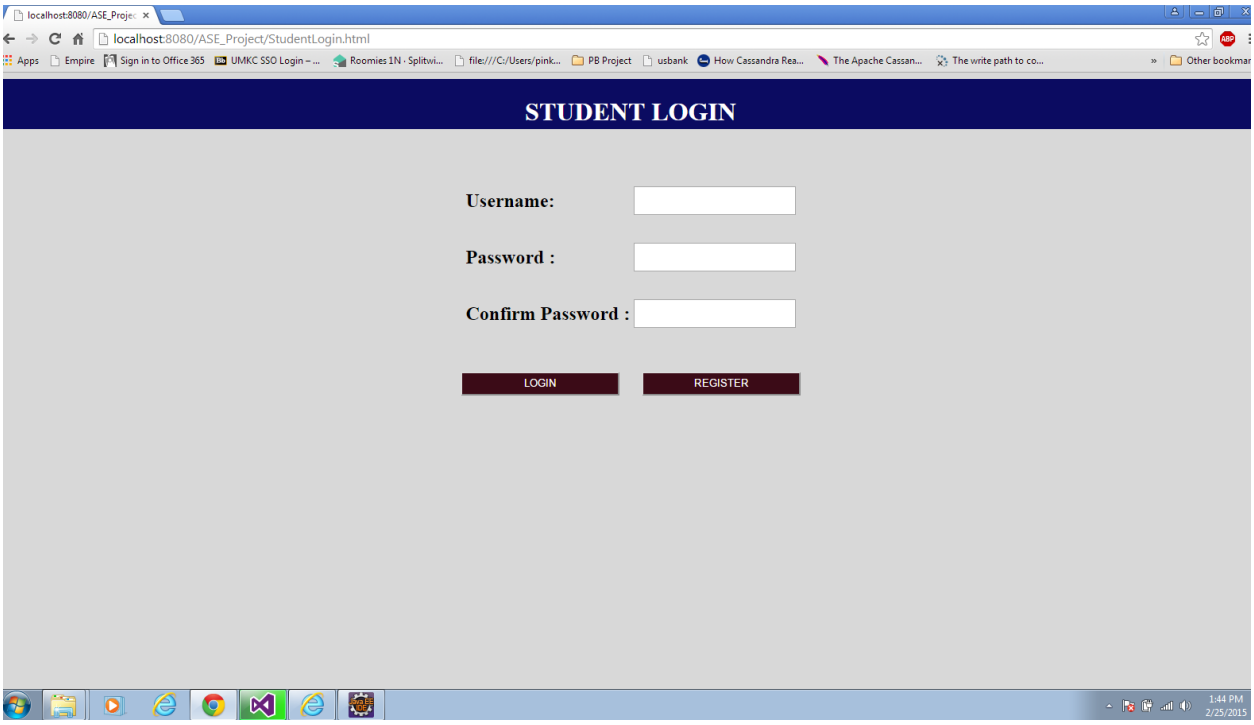
The following page allows administrator to enter the details of the location and other contact information about the available vacancies.



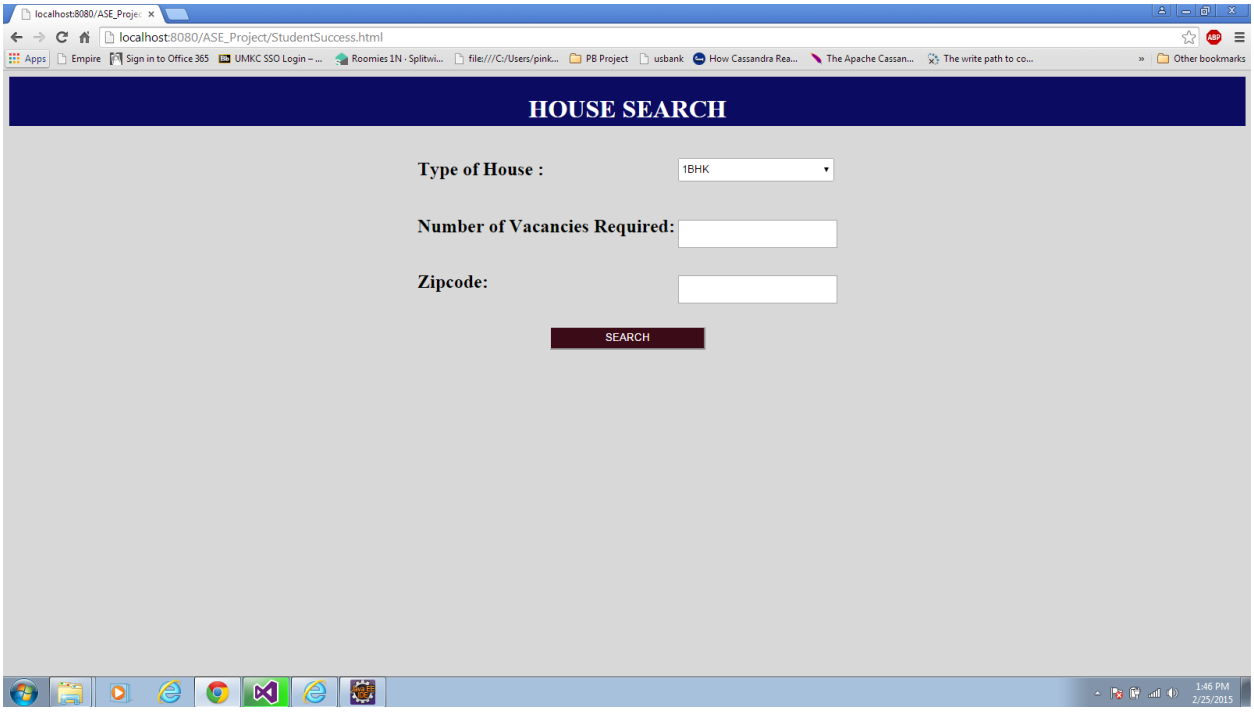
The following registration page allows students to register to the system for further access



The following log-in page allows students to login to the system provided they are already registered to the system.



The following page allows students to search for the vacancies at a particular location.



PROJECT MANAGEMENT:

- **Implementation status report**
  - **Work completed:**
    - Description: Created user interface for student login, administrator login. Provided validation for the student registration and log-in form.
    - Created Database connection using JDBC Driver.
    - Responsibility:
      Task- UI Implementation with validations (Manusha Reddy)
      Task –Database connection and Documentation (Sneha Lagandula)
      Task- CSS (Dheeraj Reddy Bethi)
    - Time Taken: 3 Days
  - **Work to be completed:**
    - **Description:** Data Storage and Retrieval operations for the UI developed. Should create Mash-up using API's