Software Requirements Specification

For

Online Temporary Residential Facility Reservation System

Version 1.0

Prepared by Kaif

CODEMO

<10.2019>

Table of Contents

Revision History

Name	Date	Reason For Changes	Version
Kaif	07.10.2019	Initial document	1.0
Zajjith	07.10.2019	Initial document	1.0
Chithra	07.10.2019	Initial document	1.0
Kasun	07.10.2019	Initial document	1.0

1. Introduction

This section gives a scope description and overview of everything included in this SRS document. Also, the purpose for this document is described and a list of abbreviations and definitions is provided.

1.1 Purpose

The purpose of this document is to provide a detailed description of the Online Temporary Residential Facility System. It will explain the purpose and features of the system, the interfaces of the system, what the system will do, the constraints under which it must operate and how the system will react to external stimuli.

1.2 Document Conventions

No document conventions are being used at this time.

1.3 Intended Audience and Reading Suggestions

This document is intended to be refer by members of the project team that will implement and verify the correct functioning of the system. This has been implemented under the guidance of University lecturers.

1.4 Product Scope

This project intends to develop Hostel Information with Online Reservation and Billing System for online temporary residential facility reservation system. The proposed system would help provide students an easy access for room reservation and a billing system that would provide the management convenience in computing payments. Refunding of payment is not covered by the system. Every transaction is automatically saved in a database which can only be accessed by the management/admin with the use of password for security purposes. The students rescheduling, rebooking and cancellation of reservation shall also be covered by the system. Future additional services are to be discussed with the Hostel's management. The system will also allow reports on that data to be easily generated, printed or exported.

1.5 References

1. OUSL website (Temporary Residential Facility
   Page URL:http://www.ou.ac.lk/home/index.php/2013-11-10-08-19-44)

2. Overall Description

This section will give an overview of the whole system. The system will explained in context to show how the system interacts with the other systems and introduce the basic functionality of it. It will also describe what type of stakeholders that will use the system and what functionality is available for each type.

2.1 Product Perspective

The proposing web based system is a new system that replaces the current manual processes done in the hostel. Figure 1 illustrate the external entities and system interfaces.

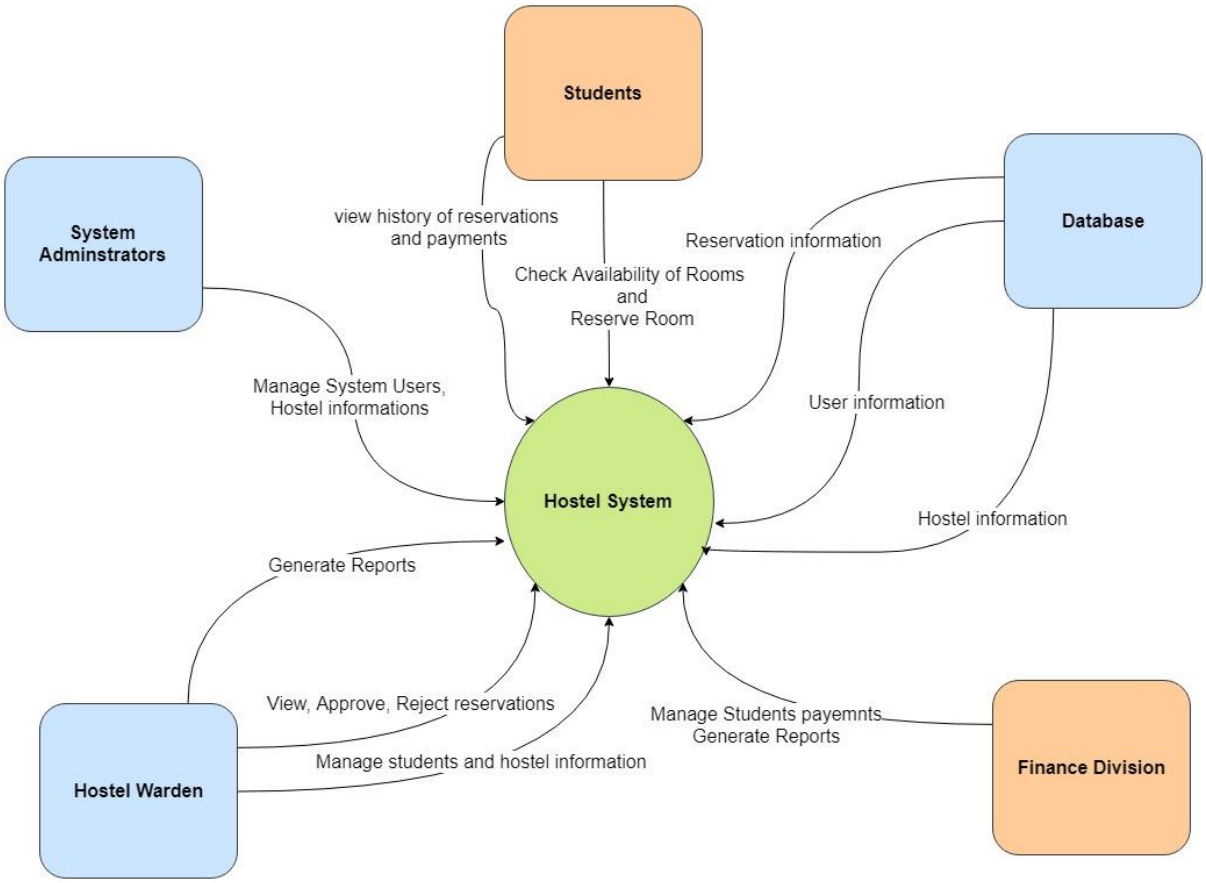

Figure 1

Note: Administrative Users have access to all actions within the system in addition to their own specialized action.

2.2 Product Functions

With the online hostel reservation system where the user is able to check the availability and reserve rooms, cancel reservations, view previous reservations and payments. There are several criteria and it will be possible for the administrator of the system to manage the

option for those criteria that have that. Such as approve reservation, grant access to reserve rooms for a particular user, change the room status and rates.

The web portal will provide functionality to manage the system and the hostel information. It will also provide information about the system, for example show when there is a new update.

2.3 User Classes and Characteristics

Student	Student is the main user of the system and who can request to reserve a room from the hostel facility to get accommodation. Also have the privilege to cancel the request.
Warden	A Warden is charge of managing Hostel Facilities. Warden will use the new system in the same way as a regular user, but will also have the ability to accept/reject the student's request, view hostel room status and approve/reject reports. Warden will also need the ability to manage students' information hostel details and reservation as well. Warden will have the ability to generate reports of student who reserves rooms.
System Administrators	The Administrator is the person or people who will have any and all the privileges of all other user types. They will be able to impersonate students within the system. They will also have the authority to edit the hostel details and manage/generate reports regarding all reservations, payments and students.
Finance Division	The Finance Division user is the person or group of people that will be in charge of all payment activities. They will need to have the ability to generate reservation reports

2.4 Operating Environment

The system shall operate in newest version of all web browsers. Continues services is preferred, but as long as there is no data, minor service interruption can be tolerated.

The Internet connection is also a constraint for the application. Since the application fetches data from the database over the Internet, it is crucial that there is an Internet connection for the application to function.

Personal data is stored in database. Since the web application fetch data from database and the database shared between many online applications in OUSL it may be forced to queue incoming requests and therefore increase the time it takes to fetch the data.

HTML, CSS and PHP libraries will be needed to create the user interface for the system.

2.5 Design and Implementation Constraints

The interface will most likely not be the same for every one of them. Also, there may be a difference between what navigation features each of them provide. User interface shall be composed using JavaScript library.

2.6 User Documentation

No user document information at this time.

2.7 Assumptions and Dependencies

Students are assumed to have a fair estimate of request execution times, so that the decision to accept or reject a request is facilitated.

One assumption about the product is that it will always be used on device can access internet and that have enough performance. If the device does not have enough hardware resources available for the application, for example the users might have allocated them with other applications, there may be scenarios where the application does not work as intended or even at all.

Since all the data is stored in the OUSL database and if the database fail to process due to any maintenance or any other problem occurred the online system will fail to respond to the user because it depends on the database to fetch the data.

3. External Interface Requirements

This section provides a detailed description of all inputs into and outputs from the system. It also gives a description of the hardware, software and communication interfaces and provides basic prototypes of the user interface

3.1 User Interfaces

A first-time user of the application should see the log-in page when he/she opens the application, If, the user has not registered, he/she should be able to do that on contacting the **Student Affair Division**.

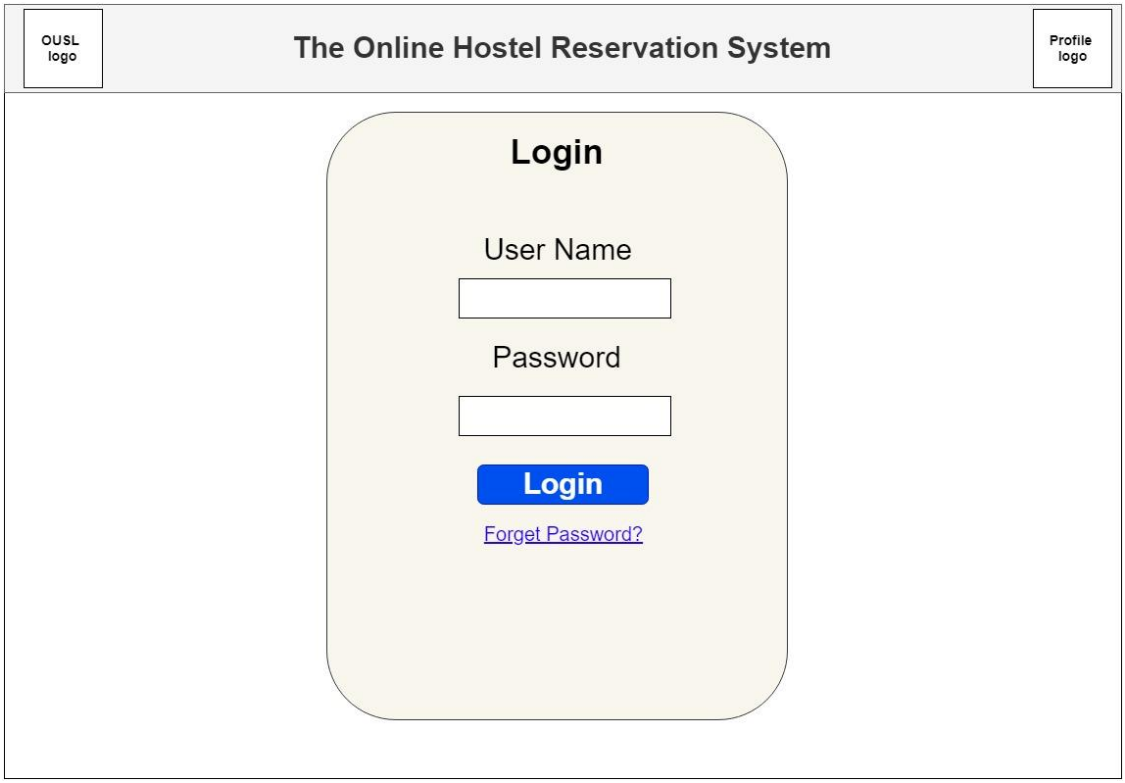

Figure 1

When the user give the correct credentials then user will be directed to the Home page of the application. The view is given below.

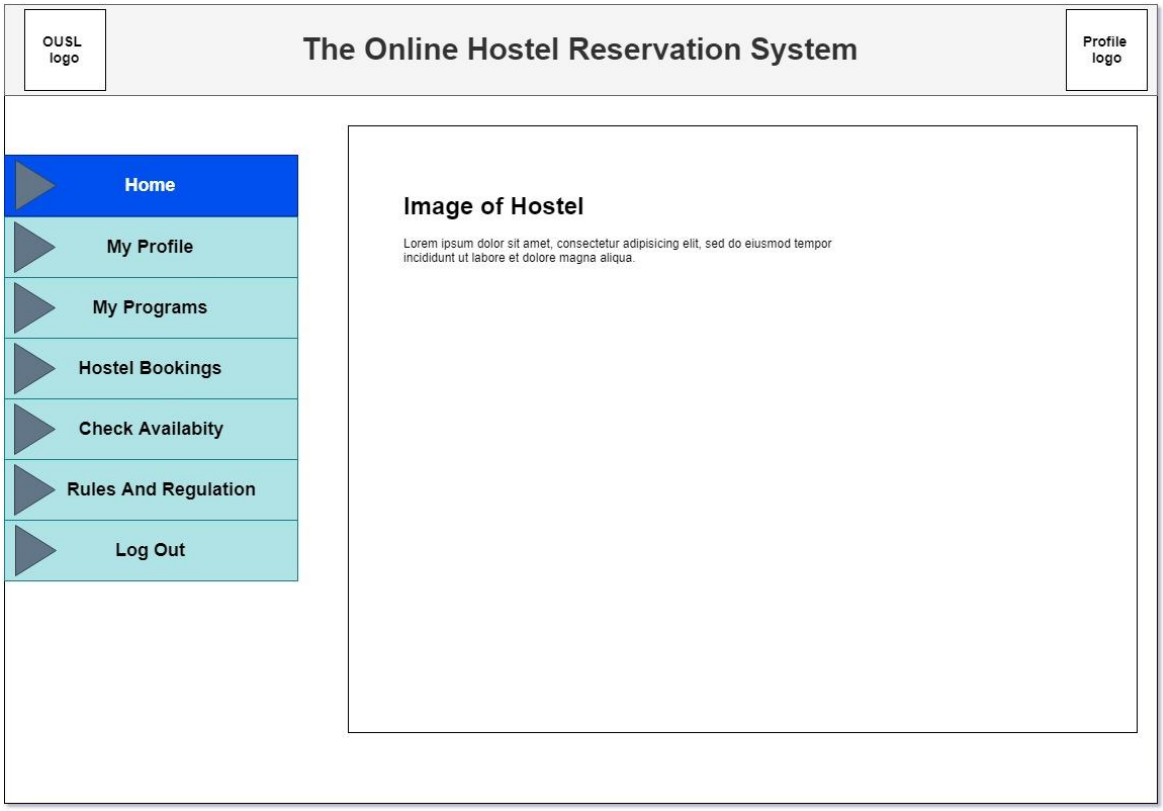

Figure 2

The user can see and edit some of the details of his/her profile in the following interface (figure 3). Some details can be edited who has the administrative privilege in the system.

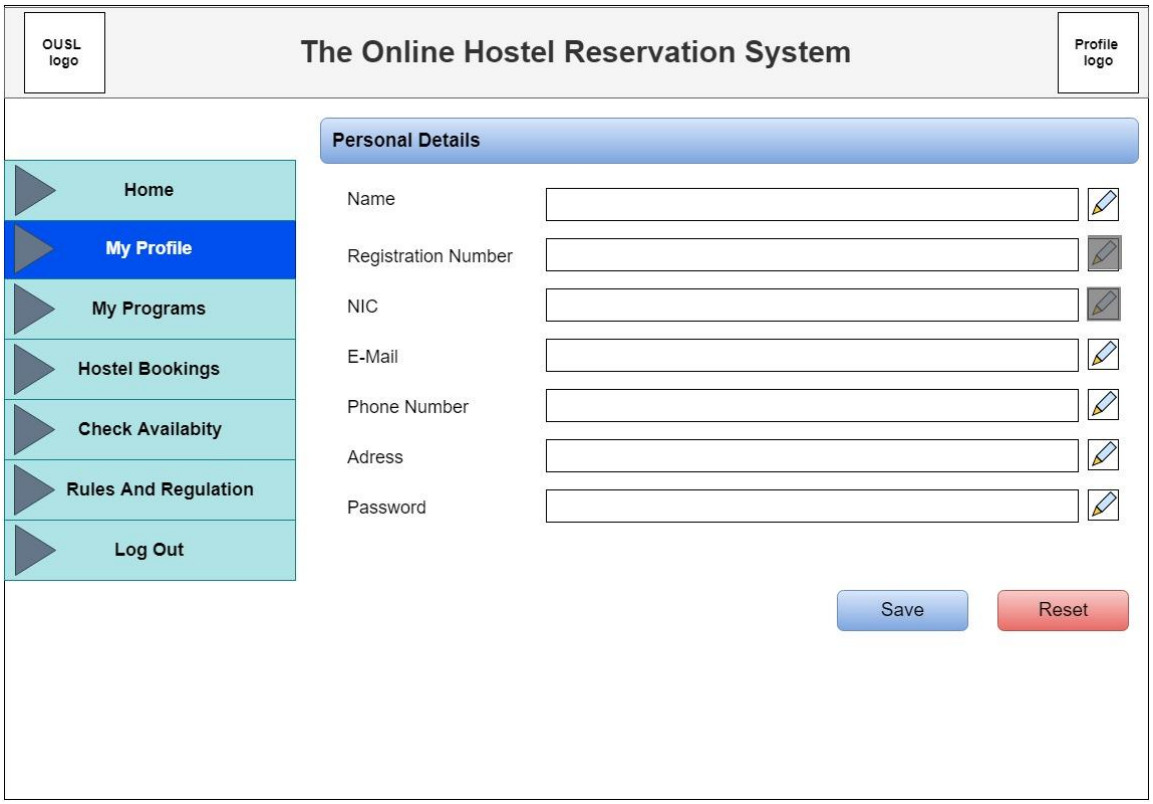

Figure 3

From the following interface (figure 4) an admin user can check the programs and the following courses of a student.

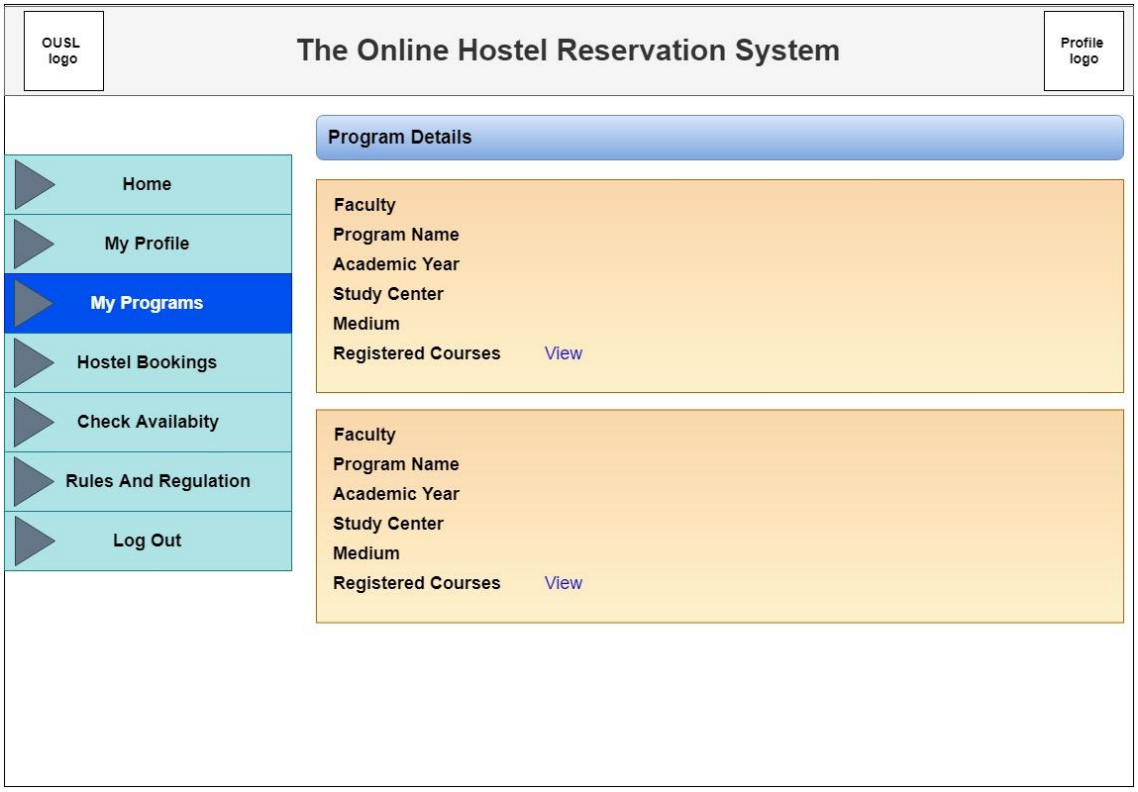

Figure 4

From the following view (Figure 5) a student can request to book a room for consecutive days or for a day. User should enter the staying period and the room number.

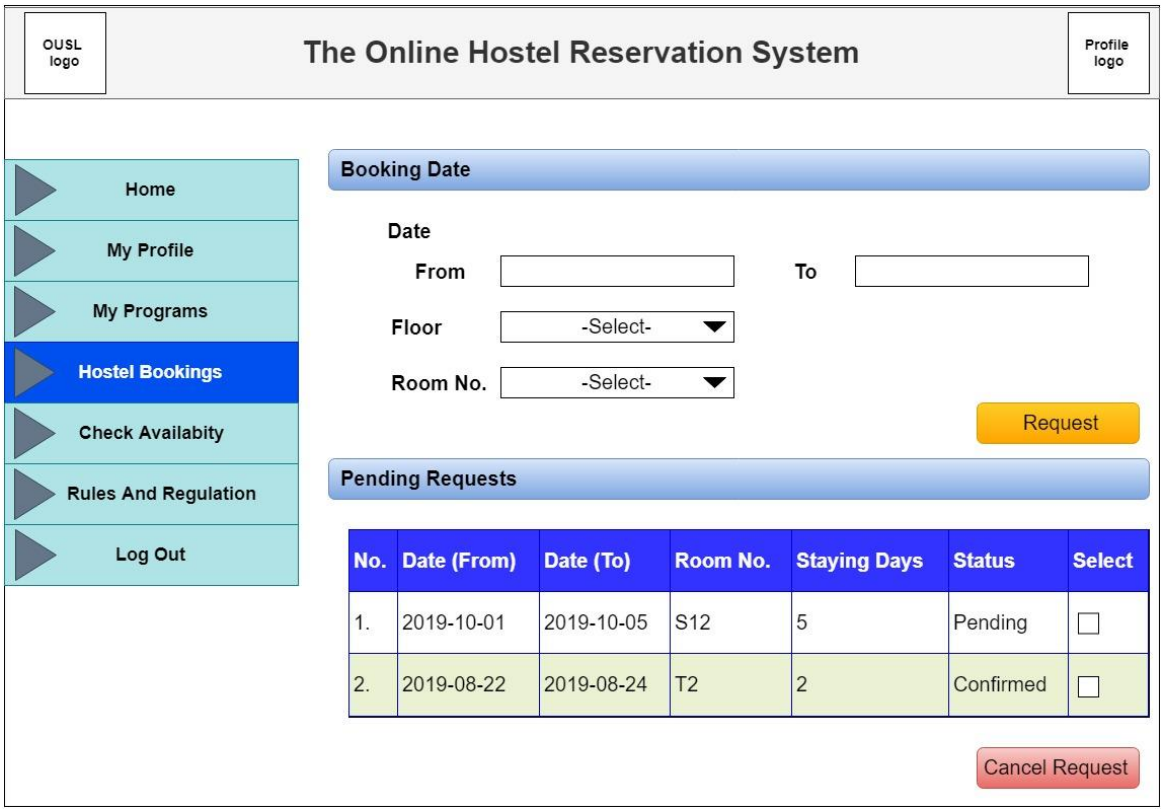

Figure 5

From the following interface (Figure 6) a user can view the availability of the rooms. It shows the available room for the entered date in the top of the list and others after it.

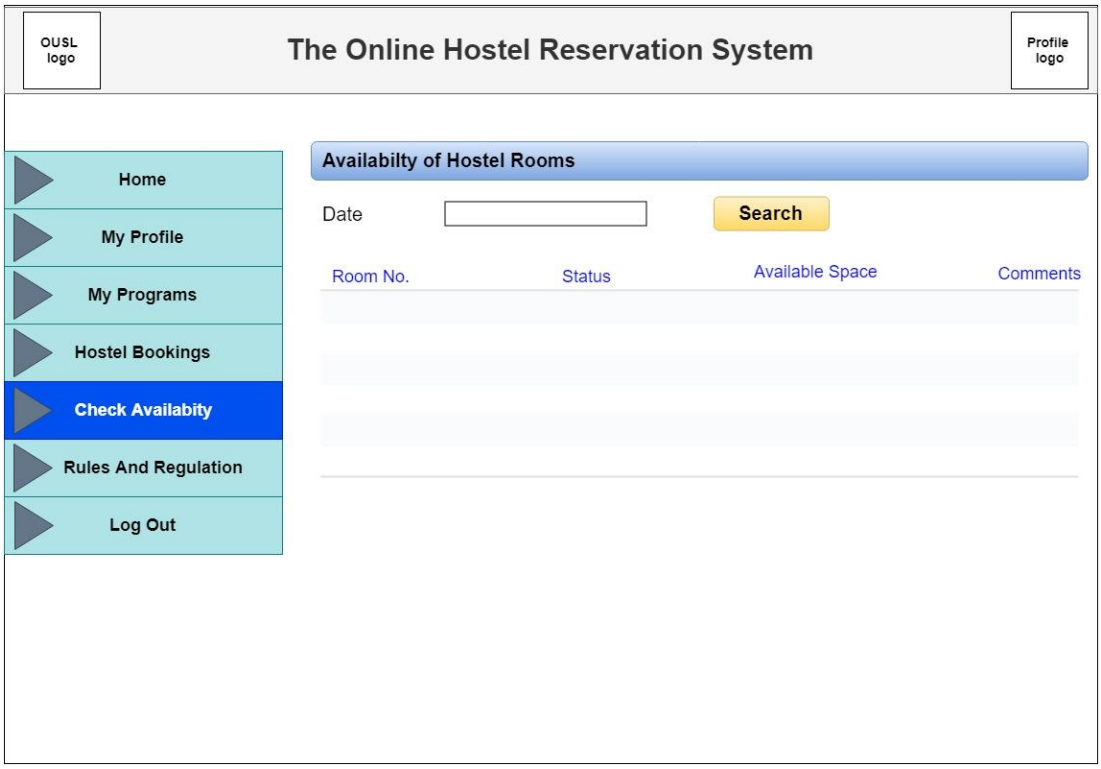

Figure 6

From the following interface (figure 7) user can view the rules and regulation of the hostel reservation.

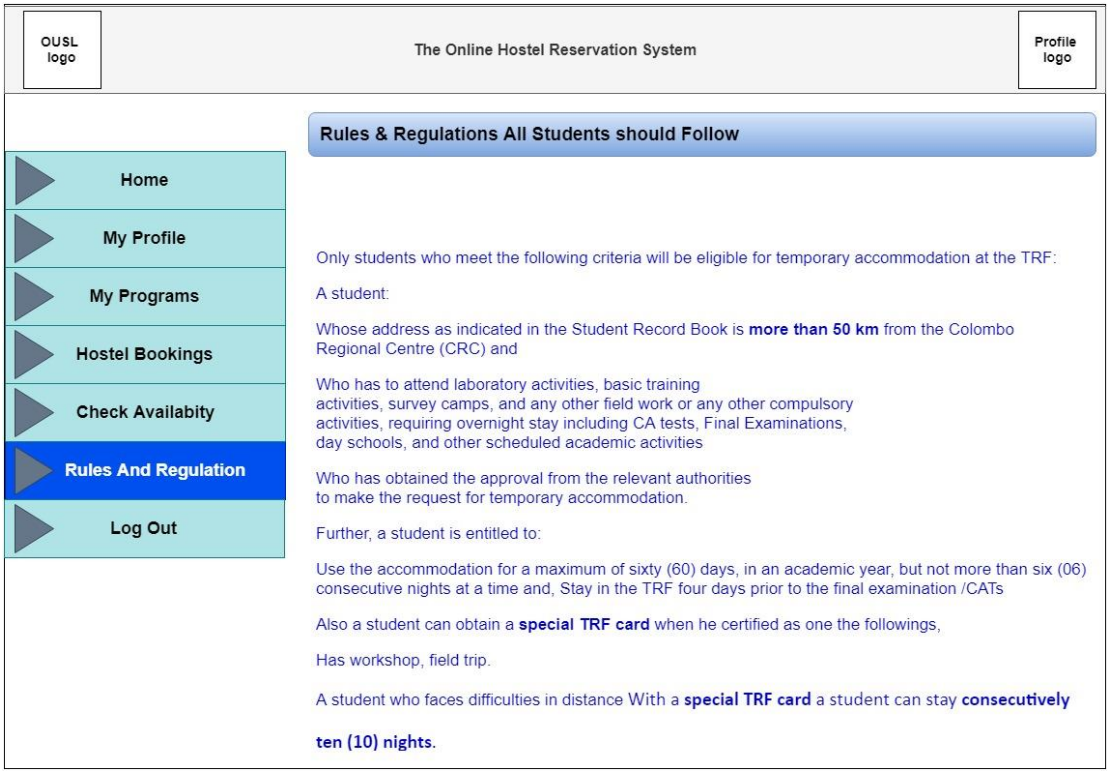

Figure 7

The following interface (figure 8) is only can be viewed by a user who has the admin privileges to the system. From the below interface user can view the students who has booked or currently in the room.

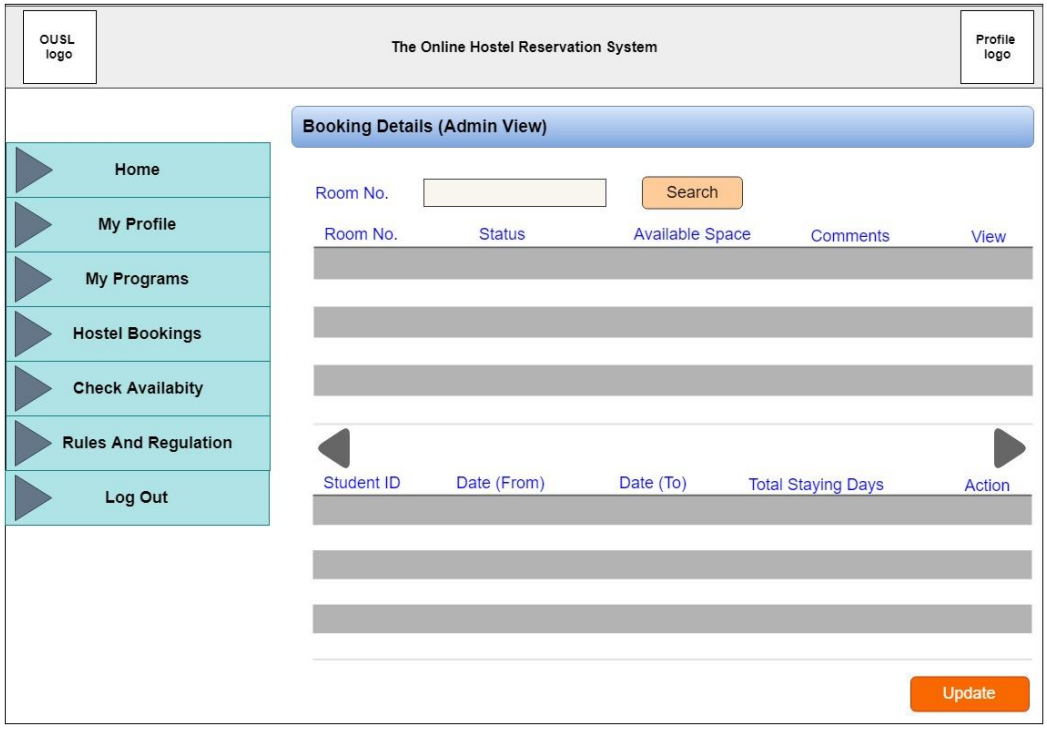

Figure 8

The below interface (figure 9) shows the requests of the students for the hostel system. From this the user can approve or reject the request of a student.

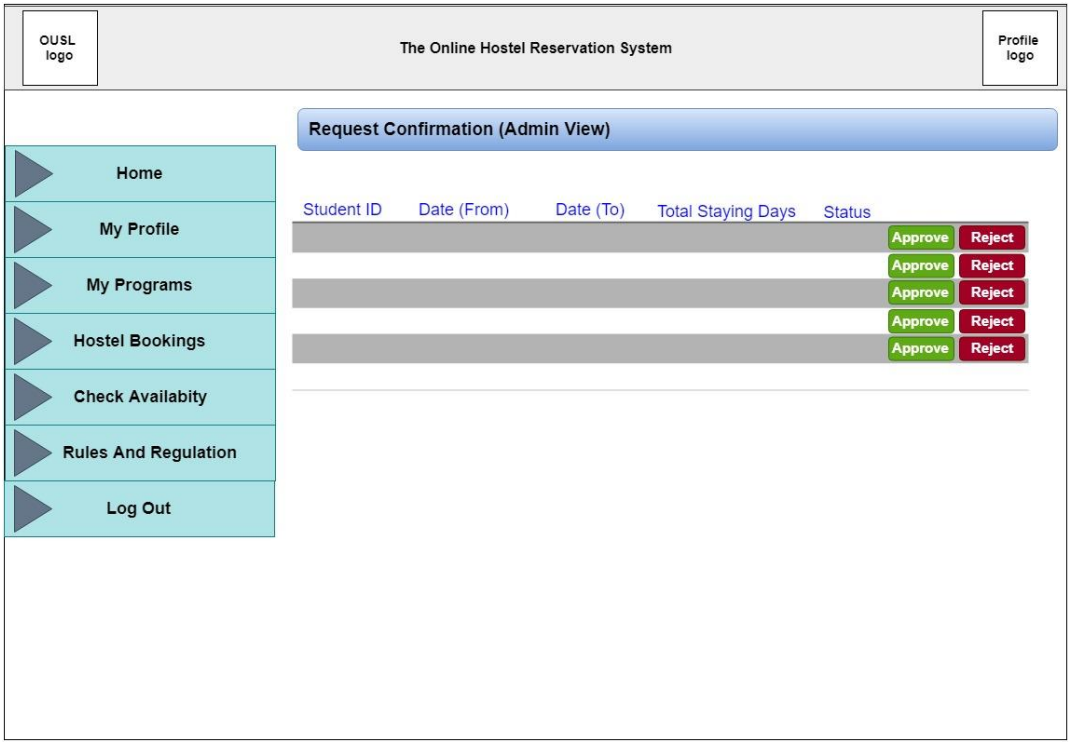

Figure 9

From the following interface (figure 10) user can view the booking history of a student. Also can block or unblock a user for this system.

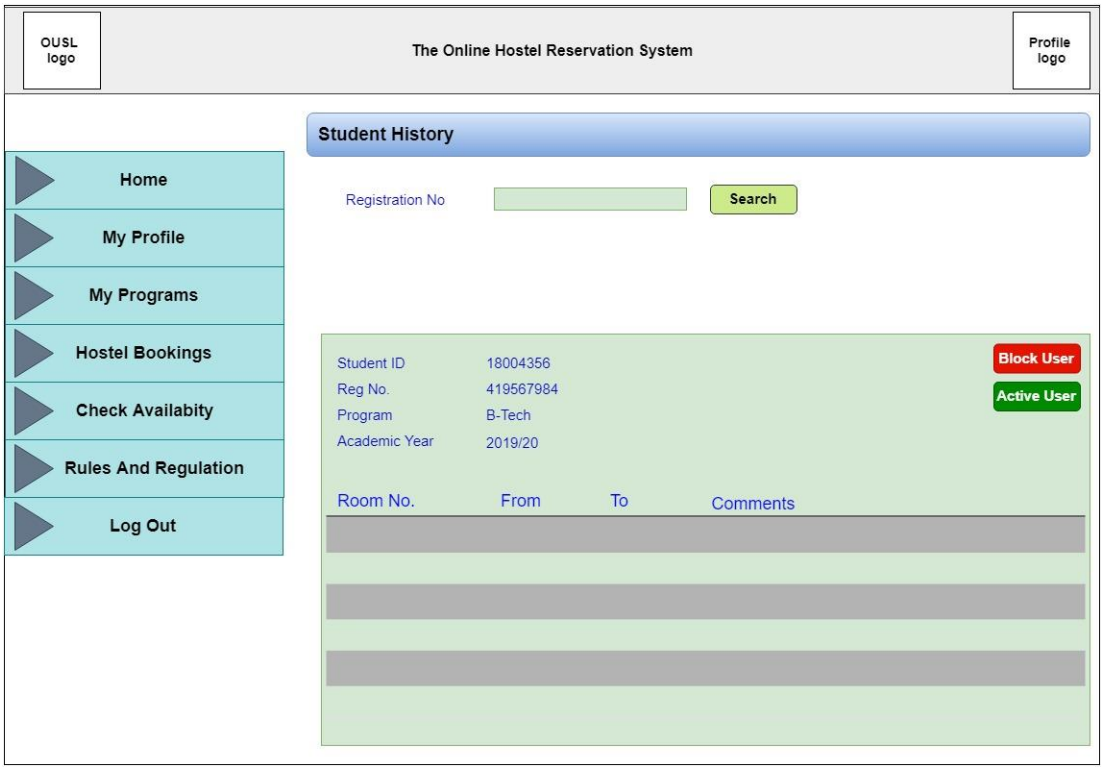

Figure 10

From the below interface user can edit, update or create a user for the system. It also allows to create any type of account such as student and admin type.

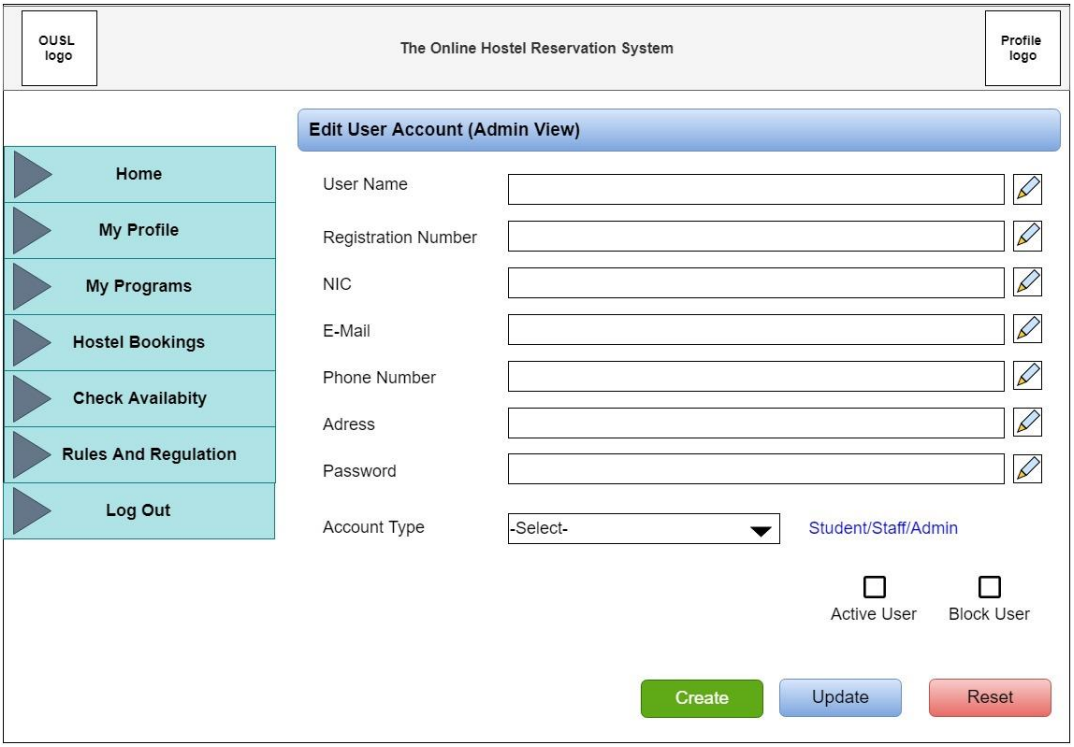

Figure 11

3.2 Hardware Interfaces

Since the application must run over the internet, all the hardware shall require to connect internet will be hardware interface for the system. As for e.g. Modem, WAN – LAN, Ethernet. A browser which supports CGI, HTML & JavaScript.

3.3 Software Interfaces

Following are the software used for the Hostel management online application

Operating system - We have chosen Windows operating system for its best support and user-friendliness.

Database - To save the Hostel records, Students records we have chosen MySQL database.

Back end - PHP

Front End - Html 5, CSS, JavaScript

3.4 Communications Interfaces

The communication between the different parts of the system is important since they depend on each other. This project supports all types of web browsers. We are using simple electronic forms for the reservation forms, room booking etc.

4. System Features

The hostel reservation system maintains information on rooms, availability of rooms, student information and bookings. Of course, this project has a high priority because it is very difficult to reserve a room who are far away from the University and should meet the particular staff only the time when they are available.

4.1 Request to reserve a room

4.1.1 Description and priority

Student will be able to log in to system and request to allocate room.

Priority: High

4.1.2 Stimulus/Response sequence

Stimulus: user request to allocate room
Response: system checks the database and if request does not already exist in database adds it and displays list of current request with the notification of newest request added at the top

Stimulus: user request to cancel allocation
Response: system checks the database and if the request exist in database remove it and displays the previous request with the notification of canceled request.

4.1.3 Functional Requirement

Add Request	The system shall let a User who is a student with the correct permission who logged into the system to add a request to allocate a room.
Cancel request	The system shall let a User who is a student with the correct permission who logged into the system to cancel a request of allocating a room.

4.2 Create an account

4.2.1 Description and priority

Administrators will be able to log into the system and manage user information.

Priority: high

4.2.2 Stimulus/Response sequence

Stimulus: Admin user request to review student
Response: system displays information to requested student

Stimulus: user request to create an account
Response: system checks the user permission and if user account does not exist in database adds it and displays a notification of the new account created.

Stimulus: user request to remove/block a user account
Response: system checks user permission and if user account exist in database removes or blocks the user from the system with the notification of the removed or blocked user.

Stimulus: user request submit changes to student
Response: system checks user permission and updates student in the database.

Stimulus: user request to view all students
Response: system checks user permission and displays list of all current students.

4.2.3 Functional Requirement

Student add	The system shall let a User that is an administrator with correct permissions who is logged into the system to add students or create accounts for new students.
Student remove/block	The system shall let a User that is an administrator with correct permissions who is logged into the system to remove or block students from the system.
Student edit	The system shall let a User that is an administrator with correct permissions who is logged into the system to edit student.
Student view	The system shall let a User that is an administrator with correct permissions who is logged into the system to review students.

Approve/reject request

4.3.1 Description and priority

Admin users will be able to add/remove students from rooms they have been allocated.
Priority: high

4.2.4 Stimulus/Response sequence

Stimulus: user request to view requests
Response: system displays the requests submitted by the students

Stimulus: user request to approve request of student
Response: system checks user permission and flag request as “approved” in the database. System then generates notification to be displayed to the user who submitted the request.

Stimulus: user request to reject request of the student
Response: system checks user permission and flag request as “rejected” in the database. System then generates notification to be displayed to the user who submitted the request.

4.2.5 Functional Requirement

Request view	The system shall let a user who is an administrator with the correct permissions and logged into the system to view requests submitted by students.
Approve request	The system shall let a user who is an administrator with the correct permissions and logged into the system to approve requests submitted by students.
Reject request	The system shall let a user who is an administrator with the correct permissions and logged into the system to reject requests submitted by students.
Cancel request	The system shall let a user who is an administrator with the correct permissions and logged into the system to exist the view of requests without either approving or rejecting it.

4.4 Generate report

4.4.1 Description and priority

Application will need to be able to generate reports for administrators. Students could generate and view reports for their records.

Priority: high

4.4.2 Stimulus/Response sequence

Stimulus: User requests to generate a report.
Response: System displays form containing parameters and submit button.

Stimulus: User requests to submit a report generation form.
Response: System retrieves information from database and displays report in web browser along with ‘Export’ and ‘Print’ buttons.

Stimulus: User requests to export report.
Response: System exports report to selected export option and prompts user to open/save/cancel in web dialog box.

Stimulus: User requests to print report.
Response: System redirects user to pdf version of report in web browser

4.4.3 Functional Requirement

Report Submit	The system shall allow the user to fill out a form to be submitted through the website.
Report Submit Incomplete	The system shall check to see if all required information is present, and prompt the user to fill out missing data and resubmit
Report Submit Recall	The system shall recall any previously saved data and fill in the application automatically
Report Export	The system shall recall data from database and format in file according to export preference.
Report Print	The system shall recall data from database, generate a PDF of formatted data and display PDF.

5. Other Nonfunctional Requirements

5.1 Performance Requirements

The database shall be able to accommodate a thousand record to store.

The software shall support use of multiple users at a time.

There are no other specific performance requirements that will affect development

5.2 Safety Requirements

The database may get crashed at any certain time due to virus or operating system failure. Therefore, it is required to take the database backup.

5.3 Security Requirements

Security systems need database storage just like many other applications. However, the special requirements of the security market mean that vendors must choose their database partner carefully.

5.4 Software Quality Attributes

AVAILABILITY: The rooms should be available on the specified date and specified time as many students are doing advance reservations.

MAINTAINABILITY: The administrators and hostel in chargers should maintain correct availability of rooms.

USABILITY: The hostel should satisfy a maximum number of students' needs.

6. Appendix A: Glossary

Admin or Administrator – the person who control the system
User – who interact with the system.

7. Appendix B: Analysis Models

<Optionally, include any pertinent analysis models, such as data flow diagrams, class diagrams, state-transition diagrams, or entity-relationship diagrams.>

8. Appendix C: To Be Determined List

No issues have been identified.