**Final Year Project Report**

**Hostel Booking Market**

****

**Project Advisor**

Noman Saleem

**Submitted by:**

Ahsan Yousaf F2019266283

Mujtaba Mubasher F2019266267

Muhammad Abdullah Khalid F2019266264

Session

2019-2023

**University of Management and Technology**

**C-II Johar Town Lahore Pakistan**

**Dedication**

I dedicate this project to my family, who has supported me throughout my academic journey. Their unwavering love and encouragement have been a constant source of inspiration, and I could not have accomplished this without them.

I also dedicate this project to my friends, who have been a source of motivation, laughter, and camaraderie. Their feedback, advice, and encouragement have been invaluable throughout the development of this project.

Lastly, I dedicate this project to all the travelers and students who are looking for affordable and reliable accommodation. I hope this project will simplify the process of finding and booking accommodations, and provide a platform for hosts and renters to connect and create meaningful experiences

## 

**Final Approval**

* **Head of Department**  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Department of Informatics & Systems

School of Systems & Technology

UMT Lahore

* **Director (Final Year Projects-CS)** \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Department of Computer Science.

School of Systems & Technology

UMT Lahore

* **Supervisor** \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_
* Department of Informatics & Systems.
* School of Systems & Technology

UMT Lahore

* **Co-Supervisor** \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Acknowledgment**

We would like to express our sincerest gratitude to all those who have contributed to the success of our final year project on hostel booking marketplace.

Firstly, we would like to thank our project advisor **Noaman Saleem** for their guidance, support, and encouragement throughout the entire project. Their expertise and insights have been invaluable in shaping the direction and scope of our work.

We would also like to extend our appreciation to all the individuals who took the time to participate in our survey and user testing. Your feedback and suggestions helped us improve the functionality and usability of our platform.

Furthermore, we would like to thank our friends and family who provided us with unwavering support and motivation, even during the most challenging times of the project.

Lastly, we acknowledge the contributions of all the resources we utilized, such as open-source libraries, online forums, and relevant literature, which provided us with the necessary knowledge and tools to complete our project.

Without the support and assistance of all these individuals and resources, this project would not have been possible. Thank you all for your contributions and for being a part of this journey with us.

**Project Title**

Hostel Booking Market

**Objective**

It is an online hostel booking marketplace which provides a platform to all the hostel management to create their virtual hostels profiles on our platforms. And provide the facility to students to find hostel in required location without any hustle. And the interface of our project will be user friendly and very easy to understand.

**Undertaken by**

Ahsan Yousaf F2019266283

Mujtaba Mubasher F2019266267

Muhammad Abdullah Khalid F2019266264

**Supervised by**

Noman Saleem

**Starting Date**

3-10-2022

**Completion Date**

30-06-2023

**Tools Used**

* React.js
* Node.js
* Express.js
* Mongo DB

**Plagiarism ReportAbstract**

This project aims to develop a hostel booking marketplace, which will serve as a platform for students and travelers to search and book affordable accommodation in various locations. The marketplace will provide a user-friendly interface, allowing hosts to list their properties and renters to search for and book available accommodations based on their preferences. The system will also incorporate a payment gateway and user reviews to enhance transparency and reliability. The project will utilize modern web technologies, such as React, Node.js, and MongoDB, to ensure scalability, security, and robustness. The main objectives of this project are to create a convenient, reliable, and affordable marketplace that will simplify the booking process for renters and hosts, and to provide an easy-to-use platform for managing reservations, payments, and reviews. The success of this project will depend on effective marketing and adoption by users, which will require thorough testing and continuous improvements.

Revision Chart

| **Version** | **Primary Author(s)** | **Description of Version** | **Date Completed** |
| --- | --- | --- | --- |
| Capstone-1 | Ahsan Yousaf  Mujtaba Mubasher  Muhammad Abdullah Khalid | Started working on chapter 1 & 2 of the documentation. | 12/11/2022 |
| Capstone-1 | Ahsan Yousaf  Mujtaba Mubasher  Muhammad Abdullah Khalid | Started working on chapter 3 & 4 of the documentation. | 15/12/2022 |
| Capstone-1 | Ahsan Yousaf  Mujtaba Mubasher  Muhammad Abdullah Khalid | Started working on chapter 5 of the documentation. | 03/01/2023 |

# Contents

[**2.**](#_heading=h.2s8eyo1) Contents 14

[**a.**](#_heading=h.17dp8vu) Definitions and Acronyms 15

[b.](#_heading=h.26in1rg) List of Figures 16

[c.](#_heading=h.lnxbz9) List of Tables 17

[**1.**](#_heading=h.35nkun2) Introduction 19

[1.1](#_heading=h.1ksv4uv) Motivations 19

[1.2](#_heading=h.44sinio) Project Overview 19

[1.3](#_heading=h.2jxsxqh) Problem Statement 19

[1.4](#_heading=h.z337ya) Objectives 19

[**2.**](#_heading=h.3j2qqm3) Domain Analysis 20

[2.1](#_heading=h.1y810tw) Customer 20

[2.2](#_heading=h.4i7ojhp) Stakeholders 20

[2.3](#_heading=h.1ci93xb) Affected Groups with social or economic impact 20

[2.4](#_heading=h.3whwml4) Dependencies/ External Systems 20

[2.5](#_heading=h.2bn6wsx) Reference Documents 21

[2.5.1 Related Projects 21](#_heading=h.qsh70q)

[2.5.2 Feature Comparison 21](#_heading=h.3as4poj)

[**3.**](#_heading=h.2p2csry) Requirements analysis 22

[3.1](#_heading=h.147n2zr) Requirements 22

[3.2 List of Actors 24](#_heading=h.1hmsyys)

[3.3](#_heading=h.41mghml) List of use cases 25

[3.4](#_heading=h.2grqrue) System use case diagram 26

[3.5](#_heading=h.vx1227) Extended use cases For Client 27

[3.6](#_heading=h.206ipza) User interfaces (mock screens) 39

[4.](#_heading=h.2zbgiuw) Data flow diagram (optional) 41

[4.1](#_heading=h.1egqt2p) Data Flow Diagram Level 0 41

[4.2](#_heading=h.3ygebqi) Data Flow Diagram Level 1 41

[4.3 Data Flow Diagram Level 2 42](#_heading=h.2dlolyb)

[**5.**](#_heading=h.sqyw64) System Design 43

[5.1](#_heading=h.3cqmetx) System Architecture Diagram 43

[5.2](#_heading=h.1rvwp1q) Class Diagram 43

[5.3](#_heading=h.4bvk7pj) Sequence Diagrams 44

[5.4](#_heading=h.2r0uhxc) Collaboration Diagrams 45

[5.5](#_heading=h.1664s55) ERD 46

[5.6](#_heading=h.3q5sasy) Data Dictionary 46

[**6.**](#_heading=h.3hv69ve) **Implementation details** 50

[**6.1**](#_heading=h.1x0gk37) **Development Setup** 50

[**6.2**](#_heading=h.4h042r0) **Algorithms** 50

[**6.4 Constraints** 55](#_heading=h.2nusc19)

[**6.4.1 Assumptions** 55](#_heading=h.1302m92)

[**6.4.2 System constraints** 55](#_heading=h.3mzq4wv)

[**6.4.3 Restrictions** 55](#_heading=h.2250f4o)

[**6.4.4 Limitations** 55](#_heading=h.haapch)

[7. TESTING 56](#_heading=h.319y80a)

[7.1 Extended Test Cases 56](#_heading=h.1gf8i83)

[7.2 Decision Table 62](#_heading=h.184mhaj)

[7.2.2 Decision coverage table 62](#_heading=h.3s49zyc)

[8. RESULTS/OUTPUT/STATISTICS 64](#_heading=h.2koq656)

[9. Conclusion 66](#_heading=h.2y3w247)

[10. Future Work 67](#_heading=h.1d96cc0)

## Definitions and Acronyms

| **Acronym** | **Definition** |
| --- | --- |
| UMT | University of Management and Technology |
| POS | Point of Sale |
| UCD | Use Case Diagram |
| DFD | Data Flow Diagram |
| ERD | Entity Relationship Diagram |

**Table a-1: table of acronyms and definitions**

## List of Figures

[Figure 1: System Use case Diagram](about:blank) 25

[Figure 2 DFD Level 0](about:blank) 39

[Figure 3 DFD Level 1](about:blank) 39

[Figure 4 DFD Level 2](about:blank) 40

[Figure 5 System Architecture](about:blank) 41

[Figure 6 Class Diagram](about:blank) 41

[Figure 7 Sign Up Sequence diagram](about:blank) 42

[Figure 8 Login Sequence Diagram](about:blank) 42

[Figure 9 Create Hostel Sequence Diagram](about:blank) 43

[Figure 10 Collaboration diagram](about:blank) 43

[Figure 11 ERD Diagram](about:blank) 44

## List of Tables

[Table a-1: table of acronyms and definitions 15](#_heading=h.3rdcrjn)

[Table 2.2-1: list of stakeholders 20](#_heading=h.2xcytpi)

[Table 2.5-1: Feature Comparison 21](#_heading=h.49x2ik5)

[Table 3.1-1: List of Requirement Category 22](#_heading=h.23ckvvd)

[Table 3.1-2: List of Requirement 24](#_heading=h.32hioqz)

[Table 3.5-1: UC-01 Sign Up 27](#_heading=h.3fwokq0)

[Table 3.5-2: UC-02 Login 28](#_heading=h.1v1yuxt)

[Table 3.5-3: UC-03 Logout 29](#_heading=h.4f1mdlm)

[Table 3.5-4: UC-04 Create Hostel 30](#_heading=h.2u6wntf)

[Table 3.5-5: UC-05 Hostel Approval 31](#_heading=h.19c6y18)

[Table 3.5-6: UC-06 Account Management 32](#_heading=h.28h4qwu)

[Table 3.5-7: UC-07 Create Portfolio 33](#_heading=h.nmf14n)

[Table 3.5-8: UC-08 Upload Multimedia Content 34](#_heading=h.37m2jsg)

[Table 3.5-9: UC-09 Search 35](#_heading=h.46r0co2)

[Table 3.5-10: UC-10 Reservation 35](#_heading=h.2lwamvv)

[Table 3.5-11: UC-11 Ratings & Comments 36](#_heading=h.111kx3o)

[Table 3.5-12: UC-12 Online Payment 37](#_heading=h.3l18frh)

[Table 5.6-1: Data Dictionary 5.6.1- Customer 45](#_heading=h.25b2l0r)

[Table 5.6-2: Data Dictionary 5.6.2- Hostel Owner 46](#_heading=h.kgcv8k)

[Table 5.6-3: Data Dictionary 5.6.3- Payment 46](#_heading=h.34g0dwd)

[Table 5.6-4: Data Dictionary 5.6.4- Reservation 47](#_heading=h.1jlao46)

[Table 5.6-5: Data Dictionary 5.6.5- Services 47](#_heading=h.43ky6rz)

[Table 5.6-6: Data Dictionary 5.6.6- Booking 48](#_heading=h.2iq8gzs)

[Table 5.6-7: Data Dictionary 5.6.7- Admin 48](#_heading=h.xvir7l)

[Table 7.1-1 sign up testing 51](#_heading=h.40ew0vw)

[Table 7.1-2 Test case- log in 52](#_heading=h.2fk6b3p)

[Table 7.1-3 test case – current location 53](#_heading=h.upglbi)

[Table 7.1-4 test case- nearest hostel 54](#_heading=h.3ep43zb)

[Table 7.1-5 test case- online payment 55](#_heading=h.4du1wux)

[Table 7.1-6 test case - feedback 56](#_heading=h.2szc72q)

[Table 7.1-1 DT-Sign up 57](#_heading=h.279ka65)

[Table 7.1-2 DT-Login 57](#_heading=h.meukdy)

[Table 7.1-3 DT-User Home Page 57](#_heading=h.36ei31r)

[Table 7.1-4 DT-Search Function 58](#_heading=h.1ljsd9k)

[Table 7.1-5 DT- Hostel Profile 58](#_heading=h.45jfvxd)

# Introduction

## Motivations

Internet is becoming more and more popular, and businesses are under a lot of pressure to create effective strategies for their online operations. According to research so far, online businesses must rely more on methods of distinctiveness and focused market scope than on pricing management and fee. This project focuses on how distinction and market scope strategies interact and have conflicting effects on customer pride and loyalty, a topic that has received little attention in the online environment.

## Project Overview

It is an online hostel-booking marketplace,which provides a platform to all the hostel management to create their virtual hostels profiles on our platforms.

## Problem Statement

Problem is that students come from out of cities for the admissions and it’s hard for them to find a hostel with the hectic routine task of getting admission in universities because they are very busy in apply for admission in different universities and when they get then they start to find the hostel which leads them to face a problem to find a good hostel around the university and mostly hostel are full so to get and the city is unknown for them so it’s difficult for them to explore hostels manually.

## Objectives

We will create a market place which is web based it is a marketplace for all the hostel all over in Pakistan but as a startup we currently only work in Lahore later on we will expand it. All hostel management in Lahore will be able to create their virtual hostel with full details, location and pictures of their hostels rooms and facilities they providing it helps students to find good hostels at one click from home and they can reserve the room according to their requirement. The payment will be done after visiting by hand to avoid any scam.

We except that it will be helpful for the students to relief the stress of finding good and hygienic hostel without any hustle and find many hostel at one place and he can also compare their facilities and rates.

# Domain Analysis

## Customer

Customers who struggle to find good hostels in their area or customers who struggle to go out to hostels due to rising environmental issues or hazards will be the main goal of the web application. By comparing features, they may locate a decent hostel and make an informed decision.

* Students
* Hostel Owners

## Stakeholders

| **Stakeholder** | **Role in System** |
| --- | --- |
| Administrator | Will be checking over the entire system. Managing the connected entity's rights will ensure that everything runs smoothly. |
| Customer | Customer seeking the hostel room. |
| Hostel Owners | Owners of hostels that wish to use this platform to register their operation. |

**Table 2.2-1: list of stakeholders**

## Affected Groups with social or economic impact

An opportunity arises for the proprietors through this platform to enhance their business and engage their users by providing them facilities regarding Hostel.

* Hostel Owners
* They can promote their business and get revenue.
* They do not require any medium to advertise their hostels.
* Users
* Secure user time to find their desired services.
* Provide multiple Hostels at one platform to users.

## Dependencies/ External Systems

The system is directly dependent upon:

* Internet
* Hardware (PC, Laptops, Smart Mobiles)
* Database

## Reference Documents

The references that have been consulted during analysis phase are:

### Related Projects

In order to develop Hostel Booking Market place, we looked up several similar systems. Their details are given below

1. [Fiverr](https://www.fiverr.com/)

Developed by Micha Kaufman and Shai Wininger. The concept of payment i.e the admin will hold the payment until the client is satisfied of the hostel and then deliver the payment to the hostel owner.

1. [Hostelworld](https://www.hostelworld.com/)

The hostel owner will have a separate panel where he will be able to see the details of number of rooms which are available and which are not so they can limit the number of clients.

1. [Uber](https://www.uber.com/pk/en/)

The feature of target location. The client will be able to put in the target location where he currently is and the website will provide with the nearby hostels.

### Feature Comparison

| **Sr No.** | **Comparison Feature** | **Fiverr** | **Hostelworld** | **Uber** | **Remarks** |
| --- | --- | --- | --- | --- | --- |
| 1 | Payment Holding System | Fiverr covers the feature of payment. | Hostel world does not cover this feature to hold the payment until the client is satisfied, | Uber does not cover this feature. | The feature of holding the payment until the client is satisfied with the hostel is taken from fiver. |
| 2 | Admin Panel | Fiverr does not cover this feature | Hostelworld covers this feature | Uber does not have this feature | This feature is taken from Hostelworld and will allow us to maintain a number of clients in hostels within the capacity. |
| 3 | Target Current Location | Fiverr does not-cover this feature | Hostel world covers this feature | Uber-has this-feature available. | This feature is taken from uber and will allow the client to check nearby hostels |

**Table 2.5-1: Feature Comparison**

# Requirements analysis

## Requirements

Requirements of Hostel Booking Market Place Project are as follows:

**3.1.1 Performance requirement:**

There are not any specific performance requirements because all our end customers need in order to use and access our platform and carry out specific tasks is an internet connection.

**3.1.2 Design constraints:**

There are no design restrictions because our primary goal is to provide our users with the greatest design possible, one that is interactive and user-friendly with the fewest restrictions feasible.

**3.1.3 Programming language:**

MERN-Stack is used in development of this project, which includes:

* React JS
* Node JS
* Express JS
* MongoDB

**List of Requirement Category**

**Function Categories**

| **Functional**  **Requirements** | Registration, Login, Create Post, Update Post, Delete Post, Select Hostel, Make Payments |
| --- | --- |
| **Non-Functional**  **Requirements** | Database, data in form of visualization and create the post date. |
| **Data Requirements** | Data is stored in the form of structure and unstructured. |
| **Constraints** | Internet will be the main problem as this application will be online. |

**Table 3.1-1: List of Requirement Category**

**List of Requirements**

The following table shows the functional and non-functional requirements of our system.

It also shows the details require performing that requirement.

| **RID** | **description** | **Category** | **Attribute** | **Details & Boundary Constraints** |
| --- | --- | --- | --- | --- |
| **FR.1** | Users(Client, Owner)first need to register on our platform by giving valid email | Functional | Registration | * Enter Email * Email address * Enter name. * Enter password. |
| **FR.2** | Admin also need to sign  up on this application | Functional | Registration | * Enter Email * Enter name. * Enter password. |
| **FR.3** | Users (Client, Owner) after registering on our platform needs to login to the system by giving its email and password. | Functional | Login | * Enter Email * Enter name. * Enter password. |
| **FR.4** | Client can search for what he wants filling the required fields | Functional | Searching | Client can search about specific hostel he/she want to get live. |
| **FR.5** | Owner can create post by  filling the required fields | Functional | Post | Owner can make post about what he/she ca stitch. |
| **FR.6** | Client can make payment | Functional | Payment | Client first need to have valid account to send payment to our accounts |
| **FR.7** | Owner can receive  payment | Functional | Payment | Owner can receive payment of client from accounts after Providing service to Client. |
| **FR.8** | Owner/Admin can delete post | Functional | Post | Owner can delete the post which he has posted earlier |
| **FR.9** | Owner can Update Post | Functional | Post | Owner can update their post. |
| **FR.10** | User can search | functional | searching | User can search required hostel by applying different filter. |
| **FR.11** | Nearby location | functional | Location | User can pin his location and he/she will see the hostels available nearby. |
| **NFR.1** | We will save information of our Users | Non- Functional | Data Storage | * Save personal information of users. * Save posts information. |
| **NFR.2** | We are providing option so our users can see when post was created. | Non- Functional | Post date | * Post created time will be display. |
| **NFR.3** | We will maintain our system to keep it working or want to upgrade something. | Non- Functional | Maintenance | * Try to remove errors or any bugs. * Want to update something in the system. |

**Table 3.1-2: List of Requirement**

## List of Actors

There are three main actors of our system and application, the list of applications is given below:

1. **Client**

Customers are a crucial component of our system and they are the ones who decide what kind of hostel they wish to stay in.

1. **Hostel Owner**

The second significant element in our system is the owner of the hostel; they advertise and provide descriptions of the services they offer.

1. **Admin**

In our system, administrators play a key role. He will administer the system, work to eliminate violations, and oversee client and hostel owner action.

## List of use cases

According to the users, which will use our system, the number of use-cases are given below:

**UC-01 Sign up**: users can register themselves according to their own domain, so that they can access restricted system services.

**UC-02 Log in**: users can simply enter the system by providing their username and passwords.

**UC-03 Log out**: user can leave the system by maintaining security.

**UC-04 Create Hostel**: Hostel owners can create their Hostel on the system after Hostel approval.

**UC-05 Hostel Approval:** Hostel owner can request for Hostel approval. Hostel Approval can be accepted and rejected.

**UC-06 Account Management**: Admin has control over accounts to avoid any difficulty against accounts.

**UC-07 Create Portfolio:**  Hostel owner can create portfolios of products, services and perform create, read, update and delete operations the portfolios depict all the detail that should be included in the system.

**UC-08 Upload Multimedia Content:**  Hostel owner can upload, delete and view content of video, images and can use it to for the promotion of their hostels.

**UC-09 Search:** User can search services, products, hostels by their name, location, and ranking.

**UC-10 Appointment:** Customer can request for booking any service and product of any Hostel and Hostel owner can manage appointments according to their own demand.

**UC-11 Ratings & Comments:** Customer can give and delete comments and ratings of products, services and Hostels, and Hostel owner can view, reply the ratings and reviews provided by the customers.

**UC-12 Online Payment:** Customer can pay amount of the services through an online gateway.

## System use case diagram





## Extended use cases For Client

**Table 3.5-1: UC-01 Sign Up**

| **Use Case ID:** | UC-1 | | | |
| --- | --- | --- | --- | --- |
| **Use**  **Case Name:** | Sign Up | | | |
| **Created By:** | Mujtaba Mubasher  Ahsan Yousaf | | **Last Updated By:** | Mujtaba Mubasher  Ahsan Yousaf |
| **Date Created:** | 12/21/2022 | | **Last Revision Date:** |  |
| **Actors:** | | Client, Owner | | |
| **Description:** | | It is **necessary** for clients to **register** on our website. If he wants to **book a hostel**. | | |
| **Trigger:** | | If a client **visited** on to our website **first time** or **without have an account** then **Sign Up** or **Registration Popup box** will appear Which have button when the user or client click on that then website transfer a client to a **registration page** | | |
| **Preconditions:** | | The information entered during registration process should be correct.   1. Client must have valid **E-mail or Gmail account** 2. Client must have a valid **home address** 3. Customer must have a **Valid Phone Number** | | |
| **Post conditions:** | | After completing the registration process, user information should be stored in the database   1. **Verify** the **Email** address by sending the **OTP** 2. **Verify** the client **phone number** by sending the OTP 3. **Validate** Client **Passwords** **Match Or Not** | | |
| **Normal Flow:** | | User will click on the registration button.  Enter all the required information.  Our system will cross check the information entered by user.  Then save its information to the database. | | |
| **Alternative Flows:**  **[Alternative Flow 1 – Not in Network]** | | If the user email already exists during registration process, Use an another email or Login | | |
| **Exceptions:** | | 1. Registration **Approved** or **Successful** 2. We are **unable** to **register** your **account**. Please **Try Again Later** 3. Password **does not matc**h 4. Invalid **OTP** 5. Invalid **Email Address** | | |
| **Includes:** | | \_ | | |
| **Frequency of Use:** | | **1 Time for each actor** | | |
| **Special Requirements:** | | 1: Our System **Must check** or **validate their user network**.  2: Our System **must validate** the user **IP**. Is it **Real or Fake** | | |
| **Assumptions:** | | User can **register** using **phone number** in case of **no Email** | | |
| **Notes and Issues:** | | Without **Email verification** user can’t book any hostel from website. | | |

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Table 3.5-2: UC-02 Login**

| **Use Case ID:** | UC-2 | | | |
| --- | --- | --- | --- | --- |
| **Use**  **Case Name:** | Login | | | |
| **Created By:** | Mujtaba Mubasher,  Ahsan Yousaf | | **Last Updated By:** | Mujtaba Mubasher,  Ahsan Yousaf |
| **Date Created:** | 12/21/2022 | | **Last Revision Date:** | 05/02/2023 |
| **Actors:** | | Client, Owner | | |
| **Description:** | | It is **necessary** for clients to **Login** on our website. If he wants to **book a hostel**. | | |
| **Trigger:** | | If a client **visited** on to our website **without Login into an account,** then **Login** or **Popup box** will appear Which have button when the user or client click on that then website transfer a client to a **Login Page** | | |
| **Preconditions:** | | * Client must have valid **E-mail account**   **OR**   * Customer must have a **Valid Phone Number** * Client must have a valid **Password** | | |
| **Post conditions:** | | * **Verified** the **Email** address by sending the **OTP**   **OR**   * **Verified** the client **phone number** by sending the **OTP** * **Validate** Client **Passwords** **Match or Not** | | |
| **Normal Flow:** | | **Note:** Client has **Two** Option for **Login** with **Email / Phone No**   1. Client enters their **Email address**   **OR**   1. Client enters their **Phone Number** 2. Client enters their **Password** 3. Client **clicks** on the **Login Button** | | |
| **Alternative Flows:**  **[Alternative Flow 1 – Not in Network]** | | If user does not have an account, then register on our website. | | |
| **Exceptions:** | | * Login **Approved** or **Successful** * We are **unable** to **Login** intoyour **account**. Please **Try Again Later** * Invalid **Password** * Invalid **OTP** * Invalid **Email Address** | | |
| **Includes:** | | Sign Up | | |
| **Frequency of Use:** | | **It appears every time when actor logout or session expired.** | | |
| **Special Requirements:** | | 1: Our System **Must check** or **validate their user network**.  2: Our System **must validate** the user **IP**. Is it **Real or Fake** | | |
| **Assumptions:** | | User can **Login** using **phone number** in case of **no Email** | | |
| **Notes and Issues:** | | Without **Email verification** user can’t book any hostel from website. | | |

**Table 3.5-3: UC-03 Logout**

| **Use Case ID:** | UC-3 | | | |
| --- | --- | --- | --- | --- |
| **Use Case Name:** | Logout | | | |
| **Created By:** | Mujtaba Mubasher,  Ahsan Yousaf | | **Last Updated By:** | Mujtaba Mubasher,  Ahsan Yousaf |
| **Date Created:** | 21/12/2022 | | **Last Revision Date:** | 05/02/2023 |
| **Actors:** | | Hostel Owner, Customer, Administrator | | |
| **Description:** | | The actors will be able to log out of the system after using it. | | |
| **Trigger:** | | When actors want to log out after using the functionalities that are provided by the system. | | |
| **Precondition:** | | Actors must be logged in to the system. | | |
| **Post condition:** | | Actors then successfully log out from the system and the Login page will be shown. | | |
| **Normal Flow:** | | 1. Actors will press the Log out button. 2. The system session will be dismissed. 3. System will redirect the actors to the login page. | | |
| **Alternative Flows:**  **[Alternative Flow 1 –Not in Network]** | | 1. System faces a connectivity issue and the session not ended. 2. System prompt “System Connection Error”. | | |
| **Exceptions:** | | 1. The customer doesn’t click the Logout button 2. Actors will not log out from the system successfully | | |
| **Includes:** | | Sign up and Login | | |
| **Frequency of Use:** | | Many times, whenever actors want to logout from the system to end its system  Session. | | |
| **Special Requirements:** | | N/A | | |
| **Assumptions:** | | The Hostel owner has an active Internet Access | | |
| **Notes and Issues:** | | N/A | | |

**Table 3.5-4: UC-04 Create Hostel**

| **Use Case ID:** | UC-4 | | | |
| --- | --- | --- | --- | --- |
| **Use Case Name:** | Create Hostels | | | |
| **Created By:** | Mujtaba Mubasher  Ahsan Yousaf | | **Last Updated By:** | Mujtaba Mubasher,  Ahsan Yousaf |
| **Date Created:** | 21/12/2022 | | **Last Revision Date:** | 05/02/2023 |
| **Actor:** | | Hostel Owner, Administrator | | |
| **Description:** | | The Hostel owner will be able to create a hostel on the system | | |
| **Trigger:** | | When the Hostel owner wants to grow its Hostel business and wants to sell its services through an online platform. | | |
| **Precondition:** | | The Hostel owner must have a registered account. | | |
| **Post condition:** | | The Hostel owner then successfully creates a hostel after hostel approval acceptance. | | |
| **Normal Flow:** | | 1. Hostel owner will log in to the system with valid credentials and redirect to the dashboard. 2. Click on Create Hostel. 3. A dialogue box appears that depicts to Hostel owner to mention    1. Hostel name    2. Contact Details    3. Hostel address    4. Hostel Description    5. reservation details    6. Prices detail    7. Services type and details    8. Upload Multimedia Content 4. Actor will enter the Submit button. 5. The Hostel approval request will be forwarded to the Administrator. 6. Hostel owner will receive confirmation email or status. 7. Administrator will click on the Accept Hostel Approval. 8. Hostel created successfully. | | |
| **Alternative Flows:**  **[Alternative Flow 1 – Not in Network]** | | 1a. Username or password is not valid.   1. System prompt “Invalid username or password”.   3.1.a. The Hostel name field is left empty.   1. System prompt “Fill the empty field”.   3.2.a. The contact details field is left empty.   1. System prompt “Fill the empty field”.   3.3.a. The Hostel address field is left empty.   1. System prompt “Fill the empty field”.   3.4.a. The appointment details field is left empty.   1. System prompt “Fill the empty field”.   3.5.a. The price details field is left empty.   1. System prompt “Fill the empty field”.   3.6.a. Services type and details field are left empty.   1. System prompt “Fill the empty field”.   6a. Hostel approval rejected.   1. System prompt “Something went wrong”. | | |
| **Exceptions:** | | 4a. The actor doesn’t press the Submit Button   1. Hostel not created successfully. | | |
| **Includes:** | | Registered as Hostel Owner, login, Hostel Approval | | |

| **Use Case ID:** | | UC-4 | | |
| --- | --- | --- | --- | --- |
| **Use Case Name:** | | Create Hostels | | |
| **Created By:** | | Mujtaba Mubasher,  Ahsan Yousaf | **Last Updated By:** | Mujtaba Mubasher,  Ahsan Yousaf |
| **Date Created:** | | 21/12/2022 | **Last Revision Date:** | 05/02/2023 |
| **Frequency of Use:** | Only once per user | | | |
| **Special Requirements:** | N/A | | | |
| **Assumptions:** | Hostel owners have an Active Internet Access | | | |
| **Notes and Issues:** | N/A | | | |

**Table 3.5-5: UC-05 Hostel Approval**

| **Use Case ID:** | UC-05 | | | |
| --- | --- | --- | --- | --- |
| **Use Case Name:** | Hostel Approval | | | |
| **Created By:** | Mujtaba Mubasher,  Ahsan Yousaf | | **Last Updated By:** | Mujtaba Mubasher,  Ahsan Yousaf |
| **Date Created:** | 21/12/2022 | | **Last Revision Date:** | 05/02/2023 |
| **Actor:** | | Administrator, Hostel Owner | | |
| **Description:** | | The Administrator will be able to accept or reject hostel approval provided by the Hostel Owner. | | |
| **Trigger:** | | When the Hostel Owner creates hostel on the system. | | |
| **Precondition:** | | The Administrator have to login to the system. | | |
| **Post condition:** | | Produce Hostel approval status successfully | | |
| **Normal Flow:** | | 1. Administrator will login to the system using valid credentials that redirect it to Dashboard. 2. Administrator will accept hostel approval. 3. Administrator will reject hostel approval. 4. Administrator will send approval status to the hostel owner. | | |
| **Alternative Flows:**  **[Alternative Flow 1 – Not in Network]** | | 1a. Username or password is incorrect.   * System prompt “Invalid username or password”. | | |
| **Exceptions:** | | N/A. | | |
| **Includes:** | | Login | | |
| **Frequency of Use:** | | Only one time per request. | | |
| **Special Requirements:** | | N/A | | |
| **Assumptions:** | | Administrator face no technical issues | | |
| **Notes and Issues:** | | N/A | | |

**Table 3.5-6: UC-06 Account Management**

| **Use Case ID:** | UC-6 | | | |
| --- | --- | --- | --- | --- |
| **Use Case Name:** | Account Management | | | |
| **Created by:** | Mujtaba Mubasher,  Ahsan Yousaf | | **Last Updated By:** | Mujtaba Mubasher,  Ahsan Yousaf |
| **Date Created:** | 21/12/2022 | | **Last Revision Date:** | 05/02/2023 |
| **Actor:** | | Administrator | | |
| **Description:** | | The actor will be able to add, verify, restrict, block, and delete the user accounts. | | |
| **Trigger:** | | The situation arose those challenges or violates the security of the system or for monitoring the account details. | | |
| **Precondition:** | | For monitoring purposes or if any user violates the security of the system or a request is generated by the Hostel Owner to authenticate its hostel account | | |
| **Post condition:** | | The actors then successfully manage the system accounts. | | |
| **Normal Flow:** | | 1. The administrator will log in to the system to manage the account. 2. The actor will verify the account.    1. Administrator will verify the account of the Hostel owner.       1. Administrator will verify Hostel Name       2. Administrator will verify Hostel Address       3. Administrator will verify the Contact Number       4. Administrator will verify the Email Address       5. Administrator will verify the Password    2. Administrator will verify the Customer account.       1. Administrator will verify the Contact Number       2. Administrator will verify the Email Address       3. Administrator will verify the Password 3. Administrator will send a verification email. 4. The actor will restrict the account. 5. The actor will block the account. 6. The actor will delete the account. 7. Account managed successfully | | |
| **Alternative Flows:**  **[Alternative Flow 1 – Not in Network]** | | 7a. System faces a maintenance activity.   * System prompt “System Maintenance Activity is in progress”. | | |
| **Exceptions:** | | N/A | | |
| **Includes:** | | Login | | |
| **Frequency of Use:** | | Many times, depending on the situation. | | |
| **Special Requirements:** | | N/A | | |
| **Assumptions:** | | No technical Issues | | |
| **Notes and Issues:** | | N/A | | |

**Table 3.5-7: UC-07 Create Portfolio**

| **Use Case ID:** | UC-7 | | | |
| --- | --- | --- | --- | --- |
| **Use Case Name:** | Create Portfolio | | | |
| **Created By:** | Mujtaba Mubasher,  Ahsan Yousaf | | **Last Updated By:** | Mujtaba Mubasher,  Ahsan Yousaf |
| **Date Created:** | 21/12/2022 | | **Last Revision Date:** | 05/02/2023 |
| **Actor:** | | Hostel Owner | | |
| **Description:** | | The Hostel owner will be able to create a Portfolio of services and products on the system. | | |
| **Trigger:** | | When the Hostel owner wants to create a portfolio of its Hostel services, Hostel and products. | | |
| **Precondition:** | | The Hostel owner must be login to the system. | | |
| **Post condition:** | | The Hostel owner then successfully creates a portfolio of services, products and Hostel. | | |
| **Normal Flow:** | | 1. Hostel owner will log in to the system with valid credentials and redirect to the dashboard page. 2. Click on Create Portfolio. 3. Choose either a Service portfolio, Product Portfolio, or Hostel Portfolio.   3.1. Suppose the Hostel owner chooses Service Portfolio then a dialogue box will appear that depicts to Hostel owner to mention   * 1. Service name   2. Service Description   3. Appointment details   4. Price details   5. Services type   6. Upload Multimedia Content   7. Actor will enter the Update button.   8. Service Portfolio created successfully.   3.2. Suppose the Hostel owner chooses Product Portfolio then a dialogue box will appear that depicts to Hostel owner to mention   1. Product name 2. Product Description 3. Price details 4. Actor will enter the Update button. 5. Product Portfolio created successfully.   3.3. Suppose the Hostel owner chooses Hostel Portfolio then a dialogue box will appear that depicts to Hostel owner to mention   * 1. Hostel name   2. Hostel Description   3. Appointment details   4. Price details   5. Services type   6. Upload Multimedia Content   7. Actor will enter the Update button.   8. Hostel Portfolio created successfully.  1. After updating desired Portfolio, the system will show Dashboard. | | |
| **Alternative Flows:**  **[Alternative Flow 1 – Not in Network]** | | 1a. Username or password is incorrect.   1. System prompt “Invalid username or password”. | | |

| **Use Case ID:** | UC-7 | | | |
| --- | --- | --- | --- | --- |
| **Use Case Name:** | Create Portfolio | | | |
| **Created By:** | Mujtaba Mubasher,  Ahsan Yousaf | | **Last Updated By:** | Mujtaba Mubasher,  Ahsan Yousaf |
| **Date Created:** | 21/12/2022 | | **Last Revision Date:** | 05/02/2023 |
| **Exceptions:** | | 4a. The actor doesn’t press the Update Button   1. Portfolio not created successfully. | | |
| **Includes:** | | Login | | |
| **Frequency of Use:** | | Many Times | | |
| **Special Requirements:** | | N/A | | |
| **Assumptions:** | | Hostel owners have an Active Internet Access | | |
| **Notes and Issues:** | | N/A | | |

**Table 3.5-8: UC-08 Upload Multimedia Content**

| **Use Case ID:** | UC-8 | | | |
| --- | --- | --- | --- | --- |
| **Use Case Name:** | Upload Multimedia Content | | | |
| **Created By:** | Mujtaba Mubasher,  Ahsan Yousaf | | **Last Updated By:** | Mujtaba Mubasher,  Ahsan Yousaf |
| **Date Created:** | 21/12/2022 | | **Last Revision Date:** | 05/02/2023 |
| **Actor:** | | Hostel Owner | | |
| **Description:** | | The Hostel owner will be able to upload multimedia content on the system. | | |
| **Trigger:** | | When the Hostel owner wants to upload audio, video, and ads to advertise its Hostel. | | |
| **Precondition:** | | The Hostel owner must be login to the system. | | |
| **Post condition:** | | The Hostel owner then successfully uploads the multimedia content on the system. | | |
| **Normal Flow:** | | 1. Hostel owner will log in to the system with valid credentials and redirect to the dashboard page. 2. Click on the Upload Button. 3. Select either audio, video, or image. 4. Multimedia content uploaded successfully. 5. After uploading multimedia content the system will show Dashboard. | | |
| **Alternative Flows:**  **[Alternative Flow 1 – Not in Network]** | | 1a. Username or password is incorrect.   1. System prompt “Invalid username or password”. | | |
| **Exceptions:** | | 4a. The actor doesn’t press the Upload Button   1. Content not uploaded successfully. | | |
| **Includes:** | | Login | | |

|  | UC-8 | | | |
| --- | --- | --- | --- | --- |
| **Use Case Name:** | Upload Multimedia Content | | | |
| **Created By:** | Mujtaba Mubasher,  Ahsan Yousaf | | **Last Updated By:** | Mujtaba Mubasher,  Ahsan Yousaf |
| **Date Created:** | 21/12/2022 | | **Last Revision Date:** | 05/02/2023 |
| **Special Requirements:** | | N/A | | |
| **Assumptions:** | | Hostel owners have an Active Internet Access | | |
| **Notes and Issues:** | | N/A | | |

**Table 3.5-9: UC-09 Search**

| **Use Case ID:** | UC-9 | | | |
| --- | --- | --- | --- | --- |
| **Use Case Name:** | Search | | | |
| **Created By:** | Mujtaba Mubasher,  Ahsan Yousaf | | **Last Updated By:** | Mujtaba Mubasher,  Ahsan Yousaf |
| **Date Created:** | 21/12/2022 | | **Last Revision Date:** | 05/02/2023 |
| **Actor:** | | Customer | | |
| **Description:** | | The customer will be able to search for services, products, and Hostels on the system. | | |
| **Trigger:** | | When the customer wants to use the services of desired Hostels | | |
| **Precondition:** | | The customer should be log in to the system. | | |
| **Post condition:** | | The customers then successfully search for services provided by the system. | | |
| **Normal Flow:** | | 1. Customer will login to the system. 2. Customer will click on the search button 3. Customer will search Hostel by name, location, Ranking, 4. Customer will successfully search Hostel. | | |
| **Alternative Flows:**  **[Alternative Flow 1 – Not in Network]** | | 1a. Username or password is incorrect.   1. System prompt “Invalid username or password”. | | |
| **Exceptions:** | | 2a. The actor doesn’t press the search Button   1. Customer will not search Hostel successfully. | | |
| **Includes:** | | Login | | |
| **Frequency of Use:** | | Many Times | | |
| **Special Requirements:** | | N/A | | |
| **Assumptions:** | | Customers have an Active Internet Access | | |
| **Notes and Issues:** | | N/A | | |

**Table 3.5-10: UC-10 Reservation**

| **Use Case ID:** | | UC-10 | | |
| --- | --- | --- | --- | --- |
| **Use Case Name:** | | Reservation | | |
| **Created By:** | | Mujtaba Mubasher,  Ahsan Yousaf | **Last Updated By:** | Mujtaba Mubasher,  Ahsan Yousaf |
| **Date Created:** | | 21/12/2022 | **Last Revision Date:** | 05/02/2023 |
| **Actors:** | Customer, Hostel Owner | | | |
| **Description:** | The customer will request for appointment after confirming it with the Hostel owner. | | | |
| **Trigger:** | When the customer wants to use the services of the selected Hostel | | | |
| **Precondition:** | The customer should log in to the system for requesting an appointment. | | | |
| **Post condition:** | The customers then successfully make an appointment. | | | |
| **Normal Flow:** | 1. Customer will login to the system with valid credentials and redirect to the dashboard page. 2. Customer will click on the Make an Appointment button 3. Customer will Select the time provided by the Hostel owner, 4. Hostel owner will confirm the appointment. 5. Customer gets the confirmation email. 6. Customer appointment is confirmed successfully. | | | |
| **Alternative Flows:**  **[Alternative Flow 1 – Not in Network]** | 5a. Customer didn’t get confirmation mail or appointment is not confirmed.   1. System prompt “Appointment Request Canceled”. | | | |
| **Exceptions:** | 1a. Username or password is incorrect.   1. System prompt “Invalid username or password”.   2a. The customer doesn’t press the Make an appointment Button   1. Customer will not make an appointment successfully. | | | |
| **Includes:** | Login | | | |
| **Frequency of Use:** | Many Times | | | |
| **Special Requirements:** | N/A | | | |
| **Assumptions:** | Actors have an Active Internet Access | | | |
| **Notes and Issues:** | N/A | | | |

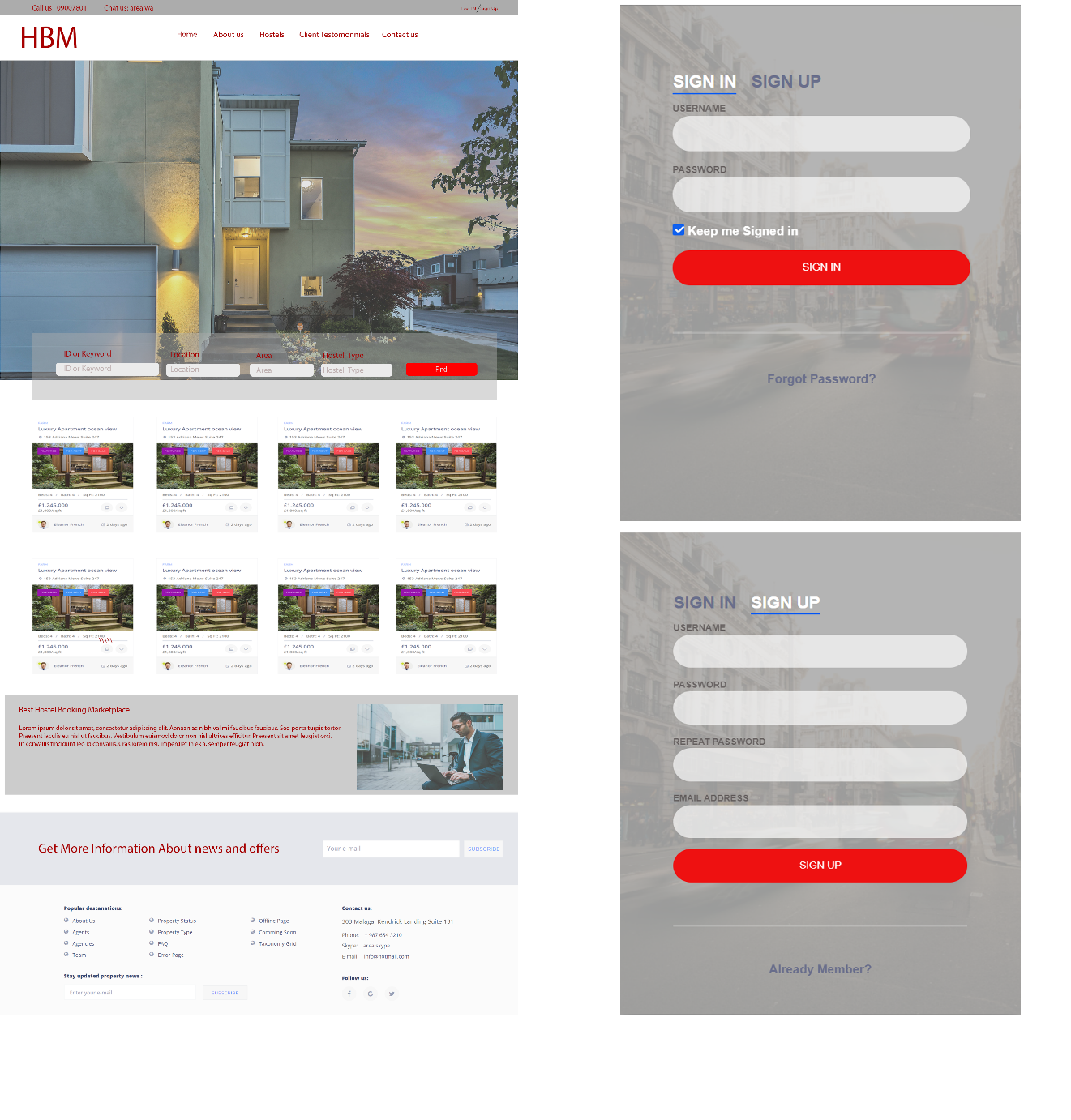
**Table 3.5-11: UC-11 Ratings & Comments**

| **Use Case ID:** | UC-11 | | | |
| --- | --- | --- | --- | --- |
| **Use Case Name:** | Ratings & Comments | | | |
| **Created By:** | Mujtaba Mubasher,  Ahsan Yousaf | | **Last Updated By:** | Mujtaba Mubasher,  Ahsan Yousaf |
| **Date Created:** | 21/12/2022 | | **Last Revision Date:** | 05/02/2023 |
| **Actors:** | | Customer, Hostel Owner | | |
| **Description:** | | The customer may give ratings and comments to a Hostel and the Hostel owner can view it and reply it. | | |
| **Trigger:** | | When the customer wants to share the service experience provided by the selected Hostel. | | |
| **Precondition:** | | N/A | | |
| **Post condition:** | | The customers then successfully give ratings and comments and the Hostel owner will successfully respond to them. | | |
| **Normal Flow:** | | 1. Customer will log in to the system. 2. Customer will select the Hostel for rating and comment. 3. Customer will rate the Hostel by selecting a rating from 0 to 5. 4. Customer may also give comments & ratings. 5. Customer then successfully gives ratings and comments to the selected Hostel. 6. Customer can also delete comments & ratings 7. Customer then successfully delete ratings and comments to the selected Hostel. 8. Hostel owner will log in to the system to see the ratings and comments. 9. Hostel owners may click on the reply button to respond to the comments. 10. Ratings and comments are successfully viewed and comments are successfully replied. | | |
| **Alternative Flows:**  **[Alternative Flow 1 – Not in Network]** | | 5a. Customer didn’t get confirmation mail or appointment is not confirmed.   1. System prompt “Appointment Request Canceled”. | | |
| **Exceptions:** | | 1a. Username or password is incorrect.   1. System prompt “Invalid username or password”. | | |
| **Includes:** | | Login | | |
| **Frequency of Use:** | | Many Times | | |
| **Special Requirements:** | | N/A | | |
| **Assumptions:** | | Actors have an Active Internet Access | | |
| **Notes and Issues:** | | N/A | | |

**Table 3.5-12: UC-12 Online Payment**

| **Use Case ID:** | UC-12 | | | |
| --- | --- | --- | --- | --- |
| **Use Case Name:** | Online Payment | | | |
| **Created By:** | Mujtaba Mubasher,  Ahsan Yousaf | | **Last Updated By:** | Mujtaba Mubasher,  Ahsan Yousaf |
| **Date Created:** | 21/12/2022 | | **Last Revision Date:** | 05/02/2023 |
| **Actor:** | | Customer | | |
| **Description:** | | The customer will be able to make online payments for Services. | | |
| **Trigger:** | | When the customer wants to use the services of desired Hostels and make an online payment. | | |
| **Precondition:** | | The customer should log in to the system. | | |
| **Post condition:** | | The customers then successfully make an online payment. | | |
| **Normal Flow:** | | 1. Customer will login to the system using valid credentials that redirect it to Dashboard. 2. Customer will click on the Payment button 3. Customer will enter credit card or Debit Card Details i.e., card number Owner name, 4. Customer will press the Finish Button. 5. Amount will be deducted according to service charges. 6. Online payment performed successfully to Hostel owner. | | |
| **Alternative Flows:**  **[Alternative Flow 1 – Not in Network]** | | 1a. Username or password is incorrect.   1. System prompt “Invalid username or password”.   3a. Invalid card number or owner name.   1. System prompt “Something went wrong”. | | |
| **Exceptions:** | | 2a. The actor doesn’t press the Finish Button   1. Customer will not make online payment successfully. | | |
| **Includes:** | | Login | | |
| **Frequency of Use:** | | Only one time per service. | | |
| **Special Requirements:** | | N/A | | |
| **Assumptions:** | | Customers have an Active Internet Access | | |
| **Notes and Issues:** | | N/A | | |

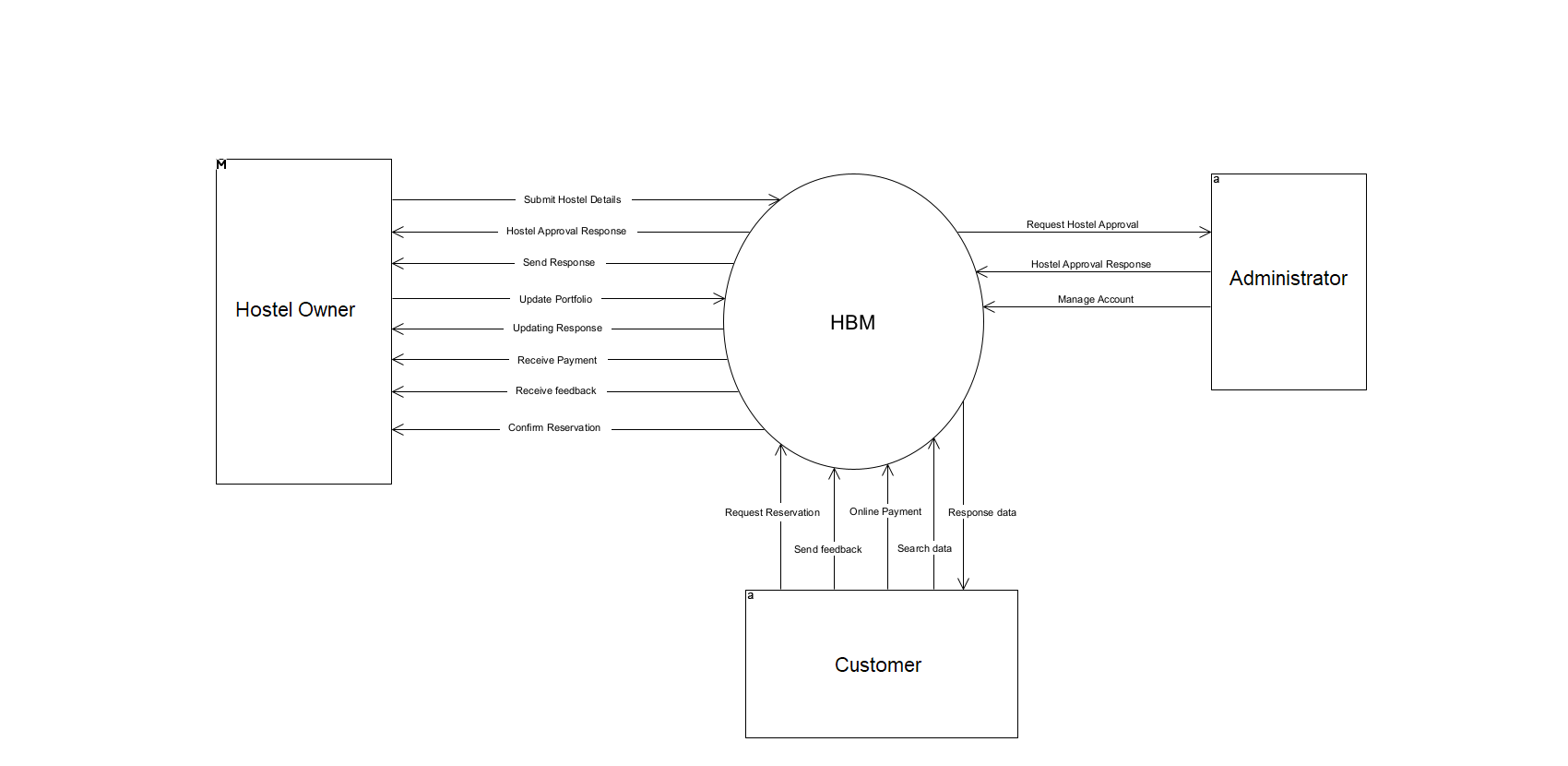
## User interfaces (mock screens)



**Prototype1: (P1) register a new member and home page**

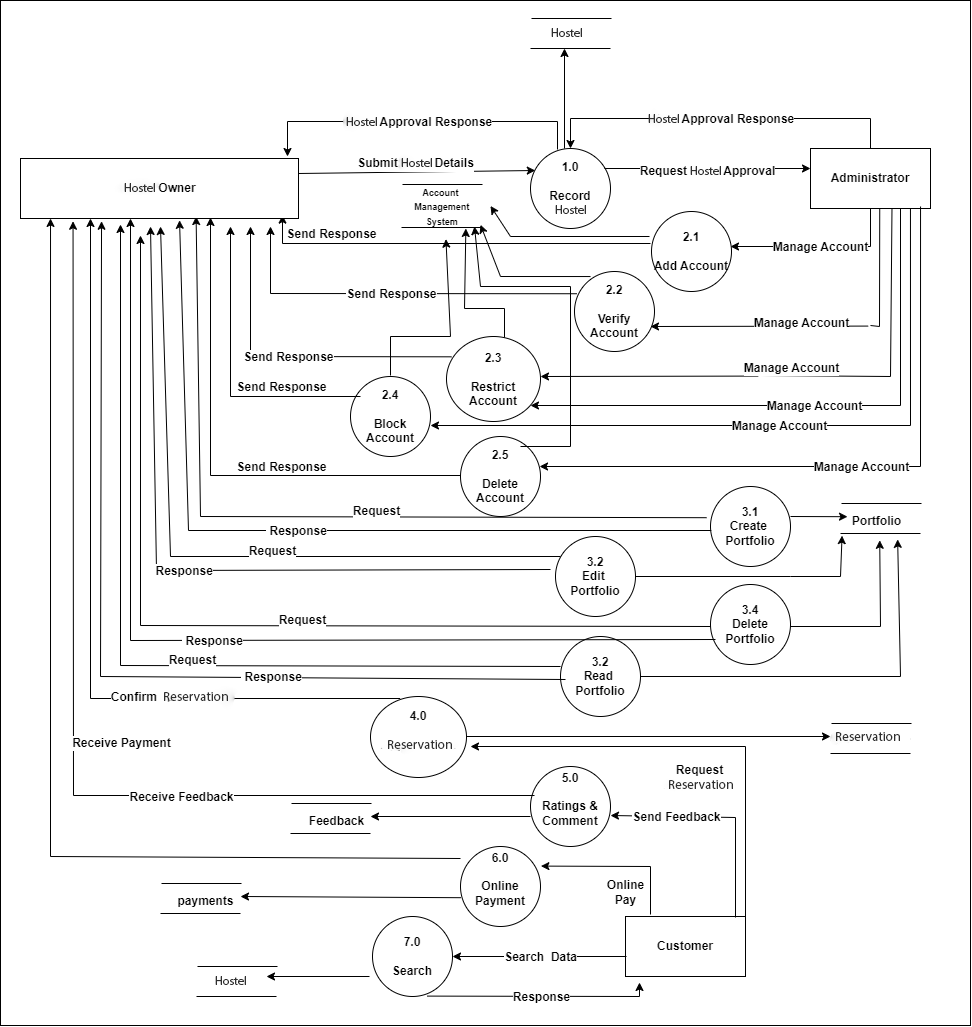
# Data flow diagram (optional)

## Data Flow Diagram Level 0



## Data Flow Diagram Level 1

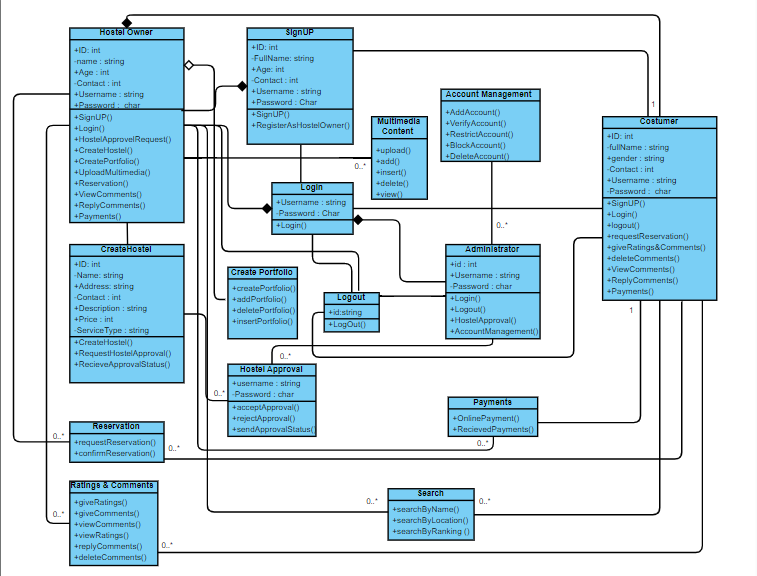
## 4.3 Data Flow Diagram Level 2



# System Design

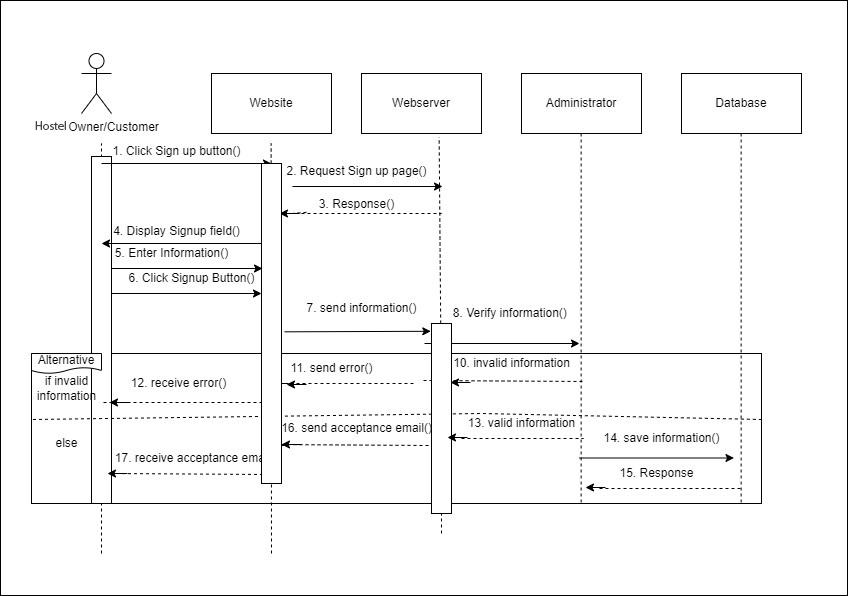
## System Architecture Diagram

## Class Diagram

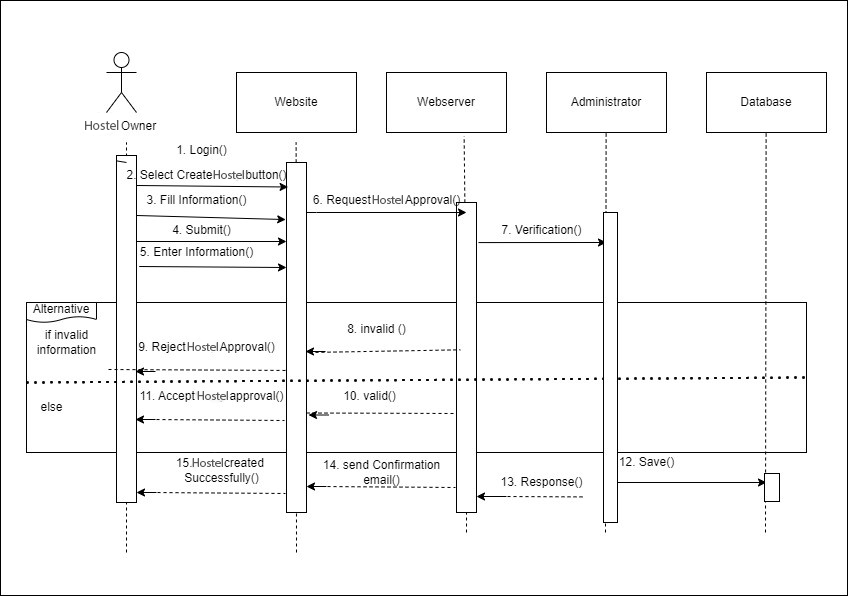


## Sequence Diagrams

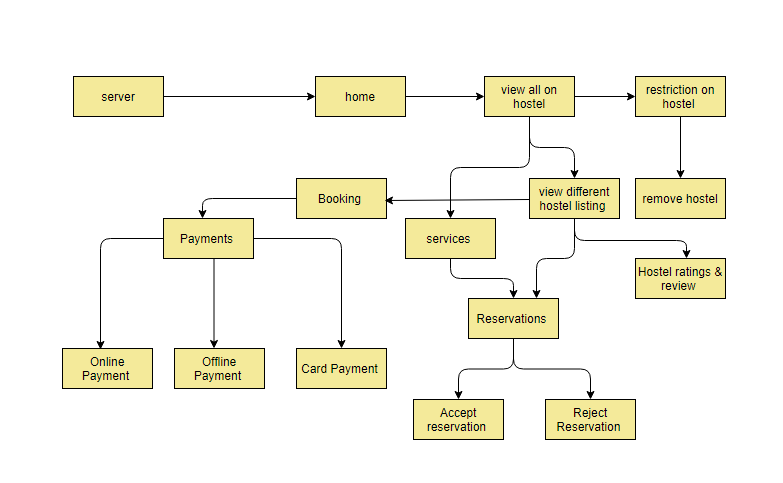
Sign Up







## Collaboration Diagrams



## ERD

## Data Dictionary

**Table Name:** Customer

Primary Key: C\_ID (Auto incremented)

**Table 5.6-1: Data Dictionary 5.6.1- Customer**

| **Element Name** | **Type** | **Validation** | **Mandatory** | **Remarks** |
| --- | --- | --- | --- | --- |
| C\_ID | int |  | Yes |  |
| C\_Username | char |  | Yes |  |
| Email | email |  | Yes |  |
| Contact | int |  | Yes |  |
| Address | char |  | Yes |  |

**Table Name:** Hostel Owner

Primary Key: H\_ID (Auto incremented)

**Table 5.6-2: Data Dictionary 5.6.2- Hostel Owner**

| **Element Name** | **Type** | **Validation** | **Mandatory** | **Remarks** |
| --- | --- | --- | --- | --- |
| H\_Id | int |  | Yes |  |
| Phone\_Number | int |  | Yes |  |
| H\_Username | char |  | Yes |  |
| Address | char |  | Yes |  |

**Table Name:** Payment

Primary Key: P\_ID (Auto incremented)

Foreign Key: C\_ID

Foreign Key: H\_ID

**Table 5.6-3: Data Dictionary 5.6.3- Payment**

| **Element Name** | **Type** | **Validation** | **Mandatory** | **Remarks** |
| --- | --- | --- | --- | --- |
| P\_Id | int |  | Yes |  |
| Contact | int |  | Yes |  |
| Username | char |  | Yes |  |
| Address | char |  | Yes |  |
| Date/time | Datetime |  | Yes |  |
| C\_ID | int |  | Yes |  |
| H\_ID | int |  | Yes |  |
| Amount | int |  | yes |  |

**Table Name:** Reservation

Primary Key: R\_ID (Auto incremented)

Foreign Key: C\_ID

Foreign Key: H\_ID

**Table 5.6-4: Data Dictionary 5.6.4- Reservation**

| **Element Name** | **Type** | **Validation** | **Mandatory** | **Remarks** |
| --- | --- | --- | --- | --- |
| R\_Id | int |  | Yes |  |
| Contact | int |  | Yes |  |
| Username | char |  | Yes |  |
| Address | char |  | Yes |  |
| Date/time | Datetime |  | Yes |  |
| C\_ID | int |  | Yes |  |
| H\_ID | int |  | Yes |  |

**Table Name:** Services

Primary Key: S\_ID (Auto incremented)

Foreign Key: C\_ID

**Table 5.6-5: Data Dictionary 5.6.5- Services**

| **Element Name** | **Type** | **Validation** | **Mandatory** | **Remarks** |
| --- | --- | --- | --- | --- |
| S\_ID | int |  | Yes |  |
| C\_ID | int |  | yes |  |
| C\_Username | char |  | Yes |  |
| Email | email |  | Yes |  |
| Contact | int |  | Yes |  |
| Address | char |  | Yes |  |

**Table Name:** Booking

Primary Key: B\_ID (Auto incremented)

Foreign Key: C\_ID

Foreign Key: H\_ID

**Table 5.6-6: Data Dictionary 5.6.6- Booking**

| **Element Name** | **Type** | **Validation** | **Mandatory** | **Remarks** |
| --- | --- | --- | --- | --- |
| B\_Id | int |  | Yes |  |
| Contact | int |  | Yes |  |
| B\_Username | char |  | Yes |  |
| Address | char |  | Yes |  |
| Date/time | Datetime |  | Yes |  |
| C\_ID | int |  | Yes |  |
| H\_ID | int |  | Yes |  |

**Table Name:** Admin

Primary Key: A\_ID (Auto incremented)

Foreign Key: C\_ID

Foreign Key: H\_ID

**Table 5.6-7: Data Dictionary 5.6.7- Admin**

| **Element Name** | **Type** | **Validation** | **Mandatory** | **Remarks** |
| --- | --- | --- | --- | --- |
| A\_ID | int |  | Yes |  |
| A\_Username | char |  | Yes |  |
| C\_ID | Int |  | Yes |  |
| H\_ID | Int |  | Yes |  |

# Implementation details

## Development Setup

The platform we choose to work on with is visual studio and we used:

* React.JS for frontend
* Node.js for backend
* MongoDB for database

Following are the APIs which we have used:

* geolocator: ^9.0.2

Currently, the project is hosted on a physical laptop device because the website is still in the testing phase but after completion, it will be available on the Google and will be available for used.

## Algorithms

**Code Snippets**

**Home Screen** 



**Search Filters**

**Add hostel**



**Admin**

**Login / Signup**



## 6.4 Constraints

## 6.4.1 Assumptions

* Users will have access to a reliable internet connection to use the website.
* Users will have smartphones or devices compatible with the website.
* hostel owners and admins will have the necessary technical infrastructure to receive and respond to service requests.
* The website will primarily focus on connecting users with hostels within a specified radius.

## 6.4.2 System constraints

* The website must be compatible with the devices supported by the target platforms google.
* The website should have a user-friendly interface.
* The website should be able to accurately determine the user's location and provide relevant search results based on proximity.

## 6.4.3 Restrictions

These restrictions must be kept in mind while using this app:

* Internet connection must be available on your device to use the website.
* mobile and laptop/pc will only support this website.
* Users must know how to use website functionalities.

## 6.4.4 Limitations

The website may not be able to guarantee the availability of hostels at all times, as it relies on their participation and availability within the platform.

# TESTING

## Extended Test Cases

**Table 7.1-1 sign up testing**

| **Test Case ID:001** | | | | **Test Designed by: ahsan yousaf** | | | |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Test Module Name:** Signup Testing. | | | | **Test Designed Date:** 01/6/2023 | | | |
| **Test Priority:** High | | | | **Test Executed by: Ahsan yousaf** | | | |
| **Test Title Name:** To test the signup functionality. | | | | **Test Executed Date:01/6/2023** | | | |
| **Description:** Check system behavior when valid text field information is filled for signing up. | | | |  | | | |
| **Pre-Conditions:** The user has internet access to make an account. | | | | | | | |
| **Dependencies:** | | | | | | | |
| **Step** | **Test Step** | **Test Data** | **Expected Result** | | **Actual Result** | **Status(Pass/Fail)** | **Notes** |
| **1.** | Enter Username |  |  | |  | **Pass** |  |
| **2.** | Enter Password. |  |  | |  |
| **3.** | Enter Confirm Password. | Username/Email = ahsan@gmail.com  Password = ah1234 |  | |  |
| **4.** | Enter Phone no | Phone no: 03203124589 |  | |  |  |  |
| **5.** | Enter Email | ahsan@gmail.com |  | |  |  |  |
| **6.** | Click Submit/Login button. |  | The User will be able to sign up for his/her new account. | | The User will be able to sign up for his/her new account. |  |  |
| **Post-Conditions:** The user receives a code of confirmation given email/phone number. | | | | | | | |

**Table 7.1-2 Test case- log in**

| **Test Case ID:002** | | | | **Test Designed by: ahsan yousaf** | | | |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Test Module Name:** Login Testing. | | | | **Test Designed Date:** 01/6/2023 | | | |
| **Test Priority:** High | | | | **Test Executed by: ahsan yousaf** | | | |
| **Test Title Name:** To test the login functionality. | | | | **Test Executed Date:** 01/6/2023 | | | |
| **Description:** Check system behavior when a valid email & password is entered. | | | |  | | | |
| **Pre-Conditions:** The user has access to the internet and must have entered a valid email and password. | | | | | | | |
| **Dependencies:** | | | | | | | |
| **Step** | **Test Step** | **Test Data** | **Expected Result** | | **Actual Result** | **Status(Pass/Fail)** | **Notes** |
| **1.** | Enter your Username/ Email. |  |  | |  | **Pass** |  |
| **2.** | Enter Password. |  |  | |  |
| **3.** | Enter Confirm Password. | Username/Email = ahsan@gmail.com  Password = 11223344 |  | |  |
| **4.** | Click Submit/Login button. |  | The User will be able to log into his/her account. | | The User will be able to log into his/her account. |  |  |
| **Post-Conditions:** The user must have an email account to log in. | | | | | | | |

**Table 7.1-3 test case – current location**

| **Test Case ID:003** | | | | **Test Designed by: ahsan yousaf** | | | |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Test Module Name:** Location Testing | | | | **Test Designed Date:** 01/6/2023 | | | |
| **Test Priority:** High | | | | **Test Executed by: ahsan yousaf** | | | |
| **Test Title Name:** To test the current location. | | | | **Test Executed Date:**01/6/2023 | | | |
| **Description:**  The customer opens a live location and tracks the hostel location, which set by the hostel. | | | |  | | | |
| **Pre-Conditions:** The user must have the application and the user must have an account. | | | | | | | |
| **Dependencies:** | | | | | | | |
| **Step** | **Test Step** | **Test Data** | **Expected Result** | | **Actual Result** | **Status(Pass/Fail)** | **Notes** |
| **1.** | Log in to your account. |  |  | |  | **Pass** |  |
| **2.** | Enter his/her current location. |  | The user enters his/her location successfully. | | The user enters his/her location successfully. |
| **3.** | Choose hostel |  |  | |  |
| **Post-Conditions:** The user is logged in to the system. The user has access to the functions of the system | | | | | | | |

**Table 7.1-4 test case- nearest hostel**

| **Test Case ID:004** | | | | | **Test Designed by: ahsan yousaf** | | |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Test Module Name:** Nearest hostel. | | | | | **Test Designed Date:** 01/6/2023 | | |
| **Test Priority:** High | | | | | **Test Executed by: ahsan yousaf** | | |
| **Test Title Name:** To test the nearest hostel functionality. | | | | | **Test Executed Date:**01/6/2023 | | |
| **Description:** Customer login to their account. He/she searches for the nearest hostel. Select hostel. Submit | | | | |  | | |
| **Pre-Conditions:** The user must have the website and the user must have an account. | | | | | | | |
| **Dependencies:** | | | | | | | |
| **Step** | **Test Step** | **Test Data** | **Expected Result** | **Actual Result** | | **Status(Pass/Fail)** | **Notes** |
| **1.** | Enter his/her current location. |  |  |  | | **Pass** |  |
| **2.** | Choose hostel | Hostel |  |  | |
| **3.** | The nearest hostel will be shown on the map. |  | Users can find the hostel successfully by entering their location. | Users can find the hostel successfully by entering their location. | |
| **4.** | Search for the hostel. |  |  |  | |  |  |
| **Post-Conditions:** The user is logged in to the system. The user has access to the functions of the system | | | | | | | |

**Table 7.1-5 test case- online payment**

| **Test Case ID:005** | | | | | **Test Designed by: ahsan yousaf** | | |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Test Module Name:** Online Payment testing | | | | | **Test Designed Date:** 04/6/2023 | | |
| **Test Priority:** high | | | | | **Test Executed by: ahsan yousaf** | | |
| **Test Title Name:** To test the online payment functionality. | | | | | **Test Executed Date:**04/6/2023 | | |
| **Description:** Customer login to their account. He/she searches for available hostel. Customers will connect with them by boking. After booking, users can pay them through cash or online method. | | | | |  | | |
| **Pre-Conditions:** The user must have the website and the user must have an account. | | | | | | | |
| **Dependencies:** | | | | | | | |
| **Step** | **Test Step** | **Test Data** | **Expected Result** | **Actual Result** | | **Status(Pass/Fail)** | **Notes** |
| **1.** | Log in to your account. |  |  |  | | **Pass** |  |
| **2.** | Enter his/her current location. |  |  |  | |
| **3.** | Search for the hostel . |  |  |  | |
|  | Payment |  | After booking, the user can pay them through cash or online method. | After booking, the user can pay them through cash or online method. | |  |  |
| **Post-Conditions:** The user is logged in to the system. The user has access to the functions of the system | | | | | | | |

**Table 7.1-6 test case - feedback**

| **Test Case ID:006** | | | | | **Test Designed by: ahsan yousaf** | | |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Test Module Name:** Rating/Feedback Testing | | | | | **Test Designed Date:** 04/6/2023 | | |
| **Test Priority:** High | | | | | **Test Executed by: ahsan yousaf** | | |
| **Test Title Name:** To test the rating functionality. | | | | | **Test Executed Date:**04/6/2023 | | |
| **Description:** Customer login to their account. He/she searches for available hostel. Customers will connect with them through booking. the user gives them a rating after using the service. | | | | |  | | |
| **Pre-Conditions:** The user must have the website and the user must have an account. | | | | | | | |
| **Dependencies:** | | | | | | | |
| **Step** | **Test Step** | **Test Data** | **Expected Result** | **Actual Result** | | **Status(Pass/Fail)** | **Notes** |
| **1.** | Log in to your account. |  |  |  | | **Pass** |  |
| **2.** | Enter his/her current location. |  |  |  | |
| **3.** | Search for the hostel . |  |  |  | |
| **4.** | The user gives feedback/rating. |  | After using the services, the user can give feedback/rating. | After using the services, the user can give feedback/rating. | |  |  |
| **Post-Conditions:** The user is logged in to the system. The user has access to the functions of the system. | | | | | | | |

# Decision Table

### 7.2.2 Decision coverage table

**Table 7.1-1 DT-Sign up**

| Test Steps | Condition: Navigate to the Signup Page | Condition: Enter Signup Information. | Condition: Click on the Create button. | Actions |
| --- | --- | --- | --- | --- |
| 1. | True | - | - | - |
| 2. | - | True | - | - |
| 3. | - | - | True | Sign Up |

**Table 7.1-2 DT-Login**

| Test Steps | Condition: Navigate to the Login Page | Condition: Enter Login Information. | Condition: Click on the Login button. | Actions |
| --- | --- | --- | --- | --- |
| 1. | True | - | - | - |
| 2. | - | True | - | - |
| 3. | - | - | True | Login |

**Table 7.1-3 DT-User Home Page**

| Test Steps | Condition: User have to login. | Condition: Navigate to user home page. | Condition: Enter his current location. | Condition: Choose hostel. | Actions |
| --- | --- | --- | --- | --- | --- |
| 1. | True | - | - | - | Login |
| 2. | - | True | - | - | Home page |
| 3. | - | - | True | - | Current location. |
| 4. | - | - | - | True | Hostel |

**Table 7.1-4 DT-Search Function**

| Test Steps | Condition: Navigate to user home page. | Condition: Enter his current location. | Condition: Choose hostel. | Condition: Search for the nearest hostel. | Actions |
| --- | --- | --- | --- | --- | --- |
| 1. | True | - | - | - | Home page |
| 2. | - | True | - | - | Current location. |
| 3. | - | - | True | - | Hostel |
| 4. | - | - | - | True | Nearest Hostel |

**Table 7.1-5 DT- Hostel Profile**

| Test Steps | Condition: Choose hostel. | Condition: Search for the nearest hostel. | Condition: Click on the book icon. | Condition: Click on hostel profile. | Actions |
| --- | --- | --- | --- | --- | --- |
| 1. | True | - | - | - | Hostel |
| 2. | - | True | - | - | Search nearest hostel. |
| 3. | - | - | True | - | Click on book. |
| 4. | - | - | - | True | hostel profile |

# RESULTS/OUTPUT/STATISTICS

* 1. **%completion**

**Table 8.1: Completion**

| **RID** | **Requirement Description** | **UCID** | **Status (Pass/Fail)** | **Completion** |
| --- | --- | --- | --- | --- |
| FR-1 | Signup | UC-1 | Pass | 100% |
| FR-2 | Log in | UC-2 | Pass | 100% |
| FR-2 | Log out | UC-3 | Pass | 100% |
| FR-3 | Create hostel | UC-4 | Pass | 100% |
| FR-5 | Hostel approval | UC-5 | Pass | 100% |
| FR-6 | Account manages | UC-6 | Pass | 100% |
| FR-7 | Create portfolios | UC-7 | Pass | 100% |
| FR-8 | Search | UC-8 | Pass | 100% |
| FR-9 | payments | UC-9 | Pass | 100% |
| FR-10 | Feedback | UC-10 | Pass | 100% |

* 1. **%accuracy**

**Table 8.2: Accuracy**

| **RID** | **Requirement Description** | **TID** | **Status (Pass/Fail)** | **Completion** |
| --- | --- | --- | --- | --- |
| FR-1 | Signup | TID-1 | Pass | 100% |
| FR-2 | Log in | TID-2 | Pass | 100% |
| FR-3 | Current Location | TID-3 | Pass | 100% |
| FR-4 | Nearest Hostel | TID-4 | Pass | 100% |
| FR-5 | Booking | TID-5 | Pass | 100% |
| FR-6 | Pricing | TID-6 | Pass | 100% |
| FR-7 | Payment | TID-7 | Pass | 100% |
| FR-8 | Feedback | TID-8 | Pass | 100% |

* 1. **%correctness**

**Table 8.3: Correctness**

| **RID** | **Requirement Description** | **TID** | **Status (Pass/Fail)** | **Test Result** |
| --- | --- | --- | --- | --- |
| UC-1 | Signup | TID-1 | Pass | Successful |
| UC-2 | Log in | TID-2 | Pass | Successful |
| UC-3 | Current Location | TID-3 | Pass | Successful |
| UC-4 | Nearest Hostel | TID-4 | Pass | Successful |
| UC-5 | Booking | TID-5 | Pass | Successful |
| UC-6 | Pricing | TID-6 | Pass | Successful |
| UC-7 | Payment | TID-7 | Pass | Successful |
| UC-8 | Feedback | TID-8 | Pass | Successful |

# Conclusion

The Hostel Booking website is a valuable tool that simplifies the process of finding and reserving accommodations, catering specifically to budget-conscious travelers. With its user-friendly interface, advanced search features, and real-time availability updates, the website enhances the overall user experience.

By utilizing location-based services and offering detailed hostel profiles, the website empowers customers to make informed decisions based on ratings, reviews, and various booking options. The seamless coordination of bookings is ensured through integrated communication features, allowing travelers to interact directly with hostel owners or staff.

Looking ahead, the website's future endeavors will focus on enhancing the UI/UX, expanding search capabilities, implementing real-time booking status tracking, and integrating secure payment gateways. Robust security measures, including authentication protocols, data encryption, and regular audits, are in place to safeguard user information, establishing trust and confidence among users.

The Hostel Booking website not only benefits travelers by providing convenient access to affordable accommodations but also supports hostel owners by connecting them with a wider customer base. Its potential for growth lies in continuous improvement, incorporating user feedback, and expanding its services to cater to a broader range of destinations.

# 10. Future Work

The future work for the Hostel Booking website includes enhancing the user experience, expanding accommodation options, improving filtering and search capabilities, implementing real-time booking updates, integrating with local attractions and services, ensuring security and privacy measures, and expanding to additional regions. The website aims to provide a seamless and personalized booking experience for travelers while supporting hostel owners in reaching a broader customer base.

