**Object Oriented Software Engineering – OOSE**  
**SUBJECT CODE: SE 202  
LAB FILE**

Submitted by –

Dheeraj Tokas (2K21/SE/61)

&  
Dilwinder Singh (2K21/SE/67)  
Batch – SE A1 G2 (2nd year)

Under the supervision of  
Ms. Anjali Bansal



**DEPARTMENT OF SOFTWARE ENGINEERING**

DELHI TECHNOLOGICAL UNIVERSITY   
(FORMERLY DELHI COLLEGE OF ENGINEERING)  
MAIN BAWANA ROAD, Delhi 110042

**INDEX**

| **S.NO** | **TITLE** | **DATE** | **Signature** |
| --- | --- | --- | --- |
| 1. | To write problem statement for educational website |  |  |
| 2. | To draw the initial requirement document for the diagram |  |  |
| 3. | To write the Software Requirement Specification (SRS) document for the educational website |  |  |
| 4. | To generate test case matrix |  |  |

**Experiment - 01**

**HOTEL WEBSITE**

**PROBLEM STATEMENT**

This is a website that will be developed to attract the customers to the hotel and manage everything efficiently. This includes the images of the rooms, lobby, swimming pool, parking. garden and other amenities. In the current system, the customer has to visit the hotel desk to enquire about the rooms and services and booking. A lot of paper work needed which was not environment friendly. This system takes lot of time and wastage of resources. So, the limitations of the current system are as follows:

* Difficulty in maintaining record of registered users.
* Difficulty in updating status of rooms on daily basis.
* Taking on-time payment from those who want to book a hotel.
* Losing track of maintenance requests.
* No clear profit or loss margins for the Hotel Owner.
* No records of inspection report.
* Management of Room, Staff and Inventory of the Hotel.
* Difficulty in generating annual income reports and monthly profits report.

So, to overcome the above limitations we are proposing a website that will deal with above limitations and will add on some new features for the customer case, that consists of following features:

* Online Database of all the registered users.
* Daily basis update of the hotel rooms on the system.
* Generation of automated bills and payment receipts online.
* Online Payment of bill through the website by the customer.
* Easily view and update all available and booked rooms.
* View and book the facilities provided by the hotel like swimming pool, cafeteria, food court, gym etc.
* Taking feedback and complaints from user online.
* Generate annual income reports and monthly profits.

**Experiment-02**

**INITIAL REQUIREMENT DOCUMENT**

| **Title of the project** | **Hotel Website** | |
| --- | --- | --- |
| Stakeholders involved in capturing requirements | Hotel Employees, Guests | |
| Techniques used for requirement capturing | Brainstorming and Interviewing | |
| Name of the persons along with designation | Dheeraj Tokas (2K21/SE/61)  Dilwinder Singh (2K21/SE67) | |
| Date | 2 February, 2023 | |
| Version | 1.0 | |
| **Consolidated list of initial requirements:**   1. A system to be implemented which can run on the LAN and WAN network. 2. There will be three type of users Admin, Customer, Staff 3. System will have the option for user registration. 4. System will have a login window, which will ask for the user Login ID, Password, Forgot Password option. 5. System will show the available rooms on a particular date and the user will be able to search it. 6. Users will be able to book rooms in the hotel by entering check in and check out dates. 7. System will have a payment gateway which will provide various payment options to users. 8. System will generate an e-receipt for the user upon successful payment. 9. Guests can avail hotel facilities like ordering of food, laundry and request for room cleaner. 10. System will allow the admin to manage rooms, staff and inventory like addition, deletion and modification of rooms, rate list, cafeteria, user profiles. 11. System shall have the option to modify the check-in and check-out dates of the booked rooms as per availability. 12. Guests can also change the type of service he wants and also the number of rooms as per availability. 13. System shall generate the statistical reports of hotel rooms and services used. Report shall be available in various formats. 14. System shall provide an option to the user to give his feedback after successful stay in the hotel. 15. System shall have an option to register any user complaints. | |

**Experiment - 03**

**Software Requirement Specification**

**Version 1.0.1**

**April 26th, 2023**

**DILWINDER SINGH (2K21/SE/67)**

**DHEERAJ TOKAS (2K21/SE/61)**

**B.Tech., Software Engineering**

**4th Semester**

**Index**

| 1. | Introduction  1.1 Purpose  1.2 Scope  1.3 Definitions, Acronyms and Abbreviations  1.4 References  1.5 Overview |
| --- | --- |
| 2. | Overall Description  2.1 Product Perspective  2.1.1 System Interfaces  2.1.2 User Interfaces  2.1.3 Hardware Interfaces  2.1.4 Software Interfaces  2.1.5 Communications Interfaces  2.1.6 Memory Constraints  2.1.7 Operations  2.1.8 Site Adaptation Requirements  2.2 Product Functions  2.3 User Characteristics  2.4 Constraints  2.5 Assumptions and Dependencies  2.6 Apportioning of Requirements |
| 3. | Specific Requirements  3.1 External Interface  3.2 Functions  3.3 Performance Requirements  3.4 Logical Database Requirements  3.5 Design Constraints  3.6 Software System Attributes  3.6.1 Reliability  3.6.2 Availability  3.6.3 Security  3.6.4 Maintainability  3.6.5 Portability  3.7 Additional Comments |
| 4. | Supporting Information |

**Revision History**

| **Name** | **Date** | **Version** |
| --- | --- | --- |
| Hotel Website | 22th April 2023 | 1.0 |
|  |  |  |

# Introduction

Hotels nowadays are focusing on maximizing their revenues as all other businesses do and the main reason for that is the ever-increasing competition. The online world has made it difficult for hotels and resorts to compete by providing guests a plenty of choices including Homestay, Hostel with bunk beds or even a modular option. Therefore, there is an increasing demand for the best hotel website system in order to facilitate the management of hotel operations and functions. Operating a successful hotel business today is a challenge in itself.

A hotelier has to manage various proposals such as operations, staff and maintenance, meanwhile keeping costs under control and balances as it is one of the most important and critical issues for a hotel business to increase their revenues and to compete with other hotels. To improve the efficiency of this process, a good hotel website system which uses the modernizing techniques must be provided.

## Purpose

The key to reaping the benefits of an effective hotel website software system is to select the right one for your property. It is critical that you know exactly what this hotel website technology is, and why it is important for you to implement it at your hotel. These days every person can find different options of the hotel reservation software free on the internet, however one has to judge the solution with the quantum of features and quality that it is providing. There should be a complete functionality as a hotel website can be both basic and advanced based on the pricing options that are available as well.

The Hotel website is a key element for a very important branch of the economy, which is tourism.

Our software aims to have all the features that a hotel needs in order to adapt to the management structure of the business, and by making practical and effective use of these features every hotelier's work life will be much easier than they have ever imagined by using this product.

## Scope

The name of the software is Hotel Website System (HWS). The proposed HWS must be able to perform the following functions:

1. Issue Login id and password to system operators.
2. Maintain details of rooms available in the hotel.
3. Maintain details of the hotels and employees to provide employee membership.
4. Check the availability of rooms
5. View a room
6. Book a room
7. Make transaction
8. Booking cancellation
9. Generate the following reports:

(a) List of rooms booked by a customer.

(b) Details of rooms booked and returned on daily basis.

(c) Total available rooms in the hotel.

(d) List of customers along with booked rooms.

(e) List of available rooms in the hotel.

## Definition, Acronym, Abbreviation

This should define all technical terms and abbreviations used in the document

* CPU – Central Processing Unit
* CSS – Cascading stylesheet
* DFD – Data Flow Diagram
* HMS – Hotel management system
* HTML – Hypertext Markup Language
* IEEE – Institute of Electrical and Electronics Engineers
* PC – Personal Computer
* RAM – Random Access Memory
* SRS – Software Requirement Specification

## References

* Yogesh Singh, Ruchika Malhotra. (2012). Object Oriented Software Engineering.1st Ed. PHI Learning
* IEEE Recommended Practice for Software Requirements Specifications- IEEE  
  Std 830-1998.
* IEEE Standard for Software Test Documentation—IEEE Std. 829-1998.

## Overview

* The SRS contains an analysis of the requirements necessary to help easily design the system.
* The overall description provides interface requirements for the Hotel Website, product perspective, hardware interfaces, software interfaces, communication interface, memory constraints, product functions, user characteristics, and other constraints.
* Succeeding pages illustrate the characteristics of typical naive users accessing the system along with legal and functional constraints enforced that affect the Hotel Website in any fashion.

# Overall Description

This document contains the problem statement that the current system is facing which is hampering the growth opportunities of the company. It further contains a list of the stakeholders and users of the proposed solution. It also illustrates the needs and wants of the stakeholders that were identified in the brainstorming exercise as part of the requirements workshop. It further lists and briefly describes the major features and a brief description of each of the proposed systems.

## Product Perspective

The software system is a web-based application developed using PHP and My SQL  
that will be accessed through a web browser. The system will be designed to work on  
multiple platforms such as Android , MacOS, IOS, Linux and will be scalable to  
accommodate a large number of users.

### System interfaces

The software system will interact with other systems through APIs to  
exchange data.

### User interfaces

The user interface of the system will be a web-based interface that is easy to  
navigate and use which will be as follows:

1. Login: to allow entry of only authorized users through valid ID and password
2. Registration: to maintain details of the guest and staff by administration for maintaining their records.
3. Booking Room: to maintain booking room for the guest during a specified time period.
4. User Feedback: to maintain the guest has experienced the services and room, they give feedback for improvement of hotel management.
5. Payment: After booking room selecting payment method and doing payment for successful booking.

### Hardware interfaces

The software system will be compatible with a variety of hardware  
configurations.

* Screen resolution of at least 640x480 or above.
* Support for printers (dot matrix, Deskjet, laserjet).
* Computer systems will be in the networked environment as it is a  
  multi-user system

### Software interfaces

* MS-Windows Operating System (NT/XP/Vista).
* Microsoft Visual Basic 6.0 for designing front end.
* MS SQL Server 2005 for back-end.

### Communications interfaces

The Hotel website will use LAN, web-enabled services, or stand-alone systems. In the restaurant premise communication is via local area network (LAN) and web-enabled services.

### Memory constraints

At least 512 MB RAM and 500 MB space of hard disk will be required to run  
the software.

## Product Function

* The website should be able to provide a clear and easy to navigate menu including prices and photos of rooms and food.
* The website should be able to provide a physical address of the Hotel, phone number, and email address.
* The website should have social media handles of the hotel.
* The system should be able to provide information about the hotel history, mission, and values.
* The website must have an option to purchase and redeem gift cards and loyalty discounts should be provided to regular customers.
* The website should be optimised for viewing on mobile devices.
* The website should be able to accept hotel’s catering services.
* The website should be able to provide customers and staff members login credentials.
* The website should provide a Gallery section featuring photos of the hotel’s interior, exterior and dishes to give customers a sense of what to expect.

## User Characteristics

* Qualification: At least matriculation and comfortable with English.
* Experience: Should be well versed/informed about the processes of booking tickets  
  on the online portal.
* Technical experience: Elementary knowledge of computers.

## Assumptions and Dependencies

**Assumptions**

We assume:

1. All users have basic knowledge in English language.

2. All users have basic knowledge in computer and smartphone usage.

3. Stakeholders of the hotel have basic knowledge on how to use the system due to previous experiences with other systems.

4. Hotel is equipped with PC/Laptop/Tablet, printer, mobile phone.

5. Hotel must have internet connection all the time.

6. It is assumed that the Hotel provides Credit Card payments.

## Dependencies

As it is a web-based application it is always dependent on internet connection. A connection is required to send commands and receive answers, usually in the form of a result set.

* 1. **Constraints**

1. All users must be logged in order to use the product and to access the information.

2. Phone memory of the users is another problem, prices and quality of android phones varies and so the memory, most of the workers are likely to have cheaper phones and that can be a problem for the application usage.

**2.6** **Apportioning of Requirements**

Not Required

**3.** **Specific Requirements**

This section contains the software requirements to a level of detail sufficient to enable designers to design the system, and testers to test that system.

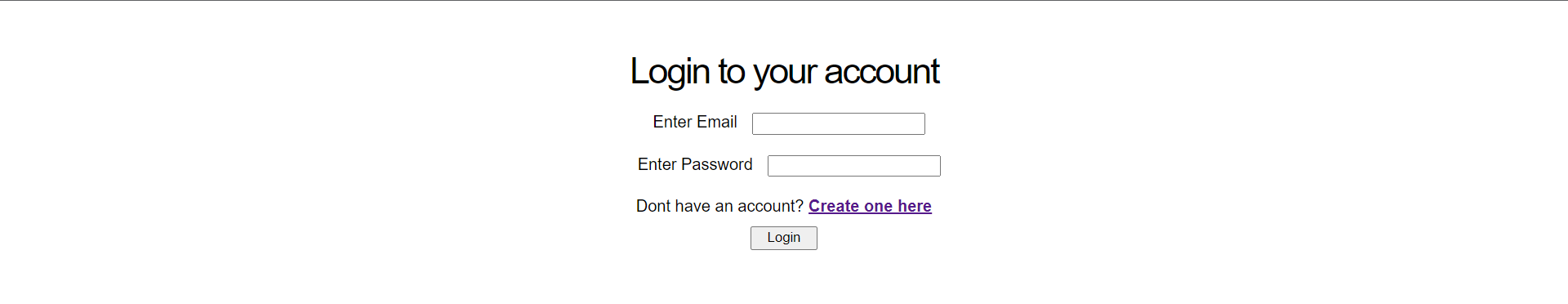
**3.1 External Interface**

It defines how the hotel website will interact with external systems and services, providing a clear and comprehensive overview of the system's integration requirements.

**3.1.1** **User Interface**

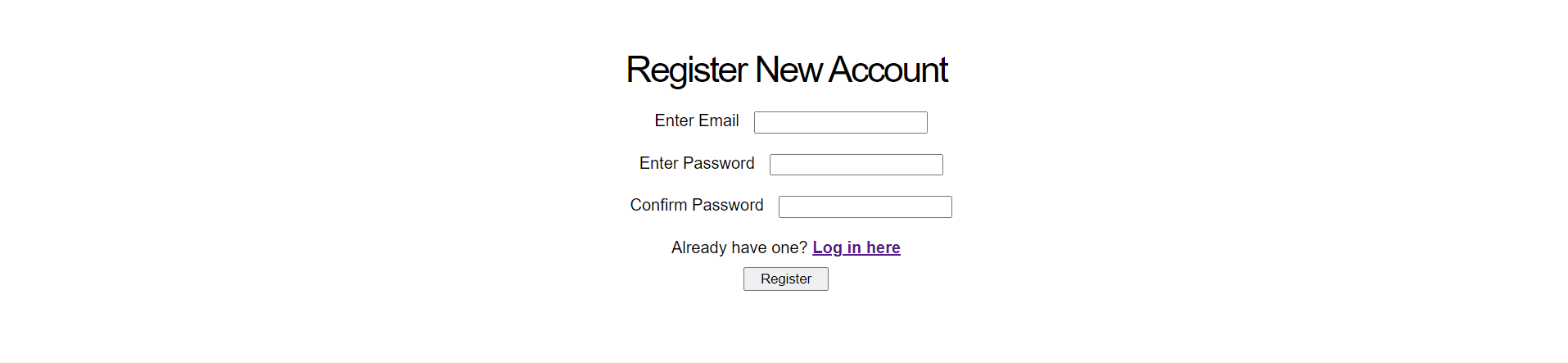
(a) **Login Form:** This will be the first form, which will be displayed. It will allow the user to access the different screens based on their role.

Various fields available on this form will be:

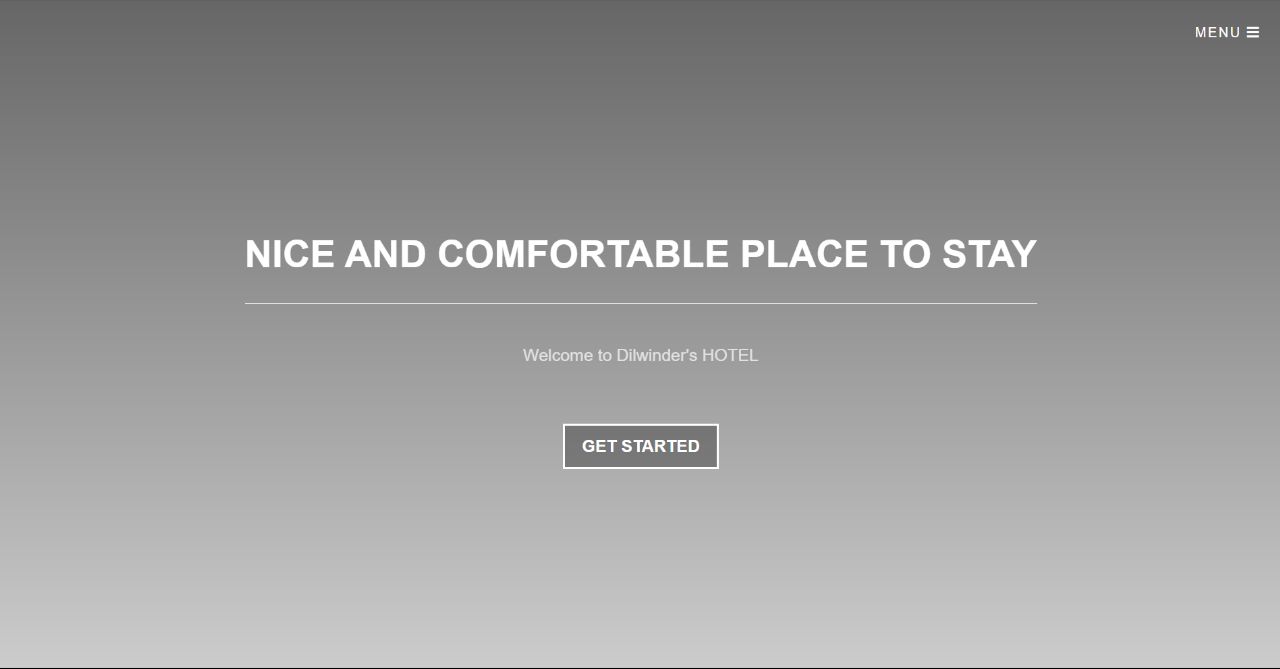
* Login ID: Alphanumeric of length in the range of 4 to 15 characters. Special characters and blank spaces are not allowed.
* Password: Alphanumeric of length in the range of 4 to 15 characters. Blank spaces are not allowed. However, special characters are allowed. 

(b) **Register Form:** It will allow the user to register on the website.

* Login ID: Alphanumeric of length in the range of 4 to 15 characters. Special characters and blank spaces are not allowed.
* Password: Alphanumeric of length in the range of 4 to 15 characters. Blank spaces are not allowed. However, special characters are allowed.

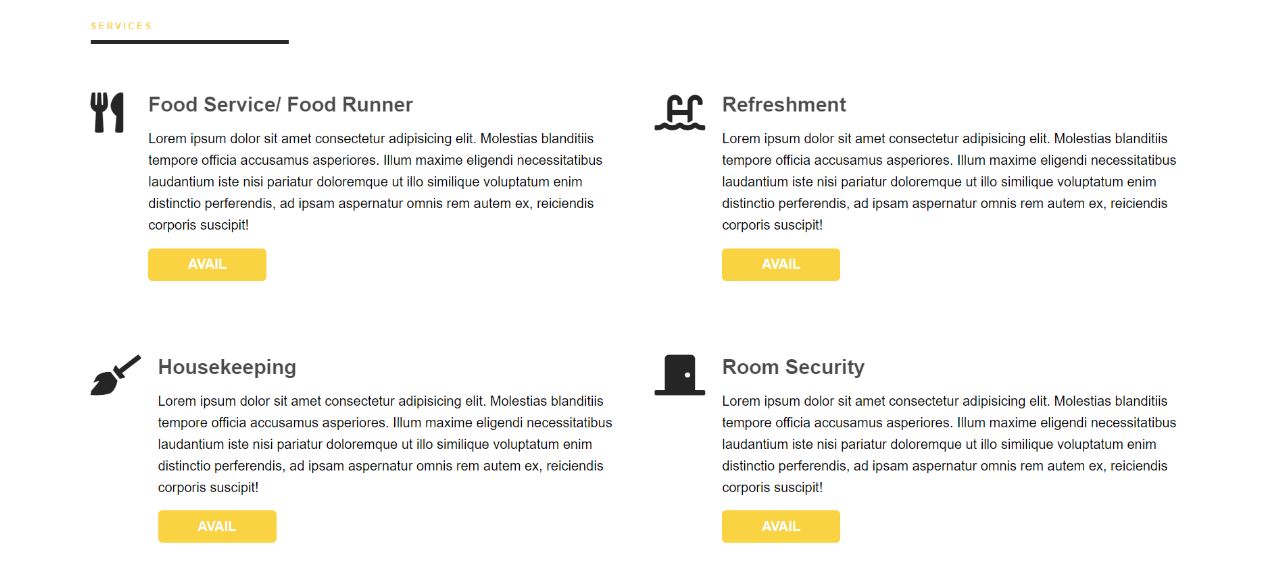


(c) **Home Page:**

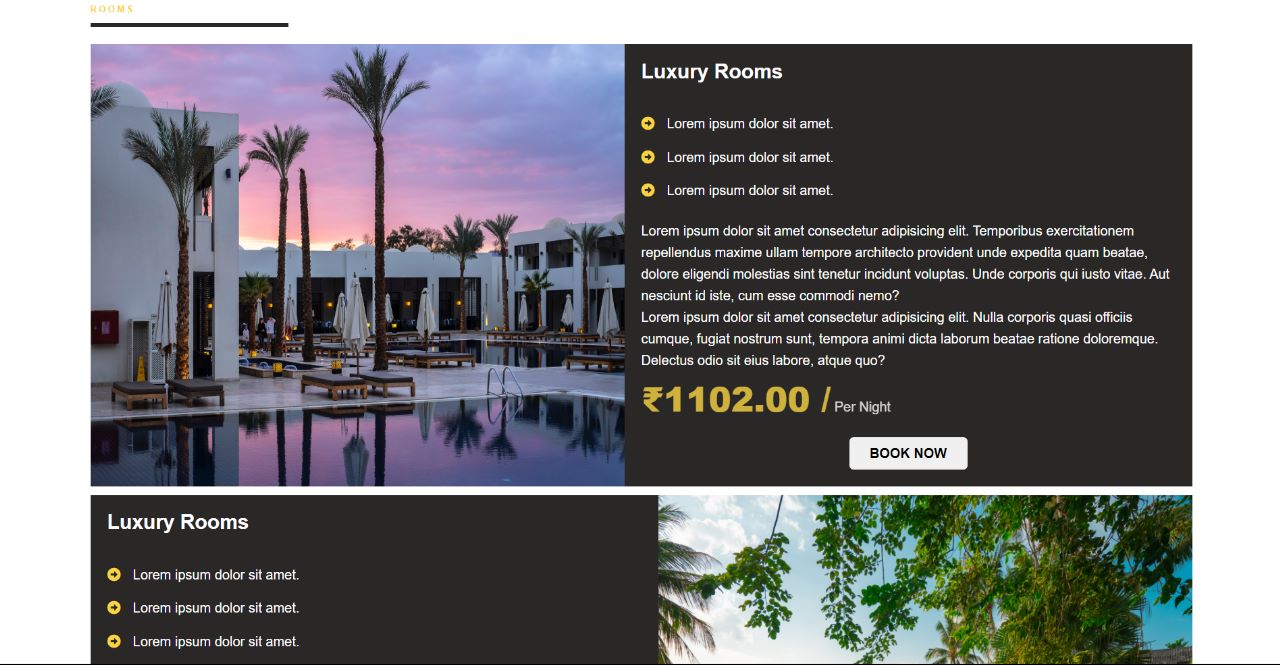
****

‘

(d) **Avail Hotel Facility:**



(e) **Room Book:**

****

****

**3.1.2** **Hardware Interface**

The software system will be compatible with a variety of hardware  
configurations.

* Screen resolution of at least 640x480 or above.
* Support for printers (dot matrix, Deskjet, laserjet).
* Computer systems will be in the networked environment as it is a  
  multi-user system

**3.1.3** **Software Interface**

* MS-Windows Operating System (NT/XP/Vista).
* Microsoft Visual Basic 6.0 for designing front end.
* MS SQL Server 2005 for back-end.

**3.1.4** **Communication Interface**

The Hotel website will use LAN, web-enabled services, or stand-alone systems. In the restaurant premise communication is via local area network (LAN) and web-enabled services.

**3.2** **Functions**

**1. Sign Up**

| **1.** | **Introduction**  Allows the user to sign up to hotel website system |
| --- | --- |
| **2.** | **Actor**  **·** Guest  · Admin  · Hotel Staff |
| **3.** | **Precondition**   * There is space in database for new user to be added. |
| **4.** | **Postcondition**   * The user has created their account and their details are stored in database. |
| **5.** | **Basic Flow**   * The user navigates to the hotel website and clicks on the "Sign Up" button. * The system presents the user with a sign up form, where they enter their required details. * The system verifies the user's phone number and grants permission to create their account. |
| **6.** | **Alternative Flow**   * If the phone number of user isn’t verified then the system displays an error message and prompts the user to try again. |
| **7.** | **Special Requirements**  None |
| **8.** | **Associated Use Cases**  None |

**2. Login**

| **1.** | **Introduction**  Allows the user to login to hotel website system |
| --- | --- |
| **2.** | **Actor**  · Guest  · Admin  · Hotel Staff |
| **3.** | **Precondition**   * The user is already registered with the hotel website. * The user has a valid username and password. |
| **4.** | **Postcondition**  · The user is logged in to their account and has access to their reservation information and account details. |
| **5.** | **Basic Flow**   * The user navigates to the hotel website and clicks on the "Login" button. * The system presents the user with a login form, where they enter their username and password. * The system verifies the user's credentials and grants access to their account. * The system displays the user's account dashboard, which includes their reservation information, account details, and any other relevant information. * The user can make changes to their account details, view their reservation information, or take any other actions available to registered users. |
| **6.** | **Alternative Flow**  · If the user enters an incorrect username or password, the system displays an error message and prompts the user to try again. |
| **7.** | **Special Requirements**  None |
| **8.** | **Associated Use Cases**  None |

**3. Maintain Booking**

| **1.** | **Introduction**  This use case allows the guest to make a reservation and view the information for a room. |
| --- | --- |
| **2.** | **Actor**  · Guest |
| **3.** | **Precondition**  · The user has logged in to the website.  · The user has checked the availability of rooms and selected a room for specific dates. |
| **4.** | **Postcondition**  The user has successfully made a reservation for specific dates and has received a confirmation number and receipt.  The existing user exits the website. |
| **5.** | **Basic Flow**   * The user selects the room they want to book and clicks on the "Book Now" or "Reserve" button. * The system displays a form where the user can confirm their reservation details, such as the check-in and check-out dates, room type, and price. * The user confirms their reservation details and clicks on the "Continue" or "Book Now" button. * The system prompts the user to enter their personal information, such as their name, email address, and phone number. * The user enters their personal information and clicks on the "Continue" or "Next" button. * If payment is required, the system prompts the user to enter their payment details, such as their credit card number, expiration date, and CVV code. * The user enters their payment details and clicks on the "Continue" or "Next" button. * The system confirms the reservation details and provides the user with a confirmation number and receipt. * The user can print or save the confirmation details and receipt for future reference. |
| **6.** | **Alternative Flow**  If the user encounters an error or the system is not functioning properly, the system displays an error message and prompts the user to try again later. |
| **7.** | **Special Requirements**   * Cancellation Policy: To avoid disputes and manage user expectations, the system should clearly state the cancellation policy and any fees associated with cancellations or changes to reservations. |
| **8.** | **Associated Use Cases**  Payment |

**4. Payment**

| **1.** | **Introduction**  This use case allows a user to pay for the room and hotel facilities |
| --- | --- |
| **2.** | **Actor**  Guest |
| **3.** | **Precondition**  **●**  The user has made a booking on the hotel website and received a confirmation number.  ● The user has access to a valid payment method, such as a credit card or UPI  ● The payment system is accessible and functioning properly. |
| **4.** | **Postcondition**  The user has successfully paid for their reservation and received a confirmation number and receipt. |
| **5.** | **Basic Flow**   * The user clicks on the "Pay Now" or "Complete Payment" button on the reservation confirmation page. * The system redirects the user to the payment page where they can enter their payment information. * The user enters their payment information, such as their credit card number, expiration date, and CVV code. * The user reviews the payment details and confirms the transaction by clicking on the "Submit" or "Pay Now" button. * The payment system processes the transaction and sends a confirmation to the user and the hotel. * The system displays a confirmation page with the transaction details and provides the user with a receipt and confirmation number. |
| **6.** | **Alternative Flow**  If the user encounters an error or the payment system is not functioning properly, the system displays an error message and prompts the user to try again later. |
| **7.** | **Special Requirements**   * Payment Options: The system should support multiple payment methods, such as credit cards, PayPal, or Apple Pay, to accommodate different user preferences. * Payment Gateway Integration: The system should integrate with a payment gateway to ensure secure and reliable transactions. * Refund Policy: The system should clearly state the refund policy and any fees associated with cancellations or changes to reservations. * Currency Conversion: The system should support currency conversion for international payments and display the total amount in the user's local currency. |
| **8.** | **Associated Use Cases**  None |

**5. Avail Hotel Facility**

| **1.** | **Introduction**  This use case allows a user to avail hotel facilities |
| --- | --- |
| **2.** | **Actor**  **·**  Guest  · Hotel Staff |
| **3.** | **Precondition**  ● The user is logged on hotel website  ● User is hotel staff or has booked a room at hotel |
| **4.** | **Postcondition**  The user has availed |
| **5.** | **Basic Flow**  **Basic Flow 1:**   * The user navigates to the "Hotel facilities" page on the hotel website. * The system displays list of hotel facilities avilable * The user selects the desired facility and avails it. * System confirns that the hotel facility is availed   **Basic Flow 2:**   * Hotel staff navigates to the "Hotel Facilities" page on website * The system displays list of hotel facilities availed by users * Hotel staff can view and proved the hotel facilities to guests. |
| **6.** | **Alternative Flow**  If the hotel facility is not available the system displays a message and takes user back to hotel facilities page. |
| **7.** | **Special Requirements**  None |
| **8.** | **Associated Use Cases**  Payment |

**6. Maintain Room Details**

| **1.** | **Introduction**  This use case allows the guest to check the availability of rooms and view the options available and allows hotel staff and admin to edit the details. |
| --- | --- |
| **2.** | **Actor**  · Guest  · Admin  · Hotel Staff |
| **3.** | **Precondition**  The user is on the hotel website and wants to check the availability of rooms for specific dates. |
| **4.** | **Postcondition**  **·** The updated room details are now available on the hotel website.  · Guest who visit the hotel website can view the updated room details.  · The existing user exits the website or continue managing other rooms. |
| **5.** | **Basic Flow**    **Basic Flow 1:**  **·**  The user navigates to the hotel website and clicks on the "Room Details" button.   * The system displays a form where the user can select the check-in and check-out dates, as well as the number of guests and rooms required. * The user enters the required information and submits the form. * The system queries the reservation system to check the availability of rooms for the specified dates. * If rooms are available, the system displays a list of available rooms with their rates and allows the user to select a room and proceed to the booking process. * If no rooms are available, the system displays a message indicating that no rooms are available for the specified dates and prompts the user to select different dates or room types.   **Basic Flow 2:**  · Hotel staff and Admin navigates to the hotel website and clicks on the "Room Details" button.  · The hotel staff member navigates to the hotel website and logs in using their username and password.  · The staff member selects the "Manage Rooms" option from the website's dashboard.  · The website displays a list of available rooms.  · The staff member selects the room they want to update and clicks the "Edit" button.  · The website displays the current room details in an editable format.  · The staff member updates the desired details, such as the room type, amenities, price, and availability.  · The staff member clicks the "Save Changes" button.  · The website updates the room details and displays a success message to the staff member. |
| **6.** | **Alternative Flow**  If the user encounters an error or the system is not functioning properly, the system displays an error message and prompts the user to try again later. |
| **7.** | **Special Requirements**  None |
| **8.** | **Associated Use Cases**  None |

**7. Manage User Details**

| **1.** | **Introduction**  This use case allows users to change their personal information on the website. |
| --- | --- |
| **2.** | **Actor**  **·** Guest  · Admin  · Hotel Staff |
| **3.** | **Precondition**  The user is logged into website. |
| **4.** | **Postcondition**  The details of user has been updated. |
| **5.** | **Basic Flow**  **·** The user logs in to their account on the hotel website.  · The system displays the user's account dashboard or profile page.  · The user selects the "manage user details" option.  · The system displays the user's current account information, including their name, email address, password, and any other relevant details.  · The user can edit or update their account information as needed, such as changing their email address, updating their billing information, or resetting their password.  · After making any changes, the user submits the updated information to the system for processing.  · The system updates the user's account information and displays a confirmation message to the user. |
| **6.** | **Alternative Flow**  **·** If the user submits invalid or incomplete information, such as an invalid email address or password, the system will display an error message and prompt the user to correct the relevant fields before resubmitting the form.  · If the user decides to cancel any changes they have made, they can choose to revert back to their previous settings or cancel the update altogether. |
| **7.** | **Special Requirements**  **·** User authentication: The system must authenticate the user before allowing them to access their account details or make changes to their account information. This includes verifying the user's login credentials, such as their email address and password, and may include additional security measures such as two-factor authentication or biometric authentication.  · Scalability: The system should be designed to handle a large volume of user account management requests and accommodate future growth and expansion of the hotel website. |
| **8.** | **Associated Use Cases**  None |

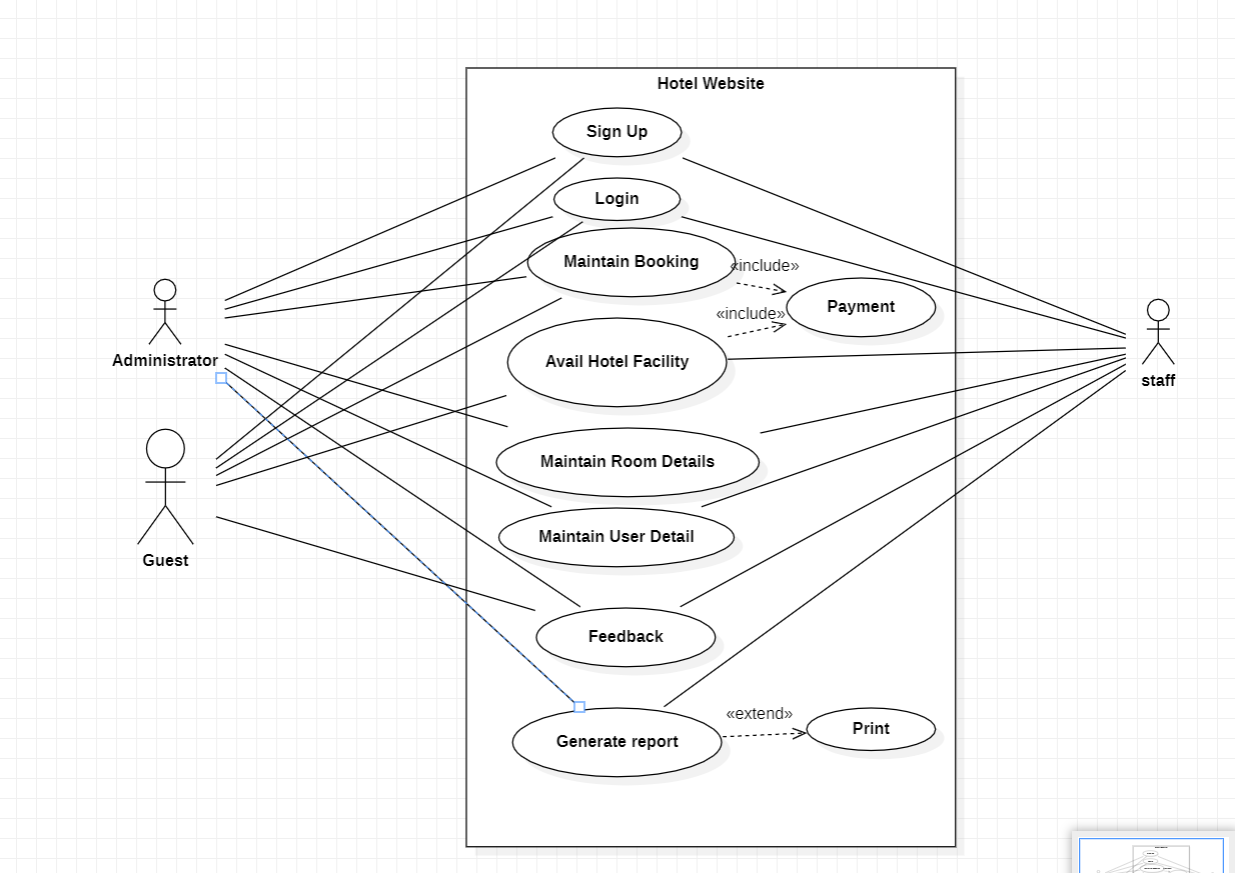
**8. Feedback**

| **1.** | **Introduction**  This use case allows a user to give feedback. |
| --- | --- |
| **2.** | **Actor**  · Guest  · Admin  · Hotel Staff |
| **3.** | **Precondition**   * The user is logged into the website. * The user has booked a room. |
| **4.** | **Postcondition**  The user has successfully given feedback to hotel staff and admin. |
| **5.** | **Basic Flow**  **·** The guest gives his precious feedback to hotel staff.  · The admin and hotel staff will absorb their feedback. |
| **6.** | **Alternative Flow**  If the user encounters any issues while filling in the contact form, such as missing or invalid information, the system displays an error message and prompts the user to correct their input. |
| **7.** | **Special Requirements**   * Message Encryption: The system should encrypt the user's message to ensure the privacy and security of their personal information. * Response Time: The hotel staff should respond to the user's message in a timely manner, preferably within 24 hours of receiving it. * Contact History: The system should maintain a record of the user's past messages and conversations with the hotel staff, allowing them to easily view and manage their communication history. |
| **8.** | **Associated Use Cases**  None |

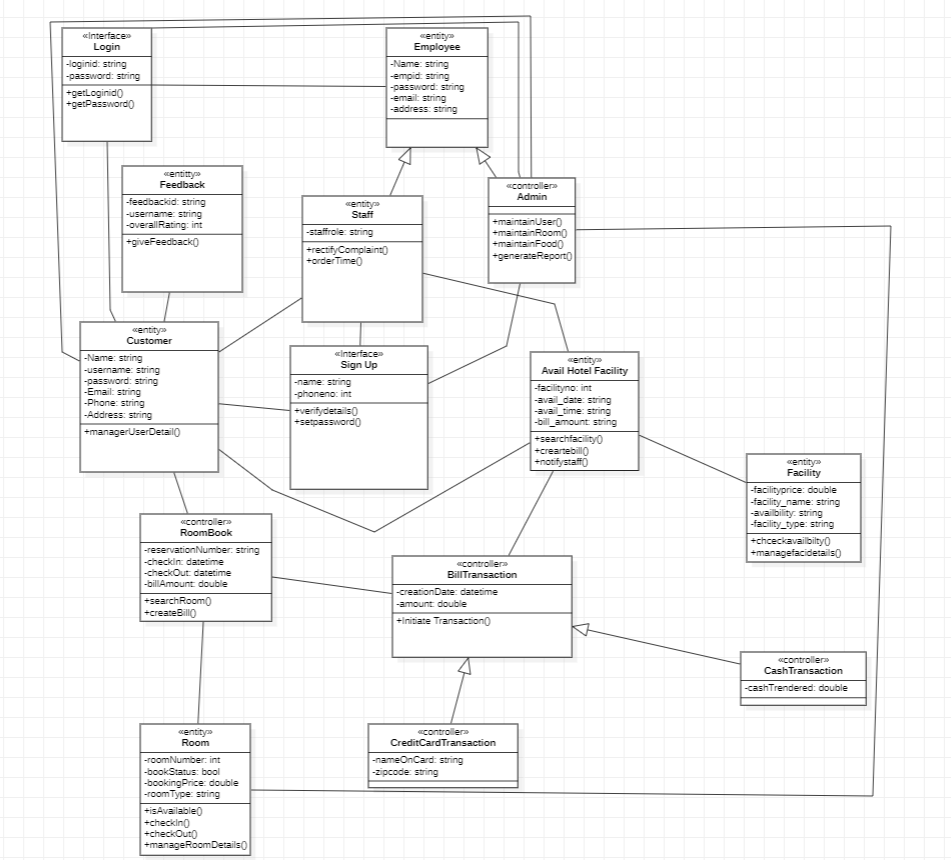
**9. Generate Report**

| **1.** | **Introduction**  This use case allows users to generate reports. |
| --- | --- |
| **2.** | **Actor**  **·** Guest  · Admin  · Hotel Staff |
| **3.** | **Precondition**  The user is logged into the system. |
| **4.** | **Postcondition**  The report is generated. |
| **5.** | **Basic Flow**   * The user logs in to their account on the hotel website. * The user selects the "manage user details" option. * The system displays the user's current account information, including their name, email address, password, and any other relevant details. * The user can edit or update their account information as needed, such as changing their email address, updating their billing information, or resetting their password. * After making any changes, the user submits the updated information to the system for processing. * The system updates the user's account information and displays a confirmation message to the user. |
| **6.** | **Alternative Flow**  **·**  If the user submits invalid or incomplete information, such as an invalid email address or password, the system will display an error message and prompt the user to correct the relevant fields before resubmitting the form.  · If the user decides to cancel any changes they have made, they can choose to revert back to their previous settings or cancel the update altogether. |
| **7.** | **Special Requirements**   * User authentication: The system must authenticate the user before allowing them to access their account details or make changes to their account information. This includes verifying the user's login credentials, such as their email address and password, and may include additional security measures such as two-factor authentication or biometric authentication. * Data validation: The system must validate all user input data to ensure that it meets the required format and content, and prevent any potential data errors or security vulnerabilities. This includes verifying that the user's email address is valid, that their password meets any relevant complexity requirements, and that any updated billing or payment information is accurate.   · Scalability: The system should be designed to handle a large volume of user account management requests and accommodate future growth and expansion of the hotel website. |
| **8.** | **Associated Use Cases**  None |

**Use Case Diagram**

****

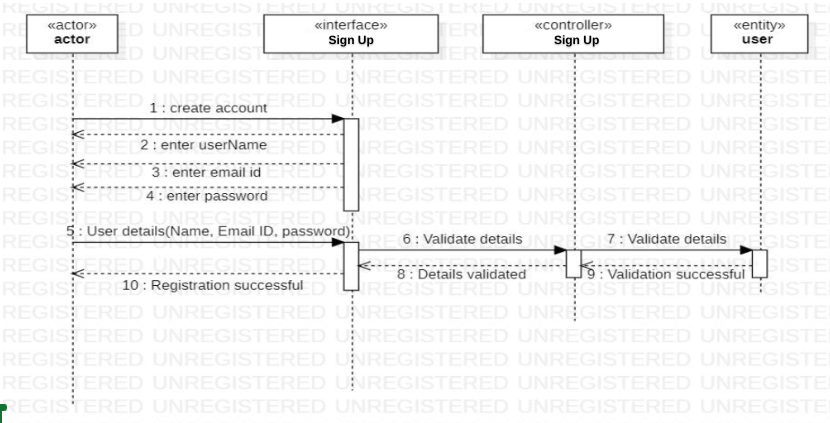
**Class Diagram**

****

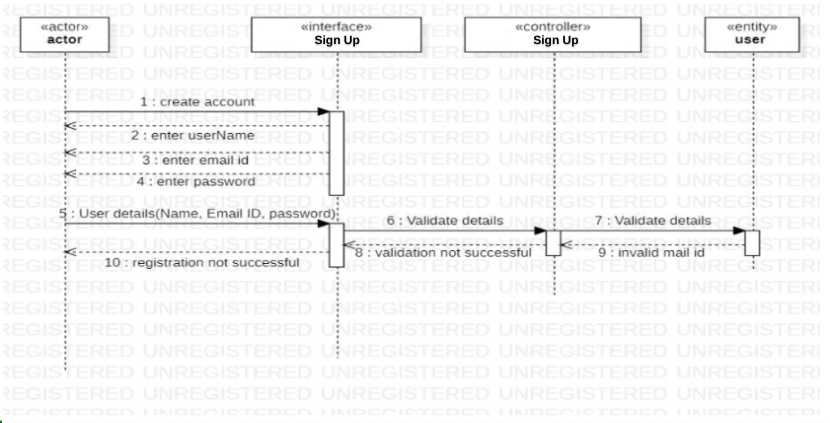
**Sequence Diagram**

**Sign Up:-**

**Basic Flow:**

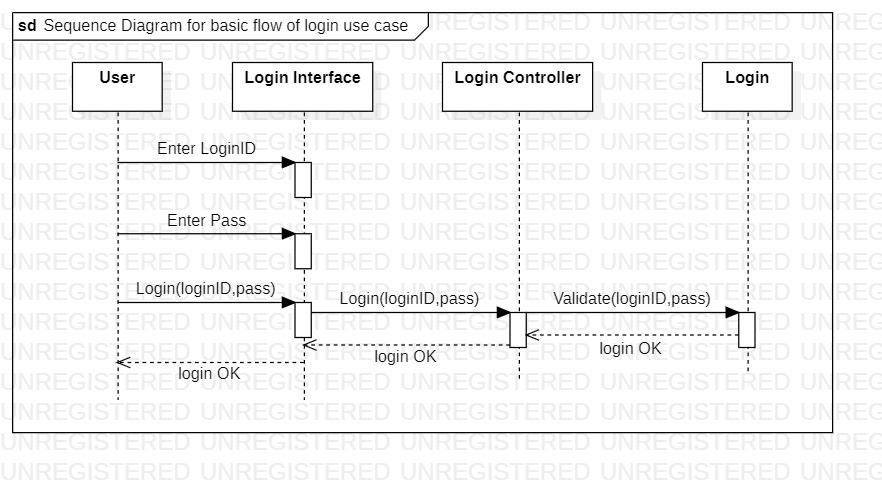


**Alternate Flow:**

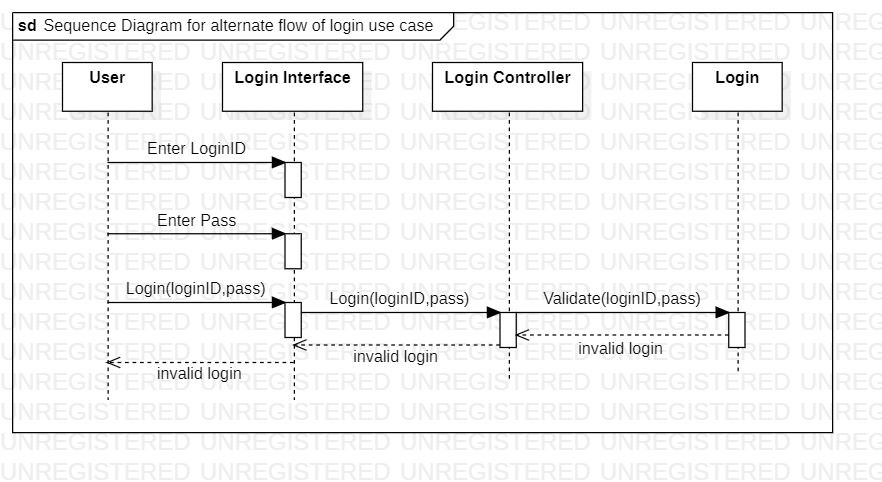


**Login:-**

**Basic Flow:**

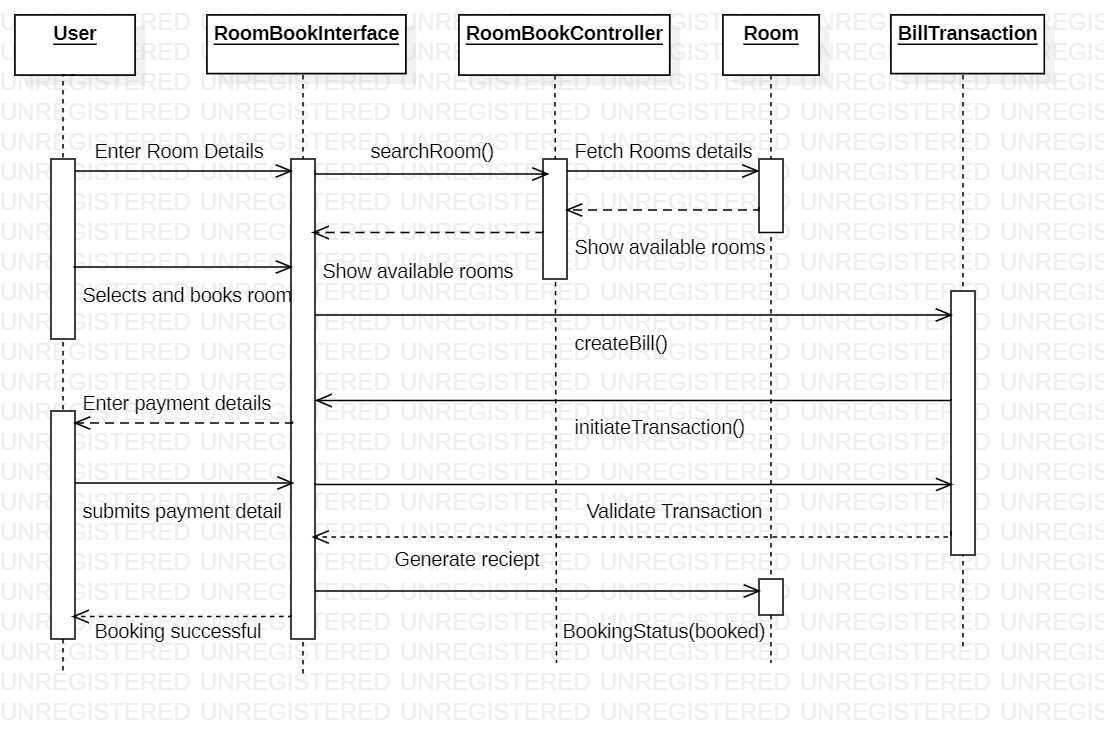


**Alternate Flow:**

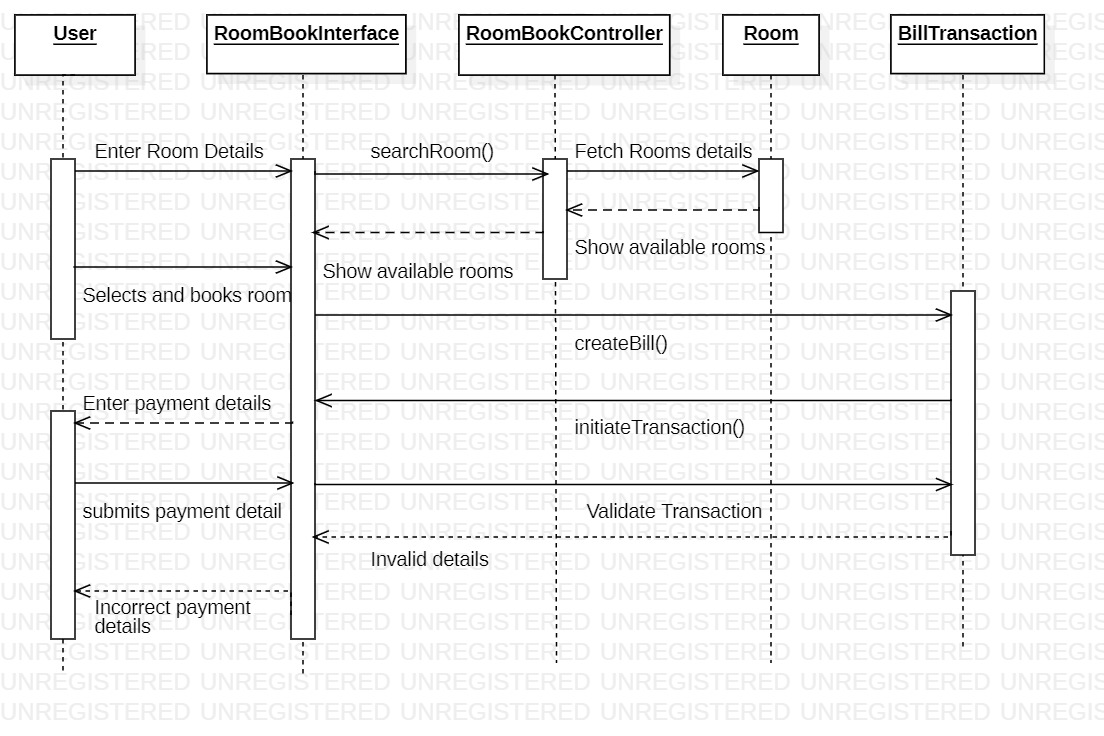


**Room Booking:-**

**Basic Flow:**

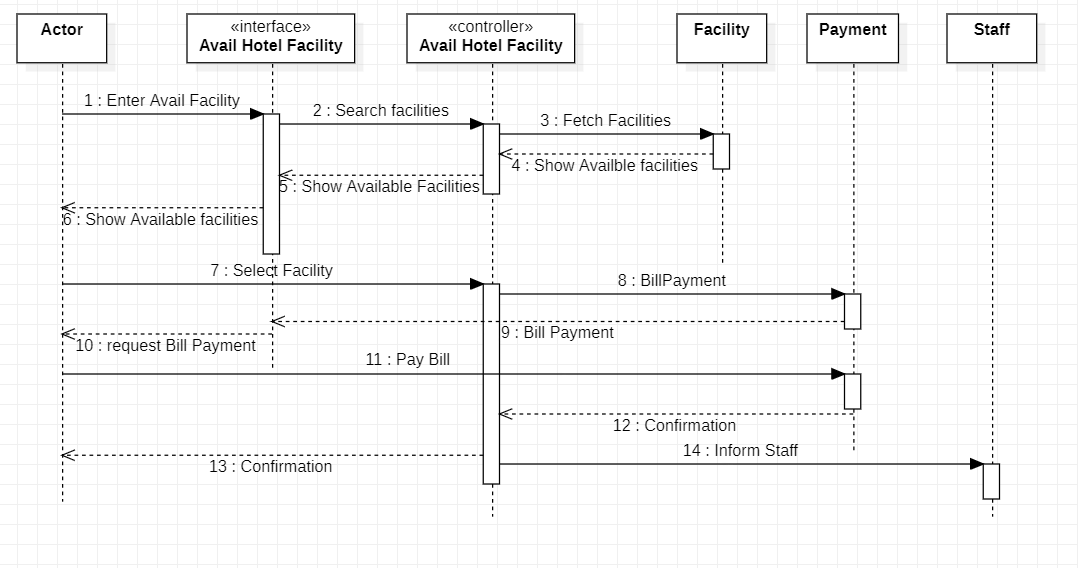
****

**Alternate Flow:**

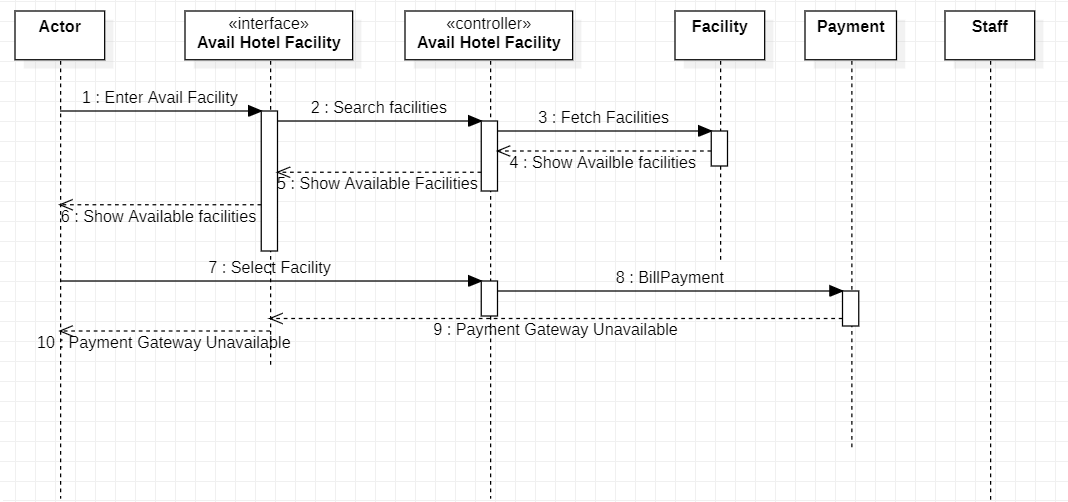
****

**Avail Hotel Facility:-**

**Basic Flow:**

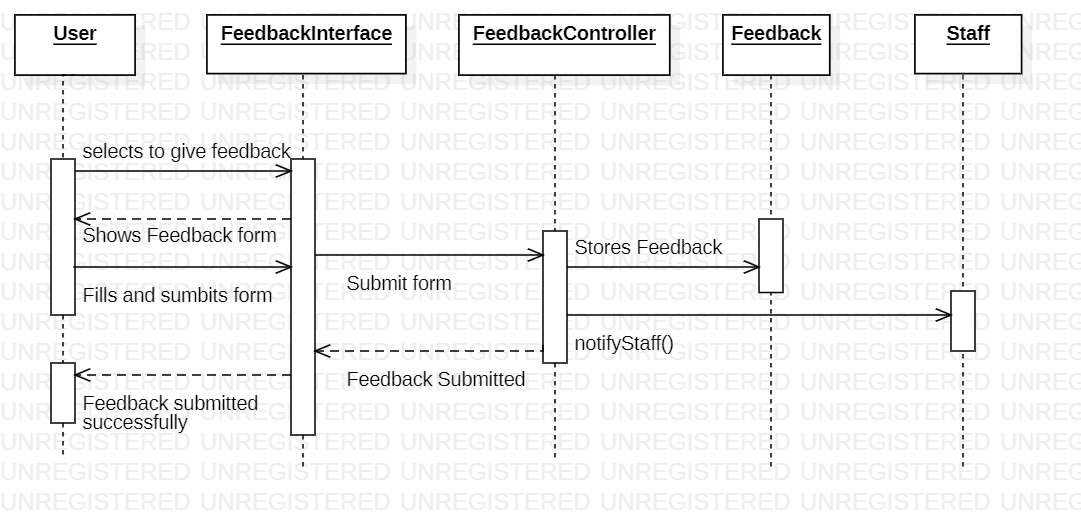
****

**Alternate Flow:**

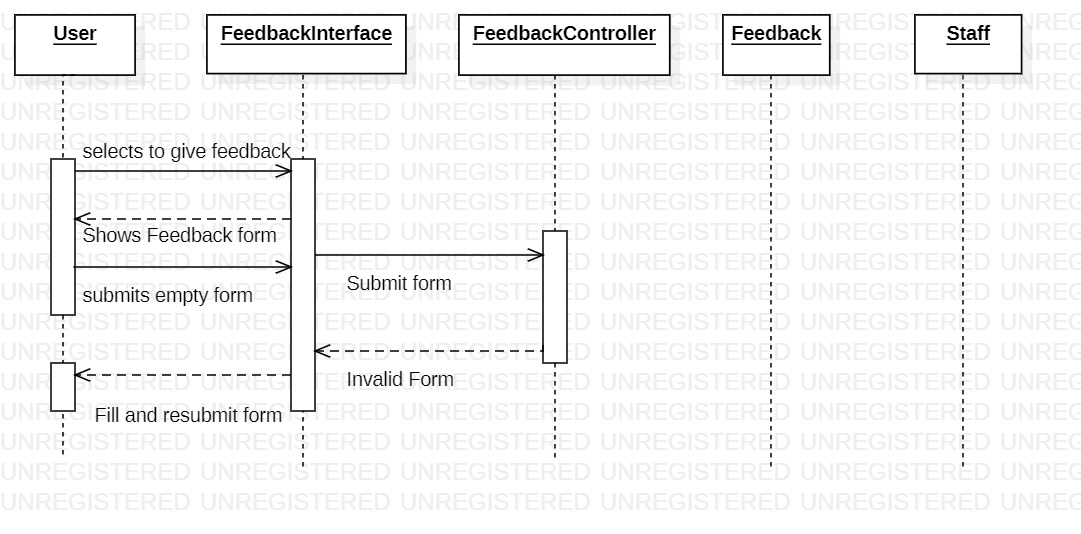
****

**Feedback:-**

**Basic Flow:**

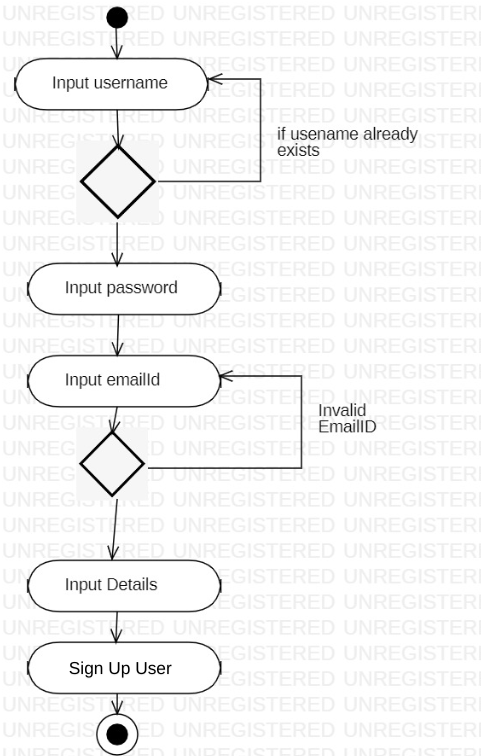
****

**Alternate Flow:**

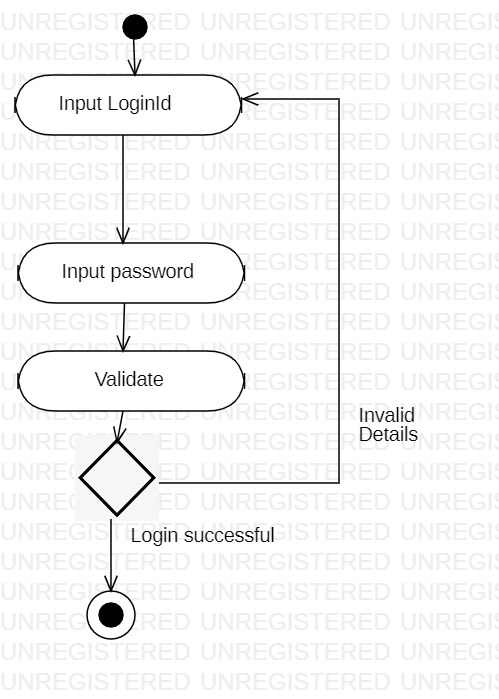
****

**Activity Diagram**

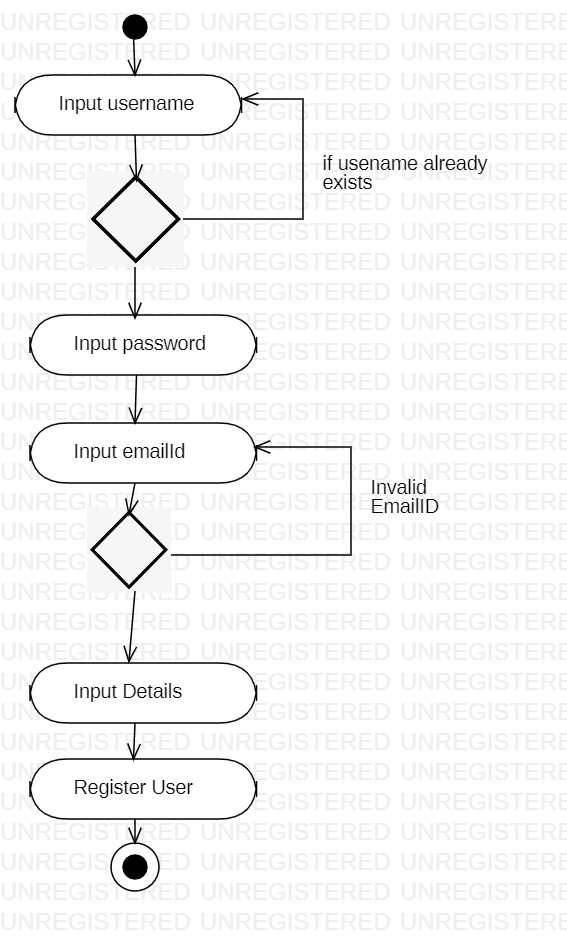
**Sign Up:-**

****

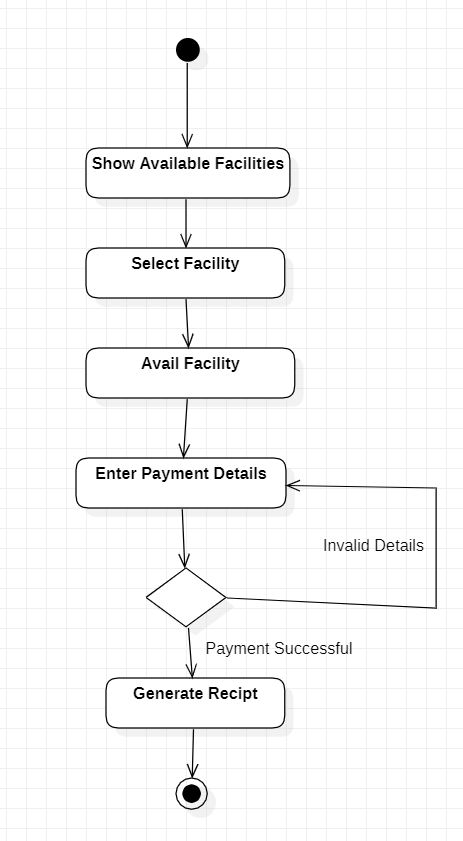
**Login:-**

****

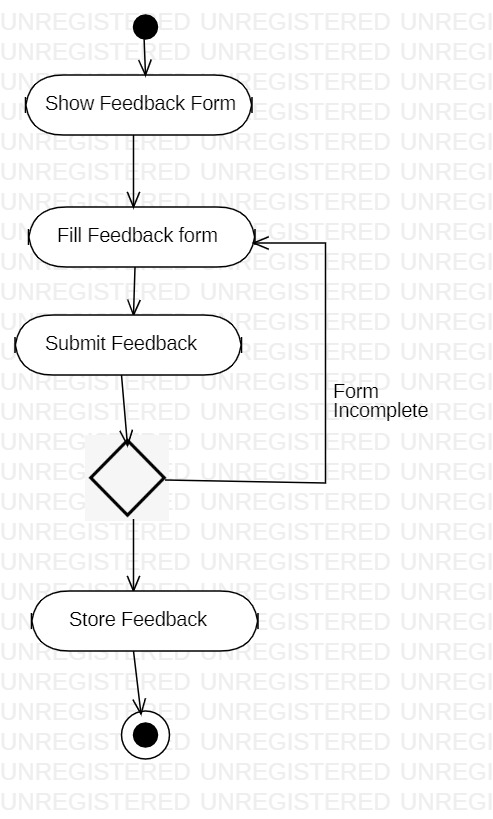
**Maintain Booking:-**

****

**Avail Hotel facility:-**

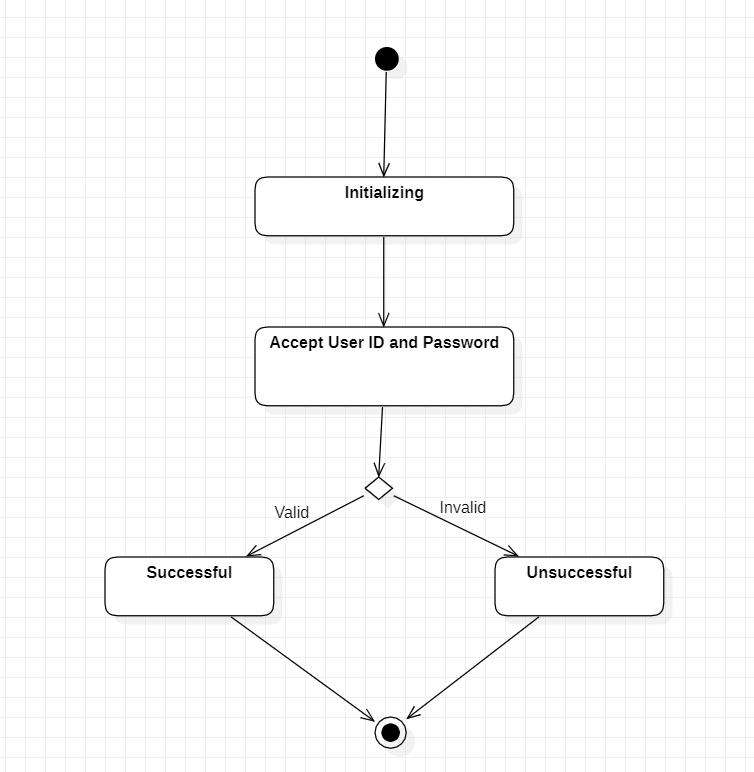
****

**Feedback:-**

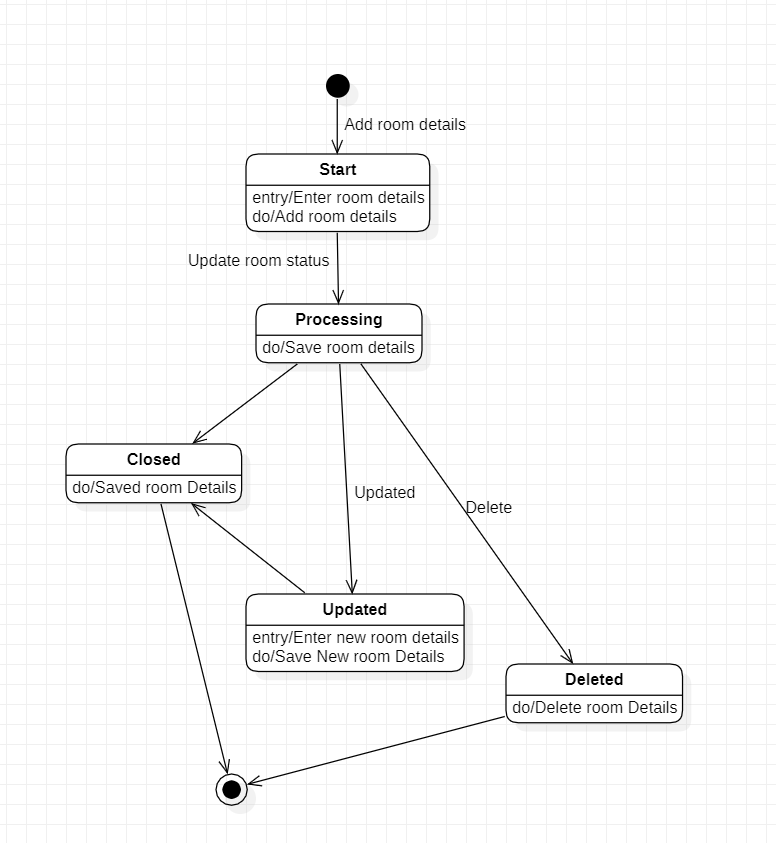


**State Chart Diagram**

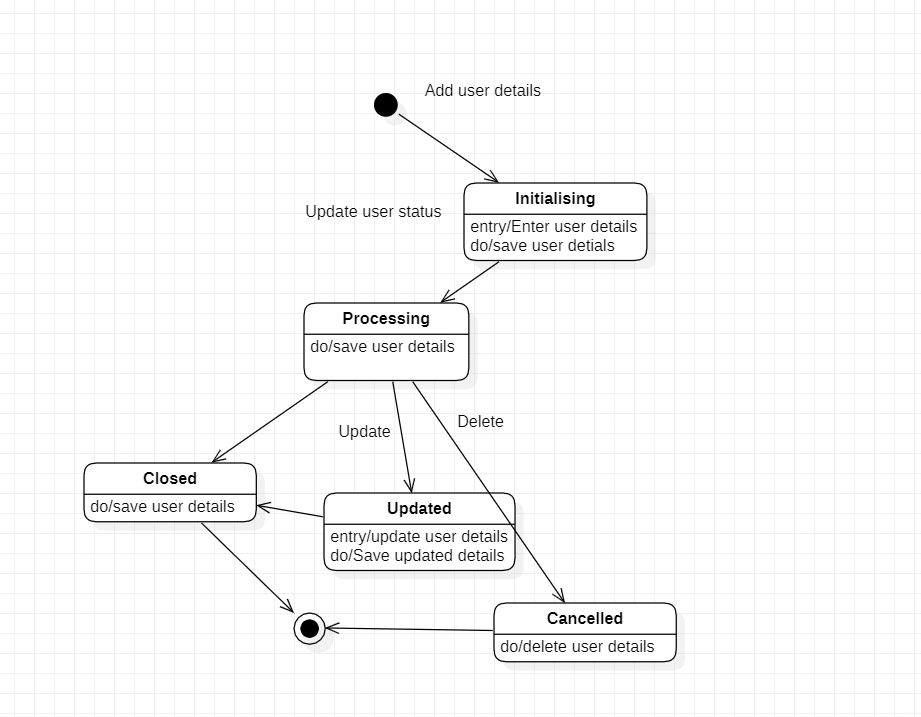
**Login:**

****

**Maintain Room Details(Add details):**

****

**Maintain User Details(Add details):**

****

**EXPERIMENT-4**

**Test Case Matrix**

**Login:**

| **Case ID** | **Scenario Case and**  **Description** | **Input1**  **User ID** | **Input2**  **Password** | **Expected**  **Output** | **Remarks** |
| --- | --- | --- | --- | --- | --- |
| TC1 | Scenario:  Login | Valid | Valid | Logged In  Successfully | - |
| TC2 | Scenario:  Login Alternative:  Invalid Entry | Valid | Invalid | Login Unsuccessful | Incorrect Password |
| TC3 | Scenario:  Login Alternative:  Invalid Entry | Invalid | Valid | Login Unsuccessful | Incorrect  UserID |
| TC4 | Scenario:  Login Alternative:  Invalid Entry | Invalid | Invalid | Login Unsuccessful | Incorrect  UserID |
| TC5 | Scenario:  Login Alternative:  User Exits | valid/invalid | valid/invalid | User Exits  System | - |

**Room Booking:**

| **Case ID** | **Scenario Case and**  **Description** | **Input 1**  **CheckIn Date** | **Input 2**  **CheckOut**  **Date** | **Input 3**  **Room Available** | **Expected Output** | **Remarks** |
| --- | --- | --- | --- | --- | --- | --- |
| TC1 | Scenario:  Room Book | Valid | Valid | Valid | Room booked  Successfully | - |
| TC2 | Scenario:  Room Book  Alternative:  Invalid Entry | Valid | Valid | Invalid | Room Book  Unsuccessful | Room not available in between those dates |
| TC3 | Scenario:  Room Book  Alternative:  Invalid Entry | Valid | Invalid | valid/invalid | Room Book  Unsuccessful | Invalid Date |
| TC4 | Scenario:  Room Book  Alternative:  Invalid Entry | Invalid | Valid | valid/invalid | Room Book  Unsuccessful | Invalid Date |
| TC5 | Scenario:  Room Book  Alternative:  User Exits | valid/invalid | valid/invalid | valid/invalid | User Comes  out of the  system | - |

**Payment**

| **Case ID** | **Scenario Case and**  **Description** | **Input 1**  **Card Number** | **Input 2**  **CVV** | **Input 3**  **DOE** | **Expected Output** | **Remarks** |
| --- | --- | --- | --- | --- | --- | --- |
| TC1 | Scenario:  Payment | Valid | Valid | Valid | Room booked  Successfully | - |
| TC2 | Scenario:  Payment  Alternative:  Invalid Entry | Valid | Valid | Invalid | Room Book  Unsuccessful | Room not available in between those dates |
| TC3 | Scenario:  Payment  Alternative:  Invalid Entry | Valid | Invalid | Valid | Room Book  Unsuccessful | Invalid CVV |
| TC4 | Scenario:  Payment  Alternative:  Invalid Entry | Invalid | valid/invalid | valid/invalid | User Comes  out of the  system | - |
| TC5 | Scenario:  Payment  Alternative:  Payment Gateway Error | valid/invalid | valid/invalid | valid/invalid | Payment Unsuccessful | - |
| TC6 | Scenario:  Payment  Alternative:  User Exits | valid/invalid | valid/invalid | valid/invalid | User comes out of the system | - |

**Maintain Room Details**

| **Case ID** | **Scenario Case and**  **Description** | **Input 1**  **Room Number** | **Input 2**  **Room Type** | **Input 3**  **Cost** | **Input 4**  **Availability** | **Confirm**  **Deletion** | **Confirm**  **Updation** | **Expected Output** | **Remarks** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| TC1 | Scenario:  Add | Valid | Valid | Valid | Valid | - | - | Details Added  Successfully | - |
| TC2 | Scenario:  Add  Alternative:  Invalid Entry | Invalid | valid/invalid | valid/invalid | valid/invalid | - | - | Room Book  Unsuccessful | This room is not available |
| TC3 | Scenario:  Add  Alternative:  Invalid Entry | valid/invalid | Invalid | valid/invalid | valid/invalid | - | - | Room Book  Unsuccessful | This room type is not available |
| TC4 | Scenario:  Add  Alternative:  Invalid Entry | valid/invalid | valid/invalid | Invalid | valid/invalid | - | - | Room Book  Unsuccessful | Room in this cost is not available |
| TC5 | Scenario:  Add  Alternative:  Invalid Entry | valid/invalid | valid/invalid | valid/invalid | Invalid | - | - | User Comes  out of the  system | This room is currently not available |
| TC6 | Scenario:  Add  Alternative:  Payment Gateway Error | valid/invalid | valid/invalid | valid/invalid | valid/invalid | - | - | Payment Unsuccessful | Payment error |
| TC7 | Scenario:  Add  Alternative:  User Exits | valid/invalid | valid/invalid | valid/invalid | valid/invalid | - | - | User comes out of the system | - |
| TC8 | Scenario: View | Valid | Valid | Valid | Valid | - | - | Room Details are displayed | - |
| TC9 | Scenario:  View  Alternative: Invalid Entry | Invalid | valid/invalid | valid/invalid | valid/invalid | - | - | Error Message is displayed | This room is not available |
| TC10 | Scenario:  View  Alternative:  Invalid Entry | valid/invalid | Invalid | valid/invalid | valid/invalid | - | - | Error Message is displayed | This room type is not available |
| TC11 | Scenario:  View  Alternative:  Invalid Entry | valid/invalid | valid/invalid | Invalid | valid/invalid | - | - | Error Message is displayed | Room in this cost is not available |
| TC12 | Scenario:  View  Alternative:  Invalid Entry | valid/invalid | valid/invalid | valid/invalid | Invalid | - | - | Error Message is displayed | This room is currently not available |
| TC13 | Scenario:  View  Alternative:  User Exits | valid/invalid | valid/invalid | valid/invalid | valid/invalid | - | - | User comes out of the system | - |
| TC14 | Scenario: Delete | Valid | Valid | Valid | Valid | Valid | - | Room Details are deleted | – |
| TC15 | Scenario: Delete | Valid | Valid | Valid | Valid | Invalid | - | Error Message is displayed | - |
| TC16 | Scenario:  Delete  Alternative:  Invalid Entry | Invalid | valid/invalid | valid/invalid | valid/invalid | - | - | Error Message is displayed | This room is not available |
| TC17 | Scenario:  Delete  Alternative:  Invalid Entry | valid/invalid | Invalid | valid/invalid | valid/invalid | - | - | Error Message is displayed | This room type is not available |
| TC18 | Scenario:  Delete  Alternative:  Invalid Entry | valid/invalid | valid/invalid | Invalid | valid/invalid | - | - | Error Message is displayed | Room in this cost is not available |
| TC19 | Scenario:  Delete  Alternative:  Invalid Entry | valid/invalid | valid/invalid | valid/invalid | Invalid | - | - | Error Message is displayed | This room is currently not available |
| TC20 | Scenario:  Delete  Alternative:  User Exits | valid/invalid | valid/invalid | valid/invalid | valid/invalid | - | - | User comes out of the system | - |
| TC21 | Scenario: Update | Valid | Valid | Valid | Valid | - | Valid | Room Details are updated | - |
| TC22 | Scenario: Update | Valid | Valid | Valid | Valid | - | Invalid | Error Message is displayed | – |
| TC23 | Scenario:  Update  Alternative:  Invalid Entry | Invalid | valid/invalid | valid/invalid | valid/invalid | - | - | Error Message is displayed | This room is not available |
| TC24 | Scenario:  Update  Alternative:  Invalid Entry | valid/invalid | Invalid | valid/invalid | valid/invalid | - | - | Error Message is displayed | This room type is not available |
| TC25 | Scenario:  Update  Alternative:  Invalid Entry | valid/invalid | valid/invalid | Invalid | valid/invalid | - | - | Error Message is displayed | Room in this cost is not available |
| TC26 | Scenario:  Update  Alternative:  Invalid Entry | valid/invalid | valid/invalid | valid/invalid | Invalid | - | - | Error Message is displayed | This room is currently not available |
| TC27 | Scenario:  Update  Alternative:  User Exits | valid/invalid | valid/invalid | valid/invalid | valid/invalid | - | - | User comes out of the system | - |