# Table of Contents

[1](#__RefHeading___Toc186_3275986575)

[1. Introduction 2](#__RefHeading___Toc188_3275986575)

[1.1 System Introduction 2](#__RefHeading___Toc190_3275986575)

[1.2 Stakeholders 3](#__RefHeading___Toc192_3275986575)

[3](#__RefHeading___Toc194_3275986575)

[2. Requirement Analysis 3](#__RefHeading___Toc196_3275986575)

[2.1 Use Case Diagram 4](#__RefHeading___Toc198_3275986575)

[2.2 Functional and Non-Functional Requirements 4](#__RefHeading___Toc200_3275986575)

[3. High Level Design 5](#__RefHeading___Toc202_3275986575)

[3.1 Classes 5](#__RefHeading___Toc204_3275986575)

[3.2 Class Diagram 5](#__RefHeading___Toc206_3275986575)

[4. Data Modeling 5](#__RefHeading___Toc208_3275986575)

[4.1 Entity Relationship Diagram 5](#__RefHeading___Toc210_3275986575)

[6](#__RefHeading___Toc212_3275986575)

[5. Detail Design 6](#__RefHeading___Toc214_3275986575)

[5.1 Screen Flow Diagram 7](#__RefHeading___Toc216_3275986575)

[6. System Testing 7](#__RefHeading___Toc218_3275986575)

[6.1 Login Testing 7](#__RefHeading___Toc220_3275986575)

[6.2 Add Data Testing 7](#__RefHeading___Toc222_3275986575)

[6.2.1 Add Room 8](#__RefHeading___Toc224_3275986575)

[6.2.2. Add Guest 8](#__RefHeading___Toc226_3275986575)

[6.2.3 Add Booking 9](#__RefHeading___Toc228_3275986575)

[6.3 Delete Data Testing 9](#__RefHeading___Toc230_3275986575)

[6.3.1 Delete Room 9](#__RefHeading___Toc232_3275986575)

[6.3.2 Delete Guest 10](#__RefHeading___Toc234_3275986575)

[6.3.3 Delete Booking 10](#__RefHeading___Toc236_3275986575)

[7. Conclusion 10](#__RefHeading___Toc238_3275986575)

# 1. Introduction

## 1.1 System Introduction

The Hotel Room Booking System is a comprehensive software solution designed to streamline the management of room, reservations, guest details and availability in a small hotel. This system addresses the essential needs of hotel management by organizing information related to rooms, guests, and bookings in a centralized database. Key features include the creation of new bookings, management of guest data, and real-time availability checks, providing a seamless booking experience for both the hotel staff and their guests. This system is implemented using HTML and CSS for the front-end interface, PHP for the server-side processing and MySQL for database management, ensuring robust and efficient operation tailored to the requirements of small-scale hotel operations.

## 1.2 Stakeholders

The stakeholders are people or organizations who have an interest in the development of the system architecture.

**Hotel Staff**

Hotel staff includes people who are interact with the system. They use system to manage reservations, manage rooms and their availability , and manage guests.

**Hotel Guests**

Guests do not interact directly with the system back-end. But still they are indirect stakeholders as their reservations, personal details and room preferences are managed through the system.

**Hotel Management**

Hotel Management relies on the system for accurate reporting, financial tracking, performance analysis etc.

**IT Developers / Support**

The people who are responsible for maintaining, troubleshooting, and upgrading the system as required. They ensure the system remains functional, secure, and up-to-date with any new requirements or technological improvements.

# 

# 2. Requirement Analysis

## 2.1 Use Case Diagram

Figure 1 – Use Case Diagram

## 2.**2 Functional and Non-Functional Requirements**

**Functional Requirements :**

* Admin can log into the system using a secure username and password.
* Admin can create, view, and delete bookings.
* The system links each booking to a specific guest and room, tracking check-in and check-out dates.
* Admin shall be able to add, view and delete room details and guests details through relevant interfaces.

**Non-Functional Requirements :**

* The system should handle multiple bookings ,add guests information efficiently without delays.
* Secure login mechanism with password encryption.
* Clear error messages and validation to guide users.
* The system should be scalable enough to bare growing number of rooms and guests.
* The system should be structured so that updates, bug fixes, or feature additions can be applied with minimal disruption.

# 3. High Level Design

## 3.1 Classes

For the Hotel Room Booking System, followings are the classes that can be identified.

* Guest
* Room
* Booking
* User / Admin

## 3.2 Class Diagram

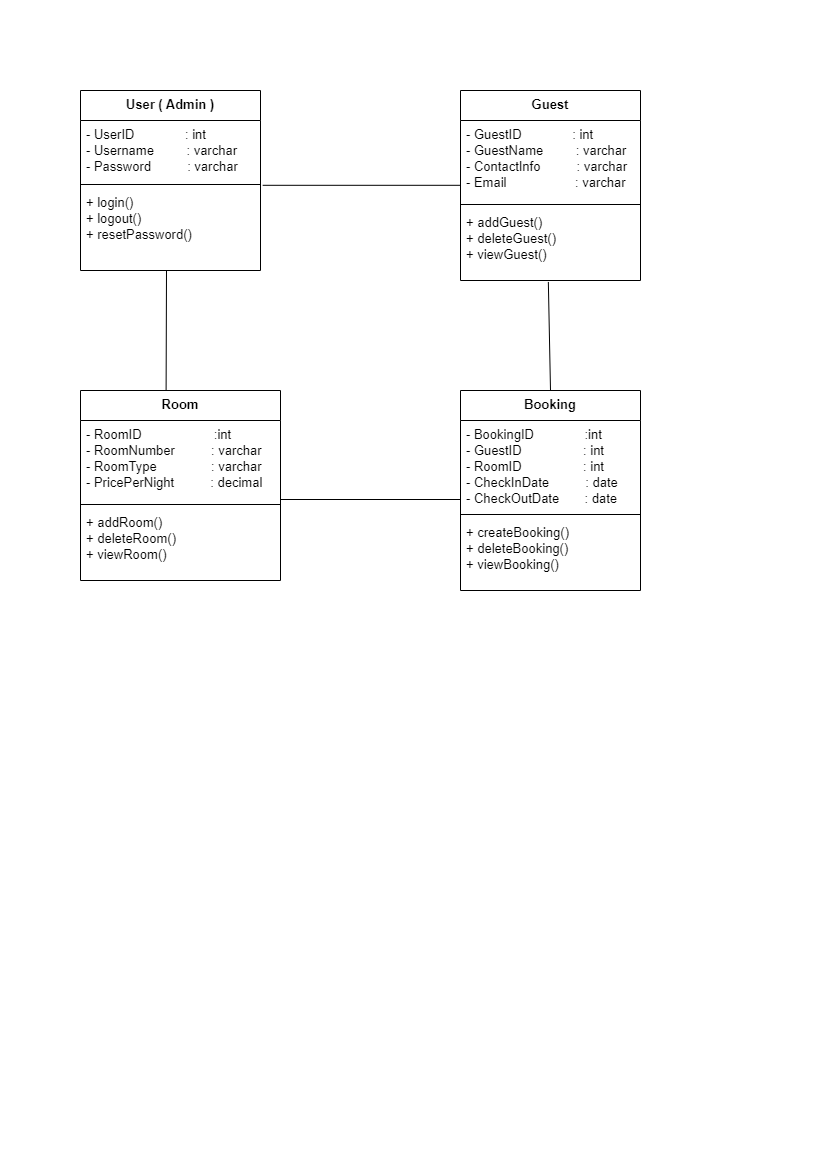
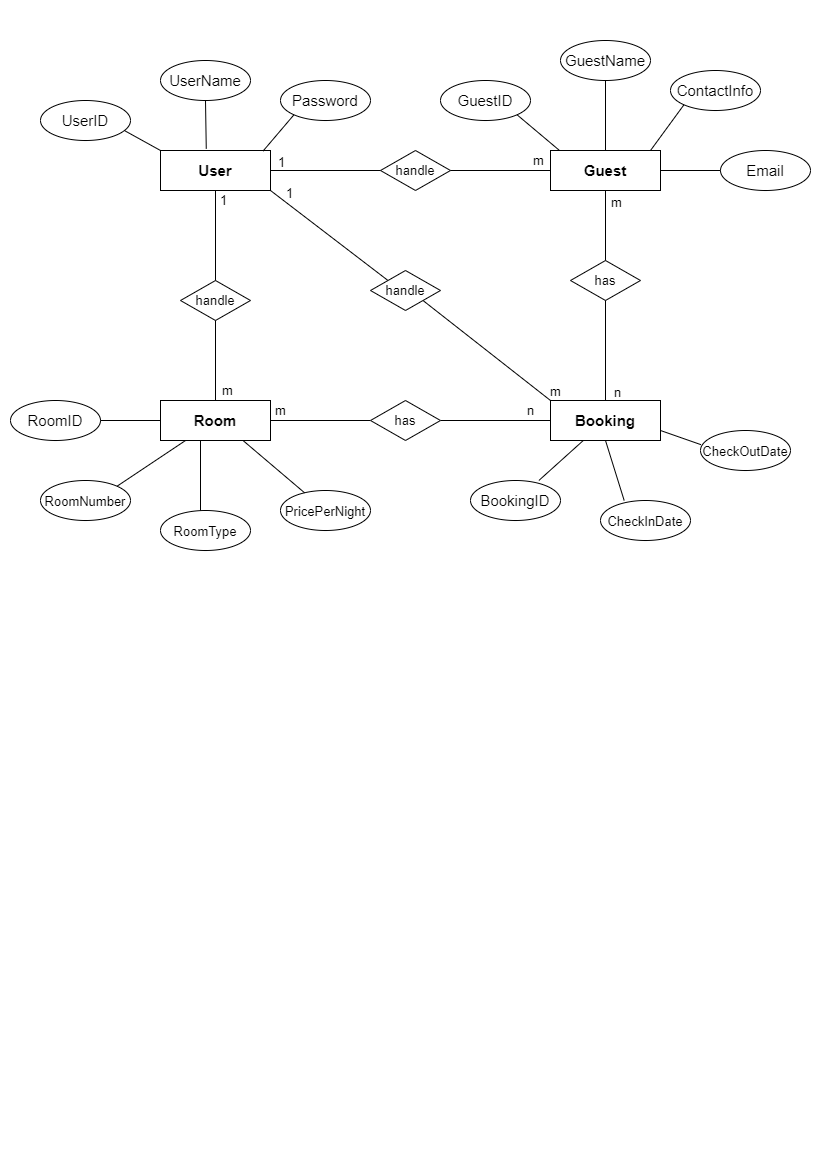


Figure 2 – Class Diagram

# 4. Data Modeling

## 4.1 Entity Relationship Diagram

Figure 3 – ER Diagram



# 

# 5. Detail Design

### 5.1 Screen Flow Diagram

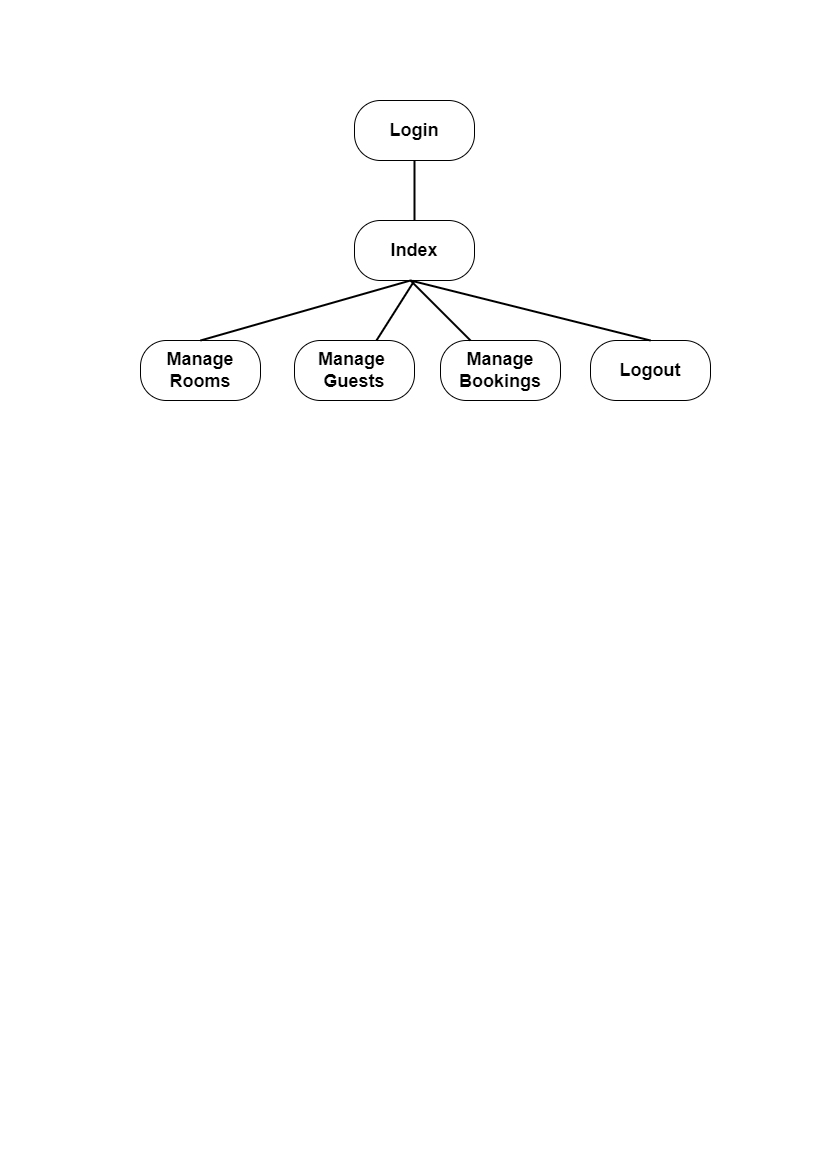
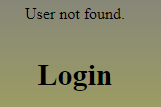
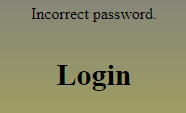


Figure 4 – Screen Flow Diagram

# 6. System Testing

## 6.1 Login Testing

* Login page will display an error message when entered incorrect credentials.



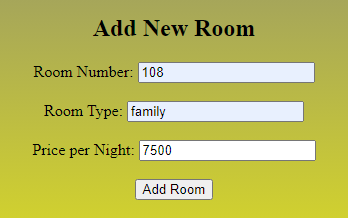
* Correct credentials will direct user to the dashboard.



## 6.2 Add Data Testing

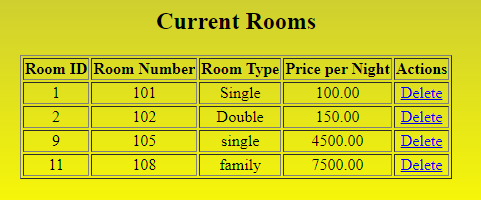
### 6.2.1 Add Room

User has to fill the Add New Room form with relevant data types, in order to add new room.



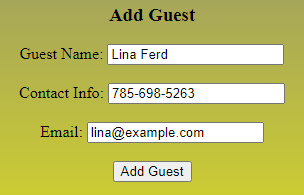
**Click to Add New Room**

After entering the room, it will show in the Current Rooms table.



### 6.2.2. Add Guest

User has to fill the Add Guest form with relevant data types, in order to add new guest.



**Click to add New Guest**

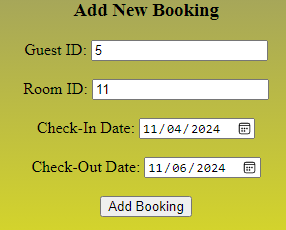
**Click to add New Guest**

Added Guest will show in the Existing Guests table.

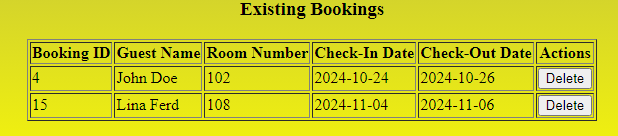


### 6.2.3 Add Booking

User has to fill the Add Booking form with relevant data types, in order to add new booking. And added booking will show in the Existing Bookings table.



**Click to add New Booking**



## **6.3 Delete Data Testing**

### 

## 6.3 Delete Data

### **6.3.1 Delete Room**

User can easily delete a room by clicking on Delete Button in the Actions column in Current Rooms table.

### 6.3.2 Delete Guest

User can easily delete a guest by clicking on Delete Button in the Actions column in Existing Guests table.

### 6.3.3 Delete Booking

User can easily delete a booking by clicking on Delete Button in the Actions column in Existing Bookings table.

# 7. Conclusion

In conclusion, the Hotel Room Booking System was designed and developed to streamline the reservation process, improve guest data management, and facilitate room availability tracking for a small hotel. By implementing essential functionalities such as room and guest management, secure login, and a user-friendly booking interface, this system successfully supports the hotel’s operational needs. Utilizing HTML, PHP, and MySQL, the system leverages a robust relational database structure and an accessible web interface that simplifies administrative tasks. Overall, this project demonstrates a practical approach to developing an admin-controlled booking system, providing a foundation for potential future enhancements such as real-time availability updates and customer-facing reservation features.