Meerab Khan, Mariam Salim

Hotel Management System

# 1-Introduction

The Hotel Management System is a small application that allows hotel staff (receptionists and managers) and guests to interact with the system. Receptionists can add new room details, check room availability, books rooms, check guests in and out, and view booking records. Guests can search for available rooms based on type (single, double, suite, etc.) or price, book a room, and view their booking history.

# 2-Functional Requirements

The system is designed for a small to medium-sized hotel providing different type of services based on the type of users [Hotel Staff (Receptionist/Manager) or Client] to mainly manage hotel bookings.

1.Hotel Staff

* Add new room to the list of available rooms
* Update or remove room details (price, type, availability)
* View all rooms (available and booked)
* View all current and past booking records
* View booking history with information about clients
* Change and cancel guest’s booking

2. Client

* Search for available rooms by room type (Single, Double, Triple, Suite)
* Search rooms by price range
* View all available rooms
* Book a room by giving personal and stay details
* View all current and past booking history

# 3-Database Structure

We will create and populate tables using MySQL.

Rooms table:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| RoomNo | Type | Price | Availability | AddedDate |

* Room No: room number is the primary key
* Type: type of room like Suite, Family size, Double, Triple
* Price: cost per night
* Availability: status (“Yes”, “No”)
* AddedDate: date the room was added

Clients table:

|  |  |  |
| --- | --- | --- |
| ClientID | Name | Contact |

* ClientID: client number is the primary key.
* Name: client’s name
* Contact: contact number or email represented as string

Bookings table:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| BookingID | RoomNo | ClientID | ClientName | Contact | ArrivalDate | DepartureDate |

BookingID: auto-increment primary key

RoomNo: foreign key from rooms’ table

ClientID: foreign key from clients' table

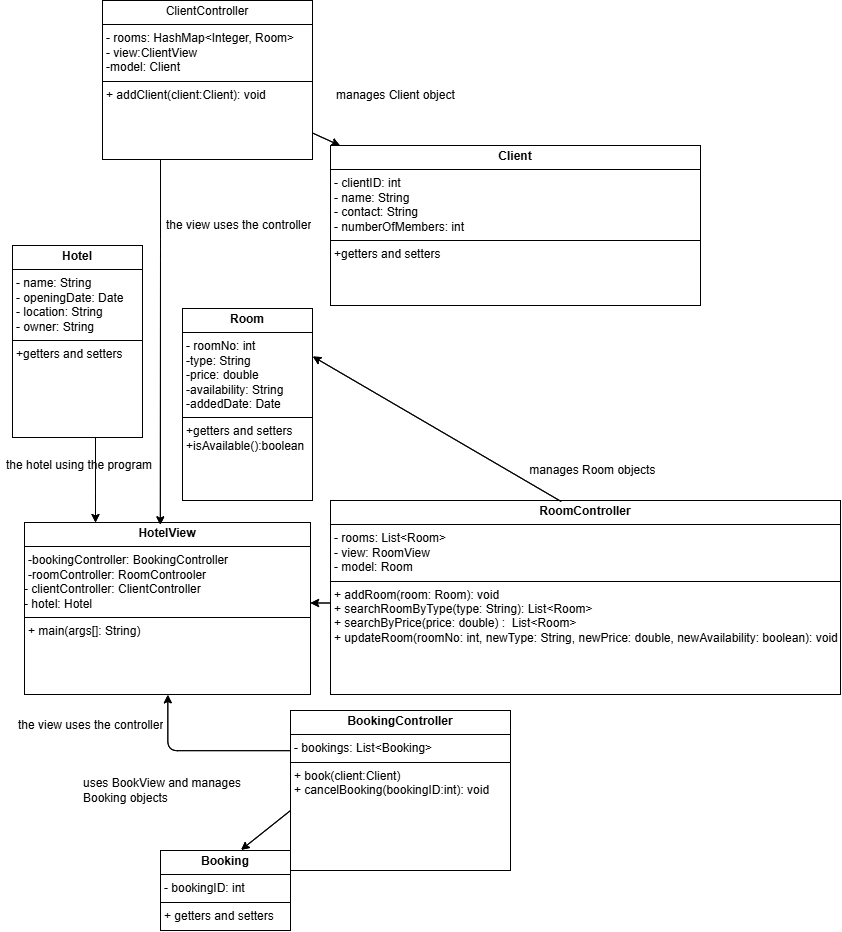
Contact: client’s contact number

ClientName: name of client

ArrivalDate: when the guest arrives at the hotel

DepartureDate: when guest leaves hotel

# 4-Implementation Details

**a. UML of the main classes**

**b. Main Methods specifications**

viewAvailableRooms(): returns a list of available rooms sorted by room number.

viewClient(roomID): lists the client renting the provided room (if it’s rented) and the number of members in the room.

searchRoomsByType(String type): returns rooms that match the targeted type of room from the Rooms table.

searchRoomByPriceRange(double min, double max): returns available rooms that fit the provided price range.

book (int nightsNo, int clientID, int roomNo, int membersNo): books a room for a client

addClient(client:Client): adds a client to the client's list

addRoom(room:Room): adds room to the list of rooms

cancelBooking(bookingID:int): sets the availability back to available based on the booking id provided that is occupied by a client.

viewBookings(): returns records from the bookings table sorted by booking id.

main (String args[]): handles the I/O from the user of the application. It will first check if the user is a guest or staff and enables them to do the actions allowed.