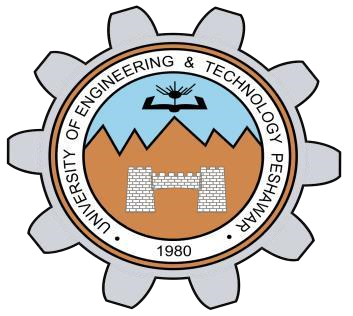
**PROJECT REPRORT**



**DBMS Final Project**

**Submitted By: Faizan**

**Project Group #19:**

FAIZAN (22pwcse2134)

ANESS UL RAHMAN (22pwcse2168)

ZEESHAN (22pwcse2117)

**Section:** B

**Submitted to:** Engr.Sumayyea Salahuddin

“On my honor, as student of University of Engineering and Technology, I have neither given nor received unauthorized assistance on this academic work”

Department of Computer Systems Engineering

University of Engineering and Technology Peshawar

### **User Panel: Booking & Contact**

#### **Booking Management**

* **Users** can:
  + View available rooms and search for rooms based on filters (e.g., room type, price, availability).
  + Book rooms by providing personal details (name, email, phone, dates).
  + View current and past bookings.
  + Modify or cancel bookings (if not yet confirmed or within a cancellation window).
  + View booking details such as room, dates, customer info, and status (confirmed/pending).

#### **Contact Management**

* **Users** can:
  + Submit a contact request with information such as:
    - Name, email, phone number.
    - Subject and message details (e.g., inquiries, issues).
  + Track the status of their contact request (e.g., unread, read, or replied).
  + View past messages.

### **Admin Panel: Room Management & Dashboard Overview**

#### **Room Management**

* **Admins** can:
  + **Add New Room**:
    - Enter room details including title, description, price, images, room type, and availability.
  + **View All Rooms**:
    - List all rooms with details such as name, price, room type, and availability.
    - Access individual room details and edit them if needed.
  + **Modify Room Details**:
    - Edit room information like price, description, or images.
    - Update room status (e.g., available or unavailable).
  + **Delete Room**:
    - Remove rooms from the system if no longer needed.

#### **Admin Dashboard Overview**

* **Quick Stats**:
  + Display **Total Rooms**: Number of rooms currently available in the system.
  + Display **Total Bookings**: Number of bookings made by users.
  + Display **Total Revenue**: Total amount generated by confirmed bookings (usually calculated from invoicing).
* **System Status**:
  + Real-time information on the **System Status** (online/offline), ensuring the hotel management system is operational and secure.
* **Database Connection**:
  + A clear indication if the **Database** is connected and secure, ensuring no issues with data integrity.
* **Quick Actions**:
  + Admin can quickly **add a new room** or **view all rooms**.
  + View **Bookings** to manage the reservations and status of rooms.
* **Messages**:
  + Admin can view messages or **contact requests** submitted by users in the **Contact Management** system.

### **Admin Interface Features**

#### **UI & Interaction**:

* **Admin Dashboard** uses Laravel Blade templates with Bootstrap components for a clean and responsive interface.
* **Quick Actions** allow admins to manage rooms, bookings, and contact messages with ease.
* Real-time insights give an overview of how the hotel is performing in terms of bookings, revenue, and system health.

### **Relationships Between Tables**:

1. **User (1) -> Booking (Many)**: One user can have many bookings.
2. **Room (1) -> Booking (Many)**: One room can have many bookings.
3. **User (1) -> Contact (Many)**: One user can submit many contact requests.
4. **Room** and **Contact** have no direct relationship.

***Table details and business rules:*  
User Table:**

| **Field** | **Description** |
| --- | --- |
| **Id** | Primary key, unique identifier for each user. |
| **Name** | The user's full name. |
| **Email** | The user's email address. |
| **Phone** | The user's phone number. |
| **Usertype** | Type of the user. Can be 'admin' or 'user'. |
| **email\_verified\_at** | Timestamp when the user's email was verified. |
| **Password** | The hashed password for authentication. |
| **remember\_token** | Token for remembering the user session. |
| **current\_team\_id** | ID of the current team associated with the user (if applicable). |
| **profile\_photo\_path** | Path to the user's profile photo (if applicable). |
| **created\_at** | Timestamp when the user was created. |
| **updated\_at** | Timestamp when the user details were last updated. |

**Room Table**:

| **Field** | **Description** |
| --- | --- |
| **id** | Primary key, unique identifier for each room. |
| **room\_title** | Title or name of the room (e.g., "Luxury Suite", "Standard Room"). |
| **image** | Path to an image representing the room. |
| **description** | Detailed description of the room, including features, size, etc. |
| **price** | Price per night or booking period for the room. |
| **wifi** | Indicates if the room has Wi-Fi (yes/no). |
| **room\_type** | Type of room (e.g., "Single", "Double", "Suite"). |
| **created\_at** | Timestamp when the room was created. |
| **updated\_at** | Timestamp when the room details were last updated. |

**Booking Table**:

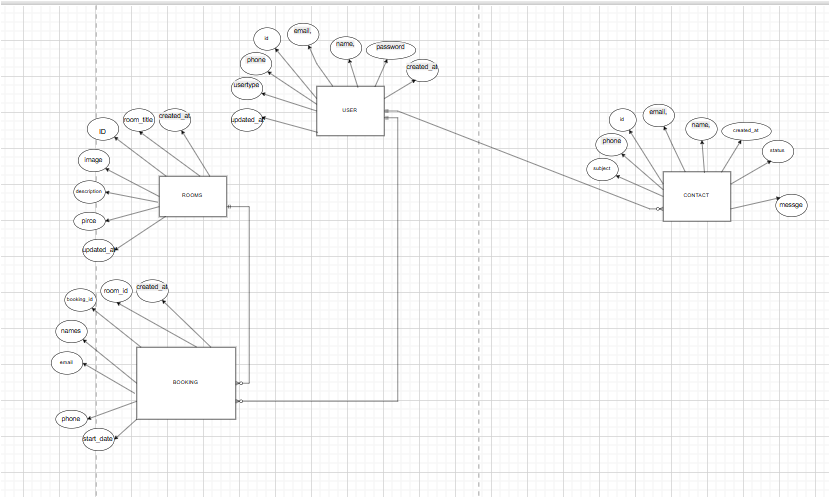
| **Field** | **Description** |
| --- | --- |
| **id** | Primary key, unique identifier for each booking. |
| **room\_id** | Foreign key referencing the **Room** table (the room that was booked). |
| **name** | Name of the person who made the booking. |
| **email** | Email address of the person who made the booking. |
| **phone** | Phone number of the person who made the booking. |
| **start\_date** | The start date of the booking. |
| **end\_date** | The end date of the booking. |
| **created\_at** | Timestamp when the booking was created. |
| **updated\_at** | Timestamp when the booking details were last updated. |

**Contact Table:**

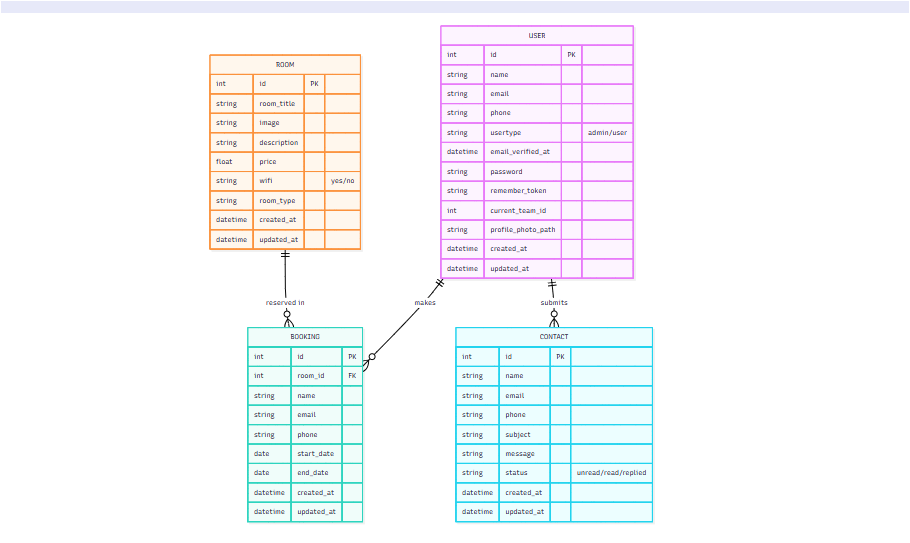
| **Field** | **Description** |
| --- | --- |
| **id** | Primary key, unique identifier for each contact entry. |
| **name** | Name of the person submitting the contact request. |
| **email** | Email address of the person submitting the contact request. |
| **phone** | Phone number of the person submitting the contact request. |
| **subject** | The subject or title of the contact message. |
| **message** | The content or body of the contact message. |
| **status** | Current status of the message (e.g., 'unread', 'read', 'replied'). |
| **created\_at** | Timestamp when the contact entry was created. |
| **updated\_at** | Timestamp when the contact entry was last updated. |

**Finalized Conceptual Schema:**

**ERD (Entity Relationship Diagram):**



**Relational schema:**



**Final Normalized Forms:**

## 4. Normalization

## The database is normalized to 3NF:

## - 1NF: All tables have atomic columns (no repeating groups).

## - 2NF: All non-key attributes are fully functionally dependent on the primary key.

## - 3NF: No transitive dependencies exist; all attributes are dependent only on the primary key.

## **Conclusion**

The Restaurant Management System simplifies restaurant operations and enhances the customer experience. With a modular architecture and modern technologies, it can be extended further by integrating features like online payments, notification emails, or multi-branch support.

**References:**

* ChatGPT
* Youtube
* Draw.io
* Cursor AI
* Claude