# *DBS PROJECT REPORT*

*PROJECT TITLE*

*Hotel Management System*


**Submitted to:**

SIR REHAN

**Submitted by:**

Shaheer Arshad (2020-CS-639)

Fatima Azeem (2020-CS-646)

M. Ateeb Khan (2020-CS-630)

Muhammad Jafar (2020-CS-644)

**University of Engineering and Technology,**

**New Campus, Lahore.**

# HOTEL MANAGEMENT SYSTEM

## Introduction

Database Table for Hotel Management System-Hotel management system is ever green project for real life.

Visitors or client comes in hotel and book his/her room for some duration. Visitor will provide his
own personal details and will get room number according to availability. Some charges for every room according to facility. After payment, visitor will checkout from the hotel.

Hotel needs to maintain the record of Customers and reserve rooms beforehand. Customers should be able to know the availability of the rooms on a particular date. They should be able to reserve the available rooms according to their need in advance. To make their stay comfortable, they are provided with food and other services. The record of the food taken by each customer and the services availed by the customer should be kept. These records help in generating bill.

## Functionality

The functionality of the system is that it stores the information about the employees of the hotel and the expenses they are given in extra by maintaining the records in respective tables.

The information about bookings and rooms is kept in the respective tables. Whenever a customer enters the hotel, he/she is asked for some information to make record of customer and allotted a room according to their taste. The hotel also offers food to the residents of the hostel. The record is maintained the customer is charged a bill at the time of checkout which can be paid in numerous ways.

# End Users

The end user of our system are the administration of the hotel that maintain records of employees.

The records for customers, bookings and other services are maintained by the employees of the respective field.

# Tables Description

## BOOKING

This table allow the customer to book his/her room. The management system gives him the Room ID and Booking Id which is uniquely identity for every customer allow the customer and customer Id so it is easy for the management to identify every customer. In this table user can check their arrival and departure time.

## EMPLOYEE

This is the employee of hotel, who serve the hotel customer and take care their needs. Provides their customer what they want. Every employee has assigned a unique identity through which we can easily get their information. Every employee has login Id and Password through which it can assure their presence.

## CUSTOMER

Customer table is used to save the whole data for all the customer who used the hotel. Hotel save their data by their Id which is unique for every customer and we can get their name, address, contact info and profile image. Customer have the rights whenever he wants to leave the room before booking contract.

## REMINDER

Reminder has a booking id and reminder id which have a reminder detail of the customer. Reminder performs a job in case of when a customer has reached their time limit of booking reminder reminds the customer If they want to extend their booking then they have to pay mor cost, otherwise his/her booking contract over soon. So, it alerts the customer of his /her booking contract. End user is Management and Customer.

## ROOMTYPE

The room is allotted to the customer and every room is unique Id. Every room has different shape and size and have different cost. So, it is up to the customer what type of room he wants.

## ROOM

Room has different term and condition. Some room has air condition and luxury furniture have different cost. Cost depends on the condition provides to the customer.

## ITEM

The hotel provides their customer food facilities. They have menu for different item and have unique itemId and have a different cost so customer can order for different item.

## ORDER

The table holds the record of items ordered by a certain booking and the price of that order. It creates a relation between Items and bookings.

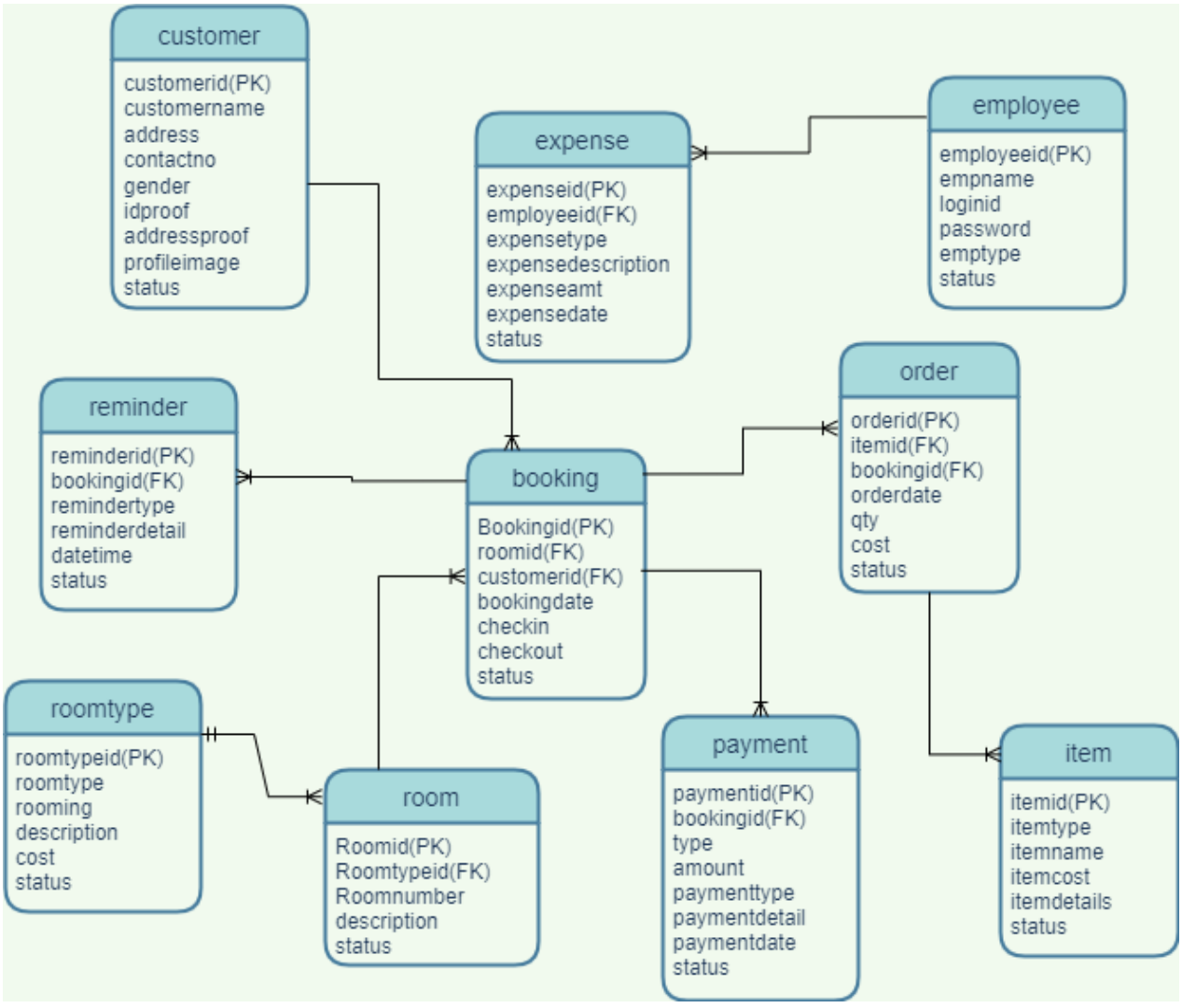
## EXPENSE

This table maintains the record of expenses that are given to employees for there ease.

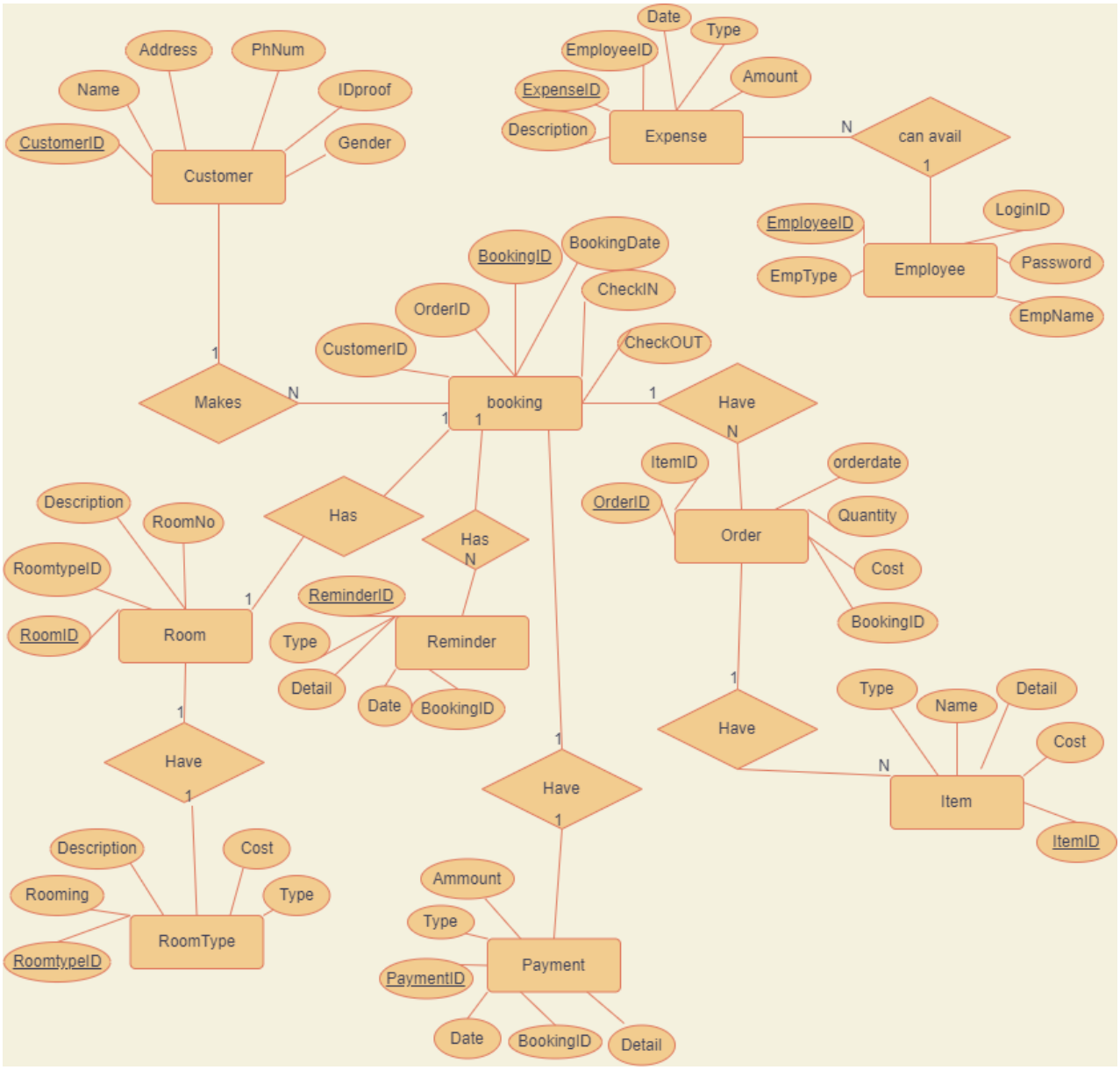
## PAYMENT

This table hold the payment for every customer through their ID which is referenced from customer ID. The payment is calculated by on the basis of day. Customer pay their payment according to their use of days.

# ER Diagram



# Relational Model



# Conceptual Level

Database normalization is the process of restructuring a relational database in accordance with a series of so-called normal forms in order to reduce data redundancy and improve data integrity. Generally, if a database is normalized until third normal form, then it is considered to be normalized. We tried to normalize the database until third normal form.

## BOOKING

1. **Primary Key:** BookingId
2. **Foreign Key:** CustomerId, RoomID
3. **Domains:** Unique Primary Key, Checkout date must be greater than Checkin
4. **Integrity Checks:** not null
5. **Functional Dependances:** BookingID → bookingdate, checkin, checkout

## EMPLOYEE

1. **Primary Key:** EmployeeID
2. **Foreign Key:** ExpenseID
3. **Domains:** Unique Primary Key, Password must be strong
4. **Integrity Checks:** not null
5. **Functional Dependances:** EmploeeID → name, login, password, type

## CUSTOMER

1. **Primary Key:** CustomerID
2. **Foreign Key:** No
3. **Domains:** Unique Primary Key
4. **Integrity Checks:** not null
5. **Functional Dependances:** CustomerID → name, addr, contact, gender, idproof, adrproof

## REMINDER

1. **Primary Key:** ReminderID
2. **Foreign Key:** BookingID
3. **Domains:** Unique Primary Key, Reminder Date Must be Within Checkin and Checkout
4. **Integrity Checks:** not null
5. **Functional Dependances:** ReminderID → typr, detail, date

## ROOMTYPE

1. **Primary Key:** RootypeId
2. **Foreign Key:** No
3. **Domains:** Unique Primary Key
4. **Integrity Checks:** not null
5. **Functional Dependances:** RoomtypeId → type, rooming, description, cost

## ROOM

1. **Primary Key:** RoomId
2. **Foreign Key:** RoomtypeId
3. **Domains:** Unique Primary Key, Room Number Must be Unique
4. **Integrity Checks:** not null
5. **Functional Dependances:** RoomId → type, number, description

## ITEM

1. **Primary Key:** ItemId
2. **Foreign Key:** No
3. **Domains:** Unique Primary Key
4. **Integrity Checks:** not null
5. **Functional Dependances:** ItemId → type, name, cost, detail

## ORDER

1. **Primary Key:** OrderId
2. **Foreign Key:** ItemId, BookingId
3. **Domains:** Unique Primary Key, Order Date between Checkin and Checkout of Booking
4. **Integrity Checks:** not null
5. **Functional Dependances:** OrderId → date, qty, cost

## EXPENSE

1. **Primary Key:** ExpenseId
2. **Foreign Key:** EmployeeId
3. **Domains:** Unique Primary Key
4. **Integrity Checks:** not null
5. **Functional Dependances:** ExpenseId → type, date, description, amount

## PAYMENT

1. **Primary Key:** PaymentId
2. **Foreign Key:** BookingId
3. **Domains:** Unique Primary Key, Payment Date between Checkin and Checkout of Booking
4. **Integrity Checks:** not null
5. **Functional Dependances:** PaymentId → type, amount, detail, date

# Internal View

**Customer Detail**: This view shows all information of all the customers between 1-1-2021 to 29-4-2022.

**Employee Detail**: This view gives the information of all employees having allowances.

**V. I. P Customers:** This table gives the information of V. I. Ps staying in the Hotel.

**Customers to Remind:** This table gives the information of all the customers whose booking is ending today.

# Data Dictionary

## Table name: booking

| Column | Type | Index | Description |
|---|---|---|---|
| ***Bookingid*** | int(10) | Primary key | Booking ID |
| roomid | int(10) | Foreign key | Room ID |
| customerid | int(10) | Foreign key | Customer ID |
| bookingdate | date | Notnull | Booking Date |
| checkin | datetime | Notnull | Check IN |
| checkout | datetime | Notnull | Check OUT |
| status | varchar(10) | Notnull | Status |

## Table name: order

| Column | Type | Index | Description |
|---|---|---|---|
| ***orderid*** | int(10) | Primary key | Order ID |
| itemid | int(10) | Foreign key | Item ID |
| bookingid | Int(10) | Foreign key | Booking ID |
| orderdate | date | Not null | Order Date |
| qty | Float(10,2) | Not null | Quantity |
| cost | Float(10,2) | Not null | Cost |
| status | varchar(10) | Not null | Status |

## Table name: employee

| Column | Type | Index | Description |
|---|---|---|---|
| ***employeeid*** | int(10) | Primary key | Employee ID |
| empname | varchar(25) | Not null | Employee Name |
| loginid | varchar(25) | Not null | Login ID |
| password | varchar(100) | Not null | Password |
| emptype | varchar(25) | Not null | Employee Type |
| status | varchar(10) | Not null | Status |

## Table name: customer

| Column | Type | Index | Description |
|---|---|---|---|
| ***customerid*** | int(10) | Primary key | Customer ID |
| customername | varchar(50) | Not null | Customer Name |
| address | text | Not null | Address |
| contactno | varchar(15) | Not null | Contact Number |
| gender | varchar(10) | Not null | Gender |
| idproof | varchar(100) | Not null | Id Proof |
| addressproof | varchar(100) | Not null | Address Proof |
| profileimage | varchar(100) | Not null | Profile Image |
| status | varchar(10) | Not null | Status |

## Table name: reminder

| Column | Type | Index | Description |
|---|---|---|---|
| ***reminderid*** | int(10) | Primary key | Reminder ID |
| bookingid | int(10) | Foreign key | Booking ID |
| remindertype | varchar(25) | Not null | Reminder Type |
| reminderdetail | text | Not null | Reminder Detail |
| datetime | Datetime | Not null | Date and Time |
| status | varchar(10) | Not null | Status |

## Table name: roomtype

| Column | Type | Index | Description |
|---|---|---|---|
| ***roomtypeid*** | int(10) | Primary key | Roomtype ID |
| roomtype | varchar(100) | Not null | RoomTtype |
| roomimg | varchar(100) | Not null | Room Image |
| description | text | Not null | Description |
| cost | Float(10,2) | Not null | Cost |
| status | varchar(10) | Not null | Status |

# Table name: room

| Column | Type | Index | Description |
|---|---|---|---|
| ***Roomid*** | int(10) | Primary key | Room ID |
| Roomtypeid | Int(10) | Foreign key | Room Type ID |
| Roomnumber | Varchar(10) | Not null | Room Number |
| description | text | Not null | Description |
| status | varchar(10) | Not null | Status |

# Table name: item

| Column | Type | Index | Description |
|---|---|---|---|
| ***itemid*** | int(10) | Primary key | Item Id |
| itemtype | varchar(25) | Not null | Item Type |
| itemname | varchar(100) | Not null | Item Name |
| itemcost | float(10,2) | Not null | Item Cost |
| itemdetails | text | Not null | Item Details |
| status | varchar(10) | Not null | Status |

# Table name:expense

| Column | Type | Index | Description |
|---|---|---|---|
| ***expenseid*** | int(10) | Primary key | Expense ID |
| employeeid | int(10) | Foreign key | Employee ID |
| expensetype | varchar(20) | Not null | Expense Type |
| expensedescription | text | Not null | Expense Description |
| expenseamt | float | Not null | Expense Amount |
| expensedate | date | Not null | Expense Date |
| status | varchar(10) | Not null | Status |

## Table name: payment

| Column | Type | Index | Description |
|---|---|---|---|
| ***paymentid*** | int(10) | Primary key | Payment ID |
| bookingid | int(10) | Foreign key | Booking ID |
| type | varchar(20) | Not null | Type |
| amount | float(10,2) | Not null | Amount |
| paymenttype | Varchar(20) | Not null | Payment Type |
| paymentdetail | text | Not null | Payment Detail |
| paymentdate | date | Not null | Payment Date |
| status | varchar(10) | Not null | Status |

**THE END** 😊