Sri Lanka Institute of Information Technology

Project report

MLB_17.01_02

Hotel Reservation System for Special Events

Information Systems and Data Modeling – IT1090

B.Sc. (Hons) in Information Technology

SLIIT
Discover Your Future

PROJECT ID	MLB_17.01_02
CASE STUDY NAME	Hotel Reservation System for Special Events
CAMPUS/CENTER	Malabe Campus

Group Details:

	Student Registration Number	Student Name
1	IT22562142	BANDARA H.M.K.M.
2	IT22560544	LAKSHAN K.K.C.
3	IT22561084	JEESARA K.G.N.
4	IT22638540	CHATHURANGA K.K.G.H.
5	IT20039486	RANATHUNGA R.L.W.U

Contents

1 Introduction

This project entails developing a hotel reservation system for special events. It plans to provide consumers with a platform where they can browse for available hotel options and make reservations for specific events. To build such a system, several technologies must be investigated and comprehended. These include multi-tiered design, server and client-side scripting techniques, implementation technologies such as JSP, computer languages such as Java, JavaScript, and HTML, and relational databases such as MySQL or Access.

The hotel reservation system will function as a virtual platform for customers to browse the hotel catalog and select the best accommodation for special events. The desired rooms or packages can be added to a reservation basket. During the checkout process, the system will prompt the user to submit more information in order to confirm the reservation. Billing information, delivery preferences (where applicable), and payment information such as credit card numbers may be included.

2 Hypothetical Scenario

Hypothetical Reservation System Scenario for Hotel Phoenix:

Hotel Phoenix is known for its luxurious accommodations and great service, catering to a select clientele looking for premium experiences. The hotel's reservation system is intended to make bookings easier and to improve the entire guest experience. Let's look at the system's features and functionality:

1. User Interface: The Hotel Phoenix website has an attractive and user-friendly interface that exudes refinement while highlighting the hotel's premium amenities, services, and special events.

2. Guest User Features: As a guest user, you can explore the hotel's offers and make educated decisions about your stay by accessing various sections of the website. There are the following functionalities available:

a. Promotions and Special Events: Hotel Phoenix hosts exclusive promotions and special events such as gala dinners, themed parties, and spa retreats.

b. Feedback and Testimonials: Guests' feedback and testimonials can be read by users, offering vital insight into the hotel's great service and overall guest happiness.

c. Accommodation Options: Guests can choose from a variety of room categories, each with exquisite facilities, spectacular views, and personalized services.

d. Availability and Reservations: Users can check real-time availability of rooms and make reservations for their preferred room type(s) by entering their preferred dates of stay.

e. Frequently Asked Questions: Guests get access to a comprehensive FAQ section that answers frequently asked questions about the hotel's facilities, amenities, policies, and neighboring attractions.

f. Signup: By clicking on the signup icon, guests can create a user account and have access to extra services and perks.

g. Login: Registered users can use their credentials to log in to the system, giving them access to personalized features and information.

3. Registered User services: By creating an account, guests receive access to a variety of additional services and perks, including:

a. Reservation Management: Within the system, registered customers may easily manage their reservations, check booking details, adjust room choices, and make secure payments.

b. Reservation Requests: Users can make special requests, such as room preferences, dietary restrictions, or certain facilities, allowing the hotel to customise their stay accordingly.

c. Experiences and comments: Registered users can share their outstanding Hotel Phoenix experiences, provide comments, and rate their stay, all while contributing to the hotel's reputation and constant improvement.

d. Account Management: Users can change their personal information, such as contact information, preferences, and notification settings, to ensure a personalized and seamless experience.

e. Contact Us: The system has a "Contact Us" button that displays the hotel's contact information and allows users to communicate directly with hotel staff for questions, assistance, or special requests.

f. Direct Messaging: Using the direct messaging option, users can contact with professional guest services representatives who give tailored support and immediately address specific inquiries.

4.Features of the System Admin and Manager:

The system administrator and manager have enhanced access within the reservation system, ensuring its smooth functioning and providing sophisticated features:

a. System Administrator:

- Staff Account Management: The admin is in charge of managing staff accounts, which includes creating new accounts, assigning responsibilities, modifying access credentials, and deactivating accounts as needed.

- User Account Management: The administrator is in charge of user accounts, guaranteeing data security, validating user credentials, and resolving any account-related difficulties.

- Reservation Approval: The administrator analyzes and approves reservation requests, ensuring that all bookings are in accordance with the hotel's policies and availability.

- Refunds and Cancellations: Once approved cancellation requests are received, the admin handles refunds as needed, in accordance with the hotel's cancellation rules.

- Website material Management: The administrator is in charge of updating and managing the hotel's website material, which includes photographs, descriptions, and special event information.

- Social Media Integration: The administrator is in charge of integrating the hotel's social media accounts, assuring regular updates, connecting with guests, and sharing compelling content.

b. management: - Report Generation: To get useful insights and make educated decisions, the management prepares comprehensive reports by examining key performance metrics, occupancy rates, revenue, and guest comments.

- Fund Management: The manager is in charge of the hotel's finances, including budgeting, tracking spending, and optimizing revenue streams.

- Staff Salary Management: The manager is in charge of staff payroll, ensuring that all employees are paid accurately and on schedule.

- Operational Efficiency: The manager is responsible for optimizing operational processes, identifying areas for improvement, and putting plans in place to improve visitor satisfaction and overall efficiency.

3 Requirement Analysis

3.1 Main Requirements

3.1.1 Functional Requirements

Hotel Phoenix is a sophisticated hotel reservation system created exclusively for special events. It connects consumers to the system in a seamless manner to allow event bookings, delivering a smooth and easy experience for all parties involved. Hotel Phoenix's extensive set of features and functionality enables visitors, registered users, administrators, and developers to interact with the system in various ways that correspond to their roles and responsibilities.

1. Guest and Registered User (Front-end Access):

User Requirements:

-Guests can donate money to the hotel's designated charity without logging in.

- Guests can view special event package promotions.

- Guests can view FAQs about special events.

- Guests can examine feedback from prior special events.

- Guests can register for the system by giving the necessary information.

- Guests can check availability for event reservations.

- Visitors can explore exclusive event details and packages.

- Registered users can log in to the system using their credentials.

- Registered users can make reservations for special events.

- Registered users can comment on special events and share their experiences.

- Registered users have the option to cancel their reserved event bookings.

- Registered users can change their account information.

- Registered users can contact Hotel Phoenix via the contact page of the website.

- Registered users can communicate with the event planner via the system's communication channels.

System Requirements:

- The system should validate the guest's credentials.

- Donation information should be recorded and tracked by the system.

- The system should present available special event promotions and feedback.

- The system should validate the registration information and create a user account.

- On the website, the system should reflect event reservation availability.

- The system should display special event details and packages.

- The login credentials entered by registered users should be validated by the system.

- The system should submit the event reservation request to the event organiser and record the information.

- The system should allow users to contribute feedback and experiences about special events.

- The system should provide feedback uploading requests to the event organiser for approval and record the information.

- The system should communicate event cancellation requests to the event planner and save the information.

- The system should save and store changed user account information.

- The system should have channels for users to contact Hotel Phoenix or the event planner assigned to them.

2. Event Planner (Back-end Access):

User Requirements:

-Event planners can use their credentials to log onto the system.

- Special events can be planned and organized by event planners.

- Event planners can handle event specifics such as the date, time, location, and prerequisites.

- Event planners can work with visitors and other staff members to coordinate.

- Event organisers can inform guests about event updates and changes.

- Event organizers can create event reports.

System Requirements:

- The system should validate the event planner's login credentials.

- Event planners should be able to create and manage special events using the system.

- The system should save event information such as the date, time, location, and requirements.

- The system should have channels for event planners to communicate with attendees and staff members.

- Event planners should be able to provide event updates and modifications to guests via the system.

- For event planners, the system should generate event reports.

3. Administrator and Manager (Backend Access):

User Requirements:

-Administrators sign on to the website using their login credentials.

- Administrators have the authority to approve reservations for special events.

- Administrators can create and delete staff accounts.

Administrators have the ability to activate and cancel user accounts.

- Administrators have the ability to approve canceled reservations, delete marked reservations from the calendar, and arrange reimbursements.

- Administrators can change special event details, such as information, photographs, videos, and packages.

Administrators can add and remove shared experiences.

- Administrators can monitor member feedback, contacts, and reviews.

- Managers can access the system with their login credentials.

- Managers can create reports such as donation reports, reservation reports, and event booking reports.

- Managers can oversee gifts and funds.

- Managers can control staff wages.

System Requirements:

-User login credentials should be validated by the system.

-Details of deleted safari information and member accounts should be erased by the system.

-The system should update information about downgraded users and accounts that have been updated.

3.1.2 Non-Functional Requirements

Quality characteristics are another name for non-functional criteria. It describes the user-visible aspects of the system that are not directly tied to specific functions.

- Availability - 24 hour availability will be provided till the internet connections fail.
- Security - Prevent unauthorized access to the system, and only the administrator has direct access to the system database. Furthermore, by providing a unique user ID and password, no one else can use anyone else's user ID and password to access the system.
- User-friendly - Due to the system's ease of use, even newly registered users can use it without difficulty.
- Reliability - The system's mean time to failure will be very low, as will its reliability.
- Performance - Any number of users can access the system at the same time, and the system responds quickly to user requests.

- System Modification - Only the administrator can make system modifications, and updated adjustments will be applied as a real-time method, bringing the system up to date.

- Robustness - Due to real-time data savings, the number of events causing failure and the probability of data corruption on failure is very low, and even if it fails, the time to resume is relatively quick due to backup systems.

3.2 Data Requirements

1. Hotel
   - Hotel_ID (PK)
   - Name
   - Address
   - Contact_Number
   - Number_of_Rooms

2. Event
   - Event_ID (PK)
   - Name
   - Date

- Time
- Location
- Maximum_Capacity

3. Guest
   - Guest_ID (PK)
   - Name
   - Contact_Info
   - Email
   - Phone_Number

4. Feedback
   - Feedback_ID (PK)
   - Event_ID (FK)
   - Rating
   - Comments
   - Date

5. Payment
   - Payment_ID (PK)
   - Amount
   - Customer_ID (FK)
   - Transaction_Date

6. Reservation
   - Reservation_ID (PK)
   - Guest_ID (FK)
   - Room_ID (FK)
   - Checkin_Date
   - Checkout_Date

7. Room
   - Room_ID (PK)
   - Room_Type
   - Price
   - Hotel_ID (FK)

8. Employee
   - Employee_ID (PK)
   - Position_ID (FK)
   - Contact_Info
   - Employee_Salary
   - Employee_Name
   - Department_ID (FK)

9. Event_Planner

- Planner_ID (PK)
- Planner_Name
- Planner_Contact_Info
- Expertise

10. Invoice
    - Invoice_ID (PK)
    - Reservation_ID (FK)
    - Due_Date
    - Amount
    - Issued_Date

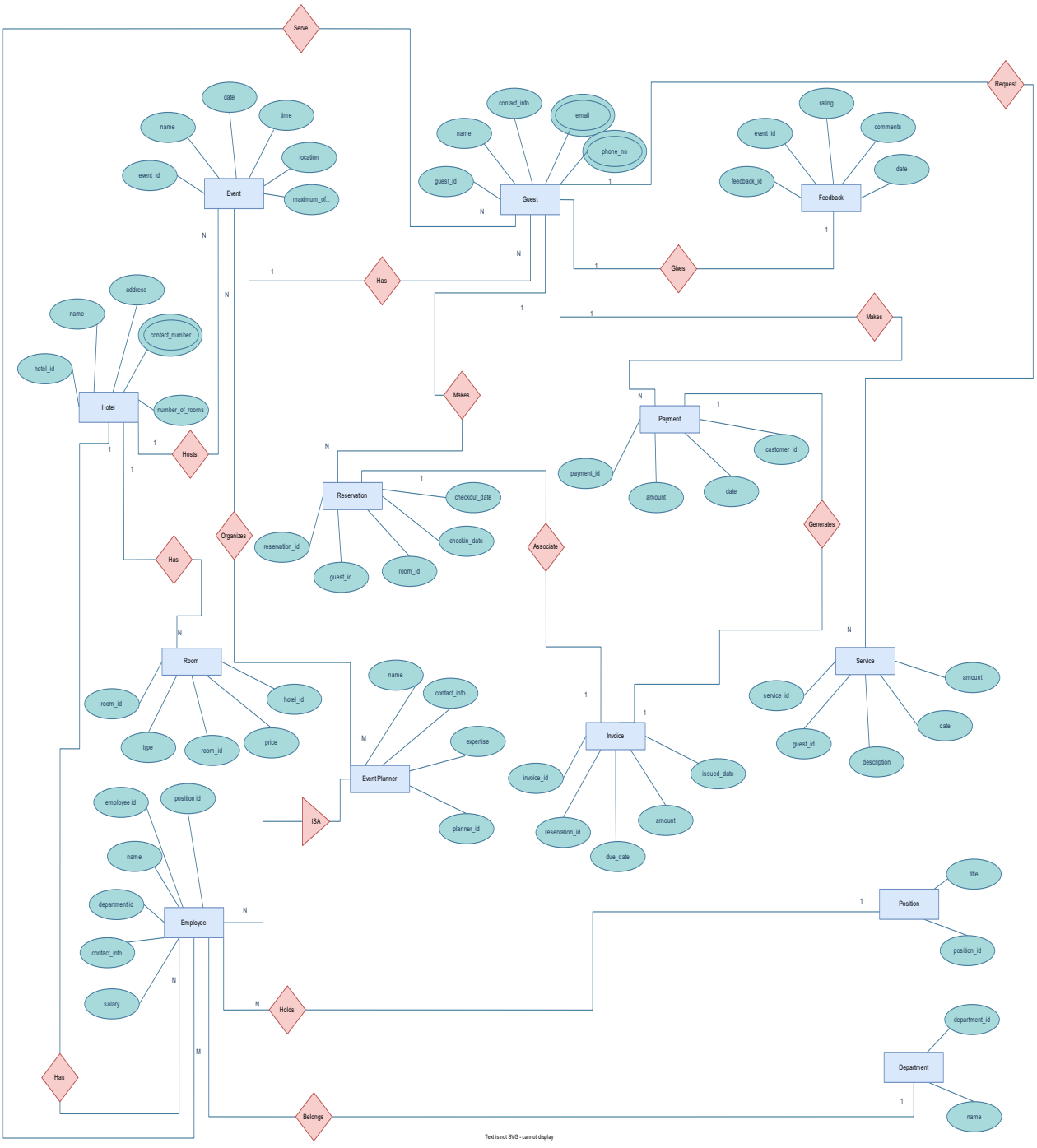
11. Service
    - Service_ID (PK)
    - Guest_ID (FK)
    - Description
    - Date
    - Amount

12. Position
    - Position_ID (PK)
    - Position_Title

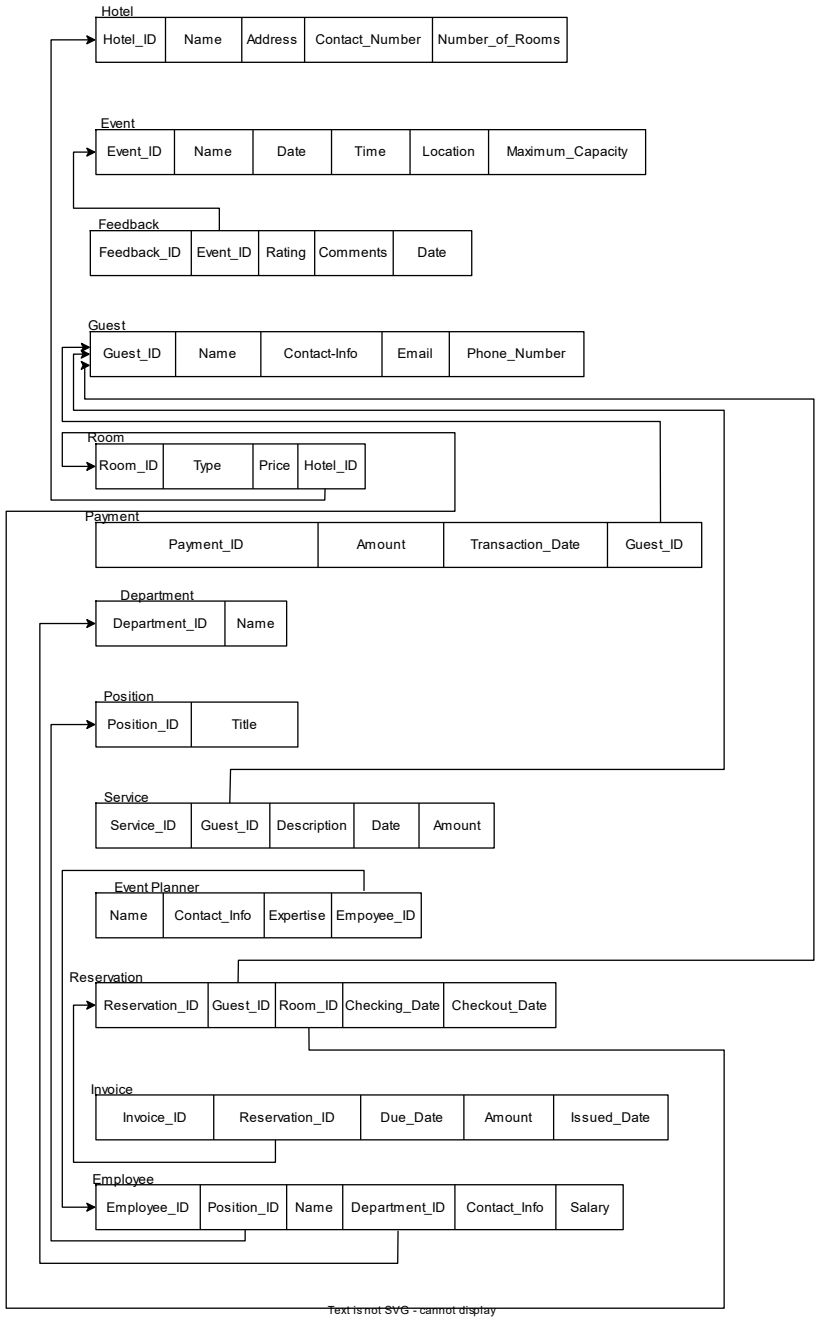
13. Department

- Department_ID (PK)
- Department_Name

4 Entity Relationship (ER Diagram)


Serve
Request
Event
date
time
name
location
event_id
maximum_of...
Guest
contact_info
email
name
phone_no
guest_id
Feedback
rating
comments
event_id
feedback_id
date
Has
Gives
Makes
Hotel
address
name
contact_number
hotel_id
number_of_rooms
Hosts
Makes
Payment
payment_id
amount
date
customer_id
Reservation
checkout_date
checkin_date
reservation_id
guest_id
room_id
Organizes
Associate
Generates
Has
Room
room_id
hotel_id
type
room_id
price
EventPlanner
name
contact_info
expertise
planner_id
ISA
Invoice
invoice_id
reservation_id
amount
due_date
issued_date
Service
service_id
guest_id
amount
date
description
employee id
position id
name
department id
contact_info
salary
Employee
Holds
Position
title
position_id
Has
Belongs
Department
department_id
name
Text is not SVG - cannot display

5 Relational Schema



6 SQL Queries

6.1 Data Base Create

```
CREATE DATABASE Hotel_reservation;

USE Hotel_reservation;

CREATE TABLE Hotel (
	Hotel_ID CHAR(6) NOT NULL,
	Name VARCHAR(50) NOT NULL,
	Address VARCHAR(50) NOT NULL,
	Number_of_Rooms INT NOT NULL,

	CONSTRAINT PK_Hotel_Hotel_ID PRIMARY KEY (Hotel_ID)
);


CREATE TABLE Event (
	Event_ID CHAR(6) NOT NULL,
	Name VARCHAR(50) NOT NULL,
	Date DATE NOT NULL,
	Time TIME NOT NULL,
	Location VARCHAR(50) NOT NULL,
	Maximum_Capacity INT NOT NULL,

	CONSTRAINT PK_Event_Event_ID PRIMARY KEY (Event_ID)
);


CREATE TABLE Guest (
	Guest_ID CHAR(6) NOT NULL,
	Name VARCHAR(50) NOT NULL,
	Contact_Info VARCHAR(100),

	CONSTRAINT PK_Guest_Guest_ID PRIMARY KEY (Guest_ID)
);
```

```
CREATE TABLE Feedback (
	Feedback_ID CHAR(6) NOT NULL,
	Event_ID CHAR(6) NOT NULL,
	Rating INT,
	Comments VARCHAR(250),
	Date DATE,

	CONSTRAINT PK_Feedback_Feedback_ID PRIMARY KEY (Feedback_ID),
	CONSTRAINT FK_Feedback_Event_ID FOREIGN KEY (Event_ID) REFERENCES Event(Event_ID)
);


CREATE TABLE Payment (
	Payment_ID CHAR(6) NOT NULL,
	Amount FLOAT NOT NULL,
	Guest_ID CHAR(6) NOT NULL,
	Transaction_Date DATE,

	CONSTRAINT PK_Payment_Payment_ID PRIMARY KEY (Payment_ID),
	CONSTRAINT FK_Payment_Guest_ID FOREIGN KEY (Guest_ID) REFERENCES Guest(Guest_ID)
);


CREATE TABLE Room (
	Room_ID CHAR(6) NOT NULL,
	Room_Type VARCHAR(50) NOT NULL,
	Price FLOAT NOT NULL,
	Hotel_ID CHAR(6),

	CONSTRAINT PK_Room_Room_ID PRIMARY KEY (Room_ID),
	CONSTRAINT FK_Room_Hotel_ID FOREIGN KEY (Hotel_ID) REFERENCES Hotel(Hotel_ID)
);


CREATE TABLE Reservation (
	Reservation_ID CHAR(6) NOT NULL,
	Guest_ID CHAR(6) NOT NULL,
	Room_ID CHAR(6) NOT NULL,
	Checkin_Date DATE NOT NULL,
	Checkout_Date DATE NOT NULL,

	CONSTRAINT PK_Reservation_Reservation_ID PRIMARY KEY (Reservation_ID),
	CONSTRAINT FK_Reservation_Guest_ID FOREIGN KEY (Guest_ID) REFERENCES
Guest(Guest_ID),
	CONSTRAINT FK_Reservation_Room_ID FOREIGN KEY (Room_ID) REFERENCES Room(Room_ID)
);
```

```
CREATE TABLE Position (
	Position_ID CHAR(6) NOT NULL,
	Position_Title VARCHAR(50) NOT NULL,

	CONSTRAINT PK_Position_Position_ID PRIMARY KEY (Position_ID)
);


CREATE TABLE Department (
	Department_ID CHAR(6) NOT NULL,
	Department_Name VARCHAR(50) NOT NULL,

	CONSTRAINT PK_Department_Department_ID PRIMARY KEY (Department_ID)
);

CREATE TABLE Employee (
	Employee_ID CHAR(6) NOT NULL,
	Position_ID CHAR(6) NOT NULL,
	Contact_Info VARCHAR(100),
	Employee_Salary FLOAT NOT NULL,
	Employee_Name VARCHAR(50) NOT NULL,
	Department_ID CHAR(6) NOT NULL,

	CONSTRAINT PK_Employee_Employee_ID PRIMARY KEY (Employee_ID),
	CONSTRAINT FK_Employee_Position_ID FOREIGN KEY (Position_ID) REFERENCES
Position(Position_ID),
	CONSTRAINT FK_Employee_Department_ID FOREIGN KEY (Department_ID) REFERENCES
Department(Department_ID)
);


CREATE TABLE Event_Planner (
	Planner_ID CHAR(6) NOT NULL,
	Planner_Name VARCHAR(50) NOT NULL,
	Planner_Contact_info VARCHAR(100),
	Expertise VARCHAR(100),

	CONSTRAINT PK_Event_Planner_Planner_ID PRIMARY KEY (Planner_ID),
);
```

```
CREATE TABLE Invoice (
	Invoice_ID CHAR(6) NOT NULL,
	Reservation_ID CHAR(6) NOT NULL,
	Due_Date DATE,
	Amount FLOAT NOT NULL,
	Issued_Date DATE,

	CONSTRAINT PK_Invoice_Invoice_ID PRIMARY KEY (Invoice_ID),
	CONSTRAINT FK_Invoice_Reservation_ID FOREIGN KEY (Reservation_ID) REFERENCES
Reservation(Reservation_ID)
);


CREATE TABLE Service (
	Service_ID CHAR(6) NOT NULL,
	Guest_ID CHAR(6) NOT NULL,
	Description VARCHAR(250),
	Date DATE,
	Amount FLOAT NOT NULL,

	CONSTRAINT PK_Service_Service_ID PRIMARY KEY (Service_ID),
	CONSTRAINT FK_Service_Guest_ID FOREIGN KEY (Guest_ID) REFERENCES Guest(Guest_ID)
);


CREATE TABLE Hotel_Contact_Number (
	Hotel_ID CHAR(6) NOT NULL,
	Mobile_Number CHAR(10) NOT NULL,

	CONSTRAINT PK_Hotel_Contact_Number PRIMARY KEY (Hotel_ID,Mobile_Number),
	CONSTRAINT FK_Hotel_Contact_Number FOREIGN KEY (Hotel_ID) REFERENCES
Hotel(Hotel_ID)
);


CREATE TABLE Guest_Email (
	Guest_ID CHAR(6) NOT NULL,
	Email VARCHAR(50) NOT NULL,

	CONSTRAINT PK_Guest_Email PRIMARY KEY (Guest_ID,Email),
	CONSTRAINT FK_Guest_Email FOREIGN KEY (Guest_ID) REFERENCES Guest(Guest_ID)
);
```

```
CREATE TABLE Guest_Phone_Number (
	Guest_ID CHAR(6) NOT NULL,
	Phone_Number CHAR(10) NOT NULL,

	CONSTRAINT PK_Guest_Phone_Number PRIMARY KEY (Guest_ID,Phone_Number),
	CONSTRAINT FK_Guest_Phone_Number FOREIGN KEY (Guest_ID) REFERENCES Guest(Guest_ID)
);


CREATE TABLE Oraganize (
	Event_ID CHAR(6) NOT NULL,
	Planner_ID CHAR(6) NOT NULL,

	CONSTRAINT PK_Oraganize PRIMARY KEY (Event_ID,Planner_ID),
	CONSTRAINT FK_Oraganize_Event_ID FOREIGN KEY (Event_ID) REFERENCES
Event(Event_ID),
	CONSTRAINT FK_Oraganizer_Planner_ID FOREIGN KEY (Planner_ID) REFERENCES
Event_Planner(Planner_ID)
);
```

6.2 Data Store in Data Base

```
INSERT INTO Hotel VALUES ('H10001','Grand Hotel','No10,Galle Roard,Wellawaththa',45);
INSERT INTO Hotel VALUES ('H10002','River Side Hotel','No125/1,kaduwela
road,Baththaramulla',30);
INSERT INTO Hotel VALUES ('H10003','Sea Side Hotel','No18,Galle Road,Panadura',38);
INSERT INTO Hotel VALUES ('H10004','Mount Hill Hotel','No131/2,Peradeniya,Kandy',60);
INSERT INTO Hotel VALUES ('H10005','Golden Lotus Resort','No108,Raja
Mawatha,Mathara',50);

INSERT INTO Event VALUES ('E10001','Sachini Birthday Party' ,'2022-11-
30','7PM','Rooftop,Grand Hotel',50);
INSERT INTO Event VALUES ('E10002','Sadaru and Kasuni Weeding','2022-12-
03','10AM','Conference Hall,River Side Hotel',400);
INSERT INTO Event VALUES ('E10003','Dineth and Kavya Anniversary','2023-01-
11','6PM','Grand Ballroom,Sea Side Hotel',40);
```

```
INSERT INTO Event VALUES ('E10004','ABC Company Annual Get Together Party','2023-01-
18','3PM','Garden Venue,Mount Hill Hotel',200);
INSERT INTO Event VALUES ('E10005','Isuru and Kaveesha Weeding','2023-02-
05','9PM','Beachfront,Golden Lotus Resort',100);

INSERT INTO Guest VALUES ('G10001','Tharindu','Home Address:No31/1,Jaya
Mawatha,Rajagiriya');
INSERT INTO Guest VALUES ('G10002','Sadaru','Work Address:No 234,Salmal uyana,Kottawa');
INSERT INTO Guest VALUES ('G10003','Sarath','Emergency
Contact:Dineth(Relashionship:Son,Phone no:0778695432');
INSERT INTO Guest VALUES ('G10004','Pasan','pasangunawardhna@gmail.com');
INSERT INTO Guest VALUES ('G10005','Isuru','isurumadushan@gmail.com');


INSERT INTO Feedback VALUES ('F10001','E10001',3,'"What a fantastic event!" You did an
excellent job coordinating everything','2022-12-03');
INSERT INTO Feedback VALUES ('F10002','E10002',4,'"During the event, I had a few issues
with the service." "It could be better"','2022-12-05');
INSERT INTO Feedback VALUES ('F10003','E10003',5,'"The food was delicious, and the
service was excellent." "Strongly recommended"','2023-01-12');
INSERT INTO Feedback VALUES ('F10004','E10004',5,'"The venue was beautiful, but there was
a delay in starting the event"','2023-01-19');
INSERT INTO Feedback VALUES ('F10005','E10005',4,'"I appreciate the attention to detail
and excellent customer service"','2023-02-07');

INSERT INTO Payment VALUES ('P10001','50000','G10001','2022-12-01');
INSERT INTO Payment VALUES ('P10002','1200000','G10002','2022-12-05');
INSERT INTO Payment VALUES ('P10003','80000','G10003','2023-01-13');
INSERT INTO Payment VALUES ('P10004','200000','G10004','2023-01-19');
INSERT INTO Payment VALUES ('P10005','250000','G10005','2023-02-08');


INSERT INTO Room VALUES ('R10001','Family Room',18000,'H10001');
INSERT INTO Room VALUES ('R10002','Dulex Room',12000,'H10002');
INSERT INTO Room VALUES ('R10003','Junior Suite Room',10000,'H10003');
INSERT INTO Room VALUES ('R10004','Standard Room',7000,'H10004');
INSERT INTO Room VALUES ('R10005','Luxury Suite Room',25000,'H10005');

INSERT INTO Reservation VALUES ('RE1001','G10001','R10001','2022-11-30','2022-12-01');
INSERT INTO Reservation VALUES ('RE1002','G10002','R10002','2022-12-03','2022-12-05');
INSERT INTO Reservation VALUES ('RE1003','G10003','R10003','2023-01-11','2023-01-12');
INSERT INTO Reservation VALUES ('RE1005','G10005','R10005','2023-02-05','2023-02-08');

INSERT INTO Position VALUES ('P10001','Event Coordinator');
INSERT INTO Position VALUES ('P10002','Sales Manager');
INSERT INTO Position VALUES ('P10003','Banquet Manager');
```

```
INSERT INTO Position VALUES ('P10004','Front Desk Supervisor');
INSERT INTO Position VALUES ('P10005','Marketing Specialist');

INSERT INTO Department VALUES ('D10001','Event Management Department');
INSERT INTO Department VALUES ('D10002','Business Development Department');
INSERT INTO Department VALUES ('D10003','Food and Beverage Department');
INSERT INTO Department VALUES ('D10004','Guest Services Department');
INSERT INTO Department VALUES ('D10005','Marketing and Communications Department');

INSERT INTO Employee VALUES
('EM1001','P10001','prabathmadushanka97@gmail.com',75000,'Prabath','D10001');
INSERT INTO Employee VALUES
('EM1002','P10002','Kasunpathirana@gmail.com',125000,'Kasun','D10002');
INSERT INTO Employee VALUES
('EM1003','P10003','supundilshan12@gmail.com',110000,'Supun','D10003');
INSERT INTO Employee VALUES
('EM1004','P10004','gayangunasekara@gmail.com',100000,'Gayan','D10004');
INSERT INTO Employee VALUES
('EM1005','P10005','chamindasampath@gmail.com',105000,'Chaminda','D10005');



INSERT INTO Event_Planner VALUES ('EP1001','Kusal','kusalarachchi@gmail.com','Wedding
planning');
INSERT INTO Event_Planner VALUES ('EP1002','Sadun','sadunperera@gmail.com','Corporate
events');
INSERT INTO Event_Planner VALUES ('EP1003','Lahiru','lahirulakshan25@gmail.com','Social
gatherings');
INSERT INTO Event_Planner VALUES
('EP1004','Udara','udarakodithuwakku@gmail.com','Conference planning');
INSERT INTO Event_Planner VALUES
('EP1005','Sampath','sampathdisanayake@gmail.com','Charity events');

INSERT INTO Invoice VALUES ('I10001','RE1001','2022-12-02',18000,'2022-12-01');
INSERT INTO Invoice VALUES ('I10002','RE1002','2022-12-07',18000,'2022-12-05');
INSERT INTO Invoice VALUES ('I10003','RE1003','2023-01-12',18000,'2023-01-12');
INSERT INTO Invoice VALUES ('I10005','RE1005','2023-02-10',18000,'2023-02-08');

INSERT INTO Service VALUES ('S10001','G10001','Dinner','2022-11-30',1500);
INSERT INTO Service VALUES ('S10002','G10002','Lunch and dinner','2022-12-04',5000);
INSERT INTO Service VALUES ('S10003','G10003','Two Coffe','2023-01-12',500);
INSERT INTO Service VALUES ('S10005','G10005','Set Menu','2023-02-06',9000);

INSERT INTO Hotel_Contact_Number VALUES ('H10001','0112345331');
INSERT INTO Hotel_Contact_Number VALUES ('H10002','0114564569');
INSERT INTO Hotel_Contact_Number VALUES ('H10003','0113323331');
```

```
INSERT INTO Hotel_Contact_Number VALUES ('H10004','0774567892');
INSERT INTO Hotel_Contact_Number VALUES ('H10005','0776611340');

INSERT INTO Guest_Email VALUES ('G10001','tharindubandara@gmail.com');
INSERT INTO Guest_Email VALUES ('G10002','sadarunawod98@gmail.com');
INSERT INTO Guest_Email VALUES ('G10003','sarathkumara68@gmail.com');
INSERT INTO Guest_Email VALUES ('G10004','pasantharaka@gmail.com');
INSERT INTO Guest_Email VALUES ('G10005','isurushanaka@gmail.com');

INSERT INTO Guest_Phone_Number VALUES ('G10001','0774469870');
INSERT INTO Guest_Phone_Number VALUES ('G10002','0753215548');
INSERT INTO Guest_Phone_Number VALUES ('G10003','0717624333');
INSERT INTO Guest_Phone_Number VALUES ('G10004','0779086542');
INSERT INTO Guest_Phone_Number VALUES ('G10005','0772206541');

INSERT INTO Oraganize VALUES ('E10001','EP1001');
INSERT INTO Oraganize VALUES ('E10002','EP1002');
INSERT INTO Oraganize VALUES ('E10003','EP1003');
INSERT INTO Oraganize VALUES ('E10004','EP1004');
INSERT INTO Oraganize VALUES ('E10005','EP1005');
```

7 Performance Requirement

Performance Requirements play a critical role in the system's success. They're as follows: Reservation System Performance Requirements for Hotel Phoenix:

1. Continuous System Availability: - The system must be operational 24 hours a day, 365 days a year, to ensure uninterrupted access for registered users.

2. Login procedure and Page Loading Time: - The login procedure should be as quick as possible, allowing registered users to submit their credentials and access the system in a matter of seconds.

- Loading pages in a timely manner offers a seamless and efficient user experience.

3. Speed and Usability: The system should be developed for maximum speed and usability, allowing for quick and responsive navigation across the website's features and functionalities.

4. Viewing Safari Details: Registered users should be able to quickly access specific information regarding safari packages, such as itinerary, pricing, inclusions, and availability.

5. Account Management: Registered users should be able to change or remove their account information, such as personal information, preferences, and notification settings.

6. Responsibilities of the Administrator: - The administrator should be able to add or remove shared experiences and feedback, authorize reservation requests, and manage reservation cancellations.

- The administrator should be in charge of user account management, which includes account creation, update, and deactivation.

- The administrator should be able to edit and update safari information, such as adding maps, photographs, videos, and package information.

7. Guest Donations: The system should enable a smooth donation process for guests, allowing them to give monies without the need for login credentials.

8. Tour Guide Communication: - Tour guides should be able to respond to user messages quickly, answering their questions, providing assistance, and sharing important information.

9. Development and Maintenance: - Developers should be able to create new functions and features to improve the functioning and user experience of the website.

- System upgrades and adjustments should be simple for developers to do in order to keep the system current and efficient.

- To maintain system stability and performance, developers should undertake bug identification and error correction on a regular basis.

10. User-Friendly Interface: - The system should have a user-friendly interface that allows for easy navigation, intuitive controls, and clear information presentation.

11. Device and Browser Compatibility: - Users should be able to access the website and system features from any device (desktop, laptop, tablet, mobile) or web browser at any time, ensuring optimal accessibility.

12. Staff Account Management: - The administrator should be able to manage staff accounts, including the creation of new accounts, the assignment of roles, the modification of access capabilities, and the deactivation of accounts as needed.

8 Security Requirements

Hotel Phoenix Reservation System Security Requirements:

1. Encryption of Personal User Details: - Before being saved in the database, all personal details of users, including names, contact information, and payment information, should be encrypted

using secure techniques. This maintains secrecy and prevents unauthorized access to critical information.

2. Access Control and Authorization: The system should incorporate stringent access control methods to prevent unauthorized users from accessing restricted features or sensitive data.

- It is necessary to set user roles and permissions, allowing suitable access levels based on job duties and authorized actions.

3. Data Backup and Recovery: All data stored in the system's database should be backed up on a regular basis. This ensures that data can be restored to its most recent condition in the case of data loss or system failure, reducing the risk of data loss or corruption.

4. Strong Password Requirements: - Strong password requirements for user accounts should be enforced, requiring the usage of a combination of uppercase letters, lowercase letters, numbers, and special characters. This helps to improve user account security and reduces the danger of unwanted access via brute-force or dictionary assaults.

5. One User Account per Email Address: Within the system, each email address should be associated with just one user account. This inhibits the creation of numerous user accounts with the same email address, maintaining the integrity of user identities and preventing account duplication.

6. Redundant Database Server: To provide high availability and data resilience, the database server should be setup with redundancy, using backup servers. This protects against server outages, hardware malfunctions, and other unanticipated events, reducing downtime and data loss threats.

7.Administrators have limited data access: Only authorized administrators should have access to the system's administrative activities and data. Strict access control measures should be in place to ensure that administrators can access and edit data within their defined area of duty while preventing unauthorized access or data manipulation.

8. Security Audits and Monitoring: - Regular security audits and monitoring should be performed to identify vulnerabilities, detect potential breaches, and maintain compliance with security best practices.

- Intrusion detection systems, log monitoring, and real-time notifications should be deployed to detect and respond to any security incidents as soon as possible.