

Information Systems and Data Modeling – IT1090



Assignment 02

Title: Wild-Life Safari Trip Management System

Batch Number: WE_01.01

Group Number:
MLB_WE_01.01_08

Declaration:

We hold a copy of this assignment that we can produce if the original is lost or damaged.

We hereby certify that no part of this assignment has been copied from any other group's work or from any other source. No part of this assignment has been written / produced for our group by another person except where such collaboration has been authorized by the subject lecturer/tutor concerned.

Group Members:

IT23224902	E.D.A. Silva signature
------------	--------------	--------------------

IT23209534	A.A.I. Nethmika signature
------------	-----------------	--------------------

IT23185920	S.S. Abishan signature
------------	--------------	--------------------

IT23142282	R.D.H. Madhusankha signature
------------	--------------------	--------------------

IT23139244	P.G.N.S Perera signature
------------	----------------	--------------------

Submitted on: <dd/mm/yyyy>

Table of Contents

1. Introduction	3
2. Hypothetical Scenario	4
3. Requirement Analysis	6
3.1. Main Requirements	6
3.1.1. Functional Requirements	6
3.1.2. Non-Functional Requirements	6
3.2. Data Requirements	7
4. Entity Relationship (ER Diagram)	8
5. Relational Schema	9
6. SQL Queries	10
6.1. Creating the Database	10
6.2. Storing Data in the Database	15
7. Performance Requirements	18

1. Introduction

The need of effective management systems are becoming rapidly important globally when it comes to modern travel and exploration, specifically for wildlife safari experiences. The “Safrik” Wildlife Safari Trip Management System is an essential tool that satisfies many needs of customers who wish to easily organize trips, make reservations for accommodation, research places and get guide assistance.

This platform builds an advanced system that facilitates user interactions and improves the skills of administrators and other stakeholders by combining module information systems and data modeling. Customers can search through various safari options, verify their availability, reserve tours, and customize their experience. Accommodation options are provided with all the information needed for people to make decisions based on their requirements and preferences. To maximize the overall safari experience, the system also provides detailed information on destinations, including wildlife sightings, geographical features and cultural insights. Users can also request guide assistance, offering tailored and informative tours. As an extension of the services, the users are able to immerse themselves in the breathtaking beauty of nature with the systems’ captivating gallery of high-quality images and videos.

The system provides business tools such as user management tools, accommodation and tourism tracking, tour management, data analytics and reporting tools. Third-party stakeholders, such as accommodation providers and tour guides can easily connect to the system to manage their services and communicate with the clients. The Wildlife Safari Trip Management System increases user experience, streamlines operations, and promotes efficient decision-making by utilizing data modeling methodologies. It acts as a lively center where nature and technology come together, providing enthusiasts from all over the world with captivating and life-changing wildlife experiences.

The Safrik Wild-Life Safari Trip Management System gives the opportunity to its users, regardless of the role, to achieve their respective tasks.

2. Hypothetical Scenario

The proposed system hence allows the users, administration, and the third-party users to manage their tasks efficiently and conveniently. The Safrik Wild-Life Safari Trip Management System gives the opportunity to its users, regardless of the role, to achieve their respective tasks.

If a new user who is planning to visit a safari destination visits the website, they have to register to the system by entering details, such as the full name, mobile number, email ID. Once registered they can use the created credentials and log in as a registered user. Upon successfully logging in, they can use the system to browse various safari options available and select a suitable package according to their requirements. The packages include accommodation in a lodge or a camp. Once the user selects a package of their interest, they will be required to fill in a booking form with personal, contact & trip details. Additionally, if the user requires a guide, they can also select a well experienced tourist guide from the system. Afterwards, the user can select a payment option: online or offline and submit the card details/payment receipt accordingly. The user receives an email with an OTP confirming the payment, which is required to complete the booking process. Also, the system sends another email to the user confirming the booking and details of the upcoming trip.

The system has a gallery feature which allows users such as travel bloggers, research scientists and other individuals who are seeking to explore the wonders of the wild, to browse and download quality media content. Once they navigate to the gallery the users can use the designed search filters to enhance the search and select the media items for download. If the selected number of contents are less than 10, the user can go ahead with the download and if the number of contents exceeds 10, the user will be navigated to the register page.

The system offers an admin dashboard for the administrators, which is designed using advanced tools to keep track of the insights and statistics in detail. This allows the admins to monitor user activities, analyze the booking trends and take data-driven decisions. A data analyst in the system can log in to the system using the admin credentials and reply to the received feedback in the feedback management section. Once the response is submitted, the system automatically updates the status of the feedback to 'Replied'. Also, he can identify any recurring issues by the ratings and the overall satisfaction levels of the users.

The financial manager, who is also an admin, logs in to the system using his admin credentials to oversee the financial transactions and booking made by the users. Once a booking is selected from the list, the system provides details about the customer, the booking and the relevant payment slip to the admin, where they can cross-check the payment with the bank and validate the transaction made. Upon validity, the 'Valid Payment' option is chosen and the system automatically generates a detailed receipt, which the Financial Manager sends to the customer. If the payment is confirmed negative by the bank, the admin chooses the 'Invalid Payment' options and contacts the customer.

A tourist guide logs in to the system using admin credentials to access the booking schedule. This provides detailed information on scheduled trips and accompanying tourist. The system also displays a comprehensive booking schedule, which indicates the trip details such as safari destinations, safari duration, group size and the tourist profile. The guide can review the bookings and assess the nature of each trip based on the tourist preferences and the activities planned. After considering all the above, the guide proceeds with 'confirm booking' within the system, ensuring the aspects of the trip are in order. If two bookings have been made on the same day, the system allows the tourist guide to either contact the customer and 'reschedule' or 'cancel booking.'

3. Requirement Analysis

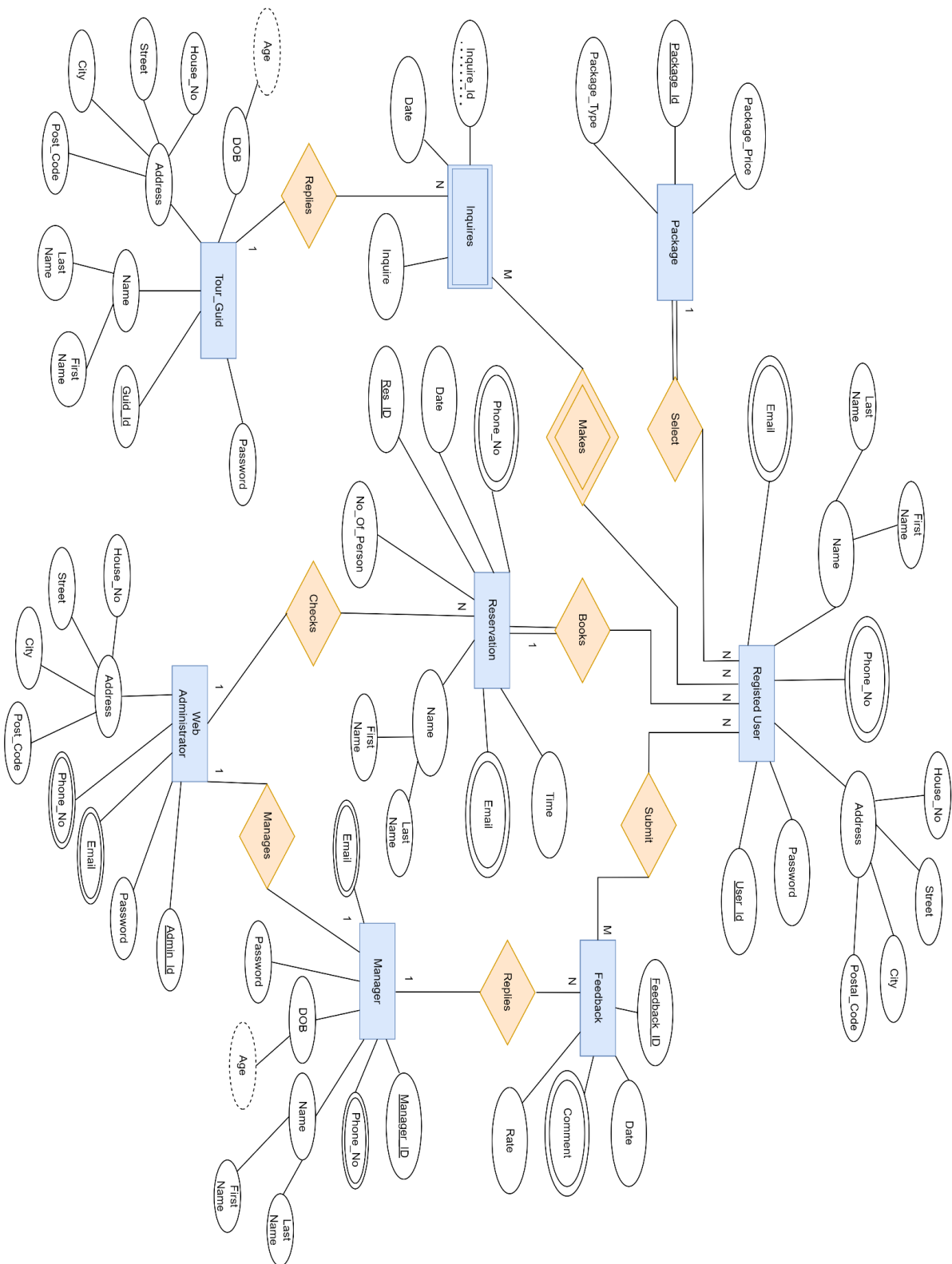
3.1. Main Requirements

3.1.1. Functional Requirements

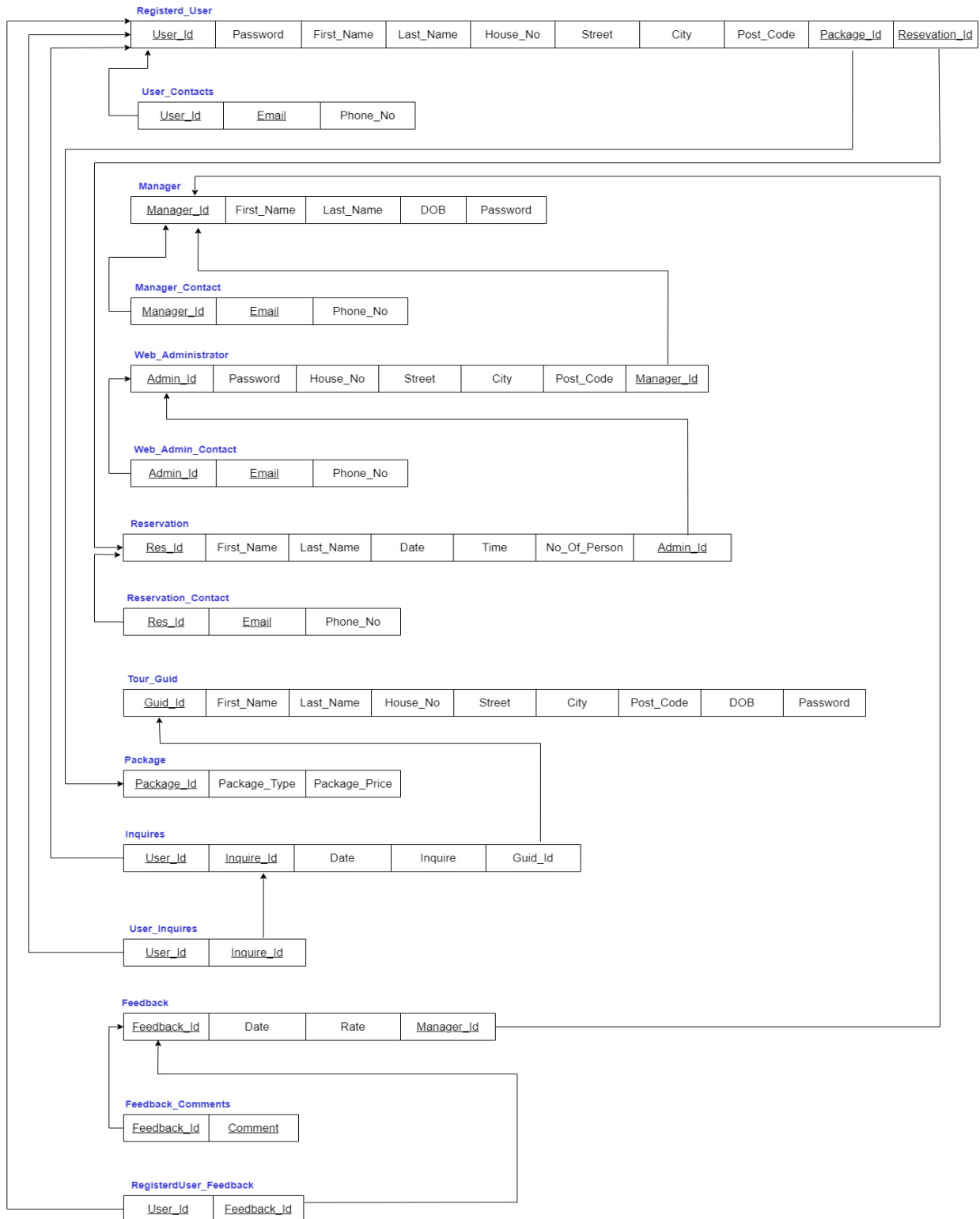
3.1.2. Non-Functional Requirements

3.2. Data Requirements

4. Entity Relationship (ER Diagram)



5. Relational Schema



6. SQL Queries

6.1. Creating the Database

```
/* Create Manager Table */
CREATE TABLE Manager
(
    manager_id int NOT NULL,
    first_name varchar(20) NOT NULL,
    last_name varchar(20) NOT NULL,
    dob DATE NOT NULL,
    password varchar(10) NOT NULL,

    CONSTRAINT manager_PK PRIMARY KEY (manager_id),
    CONSTRAINT manager_password_check CHECK(
        LEN(password) BETWEEN 6 AND 10
        AND password LIKE '%[A-Z]%'
        AND password LIKE '%[0-9]%'
    )
);

/* Create Web Admin Table */
CREATE TABLE Web_Admin
(
    admin_id int NOT NULL,
    password varchar(10) NOT NULL,
    house_no varchar(15) NOT NULL,
    street varchar(30) NOT NULL,
    city varchar (30) NOT NULL,
    post_code varchar (20) NOT NULL,
    manager_id int NOT NULL,

    CONSTRAINT admin_PK PRIMARY KEY(admin_id),
    CONSTRAINT admin_FK FOREIGN KEY(manager_id) REFERENCES Manager(manager_id),
    CONSTRAINT admin_password_check CHECK(
        LEN(password) BETWEEN 6 AND 10
        AND password LIKE '%[A-Z]%'
        AND password LIKE '%[0-9]%'
    )
);

/* Create Reservation Table */
CREATE TABLE Reservation
(
    res_id int NOT NULL,
    first_name varchar(30) NOT NULL,
    last_name varchar (30) NOT NULL,
    date DATE NOT NULL,
    time TIME NOT NULL,
    no_of_person int NOT NULL,
    admin_id int NOT NULL,

    CONSTRAINT reservation_PK PRIMARY KEY(res_id),
    CONSTRAINT reservation_FK FOREIGN KEY (admin_id) REFERENCES Web_Admin(admin_id),
);
```

```

/* Create Tour Guid Table */
CREATE TABLE Tour_Guid
(
    guid_id int NOT NULL,
    first_name varchar(20) NOT NULL,
    last_name varchar(20) NOT NULL,
    house_no varchar(15) NOT NULL,
    street varchar(30) NOT NULL,
    city varchar(30) NOT NULL,
    post_code varchar(20) NOT NULL,
    dob DATE NOT NULL,
    password varchar(10) NOT NULL,

    CONSTRAINT tour_guid_PK PRIMARY KEY(guid_id),
    CONSTRAINT tour_guid_password_check CHECK(
        LEN(password) BETWEEN 6 AND 10
        AND password LIKE '%[A-Z]%'
        AND password LIKE '%[0-9]%'
    )
);

/* Create Package Table */
CREATE TABLE Package
(
    package_id int NOT NULL,
    package_type varchar(20) NOT NULL,
    package_price real NOT NULL,

    CONSTRAINT package_PK PRIMARY KEY(package_id)
);

/* Create Feedback Table */
CREATE TABLE Feedback
(
    feedback_id int NOT NULL,
    date DATE NOT NULL,
    rate varchar(20) NOT NULL,
    manager_id int NOT NULL,

    CONSTRAINT feedback_PK PRIMARY KEY (feedback_id),
    CONSTRAINT feedback_FK FOREIGN KEY (manager_id) REFERENCES Manager(manager_id)
);

```

```

/* Create Registerd User Table */
CREATE TABLE Registerd_user
(
    user_id int NOT NULL,
    password varchar(20) NOT NULL,
    first_name varchar(30) NOT NULL,
    last_name varchar(30) NOT NULL,
    house_no varchar(15) NOT NULL,
    street varchar(30) NOT NULL,
    city varchar (30) NOT NULL,
    post_code varchar (20) NOT NULL,
    package_id int NOT NULL,
    res_id int NOT NULL,

    CONSTRAINT user_PK PRIMARY KEY(user_id),
    CONSTRAINT user_FK1 FOREIGN KEY (package_id) REFERENCES Package(package_id),
    CONSTRAINT user_FK2 FOREIGN KEY (res_id) REFERENCES Reservation(res_id),
    CONSTRAINT user_password_check CHECK(
        LEN(password) BETWEEN 6 AND 20
        AND password LIKE '[A-Z]%'
        AND password LIKE '[0-9]%'
    )
);

/* Create Inquires Table */
CREATE TABLE Inquire
(
    inquire_id int NOT NULL,
    date DATE NOT NULL,
    inquire varchar(800) NOT NULL,
    user_id int NOT NULL,
    guid_id int NOT NULL,

    CONSTRAINT inquire_PK PRIMARY KEY(inquire_id,user_id),
    CONSTRAINT inquire_FK1 FOREIGN KEY (user_id) REFERENCES Registerd_user(user_id),
    CONSTRAINT inquire_FK2 FOREIGN KEY (guid_id) REFERENCES Tour_Guid(guid_id),
);

/* Create User Contacts Table */
CREATE TABLE User_Contacts
(
    user_id int NOT NULL,
    email varchar(30) NOT NULL,
    phone_no varchar(10) NOT NULL,

    CONSTRAINT user_contact_PK PRIMARY KEY (user_id,email),
    CONSTRAINT user_contact_FK FOREIGN KEY (user_id) REFERENCES
Registerd_User(user_id),
    CONSTRAINT check_uc_email CHECK(email LIKE '%@%._%'),
    CONSTRAINT check_uc_phone CHECK (phone_no BETWEEN 0000000000 AND 9999999999)
);

```

```

/* Create Manager Contacts Table */
CREATE TABLE Manager_Contact
(
    manager_id int NOT NULL,
    email varchar(30) NOT NULL,
    phone_no varchar(10) NOT NULL,

    CONSTRAINT manager_contact_PK PRIMARY KEY (manager_id,email),
    CONSTRAINT manager_contact_FK FOREIGN KEY (manager_id) REFERENCES
Manager(manager_id),
    CONSTRAINT check_mc_email CHECK(email LIKE '%@%._%'),
    CONSTRAINT check_mc_phone CHECK (phone_no BETWEEN 0000000000 AND 9999999999)
);

/* Create Web Admin Contacts Table */

CREATE TABLE Web_Admin_Contact
(
    admin_id int NOT NULL,
    email varchar(30) NOT NULL,
    phone_no varchar(10) NOT NULL,

    CONSTRAINT admin_contact_PK PRIMARY KEY (admin_id,email),
    CONSTRAINT admin_contact_FK FOREIGN KEY (admin_id) REFERENCES Web_Admin(admin_id),
    CONSTRAINT check_ac_email CHECK(email LIKE '%@%._%'),
    CONSTRAINT check_ac_phone CHECK (phone_no BETWEEN 0000000000 AND 9999999999)
);

/* Create Reservation Contacts Table */
CREATE TABLE Reservation_Contact
(
    res_id int NOT NULL,
    email varchar(30) NOT NULL,
    phone_no varchar(10) NOT NULL,

    CONSTRAINT Reservation_contact_PK PRIMARY KEY (res_id,email),
    CONSTRAINT Reservation_contact_FK FOREIGN KEY (res_id) REFERENCES
Reservation(res_id),
    CONSTRAINT check_rc_email CHECK(email LIKE '%@%._%'),
    CONSTRAINT check_rc_phone CHECK (phone_no BETWEEN 0000000000 AND 9999999999)
);

/* Create User Inquires Table */
CREATE TABLE User_Inquire
(
    user_id INT NOT NULL,
    inquire_id INT NOT NULL,

    CONSTRAINT user_inquire_PK PRIMARY KEY (user_id, inquire_id),
    CONSTRAINT user_inquir_FK1 FOREIGN KEY (user_id) REFERENCES Registerd_User(user_id),
    CONSTRAINT user_inquir_FK2 FOREIGN KEY (inquire_id,user_id) REFERENCES
Inquire(inquire_id,user_id),
);

```

```

/* Create Feedback Comment Table */
CREATE TABLE Feedback_Comment
(
    feedback_id int NOT NULL,
    comment varchar(800) NOT NULL,

    CONSTRAINT feedback_Comment_PK PRIMARY KEY(feedback_id,comment),
    CONSTRAINT feedback_Comment_FK FOREIGN KEY (feedback_id) REFERENCES
Feedback(feedback_id)
);

/* Create Register User Feedback Table */
CREATE TABLE Registerd_User_Feedback
(
    user_id int NOT NULL,
    feedback_id int NOT NULL,

    CONSTRAINT Registerd_User_Feedback_PK PRIMARY KEY(user_id,feedback_id),
    CONSTRAINT Registerd_User_Feedback_FK1 FOREIGN KEY (user_id) REFERENCES
Registerd_user(user_id),
    CONSTRAINT Registerd_User_Feedback_FK2 FOREIGN KEY (feedback_id) REFERENCES
Feedback(feedback_id)
);

```

6.2. Storing Data in the Database

```
/* Inset Data into Manager Table */

INSERT INTO Manager VALUES(001, 'Nimal', 'Karunarathne', '1973-5-8', 'M11234');
INSERT INTO Manager VALUES(002, 'Kamal', 'Abesinghe', '1970-2-1', 'M21234');
INSERT INTO Manager VALUES(003, 'Amal', 'Gunarathne', '1965-5-20', 'M31234');
INSERT INTO Manager VALUES(004, 'Piyal', 'Nishantha', '1971-4-8', 'M41234');
INSERT INTO Manager VALUES(005, 'Nuwan', 'Perera', '1971-1-15', 'M51234');

/* Inset Data into Web Admin Table */

INSERT INTO Web_Admin VALUES (001, 'A11234', 'A1', '1st Lane', 'Kiribathgoda', '32100', 001);
INSERT INTO Web_Admin VALUES (002, 'A21234', 'A2', '2nd Lane', 'Nugegoda', '33100', 002);
INSERT INTO Web_Admin VALUES (003, 'A31234', 'A3', '3rd Lane', 'Malabe', '35100', 003);
INSERT INTO Web_Admin VALUES (004, 'A41234', 'A4', '4th Lane', 'Nittabuwa', '36100', 004);
INSERT INTO Web_Admin VALUES (005, 'A51234', 'A5', '5th Lane', 'Kelaniya', '37100', 005);

/* Inset Data into Reservation Table */

INSERT INTO Reservation VALUES (001, 'Chanupa', 'Athsara', '2023-12-02', '12:32:00', 5, 001);
INSERT INTO Reservation VALUES (002, 'Amali', 'Senarathne', '2024-02-10', '14:12:00', 10, 002);
INSERT INTO Reservation VALUES (003, 'Ruwan', 'Harshana', '2024-03-15', '06:25:00', 2, 002);
INSERT INTO Reservation VALUES (004, 'Chamali', 'Jayawardhane', '2024-03-20', '15:30:00', 4, 003);
INSERT INTO Reservation VALUES (005, 'Nipun', 'Suranga', '2024-04-02', '17:42:00', 15, 005);

/* Inset Data into Tour Guid Table */

INSERT INTO Tour_Guid VALUES (001, 'Pasindu', 'Sathsara', 'B1', '1st Lane', 'Kurunegala', '71200', '1991-05-10', 'TG11234');
INSERT INTO Tour_Guid VALUES (002, 'Keshana', 'Kavindu', 'B2', '2nd Lane', 'Kegalle', '41300', '1991-05-10', 'TG21234');
INSERT INTO Tour_Guid VALUES (003, 'Radeesh', 'Nisal', 'B3', '3rd Lane', 'Kandy', '85400', '1991-05-10', 'TG31234');
INSERT INTO Tour_Guid VALUES (004, 'Sithara', 'Wijesinghe', 'B4', '4th Lane', 'Hatton', '36400', '1991-05-10', 'TG41234');
INSERT INTO Tour_Guid VALUES (005, 'Nuwani', 'Sashikala', 'B5', '5th Lane', 'Mathara', '12400', '1991-05-10', 'TG51234');

/* Inset Data into Package Table */

INSERT INTO Package VALUES (001, 'BRONZE PACKAGE', 2500.00);
INSERT INTO Package VALUES (002, 'SILVER PACKAGE', 3500.00);
INSERT INTO Package VALUES (003, 'GOLD PACKAGE', 5000.00);
INSERT INTO Package VALUES (004, 'CHILD PACKAGE', 1000.00);
INSERT INTO Package VALUES (005, 'SPECIAL PACKAGE', 7500.00);

/* Inset Data into Feedback Table */

INSERT INTO Feedback VALUES (001, '2023-05-10', 'Poor', 005);
INSERT INTO Feedback VALUES (002, '2023-07-20', 'Good', 004);
INSERT INTO Feedback VALUES (003, '2023-08-11', 'Good', 003);
INSERT INTO Feedback VALUES (004, '2023-10-15', 'Satisfide', 002);
INSERT INTO Feedback VALUES (005, '2024-01-13', 'Exelent', 002);
```

```

/* Inset Data into Registerd User Table */

INSERT INTO Registerd_User VALUES (001, 'Nuwan1234', 'Nuwan', 'Sagara', 'C1', '1st
Lane', 'Galle', '41300', 005, 001);
INSERT INTO Registerd_User VALUES (002, 'Pasindu1234', 'Pasindu', 'Randula', 'C2', '2nd
Lane', 'Ruwanwella', '25300', 004, 002);
INSERT INTO Registerd_User VALUES (003, 'Akash1234', 'Akash', 'Viduranga', 'C3', '3rd
Lane', 'Gampaha', '423600', 003, 003);
INSERT INTO Registerd_User VALUES (004, 'Chanuka1234', 'Chanuka', 'Gayan', 'C4', '4th
Lane', 'Dehiwala', '30300', 002, 004);
INSERT INTO Registerd_User VALUES (005, 'Piyumi1234', 'Piyumi', 'Maleesha', 'C5', '4th
Lane', 'Anuradhapura', '50100', 001, 005);

/* Inset Data into Inquires Table */

INSERT INTO Inquire VALUES (001, '2023-05-05', 'What are the top five wildlife species that
visitors are most likely to encounter during a safari in our designated area?', 001, 005);
INSERT INTO Inquire VALUES (002, '2023-06-13', 'How does our safari tour contribute to
wildlife conservation efforts in the region?', 002, 004);
INSERT INTO Inquire VALUES (003, '2023-08-02', 'What measures are in place to ensure the
safety and well-being of both wildlife and visitors during safari excursions?', 003, 003);
INSERT INTO Inquire VALUES (004, '2023-11-01', 'Can visitors expect to witness any seasonal
migrations or unique behaviors of wildlife during specific times of the year?', 004, 002);
INSERT INTO Inquire VALUES (005, '2023-12-25', 'How does our safari experience cater to
both seasoned wildlife enthusiasts and those who are new to safari adventures?', 005, 001);

/* Inset Data into User Contacts Table */

INSERT INTO User_Contacts VALUES (001, 'Nuwan@gmail.com', '0712345698');
INSERT INTO User_Contacts VALUES (002, 'Pasindu@gmail.com', '0765254569');
INSERT INTO User_Contacts VALUES (003, 'Akash@gmail.com', '0745858552');
INSERT INTO User_Contacts VALUES (004, 'Chanuka@gmail.com', '0712526321');
INSERT INTO User_Contacts VALUES (005, 'Piyumi@gmail.com', '0773139865');

/* Inset Data into Manager Contacts Table */

INSERT INTO Manager_Contact VALUES (001, 'M1@gmail.com', '0714525365');
INSERT INTO Manager_Contact VALUES (002, 'M2@gmail.com', '0774541236');
INSERT INTO Manager_Contact VALUES (003, 'M3@gmail.com', '0711489632');
INSERT INTO Manager_Contact VALUES (004, 'M4@gmail.com', '0765459365');
INSERT INTO Manager_Contact VALUES (005, 'M5@gmail.com', '0771489741');

/* Inset Data into Web Admin Contacts Table */

INSERT INTO Web_Admin_Contact VALUES (001, 'A1@gmail.com', '0712369852');
INSERT INTO Web_Admin_Contact VALUES (002, 'A2@gmail.com', '0774547135');
INSERT INTO Web_Admin_Contact VALUES (003, 'A3@gmail.com', '0712525220');
INSERT INTO Web_Admin_Contact VALUES (004, 'A4@gmail.com', '0774523698');
INSERT INTO Web_Admin_Contact VALUES (005, 'A5@gmail.com', '0782469321');

/* Inset Data into Reservation Contacts Table */

INSERT INTO Reservation_Contact VALUES (001, 'Chanupa@gmail.com', '0712489632');
INSERT INTO Reservation_Contact VALUES (002, 'Amali@gmail.com', '0774541256');
INSERT INTO Reservation_Contact VALUES (003, 'Ruwan@gmail.com', '0785247896');
INSERT INTO Reservation_Contact VALUES (004, 'Chamali@gmail.com', '0711214156');
INSERT INTO Reservation_Contact VALUES (005, 'Nipun@gmail.com', '0762541789');

```



```
/* Inset Data into User Inquires Table */
```

```
INSERT INTO User_Inquire VALUES (001,001);  
INSERT INTO User_Inquire VALUES (002,002);  
INSERT INTO User_Inquire VALUES (003,003);  
INSERT INTO User_Inquire VALUES (004,004);  
INSERT INTO User_Inquire VALUES (005,005);
```

```
/* Inset Data into Feedback Comment Table */
```

```
INSERT INTO Feedback_Comment VALUES(001,'Absolutely breathtaking! Seeing the majestic  
lions up close was an experience I will never forget. The guides were knowledgeable and  
respectful of the animals space.');
```

```
INSERT INTO Feedback_Comment VALUES(002,'A safari like no other! From spotting elusive  
leopards to witnessing a herd of elephants bathing in the river, every moment was filled  
with wonder and excitement. Highly recommend!');
```

```
INSERT INTO Feedback_Comment VALUES(003,'I was blown away by the diversity of wildlife on  
this safari. From the tiny dung beetles to the massive rhinos, every creature played a  
vital role in the ecosystem. Truly a humbling experience.');
```

```
INSERT INTO Feedback_Comment VALUES(004,'The conservation efforts of this safari company  
are commendable. Its clear they prioritize the well-being of the animals and their  
habitats above all else. Proud to support such a responsible tourism initiative.');
```

```
INSERT INTO Feedback_Comment VALUES(005,'As a first-time safari-goer, I couldnt have  
asked for a better introduction to the wonders of the African wilderness. The guides were  
patient and passionate, and I left with a newfound appreciation for nature.');
```

```
/* Inset Data into Register User Feedback Table */
```

```
INSERT INTO Registerd_User_Feedback VALUES(001,005);  
INSERT INTO Registerd_User_Feedback VALUES(002,004);  
INSERT INTO Registerd_User_Feedback VALUES(003,003);  
INSERT INTO Registerd_User_Feedback VALUES(004,002);  
INSERT INTO Registerd_User_Feedback VALUES(005,001);
```

7. Performance Requirements