Information Systems and Data Modeling – IT1090

Assignment

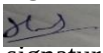
Title: Car Rental System	
Batch Number: Y1S2CSNE1.1	Group Number: MLB_WD_CSNE_01.01_12

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:

IT22253330	Rashminda Iddamalgoda	 signature
IT22339188	R.M.D.E.Rajapaksha	 signature
IT22911858	S.A.H.D.M.Perera	 signature
IT22312594	D.K.D.G.Perera	 signature
IT22073082	E.M.B.Vihanga Bandara	 signature

Submitted on: 28/05/2023

Table of Contents

INTRODUCTION

When analyzing car rental systems, we analyzed how the Internet links consumers and businesses as well as how a website may increase the usability of a service for customers. As a result, we created a website where consumers can explore and reserve car rental packages, providing a solution for the car rental business. Customers can register on our website and use it to suggest improvements, ask questions about our services, and receive customer service. To create a website for transport management, a rental system of technologies must be investigated and comprehended.

A variety of technologies must be explored and understood to develop a car rental system website. They included relational databases (like My SQL) and programming languages (including HTML, CSS, and JavaScript).

A database needs to be able to handle a variety of data types. Because managing a computer-based stored data system is simpler than managing written or typed materials, a database is necessary. This saves time and effort for users and admins. Furthermore, there is no chance of losing the data because it is kept in a database. The only people who have power over the system are its users. When creating a database, it is crucial to gather requirements and do a requirement analysis. The data requirements and the functional and nonfunctional specifications can then be identified.

HYPOTHETICAL SCENARIO

The Car Rental System is going to allow customers to make reservations for their cars from anywhere in the world. Customers fill out their personal details to submit details to this application. A customer can reserve a car after having created an account on the website. The one that has been suggested is a fully integrated online system. It streamlines and streamlines manual processes. Customers are helped by this automated process, which enables them to fill in the details as needed. It includes details on the location and the kind of car they want to rent. The system's objective is to develop a website where users can purchase cars and make service requests from anywhere in the world.

Requirement Analysis

Main Requirements

Functional Requirements

The main Functions of the website and Events that take place between the users and the system are described by the Functional requirements. Four users are using this car rental system. Namely: the registered customer, owner, admin, and the driver. They can access this system in different ways where it is related to them.

Registered customer

User requirements:

- Registered customers can check promotions available in packages.
- Registered customer views the available Feedback.
- Customers can get register to the system by providing the required details for the registration.
- Customers can check availabilities for cars.
- Customers can check packages on the website.
- Registered customer login to the system using by providing required customer login credentials.
- Registered customers can do bookings for a car.
- Registered customer can edit their own Account details.
- Registered customers can cancel bookings.
- Registered customers can Contact drivers and admin on the Contact Us page of the website.
- Registered customers can get their transaction details.

System requirements:

- The system should validate the credential entered by the customer.
- The system should display available promotions about packages and shared experiences and feedback uploaded by the customers on the website.
- The system should approve registration details and create a user account.

- The system should display the availabilities of bookings on the website.
- The system should display packages with facilities on the website so that customers can check packages.
- The system should validate the login credentials entered by the registered customer.
- The system should display/ send the booking Request to the Admin and store details.
- The system should provide the ability to upload feedback and experiences.
- The system Should send feedback uploading requests to the admin for authorization and store the details/data.
- The system should send the booking cancellation request to the admin and store details.
- The system should store/ save modified customer account details.

Driver

User Requirement:

- The driver can log in to the system by entering login credentials.
- The driver replies to the messages sent by the customer and admin.

System requirement:

- The system should validate the login credentials entered by the driver.
- The system displays the messages query list to the driver.
- The system allows for the coordination between the driver and the customers.

Admin

User requirements:

- Admin signs into the website by providing the required login credentials.
- Admin can approve the bookings.
- Admin can add and remove staff accounts.
- Admin can activate and deactivate customer accounts.
- Admin can approve canceled bookings and remove marked bookings from the calendar and arrange the refund.
- Admin can update location details. (map, about destination)
- Admin can add and remove share experiences.

- Admin checks member feedback, contacts, and reviews.
- Admin can log in to the system by entering login credentials.
- Admin can generate reports about the transaction.
- Admin can manage salaries of drivers.

System requirements:

- The system should validate the user login credentials.
- The system should delete details of the deleted location details by the user.
- The system should delete details of the drivers deleted by the user.
- The system should update the details of the downgraded users and modified accounts in the database.
- The system stores the customer feedback, contacts, and reviews and displays them.
- The system should approve canceled bookings and remove marked bookings from the calendar and refund the user.
- The system should validate login Credentials entered by the admin.
- The system should generate Reports such as transaction reports.

Owner

User requirements:

- The owner can get their payments at the end of the month.
- The owner can cancel the agreement of the car.

System requirements:

- The system can generate a report of the payment at the end of the month.
- The system can display, update and delete the details of the owner.

Non-Functional Requirements

We simply refer to quality qualities as non-functional needs. It describes system traits that aren't specifically related to a given functionality. Criticality levels for functional and non-functional needs may differ. The system can be worthless if these are not realized.

Speed

- The system needs to operate quickly.
- More people can log in simultaneously without experiencing any problems with the system.

Availability

- 24/7 accessibility is required for the system.

User friendly

- Users with less IT knowledge ought to be able to use the system.

Reliability

- The system must be able to recognize incorrect user credentials.

Security

- The system should be able to protect user data and prevent unwanted access, misuse, and fakes
- Also, by providing a unique user ID and password, no one can access the system by using any other's user ID and password.

Scalability

- The system should be able to handle a higher workload on demand.

Performance

- Admin can add, edit, remove, and update properties
- Any number of users can be able to access the system at the same time and the response of the system regarding to the user requests will be very high.

Data Requirement

Admin

- Admin_ID
- Name
- Address
- Gender
- DOB
- Email
- Phone_Num

Customer

- Cust_ID
- Name
- Address
- Gender
- DOB
- Phone_Num
- Email

Driver

- Driver_ID
- Name
- NIC
- Gender
- Commission
- Phone_Num
- Email

Rental

- Rental_ID
- Rerurn_Date
- Rental_Date
- Start_Time
- End_Time
- Destination
- Payment

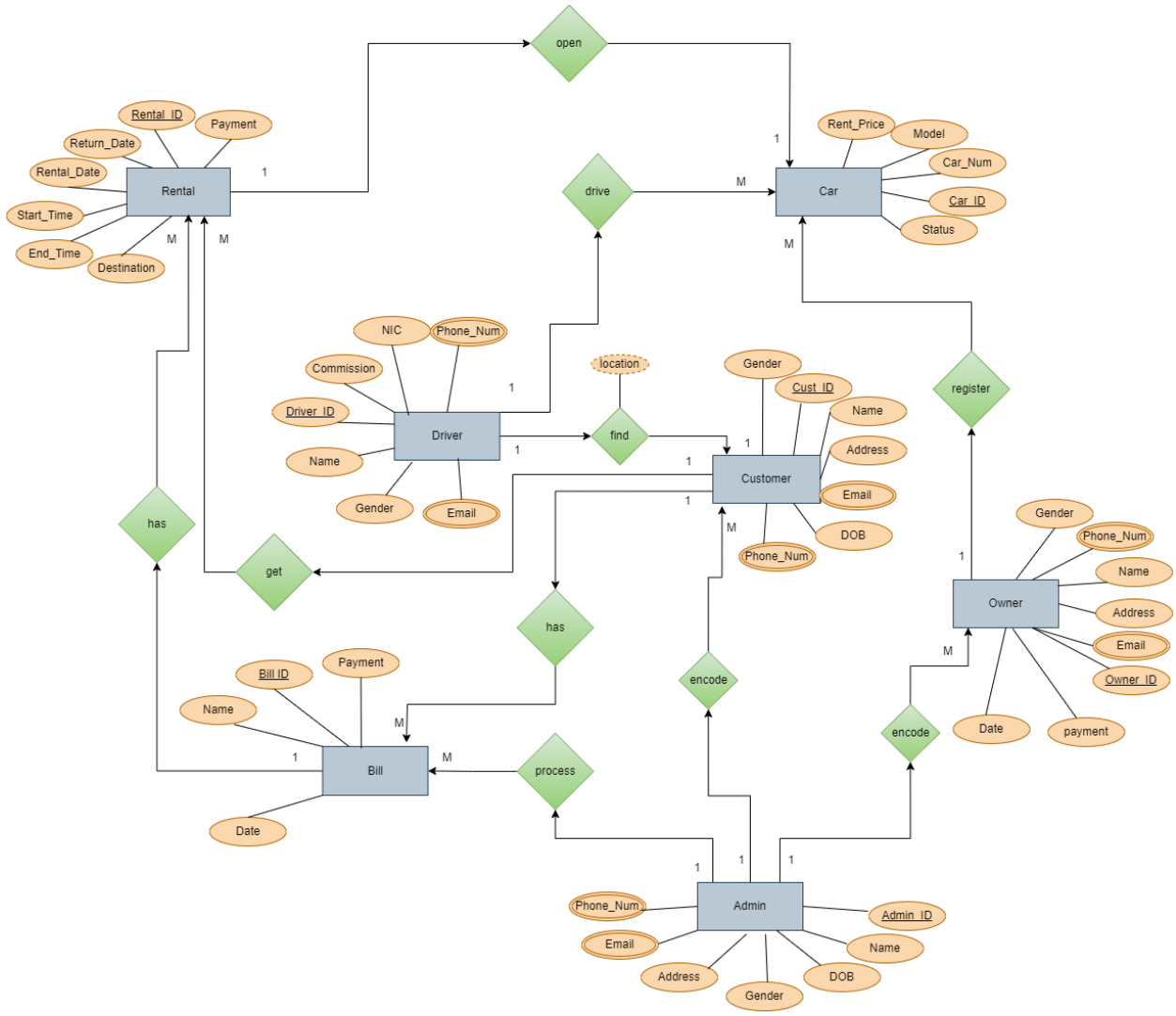
Car

- Rent_Price
- Model
- Car_Num
- Car_ID
- Status

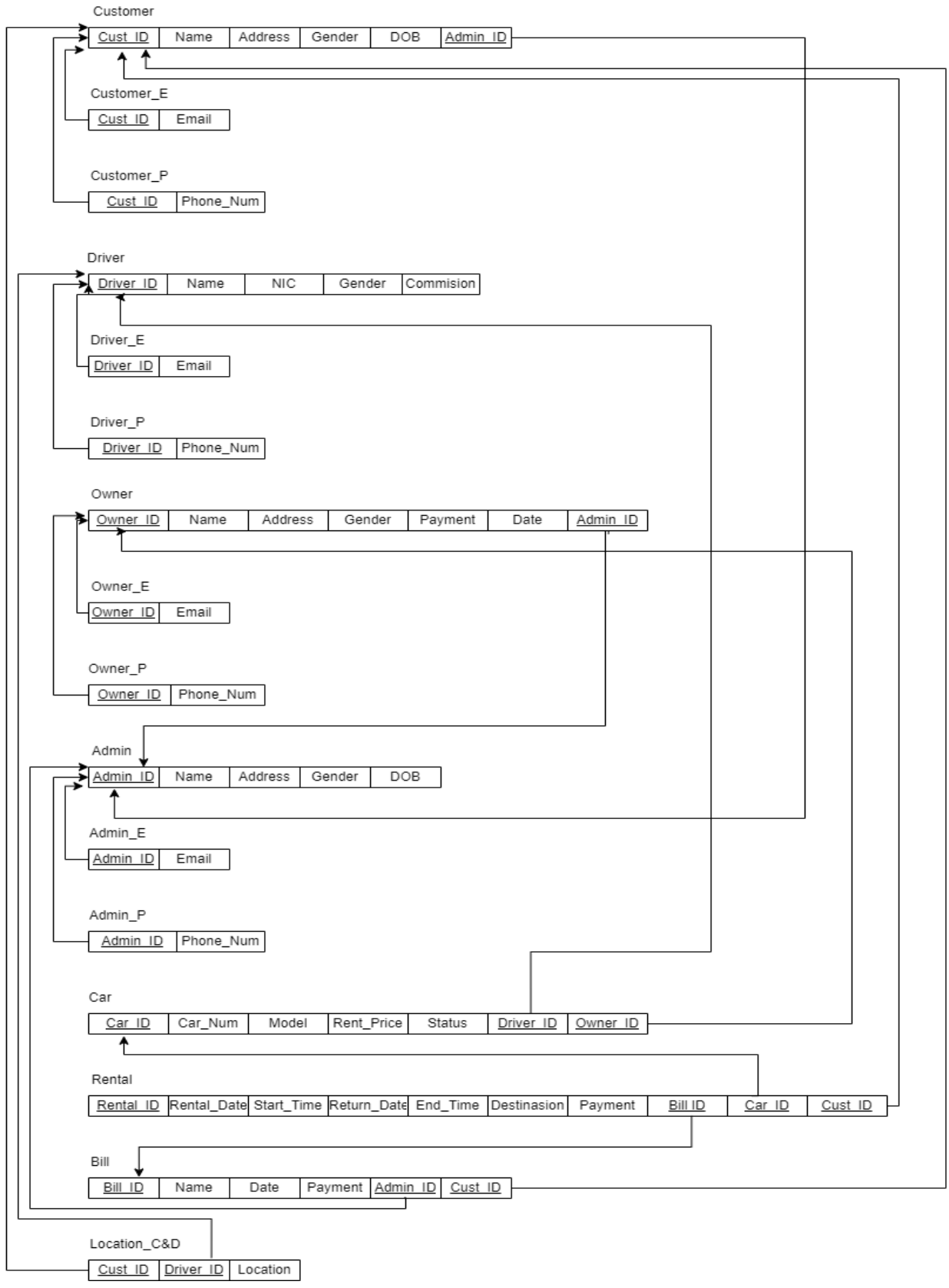
Owner

- Owner_ID
- Name
- Gender
- Phone_Num
- Age
- Address
- Email

Entity Relationship (ER diagram)



Relational Schema



SQL Queries

Data Base Create

```
/*Table Admin*/
CREATE TABLE Admin
(
Admin_ID varchar(15) not null,
Name varchar(40) not null,
Address varchar(40) not null,
Gender varchar(10) not null,
DOB date not null,

CONSTRAINT Admin_PK PRIMARY KEY(Admin_ID),
);


/*Table Admin_Email*/
CREATE TABLE Admin_E
(
Admin_ID varchar(15) not null,
Email varchar(40) CHECK (Email LIKE '%_@__%.__%') not null,

CONSTRAINT Admin_E_FK FOREIGN KEY(Admin_ID) References Admin (Admin_ID),
);


/*Table Admin_Phone no*/
CREATE TABLE Admin_P
(
Admin_ID varchar(15) not null,
Phone_Num integer not null,

CONSTRAINT Admin_P_FK FOREIGN KEY(Admin_ID) References Admin (Admin_ID),
);


/*Table customer*/
CREATE TABLE Customer
(
Cust_ID varchar(15) not null,
Name varchar(40) not null,
Address varchar(40) not null,
Gender varchar(10) not null,
DOB date not null,
Admin_ID varchar(15) not null,

CONSTRAINT Customer_PK PRIMARY KEY(Cust_ID),
CONSTRAINT Customer1_FK FOREIGN KEY(Admin_ID) References Admin (Admin_ID),
);
```

```sql
/*Table Customer_Email*/
CREATE TABLE Customer_E
(
Cust_ID varchar(15) not null,
Email varchar(40) CHECK (Email LIKE '%_@__%.__%') not null,

CONSTRAINT Customer_E_FK FOREIGN KEY(Cust_ID) References Customer (Cust_ID),
);


/*Table Customer_Phone no*/
CREATE TABLE Customer_P
(
Cust_ID varchar(15) not null,
Phone_Num integer not null,

CONSTRAINT Customer_P_FK FOREIGN KEY(Cust_ID) References Customer (Cust_ID),
);


/*Table Owner*/
CREATE TABLE Owner
(
Owner_ID varchar(15) not null,
Name varchar(40) not null,
Address varchar(40) not null,
Gender varchar(10) not null,
Payment DECIMAL(10, 2),
Date date not null,
Admin_ID varchar(15) not null,

CONSTRAINT Owner_PK PRIMARY KEY(Owner_ID),
CONSTRAINT Owner_FK FOREIGN KEY(Admin_ID) References Admin (Admin_ID),
);


/*Table Owner_Email*/
CREATE TABLE Owner_E
(
Owner_ID varchar(15) not null,
Email varchar(40) CHECK (Email LIKE '%_@__%.__%') not null,

CONSTRAINT Owner_E_FK FOREIGN KEY(Owner_ID) References Owner (Owner_ID),
);


/*Table Owner_Phone No*/
CREATE TABLE Owner_P
(
Owner_ID varchar(15) not null,
Phone_Num integer not null,

CONSTRAINT Owner_P_FK FOREIGN KEY(Owner_ID) References Owner (Owner_ID),
);
```

```
/*Table Driver*/
CREATE TABLE Driver
(
Driver_ID varchar(15) not null,
Name varchar(40) not null,
NIC varchar(10) CHECK (NIC LIKE ('[0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9][V-
v]'))not null,
Gender varchar(10) not null,
Commission DECIMAL(10,2) not null,

CONSTRAINT Driver_PK PRIMARY KEY(Driver_ID),
);


/*Table Driver_Email*/
CREATE TABLE Driver_E
(
Driver_ID varchar(15) not null,
Email varchar(40) CHECK (Email LIKE '%_@__%.__%') not null,

CONSTRAINT Driver_E_FK FOREIGN KEY(Driver_ID) References Driver (Driver_ID),
);


/*Table Driver_Phone No*/
CREATE TABLE Driver_P
(
Driver_ID varchar(15) not null,
Phone_Num integer not null,

CONSTRAINT Driver_P_FK FOREIGN KEY(Driver_ID) References Driver (Driver_ID),
);


/*Table Car*/
CREATE TABLE Car
(
Car_ID varchar(15) not null,
Car_Num varchar(10) not null,
Model varchar(20) not null,
Rent_Price DECIMAL(10, 2),
Status varchar(40) not null,
Driver_ID varchar(15) not null,
Owner_ID varchar(15) not null,

CONSTRAINT Car_PK PRIMARY KEY(Car_ID),
CONSTRAINT Car_FK FOREIGN KEY(Driver_ID) References Driver (Driver_ID),
CONSTRAINT Car1_FK FOREIGN KEY(Owner_ID) References Owner (Owner_ID),
);
```

```sql
/*Table Bill*/
CREATE TABLE Bill
(
Bill_ID varchar(15) not null,
Name varchar(40) not null,
Date date not null,
Payment DECIMAL(10, 2),
Admin_ID varchar(15) not null,
Cust_ID varchar(15) not null,

CONSTRAINT Bill_PK PRIMARY KEY(Bill_ID),
CONSTRAINT Bill_FK FOREIGN KEY(Admin_ID) References Admin (Admin_ID),
CONSTRAINT Bill1_FK FOREIGN KEY(Cust_ID) References Customer (Cust_ID),
);


/*Table Rental*/
CREATE TABLE Rental
(
Rental_ID varchar(15) not null,
Rental_Date date not null,
Start_Time varchar(10) not null,
Return_Date date not null,
End_Time varchar(10) not null,
Destination varchar(20) not null,
Payment DECIMAL(10, 2),
Bill_ID varchar(15) not null,
Car_ID varchar(15) not null,
Cust_ID varchar(15) not null,

CONSTRAINT Rental_PK PRIMARY KEY(Rental_ID),
CONSTRAINT Rental_FK FOREIGN KEY(Bill_ID) References Bill (Bill_ID),
CONSTRAINT Rental1_FK FOREIGN KEY(Car_ID) References Car (Car_ID),
CONSTRAINT Rental2_FK FOREIGN KEY(Cust_ID) References Customer (Cust_ID),
);


/* Table for Location */

CREATE TABLE Location_CD
(
Cust_ID varchar(15) not null,
Driver_ID varchar(15) not null,
Location varchar(40) not null

CONSTRAINT Location_CD1_FK FOREIGN KEY(Cust_ID) References Customer (Cust_ID),
CONSTRAINT Location_CD2_FK FOREIGN KEY(Driver_ID) References Driver (Driver_ID),
);
```

Data Store in Data Base

```
/* Details of Admin */

Insert INTO Admin values('A7001','Savindu Perera','Ahangama,Galle','Male','1978-09-23');
Insert INTO Admin values('A7002','Sandeepa
Imalshi','Galenbidunuwewa,Anuradhapura','Female','1969-09-07');
Insert INTO Admin values('A7003','Iduru Liyanage','Sooriya
Uyana,Piliyandala','Male','1976-06-16');
Insert INTO Admin values('A7004','Akarsha Silva','Poramba,Ambalangoda','Male','1974-09-
23');
Insert INTO Admin values('A7005','Dilshani Rajapaksha','Bandaranayaka
Mawatha,Colombo','Female','1968-04-26');

/* Details of Admin's Phone numbers */

Insert INTO Admin_P values('A7001',0725123478);
Insert INTO Admin_P values('A7001',0786543123);
Insert INTO Admin_P values('A7002',0715126980);
Insert INTO Admin_P values('A7003',0761234564);
Insert INTO Admin_P values('A7005',0776028279);

/* Details of Admin's Emails */

Insert INTO Admin_E values('A7001','A7001@gmail.com');
Insert INTO Admin_E values('A7002','A7002@gmail.com');
Insert INTO Admin_E values('A7002','A7002_new@gmail.com');
Insert INTO Admin_E values('A7003','A7003@gmail.com');
Insert INTO Admin_E values('A7005','A7005@gmail.com');

/*Details of customers*/

Insert INTO Customer values ('C1001','Sunil Perera','Thunmulla,Alawala','Male','1999-10-
02','A7005');
Insert INTO Customer values ('C1002','Nimal Subasinghe','Gonawala,Kelaniya','Male','1970-
12-10','A7001');
Insert INTO Customer values ('C1003','Mala
Dissanayake','Heiyanthuduwa,Thlawa','Female','1985-11-14','A7003');
Insert INTO Customer values ('C1004','Kumara
Aberathne','Eppawala,Anuradhapura','Male','1981-03-15','A7001');
Insert INTO Customer values ('C1005','Chamara
Kulathunga','Thumbulla,Nawathalwattha','Male','1950-06-09','A7002');

/*Details of customer's Phone numbers*/

Insert INTO Customer_P values('C1001',0775846952);
Insert INTO Customer_P values('C1002',0714586996);
Insert INTO Customer_P values('C1003',0758465985);
Insert INTO Customer_P values('C1004',0714523632);
Insert INTO Customer_P values('C1005',0768524123);
```

```sql
/*Details of customer's Emails*/

Insert INTO Customer_E values ('C1001','Sunil@gmail.com');
Insert INTO Customer_E values ('C1002','Nimal@gmail.com');
Insert INTO Customer_E values ('C1003','Mala@gmail.com');
Insert INTO Customer_E values ('C1004','Kumara@gmail.com');
Insert INTO Customer_E values ('C1005','Chamara@gmail.com');

/* Details of Owner */

Insert INTO Owner values('O5001','Bimal
Rajapaksha','Bandaranayakapura,Rajagiriya','Male',40000.00,'2023-05-22','A7001');
Insert INTO Owner values('O5002','Duminda
Sudharshana','Isurupura,Kothalawala','Male',20000.00,'2023-05-01','A7005');
Insert INTO Owner values('O5003','Kavindu
Perera','Kaduruduwa,Galle','Male',30000.00,'2023-05-06','A7005');
Insert INTO Owner values('O5004','Nimna
Rathnayaka','Bogahagoda,Malabe','Male',70000.00,'2023-04-23','A7003');
Insert INTO Owner values('O5005','Ashen
Bandara','Wanchawala,Ambalangoda','Male',10000.00,'2023-05-23','A7003');

/* Details of Owner's phone numbers */

Insert INTO Owner_P values ('O5001',0714569871);
Insert INTO Owner_P values ('O5002',0784561239);
Insert INTO Owner_P values ('O5003',0764654790);
Insert INTO Owner_P values ('O5004',0784534123);
Insert INTO Owner_P values ('O5005',0725126980);

/* Details of Owner's Emails */

Insert INTO Owner_E values('O5001','Bimal@gmail.com');
Insert INTO Owner_E values('O5002','Duminda@gmail.com');
Insert INTO Owner_E values('O5003','Kavindu@gmail.com');
Insert INTO Owner_E values('O5004','Nimna@gmail.com');
Insert INTO Owner_E values('O5005','Ashen@gmail.com');

/*Details of Drivers */

Insert INTO Driver values ('D2001','Geeth Subasinghe','685421365v','Male',45000.00);
Insert INTO Driver values ('D2002','Sameera Ranathunga','794563218v','Male',60000.00);
Insert INTO Driver values ('D2003','Gayan Thilakarathne','659841215v','Male',38000.00);
Insert INTO Driver values ('D2004','Kasun Tharaka','975461546v','Male',56000.00);
Insert INTO Driver values ('D2005','Kumari De Alwis','885465987v','Female',63000.00);

/*Details of Driver's phone numbers */

Insert INTO Driver_P values ('D2001',0741112223);
Insert INTO Driver_P values ('D2002',0701576548);
Insert INTO Driver_P values ('D2003',0785132459);
Insert INTO Driver_P values ('D2004',0745861235);
Insert INTO Driver_P values ('D2005',0785492635);
```

```
/*Details of Driver's Emails */

Insert INTO Driver_E values ('D2001','Geeth@gmail.com');
Insert INTO Driver_E values ('D2002','Sameera@gmail.com');
Insert INTO Driver_E values ('D2003','Gayan@gmail.com');
Insert INTO Driver_E values ('D2004','Kasun@gmail.com');
Insert INTO Driver_E values ('D2005','Kumari@gmail.com');

/* Details of Car */

Insert INTO Car values ('V1001','CD-1020','Alto',3500.00,'Petrol','D2002','O5002');
Insert INTO Car values ('V1002','CAB-2179','Starlet',2000.00,'Diesel','D2003','O5001');
Insert INTO Car values ('V1003','CAA-1254','BMW',18000.00,'Petorl','D2005','O5003');
Insert INTO Car values ('V1004','CAE-1523','Wagon R',2500.00,'Petrol','D2004','O5002');
Insert INTO Car values ('V1005','CAD-8745','Maruti',1300.00,'Diesel','D2005','O5004');

/* Details of Bill */

Insert INTO Bill values ('B0001','Bill_1','2022-11-29',5000.00,'A7005','C1001');
Insert INTO Bill values ('B0002','Bill_2','2023-09-12',34000.00,'A7002','C1003');
Insert INTO Bill values ('B0003','Bill_3','2022-11-24',7000.00,'A7004','C1005');
Insert INTO Bill values ('B0004','Bill_4','2023-08-12',20000.00,'A7001','C1004');
Insert INTO Bill values ('B0005','Bill_5','2023-05-17',15000.00,'A7004','C1003');


/*Details of Rental */

Insert INTO Rental values ('R5001','2023-02-08','10.30A.M.','2023-02-
10','10.30P.M','Kelaniya',5000.00,'B0005','V1005','C1001');
Insert INTO Rental values ('R5002','2023-04-10','11.00A.M','2023-04-
14','09.00P.M','Kandy',6100.00,'B0001','V1001','C1005');
Insert INTO Rental values ('R5003','2022-12-31','12.30P.M','2023-01-
07','11.00A.M','Colombo',8500.00,'B0004','V1003','C1002');
Insert INTO Rental values ('R5004','2023-05-10','08.00A.M','2023-05-
10','10.00P.M','Warakapola',3000.00,'B0002','V1004','C1001');
Insert INTO Rental values ('R5005','2023-06-14','07.00A.M','2023-06-
15','08.40A.M','Galle',5200.00,'B0003','V1002','C1004');

/*Details of Location */

INSERT INTO Location_CD values ('C1005','D2002','Kandy');
INSERT INTO Location_CD values ('C1004','D2003','Malabe');
INSERT INTO Location_CD values ('C1001','D2001','Ganemulla');
INSERT INTO Location_CD values ('C1002','D2004','Nittambuwa');
INSERT INTO Location_CD values ('C1003','D2005','Danowita');
```

Performance Requirement

A major role is played by Performance Requirements to make the system successful. They are as follow,

- System must be active 24 hours, 365 days for a registered customer to access the system without any inconvenience.
- A Registered customer can access the system numerous times by entering his/ her login credentials.
- The login process and page loading must be completed in just a few seconds of time.
- Speed and usability are the performance requirements for this system.
- Registered customers can view rental details.
- Registered customers can edit or delete his/her account details.
- System loads in a short period of time.
- Admin can add or remove shared experiences and feedback, Approve bookings and cancel bookings, manage user accounts, and edit/ update rental details.
- The system allows the driver to respond the customer messages.
- Developers can develop new functions for the website.
- Developer can update the system.
- The developer needs to detect bugs and errors to fix them.
- Design user friendly user interface.
- Users must be able to access the website at anytime using any device or browser.
- System provide ability for Admin to manage the staff accounts.

Security Requirements

- Before sending to the database, customers' personal information should be encrypted.
- Unauthorized users should be unable to access restricted features.
- A backup of all the data in the system should be kept in the database.
- The password of the customer account must be a strong password which is included uppercase letters, lowercase letters, numbers, and special characters.
- There should be just one user account per email address.
- Database server must maintain with redundant server.
- The system's data can only be accessed and modified by the admin.