Information Systems and Data Modeling – IT1090

Assignment

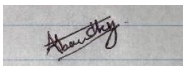
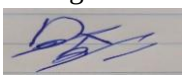
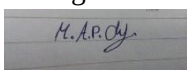
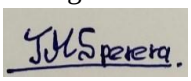
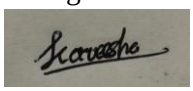
Title: Automated parking system	
Batch Number: Group 01.01	Group Number: **MLB_WD_01.01_09**

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:

IT22111692	ABEWARDHANA J.H. K	 signature
IT22272768	BASNAYAKA W.B.M.D. K	 signature
IT22127778	CHAVINDEE M.A. P	 signature
IT22138668	PERERA T.M. S	 signature
IT22116260	THARINDI W.A. K	

Submitted on: 13/10/2023

Content

1 Introduction

When examining automated parking systems, we considered how the internet links consumers and businesses and how a website may make a service more readily available to customers. As a result, we developed a website where individuals can browse and book parking spaces in order to find a solution for the automated parking system. Customers can register on our website and use it to ask questions about our services, get customer service, and make improvement suggestions. We offer a number of online services for urban areas' automated parking systems.

In order to create an automated parking website, a range of technologies must be investigated and comprehended. They included relational databases (like My SQL) and programming languages (including HTML, CSS, and JavaScript).

The variety of data that may be managed by a database must be substantial. Because maintaining an electronic system for storing data is simpler than maintaining written or typed documents, databases are necessary. The result is time and effort savings for both users and administrators. Additionally, there is no chance of losing the data since it is kept in a database. The only people who can manage the system are its users. Performing a requirement analysis and gathering requirements are essential steps in the database construction process. It is thus possible to determine the data needs, together with the functional and nonfunctional specifications.

2 Hypothetical Scenario

The online automated parking system "Arrow" offers excellent service in the urban region. When a visitor uses the sign-up button, they can register for an account and then use the login form to access the website.

The user will be verified by the website's database and granted access to the website when they have successfully registered and logged in. In addition to these features, a registered user has the ability to change his personal information. Additionally, by using the "support page" feature, he can get in touch with the website owners and request other services. The website's contact information is displayed to the user in that field. Using this option, the user can submit a request if desired.

The parking ticket and available parking spaces are sent to the registered users' mobile phone by the system operator when they land at the parking land inspection point. After that, the registered user can go to the designated location and hand over their vehicle to a maintenance technician so that it can be parked on racks. Customers can report any incidents that occurred during this parking period to the security guard, and the guard will resolve them.

Staff workers can also submit a request to have any malfunctioning equipment in the parking area fixed. The security guard and the maintenance technician should each turn in a report detailing their completed tasks after finishing their shift. Lastly, the system operator should provide the system manager with a summary report that has been created.

3 Requirement Analysis

3.1 Main Requirements

3.1.1 Functional Requirements

The functional requirements explain the primary functions of the website and the relationships between users and the system. This Arrow Automated parking system has five users. The Visitor, System Operator, Head of security, System Manager and Maintenance Technician. When it relates to them, they use various methods to access this system.

1. **Visitor and Registered user (They can access the front-end of the system).**

User requirements

- Guests can check FAQ and Support page.
- Guest views the available Feedback.
- Guests can get register to the system by providing required details for the registration.
- Guests can get username and user ID for system.
- Guests can check availability for reservations.
- Registered User login to the system using by providing required user login credential.
- Registered User can Contact Arrow Automated parking system quest in the Contact number of the website.
- Registered User can Edit their own Account details.

System requirements

- Registered User can Edit their own Account details.
- System should display Feedback Uploaded by the customers on the website.
- System should approve registration Details and Create a user Account.
- The system should display the availabilities of Reservations on the website.
- System should validate the login Credentials entered by the Registered User.
- System should provide ability to Upload Feedback and Experiences.
- System should send the Reservation Cancellation Request to the admin and store details.

2. **Maintenance technician**

User requirements-

- Maintenance technician login to the system using by providing required user login credentials.
- Maintenance technician can Generate reports on the system's performance.
- Maintenance technician can Receive alerts when problems are detected in the system.
- A maintenance technician can check the quality of security features in parking area.

System requirements-

- System should validate user login Credentials entered by the maintenance technician.
- The system displays emergency technical cases to maintenance technician.
- The system allows for the coordination between customer and maintenance technician.
- System allows to check support requests to maintenance technician.
- System allows to mark completed requests.

3.Head of security

User requirements-

- Security officer login to the system using by providing required user login credentials.
- Maintenance technician can Generate reports on the security of parking area.
- Maintenance technician can Receive alerts about security.

System requirements-

- System should validate user login Credentials entered by the security officer.
- The system displays security cases to the maintenance technician.
- The system allows for the coordination between customer and security officer.
- System allows to check complain requests to security officer.

4.System Operator

User requirements-

- System Operator can sign into the website using login credentials.
- System Operator approve Registered user.
- System Operator can validate approved registered users.
- System Operator can make user's payments of ticket.
- System Operator can send parking tickets to the paid user.
- System Operator can park location to the user.
- System Operator can manage and troubleshoot user accounts.
- System Operator can generate summary reports.

System requirements-

- System should validate the user login credentials.
- The system should store the validate user's details.
- The system should check and stores payment details of the users.
- The system should reserve parking locations according to the payment.
- System should troubleshoot.

5.System Manager

User requirements-

- System Manager can login to the system using a username and password.
- The system manager should be able to add, remove or modify parking space.
- System managers manage User Accounts.
- System manager Access permissions for different user groups.
- System manager handles payment reports.
- System managers monitor the system's performance from other actors.

System requirements-

- The system should support user authentication.
- The system should provide a dashboard that displays real-time information about the system.
- The system should support the allocation of parking spaces.
- The system should support maintaining reports of payment.
- The system should support integration with external systems.
- The system should delete, update ,store the details of the database of parking system

3.1.2 Non-Functional Requirements

The term "quality attributes" refers to non-functional needs. The system's properties that are not directly related to a given functionality are described. It's possible that non-functional needs are more important than functional requirements. The system can be useless if these goals are not met.

Speed

- The system needs to operate quickly.
- The system may connect to more users concurrently without experiencing any issues.

User friendly

- Users with limited IT literacy should be able to use the system.

Response Time:

- The system must react to user requests in a predetermined amount of time.

Availability

- The system should be available 24 hours and 7 days.
- The system should be accessible to users with low IT literacy.
- A certain amount of time should be allowed for the system to remain accessible and functional (for example, 99.9% uptime).

Reliability

- The system must have ability to detect the invalid user credentials.
- Even if there are hardware or software problems, the system should still function.

Security

- The system should have ability to prevent unauthorized access, misuse, forgery and secure user data.
- Also, by providing unique user ID and password, no one can access the system by using any other's user ID and password.

Scalability

- By boosting resources, the system ought to be able to handle greater loads.

Reliability

- An administrator can add, modify, remove, and update properties.
- The system can be used by any number of users simultaneously, and it will respond to user requests in a very quick manner.

Data Requirements

1.Registered user

- User_ID
- User Name
- F_Name
- L_Name
- Address
- DOB
- Email
- PhoneNo
- Password
- Age

2.Visitor

- F_Name
- L_Name
- Email
- Address
- Gender
- DOB
- Age
- PhoneNo

Maintenance technician

- Name
- M_ID
- DOB
- Phone No
- Email

4.Head of Security

- S_ID
- Name
- Address
- Phone_No
- Email

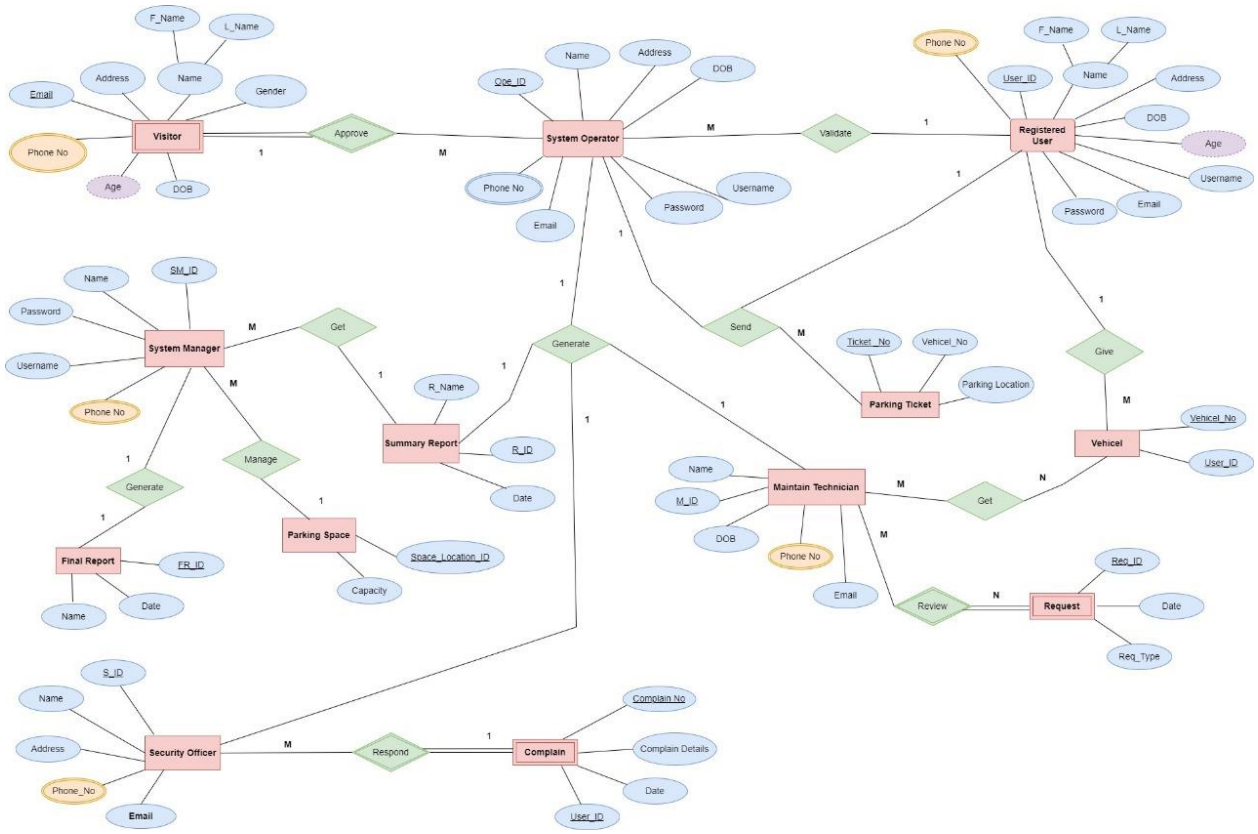
5.System Manager

- SM_ID
- Name
- username
- Phone Number
- Password

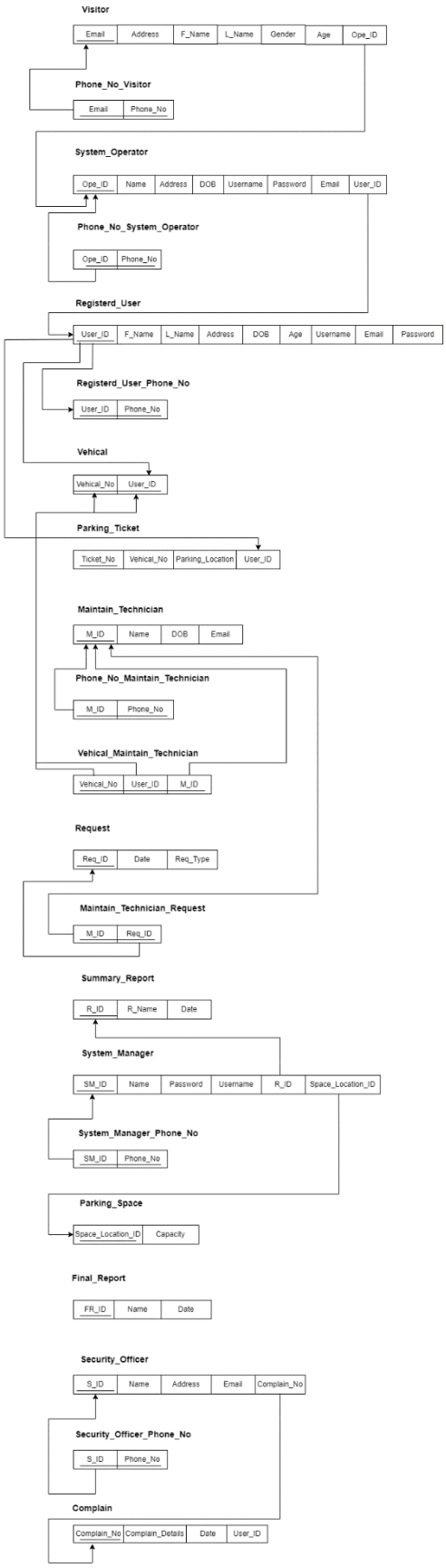
6.System Operator

- Ope_ID
- Name
- Address
- DOB
- Username
- Password
- Email
- Phone No

4 Entity Relationship (ER Diagram)



5 Relational Schema


Visitor
Email
Address
F_Name
L_Name
Gender
Age
Ope_ID
Phone_No_Visitor
Email
Phone_No
System_Operator
Ope_ID
Name
Address
DOB
Username
Password
Email
User_ID
Phone_No_System_Operator
Ope_ID
Phone_No
Registerd_User
User_ID
F_Name
L_Name
Address
DOB
Age
Username
Email
Password
Registerd_User_Phone_No
User_ID
Phone_No
Vehical
Vehical_No
User_ID
Parking_Ticket
Ticket_No
Vehical_No
Parking_Location
User_ID
Maintain_Technician
M_ID
Name
DOB
Email
Phone_No_Maintain_Technician
M_ID
Phone_No
Vehical_Maintain_Technician
Vehical_No
User_ID
M_ID
Request
Req_ID
Date
Req_Type
Maintain_Technician_Request
M_ID
Req_ID
Summary_Report
R_ID
R_Name
Date
System_Manager
SM_ID
Name
Password
Username
R_ID
Space_Location_ID
System_Manager_Phone_No
SM_ID
Phone_No
Parking_Space
Space_Location_ID
Capacity
Final_Report
FR_ID
Name
Date
Security_Officer
S_ID
Name
Address
Email
Complain_No
Security_Officer_Phone_No
S_ID
Phone_No
Complain
Complain_No
Complain_Details
Date
User_ID

6 SQL Queries

6.1 Data Base Create

```
--create Table RegisteredUser create by J.H.K ABEWARDHANA--
create table  RegisteredUser(
	userId varchar(9) not null,
	user_name varchar(50) not null,
	F_Name char(50) not null,
	L_Name char(50) not null,
	Address varchar(100) not null,
	DOB date not null,
	Email varchar(100) not null,
	Password varchar(100) not null,
	Age int not null,
	constraint RegisteredUuser_PK PRIMARY KEY(userId)
);



--create Table RegisteredUser_PhoneNo create by J.H.K ABEWARDHANA--
create table RegisteredUser_PhoneNo(
	userId varchar(9) not null,
	PhoneNo int not null,
	constraint RegisteredUser_PhoneNo_PK PRIMARY KEY(userId),
	constraint RegisteredUser_PhoneNo_FK FOREIGN KEY(userId)  REFERENCES
RegisteredUser(userId)
);



--create Table visitor create by J.H.K ABEWARDHANA
create table Visitor(
	Vid int,
	Email varchar(100) not null,
	F_Name char(50) not null,
	L_Name char(50) not null,
	Address varchar(100) not null,
	Gender char(10) not null,
	DOB date not null,
	Age int not null,
	--OpeID varchar(50),
	constraint Visitor_PK PRIMARY KEY(Vid),
    --constraint Visitor_OpeID_FK FOREIGN KEY(OpeID) REFERENCES systemOperator(OpeID)
);
```

```
-- create table  Phone_Visitor create by J.H.K ABEWARDHANA
create table Phone_Visitor(
	Vid int,
	PhoneNo int not null,
	constraint Phone_Visitor_PK PRIMARY KEY(Vid),
	constraint Phone_Visitor_FK FOREIGN KEY(Vid) REFERENCES Visitor(Vid)
);


---create Table systemOperator create by PERERA T.M. S  --
create table systemOperator(
	OpeID varchar(50),
	Name char(50) not null,
	Address varchar(100) not null,
	DOB date not null,
	userName varchar(50) not null,
	Password varchar(100) not null,
	Email varchar(100) not null,
	userId varchar(9) not null,
	constraint systemOperator_PK PRIMARY KEY(OpeID),
	constraint systemoOperator_userId_FK FOREIGN KEY(userId) REFERENCES
RegisteredUser(userId)
);


--create table PhoneNo_systemOperator create by PERERA T.M. S
create table PhoneNo_systemOperator(
	OpeID varchar(50),
	PhoneNo int not null,
	constraint PhoneNo_systemOperator_PK PRIMARY KEY(OpeID),
	constraint PhoneNo_systemOperator_OpeID_FK FOREIGN KEY(OpeID) REFERENCES
systemOperator(OpeID)
);


--create table vehicle create by PERERA T.M. S
create table vehicle(
	VehicelNo varchar(10) not null,
	userId varchar(9) not null,
	constraint vehicle_PK  PRIMARY KEY(VehicelNo),
	constraint vehicle_userId_FK FOREIGN KEY(userId) REFERENCES RegisteredUser(userId)
);
```

```
--create ParkingTicket create by CHAVINDEE M.A. P
create table ParkingTicket(
	TicketNo varchar(50),
	VehicelNo varchar(10) not null,
	ParkingLocation varchar (50) not null,
	userId varchar(9) not null,
	constraint ParkingTicket_PK PRIMARY KEY(TicketNo),
	constraint ParkingTicket_userId_FK FOREIGN KEY(userId) REFERENCES
RegisteredUser(userId)
);



--create MaintainTechnician create by CHAVINDEE M.A. P
create table MaintainTechnician(
	MID varchar(10) not null,
	Name varchar(50) not null,
	DOB date not null,
	Email varchar(100) not null,
	constraint  MaintainTechnician_PK PRIMARY KEY(MID),
);



--create table  PhoneNo_MaintainTechania create by CHAVINDEE M.A. P
create table PhoneNo_MaintainTechania(
	MID varchar(10) not null,
	PhoneNo int not null,
	constraint PhoneNo_MaintainTechania_PK PRIMARY KEY(MID),
	constraint PhoneNo_MaintainTechania_MID_FK FOREIGN KEY(MID) REFERENCES
MaintainTechnician(MID)
);



--create table Vehicel_MaintainTechnician create by CHAVINDEE M.A. P
create table Vehicel_MaintainTechnician(
	VehicelNo varchar(10) not null,
	userId varchar(9) not null,
	MID varchar(10) not null,
	constraint Vehicel_MaintainTechnician_VehicelNo_FK FOREIGN KEY(VehicelNo)
REFERENCES vehicle(VehicelNo),
	constraint Vehicel_MaintainTechnician_userId_FK FOREIGN KEY(userId) REFERENCES
RegisteredUser(userId),
	constraint Vehicel_MaintainTechnician_MID_FK FOREIGN KEY(MID) REFERENCES
MaintainTechnician(MID)
);
```

```
--create Table Request create by BASNAYAKA W.B.M.D. K
create table Request(
	RegID varchar (20) not null,
	DOB date not null,
	Request_type varchar(50) not null,
	constraint Request_PK PRIMARY KEY(RegID)
);


----create Table SummaryReport create by PERERA T.M. S
create table SummaryReport(
	RID int not null,
	Rname varchar(50) not null,
	date date not null,
	constraint SummaryReport_PK PRIMARY KEY(RID)
);



--create table maintainTechicia_request create by CHAVINDEE M.A. P
create table maintainTechicia_request(
	MID varchar(10) not null,
	RegID varchar (20) not null,
	constraint maintainTechicia_request_MID_FK FOREIGN KEY(MID) REFERENCES
MaintainTechnician(MID),
	constraint maintainTechicia_request_FK FOREIGN KEY(RegID) REFERENCES
Request(RegID)
);



--create Table parkingSpace  create by BASNAYAKA W.B.M.D. K
create table parkingSpace(
	spaceLocationID int not null,
	capacity int not null,
	constraint parkingSpace_PK PRIMARY KEY(spaceLocationID)
);
```

```
---create Table systeManager create by THARINDI W.A. K
create table systeManager(
	SMID varchar(10) not null,
	Name char(50) not null,
	Password varchar(100) not null,
	user_name char(50) not null,
	RID int not null,
	spaceLocationID int not null,
	constraint system_manager_PK PRIMARY KEY(SMID),
	constraint  system_manager_RID_FK FOREIGN KEY(RID) REFERENCES  SummaryReport(RID),
	constraint  system_manager_spaceLocationID_FK FOREIGN KEY(spaceLocationID)
REFERENCES parkingSpace(spaceLocationID)
);



---create Table PhoneNo_systeManager create by THARINDI W.A. K
create table PhoneNo_systeManager(
	SMID varchar(10) not null,
	phoneNo int not null,
	constraint PhoneNo_systeManager_PK PRIMARY KEY(SMID)
);



--creat table finalReport create by THARINDI W.A. K
create table finalReport(
	FRid int not null,
	F_Date date not null,
	name varchar (50) not null,
	constraint finalReport_PK PRIMARY KEY(FRid)
);



--creat table Security_Officer create by BASNAYAKA W.B.M.D. K

CREATE TABLE Security_Officer(
    S_ID varchar(15) not null,
	Name varchar(40) not null,
	Address varchar(50) not null,
	--Complain_No varchar(15) NOT NULL,
	Email varchar(40) CHECK (Email LIKE '%_@__%.__%') not null,

	CONSTRAINT SECURITY_OFFICER_PK PRIMARY KEY(S_ID),
	--CONSTRAINT SECURITY_OFFICER_FK FOREIGN KEY(Complain_No)REFERENCES
Complain(Complain_No)
);
```

```
--creat table Complain create by BASNAYAKA W.B.M.D. K
CREATE TABLE Complain(
   Complain_No varchar(15) NOT NULL,
   DOB date not null,
   Complain_Details varchar(100)NOT NULL,
  userId varchar(9) not null,

   CONSTRAINT Complain_PK PRIMARY KEY(Complain_No),
   CONSTRAINT Complain_userId_FK FOREIGN KEY(userId)REFERENCES RegisteredUser(userId)
);



--creat table  Security_Officer_phone    by BASNAYAKA W.B.M.D. K
create table Security_Officer_phone(
       SOPid int,
       phoneNO int,
       S_ID varchar(15) not null,
       CONSTRAINT Security_Officer_phone_PK PRIMARY KEY(SOPid),
       CONSTRAINT Complain_S_ID_FK FOREIGN KEY(S_ID)REFERENCES Security_Officer(S_ID)
);
```

6.2 Data Store in Data Base

```
--insert values for RegisteredUser create by J.H.K ABEWARDHANA
INSERT INTO RegisteredUser VALUES('US01','Tharidu','J.K.Tharidu
Kumara','Kumara','NO231/Balngoda','2001/01/04','tharidu@gmail.com','taridu23&3','21');
INSERT INTO RegisteredUser VALUES('US02','Lashan','M.K.Lashan
Rashmika','Rashmika','NO251/Pinnawala','2002/04/13','lashan@gmail.com','lashan5542#','21'
);
INSERT INTO RegisteredUser VALUES('US554','Pasindu','P.K.Pasind
Maduranga','Pasindu','NO55/Badulla','1998/2/05','pasindu@gmail.com','pasindu@453','25');
INSERT INTO RegisteredUser VALUES('US550','Hashen ','H.J.K .Hashen
Kavishka','Hashen','NO5/B/Badulla','1999/2/05','hashen@gmail.com','hashen1234','24');
INSERT INTO RegisteredUser
VALUES('US500','Kaushalya','D.K.kaushalya','Kaushalya','NO75/Biyagama','2000/3/05','kaush
alya@gmail.com','kaushalya@453#','23');
```

```
--insert RegisteredUser_PhoneNo table create by J.H.K ABEWARDHANA
INSERT INTO RegisteredUser_PhoneNo VALUES('US01','0774561274');
INSERT INTO RegisteredUser_PhoneNo VALUES('US02','0798561264');
INSERT INTO RegisteredUser_PhoneNo VALUES('US554','0721751296');
INSERT INTO RegisteredUser_PhoneNo VALUES('US550','0701751096');
INSERT INTO RegisteredUser_PhoneNo VALUES('US500','0751758296');
```

```
--insert visitor table create by J.H.K ABEWARDHANA
INSERT INTO Visitor VALUES('101','Nuwan@gmail.com','L.M.Nuwan
Premarthna','Premarthna','NO45/Manarama','Male','2001/8/12','21');
INSERT INTO Visitor VALUES('102','Kumari@gmail.com','M.P.Kumari
Jayamaha','Jayamaha','NO98/Monaragala','Female','1997/12/02','26');
INSERT INTO Visitor VALUES('103','Lakshan@gmail.com','K.K.Lakshan
Suraweera','suraweera','NO102/Parakaduwa','Male','1999/4/04','24');
INSERT INTO Visitor VALUES('104','suharda@gmail.com','M.L.Suharda
Jayarathna','Jayarathna','NO100/Kandy','Female','1998/5/04','25');
INSERT INTO Visitor VALUES('105','Nishantha@gmail.com','Nishantha
Gamage','Gamage','NO102/90/Galle','Male','2000/09/04','23');
```

```
--insert Phone_Visitor create by J.H.K ABEWARDHANA
INSERT INTO Phone_Visitor VALUES('101','0784561234');
INSERT INTO Phone_Visitor VALUES('102','0714582345');
INSERT INTO Phone_Visitor VALUES('103','0778947545');
INSERT INTO Phone_Visitor VALUES('104','0708947549');
INSERT INTO Phone_Visitor VALUES('105','0748947045');


/* Insert data to the system operator table create by PERERA T.M. S*/
INSERT INTO systemOperator VALUES ('S001','Maheli
Perera','No.12,Dikhenapura,Horana.','1993-10-
02','Oparator1','Suha#123mn','Perera21@gmail.com','US01');
INSERT INTO systemOperator VALUES ('S002','Sanduni Fernando','No.70/A,Kaluthara.','1975-
11-17','Oparator2','1234@','Sandu12v@gmail.com','US02');
INSERT INTO systemOperator VALUES ('S003','Nimal Bandara','No.123,Kandy
Road,Malabe.','1970-02-08','Oparator3','shan12@','NimalB57@gmail.com','US554');
INSERT INTO systemOperator VALUES ('S004','Keshara Abesingha','No.12,Kandy
Road,Kaduwela.','1990-01-30','Oparator4','
       Keshara123','Keshara004@gmail.com','US550');
INSERT INTO systemOperator VALUES ('S005','Bimsara Kavindu','No.07,Kotte
Road,Baththaramulla.','2000-02-
10','Oparator5','Bimsara#123','Bimsara@gmail.com','US500');


--insert data to the PhoneNo_systemOperator     create by PERERA T.M. S
INSERT INTO PhoneNo_systemOperator VALUES ('S001','0772367253');
INSERT INTO PhoneNo_systemOperator VALUES ('S002','0713498675');
INSERT INTO PhoneNo_systemOperator VALUES ('S003','0770912397');
INSERT INTO PhoneNo_systemOperator VALUES ('S004','0770012390');
INSERT INTO PhoneNo_systemOperator VALUES ('S005','0760982307');


--insert data to ParkingTicket create by PERERA T.M. S
INSERT INTO ParkingTicket VALUES ('T001','GP-7345','LG10','US01');
INSERT INTO ParkingTicket VALUES ('T002','CAT-3366','LG20','US02');
INSERT INTO ParkingTicket VALUES ('T003','GJ-9345','LG11','US554');
INSERT INTO ParkingTicket VALUES ('T004','GK-9905','LG18','US550');
INSERT INTO ParkingTicket VALUES ('T005','GM-9300','LG15','US500');
```

```
--insert data to  vehicle create by BASNAYAKA W.B.M.D. K
INSERT INTO vehicle VALUES ('GP-7345','US01');
INSERT INTO vehicle VALUES ('CAT-3366','US02');
INSERT INTO vehicle VALUES ('GJ-9345','US554');
INSERT INTO vehicle VALUES ('GK-9905','US550');
INSERT INTO vehicle VALUES ('GM-9300','US500');
```

```
---insert data to  MaintainTechnician create by CHAVINDEE M.A. P
INSERT INTO MaintainTechnician
VALUES('M0001','R.K.A.Fernando','1980/01/01','rkasfernando@gmail.com');
INSERT INTO MaintainTechnician
VALUES('M0002','R.Sarath','1975/08/08','rsarath@gmail.com');
INSERT INTO MaintainTechnician VALUES('M0003','Kumara
Perera','1970/05/01','pererakumara@gmail.com');
INSERT INTO MaintainTechnician VALUES('M0004','Anushka
Kuruppu','1975/10/01','anushkakuruppu@gmail.com');
INSERT INTO MaintainTechnician VALUES('M0005','Anura
Kumara','1978/06/07','anurakumara@gmail.com');
```

```
--insert data to PhoneNo_MaintainTechania create by CHAVINDEE M.A. P
INSERT INTO PhoneNo_MaintainTechania  VALUES('M0001','0712345678');
INSERT INTO PhoneNo_MaintainTechania VALUES('M0002','0782345670');
INSERT INTO PhoneNo_MaintainTechania VALUES('M0003','0702345678');
INSERT INTO PhoneNo_MaintainTechania VALUES('M0004','0700345078');
INSERT INTO PhoneNo_MaintainTechania VALUES('M0005','0712305608');
```

```
--insert data to Vehicel_MaintainTechnician create by CHAVINDEE M.A. P
INSERT INTO Vehicel_MaintainTechnician VALUES('GP-7345','US01','M0001');
INSERT INTO Vehicel_MaintainTechnician VALUES('CAT-3366','US02','M0002');
INSERT INTO Vehicel_MaintainTechnician VALUES('GJ-9345','US554','M0003');
INSERT INTO Vehicel_MaintainTechnician VALUES('GK-9905','US550','M0004');
INSERT INTO Vehicel_MaintainTechnician VALUES('GM-9300','US500','M0005');
```

```
--insert data to Request create by CHAVINDEE M.A. P
INSERT INTO Request VALUES('R001','2023/09/30','Emergency repairs');
INSERT INTO Request VALUES('R002','2023/10/01','Emergency repairs');
INSERT INTO Request VALUES('R003','2023/10/03','Equipment repairs');
INSERT INTO Request VALUES('R004','2023/10/05','Emergency repairs');
INSERT INTO Request VALUES('R005','2023/09/29','Equipment repairs');
```

```
--insert data to SummaryReport create by PERERA T.M. S
INSERT INTO SummaryReport VALUES ('001','User Account summary report','2023/10/01');
INSERT INTO SummaryReport VALUES ('002','User Payment monthly report','2023/10/03');
INSERT INTO SummaryReport VALUES ('003','Ticket issuing report','2023/10/05');
INSERT INTO SummaryReport VALUES ('004','Equipment issue report','2023/10/03');
INSERT INTO SummaryReport VALUES ('005','Repairs issue report','2023/10/02');
```

```
--insert data to maintainTechicia_request create CHAVINDEE M.A. P
INSERT INTO maintainTechicia_request VALUES ('M0001','R001');
INSERT INTO maintainTechicia_request VALUES ('M0002','R002');
INSERT INTO maintainTechicia_request VALUES ('M0003','R003');
INSERT INTO maintainTechicia_request VALUES ('M0004','R004');
INSERT INTO maintainTechicia_request VALUES ('M0005','R005');
```

```
--insert data to parkingSpace create by  BASNAYAKA W.B.M.D. K
INSERT INTO parkingSpace VALUES ('50012','1');
INSERT INTO parkingSpace VALUES ('50003','1');
INSERT INTO parkingSpace VALUES ('50005','1');
INSERT INTO parkingSpace VALUES ('50006','2');
INSERT INTO parkingSpace VALUES ('50009','1');
```

```
--insert data to systeManager create by THARINDI W.A. K
INSERT INTO systeManager VALUES
('SM001','W.A.K.Tharindi','smoo1@T01','Lashan','001','50012');
INSERT INTO systeManager
VALUES('SM002','S.D.N.Vindana','smoo2@T02','kaveesha','002','50003');
INSERT INTO systeManager VALUES
('SM003','A.A.H.Pramuditha','smoo3@T03','pasidu','003','50005');
INSERT INTO systeManager VALUES ('SM004','S.A.D.Sasanga
','smoo4@T04','Sasanga','004','50006');
INSERT INTO systeManager VALUES
('SM005','A.G.I.Malshika','smoo5@T05','Malshika','005','50009');
```

```
--insert data to systeManager_phone create by THARINDI W.A. K
insert into PhoneNo_systeManager values('SM001','0761758678');
insert into PhoneNo_systeManager values('SM002','0702172567');
insert into PhoneNo_systeManager values('SM003','0772477309');
insert into PhoneNo_systeManager values('SM004','0702477300');
insert into PhoneNo_systeManager values('SM005','0752407909');
```

```
--insert data to final report create by THARINDI W.A. K
insert into finalReport VALUES('2789','2023/10/10','W.A.K.Tharindi');
insert into finalReport VALUES('1345','2023/10/10','S.D.N.Vindana');
insert into finalReport VALUES('7893','2023/10/10','A.A.H.Pramuditha');
insert into finalReport VALUES('1234','2023/10/10','S.A.D.Sasanga');
insert into finalReport VALUES('5678','2023/10/10','A.G.I.Malshika');
```

```
--insert values for Security_Officer create by  BASNAYAKA W.B.M.D. K
INSERT INTO Security_Officer VALUES('S0001','Kumara
Perera','Balangoda22','kumaraperera@gmail.com');
INSERT INTO Security_Officer VALUES('S0002','Kushan
Alwis','rathnapura25','kushanalwis@gmail.com');
INSERT INTO Security_Officer VALUES('S0003','Rohan
Fernando','palmadulla24','rohanfernando@gmail.com');
INSERT INTO Security_Officer VALUES('S0004','Lakmal
Gamamge','Ahaliyagod22','lakmalgamamge@gmail.com');
INSERT INTO Security_Officer VALUES('S0005','Thushara
Fonseka','Horana20','thusharafonseka@gmail.com');
```

```
--insert values for Complain table create by  BASNAYAKA W.B.M.D. K
INSERT INTO Complain VALUES('C001','2023/10/23','In my car headlight is damaged','US01');
INSERT INTO Complain VALUES('C002','2023/10/27','In my car right side mirro is
missing','US02');
INSERT INTO Complain VALUES('C003','2023/10/31','In my car wiper is not working
','US554');
INSERT INTO Complain VALUES('C004','2023/09/28','In my car breake light is
damaged','US550');
INSERT INTO Complain VALUES('C005','2023/10/01','In my left side front door is damaged
','US500');
```

```
--insert values for Security_Officer_phone create by  BASNAYAKA W.B.M.D. K
INSERT INTO  Security_Officer_phone  VALUES('1','0778890789','S0001');
INSERT INTO  Security_Officer_phone  VALUES('2','0757678790','S0002');
INSERT INTO  Security_Officer_phone  VALUES('3','0753432456','S0003');
INSERT INTO  Security_Officer_phone VALUES('4','0789089789','S0004');
INSERT INTO  Security_Officer_phone VALUES('5','0709089089','S0005');
```

7. Performance Requirement

- A major role is played by Performance Requirements to make the system successful. They are as follow,
- For a registered user to access the system without any disruption, it must be always operational, 365 days a year.
- A Registered User only needs to input their login information to access the system numerous times.
- The page loads and the login process must be completed in a few seconds.
- The system's performance requirements include speed and usability.
- Users who have registered can see parking details.
- A registered user can change or remove account information.
- The administrator can manage user accounts, add or remove shared experiences and feedback, and change or update parking details.
- The website's developer can add additional functionality.
- Design user friendly user interface.
- The system allows the administrator to control employee accounts.
- Any device or browser must be able to access the website at any time for users.

8.Security Requirements

• User personal information ought to be encrypted before being sent to the database.

• The system's data can only be accessed and changed by administrators.

• Restricted functions ought to be inaccessible to unauthorized users.

• There should only be one user account per email address.

• Redundant servers are required for database maintenance.

• The user account password needs to be a strong password that consists of capital, lowercase, digits, and special characters.

• Every piece of data in the system should be backed up in the database.