**DATABASE DESIGN PROJECT REPORT**

**CAR RENTAL SYSTEM**

**INDEX**

1. Introduction

2. Entities

3. Relations

4. E-R Diagram

5. Functional Dependencies

6. Normalization

7. Relational schema

8. SQL Statements

9. Conclusion

++++++++

**1. INTRODUCTION**

We have chosen to produce a Car Rental system. In our system, Customer can rent a car based on make and a model. Our system provides customer to have different pick-up and drop off locations and will impose late fee if the rental car is returned beyond the return date and time. The Customers can purchase car rental insurance which is optional and can use upto one discount coupon to their final bill. Customers who have membership will be by default given a 10% discount in their final bill. We will see detailed description below.

**1.1 REQUIREMENTS**

a) Car rental agency should have collection of cars.

b) Each car should belong to a particular Car Category and each car will belong to a particular location.

c) Customer, based on his location and car category preferences, rents a car. d) Based on his location and car category preferences, list of cars available to rent will be shown along with available date and time (from and to).

e) Customer will select a car from the suggestions and should be able to reserve it for rent. f) When a customer reserves a car, he/she should be able to optionally purchase a Car Insurance Plan and should be able to apply at most one discount code.

g) If a customer is also a member of the car rental agency and has a membership ID then he/she will be given a default 10% discount in additional to the discount code applied. Therefore the total discount percentage will be 10 plus the discount percentage given by the discount code applied.

h) Billing is generated when a car is returned.

i) Customer can return the car before the due date, on the due date or he/she can return it late also.

j) If a customer returns a car after the due date, additional late fee is calculated and added to the bill.

k) A default 8.25% tax is applied on the amount which also includes the late fee and this tax is added to obtain the total amount to which the discount will be applied and a final amount is obtained.

l) Once the car is returned it becomes available for the booking.

m) A booking can be cancelled until 5 days before the actual pick up.

n) Company may have several discount plans like weekend discount, corporate discount etc.

o) Car price will be calculated based on the selected make and model.

**2. ENTITIES**

**a) Customer:**

Customer will be the one who is using car rental system for reserving a car. Customer entity will store details like customer driving license number, email, address, name, and phone number.

**b) Car:**

Car entity will have list of cars available in the system. Each car will be associated with a car category and car will have attributes like make, model, mileage and registration number. Car will also have separate flag to check the availability of the car.

**c) Car Category:**

Every car has a car category. Price is calculated based on the car category. Car category will have attributes like no of person, no of luggage’s, name, and cost per day and late fee per hour.

**d) Location**

Location entity here denotes the pickup and drop off location of the car. Customer can pick up the car from the particular location and can have same or different drop off location. Location will have attributes like Location id, name and address.

**e) Booking**

Each car reservation will be monitored in the entity called booking. Booking will have attributes like booking id, from date and time of booking and due return date and time and actual return date and time of the booking, and booking status. This booking amount might also include rental insurance and discount code.

**f) Billing**

When a customer returns a car, a bill will be generated on the particular booking. Billing have attributes like Bill ID, bill date, bill status, total late fee, tax amount, and total amount.

**g) Discount**

Customer can apply discount code while the bill is generated. Each discount code has different discount percentage. Discount will have attributes like discount code, name, expiry date and discount percentage.

**h) Car Rental Insurance**

Customer may already have car rental insurance or can buy one while booking the car. Car rental insurance will have attributes like insurance code, coverage type, name and cost per day.

**3. RELATIONS**

**a) Car to Car Category:**

Every car is associated with a car category. Once customer selects a car, the cost per day is obtained from the car category that the selected car belongs to. The relation name is ‘Belongs to’.

**b) Car to Location:**

Customer will be picking up or dropping the car in a particular location. Customer can pick up or drop-off the car at the particular location. So, cars will be present at a location. The relation name is ‘Current location’

**c) Booking to Billing:**

Once customer returns a car bill will be generated for each booking. There can be case like booking is cancelled in that case no bill will be associated with the booking. The relation name is ‘Gives’.

**d) Booking to Discount:**

Customer may apply a discount code when he/she books a car. This discount will be applied to the total amount after tax and late fee while the bill is generated. Based on the discount code total amount will be reduced by some percentage. The relation name is ‘Has’.

**e) Booking to Car Rental Insurance:**

Customer can select rental insurance while booking a car so that rental insurance will cover damages based on the coverage type. The relation name is ‘Includes’.

**f) Booking to Location:**

Customer can pick a car for rent from a particular location. The relation name is ‘Pick up location’.

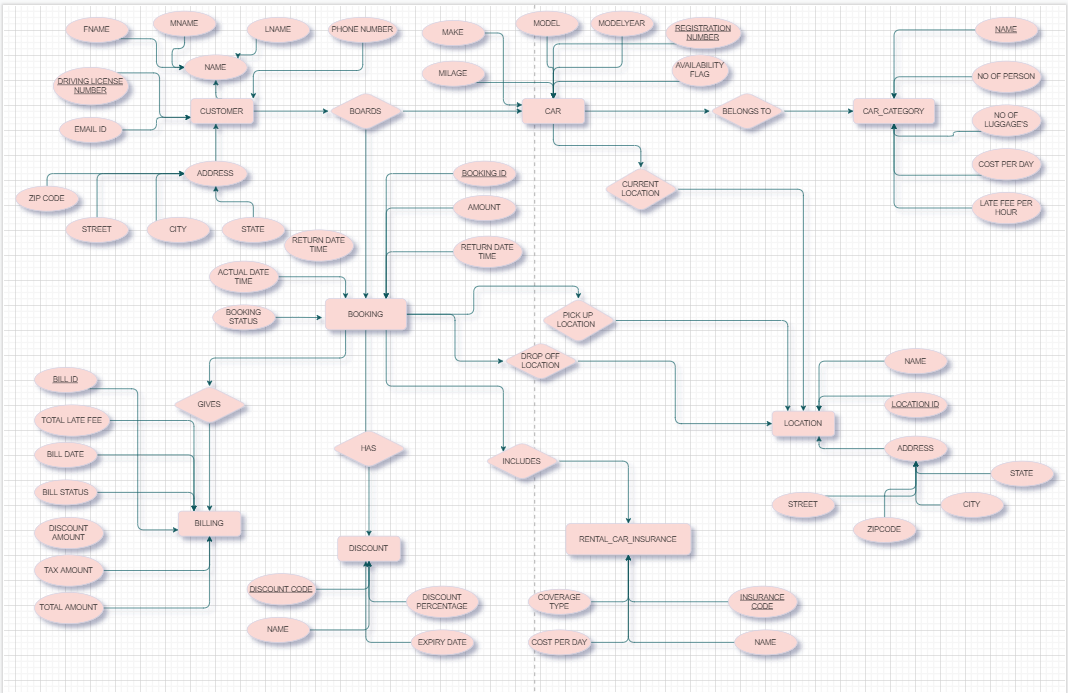
**g) Booking to Location:**

Customer can drop off rental car in a particular location. The relation name is ‘Drop off location’.

**h) Customer to Car to Booking:**

Customer will select car for rent. So the customer will be related to the both car and the booking. The relation between these 3 entities is a ternary relation and the relation name is ‘Rents’.

**4. ER DIAGRAM**



**5. FUNCTIONAL DEPENDENCIES**

**a) Customer\_Details Relation:**

∙ DL\_number -> Fname, Mname, Lname, Phone\_number, Email\_id, Street, City, State, Zipcode,

∙ Zipcode -> State,City

**b) Car Relation:**

∙ Registration\_number -> Model, Make, Model\_year, Car\_category\_name, Loc\_id, Mileage, Availability\_flag

∙ Model -> Make

**c) Car\_Category Relation:**

∙ Category\_name -> No\_of\_luggage, No\_of\_person, Cost\_per\_day,

Late\_fee\_per\_hour

**d) Location \_Details Relation:**

∙ Location\_id -> Name,Street,City,State,Zipcode

∙ Zipcode -> State,City

**e) Booking\_Details Relation:**

∙ Booking\_id -> From\_dt\_time, Ret\_dt\_time, Amount, Booking\_status, Pickup\_loc, Drop\_loc, Reg\_num, DL\_num, Ins\_code, Act\_ret\_dt\_time,

Discount\_code

**f) Billing\_Details Relation:**

∙ Bill\_id -> Bill\_date, Bill\_status, Discount\_amt, Total\_amt, Tax\_amt, Booking\_id, Total\_late\_fee

**g) Discount\_Details Relation:**

∙ Discount\_code -> Discount\_name,Expiry\_date,Discount\_percentage ∙ Discount\_name -> Discount\_code ,Expiry\_date,Discount\_percentage

**h) Rental\_Car\_Insurance Relation:**

∙ Insurance\_code -> Insurance\_name,Coverage\_type,Cost\_per\_day ∙ Insurance\_name -> Insurance\_code ,Coverage\_type,Cost\_per\_day

6. **Functional dependencies that violated Normalization rules**

∙ **Customer\_Details Relation**

o DL\_number -> Zipcode

o Zipcode -> State, City

∙ **Car Relation**

o Registration\_number -> Model\_name

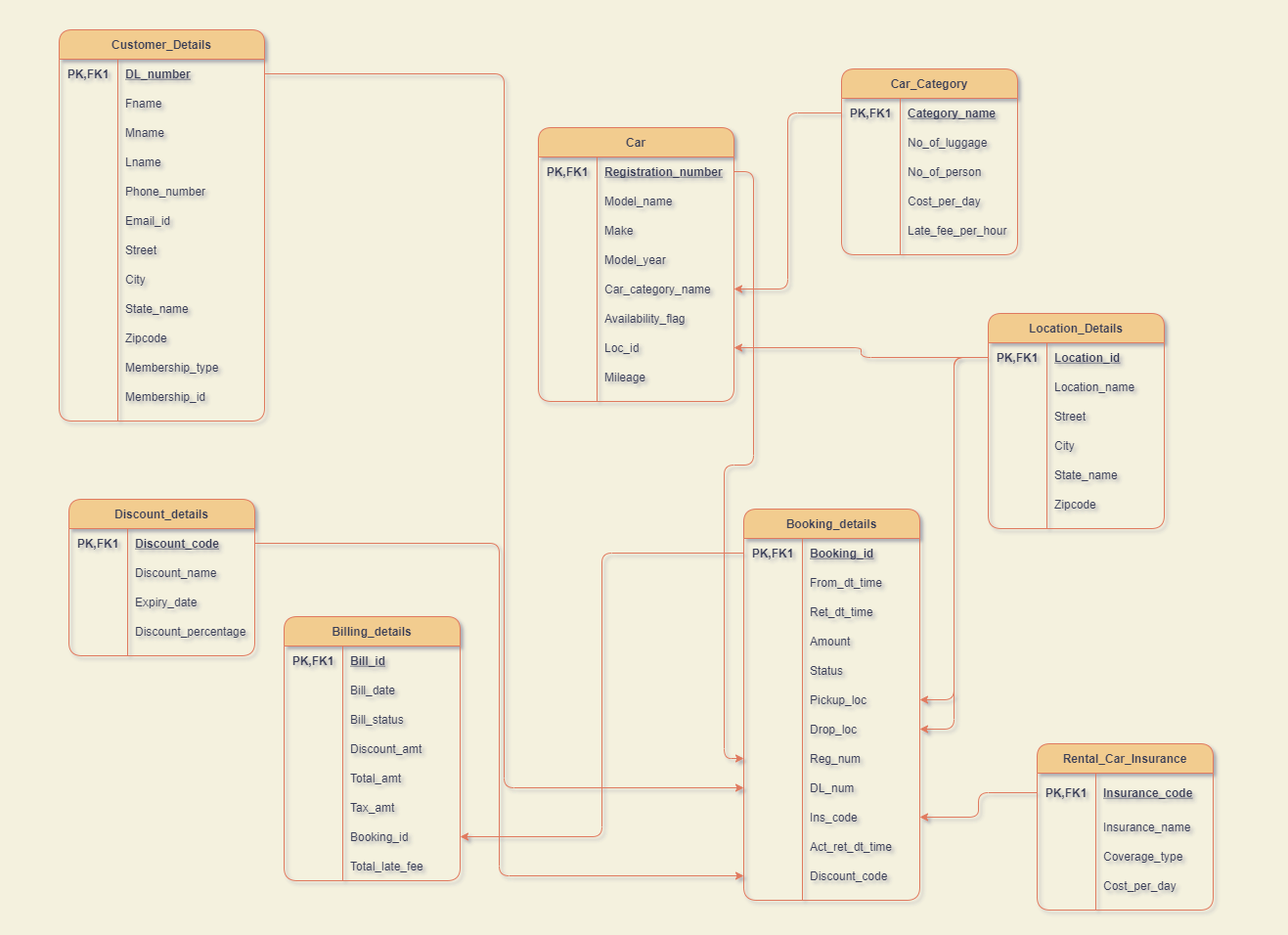
o Model\_name -> Make

∙ **Location\_Details Relation**

o Location\_id -> Zipcode

o Zipcode -> State, City

**7.RELATIONAL SCHEMA**

****

**8. SQL STATEMENTS**

**8.1 Create table Statements**

a) **Customer\_Details**

CREATE TABLE CUSTOMER\_DETAILS

( DL\_NUMBER CHAR(8) NOT NULL,

FNAME VARCHAR(25) NOT NULL,

MNAME VARCHAR(15),

LNAME VARCHAR(25) NOT NULL,

PHONE\_NUMBER NUMBER(10) NOT NULL,

EMAIL\_ID VARCHAR(30) NOT NULL,

STREET VARCHAR(30) NOT NULL,

CITY VARCHAR(20) NOT NULL,

STATE\_NAME VARCHAR(20) NOT NULL,

ZIPCODE NUMBER(5) NOT NULL,

MEMBERSHIP\_TYPE CHAR(1) DEFAULT 'N' NOT NULL, MEMBERSHIP\_ID CHAR(5),

CONSTRAINT CUSTOMERPK

PRIMARY KEY (DL\_NUMBER)

);

**b) Car\_Category**

CREATE TABLE CAR\_CATEGORY

( CATEGORY\_NAME VARCHAR(25) NOT NULL,

NO\_OF\_LUGGAGE INTEGER NOT NULL,

NO\_OF\_PERSON INTEGER NOT NULL,

COST\_PER\_DAY NUMBER(5,2) NOT NULL,

LATE\_FEE\_PER\_HOUR NUMBER(5,2) NOT NULL, CONSTRAINT CARCATEGORYPK

PRIMARY KEY (CATEGORY\_NAME)

);

**c) Location\_Details**

CREATE TABLE LOCATION\_DETAILS

( LOCATION\_ID CHAR(4) NOT NULL,

LOCATION\_NAME VARCHAR(50) NOT NULL,

STREET VARCHAR(30) NOT NULL,

CITY VARCHAR(20) NOT NULL,

STATE\_NAME VARCHAR(20) NOT NULL,

ZIPCODE NUMBER(5) NOT NULL,

CONSTRAINT LOCATIONPK

PRIMARY KEY (LOCATION\_ID)

);

**d) Billing\_Details**

CREATE TABLE BILLING\_DETAILS

( BILL\_ID CHAR(6) NOT NULL,

BILL\_DATE DATE NOT NULL,

BILL\_STATUS CHAR(1) NOT NULL,

DISCOUNT\_AMOUNT NUMBER(10,2) NOT NULL,

TOTAL\_AMOUNT NUMBER(10,2) NOT NULL,

TAX\_AMOUNT NUMBER(10,2) NOT NULL,

BOOKING\_ID CHAR(5) NOT NULL,

TOTAL\_LATE\_FEE NUMBER(10,2) NOT NULL,

CONSTRAINT BILLINGPK

PRIMARY KEY (BILL\_ID),

CONSTRAINT BILLINGFK1

FOREIGN KEY (BOOKING\_ID) REFERENCES BOOKING\_DETAILS(BOOKING\_ID) );

**9.CONCLUSION**

During the course of this project, we learnt a lot of the work and best practices that go into creating a database, the rules to construct a good ER diagram, How to come up with relational schema mapping from the ER diagram, deriving the functional dependencies and how to normalize the relational schema. We learnt on how to design a system from Database perspective and how to efficiently store and manipulate data.