**Ahmed Salman** 109363909

**Loureen Viloria** 109068508

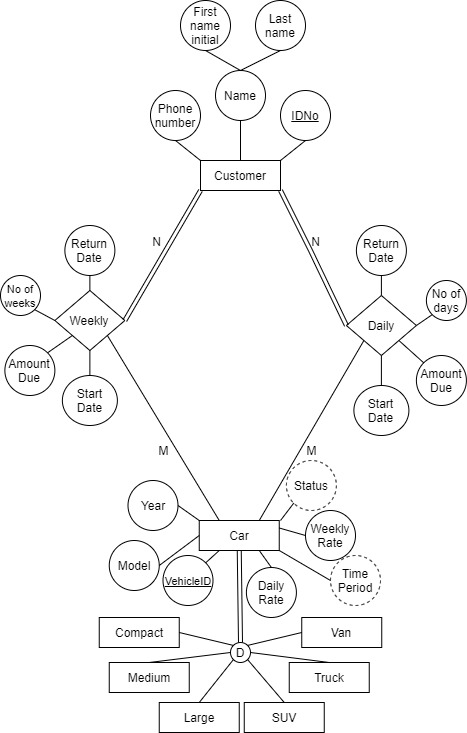
Homework 2 part 2

May 5, 2020

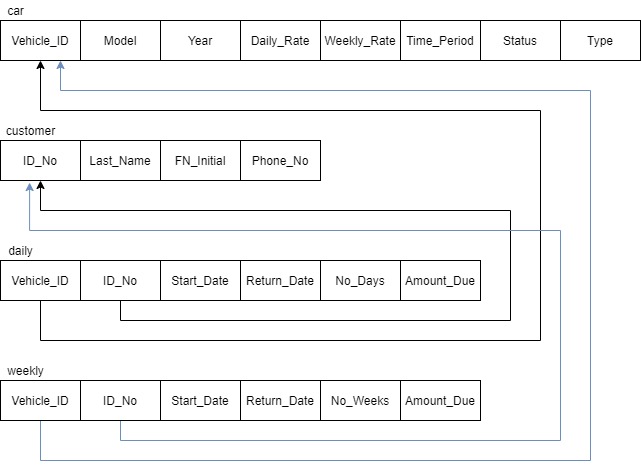
**Part 2.1:**

knowing that Part1 was kept as it was, for this part, we did the following:

(a) Mapped the EER schema design to a relational database schema, as shown below.



(b) Created the tables corresponding to the relational schema using MySQL.



(c) Specified all constraints needed as shown above.

**Part 2.2:**

The following are the tasks that have been achieved for the second part of the project:

**2.2.1** We created a database named carrentaldb, it contains four tables which

are Car, Customer, Daily, Weekly

**2.2.2** We Loaded some initial data into the database tables. The data are kept in a csv file so that it can easily be reloaded during the debugging.

**2.2.3** We Wrote some queries to retrieve all the data that we entered, such as:

**addCar.php**

$sql = "INSERT INTO car (Model, Year, Daily\_Rate, Weekly\_Rate, Time\_Period, Status, Type) VALUES ('**$model**', '**$year**', '**$dRate**', '**$wRate**', '**$tPeriod**', '**$status**', '**$type**')";

**createRental.php**

$sqlRental = "INSERT INTO daily (Vehicle\_ID, ID\_No, Start\_Date, Return\_Date, No\_Days) VALUES ('**$vehicleNum**', '**$lasCustID**','**$carStartDate**', '**$carEndDate**', '**$dateDiff**')";

$availCar = "SELECT \* FROM car WHERE Status = 'Available'";

**returnRental.php**

$sqlReturn = "SELECT daily.\*, customer.\*, weekly.\* FROM customer, daily, weekly WHERE customer.Last\_Name = '**$rLName**' AND customer.ID\_No = '**$rID**' AND daily.ID\_No = '**$rID**' or weekly.ID\_No = '**$rID**'";

$sqlCarReturn = "UPDATE car SET Status = 'Available' WHERE Vehicle\_ID = '**$vehicleID**'";

**updateCar.php**

$sqlUp = "UPDATE car SET Daily\_Rate = '**$updatePrice**', Weekly\_Rate = '**$updateWeeklyPrice**' Where Type = '**$cType**'";

**addCustomer.php**

$sql = "INSERT INTO customer (FN\_Initial, Last\_Name, Phone\_No) VALUES ('**$fInitial**','**$lname**','**$phone**')";

**2.2.4** We have created a menu which contains all the transaction available, such as add a new customer, add a new car, update car, create rental, update rental, return rental. The menu has been created using switch-statement with many cases, each case handles a transaction. Implementation is shown in the attached Index.php file.

**2.2.5** We Wrote the following database update transactions using PHP programming language to do the following:

**(a) Adding a new Customer**

Once AddCustomer is selected, user will be asked to enter new CUSTOMER information through a user-friendly interface. Implementation is shown in the attached addCustomer.php file.

**(b) Adding a new Car**

Once AddCar is selected, user will be asked to enter new CAR information through a user-friendly interface. Implementation is shown in the attached addCar.php file.

**(c) Adding information regarding a new RENTAL reservation.**

To get this function worked, we have used if-statement along with while-loop, so that it easy to check which car is available and which is not. However, the SQL query $sqlRental = "INSERT INTO weekly (Vehicle\_ID, ID\_No, Start\_Date, Return\_Date, No\_Weeks) VALUES ('**$vehicleNum**', '**$lasCustID**','**$carStartDate**', '**$carEndDate**', '**$dateDiff**')" is not working and is unable to add the information of the weekly rental, but the daily rental is working. Implementation is shown in the attached createRental.php file.

**(d) Handling the return of a rented car.**

The code is working properly, once the user inputted the information the code will retrieve all information about the user. Once the user hits submit, it will show the rental information of the user and the amount due, it will also show a message “Car Returned Successfully!” and the car’s status will change from “Unavailable” to “Available”. I used the information from the daily and weekly tables as an example. Implementation is attached in returnRental.php file.

**(e) Updating the rental rates (daily and weekly) for a type of car.**

This function has been done using switch-statement along with if-statement, user must make the choice of which rate needs to be updated, weekly or daily. Implementation is attached in updateCar.php file.

**2.1.6** For each transaction we have set a user-friendly interface to inter the information needed for each transaction. The user needs to click submit for the code and database to retrieve, update and insert data.

**2.1.7** Transactions have been tested by adding a few new customers, cars, reservations, and by changing some rental rates and reservations rates.

**Hint**: We have created a pseudo code which works for all transaction to make sure that

all the fields or required fields are filled.