MURANG'A UNIVERSITY OF TECHNOLOGY

Project Documentation Report For:

SC212/1418/2017

ODHIAMBO THOMAS ATIENO

SUPERVISOR: TIRUS MUYA

Submitted in fulfillment of the requirements of

SCS 203 Programming and Database Practicum

Project Title: Neo Car Rentals Management System

INTRODUCTION

Neo Car Rentals is a web-based system which will purposefully be used in online car rental services.

The system is brought up through the use of PHP programming language.

Due to the continuous problems with the current manual ordering system which has brought a lot of inconveniencies to Neo Car Rentals company hindering it from performing best as expected, the company management staff thought and found it clear that computerization of the system will ensure efficiency of the car renting process.

Proper information gathering was carried out, close analysis of samples collected during the problem definition stage, the management team realized that all the hardware and software support needed for the implementation, running and maintenance of this new system are available and affordable by Neo Car Rentals.

Much considerations to ensure maximum efficiency of the system have been made.

This system helps the company management and its staff members to manage the car rental process with high efficiency and accuracy thus making them obtain highest profits from the use of this system.

EXPLANATION OF CONTENTS

ABSTRACT

My main aim is to design and create a data management system for a car rental company. This will enable a customer to rent a vehicle for use which will be confirmed by the system administrator after paying money during a specified time period. This system will increase customer retention and simplify vehicle management in an efficient way.

This system has a very friendly user interface. Thus, users will find it very easy to use. Through this system, the admin can manage their rental. Car information can be added into the system or an existing information can be edited or deleted too by the system admin, transaction reports are also available hence no delay in the availability of car information.

ACKNOWLEDGEMENT

I would like to express my sincere gratitude to my lecturer Tirus Muya who offered me a golden opportunity to do this wonderful project on car rental system, and who has been always there to give advice when necessary. This also helped me in doing a lot of research that I came to know so many things.

Secondly, I would like to thank my parents, siblings and friends who helped me a lot in coming up with this project.

TABLE OF CONTENTS

Contents

CHAPTER 1: INTRODUCTION 6

1.1 Background 6

1.2 Objectives 6

1.3 Purpose, Scope and Applicability 7

1.3.1 Purpose 7

1.3.2 Scope 7

1.3.3 Applicability 7

CHAPTER 2: SURVEY OF TECHNOLOGIES 7

Advantages Of PHP 8

Disadvantages of PHP 8

Relational database management systems (RDBMS) 8

Advantages of MySQL 9

CHAPTER 3: REQUIREMENT ANALYSIS 9

3.1 Problem Definition 9

3.2 Requirement Specification 10

i) Functional Requirements 10

ii) Non-Functional Requirements 10

3.2.1 Hardware Requirements. 10

3.2.2 Software Requirements 11

3.3 Preliminary Product Description 11

3.4 Conceptual Models 12

System Flowchart 12

System ER Diagram 12

System Use Case Diagram 14

Login 14

Reservation 14

Update car 15

Add Car 15
View Available cars 16
Registration 16
Level 0 Data Flow Diagram 18
Level 1 Data Flow Diagram 19
Sequence Diagram 19
LOGIN ..... **Error! Bookmark not defined.**
Reservation 19
View Available Cars 20
Reservations 20
Registration 21
Update car 21
CHAPTER 4: SYSTEM DESIGN 22
BASIC MODULES 22
4.2 Security 39
4.3 Test Cases Design 39
CHAPTER 5: IMPLEMENTATION AND TESTING 43
5.1 Implementation Approaches 43
5.2 Testing 43
CHAPTER 6: CONCLUSIONS 46
6.1 Conclusion 46
6.2 Limitations of the system 46
REFERENCES 47

TABLE OF FIGURES

System Flowchart Figure 1 12
System ER Diagram Figure 2 12
Use Case Diagram Figure 3 17
Level 0 Data Flow Diagram Figure 4 18
Login Sequence Diagram Figure 5 19
Reservation Sequence Diagram Figur-+++++e 6 19

View Available Cars Sequential Diagram Figure 7 20
Reservations Sequential Diagram Figure 8 20
Registration Sequence Diagram Figure 9 21
Update car Sequence Diagram Figure 10 21
User Login Figure 11 22
Login code screenshot Figure 12 23
Search Car Result Screenshot Figure 13 24
Search Car Code Screenshot1 Figure 14 24
Search Car Code Screenshot2 Figure 15 25
Search Car Code Screenshot3 Figure 16 25
Search Car Code Screenshot4 Figure 17 25
Booking Module Screenshot1 Figure 18 26
Booking Module Screenshot2 Figure 19 27
Booking Code Screenshot Figure 20 27
Admin Dashboard Module Screenshot Figure 21 28
Admin Dashboard Code Screenshot1 Figure 22 28
Admin Dashboard Code Screenshot2 Figure 23 29
Admin Dashboard Code Screenshot3 Figure 24 29
Post New vehicle Module Screenshot1 Figure 25 30
Post New Vehicle Module Screenshot2 Figure 26 30
Posting New Vehicle Code Screenshot1 Figure 27 31
Posting New Vehicles Coode Screenshot 2 Figure 28 31
Posting New Vehicle Code Screenshot3 Figure 29 32
Manage Existing Vehicles Module Screenshot Figure 30 32
Delete Vehicle Code Screenshot Figure 31 33
Edit Vehicle Details Code Screenshot Figure 32 33
Edit vehicle details Code Screenshot2 Figure 33 34
Manage Bookings Module Figure 34 34
Manage Bookings Code Screenshot1 Figure 35 35
Manage Bookings Code Screenshot2 Figure 36 35
Manage Bookings Code Screenshot3 Figure 37 36
User Profile Screenshot1 Figure 38 36
User Profile Screenshot2 Figure 39 37
User Profile Screenshot 3 Figure 40 37
User Profile Code screenshot1 Figure 41 38
User Profile Code Screenshot2 Figure 42 38
Admin Profile Screenshot Figure 43 39
Code for admin profile Figure 44 39
User Signup Figure 45 44
Registration Success Message Figure 46 45
Posting a vehicle Figure 47 45
Post vehicle Success Message Figure 48 46

CHAPTER 1: INTRODUCTION

1.1 Background

In an organization running a car rental service, the process of searching client details is slow if a manual system is being used and there is a huge number of clients to serve. Besides that, booking has to be recorded manually which is very difficult to produce timely reports. A manual system does not allow client to look online and it's also hard to keep track on the record of rental cars.

A major drawback of the manual system is that the client has to meet the car owner in order to rent or to confirm whether the car is available which is time wasting.

Disadvantages Of existing system

a) User should manually go and look for the car.
b) It is a time taking process and cost also.
c) Doesn't properly fulfill client requirements.

Proposed System advantages

a) The user can directly interact though this system to book a car online and so saves time.
b) The car rental company can also maintain their schedules effectively and be able to generate reports wherever needed.
c) Data is centralized which helps to solve problems experienced in previous systems.
d) It acts as an office open 24/7

1.2 Objectives

The main objective of this study is to design and implement a car rental system for a car rental company with main objectives being:

a) To develop a simple and secure system that protects clients' information and organization's confidential information.
b) To develop a system that stores bookings and reservation information as well as payment history to help the organization keep track of transactions.
c) To design a system that enables clients pay their car rent online.
d) To design a user-friendly system that enables client check for availability of vehicle and book or reserve the vehicle online.

1.3 Purpose, Scope and Applicability

1.3.1 Purpose

The purpose of this system is to find out a solution to reduce operating costs of a company. It is developed to handle the business needs of renting out vehicles to customers, maintaining records and data on vehicle fleet, operating the customer portal website, and reporting the state of the system to the company. Apart from this, the system does not fulfil any other need of the business.

1.3.2 Scope

This system is for use by only the car rental company that wants to bring their services online to their clients. Therefore, it will not be applicable to those companies that do not have the need to reach their clients online.

Customer charges will be made according to terms and conditions specified.

The different features of this car rental system include:

Managing details of vehicles.

Management of client details.

Producing timely (Daily, weekly and monthly) reports.

Managing vehicle details.

Booking vehicles and cancellation of vehicle bookings.

Posting of testimonials by clients.

1.3.3 Applicability

This system enables easy online vehicle booking hence customers find it easy to make their bookings.

There is low estimated cost of developing the system.

The system allows easy and fast access to car bookings and provides better management facilities. Security is also enhanced by the system to protect user and the car rental company's information therefore reduced data security threats.

This system will help reduce congestion of customers who want to rent vehicles in the booking office.

CHAPTER 2: SURVEY OF TECHNOLOGIES

To bring out this system, a group of technologies has been combined to ensure it worked out and these are the technologies applied.

The system is developed using PHP (Hypertext Pre-Processor) programming language with the help of MySQL database.

PHP is a server scripting language used to create websites and web applications.

PHP was introduced by Rasmus Laird in 1994 and by 2015 it had been installed in more than 2.1 million servers with approximately 240 million websites.

Advantages Of PHP

Ease of Use: any individual in programming can learn to use within a very short period of time. It also has a similar syntax to C language.

Cross platform: Programs on php can run on various platforms. Php code can run on all operating systems.

Speed: In web development, speed is vital and therefore websites made in php load faster which gives users a positive taste of the site.

Open source and powerful library support: PHP is developed and maintained by a cluster of developers which helps in making support community, extensive extension library. PHP additionally has immense collection of functional modules and a few of modules in PHP include graphics and pdf etc.

Stable: PHP has been in existence for almost 25 years and over these years' developers have worked on the application to enhance usage of the application. Lots of bugs are discovered over the years and corrected.

Disadvantages of PHP

Weak type: due to the way PHP was created, it will be difficult to use it in programming large applications. Since it is not extremely modular, large applications created out of the language are difficult to keep. Furthermore, it is an excellent language for web scripting and developing small applications.

Conclusion: PHP is one of the most popular programming languages. It is currently used by large number of in creating numerous types of applications. It is primarily used as a server-side scripting for websites.

Relational database management systems (RDBMS)

This is a RDBMS based project which uses MySQL for all the transaction statements. MySQL is an open source RDBMS.

A relational database management system is a database management system based on the relational model invented by E.F Codd, of IBM's Jose research Laboratory. Many popular databases are currently based on the relational database model.

RDBMS became a predominant choice for the storage of information in new databases used for financial records, manufacturing and logistical information, personnel data, and much more since 1980's. They have often replaced legacy hierarchical databases and network databases because they are easier to understand and use. However, relational databases have been challenged by Object databases, which were introduced in attempt to address the object-relational impedance mismatch in relational database, and XML databases.

Advantages of MySQL

Maturity: MySQL is a setup database, implying that there is a large network, broad testing and a little bit of strength.

Compatibility: MySQL is accessible for every single real stage including Linux, Windows, Mac, BSD and Solaris. It additionally has connectors to languages like Node.js, Ruby, C#, C++, Java, Perl, PHP, implying that it is not constrained to SQL query language.

Cost Effective: The database is open source and free.

Replicable: MySQL database can be mirrored across various nodes, implying that the work load can be decreased and the scalability and accessibility of the application cab be expanded.

Sharding: While sharding is impossible in most SQL databases, it is possible in MySQL servers which is useful and cost-effective

CHAPTER 3: REQUIREMENT ANALYSIS

3.1 Problem Definition

By definition, a car rental is that vehicle which can be used temporarily only after a fee for a specified limit of time. This helps people around the world even if they do not have their own cars or do not have access to their vehicles. An individual who needs a car has to contact car rental company and contract out for a vehicle. This leads to increased customer retention and simplify vehicle and staff management.

The manual system has not been able to realize its maximum potential due to huge losses, suffered frequent errors, delay of activities.

Neo Car Rentals has opted for computerized system which involve online car booking.

This system will be: -

- Friendly to staff and Customers.
- Increase the company performance.
- Reduce operational costs of the company.
- Enable fast and easy retrieval of car records.
- Enable Online booking by the internet.

3.2 Requirement Specification

i) Functional Requirements

These are the requirements used to illustrate the internal working nature of the system, the description of the system, and explanation of each subsystem. In our system the functional requirements are:

- **Customers' Registration:** The system should allow new users to register online.
- **Online reservation of Cars:** customers should be able to use the system to make bookings and online reservation.
- **Automatic update to database once reservation is made or new customer registered:** Whenever there's a new reservation or new registration, the system should be able to update the database without any additional efforts from the admin.
- **Feedback from the customers:** It should provide means for the customers to leave feedback**.**

ii) Non-Functional Requirements

- **Security:** The system should provide a high level of security and integrity of the data held by the system, only authorized personnel of the company can gain access to the company's secured page in the system; only users with valid password and username can login to view user's page.
- **Performance and response time:** The system should a have high performance rate when executing user's input and should be able to provide feedback or response within a short span of 35 seconds for highly complicated task and 10-15 seconds for less complicated tasks.
- **Error handling:** Error should be considerably minimized and an appropriate error message that guides the user recover from an error should be provided. Validation of user's input is highly essential. Also, the standard time taken to recover from an error should be 15 to 20 seconds.
- **Availability:** This system should always be available for access at 24 hours,7 days a week. Also, in the occurrence of any major system malfunctioning, the system should be available in 1 to 2 working days, so that the business process is not severely affected.

3.2.1 Hardware Requirements.

Hard Disk	**15GB**
RAM	**512MB**

Processor **Pentium or Intel.**

3.2.2 Software Requirements

Operating system Windows 7/8/10, Linux, Mac OS X, Solaris.

Programming Language PHP(Backend).

Web Server XAMPP/WAMP/LAMP.

Web Browser: Google Chrome, Firefox, Microsoft Edge, Internet Explorer Or later.

3.3 Preliminary Product Description

Product functions

The car rental management system has the following;

Benefit to organization: The car rental company will obviously be able to gain benefits such as savings in operating cost, reduction in paperwork, better utilization of human resources and more presentable image increasing goodwill.

Initial cost: Initial cost of setting up the system will include the cost of hardware/ software (OS, add on software, utilities) and labor (setup and maintenance). The same has to bear by the organization.

Running cost: This includes the stationary charges, cost for human resources, cost for update/ renewal of various related software.

Training: The users along with the administration need to be trained at the time of Implementation of the system for smooth running of the system. The client adopting the system will provide the training site.

We talked to the management people who were managing all the financial issues of Neo Car Rentals, staff who were keeping the records in lots of the registers and the reporting manager regarding their existing system.

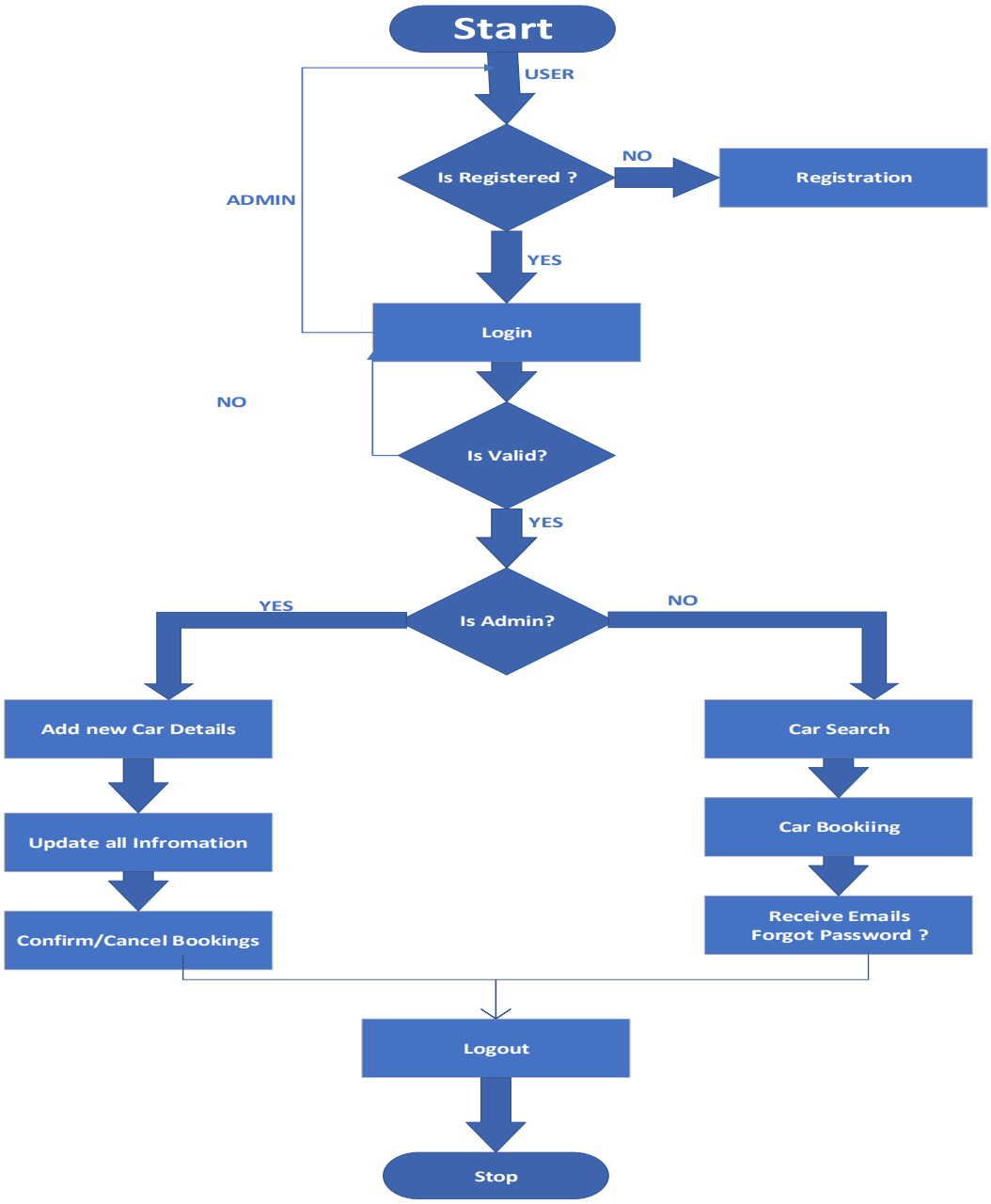
Reliable, accurate and secure data was also considered to be complex without this system. Because there was no such record for keeping track of all activities, which was done by the car rental system on daily basis.

This new system developed by me will ease task of the organization in consideration it will be helpful in generating the required reports by the staff, which will help them to track their progress and services. Thus, it will ease the task of management to a great extent as all the major activities to be performed are computerized through this system.

3.4 Conceptual Models

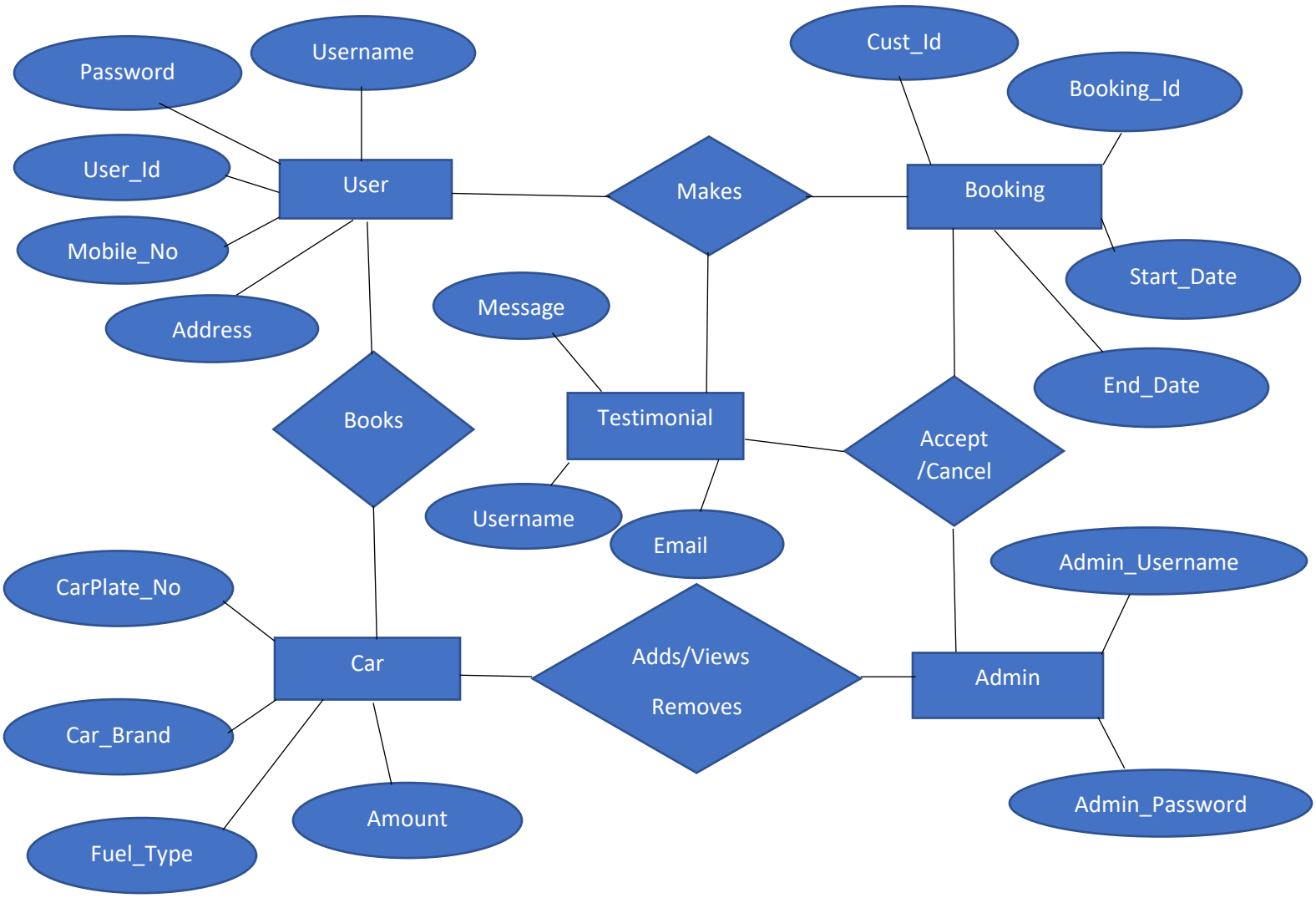
System Flowchart

Figure 1



System ER Diagram

Figure 2


Password
Username
Cust_Id
Booking_Id
User_Id
User
Makes
Booking
Mobile_No
Start_Date
Address
Message
End_Date
Books
Testimonial
Accept
/Cancel
Username
Email
Admin_Username
CarPlate_No
Car
Adds/Views
Removes
Admin
Car_Brand
Amount
Admin_Password
Fuel_Type

System Use Case Diagram

Login

Use case name	Log in
Actors	Admin and Customer
Pre-Condition	Admin/Customer should be registered
Post Condition	User enters the system (Logs in)
Flow of events	1) Open webpage 2) Login screen display 3) Enter username and password 4) Click login
stimulus	User selects sign in
Summary	As soon as the user clicks on sign in they'll have to give their id and password to sign into the system and perform their desired actions. Types of actions are different according to the type of person who is logging in. If the person has entered correct id and password they can log in else a message is displayed "Please enter valid id and password".
Response	Customer should login to account
comment	Login successful

Reservation

Use case name	Reservation
Actors	Customer
Pre-Condition	Customer should be registered, Cars should be available in stock
Post Condition	Customer has finally rented a car
Flow of events	1) Open webpage 2) Fill requirement form 3) Select car 4) Enter personal details
Stimulus	User Reserve a car
Summary	When the customers visit a car rental shop they are accommodated by the admin working for the company who have the authority to rent out the car according to company rules. Admin performs the search for the desired car to check if it is available in stock to be rented out. If it is available, he updates the current rent record and the customer details. If the customer is not coming for the first time his details are updated easily and no new details are required

	else if he is coming for the first time his address and contact number along with other necessary information may be required. After this the customers can take the car for as long as they want and pay on return.
Response	The car has been reserved
Comment	Car should be return within days

Update car

Use case name	Update car
Actors	System Admin
Pre-Condition	Admin is logged in
Post Condition	Car details are updated
Flow of event	1) Enter car details 2) Select car or edit car 3) Update or enter new information
Stimulus	Admin selects to update car
Summary	The Admin can only use their own account to perform this and the details of the car can be modified or changed according to the need of the owner.
Response	Car is updated
Comment	Car details has been updated

Add Car

Use case name	Add Car
Actors	System Admin
Pre-Condition	Admin is logged in
Post-condition	Enter new car
Flow of event	1) Open car menu 2) Select category 3) Enter car details 4) Add car
Stimulus	Admin selects to add a car

Summary	The Admin can only use their own account to do this and need the details of the car when adding such as model, year, etc. Before completing this process, the car needs to be given a unique ID to be searched or recognized. After all this the car is added.
Response	Car is added into stock
Comment	Added car is available in stock

View Available cars

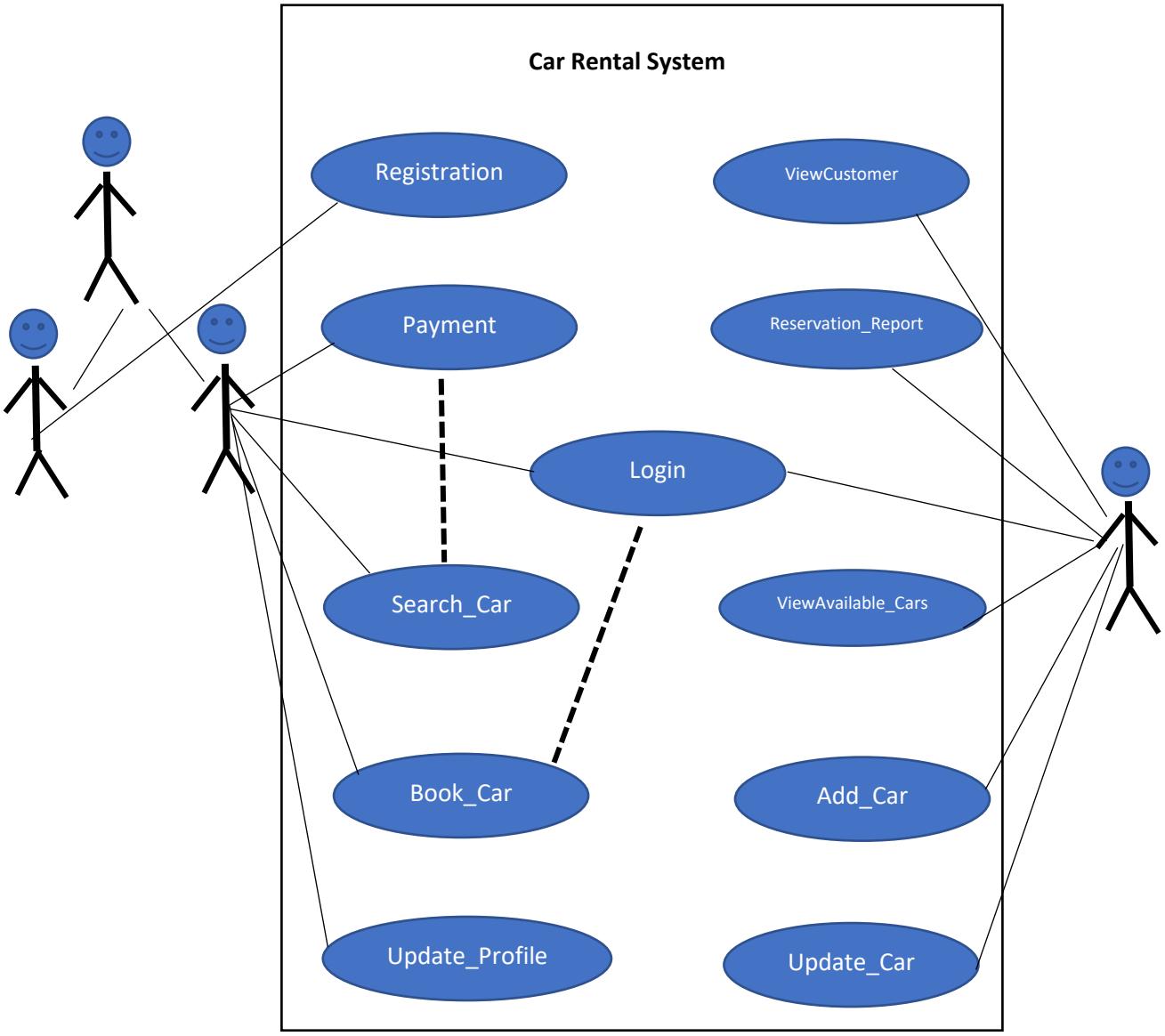
Use case name	View available car(search)
Actors	Admin and Customer
Pre-Condition	Admin and Customer is not necessary to registered
Post Condition	User knows the status of the car
Flow of event	1) Open webpage 2) Enter car details 3) Display cars on screen
Stimulus	User selects to search a car
Summary	When the user wants to check details about the car like if it is currently rented out or in stock it needs to be searched. To search a car the user can perform the operation using car model, year or car ID.
Response	Cars display on the screen
Comment	Cars available

Registration

Use case name	Registration
Actor	Customer
Pre-condition	Customer must enter personal details
Post-condition	Customer registered and have account
Flow of events	1) Open webpage 2) Enter personal details 3) Verified details 4) Register

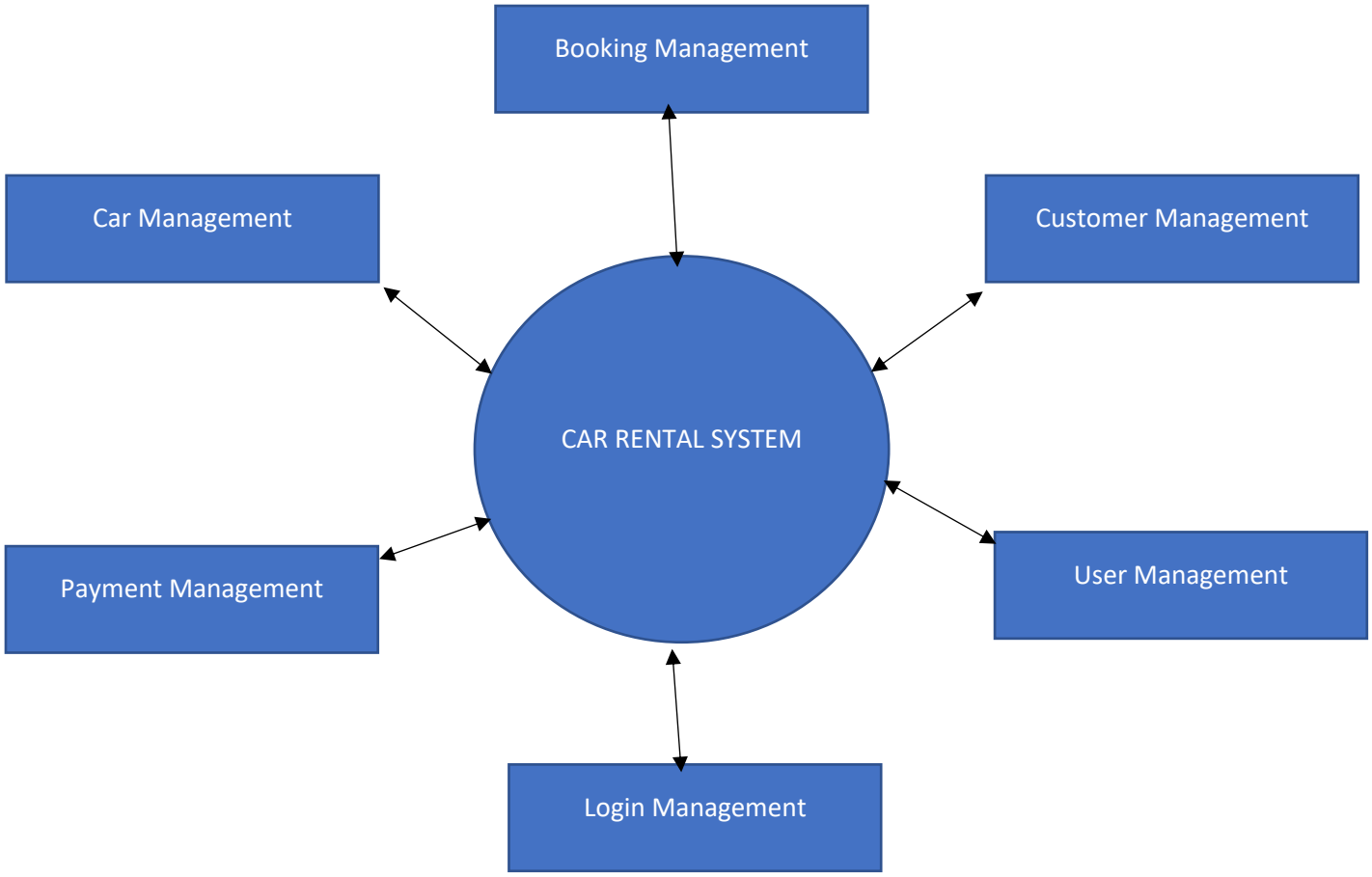
Description	Customer will enter personal details such as, name, email, contact, address, password etc. and then click on register button. After register customer have authority to reserve car.
Stimulus	Customer registration
Response	Successfully Registered now you can login!
Comment	You are able to reserve car

Use Case Diagram *Figure 3*



Level 0 Data Flow Diagram

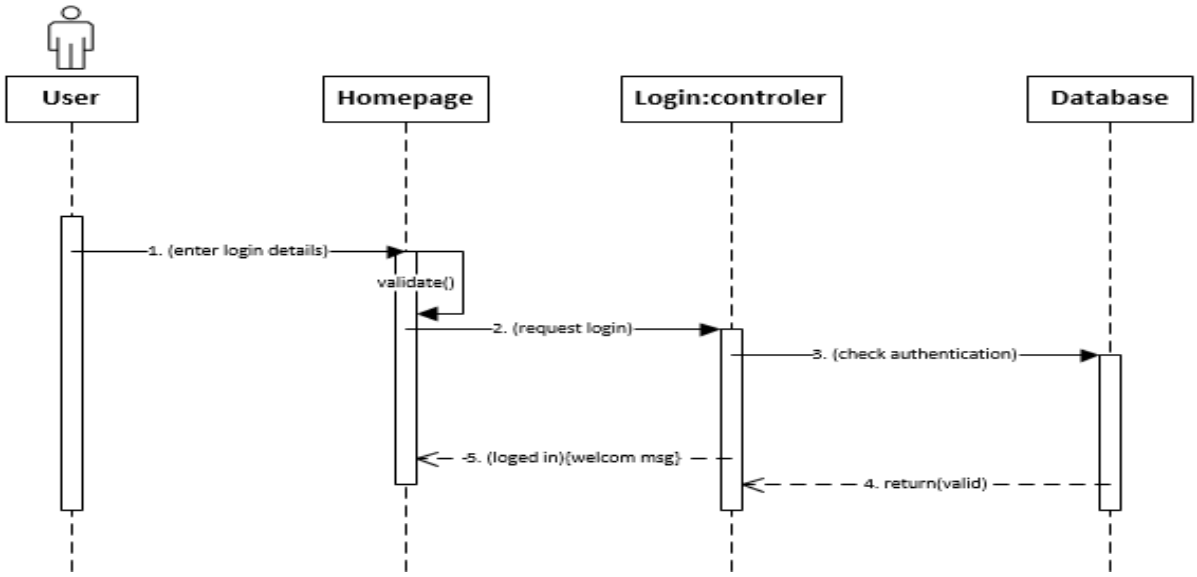
Figure 4



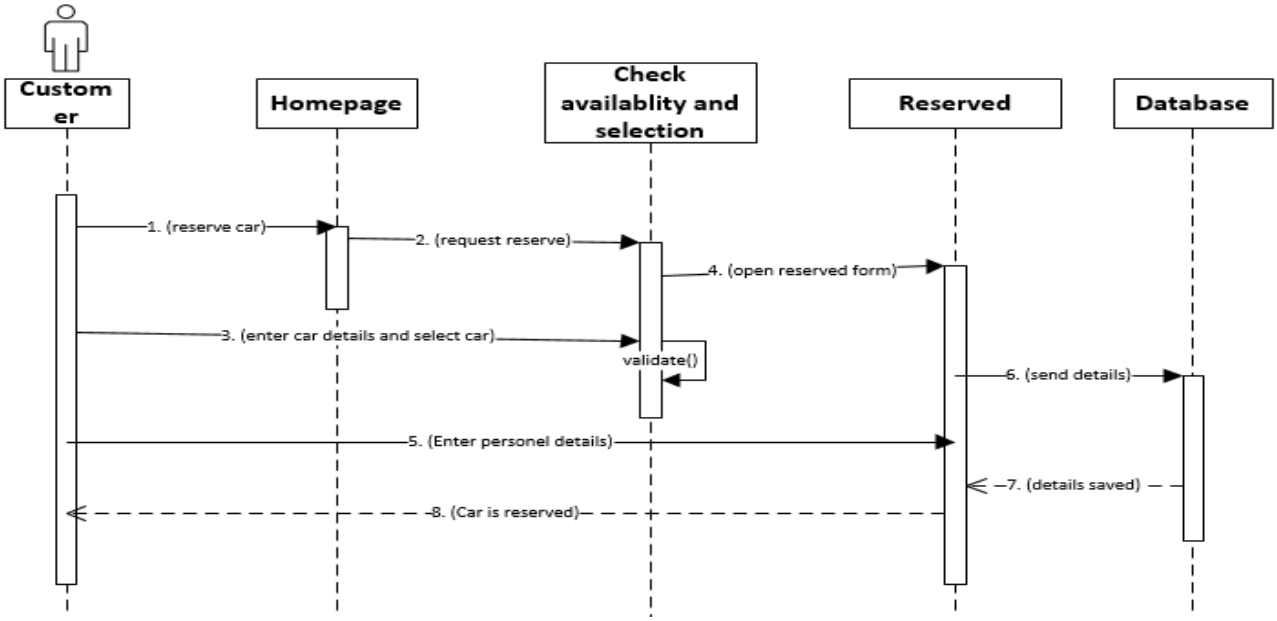
Level 1 Data Flow Diagram

Sequence Diagram

Login Sequence Diagram Figure 5

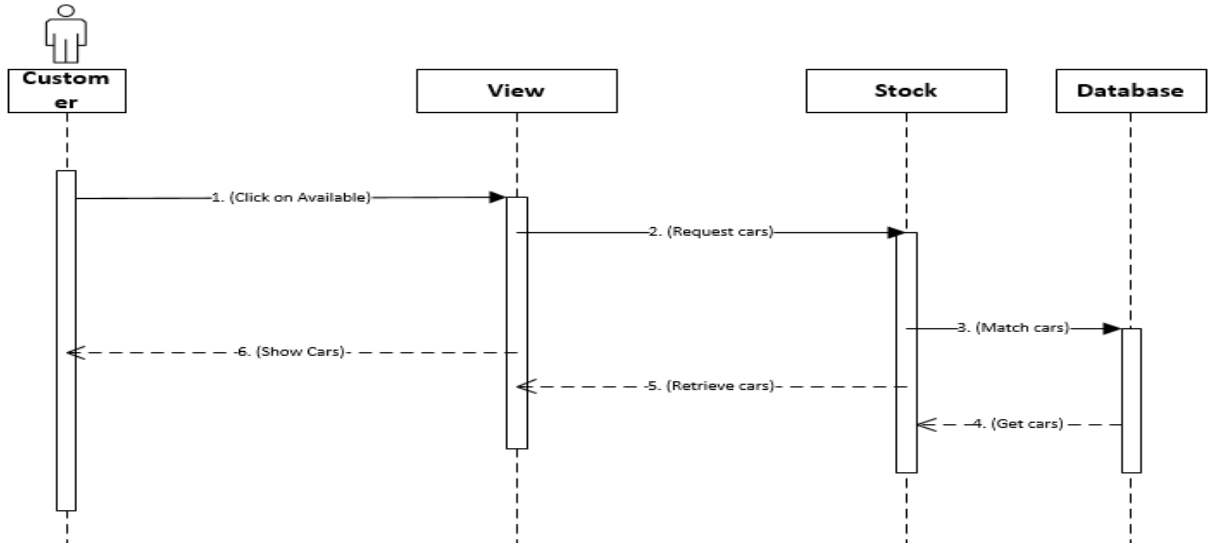


Reservation Sequence Diagram *Figure 6*



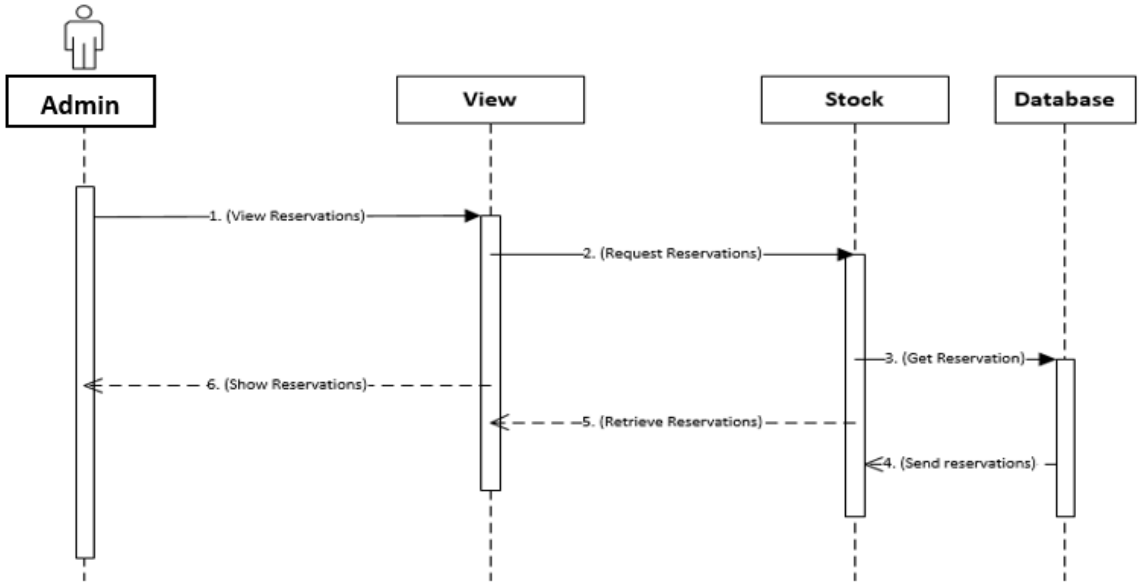
View Available Cars Sequential Diagram

Figure 7

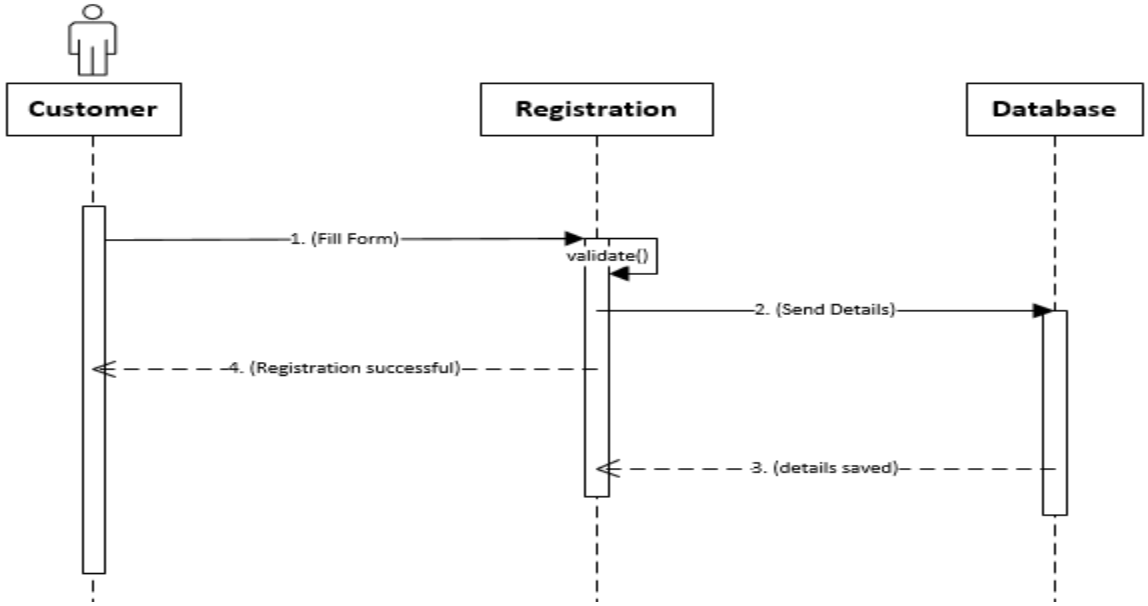


Reservations Sequential Diagram

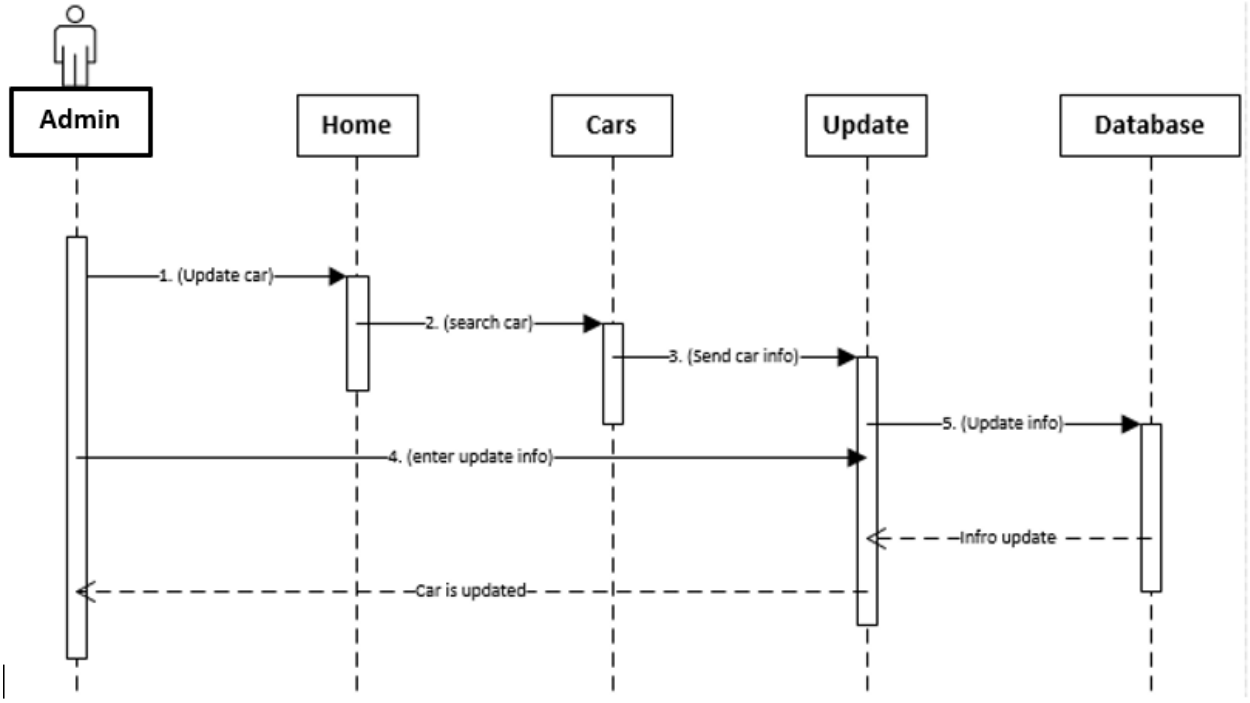
Figure 8



Registration Sequence Diagram Figure 9



Update car Sequence Diagram Figure 10



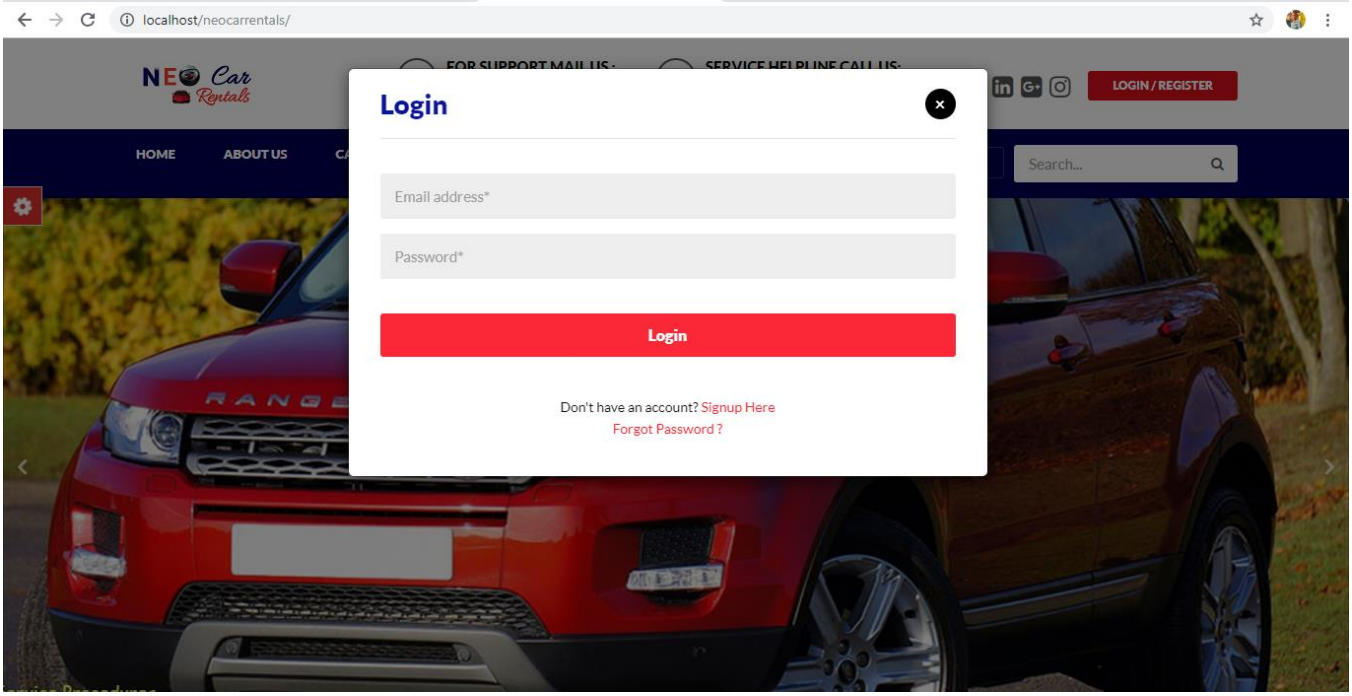
CHAPTER 4: SYSTEM DESIGN

BASIC MODULES

1) USER LOGIN MODULE

The user details are to be verified against the details in the users table and if it is valid, they should be able to login into the system. If the users have no username and password, they should register.

User Login *Figure 11*



Login code screenshot

Login code screenshot Figure 12

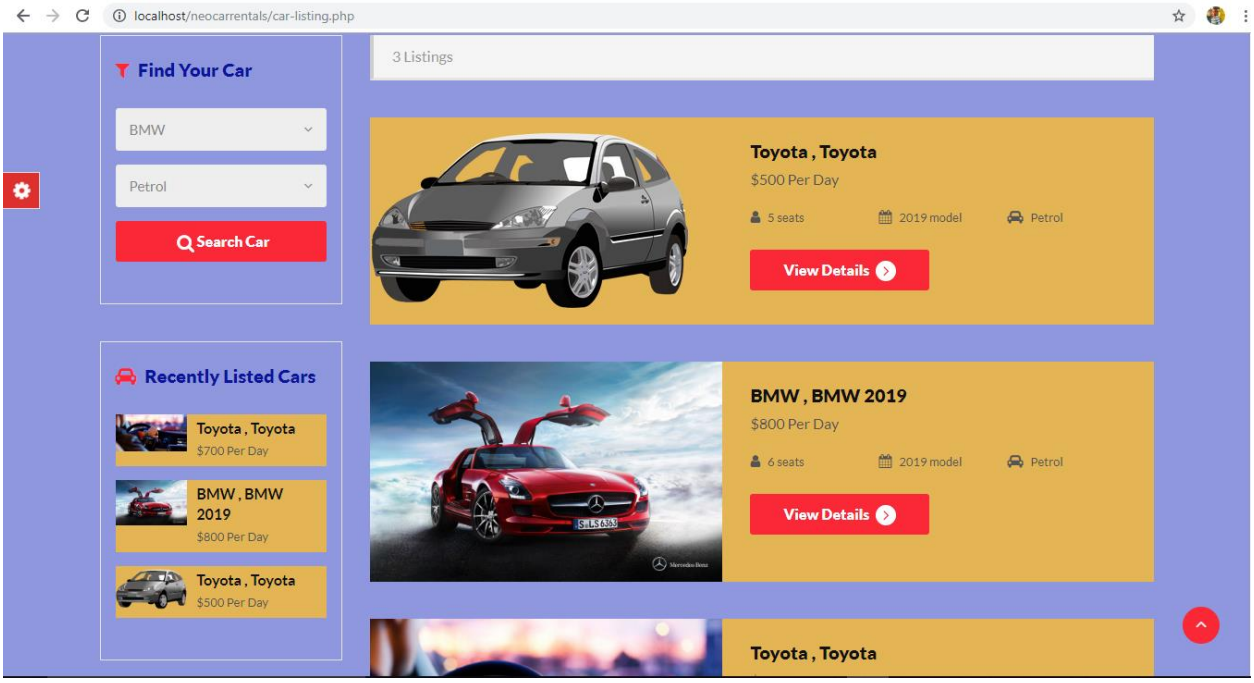
```php
<?php
if(isset($_POST['login']))
{
$email=$_POST['email'];
$password=md5($_POST['password']);
$sql ="SELECT EmailId,Password,FullName FROM neorent_users_table WHERE EmailId=:email and Password=:password";
$query= $dbh -> prepare($sql);
$query-> bindParam(':email', $email, PDO::PARAM_STR);
$query-> bindParam(':password', $password, PDO::PARAM_STR);
$query-> execute();
$results=$query->fetch(All(PDO::FETCH_OBJ);
if($query->rowCount() > 0)
{
$_SESSION['login']=$_POST['email'];
$_SESSION['fname']=$results->FullName;
$currentpage=$_SERVER['REQUEST_URI'];
echo "<script type='text/javascript'> document.location = '$currentpage'; </script>";
} else{

  echo "<script>alert('Invalid Details');</script>";

}

}
?>
```

2) SEARCH CAR MODULE

Customer can search for a vehicle of his/ her own choice from the available list of vehicles listed in the system.

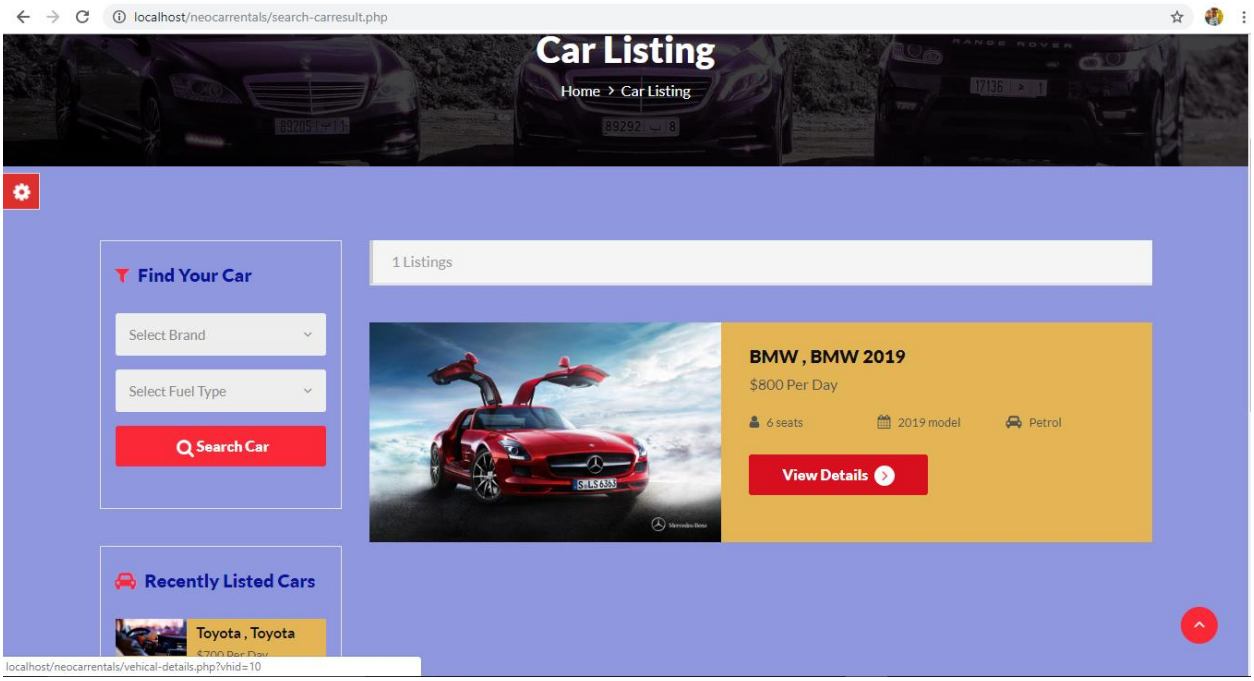
Search a car Screenshot

Search Car Module Screenshot 1



Search result screenshot

Search Car Result Screenshot *Figure 13*



Code for search module

Search Car Code Screenshot1 *Figure 14*

```
<?php
//Query for Listing count
$brand=$_POST['brand'];
$fueltype=$_POST['fueltype'];
$sql = "SELECT id from neorent_vehicles_table where neorent_vehicles_table.VehiclesBrand=:brand and neorent_vehicles_table.FuelType=:fueltype";
$query = $dbh -> prepare($sql);
$query -> bindParam(':brand',$brand, PDO::PARAM_STR);
$query -> bindParam(':fueltype',$fueltype, PDO::PARAM_STR);
$query->execute();
$results=$query->fetchAll(PDO::FETCH_OBJ);
$cnt=$query->rowCount();
?>
<p><span><?php echo htmlentities($cnt);?> Listings</span></p>
</div>
</div>

<?php

$sql = "SELECT neorent_vehicles_table.*,neorent_vehiclebrands_table.BrandName,neorent_vehiclebrands_table.id as bid  from neorent_vehicles_table join
neorent_vehiclebrands_table on neorent_vehiclebrands_table.id=neorent_vehicles_table.VehiclesBrand where neorent_vehicles_table.VehiclesBrand=:brand
and neorent_vehicles_table.FuelType=:fueltype";
$query = $dbh -> prepare($sql);
$query -> bindParam(':brand',$brand, PDO::PARAM_STR);
$query -> bindParam(':fueltype',$fueltype, PDO::PARAM_STR);
$query->execute();
$results=$query->fetchAll(PDO::FETCH_OBJ);
$cnt=1;
if($query->rowCount() > 0)
{
foreach($results as $result)
{ ?>
```

Search Car Code Screenshot2 *Figure 15*

```
<?php
        <div class="product-listing-m gray-bg">
          <div class="product-listing-img"><img src="admin/img/vehicleimages/<?php echo htmlentities($result->Vimage1);?>" class="img-responsive"
          alt="Image" /> </a>
          </div>
          <div class="product-listing-content">
            <h5><a href="vehical-details.php?vhid=<?php echo htmlentities($result->id);?>"><?php echo htmlentities($result->BrandName);?> , <?php echo
            htmlentities($result->VehiclesTitle);?></a></h5>
            <p class="list-price">$<?php echo htmlentities($result->PricePerDay);?> Per Day</p>
            <ul>
              <li><i class="fa fa-user" aria-hidden="true"></i><?php echo htmlentities($result->SeatingCapacity);?> seats</li>
              <li><i class="fa fa-calendar" aria-hidden="true"></i><?php echo htmlentities($result->ModelYear);?> model</li>
              <li><i class="fa fa-car" aria-hidden="true"></i><?php echo htmlentities($result->FuelType);?></li>
            </ul>
            <a href="vehical-details.php?vhid=<?php echo htmlentities($result->id);?>" class="btn">View Details <span class="angle_arrow"><i class="fa
            fa-angle-right" aria-hidden="true"></i></span></a>
          </div>
        </div>
      <?php }} ?>
        </div>
```

Search Car Code Screenshot3 *Figure 16*

```
                    <?php $sql = "SELECT * from  neorent_vehiclebrands_table ";
query = $dbh -> prepare($sql);
$query->execute();
$results=$query->fetchAll(PDO::FETCH_OBJ);
$cnt=1;
if($query->rowCount() > 0)
{
foreach($results as $result)
{      ?>
<option value="<?php echo htmlentities($result->id);?>"><?php echo htmlentities($result->BrandName);?></option>
<?php }} ?>

                  </select>
                </div>
                <div class="form-group select">
                  <select class="form-control">
                    <option>Select Fuel Type</option>
<option value="Petrol">Petrol</option>
<option value="Diesel">Diesel</option>
<option value="CNG">CNG</option>
                  </select>
                </div>

                <div class="form-group">
                  <button type="submit" class="btn btn-block"><i class="fa fa-search" aria-hidden="true"></i> Search Car</button>
                </div>
              </form>
            </div>
          </div>
```

Search Car Code Screenshot4 *Figure 17*

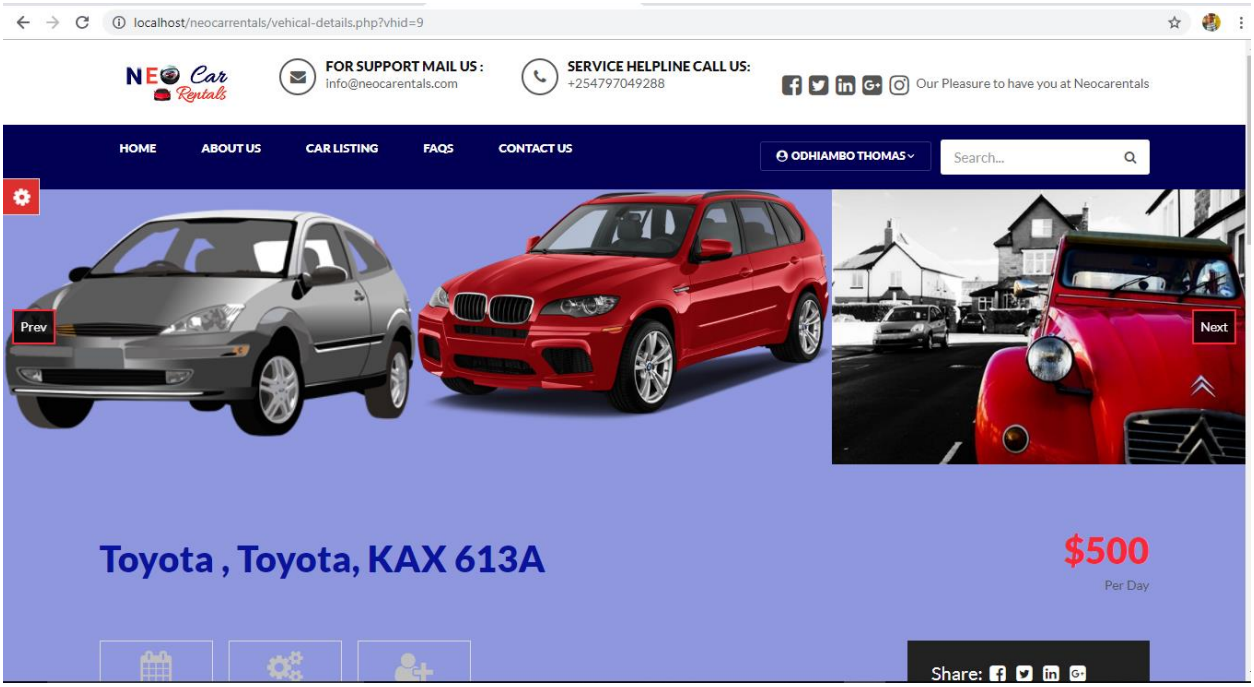
```
          </div>
          <div class="recent_addedcars">
            <ul>
<?php $sql = "SELECT neorent_vehicles_table.*,neorent_vehiclebrands_table.BrandName,neorent_vehiclebrands_table.id as bid  from neorent_vehicles_table
join neorent_vehiclebrands_table on neorent_vehiclebrands_table.id=neorent_vehicles_table.VehiclesBrand order by id desc limit 4";
query = $dbh -> prepare($sql);
$query->execute();
$results=$query->fetchAll(PDO::FETCH_OBJ);
$cnt=1;
if($query->rowCount() > 0)
{
foreach($results as $result)
{  ?>

              <li class="gray-bg">
                <div class="recent_post_img"> <a href="vehical-details.php?vhid=<?php echo htmlentities($result->id);?>"><img src=
                "admin/img/vehicleimages/<?php echo htmlentities($result->Vimage1);?>" alt="image"></a> </div>
                <div class="recent_post_title"> <a href="vehical-details.php?vhid=<?php echo htmlentities($result->id);?>"><?php echo htmlentities($result
                ->BrandName);?> , <?php echo htmlentities($result->VehiclesTitle);?></a>
                  <p class="widget_price">$<?php echo htmlentities($result->PricePerDay);?> Per Day</p>
                </div>
              </li>
              <?php }} ?>
```

3) CAR BOOKING MODULE

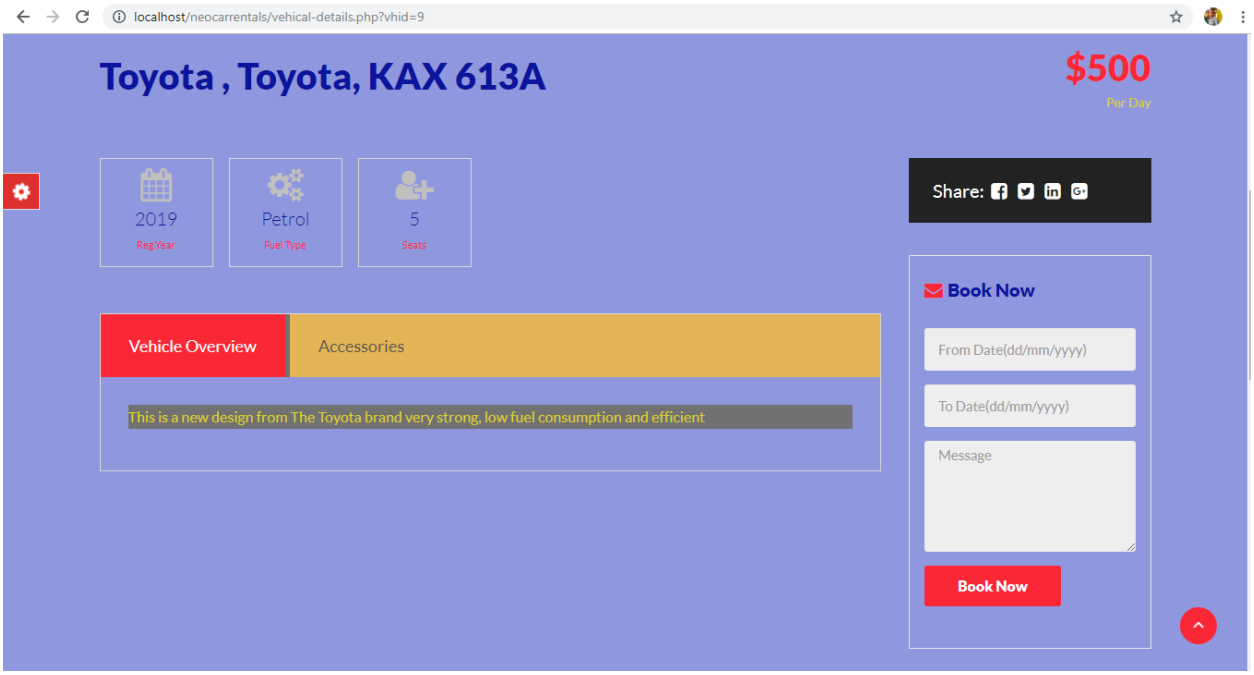
After a user successfully logs in to the system, he/she can search and book a vehicle of their choice. The client is supposed to pick on a vehicle, enter the date from which he/she is booking for the vehicle and the time at which he/she will return the vehicle.

Prior to booking of a vehicle, the user should also know the specifications of the vehicle e.g. seating capacity, fuel type, model year and accessories and a brief description about the vehicle.

Booking Module Screenshot1 *Figure 18*



Booking Module Screenshot2 *Figure 19*



Code for vehicle booking

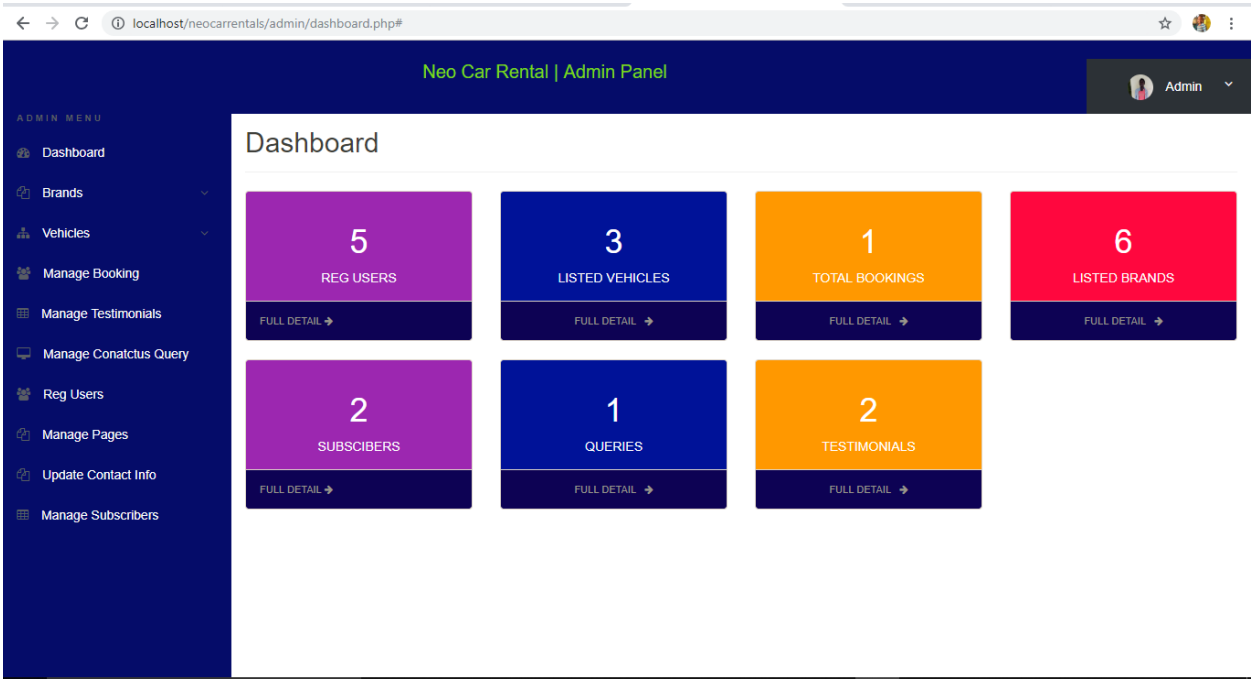
Booking Code Screenshot *Figure 20*

```php
<?php
session_start();
include('includes/config.php');
error_reporting(0);
if(isset($_POST['submit']))
{
$fromdate=$_POST['fromdate'];
$todate=$_POST['todate'];
$message=$_POST['message'];
$useremail=$_SESSION['login'];
$status=0;
$vhid=$_GET['vhid'];
$sql="INSERT INTO  neorent_booking_table(userEmail,VehicleId,FromDate,ToDate,message,Status)
VALUES(:useremail,:vhid,:fromdate,:todate,:message,:status)";
$query = $dbh->prepare($sql);
$query->bindParam(':useremail',$useremail,PDO::PARAM_STR);
$query->bindParam(':vhid',$vhid,PDO::PARAM_STR);
$query->bindParam(':fromdate',$fromdate,PDO::PARAM_STR);
$query->bindParam(':todate',$todate,PDO::PARAM_STR);
$query->bindParam(':message',$message,PDO::PARAM_STR);
$query->bindParam(':status',$status,PDO::PARAM_STR);
$query->execute();
$lastInsertId = $dbh->lastInsertId();
if($lastInsertId)
{
echo "<script>alert('Booking successfull.');</script>";
}
else
{
echo "<script>alert('Something went wrong. Please try again');</script>";
}
```

4) Admin Dashboard Module

In this module, the system administrator is capable of viewing all the registered users in in the system, the number of bookings, number of vehicles available in the system, listed brands, testimonials by customers and enquiries sent by customers.

Admin Dashboard Module Screenshot *Figure 21*



Code for admin Dashboard

Admin Dashboard Code Screenshot1 *Figure 22*

```
                                    <div class="stat-panel text-center">
<?php
$sql ="SELECT id from neorent_users_table ";
$query = $dbh -> prepare($sql);
$query->execute();
$results=$query->fetchAll(PDO::FETCH_OBJ);
$regusers=$query->rowCount();
?>
                                        <div class="stat-panel-number h1 "><?php echo htmlentities($regusers);?></div>
                                        <div class="stat-panel-title text-uppercase">Reg Users</div>
                                    </div>
                                </div>
                                <a href="reg-users.php" class="block-anchor panel-footer">Full Detail <i class="fa fa-arrow-right"></i></a>
                            </div>
                        </div>
                        <div class="col-md-3">
                            <div class="panel panel-default">
                                <div class="panel-body bk-success text-light">
                                    <div class="stat-panel text-center">
                                    <?php
$sql1 ="SELECT id from neorent_vehicles_table ";
$query1 = $dbh -> prepare($sql1);;
$query1->execute();
$results1=$query1->fetchAll(PDO::FETCH_OBJ);
$totalvehicle=$query1->rowCount();
?>
                                        <div class="stat-panel-number h1 "><?php echo htmlentities($totalvehicle);?></div>
                                        <div class="stat-panel-title text-uppercase">Listed Vehicles</div>
                                    </div>
                                </div>
                                <a href="manage-vehicles.php" class="block-anchor panel-footer text-center">Full Detail   <i class="fa
                                fa-arrow-right"></i></a>
```

Admin Dashboard Code Screenshot2 *Figure 23*

```
                                fa-arrow-right"></i></a>
92                            </div>
93                          </div>
94                          <div class="col-md-3">
95                            <div class="panel panel-default">
96                              <div class="panel-body bk-info text-light">
97                                <div class="stat-panel text-center">
98  <?php
99    $sql2 ="SELECT id from neorent_booking_table ";
100   $query2= $dbh -> prepare($sql2);
101   $query2->execute();
102   $results2=$query2->fetchAll(PDO::FETCH_OBJ);
103   $bookings=$query2->rowCount();
104   ?>
105                                
106                                  <div class="stat-panel-number h1 "><?php echo htmlentities($bookings);?></div>
107                                  <div class="stat-panel-title text-uppercase">Total Bookings</div>
108                                </div>
109                              </div>
110                              <a href="manage-bookings.php" class="block-anchor panel-footer text-center">Full Detail   <i class="fa
                                 fa-arrow-right"></i></a>
111                           </div>
112                         </div>
113                         <div class="col-md-3">
114                           <div class="panel panel-default">
115                             <div class="panel-body bk-warning text-light">
116                               <div class="stat-panel text-center">
117 <?php
118   $sql3 ="SELECT id from neorent_vehiclebrands_table ";
119   $query3= $dbh -> prepare($sql3);
120   $query3->execute();
121   $results3=$query3->fetchAll(PDO::FETCH_OBJ);
```

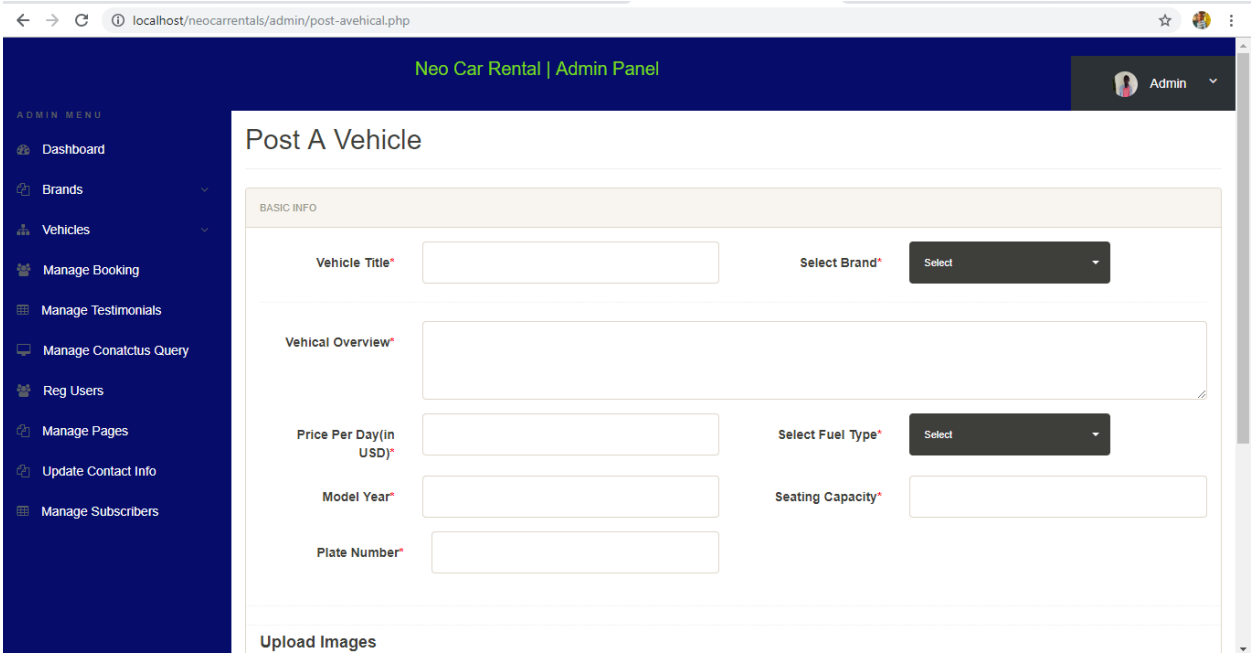
Admin Dashboard Code Screenshot3 *Figure 24*

```
144                 <div class="col-md-12">
145                   <div class="row">
146                     <div class="col-md-3">
147                       <div class="panel panel-default">
148                         <div class="panel-body bk-primary text-light">
149                           <div class="stat-panel text-center">
150 <?php
151   $sql4 ="SELECT id from neorent_subscribers_table ";
152   $query4 = $dbh -> prepare($sql4);
153   $query4->execute();
154   $results4=$query4->fetchAll(PDO::FETCH_OBJ);
155   $subscribers=$query4->rowCount();
156   ?>
157                                 <div class="stat-panel-number h1 "><?php echo htmlentities($subscribers);?></div>
158                                 <div class="stat-panel-title text-uppercase">Subscibers</div>
159                               </div>
160                             </div>
161                             <a href="manage-subscribers.php" class="block-anchor panel-footer">Full Detail <i class="fa fa-arrow-right"></i></a>
162                           </div>
163                         </div>
164                         <div class="col-md-3">
165                           <div class="panel panel-default">
166                             <div class="panel-body bk-success text-light">
167                               <div class="stat-panel text-center">
168                                 <?php
169   $sql6 ="SELECT id from neorent_contactusquery_table ";
170   $query6 = $dbh -> prepare($sql6);
171   $query6->execute();
172   $results6=$query6->fetchAll(PDO::FETCH_OBJ);
173   $query=$query6->rowCount();
174   ?>
175                                 <div class="stat-panel-number h1 "><?php echo htmlentities($query);?></div>
```

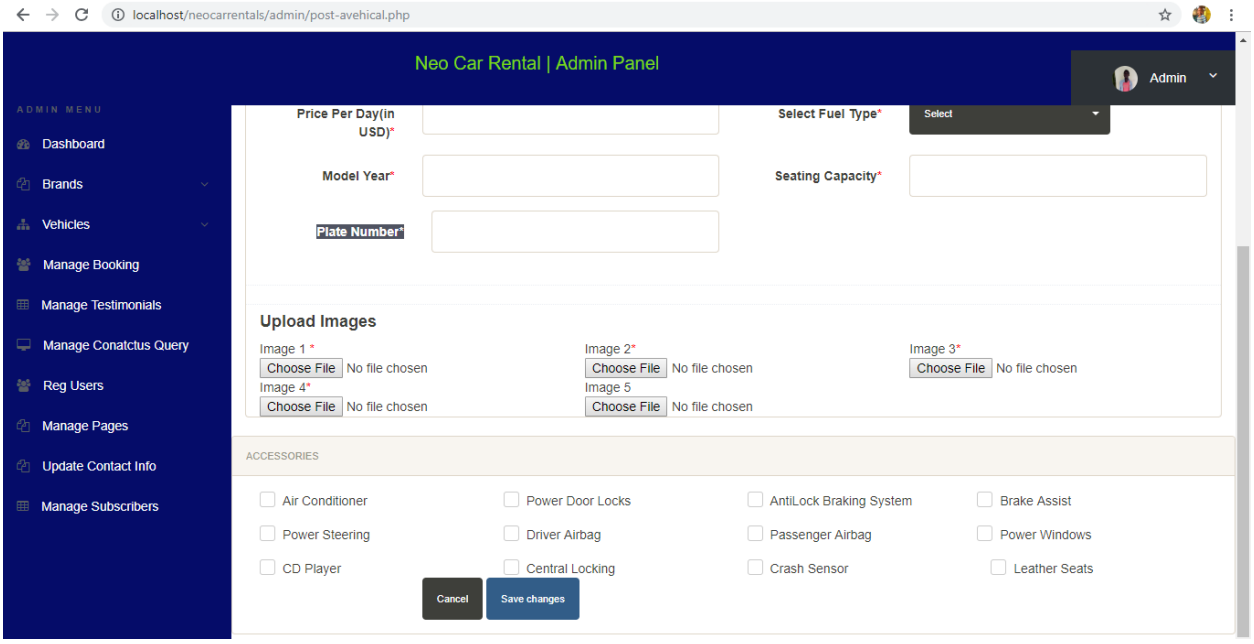
5) Add new car Module

The system administrator can add a new vehicle into the system. This new car will be listed for the user to be able to search and book them among the already available list.

Post New vehicle Module Screenshot1 *Figure 25*



Post New Vehicle Module Screenshot2 *Figure 26*



Code for posting new vehicle

Posting New Vehicle Code Screenshot1 *Figure 27*

```php
<?php
session_start();
error_reporting(0);
include('includes/config.php');
if(strlen($_SESSION['alogin'])==0)
  {
header('location:index.php');
}
else{

if(isset($_POST['submit']))
 {
$vehicletitle=$_POST['vehicletitle'];
$brand=$_POST['brandname'];
$vehicleoverview=$_POST['vehicalorcview'];
$priceperday=$_POST['priceperday'];
$fueltype=$_POST['fueltype'];
$modelyear=$_POST['modelyear'];
$seatingcapacity=$_POST['seatingcapacity'];
$platenumber=$_POST['platenumber'];
$vimage1=$_FILES["img1"]["name"];
$vimage2=$_FILES["img2"]["name"];
$vimage3=$_FILES["img3"]["name"];
$vimage4=$_FILES["img4"]["name"];
$vimage5=$_FILES["img5"]["name"];
$airconditioner=$_POST['airconditioner'];
$powerdoorlocks=$_POST['powerdoorlocks'];
$antilockbrakingsys=$_POST['antilockbrakingsys'];
$brakeassist=$_POST['brakeassist'];
$powersteering=$_POST['powersteering'];
$driverairbag=$_POST['driverairbag'];
$passengerairbag=$_POST['passengerairbag'];
```

Posting New Vehicles Coode Screenshot 2 *Figure 28*

```php
$powerwindow=$_POST['powerwindow'];
$cdplayer=$_POST['cdplayer'];
$centrallocking=$_POST['centrallocking'];
$crashcensor=$_POST['crashcensor'];
$leatherseats=$_POST['leatherseats'];
move_uploaded_file($_FILES["img1"]["tmp_name"],"img/vehicleimages/".$_FILES["img1"]["name"]);
move_uploaded_file($_FILES["img2"]["tmp_name"],"img/vehicleimages/".$_FILES["img2"]["name"]);
move_uploaded_file($_FILES["img3"]["tmp_name"],"img/vehicleimages/".$_FILES["img3"]["name"]);
move_uploaded_file($_FILES["img4"]["tmp_name"],"img/vehicleimages/".$_FILES["img4"]["name"]);
move_uploaded_file($_FILES["img5"]["tmp_name"],"img/vehicleimages/".$_FILES["img5"]["name"]);

$sql="INSERT INTO
neorent_vehicles_table(VehiclesTitle,VehiclesBrand,VehiclesOverview,PricePerDay,FuelType,ModelYear,SeatingCapacity,PlateNumber,Vimage1,Vimage2,Vimage3,Vimage4,Vimage5,AirConditioner,PowerDoorLocks,AntiLockBrakingSystem,BrakeAssist,PowerSteering,DriverAirbag,PassengerAirbag,PowerWindows,CDPlayer,CentralLocking,CrashSensor,LeatherSeats)
VALUES(:vehicletitle,:brand,:vehicleoverview,:priceperday,:fueltype,:modelyear,:seatingcapacity,:platenumber,:vimage1,:vimage2,:vimage3,:vimage4,:vimage5,:airconditioner,:powerdoorlocks,:antilockbrakingsys,:brakeassist,:powersteering,:driverairbag,:passengerairbag,:powerwindow,:cdplayer,:centrallocking,:crashcensor,:leatherseats)";
$query = $dbh->prepare($sql);
$query->bindParam(':vehicletitle',$vehicletitle,PDO::PARAM_STR);
$query->bindParam(':brand',$brand,PDO::PARAM_STR);
$query->bindParam(':vehicleoverview',$vehicleoverview,PDO::PARAM_STR);
$query->bindParam(':priceperday',$priceperday,PDO::PARAM_STR);
$query->bindParam(':fueltype',$fueltype,PDO::PARAM_STR);
$query->bindParam(':modelyear',$modelyear,PDO::PARAM_STR);
$query->bindParam(':seatingcapacity',$seatingcapacity,PDO::PARAM_STR);
$query->bindParam(':platenumber',$platenumber,PDO::PARAM_STR);
$query->bindParam(':vimage1',$vimage1,PDO::PARAM_STR);
$query->bindParam(':vimage2',$vimage2,PDO::PARAM_STR);
$query->bindParam(':vimage3',$vimage3,PDO::PARAM_STR);
$query->bindParam(':vimage4',$vimage4,PDO::PARAM_STR);
$query->bindParam(':vimage5',$vimage5,PDO::PARAM_STR);
```

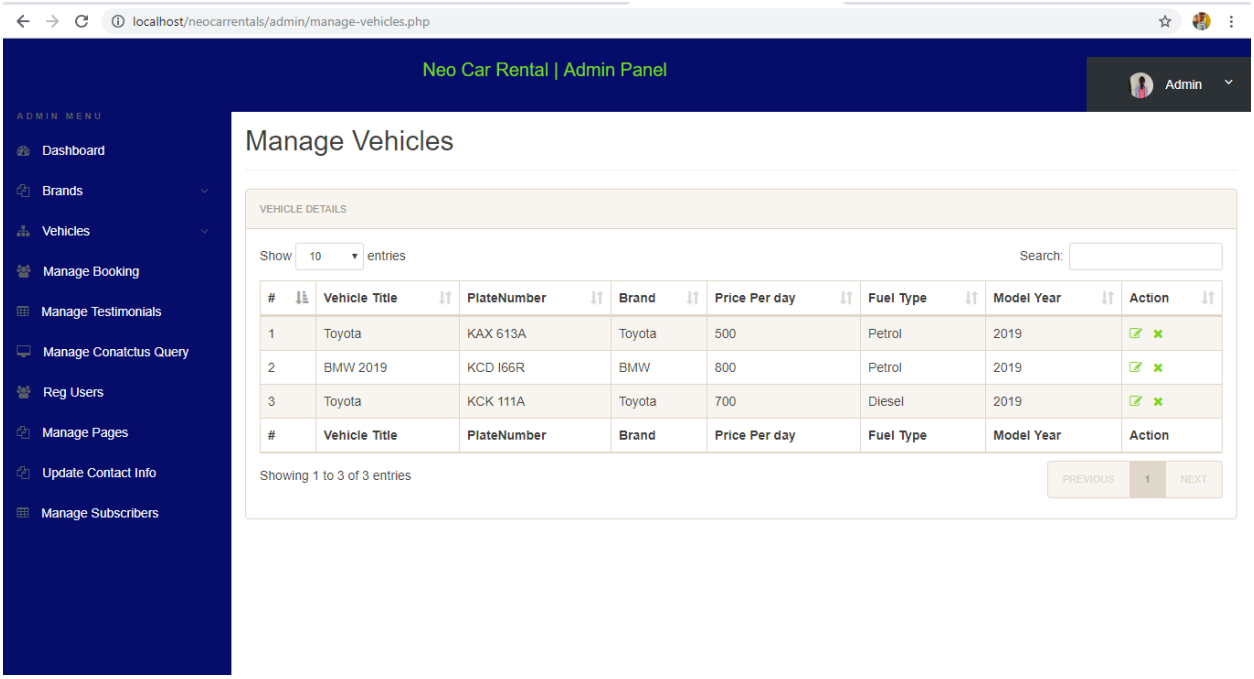
Posting New Vehicle Code Screenshot3 *Figure 29*

```
$query->bindParam(':vimage2',$vimage2,PDO::PARAM_STR);
$query->bindParam(':vimage3',$vimage3,PDO::PARAM_STR);
$query->bindParam(':vimage4',$vimage4,PDO::PARAM_STR);
$query->bindParam(':vimage5',$vimage5,PDO::PARAM_STR);
$query->bindParam(':airconditioner',$airconditioner,PDO::PARAM_STR);
$query->bindParam(':powerdoorlocks',$powerdoorlocks,PDO::PARAM_STR);
$query->bindParam(':antilockbrakingsys',$antilockbrakingsys,PDO::PARAM_STR);
$query->bindParam(':brakeassist',$brakeassist,PDO::PARAM_STR);
$query->bindParam(':powersteering',$powersteering,PDO::PARAM_STR);
$query->bindParam(':driverairbag',$driverairbag,PDO::PARAM_STR);
$query->bindParam(':passengerairbag',$passengerairbag,PDO::PARAM_STR);
$query->bindParam(':powerwindow',$powerwindow,PDO::PARAM_STR);
$query->bindParam(':cdplayer',$cdplayer,PDO::PARAM_STR);
$query->bindParam(':centrallocking',$centrallocking,PDO::PARAM_STR);
$query->bindParam(':crashicensor',$crashicensor,PDO::PARAM_STR);
$query->bindParam(':leatherseats',$leatherseats,PDO::PARAM_STR);
$query->execute();
$lastInsertId = $dbh->lastInsertId();
if($lastInsertId)
{
$msg="Vehicle posted successfully";
}
else
{
$error="Something went wrong. Please try again";
}

}
```

6) Manage Existing vehicles Module

The system admin will be able to manage all the existing vehicles in the system i.e. he/she is able to edit the vehicle details or permanently delete the vehicle from the system.

Manage Existing Vehicles Module Screenshot *Figure 30*



Source code for the module

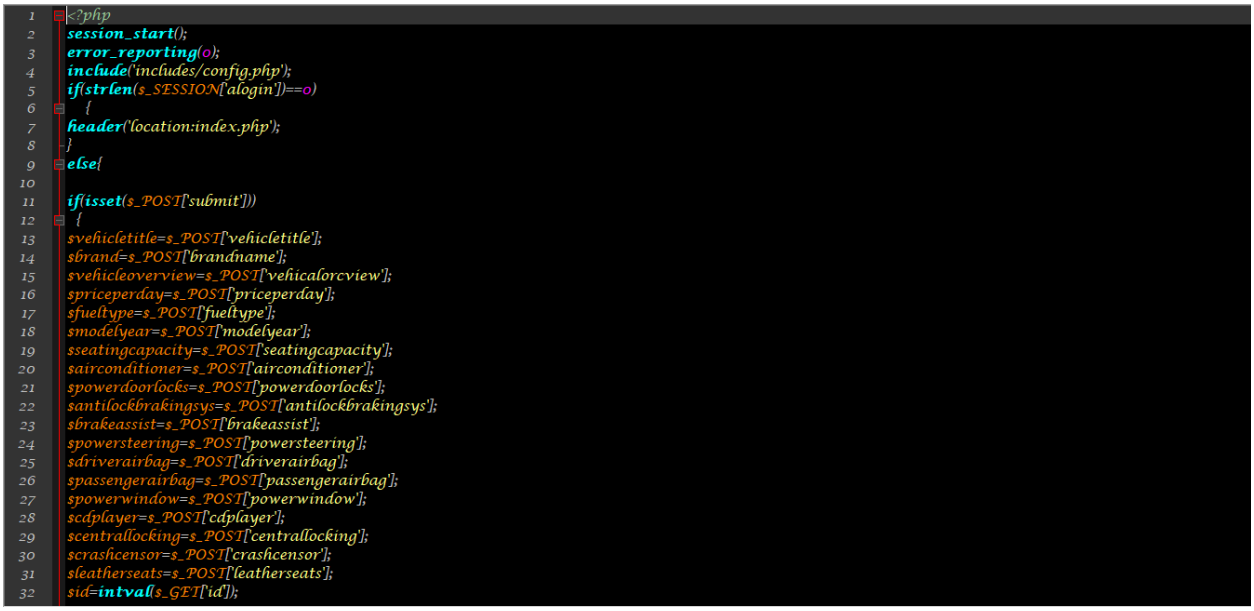
Code for deleting vehicle

Delete Vehicle Code Screenshot *Figure 31*

```php
<?php
session_start();
error_reporting(0);
include('includes/config.php');
if(strlen($_SESSION['alogin'])==0)
  {
header('location:index.php');
}
else{

if(isset($_REQUEST['del']))
  {
$delid=intval($_GET['del']);
$sql = "delete from neorent_vehicles_table  WHERE  id=:delid";
$query = $dbh->prepare($sql);
$query -> bindParam(':delid',$delid, PDO::PARAM_STR);
$query -> execute();
$msg="Vehicle  record deleted successfully";
}


 ?>
```

code for editing Vehicle Details

Edit Vehicle Details Code Screenshot *Figure 32*



```php
<?php
session_start();
error_reporting(0);
include('includes/config.php');
if(strlen($_SESSION['alogin'])==0)
  {
header('location:index.php');
}
else{

if(isset($_POST['submit']))
 {
$vehicletitle=$_POST['vehicletitle'];
$brand=$_POST['brandname'];
$vehicleoverview=$_POST['vehicalorcview'];
$priceperday=$_POST['priceperday'];
$fueltype=$_POST['fueltype'];
$modelyear=$_POST['modelyear'];
$seatingcapacity=$_POST['seatingcapacity'];
$airconditioner=$_POST['airconditioner'];
$powerdoorlocks=$_POST['powerdoorlocks'];
$antilockbrakingsys=$_POST['antilockbrakingsys'];
$brakeassist=$_POST['brakeassist'];
$powersteering=$_POST['powersteering'];
$driverairbag=$_POST['driverairbag'];
$passengerairbag=$_POST['passengerairbag'];
$powerwindow=$_POST['powerwindow'];
$cdplayer=$_POST['cdplayer'];
$centrallocking=$_POST['centrallocking'];
$crashcensor=$_POST['crashcensor'];
$leatherseats=$_POST['leatherseats'];
$id=intval($_GET['id']);
```

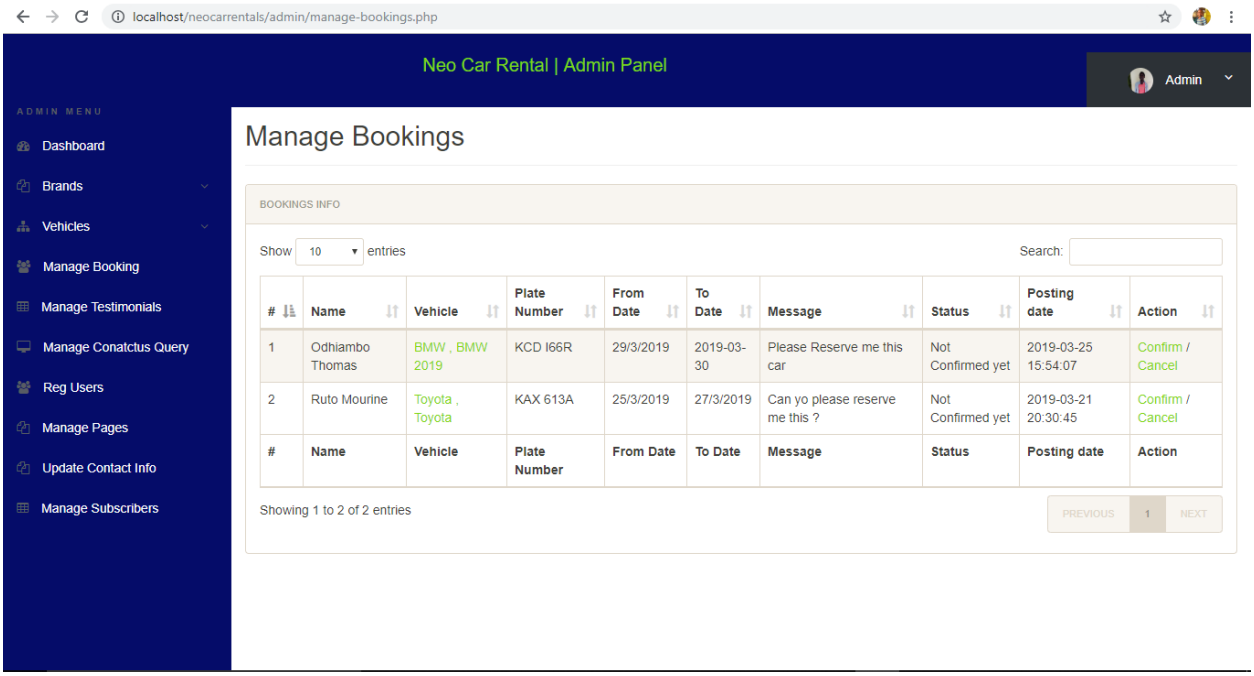
Edit vehicle details Code Screenshot2 *Figure 33*

```
$id=intval($_GET['id']);

$sql="update neorent_vehicles_table set
VehiclesTitle=:vehicletitle,VehiclesBrand=:brand,VehiclesOverview=:vehicleoverview,PricePerDay=:priceperday,FuelType=:fueltype,ModelYear=:modelyear,SeatingCapacity=:seatingc
apacity,AirConditioner=:airconditioner,PowerDoorLocks=:powerdoorlocks,AntiLockBrakingSystem=:antilockbrakingsys,BrakeAssist=:brakeassist,PowerSteering=:powersteering,Drive
rAirbag=:driverairbag,PassengerAirbag=:passengerairbag,PowerWindows=:powerwindow,CDPlayer=:cdplayer,CentralLocking=:centrallocking,CrashSensor=:crashcensor,LeatherSeats
=:leatherseats where id=:id";
$query = $dbh->prepare($sql);
$query->bindParam(':vehicletitle',$vehicletitle,PDO::PARAM_STR);
$query->bindParam(':brand',$brand,PDO::PARAM_STR);
$query->bindParam(':vehicleoverview',$vehicleoverview,PDO::PARAM_STR);
$query->bindParam(':priceperday',$priceperday,PDO::PARAM_STR);
$query->bindParam(':fueltype',$fueltype,PDO::PARAM_STR);
$query->bindParam(':modelyear',$modelyear,PDO::PARAM_STR);
$query->bindParam(':seatingcapacity',$seatingcapacity,PDO::PARAM_STR);
$query->bindParam(':airconditioner',$airconditioner,PDO::PARAM_STR);
$query->bindParam(':powerdoorlocks',$powerdoorlocks,PDO::PARAM_STR);
$query->bindParam(':antilockbrakingsys',$antilockbrakingsys,PDO::PARAM_STR);
$query->bindParam(':brakeassist',$brakeassist,PDO::PARAM_STR);
$query->bindParam(':powersteering',$powersteering,PDO::PARAM_STR);
$query->bindParam(':driverairbag',$driverairbag,PDO::PARAM_STR);
$query->bindParam(':passengerairbag',$passengerairbag,PDO::PARAM_STR);
$query->bindParam(':powerwindow',$powerwindow,PDO::PARAM_STR);
$query->bindParam(':cdplayer',$cdplayer,PDO::PARAM_STR);
$query->bindParam(':centrallocking',$centrallocking,PDO::PARAM_STR);
$query->bindParam(':crashcensor',$crashcensor,PDO::PARAM_STR);
$query->bindParam(':leatherseats',$leatherseats,PDO::PARAM_STR);
$query->bindParam(':id',$id,PDO::PARAM_STR);
$query->execute();

$msg="Data updated successfully";


}


  ?>
```

7) Mange bookings Module

The system admin will be able to view all the vehicle bookings and cancel or grant the vehicle booking requests. If the booking request has been accepted, the user is informed through the system that the booking was confirmed. If the booking request is denied, the user is also informed that the boking has been cancelled and appropriate reasons provided.

Manage Bookings Module *Figure 34*



Code for managing bookings

Manage Bookings Code Screenshot1 *Figure 35*

```php
<?php
session_start();
error_reporting(0);
include('includes/config.php');
if(strlen($_SESSION['alogin'])==0)
  {
header('location:index.php');
}
else{
if(isset($_REQUEST['eid']))
  {
$eid=intval($_GET['eid']);
$status="2";
$sql = "UPDATE neorent_booking_table SET Status=:status WHERE  id=:eid";
$query = $dbh->prepare($sql);
$query -> bindParam(':status',$status, PDO::PARAM_STR);
$query-> bindParam(':eid',$eid, PDO::PARAM_STR);
$query -> execute();

$msg="Booking Successfully Cancelled";
}


if(isset($_REQUEST['aeid']))
  {
$aeid=intval($_GET['aeid']);
$status=1;

$sql = "UPDATE neorent_booking_table SET Status=:status WHERE  id=:aeid";
$query = $dbh->prepare($sql);
$query -> bindParam(':status',$status, PDO::PARAM_STR);
$query-> bindParam(':aeid',$aeid, PDO::PARAM_STR);
$query -> execute();

$msg="Booking Successfully Confirmed";
}
```

Manage Bookings Code Screenshot2 *Figure 36*

```php
                                        <?php $sql = "SELECT
                                        neorent_users_table.FullName,neorent_vehiclebrands_table.BrandName,neorent_vehicles_table.VehiclesTitle,neorent_vehicles_table.PlateNumber,neorent_booking_
                                        table.FromDate,neorent_booking_table.ToDate,neorent_booking_table.message,neorent_booking_table.VehicleId as
                                        vid,neorent_booking_table.Status,neorent_booking_table.PostingDate,neorent_booking_table.id  from neorent_booking_table join neorent_vehicles_table on
                                        neorent_vehicles_table.id=neorent_booking_table.VehicleId join neorent_users_table on neorent_users_table.EmailId=neorent_booking_table.userEmail join
                                        neorent_vehiclebrands_table on neorent_vehicles_table.VehiclesBrand=neorent_vehiclebrands_table.id  ";
$query = $dbh -> prepare($sql);
$query->execute();
$results=$query->fetchAll(PDO::FETCH_OBJ);
$cnt=1;
if($query->rowCount() > 0)
{
foreach($results as $result)
{               ?>
                                            <tr>
                                            <td><?php echo htmlentities($cnt);?></td>
                                            <td><?php echo htmlentities($result->FullName);?></td>
                                            <td><a href="edit-vehicle.php?id=<?php echo htmlentities($result->vid);?>"><?php echo htmlentities($result->BrandName);?> , <?php echo
                                            htmlentities($result->VehiclesTitle);?></td>
                                            <td><?php echo htmlentities($result->PlateNumber);?></td>
                                            <td><?php echo htmlentities($result->FromDate);?></td>
                                            <td><?php echo htmlentities($result->ToDate);?></td>
                                            <td><?php echo htmlentities($result->message);?></td>
                                            <td><?php
if($result->Status==0)
{
echo htmlentities('Not Confirmed yet');
} else if ($result->Status==1) {
echo htmlentities('Confirmed');
}
else{
   echo htmlentities('Cancelled');
}
                                            ?></td>
                                            <td><?php echo htmlentities($result->PostingDate);?></td>
                                            <td><a href="manage-bookings.php?aeid=<?php echo htmlentities($result->id);?>" onclick="return confirm('Do you really want to Confirm this
                                            booking')"> Confirm</a> /
```

Manage Bookings Code Screenshot3 *Figure 37*

```
                                <td><?php echo htmlentities($result->ToDate);?></td>
                                <td><?php echo htmlentities($result->message);?></td>
                                <td><?php
if($result->Status==0)
{
echo htmlentities('Not Confirmed yet');
} else if ($result->Status==1) {
echo htmlentities('Confirmed');
}
else{
    echo htmlentities('Cancelled');
}
                                ?></td>
                                  <td><?php echo htmlentities($result->PostingDate);?></td>
                                <td><a href="manage-bookings.php?aeid=<?php echo htmlentities($result->id);?>" onclick="return confirm('Do you really want to Confirm this booking')"> Confirm</a> /


<a href="manage-bookings.php?eid=<?php echo htmlentities($result->id);?>" onclick="return confirm('Do you really want to Cancel this Booking')"> Cancel</a>
</td>

                                </tr>
                                <?php $cnt=$cnt+1; }} ?>
```

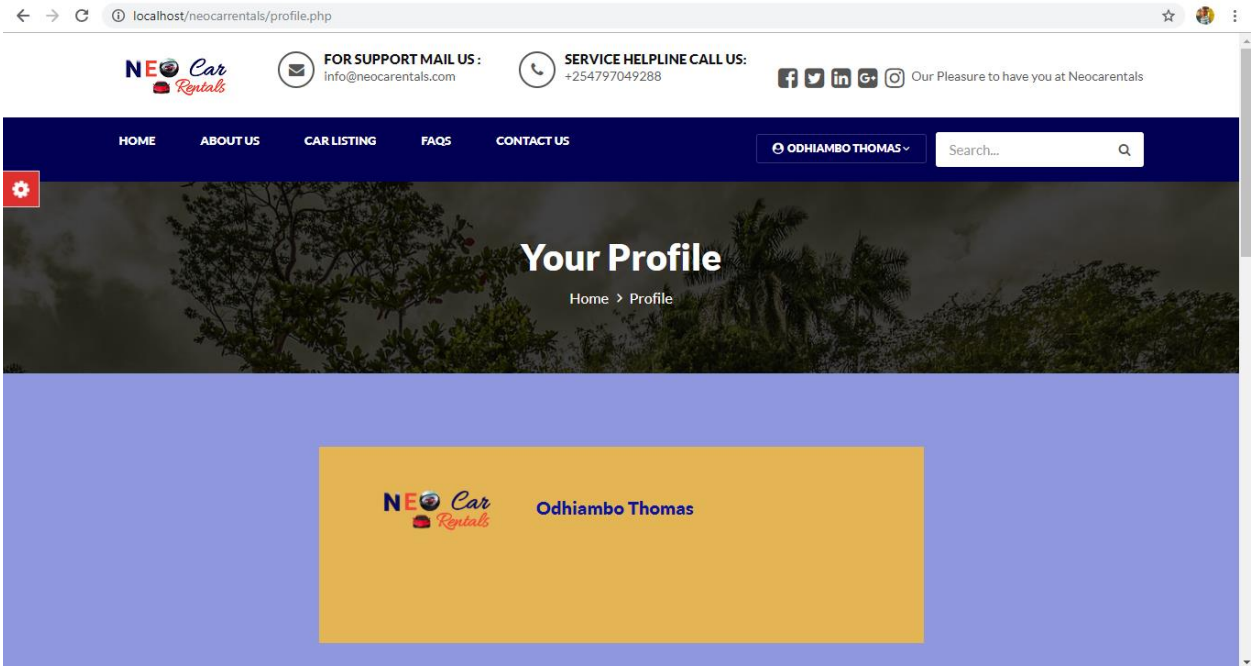
8) Manage Profile module

This module is available in both the user and the admin sides. The user can view his/her registration information, change their usernames and passwords or even contact information and also view their booking history and their posted testimonials.

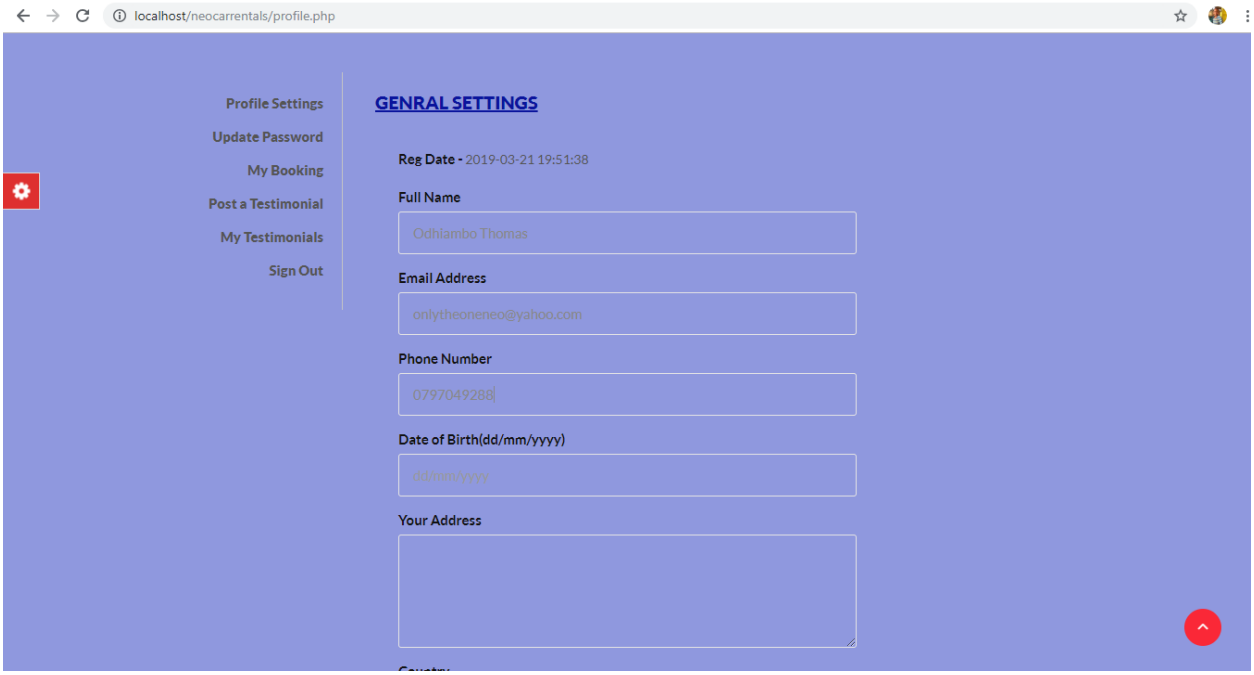
The admin is able to change their username and passwords.

User profile

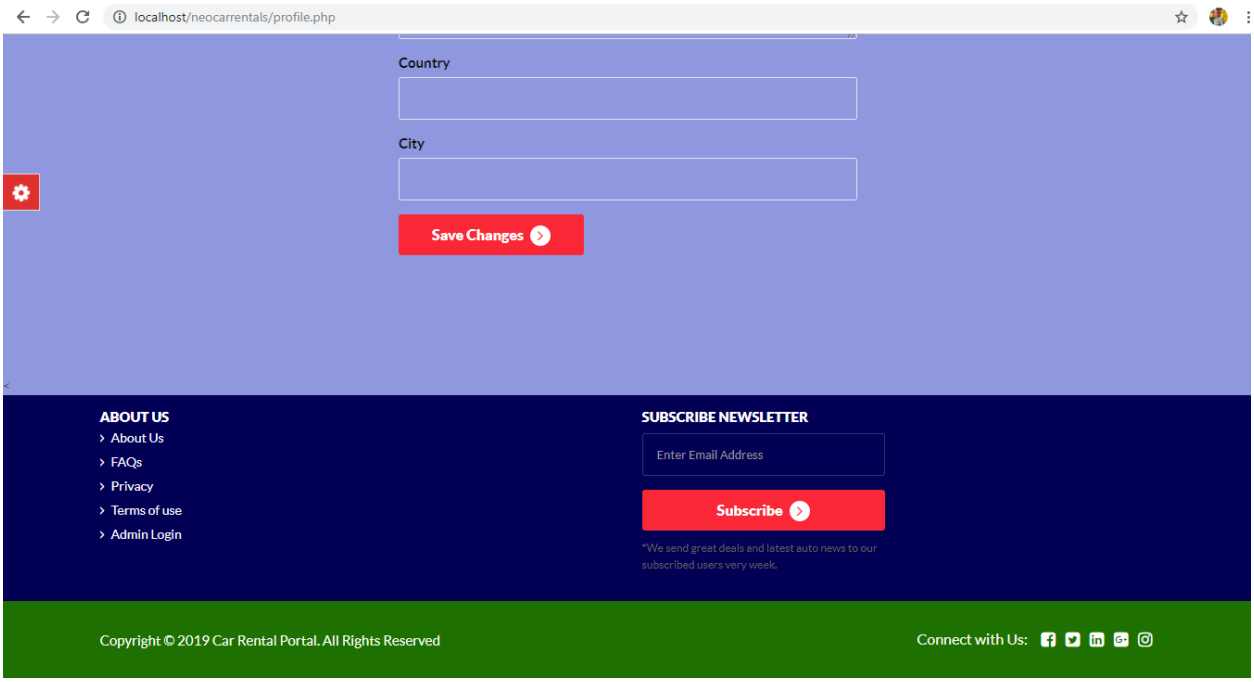
User Profile Screenshot1 *Figure 38*



User Profile Screenshot2 *Figure 39*

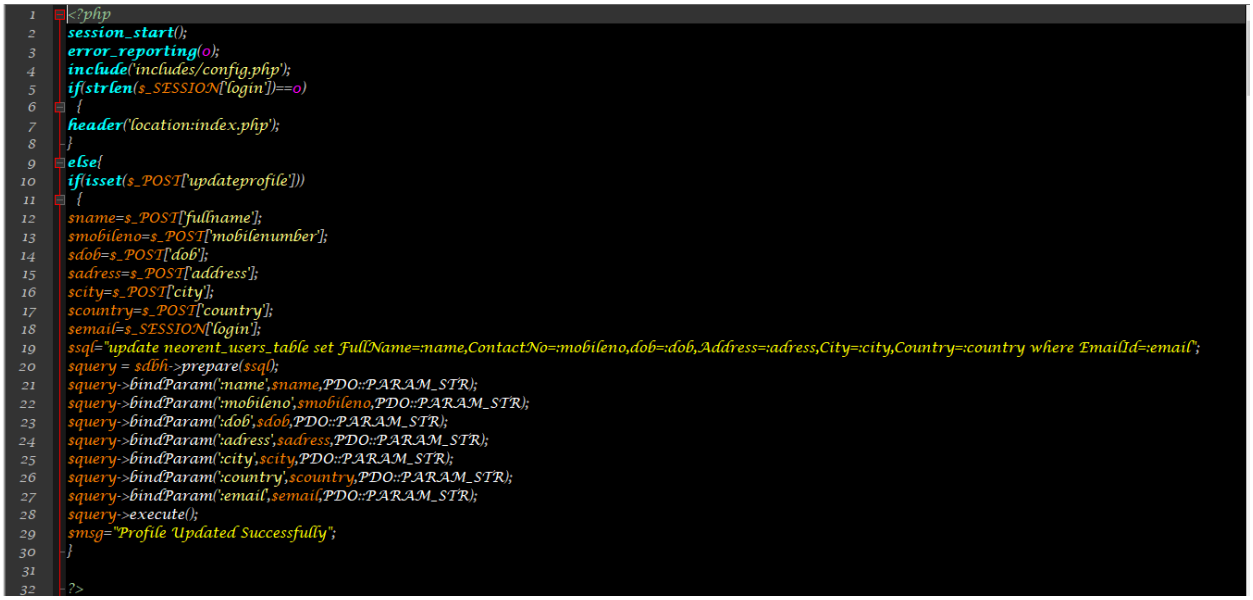


User Profile Screenshot 3 *Figure 40*



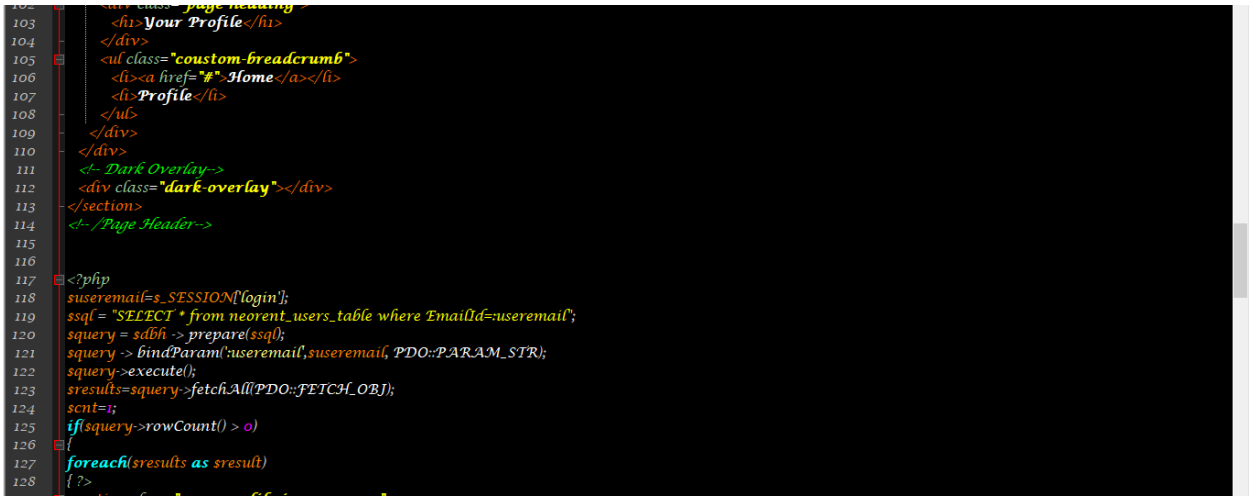
Code for user profile

User Profile Code screenshot1 *Figure 41*



```php
<?php
session_start();
error_reporting(0);
include('includes/config.php');
if(strlen($_SESSION['login'])==0)
 {
header('location:index.php');
}
else{
if(isset($_POST['updateprofile']))
 {
$name=$_POST['fullname'];
$mobileno=$_POST['mobilenumber'];
$dob=$_POST['dob'];
$adress=$_POST['address'];
$city=$_POST['city'];
$country=$_POST['country'];
$email=$_SESSION['login'];
$sql="update neorent_users_table set FullName=:name,ContactNo=:mobileno,dob=:dob,Address=:adress,City=:city,Country=:country where EmailId=:email";
$query = $dbh->prepare($sql);
$query->bindParam(':name',$name,PDO::PARAM_STR);
$query->bindParam(':mobileno',$mobileno,PDO::PARAM_STR);
$query->bindParam(':dob',$dob,PDO::PARAM_STR);
$query->bindParam(':adress',$adress,PDO::PARAM_STR);
$query->bindParam(':city',$city,PDO::PARAM_STR);
$query->bindParam(':country',$country,PDO::PARAM_STR);
$query->bindParam(':email',$email,PDO::PARAM_STR);
$query->execute();
$msg="Profile Updated Successfully";
}

?>
```

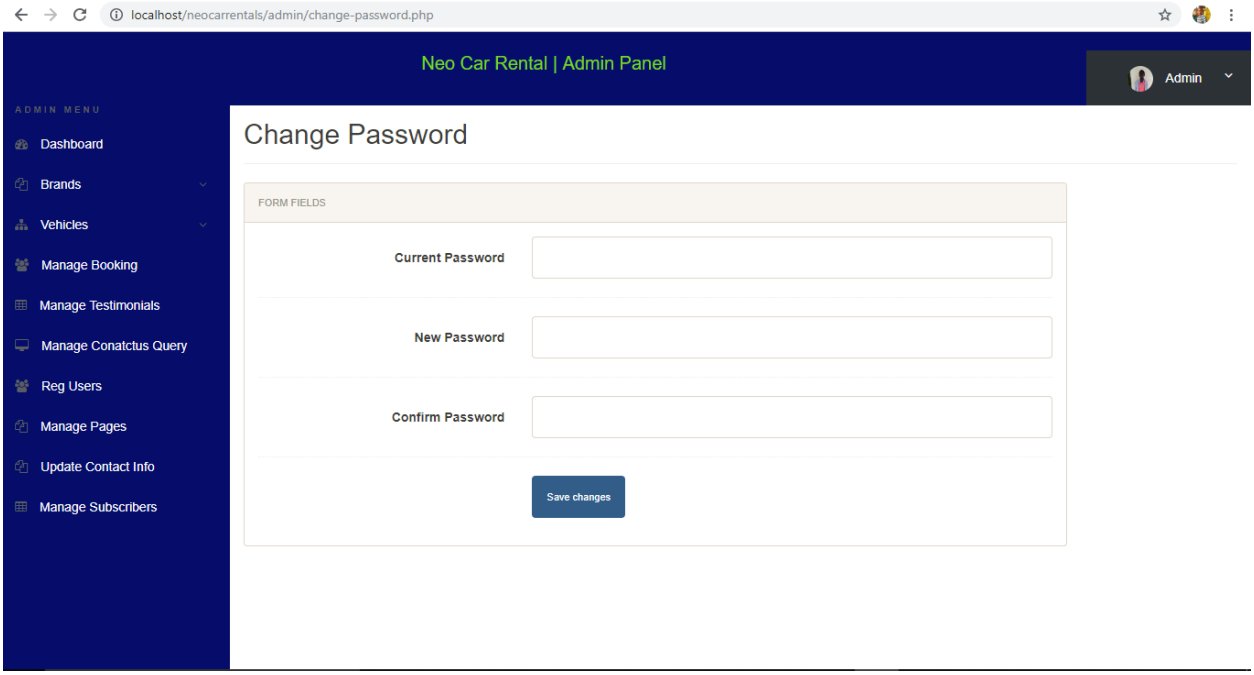
User Profile Code Screenshot2 *Figure 42*



```
        <h1>Your Profile</h1>
      </div>
      <ul class="coustom-breadcrumb">
        <li><a href="#">Home</a></li>
        <li>Profile</li>
      </ul>
    </div>
  </div>
  <!-- Dark Overlay-->
  <div class="dark-overlay"></div>
</section>
<!-- /Page Header-->


<?php
$useremail=$_SESSION['login'];
$sql = "SELECT * from neorent_users_table where EmailId=:useremail";
$query = $dbh -> prepare($sql);
$query -> bindParam(':useremail',$useremail, PDO::PARAM_STR);
$query->execute();
$results=$query->fetchAll(PDO::FETCH_OBJ);
$cnt=1;
if($query->rowCount() > 0)
{
foreach($results as $result)
{ ?>
```

Admin Profile

Admin Profile Screenshot *Figure 43*



Code for admin profile

Figure 44

```php
<?php
session_start();
error_reporting(0);
include('includes/config.php');
if(strlen($_SESSION['alogin'])==0)
  {
header('location:index.php');
}
else{
// Code for change password
if(isset($_POST['submit']))
  {
$password=md5($_POST['password']);
$newpassword=md5($_POST['newpassword']);
$username=$_SESSION['alogin'];
$sql ="SELECT Password FROM admin WHERE UserName=:username and Password=:password";
$query= $dbh -> prepare($sql);
$query-> bindParam(':username', $username, PDO::PARAM_STR);
$query-> bindParam(':password', $password, PDO::PARAM_STR);
$query-> execute();
$results = $query -> fetchAll(PDO::FETCH_OBJ);
if($query -> rowCount() > 0)
{
$con="update admin set Password=:newpassword where UserName=:username";
$chngpwd1 = $dbh->prepare($con);
$chngpwd1-> bindParam(':username', $username, PDO::PARAM_STR);
$chngpwd1-> bindParam(':newpassword', $newpassword, PDO::PARAM_STR);
$chngpwd1->execute();
$msg="Your Password succesfully changed";
}
else {
$error="Your current password is not valid.";
}
}
?>
```

4.2 Security

4.3 Test Cases Design

Test Case No.	Test Case Name	Purpose	Precondition	Test Steps	Expected Results
01	Register for reservation	Enable customer to register in the system	Customer has valid email id.	Click register button. Provide required information such as, name, email, phone no. Verify email provided by using verification link sent to email.	Customer will be registered in the system.
02	View vehicle description	Enable customer to view detailed description of particular vehicle	Customer is logged in the system	Click to see the vehicles of the company. Click on a particular vehicle. Detailed description of that vehicle will be displayed.	Detailed description of all vehicles in the system are displayed upon request.
03	Notify selection of unavailable vehicle while reservation	Notify customer that selected vehicle is unavailable for reservation	Customer is logged in the system. Customer has attempted to reserve a vehicle.	Click on reserve vehicle. Get notification that vehicle is unavailable.	Notification is displayed for all unavailable vehicles in the system upon attempt of reservation.
04	Advanced search by vehicle category	Enable customer to limit search of vehicle by brand, type and model.	Customer is logged in the system.	Select advanced search. Select category of search. Result will be displayed according to the parameters set.	All available search parameters display appropriate results.
05	Display list of available vehicle for reservation	Enable customer to view the list of vehicles available for reservation	Customer is logged into the system	Click on the button available vehicles A list of vehicles available for reservation will be displayed	All vehicles available for reservation will be on the list.

06	Cancel reservation	Enable customer to cancel reservation of vehicle	Customer is logged in the system. Customer has a vehicle reserved	Select cancel reservation. Provide reservation confirmation number. Reservation confirmation number will be verified by the system. Notification of successful/failed cancelation will be displayed.	Customer will receive a notification of successful cancelation of reservation. If failed, show error message.
07	Update reservation information	Enable Admin to update reservation information	Admin is logged in the system.	Select update reservation information Make necessary changes. Save changes	Updated information will be saved and displayed.
08	View list of reservations	Enable Admin to view reservations made by customers	Admin is logged in the system. Employee has	Select reservations	List of reservations made will be displayed.

09	Provide unique reservation number	Enable identification of reservations	Reservation has been made	Generate reservation number. Verify its uniqueness	Generate unique numbers for all reservations made
10	Display reservation summary	Enable customer to view a summary of a committed reservation	Customer is logged in the system Customer has made a successful reservation	Select view summary	Summary of reservation will be displayed.
11	User log in	Enable Users to log in to the system	User has required credentials to log in	Input username Input password Select user type (customer or admin) Verify username, password and type	If verified, grant access to the system. If not, show error message.

12	Create new customer account	Enable new customers to use the system	Customer has valid email id	Click sign up button. Provide required information such as, name, email, phone no. Verify email provided by using verification link sent to email.	Customer will be registered in the system after verification.
14	User log out	Enable user to log out of the system	User is logged in	Click log out button.	User will be logged out
15	Change password	Allow user to change password	User exists in the system	Click forgot password in log in page or change password in account settings. If forgot password, then recovery link will be sent to email. After clicking link, input new password, save password.	Log in credentials will be updated with new password.

				If from account settings, input current password, input new password, save password.	
16	Register new vehicle	Enable registration of new vehicle in the system	Employee is logged in Employee has clearance to register new vehicle	Input information of vehicle. Add picture. Save	New vehicle information will be saved and vehicle roster will be updated.
17	Update vehicle information	Enable Admin to update vehicle information to current status	Admin is logged in he/she has clearance to update vehicle information	Select vehicle Make necessary changes to the information of vehicle. Save changes.	Updated information will be saved and displayed.
18	View vehicle list	Enable Admin to view list of all vehicles	Admin is logged in	Select view all vehicles	List of all vehicles of the company will be displayed

19	View available vehicle list	Enable Admin to view list of available vehicles	Admin is logged in	Select view available vehicles	List of vehicles available for reservation will be displayed
20	View rented vehicle list	Enable Admin to view list of rented vehicles	Admin is logged in	Select view rented vehicles	List of vehicles currently rented will be displayed

CHAPTER 5: IMPLEMENTATION AND TESTING

5.1 Implementation Approaches

This is a phase in which the system analyst did an evaluation of the changeover method that should be used to switch from the existing manual system to the developed computerized system.

After a close analysis, the analyst decided to choose upon parallel method of changeover as the most appropriate.

In parallel method, the computerized system will run concurrently with the manual system before discarding the manual system. Although an expensive method, it will prove to be the most efficient because:

Parallel changeover provides time for the database administrator to update all the guest files before a total changeover to the new system.

Provides ample time for the employees to learn and adapt to the new system.

Lowers the risk to the management in case of a technical hitch or breakdown as the manual system will be still in place as the analyst fixes the technical hitch.

Its possible to troubleshoot any errors arising from loading process without affecting the car rental's transaction as the manual system will still be in place to carry out the car rental company's activities smoothly.

5.2 Testing

Testing is the process of detecting errors or bugs in a computer program. Testing performs a very critical role for quality assurance and for ensuring the reliability of software. The results of testing are used later on during maintenance also.

Purpose of Testing

The aim of testing is often to demonstrate that a program works by showing that it has no errors. The basic purpose of testing phase is to detect the errors that may be present in the program. Hence one should not start testing with the intent of showing that a program works, but the intent should be to show that a program doesn't work. Testing is the process of executing a

program with the intent of finding errors.

Testing Objectives:

The main objective of testing is to uncover a host of errors, systematically and with minimum effort and time. Testing is a process of executing a program with the intent of finding an error.
A successful test is one that uncovers an as yet undiscovered error.
A good test case is one that has a high probability of finding error, if it exists.
The tests are inadequate to detect possibly present errors.
The software more or less confirms to the quality and reliable standards.

Levels of Testing:

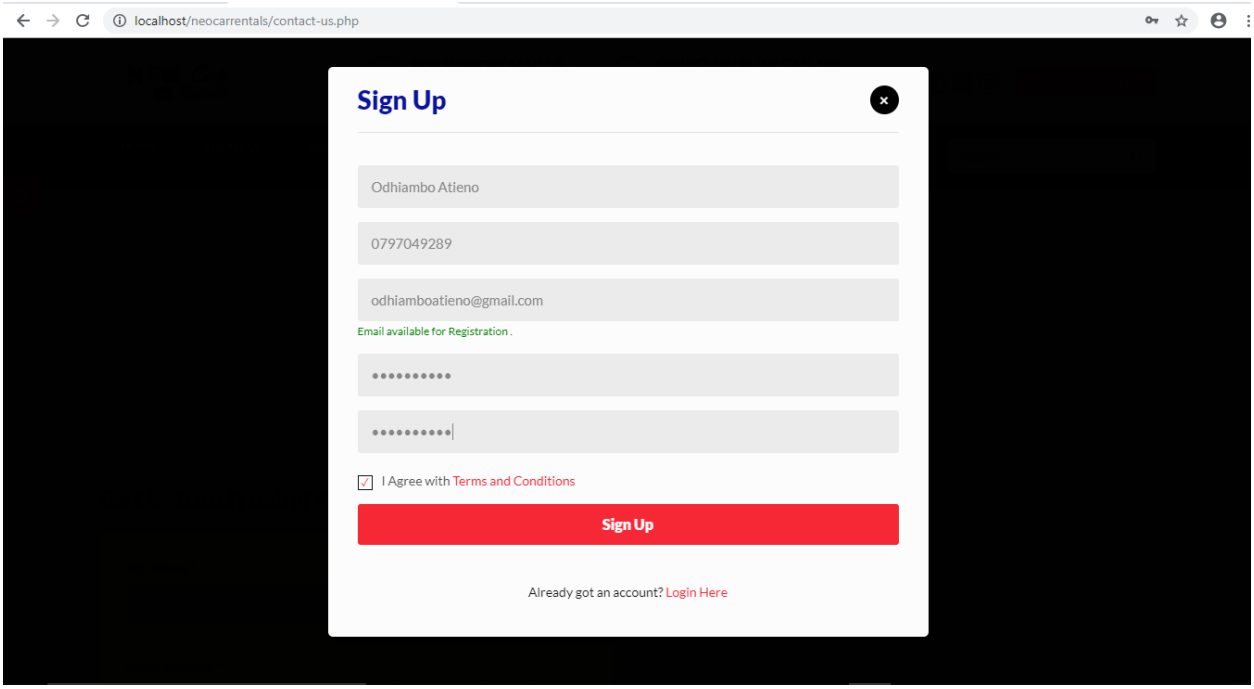
In order to uncover the errors, present in different phases we have the concept of levels of testing. The basic levels of testing are as shown below…

Client Needs
Requirements
Design
Code

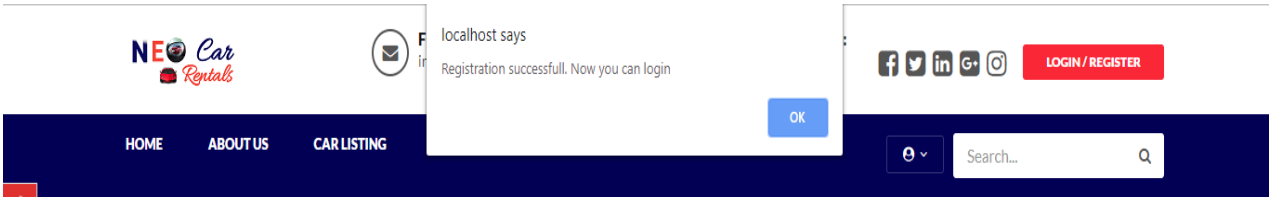
5.3 Installation and project description

The database as it is developed by MySQL can be installed only by using the export and import concepts.
Using XAMP Server Upload the system files to the htdocs directory then open your server in the browser by typing http://localhost/phpmyadmin to import the database
The project can be described by the screenshots in the project as follows

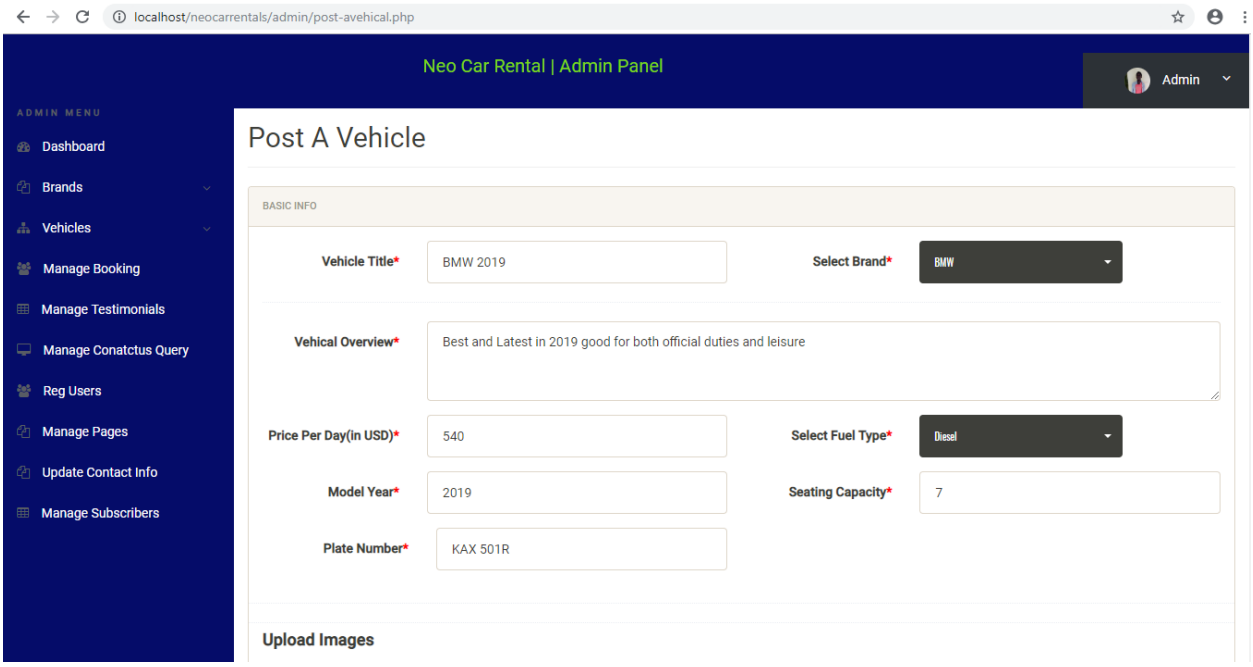
User Signup *Figure 45*

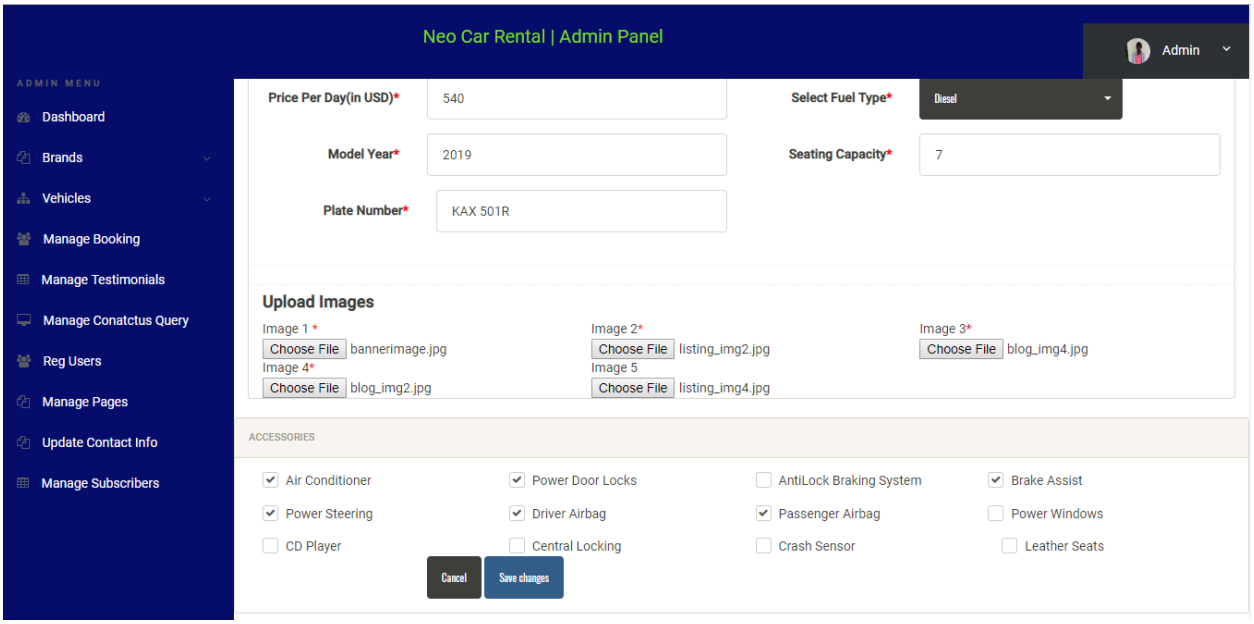


Registration Success Message *Figure 46*

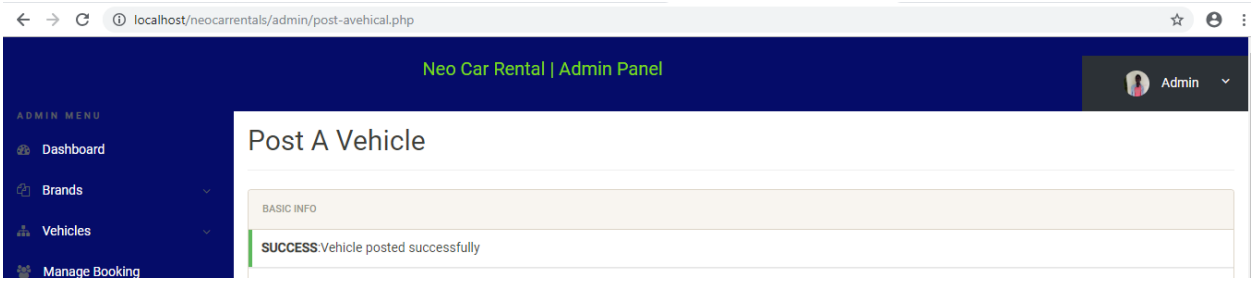


Posting a vehicle *Figure 47*





Post vehicle Success Message *Figure 48*



CHAPTER 6: CONCLUSIONS

6.1 Conclusion

Car rental business has emerged with a new goodies compared to the past experience where every activity concerning car rental business is limited to a physical location only.

Even though the physical location has not been totally eradicated, the nature of functions and how they are achieved has been reshaped by the power of internet.

Nowadays, customers can reserve online, rent a car online, and have the car brought to where they need it once a customer is a registered member or he/she may still visit the car rental company to pick the car.

This web-based car rental system has offered an advantage to both customers as well as Car Rental Company to efficiently and effectively manage the business and satisfies customers' need simply by a click of a button.

6.2 Limitations of the system

Although all efforts have been made to ensure the delivery of an efficient system. Some limitations could not be properly handled and they include:

- The car rental company will have to incur extra cost on electricity and internet bills due to computerization of the car rental management system.
- The car rental company will be required to train its employees on how to manage the system hence the car rental company output will reduce slightly during the training period.
- People who are not computer literate or those who cannot access computer services will not be able to use the system.

REFERENCES

- www.wikipedia.org
- www.w3schools.com
- www.wikipedia.org
- Mindtools.com, Online. 'Swim Lane/Rummler-Brache Diagrams: Mapping and Improving Processes in Your Organization'. N.p., 2015. Web. 9 June 2015.
- Laudon, Kenneth C, and Jane Price Laudon. Management Information Systems. Upper Saddle River, NJ: Prentice Hall, 2000. Print.
- Menkus, Belden. 'Car Rental Chain Former Owners Charged With Computer Frauds'. Computer Fraud & Security Bulletin 1993.3 (1993): 3-4. Web.
- Li, Zhang. 'Design And Realization Of Car Rental Management System Based On AJAX+SSH'. Information Technology J. 12.14 (2013): 2756-2761. Web.
- Scribd.com, Online. '49930505 Car Rental System Project Report'. N.p., 2015. Web. 9 June 2015.

GLOSSARY

Following are the list of abbreviations used in the documentation and their full meanings.

1. MB megabytes.
2. RAM random access memory.
3. Id Identity.
4. PHP Hypertext Pre-Processor.
5. Email Electronic mail.
6. GB Gigabytes.
7. Admin Administrator.
8. SQL Structured Query Language.