VISVESVARAYA TECHNOLOGICAL UNIVERSITY

JNANA SANGAMA, BELAGAVI – 590 018, KARNATAKA

A MINI PROJECT REPORT ON

"CAR RENTAL SYSTEM"

Submitted By

Deshwanth S
USN: 1VK20CS023

MANOJ LK
USN: 1VK20CS040

Submitted in the partial fulfilment of the requirements for the fifth Semester Database Management Lab (18CSL58)

BACHELOR OF ENGINEERING

IN

COMPUTER SCIENCE AND ENGINEERING

UNDER THE GUIDANCE OF

Prof. Mrs. Prajwala R
Asst. Professor, Dept. of CSE VKIT,
Bengaluru

Prof. Ms. Sowmya George
Asst. Professor, Dept. of ISE VKIT,
Bengaluru

Janatha Education Society®

VIVEKANANDA INSTITUTE OF TECHNOLOGY

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING GUDIMAVU,

KUMBALAGODU POST, KENGERI HOBLI, BENGALURU –560 074

2022-2023

Janatha Education Society®

VIVEKANANDA INSTITUTE OF TECHNOLOGY

Gudimavu, Kumbalagodu Post, Kengeri Hobli, Bengaluru – 560 074

Department of Computer Science and Engineering

CERTIFICATE

This is to certify that the Mini project work entitled **"Car rental System"** carried out by **Deshwanth S** bearing the **USN: 1VK20CS023** and **Manoj LK** bearing the **USN: 1VK20CS040** is a bonafide student of **Vivekananda Institute of Technology**, **Bengaluru** in partial fulfilment of the requirements for the fifth semester **DBMS LABORATORY WITH MINI PROJECT (18CSL58)** of **Bachelor of Engineering** in **Computer Science and Engineering** of the Visveshwaraya Technological University, Belagavi during the year 2022-23. It is certified that all corrections/suggestions indicated for internal assessment have been incorporated in the report deposited in the departmental library. The mini project report has been approved as it satisfies the academic requirements in respect of mini Project work prescribed for the said degree.

Signature of the Guide	Signature of the HOD	Signature of the Principal
Prof. Mrs. Prajwala R	**Dr. Vidya A**	**Dr. Padmanabha S**
Asst. Professor,	Prof. & Head,	Principal
Dept. of CSE	Dept of. CSE	VKIT, Bengaluru
VKIT, Bengaluru	VKIT, Bengaluru	

External Viva-Voce

Name of the Examiners	Signature with date
1._____	_____
2._____	_____

ACKNOWLEDGEMENT

It gives us immense pleasure to write an acknowledgement to this mini project, a contribution of all people who helped me realize it. We are very thankful to our Principal, **Dr. Padmanabha S,** VKIT, Bengaluru, for being kind enough to provide us an opportunity to work on a project in this institution. We would like to convey our heartfelt thanks to our beloved HOD, **Dr. Vidya A, Department of Computer Science and Engineering,** VKIT**,** Bengaluru for giving us the opportunity to embark up to this topic. We would like to sincerely thank our project guide **Mrs. Prajwala R Asst. Professors, Dept. of Computer Science and Engineering** and **Ms. Sowmya, Asst. Professors, Dept. of Information Science and Engineering** for their valuable guidance, constant assistance, support and constructive suggestions for the betterment of the project, without which thisproject would have not been possible.

Finally, it is a pleasure and happiness to the friendly co-operation showed by all the staffs of Computer Science and Engineering Department, VKIT.

Deshwanth S (1VK20CS023)

Manoj LK (1VK20CS040)

ABSTRACT

This project is designed so as to be used by Car Rental Company specializing in renting cars to customers. It is an online system through which customers can view available cars, register, view profile and book car..

A car rental is a vehicle that can be used temporarily for a fee during a specified period. Getting a rental car helps people get around despite the fact they do not have access to their own personal vehicle or don't own a vehicle at all. The individual who needs a car must contact a rental car company and contract out for a vehicle. This system increases customer retention and simplify vehicle and staff management.

TABLE OF CONTENTS

LIST OF SNAPSHOTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVATIONS

[1]	DBMS	:	Database Management System
[2]	MySQL	:	My Structured Query Language
[3]	JDK	:	Java Development Kit
[4]	JDBC	:	Java Database Connectivity
[5]	1NF	:	First Normal Form
[6]	2NF	:	Second Normal Form
[7]	3NF	:	Third Normal Form
[8]	BCNF	:	Boyce–Codd Normal Form

CHAPTER 1

INTRODUCTION

A Database Management System (DBMS) refers to the technology for creating and managing databases. Basically DBMS is a software tool to organize (create, retrieve, update and manage) data in a database.

The main aim of a DBMS is to supply a way to store up and retrieve database information that is both convenient and efficient. By data, we mean known facts that can be recorded and that have embedded meaning. Normally people use software such as DBASE IV or V, Microsoft ACCESS, or EXCEL to store data in the form of database. Database system is meant to handle large collection of information. Management of data involves both defining structures for information and providing mechanisms that can do the manipulation of those stored information. Moreover, the database system must ensure the safety of the information stored, despite system crashes or attempts at unauthorized access.

1.1 PROJECTOVERVIEW

- Enhance Business Processes: To be able to use internet technology to project the rental company to the global world instead of limiting their services to their local domain alone, thus increase their return on investment (ROI).
- Online Vehicle Reservation: A tools through which customers can reserve available cars online prior to their expected pick-up date or time.
- Customer's registration: A registration portal to hold customer's details, monitor their transaction and used same to offer better and improve services to them.
- Group bookings: Allows the customer to book space for a group in the case of weddings or corporate meetings (Event management).

1.2 PROJECTOBJECTIVES

This project is designed so as to be used by Car Rental Company specializing in renting cars to customers. It is an online system through which customers can view available cars, register, view profile and book car.

- To produce a web-based system that allow customer to register and reserve car online and for the company to effectively manage their car rental business.
- To ease customer's task whenever they need to rent a car.

1.3EXISTINGSYSTEM

The existing system is a type of system being used at the counter is an internal system which is manually used in book the cars, etc. to store the information like booking details, cancellation details, managing records of rented cars. It is very difficult to maintain historical data.

1.4 PROPOSEDSYSTEM

Proposed System is a software application which avoids more manual hours that need to spend in record keeping and generating reports. This project is handled by administrator alone. The propose car rental System will make the process faster. paperwork popular to this day. In the computer system of the cargo service computation of the rate is easily and quickly done. Currently, the type of system being used at the counter is an internal system which is manually used in booking cars. The problems facing the company are that customers have to go to the counter to book cars or ask for availability, customers will also have to queue up for a long time in order to secure a car and will also need to pay cash when they book a car.

1.5 PURPOSE

A car rental is a vehicle that can be used temporarily for a period of time with a fee. Renting a car assists people to get around even when they do not have access to their own personal vehicle or don't own a vehicle at all. The individual who want to rent a car must first contact the car rental company for the desire vehicle. This can be done online. At this point, this person has to supply some information such as; dates of rental, and type of car. After these details are worked out, the individual renting the car must present a valid Identification Card. Most companies throughout the industry make a profit based of the type of cars that are rented. The rental cars are categorized into economy, compact, compact premium, premium and luxury. And customers are free to choose any car of their choice based on their purse and availability of such car at the time of reservation.

1.6 ADVANTAGES

- This online car rental solution is fully functional and flexible.
- It is very easy to use.
- This online car rental system helps in back-office administration by streamlining and standardizing the procedures. It reduces the burden to staff.
- It saves a lot of time, money, and labor.
- Eco-friendly: The monitoring of the vehicle activity and the overall business becomes easy and includes the least of paper work.
- It provides custom features development and support with the software.
- The software acts as an office that is open 24/7.

1.7 SCOPE

The scope of the project is managing a consistency and storage of data by dedicated data administrator. It provides most of the features that a Database Management System should have. It is developed by using MySQL database. It has been implemented in WINDOWS platform.

CHAPTER 2

HARDWARE AND SOFTWARE REQUIREMENTS

2.1 FEASIBILITY STUDY:

After analyzing all the existing or required functionalities of the system, the next task is to do the feasibility study for the project. All projects are feasible –given unlimited resources and infinite time. Feasibility study includes consideration of all the possible ways to provide a solution to the given problem. The proposed solution should satisfy all the user requirements and should be flexible enough so that future changes can be easily done based on the future upcoming requirements.

2.1.1 TECHNICAL FEASIBILITY:

The technical feasibility in the proposed system deals with the technology used in the system. It deals with the hardware and software used in the system whether they are of latest technology or not. With feasibility comes responsibility to reach a larger crowd. This project handles this easily with JDK world‘s best way of creating feasible software. With any system running on Java JRE can easily handle this application. The project was coded with Java Apache NetBeans 12(JDK -14) and data handled by MYSQL.

2.1.2 ECONOMICAL FEASIBILITY:

Economic analysis is the most frequently used method for evaluating the cost efficiency of a new system. This project is completely pocket friendly as its Open Source. The source codes will be made available on GitHub once the project is approved. The project is made with Open Source code editors and will comply with GPLv2.

2.1.3 OPERATIONAL FEASIBILITY:

A project becomes useless if cannot be understood by a layman with little or no knowledge about computer terms. Therefore, operational feasibility is the important aspect. In terms of this the project will ensure full operability with both advanced and inexperienced users.

2.2 HARDWARE REQUIREMENTS:

The section of hardware configuration is an important task related to the software development insufficient random access memory may affect adversely on the speed and efficiency of the entire system. The process should be powerful to handle the entire operations. The hard disk should have sufficient capacity to store the file and application.

Processor: INTEL i5 11^{th} GEN

Processor speed: 1.0 Hz Onwards

System memory: 500MB or more.

Cache size: 1024 KB

RAM: 512 MB (Minimum)

Hard disk: 2GB (Minimum)

Monitor: SVGA Color 15‖

Mouse: 104 keys US Key Serial, USB or PS/2

2.3 SOFTWARE REQUIREMENTS:

A major element in building a system is the section of compatible software since the software in the market is experiencing in geometric progression. Selected software should be acceptable by the firm and one user as well as it should be feasible for the system. This document gives a detailed description of the software requirement specification. The study of requirement specification is focused specially on the functioning of the system. It allows the developer or analyst to understand the system, function to be carried out the performance level to be obtained and corresponding interfaces to be established.

Front end tool: PHP my Admin

Backend: MySQL, XAMPP Sever

Operating system: Linux, Windows (XP and above), MAC.

2.4 About technologies Used:

- **PHP**:phpMyAdmin is a free and open source administration tool for MySQL and MariaDB. As a portable web application written primarily in PHP, it has become one of the most popular MySQL administration tools, especially for web hosting services.

- **XAMPP**: XAMPP is a free and open source cross platform web server solution stack package developed by Apache Friends, consisting mainly of the Apache HTTP Server, Maria DB database, and interpreters for scripts written in IPL DATABASE MANAGEMENT Dept. of CSE, VKIT 2019-2020 4 the PHP and Perl programming languages. XAMPP stands for Cross-Platform, Apache, Maria DB, PHP and Perl. It is a simple, lightweight Apache distribution that makes it extremely easy for developers to create a local web server for testingand deployment purposes. Since most actual web server deployments use the same components as XAMPP, it makes transitioning from a local test server to a live serverextremely easy aswell.

- **MYSQL CONNECTOR**: MySQL Connector, once known as My ODBC, is computer software from Oracle cooperation. It is an ODBC interface and allows programming languages that support the ODBC interface to communicate with a MySQL database. MySQL Connector/ODBC was originally created by MYSQLAB. It provides connectivity to the MYSQL server and the java Apache NetBeans.

- **MYSQL**: MYSQL is the language used to manipulate relational databases. It is tied closely with the relational model. It is issued for the purpose of data definition and data manipulation. Program runs as a server providing multi-user access to a number of databases. MySQL is a multithreaded, multi-user SQL database management system (DBMS). It includes facilities to add, modify or delete data from the database, ask questions (or queries) about the data stored in the database and produce reports summarizing selected contents.

CHAPTER 3

SYSTEM DESIGN

In this phase initially ER Diagram has been designed in order to identify various entities and relationship set, entity set, attributes.

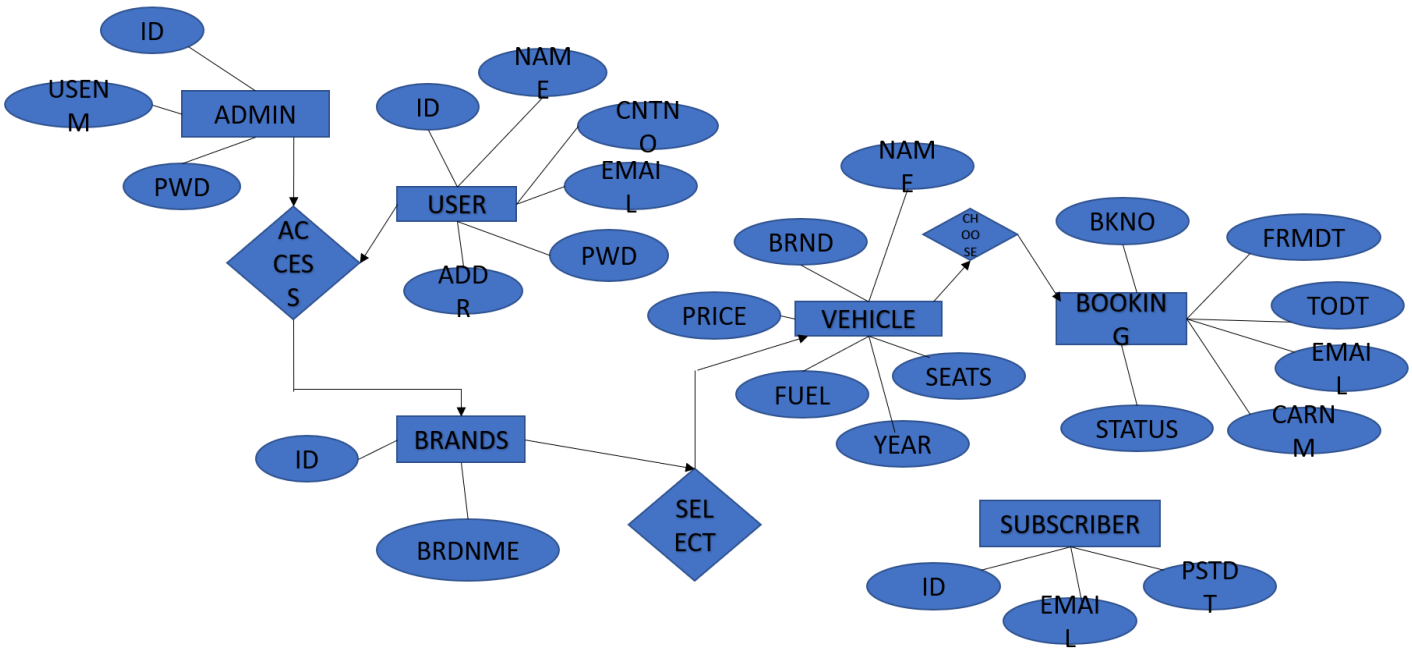
After this step database for the new system has been designed and normalized it.

3.1 INPUT DESIGN

The login page has username and password for user login.

- Admin: The admin can do various functions like add airlines, delete airlines, update airlines details, keep track of the bookings, can search various airlines and its details, etc.

3.2 ER DIAGRAM



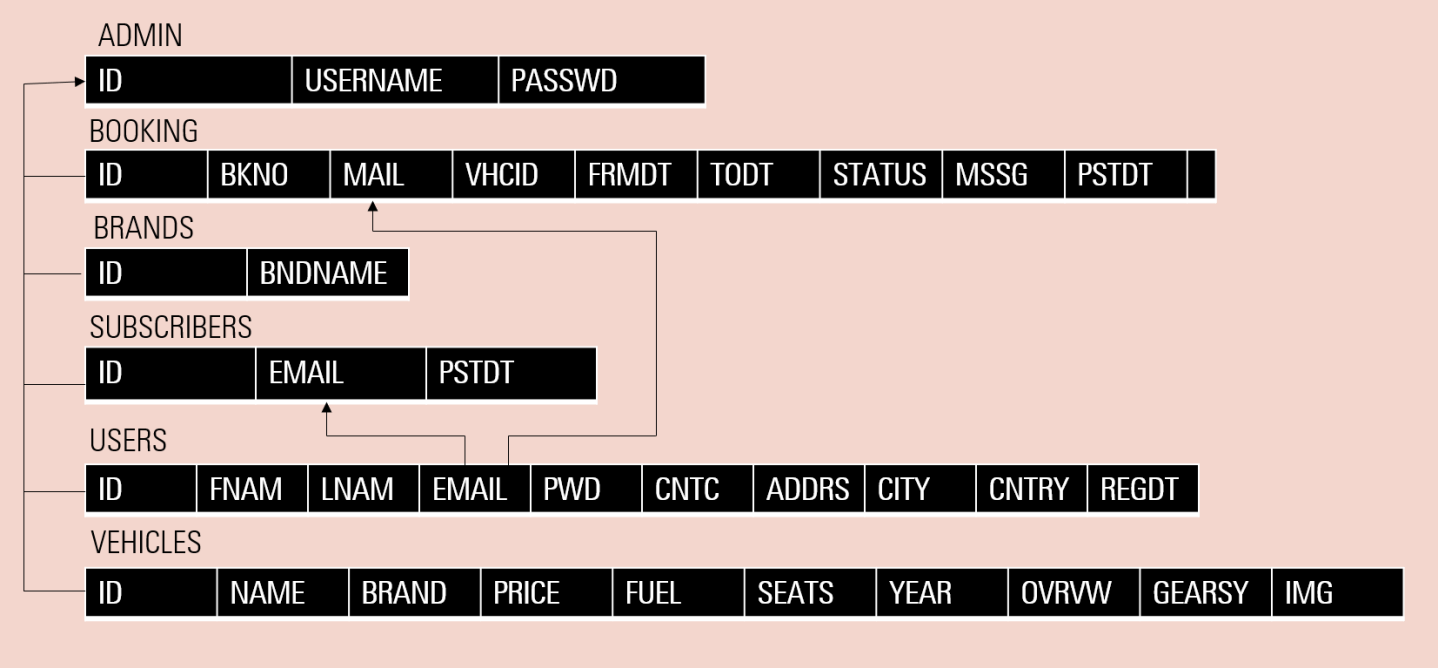
ER DIAGRAM

An entity–relationship model (ER model) describes inter-related things of interest in a specific domain of knowledge. An ER model is composed of entity types (which classify the things of interest) and specifies relationships that can exist between instances of those entitytypes. In software engineering an ER model is commonly formed to represent things that a airline needs to remember in order to perform airlines processes. Consequently, the ER modelbecomes an abstract data model that defines a data or information structure that can be implemented in a database, typically a relational database.

An entity–relationship model is usually the result of systematic analysis to define and describe what is important to processes in an area of a airlines. An E-R model does not define the airlines processes; it only presents a airlines data schema in graphical form. It is usually drawn in a graphical form as boxes (entities) that are connected by lines (relationships) which express the associations and dependencies between entities. An ER model can also be expressed in a verbal form, for example: one building may be divided into zero or more apartments, but one apartment can only be in one building. Entities may be characterized not only by relationships, but also by additional properties (attributes), which include identifiers called "primary keys".

Diagrams created to represent attributes as well as entities and relationships may be called entity attribute-relationship diagrams, rather than entity-relationship models. An ER model is typically implemented as a database. In a simple relational database implementation, each row of a table represents one instance of an entity type, and each field in a table represents an attribute type. In a relational database a relationship between entities is implemented by storing the primary key of one entity as a pointer or "foreign key" in the table of another entity. There is a tradition for ER/data models to be built at two or three levels of abstraction. Note that the conceptual-logical- physical hierarchy below is used in other kinds of specification, and is different from the three- schema approach to software engineering.

3.3 SCHEMA:



SCHEMA DIAGRAM

A schema diagram is a diagram which contains entities and the attributes that will define that schema. A schema diagram only shows us the database design. It does not show the actual data of the database. Schema can be a single table or it can have more than one table which is related. The schema represents the relationship between these tables. The term "schema" refers to the organization of data as a blueprint of how the database is constructed (divided into database tables in the case of relational databases). The formal definition of a database schema is a set of formulas (sentences) called integrity constraints imposed on a database. A relational schema shows references among fields in the database. When a primary key is referenced table in the database, it is called a foreign key. This is denoted by an arrow with the head pointing at the referenced key attribute.

Entities and Attributes:

1) ADMIN: id, Username, password
2) BOOKING: id, mail, vehicle id, from date, to date, booked id
3) VEHICLES: id, name, brand, price, fuel, seats
4) BRANDS: plane id, brand name
5) USER: id, name, password, email, address, city, contact NO

CHAPTER 4

SYSTEM IMPLEMENTATION

Implementation is the stage in the project where the theoretical design is turned into a working system and is giving confidence on the new system for the users that it will work efficiently and effectively. It involves careful planning, investigation of the current system and its constraints on implementation, design of methods to achieve the changeover, an evaluation of change over methods. Implementation is the most important phase. The most critical stage in achieving a successful new system is giving the users confidence that the new system will work and be effective. Any system developed should be secured and protected against possible hazards. Security measures are provided to prevent unauthorized access of the database at various levels. Password protection and simple procedures to prevent the unauthorized access are provided to the users. The system allows the user to enter the system only through proper user name and password.

4.1 ADMIN LOGIN CODE:

This page is used to login where this facility is available only to the administrator to check in the ongoings of the airlines booked and its details.

```
 <?php
session_start();
include('includes/config.php');
if(isset($_POST['login']))
{
$email=$_POST['username'];
$password=md5($_POST['password']);
$sql ="SELECT UserName,Password FROM admin WHERE UserName=: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->fetchAll(PDO::FETCH_OBJ);
if($query->rowCount() > 0)
{
$_SESSION['alogin']=$_POST['username'];
```

```
echo "<script type='text/javascript'> document.location = 'dashboard.php'; </script>";
} else{


placeholder="Username" name="username" class="form-control mb">

<label for="" class="text-uppercase text-sm">Password</label>
<input type="password"
placeholder="Password" name="password" class="form-control mb">


<button class="btn btn-primary btn-block" name="login" type="submit">LOGIN</button>

</form>

<p style="margin-top: 4%" align="center"><a href="../index.php">Back to Home</a>          </p>
</div>

</div>

</div>
</div>
```

4.2 HOME PAGE CODE:

This page is also handled by the administrator itself. It has the customers information and registers about them in this page.

```
<?php
session_start();
include('includes/config.php');
error_reporting(0);

?>

<!DOCTYPE HTML>
<html lang="en">
<head>

<title>Car Rental Portal</title>
<!--Bootstrap -->
<link rel="stylesheet" href="assets/css/bootstrap.min.css" type="text/css">
<link rel="stylesheet" href="assets/css/style.css" type="text/css">
<link rel="stylesheet" href="assets/css/owl.carousel.css" type="text/css">
<link rel="stylesheet" href="assets/css/owl.transitions.css" type="text/css">
<link href="assets/css/slick.css" rel="stylesheet">
<link href="assets/css/bootstrap-slider.min.css" rel="stylesheet">
<link href="assets/css/font-awesome.min.css" rel="stylesheet">
        <link rel="stylesheet" id="switcher-css" type="text/css" href="assets/switcher/css/switcher.css"
media="all" />
        <link rel="alternate stylesheet" type="text/css" href="assets/switcher/css/red.css" title="red" media="all"
data-default-color="true" />
        <link rel="alternate stylesheet" type="text/css" href="assets/switcher/css/orange.css" title="orange"
media="all" />
        <link rel="alternate stylesheet" type="text/css" href="assets/switcher/css/blue.css" title="blue"
media="all" />
        <link rel="alternate stylesheet" type="text/css" href="assets/switcher/css/pink.css" title="pink"
media="all" />
        <link rel="alternate stylesheet" type="text/css" href="assets/switcher/css/green.css" title="green"
media="all" />
        <link rel="alternate stylesheet" type="text/css" href="assets/switcher/css/purple.css" title="purple"
media="all" />
<link rel="apple-touch-icon-precomposed" sizes="144x144" href="assets/images/favicon-icon/apple-touch-icon-
144-precomposed.png">
<link rel="apple-touch-icon-precomposed" sizes="114x114" href="assets/images/favicon-icon/apple-touch-icon-
```

```
114-precomposed.html">
<link rel="apple-touch-icon-precomposed" sizes="72x72" href="assets/images/favicon-icon/apple-touch-icon-72-precomposed.png">
<link rel="apple-touch-icon-precomposed" href="assets/images/favicon-icon/apple-touch-icon-57-precomposed.png">
<link rel="shortcut icon" href="assets/images/favicon-icon/favicon.png">
<link href="https://fonts.googleapis.com/css?family=Lato:300,400,700,900" rel="stylesheet">
</head>
<body>

<!-- Start Switcher -->
<?php include('includes/colorswitcher.php');?>
<!-- /Switcher -->

<!--Header-->
<?php include('includes/header.php');?>
<!-- /Header -->

<!-- Banners -->
<section id="banner" class="banner-section">
<div class="container">
<div class="div_zindex">
<div class="row">
<div class="col-md-5 col-md-push-7">
<div class="banner_content">
<h1> </h1>
<p>  </p>
</div>
</div>
</div>
</div>
</div>
</section>
```

4.3 USER PAGECODE:

This page is also handled by the administrator itself

```
<?php
session_start();
include('includes/config.php');
error_reporting(0);
?>

<!DOCTYPE HTML>
<html lang="en">
<head>

<title>Car Rental Portal | Car Listing</title>
<!--Bootstrap -->
<link rel="stylesheet" href="assets/css/bootstrap.min.css" type="text/css">
<!--Custome Style -->
<link rel="stylesheet" href="assets/css/style.css" type="text/css">
<!--OWL Carousel slider-->
<link rel="stylesheet" href="assets/css/owl.carousel.css" type="text/css">
<link rel="stylesheet" href="assets/css/owl.transitions.css" type="text/css">
<!--slick-slider -->
<link href="assets/css/slick.css" rel="stylesheet">
<!--bootstrap-slider -->
<link href="assets/css/bootstrap-slider.min.css" rel="stylesheet">
<!--FontAwesome Font Style -->
<link href="assets/css/font-awesome.min.css" rel="stylesheet">

<!-- SWITCHER -->
<link rel="stylesheet" id="switcher-css" type="text/css" href="assets/switcher/css/switcher.css" media="all" />
<link rel="alternate stylesheet" type="text/css" href="assets/switcher/css/red.css" title="red" media="all" data-
default-color="true" />
<link rel="alternate stylesheet" type="text/css" href="assets/switcher/css/orange.css" title="orange" media="all" />
<link rel="alternate stylesheet" type="text/css" href="assets/switcher/css/blue.css" title="blue" media="all" />
<link rel="alternate stylesheet" type="text/css" href="assets/switcher/css/pink.css" title="pink" media="all" />
<link rel="alternate stylesheet" type="text/css" href="assets/switcher/css/green.css" title="green" media="all" />
<link rel="alternate stylesheet" type="text/css" href="assets/switcher/css/purple.css" title="purple" media="all" />

<!-- Fav and touch icons -->
<link rel="apple-touch-icon-precomposed" sizes="144x144" href="assets/images/favicon-icon/apple-touch-icon-
144-precomposed.png">
<link rel="apple-touch-icon-precomposed" sizes="114x114" href="assets/images/favicon-icon/apple-touch-icon-
```

```
114-precomposed.html">
<link rel="apple-touch-icon-precomposed" sizes="72x72" href="assets/images/favicon-icon/apple-touch-icon-72-precomposed.png">
<link rel="apple-touch-icon-precomposed" href="assets/images/favicon-icon/apple-touch-icon-57-precomposed.png">
<link rel="shortcut icon" href="assets/images/favicon-icon/favicon.png">
<link href="https://fonts.googleapis.com/css?family=Lato:300,400,700,900" rel="stylesheet">
</head>
<body>

<!-- Start Switcher -->
<?php include('includes/colorswitcher.php');?>
<!-- /Switcher -->

<!--Header-->
<?php include('includes/header.php');?>
<!-- /Header -->

<!--Page Header-->
<section class="page-header listing_page">
<div class="container">
<div class="page-header_wrap">
<div class="page-heading">
<h1>Car Listing</h1>
</div>
<ul class="coustom-breadcrumb">
<li><a href="#">Home</a></li>
<li>Car Listing</li>
</ul>
</div>
</div>
<!-- Dark Overlay-->
<div class="dark-overlay"></div>
</section>
<!-- /Page Header-->

<!--Listing-->
<section class="listing-page">
<div class="container">
<div class="row">
<div class="col-md-9 col-md-push-3">
<div class="result-sorting-wrapper">
<div class="sorting-count">
<?php
```

```
//Query for Listing count
$sql = "SELECT id from tblvehicles";
$query = $dbh -> prepare($sql);
$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 tblvehicles.*,tblbrands.BrandName,tblbrands.id as bid  from tblvehicles join tblbrands on
tblbrands.id=tblvehicles.VehiclesBrand";
$query = $dbh -> prepare($sql);
$query->execute();
$results=$query->fetchAll(PDO::FETCH_OBJ);
$cnt=1;
if($query->rowCount() > 0)
{
foreach($results as $result)
{  ?>
<div class="product-listing-m gray-bg">
<div class="product-listing-img"><img src="admin/img/vehicleimages/<?php echo htmlentities($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>


<!--Register-Form -->
<?php include('includes/registration.php');?>

<!--/Register-Form -->

<!--Forgot-password-Form -->
<?php include('includes/forgotpassword.php');?>
```

```
<!-- Scripts -->
<script src="assets/js/jquery.min.js"></script>
<script src="assets/js/bootstrap.min.js"></script>
<script src="assets/js/interface.js"></script>
<!--Switcher-->
<script src="assets/switcher/js/switcher.js"></script>
<!--bootstrap-slider-JS-->
<script src="assets/js/bootstrap-slider.min.js"></script>
<!--Slider-JS-->
<script src="assets/js/slick.min.js"></script>
<script src="assets/js/owl.carousel.min.js"></script>

</body>
</html>


<?php
// DB credentials.
define('DB_HOST','localhost');
define('DB_USER','root');
define('DB_PASS','');
define('DB_NAME','carrental');
// Establish database connection.
try
{
$dbh = new PDO("mysql:host=".DB_HOST.";dbname=".DB_NAME,DB_USER,
DB_PASS,array(PDO::MYSQL_ATTR_INIT_COMMAND => "SET NAMES 'utf8'"));
}
catch (PDOException $e)
{
exit("Error: " . $e->getMessage());
}
?>
```

CHAPTER 5

DATABASE TECHNIQUES

5.1 DATABASE MANAGEMENTSYSTEM

A database-management-system is a computer-software application that interacts with end- users, other applications, and the database itself to capture and analyse data. A general- purpose DBMS allows the definition, creation, querying, update and administration of databases.

5.1.1 The CREATE TABLE Command in SQL

- The CREATE TABLE command is used to specify a new relation by giving it a name and specifying its attributes and initial constraints.
- The attributes are specified first, and each attribute is given a name, a data type to specify its domain of values, and any attribute constraints, such as NOTNULL.
- The key, entity integrity, and referential integrity constraints can be specified within the CREATE TABLE statement after the attributes are declared, or they can be added later using the ALTER TABLE command.
- The schema name can be explicitly attached to the relation name, separated by a period.
- Syntax: create table <table name> (<attribute1><type1>,<attribute2><type2>);

5.1.2 The INSERT Command

- INSERT is used to add a single tuple to a relation. The relation name and a list of values arespecified for the tuple.
- The values should be listed in the same order in which the corresponding attributes were specified in the CREATE TABLE command.
- A second form of the INSERT statement allows the user to specify explicit attribute namesthat correspond to the values provided in the INSERT command.
- This is useful if a relation has many attributes but only a few of those attributes are assignedvalues in the new tuple.

- Syntax: 1. insert into <table name> values(_value1‘,‘value2‘);

  2. insert into <table name> (<attribute1>,<attribute2>) values (_value1‘,‘value2‘);

5.1.3 The DELETE Command

- The DELETE command removes tuples from a relation.
- It includes a WHERE clause, to select the tuples to be deleted.
- Tuples are explicitly deleted from only one table at a time
- A missing WHERE clause specifies that all tuples in the relation are to be deleted but thetable remains in the database as an empty table.
- Syntax: delete from <table name> where<condition>;

5.1.4 The UPDATE Command

- The **UPDATE** command is used to modify attribute values of one or more selected tuples.
- A WHERE clause in the UPDATE command selects the tuples to be modified from a single relation.
- A **SET** clause in the UPDATE command specifies the attributes to be modified and their new values.
- Syntax: update <table name> set <attribute>=<new value> where<attribute>=<value>;

5.1.5 Unspecified WHERE Clause and Use of the Asterisk

- A missing WHERE clause indicates no condition on tuple selection. All tuples of the relation specified in the FROM clause qualify and are selected for the query result.
- If more than one relation is specified in the FROM clause and there is no WHERE clause, then the CROSS PRODUCT-all possible tuple combinations-of these relations is selected.
- To retrieve all the attribute values of the selected tuples, an asterisk (*) can be specified, which stands for all the attributes.

5.2 TRIGGER

A trigger is a stored procedure in database which automatically invokes whenever a special event the database occurs. For example, a trigger can be invoked when a row is inserted into a specified table or when certain table columns are being updated

```
CREATE TRIGGER ‘login_ins’ AFTER INSERT ON ‘admin’
 FOR EACH ROW BEGIN
INSERT into admin values(NEW.username,NEW.password);
 END
```

Incrementing after new value added to the admin table

5.3 STORED PROCEDURES

- A stored procedure is a program that is executed through a single SQL statement that can be locally executed and completed within the process space of the database server.
- The results can be packaged into one big result and returned to the application, or the application logic can be performed directly at the server, without having to transmit the results to the client.
- Stored procedures are also beneficial for software engineering because once a stored procedure is registered with the database server, different users can re-use the stored procedure, eliminating duplication of efforts in writing SQL queries or application logic, and making code maintenance easy.

5.4 NORMALIZATION

Database normalization, or simply normalization, is the process of organizing the columns (attributes) and tables (relations) of a relational database to reduce data redundancy and improve data integrity. Normalization is also the process of simplifying the design of a database so that it achieves the optimal structure composed of atomic elements.

Normalization involves arranging attributes in relations based on dependencies between attributes, ensuring that the dependencies are properly enforced by database integrity constraints Normalization is accomplished by applying some formal rules either by a process of synthesis or decomposition. Synthesis creates a normalized database design based on a known set of dependencies. Decomposition takes an existing (insufficiently normalized) database design and improves it based on the known set of dependencies.

5.4.1 FIRST NORMALFORM

First normal form is an essential property of a relation in a relational database. Database normalization is the process of representing a database in terms of relations in standard normal forms,where first normal is a minimal requirement.

First normal form enforces these criteria:

- Eliminate repeating groups in individual tables.
- Create a separate table for each set of related data.
- Identify each set of related data with primary key.

5.4.2 SECOND NORMALFORM

A relation that is in first normal form (1NF) must meet additional criteria if it is to qualify for second normal form. Specifically: a relation is in 2NF if it is in 1NF and no non-prime attribute is dependent on any proper subset of any candidate key of the relation. A non-prime attribute of a relation is an attribute that is not a part of any candidate key of the relation.

5.4.3 THIRD NORMALFORM

Third normal form is a normal form that is used in normalizing a database design to reduce the duplication of data and ensure referential integrity by ensuring that (1) the entity is in second normal form, and (2) all the attributes in a table are determined only by the candidate keys of that relation and not by any non-prime attributes. 3NF was designed to improve database processing while minimizing storage costs. 3NF data modelling was ideal for online transaction processing (OLTP) applications with heavy order entry type of needs.

5.4.4 FOURTH NORMALFORM

Fourth normal form (4NF) is a normal form used in database normalization. Introduced by Ronald Fagin in 1977, 4NF is the next level of normalization after Boyce–Codd normal form(BCNF). Whereas the second, third, and Boyce–Codd normal forms are concerned with functional dependencies, 4NF is concerned with a more general type of dependency known as a multi valued dependency. A table is in 4NF if and only if, for every one of its non-trivial multi valued dependencies X, Y, X is a super-key—that is, X is either a candidate key or a superset.

5.4.5 FIFTH NORMALFORM

A database is said to be in 5th Normal Form, if and only if, it is in 4th Normal Form and if we decompose the table further to eliminate redundancy and anomaly and when we rejoin the decomposed tables by means of candidate keys we should not be losing the original data or any new record set should not arise. In simple words joining two or more decomposed tables should not lose records or create new records.

5.5 TESTING

The purpose of testing is to discover errors. Testing is the process of trying to discover every convincible fault or weakness in a work product. It provides a way to check the functionality of components, sub-assemblies, assemblies, and/or finished product. It is the process of exercising software with the intent of ensuring that the software system meets its requirements and user expectations and does not fail in an unacceptable manner.

LEVELS OF TESTING:

5.5.1 UNIT TESTING:

Unit testing is a method by which individual units of source code, sets of one or more computer program modules together with associated control data, usage procedures and operating procedures. For unit testing first we adopted the code testing strategy, which examined the logic of program. During the development process itself all the syntax errors get rooted out. For this developed test casethat result in executing every instruction in the program.

5.5.2 USER ACCEPTANCE TESTING:

User acceptance testing of the system is the key factor for the success of the system. A system under consideration is tested for user acceptance by constantly keeping in touch with the prospective system at the time of development and making change whenever required. This is done regarding the input screen design and output screen design.

5.5.3 GUI Testing:

GUI testing is use to ensure the visual clarity of the system, flexibility of the system, user friendliness of the system. The various components which are to be tested are:

- Relative layout
- Various Link and Buttons

5.5.4 VALIDATION TESTING:

At the culmination of black box testing, software is completely assembled is a package. Interfacing errors have uncovered and the correct and final states of tests i.e. validation is defined with a simple definition that validation succeeds when the software function in a manner that can be reasonably accepted by the customer.

5.5.5 OUTPUT TESTING:

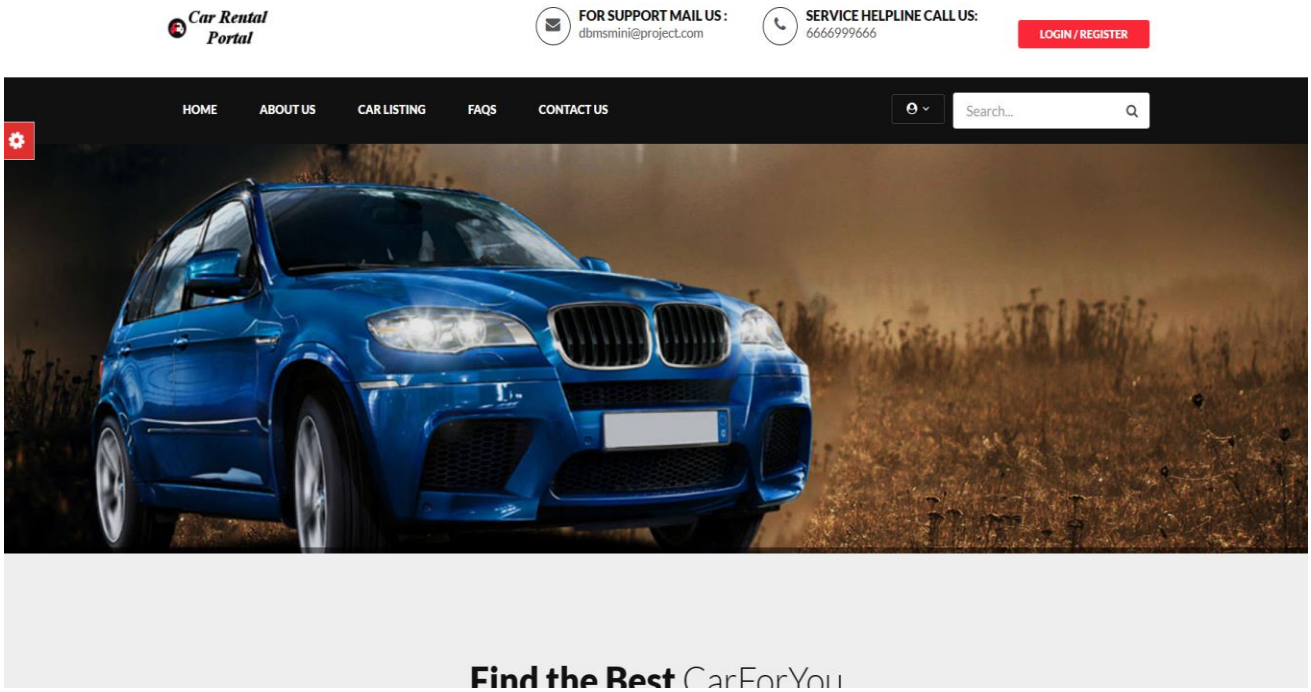
After performing validation testing, the next step is output testing of the proposed system. Since the system cannot be useful if it does not procedure the required output. Asking the user about the user about this required format in which the system is required tests the output displayed or generated by the system under consideration.

Chapter 6

SCREENSHOTS

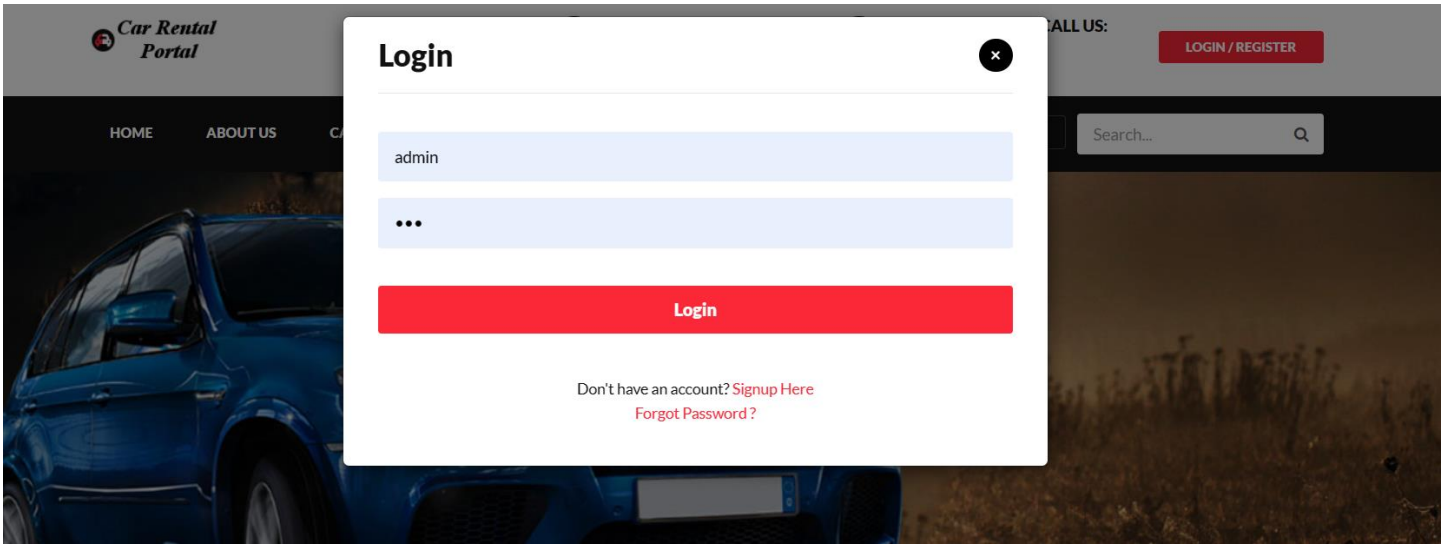
User Login Page

.

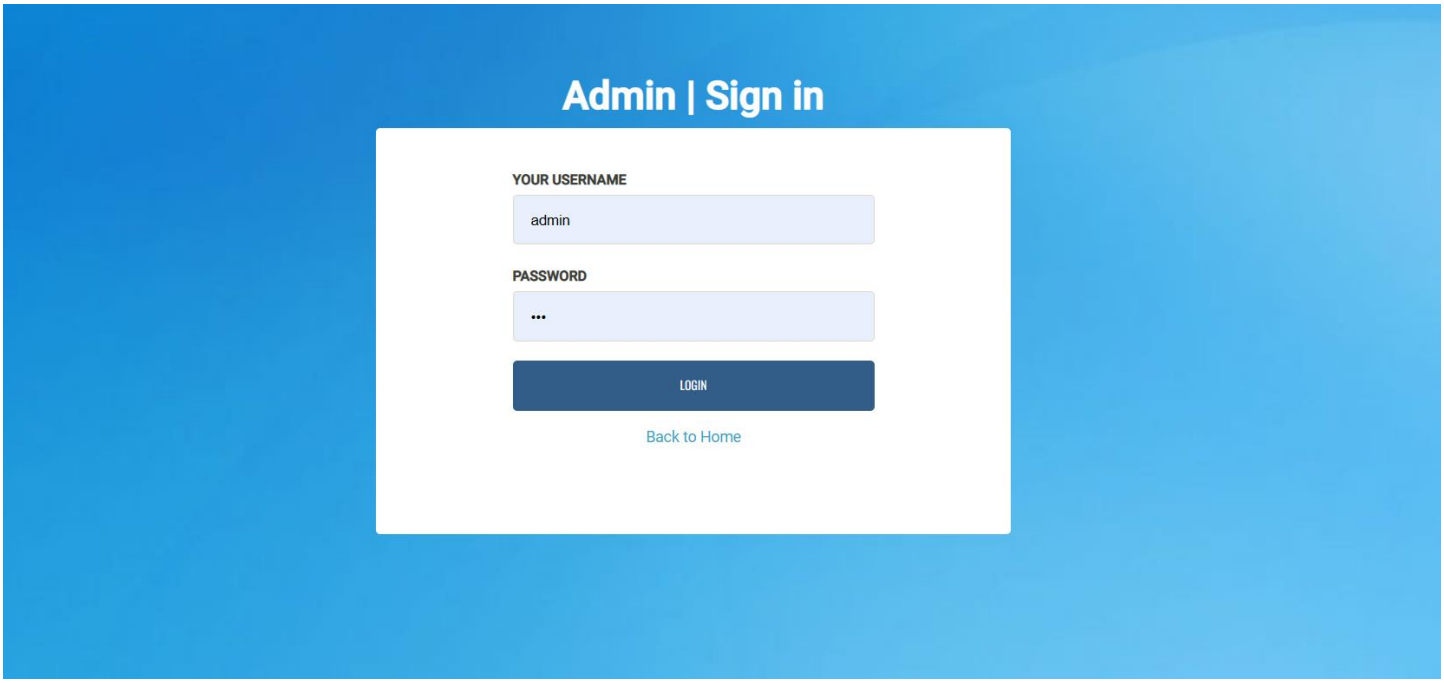


Description: Login page allows the admin user to login with their respective credential.

User login



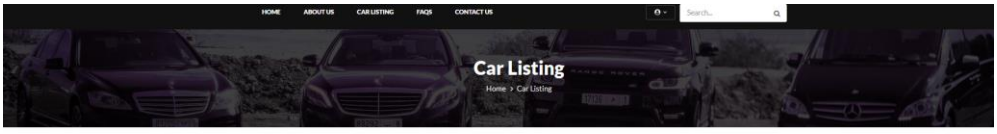
Admin page

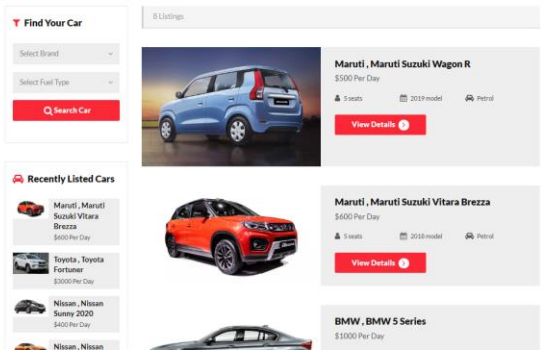


Description: User page allows the admin to login with their respective credential

car_list Page

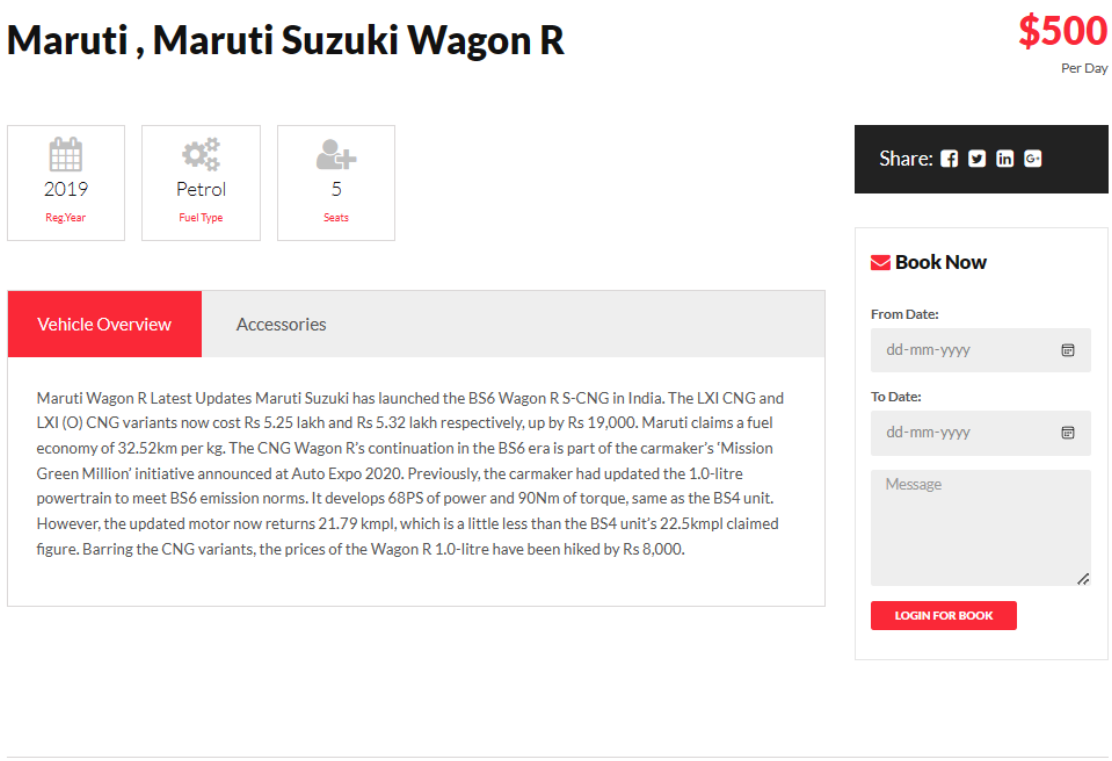
Description: It contains the details of available car for rent with details





Manage bookings

Description: Here, we can add the details of car rented



CONCLUSION AND FUTURE SCOPE

Car rental business has emerged with a new goody 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 these functions are achieved has been reshaped by the power of internet. Nowadays, customers can reserve cars online, rent car online, and have the car brought to their door step once the customer is a registered member or go to the office to pick the car. The 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 at the click of a button.

BIBLIOGRAPHY AND REFERENCES

Books Used:

• Software Engineering - R.S. Pressman
• PHP For Dummies •
PHP Begineers Guide By McGrawhill Publication
• Javascript By McGrawhill Publication

References Used:

• http://www.carrentingsolutions.com/
• http://www.flashvortex.com/
• http://www.imscart.com/car_rental_software.html
• Wikipedia.org
• www.w3schools.c