

MADI MOVERS (Online Vehicle Rental and Mover System)

Md Imran Khan, Anurag Yadav, Mukesh Kumar Prajapati

Md Imran Khan, Information Technology, IMS Engineering College, UP, India

Anurag Yadav, Information Technology, IMS Engineering College, UP, India

Mukesh Kumar Prajapati, Information Technology, IMS Engineering College, UP, India

Dr. Avdhesh Gupta (Ass. Prof.), Dept. of I.T., IMS Engineering College, U.P, India

-----***-----

ABSTRACT: *The inspiration behind this project is the growing popularity of online website systems and the requirement to explore the messaging (message notification) technology that industries could tap into to enhance their services to the customers. This research paper explained a notification-based content alert and web-based system using short message service (SMS) technology. It is going intentionally developed for the alert notification to the customers about the vehicle rental information, and the availability of the vehicle reserved. The main objective of developing SMS based content alert for vehicle rental system is to reduce the cost and time consumed, which is beneficial to the vehicle rental agencies and customers. Therefore, the system was designed automatically to send an alert SMS to the customers about the availability of the vehicle reserved. This portal is developed based on System Development Life Cycle (SDLC) using the waterfall model as a methodology.*

The online vehicle reservations which gives much benefit to user. A rental portal is a portal in which customers arrive to request the hire of a rental unit. It is more easy than carrying the cost of owning and maintaining the unit.

In conclusion, it is a system design specially for large, premium and small vehicle rental business. The vehicle rental portal provides complete functionality of (managing) listing and booking.

Key Words: SMS based, Vehicle Rental Portal, Computerized, Google Chrome, time efficiency.

1. INTRODUCTION

1.1. Project Overview:

This project is created to be used by vehicle rental company specializing in rental vehicles to customers. It is an online platform through which customers can view available vehicles, register, view profile and book vehicles.

Madi Mover (vehicle rental portal) is a web-based system for the users that rents out vehicles. This platform enables the users to make their vehicles available to the public through the internet and also keep records about their vehicles. Many people became interested in the vehicle rental business and hence got involved.

1.2. Objectives:

- MADI Mover app enables users to locate, unlock and pick up it's scooters cars and bikes at one point, and drop them off at a different point, without the need for a docking station. Our long-term goal is to turn into a 100-city play with dense penetration in key cities.

- **MADI Mover** is resolving the problems of consumers whether it is about to lessen hassles in transportation or simply cut out the expenses of commuting, which is generally one of the major concerns in people's lives. With **Bounce** it is not only simple to commute by being simply available anywhere but cost-effective too.

2. LITERATURE REVIEW

- **Bounce** is India's first smart urban mobility solution, with a mission of making daily commute stress-free, time-saving, reliable and convenient. **Bounce** offers One Way Rental service with the all-new keyless scooters, cars, where you can pick up and drop the bike anywhere you need to. Taking inspiration from bounce we are developing MADI Movers with some special functionalities that user can post their vehicle as well.
- Vogo is a dock less scooter rental company that lets customers rent scooters for short one-way trips. The company's VOGO box is attached to each scooter and lets customers access the key without any human intervention and start riding. Vogo is the fastest commute option available today in India at a fraction of the cost of other alternatives. Vogo is currently available at limited pick-up and drop-off points across Bangalore and Hyderabad.

The inspirational idea is extracted from **bounceapp.com and vogo.in**

3. SYSTEM ANALISYS

3.1 Existing Systems: -

- An existing system provides lots of paper work that is manual.
- The user has to go at the place (office) where user can get the vehicle on rent and book their vehicle.
- In the existing system one (user) cannot provide their feedback to the admin online.
- In this system a lot of time wastes because of much manual work.

3.2 Need for New Systems(Proposed System): -

- A new system provides features like time efficiency to show vehicles details, userprofiles and whatever the customer will give the feedback to the admin.
- The new system is totally automated system(computerized).
- This system provides tourism and travelling facilities.

- An enquiry is easily done by user in the system using this platform.

3.3 Feasibility Study: -

The feasibility study is one of the most important thing at the early stage of the project development .This stage requires a lot of efforts to understand the main aspect of the system answer to the question “ Is the project feasible?” in this stage the possibilities of project development is identified by conducting technical and operational feasibility of the proposed system.

3.3 Technical Feasibilities: -

Technical feasibility corresponds to determination of whether it is technically feasible to develop the software. It refers to the ability of the process to take advantage of the current state of the technology in pursuing further improvement. The technical capability of the available technology should be considered.

3.4 Operational Feasibility: -

Proposed projects are beneficial only if they can be turned into information system that will meet the organization’s operating requirement. The following areas declare the proposed project as operationally feasible.

- There is sufficient support from the management and from the intended users of the systems.
- As its backbone developing for all the modules it was necessary that all other people developing other modules agree and due to their support it is operationally feasible.

4. FUNCTIONAL SPECIFICATION

4.1. User Specification: -

- **Admin:**

Admin can add a vehicle, manage booking vehicle and rent and also view feedback and enquiry.

- **User:**

User can view information of available vehicle, booking vehicle, easily get the vehicle on rent and also give feedback and can enquiry.

4.2 Module Specification: -

- **Check Available vehicles:**

It is a platform designed especially for large, premium and small vehicles rental business. The user can check available vehicles and user can book for that vehicles.

- **Booking Vehicles:**

The user can Check available vehicles and user can book for that vehicles.

- **Easily Get the Vehicles on rent:**

The Customer can easily get the vehicle whenever they need to on the rent with use of this system.

- **Give Feedback:**

The customer will give the feedback to the admin.

- **Enquiry:**

The inquiry can easily done by user.

- **Add Vehicle:**

The admin can add the vehicle so that the user can check(view) the available cars and book the vehicle.

- **Manage Rent:**

The admin can manage the rent so that the user can check(view) the rent and book the vehicle.

- **View Feedback:**

The admin easily view the feedbacks and solve the query.

5. SYSTEM SPECIFICATION

5.1 Hardware Requirements:

- **Client Side:**

- RAM: 512 MB
- HARD DISK: 10 GB
- PROCESSOR: 1.0 GHz

- **Server side:**

- RAM: 1 GB
- HARD DISK: 20 GB
- PROCESSOR: 2.0 GHz

5.2 Software Requirements:

➤ Client Side:

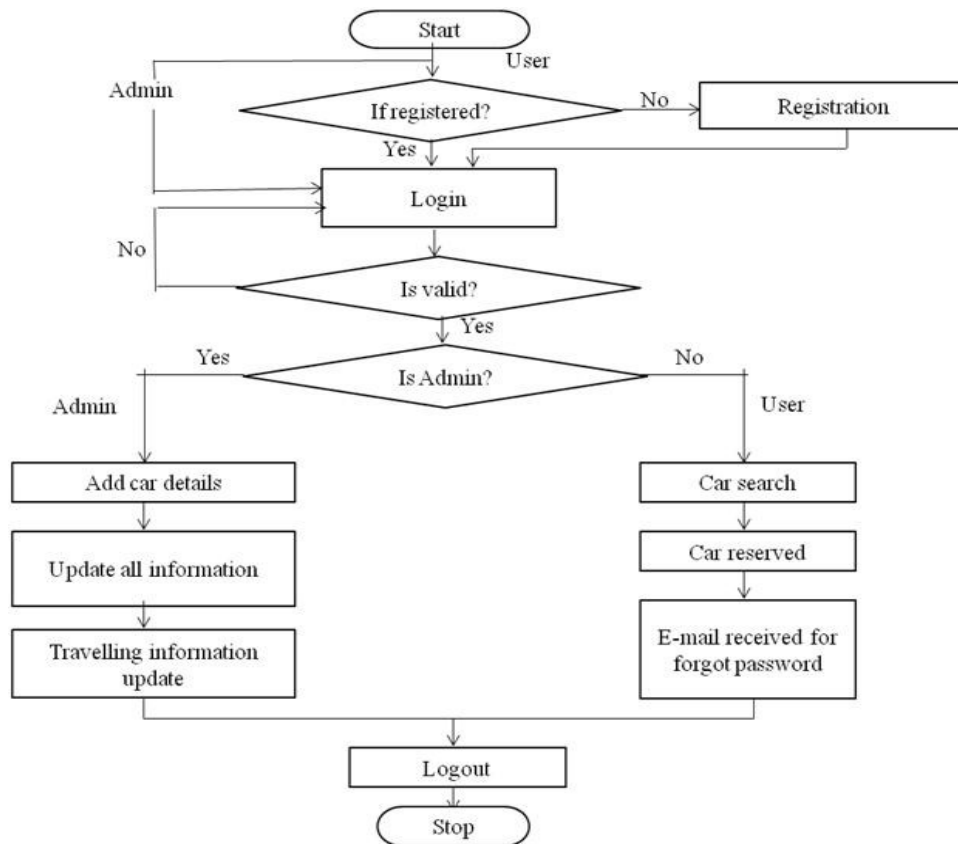
- **Web Browser:** Internet Explorer 6 or above version or any compatible browser (Google Chrome, etc).
- **Operating System:** Windows or any equivalent OS.

➤ Server Side:

- **Web Server:** IIS 7.5
- **Framework:** .NET 4.0 with C#
- **Database Server:** MS SQL Server 2008
- **Web Browser:** Google Chrome or any compatible browser
- **Operating System:** Windows Server 2007.

6. PROJECT DESCRIPTION

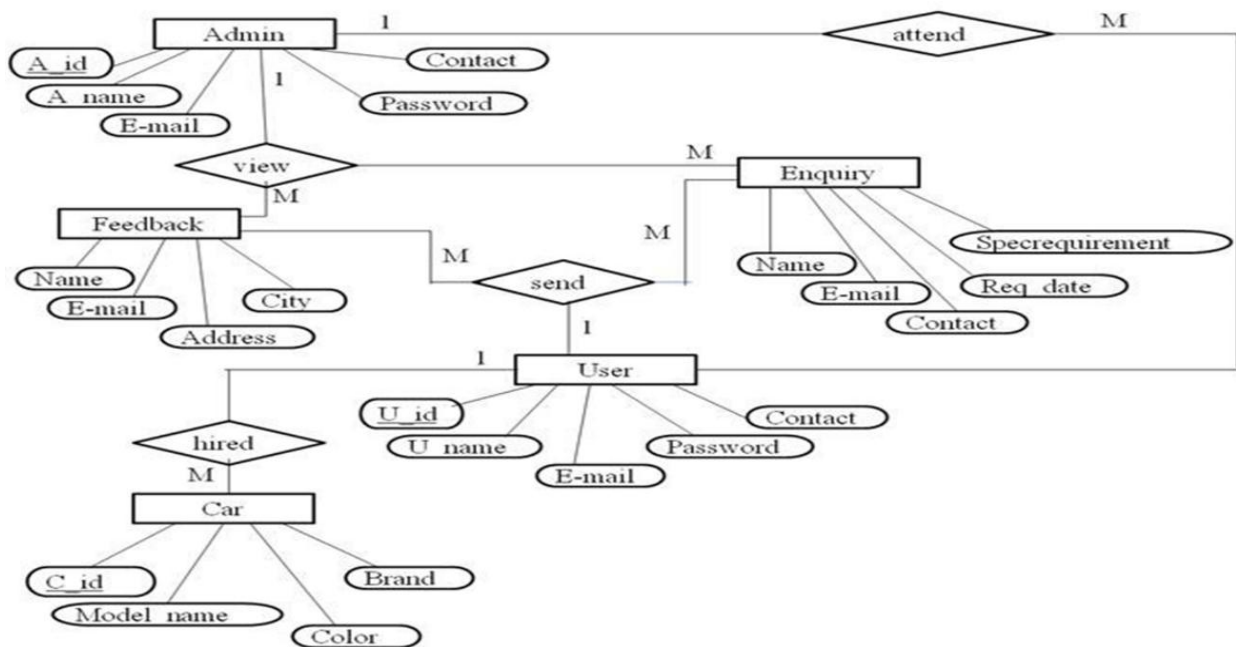
6.1. System Flowchart:



6.2. Time Line Chart:

Development Phase	90 Days						Duration (Day)
	0 to 15 Day	16 to 30 Day	31 to 45 Day	46 to 60 Day	61 to 75 Day	76 to 90 Day	
Requirement Gathering	█						10
Analysis		█					15
Design		█	█	█			30
Coding				█	█		25
Testing					█	█	12
Implementation						█	08
Documentation		█	█	█	█	█	80
Total Time (Days)							90

6.3. E-R Diagram:



6.4. DATA DICTIONARY

Admin:

Table Name	Adminis
Description	This table is store information about Adminis
Primary Key	A_id

Foreign Key	-
--------------------	---

Sr. No	Field Name	Data type(Size)	Constraints	Description
1	Adm_id	Int	Primary Key	It stores Admin id.
2	Adm_Name	Varchar (10)	Not Null	It stores admin name.
3	Adm_Password	Varchar (20)	Not Null	It stores the password of Admin.
4	Adm_Contact_No	Varchar (12)	Not Null	It stores the Contact no of Admin.
5	Adm_E_mail	Varchar (70)	Not Null	It stores the Email id of Admin

User Registration:

Table Name	UserM
Description	This table is provide the information about user(customer) registration
Primary Key	U_Id
Foreign Key	-

Sr. No	Field Name	Data type(Size)	Constraints	Description
1	Vehicle_Id	Int	Primary Key	It stores vehicle_id
2	Model_Name	Varchar(25)	Not Null	It stores model name of vehicle.
3	Brand_Name	Varchar(25)	Not Null	It stores brand name of the vehicle.

4	Color_Name	Varchar(15)	NotNull	It stores color name of the car
5	No_of_Pas	Varchar(15)	NotNull	It stores information of passenger Seats
6	Price	Int	Not Null	It stores price of the car
7	Image	Varchar(60)	Not Null	It stores images of car
8	Fuel_type	Varchar(20)	Not Null	It stores fuel type in car

Booking Table:

Table Name	Booking_vehicle
Description	This table is provide the information about booking the vehicle.
Primary Key	-
Foreign Key	vehicle_Id

Sr. No	Field Name	Data type(Size)	Constraints	Description
1	vehicle_Id	Int	Foreign Key	It references to vehicle_Id from vehicle.
2	Name	Varchar(20)	Not Null	It stores name of the vehicle.
3	Date	DateTime	Not Null	It stores booking date of vehicle.
4	S_address	Varchar(50)	Not Null	It stores source address.
5	D_address	Varchar(50)	Not Null	It stores information about destination address.
6	Email_Id	Varchar(50)	Not Null	It stores email address.

7	No_Pieces	Int	Not Null	It stores information how many vehicle. available in time
8	Contact_No	Varchar(12)	Not Null	It stores contact_no of the user.

Enquiry Table:

Table Name	Enquiry
Description	This table will store the information of car enquiry of user
Primary Key	E_Id
Foreign Key	-

Sr. No	Field Name	Data type(Size)	Constraints	Description
1	E_Id	Int	Primary Key	It stores enquiry id.
2	User_Name	Varchar(20)	Not Null	It stores user name.
3	Email_Id	Varchar(50)	Not Null	It stores email id.
4	Contact_No	Varchar(12)	Not Null	It stores contact no for user.
5	Date_From	DateTime	Not Null	Display date of which the car should take.
6	To_Date	DateTime	Not Null	Display date that is required date will come.
7	Query	Varchar(50)	Not Null	It store query of the user.

Feedback Table:

Table Name	Feedback
Description	This table store information about feedback

Primary Key	F_Id
Foreign Key	-

Sr. No	Field Name	Data type(Size)	Constraints	Description
1	F_Id	Int	Primary Key	It stores feedback id.
2	F_Name	Varchar(20)	Not Null	It stores user name.
3	Email_Id	Varchar(50)	Not Null	It stores email id.
4	Address	Varchar(50)	Not Null	It stores user address.
5	City	Varchar(20)	Not Null	It stores user city name.
6	Postal_Code	Varchar(6)	Not Null	It stores postal code.
7	Phone_No	Varchar(12)	Not Null	It stores phone no of user.
8	Service	Varchar(20)	Not Null	It stores service of user required.
9	Opinion	MAX	Not Null	It stores user opinion.

Payment Table:

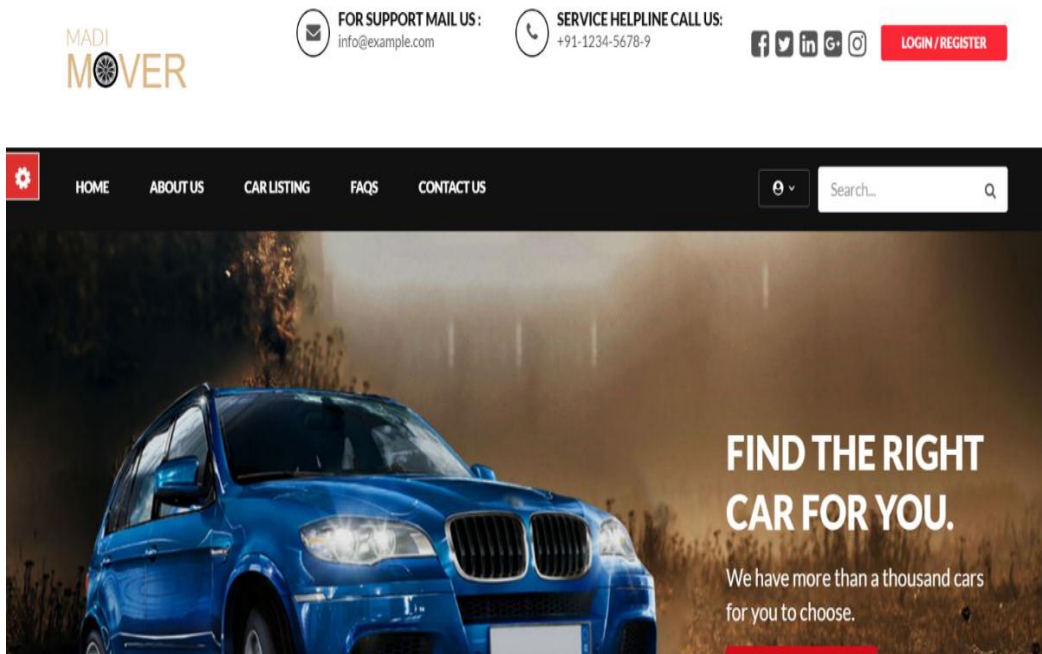
Table Name	Payment
Description	This table store information about payment
Primary Key	P_Id
Foreign Key	-

Sr. No	Field Name	Data type(Size)	Constraints	Description
1	P_Id	Int	Primary Key	It stores Payment id of user.
2	Car_Name	Varchar(20)	Not Null	It stores vehicle name.
3	Email	Varchar(70)	Not Null	It stores email id for user.
4	Amount	Int	Not Null	It stores Amount of vehicle rent.

7. DESIGN

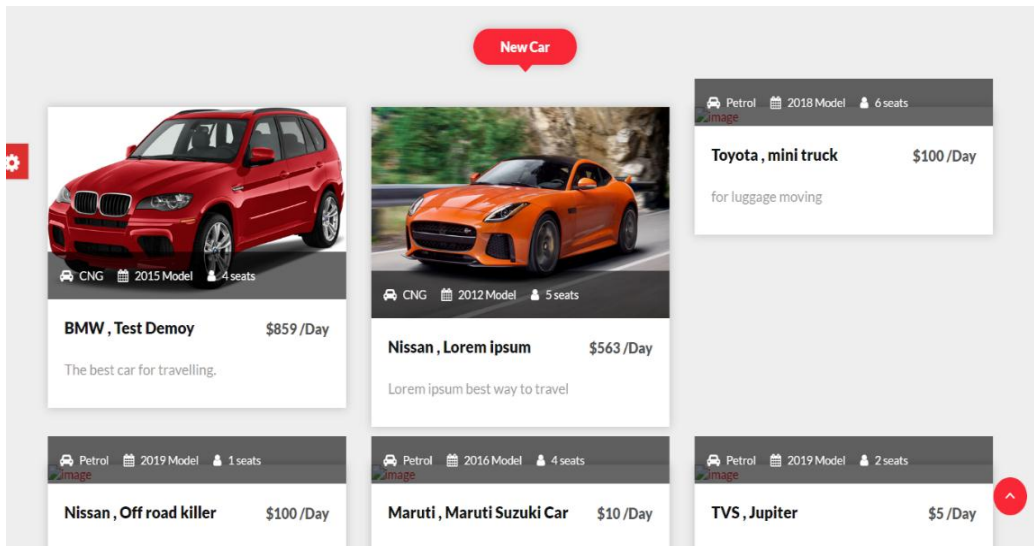
7.1. INPUT DESIGN

➤ **User Home Page:**



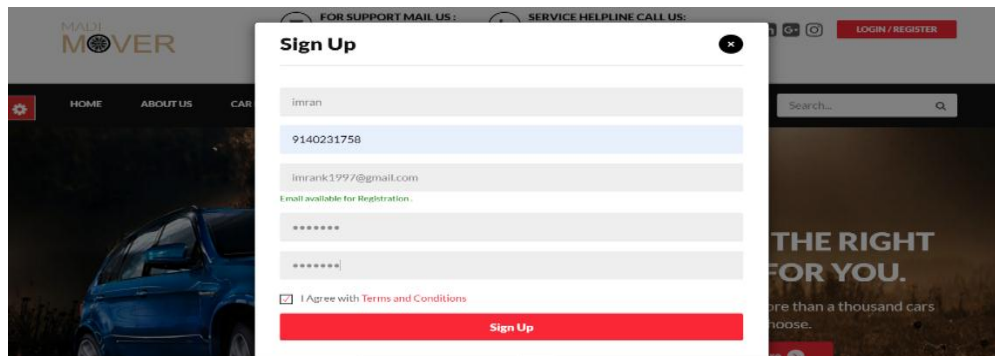
- This is home page of user.

➤ Service Page:

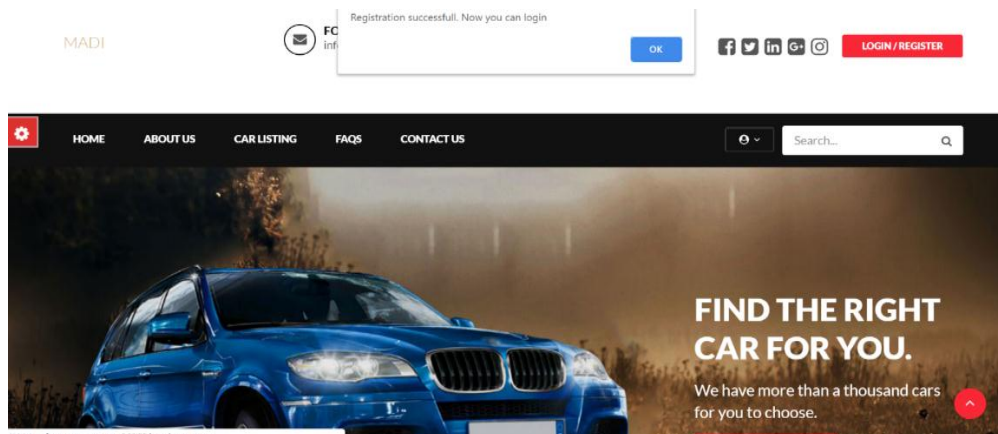


- This is service page of user.

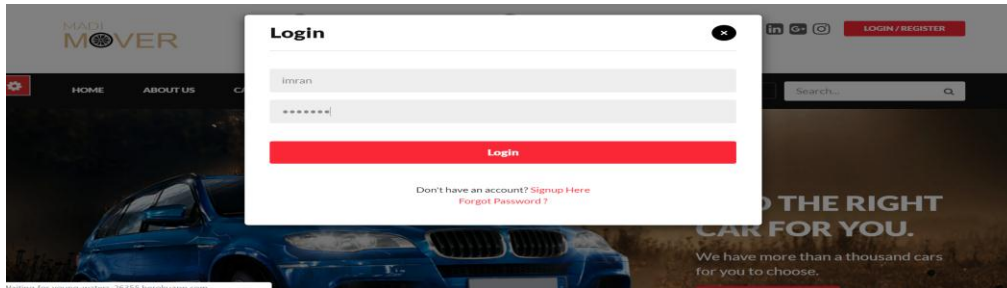
➤ Registration validation page:



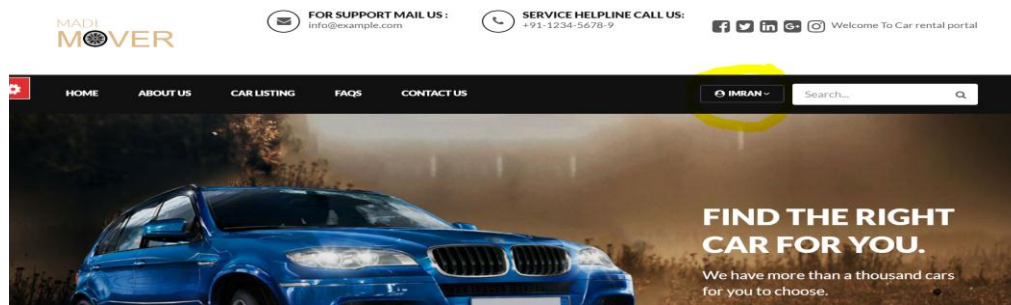
➤ Registration done successfully:



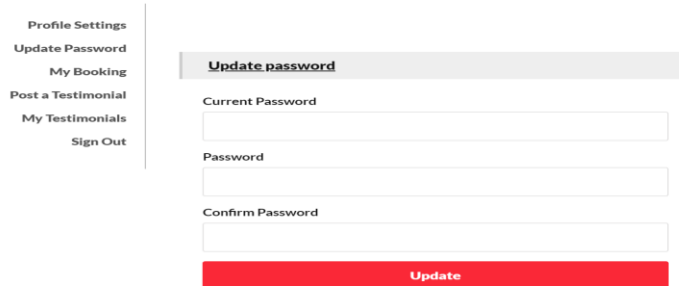
➤ Login Page:



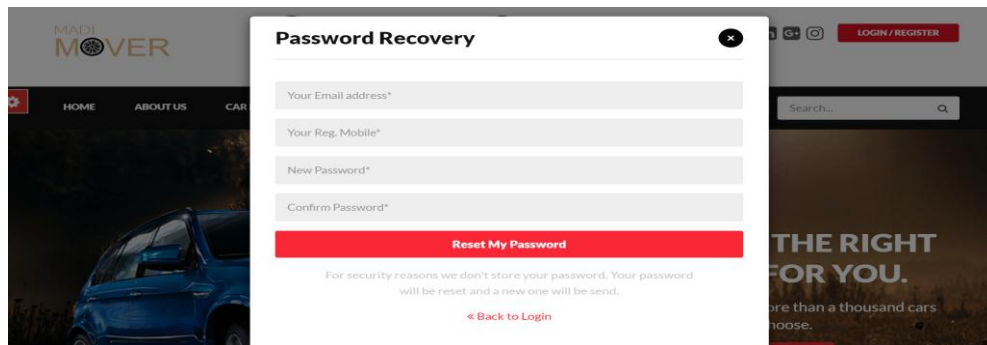
➤ Successfully login:



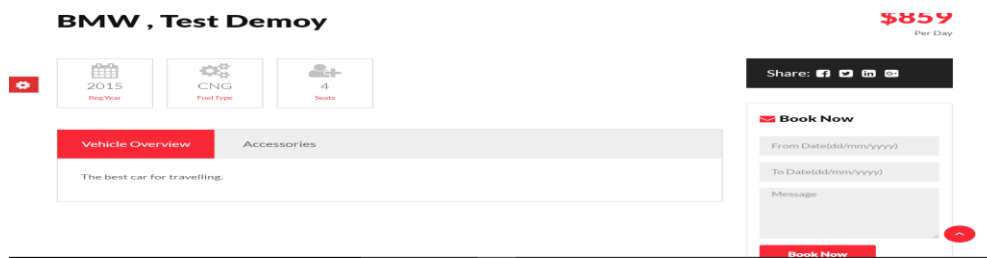
➤ Updating password:



➤ Forgot Password Page:



➤ Booking Page:

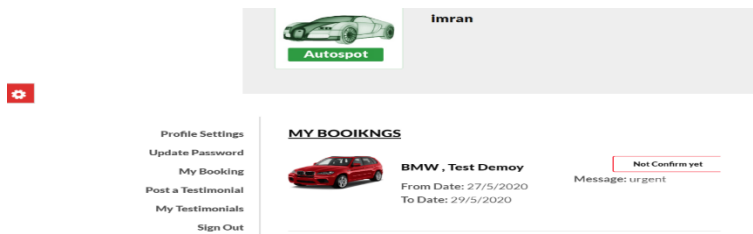


➤ Booking Successfully page:



7.2. OUTPUT DESIGN

➤ Booking Confirmation:



➤ User Registration Report:

id	fullname	emailid	contactno	address	12-11-2014	birthdate
1	dsd	dsd@gmail.com	121	dsada		dsd
2	dsd	dsd@gmail.com	121	dsada		dsd
3	dsd	dsd@gmail.com	121	dsada		dsd
4	dsd	dsd@gmail.com	121	dsada		dsd
5	dsd	dsd@gmail.com	121	dsada		dsd
6	dsd	dsd@gmail.com	121	dsada		dsd
7	dsd	dsd@gmail.com	121	dsada		dsd
8	dsd	dsd@gmail.com	121	dsada		dsd
9	dsd	dsd@gmail.com	121	dsada		dsd
10	dsd	dsd@gmail.com	121	dsada		dsd
11	dsd	dsd@gmail.com	121	dsada		dsd
12	dsd	dsd@gmail.com	121	dsada		dsd
13	dsd	dsd@gmail.com	121	dsada		dsd
14	dsd	dsd@gmail.com	121	dsada		dsd
15	dsd	dsd@gmail.com	121	dsada		dsd
16	dsd	dsd@gmail.com	121	dsada		dsd
17	dsd	dsd@gmail.com	121	dsada		dsd
18	dsd	dsd@gmail.com	121	dsada		dsd
19	dsd	dsd@gmail.com	121	dsada		dsd
20	dsd	dsd@gmail.com	121	dsada		dsd
21	dsd	dsd@gmail.com	121	dsada		dsd
22	dsd	dsd@gmail.com	121	dsada		dsd
23	dsd	dsd@gmail.com	121	dsada		dsd
24	dsd	dsd@gmail.com	121	dsada		dsd
25	dsd	dsd@gmail.com	121	dsada		dsd
26	dsd	dsd@gmail.com	121	dsada		dsd
27	dsd	dsd@gmail.com	121	dsada		dsd
28	dsd	dsd@gmail.com	121	dsada		dsd
29	dsd	dsd@gmail.com	121	dsada		dsd
30	dsd	dsd@gmail.com	121	dsada		dsd
31	dsd	dsd@gmail.com	121	dsada		dsd
32	dsd	dsd@gmail.com	121	dsada		dsd
33	dsd	dsd@gmail.com	121	dsada		dsd

➤ **Vehicle Report:**

id	fullname	emailid	contactno	address	birthdate
2	dsa	dsasd	121	dzada	sdas
3	dhurj	as@dfgh.com	1249	kfo	27/3/344
4	falgun	fg@gmail.com	8785	mehsana	09/13/1998
9	rohani	rg@gmail.com	10101010	surat	2/2/1992
10	prjanka	pg@gmail.com	12121212	stff	27/3/344
11	kruca	kg@gmail.com	12345678	mehsana	31/8/92
12	Patel Kajal	kj@gmail.com	12345	10/gfk	21/3/1993
13	patel ki	ki@gmail.com	1234567890	Silakff	21/4/1992
14	khushbu	khuc@yahoo.com	9824210556	mehsana	16-4-1993
15	Rohit	rcp1357@gmail.com	9890586854		
16	sonu patel	ksu@7914@yahoo.com	9879146260	sardardham	15-07-1990
17	sonu patel	khushbupatel778@yahoo.com	9890586854	sardardham	
18	sonu patel	khushbupatel778@yahoo.com	9879146260	sardardham	19-01-1990
19	sonu patel	khushbupatel778@yahoo.com	9879146260	sardardham	19-01-1990
20	sonu patel	khushbupatel778@yahoo.com	9879146260	sardardham	19-01-1990
21	sonu patel	khushbupatel778@yahoo.com	9879146260	sardardham	19-01-1990
22	sonu patel	khushbupatel778@yahoo.com	9879146260	sardardham	19-01-1990
23	sonu patel	khushbupatel778@yahoo.com	9879146260	sardardham	19-01-1990
24	sonu patel	khushbupatel778@yahoo.com	9879146260	sardardham	19-01-1990
25	sonu patel	khushbupatel778@yahoo.com	9879146260	sardardham	19-01-1990
26	sonu patel	khushbupatel778@yahoo.com	9879146260	sardardham	19-01-1990
27	patel sonu	rcp1357@gmail.com	987	asfa	19-01-1990
28	patel palak	abc111@gmail.com	9876543210	sardardham	19-01-1990
29	patel palak	abc111@gmail.com	9879146260	saibag	19-01-1990
30	patel kiran	kiran123@yahoo.com	9876543210	saibag	12-02-2014
31	khushbu patel	kh123@gmail.com	9879146260	saibag	12-02-2014
32	Rohit Patel	rohi@yahoo.com	9879146260	sardardham	12-02-1992
33	Khushbu Patel	khushbu123@yahoo.com	9876543210	sardardham	08-09-1997

8. SYSTEM TESTING

The aim of the system testing process was to determine all defects in our project. The program was subjected to a set of test inputs and various observations were made and based on these observations it will be decided whether the program behaves as expected or not. Our project went through two levels of testing i.e. unit testing and acceptance testing.

8.1 Unit Testing:

Unit testing is undertaken when a module has been created and successfully reviewed. In order to test a single module, we need to provide a complete environment.

8.2 Acceptance Testing:

The purpose of Acceptance Testing (AT) is to ensure that the solution by the project meets the functional and non-functional requirements specified in the business requirements. Acceptance Testing is the final step before rolling out the solution. Acceptance Testing is typically carried out by end users in an environment that closely models the real world. A well-managed Acceptance Testing process will give the Project Sponsor, project team and end users confidence that the solution being delivered meets the requirements.

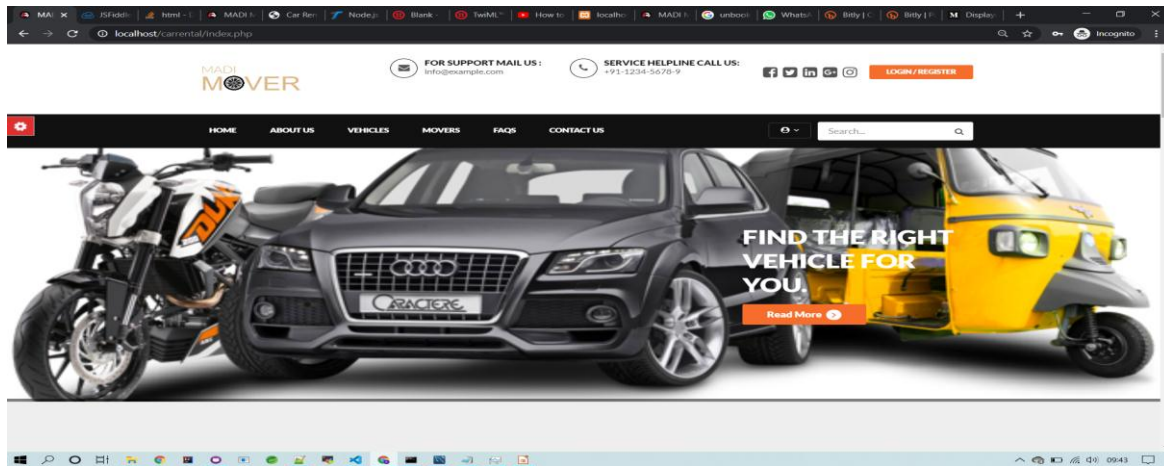
8.3 Test cases:

No.	Data I/P	Excepted O/T	Actual O/T	Pass / Fail
1	All files are empty	Error message: "indicates compulsory field(attribute)"	Error message: "indicates compulsory field(attribute)"	Pass

2	Email	Error message: Invalid Email-add.	Error_message: Invalid Email-address	Pass
3	Password and confirm_password	Error message: Both password does not match	Error message: Both password do not match	Pass
4	Login	Login to the portal should be try with the login assigned by the admin and the correct password	Login should be successful and the user should enter into the portal.	Fail
		The portal give an error and denied from the Login.	Login should fail with an error "Invalid_Details"	Pass
5	User	Login should be allow and admin get Admin home page.	Login successfully and admin get its admin home Page	Pass
		Login should be allow and Travel admin get Travel admin home page.	Login successfully and Travel admin get its Travel admin home page	Pass
		Login should be allow and User get Visitor side User page.	Login successfully and User gets its user page.	Pass

6	Validation Test cases	Pre-define format must be required in control	System give error to enter valid input	Pass
		Enter data in a compulsory field with required field validations.	Data must be field in compulsory field otherwise its messages are displayed.	Pass

9. RESULTS



10. CONCLUSIONS

In this research paper we have presented some insight on user technology to construct and integrating the web-based system with messaging technology to improve the service provided by the vehicle rental company.

In conclusion, the integration of web-based and SMS technology along with the android app in the vehicle rental portal is the best way to take the advantages of today technology, in order to scale the productivity and efficiency of organization. In real days, messaging(SMS) has been chosen by many users and has in fact, become extremely popular. Despite their restrictions, mobiles devices, especially mobile phones have become a natural part of the everyday life of a huge number of people, especially the younger generation growing up with computing and Internet Technologies.

11. FUTURE SCOPE OF STUDY

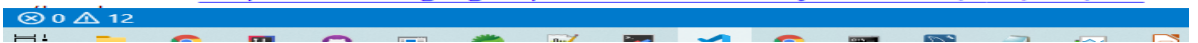
- Because of creaky price of fuels, even the vehicles owners also moving to rental system to cut out the expenses.
- Growth of phones and telecommunication system is also the reason for the growth of vehicle rental system.
- Money sharing options and exclusive ways to choose their vehicle.

12. APPENDIX(Important codes only):

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

?>

<!DOCTYPE HTML>
<html lang="en">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">
<meta http-equiv="X-UA-Compatible" content="IE=edge">
<meta name="viewport" content="width=device-width,initial-scale=1">
<meta name="keywords" content="">
<meta name="description" content="">
<title>MADI MOVER</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 rel="stylesheet" href="assets/css/style2.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/s
<link rel="alternate stylesheet" type="text/css" href="assets/switcher/
<link rel="alternate stylesheet" type="text/css" href="assets/switcher/
<link rel="alternate stylesheet" type="text/css" href="assets/switcher/
<link rel="alternate stylesheet" type="text/css" href="assets/switcher/
<link rel="alternate stylesheet" type="text/css" href="assets/switcher/
<link rel="apple-touch-icon-precomposed" sizes="144x144" href="assets/image
<link rel="apple-touch-icon-precomposed" sizes="114x114" href="assets/image
<link rel="apple-touch-icon-precomposed" sizes="72x72" href="assets/images/
<link rel="apple-touch-icon-precomposed" href="assets/images/favicon-icon/a
<link rel="shortcut icon" href="assets/images/favicon-icon/favicon.png">
<link href="https://fonts.googleapis.com/css?family=Lato:300,400,700,900" r
```



```
const express= require("express");
const parser= require("body-parser");

var app= express();
app.listen('3000',()=>{
  console.log("servr stated on port 3000");
});

app.use(parser.json());
app.use(parser.urlencoded({ extended: true }));

const accountId = 'AC306b756764d42acbcd357edb51adc349';
const authToken = '2368c2f358ed9299d64cd6a9009c629c';
const client = require('twilio')(accountId, authToken);

function say(){
  console.log("under say()");
}

app.get("/",(req,res)=>{
  client.messages
    .create({
      from: 'whatsapp:+14155238886',
      body: 'Hello there!',
      to: 'whatsapp:+919717679964'
    })
    .then(message => console.log(message.sid));
});

app.post("/prompt",(req, res)=>{
  const id=req.query.id;
  const vehicleid=req.query.vehicleid;
  const status=req.body.status;
  const photo=req.body.photo;
  console.log(id+"\n"+vehicleid+"\n"+status+"\n"+photo);
  res.send("thank you!!");
});
```

```
<body onload="xyyy()">
  <div class="OOOform-container">
    <form action="" method="post" class="OOOform">
      <input type="text" value="" name="status" id="OOOhide" />
      <button class="OOObutton OOObutton1" onclick="acceptAction()">ACCEPT</button>
      <button class="OOObutton OOObutton2" onclick="ignoreAction()">IGNORE</button>
    </form>
  </div>
  <script src="nodejs/private3.js" ></script>
  <script>
    function xyyy(){
      //say();
    }
    function parseURLParams(url) {
      var queryStart = url.indexOf("?") + 1,
          queryEnd   = url.indexOf("#") + 1 || url.length + 1,
          query      = url.slice(queryStart, queryEnd - 1),
          pairs      = query.replace(/\+/g, "%").split("&"),
          parms      = {}, i, n, v, nv;

      if (query === url || query === "") return;

      for (i = 0; i < pairs.length; i++) {
        nv = pairs[i].split("=", 2);
        n  = decodeURIComponent(nv[0]);
        v  = decodeURIComponent(nv[1]);

        if (!parms.hasOwnProperty(n)) parms[n] = [];
        parms[n].push(nv.length === 2 ? v : null);
      }
      return parms;
    }
    function acceptAction(){
      var params= parseURLParams(window.location.search);
      var url="http://localhost:3000/prompt?id="+params.id[0]+"&vehicleid="+params.vehicleid[0];
      //console.log(url);
    }
  </script>

```

13. REFERENCES

- www.code-project.com
- www.bounceapp.com
- www.olacabs.com
- www.vogo.in