A

PROJECT REPORT

ON

ORVRS

SUBMITED
TO
SAVITRIBAI PHULE PUNE UNIVERSITY, PUNE

FOR THE PARTIAL FULLFILMENT OF

MASTER OF COMPUTER APPLICATION

(MCA-I, SEM.- II)

BY

Ravi Kumar

UNDER THE GUIDANCE OF

Prof. Yogesh Sharma

THROUGH

THE DIRECTOR SINHGAD INSTITUTE OF MANAGEMENT AND COMPUTER APPLICATION (SIMCA),
NARHE, PUNE
(AY. 2021-2022)

SINHGAD TECHNICAL EDUCATION SOCIETY'S

SINHGAD INSTITUTE OF
MANAGEMENT & COMPUTER APPLICATION

(Affiliated to Savitribai Phule Pune University &Approved by (AICTE)

'NAAC' Accredited with 'A' Grade

S. No. 49/1, Off Westernly Bypass, Pune – Mumbai Expressway, Narhe, Pune – 411041,
Tel : (020) 66831896 / 66831908 / 66831907
E-mail : director_mca_simca@sinhgad.edu Website : www.sinhgad.edu

Prof. M. N. Navale	**Dr. (Mrs.) Sunanda M. Navale**	**Dr. Vilas Nandavadekar**
M.E. (Elec), MIE, MBA	B.A., P.P.M., Ph.D.	M.C.A., M.P.M., M.B.A., Ph. D.
FOUNDER PRESIDENT	FOUNDER SECRETARY	DIRECTOR, SIMCA - MCA

CERTIFICATE

This is to certify that, the project entitled **ORVRS (On Road Vehicle Repairing System)** being submitted for the Partial Fulfillment of the degree of **Master of Computer Application** by him to **Sinhgad Institute of Management and Computer Application affiliated to Savitribai Phule Pune University, Pune** is the result of the original work completed by ***Ravi Kumar*** under the guidance of ***Prof. Yogesh Sharma***

To the best of our knowledge and belief, this work has not been previously submitted by the any degree or diploma of Savitribai Phule Pune University or any other University.

PLACE: Pune

DATE:

Prof. Yogesh Sharma	**Dr. Ashwini Brahme**	**Dr. Vilas Nandavadekar**
Internal Guide	**Project Co-Ordinator**	**Director SIMCA-MCA**

External Examiner

DECLARATION

I, the undersigned hereby declare that the project titled **ORVRS** being submitted for the partial fulfillment of degree of **Master of Computer Application** by me to **Sinhgad Institute of Management and Computer Application (SIMCA) affiliated to Savitribai Phule Pune University** is the result of an independent work carried out under the guidance of **Prof. Yogesh Sharma,** is my original work. Further I declare that this project has not been submitted to this or any Institution for the award of any degree.

PLACE: PUNE Ravi Kumar

DATE:

ACKNOWLEDGEMENT

The project developed for the MCA was not possible without the persons and organizations that helped me in completing this. I am deeply grateful to all whose enthusiasm and energy transformed my vision of this study into reality.

I take this opportunity to thank my guide **Prof. Yogesh Sharma**, project coordinator **Dr. Ashwini Bramhe** and our Director **Dr. Vilas Nandavadekar**, for encouragement and guidance throughout the progress of this report.

Ravi Kumar

1. Introduction

Under the direction of Prof. Yogesh Sharma, Ravi Kumar created the On Road Vehicle Repairing System (ORVRS). The On Road Vehicle Repairing System website was created with the intention of offering the ability to repair vehicles on the spot when they have issues. There are three dashboards on this website: an admin dashboard, a user dashboard, and a mechanic dashboard. On each, the capability for their individual needs has been supplied in accordance with the requirements of the system. The administrator of this website has been given an interface via which he may examine user information, user requests, and mechanic information. Administrators can manage both users and mechanics for any system-required function. For the first time, the user of this system must register their car with the correct information about both the car and themselves. After this, the user can log in and visit the dashboard, where he has been given the facility for the car and from which he can request service from the mechanic for the repair of his car. Additionally, that request will be reflected on the mechanic's dashboard, and the mechanic will respond. Here, the mechanic must register himself with the appropriate personal information and the name of the repair facility where he works. After registering, he must check in using his credentials before being forwarded to his dashboard, where he may manage requests that have been mirrored from the user side to it. Here, the technician will respond to the user before visiting the appropriate location to diagnose and fix the problem with the car.

1.1 Abstract

- ORVRS – On Road Vehicle Repairing System Website is designed primarily for repairing the vehicle on the spot or place where it got some problem.
- User will be allowed to request for the repair of vehicle where it has encountered some problem.
- This system will make user more flexible for the repairing of his or her vehicle.
- In this system, user can register himself and create account and will be able to request for service.

1.2 Existing System and Need for System

Today, a vehicle is necessary for our everyday lives. Most individuals utilize various types of transportation to keep visiting new locations. One way for a user to travel to new and beautiful locations while operating his automobile more conveniently and pleasantly. When using a car to reach to one's location, one might feel more at ease and be more productive at work.

Many person encounter car problems while on the road, making it impossible for them to reach the repair facility in such a state. The capacity to fix a car on the spot where a person has a problem with their vehicle is currently unavailable from service Centre. No one wants to wait around when travelling, but since it takes a while to notify the mechanic and another while for the mechanic to respond, people are forced to wait around for hours or even days. We built a website where users won't have to wait a long time to solve this problem.

Our website is designed to offer the service of fixing the car on the road or the location where it encountered any issues. The user can request the repair of his car at any time while on the road without being interrupted.

1.3 Scope of System

This website is specifically designed to offer automobile repair services close to where the vehicle has experienced a problem. In this case, the individual does not have to wait a long time to call the mechanic and receive a response. Users must register their vehicles on this website with their personal information before they may submit a service request from the dashboard, where several skilled mechanics are available. In only a few seconds, a user may submit a request, and just as quickly, a mechanic will react.

1.4 System Requirement

Hardware Requirements**:**

Hardware Specification: - Processor Core i3 or higher

System Bus: - 64 bits

RAM: - 4GB

HDD: - 512GB

Monitor: - LCD Monitor

Software requirements**:**

Operating System: - Window 10

Code Editor: - V. S. Code Editor

Front End: - HTML, CSS, JavaScript

Back End: - Node.js, Express.js

Database: - MongoDB

2. Proposed System

2.1 Study of Similar Systems

The demands of users were not being met by the currently operational System. Users now have issues when travelling to destinations or offices. When the user is away from the service facility, they are unable to phone the mechanic. Contacting a mechanic or repair facility from the location where the incident occurred may take a while.
User need to get their vehicle repaired at spot whenever required. But

This website is mainly focused on the user's requirement for vehicle repairing. Our website plays a crucial role in supplying the mechanic for the repairing of the vehicle at the location where it has issue because the majority of mechanics are unable to provide the service at the place where the vehicle has some issue and there is no mechanic for the repairing of the vehicle at that time. That's why user has to face a big problem which take user lot of time for the repairing of his vehicle. And he is unable to reach the destination at a time due to which user has to face a big loss in his or her work.

As a result, the user won't have to wait long for the mechanic to arrive at the location. Anyone may easily utilise this website and benefit from the service. This website is providing the facilities for user to request service for the mechanic on one click and response immediately. And solve the problem of the customer related to the vehicle. And user will be free from the current problem of his vehicle repairing.

2.2 Feasibility Study

Feasibility study can help you determine whether or not you should proceed with your project. It is essential to evaluate cost and benefit. It is essential to evaluate cost and benefit of the proposed system. Four types of feasibility study are taken into consideration.

Technical feasibility: It includes finding out technologies for the project, both hardware and software. For On Road Vehicle Repairing System, user must have phone or pc to use this application and active internet connection for the proper use of the system. These are becoming the need of a person, now a days everyone generally possesses them. Besides, system needs internet connection.

Operational feasibility: It is the ease and simplicity of operation of proposed system. System does not require any special skill set for users to operate it. In fact, it is designed to be used by almost everyone. It is too simple to use it, user having only knowledge of interacting with internet can operate it.

Economical feasibility: Here, we find the total cost and benefit of the proposed system over current system. For this project, the main cost is documentation cost. User also would have to pay for the required system and tools. Again, they are cheap and available. As far as maintenance is concerned, these won’t cost too much.

Organizational feasibility: This shows the management and organizational structure of the project. This project is not built by a team. The management tasks are all to be carried out by a single person. That won’t create any management issues and will increase the feasibility of the project.

2.3 Objectives of Proposed System

- To analyse people’s, need and provide the service they require.
- To provide the service which made by this website.
- To get more requirements of the services what they need.
- To create jobs and generate more revenue.
- To provide user friendly interface for easy usability of the mechanic and user.
- To provide the services for the user in more efficient way.
- To provide the user-friendly interfaces for the mechanic too.

3. Analysis and Design

3.1 System Requirements

- The On Road Vehicle Repairing System that we are putting out here makes fixing a car a lot easier. It offers the user the ability to register, log in, and request that the car be repaired where the problem is. Users may choose from a group of mechanics according to the type of vehicle they need, whether it be a two-wheeler or something larger. Any difficulty the consumer is having will be resolved when the mechanic arrives on the scene.

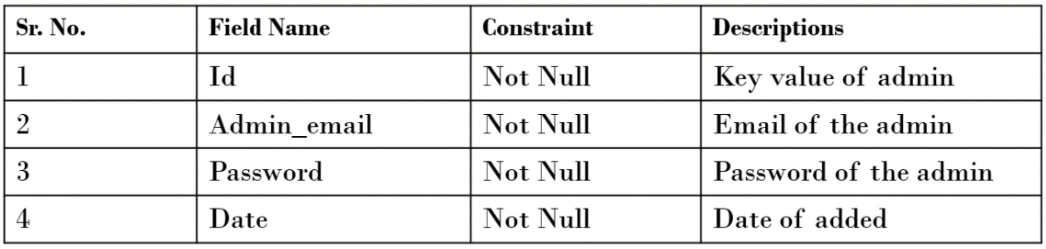
- System presents an interactive and up-to-date data about services and mechanic who are responsible for the repairing the vehicle.

- Users can use the website to submit a service request from their home or from the place where their vehicle is having trouble.

- After logging in and registering, a user can request the service that his car needs from the location.

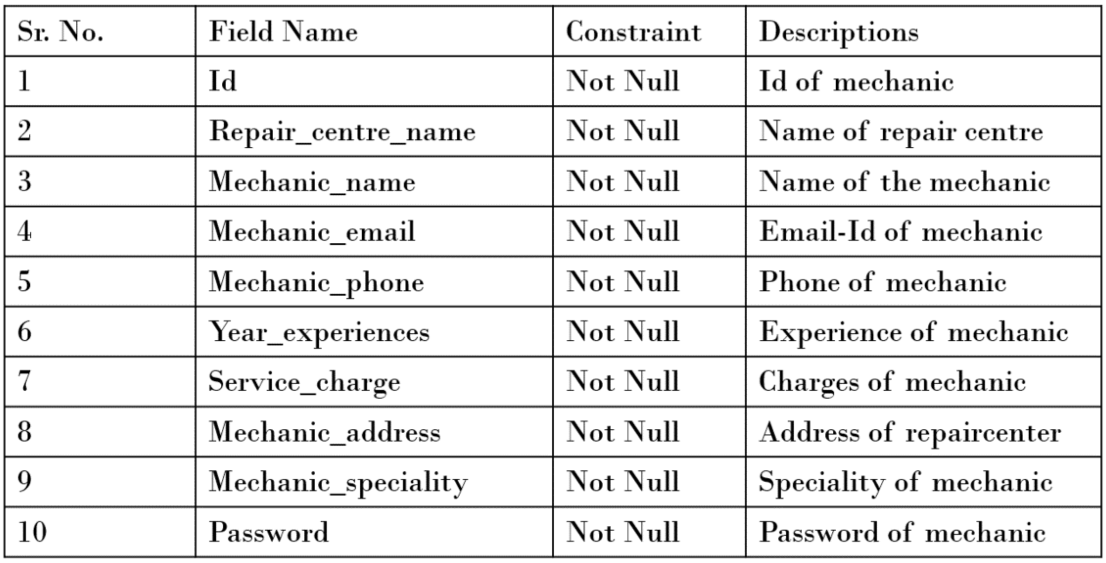
3.2 Table Structure

1. Admin table



Sr. No.	Field Name	Constraint	Descriptions
1	Id	Not Null	Key value of admin
2	Admin_email	Not Null	Email of the admin
3	Password	Not Null	Password of the admin
4	Date	Not Null	Date of added

2. Mechanic table



Sr. No.	Field Name	Constraint	Descriptions
1	Id	Not Null	Id of mechanic
2	Repair_centre_name	Not Null	Name of repair centre
3	Mechanic_name	Not Null	Name of the mechanic
4	Mechanic_email	Not Null	Email-Id of mechanic
5	Mechanic_phone	Not Null	Phone of mechanic
6	Year_experiences	Not Null	Experience of mechanic
7	Service_charge	Not Null	Charges of mechanic
8	Mechanic_address	Not Null	Address of repaircenter
9	Mechanic_speciality	Not Null	Speciality of mechanic
10	Password	Not Null	Password of mechanic

3. Vehicle table

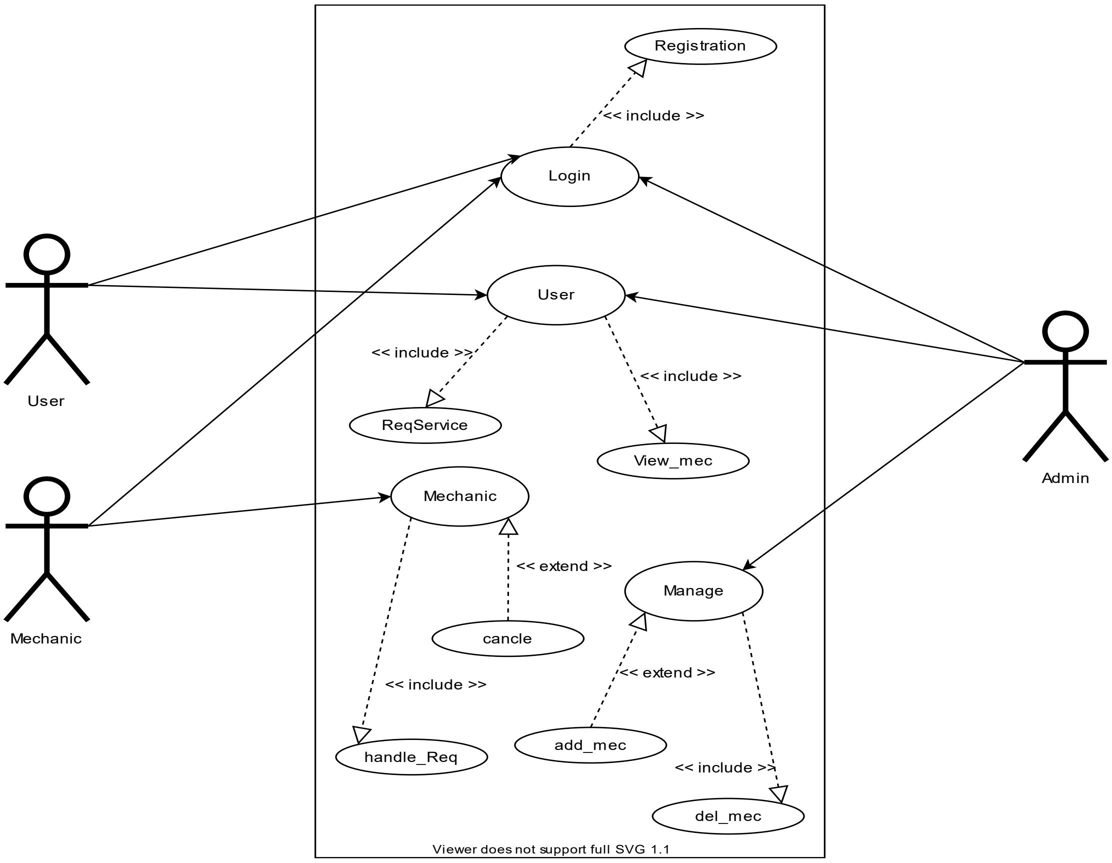
Sr. No.	Field Name	Constraint	Descriptions
1	Id	Not Null	Id of the user with vehicle
2	User_name	Not Null	Name of the user
3	User_email	Not Null	Email-Id of the user
4	User_phone	Not Null	Phone number of user
5	Vehicle_name	Not Null	Vehicle name of user
6	Vehicle_number	Not Null	Number of the vehicle
7	Vehicle_model	Not Null	Model of the vehicle
8	Vehicle_color	Not Null	Colour of the vehicle
9	Address	Not Null	Address of the user
10	Password	Not Null	Password of the user

4. Request service table

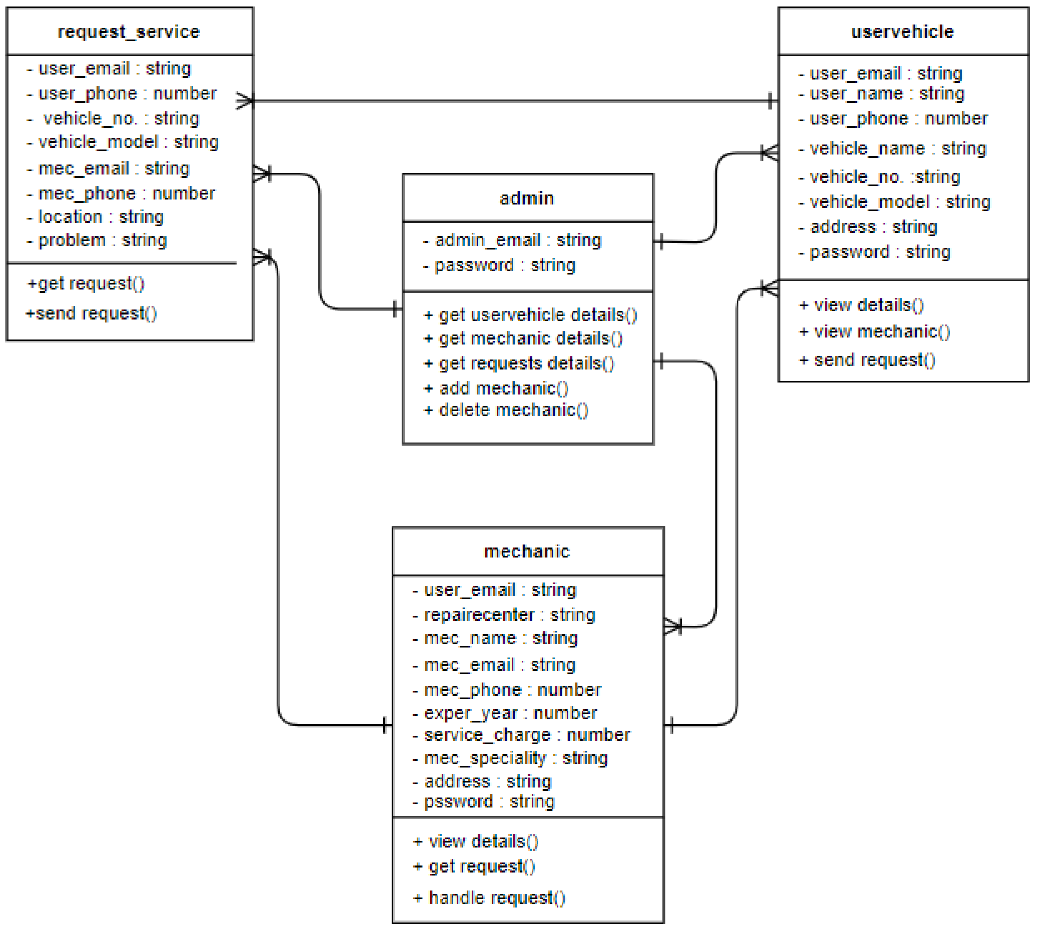
Sr. No.	Field Name	Constraint	Descriptions
1	Id	Not Null	Id of requested service
2	User_email	Not Null	Email-Id of user
3	User_phone	Not Null	Phone number of user
4	Vehicle_number	Not Null	Number of the vehicle
5	Vehicle_model	Not Null	Model of the vehicle
6	Mechanic_email	Not Null	Email-Id of mechanic
7	Mechanic_phone	Not Null	Phone number of mechanic
8	location	Not Null	Location of the user
9	Problem	Not Null	Problem of the vehicle

4. Diagram

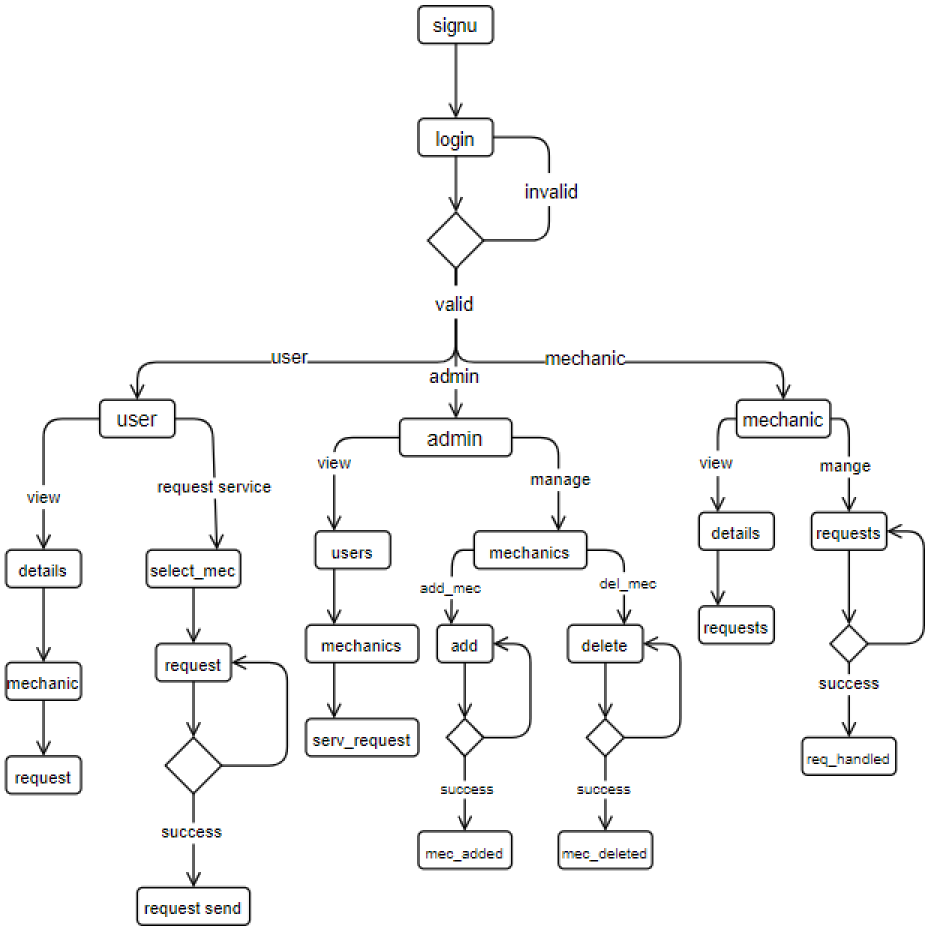
4.1 Use Case Diagrams



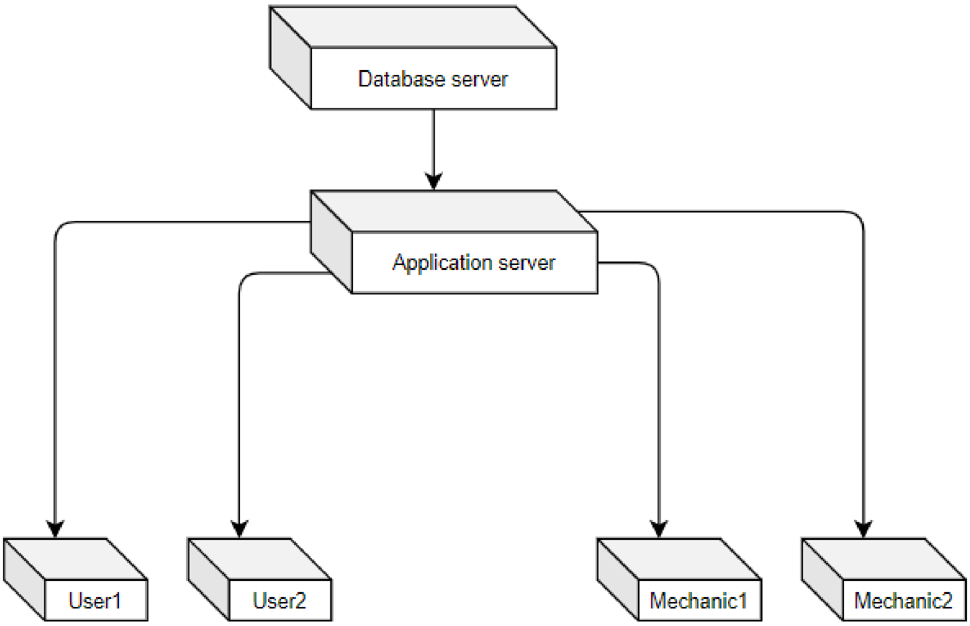
4.2 Class Diagram



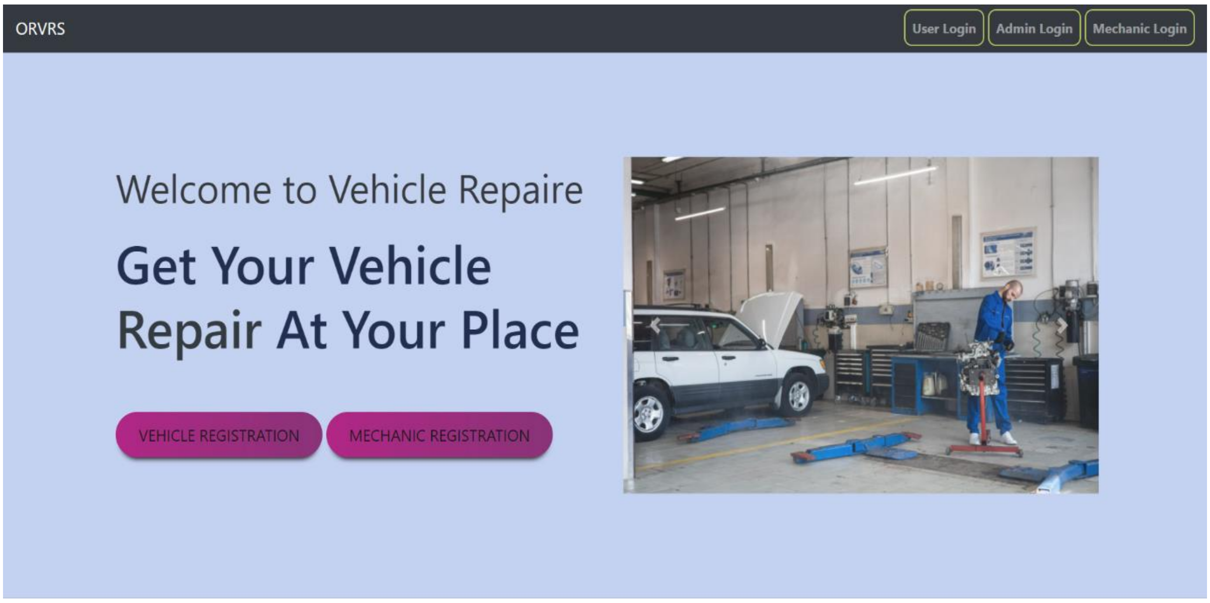
4.3 Activity Diagram



4.4 Deployment Diagram

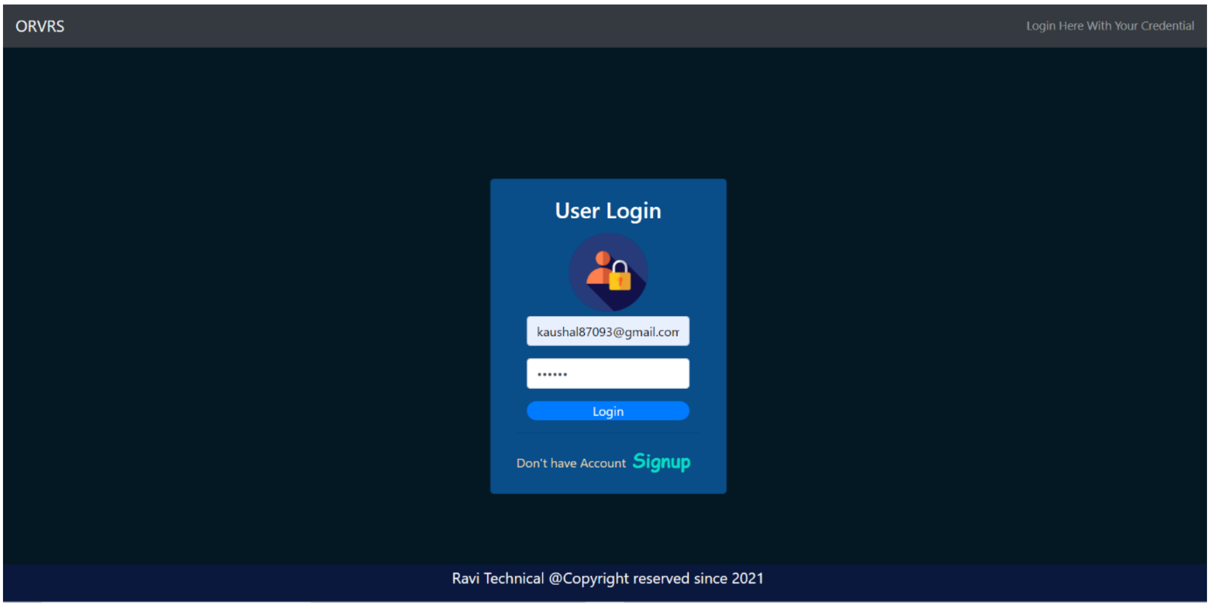


5. Sample Input and Output Screens



Welcome to Vehicle Repaire

Get Your Vehicle Repair At Your Place

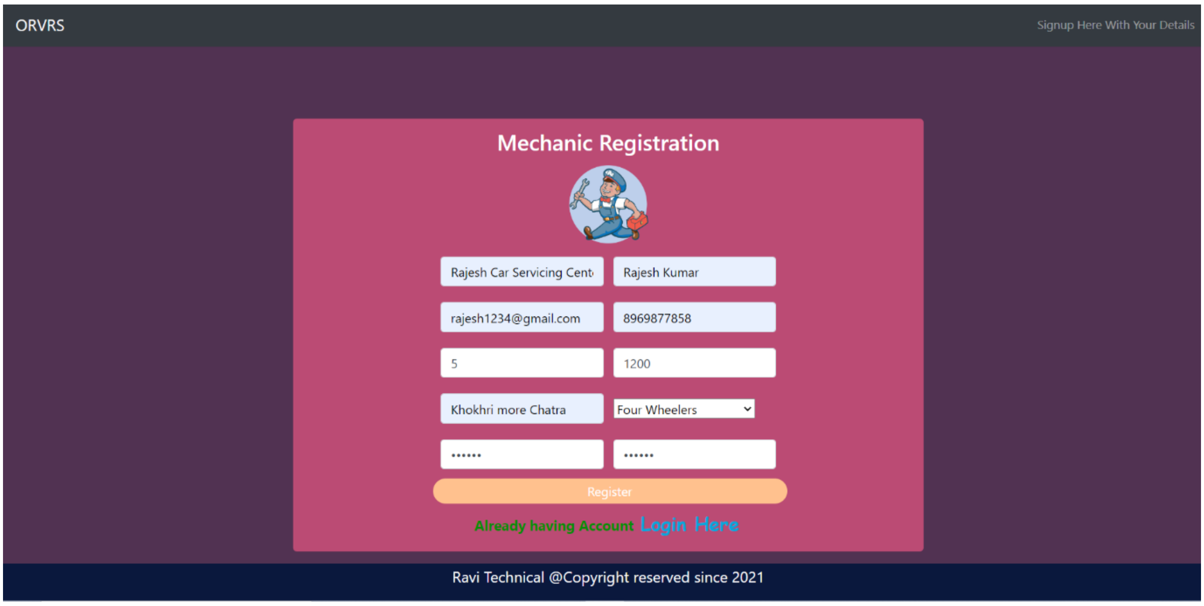


kaushal87093@gmail.com

••••••

Don't have Account Signup

Ravi Technical @Copyright reserved since 2021



ORVRS

Signup Here With Your Details

Mechanic Registration

Rajesh Car Servicing Cent

Rajesh Kumar

rajesh1234@gmail.com

8969877858

5

1200

Khokhri more Chatra

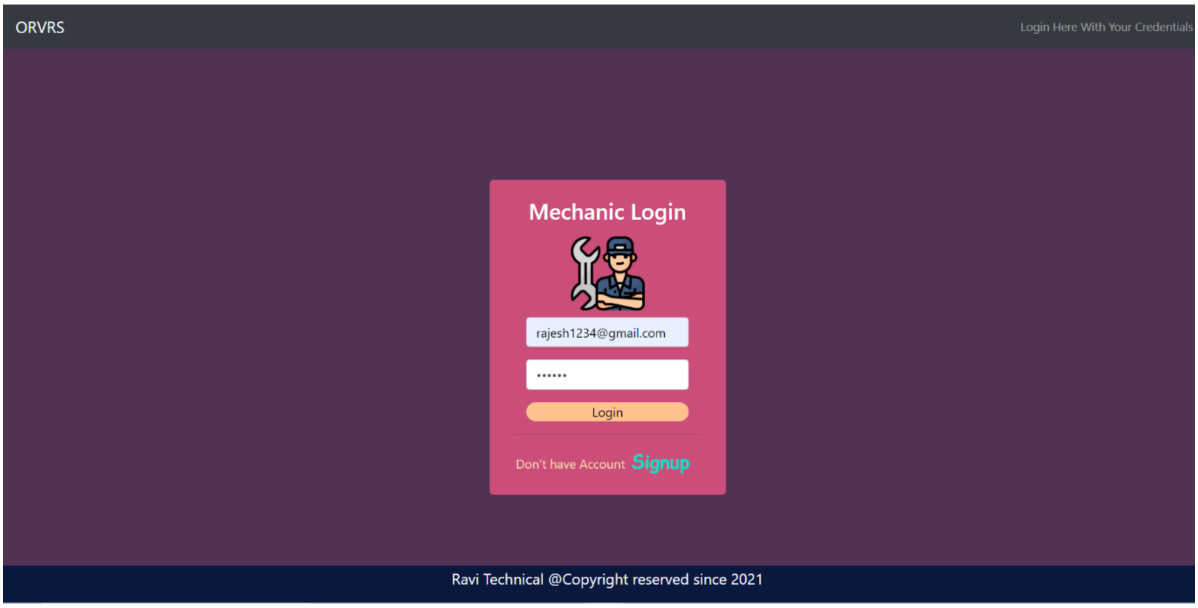
Four Wheelers

••••••

••••••

Already having Account Login Here

Ravi Technical @Copyright reserved since 2021



ORVRS

Login Here With Your Credentials

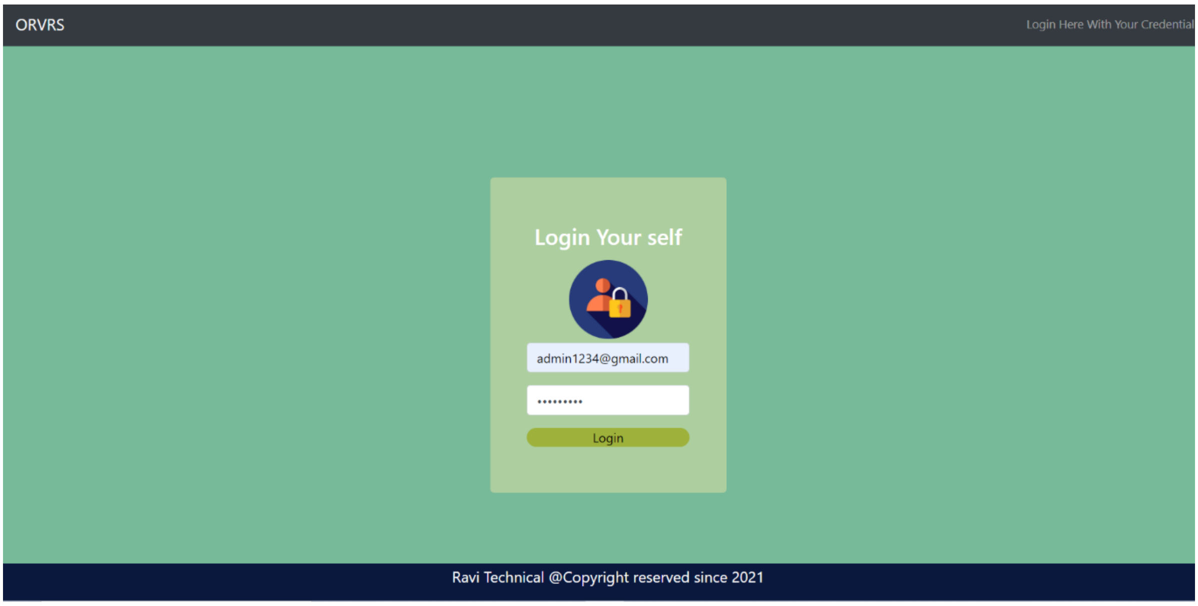
Mechanic Login

rajesh1234@gmail.com

••••••

Don't have Account Signup

Ravi Technical @Copyright reserved since 2021

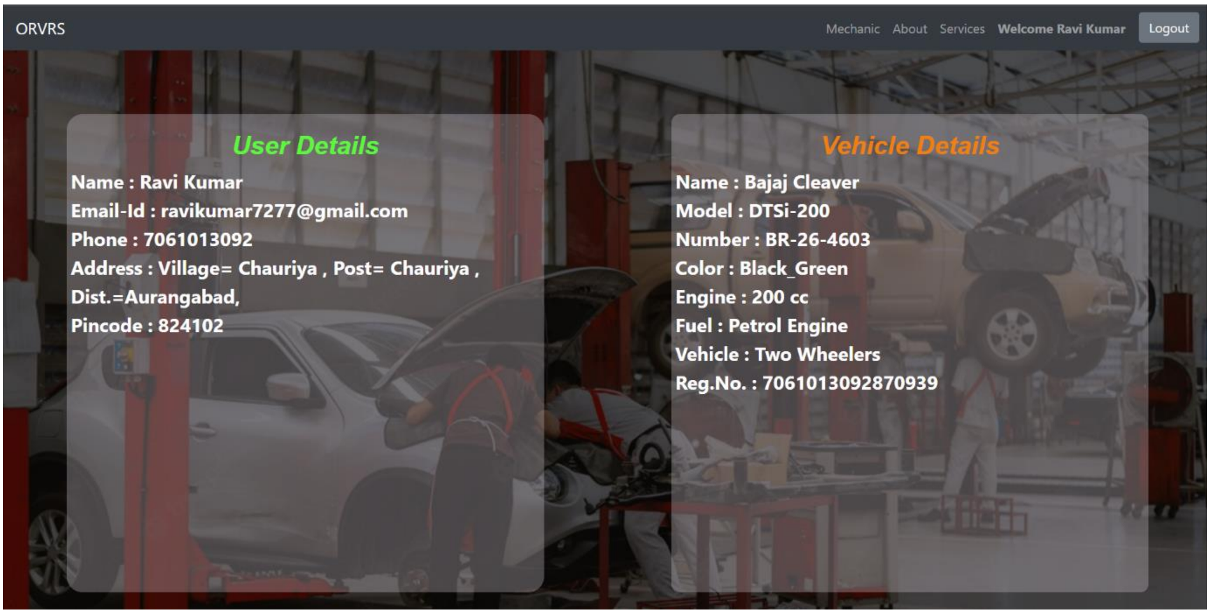


Login Your self

admin1234@gmail.com

•••••••••

Ravi Technical @Copyright reserved since 2021

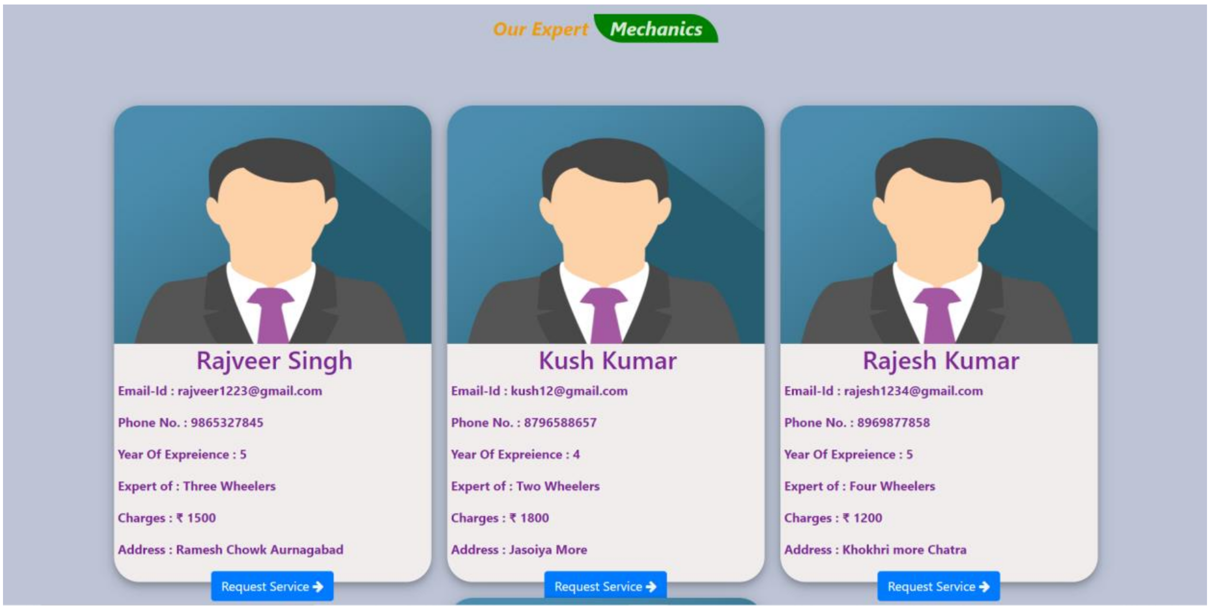


User Details

Name : Ravi Kumar
Email-Id : ravikumar7277@gmail.com
Phone : 7061013092
Address : Village= Chauriya , Post= Chauriya , Dist.=Aurangabad,
Pincode : 824102

Vehicle Details

Name : Bajaj Cleaver
Model : DTSi-200
Number : BR-26-4603
Color : Black_Green
Engine : 200 cc
Fuel : Petrol Engine
Vehicle : Two Wheelers
Reg.No. : 7061013092870939



Our Expert Mechanics

Rajveer Singh

Email-Id : rajveer1223@gmail.com

Phone No. : 9865327845

Year Of Expreience : 5

Expert of : Three Wheelers

Charges : ₹ 1500

Address : Ramesh Chowk Aurnagabad

Kush Kumar

Email-Id : kush12@gmail.com

Phone No. : 8796588657

Year Of Expreience : 4

Expert of : Two Wheelers

Charges : ₹ 1800

Address : Jasoiya More

Rajesh Kumar

Email-Id : rajesh1234@gmail.com

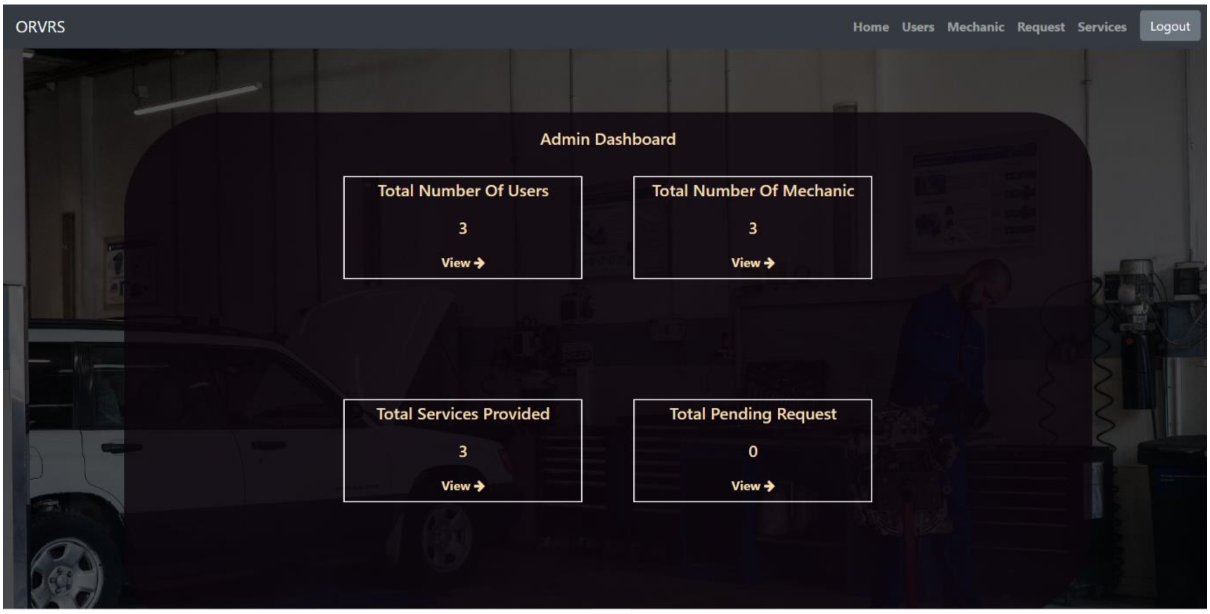
Phone No. : 8969877858

Year Of Expreience : 5

Expert of : Four Wheelers

Charges : ₹ 1200

Address : Khokhri more Chatra



Admin Dashboard

Total Number Of Users
3

Total Number Of Mechanic
3

Total Services Provided
3

Total Pending Request
0

6. Algorithms

Step 1: User has to enter all the valid details while signing up.

Step 2: After step 1, user reaches the home dashboard and from there all facilities would be make available such as viewing his own details, viewing mechanic etc.

Step 3: If users want to put a service request, they can easily do it by clicking on the button request service in his dashboard a request form will be opened and used have to fill the location and problem and submit the request ana q request will be send to the mechanic and that request will be shown to the user too.

Step 4: After completing step 3, user will be responded by the mechanic in a few moments and mechanic will reach to the destination and easily handle the issue of the users.

Step 5: After that mechanic will view the request for the service which come from the user side to his side and after it will be managed by the mechanic and he will visit the location or spot where vehicle has encounter problem and he will solve the problem.

Step 6: After all the above steps the user will be satisfied with the work done by the mechanic and he will reach to his destination point and after that user will be logout from the website.

7. Code snippets

- **Login**

```
        <!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    {{>links}}
    <title>Document</title>
</head>

<body class="boody">
    <div>
        <nav class="navbar navbar-expand-lg navbar-dark bg-dark">
            <a class="navbar-brand" href="/">ORVRS</a>
            <button class="navbar-toggler" type="button" data-toggle="collapse" data-
target="#navbarSupportedContent"
                aria-controls="navbarSupportedContent" aria-expanded="false" aria-label="Toggle
navigation">
                <span class="navbar-toggler-icon"></span>
            </button>

            <div class="collapse navbar-collapse" id="navbarSupportedContent">
                <ul class="navbar-nav ml-auto">
                    <li class="nav-item">
                        <a class="nav-link" href="" style="font-size:15px;">Login Here With Your
Credential</a>
```

```
          </li>
          {{!-- <li class="nav-item land_anchor">
              <a class="nav-link" href="/">Back</a>
            </li> --}}
        </div>
      </nav>
    </div>
    <div class="displaymessage">
      {{#if message}}
        <div style="text-align:center;" class="alert alert-secondary">
          <button type="button" class="close" data-dismiss="alert">&times;</button>
          {{message}}
        </div>
        {{/if}}
    </div>
    {{!-- Login form start here --}}
    <div class="login-content">

      <div class="login-box">
        <h3>User Login</h3>
        <img src="../../public/images/login.png" alt="login.png">

        <form action="/login" method="post">

          <div class="col-md">
            <div class="form-group">
              <input type="email" class="form-control" name="email" placeholder="Enter
email">
            </div>
            <div class="form-group">
```

```
                        <input type="password" class="form-control" name="password"
placeholder="Password">
                    </div>
                </div>
                <div class="col-md">
                    <input type="submit" value="Login" class="logbtn btn-primary">
                </div>
                <hr class="hr1">
                <p>Don't have Account <a href="/vehicleregister">Signup</a> </a></p>
            </form>
        </div>
    </div>

    {{>footer}}
</body>

</html>
```

- **Request**

```
<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Request Service Page</title>
    {{>links}}
</head>
```

```
<body>
    <nav class="navbar navbar-expand-lg navbar-dark bg-dark">
        <a class="navbar-brand" href="#">ORVRS</a>
        <button class="navbar-toggler" type="button" data-toggle="collapse" data-target="#navbarSupportedContent"
            aria-controls="navbarSupportedContent" aria-expanded="false" aria-label="Toggle navigation">
            <span class="navbar-toggler-icon"></span>
        </button>

        <div class="collapse navbar-collapse" id="navbarSupportedContent">
            <ul class="navbar-nav ml-auto">
                <li class="nav-item">
                    <a class="nav-link" href="#" style="font-weight:bold;">
                        Welcome {{data.user_name}}
                    </a>
                </li> &nbsp;&nbsp;
                <li class="nav-item">
                    <form action="/logout" method="post">
                        <button class="btn btn-secondary">Logout</button>
                    </form>
                </li>
        </div>
    </nav>



    {{!================ SERVICE FORM START FROM HERE ========================================}}
    <div class="servicediv">
        <div class="serviceformdiv">
```

```
            <div class="service-formdiv-head"><i>Request For <p>Service</i> &nbsp;<i
class="fa fa-arrow-right"
                    style="color:#f6f6f6;"></i></p>
            </div>
            <hr class="new1">
            <form action="/requestservice" method="post">


                <div class="service-formdiv-userdetails">
                    <div class="userdetail-heading">
                        <h4>UserDetails</h4>
                    </div>
                    <div class="service-form-divcolumn">
                        <div class="service-formdiv-userdetails-1">
                            <label class="inset" name="user_name"><b>Name :
</b>{{data.user_name}}</label>
                            <label class="inset" name="user_email"><b>Email-Id :
</b>{{data.user_email}}</label>
                            <label class="inset" name="user_phone"><b>Phone :
</b>{{data.user_phone}}</label>
                        </div>
                        <div class="service-formdiv-userdetails-2">
                            <label class="inset" name="vehicle_name"><b>Vehicle Name :
</b>{{data.vehicle_name}}</label>
                            <label class="inset" name="vehicle_number"><b>Vehicle Number :
                                </b>{{data.vehicle_number}}</label>
                            <label class="inset" name="vehicle_registration_no"><b>Vehicle
Reg. No. :
                                </b>{{data.vehicle_registration_no}}</label>
                            </di>
                        </div>
                    </div>
                </div>
            <hr class="new1">
            <div class="service-formdiv-mechdetails">
              <div class="mechanic-heading">
                <h4>MechanicDetails</h4>
              </div>
              <div class="service-form-mechanicdetails-1">
                <div class="service-from-mechanicdetails-1-1">
                  <label class="inset" name="mechanic_name"><b>Name :
</b>{{mecdata.mechanic_name}}</label>
                  <label class="inset" name="mechanic_email"><b>Email-Id :
                    </b>{{mecdata.mechanic_email}}</label>
                  <label class="inset" name="mechanic_phone"><b>Phone :
</b>{{mecdata.mechanic_phone}}</label>
                  <label class="inset" name="mechanic_phone"><b>Repair
CenterName : </b>{{mecdata.repairecenter_name}}</label>
                </div>
                <div class="service-from-mechanicdetails-1-2">


                  <label class="inset" name="mechanic_charge"><b>Service Charge
: ₹
                    </b>{{mecdata.mechanic_services_charge}}</label>
                  <label class="inset" name="mechanic_spec"><b>Mechanic Of :
                    </b>{{mecdata.mechanic_speciality}}</label>
                  <label class="inset" name="mechanic_exp"><b>Experience :
                    </b>{{mecdata.mechanic_year_experiences}}
                    Years</label>
                    <label class="inset" name="mechanic_exp"><b>Address :
                    </b>{{mecdata.mechanic_address}}
                    Years</label>
                </div>
```

```
          </div>


        </div>
        <hr class="new1">
        <div class="service-formdiv-problem">
          <div class="problem-heading">
            <h4>Vehicle Issue</h4>
          </div>
          <div class="form-inline padd">
            <div class="form-group padd">
              <input type="text" name="location" placeholder="Enter Your Location">
            </div>
            <div class="form-group padd">
              <input type="text" class="longinputissue" name="problem" placeholder="Enter Your Query Releted Vehicle">
              <input type="text" name="status" value="pending" hidden>
            </div>
          </div>
        </div>
        <div class="service-formdiv-button">
          <button type="submit" class="btn btn-primary">Submit Request</button>
        </div>
      </form>
    </div>
  </div>
  {{>footer}}
</body>
</html>
```

- **App.js**

```
const express = require('express');
const path = require('path');
const hbs = require('hbs');
const mongoose = require('mongoose');
const bcryptjs = require('bcryptjs');
const flash = require('connect-flash');
const conn = require('../src/db/conn')
const session = require('express-session');
const MongodbSession = require('connect-mongodb-session')(session)

require('dotenv').config();
const Register = require('./models/register')

 jwt = require('jsonwebtoken');
 if (typeof localStorage === "undefined" || localStorage === null) {
    var LocalStorage = require('node-localstorage').LocalStorage;
    localStorage = new LocalStorage('./scratch');
  }

const PORT = process.env.PORT;
const app = express();

const templetsPath = path.join(__dirname,"../templets/views");
const partialsPath = path.join(__dirname, "../templets/partials");

app.use(flash())
app.use("/public", express.static('public'));
app.use(express.urlencoded({extended:true}));
app.set('view engine', "hbs");
app.set("views",templetsPath);
hbs.registerPartials(partialsPath);

const store = new MongodbSession({
    uri:conn.MongoUri,
    collection:"mySession"
});
app.use(session({
    secret:"Key that will sign cookies",
    resave:false,
    saveUninitialized:false,
    store:store
}));

const isAuth = (req, res, next) => {
    if(req.session.isAuth){
        next();
    }else{
        res.redirect('/')
    }
};



//== LANDING PAGE SECTIN START HERE ==
app.get("/", async(req, res) => {

    var mecdata = await Register.mechanicRegister.find({}).exec();

      res.render("landing",{mdata:mecdata})
```

```
});
//== LANDING PAGE SECTIN END HERE ==



//== USER PAGE SECTION START HERE ==
//======= USER LOGIN ==========
app.get("/login", (req, res) => {
   res.render("login",{message:req.flash('message')})
});
app.post("/login", async (req, res) => {
   try{
      const email =req.body.email;
      const password =req.body.password;
     const useremail = await Register.vehicleRegister.findOne({user_email:email});
     if(!useremail){
         req.flash('message', "Invalid login details");
         return res.redirect('/login')
       }
     const isMatch = await bcryptjs.compare(password, useremail.password)
     if(!isMatch){
      req.flash('message', "Invalid login details");
       return res.redirect('/login')
     }
     var getUserId = useremail._id;
     var token = jwt.sign({userId:getUserId}, 'loginToken')
     localStorage.setItem('userToken', token)
     localStorage.setItem('loginUser', email)
      req.session.isAuth=true;
      req.flash('message',getUserId)
      res.redirect('/dashboard')
```

```
    }catch(error){
        req.flash('message', "Invalid login details");
        res.redirect('/login')
    }
});

//== VEHICLE REGISTER SECTION START HERE ==



app.get("/vehicleregister", (req, res) => {
    res.render('vehicleregister',{message:req.flash('message')})
});

// VEHICLE REGISTERR
app.post("/vehicleregister", async(req, res) => {

    const
{user_name,user_email,user_phone,vehicle_name,vehicle_model,vehicle_number,vehicle_color,vehicle_engine_cc,vehicle_fuel,vehicle_type,vehicle_registration_no,pincode,address,cpassword,password} = req.body;

    if(password != cpassword){
             req.flash('message', "Password does not matched");
                return res.redirect('/vehicleregister');
            }

    let user = await Register.vehicleRegister.findOne({user_email});
        if(user){
          req.flash('message', "User Email Already Exist");
          return res.redirect('/vehicleregister')
        }
    let vehicle_num = await Register.vehicleRegister.findOne({vehicle_number});
    if(vehicle_num){
        req.flash('message', "Vehicle Number Exist");
       return res.redirect('/vehicleregister')
    }

    let vehicleregno = await Register.vehicleRegister.findOne({vehicle_registration_no});
    if(vehicleregno){
        req.flash('message', "Vehicle Registration Number Exist");
       return res.redirect('/vehicleregister')
    }
    const hashPassword = await bcryptjs.hash(password, 10);
    vehicle = new Register.vehicleRegister({

user_name,user_email,user_phone,vehicle_name,vehicle_model,vehicle_number,vehicle_color,vehicle_engine_cc,vehicle_fuel,
        vehicle_type,vehicle_registration_no,
        pincode,address,
        password:hashPassword

    })
    await vehicle.save();
    req.flash('message', "Vehicle Registertion Successfull....");
    res.redirect('/login');
});
//== VEHICLE REGISTER SECTION END HERE =



//== USER DASHBOARD ==
app.get("/dashboard", isAuth, async(req, res) => {
    var loginUser = localStorage.getItem('loginUser');
        let userdata = await
Register.vehicleRegister.findOne({user_email:loginUser}).exec();
        let mechdata = await Register.mechanicRegister.find({}).exec();
        let reqserv = await Register.requestService.find({}).exec();


     res.render("dashboard",{data:userdata,mecdata:mechdata,reqdata:reqserv}
});
app.get("/dashboard/request/:id", async(req, res) => {
    let reqId = req.params.id;
    var mechanicdata = await Register.mechanicRegister.findOne({_id:reqId}).exec();
    var mechanicId = mechanicdata._id;
    var mecmail = mechanicdata.mechanic_email;
    var mtoken = jwt.sign({mecId:mechanicId}, 'mechanicToken');
    localStorage.setItem('mecToken', mtoken)
    localStorage.setItem('mechanicEmail',mecmail)
    var loginUser = localStorage.getItem('loginUser')
    let userdata = await
Register.vehicleRegister.findOne({user_email:loginUser}).exec();
    let mechdata = await Register.mechanicRegister.findOne({_id:reqId}).exec();
    res.render("requestservice",{data:userdata,mecdata:mechdata})
})
app.post("/requestservice", async(req, res, next) => {
    var loginUser = localStorage.getItem('loginUser');
    var mechmail = localStorage.getItem('mechanicEmail')
    var user_Data = await
Register.vehicleRegister.findOne({user_email:loginUser}).exec();
    var mechanic_Data = await
Register.mechanicRegister.findOne({mechanic_email:mechmail}).exec()
    var user_name = user_Data.user_name;
    var user_email = user_Data.user_email;
    var user_phone = user_Data.user_phone;
    var vehicle_name = user_Data.vehicle_name;
    var vehicle_number = user_Data.vehicle_number;
    var vehicle_registration_no = user_Data.vehicle_registration_no
    var mechanic_name = mechanic_Data.mechanic_name;
    var mechanic_phone = mechanic_Data.mechanic_phone;
    var mechanic_email = mechanic_Data.mechanic_email;
    var mechanic_address = mechanic_Data.mechanic_address;
    var mechanic_speciality = mechanic_Data.mechanic_speciality;
    var mechanic_year_experiences = mechanic_Data.mechanic_year_experiences;
    var mechanic_services_charge = mechanic_Data.mechanic_services_charge;
    var repairecenter_name = mechanic_Data.repairecenter_name;

    var {location,problem,status} = req.body;
    requestServ = new Register.requestService({
        user_name,user_email,user_phone,vehicle_name,
        vehicle_number,vehicle_registration_no,
        mechanic_name,mechanic_phone,mechanic_email,
        mechanic_address,mechanic_speciality,
        mechanic_year_experiences,mechanic_year_experiences,
        mechanic_services_charge,repairecenter_name,
        location,problem,status
    })
    await requestServ.save();
    res.redirect('/dashboard');
})
//== USER/CLIENT SECTION END HERE ==
//== ADMIN SECTION START HERE ==
app.get("/adminlogin", (req, res) => {
    res.render('adminlogin',{message:req.flash('message')})
})
// ADMIN LOGIN
app.post("/adminlogin", async(req, res) => {
    try{
        const email =req.body.email;
        const password =req.body.password;
        const admindata = await Register.adminLogin.findOne({}).exec();
        const adminemail = admindata.admin_email;
        const adminpassword = admindata.password;
       if(email!=adminemail){
        req.flash('message', 'Invalid login details')
            return res.redirect('/adminlogin')
          };
       if(password!=adminpassword){
        req.flash('message', 'Invalid login details')
         return res.redirect('/adminlogin')
       };
        req.session.isAuth=true;
         return res.redirect('/admindashboard');
    }catch(error){
        req.flash('message', 'Invalid login details');
        res.redirect('/adminlogin');
    }
})
// ADMIN DASHBORAD
app.get("/admindashboard", isAuth, async(req,  res) => {
    const totaluser = await Register.vehicleRegister.count();
    const totalmechanic = await Register.mechanicRegister.count();
    const totalreqpend = await Register.requestService.count({status:"pending"});
    const totalreqcomplet = await Register.requestService.count({status:"completed"})

res.render('admindashboard',{toutalUser:totaluser,totalmechanic:totalmechanic,totalre
qpend:totalreqpend,totalreqcomp:totalreqcomplet})
});
app.get("/totalusers", async(req, res) => {
    const userDetails = await Register.vehicleRegister.find({});
    res.render("totalusers",{userdata:userDetails})
})
app.get("/totalmechanics", async(req, res) => {
    const mechanicDetails = await Register.mechanicRegister.find({});
    res.render("totalmechanics",{mecdata:mechanicDetails})
}
//addmechanics by admin
app.get("/addmechanics",(req, res) =>{
    res.render('addmechanics',{message:req.flash('message')})
}
app.post("/addmechanics", async(req, res) =>{
    const {repairecenter_name, mechanic_name, mechanic_email, mechanic_phone,
mechanic_year_experiences, mechanic_services_charge, mechanic_address,
mechanic_speciality, password,cpassword}  = req.body
    console.log(repairecenter_name, mechanic_name, mechanic_email,
mechanic_phone, mechanic_year_experiences, mechanic_services_charge,
mechanic_address, mechanic_speciality, password,cpassword);
    //res.render("addmechanics"
    if(password != cpassword){
        req.flash('message',"Password does not match..!")
        return res.redirect("/addmechanics")
    }
    const mechanicEmail = await
Register.mechanicRegister.findOne({mechanic_email})
    if(mechanicEmail){
        req.flash('message', "Mechanic Emai Already Exist...!")
        return res.redirect("/addmechanics")
    }
    const mecphone = await Register.mechanicRegister.findOne({mechanic_phone})
    if(mecphone){
        req.flash('message',"Mechanic Phone Already Exist...!")
        return res.redirect("/addmechanics")
    }
    const hashPassword = await bcryptjs.hash(password, 10);
    mechanic = new Register.mechanicRegister({
        repairecenter_name, mechanic_name,mechanic_email,
        mechanic_phone,mechanic_year_experiences,
        mechanic_services_charge,mechanic_address,
        mechanic_speciality,
        password:hashPassword,
    })
    await mechanic.save();
    // req.flash('message',"Mechanic Registration Successfull...! Now Login with
credentials")
    res.redirect("/totalmechanics")
})
app.get("/requests", async(req, res) => {
    const serReqPend = await Register.requestService.find({status:"pending"});
    res.render("requests",{reqpend:serReqPend})
})
app.get("/servicesprovided", async(req, res) => {
    const serReqCopm = await Register.requestService.find({status:"completed"});
    res.render("servicesprovided",{reqcomp:serReqCopm})
})
app.get("/admindashboard/delete/:id",isAuth ,async(req, res, next) => {
    const mechId = req.params.id;
    const mechancidelete = await
Register.mechanicRegister.findByIdAndDelete({_id:mechId});
    if(mechancidelete){
        res.redirect('/totalmechanics')
    }
})
// == ADMIN SECTION END HERE ==
// == REPAIRE CENTER/ MECHANIC SECTION START HERE ==
app.get("/mechaniclogin", (req, res) => {
    res.render('mechaniclogin',{message:req.flash('message')})
};
//MECHANIC LOGIN
app.post("/mechaniclogin", async(req, res) =>
    try{
    const {email, password} =req.body;
    const mechanicEmail = await
Register.mechanicRegister.findOne({mechanic_email:email}
    if(!mechanicEmail){
        req.flash('message',"Invalid mechanic Details")
        return res.redirect("/mechaniclogin")

    const isMatch = await bcryptjs.compare(password, mechanicEmail.password)
    if(!isMatch){
        req.flash('message',"Invalid ismathc Details")
        return res.redirect("/mechaniclogin")

    const getmecId = mechanicEmail._id;
    const mechToken = jwt.sign({'mechanicToken':getmecId}, 'mechanicLogin');
    localStorage.setItem('mechToken',mechToken);
    localStorage.setItem('mechanic_login',email)
    req.session.isAuth=true;
    // req.flash('message',getmecId)
    res.redirect('/mechanicdashboard')
    }catch(err){
        req.flash('message',"Invalid catch Details")
        res.redirect("/mechaniclogin
})
app.get("/mechanicregister", (req, res) => {
    res.render("mechanicregister",{message:req.flash("message")})
});

//MECHANIC REGISTERATION
app.post("/mechanicregister", async(req, res) =>
    const {repairecenter_name, mechanic_name, mechanic_email, mechanic_phone,
mechanic_year_experiences, mechanic_services_charge, mechanic_address,
mechanic_speciality, password,cpassword} = req.body;
    if(password != cpassword){
        req.flash('message',"Password does not match..!")
        return res.redirect("/mechanicregister")
    }
    const mechanicEmail = await
Register.mechanicRegister.findOne({mechanic_email})
    if(mechanicEmail){
        req.flash('message', "Mechanic Emai Already Exist...!")
        return res.redirect("mechanicregister")
    }
    const mecphone = await Register.mechanicRegister.findOne({mechanic_phone})
    if(mecphone){
        req.flash('message',"Mechanic Phone Already Exist...!")
        return res.redirect("mechanicregister")
    }
    const hashPassword = await bcryptjs.hash(password, 10);
    mechanic = new Register.mechanicRegister({
        repairecenter_name, mechanic_name,mechanic_email,
        mechanic_phone,mechanic_year_experiences,
        mechanic_services_charge,mechanic_address,
        mechanic_speciality,
        password:hashPassword,
    })
    await mechanic.save();
    req.flash('message',"Mechanic Registration Successfull...! Now Login with
credentials")
    res.redirect("/mechaniclogin")
}
//dashboard
app.get("/mechanicdashboard", isAuth, async(req, res) => {
    const mecId = localStorage.getItem("mechanic_login")
    const mechanicData = await Register.mechanicRegister.
findOne({mechanic_email: mecId}).exec();
    const mecEmail = mechanicData.mechanic_email;
   // const mecReqServ = await Register.requestService.
find({mechanic_email:mecEmail}
    const mecReqPending = await
Register.requestService.find({mechanic_email:mecEmail,status:"pending"})
    const mecReqComplet = await
Register.requestService.find({mechanic_email:mecEmail,status:"completed"})
   res.render('mechanicdashboard',{data:mechanicData,
    // reqdata:mecReqServ
    reqpend:mecReqPending,reqcomp:mecReqComplet});
})
```

```
app.get("/mechanicdashboard/updaterequest/:id",isAuth ,async(req, res, next) => {
        const reqId = req.params.id;
        const pendReq = await Register.requestService.findOne({_id:reqId});
        const usereMail = pendReq.user_email;
        const old_status = {user_email:usereMail};
        const new_status = {$set:{status:"completed"}}
        const result = await Register.requestService.updateOne(old_status,new_status);
        res.redirect("/mechanicdashboard")
})
// ==  MECHANIC SECTION END HERE ==
//==404 ERROR PAGE START HERE ==

app.get("/*", (req, res) => {
    res.render("404error",{errorMsg:"Oooppsss... Page Not Found"})
});

//== 404 ERROR PAGE END HERE ==

//== Logout Section =====
app.post("/logout", (req, res) => {
    req.session.destroy((err) => {
        if(err) throw err;
        res.redirect('/')
    })
});



app.listen(PORT, () => {
    console.log(`This application is running on the port no ${PORT}`)
}
```

- **Landing Page**

```
<!DOCTYPE html>
<html lang="en">

<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>{{title}}</title>
  {{>links}}
</head>

<body>
  <nav class="navbar navbar-expand-lg navbar-dark bg-dark">
    <a class="navbar-brand" href="#">ORVRS</a>
    <button class="navbar-toggler" type="button" data-toggle="collapse" data-target="#navbarSupportedContent"
      aria-controls="navbarSupportedContent" aria-expanded="false" aria-label="Toggle navigation">
      <span class="navbar-toggler-icon"></span>
    </button>

    <div class="collapse navbar-collapse" id="navbarSupportedContent">
      <ul class="navbar-nav ml-auto">
        <li class="nav-item land_anchor">
          <a class="nav-link " href="/login">User Login</a>
        </li>
        <li class="nav-item land_anchor">
          <a class="nav-link" href="/adminlogin">Admin Login</a>
        </li>
        <li class="nav-item land_anchor">
          <a class="nav-link" href="/mechaniclogin">Mechanic Login</a>
        </li>
    </div>
  </nav>
  <div class="opacity">
    <div class="container-fluid main_header space">
      <div class="row">
        <div class="col-md-10 col-12 mx-auto">
          <div class="row">
            <!-- left side div -->
            <div class="col-md-6 col-12 main_header_left">
              <p>Welcome to Vehicle Repaire</p>
              <h1>Get Your Vehicle <span class="txt_clr">Repair</span> at Your
Place</h1>
              <a href="/vehicleregister"><button>Vehicle Registration</button></a>
              <a href="/mechanicregister"><button>Mechanic Registration</button></a>
            </div>
            <! -- right side div -->
            <div class="col-md-6 col-12 main_haeader_right">


              <div id="carouselExampleControls" class="carousel slide cornershort" data-
ride="carousel">
                <div class="carousel-inner">
                  <div class="carousel-item active">
                    <img class=" w-100" src="../../public/images/bg1.jpg" alt="First slide">
                  </div>
```

```
         <div class="carousel-item">
          <img class=" w-100" src="../../public/images/bg2.jpg" alt="Second slide">
         </div>
         <div class="carousel-item">
          <img class=" w-100" src="../../public/images/bg3.jpg" alt="Third slide">
         </div>
         <div class="carousel-item">
          <img class=" w-100" src="../../public/images/bg4.jpg" alt="Third slide">
         </div>
         <div class="carousel-item">
          <img class=" w-100" src="../../public/images/bg5.jpg" alt="Third slide">
         </div>
        </div>
        <a class="carousel-control-prev" href="#carouselExampleControls" role="button" data-slide="prev">
         <span class="carousel-control-prev-icon" aria-hidden="true"></span>
         <span class="sr-only">Previous</span>
        </a>
        <a class="carousel-control-next" href="#carouselExampleControls" role="button" data-slide="next">
         <span class="carousel-control-next-icon" aria-hidden="true"></span>
         <span class="sr-only">Next</span>
        </a>
       </div>

      </div>
     </div>

    </div>
   </div>
```

```
  </div>
 </div>
 {{!-- OUR MECHANIC --}}
{{!---------------------- MECHANIC DETAILS IN DASHBOARD ----------------------
---------}}
 <div class="mechdashcontent">
  <div class="mechhead"><i>Our Expert</i> &nbsp;<p><i>Mechanics</i></p>
  </div>
  <div class="mechanicdash">
   {{#each mdata}}
   <div class="mechdashinside">
    <div class="mechanicdash-img">
     <img src="../../public//images/img_avatar.png" alt="">
    </div>
    <div class="mehcnaicdash-details">
     <h2 class="text-center colour">{{mechanic_name}}</h2>
     <p class="colour">Email-Id : {{mechanic_email}}</p>
     <p class="colour">Phone No. : {{mechanic_phone}}</p>
     <p class="colour">Year Of Expreience : {{mechanic_year_experiences}}</p>
     <p class="colour">Expert of : {{mechanic_speciality}}</p>
     <p class="colour">Charges : ₹ {{mechanic_services_charge}}</p>
     <p class="colour">Address : {{mechanic_address}}</p>
    </div>
   </div>
   {{/each}}
  </div>
 </div>
 {{!------------------------------ MECHANIC DETAILS IN DASHBOARD -------------
-------------}}
 <div class="servicecontent">
```

```
<div class="heading">
  <div class="ourserve"><i>Our Most</i></div>
  <div class="ourservice"><i> Popular Services</i></div>
</div>
<div class="service">
  <div class="servicebox1">
    <img src="../../public//images/s1.png" alt="Tyre Service">
    <p class="headpink">Battery Service</p>
  </div>
  <div class="servicebox2">
    <img src="../../public/images/s2.png" alt="Tyre servie">
    <p class="headpink">Tyre Service</p>
  </div>
  <div class="servicebox3">
    <img src="../../public/images/s3.png" alt="Car altenator">
    <p class="headpink">Car Alternator</p>
  </div>
</div>
<div class="service">
  <div class="servicebox4">
    <img src="../../public/images/s4.png" alt="Car Joints">
    <p class="headpink">Car Ball & Joints</p>
  </div>
  <div class="servicebox5">
    <img src="../../public//images/s5.png" alt="Lubricant service">
    <p class="headpink">Lubricant Service</p>
  </div>
  <div class="servicebox6">
    <img src="../../public//images/s6.png" alt="Normal service">
    <p class="headpink">Normal Service</p>
```

```
              </div>
            </div>
          </div>
          {{!-- OUR SERIVECS END HERE --}}
          </div>
          {{>footer}}
        </body
        </html>
```

- **Dashboard**

```
          <!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>{{title}}</title>
  {{>links}}
</head>
<body>
  <nav class="navbar navbar-expand-lg navbar-dark bg-dark">
    <a class="navbar-brand" href="#">ORVRS</a>
    <button class="navbar-toggler" type="button" data-toggle="collapse" data-
target="#navbarSupportedContent"
      aria-controls="navbarSupportedContent" aria-expanded="false" aria-label="Toggle
navigation">
      <span class="navbar-toggler-icon"></span>
    </button>
    <div class="collapse navbar-collapse" id="navbarSupportedContent">
```

```
      <ul class="navbar-nav ml-auto">
        <li class="nav-item land_anchor">
          <a class="nav-link " href="/login">User Login</a>
        </li>
        <li class="nav-item land_anchor">
          <a class="nav-link" href="/adminlogin">Admin Login</a>
        </li>
        <li class="nav-item land_anchor">
          <a class="nav-link" href="/mechaniclogin">Mechanic Login</a>
        </li>
    </div>
  </nav>
  <div class="opacity">
    <div class="container-fluid main_header space">
      <div class="row">
        <div class="col-md-10 col-12 mx-auto">
          <div class="row">
            <!-- left side div -->
            <div class="col-md-6 col-12 main_header_left">
              <p>Welcome to Vehicle Repaire</p>
              <h1>Get Your Vehicle <span class="txt_clr">Repair</span> at Your Place</h1>
              <a href="/vehicleregister"><button>Vehicle Registration</button></a>
              <a href="/mechanicregister"><button>Mechanic Registration</button></a>
            </div>
            <!-- right side div -->
            <div class="col-md-6 col-12 main_haeader_right">
              <div id="carouselExampleControls" class="carousel slide cornershort" data-ride="carousel">
                <div class="carousel-inner">
                  <div class="carousel-item active">
              <img class=" w-100" src="../../public/images/bg1.jpg" alt="First slide">
            </div>
            <div class="carousel-item">
              <img class=" w-100" src="../../public/images/bg2.jpg" alt="Second slide">
            </div>
            <div class="carousel-item">
              <img class=" w-100" src="../../public/images/bg3.jpg" alt="Third slide">
            </div>
            <div class="carousel-item">
              <img class=" w-100" src="../../public/images/bg4.jpg" alt="Third slide">
            </div>
            <div class="carousel-item">
              <img class=" w-100" src="../../public/images/bg5.jpg" alt="Third slide">
            </div>
          </div>
          <a class="carousel-control-prev" href="#carouselExampleControls" role="button"
data-slide="prev">
            <span class="carousel-control-prev-icon" aria-hidden="true"></span>
            <span class="sr-only">Previous</span>
          </a>
          <a class="carousel-control-next" href="#carouselExampleControls" role="button"
data-slide="next">
            <span class="carousel-control-next-icon" aria-hidden="true"></span>
            <span class="sr-only">Next</span>
          </a>
        </div>

      </div>
    </div>
```

```
        </div>
      </div>
    </div>
  </div>
  {{!-- OUR MECHANIC --}}
 {{!---------------------- MECHANIC DETAILS IN DASHBOARD ------------------------------
}}
  <div class="mechdashcontent">
    <div class="mechhead"><i>Our Expert</i> &nbsp;<p><i>Mechanics</i></p>
    </div>
    <div class="mechanicdash">
      {{#each mdata}}
      <div class="mechdashinside">
        <div class="mechanicdash-img">
          <img src="../../public//images/img_avatar.png" alt="">
        </div>
        <div class="mehcnaicdash-details">
          <h2 class="text-center colour">{{mechanic_name}}</h2>
          <p class="colour">Email-Id : {{mechanic_email}}</p>
          <p class="colour">Phone No. : {{mechanic_phone}}</p>
          <p class="colour">Year Of Expreience : {{mechanic_year_experiences}}</p>
          <p class="colour">Expert of : {{mechanic_speciality}}</p>
          <p class="colour">Charges : ₹ {{mechanic_services_charge}}</p>
          <p class="colour">Address : {{mechanic_address}}</p>
        </div>
      </div>
      {{/each}}
    </div>
  </div>
  {{!------------------------------- MECHANIC DETAILS IN DASHBOARD -----------------------
----}}
  <div class="servicecontent">
    <div class="heading">
      <div class="ourserve"><i>Our Most</i></div>
      <div class="ourservice"><i> Popular Services</i></div>
    </div>
    <div class="service">
      <div class="servicebox1">
        <img src="../../public//images/s1.png" alt="Tyre Service">
        <p class="headpink">Battery Service</p>
      </div>
      <div class="servicebox2">
        <img src="../../public/images/s2.png" alt="Tyre servie">
        <p class="headpink">Tyre Service</p>
      </div>
      <div class="servicebox3">
        <img src="../../public/images/s3.png" alt="Car altenator">
        <p class="headpink">Car Alternator</p>
      </div>
    </div>
    <div class="service">
      <div class="servicebox4">
        <img src="../../public/images/s4.png" alt="Car Joints">
        <p class="headpink">Car Ball & Joints</p>
      </div>
      <div class="servicebox5">
        <img src="../../public//images/s5.png" alt="Lubricant service">
        <p class="headpink">Lubricant Service</p>
      </div>
    <div class="servicebox6">
      <img src="../../public//images/s6.png" alt="Normal service">
      <p class="headpink">Normal Service</p>
    </div>
  </div>
</div>
{{!-- OUR SERIVECS END HERE --}}
{{!-- WE HAVE THE PRODUCT START HERE --}}
<div class="productcontent">
  <div class="heading">
    <div class="ourpro"><i>We Have</i></div>
    <div class="ourproduct"><i>Our Products</i></div>
  </div>
  <div class="product">
    <div class="productbox1">
      <img src="../../public/images/p1.png" alt="Tyres">
      <p class="headgreen">Oil & Tyres</p>
    </div>
    <div class="productbox2">
      <img src="../../public/images/p2.png" alt="Tyres">
      <p class="headgreen">Car AirFilter</p>
    </div>
    <div class="productbox3">
      <img src="../../public/images/p3.png" alt="Tyres">
      <p class="headgreen">Car Alternator</p>
    </div>
  </div>
  <div class="product">
    <div class="productbox4">
      <img src="../../public/images/p4.png" alt="Tyres">
      <p class="headgreen">Car Battery</p>
    </div>
    <div class="productbox5">
      <img src="../../public/images/p5.png" alt="Tyres">
      <p class="headgreen">Car Boll Joints</p>
    </div>
    <div class="productbox6">
      <img src="../../public/images/p6.png" alt="Tyres">
      <p class="headgreen">Car Headlights</p>
    </div>
  </div>
</div>
</div>
{{!-- WE HAVE THE PRODUCT END HERE --}}

{{!-- SERVICES PACKAGES START HERE --}}
<div class="servicepackagecontent">
  <div class="heading">
    <div class="ourpack"><i>Maintanenence</i></div>
    <div class="ourpackages"><i>Service Packages</i></div>
  </div>
  <div class="packages">
    <div class="packagebox1">
      <p class="freeserv">FREE SERVICES</p>
      <div class="packagebox1-1">
        <div class="pack1-1-1">
        <table>
          <tr>
            <td><b>Wiper Blade</b> Installation</td>
          </tr>
          <tr>
            <td><b>Check Engine</b> Lignt Code Retrieval</td>
          </tr>
          <tr>
            <td><b>Air Filter</b> Replacement</td>
          </tr>
          <tr>
            <td><b>Starting and Charging</b> System Evolution</td>
          </tr>
        </table>
        </div>
        <div class="pack1-1-2">
          <table>
            <tr>
              <td>Free</td>
            </tr>
            <tr>
              <td>Free</td>
            </tr>
            <tr>
              <td>Free</td>
            </tr>
            <tr>
              <td>Free</td>
            </tr>
          </table>
        </div>
      </div>
    </div>
    <div class="packagebox2">
```

```
<p class="packheadgreen">TUNE-UP</p>
<div class="packagebox2-1">
  <div class="pack2-1-1">
  <table>
    <tr>
      <td><b>Spring Maintanenence</b> Package</td>
    </tr>
    <tr>
      <td><b>Summer Maintanenence</b> Package</td>
    </tr>
    <tr>
      <td><b>Winter Maintanenence</b> Package</td>
    </tr>
    <tr>
      <td><b>Smoke Test Diagnostic</b> Service</td>
    </tr>
  </table>
  </div>
  <div class="pack2-1-2">
    <table>
      <tr>
        <td>From ₹ 1859</td>
      </tr>
      <tr>
        <td>From ₹ 1599</td>
      </tr>
      <tr>
        <td>From ₹ 1769</td>
      </tr>
      <tr>
```

```
            <td>From ₹ 650</td>
          </tr>
        </table>
      </div>
    </div>
  </div>
</div>
<div class="packages">
  <div class="packagebox3">
    <p class="packheadgreen">MAINTENANCE SERVICES</p>
    <div class="packagebox3-1">
      <div class="pack3-1-1">
      <table>
        <tr>
          <td><b>Head Light </b>Package</td>
        </tr>
        <tr>
          <td><b>Small Bulb Installation </b>Package</td>
        </tr>
        <tr>
          <td><b>Cabin Air Filter Install </b>Package</td>
        </tr>
        <tr>
          <td><b>Spring Maintanenence </b>Package</td>
        </tr>
      </table>
      </div>
      <div class="pack3-1-2">
        <table>
          <tr>
```

```
     <td>From ₹ 2599</td>
    </tr>
    <tr>
     <td>From ₹ 780</td>
    </tr>
    <tr>
     <td>From ₹ 899</td>
    </tr>
    <tr>
     <td>From ₹ 8999</td>
    </tr>
   </table>
  </div>
 </div>
</div>
<div class="packagebox4">
 <p class="packheadgreen">FLUID SERVICES</p>
 <div class="packagebox4-1">
  <div class="pack4-1-1">
  <table>
   <tr>
    <td><b>Engine Flush</b> Service</td>
   </tr>
   <tr>
    <td><b>Transmission Fluid</b> Service</td>
   </tr>
   <tr>
    <td><b>Break Fluid</b> Exchange</td>
   </tr>
   <tr>
```

```
          <td><b>Coolant Drain & Fill </b>Package</td>
        </tr>
      </table>
      </div>
      <div class="pack4-1-2">
        <table>
          <tr>
            <td>From ₹ 1550</td>
          </tr>
          <tr>
            <td>From ₹ 1199</td>
          </tr>
          <tr>
            <td>From ₹ 425</td>
          </tr>
          <tr>
            <td>From ₹ 750</td>
          </tr>
        </table>
      </div>
    </div>
  </div>
</div>
</div>
{{!-- SERVICES PACKAGES END HERE --}}

{{>footer}}
</body>
</html>
```

8. Testing

8.1 Unit Test Plan

- A unit is the smallest testable part of software.
- It usually has one or a few inputs and usually a single input
- In procedural programming a unit may be an individual program, function, procedure, etc.
- In object-oriented programming, the smallest unit is a method which to a base/super class, abstract class or derived/child class.
- Some treat a module of an application as a unit.
- This is to be discouraged as there will probably be many individual units within that module.
- Unit testing is the first level of testing and is performed prior to Integration Testing

8.2 Acceptance Test Plan

- Functional Testing is a type of software testing where, the system is tested against the functional requirements/specifications.
- Functions are tested by feeding them input & examining the output.
- Functional testing ensures that the requirements are properly satisfied by the website Application. This type of testing is not concerned with how processing occurs, but rather, with the results of processing.
- During functional testing, Black Box Testing technique is used in which the internal logic of the system being tested is not known to the tester.
- User Registration: Process has been properly working and all fields are validated through various validation checks.
- Login & password validation process has been co-operated properly.
- Payment gateway has been done properly & accurately.
- Contents are playing safely.
- Admin side handling & dynamic & real-time data integration module is implemented successfully.

8.3 Test Case

Name	**User login on user side**
Description	This test case will check whether email & password is valid or not. It will then allow the user to login.
Input Data	✓ System allows login ✓ Define blank Email & Password ✓ Valid Email & Blank Password ✓ Valid Email but Wrong Password ✓ Wrong Email but Valid Password ✓ Wrong Email & Password
Output	✓ System allows login ✗ Email required ✗ Password required ✗ System prompts error that user is not valid ✗ System prompt error that enter email & password
Actual Output	System allows login

9. Limitation of the Proposed System

- Does not fetch the user live location in the current proposed On Road Vehicle Repairing System.
- Does not fetch the live location of the mechanic in the current proposed system of the On Road Vehicle Repairing System.
- User has to enter the location manually of his vehicle where it has issued some problem.
- Online payment getaway is not available for the payment to the mechanic.
- Not any report is being generated for the any task.

10. Proposed Enhancement

- There will be option available to choose the live location of the vehicle for the exact location of the vehicle.
- Better user interface will be available to the user to access the things more quickly.
- The live location for the mechanic will be available so that user can easily track the location of the mechanic, that where he has reached.
- There will be Online payment getaway available for the payment of the mechanic in the On Road Vehicle Repairing System.
- The Report will be generated for the User and Mechanic for their respective task completed.

11. Conclusion

Today's world has made vehicles a necessity for getting about and reaching destinations. It is one of the greatest methods for long-distance travel with your loved ones and friends. To ensure that passengers have the best possible experience when travelling, vehicles are always being upgraded and improved. However, as a car is a machine, problems might arise. A mechanic should be called in to fix them. If someone is travelling a long distance and experiences a problem with their car but is too far away to make it to a mechanic shop or a repair facility, this presents a serious dilemma for them.

We have one of the top websites for vehicle repair systems, ORVRS. Users may simply solve any issues they run into while travelling to their destinations or to other locations by utilizing our website. They may easily request the services, and the technician will address the request in a short amount of time. The user or individual can then travel his destination without being bothered.

12. Bibliography

- https://www.stackoverflow.com
- https://youtube.com
- https://google.co.in
- https://github.com
- https://www.visual-paradigm.com
- https://lottiefiles.com

Books References

• Mastering Node.js	Sandro Pasquali
• Node.js in action	Mike Cantelon
• Node Web Development	David Herron