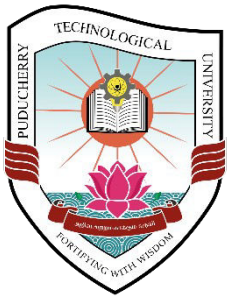
PUDUCHERRY TECHNOLOGICAL UNIVERSITY



IT225 - SOFTWARE ENGINEERING LABORATORY

SUBMITTED BY:
GANESH A (20IT1021)
DEENA R (20IT1018)

ONSPOT VEHICLE SERVICE SYSTEM

SOFTWARE REQUIREMENTS SPECIFICATION (SRS)

ABSTRACT

Onspot vehicle service system is a technological solution that enables vehicle owners to receive on-demand repair and maintenance services for their vehicles. The system is designed to provide a convenient and efficient way for vehicle owners to have their vehicles serviced, without the need to physically take their vehicles to a repair shop. Through the use of a website, vehicle owners can request for a technician to come to their location and provide the necessary services. The services may range from simple maintenance tasks such as oil changes and tire rotations, to more complex repairs such as engine diagnostics and brake system repairs.The system relies on a network of trained and certified technicians who are dispatched to the customer's location. The technicians are equipped with the necessary tools and equipment to diagnose and fix the issues with the vehicle on the spot.

LITERATURE SURVEY

Web pages related to onspot service by various companies were reviewed and studied to acquire a general image of the existing knowledge on the topic “Onspot Vehicle Service System.”

1. Readyassist does a great job in providing overall services provided in onspot vehicle .It works based on subscription basis. This was modified in our system user can only pay for only whenever required.

2. Ondemandcarcare.provides various vehicle cleaning services where anybody having a license and and good experience in handling car can apply for job. Thus extending this idea.Our system hiers mechanics based on their experience and are certified.

3. GoMechanics hires mechanics and sends their technicians when customers require for Services.Issue raises when multiple customers request for service. In this system technicians who are certified or who have their own shops are hired and when the service requires the vehicle to be taken to garage they can move them to their own garage which increases scalabiliy.

4. Parcelforce assign courier delivery orders to delivery boys based on who is near to the customer.Extending this concept technicians are assigned services who are near to the customer.

5 Swiggy Provides the delivery boys the offer of choosing themselves timing in whichthey are going to work and provide extra money those who work after 10.pm.Extending this concept mechanics are allowed to set their own timing and will be provided extra salary when they work on night.Some other service providers hiers technicians and gives them service courses ,provide proper training through either virtual courses or offline classes.

6 "Onspot Service: A New Way of Servicing Your Vehicle" by A. Sudhir and M. Venkateswarlu. This article discusses the concept of onspot vehicle service and its benefits, such as convenience, time-saving, and cost-effectiveness.

7 "On-Demand Vehicle Services: The Future of Car Maintenance?" by A. O'Donnell. This article provides an overview of onspot vehicle service and its potential to disrupt the traditional repair shop model.

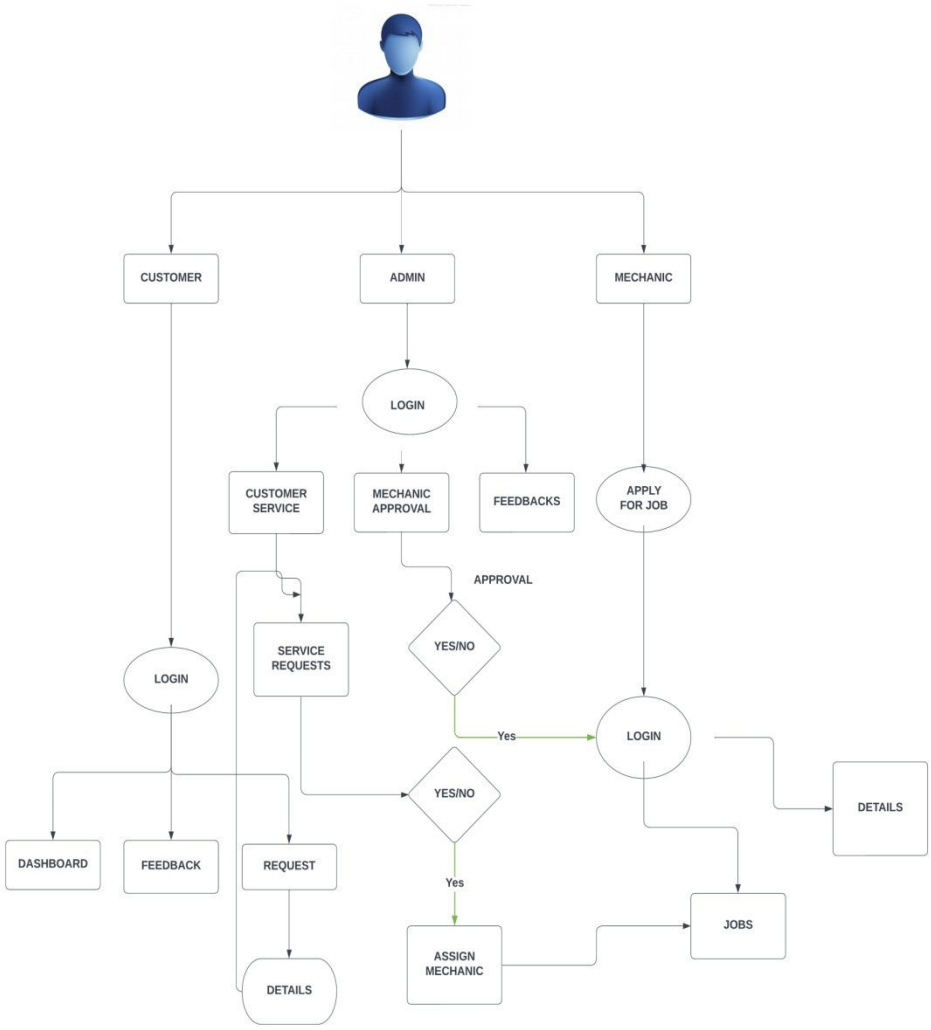
PROBLEMS IN EXISTING SYSTEM:

In most of the existing the issue is the scalability .When multiple customers require for a service or the vehicle should be taken to garage to repair the scalability issue raises.The existing systems are also works based on subscriptions charges where a customer had to pay a large sum of money.

PROPOSED SYSTEM:

In this system the mechanics are hired who have their own garage which and can work on their own working hours.Here it is not restricted to a particular company any mechanic could apply for services which increases scalability. Customers can also pay only when they request service.

HIGH-LEVEL ARCHITECTURE



Modules:

1 Admin
2 Customer
3 Mechanic

1 ADMIN:

Login as a Admin, who can approve mechanics who apply for a job and approve requests from the customers requesting for services. Further superadmin can maintain details about the mechanics work hours. Admin functionalities include,

1 Approve Mechanic

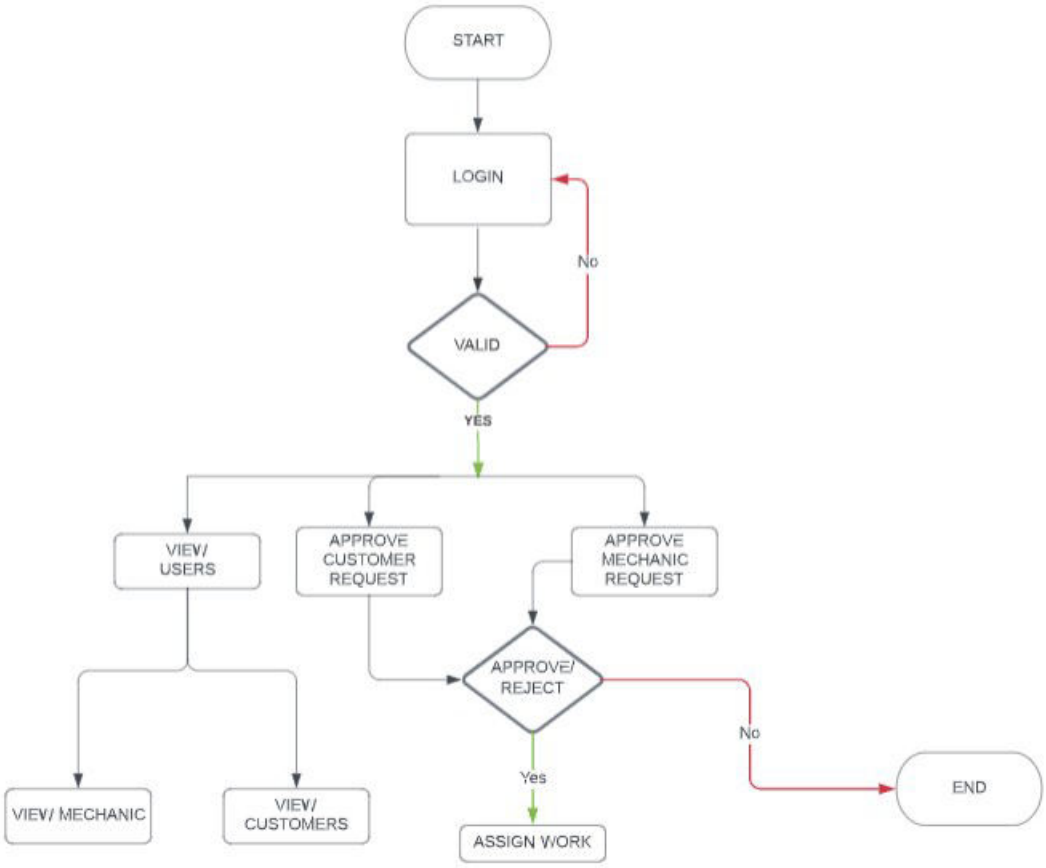
2 Approve Customer Requests

3 Assign Mechanic

4 Maintain user Details

5 Add Customer

6 Add Mechanic



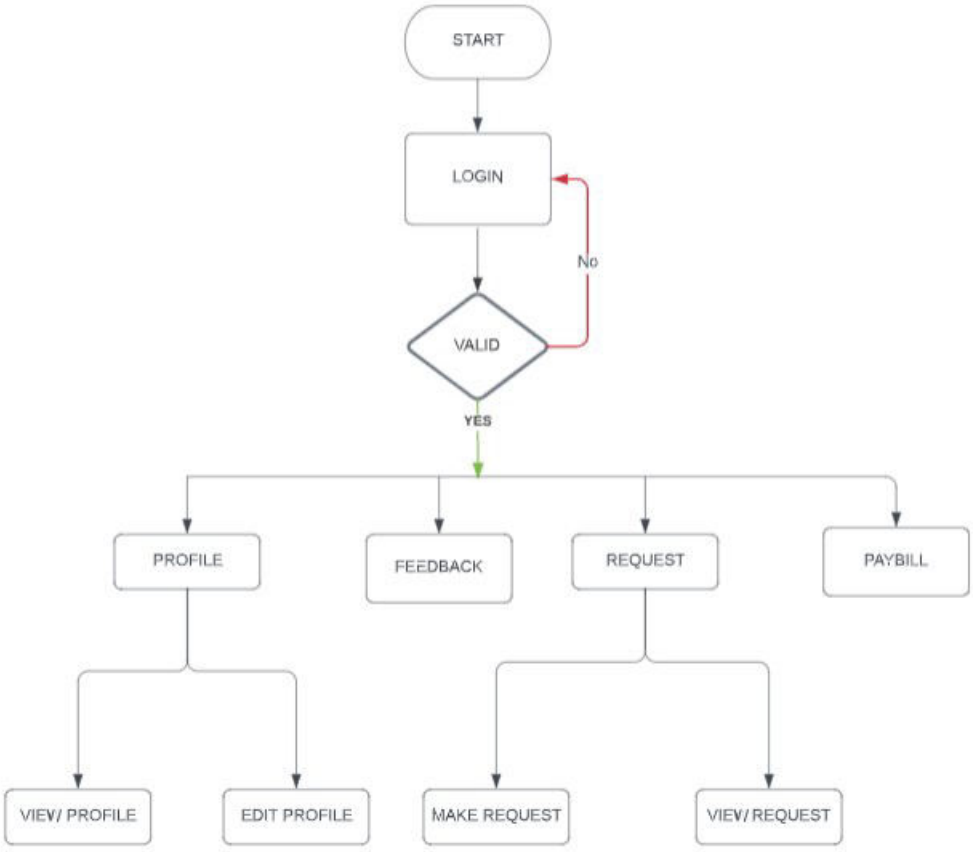
2 CUSTOMER:

Customers can make request for services and can provide details about their vehicle and problem. Customers can also sent feedbacks.Customer login includes following features

1 Problem Description

2 Service Request

3 Feedbacks



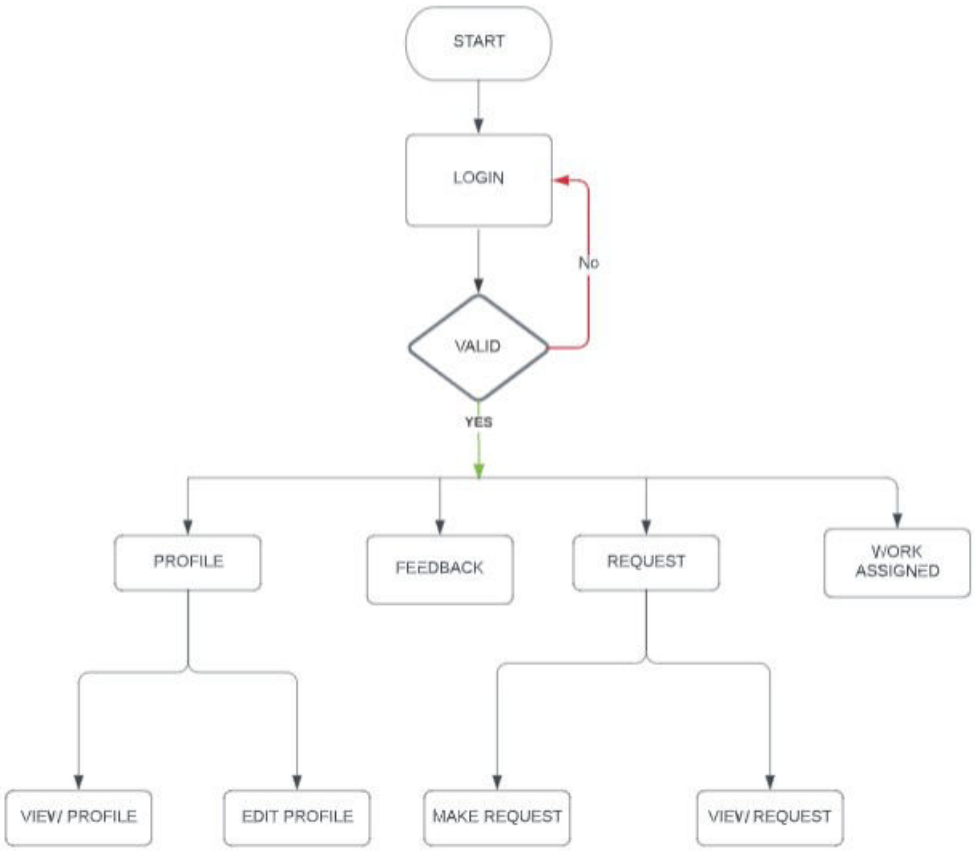
3 MECHANIC:

Mechanic can make apply for job and can check details about their assigned job. Mechanics can also sent feedbacks. Mechanic login includes following features

1 Apply For Job

2 Job Details

3 Feedbacks



NON-FUNCTIONAL REQUIREMENTS:

Reliability:

Reliability refers to the ability of the system to function properly and consistently over time. In the context of an on-demand vehicle service system, this means that the system should be available and responsive at all times, with minimal downtime or service disruptions. The system should be designed with redundancy and failover mechanisms in place to ensure that it can continue to function even if one or more components fail

Scalability:

Scalability refers to the ability of the system to handle increasing numbers of users and requests. In the context of an on-demand vehicle service system, this means that the system should be able to handle a large volume of requests and users without experiencing significant delays or performance issues. The system should be designed with scalability in mind, with the ability to add more servers or resources as needed to accommodate increasing demand.

Performance:

Performance refers to the speed and efficiency with which the system can process user requests and deliver results. In the context of an on-demand vehicle service system, this means that the system should respond quickly to user requests and provide a fast, seamless experience. The system should be designed with performance optimization techniques in mind, such as caching, load balancing, and data compression, to minimize latency and delays.

Security:

Security refers to the measures put in place to protect the system and user data from unauthorized access or attacks. In the context of an on-demand vehicle service system, this means that the system should be designed with robust security measures, such as encryption, firewalls, and intrusion detection and prevention systems. The system should also be regularly audited and updated to address new security threats as they arise.

Usability:

Usability refers to how easy and intuitive the system is to use for the average user. In the context of an on-demand vehicle service system, this means that the system should have a user-friendly interface that is easy to navigate and understand. The system should also provide clear and concise instructions and feedback to help users accomplish their goals quickly and efficiently.

Accessibility:

Accessibility refers to the ability of the system to be used by people with disabilities or impairments. In the context of an on-demand vehicle service system, this means that the system should be designed with accessibility features in mind, such as support for screen readers, alternative input devices, and other assistive technologies.

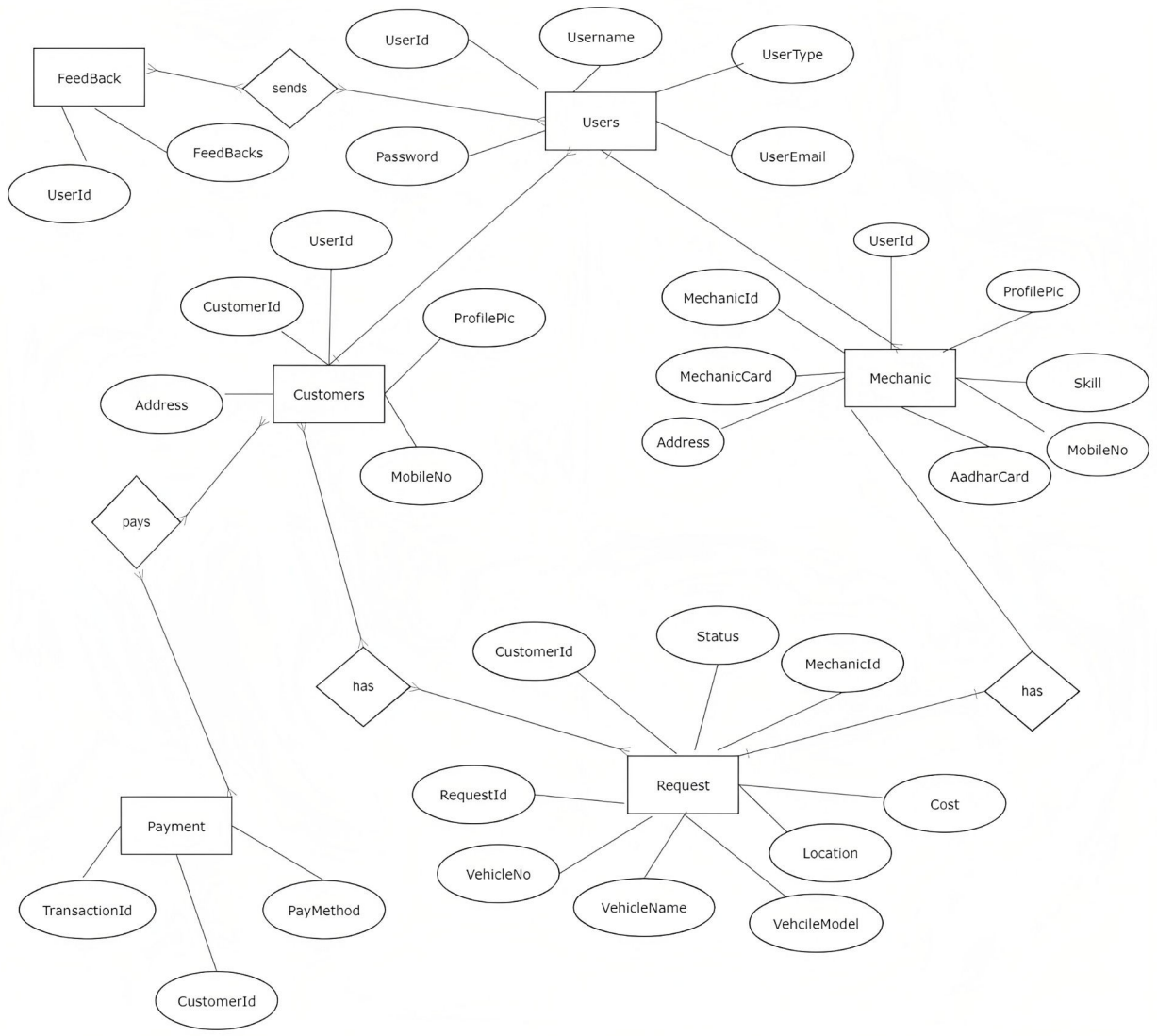
Compatibility:

Compatibility refers to the ability of the system to work with a wide range of devices and platforms. In the context of an on-demand vehicle service system, this means that the system should be compatible with a variety of web browsers, mobile devices, and operating systems, to ensure that users can access the system from any device they choose.

Maintainability:

Maintainability refers to how easy it is to maintain and update the system over time. In the context of an on-demand vehicle service system, this means that the system should be designed with maintainability in mind, with clear and well-documented code, modular architecture, and automated testing and deployment processes. The system should also be regularly updated to address bugs and other issues as they arise.

ER DIAGRAM:


FeedBack
sends
UserId
Username
UserType
Users
FeedBacks
Password
UserEmail
UserId
UserId
CustomerId
ProfilePic
Address
Customers
MobileNo
pays
UserId
MechanicId
ProfilePic
MechanicCard
Mechanic
Skill
Address
MobileNo
AadharCard
has
CustomerId
Status
MechanicId
has
RequestId
Request
Cost
Location
VehicleNo
VehicleName
VehcileModel
Payment
TransactionId
PayMethod
CustomerId

Hardware Requirements

- Processor: Pentium 4 or higher
- RAM: 512 MB or more
- Memory Space 80 GB or higher.

Software Requirements

- Django3.2
- Sqlite3 Database
- Python
- Browser
- Firebase

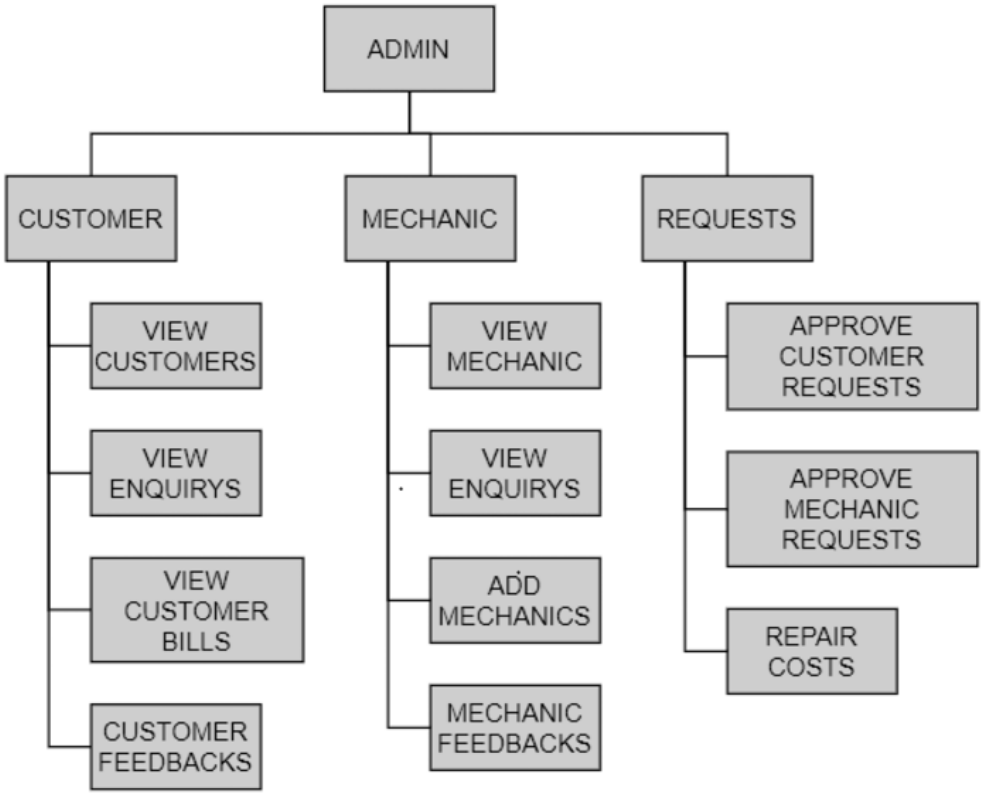
SDS

Software Design Specification

DECOMPOSITION OF MODULES:

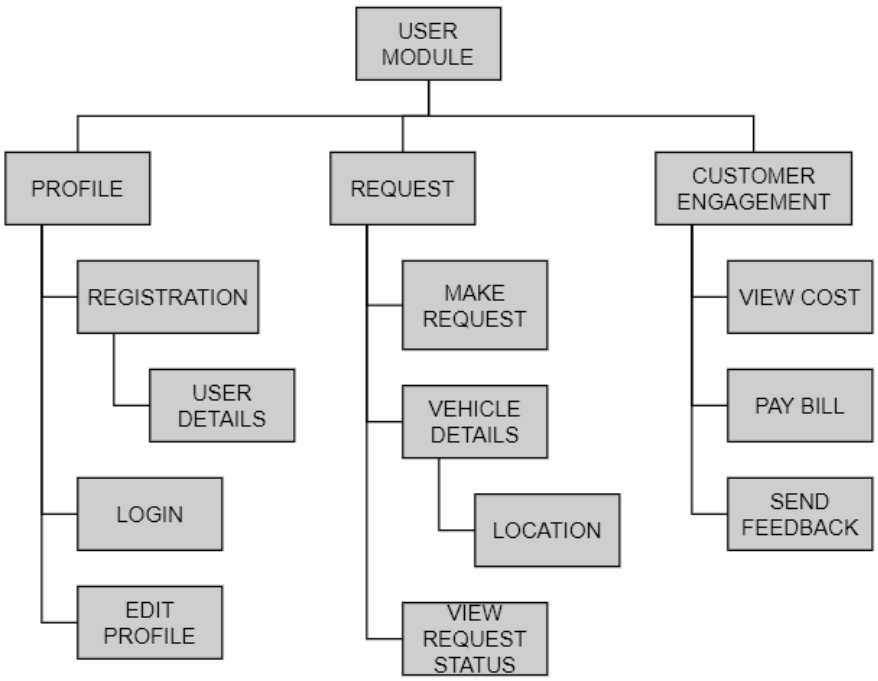
MODULE 1: Admin.

- Customers
  - View Customer
  - View Enquiries
  - View Customer Bills
  - Customer Feedbacks
- Mechanic
  - View Mechanics
  - View Enquiries
  - Add Mechanics
  - Mechanic Feedbacks
- Requests
  - Approve Customer Requests
  - Repair Costs
  - Approve Mechanic Requests



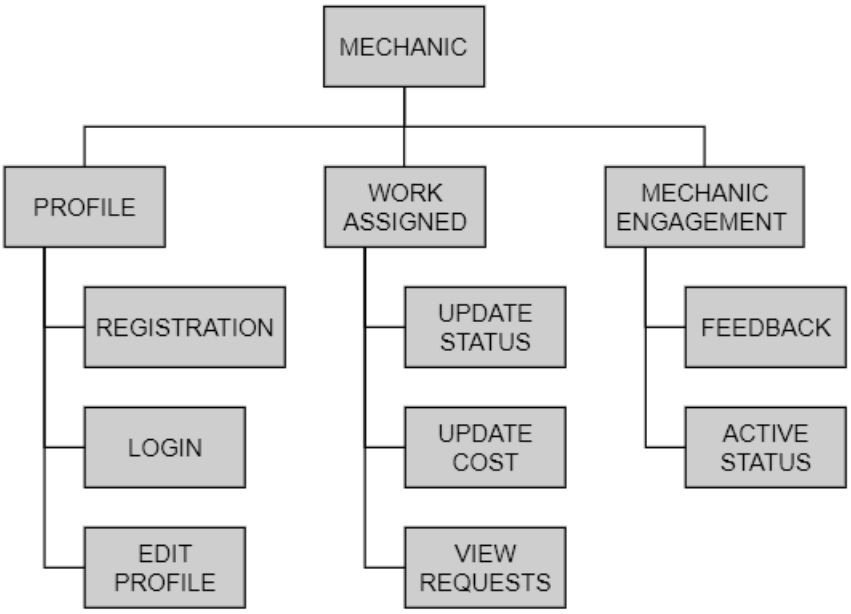
MODULE 2: Customer

- Profile
  - Registration
  - Login
  - Edit Profile
- Request
  - Make Request
  - Vehicle Details
  - View Request Status
- Customer Engagement
  - View Cost
  - Pay Bill
  - Send Feedback



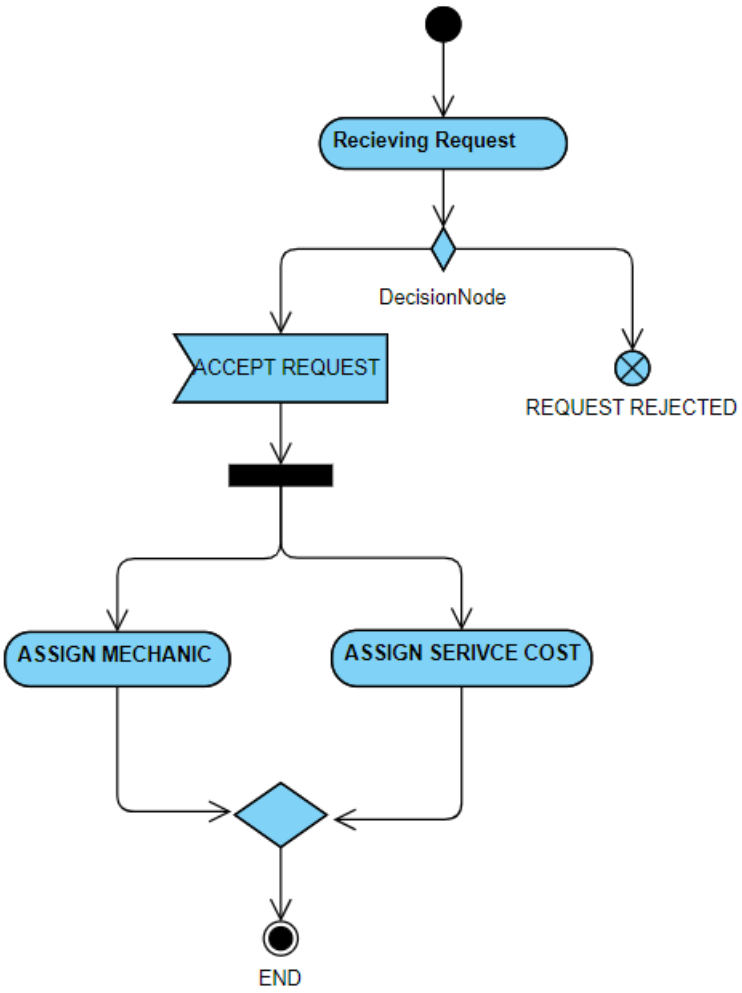
MODULE 3:Mechanic

- Profile
  - Registration
  - Login
  - Edit Profile
- Work Assigned
  - Update Status
  - Update Cost
  - View Requests
- Customer Engagement
  - Feedback
  - Active Status



ACTIVITY CHART:

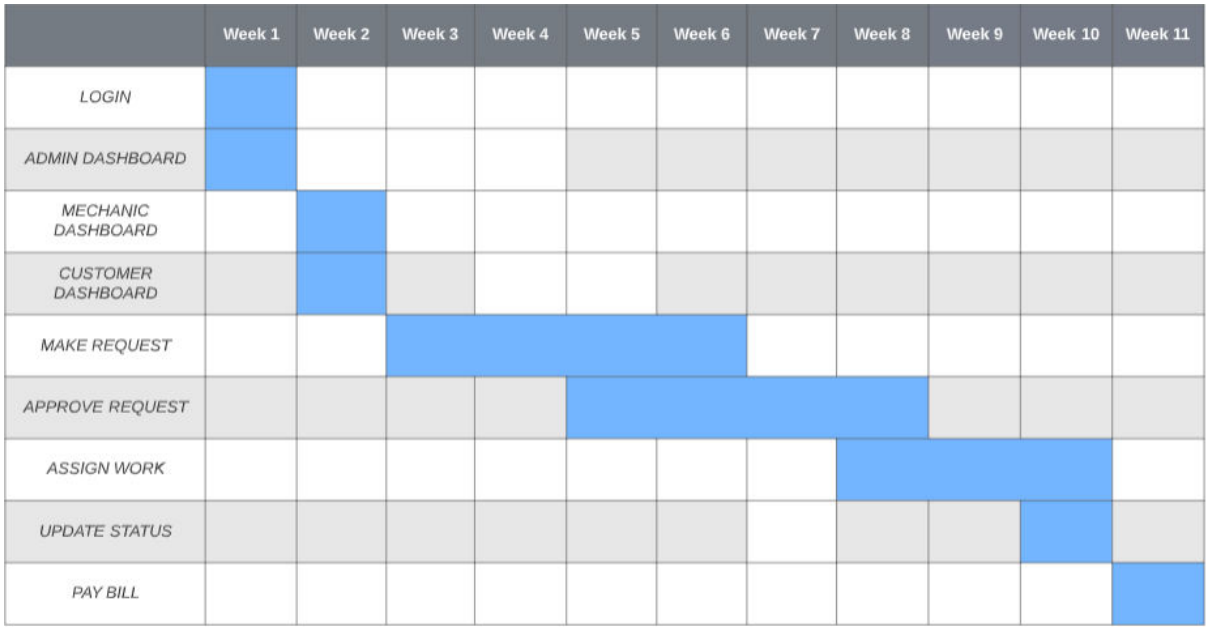
1 Requirement gathering and analysis
2 Designing the database schema
3 Setting up the server environment (e.g. Sqlite Studio, VS Code)
4 Creating the login pages
5 Creating the Admin dashboard pages
6 Creating the Customer dashboard page
7 Creating the Mechanic dashboard page
8 Creating the Customer Request page
9 Designing the website layout using HTML/CSS/BOOTSTRAP
10 Implementing user interface interactions using JavaScript
11 Integrating the backend code with the user interface using Python DJANGO
12 Testing the application for functionality, security, and performance
13 Deploying the application to a live server
14 Providing user support and maintenance



TIME SCHEDULING FOR ENTIRE PROJECT:

To develop the project, we need at least 90 days to complete.

DEVELOPMENT PHASE	90 DAYS						DURATION (DAYS)
	0 – 15 DAYS	16 – 30 DAYS	31 – 45 DAYS	46 – 60 DAYS	61 – 75 DAYS	76 – 90 DAYS	
REQUIREMENTS GATHERING							5
ANALYSIS							5
DESIGN							12
CODING							45
IMPLEMENTATION							14
TESTING							6
DEPLOYMENT & MAINTENANCE							2
DOCUMENTATION							
TOTAL TIME (DAYS)							90



TASK NAME	START DATE	END DATE	DURATION (days)
Project start	06/01/2023	24/01/2023	18
User Registration	25/01/2023	28/01/2023	3
Login /Logout	29/01/2023	31/01/2023	2
Admin dashboard	01/02/2023	05/02/2023	4
Customer Dashboard	06/02/2023	09/02/2023	3
Mechanic Dashboard	10/02/2023	08/03/2023	4
Make Request	17/02/2023	15/03/2023	26
Approve Request	16/03/2023	26/03/2023	26
Assign Work	26/03/2023	05/04/2023	16
Update Status	05/04/2023	07/04/2023	2
Pay Bill	07/04/2023	13/04/2023	6
Project End	14/04/2023	-	0

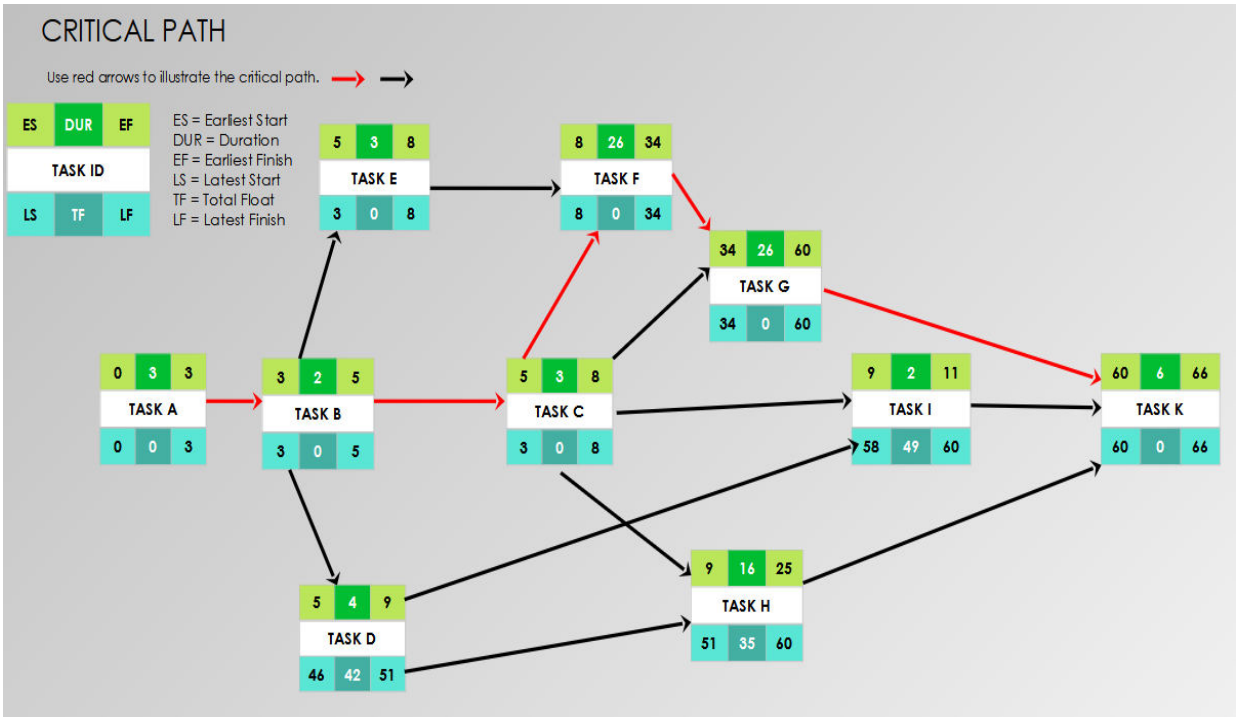
TASK DEPENDENCY:

Task	Duration (days)	Dependencies	Effort (person-days)
Home Page	3	-	5
Login Page	2	-	5
Admin dashboard	3	Login Page	10
Mechanic Dashboard	4	Login Page	25
Customer Dashboard	3	Login Page	20
Make Request	26	Customer Dashboard	30
Approve Request	26	Admin Dashboard/Make Request	8
Assign Work	16	Admin Dashboard/Mechanic Dashboard	8
Update Status	2	Admin Dashboard/Mechanic Dashboard	2
Pay Bill	6	Customer Dashboard	3
Website Design	15	All the above tasks	45

COST ESTIMATION:

1. Average human required = 2 persons.
2. No. of days working on the project: 90 days
3. Software and Hardware Costs for 2 people ,90 days working for 6 hours per day on various resources / platforms .
4. Maintenance cost on after deploying the project
5. Based on the number of lines per code, Documentation process, Human resources and their efforts it costs around 40,000 rupees for development of this project

CRITICAL PATH FOR THE PROJECT:



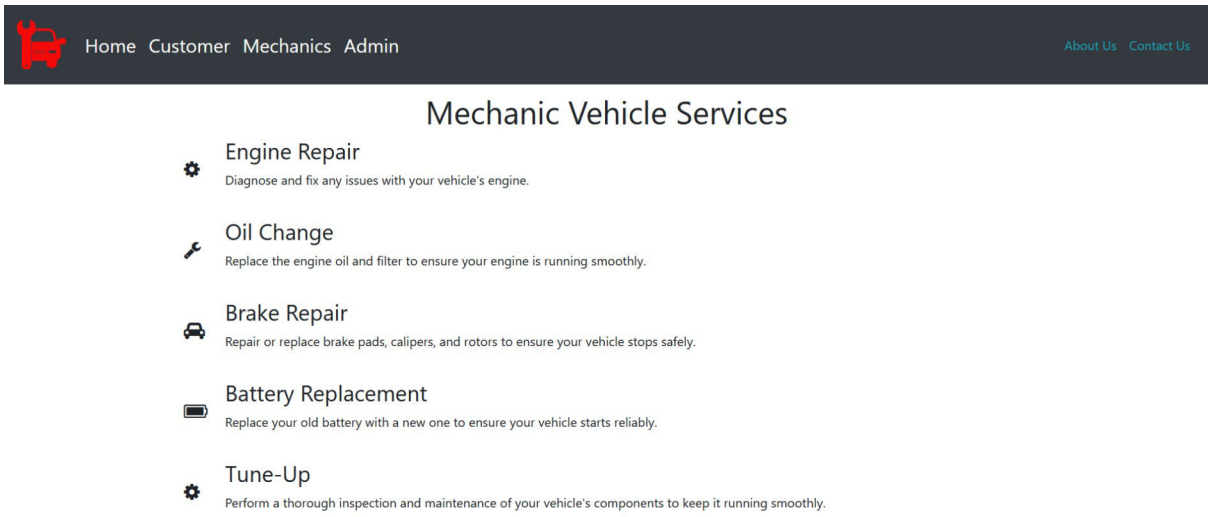
TABULATION

S no	Module	Task	Activities
1	Admin (Existing)	Customers	1)Admin view all the customers 2)Admin view all requests 3)Admin view customer bills 4)Admin view customer Feedback 5)Admin add customers
		Mechanics	1)Admin view all the Mechanics 2)Admin view all requests 3)Assign Mechanic Salary 4)Admin view Mechanic Feedback 5)Admin add Mechanic
		Requests	1)Admin approve customer requests 2)Admin approve mechanic requests 3)Admin reject customer requests 4)Admin reject mechanic requests 5)Admin update repair costs 6)Admin view customer location 7)Admin assign nearby mechanic
2	Customer (Proposed)	Profile	1)Customer Register in Website 2)Customer View Profile 3)Customer Edit Profile Pic 4)Customer Update Profile Details 5)Customer View Dashboard
		Customer Engagement	1)Customer View Bill 2)Customer Pay Bill 3)Customer Send Feedback 4)Customer View Feedbacks 5)Customer Logout
		Requests (Proposed)	1)Customer Make Request Service 2)Customer Enter Vehicle Details 3)Customer Update Live Details 4)Customer View Request Status 5)Customer View Mechanic Assigned

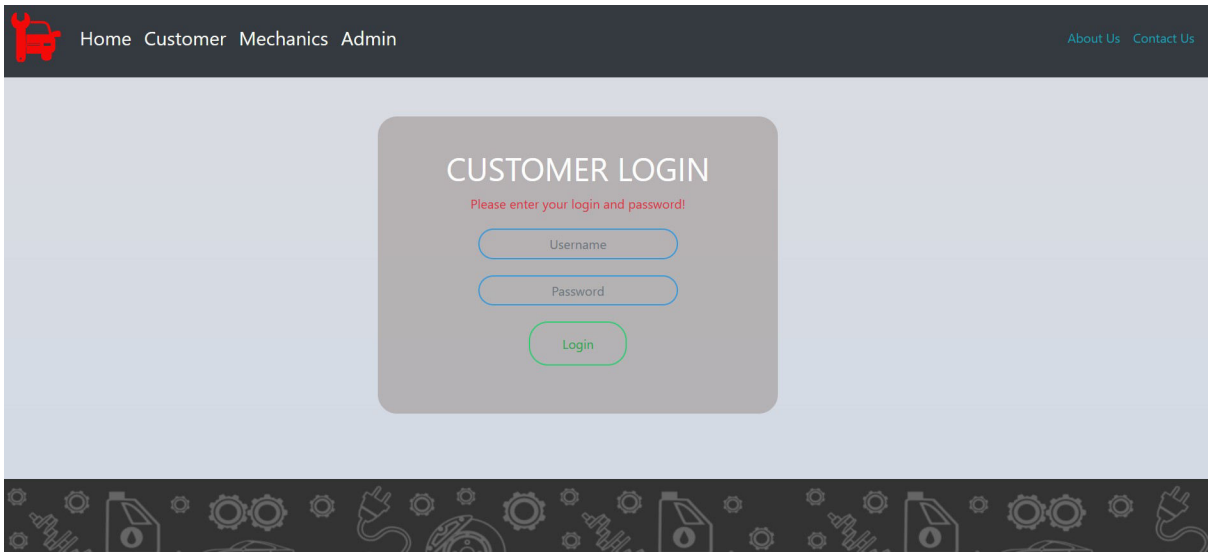
3	Mechanic (Existing)	Profile	1)Mechanic Login 2)Mechanic View Profile 3)Mechanic Edit Profile Pic 4)Mechanic Update Profile Details 5)Mechanic Apply For Job
		Mechanic Engagement	1)Mechanic Edit Active/Offline Status 2)Mechanic View Customer To Service 3)Mechanic Send Feedback 4)Mechanic View Feedbacks 5)Mechanic Logout
		Work Assigned (Proposed)	1)Mechanic View Dashboard 2)Mechanic View Work Assigned 3)Mechanic View Customer Location 4)Mechanic Update Request Service 5)Mechanic Update Bill

SCREENSHOTS

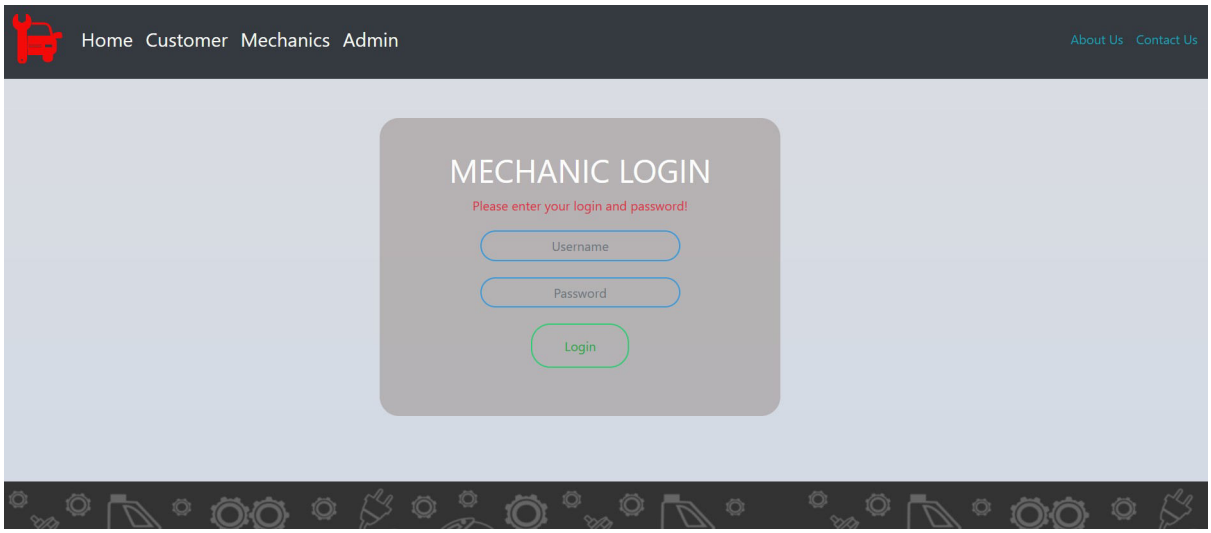
1) Homepage



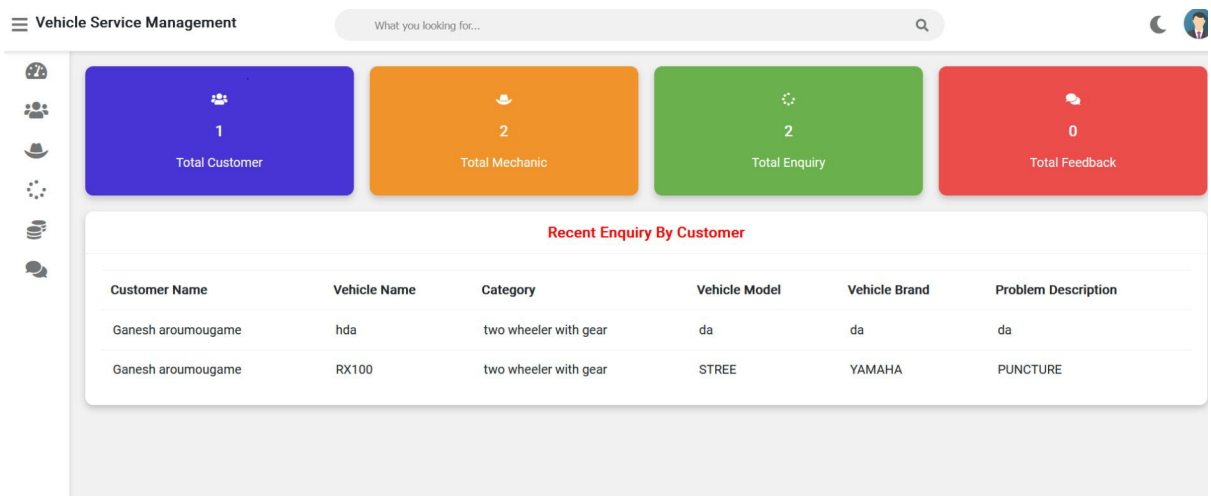
2)Customer Login Page



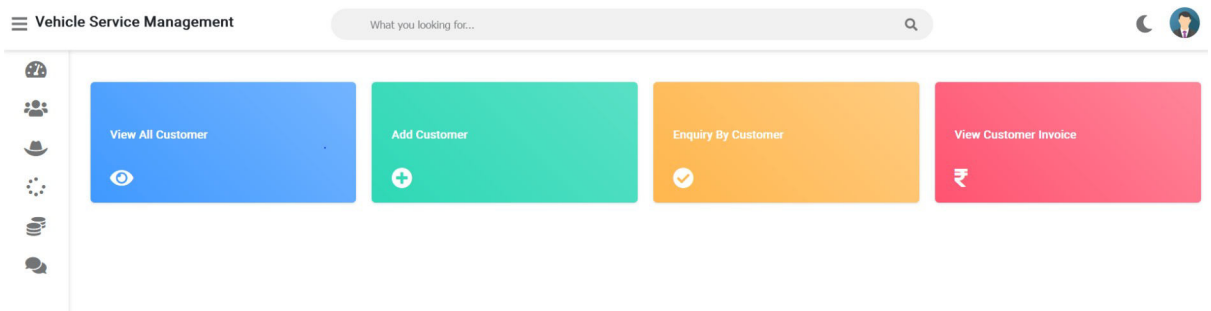
3)Mechanic Login



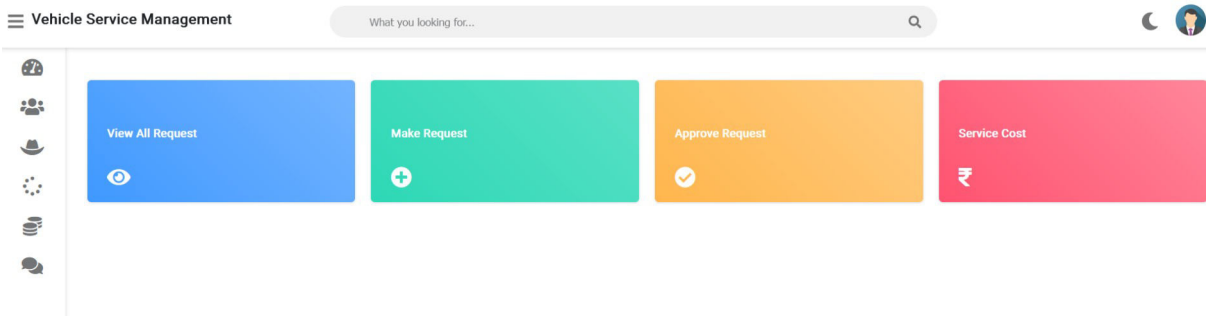
4)Admin DashBoard



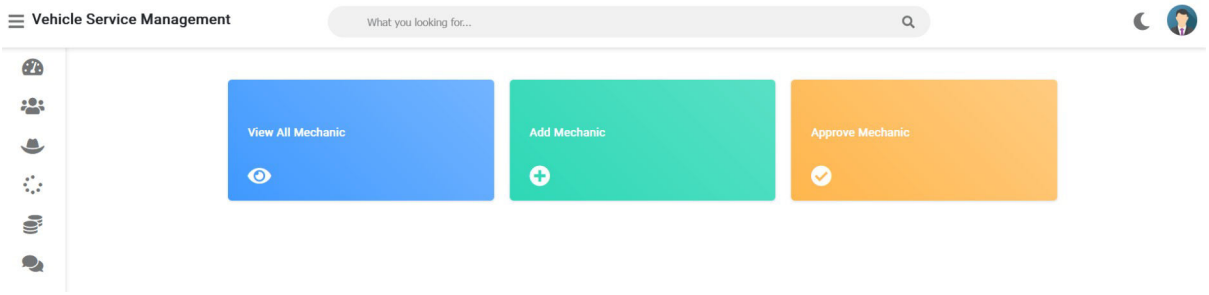
5)View Customers



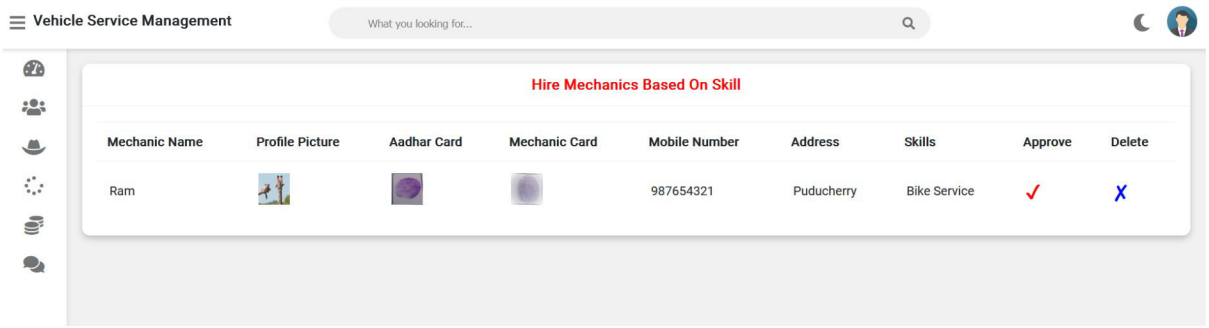
6)View Requests



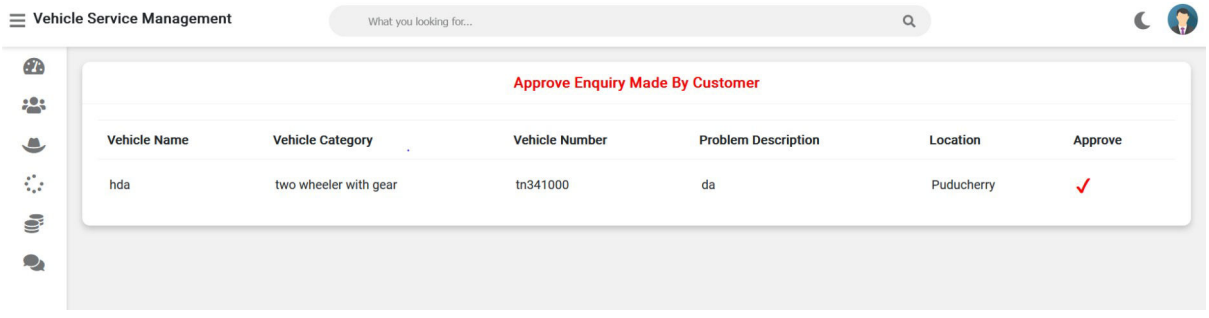
7)View Mechanics



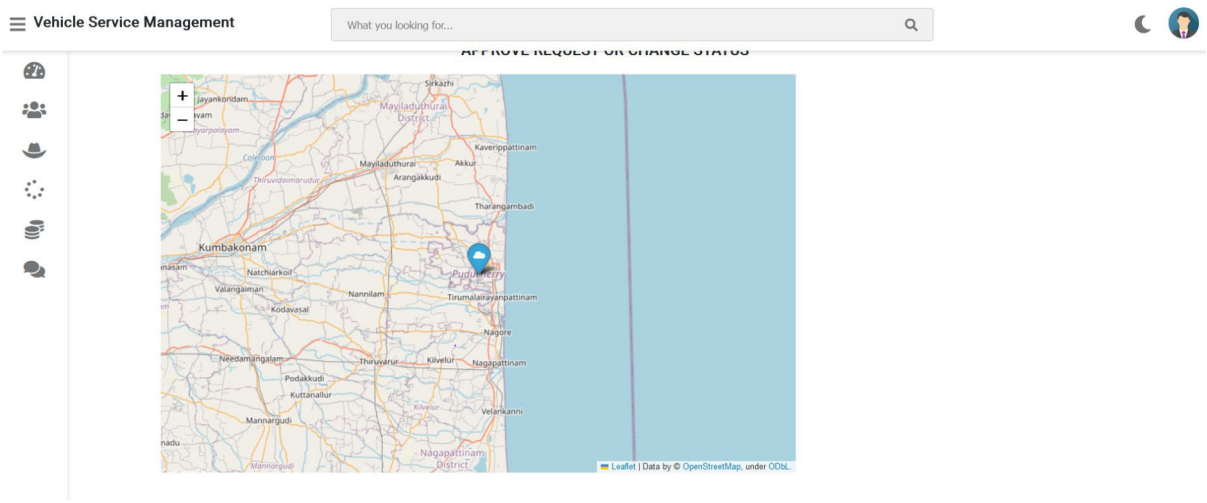
6)Approve Mechanic Requests



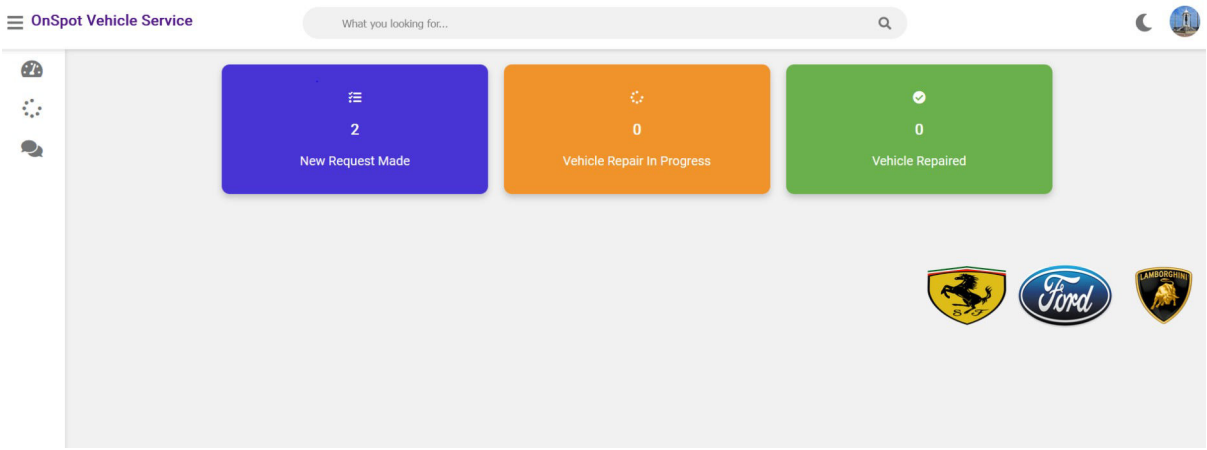
8)Approve Customer Requests



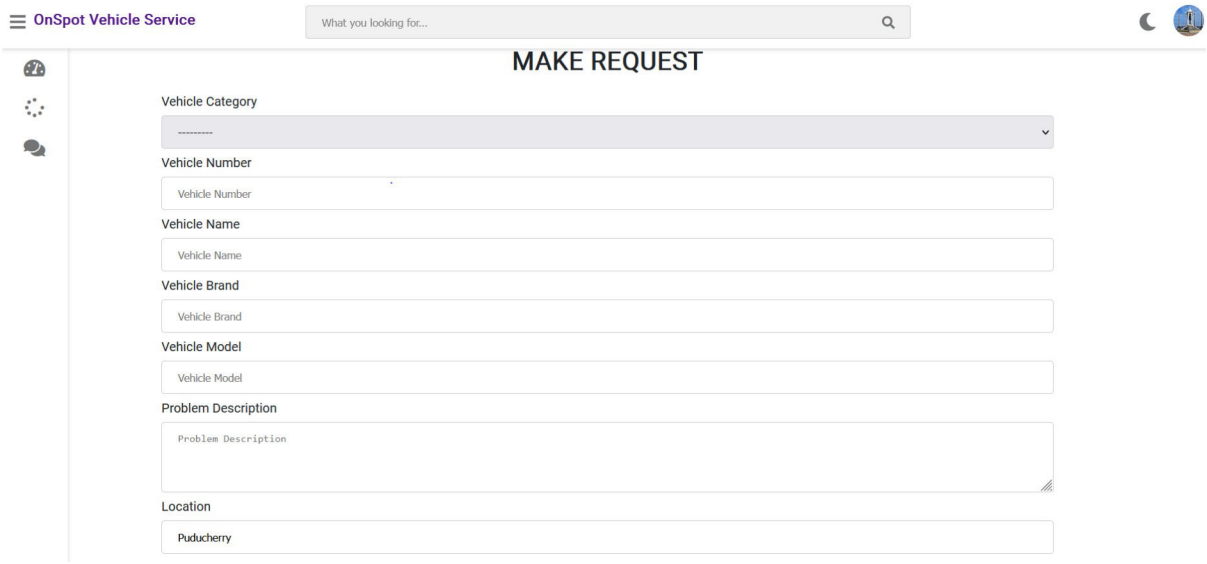
9)View Customer Location:



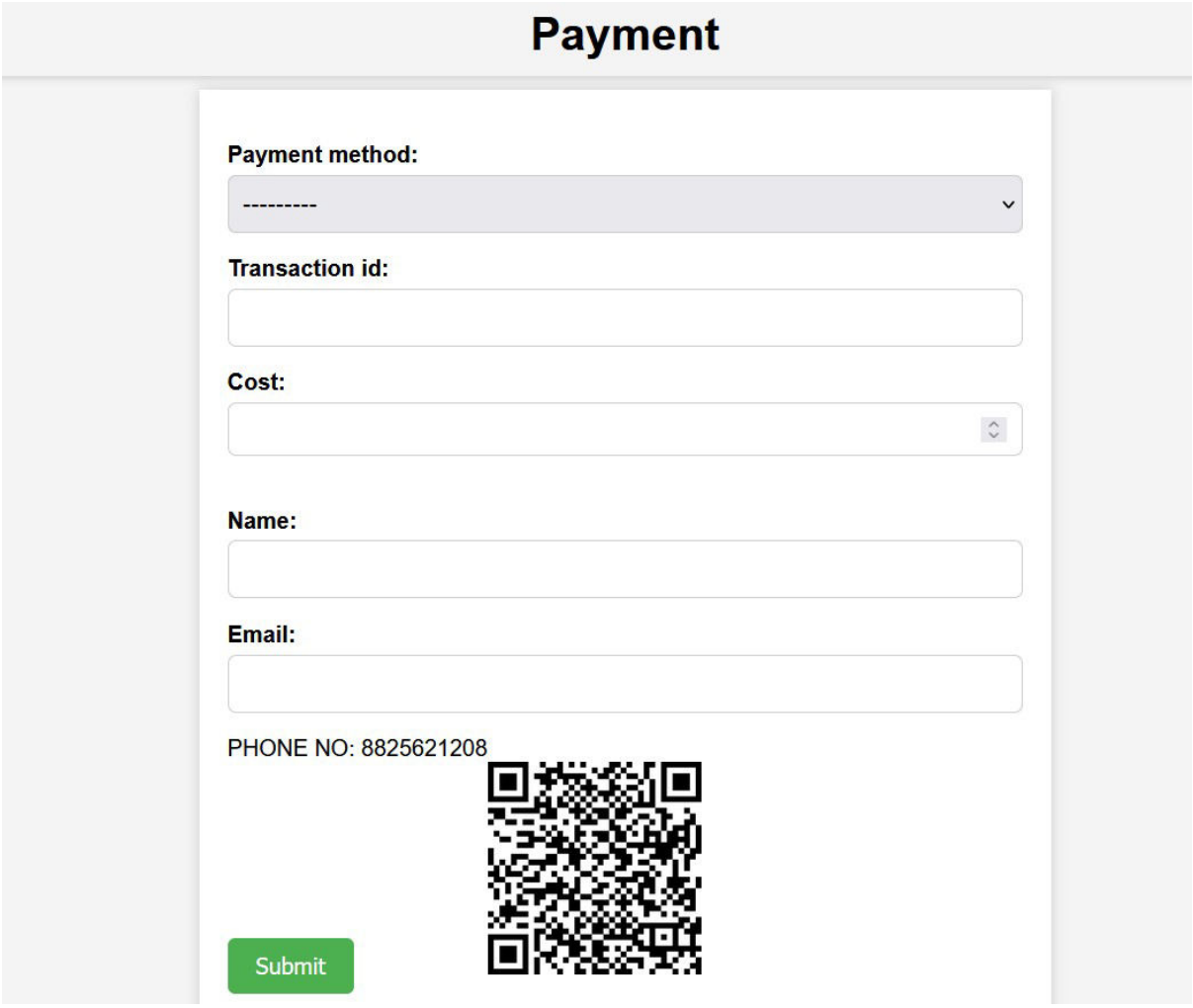
10)Customer Dashboard



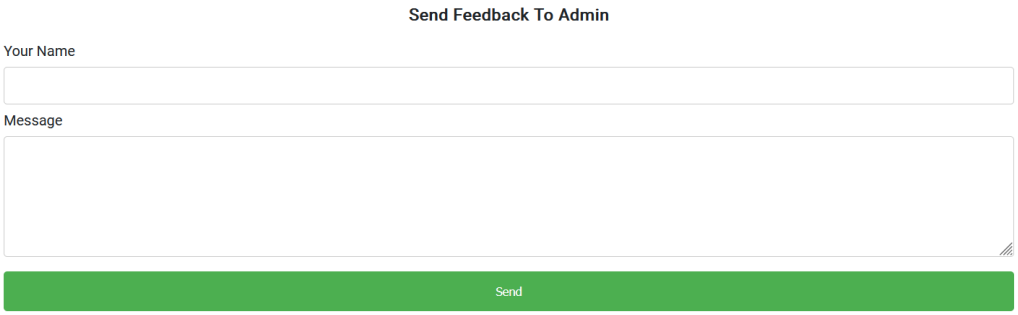
11)Customer Make Request



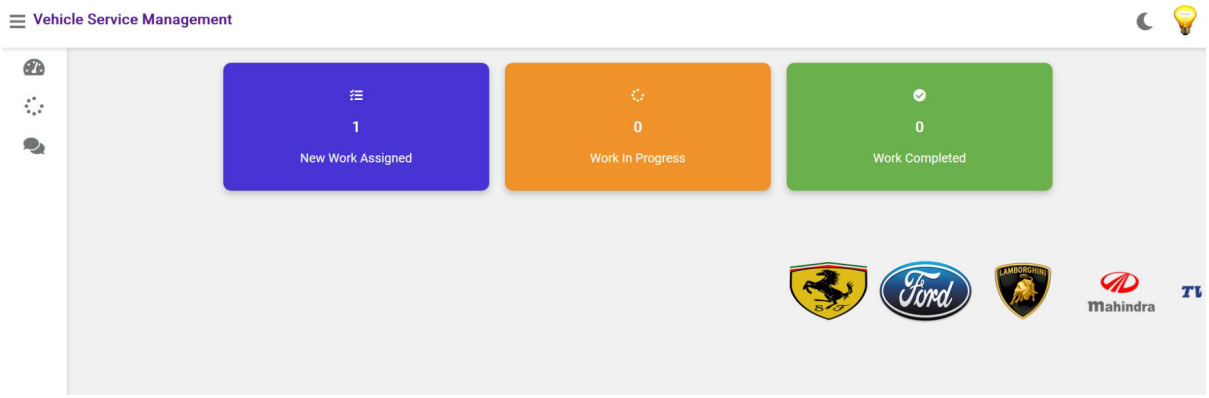
12)PayBill



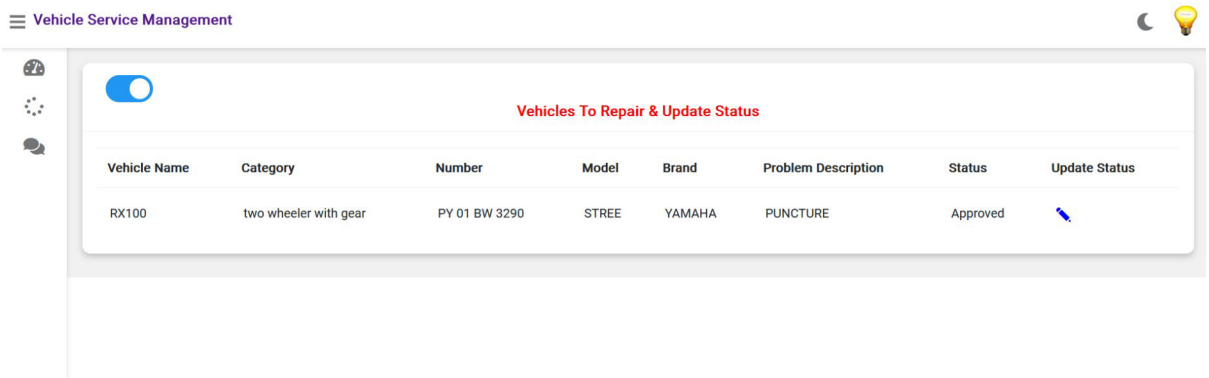
13)Feedback:



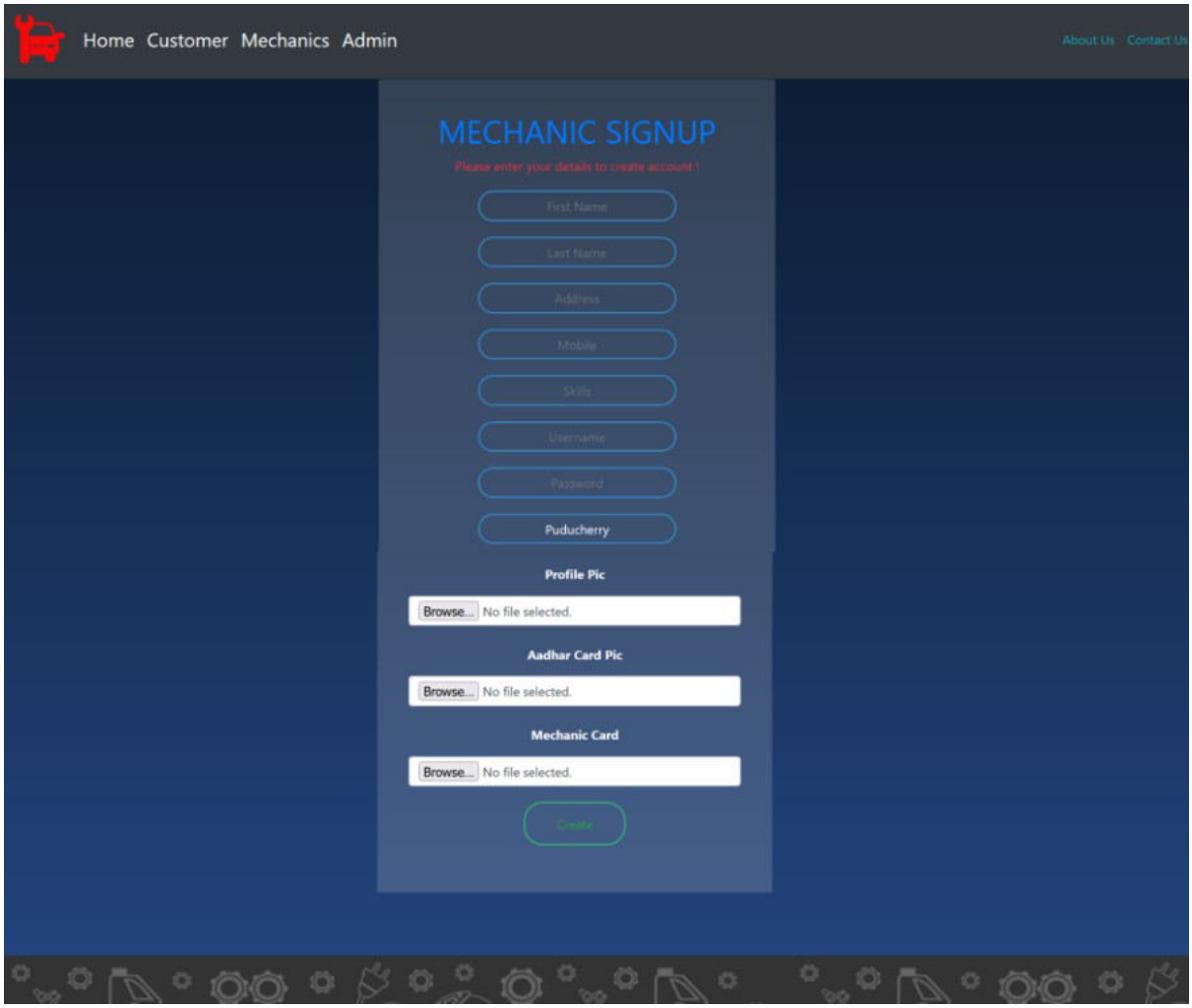
14)Mechanic Dashboard



15)Mechanic Work Assigned



16)Mechanic Registration



Tables:

1) User Table

USERNAME	EMAIL ADDRESS	FIRST NAME	LAST NAME	STAFF STATUS
Ram		Ram	s	✗
admin				✓
ghmli		Ganesh	aroumougame	✗
kumar	12345deena123@gmail.com	Kumar	K	✗
vinay	ganeshhip2003@gmail.com	Vinay	V	✗

2) Customer

Name	Data type	Primary Key	Foreign Key	Unique	Check	Not NULL
id	integer	🔑				✓
profile_pic	varchar (100)					
address	varchar (40)					✓
mobile	varchar (20)					✓
user_id	integer					✓

id	profile_pic	address	mobile	user_id
2	profile_pic/CustomerProfilePic/gy.jpg	krishna nagar,villianur	65984312474	4

3)MECHANIC TABLE:

id	profile_pic	address	mobile	skill	salary	status	mech_locatio	aadhar_card	mechanic_card	user_id	onoff
2	profile_pic/...	19,A,Perum...	7094880074	Bike Service	1000	1	Puducherry	aadhar_pic/...	mechanc_card/...	5	1
3	profile_pic/...	1,NehruStre...	8970965723	Bike Service	1000	1	Chennai	aadhar_pic/...	mechanc_card/...	6	1
4	profile_pic/...	Puducherry	987654321	Bike Service	1000	0	Puducherry	aadhar_pic/...	mechanc_card/...	7	1

4)REQUEST TABLE:

id	category	vehicle_nan	vehicle_mo	vehicle_bra	problem_des	location	date	cost	status	customer_ic	mechanic_ic	vehicle_no
5	two wheeler with gear	RX100	STREE	YAMAHA	PUNCTURE	VILLUPURAM	2023-04-13	200	Approved	2	2	PY 01 BW 3290
6	two wheeler with gear	hda	da	da	da	Puducherry	2023-04-17	NULL	Pending	2	NULL	tn341000

5)FEEDBACK TABLE:

id	date	by	message
4	20/4/2023	Vinay	The Customer Was Kind
3	20/4/2023	ganesh	Your Service was fantastic

TESTING

TESTING FOR THE PROJECT:

The on-demand vehicle repair service requires extensive testing to ensure that it meets the needs of the users, is reliable, and operates smoothly. There are several types of testing that are typically performed. They are

1. Unit Testing
2. Integrating Testing
3. Alpha Testing
4. Beta Testing
5. Verification Testing

UNIT TESTING:

The unit testing for the on-demand vehicle repair service implemented in Django was conducted to ensure that each module of the system is working as expected. The testing was performed using the Django test framework. The testing approach involved testing each module of the system individually, and then integrating them to ensure that they work together as expected. The unit testing was performed on a development environment using the following tools: Python 3.8, Django 3.2, Django Rest Framework 3.12, and Pytest 6.2. The unit testing included testing the following modules of the system: user authentication, vehicle diagnosis, repair cost estimation, repair scheduling, and payment processing. The unit testing helped to identify and fix any issues with the individual modules, ensuring that the system works seamlessly. Based on the unit testing results, it can be concluded that the on-demand vehicle repair service implemented in Django is working as expected.

INTEGRATION TESTING:

The integration testing for the on-demand vehicle repair service implemented in Django was conducted to ensure that the individual modules of the system are integrated properly and are working together as expected. The testing approach involved testing the interaction between different modules of the system. The integration testing was performed on a development environment using the following tools: Python 3.8, Django 3.2, Django Rest Framework 3.12.The integration testing included testing scenarios such as a user logging in, booking a repair service, and successfully making a payment, as well as scenarios where users receive error messages due to invalid information or

failed payments. All the scenarios passed successfully, indicating that the different modules of the system are integrated properly and are working together seamlessly. The integration testing helped to ensure that the on-demand vehicle repair service implemented in Django is functioning correctly.

ALPHA TESTING

The alpha testing for the on-demand vehicle repair service was conducted to evaluate the system's performance, reliability, and usability. The testing approach involved inviting a small group of internal users to use the system in a controlled environment. During the alpha testing, the internal users were asked to perform various tasks such as booking a repair service, making a payment, and providing feedback on their experience with the system. The feedback received from the users was used to identify any issues with the system and improve its performance, reliability, and usability. Based on the feedback received from the internal users, the system was found to be easy to use and navigate. Users were able to book a repair service and make a payment without any issues. However, a few issues were identified, such as slow loading times in certain parts of the system and some confusing messaging. Overall, the alpha testing was a success, and the feedback received from the internal users was used to improve the system's performance, reliability, and usability. The system is now ready for beta testing, where it will be tested by a larger group of external users in a real-world environment.

BETA TESTING:

The beta testing for the on-demand vehicle repair service was conducted to evaluate the system's performance, reliability, and usability in a real-world environment. The testing approach involved inviting a group of external users to use the system in a live environment, with real repair services being provided. Based on the feedback received from the external users, the system was found to be user-friendly and easy to use. Users were able to book a repair service and make a payment without any issues. However, a few issues were identified, such as a delay in receiving confirmation of the booking and some errors when attempting to process payments. The feedback received from the external users was used to improve the system's performance, reliability, and usability. The system was updated with bug fixes and performance improvements, and the payment processing system was improved to provide a smoother experience for users. Overall, the beta testing was a success, and the feedback received from the external users was used to improve the system's performance, reliability, and usability. The system is now ready for release, and we are confident that it will provide a high-quality on-demand vehicle repair service to our customers.

VERIFICATION TESTING:

Verification testing for the on-demand vehicle repair service was conducted to ensure that the system meets the specified requirements and performs as expected. The testing approach involved testing each individual feature of the system to ensure that it is functioning correctly. The verification testing included testing scenarios such as user authentication, booking a repair service, making a payment, and canceling a booking. Each scenario was tested to ensure that it met the specified requirements and performed as expected. Any issues identified during the testing were logged and addressed before proceeding with further testing. Based on the results of the verification testing, the system was found to meet all of the specified requirements and performed as expected. No major issues were identified, and any minor issues that were identified were addressed before proceeding with further testing. Overall, the verification testing was a success, and the system met all of the specified requirements. The testing helped to ensure that the on-demand vehicle repair service is functioning correctly and is ready for release.

CONCLUSION:

The Onspot vehicle service system offers several benefits, including convenience, time savings, and increased flexibility. It eliminates the need for vehicle owners to take time off work or disrupt their schedules to take their vehicles to a repair shop. It also provides a more efficient way of addressing vehicle issues, reducing downtime and increasing productivity. Overall, the onspot vehicle service system is a reliable and convenient solution for vehicle owners who value convenience and flexibility in their vehicle maintenance and repair needs.

REFERENCES

1. https://readyassist.in
2. https://www.ondemandcarcare.com/
3. https://gomechanic.in/
4. https://www.parcelforce.com/
5. https://www.swiggy.com
6. https://www.ijiset.com/vol1/ijisetv1i102
7. https://www.caranddriver.com/features/a21787205/on-demand-vehicle-services-the-future-of-car-mechanics