A

Mini-Project Report on

Vehicle Service Management System using Django

Submitted in partial fulfillment of the requirements
for the degree of
BACHELOR OF ENGINEERING
IN
Computer Science & Engineering
Artificial Intelligence & Machine Learning

By

Atharva Patil (22106039)

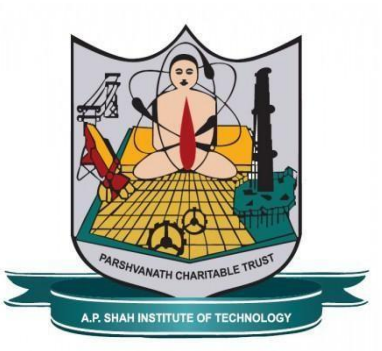
Brahmjot Singh(22106004)

Shraavani Salunkhe (22106031)

Sujal Yadav (22106083)

Under the guidance of

Prof. Viki Patil



Department of Computer Science & Engineering
(Artificial Intelligence & Machine Learning)
A. P. Shah Institute of Technology
G. B. Road, Kasarvadavali, Thane (W)-400615
University Of Mumbai

2023-2024

A. P. SHAH INSTITUTE OF TECHNOLOGY

CERTIFICATE

This is to certify that the project entitled "**Vehicle Service Management System using Django**" is a bonafide work of Atharva Patil (22106039), Brahmjot Singh (22106004), Shraavani Salunkhe (22106031), Sujal Yadav (22106083) submitted to the University of Mumbai in partial fulfillment of the requirement for the award **of Bachelor of Engineering in Computer Science & Engineering (Artificial Intelligence & Machine Learning).**

_____	_____
Prof. Viki Patil	Dr. Jaya Gupta
Mini Project Guide	Head of Department

A. P. SHAH INSTITUTE OF TECHNOLOGY

Project Report Approval

This Mini project report entitled "**Vehicle Service Management System using Django"** by Atharva Patil (22106039), Brahmjot Singh (22106004), Shraavani Salunkhe (22106031), Sujal Yadav (22106083) is approved for the degree of ***Bachelor of Engineering*** in ***Computer Science &Engineering***, (AIML)

2023-24.

External Examiner: _____

Internal Examiner: _____

Place: APSIT, Thane
Date:

Declaration

We declare that this written submission represents my ideas in my own words and whereothers' ideas or words have been included, I have adequately cited and referenced the original sources. I also declare that I have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in my submission. I understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission hasnot been taken when needed.

Atharva Patil	Brahmjot Singh	Shraavani Salunkhe	Sujal Yadav
(22106039)	(22106004)	(22106031)	(22106083)

ABSTRACT

This project describes the development of a Vehicle Service Management System (VSMS) using Django, a popular web framework. The VSMS aims to create a central hub for managing vehicles and their data. It will offer functionalities like user management with different access levels (like admin, customer, mechanic), comprehensive vehicle data capture, work status updates (on how much work is in progress or is completed), and an organized enquiry table for the admin. Depending on the project scope, additional features like a booking system, maintenance tracking, and reporting can be incorporated. The benefits include improved data organization, increased efficiency in bookings and maintenance, and streamlined communication for those involved in vehicle operations. Built with Django and a chosen database (e.g., PostgreSQL), this VSMS is ideal for organizations with fleets, car rental businesses, or anyone needing centralized vehicle management.

Keywords: Vehicle Service Management System (VSMS), Django, Access levels, Booking system, Data organization, Centralized management.

Index

CHAPTER 1
INTRODUCTION

1. INTRODUCTION

In our modern society, the sight of vehicles navigating through bustling streets has become a ubiquitous part of daily life. However, behind this seamless movement lies a complex network of maintenance, scheduling, and management tasks essential for ensuring the smooth operation of vehicles. From corporate fleets to rental services and individual vehicle owners, the need for effective vehicle service management has never been more critical.

Observing the current landscape of vehicle service management, it becomes evident that existing solutions often fall short in addressing the myriad challenges faced by stakeholders in this domain. Traditional methods reliant on manual record-keeping and disjointed communication channels lead to inefficiencies, delays, and increased operational costs. Moreover, the lack of centralized platforms exacerbates issues related to data disorganization, service scheduling conflicts, and suboptimal resource allocation.

Motivation:

The motivation to embark on developing a Vehicle Service Management System (VSMS) using the Django framework stems from real-time observations of these persistent challenges within the vehicle service domain. Witnessing firsthand the frustrations experienced by fleet managers, service technicians, and vehicle owners alike, there is a pressing need for a comprehensive, user-friendly solution that streamlines operations and enhances overall efficiency.

Objectives:

The primary objective of this project is to address the inherent issues plaguing vehicle service management through the development of a robust VSMS. By leveraging the capabilities of the Django framework, we aim to tackle the following key issues:

1. Inefficient Data Management: Implementing a centralized platform to capture, store, and organize vehicle data, including maintenance records, service history, and customer information, to eliminate data silos and improve accessibility.

2. Fragmented Communication: Facilitating seamless communication channels between stakeholders, such as administrators, mechanics, and customers, to ensure timely updates, status notifications, and resolution of service-related inquiries.

3. Suboptimal Resource Utilization: Optimizing resource allocation by automating service scheduling, prioritizing tasks based on urgency and availability, and providing real-time

insights into workforce utilization and service demand.

4. Lack of Scalability and Flexibility: Designing a modular and extensible system architecture that accommodates future growth and customization, allowing for the integration of additional features and adaptation to evolving business requirements.

By addressing these objectives, our proposed VSMS aims to revolutionize vehicle service management practices, empowering organizations to streamline operations, enhance customer satisfaction, and achieve greater operational efficiency in the dynamic landscape of the automotive industry.

CHAPTER 2
LITERATURE SURVEY

2. LITERATURE SURVEY

2.1-HISTORY

The history of vehicle service management systems can be traced back to the early stages of the automotive industry when basic maintenance records were manually documented. As the automotive sector advanced, the need for more efficient management systems became apparent, leading to the emergence of computerized solutions in the latter half of the 20th century.

Early vehicle service management systems primarily focused on basic maintenance scheduling and inventory management. These systems were often standalone software applications developed for specific organizational needs. However, with the rise of the internet and web technologies, there was a shift towards more sophisticated web-based solutions that offered centralized management capabilities and real-time data access.
In the late 1990s and early 2000s, the advent of web frameworks such as Django revolutionized the development of web applications. Django, with its robust features and rapid development capabilities, became a popular choice for building complex and scalable web-based systems, including vehicle service management systems.
Literature on vehicle service management systems using Django framework emerged as organizations recognized the benefits of leveraging modern web technologies for streamlining operations and improving efficiency. Research and case studies highlighted the advantages of using Django for developing vehicle service management systems, such as its modular architecture, built-in security features, and extensive community support.

Over the years, the literature on vehicle service management systems has evolved to cover various aspects, including:

1. System Architecture: Studies have explored different architectural patterns and design principles for building scalable and maintainable vehicle service management systems using Django.
2. Data Management: Research has focused on best practices for managing vehicle data, including storing, querying, and analyzing data efficiently using Django's ORM (Object-Relational Mapping) capabilities.
3. User Interface Design: Literature has emphasized the importance of user-centered design principles for developing intuitive and user-friendly interfaces for vehicle service management systems, leading to improved user satisfaction and productivity.

2.2-LITERATURE REVIEW

Analysis of Issues: Despite the availability of existing solutions, several common issues persist across the board. These include:

1. Fragmented Communication: Many existing systems lack seamless communication channels between stakeholders, leading to delays in service updates and resolution of issues.
2. Data Disorganization: Manual data entry and disparate systems result in data silos, making it challenging to access and analyze critical information.
3. Limited Scalability: Some solutions struggle to scale with growing business needs, hindering adaptability and future expansion.
4. Complex User Interfaces: Cluttered and unintuitive user interfaces contribute to user frustration and reduced productivity.

Literature Survey:

A literature survey reveals a wealth of research and development efforts focused on vehicle service management systems, driven by the growing complexities of managing modern vehicle fleets. Early studies often centered on optimizing maintenance schedules, improving resource allocation, and enhancing communication channels within organizations. With the advent of web technologies, there has been a shift towards web-based solutions offering centralized management capabilities, real-time data access, and integration with external systems.

Key themes identified in the literature include:

1. Centralized Data Management: Many studies highlight the importance of centralizing vehicle data to streamline operations and improve decision-making processes. By maintaining a centralized repository of vehicle information, organizations can easily access and analyze critical data related to maintenance schedules, service histories, and operational metrics.

2. Integration with External Systems: Integration with external systems such as GPS tracking devices, diagnostic tools, and maintenance databases is a common theme in the literature. By integrating these systems, organizations can leverage real-time data to optimize service schedules, diagnose issues proactively.

3. User Experience and Interface Design: Several studies emphasize the importance of user-friendly interfaces and intuitive design principles in vehicle service management systems. A well-designed interface can improve user satisfaction, reduce training time, and enhance overall productivity.

4. Scalability and Flexibility: Scalability and flexibility are key considerations in the design and implementation of vehicle service management systems. As organizations grow and evolve, the system must be able to adapt to changing requirements and accommodate an expanding fleet of vehicles.

Overall, the literature survey underscores the importance of effective vehicle service management in optimizing fleet operations, reducing costs, and improving overall efficiency. By leveraging the Django framework, organizations can develop robust, scalable, and customizable solutions tailored to their specific needs and requirements.

1. Title : Improvement of Vehicle Management System (IVMS)
Authors : Falah Y.H. Ahmed, Muthukumaran al Thiruchelvam, Sim Liew Fong.
Published in: 2019 IEEE International Conference on Automatic Control and Intelligent Systems (I2CACIS).
Summary: This research paper details the development of a Vehicle Management System (VMS) for Unihomes, aimed at addressing the challenges posed by conventional manual vehicle management practices. The system resolves issues such as lack of centralized vehicle and renting details, lengthy report generation processes, and booking inaccuracies. Developed using Microsoft Visual Studio 2016, Crystal Reports, and MySQL, the VMS adopts a relational database approach for efficient data storage and management. The Agile Unified Process methodology was employed for project development, ensuring flexibility and adaptability. By implementing this computerized membership management system, Unihomes can streamline operations, enhance data organization, and improve overall efficiency in vehicle management.

2. Title : Web-Based Automobile Service Management System for MAS Motors LLC

Authors: Abdelsalam Shahlol, Abigail Alix, Ace Lagman.

Published in: 2018 IEEE 10th International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment and Management (HNICEM).

Summary: This project aimed to develop a Web-based Automobile Service Management System for MAS Motors LLC, a Toyota dealership in Libya, to streamline their service division's operations across branches. With a focus on reducing manual processes and accommodating the needs of various stakeholders, including Service Advisers, Workshop Managers, Technicians, and Customers, the system aimed to address the inefficiencies arising from the company's growth. Through Alpha and Beta testing and assessment using the Hewlett-Packard FURPS model, the system was found to be functional and capable of meeting client requirements, providing a centralized solution to enhance business operations.

3. Title : Management System Design for Automobile Maintenance and Repair Based on Strusts2

Author: Quan-Peng Ji

Published in: 2021 2nd International Conference on Big Data & Artificial Intelligence & Software Engineering (ICBASE)

Summary: This article delves into the management challenges faced by auto repair companies and addresses the growing demand for informatization within the industry. By standardizing business processes and adopting efficient management practices, the article proposes the development of an Automobile Maintenance and Repair Management System. Leveraging the MVC design model and implemented with Struts2 technology, the system aims to streamline operations and improve customer experience through modular functionality and user-friendly interfaces. Divided into layers such as business processing, decision-making management, and system maintenance, the architecture ensures efficient management of auto repair enterprises while paving the way for future enhancements to further optimize operations and overall management efficiency.

CHAPTER 3

Problem Statement

3. PROBLEM STATEMENT

The conventional vehicle service management systems are plagued by inefficiencies and lack of real-time updates, leading to delays, suboptimal resource utilization, and poor customer satisfaction. Moreover, existing systems fail to consider the impact of sudden incidents or dynamically changing conditions, resulting in unreliable route suggestions and increased operational costs. To address these challenges, there is a pressing need to develop a modern, dynamic, and context-aware Vehicle Service Management System (VSMS) using the Django framework.

CHAPTER 4

Experimental Setup

4. EXPERIMENTAL SETUP

4.1 Hardware Setup

1. Networking Equipment:

- Network switch or router to connect your server to the local network or the internet.

2. Laptop

4.2 Software Setup

1.Django Framework:

- Install Django framework on your server. You can use pip, the Python package installer, to install Django: pip install Django.

2. Database Management System:

- Install and configure your chosen database management system. For example, if you're using PostgreSQL:

- Install PostgreSQL: sudo apt-get install postgresql

- Install psycopg2, the PostgreSQL adapter for Python: pip install psycopg2

- Configure your web server to serve your Django application.

4. Version Control System:

- Use Git for version control to manage your Django project's source code. Set up a Git repository to track changes and collaborate with team members if necessary.

5. Python Virtual Environment:

- Create a Python virtual environment to isolate your Django project's dependencies.

```
python3 -m venv myenv
source myenv/bin/activate # Activate the virtual environment
```

- Install Django and other dependencies within this virtual environment.

6. Django Packages and Libraries:

- Install any additional Django packages and libraries required for your vehicle service management system. These may include:

- django-rest-framework for building RESTful APIs.

- django-crispy-forms for better form rendering.

- django-filter for filtering querysets dynamically.

- django-allauth for user authentication and account management.

- django-cors-headers for handling Cross-Origin Resource Sharing (CORS) if your frontend is separate.

pip install django-rest-framework django-crispy-forms django-filter django-allauth django-cors-headers

7. Development Tools:

- Install development tools such as IDEs (e.g., PyCharm, Visual Studio Code), text editors (e.g., Sublime Text, Atom), and debugging tools to facilitate development and troubleshooting.

8. Testing and Debugging:

- Set up testing frameworks like Django's built-in test framework, pytest, or unittest for automated testing of your Django application.

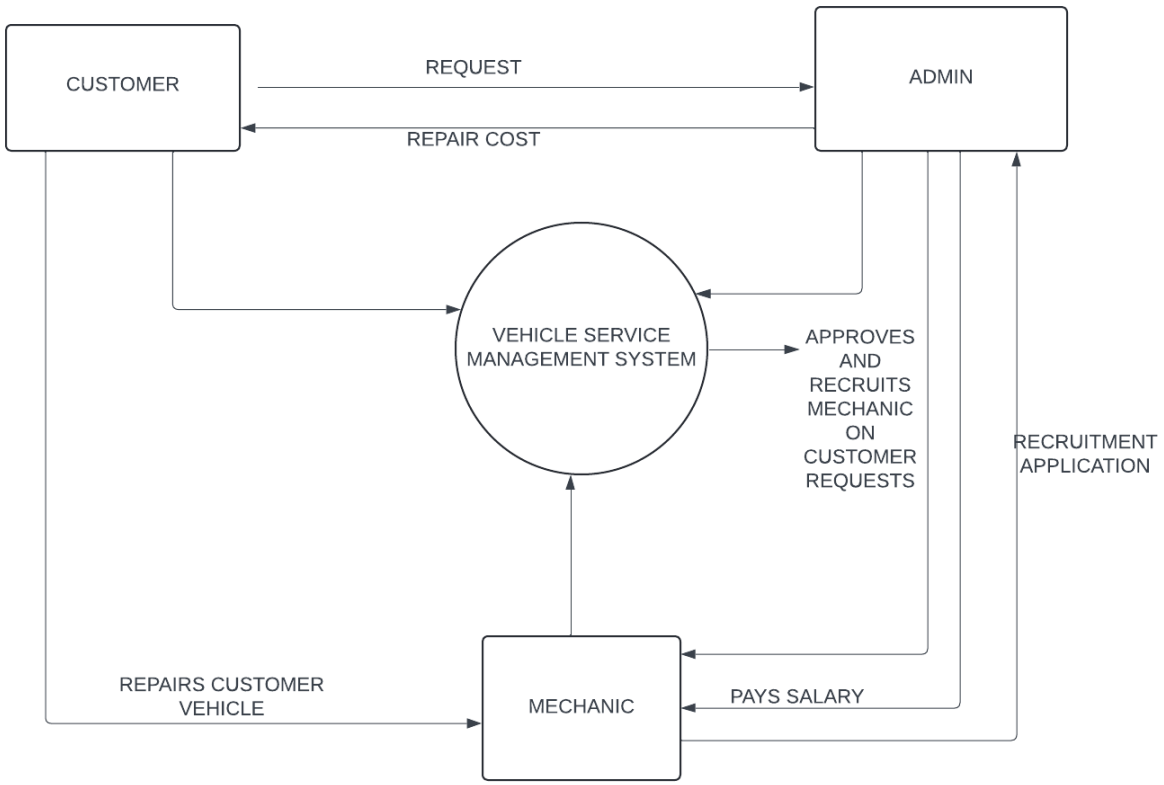
- Use Django Debug Toolbar or similar tools for debugging and optimizing your application's performance.

CHAPTER 5

Proposed System & Implementation

5. PROPOSED SYSTEM AND IMPLEMENTATION

5.1 Block diagram of proposed system



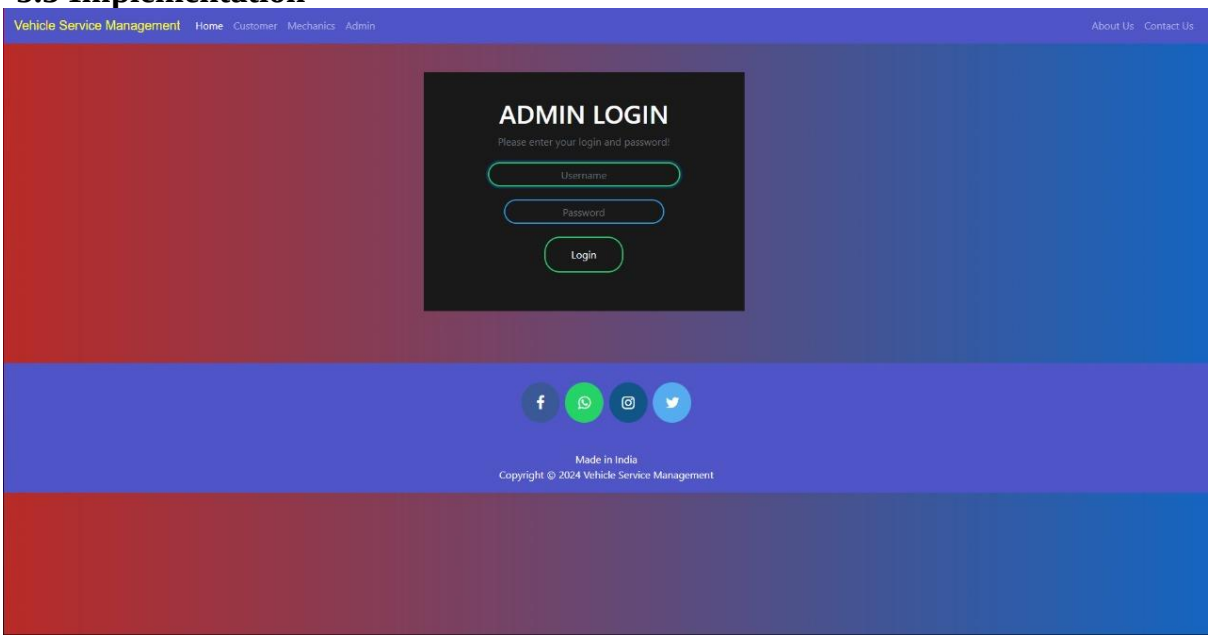
5.2 Description of block diagram

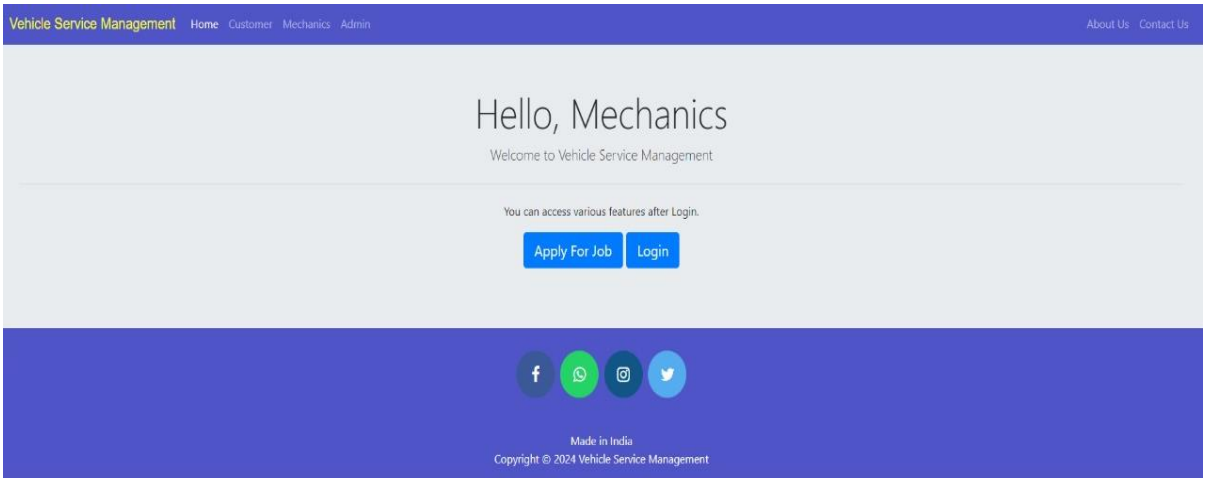
Here's a breakdown of the process:

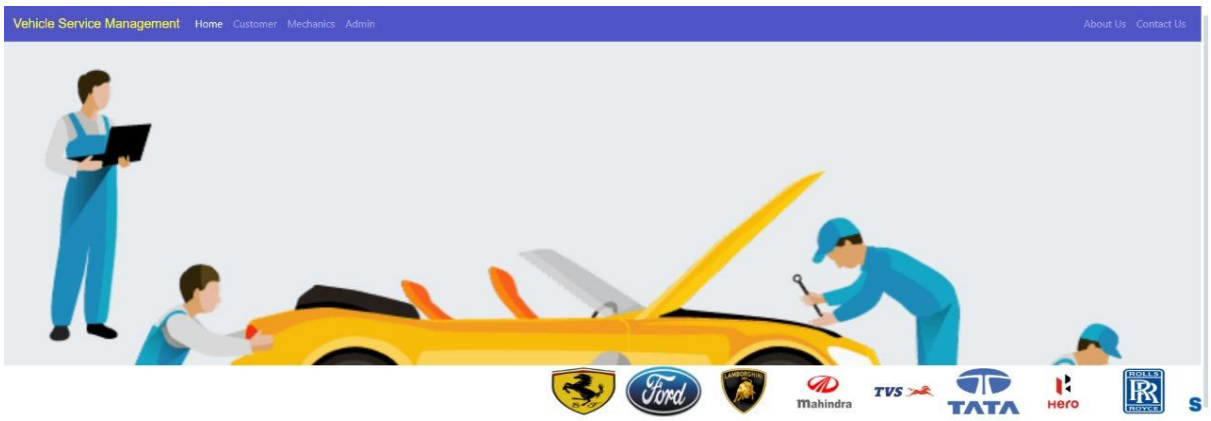
- The process starts with a new request made by the customer.
- The customer then provides details about themself and their vehicle, including their name, contact information, vehicle make, model, category, and a description of the problem.
- This information is likely entered into a customer relationship management (CRM) system by the customer themself or a customer service representative.
- Once the request is logged, it goes to the total enquiries section. Here, the request might be reviewed and assigned to a mechanic, placed in a queue, or categorized based on urgency or service type.

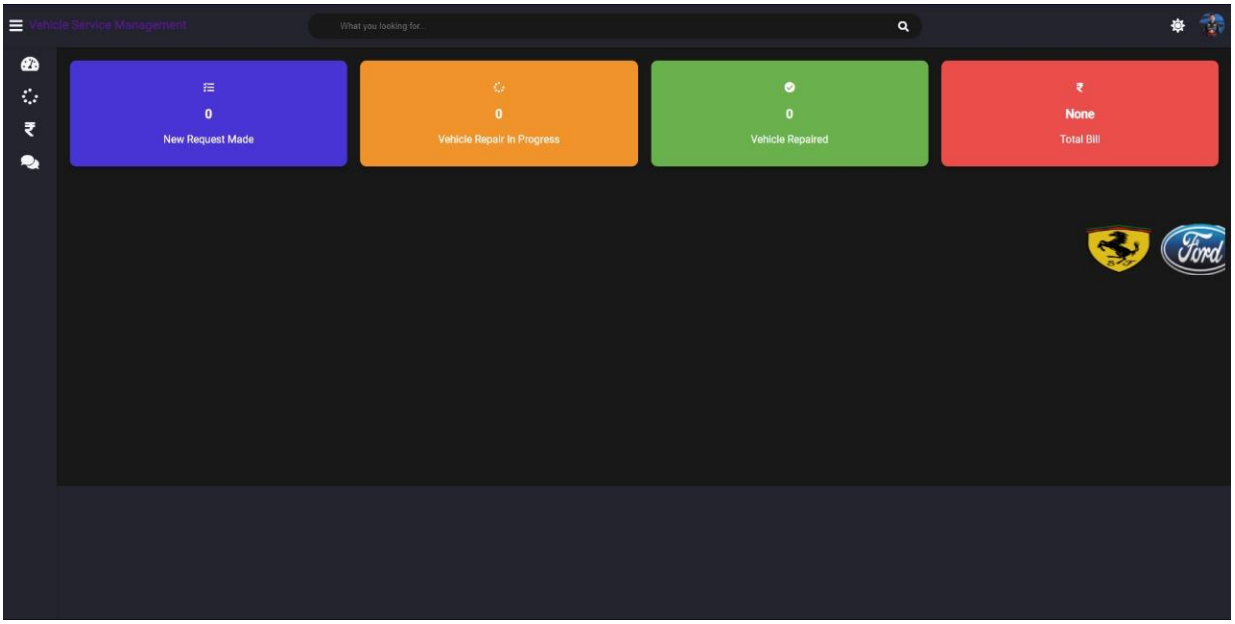
- The mechanic would then move the request to the work in progress section. This likely indicates the mechanic has begun diagnosing the problem.
- Once a mechanic diagnoses the problem, they can provide the customer with an estimate for the repair. The block diagram doesn't show how this would be communicated to the customer but it likely involves the customer service representative or an online portal.
- The customer would then approve or reject the repairs.
- If the repairs are approved, the mechanic would move the request to the vehicle repair in progress section.
- Once the mechanic completes the repair, they would move the request to the vehicle repair completed section.
- The system would likely generate a bill which would be reviewed by the customer.
- The customer would then pay the bill and their vehicle would be returned to them.
- The block diagram also shows a section for total feedback. This suggests that the system allows customers to provide feedback about their repair experience.

5.3 Implementation




Vehicle Service Management
Home
Customer
Mechanics
Admin
About Us
Contact Us
Hello, Mechanics
Welcome to Vehicle Service Management
You can access various features after Login.
Apply For Job
Login
Made in India
Copyright © 2024 Vehicle Service Management


Vehicle Service Management
Home
Customer
Mechanics
Admin
About Us
Contact Us
Mahindra
TVS
TATA
Hero


Vehicle Service Management
What you looking for...
0
New Request Made
0
Vehicle Repair In Progress
0
Vehicle Repaired
None
Total Bill

CHAPTER 6

Conclusion

6. CONCLUSION

The development of the vehicle service management system using Django provides a comprehensive solution for efficiently managing various aspects of vehicle maintenance and service operations. By leveraging the Django framework, we have created a robust and scalable platform that streamlines processes such as scheduling service appointments, tracking vehicle maintenance history, managing inventory, and generating reports. This system enhances the overall efficiency and effectiveness of vehicle service centers, ultimately leading to improved customer satisfaction and business profitability.

Future Scope:

1. Enhanced User Experience: Implementing a more intuitive and user-friendly interface can enhance the overall user experience. Introducing features like drag-and-drop scheduling, interactive dashboards, and mobile-friendly interfaces can further improve usability.

2. Integration with IoT Devices: Integrating Internet of Things (IoT) devices such as vehicle diagnostic tools and sensors can enable real-time monitoring of vehicle health and performance. This integration can provide proactive maintenance alerts and predictive analytics, helping to prevent breakdowns and optimize maintenance schedules.

3. Advanced Reporting and Analytics: Enhance reporting capabilities by integrating advanced analytics tools. By analyzing service data trends, customer feedback, and operational metrics, the system can provide valuable insights for optimizing service offerings, resource allocation, and business strategies.

4. Integration with Payment Gateways: Implementing integration with payment gateways allows for seamless online payment processing for service bookings and invoices. This enhances convenience for customers and streamlines financial transactions for the service center.

REFERENCES:

- Prasad, R.H.C., Mounika, M.J. and Gopi, S., System for Managing Vehicle Services.
- Kumar, S. and Hooda, D., 2023. Customer-Vehicle Management System.
- Gomathy, C.K., Chandrasekhar, M.P., Mallikarjun, K. and Geetha, V., 2022. The vehicle service management system. *International Journal of Early Childhood Special Education (INTJECSE) Doi*, *10*.
- Prasad, Rayapati Hari Chandra, Mrs J. Mounika, and Sangala Gopi. "System for Managing Vehicle Services."
- Kumar, Sumit, and Diksha Hooda. "Customer-Vehicle Management System." (2023).
- Gomathy, C. K., M. Pedda Chandrasekhar, K. Mallikarjun, and V. Geetha. "The vehicle service management system." *International Journal of Early Childhood Special Education (INTJECSE) Doi* 10 (2022).