

GARAGE - CAR SERVICE MANAGEMENT SYSTEM

PROJECT REPORT SUBMITTED IN PARTIAL FULFILMENT OF THE
REQUIREMENTS FOR THE AWARD OF

BACHELOR OF COMPUTER APPLICATIONS

To

MARIAN COLLEGE KUTTIKKANAM AUTONOMOUS

Affiliated to

MAHATMA GANDHI UNIVERSITY, KOTTAYAM

By

NAVEEN E SAJI

(Reg.No:23UBC243)

VIVEK BENNY

(Reg.No:23UBC259)

Guided By

Ms. ROSHNI RAJENDRAN



**DEPARTMENT OF COMPUTER APPLICATIONS
MARIAN COLLEGE KUTTIKKANAM AUTONOMOUS**

PEERMADE – 685531

OCTOBER, 2025

DECLARATION

We, **Naveen E saji [Reg.no 23UBC243]**, **Vivek Benny [Reg.no 23UBC259]** certify that the Mini project report entitled "**Garage-Car service management system**" is an authentic work carried out by us at **Marian College Kuttikkanam Autonomous**. The matter embodied in this project work has not been submitted elsewhere for the award of any degree or diploma to the best of our knowledge and belief.

Signature of the Students:

Naveen E Saji

Vivek Benny

Date:

BONAFIDE CERTIFICATE

This is to certify that this project work entitled “**Garage- Car Service Management System**” is a bonafide record of work done by **Mr. Naveen E Saji [Reg.no 23UBC243]**, **Mr. Vivek Benny [Reg.no 23UBC259]** at **Marian College Kuttikkanam Autonomous** in partial fulfilment for the award of the **Degree of Bachelor of Computer Applications of Mahatma Gandhi University, Kottayam.**

This work has not been submitted elsewhere for the award of any degree to the best of our knowledge.

Head of the Department

Internal Guide

Dr Rajimol A

Ms. Roshni Rajendran

Dept. of Computer Applications

Dept. of Computer Applications

Marian College Kuttikkanam

Marian College Kuttikkanam

Peermade – 685531

Peermade - 685531

Submitted for the Viva-Voce Examination held on

DEPARTMENT SEAL

EXTERNAL EXAMINER

ACKNOWLEDGEMENT

“Gratitude is a feeling which is more eloquent than words, more silent than silence.” In undertaking this project, we needed the direction, assistance and cooperation of various individuals and organizations, which is received in abundance with grace of God, without their unconstrained support, the project could not have been completed. If words are considered as the symbol of approval and token of acknowledgement, then let the following words play the heralding role of expressing our gratitude.

We wish to acknowledge our sincere gratitude to our Manager, **Rev.Fr. Boby Alex Mannamplackal** and **Prof Dr Ajimon George**, Principal **Marian College Kuttikkanam Autonomous**, for all their efforts and administration in educating us in their premier institution.

We extend our gratitude to **Dr Rajimol A**, Head of the Department of Computer Applications, who is a constant source of inspiration and whose advice helped us to complete this project successfully. We express our deep sense of gratitude to our internal project guide, Ms Roshni Rajendran for her profound guidance for the successful completion of this project. With great enthusiasm we express our gratitude to all the faculty members of Department of Computer Applications for their timely help and support. Finally, we express our deep appreciation to all our friends and family members for the moral support and encouragement they have given to complete this project successfully.

Naveen E Saji, Vivek Benny

ABSTRACT

ABSTRACT

The project named “**Garage – Car Service Management System**”, is a web-based platform for the management of car service operations. In today’s digital era, traditional paper-based or manual management of vehicle maintenance is inefficient and prone to errors. This web-based platform overcomes those challenges by allowing users to book and track services online, manage vehicle details, and communicate directly with service staff. By providing a centralized system for users, staff, and administrators, it ensures that each stakeholder has access to the right tools and information. This improves transparency, reduces waiting times, and enhances overall service quality, making the experience more convenient for customers and more manageable for service providers.

From a technical perspective, the use of HTML, CSS, and JavaScript for the front end creates an intuitive, responsive, and user-friendly interface, ensuring smooth user interactions. The PHP and MySQL back end provides the system with a secure and efficient way to store, retrieve, and process data such as bookings, service records, and staff schedules. This combination of technologies enables seamless communication between the client and server, allowing real-time updates, user authentication, and data-driven decisions. Overall, the chosen architecture justifies the project’s objective of delivering a reliable, scalable, and user-centric solution for managing car service operations efficiently.

TABLE OF CONTENTS

TITLE	PAGE NUMBER
1. INTRODUCTION	10
1.1 ABOUT THE PROJECT	10
1.2 EXISTING SYSTEM	10
1.3 PROPOSED SYSTEM	11
2. SYSTEM ANALYSIS	13
2.1 PROBLEM DEFINITION	13
2.2 FEASIBILITY ANALYSIS	13
2.3 RECOMMENDED IMPLEMENTATIONS	14
3. SOFTWARE REQUIREMENT SPECIFICATION	17
3.1 PURPOSE	17
3.2 SCOPE	17
3.3 TECHNICAL OVERVIEW	18
3.4 FUNCTIONAL REQUIREMENTS	18
3.5 NON-FUNCTIONAL REQUIREMENTS	19
3.6 STATED REQUIREMENTS	20
3.7 EXTERNAL INTERFACE REQUIREMENTS	21
4. SYSTEM DESIGN	24
4.1 DESIGN METHODOLOGY	25
4.2 SYSTEM ARCHITECTURE AND PROCESS FLOW	27
4.3 MODULE DETAILS	42
4.4 PERFORMANCE CONSIDERATIONS	44
4.5 SECURITY CONSIDERATIONS	44
4.6 TABLE DESIGN	44
5. CODING	53
5.1 SELECTION OF SOFTWARE	53
5. CODING PHASE	53

6. TESTING	56
6.1 TESTING	56
6.2 TESTING METHODS	57
6.3 IMPLEMENTATION	60
7. MAINTENANCE AND ENHANCEMENT	62
7.1 MAINTENANCE	62
7.2 ENHANCEMENT	62
8. CONCLUSION	65

INTRODUCTION

1. INTRODUCTION

1.1 ABOUT THE PROJECT

The Car Service Management System is a digital platform designed to streamline automotive service operations and enhance customer experience. It provides a comprehensive suite of features for customers to book services, track vehicle maintenance, and communicate with service staff. The website's user-friendly interface ensures smooth navigation experience, allowing users to easily view service details, schedule appointments, and manage their vehicle information with automotive professionals for support and maintenance.

1.1.1 THE PURPOSE AND SCOPE

The primary purpose of the Car Service Management System Project is to establish an efficient and convenient online platform for automotive service operations. The project aims to provide a centralized hub where customers can book services, track vehicle maintenance, communicate with mechanics and manage their automotive needs seamlessly. The scope of the project includes enabling customers to book various automotive services online. Empowering users to manage vehicle information and service history with detailed records. Facilitating customer-staff communication through integrated messaging system. Implementing admin functionalities to manage services, staff scheduling, mechanical assignments, and customer support.

1.2 EXISTING SYSTEM

In the current scenario, accessing automotive service solutions often involves fragmented and time-consuming manual processes. Customers typically rely on phone calls, physical visits, and limited-service tracking methods. This manual approach can lead to inefficiencies, scheduling conflicts, delays, and limited visibility into service progress and vehicle maintenance history.

1.3 PROPOSED SYSTEM

The main objective of the proposed Celestia Interior & Exterior Design Website is to eliminate the limitations of the existing manual system and make it more user-friendly. Most of the limitations of the current system can be overcome by the proposed system. Speed and accuracy are the main advantages of the proposed system, with no redundancy of data. The proposed software requires fewer resources to manage the entire design platform. Since all details are stored

digitally, search time is reduced. Information can be more secure because computer systems provide better data security. The proposed system eliminates many drawbacks of the existing system and enhances data security, user experience, and operational efficiency.

SYSTEM ANALYSIS

2.SYSTEM ANALYSIS

2.1 PROBLEM DEFINITION

The current system does not effectively utilize digital solutions for automotive service management needs and lacks refinement. Even when records are kept digitally, they are often managed inefficiently and are not integrated with modern web technologies. This makes it difficult to handle various aspects like customer interactions, service scheduling, mechanic assignments, and inventory management efficiently. Much of the record-keeping and appointment scheduling are still done manually, leading to errors, double bookings, and delays. To overcome these limitations, the Car Service Management System is proposed.

2.1.1 ADVANTAGES OF PROPOSED SYSTEM

2.2 FEASIBILITY ANALYSIS

Feasibility study is a test of a system proposal according to its workability, ability to meet user needs and effective use of resources. The objective of feasibility is not to solve the problem but to acquire a sense of its scope. The main aim of the feasibility study is to test the technical, social and economic feasibility of the system. The feasibility study can be classified into the following categories:

Operational Feasibility, Technical Feasibility, Economic Feasibility

2.2.1 OPERATIONAL FEASIBILITY

The proposed system offers user friendliness combined with greater processing speed for automotive service operations. Therefore, the work done can be reduced significantly. Since the processing speed is very high compared with manual system a lot of time can be saved in services scheduling and management. The workload is reduced, and this system requires only small amount of work from admin who manages the whole system, along with staff members who can efficiently manage their schedules and customer interactions. Hence, this project is operationally feasible.

2.2.2 TECHNICAL FEASIBILITY

Technical feasibility deals with hardware as well as software requirements and to what extend it can support the proposed system. The hardware required is a computer system with internet connectivity and software includes PHP, MySQL, and web technologies. If the necessary requirements are made available with the system, then the proposed system is said to be technically feasible.

2.2.3 ECONOMIC FEASIBILITY

Economic feasibility is an important factor. Since the existing system is manual, the feasibility for wrong data entry is higher and consumes a lot of time and can occur errors in service scheduling and customer management. But the proposed system aims at processing of automotive service information efficiently, thus saving time and reducing operational costs. The new system needs only a computer system with internet connectivity which is already available therefore the cost is negligible. Proposed system uses validation checks so there are minimal errors. Even though an initial investment must be made on the software and the hardware aspects, the proposed system aims at processing of service information efficiently. Thus, the benefits acquired out of the system are sufficient for the project to be undertaken. So, the proposed system is economically feasible.

2.3 RECOMMENDED IMPLEMENTATIONS

Two principal sources of data are:

Written documents, Data from persons who are involved in the operation of the automotive service system under study. The different fact-finding techniques are Questionnaires, Personal Interviews, Observations

Questionnaires

Questionnaires are best methods to probe data out of the customers and service staff. In this case, questionnaires were used to collect feedback from existing customers about their service experience and expectations from an automated system.

Personal Interviews

Personal interviews are the best way to gather facts. This was the primary source of fact finding used for

this project. The service centre owner, mechanics, and staff were interviewed and data collected. They were asked how the service operations and record keeping happened under the existing system. And suggestions were taken on what they wanted to add in this automated system. Almost all their suggestions were integrated into this project.

Observations

A person can understand a lot about a system just by observing it. By being a bystander and observing how a day passes in the automotive service centre helped to kick off this project. Using this method resulted in a better understanding of the workings of the service operations and what to do to make this a comprehensive web application. Observing the current system, one can understand that a lot of paperwork and manual coordination is involved in the service scheduling and customer management area. The way in which the records are kept gives an idea for a strong database model.

SOFTWARE REQUIREMENT SPECIFICATIONS

3 SOFTWARE REQUIREMENT SPECIFICATION

Requirements specification is the starting step for the development activities. It is currently one of the weak areas of software engineering. During requirement specification, the goal is to produce a document of the client's requirements. This document forms the basis of development and software validation. The basic reason for the difficulty in software requirements specification comes from the fact that there are three interested parties- the client, the end users and the software developer.

3.1 PURPOSE

The origin of most software systems needs a client, who either wants to automate an existing manual system or desires a new software system. The software system itself was created by the developer. Finally, the completed system will be used by the end users. Thus, there are three major parties interested in a new system: the client, the users and the developer. A basic purpose of software requirements specification is to bridge the communication gap. SRS is the medium through which the client and user needs are accurately specified. Indeed, SRS forms the basis of software development. A good SRS should satisfy all the parties, something very hard to achieve, and involves trade-offs and persuasion. Another important purpose of developing an SRS is helping the clients understand their own needs. Advantages are:

An SRS establishes the basis for agreement between the client and the supplier on what the software product will do An SRS provides a reference for validation of the final product a high-quality SRS is a prerequisite to high-quality software. A high-quality SRS reduces the development cost.

3.2 SCOPE

3.2.1 SYSTEM STATEMENT OF SCOPE

The Car Service Management System was developed to provide a platform for managing automotive service operations, scheduling appointments, and tracking vehicle maintenance. The system further offers features such as service booking, staff management, mechanic assignment, and customer communication. Reports can be generated to retrieve operational data.

3.3 TECHNICAL OVERVIEW

3.3.1 USER CHARACTERISTICS

The system can be accessed by three types of users: Admin, Staff, and Customers. Admins have access to the complete admin dashboard, staff have access to their work management interface, while customers have access to the booking system and service tracking. The admin and staff functionalities are invisible to customers.

3.4 FUNCTIONAL REQUIREMENTS

The functional requirements of this website are as follows:

Customers provide essential details like name, email, password, and phone number. Users log in by entering their registered email and password. The system verifies credentials against stored data to allow access. Users can securely log out, ending their active sessions to prevent unauthorized access. Customers can add and manage their vehicle information including type, brand, model, registration number, and year. Multiple vehicles can be registered under one customer account. Customers can browse available automotive services with detailed descriptions and pricing. Customers can book services by selecting preferred date, time slot, and required services. Customers can schedule service appointments with preferred mechanics. Admins can manage and oversee service bookings, ensuring efficient scheduling. Mechanics can be assigned to specific bookings based on their expertise and availability. Staff members can apply for leave by specifying dates and reasons. Admins can approve or reject leave applications. Staff schedules can be managed efficiently. Admins can add new

mechanics with details like name, age, profession, and contact information. Mechanic availability and assignment status can be tracked. Mechanics can be assigned to specific service types based on their expertise. Customers can send messages to staff for inquiries or support. Staff can respond to customer messages through the system. Message history is maintained for reference. Admins can oversee all bookings and service operations. Admins can manage services, staff, mechanics, and customer accounts. Comprehensive reporting and analytics for business insights.

3.5 NON-FUNCTIONAL REQUIREMENTS

The non-functional requirements of this website are as follows:

Usability: Intuitive and user-friendly interface for all user types. Comprehensive service booking features and clear navigation.

Reliability: System performs reliably in 95% of use cases monthly. Ensure consistent and accurate data processing for service operations.

Availability: Accessible globally 24/7 for customer convenience. Backup and recovery mechanisms to restore service within an hour of failure.

Security: Hourly backups of the database. Secure handling of customer and vehicle data to protect privacy and prevent unauthorized access.

Performance: Quick response times and minimal latency for booking operations. Efficient handling of large volumes of service data and customer information.

Scalability: Capable of supporting a growing number of customers, vehicles, and service bookings. Modular architecture to facilitate easy expansion of services.

Maintainability: Codebase should be well-documented and modular for ease of updates and bug fixes. Regular maintenance schedules to ensure system health and performance.

3.6 STATED REQUIREMENTS

3.6.1 GENERAL REQUIREMENTS

The system has 8 functional modules divided between admin, staff, and customers.

Login Only registered users (customers, staff, and admins) can log in to the system to avail the services. The registered users use their email and password to log in. The user email should always be a valid email address. The password can contain both uppercase and lowercase alphabetic characters, numbers, and special characters. Admins will be redirected to the admin panel when logging in with admin credentials. Staff will be directed to the staff dashboard when logging in with staff credentials. Customers will be directed to the main booking service. page when login is successful.

Sign Up New customers need to register to use the platform's services. This includes several fields: Name, Email Address, Password, Phone Number. Admin Panel the admin panel contains different admin processes such as: Managing service bookings and assignments Managing staff and mechanics Managing services and pricing Viewing reports and analytics. Staff Dashboard the staff dashboard allows staff members to: View assigned work schedules, apply for leave, Communicate with customers, Update service status. Customer service booking customers can browse and book various automotive services. Service details include descriptions, pricing, and estimated time. Vehicle Management customers can add multiple vehicles to their profile. Vehicle information includes type, brand, model, registration number, and year. Mechanical Assignment admins can assign mechanics to specific bookings based on expertise. Mechanical availability and status can be tracked. Communication System customers can send messages to staff for support. Staff can respond to customer inquiries.

3.6.2 INPUTS

The Car Service platform will take various inputs to provide its services:

Customer Registration: Name, Email Address, Password, Phone Number.

Login: Email Address, Password.

Vehicle Information: Vehicle Type (SUV, Sedan, Hatchback etc.), Brand, Model, Registration Number, Year.

Service Booking: Selected Services, Preferred Appointment Date, Time Slot, Vehicle Selection.

Staff Leave Application: Leave Reason, From Date, To Date, **Mechanic Information:** Name, Age, Profession, Contact Details Specialization

3.6.3 PROCESSING

The system will perform the following key processing tasks:

Validate email format and password complexity. Ensure required fields are not empty. Validate vehicle registration number format. Customers book services and manage appointments. Admins oversee bookings and assign mechanics. Service status tracking and updates. Leave application processing and approval. Mechanic assignment based on availability and expertise. Work schedule management. Message routing between customers and staff. Response tracking and history maintenance.

3.6.4 OUTPUTS

The system produces the following outputs:

Service Name, Description, Price. Estimated Time and Requirements. Booking details with assigned mechanic. Appointment date and time Confirmation. Complete vehicle information and service Leave status, work assignments Communication. Service analytics and business reports. Customer and revenue insights.

3.7 EXTERNAL INTERFACE REQUIREMENTS

3.7.1 USER INTERFACES

All user interfaces will be GUI interfaces, designed for ease of use and high functionality. The

interfaces will have a pleasing appearance and intuitive design.

The interface will use suitable design elements and pleasing colours to create a comfortable and attractive environment for users. Consistent design themes will be maintained across all pages. Usability Textboxes, combo boxes, and buttons will be used to facilitate easy data entry. Clear labels and instructions will be provided for all input fields. Intuitive navigation elements will help users move seamlessly through the platform. The user interface will be responsive to ensure usability across various devices, including desktops, tablets, and smartphones.

3.7.2 HARDWARE INTERFACES

The system needs a computer or any other smart devices with network availability to access the web application. No other external hardware is required.

Hardware Specification

Processor: Intel Pentium or higher

RAM: 256 MB or higher

Hard Disk Drive: 100 MB required on disk

Keyboard: Standard QWERTY keyboard

Implementation Specification

Operating System: Windows OS

3.7.3 SOFTWARE INTERFACES

Software Specification

- **Operating System:** Windows 11
- **DBMS:** MySQL
- **Tool Used:** PHP

SYSTEM DESIGN

4. SYSTEM DESIGN

The design phase aims to develop a solution for the problems identified during the analysis phase. This phase marks the transition from understanding the problem to creating a solution. System design details the enquired features and operations, including screen layouts, business rules, process diagrams, pseudocode, and other relevant documentation. During this phase, the overall structure and specifics of the software are defined. This includes determining the number of tiers needed for the architecture, designing inputs and outputs, and establishing database and data structure designs. Proper analysis and design are critical to the development cycle since errors in this phase can be costly to fix later. Therefore, careful attention is given during the design phase. The logical framework of the product and its physical attributes are outlined during this stage. The operating environment is set up, and key resources are identified. Any element that requires user input or approval must be documented and reviewed by the user. The physical aspects of the system are specified, and a detailed design is prepared. Subsystems identified during design are used to create a detailed system structure. Each subsystem is divided into one or more design units or modules, and detailed logic specifications are created for each module. The module logic is typically described in a high-level design language, which is independent of the final implementation language.

A good design should consider:

The design should be straightforward and clear, guiding users to their desired outcomes intuitively. Research shows that users can retain about six words in their short-term memory. The number of choices presented to users should ideally be four or fewer to prevent confusion and forgetfulness. Users dislike going through many steps to access a service. More than five steps can cause impatience, so minimizing the number of steps helps reduce frustration. Users should easily navigate back and forth between different steps, allowing them to access various parts of the dialog seamlessly. Avoid choices with similar pronunciations to reduce confusion and ensure users can clearly distinguish between options. Implementing graceful error handling helps decrease dependency on operators by managing mistakes effectively. Keep users informed about the ongoing process to maintain their engagement and understanding. For the general design, one or more potential designs are proposed and broadly sketched. These alternatives are then presented to the users, who choose the design that best suits their requirements while staying within project constraints. The detailed design stage specifies the user interface, database, programs, hardware, and training and system documentation. Several structured techniques are used during the design phase. To design the software components, the designer transforms the automated processes in the physical data flow diagram into a program structure chart, which decomposes software

processes into detailed modules and shows control paths between modules.

4.1 DESIGN METHODOLOGY

4.1.1 INPUT DESIGN

Input design focuses on converting user-oriented inputs into a format recognizable by the computer. Collecting input data is one of the costliest parts of the system in terms of equipment, time, and user involvement. The goal of input design is to make data entry as simple, logical, and error-free as possible. Input design serves as the bridge between the information system and its users, transforming transaction data into a form suitable for processing. This process can involve reading data from printed documents or directly keying data into the system. Effective input design controls the amount of input required, minimizes errors, avoids delays and extra steps, and keeps the process straightforward. System analysis determines the following input design details: What data to input, what medium to use, How the data is arranged and coded, Data items and transactions requiring validation to detect errors. Activities involved in input design include Data Recording Collecting data for input. Data Verification Ensuring the accuracy of the collected data. Data Conversion transforming data into the required format. Data Validation Checking data for errors. Data Correction: Fixing any identified errors.

4.2.1 OUTPUT DESIGN

Output design involves creating necessary outputs tailored to meet the requirements of various users. It is essential to approach the design of computer outputs in a well-considered manner. Outputs refer to any information generated by the information system, whether in printed or displayed form. Analysts design computer outputs by identifying specific outputs required to fulfil system requirements. Computers serve as crucial sources of information for users. Efficient and thoughtful output design enhances the system's interaction with users and supports decision-making processes. When designing outputs, system analysts must achieve the following objectives:

Determine the information to be presented Decide whether to display, print, or verbally communicate the information, and select the appropriate output medium format the information in an easily understandable manner. Output design is critical to the success of any system as it bridges the gap between the user and the system's operations. Effective output design includes specifications and procedures for presenting data clearly to users. Users should never be left uncertain about system activities, as appropriate error messages and acknowledgment messages are provided.

4.2.2 CODE DESIGN

The coding phase transforms the detailed software design into a programming language. It translates the software's detailed design representation into executable code. Code design aims to minimize the lines of code used while modularizing the implementation. Modules hide complexity by encapsulating executable statements under named functions or procedures. Effective information hiding enhances program understanding at higher abstraction levels. Module names should accurately describe their actions to avoid confusion. In this software, a modularized approach is employed with different functions created for various operations, each named to reflect its action.

4.2.3 DATABASE DESIGN

Database design identifies relevant data relationships and defines tables using standard methods. Each table's attributes are carefully defined to optimize database performance, ensure data integrity, minimize redundancy, and enhance security. A database system is a computer representation of an information system designed to handle integrated data efficiently. It minimizes redundancy to provide quick, flexible, and cost-effective information access. Database design considers several specific objectives:

Controlled redundancy, User-friendly interface, Data independence, Cost-effective data retrieval, Accuracy and integrity, Failure recovery, Privacy and security, Performance optimization Database design involves creating multiple views of data, including logical and physical views. The logical view represents data independently of its storage, focusing on how users and programmers interact with it. The physical view describes how data are stored and accessed in physical storage. Each table in the database typically includes a primary key, a unique column (or combination of columns) that uniquely identifies each record. Primary keys ensure data integrity by enforcing uniqueness and cannot contain null values. Normalization is employed to organize database data, minimizing redundancy and anomalies during data insertion, update, and deletion operations.

4.2 SYSTEM ARCHITECTURE AND PROCESS FLOW

UML DIAGRAMS

4.2.1 USE CASES

USER REGISTRATION

Use Case Id:	CL_UC_01
Use Case Name:	Registration
Created by:	Naveen & Vivek
Date Created:	20-07-2025
Description:	Allows new customers to register for an account on the car service platform.
Primary actor:	Customer
Secondary actor:	None
Precondition:	Customer navigates to the registration page.
Postcondition:	Customer accounts are successfully created.
Main flow:	<ol style="list-style-type: none">1. Customer navigates to the registration page2. Customer provides required details: name, email, phone number, and password.3. Customer submits the registration form.4. System validates the information and creates the customer account. <p>Use case ends.</p>

LOGIN

Use Case Id:	CL_UC_02
Use Case Name:	Login
Created by:	Naveen & Vivek
Date Created:	24-06-2025
Description:	<p>This use case enables users to access the system by entering their credentials. There are three user roles: admin, staff, and customer. Each role logs in using their respective email and password. Once logged in, customers can book services, staff can manage their work, and admins can manage the entire system.</p>
Primary actor:	Customer/Staff/Admin
Secondary actor:	None
Precondition:	The user should have a valid account.
Postcondition:	The system displays relevant Homepage based on user role
Main flow:	<ol style="list-style-type: none"> 1. The user goes to the login page. 2. The user inputs their registered email and password. 3. The user submits the login form. 4. The system checks the user's credentials. 5. If the credentials are correct, the system grants the user access. 6. The use case concludes.

VIEW SERVICES

Use Case Id:	CL_UC_03
Use Case Name:	View Services
Created by:	Naveen & Vivek
Date Created:	24-06-2025
Description:	<p>This use case allows users to view details of automotive services available on the platform. Customers can browse services with descriptions, prices, and estimated time. Admins can see options to edit service details.</p>
Primary actor:	Customer/Admin
Secondary actor:	None
Precondition:	The user should be logged into their account.
Postcondition:	The system displays the detailed information of the selected service.
Main flow:	<ol style="list-style-type: none">1. The user navigates to the services page.2. The user selects a service they wish to View.3. The system retrieves the service details.4. The system displays the service details, including description, price, and estimated time. details.5. If the user is an admin, they also see options to edit the service details. <p>The use case concludes.</p>

BOOK SERVICE

Use Case Id:	CL_UC_04
Use Case Name:	Book Service
Created by:	Naveen & Vivek
Date Created:	24-06-2025
Description:	This use case allows customers to book automotive services from the platform. Customers select services, choose appointment time, and complete the booking process.
Primary actor:	Customer
Secondary actor:	None
Precondition:	The customer must be logged into their account and have registered vehicles.
Postcondition:	The system confirms the booking and updates the appointment schedule.
Main flow:	<ol style="list-style-type: none"> 1. The customer navigates to the services page and selects the services they wish to book. 2. The customer selects their vehicle from the registered vehicles list. 3. The customer chooses preferred appointment date and time slot. 4. The customer provides any additional requirements or notes. 5. The customer submits the booking request. 6. The system processes the booking and assigns a mechanic if available. 7. The system confirms the booking and provides a booking confirmation to the customer. <p>The use case concludes.</p>

MANAGE SERVICES

Use Case Id:	CL_UC_05
Use Case Name:	Manage Services
Created by:	Naveen & Vivek
Date Created:	24-06-2025
Description:	<p>This use case allows admins to manage services on the platform. Admins can add new services, edit existing service details, and remove services as needed. This functionality ensures that the service catalog remains up-to-date and accurate.</p>
Primary actor:	Admin
Secondary actor:	None
Precondition:	The admin must be logged into their account with appropriate privileges.
Postcondition:	The service catalog is updated with the new, edited, or removed service details.
Main flow:	<ol style="list-style-type: none"> 1. The admin navigates to the product management page. 2. The admin selects an option to add, edit, or remove a product. <ul style="list-style-type: none"> • To add a Product: <ol style="list-style-type: none"> 1. The admin enters the services details, including name, description, price, and estimated time. 2. The admin submits the new service information. 3. The system updates the product catalog with the new service.

- **To Edit a Service:**
 1. The admin selects the service to be edited from the catalog.
 2. The admin modifies the service details as needed.
 3. The admin submits the updated service information.
 4. The system updates the service catalog with the edited product details.
 - **To Remove a Service:**
 1. The admin selects the service to be removed from the catalog.
 2. The admin confirms the removal.
 3. The system removes the service from the catalog.
 3. The system confirms the action (add, edit, or remove) and updates the service catalog accordingly.
- The use case concludes.

MANAGE VEHICLES

Use Case Id:	CL_UC_07
Use Case Name:	Manage Vehicles
Created by:	Naveen & Vivek
Date Created:	24-06-2025
Description:	This use case allows customers to manage their vehicle information on the platform. Customers can add new vehicles, edit existing vehicle details, and remove vehicles from their account.
Primary actor:	Customer
Secondary actor:	None
Precondition:	The customer must be logged into their account.
Postcondition:	The customer's vehicle information is updated in the system.
Main flow:	<ol style="list-style-type: none"> 1. The customer navigates to the vehicle management section in their profile.. 2. The customer selects an option to add, edit, or remove a vehicle. <p>• To Edit a Vehicle:</p> <p>The customer submits the updated information.</p> <p>• To Remove a Vehicle:</p> <p>The customer selects the vehicle to be removed. The customer confirms the removal. The system removes the vehicle from the customer's account.</p> <p>The use case concludes.</p>

MANAGE BOOKINGS

Use Case Id:	CL_UC_08
Use Case Name:	Manage Bookings
Created by:	Naveen & Vivek
Date Created:	24-06-2025
Description:	This use case allows admins to manage service bookings on the platform. They can review, assign mechanics, update status, or cancel bookings to ensure smooth service delivery.
Primary actor:	Admin
Secondary actor:	None
Precondition:	The admin must be logged into their account with appropriate privileges.
Postcondition:	The booking details are updated as per the actions taken (assigned, status updated or cancelled).

Main flow:	<ul style="list-style-type: none"> i. The admin navigates to the booking management page. ii. The admin reviews the list of upcoming and pending bookings. iii. The admin selects a booking to manage. <p>• To Assign a Mechanic:</p> <p>The admin reviews available mechanics and their expertise.</p> <p>The admin assigns a suitable mechanic to the booking.</p> <p>The system updates the booking with mechanic assignment.</p> <p>To Update Booking Status:</p> <p>The admin selects the appropriate status (Confirmed, In Progress, Completed, etc.).</p> <p>The system updates the booking status and notifies the customer.</p> <p>To Cancel a Booking:</p> <p>The admin selects the booking to be cancelled.</p> <p>The admin confirms the cancellation.</p> <p>The system updates the status to "Cancelled" and sends a cancellation notice to the customer.</p> <p>The system confirms the action and updates the booking records accordingly.</p> <p>The use case concludes.</p>
------------	---

STAFF LEAVE MANAGEMENT

Use Case Id:	CL_UC_04
Use Case Name:	Book Service
Created by:	Naveen & Vivek
Date Created:	24-06-2025
Description:	This use case allows staff members to apply for leave and admins to manage leave applications. Staff can submit leave requests with dates and reasons, while admins can approve or reject them.
Primary actor:	Staff/Admin
Secondary actor:	None
Precondition:	The staff member or admin must be logged into their account.
Postcondition:	Leave application is submitted or processed based on the user role.
Main flow:	<ol style="list-style-type: none"> 1. Staff member navigates to leave application page. 2. Staff member fills in leave reason and dates (from and to). 3. Staff member submits the leave application 4. System records the application and notifies admin. <p>• For Admin:</p> <p>Admin navigates to leave management page. Admin reviews pending leave applications. Admin approves or rejects the application. Admin approves or rejects the application. The staff member.</p> <p>The use case concludes.</p>

LOG OUT

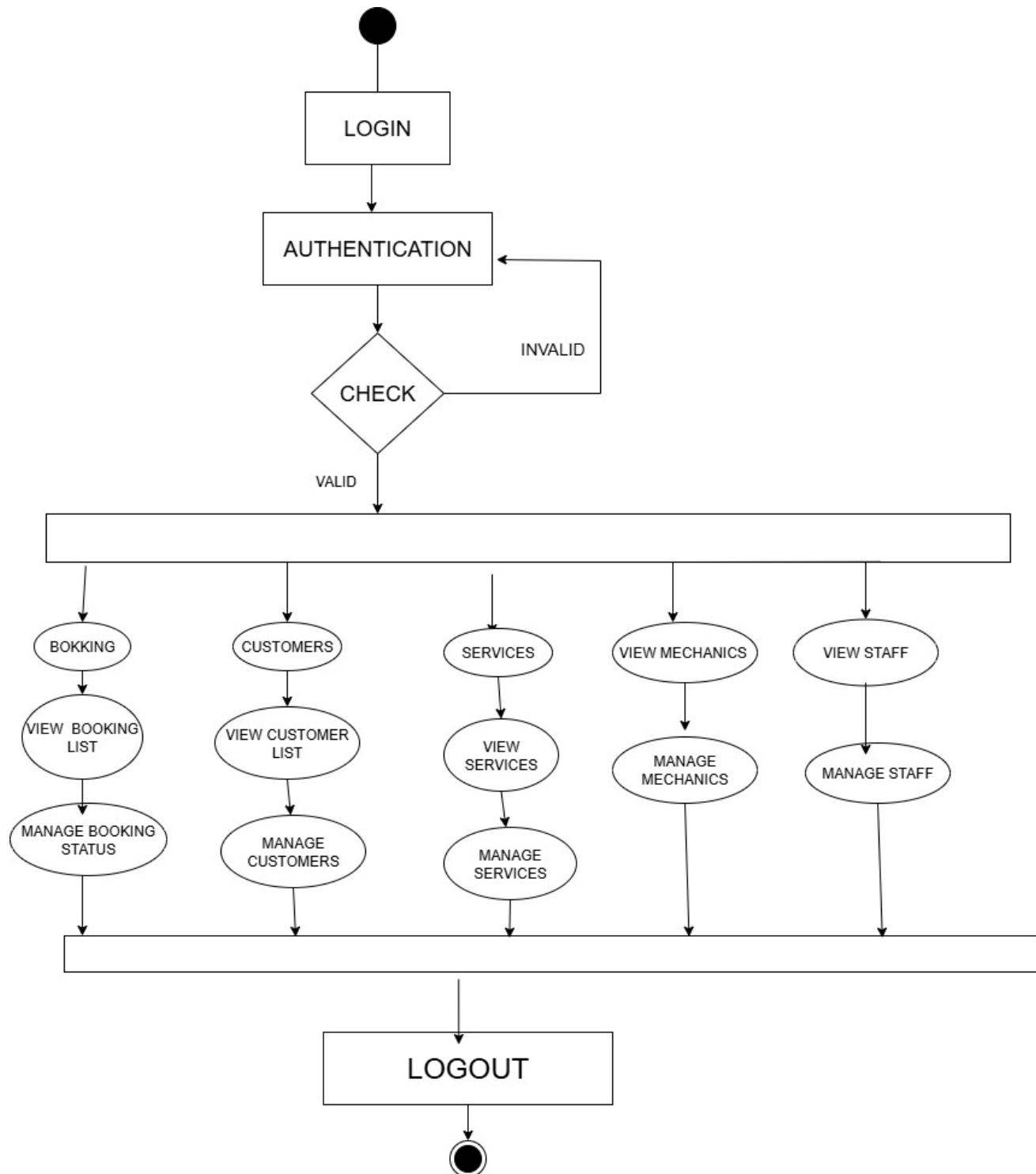
Use Case Id:	CL_UC_10
Use Case Name:	Log Out
Created by:	Naveen & Vivek
Date Created:	24-06-2025
Description:	This use case allows users (customers, staff and admins) to log out of their accounts on the platform. Logging out ensures that the session is securely ended, and access is restricted until the user logs in again.
Primary actor:	Customer/Staff/Admin
Secondary actor:	None
Precondition:	The user must be logged into their account.
Postcondition:	The user session is terminated and redirected to login page.
Main flow:	<ol style="list-style-type: none"> 1. The user navigates to the log out option. 2. The user selects the option to log out 3. The system terminates the user session. 4. The system redirects the user to the login page. 5. The system confirms that the user has been successfully logged out. <p>Use case ends.</p>

4.2.2 USECASE DIAGRAM

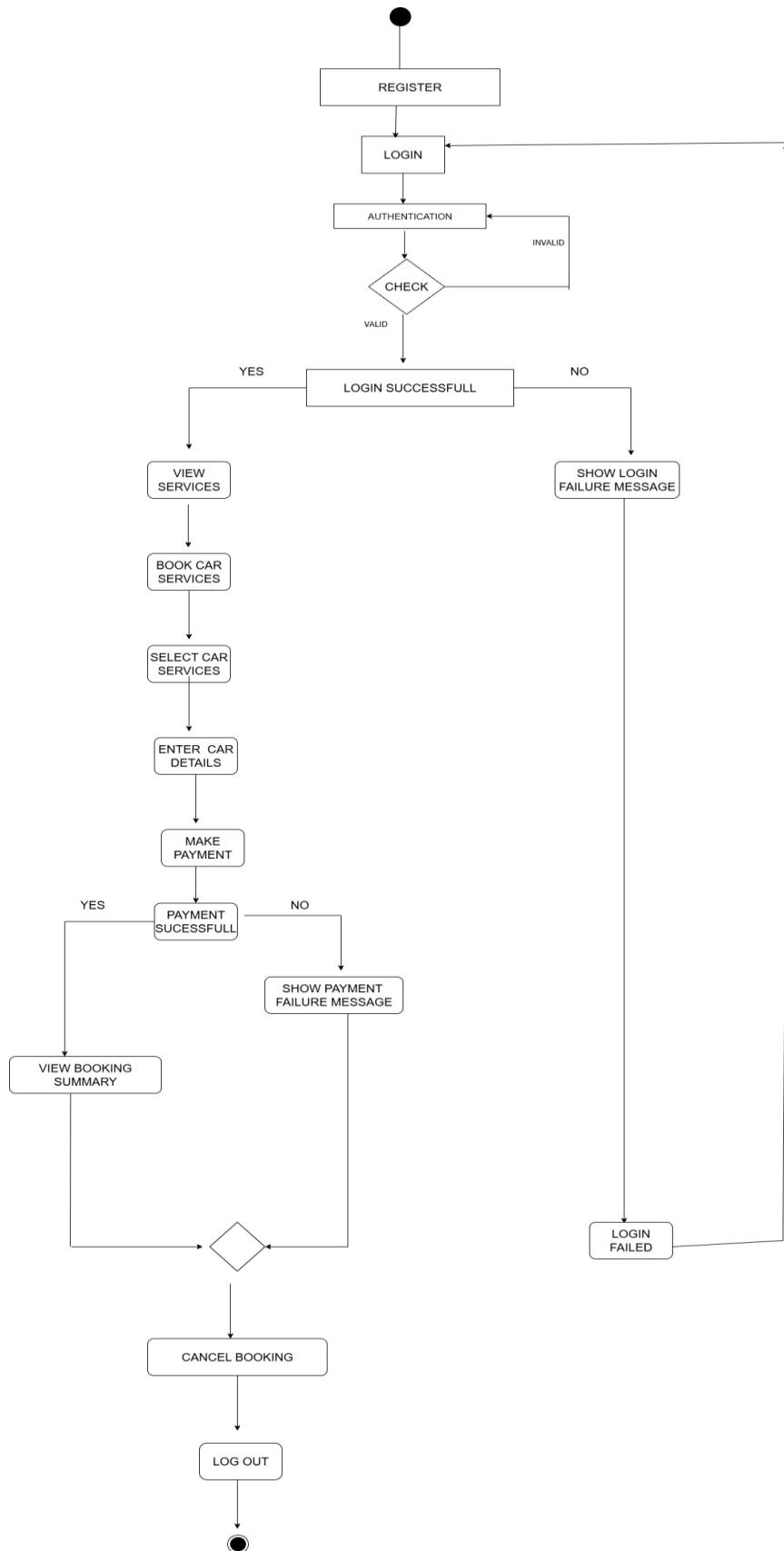


4.2.3 ACTIVITY DIAGRAMS

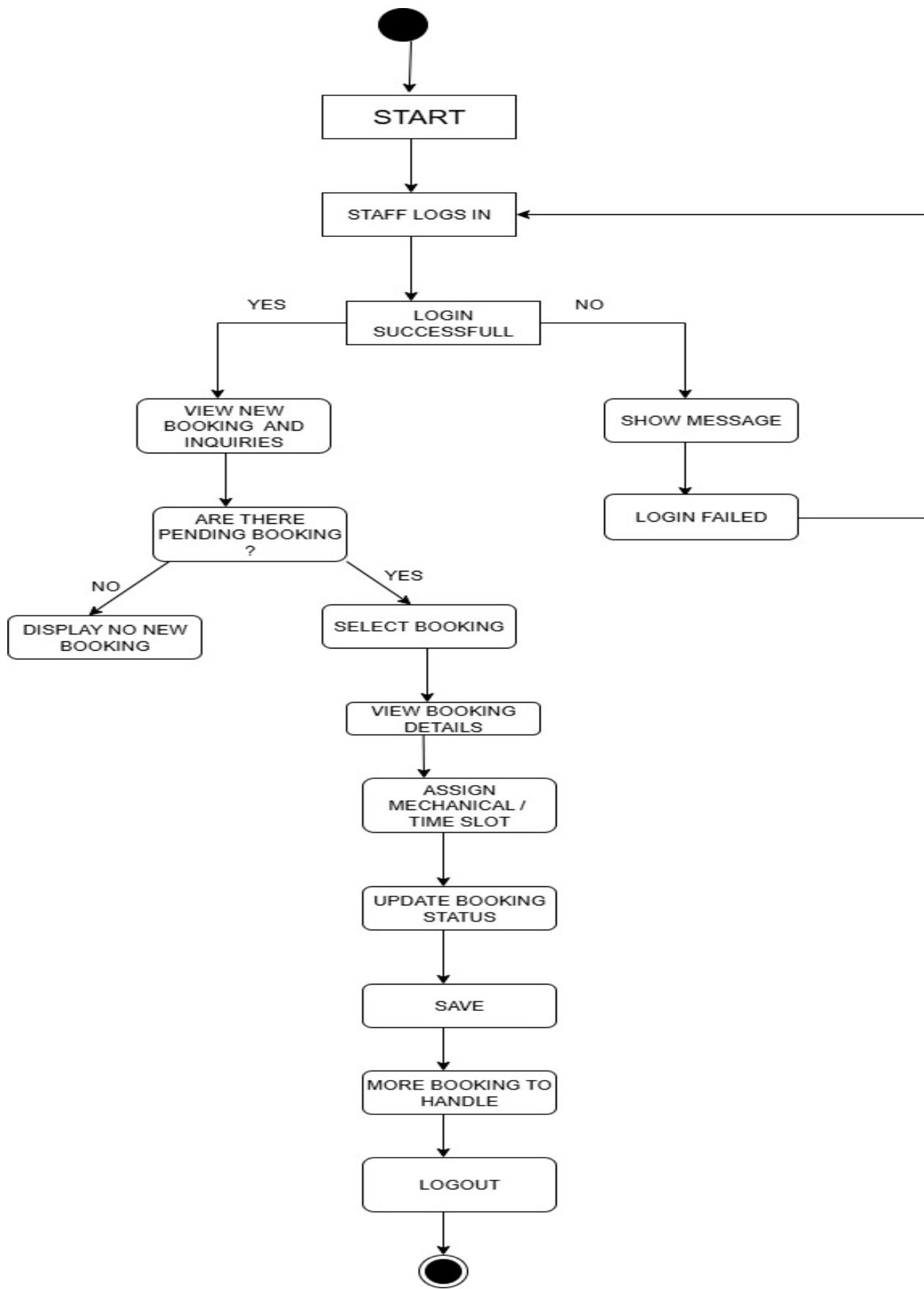
ADMIN SIDE



USER SIDE



STAFF SIDE



4.3 MODULE DETAILS

There are eight main modules in this website:

Login Module, Registration Module, Service Management Module, Booking Management Module, Vehicle Management Module, Staff Management Module, Communication Module
Admin Dashboard Module

1. Login Module

The login module allows registered users (customers, staff, and administrators) to securely access the Car Service Management System. It verifies user credentials (email and password) to grant access to role-specific functionalities such as booking services, managing schedules, and overseeing operations.

2. Registration Module

The registration module enables new customers to create accounts on the platform. Users provide personal details including name, email, phone number, and password to register and access features like vehicle management, service booking, and communication with staff.

3. Service Management Module

The service management module allows customers to browse available automotive services and admins to manage the service catalog. Customers can view detailed service information including descriptions, pricing, and estimated time, while admins can add, edit, or remove services

4. Booking Management Module

The consultation booking module enables users to schedule virtual or in-person consultations with design experts through Celestia and receive confirmation of their consultation bookings.

5. Vehicle Management Module

The vehicle management module allows customers to register and manage their vehicle information including type, brand, model, registration number, and year. Multiple vehicles can be associated with a single customer account for convenient service booking.

6. Staff Management Module

The staff management module provides functionality for staff members to manage their work schedules, apply for leave, and track their assignments. Admins can manage staff accounts, approve leave applications and monitor staff performance.

7. Communication Module

The communication module facilitates messaging between customers and staff members. Customers can send inquiries or support requests, while staff can respond to customer messages and provide assistance throughout the service process.

8. Admin Dashboard Module

The admin dashboard module provides administrators with comprehensive tools to oversee all system operations including booking management, staff coordination, service catalog management, and business analytics. Admins can ensure smooth platform operation and maintain data integrity.

4.3 PERFORMANCE CONSIDERATIONS

Hardware Requirements

The system is designed to perform optimally with a minimum of 4GB RAM and is compatible with Windows OS versions and higher. The web-based nature ensures accessibility across various devices and operating systems.

4.4 SECURITY CONSIDERATIONS

Access Control

Authorized Access: Only users with valid email addresses and passwords are allowed to access the Car Service Management System. **Role-based Security:** The system implements role-based access control ensuring customers, staff, and admins only access appropriate functionalities.

Data Protection: All sensitive information including customer data and vehicle information.

4.5 TABLE DESIGN

1.Table name: register (customers)

Sl.no	Field Name	Data Type	Constraint	Description
1.	uid	int (11)	Primary Key, Auto_increment, Not null	Customer user id
2.	name	varchar (30)	Not null	Name of the customer
3.	email	varchar (50)	Unique Key, Not null	Email of the customer
4.	Password	varchar (255)	Not null	Password of the customer
5.	Phone number	varchar (10)	Not null	Phone number of the customer

6.	Profile picture	varchar (255)	Default null	Profile picture path
7.	Created at timestamp	Default	Default current time stamp	Registration timestamp

2.Table name: Admins

Sl.no	Field Name	Data Type	Constraint	Description
1.	aid	int (11)	Primary key, Auto_increment, Not null	Admin id
2.	Username	varchar (30)	Not null	Admin username
3.	email	varchar (50)	Unique key, not null	Admin email
4.	password	varchar (255)	Not null	Admin password
5.	Role	Enum(admin')	Default 'admin'	Shows whether deleted or not
6.	Profile picture	varchar (255)	Default null	Profile picture path

3.Table name: Staff

Sl.no	Field Name	Data Type	Constraint	Description
1.	Staff_id	int (11)	Primary key, Auto_increment, Not null	Staff id
2.	Staff name	varchar (30)	Not null	Staff member name
3.	email	varchar (50)	Unique key, Not null	Staff email
4,	Password	varchar (255)	Not null	Staff phone number
5.	Phone	varchar (15)	Not null	Staff phone number
6.	Created at timestamp	Default	Current timestamp	Role specification

7.	Role Enum(staff)	Default	'staff'	Role specification
8.	Profile picture	Varchar (255)	Default null	Profile picture path

4.Table name: vehicles

Sl.no	Field Name	Data Type	Constraint	Description
1.	Vehicle_id	int (11)	Primary key, Auto_increment, Not null	Vehicle id
2.	User_id	int (11)	Foreign key, Not null	Customer id from register table
3.	Vehicle_type	varchar (50)	Not null	Type of vehicles
4.	brand	Varchar (50)	Not null	Vehicle brand
5.	model	Varchar (50)	Not null	Vehicle model
6.	Registration no	Varchar (20)	Not null	Vehicle registration number
7.	year	Int (11)	Not null	Manufacturing year

5.Table name: Services

Sl.no	Field Name	Data Type	Constraint	Description
1.	Service_id	int(11)	Primary key, Auto_increment, Not null	Service id
2.	Service_name	Varchar (100)	Not null	Service name
3.	Description	text	Not null	Service description
4.	Price	Decimal (10,2)	Not null	Service price

5.	Estimated time	Varchar (50)	Not null	Estimated service time
6.	Category	Varchar (100)	Not null	Service category
7.	Status	Enum('active', 'inactive')		Service status
8.	Duration minutes	Int (11)	Not null	Duration in minutes
9.	Image	Varchar (255)	Not null	Service image path
10.	Marketing description	text	Default null	Marketing description

6.Table name: bookings

Sl.no	Field Name	Data Type	Constraint	Description
1.	Booking_id	int (11)	Primary key, Auto_increment, Not null	Booking id
2.	User_id	int (11)	Foreign key, Not null	Customer id from register table
3.	Vehicle_id	Int (11)	Foreign key, Not null	Vehicle id from vehicles table
4.	Booking_datetime	datetime	Not null	Booking creation time
5.	Status	Varchar (50)	Default	Pending booking status
6.	Mechanic_id	Int (11)	Foreign key, Default null	Mechanic id from mechanic table
7.	Time_slot	Varchar (100)	Not null	Preferred time slot
8.	Appointment_date	date	Not null	Appointment date
9.	Preferred date	date	Not null	Users preferred date

7.Table name: booking _services

Sl.no	Field Name	Data Type	Constraint	Description
1.	id	int (11)	Primary key, Auto_increment, not null	Record id
2.	Booking_id	Int (11)	Not null	Booking id from booking table
3.	Service_id	Int (11)	Foreign Key, Not null	Services id from services table
4.	Services_price	Decimal (10,2)	Not null	Service price at booking time

8.Table name: mechanics

Sl.no	Field Name	Data Type	Constraint	Description
1.	Mechanic_id	int (11)	Primary key, Auto_increment, not null	Mechanic_id
2.	name	Varchar (100)	Not null	Mechanic name
3.	age	Int (11)	Not null	Mechanic age
4.	profession	Varchar (100)	Not null	Mechanic specialization
5.	Status	Enum ('free', 'assigned')	Default 'free'	Availability status
6.	Joined date	Date	Not null	Joining date
7.	Address	Text	Yes null, Default unread	Mechanic address
8.	Phone number	Varchar (15)	Not null	Phone number

9.	email	Varchar (100)	Not null	Email address
----	-------	---------------	----------	---------------

9.Table name: leave applications

Sl.no	Field Name	Data Type	Constraint	Description
1.	id	int (11)	Primary key, Auto_increment, Not null	Application_id
2.	Staff_id	Int(11)	Foreign key, Not null	Staff id from staff table
3.	Leave_reason	text	Not null	Reason for leave
4.	For_when	date	Not null	Leave start date
5.	Till_when	date	Not null	Leave end date
6.	Created_at_timestamp	Default	Current_timestamp	Application timestamp
7.	Status	Varchar (40)	Default null	Application status

10. Table name: messages

Sl.no	Field Name	Data Type	Constraint	Description
1.	id	int (11)	Primary key, Auto_increment, not null	message_id
2.	User_id	Int (11)	Foreign key, not null	customer id from register table
3.	Staff_id	Int (11)	Foreign key, default null	Staff id from staff table

4.	message	Text	Not null	Customer message
5.	response	Text	Default null	Staff response
6.	Created_at timestamp	Default	Current_timestamp	Message timestamp
7.	Responded_at	timestamp	Default null	Response timestamp

11. Table name: Service includes

Sl.no	Field Name	Data Type	Constraint	Description
1.	id	int (11)	Not null	Service_id
2.	service_id	Int (11)	Not null	Service_id
3.	Included_item	Varchar (255)	Not null	What and all included in the service

12. Table name: Service details

Sl.no	Field Name	Data Type	Constraint	Description
1.	id	int (11)	Not null	Service_id
2.	Service id	Int (11)	Not null	Service_id
3.	Why choose	Text	Not null	Marketing description

12. Table name: customer feedback

Sl.no	Field Name	Data Type	Constraint	Description
1.	Feedback_id	int (11)	Primarykey,Auto_in crement	Unique identifier for each feedback
2.	Use_id	Int (11)	Foreign key	Service_id
3.	Booking_id	Int	Foreign key, null	Links to
4.	rating	Int	Not null	Rating between 1-5
5.	comments	Text	null	
6.	Submitted_at	Datetime	null	Automatically stores when the feedback was given.

CODING

5 CODING

Coding section is where the magic happens. All the planning and the designing done in the previous sections come to life in this section. After this section can only the programmer enjoy the result of his/her hard work when he/she runs the program for the first time.

5.1 SELECTION OF SOFTWARE

PHP

PHP, an acronym for Hypertext Preprocessor, is a versatile server-side scripting language that falls under the broader category of software development. It is widely recognized for its pivotal role in web development and boasts several essential features that make it a preferred choice for building dynamic websites and web applications. Here are some of its key features:

open Source, Database Integration, Embedded in HTML, Cross-Platform Compatibility, Security

MYSQL

MySQL is an open-source relational database system, widely used for web development tasks like data storage, manipulation, and retrieval. It seamlessly integrates into web applications, eliminating the need for complex setup. MySQL is embedded within web development environments, making administrative tasks effortless. It operates as an SQL-based database, storing data in structured tables. Unlike systems requiring complex configuration, MySQL simplifies data access with its broad range of relational database features. Its features are: Zero configuration, Server less, Less memory, Self-contained, Transactional

5.2 CODING PHASE

The goal of the coding or programming phase is to translate the design of the system produced during

the design phase into code in each programming language, which can be executed by a computer and that performs the computation specified by the design. The coding phase affects both testing and maintenance profoundly. The coding phase does not affect the structure of the system; it has great impact on the internal structure of modules, which affects the testability of the system. The goal of the coding phase is to produce clear simple programs. The aim is not to reduce the coding effort, but to program in a manner so that testing and maintenance costs are reduced. Programs should not be constructed so that they are easy to write; they should be easy to read and understand. Reading programs is a much more common activity than writing programs. Hence, the goal of the coding phase is to produce simple programs that are clear to understand and modify.

5.2.1 CODING STANDARDS

The standard used in the development of the system is industry-standard PHP programming practices. It includes naming conventions of variables, constants and objects, standardized formats for labelling and commenting code, spacing, formatting and indenting.

Naming Conventions

The database tables and fields follow consistent naming patterns. Tables are named descriptively (register, bookings, services, etc.), fields are prefixed appropriately (user_id, booking_id, etc.), and relationships are clearly defined through foreign keys.

Labels and Comments

The functions of each module are clearly documented in the code. The database structure includes comments and constraints so that other developers using the system in future might understand the module functions better.

TESTING

&

IMPLEMENTATION

6 TESTING

Software testing is a critical element of software quality assurance and represents the ultimate review of specifications design and coding. Testing presents an interesting anomaly for the software engineer. Testing is a quality measure process, which reveals the errors in the program. During testing, the program is executed with a set of test cases and the output of the program for the test cases is evaluated to determine if the program is performing as it is expected. Testing plays a very critical role in determining the reliability and efficiency of the software and it is a very important stage in software development.

6.1 TESTING

System testing is a series of different tests whose primary purpose is to fully exercise the computer-based systems. Although each test has a different purpose, all work to verify that all system elements have been properly integrated and perform allocated functions. System testing is done to ensure that the system developed doesn't fail at any point. Before implementation, the system is tested with experimental data to ensure that it will meet the specified requirements. Special test data are input for processing and results examined.

6.1.1 TEST PLAN

Preparation of test data

Testing of the Car Service Management System is done using various kinds of test data. Preparation of test data plays a vital role in the system testing. After preparing the test data, the system under study is tested using that test data. While testing the system by using test data, errors are uncovered and corrected using testing steps and corrections are also noted for future use. Two kinds of test data were collected and used:

Using live test data

Live test data are those that are extracted from real automotive service operations. After the system is partially constructed, developers often ask service centre staff to enter a set of data from their normal

activities. Then, the system developers use this data to partially test the system. Customer bookings, vehicle information, and service records from actual operations are used for testing.

Using artificial test data

Artificial test data are created solely for test purposes, since they can be generated to test all combinations of formats and values. In other words, the artificial data, which can quickly be prepared by a data generating utility program, make possible the testing of all logic and control paths through the program. The most effective test program uses artificial test data generated by persons other than those who wrote the program. In this project, invalid data was entered to test whether the program would break or not. These invalid data entries included incorrect email formats, invalid vehicle registration numbers, and improper date formats. Many people were given the software for testing. They used various invalid inputs to test if every validation holds strong.

6.2 TESTING METHODS

Testing is generally done at two levels - testing of individual modules and testing of the entire system. During system testing, the system is used experimentally to ensure that the software does not fail, that is, that it will run according to its specifications and the results examined. A limited number of users may be allowed to use the system so analysts can see whether they use it in unforeseen ways. It is preferable to discover any surprises before the organization implements the system and depends on it. Testing is done throughout system development at various stages. It is always a good practice to test the system at many different levels at various intervals, that is, sub systems, program modules as work progresses and finally the system. During testing the major activities are concentrated on the examination and modification of the source code. Usually, this testing is to be performed by persons other than those who have coded it. This is done to ensure more complete and unbiased testing for making the software more reliable.

There are two types of testing:

- Black box testing
- White box testing

6.2.1 WHITE BOX TESTING

In white box testing, the internal logic of the modules is considered. Following levels of testing are performed for the developed project:

6.2.1.1 Unit Testing

This involves the tests carried out on modules programs, which make up a system. This is also called testing programs. The units in a large system have many modules at different levels that are needed. Unit testing focuses on the modules, independently of one another, to locate errors. The program should be tested for correctness of logic applied and should detect errors in coding. Before proceeding one must make sure that all the programs are working independently.

6.2.2 BLACK BOX TESTING

The concept of the black box is used to represent a system who's inside workings are not available for inspection. In a black box, the test item is treated as "black", since its logic is unknown; all that is known is what goes in and what comes out, or the input and output.

6.2.2.1 System Testing

The system testing is conducted on a complete, integrated system to evaluate the system's compliance with its specified requirement. It falls within scope of black box testing, so no knowledge of inner design or logic is needed. As a rule, system testing takes, as its input, all the integrated software components that have passed integration testing and the software system itself is integrated with any applicable hardware system. The purpose of the integration testing is to detect any inconsistencies between software units. System testing is the stage of implementation, which is aimed at ensuring that the system works accurately and efficiently before live operation commences. The logical design and the physical design should be thoroughly and continually examined on paper as ensure that they will work when implemented.

6.2.2.1 Integration Testing

Integration testing is a systematic technique for constructing the program structure, while at the same time conducting tests to uncover errors associated with interfacing. This is the program that is constructed and tested in small segments, which makes it easier to isolate and the following common types of integration problems may be observed:

6.2.2.2 Validation Testing

At the culmination of the integration testing, the software was completely assembled as a package, interfacing errors have been uncovered and corrected and a final series of software validation testing began.

In validation testing we test the system functions in a manner that can be reasonably expected by customers. The system was tested against system requirement specifications. Different unusual inputs that the users may use were assumed and the outputs were verified for such unprecedented inputs. Deviations or errors discovered at this step are corrected prior to the completion of this project with the help of users by negotiating to establish a method for resolving deficiencies. Thus, the proposed system under consideration has been tested by using validation testing and found to be working satisfactorily. Validation checking is performed on:

The numeric fields can contain only numbers from 0 to 9. Entry of any other character displays an error message. Fields like phone numbers, vehicle years, and service prices are validated. This field can only contain letters from A-Z and a-z. It is useful for name fields, vehicle brands, and service descriptions. Before entering values into the database or when updating, a validation is done to check whether any null fields are present. Email validation ensures proper email format with necessary validation checks. All email entries are verified before storage. Password validation ensures minimum security requirements are met. All password fields are validated before storage. Date validation ensures proper date formats for appointment booking and leave Applications. Registration number validation ensures proper format according to regional standards.

6.2.3 OUTPUT TESTING

After performing validation tests, the next phase is output testing of the system, since no system could be useful if it does not produce the desired output in the desired format. The format of reports/outputs were generated and tested. Here output formats were considered in multiple ways:

on the display screen, in booking confirmations, and in service reports.

6.2.4 USER ACCEPTANCE TESTING

User acceptance testing of the system is the key factor for the success of the system. The system under consideration was tested for user acceptance by keeping constant contact with the perspective users of the system at the time of design, development and making changes whenever required. This was done about the following points:

Input screen design, Output design, Navigation flow, Functionality completeness.

Users from each of the 3 user types (Admin, Staff, Customer) were selected for user acceptance testing. The admin was given the software for testing with admin credentials. Admin actions like managing services, bookings, and staff were performed to see whether all details are entering into the database and working properly as expected. The staff side was tested using staff credentials to verify leave applications and customer communication features. The customer side was tested by registering new customers and allowing them to book services, manage vehicles, and communicate with staff.

6.3 IMPLEMENTATION

Implementation is the stage of the project when the theoretical design is turned into a working system. The implementation stage is a system project. It includes careful planning, investigation of current system and its constraints on implementation, design of methods to achieve the changeover, training of the staff in the changeover procedure and evaluation of the changeover method.

The first task in implementation is planning - deciding on the methods and time scale to be adopted. Once the planning has been completed, the major effort is to ensure that the programs in the system are working properly when the users have been trained. The complete system involving both computer and users can be executed effectively. Thus, clear plans are prepared for the activities. Successful implementation of the new system design is a critical phase in the system life cycle. Implementation means the process of converting a new or a revised system design into an operational one.

MAINTENANCE

&

ENHANCEMENT

7 MAINTENANCE AND ENHANCEMENT

7.1 MAINTENANCE

This software can be modified as needs occur. Maintenance includes all the activities after installation of the software that are performed to keep the system operational. The process of maintenance involves: Understanding the existing software, Understanding the effect of changes, testing for satisfaction. This software requires minimal maintenance. During the testing phase, most maintenance duties are performed. If a maintenance requirement occurs, it can be solved with ease due to the modular architecture and well-documented code structure.

7.2 ENHANCEMENT

The Car Service Management System is built with a modular architecture, allowing for easy expansion and additional functionalities. As the automotive service business grows and customer demands evolve, the platform can seamlessly integrate new features to enhance the user experience. Future enhancements to the Car Service Management System could include:

Develop native mobile apps for Android and iOS platforms, allowing customers to book services, track progress, and communicate with staff on-the-go.

Integrate GPS tracking for service vehicles, allowing customers to track the arrival of mechanics for on-site services.

Implement online payment processing to allow customers to pay for services directly through the platform using credit cards, digital wallets, or other payment methods. Implement automated reminders for regular maintenance services based on vehicle mileage, time intervals, or manufacturer recommendations. Add inventory tracking for spare parts and consumables, with automatic reorder points and supplier management. Implement a comprehensive feedback system allowing customers to rate services and provide reviews, helping maintain service quality. Develop detailed analytics and reporting features for business intelligence, including revenue tracking, customer behaviour analysis, and operational efficiency metrics. Extend the system to support multiple service centre locations with centralized management and location-specific operations. Implement machine learning algorithms to suggest appropriate services based on vehicle history, age, and usage patterns. Connect with vehicle manufacturer systems for warranty information, recalls, and technical service bulletins.

These future developments will help enhance the Car Service Management System's offerings, attract new customers, and provide a seamless, comprehensive experience that stays competitive as technology advances in the automotive service industry.

CONCLUSION

8 CONCLUSION

The Car Service Management System revolutionizes the automotive service industry by digitizing traditional processes and enhancing customer experience. It provides a user-friendly platform for booking services, tracking maintenance, and communicating with service professionals efficiently. By addressing challenges like scheduling conflicts, poor communication, and manual record keeping, the system improves operational efficiency, transparency, and customer satisfaction. With features such as streamlined booking, resource management, and analytics, it supports business growth. Moving forward, the system aims to integrate mobile apps, IoT, and AI to further enhance service delivery and set new standards in digital automotive management.

BIBLIOGRAPHY

BIBLIOGRAPHY

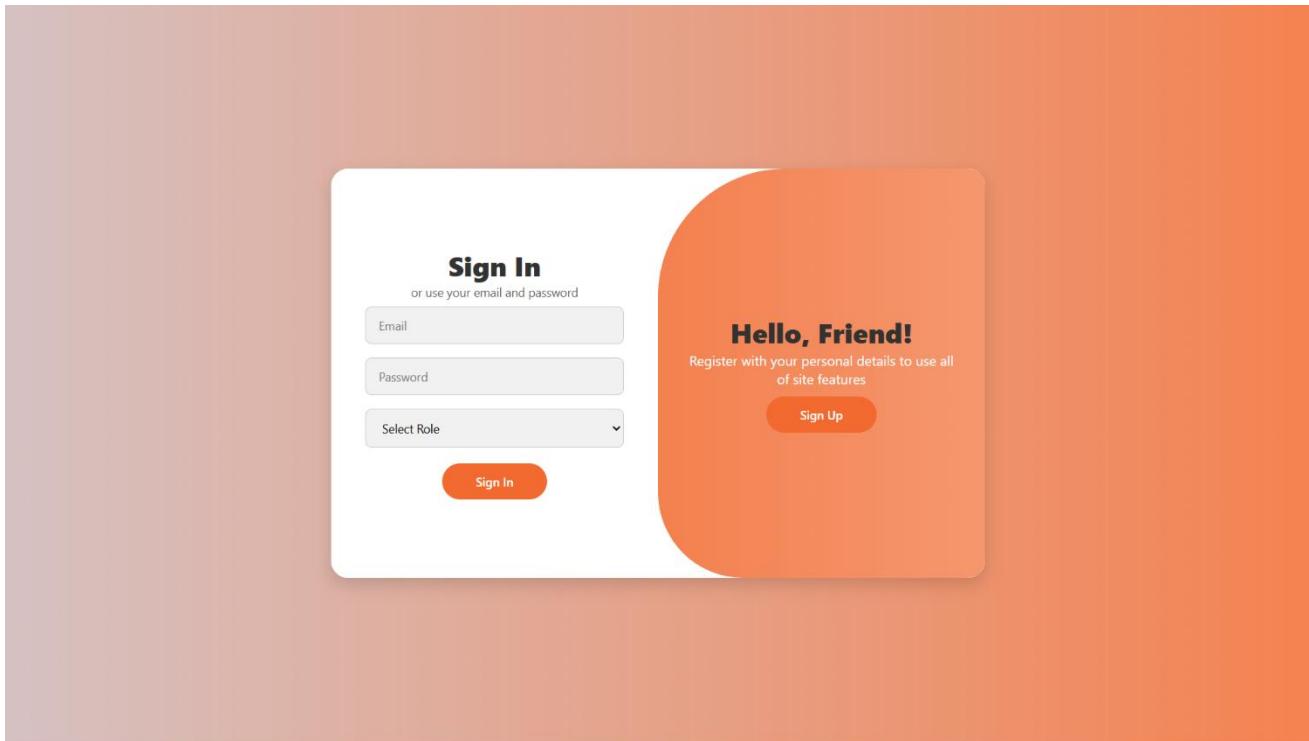
WEBSITES

- Chatgpt
- www.wikipedia.org
- www.tutorialpoint.com
- www.pinterest.com
- www.pexels.com
- www.w3schools.com

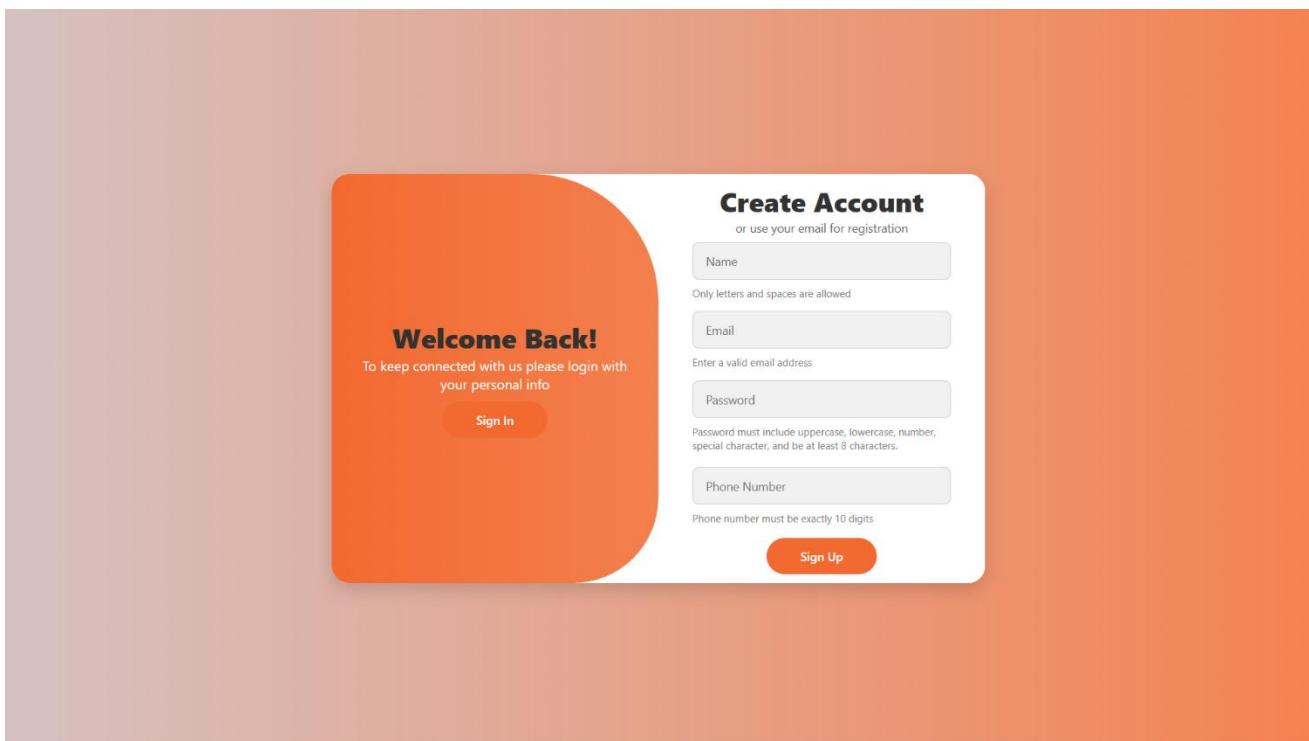
APPENDIX

SCREENSHOTS

1.LOGIN PAGE



2.REGISTRATION PAGE



3.HOME PAGE

The image shows a screenshot of a car service website. At the top, there's a navigation bar with the word "GARAGE" in red on the left and "Services", "Booking", "Contact", and "Profile" on the right. Below the navigation is a large, colorful illustration of a black pickup truck being lifted by a red hydraulic lift in a garage. The background features a stylized city skyline at night. The central text "Welcome to Garage" is displayed in white, bold letters over the truck. In the foreground, the main heading "Your Trusted Car Service Partner" is centered in a dark font. Below it, a sub-headline "Excellence in Every Service, Trust in Every Mile" is written in a smaller, lighter font. A welcome message "Welcome to Garage, your trusted car" is followed by a red "view Services" button. The page then displays four service offerings in boxes: "Expert Technicians" (illustrated with a technician and a car), "Quick Service" (illustrated with two people at a computer), "Affordable Pricing" (illustrated with a blue bag and a red arrow pointing up), and "Quality Guarantee" (illustrated with a person in a red ribbon wreath). At the bottom, there are four orange rectangular buttons.

GARAGE

Welcome to Garage

Your Trusted Car Service Partner

Excellence in Every Service, Trust in Every Mile

Welcome to Garage, your trusted car

view Services

Expert Technicians

Quick Service

Affordable Pricing

Quality Guarantee

150+

Registered Companies

3000+

Vehicles Serviced

20+

Expert Technicians

10 Years

Industry Experience

Auto-play: ON

Customer Testimonials

What Our Customers Have To Say



"Service was timely and efficient. My AC was fixed and I was updated with progress photos. Great experience!"

" Neha

Google Review



Contact Us

123 Garage St., Auto City, AC 12345
Email: support@garage.com
Phone: +1 (555) 123-4567

Follow Us

[Facebook](#) | [Twitter](#) | [Instagram](#)

© 2025 Garage. All rights reserved.

4. VEHICLE MANAGEMENT PAGE

GARAGE

Our Premium Services

Professional automotive care with cutting-edge technology and expert technicians

AC Service
Clean & refill AC gas
₹2,500.00 90 mins [More About](#)

TIRE PRESSURE
Check & refill air pressure
₹100.00 10 mins [More About](#)

Battery Replacement
Install new battery
₹4,000.00 60 mins [More About](#)

Car Washing
Exterior & interior wash
₹500.00 30 mins [More About](#)

Engine Cleaning
Deep clean engine
₹2,500.00 120 mins [More About](#)

General Inspection
Full vehicle check-up
₹1,000.00 60 mins [More About](#)

Oil Change
Replace engine oil
₹1,200.00 40 mins [More About](#)

Polishing & Waxing
Full body polish & wax
₹2,000.00 90 mins [More About](#)

Tyre Service
Tyre check & replacement
₹800.00 45 mins [More About](#)

Home Services Booking Contact Profile

5. SERVICE BOOKING PAGE

The screenshot displays two pages of a mobile application interface for a garage service provider.

Vehicle Registration & Service Booking

Add your vehicle for personalized service experience

Vehicle Registration

Add your vehicle for personalized service experience

Vehicle Type: Select Vehicle Type (dropdown menu)

Brand: Enter vehicle brand (text input)

Model: Enter vehicle model (text input)

Registration No (Format: KL XX YYYY): KL 01 A 1234

Example: KL 01 A 1234 (Districts: 01-14, Series: 1-2 letters, Numbers: 0001-9999)

Year: Enter manufacturing year (text input)

Vehicle should be between 1995 and 2025

Register Vehicle (orange button)

Book Services

Choose vehicle, select services & preferred date

Choose Vehicle: -- Select Vehicle -- (dropdown menu)

Preferred Date: dd-mm-yyyy (date input field)

Select your preferred service date (must be today or later)

AC Service ₹2500.00 (checkbox) 90 mins
Clean & refill AC gas (More About)

Air Filling ₹100.00 (checkbox) 10 mins
Check & refill air pressure (More About)

Battery Replacement ₹4000.00 (checkbox) 60 mins
Install new battery (More About)

Car Washing ₹500.00 (checkbox) 30 mins
Exterior & interior wash (More About)

Engine Cleaning ₹2500.00 (checkbox) 120 mins
Deep clean engine (More About)

General Inspection ₹1000.00 (checkbox) 60 mins
Full vehicle check-up (More About)

Home Services Booking Contact Profile

6. BOOKING CONFIRMATION PAGE

The screenshot shows a mobile application interface for 'GARAGE Auto Service Center'. At the top, there's a navigation bar with 'GARAGE' on the left, a user profile for 'naveen' in the center, and 'Services 7 TOTAL BOOKINGS', 'Booking 2 PENDING', 'Contact 3 PENDING', 'Profile', and 'Logout' on the right. Below the navigation, a large orange dialog box titled 'Confirm Payment' is displayed. The dialog contains the service provider's logo ('GARAGE Auto Service Center - Professional Car Service & Maintenance'), payment details (Customer Name: naveen, Booking ID: #17, Services: Air Filling, Battery Replacement), and the amount to pay (₹4,100.00). It includes two buttons: a green 'Confirm Payment' button with a checkmark icon and a grey 'Cancel' button with a cross icon. A note at the bottom of the dialog states: 'Note: This is a demo payment system. In production, integrate with actual payment gateways like Razorpay, Stripe, or PayPal.' In the background, there are sections for 'My Bookings' showing 'Booking #17' and 'Booking #16', and a summary of total bookings and paid amounts.

★ Service Feedback

We value your opinion! Please share your experience with our service.

Booking Details

Booking ID: #16

Status: Pending

How would you rate our service?



Select a rating

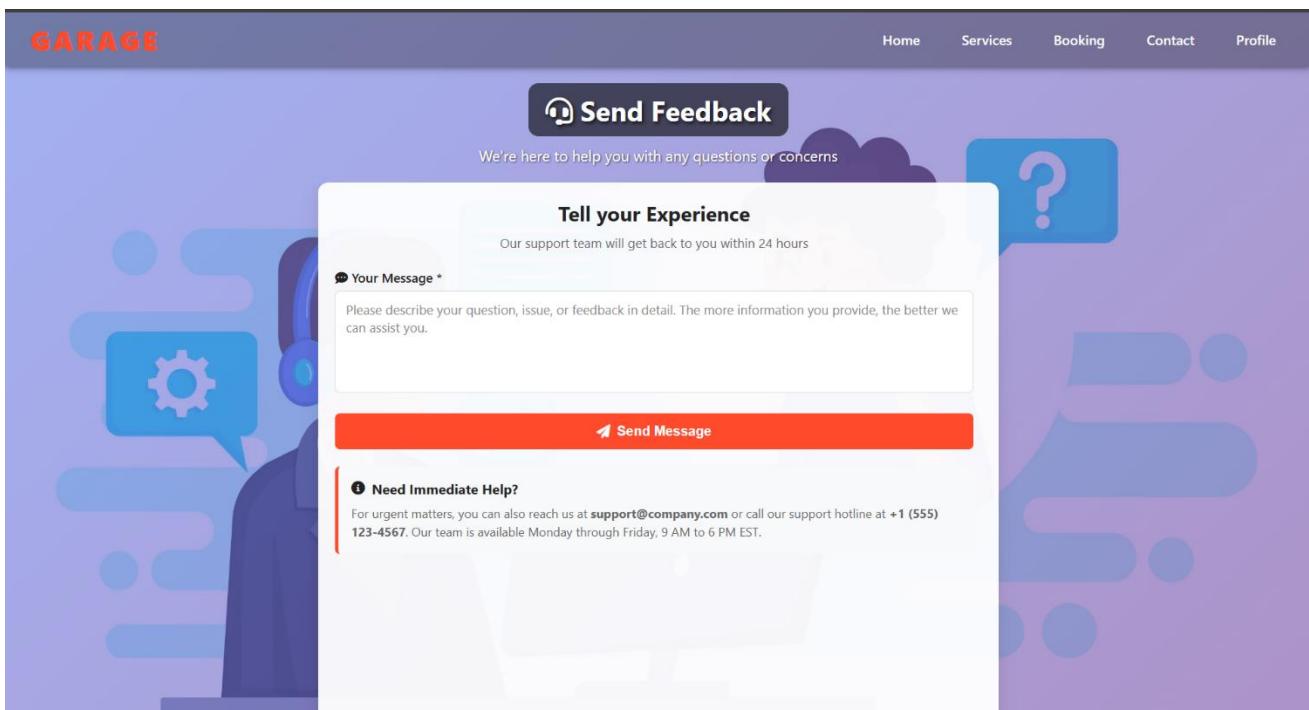
Additional Comments (Optional)

Please share any additional feedback about your experience...

Submit Feedback

Later

7. CUSTOMER MESSAGING PAGE



STAFF MODULE

1. STAFF DASHBOARD

A screenshot of the staff dashboard. On the left, a sidebar shows navigation options: Dashboard (selected), Bookings (7), All Bookings, Enquires, and Leave. At the bottom of the sidebar is a "Logout" button. The main area is titled "Staff Dashboard" and shows four summary cards: "Total Bookings" (11), "Pending Bookings" (7), "In Progress" (2), and "Completed" (0). Below these is a section titled "Pending Bookings" for "Booking #12" on Sep 29, 2025. It details the booking: Customer (naveen), Vehicle (MERCEDES c700, KL 06 2056), and Customer Preferred Date (Sep 30, 2025, Tuesday). Under "Services Requested", it lists AC Service (₹2500.00) and Oil Change (₹1200.00).

The screenshot displays the Garage Staff Dashboard interface. On the left, a sidebar menu includes options like Dashboard, Bookings (7), All Bookings, Enquiries, Leave, and Logout. The main content area shows a booking summary and a booking assignment form.

Services Requested

- AC Service: ₹2500.00
- Oil Change: ₹1200.00

Total Service Cost
₹3700

Assign Booking Details

Assign Mechanic: Select Available Mechanic (dropdown)

Time Slot: -- : -- : --

Status: Assigned

Schedule Service Date: Customer prefers: Sep 30, 2025

Appointment Date: 30 - 09 - 2025 (Cannot be set to past dates)

Minimum date: Sep 29, 2025 (Today) | Pre-filled with customer's preferred date

Update Booking

Booking #11 (Sep 26, 2025)

Customer: naveen

Vehicle: bharat c700
KL 14 AF 6363

Customer Preferred Date: Oct 04, 2025 Saturday

Services Requested

- AC Service: ₹2500.00
- Air Filling: ₹100.00

Total Service Cost
₹2600

2.LEAVE APPLICATION PAGE

The screenshot shows the Garage Staff Dashboard with a sidebar on the left containing links for Dashboard, Bookings, All Bookings, Enquiries, and Leave. The Leave link is highlighted with a red background. The main content area is titled "Leave Application" and "Home > Leave Management". It features a header "Leave Application System" with a welcome message "Welcome, staff1. Manage your leave requests efficiently." Below this is a form titled "Apply for Leave" with fields for "Leave Reason *", "From Date" (set to Sep 30, 2025), "Number of Days *", "Till Date" (with placeholder "Select number of days"), and a green "Submit Leave Application" button.

3.CUSTOMER MESSAGES PAGE

The screenshot shows the Garage Staff Dashboard with a sidebar on the left containing links for Dashboard, Bookings (with a notification count of 7), All Bookings, Enquiries (with a notification count of 1), and Leave. The Enquiries link is highlighted with a red background. The main content area is titled "Messages Management" and "Home > Messages". It displays summary statistics: 2 Total Messages, 1 Pending Replies, 1 Replied, and 0 Today's Messages. Below this is a section titled "Customer Messages" showing a message from "adithian sonthosh" (adithian12@gmail.com | 6238167077) received on Sep 17, 2025 at 18:38 with status "Pending". The message content is "bad experience". A "Send SMS Reply" button is present, with a note that the SMS will be sent to 6238167077. The reply message field is empty.

The screenshot displays the Garage Staff Dashboard. On the left sidebar, there are several navigation items: Dashboard, Bookings (with 7 notifications), All Bookings, Enquiries (with 1 notification), and Leave. At the bottom of the sidebar, it shows the user is 'staff' and provides a 'Logout' button.

The main content area is titled 'Send SMS Reply'. It shows the recipient's phone number: 6238167077. Below this is a text input field labeled 'Your Reply Message' with the placeholder 'Type your reply message here. This will be sent as SMS to the customer's phone number...'. A green button labeled 'Send SMS Reply' is located below the message input.

Below this section, a message history is shown between the staff member 'vishnu' and a customer. The staff message 'jkhhgrytdtuk' is timestamped 'Sep 16, 2025 - 09:58' and has a green 'Replied' button. The customer's message 'vjhbjj' is timestamped 'Sep 16, 2025 am30 09:58'. A green button labeled 'Your Reply (SMS Sent)' is visible above the customer's message.

ADMIN MODULE

1. ADMIN DASHBOARD

The screenshot shows the Admin Dashboard interface. On the left is a sidebar with the title "Admin Panel" and "Garage". The sidebar contains the following navigation items:

- Manage Bookings (selected)
- Manage Users
- Manage Staff
- Manage Services
- Manage Mechanics
- Manage Leave
- Settings
- Logout

The main content area is titled "Manage Bookings". It displays five summary cards:

- Total Bookings: 11
- Pending: 7
- Confirmed: 1
- In Progress: 2
- Completed: 0
- Cancelled: 0

Below these cards is a section titled "All Bookings" with a table. The table has the following columns:

Booking ID	Customer	Vehicle	Services	Date & Time	Mechanic	Status	Amount	Actions
#0012	naveen ID: 1	MERCEDES c700	AC Service, Oil Change	Nov 30, -0001	Not Assigned	PENDING	\$3,700.00	

2. MANAGE SERVICE PAGE

Admin Panel

Garage

- [Manage Bookings](#)
- [Manage Users](#)
- [Manage Staff](#)
- [Manage Services](#)
- [Manage Mechanics](#)
- [Manage Leave](#)
- [Settings](#)
- [Logout](#)

Manage Services

Services
Service Includes
Service Details
Service Images

Add New Service

Service Name	Description	Price (\$)
<input type="text"/>	<input type="text"/>	<input type="text"/>
Duration (minutes)	Category	
<input type="text"/>	<input type="text"/>	
Add Service		

Existing Services



AC Service
Car service

Clean & refill AC gas

\$ \$2,500.00 90 mins

[Edit](#) [Deactivate](#) [Delete](#)



TIRE PRESSURE
Car service

Check & refill air pressure

\$ \$100.00 10 mins

[Edit](#) [Deactivate](#) [Delete](#)



Battery Replacement
Car service

Install new battery

\$ \$4,000.00 60 mins

[Edit](#) [Deactivate](#) [Delete](#)



CAR WASH





81

Service	Status	Description	Price	Duration
Car Washing	ACTIVE	Exterior & interior wash	\$ 500.00	30 mins
Engine Cleaning	ACTIVE	Deep clean engine	\$ 2,500.00	120 mins
General Inspection	ACTIVE	Full vehicle check-up	\$ 1,000.00	60 mins
Refueller	PENDING	Car service		
Oil Change	ACTIVE	Replace engine oil	\$ 1,200.00	40 mins
Polishing & Waxing	ACTIVE	Full body polish & wax	\$ 2,000.00	90 mins
Tyre Service	ACTIVE	Tyre check & replacement	\$ 800.00	45 mins

3. MANAGE BOOKING PAGE

All Bookings

Booking ID	Customer	Vehicle	Services	Date & Time	Mechanic	Status
#0012	naveen ID: 1 naveensaji321@gmail.com 6282952601	MERCEDES c700 Sedan KL 06 2056	AC Service, Oil Change	Nov 30, -0001	Not Assigned	PENDING
#0011	naveen ID: 1 naveensaji321@gmail.com 6282952601	bharat c700 Motorcycle KL 14 AF 6363	AC Service, Air Filling	Nov 30, -0001	Not Assigned	PENDING

Admin Panel Garage	#0004 sreenari ID: 5 sree234@gmail.com 4356253763	BMW c700 Sedan KL 03 2014	Air Filling, General Inspect...	AM - 12:00 PM Booked: Sep 19, 2025 08:53	Not Assigned Assign	PENDING	\$4,300.00	
	#0003 naveen ID: 1 naveneesajj321@gmail.com 6282952601	MERCEDES BENZ SUV KL 06 2009	Car Washing	Sep 29, 2025 14:04 Booked: Sep 19, 2025 08:33	sasi ID: 1	IN PROGRESS	\$500.00	
	#0002 vivek ID: 3 bennyvivek3@gmail.com 6282795821	BMW C Sedan KL 03 6878	Air Filling	Sep 20, 2025 14:09 Booked: Sep 17, 2025 08:38	sasi ID: 1	CONFIRMED	\$100.00	
	#0001 naveen ID: 1 naveneesajj321@gmail.com 6282952601	MERCEDES BENZ SUV KL 06 2009	Polishing & Waxing	Sep 18, 2025 10:57 Booked: Sep 17, 2025 05:25	sasi ID: 1	ASSIGNED	\$2,000.00	

4. MANAGE STAFF PAGE

The screenshot shows the 'Manage Staff' section of the Admin Panel. On the left sidebar, under the 'Admin Panel' heading, 'Manage Staff' is highlighted with a purple background. The main content area has a header 'Manage Staff' with a three-line menu icon. Below it is a section titled 'Staff Statistics' with four colored boxes: teal (Total Staff: 3), orange (Staff with Leaves: 1), red (This Month: 3), and light green (Today: 0). The next section is 'Add New Staff' with fields for Staff Name, Email Address, Password, and Phone Number, followed by a blue 'Add Staff' button. At the bottom is a section titled 'All Staff Members' listing three staff members: 'thambu' (ID: 287), 'Rahul' (ID: 75), and 'staff1' (ID: 1). Each card shows their name, email, phone number, role (staff), and last login date. Below each card are 'View Leaves', 'Edit', and 'Delete' buttons.

Manage Staff

Staff Statistics

3 Total Staff

1 Staff with Leaves

3 This Month

0 Today

Add New Staff

Staff Name:

Email Address:

Password:

Phone Number:

Add Staff

All Staff Members

thambu
tham2@gmail.com
9347562388
staff
Sep 26, 2025
ID: 287

Rahul
rahul@gmail.com
7868772344
staff
Sep 17, 2025
ID: 75

staff1
staff@example.com
2345678910
staff
Sep 09, 2025
ID: 1

Table View

5. MANAGE MECHANICS PAGE

≡ Manage Mechanics

2 Total Mechanics 2 Available 0 Assigned

Add New Mechanic

ⓘ Status will be set to "Available" and joined date will be automatically set to today's date.

Full Name	Age	Profession/Specialization	Phone Number
<input type="text"/>	<input type="text"/>	<input type="text"/> e.g., Engine Specialist, Brake Expert	<input type="text"/>
Email Address	Address	<input type="text"/>	
+ Add Mechanic			

Existing Mechanics (2)

sreehari <small>Tyre Service</small> <small>FREE</small>	sasi <small>engine</small> <small>FREE</small>
<small>32 years old</small> <small>Joined: 2025-09-19</small> <small>852478391</small> <small>sree432@gmail.com</small> <small>malabari p.o</small>	<small>34 years old</small> <small>Joined: 2025-09-17</small> <small>877878723</small> <small>sasi@gmail.com</small> <small>kdjbfjbd</small>
<small>Edit</small> <small>Delete</small>	<small>Edit</small> <small>Delete</small>

6.LEAVE MANAEMENT PAGE

Admin Panel
Garage

- Manage Bookings
- Manage Users
- Manage Staff
- Manage Services
- Manage Mechanics
- Manage Leaves**
- Settings
- Logout

Manage Leave Applications

Leave Statistics

4
Total Applications

2
Pending Approval

2
Approved

0
Currently On Leave

All Leave Applications

staff1	staff@example.com	PENDING APPROVAL	1 day(s)
From: Sep 27, 2025	To: Sep 27, 2025	Applied: Sep 26, 2025 12:54	2345678910
Leave Reason Emergency Leave			
<input checked="" type="button"/> Approve	<input type="button"/> Reject	<input type="button"/> Delete	

7. MANAGE FEEDBACK

Admin Panel
Garage

- Manage Bookings
- Manage Users
- Manage Staff
- Manage Services
- Manage Mechanics
- Manage Leave**
- Manage Feedback**
- Settings
- Logout

Manage Feedback

Feedback Overview

Average Rating

3/5

Based on 4 reviews

Total Feedback

4

Customer Reviews

Satisfaction Rate

25%

4 & 5 Star Reviews

Rating Distribution

5 0 4 3 2 1 0

Customer Feedback

naveen	naveensaji321@gmail.com	★★★★★ 3/5
--------	-------------------------	-----------