A Project Report On

"VEHICLE SERVICE MANAGEMENT"

Submitted to the

SCHOOL OF COMPUTER SCIENCE

In partial fulfillment of the

MASTER OF COMPUTER APPLICATION

Under the guidance of

***Asst. Prof.* AMAL DEV K**

Project Done by

THOMAS MATHEW SUNNY

(Reg. No:243242210221)

SCHOOL OF COMPUTER SCIENCE

DE PAUL INSTITUTE OF SCIENCE & TECHNOLOGY (DiST)

ANGAMALY SOUTH, KERALA

(Affiliated to Mahatma Gandhi University, Kottayam)

November 2025

DE PAUL INSTITUTE OF SCIENCE & TECHNOLOGY (DiST)

ANGAMALY SOUTH, KERALA

BONAFIDE CERTIFICATE

Certified that the Project Work entitled

"VEHICLE SERVICE MANAGEMENT"

is a bonafide work done by

THOMAS MATHEW SUNNY

In partial fulfillment of the requirement for the Award of
MASTER OF COMPUTER APPLICATIONS

Degree From

Mahatma Gandhi University, Kottayam(2024-2026)

Head of the Department *Project Guide*

Submitted for the Viva-Voce Examination held on……………………………………..

External Examiner 1 **External Examiner 2**

DE PAUL INSTITUTE OF SCIENCE & TECHNOLOGY (DiST)
ANGAMALY SOUTH, KERALA

CERTIFICATE

This is to certify that the project entitled **“VEHICLE SERVICE MANAGEMENT”** has been successfully carried out by ***THOMAS MATHEW SUNNY*** (Reg. No: 243242210221) in partial fulfilment of the Course **Master of Computer Applications.**

INTERNAL GUIDE

Date: **HEAD OF THE DEPARTMENT**

DE PAUL INSTITUTE OF SCIENCE & TECHNOLOGY (DiST)
ANGAMALY SOUTH, KERALA

CERTIFICATE

This is to certify that the project entitled “**VEHICLE SERVICE MANAGEMENT**” has been successfully carried out by **THOMA SMATHEW SUNNY** (Reg. No: 243242210221) in partial fulfilment of the course **Master of Computer Applications** under my guidance.

INTERNAL GUIDE

Date: **Asst. Prof. AMAL DEV K**

DE PAUL INSTITUTE OF SCIENCE & TECHNOLOGY (DiST)

ANGAMALY SOUTH, KERALA

DECLARATION

I, **THOMAS MATHEW SUNNY**, hereby declare that the project work entitled "**VEHICLE SERVICE MANAGEMENT**" is an authenticated work and is carried out by me at ***D'Soft Solutions.*** under the guidance of **Asst. Prof. AMAL DEV K** for the partial fulfilment of the course **MASTER OF COMPUTER APPLICATIONS**. This work has not been submitted for similar purpose anywhere else except to **DE PAUL INSTITUTE OF SCIENCE & TECHNOLOGY (DiST).**

I understand that detection of any such copying is liable to be punished in any way the school deems fit.

Date:

Place: **THOMAS MATHEW SUNNY**

ACKNOWLEDGMENT

First and foremost, praises and thanks to the God, the Almighty, for His showers of blessings throughout my project work to complete the project successfully.

I respect and thank **Fr. (Dr). Johny Chacko Mangalath V C,** Our Principal of De Paul Institute of Science & Technology (DiST) Angamaly giving me all necessary facilities in the college for successful completion of this dissertation.

I record my gratitude to **Assoc. Prof. Jacob Thaliyan** HOD, School of Computer Science (PG) for his support and guidance which made me complete the project duly. I am extremely thankful to him for providing such a nice support and guidance, although he had busy schedule.

I owe my deep gratitude to **Asst. Prof. AMAL DEV K**, Guide of this project who took keen interest on my project work and guided me all along, till the completion of my project work by providing all the necessary information for developing a good system.

I hereby express my sincere thanks to **Asst. Prof. Rosemary Varghese** class in charge , for the kind cooperation and valuable suggestions throughout the course of my project.

I am thankful to and fortunate enough to get constant encouragement, support and guidance from all teaching staffs of SCHOOL OF COMPUTER SCIENCE which helped me in successfully completing my research work.

I am thankful to my family and friends who were a source of inspiration for the successful completion of this project work. Finally, yet importantly, my thanks go to all the people who have supported me to complete the project directly or indirectly.

THOMAS MATHEW SUNNY

TABLE OF CONTENTS

1. INTRODUCTION 1

1.1 Introduction 2

1.2 Problem statement 2

1.3 Scope and relevance of the project 2

1.4 Objectives 3

2. SYSTEM ANALYSIS 4

2.1 Introduction 5

2.2 Existing system 5

2.3 Proposed system 6

2.4 Feasibility study 6

2.5 Software engineering paradigm applied 8

3. SYSTEM DESIGN 9

3.1 Introduction 10

3.2 Database design 10

3.3 Object Oriented Design-UML diagrams 16

3.4 MODULES DESCRIPTION 26

3.4 Input design 27

3.5 Output design 28

4. SYSTEM ENVIRONMENT 29

4.1 Introduction 30

4.2 Software requirement specification 30

4.3 Hardware requirements specification 30

4.4 Tools, Platforms Front end tool 30

Back end tool 32

PHP 32

MYSQL 32

WAMPSERVER 33

OPERATING SYSTEM 33

WINDOWS 11.. 33

5. SYSTEM IMPLEMENTATION….. **34**

5.1 Introduction… 35

5.2 Sample coding with home page...35

6. SYSTEM TESTING...**44**

6.1 Introduction… 45

6.2 Unit Testing..45

6.3 System Testing… 46

6.4 Integration Testing 46

7. SYSTEM MAINTENANCE **47**

7.1 Introduction… 48

7.2 Maintenance 48

8. FUTURE ENHANCEMENT&SCOPE OF DEVELOPMENT… **49**

8.1 Introduction… 50

8.2 Merit of the system…..50

8.3 Limitation of the system…..50

8.4 Future Enhancement of system… 50

9. CONCLUSION…...**51**

Conclusion…...51

10. BIBLOGRAPHY… **53**

References… 54

11. APPENDIX **55**

SCREENSHOTS 56

CODE… 60

INTRODUCTION

1.1 Introduction

The **Vehicle Service Management System** is a **web-based application** that helps customers easily book vehicle maintenance and repair services online. Users can register, browse available services, and schedule appointments according to their preferred time slots. In the existing system, customers need to call or visit the service center to book a slot, which can be time-consuming and lead to miscommunication or delays. This online system removes such problems by providing an easy-to-use platform where customers can view available services, check real-time availability, and book appointments conveniently. It ensures faster scheduling, improved accuracy, and a better overall experience for both customers and service providers.

1.2 Problem statement

Efficiency and effectiveness are major concerns in the vehicle service industry. Vehicle owners and service centers often face challenges in managing service bookings, leading to delays, miscommunication, and scheduling conflicts, which negatively affect overall service performance. There is a need for a system that allows customers to book vehicle services conveniently and ensures smooth management of appointments by the service center. I therefore propose the design and implementation of a **Vehicle Service Management System** that enables customers to schedule services online, while allowing administrators and drivers to efficiently manage bookings, track service progress, and reduce manual errors, ensuring timely and hassle-free vehicle maintenance.

1.2 Scope and relevance of the project

The project aims to develop a **user-friendly web application for vehicle service management**, making the process of booking and managing vehicle maintenance easier and more efficient. It includes features such as online service booking, real-time service status tracking, customer profile management, driver assignment for pickup and delivery, and comprehensive reporting. This system improves the user experience for customers while enabling service centers to manage appointments, resources, and service operations more effectively.

By digitizing the vehicle service workflow, the project ensures timely service, reduces administrative errors, and enhances overall efficiency for all stakeholders involved.

1.3 Objectives

- To digitalize and streamline traditional vehicle servicing processes by providing a fully automated online platform for service booking, tracking, and delivery
- To offer customers a user-friendly system for booking vehicle services, selecting service types, scheduling appointments, tracking service status, making secure payments, and giving feedback.
- To enable service centers, mechanics, and drivers to efficiently manage service requests, update service progress, handle customer communication, and maintain vehicle service records.
- To provide administrators with a centralized dashboard for managing users, mechanics, drivers, service categories, bookings, payments, service history, and overall system operations.

SYSTEM ANALYSIS

2.1 Introduction

System analysis is a crucial phase in system development. It involves a detailed study of the system to understand its processes and workflows thoroughly. In this phase, the system analyst examines the current vehicle service management process, identifying inputs, operations, and outputs. Data is collected through interviews, observation, and questionnaires, and analyzed to detect problems and inefficiencies. The goal is to gain a clear understanding of the system's functions and determine areas for improvement. Multiple solutions are explored during this phase, and the most feasible one is selected to address the identified challenges in the vehicle service process, ensuring efficiency and accuracy

2.2 Existing system

The current vehicle service system relies heavily on manual booking and management processes. Customers must call or visit the service center to schedule appointments. This approach is time-consuming and prone to errors such as double bookings, miscommunication about service requirements, and long waiting times. Service records, customer details, and booking information are maintained in separate files, making it difficult for administrators to access, track, or update information quickly. The process is inefficient due to its reliance on paperwork, phone calls, and manual coordination. A modern, automated system is needed to improve speed, accuracy, and convenience for both customers and service centers.

Limitations of the Existing System

1. **Time-Consuming:** Manual booking and service management involve multiple steps, taking up significant time for both customers and service staff.

2. **Difficult Maintenance** : Maintaining service records, customer details and booking information in separate files makes data management challenging and prone to error

2.3 Proposed system

The proposed system aims to address the issues in the existing system by automating the entire process. It is a web-based application that maintains a centralized database for all relevant information, making it easier to manage compared to the current system. The system is developed using PHP and MySQL for the backend, and HTML, CSS, and JavaScript for the frontend, ensuring a user-friendly experience while reducing manual workload. This system provides an efficient solution for restaurant reservation and managing related data, meeting customer demands seamlessly and without delay

Advantages of proposed system

- Saves time
- User friendly and easy to use
- 24/7 Accessibility
- Enhanced Experience

2.4 Feasibility study

All projects are feasible when given unlimited resources and infinite time. It is both necessary and prudent to evaluate the feasibility of a project at the earliest possible time. An estimate is made of whether the identified user needs may be satisfied using current software and hardware technologies. The study will decide if the proposed system will be cost effective from the business point of view and if it can be developed in the given existing budgetary constraints. The feasibility study should be relatively cheap and quick. The result should inform the decision of whether to go ahead with a more detailed analysis.

Technical feasibility

The project requires the system to be functional and multi-user one. It should be based on specified technology. The system understudy must be portable and platform independent. It should be compatible with all kind of existing system in industry and should not provide any overhead to the user. Technical feasibility senders on the existing system and to what extent it can support the proposed system. Updating and viewing the previous edition are tedious in the existing system. Implementation of proposed system does not require saving of the existing configuration of the system. Any system developed must not have a high demand on the available technical resources. This will lead to high demands on the available technical resources and being placed on the client. The developed system must have a modest requirement as only minimal or null changes are required for implementing this system. This study is carried out to check the technical feasibility that is the technical requirement of the system. Technical analysis center on the existing computer system (hardware, software, etc.) and to what extend it can support the proposed system. Technical 14 feasibility study deals with the hardware as well as software requirements. The scope was whether the work for the project is done with the current equipment's and the existing software technology. Our system requires higher software requirements such as PHP, MYSQL, windows 8 or higher OS , a web browser and internet. Since these are all available, this system is technically feasible.

Operational feasibility

The system operation is longest phase in the development life cycle of a system so operational feasibility should be given much important. The users of the system don't need go through training on the system. All they are expected to know to operate the system is the basic net surfing knowledge. It has a user-friendly interface. The present system is easily understandable and maintenance and working of a new system needs minimal human efforts. The proposed project is beneficial to the organization and is user friendly. So the system can be judged operationally feasible. The purpose of the operational feasibility study is to determine whether the new system will be used if it is developed and implemented. Our system will operate after it is developed and be operative once it is installed. The system is opera table by the user easily. Thus this system is operationally feasible.

Economic feasibility

The proposed system is economical feasible. The economic feasibility is the most important and used method for evaluating the effectiveness of the proposed system. It is very essential because the main goal of the proposed system is to have economically better result along with increased efficiency. It is also known as cost benefit analysis. If the benefit is more than the cost, then the system is approved. The given system can be developed under optimal expense with the available hardware and software. Besides it is good economic to invest such a kind of software from the project manager's point of view as the benefit overweight the cost. The resources need to run the above project should be less in cost and highly reliable so that there might be no hanging and minimum level of expense to implement the software

2.5 Software engineering paradigm applied

In this project, incremental software development paradigm approach is used build the software, which means the entire project is divided into small part and each part is developed and delivered incrementally. The incremental is iterative in nature, allowing for the development of a functional product through successive cycles or incremental. The incremental software paradigm is particularly beneficial in projects where requirements are not well-defined initially, as it allows for flexibility and responsiveness to changing needs. It promotes early and continuous delivery of valuable functionality, reducing the time to market and increasing customer satisfaction.

SYSTEM DESIGN

3.1 Introduction

System design is the solution to the creation of a new system. This phase is composed of several systems. This phase focuses on the detailed implementation of the feasible system. System design has two phases of development logical and physical design. During logical design phase the analyst describes inputs (sources), out puts (destinations), databases (data sores) and procedures (data flows) all in a format that meats the uses requirements. Design goes through the logical and physical stages of development. At an early stage in designing a new system, the system analyst must have a clear understanding of the objectives, which the design is aiming to fulfil. Second input data and master files (database) have to be designed to meet the requirements of the proposed output. The operational (processing) phases are handled through program construction and testing. The system design includes:

- Output design
- Database design
- Input design
- Form design
- Architectural design
- System modules

3.2 Database design

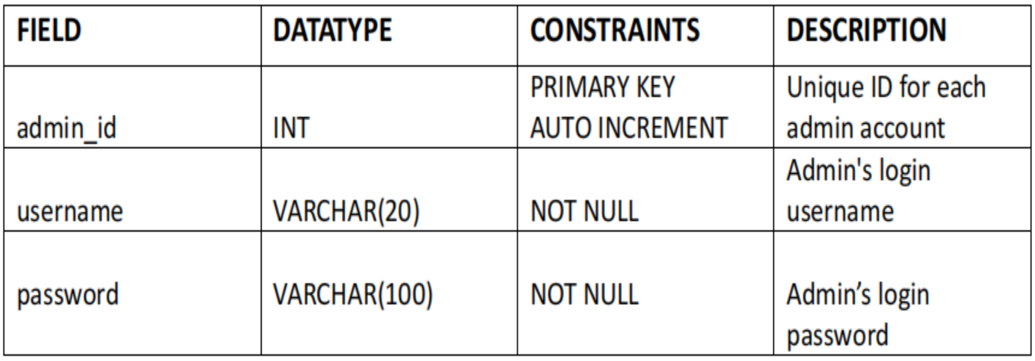
Data Base design is the logical form of design of data storage in the form of records in a particular structure in the form of tables with fields which is not transparent to the normal user but it actually acts as the backbone of the system. As we know database is a collection of which helps the system to manage and store data is called database management system. Data base management system builds some form of constraints like integrity constraints, i.e., the primary key or unique key and referential integrity which help to keep data structure storage and access of data from tables efficiently and accurately and take necessary steps to concurrent access of data and avoid redundancy of data in tables by normalization criterions. Normalization is the method of breaking down complex table structures into simple table structures by using certain rules thus reduce redundancy and inconsistency and disk space usage and thus increase the performance of the system or applications which directly linked to the database design and also solve the problems of anomalies. There are different forms of normalization, some are:

- First Normal form (1NF)
- Second Normal form (2NF)
- Third Normal form (3NF)

The database design of the new system is in Second normal form and every non-key attribute is functionally depends only on the primary key. The master and transaction tables and their structure are shown below.

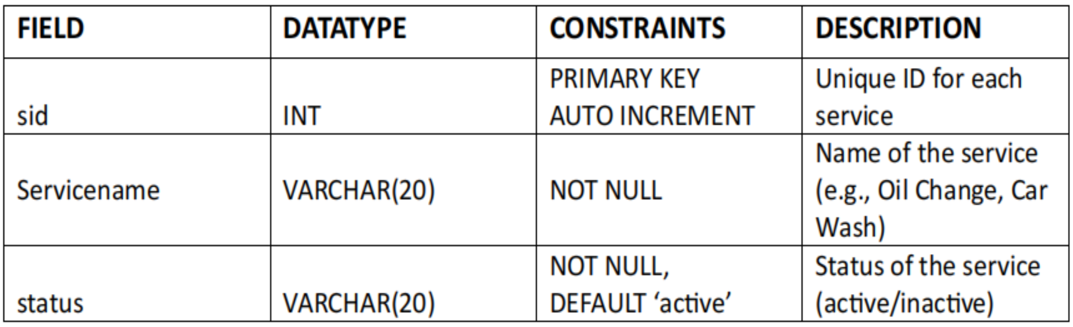
Table design

1. adminlogin



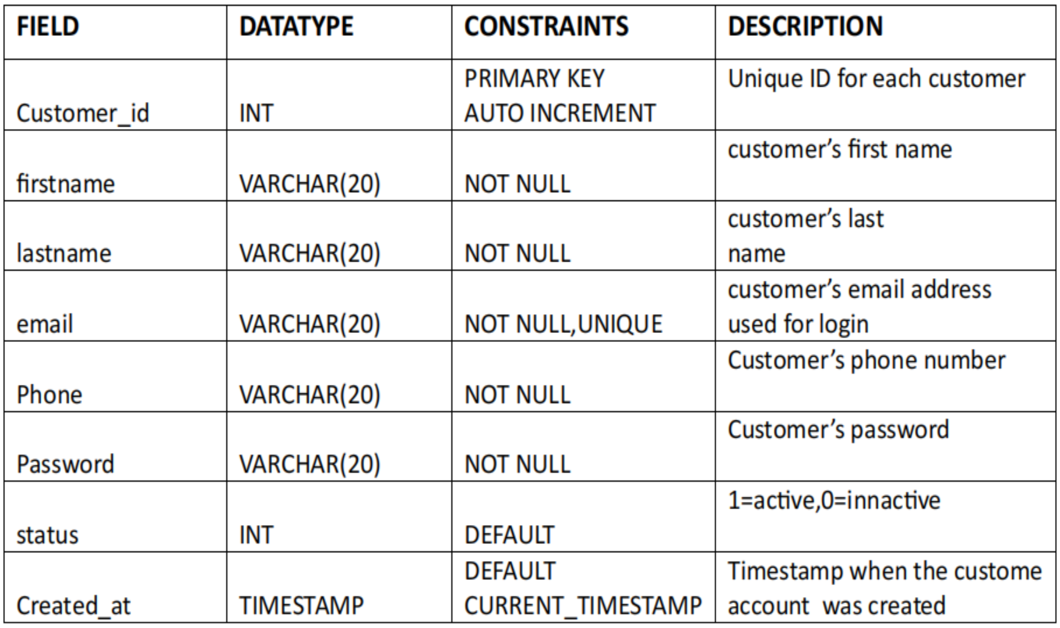
FIELD	DATATYPE	CONSTRAINTS	DESCRIPTION
admin_id	INT	PRIMARY KEY AUTO INCREMENT	Unique ID for each admin account
username	VARCHAR(20)	NOT NULL	Admin's login username
password	VARCHAR(100)	NOT NULL	Admin’s login password

2.services



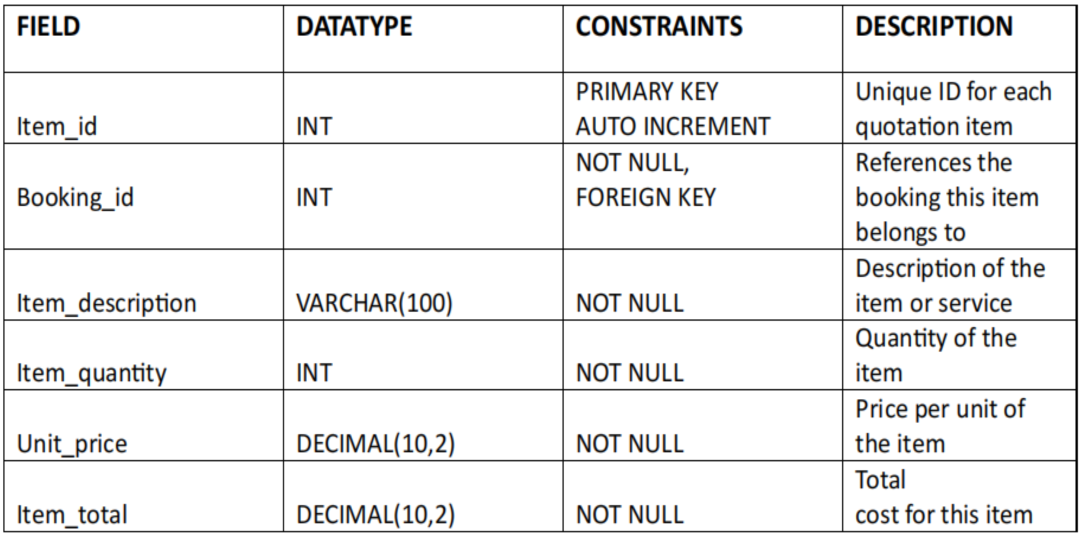
FIELD	DATATYPE	CONSTRAINTS	DESCRIPTION
sid	INT	PRIMARY KEY AUTO INCREMENT	Unique ID for each service
Servicename	VARCHAR(20)	NOT NULL	Name of the service (e.g., Oil Change, Car Wash)
status	VARCHAR(20)	NOT NULL, DEFAULT ‘active’	Status of the service (active/inactive)

3.customerreg



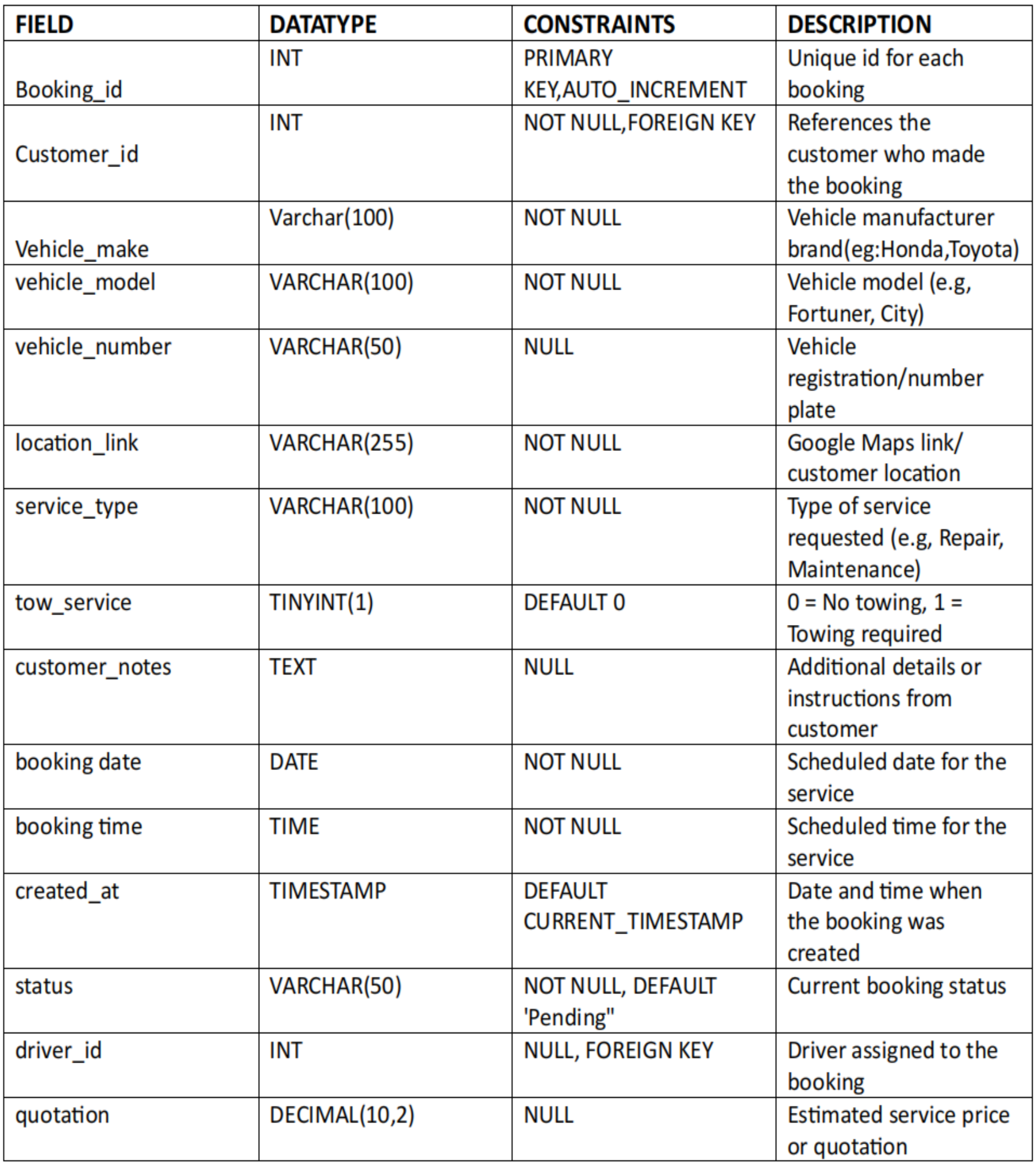
FIELD	DATATYPE	CONSTRAINTS	DESCRIPTION
Customer_id	INT	PRIMARY KEY AUTO INCREMENT	Unique ID for each customer
firstname	VARCHAR(20)	NOT NULL	customer's first name
lastname	VARCHAR(20)	NOT NULL	customer's last name
email	VARCHAR(20)	NOT NULL,UNIQUE	customer's email address used for login
Phone	VARCHAR(20)	NOT NULL	Customer's phone number
Password	VARCHAR(20)	NOT NULL	Customer's password
status	INT	DEFAULT	1=active,0=innactive
Created_at	TIMESTAMP	DEFAULT CURRENT_TIMESTAMP	Timestamp when the custome account was created

4.quation_items



FIELD	DATATYPE	CONSTRAINTS	DESCRIPTION
Item_id	INT	PRIMARY KEY AUTO INCREMENT	Unique ID for each quotation item
Booking_id	INT	NOT NULL, FOREIGN KEY	References the booking this item belongs to
Item_description	VARCHAR(100)	NOT NULL	Description of the item or service
Item_quantity	INT	NOT NULL	Quantity of the item
Unit_price	DECIMAL(10,2)	NOT NULL	Price per unit of the item
Item_total	DECIMAL(10,2)	NOT NULL	Total cost for this item

5.bookings



FIELD	DATATYPE	CONSTRAINTS	DESCRIPTION
Booking_id	INT	PRIMARY KEY,AUTO_INCREMENT	Unique id for each booking
Customer_id	INT	NOT NULL,FOREIGN KEY	References the customer who made the booking
Vehicle_make	Varchar(100)	NOT NULL	Vehicle manufacturer brand(eg:Honda,Toyota)
vehicle_model	VARCHAR(100)	NOT NULL	Vehicle model (e.g, Fortuner, City)
vehicle_number	VARCHAR(50)	NULL	Vehicle registration/number plate
location_link	VARCHAR(255)	NOT NULL	Google Maps link/ customer location
service_type	VARCHAR(100)	NOT NULL	Type of service requested (e.g, Repair, Maintenance)
tow_service	TINYINT(1)	DEFAULT 0	0 = No towing, 1 = Towing required
customer_notes	TEXT	NULL	Additional details or instructions from customer
booking date	DATE	NOT NULL	Scheduled date for the service
booking time	TIME	NOT NULL	Scheduled time for the service
created_at	TIMESTAMP	DEFAULT CURRENT_TIMESTAMP	Date and time when the booking was created
status	VARCHAR(50)	NOT NULL, DEFAULT 'Pending"	Current booking status
driver_id	INT	NULL, FOREIGN KEY	Driver assigned to the booking
quotation	DECIMAL(10,2)	NULL	Estimated service price or quotation

6. drivermanage

FIELD	DATATYPE	CONSTRAINTS	DESCRIPTION
did	INT	PRIMARY KEY AUTO INCREMENT	Unique ID for each driver
drivername	VARCHAR(30)	NOT NULL	Driver's full name
phone	CHAR(10)	NOT NULL	Driver's contact number
email	VARCHAR(30)	NOT NULL	Driver's email address
password	VARCHAR(30)	NOT NULL	Driver's password (should be stored as hashed/encrypted
liscence_no	VARCHAR(30)	NOT NULL	Driver's license number
address	VARCHAR(100)	NULL	Driver's residential address
status	ENUM	DEFAULT ’available’,values (active,inactive)	Current status of the driver
Created_at	TIMESTAMP	DEFAULT CURRENT_TIMESTAMP	Timestamp when the driver record was created

7.payments

FIELD	DATATYPE	CONSTRAINTS	DESCRIPTION
Payment_id	INT	PRIMARY KEY AUTO INCREMENT	Unique ID for each payment
booking_id	INT	NOT NULL,FOREIGN KEY	References the booking for which payment was made
Customer_id	INT	NOT NULL,FOREIGN KEY	References the customer making the payment
amount_paid	DECIMAL(10,2)	NOT NULL	Amount paid by the customer
payment_method	VARCHAR(30)	NOT NULL	Payment method (e.g., Credit Card, UPI, Cash)
Transaction_id	VARCHAR(30)	NOT NULL	Transaction ID provided by the payment gateway (optional)
Payment_status	VARCHAR(30)	DEFAULT ‘Success’	Status of the payment (Success, Failed, Pendin
Payment_date	DATETIME	DEFAULT CURRENT_TIMESTAMP	Date and time when the payment was made

4.feedback

FIELD	DATATYPE	CONSTRAINTS	DESCRIPTION
Feedback_id	INT	PRIMARY KEY AUTO INCREMENT	Unique ID for each feedback
Customer_id	INT	NOT NULL, FOREIGN KEY	References the customer who gave the feedback
Booking_id	INT	NOT NULL,FOREIGN KEY	References the booking for which feedback is given
rating	TINYINT	NOT NULL	Rating provided by the customer (e.g., 1-5)
Feedback_text	VARCHAR(30)	NULL	Optional detailed feedback text
Created_at	TIMESTAMP	DEFAULT CURRENT_TIMESTAMP	Timestamp when FEEDBACK was created

5. Contact_messages

FIELD	DATATYPE	CONSTRAINTS	DESCRIPTION
id	INT	PRIMARY KEY, AUTO_INCREMENT	Unique ID for each contact message
name	VARCHAR(100)	NOT NULL	Name of the person sending the message
email	VARCHAR(100)	NOT NULL	Email address of the sender
subject	VARCHAR(200)	NOT NULL	Subject line of the message
message	TEXT	NOT NULL	Content of the message
status	ENUM	DEFAULT ‘new’	Current status of the message
created at	TIMESTAMP	NOT NULL, DEFAULT CURRENT_TIMESTAMP	Timestamp when the message was created
replied_at	TIMESTAMP	NULL, DEFAULT NULL	Timestamp when the message was replied

3.3 Object Oriented Design-UML diagrams

UML stands for Unified Modelling Language. UML is a language for specifying, visualizing and documenting the system. This is the step while developing any product after analysis. The goal from this is to produce a model of the entities involved in the project which later need to be built. The representation of the entities that are to be used in the product being developed need to be designed. Software design is a process that gradually changes as various new, better and more complete methods with a broader understanding of the of the whole problem in general come into existence. There are various kinds of methods in software design. They are as follows:

- Use case diagram
- Activity diagram
- Sequence diagram
- Class diagram

Use case Diagrams:

Use case diagrams model behavior within a system and helps the developers understand of what the user requires. The stick man represents what’s called an actor. An actor represents an outside entity- either human or technological. Use case diagrams can be useful for getting an overall view of the system and clarifying who can do and more importantly what they can’t do. Use case Diagram consists of use cases and actors and shows the interaction between the use case and actors. The purpose is to show the interactions between use cases and actor. To represent the system requirements from user’s perspective. It must be remembered that the usecases are the functions that are to be performed in the module. An actor could be the end-user of the system or an external system.

Activity Diagram:

The purpose is to show the activities which the users performed. Actives are shown parallel and sequentially in which order they are performed. Some activities are joined and split according to its flow. Flow of data is represented using arrows.

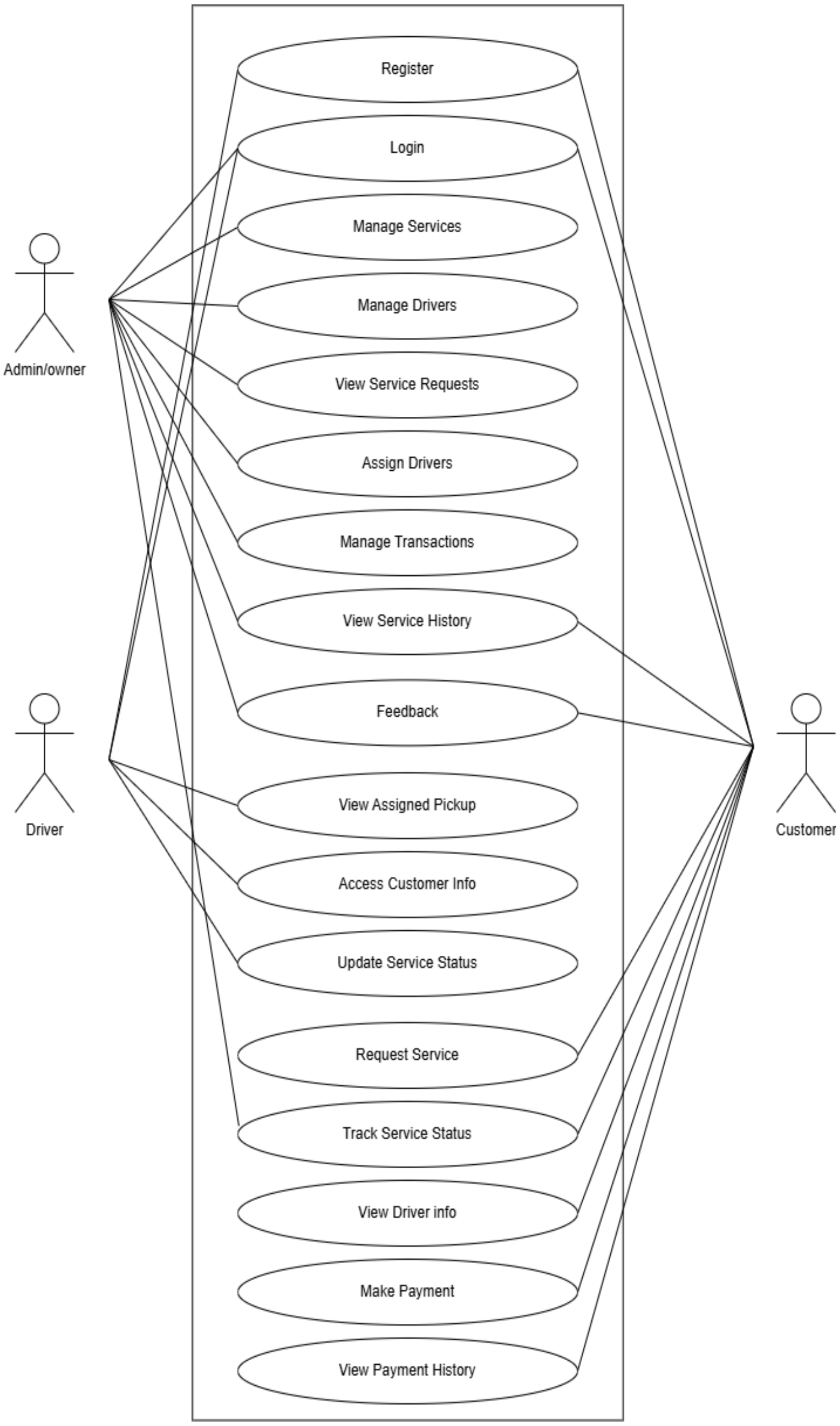
Sequence Diagram:

The purpose is to show the sequential flow through of activities. In other Words, we call it mapping processes in terms of data transfers from the actor through corresponding objects. To represent the logical flow of data with respect to a process. It must be remembered that the sequence diagram display objects and not the classes.

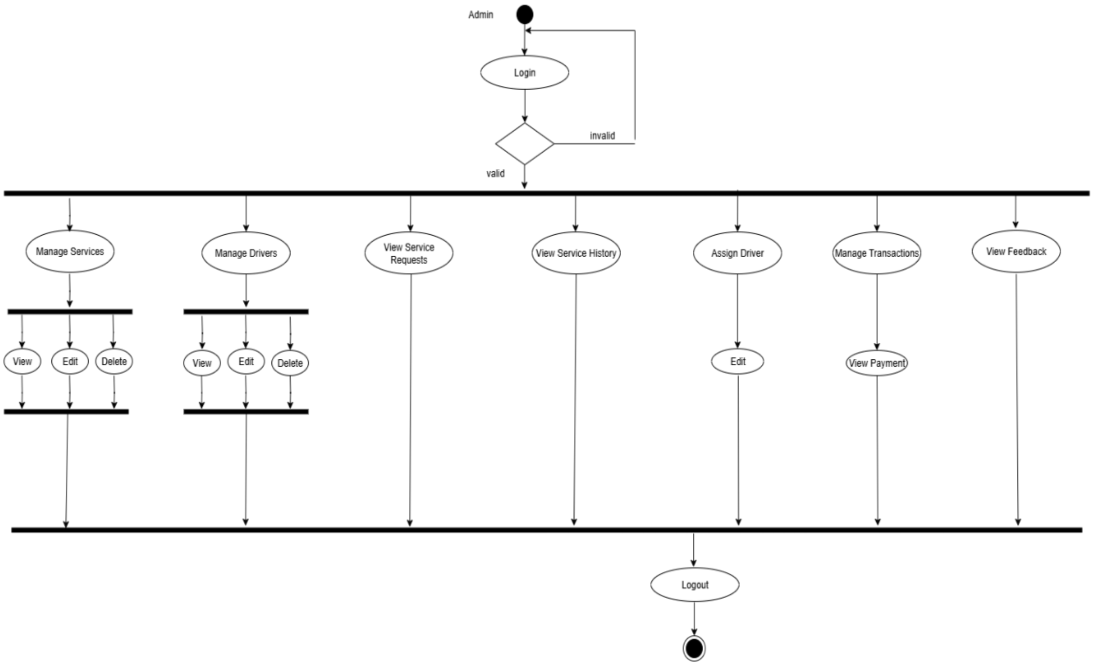
Class Diagram:

This is one of the most important of the diagrams in development. The diagram breaks the class into three layers. One has the name, the second describes its attributes and the third its methods. The private attributes are represented by a padlock to left of the name. The relationships are drawn between the classes. Developers use the Class Diagram to develop the classes. Analyses use it to show the details of the system. Architects look at class diagrams to see if any class has too many functions and see if they are required to be split.

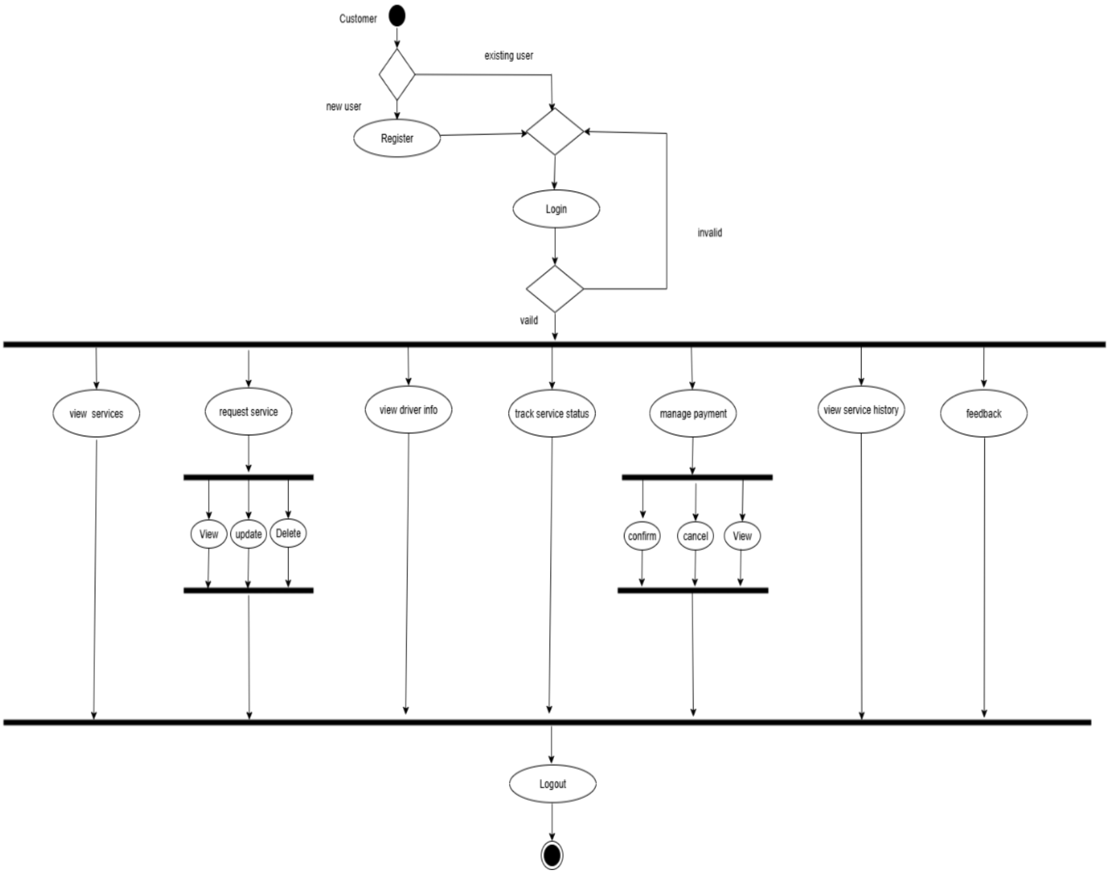
Use Case Diagram



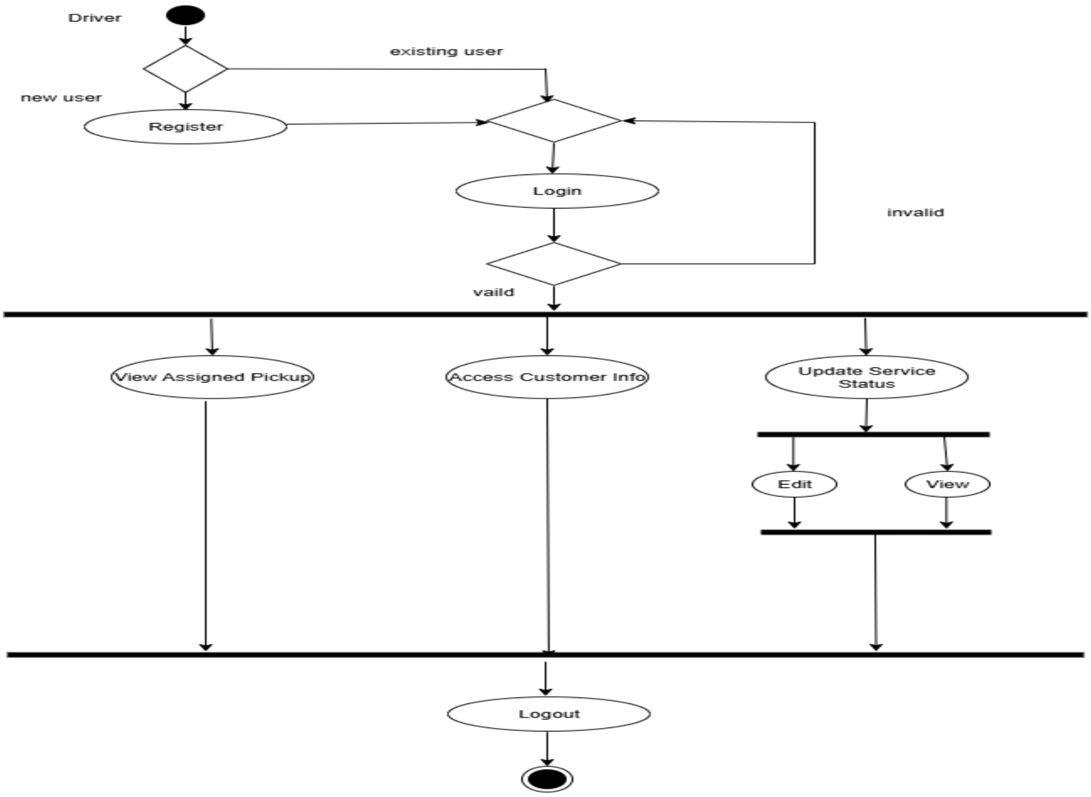
Activity Diagram : Admin



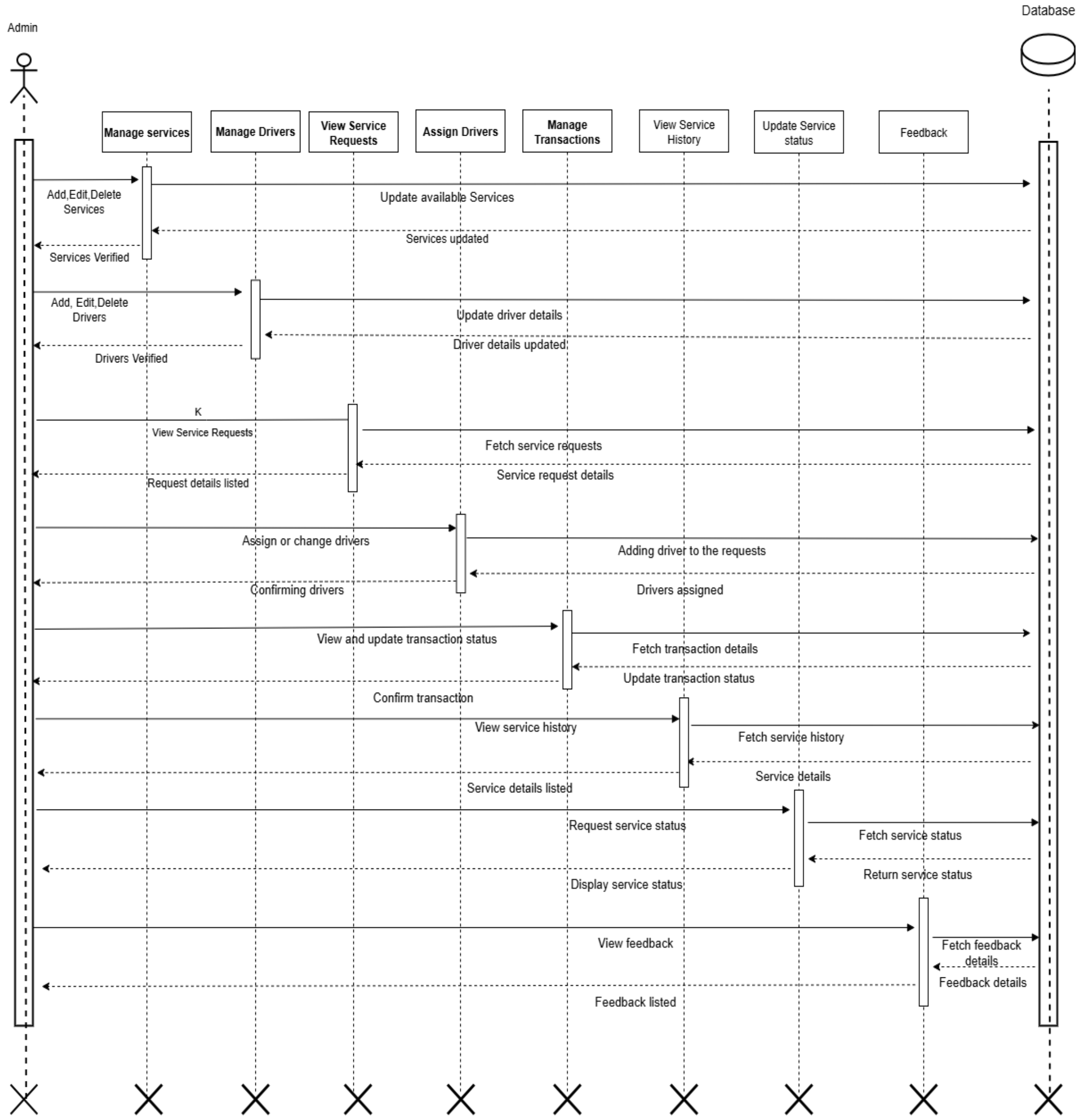
Activity Diagram : customer



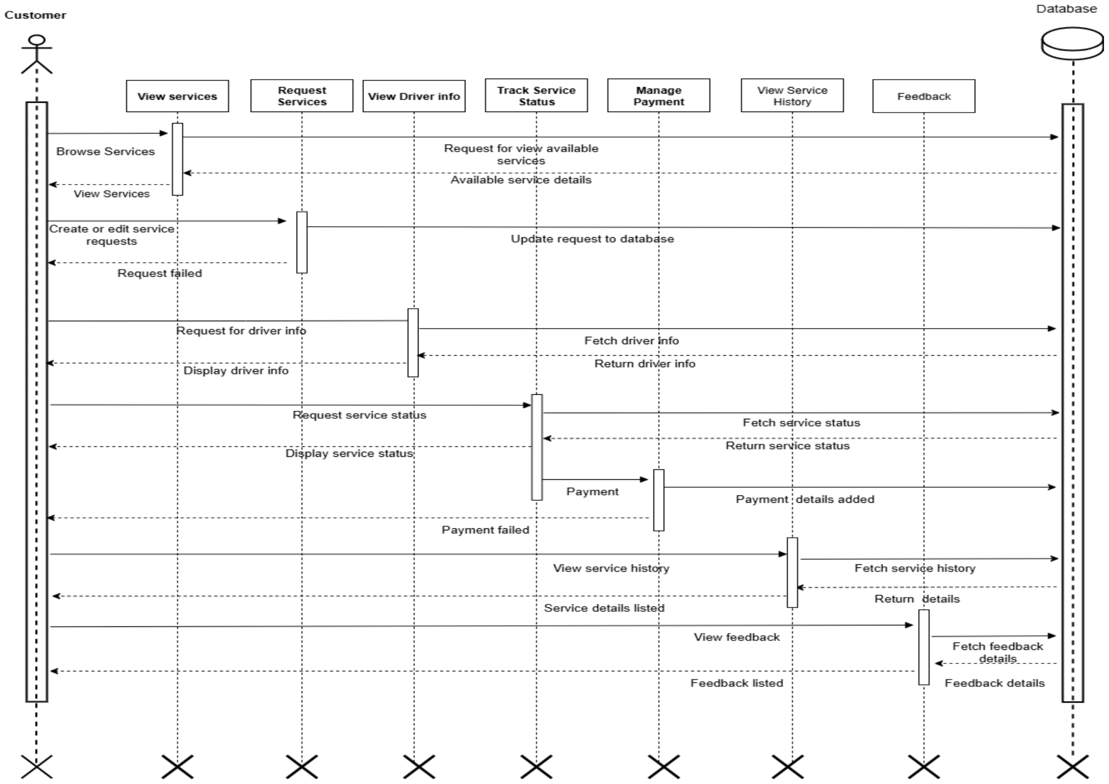
Activity Diagram : driver



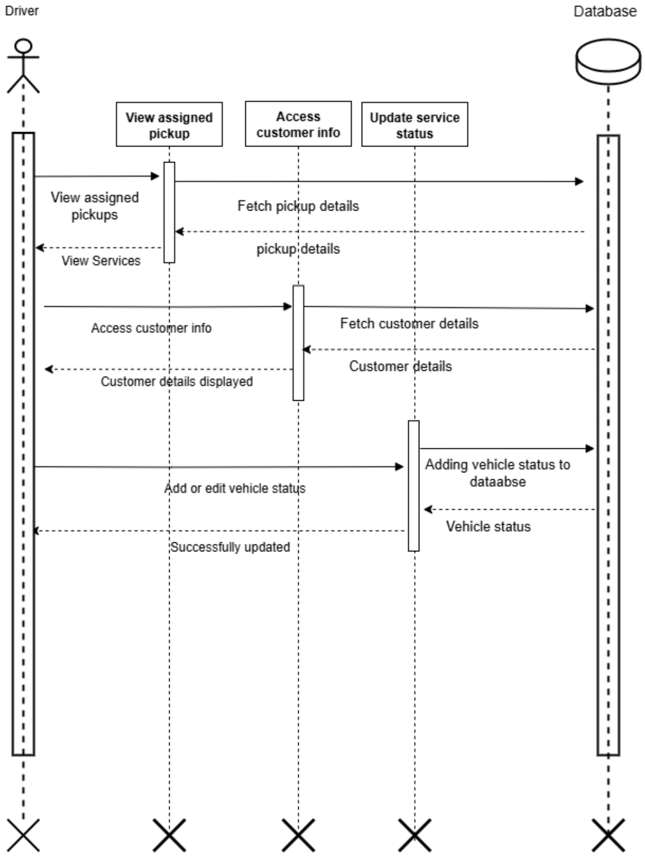
Sequence Diagram : Admin



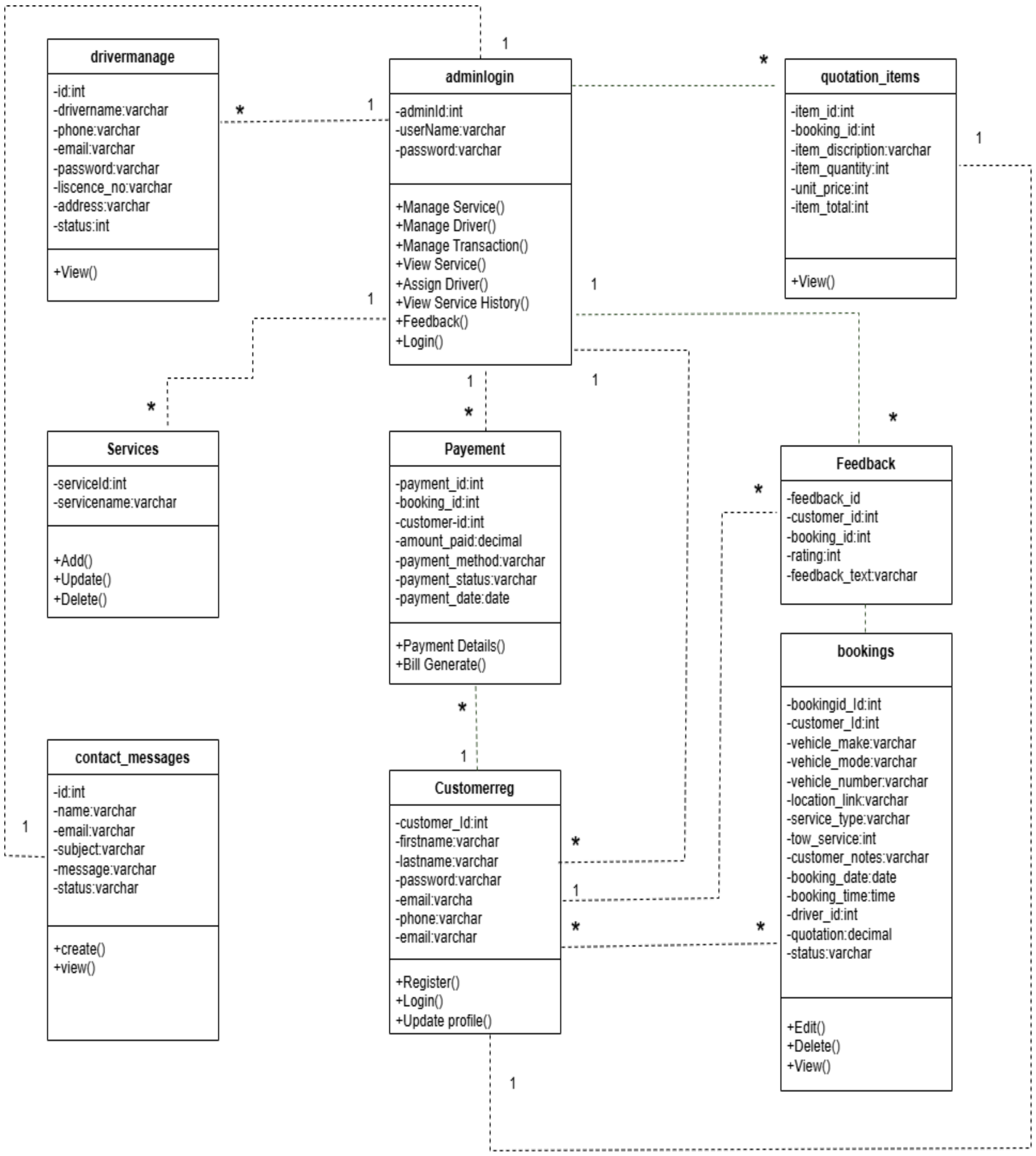
Sequence Diagram : Customer



Sequence Diagram : Driver



Class Diagram



3.4. MODULES DESCRIPTION

A software system is often divided into several subsystems to simplify development, testing, and maintenance. The process of dividing a system into smaller, manageable units is called modularization or decomposition. In the Vehicle Service Management System, the application has been divided into several modules to streamline development and ensure clear separation of responsibilities. The main modules in the system are **Admin**, **Customer**, and **Driver**, each designed to handle specific functionalities efficiently, enabling smooth operation and better user experience across the platform.

Admin Module

The **Admin** module is designed for the service center administrator, who manages all aspects of the system. Admins can add, edit, or remove services, assign drivers for pickup and delivery, and oversee all customer bookings. They can also generate detailed reports on services, drivers, and transactions for analytics purposes. The admin ensures the smooth operation of the system by monitoring bookings, tracking service completion, and managing overall workflow. This module is critical for maintaining control over services, optimizing resources, and ensuring the timely and efficient handling of customer requests.

Customer Module

The **Customer** module is intended for vehicle owners who register to use the system. Customers can browse available services with detailed descriptions and images, book appointments online, and choose preferred time slots for their vehicle maintenance. They can also track the status of their bookings in real -t ime, manage personal profiles, view service history, and provide feedback on completed services. This module ensures that customers have a convenient, transparent, and seamless experience when booking and managing vehicle services without the need for phone calls or in -person visits.

Driver Module

The **Driver** module allows drivers to manage assigned pickup and delivery tasks efficiently. Drivers can view their scheduled bookings, update service status during pickup, maintenance, or delivery, and manage their daily workflow. This module ensures proper communication between drivers, customers, and administrators, improving the coordination and timeliness of services. It also allows drivers to mark tasks as completed, report delays or issues, and maintain an accurate log of services handled. This module is essential for streamlining service logistics and ensuring customer satisfaction.

3.5 INPUT DESIGN

The input design is the link between the information system and the user. It comprises the developing specification and procedures for data preparation and those steps are necessary to put transaction data into a usable form for processing data entry. The activity of putting data into the computer for processing can be achieved by inspecting the computer to read data from a written or printed document or it can occur by having people keying the data directly into the system.

- **Login**
  Users enter login details to access their accounts. users required username and password.
- **Registration**
  Users input personal details to create a new account
- **Booking**
  Customer input booking details, such as vehicle information , required service details, dates, time slot, a nd other requirements .

Key Considerations for Input Design

Required Data: What information needs to be collected from customers, drivers, or administrators for bookings, services, and updates.

Dialogue and Guidance: Instructions to help users provide accurate input, including prompts and feedback messages.

3.6. OUTPUT DESIGN

Computer output is the most important and direct information source to the user. Output design is a process that involves designing necessary outputs in the form of reports that should be given to the users according to the requirements. Efficient, intelligible output design should improve the system's relationship with the user and help in decision making. So, while designing output the following things are to be considered.

- Determine what information to present
- Arrange the presentation of information in an acceptable format
- Decide how to distribute the output to intended receipts.

Key System Outputs

View Customers

The admin of the Vehicle Service Management System can view all registered customers who have joined the system

View Bookings

The admin can view all service bookings made by customers. This feature helps track scheduled appointments, monitor service progress, and manage workflow efficiently.

View and Manage Services

Admin can view listed services with details.

View Feedback & Reviews

Admin and sellers can view customer reviews and ratings for purchased products.

View Reports

Admin can view sales reports, order trends, and transaction summaries for decision-making.

System Environment

4.1 Introduction

It encompasses the hardware, software, data, personnel, and networks that collectively create the context within which a software system operates. Understanding and carefully managing the system environment is essential for ensuring optimal performance, reliability, and scalability of the software. This introduction provides an overview of the significance of the system environment in the realm of software projects.

4.2 Software requirement specification

Operating System : Microsoft Windows 11 Home Single

Language Front End : HTML, JavaScript, CSS

Back End : PHP,

MySQL Server :

Apache

Software Used : Visual Studio Code, WAMP

4.3 Hardware requirements specification

Processor : Intel(R) Core(TM) i5-

12450H Primary : 512.0MB RAM or

Higher Secondary : 2.0GB hard disc

or higher Monitor : CRT or TFT or

higher Keyboard : 104 K

Pointing device : 2 or 3 button mouse

Printer : Dot Matrix or Ink Jet or Laser Printer

4.4 Tools, Platforms Front end tool

The front-end of an application is distinctly human. It's what the user sees, touches and experiences. In this respect, empathy is a required characteristic of a good frontend developer. The front-end of an application is less about code and more about how a user will interpret the interface into an experience. That experience can be the difference between a billion-dollar company and complete collapse. If you were a Myspace user in 2004, you were probably content with experience.

But once you started to use Facebook, you almost certainly had a better experience. You realized that you could socialize with a simpler design, no flashing banner ads, easy-to-find friends, etc.Facebook and Myspace had a lot of differences under the hood as well(back-end), but at least part of Facebook's triumph can be attributed to a better frontend and user experience.The technologies used in front-end development commonly include:

HTML – All code in a web application is eventually translated to HTML. It's the language that web browsers understand and use to display information to users. A web developer's understanding of HTML is analogous to a carpenter's understanding of a screwdriver. It's so important and necessary that it's often assumed for employment.

CSS – By itself, HTML is quite plain. HTML does provide some basic style options, but to build a good front-end, developers must have experience with CSS. CSS provides the paint, templates, glitter, buttons, tassel, lights, and many other things that can be used to improve the presentation of a web page. CSS is so commonly used that languages have been built to make writing CSS easier. These languages – like Sass and LESS – are also known as CSS pre-compilers, but they are simply used to write more efficient and manageable CSS code.

JavaScript – If you could only learn one language in your lifetime, you'd be well advised to choose JavaScript. Though it's not exclusively a front-end language, that's where it's most commonly used. JavaScript is a language that is run on a client machine, i.e., a user's computer. This means that JavaScript can be used to program fast, intuitive and fun user experiences, without forcing a user to refresh their web page. Drag-and drop, infinite-scroll and videos that come to life on a web page can all be programmed with JavaScript. JavaScript is so popular that entire frameworks have been built just to make building application front-ends easier. Frameworks like Angular, Ember, React and Backbone are all very widely used for JavaScript-heavy front-ends.

Back end tool

The back-end of a web application is an enabler for a front-end experience. An application's front-end may be the most beautifully crafted web page, but if the application itself doesn't work, the application will be a failure. The back-end of an application is responsible for things like calculations, business logic, database interactions, and performance. Most of the code that is required to make an application work will be done on the back-end. Back-end code is run on the server, as opposed to the client. This means that back-end developers not only need to understand programming languages and databases, but they must have an understanding of server architecture as well. If an application is slow, crashes often, or constantly throws errors at users, it's likely because of back-end problems.

PHP

PHP is an acronym for "PHP: Hypertext Pre-processor". PHP is a widely-used, opensource scripting language. PHP scripts are executed on the server. PHP is free to download and use. PHP is an amazing and popular language. It is powerful enough to be at the core of the biggest blogging system on the web. It is deep enough to run the largest social network. It is also easy enough to be a beginner's first server-side language files can contain text, HTML, CSS, JavaScript, and PHP code. PHP code are executed on the server, and the result is returned to the browser as plain HTML. PHP files have extension ".php". PHP can generate dynamic page content. PHP can create, open, read, write, delete, and close files on the server. PHP can collect form data. PHP can send and receive cookies. PHP can add, delete, and modify data in your database. PHP can be used to control user-access. PHP can encrypt data. With PHP you are not limited to output HTML. You can output images, PDF files, and even flash movies.

You can also output any text, such as XHTML and XML.

MYSQL

MySQL is a popular open-source relational database management system used to build many software applications. It is developed and supported by MySQL AB, a Swedish company. This tutorial provides a quick introduction to MySQL and helps you get started with MySQL programming.

WAMPSERVER

WampServer is a Web development platform on Windows that allows you to create dynamic Web applications with Apache2, PHP, MySQL and MariaDB. WampServer automatically installs everything you need to intuitively develop Web applications. You will be able to tune your server without even touching its setting files.

Operating system

WINDOWS 11

Windows 11 is the latest major release of Microsoft's Windows operating system, launched in October 2021 as a free upgrade for eligible Windows 10 devices. It brings a redesigned interface inspired by the cancelled Windows 10X, including a centered Start menu, a new widget panel, and enhanced window management features like Snap Layouts and Snap Groups. Windows 11 also introduces gaming improvements such as Auto HDR and Direct Storage, similar to Xbox Series consoles. Internet Explorer is replaced by the Chromium-based Microsoft Edge, and Microsoft Teams is integrated into the system

The Microsoft Store has been updated to support more types of apps, including Android applications through a partnership with Amazon. Windows 11 has stricter hardware requirements, including TPM 2.0 and newer CPU models, to improve system security. Although it can run on unsupported hardware, updates are not guaranteed. Overall, Windows 11 received mixed feedback: users appreciated the new design and performance improvements but criticized the strict hardware requirements and some interface changes

System Implementation

5.1 Introduction

It is a critical phase in the software development life cycle where the designed system is put into action. It involves translating the specifications and plans developed during earlier stages into a functioning and operational software system. This process encompasses the actual coding, configuration, testing, and deployment of the software. In this introduction, we will delve into the significance of system implementation and provide a general description of the key activities involved.

5.2 Coding Sample codes home page

```
<?php
session_start();
$logged_in = isset($_SESSION['customer_id']);
// Database connection
require_once '../includes/db_config.php';
// Initialize variables
$name = $email = $subject = $message = '';
$form_success = $form_error = '';
// Handle form submission
if ($_SERVER['REQUEST_METHOD'] == 'POST' && isset($_POST['name'],
$_POST['email'], $_POST['subject'], $_POST['message'])) {
$name = trim($_POST['name']);
$email = trim($_POST['email']);
$subject = trim($_POST['subject']);
$message = trim($_POST['message']);
// Validate inputs
if (empty($name) || empty($email) || empty($subject) || empty($message)) {
$form_error = "All fields are required.";
} elseif (!filter_var($email, FILTER_VALIDATE_EMAIL)) {
$form_error = "Please enter a valid email address.";
} else {
// Save to database
try {
$pdo = getDBConnection();
$stmt = $pdo->prepare("INSERT INTO contact_messages (name, email,
subject, message, status, created_at) VALUES (?, ?, ?, ?, 'new', NOW())");
$stmt->execute([$name, $email, $subject, $message]);
if ($stmt->rowCount() > 0) {
$form_success = "Thank you for contacting us! We'll get back to you soon.";
// Clear form fields
$name = $email = $subject = $message = '';
} else {
$form_error = "Sorry, there was an error sending your message. Please try
again.";
}
```

```
} catch (PDOException $e) {
$form_error = "Sorry, there was an error sending your message. Please try
again.";
        // Function to get services from database function getServices() { try
 { $pdo = getDBConnection(); $stmt = $pdo->prepare("SELECT * FROM
 services WHERE status 'active' OR
 status = '1' ORDER BY servicename");
 $stmt->execute();
 return $stmt->fetchAll();
 } catch (PDOException $e) {
 return [];
 }
 }
 // Get services for display
 $services = getServices();
 ?>
 <!DOCTYPE html>
 <html lang="en">
 <head>
 <meta charset="utf-8">
 <meta content="width=device-width, initial-scale=1.0" name="viewport">
 <title>AutoFix - Premium Vehicle Services</title>
 <meta name="description" content="Professional vehicle maintenance and repair
 services with 25 years of excellence">
 <meta name="keywords" content="vehicle service, car repair, maintenance,
 denting, painting, auto service">
 </head>
 <body>
 <!-- Header -->
 <header>
 <div>
 <a href="index.php">
 AutoFixDe Paul Institute of Science & Technology (DiST)
 </a>
 <ul>
 <li><a href="#hero">Home</a></li>
 <li><a href="#about">About</a></li>
 <li><a href="#services">Services</a></li>
 <li><a href="#contact">Contact</a></li>
 </ul>
 <?php if ($logged_in): ?>
 <a href="../profile.php">My Profile</a>
 <?php else: ?>
 <a href="../customerlogin.php">Login</a>
 <?php endif; ?>
 </div>
 </header>
 <main>
 <!-- Hero Section -->
```

```
<section id="hero">
<h1>Premium Vehicle Service &Maintenance</h1>
<p>We Take Care of Your Wheels with 25 Years of Excellence</p>
<?php if ($logged_in): ?>
<a href="../bookingfor.php">Book Service Now</a>
<?php else: ?>
<a href="../customerlogin.php">Login to Book</a>
<a href="../registeration2.php">Register Now</a>
<?php endif; ?>
</section>
<!-- About Section -->
<section id="about">
<h2>About Our Service</h2>
<p>A Reputation of 25 Years in the Automotive Industry</p>
<h3>Our Mission</h3>
<p>Our mission is to enable premium quality care for your luxury car service at
affordable pricing. We ensure real time updates for complete car care needs with a
fair and transparent pricing mechanism.</p>
<p>Skilled technicians, working at our state of the art German technology
workshop, ensure that only genuine OEM parts are used for your car care
needs.</p>
<h3>Perfect Service Partner</h3>
<p>Customer satisfaction is the core of all initiatives at AutoFix. Online
appointment scheduling with doorstep, same day pickup and drop anywhere in
Pune is our constant endeavor to maximize customer convenience.</p>
<p>Our commitment stands for reliability and unequalled professionalism to
provide dealer quality auto-service at a fair price.</p>
<img src="../images/about 2.jpg" alt="Our Team">
</section>
<!-- Services Section -->
<section id="services">
<h2>Our Services</h2>
<p>Comprehensive Vehicle Care Solutions</p>
<?php if (!empty($services)): ?>
<?php foreach ($services as $service): ?>
<div>De Paul Institute of Science & Technology (DiST)
<?php
// Determine service image
if (!empty($service['image']) && file_exists('../../uploads/services/' .
$service['image'])){
$image = '../../uploads/services/. $service['image'];
} else {
$serviceName = strtolower($service['servicename']);
$image = "../images/car-painting-spray.jpg";
} elseif (strpos($serviceName, 'major') !== false) {
$image = "../images/major.jpeg";
} elseif (strpos($serviceName, 'detail') !== false |
| strpos($serviceName, 'wash') !== false) {
```

```
$image = "../images/carwash.jpg";
} elseif (strpos($serviceName, 'ac') !== false
|| strpos($serviceName, 'air') !== false) {
$image = "../images/car ac.jpg";
} elseif (strpos($serviceName, 'tire') !== false) {
$image = "../images/tire-maintenance.jpg";
} elseif (strpos($serviceName, 'ceramic') !== false
|| strpos($serviceName, 'coat') !== false) {
$image = "../images/ceramic coating.jpeg";
} else {
$image = "../images/periodic.jpeg";
}
}
? <img src="<?php echo $image; ?>"
alt="<?php htmlspecialchars($service['servicename']); ?>">
<h3><?php echo htmlspecialchars($service['servicename']); ?></h3>
<?php endforeach; ?>
<?php else: ?>
<div>
<h3>Periodic Services</h3>
<p>Stop waiting for a problem to happen. Be proactive with your vehicle's
health. Book your next periodic service with us.</p>
</div>
<?php endif; ?>
</section>
<!-- Contact Section -->
<section id="contact">
<h2>Contact Us</h2>
<p>Get in Touch with Our Team</p>
<div>
<div>
<h3>Our Location</h3>
<p>Aluva To Nedumbassery Airport Road<br>Opposite
Federal Bank<br>PIN: 535022</p>
</div>
<div>
<h3>Call Us</h3>
<p>+91 9961248888</p>
</div>
<div>
<h3>Email Us</h3>
</div>
<div>
<?php if ($form_success): ?>
<?php endif; ?>
<?php if ($form_error): ?>
<p><?php echo $form_error; ?></p>
<?php endif; ?>
<form action="#contact" method="post">
```

```
<input type="text" name="name" placeholder="Your Name" value="
<?php echo htmlspecialchars($name); ?>" required>
<input type="email" name="email" placeholder="Your Email" value="
<?php echo htmlspecialchars($email); ?>" required>
<input type="text" name="subject" placeholder="Subject" value="
<?php echo htmlspecialchars($subject); ?>" required>
<textarea name="message" placeholder="Your Message" required>
<?php echo htmlspecialchars($message); ?></textarea>
<button type="submit">Send Message</button>
</form>
</div>
</section>
</main>
<!-- Footer -->
<footer>
<h4>AutoFix</h4><li><a href="#about">About Us</a></li>
<li><a href="#services">Services</a></li>
<li><a href="#contact">Contact</a></li>
<?php if ($logged_in): ?>
<li><a href="../profile.php">My Profile</a></li>
<?php else: ?>
<li><a href="../customerlogin.php">Login</a></li>
<?php endif; ?>
</div><div>
<h4>Our Services</h4>
<ul>
<?php if (!empty($services)): ?>
<?php foreach ($services as $service): ?>
<li><?php echo htmlspecialchars($service['servicename']); ?></li>
<?php endforeach; ?>
<?php else: ?>
< <li>Denting & Painting</li>
</ul>
</div>
</footer>
</body>
</html>
$sql="SELECT * FROM admin WHERE userid='$userid' AND password='$password'
";
$rlt= mysqli_query($con,$sql);
if($rlt)
{
if(mysqli_num_rows($rlt)>0)
{
session_start();
$_SESSION['is login']=true;
$_SESSION['userid']=$userid;
header("Location: admin.php");
}
else
{
echo "Invalid username or Password";
}
}
}
?>
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<meta http-equiv="X-UA-Compatible" content="IE=edge">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>login form design</title>
<style>
<?php include "adminlogin.css" ?>
</style>
<script type="text/javascript" >
function preventBack(){window.history.forward();}
setTimeout("preventBack()", 0);
window.onunload=function(){null};
</script>
</head>
<body>
<div class="hero">
<div class="form-box">
<div class="button-box">
<div class="btnn"></div>
<button type="button" class="btn" onclick="Login()">Log In</button>
</div>
<div class="social-icons">
<img src="fb.png">
<img src="tw.png">
<img src="gp.png">
</div>
<form id="Login" class="input-group" method="POST">
<input type="text" name="userid" class="input-field" placeholder="User Id"
required>
<input type="password" name="password" class="input-field"
placeholder="Enter password" required>
<input type="checkbox" class="check-box"><span>Remember
Password</span>
<button type="submit" name="login" class="submit-btn">Log
In</button>
</form>
</div>
</div>
<script>
var x = document.getElementById("Login");
var y = document.getElementById("Register");
var z = document.getElementById("btn");
function Register()
{
x.style.left = "-400px";
y.style.left = "50px";
z.style.left = "110px";
}
function Login()
{
x.style.left = "50px";
y.style.left = "450px";
z.style.left = "0";
}
</script>
</body>
</html>
```

Login page

```
<?php
session_start();
$login_error = $_SESSION['login_error'] ?? null;
unset($_SESSION['login_error']);
?>
<!DOCTYPE html>
<html lang="en">
<head>
   <meta charset="UTF-8">
   <meta name="viewport" content="width=device-width, initial-scale=1.0">
   <title>Green Basket - Log In</title>
   <link
href="https://fonts.googleapis.com/css2?family=Poetsen+One&family=Inter:wght@400;7
0 0&display=swap" rel="sty
   <style>
          * { margin: 0; padding: 0; box-sizing: border-box; font-family: 'Inter', sans-
          serif; } body { background: #f7f7f7; display: flex; justify-content: center; align-
          items: center;
    min-height: 100vh; }
          .form-container {
             background:
             rgba(255,255,255,0.95); padding:
             40px 30px;
             border-radius: 12px;
             box-shadow: 0 10px 25px
             rgba(0,0,0,0.15); width: 100%;
             max-width: 400px;
             animation: fadeIn 0.8s ease forwards;
          }
```

```
      .auth-form h2 {
         font-family: 'Poetsen One', sans-
         serif; font-size: 2rem;
         text-align: center;
         background: linear-gradient(90deg, #3fd734, #0c842e);
         -webkit-background-clip: text;
         -webkit-text-fill-color:
         transparent; margin-bottom:
         10px;
      }
      label-text { display: block; margin-bottom: 5px; font-weight: 600; color: #333; }
      input[type="text"], input[type="password"] {
         width: 100%; padding: 10px 15px; border-radius: 8px; border: 1.5px
 solid #ccc; font-size: 1rem; transition: all 0.3s ease;
      }
      input:focus { border-color: #2eb535; box-shadow: 0 0 5px
 rgba(46,181,53,0.3); outline: none; }
      .submit-button {
         width: 100%;
         padding: 12px;
         border: none;
         border-radius:
         25px;
         background: linear-gradient(90deg, #1dcc14,
         #38dc2c); color: white;
         font-size:
         1.1rem; font-
         weight: 600;
         cursor: pointer;
         transition: all 0.4s
         ease; margin-top:
         10px;
      }
      .submit-button:hover { background: linear-gradient(270deg, #1dcc14, #38dc2c);
 transform: scale(1.05); }
         from { opacity: 0; transform: translateY(20px); }

 </style>
</head>
<body>
<div class="form-container">
   <form class="auth-form" id="loginForm" action="../session/session.php"
method="POST" novalidate>
      <h2>Welcome Back</h2>
      <p class="subtitle">Log in to access your Green Basket account.</p>

<?php if ($login_error): ?>
<div class="error-message"><?= htmlspecialchars($login_error) ?></div>
<?php endif; ?>
```

```
<label for="username">
<span class="label-text">Username</span>
        <input type="text" id="username" name="username" placeholder="Enter your
username" required>
</label>

<label for="password">
<span class="label-text">Password</span>
        <input type="password" id="password" name="password" placeholder="Enter
your password" required>
</label>
<button type="submit" class="submit-button" name="login">Log In</button>
<p class="form-footer">
Don’t have an account? <a href="register.php">Register here</a>
</p>
 </form>
</div>
<script>
 document.getElementById('loginForm').addEventListener('submit',
function(e) { const username =
document.getElementById('username').value.trim();
const password =
document.getElementById('password').value.trim(); let errorMsg =
'';

if (username.length < 3) {
errorMsg += 'Username must be at least 3 characters long.\n';
}
if (password.length < 6) {
errorMsg += 'Password must be at least 6 characters long.\n';
}

 if (errorMsg) {
e.preventDefault();
alert(errorMsg);
}
});
</script>

</body>
</html>
```

System Testing

6.1 Introduction

System testing involves both verification and validation, serving as a crucial quality control measure during software development. Given the complexity of large systems, testing is broken down into smaller activities. Each module in the Green Basket was tested individually (unit testing) to detect and fix coding errors. Special test data was used as input, and the output was examined to ensure the system functions as specified and meets user expectations. This dynamic testing approach identifies errors but does not always pinpoint their exact cause.

The testing process is guided by the requirements document, with the goal of ensuring the software meets the specified requirements. Testing focuses on the system's external behavior rather than its internal logic. For the Green Basket, the entire system was divided into smaller modules, each tested individually. Afterward, these modules were integrated and tested as a whole. Finally, the entire system was tested using real-world data to identify any remaining errors.

- Unit testing
- Integration testing
- Validation testing
- Alpha testing
- Beta testing

6.2 Unit testing

This is the first level of testing. A unit testing focuses verification effort on the smallest unit of software design. This testing is carried out during the coding itself. In this testing step, each module is tested, for example in the registration password and login. Module interface is tested to ensure that the information properly flow and out of the program under test. Data consistency is tested to ensure that the data stored maintains its integrity during all steps in algorithm execution. All independent paths are examined to ensure that all the statement in the module have been executed at least once.

6.3 Integration Testing

Integration testing is a systematic test for constructing the program structure while conducting tests to uncover errors related to interfacing. The objective is to take unit tested modules and build a program structure that has been dictated by design. The data entered in the front end is successfully stored in the back end. For example, in the registration they entered username and password are successfully stored in the back end. This testing is conducted in the administrator module and user module.

6.4 System testing

System Testing is a crucial phase in the software testing life cycle where the entire software system is tested as a whole to ensure that it meets the specified requirements. This testing phase comes after integration testing and before acceptance testing. The primary objective is to verify the functional and non-functional aspects of the system in an integrated environment.

By systematically testing the integrated system under various conditions, organizations can identify and rectify issues, ensuring a robust and reliable software product. Successful system testing contributes to a smoother acceptance testing phase and, ultimately, the successful deployment of the software.

System maintenance

7.1 Introduction

Even though the changeover of the system is fully correct and complete, it is not the end of the matter. The system should be given proper security and maintenance in order to keep them efficient and up-to-date.

7.2 Maintenance

This software can be modified as need occurs. Maintenance includes all the activities after installation of the software that is performed to keep the system operational. The process of maintaining involves:

- Understanding the existing software
- Understand the effect of change
- Test for satisfaction

This system can be easily maintained and updated by adding new requirements or features as the platform grows, such as new product categories, payment options, or delivery features. Maintenance also helps fix unexpected bugs or issues that may appear after deployment, ensuring smooth and reliable operation. Since system upkeep is an ongoing process, it becomes one of the most important phases in the project lifecycle. Security plays a key role in maintenance — user accounts, order data, and payment information must remain protected. Role-based access, secure authentication, and restricted admin controls help prevent unauthorized access, while regular backups protect data from loss or system failures. Overall, this system is built to be scalable, secure, and easy to enhance in the future.

Future enhancement and scope of further development

8.1 Introduction

The developed system is feasible and changes can be made easily. The system is done with an insight into the necessary that be required in the future. Hence, the system can be maintained successfully without much rework. In the future the system can be further modified by including more features very easily.

8.2 Merits of the system

Including more features very easily. The future trends and development that can occur with regard to this project have been taken into consideration while designing the project. The project is developed using PHP technology, which is the latest in the present scenario and hence upgrading the project would be easier in the future.

8.3 Limitations of the system

The current system is limited to the basic service operations related to vehicle service management. It does not support additional features such as migration assistance, scholarship support, or financial assistance for activities like educational loans. These services are outside the current scope and can be considered for future upgrades.

8.5 Future Enhancement of the system

Although all essential functionalities have been implemented, there is still room for enhancement. As technology evolves rapidly, the system may gradually become outdated unless updated regularly. For the time being, minor modifications and adjustments can address immediate requirements. However, additional features can be added in the future to improve efficiency, performance, and user experience. The current system provides a strong foundation for introducing more advanced modules and expanding its capabilities.

Conclusion

CONCLUSION

The *Vehicle Service Management System* successfully integrates various aspects of vehicle servicing into a single, unified digital platform. By automating tasks such as service booking, customer management, service history tracking, mechanic assignment, and billing, the system significantly reduces manual workload and the chances of human error. It ensures faster operations, improved accuracy, and enhanced coordination among administrators, technicians, and customers.

Through its user-friendly interface, customers can easily schedule services, track progress, and receive updates, improving their overall service experience. For service centers, the system provides better control over resource allocation, real-time monitoring of ongoing tasks, and efficient record-keeping, ultimately leading to better decision-making and operational efficiency.

Overall, the Vehicle Service Management System enhances transparency, reliability, and productivity in vehicle service operations. It not only streamlines day-to-day activities but also sets a strong foundation for future scalability and integration with advanced technologies such as IoT-based vehicle diagnostics and automated inventory management. This project demonstrates the potential of digital transformation in modernizing vehicle service workflows and delivering a more efficient and customer-centric service environment.

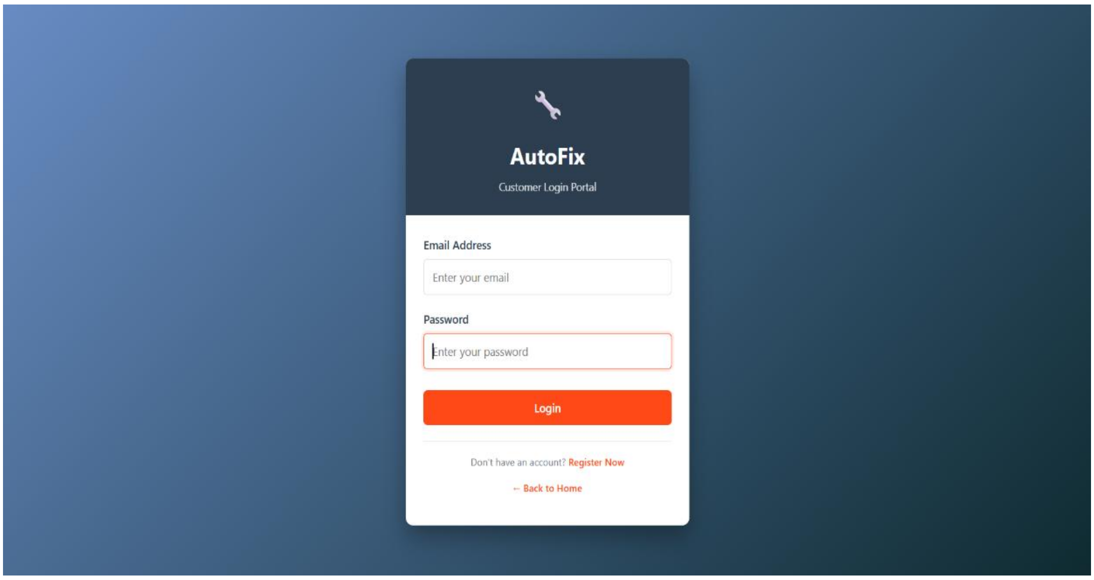
Bibliography

REFERENCES

- OpenAI ChatGPT – Utilized for programming assistance, database model guidance, documentation structuring, and debugging support during system development.
- Google Gemini – Referred for interface design ideas, feature planning suggestions, and supplementary coding guidance.
- W3Schools (www.w3schools.com) – Used as an online resource for learning and reference in HTML, CSS, JavaScript, and web development fundamentals.
- Figma – Applied for conceptualizing and designing the user interface layouts and wireframes.
- Google Search – Consulted for general technical research, best development practices, and verification of implementation approaches

Appendix

SCREENSHOT



AutoFix

Customer Login Portal

Email Address

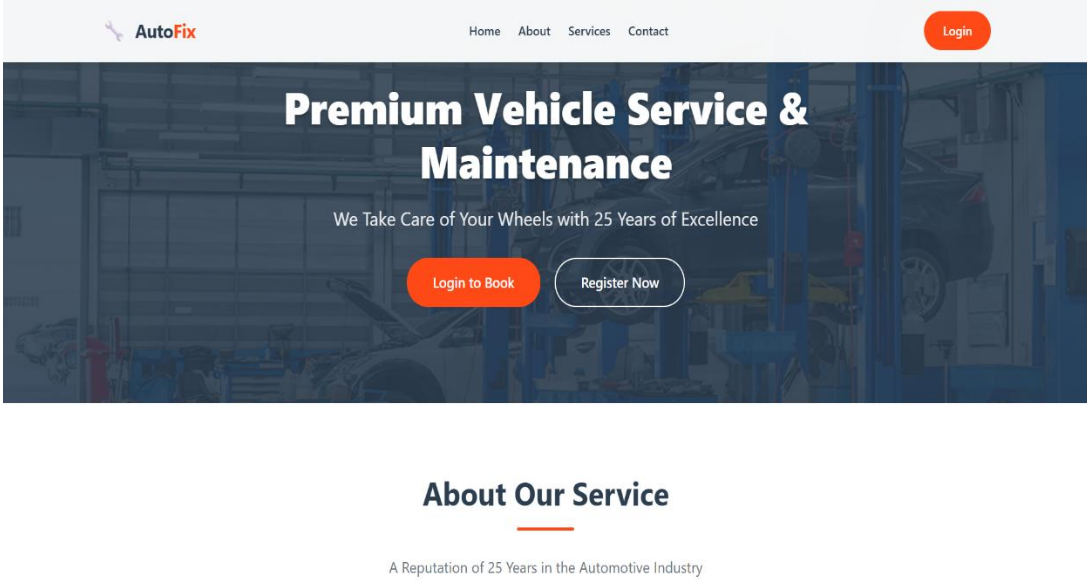
Enter your email

Password

Enter your password

Don't have an account? Register Now

← Back to Home

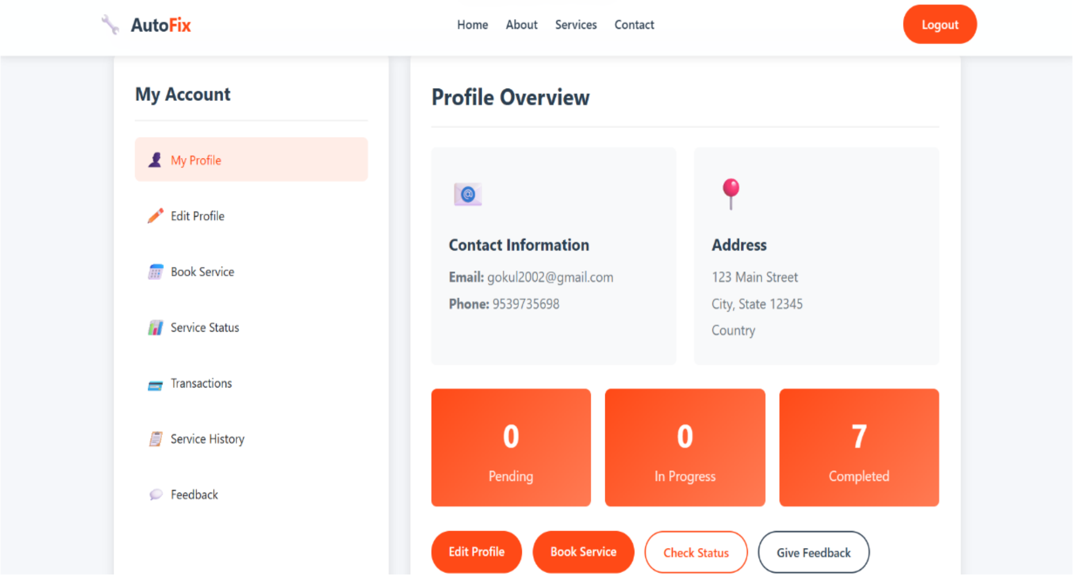


Premium Vehicle Service & Maintenance

We Take Care of Your Wheels with 25 Years of Excellence

About Our Service

A Reputation of 25 Years in the Automotive Industry



AutoFix

Home About Services Contact

Logout

My Account

- My Profile
- Edit Profile
- Book Service
- Service Status
- Transactions
- Service History
- Feedback

Profile Overview

Contact Information

Email: gokul2002@gmail.com

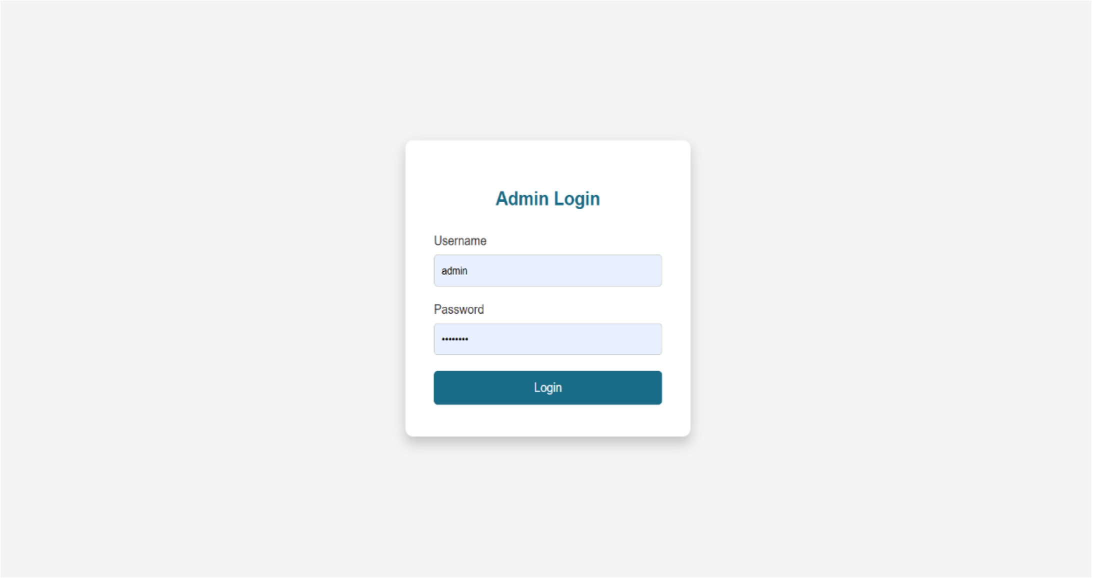
Phone: 9539735698

Address

123 Main Street
City, State 12345
Country

0	0	7
Pending	In Progress	Completed

Edit Profile Book Service Check Status Give Feedback

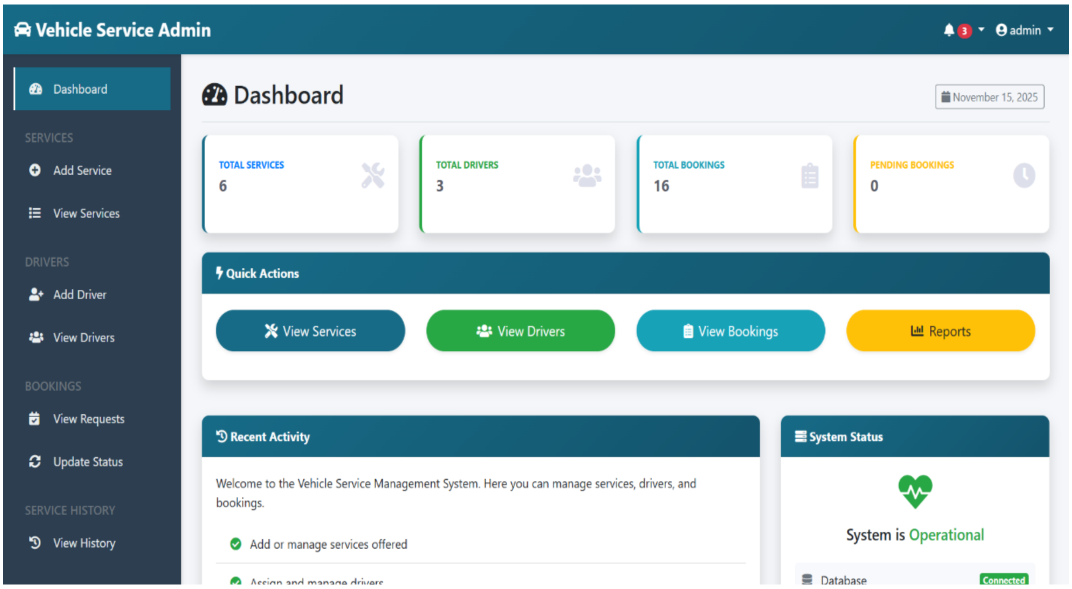


Admin Login

Username: admin

Password: ••••••••

Login



Dashboard

November 15, 2025

TOTAL SERVICES 6

TOTAL DRIVERS 3

TOTAL BOOKINGS 16

PENDING BOOKINGS 0

Recent Activity

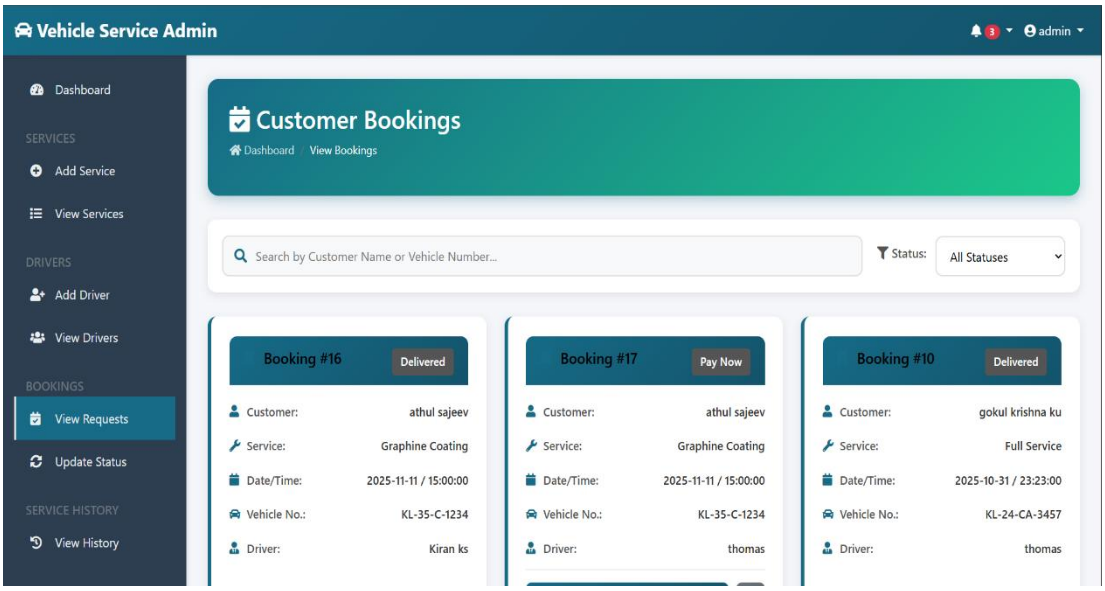
Welcome to the Vehicle Service Management System. Here you can manage services, drivers, and bookings.

- Add or manage services offered
- Assign and manage drivers

System Status

System is Operational

Database Connected



Customer Bookings

Search by Customer Name or Vehicle Number...

Status: All Statuses

Booking #16 Delivered

Customer: athul sajeev

Service: Graphine Coating

Date/Time: 2025-11-11 / 15:00:00

Vehicle No.: KL-35-C-1234

Driver: Kiran ks

Booking #17 Pay Now

Customer: athul sajeev

Service: Graphine Coating

Date/Time: 2025-11-11 / 15:00:00

Vehicle No.: KL-35-C-1234

Driver: thomas

Booking #10 Delivered

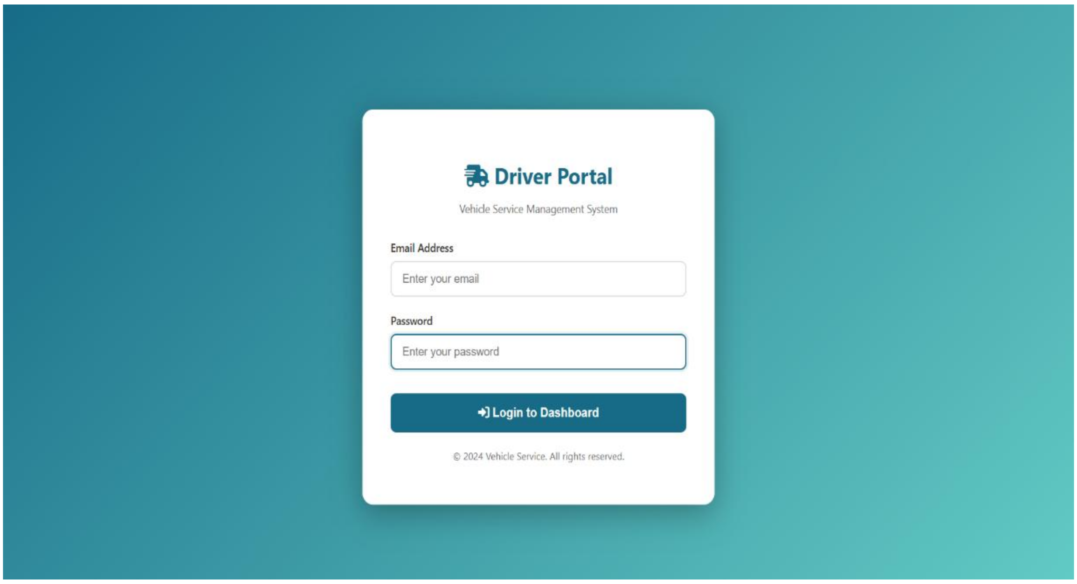
Customer: gokul krishna ku

Service: Full Service

Date/Time: 2025-10-31 / 23:23:00

Vehicle No.: KL-24-CA-3457

Driver: thomas



Driver Portal

Vehicle Service Management System

Email Address

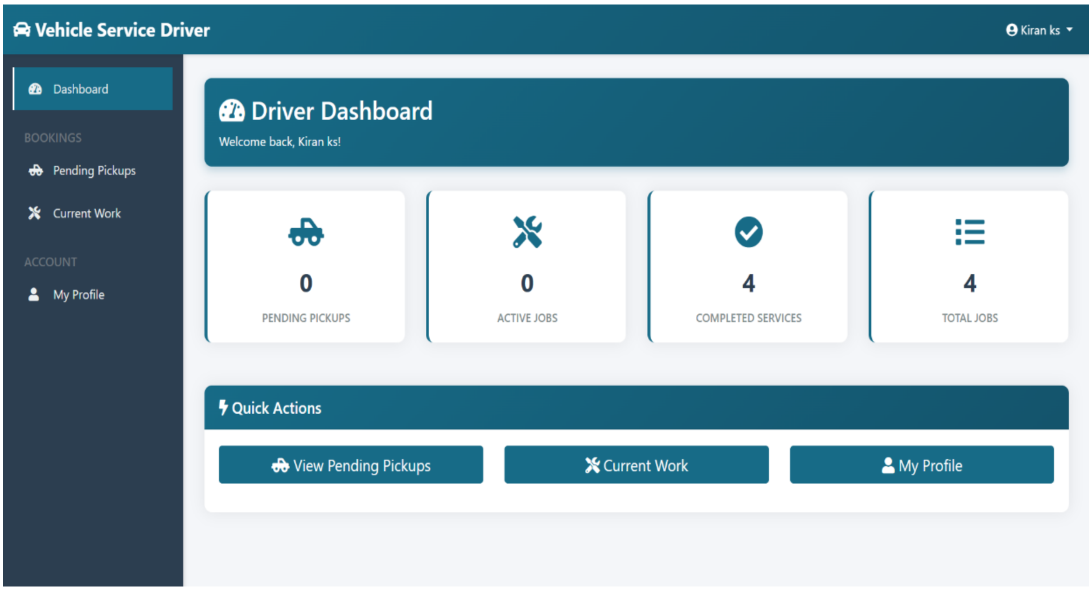
Enter your email

Password

Enter your password

Login to Dashboard

© 2024 Vehicle Service. All rights reserved.



Vehicle Service Driver

Kiran ks

Dashboard

BOOKINGS

Pending Pickups

Current Work

ACCOUNT

My Profile

Driver Dashboard

Welcome back, Kiran ks!

0	0	4	4
PENDING PICKUPS	ACTIVE JOBS	COMPLETED SERVICES	TOTAL JOBS

Quick Actions

View Pending Pickups

Current Work

My Profile

Code

```
<?php
require('../config/autoload.php');

// Set page title
$page_title = "Admin Dashboard";

// Include header and status styles
include("includes/admin_header.php");

// Initialize database access
$dao = new DataAccess();

// Get all data first using the working approach
$all_services = $dao->getData('*', 'services', '1');
all_services = is_array($all_services) ? $all_services : [];
$total_services_count = count($all_services);

$all_drivers = $dao->getData('*', 'drivermanage', '1');
$all_drivers = is_array($all_drivers) ? $all_drivers : [];
$total_drivers_count = count($all_drivers);

$all_bookings = $dao->getData('*', 'bookings', '1');
$all_bookings = is_array($all_bookings) ? $all_bookings : [];
$total_bookings_count = count($all_bookings);

pending_bookings_data = $dao->getData('*', 'bookings', "status='pending' OR
status='Pending'");
$pending_bookings_data = is_array($pending_bookings_data)
 ? $pending_bookings_data : [];
$pending_bookings_count = count($pending_bookings_data);

// Get system status information
$system_status = 'Operational';
$system_status_class = 'success';
$database_status = 'Connected';
$database_status_class = 'success';

// Check database connection
$conn = new mysqli($DB_HOST, $DB_USER, $DB_PASSWORD, $DB_NAME);
if ($conn->connect_error) {
    $database_status = 'Connection Failed';
  $database_status_class = 'danger';
   $system_status = 'Degraded';
   $system_status_class = 'warning';
}
$conn->close();

// Get additional system info
$total_customers_result = $dao->getData('*', 'customerreg', '1');
$total_customers_result = is_array($total_customers_result) ? $total_customers_result : [];
$total_customers = count($total_customers_result);
```

```
$active_drivers_result = $dao->getData('*', 'drivermanage', "status='available'");
$active_drivers_result = is_array($active_drivers_result) ? $active_drivers_result : [];
$active_drivers = count($active_drivers_result);
?>

<div class="d-flex justify-content-between flex-wrap flex-md-
nowrap align-items-center pb-2 mb-3 border-bottom">
  <h1 class="h2"><i class="fas fa-tachometer-alt"></i> Dashboard</h1>
  <div class="btn-toolbar mb-2 mb-md-0">
    <div class="btn-group mr-2">
      <button type="button" class="btn btn-sm btn-outline-secondary">
        <i class="fas fa-calendar"></i> <?= date('F d, Y') ?>
      </button>
    </div>
  </div>
</div>

<!-- Enhanced Statistics Cards with Icons and Animations -->
<div class="row">
    <!-- Total Services -->
    <div class="col-xl-3 col-md-6 mb-4">
        <div class="card border-left-primary shadow h-100 py-2 card-hover">
            <div class="card-body">
                <div class="row no-gutters align-items-center">
                    <div class="col mr-2">
                        <div class="text-xs font-weight-bold text-primary text-uppercase mb-1">Total
Services</div>
                        <div class="h5 mb-0 font-weight-bold text-gray-800">           <?=
$total_services_count ?? 0 ?></div>
                    </div>
                    <div class="col-auto">
                        <i class="fas fa-tools fa-2x text-gray-300"></i>
                    </div>
                </div>
            </div>
        </div>
    </div>

    <!-- Total Drivers -->
    <div class="col-xl-3 col-md-6 mb-4">
        <div class="card border-left-success shadow h-100 py-2 card-hover">
            <div class="card-body">


                <div class="row no-gutters align-items-center">
                    <div class="col mr-2">
                        <div class="text-xs font-weight-bold text-success text-uppercase mb-1">Total
Drivers</div>
                        <div class="h5 mb-0 font-weight-bold text-gray-800"><?= $total_
drivers_count ?? 0 ?></div>
                    </div>
    <div class="col-auto">
          <i class="fas fa-users fa-2x text-gray-300"></i>
        </div>
      </div>
    </div>
  </div>
</div>

<!-- Total Bookings -->
<div class="col-xl-3 col-md-6 mb-4">
  <div class="card border-left-info shadow h-100 py-2 card-hover">
    <div class="card-body">
      <div class="row no-gutters align-items-center">
        <div class="col mr-2">
          <div class="text-xs font-weight-bold text-info text-uppercase mb-1">Total
    Bookings</div>
          <div class="h5 mb-0 font-weight-bold text-gray-800
    $total_bookings_count ?? 0        ?></div>
        </div>
        <div class="col-auto">
          <i class="fas fa-clipboard-list fa-2x text-gray-300"></i>
        </div>
      </div>
    <!-- Pending Bookings -->
    <div class="col-xl-3 col-md-6 mb-4">
  <div class="card border-left-warning shadow h-100 py-2 card-hover">
    <div class="card-body">
      <div class="row no-gutters align-items-center">
        <div class="col mr-2">
          <div class="text-xs font-weight-bold text-warning text-uppercase mb-1">Pending
    Bookings</div>
          <div class="h5 mb-0 font-weight-bold text-gray-800"><?= $pending_bookings_count
    ?? 0 ?></div>
        </div>
        <div class="col-auto">
          <i class="fas fa-clock fa-2x text-gray-300"></i>
        </div><!-- Enhanced Quick Actions Section -->
    <div class="row mb-4">
<div class="col-12">
  <div class="card shadow">
    <div class="card-header py-3" style="background: linear-gradient(135deg, #176B87 0%,
    #14546c 100%); color: white;">
      <h6 class="m-0 font-weight-bold"><i class="fas fa-bolt"></i> Quick Actions</h6>
    </div>
    <div class="card-body">
      <div class="row">
    <div class="col-md-3 mb-3">
          <a href="viewservices1.php" class="btn btn-primary btn-block btn-lg rounded-pill
    shadow-sm">

                    <i class="fas fa-tools mr-2"></i>View Service div>

                <div class="col-md-3 mb-3">
    <i class="fas fa-users mr-2"></i>View Drivers
                    </a>
                </div>
                <div class="col-md-3 mb-3">
                    <a href="admin_view_requests.php" class="btn btn-info btn-block btn-lg
rounded-pill shadow-sm">
                        <i class="fas fa-clipboard-list mr-2"></i>View Bookings
                    </a>
                </div>
                <div class="col-md-3 mb-3">
                    <a href="reports.php" class="btn btn-warning btn-block btn-lg rounded-pill
shadow-sm"> <i class="fas fa-chart-bar mr-2"></i>Reports
                    </a>
                </div>
            </div>
        </div>
<!-- Enhanced Recent Activity Section -->
<div class="row">
    <div class="col-lg-8">
        <div class="card shadow mb-4">
            <div class="card-header py-3" style="background: linear-gradient(135deg, #176B87
0%, #14546c 100%); color: white;">
                <h6 class="m-0 font-weight-bold"><i class="fas fa-history"></i> Recent
Activity</h6>
            </div>
            <div class="card-body">
                <div class="row">
                    <div class="col-md-12">
                        <p>Welcome to the Vehicle Service Management System.
Here you can manage services, drivers, and bookings.</p>
                        <ul class="list-group list-group-flush">
                            <li class="list-group-item"><i class="fas fa-check-circle text-success mr-
2"></i> Add or manage services offered</li>
                            <li class="list-group-item"><i class="fas fa-check-circle text-success mr-
2"></i> Assign and manage drivers</li>
                            <li class="list-group-item"><i class="fas fa-check-circle text-success mr-
2"></i> View and update booking requests</li>
                            <li class="list-group-item"><i class="fas fa-check-circle text-success mr-
2"></i> Generate reports and analytics</li>
                        </ul>
                    </div>
```

```
<div class="actio
<style>
    .border-left-primary {
        border-left: 4px solid #176B87 !important;
    }
    .border-left-success {
        border-left: 4px solid #28a745 !important;
    }
    .border-left-info {
        border-left: 4px solid #17a2b8 !important;
    }
    .border-left-warning {
        border-left: 4px solid #ffc107 !important;
    }
    .card {
        border: none;
        border-radius: 10px;
        box-shadow: 0 0 20px rgba(0,0,0,.08);
        margin-bottom: 20px;
        transition: transform 0.3s ease, box-shadow 0.3s ease;
    }
    .card-hover:hover {
        transform: translateY(-5px);
        box-shadow: 0 10px 25px rgba(0,0,0,.15) !important;
    }
    .card-header {
        border-radius: 10px 10px 0 0 !important;
        padding: 15px 20px;
    }
    .rounded-pill {
        border-radius: 50rem !important;
    }
    .btn-lg {
        padding: 0.75rem 1.5rem;
        font-size: 1
color: #5a5c69 !imtant;
}
.text-gray-300 {
    color: #dddfeb !important;
  }
  .font-weight-bold {
    font-weight: 700 !important;
  }
  .text-xs {
    font-size: .75rem;
  }
  .status-item {
    padding: 10px;
    border-radius: 5px;
    background-color: #f8f9fc;
    transition: background-color 0.2s;
  }
  .status-item:hover {
    background-color: #eef2f7;
  }
</style>

<?php include("includes/admin_footer.php"); ?>
```