શ્રી સ્વામિનારાયણ ગુરુકુલ રાજકોટ સંસ્થાન
શાસ્ત્રી સ્વામી શ્રી ધર્મજીવનદાસજી
સાયન્સ & iT ગુરુકુલ કોલેજ
ગુરુકુલ કેમ્પસ, કોલેજ રોડ, જૂનાગઢ

Jay Chamunda Auto Garage & Service Station

Project Partners:

Mr. Parth N. Chhaya

:: submitted to ::
BKNM University, Junagadh

:: GUIDED BY ::

Mr. Ripal V. Pandya

Mr. Milind Anandpara

શ્રી સ્વામિનારાયણ ગુરુકુલ રાજકોટ સંસ્થાન

શાસ્ત્રી સ્વામી શ્રી ધર્મજીવનદાસજી

સાયન્સ & IT ગુરુકુલ કોલેજ

ગુરુકુલ કેમ્પસ, કોલેજ રોડ, જૂનાગઢ

(Affiliated to **Bhakta Kavi Narsinh Mehta University, Junagadh**)

Project Completion Certificate

This certificate is awarded to

Mr. Parth N. Chhaya	**BCA5-2024**

in completion of project work

PHP	**07/06/2024**
MySQL	**11/10/2024**

Mr. Ripal V. Pandaya

Mr. Milind Anandpara

Project Guide

Director

www.sssdiit.junagadhgurukul.org

TO WHOMSOEVER IT MAY CONCERN

This is to certify that MR. Parth N. Chhaya BCA 5th Semester Student of SSSDIIT College has undergone Project from JULY 2024 to OCTOBER 2024 in our Jay Chamunda Auto Garage & Service Station.

During Live Project period, He successfully completed a Jay Chamunda Auto Garage & Service Station Website using *PHP & SQL SERVER* Tools & Technology

His performance during the live project period was appreciable.

Stamp

Mr. Nareshbhai G. Chhaya

Project Guide

Mr. Ripal V. Pandya

Mr. Milind Anandpara

A

PROJECT REPORT ON

Jay Chamunda Auto Garage & Service Satation

Submitted in Fulfillment of Requirements

For Completion of Semester - 5 in

Bachelor of computer application

Year 2024

To

SHASHTRI SWAMI SHREE DHARMAJIVANDASJI

INSTITUTE OF INFORMATION TECHNOLOGY

JUNAGADH

Guided By:

Mr. Ripal V. Pandaya

Mr. Milind Anandpara

Prepared By:

Mr. Parth N. Chhaya

PREFACE

The Chamunda Auto Garage and Service Station project has been developed as part of the curriculum for the Bachelor of Computer Application program, undertaken during the 5th semester at Shastri Swami Shree Dharmjivandasji Institute of Information & Technology College - Junagadh, affiliated with Bhakt Kavi Narshih Maheta University. This project aims to provide a detailed understanding of the operations and services offered by Chamunda Auto Garage, a motorcycle repair and service station.

With the growing number of motorcycles on the road and the increasing demand for quality repair services, the role of local garages has become essential in ensuring the safety and performance of vehicles. Chamunda Auto Garage offers a range of services, including routine maintenance, engine diagnostics, brake repairs, and more, catering to a diverse set of customers.

This project focuses on the efficient management of motorcycle repair services, highlighting the various tools and techniques used at the garage. From customer interaction to service execution, the project provides a comprehensive overview of how Chamunda Auto Garage operates. The garage is equipped with modern tools and employs skilled mechanics who ensure that every motorcycle is serviced with the utmost care and precision.

I would like to express my heartfelt gratitude to my mentors, peers, and the faculty at Shastri Swami Shree Dharmjivandasji Institute of Information & Technology College - Junagadh for their guidance and support throughout this project. Special thanks to Chamunda Auto Garage for allowing me to observe their day-to-day operations and gain valuable practical experience. This project would not have been possible without their cooperation and assistance.

ACKNOWLEDGEMENT

We extend our sincere gratitude to all those who contributed to the successful completion of this project. It is with great pride and satisfaction that we present this website project. First and foremost, we would like to express our deep appreciation to "**BKNM University**" for providing us with the opportunity to undertake and complete this project.

Before diving into the specifics, we wish to take a moment to acknowledge those whose invaluable support and guidance made this project possible. We are especially grateful to '**Mr. Ripal V. Pandya**' for their constant encouragement and insightful direction throughout the project's development. His mentorship has been instrumental in our success, and we remain deeply indebted to them.

We would also like to express our heartfelt thanks to '**Sadhu Rushikeshdashji Swami**', Director, and '**Mr. Rajesh Bharad'**, Assistant Director, of the "**Shastri Swami Shree Dharmajivandasji Institute of Information Technology, Junagadh**", for their unwavering support.

Furthermore, we extend our thanks to our classmates and others who contributed, directly or indirectly, by helping us resolve challenges and enhance the efficiency and appeal of our web development project.

Thank you.

Date: Mr. Parth N. Chhaya

Place: JUNAGADH

INDEX

PROJECT PROFILE

Project Title	JAY CHAMUNDA AUTO GARAGE & SERVICE STATION
Project Description	We extend our sincere gratitude to all those who have contributed to the successful completion of our project, **Jay *Chamunda Auto Garage & Service Station***. It is with immense pride and satisfaction that we present this project, which is a testament to the collective effort of our team. We are particularly grateful to ***BKNM University*** for providing us with the opportunity to work on this project and to apply our knowledge in a real-world context.
Front End	PHP
Back End	MYSQL
Other Tools	HTML, CSS, BOOTSTRAP, JAVASCRIPT, JQuery
Guide	Mr. Ripal V. Pandya
Submitted To	S.S.S.D.I.I.T College

USE OF SYSTEM DEVELOPMENT LIFE CYCLE MODEL

The Software Development Life Cycle (SDLC) is a structured process used to guide the development of software applications. It consists of distinct phases that ensure the systematic creation, testing, and maintenance of a software system.

The SDLC begins with *requirement gathering*, where information is collected using methods such as questionnaires, personal interviews, and observations. This phase focuses on identifying and documenting the needs of the system. In the *analysis phase*, a Software Requirement Specification (SRS) is created, and the gathered data is carefully analyzed to establish a clear understanding of the project scope.

The *design phase* involves creating the blueprint for the software, including database design and graphical user interface (GUI) layouts. Once the design is finalized, the project enters the *coding phase*, where developers write the code for the various modules and components of the software.

The next step is the *testing phase*, where the software undergoes various types of testing, such as unit testing, integration testing, and system testing, to identify and fix any issues. Finally, the software is deployed, and ongoing *maintenance* ensures that the system continues to function effectively over time.

Several SDLC models exist, each with its own approach to these phases. Common examples include:

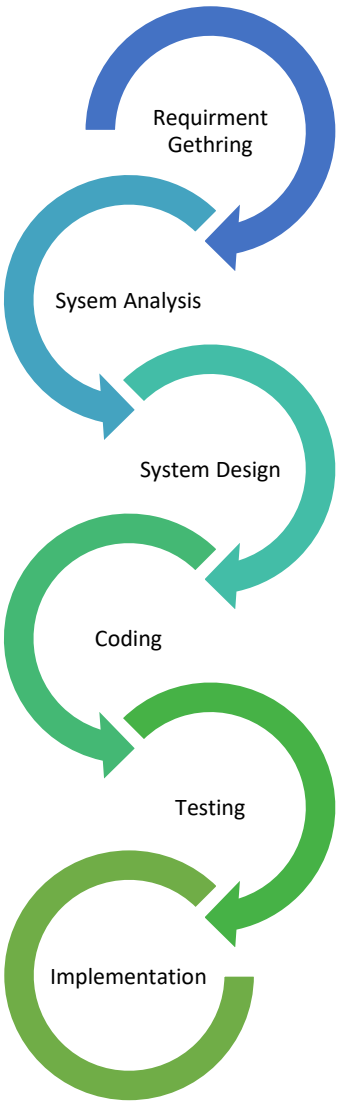
- Waterfall Model
- Iterative Waterfall Model
- Prototyping Model
- Evolutionary Model
- Spiral Model
- Rapid Application Development (RAD) Model

For this project, we have adopted the *Waterfall Model*, a linear and sequential approach to software development. Below is a breakdown of the Waterfall Model's phases:

Waterfall Model

- **Requirement Gathering and Analysis**: This phase involves capturing all the requirements for the system. The gathered information is documented in a comprehensive requirements specification.

- **System Design**: Based on the requirements, a system design is prepared to define the overall architecture of the software. This phase serves as a blueprint for the next stages of development.
- **Implementation**: In this phase, the system is developed by creating small functional units or modules. Each unit undergoes unit testing to ensure its functionality.
- **Integration and Testing**: After individual units are developed and tested, they are integrated into a complete system. Once integrated, the entire system is thoroughly tested for faults and failures.
- **Deployment**: Once all testing is complete, the software is deployed to the customer's environment or released to the market.
- **Maintenance**: Following deployment, the software is continuously maintained to address any bugs or issues and ensure smooth operation over time.



FEASIBILITY STUDY

The development of the "**Jay Chamunda Auto Garage & Service Station**" management system requires a comprehensive feasibility study to evaluate its technical, operational, economic, legal, and schedule aspects. This analysis ensures that the system can be implemented successfully within the given constraints of time, cost, and resources. Below is a detailed analysis of the project's feasibility:

1. Technical Feasibility:

This aspect evaluates whether the existing technology and infrastructure can support the implementation of the system for *Jay Chamunda Auto Garage & Service Station*.

- **Programming Language and Framework:** The project will be developed using *PHP* for the front end, which is a widely used and robust server-side scripting language. It is known for its flexibility and compatibility with web servers, making it ideal for this type of application.

- **Database Management:** 'MySQL' will be employed for database management. MySQL is highly scalable and efficient in handling large amounts of data, making it ideal for managing customer details, vehicle records, service logs, and invoices.

- **Front-End Tools:** The user interface will be designed using 'HTML', 'CSS', 'JavaScript', 'jQuery', and 'Bootstrap'. These technologies ensure that the system is responsive, interactive, and user-friendly across multiple devices.

- **Server and Hosting Requirements:** A web server capable of running PHP scripts and MySQL databases will be required. Cloud-based hosting services can be employed for scalability, ensuring the system's availability and security.

- **Security:** The system will incorporate authentication mechanisms, data encryption, and secure transmission using HTTPS to ensure that sensitive data such as customer and vehicle details remain protected.

Conclusion: The necessary technical tools are readily available and can be integrated within the current technological infrastructure, making the project technically feasible.

2. Operational Feasibility:

Operational feasibility examines whether the system will function effectively within its intended user environment at the garage and service station.

- **User Requirements:** The system will cater to the needs of three key user groups:

I. **Garage Owners/Managers:** To manage customer records, service schedules, and billing information.
II. **Technicians/Staff:** To update service status, enter repair details, and manage parts inventory.
III. **Customers:** To check service history, book appointments, and receive invoices.

- **Ease of Use:** A user-friendly interface will be designed to ensure smooth operation for both technicians and managers. The system will be optimized for mobile and desktop devices, allowing real-time updates and customer interaction.

- **Training and Support:** Minimal training will be required for users as the system will offer intuitive navigation and clear instructions. Documentation and support resources will also be provided.

Conclusion: The system is operationally feasible and will significantly streamline day-to-day operations at *Jay Chamunda Auto Garage & Service Station*, enhancing overall service efficiency.

3. Economic Feasibility:

The economic feasibility focuses on the cost-benefit analysis of the system implementation.

- **Development Costs:** Development costs will be minimized as the tools (‘PH'P’, ‘MySQL’, ‘HTML’, ‘CSS’, ‘JavaScript’, ‘jQuery’, ‘Bootstrap’) are open-source and freely available. The primary expenses will include server and hosting costs.

- **Implementation Costs:** The system's deployment will involve costs for web hosting and domain registration. These costs will be relatively low, and affordable hosting plans can be selected to manage expenses efficiently.

- **Long-Term Benefits:** The system will reduce the manual work of managing customer information, service records, and invoicing. By automating these processes, the garage will save time, reduce errors, and enhance customer satisfaction, leading to increased revenue in the long run.

Conclusion: The project is economically feasible with minimal development costs and significant long-term benefits that will outweigh the initial investment.

4. Legal Feasibility:

Legal feasibility assesses the legal requirements and considerations associated with the system.

- **Data Privacy and Security**: The system will handle sensitive customer and vehicle information. Adherence to local data protection regulations will be ensured, including the use of secure storage and transmission protocols (e.g., GDPR compliance where applicable).

- **Legal Compliance:** The system will be designed in compliance with all relevant laws governing customer data, billing, and electronic records management.

Conclusion: By following legal regulations, the project is legally feasible and can be implemented without significant legal hurdles.

5. Schedule Feasibility:

This section evaluates whether the project can be completed within the proposed timeline.

Development Phases:

Phase 1: Requirement gathering and initial design (1-2 weeks).
Phase 2: Database design and back-end development using **PHP** and **MySQL** (3-4 weeks).
Phase 3: Front-end development and user interface integration (2-3 weeks).
Phase 4: Testing and debugging (1-2 weeks).
Phase 5: Final deployment and documentation (1 week).

Conclusion: The project timeline is achievable within the allocated time frame, making it schedule-feasible.

Overall Conclusion:

The feasibility study confirms that the development of the "Jay Chamunda Auto Garage & Service Station" management system is technically, operationally, economically, legally, and schedule feasible. The project can be successfully executed within the given constraints and will provide significant long-term benefits to the business.

REQUIRMENT GATHERING

The *Jay Chamunda Auto Garage & Service Station* project aims to enhance the efficiency of motorcycle repair and maintenance services through a comprehensive online management system. The requirement gathering phase is critical in understanding the needs of various stakeholders, including customers, mechanics, and administrative staff, ensuring that the system aligns with its intended objectives. This phase captures both functional and non-functional requirements, guaranteeing that all essential features—such as appointment scheduling, service tracking, and customer notifications—are effectively addressed. By thoroughly engaging with stakeholders during this phase, we aim to create a user-centric system that improves service delivery and customer satisfaction.

1. Functional Requirements:

Functional requirements describe the specific behaviors and functionalitiesthe system must have.

1.1 User Roles and Access Control

- **Administrator**
  - Manage Photos of Gallery
  - Manage Customer Feedback

- **Client**
  - Know About the Garage
  - Know Special Offers
  - Book Appointment
  - Contact with Staff
  - Know About Services

1.2 Gallery Management

 Photo Storage:

- Store photos in the database to maintain a history of uploaded images for reference.
- Implement criteria for photo size and file type when adding images through the admin panel to ensure consistency and optimize storage.

 Photo Display:

- Ensure photos are displayed in a user-friendly and responsive manner within the gallery section on the client-facing website.
- Display photos on the admin panel with intuitive, customized controls for easy management (e.g., edit, delete, and view options).

Photo Management:

- Enable the admin to manage the gallery effectively, with the ability to view, edit, or delete photos as needed.
- Ensure the photo management system is secure, accessible, and straightforward, making it easy for the admin to upload and organize photos efficiently.

1.3 Feedback Management

Feedback Storage:

- **Purpose**: To store customer feedback in a structured database for historical reference and performance analysis.
- **Implementation**: Ensure criteria for feedback content, such as text length, rating scale, and attachment formats, are clearly defined. Implement validation mechanisms to ensure feedback meets the required standards before submission in the admin panel.

Feedback Display:

- **Client-Side Display**: Present customer feedback on the client-side interface in an organized, user-friendly manner. Ensure the layout is responsive and accessible across various devices for seamless viewing.
- **Admin Panel Display**: Display feedback within the admin panel with clear, easy-to-navigate controls. Customized buttons for actions such as responding, archiving, or marking feedback as resolved will facilitate effective management.

Feedback Management:

- **Purpose**: Admins must have full control over feedback, with the ability to view, archive, or delete feedback entries as necessary.
- **Ease of Use**: The feedback management system should be intuitive, ensuring admins can quickly access, sort, and manage feedback data. Security protocols must also be in place to safeguard sensitive information, ensuring only authorized personnel can manipulate feedback records.

1.4 Authentication and Security

- **Login System:**
  - Secure login system for administrators.
  - Password encryption and secure data transmission (HTTPS).
- **Two-Factor Authentication (Optional):**
  - Implement two-factor authentication to enhance security.

2. Stockholder's Requirements:

Understanding stakeholder expectations is crucial to meeting projectobjectives.

2.1 Students

- **Easy to Access**: Ability to register, log in, and take exams withouttechnical difficulties.

- **Easy to Manage**: Ability to manage things for handle many things like gallery, feedback, offers, etc.

REQUIREMENT ANALYSIS

1. Hardware and Software Requirements

1.1 Hardware Requirements

- **Development Machine**:
  - **Processor**: Intel Core i5 or higher / AMD equivalent for optimal performance.
  - **RAM**: Minimum 8 GB (16 GB recommended for smooth multitasking during development).
  - **Storage**: At least 256 GB SSD for faster performance and quicker load times.
  - **Network**: High-speed internet connection for efficient code deployment, testing, and access to remote resources.

1.2 Software Requirements

- **Operating System**:
  - Development on Windows, macOS, or Linux, depending on developer preference.
- **Web Server**:
  - Apache HTTP Server (recommended) or Nginx for hosting the project.
- **Database**:
  - MySQL, with the option to use MariaDB as an alternative for database management.
- **Programming Language**:
  - PHP (Version 7.4 or higher) for server-side scripting and back-end logic.
- **Text Editor/IDE**:
  - Development tools such as Visual Studio Code, Sublime Text, or PHPStorm for writing and managing code.
- **Browser Compatibility**:
  - Ensure compatibility across major web browsers like Chrome, Firefox, Safari, and Microsoft Edge.

2. Front-End Tools

Although PHP is primarily a back-end scripting language, it interacts closely with front-end technologies by dynamically generating HTML, CSS, and JavaScript content on the server.

- **PHP**:
  - **Purpose**: PHP is responsible for server-side logic that dynamically generates web pages and serves content to the client.

- **Use in Front-End**: PHP plays a crucial role in embedding dynamic data such as customer information, service history, and invoices into the front-end (HTML). It fetches this data from the MySQL database, creating HTML pages tailored to individual users and their service needs.

---

3. Back-End Tools

MySQL will be the primary database management system, facilitating the storage and management of data for *Jay Chamunda Auto Garage & Service Station*.

- **MySQL**:
  - **Purpose**: A relational database management system used to store and retrieve structured data.
  - **Use**:
    - **Customer Data**: Stores all relevant information about customers, their vehicles, and service history.
    - **Service Management**: Manages data on service appointments, repair details, and spare parts inventory.
    - **Invoice Data**: Tracks and processes customer billing and invoices.
    - **Relationships**: Maintains relationships between various data tables, linking customers to their vehicles, services, and invoices.
  - **Features**: Supports CRUD operations (Create, Read, Update, Delete), relational data management, indexing for performance optimization, and robust data security.

---

4. Other Tools & Technologies Used

4.1 HTML (Hypertext Markup Language)

- **Purpose**: HTML forms the foundation of the website's structure.
- **Use**: It will be used to create the layout of all pages within the system, such as the booking form, service records page, and customer dashboards. PHP will dynamically inject content into these HTML templates based on real-time data from the MySQL database.

4.2 CSS (Cascading Style Sheets)

- **Purpose**: CSS controls the appearance and layout of the web pages.
- **Use**: CSS will be used to style the HTML content, providing a visually appealing interface. This includes setting colors, fonts, and responsive layouts to ensure the website is user-friendly and looks professional across all devices.

4.3 Bootstrap

- **Purpose**: A front-end framework used for creating responsive and mobile-friendly designs.
- **Use**: Bootstrap will enable quick and consistent front-end development by providing pre-built responsive components like buttons, forms, and grids. This will ensure that the system is optimized for both desktop and mobile users, offering a seamless user experience.

4.4 JavaScript

- **Purpose**: JavaScript adds interactivity and enhances the user experience by enabling dynamic client-side functionality.
- **Use**: JavaScript will be utilized for real-time validation (e.g., form submissions, service bookings), handling user interactions (e.g., selecting service types, confirming bookings), and enhancing overall usability. Features such as auto-refreshing service status, dynamically loading content without page reloads, and countdown timers for appointments will be implemented using JavaScript.

4.5 jQuery

- **Purpose**: jQuery is a fast, small, and feature-rich JavaScript library designed to simplify HTML document traversal and manipulation, event handling, and animation. It enhances the capabilities of standard JavaScript by providing a straightforward API that works across a multitude of browsers.
- **Use**: jQuery will be employed to streamline the development of interactive elements. Key applications include:
    - **DOM Manipulation**: jQuery allows for efficient manipulation of HTML elements, enabling developers to modify the content, structure, and style of web pages dynamically.
    - **Event Handling**: It simplifies the process of capturing and responding to user interactions, such as clicks, form submissions, and other events. This is crucial for features like service selection and appointment scheduling.
    - **AJAX Requests**: jQuery's AJAX capabilities facilitate asynchronous communication with the server, allowing for real-time updates without refreshing the page. This is particularly useful for loading service information, customer notifications, and status updates seamlessly.
    - **Animations and Effects**: jQuery provides a variety of built-in animation effects that can enhance user experience by making transitions smoother and more engaging.

PROJECT ABSTRACTS

The *Jay Chamunda Auto Garage & Service Station* website is designed to showcase the services offered by the garage, helping customers find a reliable place for motorcycle repairs and maintenance. The website provides comprehensive information, including garage details, service offerings, pricing, special promotions, contact details, and the ability to book appointments online. It is targeted at individuals searching for a professional and trustworthy garage for their motorcycles.

The project is built to support two key user roles:

- **Admin**: The admin has full control over managing the website's content. This includes features such as gallery/photo management, feedback management, offer updates, and overall site maintenance. The admin can also add or manage other admins through a secure login system.
- **Client**: The client-facing side allows users to view the list of services and their prices, garage contact information, emergency assistance, feedback, and book appointments. Additionally, the website offers a navigation feature to help users locate the garage easily.

Through automation, the website significantly reduces administrative overhead, providing real-time updates and seamless data management through APIs. The codebase is optimized for production, ensuring the site is fast, responsive, and easy to access. This system improves overall service efficiency while enhancing the customer experience.

Key Features

1.User Management

- Admins can create admin.
- Admin can manage content of website.

2. **Gallery:**
   - It contains images of the garage.
   - It can also define the services of the services.
3. **Feedback**
   - It contains feedbacks given by the client or visitors.
   - Feedbacks can handle by admin like delete the feedback.
4. **About**

- It contains the information about the garage like history and experience.
- It also contains the vison and mission of the owner of the garage.

5. **Services**
   - It contains the service provided by the garage.
   - It contains the special packages of the services which is listed by the owner and contains the information of expert services by the staff.

PROPOSED SYSTEM

1. Home Page

- **Objective**: To serve as the primary landing page that introduces users to the garage's services and offers a welcoming, engaging experience.
- **Key Features**:
  - Hero section with image sliders showcasing services and projects.
  - Call-to-action buttons such as “Book an Appointment” and “Contact Us.”
  - Highlights of services, linking to the detailed services page.
  - Testimonials and reviews section.

2. Services Page

- **Objective**: To provide detailed information on all services offered by the garage.
- **Key Features**:
  - Comprehensive list of services (repairs, maintenance, customization, etc.).
  - Pricing information for standard services.
  - Service booking form.
  - WhatsApp API integration for direct bookings.

3. About Us Page

- **Objective**: To communicate the garage’s mission, values, and team expertise.
- **Key Features**:
  - Information on the garage's history and background.
  - Details about the team members and their qualifications.
  - Photos and visual elements showcasing the garage and team.

4. Gallery Page

- **Objective**: To visually showcase past projects and the garage’s capabilities.
- **Key Features**:
  - Dynamic image gallery with categorized sections.
  - Admin functionality to manage and organize images.
  - Interactive display with sliders or grid layouts.

5. Contact Page

- **Objective**: To facilitate easy communication between customers and the garage.
- **Key Features**:
  - Contact form for inquiries (name, email, subject, message).
  - Feedback component for customer reviews.
  - Google Maps API integration for location display.
  - WhatsApp API for instant messaging.

6. Admin Login

- **Objective**: To provide a secure portal for administrators to manage website content and operations.
- **Key Features**:
  - Secure login with password protection.
  - Admin dashboard with metrics and operations overview.
  - Role-based access and permissions.

7. Admin Add Functionality

- **Objective**: To enable the admin to add and update content such as services, feedback, and gallery items.
- **Key Features**:
  - CMS for adding or editing services, text, and multimedia.
  - Ability to manage and add new services.

8. Gallery Management

- **Objective**: To provide an easy-to-use tool for the admin to manage images in the gallery.
- **Key Features**:
  - Upload, delete, and organize images.
  - Image preview functionality.
  - Categorization and tagging options.

9. Feedback Management

- **Objective**: To allow the admin to review, respond to, and manage customer feedback.
- **Key Features**:
  - Feedback list with sorting and search functionality.
  - Mark feedback as read, delete, or respond.
  - Analytics on customer feedback trends.

10. Book Appointment (WhatsApp API Integration)

- **Objective**: To streamline the appointment booking process through direct communication on WhatsApp.
- **Key Features**:
  - Form for users to schedule appointments.
  - WhatsApp API integration for automated conversation initiation.
  - Appointment confirmation through WhatsApp.

11. Map API Integration

- **Objective**: To provide users with a clear, interactive map for finding the garage's location.
- **Key Features**:
  - Google Maps integration with location markers.
  - Option for users to get directions.
  - Display of nearby services using Google Maps API.

Conclusion:

This proposed system offers a robust platform to improve customer interaction and enhance the operational efficiency of the motorcycle garage. With user-friendly design, admin control features, and API integrations for real-time bookings and location services, the website will serve as a complete solution for both customers and administrators.

ADVANTAGES & LIMITATIONS OF PROPOSED SYSTEM

Advantages:

1. **Enhanced Customer Experience**:
   - **Easy Navigation**: The website's clear structure and responsive design ensure a seamless user experience across all devices, helping customers easily find services, view the gallery, and contact the garage.
   - **Convenient Appointment Booking**: The integration of WhatsApp API allows users to book appointments instantly through a familiar messaging platform, making the booking process fast and efficient.
   - **Engaging Visual Content**: The gallery and service descriptions will visually showcase the garage's work, attracting potential customers and building trust.
   - **Real-Time Interaction**: With WhatsApp and Google Maps API, users can interact in real time, inquire, and locate the garage with ease.
   - **Feedback and Review Feature**: Customers can share their experiences via the feedback component, fostering transparency and improving service quality based on feedback trends.
2. **Streamlined Administrative Control**:
   - **Admin Dashboard**: The admin portal offers easy management of services, gallery images, and customer feedback, helping administrators maintain and update the website without technical expertise.
   - **Service & Content Flexibility**: Admins can quickly add, edit, or remove services, feedback, and images, ensuring the site remains up to date with minimal effort.
   - **Feedback Management**: The admin can view and respond to customer feedback directly, enhancing customer relations and allowing quick resolution of issues.
   - **Cost-Effective Appointment Handling**: With the WhatsApp integration, booking can be handled without the need for complex scheduling systems, reducing operational costs.
   - **Scalability**: As the garage grows, the website's modular structure allows easy integration of new features, such as additional services or locations.
3. **Brand Credibility and Growth**:
   - **Professional Presentation**: The website enhances the brand's professional image, with well-designed layouts and a modern look.
   - **Improved Customer Engagement**: The interactive elements (feedback, gallery, real-time communication) help engage customers, increasing loyalty and encouraging repeat visits.

- **Search Engine Visibility**: With optimized pages and API integration, the site will likely improve search engine rankings, attracting more organic traffic.

Limitations:

1. **Limited Customization for Complex Operations**:
   - **Basic Booking System**: The WhatsApp-based booking system, while convenient, may not offer the advanced scheduling capabilities of a dedicated appointment management system, such as real-time availability tracking or multi-step booking processes.
   - **Simple Admin Interface**: The content management system (CMS) is designed for ease of use but may lack customization options for admins with more complex needs, such as advanced reporting or marketing tools.
2. **Reliance on Third-Party APIs**:
   - **API Downtime or Changes**: The system relies on WhatsApp API and Google Maps API for critical functions (booking and location services). Any changes or downtime in these APIs could disrupt the user experience, limiting customer interaction and affecting operations.
   - **Data Privacy Concerns**: Integration with third-party platforms such as WhatsApp and Google Maps could raise concerns about data privacy and security, requiring compliance with global privacy standards like GDPR.
3. **Resource Intensive for Management**:
   - **Manual Feedback and Appointment Handling**: Admins need to manually respond to feedback and appointment requests via WhatsApp, which can become time-consuming as the number of customers grows, especially if automated systems are not implemented.
   - **Gallery Management**: While easy to manage, the manual upload and organization of images may become tedious as the gallery expands with more content.
4. **Limited Offline Functionality**:
   - **No Offline Access**: The system heavily relies on an active internet connection for all functionalities, such as viewing the gallery, contacting via WhatsApp, and using the map API. Customers or admins cannot access features offline, which may be an issue for users in areas with unstable internet.
5. **Scalability Limitations for Larger Operations**:
   - **Not Optimized for Multi-Branch Garages**: The proposed system is best suited for a single garage location. If the garage expands to multiple locations, significant adjustments and more complex integration of services will be required to ensure smooth functionality.
   - **Limited Multi-User Admin Management**: The current admin structure is simple and might face limitations if there's a need for multiple admins managing different aspects of the site simultaneously, requiring role-based permissions and enhanced security features.

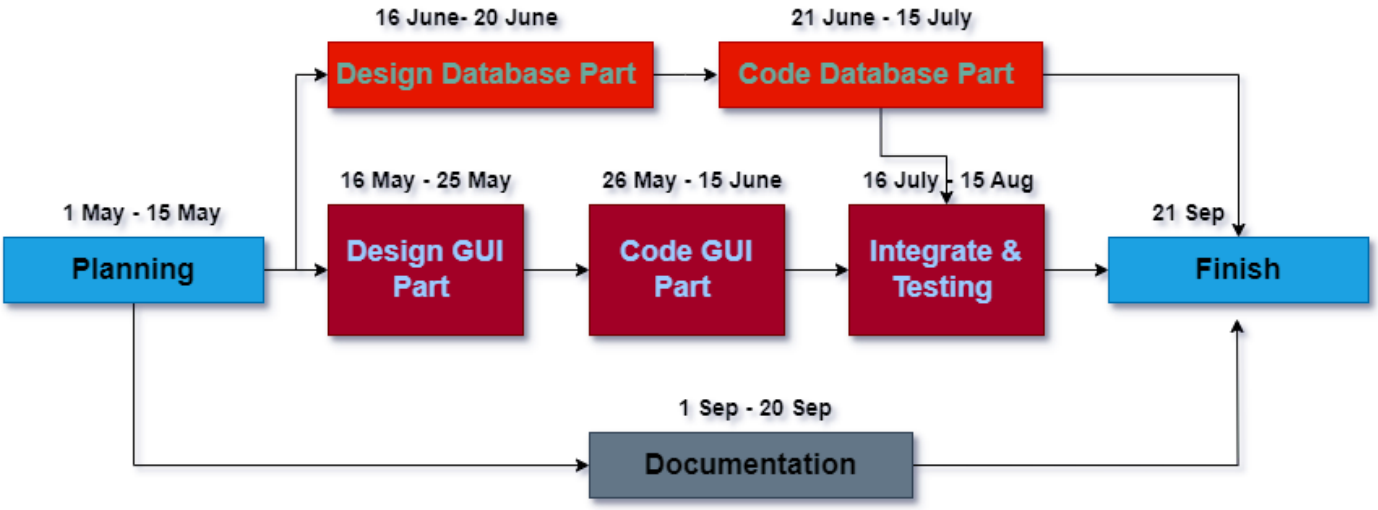
PERT CHART AND GANTT CHART OVERVIWE

PERT CHART:

A **PERT (Project Evaluation and Review Technique) chart** is a visual representation of project tasks and their dependencies, structured as a network of boxes and arrows. In a PERT chart, the boxes symbolize project activities, while the arrows depict the dependencies between those tasks.

What sets a PERT chart apart from other activity charts is its ability to incorporate multiple time estimates for each task: pessimistic, most likely, and optimistic. These estimates are typically annotated within the task boxes. As a result, instead of identifying a single critical path, PERT charts account for multiple potential critical paths, depending on the various combinations of time estimates for each task. This flexibility allows for a more comprehensive analysis of potential project timelines, with shaded boxes often used to highlight critical paths.

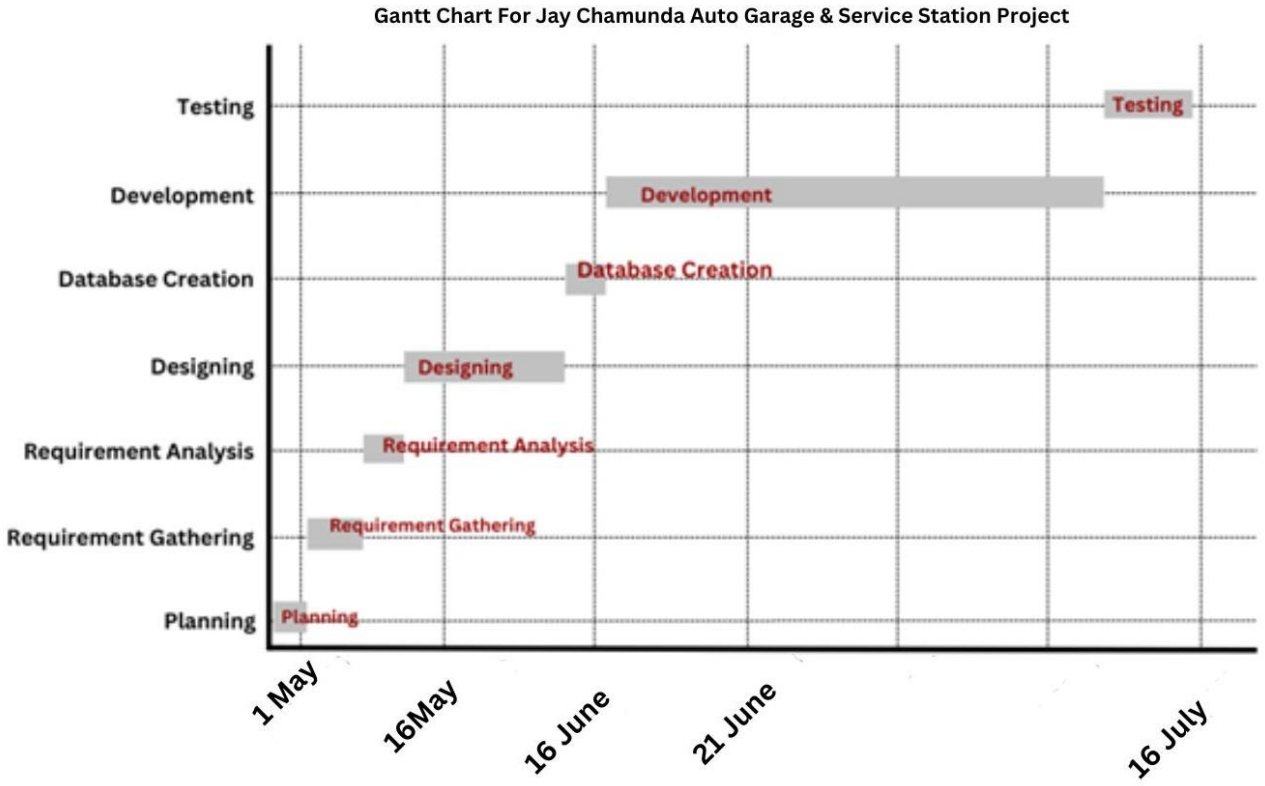
The following PERT chart provides a representation of the MIS problem.



GANTT CHART:

A **Gantt chart** is primarily used for project scheduling, budgeting, and resource planning. It allocates various resources—such as staff, hardware, and software—to specific activities within the project.

A Gantt chart is a specialized bar chart where each bar represents a distinct activity. These bars are plotted along a timeline, with the length of each bar proportional to the planned duration of the corresponding activity. This visual representation allows for easy tracking of project progress and resource allocation across the project lifecycle.



DATA FLOW DIAGRAM

In the **Detailed Life Cycle of the Project**, it is crucial to focus on the **Data Flow Diagram (DFD)** during the development of the software project. The DFD plays a key role in illustrating the system's processes and transactions.

What is a Data Flow Diagram (DFD)?

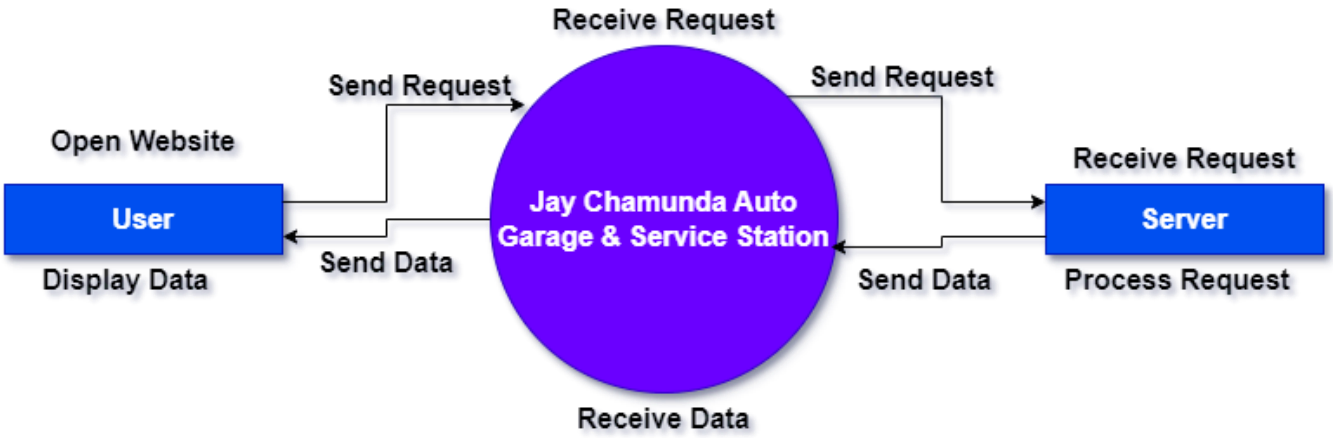
A **Data Flow Diagram** is a graphical representation of the processes and data flows within a system. It allows both developers and end-users to easily understand the system's structure in a concise and efficient manner. By using a DFD, we can visually depict how data moves through the system, highlighting inputs, processes, and outputs.

Why is a DFD Important?

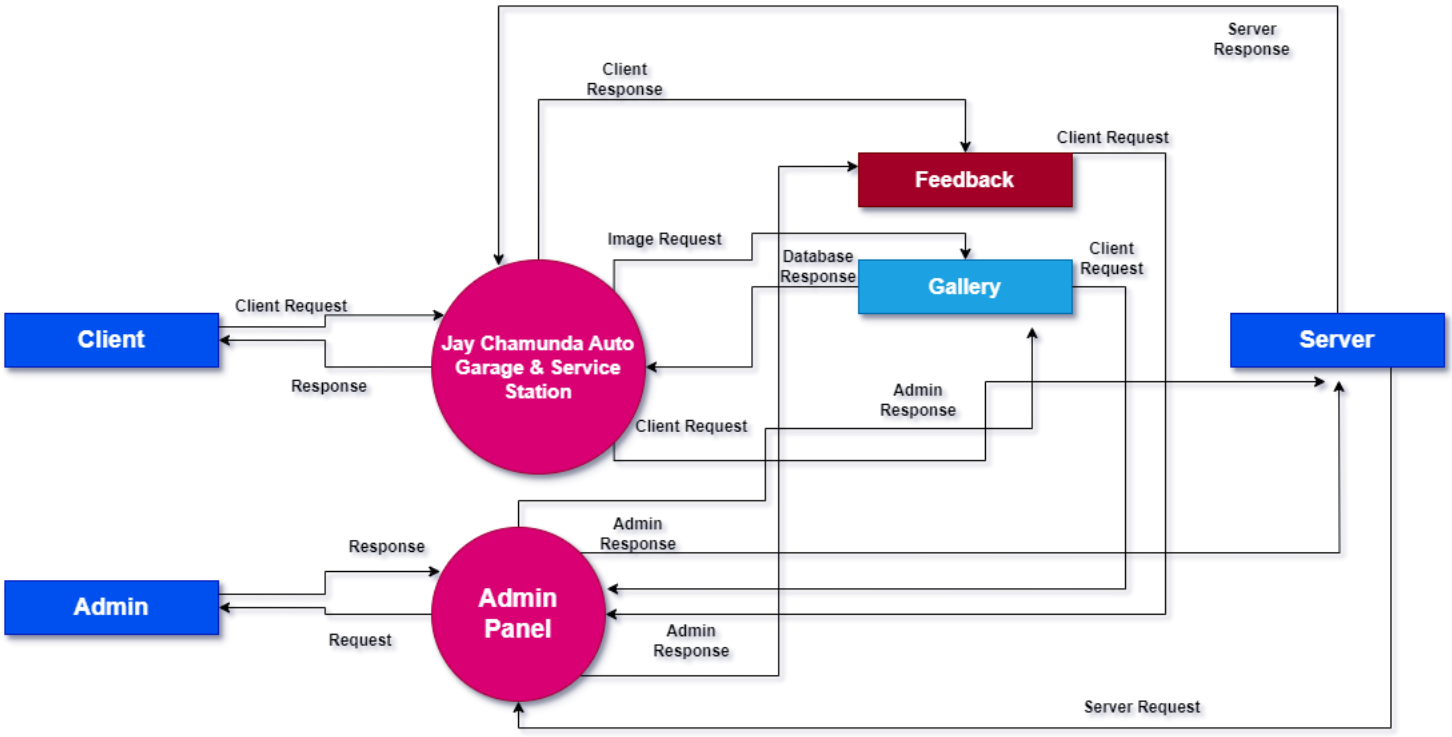
- **Enhanced Communication:** The DFD helps bridge the gap between technical teams and end-users, ensuring that all stakeholders understand the system.
- **Problem Identification:** It facilitates the identification of potential issues or inefficiencies within the system during the early stages of development.
- **Verification of Requirements:** The DFD provides a platform for validating whether the system design aligns with customer requirements, ensuring that the project remains on track.
- **Essential Development Phase:** As a fundamental part of the software development lifecycle, the DFD ensures that the system is clearly understood before moving forward with development, helping to reduce misunderstandings or errors.

In summary, the **Data Flow Diagram** is a critical tool for both understanding and developing the system according to customer needs, making it an indispensable phase in the software development process.

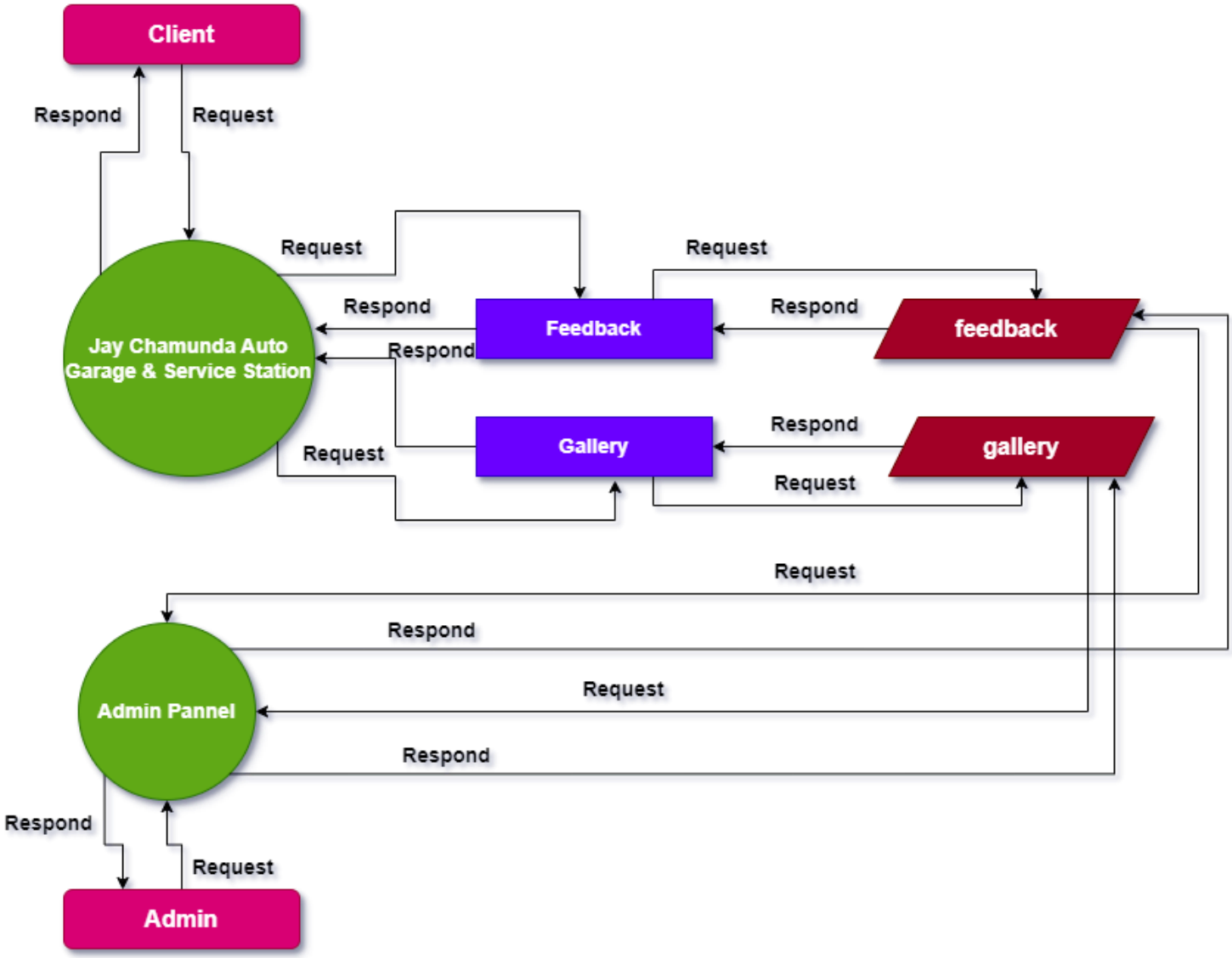
1) 0 Level Diagram:



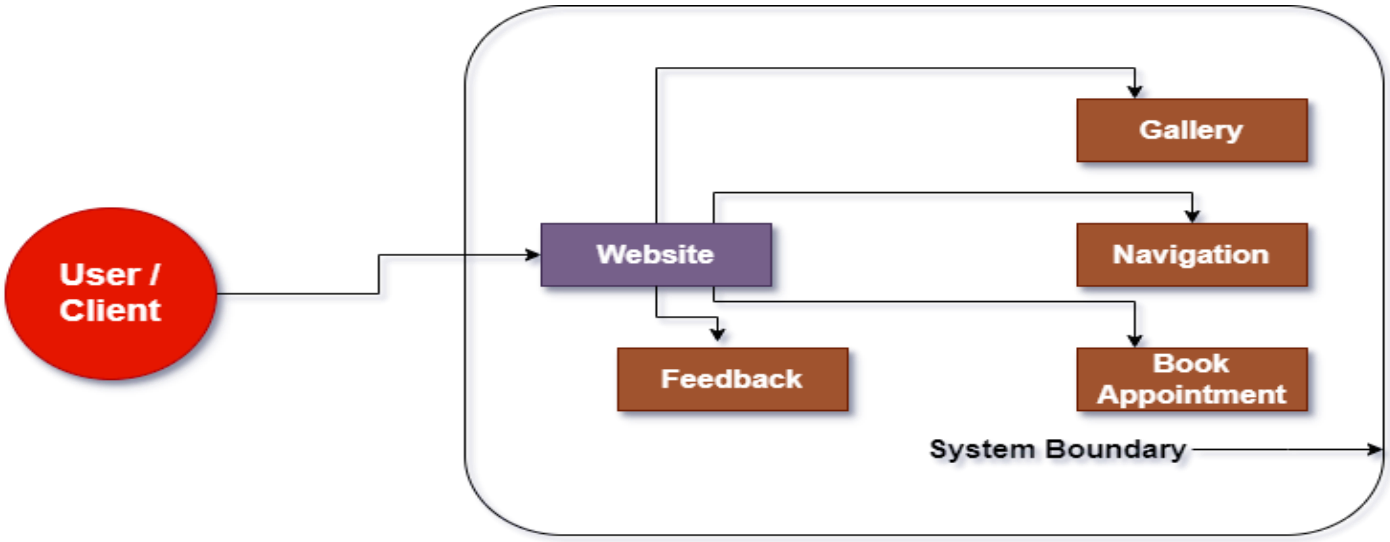
2)1st Level Diagram:



3) 2nd Level Diagram:



USE CASE DIAGRAM



This diagram illustrates a **Use Case Diagram** for Jay Chamunda Auto Garage & Service Station Website, depicting the interaction between the **User/Client** and different modules within the system boundary. Here's a detailed description:

1) User/Client (External Actor):

- Represented by the red circle, the user or client interacts with the website. They perform different actions depending on their needs, such as navigating the site, viewing the gallery, booking appointments, or providing feedback.

2) Website (Main System):

- The **Website**, represented in purple, serves as the primary interface through which the user interacts with the system. All actions taken by the user are routed through this website.

3) Modules within the System Boundary:

- **Gallery:** The user can access a gallery of images or information within the website.
- **Navigation:** This allows the user to explore and navigate different sections or functionalities of the website.
- **Book Appointment:** The user can schedule appointments via this feature, indicating that the system may be offering services that require booking.
- **Feedback:** This module allows users to provide feedback on their experience, which can be crucial for improving the system's performance or offerings.

4) System Boundary:

- This boundary represents the encapsulated functions and interactions of the website. All user interactions happen within this boundary, and the diagram shows the flow of actions starting from the user and moving through various system components.

FLOWCHART

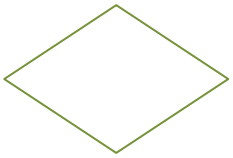
A Flow is a pictorial representation of an algorithm. Programmers often use it as a program-planning tool for visually organizing a sequence of steps necessary to solve a problem using computer. It uses boxes of different shapes to denote different type of instructions. The actual instructions are written within these boxes using clear and concise statements. Solid lines having arrowmarks connect these boxes to indicate the flow of operation, that is, the exact sequence in which to execute the instructions. The process of drawing a flowchart for an algorithm is known as flowcharting.

Basic Flowchart Symbols:

Terminal

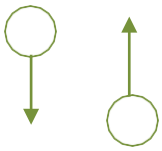
Input/Output

Processing



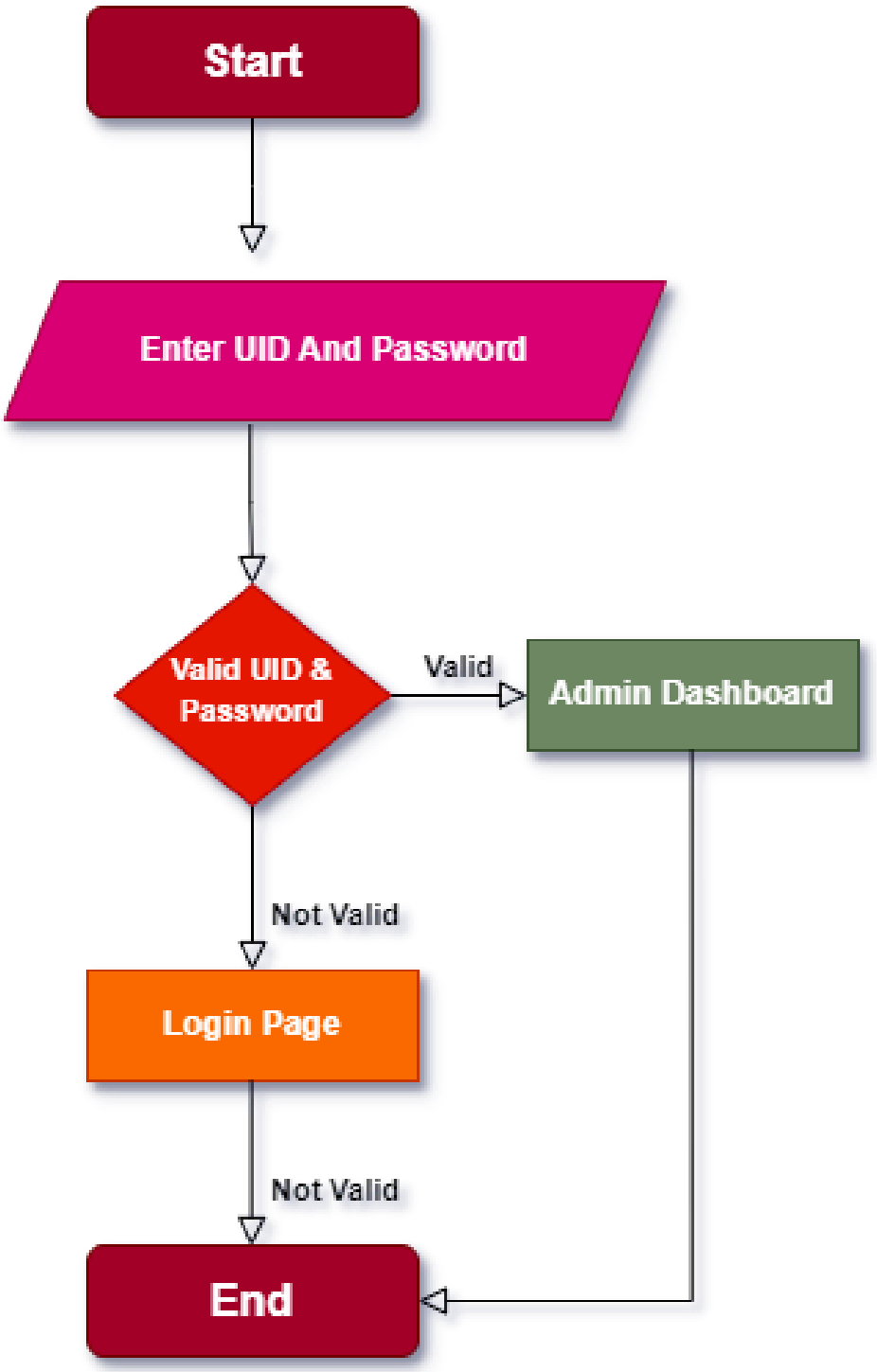
Decision

Flow Lines

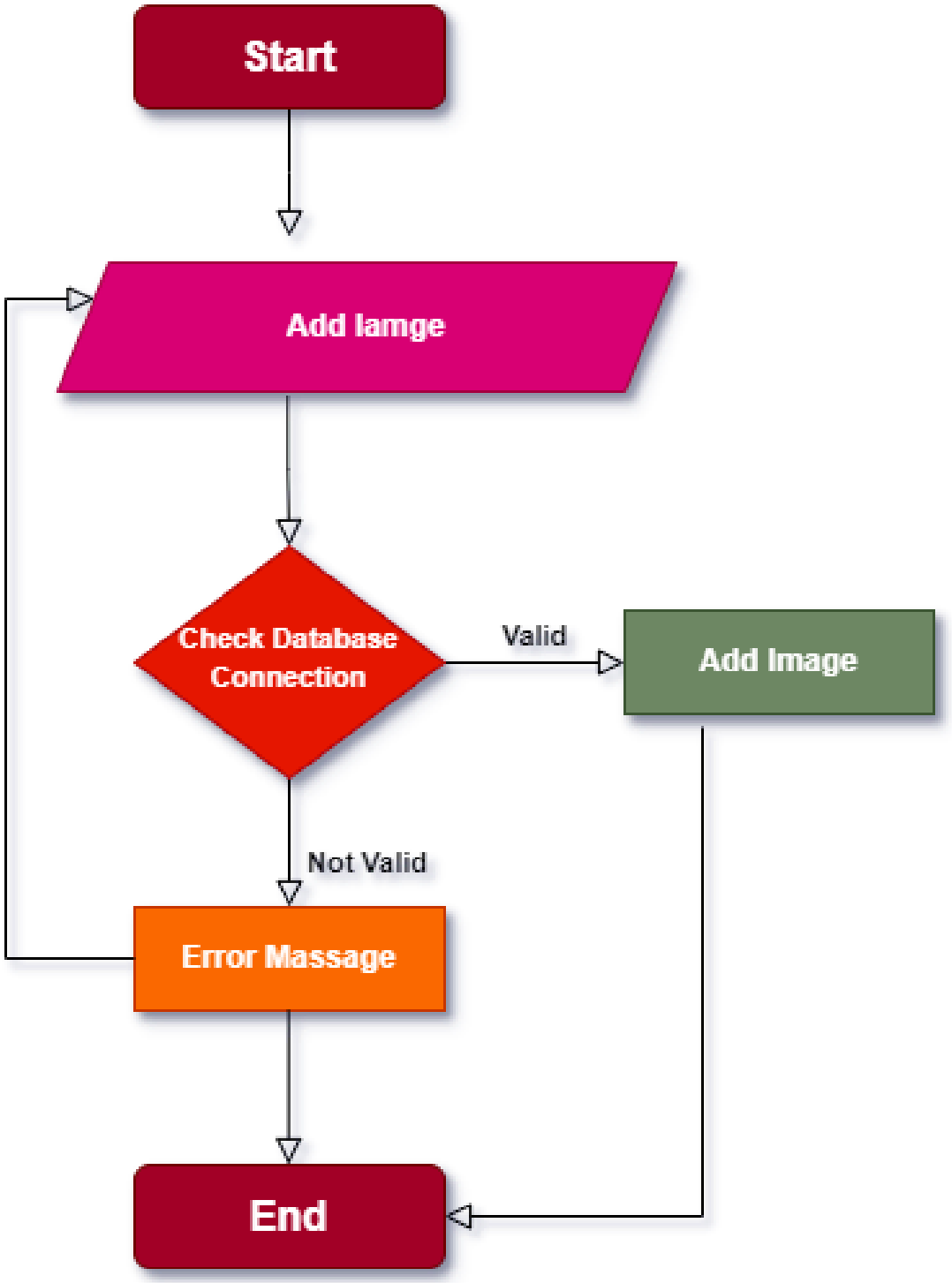


Connectors

1)


Start
Enter UID And Password
Valid UID & Password
Valid
Admin Dashboard
Not Valid
Login Page
Not Valid
End

2)


Start
Add lamge
Check Database Connection
Valid
Add Image
Not Valid
Error Massage
End

COST ESTIMATION

Cost estimation for an "Online Examination System" project can depend on various factors such as development time, hosting services, software tools, and resources.

1. Development Costs

- **Frontend & Backend Development:**
  - If you're hiring developers or contributing your own time, a freelancer may charge anywhere between ₹300 - ₹1,500 per hour depending on experience.
  - Total development time for a simple system could range from 100 to 300 hours, so:
    - Low estimate: ₹30,000 (₹300/hr * 100 hours)
    - High estimate: ₹4,50,000 (₹1,500/hr * 300 hours)
- Your time (if self-developed): This may not have a direct cost but is important to factor in terms of effort and value.

2. Software and Tools

- **IDE (Integrated Development Environment):** Free (VS Code, Sublime Text, etc.) to paid options (₹2,000 - ₹5,000)
- **Version Control System (e.g., GitHub):** Free for basic use, up to ₹500 per month for advanced collaboration tool

3. Testing and QA

- **Manual Testing:** Freelancers or testers might charge ₹500 - ₹1,000 per hour, or use testing tools (some free options available).
- **Total testing cost:** ₹10,000 - ₹50,000 depending on project complexity.

4. Maintenance and Updates

- If you plan to maintain the system post-launch, annual maintenancecould cost ₹10,000 - ₹50,000 depending on updates and bug fixes.

Overall Estimated Cost

- **Basic Version:** ₹40,000 - ₹1,00,000
- **Advance Version (with scaling and cloud services):** ₹1,00,000 - ₹5,00,000

Cost Estimation Table for "Online Examination System" Project

No.	Cost Item	Estimated Cost (₹)	Details
1	Development Costs	₹30,000 - ₹4,50,000	Includes frontend and backend development efforts
2	Software and Tools	₹3,000 - ₹10,500/year	For advanced collaborationtools (optional)
3	Testing and QA	₹10,000 - ₹50,000	Freelance testers or testing tools
4	Maintenance and Updates	₹10,000 - ₹50,000/year	For ongoing updates and bug fixes
5	Overall Estimated Cost	₹40,000 - ₹5,00,000	Based on scope, scaling, andfeatures

DATA DICTIONARY & NORMALIZATION

Database Name: Chamunda

Table1: aaa

Column Name	Datatype(Size)	Constraints	Remarks
Id	Int (100)	primary Key	
user	Varchar(50)		
pass	Varchar(18)		

Table 2: Feedback

Column Name	Datatype(Size)	Constraints	Remarks
id	Int (100)	primary Key	
nam	Varchar(255)		
email	Varchar(255)		
phone	Varchar(15)		
subjec	Varchar(255)		
Messag	Varchar(255)		

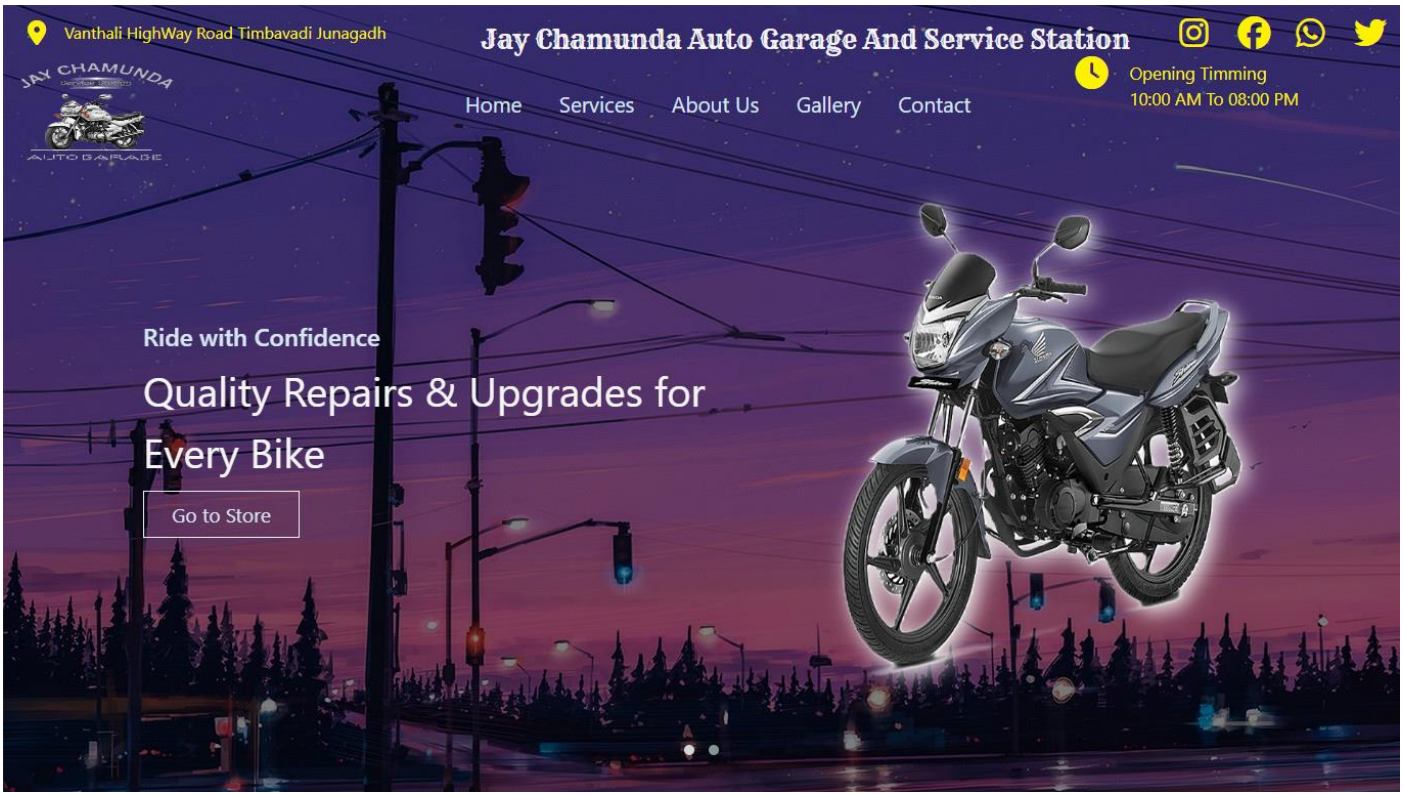
Table 3: Gallery

Column Name	Datatype(Size)	Constraints	Remarks
id	Int (11)	primary Key	
image	Varchar(255)		

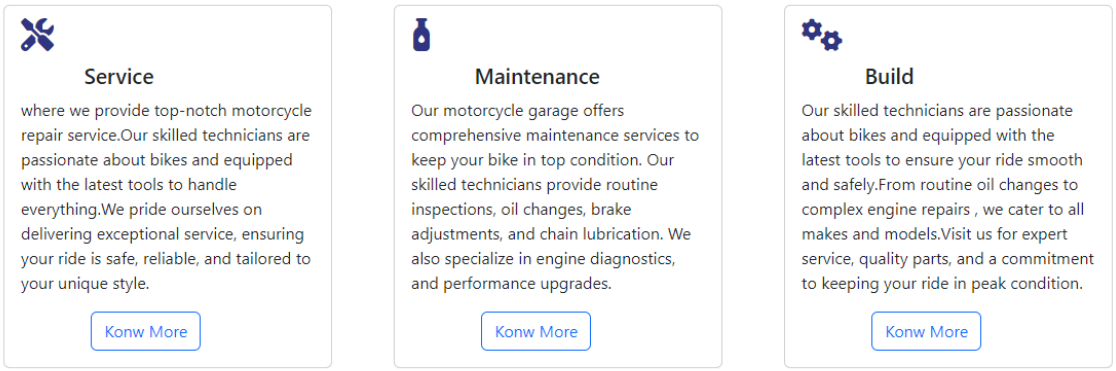
SCREEN LAYOUTS

- **Client Side:**

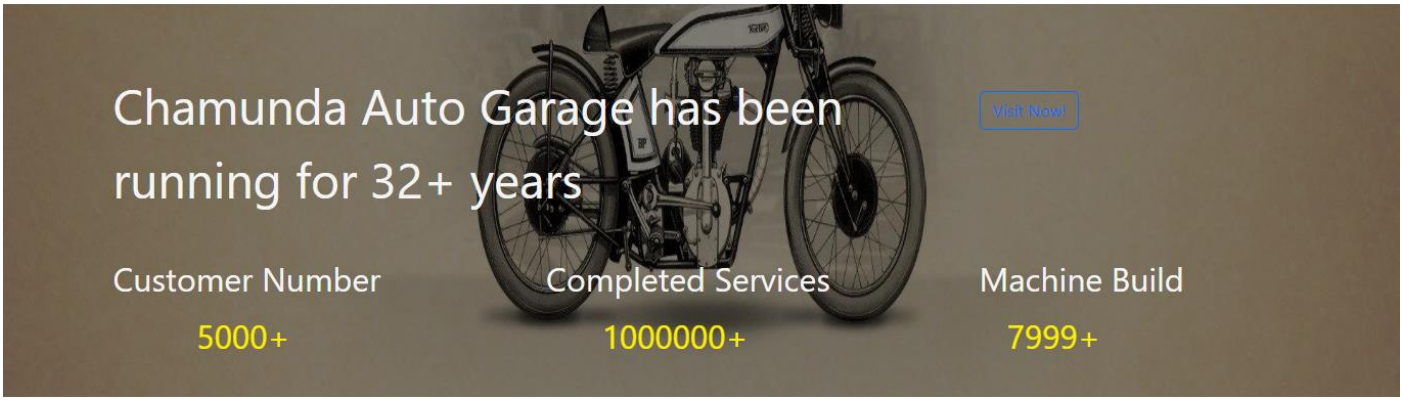
1. Index.php



2. Index.php (service component)

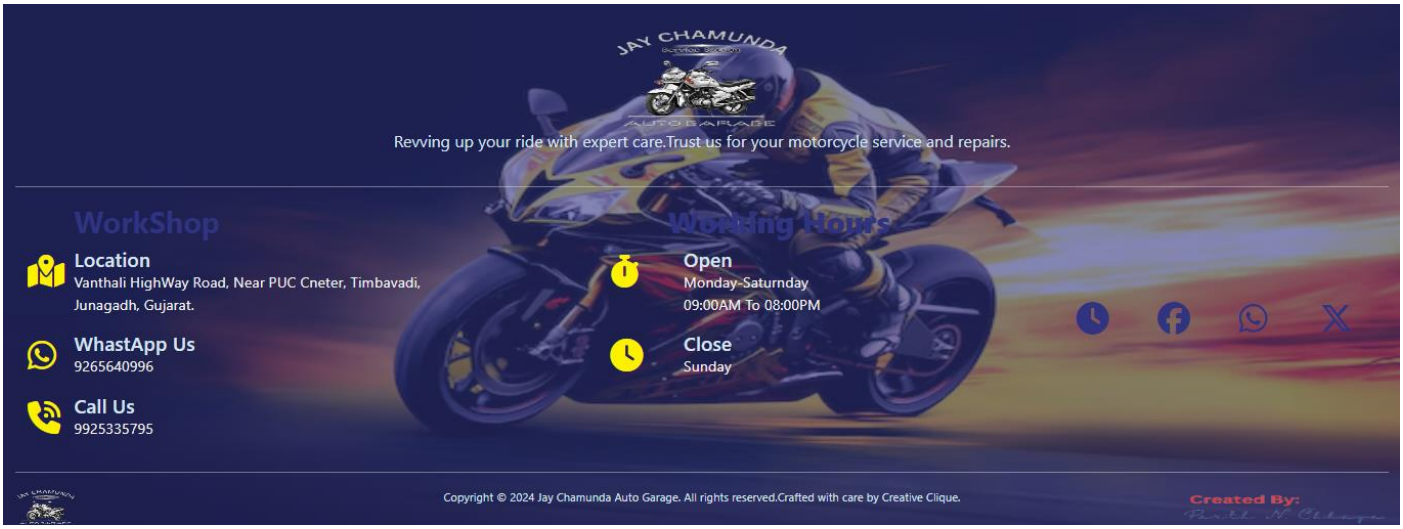


3. Index.php (History Component)

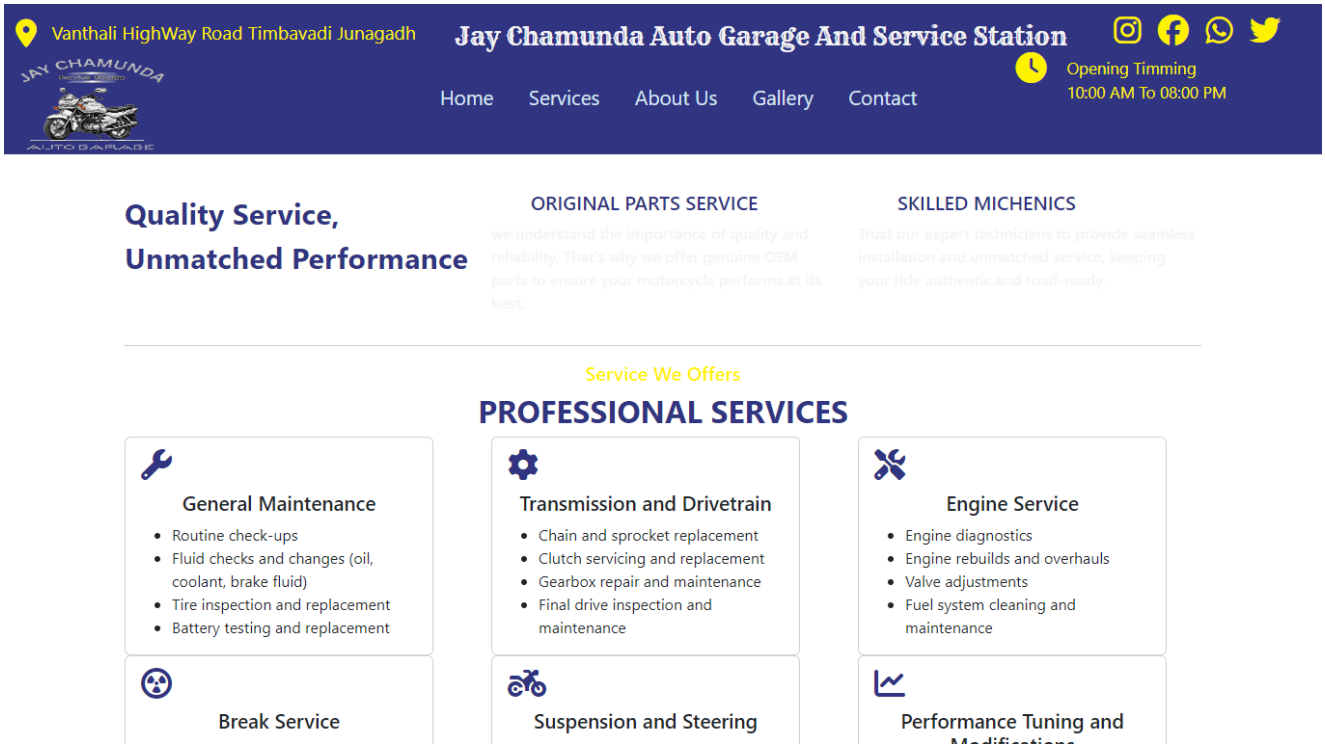


4. Index.php (Map Component)

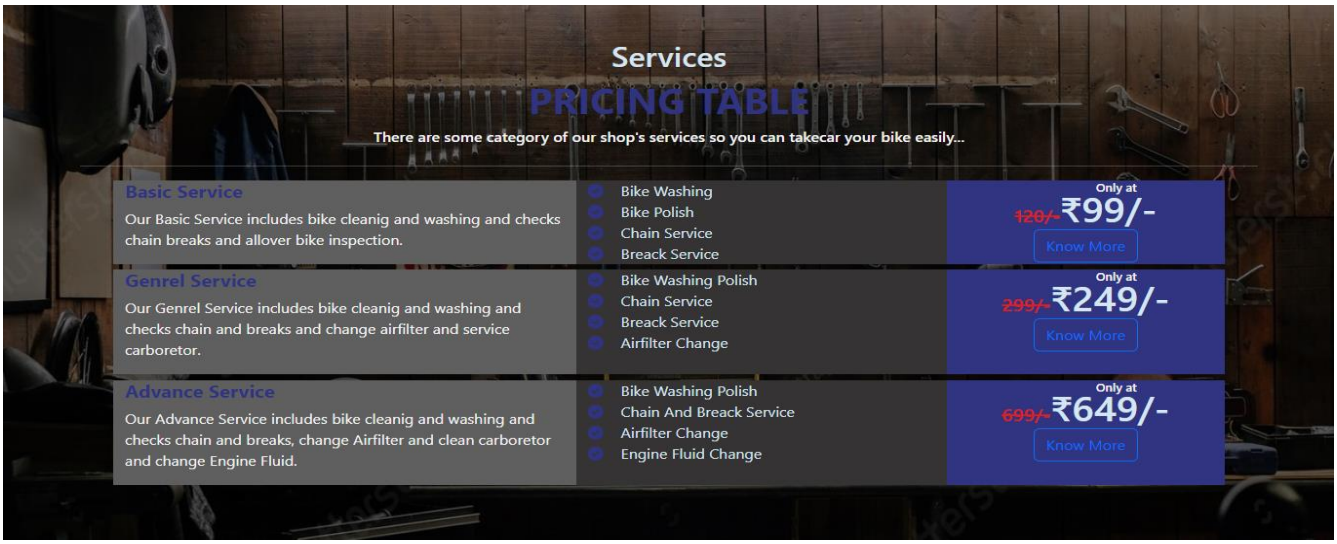
5. Index.php (Footer)



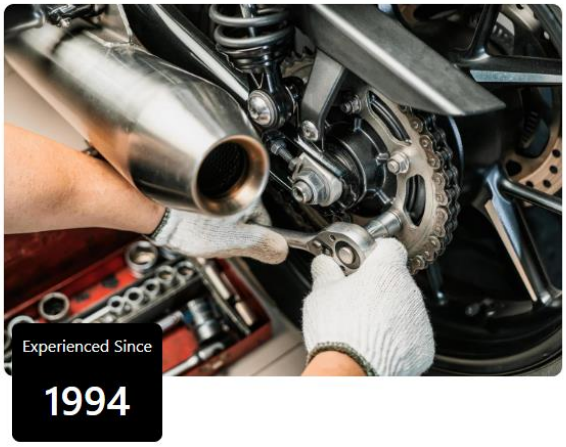
6. Service.php



7. Service.php (Price)



8. Service.php (Advantage & Emergency Service)



OUR ADVANTAGES

WHY CHOOSE US

Our team consists of certified motorcycle specialists with years of experience in repair, maintenance, and customization. We are equipped with the latest tools and technology.

Expert Mechanics

Reasonable Price

Fast Work & Services

Trusted & Professional

Emergency Services Available..! Call Us Now!

9. aboutus.php

Vanthali HighWay Road Timbavadi Junagadh

Jay Chamunda Auto Garage And Service Station

Opening Timming
10:00 AM To 08:00 PM

Home Services About Us Gallery Contact

About Us

"At our motorcycle garage, we're more than just mechanics – we're a team of passionate riders dedicated to delivering top-notch service, quality repairs, and custom builds that reflect the unique spirit of every bike and rider. Driven by craftsmanship and a love for motorcycles, we strive to provide an exceptional experience for our customers on every ride."

10. aboutus.php (first component)

ABOUT US

Top-Quality Craftsmanship with Hands-On Care.

Welcome to Jay Chamunda Auto Garage And Service Station, your go-to motorcycle garage for top-quality repairs, maintenance, and customization. Located in Junagadh Gujarat, we specialize in providing expert services. Our skilled mechanics are passionate riders themselves, bringing years of experience and a commitment to excellence to every job. Whether you need routine maintenance, complex repairs, or custom builds, we've got you covered.

At Jay Chamunda Auto Garage And Service Station, we pride ourselves on delivering exceptional customer service and reliable, honest work. We use only the best parts and equipment to ensure your motorcycle is road-ready and performing at its best. Our shop also features a wide selection of motorcycle accessories, tires, and performance upgrades to meet every rider's needs.

Join our growing community of satisfied riders who trust us to keep their bikes in peak condition. Visit us today and experience the Jay Chamunda Auto Garage And Service Station difference!

1994 30+

11.aboutus.php (Second Component)

PROVIDING FULL RANGE OF CUSTOM SERVICE

We Build for the Ultimate Ride!

At Jay Chamunda Auto Garage, we provide a comprehensive range of custom services tailored to meet the unique needs of every rider. From routine maintenance and repairs to complete overhauls and custom builds, our skilled mechanics are committed to delivering exceptional craftsmanship and quality.

We understand that every bike has its own story and personality, which is why we take a personalized approach to every project, ensuring your motorcycle performs at its best and reflects your style.

Nareshbhai Chhaya

Founder Jay Chamunda Auto Garage And Service Station

Nareshbhai Chhaya

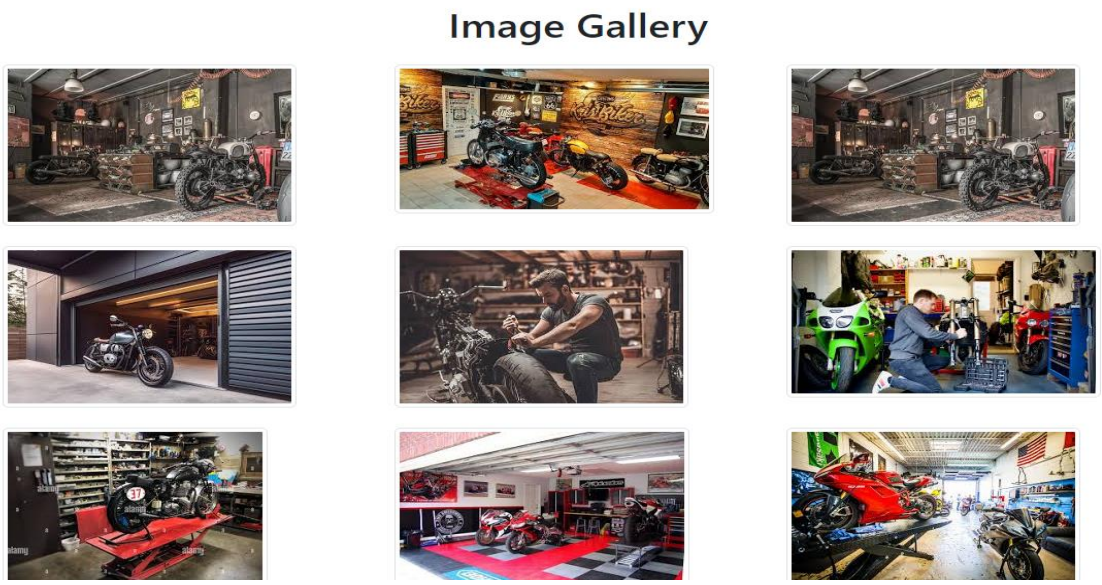
» Our Mission

"Our mission is to deliver exceptional, customized motorcycle care and craftsmanship, ensuring every rider experiences the ultimate ride with safety, style,

» Our Vision

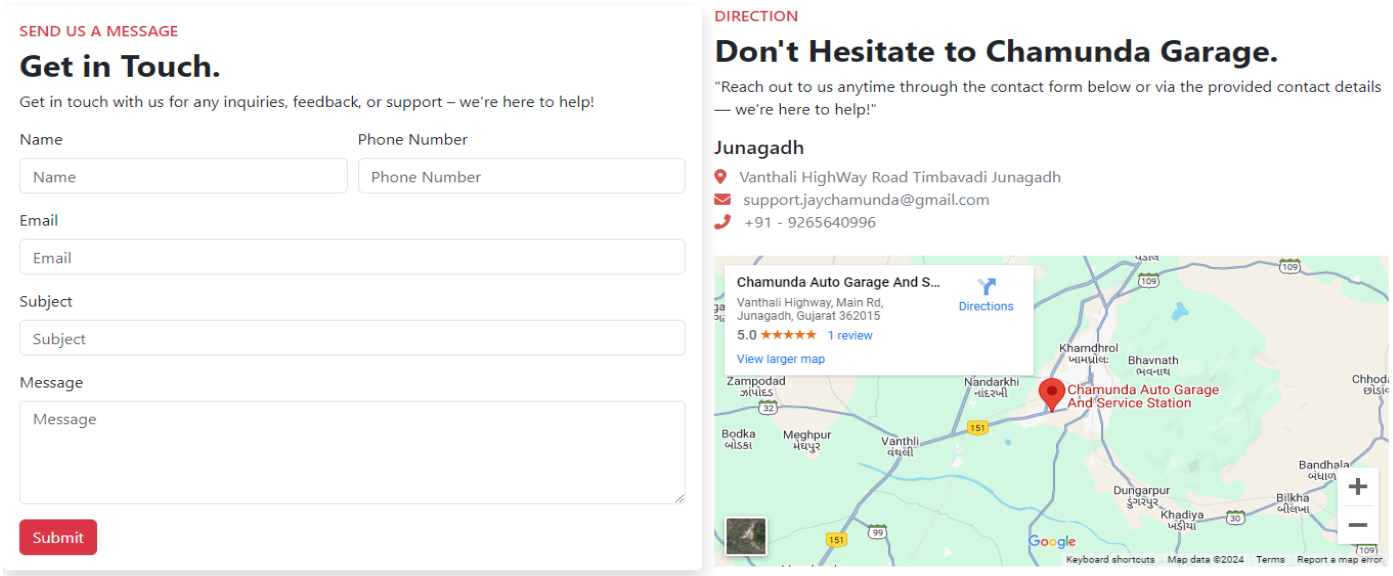
"Our vision is to be the leading motorcycle garage that combines expert craftsmanship, personalized service, and a passion for motorcycles to deliver the

12.gallery.php




```php
// Fetch images from the database
            $sql = "SELECT * FROM gallery";
            $result = $conn->query($sql);

            if ($result->num_rows > 0) {
                while ($row = $result->fetch_assoc()) {
                    echo '<div class="col-lg-4 col-md-6 col-sm-12 mb-4 img_con">
                            <img src="../uploads/' . $row["image"] . '" class="img-fluid
img-thumbnail" alt="Gallery Image">
                          </div>';
                }
            } else {
                echo '<p>No images found in the gallery.</p>';
            }

            $conn->close();
```

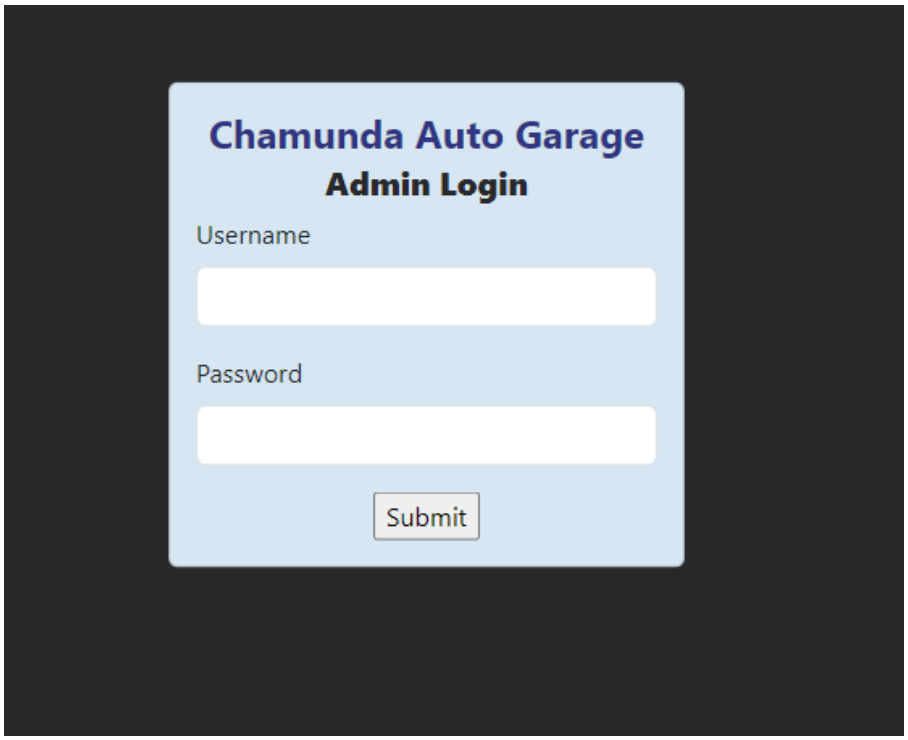
13.contactus.php




```
$query = "INSERT INTO feedback (nam, email, phone, subjec, messag) VALUES ('$nm', '$em', '$ph', '$sub', '$mgs')";
    $data = mysqli_query($conn, $query);
```

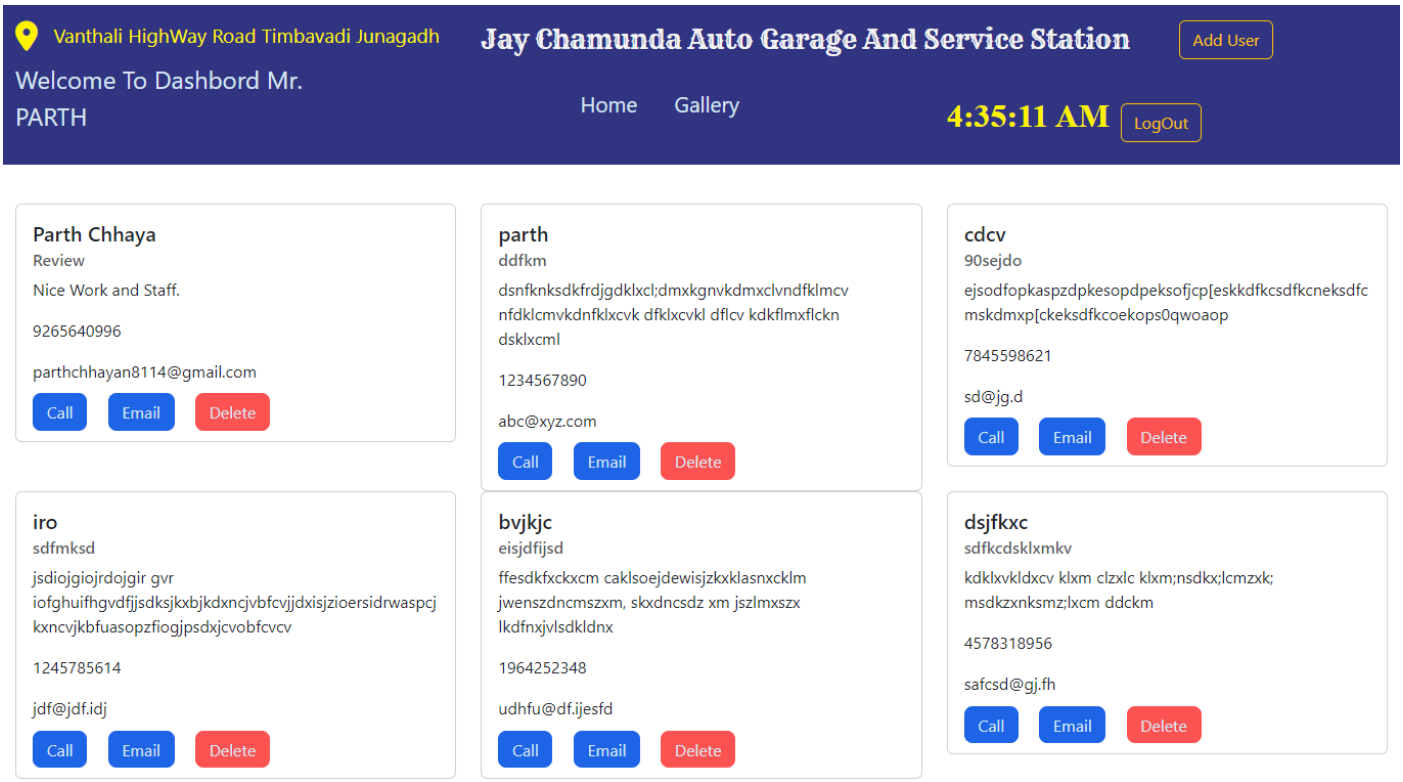
- **Admin Side:**

1. **Login.php**



```
$query = "SELECT * FROM aaa WHERE user = '$user' AND pas = '$pass'";
    $data = mysqli_query($conn, $query);
    $total = mysqli_num_rows($data);
```

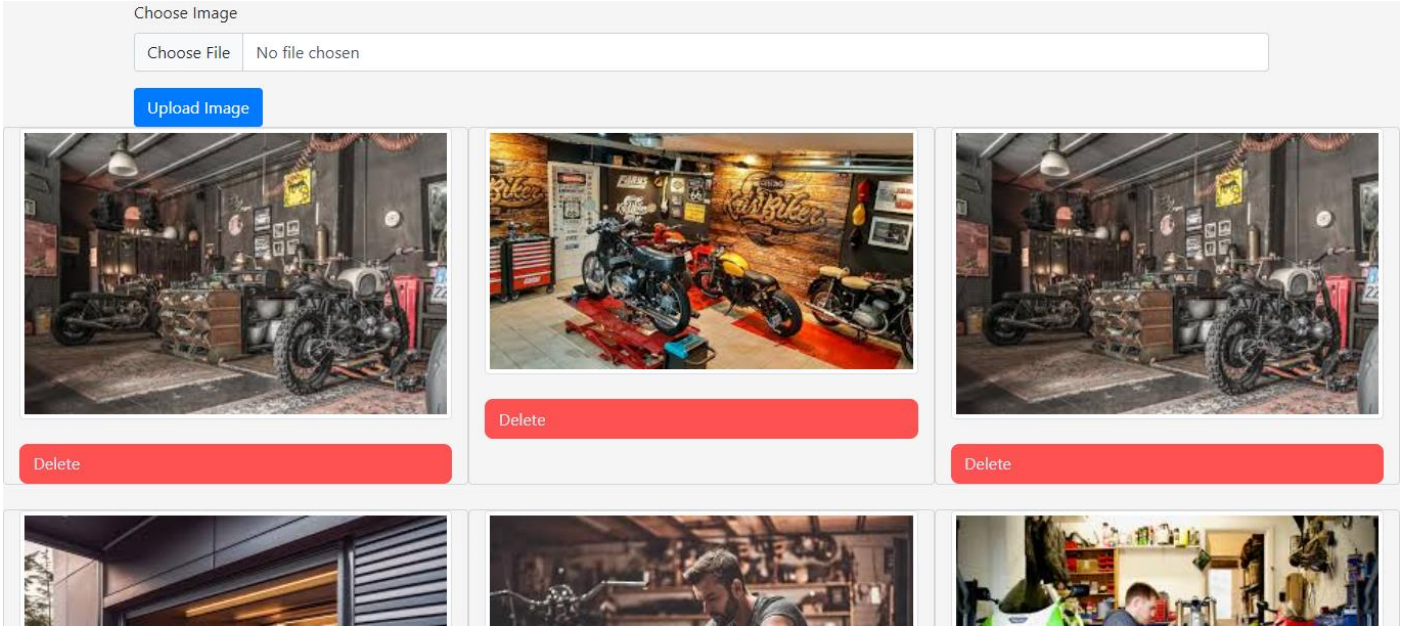
2. Dashboard.php




```
$query = "SELECT * FROM feedback";
            $data = mysqli_query($conn, $query);

            if ($data->num_rows > 0) {
                while ($row = $data->fetch_assoc()) {

                    echo '<div class="col-md-4"><!-- Main Column Starting -->';
                    echo '<div class="card"><!-- Card Starting -->'; // Removed fixed
width here
                    echo '<div class="card-body"><!-- Card Body Starting -->';

                    echo '<h5 class="card-title">' . htmlspecialchars($row['nam']) .
'</h5>';
                    echo '<h6 class="card-subtitle mb-2 text-body-secondary"> ' .
htmlspecialchars($row['subjec']) . '</h6>';
                    echo '<p class="card-text"> ' . htmlspecialchars($row['messag']) .
'</p>';
                    echo '<p class="card-text">' . htmlspecialchars($row['phone']) .
'</p>';
```

```
                    echo '<p class="card-text">' . htmlspecialchars($row['email']) .
'</p>';
                    echo ' <a href="tel: '. htmlspecialchars($row['phone']) .' " '
.'class="card-link crd_btn">Call</a>';
                    echo ' <a href="mailto: '. htmlspecialchars($row['email']) .' " '
.'class="card-link crd_btn">Email</a>';
                    echo '  <a href="delete.php?id=' . $row['id'] . ' " ' . 'class="card-
link crd_det">Delete</a>';

                    echo '</div> <!-- Card Body Ending -->';
                    echo '</div> <!-- Card Ending -->';

                    echo '</div> <!-- Main Column Ending -->';
                }
            } else {
                echo '<p>No feedback found.</p>';
            }
```

3. Gallery.php

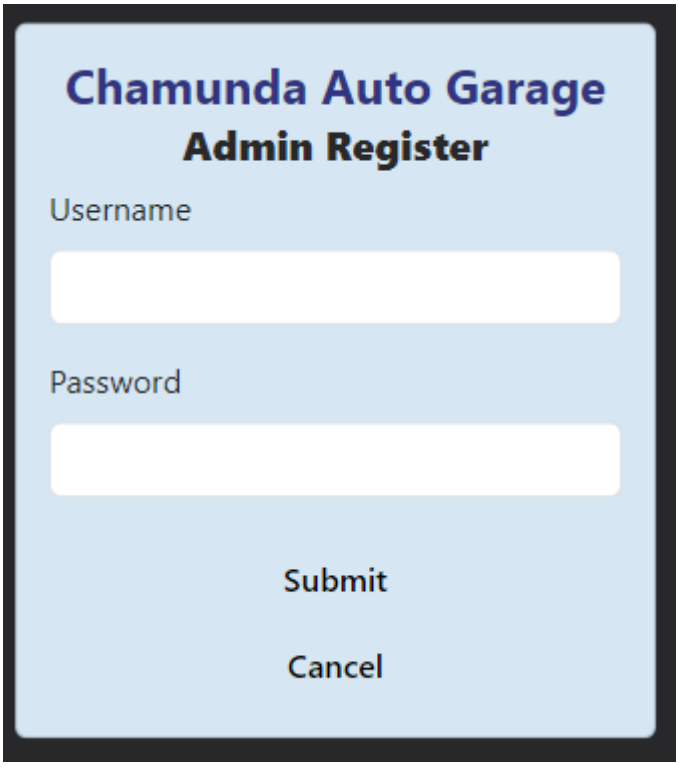


```php
if ($_SERVER['REQUEST_METHOD'] == 'POST') {

    // Handle image upload
    if (isset($_FILES['image']) && $_FILES['image']['error'] === UPLOAD_ERR_OK) {
        $image = $_FILES['image'];
        $imageName = basename($image['name']);
        $uploadDir = '../uploads/';

        // Move the uploaded file to the uploads folder
        if (move_uploaded_file($image['tmp_name'], $uploadDir . $imageName)) {
            // Insert the image name into the database
            $sql = "INSERT INTO gallery (image) VALUES ('$imageName')";
            if ($conn->query($sql) === TRUE) {
                // echo "Image uploaded successfully!";
                ?>

                <script>

                    alert("Image uploaded successfully!");

                </script>

                <META http-equiv="refresh" content="0;gallery.php">


                <?php


            } else {
                echo "Error: " . $sql . "<br>" . $conn->error;
            }
        } else {
            echo "Failed to move uploaded file.";
        }
    } else {
        echo "No file uploaded or error during upload.";
    }

    $conn->close();
```

```php
$sql = "SELECT * FROM gallery";
            $result = $conn->query($sql);

            if ($result->num_rows > 0) {
                while ($row = $result->fetch_assoc()) {
                    echo '<div class="col-lg-4 col-md-6 col-sm-12 mb-4 card img_con">
                            <img src="../uploads/' . $row["image"] . '" class="img-fluid
img-thumbnail" alt="Gallery Image">
                            <br>
```

```
                            <a href="imgdelete.php?id=' . $row["id"] . ' " class="det">
Delete </a>
                        </div>';
                }
            } else {
                echo '<p>No images found in the gallery.</p>';
            }

            $conn->close();
```

4. Adduser.php




```
$query = "INSERT INTO aaa (user, pas) VALUES ('$user', '$pass')";
   $data = mysqli_query($conn, $query);
```

SPECIAL UTILITIES

1) Secure Login System

- A secure login system ensures that only authenticated users can access the platform.

2) Feedback System

- Client or user can give feedback about their experience with the garage and website. Those feedbacks are shown at the admin panel.

3)Dynamic Image Gallery

- All images which shown at gallery page is called dynamically from database and can be add or delete from admin panel.

TESTING

Testing is a process of executing a program with the indent of finding an error. Testing is a crucial element of software quality assurance and presents ultimate review of specification, design and coding.

System Testing is an important phase. Testing represents an interesting anomaly for the software. Thus a series of testing are performed for the proposed system before the system is ready for user acceptance testing.

A good test case is one that has a high probability of finding an as undiscovered error. A successful test is one that uncovers an as undiscovered error.

Testing Objectives:

1.Testing is a process of executing a program with the intent of finding an error

2.A good test case is one that has a probability of finding an as yet undiscovered error

3.A successful test is one that uncovers an undiscovered error

Testing Principles

1. All tests should be traceable to end user requirements
2.Tests should be planned long before testing begins

3.Testing should begin on a small scale and progress towards testing in large
4.Exhaustive testing is not possible
5.To be most effective testing should be conducted by a independent third party

The primary objective for test case design is to derive a set of tests that has the highest livelihood for uncovering defects in software. To accomplish this objective two different categories of test case design techniques are used. They are

White box

Black box

White-box testing:

White box testing focus on the program control structure. Test cases are derived to ensure that all statements in the program have been executed at least once during testing and that all logical conditions have been executed.

Block-box testing:

Black box testing is designed to validate functional requirements without regard to the internal workings of a program. Black box testing mainly focuses on the information domain of the software, deriving test cases by partitioning input and output in a manner that provides through test coverage. Incorrect and missing functions, interface errors, errors in data structures, error in functional logic are theerrors falling in this category.

Testing strategies:

A strategy for software testing must accommodate low-level tests that are necessary to verify that all small source code segment has been correctly implemented as well as high-level tests that validate major system functions against customer requirements.

Testing fundamentals:

Testing is a process of executing program with the intent of finding error. A good test case is one that has high probability of finding an undiscovered error. If testing is conducted successfully it uncovers the errors in the software. Testing cannot show the absence of defects, it can only show that software defects present.

Testing Information flow:

Information flow for testing flows the pattern. Two class of input provided to test the process. The software configuration includes a software requirements specification, a design specification and source code.

Test configuration includes test plan and test cases and test tools. Tests are conducted and all the results are evaluated. That is test results are compared with expected results. When erroneous data are uncovered, an error is implied and debugging commences.

Unit testing:

Unit testing is essential for the verification of the code producedduring the coding phase and hence the goal is to test the internal logic of the modules. Using the detailed design description as a guide, important paths are tested to uncover errors with in the boundary of the modules. These tests were carried out during the programming stage itself. All units of ViennaSQL were successfully tested.

Integration testing:

Integration testing focuses on unit tested modules and build the program structure that is dictated by the design phase.

System testing:

System testing tests the integration of each module in the system. It also tests to find discrepancies between the system and it's original objective, current specification and system documentation. The primary concern is the compatibility of individual modules. Entire system is working properly or not will be tested here, and specified path ODBC connection will correct or not, and giving output or not are tested here these verifications and validations are done by giving input values to the system and by comparing with expected output. Top-down testing implementing here.

Acceptance Testing:

This testing is done to verify the readiness of the system for the implementation. Acceptance testing begins when the system is complete. Its purpose is to provide the end user with the confidence that the system is ready for use. It involves planning and execution of functional tests, performance tests and stress tests in order to demonstrate that the implemented system satisfies its requirements.

Tools to special importance during acceptance testing include:

Test coverage Analyzer – records the control paths followed for each test case.

Timing Analyzer – also called a profiler, reports the time spent in various regions of the code are areas to concentrate on to improve system performance.

Coding standards – static analyzers and standard checkers are used to inspect code for deviations from standards and guidelines.

Test Cases:

Test cases are derived to ensure that all statements in the program have been executed at least once during testing and that all logical conditions have been executed. Using White-Box testing methods, the software engineer can drive test cases that

- Guarantee that logical decisions on their true and false sides.
- Exercise all logical decisions on their true and false sides.
- Execute all loops at their boundaries and with in their operationalbounds.
- Exercise internal data structure to assure their validity.

The test case specification for system testing has to be submitted for review before system testing commences.

IMPLIMENTATION

This application avoids the manual work and the problems concern with it. It is an easy way to obtain the information regarding thedifferent scheduled examinations information that are Currently issued.

Well I and my team members have worked hard in order to present an improved website better than the existing one's regarding the information about the various activities. Still ,we found out that the project can be done in a better way. Primarily, when we request information about a particular schedules it just shows the exam date and platform. So, after getting the information we can get access to the online exam.

The enhancement that we can add the searching option. We can directly search to the particular student details from this site.

The implementation is processed from review and reports from developer cover the following areas:

- Good working conditions.
- Useful for gathering information.
- Update website easily.
- Attractive layouts.
- Working for as per requirements.

BIBLIOGRAPHY

The following references were utilized in the development of this project. They include video tutorials, Reference Websites, Learning Websites, Libraries, GitHub repositories, and other relevant resources.

1) Video Tutorials

- **Html and CSS:**

  https://youtube.com/playlist?list=PLu0W_9lII9agq5TrH9XLIKQvv0iaF2X3w&si=P0b2l99KjzykhLGa

- **PHP and CURD operations:**

  https://youtube.com/playlist?list=PLejKBGxF74J67G7KSGHA4JPOKW9H3cQq3&si=cNAy4DOiJUlzE8Va

- **Bootstrap:**

  https://youtube.com/playlist?list=PL0b6OzIxLPbz1cgxiH5KCBsyQij1HsPtG&si=QzJCBQ3xUgpw0V9k

- **JavaScript:**

https://youtube.com/playlist?list=PLu71SKxNbfoBuX3f4EOACle2y-tRC5Q37&si=wWbN7Pwymg9JP5vS

2) Reference Websites:

https://preview.themeforest.net/item/moge-motorcycle-repair-service-elementor-template-kit/full_screen_preview/32202947?_ga=2.156101181.764212109.1709842613-1462368451.1709842613

https://preview.themeforest.net/item/gastros-garage-motorcycle-service-repair-elementor-template-kit/full_screen_preview/33101577?_ga=2.156101181.764212109.1709842613-1462368451.1709842613

https://preview.themeforest.net/item/motohero-motorcycle-repair-custom-service-business-theme/full_screen_preview/15803039?_ga=2.177129891.764212109.1709842613-1462368451.1709842613

- **For show the oldness of shop banner**

https://preview.themeforest.net/item/motocross-motorcycle-atv-wordpress-theme/full_screen_preview/21042666?_ga=2.98575388.304660387.1719674408-1889579695.1719505924

- **For Services Card Design**

https://preview.themeforest.net/item/keymoto-motorcycle-html-template/full_screen_preview/29947988?_ga=2.56413993.304660387.1719674408-1889579695.1719505924

- **For Color Pallet**

https://colorhunt.co/palette/b6eb7af7f7eefb781317706e

3) Learning Websites

- **W3Schools**
  https://www.w3schools.com/html/
  https://www.w3schools.com/w3css/defaulT.asp
  https://www.w3schools.com/bootstrap5/
  https://www.w3schools.com/js/
  https://www.w3schools.com/jquery/
  https://www.w3schools.com/php/
  https://www.w3schools.com/MySQL/default.asp

4) Libraries

- **For Animation**

https://preview.themeforest.net/item/bikersclub-motorcycle-responsive-wordpress-theme/full_screen_preview/20699982?_ga=2.62318382.304660387.1719674408-1889579695.1719505924

- **For Bootstrap**
  https://getbootstrap.com/

- **For JQuery**
  https://www.jqueryscript.net/tags.php?/Carousel/