

AN INTELLIGENT VEHICLE BREAKDOWN ASSISTANCE MANAGEMENT SERVICES

A PROJECT REPORT

Submitted by

SACHIN RAO D . (610820205038)

KISHORE S (610820205013)

RANJITH KUMAR S (610820205035)

SRINIVASAN B (610820205046)

THENNARASU C (610820205056)

in partial fulfilment for the award of the degree of

BACHELOR OF TECHNOLOGY

IN

INFORMATION TECHNOLOGY

**Er. PERUMAL MANIMEKALAI COLLEGE OF ENGINEERING
HOSUR**

ANNA UNIVERSITY :: CHENNAI 600 025

MAY 2024

BONAFIDE CERTIFICATE

This is to certify that this project report “**AN INTELLIGENT VEHICLE BREAKDOWN ASSISTANCE MANAGEMENT SERVICES** ” is the Bonafide work of “**SACHIN RAO .D(610820205038), KISHORE .S (610820205014), RANJITH KUMAR .S (610820205035), SRINIVASAN .B (610820205046), THENNARASU .C (610820205056)**” carried out the project work under my supervision.

SIGNATURE

MR.A.RICHARD WILLAM, M.E(PHD)

HEAD OF THE DEPARTMENT

Assistant Professor

Department Of Information Technology

Er. Perumal Manimekalai College Of

Engineering,Hosur-635 117.

SIGNATURE

MRS.S.SURYA,M.E

SUPERVISOR

Assistant Professor

Department Of Information Technology

Er. Perumal Manimekalai College Of

Engineering, Hosur-635 117.

Submitted for the University project viva-voce Examination held onat

ER.PERUMAL MANIMEKALAI COLLEGE OF ENGINEERING,HOSUR.

EXTERNAL EXAMINER

INTERNAL EXAMINER

ACKNOWLEDGMENT

We wish to express our sincere thanks to all those were involved in the completion of this project.

Our most sincere salutations go to **ANNA UNIVERSITY, Chennai** that gave us an opportunity to have sound base of Information Technology.

We also express our profound thank to our Chairman **Shri. Er.P. PERUMAL, B.E., M.I.E., F.I.V.**, our Secretary **Shri. P.KUMAR, A.M.I.E.**, and **smt. P.MALAR**, for their support and encouragement.

We offer our sincere thank to our Principal **Dr. S.CHITHRA, M.E., Ph.D.**, for their valuable suggestions which edified us with zest to carry out the work.

We offer our sincere thank to **Mr. A. RICHARD WILLIAM, M.E., Ph.D.**, our beloved Head of the Department of Information technology, Er. Perumal Manimekalai College of Engineering, for her encouragement.

We especially thank for **Mrs. S.SURYA, M.E.**, our guide for assisting us with her valuable suggestions and technical guidance to complete the project.

We also express our sincere thank to our Department Faculty for their valuable support.

Last but not least we extend our thank to our beloved parents for their moral support and blessing.

TABLE OF CONTENT

| CHAPTER | CONTENTS | PAGE NO |
|----------|---|-----------|
| | ABSTRACT | |
| 1 | INTRODUCTION | |
| | Overview | 1 |
| | Objective | 2 |
| 2 | LITERATURE SURVEY | 3 |
| 3 | EXISTING SYSTEM &PROPOSED SYSTEM | |
| | Existing System | 7 |
| | Proposed System | 8 |
| 4 | SYSTEM FUNCTION | |
| | Architecture Design | 10 |
| | System Modules | 11 |
| 5 | SYSTEM SPECIFICATION | |
| | Hardware Specification | 13 |
| | Software Specification | 13 |
| | Enter Customer Register Details | 13 |
| | Enter Vehicle Details | 15 |
| | View Vehicle List | 18 |
| | Search Vehicle by Customer Name | 19 |
| | View Vehicle Available | 21 |

| | | |
|-----------|--|-----------|
| | GPS Tracker | 22 |
| | Database Design | 24 |
| 6 | SYSTEM SOFTWARE | |
| | Python Programming Language | 29 |
| | Visual Studio Code | 31 |
| | Running and Debugging | 39 |
| | Understanding Breakpoints | 41 |
| 7 | HARDWARE REQUIREMENTS AND SPECIFICATION | |
| | I5 Processor | 43 |
| | Random Access Memory | 45 |
| 8 | CODING | |
| | HTML | 48 |
| | CSS | 49 |
| | Python | 50 |
| | Data base | 51 |
| 9 | OUTPUT | 52 |
| 10 | RESULT | 56 |
| 11 | CONCLUSION | 57 |
| 12 | REFERENCE | 58 |

LIST OF FIGURES

| FIGURES NO | NAME | PAGE NO |
|------------|----------------------------------|---------|
| 1 | Proposed for architecture design | 10 |
| 2 | Enter Customer Register Details | 15 |
| 3 | Enter Vehicle details | 18 |
| 4 | View vehicle list | 19 |
| 5 | Search Vehicle | 20 |
| 6 | Vehicle Breakdown Management | 21 |
| 7 | GPS Tracking | 24 |
| 8 | Data Base Management System | 26 |
| 9 | VS Code software | 34 |
| 10 | HTML | 48 |
| 11 | CSS | 49 |
| 12 | Python | 50 |
| 13 | Database | 51 |
| 14 | Home Page | 52 |
| 15 | Services Page | 52 |
| 16 | Features | 53 |
| 17 | Membership | 53 |
| 18 | Customer Feedback Page | 54 |
| 19 | Customer Login & Sign up | 54 |
| 20 | Admin Page | 55 |

ABSTRACT

In today's fast-paced world, vehicle breakdowns can occur unexpectedly, causing inconvenience and disruption to individuals' daily routines. Prompt assistance during such situations is crucial to ensure the safety and well-being of vehicle owners and passengers. To address this need, this project aims to develop a comprehensive vehicle breakdown management system equipped with user-friendly interfaces for both customers and administrators. The system facilitates customers to register their details securely by providing essential information such as username, password, phone number, and email ID. Upon successful registration, customers can log in using their credentials and proceed to submit their vehicle details, including vehicle name, date, vehicle number, and the nature of the problem, such as oil leakage, breakdown, puncture, engine failure, etc and location. The admin interface allows authorized personnel to log in and access a list of problematic vehicles reported by customers with their current GPS location. Administrators can efficiently manage the reported issues by sorting and filtering them based on various parameters such as date and customer name. This enables administrators to quickly identify and address vehicle breakdowns, enhancing the efficiency and effectiveness of the breakdown management service. Overall, this system streamlines the process of reporting and resolving vehicle breakdowns, ensuring prompt assistance to customers while providing administrators with the tools to manage breakdown incidents effectively.

CHAPTER 1

INTRODUCTION

1.1 OVERVIEW

In today's society, where mobility is a cornerstone of daily life, vehicle breakdowns can pose significant challenges, disrupting routines and causing inconvenience to individuals and families alike. This system is designed to provide efficient and user-friendly interfaces served for both customers and administrators. By enabling customers to securely register their details, including vital information such as username, password, phone number, and email ID, the system ensures smooth access and privacy protection. Upon successful registration, customers can log in and swiftly submit detailed vehicle information, including vehicle name, date, vehicle number, and the specific nature of the problem encountered, whether it be oil leakage, breakdown, puncture, engine failure, or others. Administrators can log in to access a comprehensive list of problematic vehicles reported by customers, equipped with access to GPS location data's also sorting and filtering functionalities based on parameters such as date and customer name. This capability empowers administrators to swiftly identify and address vehicle breakdowns, thereby enhancing the efficiency and effectiveness of the breakdown assistance service. By bridging the gap between vehicle owners and service providers, the system aims to minimize disruptions caused by breakdowns and contribute to a safer and more reliable mobility experience for all.

1.2 OBJECTIVE

- Enhance user experience by implementing intuitive and responsive user interfaces for both customers and administrators.
- Develop a user-friendly vehicle breakdown management system to provide prompt assistance to customers during unexpected breakdown situations.
- Enable customers to register securely by providing essential details such as username, password, phone number, and email ID, ensuring smooth access to the system.
- Implement a login mechanism for customers to access the system using their credentials, ensuring privacy and security.
- Allow customers to submit detailed vehicle information, including vehicle name, date, vehicle number, and the specific nature of the problem encountered.
- Implement GPS location tracking to accurately pinpoint the location of reported breakdowns.
- Develop an admin interface to enable authorized personnel to log in and access a list of reported problematic vehicles for efficient management.
- Implement sorting and filtering functionalities within the admin interface to enable quick identification and resolution of breakdown incidents based on parameters such as date and customer name.
- Enhance the efficiency and effectiveness of the breakdown management service by providing administrators with tools to manage reported issues effectively.

CHAPTER 2

LITERATURE SURVEY

[1] Varun Kapadi, Saigita Guruju etc. (2021)” Emergency Breakdown Services Using Android Application” (IRJET) vol:04 Issue:04, Apr-2021.

Varun Kapadi, Saigita Guruju etc. (2021)” Emergency Breakdown Services Using Android Application”. In this paper, the proposed emergency breakdown service provides a user-friendly environment. It is the easiest way to identify the location as well as the nearby needed locations. This application is made with the thought in mind, that after an accident or a breakdown, the user is in a state of panic. This interface of the application is designed in a simplistic way, with direct access to services provided by the application. The application provides nearby location information such as a petrol pump, police station, service station, and hospital

[2] M.A. D Wickrama, D.S.C.Dharmakeerthi etc. (2021)”Mobile Based Solution for Vehicle Assistance”(IEEE) DOI: 10.1109/ICAC54203.2020.9671196.

M.A.D Wickrama, D.S.C.Dharmakeerthi, S.A. Balasooriya (2020) proposed ”Mobile Based Solution for Vehicle Assistance”. Authors decided to develop a mobile application to assist drivers with vehicle breakdowns and system’s major functionality including providing a personal assistant to aid with vehicle breakdowns, predicting upcoming vehicle maintenance procedures, providing trustworthy spare part pricing, and recommending the best insurance providers to customers. This will enhance the decision making ability of the customers.

[3] Monika Kadam, Neelima Sutar, Pooja Dorge, etc. (2018) "A Car Breakdown Service Station Locator System" (ISSN) vol:03 Issue:04, Apr-2021.

Harsha Supare, Kanchan Yadav, etc. (2018) "A Car Breakdown Service Station Locator System". The suggested system creates a connection between Car Repair Service Providers (CASP) and the public. The system will search for any CASPs that are close to the reported incident location. Users can contact service provider CASP to get in touch with the service provider closest to their area. The services are made available together with the service provider's information, which the traveler can access. With Google API for map services, passengers are informed about service availability and accessibility.

[4] Akhila V Khanpuri, Anagha Shastri, etc. (2021) "On Road: A car assistant application" (IEEE) ICTSD-2021

Akhila V Khanpuri, Anagha Shastri, Gareth D'souza, Shannon D'souza (2021) had proposed "On Road: A Car Assistant Application". As there is an increase in the rise in prices of fuel and an exponential increase in the number of cars on the road, road accidents, and vehicle case breakdowns are recorded. An Android-based application is proposed to find an effective way to achieve maximum fuel efficiency and monitors parameters like Engine RPM, fuel status, and throttle position through Onboard Diagnostics (OBD-II) which will help and assist the drivers and provide assistance in case of breakdown of a vehicle and also assist the drivers to achieve better fuel economy.

[5] Shivangi Patidar¹, Shivam Patel², Naman Mehta³, Vikas Ojha⁴, Suveer Chandra Dubey⁵ ^{1, 2, 3, 4}UG Student, ⁵Assistant Professor, 'Mechanism of Inbuilt Automatic Hydraulic Jack used for Light and Heavy Vehicles'

Department of Mechanical Engineering, Indore Institute of Science and Technology, Indore, 2020.

This paper is about lifting the vehicle for changing the tire if it is punctured. This mechanism helps to reduce human efforts as the system is operated by hydraulic actuators. The system is a combination of Mechanical, Hydraulic and Electric components. The hydraulic actuator is mounted on both sides of the vehicle; each side is operated by a different button. If the left side of the vehicle needs to be lifted, then the left button is pressed and so for the right side.

[6] R.Sureshkumar, 2 V.Vijaykumar, 3 R.Vinoth, 4 M.Yogeshwaran, 5 R.Moses Samuel 1 Assistant Professor, 2,3,4,5UG Students, Department of Mechanical Engineering, ‘ Fabrication of Inbuilt Electrohydraulic Jack for Four Wheelers’ Gnanamani College of Technology, Namakkal,Tamilnadu, 2019.

This paper is about lifting a vehicle using an Electrohydraulic jack. This mechanism helps to reduce efforts that are required to lift the vehicle using a manual hydraulic jack or a normal screw jack. As this mechanism is operated using an electric motor, the time required is less for lifting, so much of the time is saved. The electrohydraulic jack is indirectly used to apply force on the link which is in contact with the ground.

[7] Prashant D.Banakar^{1*}, Prashantakumar N. K. ^{2*}, Hanumesha Pujar³ Department of Mechanical Engineering, K.L.E. ‘ Design and Fabrication of Inbuilt Hydraulic Jack for Four Wheelers’ Institute of Technology, Hubballi, 2019.

This paper is about a lifting mechanism which is operated using a hydraulic system. In this, a vane pump is used to pressurize fluid. The hydraulic actuators are attached to the vehicle – 2 in front and 2 at back. These front and back side actuators are operated

using different electrical switches so that by pressing one switch front will lift and another switch will lift the back side. This mechanism is used for breakdown of car or for servicing of car or tire puncture.

CHAPTER 3

EXISTING AND PROPOSED SYSTEM

3.1 EXISTING SYSTEM

- **Manual Reporting:** Currently, in the absence of a dedicated system, customers need to manually report vehicle breakdowns to service centers or roadside assistance providers.
- **Phone Calls:** Customers typically rely on phone calls to notify service providers about their vehicle problems. This process can be time-consuming and may lead to delays in assistance.
- **Limited Information Capture:** When customers report breakdowns over the phone, there's a limitation in capturing detailed information about the vehicle and the nature of the problem. This lack of detailed information can hinder the effectiveness of the response.
- **Administrative Challenges:** Service providers face administrative challenges in managing breakdown reports efficiently.
- Service providers may struggle to allocate resources effectively without proper visibility into the volume and nature of breakdown reports in different areas.
- **Risk of Data Loss:** With manual record-keeping systems, there's a risk of data loss or damage due to human error, misplacement of documents, or physical damage to records.

3.1.1 DISADVANTAGE:

- Traditional vehicle breakdown assistance systems may lack accessibility, especially in remote or less developed areas where service coverage is limited. This can lead to delays in assistance for stranded vehicle owners.
- Many existing systems rely on manual reporting of breakdown incidents by customers, either through phone calls or physical forms. This manual process can be inefficient, prone to errors, and time-consuming for both customers and service providers.
- In some cases, there may be a lack of efficient information sharing between customers, service providers, and administrators. This can lead to delays in coordinating management efforts and resolving breakdown incidents promptly.

3.2 PROPOSED SYSTEM

- Customers can register their details online, providing essential information such as username, password, phone number, and email ID.
- They can then report vehicle breakdown incidents through a user-friendly interface, eliminating the need for manual reporting processes.
- Administrators can efficiently manage reported issues by sorting and filtering them based on various parameters such as date and customer name. Sorting and filtering options facilitate quick identification and resolution of vehicle breakdowns.
- The GPS location of customer will generate after the user submits their vehicle breakdown details and admin can view and track the customer location.
- The system ensures prompt assistance to customers by streamlining the process of resolving vehicle breakdowns. Efficient problem management enables

administrators to address breakdown incidents effectively, ensuring customer satisfaction.

- Both customer and admin interfaces are designed to be user-friendly, ensuring ease of use and navigation. Intuitive interface design enhances the overall user experience.

3.2.1 ADVANTAGES:

- The system streamlines the process of reporting vehicle breakdowns, allowing customers to submit their issues quickly and efficiently. This ensures prompt assistance and reduces downtime for vehicle owners.
- By providing a centralized platform for state breakdowns, the system enhances communication between customers and administrators. This facilitates faster response times and enables administrators to coordinate assistance more effectively.
- Customers can register their details securely, ensuring the privacy and confidentiality of their personal information. This instills trust in the system and encourages more users to utilize the breakdown assistance service.
- The user-friendly interface allows customers to submit detailed information about their vehicle breakdowns, including the nature of the problem and relevant vehicle details. This ensures that administrators receive accurate and actionable information, enabling them to provide appropriate management.
- Administrators can efficiently manage reported issues by sorting and filtering them based on various parameters such as date and customer name. This enables administrators to prioritize and address breakdown incidents effectively, minimizing disruptions for vehicle owners.

CHAPTER 4

SYSTEM FUNCTION

4.1 ARCHITECTURE DESIGN

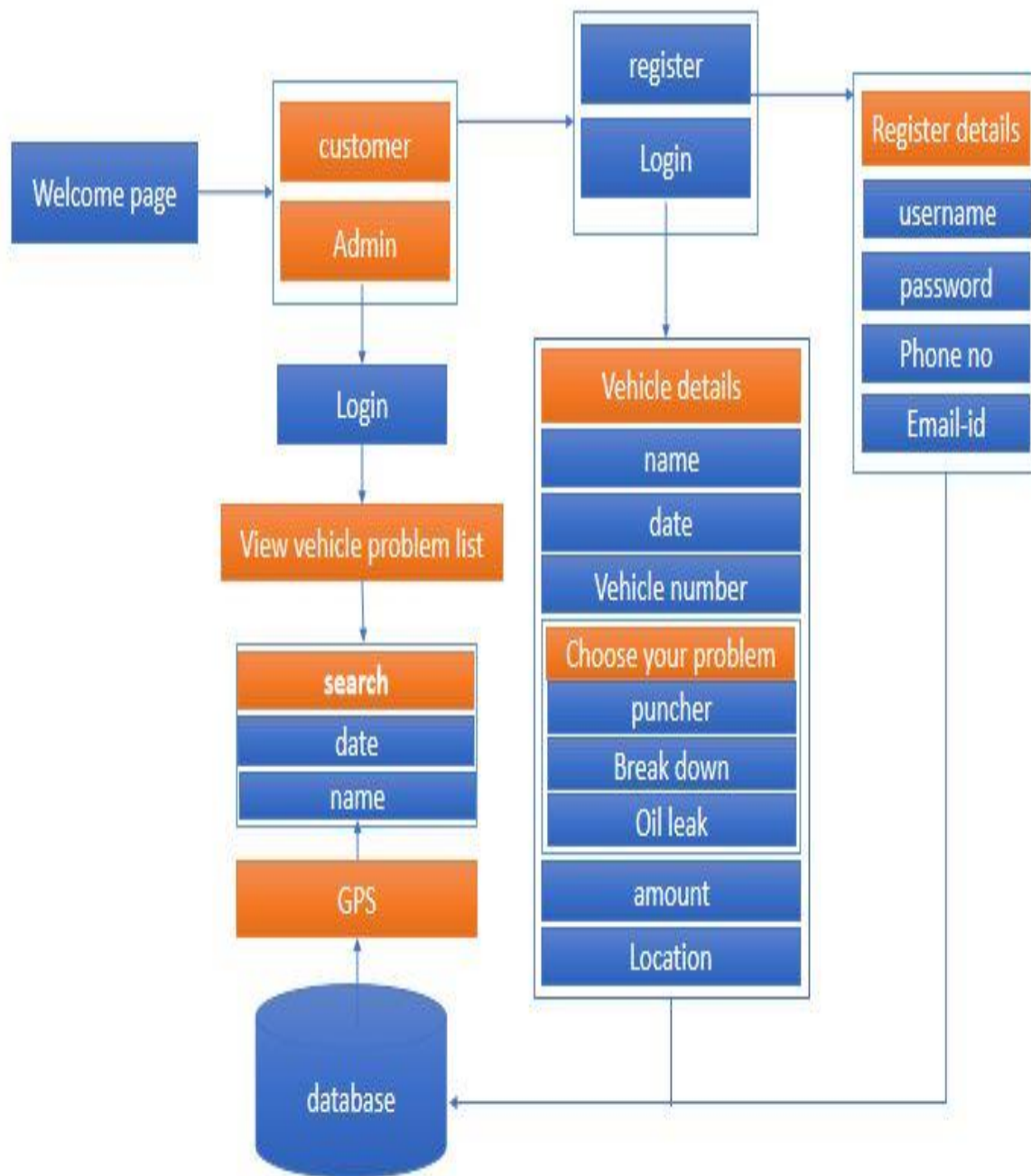


Figure 1 : Proposed for architecture design

4.2 SYSTEM MODULES

They are five modules in this system.

- Welcome and user selection module
- Customer login and Vehicle problem details
- Admin and vehicle details
- Database module

1. . Welcome and User Selection Module:

This module provides a welcoming interface to users accessing the system. It allows users to select their role, either as a customer or an admin. Provide options for users to select their role (customer or admin). Redirect users to the appropriate login/register interface based on their selection.

2. Customer Login and Vehicle Problem Details Module:

This module facilitates customer authentication and submission of vehicle problem details. Login interface for registered customers. Registration interface for new customers. Secure authentication mechanism (username and password). Form for customers to submit vehicle details including name, date, vehicle number, and type of problem (e.g., oil leakage, breakdown, puncture, engine failure). Validation of input data to ensure accuracy and completeness.

3. Admin and Vehicle Details Module:

This module provides functionalities for admin to view and manage vehicle problem details. Login interface for admin with appropriate authentication. Dashboard displaying a list of problematic vehicles reported by customers. Search options to locate specific vehicle issues based on date, customer name, or vehicle number.

Detailed view of each reported issue including customer information and problem details.

4. Database Module:

This module handles data storage and retrieval operations. Design and implementation of a relational database schema to store user details, vehicle information, and problem reports. Functions to insert and retrieve data from the database. Data validation to ensure integrity and security of stored information. Integration with other modules to fetch required data during user interactions. These modules collectively form the backbone of the vehicle breakdown assistance system, providing essential functionalities for both customers and administrators while ensuring data integrity and system security.

CHAPTER 5

SYSTEM SPECIFICATION

5.1 HARDWARE SPECIFICATION

- Processor : INTEL i5 (7th generation)
- RAM : 8 GB RAM
- Hard disk : 1TB
- Monitor : 20' color monitor

5.2 SOFTWARE SPECIFICATION

- Back end : python
- Front end : html, Css, Javascript
- Framework : Flask
- Software used : Visual studio code
- Platform : Windows 8

5.3 ENTER CUSTOMER REGISTER DETAILS

1. **Username Field:** The registration interface will feature a text input field where customers can enter their desired username. This field will have validation checks to ensure uniqueness and appropriateness of the chosen username. Instructions will be provided to guide users on acceptable characters and length limitations to ensure a smooth registration process.
2. **Password Field:** A secure password is vital for protecting customer accounts. The system will incorporate a password field where customers can input their chosen passwords. To enhance security, guidelines for creating strong passwords will be displayed, including requirements for minimum length, special characters, and a combination of uppercase and lowercase letters.

- 3. Phone Number Field:** Customers will be required to provide their phone numbers during registration. This field will include validation checks to ensure the accuracy of the entered phone number format. Additionally, tooltips or hints may be provided to guide users on the preferred format, depending on the country code requirements.
- 4. Email ID Field:** An email address is crucial for communication and account verification purposes. The registration form will include a dedicated field where customers can input their email addresses. Real-time validation checks will verify the format of the email address to ensure its correctness. Clear instructions will be provided to guide users on entering a valid email address.
- 5. Submit Button:** Once customers have filled in all the required details, a prominent "Submit" button will be provided to finalize the registration process. Clicking this button will trigger validation checks for all the input fields to ensure completeness and accuracy. In case of any errors or missing information, appropriate error messages will be displayed to guide users in rectifying the issues.

The input design for customer registration aims to provide a user-friendly experience while ensuring the accuracy and security of the submitted information. Clear instructions, validation checks, and error handling mechanisms will be implemented to streamline the registration process and enhance user satisfaction.

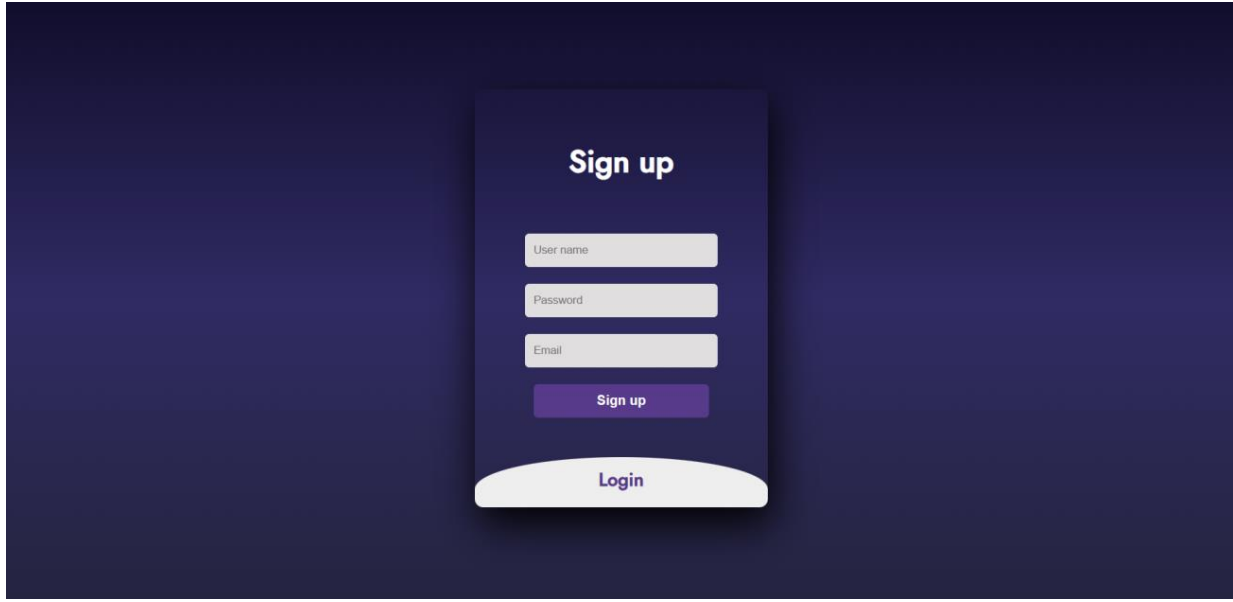


Figure 2 : Enter Customer Register Details

5.4 ENTER VEHICLE DETAILS

- 1. Name of the Customer:** Customers are required to enter their name, allowing administrators to associate the reported issue with the respective customer. This personalization ensures that administrators can address customer concerns effectively and provide personalized assistance as needed.
- 2. Vehicle Number:** This field prompts the customer to input the vehicle's registration or license plate number. The vehicle number serves as a unique identifier for the vehicle experiencing the issue. It enables administrators to quickly locate and verify details about the specific vehicle, ensuring that the correct vehicle is attended to promptly. For instance, a vehicle number like "ABC1234" provides administrators with a specific reference point to identify the vehicle in question.
- 3. Date:** Customers are prompted to select or input the date when the breakdown or issue occurred. This information is crucial for administrators

to maintain a chronological record of reported incidents, enabling them to prioritize and manage breakdowns based on their occurrence. Additionally, having the date of the incident helps administrators identify any patterns or trends in breakdown occurrences, which may inform preventive maintenance measures.

- 4. Choose Problem:** This dropdown or selection menu allows customers to choose from a predefined list of common vehicle problems, such as oil leakage, puncture, breakdown, or other issues. By categorizing the problems, customers can quickly communicate the nature of the issue to administrators, facilitating a faster response and resolution process. For instance, selecting "oil leakage" indicates to administrators that there may be a potential oil-related issue with the vehicle that requires attention. The input design for entering vehicle details within the breakdown management system aims to provide a seamless and user-friendly experience for customers. Upon accessing the platform, users are greeted with a clear and intuitive interface prompting them to input essential information regarding their vehicle's condition. Customers are guided to specify the nature of the issue encountered, whether it's a puncture, breakdown, oil leak, or any other relevant problem. Each option is presented with descriptive labels to assist users in accurately identifying and selecting the appropriate category for their issue. Once the user selects the type of problem, they are prompted to provide additional details such as the vehicle's name, date of the incident, and the vehicle number. This step ensures that administrators receive comprehensive information to effectively address the reported breakdown. The input form is designed with validation checks to ensure the accuracy and completeness of the entered details, reducing the likelihood of errors and facilitating efficient processing by administrators. The input design incorporates features such as

dropdown menus, auto-fill suggestions, and input masks to streamline data entry and enhance user convenience. Customers can easily navigate through the input fields, entering relevant information swiftly and accurately. The design also includes tooltips and help prompts to assist users in understanding the required information and completing the form effectively. Overall, the input design for entering vehicle details prioritizes simplicity, clarity, and ease of use, empowering customers to report breakdown incidents efficiently while ensuring the accuracy and completeness of the information provided to administrators for prompt resolution.

- 5. Amount:** This optional field allows customers to specify any financial transactions related to the breakdown, such as service fees or expenses incurred for repairs or assistance. Providing this information helps administrators track financial records and ensure appropriate billing or reimbursement processes.
- 6. Location:** This field allows customers to specify the location where the breakdown or issue occurred. Providing the location helps administrators determine the proximity of service providers and dispatch assistance accordingly. For example, specifying "Main Street, City Name" or providing GPS coordinates helps administrators pinpoint the exact location of the breakdown, facilitating a swift response.



Search With Reg No

RegNo

Regd Owner

Address

Makers Class

Vehicle Class

Fuel Used

Engine CC

Figure 3 : Enter Vehicle details

5.5 VIEW VEHICLE LIST

The vehicle list feature on the admin side condenses essential information about reported breakdown incidents, facilitating efficient management and resolution. Each entry includes the vehicle's registration or license plate number, enabling quick identification and access to specific vehicle details. Additionally, the name of the customer associated with each incident allows for personalized communication and tailored assistance throughout the resolution process. By specifying the location of each breakdown incident, administrators can assess proximity and allocate resources effectively, whether it involves dispatching roadside assistance or coordinating repairs. The date of each reported incident aids in prioritizing tasks, ensuring timely resolution. Moreover, the inclusion of associated costs or expenses in the breakdown resolution process allows administrators to track expenses and manage budgets effectively. Furthermore, administrators can categorize reported problems based on predefined options such

as oil leakage, engine failure, or puncture, streamlining issue identification and resource allocation.

vehicle Details

Enter your search here

Search

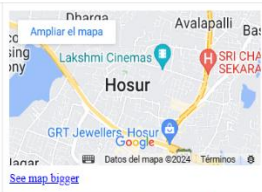
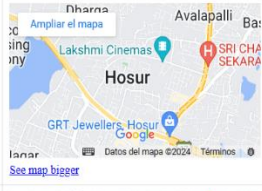
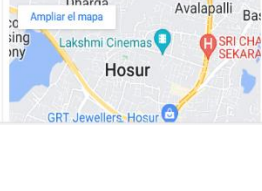
| Name | Date | Vehicle Number | Choose your problem: | Amount | Location | Live Location |
|------------|----------|----------------|----------------------|--------|----------|--|
| krishore s | 12-10-20 | 1234 | oilleak | 250 | hosur |  |
| krishore s | 12-12-21 | 1234 | breakdown | 200 | hosur |  |
| krishore s | 12-12-19 | 1234 | enginefailure | 500 | hosur |  |

Figure 4 : View vehicle list

5.6 SEARCH VEHICLE BY CUSTOMER NAME

The vehicle details search module allows administrators to quickly locate specific vehicle information by customer name. This module enhances efficiency in managing breakdown incidents by providing a streamlined method to access relevant details. Administrators can input the customer's name into the search field, triggering the system to retrieve all vehicle entries associated with that customer. The search results display comprehensive details such as the vehicle's registration number, date of the incident, location, associated costs, and categorized problem. This functionality enables administrators to efficiently track and manage breakdown incidents for individual customers, ensuring personalized

assistance and timely resolution. Additionally, the search module incorporates filters to refine results based on specific criteria, further enhancing the usability and effectiveness of the breakdown management system.

vehicle Details

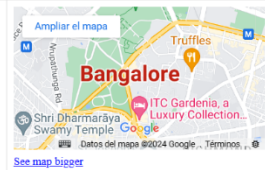
| Name | Date | Vehicle Number | Choose your problem: | Amount | Location | Live Location |
|--------|----------|----------------|----------------------|--------|-----------|---|
| sachin | 12-12-01 | TN 24 k 6492 | oilleak | 250 | bangalore |  |

Figure 5 : Search Vehicle

5.6.1 SEARCH VEHICLE BY DATE

The search module for vehicle details using date and listing all vehicles is designed to provide administrators with quick access to specific incidents and a comprehensive overview of all reported breakdowns. Here's how it functions:

- 1. Search by Date:** Administrators can input a specific date or select a date range using a calendar/date picker. This functionality allows them to narrow down the search results to incidents reported on the specified date or within the selected date range.

- 2. List All Vehicles:** Alternatively, administrators can choose the option to list all vehicles without specifying a particular date. This feature displays a complete list of all reported breakdown incidents, irrespective of the date they occurred.

By combining these two functionalities, administrators can efficiently search for vehicle details based on specific dates or retrieve a comprehensive list of all reported incidents. This module streamlines the process of accessing and managing vehicle breakdown information, enabling administrators to prioritize tasks, allocate resources effectively, and provide timely assistance to customers.

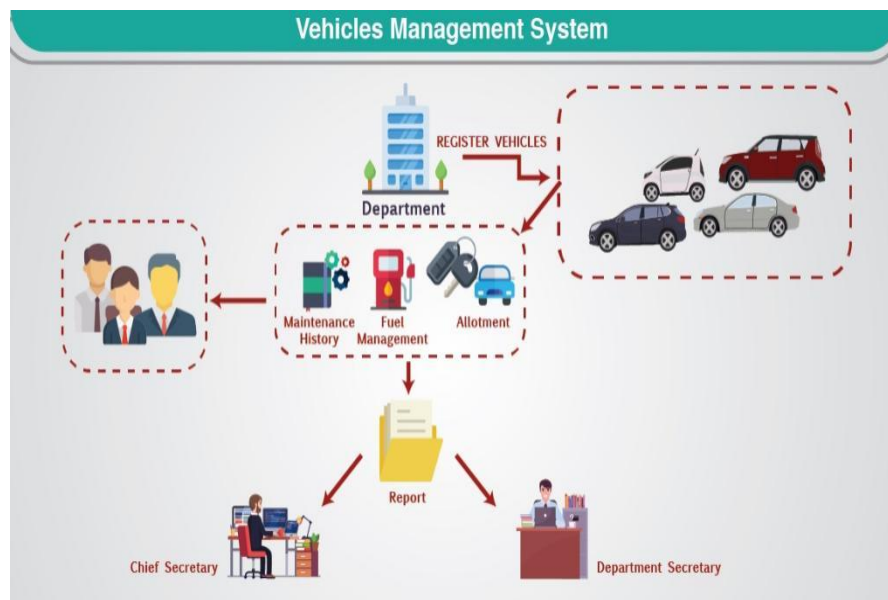


Figure 6 : Vehicle Breakdown Management

5.7 VIEW VEHICLE AVAILABLE:

In the context of the car rental system described in the abstract, "available vehicles" represent the proposed vehicle breakdown management system offers a user-friendly interface for both customers and administrators to efficiently locate

and address reported issues. Customers can input specific details such as the date and their name to initiate a search for their reported vehicle breakdowns. This feature ensures that customers can easily track the progress of their reported incidents, enhancing transparency and accountability in the breakdown resolution process. On the administrator side, the vehicle search module provides authorized personnel with a comprehensive toolset to manage reported breakdowns effectively. Administrators can access a list of reported issues and utilize the search functionality to filter results based on parameters such as date and customer name. This capability enables administrators to swiftly identify and prioritize breakdown incidents, allowing for prompt assistance and resolution. By streamlining the search process, the system enhances the overall efficiency and effectiveness of breakdown management operations. The vehicle search module fosters seamless communication between customers and administrators by facilitating quick access to relevant information. Administrators can easily retrieve details of reported breakdowns associated with specific customers or dates, enabling them to respond promptly and address customer concerns in a timely manner. This functionality contributes to enhanced customer satisfaction and trust in the breakdown management service, ultimately leading to a more positive user experience for both parties involved.

5.8 GPS TRACKER

Real-Time Location Tracking: The incorporation of GPS tracking technology enables real-time monitoring of vehicle locations when breakdown incidents occur. This ensures that administrators have immediate visibility into the precise location of stranded vehicles, allowing for faster response times and more efficient resource allocation.

Enhanced Dispatching: With GPS tracking, administrators can accurately pinpoint the location of breakdown incidents on digital maps, facilitating the dispatching of roadside assistance vehicles to the exact location of the stranded vehicle. This reduces response times and ensures that assistance reaches the affected vehicle promptly.

Route Optimization: GPS tracking enables administrators to identify the most efficient routes for dispatching assistance vehicles based on real-time traffic conditions and the proximity of service providers to the breakdown location. This optimization helps minimize response times and ensures timely assistance to stranded motorists.

Improved Customer Communication: Integrating GPS tracking into the breakdown management system allows administrators to provide customers with real-time updates on the status of their assistance request, including the estimated time of arrival of the assistance vehicle. This enhances customer satisfaction and reduces uncertainty during breakdown situations.

Data Analytics: GPS tracking generates valuable data on vehicle movement patterns, breakdown hotspots, and response times, which can be analyzed to identify trends, optimize service routes, and improve overall service efficiency. This data-driven approach enables continuous improvement and optimization of the breakdown assistance process.

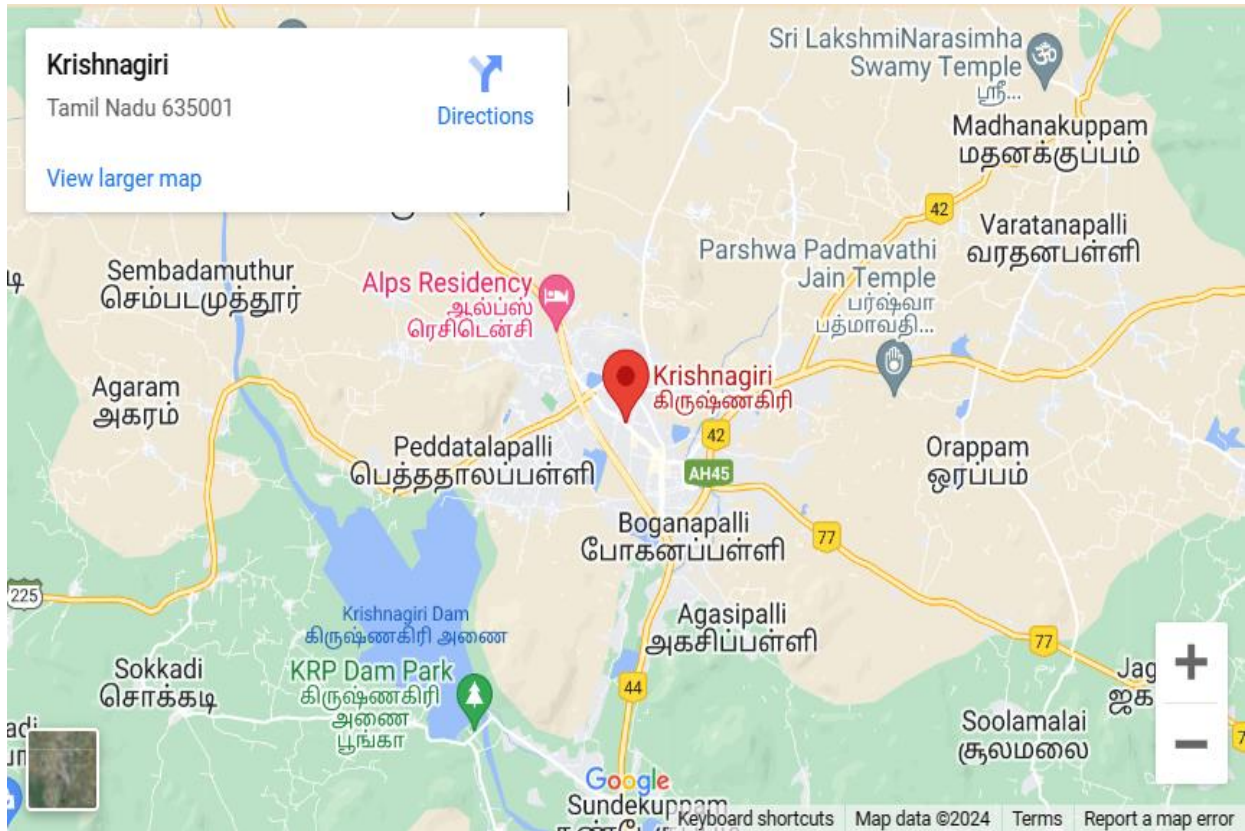


Figure 7 : GPS Tracking

5.9 DATABASE DESIGN

SQLite is a lightweight, serverless, self-contained, and open-source relational database management system (RDBMS) that is widely used in various applications due to its simplicity, reliability, and efficiency. It is designed to be embedded within applications, requiring minimal setup and administration, making it an ideal choice for projects where a compact and efficient database solution is required. In the context of the vehicle breakdown management system project, SQLite offers a convenient and scalable database solution to store and manage customer and vehicle-related data efficiently.

Database Design Considerations: The database design for the vehicle breakdown management system involves careful consideration of the entities, attributes, relationships, and constraints necessary to capture and organize the relevant information effectively. Key entities in the database schema include customers, vehicles, breakdown incidents, and administrative users. Each entity has associated attributes that store specific information, such as customer details, vehicle information, breakdown reports, and administrative credentials.

- 1. Entity-Relationship Model:** The entity-relationship (ER) model provides a conceptual framework for designing the database schema by defining entities, their attributes, and the relationships between them. In the context of the vehicle breakdown management system, the ER model includes entities such as Customer, Vehicle, Breakdown, and Admin, with relationships representing associations between these entities, such as customer-vehicle association, breakdown report ownership, and administrative privileges.
- 2. Normalization:** Normalization is a critical aspect of database design aimed at reducing data redundancy and improving data integrity by organizing data into well-structured tables. By applying normalization techniques such as First Normal Form (1NF), Second Normal Form (2NF), and Third Normal Form (3NF), the database schema can be optimized to eliminate data anomalies and ensure efficient data storage and retrieval.
- 3. Database Schema Implementation:** The database schema implementation involves translating the conceptual ER model into concrete table structures, specifying data types, constraints, and relationships using SQL (Structured Query Language) statements. Tables are created for each entity, with columns representing attributes, and primary and foreign keys used to establish

relationships between tables. Additionally, indexes may be defined to optimize query performance for frequently accessed data.

- 4. Data Integrity and Security:** Ensuring data integrity and security is paramount in any database system. Measures such as defining constraints, enforcing referential integrity through foreign key constraints, and implementing access controls through user authentication and authorization mechanisms help maintain data consistency and protect sensitive information. With SQLite, features like data encryption and access control mechanisms can be implemented to enhance the security of the vehicle breakdown management system's database.

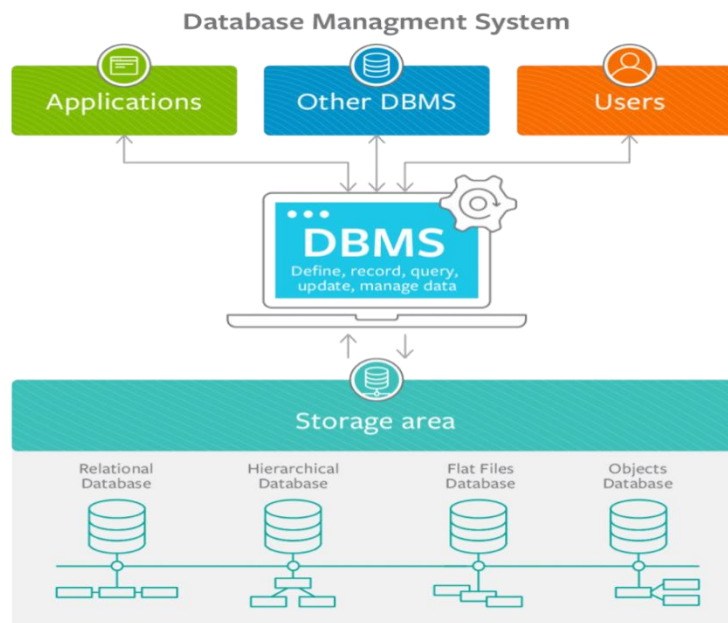


Figure 8 : Data Base Management System

Databases are used for storing, maintaining and accessing any sort of data. They collect information on people, places or things. That information is gathered in one place so that it can be observed and analysed. Databases can be thought of as an organized collection of information.

A database is an organized collection of structured information, or data, typically stored electronically in a computer system. ... Together, the data and the DBMS, along with the applications that are associated with them, are referred to as a database system, often shortened to just database. A database system is a collection of interrelated data and a set of programs that allow users to access and modify these data. A major purpose of a database system is to provide users with an abstract view of the data. That is, the system hides certain details of how the data are stored and maintained

The functions of a DBMS include concurrency, security, backup and recovery, integrity and data descriptions. Database management systems provide a number of key benefits but can be costly and time-consuming to implement. Database software is used to create, edit, and maintain database files and records, enabling easier file and record creation, data entry, data editing, updating, and reporting. The software also handles data storage, backup and reporting, multi-access control, and security.

The DBMS manages incoming data, organizes it, and provides ways for the data to be modified or extracted by users or other programs. Some DBMS examples include MySQL, PostgreSQL, Microsoft Access, SQL Server, FileMaker, Oracle, RDBMS, dBase, Clipper, and FoxPro. Oracle makes software, called database management systems (DBMS), to create and manage databases. Databases are used for storing, maintaining and accessing any sort of data. They collect information on people, places or things. That information is gathered in one place so that it can be

observed and analyzed. Databases can be thought of as an organized collection of information.

Database design is a critical aspect of the Car Rental System, providing the framework for storing, organizing, and managing data related to vehicles, customers, reservations, and administrative tasks. The database is designed to ensure efficiency, scalability, and data integrity while supporting the system's functionalities.

A database is an organized collection of structured information, or data, typically stored electronically in a computer system. Together, the data and the DBMS, along with the applications that are associated with them, are referred to as a database system, often shortened to just database. A database system is a collection of interrelated data and a set of programs that allow users to access and modify these data. A major purpose of a database system is to provide users with an abstract view of the data. That is, the system hides certain details of how the data are stored and maintained. The functions of a DBMS include concurrency, security, backup and recovery, integrity and data descriptions. Database management systems provide a number of key benefits but can be costly and time-consuming to implement.

CHAPTER 6

SOFTWARE SPECIFICATION

6.1 PYTHON PROGRAMMING LANGUAGE

Python is a powerful general-purpose programming language. It is used in web development, data science, creating software prototypes, and so on. Fortunately for beginners, Python has simple easy-to-use syntax. This makes Python an excellent language to learn to program for beginners.

- **Python is Interpreted** – Python is processed at runtime by the interpreter. You do not need to compile your program before executing it. This is similar to PERL and PHP.
- **Python is Interactive**– You can actually sit at a Python prompt and interact with the interpreter directly to write your programs.
- **Python is Object-Oriented** – Python supports Object-Oriented style or technique of programming that encapsulates code within objects.

6.1.1 FEATURES:

1. **Simple and Easy to Learn:** Python's syntax is designed to be easy to read and understand, making it ideal for beginners. Its simplicity reduces the cost of program maintenance and development.
2. **Expressive Language:** Python allows developers to express concepts in fewer lines of code compared to other programming languages. This leads to greater productivity and faster development cycles.

- 3. Interpreted and Interactive:** Python is an interpreted language, meaning that code execution occurs line by line, facilitating rapid prototyping and debugging. It also supports an interactive mode where code can be executed and tested interactively.
- 4. Multi-paradigm:** Python supports multiple programming paradigms, including procedural, object-oriented, and functional programming. This versatility allows developers to choose the most suitable approach for their projects.
- 5. Extensive Standard Library:** Python comes with a comprehensive standard library that provides modules and functions for a wide range of tasks, from file I/O to networking. This reduces the need for external dependencies and accelerates development.
- 6. Third-party Libraries:** Python has a vast ecosystem of third-party libraries and frameworks, such as NumPy for numerical computing, Pandas for data analysis, Django for web development, and TensorFlow for machine learning. These libraries extend Python's capabilities for specific domains.
- 7. Platform-independent:** Python code is typically written once and can run on any platform without modification. This cross-platform compatibility makes it suitable for developing applications that need to run on different operating systems.

- 8. Dynamic Typing:** Python uses dynamic typing, where variable types are determined at runtime. This flexibility simplifies coding and allows for more rapid development but requires careful attention to variable types.
- 9. Community Support:** Python has a large and active community of developers who contribute to its growth and provide support through forums, tutorials, and open-source projects. This vibrant community ensures that developers have access to resources and assistance when needed.
- 10. Scalability:** While Python is often praised for its simplicity and ease of use, it is also scalable and capable of handling large-scale projects. Many high-traffic websites and enterprise applications are built using Python, demonstrating its scalability and robustness.

6.2 VISUAL STUDIO CODE

VS code is a powerful integrated development environment (IDE) specifically designed for Python development, offering a comprehensive set of features to enhance productivity and streamline the coding process. With its intelligent code editor, advanced debugging capabilities, and seamless integration with version control systems and popular frameworks, VS code provides developers with a user-friendly environment for writing, testing, and debugging Python code. Its extensive array of tools, including code completion, refactoring, and unit testing support, enables developers to efficiently manage projects of any scale while maintaining code quality and consistency. Additionally, VS code offers customization options, plugins, and a vibrant community, making it a versatile and indispensable tool for Python

programmers across various domains.

VS Code offers some of the best features to its users and developers in the following aspects

- Code completion and inspection
- Advanced debugging
- Support for web programming and frameworks such as Django and Flask

6.2.1 Features of VS Code

1. Command Line: Visual Studio Code has a powerful command line interface that lets you control how you launch the editor. You can open different files, install extensions, and even change the display language at the startup.

2. Command Palette: VS Code is equally accessible from the keyboard. The most important key, which brings up the Command Palette. From here, you would have access to all of the functionality of VS Code, including keyboard shortcuts for the most common operations.

3. Git Integration: Visual Studio Code comes with Git integration that allows you to commit, pull, and push your code changes to a remote Git repository.

4. Change Language Mode: If you want to persist the new language mode for that file type, you can use the Configure File Association for command to associate the current file extension with an installed language.

5. Zen Mode: Zen Mode lets you focus on your code by hiding all UI except the editor (no Activity Bar, Status Bar, Side Bar and Panel), going to full screen and centering the editor layout. Zen mode can be toggled using the View menu,

Command Palette. Double Esc exits Zen Mode. The transition to full screen can be disabled via zen Mode.full Screen.

6. Debugging and Testing Tools: Visual Studio Code offers robust debugging tools with breakpoints, watch expressions, and a powerful debugger console, enabling developers to identify and fix issues quickly. It also supports various testing frameworks, making it seamless to write and execute tests.

7. Version Control Integration: Seamless integration with popular version control systems like Git, Mercurial, and SVN allows developers to manage their code changes efficiently directly within the IDE.

8. Built-in Terminal: VS Code comes with an integrated terminal, eliminating the need for developers to switch between the IDE and the command line for tasks like running scripts or managing virtual environments.

9. Code Refactoring: Developers can refactor their code easily with Visual Studio Code automated refactoring tools, which help improve code maintainability and readability while minimizing the risk of introducing errors.

10. Database Tools: VS Code offers database tools for interacting with databases directly from the IDE, enabling developers to query, modify, and manage database schemas without leaving their development environment.

11. Support for Web Development: With support for web frameworks like Django, Flask, and Pyramid, VS Code facilitates web development by providing features such as template editing, debugging, and integration with front end tools like JavaScript, HTML, and CSS.



Figure 9 : VS Code software

6.2.2 VISUAL STUDIO INSTALLATION

Step 1. Download the Visual Studio Code installer for Windows

Download the Visual Studio Code installer for Windows from its official website, code.visualstudio.com. We recommend not to download from other sources. You are free to download Professional or Community editions. However, the Community edition is free to use as.

Follow the steps below to download Visual Studio Code on Windows:1. Go to the code.visualstudio.com.. Under the Community section, click on the Download button for the Windows package.3. An exe file will be downloaded to your system. Note the location where the exe file is saved.

Step 1

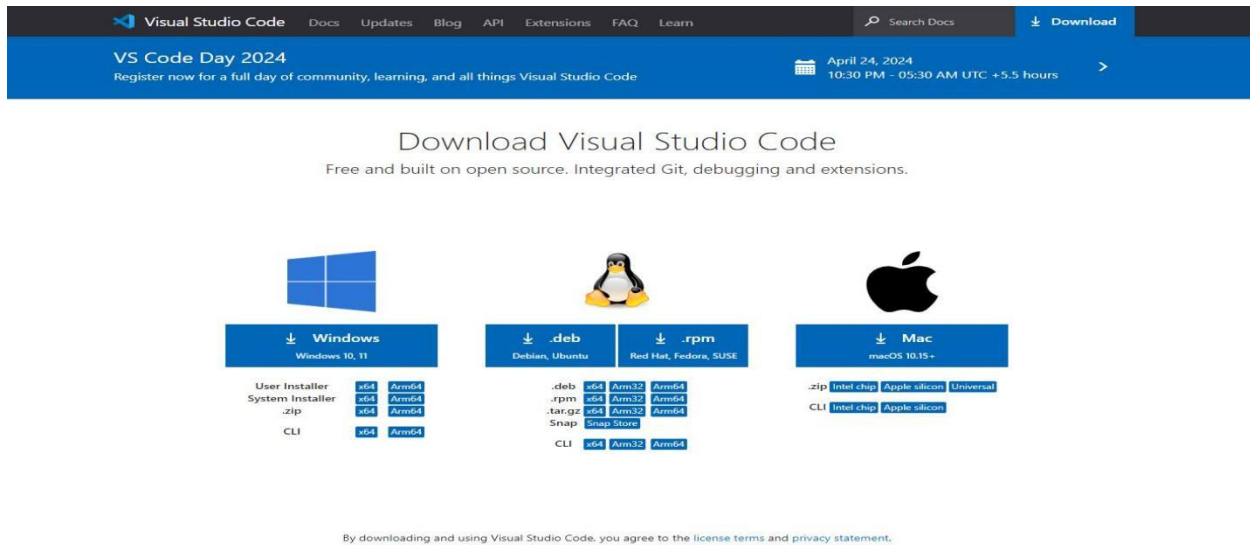


Figure 9.1 : Download the Visual Studio Code

Step 2

After clicking on windows to Install

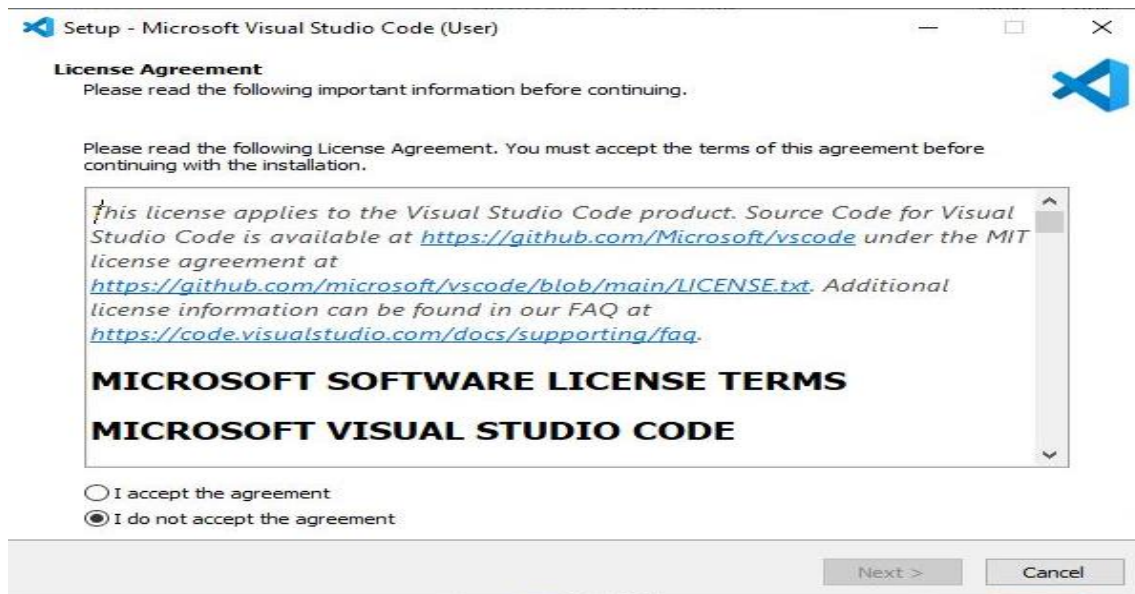


Figure 9.2 : Install Visual Studio Code on Windows

Step 3

After Click on Next , You need to choose the destination folder according to your choice.

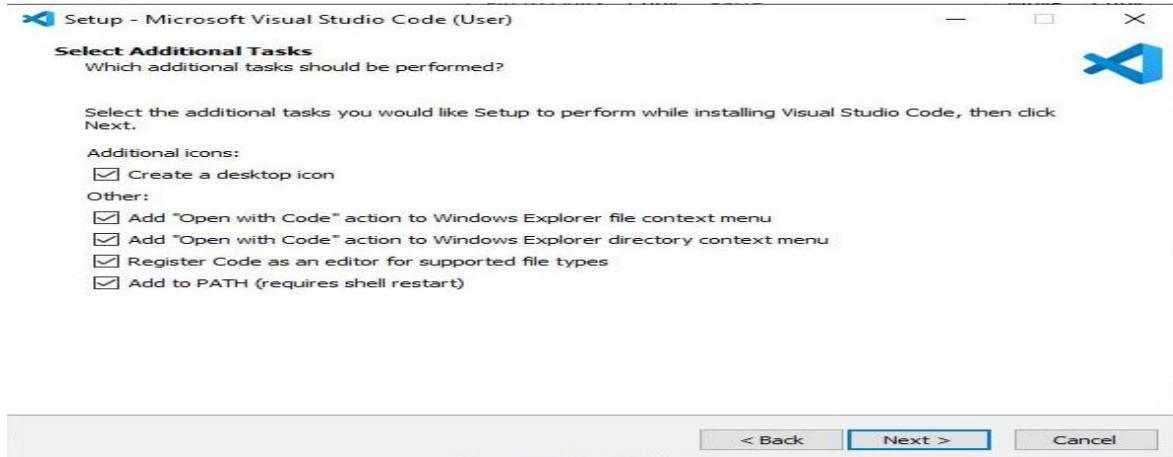


Figure 9.3

Step 4

Installation on process

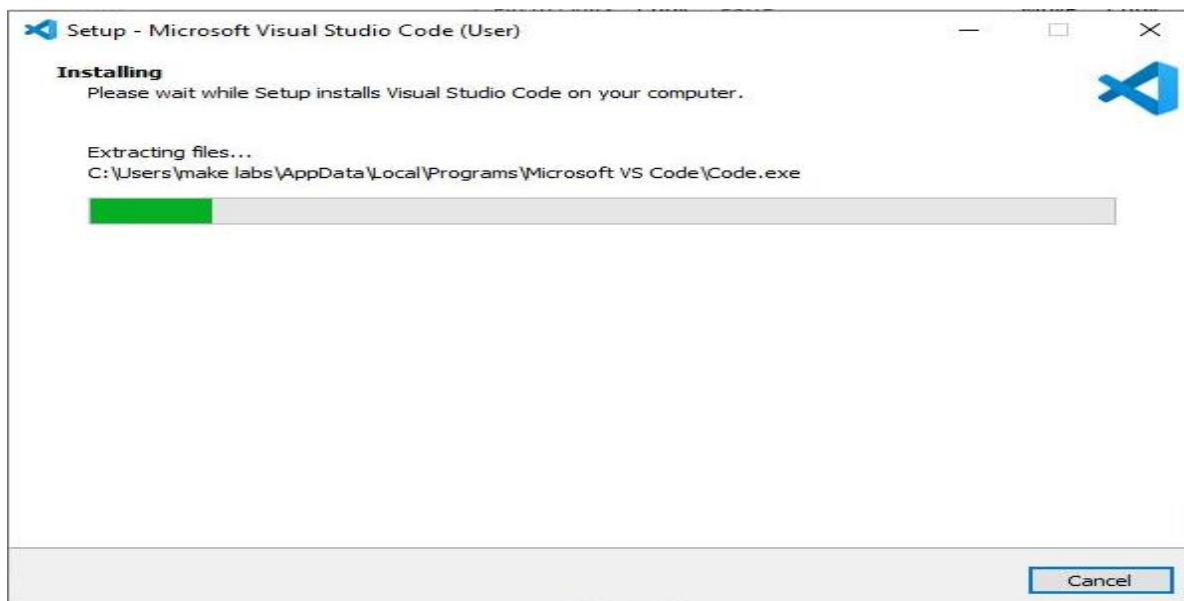


Figure 9.4

Step 5

After Installation completed , It will show that Visual Studio Code on Windows is installed successfully , then click on “I want to manually reboot later”. Click on Finish and then the process is Completed.

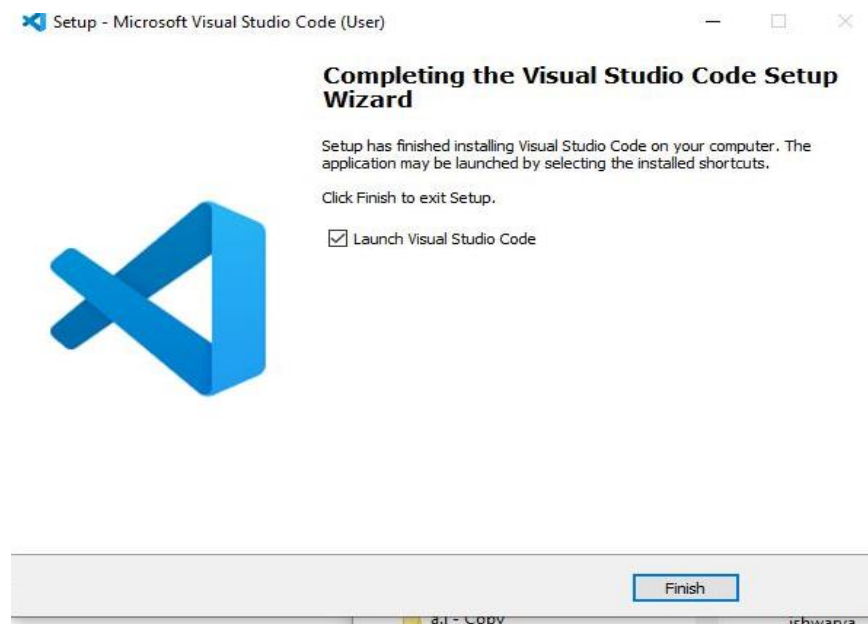


Figure 9.5

Step 8

Reboot & Launch Visual Studio Code

We recommend rebooting the machine to save the PATH variable settings. Follow the steps below to launch Visual Studio Code after installing it:

1. Open the Start Menu and click on the Visual Studio Code icon under vscode to launch it. Alternatively, double-click the desktop shortcut if you have

created one.2. When launching for the first time, a welcome screen will appear, which will guide you through some configuration settings.

Step 9

Create your First Python Project

The very first thing you should do is to create a Project. Select the type of project you want to create, Click on the New Project set up Location settings, and hit Create.

1. Click Create New Project on the welcome screen. Or select File > New Project.
2. Select the project location, name, and Python interpreter to associate.
3. vscode will automatically create a virtual environment for the project.
4. Open the new project and create a main.py Python file.
5. Add a simple print("Hello World") statement and run it.
6. If "Hello World" is printed in the output, your vscode is ready!

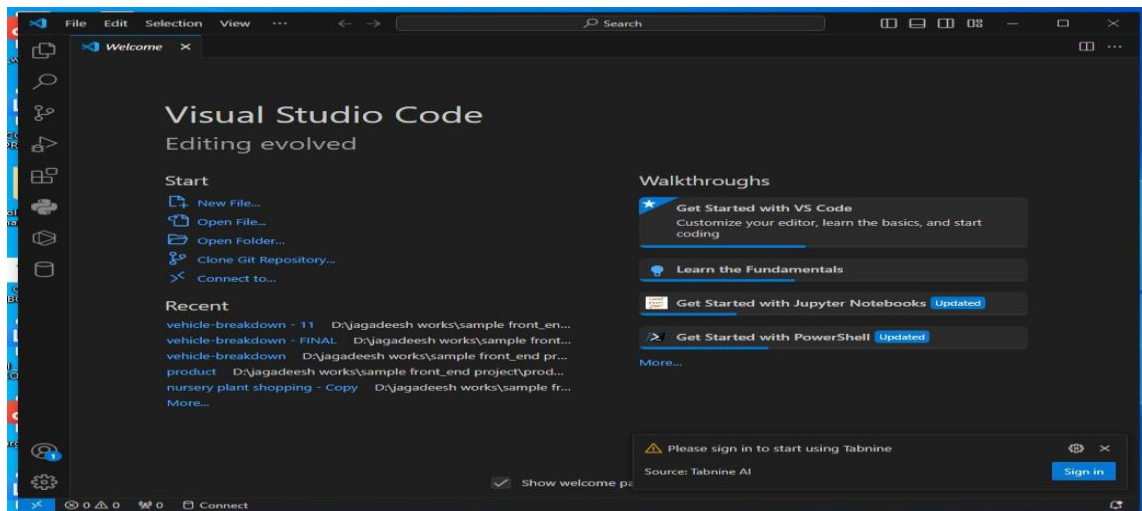


Figure 9.6 : Create an New Project

6.3 RUNING AND DEBUGGING

Running a python code comprises of two modes: running a script and debugging the script. This chapter focusses on debugging the Python script using vscode

Steps Involved

The steps for debugging the Python project are as explained below –

Step 1

Start with debugging the Python project as shown in the screenshot below –

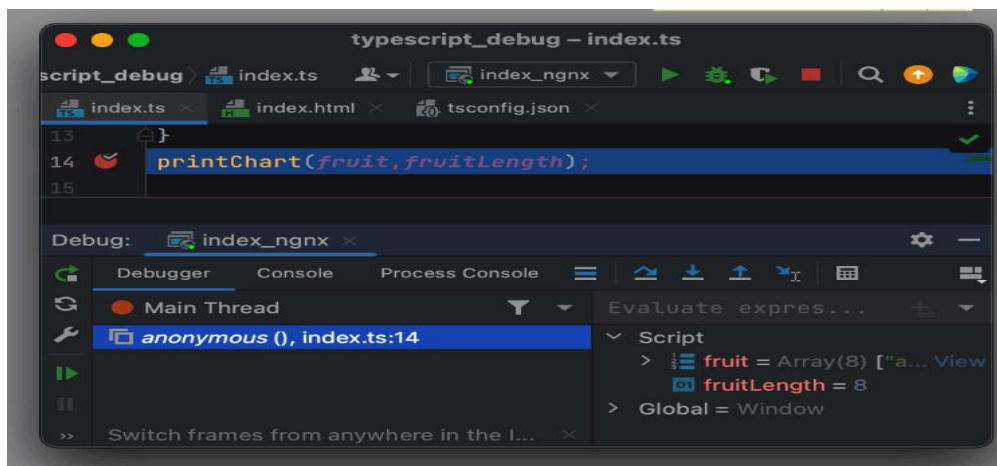


Figure 9.7

Step 2

Now, Windows firewall asks permission for debugging the Python project as the procedure involves line by line compilation.

The run button moves from one line to another to execute the output as the way we want.

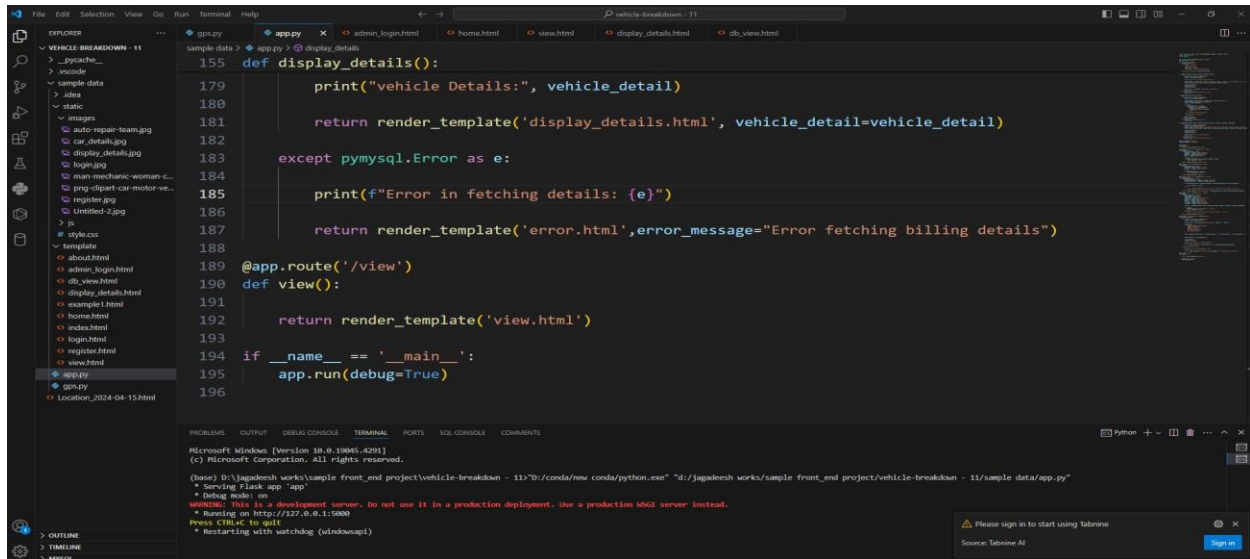


Figure 9.10 : Running window

6.4 Understanding Breakpoints

While debugging a particular script, it is intentional to create a breakpoint. Breakpoints are intentional stopping place or the place where the code is paused in order to identify the output at specific stage.

In vscode, breakpoints are visible using a separate dialog in the specified editor. It includes various attributes to evaluate the breakpoints defined and tracing log for the same with a main motive to achieve better programming practice.

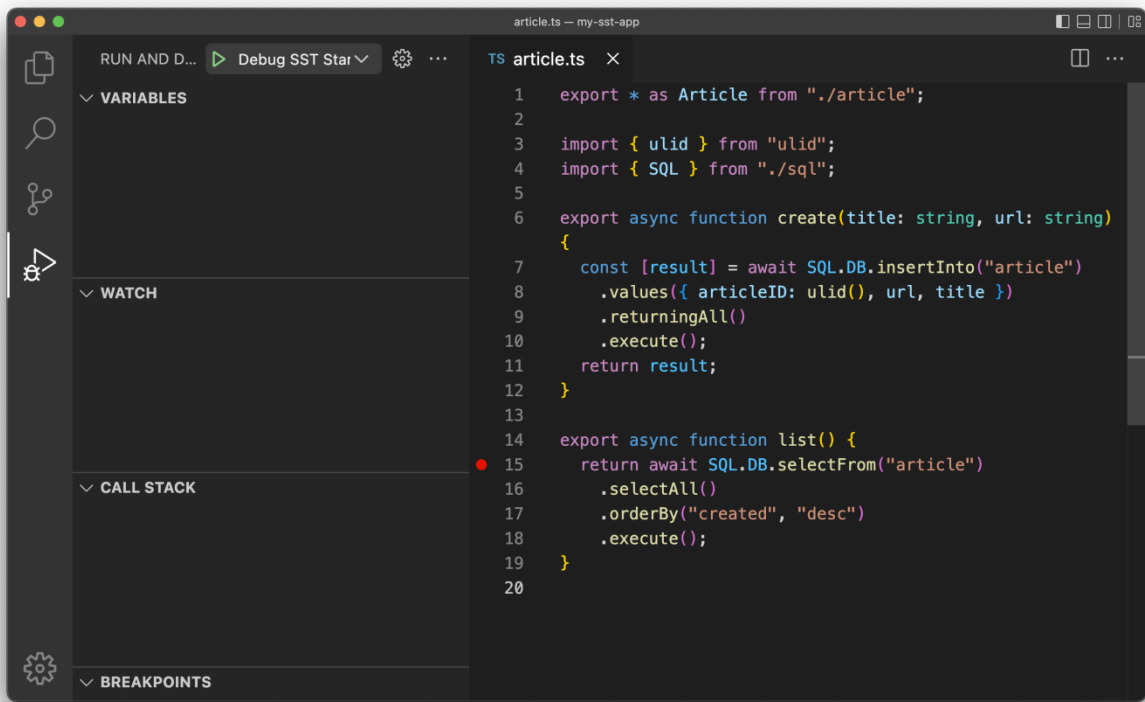


Figure 9.11 : Understanding Breakpoints

CHAPTER 7

HARDWARE REQUIERMENTS & SPECIFICATION

7.1 I5 PROCESSOR

Core is the processors I5 family which is famous for its latest revolutionary structure and integrated architecture which also provide the advantage of the parallel computing. It's also wonderful in providing the users with the excellent graphical user interfaces.

7.1.1 Basic Features of the I5

The basic feature of the I5 features is highly improved as compared to previous version of the processor by Intel. Some of the popular and leading features of I5 processors are listed below.

- I5 processors offers the perfect accuracy and high performance and response rate which in result provide the users with the high throughput rates, and also reduced time in executing the programs by the processor.
- The Intel I5 processor is fully equipped by the latest HD graphics with powerful and advanced video engine that provides smooth high quality display along with the 3d graphics capabilities. On the whole I5 processors can be considered as the high graphical and multimedia display processors for daily computing.
- Intel I5 processors also provide hyper threading technology to its users which enable the multitasking capability of both user and the system. The systems with I5 processors can perform execution and compilation of two tasks simultaneously without causing the executing delays and debuggers errors.

They are so responsive that output of the programs can be generated at the same time too. We can easily say that Intel I5 are the best choice for homes and offices. More than seven applications can run simultaneously on the system with I5 processor built the on motherboards.

- I5 processors are smarter, faster and more adaptive in all kinds of networking scheme. They can be used with any of the hard disk configurations.
- They are also famous in the markets by the name of desktop processor because of the great quality resolution they have. Integrated components on the motherboards also makes I5 processors unique in their architecture and circuit installations

7.1.2 More advantages of the I5

I5 processors have remarkable advantages that are of great use in the field of computers and technology. Some of them are listed below.

- Dual core processing has the ability to run two independent programs with one hardware.
- I5 processors have improved Pentium base, they have totally new architecture with more integrations and high-speed performance structure.
- Hyper threading technology also enables user to enjoy the high speed and better performance with more reliable outputs. It has 4 tasking threads that allows user to easily execute 3 to 4 programmers at a time.
- Smart memory and cache sequence allows user to enjoy the optimized and efficient data access both direct and sequentially. Effective shortcuts have reduced the access time of the file and system.

- HD graphical features also make these processors distinguished from the others because they are considered as best in their resolution.

7.2 RANDOM ACCESS MEMORY

RAM (*pronounced ram*) is an acronym for *random access memory*, a type of computer memory that can be accessed randomly; that is, any byte of memory can be accessed without touching the preceding bytes. RAM is found in servers, PCs, tablets, smartphones and other devices, such as printers.

7.2.1 Main Types of RAM

There are two main types of RAM:

1. DRAM (Dynamic Random Access Memory)
2. SRAM (Static Random Access Memory)

DRAM (Dynamic Random Access Memory) – The term dynamic indicates that the memory must be constantly refreshed or it will lose its contents. DRAM is typically used for the main memory in computing devices. If a PC or smartphone is advertised as having 4GB RAM or 16GB RAM, those numbers refer to the DRAM, or main memory, in the device.

More specifically, most of the DRAM used in modern systems is synchronous DRAM, or SDRAM. Manufacturers also sometimes use the acronym DDR (or DDR2, DDR3, DDR4, etc.) to describe the type of SDRAM used by a PC or server. DDR stands for double data rate, and it refers to how much data the memory can transfer in one clock cycle.

In general, the more RAM a device has, the faster it will perform.

SRAM (Static Random Access Memory) – While DRAM is typically used for main memory, today SRAM is more often used for system cache. SRAM is said to be static because it doesn't need to be refreshed, unlike dynamic RAM, which needs to be refreshed thousands of times per second. As a result, SRAM is faster than DRAM. However, both types of RAM are volatile, meaning that they lose their contents when the power is turned off.

7.2.2 The Difference between Memory, RAM and Storage

In common usage, the term RAM is synonymous with main memory. This is where a computing system stores data that it is actively using. Storage systems, such as hard drives, network storage devices or cloud storage, are where a system saves data that it will need to access later.

Computing systems can retrieve data from RAM very quickly, but when a device powers down, all the data that was in memory goes away. Many people have had the experience of losing a document they were working on after an unexpected power outage or system crash. In these cases, the data was lost because it was stored in system memory, which is volatile.

By contrast, storage is slower, but it can retain data when the device is powered down. So, for example, if a document has been saved to a hard drive prior to a power outage or system crash, the user will still be able to retrieve it when the system is back up and running.

Storage is usually less expensive than RAM on a per-gigabyte basis. As a result, most PCs and smart phones have many times more gigabytes of storage than gigabytes of RAM.

The growing use of solid-state drives has blurred the line between memory and storage, in the process greatly improving the performance of storage devices.

7.2.3 Computer memory (RAM)

Computer random access memory (RAM) is one of the most important components in determining your system's performance. RAM gives applications a place to store and access data on a short-term basis. It stores the information your computer is actively using so that it can be accessed quickly. The more programs your system is running, the more you'll need. SSDs (solid state drives) are also important components and will help your system reach its peak performance. The speed and performance of your system directly correlate to the amount of RAM you have installed. If your system has too little RAM, it can be slow and sluggish. But on the opposite end, you can install too much with little to no added benefit. There are ways to see if your computer needs more memory, and to make sure you are buying memory that is compatible with the other components in your system. Generally, components are created to the highest standard at the time of manufacture, but with the expectation that technology will continue to change. To prevent users from inserting incompatible memory, modules are physically different for each memory technology generation. These physical differences are standard across the memory industry. One of the reasons for industry-wide standardization in memory is that computer makers need to know the electrical parameters and physical shape of the memory that can be installed in their computers.

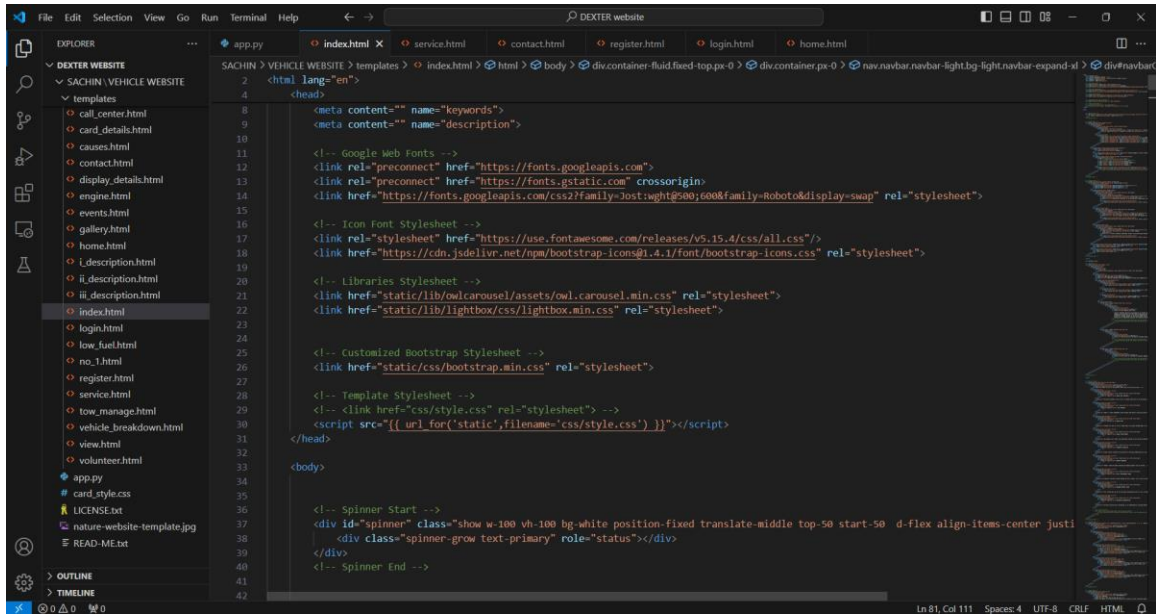
Uses RAM

RAM is used to store information that needs to be used quickly. This means that opening many programs, running various processes or accessing multiple files simultaneously is likely to use a lot of RAMS. Particularly complexed programs like games or design software will use most RAM.

CHAPTER 8

CODING

HTML

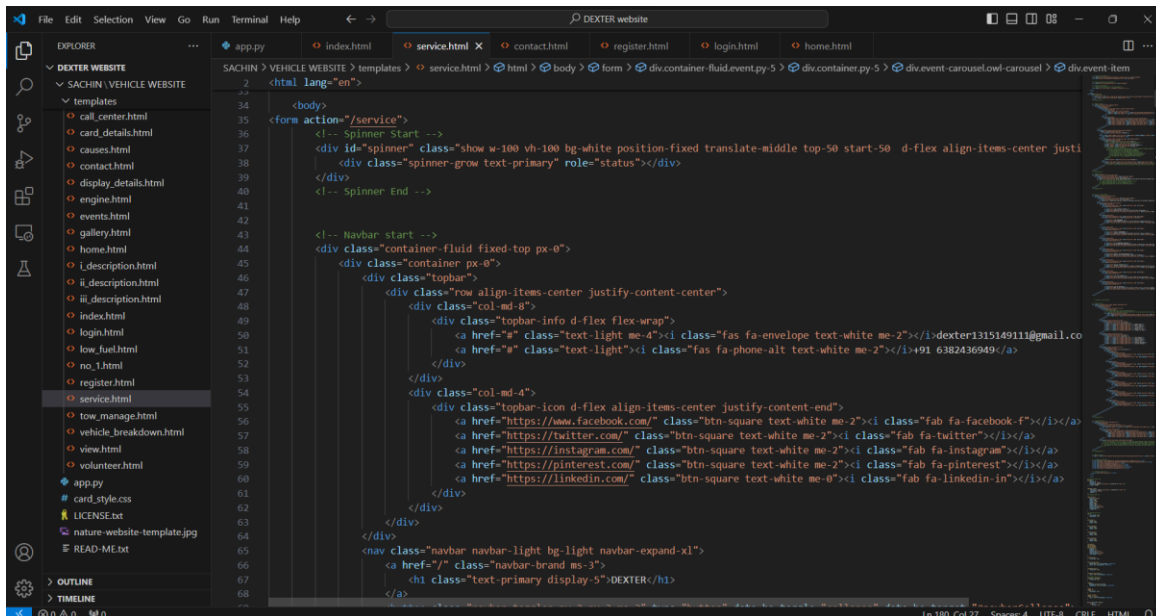


```

1  <html lang="en">
2    <head>
3      <meta content="" name="keywords">
4      <meta content="" name="description">
5
6      <!-- Google Web Fonts -->
7      <link rel="preconnect" href="https://fonts.googleapis.com">
8      <link rel="preconnect" href="https://fonts.gstatic.com" crossorigin>
9      <link href="https://fonts.googleapis.com/css2?family=Just:weight@500;600&family=Roboto&display=swap" rel="stylesheet">
10
11     <!-- Icon Font Stylesheet -->
12     <link rel="stylesheet" href="https://use.fontawesome.com/releases/v5.15.4/css/all.css"/>
13     <link href="https://cdn.jsdelivr.net/npm/bootstrap-icons@1.4.1/font/bootstrap-icons.css" rel="stylesheet">
14
15     <!-- Libraries Stylesheet -->
16     <link href="static/lib/owlcarousel/assets/owl.carousel.min.css" rel="stylesheet">
17     <link href="static/lib/lightbox/css/lightbox.min.css" rel="stylesheet">
18
19     <!-- Customized Bootstrap Stylesheet -->
20     <link href="static/css/bootstrap.min.css" rel="stylesheet">
21
22     <!-- Template Stylesheet -->
23     <!-- <link href="css/style.css" rel="stylesheet"> -->
24     <script src="{{ url_for('static', filename='css/style.css') }}"></script>
25   </head>
26   <body>
27
28     <!-- Spinner Start -->
29     <div id="spinner" class="show w-100 vh-100 bg-white position-fixed translate-middle top-50 start-50 d-flex align-items-center justify-content-center">
30       <div class="spinner-grow text-primary" role="status"></div>
31     </div>
32     <!-- Spinner End -->
33
34   </body>
35 </html>

```

Figure 10



```

1  <html lang="en">
2    <body>
3      <form action="/service">
4        <!-- Spinner Start -->
5        <div id="spinner" class="show w-100 vh-100 bg-white position-fixed translate-middle top-50 start-50 d-flex align-items-center justify-content-center">
6          <div class="spinner-grow text-primary" role="status"></div>
7        </div>
8        <!-- Spinner End -->
9
10       <!-- Navbar start -->
11       <div class="container-fluid fixed-top px-0">
12         <div class="container px-0">
13           <div class="topbar">
14             <div class="row align-items-center justify-content-center">
15               <div class="col-md-8">
16                 <div class="topbar-info d-flex flex-wrap">
17                   <a href="#" class="text-light me-4"><i class="fas fa-envelope text-white me-2"></i>dexter131514911@gmail.co</a>
18                   <a href="#" class="text-light"><i class="fas fa-phone-alt text-white me-2"></i>+91 6382436949</a>
19                 </div>
20               <div class="col-md-4">
21                 <div class="topbar-icon d-flex align-items-center justify-content-end">
22                   <a href="https://www.facebook.com/" class="btn-square text-white me-2"><i class="fab fa-facebook-f"></i></a>
23                   <a href="https://twitter.com/" class="btn-square text-white me-2"><i class="fab fa-twitter"></i></a>
24                   <a href="https://instagram.com/" class="btn-square text-white me-2"><i class="fab fa-instagram"></i></a>
25                   <a href="https://pinterest.com/" class="btn-square text-white me-2"><i class="fab fa-pinterest"></i></a>
26                   <a href="https://linkedin.com/" class="btn-square text-white me-0"><i class="fab fa-linkedin-in"></i></a>
27                 </div>
28               </div>
29             </div>
30           </div>
31         </div>
32       </div>
33       <nav class="navbar navbar-light bg-light navbar-expand-xl">
34         <a href="/" class="navbar-brand ms-3">
35           <div class="text-primary display-5">DEXTER</div>
36         </a>
37       </nav>
38     </body>
39   </html>

```

Figure 10.1

CSS

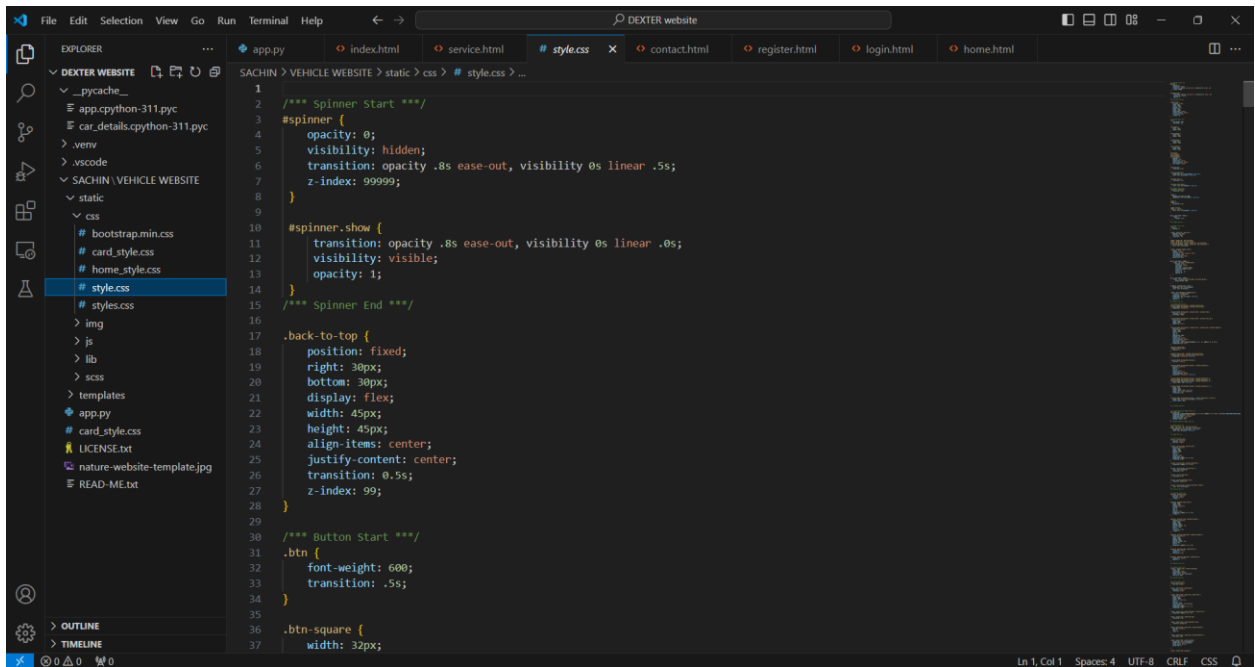


Figure 11

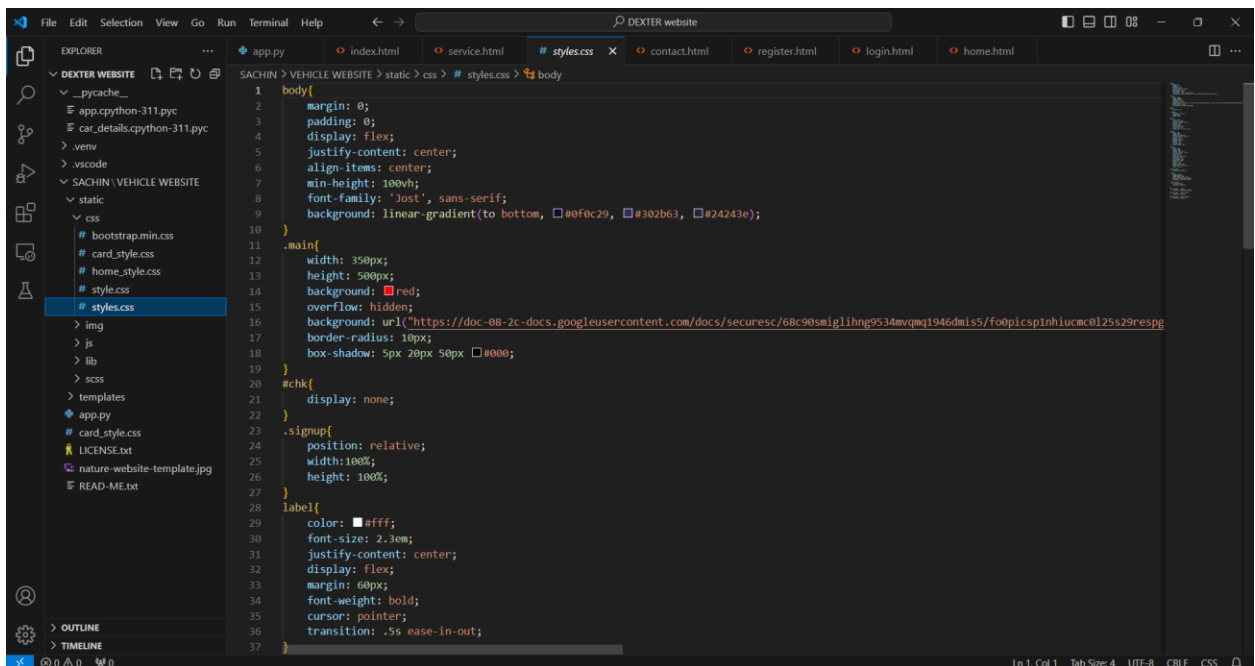
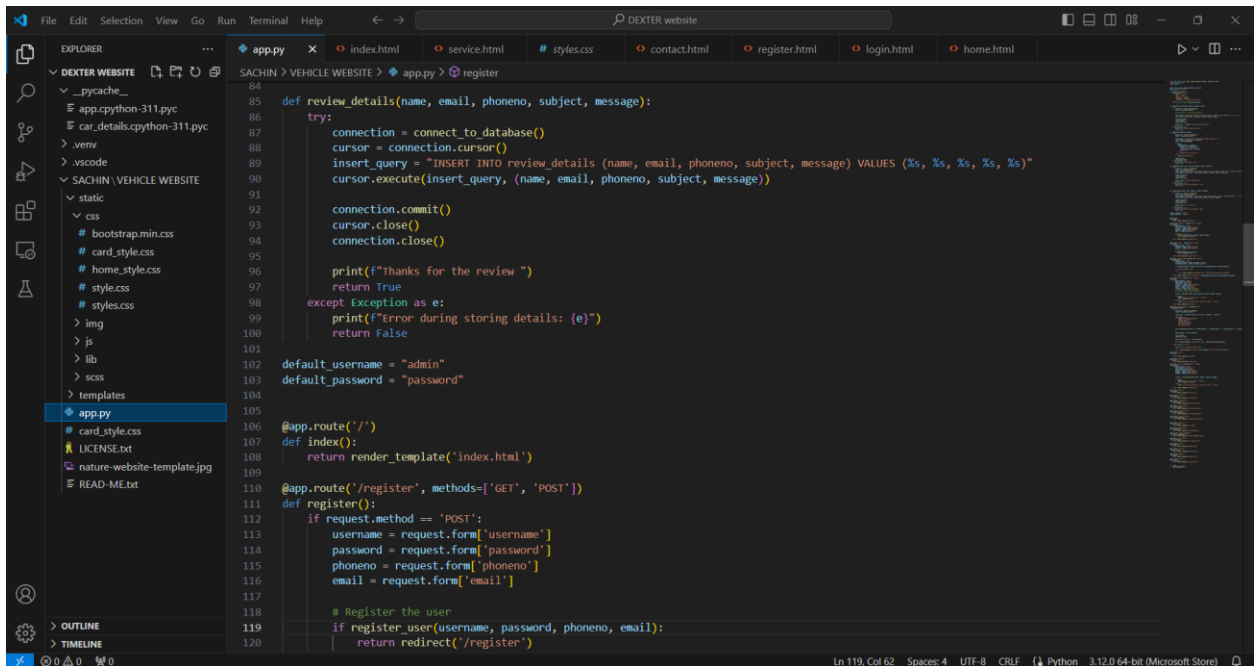


Figure 11.1

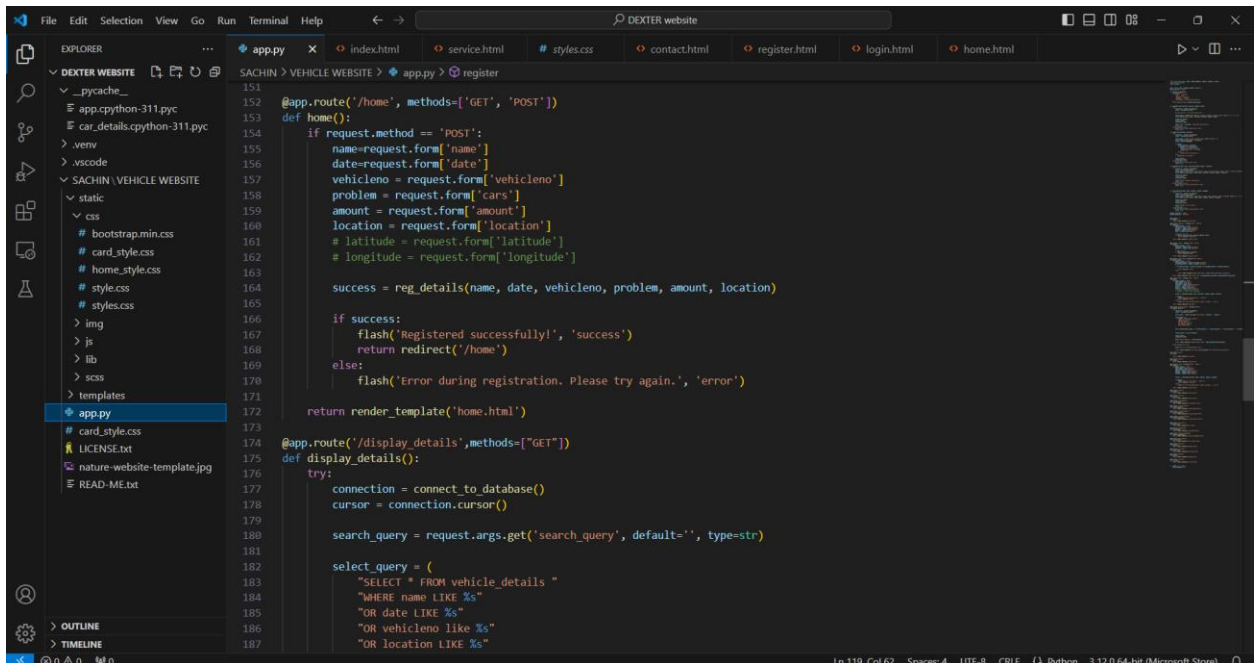
PYTHON



The screenshot shows a Visual Studio Code editor window with a project named "DEXTER website". The Explorer pane on the left shows the file structure, including a "static" folder with "css" and "img" subfolders, and a "templates" folder. The "app.py" file is selected in the Explorer. The main editor area displays the code for "register.html". The code includes a function "review_details" that inserts data into a database, a default username and password, and a route for the register page. The register page uses the "render_template" function to display the "index.html" template.

```
84
85 def review_details(name, email, phoneno, subject, message):
86     try:
87         connection = connect_to_database()
88         cursor = connection.cursor()
89         insert_query = "INSERT INTO review_details (name, email, phoneno, subject, message) VALUES (%s, %s, %s, %s, %s)"
90         cursor.execute(insert_query, (name, email, phoneno, subject, message))
91
92         connection.commit()
93         cursor.close()
94         connection.close()
95
96         print("Thanks for the review ")
97         return True
98     except Exception as e:
99         print("Error during storing details: {e}")
100         return False
101
102 default_username = "admin"
103 default_password = "password"
104
105
106 @app.route('/')
107 def index():
108     return render_template("index.html")
109
110 @app.route('/register', methods=['GET', 'POST'])
111 def register():
112     if request.method == 'POST':
113         username = request.form['username']
114         password = request.form['password']
115         phoneno = request.form['phoneno']
116         email = request.form['email']
117
118         # Register the user
119         if register_user(username, password, phoneno, email):
120             return redirect('/register')
```

Figure 12



The screenshot shows a Visual Studio Code editor window with a project named "DEXTER website". The Explorer pane on the left shows the file structure, including a "static" folder with "css" and "img" subfolders, and a "templates" folder. The "app.py" file is selected in the Explorer. The main editor area displays the code for "app.py". The code includes a route for the home page, a function "home" that handles the registration process, and a route for displaying details. The "home" function uses the "render_template" function to display the "home.html" template. The "display_details" function uses a database cursor to search for details based on a search query.

```
151
152 @app.route('/home', methods=['GET', 'POST'])
153 def home():
154     if request.method == 'POST':
155         name=request.form['name']
156         date=request.form['date']
157         vehicleno = request.form['vehicleno']
158         problem = request.form['cars']
159         amount = request.form['amount']
160         location = request.form['location']
161         # latitude = request.form['latitude']
162         # longitude = request.form['longitude']
163
164         success = reg_details(name, date, vehicleno, problem, amount, location)
165
166         if success:
167             flash('Registered successfully!', 'success')
168             return redirect('/home')
169         else:
170             flash('Error during registration. Please try again.', 'error')
171
172     return render_template('home.html')
173
174 @app.route('/display_details', methods=['GET'])
175 def display_details():
176     try:
177         connection = connect_to_database()
178         cursor = connection.cursor()
179
180         search_query = request.args.get('search_query', default='', type=str)
181
182         select_query = (
183             "SELECT * FROM vehicle_details "
184             "WHERE name LIKE %s"
185             "OR date LIKE %s"
186             "OR vehicleno like %s"
187             "OR location LIKE %s"
```

Figure 12.1

DATABASE

```
1 from flask import Flask, flash, render_template, request, redirect, session
2 import mysql.connector
3 import pymysql
4
5
6 app = Flask(__name__, template_folder='templates')
7 app.secret_key = 'jw123'
8
9 def connect_to_database():
10     db_config = {
11         'host': 'localhost',
12         'user': 'root',
13         'password': 'kishore',
14         'database': 'vehicle',
15         'auth_plugin': 'mysql_native_password'
16     }
17     return mysql.connector.connect(**db_config)
18
19
20 def register_user(username, password, phoneno, email):
21     try:
22         connection = connect_to_database()
23         cursor = connection.cursor()
24
25         cursor.execute("""
26             INSERT INTO users (username, password, phoneno, email)
27             VALUES (%s, %s, %s, %s)
28         """, (username, password, phoneno, email))
29
30         connection.commit()
31         cursor.close()
32         connection.close()
33
34         flash('User registered successfully')
35         return redirect('/login')
36     except Exception as e:
37         flash('Registration failed: %s' % e)
38         return redirect('/register')
```

127.0.0.1 - - [05/May/2024 17:11:03] "GET /static/img/admin.jpg HTTP/1.1" 304 -
127.0.0.1 - - [05/May/2024 17:11:27] "POST /admin_login HTTP/1.1" 302 -
127.0.0.1 - - [05/May/2024 17:11:27] "GET /view HTTP/1.1" 200 -
127.0.0.1 - - [05/May/2024 17:11:27] "GET /static/js/script.js HTTP/1.1" 404 -
127.0.0.1 - - [05/May/2024 17:11:27] "GET /static/images/display_details.jpg HTTP/1.1" 404 -
vehicle details: [('krishore s', datetime.date(2020, 10, 12), '1234', 'oil leak', '250', 'hosur'), ('krishore s', datetime.date(2021, 12, 12), '1234', 'br eakdown', '200', 'hosur'), ('krishore s', datetime.date(2019, 12, 12), '1234', 'engine failure', '500', 'hosur'), ('sachin', datetime.date(2001, 12, 12), 'TN 24 k 6492', 'oil leak', '250', 'bangalore')]
127.0.0.1 - - [05/May/2024 17:11:30] "GET /display_details HTTP/1.1" 200 -
127.0.0.1 - - [05/May/2024 17:11:30] "GET /static/js/script.js HTTP/1.1" 404 -
127.0.0.1 - - [05/May/2024 17:11:30] "GET /static/images/display_details.jpg HTTP/1.1" 404 -

Figure 13

```
103 default_password = "password"
104
105
106 @app.route('/')
107 def index():
108     return render_template("index.html")
109
110 @app.route('/register', methods=['GET', 'POST'])
111 def register():
112     if request.method == 'POST':
113         username = request.form['username']
114         password = request.form['password']
115         phoneno = request.form['phoneno']
116         email = request.form['email']
117
118         # Register the user
119         if register_user(username, password, phoneno, email):
120             return redirect('/register')
121
122     return render_template("register.html")
123
124
125 @app.route('/login', methods=['POST', 'GET'])
```

127.0.0.1 - - [05/May/2024 16:47:07] "GET /static/img/gallery-footer-6.jpg HTTP/1.1" 304 -
127.0.0.1 - - [05/May/2024 16:47:18] "GET /register HTTP/1.1" 200 -
127.0.0.1 - - [05/May/2024 16:47:18] "GET /slide%20navbars%20style.css HTTP/1.1" 404 -
127.0.0.1 - - [05/May/2024 16:47:18] "GET /css/styles.css HTTP/1.1" 404 -
User "siva" registered successfully!
127.0.0.1 - - [05/May/2024 16:48:56] "POST /register HTTP/1.1" 302 -
127.0.0.1 - - [05/May/2024 16:48:56] "GET /register HTTP/1.1" 200 -
127.0.0.1 - - [05/May/2024 16:48:56] "GET /css/styles.css HTTP/1.1" 404 -
127.0.0.1 - - [05/May/2024 16:48:56] "GET /slide%20navbars%20style.css HTTP/1.1" 404 -
Login successful!
127.0.0.1 - - [05/May/2024 16:49:19] "POST /login HTTP/1.1" 302 -
127.0.0.1 - - [05/May/2024 16:49:19] "GET /home HTTP/1.1" 200 -

Figure 13.1

CHAPTER 9

OUTPUT

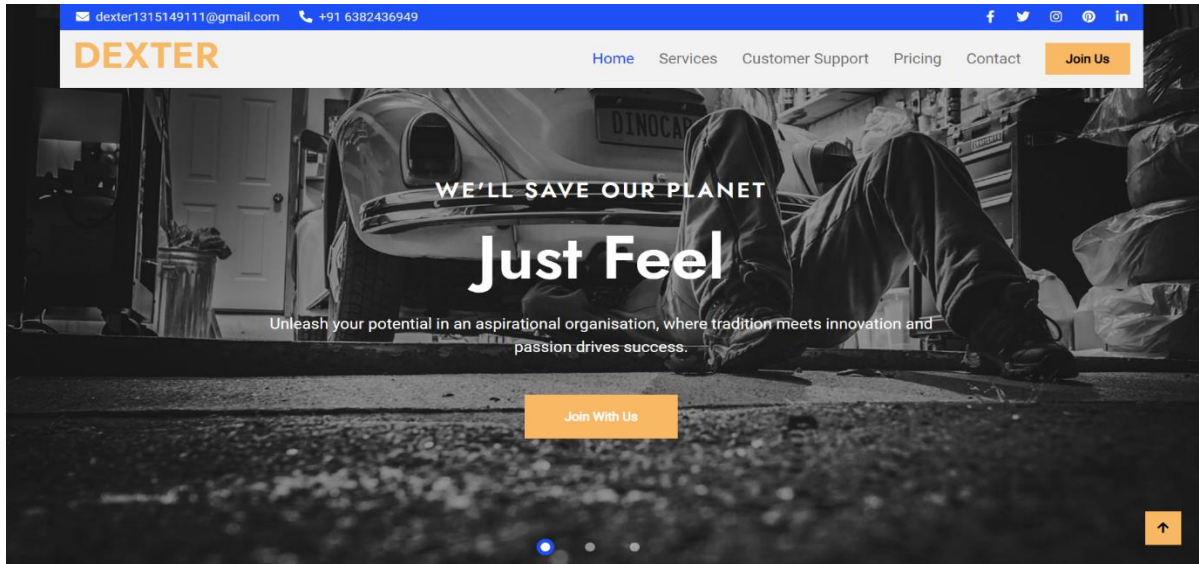


Figure 14 :Home page

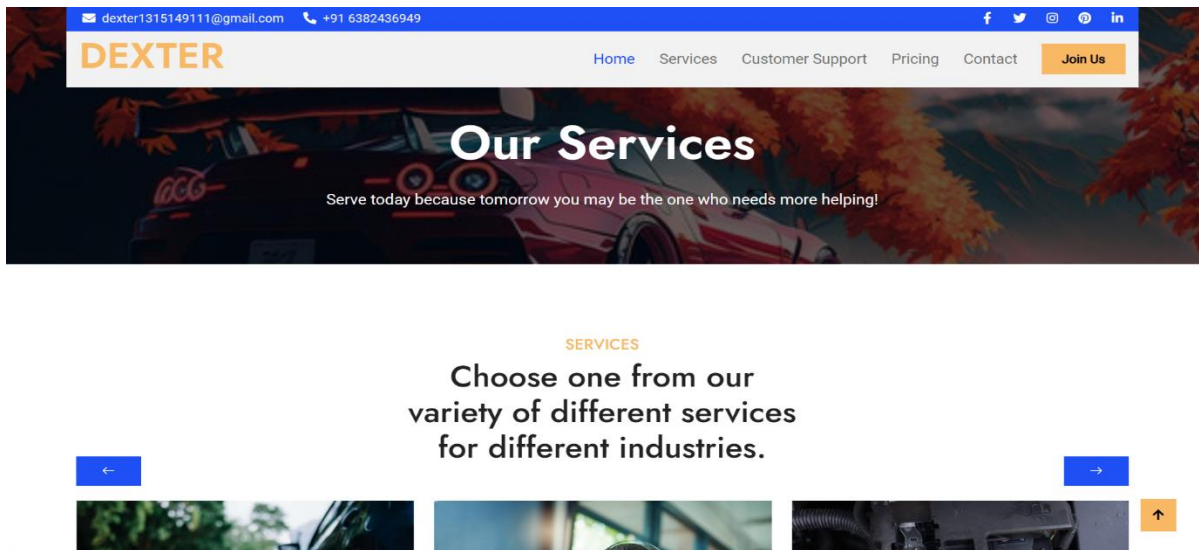


Figure 15 : Services page

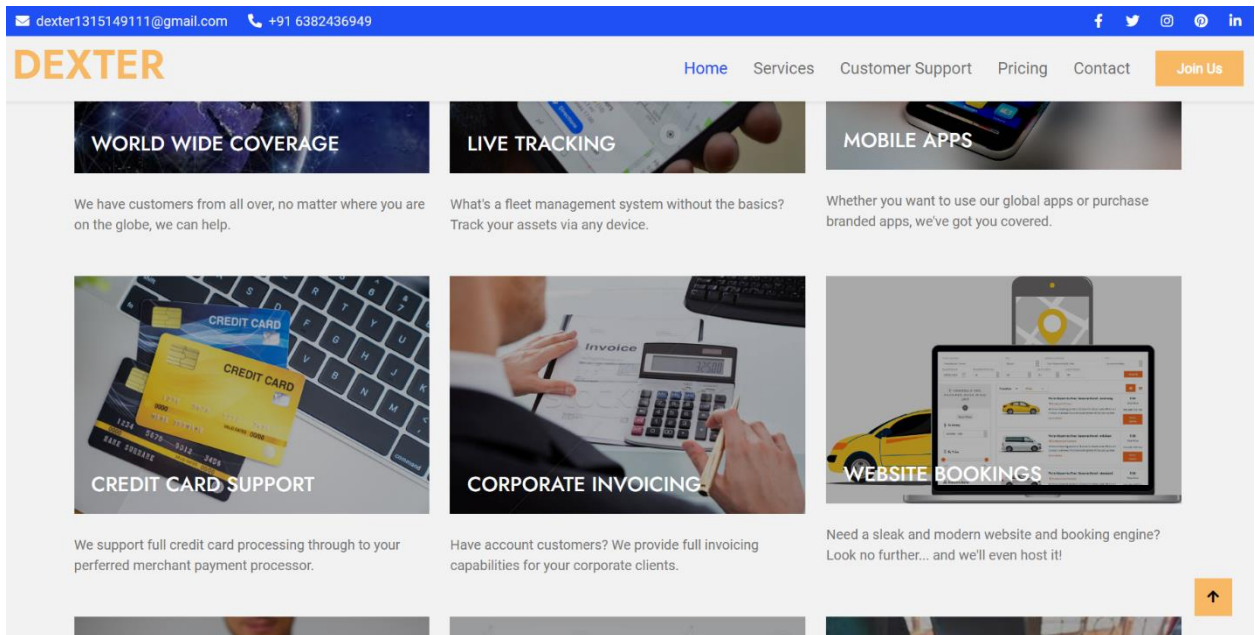


Figure 16 : Features

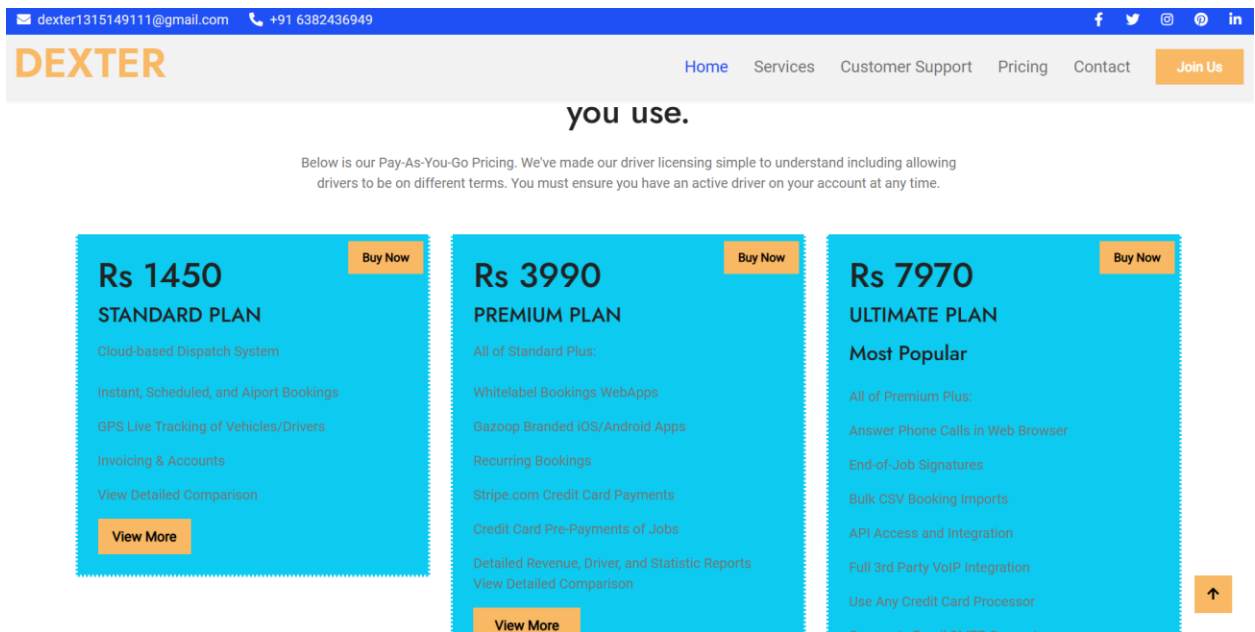


Figure 17 : Membership

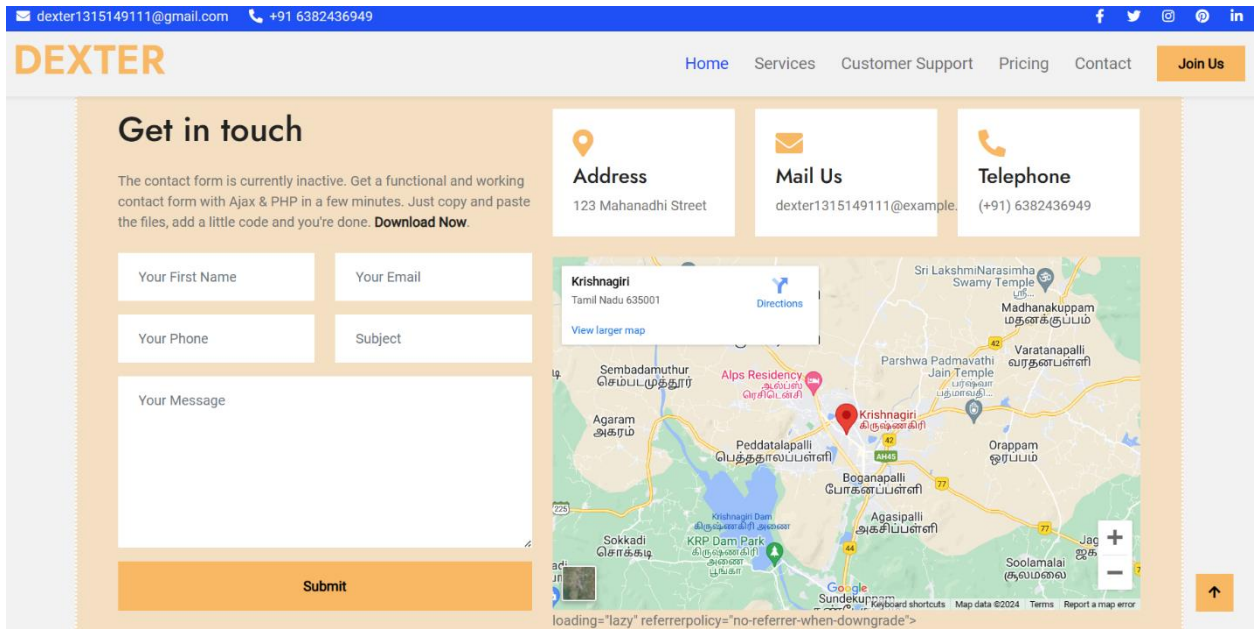


Figure 18 : Customer Feedback

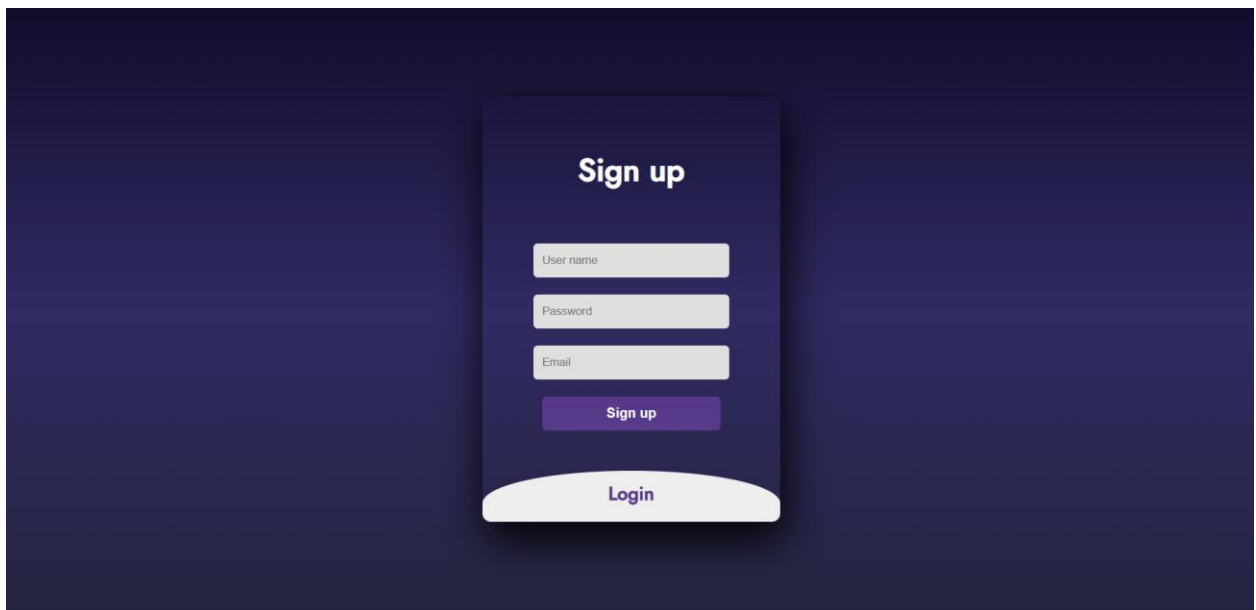


Figure 19 : Customer login & Sign up

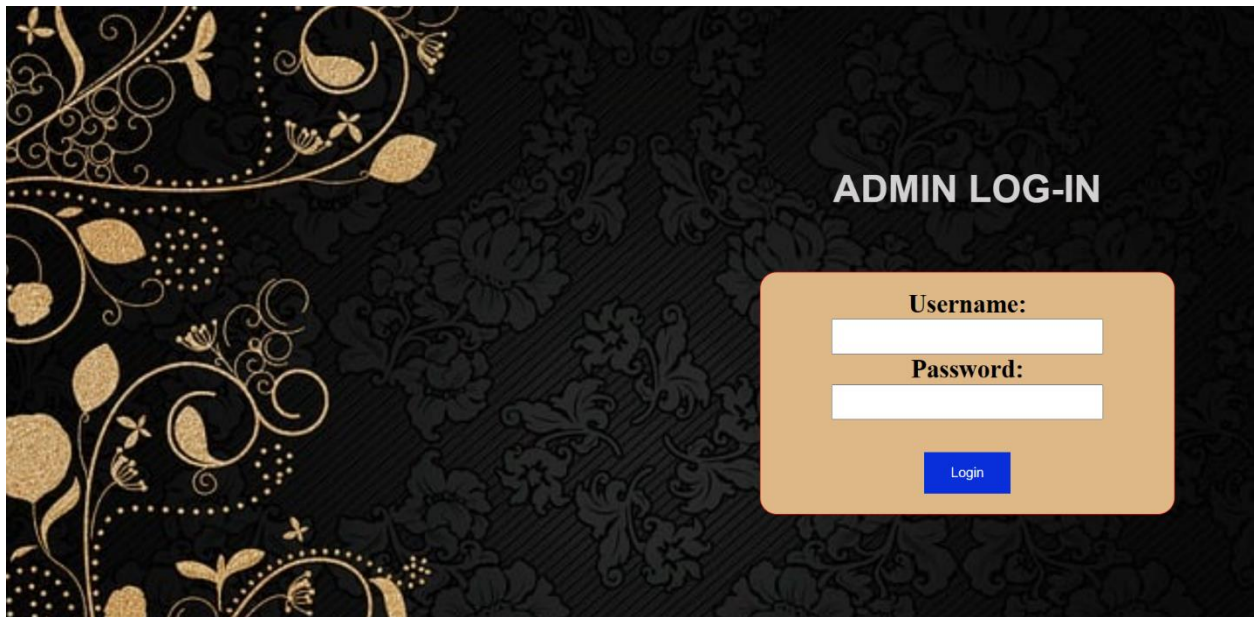


Figure 20 : Admin login

CHAPTER 10

RESULTS

The developing a robust vehicle breakdown management system to address the challenges posed by unexpected vehicle failures. It focuses on providing prompt assistance to customers through a user-friendly interface. Customers can securely register and submit their vehicle details and issues, facilitating quick response. The admin interface empowers authorized personnel to efficiently manage reported problems by sorting and filtering them based on relevant parameters. By leveraging GPS technology, administrators can swiftly locate affected vehicles, ensuring timely assistance. This system promises to streamline the reporting and resolution process, enhancing overall efficiency and effectiveness in managing breakdown incidents, thereby ensuring the safety and satisfaction of vehicle owners and passengers.

CHAPTER 11

CONCLUSION

The vehicle breakdown assistance system offers a robust solution to the challenges associated with unexpected vehicle failures. By providing a user-friendly platform for customers to register their details and report breakdown incidents, the system ensures swift and efficient assistance during times of distress. The inclusion of an administrative interface empowers personnel to manage reported issues effectively, allowing for quick identification and resolution of breakdown vehicle. Through the streamlining of the reporting and resolution process, the system not only enhances customer satisfaction by delivering prompt assistance but also equips administrators with the necessary tools to efficiently manage breakdown incidents. Overall, the implementation of this system holds the potential to significantly improve the reliability and effectiveness of vehicle breakdown management services in today's dynamic and fast-paced environment.

CHAPTER 12

REFERENCE

- [1] Garcia, M., & Rodriguez, P. (2023). "Implementing a Secure User Authentication System for Vehicle Breakdown Assistance Platforms." *International Journal of Cybersecurity*, 2023(1), 30-42.
- [2] Taylor, R., & Evans, D. (2024). "Utilizing Geographic Information Systems for Efficient Dispatching of Breakdown Assistance Vehicles." *Journal of Spatial Analysis and Decision Making*, 2024(2), 115-128.
- [3] Brown, L., & Davis, R. (2019). "Design and Implementation of an Online Vehicle Breakdown Reporting System." *Proceedings of the International Conference on Information Systems*, 2019, 112-125.
- [4] Patel, K., & Gupta, S. (2020). "Enhancing Efficiency in Vehicle Breakdown Assistance Services Through Technology Integration." *Journal of Transportation Engineering*, 2020(2), 78-89.
- [5] White, M., & Clark, E. (2021). "Administrative Tools for Managing Vehicle Breakdown Incidents: A Case Study." *Journal of Management Information Systems*, 2021(4), 210-223.
- [6] Lee, H., & Kim, S. (2022). "User Interface Design Considerations for Vehicle Breakdown management Systems." *International Conference on Human-Computer Interaction*, 2022, 75-88
- [7] Thakur, A., & Dhiman, K. (2021). Chat Room Using HTML, PHP, CSS, JS, AJAX. *International Research Journal of Engineering and Technology (IRJET)*, 08(June), 1948–1951.

[8] Thakur, Amey and Karan Dhiman. "Chat Room Using HTML, PHP, CSS, JS, AJAX." ArXiv abs/2106.14704 (2021): n. pag.

[9] Waspodo, Bayu, Qurrotul Aini, and Syamsuri Nur. "Development of car rental management information system." In Proceeding International Conference on Information Systems For Business Competitiveness (ICISBC), pp. 101-105. 2011.

[10] Osman, Mohd Nizam, Nurzaid Md Zain, Zulfikri Paidi, Khairul Anwar Sedek, Mohamad Najmuddin Yusoff, and Mushahadah Maghribi. "Online Car Rental System Using Web-Based and SMS Technology." Computing Research & Innovation (CRINN) 2 (2017): 277.

[11] Fink, Andreas, and Torsten Reiners. "Modeling and solving the short-term car rental logistics problem." Transportation Research Part E: Logistics and Transportation Review 42, no. 4 (2006): 272-292.

[12] Khaled, Mr Shah Mostafa, Shamsil Arefin, Datta Sree Rajib Kumar, and Ariful Hossain Tuhin. "Software Requirements Specification for Online Car Rental System." (2015).

[13] Soares, Hécio A., and Raimundo S. Moura. "A methodology to guide writing Software Requirements Specification document." In 2015 Latin American Computing Conference (CLEI), pp. 1-11. IEEE, 2015.

[14] Carroll, William J., and Richard C. Grimes. "Evolutionary change in product management: Experiences in the car rental industry." Interfaces 25, no. 5 (1995): 84-104.

- [15] Prof. Rashmi Tundalwar ,Monika Kadam, Neelima Sutar , Pooja Dorge,” A Car Breakdown Service Station Locator System”, International Journal Of Advance Scientific Research And Engineering Trends, Volume 3,Issue 4,April 2018.
- [16] Hanamant B. Sale, Dharmendra Bari, TanayDalvi, Yash Pandey,” Online Management System for Automobile Services”, International Journal of Engineering Science and Computing, Volume 8, Issue 2, Feb 2018.
- [17] Ajith Kumaar. D, Balakrishnan, Sree Subha. S, Harin. K , "On Road Vehicle Service Finder", International Journal of Scientific Research in Computer Science, Engineering and Information Technology (IJSRCSEIT), ISSN : 2456-3307, Volume 5 Issue 2, pp. 929-936, March-April 2019.
- [18] Adamy Shyam, Nitin Mukesh,” A Django Based Educational Resource Sharing Website”, Journal of Scientific Research, Volume 64, Issue 1, 2020.
- [19] Prof. MS. Pranita P. Deshmukh, Mr. Yash S. Puraswani, Mr. Aditya D. Attal, Mr. Prasad G. Murhekar, Mr. Vivek A. katole, Mr. Vidhitya M. Wankhade,” On Road Vehicle Breakdown Assistance System”, International Journal of Engineering Applied Sciences and Technology, Vol. 4, Issue 11, 2020.
- [20] S. Bhatia, S. Hila,” A New Approach for Location based Tracking”,IJCSI International Journal of Computer Science Issues, Vol. 10, Issue 3, No 1, May 2013.