An Najah National University
Faculty of Engineering & Technology
Department Computer Engineering

CarHub Applecation

Prepared By:
EzzAldeen Shareef Mletat
Obada Kamal Jabr

Supervised By:
Dr. Haya Samaana

A Graduation Project submitted to the Department of Electrical and Computer Engineering in partial fulfillment of the requirements for the degree of B.Sc. in Computer Engineering

June, 2024

Acknowledgment

"We would like to express our heartfelt gratitude to Dr.Haya Samaana for her continuous guidance and unwavering support throughout our project. Special thanks to the Computer Engineering Department at An Najah National University for providing a conducive learning environment and to all the academics who generously shared their knowledge. To our friends and family, your encouragement and belief in us were invaluable. Thank you for being the driving force behind our success."

Disclaimer statement

This report was written by Ezz Mletat and Obada Jaber at the Computer Engineering Department, Faculty of Engineering, An-Najah National University. The information presented in this report is intended for informational purposes only. The authors and the organization are not liable for any actions taken or decisions made based on the information provided. Readers are advised to consult with relevant experts or professionals before making any decisions or taking any actions. The authors do not endorse or guarantee the accuracy or completeness of the information in this report.

Abstract

CarHub is a platform designed to simplify and optimize the process of finding garages, car dealerships, car rental offices, and car parts shops. By integrating a robust database of car information, CarHub makes it easy to locate these automotive services. Given the importance of transportation and vehicle maintenance, CarHub aims to enhance the overall experience for car renters, buyers, and repair seekers by providing essential information, improving competitiveness, and increasing efficiency in the automotive sector. Key aspects include a comprehensive database of historical customer and car data, rental management features for agencies, real-time chat between agencies and customers, access to car service information, and agency management capabilities for buying and selling cars. While other platforms exist, none offer the depth of integrated data and customer ratings provided by CarHub.

Table of Contents

List of Figures

Chapter 1

Introduction

Contents

In the context of the technological advancement we are experiencing, the use of internet sites and smart mobile applications has become indispensable in people's lives. These platforms offer a variety of life services that save time and effort for users. CarHub aims to meet the needs of car owners, renters, and buyers by providing a comprehensive platform to access essential automotive services. It serves as an advanced solution to traditional methods of locating and managing car rentals, dealerships, garages, and car parts shops, facilitating a more efficient and user-friendly experience , instead of going around to ask around a place to fulfill your demands , you can use CarHub. Throughout the software development lifecycle, CarHub underwent design, development, and testing phases, resulting in a robust graphical user interface (GUI) with custom technical components to implement its functions.

1.1 Motivation

Car owners, renters, and buyers are the primary target group for CarHub, which simplifies the process of finding car rental offices, car dealerships, garages, and car parts shops. The platform provides a seamless experience for users to locate and access these services without the need to physically visit multiple locations. This reduces the time and effort typically required for such tasks. Additionally, CarHub improves communication between car rental agencies , dealerships , garages and car parts shops with customers through real-time chat features, making it easier to address inquiries and coordinate services.

The platform also targets automotive service providers, as it streamlines their operations and enhances communication among employees for work-related inquiries -such as garage and car parts shop -. The primary motivation for CarHub's concept is the realization that traditional methods of finding and managing automotive services are often fragmented and inefficient. CarHub is the first practical solution that integrates these services into a single, user-friendly mobile application and system, aligning with the aspirations of the automotive industry to offer more advanced and convenient services.

1.2 Objectives

The objective of CarHub is to create a user-friendly platform that allows car owners, renters, and buyers to manage their automotive needs from one convenient location. By simplifying access to car rentals, dealerships, garages, and car parts shops, CarHub enhances the overall user experience and improves service efficiency. The platform aims to streamline rental management, facilitate real-time communication between service providers and customers, and offer a comprehensive database for better service insights.CarHub mainly is about reducing the time and effort required for these tasks, ultimately advancing the automotive industry's service capabilities.

1.3 Organization of the report

Our report is thoughtfully divided into key sections to guide you through the CarHub experience. Begin with the Introduction and Motivation and objective and aims, followed by the Methodology detailing our approach.

Transition to the Results and Analysis for empirical findings and interpretations. The Discussion section places results in a broader context, while Conclusions and Recommendations summarize achievements and suggest future directions.

The report concludes with References, acknowledging our sources.

Chapter 2

Constraints and Earlier Coursework

Contents

2.1 Constraints and Limitations

When we started this project, we faced various problems and challenges. One of the main issues was building a strong base and really understanding the basic concepts. We also had difficulty with new programming languages that we were not familiar with, as we had no experience in developing smartphone apps before. Additionally, we struggled with large software programs that were hard to install on devices. Time was another big challenge, as we needed to organize our schedule to make sure we had enough time for the project. Finding educational resources was tough too, since most training materials were focused on other programming languages like Flutter and Java. This meant we had to spend more time on research and solving problems. To overcome these hurdles, we looked for help from experts and knowledgeable individuals to find the right solutions.

2.2 Earlier coursework

The Computer Engineering courses that we attended set us up with valuable knowledge that we applied in developing this app. The principles of Web Programming and Object-Oriented Programming, which we learned in our coursework, were the building blocks of this project. Additionally, we augmented our skills by enrolling in online courses on Node.js, MongoDB, Flutter, which were instrumental in the successful implementation and development of our app.

2.3 Standards

2.3.1 MVC Pattern

The MVC - Model View Controller - pattern was employed in our system. We could divide the entire project into different layers to make it easier to follow the flow of work. These layers are as follows:

- Model Layer: - Manages data related to car rentals, dealerships, garages, and user profiles. - Contains business logic for rentals, car listings, and services. - Interacts with databases and APIs to fetch and store data.
- View Layer: - Represents the user interface. - Displays data to users and updates based on changes from the Model. - Ensures a responsive design for various devices.
- Controller Layer: - Handles user inputs and interactions. - Fetches data from the Model and updates the View. - Manages the workflow between the Model and the View.

By using the MVC pattern, CarHub's structure becomes clearer and more manageable, with each layer having a distinct role.

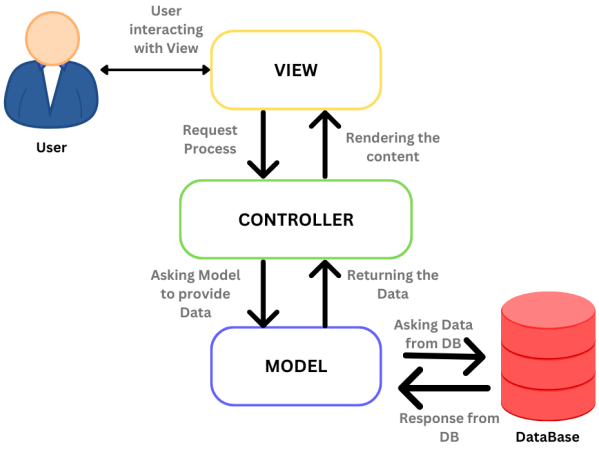


Figure 2.1: MVC pattern Figure explanation

2.3.2 Agile Model

The Agile methodology is a project management approach that involves breaking the project into phases and emphasizes continuous collaboration and improvement. Teams follow a cycle of planning, executing, and evaluating. We classified the tasks we had to do into the following five groups:

- Brainstorming: Gather all needed requirements and features required.
- Design: Create the UML diagram to clarify the functionality of new features and their interaction with the system.
- Development: Begin writing the code for the front end, followed by the back end for each screen in the app.
- Testing: Connect the back end with the front end and ensure that every page works correctly.
- Feedback: Continuously get feedback from the supervisor and colleagues to ensure the app meets the requirements.

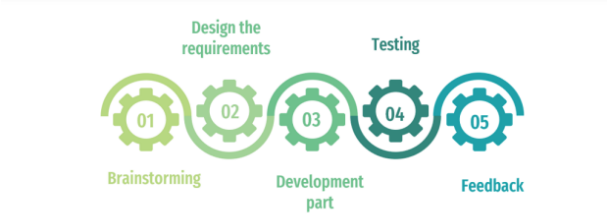


Figure 2.2: Agile methodology Figure

2.4 Earlier work

In the context of platforms that offers cars things , there is "sho bdk mn falsteen" which is a Palestinian website that is about putting offers of cars to sell and etc , but in our app we put all these offers and all other services such as garages and rentals , with filtering model that can give you even the precise things you are searching for , or the best deals you might like , with a real-time chatting service to get more information from the merchant who is the owner of this service .

Chapter 3

Literature Review

CarHub applecation is designed to improve how people find garages, car dealerships, car rental offices, and car parts shops. Research shows that maps features make it easier and more convenient for users to find what they need. For example, systems with location services see up to a 40% increase in user satisfaction and engagement. CarHub uses this technology to help users quickly locate nearby services.

CarHub also integrates a large database of customer and car information, making it easy to access past records and reviews. This kind of database can improve service quality and user trust. Studies indicate that platforms with comprehensive data see higher user satisfaction and retention.

Real-time chat between merchants and customers is another important feature. Instant communication helps resolve issues quickly and improves the customer experience. Research suggests that platforms offering real-time communication have higher rates of customer satisfaction and loyalty.

Moreover, CarHub provides detailed information about car maintenance, repairs, and detailing services. Access to such information helps users make informed decisions and keeps their vehicles in good condition.

In summary, CarHub sets itself apart by combining extensive data integration, user-friendly , efficient rental, and real-time communication, all of which contribute to a better overall experience in the automotive sector.

Chapter 4

Methodology

Contents

4.1 Introduction

This chapter outlines the methodology adopted for the design and implementation of the CarHub Application , with a specific emphasis on the design process, data collection, analysis, and processing.,tools and technologies utilized and details of usage .

4.2 Non-Functional Requirements

- **Adaptive User Interface Design** For the CarHub application, adapting to different screens is essential to ensure a smooth experience across various devices. Using responsive design, the app adjusts fonts, buttons, and layouts to fit different screen sizes. Whether accessed on smartphones, tablets, or desktops, CarHub maintains a consistent and user-friendly interface. Extensive testing guarantees that users can navigate effortlessly, regardless of their device, enhancing accessibility for a wide range of users.

- **Ease-of-use** CarHub is designed to be simple and easy to use. Whether users are looking for garages, dealerships, rental offices, or car parts shops, the menus are clear, and the features are well-organized. The process of creating an account and using the app is straightforward, ensuring that users can access automotive services with ease. The goal is to facilitate the search and booking of services for everyone, making the process simple and accessible at any time and place.

- **Usability**
  The CarHub app is highly user-friendly. Both customers and service providers find it easy to navigate and use. The interface is intuitive, with straightforward forms for account creation and service booking. CarHub aims to make the process of finding and accessing automotive services a breeze for all users, ensuring a smooth and efficient experience.

4.3 Tools, Methods and Programming Languages

In order to develop the CarHub application, we utilized a variety of tools, methods, and programming languages. For the front-end development, we used Flutter and Dart programming languages, which allowed us to create a modern and intuitive user interface. For the backend development, we utilized Node.js and Express.js to create a RESTful API for handling the app's data and logic. We also designed a non-relational database schema using MongoDb to store and manage the app's data. Throughout the development process, we followed the Agile methodology .

4.3.1 Client Side

- **Design**
  The design of the client side of the CarHub App began with the selection of the idea, followed by extensive research to devise a creative and innovative concept for the application's design.

  Our decision to pursue this idea was based on the suffer people get while searching for automotive services in Palestine , and a desire to create a more effective and user-friendly app to deal with this sector of industry.

  Rather than using a pre-designed application UI template, we devoted considerable effort to designing a unique user interface (UI) for the application. The UI design was developed internally based on the research conducted, taking into consideration user requirements and industry top 10 thoughts that come to mind while searching for a service.

  This allowed us to create a custom, user-friendly, and aesthetically pleasing design that is consistent with the overall theme and brand of the application. and we carefully selected a color theme that would enhance the user experience and enable them to feel at ease while using the application.

  The design of the majority of the application and website pages was kept simple, easy to use, and basic to ensure user-friendliness. In addition, we incorporated modern design elements to keep the application up-to-date and in line with current trends in the industry.

- **Framework**
  We decided to leverage the power of Google's cross-platform UI framework for client-side development. Flutter is a popular open-source mobile application development framework created by Google that utilizes the Dart programming language.

  It offers a wide range of features,including hot-reload, widget-based development, and platform-specific widgets,among others.

  Through Flutter, we can create mobile apps for iOS and Android with a single code base that renders everything on its own, eliminating the need for previous native programming for these two platforms and saving a significant amount of time and effort in developing cross- platform apps.

  Therefore, Flutter is a viable and efficient tool for developing mobile applications, and we chose to use it for our app for its reliability, speed, and cross-platform capabilities.

  Flutter's popularity has grown steadily over the years, with Google's support and the community's contributions. As of May 2023, Flutter has been downloaded more than 290 million times worldwide, and there are over 120,000 Flutter packages available for developers to use, demonstrating its extensive ecosystem.

- **Programming Languages:**

  The programming language used in Flutter for client-side development is Dart. Developed by Google, Dart is an object-oriented language with similarities to C++, Java, and JavaScript.

  It is a compiled language, type-safe, and has scripting capabilities similar to Python and JavaScript. Our familiarity with these languages from university made Dart easy to learn and work with.

  Despite its relative youth, Dart has a wealth of libraries that simplify and speed up the script-writing process, and its popularity is rapidly increasing.

- **Website:** The website is built using the same client-side technology as the mobile app, leveraging the cross-platform UI framework of Flutter.

  This approach ensures a consistent design and functionality across all platforms, delivering a seamless user experience.

  Additionally, Flutter's capability to render high-performance web applications with the same codebase used for mobile apps offers a significant advantage, reducing development time and effort while maintaining code maintainability.

4.3.2 Server Side

- **Architecture of the Server-Side:**

  We employed a client-server architecture for our application , allowing both the website and mobile application to use the same API. Our servers are designed to support a RESTful API, facilitating efficient communication between clients and the server.

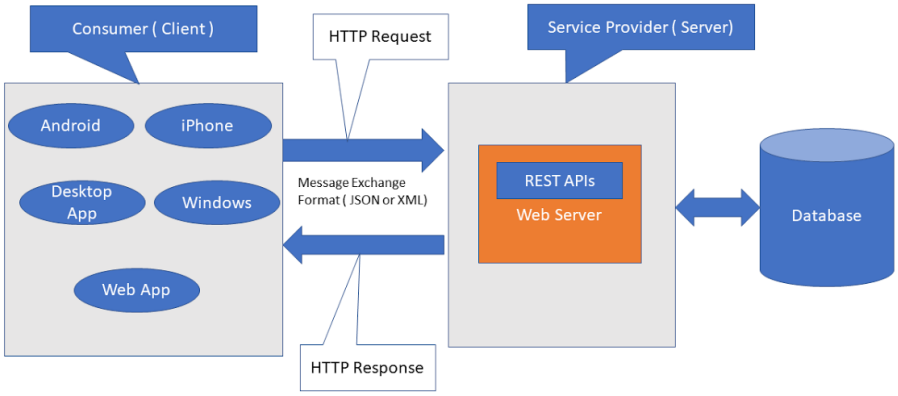


Figure 4.1: Client-Server Architecture

- **Server-Side Framework:**

  We chose to work with Node.js and Express.js for the server-side environment.

  Node.js is known for its ease of learning, extensive documentation, scalability, and performance, making it an ideal choice for our needs.

  Express.js, a widely-regarded back-end framework for Node.js, complements Node.js perfectly, providing a reliable and efficient server-side structure for our application.

4.3.3 Integrated Development Environments (IDEs):

For writing JavaScript code for the REST API, we primarily used Visual Studio Code.

This free, open-source, cross-platform code editor is renowned for its powerful extensions, which allow developers to customize their development environment to meet specific needs.

We found Visual Studio Code to be robust and user-friendly, with advanced features like IntelliSense, debugging, and code navigation that helped us write clean, efficient, and error-free code.

Given that our project required developing for both Android and iOS platforms, we needed an IDE that could handle both.

We chose Android Studio, which allowed us to develop, debug, and test our code for both platforms in one place.

4.3.4 Version Control:

To manage the complexity and size of our CarHub project, we utilized Two separate GitHub repositories. One repository stored the app code, another held the server-side RESTful API code.

Version control was vital for our project, enabling us to track code changes over time, identify and fix issues, and collaborate effectively as a team.

It also provided a secure backup of all the code, allowing us to revert to previous versions if necessary.

This streamlined our development process, reduced errors and conflicts, and ensured the project's stability and scalability.

4.3.5 Testing of the Server and API:

Testing the server and API was crucial to ensure the smooth operation of our CarHub project , We used Postman for our testing.

Postman, a popular tool for API testing, allowed us to automate the testing process, saving time and effort. It enabled us to simulate various requests and responses, test edge cases, and validate the API's functionality and performance and reliability.

4.3.6 DataBase Desgin

Database design is a crucial aspect of any system development project, involving the definition of the structure, organization, and relationships of data within a database. For the CarHub application, the database design aims to capture and manage the relevant information required to effectively handle car-related services such as garages, dealerships, rental offices, and parts shops.
Our choice for the database management system is MongoDB. MongoDB is an open-source, NoSQL database system that allows for flexible data storage and manipulation. Its schema-less nature makes it suitable for handling diverse data types and evolving application requirements. The objective of this design is to accommodate a wide range of database applications and provide robust functionalities to handle the storage, access, and management of data.

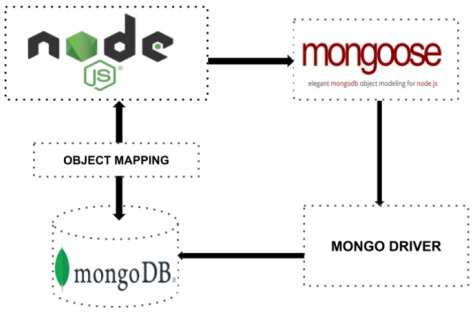


Figure 4.2: MongoDB with Node.js server

Database design is a crucial aspect of any system development project, involving the definition of the structure, organization, and relationships of data within a database. For the CarHub application, the database design aims to capture and manage the relevant information required to effectively handle car-related services such as garages, dealerships, rental offices, and parts shops.

Our choice for the database management system is MongoDB. MongoDB is an open-source, NoSQL database system that allows for flexible data storage and manipulation. Its schema-less nature makes it suitable for handling diverse data types and evolving application requirements.

Our database design for the CarHub application includes a set of collections that represent the various entities involved in the system. These entities include users, car items, and car shops, each with specific attributes necessary for the application's functionality.

The MongoDB is a NOSQL database , but for the complexity of this project we had to make some relationships between collections by some way . The collections in our database are linked using relationships that define how data in one collection is related to data in another. For instance, the carItems collection is linked to the carShops collection via the shopID field to indicate which shop owns which items.

In addition to collections and relationships, the database design includes constraints and indexes that ensure data integrity and improve query performance. These constraints define rules and conditions that data must meet before it can be added or updated in the database.

We also used **Mongoose** To interact with MongoDB, we use Mongoose, a popular Object Data Modeling (ODM) library for Node.js. Mongoose simplifies working with MongoDB by providing a straightforward schema-based solution for modeling application data.

Overall, the database design for the CarHub application is a vital component of the system's architecture. It provides a structured way to organize, store, and retrieve data, enabling efficient and effective management of car-related services.

4.4 System Features and Implementation

4.4.1 Login and Registration Feature

This is the login page which is the first page that appears for the user of the application where clients, merchants, and admins can log in. The backend handles the user type and redirects them to the appropriate frontend.

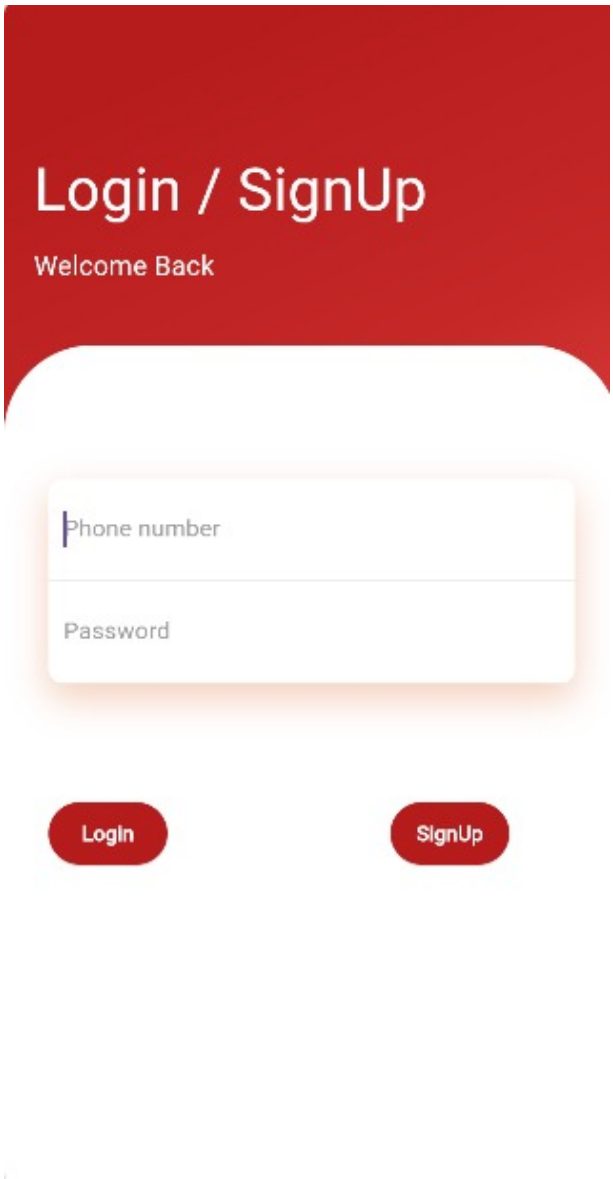


Figure 4.3: Login page for all types

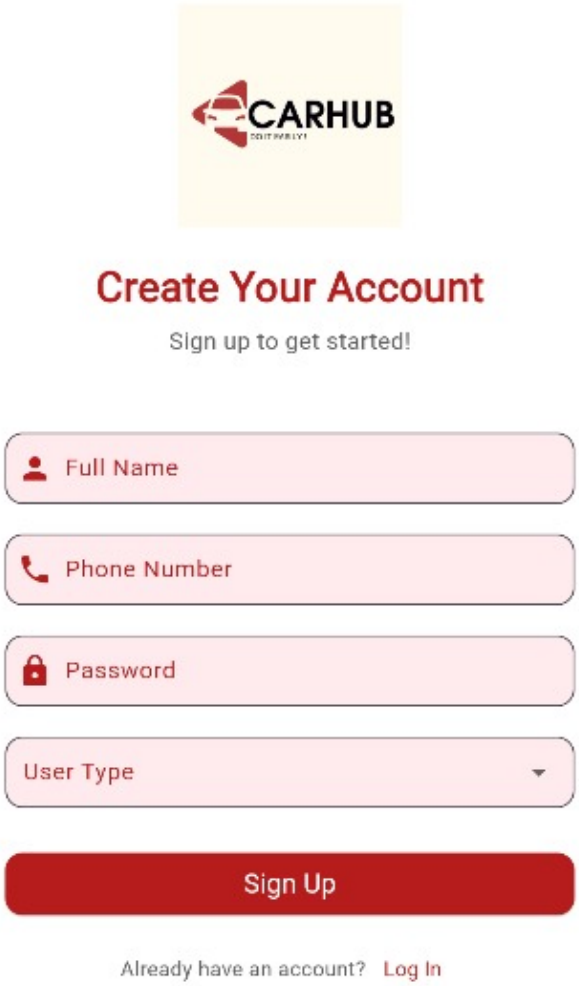


Figure 4.4: Sign Up!

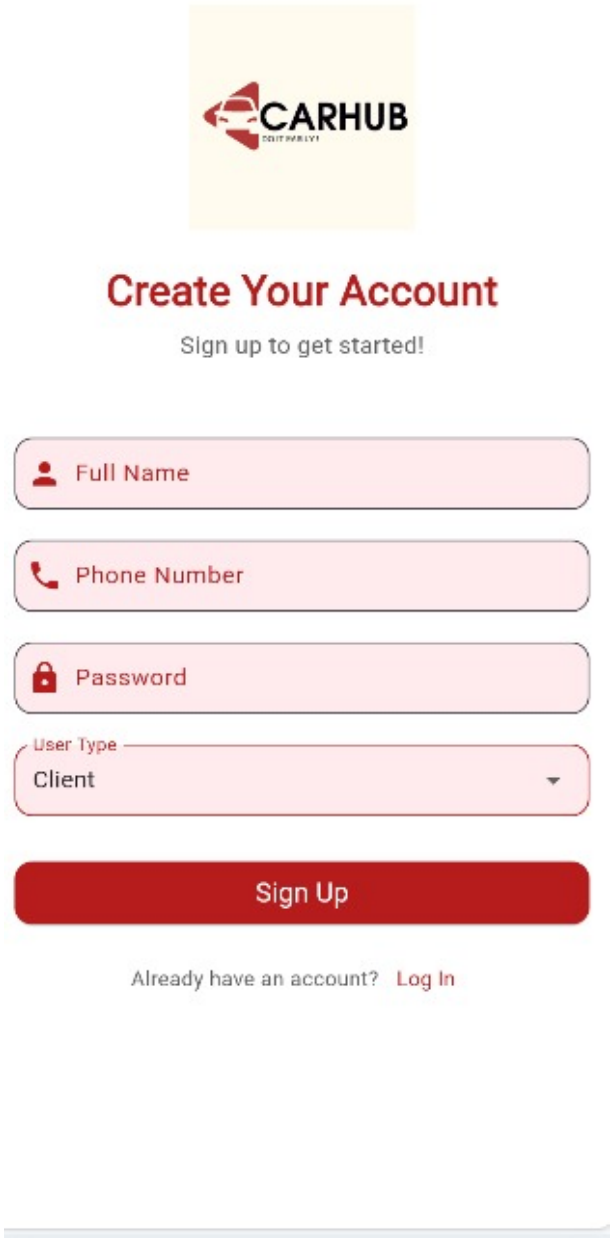


Figure 4.5: Sign Up as a client type

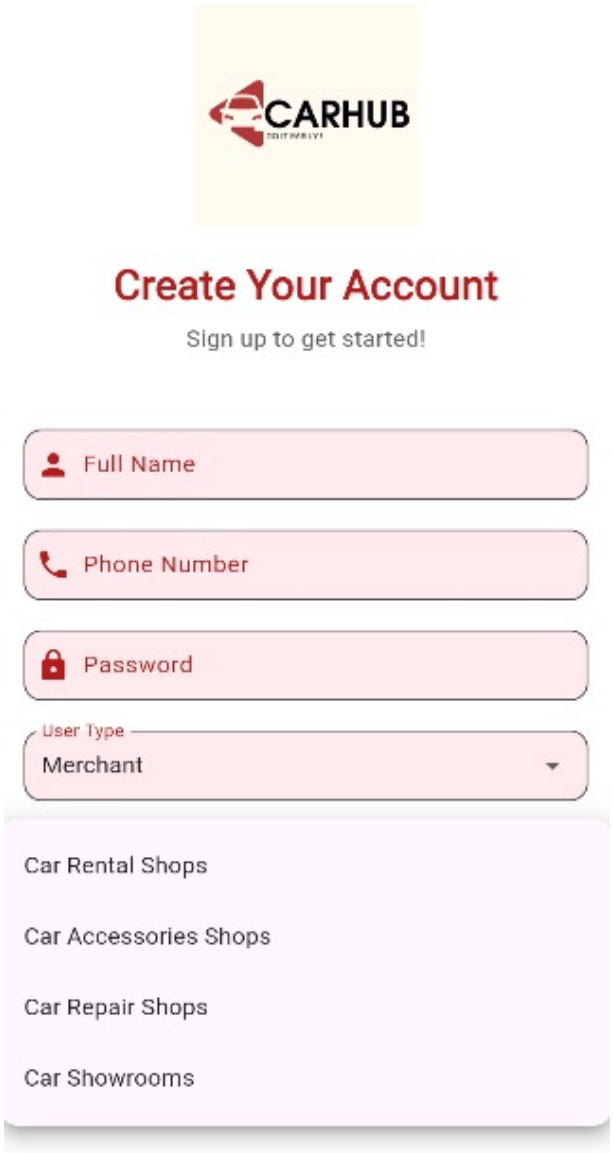


Figure 4.6: Sign Up as a Merchant

4.4.2 Client User Interface (UI)

Now let's go through the client interface that the user will interact with in the application. These interfaces represent the client side of the CarHub app.

Figure 4.7: Client Welcome Screen

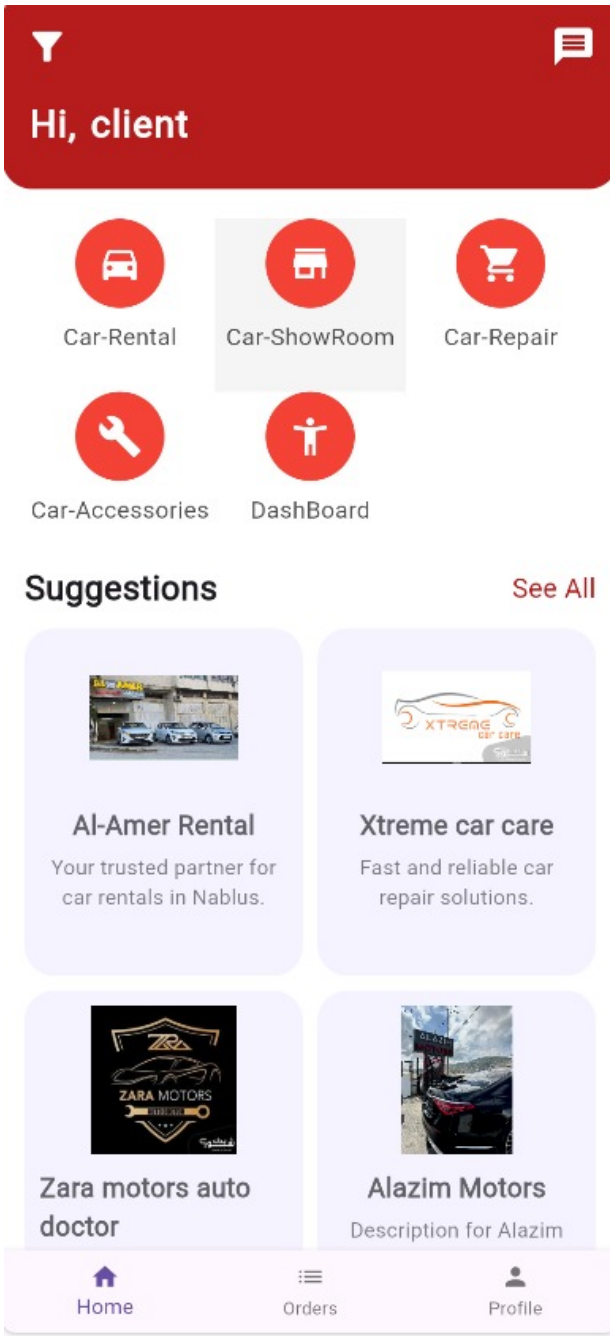


Figure 4.8: Client Home Page
Suggestions are shown randomly to the clients to see if there are any services they might like.

Above, we see the services that the client can choose, such as all the shops or merchants of a particular type (e.g., car dealerships, car accessories, car rents, car garages).

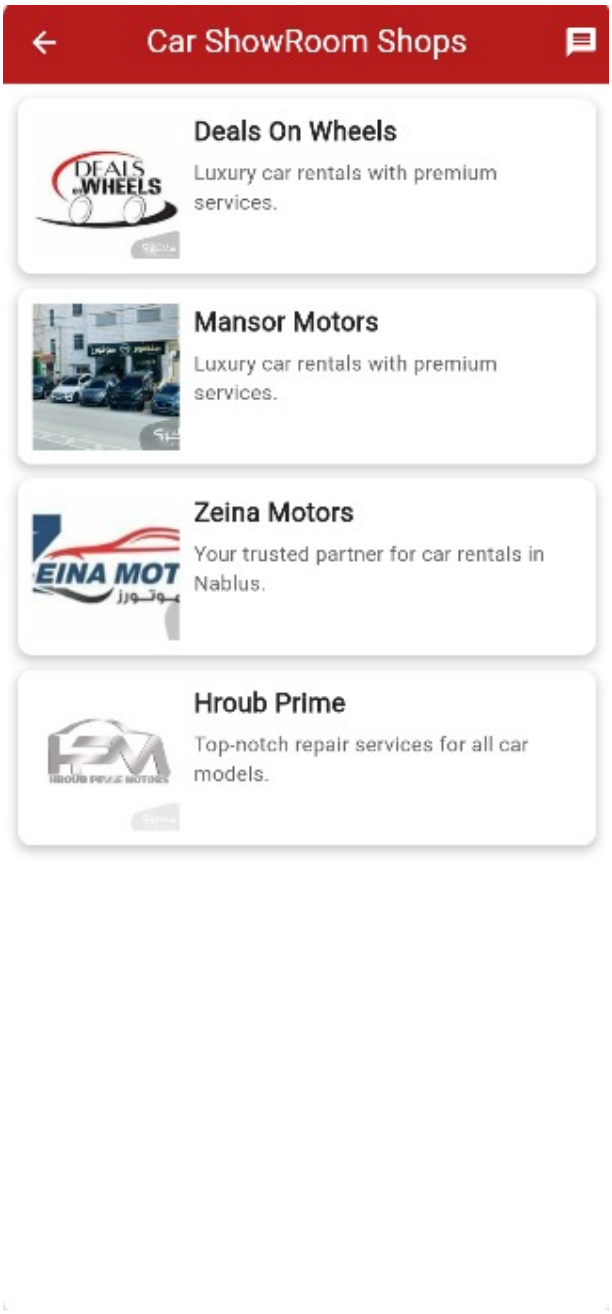


Figure 4.9: Cars Showrooms

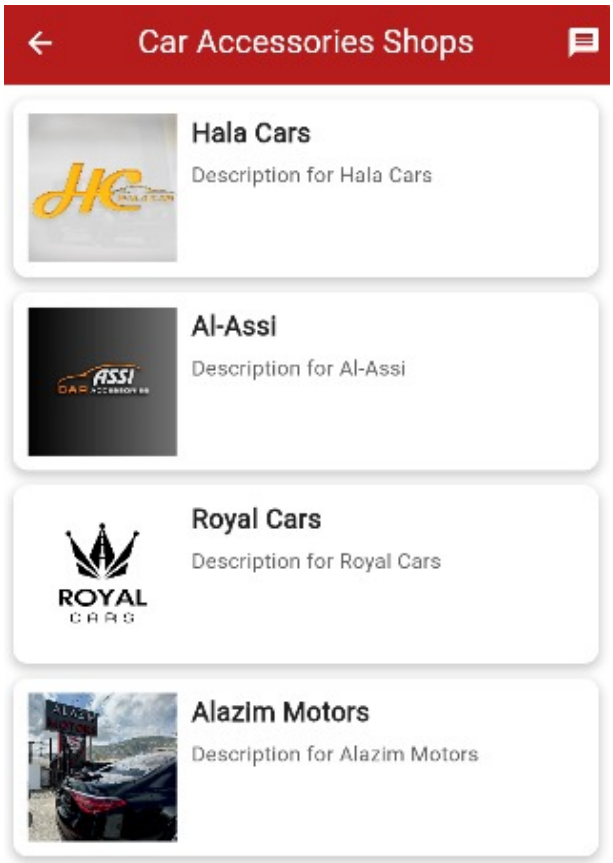


Figure 4.10: Cars Accessories

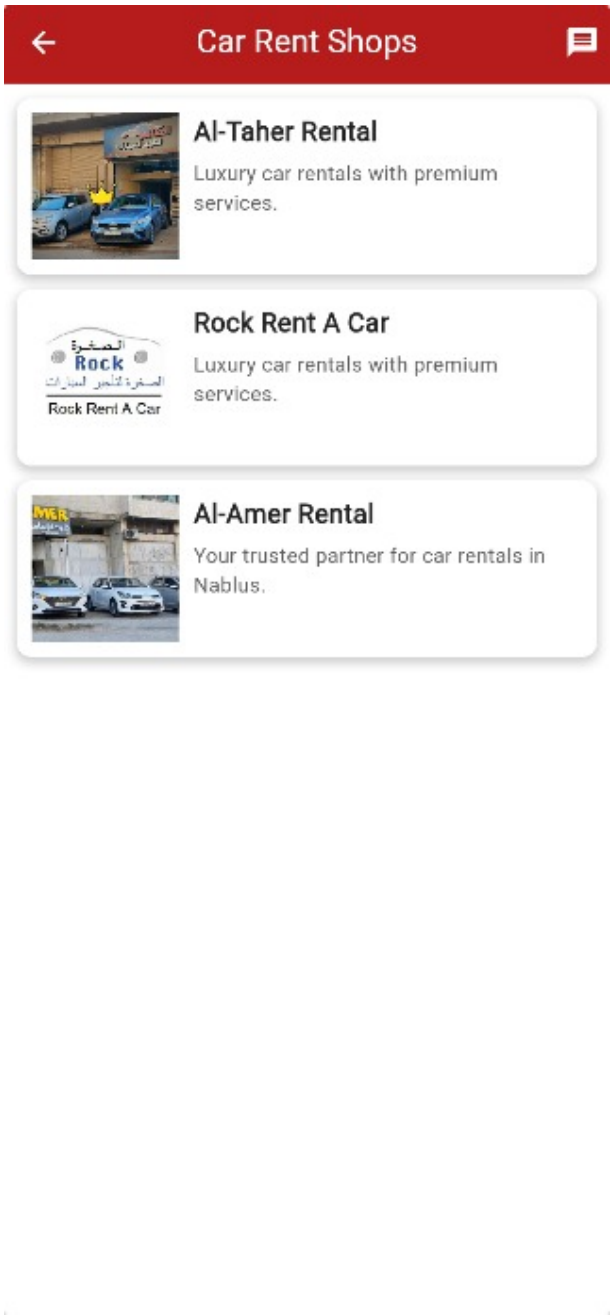


Figure 4.11: Cars Rent offices

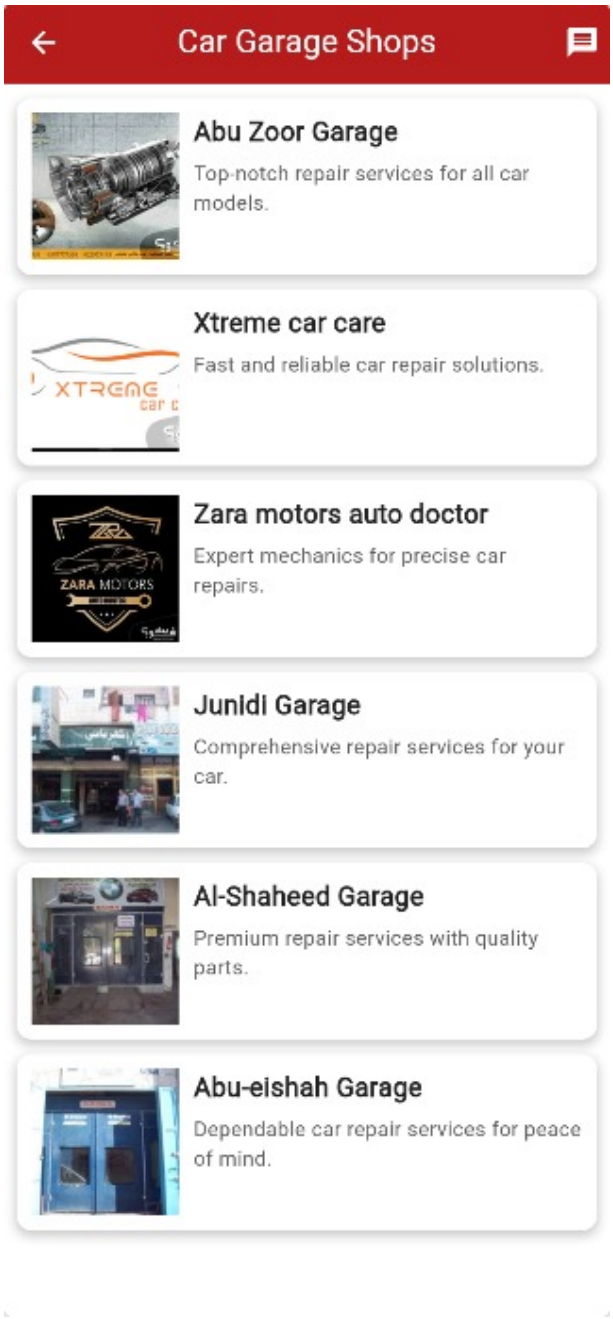
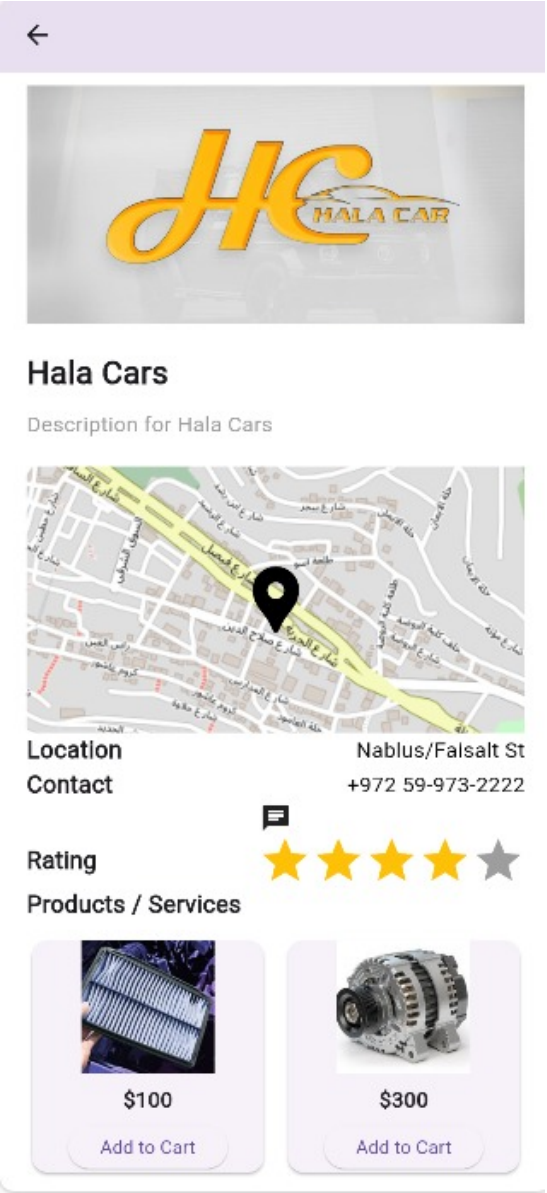
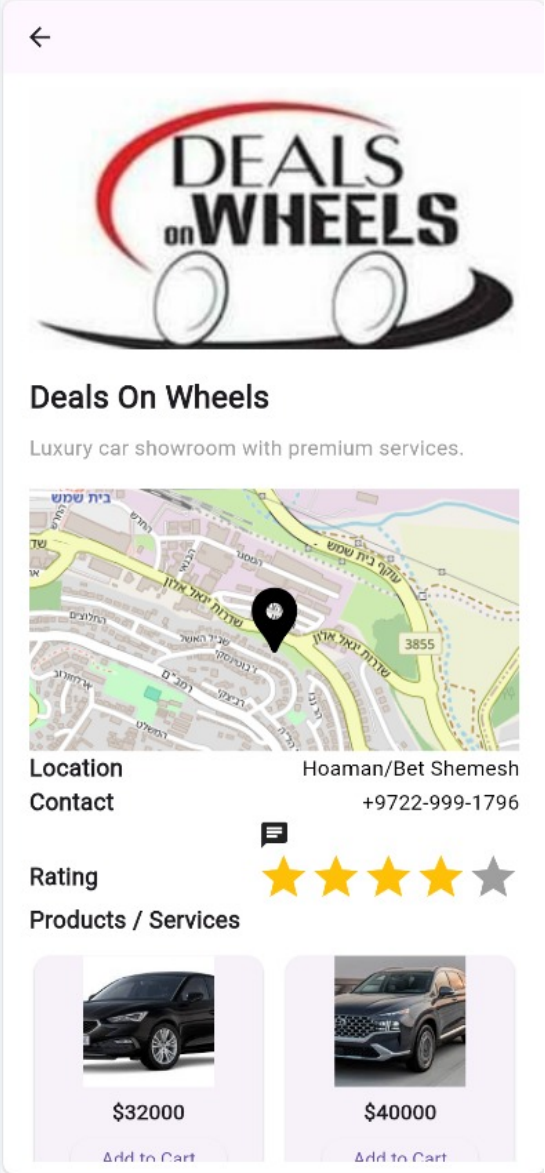


Figure 4.12: Cars Garages

Let's show how the service provider page is displayed to the client after clicking on a shop. For example:



(a) Hala Car Accessories Shop



(b) Deals On Wheels

4.4.3 Filtering for Search

We implemented filtering using fuzzy logic, where each letter the client types provides search results. Let's see how it works!

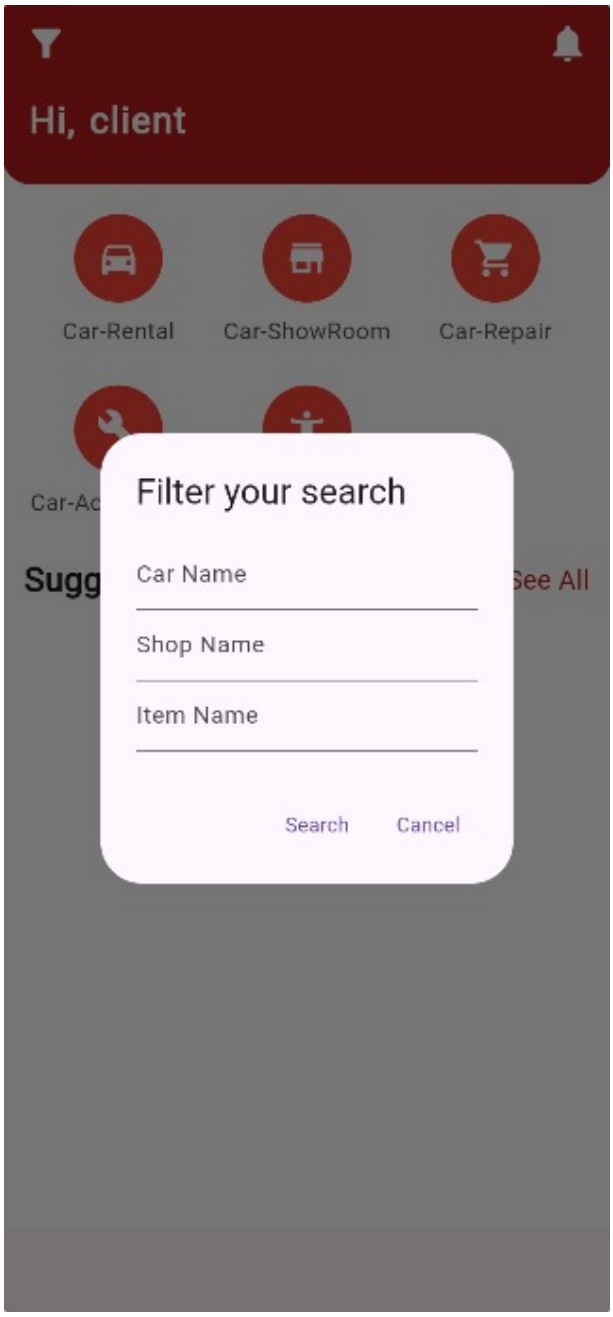


Figure 4.14: Filter Button Clicked

Testing the filtering and finding the required search gives these results:

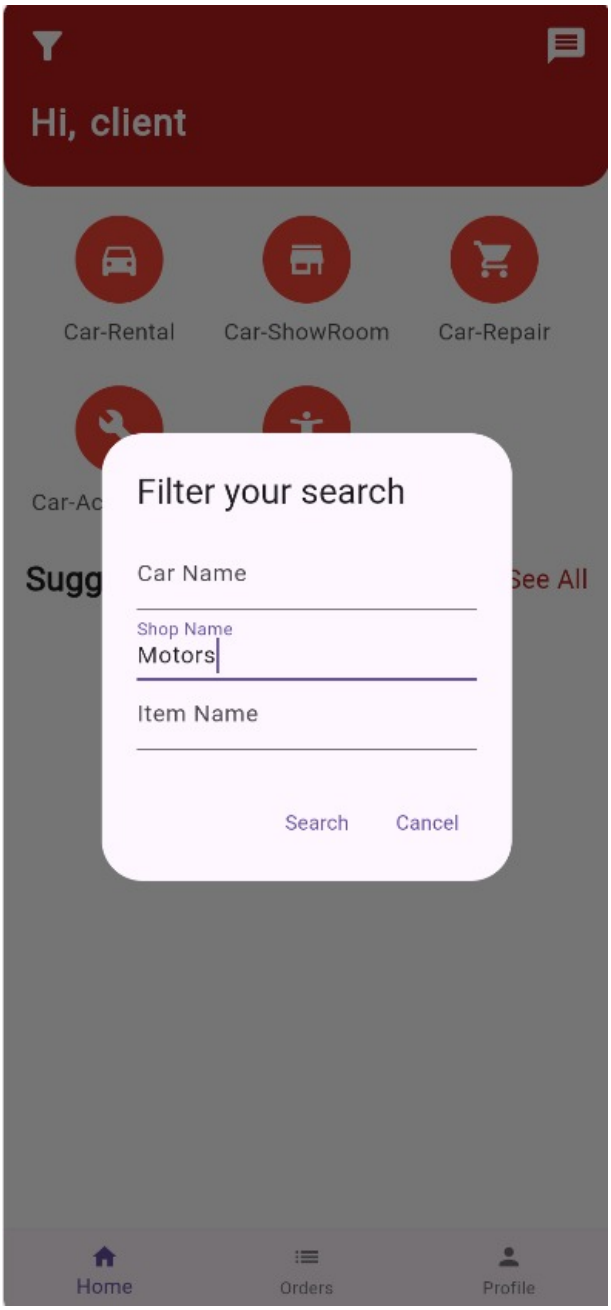


Figure 4.15: Filtering on motors

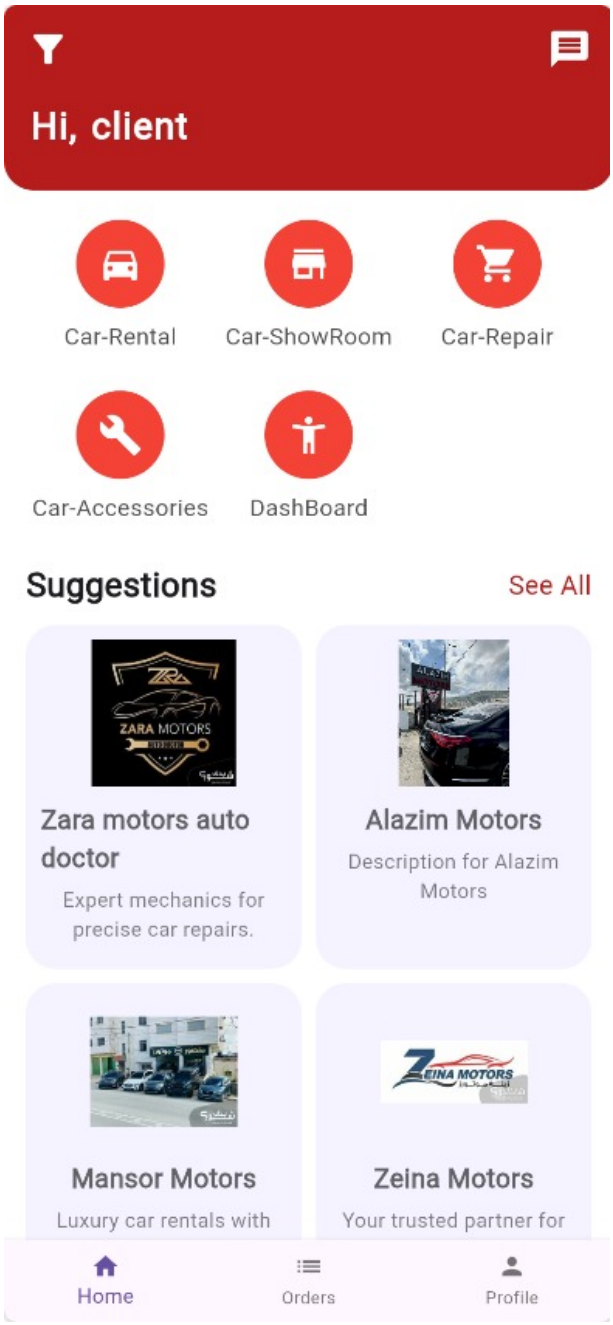


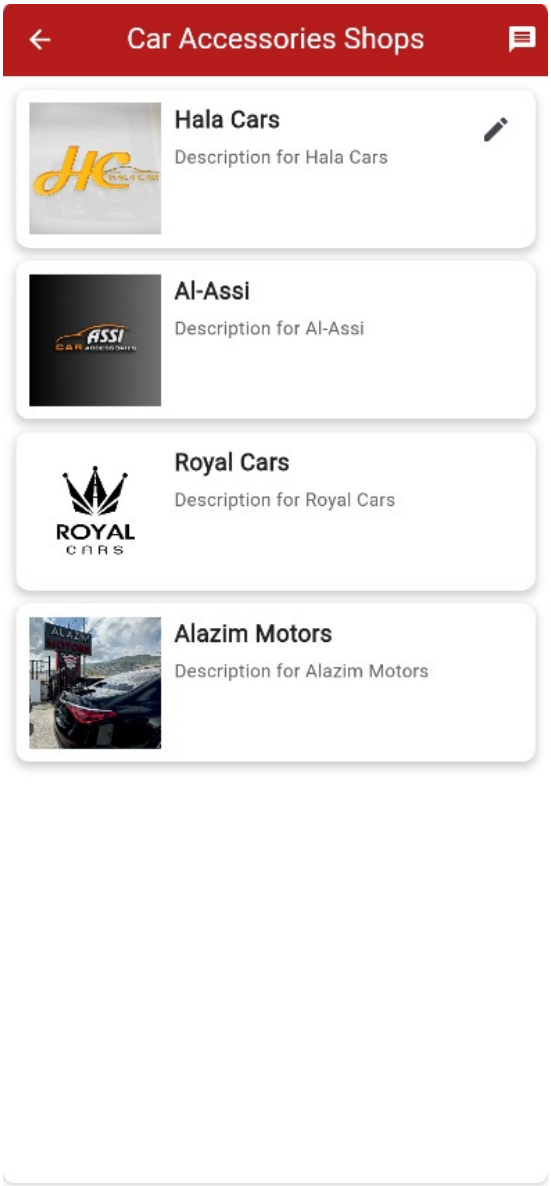
Figure 4.16: Result for filtering

4.4.4 Merchant interface

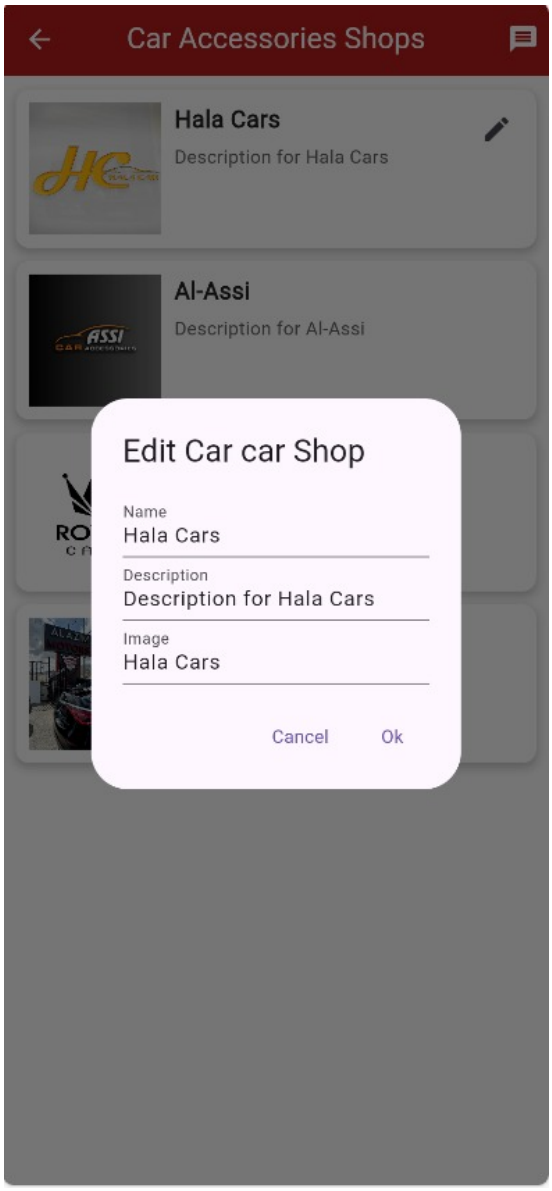
The merchant interface is like the client one but except for a little differences , which are : Merchant can edit only his page , his profile and etc .

Figure 4.17: Welcome Merchant

Now let's take hala car as a merchant !



(a) Hala Car Accessories Shop Merchant



(b) Hala Car Editing

4.4.5 Admin functionalist

The Admin can see the statistics of the activity of clients and others on the dashboard , can edit and delete shops and services , and Can track orders and etc !

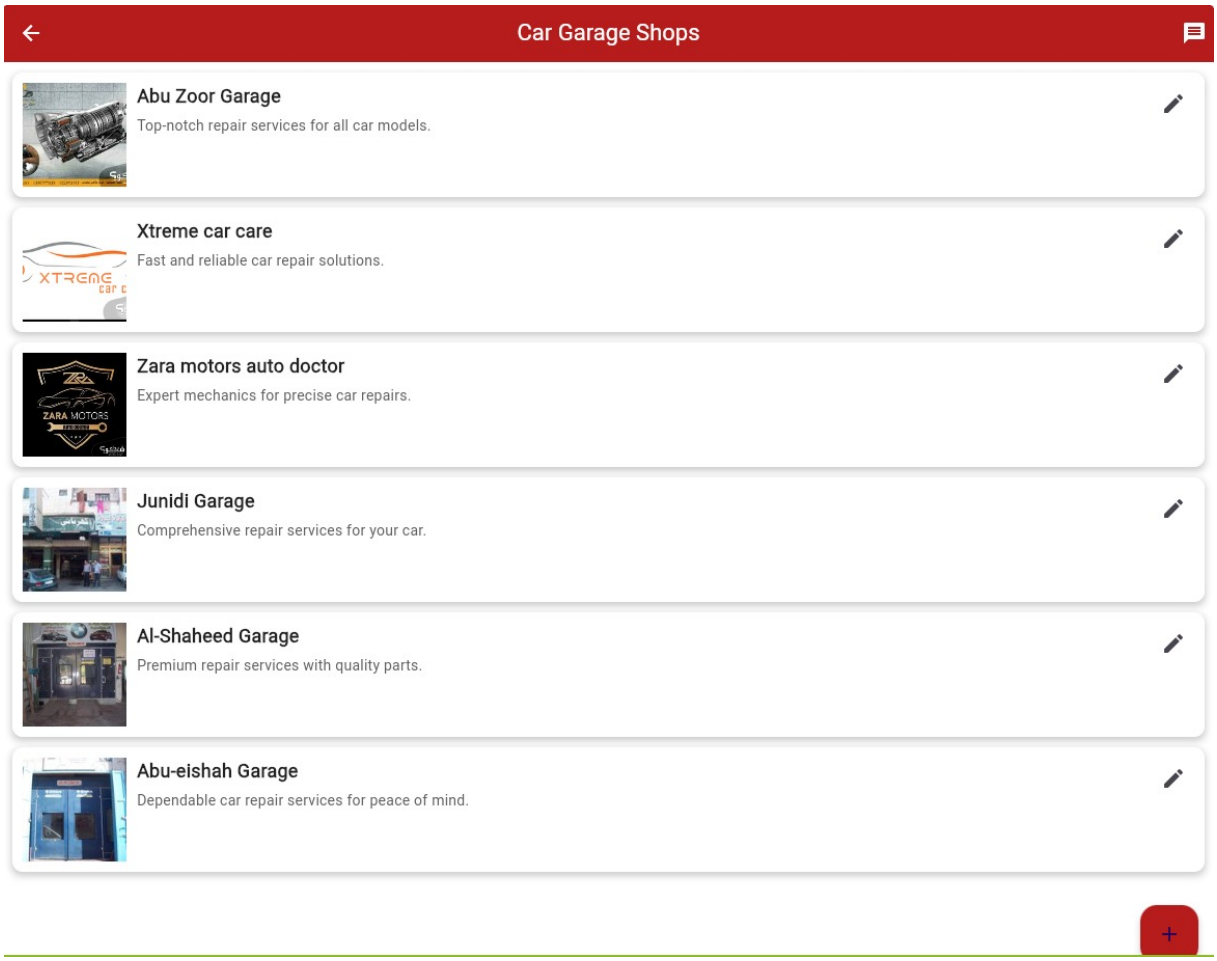


Figure 4.19: Admin editing garage shops

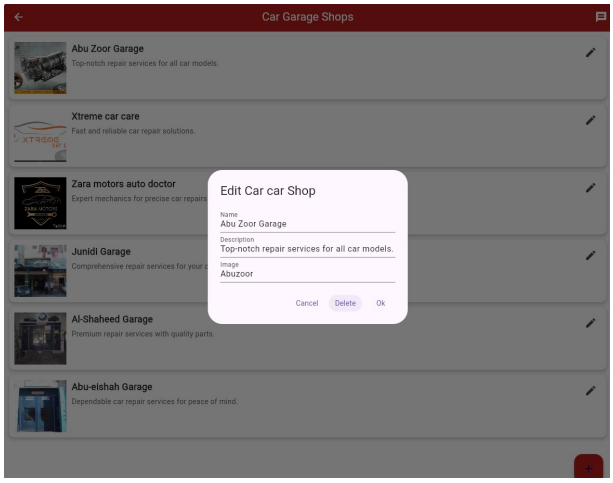


Figure 4.20: Admin Deleting a garage

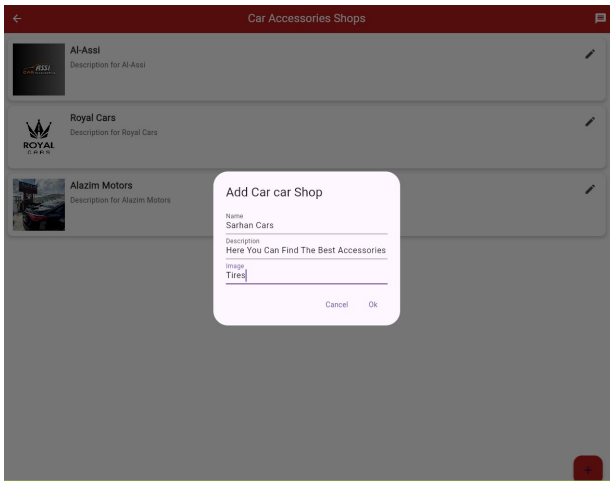


Figure 4.21: admin adding a car accessories shop

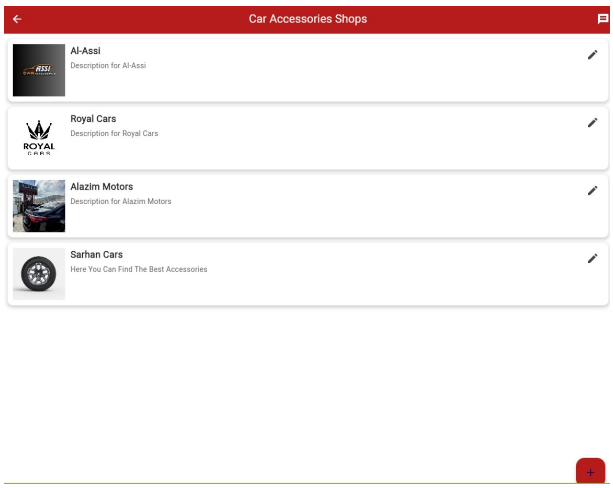


Figure 4.22: Car Accessories shop added successfully!

After the Admin controlling each service shop , we will track orders and see activities on the admin page , where each client or merchant doing an activity it's shown for the admin.

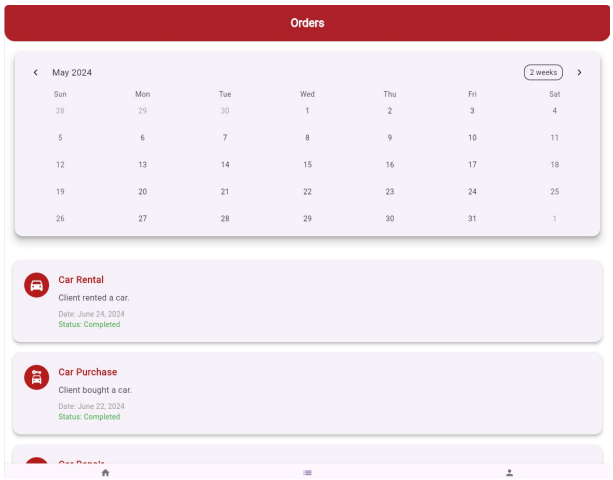


Figure 4.23: Orders in admin page

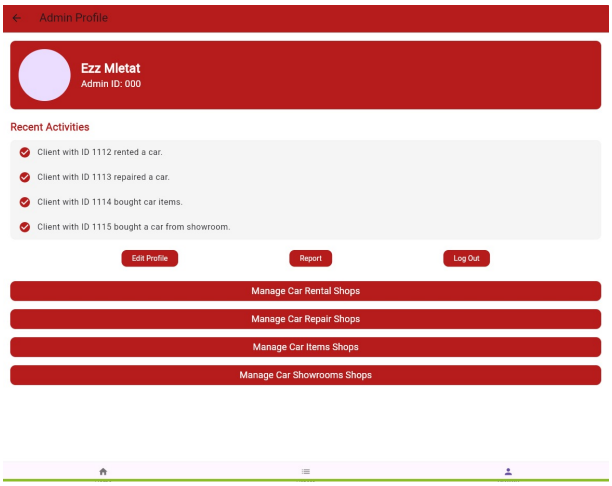
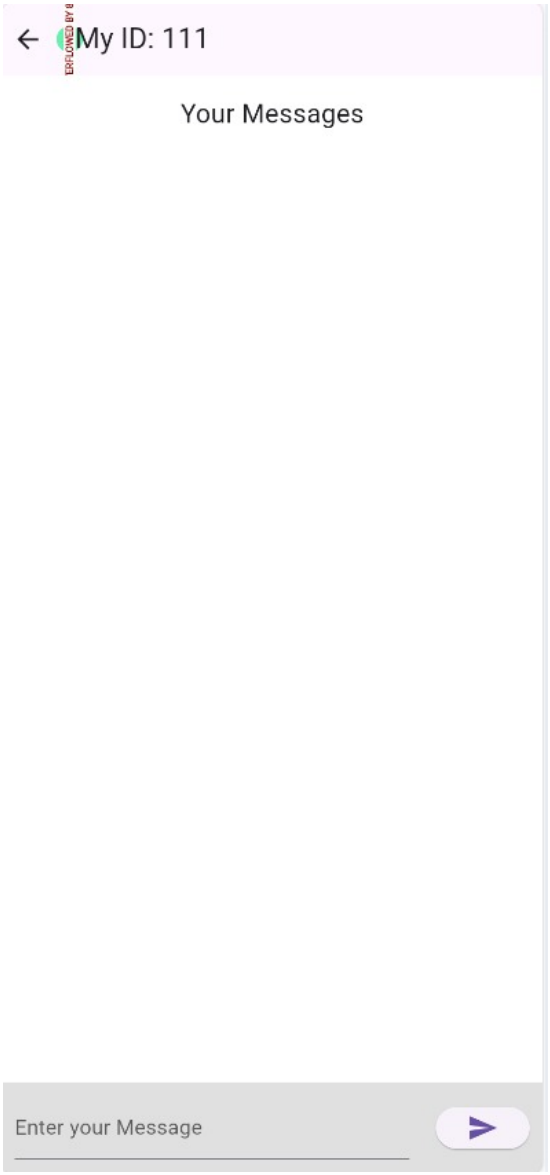


Figure 4.24: Activities

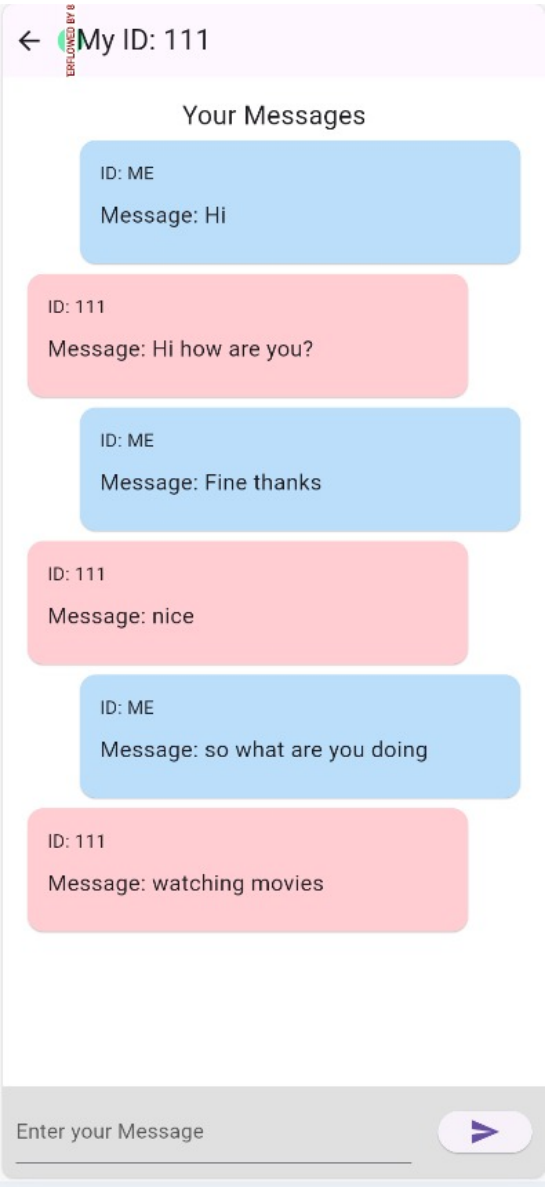
4.4.6 Chatting between users

Users can chat with each other whatever their type , lets take some examples

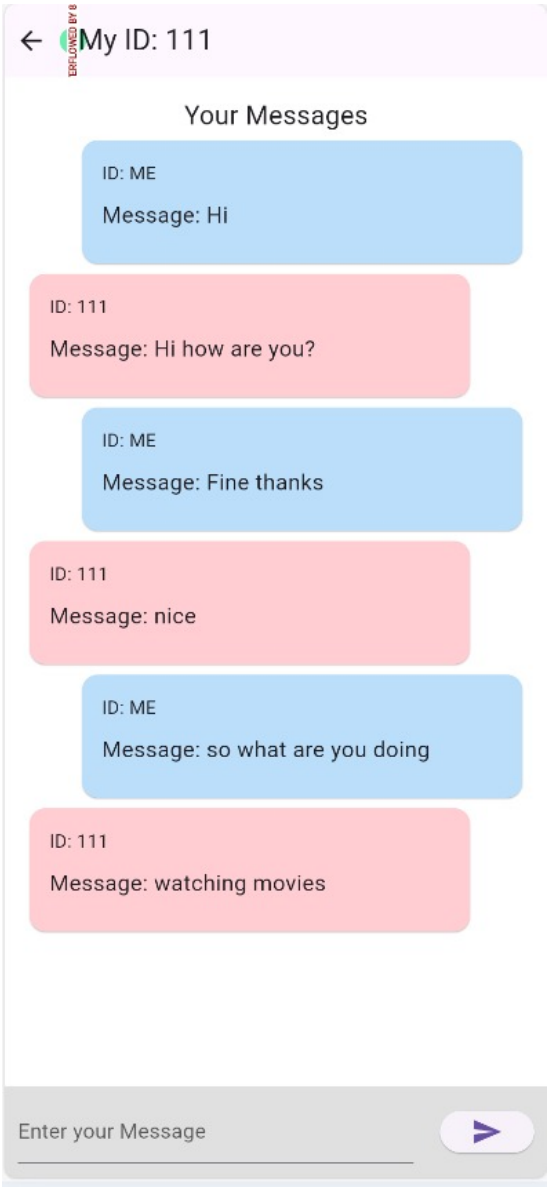


(a) Chat from side 1

(b) Chat from side 2



(a) Chat from side 1



(b) Chat from side 2

Chapter 5

Conclusion

Contents

5.1 Conclusion

In conclusion, CarHub has made significant strides in improving access to automotive services. The map routes, comprehensive database, and real-time chat have enhanced user satisfaction and engagement.

Users benefit from easy access to information on nearby garages, dealerships, rental offices, and parts shops. The platform's detailed maintenance and repair information helps users make informed decisions about their vehicles , and also made a big benefit for the merchants to market their business and help people to find the exact product or service searching for .

CarHub's features for listing and selling cars simplify the automotive sales process, making it more efficient. The application sets a new standard for convenience and efficiency in the automotive service industry, greatly benefiting users and service providers .

Palestine needed an application like this , and CarHub was the one to help this sector rise .

5.2 Future work

Future developments for the CarHub application will focus on expanding its features and improving user experience. Key areas for enhancement include:

Expanded Service Coverage: Increasing the number of listed garages, dealerships, rental offices, and parts shops to cover more regions and provide users with more options.

Service Reminders: Implementing a reminder system for users to schedule regular car maintenance and services.

User Training Videos: Adding tutorial videos to help users understand how to use the app's features effectively.

These enhancements aim to further streamline automotive service access, improve user satisfaction, and solidify CarHub's position as a leading platform in the industry.

Chapter 6

Results And Discussion

Contents

6.1 Results

The CarHub application has shown significant improvements in user interaction with automotive services. The map routes feature has made it easier for users to find directions to nearby garages, dealerships, rental offices, and parts shops. The comprehensive database has improved service quality and user trust by providing access to detailed customer and car information. Users are more satisfied and make better decisions with this information. The real-time chat feature has enhanced customer support by allowing quick communication between merchants and customers. This has led to faster issue resolution and higher user satisfaction. The filtering system allows users to sort services based on various criteria, making it easier to find the most relevant options. The user feedback system has provided valuable insights, enabling continuous improvements and maintaining high service quality. Additionally, CarHub offers detailed information on car maintenance and repairs, helping users keep their vehicles in good condition. The platform also simplifies the process of listing and selling cars, making it more efficient for users.

6.2 Discussion

The CarHub application effectively addresses the challenges in accessing automotive services. The map routes feature improves navigation to essential services, making it more convenient for users.. Real-time chat functionality significantly boosts customer service by enabling instant communication and quick problem-solving, which increases user satisfaction and loyalty. Detailed maintenance and repair information empower users to take better care of their vehicles. The streamlined process for listing and selling cars makes the platform user-friendly and efficient for all transactions. The filtering system allows users to easily find the services they need, improving the overall user experience. The feedback system has been crucial for gathering user opinions and making necessary adjustments to ensure service quality. CarHub's innovative features and simple design have set a new standard in the automotive service industry, offering considerable benefits to both users and service providers. in the end , CarHub has made this industry in Palestine much easier since we don't hence such applications , and made it much more time , effort , and money saving in this industry .