

24/7 Automobile Assistance



Group Members

Shehryar Ahmed Satti (01-131172-025)

Atta Ur Rehman Tariq (01-131172-035)

Supervisor: Dr Awais Majeed

A Final Year Project submitted to the Department of Software Engineering,
Faculty of Engineering Sciences, Bahria University, Islamabad in the partial
fulfillment for the award of degree in Bachelor of Software Engineering

July 2021

PROJECT COMPLETION CERTIFICATE

Student Name:Shehryar Ahmed Satti

Enrolment No:01-131172-025

Student Name:Atta Ur Rehman Tariq

Enrolment No:01-131172-035

Program of Study: Bachelor of Software Engineering

Project Title: 24/7 Automobile Assistance

It is to certify that the above students' project has been completed to my satisfaction and to my belief, its standard is appropriate for submission for evaluation. I have also conducted plagiarism test of this thesis using HEC prescribed software and found similarity index at _____ that is within the permissible limit set by the HEC. I have also found the thesis in a format recognized by the department.

Supervisor's Signature: _____

Date: _____ Name: _____

CERTIFICATE OF ORIGINALITY

This is to certify that the intellectual contents of the project **24/7 Automobile Assistance** are the product of my/our own work except, as cited properly and accurately in the acknowledgements and references, the material taken from such sources as research journals, books, internet, etc. solely to support, elaborate, compare, extend and/or implement the earlier work. Further, this work has not been submitted by me/us previously for any degree, nor it shall be submitted by me/us in the future for obtaining any degree from this University, or any other university or institution. The incorrectness of this information, if proved at any stage, shall authorities the University to cancel my/our degree.

Name of the Student: Atta Ur Rehman Tariq

Signature: _____ Date: _____

Name of the Student: Shehryar Ahmed Satti

Signature: _____ Date: _____

CERTIFICATE FOR APPROVAL OF PROJECT

ABSTRACT

Number of vehicles on the road have increased significantly in the recent decade. We observe many cases of vehicle breakdown around us daily on different spots. Such breakdowns can happen any time and immediate help might not be available in some cases. Moreover, commuters in the vehicle as well as the drivers also require additional services like tyre shops, rest areas, petrol stations and roadside restaurants. Our app can solve the problems of the drivers and commuters by providing them information about nearby Mechanics, workshops, rest areas, petrol stations and other allied facilities. Mechanics are registered in our app and Drivers will be able to see list of available mechanics and workshop nearby. Driver can contact the selected mechanic and discuss about his/her Automobile Problem before and Mechanic arrives at his/her location using Driver's location available on map. Another salient feature of our app is self-guide (methods to fix problem offline). Drivers are able to use this feature in their emergency situation to fix their automobile. The system also provides the feature of advance booking, in case a driver does not want to go to workshop before confirming his/her time slot, he/she just need to book appointment with a selected service provider from home, and after confirmation of the selected time slot, he/she can move to workshop at the given time, so drivers do not need to wait for their turns on workshops. The service providers will have the other module of the app which will notify them about the service requests from the drivers and they will respond to those requests. After service request is sent to the selected workshop, if the Service Provider accepts request user will receive a notification of booking done and will receive a booking cancellation notification if service provider rejected it.

Keywords: Automobile Assistance, self-guide, 24-7 assistance, Vehicle breakdown,

Dedication

We dedicate our work to

Our parents, teachers, friends and supervisor.

Acknowledgments

First of all, we thank **Almighty Allah** for all the blessings He showered on us. We thank our parents, siblings and all family members because without their support it would never be possible. Our project supervisor, project coordinator, teachers, faculty and other university staff supported us whenever we need any of their support. We acknowledge all of their efforts and support and thank them all from the core of our heart. Specially we are very thankful to our respected supervisor Sir Dr. Awais Majeed who guided us, checked our progress, provided us feedback and motivated us to complete our project. Our project coordinator respected ma'am Sadaf Farhan was very kind to us. she dealt with us very politely and helped a lot to meet deadlines. Our special thanks to her for her coordination and support. It will not be a justified act to ignore the support and kind advices that we got from respected sir Bilal Asharf Awan and ma'am Iqra Bashart. We thank a lot to both of them too. We acknowledge all the support and contributions of friends specially Ahba'ab group, who backed us up during the good and bad times of this project.

Table of Contents

Project Completion Certificate	1
Certificate Of Originality	2
Certificate For Approval Of Project	3
Abstract	4
Dedication	5
Acknowledgments	6
Table of Contents	7
List of Figures	11
List of Tables	13
Chapter 1	14
Introduction	14
1.1. Purpose	14
1.2. Motivation	14
1.3. Problem statement or research questions	14
1.4. Objectives.....	14
1.5. Main contributions	15
1.5.1. Existing System	15
1.5.2. Proposed System.....	16
1.6. Report organisation	16
Chapter 2	18
Background Study/Literature Review	18
2.1. A Car Breakdown Service Station Locator System	18
2.2. Motorway Breakdown Response Service.....	19
2.3. On Road Vehicle Breakdown Assistance.....	19
2.4. 24-Hour Roadside Assistance	20
2.5. On Road Vehicle Breakdown Assistance Finder	20
2.6. Vehicle Breakdown Assistance	21
2.7. Online Auto Mechanic Finder Project in PHP or ASP.NET	21
2.8. HelpMe.....	21
Chapter 3	23
System Requirements	23
3.1. Functional Requirements.....	23
3.1.1. Functional requirements for driver	23
3.1.2. Functional requirements for service provider	24
3.2. Use case diagram.....	24

3.2.1. System level use case for 24/7 AA	24
3.2.2. Use case diagram for driver	25
3.2.3. Use case diagram for service provider.....	26
3.2.4. Use case diagram for Manage Automobile group	27
3.2.5. Use case diagram for find resources group.....	28
3.2.6. Use case diagram for Manage Accounts group	28
3.3. System features	29
3.3.1. User registration	29
3.3.2. Automobile Registration.....	32
3.3.3. Manage Registered automobiles.....	32
3.3.4. Find Resources/service providers	34
3.3.5. Select service provider.....	35
3.3.6. Contact service provider:.....	36
3.3.7. Self-Guide.....	37
3.3.8. Normal Maintenance	38
3.3.9. Service provider management	39
3.4. Interface Requirements.....	41
3.4.1. User Interfaces.....	41
3.4.2. Hardware Interfaces.....	42
3.4.3. Software Interfaces	42
3.4.4. Communications Interfaces	42
3.5. Database Requirements	42
3.6. Non-Functional Requirements	42
3.6.1. Performance Requirements.....	43
3.6.2. Safety Requirements.....	43
3.6.3. Security Requirements.....	43
3.6.4. Software Quality Attributes.....	43
3.7. Project Feasibility.....	45
3.7.1. Technical Feasibility.....	45
3.7.2. Operational Feasibility.....	45
3.7.3. Legal and Ethical Feasibility	45
Chapter 4	46
System Design.....	46
4.1. Design approach.....	46
4.2. System Architecture	46
4.2.1. Presentation tier:	46
4.2.2. Logic tier:	46

4.2.3. Data tier:	46
4.3. Logical Design	47
4.3.1. Class diagram	47
4.4. Dynamic view	47
4.4.1. Use case Diagram	48
4.4.2. Sequence Diagram of Use cases	48
4.5. Component design.....	55
4.5.1. Deployment diagram	55
4.5.2. Work Breakdown Structure	55
4.5.3. Work Breakdown Dictionary.....	56
4.5.4. Network Model.....	57
4.6. Data model	57
4.7. Interface Design	59
4.7.1. System Prototype.....	59
4.7.2. User Interface Design	63
Chapter 5	70
System Implementation	70
5.1. Strategy.....	70
5.2. Tools Used.....	70
5.3. System Architecture	70
5.4. Steps in Implementation.....	70
5.4.1. Requirement Gathering.....	70
5.4.2. Software Requirements Specification (SRS) development	71
5.4.3. Software Design Specifications (SDS) development	71
5.4.4. System prototypes and Front end development.....	71
5.4.5. Backend Development.....	71
5.5. Problems faced during development	72
5.5.1. Selection of Framework.....	72
5.5.2. React Native Installation.....	72
5.5.3. Database Connectivity and Authentication:	72
5.6. Methodologies	72
Chapter 6	73
System Testing & Evaluation.....	73
6.1. Test Strategy.....	73
6.2. Unit Testing.....	73
6.3. Component Testing	73
6.4. Integration Testing	73

6.5. System Testing	74
6.6. Test cases.....	74
6.6.1. Test Case for Signup.....	74
6.6.2. Test Case for Login	75
6.6.3. Test Case for Register Automobile.....	76
6.6.4. Test Case for View Registered Vehicle.....	77
6.6.5. Test Case for Edit vehicle details.	78
6.6.6. Test Case for Delete Vehicle.	79
6.6.7. Test Case for find resources.	80
6.6.8. Test Case for contact service provider.....	81
6.6.9. Test Case for Open Self-Guide.....	82
6.6.10. Test Case for Booking appointment for maintenance	83
6.6.11. Test Case for Check maintenance status.....	84
6.6.12. Test Case for Register workshop.....	85
6.6.13. Test Case for Accept Request.....	86
6.6.14. Test Case for Update maintenance status	87
Chapter 7	88
Conclusion	88
7.1. Contributions	88
7.2. Reflections.....	88
7.3. Future work	88
References.....	90

List of Figures

<i>Figure 3.1 Use Case Diagram for 24/7 Automobile Assistance</i>	24
<i>Figure 3.2 use case diagram (Driver's Module)</i>	25
<i>Figure 3.3 Use case diagram (Service Provider's Module)</i>	26
<i>Figure 3.4 Use case diagram (Manage Registered Automobiles)</i>	27
<i>Figure 3.5 Use case diagram (Find Resources)</i>	28
<i>Figure 3.6 Use case diagram (Manage Accounts)</i>	28
<i>Figure 4.1 System Architecture of 24/7 Automobile Assistance</i>	46
<i>Figure 4.2 Logical view of the system</i>	47
<i>Figure 4.3 Use case diagram of the system</i>	48
<i>Figure 4.4 Sequence Diagram for Sign Up</i>	49
<i>Figure 4.5 Sequence Diagram for Log In</i>	49
<i>Figure 4.6 Sequence diagram for Forget Password</i>	50
<i>Figure 4.7 Sequence Diagram for Register Automobile</i>	50
<i>Figure 4.8 Sequence Diagram for View Registered Automobiles</i>	51
<i>Figure 4.9 Sequence Diagram for Edit Vehicle details</i>	51
<i>Figure 4.10 Sequence Diagram for Find Resources</i>	52
<i>Figure 4.11 Sequence Diagram for Delete Vehicle</i>	52
<i>Figure 4.12 Sequence Diagram for Select Service Provider</i>	53
<i>Figure 4.13 Sequence Diagram for Contact Service Provider</i>	53
<i>Figure 4.14 Sequence Diagram for Book Appointment</i>	54
<i>Figure 4.15 Sequence Diagram for Check Maintenance Status</i>	54
<i>Figure 4.16 Sequence Diagram for Register Workshop</i>	54
<i>Figure 4.17 Deployment View of the System</i>	55
<i>Figure 4.18 Work Breakdown Structure</i>	55
<i>Figure 4.19 CPM Network diagram</i>	57
<i>Figure 4.20 Data model of the system (1 of 4)</i>	57
<i>Figure 4.21 Data model of the system (2 of 4)</i>	58
<i>Figure 4.22 Data model of the system (3 of 4)</i>	58
<i>Figure 4.23 Data Model of the System (4 of 4)</i>	59
<i>Figure 4.24 System Prototype (Main Screen)</i>	60
<i>Figure 4.25 System Prototype (Login Screen)</i>	60
<i>Figure 4.26 System Prototype (Signup Screen)</i>	61
<i>Figure 4.27 System Prototype (Reset Password)</i>	61
<i>Figure 4.28 System Prototype (Driver's Main Screen)</i>	62
<i>Figure 4.29 System Prototype (Service Provider's Main Screen)</i>	62

<i>Figure 4.30 UI of Login Screen</i>	<i>63</i>
<i>Figure 4.31 UI of Driver's Main Screen</i>	<i>64</i>
<i>Figure 4.32 Automobile Registration UI</i>	<i>65</i>
<i>Figure 4.33 Manage Registered Automobiles UI</i>	<i>65</i>
<i>Figure 4.34 Find Resources/Service Providers UI.....</i>	<i>66</i>
<i>Figure 4.35 Select Service Provider UI.....</i>	<i>66</i>
<i>Figure 4.36 Book Appointment UI.....</i>	<i>67</i>
<i>Figure 4.37 Check Request UI.....</i>	<i>67</i>
<i>Figure 4.38 UI of Service Provider's Main Screen.....</i>	<i>68</i>
<i>Figure 4.39 Register Workshop UI.....</i>	<i>69</i>
<i>Figure 4.40 Manage Requests UI.....</i>	<i>69</i>

List of Tables

<i>Table 1: Use case description (Sign up)</i>	29
<i>Table 2: : Use case description (Log in)</i>	30
<i>Table 3: Use case description (Forget Password)</i>	31
<i>Table 4: Use case description (Register Automobile)</i>	32
<i>Table 5: Use case description (View registered vehicles)</i>	32
<i>Table 6: Use case description (Edit vehicle details)</i>	33
<i>Table 7: Use case description (Delete Vehicle)</i>	33
<i>Table 8: Use case description (Find resources)</i>	34
<i>Table 9: Use case description (Select Service Provider)</i>	35
<i>Table 10: Use case description (Contact Service Provider)</i>	36
<i>Table 11: Use case description (open self-guide)</i>	37
<i>Table 12: Use case description (Book appointment for maintenance)</i>	38
<i>Table 13: Use case description (Check maintenance status)</i>	39
<i>Table 14: Use case description (Register workshop)</i>	39
<i>Table 15: Use case description (Accept request)</i>	40
<i>Table 16: Use case description (Respond to customer)</i>	40
<i>Table 17: Use case description (Update maintenance status)</i>	41
<i>Table 18: Work breakdown structure</i>	56
<i>Table 19: Test case (Sign up)</i>	74
<i>Table 20: Test case (Log in)</i>	75
<i>Table 21: Test case (Register Automobile)</i>	76
<i>Table 22: Test case (View registered vehicles)</i>	77
<i>Table 23: Test case (Edit vehicle details)</i>	78
<i>Table 24: Test case (Delete vehicle)</i>	79
<i>Table 25: Test case (Find resources)</i>	80
<i>Table 26: Test case (Contact service provider)</i>	81
<i>Table 27: Test case (Open self-guide)</i>	82
<i>Table 28: Test case (Book appointment for maintenance)</i>	83
<i>Table 29: Test case (Check maintenance status)</i>	84
<i>Table 30: Test case (Register workshop)</i>	85
<i>Table 31: Test case (Accept request)</i>	86
<i>Table 32: Test case (Update maintenance status)</i>	87

Chapter 1

Introduction

24/7 Automobile Assistance system is a mobile application that will provide assistance to automobile users for maintenance process and link them to the service providers. The document will contain the system's problem statement, background knowledge, system requirements, system architecture and design, implementation details, evaluation and testing. Report organization describes the more detailed classification of the document on the basis of chapters.

1.1. Purpose

The purpose for developing 24/7 automobile assistance is to connect the automobile users and service providers. It is designed to provide assistance to the drivers facing vehicle breakdown or looking for a vehicle maintenance.

This document is to render a comprehensive overview of 24/7 Automobile Assistance System. The document will enlighten the objectives, background knowledge and features of the system. It will also document primary requirements of the system, its design and evaluation & testing details.

1.2. Motivation

Today is the era of technology and technology is providing ease in each field of life. The motivation behind the system is to automatize the auto mechanics industry. There should be a common platform to connect Drivers and service providers. The idea behind the system is to develop a system that can connect automobile users and auto mechanics (service providers).

1.3. Problem statement or research questions

Drivers and automobile users face Various issues in daily life during driving. They face

- On-Road Automobile Breakdown issues
- Daily Maintenance Issues

1.4. Objectives

The goal of the 24/7 AA is to provide assistance to its users. Our objective is to

- build a common platform that will connect Drivers/automobile users and Mechanics/service providers.
- provide help to user in case of emergency.
- facilitate user in everyday maintenance process.

The system will provide following features;

- **Find resources**
- **Select service provider**
- **Contact service provider**
- **Self-guide**
- **Book appointment for maintenance**
- **Check maintenance status**
- **User registration**
- **Automobile registration**
- **Manage registered automobiles**

1.5. Main contributions

The system is developed for automobile users/drivers and service providers/auto mechanics. Both of the user types can use the system and get facilitated through its features. The system will assist automobile users in their maintenance process and will provide the service providers to increase their customers circle by exposing to automobile users through the app.

1.5.1. Existing System

There are very less existing systems which provide services in the desired field. The systems that are designed for mechanics or automobile users are either limited to less functionality or single purpose systems, which are not meeting the requirements of the users. There are some major auto mechanic franchises who have their own systems to assist their customers but those systems are their intellectual properties and they provide the features to connect with them only. Mainly the automobile mechanic industry is working on manual systems and technology is not playing its part to enhance the productivity efficiency and portability of this field.

1.5.2. Proposed System

As discussed in the previous paragraph that the existing systems either provide limited functionality or are the intellectual property of the franchises so the user have to use a lot of systems to avail complete functionality and connect with different auto mechanic franchises. The proposed system is designed to provide these functionalities to the users under a single umbrella. User can see all the auto mechanics and franchises and connect to any of them. Moreover it will also provide offline guide (limited functionality) to its users in case they have no access to internet. The system will provide the following silent features;

- **Find resources:** Drivers will be able to find nearby resources i.e. (workshops, tire shops, petrol stations, service stations, rest areas etc.)
- **Select service provider**
- **Contact service provider:** The system will allow the driver to contact with a selected service provider and discuss his/her problem.
- **Track service provider:** The driver will be able to track the position of selected service provider by using GPS.
- **Self-guide:** This feature will provide offline help to the drivers. The system will assist them in fixing their problem by providing step by step solution.
- **Book appointment for maintenance:** The driver will be able to reach a mechanic and book appointment for his/her vehicle maintenance from home.
- **Check maintenance status:** The driver can check the maintenance status of vehicle after he/she has dropped the car for maintenance.
- **User registration**
- **Automobile registration**
- **Manage registered automobiles**

1.6. Report organisation

Chapter 1 is the introduction of the project. It describes the purpose of the project, motivation behind doing the project, problem statement, goals & objectives to be achieved by this system and our contribution in this project.

Chapter 2 is about the Background knowledge and literature review of the system. Existing system that can relate to 24/7 Automobile Assistance System are discussed and reviewed.

Chapter 3 talks about the system requirements. It covers the system level use case, functional, non-functional and other requirements e.g. interface & database requirements. It also includes system feasibility.

Chapter 4 discusses system modelling and designs. It includes system architecture & design approach, system views e.g. logical & dynamic view, user interface design and prototypes.

Chapter 5 is about the implementation following chapter 6 which is about testing and evaluation. It includes testing strategy & approaches, testing and testcases.

The last chapter is about conclusion of the project. It also writes about contributions, reflections and suggests the modifications that can be made in future.

Chapter 2

Background Study

Automobile industry is one of the major business industries of Pakistan. Almost 100000 vehicles are produced each year in Pakistan and the total volume of sales ranges up to 20 billion rupees. As the no. of automobiles are increasing each day, the need for automobile maintenance is also increasing and Automobile maintenance has shaped to become a separate industry.

Automobile maintenance or simply Auto mechanic industry despite being a huge industry is still not up to the mark of modern technologies. There are very less systems that connect the stakeholders of this industry. The systems that are there related to this industry are providing limited functionality. In case of on road vehicle breakdown there are systems that provide assistance to users but no such renowned system is there in Pakistan. However when it comes to normal day to day maintenance process, there are some major automobile workshop chains that provide such services but those systems are limited to only their own service providers. Our system is an open source system where anyone can register his/her workshop and start providing services to the users to increase his/her profitability.

Some of the existing systems from different places were analysed and a system that contains a complete functionality after merging those systems was proposed as 24/7 AA System. Related systems are analysed and discussed in the following lines.

2.1. A Car Breakdown Service Station Locator System

This is a quit Important and sensitive issue especially for drivers that what they can do when their car is suffering from an accident or having some mechanical problem. In order to solve this problem they should know that there are some companies which are known as car repair service providers (CRSP). This system actually helps such kind of people who have been suffered from road accident. One problem is being faced by these service providers is to trace the exact location of car. In order to solve this issue, these companies and services providers developed on location tracking system based on different analysis and experiments. This is also known as service station locator system. This system actually connects CRSP and public with each

other. After the successful launching of this system the drivers will be able to get exact location of their car, accident place, distance and time from service provider. Basically this system talks about the on road vehicle breakdown. It provides the solutions and assistance in case of accidents but does not talk about normal maintenance process [1].

2.2. Motorway Breakdown Response Service

If you are driving on motorway then you should know that you are having a very good service regarding vehicles which is known as motorway breakdown response service. Sometimes when we face some mechanical issues during our journey on motorway, we need someone to assist us and this system is there to respond you and assist you. Under many circumstances, this is a critical time for all when someone is facing an on road breakdown. We all know that on motorway speed of vehicles is very fast hence there is always a risk of accident in such cases. Moreover due to fast speed people cannot stop to help. At this time we must know that we are having a good service which is known as motorway break down response service. We must need to work this service in order to get our vehicles on side of the road so we can get rid of danger of hitting by another vehicle from backsides. This service is available for all vehicles under 2.5 tones. This system also provides limited maintenance services and assistance to the users.

This is a good system that provides help in case of emergency specially in case of accident but this is a manual system in which you have to call the motorway response unit and tell about their problem. There is no such automated system that provide the details about the services they provide, distance of the unit from vehicle etc. another problem with this system is that it is only for the drivers travelling on motorway as well as this system is also a licensed system where none of the uses other than Motorway Response units can provide the services [2].

2.3. On Road Vehicle Breakdown Assistance

People who are suffering mechanical issues with their vehicles now have a facility of on road vehicle breakdown assistance (ORVBA). This facility is very helpful especially when people are having vehicle breakdown issues in the remote location. A good thing in ORVBA is that it is registered public and are connected with some specific mechanics which is trustworthy. In ORVBA all mechanics are legally

licensed and approved so people do not need to worry that the mechanics are being monitored by ORVBA system in order to not charging any extra service fee from customers. The project is very beneficial and helpful for those people especially who are new in any destination and are not having much knowledge about mechanics and experts in that area. This is very beneficial in remote areas where there are no mechanics shops in nearby location.

On Road Vehicle Breakdown Assistance is a system that provides maximum of the functionality that our proposed system will provide. However this system covers only on road breakdown cases as suggested by its name. It does not cover the normal maintenance module. Moreover the system is a paid app where the service providers need to pay to the system in order to register their workshop as well as drivers also need to pay to use the system [3].

2.4. 24-Hour Roadside Assistance

There are different roadsides services available for riders on the road during any problem. At depend on drivers that which facility reliable for them . One of these facility is 24 _hour roadside assistance. This service is very useful if customers one willing to avail this service. This is very fast and reliable and trackable service. This is useful in different problems i-e need a tow ,jump-start, help with a flat tire or lockout assistance. This service is available during 24 hours and seven days. There are basically three types of services for customer. There are options available for customers to avail annual membership plan and also pay per use service. One of the interesting thing it is that if you have car insurance then you can add this service your policy of insurance. So we can say that no matter you are in a hard area on a side area where no service are available on road then you should have peace of mind that you have of reliable and good service of 24-hour Road side assistance [4].

2.5. On Road Vehicle Breakdown Assistance Finder

It is too difficult to find the mechanic wherever you are traveling. Online system help you to find out the mechanics quickly. It is also safe and time saving. Admin send you information about mechanic and the user also sends his/her information to the admin which is shared with the mechanic later. After work has completed you can send the feedback. Here these points are important about the system. It is a simple and time

saving system. Mechanic helps you quickly by registration but requires an active network [5].

2.6. Vehicle Breakdown Assistance

The presented application aids in finding mechanics easily and effortlessly. It is hard to find mechanics in nearby area when a person is travelling in an unknown area. This proposed system will help to overcome this issue by providing a mechanic and his details in just one click. Here the locator allows you to search mechanics around different locations. Admin is allowed to access and manage the mechanic details. This online mechanic locator reduces the efforts for a person needing a mechanic and can easily find the mechanics from different locations. It is time and cost effective. The main objective is to provide a better service and to make the process easily and finally appointing a mechanic quickly. Presented system is used by three entities namely, Admin, Mechanic and User. A mechanic can perform task such as viewing request received from users and can also send feedback to the admin. User can send a request and can appoint a mechanic on respective date-time [6].

2.7. Online Auto Mechanic Finder Project in PHP or ASP.NET

This proposed system will provide an interface through which users can easily find automobile mechanics from various areas. Users can register and can search for mechanics based on the mechanic's and the user's location and see the workshop address details. To send a request to the mechanic, the user is first required to register. Registered users can also post feedback regarding mechanics. This feedback should be visible to all the users. A mechanic can view the requests sent and also the feedback posted by the users regarding him [7].

2.8. HelpMe

HelpMe application is to assist a vehicle-breakdown on the road. By using this application, the user will be able to search nearby mechanics or a spare-part shop to their location. A driver having an android phone can use this vehicle breakdown assistance application "HelpMe", and user will be able to find a suitable mechanic at that location within few minutes. The user have to enter the information of the place where breakdown has occurred and "HelpMe" will automatically search for nearby Car repairing service provider who have registered on the system. A chat platform is

also added in the system, to discuss the type of breakdown and also exchange ideas about the vehicle's breakdown. "HelpMe" is designed in such a way that it captures user's location from mobile network tower and then connect to firebase database and check nearby mechanics. This data is the sent back to the user. There is an admin to add the mechanic and can also see the user details registered with the application [8].

Chapter 3

System Requirements

System requirements are discussed, analysed, and documented into the chapter.

3.1. Functional Requirements

Functional requirements are the functions that the system will allow its users to perform. Both of the users type i.e. Drivers and service providers will have their own functionalities to perform so their functional requirements are written separately.

3.1.1. Functional requirements for driver

Driver will be able to;

- Create his/her account by providing his personal information.
- Login to the system by providing his/her valid credentials.
- Register his/her vehicle to the system after he/she is logged in.
- Manage registered automobiles after he/she is logged in to the system. He/she can view, edit and delete the details of registered vehicles.
- Find nearby resources by selecting a resource type i.e. mechanic, petrol station, tyre shop and rest area. He/she needs to login to the system first.
- Select a service provider. After viewing his/her profile and list of services he/she provides, the user can book appointment.
- Contact with the selected service provider. The user needs to first login to the system and select a service provider to contact him/her.
- Book an appointment with a service provider of his/her on choice after logging into the system.
- Check the maintenance status after booking is confirmed
- Use the feature of self-guide in case of unavailability of internet.
- Manage his/her account. He/she can edit his account information, delete his/her account after logging into the system.

3.1.2. Functional requirements for service provider

The system will allow service provider to;

- Create his/her account by providing his personal information.
- Login to the system by providing his/her valid credentials.
- Register his/her workshop to the system by entering required information of the workshop after logging into the system.
- Accept/reject the requests coming from the drivers.
- Respond to customer queries.
- Update maintenance status of the vehicles that are in the maintenance phase. Service provider must have accepted the request of a vehicle to update its status.
- Manage his/her account. He/she can edit his account information, delete his/her account after logging into the system.

3.2. Use case diagram

3.2.1. System level use case for 24/7 AA

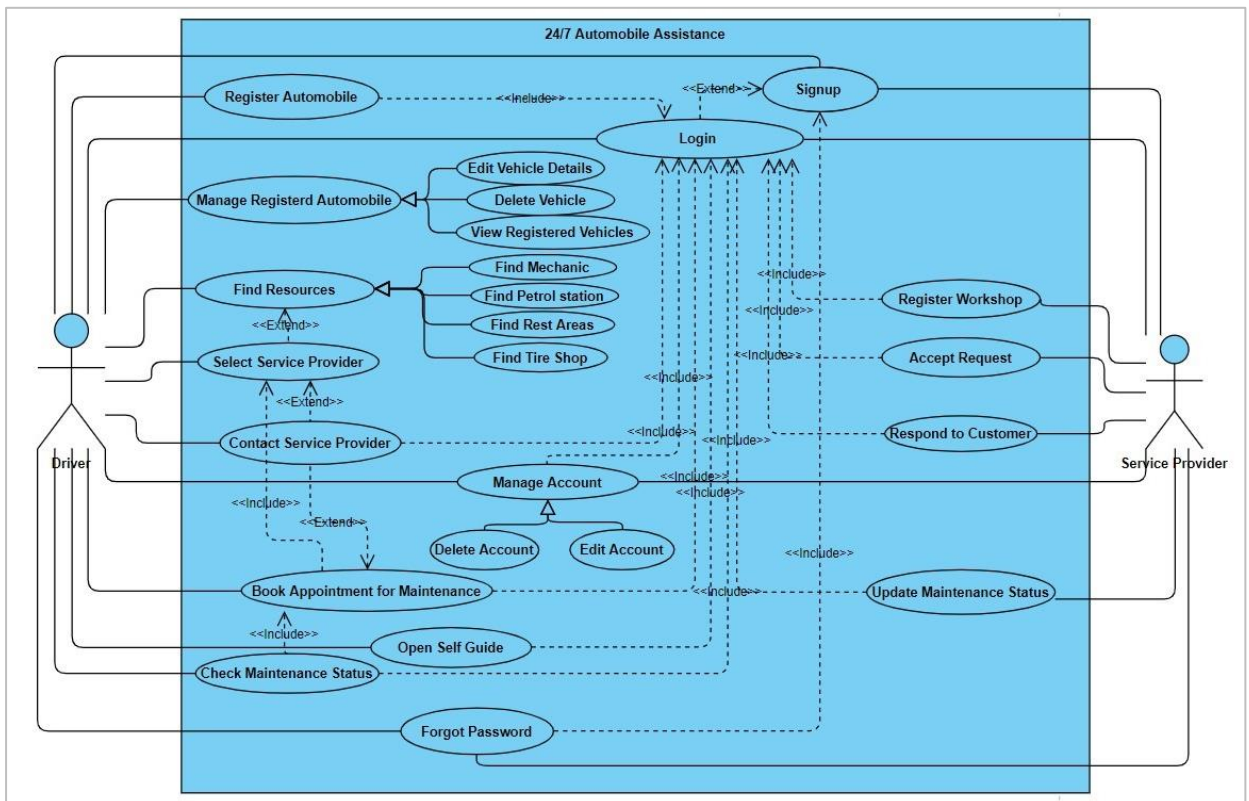


Figure 3.1 Use Case Diagram for 24/7 Automobile Assistance

Use case diagram of 24/7 Automobile Assistance is used to show the functionalities that the system will contain and the details of these use cases is provided in coming paragraphs as Use Case Descriptions.

3.2.2. Use case diagram for driver

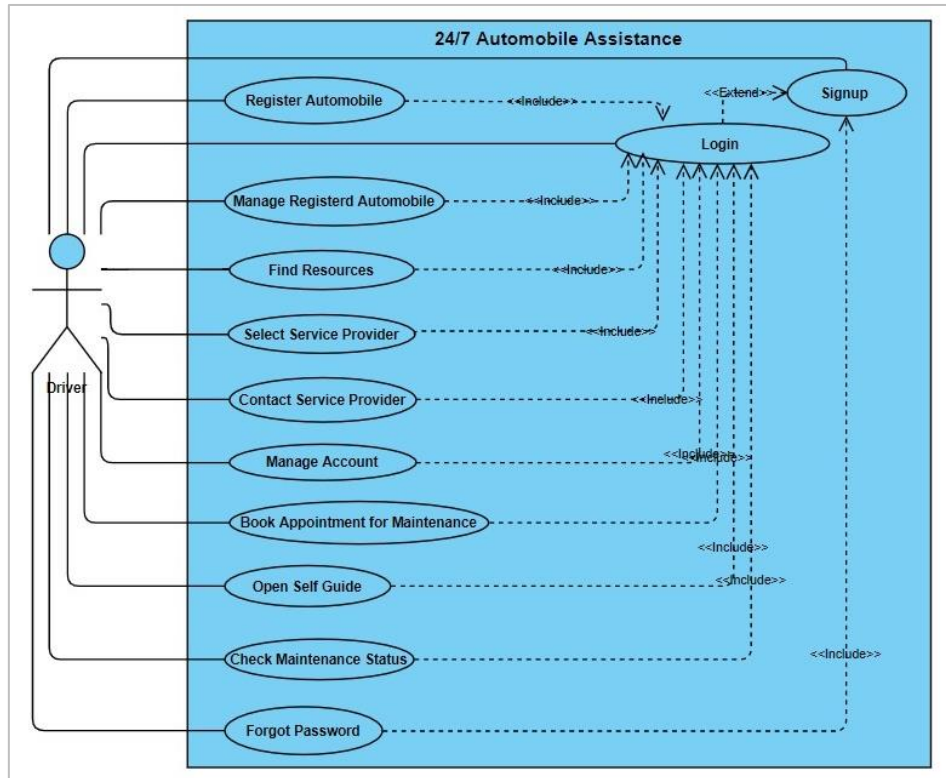


Figure 3.2 use case diagram (Driver's Module)

Use case diagram for driver presents the requirements for driver module as use cases. Each use case of this diagram shows a functionality of driver module.

3.2.3. Use case diagram for service provider

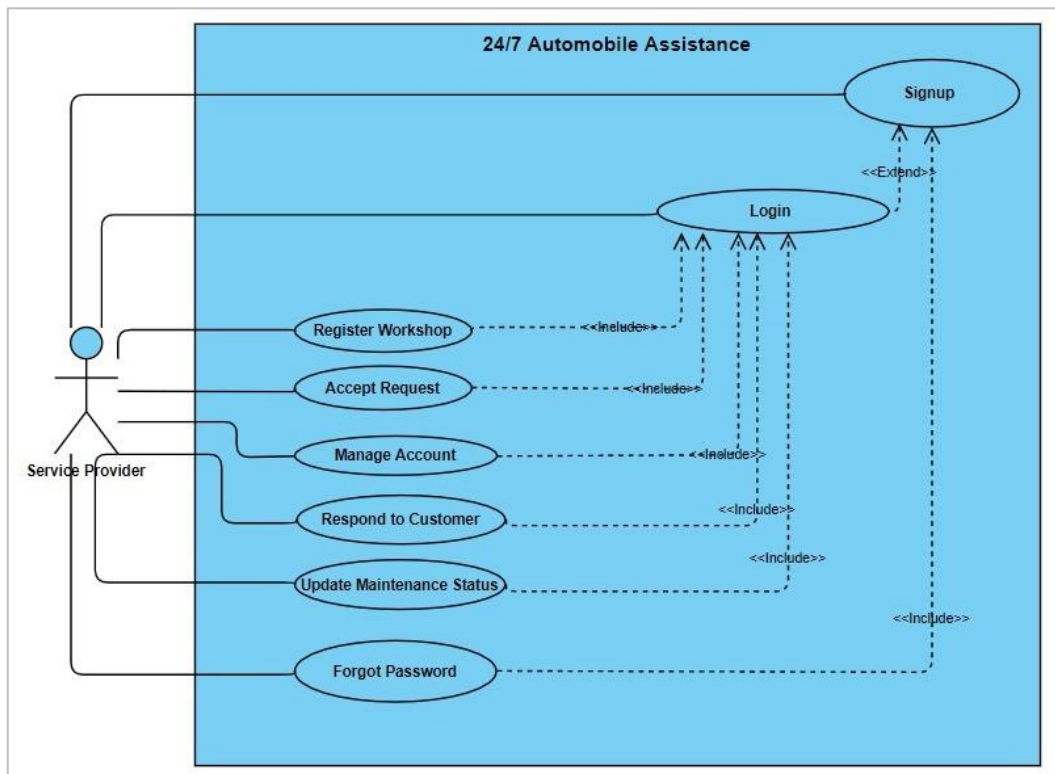


Figure 3.3 Use case diagram (Service Provider's Module)

Use case diagram for service provider presents the requirements for service provider's module as use cases. Each use case of this diagram shows a functionality of this module.

3.2.4. Use case diagram for Manage Automobile group

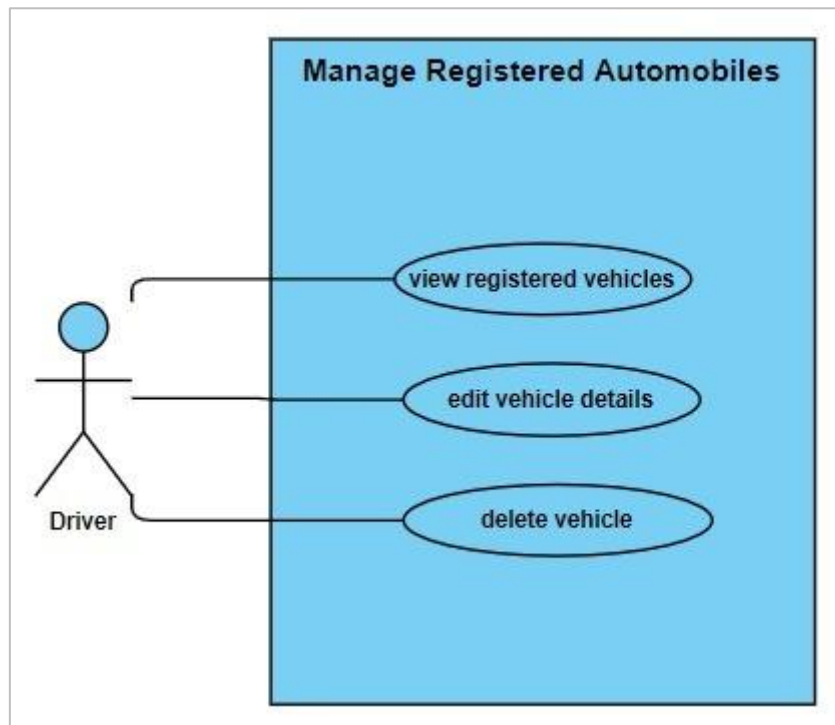


Figure 3.4 Use case diagram (Manage Registered Automobiles)

Manage Automobile is a group that consists of three further use cases. Due to similar functionality these use cases are grouped to “manage registered automobiles” use case in system level use case.

3.2.5. Use case diagram for find resources group

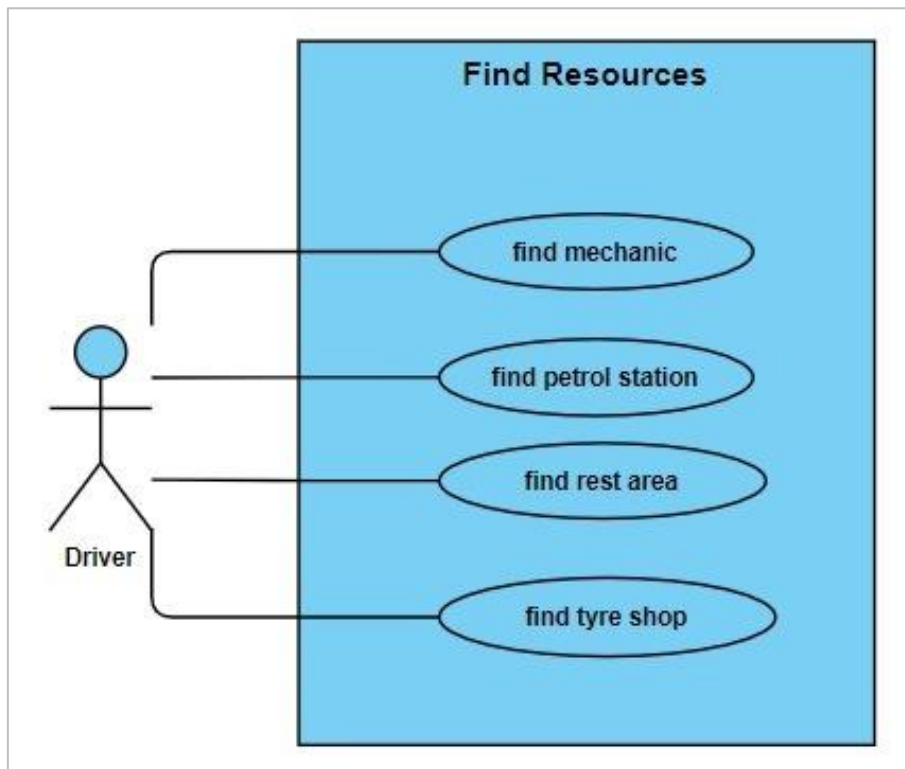


Figure 3.5 Use case diagram (Find Resources)

Find resources group has four further use cases, grouped due to similar functionality. User needs to select any of the type of resource to find that particular kind of resources.

3.2.6. Use case diagram for Manage Accounts group

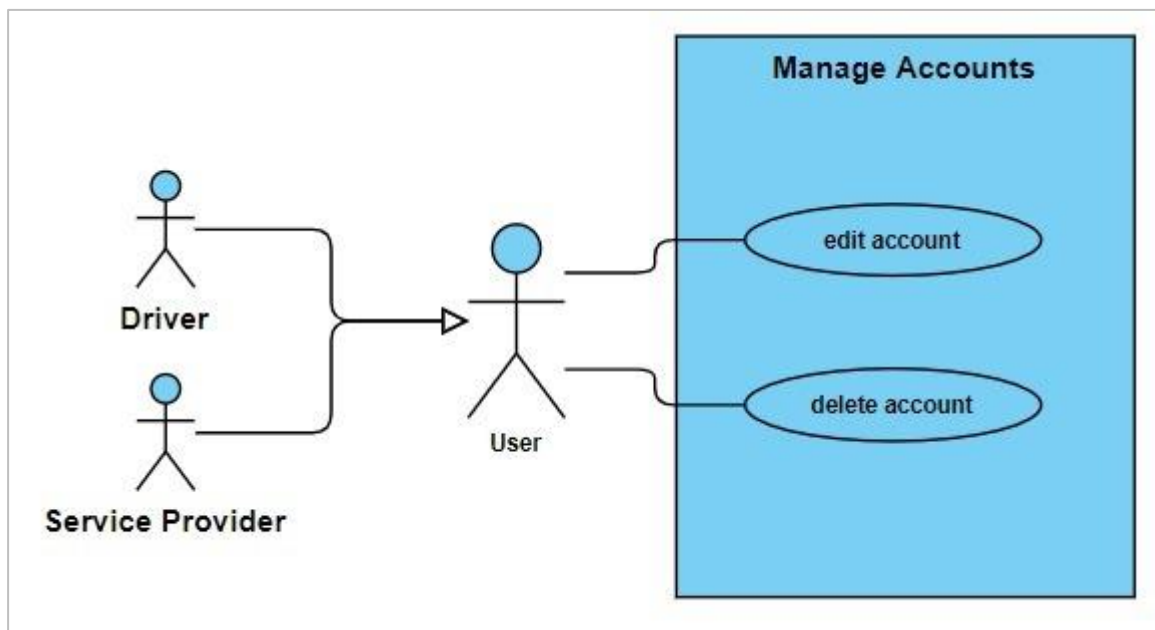


Figure 3.6 Use case diagram (Manage Accounts)

Manage accounts group has 2 sub use cases and it can be triggered by any type of user i.e. driver or service provider.

3.3. System features

System functional requirements are discussed earlier and described through use case diagrams. For a detailed overview each feature and its use case description covered in the coming subheadings.

3.3.1. User registration

3.3.1.1. Use case description for signup

Table 1: Use case description (Sign up)

Use Case ID:	AA001	
Use Case Name:	Signup	
Actor(s):	Driver, service provider	
Pre-Conditions:	User should not have created an account yet.	
Priority:	High	
Basic Flow:	User provides his/her basic information to create his/her account and get register to the system.	
Actor's Actions	System's Response	
1. User clicks on Signup button.	2. System asks the user to enter personal information.	
3. User provides his/her name, CNIC, mobile number, username, password and category (driver/service provider) and click Signup.	4. System will ensure that the provided CNIC is not already present in the system.	
	5. The system will save the user's information in database, register the user and take him/her to Login as driver screen.	
Alternative Course of Action		
Actor's Action	System's Response	
1. User clicks on signup button	2. If user's CNIC is already present in the system's database, the system will ask the user to login instead of creating new account.	

3.3.1.2. Use case description for Login

Table 2: : Use case description (Log in)

Use Case ID:	AA002	
Use Case Name:	Login	
Actor(s):	Driver, service provider	
Pre-Conditions:	User should be registered in the system.	
Priority:	High	
Basic Flow:	User provides his/her username and password to login to the system.	
Actor's Actions	System's Response	
1. User clicks on Login button.	2. System asks the user to select his category and enter his/her user name and password.	
3. User selects a category (either driver or service provider) and provides his/her username and password and clicks Login button.	4. System will verify the username and password.	
	5. The system makes the user login to the system successfully.	
Alternative Course of Action (Alternative case 1)		
Actor's Action	System's Response	
1. User provides invalid username or password.	2. The system will ask the user to re-enter his/her credentials.	
	2.2. The system provides the option of Forget password.	
3. The user provides valid credentials.	4. The system makes the user login successfully.	
Alternative Course of Action (Alternative case 2)		
Actor's Action	System's Response	
5. The user resets his/her password by using Forget password button.	6. The system sets the new password and asks the user to login with the new password.	

3.3.1.3. Use case description for Forget password

Table 3: Use case description (Forget Password)

Use Case ID:	AA020	
Use Case Name:	Forget Password	
Actor(s):	Driver, service provider	
Pre-Conditions:	User should be registered in the system.	
Priority:	Medium	
Basic Flow:	User provides his/her username and password to login to the system.	
Actor's Actions	System's Response	
1. User clicks on Forget Password button.	2. System asks the user to enter email.	
3. The user enters his/her email.	4. system verifies if the provided mail is registered in the system or not.	
	5. The system sends a verification email containing Password Reset Link and asks the user to reset his/her password through that link.	
6. The user opens the mail and clicks the link.	7. System allows the user to reset password.	
8. user resets his/her password and clicks OK.	8. System updates the password and asks the user to login.	
Alternative Course of Action (Alternative case 1)		
Actor's Action	System's Response	
1. The entered email does not exist.	2. System will show error message "The user with this email is not registered" and takes the user to sign up.	

3.3.2. Automobile Registration

3.3.2.1. Use case description for Register automobile

Table 4: Use case description (Register Automobile)

Use Case ID:	AA003
Use Case Name:	Register Automobile
Actor(s):	Driver
Pre-Conditions:	User must be registered and logged in to the system as a driver.
Priority:	High
Basic Flow:	User provides automobile information and the system will register the automobile.
Actor's Actions	System's Response
1. User clicks on Register Automobile button.	2. System will ask the user to enter vehicle details.
3. User will provide vehicle details e.g. category, company, model, number etc. and click register button.	4. The system will check if the vehicle with these details is already registered.
	5. The system will register the vehicle and show success message.
Alternative Course of Action	
Actor's Action	System's Response
1. User provides automobile details and clicks register button.	2. Vehicle with these details already exists.
	3. The system will show an error message, "Vehicle with this number is already registered by you".

3.3.3. Manage Registered automobiles

3.3.3.1. Use case description for view registered vehicles

Table 5: Use case description (View registered vehicles)

Use Case ID:	AA004
Use Case Name:	View registered vehicles
Actor(s):	Driver
Pre-Conditions:	User must be registered and logged in to the system as a driver.
Priority:	Low
Basic Flow:	User will be able to view the list of registered vehicles.
Actor's Actions	System's Response
1. User clicks on manage registered automobiles and selects view registered vehicles button.	2. System will show the list of registered vehicles to that user.
Alternative Course of Action	
Actor's Action	System's Response
Nil	Nil

3.3.3.2. Use case description for Edit vehicle details

Table 6: Use case description (Edit vehicle details)

Use Case ID:	AA005	
Use Case Name:	Edit vehicle details	
Actor(s):	Driver	
Pre-Conditions:	User must be registered and logged in to the system as a driver.	
Priority:	Medium	
Basic Flow:	User will be able to edit automobile details.	
Actor's Actions	System's Response	
1. User clicks on manage registered automobiles and selects Edit vehicle details button.	2. System will show a list of registered vehicles by that user.	
3. The user selects the vehicle which he/she wants to modify details.	4. The system opens the form containing the selected vehicle details.	
5. The user edits the vehicle detail(s) and clicks save button.	6. The system asks the user to login again.	
7. The user provides valid login credentials and logs in successfully.	8. The system will update the vehicle details and display a success message.	
Alternative Course of Action		
Actor's Action	System's Response	
1. user provides invalid login credentials.	2. The system will show error message "Invalid Username or Password" and logout the user from the system.	

3.3.3.3. Use case description for delete vehicle

Table 7: Use case description (Delete Vehicle)

Use Case ID:	AA006	
Use Case Name:	Delete vehicle	
Actor(s):	Driver	
Pre-Conditions:	User must be registered and logged in to the system as a driver.	
Priority:	Low	
Basic Flow:	User will be able to unregister a registered automobile.	
Actor's Actions	System's Response	
1. User clicks on Delete vehicle button.	2. System will show a list of registered vehicles by that user.	
3. The user selects the vehicle which he/she wants to unregister.	4. The system asks the user to login again in order to delete the selected record and unregister the .	
7. The user provides valid login credentials and logs in successfully.	6. The system deletes the record and the vehicle gets unregistered.	
Alternative Course of Action		
Actor's Action	System's Response	
1. user provides invalid login credentials.	2. The system will show error message "Invalid Username or Password" and logout the user from the system.	

3.3.4. Find Resources/service providers

3.3.4.1. Use case description for find resources

Table 8: Use case description (Find resources)

Use Case ID:	AA007	
Use Case Name:	Find Resources	
Actor(s):	Driver	
Pre-Conditions:	User must be registered and logged in to the system.	
Priority:	High	
Basic Flow:	User will be able to find resources.	
Actor's Actions	System's Response	
1. User clicks on Find Resources button.	2. System will ask the user to select a resource type.	
3. The user select one of the resource types i.e. Tire shop, Rest area, petrol station and local petrol agencies.	4. The system will display a list of available recourse of the selected type to the user.	
Alternative Course of Action		
Actor's Action	System's Response	
Nil	Nil	

3.3.5. Select service provider

3.3.5.1. Use case description for select service provider:

Table 9: Use case description (Select Service Provider)

Use Case ID:	AA008	
Use Case Name:	Select service provider	
Actor(s):	Driver	
Pre-Conditions:	User must be registered and logged in to the system.	
Priority:	High	
Basic Flow:	User will be able to select an available service provider and the service provider will be notified.	
Actor's Actions	System's Response	
1. User clicks on select service provider button.	2. System asks the user to select the type of service provider.	
3. The user selects a type of service provider e.g. mechanics, petrol station workers, tire shop workers etc.	4. The system will show the available service providers of the selected type. the user can click on a service provider and view the details.	
5. The user selects a service provider.	6. The system notifies the selected service provider and the user can contact with the service provider.	
Alternative Course of Action		
Actor's Action	System's Response	
1. The selected service provider cancels the request.	2. The system will show error message to the user (driver) "The selected service provider is not available now" and ask the user to select another service provider.	

3.3.6. Contact service provider:

3.3.6.1. Use case description for contact service provider

Table 10: Use case description (Contact Service Provider)

Use Case ID:	AA009
Use Case Name:	Contact service provider
Actor(s):	Driver
Pre-Conditions:	User must be registered and logged in to the system and selected a service provider.
Priority:	High
Basic Flow:	User will be able to contact with a selected service provider by using the selected method.
Actor's Actions	System's Response
1. User has selected a service provider and clicks on Contact service provider button.	2. System asks the user to choose a method. i.e. Chat, call.
3. The user selects a method to contact the selected service provider	4. The system contacts the selected service provider through selected method.
	5. The system makes the user talk with the service provider and discuss his/her problem.
Alternative Course of Action	
Actor's Action	System's Response
1. User contacts the service provider but the service provider did not respond.	2. The system shows error message "Cannot reach selected Service provider"

3.3.7. Self-Guide

3.3.7.1. Use case description for Open self-guide:

Table 11: Use case description (open self-guide)

Use Case ID:	AA011	
Use Case Name:	Open self-guide	
Actor(s):	Driver	
Pre-Conditions:	User must be registered and logged in to the system.	
Priority:	High	
Basic Flow:	The system will show offline solutions to the user.	
Actor's Actions	System's Response	
1. User clicks on Self-Guide button.	2. The system asks the user to select his problem from a list of problems for which offline solutions are available.	
3. The user selects a problem.	4. The system will provide the step by step procedure to fix user's problem.	
Alternative Course of Action		
Actor's Action	System's Response	
1. The user does not find his/her problem in the list of problems and selects "problem not found in the list" option.	2. The system will provide a list of some service providers' contact so that user can contact with the service providers and ask for help.	

3.3.8. Normal Maintenance

3.3.8.1. Use case description for Book appointment for maintenance:

Table 12: Use case description (Book appointment for maintenance)

Use Case ID:	AA012
Use Case Name:	Book appointment for maintenance
Actor(s):	Driver
Pre-Conditions:	User must be registered and logged in to the system.
Priority:	High
Basic Flow:	User will be able to select a mechanic and fix appointment for his/her car maintenance.
Actor's Actions	System's Response
1. User clicks on Book Appointment button.	2. System asks the user to select a service provider. The user can also select a service provider from Starred service providers.
3. The user selects a service provider.	4. The system opens the selected service provider's schedule, tells the user about the estimated time and asks for confirmation.
5. The user confirms booking.	6. The system notifies the selected service provider about the booking and shares car details.
Alternative Course of Action	
Actor's Action	System's Response
Nil	Nil

3.3.8.2. Use case description for Check maintenance status:

Table 13: Use case description (Check maintenance status)

Use Case ID:	AA013	
Use Case Name:	Check maintenance status	
Actor(s):	Driver	
Pre-Conditions:	User must be registered and logged in to the system and should have dropped his/her vehicle for maintenance after booking appointment.	
Priority:	Medium	
Basic Flow:	User will be able to view the maintenance status of vehicle and the remaining time to complete maintenance. After the maintenance is done the user will be notified.	
Actor's Actions	System's Response	
1. User clicks on check maintenance status button.	2. System will show the maintenance status of the vehicle.	
	3. The system will show the remaining time to complete maintenance.	
Alternative Course of Action		
Actor's Action	System's Response	
Nil	Nil	

3.3.9. Service provider management

3.3.9.1. Use case description for Register workshop:

Table 14: Use case description (Register workshop)

Use Case ID:	AA014	
Use Case Name:	Register workshop	
Actor(s):	Service provider	
Pre-Conditions:	User must be registered and logged in to the system as a service provider.	
Priority:	High	
Basic Flow:	User will be able to register workshop by providing details.	
Actor's Actions	System's Response	
1. User clicks on Register workshop button after logging in as service provider.	2. System opens a form and asks the user to enter workshop details.	
3. User enters workshop details i.e. workshop name, list of services the workshop will provide, rates of services etc.	4. The system stores the information and registers the workshop.	
Alternative Course of Action		
Actor's Action	System's Response	
Nil	Nil	

3.3.9.2. Use case description for Accept request:

Table 15: Use case description (Accept request)

Use Case ID:	AA015	
Use Case Name:	Accept request	
Actor(s):	Service provider	
Pre-Conditions:	User must be registered and logged in to the system as a service provider and there must be a request to be accepted.	
Priority:	High	
Basic Flow:	The service provider will be notified about a new request and he/she will accept/deny that request.	
Actor's Actions	System's Response	
1. The user clicks on check for requests or is notified about a new request.	2. The system shows the new request and its details and asks the user to accept/deny the request.	
3. The service provider accepts the request.	4. The system will notify the driver who has made the request and show him/her the time for service provider to reach the driver.	
Alternative Course of Action		
Actor's Action	System's Response	
1. The service provider does not accept the request.	2. The system asks to enter the reason for denial.	
3. The user enters the reason for denial.	4. System notifies the driver about request denial and asks the driver to choose another service provider.	

3.3.9.3. Use case description for Respond to customer

Table 16: Use case description (Respond to customer)

Use Case ID:	AA016	
Use Case Name:	Respond to customer	
Actor(s):	Service provider	
Pre-Conditions:	User must be registered and logged in to the system as a service provider and a driver tries to contact service provider.	
Priority:	Medium	
Basic Flow:	The service provider will respond to the customer.	
Actor's Actions	System's Response	
1. User gets a message or call from the customer (driver).	1. The system notifies the user that a customer is trying to contact you.	
3. The service provider answers the customer's call or replies customer's message.	4. The system connects the service provider and the driver.	
Alternative Course of Action		
Actor's Action	System's Response	
Nil	Nil	

3.3.9.4. Use case description for Update maintenance status

Table 17: Use case description (Update maintenance status)

Use Case ID:	AA018	
Use Case Name:	Update maintenance status	
Actor(s):	Service provider	
Pre-Conditions:	User must be registered and logged in to the system as a service provider and there is a vehicle in maintenance status.	
Priority:	Medium	
Basic Flow:	The service provider will be able to update the maintenance status of a vehicle.	
Actor's Actions	System's Response	
1. User click on update maintenance status.	2. The system allows the user to edit the current maintenance status of the vehicle under maintenance.	
	3. The system notifies the vehicle owner about the maintenance status update.	
Alternative Course of Action		
Actor's Action	System's Response	
Nil	Nil	

3.4. Interface Requirements

Interface requirements are further divided into sub sections.

3.4.1. User Interfaces

Detailed user Interface is given in chapter 4. There will be 3 interfaces of the system.

3.4.1.1. Main interface:

Main interface will contain login, signup, forget password and reset password screens.

3.4.1.2. Driver's interface:

Driver's interface will include a main screen which will offer the functionalities a driver can perform. Each functionality will contain a screen to present its functionality.

3.4.1.3. Service Provider's Interface:

Service provider's interface will include a main screen which will offer the functionalities a service provider can perform. Each functionality will contain a screen to present its functionality.

3.4.2. Hardware Interfaces

No external hardware was used in the system.

3.4.3. Software Interfaces

- For Database services system shall use to Firebase latest version.
- System will be able to run on android version above or equal to 6.0 and on iOS 11.0 or above.
- System is developed over a cross platform framework but debugging the app for iOS applications need a mac system, so initially only android application will be launched, however in case a mac system is available, the same system will be able to debug on mac and operate it on iOS.

3.4.4. Communications Interfaces

24/7 Automobile Assistance will be a mobile application which will HTTP Protocol. As the database that the system will use will be firebase. HTTP is the protocol that is used in firebase for making connection with firebase so the communication interface will be a HTTP protocol which will provide interface for communication between database and system.

3.5. Database Requirements

The system will use firebase real-time database for storage of data. All the requirements of fire base database are fulfilled and user's data privacy & secrecy is maintained. As the system is connected to a real time database so it will need active internet connection for the user to communicate with database. Service providers must need internet to use the system and communicate to the database however drivers/automobile users will be able to use limited functionalities while they are offline. The feature of self-guide may be used even if a user is offline and this feature will use the own database of the app.

3.6. Non-Functional Requirements

Non-functional requirements are further categorized and explained in coming sections.

3.6.1. Performance Requirements

- The system will be interactive, and there will be no or minimum delays in each action's response.
- In case of scrolling through the menu the delay will not be more than 2 seconds.
- The application will respond and communicate the data in real time & immediately

3.6.2. Safety Requirements

- The software is completely environmentally friendly and does not cause any safety violations.
- The interface and menus will have a flexible font and background color to avoid over constrain the eyes.

3.6.3. Security Requirements

- The system will make sure a proper and encrypted login authentication for both types of users.
- Information transmission will be securely transmitted to datastore (Firebase) without any changes to make sure data privacy and secrecy.
- The connection between the database and application will be secure.

3.6.4. Software Quality Attributes

Following quality attributes will be assured during the development of the system.

3.6.4.1. Availability:

The system will be made available 24/7, however It will be assured that in any case the server does not get down for more than a few minutes to avoid inconvenience of the users.

3.6.4.2. Correctness:

- The calculations and data that will be provided to any of the user types from system end will be assured to be correct.

- The data about the users that will be provided from client to client i.e. from service provider to driver or vice versa may or may not be correct. Its correctness depends upon the user who is providing data. For example if an auto mechanic/service provider provides an incorrect mobile no. or one which is not being used by him, the same will be shown to drivers while they will be seeing that service provider's profile.

3.6.4.3. Usability:

Interface of the software will be simple and easy to use.

3.6.4.4. Reliability:

- The system should provide accurate and correct results in any cases and any circumstances.
- There should be a reliable connection between the mobile application and the database.

3.6.4.5. Reusability:

Software will be usable. If the software is modified in future, the current system will be reused.

3.6.4.6. Robustness:

The system will ensure the validity of different inputs by implementing different checks.

3.6.4.7. Flexibility:

The system will be made flexible to adopt any changing requirements in future.

3.6.4.8. Maintainability:

Software will be made maintainable so that it any mobile developer can repair it and fix bugs and issues easily.

3.6.4.9. Portability:

Software will be able to be installed on all compatible devices and should run smoothly according to the user requirement.

3.7. Project Feasibility

To study the feasibility of a project, we need to know about two factors, firstly what problems the system is going to solve and secondly what will be the advantages of using this system. 24/7 Automobile assistance system will be a great contribution to the traditional automobile mechanic industry. The system will ease and catalyze the normal maintenance process and increase the market value of auto mechanic industry. The feasibility study of the system is categorized as Technical, operational and ethical feasibility and each type is discussed separately one by one.

3.7.1. Technical Feasibility

The system is developed using react native framework. The system is designed to cope with real life maintenance problems of drivers and automobile owners. It is also to increase the efficiency and profitability of mechanics and service providers. Currently the system is operable in the premises of Islamabad.

3.7.2. Operational Feasibility

The system was created in response to the current region's conditions and can be easily applied in this setting. The system is functional in the current environment and can withstand all the conditions and stresses of this region. The system can be moved to any other environment and easily configured for that environment. The system is highly scalable and can be configured in any way.

3.7.3. Legal and Ethical Feasibility

After installation, this application will prompt users to grant permissions for using mobile and accessing current location, addressing all legal and ethical concerns. The system will not have access to the user's personal information, rather it would be store on a real time database and will not be shared with any third party without user's permission.

Chapter 4

System Design

4.1. Design approach

The design approach used for 24/7 Automobile Assistance System is function oriented. The system's design is classified into parts and each part has a separate mobile screen and each part will provide one or more specific functionality.

4.2. System Architecture

The system will use n-tier client server architecture. There will be client end and server end and both will be able to run on a separate machine. Client and server end will contain the tiers as follows;

4.2.1. Presentation tier:

Presentation tier contains presentation logic so it will provide user interface or front end and interaction with users. Presentation logic is implemented in client side so client end will contain the presentation tier.

4.2.2. Logic tier:

Logic tier will contain the business logic so it will include backend code. The main functionality will be developed in the logic layer and this tier will also be implemented on client end.

4.2.3. Data tier:

Data tier will contain data access logic and will connect the database server. Data will be stored and managed in this layer.

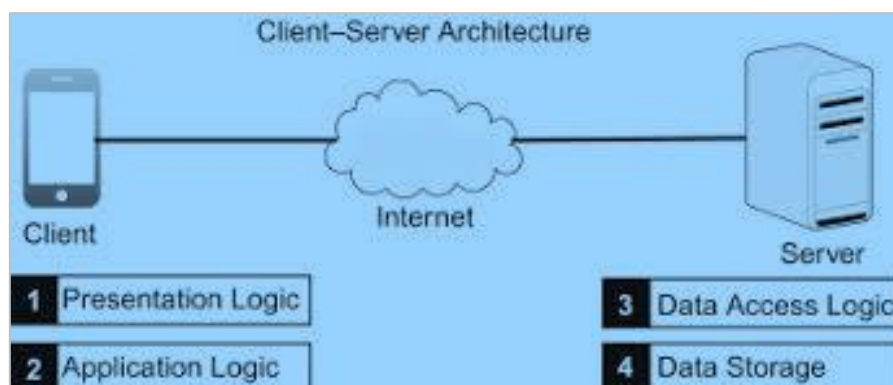


Figure 4.1 System Architecture of 24/7 Automobile Assistance

4.3. Logical Design

Logical view/design of the system is described through class diagram.

4.3.1. Class diagram

Class diagram explains the logical connections between different classes.

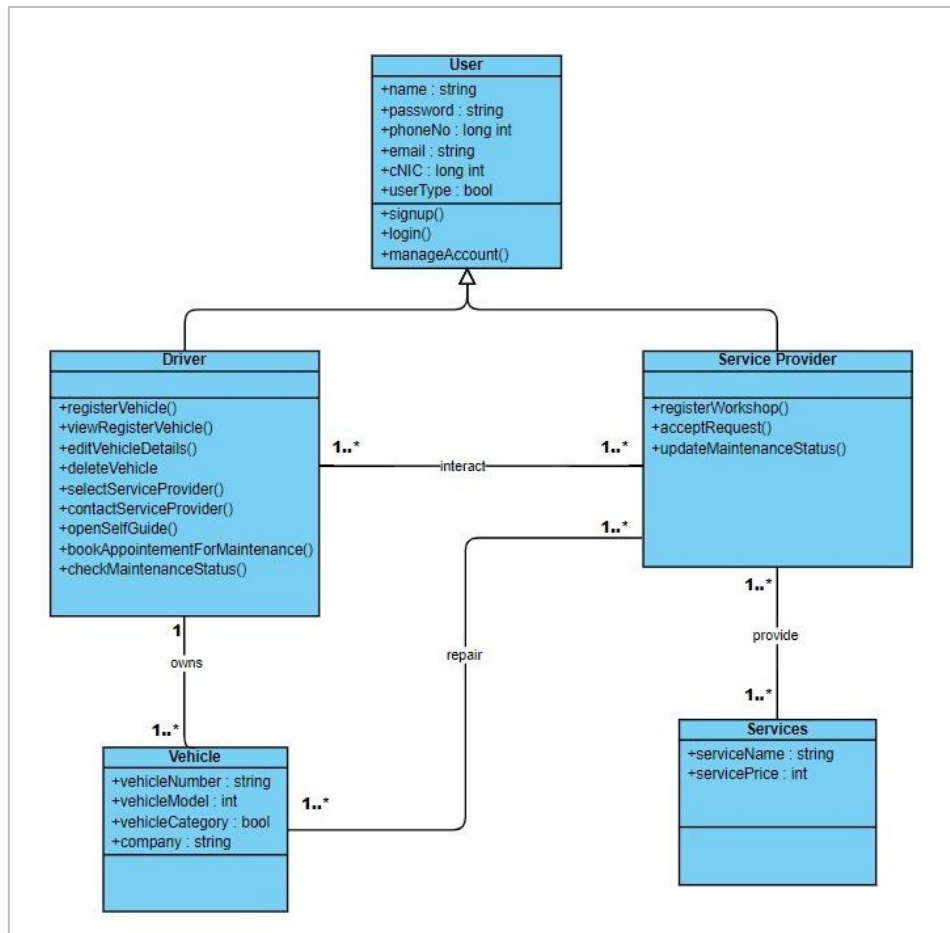


Figure 4.2 Logical view of the system

4.4. Dynamic view

Dynamic view of the system is described through use case diagram and sequence diagrams for those use cases.

4.4.1. Use case Diagram

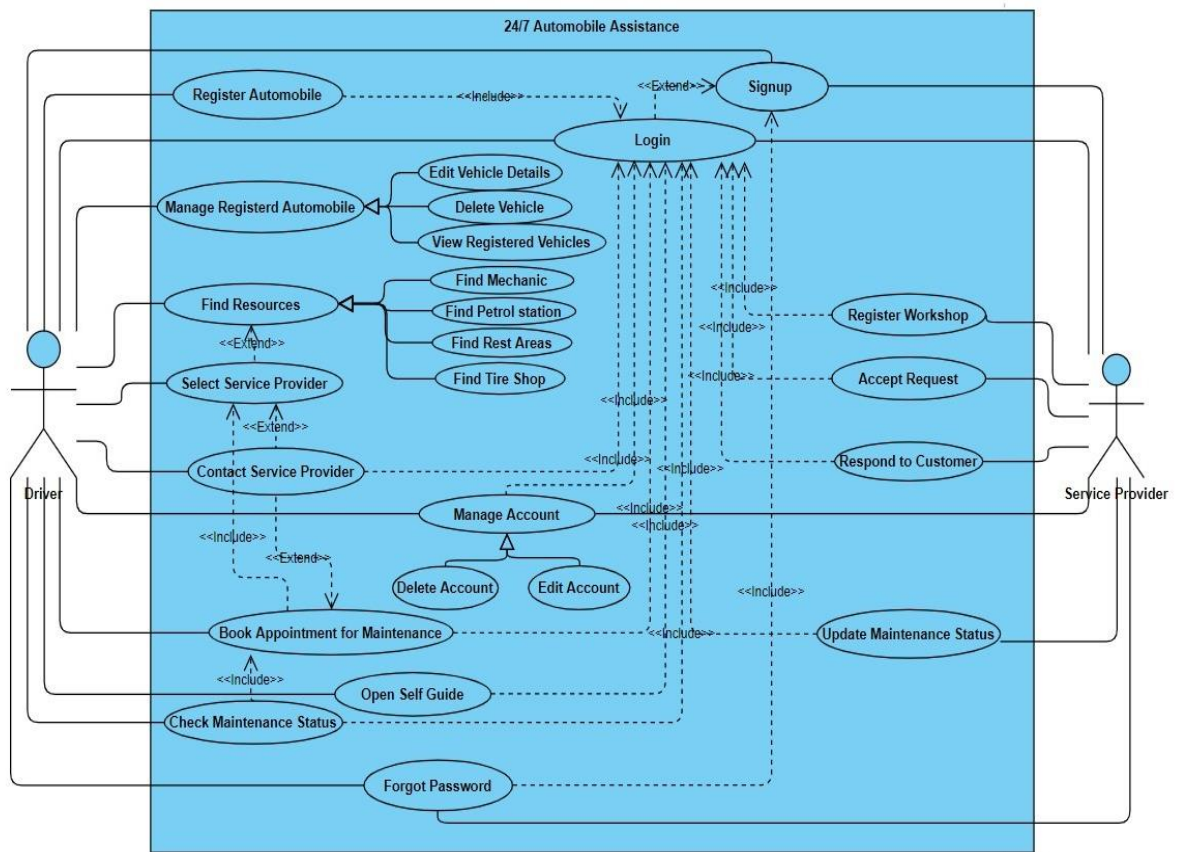


Figure 4.3 Use case diagram of the system

4.4.2. Sequence Diagram of Use cases

Sequence diagrams show the flow of different activities. Sequence diagram against each use case is given and it explains the user's actions and system responses against each action of that particular use case.

4.4.2.1. Sequence Diagram for Sign up

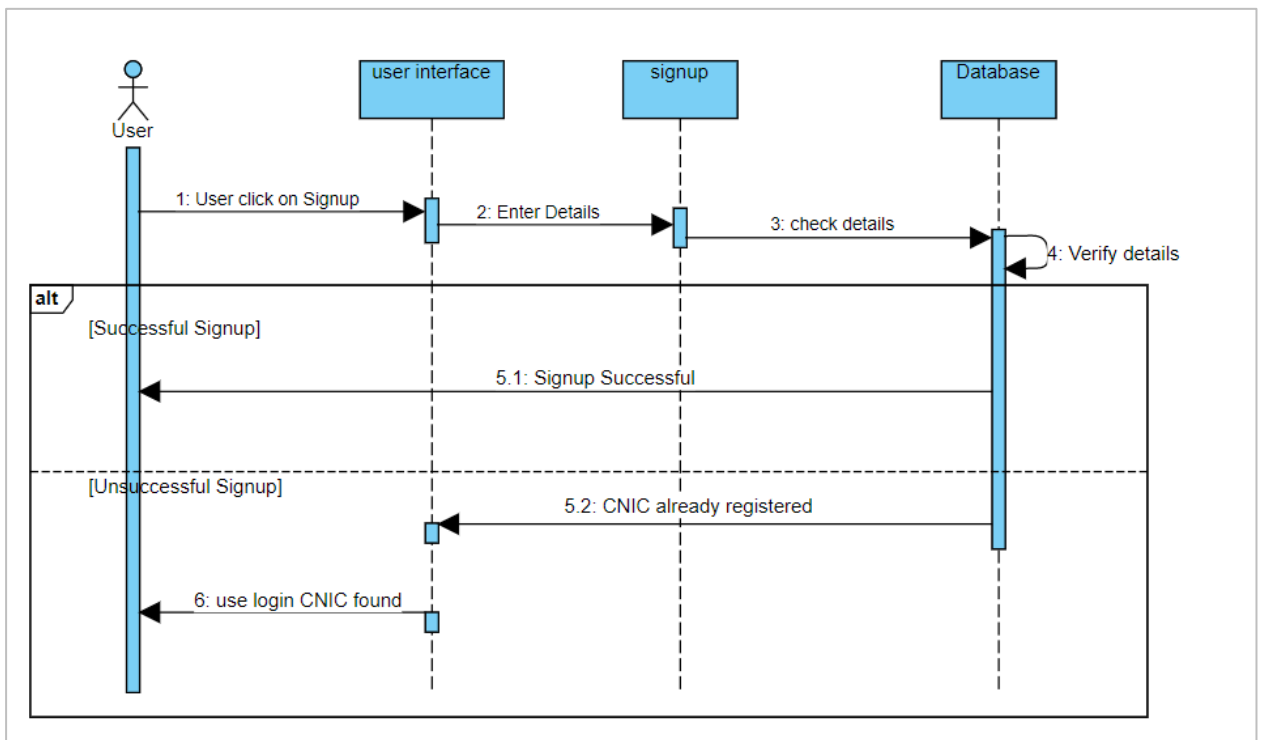


Figure 4.4 Sequence Diagram for Sign Up

4.4.2.2. Sequence Diagram for Log In

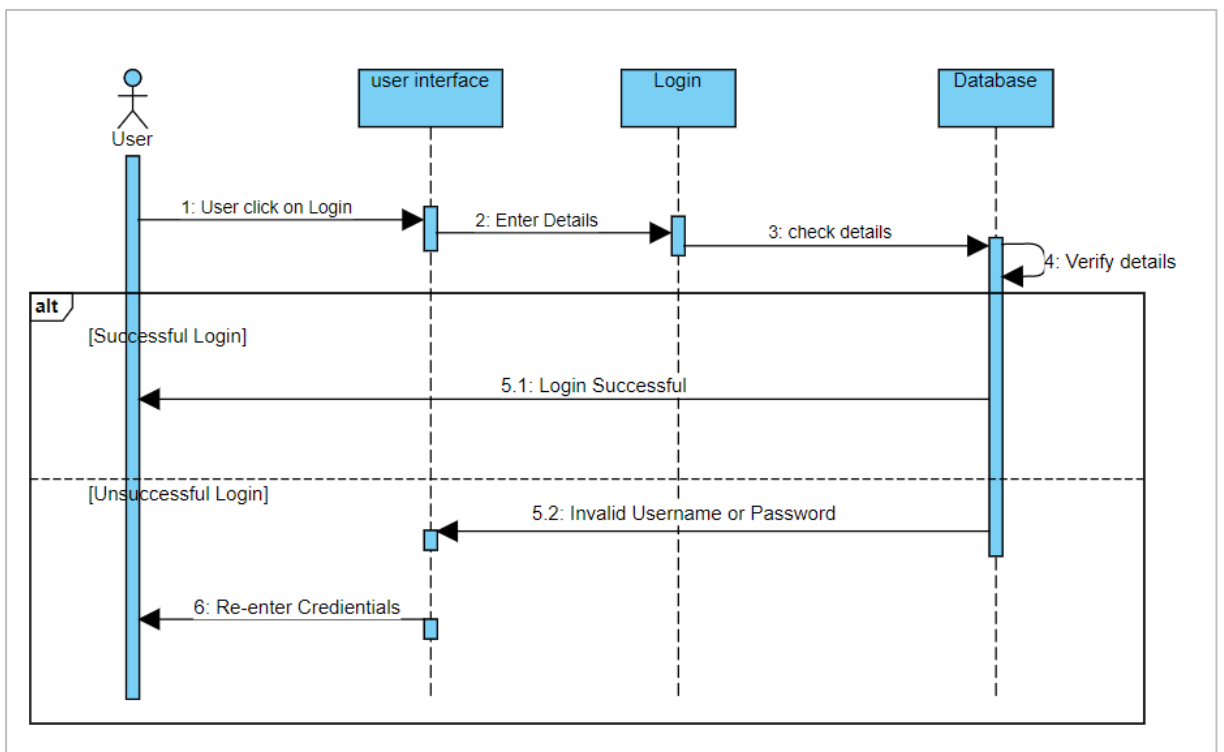


Figure 4.5 Sequence Diagram for Log In

4.4.2.3. Sequence Diagram for Forget Password

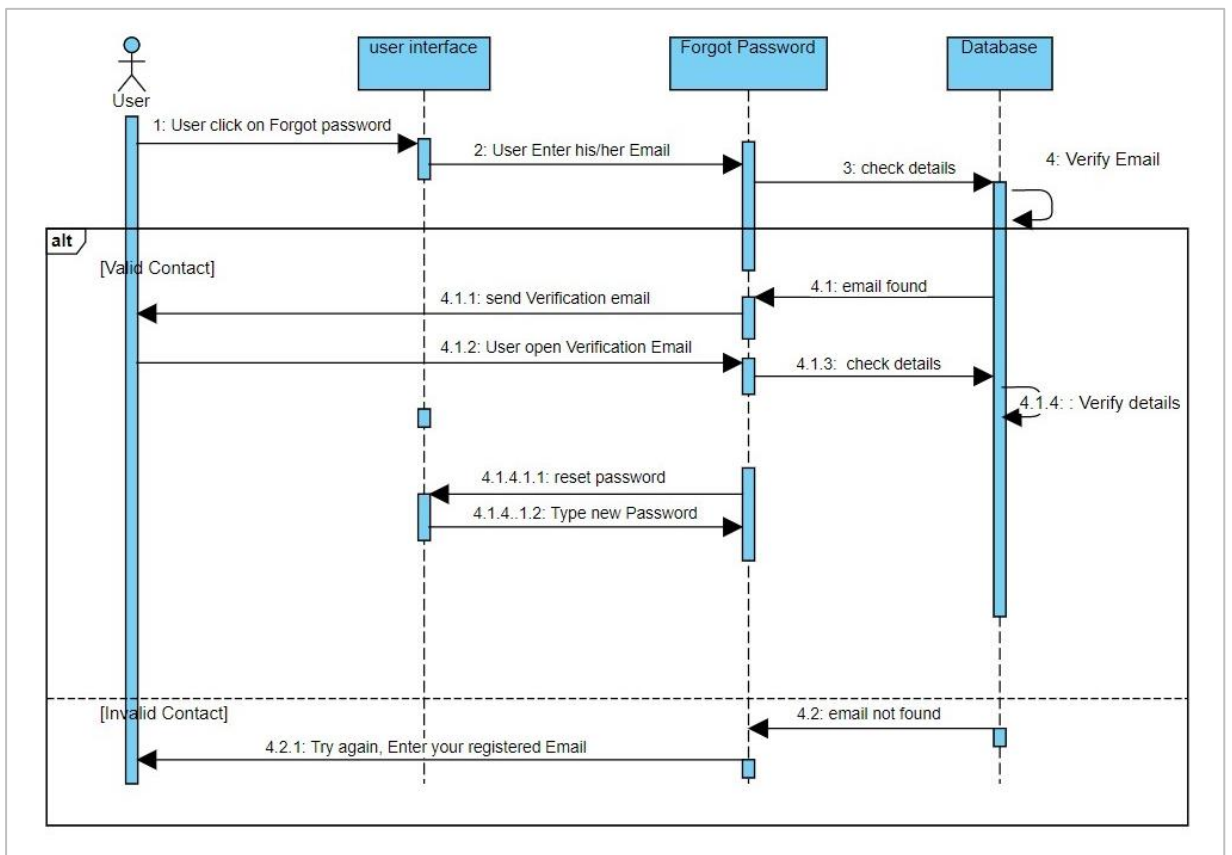


Figure 4.6 Sequence diagram for Forget Password

4.4.2.4. Sequence Diagram for Register Automobile

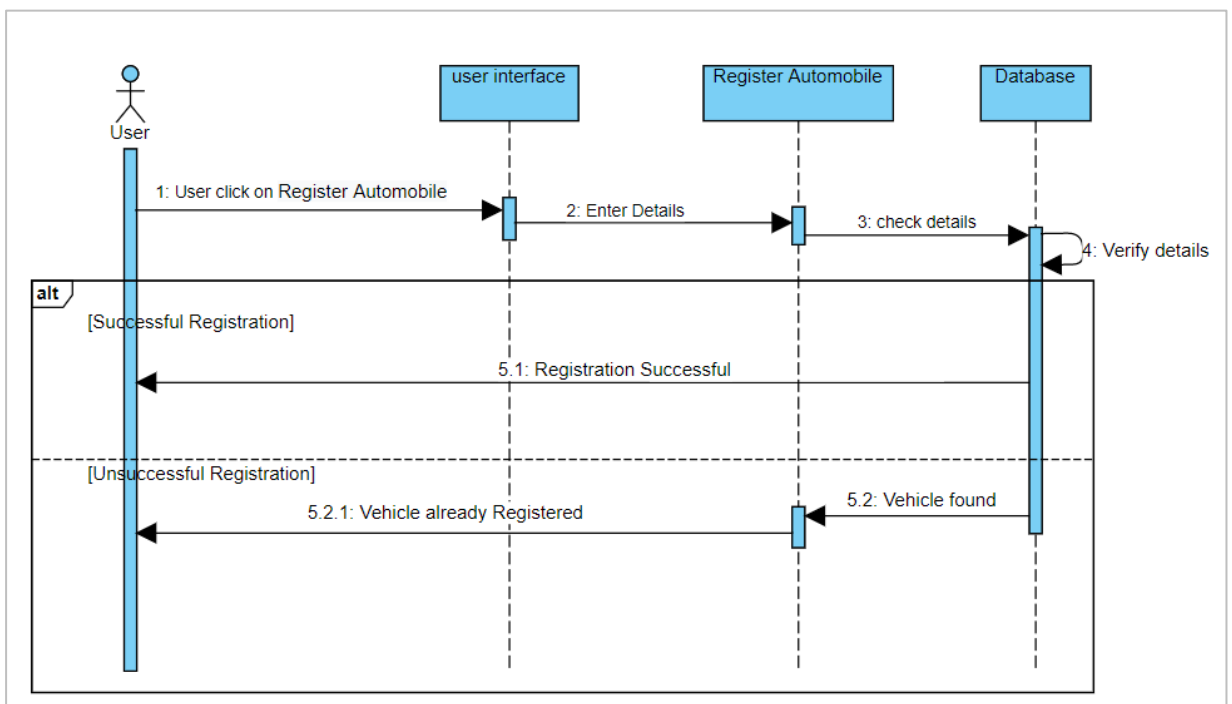


Figure 4.7 Sequence Diagram for Register Automobile

4.4.2.5. Sequence Diagram for View Registered Automobiles

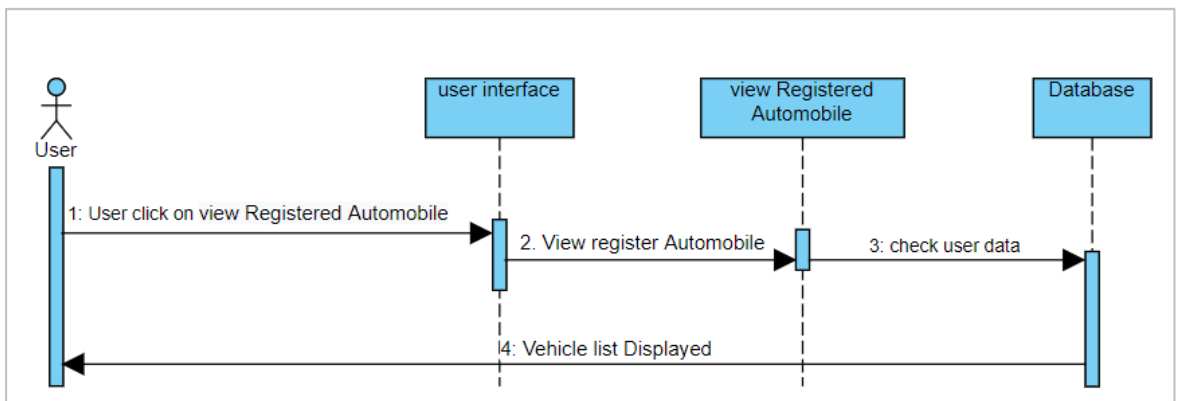


Figure 4.8 Sequence Diagram for View Registered Automobiles

4.4.2.6. Sequence Diagram for Edit Vehicle Details

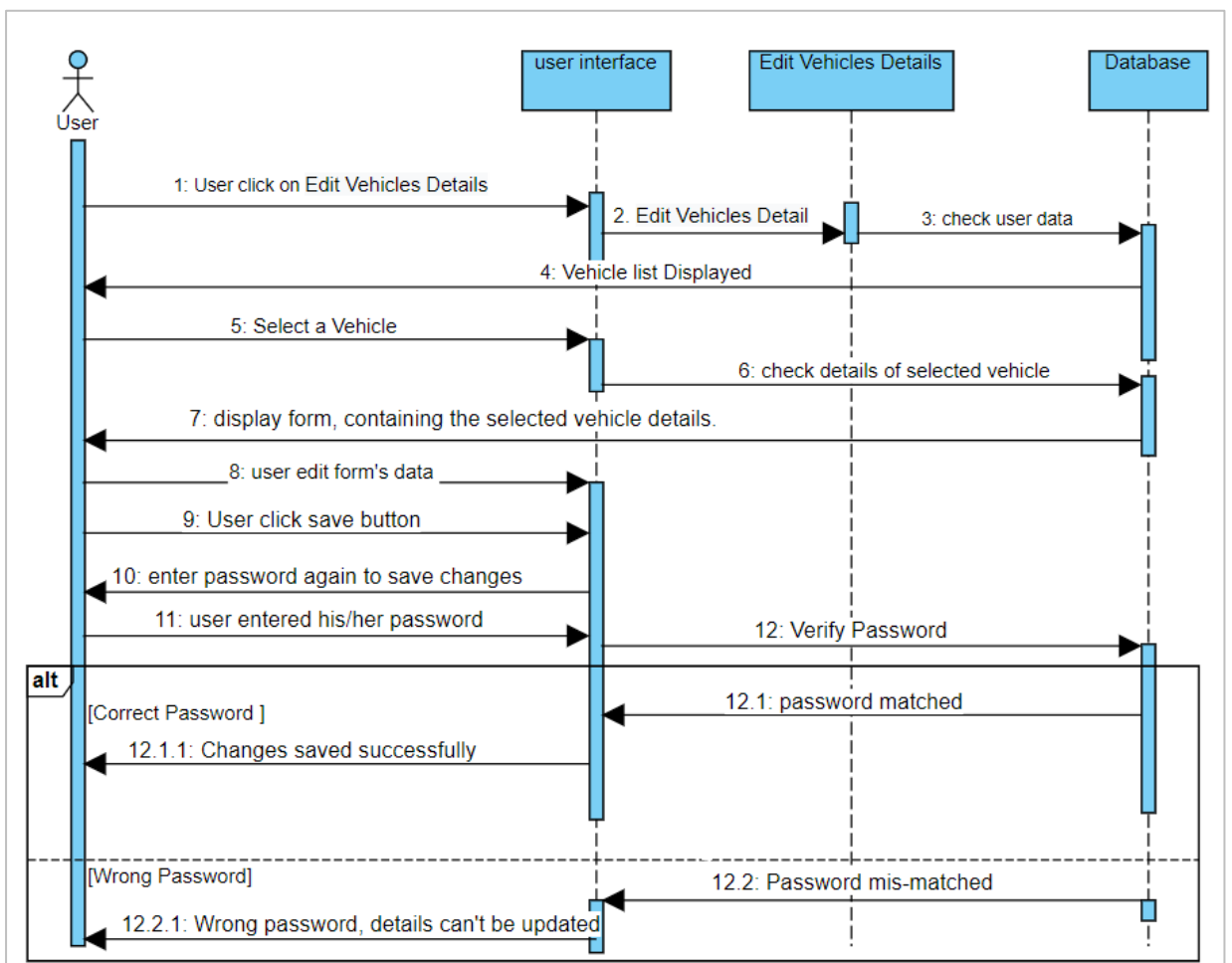


Figure 4.9 Sequence Diagram for Edit Vehicle details

4.4.2.7. Sequence Diagram for Find Resources

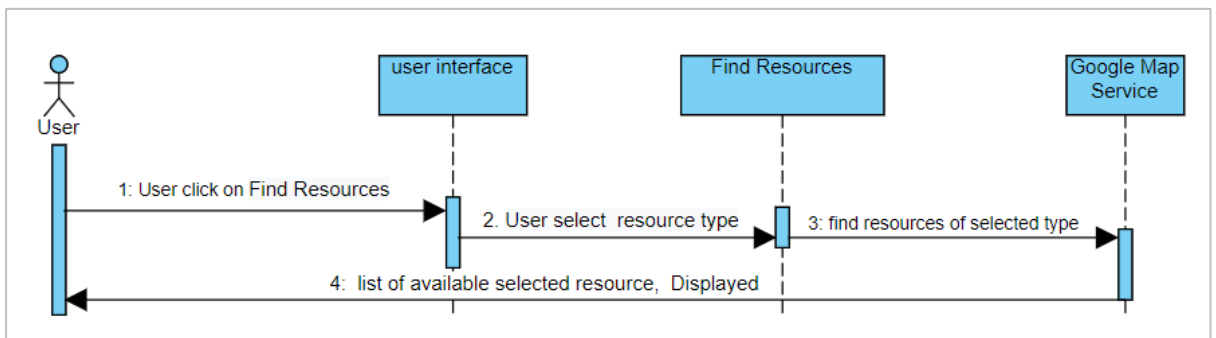


Figure 4.10 Sequence Diagram for Find Resources

4.4.2.8. Sequence Diagram for Delete Vehicle

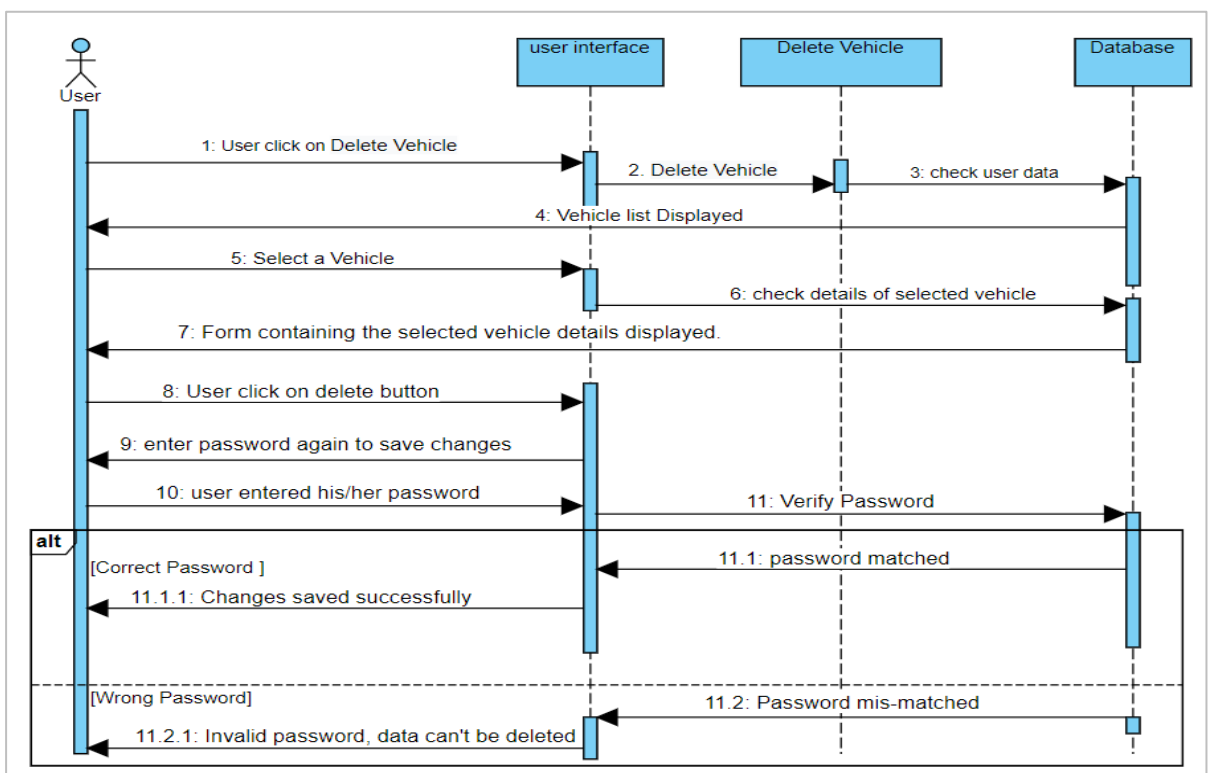


Figure 4.11 Sequence Diagram for Delete Vehicle

4.4.2.9. Sequence Diagram for Select Service Provider

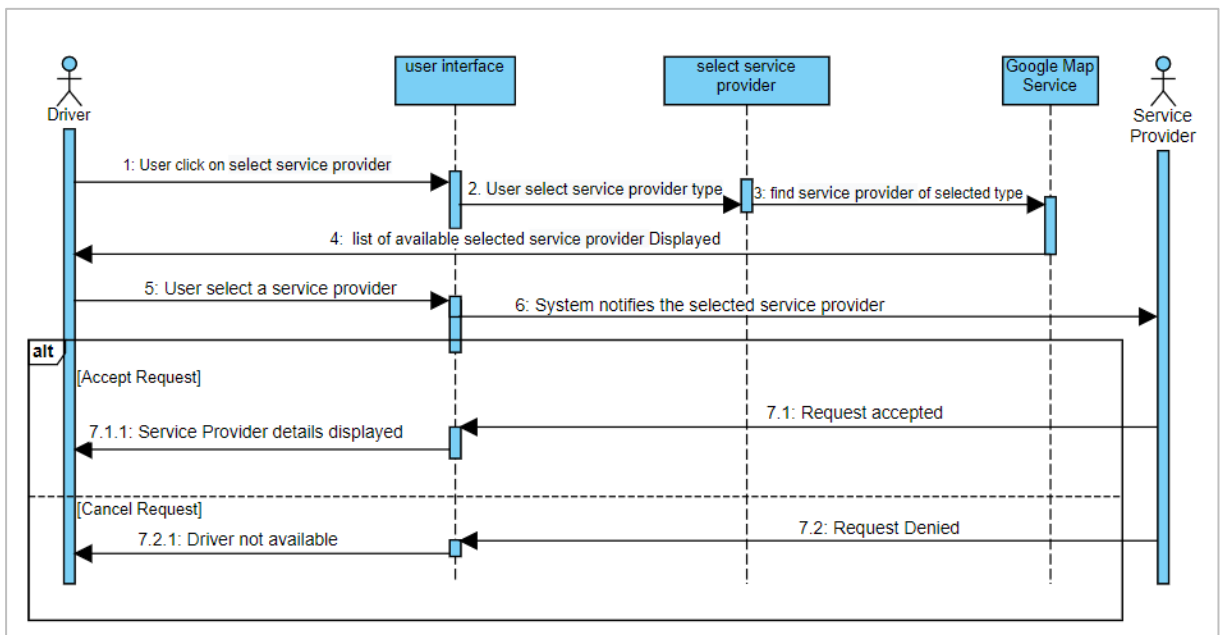


Figure 4.12 Sequence Diagram for Select Service Provider

4.4.2.10. Sequence Diagram for Contact service Provider

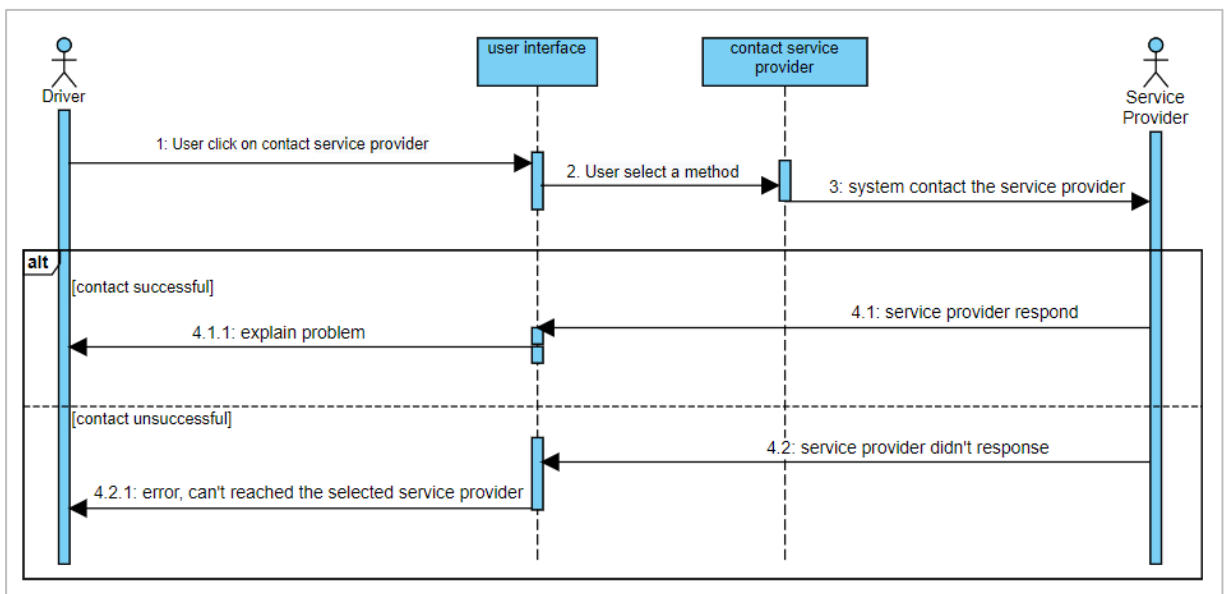


Figure 4.13 Sequence Diagram for Contact Service Provider

4.4.2.11. Sequence Diagram for Book Appointment

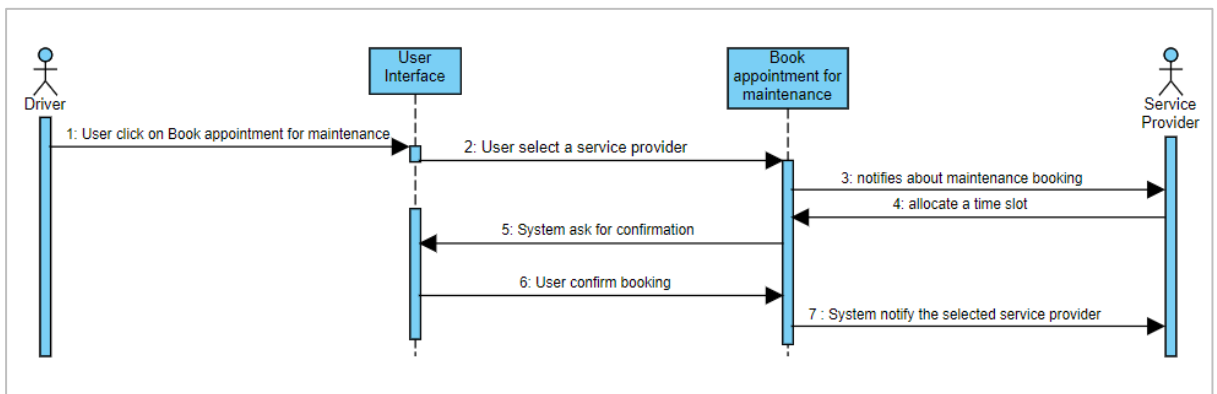


Figure 4.14 Sequence Diagram for Book Appointment

4.4.2.12. Sequence Diagram for Check Maintenance Status

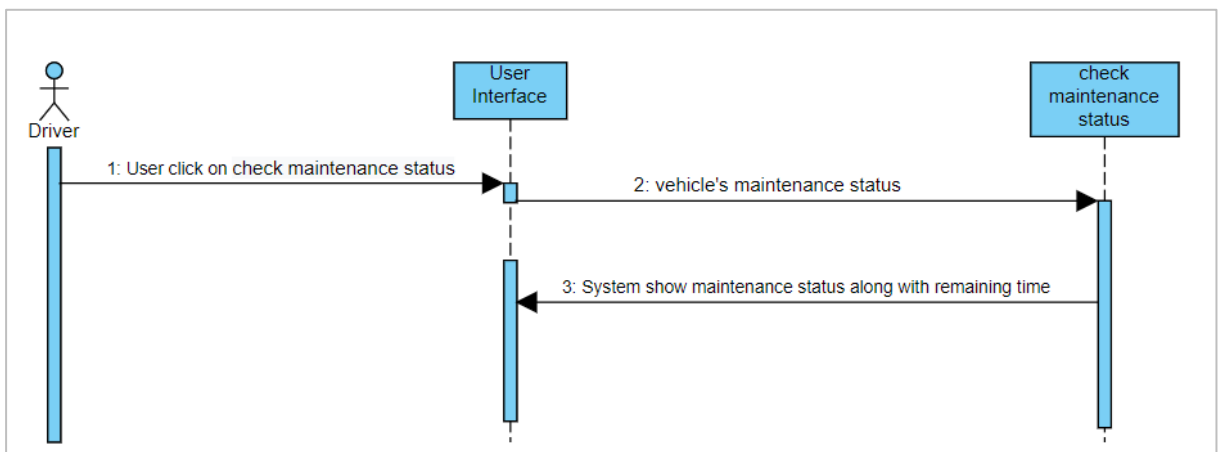


Figure 4.15 Sequence Diagram for Check Maintenance Status

4.4.2.13. Sequence Diagram for Register Workshop

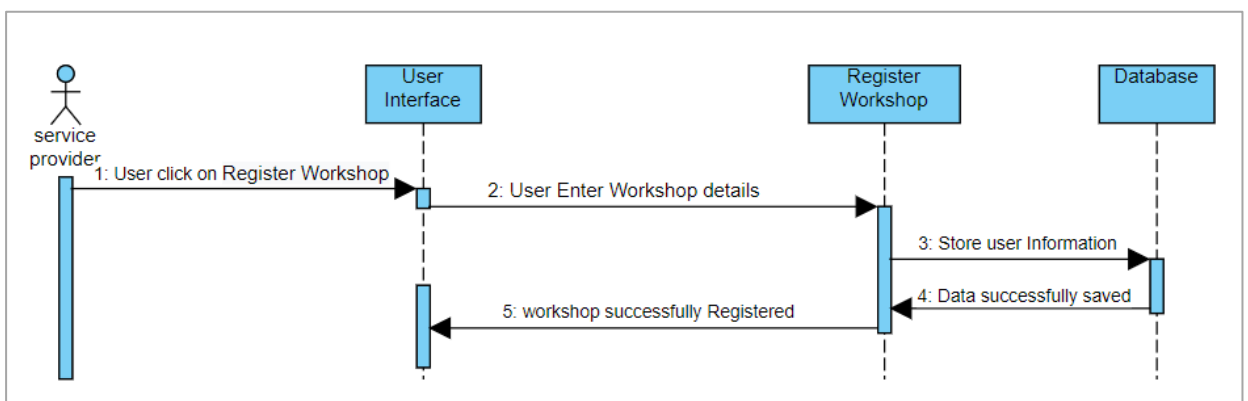


Figure 4.16 Sequence Diagram for Register Workshop

4.5. Component design

Component design of the system describes the whole system as a set or group of components, modules or activities. It consists of deployment view and work breakdown structure, to elaborate how the system is broken down into sub activities, developed and deployed.

4.5.1. Deployment diagram

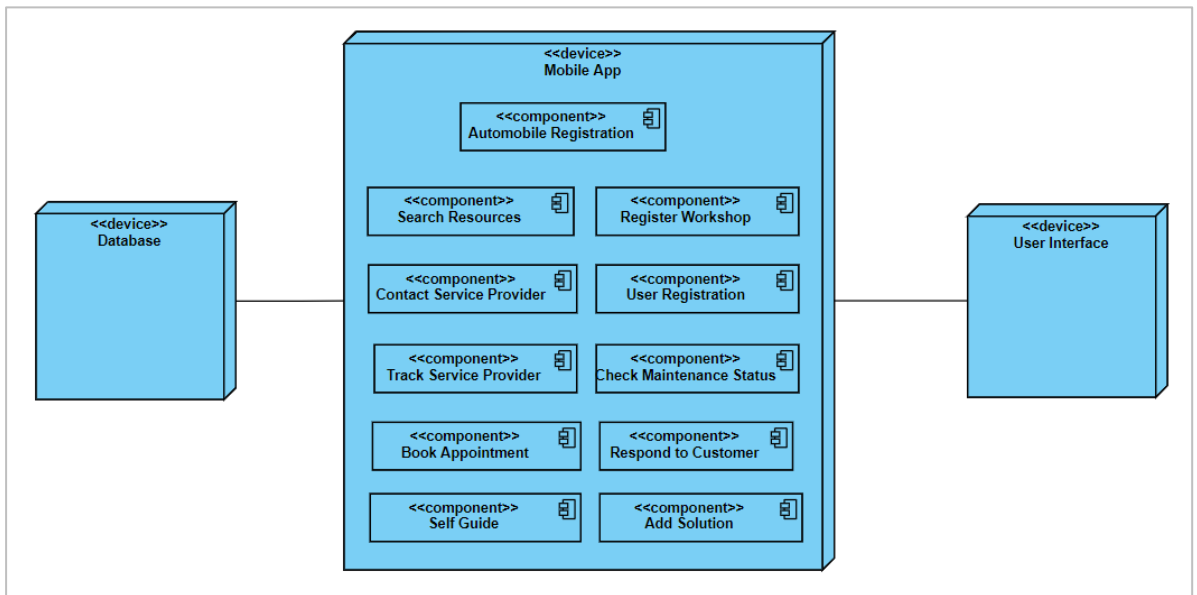


Figure 4.17 Deployment View of the System

4.5.2. Work Breakdown Structure

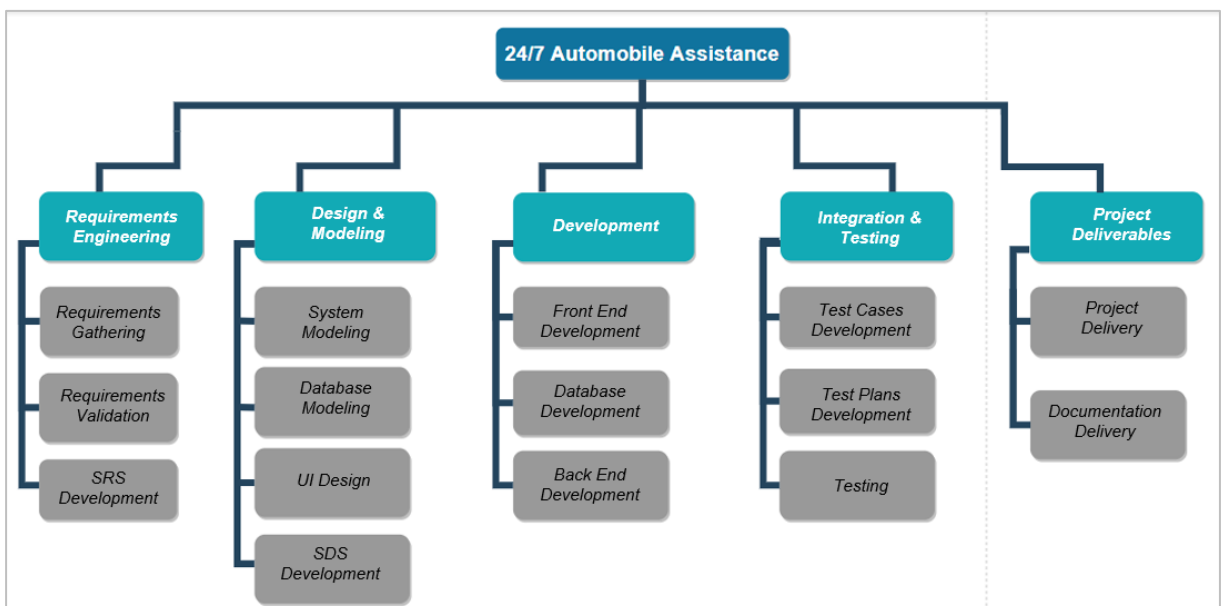


Figure 4.18 Work Breakdown Structure

4.5.3. Work Breakdown Dictionary

Table 18: Work breakdown structure

ID	Name	Description
0.	24/7 Automobile Assistance	System to be developed
1.	Requirements Engineering	The process of dealing with requirements. Further divided in sub processes.
1.1.	Requirements Gathering	Requirements of the proposed system to be gathered by different stakeholders.
1.2.	Requirements Validation	The gathered requirements will be analysed, verified and validated.
1.3.	SRS Development	After verifying and validating requirements, they will be documented properly, which will produce System Requirements Specification document.
2.	Design & Modelling	The process of system modelling. Further divided in sub processes.
2.1	System Modelling	System to be modelled on the basis of Requirements gathered and documented during the Requirements phase. Different perspectives and views of the system e.g. logical view , dynamic view are modelled.
2.2.	Database Modelling	Data models of the system to be produced in this phase. It may contain database design/ schema and data model e.g. ER model.
2.3.	UI Design	User Interface of the system will be designed in this phase. Will contain mock-up screens with no functionality.
2.4.	SDS Development	Software Design Specification document will contain complete design of the system and its details.
3.	Development	The process of system development. Further divided in sub processes.
3.1.	Frontend Development	Front End of the system to be developed on the basis of the UI design.
3.2.	Database Development	Database of the system will be created on the basis of the schema designed during database modelling.
3.3.	Backend Development	Frontend and database will be linked and complete functionality will be implemented in this phase.
4.	Testing	The process performs the testing of the proposed system after it is developed. Further divided in sub processes.
4.1.	Test Cases Development	Test cases will be developed for the system to test the functionalities.
4.2.	Test Plans Development	Test plans will be developed to execute the test cases.
4.3.	Testing	Testing will be performed by executing all the test cases.
5.	Project deliverables	Includes the deliverables that will be produced after the successful completion of the project. Further divided in sub processes.
5.1.	Project delivery	Complete functional project to be delivered.
5.2.	Document delivery	All the documentation e.g. SRS, SDS to be delivered.

4.5.4. Network Model

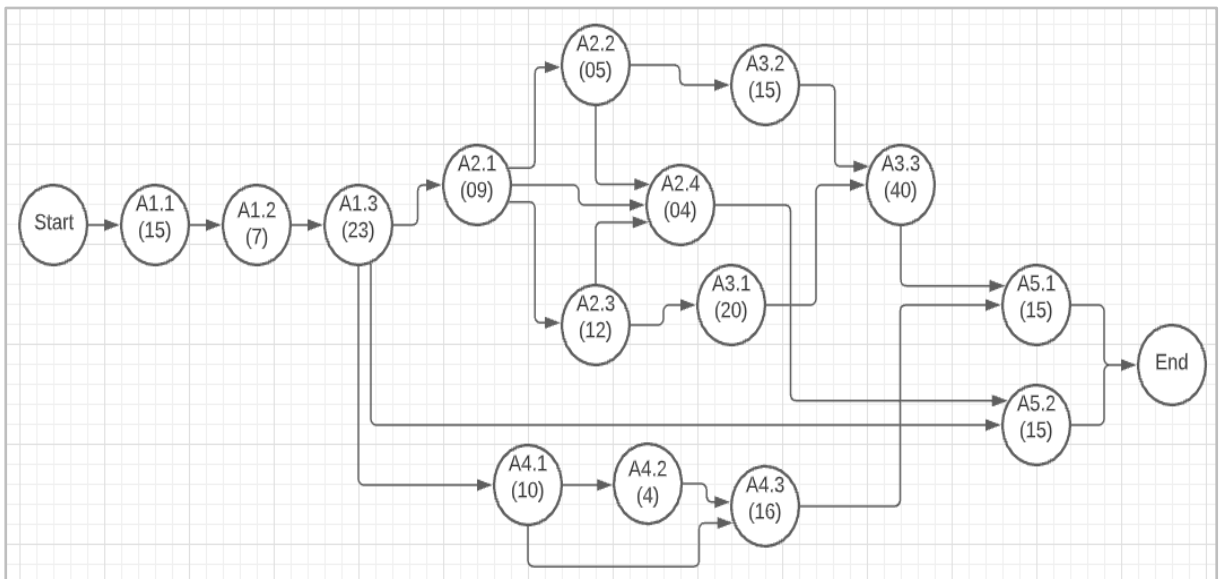


Figure 4.19 CPM Network diagram

4.6. Data model

As the system uses firebase as database and firebase does not have a specified ER model. However the data model of the system will be as follows

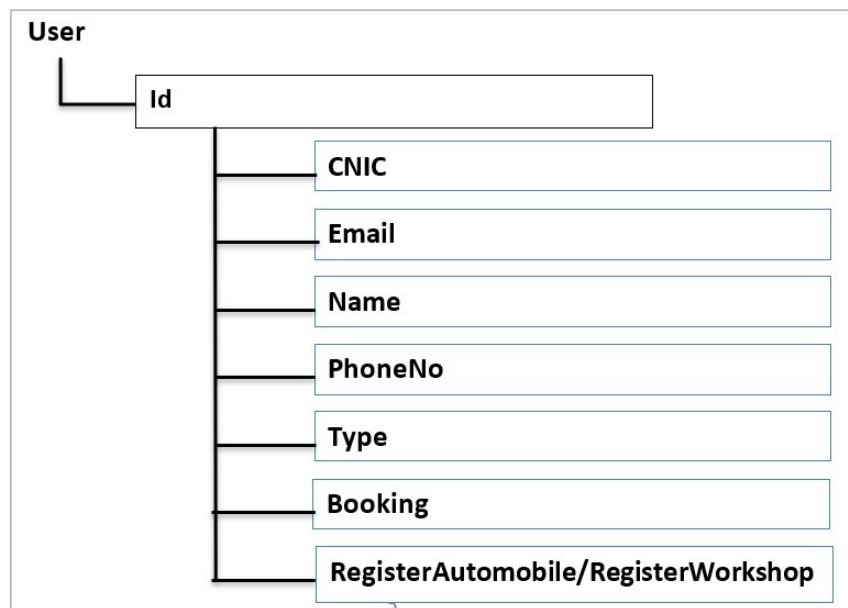


Figure 4.20 Data model of the system (1 of 4)

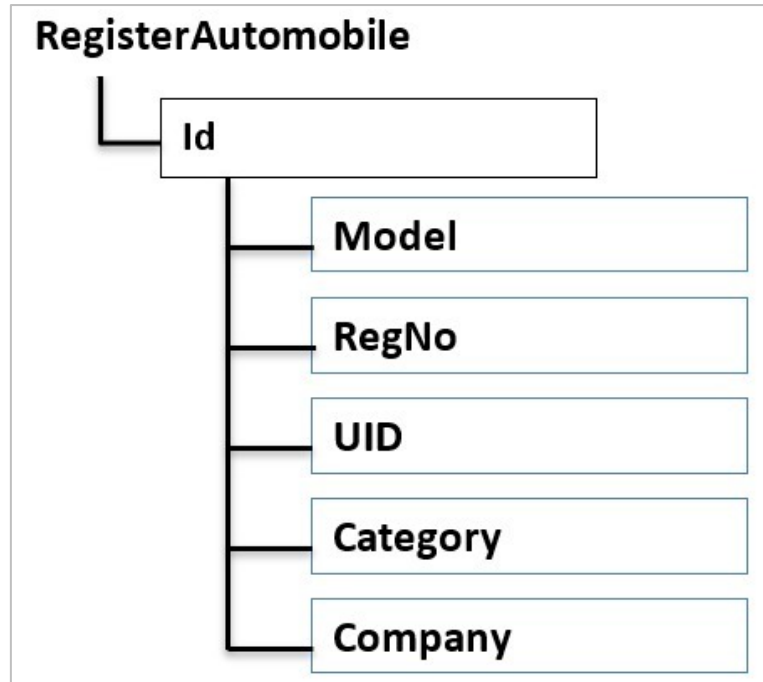


Figure 4.21 Data model of the system (2 of 4)

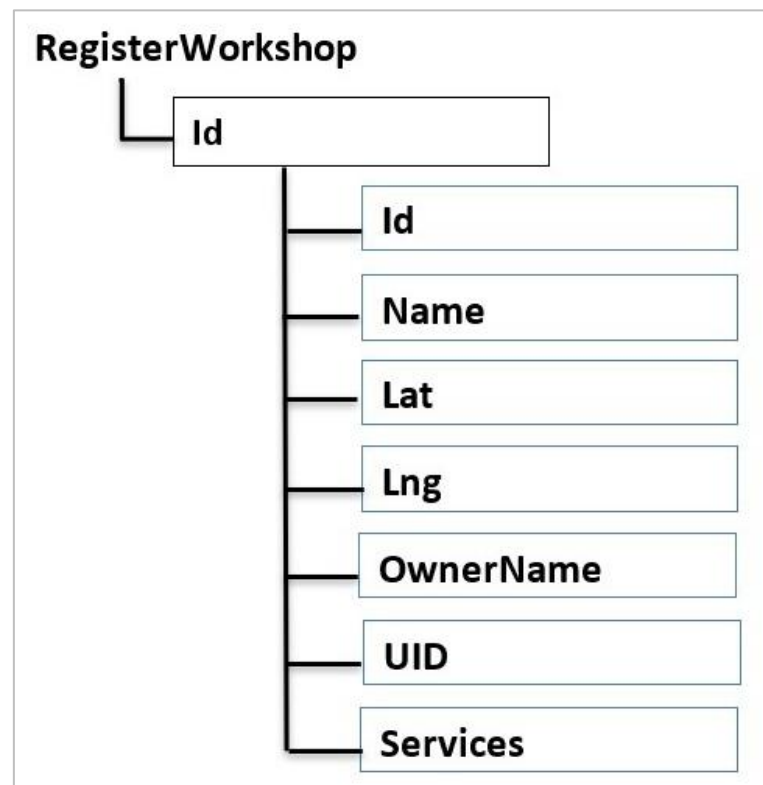


Figure 4.22 Data model of the system (3 of 4)

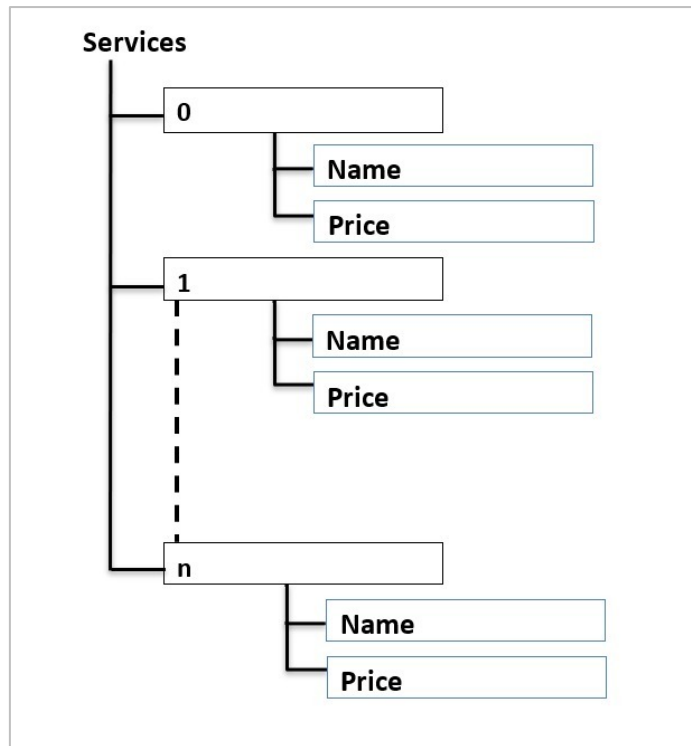


Figure 4.23 Data Model of the System (4 of 4)

4.7. Interface Design

Complete interface design of the system is explained through system prototype and User Interface. System prototype includes the initial interface and layout of the system while complete functional UI is also designed to elaborate the look and feel of the system more clearly.

4.7.1. System Prototype

System prototype describes the basic interface design of the system. Changes were imposed and made on the recommendation of respected project supervisor and evaluation team.

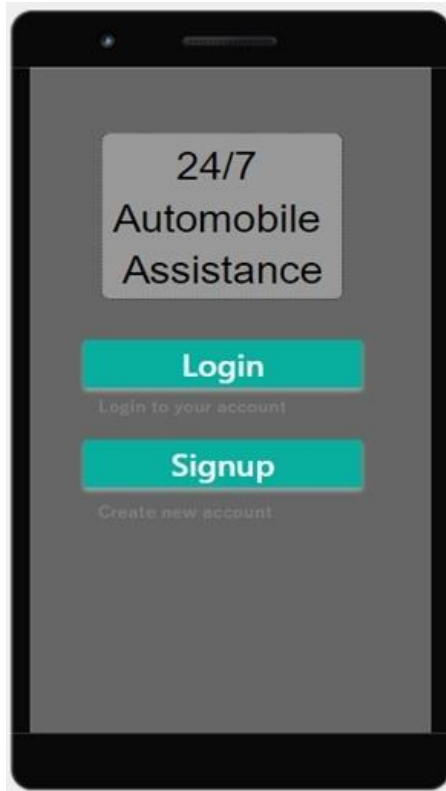


Figure 4.24 System Prototype (Main Screen)

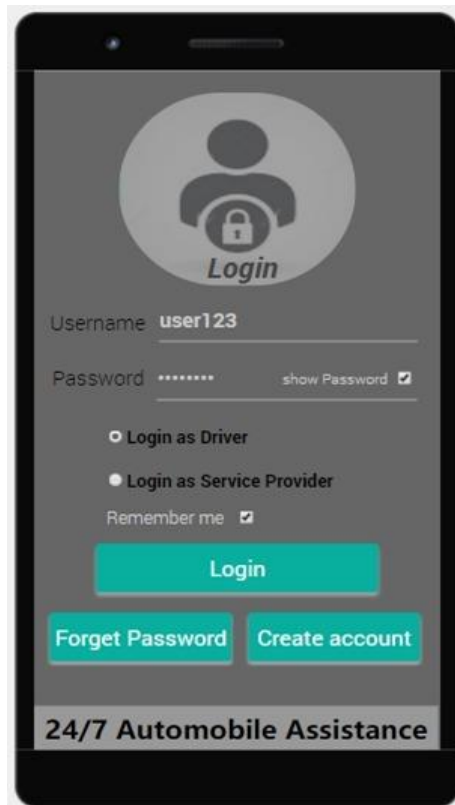


Figure 4.25 System Prototype (Login Screen)

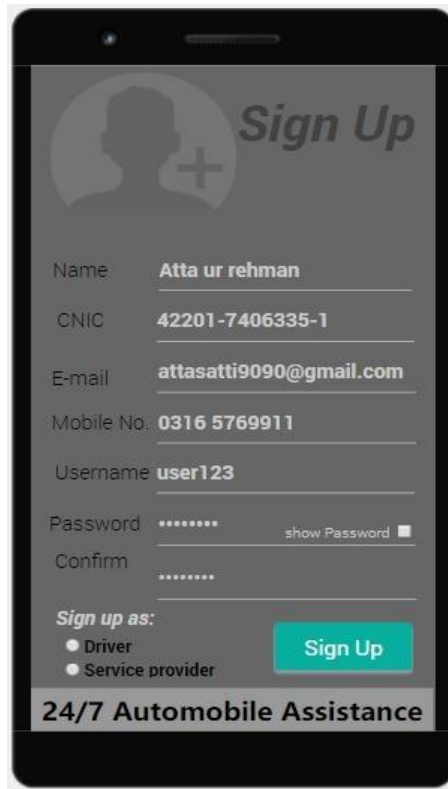


Figure 4.26 System Prototype (Signup Screen)

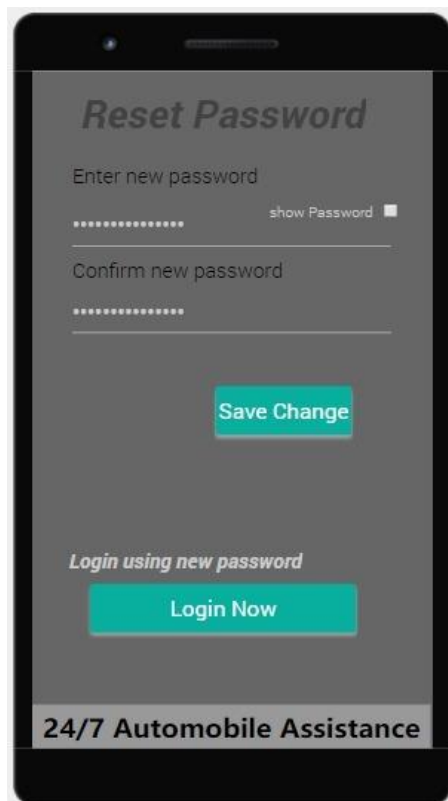


Figure 4.27 System Prototype (Reset Password)

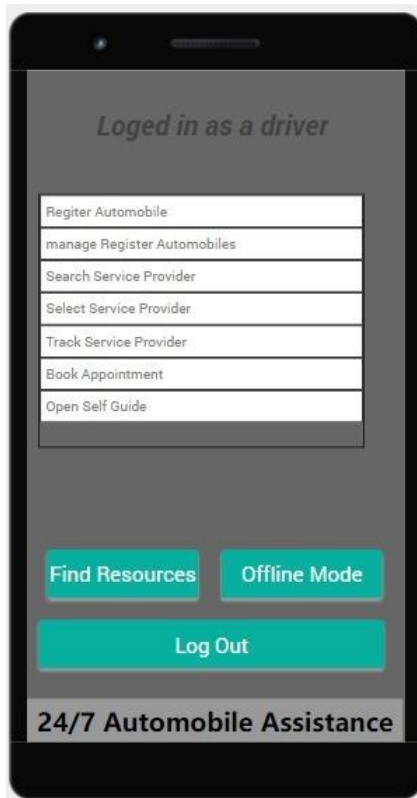


Figure 4.28 System Prototype (Driver's Main Screen)

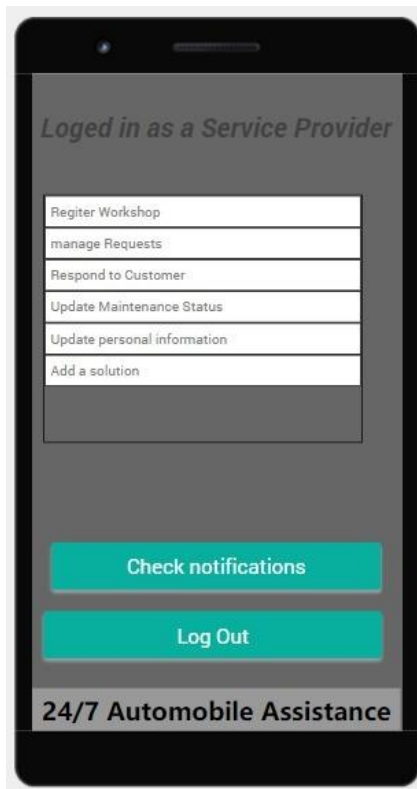


Figure 4.29 System Prototype (Service Provider's Main Screen)

4.7.2. User Interface Design

User Interface was designed on the basis of prototypes, however changes were made accordingly. UI includes the following screens.

4.7.2.1. Welcome/Login Screen:

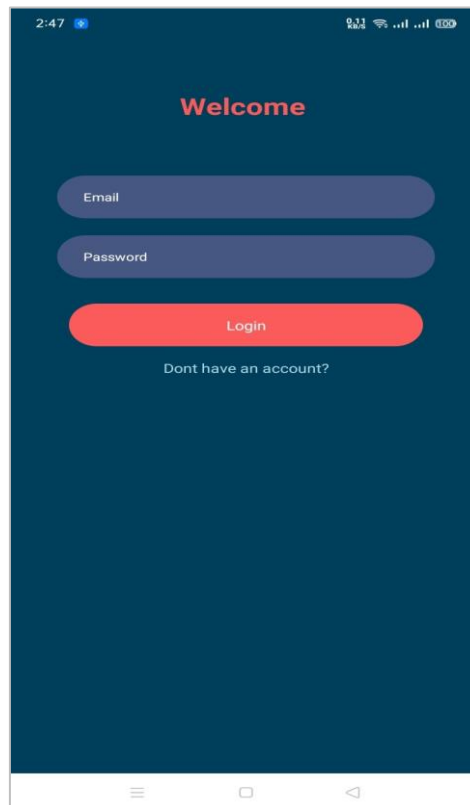


Figure 4.30 UI of Login Screen

When the user opens the app, the screen will be visible to the user (both driver and service provider) & he/she will be able to login to the system or a new user can create his/her account from the same screen.

4.7.2.2. Driver's Interface:

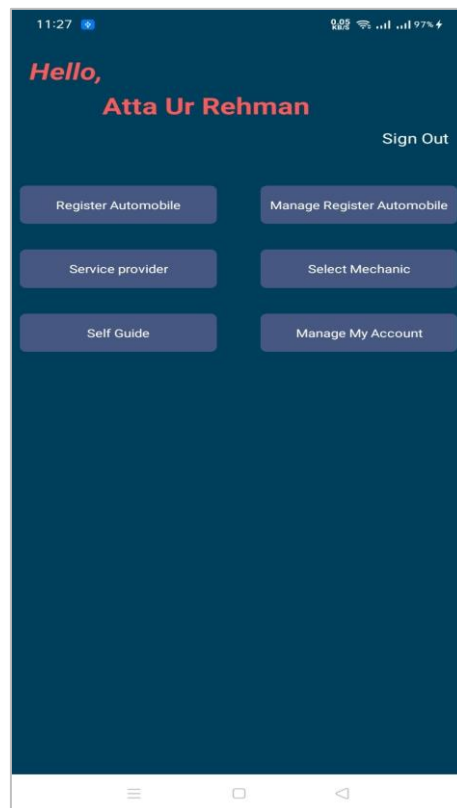


Figure 4.31 UI of Driver's Main Screen

After logging into the system, the system will navigate the driver to this screen, where he/she can perform different functions.

- Register Automobile
- Manage Registered Automobiles
- Find Resources/ service Providers
- Select Service Provider
- Self-guide
- Manage my account

The respective screens for each of driver's functionality is given below.

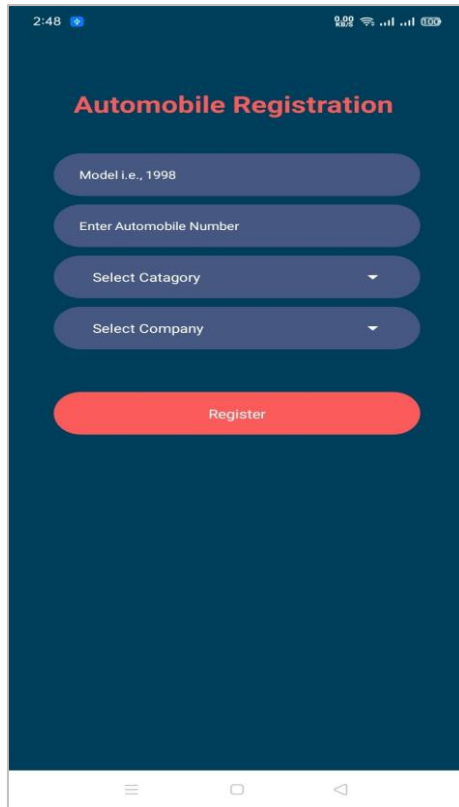


Figure 4.32 Automobile Registration UI

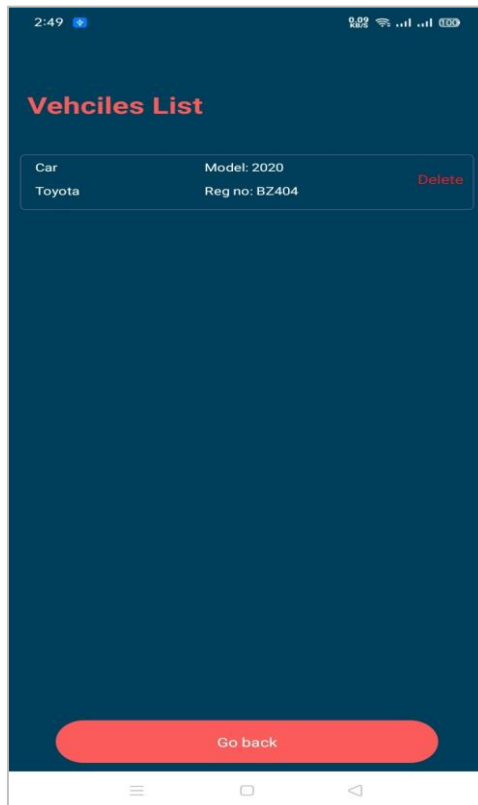


Figure 4.33 Manage Registered Automobiles UI

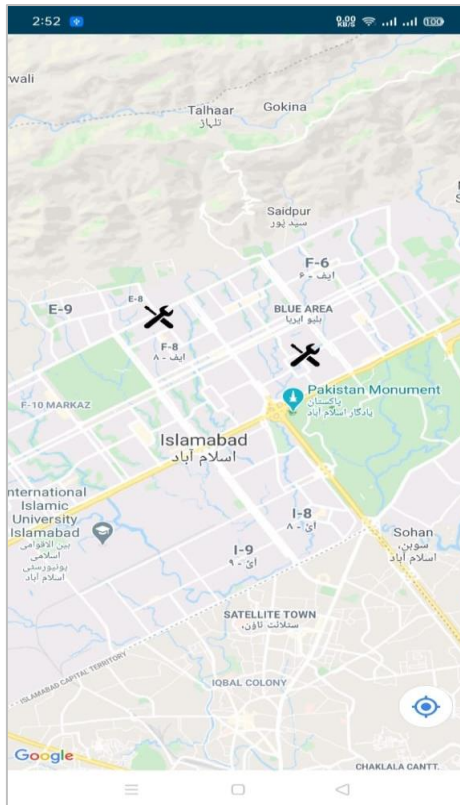


Figure 4.34 Find Resources/Service Providers UI

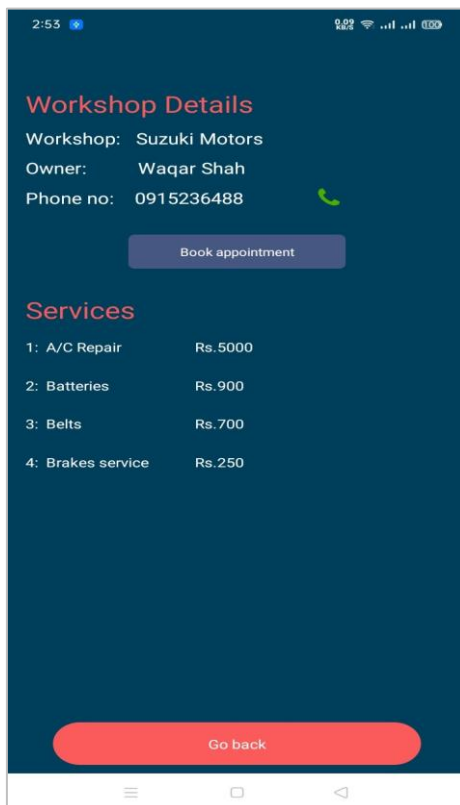


Figure 4.35 Select Service Provider UI

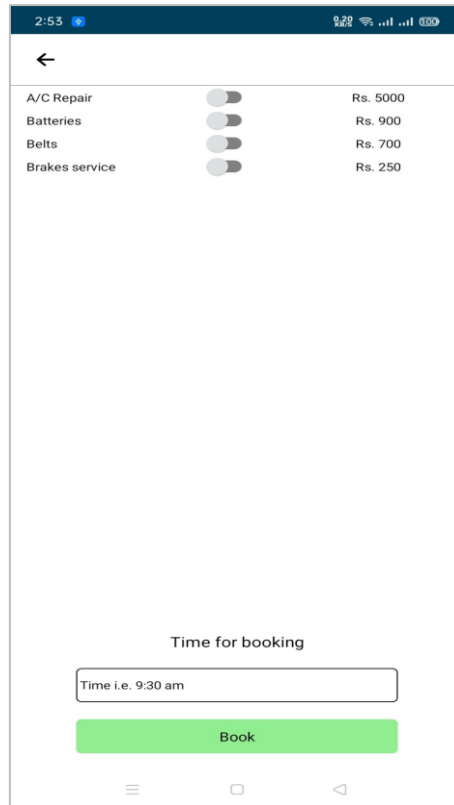


Figure 4.36 Book Appointment UI

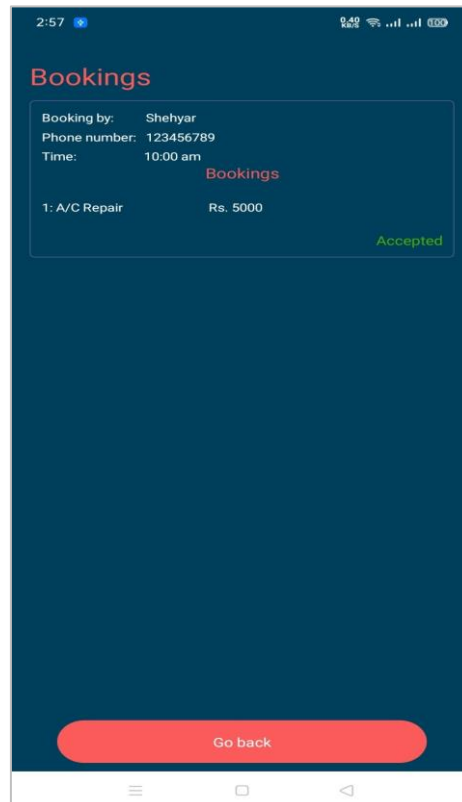


Figure 4.37 Check Request UI

4.7.2.3. Service Provider's Interface:

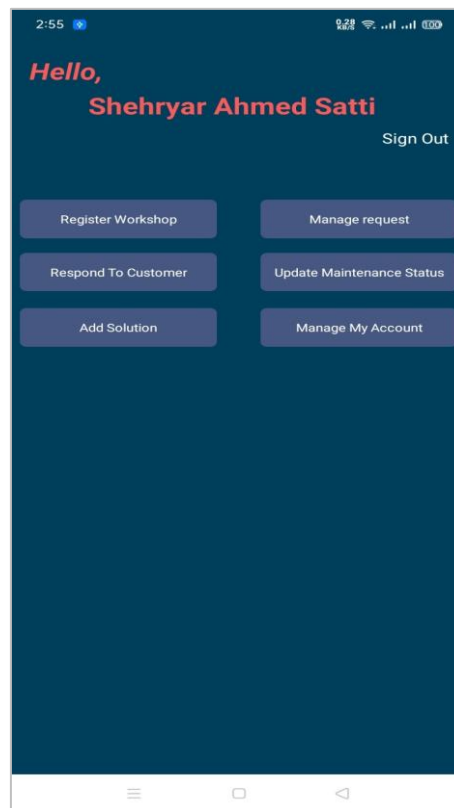


Figure 4.38 UI of Service Provider's Main Screen

After logging into the system, the system will navigate the service provider (mechanic) to this screen, where he/she can perform different functions.

- Register Workshop
- Manage Requests
- Respond to customers
- Update Maintenance Status
- Add a Solution
- Manage my account

The respective screens for each of service provider's functionality is given below.

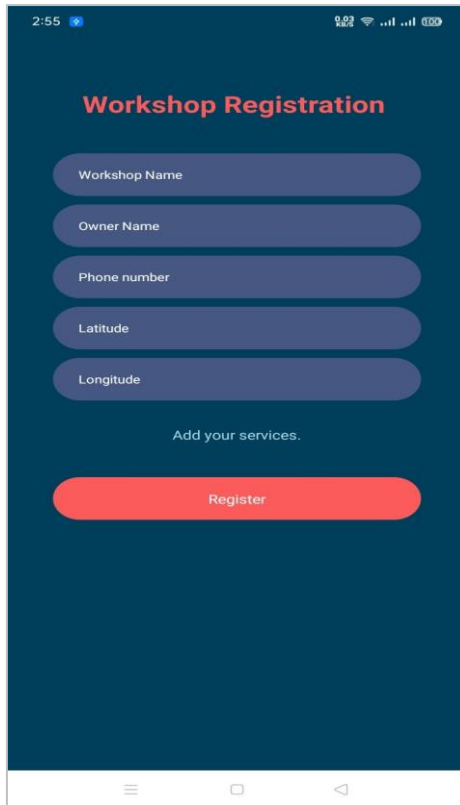


Figure 4.39 Register Workshop UI

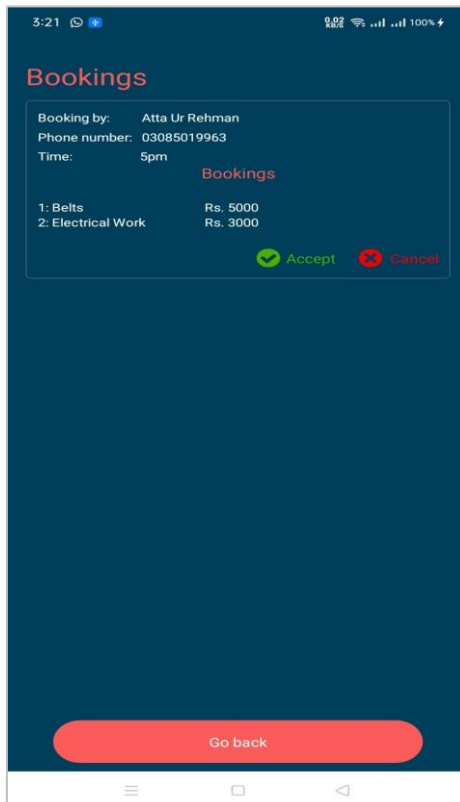


Figure 4.40 Manage Requests UI

Chapter 5

System Implementation

5.1. Strategy

As per requirements, the proposed system should be a cross platform system so we have Used React Native because it supports cross platform development. We have created the system based on actual data collected from the users (including Driver and the Service Providers). The main purpose of this system is to connect the two stakeholders i.e. drivers and service providers, increase the efficiency of assistance in case of vehicle breakdown and automatize the normal maintenance process.

5.2. Tools Used

To implement these functionalities and features, we have used following tools:

- Justin mind (for system prototype)
- Visual Studio Code
- Android Studio Emulator
- Node JS
- Firebase (Real-time database and Authentication)
- Framework for design and Development, we have used React Native

5.3. System Architecture

The system will use n-tier client server architecture. There will be client end and server end and both can be operated separately. There will be Presentation, Data and Logic tier in the architecture. System architecture is discussed in detail back in Chapter 4.

5.4. Steps in Implementation

Work breakdown structure and work breakdown dictionary are given in system design section. Major activities and steps of each phase are discussed here for more elaboration of each phase.

5.4.1. Requirement Gathering

The first and the most important step for any project or idea is to get the right information. Before starting the project, we had searched about the vehicle breakdown

and their recovery process. Then we looked into the market if there is already a system that offers service like on road maintenance or maintenance from home, we found no such systems which provide all the required functionalities.

Then we studied different systems who were a bit similar or had some common module with our system and started gathering requirements by analysing those existing systems, meeting the stakeholder and getting their point regarding what they want to be in the system and listed their input. After gathering complete information and requirements, we moved toward the development of project.

5.4.2. Software Requirements Specification (SRS) development

After getting complete requirements and performing analysis on requirements, the next step was SRS and SDS development. First of all we converted our gathered requirements into a use case diagram to shape it like a system, then the use case descriptions were added for each use case present in the use case diagram. Then the process of requirements completion was carried on to produce a Requirements Specification Document called SRS.

5.4.3. Software Design Specifications (SDS) development

After completion of software requirements Specification document, the next step was system modelling in which system's different views/models were developed. It ended as the development of Software Design Specifications document.

5.4.4. System prototypes and Front end development

With the help of SRs and SDS documents low fidelity prototype was developed by using just-in-mind. This was a mock-up system without any functionality that was developed only to demonstrate how the system would look like.

Low fidelity prototype was then converted into a high fidelity prototype which was actually the front end of the system and was developed by using java script in react native. So the front end or user interface of the system was completed and the system was ready to adopt main functionality.

5.4.5. Backend Development

The backend development of the system is also done using React Native, which was proposed at the start. We have used Firebase as the database to store and deal with the data.

5.5. Problems faced during development

During the process of system development there were several issue that we faced and managed to resolve those issues. Some of those are discussed below;

5.5.1. Selection of Framework

It was a challenge to choose a framework for cross platform development, as all the cross platform frameworks were new for us. Initially Xamarin was adopted due to the language that it uses but it was realized soon that due to less community of Xamarin there is no support available for its users and one often cannot find solution of very common problems.

Then we shifted towards React native as it has a vast community as well as it is more advanced and efficient as compared to Xamarin.

5.5.2. React Native Installation

In React Native we have faced a lot of difficulties in Installation of react Native app as it need some persecutive and their manual configuration (using command line interface) like Node, Java SE Development Kit (JDK), Android Studio and its components.

5.5.3. Database Connectivity and Authentication:

It was planned that MySQL will be used as database but it was very complex to connect and communicate with this database. Moreover the process of user authentication was very complex so it was decided to use firebase as database which is one of the most advanced and real time database.

5.6. Methodologies

24/7 Automobile Assistance System followed iterative development methodology. The system was built in chunks and each iteration added some new functionality to the system and the system was tested on the basis of functionality added in the new iteration. At the completion of a module it was tested as a whole and then the next module was developed in iteration in the same way like first one. Hence iteration after iteration resulted as a module and module after module when integrated together yielded as the complete system.

Chapter 6

System Testing & Evaluation

Testing is an important part of software development life cycle. It assures that the system fulfils the requirements specified in the SRS document and its outputs are valid. The aim of the testing was to find any bugs present at any level of the system in advance and fix those errors and bugs.

6.1. Test Strategy

The aim of the Test strategy is to document a methodology or pathway about how the system will be tested. It provides guidelines for effective testing of the system. Our strategy was to test the system from very basic level (unit Testing) test each function point and each statement in each unit and then move to higher level of the hierarchy by combining units to modules, and then testing the complete system at the end.

6.2. Unit Testing

Unit testing is the basic level of testing. Whenever a unit (feature) was developed it was tested separately to make sure that it is providing true results. Each of the unit was separately tested and only in case it performed successful result it was joined to its modules. All the units, features and functionalities of the system were tested successfully and were providing the same results as expected in the use case descriptions.

6.3. Component Testing

Components testing refers to perform testing on the components or modules of a system. A module consists of different units which are combined together. In 24/7 AA there were two basic modules of the system i.e. driver's module and service provider's module. Driver's module has further two sub-modules based on normal maintenance and on road vehicle breakdown. All the modules of the system were tested and were working properly both separately.

6.4. Integration Testing

All the separately working modules were integrated to each other and then tested again to assure if they are working properly after integration or not. After integrating

all the components Integration testing assured that those components are working collectively and providing true outputs.

6.5. System Testing

When all the modules were integrated and tested, a complete system emerged as a result of successful integration. All the integrated modules developed a system and the complete system was tested to make sure that the system as a whole is working properly.

6.6. Test cases

6.6.1. Test Case for Signup

Table 19: Test case (Sign up)

Test Scenario ID		AA-TC-001		Test Case ID		TC-001
Test Case Description		Signup		Test Priority		High
Pre-Requisite		Internet		Post Requisite		User registered
Test Execution Steps:						
S. No.	Action	Inputs	Expected Output	Actual Output	Test Result	Remarks
1	Triger signup	User click signup button	Signup page displayed to user	Same as Expected	Pass	positive
2	Triger signup	User input invalid or incomplete inputs	Error message displayed to users	Same as expected	Pass	Negative
3	Triger signup	User input valid and complete inputs	User is registered	Same as expected	Pass	Positive

6.6.2. Test Case for Login

Table 20: Test case (Log in)

Test Scenario ID		AA-TC-002		Test Case ID		TC-002
Test Case Description		Positive Test Case Login		Test Priority		High
Pre-Requisite		Portal Loaded Registered User		Post Requisite		User Logged In
Test Execution Steps:						
S. No.	Action	Inputs	Expected Output	Actual Output	Test Result	Remarks
1	Try to login with unregistered credentials	Enter unregistered credentials	Error message displayed to user	Same as Expected	Pass	Negative
2	Try to login with registered credentials	Enter registered credentials	Home page is displayed to user	Same as expected	Pass	Positive

6.6.3. Test Case for Register Automobile.

Table 21: Test case (Register Automobile)

Test Scenario ID		AA-TC-003		Test Case ID		TC-003
Test Case Description		Register Automobile		Test Priority		High
Pre-Requisite		User is already logged in		Post Requisite		Automobile registered
Test Execution Steps:						
S. No.	Action	Inputs	Expected Output	Actual Output	Test Result	Remarks
1	Register Automobile	User Click Register button	Vehicle registration form is displayed	Same as Expected	Pass	Positive test
2	Register Automobile	User click Register button	Vehicle registration form is displayed	Form does not display due to some broken links	Fail	Negative test
3	Register Automobile	User Enter valid inputs	Vehicle registered	Same as expected	Pass	Positive test
4	Register Automobile	User Enter valid inputs	Vehicle registered	Error message Vehicle already registered	Fail	Negative test
5	Register Automobile	User Enter invalid inputs	Error message displayed to user	Error message displayed to user	Pass	Negative test

6.6.4. Test Case for View Registered Vehicles

Table 22: Test case (View registered vehicles)

Test Scenario ID		AA-TC-004		Test Case ID		TC-004
Test Case Description		View Registered Vehicle		Test Priority		Low
Pre-Requisite		Vehicle must be registered		Post Requisite		List of Registered vehicle displayed
Test Execution Steps:						
S. No.	Action	Inputs	Expected Output	Actual Output	Test Result	Remarks
1	View Registered Vehicles	User clicks view registered vehicles	List of registered vehicles displayed	Same as Expected	Pass	Positive test
2	View Registered Vehicles	User clicks view registered vehicles	List of registered vehicles displayed	Display Data "No Data Found"	Fail	Negative test

6.6.5. Test Case for Edit vehicle details.

Table 23: Test case (Edit vehicle details)

Test Scenario ID		AA-TC-005		Test Case ID		TC-005
Test Case Description		Edit Vehicle Details		Test Priority		Medium
Pre-Requisite		Vehicle must be registered		Post Requisite		Vehicle details updated
Test Execution Steps:						
S. No.	Action	Inputs	Expected Output	Actual Output	Test Result	Remarks
1	Edit Vehicles Details	User clicks edit details button	Form with current details is displayed	Same as Expected	Pass	Positive test
2	Edit Vehicles Details	User clicks edit details button	Form with current details is displayed	Form do not displayed due to broken links	Fail	Negative test
3	Edit Vehicles Details	Users change the details	Details changed in database	Error message displayed due to database connection	Fail	Negative test
4	Edit Vehicles Details	Users change the details	Details changed in database	Data modified in database	Pass	Positive test

6.6.6. Test Case for Delete Vehicle.

Table 24: Test case (Delete vehicle)

Test Scenario ID		AA-TC-006		Test Case ID		TC-006
Test Case Description		Delete Vehicle		Test Priority		Low
Pre-Requisite		Vehicle must be registered already		Post Requisite		Vehicle deleted from database
Test Execution Steps:						
S. No.	Action	Inputs	Expected Output	Actual Output	Test Result	Remarks
1	Delete Vehicle	User clicks delete vehicle button	Vehicle deleted from database	Same as Expected	Pass	Positive test
2	Delete Vehicle	User clicks delete vehicle button	Vehicle deleted from database	Vehicle does not delete due to database connection	Fail	Negative test
3	Delete Vehicle	User clicks delete vehicle button	Vehicle deleted from database	No vehicle found in database	Fail	Negative test

6.6.7. Test Case for find resources.

Table 25: Test case (Find resources)

Test Scenario ID		AA-TC-007		Test Case ID		TC-007
Test Case Description		find resources		Test Priority		High
Pre-Requisite		Registered User logged in as driver		Post Requisite		Get the nearest workshops
Test Execution Steps:						
S. No.	Action	Inputs	Expected Output	Actual Output	Test Result	Remarks
1	Find Resources	Click on find resources button	System displays the available resources	Same as Expected	Pass	Positive test
2	Find Resources	Click on find resources button	System displays the available resources	No resources available	Fail	Negative test
3	Find Resources	Click on find resources button	System display “No resources found”	Same as expected	Pass	Negative test

6.6.8. Test Case for contact service provider.

Table 26: Test case (Contact service provider)

Test Scenario ID	AA-TC-009	Test Case ID	TC-009			
Test Case Description	Contact Service Provider	Test Priority	High			
Pre-Requisite	Registered User logged in as driver and already selected service provider	Post Requisite	Contact Service Provider			
Test Execution Steps:						
S. No.	Action	Inputs	Expected Output	Actual Output	Test Result	Remarks
1	Contact Service Provider	Click on Contact Service Provider button	System display necessary info about service provider	Same as Expected	Pass	Positive test
2	Contact Service Provider	Click on Contact Service Provider button	System display necessary info about service provider	No Data is available about service provider	Fail	Negative test
3	Contact Service Provider	Click on Contact Service Provider button	System display necessary info about service provider	Service provider out of reach	Fail	Negative test

6.6.9. Test Case for Open Self-Guide.

Table 27: Test case (Open self-guide)

Test Scenario ID		AA-TC-011		Test Case ID		TC-011
Test Case Description		Contact Service Provider		Test Priority		High
Pre-Requisite		Registered User logged in as driver		Post Requisite		Solution to the problem found
Test Execution Steps:						
S. No.	Action	Inputs	Expected Output	Actual Output	Test Result	Remarks
1	Self-Guide	Click on self-guide button	List of all the offline solution is displayed	Same as Expected	Pass	Positive test
2	Self-Guide	Click on self-guide button	List of all the offline solution is displayed	No Data is available offline	Fail	Negative test
3	Self-Guide	Click on self-guide button	User Do not find the solution for his problem	Same as Expected	Pass	Negative test
3	Self-Guide	Click on self-guide button	User Do not find the solution for his problem	Same as Expected	Pass	Negative test

6.6.10. Test Case for Booking appointment for maintenance

Table 28: Test case (Book appointment for maintenance)

Test Scenario ID		AA-TC-012			Test Case ID		TC-012		
Test Case Description		Contact Service Provider			Test Priority		High		
Pre-Requisite		Registered User logged in as driver			Post Requisite		Service Provider booked.		
Test Execution Steps:									
S. No.	Action	Input 1	Expected Output	Actual Output	Input 2	Expected Output	Actual Output	Test Result	Remarks
1	Appointment	clicks on Appointment button	Service Providers Displayed	Same as Expected	User Confirm Service provider	Service provider confirmed	Same as Expected	Pass	Positive test
2	Appointment	clicks on Appointment button	Service Providers Displayed	Same as Expected	User Confirm Service provider	Service provider confirmed	Service provider is not confirmed	Fail	Negative test
3	Appointment	clicks on Appointment button	Service Providers Displayed	Service Providers not Displayed	NIL	NIL	NIL	Fail	Negative test

6.6.11. Test Case for Check maintenance status.

Table 29: Test case (Check maintenance status)

Test Scenario ID		AA-TC-013		Test Case ID		TC-013
Test Case Description		Check maintenance status		Test Priority		Medium
Pre-Requisite		Registered User logged in as driver		Post Requisite		Get Maintenance Status of Car
Test Execution Steps:						
S. No.	Action	Inputs	Expected Output	Actual Output	Test Result	Remarks
1	Check maintenance status	Click on Check maintenance status button	Maintenance Status displayed	Same as Expected	Pass	Positive test
2	Check maintenance status	Click on Check maintenance status button	Message “No car for Maintenance”	Same as expected	Pass	Negative test
3	Check maintenance status	Click on Check maintenance status button	Maintenance Status displayed	No car for Maintenance	Fail	Negative test
3	Check maintenance status	Click on Check maintenance status button	Maintenance Status displayed	No Data displayed due to database error	Fail	Negative test

6.6.12. Test Case for Register workshop

Table 30: Test case (Register workshop)

Test Scenario ID		AA-TC-014						Test Case ID	TC-014
Test Case Description		Register workshop						Test Priority	High
Pre-Requisite		Registered User logged in as Mechanic						Post Requisite	Workshop Registered
Test Execution Steps:									
S. No.	Action	Input 1	Expected Output	Actual Output	Input 2	Expected Output	Actual Output	Test Result	Remarks
1	Register workshop	clicks on Register workshop button	Registration Form is displayed	Same as Expected	User fill the form	Workshop is registered	Same as Expected	Pass	Positive test
2	Register workshop	clicks on Register workshop button	Registration Form is displayed	Registration Form is not displayed	Nil	Nil	Nil	Fail	Negative test
3	Register workshop	clicks on Register workshop button	Registration Form is displayed	Registration Form is displayed	User fills the form	Workshop is registered	Workshop is not registered	Fail	Negative test

6.6.13. Test Case for Accept Request

Table 31: Test case (Accept request)

Test Scenario ID		AA-TC-015				Test Case ID		TC-015	
Test Case Description		Accept Request				Test Priority		High	
Pre-Requisite		Registered User logged in as Mechanic				Post Requisite		Notify Users about request	
Test Execution Steps:									
S. No.	Action	Input 1	Expected Output	Actual Output	Input 2	Expected Output	Actual Output	Test Result	Remarks
1	Accept Request	clicks on Check Request button	List of requests displayed	Same as Expected	User Accepts the request	User is notified	Same as Expected	Pass	Positive test
2	Accept Request	clicks on Check Request button	List of requests displayed	Same as Expected	User rejects the request	User is notified	Same as Expected	Pass	Positive test
3	Accept Request	clicks on Check Request button	List of requests displayed	List is not displayed	Nil	Nil	Nil	Fail	Negative test
4	Accept Request	clicks on Check Request button	List of requests displayed	Same as Expected	User Accepts the request	User is notified	User is not notified	Fail	Negative test
5	Accept Request	clicks on Check Request button	List of requests displayed	Same as Expected	User rejects the request	User is notified	User is not notified	Fail	Negative test

6.6.14. Test Case for Update maintenance status

Table 32: Test case (Update maintenance status)

Test Scenario ID	AA-TC-015						Test Case ID	TC-015	
Test Case Description	Update maintenance status						Test Priority	High	
Pre-Requisite	Registered User logged in as Mechanic						Post Requisite	Notify Users about request	
Test Execution Steps:									
S. No	Action	Input 1	Expected Output	Actual Output	Input 2	Expected Output	Actual Output	Test Result	Remarks
1	Update maintenance status	clicks on Update maintenance status button	Current Status is displayed	Same as Expected	User update the status	Status is updated	Same as Expected	Pass	Positive test
2	Update maintenance status	clicks on Update maintenance status button	Current Status is displayed	Same as Expected	User update the status	Status is updated	Status is not updated	Fail	Negative test
3	Update maintenance status	clicks on Update maintenance status button	Current Status is displayed	Current status is not displayed	Nil	Nil	Nil	Fail	Negative test

Chapter 7

Conclusion

This document is written to cover all the aspects of the 24/7 Automobile Assistance System including its all kinds of requirements and constraints. Our system will be a prototype to solve the problems in traditional automobile maintenance field. It will be an innovative step to modernize the auto mechanic industry. There are some advancements and modifications that can be further made into our system in future to enhance its productivity and efficiency which will be discussed in future work section.

7.1. Contributions

24/7 Automobile Assistance will make the maintenance process easy and connect the stakeholders to each other. It has analysed the problems faced by auto mechanic industry and proposes digital solutions for these problems. It will take the automobile maintenance industry towards a new way of technology. The system will act as a bridge between drivers and service providers and will minimize the gap between them. It will make the maintenance process more easy, safe and reliable for users. The system will fulfil the objectives that were introduced at the start of project to be met and solve the problems faced by the stakeholders of the system.

7.2. Reflections

This system can be used as a single system for a lot of issues of the industry unlike the previous systems. Previous systems include a single domain of industry e.g. on road breakdown or normal maintenance etc. however our system will be a single package to provide many solutions.

7.3. Future work

The system can be further enhanced and modified to add more functionalities that could not be added to the system by our group either due to lack of resources or any other reasons.

- Real time Tracking can be added to the system to make the maintenance process more efficient and accountable, however real time tracking can be

added to the project only if you are launching the system in market as a commercial system as all the API's used for tracking are paid.

- In app Payment module can be added to the system so that the system to increase the security and transparency of the whole process.
- The feature of Car Recovery in case of accident can also be added to the system in future so that the enterprises that provide Emergency Recovery Services can also be connected to the system and emergency recovery units can also be registered and operable along with workshops.

By adding above mentioned modules to the system its performance and market value can be increased and it can be made commercially more sustainable.

REFERENCES

1. K.J. Sheng, Ahmad Suhaimi Baharudin, Kamal Karkonasasi, "A Car Breakdown Service Station Locator System"
2. TransLink. 2019. Motorway Breakdown Response Service [online]. Available: <https://www.tmr.qld.gov.au/Travel-and-transport/Road-and-traffic-info/Incidents/Motorway-breakdowns>
3. Codeshoppy. 2020. On Road Vehicle Breakdown Assistance [online]. Available: <https://codeshoppy.com/shop/product/on-road-vehicle-breakdown-assistance-app/>
4. Allstate. 1993. 24-Hour Roadside Assistance [online]. Available: <https://www.allstate.com/roadside-assistance.aspx>
5. NevonSolutionsPvt.Ltd. "On Road Vehicle Breakdown Assistance Finder". Nevonprojects.com/on-road-assistance-finder. <https://nevonprojects.com/on-road-assistance-finder/>
6. Dr. Bhaskar Bhuyan, "Vehicle Breakdown Assistance", Bachelor's Project Report, Bachelor of Technology, (Information Technology), Sikkim Manipal University, Gangtok, India, 2020. [online]. Available: [B.Tech Project Report Of Vehicle Breakdown Assistance — Filemavr](#)
7. Professor Fazal Rehman Shamil, "Online Auto mechanic finder", t4tutorials, mianwali, Punjab, Pakistan, April 3rd, 2021. June 14th, 2021. [online], Available: <https://t4tutorials.com/online-auto-mechanic-finder-project-in-php-or-asp-net-srs-document>
8. I.H Dhanuka Nadeeshani, "On-Road Vehicle Breakdown Assistance", ResearchGate, Germany, June 2020, June 14th, 2021, [online], Available: [\(PDF\) On-Road vehicle Breakdown Assistance \(researchgate.net\)](#)