VISVESVARAYA TECHNOLOGICAL UNIVERSITY

BELAGAVI - 590 018, KARNATAKA.

MOBILE APPLICATION DEVELOPMENT
MINI PROJECT REPORT
ON

"ON ROAD VEHICLE BREAKDOWN HELP ASSISTANCE"

Submitted in the partial fulfillment of requirements

FOR

MOBILE APPLICATION DEVELOPMENT LABORATORY (18CSMP68)

Submitted by

ABHISHEK V	**4BD20CS004**
JEEVAN H K	**4BD20CS040**

PROJECT GUIDES:

Dr. SANTHOSH K C Ph.D.,
Associate Professor, Dept. of CS&E

Prof. RADHIKA PATIL M.Tech.,
Asst. Professor, Dept. of CS&E

2022-2023

Department of Computer Science and Engineering

Bapuji Institute of Engineering & Technology

Davanagere- 577004

Bapuji Institute of Engineering and Technology Davangere-577004

Department of Computer Science and Engineering

CERTIFICATE

This is to certify that **ABHISHEK V** and **JEEVAN H K** bearing USN **4BD20CS004, 4BD20CS040** respectively of **Computer Science and Engineering** department have satisfactorily submitted the mini project report entitled **"ON ROAD VEHICLE BREAKDOWN HELP ASSISTANCE"**. The report of the project has been approved as it satisfies the academic requirements in respect of project work prescribed for the academic year 2022-23.

Project Guide 1

Dr. SANTOSH K C Ph.D.,
Associate Professor
Department of CS&E,
B.I.E.T, Davanagere

Project Guide 2

Prof. RADHIK PATIL M.Tech.,
Asst. Professor
Department of CS&E,
B.I.E.T, Davanagere

Head of Department

Dr. Nirmala C.R Ph.D.
Prof. & Head,
Department of CS&E,
B.I.E.T, Davanagere

Date:

Place: Davanagere

Signature of Examiners:

(1) ____________________

(2) ____________________

ACKNOWLEDGEMENT

Salutations to our beloved and highly esteemed institute**, "BAPUJI INSTITUTE OF ENGINEERING AND TECHNOLOGY"** for having well qualified staff and lab furnished with necessary equipment's.

We express our sincere thanks to our guides **Dr. Santhosh K C** for giving us constant encouragement, support and valuable guidance throughout the course of project without whose guidance this project would not have been achieved.

We express whole hearted gratitude to **Dr. Nirmala C R,** H.O.D of Computer Science and Engineering Department. We wish to thank her for making our task easy by providing her valuable help and encouragement.

We also express our whole hearted gratitude to our principal, **Dr. H B Aravind** for his moral support and encouragement.

We would like to extend our gratitude to all the staff of **Computer Science and Engineering Department** for their help and support. We have benefited a lot from the feedback, suggestions given by them.

We would like to extend our gratitude to all our family members and friends for their advice and moral support.

Abhishek V(4BD20CS004)
Jeevan H K(4BD20CS040)

Vision and Mission of the Institute

Vision

“To be a center of excellence recognized nationally and internationally, in distinctive areas of engineering education and research, based on a culture of innovation and invention.”

Mission

“BIET contributes to the growth and development of its students by imparting a broad based engineering education and empowering them to be successful in their chosen field by inculcating in them positive approach, leadership qualities and ethical values.”

Vision and Mission of the Computer Science and Engineering Department

Vision

“To be a center-of-excellence by imbibing state-of-the-art technology in the field of Computer Science and Engineering, thereby enabling students to excel professionally and be ethical.”

Mission

1. Adapting best teaching and learning techniques that cultivates Questioning and Reasoning culture among the students.
2. Creating collaborative learning environment that ignites the critical thinking in students and leading to the innovation.
3. Establishing Industry Institute relationship to bridge the skill gap and make them industry ready and relevant.
4. Mentoring students to be socially responsible by inculcating ethical and moral values.

Program Educational Outcomes (PEOs):

PEO1	To apply skills acquired in the discipline of Computer Science and Engineering for solving societal and industrial problems with apt technology intervention.
PEO2	To continue their career in industry/academia or to pursue higher studies and research.
PEO3	To become successful entrepreneurs, innovators to design and develop software products and services that meets the societal, technical and business challenges.
PEO4	To work in the diversified environment by acquiring leadership qualities with effective communication skills accompanied by professional and ethical values.

Program Specific Outcomes (PSOs)

PSO1	Analyze and develop solutions for problems that are complex in nature by applying the knowledge acquired from the core subjects of this program.
PSO2	Ability to develop Secure, Scalable, Resilient and distributed applications for industry and societal requirements.
PSO3	Ability to learn and apply the concepts and construct of emerging technologies like Artificial Intelligence, Machine learning, Deep learning, Big Data Analytics, IoT, Cloud Computing, etc for any real time problems.

Course Outcomes (COs)

CO1	Create, test and debug Android application by setting up Android development environment.
CO2	Implement adaptive, responsive user interfaces that work across a wide range of devices.
CO3	Infer long running tasks and background work in Android applications.
CO4	Demonstrate methods in storing, sharing and retrieving data in Android applications.
CO5	Infer the role of permissions and security for Android applications.

ABSTRACT

On Road Vehicle Breakdown Help Assistance goes to be an honest solution for the people who seek help within the remote locations with mechanical problems with their vehicle. it'll be the registered public and that they are getting connected with the actual mechanic through the trustworthy application system. In an existing system there are users who have their own mechanic database which is incredibly minimal. And also, they need no idea if their vehicles are breaking down or had any mechanical issue in remote locations or any long distant locations from their known mechanic shops. in a very proposed Here the users of On Road Vehicle Breakdown Assistance (ORVBA) system can hunt for list of mechanic at any location or the nearby locations which can help them in an unexpected situation raised by the mechanical problems with their vehicles.

CONTENTS

TOPIC **PAGE NO**

CHAPTER 1: INTRODUCTION

CHAPTER 2: REQUIREMENT SPECIFICATIONS

CHAPTER 3: DESIGN

CHAPTER 4: IMPLEMENTATION

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

1.1 ANDROID STUDIO

Android Studio is an integrated development environment (IDE) for the Google Android Operating System. It is built based on JetBrains' IntelliJ IDEA Community Edition, and it is specifically designed for creating applications on Android devices. Some of the key features of Android Studio are as follows:

- **Instant Run** – a feature that pushes code and resource changes to the running app. It allows changes to be made to the app without the need to restart the app, or rebuilding theAPK, so that the effects can be seen instantly.
- **An Emulator** – a virtual android device that can simulate a variety of hardware features such as GPS location, network latency, motion sensors, and multi-touch input that can be used to run and install the app. It can then be used for testing purposes.
- **Testing Tools and Frameworks** – extensive testing tools such as JUnit 4 and functional UI test frameworks are included with Android Studio. Espresso Test Recorder can generate UI test code by recording the developer's interactions with the app on a deviceor emulator. The tests can be run on a device, an emulator, in a Firebase Test Lab, or on a continuous integration environment.

1.1.1 ARCHITECTURE OF ANDROID STUDIO

Android architecture contains different number of components to support any android device needs. Android software contains an open-source Linux Kernel having collection of number of C/C++ libraries which are exposed through an application framework service. Among all the components Linux Kernel provides main functionality of operating system functions to smartphones and Dalvik Virtual Machine (DVM) provide platform for running an android application.

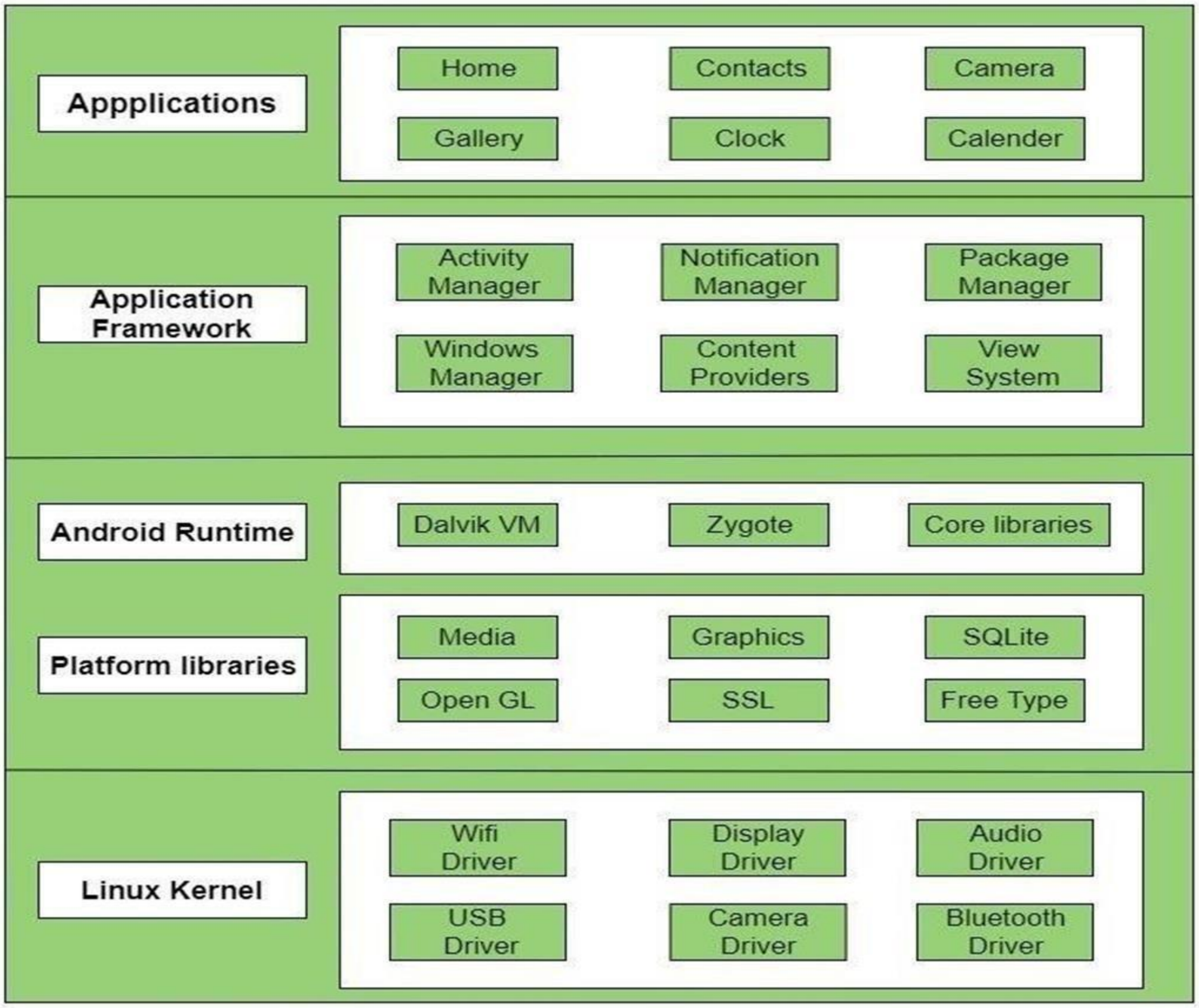


Fig1.1.1. Android Architecture

1.1.2 INSTALLING AND RUNNING APPLICATIONS ON ANDROIDSTUDIO

1.1.2.1 SYSTEM REQUIREMENTS

- OS: Windows 8/8.1/10/11 (64-bit)
- CPU: 2nd generation Intel CPU (Sandy Bridge) or newer, AMD CPU with support for a Windows Hypervisor
- Memory: 8 GB RAM
- Free storage: 8 GB
- Screen resolution: 1280 x 800

1.1.2.2 SET UP OF ANDROID STUDIO

To set up Android Studio, you can follow these steps:
Download Android Studio: Go to the official Android Studio website (https://developer.android.com/studio) and download the latest version of Android Studio for your operating system (Windows, macOS, or Linux). Install Android Studio: Once the download is complete, run the installer and follow the on-screen instructions to install Android Studio on your computer. The installation process may take a few minutes.Configure Android SDK: After installation, launch Android Studio. On the welcome screen, select "Configure" and then choose "SDK Manager." The SDK Manager allows you to download the necessary Android SDK components for development.

a. In the SDK Platforms tab, select the Android versions you want to target with your app. It's recommended to choose the latest stable version as well as the minimum version you want to support.
b. In the SDK Tools tab, select the components you need, such as the Android Emulator, Android SDK Build-Tools, and others. Again, it's recommended to use the latest stable versions.
c. Click "Apply" to start downloading and installing the selected SDK components.

1.1.2.3 CREATE ANDRIOD VIRTUAL DEVICE

- Once the SDK installation is complete, click on the "AVD Manager" icon on the toolbar or go to "Tools" > "AVD Manager."
- Click on "Create Virtual Device."
- Choose a device definition that matches your testing requirements and "Next."
- Select a system image for the virtual device. It's recommended to choose the image with the latest API level. Click "Next."
- Configure any additional settings for the virtual device, such as RAM and storage, or leave them as default. Click "Finish" to create the virtual device.

1.2 INTRODUCTION TO ON ROAD VEHICLE BREAKDOWN HELP ASSISTANCE

The On-Road Vehicle Breakdown Help Assistance Project aims to provide efficient and reliable support to motorists who experience unexpected breakdowns or emergencies while traveling. Vehicle breakdowns can occur at any time and can leave drivers stranded on highways, city roads, or remote areas, leading to inconvenience, safety risks, and delays. This project seeks to address these challenges by establishing a comprehensive assistance system that swiftly responds to breakdown situations, ensuring the safety and well-being of motorists.

The primary objective of the On-Road Vehicle Breakdown Help Assistance Project is to offer immediate aid and support to distressed drivers in need. This assistance will encompass a wide range of services, including mechanical repairs, towing services, emergency fuel delivery, flat tire replacement, jump-starting dead batteries, and other necessary on-site repairs. The project will also prioritize the safety of motorists by promptly alerting the appropriate authorities in case of severe accidents or medical emergencies.

To implement this project effectively, a robust infrastructure will be established, comprising of dedicated call centres equipped with well-trained operators who can efficiently manage breakdown requests. These call centres will operate around the clock, ensuring assistance is available 24/7, irrespective of the time or location of the breakdown. In addition to phone-based assistance, a mobile application will be developed, enabling motorists to request help quickly and track the progress of their assistance in real-time.

The On-Road Vehicle Breakdown Help Assistance Project will collaborate with towing companies, mechanics, and other service providers to ensure a wide coverage area and a rapid response time. Strategic partnerships with local authorities, traffic control centres, and emergency services will also be established to facilitate a coordinated approach in managing breakdown incidents.

1.3 HISTORY

Before the advent of the internet, currency conversion was primarily done through manual calculations or by referring to printed currency exchange rate tables. These tables were often published in newspapers, travel guides, or available at banks and currency exchange counters. With the rise of the internet in the late 1990s and early 2000s, online currency converters began to appear. Websites like XE.com (established in 1993 as Xenon Laboratories) and OANDA (founded in 1996) provided online tools to calculate currency conversions. These early converters allowed users to input specific amounts and select the desired currencies to get real-time exchange rates. The advent of smartphones and mobile applications brought currency conversion tools to handheld devices. As mobile internet access became more widespread, various currency converter apps emerged. These apps provided convenience and portability, allowing users to convert currencies on the go. As the popularity of smartphones grew, currency converters started to be integrated into broader finance and travel apps. Many banking and financial institution apps included currency conversion features as part of their services. Travel apps also incorporated currency converters to assist travelers in managing their expenses and budgets. Over time, currency converter apps have become more sophisticated in terms of design, functionality, and user experience. They offer features like historical exchange rate charts, multiple currency conversions, offline access, and notifications for rate updates. Some apps even include additional features such as expense tracking and budget management. Recent advancements in artificial intelligence (AI) and machine learning (ML) have enabled currency converter apps to provide more accurate and personalized exchange rate predictions. These technologies analyze historical data, market trends, and other factors to improve the accuracy of exchange rate calculations.

1.4 OBJECTIVES

Vehicle Break Down Help Assistant is an android application that helps you locate the nearest mechanic and book them in case a vehicle breaks down. No matter how distant you are from the city, all you have to do is enter your car information, and the app will display a list of available mechanics who can assist.

To achieve the above demands we have framed the following objectives:

- To build a common platform that connect with mechanical and driver.
- To design an app that can identify a mechanic from the nearest location.
- To provide on-demand fuel delivery.
- To develop an application for driver and passengers to provide a efficient assistance in case of a vehicle break down

1.5 IMPORTANT FEATURES

1. Incorporating Search Feature

Incorporate the advanced search feature for mobile app users that helps them find and scour for the content they need. This adds a huge impact, especially if the app has a vast database of content; it quickly becomes cumbersome for users to access easily. With the advanced search feature and the search box, top-tier apps allow users to search for relevant content with the help of keywords and key phrases. The search feature makes the app content accessible to all kinds of users.

2. Incorporate Social Media Integration

Social media sharing is not restricted to picture sharing anymore. Most apps now include social media integration so that users can communicate and collaborate with other people, irrespective of whether they have the same app or not. App owners also reap the advantage of having social media on their apps since more users sharing their stuff will, in turn, impact their reach and growth positively Making social media integration seamless is the key here with social sharing buttons directly at the side of every app page.

3. Use of Responsive App Design

Mobile app design based on every screen size and resolution is quite robust. Responsive app design resolves this problem helping the app acclimatize to any screen size and resolution of small smartphones as well as larger tablets. Some smartphones also feature screens that hover around the size of a tablet. Check out the newest iPhones and its competitors in the market; they are large enough to be called mini-tablets. By incorporating responsive app design during app development, users will experience the app in the best light irrespective of the device they are using.

4. Simplicity in Design with Uncluttered Elements

The mobile app design has now evolved into a smart aesthetic exercise where proper use of white space is quite essential. Irrespective of the logic complexity of the app, the user experience design is expected to be uncluttered and straightforward with the elements placed in proper context, space, and size. Mobile app users are always keen to use apps that are easy to-use and can be navigated without a problem. This is where a simple UI/UX design can help companies describe the navigation structure, workflow, and principles of the app succinctly to its intended audience.

1.6 ADVANTAGES OF ANDROID STUDIO

. • **Android is an open-source platform allowing UI customization**

Licensed under Apache, Android is an open-source operating system whose code developers can change to build customized User Interfaces. App developers building applications for this platform can get access to the core codes and are at a liberty to change the theme to get the customized outcomes. This is not possible when it comes to iOS and apps have to strictly adhere to the core code specifications when developing apps for the specific platform.

- **Supports cloud storage enabling sync of devices with G-account**

Since Android is a Google product, users having a Gmail account can have access to cloud storage that is supported by the tech company. This means that users can sync devices using Google accounts. Moreover, Google gives 15GB of free cloud storage to every user, which is good for an average person using it for personal purposes.

- **Continual improvement & removal of old features**

Google Android is supported by a huge community of developers and also users who continue to give feedback about the features, their pros and cons. As a result, there is continuous check on the codes and features, making modifications and alterations, bringing in better upgrades all the time. This is one of the reasons why Android is always adding new features while removing older ones that users do not like.

- **Supports 3rd party widget & information display on screen**

Android gives users the freedom to download 3rd party widgets and also display their content ont he home screen. If a user wishes to view time and temperature shown by a specific widget on the home screen, it is possible with devices running on the platform.

- **Supports running multiple apps simultaneously**

With Android running on a device with good hardware specification, as a user you can have multiple apps running simultaneously. You can continue to listen to music as you check your messages or download files that you've received or even upload them from your device or drive. There are a lot of Android app development companies who build applications based on Android that are very useful in our daily lives.

- **Expandable memory & runs on affordable large devices**

One of the biggest advantages of using devices running on the Android platform is that it supports expandable memory. iOS devices on the other hand do not support external memory expansion by adding a memory card to the phone. Users of this platform enjoy the privilege of storing e- books, music, videos and games on their devices.

CHAPTER 2

REQUIREMENT SPECIFICATION

2.1 Functional Requirements

- **Registration and User Profile:** User registration and login functionality. Profile creation and management, including personal details and vehicle information.
- **Breakdown Reporting:** Allow users to report vehicle breakdowns through the application. Capture essential information about the breakdown, including location, vehicle details, and breakdown symptoms.
- **Location Tracking and Mapping:** Integrate GPS functionality to track the user's location in real-time. Display the user's location on a map to assist service providers in reaching the breakdown site efficiently.
- **Service Provider Matching:** Implement a system to match the user's breakdown request with nearby service providers. Consider factors like service availability, proximity, and specialization of service providers.
- **Service Request and Confirmation:** Enable users to send service requests to the matched service providers. Provide a mechanism for service providers to accept or decline service requests.
- **Communication:** Enable secure and real-time communication between users and service providers through chat or call functionalities. Ensure privacy and encryption of communication channels.
- **Feedback and Rating System:** Implement a feedback and rating system to collect user reviews about service providers. Enable users to rate the quality and timeliness of the provided services.

2.2 Non-functional Requirements

- **Performance and Scalability:** Ensure the application performs well under different user loads and scale as the user base grows.
- **Security and Privacy:** Implement robust security measures to protect user data and prevent unauthorized access. Comply with relevant data protection and privacy regulations.
- **Compatibility and Platform Support:** Develop the application to be compatible with popular mobile platforms, such as iOS and Android. Optimize the application for various screen sizes and resolutions.
- **Reliability and Availability:** Ensure the application is reliable and available for use 24/7, minimizing downtime and service disruptions.
- **Usability and User Experience:** Design an intuitive and user-friendly interface, considering the diverse user base and their varying technological expertise.

2.3 Software Requirements

SOFTWARE REQUIREMENTS	
Application Required	Android studio Application (Chipmunk)
Windows	Windows 10 and Above
Language Used	Java, XML
Emulator	Pixel 3 XL, Pixel 4a, Pixel 5, Pixel 6

2.4 Hardware Requirements

HARDWARE REQUIREMENTS (MINIMUM)	
RAM Memory	8GB
Processor	Intel® Core (TM) i3-1005G1 CPU @ 1.20GHz 1.19 GH
System Type	64-bit operating system, x64-based processor
Mobile Device	Any android mobile(Compatible with android studio)
Screen Resolution	1920 * 1080

CHAPTER 3

SYSTEM DESIGN

3.1 Initialization

To initialize and design the Photo Management app, you can follow these steps:

Step 1: Identify the main components of the app:

- MainActivity: This is the entry point of the app and handles the user interface.
- XML layout files: Design the user interface using XML layout files, such as activity_main.xml for MainActivity and activity_register_number.xml for RegisterNumberActivity.

Step 2: Create the necessary classes and XML layout files:

- Create MainActivity.java class and corresponding activity_main.xml layout file.

Step 3: Define the activities and services in the AndroidManifest.xml file:

- Register the MainActivity as activities.

Step 4: Design the user interface in the XML layout files:

- Customize the activity_main.xml files to include the necessary views, buttons, text fields, and any other components required for your app's design.

Step 5: Implement the logic and functionality in the Java classes:

- In MainActivity, implement the methods for starting and stopping the service, handling user input, displaying the tags, and opening the popup menu.

Step 6: Implement necessary permissions and dependencies:

- Ensure that the necessary permissions, such as ACCESS_MEDIA, are declared in the AndroidManifest.xml file.
- Add any required dependencies to the build.gradle file.

Step 7: Customize the app's design and functionality:

- Modify the code and XML layouts as per your specific requirements. You can customize the UI design, add additional features, or integrate with external APIs for location tracking or emergency contact notifications.

Step 8: Build and run the app:

- Build the app and run it on an emulator or a physical device to test the initialization and Display.

3.2 Display

To display the On Road Vehicle Break Down Help Assistance app to the user, you need to design a user-friendly interface that allows users to perform the following actions:

- **Login Page And Registration page:** user have to login to use the application .For login first user has to register.
- **Side Bar Page:** The sidebar consist of a Profile info ,Dashboard and About section in the profile section user has to create their profile by filling the information and dashboard consist of online help like YouTube videos and others .
- **Need Help Button Page:** Homepage having a need help button by clicking the button. The options of the service stations is displayed then user can select, view and send the request help to the service station.
- **Request Cancellation Page:** user can cancel the request while the request is pending which was not accepted by the service station.
- **Contacting Page:** user can contact the service station through the cell phone number which was given by the service station.
- **Vehicle Owner Request Page:** service station owners can accept the request or cancel the request and they can view the vehicle owners request**.**

3.3 Flowchart

The following flowchart shows the implementation of “ On Road Vehicle Breakdown Help Assistance”.

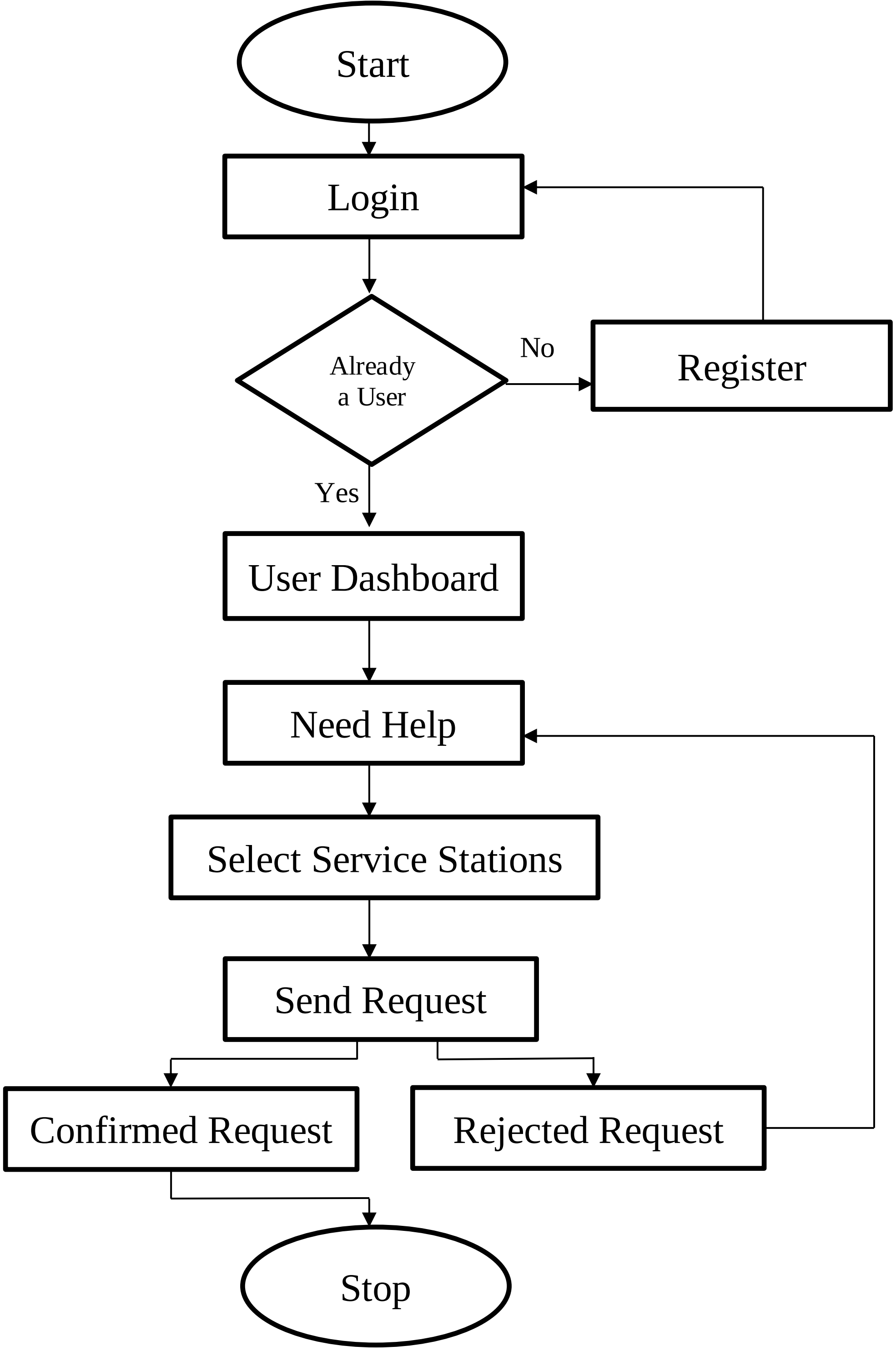


Fig. 3.3.1 Flowchart of Vehicle Owner

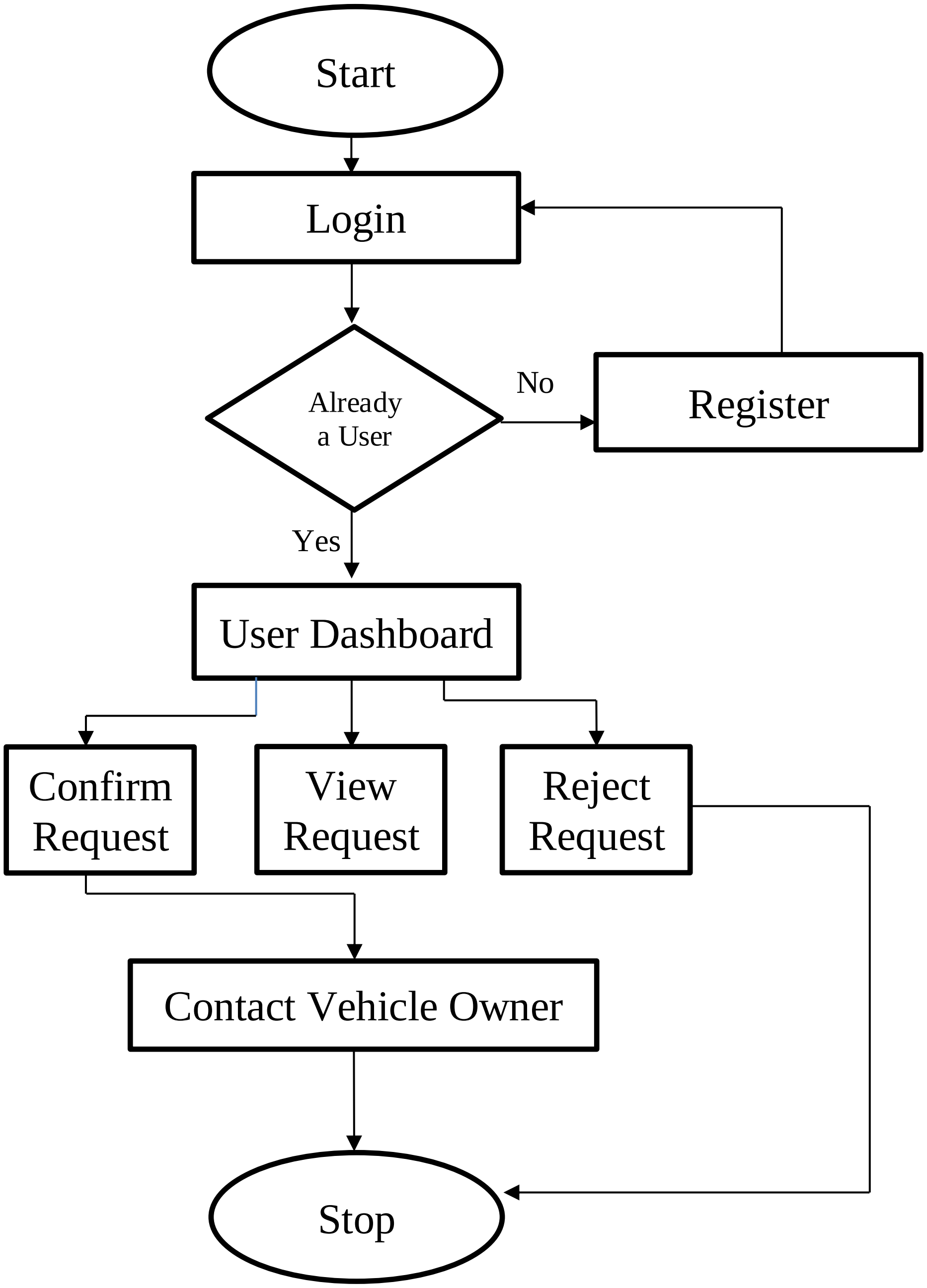


Fig. 3.3.1 Flowchart of Service Station

CHAPTER 4

IMPLEMENTATION

4.1 OVERVIEW

This project is a demonstration of mobile application on" On Road Vehicle Breakdown Help Assistance ". We have taken the helpof built-in functions present in the header file. To provide functionality to our project we have written sub functions. These functions provide us the efficient way to design the project. In this chapter we are describing the functionality of our project using these functions.

4.2 CODE

4.2.1 Main Activity .java

```
package com.example.loginapp;


import android.app.ProgressDialog;
import android.content.Intent;
import androidx.annotation.NonNull;
import androidx.appcompat.app.AppCompatActivity;
import android.os.Bundle;
import android.text.TextUtils;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;
import android.widget.Toast;
import com.google.android.gms.tasks.OnCompleteListener;
import com.google.android.gms.tasks.Task;
import com.google.firebase.auth.AuthResult;
import com.google.firebase.auth.FirebaseAuth;
import com.google.firebase.auth.FirebaseUser;

 public class MainActivity extends AppCompatActivity {

 private Button signupbutton;
 private EditText emaileditText;
 private EditText pweditText;
 private TextView signuptextView;
 private ProgressDialog progressDialog;

 private FirebaseAuth firebaseAuth;

 @Override
 protected void onCreate(Bundle savedInstanceState) {
     super.onCreate(savedInstanceState);
     setContentView(R.layout.activity_main);
     signupbutton=(Button)findViewById(R.id.signupbutton);
     emaileditText=(EditText)findViewById(R.id.emaileditText1);
     pweditText=(EditText)findViewById(R.id.pweditText1);
     signuptextView=(TextView)findViewById(R.id.signuptextView);
     progressDialog=new ProgressDialog(this);

     firebaseAuth=FirebaseAuth.getInstance();

     Intent gotoVOMap=new Intent(getApplicationContext(),VOMapActivity.class);
     if(firebaseAuth.getCurrentUser() !=null){
         finish();

            startActivity(gotoVOMap);

         }
```

```
        signupbutton.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                registerUser();
            }
        });
        signuptextView.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                 finish();
                Intent gotosignin=new Intent(getApplicationContext(),LoginActivity.class);
                startActivity(gotosignin);
            }
        });
    }
    private void registerUser(){
        String email=emaileditText.getText().toString().trim();
        String password=pweditText.getText().toString().trim();

        if(TextUtils.isEmpty(email)){


            Toast.makeText(this,"You cannot leave email field
empty",Toast.LENGTH_SHORT).show();
            return;
        }
        if(TextUtils.isEmpty(password)){
            Toast.makeText(this,"You cannot leave password field
empty",Toast.LENGTH_SHORT).show();
            return;
        }
        progressDialog.setMessage("Registering the user");
        progressDialog.show();

firebaseAuth.createUserWithEmailAndPassword(email,password).addOnCompleteListener(this,
new OnCompleteListener<AuthResult>() {
            @Override
            public void onComplete(@NonNull Task<AuthResult> task) {

                if(task.isSuccessful()){
                    sendEmailVerification();
                    progressDialog.dismiss();
                }
                else {
                    Toast.makeText(MainActivity.this,"Registering is not
 Successfull",Toast.LENGTH_SHORT).show();
                    progressDialog.dismiss();
                }
            }
        });
    }
    private void sendEmailVerification(){

        FirebaseUser firebaseUser=firebaseAuth.getCurrentUser();

        if (firebaseUser!=null){

            firebaseUser.sendEmailVerification().addOnCompleteListener(new
 OnCompleteListener<Void>() {
                @Override
                public void onComplete(@NonNull Task<Void> task) {

                    if (task.isSuccessful()){
```

```
 Toast.makeText(MainActivity.this,"Successfully Registered, Verification mail has been
 sent",Toast.LENGTH_SHORT).show();
                          Intent gotogetinfo=new Intent(getApplicationContext(),UserInfo.class);
                          finish();
                          startActivity(gotogetinfo);
                     }
                     else{
                         Toast.makeText(MainActivity.this,"Verification mail hasn't been
sent",Toast.LENGTH_SHORT).show();
                     }
                 }
             });
         }
     }
 }
```

CHAPTER 5

TESTING

The testing of a On Road Vehicle Breakdown Help Assistance Application typically involves various aspects to ensure its functionality, accuracy, and reliability. Here are some common types of testing that can be performed:

- **Performance Testing:** Outline the performance testing approach and tools used to evaluate the application's performance under different loads and scenarios. Present the performance test results, including response times, resource utilization, and system stability.

- **Usability Testing:** Describe the usability testing process, including the selection of representative users and test scenarios. Present the findings and user feedback regarding the application's ease of use, navigation, and overall user experience.

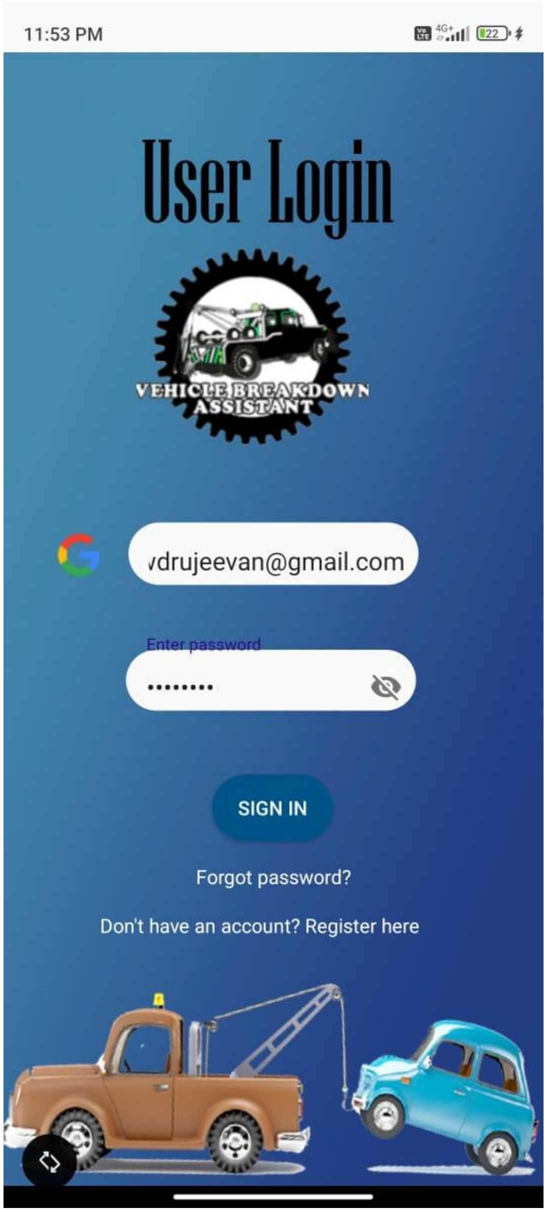
- **Security Testing:** Explain the security testing methodologies and tools used to identify vulnerabilities and ensure data protection. Document any security issues identified and the steps taken to mitigate them

- **Test Coverage and Compliance:** Assess the overall test coverage achieved during the testing phase. Verify compliance with relevant standards, regulations, and best practices.

CHAPTER 6

SNAPSHOTS

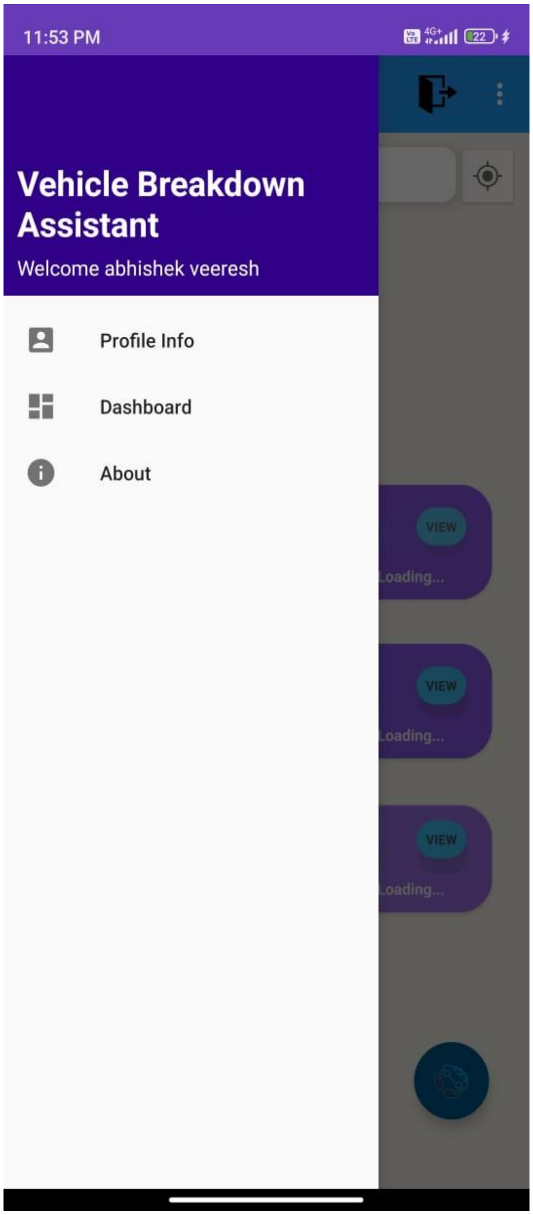
➢ VEHICLE OWNER PAGE:


11:53 PM
User Login
VEHICLE BREAKDOWN ASSISTANT
Enter password
SIGN IN
Forgot password?
Don't have an account? Register here

vdrujeevan@gmail.com

••••••••

Fig 6.1.Login Page.



Welcome abhishek veeresh

Fig.6.2.Vehicle Owner Side Bar Menu

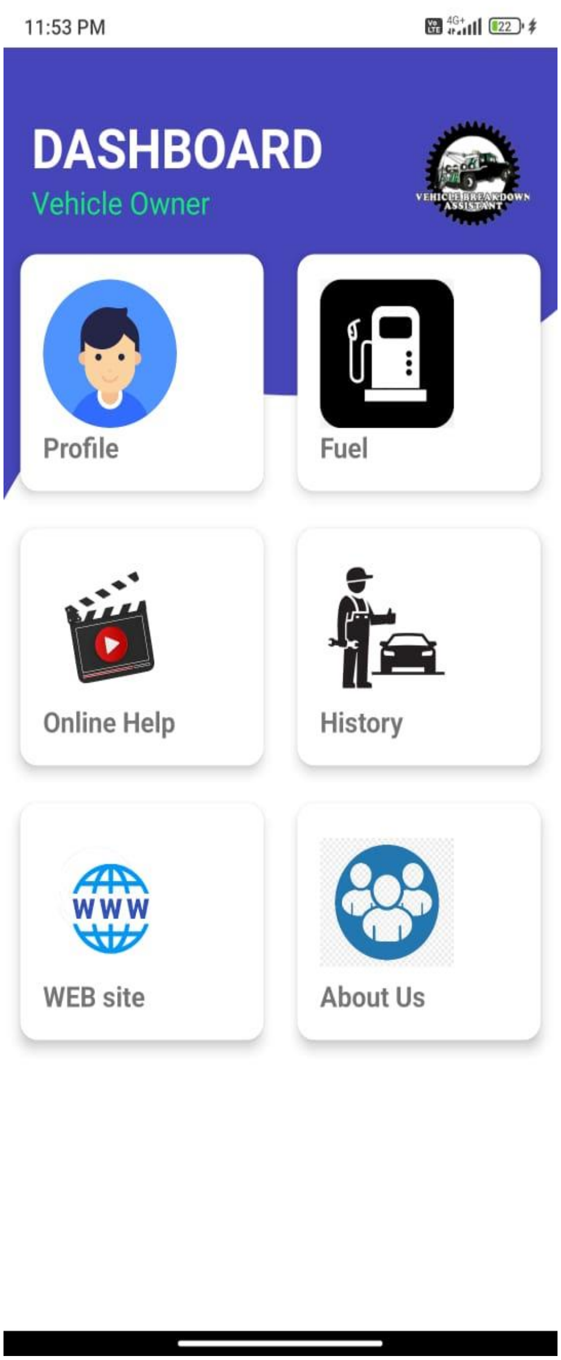


Fig.6.3.Vehicle Owner Dashboard

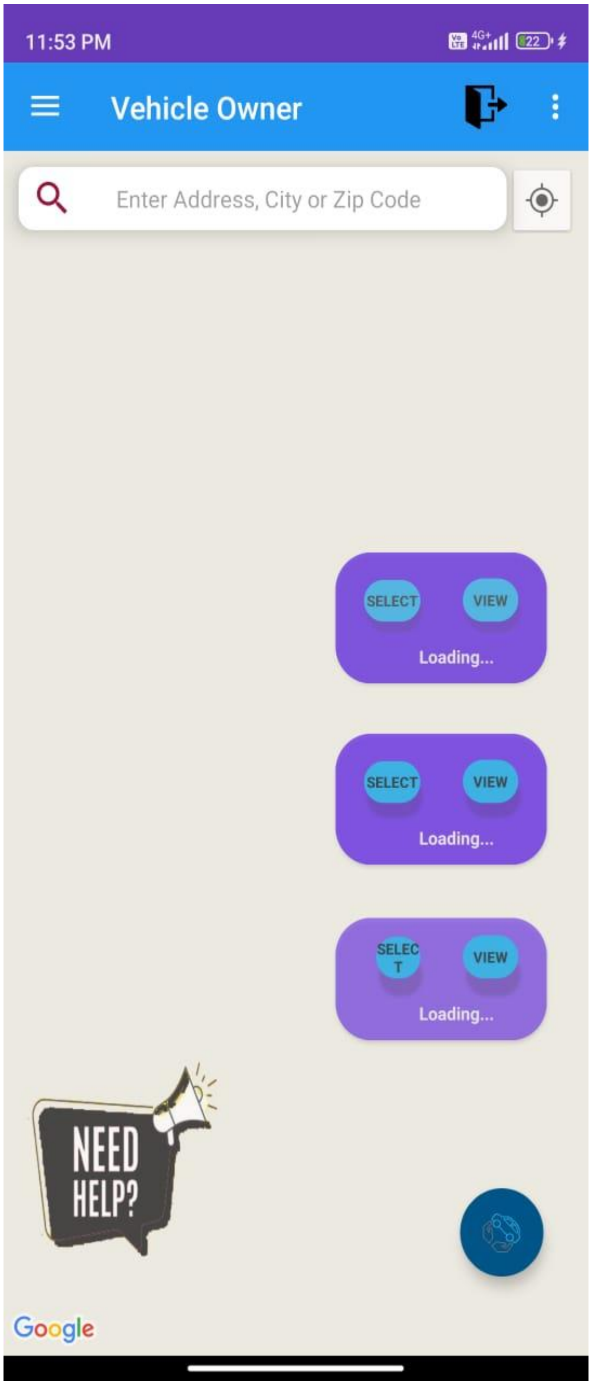
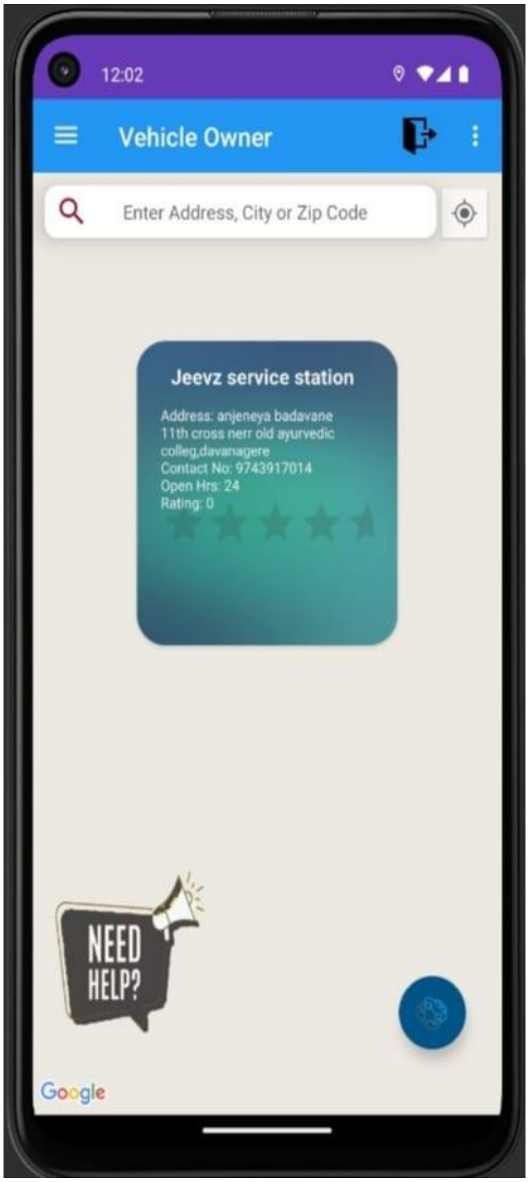
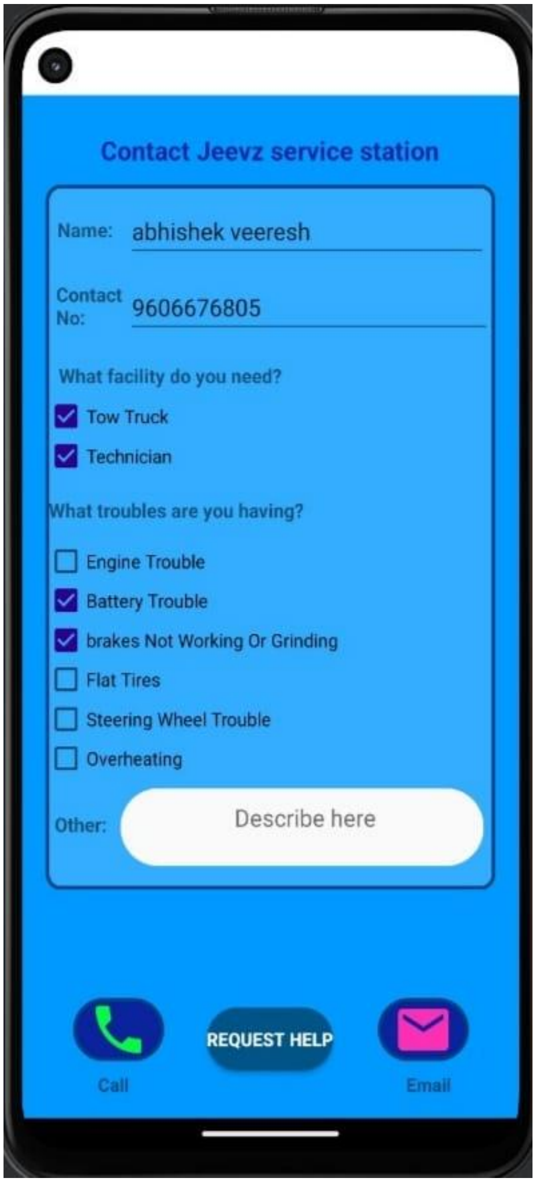


Fig.6.4. Need Help



Jeevz service station

Address: anjeneya badavane
11th cross nerr old ayurvedic
colleg,davanagere
Contact No: 9743917014
Open Hrs: 24
Rating: 0



Contact Jeevz service station

Name: abhishek veeresh

Contact No: 9606676805

What facility do you need?

☑ Tow Truck
☑ Technician

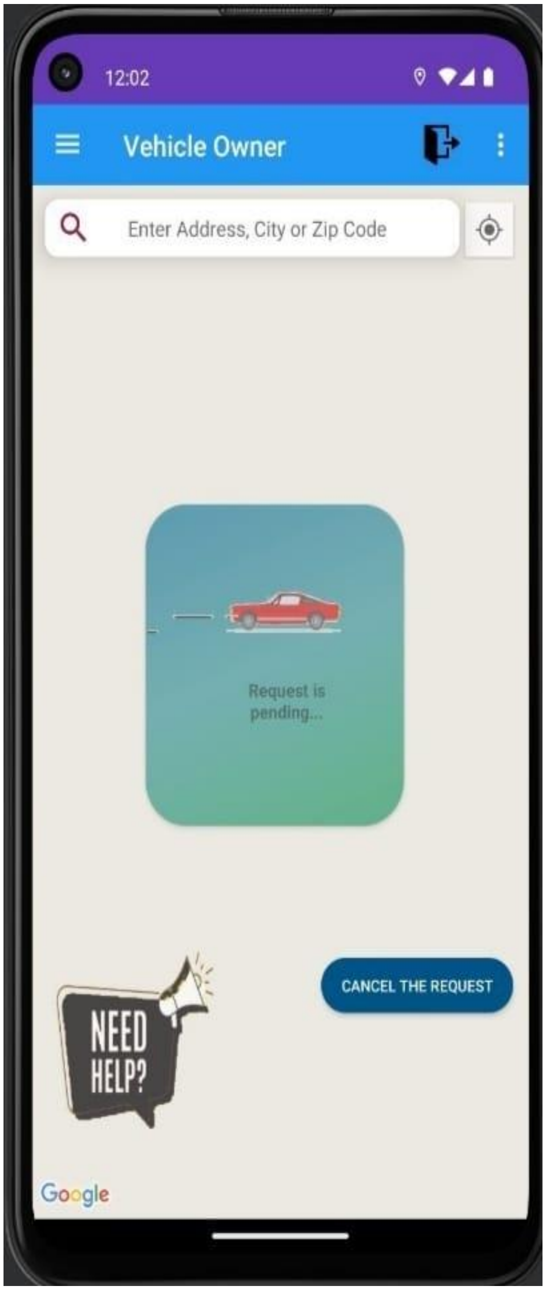
What troubles are you having?

☐ Engine Trouble
☑ Battery Trouble
☑ brakes Not Working Or Grinding
☐ Flat Tires
☐ Steering Wheel Trouble
☐ Overheating

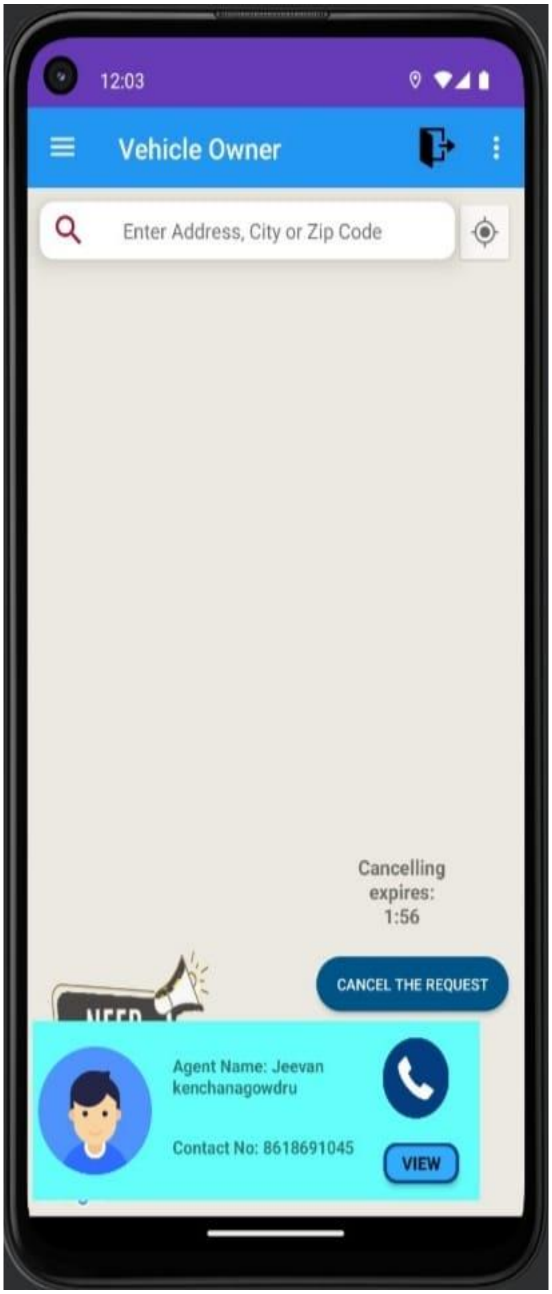
Other: Describe here

Fig.6.5.View And Select Service Stations

Fig.6.6.Choose Options And Request For Help



Request is pending...



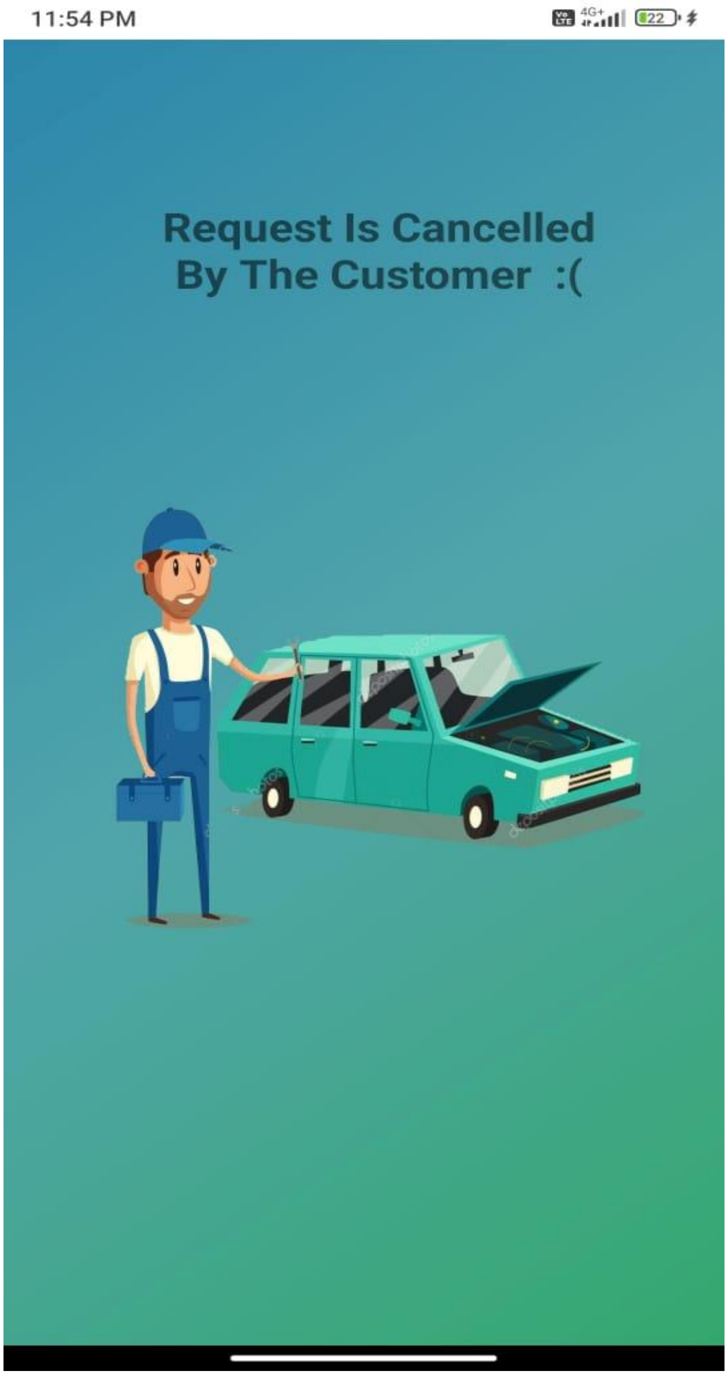
Cancelling expires: 1:56

Agent Name: Jeevan kenchanagowdru

Contact No: 8618691045

Fig.6.7.Waiting For The Response

Fig.6.8.Contacting The Service Station



Request Is Cancelled By The Customer :(

Fig.6.9.Cancelling The Request

➢ SERVICE SATATION PAGE:

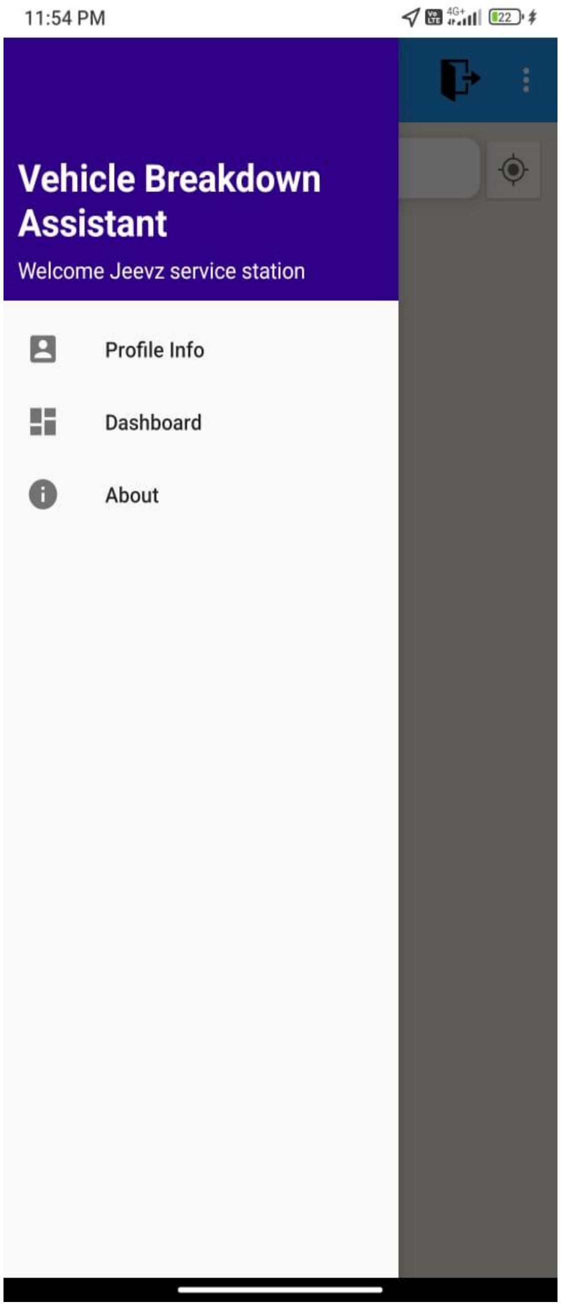


Fig.6.10.Service Station Side Bar Menu

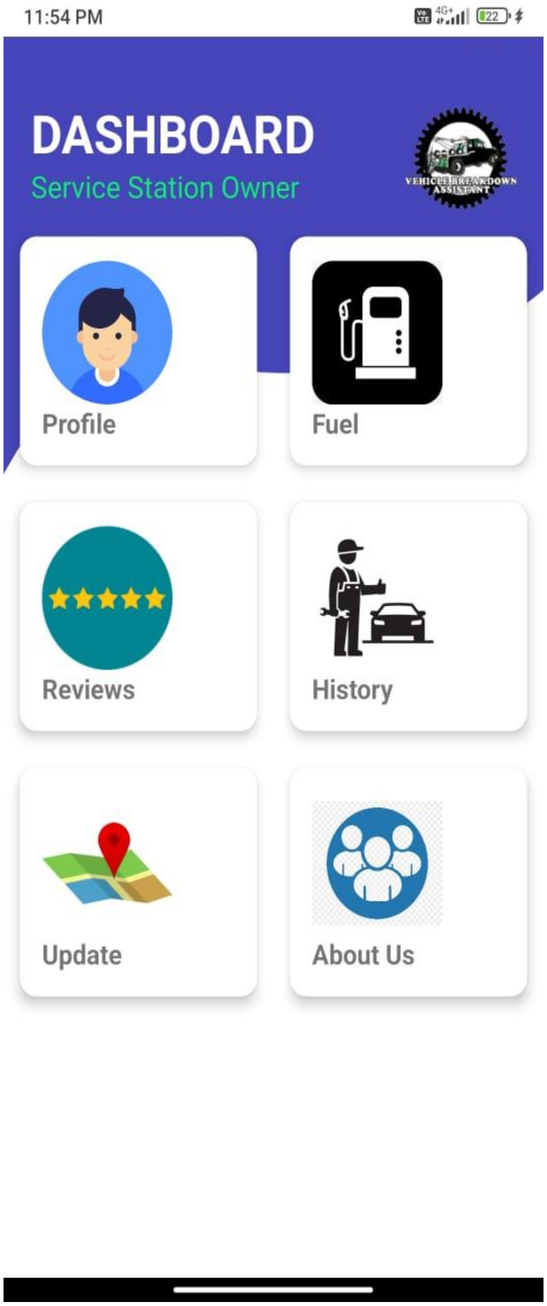
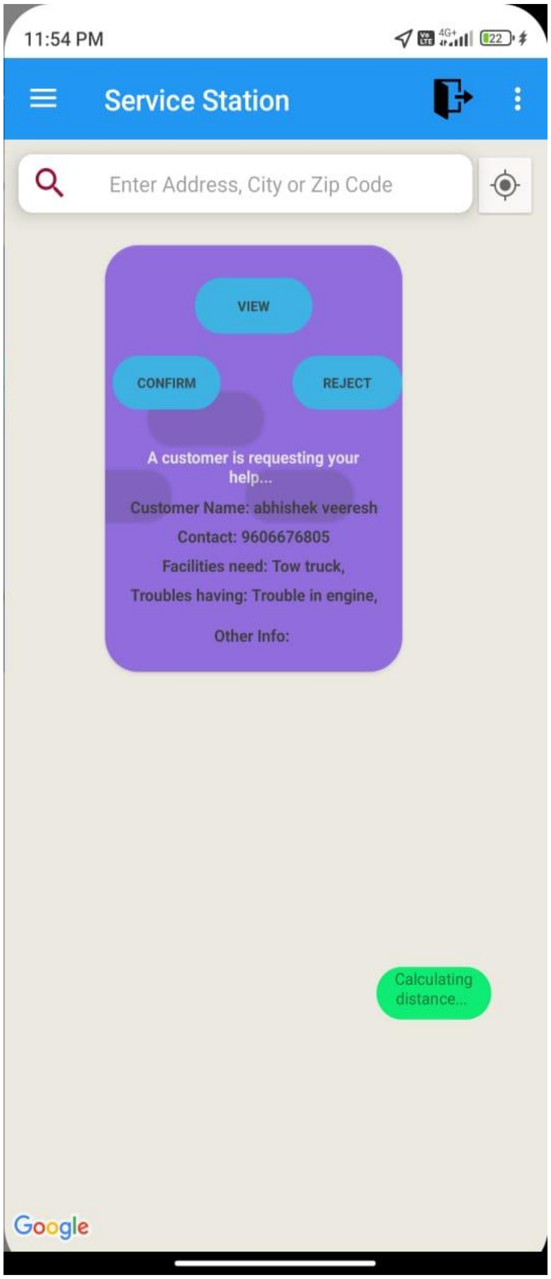


Fig.6.11.Service Station Dashboard



A customer is requesting your help...

Customer Name: abhishek veeresh

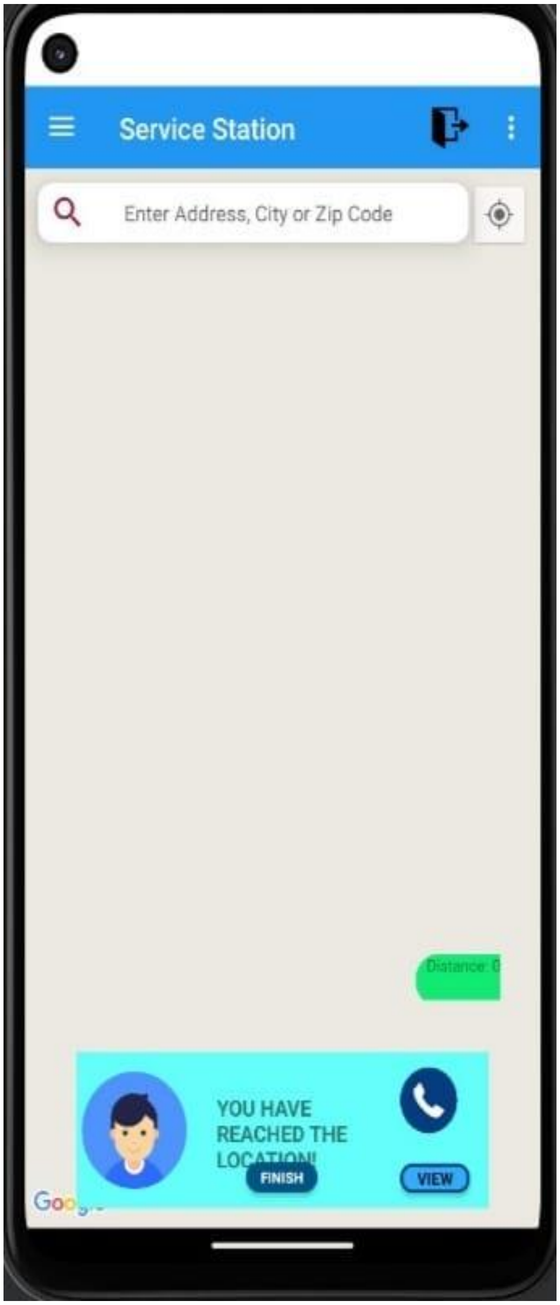
Contact: 9606676805

Facilities need: Tow truck,

Troubles having: Trouble in engine,

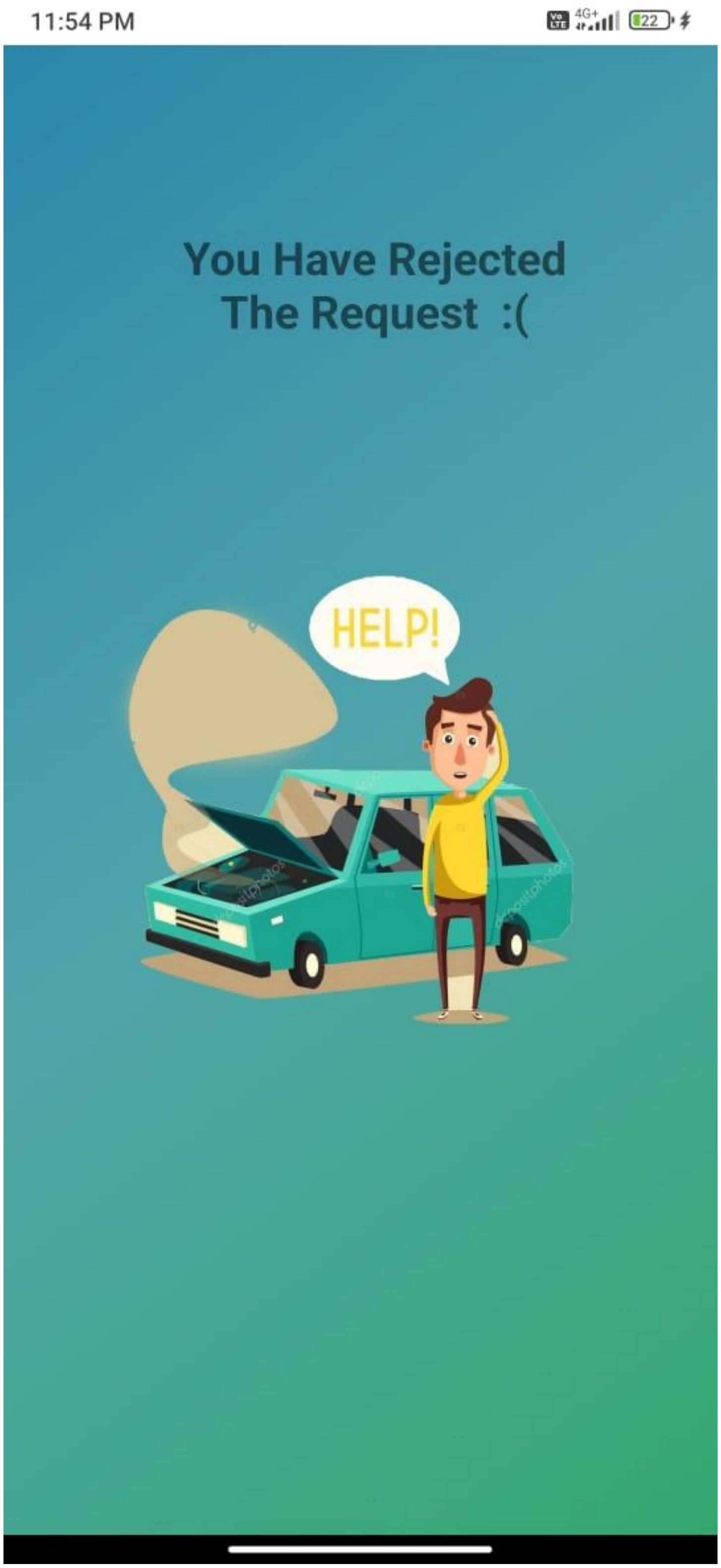
Other Info:

Fig6.12.View,Confirm,Rejection Page



YOU HAVE REACHED THE LOCATION!

Fig.6.13.Contacting The Vehicle Owner


11:54 PM

You Have Rejected The Request :(

Fig.6.14.Rejecting the Request

CONCLUSION

The On-Road Vehicle Breakdown Help Assistance mobile application has successfully achieved its goals of providing a reliable and efficient solution for users facing unexpected vehicle breakdowns. By addressing the user requirements through features such as breakdown reporting, online help through YouTube videos, service provider matching, communication channels, the application ensures a seamless experience for users in distress. The project team followed a structured development process, incorporating modern technologies and adhering to security and privacy standards. The application's intuitive user interface and scalability make it a valuable tool for users, offering timely assistance and enhancing their overall experience during stressful situations on the road.

REFERENCES

BOOKS:

- Google Developer Training. "Android Developer Fundamentals Course-Concept Reference", Google Developer Training Team, 2017.
- Erik Hellman, "Android Programming - Pushing the Limits", 1" Edition, WileyIndia Pvt Ltd, 2014.
- Dawn Griffiths and David Griffiths, "Head First Android Development", 1"Edition, O'Reilly SPD Publishers, 2015.
- Bill Phillips, Chris Stewart and Kristin Marsicano, "Android Programming: TheBig Nerd Ranch Guide", 3 Edition, Big Nerd Ranch Guides, 2017.