On-Road Vehicle Services

Miss. Sindhu S L [1] , Shravanakumar Bellatti [2]

[1]*Assistant Professor, Department of MCA, BIET, Davanagere*

[2]*Student,4th Semester MCA, Department of MCA, BIET, Davanagere*

ABSTRACT

The On-Road Vehicle Road Services App is a mobile application for Android, designed to assist vehicle users in need of immediate roadside help. This app proves to be especially beneficial in unforeseen circumstances such as vehicle breakdowns, fuel shortages, battery issues, tire punctures, or the necessity for emergency towing. Often, drivers may find themselves stranded on highways or in remote locations without immediate assistance, resulting in heightened stress and delays. The objective of this project is to create a centralized, location-based assistance system that links vehicle owners with nearby certified service providers, including mechanics, towing experts, fuel delivery services, battery replacement services, and tire support. This application utilizes Java for Android development, Firebase for user authentication and real-time updates, and MySQL for structured data storage, encompassing user records, service history, billing, feedback, and administrative functions. It guarantees service availability around the clock across various cities by connecting a network of registered service providers to end-users in real-time through GPS location tracking. Users have the ability to book services, view estimated arrival times, communicate directly with providers via in-app calling or chat, rate the services, and process payments (either online or in cash). The app features user-friendly interfaces, a dynamic UI/UX, and appropriate role-based access control. Administrators can oversee users, service categories, locations, service personnel, feedback, and manage emergency escalations in the event of service failures. The entire system provides a scalable and adaptable solution for digital vehicle assistance and road safety.

Keywords & Core Technologies: The project leverages Java for Android development and Firebase for real-time authentication, database syncing, and notifications. It uses MySQL for structured data storage, Google Maps API for GPS tracking, and RESTful APIs for backend communication. The system integrates real-time location-based service matching, secure authentication, and dynamic UI/UX design to ensure seamless roadside assistance.

I. INTRODUCTION

In the contemporary fast-paced environment, the utilization of personal and commercial vehicles has surged significantly, leading to an increase in the occurrence of vehicle breakdowns, malfunctions, and emergencies. Most roadside incidents necessitate prompt intervention; however, the absence of real-time assistance platforms forces users to depend on manual calls or to search for help, which can be both risky and time-consuming.

The On-Road Vehicle Road Services App addresses these challenges. This initiative aims to create a digital mobile platform that provides on-demand, location-based roadside assistance services. Users can request emergency help by simply launching the app and selecting the required service type. A request is subsequently sent to the nearest available professional, who can accept the request and reach the user's location within minutes.

The primary benefit of this project is its simplicity, flexibility, and real-time automation. Both users and service providers engage through the same system, minimizing manual errors, fraud, and wasted time. The platform guarantees that every interaction is recorded, ensuring transparency, accountability, and operational analytics.

assistance. This often involves locating nearby mechanics through personal recommendations, arbitrary Google searches, or relying on towing services offered by car insurance providers, which frequently entail lengthy wait times. While some well-known automobile

The absence of immediate roadside assistance during vehicle breakdowns often results in frustration, wasted time, and safety hazards for travelers. Whether it involves a flat tire, engine malfunction, running out of fuel, or requiring towing services, vehicle owners often find it challenging to quickly identify trustworthy service providers, particularly in unfamiliar locations. Conventional approaches, such as reaching out to friends or manually searching for mechanics or gas stations, tend to be ineffective and at times unreliable.

There is an urgent demand for a mobile solution that can swiftly connect stranded vehicle owners with nearby service providers, including mechanics, towing services, fuel delivery agents, and other emergency vehicle-related assistance. The application should provide a platform that consolidates all service providers in one place and enables real-time location tracking and service booking. The lack of such a centralized system has resulted in a deficiency in emergency vehicle support, which this Android application seeks to address.

II.RELATED WORK

Developers of Android (n.d.).
Important instructions for creating Android applica tions are provided in this official documentation.
In the On-
Road Vehicle Services App, it serves as the funda mental development framework, especially when it comes to organizing user interfaces, activity lifecy cles, and system interactions.enabling rapid prototyping and deployment of Android applications.[1]

Firebase, n.d.

Your app's realtime communication, user login system, and service status updates depend on features like Realtime Database, Authentication, and Cloud Messaging, all of which are described in the Firebase documentation.[2]

Documentation for MySQL (n.d.).

Your project uses MySQL to store service history, user profiles, and ratings in a structured backend manner.

The fundamental information needed to plan and execute the system's relational database features is provided in this handbook.[3]

Verma, A., and R. Sharma (2020).

The authors of Android-Based Emergency Assistance System Using Firebase created an emergency response system using Firebase and Android, which is very similar to your strategy of providing real-time assistance with Firebase and GPS.[4]

Kulkarni, A., and S. Patil (2019).

The application of Android in car breakdown situations is demonstrated by their work on a Smart Vehicle Breakdown Assistance System.

It encourages the use of mobile platforms for roadside assistance and confirms the problem domain of your app.[5] **Mehta, R., and Kumar, A. (2021).**

Their system, Mobile App for On-Road Vehicle Service Management Using Firebase, is quite relevant to your project because it has nearly the same objectives as yours: real-time vehicle servicing, user-provider communication, and Firebase connection.[6]

AndroidHive, n.d.

This guide describes how to use Google Maps and location services on Android, which are necessary for the GPS-based provider tracking and mapping features of your app.[7]

(n.d.) GeeksforGeeks.

The Firebase CRUD Operations tutorial

The main functionality of your applicatioN which involves user-provider interaction through Firebase-based service requests and updates, is directly tied to the use of Java.[8]

(n.d.) Stack Overflow.

This communitybased resource aids in resolving typical Android and Firebase integration problems, particularly those encountered when integrating listener and real-time data synchronization features into your application.[9]

W3Schools, n.d.

Their Introduction to MySQL provides andon foundation for learning how to use MySQL in Android applications.This helps your project manage feedbackand service data in an organized manner.[10]

(n.d.) Google Developers.

Your app can effectively match consumers with local service providers by implementing reatime location services, which are supported by the Google Maps Platform Documentation.[11]

Deshpande, S., and Rao, R. (2020).

Their Emergency Vehicle Management System in Real Time

Android investigates emergency vehicle tracking and routing, which closely resembles the purpose and design of your suggested system.[12]

III. METHODOLOGY

The development of the On-Road Vehicle Road Services Android Application follows a modular, layered, and agile methodology to ensure robust functionality, real-time performance, and scalability. The methodology is divided into several distinct phases, from requirement analysis to deployment, ensuring the system meets user needs effectively.

Requirement Analysis

In the initial phase, functional and non-functional requirements were gathered through brainstorming, problem identification, and analyzing existing gaps in current roadside assistance systems. This included.Identifying user roles: vehicle owners. service providers, and administrators.

System Design

Architecture Planning: Designing a client-server architecture where the Android app interacts with Firebase for real-time operations and with MySQL for structured data via RESTful APIs.

UML Diagrams: Use case diagrams, data flow diagrams (DFDs), and sequence diagrams were created to map system behavior and data interaction.

Frontend Development (Android App using Java)

Tools Used: Android Studio, Java, and XML for UI design.

Modules Implemented:

User Registration/Login (using Firebase Authentication)

Service Selection (e.g., mechanic, fuel, towing)

GPS Location Sharing (Google Maps API)

Real-time Request Updates (Firebase Realtime Database)

UI/UX designed for both users and service providers.

Backend Development

Firebase Integration:

Authentication: Secured user login with OTP/email using Firebase Authentication.

Realtime-Database: Service requests, availability status, and response handling.

Firebase Cloud Messaging (Optional): For real-time notifications about request acceptance and service status.

MySQL Integration:

Used to store historical data, service logs, user profiles, and feedback.Communication established using PHP-based RESTful APIs.

Location & Mapping Functionality

Integrated Google Maps SDK to:

Track users' and service providers' locations in real time.

Display available nearby service providers on the map.

Provide navigation for service personnel to the user's location.

Testing and Debugging

Unit Testing: Individual modules tested for correct functionality.

Integration Testing: Ensured seamless interaction between frontend and backend.

Device Compatibility Testing: App tested across various Android devices and screen sizes.

Performance Testing: Tested Firebase sync speed, GPS accuracy, and response time.

System Requirement Specification

Functional Requirements

• User Registration and Login: Users and service providers are required to create accounts using either phone numbers or email addresses, secured through Firebase Authentication, allowing them to access their respective dashboards.

• Live Location Tracking and Map Integration: The application utilizes the Google Maps API to automatically identify the user's current location, enabling both users and service providers to track each other in real-time for effective assistance.

• Service Request Feature: Users have the ability to submit requests for emergency services such as mechanical assistance, fuel delivery, towing, etc. The application sends notifications to the nearest available service provider and shares the user's location.

• Provider Dashboard: Service providers can receive requests, choose to accept or decline them, and obtain turn-by-turn navigation. They also have the option to mark services as completed and indicate their availability status as online or offline.

• Real-Time Notifications: Through Firebase Cloud Messaging (FCM), both users and providers receive immediate notifications regarding request statuses, confirmations, arrivals, and service completions.

• User Review and Rating System: Upon the completion of a service, users are able to rate the provider and provide feedback, which assists future users in making informed choices and enhances the quality of service.

• Admin Control Panel (Optional): Administrators have the capability to oversee system usage, manage registrations of service providers, review user complaints, and ensure a healthy ecosystem through proper tracking and support.

Architecture Diagram

Fig 4.1. Architecture Review

1. User Interface Layer (Android Mobile App)

• The user engages with the mobile application created using Java and XML within Android Studio.

• This interface facilitates:

o User Registration & Login

o Service Selection (e.g., mechanic, fuel, towing)

o Location Sharing through Google Maps

o Request Submission with necessary details and images if required

o Service Status Tracking

2. Backend Connectivity

• The application interfaces with backend services utilizing:

o Firebase Authentication – for user login & registration

o Firebase Realtime Database – to oversee service requests and real-time updates

o MySQL Database (via PHP API) – for organized storage such as user profiles, service provider information, and historical records

3. Location & Mapping Services

• The Google Maps API is incorporated into the application to:

o Capture the user's current location

o Display nearby service providers

o Monitor the movement of service providers

o Direct service providers to the user's location

4. Firebase Services

• Authentication: Manages login through phone/email securely.

• Realtime Database: Synchronizes service request status and service provider responses in real-time.

• Cloud Messaging (Optional for future): For push notifications when a service is accepted or completed.

5. PHP-Based REST API

• Serves as a conduit between the Android application and MySQL database.

• Manages:

o User data storage/retrieval

o Service provider assignment

o Request logging

o Service history retrieval

6. Admin Dashboard (Optional Web Panel)

• Developed using HTML, PHP, and JavaScript

• Utilized to:

o Oversee ongoing service requests

o Administer user and service provider accounts

o Generate analytics/reports

7. MySQL Database

• Preserves the following:

o User Data: Name, contact, vehicle information

o Service Requests: Type, time, location, status

o Provider Information: Availability, assigned requests

o History Logs: Completed services, reviews, ratings

8. Service Provider Application/Panel (Potential Future Development)

• Distinct application or web interface for mechanics and service agents

o Review and accept requests

o Navigate to the user's location

o Indicate task completion

o Modify availability

9. Essential Technologies Utilized

• Android Studio (Java, XML)

• Firebase (Authentication + Realtime Database)

• Google Maps Software Development Kit

• PHP and MySQL for backend processing

• REST API for communication between Android and the database.

IV. IMPLEMENTATION

The initiative has been created as an Android mobile application to guarantee maximum accessibility, user convenience, and effective functionality. Android, which is developed by Google, is an open-source, Linux-based platform that provides a comprehensive array of development tools, libraries, and APIs, rendering it exceptionally suitable for constructing feature-rich and scalable mobile applications. By utilizing Android Studio, the official Integrated Development Environment, the development process was enhanced with sophisticated coding, design, and testing capabilities.

Java was selected as the primary programming language due to its stability, extensive support, and compatibility with Android devices. The user interface of the application was crafted using XML, ensuring a clean, intuitive, and responsive layout that adjusts well to various screen sizes. RESTful APIs were incorporated to enable secure and efficient communication between the application and backend servers, while the Google Maps API offered precise real-time location services, allowing users to effortlessly locate nearby fuel stations. Focusing on security, the application includes robust user authentication mechanisms, encrypted data storage, and secure transaction processing to safeguard sensitive user information. Comprehensive testing was performed across numerous Android devices and OS versions to guarantee optimal performance, compatibility, and a consistent user experience. The flexibility and rich features of Android made it the perfect platform for delivering the e-Refuel Hub solution.

V. TECHNOLOGY USED

The development of the On-Road Vehicle Road Services Android App integrates multiple technologies to ensure real-time performance, user scalability, and reliability. The system is built using the following key tools and platforms:

Android Studio

- Purpose: Main Integrated Development Environment (IDE) used for developing the Android mobile application.
- Language: Java (for backend logic) and XML (for UI layout).

Features Used:

Activity lifecycle management

Layout editors

Emulator testing tools

Java

Purpose: Core programming language for Android app logic.

Usage:

User authentication and session management

Firebase database interaction

GPS tracking and API calls

MySQL

Purpose: Used for structured data storage and management.

Data Stored:

User and service provider profiles

Service history, reviews, ratings

Admin records and logs

Accessed Through: PHP-based RESTful APIs

PHP

Purpose: Acts as a bridge between Android (client) and MySQL (server).

Usage:

Create, Read, Update, Delete (CRUD) operations for persistent data

RESTful API development for backend communication

Google Maps API

Purpose: Provides GPS-based location services.

Features Used:

Real-time user and provider location tracking

Display of nearby service providers

Navigation and route guidance

XML

Purpose: Used for designing Android UI layouts.

Usage:

Form-based interfaces (login, registration, feedback)

Dynamic layout rendering for various screen sizes

RESTful APIs

Purpose: Enables secure and standardized communication between mobile frontend and backend server (PHP-MySQL).

Functionality:

Data exchange (JSON format)

Login validation

Request logging and history access

Web Technologies (for Admin Panel)

Tools Used: HTML, CSS, JavaScript, PHP

Purpose: Develop a responsive web-based admin dashboard

Functions:

Manage users and service providers

View service logs

Handle feedback and complaints

VI. RESULTS AND SNAPSHOT

Fig:6.1 HOMEPAGE

Welcome to **ORVSA (On-Road Vehicle Service Assistant)** – your trusted companion when vehicle

trouble strikes unexpectedly. Whether you're facing a flat tire, a sudden breakdown, or any other emergency on the road, ORVSA is designed to provide quick and reliable help anytime, anywhere. Our platform connects you to nearby verified mechanics who can offer prompt on-site assistance, ensuring your safety and convenience.

Fig:6.2 LOGIN PAGE

Welcome to the ORVSA Login Portal. This secure gateway allows registered users to access their personalized dashboards based on their roles—Admin, Mechanic, or Public User. Please enter your **username** and **password** in the fields provided to log in and manage your tasks, requests, or services efficiently.

Fig:6.3 REGISTER COMPLAINT

The **Complaint Registration** page allows users to officially log any issues or feedback related to the services provided by mechanics. This form is designed to ensure transparency, accountability, and quick resolution of problems faced during or after service delivery.

Fig:6.4 PUBLIC VIEW

Fig:6.5 SERVICE

The ORVSA platform offers a wide range of essential vehicle services to ensure drivers receive immediate and reliable support during breakdowns or mechanical issues. Whether you're stuck on the highway or at home, our application connects you to verified mechanics who specialize in a variety of vehicle problems.

VII. CONCLUSION

The On-Road Vehicle Road Services Android Application successfully addresses the growing need for real-time, reliable, and accessible roadside assistance. By integrating modern technologies such as GPS-based tracking, Firebase authentication, real-time database synchronization, and MySQL for structured data management, the application ensures that vehicle users can quickly obtain the help they require during emergencies. The centralized platform bridges the gap between stranded vehicle owners and certified service providers, ensuring timely interventions for various issues such as mechanical breakdowns, fuel shortages, towing requirements, battery failures, and tire punctures. Through an intuitive and user-friendly interface, the app offers seamless service booking, real-time communication, transparent feedback systems, and secure transactions, enhancing the overall user experience. Furthermore, the backend architecture, with its scalable and modular design, allows for future expansion and integration of additional features, including an administrative panel and a dedicated service provider application. This flexibility ensures the system's adaptability to various geographic regions and user demands.

In conclusion, the On-Road Vehicle Road Services App offers a comprehensive, efficient, and scalable solution to modern-day roadside emergencies, improving road safety, reducing downtime for vehicle users, and streamlining service delivery. The successful implementation of this project demonstrates the potential of mobile technology to simplify complex real-world challenges and deliver meaningful benefits to end-users.

VIII. REFERENCES

1. Android Developers. (n.d.). *Building Your First App*. Retrieved from https://developer.android.com
2. Firebase. (n.d.). *Firebase Documentation*. Retrieved from https://firebase.google.com/docs
3. MySQL Documentation. (n.d.). *MySQL Reference Manual*. Retrieved from https://dev.mysql.com/doc
4. Sharma, R., & Verma, A. (2020). Android-Based Emergency Assistance System Using Firebase. *International Journal of Computer Applications*, 175(11), 22–27.
5. Patil, S., & Kulkarni, A. (2019). Smart Vehicle Breakdown Assistance System. *Journal of Emerging Technologies and Innovative Research (JETIR)*, 6(5), 523–528.
6. Kumar, A., & Mehta, R. (2021). Mobile App for On-Road Vehicle Service Management Using Firebase. *Proceedings of the 2021 IEEE Conference on Mobile Applications*.
7. AndroidHive. (n.d.). *Android Working with Google Maps and Location Services*. Retrieved from https://www.androidhive.info
8. GeeksforGeeks. (n.d.). *CRUD Operations in Firebase Using Java*. Retrieved from https://www.geeksforgeeks.org

9. Stack Overflow. (n.d.). *Firebase Realtime Database Integration with*

10. W3Schools. (n.d.). *Introduction to MySQL*. Retrieved from https://www.w3schools.com/mysql

11. Google Developers. (n.d.). *Google Maps Platform Documentation*. Retrieved from

Android. Retrieved from https://stackoverflow.com

https://developers.google.com/maps/documentation

12. Rao, R., & Deshpande, S. (2020). Real-Time Emergency Vehicle Management System Using Android. *International Journal of Recent Technology and Engineering (IJRTE)*, 8(5), 124–128.