Mecharide Cloud: Quick Fix for Your Bike

Manjula.E[1], G.Mahalakshmi[2]
[1]*Student, Master of Computer Applications, Dr.M.G.R. Educational and Research Institute, Chennai*
[2]*Assistant Professor, Master of Computer Applications, Dr.M.G.R. Educational and Research Institute, Chennai*

Abstract - A cloud-based web and mobile application called MechaRide Cloud was created to help bikers who have mechanical issues by instantly connecting them with local mechanics. Users can register, track their location, and submit service requests based on their problems, like tire punctures or engine failures, using the app's Google Maps API, Firebase Cloud Messaging, and AI-powered request allocation system. Mechanics can accept or reject requests, and users can upload images for better help. The platform accepts Stripe and Razorpay for safe cashless transactions. Users can also review and rate mechanics, and requests can be saved for later use in an offline mode. Voice commands, AI diagnostics, and chatbot support are examples of future improvements that demonstrate the potential of cloud computing and IoT in on-demand services.

INTRODUCTION

Mobility is essential in today's fast-paced world, which has led to an increase in two-wheeler vehicles and, as a result, mechanical failures like flat tires and battery problems. These malfunctions frequently happen at erratic times and places, leaving riders stranded and unable to get help right away. Particularly in rural areas, traditional roadside services can be difficult to reach, slow, and unreliable. Bike ResQ, a location-based, real-time web application created to offer prompt and effective bike breakdown assistance, was inspired by this challenge. Bike ResQ efficiently links injured riders with local certified mechanics by leveraging cutting-edge web technologies, real-time communication protocols, and geolocation services. This guarantees prompt assistance and improves the riding experience overall. In the expanding two-wheeler market, there is a pressing need for dependable roadside assistance, which this creative solution fills.

OBJECTIVE

The BikeResQ project's primary goal is to create a real-time web-based platform that allows users (riders) to instantly connect with the closest available mechanic and request urgent assistance when they encounter bike-related problems. The platform enables real-time routing and live location tracking.

- One-on-one chat, picture, and voice communication;
- ETA-based mechanical response;
- Integration of payment (UPI/cash);
- Service history for future use

SCOPE

The project is made to function as a mobile-responsive web application, guaranteeing accessibility even on smartphones without the need for app installation. It is appropriate for independent and registered mechanics, fleet companies, and insurance services seeking a quicker roadside response in urban and semi-urban areas.

Features such as a mechanic rating and review system, predictive maintenance alerts, subscription-based premium support, and integration with government road safety databases could be added to the platform in the future.

MATH

ETA (in minutes) is calculated as follows: Distance (km) / Average Speed (km/h) * 60 Steps

1. Calculating Distance:

- Make use of either:

The distance calculated by the routing plugin (in kilometers) and the Haversine formula (fallback)

2. Define the Heuristic Speed:

- Average speed in the city: about 20 km/h; optional: retrieve historical mechanical movement speed

3. Determine the ETA:

- Enter the formula to obtain minutes plus seconds.

4. Format of Display:

• Estimated text Arrival time: 4 minutes and 30 seconds

5. Refresh Every Short While:

• Rerun the computation after every route update or on a regular basis.

Results:

• The mechanic and user dashboards show the ETA value.

REVIEW OF LITERATURE

The need for quick assistance services is highlighted by the frequent occurrence of bike breakdowns, especially in isolated locations. Since most roadside assistance services are geared toward urban areas, they frequently fall short when it comes to helping two-wheelers.

Mobile apps have become useful resources that link riders with local mechanics and offer crucial assistance when a breakdown occurs. Real-time tracking and GPS integration greatly speed up reaction times and increase rider safety.

User preferences show a high demand, particularly during emergencies, for quick, easy, and dependable assistance systems. This emphasizes the need for additional advancements in roadside assistance designed with two-wheeler riders in mind.

SYSTEM CONFIGURATION

HARDWARE SPECIFICATIONS

1. User & Mechanic Device: Mobile devices that support Android and iOS or web browsers.
2. Cloud server for backend services and data storage: AWS, Google Cloud, Firebase, and Render.

SOFTWARE SPECIFICATIONS

1. Windows (for development) as the operating system.

Windows is a popular choice for software development due to its extensive software compatibility, robust development environment, and large user base. It is particularly well-suited for development tasks that involve Microsoft technologies and frameworks.

2. Development Tools: VS Code (for frontend-HTML, CSS & Javascript. and backend - Flask/Django)

3. Cloud Platform: AWS /Render for hosting & data storage.

CLOUD SERVER

A cloud server is a virtual server that operates in a cloud computing environment and is reachable from a distance via a network, most frequently the internet. Like conventional physical servers, these servers offer computing resources like processing power, storage, and applications, but they also offer the advantages of flexibility and remote accessibility.

FLASK

A web framework is called Flask. This indicates that Flask gives you the technologies, tools, and libraries you need to create a web application. This web application can be as small as a blog, wiki, or web age, or it can be as large as a commercial website or a web-based calendar application.

The term "micro framework" is frequently used to describe Flask. It seeks to maintain an application's core functionality while allowing for expansion. Flask lacks both a built-in abstraction layer for database management and a validation support. Rather, extensions that add such functionality to the application are supported by Flask. Despite being relatively new in comparison to the majority of Python frameworks, Flask has already gained popularity among Python web developers and has a lot of potential.

CLOUD PLATFORM:

AWS /Render for hosting & data storage A cloud server is a virtual server running in a cloud computing environment, accessible remotely over a network, and often the internet. Cloud storage is a way to store digital data on remote servers managed by a third-party provider, accessible over the internet.

GOOGLE MAPS API

Developers can access and incorporate Google Maps data and functionality into their applications using a suite of APIs provided by the Google Maps Platform. These APIs cover a wide range of use cases, such as displaying maps, locating locations, figuring out routes, and retrieving comprehensive location data.

Used APIs:Google Maps API (for tracking location in real time) For push alerts, use Firebase Cloud

Messaging (FCM). API for Payment Gateways (UPI transaction.)

MONGO DB/FIREBASE

Despite being NoSQL databases, Firebase and MongoDB serve distinct purposes. While MongoDB is more reliable for intricate data modeling and big datasets, Firebase is better at real-time data synchronization and quick application development.

CONCLUSION

The BikeResQ project is a real-time, location-based platform designed to swiftly connect stranded two-wheeler users with nearby mechanics. It meets the increasing demand for on-road support through features like live location tracking, dynamic routing, OTP-based authentication, and multimedia-supported real-time chat. This system enhances user convenience while empowering mechanics digitally. Built on a Flask-SocketIO backend with OracleDB for data persistence, it ensures scalable communication. The frontend utilizes Leaflet.js and modern browser APIs to provide a responsive, interactive user experience. Extensive testing confirmed the system's reliability in routing, chat functionality, role-based data flow, and mobile compatibility. Its modular design allows for future expansions, such as multi-vehicle support and emergency services, effectively bridging the gap between traditional mechanic services and modern user expectations.

REFERENCE

[1] "Emergency Breakdown Services Using Android Application" (IRJET), vol. 04, Issue 04, April 2017, Varun Kapadi, Saigita Guruju, et al. (2017).

[2] M.A. D. Wickrama, D.S.C. Dharmakeerthi, et al. (2021) "Mobile Based Solution for Vehicle Assistance" (IEEE) DOI: 10.1109/ICAC54203.2021.9671196.

[3] Monika Kadam, Neelima Sutar, Pooja Dorge, and others (2018) "A Car Breakdown Service Station Locator System" (ISSN) vol. 03, Issue 04, April 2018.

[4] Dr. S. Anupama Kumar and Rakshit Sadanand Bhat (2021) "Application of Vehicle Breakdown Assist Model" (JETIR) vol. 08, Issue 05, May 2021.

[5] Akhila V. Khanpuri, Anagha Shastri, et al. (2015) "On Road: A car assistant application" (IEEE) ICTSD-2015.

REFERRED

1. Roadside assistance apps for emergencies: https://www.aaa.com/aaa/app/roadside-assistance
Roadside assistance features are available on the AAA mobile app for prompt assistance.

2. Mobile Vehicle Assistance Solutions https://www.mobileappdaily.com/mobile-apps-for-vehicle-assistance
An overview of the different smartphone apps made to help with driving.

3. Creating Vehicle Assistance Android Apps
Resources for app development are available on the official Android developer website, https://developer.android.com/.

4. On-Demand Vehicle Assistance Services: https://www.rideapart.com/news/516103/on-demand-motorcycle-repair-service
An article about on-demand services for motorcycle maintenance and repair.