SafeRoute AI: Mobile Safety Assistant Transforming Smartphones into Intelligent Driving Co-Pilots

Higher Diploma in Software Engineering
Final Project Documentation

2025.1F

School of Computing and Engineering

National Institute of Business Management

Colombo 7

SafeRoute AI: Mobile Safety Assistant Transforming Smartphones into Intelligent Driving Co-Pilots

SUBMITTED BY

COHNDSE251F-012	-	M.K.W.A.D.S.S. Siriwardhana
COHNDSE251F-017	-	I. N.Y. Dias
COHNDSE251F-043	-	G.N.A. Kodagoda
COHNDSE251F-074	-	H.M.D.P. Herath

Higher National Diploma in Software Engineering
School of Computing and Engineering
National Institute of Business Management
Colombo-7

24/01/2026

"The project is submitted in partial fulfilment of the requirement of the Higher National Diploma in Software Engineering HNDSE25.1F of National Institute of Business Management."

Contents

List of Keywords, Abbreviations, Tables, and Figures

Keywords

- Ride Safety Intelligence
- Road Hazard Detection
- Predictive Safety Alerts
- Geo-Spatial Hazard Mapping
- Crash Detection
- Emergency Response System
- Driver Safety Score
- Road Quality Analytics
- Preventive Intelligence
- Smart Transportation

Abbreviations

Abbreviation	**Full Form**
GPS	Global Positioning System
SOS	Save Our Souls (Emergency Alert Signal)
API	Application Programming Interface
G-Force	Gravitational Force
UML	Unified Modelling Language
IDE	Integrated Development Environment
UI	User Interface
Firebase	Google Cloud-based Real-time Database Platform

Chapter 1: Introduction

1.1 Introduction of the Domain

Road transportation plays a vital role in modern society, yet road safety continues to be a major concern worldwide. Drivers frequently encounter unexpected hazards such as potholes, speed bumps, poor road conditions, and adverse weather, which increase the risk of accidents and vehicle damage.

With the advancement of mobile sensors, cloud computing, GPS technology, and machine learning, intelligent safety systems can now assist drivers before, during, and after a ride. The proposed **Smart Ride Safety Intelligence System** is designed to enhance driver awareness, reduce accidents, and improve overall road safety through preventive intelligence, real-time monitoring, emergency response, and post-ride analytics.

1.2 Research Gap

Existing navigation and safety applications provide route guidance but lack advanced hazard detection and predictive safety capabilities.

Previous studies highlight several limitations:

- Many systems rely heavily on manual hazard reporting, which reduces data accuracy.
- Existing solutions rarely differentiate between potholes and speed bumps.
- Few applications integrate real-time vibration analysis with geo-spatial hazard mapping.
- Limited focus is given to driver wellbeing analytics and long-term safety scoring.
- Emergency response features often lack intelligent false-alarm filtering.

Therefore, there is a need for an integrated system that combines **preventive intelligence, automated hazard detection, predictive alerts, and driver safety analytics** into a single platform.

1.3 Research Problem Definition

Road transportation presents drivers with many unpredictable dangers such as potholes, uneven road conditions, and poor weather, which can lead to accidents and vehicle damage. While existing navigation systems provide directions and traffic updates, they generally lack the ability to automatically detect road hazards and warn drivers in advance. Many platforms rely on manual reporting, resulting in delayed or inaccurate information that reduces driver preparedness and overall road safety.

Additionally, most current solutions do not integrate emergency handling features or provide long-term insights into driver safety. In critical situations such as crashes, the absence of rapid alert systems can delay assistance. Furthermore, drivers often lack access to analytics that help them understand their driving patterns and improve safety. Therefore, there is a clear need for a unified system that combines automated hazard detection, predictive alerts, emergency response, and safety analytics to enhance driver awareness and promote safer travel.

.

1.4 Research Objectives

1. Main Objective
    - To develop an intelligent ride safety system that enhances driver awareness and reduces road risks through real-time monitoring and predictive analytics.

2. Specific Objectives

    - Implement a **Pre-Ride Weather Risk Guard** to identify weather-related dangers before travel.
    - Develop **Automated Road Anomaly Detection** using dynamic vibration analysis.
    - Enable **Geo-Spatial Hazard Mapping** for community-based hazard sharing.
    - Provide **Predictive Proximity Alerts** to warn drivers in advance.
    - Generate **Road Quality Analytics** for safer route selection.
    - Integrate **Smart Crash Detection** with false-alarm filtering.
    - Support **Automatic SOS and One-Tap Emergency alerts**.
    - Produce **Driver Safety Scores, trip summaries, and family-aware safety sharing** after each ride.

1.5 Proposed Solution

The proposed system is an intelligent mobile-based platform that operates across four phases of a journey.

Features of the Proposed System

1. Before Ride – Preventive Intelligence
    - Pre-Ride Weather Risk Guard:
        - Evaluates weather conditions before departure and alerts users about potential risks such as heavy rain or storms.

2. During Ride – Real-Time Safety Intelligence
    - Automated Road Anomaly Detection:
        - Uses a 3-axis accelerometer to detect vertical G-force spikes.
    - Pothole vs. Bump Classification:
        - Differentiates road depressions from elevations using algorithmic logic.
    - False-Positive Suppression:
        - Filters non-road vibrations such as phone movement or sudden braking.
    - Geo-Spatial Hazard Mapping:
        - Instant GPS geo-tagging when hazards are detected.
        - Real-time cloud synchronization via Firebase.
        - Virtual hazard markers displayed on Google Maps.
    - Predictive Proximity Alerts:
        - Early visual notifications when approaching hazards.
        - Blind-spot awareness for low-visibility conditions.
    - Road Quality Analytics:
        - Severity grading (Low, Medium, High).
        - Route safety scoring with smoother route suggestions.

3. Emergency Handling and Response
    - Smart Crash Detection with False-Alarm Filtering
    - Automatic SOS and One-Tap Emergency Button

4. After Ride – Post-Ride and Long-Term Intelligence
    - Driver Safety and Wellbeing Score Generation
    - Trip Summary and Personalized Feedback
    - Family-Aware Safety Status Sharing

1.6 Chapter Summary

This chapter introduced the transportation safety domain, identified research gaps, defined the problem, outlined objectives, and presented the proposed intelligent safety solution.

Chapter 2: Methodology

2.1 Data Collection Methods

Data will be collected through:

- Smartphone sensors (accelerometer and GPS).
- Real-time ride data.
- Cloud-stored hazard reports.
- Weather data from external APIs.

2.2 Software Process Model

The Agile methodology is adopted for this project to ensure a flexible and efficient software development process. Agile focuses on developing the system in small, manageable iterations, allowing the team to design, build, and test features step by step rather than completing the entire system at once. This approach helps identify issues early, reduces development risks, and improves the overall quality of the product.

Agile encourages continuous testing and regular feedback throughout the development cycle. Each iteration allows for quick feature enhancements and necessary adjustments based on user needs or technical requirements. As a result, the methodology supports rapid improvements, better collaboration among team members, and the delivery of a reliable and user-focused intelligent safety system.

2.3 Software Development Tools

- Programming Language : React / Flutter (Android)
- Development Environment : Android Studio
- Database : Firebase Real-time Database
- APIs : Google Maps,Weather API, SMS and Notification APIs
- Version Control : Git and GitHub
- Design Tools : Draw.io (UML and system diagrams)

2.4 Testing Strategies

1. Unit Testing
   • Verify individual modules such as hazard detection.

2. Integration Testing
   • Ensure proper communication between sensors, cloud, and alerts.

3. System Testing
   • Validate complete ride workflows.

4. User Acceptance Testing (UAT)
   • Evaluate usability and reliability with real users.

2.5 Implementation Plan

- Requirement analysis
- System design
- Sensor integration
- Cloud database setup
- Alert system development
- Emergency feature implementation
- Testing and optimization
- Deployment

2.6 Chapter Summary

This chapter outlined the methods, tools, testing strategies, and implementation plan used for developing the proposed system, ensuring an organized and effective software development process.

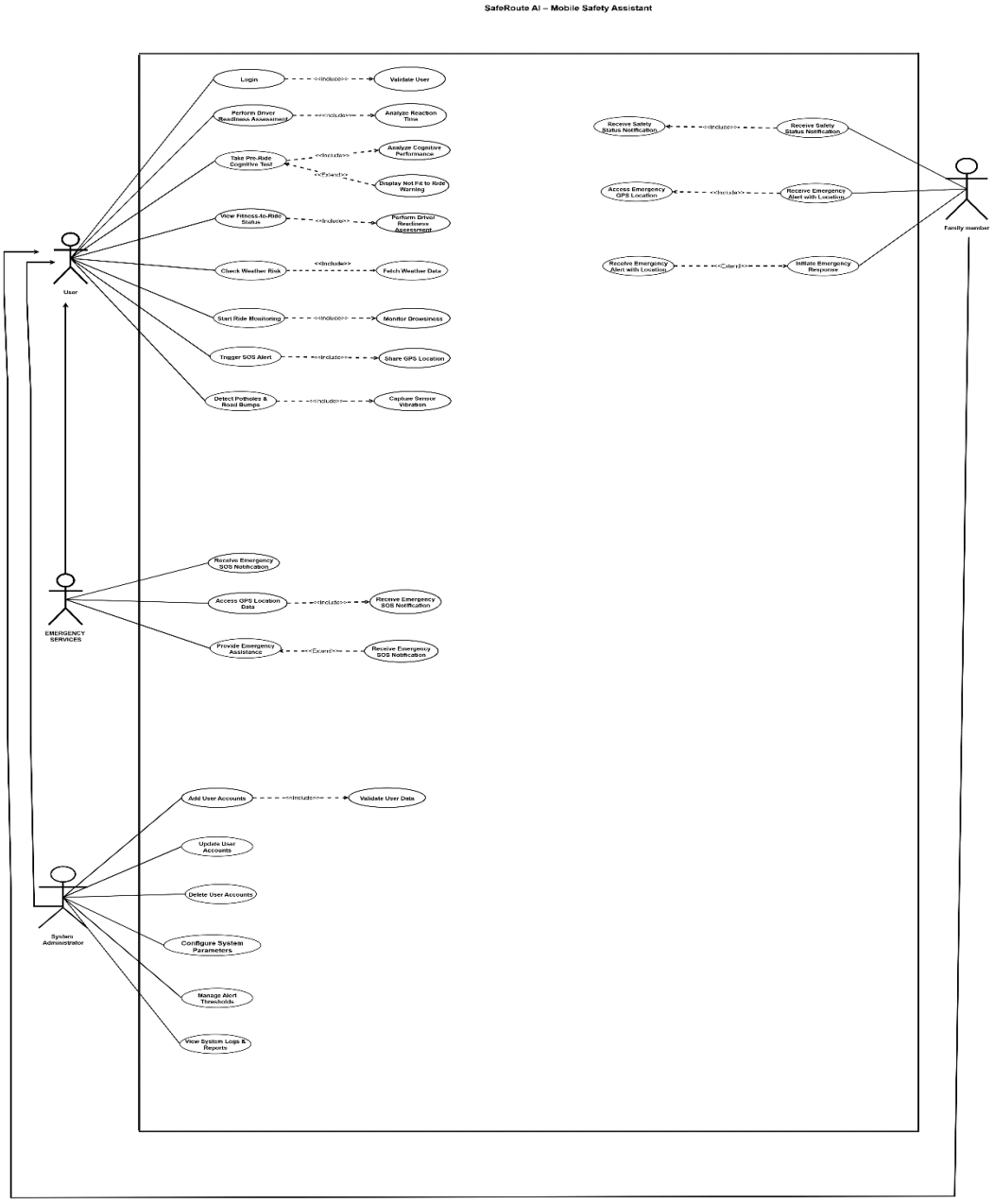
Chapter 3: Analysis

3.1 Problem and Solution Analysis

Problem–Solution Mapping

Identified Problem	**Proposed Solution**
Unexpected road hazards	Automated anomaly detection
Lack of hazard awareness	Predictive proximity alerts
Poor route safety insights	Road quality analytics
Delayed emergency response	Smart crash detection + SOS
Limited driver monitoring	Safety and wellbeing scores
Family anxiety	Family-aware safety status sharing

3.2 UML Diagrams of the Proposed System



3.3 Data Management of the Proposed System

The system stores hazard locations, ride analytics, and safety scores in Firebase.

Data includes:

- GPS coordinates
- Hazard severity
- Ride history
- Driver safety metrics
- Emergency contacts

Cloud synchronization ensures real-time updates and community-based hazard awareness.

3.4 Chapter Summary

This chapter examined the system's key problems, proposed suitable solutions, illustrated system interactions through diagrams, and described the data management approach used to support the system.

Chapter 4: Conclusion

The Smart Ride Safety Intelligence System provides a comprehensive approach to improving road safety through preventive intelligence, real-time hazard detection, emergency support, and post-ride analytics. By integrating sensor data, cloud infrastructure, and predictive algorithms, the system enhances driver awareness and promotes safer travel experiences.

Chapter 5: References

1. Abbas, Q. and Alsheddy, A., 2021. Driver Fatigue Detection Systems Using MultiSensors, Smartphone, and Cloud-Based Computing Platforms: A Comparative Analysis. *Sensors*, 21(1), p. 56.
2. Jabbar, R., Al-Khalifa, K., Kharbeche, M., Alhajyaseen, W., Jafari, M. and Shan, J., 2018. Real-time Driver Drowsiness Detection for Android Application Using Deep Neural Networks Techniques. *arXiv preprint*.
3. A Smartphone-Based Driver Safety Monitoring System Using Data Fusion, 2012. *Sensors*.
4. Technologies for Detecting and Monitoring Drivers' States: A Systematic Review, 2024. *Sensors*.
5. Balaji, C.G., Sai Krishnaa, V.R., Shyam Sundar, S. and Rajesh Kumar, S., 2025. Driver Safety Advancements: Drowsiness Detection with YOLO V8. In: *Communications in Computer and Information Science*.
6. Driver Drowsiness Detection Using Smartphone, 2025. *International Journal for Research in Applied Science and Engineering Technology (IJRASET)*.
7. Intelligent Fatigue Driving Detection Method Based on Fusion of Trajectory and Physiological Data, 2023.
8. World Health Organization, 2023. *Road Traffic Injuries Report*. World Health Organization.
9. Android Developers Documentation. Available at: https://developer.android. com
10. Google Maps Platform Documentation. Available at: https://developers.google. com/maps
11. Additional research articles on driver fatigue detection and mobile sensor-based accident detection, as cited throughout this study.

Chapter 6: Appendices

6.1 Project Timeline and Implementation Schedule

AI Driver Guard System	27th Jan	3rd Feb	10th Feb	11th Feb	17th Feb	24th Feb	7th Mar	14th Mar
Proposal Submission • Project Planning • Literature Review • Feasibility Study	**Weeks – 01**							
1st Progress Review (Design Phase) • Requirement Analysis • System Design • UML Diagrams		**Weeks – 02**						
2nd Progress Review (Development Phase) • Frontend Development • ML Model Development • Backend Development				**Weeks – 03**				
Mid Viva • System Integration • Testing Preparation						**Mid-Viva**		
Final Submission • System Testing • Bug Fixing • Documentation • Final Viva Preparation							**Weeks- 04**	**Final-Viva**