

SafeRoads Navigator:

A System for Real-Time Road Hazards Management Using Crowdsourced Data and Dual Vetting Approach

Final Capstone Project Report

Submitted by:

Arnold Aristotle Tayag

Yoobee College of Creative Innovation

Student ID: 270559700

Programme:

Master of Software Engineering

Supervisor:

Arshad Khan

Date of Submission:

20th July 2025

Declaration of Originality

I hereby declare that this project report, titled:

“Saferoads Navigator: A System for Real-Time Road Hazards Monitoring Using Crowdsourced Data and Dual Vetting Approach”,

is my own original work and has not been submitted previously, in whole or in part, for the award of any degree or diploma in any university or institution.

I have duly acknowledged all sources of information and assistance used in the preparation of this report.

I confirm that any content generated or assisted by Artificial Intelligence (AI) tools (e.g., ChatGPT, GitHub Copilot, etc.) has been used responsibly and ethically. Specifically:

- AI assistance was limited to idea generation, code support, or grammar correction.
- Any AI-generated content that influenced technical aspects or writing has been critically reviewed and edited by me.
- I accept full responsibility for the accuracy, originality, and integrity of this report.

I understand that any violation of this declaration may lead to disciplinary action in accordance with academic integrity policies.

Student Name : **Arnold Aristotle Tayag**

Student ID : 270559700

Date : 20th July 2025

Signature : _____

Acknowledgement

I have reached the culmination of another milestone in my life. In writing this capstone project study, I reminisce the times when people have reached out to lend me a hand in one way or another.

Firstly, my thanks and heartfelt gratitude go to my Family for their continuous support, love, and patience in this one whole year of my educational journey.

Secondly, I am equally thankful to my supervisor, the professors, mentors, and the staff of Yoobee College. Your assistance to me and the whole class, academically or otherwise, was a welcome experience indeed. You have helped us on this journey and I, for one, am very thankful.

Finally, I thank the Lord, without which this milestone would not have been possible.

I thank you all from the bottom of my heart for being an integral part of another milestone in my life.

Table of Contents

I. Abstract.....	6
II. Introduction	7
2.1 Background of the Study	7
2.2 Problem Statement.....	8
2.3 Objectives of the Study	10
2.4 Scope and Limitation	11
III. Literature Review	12
3.1 Overview of Existing Research	12
3.2 Identified Gaps	17
3.3 Motivation of the Study.....	19
IV. Design, Methodology, & Architecture	20
4.1 Solution Use Cases.....	20
4.2 Development Methodology.....	23
4.3 System Architecture	25
4.4 Technology Stack and Architecture	27
V. Development Plan & Implementation.....	30
5.1 Project Plan and Timeline	30
5.2 Key Features and Functionality	32
VI. Testing and Evaluation	42
6.1 Test Plan	42
6.2 Evaluation Metrics	46
VII. Results and Discussion.....	47
VIII. Conclusion & Future Work.....	50
8.1 Conclusion	50

8.2 Future Work	52
IX. References.....	54

List of Figures

Figure No.	Description	Page
Figure 1	Road Hazard Identification and Treatment (process flow)	21
Figure 2	Road Hazard Status Lifecycle Diagram	22
Figure 3	Architecture Diagram	25
Figure 4	Database Design (ERD)	26
Figure 5	Hazard Reporting Dashboard	32
Figure 6	Hazard Report Grid showing all hazards near the location of a road user, specified by RADIUS	32
Figure 7	The Interactive Map showing all hazards reported to the system, with emphasis on the ones that are near the road user	33
Figure 7.1	Interactive map highlighting hazards within the road user's vicinity	34
Figure 8	Heatmap showing concentration of hazards	35
Figure 9	Road User Upvoting/Flagging (aka Peer Vetting)	36
Figure 10	Moderator Verification (aka Authority Vetting)	37
Figure 11	Analytics Dashboard showing overview statistics	38
Figure 11.1	Analytics Dashboard showing hazard classification stats	38
Figure 12	Access the Reports Dashboard to export hazard reports into a CSV	39
Figure 13	Maintenance Dashboard	40
Figure 14	PIIs in the system are encrypted/hashed	40
Figure 15	Maintenance Dashboard showing encrypted user data	41

I. Abstract

SafeRoads Navigator is a capstone project proposal for a real-time road hazards monitoring system that leverages crowdsourced data coupled with a dual vetting mechanism to ensure information accuracy. The system empowers both the general public and road safety authorities to collaboratively identify, track, and resolve roadway safety issues in real time. The system addresses the critical need for timely and reliable hazard alerts by enabling drivers, passengers, motorists, cyclists, & pedestrians to report road dangers — such as potholes, heavy rain & flooding, obstacles, accidents, broken signage, adverse conditions, etc. — and employing two levels of validation: 1) peer confirmation & 2) expert/administrative review after broad dissemination. This proposal outlines the motivation and background for such a system, identifies gaps in existing solutions, and formulates research questions to guide development. Through an intuitive & interactive frontend and a Node.js/Express backend tied to Google Maps API, users can report road hazards via automatic GPS location capture, or manual entry as the case maybe, attach photo evidence, & select from predefined categories of hazards. Crowdsourced verification and road safety stakeholder moderation ensure data reliability, while configurable alerts automatically notify authorities when single or clustered hazards exceed severity thresholds. City planners and policy makers gain access to interactive dashboards featuring dynamic maps,

heatmaps of top hazard zones, supporting data-driven decisions for infrastructure improvements & strategic planning. By integrating user authentication, role-based access, and open data export capabilities, the system balances accountability with wide accessibility. Through iterative development, usability testing, and performance evaluations, SafeRoads Navigator aims to demonstrate a measurable reduction in reporting latency, enhanced situational awareness of road users (which can eventually lead to reduced road incidents), and provide road safety authorities with a data-driven tool, laying the groundwork for a smarter, safer urban transport network.

II. Introduction

2.1 Background of the Study

Road safety remains a pressing public health and transportation concern worldwide. Traffic crashes are one of the leading causes of death and injury, accounting for approximately 1.19 million fatalities globally each year, of which, 92% occur in low- and middle-income countries (WHO Report, 2023). In New Zealand, the NZ Transport Agency reports 341 road fatalities and 2,442 serious injuries, averaging one death and seven serious injuries per day in 2023. This makes New Zealand's rate of road deaths amongst the worst in the OECD (NZTA, 2024), with a rate of 6.5 deaths per 100,000 population (NZAA, 2025). Road hazards such as potholes, debris, poor signage, and other safety risks pose serious threats to motorists, cyclists, & pedestrians alike. Traditional approaches to identifying and fixing these hazards rely on periodic inspections or public complaints, which are often slow and reactive. Swift

awareness of road hazards can significantly reduce secondary accidents and improve emergency response times (Young et al. 2019).

Crowdsourced data has emerged as a promising solution to provide real-time, widespread coverage of road conditions. Such community-driven reporting can fill critical information gaps, especially for hazards that might not trigger immediate official action, i.e. a fallen tree branch or localised flooding. Despite their benefits, crowdsourced hazard systems face challenges regarding the reliability and accuracy of user-contributed data (Leal et al. 2017). False or outdated reports can occur, whether due to user error, changes in conditions, or even malicious misuse. Ensuring data integrity is essential for maintaining user trust and effectiveness of the system. There is a need for more robust verification to minimize false alarms while preserving the speed and breadth of crowdsourced inputs. Thus, the need for dual vetting solution.

This capstone project, **SafeRoads Navigator**, proposes a novel approach to real-time road hazards monitoring by combining the strengths of crowdsourced data with a dual vetting approach. In this system, hazard reports from the public are subjected to two layers of vetting: first by road users through peer confirmations (aka Peer-vetting in the context of this study), thru upvotes or flags and second by road safety authorities (aka Authority-vetting) to ensure higher accuracy. The goal is to create a reliable, user-driven road hazard alert network.

2.2 Problem Statement

There is a need for a real-time road hazard monitoring system that maximizes the advantages of crowdsourced data (speed and breadth of coverage) while minimizing misinformation through effective validation. Additionally, there is currently no comprehensive, real-time platform for road hazard management

that connects the general public, road maintenance authorities, and city planners on a common information system. As a result, many road hazards either go unreported, unassessed, or are addressed too slowly, leading to avoidable accidents, infrastructure damage, and inefficiencies. Traditional reporting channels (phone hotlines, periodic surveys) are reactive and often fail to capture the full scope or urgency of issues. Furthermore, data that is collected (through isolated apps or studies) remains under-utilised for long-term safety improvements due to fragmentation and lack of analytical tools.

The core problem then can be summarised as: “How can we design and implement a real-time road hazard alert management platform using crowdsourced reports enhanced with a dual-layer vetting process to deliver timely and trustworthy information to all stakeholders?” The absence of an integrated solution means that cities struggle to prioritize repairs effectively, road users lack awareness of known danger spots, and decision-makers do not have consolidated data on which to base infrastructure improvements. We need a system that not only gathers hazard reports from road users in real time, but also ensures those reports are reliable, promptly acted upon, and aggregated into meaningful insights for future road safety strategies.

SafeRoads Navigator directly addresses these problems by providing a unified platform for road hazards reporting, visualization, alerting, and analysis. By doing so, it bridges the communication gap between citizens experiencing road hazards and the authorities responsible for fixing them, while also creating a valuable data repository for planners and researchers for present and future use. In essence, the project seeks to transform the current reactive and fragmented approach into a proactive, data-driven, one-stop-shop, road hazard management process.

To tackle the stated research gaps above, the project will be guided by the following research and design questions:

1. How can real-time trust and reputation mechanisms be designed to automatically validate and filter crowdsourced road-hazard reports without significantly delaying the reporting process?

Justification: This addresses the first gap identified which is about increasing the reliability of crowdsourced data by minimizing false or malicious submissions.

2. What architectural and workflow models enable the seamless integration of an end-to-end road hazards management within a single platform?

Justification: This addresses the second gap identified by creating a unified solution covering the aspects of crowdsourced reporting, multi-layer verification, operational response, and strategic trend analysis.

3. In what ways can built-in data analytics and open-data interfaces be embedded into crowdsourced hazard reporting systems to facilitate immediate policy-relevant insights and stakeholder collaboration?

Justification: This addresses the third gap enumerated above by providing data analytics and enabling the proposed solution to export datasets on road hazard incidents to be used by city planners, researchers, maintenance teams, and other concerned relevant parties.

2.3 Objectives of the Study

The objectives of SafeRoads Navigator are defined to address the identified problems / research gaps stated above. The successful completion of the capstone project will be measured against the three objectives outlined below:

1. To design and implement a real-time road hazards system that introduces a mechanism (1) for road users to report road hazards and verify (by upvoting or flagging) existing hazard reports, and (2) allow

road safety stakeholders to filter out false or malicious submissions without introducing perceptible reporting delays, thus improving the accuracy and trustworthiness of crowdsourced road hazard data.

2. To develop a unified, end-to-end road hazard management platform prototype that integrates (a) crowdsourced road hazard reporting, (b) multi-layer verification workflows, (c) operational response coordination, and (d) strategic trend-analysis capabilities.
3. To embed interactive data analytics dashboards and open-data interfaces within the hazard management system—enabling stakeholders, i.e. city planners, maintenance teams, & researchers, to generate policy-relevant insights, export cleaned datasets, and to assess the platform’s impact on decision-making efficiency and possibly, inter-agency collaboration.

2.4 Scope and Limitation

SafeRoads Navigator is a web-based application designed to empower road users and road authorities with a centralized and interactive interface for the reporting, moderation, analysis, and management of road hazards. This capstone project was developed with the following scope in mind:

1. User Registration and Authentication – support for multiple user types
2. Hazard Reporting – road users can submit road hazard reports with details including description, photo of the hazard, type, etc.
3. Hazard Moderation – road authorities can view, approve, or reject submitted hazard reports
4. Analytics and Reporting – visualization of hazard data using maps and charts; exportable hazard reports
5. Maintenance Dashboard – for system maintenance needs
6. Hazard Mapping – radius-based display of nearby hazards on an interactive map

While SafeRoads Navigator has achieved the objectives and functionalities set forth in this study, several limitations were identified in the course of final implementation:

1. Geocoding Accuracy – there may be times when geolocation fails to capture the actual location of the reported hazard. Though this may be the case, the coordinates captured have good accuracy
2. No mobile app integration – this is identified as a future work for this project
3. Communication between road safety authorities and road maintenance officers is not currently handled in the system
4. Security considerations – while encryption and hashing are applied to sensitive data (PIIs), further security hardening such as HTTPS enforcement should be implemented

III. Literature Review

3.1 Overview of Existing Research

We all want to make transportation greener, smarter and more sustainable. However, road authorities everywhere face increasing challenges in managing traffic and infrastructure to provide critical services to citizens. It is therefore vital that cities have an integrated solution that allows them to monitor and manage traffic infrastructure & transport operations in real time. Numerous studies and projects in recent years have explored the use of crowdsourced data to improve road safety and traffic management. Early research recognised the potential of social and mobile platforms in capturing real-time traffic information. For example, in Christou et al. (2023), it highlighted that a government-backed, national-level crowdsourcing app can achieve high

uptake and rapid response times, which could lay the groundwork for broader smart-city deployments. This work reported that citizen-generated data offered more frequent and geographically comprehensive coverage, particularly on secondary roads, than traditional inspections or vehicle-based surveys. Thier app, FixCyprus—a national mobile app in Cyprus that collects citizen reports of road infrastructure defects, demonstrated the impact of a government-adopted, country-wide crowdsourcing tool. In Olma et al. (2022), it identified that traditional road-safety work in Germany and the EU largely depends on collision black-spot analyses, which are inherently reactive, slow to identify hazards (since collisions are rare events), and *ethically problematic because human harm must occur before action is taken*. They introduced a web-based platform where road users can mark “danger spots” on an interactive map, providing details on the type of hazard and road users at risk, i.e. poor visibility affecting pedestrians. This approach treats the crowd as “underestimated experts”, asserting that daily road users experience conditions first-hand and can alert others and authorities to risks before accidents occur. Indeed, the study found optimism among stakeholders that active public participation could serve as an early warning system in road safety work, though they cautioned about the subjective nature of such reports and potential misuse. It concluded that systematically collected danger-spot submissions, especially those with higher user engagement, can serve as timely, cost-effective indicators for proactive road-safety interventions. It goes further by saying that danger spots that garnered more user interactions (clicks, supports, comments) were significantly more likely to be confirmed as hazardous, indicating that collective attention reinforces report accuracy. In Telima et al. (2023), it mentioned crowdsourcing as a proactive tool, citing that systematic collection and analysis of user-generated incident reports provide an early-warning mechanism—capturing risks before collisions occur & complementing traditional crash records. It investigated the

use of crowdsourced incident reports (including not only collisions but also near-misses and infrastructure issues) to analyze pedestrian safety in urban areas. One novel result of the project was the identification of a few pedestrian hazard locations in the area of study. A complementary study by Ibtissem et al. (2022) was proposed in Tunisia to leverage data from multiple crowdsourcing platforms (mobile+web) for urban road safety improvements. The R-Secure system features two main functionalities: first, collecting rich road anomaly data (including images and GPS location) from citizens, and second, integrating expert input and analyses to enhance road safety outcomes.

In parallel to manual user reports, technology is being used to automatically detect road hazards, which can complement crowdsourcing. An innovative approach by Cafiso et al. (2022) used bicycles and e-scooters as probe vehicles to monitor road pavement conditions via built-in smartphone sensors. By collecting accelerometer data as these micro-mobility devices traversed urban roads, the researchers could identify road surface anomalies. Notably, the e-scooter and bike data were able to detect various severities of cracks and even potholes that did not register in traditional car-based road condition surveys.

There were also more focused studies done on the subject of road safety. One in particular is an IoT-based pothole detection by Desai et al. (2024) where they developed an application that aimed at revolutionizing road safety by proactively identifying and preventing accidents caused by potholes. This research leverages the integration of Internet of Things (IoT) technology to provide real-time updates and alerts to drivers, significantly mitigating the risks associated with pothole-induced accidents. This was achieved by combining LiDAR, radar, GPS, & accelerometer data to produce reliable pothole detections across varying road conditions & speeds. A similar study was done in Bhoyar et al. (2023). In this study, they devised a pothole

detection system that utilizes an ultrasonic sensor mounted on a vehicle to measure the depth of potholes. The sensor continuously measures distance to the road surface and readings exceeding an 8cm threshold are flagged as a potential pothole. One major limitation of this study, however, is that it solely focused on pothole detection. In Yang et al. (2021), the research employed technology that collects pothole data using smartphone accelerometers and cameras mounted on vehicles to obtain images of road surface anomalies. The images collected are then analysed with an image recognition-based convolutional neural network model to determine if the road surface scanned contains potholes. The final verified pothole images are then transmitted to a server along with the vehicle's GPS data. Then, the system generates road hazard information in the form of a four-level index indicating the road risk. This road hazard index can better assist road safety authorities in establishing road maintenance plans and planning their costs. In a similar study, Pena-Caballero et al. (2020) focused on a limited scope of road hazard classes, i.e. Manhole, Pothole, Blurred Crosswalk, & Blurred Street Line. It trained a segmentation model, using 1,700 annotated images from the over 4,000 images captured, that recognizes the four classes mentioned above and achieved great results with this model allowing the machine to effectively and correctly identify and classify the four classes in an image. Another recent study is the use of semantic web and ontologies to manage road hazard information. Kindo et al. (2024) introduced an ontology-based system to handle community-reported potholes, aiming to formalize how pothole data is described and shared between different systems. This ontology-based, community-assisted system bridges the gap between grassroots pothole detection and formal municipal maintenance, offering a scalable, cost-effective, and interoperable solution for improving road safety. Main limitation is that it focused primarily on pothole detection despite the many types of potential causes for road hazard incidents.

A body of literature has focused on evaluating the accuracy of crowdsourced road hazard data. In one study, danger zones were chosen from among four German cities. Thereafter, a quasi-randomised audit of 77 user-reported danger spots was done and over half were confirmed to have infrastructural deficits, i.e. true hazards, while about one quarter were classified as “uncertain” due to insufficient details. These findings (user reports and on-site inspections) were then compared with actual police collision data, Olma et al. (2022). In a related study, R-Safety, devised a mobile crowdsourcing platform that allows road users to report their “road safety feeling indexes” in real time. Their platform focuses on subjective safety perceptions and even lets users report traffic violations they witness, with the goal of informing citizens about risky areas through a vulnerability map (Khedher et al. 2022).

While the related literature presented thus far demonstrates clear benefits to using crowdsourced road hazard information, it also consistently highlights challenges that SafeRoads Navigator aims to address—data quality control. First, an open contribution model, i.e. crowdsourced, raises some concerns about misinformation. Also, these community-driven validations can be slow or insufficient, i.e. areas with few users. The present study suggests that combining community validation with an additional layer—moderator oversight (aka dual vetting)—could improve reliability.

The literature shows a strong foundation for SafeRoads Navigator. Crowdsourced data is a powerful tool for real-time road hazard monitoring, offering speed and coverage advantages over traditional methods. Multiple studies have proven that user reports are generally reliable (with low false alarm rates) and can even save lives by enabling faster responses. It can also provide rich datasets for analysis. However, to fully realize this potential, a system must tackle data redundancy and trustworthiness as well. These insights directly inform the SafeRoads Navigator project. The proposed system

builds on the success of crowdsourcing seen in prior research and explicitly addresses the identified gaps by introducing a structured dual vetting process for quality assurance. In the next section, I will define the specific problem this project will solve in light of the literature discussed and outline our research questions.

3.2 Identified Gaps

From the above review, several gaps in current solutions become evident. On one hand, crowdsourced reporting can dramatically improve the timeliness and coverage of road hazard information. On the other hand, ensuring the accuracy and credibility of these reports remains an open challenge. Current navigation apps, for example Waze, & research prototypes largely rely on single-layer crowd validation—if enough users report or confirm an incident, it is considered credible. This approach, while useful, has its limitations. Minor hazards may not get multiple confirmations before they cause harm, and false reports can linger if no moderator intervenes. There is also a gap in seamlessly integrating crowdsourced hazard data with verification mechanisms in a way that is fast & reliable. Also, community engagement features such as crowdsourced verification, voting, or commenting are not universally present. The EDDA+ danger spot platform did implement support/upvote mechanisms and discussion threads (Olma et al., 2022), but many other systems do not leverage the crowd beyond the initial report. This can lead to issues with data quality and prioritization: for instance, without community feedback, authorities might not know which of the many reports are most urgent or if some reports are duplicates. Thus, there is a gap in using the crowd not just as reporters but as filters and enhancers of the data.

Second, many crowdsourced hazard reporting systems focus on one aspect of the problem. Some are focusing only on one particular type of road

hazard, i.e. pothole detection. Some are primarily for reporting and notifying authorities, i.e. FixCyprus focuses on submitting reports to public works (Christou et al., 2023), while others emphasize collecting data for analysis or public awareness, i.e. the danger spots map in Germany for safety research (Olma et al., 2022). There is a lack of an integrated platform that seamlessly supports end-to-end hazards management—from immediate reporting and verification, through operational response, to strategic trend analysis—in a unified system.

Finally, the siloing of data is a concern. Data collected by one platform often isn't readily accessible for other purposes. The Korean citizen science study had to perform significant data processing (text mining) on the complaints to evaluate maintenance efficiency (Kim et al., 2023), implying that the system used to collect complaints wasn't inherently providing those insights. Likewise, danger spot data needed to be combined with collision and kinematic data externally to derive hazard scores (Olma et al., 2022). This suggests a gap in MIS design: hazard reporting platforms could better incorporate data analytics and open data principles from the start. **SafeRoads Navigator** will attempt to close this gap by including dashboards and data export features that make analysis easier and allow stakeholders (planners, researchers) to directly use the information.

The gaps identified are:

1. Reliability of crowdsourced data – how to filter out incorrect or malicious reports without losing the benefits of real-time community inputs. The literature indicates trust mechanisms are needed, but an effective implementation in the road hazard domain is lacking;
2. Lack of a unified system covering reporting, response, and analysis;
3. Limited use of crowdsourced verification to improve road safety initiatives as data is not readily analysed or shared for policy use.

These gaps inform the problem statement and objectives of this capstone project.

3.3 Motivation of the Study

Living in Singapore for more than ten years before coming here to New Zealand, I have seen and experienced first-hand the impacts of Information Technology and sound policy integration can do to urban transportation & safety, with the city-state being well recognized internationally for its efficient and data-driven transport management systems. Relocating here in Auckland has allowed me to observe the very apparent differences in road infrastructure, road hazards management, and overall transportation system efficiency. While Auckland's overall transportation system has many strengths in certain areas, there remain significant improvements and opportunities that can be made to further improve its current state.

At the same time, I do have a personal interest in the area of Intelligent Transport Systems (ITS). As an IT Practitioner, I have always been intrigued and fascinated by how Data Science and IT can be applied to modern day mobility and transportation safety challenges. My academic journey in YooBee also furthered this interest.

The above, coupled with a personal belief that as citizens, we have to strive to make positive contributions to our community, form the bases and motivation for undertaking this research. By designing and implementing a crowdsourced & dual-vetted road hazard management system, I aim to contribute an innovative solution that can empower both road users and road safety authorities in efficiently managing road hazards. This research study not only addresses an academic requirement for the completion of my programme. It is also a technical endeavor that aligns with my professional

experience as an IT Practitioner with a commitment to create systems that can have a positive impact on the users for which the system is intended. Ultimately, this research project is driven by the conviction that ITS innovation, rooted in global best practices and informed/sound innovation, can make a tangible difference in my new community in general and the lives of my fellow Aucklanders, in particular.

IV. Design, Methodology, & Architecture

4.1 Solution Use Cases

The identified use cases for the proposed solution are as follows:

1. Emergency road reporting – a road user notices a large pothole that could cause an accident and submits a report through the mobile web interface. The system captures the location automatically or allows the user to enter it manually. The report is then shared with all road users and plotted on a real-time map for awareness.
2. Crowdsourced validation (Peer vetting) – after a report is submitted, other users can view the hazard on the grid and map, upvote or flag a hazard whether it is genuine or not, improving data reliability. The hazards shown in the grid follow a certain radius as specified by RADIUS environment variable.
3. Road safety authorities' moderation (Authority Vetting) - road safety authorities can approve or reject a road hazard report based on the information provided, i.e. image uploaded, GPS coordinates, etc., together with the number of upvotes or flags a report has received. Reports can also be approved or rejected on the basis of inspections or audits conducted of these road incidents.

4. Peer vetting and authority vetting comprises the dual vetting approach proposed by the study.
5. Traffic and road safety analysis – city planners use the Analytics dashboard to view heatmaps of hazard reports. They identify the top 10 areas with frequent reports and engage road maintenance authorities to schedule repairs or road safety audits in those locations. A useful metric included in the Analytics dashboard is the average resolution time, which could be a barometer for efficiency for city planners.
6. Automated alerts to road safety stakeholders – road hazards reported as critical and receiving the required minimum number of upvotes (specified by `UPVOTE_THRESHOLD` environment variable) are automatically upgraded in terms of status (i.e. Authority Notified) and automatically sends notifications to the local road authorities' team for attention and rapid response.
7. Historical trends for policy making – the system aggregates data over time, providing downloadable reports and visualizations that help local authorities understand long-term road hazard trends and plan infrastructure improvements accordingly.

Road hazards reported will follow a defined process flow and lifecycle as below.

Figure 1: Road Hazard Identification and Treatment (process flow)

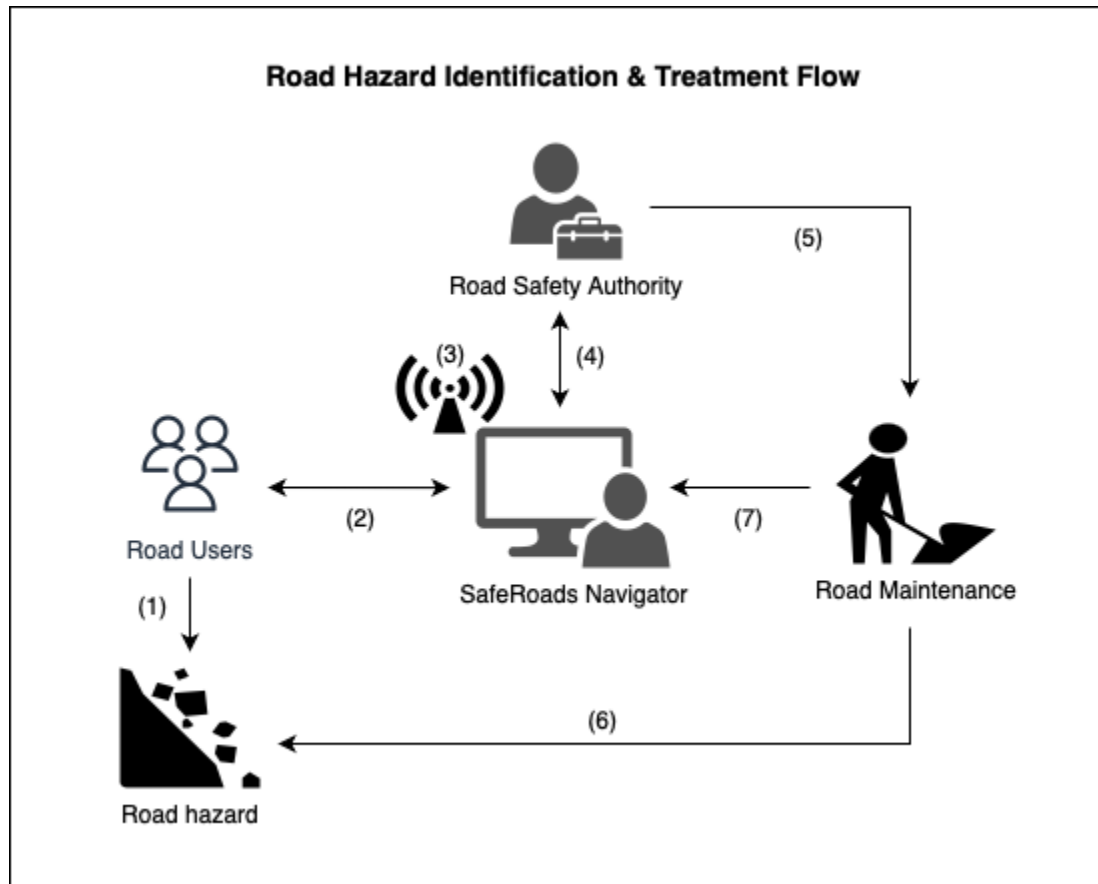
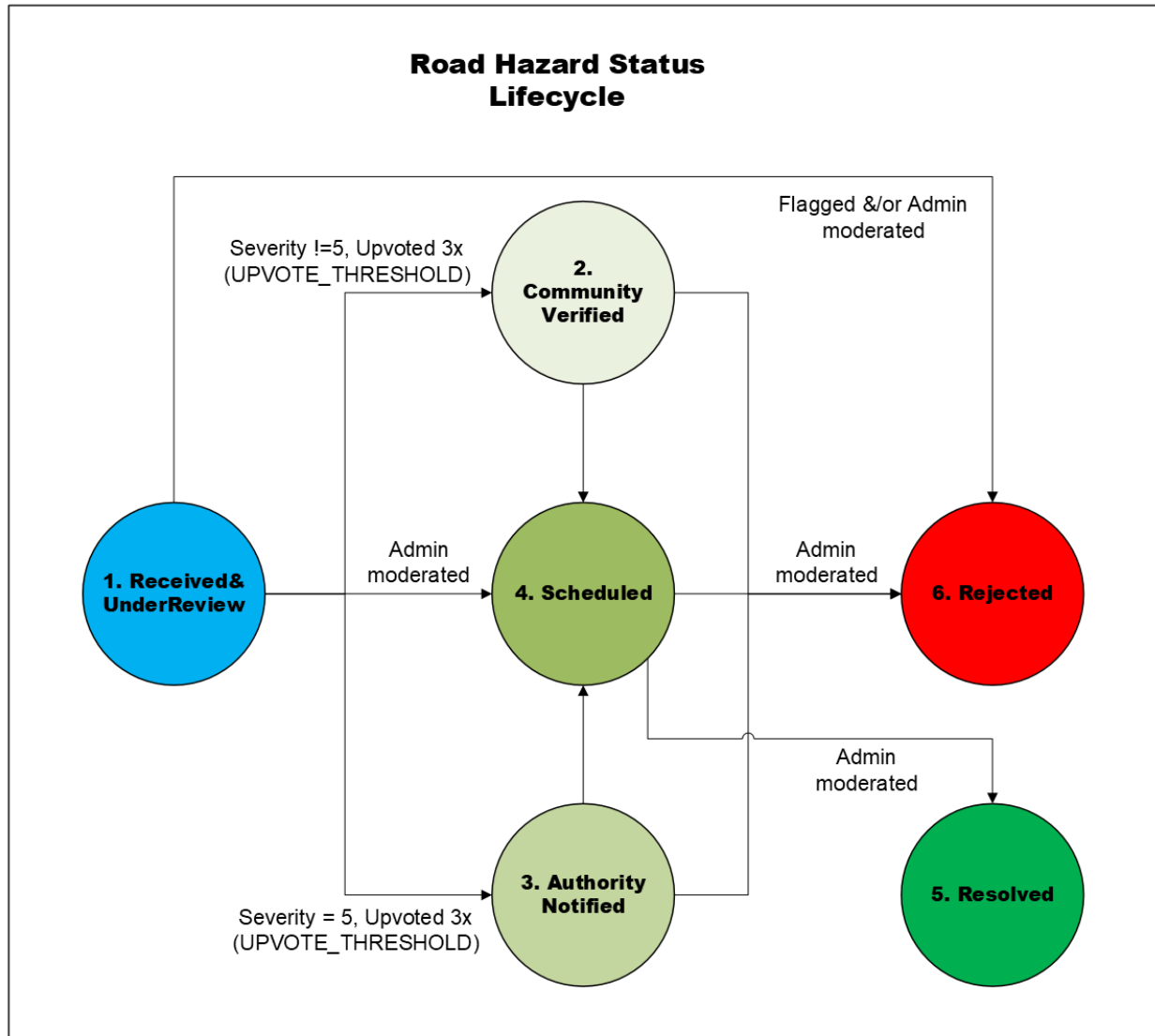


Figure 2: Road Hazard Status Lifecycle Diagram



4.2 Development Methodology

This capstone project employs an action research type of study with a design and development methodology typical of software engineering projects,

enriched with elements of design science and an Agile approach to software development. The approach can be outlined in several phases:

1. **Requirements Analysis** – based on the literature review and the defined use cases, I will detail the functional requirements (what the system should do) and non-functional requirements (performance, security, usability, etc.). This will involve creating use case diagrams and user stories, i.e. “As a driver, I want to quickly report a road hazard with a photo so that others are warned and it gets fixed”. I will also identify the data requirements—what information each report will contain (hazard description and type, location, timestamp, reporter info, etc.) and how data flows between frontend, backend, and any external services.
2. **System Design and Architecture** – using the requirements identified above, I will design the system architecture. This will follow a client-server model: the frontend application will communicate with a backend Node.js Express server via RESTful API calls using JSON data. I will design API endpoints for operations such as submitting a new report (`POST /report-hazard`), retrieving hazard data (`GET /fetch/hazards`), voting on a report (`POST /upvote-hazard`), and admin actions (`POST /admin/approve-hazard` to approve a hazard report, `POST /admin/reject-hazard` to reject a reported hazard, etc.). A relational database schema (RoadHazard.sqlite) will be designed to store users, hazard reports, votes, admin actions, geospatial information and any other related data. The Google Maps API will be incorporated for the map interface; this requires obtaining an API key and the Heatmap layer for density visualization. A particular focus in design will be on the dual vetting logic, i.e. designing how a hazard

status transitions from “Received&UnderReview” to “Resolved” or “Rejected” and what triggers that (number of confirmations or admin approval).

3. **Implementation (Iterative Development)** – as an agile practitioner, I will adopt an Agile iterative development methodology, aiming to produce incremental prototypes that can be tested and refined. For instance, in the first iteration, I will implement core reporting and mapping without all features, then gradually add verification, dual vetting, alerts, analytics, and reports in subsequent iterations. This approach allows for early feedback loops for prototyping, modifications, and fixes.
4. **Testing and Evaluation** – throughout development, I will test each component. This includes unit testing of backend logic and integration testing. I will also perform user acceptance testing: simulate a user reporting a hazard and ensure it appears on the map; simulate multiple users voting on a report and see that it escalates appropriately; test that an alert email is sent when it should, etc.
5. **Project Management and Milestones** – I will break the work into tasks with a rough timeline (see Project Plan below). Version control (Git) will be used to manage code, and regular progress meetings with the Supervisor will ensure I stay on track. If certain aspects prove too complex, i.e. implementing a fully automated clustering algorithm for alerts, I will adapt by simplifying the approach.

4.3 System Architecture

SafeRoads Navigator is proposed as a web-based application implementing the ideas discussed previously. The SafeRoads Navigator system will be composed of several integrated components, forming a cohesive solution. On the client side, a javascript-based web application will serve as Hazard Reporting and Viewer. On the server side, a Node.js application with an Express framework will act as the API Server and Processor (Figure 3), connected to a database (Figure 4) that serves as the Hazard Data Repository. Surrounding these core components, there will be external integrations: the Google Maps API for mapping, notification services, i.e. Gmail SMTP with Nodemailer, for sending out alerts, and OpenStreetMap Nominatim for geocoding services.

Figure 3: Architecture Diagram

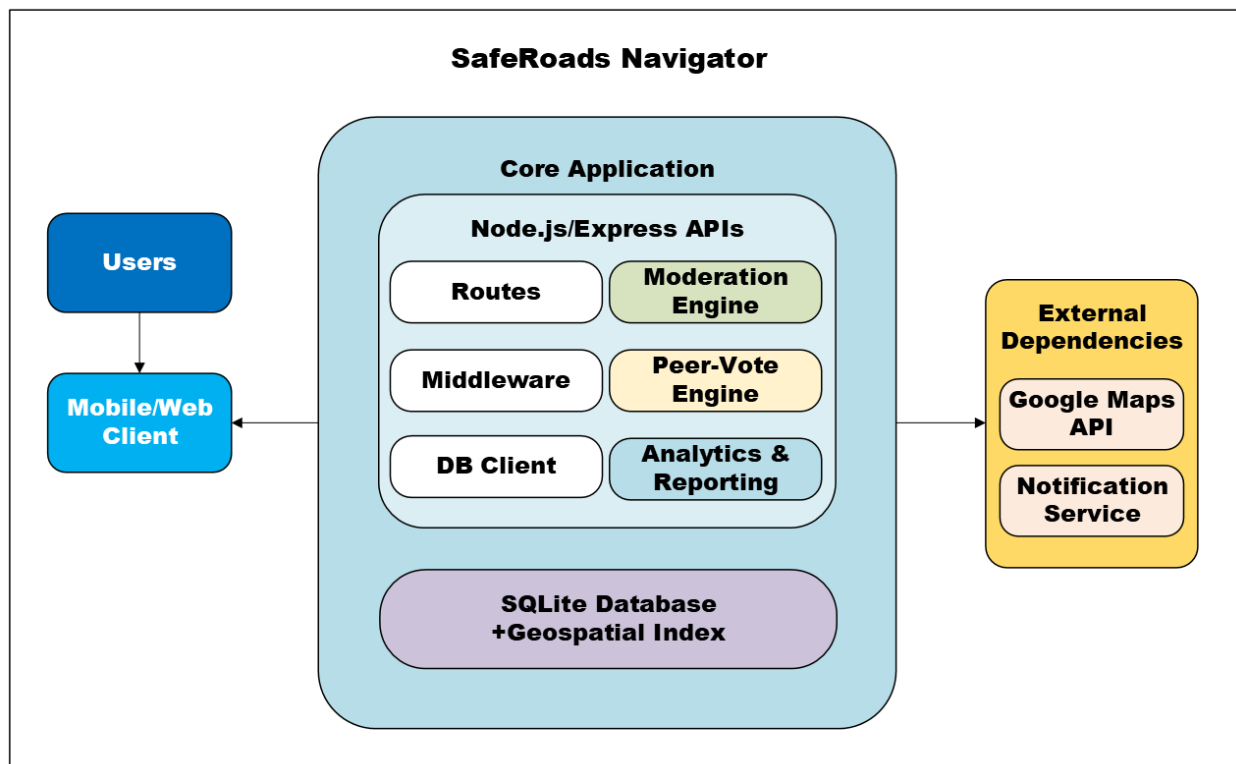
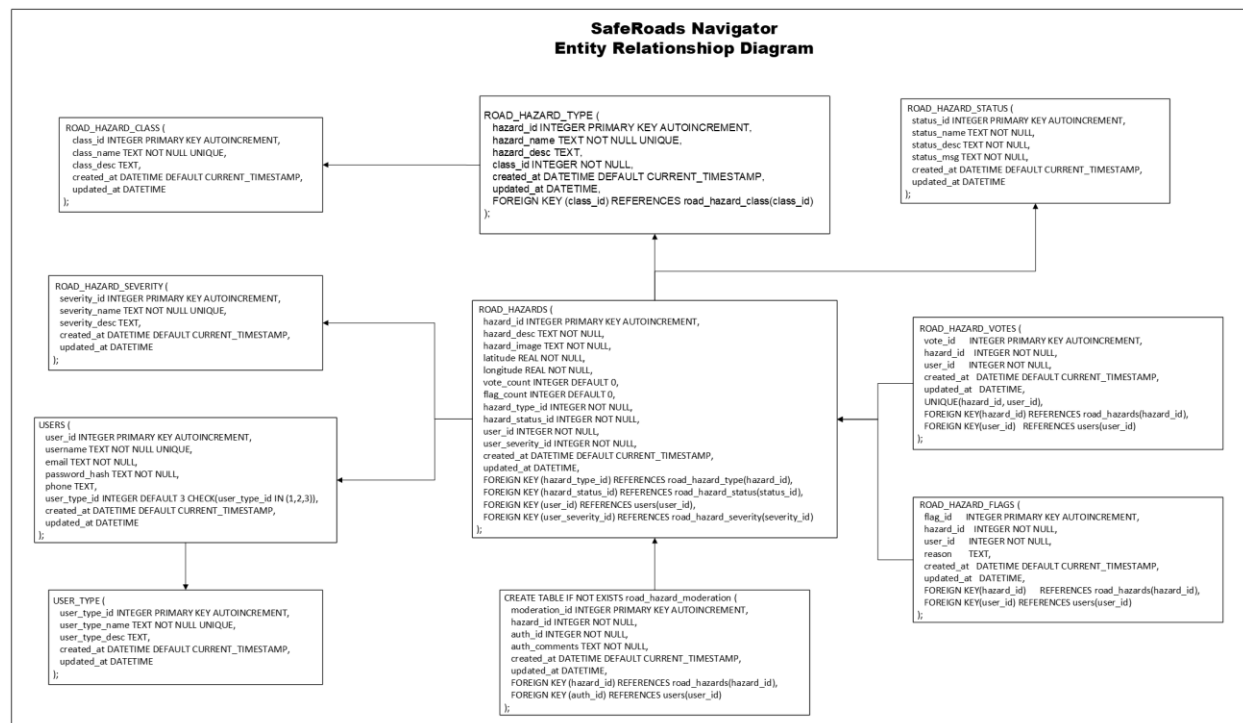


Figure 4: Database Design (ERD)



4.4 Technology Stack and Architecture

Following a modular architecture approach, the project will employ the specified technology stack below:

1. Frontend

- HTML5 & CSS3 – the client-side user interface is built using modern HTML5 and CSS3. This ensures the development of a responsive design and accessibility.
- Bootstrap 5 – this component was extensively used throughout the project due to its pre-built UI components. This made the development of the responsive layouts and grid systems a breeze. The use of bootstrap icons provided enhanced visual interfaces.
- JavaScript (Client-side) – Javascript was the logical choice for handling client-side logic, event handling, DOM manipulation, and AJAX calls to backend APIs, due to its tight integration with HTML.
- Chart.js – this component is used to deliver the functionality of the Analytics Dashboard for delivering the interactive data visualizations you see there, such as the bar and pie charts.
- Google Maps API – this third-party component is used to provide the interactive mapping functionality of SafeRoads Navigator
- Nominatim – a free & open-source geocoding software using OpenStreetMap data
- Leaflet.js – this component powers the interactive heatmap in the Analytics Dashboard that allows road users to view hazard locations and concentrations using OpenStreetMap data.
- Socket.IO Client – enables integration with the backend which allows for real-time updates on the dashboard, such as live upvotes, flags, and moderation actions.

2. Backend

- Npm – package manager for Javascript programming language
- Node.js – the backend runtime environment was developed using Node.js.

- Express.js – is a minimal and flexible Node.js web application framework that handles HTTP requests, routing, session management, and middleware integration for RESTful API web development.
- SQLite3 – is used to store application data. This was chosen because it is very lightweight and very ideal for rapid development and ease of deployment. For the photo upload, instead of saving the uploaded image in the database, a directory path is created instead because of the inherent limitations in SQLite.
- Other backend libraries:
 - Bcrypt – used for secure password hashing
 - Body-parser – used to parse incoming HTTP requests
 - Express-session – used for session management as well as user authentication
 - Multer – used for handling file uploads
 - Dotenv – used for managing application persistent environment variables
 - Nodemailer – used for sending email notifications

3. Security, Privacy, and Best Practices

- The application stack incorporates robust authentication and authorization using sessions and password hashing
- Multer and body-parser handle the secure handling of file uploads and form submissions
- Application variables are managed with dotenv to separate configuration settings from system code
- Google Maps API key management will be handled securely (not exposing keys in public code)
- User interfaces will be behind a login

- All personally identifiable information, such as emails and contact numbers, will be encrypted while user passwords will be hashed using AES-256 cryptographic algorithm.

V. Development Plan & Implementation

5.1 Project Plan and Timeline

The SafeRoads Navigator project is planned to be completed within a four-month period, which roughly translates to 15 weeks. To ensure timely completion, the following timeline with milestones is proposed (aligned with the methodology steps):

1. Month 1 (Weeks 1–4) – Planning and Design (complete requirement analysis and finalize the system design. This includes the literature review (already in progress), drafting use cases, designing the database and API, and creating wireframes for the UI. By the end of Week 4, a design review will be held with mentors to validate the approach.
 - Milestone(s): Requirements & Design Document finalized, development environment set up (Node.js server skeleton, SQLite schema created), and use cases listed and identified
2. Month 2 (Weeks 5–8) – Core Implementation (develop the core features (Iteration 1). By around Week 6, aim to have the hazard reporting and mapping functional: a user can submit a hazard and see it appear on the map (persisted in the DB). By Week 8, integrate user confirmation functionality (peer vetting) and have a basic dual vetting logic working in a rudimentary form.

- Milestone(s): By end of Month 2, internal demo of the system showing multiple users reporting and confirming hazards, with the map updating live.
- The following server endpoints need to be developed:
 - `GET /fetch/hazards` to get hazards within a certain distance (RADIUS) that have reported to the system
 - `GET /fetch/allhazards` to get all hazards reported by road users
 - `POST /report-hazard` to report a road hazard
 - `POST /upvote-hazard` to upvote on a reported hazard
 - `POST /flag-hazard` to flag a reported hazard
 - `GET /admin/load-hazards` to load all hazards for moderation
 - `POST /admin/approve-hazard` to approve (moderate) upvoted hazards
 - `POST /admin/reject-hazard` for rejecting hazards not meeting criterias
 - `GET /api/analytics` for generating analytics/statistics
 - `GET /api/reports/weekly` for generating statistics on a weekly basis
 - `GET /api/reports/monthly` for generating statistics on a monthly basis
 - `GET /api/reports/unresolved` for generating statistics of all unresolved/open hazards
 - `GET /api/reports/resolution-rate` to get the latest resolution rate for all resolved hazards
 - `GET /api/reports/resolved-rejected` for generating the list of all resolved and rejected hazards so far

3. Month 3 (Weeks 9–12) – Feature Completion and Refinement (Iteration II – implement remaining features and the full dual vetting rules. Build the admin dashboard in this phase and enforce the workflow for hazard status changes, i.e. unverified → verified or rejected). Also implement the heatmap view and any additional niceties (like user accounts or notifications) if time permits. Begin testing with sample data.
 - Milestone(s): Feature-complete beta version of SafeRoads Navigator by end of Week 12. At this point, all primary use cases can be executed on the system.
4. Month 4 (Weeks 13–14) – Final testing, Evaluation, & Finalization (Iteration III – rigorously test the system for bugs, fix any issues, and polish the UI/UX). Conduct the evaluation plan: simulate usage to gather data on how the dual vetting performs. Analyze the results relative to research questions and incorporate findings into the final report. Concurrently, prepare the capstone presentation and documentation.
 - Milestone(s): Week 15, project completion. Deliverables include the final research proposal document (with results from any experiments or tests), the working prototype deployed for demonstration, and a presentation for the academic panel.

5.2 Key Features and Functionality

SafeRoads Navigator has the potential to address many key challenges faced in road safety management. The implementation decision to use Node.js, SQLite, and third-party reusable components like Bootstrap & Google Maps enabled the rapid development of a mid-scale development project within the limited three-month period.

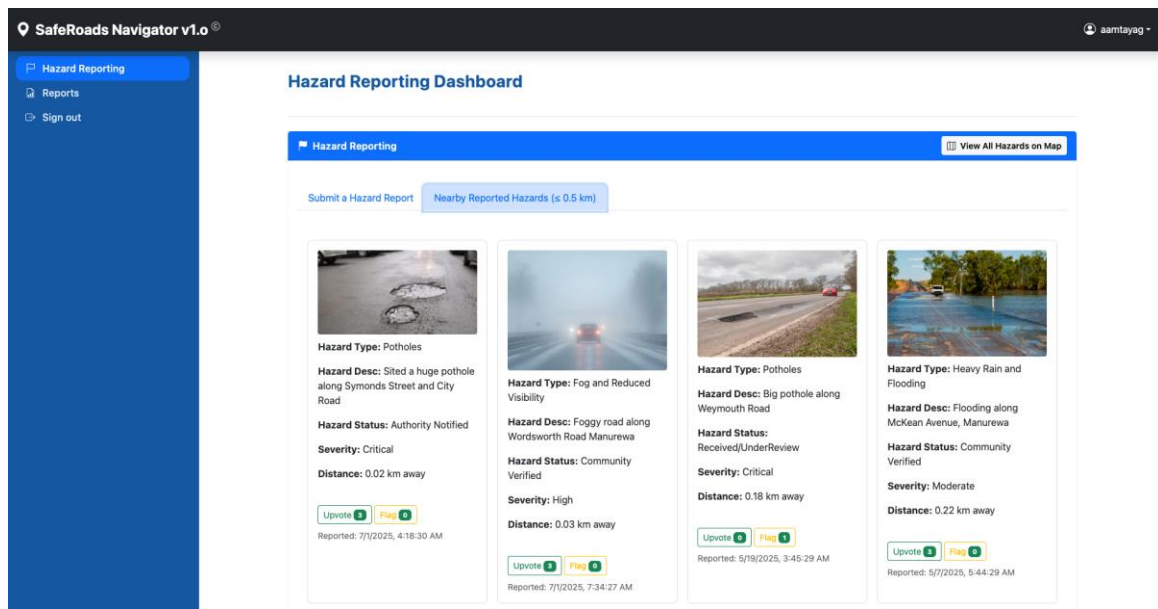
The proposed solution has the following features and functionality:

1. Crowdsourced Hazard Reporting – road users can submit reports of road hazards through a simple and intuitive web interface. A hazard report includes details such as hazard type, hazard description, the location of the hazard (captured automatically via GPS or entered manually), etc. In addition, they will also see other hazards reported to the system that is within a certain distance from their location.

Figure 5: Hazard Reporting Dashboard

The screenshot shows the 'Hazard Reporting Dashboard' within the 'SafeRoads Navigator v1.0' application. The interface features a dark blue sidebar on the left with navigation links: 'Hazard Reporting' (active), 'Reports', and 'Sign out'. The main content area is titled 'Hazard Reporting Dashboard' and includes a 'View All Hazards on Map' button. Below this, there are two tabs: 'Submit a Hazard Report' (selected) and 'Nearby Reported Hazards (≤ 0.5 km)'. The 'Submit a Hazard Report' form contains the following fields: a text area for 'Description of hazard', an 'Image' upload section with a 'Choose File' button and 'No file chosen' text, a 'Type of Hazard' dropdown menu (currently showing '— select —'), a 'Severity' dropdown menu (currently showing '— select —'), and a 'Place Name (no geolocation support)' text field. A prominent blue 'Submit Report' button is located at the bottom of the form.

Figure 6: Hazard Report Grid showing all hazards near the location of a road user, specified by RADIUS



2. Interactive Map / Heatmap with Real-Time Updates – in addition to the hazard grid view, the front-end features an interactive map (powered by Google Maps API) that displays all reported hazards as markers or icons at their locations. The map updates in real time to reflect new reports. It also shows the color-coded severity of the hazard using the following convention: 1) Negligible – green, 2) Minor – blue, 3) Moderate – yellow, 4) High – orange, and 5) Critical – red. A heatmap is also available in the Analytics Dashboard to show areas with high concentration of hazard reports over a period, helping to visualize hotspots.

Figure 7: The Interactive Map showing all hazards reported to the system, with emphasis on the ones that are near the road user

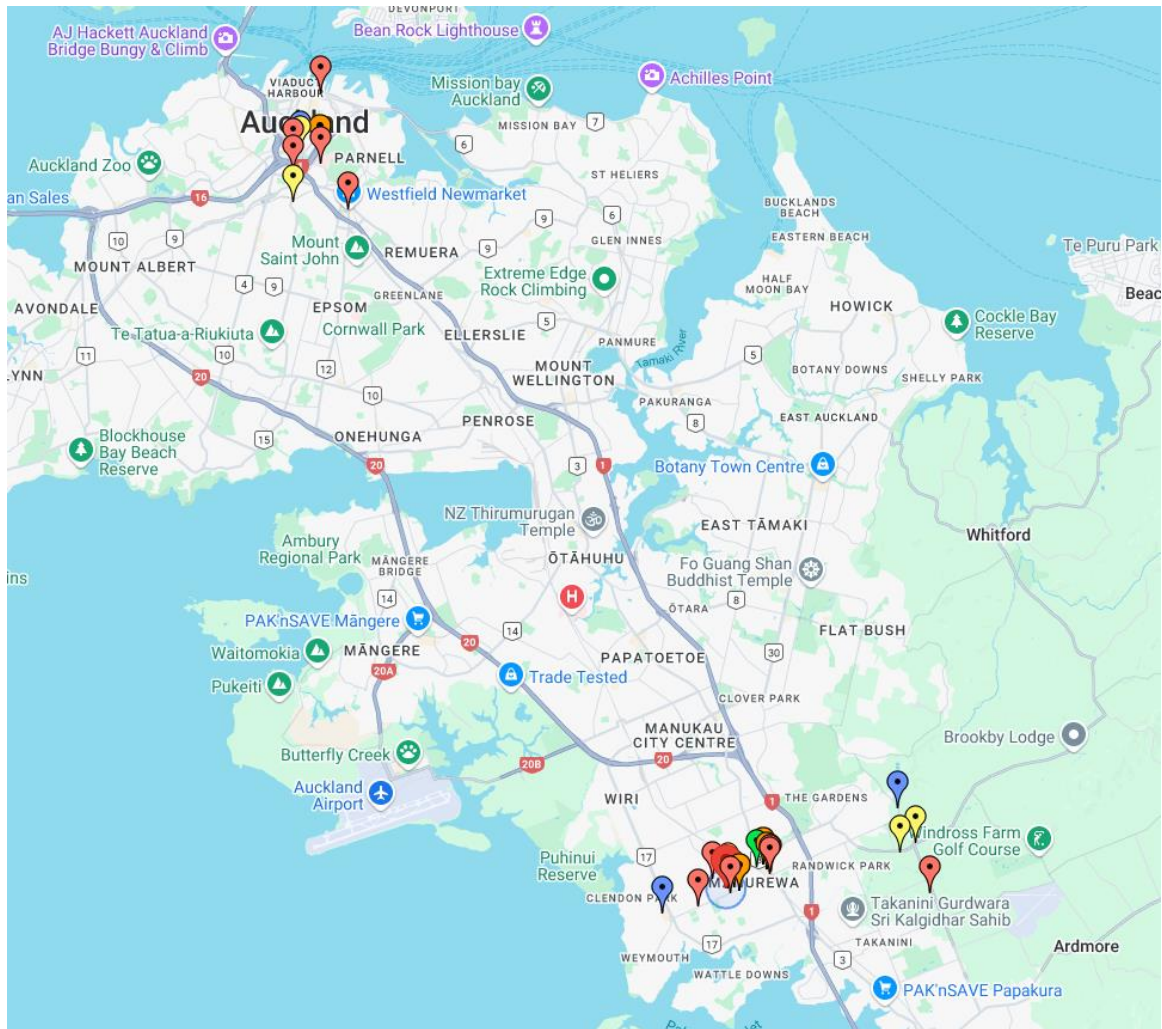


Figure 7.1: Interactive map highlighting hazards within the road user's vicinity

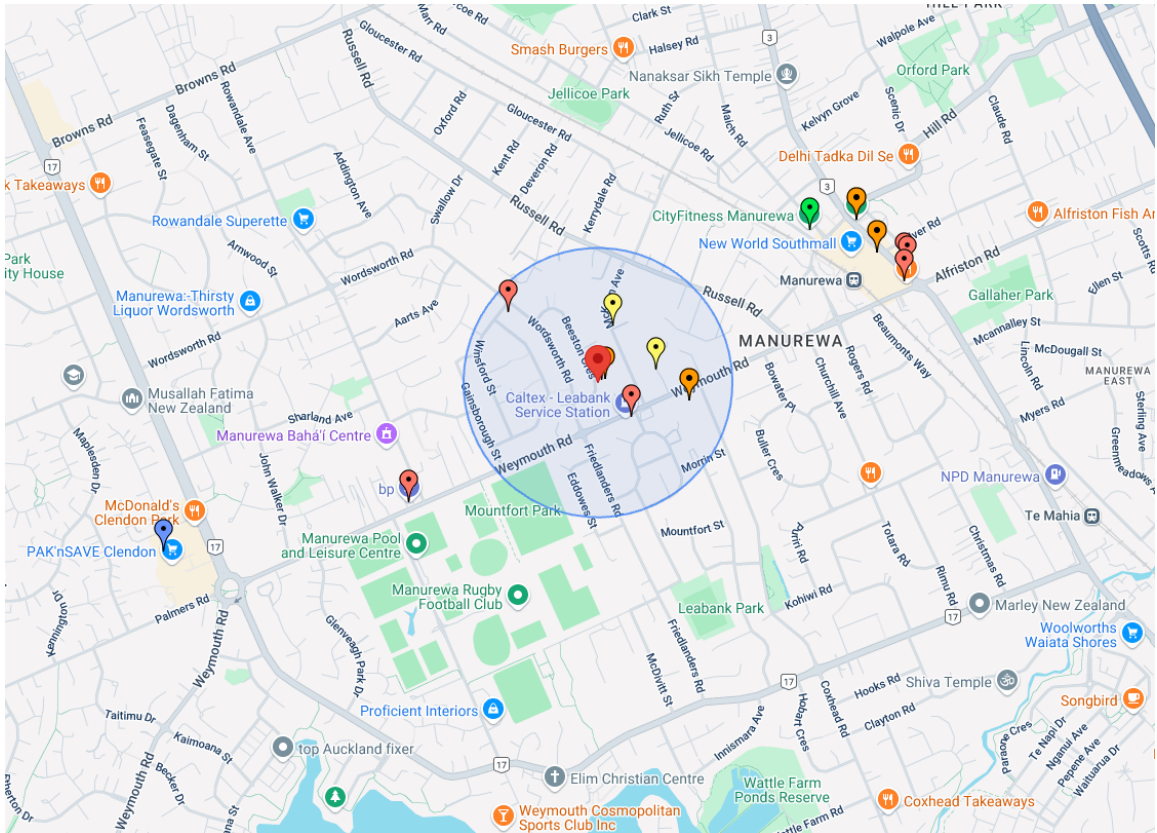
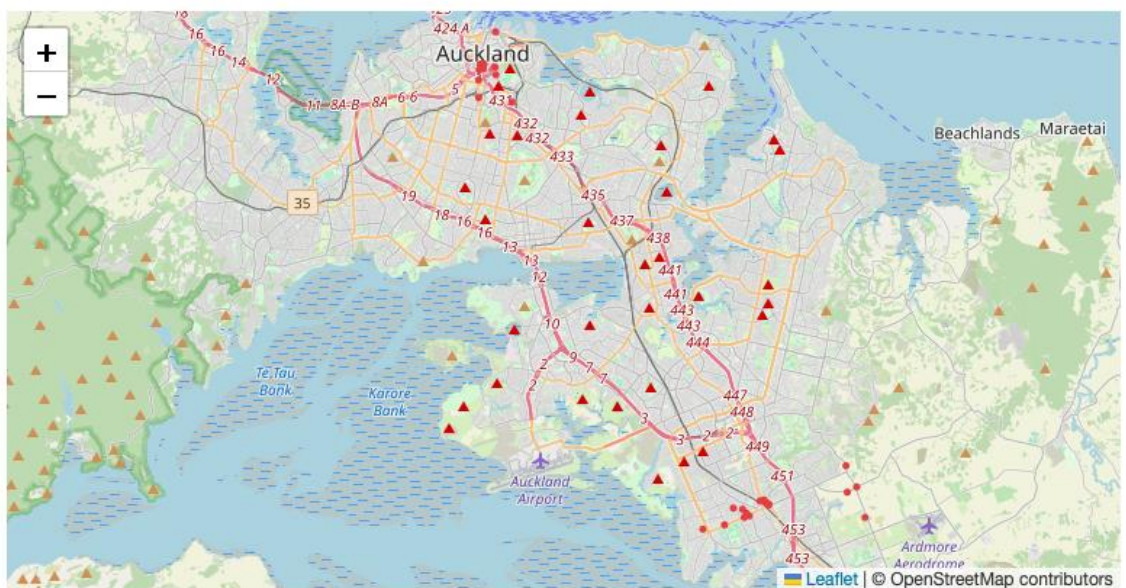


Figure 8: Heatmap showing concentration of hazards

Location-Based Analytics

Hazard Density Map



3. Dual Vetting Workflow – each hazard report goes through two layers of verification:

- Peer Confirmation – when a hazard is reported, other nearby users (or any user viewing the map) can see it marked as “Received&UnderReview” and have the option to confirm if they observe it as well. Confirming adds a vote of confidence (termed as “upvoting” in the context of this study). Likewise, users can flag a report as “not present” if they pass the location and find no such hazard. The system tallies Upvotes vs. Flags and shown in the hazard grid view.
- Moderator Verification – the system will automatically verify a report if it receives sufficient independent confirmations from the public (for example, three other users confirm it). Alternatively, an admin or moderator user (with elevated privileges) can verify or reject reports via a dashboard that lists new unverified reports. The admin can use external judgment or crosscheck, i.e. use local traffic cameras if available, to make a decision (i.e. either approve or reject the reported hazard). Once a report passes this second layer, it is marked as “Scheduled” in the system.

Figure 9: Road User Upvoting/Flagging (aka Peer Vetting)

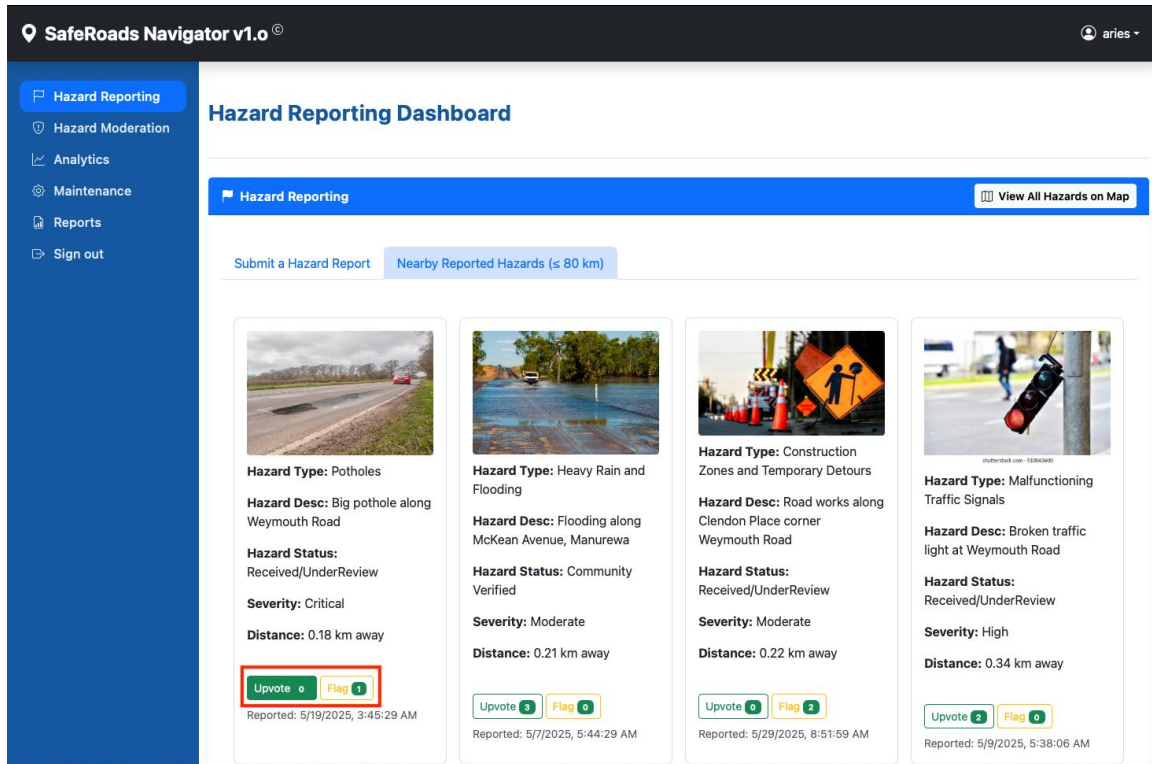
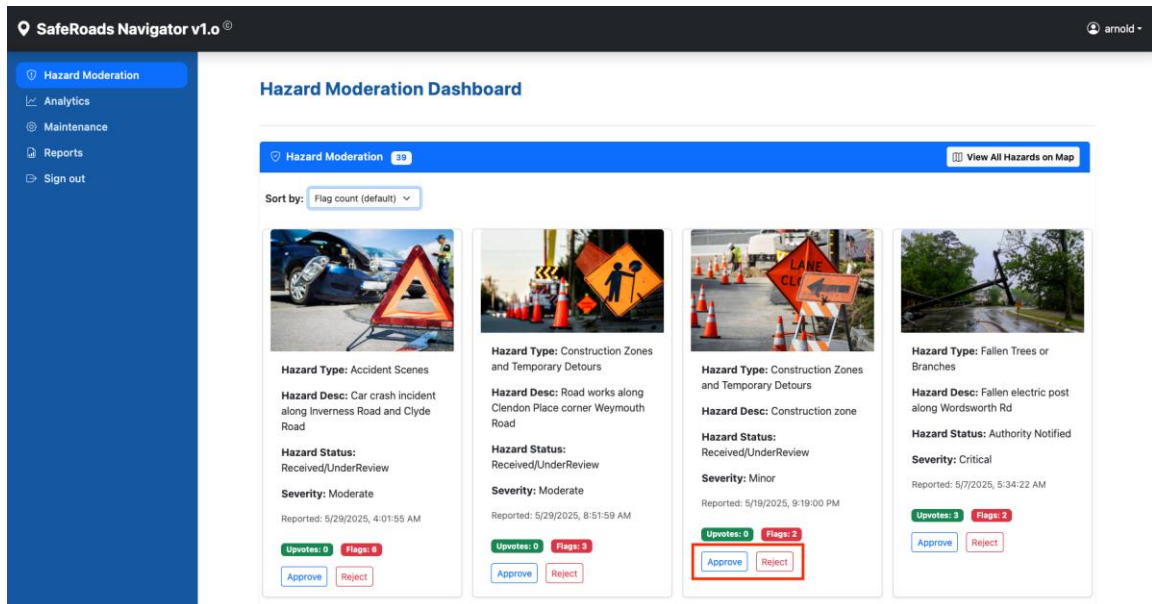


Figure 10: Moderator Verification (aka Authority Vetting)



- Analytics and Reports for Infrastructure Planning – the system provides hazard report statistics thru the Analytics dashboard. These stats

include average daily reports, average resolution time, hazard classification analytics, location-based stats, user engagement analytics, and trends / insights. The reports dashboard allows moderators to download hazard reports on a weekly/monthly basis as well as resolution rate by hazard types as CSV files.

Figure 11: Analytics Dashboard showing overview statistics

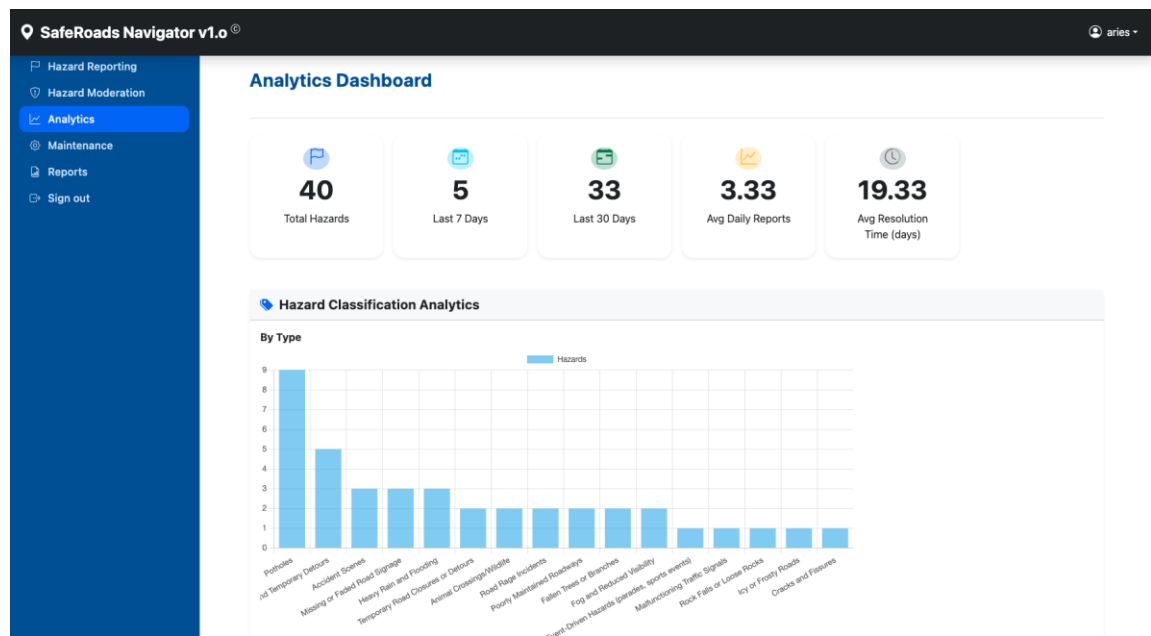


Figure 11.1: Analytics Dashboard showing hazard classification stats

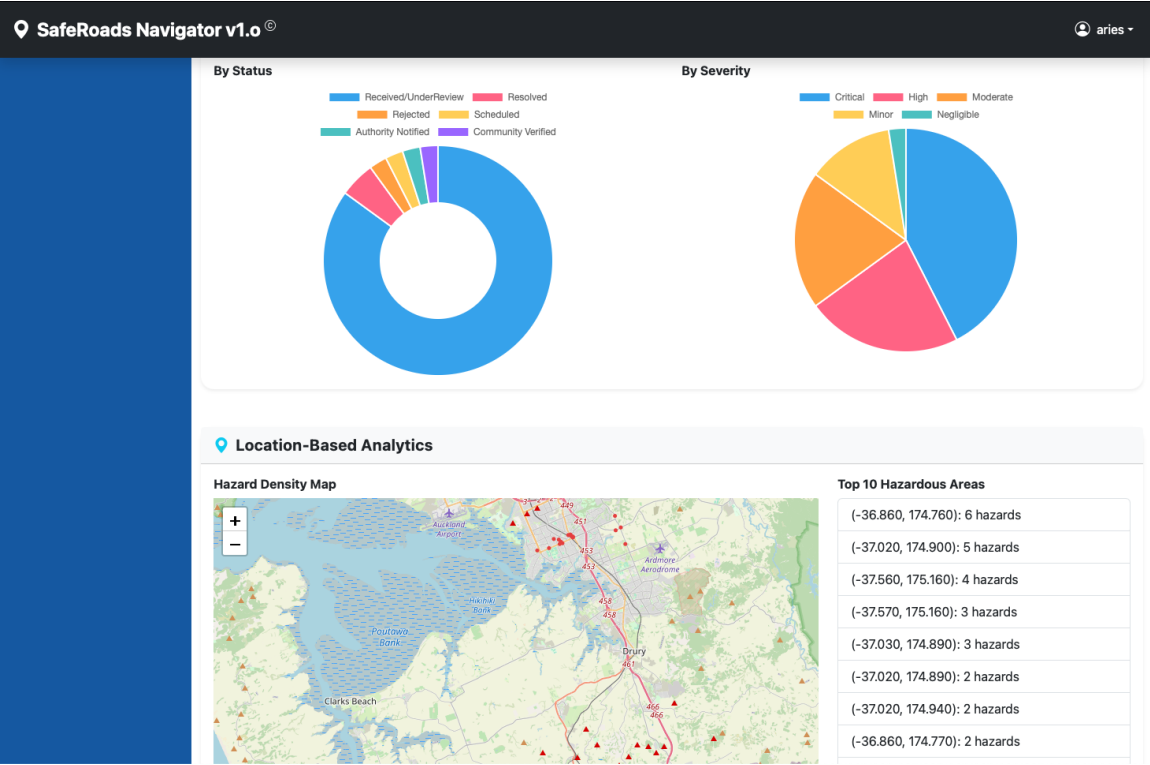
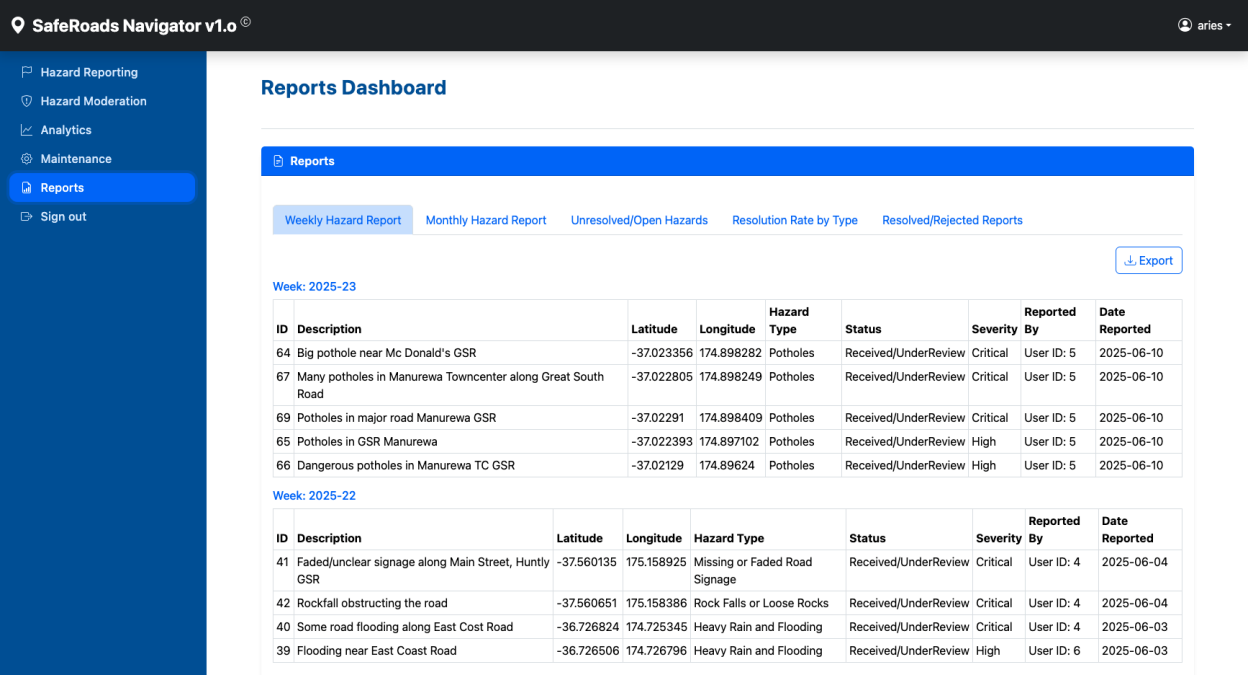
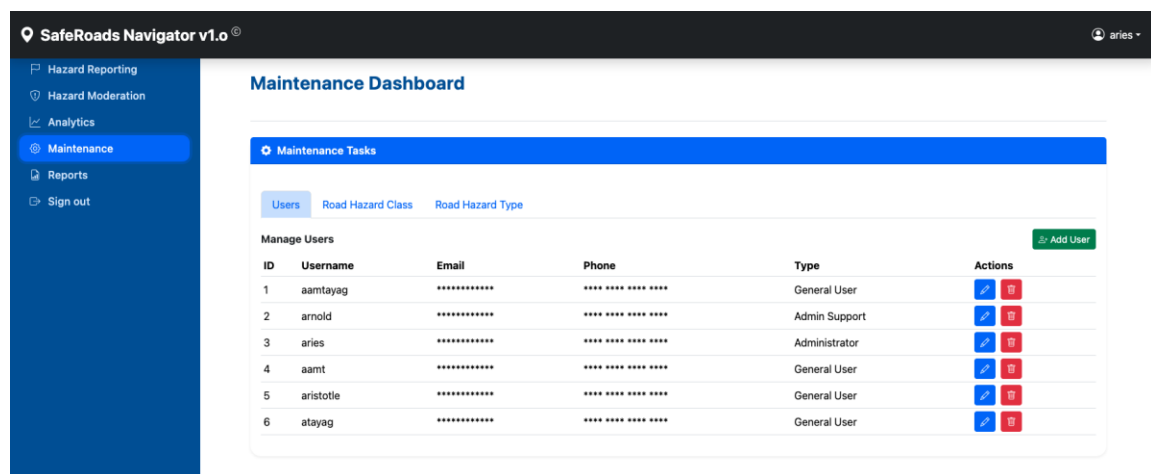


Figure 12: Access the Reports Dashboard to export hazard reports into a CSV



5. Moderation and Maintenance Dashboard – a back-end interface for an administrator or moderator to monitor hazard reports. It will list new reports with their details and any user confirmations (upvote/flags) in the hazard grid. The admin can manually mark a report as verified or reject it if it's known to be false. The dashboard also provides an overview of hazard stats and user contributions for analysis and decision making. The maintenance dashboard provides for routine maintenance tasks necessary for the system.

Figure 13: Maintenance Dashboard



In the design of the system, several features such as intuitive navigation, ease of use, and responsiveness were taken with careful consideration by making sure it displays correctly in all types of devices, i.e. smartphones, tablets, desktops, laptops. This ensures that users with varying needs can access the platform. In the context of security and legal considerations, all personally identifiable information (PIIs) stored in the system such as emails and phone numbers are encrypted using crypto, a built-in npm package for encrypting user data using AES-256-CBC algorithm. User passwords, on the other hand, are hashed using bcrypt, a javascript-based password-hashing library (see screenshot below).

Figure 14: PIIs in the system are encrypted/hashed

```
[sqlite>
[sqlite> select current_timestamp;
2025-06-29 21:37:57
[sqlite>
[sqlite> select username,email,phone,password_hash from users;
aamtayag|00c0c4cce5c35b5f0b05c6dbd3b6703:258fe40840f6257cf57fa1c6e12a3526f0e20980eaf741c857b63a0d4c785214|52a761c9456315726a0533cfab049989
:acf0cbf4f4f89fb77574438821a73c93|$2b$10$ZrR6DHA4PKUINN14LIEZwOEQ.0644zhw5cUUN2MdFxd6m9bqSzEmfa
arnold|1855288c42094b33061bc634ac52931e:1b352827e8a1730fc5988428abb8012af199e72cfba82273fb314f59a3627c6e|b75fbaea44f042d7a5ec80e413480a1a:5
798416d81be760d266e7c599752abf2|$2b$10$CIHpCnBNZIFd0V/5bj0p90mcgGHonKM0fdwVMKIRZTqeYErDrbbx6
aries|739fd54533aa5147f90a75b7507b00f:f7f9dd81d1b4373d24b6527bc7c38d1ace3355676acdbf66a743a6d92541458e|e94611d4dcd9da73a2821a6a6f450864:e0
aebb3bde94b572fbc7dd1541d9281e|$2b$10$2h183cDhe50Kcpz3Pn9avuv7pA6b6rUkHbmj7fNHG/ddgQfL4Wp9.m
aamt|73c60774f6bfa5a15c312125369dbeb3:098bc8b6829df97856b5574b76b8199f82100d949d2f03ea2d72c07a03f90a99|2bea8c660af85321029e4874af6e9fb6:906
3a6259b3f379f12df4da63d753f71|$2b$10$s.QSDwMWQto.SvIaul2jF0fuY0BRJG3nkLUPBP0m0pGJo/kK7PzGq
aristotle|d36dbbc6bfab7ef00d862f2ffc0c4767:e9c59730918da10ce1d3e3888737032bc9350db0c6f4a7ecf2053dcc3a51d89e|ab597ffa4cda4a223ce34bd37af589c
8:5acc68521a32b590faae2223fc794d1|$2b$10$/KIHu3iaLTxhId6unIrd0ebrIInqBPoGQ2AeRfzFqSfXidaYZy26
atayag|390305020936bf36694059324e37477e:ea7f9ceafa09087af766c4b4a878b43d3c1760234adbac3bf25432e30116ed44|08fa76718cd50b28bf1cd6f608f8061b:7
6e73a28698c60ca05a14e1a0db5ac|$2b$10$PFGq8ZqjtrU9j75.pf8SSOMTdgInIM2yxxFhmaAPF0x76PgB5tjcG
[sqlite>
[sqlite>
```

Figure 15. Maintenance Dashboard showing encrypted user data

SafeRoads Navigator v1.0

arnold

Hazard Moderation

Analytics

Maintenance

Reports

Sign out

Maintenance Dashboard

Maintenance Tasks

Users

Road Hazard Class

Road Hazard Type

Manage Users

ID	Username	Email	Phone	Type	Actions
1	aamtayag	*****	***-***-****	General User	Edit Delete
2	arnold	*****	***-***-****	Admin Support	Edit Delete
3	aries	*****	***-***-****	Administrator	Edit Delete
4	aamt	*****	***-***-****	General User	Edit Delete
5	aristotle	*****	***-***-****	General User	Edit Delete
6	atayag	*****	***-***-****	General User	Edit Delete

[Add User](#)

VI. Testing and Evaluation

6.1 Test Plan

The following user acceptance test (UAT) cases have been executed and verified in the development environment:

1. User Login

- wrong username – Pass
- wrong password – Pass

2. User Registration

- missing username – Pass
- missing email – Pass
- missing password – Pass
- missing phone – Pass
- duplicate username – Pass
- email and phone are encrypted – Pass

3. Road Hazard Reporting

- road hazard reporting without description – Pass
- road hazard reporting without image – Pass
- road hazard reporting without hazard type – Pass
- road hazard reporting without severity – Pass
- road hazard reported is visible in grid and map – Pass
- manual entry of place name (geocoding feature) – Pass

4. Road Hazard Upvoting

- road user cannot upvote his own report – Pass
- road user can only upvote once per report – Pass
- change RADIUS env variable and hazards report are adjusted – Pass

- road user exceed upvote limit per day (DAILY_UPVOTE_LIMIT) – Pass
- a hazard report with status=1 (Received&UnderReview), severity=5 (Critical), upvoted 3x (UPVOTE_THRESHOLD), status will change to 3 (Authority Notified) and will trigger email notification – Pass
- hazard report with status=1 (Received&UnderReview), severity!=5 (not Critical), upvoted 3x (UPVOTE_THRESHOLD), status will change to 2 (Community Verified) – Pass

5. Road Hazard Flagging

- road user cannot flag his own report – Pass
- road user can only flag once per session – Pass

6. Hazard Moderation

- all sort features are working – Pass

7. Approve road hazard (sort by hazard status)

- hazard status transition from 1/2/3 to 4 (Scheduled)
 - From 1->4 (Received/UnderReview -> Scheduled) – Pass
 - From 2->4 (Community Verified -> Scheduled) – Pass
 - From 3->4 (Authority Notified -> Scheduled) – Pass
- hazard status transition from 4 (Scheduled) to 5 (Resolved) – Pass
- road hazard is approved and record is inserted in road_hazard_moderation with comments – Pass
- road hazard is approved and record is inserted in road_hazard_moderation w/out comments – Pass

8. Reject road hazard (sort by hazard status)

- hazard status transition from 1/2/3/4 to 6 (Rejected)
 - From 1->6 (Recieved/UnderReview -> Rejected) – Pass
 - From 2->6 (Community Verified -> Rejected) – Pass
 - From 3->6 (Authority Notified -> Rejected) – Pass
 - From 4->6 (Scheduled -> Rejected) – Pass
- road hazard is rejected and record is inserted in road_hazard_moderation with comments – Pass

9. Analytics

- average resolution computation is good – Pass

10. Maintenance

- user maintenance tab load correctly and display the add user, edit, delete buttons – Pass
- add user button is working properly – Pass
- edit user button is working properly – Pass
- delete user button is working properly – Pass
- report hazard class maintenance tab load correctly and display the add user, edit, delete buttons – Pass
- add class button is working properly – Pass
- edit class button is working properly – Pass
- delete class button is working properly – Pass
- report hazard type maintenance tab load correctly and display the add user, edit, delete buttons – Pass
- add type button is working properly – Pass
- edit type button is working properly – Pass
- delete type button is working properly – Pass

11. Reports

- weekly report hazard load correctly and can export – Pass
- monthly report hazard load correctly and can export – Pass
- unresolved/open hazards load correctly and can export – Pass
- resolution rate by type load correctly and can export – Pass
- resolved/rejected reports load correctly and can export – Pass

6.2 Evaluation Metrics

The evaluation of SafeRoads Navigator will concentrate on three core dimensions:

- Accuracy and Reliability – this is to measure how well the dual-vetting mechanism filters false or malicious reports without introducing perceptible delay of road hazard reports reaching road users (Project Objective 1). Accuracy will be quantified using the following metrics:
 - Precision & recall of verified hazards vs. ground truth, i.e. field inspections
 - False Positive Rate (FPR) or the proportion of flagged hazards that are not real
 - False Negative Rate (FNR) or the proportion of real hazards missed by dual vetting
- System Integration & Performance – to assess the end-to-end platform's responsiveness, reliability, and capacity to handle real-time reporting use cases (Project Objective 2). Performance & scalability metrics can include:
 - API response times, i.e. throughput under simulated workloads
 - Throughput, i.e. maximum reports processed per second under load
 - System Uptime, i.e. percentage over evaluation period, i.e. target $\geq 99.5\%$

- Analytical Utility & Stakeholder Impact – this is to evaluate how effectively the interactive dashboards and open-data interfaces can support policy-relevant insights and decision-making (Project Objective 3). Analytical insights metrics can include:
 - Dashboard Usage, i.e. number of dashboard sessions by planners/authorities
 - Data Export Count, i.e. number of CSV/API exports performed
 - Average Resolution Time for hazard reports
 - Policy Actions Triggered, i.e. number of maintenance actions initiated via system alerts

These metrics provide a comprehensive, industry-standard view of how effectively the solution filters hazards in real time, how it performs under load, delivers measurable improvements in road-hazard awareness, gains traction among stakeholders, and drives actionable safety insights.

VII. Results and Discussion

This research addresses the key objectives, challenges, and functionalities in developing a unified & trustworthy road hazard management platform. The project's design and implementation were guided by three primary research questions, each targetting a specific research gap identified in the literature review.

1. To design a real-time trust & reputation mechanism to improve the reliability of crowdsourcing – one critical issue with crowdsourced data platforms is their susceptibility to false or malicious submissions. This

fact can greatly erode trust in the system and eventually hinder operational effectiveness. This study developed a layered trust & reputation mechanism to increase the reliability of reported road hazards without introducing significant reporting delays. Each road hazard reported is subjected to validation by peer review (from other road users) based on community upvotes & flagging occurrences. Those reports, having received a sufficient number of independent confirmations (upvotes), are escalated, while those flagged by multiple users are subject to further moderation. Also, the moderation workflow requires justification for either approval or rejection. This adds to the transparency & accountability of the decision-making process.

2. To design a unified architectural & workflow model for seamless end-to-end road hazard management – while most studies deal with one or two aspects of the entire road hazard management lifecycle, a key differentiator of this project is the adoption of the entire lifecycle in a single web-based application. General users, i.e. road users, report hazards with automatic geolocation or manual address entry, moderators can validate reports using the upvote/flagging mechanism or other external tools, and administrators analyze analytics, oversee maintenance, and reporting. This holistic design firmly addresses the second research question by providing a seamless workflow from hazard submission to resolution and long-term trend analysis, all within a single, cohesive platform.
3. To provide embedded analytics & open data interfaces for policy & collaboration – the implementation of the analytics and reporting dashboards with exportable data interfaces transforms the system into a simple but useful tool for road safety policy, decision-making, and collaboration. Through the Analytics Dashboard, road safety authorities can quickly identify issues & patterns and allocate the necessary

resources in a more efficient manner. Furthermore, the one-click export of raw and aggregated data in an industry-recognized format (CSV) empowers city planners, researchers, and maintenance teams easy and convenient access to road hazard data. This firmly addresses the third research question by ensuring that the crowdsourced data generated and dual vetted is accessible for a wide range of stakeholders.

In addition to meeting these objectives, other features that enhance the system functionality further have been implemented, such as 1) the display of a circle on the map to highlight hazards located within the vicinity of the road user in hazard mapping, 2) addition of sorting features in hazard moderation dashboard to aide admins in hazard moderation, and 3) the sending of email notification to road authorities when a hazard has been community verified, among others. The final system also complies with performance & stability requirements and demonstrates effectiveness in facilitating road hazard reporting, moderation, and real-time analytics, mapping, and updates to road users & road authorities.

Salient and notable features of the system include:

1. Hazard Reporting with Geolocation – the integration of automatic geolocation services provides ease of use to road users when reporting road hazards. The application can capture the latitude and longitude coordinates through GPS-based geolocation or through manual address-to-coordinate interpolation. Furthermore, the visual representation of road hazard locations on the interactive map significantly enhanced its usability.
2. Real-time Moderation Workflow – the hazard moderation workflow employed real-time WebSocket communication through Socket.IO. This ensures immediate updates for both users and administrators of the

system and permits system administrators to review hazard submissions more efficiently. Front-end enhancements such as dynamic hazard counts and sorting features further streamlined and simplified the moderation process.

3. Session Management & Multi-user Support – the application is able to reliably support multiple users. This was achieved by transitioning from an in-memory session store to a persistent SQLite-backed session store to eliminate session conflicts.
4. Data Analytics, Visualization, & Data Export – the analytics dashboard provides comprehensive and useful insights into the anatomy of hazards reported. The hazard mapping feature also provides a very helpful utility to inform road users of the hazards reported near their current vicinity. The CSV-exportable reports will make the application useful to third-party systems by offering data sharing capabilities in a format that is widely recognized by disparate systems.

VIII. Conclusion & Future Work

8.1 Conclusion

Road safety and maintenance management are entering a new era where community engagement and real-time data play a pivotal role. **SafeRoads Navigator** is a capstone project that proposes an innovative solution to a well-recognised problem in transportation safety: obtaining timely and reliable information on road hazards. In this study, I have proposed and designed a solution that unifies hazards reporting, response coordination, strategic analysis, and dual vetting into one platform. By harnessing the power

of crowdsourced data and addressing its weaknesses through a structured dual vetting approach, the project bridges the gap between raw community reports and trustworthy alerts that drivers and authorities can act upon. The formal literature review underscores that while crowdsourcing platforms such as Waze have revolutionised real-time traffic reporting—often detecting incidents faster than official channels, SafeRoads Navigator’s contribution is to introduce a two-layer verification mechanism that builds on prior research recommendations for trust and quality control in volunteered information and thereby increasing user confidence.

Over the course of the proposal, I’ve outlined how **SafeRoads Navigator** will work: road users can quickly report hazards via a mobile-optimised web app; those reports are visualised on an interactive map and automatically routed to the appropriate stakeholders; community feedback and admin moderation ensure the data’s integrity; and the accumulated information is analysed to guide policy and infrastructure decisions.

By achieving the project objectives, I expect to demonstrate several key outcomes: (1) a reduction in the delay between hazard occurrence and awareness, (2) an increase in citizen participation in infrastructure upkeep, and (3) enhanced knowledge for city planners about where and when hazards happen most. In simulation, the system will showcase scenarios like an emergency road obstruction being reported and broadcast to authorities in minutes, or a heatmap revealing a pattern of potholes in a particular district over a year. These use cases emphasize how proactive and data-driven approaches to road safety can save time, resources, and ultimately lives—supporting the claim that a connected community is a safer community (Olma et al, 2022).

In conclusion, **SafeRoads Navigator** aims to serve as a capstone achievement that is both academically insightful and practically impactful. The proposed system embodies the concept of “Safer Roads through Shared

Responsibility". It takes a step toward smarter cities by empowering everyday road users to be the "eyes and ears" for road safety, and by providing public agencies with a platform to act on citizen information efficiently and transparently. Should this prototype be successful, it could form the basis for a real-world pilot in partnership with the Auckland local government. By leveraging technology and community input, it strives to make travel safer for everyone. This proposal has laid out the vision, justification, and plan for realizing that vision. I am confident that, with the planned methodology and the support of the academic review panel, the project will succeed and provide a meaningful contribution to the field of Intelligent Transportation Systems in general, and a strong capstone project for the Master of Software Engineering (MSE) academic program in particular.

8.2 Future Work

Despite the good results achieved in this study, further enhancements and opportunities remain that can further enhance the usefulness of SafeRoads Navigator. Several potential enhancements can include:

1. Mobile Application Integration – implement SafeRoads Navigator as a mobile app to increase accessibility & further encourage greater community involvement and engagement.
2. Embed AI and/or Machine Learning capabilities
 - Automatic classification of road hazard type (and severity) based on uploaded image
 - Hotspot Forecasting – predict likely future hazard hotspots using time series analysis and spatial clustering

- Resolution Time Prediction - predict how long a hazard will take to resolve based on its characteristics (i.e. type, location, severity) and historical trends for the same hazard

3. Analytics Improvements

- Capture timestamps of status changes to show how quickly the authorities respond to these reported hazards
- Add anomaly/outlier detection: sudden spikes in hazard reporting by type/location
- Make the coordinates in the Top 10 Hazardous Areas clickable and would highlight in the Hazard Density Map

4. Automate the rejection of reports, i.e. set a flag count threshold

5. Promote incentivization schemes to encourage citizen involvement. The success or failure of this initiative rests greatly on the active participation of the citizenry in reporting road hazards.

IX. References

World Health Organization Global Status Report on Road Traffic Injuries. 2023, <https://www.who.int/news-room/fact-sheets/detail/road-traffic-injuries>.

NZ Transport Agency Waka Kotahi (NZTA) Road Policing Investment Programme 2024-2027, <https://www.nzta.govt.nz/assets/planning-and-investment/funding-and-investing/docs/Road-Policing-Investment-Programme-2024-27.pdf>.

New Zealand Automobile Association Media Release on Safety. 2024, <https://www.aa.co.nz/about/newsroom/media-releases/safety/2024-sees-lowest-rate-of-road-deaths-since-the-1920s>.

Christou, G., Georgiou, A., Savva, A., & Panayiotou, C. G. (2023). FixCyprus: Assessing the potential of crowdsourced data for identifying and managing road safety hazards. 15th ITS European Congress, Lisbon, Portugal, May 2023, [link](#).

Olma, J., Bode, T., Ehlers, J., Sutter, C. (2022). Road Users' Reports on Danger Spots: The Crowd as an Underestimated Expert? Safety, 8(4), 70, <https://www.mdpi.com/2313-576X/8/4/70>.

Telima, M., El Esawey, M., El-Basyouny, K., & Osama, A. (2023). The use of crowdsourcing data for analyzing pedestrian safety in urban areas. *Ain Shams Engineering Journal*, 14(6), 102140, <https://www.sciencedirect.com/science/article/pii/S2090447923000291>.

Cafiso, S., Di Graziano, A., Marchetta, V., & Pappalardo, G. (2022). Urban Road Pavements Monitoring and Assessment using Bike and E-scooter as Probe Vehicles. *Case Studies in Construction Materials*, 16, e00889, <https://www.sciencedirect.com/science/article/pii/S2214509522000213>.

Kim, J., Jeon, W., & Kim, S. (2023). Evaluating Road Hazard Maintenance Efficiency Using Citizen Science Data to Improve Road Safety. *Sustainability*, 15(20), 15074, <https://www.mdpi.com/2071-1050/15/20/15074>.

Khedher, I., Faiz, S., & Gazah, S. (2022). R-Secure: A system based on crowdsourcing platforms to improve road safety in the smart city. *Proceedings of INISTA 2022 (International Symposium on Innovations in Intelligent Systems and Applications)*, <https://ieeexplore.ieee.org/abstract/document/9894228>.

Ibtissem, K., Sami, F., S., Souhayel, G. (2022). R-Safety: a mobile crowdsourcing platform for road safety in smart cities, <https://ieeexplore.ieee.org/abstract/document/9804123>.

Bhoyar, D., Khobragade, R. H., & Khedekar, V. (2023). IoT Based Pothole Detection and Alert System*. 1st DMIHER Int. Conf. on Innovations in Engineering and Technology (July 2023),
<https://ieeexplore.ieee.org/abstract/document/10406520>.

Kindo, A. A., Kalmogo, L., & Malo, S. (2024). An Ontology-Based System for Detecting and Repairing Road Potholes Through Community Involvement. IEEE MNE3SD Conference (Nov 2024),
<https://ieeexplore.ieee.org/abstract/document/10812117>.

Desai, H., Zala, D. (2024). IoT-Based Pothole Detection System for Proactive Road Safety. Indian Journal of Technical Education Volume 47 No. 1. (Mar 2024),
https://www.researchgate.net/profile/Jhansi-Rani-M-R-2/publication/382517393_INDIAN_SOCIETY_FOR_TECHNICAL_EDUCATION/links/66a1135127b00e0ca43e2481/INDIAN-SOCIETY-FOR-TECHNICAL-EDUCATION.pdf#page=11.

Yang, C. H., Kim, J. G., Shin, S. P. (2021). Road Hazard Assessment Using Pothole and Traffic Data in South Korea. Journal of Advanced Transportation Volume 2021,
<https://onlinelibrary.wiley.com/doi/full/10.1155/2021/5901203>.

Pena-Caballero, C., Kim, D., Gonzalez, A., Castellanos, O., Cantu, A., Ho, J (2020). Real-Time Road Hazard Information System. MDPI Infrastructures 2020, <https://www.mdpi.com/2412-3811/5/9/75>

Young, S., Wang, W., Chakravarthy, B. (2019), Crowdsourced Traffic Data as an Emerging Tool to Monitor Car Crashes,

<https://pmc.ncbi.nlm.nih.gov/articles/PMC6537760>.

Leal, F., Malheiro, B., González-Vélez, H. et al. Trust-based Modelling of Multi-criteria Crowdsourced Data. Data Sci. Eng. 2, 199–209 (2017).
<https://doi.org/10.1007/s41019-017-0045-1>.