



Faculty of Computer Science and Information Technology

**REPOTHOLE- THE POTHOLE DETECTOR MOBILE APPLICATION FOR
VEHICLES USING ACCELEROMETER**

HANA SAFFIYYA BINTI AHMAD NOZAL

**Bachelor of Computer Science with Honours (Software Engineering)
Universiti Malaysia Sarawak
2025**

UNIVERSITI MALAYSIA SARAWAK

THESIS STATUS ENDORSEMENT FORM

TITLE: RePothole- The Pothole Detector Mobile Application for Vehicle Using Accelerometer

ACADEMIC SESSION: 2024/2025

HANA SAFFIYYA BINTI AHMAD NOZAL
(CAPITAL LETTERS)

hereby agree that this Thesis* shall be kept at the Centre for Academic Information Services, Universiti Malaysia Sarawak, subject to the following terms and conditions:

1. The Thesis is solely owned by Universiti Malaysia Sarawak
2. The Centre for Academic Information Services is given full rights to produce copies for educational purposes only
3. The Centre for Academic Information Services is given full rights to do digitization in order to develop local content database
4. The Centre for Academic Information Services is given full rights to produce copies of this Thesis as part of its exchange item program between Higher Learning Institutions [or for the purpose of interlibrary loan between HLI]
5. ** Please tick (√)

☐

CONFIDENTIAL

(Contains classified information bounded by the OFFICIAL SECRETS ACT 1972)

☐

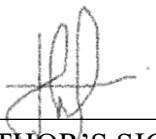
RESTRICTED

(Contains restricted information as dictated by the body or organization where the research was conducted)

☒

UNRESTRICTED

Validated by


(AUTHOR'S SIGNATURE)

(SUPERVISOR'S SIGNATURE)

Permanent Address

No 50, Kampung KEDA Kemangi,
09200,Kupang,Baling
Kedah

Date: 21/06/2025

Date: _____

DECLARATION

I therefore declare that, except for any information and materials that have been borrowed or modified from other sources, this research and all its contents are the product of my effort. According to academic and ethical guidelines, these sources have been properly credited, quoted, and acknowledged.

Signed,



.....

HANA SAFFIYYA BINTI AHMAD NOZAL

Faculty of Computer Science and Information Technology

Universiti Malaysia Sarawak.

23th June 2025

ACKNOWLEDGEMENT

Thanks to everyone's assistance and encouragement, this final year project has progressed this far. Throughout the study, I would like to sincerely thank my supervisor, Dr. Lau Sei Ping, for his support, counsel, encouragement, and patience. This project would not have progressed this far without his direction. Additionally, I want to thank Associate Professor Dr. Mohamad Nazim bin Jambli, my examiner, for his remarks and feedback on my report, which helped me improve. Additionally, I would like to thank Professor Wang Yin Chai, my coordinator for the final year project, for his guidance on the project's development. In addition, I want to express my gratitude to my friends and family for helping me finish this project. Additionally, I am grateful for the involvement of road users in the study and data collection process. Finally, I am hoping that this effort will have a positive effect on road safety.

ABSTRACT

To improve road safety and the driving experience, this project aims to develop the RePothole mobile application, a real-time pothole detection and plotting system. Potholes on roads pose serious problems, such as increased vehicle maintenance costs, safety hazards, and disruptions to daily commutes. These problems are made worse by the lack of effective and user-friendly systems to identify and address road conditions. Existing pothole management systems frequently rely on manual reporting methods, which are not only time-consuming but also neglected by road users. The RePothole mobile app uses accelerometer and GPS data from an Android device to automatically locate potholes and display them on a digital map. This provides a more streamlined and effective method by doing away with the requirement for consumers to manually report potholes. Important information on road users' experiences with potholes and preferences for such a system was obtained through a survey. The application was created with a focus on automation, simplicity, and real-time data display in response to these comments. Interactive mapping, precise location plotting, and the capability for users to choose routes that steer clear of dangerous road conditions are some of the main features. By offering a centralized platform for real-time pothole detection and visualization, RePothole empowers users with actionable information and helps authorities prioritize road maintenance efforts. This project demonstrates the potential of mobile technology in solving critical infrastructure challenges and highlights its ability to contribute to safer and more efficient transportation systems for all road users. The project is expected to result in a significant improvement in road safety, lower vehicle maintenance costs, and increased efficiency in addressing road hazards.

ABSTRAK

Untuk meningkatkan keselamatan jalan raya dan pengalaman pengguna jalan, projek ini bertujuan untuk membangunkan aplikasi mudah alih RePothole, iaitu sistem pengesanan dan pemetaan lubang jalan secara masa nyata. Lubang-lubang di jalan raya menimbulkan masalah serius seperti peningkatan kos penyelenggaraan kenderaan, bahaya keselamatan, dan gangguan kepada perjalanan kenderaan harian. Masalah ini disebabkan oleh kekurangan sistem yang efektif dan mesra pengguna untuk mengenal pasti serta menangani keadaan jalan. Sistem pengurusan lubang jalan yang sedia ada sering bergantung kepada kaedah laporan manual, yang bukan sahaja memakan masa tetapi juga jarang digunakan oleh pengguna jalan. Aplikasi RePothole menggunakan data akselerometer dan GPS daripada telefon pintar dan secara automatik mengesan lubang jalan dan memaparkannya di peta digital. Ini menunjukkan RePothole adalah kaedah yang lebih efisien dan mudah. Maklumat penting mengenai pengalaman pengguna jalan dengan lubang jalan serta keutamaan mereka terhadap sistem seperti ini telah diperoleh melalui soal selidik. Berdasarkan maklum balas ini, aplikasi tersebut direka dengan tumpuan pada automasi, kesederhanaan, dan paparan data masa nyata. Ciri utama termasuk pemetaan interaktif, pemetaan lokasi yang tepat, dan kebolehan untuk pengguna memilih laluan yang mengelakkan keadaan jalan yang berbahaya. Dengan menyediakan platform pengesanan dan visualisasi lubang jalan secara masa nyata, RePothole memberi kuasa kepada pengguna dengan maklumat yang boleh diambil tindakan dan membantu pihak berkuasa mengutamakan usaha penyelenggaraan jalan raya. Projek ini menunjukkan potensi teknologi mudah alih dalam menyelesaikan cabaran infrastruktur kritikal dan menonjolkan keupayaannya untuk menyumbang kepada sistem pengangkutan yang lebih selamat dan efisien untuk semua pengguna jalan. Projek ini dijangka menghasilkan peningkatan yang ketara dalam keselamatan jalan raya, mengurangkan kos penyelenggaraan kenderaan, dan meningkatkan keberkesanan dalam menangani bahaya jalan raya.

Contents

CHAPTER 1: INTRODUCTION	1
1.1 Introduction/Background	1
1.2 Problem Statement/Research Problem	2
1.3 Scope.....	3
1.4 Aims and Objectives	4
1.5 Brief Methodology	4
1.6 Significance of Project	6
1.7 Project Schedule.....	7
1.8 Expected Outcome	9
 CHAPTER 2: LITERATURE REVIEW	10
2.1 Overview	10
2.2 Background.....	10
2.3 Review of Existing Solutions for Pothole Detection.....	13
2.3.1 3D Laser Scanner Methods	13
2.3.2 D Image-Based Approaches	15
2.3.3 Video-Based Approaches.....	16
2.3.4 Pothole Detection Using Accelerometer and Computer Vision	17
2.4 Comparison between existing and proposed application.....	19
2.5 Analysis.....	21
2.6 Critical Review.....	22
2.7 Chapter Summary	25
 CHAPTER 3: REQUIREMENT ANALYSIS AND DESIGN	27
3.1 Introduction.....	27
3.2 Methodology	27
3.3 Analysis.....	28
3.3.1 Questionnaires.....	28
3.4 Design	40
3.4.1 Use Case Diagram.....	41
3.4.1.1 Use Case Description.....	42
3.4.2 Application Sequence Diagram.....	45
3.4.3 Activity Diagram.....	48

3.4.4 Class Diagram	50
3.5 User Interface Design.....	51
3.5.1 Mobile Application UI.....	51
3.5.2 Web Application User Interface	54
3.6 Chapter Summary.....	61
 CHAPTER 4: IMPLEMENTATION AND TESTING	 62
4.1 Introduction.....	62
4.2 Development Environment Setup.....	62
4.2.1 Frontend (Mobile App).....	62
4.2.2 Flutter Packages	62
4.2.3 File Handling and Local Storage.....	63
4.2.4 Backend and Server.....	63
4.2.5 Testing and Debugging.....	64
4.2.6 Networking and Communication	64
4.3 Software Implementation	66
4.3.1 Accelerometer Integration	66
4.3.2 Location Tracking	67
4.3.3 Data Logging	68
4.3.4 Save Pothole Information in Database	69
4.3.5 Differentiate Between Pothole and Bump	71
4.4 Chapter Summary.....	74
 CHAPTER 5: TESTING AND EVALUATION	 75
5.1 Introduction.....	75
5.2 Functional Testing.....	75
5.2.1 Test Cases	75
5.3 Non-Functional Testing.....	99
5.3.1 Pre-test and Post-test	99
5.4 Analysis.....	102
5.5 Chapter Summary.....	110
 CHAPTER 6: CONCLUSION AND FUTURE WORK	 111
6.1 Introduction.....	111

6.2 Objectives Achieved	111
6.3 Limitations and Constraints	111
6.4 Future Work.....	112
6.5 Conclusion	112
References.....	113
Appendix.....	115

List of Figures

Figure 1. 1: The Illustration of the Agile Methodology	4
Figure 2. 1: One of the potholes in Malaysia	9
Figure 2. 2: Percentage of Accident-Related Deaths in India	10
Figure 2. 3: The 3D Laser Method.....	13
Figure 2. 4:The Illustration of 2D Image-Based Approach.....	14
Figure 2. 5: Down-sampled Data from the Experiment	16
Figure 2. 6: Comparison between potholes and non-potholes down-sampled data	17
Figure 3. 1: Pie Chart-Age Range of the Responder.....	3
Figure 3. 2:Pie Chart-Driving Experience of the Responder	4
Figure 3. 3: Pie Charts- Experience with potholes on the road of the Responder.....	5
Figure 3. 4:Bar Chart- Rating of the road condition	6
Figure 3. 5:Bar Chart- Type of vehicle primarily used	7
Figure 3. 6:Pie Charts- Experience an accident with potholes.....	8
Figure 3. 7:Experience reported a pothole to local authorities.....	8
Figure 3. 8:The effect of potholes on the daily commute or travel	9
Figure 3. 9:The respondent clarification for the pothole's mobile application detector	10
Figure 3. 10:Respondent's Willingness to Share Location	11
Figure 3. 11:The Importance of Simple and User-Friendly for the Mobile Application	12
Figure 3. 12:Recommendation to Others to Use the Mobile Application	13
Figure 3. 13:Suggestions or Concerns About the Mobile Application.....	14
Figure 3. 14: Use Case Diagram for the Proposed Project.....	15
Figure 3. 15:Sequence Diagram When Pothole Detected on the Road.....	19
Figure 3. 16:View Detected Pothole on Map Sequence Diagram	20
Figure 3. 17:Navigate to Detected Pothole Sequence Diagram	20
Figure 3. 18:View Coordinates of Detected Pothole Sequence Diagram	21
Figure 3. 19:Activity Diagram of RePothole Mobile Application User and Its Interactions Before, During and After Detecting the Pothole.....	22
Figure 3. 20: The Interaction of Web User to Detect and Navigate Pothole.....	23

Figure 3.2 1: Class Diagram for the Interactions of RePothole Mobile Application	50
Figure 3.2 2: User needs to click the 'Start' button to start detecting the pothole.....	51
Figure 3.2 3: The user needs to click the 'Stop' button to stop the pothole detection on the road....	52
Figure 3.2 4: A pop-up window appears if potholes it detected when the user clicks the ‘Stop’ button.	53
Figure 3.2 5: The RePothole website map interface features a circular animated "Locate Me" button at the bottom-right, a search box with the title "Search Pothole Here" at the top-right, and the logo at the top-left.....	54
Figure 3.2 6: Filter panel slide in where the user can filter their search of the potholes based on the place, date, or status.....	55
Figure 3.2 7: By clicking on the red markers, the user can view the filtered pothole results on the map and verify specifics.	56
Figure 3.2 8: User receives a confirmation alert after checking 'mark as fixed' to verify their action before updating pothole status.	57
Figure 3.2 9: A confirmation message is sent to the user once the pothole has been successfully marked as mended.	58
 Figure 3.3 1: The user looks for potholes nearby, and the user finds their present location on the map.	 60
 Figure 4. 1: The Euclidean distance formula is used to calculate the vibrations.....	 66
Figure 4. 2: Codes of location_service.dart file.....	67
Figure 4. 3: The duplicate check by comparing new entries with the previous one.....	68
Figure 4. 4: The pothole information upload to Netlify Function in JSON Format.	68
Figure 4. 5: Status changes from UNSYNCED to SYNCED	69
Figure 4. 6: The @netlify/neon package imported to manage connection to the Neon database.	69
Figure 4. 7: Only POST requests are permitted by checking the HTTP method.	70
Figure 4. 8: The pothole informations (latitude,longitude,timestamp,status and condition_status) included in the payload.	70
Figure 4. 9: The new data for plotted pothole successfully inserted into the pothole_log.txt by the RETURNING* condition.	70
Figure 4. 10: The alert "Pothole added successfully" signed that insertion is successful.	71
Figure 4. 11:Error warning shown if there is error with the response.	71
Figure 4. 12: Accelerometer_listener.dart code which use GPS and accelerometer sensors to identify pothole.	72
Figure 4. 13: Use the detectAndLogPothole() function to record the pothole if the speed exceeds 20 km/h.	73

Figure 5. 1:Pie Chart- Familiarity of respondents to the mobile application that use sensors.	102
Figure 5. 2: Pie Chart- Level of Confidence Respondents for Mobile Phone to Detect Pothole. ..	103
Figure 5. 3: Expectation of Respondent to the Repothole Mobile Application	104
Figure 5. 4: Pie Chart- Potholes Detection When Respdent Shake the Device or Hit Over Bumps on the Road.	105
Figure 5. 5: Pie Chart- Use Website to View the Plotted Potholes	106
Figure 5. 6: Functionality of the Mobile Application that the Respondents Experienced	107
Figure 5. 7: The Accuracy of the Detection and Logging of the Potholes.	108
Figure 5. 8: Suggestions and Improvements Towards RePothole Mobile Application	109

List of Tables

Table 1. 1: The Project Schedule for the Proposed Project	7
Table 2. 1:Comparison between existing approaches and the proposed application	19
Table 3. 1: Brief Description of the Actors Involved	42
Table 3 2: The Brief Description for the Use Cases Involved.....	43
Table 4. 1: The frontend tools used and their purpose in this project.	62
Table 4. 2: The Flutter package and its usage in the RePothole project.....	63
Table 4. 3: The file handling and local storage tools and their purpose.	63
Table 4. 4: The backend and server tools and their usage.	64
Table 4. 5: The testing and debugging tools and their usage.....	64
Table 4. 6: The networking and communication tools and their usage.	65
Table 5. 1: Test Case- Pothole Detection Trigger	78
Table 5. 2: Test Case- Cooldown Mechanism.....	79
Table 5. 3: Test Case- Location Retrieval (GPS Enabled)	80
Table 5. 4: Test Case- Location Handling (GPS Disabled).....	81
Table 5. 5: Test Case- Save Log to Internal Storage (pothole_log.txt)	82
Table 5. 6: Test Case-Save Log to Server File (debug_log.txt)	83
Table 5. 7: Test Case- Automatic Upload Neon Database via Netlify	84
Table 5. 8: Update Sync Status After Upload to Neon.....	86

Table 5. 9: Test Case- Detection Notification	90
Table 5. 10: Test Case- Start and Stop Button.....	92
Table 5. 11: Test Case- View Potholes on Map	93
Table 5. 12: Marker Colour Indicator (Status Update).....	94
Table 5. 13: Test Case-Filter by Date and Place	95
Table 5. 14: Test Case-Marker Info Window	96
Table 5. 15: Test Case- Checkbox (Mark as Fixed)	97

CHAPTER 1: INTRODUCTION

1.1 Introduction/Background

Potholes and other damage to the road can pose a considerable risk to vehicles and cause major issues. These issues vary from costly maintenance when cars encounter big potholes to car crashes. In addition, poor road conditions can endanger lives, particularly if drivers fail to notice the damage in time to prevent it. Roads grow much more hazardous and require more expensive repairs the longer these problems are ignored. Atarmizi, W. A. (2024, January 20) reported in a Sinar Daily article that a nine-month pregnant woman was killed when a car encountered a pothole and collided with an electrical panel box on the Pasir Gudang Highway in Malaysia.

This tragedy serves as a reminder of the deadly accidents that might result from road damage.

Reporting road problems to authorities is now a rather inconvenient process. Typically, drivers must manually report potholes or other road problems to the local authority by calling, emailing, or using a website. According to Jabatan Kerja Raya Malaysia (JKR) (2024), road damage in Malaysia can be reported, which will be resolved within 30 days, recognized within 1 working day, and feedback given within 7 working days. However, because of this manual method, the damage might not be fixed for days or weeks, which frequently lacks real-time information. Although the mobile application Waze enables users to report barriers, the relevant authority is unaware of these reports, which provide real-time information to warn of possible risks. Because of this, road hazards put drivers in danger for far longer than necessary.

The pothole detector mobile application seeks to address these issues by automatically identifying potholes on the road in real time. When a vehicle hits a pothole, the application uses the phone's sensors to detect vibrations and GPS to record the precise location. The precise location is crucial since it guarantees that authorities are aware of the exact location of the pothole, enabling them to react effectively without collecting extra information from users. By enabling the app to notify users of dangers in their immediate area, accurate location data also improves driver safety by assisting users in avoiding potholes in real time. The mobile application allows everyone to drive safely and comfortably, as it helps road maintenance teams promptly locate and fix damaged roads.

1.2 Problem Statement/Research Problem

The safety risk that manual pothole reporting poses to drivers is one of the major issues. Drivers are more likely to use their phones or other devices while driving when they must manually report road damage, which might cause distractions. Taking one's eyes off the road for a few seconds can significantly raise the danger of accidents, even if the procedure is quick. Manually entering information, like a location or description, disrupts the driver's attention and encourages risky multitasking. This inconvenience frequently discourages quick reporting, which increases the risk of accidents brought on by hazardous road conditions because vehicles may be unable to avoid risks in time. Therefore, lowering the probability of traffic accidents and enhancing overall safety requires a safer, automated system that reduces driver engagement in real-time reporting.

Furthermore, by recording variations in speed, vibration, and orientation in response to uneven road surfaces, smartphone sensors like the accelerometer, GPS, and gyroscope provide accurate pothole detection capabilities. The accelerometer and gyroscope measure the strength and frequency of vibrations, separating potholes from small bumps or speed bumps, while the GPS locates the precise position of road damage. Since mobile phones already have the required sensors, they offer a useful, accessible, and affordable alternative to specialized equipment like Arduino sensors. Since most drivers carry mobile phones, this method does not require extra hardware, which lowers implementation costs and improves accessibility. Thus, pothole detection may be done effectively and widely without the need for specialist equipment by using a smartphone-based solution, making it practical for everyday usage.

Additionally, inaccurate location data is a serious problem with manual reporting. When driving, drivers frequently cannot pinpoint the exact position of potholes, thus they report them to authorities with inaccurate or unclear details. Road maintenance workers may find it difficult to pinpoint the precise location of the damage based on inaccurate or partial information, and this inaccuracy makes the reconstruction procedure more difficult. Even a slight variation in the stated position can cause road staff to completely overlook the pothole in complicated or heavily trafficked road networks, leaving the problem unfixed. Effective road maintenance is made more difficult by the possibility of inconsistent data resulting from depending on just the subjective opinions of various drivers. Such mistakes can be avoided by an automated system that precisely locates road damage using GPS data, guaranteeing that potholes can be found and fixed by authorities with little effort.

Lastly, potholes are frequently reported manually, which causes major delays in both identification and maintenance. Compared to automated methods, the procedure is fundamentally slower because drivers must take the time to see, record, and report road damage. Road damage may lie unreported for a lengthy period due to these reporting delays, putting other cars in danger of damage or collisions. Because reports must be examined and confirmed by authorities before any repairs can be made, the manual method also adds steps to the process. This may further increase the time it takes to resolve the problem, resulting in days or even weeks without addressing hazardous road conditions. A real-time, automated system would speed up the entire process from detection to repair, reducing vehicle risk and guaranteeing faster, more effective road maintenance.

1.3 Scope

The scope of this RePothole Mobile Application project is split into two primary user groups: local authorities and road users. Both groups gain from features specifically created to improve road safety and maintenance effectiveness.

Road User:

Supporting drivers in recognizing and navigating roads with possible pothole dangers is the main goal of RePothole. Every time a car hits a pothole, the app uses the GPS and accelerometer of the smartphone to identify it in real time. Maps automatically plot these potholes to provide a visual depiction of the state of the roads. For people who frequently travel the same routes, this function is very helpful because it enables them to take appropriate safety and smooth precautions.

Web User:

This mobile application gives online users from regular individuals to local authorities like road repair teams the ability to observe the precise locations of reported potholes in real time through the map interface. By using precise GPS data, users may quickly determine and evaluate locations that need care. This makes it easier for local authorities to plan and prioritize road maintenance projects. It provides the public with clear insight on the state of the roads in their neighborhoods. RePothole

accelerates road infrastructure management and monitoring, which eventually contributes to quicker reaction times and increased road safety.

By offering a thorough resolution to pothole-related problems and promoting safer, better-maintained roads, this dual-focus scope guarantees that the system offers significant advantages to both the public and local authorities.

1.4 Aims and Objectives

This project aims to develop a mobile application that can automatically detect the location of the potholes on the road and plot the potholes on the maps. The project's objectives are as follows:

- To use GPS data and smartphone accelerometers to automatically identify and report potholes based on road vibrations.
- To minimize user input while driving and guarantee precise GPS logging of pothole sites.
- To put reported potholes on a digital map so drivers and authorities can easily find them

1.5 Brief Methodology

Agile software development methods are incremental and iterative, with specifications that can be modified to meet the client's demands (*Sharma.S,2012*). This method is perfect for projects like the Pothole Detection Mobile Application, where user requirements and environmental elements may change as the project is being developed. Agile's focus on ongoing feedback and development guarantees that the application stays user-centered and flexible, and can produce a high-caliber, useful output that satisfies technical specifications and user expectations. Figure 1.1 below shows the illustration of the agile methodology.

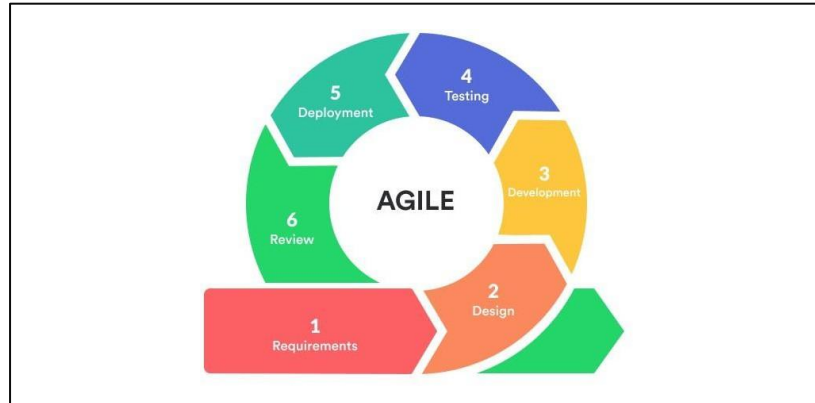


Figure 1. 1: The Illustration of the Agile Methodology

Note: from Agile Methodology by Jayathilaka, C, 2020

During a series of Agile sprint planning sessions, user needs and project objectives are gathered during the requirements phase. To explain key aspects like pothole identification utilizing GPS and accelerometers, map integration, and road hazard notifications, stakeholders like road user is approached. Using these insights, a product backlog of user stories that clearly describe feature requirements is created. As sprints move forward and new needs arise, this phase creates a flexible yet clear framework that allows the incorporation of new requirements.

The primary goal of the Design Phase is to develop a preliminary framework for the system architecture and user interface (UI) of the application. To guarantee usability and visual clarity, layouts for features like map views, pothole plotting, and pothole coordinate displays are outlined using agile storyboards and wireframes. Data storage structures for user reports and GPS logs are established by database and system architectural design. Agile promotes stakeholder engagement and iterative design reviews to ensure the design aligns with technological capabilities and user needs before development.

During the Development Phase, the application is developed in stages, with each sprint producing functional components. Core features like GPS integration and accelerometer-based pothole identification are the first to be developed. While backend development configures servers, data storage, and API endpoints, frontend development concentrates on creating an intuitive user interface. Cross-functional project alignment is ensured via regular daily stand-ups, and stakeholders can test and offer feedback during sprint reviews after each iteration. Thanks to this iterative process, every feature is tested and improved upon

before proceeding.

In Agile, testing is a continuous process that starts with each sprint to guarantee functionality and quality. For essential features like precise GPS logging and responsive map interactions, the testing phase consists of unit, integration, and user acceptability tests. Additionally, usability testing confirms that users may easily access and utilize reporting tools. Early problem and issue detection through continuous testing enables prompt implementation of fixes. Agile's emphasis on testing at every sprint enhances dependability and gets every feature ready for release.

The software is released in phases during the deployment phase, beginning with a beta version for limited user testing in the real world. Early customer feedback at this stage is released and changes are based on real usage. Backend systems are tuned to manage real-time data processing, and tutorials and documentation are developed to assist administrators and users. Agile's adaptability guarantees a seamless release by allowing for any last-minute adjustments to be made before the full-scale deployment.

The main objectives of the review and feedback phase are continuous improvement and postdeployment analysis. User input, usage analytics, and performance monitoring guide future sprints for updates, bug corrections, and the creation of new features. Regular reviews provide an opportunity to examine what went well and identify areas for improvement, which will help the development process improve over time. Thanks to Agile's iterative approach, which promotes continuous improvement, the mobile application may adjust and change in response to changing user needs.

1.6 Significance of Project

By improving road safety, improving road repair procedures, and providing users with increased convenience, this pothole detector mobile application project offers substantial advantages. The mobile application lowers the likelihood of accidents and fosters a safer driving environment by enabling drivers to avoid hazardous road conditions through automated pothole plotting and real time pothole identification. Maintenance workers can respond and fix road problems more quickly because of the app's ability to enable instantaneous, correct reporting to authorities, which also reduces the delays that come with manual reporting. By prioritizing repairs for high-risk regions, this quick-response

mechanism powered by precise GPS data allows authorities to more efficiently distribute resources, reducing operating costs.

Plus, the app benefits the environment and the economy and enhances road conditions. It promotes environmental sustainability by reducing pollution and fuel consumption brought on by interruptions and bad road conditions. Additionally, drivers save money on repairs since there are fewer cases of vehicles being damaged on dangerous roads. The design of the application puts the ease of the user first, enabling drivers to improve road quality through human reporting or automated detection easily. RePothole is a scalable, significant solution that improves infrastructure quality and road safety for the benefit of both maintenance authorities and drivers.

1.7 Project Schedule

Table 1.1 below shows the project schedule for this proposed project for FYP1.

Table 1. 1: The Project Schedule for the Proposed Project

Task	Start Date	End Date	Duration (day(s))
Brief Proposal	14/10/2024	20/10/2024	7
Full FYP Proposal	20/10/2024	13/11/2024	25
Chapter 1: Introduction	13/11/2024	21/11/2024	9
Chapter 2: Literature Review	21/11/2024	12/12/2024	22
Chapter 3: Requirement Analysis and Design	12/12/2024	09/01/2025	29
Phase 1.0(Requirement Gathering): Distribute online survey via Google form	22/12/2024	24/12/2024	3
Phase 2.0 (System Design): Prototype and Wireframe	22/12/2024	26/12/2024	5
FYP1 Report Submission	09/01/2025	17/01/2025	9
FYP1 Report Amendment	17/01/2025	19/01/2025	3

FYP1 Final Report Submission with Amendment	19/01/2025	25/01/2025	7
Submission of the proposed structure of FYP report, title and gantt chart.	27/03/2025	01/04/2025	5
Chapter 4 : Implementation and Testing	01/05/2025	15/05/2025	14
Submission of first draft for chapter 4	01/05/2025	15/05/2025	14
Chapter 5: Testing and Evaluation	16/05/2025	19/06/2025	34
Chapter 6: Conclusion and Future Work	20/06/2025	23/06/2025	3
Submission of first draft for Chapter 5,6 & Abstract for paper.	16/05/2025	31/05/2025	15
Submission of first draft for FYP2 full report & paper	01/06/2025	10/06/2025	9
Submission of Final Report	11/06/2025	23/06/2025	12
Symposium FYP Project	30/06/2025	01/07/2025	1

1.8 Expected Outcome

By lowering the chance of crashes and minimizing vehicle damage, this mobile application greatly improves driver and passenger safety by making it easier to quickly identify road hazards. This project aims to increase the effectiveness of road maintenance operations by enabling local authorities to promptly resolve these problems by utilizing real-time data. In the end, this prompt action contributes to safer and more dependable infrastructure by enabling repairs to be prioritized according to the location and severity of the road defects. Furthermore, this process is made much simpler by the application's user-friendly mapping feature. It offers a clear visual depiction of pothole sites, making it simple for drivers and law enforcement to identify places that need care. Additionally, the incorporation of user-friendly technology guarantees that all parties involved, including the public and local authorities, can interact with mobile applications without difficulty and help to maintain a safer and better road network.

CHAPTER 2: LITERATURE REVIEW

2.1 Overview

This chapter will review three related studies from earlier research and the tools that will be used to advance this project.

2.2 Background

A pothole can be determined as the shape of a small hole or depression through the carbon lining, occasionally resembling an approximately inverted cone (Fui Siew et al., 2005). According to popular justification, the holes were progressively worn deeper and deeper into the bedrock by the abrasion of the bedrock caused by rotating currents inside the potholes and the shaking of pebbles and stones by the water (E. Cox, 1975). Figure 2.1 below shows the image of potholes on one of the roads in Malaysia.



Figure 2. 1: One of the potholes in Malaysia

(Adapted from Khairulrijal, 2024)

Road users are really at risk from potholes, which can range in size from tiny depressions to massive craters. Their ubiquity is a worldwide problem, especially in regions with high rainfall, inadequate drainage, or sharp temperature changes. Potholes have a major influence on road safety, car maintenance expenses, and the general quality of the infrastructure, which makes them crucial to fix. Potholes are a major source of accidents for drivers, particularly for cyclists and motorcyclists, where even minor flaws can cause serious harm or even death. According to Kandoi et al. (2021), the Ministry of Statistics and Program Implementation in India reports

that traffic accidents account for the bulk of unintentional fatalities in the country as shown in Figure 2.2. India has two million kilometers of roads, of which one million kilometers are badly built, according to the study. According to the data set acquired by the Ministry of Road Transport and Highways, potholes are the cause of almost 50,000 accidents annually.

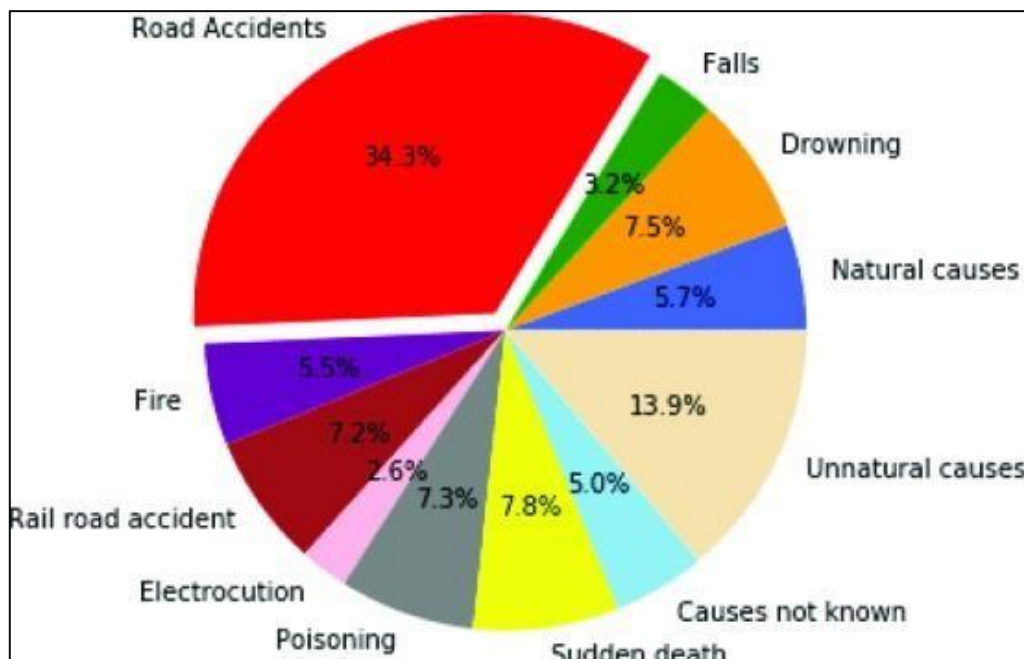


Figure 2. 2: Percentage of Accident-Related Deaths in India

(Adapted from Kandoi et al., 2021)

Additionally, they contribute to vehicle damage, including suspension failures, alignment problems, and tire blowouts, which result in expensive repairs for drivers. Furthermore, potholes and poor road maintenance in cities can cause traffic jams, decrease productivity, and result in financial losses by delaying the delivery of products and services. Data shows that many crashes involving lone motorcycle riders are caused by abrupt direction changes and a high likelihood of trying to avoid potholes, according to Ahmad Farhan Mohd Sadullah, a professor of transportation at University Sains Malaysia in one of the Article from Free Malaysia Today (2021). Ignoring pothole repairs might worsen road damage from an infrastructure viewpoint. Potholes tend to enlarge with time, particularly in places with heavy traffic or inclement weather. Local finances may be strained because of the increased complexity and expense of repairs. Additionally, potholes degrade the roadways' looks and use, which affects public happiness and confidence in local government.

Based on the journal by Masse, L. C., et.al (2005), accelerometers are acknowledged as a reliable and impartial instrument for evaluating unstructured physical activity. In other definitions, it is determined as a device that monitors acceleration forces whether brought on by motion, gravity, or vibration is called an accelerometer. Accelerometers, which are used in contemporary smartphones, measure variations in velocity along three axes: Z (depth), Y (vertical), and X (horizontal). The accelerometer is essential for detecting potholes because it records vibrations brought on by abnormalities in the road. Spikes are the abrupt variations in vertical acceleration (Z-axis) that occur when a car passes over a pothole. The application can distinguish pothole impacts from normal road vibrations or speed bumps by examining these patterns. The accelerometer's ability to identify potholes is mostly dependent on its sensitivity and accuracy. Accuracy measures how well the sensor's results match the real values, whereas sensitivity describes the smallest change in acceleration the sensor can detect. Most smartphone accelerometers have an accuracy range of ± 0.01 g to ± 0.02 g and can detect variations as little as 1 milli-g (0.0098 m/s^2). They are appropriate for identifying pothole-induced road imperfections due to these specifications. However, due to variations in sensor quality among smartphone models, certain gadgets might be more adept at picking up on minute variations in road conditions than others. Another crucial element in pothole identification is the accelerometer's sampling frequency, which controls how many data points it records in a second. A frequency of 50 Hz to 100 Hz is usually adequate for precise pothole identification. This range records the abrupt impacts brought on by potholes without flooding the system with extraneous information that can deplete the battery. Based on the studies by Kandoi et al., (2021), the system's accelerometer sensor's sampling rate is set to 50 Hz for better accuracy. While a very high sample rate could result in ineffective processing, a lower rate might miss these quick changes.

Accelerometer readings can be used to identify potholes by examining the vibrations and abrupt acceleration changes that occur when a car hits one. A smartphone's accelerometer measures force along three axes: depth (Z), vertical (Y), and horizontal (X). A rapid, noticeable spike in the vertical (Z-axis) acceleration occurs when a car passes over a pothole because of the sudden drop and subsequent increase in the road surface. The smoother patterns produced by frequent driving or little road disturbances are very different from this spike. To find these distinct patterns, the accelerometer data must be regularly monitored. A pothole impact, for instance, would register as a significant rise in vibration amplitude on the Z-axis, often accompanied by less noticeable changes on the X and Y axes due to lateral and forward motion,

as the smartphone records acceleration changes. The algorithm can differentiate potholes from other kinds of road conditions, including speed bumps or uneven patches, by examining the size and frequency of these spikes. This combination of vibration pattern recognition and accelerometer data offers a portable and affordable pothole-detecting method. The technology can consistently detect and record potholes by utilizing the sensitivity and precision of the accelerometer in conjunction with sophisticated data processing, which helps to improve road maintenance planning and make driving safer.

2.3 Review of Existing Solutions for Pothole Detection

2.3.1 3D Laser Scanner Methods

In the research by Chang et al. (2005), a grid-based processing technique was used to obtain the study's precise 3D point-cloud points and elevations during scanning and retrieve them, emphasizing certain distress aspects. A "point cloud," or collection of data points in a three-dimensional coordinate system that describes the surface, is produced by the scanner. To detect differences brought on by road distress, this point cloud has accurate elevation data. The researchers used a grid-based processing technique to speed up the analysis. Segmenting the scanned area into smaller parts enabled the system to measure the depth, size, and coverage of characteristics like potholes.

A "point cloud" that depicts the road surface in three dimensions is created when laser beams from the scanner strike it and bounce back. The height of every point in this point cloud is one of the precise details that are essential for identifying surface imperfections. The system uses a grid-based methodology to process the data after it has been gathered. To enable targeted examination of regions, the scanned road surface is divided into smaller parts called grids. To find slopes or depressions that point to potholes, the algorithm examines elevation variations inside each grid. It computes their depth, size, and coverage to decide the severity of these traits. Prioritizing which potholes require immediate repair is made easier by the severity analysis. The technology also calculates how much material will be needed to patch the potholes, which offers useful information for effectively allocating maintenance resources. The findings are then combined into a report, including the locations, sizes, and degrees of severity of the found potholes (Chang et al.,2005).

Other studies, by She et al. (2020), investigate the possibilities of measuring and analyzing pothole characteristics on asphalt pavements using 3D line laser technology. The study focuses

on determining if this technique is feasible, accurate, and efficient for analyzing road distress features and recording precise surface properties. The illustration is shown in Figure 2.3 below.

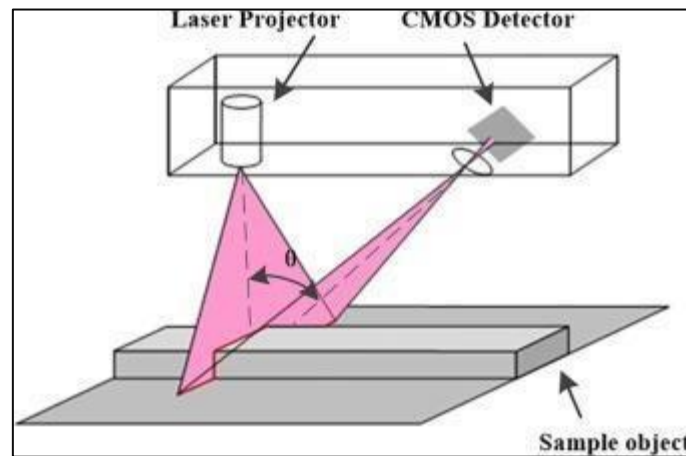


Figure 2. 3: The 3D Laser Method

(Adapted from She et al., 2020)

A 3D line laser scanning system that can record high-resolution information on the road surface was put into place by the authors. To capture exact surface elevations and produce a three-dimensional profile, this technology continuously projects a laser line across the asphalt pavement. To find potholes and quantify important metrics like depth, area, and volume, the gathered data was examined. When assessing the extent of the damage and organizing repair efforts, these metrics are essential.

The triangulation principle, shown in Figure 4, is the foundation of the 3D line laser system. A digital area scan camera with a charge-coupled device (CCD) or complementary metal oxide semiconductor (CMOS) sensor and a laser projector are the standard components of a 3D line laser system. With the projector, the camera is positioned at an oblique angle and a known distance (She et al., 2020).

According to the study, 3D line laser technology offers excellent accuracy in identifying potholes and evaluating their characteristics. Compared to conventional inspection methods, which are frequently less accurate and require more time, this approach offers substantial advantages by producing detailed digital models of the road surface.

2.3.2 D Image-Based Approaches

The use of 2D image-based techniques for identifying and evaluating road surface distress, such as potholes, was examined by Koch and Brilakis (2012). To detect and quantify potholes and other defects, these methods use traditional cameras to take pictures of the road's surface, which are subsequently processed using computer vision algorithms. Figure 2.4 below shows the illustration of 2D image-based approach.

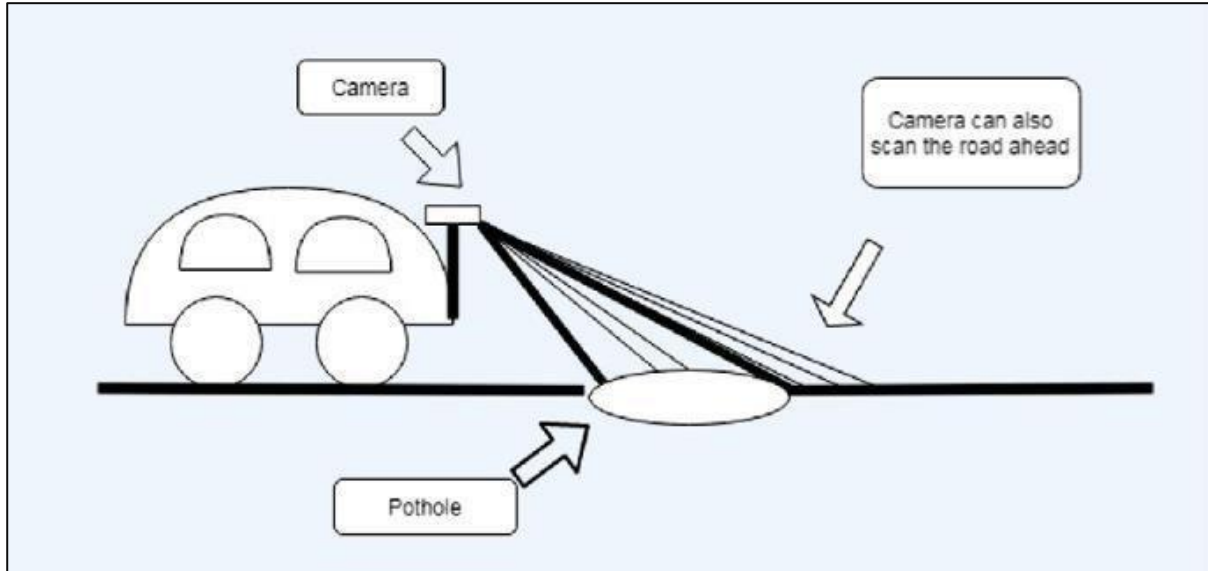


Figure 2. 4: The Illustration of 2D Image-Based Approach

(Adapted from Acharjee et al., 2020)

To find features like potholes, depressions, and cracks, the technique entails taking high-resolution 2D pictures of the road and using image-processing algorithms. Damaged areas are frequently highlighted using methods including segmentation, texture analysis, and edge detection. The system can categorize and assess the degree of road faults by comparing the discovered features to preset standards or patterns.

When compared to more sophisticated 3D technologies like laser scanning, the main benefit of 2D image-based techniques is their affordability. Large-scale road maintenance initiatives can use this method because cameras are generally accessible and simple to install. Furthermore, the processing can be completed quickly, making it possible to analyze large road networks effectively (Koch et al., 2012).

2.3.3 Video-Based Approaches

Research by Jog et al. (2012), presented a technique for measuring and detecting potholes that combines 2D recognition with 3D reconstruction. This method records road movies using a camera installed on an automobile. The 2D recognition portion relies on earlier research by Brilakis and Koch that detects potholes in individual frames. Pothole width, depth, and number are measured using comprehensive 3D images produced by the 3D reconstruction process, drawing inspiration from surface modeling and point cloud analysis. Potholes were successfully found and measured when this technique was tested at 20 mph using a Canon camera. By enabling real-time detection and minimizing the number of frames needed for reconstruction, the researchers hope to enhance the system further.

Another video-based method for identifying and evaluating potholes, fractures, and patches on roadways was created by Lokeshwor et al. (2013). This technique uses two specialized algorithms to analyze video snippets. First, frames with and without obvious road damage are separated by the DFS (Distress Frame Selection) algorithm. The damaged frames are then processed by the CDDMC (Critical Distress Detection, Measurement, and Classification) algorithm, which improves the pictures, divides them into segments, extracts visual characteristics like size, shape, and texture, and categorizes the damages. With this method, potholes (97%), cracks (94%), and patches (90%), were detected with excellent accuracy. Its accuracy was occasionally reduced, though, when things like manholes or black markers mimicked road damage (Kim & Ryu, 2014).

2.3.4 Pothole Detection Using Accelerometer and Computer Vision

Aditi Kandoi et al.'s (2021) study "Pothole Detection using Accelerometer and Computer Vision with Automated Complaint Redressal" describes a hybrid approach that efficiently detects potholes using computer vision and accelerometer data. By integrating these technologies, the system improves accuracy and offers useful visual confirmation in addition to detecting possible potholes. This dual strategy addresses the drawbacks of depending just on one technology. Accelerometer data is used in the first step of the procedure and is usually gathered via sensors mounted on devices or embedded in cell phones. With an emphasis on the Z-axis, which measures vertical motion, the accelerometer tracks vibrations in the X, Y, and Z axes. A pothole causes a car to dip suddenly and then rise again, which causes a sharp spike in the Z-axis data. By examining the accelerometer's output, these spikes can be recognized because they are characteristic. Potential potholes are flagged by this data, which acts as the initial detection technique. The dataset for this study is shown in Figure 2.5 and Figure 2.6 below.

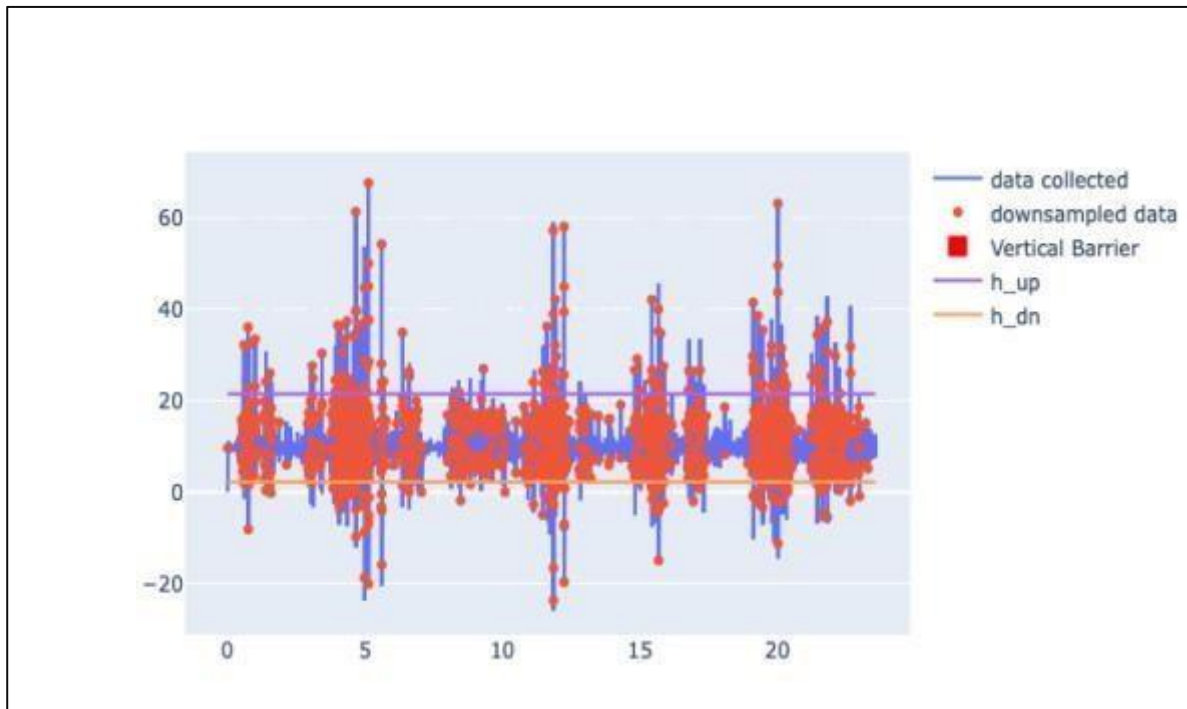


Figure 2. 5: Down-sampled Data from the Experiment

(Adapted from Adapted from Kandoi et al., 2021)

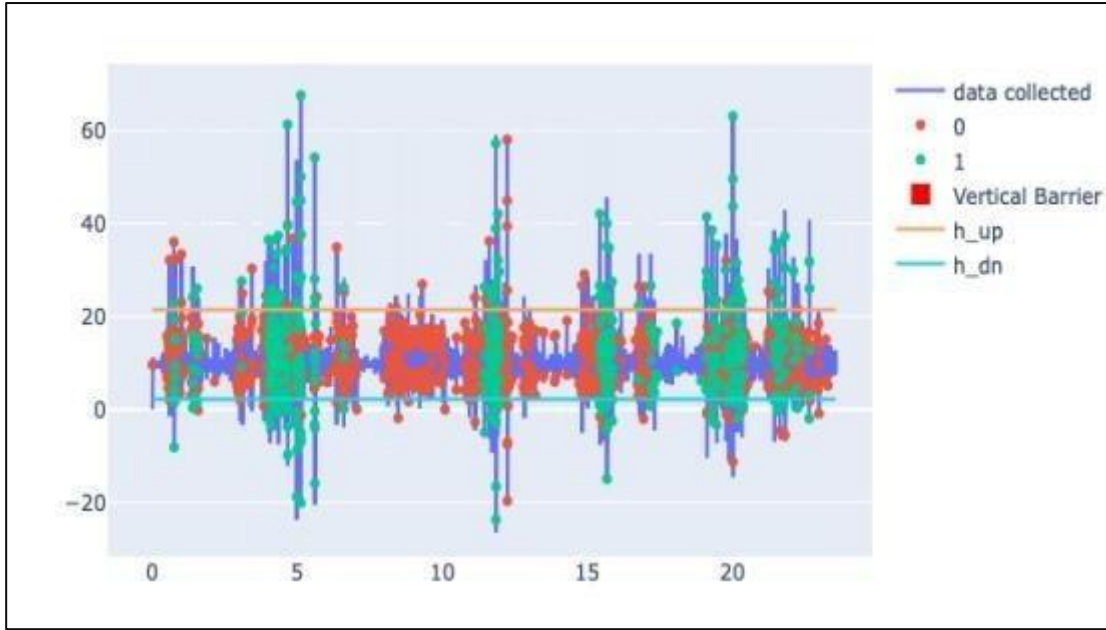


Figure 2. 6: Comparison between potholes and non-potholes down-sampled data

(Adapted from Kandoi et al., 2021)

The system incorporates computer vision techniques to decrease false positives and improve the detection process's reliability. Continuous road footage is recorded by a camera installed on the car. The system concurrently examines the associated video frames or photos when the accelerometer detects a possible pothole event. These frames are processed by computer vision algorithms, which visually verify the existence of a pothole, evaluate its severity, and exclude other potential causes of accelerometer spikes, such as speed bumps or other irregularities in the road. This integrated method yields insights that are both quantitative and qualitative. Realtime vibration-based detection is provided by the accelerometer, while classification and validation are added by computer vision. When combined, they produce a strong system that can accurately detect potholes. The system also has an automatic complaint redressal procedure that uses GPS to record the position of the identified pothole and forwards the information to the appropriate authorities for fixing. The technique is very successful in increasing road safety and maintenance efficiency because it combines accelerometer data, computer vision, and automated reporting.

2.4 Comparison between existing and proposed application

Based on the existing method that has been reviewed, the comparison is shown in Table 2.1 below.

Table 2. 1: Comparison between existing approaches and the proposed application

Method /Approaches	3D Laser Method	2D Image/Video-Based Approaches	Pothole Detection Using an Accelerometer and Computer Vision	Proposed Application (RePothole)
Cost	High: Demands costly devices and equipment for laser scanning.	Moderate: Needs a high-end camera and processing power for photos and videos.	Low: Makes use of widely accessible smartphone sensors.	Low: Makes use of widely accessible smartphone sensors.
Accuracy	Very high: Uses 3D data to provide accurate measurements of pothole dimensions (width, depth, and volume).	Moderate to High: Accuracy is affected by environmental conditions, lighting, and video quality; it has trouble measuring depth.	High: Computer vision and precise vibration analysis.	High: Accurate accelerometer-based vibration analysis for road hazards, together with real-time detection and GPS tracking.

Mobility	Low: The equipment needs to be mounted on a vehicle and is heavy.	Moderate: Depends on extra camera and video processing equipment but is portable.	High: Easily mobile, fully integrated into cell phones.	High: Completely portable and built into cell phones, which people already carry about a lot.
Usability	Low: Needs skilled workers to operate and decipher the findings of 3D laser scanning.	Moderate: Simpler to use but requires specialist software or knowledge of video processing.	High: Easy-to-use app UI; little technological expertise needed.	High: Easy-to-use interface that requires little technical expertise for drivers.
Real-Time Use	Limited: Frequently prioritizes local precision above	Moderate: Video-based techniques can detect potholes in real time;	High: GPS and vibration analysisbased real-time alarms and reporting.	High: Using GPS tracking and vibration patterns to provide real-time alerts and reporting
	real-time detection.	however, their high computing demands may cause delays.		
Data Output	Potholes in 3D, with dimensions and volume included.	Videos or still photos with comments showing the sites and the	Uses vibration analysis to display pothole locations, GPS positions, and severity.	Pothole locations, GPS coordinates, and severity based on vibration data

		extent of damage.		are displayed in map-based notifications.
--	--	-------------------	--	---

2.5 Analysis

Based on the existing related approaches and the proposed application, each one of them has its significance for the pothole detection solution. Out of the three, the 3D laser approach is the costliest. It is less practical for general use since it depends on specialist machinery, such as laser scanners, which have high upfront and ongoing expenditures. On the other hand, 2D image and video-based methods simply need cameras and processing hardware, making them a reasonably affordable substitute. In contrast, both the accelerometer-based detection and the proposed RePothole application employ smartphone sensors, which are widely available and affordable. The suggested application is the best option due to its cost-effectiveness, particularly for wider deployment.

Pothole size, depth, and volume may all be precisely recorded using the 3D laser approach. It may be too much for wide detection, but it is perfect for precise road repairs. The accuracy of the 2D image and video-based methods varies with camera quality and environmental factors, but they perform rather well overall. By examining vibration patterns, RePothole and the accelerometer-based approach provide accurate detection. Furthermore, the suggested application improves accuracy by integrating GPS for accurate location mapping, which makes it ideal for real-time use on a variety of road conditions.

One of the main drawbacks of the 3D laser technique is portability. Deploying heavy equipment across several places is difficult since it needs to be installed on cars. Although the 2D image/video-based methods are more portable, their applicability is limited by extra hardware such as cameras and processing computers. Because they make use of smartphone devices, consumers already carry the accelerometer-based approach, and RePothole thrives in mobility. This intrinsic mobility guarantees broad usage across numerous locations and ease of implementation.

One important component of broad acceptance is ease of use. The 3D laser method's

applicability for non-specialist users is limited because it requires highly qualified professionals to operate and understand the data. Like this, 2D image and video-based methods are less user-friendly since they necessitate knowledge of video processing software. However, drivers with little technological expertise can utilize the accelerometer-based detection and RePothole program because of their straightforward interfaces. Because of its emphasis on usability, RePothole is a more approachable and user-friendly solution.

For prompt hazard reporting and prevention, real-time detection capabilities are crucial. This is one of the limitations of the 3D laser approach, which prioritizes accuracy over real-time functionality. Although 2D video-based techniques provide real-time detection, their high computational requirements may cause delays. RePothole and the accelerometer-based approach are notable for their capacity to deliver real-time notifications and alarms through GPS tracking and vibration data. This improves road safety and guarantees prompt reporting.

The kind of data output that is produced is essential for efficient analysis and reaction. Although the 3D laser approach yields detailed 3D measurements, real-time applications frequently do not require this degree of information. Although 2D image and video-based methods provide annotated images or videos, they are devoid of depth information and might necessitate further processing. RePothole and the accelerometer-based approach offer useful, actionable data in the form of severity ratings and GPS positions. This is further enhanced by the suggested application, which provides map-based notifications, making it extremely useful for both authorities and end users.

In conclusion, the suggested RePothole mobile application achieves the optimum balance between cost, accuracy, mobility, usability, and real-time functionality, even though each approach has advantages. It is the best option for contemporary pothole identification and reporting since it relies on widely accessible smartphone technology and has an easy-to-use interface.

2.6 Critical Review

The Pothole Detection Mobile Application, which uses GPS and an accelerometer, is a creative and useful solution to the widespread problem of pothole-related road safety. Because it makes use of widely accessible smartphone sensors, this project is both affordable and widely available. The technology provides a real-time solution for pothole detection and reporting by using the

GPS to log accurate locations and the accelerometer to detect road abnormalities. But like any new technology, it has advantages and disadvantages that should be carefully considered.

This project's cost and scalability are two of its biggest benefits. This strategy makes use of parts found in most contemporary cell phones, as opposed to approaches that depend on pricey technology like 3D laser scanners or specialized image processing systems. This lowers the barrier of entry for users and promotes broad adoption. Additionally, it maximizes its reach by being cross-platform and compatible with both iOS and Android devices thanks to frameworks like Flutter. The application's scalability makes it a great choice for widespread deployment in both urban and rural areas, where road repair is frequently erratic or inadequately funded.

The project uses accelerometer data to detect vertical vibrations generated by potholes, balancing efficiency and simplicity in terms of accuracy. Predetermined thresholds and signal processing techniques aid in differentiating between small road flaws and serious risks. However, because user behavior and road conditions vary, the system might have trouble maintaining consistently high accuracy. Variations in car models, speeds, and smartphone locations, for example, can bring noise into accelerometer measurements, which could result in missed detections or false positives. To accommodate a variety of settings, this calls for reliable calibration and flexible algorithms, which could make the development process more difficult.

Additionally, the project is very user-friendly and portable. Using cell phones makes it possible for regular drivers to use them without the need for specialized equipment. The application is simple to install, and its user interface can be made to offer easy feedback, including mapping pothole locations and real-time notifications. However, there are drawbacks to this dependence on cell phones as well, since the app's functionality depends on the battery life and sensor quality of the device. The continual use of GPS and sensors can quickly drain the device's battery, which can affect usage over extended periods. Additionally, older or less expensive smartphones may have fewer sensitive accelerometers, which could reduce detection accuracy.

While this effort does not completely replace current technology, it does solve many of its limitations. For instance, 3D laser scanning provides unmatched accuracy in determining the size and depth of potholes, but it is too costly and impractical for general use. Like this, 2D and video-based methods offer in-depth visual insights, but they are constrained by their high computing costs and dependence on external cameras. Although less thorough, the

accelerometer plus GPS method is excellent in terms of cost, simplicity of use, and real-time capabilities, making it a complete solution for daily use.

To sum up, this Pothole Detection Mobile Application is a promising invention that satisfies the current demand for portable and reasonably priced road safety solutions. Although it tackles many of the issues raised by current approaches, its success will rely on its capacity to adjust to a variety of environments, retain high accuracy, and deliver reliable performance on a range of devices. This project could have a big impact on road safety and maintenance initiatives globally if it keeps improving its algorithms and investigates supplementary technologies like cloud-based data sharing or machine learning for better classification.

2.7 Chapter Summary

An inventive approach to the pressing problems of road safety and infrastructure upkeep is the Pothole Detection Mobile Application. Potholes are a major problem on a global scale since they compromise the safety of drivers, damage cars, and require expensive repairs. This research provides an economical, portable, and effective way to identify potholes in real time by leveraging the accelerometer and GPS capabilities of smartphones. Road dangers can be precisely identified and reported thanks to the combination of accelerometer data, vibration analysis, and GPS location monitoring, which enables authorities and drivers to take preventative action.

It is impossible to exaggerate how crucial it is to include case studies and examples in the literature evaluation. The usefulness of accelerometer-based techniques and the viability of using machine learning algorithms for road condition classification are both illustrated by these examples, such as the research conducted by Kandoi et al. (2021). They demonstrate how integrating theoretical methods with empirical data improves our comprehension of the subject matter. These case studies demonstrate the viability and applicability of accelerometer-based pothole detection systems by firmly establishing the project in current research, underscoring their function in tackling issues with road infrastructure around the world.

Furthermore, real-world examples highlight how creative solutions like this application can lower risks, lower accident rates, and enhance road maintenance scheduling. They stimulate more conversation about the advantages and disadvantages of existing approaches while promoting the investigation of cutting-edge technologies to improve the system even further. Questions concerning execution, scalability, and integration into larger smart city programs are raised by the insights gathered from this research, which serve as the basis for fruitful discussion.

In addition to making a strong case for its significance, this project encourages a cooperative and curious approach to tackling road safety concerns by utilizing these real-world case studies. These illustrations help close the gap between theory and practice by offering a strong foundation for upcoming developments and promoting participation and the investigation of novel concepts.

CHAPTER 3: REQUIREMENT ANALYSIS AND DESIGN

3.1 Introduction

This chapter will cover the design and analysis of requirements in further detail. To achieve the first goal, a questionnaire was used to collect user needs for mobile application design. This chapter will also cover the approach taken in creating the proposed project.

3.2 Methodology

A questionnaire was created and spread to collect feedback for the layout and operation of the RePothole mobile app to better understand road users' worries regarding potholes and road safety. An integral part of the research process, this questionnaire yielded insightful information that directly influenced the features and user experience of the program. The purpose of the questions was to learn more about the thoughts and experiences of drivers regarding potholes and road conditions. The survey also aimed to gauge users' frequency of pothole encounters, understanding of pothole-related problems, and willingness to embrace real-time pothole detection technologies. The idea of mapping potholes on a digital platform that allows people to see trouble spots on a map directly was also investigated in the study. By offering real-time pothole mapping and identification instead of conventional reporting to local authorities, this method aims to improve road safety.

To guarantee a well-rounded set of responses, the questions used multiple-choice, yes/no, and Likert scale (1–5, agree/disagree) formats. To get qualitative input on user experiences and recommendations for enhancement, open-ended questions were also incorporated. To guarantee a representative sample of the population, the questionnaire was given to a wide range of road users. Participants were questioned about their experiences with potholes, what they usually do when they come across one, and whether they would be open to using an app for reporting and detecting potholes. User preferences for app features like GPS tracking and real-time alerts were also covered in the poll.

To find recurring themes, problems, and possible areas for application enhancement, the gathered replies were examined. The design and functionality of the app were improved because of the insights this feedback gave into user requirements and expectations.

Following collection, the responses were grouped and examined according to recurring themes. Open-ended questions were examined qualitatively to obtain in-depth information, whereas

multiple-choice questions answers were examined numerically. The information was then utilized to develop user profiles and improve the app's functionality and user interface. The approach made sure that the RePothole app's final design matched the preferences and practical requirements of its target audience by concentrating on road user feedback.

3.3 Analysis

3.3.1 Questionnaires

A Google Form survey has been created that aims at the road users themselves. The Google Form received 40 respondents from various backgrounds. The survey was divided into 4 sections, which are General Information, Experience with Potholes, App Concept and Features, and Feedback and Improvements. In the first section, General Information, the respondent the age range, gender, how long they have been driving, do they always encounter potholes, how they rate the condition of potholes in their area, and the types of vehicles they use.

In Figure 3.1, 80% of the respondents are aged in the range of 18 to 24, 17.5% are in the 24 to 30 years old range, and 2.5% of the respondents are aged in the range of 35 to 44 years old while none of them are in the range of 45 years old and above. The Google Form survey's inclusion of a question regarding respondents' age range yields useful demographic information that may be used to customize the RePothole program to its target users' needs. Road usage trends, attitudes to embracing new technology, and technological skills may differ among age groups. Younger users are more likely to interact with real-time pothole detection and mapping functions since they may be more accustomed to smartphone applications. Older people, on the other hand, could value simple functionality and user-friendly interfaces. Knowing the age range of likely users helps to maximize the application's usability and adoption by ensuring that its features, interfaces, and communication styles are inclusive and appeal to all age groups. Additionally, by identifying the main group of people who are most likely to use or profit from the application, this data helps direct future marketing and development initiatives.

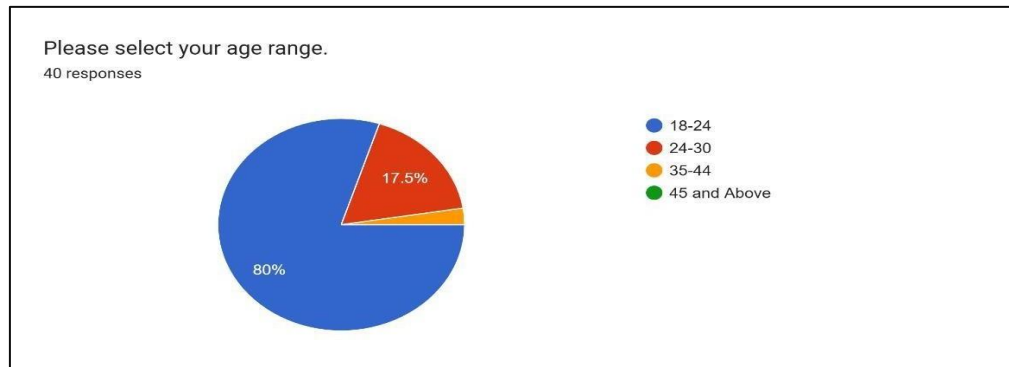


Figure 3. 1: Pie Chart-Age Range of the Responder

The pie charts shown in Figure 3.2 show that more than 45% of the respondents have been driving on the road for more than 5 years while 10% of them have experienced driving for 3 to 5 years. The other 25% driving on the road for 1 to 3 years and 20% of them have driving experience on the road for less than 5 years. Based on the driving or roadway experience of various user groups, this data aids in identifying trends in pothole encounters. While less experienced users might draw attention to the necessity for user-friendly interfaces and instructions within the application, more seasoned users might offer more thorough input on the significance of real-time pothole detection and mapping functions. Through the analysis of this data, the project can modify its functionality and design to accommodate the various needs of all road users, guaranteeing that the application is useful, easy to use, and efficient in enhancing road safety.

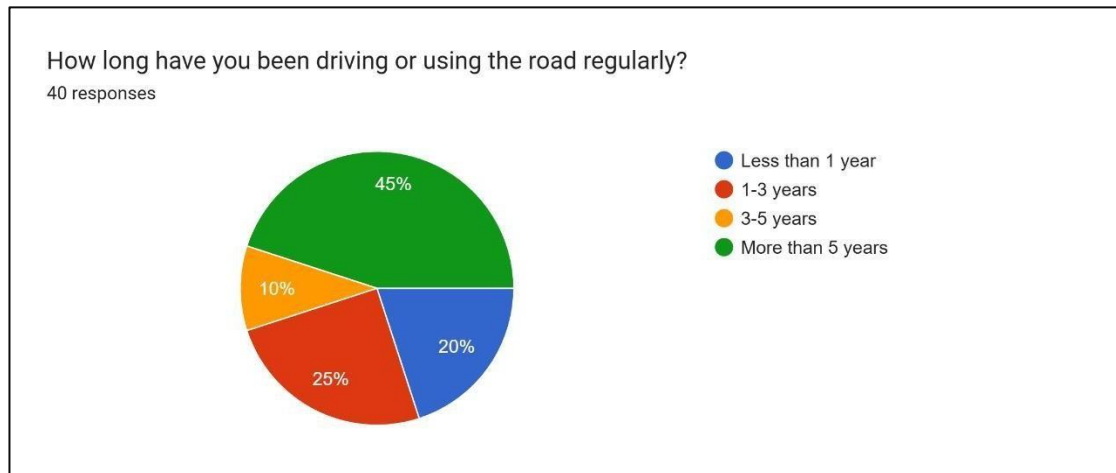


Figure 3. 2:Pie Chart-Driving Experience of the Responder

Most of the respondents, 87.5% of them, have experienced encounter potholes on the road as shown in Figure 3.3. The other 12.5% of them had no experience with potholes on the road in their area. The data offers insights into the frequency and distribution of potholes on frequently traveled routes by determining how frequently users experience them. High frequencies show that potholes are a common problem, highlighting the necessity for real- time pothole detection and mapping tools like RePothole. The desire for a technology that improves road safety by proactively recognizing and visualizing pothole areas is demonstrated by this data, which supports the app's significance. Frequent interactions also point to the possibility of high mobile application usage, which encourages the creation of a focused-on user system that tackles a major and persistent issue that road users encounter.

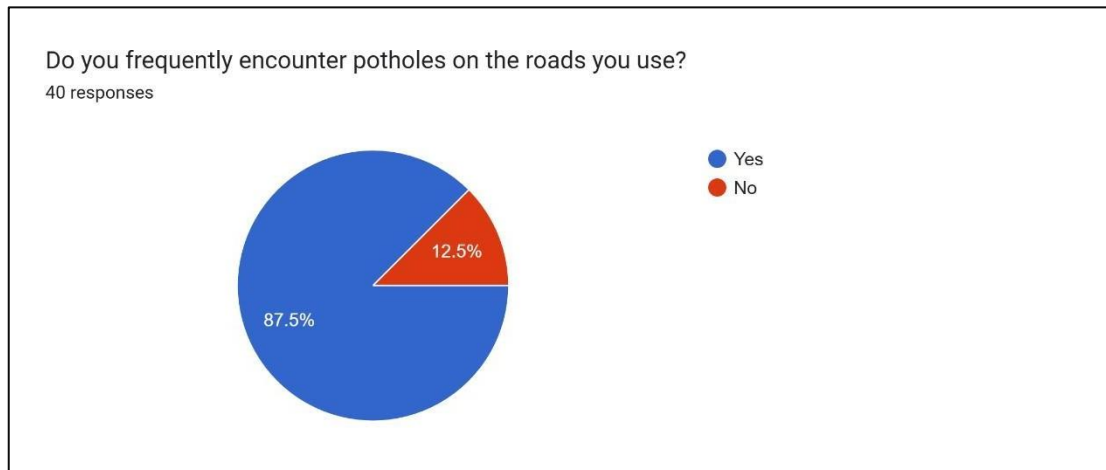


Figure 3. 3: Pie Charts- Experience with potholes on the road of the Responder

Based on the bar charts below in Figure 3.4, it is a linear-scale question in which the question is about the rating of the condition of the roads in the respondent area, from very poor to excellent. 10% of the respondents mentioned that the road condition is very poor, while 20% of the respondents mentioned that poor road conditions. 55% of them have moderate road conditions, and 10% of the respondents have good road conditions in their area. The balance of 5% of the respondents have very good road conditions in their area. The condition is rated as moderate by the majority. It draws attention to the necessity of products like RePothole to deal with road problems. Even while the roads might not be completely deteriorated, a few potholes and small damage can nonetheless seriously endanger road users' safety and cause them discomfort. The feedback supports the project's purpose by underlining the requirement for real-time pothole identification and mapping. It suggests that while full road repairs might not be urgently required, a system that identifies and visualizes localized road faults can nonetheless improve road safety and user experience. RePothole can guarantee that minor issues are identified early and dealt with effectively by concentrating on these mild conditions, which will stop future deterioration and improve road maintenance planning.

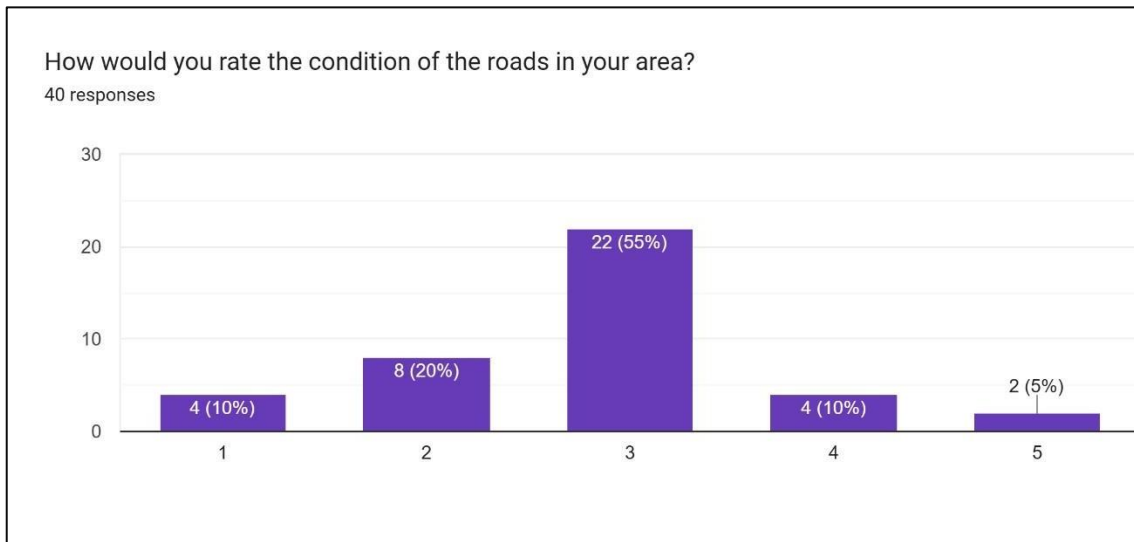


Figure 3. 4: Bar Chart- Rating of the road condition

In the bar chart shown in Figure 3.5, 87.5% of them used a car as a vehicle while 52.5% chose a motorcycle as a vehicle, and 7.5% of them used a bicycle. It is a multiple-choice question where respondents may choose more than one vehicle they primarily use on the road. Determining the diverse sensitivities and effects of potholes on various vehicles is made easier by knowing the kinds of vehicles that respondents primarily use, such as cars, motorbikes, or bicycles. Potholes, for instance, could be a minor annoyance to drivers of cars, but riders of motorcycles and bicycles are more susceptible to safety hazards because of their direct exposure to road conditions. This knowledge helps the application's mapping features and detection algorithms take into consideration the different sorts of vehicles' experiences with road hazards and vibration intensity. Additionally, the mobile application can prioritize features and interface designs that meet the needs of car users, like route navigation and mapping for larger road networks, by recognizing the primary vehicle type (most people use automobiles) in the future.

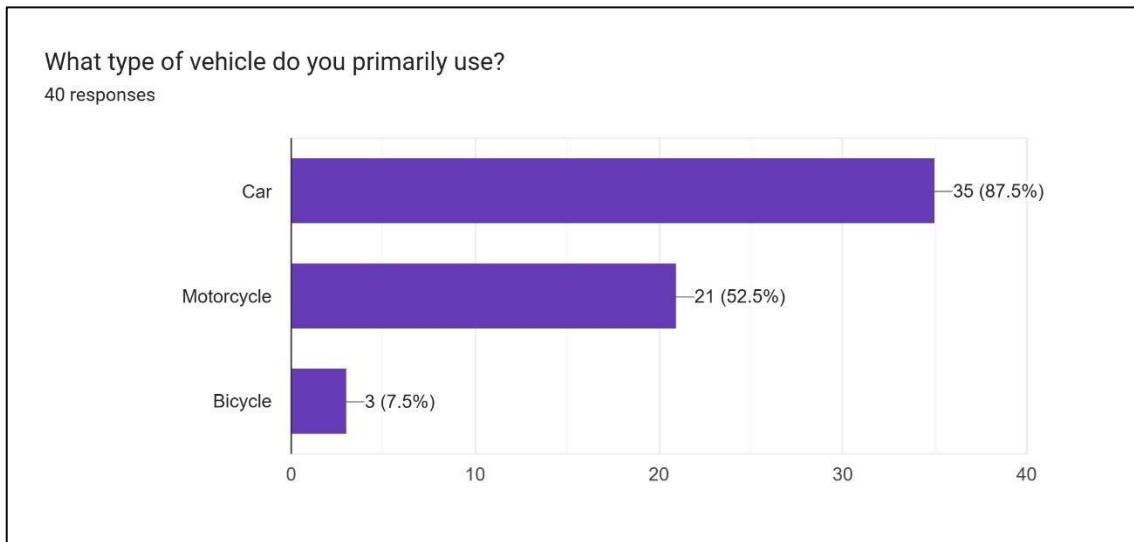


Figure 3. 5:Bar Chart- Type of vehicle primarily used

In section 2, the Experience with Potholes section, three questions were asked, including single and multiple-choice answers. This section aims to survey the presence of potholes in the daily life of the respondents.

In the first question, it is a single-choice selection question. The respondents were asked if they had any experience of accident or damage caused by potholes as shown in Figure 3.6. According to the responses, 35% of respondents had experienced accidents or damage brought on by potholes, underscoring the gravity of the problem and the possible threats pose to car maintenance and safety. This research highlights the necessity of a real-time pothole detection and mapping technology such as RePothole. By taking care of these issues, the software can assist users in proactively avoiding dangerous locations, which lowers the risk of collisions or damage to their vehicles. Additionally, 65% of respondents who selected "No" represent a population that would profit from the application's preventative features, such as increased knowledge and safer travel routes, even though they are not directly impacted. The integration of viewpoints enhances the project's rationale and illustrates its wide applicability to a range of road users.

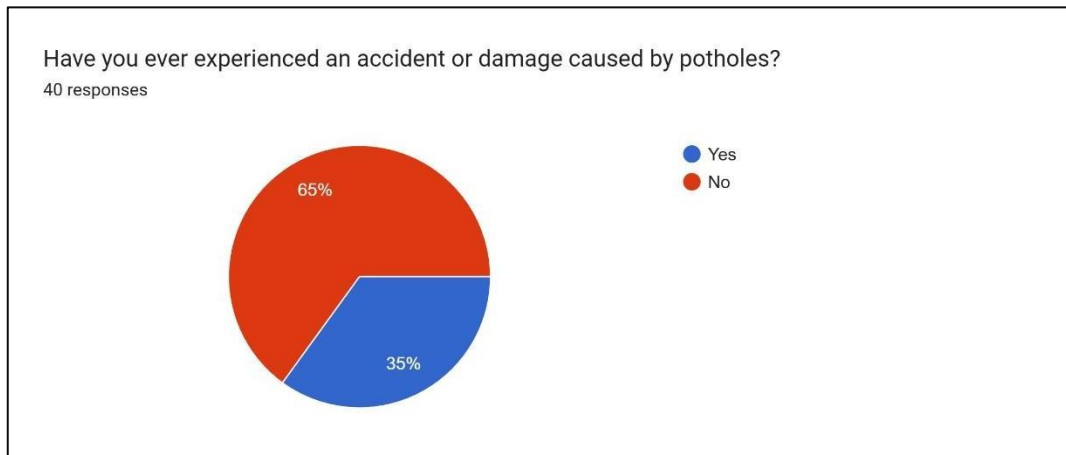


Figure 3. 6: Pie Charts- Experience an accident with potholes

In Figure 3.7, The survey's inclusion of the question "Have you ever reported a pothole to local authorities?" offers important information on how drivers behave and feel about fixing traffic hazards. The findings show a serious weakness in conventional pothole reporting procedures, with 82.5% of respondents saying they had never reported a pothole. This research highlights the necessity of a different strategy, such as the RePothole app, which does away with the requirement for consumers to manually notify authorities of problems. The application fills this gap by automating the identification and depiction of potholes on a digital map through the integration of real-time pothole detection and mapping capabilities. This increases overall road safety and opens the door for more effective road maintenance techniques by guaranteeing that road risks are recorded and available without requiring direct user action.

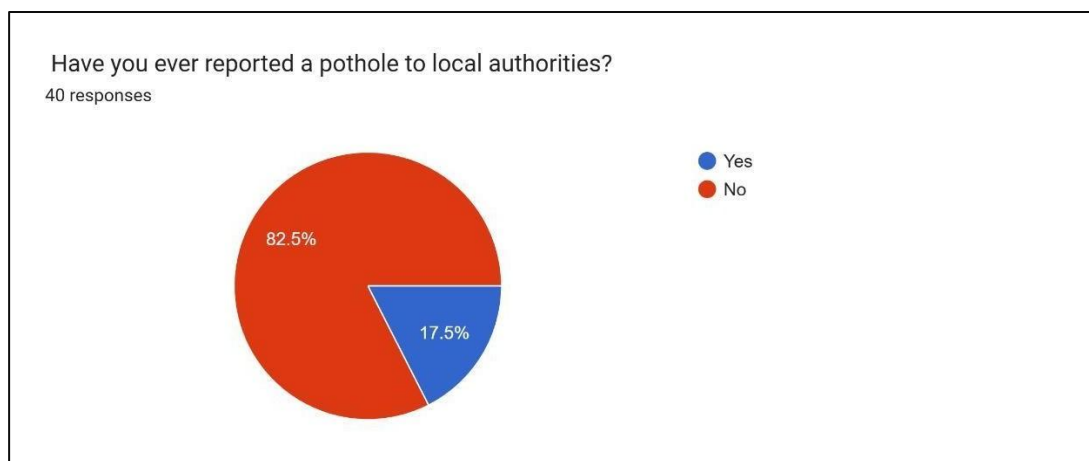


Figure 3. 7: Experience reported a pothole to local authorities

The question as shown in Figure 3.8 is a multiple-choice question where respondents were asked if the potholes affect their daily commute or travel. It offers important insights into the

practical difficulties that potholes cause for drivers. Potholes have a major impact on road safety, car condition, and the overall travel experience, as seen by the responses, which range from longer travel times (57.5%) to greater vehicle maintenance expenses (60%), and the likelihood of accidents (82.5%). The RePothole mobile application project benefits greatly from this data since it emphasizes the necessity of a prompt and effective response to pothole-related problems. The mobile application may be made to directly address these issues by determining the most prevalent and serious impacts that potholes have on day-to-day transportation. For instance, by offering real-time pothole mapping and recognition, the application can assist users in avoiding regions where potholes are common. This will cut down on travel time, prevent car damage, and eventually reduce the likelihood of accidents. By providing commuters with the means to keep clear of dangerous traffic conditions, this strategy supports the project's objective of increasing commuter efficiency and road safety.

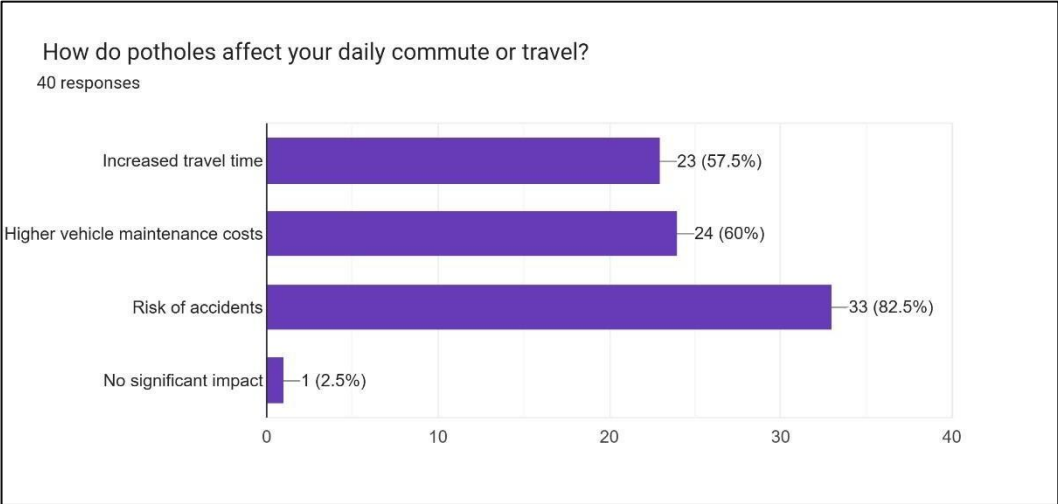


Figure 3. 8: The effect of potholes on the daily commute or travel

Road users appear to have a high need for such a solution, as evidenced by 87.5% of respondents who said they would be positively willing to use the app as shown in Figure 3.9. This input supports the necessity of a real-time mapping and pothole-detecting system, which is consistent with the main features of the RePothole app. Most positive reviews not only support the app's applicability but also serve as a guide for its design and development, guaranteeing that it lives up to user expectations. This survey result is essential for confirming the project's viability and directing the app's attention toward mapping pothole sites on a digital

platform since it indicates that consumers are receptive to implementing technology for real time pothole detection and visualization.

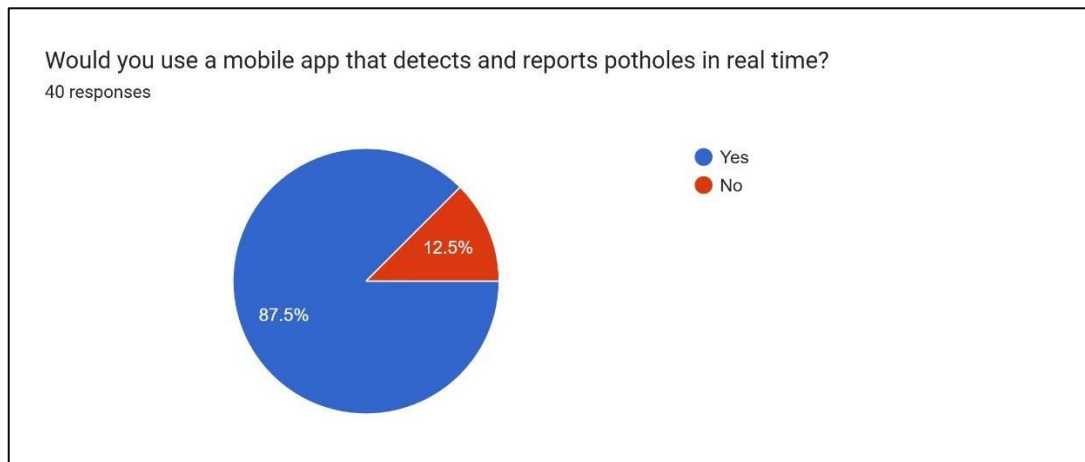


Figure 3. 9: The respondent clarification for the pothole's mobile application detector

Given that 95% of respondents said "yes," this indicates how eager users are to share their location information to enhance pothole mapping and detection in Figure 3.10. This encouraging reaction suggests that users are amenable to disclosing their current GPS location, which is a crucial function for precisely locating potholes on a map. The program can obtain accurate data from several users by taking advantage of this willingness, which enables it to produce an extensive, real-time map of potholes. The efficacy of the program depends on user-driven feedback because it will improve pothole mapping and detection accuracy and coverage across various regions, ultimately increasing road safety for all vehicles.

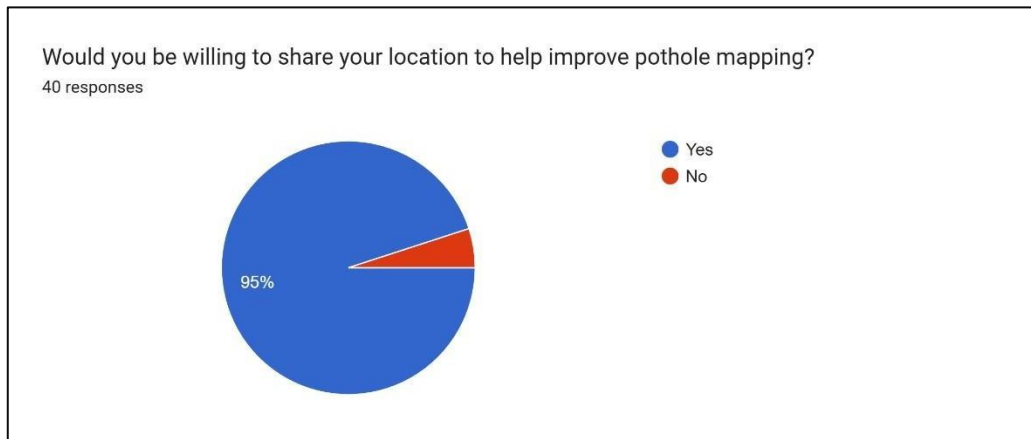


Figure 3. 10: Respondent's Willingness to Share Location

Since it directly influences the app's design and user experience, the question regarding the significance of having a straightforward and user-friendly interface is essential to the RePothole mobile application's success. According to the survey results based on Figure 3.11, a sizable majority of users (65%) think that an interface that is simple to use and intuitive is crucial, suggesting that usability is a big concern for prospective users. According to this criticism, RePothole should emphasize having a simple, user-friendly design so that users can easily interact with the app's functions, like real-time pothole mapping and detection. Because road users are more likely to use an app that is easy to use and intuitive, particularly when reporting and viewing potholes in real time, a user-friendly design will promote the app's wider adoption. By considering the preferences of the survey participants, the mobile application may concentrate on creating an interface that satisfies these requirements, improving user satisfaction and the application's overall efficacy in promoting road safety.

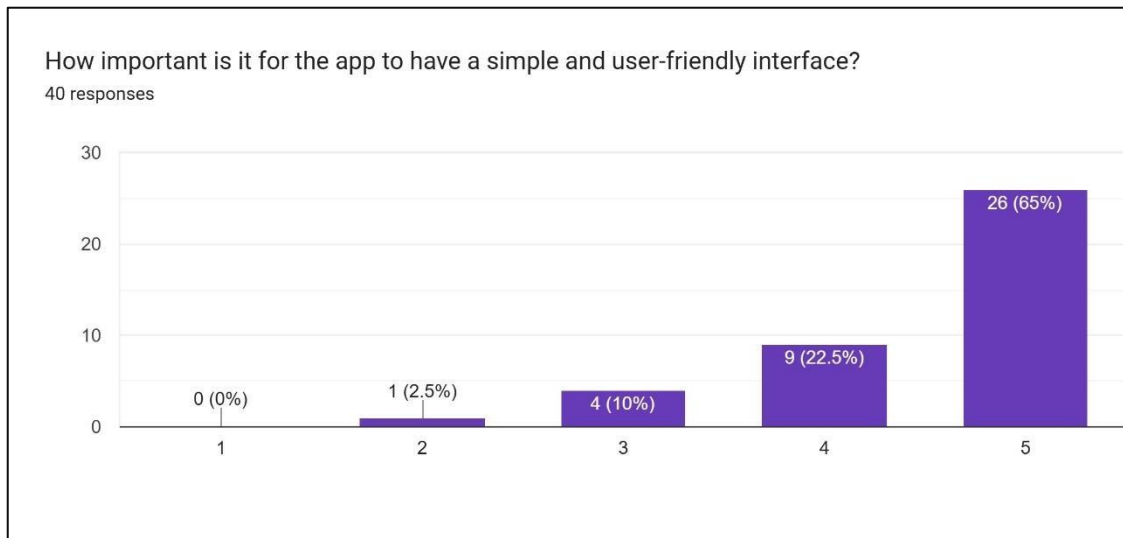


Figure 3. 11: The Importance of Simple and User-Friendly for the Mobile Application

The question asked in Figure 3.12 offers important information on the RePothole mobile application's likelihood of success and user acceptability. Since all respondents (100%) said "yes," road users are very interested in and supportive of real-time pothole mapping and identification. This feedback confirms the app's essential features and indicates that users find the software useful for sharing with others in addition to being interested in using it. The app's potential for widespread implementation is highlighted by the high desire to suggest it, which supports the project's objective of improving road safety through effective pothole mapping and identification. Additionally, by guaranteeing that the application meets user requirements and expectations, this encouraging feedback can direct future development and promote even greater road user participation and trust.



Figure 3. 12:Recommendation to Others to Use the Mobile Application

The answers to the question in Figure 3.13 offer insightful information that can greatly aid in accomplishing the objectives of the RePothole project, especially in terms of improving its features and guaranteeing its usefulness in real-world scenarios. The range of recommendations, including adding real-time reporting capabilities, pothole repair progress trackers, and road condition alerts, can assist in improving the program to better suit user requirements and preferences. These suggestions point out important areas where the app may develop, such as adding features to track pothole repair status or suggesting alternate routes.

These additions can enhance the user experience overall and increase the app's efficacy in realtime pothole mapping and detection. Additionally, suggestions about how to improve the mapping feature of the app by including information about pothole depth and severity might assist users in better comprehending how serious the discovered potholes are. Accessibility issues, such as making the app easier for senior adults to use and guaranteeing offline performance, offer helpful advice for enhancing the app's appearance and making sure it works in a variety of settings. Incorporating a feature that could enhance cooperation with local authorities and facilitate more effective repair priority is also demonstrated by the proposal to supply data to authorities such as JKR (Jabatan Kerja Raya). Together, these answers provide a clearer path for RePothole's development, ensuring that the app not only lives up to user expectations but also develops into a dependable real-time pothole mapping and detection tool that eventually improves road safety.

Do you have any suggestions or concerns about this app concept?

eg: Add a feature to track pothole repairs

40 responses

Figure 3. 13: Suggestions or Concerns About the Mobile Application

3.4 Design

The results of the user survey have a significant impact on the RePothole mobile application's design. The survey's key findings showed that 87.5% of participants regularly come across potholes, highlighting the need for a solution to deal with this enduring problem. Furthermore, 65% of users have never notified local authorities about potholes, underscoring the ineffectiveness of traditional reporting techniques and the possibility of automation. 95% of respondents said they would be prepared to share their location for precise mapping, and the majority (87.5%) said they would be willing to use an app for pothole detection. A straightforward and user-friendly interface was prioritized by 65% of respondents, demonstrating the clear emphasis on usability.

These findings informed the app's design, which gave priority to automatic reporting, real-time pothole identification, and user-friendly interfaces. A smooth experience is guaranteed by features like GPS-based mapping, accelerometer integration, and a simple design that relieve user worries. The application's emphasis on real-time updates and visualizations is in line with feedback from users regarding the urgent need for practical solutions to improve road safety.

3.4.1 Use Case Diagram

Figure 3.14 below shows the use case diagram for both mobile application and web application user for this proposed project.

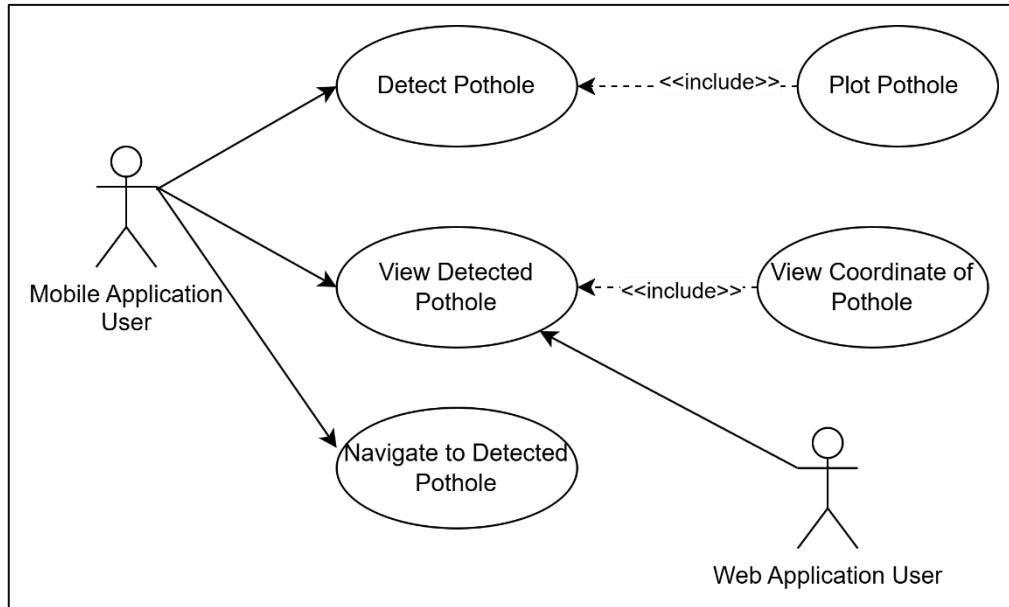


Figure 3. 14: Use Case Diagram for the Proposed Project

3.4.1.1 Use Case Description

The proposed application involves two user actors, which are the “Mobile Application User” and “Web Application User”. Table 3.1 shows a brief description of the actors involved, while Table 3.2 shows a brief description of the use cases.

Table 3. 1: Brief Description of the Actors Involved

Actor	Brief Description
Mobile Application User	The mobile app user is a person who uses their smartphone's RePothole app to find potholes in real-time. The user of the mobile app locates potholes and records their location using the GPS and accelerometer of the phone. Plotting the identified potholes on the map and keeping an eye on them when they travel their path are additional duties of this character.
Web Application User	A person who uses a web interface to access the RePothole system is known as a Web User. The locations of potholes that have been identified by users of mobile apps and plotted on a map are visible to web users. Although it cannot identify potholes, this actor can filter and work with the pothole data that is shown on the map. The main responsibilities of the web user are to track the condition of potholes and access real time mapping data, which can help with road design and management.

An overview of the use cases is shown in the use case diagram shown in Table 3.2 below:

Table 3. 2: The Brief Description for the Use Cases Involved

Use Case	Brief Description
Detect Pothole	By using the phone's GPS and accelerometer to detect and keep track of potential risks, it tackles the regular run-ins with potholes. This feature offers an early resolution to problems that users have brought to focus, like car damage and safety hazards.
Plot Pothole	This use case makes the data visible and usable by guaranteeing that identified potholes are displayed on a map. This is in line with user input that suggested an illustration of trouble spots to help plan safer journeys.
View Detected Pothole	This use case gives drivers a tool to help them avoid hazards by enabling them to keep an eye on mapped potholes. The application directly addresses users' concerns about road safety and the difficulties of navigating roads in bad condition by offering this option.

View Coordinate Pothole	For identified potholes, this use case offers exact GPS coordinates. Users can more easily plan their trips and stay clear of danger spots thanks to this tool, which lets users locate potholes with accuracy. This use case's inclusion supports the project's goal of improving road safety by reflecting user feedback regarding the significance of precise and in-depth road condition data.
Navigate to Detected Pothole	This use case combines mapping pothole locations with real-time navigation, enabling users to approach or avoid these regions as necessary. The survey's focus on safer navigation and real-time solutions is reinforced by this feature.

3.4.2 Application Sequence Diagram

Figure 3.15 below shows the process when the pothole is detected on the road while user using the RePothole Mobile Application.

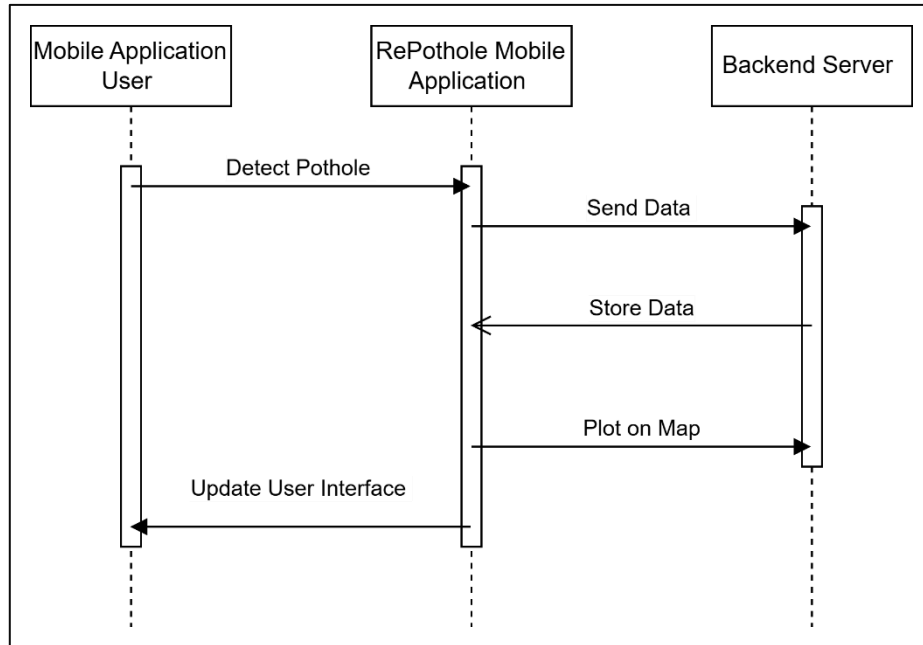


Figure 3. 15:Sequence Diagram When Pothole Detected on the Road

Figure 3.16 below shows the process when mobile application users want to view the detected pothole on the map.

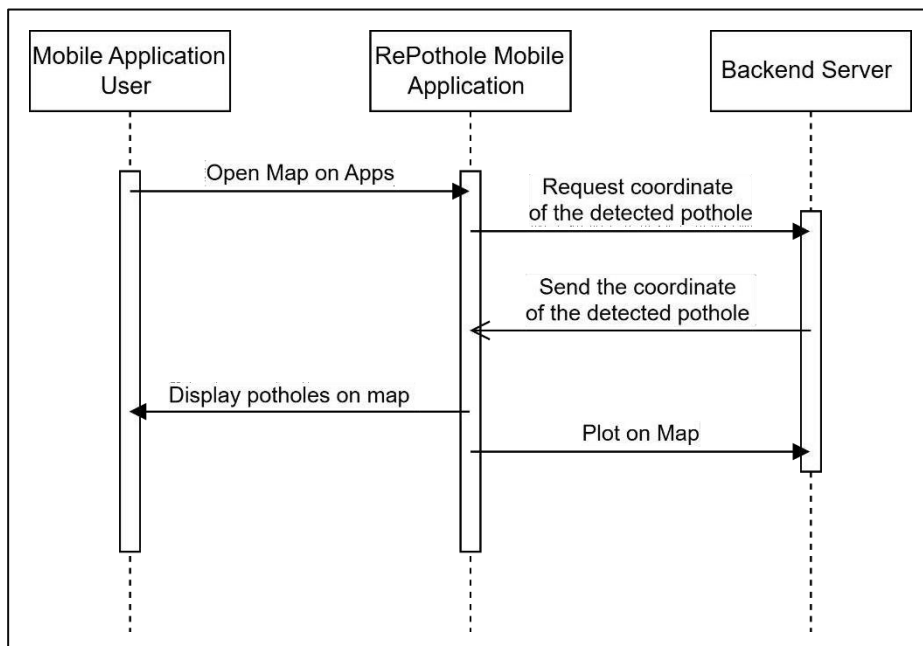


Figure 3. 16:View Detected Pothole on Map Sequence Diagram

Figure 3.17 below shows the process to navigate the detected pothole for both users.

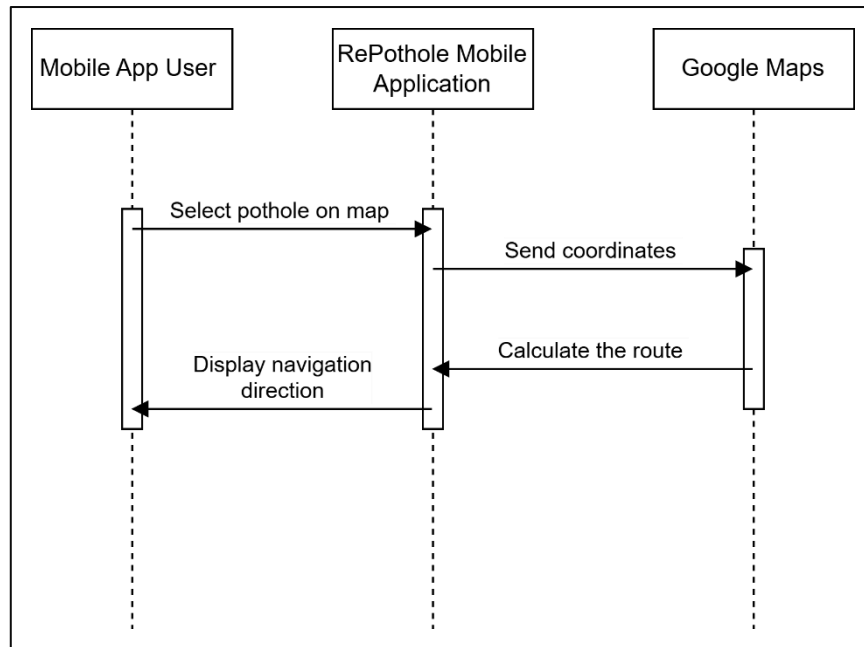


Figure 3. 17:Navigate to Detected Pothole Sequence Diagram

Figure 3.18 shows the process to view the coordinates of the detected pothole.

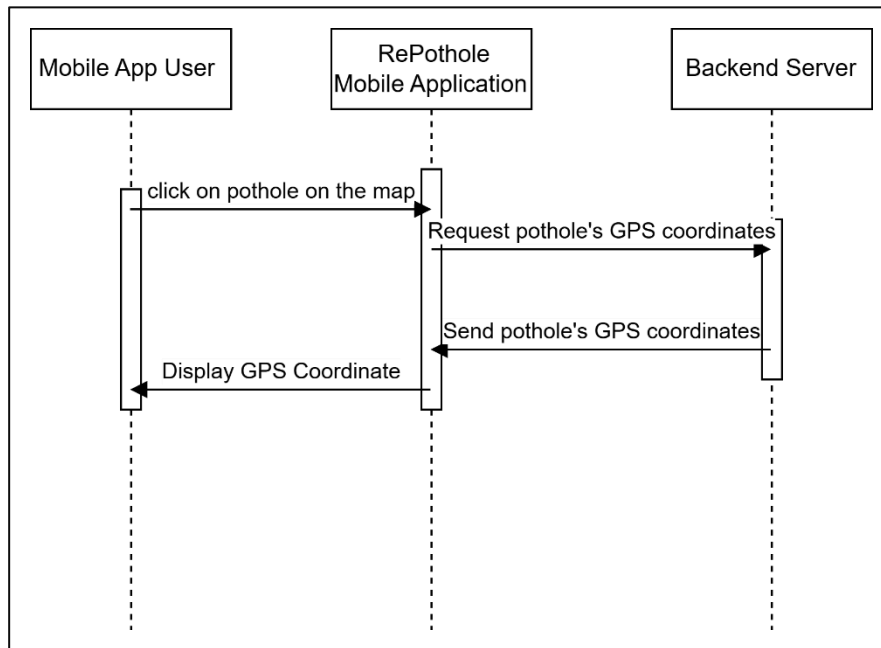


Figure 3. 18:View Coordinates of Detected Pothole Sequence Diagram

3.4.3 Activity Diagram

This section describes the entire process and interaction of RePothole mobile before, during, and after detecting the pothole as shown in Figure 3.19.



Figure 3. 19:Activity Diagram of RePothole Mobile Application User and Its Interactions Before, During and After Detecting the Pothole

The interaction for the Web User to view and navigate detected pothole is shown in the activity diagram in Figure 3.20 below.

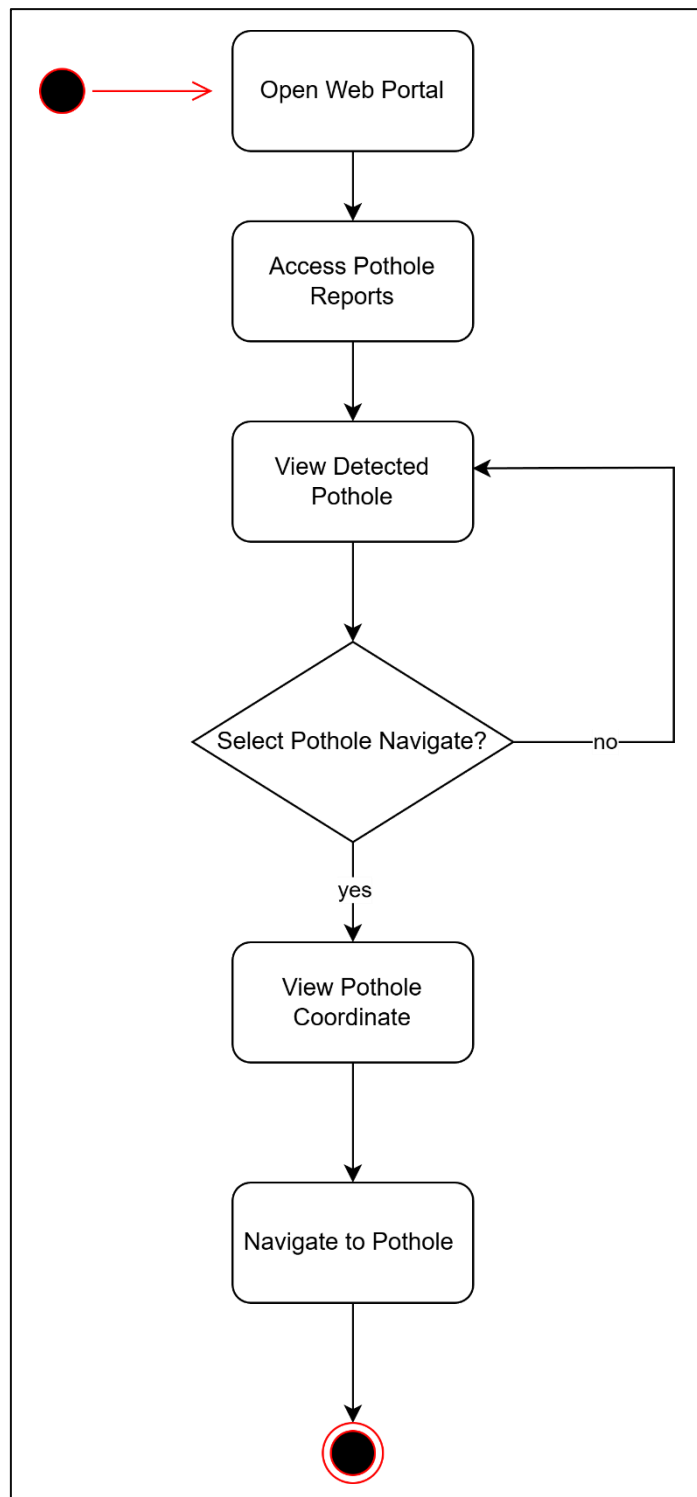


Figure 3. 20: The Interaction of Web User to Detect and Navigate Pothole

3.4.4 Class Diagram

Figure 27 shows the class diagram of this proposed project.

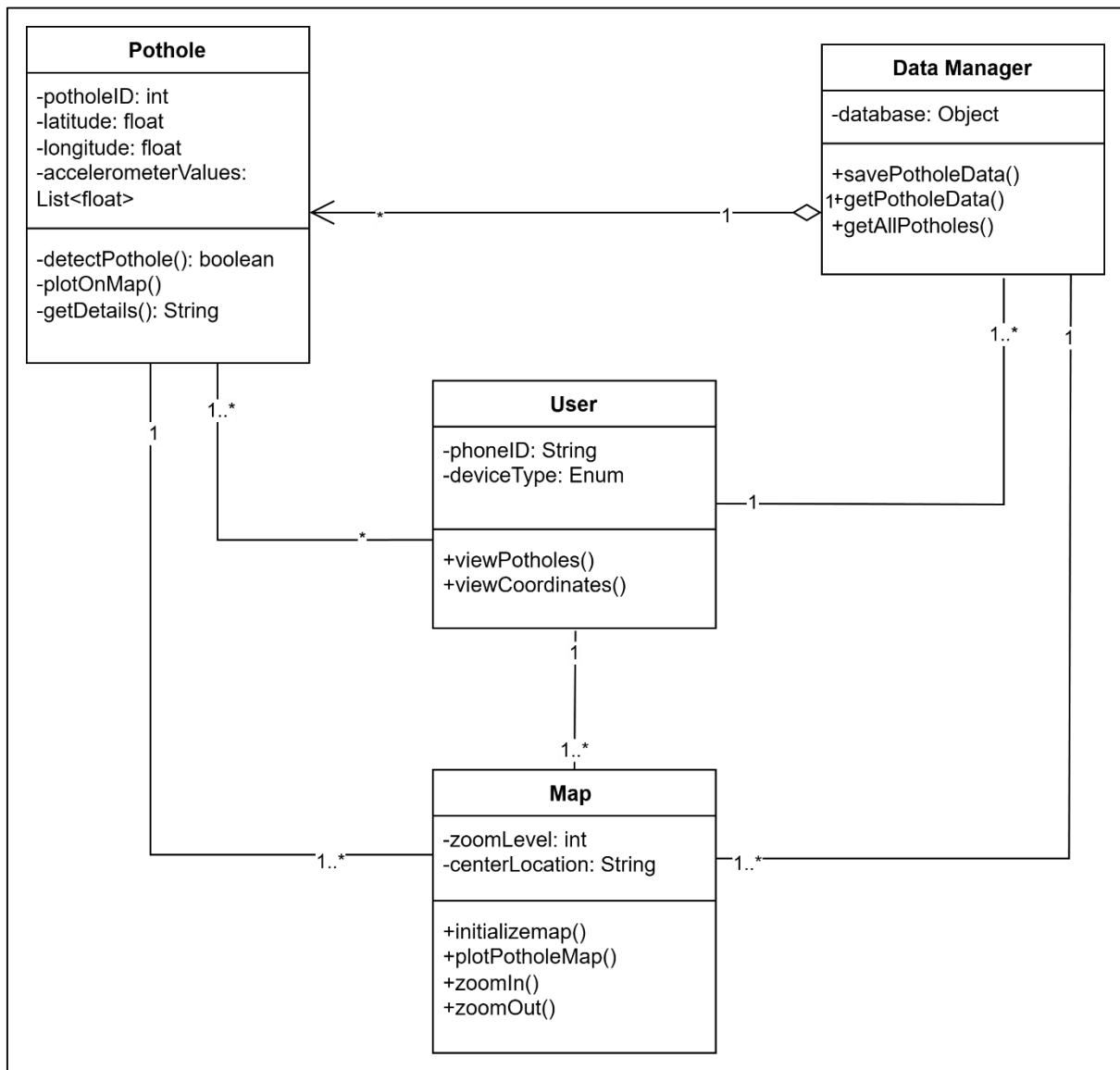


Figure 3.2 1: Class Diagram for the Interactions of RePothole Mobile Application

3.5 User Interface Design

3.5.1 Mobile Application UI

The RePothole app's splash screen is simple and clean, with a stark white backdrop that highlights the main components. The phrase "Pothole" is imaginatively enhanced in a variety of colours to strengthen brand identity, and the RePothole logo is prominently presented in a bold and colourful design in the top centre of the screen. The tagline "Smart Ride Ahead" is carefully positioned just below the logo, highlighting the app's primary goal. A circular, orange "Start" button that contrasts with the white background is positioned in the lower portion of the screen. Users are encouraged to start using the app because of its eye-catching colour and central placement, which make it simple to see and reach. The application has an appealing look thanks to its overall clean, contemporary, and user-friendly style. Figure 3.22 below shows the illustrations of the 'Start' button of the mobile application.



Figure 3.2 2: User needs to click the 'Start' button to start detecting the pothole

The main difference from the earlier section is the central button's change from "Start" to "Stop." This means that real-time pothole detection using onboard sensors has begun, and the app has entered an active state. The button's rounded shape and vivid orange colour are maintained, guaranteeing visual coherence while indicating interaction. The "Stop" button is conveniently located in the lower middle portion of the screen, allowing the user to terminate the session whenever they choose. All things considered, this page is made to be straightforward to use while communicating the app's current functioning state. Figure 3.23 below shows the interface for the illustration mentioned earlier.



Figure 3.2 3: The user needs to click the 'Stop' button to stop the pothole detection on the road.

This screen displays the RePothole application's real-time active detection state, as shown in Figure 3.24 below. With its clean white background, the prominently centered RePothole logo, and the orange "Stop" button that enables the user to stop the detection process, the layout visually resembles that of the previous "Stop" screen.

The addition of a live feedback the window at the bottom of the screen sets this screen apart. Once a

pothole has been found and successfully saved, this feedback window opens. Latitude, Longitude, Date, and Time are among the important details of the detection event that are displayed in a concise notification area. The input field is clear and placed at the bottom of the screen so that it is easily visible without interfering with the primary interface. The user is reassured by the successful logging and storage of their detection. Transparency and precision, which are essential for subsequent processing and map visualization, are supported by the clear timestamped location data. By instantly verifying each successful detection and keeping the user interface clear and useful, this design improves user confidence and usability.



Figure 3.2 4: A pop-up window appears if potholes it detected when the user clicks the 'Stop' button.

3.5.2 Web Application User Interface

The RePothole on the web interface offers a modern and flexible map-based style intended for both general users and local authorities to track pothole locations in real time, shown in Figure 3.25 below. To preserve brand identification and consistency with the mobile app, the RePothole logo is clearly displayed in the top-left corner. A modern "Search Pothole Here" search box in the upper-right corner enables visitors to find or filter pothole information fast by entering parameters such as place names, dates, or status. By making selected data easily accessible, this improves the user experience. At the bottom-right corner of the screen is a unique and easy-to-use feature: a circular animated icon that functions as the "Locate Me" button. When clicked, geolocation services are activated to determine the user's present location on the map. This function facilitates prompt decision-making and situational awareness, making it especially useful for users who want to see if there are any reported potholes nearby. Transparency, accessibility, and real-time interaction are given top priority in the overall design, which supports RePothole's objective of enabling safer and more intelligent road monitoring via an aesthetically pleasing and effective interface.

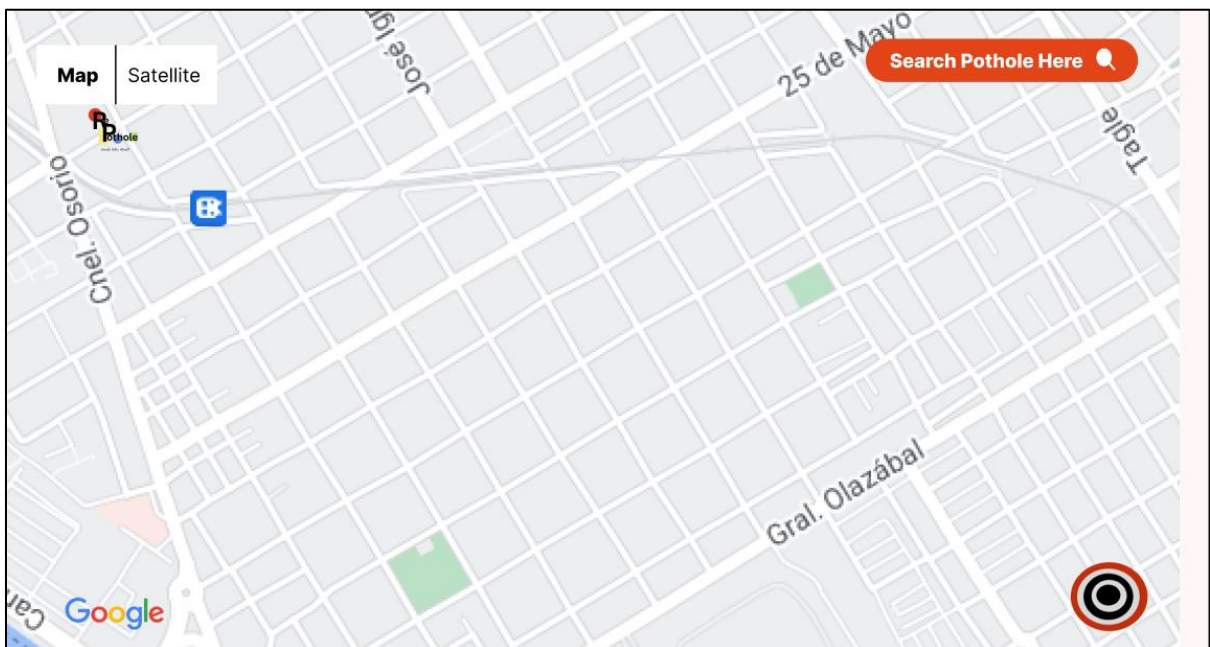


Figure 3.2 5: The RePothole website map interface features a circular animated "Locate Me" button at the bottom-right, a search box with the title "Search Pothole Here" at the top-right, and the logo at the top-left.

This screen illustrates the RePothole website's interactive filtering tool, as shown in Figure 3.26 below. As illustrated in Figure 3.25, a sliding filter panel displays seamlessly from the right side of the screen when the user hits the “Search Pothole Here” button from the main map interface. Three primary criteria, including Place (such as city or area), Date (particular detection date), and Status (such as Active or Fixed), can be used by users to filter pothole data in this panel. For more accurate results, users can apply one, two, or all three filters at once thanks to the design's support for multi-select filtering. The user can update the map with the relevant pothole data by selecting one or more filters and then clicking the "Go" button. By enabling users to rapidly filter the dataset according to their requirements or areas of interest, this feature increases user control and efficiency while also improving the system's usability.

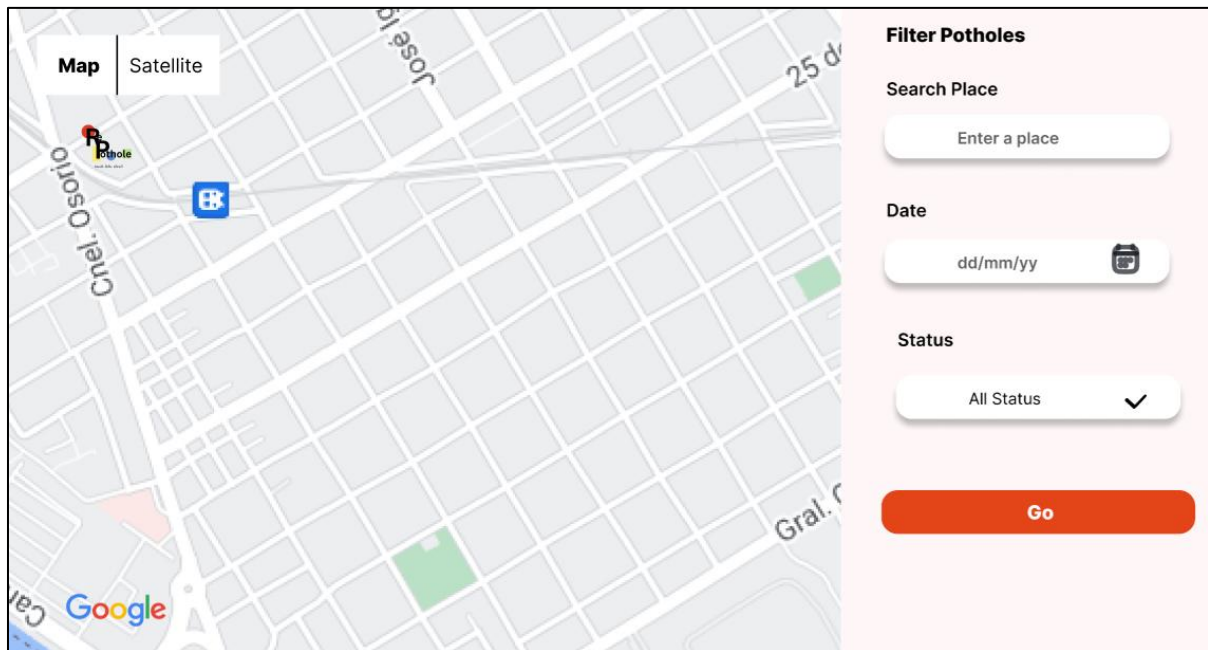


Figure 3.2 6: Filter panel slide in where the user can filter their search of the potholes based on the place, date, or status

The online map interface is made to give users an easy-to-use and interactive way to examine pothole data according to their filtered search criteria, as shown in Figure 3.27 below. The map automatically changes to show only what's needed after a user applies filters, such as location, date, or condition status. A red marker icon will appear on the map if a pothole is found in the chosen location. Since red frequently denotes a warning or a problem that requires attention, this deliberate usage of a red marker directs the user's attention to trouble spots on the map. A tiny information window, or pop-up, will show up above the selected marker when a user clicks on a red marker. Important information

regarding the pothole is displayed in this window, such as its unique ID, the date it was reported, and its current status (active or fixed). By ensuring that users can swiftly access important information without leaving the map, this feature enhances overall usability and cuts down on the number of steps required to evaluate or validate pothole data. Additionally, the pop-up window has a checkbox labelled with a prompt like "Has this pothole been fixed?" to encourage community input or field verification by road maintenance staff. This enables the user to verify whether the problem has been fixed. This section of the user interface is essential for keeping correct and current records. Through direct user interaction with each pothole entry, the system not only boosts user engagement but also seeks real-time pothole status updates, which lessens the administrative or backend team's effort. In addition to providing a useful purpose in preserving database correctness and traffic safety, its interactive design makes sure that the interface stays user-friendly.

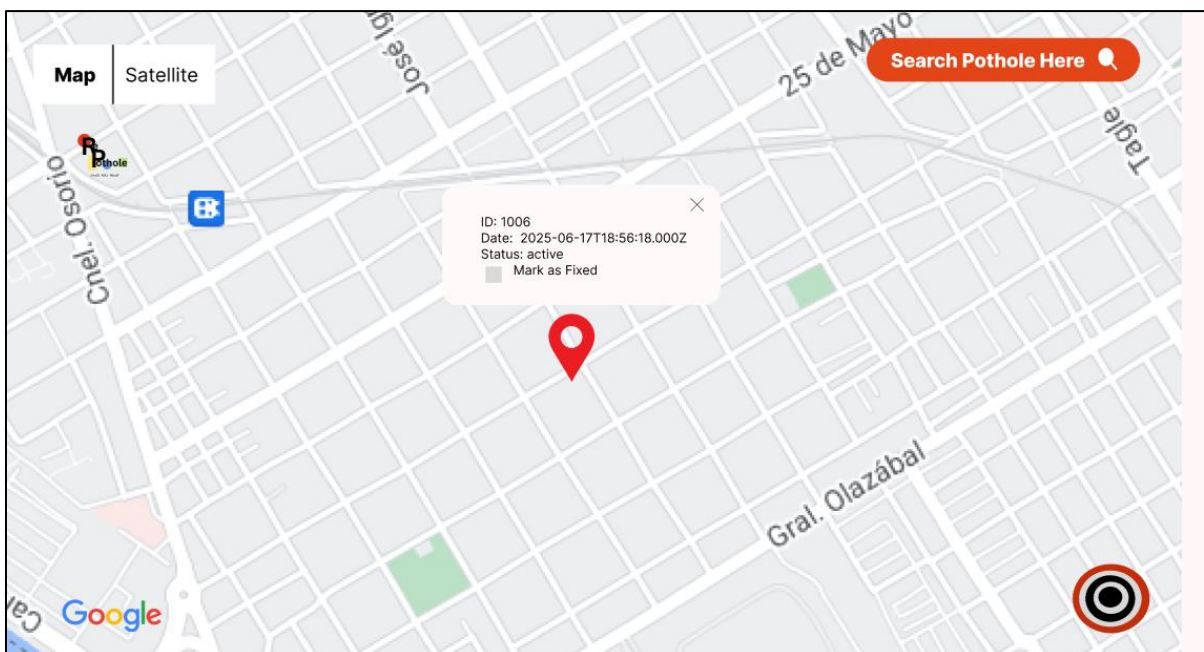


Figure 3.2 7: By clicking on the red markers, the user can view the filtered pothole results on the map and verify specifics.

When a user interacts with the "mark as fixed" button, the interface adds a confirmation layer to avoid unintentional alterations, which is referred to in Figure 3.28 below. "Confirm marking this pothole as fixed?" is the prompt that pops up on the screen as soon as the user checks the box indicating that a specific pothole has been fixed. Because it prevents users from unintentionally altering the pothole's status, this confirmation dialog box is a crucial component of the user experience design. "OK" and "Cancel" are the two options displayed in the notice. Should the user choose "OK," the system will

go ahead and change the pothole's database status to "FIXED." Nevertheless, the action will be stopped, and no modifications will be made if the user selects "Cancel." By urging users to be sure before submitting any revisions, this design offers a straightforward yet efficient precaution that maintains data quality and lowers the possibility of errors.

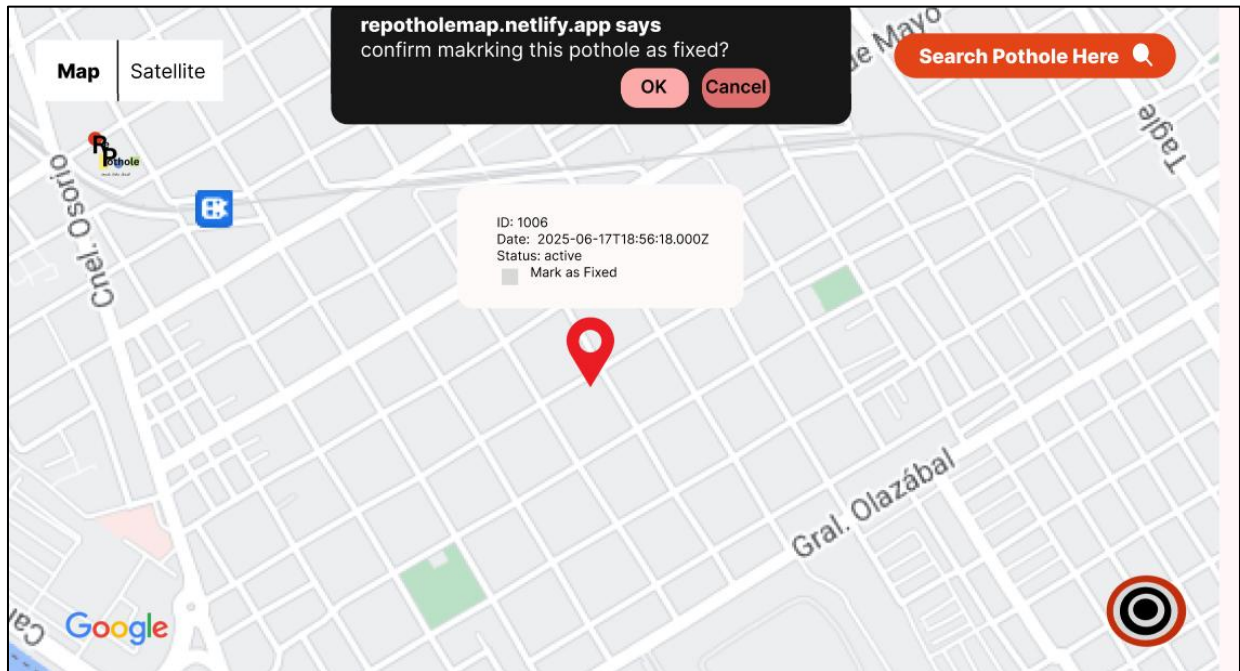


Figure 3.2 8: User receives a confirmation alert after checking 'mark as fixed' to verify their action before updating pothole status.

According to Figure 3.29 below, the system shows a last pop-up window that reads, "Status uploaded to FIXED," following the user's confirmation of the action to mark a pothole as mended. As user feedback, this confirmation message shows that the status modification was successfully entered into the database or system. The user is reassured that their input was received and handled correctly. After reading the message, the user can exit the pop-up and go back to the map interface by clicking the "OK" button. The status update flow is finished in this stage in an understandable and convenient way. Any interactive system must provide this kind of feedback since it validates the user's activity, eliminates doubt, and boosts application trust in general.

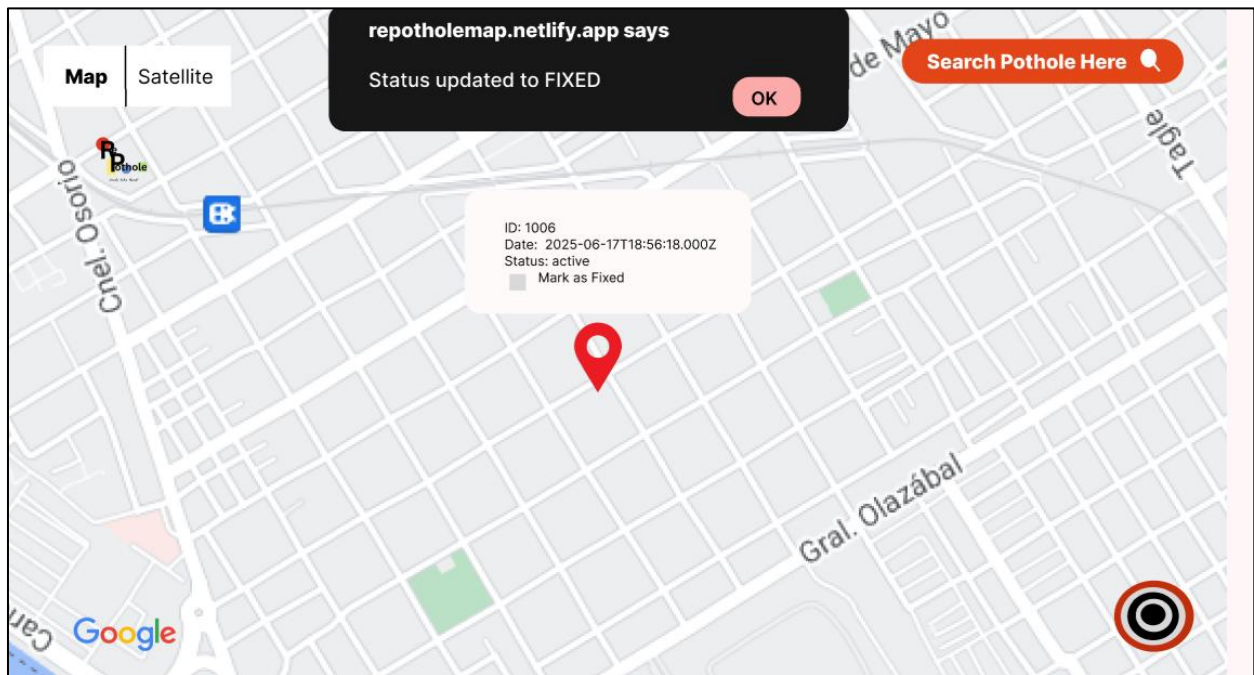


Figure 3.2 9: A confirmation message is sent to the user once the pothole has been successfully marked as mended.

According to Figure 3.30 below, the marker for a pothole turns from red to green once it has been correctly marked as corrected and the map has been updated. The interface design purposefully included this color shift to represent the pothole's revised state. Green marks show that a pothole has been rectified after being addressed, whilst red markers show that a pothole is still there or needs to be fixed. Without having to click on each marker, users can rapidly evaluate the state of potholes across the map thanks to this visual differentiation. By making it simple to discern between places that are faulty and those that have previously been fixed, it increases clarity and enriches the user experience. Additionally, this design facilitates effective decision-making for road maintenance crews or members of the public who might be monitoring the status of pothole repairs in particular areas.

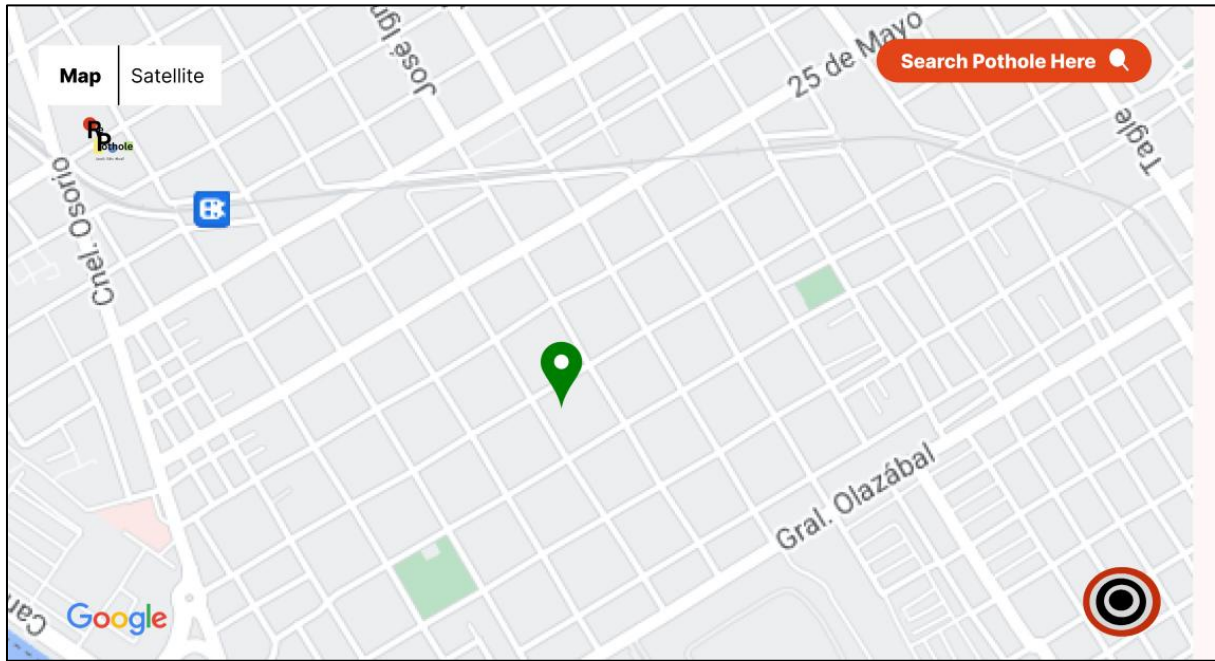


Figure 3.3 0: The Pothole has been marked as mended when the user refreshes and sees the marking change from red to green.

To assist users in rapidly navigating to their current geographic location on the map, the interface features an active circular icon called "Locate Me," shown in Figure 3.31. When the user touches this icon, the map uses the GPS capabilities of the smartphone to automatically center and zoom in on the user's current location. For customers who are constantly moving or who wish to quickly determine whether there are any potholes in their immediate area without having to browse the map, this feature is extremely helpful. The icon's motion attracts attention, signals its real-time capabilities, and gently invites interaction. This feature enhances usability, adds convenience, and facilitates real-world decision-making by letting users view nearby potholes about their current location. It allows drivers or maintenance teams to see whether there are any potholes nearby or on their current route.

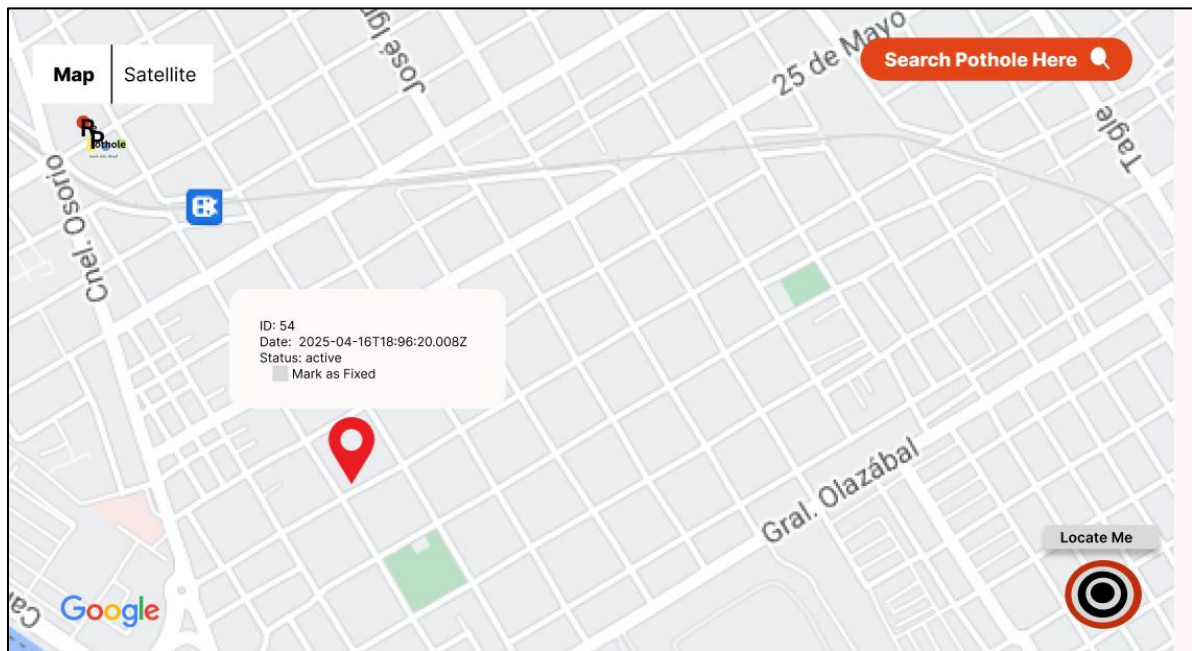


Figure 3.3 1: The user looks for potholes nearby, and the user finds their present location on the map.

3.6 Chapter Summary

A complete review of the requirements analysis and design for the RePothole mobile application is given in Chapter 3. The creation of a user survey to learn more about the requirements and preferences of road users is described in the methodology section. The application's design and functionality were influenced by important discoveries, including the high frequency of pothole encounters and the absence of efficient reporting channels. The design section shows how the survey's findings influenced the app's main functions, guaranteeing that they matched project goals and user expectations. Use case discussions highlight how each capability solves practical issues and improves road safety, further demonstrating its applicability. The RePothole project positions itself as a workable and user-centered solution to the common problem of potholes by basing the design on user feedback.

CHAPTER 4: IMPLEMENTATION AND TESTING

4.1 Introduction

This chapter will cover every aspect of the product, the implementation based on the design presented in Chapter 3. The software and techniques outlined in the previous chapter are applied in developing the planned pothole-detecting mobile application. The features and user interface of the suggested application are described in detail in the sections that follow.

4.2 Development Environment Setup

Cross-platform technology, mobile hardware access, and backend integration were all used in the creation of the RePothole mobile application to make sure the system could consistently identify potholes, log data, and store it. The main platforms, tools, and libraries utilized during implementation are described in this part, along with the justification for their selection. Tables 4.1, 4.2, 4.3, 4.4, 4.5 & 4.6 below show the setup and tools used in this RePothole mobile application project.

4.2.1 Frontend (Mobile App)

Table 4. 1: The frontend tools used and their purpose in this project.

Tool	Purpose
Flutter	A structure used to develop mobile apps that function on several platforms
Dart	The programming language that Flutter runs
Android Studio and VS Code	Integrated structure, a framework for development
Samsung Galaxy Tab A8	Physical devices for practical testing
Flutter Web (Chrome)	Used for testing via a browser when a device is not available.

4.2.2 Flutter Packages

Table 4. 2: The Flutter package and its usage in the RePothole project.

Package	Usage
path_provider	Search for the device's file storage paths.
intl	Convert DateTime into strings that can be displayed.
sensors_plus	Detect vibrations by accessing the accelerometer sensor.
geolocator	Fetch the device's latitude and longitude (GPS).
permission_handler	Demand the user's location and storage authority.

4.2.3 File Handling and Local Storage

Table 4. 3: The file handling and local storage tools and their purpose.

Tool	Purpose
file_service.dart	Pothole data is saved to a local file, and unsynced logs are uploaded.
pothole_log.txt	Each pothole event is stored in a local file (lat, long, timestamp, status).

4.2.4 Backend and Server

Table 4. 4: The backend and server tools and their usage.

Tool	Usage
JavaScript	Uses Netlify Functions and serverless functions to handle backend functionality and process data provided from Flutter.
Neon (PostgreSQL)	Pothole logs are effectively stored in a PostgreSQL database hosted in the cloud.
Netlify Function	Serves as a serverless API endpoint for the Flutter app, receiving and processing data.
JSON	Transfer of data format from Flutter to PHP script.

4.2.5 Testing and Debugging

Table 4. 5: The testing and debugging tools and their usage.

Tool	Usage
Postman	Used for manual upload.php endpoint testing using JSON data.
Flutter Console Logs	During detection, debug messages and sensor readings are.
XAMPP	Utilized for local testing at first, serverless deployment eventually took its place.

4.2.6 Networking and Communication

Table 4. 6: The networking and communication tools and their usage.

Tool	Usage
http.post()	Transmits JSON-formatted pothole data to the Netlify Function endpoint from the Flutter application.
Netlify Function URL	Serves as the backend endpoint that sends pothole data to the Neon database after receiving it.
Netlify (Frontend Hosting)	Makes the Flutter Web application available at a public URL by hosting it.
Live Internet Connection	Need to facilitate communication between the cloud-hosted backend services and the Flutter Web application.
Device Browser (Chrome)	A web browser on an actual internet-connected device is used to test and run the Flutter application.

4.3 Software Implementation

4.3.1 Accelerometer Integration

One of its main features is the RePothole mobile application's ability to use the smartphone's built-in accelerometer to identify potholes. The accelerometer can be used to measure the number of vibrations the device experiences by providing data from the sensors on the x, y, and z axes. When a car drives over a pothole, it usually causes sudden, acute vibrations. The RePothole app is made to recognize this physical behaviour.

The sensors plus package in Flutter is used by the application to continuously listen to accelerometer events to accomplish this. Acceleration values along the three axes of motion are provided by these occurrences. The Euclidean distance formula ($\sqrt{x^2 + y^2 + z^2}$) is used to determine the magnitude of these vectors, creating a single scalar number that represents the vibration's strength. Figure 4.1 below shows the formula used for the detection of the pothole.

```
double magnitude = sqrt(  
  event.x * event.x + event.y * event.y + event.z * event.z,  
);
```

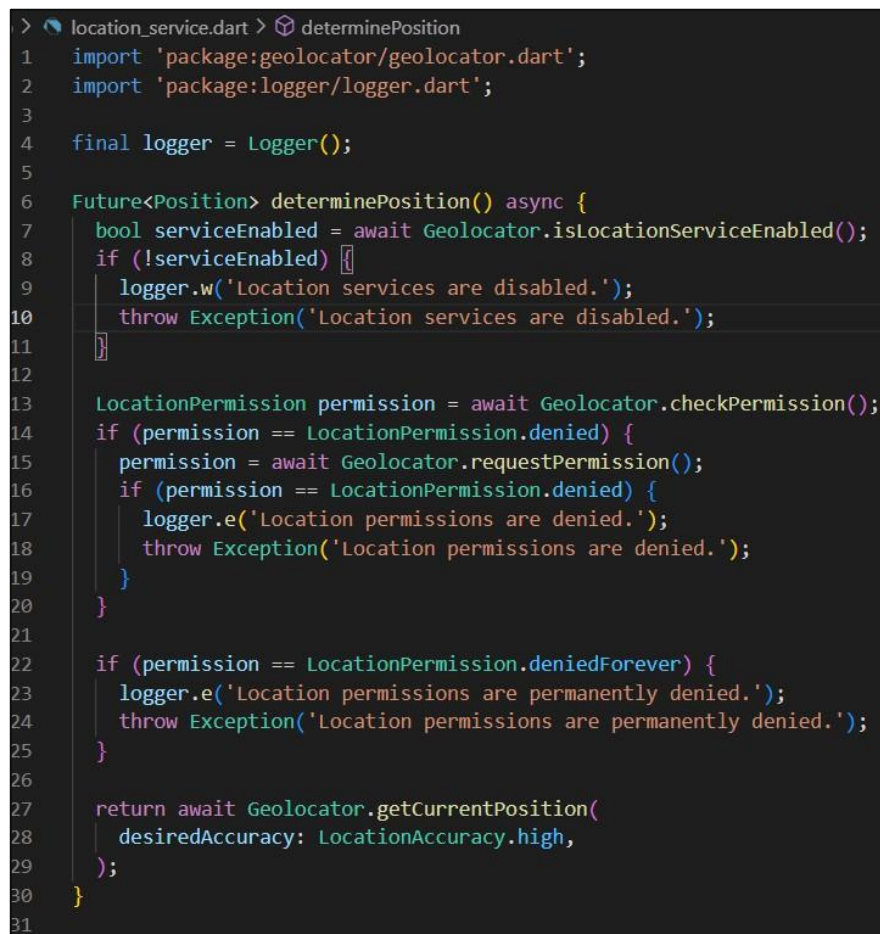
Figure 4. 1: The Euclidean distance formula is used to calculate the vibrations.

4.3.2 Location Tracking

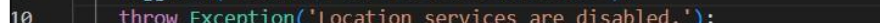
As it enables the system to identify each pothole detection with precise GPS coordinates, location tracking is an essential feature of the RePothole program. This increases the data's utility and significance for later use, such as showing pothole spots on a map or sending them to the right local government for road maintenance.

To determine the user's present position as well the application makes use of the Geolocator package. Only when the accelerometer detects a pothole is the position noted. DeterminePosition() is the main function that deals with this. A shorter form of the code is shown below in Figure 4.2.

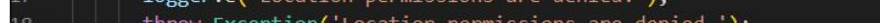
```
> location_service.dart > determinePosition
1  import 'package:geolocator/geolocator.dart';
2  import 'package:logger/logger.dart';
3
4  final logger = Logger();
5
6  Future<Position> determinePosition() async {
7    bool serviceEnabled = await Geolocator.isLocationServiceEnabled();
8    if (!serviceEnabled) {
9      logger.w('Location services are disabled.');
```



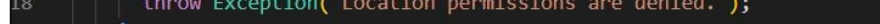
```
10   throw Exception('Location services are disabled.');
```



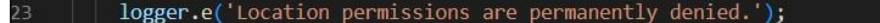
```
11 }
12
13 LocationPermission permission = await Geolocator.checkPermission();
14 if (permission == LocationPermission.denied) {
15   permission = await Geolocator.requestPermission();
16   if (permission == LocationPermission.denied) {
17     logger.e('Location permissions are denied.');
```



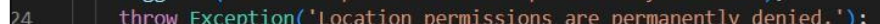
```
18     throw Exception('Location permissions are denied.');
```



```
19   }
20 }
21
22 if (permission == LocationPermission.deniedForever) {
23   logger.e('Location permissions are permanently denied.');
```



```
24   throw Exception('Location permissions are permanently denied.');
```



```
25 }
26
27 return await Geolocator.getCurrentPosition(
28   desiredAccuracy: LocationAccuracy.high,
29 );
30 }
31
```

Figure 4. 2: Codes of location_service.dart file.

4.3.3 Data Logging

To ensure that pothole detection data is preserved even while the device is offline, the RePothole program uses a fault-tolerant, local-first logging approach. The `file_service.dart` file implements the logic, which entails saving data locally and using a Netlify Function to upload it asynchronously to the Neon PostgreSQL database. The `savePotholeLog()` function is triggered when the app uses the device's accelerometer to identify a pothole. Using the `intl` package, this function generates a formatted log entry that contains the latitude, longitude, and a timestamp that is readable by humans. Two local text files, `pothole_log.txt` (for general logging) and `debug_log.txt` (used as a synchronization queue), have the log appended to them. To prevent excessive logging, the function does a quick duplicate check before writing by comparing the new entry with the previous one in `debug_log.txt` as shown in Figure 4.3 below.

```
if (await debugFile.exists()) {
  final lines = await debugFile.readAsLines();
  if (lines.isNotEmpty) {
    final lastLine = lines.last;
    if (lastLine.split('|').first.trim() ==
        '$latitude, $longitude, $formattedTimestamp') {
      logger.w('⚠ Skipped duplicate entry.');
```

Figure 4. 3: The duplicate check by comparing new entries with the previous one.

After reading the `debug_log.txt` and removing any items that are not synced, the upload function parses the components (latitude, longitude, and timestamp) and sends them all in JSON format to a Netlify Function endpoint as shown in Figure 4.4 below.

```
final response = await http.post(
  Uri.parse(url),
  headers: {'Content-Type': 'application/json'},
  body: jsonEncode({
    'latitude': latitude,
    'longitude': longitude,
    'timestamp': timestamp,
    'status': 'SYNCED',
    'condition_status': 'active',
  })),
);
```

Figure 4. 4: The pothole information upload to Netlify Function in JSON Format.

This Netlify Function manages the backend logic and adds the received data to the PostgreSQL database contained on the Neon cloud, offering a dependable and scalable cloud infrastructure independent of conventional local servers. The appropriate line in `debug_log.txt` is modified, converting the `UNSYNCED` tag to `SYNCED`, if the upload is successful (HTTP status 200 and the answer contains "success"). By doing this, repeated uploads are avoided, and only new data is kept in the sync queue. The `markLogsAsSynced()` function replaces the status after checking each line individually as shown in Figure 4.5 below.

```
final updated = allLines.map((line) {  
  return syncedEntries.contains(line.trim())  
    ? line.replaceFirst('UNSYNCED', 'SYNCED')  
    : line;  
}).toList();  
  
await file.writeAsString(updated.join('\n'));
```

Figure 4. 5: Status changes from UNSYNCED to SYNCED

Pothole data is never lost because to this complete data logging process. The solution facilitates offline use while allowing real-time cloud synchronization upon restoration of connectivity by fusing local file-based storage with contemporary cloud services such as Neon and Netlify. For road condition reporting in the RePothole application, this hybrid logging architecture enhances scalability and dependability.

4.3.4 Save Pothole Information in Database

Through a Netlify Function called `add-potholes.js`, the RePothole application transmits pothole detection data to the backend. This serverless function makes sure that every detection is consistently saved for further analysis and visualization by taking JSON-formatted data from the Flutter app and putting it into the cloud-hosted Neon PostgreSQL database. The `@netlify/neon` package's `neon` module, which is used to establish a secure, managed connection straight to the Neon database, is imported at the start of the function. The `neon()` method is used to construct a Neon SQL client, which is then assigned to the constant `sql`. Figure 4.6 below shows the code illustrated earlier.

```
import { neon } from '@netlify/neon';  
  
const sql = neon();
```

Figure 4. 6: The @netlify/neon package imported to manage connection to the Neon database.

Figure 4.7 below shows the function that verifies only POST requests are permitted by checking the HTTP method when a request is received. It enforces the proper usage pattern for data input by returning a 405 Method Not Allowed response if an alternative method is used.

```
if (event.httpMethod !== 'POST') {  
  return {  
    statusCode: 405,  
    body: 'Method Not Allowed',  
  };  
}
```

Figure 4. 7: Only POST requests are permitted by checking the HTTP method.

The function parses the incoming JSON data from the Flutter frontend for valid POST requests. Latitude, longitude, timestamp, status, and condition_status values are all included in the intended payload. Robust handling is ensured by using default values for status and condition_status in the event that they are absent. The illustration is shown in Figure 4.8 below.

```
const data = JSON.parse(event.body);  
  
const latitude = data.latitude;  
const longitude = data.longitude;  
const timestamp = data.timestamp;  
const status = data.status || 'SYNCED';  
const condition_status = data.condition_status || 'active';
```

Figure 4. 8: The pothole informations (latitude,longitude,timestamp,status and condition_status) included in the payload.

The function uses Neon's template literal syntax to run a SQL INSERT command after the data is ready. Latitude, longitude, timestamp, status, and condition_status are the five values that are inserted into the pothole_log table when the query adds a new pothole record. Figure 4.9 below shows the function that can ensure success by returning the entire inserted row thanks to the RETURNING * condition.

```
const result = await sql`  
  INSERT INTO pothole_log (latitude, longitude, timestamp, status, condition_status)  
  VALUES (${latitude}, ${longitude}, ${timestamp}, ${status}, ${condition_status})  
  RETURNING *;  
`;
```

Figure 4. 9: The new data for plotted pothole successfully inserted into the pothole_log.txt by the RETURNING condition.*

Figure 4.10 below shows that the function produces a 200 OK response, a success message, and the inserted data in JSON format if the insertion is successful. This response can be used for logging or client-side feedback.

```
return {
  statusCode: 200,
  body: JSON.stringify({
    message: 'Pothole added successfully!',
    data: result.rows,
  }),
}
```

Figure 4. 10: The alert "Pothole added successfully" signed that insertion is successful.

The function captures the exception, reports the error for debugging, and returns a 500 Internal Server Error response along with the error message if there is any error during parsing or database insertion, as shown in Figure 4.11 below.

```
}
} catch (err) {
  console.error('Error inserting pothole:', err);
  return {
    statusCode: 500,
    body: `Error inserting pothole: ${err.message}`,
  };
}
}
```

Figure 4. 11:Error warning shown if there is error with the response.

The scalable, safe, and serverless method of storing pothole data called cloud native. By using Neon for database storage and Netlify Functions for backend functionality, this method does deal with the requirement for conventional server configurations and facilitates contemporary development techniques, making the RePothole system lightweight and simple to maintain.

4.3.5 Differentiate Between Pothole and Bump

The `accelerometer_listener.dart` file is in charge of using data from the device's GPS and accelerometer sensors to identify potholes in real time. The `sensors_plus` package, which measures the device's vertical movements, is used to monitor the Z-axis acceleration, which forms the basis of the detection system. The phone suddenly drops vertically as the car hits a pothole, which is represented by a negative spike in the Z-axis value. For such a spike, which is known as

zNegativeSpike, this code establishes a threshold of -12.0. Using a stream subscription, it continuously listens to accelerometer signals and only processes those with substantial vertical deviations ($z < -8$ or $z > 12$) to eliminate normal or minor results. Using a timestamp comparison with `_lastDetection`, a 3-second cooldown delay is imposed between each detection to prevent multiple detections from the same pothole or road abnormality. The illustration is shown in Figure 4.12 below.

```
import 'dart:async';
import 'package:sensors_plus/sensors_plus.dart';
import 'package:geolocator/geolocator.dart';
import 'package:pothole_logger.dart';
import 'package:logger/logger.dart';

final logger = Logger();

class AccelerometerListener {
  StreamSubscription<AccelerometerEvent>? _subscription;

  static const double zNegativeSpike = -12.0;
  static const int cooldownSeconds = 3;

  DateTime _lastDetection = DateTime.now();

  void startListening() {
    _subscription = accelerometerEvents.listen((AccelerometerEvent event) async {
      double z = event.z;
      DateTime now = DateTime.now();

      // Only log if spike is significant
      if (z < -8 || z > 12) {
        logger.d('Z: $z');
      }
    });
  }
}
```

Figure 4. 12: `Accelerometer_listener.dart` code which use GPS and accelerometer sensors to identify pothole.

The system then uses the geolocator package to retrieve the current GPS position and vehicle speed after detecting a spike that meets the threshold and the cooldown period has elapsed. For ease of interpretation, this speed—which was originally measured in meters per second—is translated to kilometers per hour. The inclusion of speed-based filtering is a significant improvement in this code; if the speed exceeds 20 km/h, the system will only record the incident as a pothole. This filtering aids in distinguishing between potholes and typical road bumps, which are typically seen in low-speed locations like parking lots or school zones. A log notice stating that it was probably a bump is

displayed if the speed is below the threshold, and the detection is ignored. The `detectAndLogPothole()` method, which records the pothole position in the system log, is invoked if all requirements are satisfied. The system also prints comprehensive debug messages in the terminal, including spike values, speed, and any GPS retrieval problems, using the logger package. The illustration is shown in Figure 4.13 below.

```
// Detect pothole condition
if (z < zNegativeSpike &&
    now.difference(_lastDetection).inSeconds > cooldownSeconds) {
    _lastDetection = now;

    try {
        Position pos = await Geolocator.getCurrentPosition();
        double speed = pos.speed * 3.6; // m/s → km/h

        if (speed > 20) {
            logger.i('✅ Pothole detected! Z = $z | Speed = ${speed.toStringAsFixed(1)} km/h');
            await detectAndLogPothole();
        } else {
            logger.i('🟡 Ignored (likely bump). Z = $z | Speed = ${speed.toStringAsFixed(1)} km/h');
        }
    } catch (e) {
        logger.e('❌ Error getting GPS position: $e');
    }
}
});
```

Figure 4. 13: Use the `detectAndLogPothole()` function to record the pothole if the speed exceeds 20 km/h.

A singleton approach is utilized to enable this system from the main entry point of the application, where the `startAccelerometerListener()` function is used to initialize and start the `AccelerometerListener` instance. By calling this function from `main.dart`, the background listener is guaranteed to start running as soon as the application launches, even before the user navigates to any particular screen. Overall, this solution greatly improves detection accuracy and addresses the comments provided during the project review by enabling the app to intelligently detect and distinguish potholes from bumps using a combination of motion and position data.

4.4 Chapter Summary

This chapter gave a thorough rundown of the RePothole mobile application's implementation process, explaining how different system components were created and combined to provide the desired functionality. A predetermined limit and cooldown mechanism were used to minimize false positives, and the accelerometer sensor was used to identify sudden vibration spikes that might be signs of potholes. The GPS module recorded the precise location of detection to geotag each pothole incident. To guarantee data continuity even when offline, each detection was thereafter recorded as text files in local storage. Important details, including locations, timestamps, and sync status, were contained in these logs. After network access was established, a Netlify Function was used to upload unsynced logs to a cloud backend via HTTP requests. The Neon PostgreSQL database, where all detection data was securely saved for further analysis, was accessed using this serverless method. Users may interact with the map, analyze pothole data, and mark repaired potholes as fixed thanks to the mobile app's user-friendly interface. To emphasize significant elements of the implementation, especially in the areas of local data logging, cloud synchronization, and user interface interaction, a few crucial code snippets were incorporated throughout the chapter. The development method produced a fully functional application while keeping to the system's structural plan.

CHAPTER 5: TESTING AND EVALUATION

5.1 Introduction

The RePothole application's evaluation and testing phase is covered in this chapter. Validating the application's usability and functionality ensured it achieved its objectives, following the completion of the system implementation outlined in Chapter 4. The primary objective of this chapter is to illustrate the system's functional and non-functional testing procedures and how the outcomes align with the system's requirements. Simulations and controlled situations were used to test system behavior, and user-based evaluation was used to gauge the application's usefulness and viability.

5.2 Functional Testing

To make sure the RePothole application's functionalities operated properly, functional testing was done. Every part of the system, such as the location services, file logging, accelerometer sensor, and data synchronization, was evaluated separately and in sequence. Verifying that the system accurately identifies potholes, logs them with precise data, and displays this data on the user interface was the goal. All tests were conducted via USB debugging on a Samsung Galaxy Tab A8 device, and logs were examined using the Flutter debug console and Android Studio.

5.2.1 Test Cases

1. Sensor Integration and Detection

1.1 Pothole Detection Trigger

- This test case determines if a strong vibration simulated by shaking the device is effective in initiating pothole detection.

1.2 Cooldown Mechanism

- This test makes sure that the 3-second cooldown delay stops continuous shaking or numerous detections from the same pothole.

2. GPS and Location Tracking

2.1 Location Retrieval- GPS Enabled

- When location services are enabled, this test case determines whether GPS coordinates are recorded when a pothole is found.

2.2 Location Handling- GPS Enabled

- This test case confirms that the application does not crash or record incorrect data, and it properly manages GPS outages.

3. File Logging and Storage

3.1 Save Log to Internal Storage (pothole_log.txt)

- This test case makes sure that the timestamp and UNSYNCED condition are accurately recorded in the internal file along with the discovered pothole data.

3.2 Save Log to Server File (debug_log.txt)

- For synchronizing purposes, this test makes sure that every entry in the XAMPP directory is duplicated.

4. Sync and Upload Functionality

4.1 Automatic Upload to Neon Database via Netlify

- This test confirms that the application properly detects unsynced pothole records and uses the Netlify Function to upload them to the Neon cloud database when an internet connection is available. It demonstrates the system's capacity to autonomously and independently synchronize local data with the backend.

4.2 Update Sync Status After Upload

- This test makes sure that the status of a pothole log entry is changed from UNSYNCED to SYNCED in the local debug_log.txt file after it has been successfully added to the Neon database using the Netlify Function. Only once the Netlify Function successfully receives an HTTP 200 response from the Neon database does the status update take place. This effectively prevents duplicate submissions by ensuring that only confirmed entries are indicated as synced and confirming that the program tracks the data upload process appropriately.

4.3 Pothole vs. Bump Detection Test Cases

- This test makes sure that the accelerometer sensor detect only pothole and not bump on the road.

5. User Interface (UI)

5.1 Mobile App UI

5.1.1 Detection Notification

- This test scenario makes sure that the user sees a pop-up notification or a real-time detection indicator when the accelerometer detects a pothole. Verifying that the system is actively tracking road conditions requires this.

5.1.2 GPS Permission Prompt

- This test determines whether, if the location permission has not yet been granted, the application will automatically ask the user to grant GPS access. This occurs when users first use the software or when they reinstall it.

5.1.3 Start and Stop Button

- This test case confirms that the app's "Start" and "Stop" buttons work as intended. When the user presses the “Start “button, it will automatically change to “Stop”.

5.2 Website UI

5.2.1 View Potholes on Map

- This test case determines whether a map interface appropriately displays the identified potholes. Every pothole needs to show up as a marker with the precise location recorded by the mobile app.

5.2.2 Marker Color Indicator (Status Update)

- The dynamic marker color scheme is validated by this test. The original appearance of pothole markings should be red, signifying that the pothole has not been fixed. The marker should turn green to indicate resolution once the administrator or user checks the box confirming the pothole has been repaired.

5.2.3 Filter by Date,Place and Status

- This test makes sure the website's filter panel functions properly. It should be possible for users to quickly locate individual pothole reports by filtering the map view or pothole list by date range location and status.

5.2.4 Marker Info Window

- This test case verifies that when a pothole marker is clicked, an information window displaying the pothole ID, date, and time is displayed. This enables users or authorities to see the report's whole context.

5.2.5 Checkbox (Mark as Fixed)

- This test case makes sure that the marker colour on the map view changes from red to green and that the database is updated with the pothole report status when the user selects the "Pothole Fixed" checkbox.

Table 5. 1: Test Case- Pothole Detection Trigger

Test ID	TC1				
Module Name	Sensor Integration				
Test Title	Pothole Detection Trigger				
Descriptions	This test case verifies whether pothole detection can be successfully started by simulating a strong vibration by shaking the device.				
Pre-Descriptions	The user needs to shake the device to make the pothole appear.				
Pre-Condition	The RePothole mobile application is running, and the accelerometer listener is started.				
Dependencies	Accelerometer_listener.dart , sensors_plus				
Step	Test Procedures	Expected Result	Actual Result	Status	Notes
1	Use an appropriate amount of force to shake the device.	It sets off a pothole detection log or alarm.	The detection was carried out and recorded.	PASS	Confirmed detection using the terminal in VS Code Studio
Post-Condition	The log file is updated with a new item that includes the timestamp and UNSYNCED condition.				

Table 5. 2: Test Case- Cooldown Mechanism

Test ID	TC2				
Module Name	Sensor Integration				
Test Title	Cooldown Mechanism				
Descriptions	This test demonstrates that recurrent shaking or multiple detections from the same pothole are prevented by the 3-second cooldown period.				
Pre-Descriptions	The user shakes the device quickly several times within 3 seconds.				
Pre-Condition	Pothole detection is active, and the app is operating.				
Dependencies	Accelerometer_listener.dart				
Step	Test Procedures	Expected Result	Actual Result	Status	Notes
1	Shake the device one time	The initial detection is made and recorded.	A log entry was made.	PASS	None
2	Within three seconds, shake the device once more.	There is no new detection.	No second log entry	PASS	Cooldown is functioning as planned.
3	After more than three seconds, shake once more.	A new detection is made.	A second log entry was made.	PASS	None

Post-Condition	There should only be two entries. The first is from the shake, and the second is from the cooldown.
-----------------------	---

Table 5. 3: Test Case- Location Retrieval (GPS Enabled)

Test ID	TC3				
Module Name	GPS and Location Tracking				
Test Title	Location Retrieval (GPS Enabled)				
Descriptions	When location services are enabled and a pothole is detected, this test determines if GPS coordinates are recorded.				
Pre-Descriptions	Permissions are granted, and GPS is activated.				
Pre-Condition	Both the GPS and the mobile application are functional.				
Dependencies	Geolocator, location_service.dart				
Step	Test Procedures	Expected Result	Actual Result	Status	Notes

1	The user shakes the device to trigger the detection	The log includes both latitude and longitude.	GPS coordinates are accurately stored.	PASS	Confirmed using the log file
Post-Condition	Valid latitude and longitude values are included in the log entry.				

Table 5. 4: Test Case- Location Handling (GPS Disabled)

Test ID	TC4				
Module Name	GPS and Location Tracking				
Test Title	Location Handling (GPS Disabled)				
Descriptions	This test confirms that the application does not crash or save incorrect data when GPS is switched off.				
Pre-Descriptions	The GPS is disabled in the device's settings before the RePothole mobile application is launched.				
Pre-Condition	The RePothole mobile application has been launched.				
Dependencies	Geolocator, error handling in location_service.dart				
Step	Test Procedures	Expected Result	Actual Result	Status	Notes

1	To simulate detection, shake the device.	The app with care handles GPS errors.	No crash and location error resolved.	PASS	If not allowed, a GPS prompt appears.
2	Verify the log entries.	The entry was stored in a default or missing location.	The log displays a null or empty location.	PASS	can later be marked as incomplete.
Post-Condition		GPS coordinates or markings as unknown are not included in the log file, which documents the pothole incident.			

Table 5. 5: Test Case- Save Log to Internal Storage (pothole_log.txt)

Test ID	TC5
Module Name	File Logging
Test Title	Save Log to Internal Storage
Descriptions	Ensure that every pothole detection is recorded in pothole_log.txt, including a timestamp and an UNSYNCED status.
Pre-Descriptions	While the mobile application is running, detection is triggered.

Pre-Condition	The application is authorized to access internal storage.				
Dependencies	file_service.dart , path_provider				
Step	Test Procedures	Expected Result	Actual Result	Status	Notes
1	Shake the gadget to resemble detection.	A fresh log entry has been added to pothole_log.txt.	Time-stamped entry that is UNSYNCED	PASS	Verified through file examination
Post-Condition	A new entry has been added to the file.				

Table 5. 6: Test Case-Save Log to Server File (debug_log.txt)

Test ID	TC6
Module Name	File Logging
Test Title	Save Log to Server File
Descriptions	Confirms that logs in debug_log.txt are replicated to sync with the XAMPP server.
Pre-Descriptions	The detection happens if the XAMPP path is available.

Pre-Condition	'C:/xampp/htdocs/repothole/debug_log.txt' available.				
Dependencies	file_service.dart, file I/O				
Step	Test Procedures	Expected Result	Actual Result	Status	Notes
1	Detect Potholes	Debug_log.txt contains a duplicate of the same log.	The entry is in the format of an internal log.	PASS	Check manually using Notepad.
Post-Condition	The same UNSYNCED entry is present in the server file.				

Table 5. 7: Test Case- Automatic Upload Neon Database via Netlify

Test ID	TC7
Module Name	Sync and Upload
Test Title	Neon Database Automatic Upload Using Netlify
Descriptions	Verifies that the program recognizes UNSYNCED records accurately and, when internet access is available, uploads them to the Neon database using a Netlify Function. shows how local data can be automatically synchronized with the cloud backend.

Pre-Descriptions	The device being used has an internet connection. There are UNSYNCED entries in debug_log.txt.				
Pre-Condition	The Netlify Function is up and running. The Neon database is accessible online.				
Dependencies	Add-pothole.js, HTTP, Netlify Function, Neon SQL instance				
Step	Test Procedures	Expected Result	Actual Result	Status	Notes
1	Make sure you have an active internet connection when you launch the application.	application that is connected to the internet. The system looks for UNSYNCED items in debug_log.txt.	application that is connected to the internet. The system looks for UNSYNCED items in debug_log.txt.	PASS	The logs are correctly read.
2	To POST data to Neon, the application uses the Netlify Function.	UNSYNCED logs are included with the HTTP POST request that is submitted to the Netlify backend.	UNSYNCED logs are included with the HTTP POST request that is submitted to the Netlify backend.	PASS	A network request appeared in the development tools.

3	Data is inserted into Neon using SQL by the Netlify function.	If insertion is successful, Neon produces a 200 OK HTTP response.	If insertion is successful, Neon produces a 200 OK HTTP response.	PASS	confirmed through the Neon dashboard
Post-Condition	The map either shows only the filtered potholes or all of them return after being reset.				

Table 5. 8: Update Sync Status After Upload to Neon

Test ID	TC8
Module Name	Sync and upload
Test Title	Update Sync Status After Upload to Neon
Descriptions	A successful upload to the Neon database is required before pothole logs in debug_log.txt are designated as SYNCED. To avoid duplicate entries, status updates only take place following a 200 OK response from the Netlify Function.

Pre-Descriptions	200 OK was successfully returned by the Netlify function. The upload has been verified as complete.				
Pre-Condition	The uploaded entries are still listed as UNSYNCED in debug_log.txt.				
Dependencies	file_service.dart, file I/O logic, response handler				
Step	Test Procedures	Expected Result	Actual Result	Status	Notes
1	Set up upload logic to pass on details to Neon.	Uploading takes place using the Netlify Function.	Uploading takes place using the Netlify Function.	PASS	Hit the function endpoint
2	Await the Netlify backend's 200 OK answer.	For those items, the status in debug_log.txt is changed to SYNCED.	For those items, the status in debug_log.txt is changed to SYNCED.	PASS	The file displays the accurate updated status.
3	Restart the upload logic.	Past uploads are not taken into consideration.	Past uploads are not taken into consideration.	PASS	No duplicate was sent.

Post-Condition	Only newly UNSYNCED logs, if any, are now included in debug_log.txt.				

Table 5.9: Test Case- Pothole vs. Bump Detection

Test ID	TC9				
Module Name	Accelerometer Listener + Pothole Detection.				
Test Title	Speed-Based Filtering to Differentiate Between Potholes and Bumps				
Descriptions	Using Z-axis spikes and vehicle speed threshold logic, this test confirms that the system accurately detects potholes and speed bumps.				
Pre-Descriptions	The car has a mounted gadget and an active listener app.				
Pre-Condition	Motion detection is turned on, GPS is enabled, and location authorization is provided.				
Dependencies	Geolocator, sensors_plus, pothole_logger.dart				
Step	Test Procedures	Expected Result	Actual Result	Status	Notes

1	At 35 km/h, drive over a pothole with a steep dip ($Z < -15$).	.GPS-logged pothole	.Pothole logged	PASS	Console and database verification
2	Use $Z \approx -13$ to drive over speed bumps at a slow pace (10 km/h).	Low speed causes detection to be ignored.	No detection	PASS	As expected
3	At 70 km/h, hit a pothole ($Z = -16.5$)	Pothole logged	Pothole logged	PASS	Accurate detection
4	Roll across a rough surface at a leisurely pace of 8 km/h ($Z = -14.0$).	Low speed, no detection	No detection	PASS	As expected
5	Sudden brake ($Z = -7.0$) with no actual obstacle	No detection (Z below threshold)	No detection	PASS	-
6	hit a bump with a steep Z spike ($Z = -14.5$) and an abnormally high speed (30 km/h).	May trigger detection	Pothole detected	Acceptable	Acceptable false positive

7	Travel 50 km/h on a serene, level road.	No detection	No detection	PASS	No noise or false detection

Table 5. 9: Test Case- Detection Notification

Test ID	TC9
Module Name	Mobile App UI
Test Title	Detection Notification
Descriptions	guarantees that when a pothole is found, a notification will show.

Pre-Descriptions	The listener is active, and the app is running.				
Pre-Condition	The listener for the accelerometer starts.				
Dependencies	accelerometer_listener.dart, UI alert logic				
Step	Test Procedures	Expected Result	Actual Result	Status	Notes
1	Shake device	A pop-up or alert is shown.	The screen displays an alert with the position and timestamp of the plotted pothole.	PASS	Real-time user notification
Post-Condition	The user sees an alert verifying the detection.				

Table 5. 10: Test Case- Start and Stop Button

Test ID	TC10				
Module Name	Mobile App UI				
Test Title	Start and Stop Button				
Descriptions	Confirms that the Start/Stop toggle works properly to activate detection.				
Pre-Descriptions	Buttons appeared as the mobile app launched.				
Pre-Condition	The accelerometer is not operating yet.				
Dependencies	UI logic, state management				
Step	Test Procedures	Expected Result	Actual Result	Status	Notes
1	The user presses the 'Start' button.	When the listener is turned on, the button label switches to Stop.	Switch the button to Stop.	PASS	The detection process starts.
2	The user presses the 'Stop' button	The label turns to Start once the listener stops.	The Start button was toggled back.	PASS	The detection was stopped.

Post-Condition	The user controls the sensor monitoring.
-----------------------	--

Table 5. 11: Test Case- View Potholes on Map

Test ID	TC11				
Module Name	Website UI				
Test Title	See Potholes on a Map				
Descriptions	Verifies that potholes are shown as coordinate-based markers.				
Pre-Descriptions	Synced data was found in the database.				
Pre-Condition	The map module loads on the website.				
Dependencies	Google Maps, PHP Backend				
Step	Test Procedures	Expected Result	Actual Result	Status	Notes
1	Open the website's map view.	There are markers for every pothole.	Red markers appear on the loaded map.	PASS	Positions correspond to

					logs that have been posted.
Post-Condition	Potholes are shown on a map.				

Table 5. 12: Marker Colour Indicator (Status Update)

Test ID	TC12				
Module Name	Website UI				
Test Title	Indicator of Marker Color				
Descriptions	When a pothole is designated as fixed, the checkmarker turns from red to green.				
Pre-Descriptions	Marker status is kept in the database.				
Pre-Condition	The marker is visible on the map.				
Dependencies	Marker logic and the database's status column				
Step	Test Procedures	Expected Result	Actual Result	Status	Notes

1	Check the "Pothole Fixed" box.	The marker turns green.	The marker color was successfully changed.	PASS	Change is immediately apparent.
Post-Condition	Both visibly and in the database, the pothole has been fixed.				

Table 5. 13: Test Case-Filter by Date and Place

Test ID	TC13				
Module Name	Website UI				
Test Title	Filter by Date, Status, and Place				
Descriptions	Confirms that the filter panel finds potholes that match.				
Pre-Descriptions	There are several entries in the database.				
Pre-Condition	The database and the website are linked.				
Dependencies	Filtering data and query logic in the Neon database.				
Step	Test Procedures	Expected Result	Actual Result	Status	Notes

1	Enter any of the filters. Test: Place: UNIMAS Status: active	Only markers that match are displayed.	Return of correct entries	PASS	Filter logic functions properly.
Post-Condition	With filtered results, the map is updated.				

Table 5. 14: Test Case-Marker Info Window

Test ID	TC14				
Module Name	Website UI				
Test Title	Marker Info Window				
Descriptions	Make sure a pop-up with log details opens when a marker is clicked.				
Pre-Descriptions	There are marks on the map.				
Pre-Condition	On the map screen is the user.				
Dependencies	Database values and the JavaScript pop-up handler				
Step	Test Procedures	Expected Result	Actual Result	Status	Notes

1	Select any pothole marker by clicking on it.	Details are displayed in the info box (Pothole ID, timestamp).	Information is shown correctly.	PASS	contains the pothole ID and status.
Post-Condition	A pothole context assessment is made simpler by the info window.				

Table 5. 15: Test Case- Checkbox (Mark as Fixed)

Test ID	TC15
Module Name	Website UI
Test Title	Checkbox (Mark as Fixed)
Descriptions	Make certain that the marker colour changes and the checkbox updates the database.
Pre-Descriptions	Red indicates a pothole that has not been fixed.
Pre-Condition	A marker with the state "active" is available.
Dependencies	PHP backend, logic for updating markers

Step	Test Procedures	Expected Result	Actual Result	Status	Notes
1	Check the "Pothole Fixed" box.	Updated database. Turn to the green marker.	The user interface is modified to reflect the change.	PASS	confirmed by reloading the map
Post-Condition	Markers turn colour to green.				

5.3 Non-Functional Testing

Non-functional testing concentrates on elements like usability, learnability, user satisfaction, and performance under normal usage scenarios, whereas functional testing makes sure that the RePothole application's essential functions operate as intended. These elements are particularly crucial for mobile apps that depend on user experience and real-time engagement. Usability testing was done for both platforms since RePothole uses a web-based interface to monitor and manage pothole data in addition to a mobile app for pothole detection.

A pre-test and post-test methodology was used to assess usability. Using a series of tasks, this approach enabled testers to investigate the system and then report back on their experiences. Feedback on the application's logical flow, responsiveness, visual clarity, and ease of use, particularly for people with little to no technical background, was the aim of this study.

5.3.1 Pre-test and Post-test

A. On-Site and Road Testing

The RePothole mobile and web-based system's usability and effectiveness were assessed through live road testing in actual vehicle situations, as well as pre- and post-test evaluations among a group of volunteer testers from UNIMAS students. Participants in the user-based testing were between the ages of 20 and 30, and their knowledge of GPS-based systems and mobile applications varied. Nine people in all, three men and six women, participated in the structured usability test.

Figuring out how user-friendly the RePothole system was and whether it successfully identified and documented potholes in actual use were the primary goals. Both the web-based interface for storing and viewing pothole reports and the mobile application for detecting and recording potholes were evaluated.

In the initial phase of the test, a car was used for actual road testing on the UNIMAS campus and nearby public roadways. On a Samsung Galaxy Tab A8, the RePothole mobile application came pre-installed as an .apk file. To replicate its intended use while driving, the tablet was securely mounted on the vehicle's dashboard. During the test, the application operated without a USB connection to a laptop because it was already compiled and installed.

The application utilized the device's built-in accelerometer to actively monitor road conditions while the vehicle traveled at speeds ranging from 20 to 60 km/h. The device immediately recorded a log entry with the timing and GPS locations of the detection when it identified a pothole based on vibration strength. Every log entry was stored for synchronization's sake in a server-accessible file (debug_log.txt) as well as the device's internal storage (pothole_log.txt). Participants examined the app interface during the test to make sure that data was appropriately recorded for every incident and that real-time detection notifications were shown. Participants were then given instructions on how to use the RePothole website. They had to do the following:

- Looking at the map interface with the pothole markers displayed.
- To find entries by date and place, use the filter option.
- To access pothole details like time, position, and condition, click on each marker separately.
- The marking changed from red to green when the "Pothole Fixed" checkbox was checked, simulating a pothole being fixed.
- The Pothole marker is removed from the map when the user clicks the "Pothole Fixed" checkbox.

Participants received a post-test feedback form after finishing all testing tasks. This evaluated their experience in terms of the app's responsiveness, detection accuracy, map visualization utility, and user interface clarity.

Participant feedback was overwhelmingly positive. Even when driving, most users considered the app to be user-friendly and intuitive. Simple control over the sensor was made possible by the Start/Stop buttons, and active monitoring was verified by the notification popups. Each detection was recorded in the pothole log file, and GPS location reporting was precise. Users valued the ability to update their status and see the locations of potholes visually on the website. A few minor recommendations were to improve log filtering options, add a real-time map to the program, and show a sync indicator in the user interface.

To sum up, the pre-test indicated that users were interested in and cautious of how the app would work, and the post-test verified that RePothole fulfilled usability and performance standards. The system's design and preparedness for future deployment or integration with city

reporting systems were validated by the successful completion of real road testing, which showed the system's applicability in the real world.

B. Online Testing

Nine volunteers participated in an online testing session to further assess the RePothole mobile application's functionality and usability. Participants were able to download and install the app on their own Android cell phones

after it was built into an Android Package (.apk) file and shared via a Google Drive link. Outside of the scope of controlled onsite testing, this approach offered the chance to collect input from users using the app in a variety of real-world settings.

All participants were required to fill out a pre-test survey before using the application, which evaluated their knowledge of GPS-based services, road condition reporting apps, and their expectations regarding the potential functionality of pothole detection using mobile sensors. A post-test survey was given after the app was installed and used to get input on the user's experience, usability of the UI, responsiveness, detection accuracy, and ease of use.

The volunteers received instructions on how to do crucial activities such turning on and off the pothole detection, mimicking detection events by shaking the device or driving on uneven ground, looking at the pothole logs, and paying attention to the application's real-time notifications. In addition, participants were urged to examine pothole markers on the map and experiment with simple features like filtering and labeling potholes as fixed on the RePothole website.

5.4 Analysis

A. Pre-test Feedback

Participants in the pre-test poll were asked to rate their familiarity with mobile applications that make use of built-in phone sensors, like GPS and accelerometers, which are essential to the RePothole app's functionality. According to the findings, 22.2% of respondents had no prior experience with such applications, 55.6% were somewhat familiar, and 22.2% were extremely familiar. According to these results, most users had only a limited understanding of sensor-based systems, which emphasizes how crucial it is to make RePothole easy to use and intuitive, especially for non-technical people. A little learning curve was crucial because the main goal of the software is to apply motion and location sensors to detect potholes without the need for complex user input. The pre-test data also offered helpful background information for assessing whether the behavior and interface of the app could successfully assist less-experienced users in grasping its operation through practical use. Figure 5.1 below show the percentage of familiarity of respondents to the mobile application with sensors.

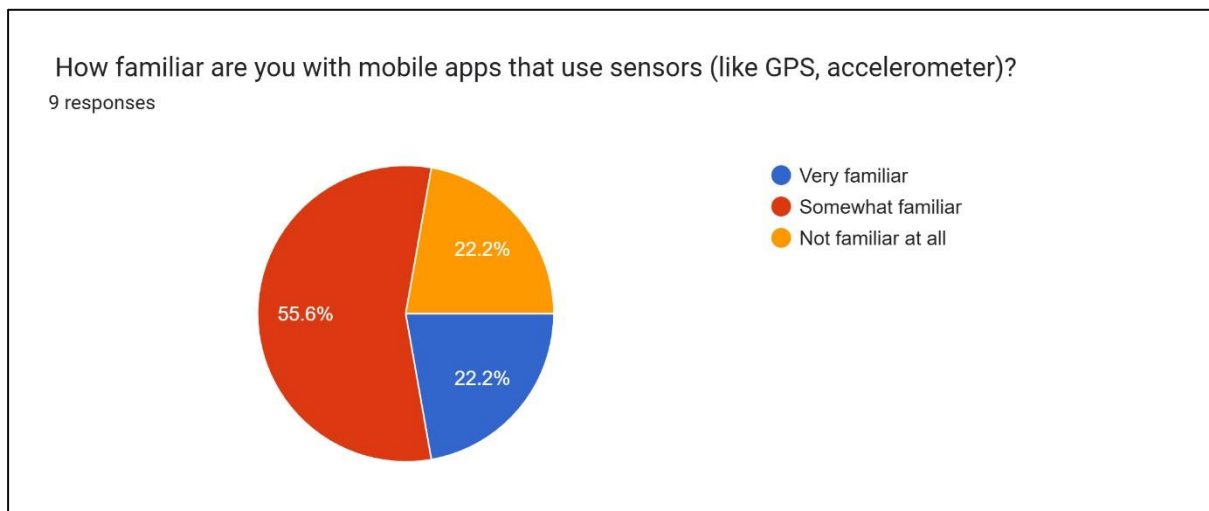


Figure 5. 1: Pie Chart- Familiarity of respondents to the mobile application that use sensors.

Participants were asked to score their level of confidence in a smartphone's capacity to identify potholes using vibration data from integrated sensors, like the accelerometer, in another pre-test question. The findings showed that just 11.1% of respondents were extremely confident in the viability of this strategy, compared to 44.4% who were not and 44.4% who were somewhat sure. These answers reveal a general lack of confidence or doubt among users regarding the accuracy of a mobile device's ability to identify anomalies in the road without the need for additional equipment. This is particularly important to the main goal of the RePothole

application, which is to show that real-world pothole identification can be accomplished efficiently using mobile sensor data. The application must demonstrate its dependability by precise logging and clear detection feedback, as highlighted by the low to moderate confidence levels seen before testing. This information provided a useful starting point for contrasting participants' opinions before and after seeing the system in operation. Figure 5.2 below shows the level of confidence of respondents in using a phone that uses vibration to detect potholes.

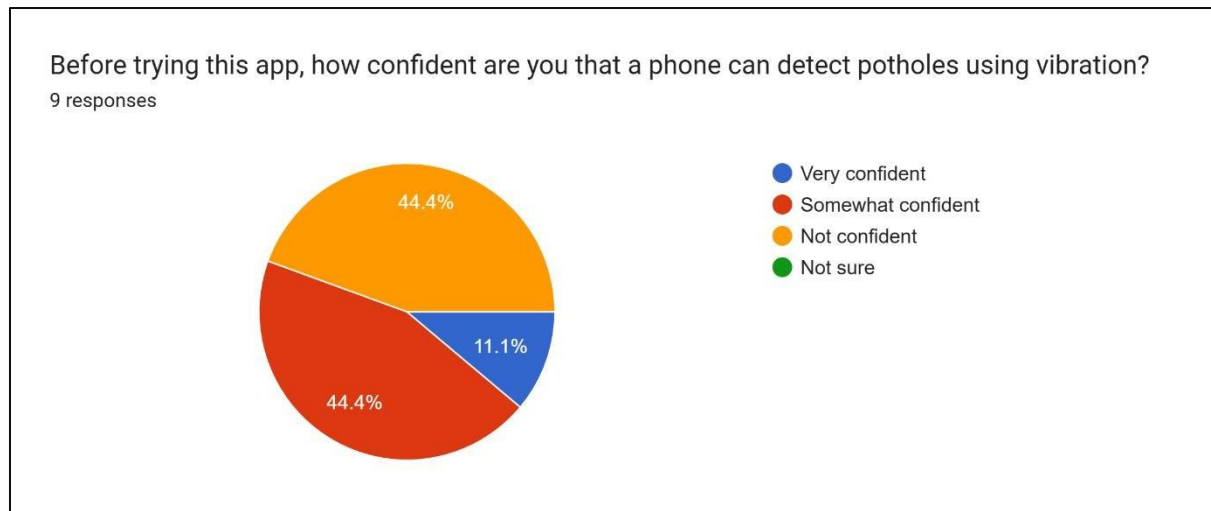


Figure 5. 2: Pie Chart- Level of Confidence Respondents for Mobile Phone to Detect Pothole.

Through a variety of answers available, participants were also asked what they hoped to accomplish using the RePothole app. "Detect potholes" (66.7%) and "view potholes on a map" (66.7%) were the most chosen expectations, with "record GPS location" (55.6%) coming in second. According to these answers, users mainly expected the application to be a tool for geolocation-based reporting and real-time road condition monitoring. This is in line with RePothole's stated goals, which include using the phone's accelerometer to find potholes, automatically tagging each find with GPS information, and using the web portal to display the data on a centralized map. The findings also imply that users place a high value on the detection capacity as well as the spatial viewing and comprehension of the data, highlighting the significance of incorporating precise GPS logging and an intuitive map-based user interface into the system. Figure 5.3 below shows the expectations of the respondent when they use the mobile application.

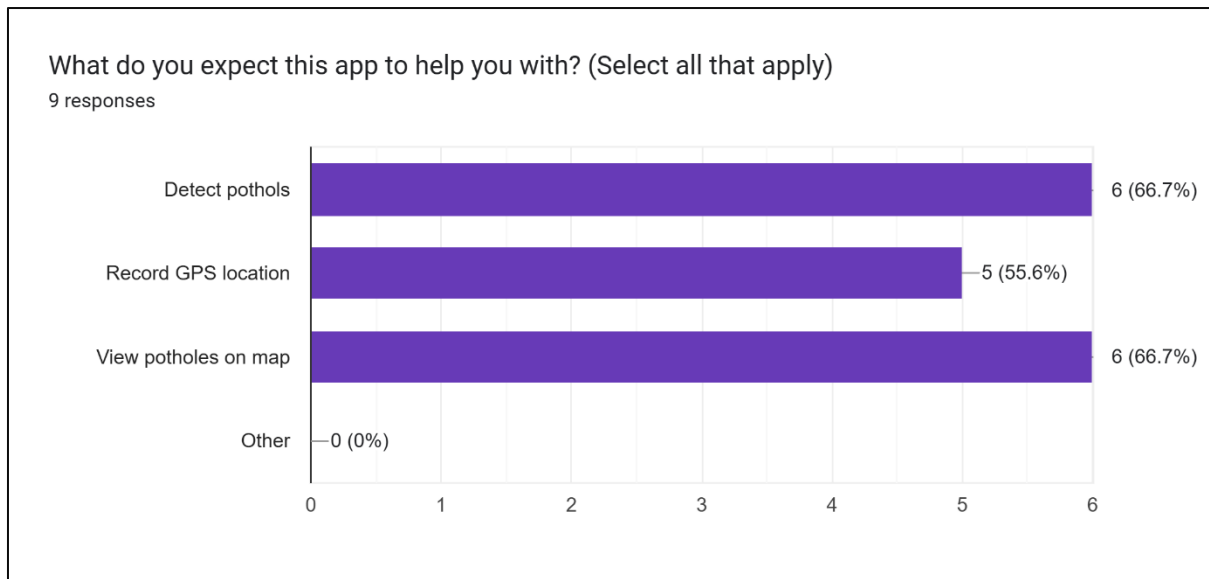


Figure 5. 3: Expectation of Respondent to the Repothole Mobile Application

B. Post-Test Feedback.

To get feedback on their experience with the RePothole mobile application and, if applicable, the related via the internet portal, users were asked to fill out a post-test survey after the testing session. Following user interaction with the system, either by testing the app while driving or by simulating pothole detection through device movement, this part collects the responses that were received. Usability, navigational ease, detection alarm clarity, GPS logging efficacy, and the utility of features like the pothole log and map display were all assessed by the post-test questions. In order to evaluate how successfully the application fulfilled user expectations in real-world and self-guided testing situations, the survey's input was a useful indicator of the system's strengths and places for improvement.

100% of participants selected "Yes" in response to the post-test question, "Did the app detect potholes when you shook the device or drove over bumpy roads?" This suggests that the RePothole program effectively handled both simulated and actual pothole scenarios. This outcome amply validates the dependability of the system's accelerometer-based detection logic. It appears that the specified sensitivity limit and cooldown method functioned as planned, enabling the app to identify important vibrations while reducing false positives, based on the consistency of detection across all nine users. This result confirms one of the project's main goals which is to demonstrate that a mobile device can efficiently use motion analysis to detect pothole events without the need for additional hardware. Figure 5.4 below shows the percentage of

respondent who successfully detects potholes when they shake the device or drive over a bumpy road.

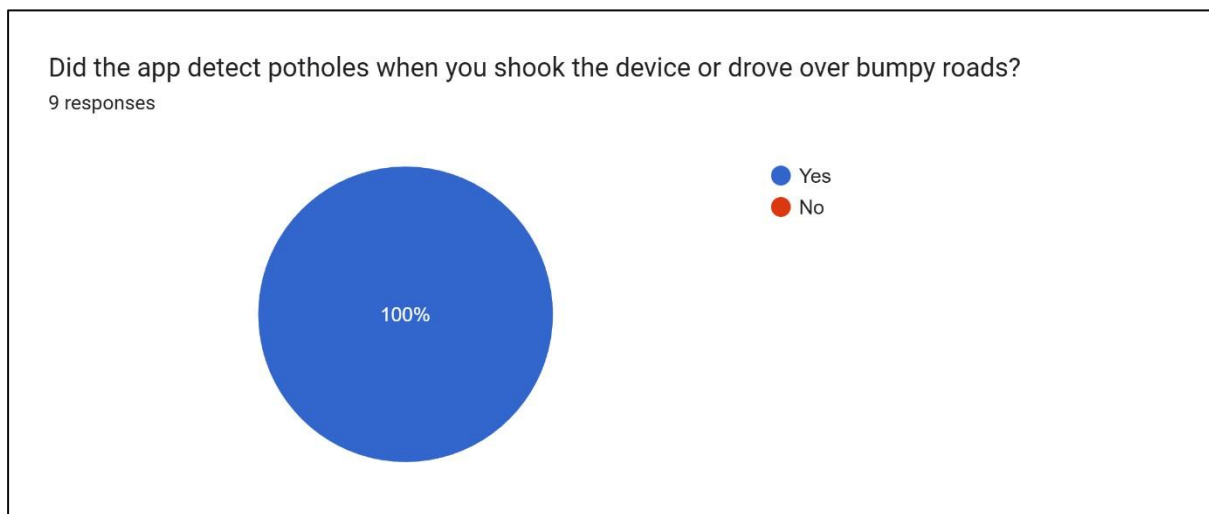


Figure 5. 4: Pie Chart- Potholes Detection When Respdent Shake the Device or Hit Over Bumps on the Road.

11.1% of participants said they did not access the RePothole website to examine detected potholes on the map, compared to 88.9% who said they did. This high rate of engagement suggests that most users experimented with the web-based interface to view the logged pothole data in addition to testing the mobile application. Users were better able to comprehend the significance and effect of real-time logging, as well as how the system links front-end detection with backend visualization, thanks in large part to the capability of viewing pothole marks on a map. The tiny proportion of people who did not visit the website might be the result of time restraints or a lack of experience with online navigation. However, the high response rate confirms that the map-based site improved the user experience overall and effectively complimented the mobile app. Figure 5.5 below shows the percentage of respondents who use the website to view the potholes.

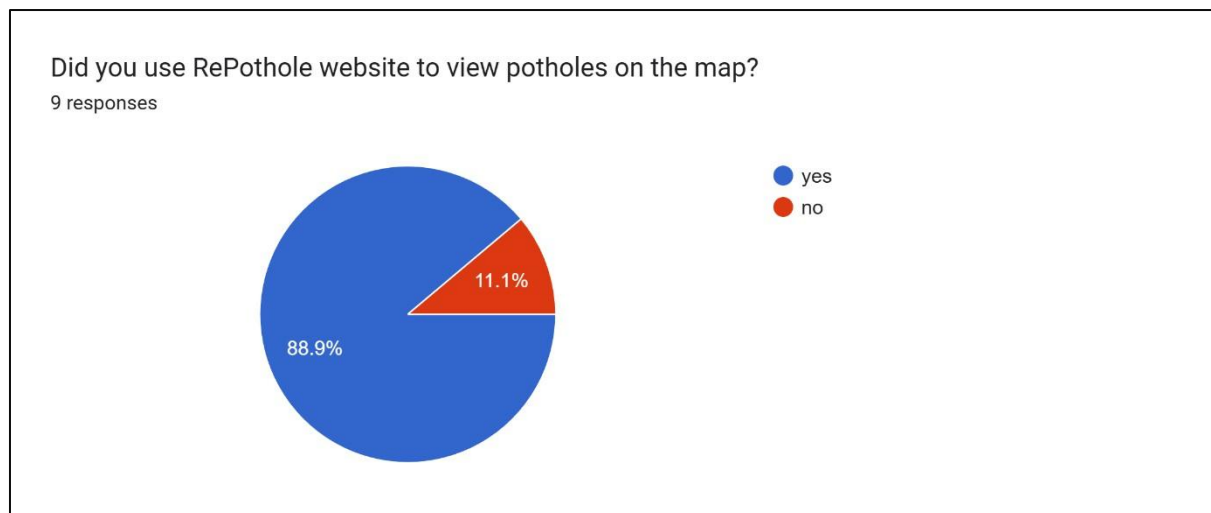


Figure 5. 5: Pie Chart- Use Website to View the Plotted Potholes

During their testing session, participants were asked to identify which components of the RePothole website worked properly. The fundamental features of map display and data filtering were fully functioning, as all respondents (100%) testified to the accuracy of pothole markers' appearance on the map and their ability to filter pothole records by date or location. Furthermore, the function that labels potholes as fixed and updates their visual state on the map was successfully accessed by 88.9% of users, who also reported being able to change the marker status using the checkbox. In the meantime, 77.8% of participants could see the information popups that show specific information like the location and timestamp when they click on a marker. Interestingly, none of the respondents checked the box for "Did not try any features," indicating that every user looked around and engaged with the website in some way. These findings demonstrate how well the online portal supported the mobile app by providing a useful, intuitive interface for managing and reviewing reported pothole data. Figure 5.6 below shows the functionality of the mobile application that the respondents able to experienced.

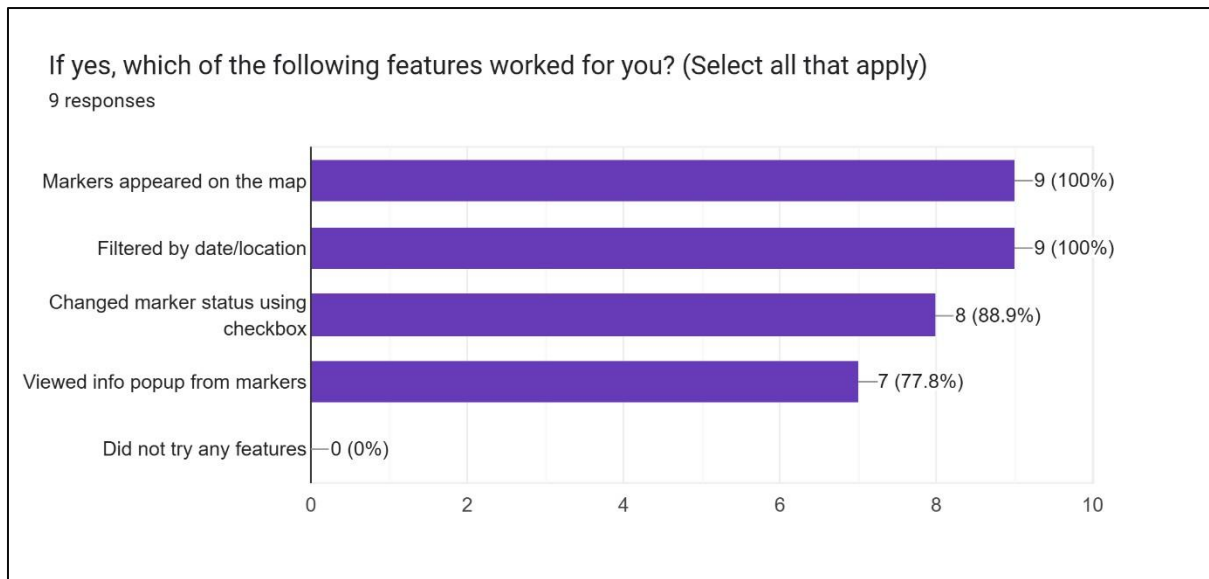


Figure 5. 6: Functionality of the Mobile Application that the Respondents Experienced

When asked about the accuracy of the pothole detecting and logging components of the RePothole application, 55.6% of participants said the system was "accurate," while 44.4% said it was "very accurate." These findings show that users are quite satisfied with the system's capacity to consistently identify traffic disruptions and record them with the proper location and time stamp. The lack of negative answers such as "inaccurate" or "not sure" indicates that during both simulated and actual road testing, the program continuously fulfilled user expectations. This encouraging review validates the conclusion that RePothole provides accurate and useful detection findings in practical situations and shows how well the vibration sensitivity settings and GPS integration work in the app. Figure 5.7 below shows the level of accuracy of the pothole detection experienced by the respondents.

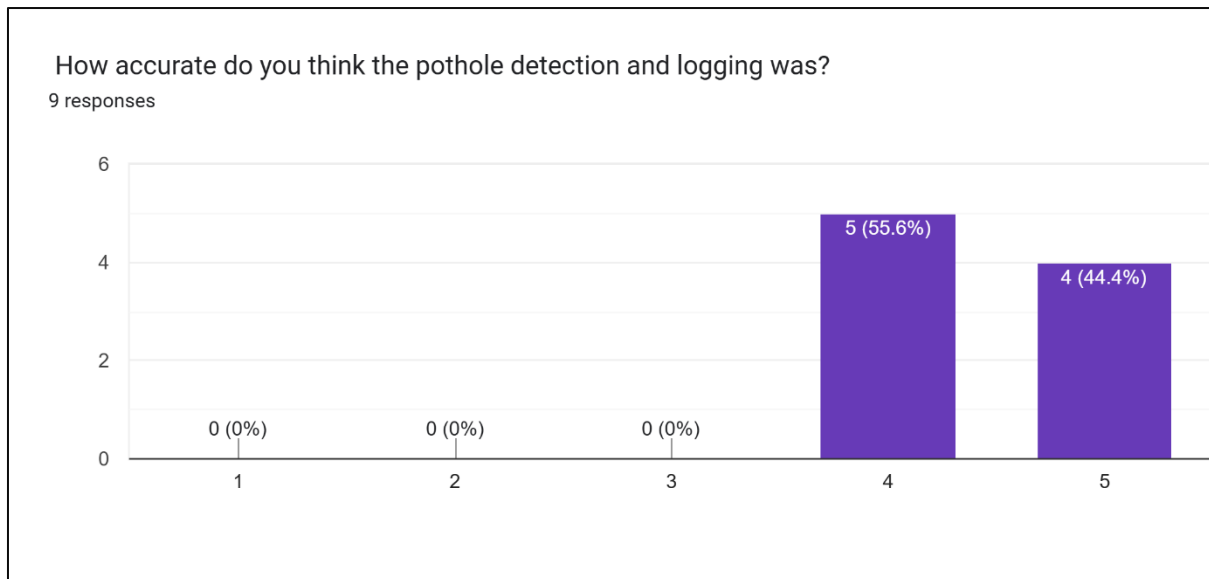


Figure 5. 7: The Accuracy of the Detection and Logging of the Potholes.

Participants offered a wide range of insightful recommendations targeted at improving system functioning and usability in answer to the open-ended inquiry about possible enhancements for the RePothole application or website. Some respondents made specific ideas, while one said no changes were needed. In order to give users and administrators an overview of data at a glance, a number of contributors proposed adding a dashboard that would display summarized statistics, such as the total number of potholes found or places with frequent reports. Incorporating a live map display into the mobile app was another popular recommendation, which would allow users to see nearby potholes without visiting the website. Additionally observed was the requirement for audio feedback during detection events, which might raise driver awareness without the need for visual focus. One participant suggested that confirmation prompts be added before a pothole is marked as patched on the website to ensure data integrity and avoid unintentional updates. To guarantee that newly synchronized data shows in real time, further recommendations included allowing users to manually enter pothole entries and implementing automated map refreshing on the online interface. These recommendations provide useful information that can direct system updates in the future and help in increasing the system's usefulness for both regular users and administrative users. Figure 5.8 below shows the suggestion and improvements from respondents towards the mobile application.

What improvements would you suggest for the app or website?

9 responses

none

Include a dashboard

Provide audio

Add confirmation prompts before marking a pothole as fixed on the website

Add a live map view inside the mobile app

Allow users to manually add a pothole

Automatically refresh the map view on the website when new pothole data is uploaded,

Figure 5. 8: Suggestions and Improvements Towards RePothole Mobile Application

5.5 Chapter Summary

The RePothole mobile application's testing and assessment procedure was covered in length in this chapter, with an emphasis on both functional and non-functional elements. Functional testing confirmed that important features like GPS location tracking, local file logging, server synchronization, accelerometer-based pothole detection, and user interface elements all functioned as planned according to specified test cases. Selected individuals who used the application during actual road testing around the UNIMAS campus participated in pre-test and post-test non-functional testing. The application successfully identified potholes and recorded each occurrence with precise timestamps and GPS data when it was installed as a compiled program ‘.apk’ on a tablet placed in a car. Participants offered feedback on the app's usability after interacting with its UI, which included features like Start/Stop detection control. To support the overall goal of creating a dependable pothole-detecting solution, the post-test responses verified that the system was responsive, user-friendly, and successful in real-world situations.

CHAPTER 6: CONCLUSION AND FUTURE WORK

6.1 Introduction

The development and evaluation of the RePothole mobile application, which utilizes mobile device sensors to identify potholes in real-time, are concluded in this chapter. The chapter provides an overview of the goals achieved during the project, the restrictions and limitations encountered during development and testing, and potential future additions and modifications that could enhance the efficacy and usability of the application.

6.2 Objectives Achieved

All the main goals listed in Chapter 1 have been effectively accomplished by the project's conclusion. The following are the specifics of these accomplishments:

1. It used the smartphone's included accelerometer and GPS data to automatically identify potholes based on vibrations in the road. Using an established threshold, the device was able to differentiate pothole-like vibrations from typical road activity and precisely record each detection.
2. Created an application that requires minimal human input while driving, just a basic Start/Stop toggle, and handles all detections and GPS data gathering automatically in the background. This prevented the driver from becoming distracted and guaranteed precise pothole location tracking.
3. Each detection is automatically copied to a server-based file (debug_log.txt) and kept locally in pothole_log.txt as part of a data logging and synchronization process. Both the timestamp and the geolocation are included in these entries.
4. Created and implemented a web-based platform that uses checkboxes to allow users and authorities to examine, filter, and update the status of reported potholes on an interactive digital map. This immediately advances the goal of providing actionable and easily available pothole data for road maintenance and monitoring.

6.3 Limitations and Constraints

Despite the RePothole mechanism's successful creation, the project ran into several issues. The reliance on a steady GPS signal is one significant drawback, which may compromise accuracy in places with poor coverage, like underpasses or crowded urban areas. Another limitation is that the accuracy of pothole identification may differ depending on the type of vehicle and where the device

is placed. This is because the sensor's findings may be affected by movement or road noise that is not linked to potholes. Furthermore, even though the app syncs automatically, users are not yet able to see real-time status indicators in the mobile interface that would notify them when entries are successfully uploaded. Additionally, the mobile app lacks a built-in map, so users must access the website independently to view data. The software was tested with a relatively small number of users due to time and resource restrictions, which might not accurately reflect wider public usage. Larger-scale deployment would assist in confirming efficacy across more varied environments and user behaviors, even though testing results were encouraging.

6.4 Future Work

The RePothole setup can be improved in the future by adding several new features that will enhance its usability, scalability, and functionality. Adding a live map display to the mobile app would be a significant improvement that would enable users to view pothole reports in their area without having to navigate to the web platform. Adding a sync status signal would also assist users in confirming that the upload of their data was successful.

In situations when sensor detection can overlook minor road problems, users can manually add potholes or attach images to the app's manual reporting options. To facilitate real-time notifications and response coordination with road maintenance authorities, the system can also be integrated with local reporting systems. Features like email notification systems for authorities, data analytics dashboards, and auto-refresh of new data can be added to the website. Finally, expanding public testing and gathering more varied user input would confirm the system's resilience and direct upcoming upgrades.

6.5 Conclusion

The RePothole application effectively illustrates the viability of utilizing mobile sensors to identify and report potholes, according to the development and testing outcomes. By offering a functional system that detects road imperfections, logs precise position data, and presents the information via an easily navigable online interface, the app achieves its primary goals. Positive feedback about usability and detection reliability was received from both online and onsite users. The RePothole system has a great chance to become a useful road monitoring tool that enhances road safety and repair response with additional development.

References

- Atarmizi, W. A. (2024, January 20). *Broken roads, broken lives: Malaysians cry foul for pothole solution* | *Sinar Daily*. Sinar Daily.
<https://www.sinardaily.my/article/214936/focus/broken-roads-brokenlivesmalaysianscryfoulfor-pothole-solution>
- Sistem Aduan JKR Malaysia. (2024). Jkr.gov.my. <https://aduan.jkr.gov.my/>
- Medium. (2020). *Agile Methodology* (C. Jayathilaka, Ed.).
<https://medium.com/@chathmini96/agile-methodology-30ec4cdf3fc>
- Sharma, S. (2012), International Journal on Computer Science and Engineering (IJCSE), *Agile Processes and Methodologies: A Conceptual Study*, 4(5), 892.
https://www.academia.edu/download/34703283/IJCSE12-04-05-186_1.pdf
- Khairulrijal, R. (2024, April 29). Court awards pothole victim RM721,000 for damages and injuries [NSTTV]. *NST Online*.
<https://www.nst.com.my/news/crimecourts/2024/04/1043922/court-awards-pothole-victim%C2%A0rm721000-damages-andinjuries%C2%A0nsttv>
- Fui Siew, E., Ireland-Hay, T., Theobald Stephens, G., J.J. Chen, J., & P. Taylor, M. (2005). A STUDY OF THE FUNDAMENTALS OF POTHOLE FORMATION. *Light Metals* 2005, 763.
- E. Cox, D. (Ed.). (1975). On the Interpretation of Potholes. *Creation Research Society Quarterly*, 12(1), 25–31.
- Kandoi, H. Agarwal, A. Revadekar and B. N. Chaudhari, "Pothole Detection using Accelerometer and Computer Vision with Automated Complaint Redressal," *2021 3rd International Conference on Signal Processing and Communication (ICPSC)*, Coimbatore, India, 2021, pp. 257-259.

- Fmt. (2021, January 4). *KJ's pothole mishap: a lesson down the road*. Free Malaysia Today
| FMT. <https://www.freemalaysiatoday.com/category/highlight/2021/01/04/kjs-potholemishap-a-lesson-down-the-road/>
- Masse, L. C., Fuemmeler, B. F., Anderson, C. B., Matthews, C. E., Trost, S. G., Catellier, D. J., & Treuth, M. (2005). Accelerometer data reduction: a comparison of four reduction algorithms on select outcome variables. *Medicine and science in sports and exercise*, 37(11), S544.
- K.T. Chang, J. R. Chang, and J. K. Liu, Detection of pavement distress using 3D laser scanning technology, In Proceedings of the ASCE International Conference on Computing in Civil Engineering (2005), 1-11.
- She, X., Hongwei, Z., Wang, Z., & Yan, J. (2020). Feasibility Study of asphalt pavement pothole properties measurement using 3D line laser technology, *International Journal of Transportation Science and Technology*, 10(1), ISSN 2046-0430.
- Koch, and I. Brilakis, Pothole detection in asphalt pavement images, *Advanced Engineering Informatics*, Vol. 25 (2011), 507-515.
- M. Jog, C. Koch, M. Golparvar-Fard, and I. Brilakis, Pothole properties measurement through 2D recognition and 3D reconstruction, In Proceedings of the ASCE International Conference on Computing in Civil Engineering (2012), 553-560.
- Lokeshwor, L. K. Das, and S. K. Sud, Method for automated assessment of potholes, cracks, and patches from road surface video clips, *Procedia –Social and Behavioral Sciences*, Vol. 104 (2013), 312-321.
- Kim, T., & Ryu, S.-K. (2014). Journal of Emerging Trends in Computing and Information Sciences. *Journal of Emerging Trends in Computing and Information Sciences*, 5(8), ISSN 2079-8407.
- Tashildar, A., Shah, N., & Gala, R. (2020). International Research Journal of Modernization in Engineering Technology and Science. *APPLICATION DEVELOPMENT USING FLUTTER*, 02(08), 1262.

Appendix

Pre-Test Survey

Hello! Thank you for taking the time to participate in this survey. I am conducting this as part of my Final Year Project titled *"RePothole – The Pothole Detector Mobile Application for Vehicles Using Accelerometer."* The purpose of this survey is to understand your experiences with potholes and gather feedback on how a mobile app can help detect and report road damage in real time.

Your responses will help guide the design and development of the application to ensure it is practical, user-friendly, and meets real-world needs. This survey will take only a few minutes, and all information you provide will be kept strictly confidential and used for academic purposes only.

Thank you for your support!

Any inquiries do not hesitate to contact me via email 78013@siswa.unimas.my

hanasaffiyya76@gmail.com [Switch accounts](#)



Not shared



Draft saved

How familiar are you with mobile apps that use sensors (like GPS, accelerometer)?

- ☐ Very familiar
- ☐ Somewhat familiar
- ☐ Not familiar at all

Before trying this app, how confident are you that a phone can detect potholes using vibration?

- ☐ Very confident
- ☐ Somewhat confident
- ☐ Not confident
- ☐ Not sure

What do you expect this app to help you with? *(Select all that apply)*

- ☐ Detect potholes
- ☐ Record GPS location
- ☐ View potholes on map
- ☐ Other

Post-Test Survey

Untitled form

Thank you for testing the *RePothole* mobile application and web interface. This feedback form is part of my Final Year Project, "*RePothole – The Pothole Detector Mobile Application for Vehicles Using Accelerometer.*" The goal of this form is to gather your thoughts and experiences after using the app, including its features, ease of use, and accuracy of pothole detection.

Your responses are important and will help evaluate the app's effectiveness and identify areas for improvement. The feedback you provide will be used strictly for academic purposes and treated with full confidentiality.

Thank you again for your time and valuable input!

Any inquiries please contact me at @78013@siswa.unimas.my

hanasaffiyya76@gmail.com [Switch accounts](#)



Not shared

Did the app detect potholes when you shook the device or drove over bumpy roads?



Yes



No

Did you use RePothole website to view potholes on the map?

☐ yes

☐ no

If yes, which of the following features worked for you? (Select all that apply)

☐ Markers appeared on the map

☐ Filtered by date/location

☐ Changed marker status using checkbox

☐ Viewed info popup from markers

☐ Did not try any features

How accurate do you think the pothole detection and logging was?

Not sure 1 2 3 4 5 Very Accurate
☐ ☐ ☐ ☐ ☐

What improvements would you suggest for the app or website?

Your answer