INFORMATICS INSTITUTE OF TECHNOLOGY

In Collaboration with

UNIVERSITY OF WESTMINSTER

Machine Learning Based Crowd Sourcing Approach to Identify Road Surface Quality with Mobile Phone Sensors

A dissertation by

Tharindu Buddhika Deshappriya – 2014232

W1582951

Supervised By

Mrs. Sulochana Rupasinghe

Submitted in partial fulfillment of the requirements for the

Bsc (Hons) Software Engineering degree

Department of Computing

May 2019

Declaration

I hereby certify that this project report and all the artifacts associated with it is my own work and it has not been submitted before, nor is currently being submitted for any degree program.

Full Name of Student: S.T.Tharindu Buddhika Deshappriya

Registration No.: 2014232

Student’s Signature: ………………………… Date: …………………………

Abstract

Transportation is crucial for development of the civilization in the world. Transport supports a major contribution on economic growth and globalization. In this present world, People seek for comfort in transportation in vehicles. Considering the number of vehicles using man made roads and poor maintenance on damaged roads, road surface quality can be an issue. Poor road face quality can cause vehicle suspension damage, tire punches and steering miss alignments. Due to those vehicle damages, people tend to use quality road surfaces.

This project’s main focus is to provide road surface quality monitoring system based on machine learning and using crowdsourcing approach. The system provides a road surface quality monitoring system for the user to take necessary precautions before travel on a poorly developed road.

To achieve the objective, data was gathered from users using crowdsourcing method. With those gathered raw sensor data, those raw data will be smoothed by using filters. After that process the data will be applied to an IRI calculation. After that those calculated data will be trained under a machine learning algorithm. With classifying those data and analyzes them, the quality of the road surfaces will be delivered.

Keywords: Machine Learning, Accelerometer, Road Transportation, Mobile Application

Acknowledgement

I would like to express my heartiest gratitude to everyone who have supported me and contributed me to complete my project successfully. And then also I would like to show my heartiest gratitude to my loving parents, colleagues who was there for me and encouraged me to complete this project.

Next, I would like to show my gratitude and appreciation towards my supervisor Mrs.Sulochana Rupasinghe, for her guidance and motivation throughout the year, and also offered valuable advices and support to complete the project successfully.

Table of Contents

List of Figures

List of Tables

List of Abbreviations

AAA	American Automobile Association
DTW	Dynamic Time Warping
IRI	International Roughness Index
MEMS	Micro Electro Mechanical System
GPS	Global Positioning System
PC	Point of Curvature
PT	Point of Tangent
SMA	Simple Moving Average
BPF	Brand Pass Filter
ML	Machine Learning

Chapter 1 : Introduction

1.1 Chapter Overview

This chapter focuses on giving a clear understanding about the background of the project. This project is done to identify road surface conditions using mobile phone accelerometer sensors in a proper approach. This chapter describes the problem domain and the problem most road users face day to day. Moreover, this chapter includes the aims, research objects accomplished and the resource requirements used in this project.

1.2 Project Background

The basic role of roads is to connect two geographical locations for human needs. Humans interact with roads in most daily activities. Roads make an important contribution to the economic growth of a country because productivity depends on the roads and also, roads impact social and environmental aspects. The condition of roads is an indicator of the development of a country in the present world we live. So, governments are trying to keep roads in good shape.

Roads require some sort of maintenance over their years of usage. As the number of vehicles occupying roads is increasing by a huge number each year, these roads get damaged accordingly. Environmental conditions also have been a main reason for roads getting damaged. Even asphalt roads are being damaged due to the bad weather conditions. Lack of proper drainage systems for roads also contributes to the damage. Some road damages are caused due to poor construction. The probability of a road getting damaged is high if the thickness of the asphalt or concrete layer is not up to standard. Another reason for poor road conditions is the lack of maintenance. Roads have been damaged for pipe laying, cable laying etc. by authorities, but after the work is done sometimes, they do not back-fill properly or fix the damaged roads. Even if they are fixed and patched, the roads are not being restored to the previous state of quality. Another main reason for damaged road conditions is using the same asphalt layering for all the roads. As we all know most of speed bumps have been laid unnecessarily not following the standards given by the relevant authorities. So, many people hate the fact that roads have speed bumps because these speed bumps disturb smooth travelling.

Several governments have been concentrating on the road surface quality monitoring from the 1980s. So, the World Bank introduced the International Roughness Index (IRI) in 1986(Yuchuan et al., 2014). IRI is commonly used for evaluating and managing road surfaces. It is based on a Vehicle's accumulated suspension motion value divided by the distance travelled in the measurement. There have been a number of road surface profiling devices to gather the data of road surface conditions. South Dakota Profiler, Profilographs, Dipstick, Mays Meter are some of them and most of these are time consuming and require a huge labor input.

Machine learning is a method of analyzing data that automates analytical model building which was founded by Arthur Samuel in 1959 (Machine Learning: What is it and Why it Matters, 2018). It's a part of Artificial Intelligence which allows computer systems to learn from data, identify patterns and take decisions with minimal human interaction. Machine learning is widely used in the present world. Nowadays, smart phones contain number of sensors such as accelerometer, gyroscope, biometrics, proximity sensor, barometer and etc. The accelerometer is used to detect the orientation of the mobile phone. The accelerometer of mobile phones is able to measure the linear movement, i.e. the directional movement of the device along the three axes.

1.3 Problem Statement

In the present world we live in, people seek comfort in most of their daily activities. When relating this to roads and transportation, people like to be comfortable while travelling. In order to achieve that, roads have to be in good condition. Most people do not like to use roads in poor condition. Road Surface Quality: (Anthony & Jim, 2017), a research conducted on what road users want from highways in England has concluded that what road users expect from a road is a surface without dips, bumps, undulations, potholes and deep ruts. Specially, people like pregnant women and patients having difficulties consider more about the quality of the roads when travelling.

With the above-mentioned aspects, road surface monitoring is a basic layer in solving this problem. What we do in normally is look at the road surface and guess the horizontal curves and pothole size. But, most of the time, what we get is different from what we guess. Specially in night travelling, the drivers can't identify the road anomalies from a distance until they come close by. Road surface monitoring can be useful to both authorities and road users. Authorities can easily identify the damaged roads with a system which uses crowd sourcing technology instead of using methods which are time costing and needs work by labor. By implementing a system to monitor the road conditions, government authorities can reduce the spending costs for identifying the road conditions by expensive mechanisms. Though road quality is a key factor in the development of a country, computational systems which can assess the road quality will be useful for governments to show the existing level of roads according to standards. When this road quality monitoring is related to road users, they can easily find out both the well-conditioned and poorly conditioned roads and road users will be able to select the roads to travel accordingly.

1.4 Project Aim

To research, design, develop, test and evaluate a crowd sourcing platform to identify the road conditions by using GPS and accelerometer sensor data of smart phones which will help users to get an idea about the road surface and minimize the accidents that occur due to poor road conditions.

1.5 Motivation

The author was able to engage in a proof of concept using smartphone accelerometer which was conducted during his placement year. The author was studious about giving a solution to a real-world problem using this sensor and has identified that there are thousands of accidents which occur due to poor road conditions annually and plenty of drivers are facing many difficulties because of this problem. With the facts the author possesses, it is clear that this is an important problem which most of the drivers are facing and which is needed to be solved.

1.6 Research Question

How to develop a machine learning based crowd sourcing system to monitor the road surface quality by considering the users' vehicle suspension variance?

IRI value will be calculated using the accelerometer sensor data and GPS data of the mobile phone. Using the IRI value, the system will be able to identify the vehicle type of the user. So, road surface condition will be categorized using the vehicle type and the IRI value.

1.7 Research Objectives

- To identify
    - Different road conditions using the data from the accelerometer sensor of smart phones.
    - The route from obtaining actual real-time location of the vehicle by using GPS technology.
- To analyze
    - Large amount of data from a single vehicle and categorize the vehicle type using machine learning.
    - The uncleared data that were affected by speed, accelerometer reorientation using different filters to smoothen the sensor data.
    - The data and configure them to classify the output by presenting the condition levels of the road.
- To implement,
    - From all the conducted data and detected condition classification of the road on a map.
- To evaluate the accuracy of the proposed system by conducting testing extensively.

1.8 Features of the Prototype

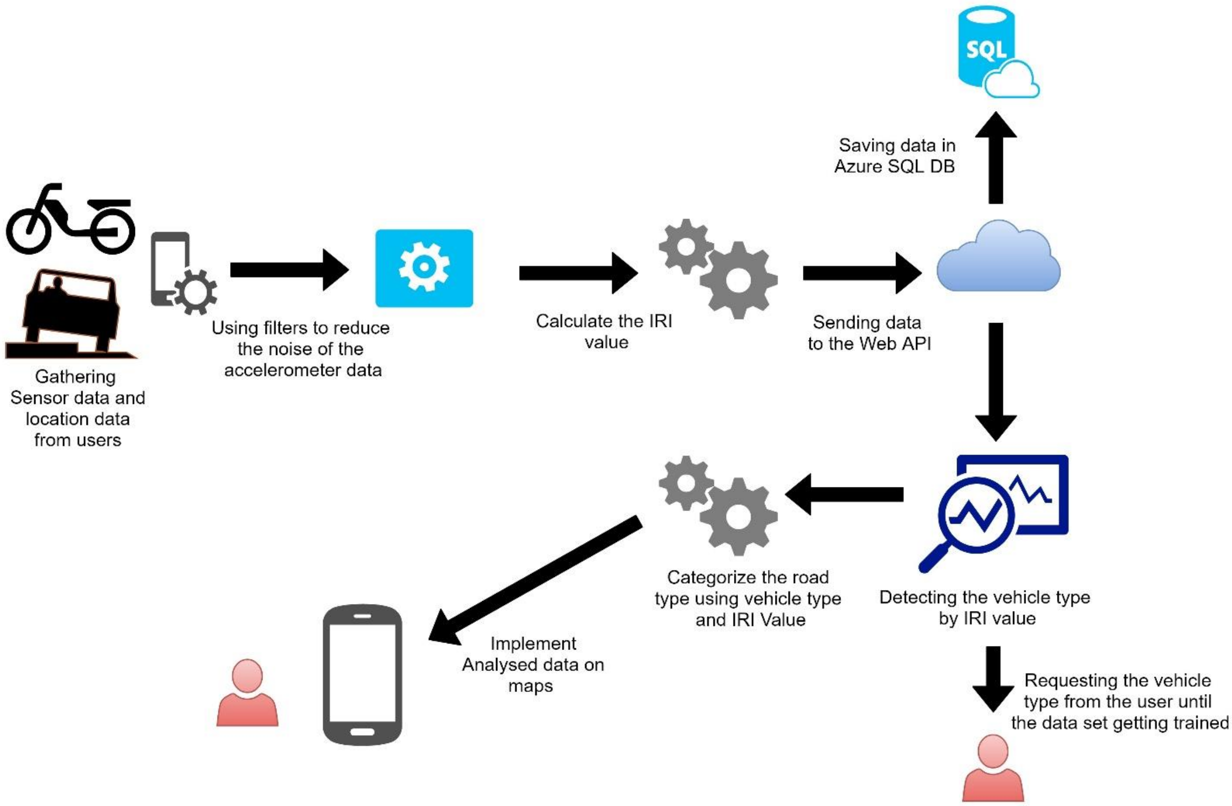


Figure 1.8.1: Rich Picture Diagram

- Gathering sensor data from users who have installed the mobile application.
- Using High Pass and Low Pass filters to reduce the noise.
- Using Z axis filter for use the z axis data.
- Getting the street name and calculate the distance travelled using GPS.
- Calculate the IRI value for each road using the gathered data from the user.
- Prompt the vehicle type from the user until the data set is enough for training.
- Identifying the vehicle type using the Machine Learning model.
- Categorize the road type using vehicle type travelled by user and IRI value of the road.

1.9 Scope of the Project

In Scope	Out of Scope
• Gather data from android devices. • Request user to select the vehicle type if the system doesn’t have data on relevant road for all three vehicle types (Two wheelers, Three wheelers, Four wheelers). • Categorize the data into three vehicle types. • Visualize the road conditions on a map into three road condition levels.	• Gathering data when the user uses the phone while travelling. • Pothole visualizing on map. • Getting data from the user at any speed (Data will be gathered if the vehicle is moving more than 5kmph.). • Data gathering when user is not connected with the internet. • Data gathering when user is not having GPS connectivity.

Table 1.9.1 : Scope of the Project

1.10 Resource Requirement

Hardware Requirements	Software Requirements
• Laptop or a Desktop PC with Core i5 processor and 8GB ram. • Number of mobile phones which are having accelerometer sensor for testing	• Android Studio • Microsoft Office Package for Documentation • Star UML, Microsoft Visio 2019 and Draw.io to draw the diagrams • Spring Tool Suite • Microsoft SQL Server 2016 • Azure SQL databases • Python 3.6.2

Table 1.10.1:Resource Requirements

1.11 Chapter Summary

According to this chapter, poor road surface quality has become a major issue that vehicle users are facing. So, the proposed approach will be useful to the users to pre identify the road surface conditions just looking at a map. This chapter embraces the overview of the background, the aims and the research objectives of this project. The features to be implemented in this project were outlined by the rich picture diagram of the system. Lastly the resource requirements were mentioned which were used to develop the proposed solution.

Chapter 2 : Literature Review

2.1 Chapter Overview

This chapter addresses a discussion about existing literature associated to identifying road surface conditions using mobile phone accelerometer sensors. This provides in-depth analysis, evaluation and a comparison review of the current approaches to solve this problem. The algorithms which are related to this project will also be discussed within this chapter. The main purpose of this chapter is to give a review about the research scope and to identify the possible and effective approaches to be used in designing the solution.

2.2 Problem Justification

Road surface roughness monitoring is declared as a key element in providing smoother and safe travelling for the road users. According to (Bumpy Road Ahead: 2018), one third (33%) of the United States' major urban roads are have been marked as roads with poor condition. Accidents happening due to poor road conditions have increased annually. Francis (2017) has concluded more than 10% of accidents happened due to the poor road conditions in Northern Corridor in Kenya. Many researches have concluded that fuel efficiency of vehicles depends on road surface quality (Smoothness Matters: 2009). Due to the frequent usage of roads which have potholes and unnecessarily laid speed bumps, vehicles get damaged in various ways such as suspension damage, tyre punctures, steering misalignments and etc. According to the American Automobile Association (AAA), drivers have to spend 300$ (USD) annually to fix issues created in vehicles due to potholes. Ventor (2008) has conducted a study about cargo damage due to poor road conditions. The researcher was able to identify that cargo companies have to spend extra money due to the damages of the cargo.

Most of the governments and road authorities still use manual methods such as doing surveys to identify the roads which have poor conditions. But, those type of manual methods have lower accuracy and they are highly labor intensive as those are large projects. And those methods take a significantly long time to conduct the analysis. And these manual methods do not have accuracy. Identifying road roughness is also a factor in decision making for governments to allocate their budgets to maintain the roads. So, many researchers and governments have considered about road surface quality monitoring for decades.

With the introduction of IRI (International Roughness Index), road roughness measuring has been standardized. IRI was introduced by the World Bank in 1986. IRI is a scale to identify the roughness of the roads using a motor vehicle. IRI is designed to give a longitudinal profile of a road travelled and give the measurements in meters per kilometer (m/km) or millimeters per meter (mm/m). The device which is used to take the measurements is fixed to a vehicle called the 'Golden Car' of which the suspension details are known. IRI has become the road surface roughness index most widely used by entities worldwide to manage and evaluate roads.

2.3 Existing System Analysis

Xiao, Daniel (2018) has conducted a research about mobile crowdsensing system for road surface assessment. This assessment includes two major components such as the mobile data gathering component and web-based server. These researches have used AndroSensor for android phones and CrowdSense for iPhones which are already implemented by previous researchers to gather accelerometer sensor data of the mobile phones. And also, the researchers have used the lightweight, low cost and cloud-based framework, Google Fusion Tables, for data visualization. The raw data of the accelerometer sensor and GPS sensor gathered by smartphones were uploaded to the cloud server. The cloud-based server was able to gather road roughness information from a number of crowdsourcers. And also, accelerometer reorientation was done to align the smart phone axis with the vehicle axis. So, the x axis and y axis sense vehicle's horizontal movements while the z axis senses the vertical movements caused because of the road surface roughness. A speed filter was used to ignore the sensor data which was gathered at a speed less than 5kmph and another filter was applied to the z axis. Most of the previous researchers have considered about the pothole and bump detection. But, in this study, they have uses IRI-proxy calculation to measure the road conditions and they have calculated transient events count. With the usage of IRI calculation, researchers have considered about the entire road condition. In this research, the road condition indexes are categorized into 4 levels based on the IRI measurement.

Road Surface Condition	IRI Value
Good	$0 \leq IRI < 4$
Fair	$4 \leq IRI < 7$
Poor	$7 \leq IRI < 10$
Bad	$IRI \geq 10$

Table 2.3.1 : Road Surface Quality With IRI

Testing of the implemented system was done in Texas. The researchers used two iPhones and one Android phone and they used a Toyota corolla car for data collection. They have concluded that high driving speed causes the z axis acceleration and identifying transient events has less accuracy because the positional accuracy of GPS sensors is 2-3m. And also, they concluded that the accuracy of the results was high and data was getting updated frequently by using the crowdsourcing methodology. The researchers were only able to detect the potholes and the researchers have not discussed the suspension effects that occurred due to the vehicle type.

Thanuka, Varun & Peter (2018) has done a research about road quality assessment using 3D accelerometers. Instead of using z axis acceleration sensor data, the researchers have used all the axis (x, y, z) data in the research. They have discussed about using MEMS (Micro Electro Mechanical System) sensors over using imaging modalities such as radar because these sensors have a lower complexity and are a low-cost solution. And also, the researchers have discussed about deploying a unit which has a MEMS accelerometer sensor virtually to all the vehicles. They have used accelerometer development board EVAL-ADXL345Z-DB and fixed that sensor to the dashboard of two vehicles. The data about road anomalies was also recorded manually with the times of testing. The recorded data from the sensors was first processed to remove effects caused by gravity. Relief Algorithm is a statistical approach to detect features with few heuristics. The Relief implementation was done in Matlab Statistics and Machine Learning Toolbox running in a classification mode. The researchers have concluded that all the accelerometer axes have a relevant contribution in road surface monitoring. And also, they have concluded that mounting sensors directly to the vehicle causes faulty readings in the accelerometer when the vehicle is moving through rough road surfaces. And also, deploying a sensor to the vehicle is not feasible as it is costly.

Another research has been done about analyzing road surface conditions using embedded smartphone sensors by Alqudah & Sababha (2017). Researchers have discussed about using a gyroscope sensor for identifying road conditions. A gyroscope sensor is able to identify the angular rotational velocity of a device. On the other hand, a gyroscope sensor uses the gravity of the earth to identify the orientation. Same as the accelerometer sensor, the gyroscope is also collected data in all 3 axes. The researchers fixed a mobile phone on the dashboard of a vehicle which was used in testing. The mobile phone is able to gather the data of speed, GPS locations and sampling time. The tests took place in 3.7km road and were repeated 5 times. The gathered data from the 5 times a seemed to be equal. So, the researchers used DTW (Dynamic Time Warping) method to identify the similarity more accurately. DTW is method which able to find the similarity between time sequences which different in speed. The researchers have concluded that a gyroscope can be used to detect road surface conditions by this research and the data is repeatable. The researchers were not able to identify the same road surface data multiple times as this research doesn't use the crowdsourcing mechanism. So, if the road anomalies have been repaired, the system won't be able to identify.

Waleed et al. (2018) has done a research on the evaluation of pavement roughness using an android phone. Researchers have used two android smart phones and the phones were fixed to the vehicle's dashboard. Android application AndroSensor was used on two mobile phones to gather the acceleration data and GPS location. The researchers were concerned about the vehicle moving speed and they conducted the test at 64kmph and 80kmph. Instead of manually marking the road conditions which similar to the other researchers, these researchers have used the South Dakota Profiler to gather data. The South Dakota Profiler is a road profiler which uses a laser mechanism. The profiler generates an actual IRI value. The IRI values gathered from the smartphone and South Dakota Profiler were analyzed using MATLAB software. Researchers have explored that the type of the smartphone is an important fact in monitoring the road surfaces. And also, the researchers have concluded that accelerometer sensor values are speed dependent. This research also does not discuss using a crowdsensing mechanism to gather data.

Smart patrolling: (Gurdit et al.,2017) conducted a research on road surface monitoring using smartphone sensors and crowdsourcing. The research focused on detecting road surface conditions using low cost technologies. So, the researchers used a mobile phone accelerometer to gather data of the acceleration. Data is gathered from many crowdsources. Since the accelerometer affected by gravity and other vibrations, the raw sensor data should be passed through different types of filters. First a speed filter was used to ignore the raw data which was gathered at less than 5kmph by using the GPS sensor. This filter was used because the accelerometer does not provide the best results about road conditions when the vehicle is moving at a low speed. Euler’s angle Mechanism is also used in this research for accelerometer reorientation. Then a Z axis filter was applied to discard the x axis and y axis data because only the z axis is used to identify vertical movements. A SMA (Simple Moving Average) filter was also used to smooth values and remove the noise captured by the accelerometer sensor due to irrelevant vibrations. Another filter called BPF (Brand Pass Filter) was used to smooth the data further to reduce the noise captured due to hardware sensitivity. BPF contains two filters called a low pass filter and a high pass filter. By using all these filters and mechanisms, researchers were able to get cleaner and more accurate data from the accelerometer sensor. DTW mechanism is used to match the patterns of the gathered data. This research has specifically discussed using the DTW mechanism instead of using machine learning and threshold-based techniques. The testing took place over the roads of Chandigarh City in India. 6 different smart phones were used and the phones were fixed in different positions of the vehicle such as Dashboard, Pilot Seat, Co-Pilot Seat and Back Seat. Detected potholes when doing the tests have also being marked manually. The road anomalies which have identified by using DTW were compared with the manually marked data. They identified that the phone which was placed in back seat of the vehicle got a low detection rate 31.00% while other phones had a high detection rate of 94.38%. The identified potholes and other road anomalies were marked in Google Maps by markers. The researchers have shown that the overall road condition is a factor in identifying road anomalies as the road anomalies in roads in a poor condition were not able to be identified because of the frequent changing of the accelerometer sensor data. The researchers were able to identify the potholes and speed bumps with a success rate of 88.66% and 88.89% from the proposed system. This system has not considered about the accelerometer sensor data variance due to the vehicle suspension.

Jonathan & Shaohu (2018) has done a research about identifying and calculating the horizontal curves of roads using Smartphone sensors. The researchers have used rotation values of the z axis measurement of the gyroscope sensor to detect the horizontal curves. The implemented system contains four main system components such as data collection, data correction, identification of curves and calculating the curve. The data collection part includes the gathering data of the GPS, accelerometer sensor and gyroscope sensor. Butterworth Low Pass Filter was used on raw accelerometer sensor data and gyroscope sensor gathered from smartphone. The Butterworth Low Pass Filter is applied to reduce the noise of high frequency measurements. Researchers have concluded that noise effects can be removed from using the Butterworth Low Pass Filter. Another filter called Extended Kalman Filter was used to gain the accuracy of GPS sensor to identify the speed and locations. To identify the horizontal curves from the gathered sensor data, K-Means algorithm which is an unsupervised machine learning algorithm was used. Horizontal curve radius was identified and calculated by using PC (Point of Curvature) and PT (Point of Tangent). Testing was done using an android phone and 100 miles of highway in South Dakota was used. Researchers were able to achieve a 97% success rate of identifying and calculating radius of horizontal curves by this research. But, with relevance to the road anomalies, horizontal curves are not the only road anomaly that affects smooth travelling.

Author	Mobile Phone Sensor Used	Crowd Sourcing	Roughness Assessment Type	Technologies Used	Filters Used	Success Rate
Xiao Li, Daniel W. Goldberg (2018)	Accelerometer, GPS	Used	IRI Value, Pothole and Bump detection	Improved special algorithm to identify road conditions.	Speed Filter, Z axis filter, Accelerometer reorientation	92.59%
Thanuka Wickramarathne, Varun Garg, Peter Bauer (2018)	Accelerometer (not mobile phone)	Not Used	Crack, Pothole, Bump, Dirt Road, Paved Road, Rough Road	Machine Learning	Not used	Not Mentioned
Yazan A Alqudah, Belal Sababha (2017)	Gyroscope, GPS	Not Used	Driver Alerting Stripes	Dynamic Time Warping	Not Used	Not Mentioned
Waleed Aleadelat, Khaled Ksaibati, Promothes Saha (2018)	Accelerometer, GPS	Not Used	IRI value	Pattern Recognition Techniques using MATLAB	Not used	Not Mentioned
Gurdit Singh, Divya Bansal, Sanjeev Sofat, Naveen Aggarwal (2017)	Accelerometer, GPS	Used	Potholes, Bumps	Dynamic Time Warping	Speed Filter, Accelerometer reorientation, Z axis filtering, Simple average Moving Filter, Band Pass Filter	88.66% (Potholes), 88.89% (Speed Bumps)
Jonathan S. Wood, Shaohu Zhang (2018)	Gyroscope, GPS	Not Used	Horizontal Curves	K-Means Machine Learning Algorithm	Butterworth Low pass Filter, Extended Kalman Filter	97%

Table 2.3.2 : Existing Systems

2.4 Algorithmic Analysis

According to research done throughout decades, several mechanisms have been implemented to detect the road conditions such as Dipstick, Profilographs, Mays meter, South Dakota Road Profiler and etc. These implemented systems are very high in cost and labor input is also very high. So, using an accelerometer sensor for detecting road anomalies came into consideration. Eriksson et al. (2008) was able to monitor the road surface using an accelerometer sensor. The accelerometer sensor was fixed to different taxis during the research. Accelerometer sensor to identify the vertical movements and vibrations of the vehicles and GPS sensor for identifying the locations will be used in this research. Accelerometer provides acceleration values of x, y and z axes. X axis and y axis identifies horizontal movements while z axis identifies the vertical movements. So, basically the z axis measurements are getting affected from the road surface quality. When designing a solution which can gather a large amount of data, deploying separate hardware components to the vehicles will not be feasible because of the below reasons.

- Applying the units to the vehicle needs approval from the vehicle manufacturers and old vehicle users have to buy the units.
- Applying the different units will not be cost effective.
- GSM modules which are very costly also have to use to send the data.

So, the road surface quality monitoring can be done in a better way using accelerometer sensor of mobile phone than using hardware component units including accelerometer and GPS sensors fixed into vehicles. With the rapid development of mobile, smart phones are able to do much more than merely making calls. In the current world, most of all the smart phones consist of a number of sensors such as GPS, accelerometer, gyroscope, proximity and etc. (Bharadwaj & Sastry, 2014). Accelerometer sensor can be used in detecting road surface conditions as,

- Mobile phones already have communication with the internet. No need of using extra GSM modules or Wi-Fi modules to connect to the internet. So, the gathered data and web can be easily synced.
- No extra cost for the users as user has to use the same mobile phone which user already use.
- Since most people have smart phones, a simple application would be enough to gather data from many users in many locations.

- Updating to the new versions can be done by updating the mobile application.

So, according to a study conducted by Viengnam & Hiroyuki (2013), they have concluded that using smartphones is a very low cost & easily operable solution approach.

In this research, the accelerometer has to be reoriented to the desired orientation to get the best results. Many researchers have used Euler Angles mechanism to reorient the accelerometer introduced by Leonhard Euler. Accelerometer reorientation has to be done to align the axis of the mobile phone with the vehicle axis in order to get the best sensor readings from the mobile phone accelerometer.

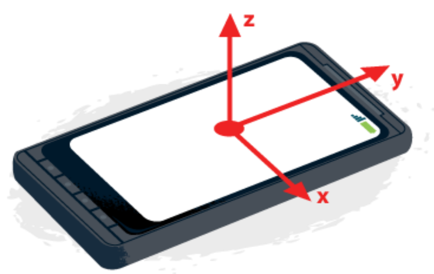


Figure 2.4.2 : Phone Axis

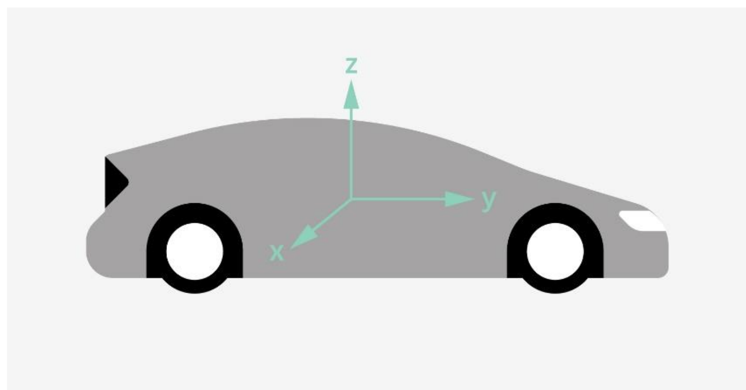


Figure 2.4.1 : Car Axis

Speed filters have to be used to the gathered data from reoriented accelerometer sensor data. When the vehicle is moving slowly, the accelerometer changes due to the road conditions are not accurate and the data can be varied. So, the previous researchers have limited the minimum speed to take the measurements to 5kmph. So, accelerometer sensor data greater than 5kmph will be take taken for further analyzing (Gurdit et al.,2017).

Z axis filters also have been used to consider only about the z axis. Though the accelerometer detects acceleration in x, y and z axis, z axis is giving best measurements of the accelerometer sensor affected by the road surface. So, z axis will be needed for analyzing the vertical acceleration changes.

Low pass filter is used to filter the attempts to pass the low frequency signals through the filter with frequencies above the cutoff frequency which is predefined. This filter has an equation as above.

output[0] = alpha * output[0] + (1 – alpha) * input[0]

Alpha can be calculated as alpha = t / (t + dt). Time constant (t) is the signal length and the dt is the sample rate of the sensor (time between the updates). Low pass filter can be used to ignore the signals that are shorter than the time constant and to alter the signals that are much longer than the time constant.

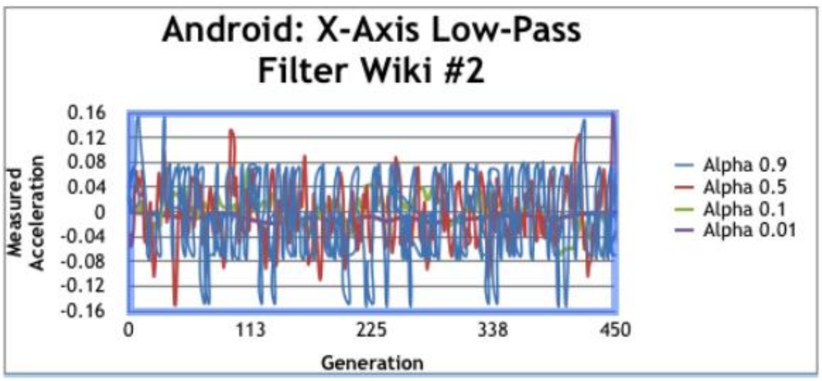


Figure 2.4.3 :Low Pass Filtered Data

According to the graph, the smoothing effect can be identified if the alpha is less than 0.1. When alpha becomes lesser, time constant is getting larger and the filter performs more smoothing to the signal.

The raw data given by the accelerometer consists of gravity force of the earth. Without using a gyroscope sensor, it's hard to remove the gravity force from the sensor readings. The gravity force can be isolated by using a high pass filter which can be conversely done by a low pass filter.

Global Positioning System (GPS) is a radio navigation system which is a free service owned by United States government and maintained by United States' Air Force. Basically, it is uses radio waves between satellites to identify the location of the device. Since smart phones include GPS sensors, phones are able to identify the locations by receiving the data (Hildenbrand, (2018). According to *gps.gov*, identifying geo-location using GPS sensor is accurate up to within a 4.9m. GPS receivers have to connect to four or more satellites at one time and figure out the distance to each and every satellite. Location can be calculated by the distance to the satellites. This process is based on a mathematical principle called trilateration (Brain & Harris, 2018).

With the digitalization of most of the things, significant changes have happened to the common man's day to day life. For further improvements of digitalization, data is gathered about human interacted activities. Crowdsourcing is an approach which can be used to carry out tasks from the unknown workforce called the crowd. Applying this approach to software development projects, crowdsourcing has been mostly used for the data gathering. Data gathering from crowd sourcing has been demanded highly because of following reasons.

- Low cost
- Large amount of data can be gathered
- Data gathered by crowdsourcing is mostly live data or recently gathered data.
- Data can be gathered from various type of users worldwide.

Machine Learning is called as an automated learning with lesser human interact. Machine learning algorithms can do predictions and analyses in software development with a high accuracy. Mostly machine learning uses sample data known as training data to perform a task instead of using programmed data. Machine learning algorithms have outputs which have fast & high accuracy. Machine learning can be classified as supervised and unsupervised.

Supervised machine learning algorithms used to get outputs from a labeled training data set and can be used to map with new examples. Basically, training data set can be named as the teacher of the supervised machine learning. Supervised machine learning can be further grouped as classification algorithms and regression algorithms. Classification algorithms can be used when the output of the algorithm is a category while regression can be used to output a value.

Unsupervised machine learning algorithms are trained using unlabeled, unclassified or uncategorized data. Discover the hidden patterns, to determine how to describe data or to find the structure of the data, unsupervised machine learning can be used. Without responding to the feedback, these algorithms identify the commonalities between the data gives the output. The most common usage of unsupervised machine learning algorithm is clustering which splits the data into groups according to similarity.

K-nearest neighbor algorithm is a supervised data classification machine learning algorithm. It uses several labeled data and uses them to learn to predict new data. When a new data inserted to the algorithm, it looks for the closest labeled data using the distance calculating metrices such as Euclidean, Manhatten and Minkowski. Then it sorts the distance between the new data and the training data. So, the label which most of the neighbors with the minimum distance values is given for the new data. “K” in the K-nearest neighbors algorithm means about how many neighbors to be checked. Though this classification is done by using the known classification of other points, K-nearest neighbor is a supervised machine learning algorithm.

2.5 Research Gap Justification

As discussed in the research papers, most of the researchers have explored and concluded about using accelerometer sensor to gather data to monitor the road conditions. Using mobile phone sensors have been identified as the best and most feasible method to identify the road surface conditions. Using mobile phone is also an easy because, GPS sensor which needs to identify the geo location is also embedded to the phone. The researchers have been used filters such as speed filters, z axis filters and etc. These filters have used to smooth the data and obtain the best results from the testing (Jonathan & Shaohu (2018)). But researchers have not considered the effects of the variance of vehicle suspension on the accelerometer sensor data. Accelerometer raw data may vary depending on the vehicle type (Thanuka, Varun & Peter (2018)). A car will sense accelerometer sensor data more smoothly while a bicycle will sense the data roughly from the sensor because of the changes of suspension in both vehicles. Since the author is using crowdsourcing technology to this research, data will be gathered from many vehicle users. If the gathered data are varied from vehicle to vehicle, IRI calculation will be affected and give different values for the same roads. As a solution for this problem, original IRI calculation for a specific road from a database will be recalculated to a new IRI value varying from the vehicle type (two wheelers, four wheelers). By applying those calculations, the IRI calculation done from different vehicle types can be equal. So, the data can be analyzed and could be able to get accurate results.

2.6 Chapter Summary

This chapter focused on discussing the previous researches done related to road surface quality assessment. The research gap was identified by reviewing the previous research papers. As discussed in this chapter, a machine learning based approach was selected to develop the prototype to identify the road surface conditions considering the vehicle type. Crowd sourcing approach also was selected because the road conditions can be getting changed daily.

Chapter 3 : Project Management

3.1 Chapter Overview

This chapter is focused on project management procedure. This chapter consists of a comprehensive detail about the current software development methodologies, research methodologies to be followed in this project. The chapter also includes the schedule of the outcomes to be completed with creating organizational diagrams such as Gantt Chart and Work Breakdown Structure.

3.2 Software Development Methodology

Software development methodology in software development is known as a mechanism used for structure, planning and controlling the process of an information system. Every software development methodology does have pros and cons.

From the software development methodologies, waterfall methodology was not used because it has a straightforward flow between the developments stages and it does not support the requirement changes to be done. Agile methodology also was rejected as it focuses mainly on development and less focus on designing and documenting. Since spiral methodology is suits for large scaled and complicated projects and final year project has to be completed within a time frame, spiral methodology also cannot be used.

For this undergraduate project, iterative methodology is the most suitable methodology to be used since researches to be done parallelly with the development of the system and the requirements can be changed according to the findings of the researches. So, iterative model supports requirement changes in different phases and this research project have to complete within a limited time frame, iterative methodology has been selected.

Iterative Methodology

Iterative model is based on implementing small set of software requirements and continue iteratively until the complete software is being implemented. Iteration methodology doesn’t need all set of requirements of the system at the beginning. Iterative methodology can be equal to several waterfall methodologies happening within the development life cycle.

Advantages:

- Parallel development can be done using this methodology
- Progress can be measured and easy to change the requirements.
- Risks can be identified easily and can be managed within the iterations.
- With every iteration, working product can be delivered.

Disadvantages:

- System design issues may be happening as not all requirements are gathered at the beginning.
- Major requirements should be identified at the beginning of the project.

3.3 Project Management Approach

Project management is a major component for the successful completion of a software project incorporating user requirements along with cost and time constraints. So, a project management methodology is needed to be selected. PRINCE2 (PRojects IN Controlled Environments) is known as a structured project management methodology. And also, it emphasizes dividing projects into manageable and controllable stages.

Since this system should complete the requirements in a limited time period, project should be organized well. PRINCE2 methodology can be used to ensure a smooth project execution under a fairy detailed monitoring and controlling process. PRINCE2 methodology was selected for this final year project with the above-mentioned facts.

3.4 Research Methodology

There are two main research methodologies such as Quantitative and Qualitative. Qualitative research is known as suitable for getting comprehensive understanding of the problem setting. Qualitative research methodology consists of non-structured methods such as interviews and group discussions and etc. Basically, it gains thorough understanding about the human expertise, behavior, intentions about the problem setting to determine the way people think and feel. Qualitative research methodology uses a subjective approach because researcher is involved. Text based or verbal based data are gathered in qualitative methodology. So, statistical tests and analysis cannot be used. Qualitative research methodology generates hypothesis and theories with the gathered data for quantitative research.

Quantitative research methodology based on methods of natural sciences. It produces hard facts and numerical data. Quantitative research methodology uses rigid and structured methods like online questionnaires, surveys and etc. Quantitative data can be objectively measured with numbers and can be counted. So, statistical tests can be done for analyzing and data can be visualized in charts, figures and etc. Quantitative data can be collected easily and have a large sample size than the qualitative data. Quantitative research can be replicated or repeated.

As this research is based on improving the accuracy of the road surface quality monitoring, there will be much more numerical and statistical analysis work to gather, calculate and configure data. So, Quantitative research methodology will be used in this research.

3.5 Work Breakdown Structure

Work breakdown structure (WBS) is recognized as an illustration of dividing the project deliverables to simpler and manageable tasks. This structure gives a brief overview about what deliverables are to be completed. The WBS for this project can be found in Appendix A.

3.6 Gannt Chart

Gannt chart is an illustration of project schedule which shows project deliverables with the allocated time for each task. Refer Appendix B for the Gannt Chart.

3.7 Monitoring Plan

Regular meetings were conducted with the supervisor of this research to monitor the progress of the project. And also, those meetings were recorded in project log book.

3.8 Chapter Summary

This chapter included details about research methodologies and why the quantitative methodology is used. For this undergraduate research project, Iterative methodology has been selected as the development methodology. Work breakdown structure and Gannt chart were produced to get an idea how the project deliverables are divided and when will be delivered.

Chapter 4 : Requirement Specification

4.1 Chapter Overview

This chapter includes the requirement specification of the proposed system. Identification of relevant stakeholders, requirement elicitation techniques, requirement analysis, use case diagrams were done in this chapter. And also, the identified functional and non-functional requirements of the system also being listed.

4.2 Stakeholder Analysis

4.2.1 Onion Model

The stakeholders who are identified as influencing the proposed system was illustrated by onion diagram. Following figure 4.2.1 describes the stakeholders who are influenced by the proposed road surface monitoring system.

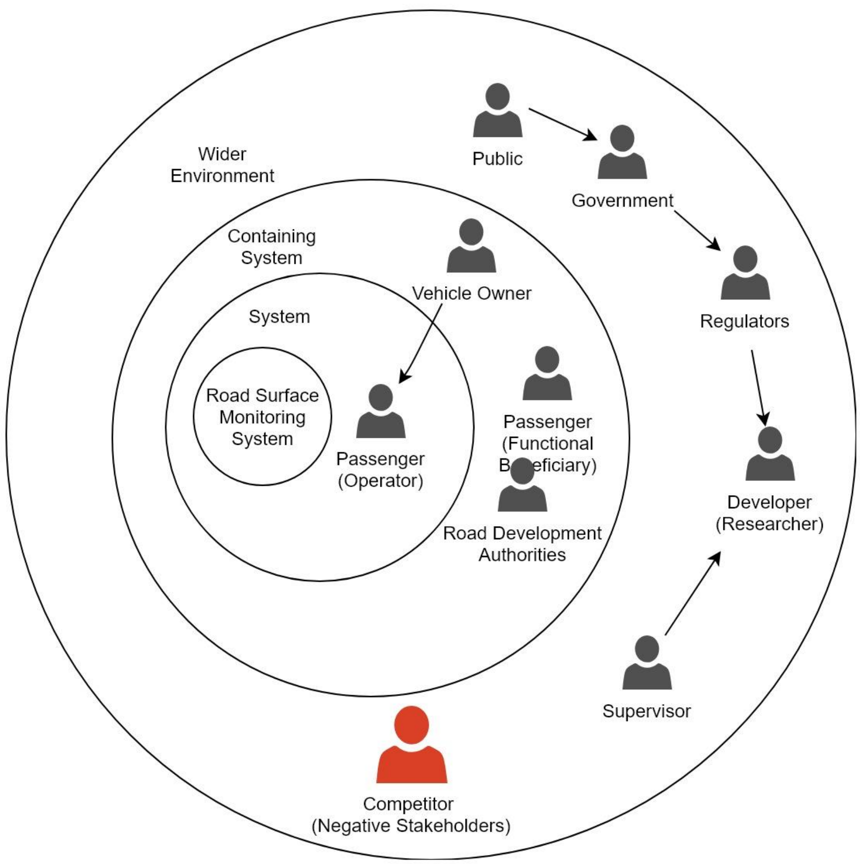


Figure 4.2.1 : Onion Model

4.2.2 Stakeholder Roles

Stakeholder Name	Role	View Point
Passenger	Operator	Data is collected from passenger to further analysis
Passenger	Functional Beneficiary	Passenger can use the mobile application to view the road conditions implemented in a map.
Vehicle Owner	Functional Beneficiary	Vehicle owners can reduce the money spend for fixing the damages happened due to the poor road surface conditions.
Road Development Authorities	Beneficiary	Road development authorities can easily identify the roads which needs immediate maintenance or have to be repaired.
Competitor	Negative Stakeholder	Competitors can develop similar systems.
Developer	Maintenance Operator	Developer have to develop and maintain the system. And also, the developer has to do bug fixing to maintain the quality of the product.

Table 4.2.1 Stakeholder Roles

4.3 Requirement Elicitation

Requirement elicitation is the procedure of discovering and identifying the requirements of a system. Requirement elicitation can be done with the identified stakeholders of the proposed system.

4.3.1 Requirement Elicitation Techniques

Literature Review:

This is one of the strongest techniques in requirement elicitation as it is based on reviewing similar applications developed recently and recently published research papers. Literature reviewing gives a good knowledge about the approaches followed in the previous work carried out. This approach can be used in the proposed system for requirement elicitation as it can collect valuable data and knowledge about the past researches.

Brainstorming:

Brainstorming can be used in collecting several ideas from set of people about the proposed system. Brainstorming is a time-consuming technique as it takes a long time to conduct the sessions. But all the parties have the ability to discuss freely about the scenario. Therefore, all the necessary information can be obtained. But it is hard to bring all suitable experts together to a single place due to their busy schedules.

Online Questionnaire:

Collecting information from large number of people could be obtained by Questionnaires. This is a very successful way of collecting information. As these questionnaires can be shared on social media and other social media platforms, it’s very easy to collect large number of data from road users. But it is limited ability in getting ideas from the participants as their answers are limited to the questions included in the questionnaire.

Formal Interviews:

Formal interviews can be done to collect individual participants answer without depending on another participant’s answers. So, it would be more effective than the collecting the information from a group at once. Interviews can be done as structured and un structured. Unstructured interviews could be more ideal as participant can address their ideas without any limitations. So, unstructured interviews can be done with the road users and gather their ideas and experiences related to the domain.

4.3.2 Requirement Elicitation Techniques Execution

The selected techniques from the above mentioned were executed to collect requirements from the pre identified stakeholders.

- **Literature Review**

  A comprehensive literature review was done associated to the proposed system. This has been done using the recently published research papers on Research Gate, IEEE explorer and other qualified websites. The literature review was done to identify and understand the approaches used in the previous researches related to road surface quality monitoring. Comprehensive literature review was taken around one month to complete.

- **Online Questionnaire**

  An online survey was created and distributed with general public using social media platforms and by emailing. has been conducted to gather data from different road users. The survey was conducted from 01/02/2019 to 14/02/2019. During that time period 103 responses were collected from the participants of the survey.

  The Questions of the survey are added in Appendix C.

- **Formal Interview**

  Two formal interviews were conducted to requirement elicitation. First interview was done with a Senior Software of Virtusa (Pvt.) Ltd to discuss about the project ide and to identify the requirements required for the project. Second interview was done with a tour guide who is travelling throughout the country with his own vehicle.

4.3.3 Requirement Analysis

Requirement analysis was done with the results collected from executing the requirement elicitation techniques.

- **Analysis of questionnaire**

Please refer Appendix C for questions of the survey and Appendix D for the responses collected from the questionnaire. Target response count for this questionnaire was 100 responses and was able to gather 104 responses.

11 questions were included in the questionnaire and they were included to the questionnaire to achieve questionnaire objectives. Below table focusses on the mapping between the questionnaire and the objectives.

No.	Questionnaire Objective	Mapped Question No.
1	Background identification of respondents	1,2,3
2	Identify the consideration about the current problem of the respondents	4,5
3	Identify the opinion of the current problem that the respondents face.	6,7
4	Collect opinions about the proposed system	8,9

Table 4.3.1 Questionnaire Objectives

Since the results of this survey was collected anonymously, the questionnaire objective 1, questions 1,2 and 3 were asked to identify the respondents.

1st question was asked from the participants of the survey to identify their vehicles they are using to travel on roads. Most common vehicle type is considerd as four wheelers with a percentage value of 80.8% from the total participants. Main purpose of this research is to categorize the road surface conditions with the data gathered from smartphones considering the vehicle type of the

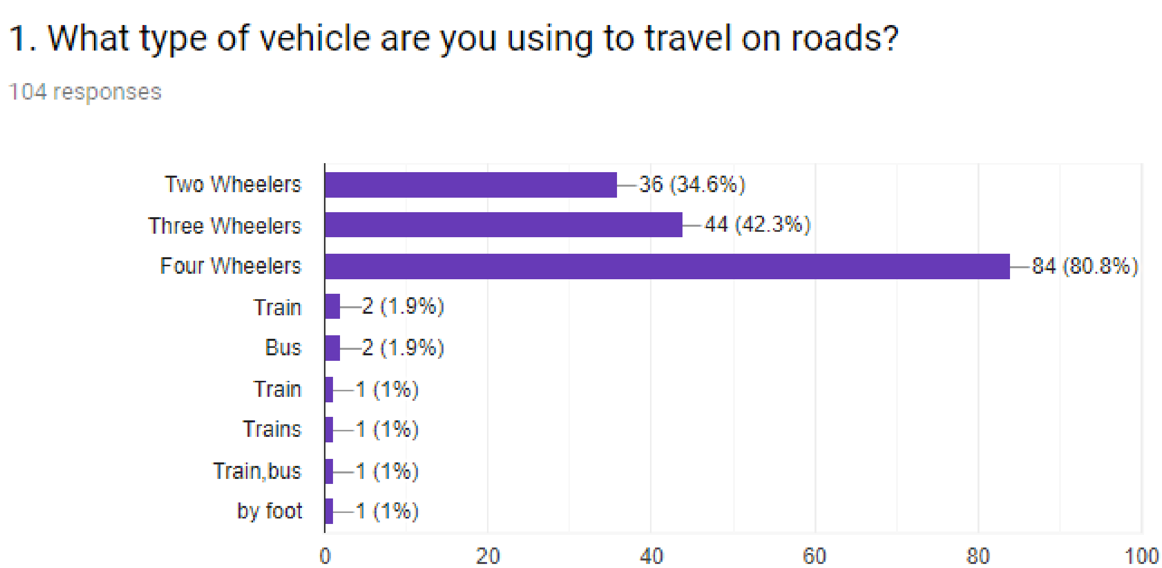


Figure 4.3.1 : Q1 responses

user. For that, most common methods of vehicle types can be identified as two wheelers, three wheelers and four wheelers.

2nd question was asked to identify the respondents' phone type. The question got an answer as all the users are using smartphones. As the proposed system is based on gathering data from the smartphone users, the percentage got for this question is impressive.

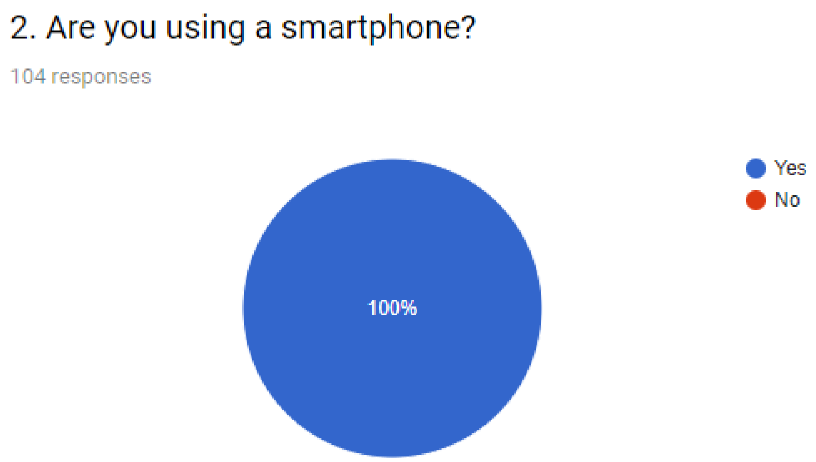


Figure 4.3.2 : Q2 Responses

The next question was asked to identify that the people who took part in this survey were aware of the mobile application such as google maps. Google maps can be used to find places, find routes and etc. The above chart clearly shows that all the users are aware about the such applications and they are using those.

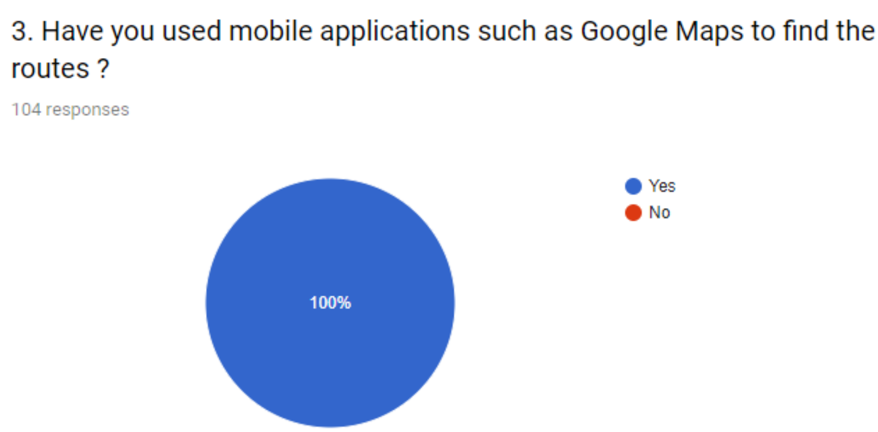


Figure 4.3.3 : Q3 Responses

Question no. 4 was asked to identify the consideration of the participants of the survey about the road surface quality when they are travelling. This question was rated as 97.1% people are like to have a smooth travelling on the road.

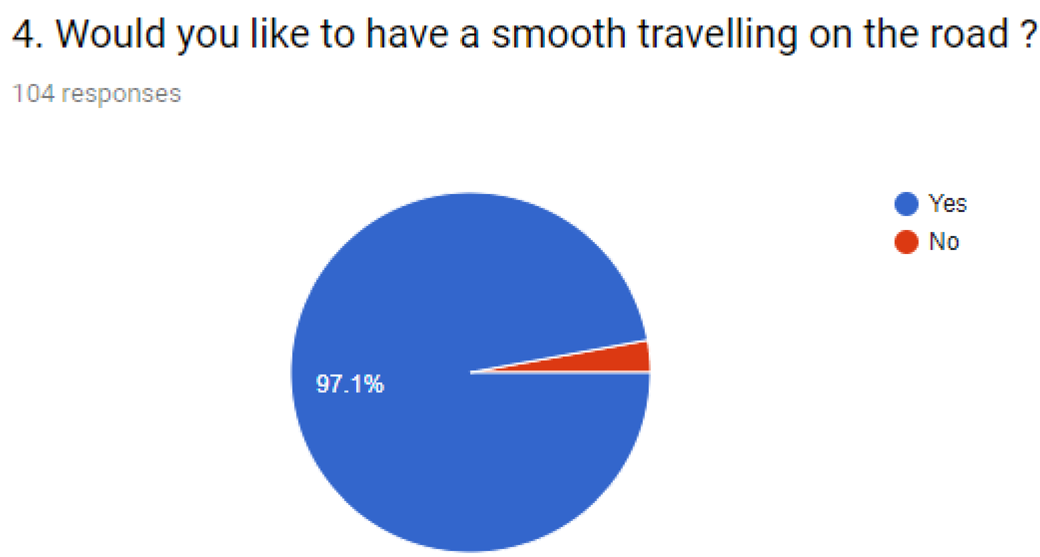


Figure 4.3.4 : Q4 Responses

5th question was asked to find out the users' consideration of the road surface quality, when they are selecting roads. 57.7% from the total participants were marked that they consider about the

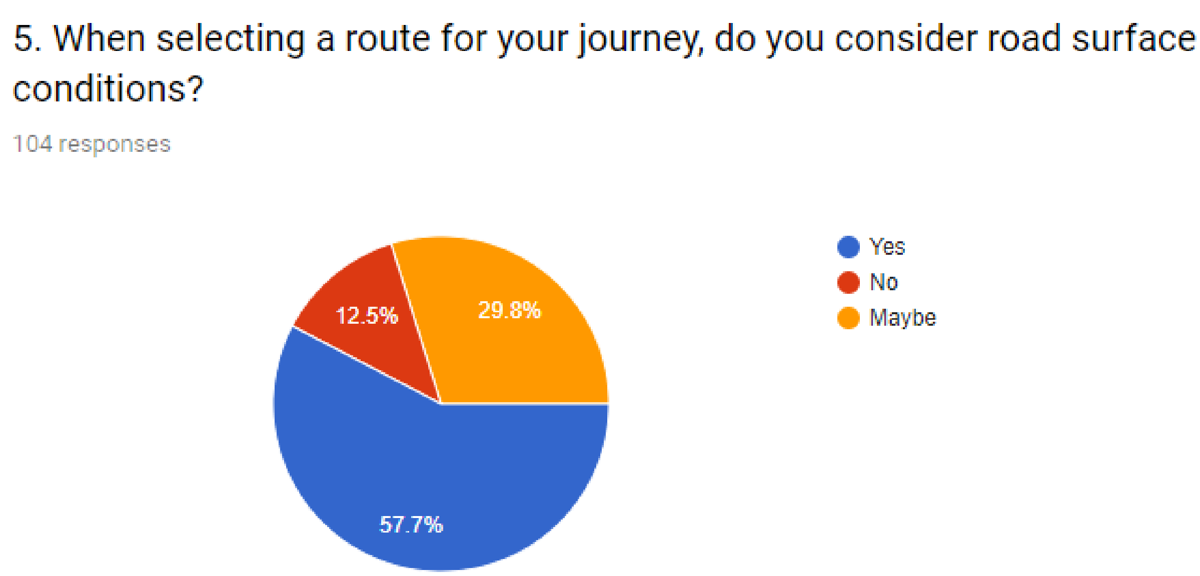


Figure 4.3.5 : Q5 Responses

road surface quality when they are deciding the route and 29.8% were marked as maybe.

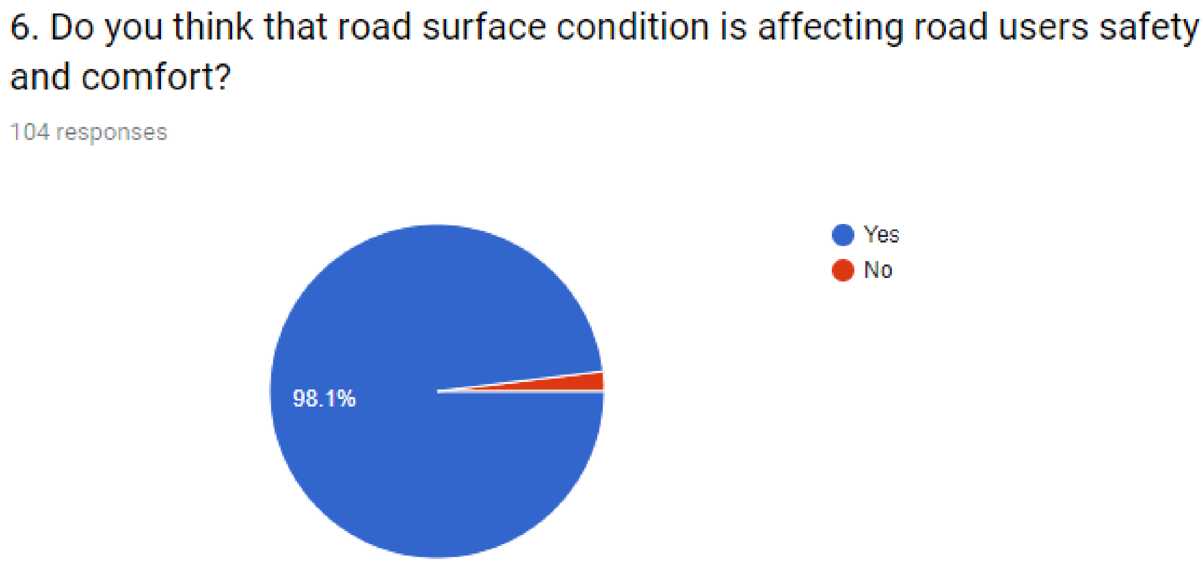


Figure 4.3.6 : Q6 responses

This question was asked to get the opinion of the respondents about the problem. 98.1% people have answered as the road surface conditions has a direct impact of road users' safety and comfort.

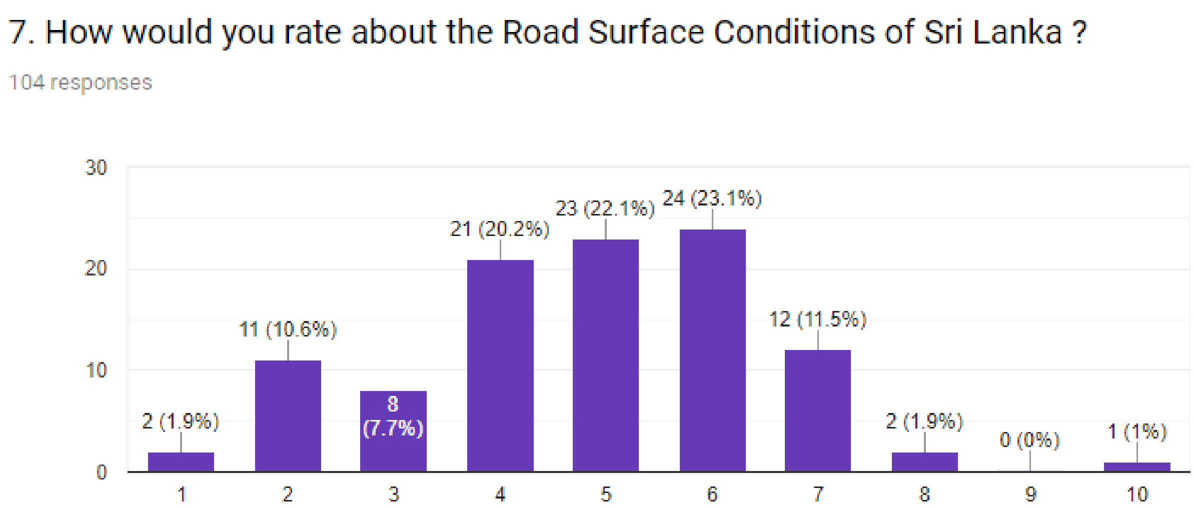


Figure 4.3.7 : Q7 Responses

The 7th question was asked to gather the opinion from the participants of the survey about the road surface conditions of Sri Lanka. Question was given to rate the road surface quality from 1 to 10. 85.6% participants have rated that road surface quality of the Sri Lanka are less than 7. By this rating, can come up with a conclusion that majority of the participants are not satisfied about the road surface quality.

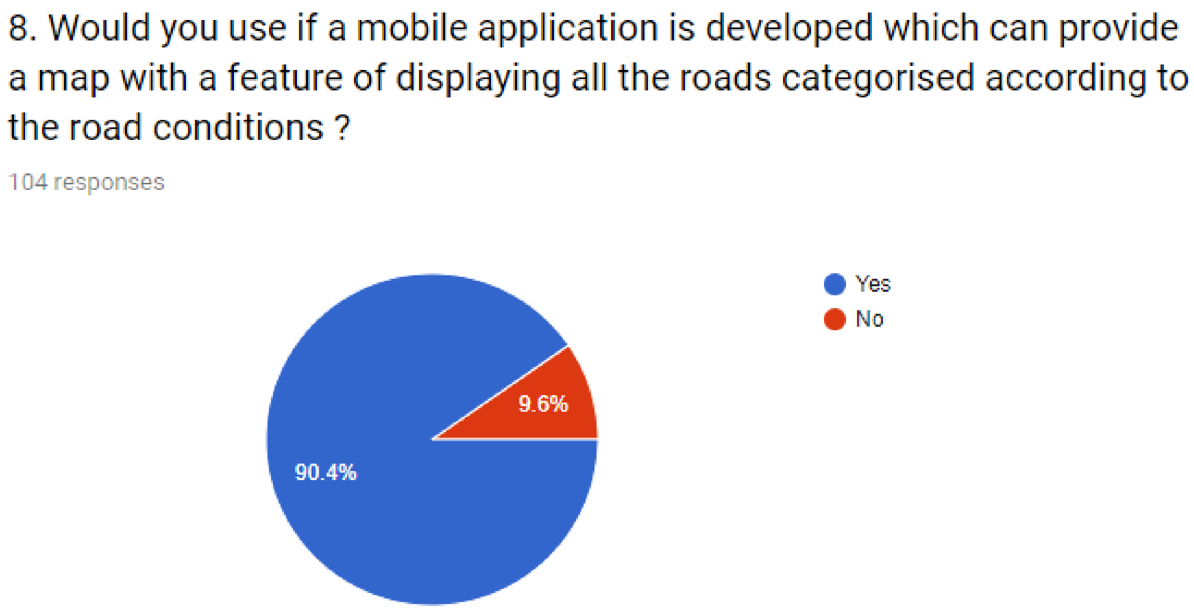


Figure 4.3.8 : Q8 responses

8th and 9th questions were asked to gather the respondents' opinion about the proposed system. 90.4% people have voted that they would like to use a mobile application which can provide a map which displays the roads categorized according to the road conditions.

From the above survey, it was verified that people are more considering about their safety and comfort when they are traveling. And also, the survey was able proven that majority of the people are unsatisfied about the road surface conditions. Most of the users are considering about the road condition when they are selecting the routes to travel. From these results, it is clear that author can continue with the selected scope for this project.

- **Analysis of formal interviews**

The interview conducted with Senior software engineer was mainly focused on discussing the project idea and finalizing the scope of the proposed solution. The interview has conducted for 1 hour.

The second interview which conducted with the travel guide was a discussion about his experiences. He is travelling daily throughout the country with his vehicle. He has mentioned that he is facing troubles due to the poor road surface conditions of the roads. Sometimes he has to take the longer routes to avoid the poor conditioned roads as tourists are more considering about their comfort. And also, he has stated that he has to annually spend money for fix the damages happened due to the road conditions. Requirements for the proposed system were discussed with him and he has made some suggestions to improve the feasibility of the product.

- **Analysis of Literature Review**

Refer the Chapter 2: Literature Review for the analysis of the Literature Review.

4.4 Use Case Diagram

Use case diagrams can be illustrated to represent the requirements of the system diagrammatically. Following figure 4.4.1 shows the use case diagram for this system.

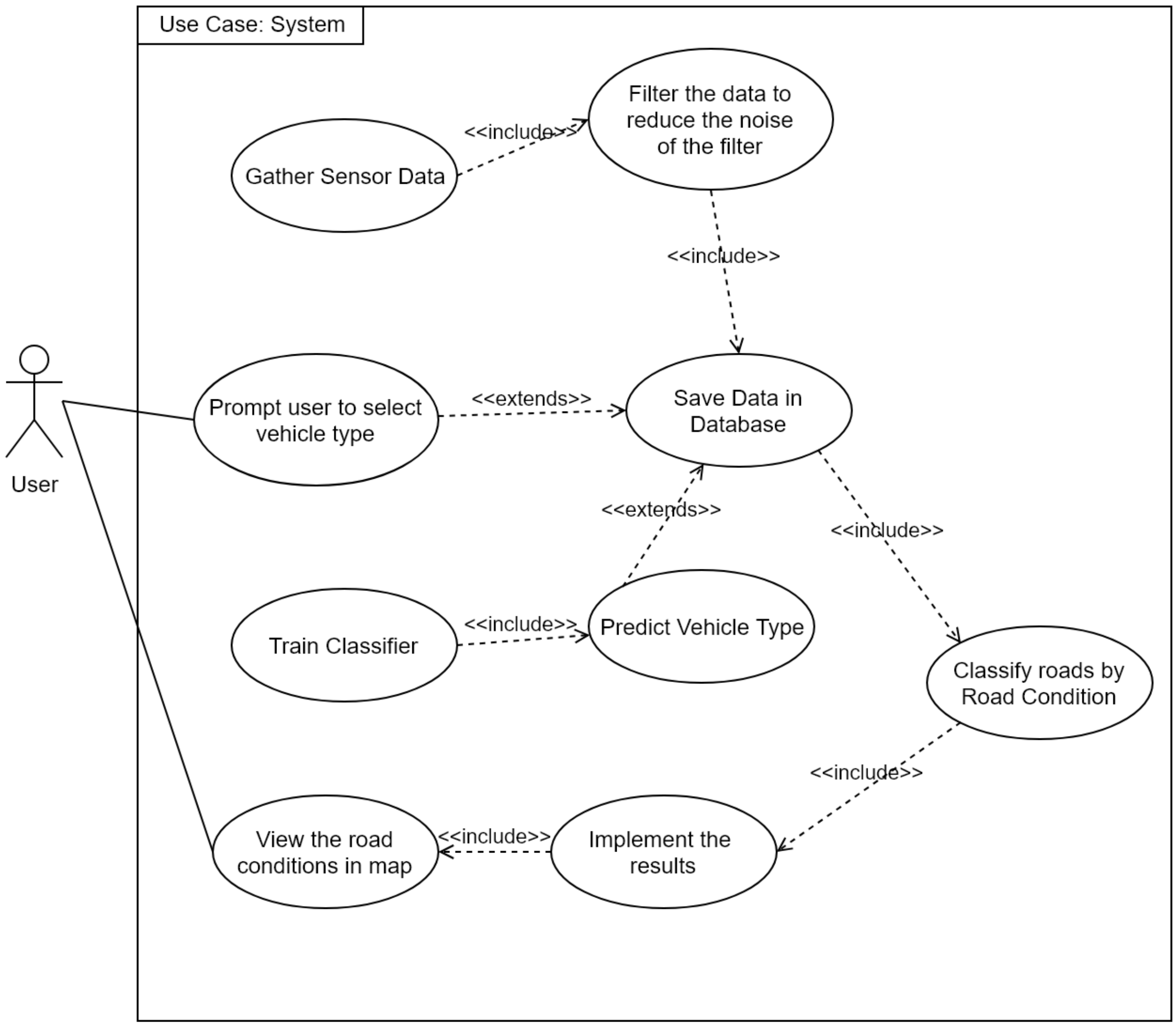


Figure 4.4.1 : Use Case Description

4.5 Use Case Descriptions

Use Case ID:	**UC001**	
Use Case Name:	**Prompt User to Select Vehicle Type**	
Description:	The system will be request to select the vehicle type which the user have travelled.	
Primary Actor:	User	
Pre-Condition:	The machine learning model should not be trained.	
Including:	None	
Extending:	None	
Main Flow:	**User**	**System:**
	2. Select the vehicle type travelled	1. Prompt user to select the vehicle type travelled 3. Save the data
Alternative Flow:	**User**	**System**
	2. Click Cancel	3. Go back to the home screen
Exceptions:	E1. The data gathered from the user will not have the vehicle type 1. The specific data will be deleted from the database.	
Post Condition:	The user selected vehicle type is saved.	

Table 4.5.1 Use Case Description 1

Use Case ID:	**UC002**
Use Case Name:	**View the road Conditions in map.**
Description:	User can view the road conditions categorized in to levels in a map.
Primary Actor:	User
Pre-Condition:	To show the categorized road conditions in the map, each road should have previous data and implemented the relevant results.
Including:	Implement the results.

<table>
<tr><td>Extending:</td><td colspan="2">None</td></tr>
<tr><td rowspan="2">Main Flow:</td><td>User</td><td>System:</td></tr>
<tr><td>1. Open the mobile app.

2. See the map.</td><td>2. Show the map with the categorized road conditions.</td></tr>
<tr><td>Alternative Flow:</td><td colspan="2">None</td></tr>
<tr><td>Exceptions:</td><td colspan="2">E2. No internet connection.
1. Mobile app shows an error message.</td></tr>
<tr><td>Post Condition:</td><td colspan="2">User is presented a map with the categorized road conditions.</td></tr>
</table>

Table 4.5.2 : Use Case Description 2

4.6 Requirements

4.6.1 Functional Requirements

Following requirements have been identified as the functional requirements for the proposed solution and listed in below table 4.6.1. The functional requirements can be prioritized by its effect to the functionality of the proposed system.

- **Critical:** Main functionalities that compulsory to be developed in the proposed solution.
- **Desirable:** Functionalities which adds a value to the system. But these functionalities are not mandatory.
- **Luxury:** These functionalities can add a luxury to the proposed system.

FR No.	Functional Requirement	Priority
1	**Gather Sensor Data** Gather sensor data from the user's mobile phone accelerometer sensor and GPS data.	Critical
2	**Filter the data** Though accelerometer returns signals which are having high noise, relevant filters have to be used to reduce the noise of the accelerometer sensor data.	Critical

3	**Calculate IRI Value** IRI value have to be calculated using the accelerometer sensor data to identify the road condition	Critical
4	**Save data in database** Save the gathered sensor data in the database	Critical
5	**Prompt user to select vehicle type** User have to select the travelled or driven vehicle type for the roads which doesn't have enough data to train the ML model.	Desirable
6	**Train classifier** Train Machine Learning model with the current data in the database.	Critical
7	**Predict vehicle type** Predict the vehicle type if the machine learning model is trained using the previous data in the database.	Critical
8	**Classify roads by road condition** Classify the roads with the usage of predicted or gathered data of the vehicle type and the calculated IRI value.	Desirable
9	**Implement the results** Implement the results of classified roads.	Critical
10	**View the road conditions in map** Users can view the categorized road conditions using the mobile application.	Critical

Table 4.6.1 : Functional Requirements

4.6.2 Non-Functional Requirements

NFR No.	Non-Functional Requirement	Description
1	Accuracy	System should identify the vehicle type of the user and measure the road quality.
2	Performance	As crowd sourcing is used in this system, data is gathered from users in every time. The data id also sending to the back end frequently. So, the performance of the system should be high.
3	Usability	System should be easy to use by the user to identify the road conditions. Data will be visualizing in a map.

4	Compatibility	The android application should be capable of run on different android smart phones.

Table 4.6.2 : Non-Functional Requirements

4.7 Scope Refinement

The requirements gathered from the online surveys, literature reviews and formal interviews were used in refining the scope of this project. The scope of the project was narrowed down to identify two vehicle types in this research. So, the implemented system was able to identify two wheelers and four wheelers using the machine learning model. This narrow down was done as project has to be completed within a limited time frame.

4.8 Activity Diagram

Figure 4.8.1 show the Activity diagram of this system.

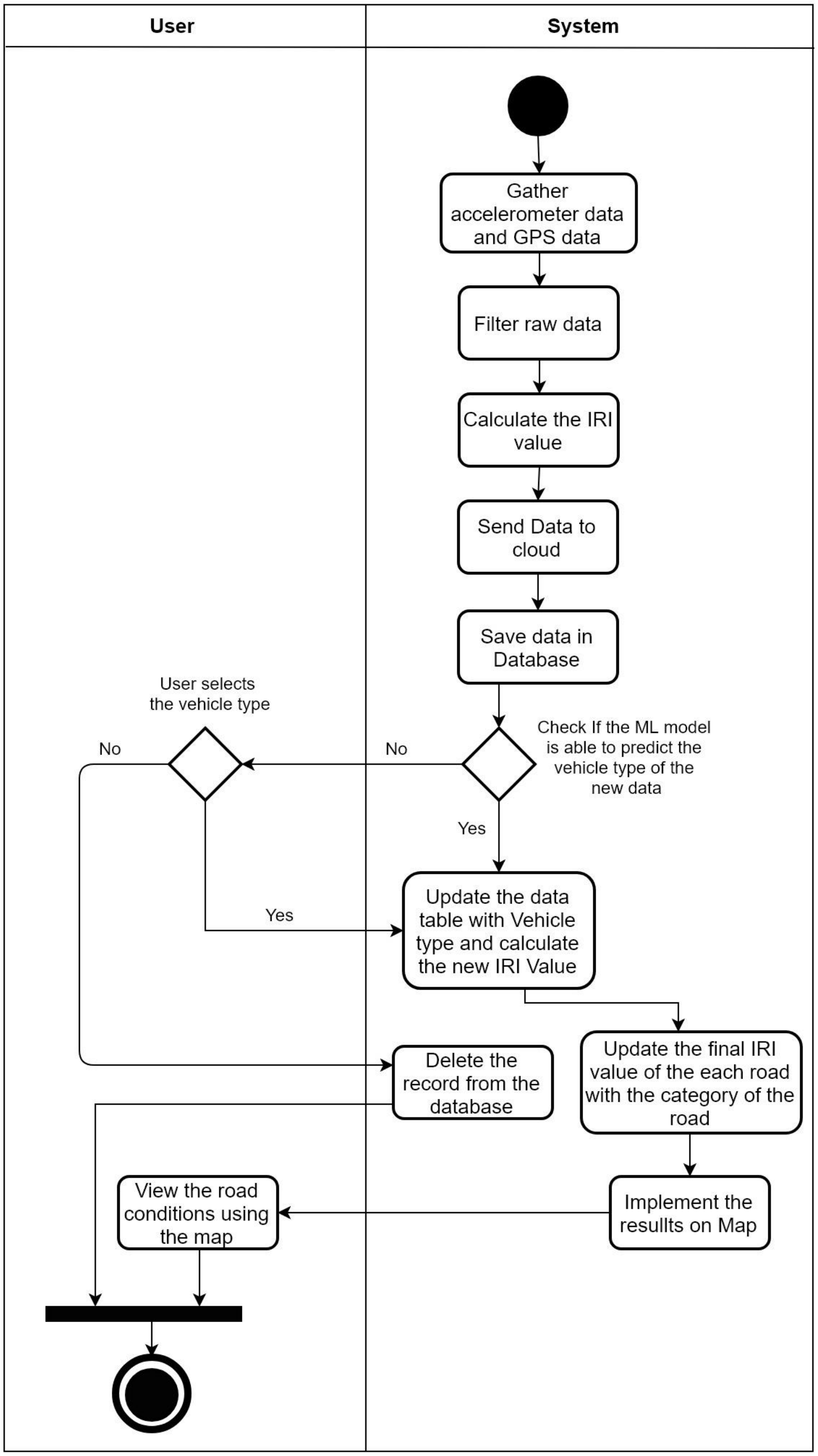


Figure 4.8.1 : Activity Diagram

4.9 Chapter Summary

This chapter included about identifying the stake holders related to this proposed system domain. The analysis of the requirement gathering techniques in order to gather functional requirements and non-functional requirements were done. And also, the chapter included a use case diagram and use case diagrams for the proposed system. This chapter also described and listed about the functional and non-functional requirements of the proposed system.

Chapter 5 : Design Specification

5.1 Chapter Overview

Previous chapter was about identifying the functional and non-functional requirements with the relevant stakeholders of the system. This chapter focusses about the design and the architecture of the proposed system to outline the approach to achieve the target through a computational program. Class diagrams and sequence diagrams will be illustrated to elaborate the complete design of the proposed system.

5.2 Design Methodology

Object Oriented Analysis Design (OOAD) approach were taken as the design methodology of this project. OOAD was taken as this methodology managing and controlling the complexity of the software because of the modularity approach. As this project has to be completed within limited time, OOAD saves the time because of its simplicity.

5.3 High Level Design

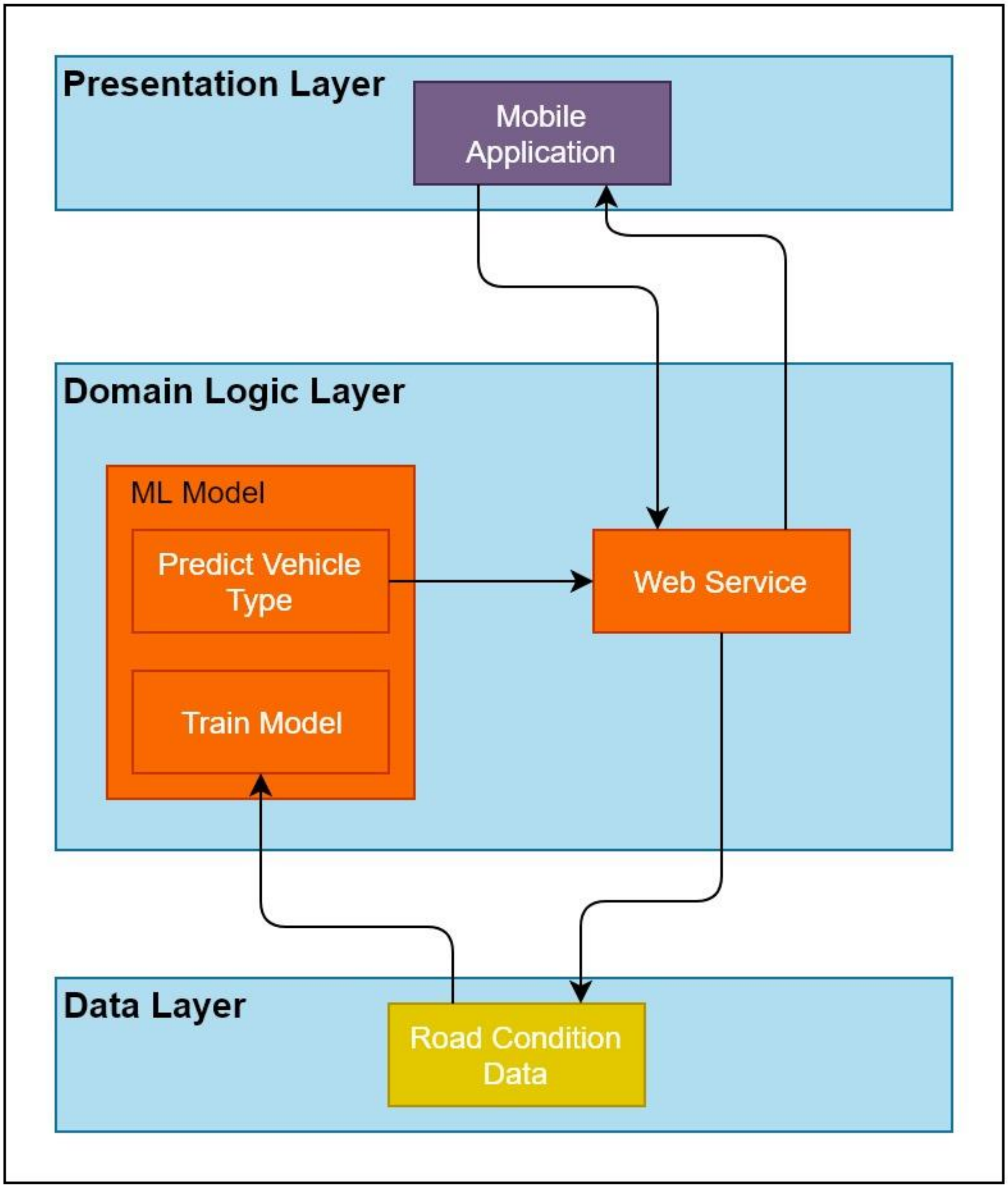


Figure 5.2.1 : High level Architecture

5.3.1 Presentation Layer

This is the layer where user interacts with the system. This is a mobile application which user can easily interact. User can view a map view which shows the road conditions of the roads with separate colors.

5.3.2 Domain Logic Layer

Domain logic layer is the most important layer in this system. This layer receive data from the mobile application and process data by saving data in database. Finally, this layer sends back the data back to the presentation layer. This layer consists of two main models.

- **ML Model:**

ML model is trained by the data in the data layer. This model is capable of predicting the vehicle type the user travelled and sending data to the Web Service.

- **Web Service:**

Web service received data from the application layer. Predicted data from the ML model also sending to the data layer using the web service. The data received from the application layer also sending to the data layer using the web service. Web service sends the data to the application layer also.

5.3.3 Data Layer

Data layer stores the data in this system. All the accelerometer sensor data gathered from the application layer will be stored in this layer. The data in this layer will be used to ML model to predict the vehicle type and categorize the roads by the road conditions. And those data will be sent back to the web server.

5.4 Design Diagrams

5.4.1 Class Diagrams

Figure 5.4.1 shows the class diagram of the web service of this system.

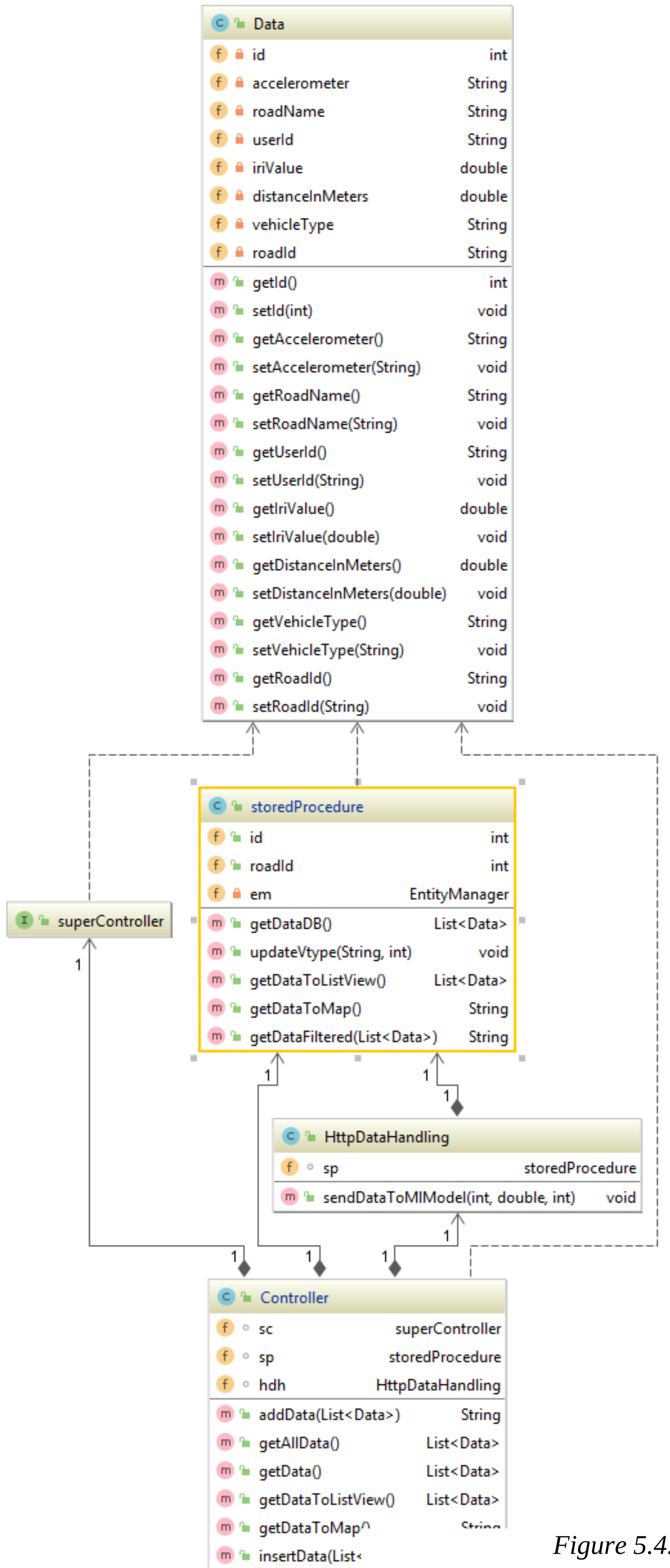


Figure 5.4.1 : Class Diagram

5.4.2 Sequence Diagrams

Following figures 5.4.2, 5.4.3 are the sequence diagrams illustrated for this project.

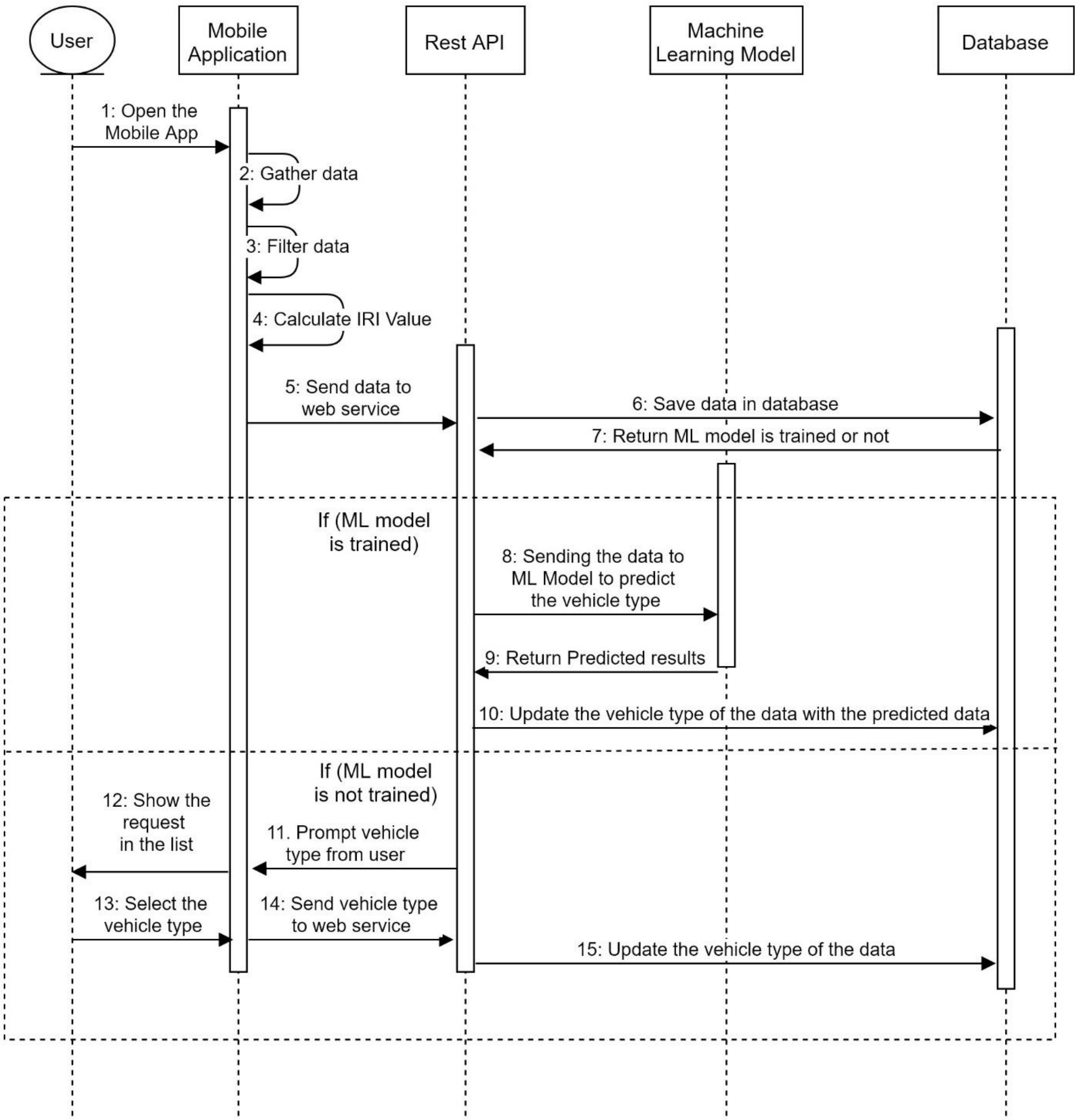


Figure 5.4.2 Sequence Diagrm 1

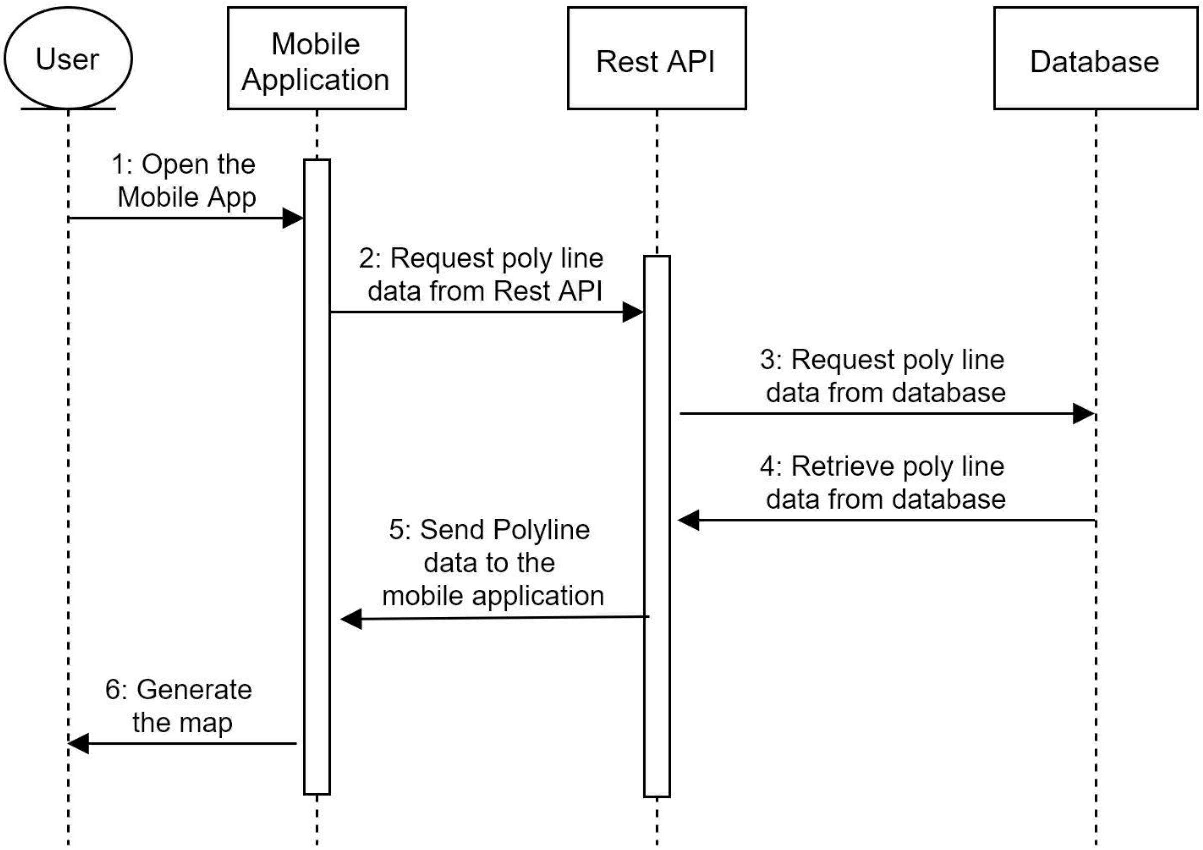


Figure 5.4.3 : Sequence Diagram 2

5.5 Chapter Summary

This chapter included the diagrammatically illustrated explanations of the selecting the reasons for the design methodology of this project. First the design methodology selection was highlighted in this system. Then the High-Level Architecture Diagram was illustrated with the details. Low level design diagrams such as Class diagram and the Sequence Diagram were also illustrated in this chapter.

Chapter 6 : Implementation

6.1 Chapter Overview

This chapter focuses on the how the implementation has been done adhering to the designs mentioned in the previous chapter. This chapter includes the technology and tools selection to implement this system. Used technologies and algorithms and issues faced when doing the implementation of the prototype and how those issues were outcome will also be discussed in this chapter. Relevant code snippets of the implemented solution have been used to describe and give a clear picture of this stage.

6.2 Implementation Plan

According to the high-level architecture diagram which has mentioned in section 5.3, this system is having 3 main layers such as Presentation layer, Domain Logic Layer and Data layer. The presentation layer consists of a mobile application which the user can mainly interacts with the system. Domain Logic Layer is having two models such as Machine Learning Model and Web Service. This layer is used to process the data and pass the data between presentation layer and the data layer. The Data Layer is having a database to store the data processed by the domain logic layer.

6.3 Technology and Tools Selection

6.3.1 Mobile Application

Mobile application development can be done as Android development, iOS development, Windows development and Native development. The scope of this project has been limited to develop the mobile application as an Android Application which can be used by the android users.

- **Selection of Programming Language**

Android application development is can be done by using Java and Kotlin. Among those two programming languages, Java has selected because of the lack of learning resources related to the Kotlin Language.

- **Selection of the IDE**

There are most commonly used 3 IDEs for development of the Mobile Application development with Java such as, Android Studio, IntelliJ Idea and Eclipse. By the time, google has announced moving away from the Eclipse IDE from the android development because the plugins are not adopting frequently and some plugins are not bug fixed and maintained. IntelliJ also not supporting some plugins which supports by the Android Studio. Android studio is the official software developed to do Android development by google. So, the integration of google services using Android studio is much easier than the other IDEs. Though this project is using google services like location and activity recognition, Android studio has been selected as the IDE to develop the mobile application.

6.3.2 Machine Learning Model

- **Selection of Programming Language**

Python, R, Java and C++ can be identified as the most common programming languages used in programming machine learning models. Below table 6.3.1 is a brief description about those programming languages.

Programming Language	**Description**
Python	Python is an interpreted, general-purpose, high-level programming language with features to facilitate data analysis and visualization.
R	R is a programming language which works on free software environment for statistical computing supported by the R Foundation for Statistical Computing.
Java	Java is a common object-oriented programming language which has a framework called WEKA, which used to apply machine learning algorithms directly from the java code.
C++	C++ is a general-purpose, object-oriented programming language which provides facilities for low-level memory manipulation.

Table 6.3.1 : Selection of Programming Languages

Python was selected for the development of the Machine Learning Model as python language supports diverse activities like web development, artificial intelligence and data analytics. And also, python is a very user-friendly language and python has huge number of machine learning libraries which can be reduce the effort of the developer.

- **Selection of the IDE**

Pycharm is selected as the IDE among from VS Code, Spyder, PyDev because It helps to be more productive and gives assistance to the developer. And also, Pycharm is integrated with the features such as debugging, testing, remote deployments and tools of the database.

6.3.3 Web Service

- **Selection of Programming Language**

Spring Boot Framework has been selected as the programming language to develop the webservice of the implemented system. Spring Boot is an open source Java based framework for developing the web services. Spring boot is used in this project as it offers an annotation-based spring application which everything is auto configured and no manual configurations are needed. So, using spring boot development, time can be minimized and easy to deploy the application.

- **Selection of the IDE**

Spring Tool Suite, which is an Eclipse based IDE which is customized for the development of the spring applications is selected as the IDE. Spring Tool Suite presents as a ready-to-use distribution of the latest Eclipse releases with the Spring IDE components pre-installed.

6.3.4 Database

Microsoft SQL Server is selected to store the data of this project. SQL databases are called as relational database management system. SQL queries can be used to retrieve large amounts of records more efficiently and quickly. And also, SQL is supporting complex queries and can be easily sync with the cloud services. So, SQL server databases are used in storing data of this system and the database is created as a Microsoft Azure SQL database. Microsoft SQL Server Management Studio was used to querying the database from on promises and syncing with the Azure SQL Database.

6.4 Implementation of the Core Functionalities

6.4.1 Implementation of Gather Data

Gathering data from the users is the first part of the system. The mobile application which is developed in java is capable of gather the data from the users without any interaction from the user with the mobile application. The acceleration data and the longitude and latitude data of the mobile phone are the data to be gathered from the users' mobile application.

Majority of android devices are having built in sensors which can measure motion, orientation and etc. Android allows to use the raw data of the sensors to use in the mobile applications. So, android provides the SensorManager to get the sensor data from the accelerometer. After declaring the SensorManager, the object of the sensor Sensor class has to be initiated calling the getDefaultSensor() method.

```
SensorManager sensorManager;
sensorManager = (SensorManager) getSystemService(Context.SENSOR_SERVICE);

Sensor accelerometer;
accelerometer = sensorManager.getDefaultSensor(Sensor.TYPE_ACCELEROMETER);
sensorManager.registerListener( listener: MainActivity.this, accelerometer,
        SensorManager.SENSOR_DELAY_NORMAL);
```

Figure 6.4.1 Initializing the Accelerometer sensor

The accelerometer raw data can be got from the onSensorChanged() method which override by the register listener. onSensorChanged() returns accelerometer reading in a sample rate of 20ms. Sensor.TYPE _ACCELEROMETR is returning values with the SI unit ms^{-2}.

- values[0] : acceleration force on the x axis with gravity
- values[1] : acceleration force on the y axis with gravity
- values[2] : acceleration force on the z axis with gravity

LocationManager class in android provide the access to the system location services. The data from the location class consists of Longitude, Latitude data of the current geographical location.

```
LocationManager locationManager;
locationManager = (LocationManager) getSystemService(Context.LOCATION_SERVICE);
if (ActivityCompat.checkSelfPermission( context: this, Manifest.permission.ACCESS_FINE_LOCATION)
        != PackageManager.PERMISSION_GRANTED && ActivityCompat.checkSelfPermission
        ( context: this, Manifest.permission.ACCESS_COARSE_LOCATION) != PackageManager.PERMISSION_GRANTED) {
    return;
}
locationManager.requestLocationUpdates(LocationManager.GPS_PROVIDER,  minTime: 0,  minDistance: 0,  listener: this);
```

Figure 6.4.2 Initializing the location services

6.4.2 Implementation of Filtering the Data

Though the raw sensor data gathered from the accelerometer sensor is having a heavy noise and the sensor data is affected with the gravity force of the earth, the data has to be filtered. First, the data is filtered using a low pass filter to ignore the signals that are shorter than the time constant and to alter the signals that are much longer than the time constant. More details about this filter have explained in section 2.4 of this document.

```
static final float ALPHA = 0.01f;

protected float[] lowPass( float[] input, float[] output ) {
    if ( output == null ) return input;

    for ( int i=0; i<input.length; i++ ) {
        output[i] = output[i] + ALPHA * (input[i] - output[i]);
    }
    return output;
}
```

Figure 6.4.3 Applying low pass filter to the data

The data which have filtered using low pass filter were further filtered by a high pass filter to remove the gravity force. Following figure shows how the high pass filter was implemented.

```
final float alpha = 0.8f;

gravity[0] = (alpha * gravity[0] + (1 - alpha) * gravSensorVals[0]);
gravity[1] = (alpha * gravity[1] + (1 - alpha) * gravSensorVals[1]);
gravity[2] = (alpha * gravity[2] + (1 - alpha) * gravSensorVals[2]);

linear_acceleration[0] = gravSensorVals[0] - gravity[0];
linear_acceleration[1] = gravSensorVals[1] - gravity[1];
linear_acceleration[2] = gravSensorVals[2] - gravity[2];
```

Figure 6.4.4 Applying high pass filter to data

6.4.3 Implementation of Calculating the IRI value

With using the filtered accelerometer sensor data and the GPS sensor data, IRI value could be calculated. IRI calculation can be done using the equation of IRI = Total Vertical Vibrations (m)/ Distance Travelled (km). The Total vibrations can be calculated using the accelerometer sensor data with the $S = 1/2at^2$ equation. This equation gives the length of the vertical movement (S) in meters by using the accelerometer sensor data (a) and the sample rate of the accelerometer (t).

```
vibration = Math.abs(0.5 *  zValueRounded * (timeInterval*timeInterval)) ;
vibrationTotal = vibrationTotal + vibration;
```

Figure 6.4.5 Calculate Vibration

The total of these vertical movements has to be divided by the distance travelled calculated using the longitude and latitude data. For the distance calculating, distanceTo() in location class can be used. It returns the distance between the current location and the last location in meters.

```
distanceInMeters += currLocation.distanceTo(lastLocation);
lastLocation = currLocation;
```

Figure 6.4.6 Calculate Distance

Finally, the IRI calculation can be calculated using the calculated distance and the total of the vertical movements occurred within that distance.

```
double iriValue = vibrationTotal / (distanceInMeters/1000);
```

Figure 6.4.7 Calculate IRI

6.4.4 Implementation of Save Data in Database

As this system is a crowdsourcing system, the data is collected by many users. A single road is taken as a one unit and save the data relevant to the road. So, the streaming of the continuous data from the mobile application is saved separated according to the roads in the database.

To identify the road, reverse geocoding was used. Reverse geocoding is a method which used to get a readable address name from the longitude and latitude data.

```
private class GeocoderHandler extends Handler {
    @Override
    public void handleMessage(Message message) {
        String locationAddress;
        switch (message.what) {
            case 1:
                Bundle bundle = message.getData();
                locationAddress = bundle.getString( key: "address");

                break;
            default:
                locationAddress = null;
        }

        AddressName = locationAddress.toString();
    }
}
```

Figure 6.4.8 Get address Name

Checking about the street name changing is done inside the onSensorChanged() method of the accelerometer sensor. So, each time accelerometer gives a new data, the total vertical movements occurred due to vibrations and the distance travelled is calculated. The sendData() method that the data sent to the web service is also called when the street name is changed.

```java
if(AddressName != null) {
    if (AddressName == "") {

    } else if ((AddressName.equals(oldAddressName)) ||
            (oldAddressName.equalsIgnoreCase( anotherString: "empty"))) {

        vibrationTotal = vibrationTotal + vibration;

        if(lastLocation == null){
            lastLocation = currLocation;
        }
        distanceInMeters += currLocation.distanceTo(lastLocation);
        lastLocation = currLocation;

    } else if (!AddressName.equals(oldAddressName)) {
        sendData();

        vibrationTotal = 0.00;

        lastLocation = null;
        distanceInMeters = 0.00;
    }
    oldAddressName = AddressName;
}
```

Figure 6.4.9 Checking for the address

The calculation of IRI value for the relevant street is done inside the sendData() method. And it creates a JsonArrayList and send it to the Submit() method. And the submit method post the data to the Spring Boot Web Service which is deployed to the google app engine.

```java
private void sendData() {
    currentTime = new SimpleDateFormat( pattern: "yyyy-MM-dd-HH-mm-ss").format(new Date());

    double iriValue = vibrationTotal / (distanceInMeters/1000);

    String data = "[{ " +
            "\"userId\"" + ":" +  "\"" + id + "\","+
            "\"roadName\"" + ":" + "\"" + oldAddressName + "\" ,"+
            "\"latlng\"" + ":" + "\"" + latlng + "\" ,"+
            "\"iriValue\"" + ":" + "\"" + iriValue + "\" ,"+
            "\"distanceInMeters\"" + ":" + "\"" + distanceInMeters +  "\" ,"+
            "\"logDate\"" + ":" + "\"" + currentTime + "\"" +
            "}]";

    Submit(data);
}
```

Figure 6.4.10 sendData()

```java
@PostMapping("/insertData")
public void insertData(@RequestBody List<Data> data) {
    String test = sp.getDataFiltered(data);

    if (test.equals("1")) {
        try {
            hdh.sendDataToMlModel(sp.roadId, data.get(0).getIriValue(), sp.id);
        } catch (ClientProtocolException e) {
            e.printStackTrace();
        } catch (IOException e) {
            e.printStackTrace();
        } catch (JSONException e) {
            e.printStackTrace();
        }
    }
}
```

Figure 6.4.11 Spring Boot Web Service

The data sent by the mobile application to the web service were added to a List in the Controller class. And the data were passed to a repository and values are set to the parameters for sending data to the data base using stored procedures.

```java
@SuppressWarnings("unchecked")
public String getDataFiltered(List<Data> data){
    StoredProcedureQuery query = em.createNamedStoredProcedureQuery("insertData");
    query.setParameter("latlng",data.get(0).getAccelerometer());
    query.setParameter("roadName",data.get(0).getRoadName());
    query.setParameter("userId",data.get(0).getUserId());
    query.setParameter("iriValue",data.get(0).getIriValue());
    query.setParameter("distanceInMeters",data.get(0).getDistanceInMeters());

    query.execute();

    id = (int) query.getOutputParameterValue("id");
    roadId = (int) query.getOutputParameterValue("id");
    return (String) query.getOutputParameterValue("result");
}
```

Figure 6.4.12 Spring Boot Web Service

```java
@NamedStoredProcedureQuery (name = "insertData", procedureName = "p_insertData", parameters = {
        @StoredProcedureParameter(mode = ParameterMode.IN, name = "latlng", type = String.class),
        @StoredProcedureParameter(mode = ParameterMode.IN, name = "roadName", type = String.class),
        @StoredProcedureParameter(mode = ParameterMode.IN, name = "userId", type = String.class),
        @StoredProcedureParameter(mode = ParameterMode.IN, name = "iriValue", type = double.class),
        @StoredProcedureParameter(mode = ParameterMode.IN, name = "distanceInMeters", type = double.class),
        @StoredProcedureParameter(mode = ParameterMode.OUT, name = "result", type = String.class),
        @StoredProcedureParameter(mode = ParameterMode.OUT, name = "id", type = int.class),
        @StoredProcedureParameter(mode = ParameterMode.OUT, name = "roadId", type = int.class)
})})
```

Figure 6.4.13 Spring Boot Web Service

When inserting the data from the stored procedure in the database, the existence of the road in the roads table is checked. If the road name exists in the roads table road id is assigned to the variable. Is the road is not existing in the database, the data is inserted to the roads table. Then, the data is inserted to the data table and Stored procedure is returning whether that vehicle type can be predicted using machine learning or not.

```
CREATE PROCEDURE [dbo].[p_insertData] @userId VARCHAR(300), @accelerometer NVARCHAR(MAX),
            @roadName VARCHAR(200), @iriValue VARCHAR(100), @distanceInMeters VARCHAR(300),
            @vehicleType VARCHAR(30),  @result VARCHAR(5) OUTPUT, @id INT OUTPUT, @roadId INT OUTPUT
AS

SET @id  =  (SELECT MAX(id) FROM data) + 1;
DECLARE @date datetime =  getdate();

IF EXISTS (SELECT RoadId FROM roads WHERE RoadName = @roadName)
BEGIN
    SELECT @roadId = RoadId FROM Roads WHERE RoadName = @roadName;
END
ELSE
BEGIN
    INSERT INTO Roads (RoadName) VALUES (@roadName)
    SELECT @roadId = RoadId FROM Roads WHERE RoadName = @roadName;
END

INSERT INTO data (id, user_id, accelerometer,road_name, log_date,iri_value,distance_in_meters,vehicle_type,road_id)
VALUES (@id, @userId, @accelerometer, @roadName, @date, @iriValue, @distanceInMeters, @vehicleType, @roadId)


SELECT @result = CASE WHEN EXISTS(SELECT * FROM data WHERE (SELECT COUNT(id) FROM data
                                  WHERE road_id = @roadId) > 30
                                  AND (SELECT COUNT(id) FROM data
                                  WHERE vehicle_type = @roadId) > 30
                                  )
THEN CAST(1 AS BIT)
ELSE CAST(0 AS BIT) END

GO
```

Figure 6.4.14 data inserting stored procedure

6.4.5 Implementation of Train Classifier and Predict Vehicle Type

The main purpose of the Machine Learning Model is to predict the vehicle type when the road and the IRI value was given. So, K-Nearest Neighbor algorithm was selected to do the classification. Steps followed in the KNN classifier can be listed as below.

- **Load the dataset**
- **Determine the value for K**
- **For test data:**
    - **Calculate Euclidean distance to all data points from the test data**

  - **Store the Euclidean distances in a list and sort it in an ascending order**
  - **Choose the first K number of points**
  - **Assign a class to the test data point based on the majority of classes present in the chosen points**
- **End**

Road id and the IRI values was the features selected as the features used in the prediction of the vehicle type of the user. X axis represents the road id and Y axis represents the IRI value of the following graph. Following places are pointed with a sample data set. Data is labeled with the vehicle type according to the graph. So, when vehicle type is predicting to the new test data, KNN classifier looks for the closest neighbor's vehicle type label according to the above-mentioned steps and return the result.

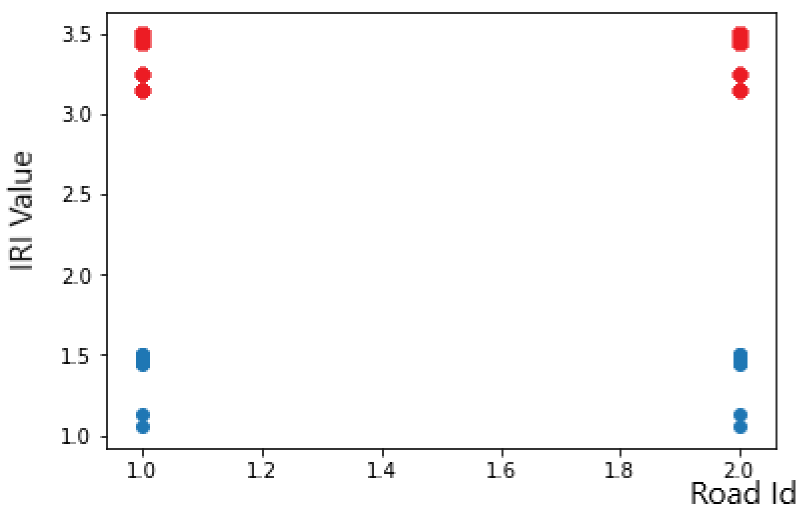


Figure 6.4.15 Sample Data

Two methods were implemented in the Machine Learning Model. First method is for train the data. Since the selected KNN algorithm is a lazy learning algorithm, the data training method was implemented to train the data frequently for retraining the algorithm with the latest data. Since the roads are getting damaged with the time, retraining the algorithm was essential. The latest data was retrieved from the database to retrain the model. Below code snippet is related to the training function in the Python Flask App.

```
@app.route("/train/",methods=['POST'])
def train():
    conn = pyodbc.connect('Driver={SQL Server};'
                          'Server=fyp2019.database.windows.net;'
                          'Database=fyp;'
                          'UID=test;PWD=login@123')
    cursor = conn.cursor()
    sql = "exec p_getAllDataWithVType"
    df = pd.read_sql(sql, conn)
    X = df.drop(['vehicle_type'], axis=1)
    y = df['vehicle_type'].values
    #split dataset into train and test data
    X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2)
    # Create KNN classifier
    knn = KNeighborsClassifier(n_neighbors = 3)
    # Fit the classifier to the data
    knn.fit(X_train,y_train)
    joblib.dump(knn, 'fypml.pkl')

    return jsonify('trained')
```

Figure 6.4.16 Training Method of ML

The second method is for predicting the vehicle type. According to the section 6.4.4, after the data was inserted to the data, stored procedure is returning whether that database is having sufficient amount of data to predict the vehicle type. If there is enough data and Machine learning model is trained for the relevant road, predict function of the Python Flask App is called. It is predicting the vehicle type of the user with the given data.

```
@app.route("/predict/",methods=['POST'])
def predict():
    data = [request.get_json()]
    knn_from_joblib = joblib.load('fypml.pkl')

    # Use the loaded model to make predictions
    preds=knn_from_joblib.predict(data)
    pred_list = preds.tolist()
    return jsonify(pred_list)
```

Figure 6.4.17 Predicting Method of ML

6.4.6 View the road conditions in map

Implemented results have shown in the map to the user. So, Google Maps API was used for that because it's easy to integrate with the android applications and Google Maps API provide a feature to draw set of line segments on the map called Polylines.

When the user opens the application, the application loads the map with the data sent by the database.

```
private void drawPolyLines(String data) {
    System.out.print(data);
    String[] dataList = data.split( regex: ", ");
    List<LatLng> latLngList = new ArrayList<LatLng>();
    PolylineOptions polyOptions = new PolylineOptions();
    polyOptions.clickable(true);

    for(String row : dataList) {
        row.split( regex: " ");
        polyOptions.add(new LatLng(Double.parseDouble(row.split( regex: " ")[2]), Double.parseDouble(row.split( regex: " ")[1])));
    }
    Polyline polyline1 = mMap.addPolyline(polyOptions);
    polyline1.setTag("A");

    mMap.setOnPolylineClickListener(this);
  }
```

Figure 6.4.18 Drawing Polylines

6.5 User Interface

User interface of this system is designed to interact with the mobile application easily. User interaction is very less in this system. Scenarios when user interacts with the system are viewing the map and selecting the vehicle type if the vehicle type is requested from the user. Following figure 6.5.1 is the main user interface of the system.

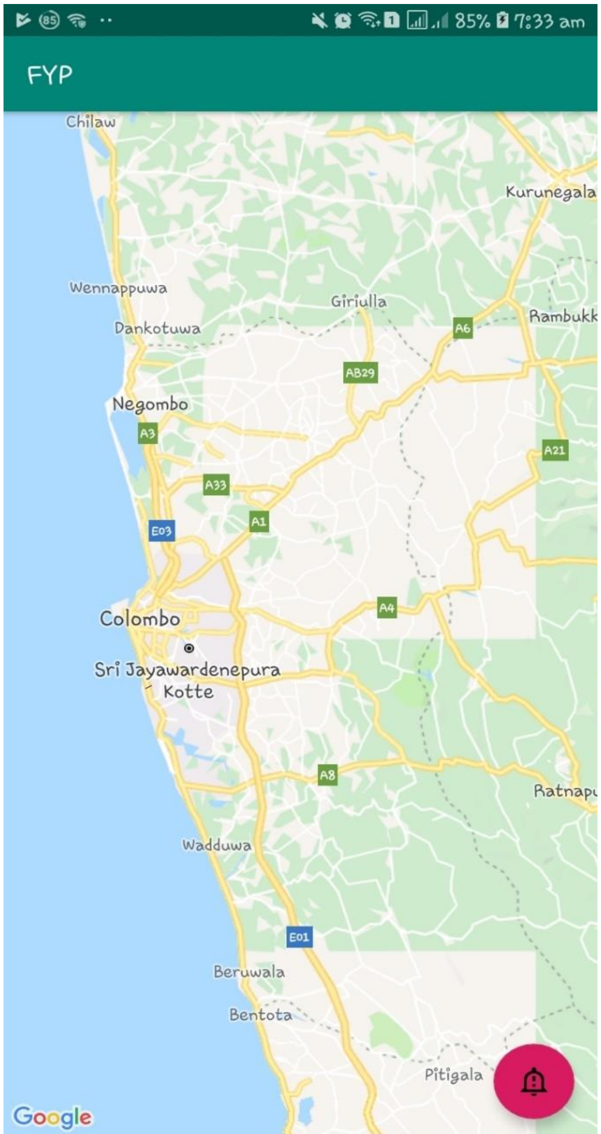


Figure 6.5.1 User Interface

6.6 Chapter Summary

This chapter included a description about the selecting of the technologies and tools used in the implementation of this project. And also, the implementation of the core functionalities of the system were elaborated with the relevant code snippets. Screen shot of the User interface also attached in this chapter.

Chapter 7 : Testing

7.1 Chapter Overview

This chapter focuses on the testing of the implemented system. Testing is a main stage in software development life cycle. Goals of testing, methodologies and various testing will be discussing throughout this chapter. This chapter also will check that the functional requirements and the non-functional requirements outlined in the Requirement Specification Chapter are properly implemented.

7.2 Goals of Testing

Testing of this implemented system is done to verify that the system functions are successfully implemented according to the expectations. Following goals have been identified to be checked in this chapter.

- Functionality of functional requirements
- Functionality of non-functional requirements
- Identification of bugs

7.3 Testing Methodology

There are functional and non-functional requirements to be tested in this system. So, both functional testing and non-functional has to be done to verify the functionality of the system. Functional testing focuses on the functionality of the system while non-functional testing focuses on the quality of the implemented system.

This project has used the iterative software development methodology (Chapter 3). So, testing phase of this project has been carried out parallelly with the development. Following testing stages are the stages used in the testing phase of this project.

7.3.1 Unit Testing

Since this system has main four modules such as Mobile Application, Machine Learning model, Web Service and the database, the unit testing has been done to verify the functionality of the functions inside each component. Unit in Unit Testing can be identified as the smallest testable part of the system. Mostly, a unit can be a method, class or a procedure. So, unit testing can be done to verify the functionality of each unit in a complex system. Unit testing has been done by the author parallelly with the developing the software. By doing the unit tests author was able to identify the bugs in early stage of development cycle.

7.3.2 Integration Testing

Integration testing has been carried out in order to test set of individual units in the software. Integration testing can be done to verify the data flow between each component. And also, integration testing can be done instead of testing whole software program at once. Below table 7.3.1 describes about the integration tests which have been carried out.

Id	Objective	Test Case Description	Expected Result	Status
1	Identify interaction between Mobile Application and the Web Service when gathering data from the users.	Sending POST request from the mobile application to Web Service.	Receive data from users with the IRI value they got for the specific road and the other data.	Passed
2	Identify interaction between Machine Learning Model and the Web Service.	Sending POST request to train the Machine Learning Model and sending POST requests to get the predicted data from the model	Train the Machine Learning Model and identify the vehicle type which has used by the user.	Passed
3	Saving the data in SQL Database and retrieving data.	Saving the data gathered from user and retrieve the data for loading the map to the user.	Save the data in the SQL tables and sending data to webservice.	Passed

4	Identify the interaction between Web Service and the Mobile Application when showing results to the user.	Getting data from the Web Service and generate the map with road conditions.	Load the map in the mobile application with showing different levels of road conditions.	Passed

Table 7.3.1 Integration Testing

7.3.3 Functional Testing

As mentioned above, functional testing carried out to test the functionality of the system and they are properly meeting the functional requirements specified in the Requirement Gathering Chapter. Following table 7.3.2 is a results summary received after testing the functional requirements in Black Box testing approach. Refer Appendix D for test cases for each functionality mentioned below.

FR ID	Functional Requirement	Executed Test Cases	Passed Test Cases	Pass Rate
1	Gather Sensor Data	3	3	100%
2	Filter the data	1	1	100%
3	Calculate IRI Value	1	1	100%
4	Save data in database	2	2	100%
5	Prompt user to select vehicle type	3	3	100%
6	Train classifier	1	1	100%
7	Predict vehicle type	2	2	100%
8	Classify roads by road condition	3	3	100%
9	Implement the results	3	3	100%
10	View the road conditions in map	4	3	80%

Table 7.3.2 Functional Requirement Testing

It is identified that all the functional requirements have passed the functional testing phase. Functional testing accuracy can be calculated as the mentioned below.

Functional Testing Accuracy = 100+100+100+100+100+100+100+100+80/10

= 98%

7.3.4 Non-Functional Testing

- **Usability**

Usability testing was done to identify that the system is user friendly to the users of the system. According to the users' feedback it was identified that the simplicity of the UI design and the ability to get familiar with the implemented system without any pre training have made the system user friendly to all the users.

- **Compatibility Testing**

This testing method is done to check whether that implemented mobile application is runs on different android smart phones. For this testing, application was run and tested on Samsung Galaxy J8, Huawei Nova 3i and Samsung Galaxy S7 edge phones. The mobile application which built was successfully given results in those 3 phones.

7.4 Chapter Summary

In the chapter, a proper testing methodologies and test cases have been delivered with enough proves and test cases. A good probability of the test cases has been passed in the test cases and test were successful at the end of the chapter. Next chapter will be on evaluation criteria of the system.

Chapter 8 : Evaluation

8.1 Chapter Overview

This chapter is about the evaluation of the implemented system. This chapter focuses on evaluation goals, evaluation criteria, selection of evaluators and a summary of responses collected from selected evaluators. And also, this chapter discusses about how the quantitative evaluation has been carried out and the self-evaluation about the system.

8.2 Evaluation Criteria

Evaluation about the system is really important because it gives a feedback about the implemented system from the domain experts and the users. Evaluation criteria can be categorized as Qualitative evaluation and Quantitative evaluation.

- **Qualitative Evaluation**

Qualitative Evaluation is used to evaluate the approaches and concepts of syste based feedback from the domain experts and users.

- **Quantitative Evaluation**

Quantitative evaluation is used to evaluate the results and statistics based on the feedback gained from the selected users of the system about features, accuracy and performance of the implemented system.

8.3 Selection of Evaluators

Evaluators from different evaluator categories was identified to perform qualitative and quantitative evaluation. Below table 8.3.1 describes about the list of selected evaluator categories to evaluate the system.

Evaluator	System Involvement	Evaluation Type	Description
Road Development Authorities	Domain Expert	Qualitative	An official from Road Development Authority were interviewed to

			collect feedback bout the system.
Software Engineers	Technical Expert	Qualitative	Get the technical feedback about the system
Vehicle Users	User	Qualitative. Quantitative	Vehicle users are the target end users of the system. Feedback from them is really important.

Table 8.3.1 Evaluator Selection

8.4 Summary of Responses from Evaluators

Below tables 8.4.1, 8.4.2, 8.4.3 shows the responses received from the selected evaluators about the implemented system.

Domain Experts	
Chief Engineer of Road Development Authority	"As the Sri Lankan roads are under development stage, there can be difficulties in travelling on those. Travelers can't identify the road condition before travelling on it so that they might face difficulties. Not only that the government and the authorities are may not aware of the road conditions. They also get to know about the conditions of the road and will take necessary steps to develop the roads for comfortableness of the people. As a suggestion i would like to state that to have an information feed for authorities to have an analyzation for them to get an idea of the related road." **Chief Engineer** **Road Development Authority**

Table 8.4.1 Domain Expert Feedback

Technical Experts	
Software Engineer	"Java is a very compatible language that can be used in cross platform units and python also a very modernized language with more libraries that can be used very easily. The system is well coded

	and managed very well. I suggest to use some codes from scratch so that the code will be simple and run time will be faster." **Senior Software Engineer** **Virtusa (Pvt) Ltd**
QA Engineer	“This is a system about how to identify the Road Surface Quality using a mobile phone, in the sense to predict the road quality through mobile sensors. So, I think this is so beneficial to the all the users to identify the quality of the road by using the map. The most significance aspect is the user doesn’t require to involve in to this, therefore all the contributions are happening from the background. But there is a possibility that the user could contribute to the system as well.” **Senior QA Engineer** **Virtusa (Pvt) Ltd**

Table 8.4.2 Technical Expert Feedback

Users	
Uber Driver	“This system is basically for users to identify the road surface quality through a mobile app. As the uber drivers we are facing so much difficulties such as there sometimes that we are unable to get the customer to their expected destination due to the traffic, the time and specially the road condition. So, this app will very useful and it is important for the uber drivers so that they can manage to find the optional routes for the users to drop their customers in to their final destination.” **Uber Driver**
Daily Road User	“The system is interesting and very useful project for me as a long-distance driver. I could use this application to choose better roads with good conditions according to the vehicle type easily. This will help to save both my time and vehicle conditions. The application is user friendly and the way road conditions are presented in the map is

	interesting. It would be great if the driver can select from the road with best condition and fastest road. Wish you good luck with the project." **Road User**
Tour Driver	"I am a tourist guiding driver and I travel all around Sri Lanka almost every place. So, I experienced so many problems when traveling some places because of roads which I haven't been a long time. When I was finding routes to some places which are hiding from towns, I couldn't find whether the route is comfortable for guest or not. Sometimes I got negative feedbacks from guests because of bad condition of roads. From this mobile application I can see whether the route is suitable or not to travel with my guests. It shows specific colors according to the condition of roads and helps anyone to decide the best routes. I highly recommend this system to tourist drivers as well as all general drivers to use in their daily routine" **Tour Guide** **Ramani Tours**

Table 8.4.3 Users' feedback

8.5 Quantitative Evaluation

For the quantitative evaluation the developed mobile application was distributed among 5 users. And they were given two days of time to use the application. Evaluation form was given to the users to fill the form to get the evaluation and responses can be found in Appendix E.

All the users have identified that viewing the map is very important feature in the mobile application. 80% of the users also selected that selecting the vehicle type also an important feature in the system. From the users, 80% have rated the accuracy of the system as "Very Good" and 20% have rates as "Neutral". This shows that the system was able to achieve a good number in accuracy. Users have voted that system is highly user friendly and satisfied with them.

Two suggestions were given related to the system mentioned in below.

- "This system should be developed for both Android an iOS users with offline features."
- "More vehicle types should be added"

Overall it can be concluded that all most all the users have satisfied with the implemented system.

8.6 Critical Self Evaluation

Self-evaluation is done in order to identify the strengths and weaknesses of the implemented system from the authors perspective.

8.6.1 Overall Concept Evaluation

Poor road condition is a major problem that vehicle users are facing in day to day life. So, road condition monitoring using an easy and reliable way is needed. Usage of crowd sourcing for data gathering, the system is capable of gathering data from many users. So, this concept which use crowd sourced data is a good solution for this problem. But the crowd sourced data can be varied from vehicle of the suspension. That variance can be minimized from the concepts used in the implementation of this system.

8.6.2 Scope Evaluation

As the project is developed to consider the variance occurred because of the vehicle suspension of each vehicle category, the scope is narrowed down to two vehicle types such as two wheelers and three wheelers in this project. This narrow downing was done because it's hard to use lot of vehicles to do the testing. But, the same approach can be used for the other vehicle types as well when enhancing the project.

8.6.3 Design, Architecture and Implementation Evaluation

Designing of the prototype has been done after examining the research papers and other relevant methodologies. Implementation of the system was carried out according to those designs. Object Oriented Design approach which was selected in design phase, made it easier to develop a prototype with expected results. Tools and programming languages were selected after justifying the usage and those were accepted by industry experts also.

8.6.4 Prototype

Author was able to build the prototype within the time period allocated. A mobile application was developed with a simple user interface which user can directly see the road condition levels of each road in a user-friendly way.

8.6.5 Reflection of Functional Requirements

During the limited time period which the project has been carried out, it was able complete all the functional requirements. So, it is considered as a great achievement for this project. Below table is a reflection about the functional requirements.

FR No.	Functional Requirement	Priority	Status
1	**Gather Sensor Data**	Critical	Implemented
2	**Filter the data**	Critical	Implemented
3	**Calculate IRI Value**	Critical	Implemented
4	**Save data in database**	Critical	Implemented
5	**Prompt user to select vehicle type**	Desirable	Implemented
6	**Train classifier**	Critical	Implemented
7	**Predict vehicle type**	Critical	Implemented
8	**Classify roads by road condition**	Desirable	Implemented
9	**Implement the results**	Critical	Implemented
10	**View the road conditions in map**	Critical	Implemented

Table 8.6.1 Functional Requirements Completion

8.7 Research Contribution

As mentioned in the Chapter 2 of this document, the previous researches which was carried out was not considered about the suspension variance occurred by vehicle type. In this research, vehicle type has been considered when categorizing the road condition. Vehicle type was identified from the previous data gathered from the users using the KNN algorithm. Since KNN algorithm is a lazy learning algorithm, retraining the Machine Learning Model has been done frequently with the latest data. The implemented system was able to achieve the goal to monitor the road conditions considering the vehicle type of the user.

8.8 Chapter Summary

This chapter focused on evaluation of the system done by the selected evaluators and the author itself. The feedbacks received from the users shows that the implemented system is highly satisfied with the users. The suggestions got from the evaluators can be used in future enhancements of this project. In the critical self-evaluation part author has evaluated concepts, scope, design, architecture and the implementation. The status of implementation the functional requirements also described in this chapter.

Chapter 9 : Conclusion

9.1 Chapter Overview

The previous chapter was about the evaluation of the system. In this chapter a brief explanation is delivered of strength and weaknesses of the system, challengers overcame, objectives completed, lessons learned and the future enhancements of the system.

9.2 Achievements

9.2.1 Project Aim

The project aim of the system is,

“To research, design, develop, test and evaluate a crowd sourcing platform to identify the road conditions by using GPS and accelerometer sensor data of smart phones which will help users to get an idea about the road surface and minimize the accidents that occur due to poor road conditions.”

The research and the implemented solution lead to a comfortable journey by helping the user to get an understanding about the roads that they have to travel by identifying the road condition. And that success of the system will minimize accidents and the uncomfortableness occur due to poor road conditions.

Objectives	Completed percentage	Remarks
Identify a real world problem and propose a solution	100%	Uncomfortable travel and transport services due to poor road conditions.
Project management	100%	A clear management by taking the maximum usage of the resources available and managing the risks properly

Literature Review	100%	Literature reviews were performed with available research papers and existing systems
SRS Document	100%	SRS were delivered with functional and non-functional requirements.
Retrieve data	100%	Data retrieving done with various places and number of users
Design, implement, Test and Evaluate the system	100%	System was successfully design, implemented and evaluated.
Complete and finalize the document	100%	Document was completed successfully.

Table 9.2.1 Project Aim

9.3 Other Achievements and Learning Outcomes

By completing this research and the implementation some extra achievements also gained.

- Knowledge about laws and services in the country related to travel and transport
- Knowledge about maps and geographical areas around the country
- Researching skills
- Time management
- Knowledge about java, python and SQL programing languages.
- Improvement in bug fixing.
- Skills of solving problems
- Documentation skills

9.4 Challenges Overcome

In the period of the research and implementation period many challenges came and those were successfully sorted out.

9.4.1 Collecting Information

To make a rich background problem, lot of information from different sides were collected. Researches were done both from public and private transporting services available. It was a challenge to get the accurate information from trustworthy people.

9.4.2 Time Management

The time period given to complete the project was about 8 months. In the period there were other exams and assignments too. Thus, the time management was given a high priority in order to complete the research project in time. Full documentation with all the researches done and the completed implementation was done in that time period as previously managed.

9.4.3 Lack of knowledge of technologies used

Almost all the technologies used to the systems are new to the author. Therefore, those had to be learned before using. Online tutorials, videos related to the learning of technologies and expert's teaching were helpful to overcome this challenge.

9.5 Limitations of the system

- The user should have a smart mobile phone with android operating system installed in it.
- The mobile should have accelerometer and GPS facilities available.
- The system only working for two wheeled vehicles and 4 wheeled vehicles only.
- A proper internet facility should be available for the system to work.

9.6 Relation of Course Content

Skills gained in the field of software engineering was very helpful in the project. The Table shows the summery of skills that gained from the project.

Module Name	Usage

Using Java	Java is a class based as well as object-based language. That feature was very helpful in making of the implementation. Java is reliable secure and fast.
Python language	Python is a general- purpose language which is very compatible in any mode. So, it called as a multipurpose language. The system consist python in models and backends.
Algorithms and complexity	KNN algorithm for python is used in here. A proper understanding about KNN and Google map API was gained by using them in the system
Using git repositories	Using functions in git was very useful to save source code and related implementation code snippets. A good understanding of using git was gained.
Cloud Deployments	The developed spring boot application was deployed to the clouds in order to do the testing. A good understanding about the web service deployments to the cloud was gained.
Project management	Having a risk management and managing available resources was a challenge and learning how to manage those was successful.
Information system	Database management was completed with looking to the tasks. The knowledge of using azure database and google API functions were improved.

Table 9.6.1 Achievements

9.7 Future enhancements

Due to limitations of time and the resources available some functions could not be implemented with the initial implementation. Future enhancements that could be done in next updates are listed below.

- **Implementing the system for some other vehicle types**

In the initial implementation only two wheeled and four wheeled vehicles can be used to detect the new paths. Three wheeled and other vehicle types also should be compatible for the system with next updates.

- **Cross platform availability**

  Currently the app works only for android platforms. From next implementations both android and ios platform is compatible for the system to get work. Not only that the system is to be implemented for any android mobile with multiple design features.

- **Offline feature**

  In the initial system there has to be a proper internet network facility to have an accurate output. The future enhancement of offline feature is to be implemented as internet facility is not reachable in everywhere at the world.

- **Accuracy**

  The system is not 100% accurate as the system is manipulating with the users' data. Thus, from the next updates an accurate system is going to be put forwarded.

- **Use multiple classifiers**

  New functions and features are inventing frequently to the world with Artificial intelligence and other new technologies. The system is going to be add some of those new technologies which is relatable to the research area.

9.8 Concluding Remark

The project aim of the system was to research, design, implement and test with reliable added features and newly added algorithms. The background problem of not having the idea of the road before traveling can be minimized through using the app and the end user can detect the condition of the road before traveling. The aim of the project was successfully achieved, and the all the objectives were reached at the end of the project research concluding the remark of the system.

References

Android Central. (2018). *How does GPS work on my phone?*. [online] Available at: https://www.androidcentral.com/how-does-gps-work-my-phone [Accessed 18 Dec. 2018].

Android Developers. (2018). *LocationManager | Android Developers*. [online] Available at: https://developer.android.com/reference/android/location/LocationManager.html [Accessed 5 Mar. 2019].

Android Developers. (2018). *SensorEvent | Android Developers*. [online] Available at: https://developer.android.com/reference/android/hardware/SensorEvent.html#values [Accessed 7 Mar. 2019].

Ascelibrary.org. (2018). *Evaluation of Pavement Roughness Using an Android-Based Smartphone*. [online] Available at: https://ascelibrary.org/doi/abs/10.1061/JPEODX.0000058 [Accessed 24 Dec. 2018].

Asphaltroads.org. (2009). *Smoothness Matters: The Influence of Pavement on Fuel Consumption*. [online] Available at: http://www.asphaltroads.org/assets/_control/content/files/Smoothness_Matters.pdf [Accessed 24 Dec. 2018].

atlas.ti. (2019). *Qualitative and Quantitative research - Data Analysis | ATLAS.ti*. [online] Available at: https://atlasti.com/quantitative-vs-qualitative-research/ [Accessed 9 Feb. 2019].

Baeldung. (2018). *A Guide to Spring in Eclipse STS | Baeldung*. [online] Available at: https://www.baeldung.com/eclipse-sts-spring [Accessed 8 Mar. 2019].

Brownlee, J. (2016). *Supervised and Unsupervised Machine Learning Algorithms*. [online] Machine Learning Mastery. Available at: https://machinelearningmastery.com/supervised-and-unsupervised-machine-learning-algorithms/ [Accessed 20 Feb. 2019].

Devarajan, S. (2016). *Techniques for Requirements Gathering*. [online] www.tutorialspoint.com. Available at: https://www.tutorialspoint.com/articles/techniques-for-requirements-gathering [Accessed 2 Mar. 2019].

Gps.gov. (n.d.). *GPS.gov: GPS Accuracy*. [online] Available at: https://www.gps.gov/systems/gps/performance/accuracy/ [Accessed 28 Dec. 2018].

Guru99.com. (2018). *What is Module Testing? Definition, Examples*. [online] Available at: https://www.guru99.com/module-testing.html [Accessed 22 Mar. 2019].

HowStuffWorks. (n.d.). *How GPS Receivers Work*. [online] Available at: https://electronics.howstuffworks.com/gadgets/travel/gps2.htm [Accessed 30 Dec. 2018].

https://www.researchgate.net. (2018). *Identification and Calculation of Horizontal Curves for Low-Volume Roadways Using Smartphone Sensors*. [online] Available at: https://www.researchgate.net/publication/324811962_Identification_and_Calculation_of_Horizontal_Curves_for_Low-Volume_Roadways_Using_Smartphone_Sensors [Accessed 27 Dec. 2018].

https://www.researchgate.net. (2018). *On the analysis of road surface conditions using embedded smartphone sensors*. [online] Available at: https://www.researchgate.net/publication/316903253_On_the_analysis_of_road_surface_conditions_using_embedded_smartphone_sensors [Accessed 22 Dec. 2018].

https://www.researchgate.net. (2018). *Toward a mobile crowdsensing system for road surface assessment*. [online] Available at: https://www.researchgate.net/publication/322225525_Toward_a_mobile_crowdsensing_system_for_road_surface_assessment [Accessed 26 Dec. 2018].

https://www.sciencedirect.com. (2017). *Smart patrolling: An efficient road surface monitoring using smartphone sensors and crowdsourcing*. [online] Available at: https://www.sciencedirect.com/science/article/pii/S1574119216301262 [Accessed 23 Dec. 2018].

https://www.sciencedirect.com/. (2016). *The impact of road improvements on road safety and related characteristics*. [online] Available at: https://www.sciencedirect.com/sciehttps://www.sciencedirect.com/science/article/pii/S0386111216300097nce/article/pii/S0386111216300097 [Accessed 22 Dec. 2018].

Ieeexplore.ieee.org. (2018). *On the Use of 3-D Accelerometers for Road Quality Assessment - IEEE Conference Publication*. [online] Available at: https://ieeexplore.ieee.org/document/8417779 [Accessed 20 Dec. 2018].

Ijirae.com. (2014). *Analysis on Sensors in a Smart Phone- Survey*. [online] Available at: http://www.ijirae.com/volumes/vol1/issue9/OCCS10086.13.pdf [Accessed 21 Dec. 2018].

Intellipaat Blog. (2017). *Which is the most promising language for Machine Learning? - Intellipaat Blog*. [online] Available at: https://intellipaat.com/blog/most-promising-machine-learning-language/ [Accessed 24 Feb. 2019].

K, J. (2018). *12 Best Software Development Methodologies with Pros & Cons*. [online] Web Solutions Blog. Available at: https://acodez.in/12-best-software-development-methodologies-pros-cons/ [Accessed 8 Jan. 2019].

Kircherelectronics.com. (2017). *Android Acceleration Sensor: Low-Pass Linear Acceleration – Kircher Electronics*. [online] Available at: http://www.kircherelectronics.com/index.php/2017/12/28/android-acceleration-sensor-low-pass-linear-acceleration/ [Accessed 3 Mar 2019].

Kircherelectronics.com. (2017). *Android Acceleration Sensor: Low-Pass Filter – Kircher Electronics*. [online] Available at: http://www.kircherelectronics.com/index.php/2017/12/28/android-acceleration-sensors-low-pass-filter/ [Accessed 3 Mar 2019].

Kircherelectronics.com. (2017). *FSensor – An Open Source Android Sensor Filtering Library – Kircher Electronics*. [online] Available at: http://www.kircherelectronics.com/index.php/2017/12/28/fsensor-an-open-source-android-sensor-filtering-library/ [Accessed 3 Mar. 2019].

KUMAR PAL, S. (2018). *Software Engineering | Spiral Model - GeeksforGeeks*. [online] GeeksforGeeks. Available at: https://www.geeksforgeeks.org/software-engineering-spiral-model/ [Accessed 6 Feb. 2019].

Modernanalyst.com. (2018). *The Top Five Go-To Requirements Elicitation Methods > Business Analyst Community & Resources | Modern Analyst*. [online] Available at:

https://www.modernanalyst.com/Resources/Articles/tabid/115/ID/2483/The-Top-Five-Go-To-Requirements-Elicitation-Methods.aspx [Accessed 2 Mar. 2019].

Netguru.co. (2018). *Types of Machine Learning Algorithms: Supervised and Unsupervised Learning*. [online] Available at: https://www.netguru.co/blog/types-of-machine-learning-algorithms-supervised-and-unsupervised-learning [Accessed 30 Dec. 2018].

Nms.csail.mit.edu. (2008). *The Pothole Patrol: Using a Mobile Sensor Network for Road Surface Monitoring*. [online] Available at: http://nms.csail.mit.edu/papers/p2-mobisys-2008.pdf [Accessed 24 Dec. 2018].

Olic, A. (2017). *Advantages and disadvantages of Agile Project Management · Activecollab Blog*. [online] ActiveCollab. Available at: https://activecollab.com/blog/project-management/agile-project-management-advantages-disadvantages [Accessed 12 Feb. 2019].

Orau.gov. (2019). *Differences Between Qualitative and Quantitative Research Methods*. [online] Available at: https://www.orau.gov/cdcynergy/soc2web/content/phase05/phase05_step03_deeper_qualitative_and_quantitative.htm [Accessed 8 Feb. 2019].

Pdfs.semanticscholar.org. (2013). *A Study on the Use of Smartphones for Road Roughness Condition Estimation*. [online] Available at: https://pdfs.semanticscholar.org/1458/ee4b0bc04ad62e0173779295d275957138ed.pdf [Accessed 25 Dec. 2018].

Powell-Morse, A. (2016). *Iterative Model: What Is It And When Should You Use It?*. [online] Airbrake Blog. Available at: https://airbrake.io/blog/sdlc/iterative-model [Accessed 9 Feb. 2019].

Publishing, P. (2018). *Machine Learning with C++*. [online] https://www.udemy.com. Available at: https://www.udemy.com/machine-learning-with-c/ [Accessed 19 Feb. 2019].

S, S. (2016). *Difference Between Qualitative and Quantitative Research (With Comparison Chart) - Key Differences*. [online] Key Differences. Available at: https://keydifferences.com/difference-between-qualitative-and-quantitative-research.html [Accessed 5 Feb. 2019].

Team, S. (2017). *Top 4 Software Development Methodologies | Synopsys*. [online] Software Integrity Blog. Available at: https://www.synopsys.com/blogs/software-security/top-4-software-development-methodologies/ [Accessed 12 Feb. 2019].

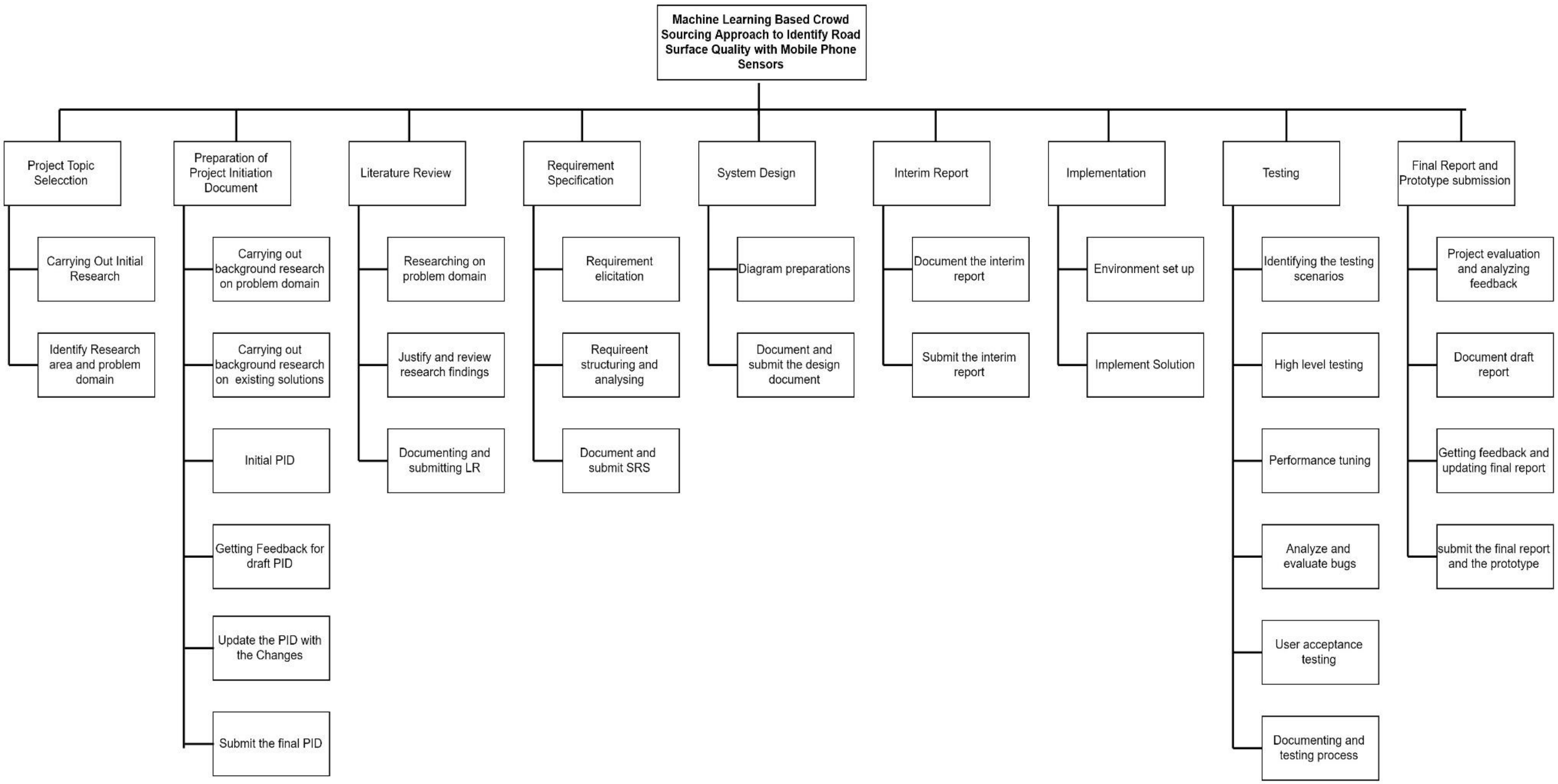
Tripnet.org. (2018). *Bumpy Road Ahead: AMERICA'S ROUGHEST RIDES AND STRATEGIES TO MAKE OUR ROADS SMOOTHER*. [online] Available at: http://www.tripnet.org/docs/Urban_Roads_TRIP_Report_October_2018.pdf [Accessed 23 Dec. 2018].

Umamaheswaran, V. (2018). *Comprehending K-means and KNN Algorithms*. [online] Becoming Human: Artificial Intelligence Magazine. Available at: https://becominghuman.ai/comprehending-k-means-and-knn-algorithms-c791be90883d [Accessed 4 Mar. 2019].

Venter, I. (2009). *Poor road conditions can have dramatic impact on cost of trucking*. [online] Engineering News. Available at: http://www.engineeringnews.co.za/article/logistic-costs-2009-05-01/rep_id:4136 [Accessed 25 Dec. 2018]

www.tutorialspoint.com. (2018). *Spring Boot Introduction*. [online] Available at: https://www.tutorialspoint.com/spring_boot/spring_boot_introduction.htm [Accessed 22 Feb. 2019].

Appendix A : Work Breakdown Structure



Appendix B : Gannt Chart

v	Task Name	Start	Finish	Duration
1	**Project Topic Selection**	**8/1/2018**	**8/28/2018**	**28d**
2	Carrying out initial research	8/1/2018	8/14/2018	14d
3	Identify research area and problem domain	8/20/2018	8/28/2018	9d
4	**Preparation of Project Initiation Document**	**9/1/2018**	**11/5/2018**	**66d**
5	Carrying out background research on problem domain	9/1/2018	9/15/2018	15d
6	Carrying out background research on existing solutions	9/16/2018	9/30/2018	15d
7	Initial Literature Review	10/1/2018	10/15/2018	15d
8	Getting feedback for draft PID	10/15/2018	10/28/2018	14d
9	Update the PID with the changes	10/16/2018	11/4/2018	20d
10	Submit the final PID	11/5/2018	11/5/2018	1d
11	**Literature Reveiw**	**11/1/2018**	**11/20/2018**	**20d**
12	Researching on problem domain	11/1/2018	11/10/2018	10d
13	Justify and review research findings	11/8/2018	11/12/2018	5d
14	Documenting & submitting LR	11/16/2018	11/20/2018	5d
15	**Requirement Specification**	**11/21/2018**	**12/15/2018**	**25d**
16	Requirement elicitation	11/21/2018	11/30/2018	10d
17	Requirement structuring and analysis	12/1/2018	12/10/2018	10d
18	Document and submit SRS	12/10/2018	12/15/2018	6d
19	**System Design**	**12/16/2018**	**12/31/2018**	**16d**
20	Diagram preparations	12/16/2018	12/25/2018	10d
21	Document and submit the design document	12/26/2018	12/31/2018	6d

V	*Task Name*	*Start*	*Finish*	*Duration*
22	**Implementation**	**1/1/2019**	**4/8/2019**	**98d**
23	Environment set up	1/1/2019	1/31/2019	31d
24	Implement solution	1/1/2019	4/8/2019	98d
25	**Testing**	**2/1/2019**	**4/8/2019**	**67d**
26	Identifying the testing scenarios	2/1/2019	2/5/2019	5d
27	High level testing	2/6/2019	2/20/2019	15d
28	Performance testing	2/10/2019	3/31/2019	50d
29	Analyze and evaluate bugs	2/20/2019	3/21/2019	30d
30	User acceptance testing	3/31/2019	4/7/2019	8d
31	Documenting the testing process	4/1/2019	4/8/2019	8d
32	**Final report and prototype submission**	**3/15/2019**	**4/25/2019**	**42d**
33	Project evaluation and analyzing feedback	3/15/2019	4/15/2019	32d
34	Document draft report	4/10/2019	4/20/2019	11d
35	Getting feedback and updating final report	4/21/2019	4/24/2019	4d
36	Submit the final report and the prototype	4/25/2019	4/25/2019	1d

Appendix C : Questions of the online survey

Final Year Project Survey

Hi, I'm Tharindu Buddhika Deshappriya, a final year student of Informatics Institute of Technology affiliated to the

What type of vehicle are you using to travel on roads? *

- [ ] Two Wheelers
- [ ] Three Wheelers
- [ ] Four Wheelers
- [ ] Other...

Would you like to have a smooth travelling on the road ? *

- ○ Yes
- ○ No

When selecting a route for your journey, do you consider road surface *

- ○ Yes
- ○ No
- ○ Maybe

Do you think that road surface condition is affecting road users safety and *

- ○ Yes
- ○ No

How would you rate about the Road Surface Conditions of Sri Lanka ? *

	1	2	3	4	5	6	7	8	9	10	
Low	◯	◯	◯	◯	◯	◯	◯	◯	◯	◯	High

Are you using a smartphone?

◯ Yes

◯ No

Have you used mobile applications such as Google Maps to find the routes ? *

◯ Yes

◯ No

Would you use if a mobile application is developed which can provide a map *

◯ Yes

◯ No

If you think this proposed system will be useful for you, what are your

Long answer text

Appendix D Test Cases for Functional Testing

FR01	**Gather Sensor Data**				
TC ID	**Scenario**	**Input**	**Expected Output**	**Actual Output**	**Result**
1	Gather sensor data from users having internet connectivity and GPS connectivity	Send accelerometer sensor data and location data to filtering.	Data should send to the filtering method	Data sent to the filtering method	Pass
2	Gather sensor data from users not having internet connection	Send accelerometer sensor data and location data to filtering.	Data should not be sent to filtering method.	Data didn't send to the filtering method	Pass
3	Gather sensor data from users not having GPC connectivity.	Send accelerometer sensor data and location data to filtering.	Data should not be sent to filtering method.	Data didn't send to the filtering method	Pass

FR02	**Filter the data**				
TC ID	**Scenario**	**Input**	**Expected Output**	**Actual Output**	**Result**
1	Data gathered from the mobile phone accelerometer should be filtered with the low pass and high pass filters.	Accelerometer sensor data	Should get noise reduced sensor data	Got noise reduced sensor data	Pass

FR03	**Calculate IRI Value**

TC ID	Scenario	Input	Expected Output	Actual Output	Result
1	Calculate IRI value	Calculated distance and calculated vertical vibration	IRI value should be calculated	IRI value calculated	Pass

FR04	**Save data in database**				
TC ID	**Scenario**	**Input**	**Expected Output**	**Actual Output**	**Result**
1	Save the calculated IRI value in the database	Send IRI value, userId and other data in a POST request with a road name which exist in the database	Data should be inserted to the database	Data inserted to the database	Pass
2	Save the calculated IRI value in the database	Send IRI value, userId and other data in a POST request with a new road name which doesn’t exist in the database	Road Name should be applied to the roads table in the database with a new road id.	Road Name inserted to the roads table in the database with a new road id.	Pass

FR05	**Prompt user to select vehicle type**

TC ID	Scenario	Input	Expected Output	Actual Output	Result
1	Request the vehicle type from the user	Select the vehicle type as two wheeled vehicles	Database should update with the vehicle type	Database updated with the vehicle type	Pass
2	Request the vehicle type from the user	Select the vehicle type as four wheeled vehicles	Database should update with the vehicle type	Database updated with the vehicle type	Pass
3	Request the vehicle type from the user	User cancels the request	Remove the request from the mobile application and update the database	Remove the request from the mobile application and update the database	Pass

FR06	**Train classifier**				
TC ID	**Scenario**	**Input**	**Expected Output**	**Actual Output**	**Result**
1	Training the machine learning classifier	Latest data retrieved from the database.	Machine learning model should be trained with the data	Machine learning model trained with the data	Pass

FR07	**Predict vehicle type**

TC ID	Scenario	Input	Expected Output	Actual Output	Result
1	Predict vehicle type as two wheeled	Road id and the calculated IRI value	Machine learning model should return the vehicle type as two wheeled for the new data and update the database	Machine learning model returned the vehicle type as two wheeled for the new data and updated the database	Pass
2	Predict vehicle type as four wheeled	Road id and the calculated IRI value	Machine learning model should return the vehicle type as four wheeled for the new data and update the database	Machine learning model returned the vehicle type as four wheeled for the new data and updated the database	Pass

FR08	**Classify roads by road condition**				
TC ID	**Scenario**	**Input**	**Expected Output**	**Actual Output**	**Result**
1	Classify the road condition as Good	Recalculated IRI value	Road should be classified as good	Road classified as good	Pass
2	Classify the road condition as Average	Recalculated IRI value	Road should be classified as Average	Road classified as Average	Pass
3	Classify the road condition as Poor	Recalculated IRI value	Road should be classified as Poor	Road classified as Poor	Pass

FR10	**View the road conditions in map**				
TC ID	**Scenario**	**Input**	**Expected Output**	**Actual Output**	**Result**
1	View the road conditions in the map when user connected with internet	Open the mobile application	Map should be loaded with the classified road conditions	Map loaded with the classified road conditions	Pass
2	View the road conditions in the map when user is not connected with internet	Open the mobile application	Request user to turn on the internet connectivity	Application crashed	Fail
3	Draw polylines with different colors to show the classified road condition	Latitude and longitude values with the road id	Map should be loaded with the classified road conditions	Map loaded with the classified road conditions	Pass

Appendix E Evaluation Form

Evaluation Form

What do you think about the features of this system?

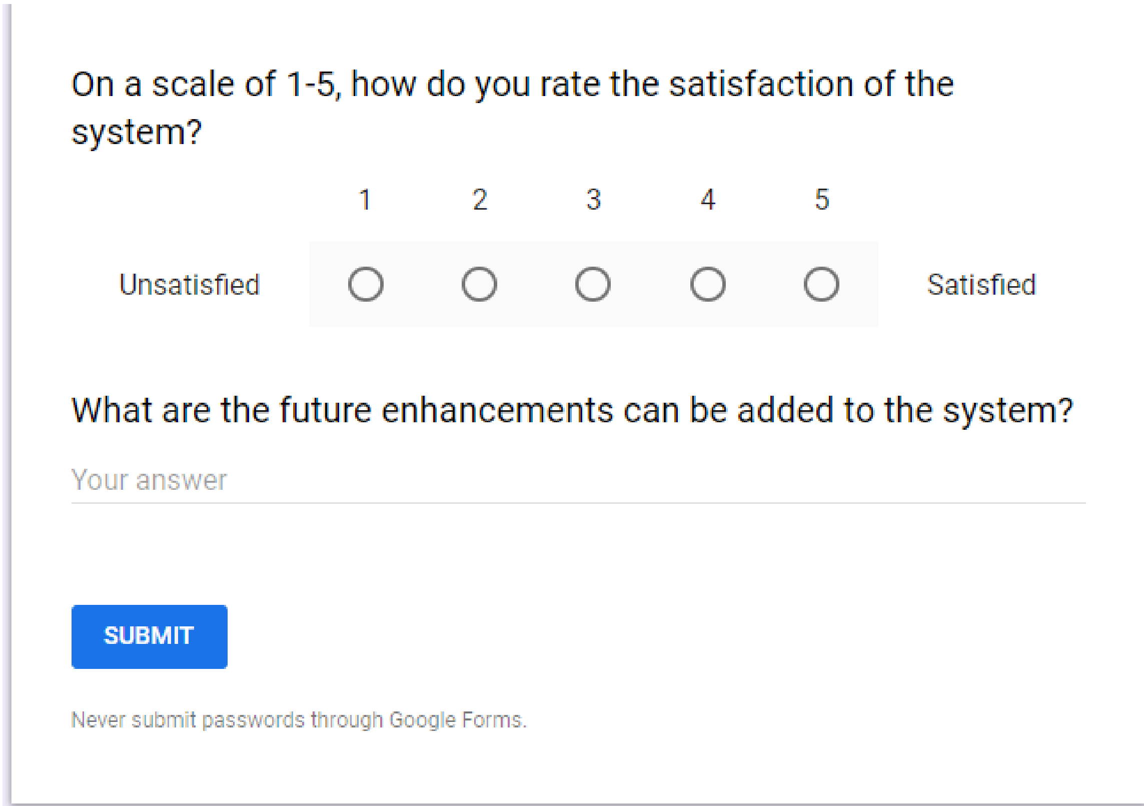
	Very Important	Somewhat Important	Neutral	Not Important
View The Map	○	○	○	○
Select Vehicle Type	○	○	○	○

How do you rate the accuracy of the system?

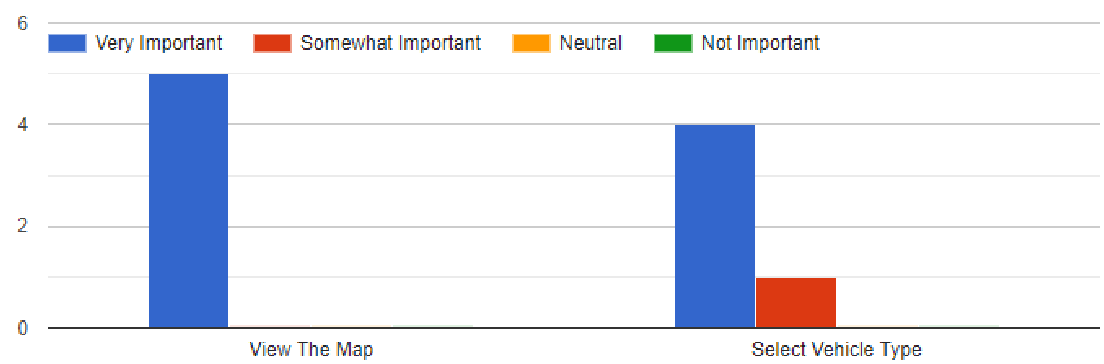
- ○ Very Good
- ○ Neutral
- ○ Not Good

On a scale of 1-5, how do you rate the user friendliness of the system?

	1	2	3	4	5	
Not User Friendly	○	○	○	○	○	Very User Friendly

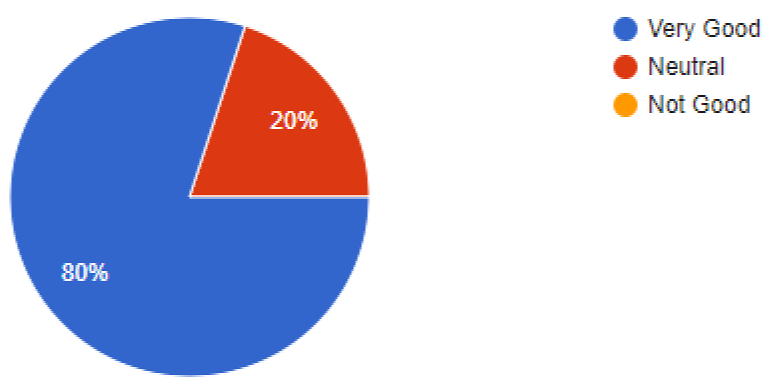


What do you think about the features of this system?



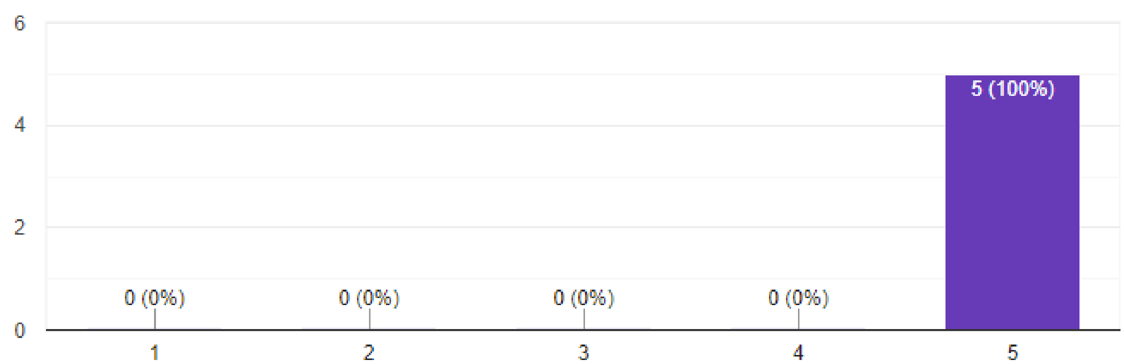
How do you rate the accuracy of the system?

5 responses



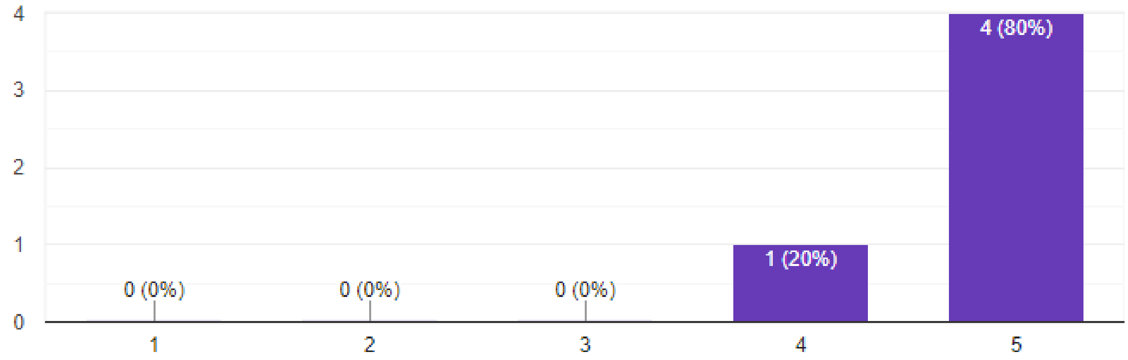
On a scale of 1-5, how do you rate the user friendliness of the system?

5 responses



On a scale of 1-5, how do you rate the satisfaction of the system?

5 responses



What are the future enhancements can be added to the system?

2 responses

This system should be developed for both Android an iOS users with offline features.

More vehicle types should be added