ENHANCEMENT OF VEHICLE SECURITY USING ANDROID TECHNOLOGY

NAME: EMILY NJOKI WANYEKI

REG NO: N16/3/0561/017

A research proposal submitted to the department of informatics and computing in the school of

Science and applied technology in partial fulfillment of the requirement for the award of Bachelor’s degree in information and communication technology in Laikipia University.

Year: 2021

# DECLARATION

This project proposal is my original work and has not been presented for a degree in any other university.

…………………………….. …………………..

Signature Date

This project proposal has been submitted for examination with my approval as university supervisor.

……………………………. …………………

**Signature Date**

# ABSTRACT

This paper propose a real time tracking application based on android technology. It involves use of two android devices where one acts as the tracking device installed in the vehicle or vehicles that have android auto software that can run android applications. One smartphone is used to view the location of the device. Each device is logged onto using a specific email attached which is synced with the tracking database. Only an authorized user can access the tracking information. The paper also examines previous suggested and developed systems used to enhance vehicle security. In order to show efficiency of the system, this paper presents experimental results of the android vehicle tracking app and future recommendations.

Contents

[DECLARATION ii](#_Toc71331)

[ABSTRACT iii](#_Toc71332)

[CHAPTER 1: 1](#_Toc71333)

[1.1 Background 1](#_Toc71334)

[1.2 Introduction 1](#_Toc71335)

[1.3 Statement of the problem 2](#_Toc71336)

[1.4 Proposed solutions 3](#_Toc71337)

[1.5 Objectives 3](#_Toc71338)

[1.5.1 Main objective 3](#_Toc71339)

[1.5.2 Specific objectives 3](#_Toc71340)

[1.6 Research questions 4](#_Toc71341)

[1.7 Justification 4](#_Toc71342)

[1.8 Proposed Research and System Methodologies 4](#_Toc71343)

[1.8.1 Research Methodology 4](#_Toc71344)

[1.8.2 System methodology 4](#_Toc71345)

[1.9 Scope 5](#_Toc71346)

[CHAPTER 2 6](#_Toc71347)

[2.1 INTRODUCTION 6](#_Toc71348)

[2.2 THEORETICAL REVIEW/CONCEPTUAL FRAMEWORK 6](#_Toc71349)

[2.2.1 Annual Crime Report Statistics 6](#_Toc71350)

[2.2.2 Explanation of Automatic Vehicle Location 8](#_Toc71351)

[2.3 Review of Previously Existing Tracking Systems 9](#_Toc71352)

[2.3.1 Arduino Vehicle Tracking Systems 9](#_Toc71353)

[2.3.2 Skynanny GPS tracker 10](#_Toc71354)

[2.3.3 Live Trac PT-10 10](#_Toc71355)

[2.3.4 The Mileage Ace GPS Mileage Tracker 10](#_Toc71356)

[2.3.5 The Trackimo 11](#_Toc71357)

[2.3.6 The Spyzie Tracking System 12](#_Toc71358)

[2.3.7 MOTO safety OBD Teenage Driving Training Car Monitoring System 12](#_Toc71359)

[2.3.8 BrickHouse Security Spark Nano 13](#_Toc71360)

[2.3.9 PocketFinder 13](#_Toc71361)

[2.3.10 The AMBERalertGPS 14](#_Toc71362)

[2.3.11 Caref GPS 14](#_Toc71363)

[2.3.12 WORKFLOW Vehicle Real Time Tracking (Arduino Mega) 15](#_Toc71364)

[2.4 RELATED TECHNOLOGIES 16](#_Toc71365)

[2.4.1 GSM Technology 16](#_Toc71366)

[2.5 Critics on the Previously Existing System 17](#_Toc71367)

[2.6 Summary 18](#_Toc71368)

[2.7 GAPS IN EXISTING SYSTEMS 18](#_Toc71369)

[CHAPTER 3 20](#_Toc71370)

[3.1 Introduction 20](#_Toc71371)

[3.2 Systems Development Methodology Used 20](#_Toc71372)

[3.2.1 Agile Methodology 20](#_Toc71373)

[3.2.2 Agile Software Methodologies 21](#_Toc71374)

[3.2.3 The Agile Process Flow 21](#_Toc71375)

[3.3 Feasibility Study 24](#_Toc71376)

[3.4 Requirements Elicitation 25](#_Toc71377)

[3.5 Data and System Analysis 26](#_Toc71378)

[3.6 System Specification 26](#_Toc71379)

[3.6.1 Functional requirements 26](#_Toc71380)

[3.6.2 Non-functional requirements 26](#_Toc71381)

[3.7 System Functionality 26](#_Toc71382)

[3.7.1 Hardware requirements 27](#_Toc71383)

[3.7.2 Software requirements 27](#_Toc71384)

[3.8 Logical and Physical design 27](#_Toc71385)

[3.8.1 Logical Design 27](#_Toc71386)

[3.8.2 Physical Design 28](#_Toc71387)

[3.8.3 Other diagrams present 29](#_Toc71388)

[3.9 System Architecture 32](#_Toc71389)

[CHAPTER 4: SYSTEM IMPLEMENTATION (CODE GENERATION AND TESTING) 33](#_Toc71390)

[**4.2 Construction (System Creation)** 33](#_Toc71391)

[**4.2.1 Statistical analysis** 33](#_Toc71392)

[4.3 Code Generation 41](#_Toc71393)

[4.3.1 Login 41](#_Toc71394)

[4.3.2 Home Activity 44](#_Toc71395)

[4.4 Testing 50](#_Toc71396)

[4.4.1 Unit Testing 50](#_Toc71397)

[4.4.2 Integration Testing 51](#_Toc71398)

[CHAPTER 5: SUMMARY, RECOMMENDATION AND CONCLUSION 52](#_Toc71399)

[5.1 Introduction 52](#_Toc71400)

[5.2 System constraints 52](#_Toc71401)

[5.3 Future enhancement 52](#_Toc71402)

[5.4 Recommendations 52](#_Toc71403)

[5.5 Conclusion 52](#_Toc71404)

[References 54](#_Toc71405)

[APPENDICES 55](#_Toc71406)

[APPENDIX A: QUESTIONNARE 55](#_Toc71407)

[Appendix B: BUDGET 56](#_Toc71408)

[APPENDIX C: GANTT CHART FOR PROPOSED ACTIVITIES 57](#_Toc71409)

List of Abbreviations

|  |  |
| --- | --- |
| **WORD** | **MEANING** |
| GPS | Global Positioning System |
| UML | unified modeling language |
| INTERPOL | International Police |
| SDLC | System development lifecycle |
| GSM | Global System for Mobile |
| AGD | Agile development technique |
| UI | User Interface |
| ERD | Entity relationship diagram |
| GPRS | General Packet Radio Service |
| HLR | Home Location Register |
| VLR | Visitor Location Register |
| AUC | Authentication Center |
| EIR | Equipment Identity Register |
| IMEI | International Mobile Equipment Identifier |
| GNSS | Global Navigation System |
| MS | Mobile Station |
| SS | Switching System |
| BSS | Base Station |

Table of Figures

Figure 1crime report ................................................................................................................... 6

Figure 2 Bar Graph ..................................................................................................................... 7

Figure 3 Pie Chart ...................................................................................................................... 7

Figure 4 GPS Module............................................................................................................... 15

Figure 5 Stages of Agile ........................................................................................................... 22

Figure 6 Vehicle Information ................................................................................................... 33

Figure 7 Prevalence of Tracking .............................................................................................. 34

Figure 8 Smart Phone Technology Prevalence ........................................................................ 36

Figure 9 Email Users ................................................................................................................ 36

Figure 10 Recommendations.................................................................................................... 37

Figure 11 Login1 ...................................................................................................................... 38

Figure 12 Login2 ...................................................................................................................... 38

Figure 13 Menu Options .......................................................................................................... 39

Figure 14 Home page ............................................................................................................... 40

Figure 15 Tracking Info ........................................................................................................... 40

# CHAPTER 1:

**INTRODUCTION**

## 1.1 Background

Looking at the World in general due to growth of economy people have been able to buy vehicles at a very high rate, which are both used to serve as personal means of transport and also been used to carry out the various business activities. ”Number of motor vehicles registered was reported at 2,989,788.000 Unit in December 2017” (KNBS, 2017). Despite this growth it has brought new challenges. The number of car theft cases has also risen drastically. Some of the vehicles stolen are smuggled to different countries making it hard to trace them. Data from INTERPOL shows about 200,000 stolen vehicles are smuggled from the United States to Mexico, Central and South America. (Kejitan, 2019) The DCI detectives recovered several high end vehicle smuggled illegally to Kenyan Country which were stolen from South Africa and United Kingdom.

These scenarios only inform about few countries which statistically shows that the number of vehicles stolen from different countries is at a very high. In Kenya for the past few years, there has been increased reports of vehicles in different areas within the country. Most of the vehicles stolen had no form of any tracking device implanted on them making very hard for the authorities and owners to track them. Those who have tracking devices installed in their vehicles, their vehicles are fully managed by third parties (companies that offer tracking services). It has also been discovered that some of those working in this tracking companies work together with car gangs to rob clients of the vehicles by providing them with tracking details. This has also posed risk on which tracking vendor to trusts. We propose an affordable system where most of the tracking operations are carried out by the user himself or herself therefore drastically reducing third parties from accessing your vehicle details.

## 1.2 Introduction

In the current advancement of technology, it has led to development of cheap affordable devices which can be used by variety of users to better their life in many aspects. Mostly when it comes to the area of mobile technology, there has been total evolution which has really impacted our lives positively. Mainly development of the smartphones which has joined up many technologies in it such as Google maps, GPS, camera, strong internet connections and etc. This technology has enabled us to come with a proposal to develop simple android application that will enable users to find and track their vehicles in real-time without requiring third parties. This project uses android phones where one acts as a tracker and another is used to monitor. The Android App Can be used to track several cars as each car has a unique email associated to it. The user must login through email or a Google account which is synced to the firebase database by the admin. For tracking to take place devices must be paired together through the email associated with the car. The user sends a tracking request to the vehicle to track and the request is accepted so as to pair the devices together. User clicks on the car email to track it and the car location is shown on map, and user can also be able to find distance between user’s current positions to that of the vehicle. The tracking system uses the smartphone inbuilt Global Positioning System to provide accurate coordinates.

## 1.3 Statement of the problem

In Kenya car theft has been very rampant in many parts of the country but especially in the major cities. In 2016 according to Knoema Crime Statistics private cars theft rate was 0.9 case per

100,000 populations. (Daghar, 2019) Provided a research that showed that in 2018, 1370 vehicles were stolen not accounting for the unreported cases. Also according to a research conducted by

Security Research and Information Center indicated that most cars that were stolen were personal cars therefore raising concern on security of vehicles. (Daghar, 2019) Indicated that a

Vehicle had been stolen from Uganda and smuggled to Kenya. Some of the companies involved in offering tracking services, some of its workers engaged with car thieves to disclose the whereabouts of customer’s vehicles raising concerns on third party access to personal

Information. Most current systems require monthly subscriptions to access tracking services therefore driving people away from using these tracking technologies. Also tracking systems provided by vendors require skilled experts to install them, cars therefore must be interfered which may lead to other problems arising within the car if not correctly installed.

## 1.4 Proposed solutions

The proposed solution is to design an android application that will allow the user to track and monitor his or her vehicles. The solution will be purely depended on smartphone technology where one of the smartphones will be used as tracker and the other will be used to monitor the vehicles. A user will have to log in with email or Google account for authorization purposes where the admin will have synced the users email with the database. Each vehicle to be tracked will have a smartphone device or android auto installed in with which the tracking app will run. Each smartphone in a vehicle is identified with a unique email associated to that vehicle. The user will be able to carry out the tracking himself or herself without requiring any special skill set.

Characteristics of the proposed system are as follows.

1. The system will be user friendly; user will just have to login using his or her email credentials and incase issues of forgotten passwords arise the system will allow a user to recover his or her password.
2. The system will be efficient as it will provide the user with real-time data. iii.

The system will be flexible allowing other functionalities to be added.

## 1.5 Objectives

### 1.5.1 Main objective

To develop and implement an android app that will allow a user to track a vehicle in real time.

### 1.5.2 Specific objectives

1. To design an app that will only allow authorized users to login.
2. To develop a system that shows current real-time location of the vehicle.
3. To design an android app that will allow a user to conduct tracking himself or herself without third party access.

## 1.6 Research questions

1. What safeguards will be implemented to ensures only authorized user is authenticated to access the system?
2. How do we relay the real-time location of the vehicle?
3. How do we build the app to allow a user to have control over the tracking process?

## 1.7 Justification

The system will minimize car theft cases as it will assist authorities and car owners track their vehicles. The system will also be very easier to as it only requires a user to have smartphone devices where one is used as tracker and the other one to monitor. The system also updates tracking information automatically. It also enhances use of existing technology and knowledge to assist human beings in solving car security problems. It will also provide the user with all control to track any vehicle at his or her convenience at very low cost without requiring third parties from accessing their personal information. The system will serve as a basis that existing technologies can be used to smoothen the life of human beings.

## 1.8 Proposed Research and System Methodologies

### 1.8.1 Research Methodology

Descriptive research design was adopted in this case. Respondents were contacted in their usual working place allowing them to respond more freely to the questions and give relevant answers to the questions asked. Data were collected using questionnaires, interviews and observation, which was later analyzed to aid in delivery of the proposed android tracking system.

### 1.8.2 System methodology

The system methodology that the system was based on is the Agile Methodology. Agile methodology allowed various stages of software development which included conception, inception, construction/iteration, release, production and finally retirement.

Justification on Agile Methodology

It allowed continues iteration of development and testing thought the SDLC. It also allowed both testing and development at the same time.

Agile system methodology also works on core values which include the following.

1. It supports both individuals and team interactions within the processes and tools.
2. Agile methodology also supports working software over comprehensive documentation.
3. Another fundamental core of agile is that it supports customer collaboration.
4. Also provide means to respond to changes in plans.

## 1.9 Scope

The scope of the study was limited to designing of android vehicle tracking application in Kenya. It will focus on developing an alternative way of tracking vehicles using current existing technologies. The study is depended on smartphone devices especially those supporting the android operating system.

## CHAPTER 2

## 2.1 INTRODUCTION

In the course of this chapter, the study involves reviewing of existing systems of the same nature as this android tracking technology proposal. This chapter will also critique in comparison the new system and other existing system in place and other researchers work. Through this, gaps will be identified from past relevant works. This makes the new system an improved version. There will also be citations made in reference to the new system. As to make it have background backing.

We discuss of car tracking technology and how the technology has played a big part in the vehicle production industry and how secure the vehicles are in this error.

## 2.2 THEORETICAL REVIEW/CONCEPTUAL FRAMEWORK

### 2.2.1 Annual Crime Report Statistics

In the past there have been several reaps in the car production industry in respect to the security of the vehicles. In Kenya police service Annual crime report (service, 2018), there have been 88,268 reported compared of 77,986 reported in 2017 which was an increase of 10,276 cases or 13%. Crime rates have rapidly increased and are projected to rise in the year 2020. Out of the crime reported in 2018, vehicles and other theft reported were 34 cases (2.4%).

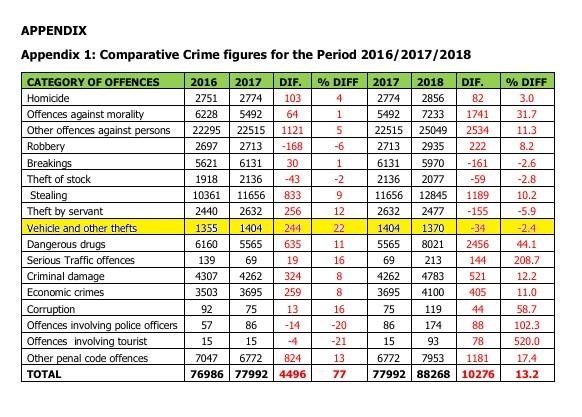


Figure 1crime report

When representing crime reports in a bar graph.

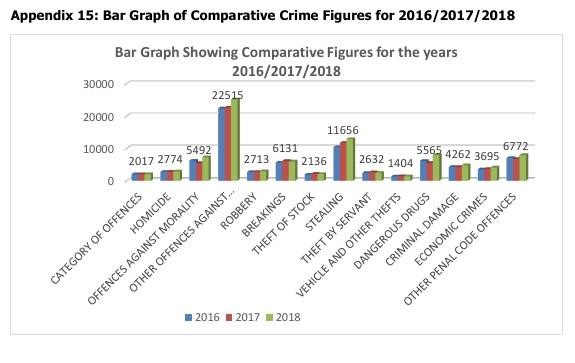
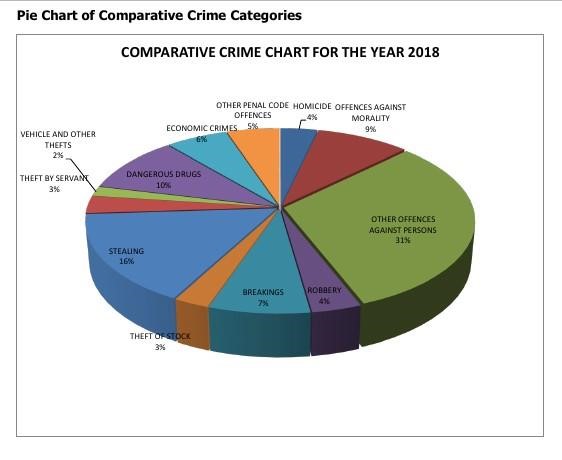


Figure 2 Bar Graph

When representing crime report in a pie chart.



*Figure 3 Pie Chart*

### 2.2.2 Explanation of Automatic Vehicle Location

According to (Wikipedia), vehicle tracking system combines Automatic Vehicle Location in a vehicle with software that collects fleet data for an accurate visual of the vehicles location. Modern vehicles tacking systems commonly use GPS or GLONASS technologies. Other technologies are electronic maps which can be viewed in the internet or specialized software.

There are several types of vehicle tracking devices. They are either: “passive” or “active”. In

“Passive” devices store GPS location, speed, and heading and sometimes trigger events such as car on/off, door open/closed. On reaching destination the data is downloaded from device to computer for evaluation. Passive systems include auto download type the data via wireless download. An “active” device collects the same information but data gets transmitted in near real-time via cellular or satellite networks do a computer or datacenter for evaluation.

Major constituents of GPS based tracking are:

1. GPS tracking: device fit in the vehicle and captures vehicle information.
2. GPS tracking server: receives data, securely stores it, serving this information on demand to the user.
3. User interface: UI determines how ne will be able to access information, view vehicle data and elicit important details from it.

Due to the risen cases in car thievery technological experts think they could provide a better solution to this prevailing problem. Clients would like tom tell their precise car location and the travelling settings of their vehicle including a clear maintenance report of their vehicles or rather cars which could reference the car performance history. For those individuals who reside in areas prone to car theft they might need to install a car track system in their personal vehicles in case an individual decides to steal your vehicle. In this case we try to embrace technology by creating an android application to install in the vehicle system and the user would create mail accounts that would trace location of the other android application connected to the same mail via the same android application we developed.

This is quite necessary because you would no longer need to pay a tracking company to track your car. Furthermore, this would cut on costs since you no longer have to pay for your vehicle to be tracked and all you need is just two android devices with the App installed.

There is a number of tracking devices that have been created before and worked perfectly fine except that you have to go through the company. Trust issues may also arise since one must trust the company to keep their information/Data as confidential as possible. There have existed prior systems Like Skynanny.

## 2.3 Review of Previously Existing Tracking Systems

### 2.3.1 Arduino Vehicle Tracking Systems

In Humaid Saif Alshamsi journal uploaded in the International Journal of Science and Technology (IJST Explore) (Alshamsi, 2017), with the advancement of technology, there has been an increase in the usage of vehicle tracking system. Arduino tracking systems enable the display of vehicle position on Google Maps. The GPS, GSM/GPRS modules controlled by Arduino mega are installed in the computer. The system works in the way that it updates the vehicle position after every 10 seconds as the vehicle is moving. Vehicle tracking is now been largely used. The system allows owners to observe and track the vehicle to know the whereabouts of their vehicle, its movements and there past activities of their automobile. This technology is popularly called a real time Vehicle Tracking System that ensures the security of the vehicles. When the vehicle gets stolen, the location data from the tracking system is used to find the location so the police can track the car. When users make a request, the GPS coordinates of the vehicle are transmitted to the specified mobile. The user will be provided the position of the vehicle to the police for further action. Position is provided in terms of latitude and longitude and is vied using Google maps. The system also has an SOS features whereby the driver can press a button to get help in case of an emergency. This system gives information to the authorized users of the system via internet websites.

How does the Vehicle Real Time Tracking (Arduino Mega) work? The hardware and software of the GPS and GSM technology has already been developed. It is divided in two parts:

1. Mobile unit.
2. Controlling station.

All the processes in the various interfaces for transmission of data worked successful. These results are compatible with current GPS technologies. This vehicle tracking device is installed in the vehicle to allow owner to identify the position (location) of the vehicle. This paper (journal) checks the Vehicle Real Time Tracking (Arduino Mega) using the GPS (Global Positioning System) and GSM (Global System of Mobile Communication), which continuously monitor the location of a moving vehicle and the status of the concerned vehicle .

### 2.3.2 Skynanny GPS tracker

Skynanny GPS tracker of 2018 for a car is a good system because of its value and functionality. The system has a panic button and a primary tracking application with an on-board SOS button used in case of emergency. The system has this one strength that it contains an SOS button that is only present on expensive cars.

The major disadvantage:

Is that the hardware used to construct the whole system is of poor quality.

### 2.3.3 Live Trac PT-10

Live Trac PT-10 of 2018 was a business grade GPS car tracker that has been adjusted for tracking and following. The tracker has amazing features like it coming along with a high capacity battery power and a motion-activated tracking feature which only correlates for an excellent tracking device. It contains time-based location update schedule that can use a smart motion-based update schedule to save the battery. This GPS tracking system came with some added on features like it having a good web portal. The system could too preserve the battery power for longer hours making it a reliable system in remote areas without the reach of electrical energy. This system apart from it having such prominent features and advantages, it has the dark side too. One of its disadvantage was that it was in a way too expensive for lower class people to afford it. Nevertheless, the system was enormous and indiscrete compared to other GPS car trackers.

### 2.3.4 The Mileage Ace GPS Mileage Tracker

Most organizations use the Mileage Ace car GPS tracker for accounting and logistics departments that depends on accurate mileage records. The designated driver generates logs once the driver returns from the trip. Apart from this, this app automatically uploads all the trips made without any further action. This tracking device had great features like having a 3-year GPS protection plan which covers the product malfunction and breakdown. It could also support WIFI uploading from the vehicle making the device more effective compared to other previous devices or other systems that existed before it. The Mileage Ace GPS Mileage Tracker came along with 100% online Tamper-Proof mileage record that protects your deduction from an audit. This system furthermore has 100% accurate Audit Compliant Mileage Logs. With this features probably the system had some advantages in it one of the being that it could automatically upload date when in the range of WIFI networks. Apart from that the system had affordable subscriptions making it cost effective hence attracted more customers to use the system. The system furthermore came along with tamper-proof records with years of history stored in the cloud. This ensured proper and secure storage of data that could be easily retrieved from the cloud storage. The system despite having such amazing features had the disadvantages too.

One of the disadvantages:

Being that the GPS may mislabel locations and it does not have a real-time tracking too.

Furthermore, the stops made by the system are difficult to highlight.

### 2.3.5 The Trackimo

Trackimo is a car GPS tracker with a powerful functionality and an economical cost of ownership. Trackimo uses a motion-activated tracking schedule to prolong the battery’s lifespan. The Trackimo had prominent features including the fact that it comes with an inbuilt GPS satellites which transmit data to the server using AT and T wireless connection. The system too offers worldwide coverage. The device is traceable through the web or an Android app everywhere. Furthermore, The Trackimo has another valuable feature that It comes with a panic button that is used to alert the user in case of an emergency through text messages or email address making the system real time and efficient in terms of handling emergencies. Furthermore it supports real-time tracking thanks to the GPS and GSM features. The system has advantages including the fact that it’s more portable and easy to use making it more understandable to the non-tech customers or users. The Trackimo battery lasts longer making it reliable to the remote rural areas without electricity and furthermore it could last like the whole day. The history data of this device is made useful too. There are a number of disadvantages too in this System. One of them being that it does not support Android apps and some users may find the day long battery life a bit short and inconvenience. To some users, the belt clip cover may be hard to install.

### 2.3.6 The Spyzie Tracking System

To each and every driver or a caring parent out there in search of the best GPS car tracker, you should not look any further than the Spyzie tracking program. Unlike other GPS trackers, this tracker comes with state-of-the-art features which can only be seen as a source of envy by other trackers. The prominent features of the Spyzie Tracking System include the fact that This app has an exceptionally in-built GPS system that keeps track of each and every movement that your car makes. The system too comes with a record and storage section that allows you to keep a record of each and every location that your car has visited. Moreover, It comes with a well detailed map that helps you in locating or tracking your car. The advantage of using this system is that you can easily monitor the current location of your car and thanks to the combination of the GPS and the geo referencing features. Moreover, it comes with a user-friendly interface which makes it easy to operate and understand. Its only one disadvantage is that You must install this app in a phone for it to function as required.

### 2.3.7 MOTO safety OBD Teenage Driving Training Car Monitoring System

The MOTO safety OBD Teenage Driving Training Car Monitoring System is a serial port tracker that gives you full information on what is happening in your vehicle. It reports coaching programs that help train new drivers and get rid of bad habits.

It supports GEO-fencing hence allows you to set-up curfew hours that warns you when the driver drives off-limits. A route replay lets you view your driver's entire driving course for the whole day. It has a daily driver’s report card which summarizes your driver's driving for the day and the scores achieved. The system has two important advantages one being that its relatively economical and can be affordable to any economical class in the society. The second advantage of this system is that it has a complete teenager driving education course, which includes scores for hard breaking and other bad habits.

The system has two alarming disadvantages:

One being that the system or the application may be relatively limited in features.

Second disadvantage is that the Acceleration alerts of the system are quite sensitive.

### 2.3.8 BrickHouse Security Spark Nano

The Spark Nano 5.0 GPS tracker for a car is a world-class car tracking device which tracks people, vehicles or assets in real time. It is perfect for small business entrepreneurs and parents who want to keep track of what they value the most. Its Features include that You can view different locations using any PC, mobile phone, or tablet.

It has a weather-resistant GPS body which makes it more versatile as compared to other trackers. Speed and position alerts are sent directly to you through text messages or email. It contains a powerful and portable battery that lasts for long. It works exclusively on the BrickHouse GPS platform. It has the best reporting coverage in the United States using the Verizon Network. The advantages of this system

It is faster, more accurate and lasts longer. It is sturdy and portable. Available for both Android and iPhone users. Free activation.

The disadvantage of this tracking system is that It only transmits information in areas with an active internet connection.

### 2.3.9 PocketFinder

PocketFinder car tracking device provides an ideal way for families and caregivers to keep track of the people they love. This car tracking device is perfect for kids going to school and seniors who value their freedom. It has prominent and outstanding features including that It contains location technology feature which uses three location technologies and GPS car tracker to give you the exact location of your vehicle. It has a low battery alarm that alerts you when the device is running out of power. The GPS tracking device keeps you updated about your car location after every one minute. The speed alert feature alerts you when the driver exceeds the speed limit set on the tracker. The Geo-fences feature is used to create several zones upon which entering and exiting are met with automatic notifications. It comes with an S.O.S Signal which activated by pressing the S.O.S button. The advantages of the system include the fact that The Position location refreshes after every sixty seconds especially when the vehicle is moving. And furthermore It is a free app with vast multiple users. In addition, you can use a single map to view various PocketFinder trackers.

The system has two disadvantages:

The Users agreement may allow them to share information with third parties

And some users may find the frequent app updates Nagy.

### 2.3.10 The AMBERalertGPS

Amber Alert GPS car tracking device is the leading provider of communication and location tracking products and services. It has some known Features one being that It has a GPS tracker which sets interval monitoring time which keeps you updated in case of any movement. The second feature being that It has a safety alert which warns you once the vehicle exits a safe area or the driver exceeds a certain set speed limit. The third feature is the system having Voice calling that allows the parent and the child to call into safety just in case an emergency occurs.

The system has a powerful battery that can sustain its power for up to 40 hours. It supports both IOS and Android operating systems. And lastly the system has a device inactivity feature that receives a warning when the device delays to report to you.

The advantage of using the system:

Is that it is compatible with both Android and IOS systems.

And it is highly durable.

The disadvantage:

The battery takes longer hours to recharge

There may be a delay in real-time tracking.

### 2.3.11 Caref GPS

Caref GPS phone watch car tracker is an excellent GPS vehicle tracker that is extremely economical, functional, and easy to use thanks to a voice over voice ability. Its features include that it supports the S.O.S alarm system. The device is water-resistant and weather-friendly. The tracking watch allows you to send and receive text messages and phone calls. The advantages of the system include that it is rich in features, such as voice-to-voice and a panic alert button. The system too has one and only disadvantage is that in the use of the system you must put an included plastic sticker over the fragile glass clock screen.

You can attach a car tracking device inside your car or connect it to the sixteen-pin. Some of these GPS vehicle trackers have monthly service subscriptions, but they all beam vital statistics to your computer, smartphone, or tablet, so you have real-time information on where your car goes. Regardless of the GPS tracker you choose to use, the fact remains that you will always stay updated regarding the location of your car.



*Figure 4 GPS Module*

### 2.3.12 WORKFLOW Vehicle Real Time Tracking (Arduino Mega).

When the GPS tracking system is installed in a car, the system will run automatically and provide data updates of the move by sending SMS messages.

Features included in this system:

* User sets their own limits: in case of car going beyond speed limit the user gets a message alert.
* When car starts moving or come to a stop the user gets an alert.
* User can shut down car engine.

According to Kunal Mauray (Kunal Maurya, 2016), the safety of private and public vehicles is a major concern, having GPS vehicle tracking system ensures safety when travelling. Police can follow signal emitted by the tracking system to locate a stolen vehicle. Vehicle tracking system are commonly used by fleet operators for fleet management functions such as routing, dispatch, on-board information and security.

## 2.4 RELATED TECHNOLOGIES.

GPS Technology.

Global Positioning System (GPS) is the only fully functional Global Navigation System (GNSS). A GPS receiver receives the signal from at least three satellites to calculate distance and uses a triangulation technique to compute its two dimension (latitude and longitude) position or at least four satellites to compute its three dimension (latitude, longitude, altitude) position. Therefore, GPS is a key technology for giving position.

GPS parameters and specifications are given below.

### 2.4.1 GSM Technology

A GSM modem is a specialized ye of modem which accepts a SIM card and operates over a subscription to a mobile operator, just like a mobile phone. GSM (Global System for Mobile) uses a process called circuit switching. This mode allows communication between two devices. Once connected a constant stream of data is streamed. GSM networks consist of three major parts: Switching System (SS), the Base Station (BSS) and the Mobile Station (MS).

1. The Switching System.

SS systems hold five databases with in it which performs different functions. If we talk about major tasks of SS system, it performs call processing and subscriber related functions. These databases from SS systems are HLR, MSC, VLR, AUC and EIR. The MSC in cooperation with Home Location register (HLR) and Visitor location register (VLR), take care of mobile calls and routing of phone calls. Authentication center (AUC) is small unit which handles the security end of the system and Equipment identity register (EIR) is another important database which holds crucial information regarding mobile equipment’s.

1. The Base Station System (BSS).

BSS are basically outdoor units which consist of iron rods and are usually of high length. BSS are responsible for connecting subscribers (MS) to mobile networks. All the communication is made in Radio transmission. The Base station System is further divided in two systems. These two systems, they are BTS and BSC. BTS (Base Transceiver station) handles communication using radio transmission with mobile station and BSC (Base station controller) creates physical link between subscriber (MS) and BTS, then manage and controls functions of it.

1. Mobile Station (subscriber).

MS consist of a mobile unit and a smart card which is also referred as a subscriber Identity Module (SIM) card. This card fitted with the GSM Modem and gives the user more personal mobility. The equipment itself is identified by a unique number known as the International Mobile Equipment Identity (IMEI).

Microcontroller

It is the interface between the GSM module and the GPS receiver. A microcontroller is a small computer on a single integrated circuit containing a processor core, data memory, A/D converter and programmable input/output peripherals. In this device the microcontroller is programmed in such a way that it stimulates the GSM modem in message forwarding when a request is send by the user. Microcontrollers are much smaller and simplified so that they can include all the functions required on a single chip. Having the microcontroller is of great use, as it has low design cost and add intelligence to the system.

## 2.5 Critics on the Previously Existing System

The Systems that have existed before worked pretty well accept that we have a little problem with their implementation and performance. Like if we highlight on one of the systems like the live track system one of its weakness is that it was in a way too expensive for lower class people to afford it. Nevertheless, the system was enormous and indiscrete compared to other GPS car

Trackers. That’s the reason we decided to come up with a cheaper way of solving this problem. Our type of system will be more of simplicity and less discrete.

The MOTO safety OBD Teenage Driving Training Car Monitoring System has the disadvantage with the application of the system which in other cases may be relatively limited in features. And the second disadvantage is that the Acceleration alerts of the system are quite sensitive. Our project is aimed at developing a simpler tracker android application in that the user only needs a smartphone with the application installed in it. The system will be owned by the user himself and will incur lesser costs or almost zero costs.

The Mileage Ace GPS Mileage Tracker has the disadvantage that the GPS may mislabel locations and it does not have a real-time tracking too. Furthermore, the stops made by the system are difficult to highlight. This even makes the system more inefficient since it is never real time and might be prone to errors too. Users or other customers would prefer a simple system that is real time and effective too. They would cherish a system that is easily adapted and perform the desired operations without failure or difficulties.

## 2.6 Summary

In conclusion, all existing technologies so far have had a huge impact on the curbing with the rampant cases of car theft. Although with the many measures that have been set apart there still have been cases reported. All the gaps in the existing technologies have proved difficult to fix. From how the data is transmitted to how it is stored and secured. To fix some of these issues the Android Tracking System solves the problem of data. The android phone gets to transmit all the data directly to the owner’s vehicle without data been transmitted to any other middle person/company. Once the data is transmitted all information is relayed to the receiver phone. This makes it easy to manage your own transmitted data as you are the soul holder of the information.

## 2.7 GAPS IN EXISTING SYSTEMS

Various gaps that have not been solved by these systems include:

1. These systems have not improved location services. There is still the question of vehicle jammers which are easily bought.
2. Current systems the data gets transmitted to the tracking companies which becomes an issue to the security information on the whereabouts of the vehicle. Any of the employees might share information of the vehicle. Thus might lead to theft as they are the soul holders of the vehicle information.

# CHAPTER 3

**SYSTEM ANALYSIS AND DESIGN**

## 3.1 Introduction

In this chapter, the following shall be covered concerning the development of the project;

1. System development methodology used
2. Feasibility Study
3. Requirements elicitation for the project
4. Data and System Analysis (v) System Specification.
5. Logical Design and Physical Design
6. System Architecture
7. Other Diagrams that may be present
8. Design Phase Report Contents.

## 3.2 Systems Development Methodology Used

In the development of the vehicle tracking system, we shall use Agile Software Development Methodology.

Agile methodology is a software development methodology by which a software development team can manage a project by breaking it up into several stages which can be managed independently.

### 3.2.1 Agile Methodology

Agile methodology is a software development methodology y

In this methodology, customers finalize requirements before development process occurs with the project manager tracking every movement of the project through each handoff and finally on to deliver.

The development team decides at the beginning of a sprint what can be accomplished in the timeframe and sets out to build a series of features, delivering working software that can be installed In a production environment at the end of the sprint.

### 3.2.2 Agile Software Methodologies.

There are a variety including;

i) Disciplined Agile Delivery(DAD) ii) Adaptive Software Development iii) Agile Modeling iv) Kanban v) Scrum vi) Scrum ban vii) Extreme Programming (XP) viii) Dynamic Systems Development (DSSM)

1. Feature Driven Development
2. Lean Software Development

### 3.2.3 The Agile Process Flow

i) Concept

This is where the project is envisioned and prioritized. ii) Inception

This is where team members are identified, funding is put in place, and initial environments are discussed.

iii) Iteration /Construction

The development team works to deliver working software based on iteration requirements and feedback. iv) Release

This is the quality testing, internal and external training, documentation development and final release of the iteration into production.

v) Production

This is the ongoing support of the software vi) Retirement

This is the end of activities, including customer notification and migration.

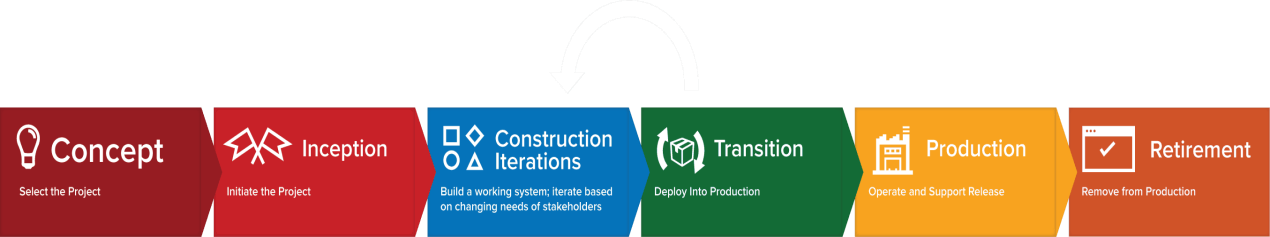


Figure 5 Stages of Agile

Agile Iteration Workflow

Agile software development consists of iterative process. Each iteration results in the next piece of the software development puzzle - working software and supporting elements, such as documentation, available for use by customers until the final product is complete. Each iteration is usually two to three weeks long and it usually has a fixed time to complete. During iteration, it is always important that customers and stakeholders provide feedback to ensure that the intended features meet their needs.

Typical iteration flow can be represented as follows;

1. Requirements

This is where you define the requirements for the iteration based on the product backlog, sprint backlog, customer and stakeholder feedback.

1. Development

This is where the software is developed basing on the defined requirements.

1. Testing

This is where the Quality Assurance testing is performed, internal and external training, documentation development

1. Delivery

This is where we integrate and deliver the working iteration into production. v) Feedback

Accept customer and stakeholder feedback and work it into the requirements of the next iteration.



**Activities to put in place to support adoption of agile workflow.**

1. Daily meetings

They should be consistent to maintain open communication, hold workers accountable and keep each iteration moving forward.

1. Live Demonstrations

It should be done to at each iterations final product to show progress.

1. Share feedback

You should receive feedback from stakeholders and share with entire team before the next iteration begins.

1. Remain Agile

Make changes to your process based on feedback to ensure each iteration improves the last.

## 3.3 Feasibility Study

Feasibility study is where the project manager assesses the strength and weakness of a proposed system to be developed, its positive and negative impacts, and the resources needed to carry out the project.

The following factors were considered when developing the vehicle tracking system;

1. Technical Feasibility

This is the assessment of the technical requirements of a project or product to find out what technical resources a project requires. The vehicle tracking system will be able to be used by all users who have basic knowledge of using an android smartphone.

All the system will require is the person to have two android smartphones.

1. Economic Feasibility

This is where a study is done to see how long it would take for the project to reach breakeven point. The capital required to complete the project is estimated. In our proposal, we will need to have carry out cost benefit analysis (CBA).

1. Legal Feasibility

This is the assessment of the project to determine whether the project meets the legal requirements that exist for implementation. Our project meets all the legal requirements of Kenya and most of the countries.

1. Operational Feasibility

This is the assessment of how well the implementation of the project fits in with the current organizational business structure. The implementation of the android car tracking system project will be easier as compared to other car tracking systems.

1. Schedule Feasibility

This is the assessment of the time the system will be completed (When the project is due), whether or not the system will be needed after that time.

## 3.4 Requirements Elicitation

This is the practice of researching and discovering the requirements of a system from user, customers, and other stakeholders.

Data Collection

It is the process of gathering data on targeted variables. We will be using questionnaires us our main method of collecting data. These will be done by posing specific questions to the targeted population. The same type of questions will be posed to all members who will participate on the process. The main reason for using this technique is that will provide quick response to questions therefore making it easy to carry out data analysis process.

## 3.5 Data and System Analysis

This is the process collecting interpreting facts, identifying the problems and decomposition of the system into its components. Data and System analysis is conducted for the purpose of studying a system its objectives.

Data collected from questionnaires will be analyzed using Microsoft excel which is a statistical tool. Main analytical presentation tools for the data will be the following.

Bar graphs people are able to interpret in an easier manner.

Pie charts this tool enables stakeholder to easily compare data and also understand quickly is it presented a nice visual manner.

## 3.6 System Specification

### 3.6.1 Functional requirements

**i.** Users are able to login into the system. **ii.** Users’ ability to track the other devices.

**iii.** Admin is able to create account for users.

### 3.6.2 Non-functional requirements

1. Usability- the system should be simple to use.
2. Reliability- the system should be able to continue to function in event of occurrence of failure.
3. Security- the system should provide authentication to avoid an authorized access.

## 3.7 System Functionality

In the android vehicle tracking system the user will be required to have two android smartphones, or several if he/she has several vehicles to trace. One device (which is retained by the user) will be used as the admin which is used to register all the other vehicles through an email addresses. When user wants to track his/her vehicles, he is required to sign in and choose the email address of the vehicle in question. He will be able to view the current location of the vehicle using Google maps.

### 3.7.1 Hardware requirements

Internet provision to establish connectivity between the remote components.

A processor from i3 generation to provide enough computations.

A hard disk of 80 GB

A minimum of 4GB RAM.

### 3.7.2 Software requirements

Microsoft Windows 8 and Above.

Firebase database which is schema less.

Android visual studio version 3 and above

## **DESIGN**

3.8 Logical and Physical design

### 3.8.1 Logical Design

**Use-Case Diagram**

### 3.8.2 Physical Design



U

ser



+



+



Create account



+



Recover password



+



Login



Add vehicle



+



+



Remove vehicle



+



Track Vehicle



+



Logout



admin

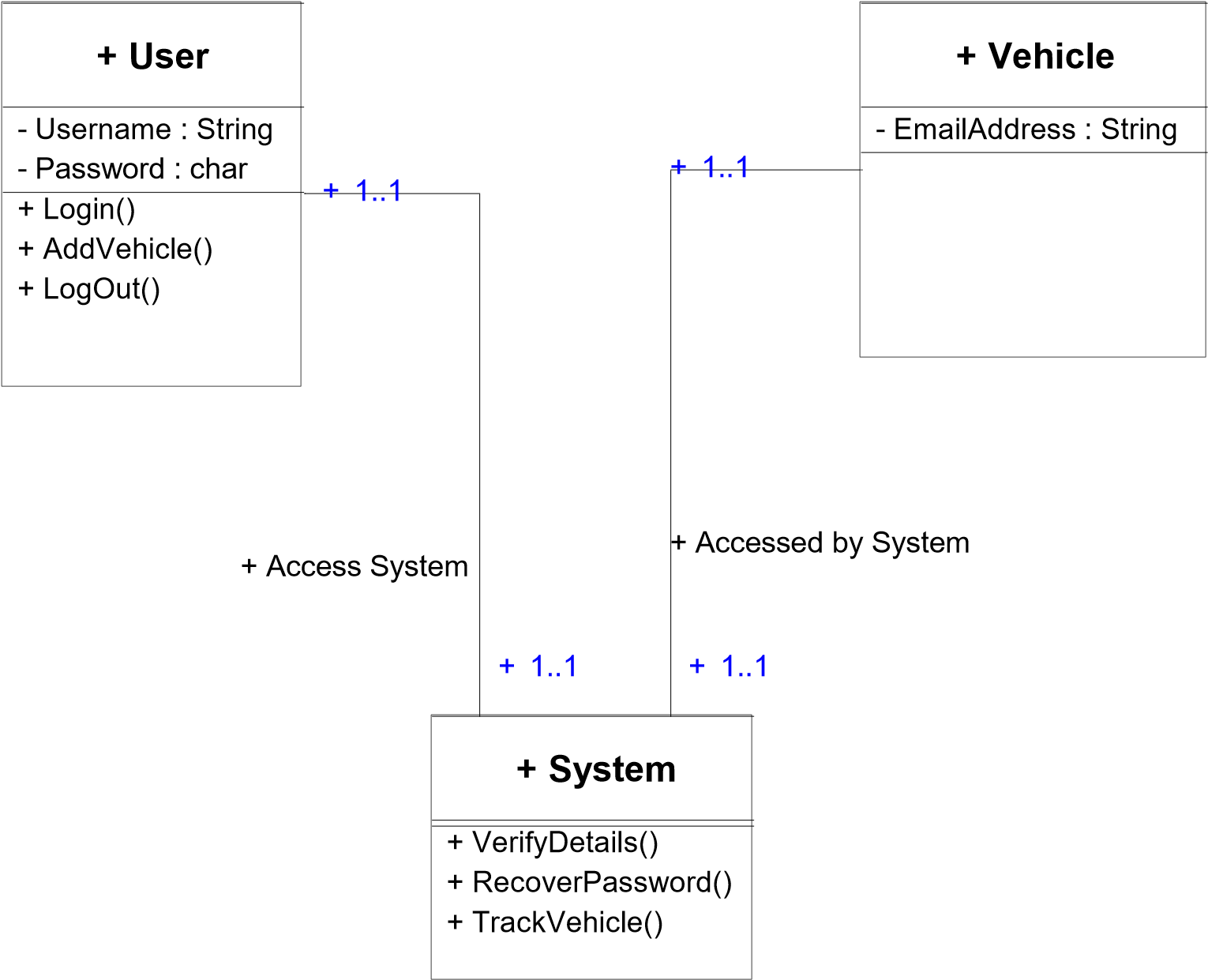


+

**Flowchart Diagram**

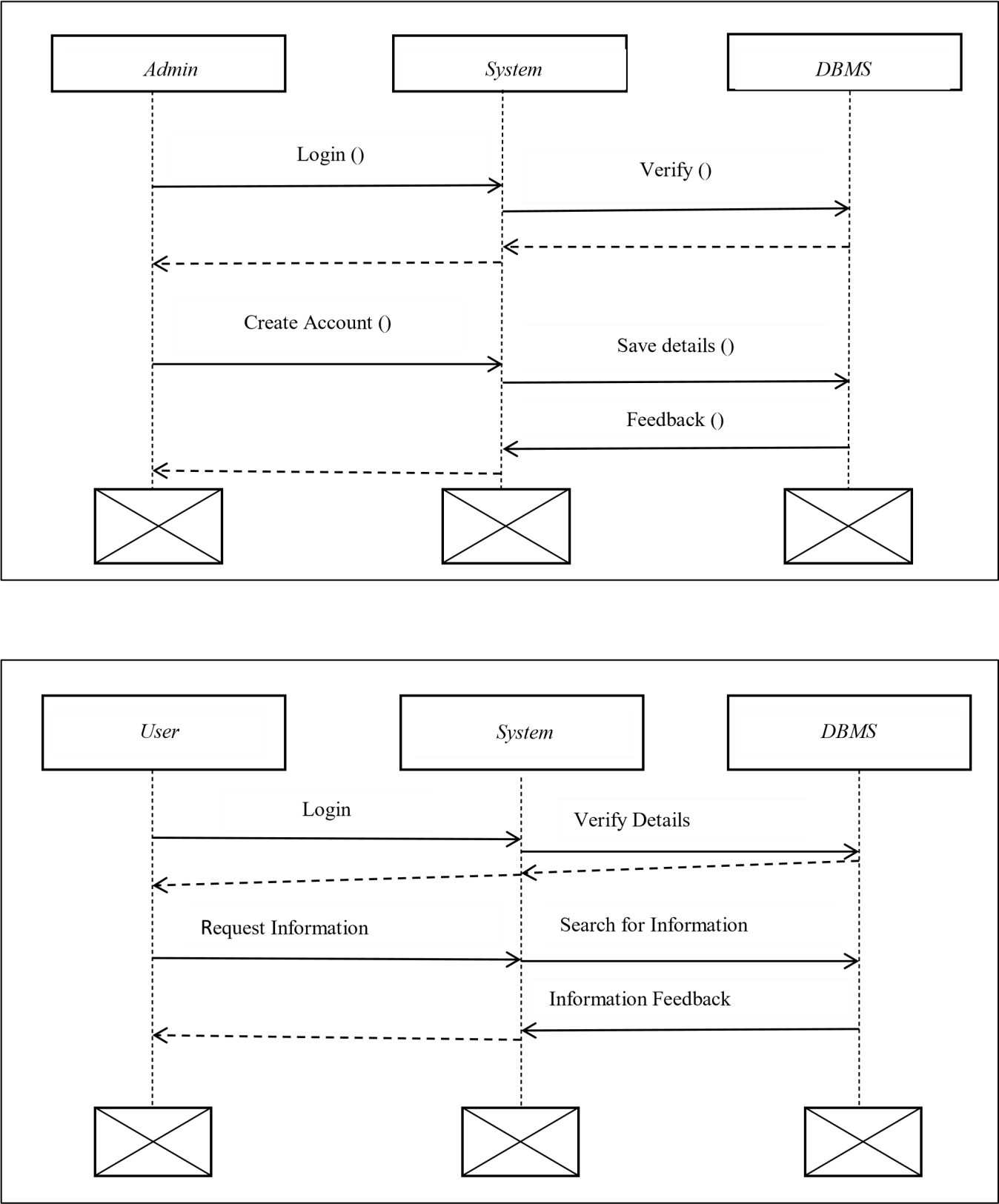
### 3.8.3 Other diagrams present

**Class Diagram**



**Activity diagrams**

They will show how the activities will take place in the system. They will show the information flow in the system and how the system will give feedback.



## 3.9 System Architecture

The main architecture that was used to design system is the two tire model because of its weighed advantages which are:

1. Security
2. Reduction of network traffic
3. Easy maintenance
4. Authorization can be implemented
5. Application services see a higher level of abstraction





Client



Application



Firebase



Vehicle



Application



Request:int(in)



Response:int(in)



Request:int(in)

# CHAPTER 4: SYSTEM IMPLEMENTATION (CODE GENERATION AND TESTING)

This chapter deals with system implementation, which involves code generation and testing of the system constructed. The system was developed using Android Studio version 3.5 as the Integrated Development Environment. An object oriented programming language was used specifically java for the development of the system. Firebase which is a schema less database system was used for the data storage and manipulation. Firebase also is a real time database which automatically updates data.

## **4.2 Construction (System Creation)**

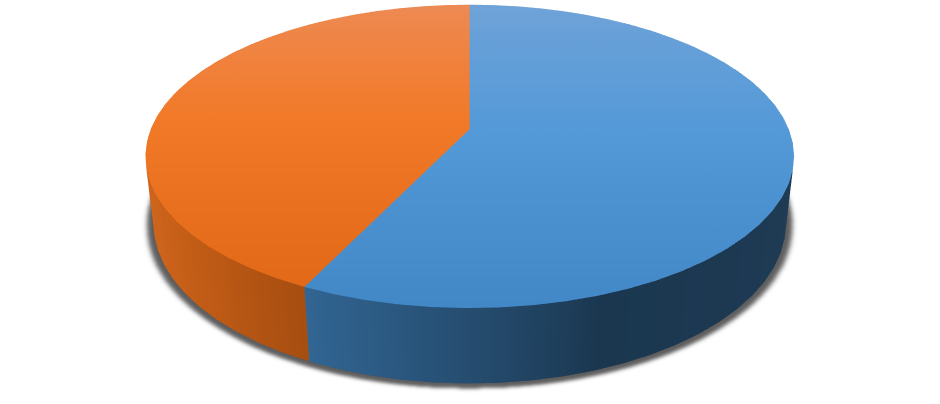
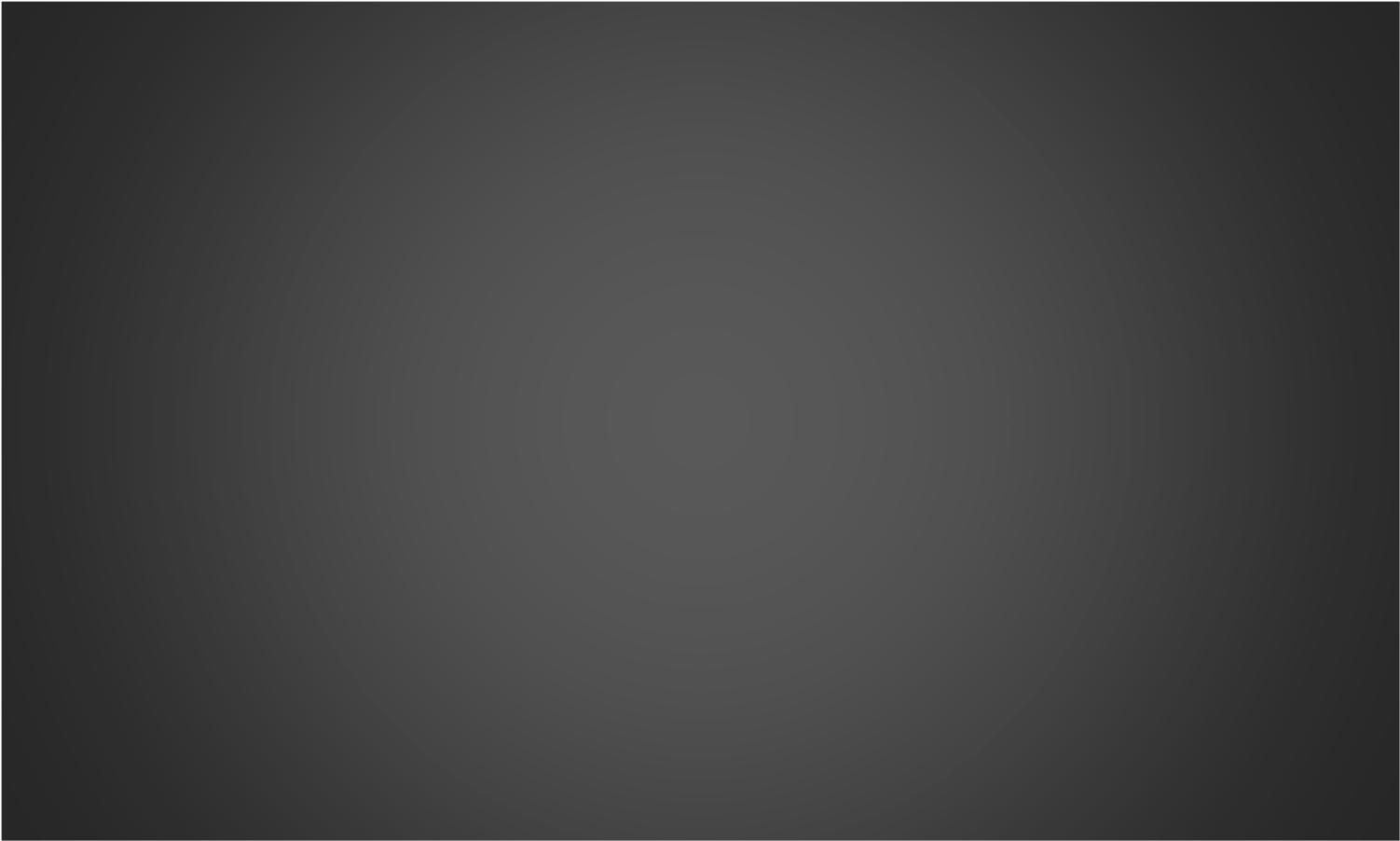
The constructions ensures development of the system meets the functional and nonfunctional requirements. Phases comprise of Code generation, Integration, Testing and documentation.

### **4.2.1 Statistical analysis**

A sample of 21 people participated in the questionnaires thought the country and results are tabulated as follows.

1. Do you or have you ever owned a vehicle?

|  |  |
| --- | --- |
| YES | 12 |
| NO | 9 |



12

,

57

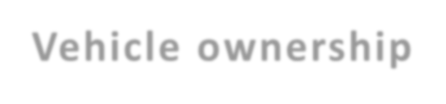
%

9

,

43

%



**Vehicle ownership**



1



2

Figure 6 Vehicle Information

Out of 21 people 57% had ever owned a vehicle or have a vehicle while 43% did not have.

1. Have you installed or use any tracking device system in your vehicle?

|  |  |
| --- | --- |
| YES | 9 |
| NO | 12 |

|  |  |
| --- | --- |
| **Prevalence of Tracking** | |
| **43**  **%**  **57**  **%** | 2  1 |
|  |

Figure 7 Prevalence of Tracking

Out of the 21, 43% had tracking installed in their vehicles while 57% had no tracking device installed in their vehicle.

1. Who carries out the tracking process?

|  |  |
| --- | --- |
| Tracking Company | 11 |
| Individual | 5 |
| Nothing | 5 |

0

2

4

6

8

10

12

1

2

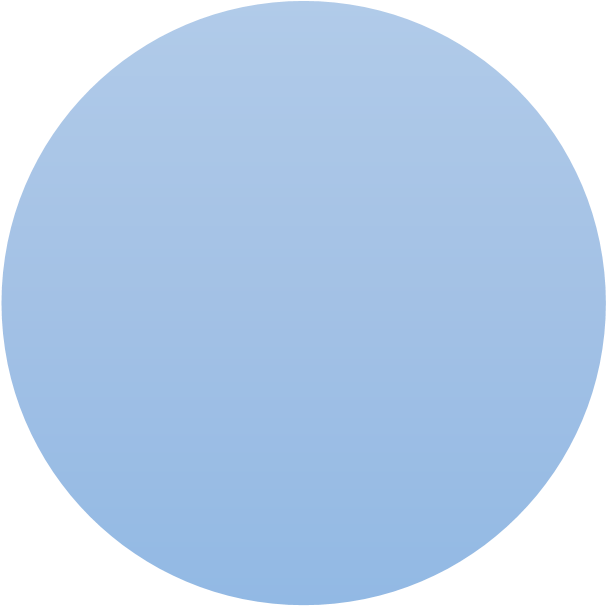
3

Tracking Operation

Most of the tracking operation was done by the tracking company.

1. Do you have android smart phone?

|  |  |
| --- | --- |
| YES | 21 |
| NO | 0 |



100

%

0

%

Android Smartphone prevalence



1



2



3

Figure 8 Smart Phone Technology Prevalence

All of the 21 people had a smartphone.

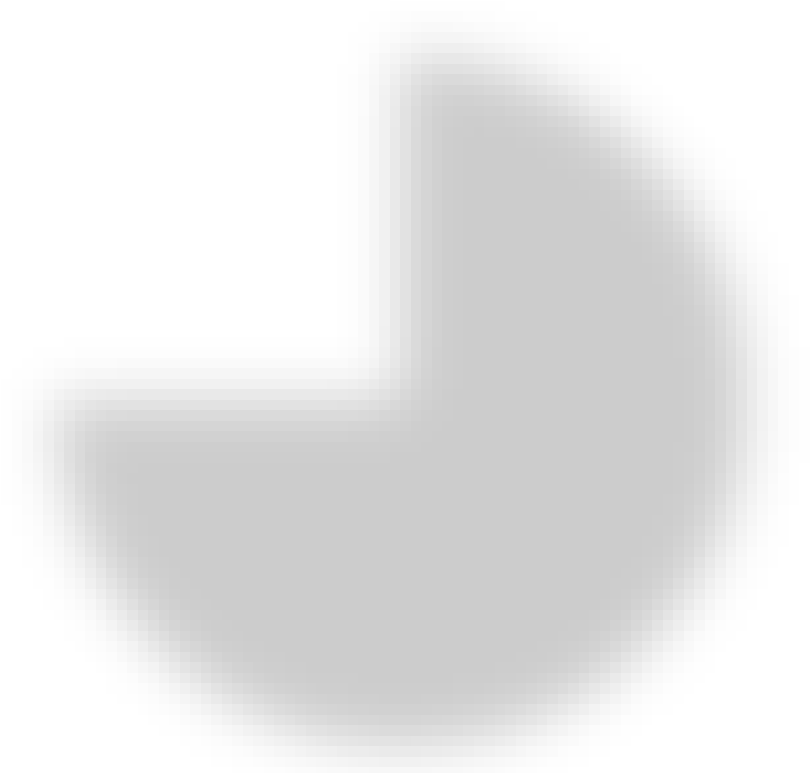
1. Do you have an active email account?

|  |  |  |  |
| --- | --- | --- | --- |
| YES | | 20 | |
| NO | | 1 | |
|  | EMAIL ENROLLMENT  25   |  |  |  | | --- | --- | --- | |  |  |  | |  |  |  | |  |  |  | |  |  |  |   20  15  10  5  0  ACTIVE MAIL INACTIVE MAIL | |  |

Figure 9 Email Users

1. What if we come up with an android based application tracking system where you can monitor/track your vehicle using smartphone at any time and place, how would you recommend it?

|  |  |
| --- | --- |
| Highly Recommended(5) | 15 |
| Recommended(4) | 1 |
| Average Recommendation(3) | 3 |
| Partial Recommendation(2) | 1 |
| Not Recommended(1) | 0 |



**%**

**75**



**%**

**5**



**15**

**%**



**5**

**%**



**0**

**%**

**Android System Recommendation**

Highly Recommended(5)

Recommended(4)

Averagely Recommended(3)

PartialLY Recommende(2)

Not Recommended(1)

Figure 10 Recommendations

Out of 21 people, 75% highly recommended an android tracking system while 15% averagely recommended development of the system, while another 5% partially recommended the development of new system.

**Login Page**

The users are required to login with their authenticated email.

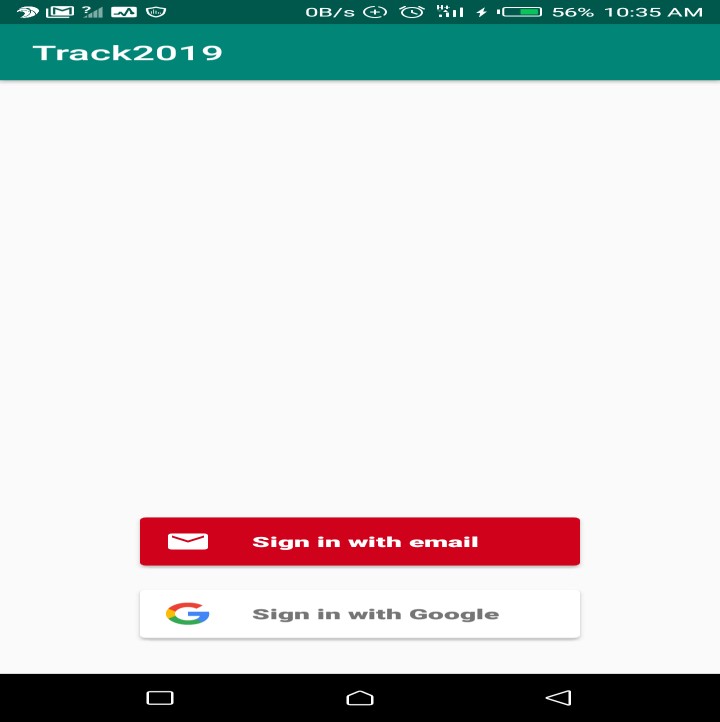


Figure 11 Login1

A user enters the password and if the user has forgotten the application will allow the user to reset the password by clicking on the Trouble signing in link.

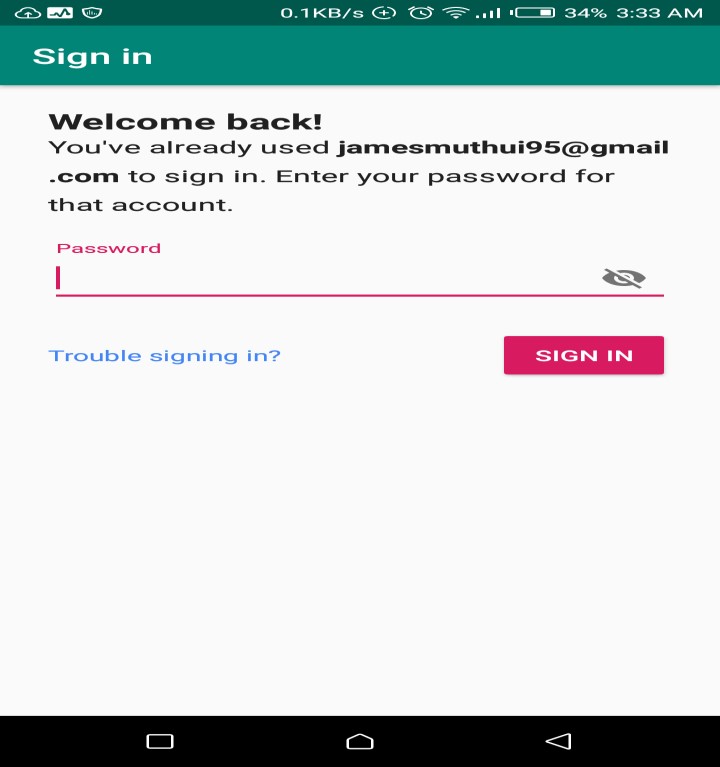


Figure 12 Login2

**Home Options**

After correct login credentials the user is shown a fragment menu which provides the user to set up tracking operation by sending tracking request to the other email.

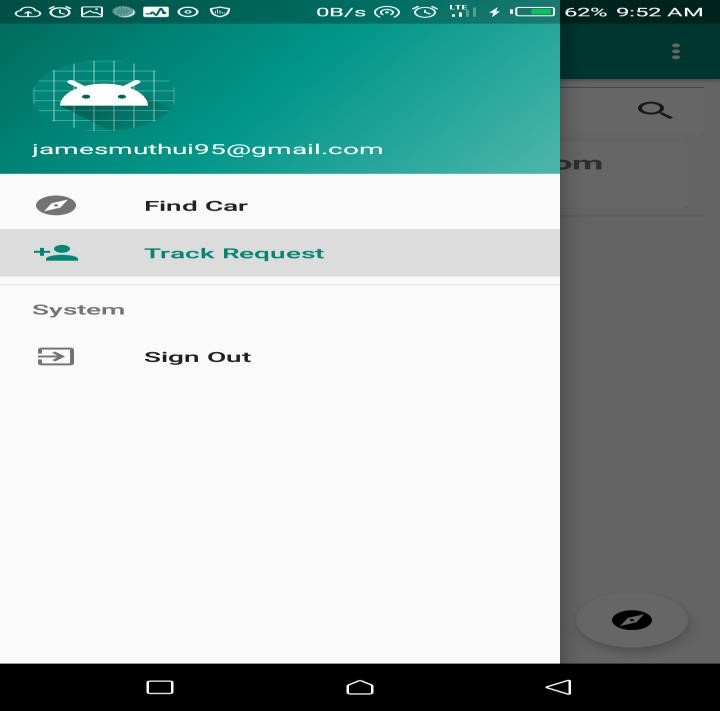


Figure 13 Menu Options

**List of emails to track**

When user finishes setting up the tracking requests, the email is added on list which is displayed on the home page.

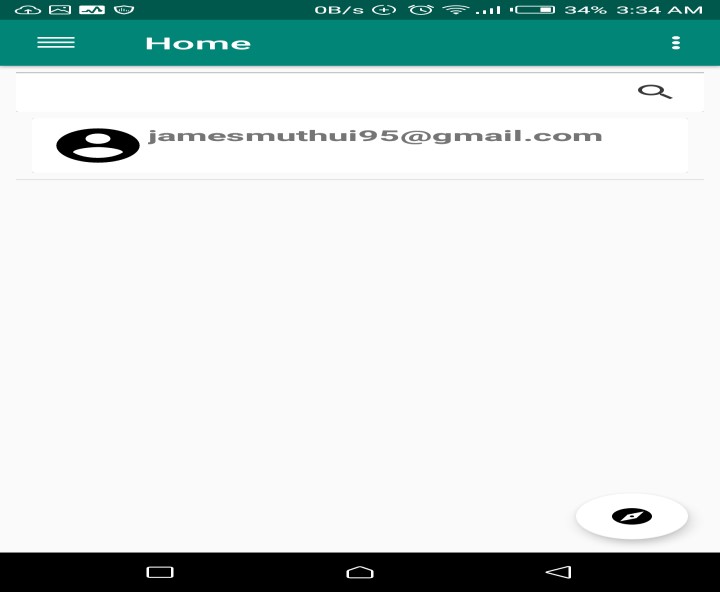


Figure 14 Home page

**Google Map Display location information on map.**

The user clicks the email on the homepage list and is redirected to the Google map where location information is displayed.

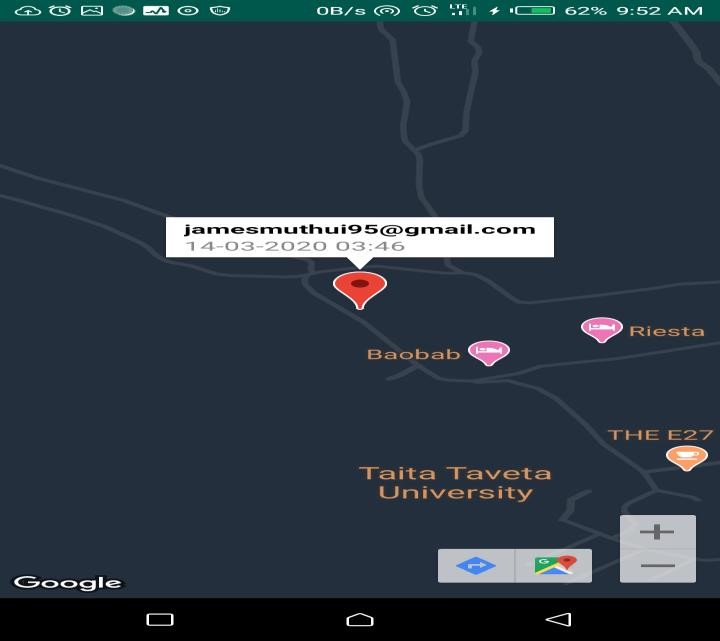


Figure 15 Tracking Info

## 4.3 Code Generation

The system consist of the front end where the client interacts with the system and backend which stores the data.

### 4.3.1 Login

**import** androidx.annotation.NonNull; **import** androidx.annotation.Nullable;

**import** androidx.appcompat.app.AppCompatActivity;

**import** android.Manifest; **import** android.content.Intent; **import** android.os.Bundle; **import** android.widget.Toast;

**import** com.example.track2019.Model.User; **import** com.example.track2019.Util.Common; **import** com.firebase.ui.auth.AuthUI; **import** com.firebase.ui.auth.IdpResponse; **import** com.google.android.gms.tasks.OnFailureListener; **import** com.google.android.gms.tasks.OnSuccessListener; **import** com.google.firebase.auth.FirebaseAuth; **import** com.google.firebase.auth.FirebaseUser; **import** com.google.firebase.database.DataSnapshot; **import** com.google.firebase.database.DatabaseError; **import** com.google.firebase.database.DatabaseReference; **import** com.google.firebase.database.FirebaseDatabase; **import** com.google.firebase.database.ValueEventListener; **import** com.google.firebase.iid.FirebaseInstanceId; **import** com.google.firebase.iid.InstanceIdResult; **import** com.karumi.dexter.Dexter; **import** com.karumi.dexter.PermissionToken;

**import** com.karumi.dexter.listener.PermissionDeniedResponse; **import** com.karumi.dexter.listener.PermissionGrantedResponse; **import** com.karumi.dexter.listener.PermissionRequest; **import** com.karumi.dexter.listener.single.PermissionListener;

**import** java.util.Arrays; **import** java.util.List; **import** java.util.Objects;

**import** io.paperdb.Paper;

**public class** MainActivity **extends** AppCompatActivity { DatabaseReference **user\_information**;

**private static final int *MY\_REQUEST\_CODE***=7117;

List<AuthUI.IdpConfig> **providers**;

@Override

**protected void** onCreate(Bundle savedInstanceState) { **super**.onCreate(savedInstanceState); setContentView(R.layout.***activity\_main***); Paper.*init*(**this**); **user\_information**=

FirebaseDatabase.*getInstance*().getReference(Common.***USER\_INFORMATION***);

*//initialise providers* **providers**= Arrays.*asList*(

**new** AuthUI.IdpConfig.EmailBuilder().build(), **new** AuthUI.IdpConfig.GoogleBuilder().build()

);

Dexter.*withActivity*(**this**)

.withPermission(Manifest.permission.***ACCESS\_FINE\_LOCATION***)

.withListener(**new** PermissionListener() {

@Override

**public void** onPermissionGranted(PermissionGrantedResponse response)

{

showSignInOptions();

}

@Override

**public void** onPermissionDenied(PermissionDeniedResponse response) { Toast.*makeText*(MainActivity.**this**, **"You must accept permission to use app"**, Toast.***LENGTH\_SHORT***).show();

}

@Override

**public void** onPermissionRationaleShouldBeShown(PermissionRequest permission, PermissionToken token) {

}

}).check();

}

**private void** showSignInOptions() { startActivityForResult( AuthUI.*getInstance*()

.createSignInIntentBuilder()

.setAvailableProviders(**providers**)

.build(),***MY\_REQUEST\_CODE***);

}

@Override

**protected void** onActivityResult(**int** requestCode, **int** resultCode, @Nullable Intent data) {

**super**.onActivityResult(requestCode, resultCode, data); **if**(requestCode == ***MY\_REQUEST\_CODE***)

{

IdpResponse idpResponse = IdpResponse.*fromResultIntent*(data); **if**(resultCode == ***RESULT\_OK***)

{

**final** FirebaseUser firebaseUser = FirebaseAuth.*getInstance*().getCurrentUser(); **user\_information**.orderByKey()

.equalTo(Objects.*requireNonNull*(firebaseUser).getUid())

.addListenerForSingleValueEvent(**new** ValueEventListener() {

@Override

**public void** onDataChange(@NonNull DataSnapshot dataSnapshot) {

**if** (dataSnapshot.getValue() == **null**)

{ **if**

(!dataSnapshot.child(firebaseUser.getUid()).exists())

{

Common.*loggedUser* = **new**

User(firebaseUser.getUid(),firebaseUser.getEmail());

**user\_information**.child(Common.*loggedUser*.getUid())

.setValue(Common.*loggedUser*);

}

} **else**

{

Common.*loggedUser* = dataSnapshot.child(firebaseUser.getUid()).getValue(User.**class**);

}

Paper.*book*().write(Common.***USER\_UID\_SAVE\_KEY***,

Common.*loggedUser*.getUid());

updateToken(firebaseUser); setupUI();

}

@Override

**public void** onCancelled(@NonNull DatabaseError databaseError) {

}

});

}

}

}

**private void** setupUI() {

startActivity(**new** Intent(MainActivity.**this**,HomeActivity.**class**)); finish();

}

**private void** updateToken(**final** FirebaseUser firebaseUser) { **final** DatabaseReference tokens = FirebaseDatabase.*getInstance*()

.getReference(Common.***TOKENS***);

FirebaseInstanceId.*getInstance*().getInstanceId()

.addOnSuccessListener(**new** OnSuccessListener<InstanceIdResult>() {

@Override

**public void** onSuccess(InstanceIdResult instanceIdResult) { tokens.child(firebaseUser.getUid())

.setValue(instanceIdResult.getToken());

}

}).addOnFailureListener(**new** OnFailureListener() {

@Override

**public void** onFailure(@NonNull Exception e) {

Toast.*makeText*(MainActivity.**this**, **""**+e.getMessage(),

Toast.***LENGTH\_SHORT***).show();

}

});

}

}

### 4.3.2 Home Activity

**package** com.example.track2019;

**import** android.Manifest; **import** android.app.PendingIntent; **import** android.content.Intent; **import** android.content.pm.PackageManager; **import** android.location.Location; **import** android.os.Bundle;

**import** com.example.track2019.Interface.IFirebaseLoadDone; **import** com.example.track2019.Interface.IRecyclerItemClickListener; **import** com.example.track2019.Model.User;

**import** com.example.track2019.Service.MyLocationReceiver; **import** com.example.track2019.Util.Common; **import** com.example.track2019.ViewHolder.UserViewHolder; **import** com.firebase.ui.database.FirebaseRecyclerAdapter; **import** com.firebase.ui.database.FirebaseRecyclerOptions; **import** com.google.android.gms.location.FusedLocationProviderClient; **import** com.google.android.gms.location.LocationRequest; **import** com.google.android.gms.location.LocationServices;

**import** com.google.android.material.floatingactionbutton.FloatingActionButton; **import** com.google.android.material.snackbar.Snackbar;

**import** android.text.Editable; **import** android.text.TextWatcher; **import** android.view.LayoutInflater; **import** android.view.MenuItem; **import** android.view.View;

**import** androidx.annotation.NonNull; **import** androidx.annotation.Nullable;

**import** androidx.appcompat.app.ActionBarDrawerToggle; **import** androidx.core.app.ActivityCompat; **import** androidx.navigation.NavController; **import** androidx.navigation.NavDestination; **import** androidx.navigation.Navigation; **import** androidx.navigation.ui.AppBarConfiguration; **import** androidx.navigation.ui.NavigationUI;

**import** com.google.android.material.navigation.NavigationView; **import** com.google.firebase.database.DataSnapshot; **import** com.google.firebase.database.DatabaseError; **import** com.google.firebase.database.DatabaseReference; **import** com.google.firebase.database.FirebaseDatabase; **import** com.google.firebase.database.Query; **import** com.google.firebase.database.ValueEventListener; **import** com.mancj.materialsearchbar.MaterialSearchBar;

**import** androidx.drawerlayout.widget.DrawerLayout;

**import** androidx.appcompat.app.AppCompatActivity; **import** androidx.appcompat.widget.Toolbar;

**import** androidx.recyclerview.widget.DividerItemDecoration; **import** androidx.recyclerview.widget.LinearLayoutManager; **import** androidx.recyclerview.widget.RecyclerView;

**import** android.view.Menu; **import** android.view.ViewGroup; **import** android.widget.TextView; **import** android.widget.Toast; **import** java.util.ArrayList; **import** java.util.List; **import** java.util.Objects;

**public class** HomeActivity **extends** AppCompatActivity **implements** IFirebaseLoadDone {

FirebaseRecyclerAdapter<User, UserViewHolder> **adapter**,**searchAdapter**;

RecyclerView **recycler\_friend\_list**;

IFirebaseLoadDone **firebaseLoadDone**;

MaterialSearchBar **searchBar**;

List<String> **suggestList** = **new** ArrayList<>();

DatabaseReference **publicLocation**;

LocationRequest **locationRequest**;

FusedLocationProviderClient **fusedLocationProviderClient**;

**private** AppBarConfiguration **mAppBarConfiguration**;

@Override

**protected void** onCreate(Bundle savedInstanceState) { **super**.onCreate(savedInstanceState); setContentView(R.layout.***activity\_home***); Toolbar toolbar = findViewById(R.id.***toolbar***); setSupportActionBar(toolbar);

FloatingActionButton fab = findViewById(R.id.***fab***); fab.setOnClickListener(**new** View.OnClickListener() {

@Override

**public void** onClick(View view) {

startActivity(**new** Intent(HomeActivity.**this**,AllPeopleActivity.**class**));

}

});

DrawerLayout drawer = findViewById(R.id.***drawer\_layout***);

NavigationView navigationView = findViewById(R.id.***nav\_view***);

View headerview= navigationView.getHeaderView(0);

TextView txt\_user\_logged

=(TextView)headerview.findViewById(R.id.***txt\_logged\_email***); txt\_user\_logged.setText(Common.*loggedUser*.getEmail());

**searchBar** =(MaterialSearchBar)findViewById(R.id.***material\_search\_bar***); **searchBar**.setCardViewElevation(10);

**searchBar**.addTextChangeListener(**new** TextWatcher() {

@Override

**public void** beforeTextChanged(CharSequence s, **int** start, **int** count, **int** after) {

}

@Override

**public void** onTextChanged(CharSequence s, **int** start, **int** before, **int** count) {

List<String> suggest = **new** ArrayList<>(); **for**(String search:**suggestList**)

{

**if**(search.toLowerCase().contains(**searchBar**.getText().toLowerCase())) suggest.add(search);

}

**searchBar**.setLastSuggestions(suggest);

}

@Override

**public void** afterTextChanged(Editable s) { } }); **searchBar**.setOnSearchActionListener(**new**

MaterialSearchBar.OnSearchActionListener() {

@Override

**public void** onSearchStateChanged(**boolean** enabled) { **if**(!enabled)

{

**if**(**adapter** != **null**)

{

**recycler\_friend\_list**.setAdapter(**adapter**);

}

}

}

@Override

**public void** onSearchConfirmed(CharSequence text) { startSearch(text.toString());

}

@Override

**public void** onButtonClicked(**int** buttonCode) {

} });

**recycler\_friend\_list** = (RecyclerView)findViewById(R.id.***recycler\_friend\_list***); **recycler\_friend\_list**.setHasFixedSize(**true**);

RecyclerView.LayoutManager layoutManager= **new** LinearLayoutManager(**this**); **recycler\_friend\_list**.setLayoutManager(layoutManager); **recycler\_friend\_list**.addItemDecoration(**new**

DividerItemDecoration(**this**,((LinearLayoutManager)layoutManager).getOrientation()));

*//publicLocation*

*=FirebaseDatabase.getInstance().getReference(Common.PUBLIC\_LOCATION);* updateLocation(); **firebaseLoadDone** = **this**; loadFriendList(); loadSearchData();

*// Passing each menu ID as a set of Ids because each // menu should be considered as top level destinations.* **mAppBarConfiguration** = **new** AppBarConfiguration.Builder( R.id.***nav\_home***, R.id.***nav\_gallery***, R.id.***nav\_slideshow***,

R.id.***nav\_tools***, R.id.***nav\_share***, R.id.***nav\_send***)

.setDrawerLayout(drawer)

.build();

NavController navController = Navigation.*findNavController*(**this**,

R.id.***nav\_host\_fragment***);

NavigationUI.*setupActionBarWithNavController*(**this**, navController, **mAppBarConfiguration**);

NavigationUI.*setupWithNavController*(navigationView, navController);

navController.addOnDestinationChangedListener(**new**

NavController.OnDestinationChangedListener() {

@Override

**public void** onDestinationChanged(@NonNull NavController controller,

@NonNull NavDestination destination, @Nullable Bundle arguments) { **if**(destination.getId()== R.id.***nav\_home***)

{

startActivity(**new**

Intent(HomeActivity.**this**,AllPeopleActivity.**class**));

} **else if**(destination.getId()== R.id.***nav\_gallery*** )

{

startActivity(**new**

Intent(HomeActivity.**this**,FriendRequestActivity.**class**));

}**else if**(destination.getId()== R.id.***nav\_share***)

{

Toast.*makeText*(HomeActivity.**this**, **"Signing out"**,

Toast.***LENGTH\_SHORT***).show();

}

}

});

}

**private void** loadSearchData() {

**final** List<String > lstUserEmail = **new** ArrayList<>();

DatabaseReference userList= FirebaseDatabase.*getInstance*()

.getReference(Common.***USER\_INFORMATION***)

.child(Common.*loggedUser*.getUid()) .child(Common.***ACCEPT\_LIST***);

userList.addListenerForSingleValueEvent(**new** ValueEventListener() {

@Override

**public void** onDataChange(@NonNull DataSnapshot dataSnapshot) { **for**(DataSnapshot userSnapshot:dataSnapshot.getChildren()){ User user = userSnapshot.getValue(User.**class**);

lstUserEmail.add(Objects.*requireNonNull*(user).getEmail());

}

**firebaseLoadDone**.onFirebaseLoadUserNameDone(lstUserEmail);

}

@Override

**public void** onCancelled(@NonNull DatabaseError databaseError) { **firebaseLoadDone**.onFirebaseLoadFailed(databaseError.getMessage());

}

});

}

**private void** loadFriendList() {

Query query= FirebaseDatabase.*getInstance*()

.getReference(Common.***USER\_INFORMATION***)

.child(Common.*loggedUser*.getUid()) .child(Common.***ACCEPT\_LIST***);

FirebaseRecyclerOptions<User> options= **new**

FirebaseRecyclerOptions.Builder<User>().setQuery(query,User.**class**).build(); **adapter** = **new** FirebaseRecyclerAdapter<User, UserViewHolder>(options) {

@Override

**protected void** onBindViewHolder(@NonNull UserViewHolder holder, **int** position, @NonNull **final** User model) {

holder.**text\_user\_email**.setText(**new** StringBuilder(model.getEmail())); holder.setiRecyclerItemClickListener(**new** IRecyclerItemClickListener() {

@Override

**public void** onItemClickListener(View view, **int** position) { Common.*trackingUser* = model;

startActivity(**new** Intent(HomeActivity.**this**,

TrackingActivity.**class**));

}

});

}

@NonNull @Override

**public** UserViewHolder onCreateViewHolder(@NonNull ViewGroup parent, **int** viewType) {

View itemView=

LayoutInflater.*from*(parent.getContext()).inflate(R.layout.***layout\_user***,parent,**false**); **return new** UserViewHolder(itemView);

} };

**adapter**.startListening();

**recycler\_friend\_list**.setAdapter(**adapter**);

}

@Override

**protected void** onStop() { **if**(**adapter** !=**null**) **adapter**.stopListening(); **if**(**searchAdapter** !=**null**) **searchAdapter**.stopListening(); **super**.onStop();

}

@Override

**protected void** onResume() { **super**.onResume(); **if**(**adapter** !=**null**) **adapter**.startListening(); **if**(**searchAdapter** !=**null**)

**searchAdapter**.startListening();

}

**private void** updateLocation() { buildLocationRequest(); **fusedLocationProviderClient** =

LocationServices.*getFusedLocationProviderClient*(**this**);

**if**(ActivityCompat.*checkSelfPermission*(**this**,Manifest.permission.***ACCESS\_FINE\_LOCATION***)

!=PackageManager.***PERMISSION\_GRANTED***)

{ **return**;

} **fusedLocationProviderClient**.requestLocationUpdates(**locationRequest**,getPendingIntent());

}

**private** PendingIntent getPendingIntent() {

Intent intent = **new** Intent(HomeActivity.**this**,MyLocationReceiver.**class**); intent.setAction(MyLocationReceiver.***ACTION***); **return**

PendingIntent.*getBroadcast*(**this**,0,intent,PendingIntent.***FLAG\_UPDATE\_CURRENT***);

}

**private void** buildLocationRequest() {

**locationRequest** = **new** LocationRequest(); **locationRequest**.setSmallestDisplacement(10f); **locationRequest**.setFastestInterval(3000); **locationRequest**.setInterval(5000);

**locationRequest**.setPriority(LocationRequest.***PRIORITY\_HIGH\_ACCURACY***);

}

**private void** startSearch(String search\_value) { Query query= FirebaseDatabase.*getInstance*()

.getReference(Common.***USER\_INFORMATION***)

.child(Common.*loggedUser*.getUid())

.child(Common.***ACCEPT\_LIST***)

.orderByChild(**"name"**)

.startAt(search\_value);

FirebaseRecyclerOptions<User> options= **new**

FirebaseRecyclerOptions.Builder<User>().setQuery(query,User.**class**).build(); **searchAdapter**= **new** FirebaseRecyclerAdapter<User, UserViewHolder>(options) {

@Override

**protected void** onBindViewHolder(@NonNull UserViewHolder holder, **int** position, @NonNull **final** User model) {

holder.**text\_user\_email**.setText(**new** StringBuilder(model.getEmail())); holder.setiRecyclerItemClickListener(**new** IRecyclerItemClickListener() {

@Override

**public void** onItemClickListener(View view, **int** position) { Common.*trackingUser* = model;

startActivity(**new** Intent(HomeActivity.**this**,

TrackingActivity.**class**));

}

});

}

@NonNull @Override

**public** UserViewHolder onCreateViewHolder(@NonNull ViewGroup parent, **int** viewType) {

View itemView=

LayoutInflater.*from*(parent.getContext()).inflate(R.layout.***layout\_user***,parent,**false**); **return new** UserViewHolder(itemView);

} };

**searchAdapter**.startListening(); **recycler\_friend\_list**.setAdapter(**adapter**);

}

@Override

**public boolean** onCreateOptionsMenu(Menu menu) {

*// Inflate the menu; this adds items to the action bar if it is present.* getMenuInflater().inflate(R.menu.***home***, menu); **return true**;

}

@Override

**public boolean** onSupportNavigateUp() {

NavController navController =

Navigation.*findNavController*(**this**,R.id.***nav\_host\_fragment***);

**return** NavigationUI.*navigateUp*(navController, **mAppBarConfiguration**)

|| **super**.onSupportNavigateUp();

}

@Override

**public void** onFirebaseLoadUserNameDone(List<String> lstEmail) { **searchBar**.setLastSuggestions(lstEmail);

}

@Override

**public void** onFirebaseLoadFailed(String message) {

Toast.*makeText*(**this**, message, Toast.***LENGTH\_SHORT***).show();

}

}

## 4.4 Testing

This is a critical practice in the development as it aims at identifying bugs and errors in the system and fixing them before the system is fully developed. This aids in reducing the added cost of fixing them after implementation. Additionally, it ensures the development of a complete system that maintains data integrity and competence. Indeed, some of the testing measure taken included unit testing and system testing.

### 4.4.1 Unit Testing

In this testing stage every module was tested individually to find out if they are all working as expected, any error noticed was fixed to allow no future errors in integration testing. The table below shows results obtained on testing of the various parts of the system.

|  |  |  |  |
| --- | --- | --- | --- |
| Module Title | Test Goal | Expected Results | Success out of 10 trials |
| Login module | To verify if a registered user can login to the android application | Should allow the user to enter the site after provision of a valid email and correct password | 8 |
| Registration module | To verify if a new entry can be registered as a  new member by the database administrator. | Should register the user after the provision of an email, password and confirmation of the password should be done. | 9 |
| Password reset module | To test if the user can reset password if forgotten. | Should enable a user to reset password only if authorized in the database. | 7 |
| Home fragment module | To test if the user can successfully view the available services provided. | Should allow the user to successfully access the tracking options | 6 |
| Track request module | To test if the user can pair devices using an email. | Should allow the user to pair the two devices. | 7 |
| Track /Google map module | To test if a user can view the location of the vehicle on a map. | Should allow the user to view the track email on map. | 10 |

### 4.4.2 Integration Testing

Here all system modules were combined and tested as a group. This test was achieved by facilitating the needed links so the user could navigate with ease through the system. This test was to expose faults in the interaction between integrated units. To view those parts work best together to achieve a common goal.

# CHAPTER 5: SUMMARY, RECOMMENDATION AND CONCLUSION

## 5.1 Introduction

This is a summary of the system; it gives the constraints of the developed system, its future advancement, recommendation and a conclusion.

## 5.2 System constraints

In the development of the project the developer encountered some challenges or limitations. The following were some of the constraints or challenges encountered.

Many of the older models vehicles don’t use smart devices that will allow installation of the android app. Therefore require new configuration which include installation of smart device that will run the tracking app which will lead to increase cost of installation. Due to lack of this devices the app could only be simulated using android phones.

Another constraint is the developer was required to enhance his skills by learning a new programming language which would be used to develop the application, which was a quite hectic process. The application also depended on network connectivity therefore enough bandwidth is required.

## 5.3 Future enhancement

This the application can only run on android devices, we will seek to develop the application so as to be supported by IOS operating system. The application can be enhanced to support other functionalities such as monitoring fuel gauge, lock the car and other more core functionalities.

## 5.4 Recommendations

We recommend that cars being developed should have smart devices installed in them so as to support android auto which will allow developers come up with off the market application that will enhance the security of vehicles. Also this cars should provide satellite connectivity which will provide a wide coverage thus making tracking of cars effective and efficient.

## 5.5 Conclusion

Due to increased penetration of technology and advancement, it has led to great innovation in many sectors thus making life easier and efficient. Mostly in the area of mobile technology which has allowed us to develop applications that can be accessed with a push of a button such as tracking of vehicles with our hand held devices and thus urging embracement of this technology.

# References

Alshamsi, H. S. (2017). Real Time Vehicle Tracking Using Arduino Mega. *Internional Journal of Science and Technology(IJST)*.

Ambade shruti Dinkar, S. A. (2011). Desing and Implementation of Vehicle tracking system using GPS. *Journal of Information Engineering and Apllications*.

Daghar, M. (2019, October 15). *East Africa*. Retrieved from enactafrica: https://enactafrica.org/research/trend-reports/car-theft-in-east-africa-on-the-rise

Kejitan, V. (2019, September 26). *Ureport*. Retrieved from standardmedia.co.ke:

https://www.standardmedia.co.ke

Kiranmayi, D. (2016). Vehicle Monitoring System Using RFID. *(IJCSIT) International Journal of Computer Science and Information Technologies*.

KNBS. (2017). *CEIC*. Retrieved from https://www.ceicdata.com/en/kenya/road-transportnumber-of-motor-vehicles-registered/road-transport-no-of-motor-vehicles-registered: /en/kenya/road-transport-number-of-motor-vehicles-registered/road-transport-no-ofmotor-vehicles-registered

Kunal Maurya, M. S. (2016). Real Time Vehicle Tracking System using GPS and GSM Technology - An Anti-Theft Tracking System. *Internation Journal of Electronics and Computer Science Engineering*.

service, k. p. (2018). *2018 Annual Crime Report.*

*WIKIPEDIA*. (2017, October). Retrieved from GLONASS:

WWW.WIKIPEDIA.COM/GLONASS

wikipedia. (n.d.). *vihecle tracking system*. Retrieved from wikipedia: www.wikipedia.com/vehicle tracking system

# APPENDICES

# APPENDIX A: QUESTIONNARE

1. Do you or have you ever owned a vehicle?

Yes

No

1. Have you installed or use any tracking device system in your vehicle?

Yes

No

1. Who carries out the tracking process?

Individual

Owner

1. Do you have android smart phone?

Yes

No

1. Do you have an active email account?

Yes

No

1. What if we come up with an android based application tracking system where you can monitor/track your vehicle using smartphone at any time and place, how would you recommend it?

Highly Recommended

Recommended

Average

Not Recommended

# Appendix B: BUDGET

|  |  |
| --- | --- |
| Description | Amount (Kshs) |
| Printing and photocopying | 200 |
| Computer | 42,000 |
| Communication | 2,500 |
| Labor | 70,000 |
| Total | 114,700 |

# APPENDIX C: GANTT CHART FOR PROPOSED ACTIVITIES

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **ACTIVITY** |  |  | **DURATION** | | | |  |  |
|  | SEP | OCT | NOV | DEC | JAN | FEB | MARCH | APRIL |
| Proposal writing and submission |  |  |  |  |  |  |  |  |
| Drafting project and requirements |  |  |  |  |  |  |  |  |
| System Analysis (Requirement elicitation) |  |  |  |  |  |  |  |  |
| System Design |  |  |  |  |  |  |  |  |
| Implementation  (Coding) |  |  |  |  |  |  |  |  |
| Evaluation/Testing |  |  |  |  |  |  |  |  |
| Work submission |  |  |  |  |  |  |  |  |
| Project  Documentation |  |  |  |  |  |  |  |  |