ASHESI UNIVERSITY

Location Determination Algorithms

CAPSTONE PROJECT

B.Sc. Computer Engineering

Kelvin Ampene

2020

ASHESI UNIVERSITY

Location Determination Algorithms

CAPSTONE PROJECT

Capstone Project submitted to the Department of Engineering, Ashesi University in partial fulfilment of the requirements for the award of Bachelor of Science degree in Computer Engineering.

Kelvin Ampene

2020

DECLARATION

I hereby declare that this capstone is the result of my own original work and that no part of it has been presented for another degree in this university or elsewhere.

Candidate's Signature: K.A

…………………………………………………………………………………………

Candidate's Name: Kelvin Ampene

…………………………………………………………………………………………

Date: 05/29/2020

…………………………………………………………………………………………

I hereby declare that preparation and presentation of this capstone were supervised in accordance with the guidelines on supervision of capstone laid down by Ashesi University

Supervisor's Signature:

…………………………………………………………………………………………

Supervisor's Name:

…………………………………………………………………………………………

Date:

…………………………………………………………………………………………

Acknowledgements

To my supervisor, Dr. Nathan Amanquah whose encouragement and academic advice helped me undertake this project.

Abstract

The public transport industry is an essential part of people's lives. However, the problem associated with the local public transport industry is, travellers usually get frustrated when waiting for a bus to arrive at a bus stop due to uncertainty on when the bus may arrive or if it may arrive at all. This leads to anxiety on the part of the traveller as well as time wastage, especially if this was not planned for in the trip. To solve this, historically GPS receivers are used in conjunction with GSM/GPRS modules or RFID technologies to keep track of the location of the buses. The caveat here is this results in a relatively expensive solution to a fundamental problem travellers face. This is because of associated costs in this solution; namely, the cost of equipping each vehicle with a GPS module and charges incurred on bus operators as these modules send the bus data to a cloud data server before being queried by travellers' devices. In this project, alternative technologies and location determination algorithms are explored to propose a cost-effective solution using LoRa technology and a variation of the trilateration algorithm to determine the location of a bus. The results from the project confirm the feasibility of using alternative methods to determine vehicle location.

Table of Content

Chapter 1: Introduction

1.1 Background

In Ghana, the “Trotro” is the most used form of transport in the country. For many years this has been the case. However, the “Trotro” system is not without its flaws; these vehicles are not tracked; neither do they follow any schedule; thus, it is highly unreliable and not dependable. Empirical evidence suggests that the time travellers spend waiting at bus stops and stations waiting for transportation compared to the time they spend on the commute is very tiresome. This is down to the amount of uncertainty involved in waiting for a bus. Being able to know the location of the bus enables travellers to manage their time and improve efficiency overall. The creation of a unified bus tracking system will lead to rider satisfaction as well as more riders on the roads, thus opening doors to new ventures which can capitalize on this new infrastructure.

1.2 Problem Definition

Road commuters without prior information about the bus or route they are about to embark on often waste time which affects their productivity.

Some of the challenges that arise from this problem are:

- No way to determine the arrival time of a bus
- Difficulty in planning multi-bus trips
- No accountability on the part of the “Trotro” drivers as the buses are not tracked

This project shall deliver a cost-effective alternative tracking means for travellers who use public transport, making use of alternative wireless technology and specialized location determination algorithms. This solution will allow users to estimate the arrival time of a bus, plan trips by knowing where buses are and finally make drivers accountable since their vehicles are tracked and whereabouts will always be remembered

1.3 Justification / Motivation of Project Topic

According to Mohamed A. Abdel-ATY et al., it has been reported that “38% of passengers who do not make use of public transport may consider using public transport systems if the appropriate bus information is made available and is easily accessible” [3]. This study opens a world of possibilities for more inflow into a sector with untapped potential. Today in the leading countries around the world technology is made use of in many areas such as the transport industry where public transport provides users with real-time bus location and arrival estimations. The creation of a system to track buses in our public transport industry will lead to an increase in efficiency throughout the masses leading to a potential rise in productivity levels among commuters through the reduction of time wasted in planning a commute.

1.4 Scope of Work

For this project, the scope is to build a tracking system for the public transport industry in Ghana, using vehicles on the Ashesi University campus as a case study. This will be a test of the application in a basic format to test the feasibility of the proposed solution to be deployed throughout the public transport industry. The system to be developed shall consist of a network of base stations each made up of a wireless gateway, to receive data transmitted from the moving vehicles for location determination. A Web application would then be designed and built to allow users to track buses in real-time for route planning. The system to be developed shall consist of the following features:

- A device to be placed on vehicles to transmit bus data to base stations
- A GSM module to be used at base stations to transmit bus data to the cloud-based server for location determination
- A cloud-based web application through which users can access bus geographical location, thus using this to estimate time of arrival.

Chapter 2 of this paper looks at other research conducted in the various technologies and industries discussed and used in this project. In Chapter 3, some of the design decisions are explored as well as the complete design of the system with an inference into the facts which led to the selection of used technologies. The next chapter walks through the various steps undertaken to achieve the desired results. Chapter 5 analyses, the results obtained from the implementation and lastly the final chapter evaluates the results and looks at some constraints and limitations of the work done and suggests some future modifications which would improve the work done.

Chapter 2: Literature Review

2.1 The current TroTro System in Ghana

The Trotro system is one of three primary forms of public transportation in Ghana; the others include the MetroMass transit system and company-owned buses for the staff of companies. Each of these systems has different modes of operations. For this work, the focus is on the Trotro system of Ghana.

Buses owned by private citizens are made available to be used to serve the general public. In this arrangement, the drivers and their conductors, usually known as “mates” decide on a start and endpoint on which they would be working. There are local regulatory bodies like the bus station operators, the driver’s association and the taxi ranks to which the drivers have consultations with to plan out their routes. Depending on the selected route, these busses associate themselves with a bus station from which to load up passengers or drive along their route chosen to fill up the bus from various bus stops along their journey. This system, due to the independence of the owners and drivers, allows a lot of citizens and buses to participate in the system, making public transportation cheap while increasing the availability of the public transportation system. Despite these advantages, this system tends to pose some significant challenges. The Trotro system is very unreliable as there is no means of knowing when the next bus to a destination would arrive or the route it would use. This causes passengers to spend much time waiting for the next bus that would be going their way, and this could take between several minutes to hours, as the buses have no frequency with which they operate. This makes the planning of one’s route to a location very difficult as there is no information as to the available buses and how far they are to the next bus stop. However, despite these issues, the independent nature of this system, allows for private individuals to invest in the public transportation system of the country easily due to its low bar to enter.

In many first world countries, the provision of a robust transportation system has been one of the priorities of the government [4]. In Ghana, the MetroMass transit system was established by the government in 2001 to ensure that Ghanaians would have a means of transport which would be dependable and cost-effective [5]. With this system, the Ghanaian public was able to obtain many benefits as students in uniform and adults of age 65 years and above could use the system for free [6]. According to research conducted on user satisfaction with this system, some advantages of it were affordability as it charged about 5% less than the other systems [7]. However, In the study, of the many problems identified by passengers concerning the transportation system, the irregular movement of the buses and the inadequate bus information were the highest-ranking, respectively.

2.2 Location Determination Technologies

The problem of providing real-time bus/transport location data to passengers has been explored in the past. Today the standard way to keep track of buses for transport is by using GPS and cellular technologies. The use of GPS is based on satellite communication which can be rather expensive when used in mass production.

The most common service used by passengers for the tracking of buses in some select cities Accra not inclusive is a service provided within Google Maps which allows users to track buses. This service from Google depends on GPS and cellular technologies. The limitation of this is the service from Google is only available in some select cities.

With this limitation and cost factor involved with Google's implementation, there has been a need for several projects in this field.

GPS is a very accurate method for determination of the bus location in transit, a GPS receiver measures the distance between itself and multiple satellites and using a technique called triangulation the location of the object is computed. Real-time web applications have been developed where tracking makes use of GPS.[8] That project [8] and other similar projects that make use of SMS based systems to track buses using GPS [9], all fundamentally have the same problem the cost, which this project seeks to reduce drastically. The cost involved in placing GPS modules in each bus as well as the cellular connection needed for it to function appropriately and send data to the server is relatively high.

Aside these methods other projects with the same aim have been tackled without the use of GPS but rather wireless communication technologies such as Bluetooth, Wi-Fi, microwave, Infrared Radiation (IR) and High Frequency (HF).

Each of these technologies has specific uses and applications; the use of LoRa has been explored to sync the buses with the last bus stop that has been passed. This is a more straightforward method of handling the tracking issue but does not provide accurate pinpoint locations of where the bus is.[10]

Furthermore, it has been shown that with the use of LoRa and geometry methods, there is no need to place GPS and GSM modules in each bus [11]. Because only one is necessary as the centre point for data transfer to the server, thus showing how the use of LoRa for tracking can be very cost-effective and power-efficient. That paper proposes a solution which makes use of a LoRa module and a LoRaWAN gateway. The gateway forwards packets received through the LoRa module via UDP/IP (User Datagram Protocol), the time of Arrival technique as well as a server with a Multilateration algorithm is used to process the data and passes along the results through the Message Query Telemetry Transport (MQTT) client. Finally, a JAVA based web application uses the data from the server to estimate the position of the bus. This is a rather elegant approach to solving the problem at hand. However, the use of the LoRaWAN gateway limits the ability of the solution especially if a gateway was to go down; also, the use of JAVA to develop the system limits the use of Google map APIs for further work regarding integration. This is because JAVA does not have a custom Google maps APIs thus limiting the features it can use.

In another similar bit of work, the proposed tracking system is made of three components; the Bus unit, Central unit and a Client-side application [12]. The bus unit contains a GPS module which transmits the location details such as the ID of the bus, longitude, latitude and timestamp every fixed interval of time to the central server through a GSM/GPRS module. The server acts as a central repository which the client application connects to read the location details of a bus which has been queried by the user using the ID. This is a simple implementation of the solution for tracking buses.

Various countries and companies adopt different techniques through which they successfully track vehicles or buses. Some of these methods include image processing-based tracking, Radio Frequency Identification (RFID) based techniques for vehicle tracking, signpost technology and Global Positioning System (GPS) based tracking [2]. Transportation technology companies such as Uber, Taxify and Lyft make use of Global Positioning System (GPS) in smart devices, to determine both a passenger's location and the location of a vehicle (through the driver's smart device) [14]. However, the average GPS device faces problems with interference and getting a clear line of sight in buildings, tunnels or around tall buildings. This problem is yet solved with an improved version of GPS known as differential GPS which has mechanisms in place to improve accuracy and location detection by calculating a user's position relative to another GPS receiver at a known location [13]. To enhance real-time GPS tracking systems, Hind A. A. Dafallah, designed and built a GPS tracking system making use of the Garmin 18 – 5 Hz GPS receiver which has differential GPS capabilities and thus improves the accuracy of readings [15].

However, from the projects described above a common theme in execution is the use of GPS technologies or satellite technologies to track these vehicles. This is a costly way of solving this problem because of the cost associated with equipping each moving vehicle with a GPS module. These associated costs begin to add up for each car.

2.3 Wireless Technologies

This project has several parts which are discussed in greater detail in later chapters. A central aspect of the project, however, is the ability of the bus to send wireless signals which are used in computing the location of a bus based on the selected algorithm. There are many wireless technologies such as Bluetooth, LoRa, Wi-Fi and GPS that can send these wireless signals. GPS is widely used when it comes to tracking vehicles [1], due to its high range and fidelity, unlike Bluetooth, but its power consumption is high. Wi-Fi technology, on the other hand, has good range but its use in this project is hampered by the fact that in Ghana we do not have enough Wi-Fi hubs to make a feasible solution based on Wi-Fi. LoRa technology provides a healthy range and low power consumption, but its signals are easily affected by the presence of obstacles such as mountains and buildings.

Before selecting a wireless technology for this project, it is worth comparing the technologies in terms of their range, cost and power consumption.

- **Cost**: Using a standard retailer the average cost of an HC-05 Bluetooth module comes in around $5.00, the average price of a LoRa module comes in at $7.00, and the average cost of a GPS module comes in at $25.00.
- **Range:** The range of LoRa technology is roughly 0-15Km, the range of Bluetooth technology is 0-50m while GPS range is 0-20,000km.
- **Power consumption:** Bluetooth and LoRa technologies consume less power than GPS. Lora has the added benefit of being to be in a sleep state until needed; this saves battery life.
- **Data Rates:** GPS comes in at 50bps, Bluetooth is 328kbps and LoRa is 27kbps. For this project, because we will be sending only small packets of data, high data rates are not a substantial factor in selecting a technology.

Table 2.1 summarizes the criteria for selecting a wireless technology for this project. Technologies which had a higher cost were given lower scores in the cost segment, and vice versa, technologies with longer ranges were given higher scores in the range segment, and vice versa, technologies with lower power consumption were given higher scores as well.

The numbers in table 2.1 represent the score out of 5 of each technology based of the selection criteria.

Table 2.1: Comparison of different communication technologies

Selection Criteria	LoRa	Bluetooth	GPS
Range	3	1	5
Size	2	2	2
Price	4	4	3
Power consumption	4	4	2
Total	13	11	12

From, comparing the above wireless technologies it is evident that the best device here taking into consideration each criterion is the LoRa technology.

2.4 LoRa

For this project, a Long Range (LoRa) wireless communication technology is used to send and receive the data packets which will be used for location determination based on the algorithm selected.

The LoRa module is the proposed wireless communication technology for this project. Lora (short for long range) is a spread spectrum modulation technique derived from chirp spread spectrum technology. Lora is a low powered wireless platform which has become the de facto for the internet of things networks. This is because LoRa offers the following in terms of advantages:

- It has a long-range of up to about 15km which can be used for penetration of dense urban environments due to sub 1GHZ frequency used.
- It is low powered and thus can be used for long without needing power supply changes, further increasing efficiency.

- It enables GPS free tracking applications, which is explored in this project.
- It has a high capacity and supports many message transmissions.

2.5 Tracking

Today the standard of tracking different systems is using GPS modules which rely on the use of at least three satellites for location determination; this has proven a very effective way of determining the location of devices such as the location of buses. In the figure 2.1 below, we see that a GPS receiver is present in the vehicle; the GPS receiver measures the distance from the satellite. It then uses the triangulation algorithm to compute its location. The information is then shared to mobile or web applications for the user to access the location more reasonably, for example; seeing buses as pins in google maps.

For LoRa, the infrastructure only needs to be built to enable multiple buses to be tracked by the infrastructure without the need of each vehicle having a cellular connection, unlike GPS solutions.

Figure 2.1 shows a GPS based bus tracking system which makes use of the internet through the GSM/GPRS network to share the information with the users.

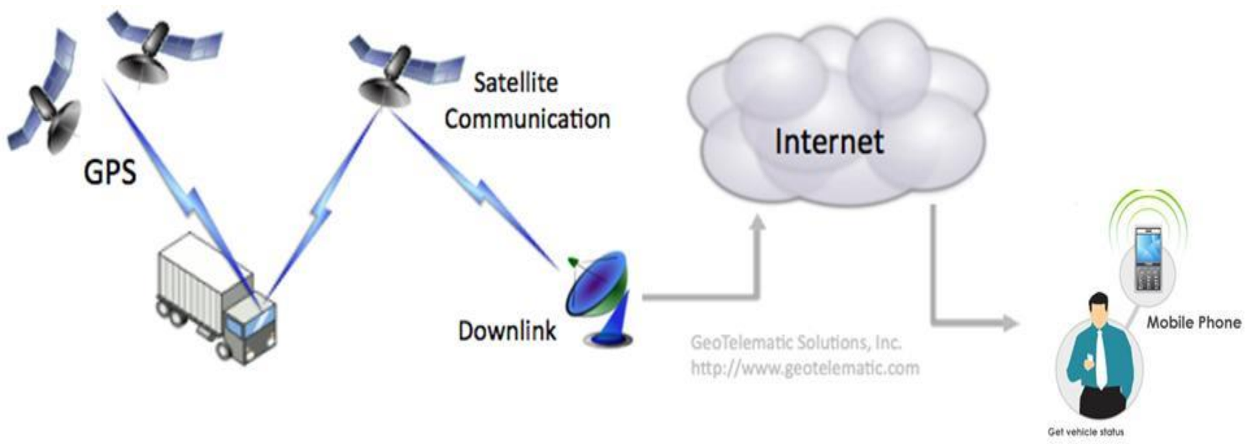

Figure 2.1: A GPS bus tracking system using internet through GSM/GPRS Network to share bus data with users.

2.6 Location Determination Algorithms

LoRa signals are electromagnetic waves as such they travel at the speed of light. The LoRa transmitter transmits packets at regular intervals which are then received by other LoRa receivers located at base stations. The distance that is covered by the LoRa signal can then be calculated by multiplying the time taken by the signal to arrive at the base station and the speed of light. R1, R2 and R3 illustrate the distance from the transmitting LoRa module to each of the different base stations in figure 2.2 these lines represent the radius of a circle. The intersection of these circles is used to determine the location of the bus containing the transmitting LoRa module.

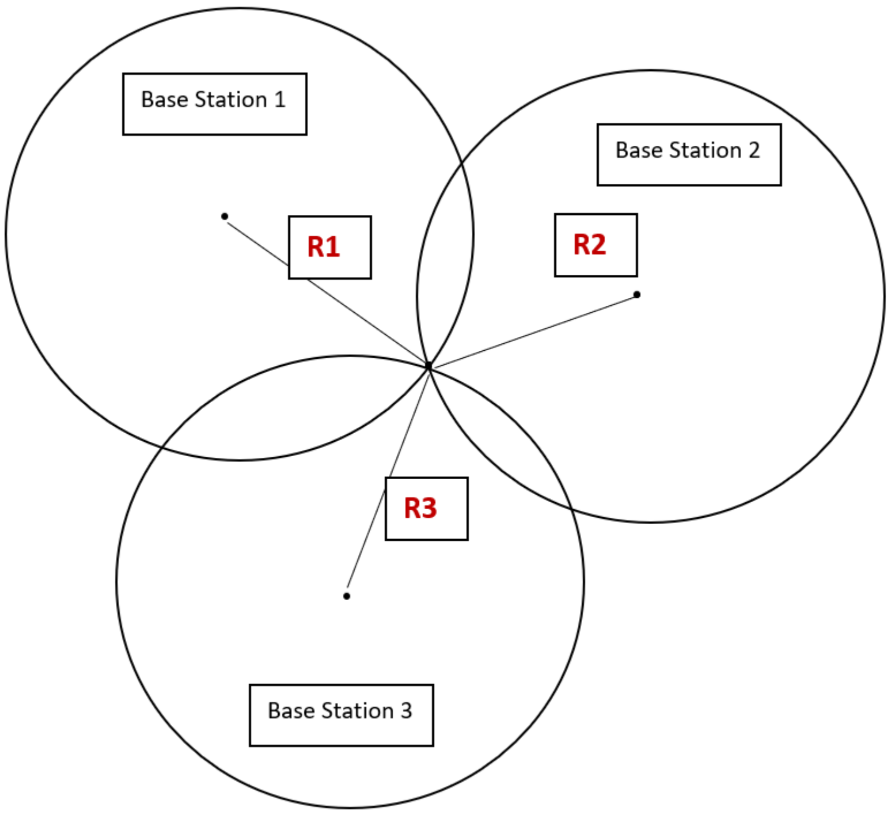

Figure 2.2 Geometry methods determining the location of Bus.

2.6.1 Base Stations

Base stations are points for communications usually located at high vantage points. These stations are made up of electronic components and some form of the gateway device, as illustrated in figure 2.3. For this project, the base stations need to be a maximum of 15Km away from the bus for LoRa receivers to receive the packets for location determination accurately. Each base station shall consist of a LoRa module connected to a microcontroller. Once the packets are received, the arrival time and signal strength are documented, and the name of a transmitting bus along with arrival time and sent time is transmitted to a server where the bus location can be calculated.

Figure 2.3 shows different end devices connected to gateways which then communicate with end applications through network servers

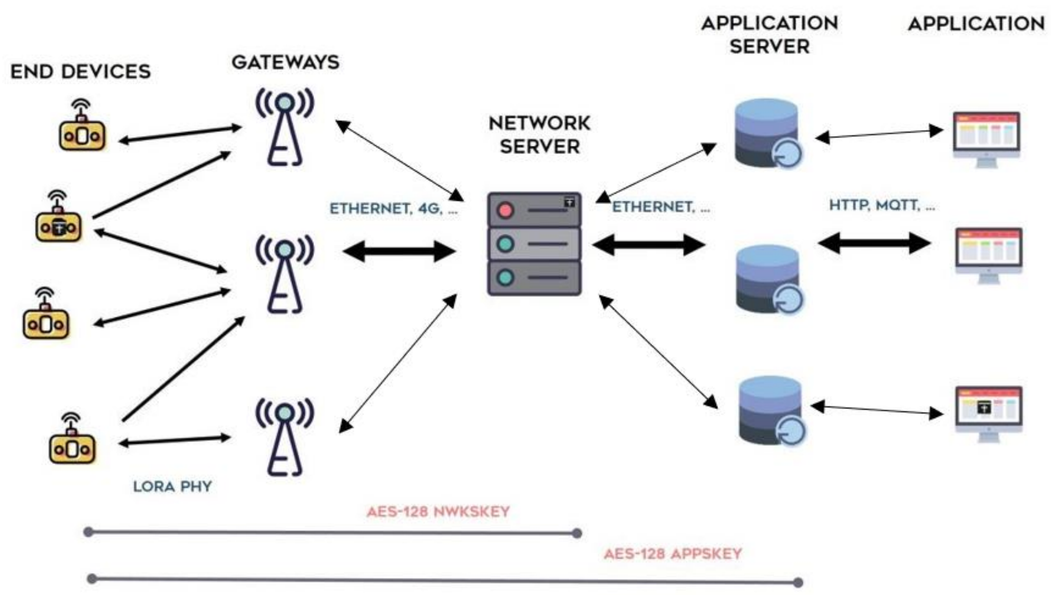

Figure 2.3 LoRa connected gateway system interfacing with end applications

2.6.2 Location Determination Algorithms Methods

For location determination using signals being transmitted between endpoints, two main techniques are used to determine the location of a vehicle being tracked.

- **Signal Strength** method involves the estimation of the location of a vehicle using the strength of the arriving packet at the receiver end. This Received Signal Strength Indication (RSSI) is a number representing the strength of the received signal. With this information and the propagation characteristics of the signal sent, it is possible to estimate the location of the device that sent the signal.
- **Time Measurement** – in this technique the time taken by a radio signal to travel from sending node to the receiver node is measured, along with this and the speed of light this method allows for the determination of the location of the sending node.

Time of Arrival

Time of arrival is a ubiquitous way to determine the distance between points, this method is based on knowing the exact time that a signal was sent from the target, the exact time the message arrives at a reference point, and the speed at which the signal travels [12]. Once these are known, the distance from the reference point can be calculated using the sample equation; $d = c * (tarrival - tsent)$ Where c is the speed of light. This distance calculated serves as the radius of a circle, as shown in figure 2.2 above. The circle has the equation of the form

$$d = \sqrt{(xref - x)^2 + (yref - y)^2}$$

Where (xref, yref) is the known position of the reference point

2.7 Geometry Methods

Multilateration, Triangulation and Trilateration are different geometry methods used to determine the location of a vehicle making use of either the radio strength or the Time measurements. Multilateration is the process of identifying locations of points by measurement of distances from known points. Multilateration uses the time difference of Arrival of signals; the receivers only recognize the exact arrival time of the transmission signal. Triangulation is the process of determining the location of a point by measuring only angles to it from known points at either end of a fixed baseline, rather than measuring distances to the point directly as in trilateration. In this project, trilateration which focuses on the time of Arrival or signal strength of packets from bus to the base stations, is used over other methods because of its simplicity.

2.7.1 Trilateration

Trilateration requires clock synchronization of the receiver and transmitter to be able to determine the time of arrival of the signal. In trilateration the transmitting device, in this case, the LoRa module regularly sends packets that may be picked up by three or more base stations made up of LoRa receivers enabling the trilateration algorithm to work. The distance travelled by a LoRa packet can be calculated by multiplying the time of arrival of the packet with the speed of light. We can then use these distances to create circles with the radius of each being the distance, as illustrated in figure 2.4. The intersection of these circles is later found out to be the location of the vehicle or object being tracked.

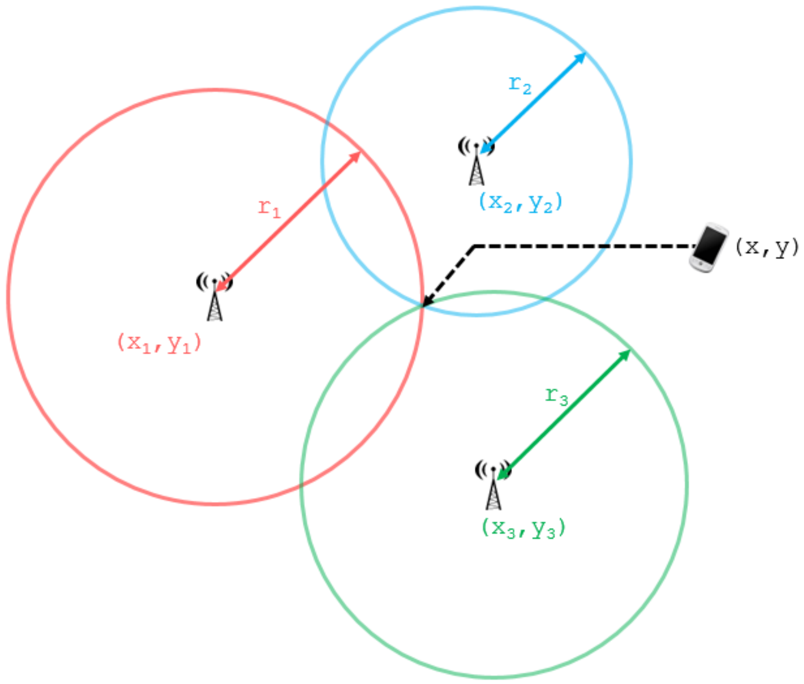

Figure 2.4: Example of location determination using geometry methods

- RSSI trilateration – Trilateration can also be applied based on the signal strength received. Signal strength has an inverse relation with distance travelled. Once the distance is found based on signal strength, we can use the algorithm described above. In later sections, we explore experiments to obtain an equation relating the signal strength to the distance travelled.

Chapter 3: System Design & Requirements

In this chapter the design considerations that went into the project as well as technical specifications shall be explored. The proposed system will be introduced and the paths through which users can interact with the system to achieve the said goal are enumerated. The system requirements will then be laid out after which the final design will be presented.

The goal of this project is to develop a system through which commuters will be able to track buses for trip planning. This system is meant for use in the public transport industry of Ghana. For this to be possible hardware and software solutions will have to be developed.

3.1 User and System Requirements

The primary users of this solution are passengers and road commuters alike. A passenger would access the system through a web application in which they can use the provided bus data and trip information to plan their journey. The use case below describes a typical scenario of a traveller improving their productivity based on results from the application.

3.1.1 Use Cases

Scenario 1

> Kwame Bediako is a national service personnel trying to make it to his workplace on time for a big presentation he has today. However, Kwame has personal errands to run before he can go to work, he then opens the application and enters the info of the bus he must pick. With this Kwame is provided with the current location of the bus. With this information Kwame can accurately plan his morning errands to make sure he is still in time to pick the bus without falling into rush hour.

3.1.2 User Requirements

The user should be able to:

- Access the web application on any internet enabled device
- Enter start point for trip details
- View all buses near a given start point
- View past locations of a queried bus

3.1.3 Non-functional Requirements

- The complete system should be cheaper than GPS solutions
- The system should be power efficient in order to avoid regular power supply replacements
- The transmitting station should be portable to allow for easy addition to buses
- Web application should be user-friendly i.e. easy to use and understand
- Calculation of bus locations should happen in near real time

3.2 System Architecture / Overview

The system is made of a software aspect as well as the hardware aspect. The hardware device is broken down into the part that goes into each bus as well as the hardware that makes up the base stations which are used by multiple bus systems transmitting bus data to these base stations simultaneously. The hardware in the bus will contain a LoRa module connected to a microcontroller. The hardware at the base station will have a similar setup with the addition of a GSM module to create a gateway where received bus data will be sent to the server for location calculations. Figure 3.1 shows the system topology.

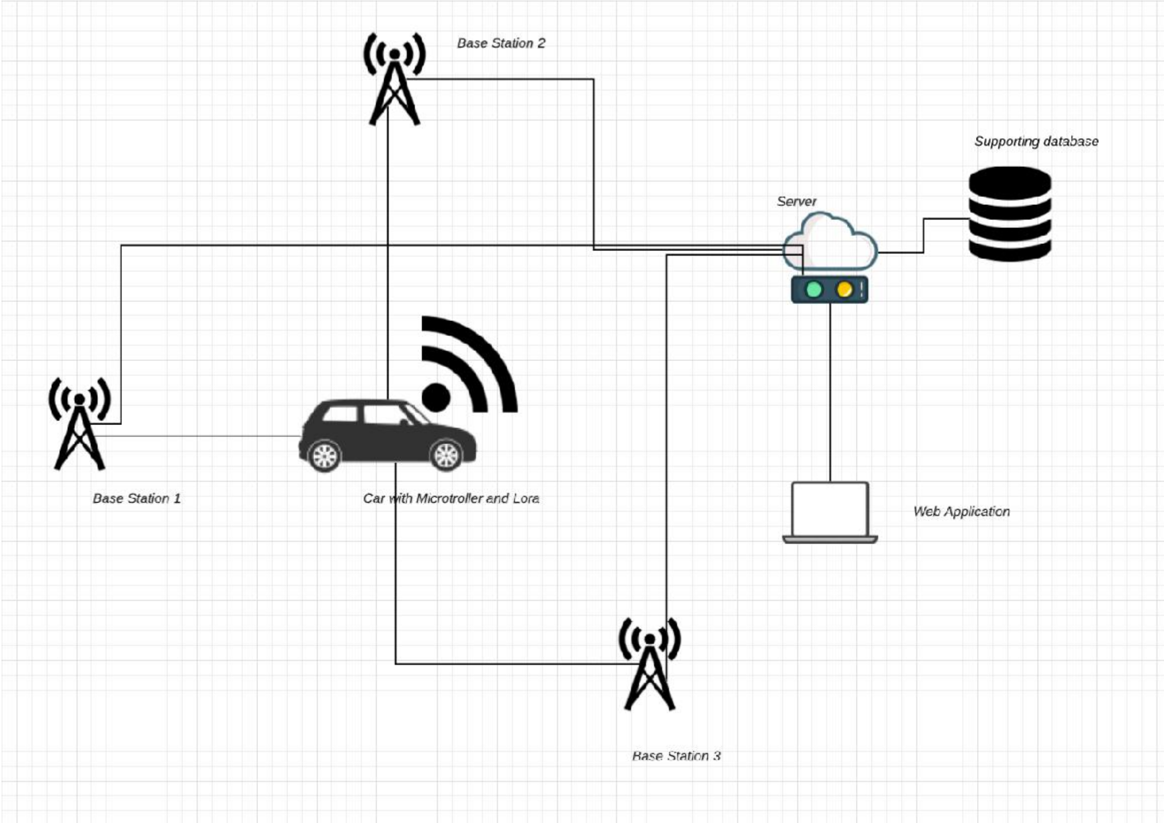

Figure 3.1: Topology of proposed System

For the project, we shall make use of specifically a Reyax LoRa module of frequency range 868/915 MHz which is based on the Semtech SX1276 Engine. The different base stations shall be set up with different LoRa transceivers which are all connected to the internet through a GSM module to send the data to the server where the computation shall take place in order to determine the exact location of the bus.

3.2.1 Microcontroller

To bring the system design to fruition, a microcontroller needs to be selected to interface with the LoRa module. There are many options when it comes to microcontrollers, namely; ATMEGA 328p, Raspberry pi and ARM-Cortex-M. Before choosing a microcontroller for this project, it is worth comparing the technologies in terms of their cost, compatibility, size and learning curve.

- **Cost**: Using a standard retailer the average price of an ATMEGA 328p is $6.00, the average cost of a Raspberry pi 3 is $50.00, and the average cost of an ARM-Cortex-M board is $15.00.

- **Compatibility:** For this project, the selected LoRa modules need to be able to interface with the microcontroller easily. The LoRa communicates through AT-commands, which is supported on both the Raspberry pi and the ATMEGA 328p.
- **Size:** The solution to be built should easily be able to fit an unobtrusive section of the bus. The ATMEGA 328p is the smallest with the raspberry pi being the largest.
- **Learning curve:** Something worth considering when selecting a microcontroller is its learning curve due to the fact, we desire easy repeatability of the project. ATMEGA 328p is the easiest microcontroller to pick up here with the Raspberry pi requiring more time to be familiar with it.

The table below summarizes the criteria for selecting a microcontroller for this project. Microcontrollers which had a higher cost were given lower scores in the cost segment and vice versa, microcontrollers of a smaller footprint were given higher scores in the size segment and vice versa, microcontrollers with a lower learning curve were given higher scores as well.

Table 3.1: Comparison of different microcontrollers

Selection Criteria	Raspberry pi	Arm-Cortex-M	Atmega 328p
Cost	2	3	4
Learning curve	2	3	4
Size	2	4	3
Compatibility	2	4	3
Total	**8**	**14**	**14**

From the table above, the Arm-Cortex or an Atmega 328p can be used as a microcontroller. However, for this project, we shall make use of an Atmega 328p for our microcontroller as this is a more straightforward system to interface with which allows for the use of different libraries to help in the development of the bus tracking system. The microcontroller is responsible for sending data to a cloud server to compute the bus location and expose this information to a user-facing web application through an Application Programmable Interface (API).

Each base station shall consist of a LoRa module connected to a microcontroller. Once the packets have been received the arrival time and signal strength is documented shown in figure 3.2. The packet data, along with arrival time, sent time and signal strength are transmitted to a server where the trilateration algorithm gets to work to locate the coordinates of the vehicle in question.

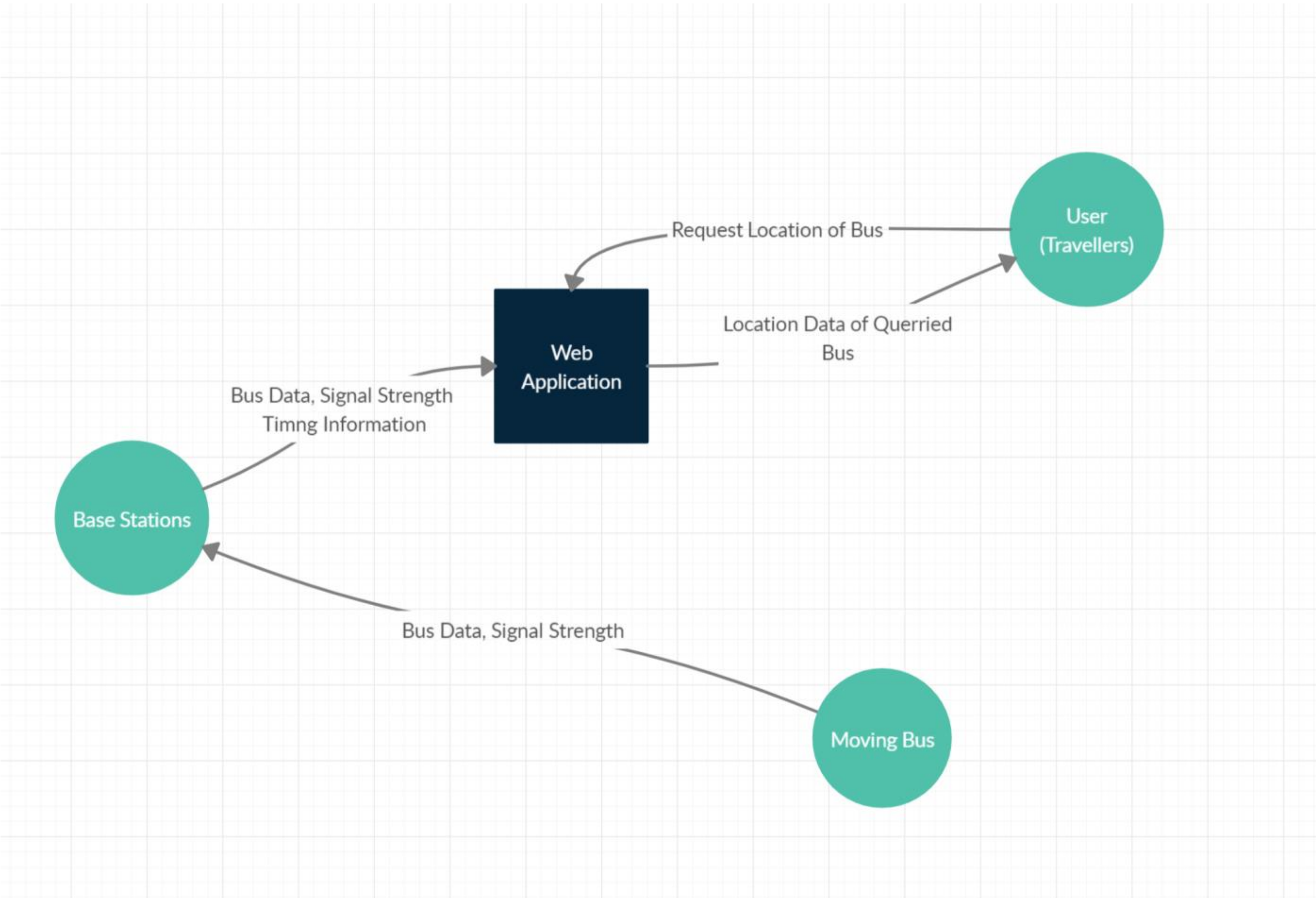

Figure 3.2 Different components of the project

3.2.2 User Interface

The user interface is a main part of the system, as this is the primary means by which users shall be able to interact with the system. The user will be able to search for the location of a bus by entering it into the application. For specialised results about how far the bus is from the user the user can enter their location to get to aid in determining these extra details. Below is an activity diagram of the web application illustrating this.

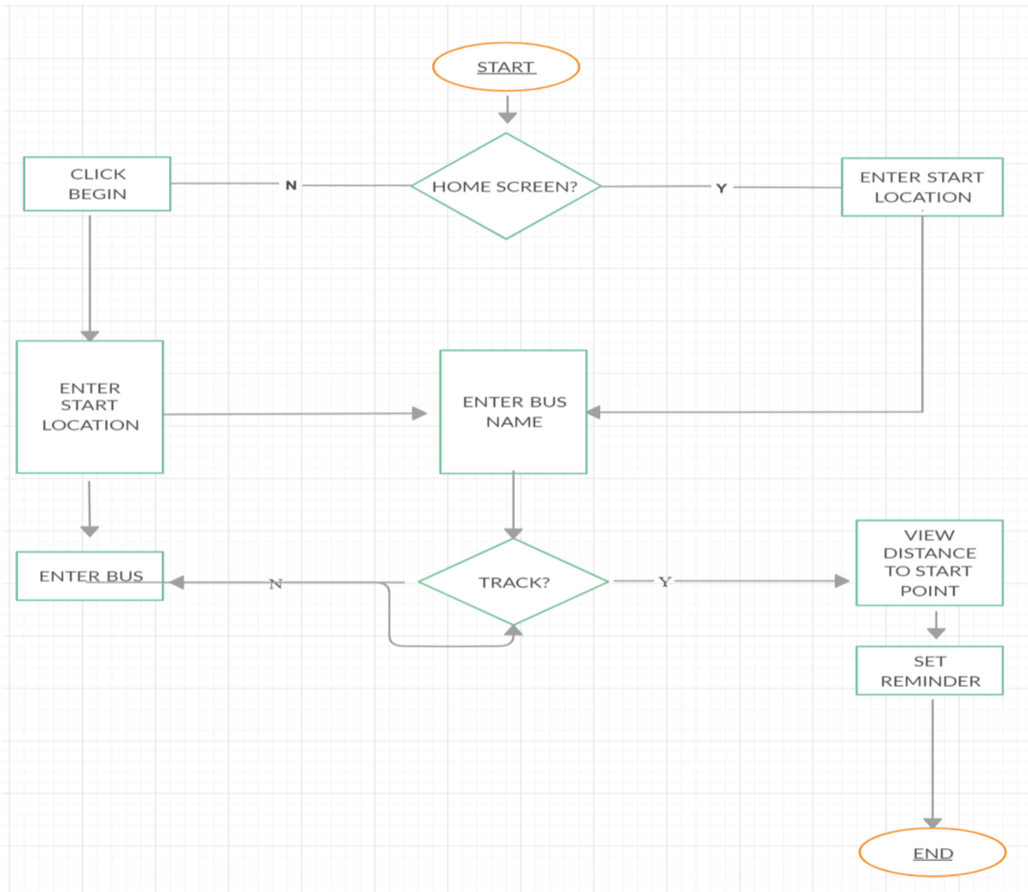

Figure 3.4: Web application activity diagram.

In the selection of these components certain design considerations were made. The table below shows a list of the components used as well as some technical specifications of each.

Table 3.2 – Components Specifications

Component	Type	Specification
Microcontroller	Atmega 328p	6 channels, 10-bit resolution ADC, 16Mhz internal oscillator, Flash memory, 1MHz CPU speed, watchdog timer, 23 programmable I/O lines, 8-bit AVR CPU
LoRa transceiver	Reyax LoRa module	**Range**: 0-15km. **Interfacing technique**: Communication over serial AT Commands
GSM Module	SIM800L EVB	Vin = 3.4v to 4.4v, 1A input current, Quadband 850/900/1800/1900MHz

Chapter 4: Methodology

4.1 Overview

This section walks through the steps taken to put together a working sample of this project. The section describes the different phases that went into putting together the project.

4.2 Hardware Implementation

The hardware for this project consists of two main parts, namely the hardware that forms the transmit node to be placed within the moving vehicles and the hardware which forms part of the receive node to be set up within each base station.

4.2.1 Hardware on Bus

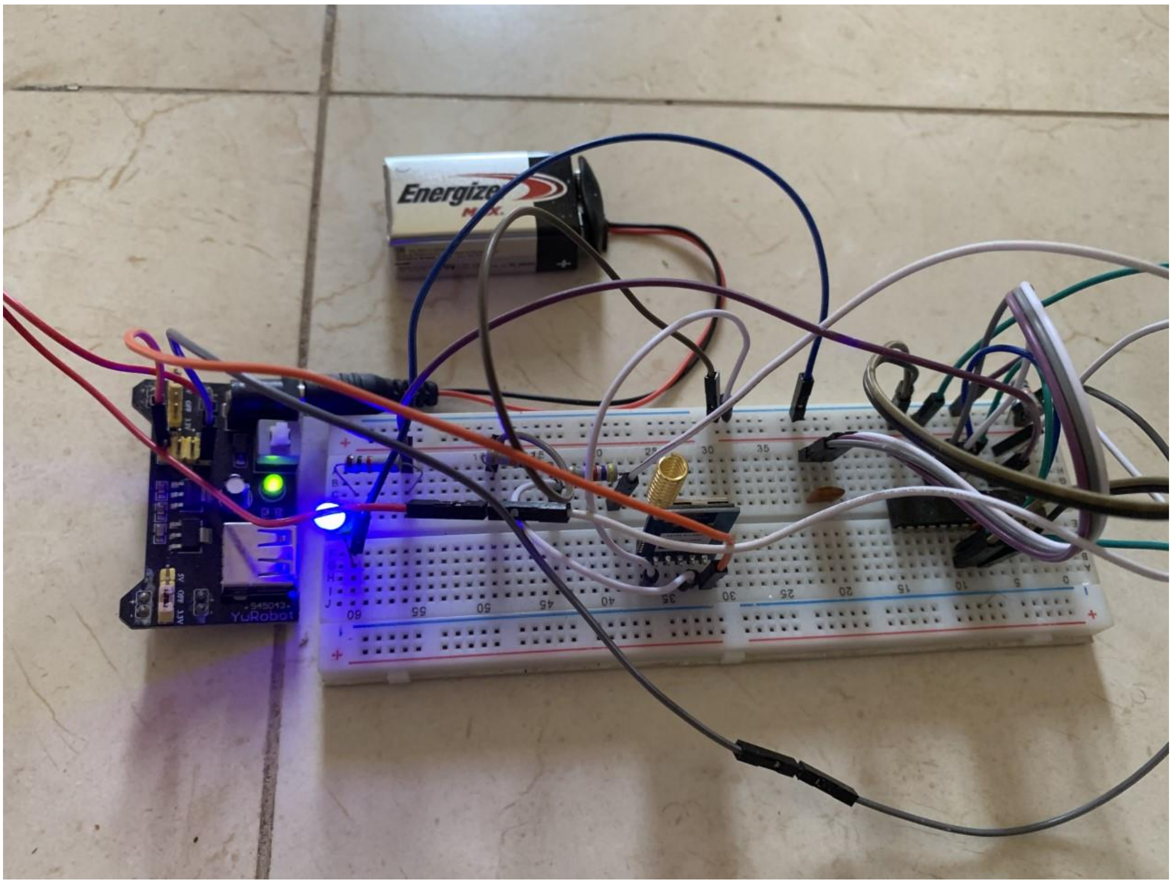

Figure 4.1 Bus hardware connection

The device in figure 4.1 consists of an ATMEGA 328p microcontroller which would interface with other hardware specifically the LoRa module to send bus data like name of bus and sent time to the hardware setup at the base stations. The LoRa transceiver acting as a transmitter in this case is set up to send these messages through AT commands. To be able to power these devices there needs to be a power source, this is provided by a single 9V battery connected to the ATMEGA 328p through a breadboard power supply which is then able to supply power to the other devices such as the LoRa module.

4.2.2 Hardware at Base Station

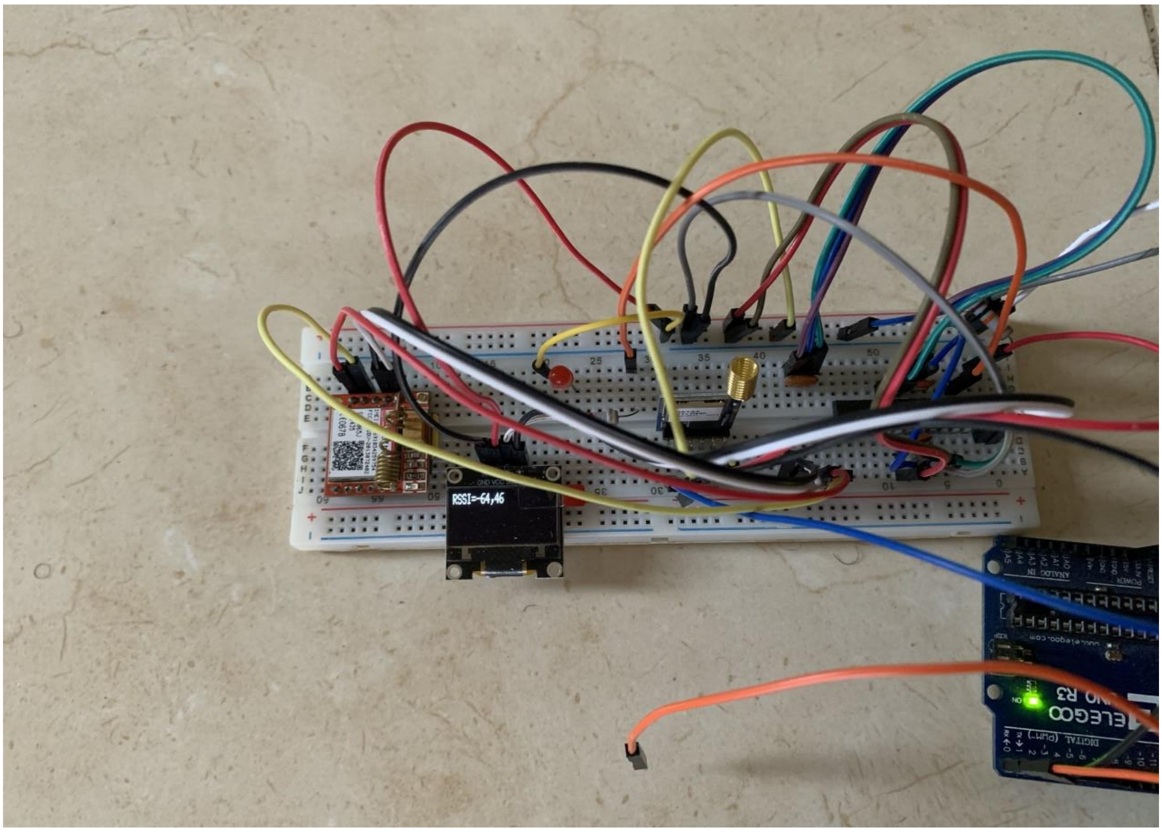

Figure 4.2: Base Station hardware connection

This part of the system shown in figure 4.2 above is the LoRa gateway which is situated at base stations and responsible for receiving the message from the moving bus and then transmitting to the server for location calculating. In building the LoRa gateway, an ATMEGA 328p was used to connect the LoRa transceiver with the GSM module an OLED screen is added to the circuit to display the received signal strength (RSSI).

When data such as the bus name is received from the transmitting node, the LoRa module attaches the signal strength (RSSI) to the received string before sending this to the ATMEGA 328p through the transmit pin. The RSSI is then extracted from the received string through string manipulation and written to the connected OLED screen. Before this, the received string is sent to the server through the GSM module and then forwarded to the database where the data gets queried for use in location calculation once the user tries to request the location of a bus.

4.3 Back-end Server Implementation

Once we have both transmit and receive components of the system set up, the received bus information needs to be uploaded onto the cloud where it can be saved in the database for easy querying whenever a user searches for a car.

For this to be possible, a messaging technology – Twilio is integrated, as shown in figure 4.3 below. Once the signal strength and Time of arrival information are received at the base station, this information is sent in a string form to an international Twilio number which then uses webhooks to call the application backend exposed on the internet through *ngrok. Ngrok* is a multiplatform tunneling, reverse proxy software that establishes secure tunnels from a public endpoint such as the internet to a locally running network service. Once the Twilio webhook calls the application API, the data is read and formatted before saving in the database.

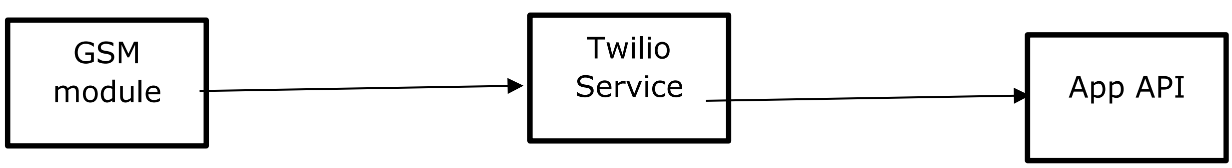

Figure 4.3 Flow of car data through System.

After the bus name, along with the RSSI and time of arrival been saved in the database, we can now discuss how the chosen trilateration algorithm was implemented in the App API.

The trilateration algorithm by standard convention works based on the cartesian plane. However, this algorithm is to be applied on the Earth's surface, which is roughly a sphere. Because of this, work was done in translating the latitude and longitudinal positions of the base stations into standard cartesian coordinates. The following equations were used to do this conversion.

$$x = R * \cos(lat) * \cos(lon)$$

$$y = R * \cos(lat) * \sin(lon)$$

$$z = R * \sin(lat)$$

Where *R* is the radius of the earth, with these coordinates and the distance travelled by the transmitting message, i.e. the radius of the circle around a base station, we apply the trilateration algorithm in sections by finding the intersecting region of two circles before overlaying that with the third circle.

After the intersecting coordinates have been found, it is converted back into longitude and latitude positions and returned to the user of the application for viewing.

The following equations were used to complete this back conversion.

$$lat = \text{asin}\left(\frac{z}{R}\right)$$

$$lon = atan2(y, x)$$

Below is the pseudocode implantation of the algorithm.

//ALGORITHM – LocationDetermination(B1 , B2 , B3 ,D)

//Input: B1 is a tuple of the coordinates of base station 1, B2 is a tuple of the coordinates of base station 2, B3 is a tuple of the coordinates of base station 3, D is a list of the distance of the moving vehicle from each of the base stations.

//Output: The algorithm returns the longitude and latitude coordinates of the bus

Convert x, y coordinates of the base stations to cartesian coordinates

Set up two corresponding circle equations as described in earlier sections

Find the intersecting points

Draw a line between these points

Check if any point of the final coordinates falls on the line

If true convert these points back to longitude and latitude, return the points as output.

A detailed view of how the algorithm was implement is attached in Appendix A.

4.4 Database Implementation

CarData

- BASE STATION ID
- CARNAME
- RSSI
- TDOA

Figure 4.4 Database Schema

The database implementation was handled, making use of mongo dB due to its flexibility and easy integration with the back-end language. The mongo dB database is hosted online through mongo labs allowing for constant access to our dB. The database is used to save vehicle information such as name, RSSI and TDOA, which are used in computing bus location.

The BASE STATION ID field is an autoincrementing integer field which is needed to keep track of which bus is being queried. The CARNAME field is a string field which is used by travelers to search for a bus to track. The RSSI field is an integer field used in the RSSI approach for estimating the distance a packet has travelled for use in the algorithm. The TDOA field is an integer field used in the TDOA approach for estimating the distance a packet has travelled for use in the algorithm.

4.5 Web Application Implementation

To build a user-friendly application, simple technologies were used to develop the web application. Some of the tools that were made use of were Hypertext Mark-up Language (HTML), Cascading Style Script (CSS) and JavaScript which are languages used in web development. The Google maps API was also used in order to add some visualization to the application.

Chapter 5: Testing & Results

In this chapter tests are carried on the application to ensure all goals of the project are achieved. A comparison is made between the RSSI approach and the TDOA approach to determine which gives best results.

5.1 Test Description

For the system to work as desired, the user application, as well as the hardware devices on the bus as well as base stations, need to be working correctly. The transmitter should send the car name to the different base stations, which then log the signal strength (RSSI) as well as the time difference of arrival. After which this is sent to the backend API before being saved in the database awaiting a user query before the location of the moving vehicle is determined. Thus, this section of the paper tested that the system design laid out in earlier chapters are valid.

5.2 System Modelling

Before we can test that the application works correctly, experiments are carried out with the transmit node, i.e. the bus hardware and the receiving node, i.e. the base station hardware. This is done in to derive a standard relation between the received signal strength (RSSI), and distance travelled. For the time difference of arrival approach, the equation is used to estimate the mileage travelled.

$$d = c * (tarrival - tsent)$$

The experiment was carried out as follows:

- Set up both the bus hardware and base station hardware as described in earlier sections
- Place the transmit node at a vantage point with a clear line of sight to the receive node.

- Using a tape measure vary the distance of the receiving node to the transmit node the while taking note of the RSSI displayed on the OLED screen and the distance from the transmitting node.

Table 5.1 shows the results of the experiment carried out on the flat level ground with a clear line of sight between the receiving node and the transmit node.

Table 5.1 – Signal Strength vs Distance

Distance (cm)	Signal Strength (RSSI)
0	-46
150	-60
300	-67
450	-76
600	-79
750	-82
900	-83
1050	-88
1200	-84
1350	-86
1500	-86
1650	-88
1800	-91
1950	-89
2100	-92
2250	-93
2400	-94
2550	-91
2700	-97
2850	-99
3000	-104

Given the RSSI (dBm) the relation to determine the distance is

$$\text{RSSI (dBm)} = -10n \log10(d) + A$$

Where A is the received signal strength in dBm at 1 meter - d is distance in metres and n is the propagation constant or path-loss exponent. [16]

The graph below shows the line best fit drawn through the scatter plot of the table above

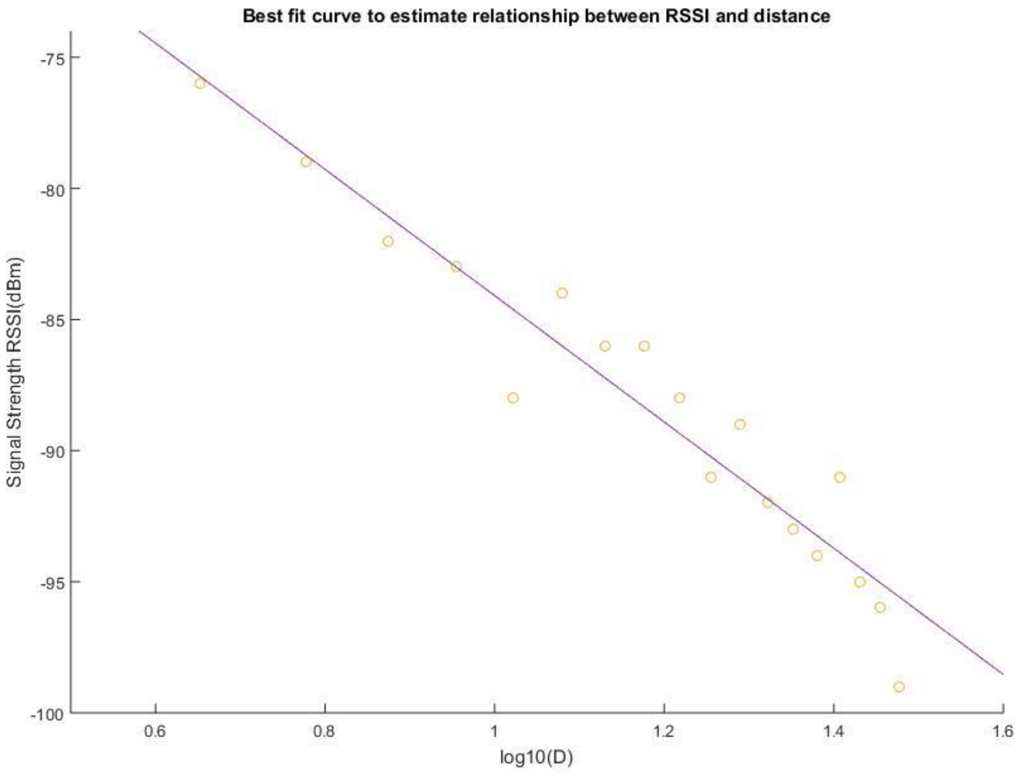

Figure 5.1: Best fit graph to derive relationship between RSSI and distance

From the graph above we can estimate *A = -46* and *n = 4.3* resulting in the relationship equation

$$d(m) = 10^{\left(\frac{RSSI(dBm)+46}{-43}\right)}$$

5.3 Application Testing

With the equation relating received signal strength to distance travelled and with Time difference of arrival technique, we can now test the user application to ensure the goals we set out to achieve are accomplished. For the experiment, Ashesi University campus was used as a testing ground. Three base stations were set up across the testing ground with the hardware set up described in chapter 3. The base stations were placed at *Queenstar hostel* (Blue pin), *Big ben cafeteria* (Green pin), and *Sports stadium* (Yellow pin) shown in figure 5.2. The transmitting station was then placed at a *Duffie hostel* and testing begun. Both approaches of time difference of arrival and using RSSI were used in the application to attempt to estimate the location of the transmit radio.

5.3.1 TDOA Approach Results

When the transmitted data is logged at the receiving base stations, the time difference of arrival is used to estimate the distance before the algorithm on the server computes the longitude and latitude coordinates of the transmitting station. For this to be feasible, a timing device or clock of nanoseconds precision is required. The distance travelled by the transmitted message is calculated by multiplying recorded time by the speed of light, hence, to record distance differences to the accuracy of at 10m the time measured needs to be in nanoseconds. This reasoning is illustrated below.

$$\frac{distance}{speed} = time$$

$$\frac{30m}{3 * 10^{8} m/s} = 1 * 10^{-7} seconds$$

From the above illustration, it is clear a clock of milliseconds precision will not provide enough accuracy, which is why a device to the degree of nanoseconds precision is needed as this will give enough overhead when trying to estimate much shorter distances.

However, the microcontroller used only provided microseconds precision. Thus, due to lack of such an effective timing mechanism for our tests, for the TDOA approach we test the accuracy of the developed algorithm on sample google maps data. Using google maps we measure the distance from the transmit location to each of the base stations shown in figure 5.2 above and save it in the database. When a user searches for *Duffie* in the application the database is queried for the distance of the transmit node to each base station, this information is the used by the algorithm implemented on the server to return longitude and latitude coordinates for the user to view the location of the transmit node representing the bus. A detailed implementation of this is attached in the appendix.

Results from the application is shown below.

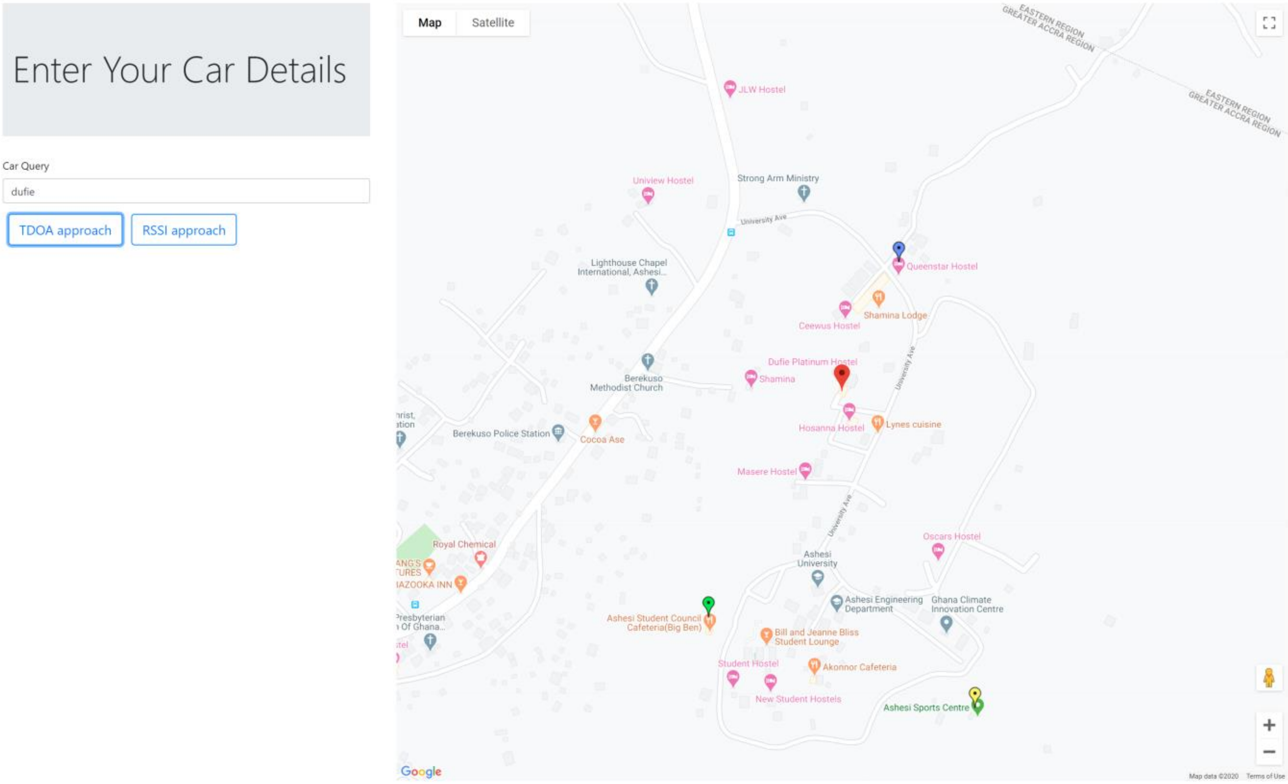

Figure 5.2: User Interface results using TDOA approach.

From figure 5.2 above the red marker represents the location of the transmit station representing a bus while the other marker colors represent the location of the three base stations.

5.3.2 RSSI Approach Results

In this approach the signal strength at the receiving stations is logged and then sent to back-end server which then uses the equation relating RSSI to distance travelled to estimate the distance between these points before applying the developed trilateration algorithm of the server. Results from this is shown below. The red marker shows where the RSSI based algorithm evaluates the transmit radio to be.

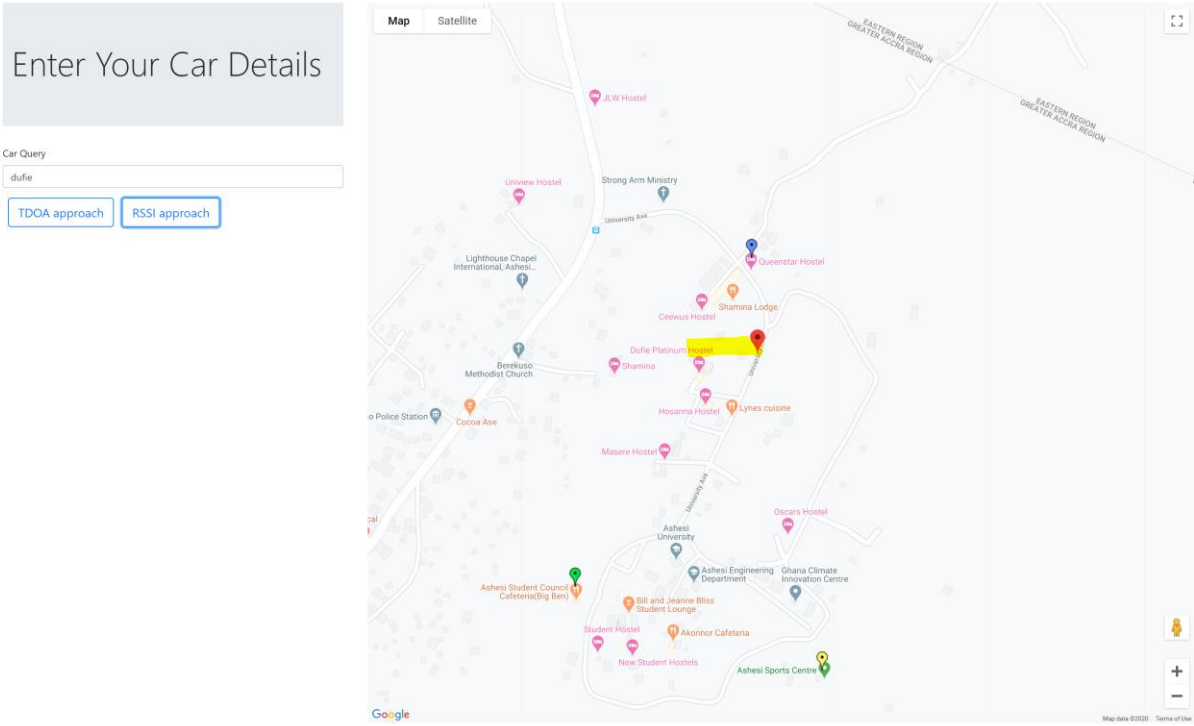

Figure 5.2: User Interface results using RSSI approach.

The section highlighted in yellow in the figure 5.2 shows the error margin from the result of the RSSI approach to what the right location should be. This result shows that the RSSI approach is off by roughly 40m in comparison to the TDOA approach. However, it is worth pointing out that the TDOA approach was performed on sample data to solely test the accuracy of the algorithm. With specialized timing gear the TDOA is likely to vary as well from the precise locations.

5.4 Statistical Analysis

Visual results are not enough to quantify the work done and to determine if the results are accurate or not. Thus, in this subsection an independent t-Test: Two-Sample Assuming Unequal Variances is carried to compare the results from the TDOA approach as well as the RSSI approach to the accurate geolocation from Google maps. From the results of the t-test we will then be able to confidently make a claim as to which approach produces best results. The user application in figures 5.2 and 5.3 returns longitude and latitude coordinates but this format is not suitable for the test thus in order to perform the test the following adjustments were made:

- *Queenstar*, one of the locations of the base stations was selected as a reference point
- The distance from locations across the Ashesi University campus to *Queenstar* is measured in meters using the Google maps service (Google Maps Data) in table 5.2.
- Each of these locations are queried in the user application and the distance from *Queenstar* to the calculated coordinates from the application is recorded. This done for both TDOA approach and the RSSI approach.

A t-test test was used as it enables us to determine if two independent datasets have a difference in their means. This result is captured in the two-tail test.

The table below shows the results of the adjustments described above. This table is used as the dataset for the t-test.

Table 5.2 Datasets for t-test

Location	Google Maps Data	TDOA Data	Google Maps Data	RSSI Data
Dufie	240.8	239.2	240.8	193.3
Columbiana	378.9	310.1	378.9	201.4
Foundation Dlab	605.8	496.4	605.8	241.1
Oscar	532.3	521.4	532.3	216.3
Shamina	341.6	333.7	341.6	158.6
Cewus	131.5	111.2	131.5	94.9

5.4.1 t-Test on TDOA

In a t-Test where we compared TDOA distances with the distances from Google maps measured from *Queenstar* to selected campus locations, table 5.3 shows the result of the t-Test carried out in excel with an alpha value of 0.05.

t-Test: Two-Sample Assuming Unequal Variances		
	Google Maps	*TDOA*
Mean	371.8166667	335.3333333
Variance	31276.75767	24136.00667
Observations	6	6
Hypothesized Mean Dif	0	
df	10	
t Stat	0.379634111	
P(T<=t) one-tail	0.356077333	
t Critical one-tail	1.812461123	
P(T<=t) two-tail	0.712154665	
t Critical two-tail	2.228138852	

Table 5.3: Result of t-Test on Google maps vs TDOA

From table 5.3 because we want to determine if the results from TDOA are the same as from google maps we are interested in the two-tail p value as this tells us if the hypothesized mean of equality is accurate. This p value was found to be 0.712154665, this value is significantly greater than our alpha value of 0.05, hence we cannot reject the null hypothesis but rather accept it.

5.4.2 t-Test on RSSI

In a t-Test where we compared RSSI distances with the distances from Google maps measured from *Queenstar* to selected campus locations, the image below shows the result of the t-Test carried out in excel with an alpha value of 0.05.

t-Test: Two-Sample Assuming Unequal Variances		
	Google Maps	*RSSI Approach*
Mean	371.8166667	184.2666667
Variance	31276.75767	2655.298667
Observations	6	6
Hypothesized Mean Dif	0	
df	6	
t Stat	2.493949206	
P(T<=t) one-tail	0.023455383	
t Critical one-tail	1.943180281	
P(T<=t) two-tail	0.046910766	
t Critical two-tail	2.446911851	

Table 5.4: Result of t-Test on Google maps vs RSSI

From table 5.4 because we want to determine if the results from RSSI are the same as from google maps we are interested in the two-tail p value as this tells us if the hypothesized mean of equality is accurate. This p value was found to be 0.046910766, this value is smaller than the alpha value of 0.05, hence we reject the null hypothesis and conclude these results are the not the same.

The implication of these test of the project is, when the TDOA approach is used in estimating bus locations it produces precise results compared to the RSSI approach which falls short.

Chapter 6: Conclusion

Using various technologies this project has been able to test and verify alternative tracking techniques. Here we will discuss some of the limitations encountered along the journey of the project as well as discuss potential future adjustments to the project before it can be fully integrated into the transport industry in Ghana.

6.1 Limitations

In this project a few challenges and limitations were faced some of them are:

1. The LoRa module used adjusts RSSI strength over time thus linear results are not produced over longer distances. This is because the LoRa module used here is able to monitor the loss in packets received over time thus if this loss is insignificant the RSSI is adjusted accordingly. The RSSI also caps out within the region of -100dBm thus affecting distance estimations
2. Due to the hilly nature of the test environment signal strength was affected thus producing inaccurate results as confirmed by the t-Test.
3. The lack of a more precise timing gear affected the ability to run tests on the TDOA approach, limiting the tests to the verification of the algorithm solely. This is because a timing gear with a precision in the nano second range is needed to be able to properly estimate distance, where as a timing gear of microsecond precision was instead available.
4. The operating voltage of the GSM module is roughly 3.4-4.5V, thus draining down the 9V battery faster. This prevented the running of the base stations for extended periods of time as the 9V battery would run down after a day of constant usage.

6.2 Future Work

This project does ground-up work regarding cost-effective location determination techniques for vehicle tracking to be used in the transport industry in Ghana. With this, there is still enough room for improvement to be made in the project before it is fully ready for deployment in the transport industry. Some future integrations to further enrich the project are as follows.:

1. The addition of a precise timing device within the nano second margin to aid in logging time of arrival of sent packets will enable the use of the TDOA approach ad described in earlier chapters.
2. Implementing a sleep mode at the base stations to prevent constant use of power from the GSM when not in use will help conserve battery power.
3. In selecting a LoRa module for the project a module which does not adjust RSSI should be considered.

For this project to work in the real world it will take a good amount of cooperation from public vehicle operators as they will need to keep the transmit nodes powered and on their vehicles. This is something work considering when the project is to be deployed.

References

[1] S. H. Sutar, R. Koul and R. Suryavanshi, "Integration of Smart Phone and IOT for development of smart public transportation system," 2016 International Conference on Internet of Things and Applications (IOTA), Pune, 2016, pp. 73-78, doi: 10.1109/IOTA.2016.7562698.

[2] P. A. Kamble and R. A. Vatti, "Bus tracking and monitoring using RFID," 2017 Fourth International Conference on Image Information Processing (ICIIP), Shimla, 2017, pp. 1-6, doi: 10.1109/ICIIP.2017.8313748.

[3] M. A. Abdel-Aty, R. Kitamura, and P. P. Jovanis, "Investigating the effect of advanced traveler information on commuter tendency to use transit," Journal of the Transportation Research Board, vol. 1550, no. 1, pp. 65-72, Jan. 1996.

[4] A. Azzez, I. A. Olateju, O. Ibrahim, and A. J. Remi, "Appraisal of the impact of team management on business performance: Study of metro mass transit limited, Ghana,"African Journal of Business Management, vol. 3, no. 9, pp. 390-395, Sept 2009

[5] Metromasstransit, Metromasstransit.com.gh, "Metro Mass Transit Ltd", 2017. [Online]. Available: http://www.metromasstransit.com.gh /about. [Accessed: 11 March 2019].

[6] R. Baah-Mintah and M. S. Adams, "Appraisal of public-private partnership in public transport service delivery: A case of Metro mass transit (MMT) limited, Ghana,"International Journal of Research in Social Sciences, vol. 2, no. 4, Nov. 2012.

[7] E. N. Aidoo, W. Agyemang, J. E. Monkah, and F. K. Afukaar, "Passenger's satisfaction with public transport services in Ghana: A case study of Kumasi-Accra route," Theoretical and Empirical Researches in Urban Management, vol. 8, no. 2, pp. 33-44, May 2013.

[8] M. Kumbhar, M. Survase, P. Mastud, and A. Salunke, “Real Time Web Based Bus Tracking System,” vol. 03, no. 02, p. 4.

[9] A. Al-Mazloum, E. Omer, and M. F. A. Abdullah, “GPS and SMS-Based Child Tracking System Using Smart Phone,” International Journal of Electronics and Communication Engineering, vol. 7, no. 2, p. 4, 2013.

[10] J. G. James and S. Nair, "Efficient, real-time tracking of public transport, using LoRaWAN and RF transceivers," TENCON 2017 - 2017 IEEE Region 10 Conference, Penang, 2017, pp. 2258-2261, doi: 10.1109/TENCON.2017.8228237.

[11] B. C. Fargas and M. N. Petersen, "GPS-free geolocation using LoRa in low-power WANs," 2017 Global Internet of Things Summit (GIoTS), Geneva, 2017, pp. 1-6.

[12] B. O’Keefe, “Finding Location with Time of Arrival and Time Difference of Arrival Techniques,” p. 3.

[13] K. M. McKay, “Integrated Automatic Vehicle Location Systems,” IEEE AES Systems Magazine, pp. 18-22, March 1997.

[14] Slavulj, Marko & Kanižaj, Krešimir & Đurđević, Siniša. (2016). The Evolution of Urban Transport - Uber.

[15] H. A. A. Dafallah, “Design and implementation of an accurate real-time GPS tracking system,” The Third International Conference on e-Technologies and Networks for Development, Beirut, Lebanon, April 2014.

[16] M. Ivanić and I. Mezei, "Distance Estimation Based on RSSI Improvements of Orientation Aware Nodes," 2018 Zooming Innovation in Consumer Technologies Conference (ZINC), Novi Sad, 2018, pp. 140-143, doi: 10.1109/ZINC.2018.8448660.

Appendix A

The code below is a python implantation of the trilateration algorithm used to compute the coordinates of vehicles in this project.

The function *calculateIntersection* finds the intersection between two circles and returns the coordinates of these intersecting circles.

The function *isInside* checks if any point on the third circle falls on a line drawn between the two points returned from the *calculateIntersection* function, and returns these points.

```
################# math for locating the vehicles
def calculateIntersection(x0 , y0 , r0 , x1 , y1 , r1):
    D = math.sqrt((x0 - x1) * (x0 - x1) + (y0 - y1) * (y0 - y1))
    if (((r0 + r1) >= D) and (D >= abs(r0 - r1))):
        a1 = D + r0 + r1
        a2 = D + r0- r1
        a3 = D - r0 + r1
        a4 = -D + r0 + r1
        area = math.sqrt(a1 * a2 * a3 * a4) / 4
        val1 = (x0 + x1) / 2 + (x1 - x0) * (r0 * r0 - r1 * r1) /
(2 * D * D)
        val2 = 2 * (y0 - y1) * area / (D * D)
        x1_1 = val1 + val2
        x2_2 = val1 - val2
        val1 = (y0 + y1) / 2 + (y1 - y0) * (r0 * r0 - r1 * r1) /
(2 * D * D)
        val2 = 2 * (x0 - x1) * area / (D * D)
        y1_1 = val1 - val2
        y2_2 = val1 + val2
        test = abs((x1_1 - x0) * (x1_1 - x0) + (y1_1 - y0) * (y1_1
- y0) - r0 * r0)
        if (test > 0.0000001):
```

```
            tmp = y1_1
            y1_1 = y2_2
            y2_2 = tmp
        return (x1_1 , y1_1 , x2_2 , y2_2)
def isInside(circ_x , circ_y , rad , x , y , z1 , z2 , z3):
    # Compare radius of circle
    # with distance of its center
    # from given point
    R = 6371000
    if ((x - circ_x) * (x - circ_x) +
        (y - circ_y) * (y - circ_y) <= rad * rad):
        z = (z1+z2+z3)/3.02
        xx = math.asin(math.degrees(z/R))
        yy = math.degrees(math.atan2((y) , (x)))
        return (xx , yy)
    else:
        return False
```