

**ADDIS ABABA UNIVERSITY**  
**ADDIS ABABA INSTITUTE OF TECHNOLOGY**  
**AFRICAN RAILWAY CENTER OF EXCELLENCE**



**MOBILE ENABLED PASSENGER INFORMATION SYSTEM**  
**FOR ADDIS ABABA LIGHT RAIL TRANSIT**

---

**A Thesis in Railway Engineering ( Traction and Train Control)**

By Rahel Abera

June 2019

Addis Ababa

**Advisor**

Mesfin Kifle (PhD)

A Thesis

Submitted in Partial Fulfillment of the Requirements for the Degree of Master of Science

The undersigned have examined the thesis entitled '**Mobile Enabled passenger information system for Addis Ababa light Rail Transit**' presented by **Rahel Abera**,

A candidate for the degree of **Master of Science** and hereby certify that it is worthy of acceptance.

_____	_____	_____
Advisor	Signature	Date
_____	_____	_____
Internal Examiner	Signature	Date
_____	_____	_____
External Examiner	Signature	Date
_____	_____	_____
Chair person	Signature	Date



## ABSTRACT

Passenger information system helps to ease the access to the public transport network for each person. Reliable real-time information should be available for passengers before and during travelling time. Providing most appropriate arrival time and train route location from the beginning to the end of their journey is necessary in order to enable passengers to plan door-to-door journey. For the case of Addis Ababa light Rail transit (AALRT) there is travel information for on board passengers, but the service lacks information for passengers who are outside the train, there is no real time information system at all. A Research has been done to solve AALRT information scarcity, information system that will be fixed at stations has been proposed before. But the solution will only be useful for passengers who are waiting at ending stations. Seeing this limitation, this thesis focuses on the passenger information system that will meet the real-time management needs, Real time location that consists on information about what really is happening, regarding the current service being executed. This passenger information system enables passengers to look at availability of train, while at home, in the office, in the hotel or while waiting in a train station, Passengers can see the location of a train on a chosen specific route, the expected time of arrival of the train at stations. Mobile enabled passenger information system that works in three delivery modes is designed; those delivery modes are android app, web and SMS (Short Message Service) provided within GSM network standard. The system needs hardware equipment's such as GPS with its receiver, server machine, GSM modems and Android Mobile phones to be implemented. This information system has been simulated successfully using simulation tools such as proteus software with Arduinio and Android studio for designing the Android app.

**Keywords: passenger information system, Android application, GPS**

## **ACKNOWLEDGMENTS**

First of all, I would like to thank my almighty God, for giving me strength through all the journey of my research works.

I would like to express my sincere gratitude to my research advisor, Dr.Mesfin Kifle for his dedication and invaluable guidance throughout this research. His dynamism, vision, sincerity, perfection, and motivation have deeply inspired me. He made me think the research in a wider way.

I thank my precious son for giving me a wonderful happiness that has added to my success in this work, Nafromi Tamene, to whom I gave birth recently while I am doing this research. I also thank my dearest husband, without him this all couldn't have come to be happened at all. Additionally I am grateful for opportunity given to me by DAAD for sponsoring and supporting me throughout my entire study and research works.

Finally I would like to give my sincere gratitude and many thanks to all my family, friends and all of people who have helped me during this period.

## TABLE OF CONTENT

<b>ABSTRACT.....</b>	<b>IV</b>
<b>ACKNOWLEDGMENTS.....</b>	<b>V</b>
<b>TABLE OF CONTENT .....</b>	<b>VI</b>
<b>LIST OF FIGURE.....</b>	<b>VIII</b>
<b>LIST OF TABLE.....</b>	<b>IX</b>
<b>ABBREVIATION.....</b>	<b>1</b>
<b>CHAPTER 1 INTRODUCTION.....</b>	<b>3</b>
1.1 Background .....	3
1.2 Statement of Problem.....	4
1.3 Objectives.....	5
1.3.1 General objectives .....	5
1.3.2 Specific objectives.....	5
1.4 Scope of the thesis.....	6
1.5 Methodology .....	7
1.6 Structure of the Thesis .....	8
<b>CHAPTER 2 LITERATURE REVIEW.....</b>	<b>9</b>
2.1 Customer Satisfaction factors for the light rail .....	9
2.2 Automatic Vehicle Location .....	10
2.3 Related passenger information work.....	11
2.3.1 Design and implementing of Passenger Information System at stations of AALRT.....	11
2.3.2 London Bus passenger information system.....	13
2.3.3 Estimation Time of Arrival (ETA) Prediction .....	14
2.3.4 Integration of Microcontroller with GPRS and GPS System to provide passenger information .....	15
<b>CHAPTER 3 SYSTEM DESCRIPTION.....</b>	<b>16</b>
3.1 PROPOSED SOLUTION .....	17
3.1.1 GPS.....	18

3.1.2 GPS Receiver .....	18
3.1.3 GSM Modem .....	19
3.1.4 Server Module .....	20
3.2 Real Time Location Information Delivery modes .....	25
3.2.1 SMS Based Train Tracking System .....	26
3.2.1.1 Major Components of the SMS based Tracking System .....	26
3.2.2 Web System .....	27
3.2.2.1 Major components of the system.....	28
3.2.3 Mobile Application .....	28
<b>CHAPTER 4 PIS SIMULATION .....</b>	<b>30</b>
4.1 Android Application Development .....	30
4.2. Tracking of location simulation by PROTEUS Software .....	37
4.3 Arduino Module .....	40
4.4 Server Design .....	42
4.5 SMS Delivery Development .....	45
4.6 Web Design Development .....	46
<b>CHAPTER 5 RESULT AND DISCUSSION.....</b>	<b>49</b>
5.1 Android Application.....	49
5.2 SMS.....	53
5.3 Web .....	54
<b>CHAPTER 6 CONCLUSION AND RECOMMENDATION .....</b>	<b>55</b>
6.1 Conclusion .....	55
6.2 Recommendation.....	56
<b>REFERENCES .....</b>	<b>57</b>
<b>APPENDIX SOURCE CODE .....</b>	<b>59</b>

## List of Figure

Figure 1: Activity Flow diagrams .....	7
Figure 2: AALRT system design architecture .....	12
Figure 3: London Bus information systems .....	13
Figure 4: RTPIS Scenario .....	14
Figure 5: AALRT System Architecture.....	17
Figure 6: Activity Life cycle .....	32
Figure 7:Android Studios 3.4 .....	34
Figure 8: Screenshot of developed AALRT Passenger information application user interface layout on Android studio .....	35
Figure 9: Screenshot of developed AALRT Passenger information application layer on Map using Android studio .....	36
Figure 10: Screenshot of AALRT stations on Map .....	36
Figure 11: screenshot of AALRT Passenger Information Android App outlook.....	37
Figure 12: Flow chart for Tracking System Simulation using Proteus Software .....	38
Figure 13: proteus simulation of AALRT train tracking system.....	39
Figure 14: Screenshot of arduino software used for proteus simulation .....	40
Figure 15: Screenshot of python.....	41
Figure 16: Screenshot of VSPE .....	42
Figure 17: Server Database.....	43
Figure 18: Flow chart for simulation of SMS based system .....	45
Figure 19: Screenshot of AALRT passenger information Wen page.....	47
Figure 20: Sublime Text windows; being used for AALRT web development.....	48
Figure 21: AALRT Application Registration layout .....	49
Figure 22: AALRT Application Log in layout .....	50
Figure 23:Screen capture of AALRT info app showing trains information.....	51
Figure 24: Saris station icon popup detailing on the next train .....	52
Figure 25: capture of SMS delivery mode process.....	53
Figure 26: AALRT web.....	54
Figure 27: capture of comment giving space for AALRT passenger on the web page....	54

## List of Table

<i>Table 1: Database storage of tracking system.....</i>	<i>42</i>
--	-----------

**ABBREVIATION**

<b>AALRT</b>	Addis Ababa light railway track
<b>ADT</b>	Android development tool
<b>CCTV</b>	Closed-circuit television
<b>DMS</b>	Dynamic message sign
<b>DVM</b>	Dalvik Virtual machine
<b>ETA</b>	Estimated time of arrival
<b>GNSS</b>	Global navigation satellite system
<b>GPRS</b>	General Package radio service
<b>GPS</b>	Global positioning system
<b>GSM</b>	Global system for mobile communication
<b>HTTP</b>	Hypertext transfer protocol
<b>HTML</b>	Hypertext markup language
<b>IDK</b>	Integrated development environment
<b>IP</b>	Internet protocol
<b>IVR</b>	Interactive voice response
<b>LAN</b>	Local area network
<b>LBS</b>	Location based service
<b>LCD</b>	Liquid crystal display
<b>LED</b>	Light emitting diode
<b>OS</b>	Operating system

*Mobile Enabled passenger information system: case of AALRT*

<b>PCMCIA</b>	Personal computer memory card international association
<b>PIS</b>	Passenger Information System
<b>POI</b>	place of interest
<b>RTPIS</b>	Real time passenger information system
<b>SDK</b>	Software development kit
<b>SIM</b>	Systematic Identification Module
<b>SMSC</b>	Short message service center
<b>SMS</b>	Short message service
<b>URL</b>	Uniform resource locator
<b>WAP</b>	Web Application Proxy
<b>Wi-Fi</b>	Wireless fidelity

## CHAPTER 1 INTRODUCTION

### 1.1 Background

Vehicle tracking system provides the passengers with the current location of the vehicle and gives the details of the estimated arrival time of the vehicle to the passenger. This system helps the passengers to make better travelling decision as passengers can decide if they should wait for the vehicle or not in accordance to the estimated arrival time. From this system passenger can also determine if the vehicle has stop operating or yet to come. The system provides the basic information which is necessary to make day to day life more efficient. There are many benefits to implementing a powerful passenger information system. The most important thing is that, passenger information systems are extremely versatile tools that can be used to deliver information through a wide variety of devices and platforms. Apart from integrated Light Emitting Diode (LED) signs and transit stop announcement boards, head signs, annunciators, and other on-board devices, these systems can also feed information to telephone Interactive voice response(IVR) services, Mobile, SMS text services, web enabled devices, and transfer stations signs. Mobile apps are changing the way we live and work in our constantly growing cities. Apps unlock the power of real-time by using the GPS coordinates of transport modes to show moving train (and other modes) to the waiting users.

AALRT is light rail transportation system is found in Addis Ababa, Ethiopia [1]. It has two lines going from East-West and North-South, the total length of both lines is 31.6 kilometers (19.6 mi), with 39 stations [2]. Trains are expected to be able to reach maximum speeds of 70 km/h. Of the two-line rail lines, the east-west line extends 17.4 kilometers (10.8 mi), stretching from Ayat Village to Torhailoch, and passing through Megenagna, Meskel Square, Legehar and Mexico Square The north-south line, which is 16.9 kilometers (10.5 mi) in length, passes through Menelik II Square, Merkato, Lideta, Legehar, Meskel Square, Gotera and Kaliti. However, two lines have a common track of about 2.7 km. The Light Rail Transit (LRT) has a total of 41 cars, each with the capacity to carry 286 passengers at a time, even though; some of the trains are not operating currently. This will enable the LRT to provide transportation to 15,000 people an hour, in one direction, and 60,000 in all four directions [1]. When we come to the information

provided about the service, there is no standard passenger information system at all. As stated above designing PIS for public transit is key factors for improved customer satisfactory service. Observing lack of PIS for AALRT service, I was motivated in order to contribute research that will bring solution to the problem for better communication. This thesis presents mobile enabled PIS that can be accessed (a) using android mobile phone (b) through web (c) using the Short Message Service (SMS) facility provided within the Global System for Mobile communication (GSM) standard. This system will enable passenger to know exact position of train while they are at home, office, or somewhere else away from station and also exactly standing at stations

## **1.2 Statement of Problem**

Passenger information is vital for developing a user-friendly public transportation system. AALRT passengers have suffered from poor information about services, particularly at times of disruption. Information such as train direction, time of arrival, destination is not provided for Passengers waiting at stations. Adding to this the railway sector is getting the lowest customer satisfaction [3]. A previous study by Akpoyomare et al[4] have shown that customer satisfaction was influenced by age, income, gender, perceived quality and perceived value but not by pre-purchasing expectation. The public transit usage can be improved if real time information of the vehicle such as, current location of the train and time taken to reach the destination are provided with easier access. It would also be helpful for the passengers to find alternate choices like of taxis and another mode of transportation depending on their circumstances. As excessive long waiting often discourages the travelers and makes them reluctant to take train. Previously, a thesis that was written by Natan jima [5] on AALRT passenger information provides system that display train arrival, destination and direction through Liquid crystal display (LCD) display fixed at a station.

But this system works only for users who are waiting at end stopping stations. To bring this communication barrier to an end, passenger information system that works through android mobile phones anywhere the user is located is introduced in this thesis. Mobile passenger information system is rapidly gaining popularity for public transport information dissemination particularly due to their anytime-anywhere availability. Mobile App that access webpage containing service status data provides information in a straightforward, easy-to-use manner and with added feature of more information about transit service that can't be displayed at station display. This thesis presents Mobile Enabled passenger information system for AALRT passengers bringing an individual travel information assistance application for Android mobile phone users .

### **1.3 Objectives**

#### **1.3.1 General objectives**

To design mobile enabled passenger information system for Addis Ababa Light Rail Transit that enables Android phone users access train real time location information using their mobile phone.

#### **1.3.2 Specific objectives**

- To Improve AALRT passenger information systems, by designing system which will be helpful enormously by giving users, more particularly potential user's a confidence in using train services.
- To provide away for passengers, a way to quickly access real-time transit information, or check the status of their train at any time, no matter their location.
- To design Real-time transit information system that can be accessed using web-enabled Android cell, offering greater flexibility for riders.
- To design system that contains most of the pertinent information, passengers will likely want to view. To design Web that will be used by foreigners and new passengers to get information relating to AALRT service.
- To provide communications link between AALRT transit agency and their riders.
- To enable the transit agency provide real-time train location, status updates, schedule data and timely announcements for passengers of AALRT.

- To vastly improve the transit experience for riders and streamlines work for agency employees.
- To reduce passengers' stress that comes out of waiting train that may take long time to arrive to the desired station or may not come at all.
- To Present SMS facility system that delivers real time location of AALRT train, which is convenient way of delivering limited amount of information, no need for dialing and negotiating a radio link data connection
- To enable passengers with all kind of mobile platform access train location information using real time train location SMS facility.

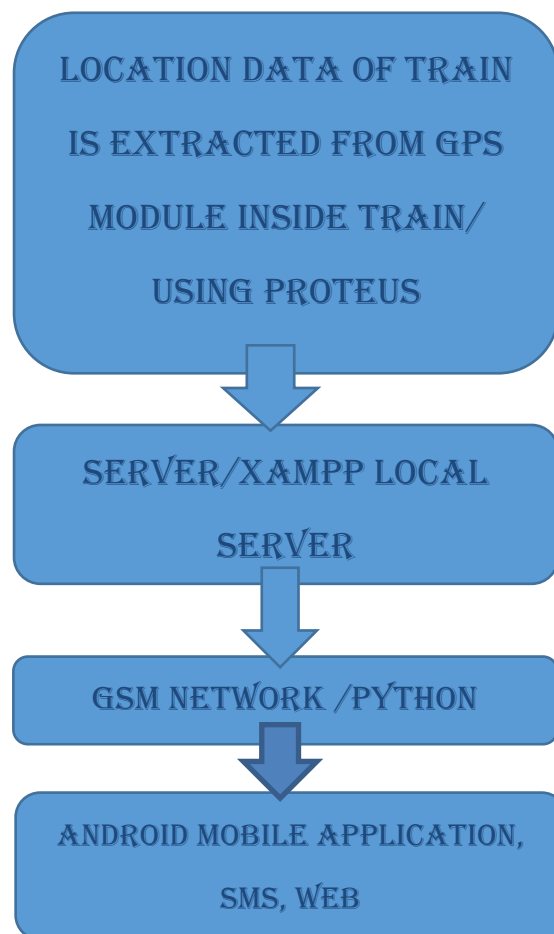
### **1.4 Scope of the thesis**

Design of PIS application that was developed was based on majority smartphone system operation users. For example, most of the smartphone used in Addis Ababa run on an Android operating system, Android is at its pick of success not only amongst the youth but entire community. Android Application that meets the user's need has become a key idea with the rapid growth and huge advances in Android. Android device has become a powerful device which provides much more than basic facilities. Because of all the reason described above Android applications were developed for PIS use. This means Application for iPhone, Windows and other kinds of smartphones were not included in this work. But the SMS and web system can be accessed by any kinds of platforms devices. SMS based tracking system which transfer train location using SMS to any mobile platforms and web that will provide service information of AALRT is designed

The other constraint in this work is that the proposed passenger information system does not operate in local languages this means users have to know English to understand the information given on the system.

## 1.5 Methodology

Development methods of this thesis can be categorized in to three main different stages one is observing AALRT platform distribution, the platform distribution is analysed through photo surveying, AALRT service operation and necessary documents for the thesis was collected from Addis Ababa light rail transit office. Secondly, different Literature relating to PIS is reviewed. Finally Passenger information system modules are briefly described and system is designed successfully using simulating tools. Android application that shows current location of trains is designed using android studio with java programming language. Web using local XAMPP server that contains AALRT service information is designed. The major steps used to simulate the real time location data of AALRT train is described in the following activity flow diagram.



*Figure 1: Activity Flow diagrams*

## **1.6 Structure of the Thesis**

This section describes how the thesis is organized. The contents of the following chapters are briefly reviewed as follows.

- **Chapter Two** reviews previously related work done in solving similar issue like of this, using GPS technology, GSM technology and different real time tracking system and also cover different literatures like of articles, researches and vehicle tracking system that is already implemented in another country.
- **Chapter Three** discusses about hardware used to develop the system and working methodology of the three delivery modes of the system.
- **Chapter Four** explains the Simulation process of passenger information system. Three different stage works for the whole project simulating, like hardware interface simulation, database management using local server on computer and android based mobile application and web and SMS system developed.
- **Chapter five** explains about result and discussion of this thesis work. Steps that has to be taken in order to operate this passenger information system is presented.
- **Chapter six** presents the conclusion and recommendation of the final work.

## **CHAPTER 2 Literature Review**

A passenger information system is the key communications link between a transit agency and their riders. Passenger information technology lets a transit agency communicate with its passengers to provide them with real-time train location and status updates, schedule data and timely announcements. It vastly improves the transit experience for riders and streamlines work for agency employees. AALRT communication system consists of digital transmission net, telephone, Closed-circuit television (CCTV), wireless communication and the relevant affiliated facilities. AALRT has on board passenger information system which informs passengers with the current place, next station and safety measurement that should have to be taken inside train. Coming to the outside of train information, there is no standard system that informs train status. This makes the service to be less reliable and low customers satisfactory. In this chapter starting from factors and analysis taken for passenger satisfaction, literatures gathered from articles, researches and previously done passenger information system in different countries is discussed below.

### **2.1 Customer Satisfaction factors for the light rail**

Customer satisfaction is key factors for public transport, in order to --fulfill those factors, the most important aspects that influence customer satisfaction when using a mode of transport are listed as follows, reliability, price, journey time, transport connections, cleanliness of train and stations, Accessibility, customer-focus staff, ticketing service, information provision, safety and security [6]. The term ‘reliability’ within a transport context relates to an uncertainty in the time taken to travel from the start to the end of a person’s journey [7]. Based on previously done research determinants of customer satisfaction are found to be refreshments, safety and security, basic facilities, information system, and behavioral aspects of service quality [8].

## **2.2 Automatic Vehicle Location**

In a paper [9] Dailey describes how a statistical model can be created to model the motion of buses along their routes. His model assumes that the position data for the buses are expressed in terms of a linear distance from the start of the route. This approach works well with a sign-post-based Automatic Vehicle Location (AVL) system [10], where the electronics in the street sense, perhaps via radio tags on the vehicles, when the vehicles pass a certain point. The computer that receives the AVL data can easily map a sign-post to a definite linear distance by looking up that information in a table of known distances. Some AVL systems, sense their location using Global Positioning System (GPS). GPS units in the vehicles receive a radio signal from GPS satellites in orbit around the Earth and use physics and trigonometry to find their exact three-dimensional position on the surface of the Earth, which is expressed as latitude, longitude, and latitude. Brendan Kidwell [10] developed different kind of algorithm that involves partitioning a route into zones. The algorithm tracks a bus' entrance into and exit from a zone, and how much time it spent in the zone. The transit time through a zone is stored in a table, and it can later be used to predict how long it will take an individual bus to get from where it is to a particular stop. There are GPS units in fixed-route buses out on the road, and at regular intervals of at most a couple of minutes, the following information is transmitted to an operations center and relayed to the algorithm in real-time, possibly along with other data:

- Bus ID unique
- ID of this vehicle
- Date Time date
- Time of the record

Route Name which route is this bus running right now

Latitude North-South position coordinate on the Earth's surface

Longitude West-East position coordinate the buses follow simple routes from a starting point, to an ending point, and back [10].

## **2.3 Related passenger information work**

Research Paper done by group of Indian students' presents Real time passenger information system (RTPIS) which provides travel information to passengers and tourists enabling them to make informed decisions about modes, routes and departure times. The RTPIS framework can be broadly divided into two contexts: Pre-trip context and On-trip context. The former provides information like timings, fares and routes well before the commencement of travel, through the Internet or the Short Messaging Service (SMS). The On-trip context provides information like location and places of interest (POI) while on the move. This is achieved using on-board and at-stop terminals (displays and audio announcement units) [11]. Real time passenger information system uses variety of technologies to track the location of bus in real time and generate the prediction of bus arrival at stops along the routes [12]. V. Yamuna, G. Rupavani, et al. [13] proposed Global navigation satellite system (GNSS) based bus monitoring system. The main objective of this system is to reduce the waiting time of passenger in bus stop by sending information about the location of buses to the passenger through SMS. GNSS based web application is developed which provide real time location of bus on Google Maps along with speed.

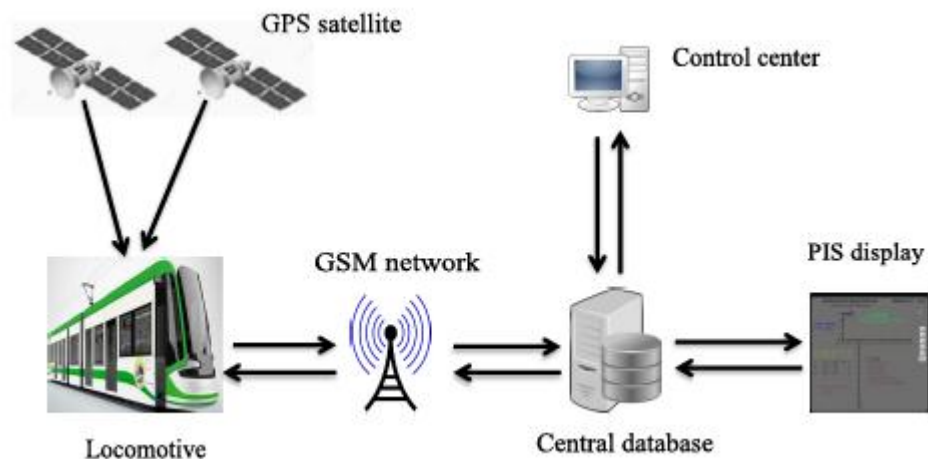
### **2.3.1 Design and implementing of Passenger Information System at stations of AALRT**

A thesis done by Addis Ababa University student Natan jima[5] regarding passenger information system for AALRT is summarized below. The author proposes new passengers information system that is designed to be implemented at AALRT station. System works by Global system for mobile telecommunication module for data transmission between central command and moving train.

## *Mobile Enabled passenger information system: case of AALRT*

A passenger information system (PIS), fixed at station provide real-time passenger information it received from the central server to a passenger at stations. The PIS uses GPS technology to collect the real time information all about the train. Once vehicle location, direction, speed and others are determined from the GPS components, it transmit this information to the server (central monitoring and control) and passenger information module at station through GSM network. On-board module deliver about present station, next bus stop, speed, and time and predetermined schedule, safety information, and operation. Figure 2 below shows previously proposed PIS architecture designed for AALRT.

With fast growing rate of technology many systems are being created to make people life's easy. As transit service continues to evolve, passenger information systems are quickly becoming a mainstay in today's public transit domain. Integrated systems that keep passengers informed along their journey are increasingly in demand, and transit agencies must find a way to keep up with this need. Finally, in this thesis work, passenger information system that enables Android phone users has been done, users of AALRT can access the information without the need to go to the stations anywhere they are.



*Figure 2: AALRT system design architecture[15]*

### **2.3.2 London Bus passenger information system**

Western Europe and Japan have adopted real time passenger information systems for over a decade now. These systems are seen in many different forms which ranges from traditional static displays to those wireless intelligent information systems providing pretty accurate arrival times. Dynamic Message Sign (DMS) [14] is a popular way to display transit information and are commonly known as electronic sign. Figure 3 [14] shows LED sign used in London to display real time arrival information of various buses at a particular bus stop



*Figure 3: London Bus information systems [14]*

### 2.3.3 Estimation Time of Arrival (ETA) Prediction

On reference [15] ETA algorithm that has two parts has been present:

1. Link updater, which estimates the travel time for each link
2. ETA calculator, which calculates the ETA for every bus stop.

The following architecture figure 3 shows the whole system design of Real Time Passenger Information System (RTPIS).

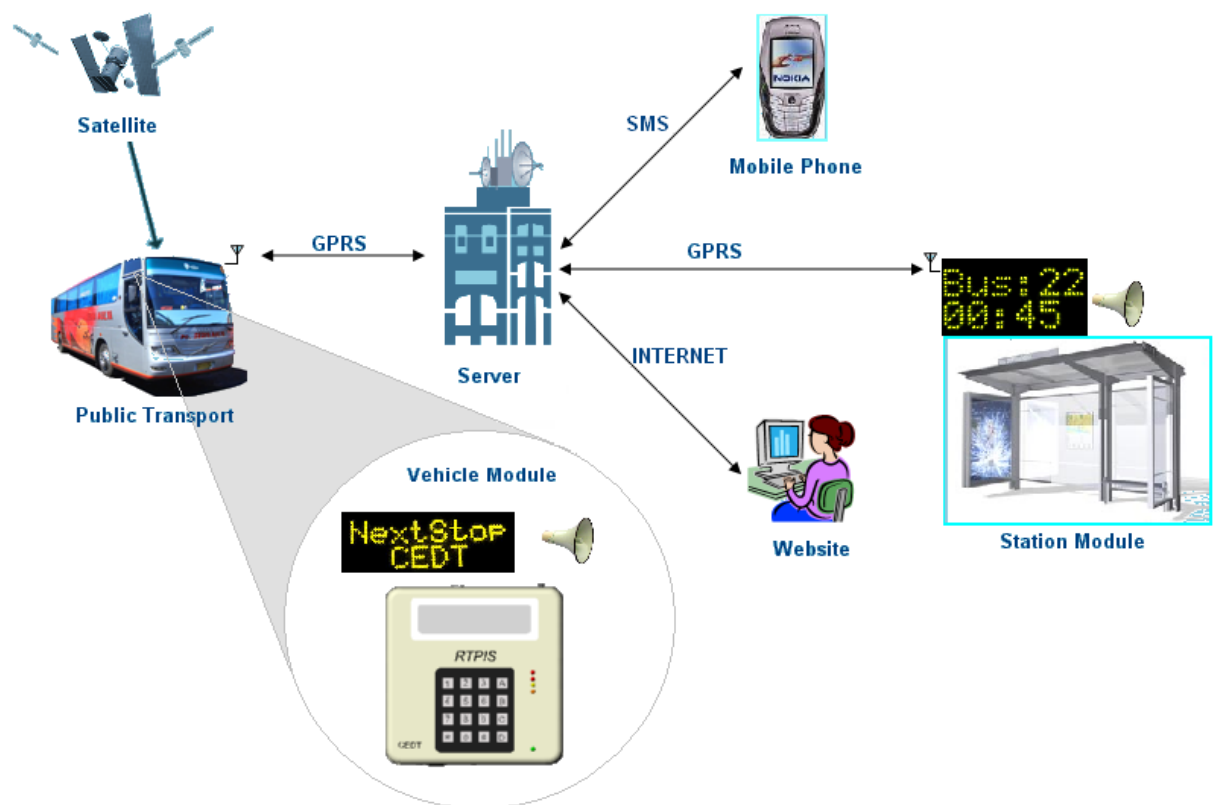


Figure 4: RTPIS Scenario [15]

### **2.3.4 Integration of Microcontroller with GPRS and GPS System to provide passenger information**

Study focusing on real time location tracking that is microcontroller-based PIS to improve the intelligent road transport system proposed system that works with GPS and GPRS. GPS and GPRS are integrated with the microcontroller; the GPS Provides a GIS functionality to provide better and updated road network information to the passenger [16].

### **CHAPTER 3 SYSTEM DESCRIPTION**

In this next chapter description of the system that has been developed in this thesis work and the three delivery systems are discussed briefly.

GPS based train tracking system provides an opportunity to the passenger to track the current location of the train and get the time it will take to reach the transit station. PIS estimates the arrival time of the train in accordance to real speed of the train. This system helps the passengers to make better travelling decision as passengers can decide if they should wait for the train or not in accordance to the estimated arrival time. GPS coordinates of Addis Ababa is Latitude: 9° 01' 29.89" N Longitude: 38° 44' 48.80" E [17]. AALRT trains doesn't have installed GPS receiver on them, beacon localization is used for transmit signals data of the train track inside stations, Beacons transmit small amounts of data via Bluetooth Low Energy (BLE) up to 50 meters, and as a result are often used for indoor location technology, although beacons can be used outside as well.

### 3.1 PROPOSED SOLUTION

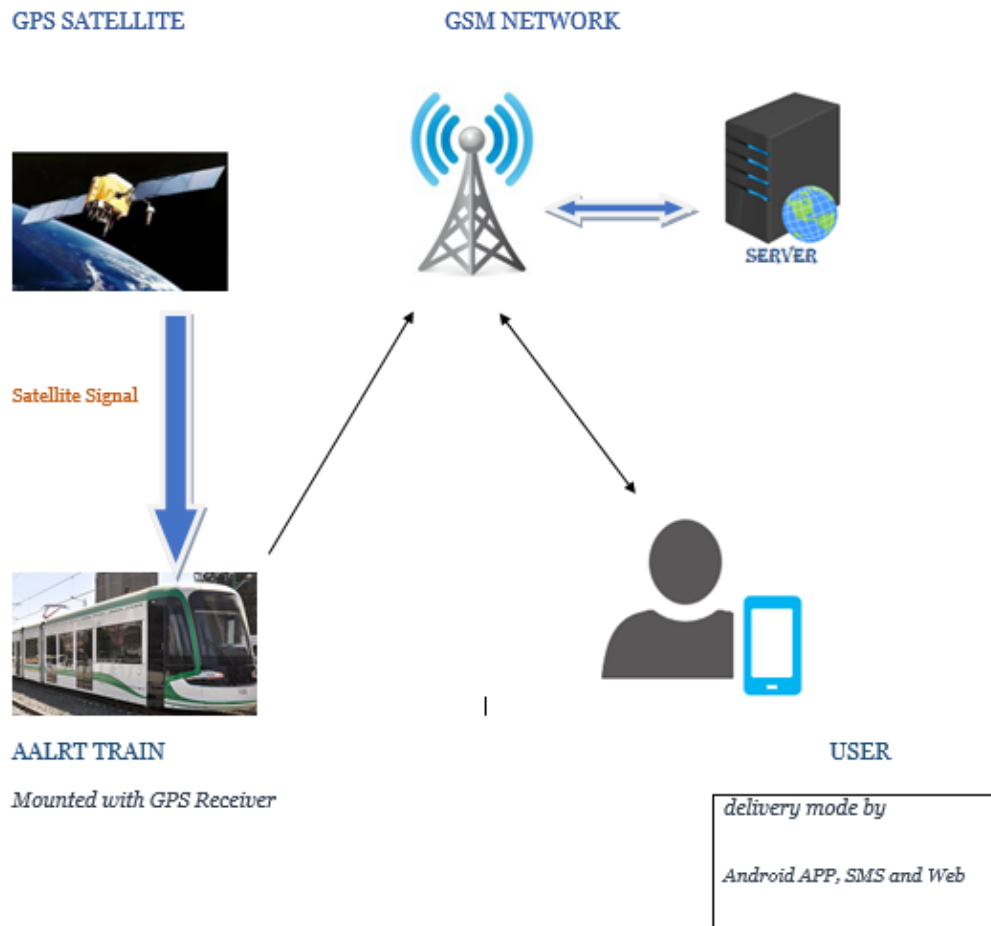


Figure 5: Architecture of Mobile enabled PIS system for AALRT

GPS is used for locating the position of AALRT trains. We can track those trains through application using GPS to find out location of a train. In these applications which requires login of administrator for train details, we use those train details from train Registration Form i.e. (train color, train code) to locate the desired train location data. This is the Administrative Activity, from that detail we can track the location of train, only registered trains location is tracked. After getting the request the admin panel gets the real-time information of the train and then sends to the user via GPRS or SMS.

Android application is to be developed for client sides as android applications are at its peak and user friendly. GPS based system that tracks the current location of the train and the stations to calculate the distance between the same. Each of the components of the system is described as follows:

### **3.1.1 GPS**

GPS was developed by the U.S. Department of Defense for the military; the Global Positioning System (GPS) is a worldwide, satellite-based, radio navigation system that will give us the exact position of our train, no matter where they are, what time it is, or what the weather is like. A total of 24 satellites orbit the Earth, monitored continuously by earth stations. The satellites transmit signals that can be detected by GPS receivers located in our train and used to determine their location with great accuracy.

Each GPS satellite transmits radio signals that enable the GPS receivers in our train to estimate the satellite's location, as well as the distance between it and our train. The receivers then use these measurements to calculate where our desired train is located on Earth and convert the calculations into geodetic latitude and longitude. A receiver needs signals from three GPS satellites to pinpoint our train's position.

GPS based system accounts for GPS based current coordinates of the train and the passenger and the real time speed of the train.

### **3.1.2 GPS Receiver**

The GPS receiver which is to be mounted on AALRT train gets a signal from each GPS satellite. The satellites transmit the exact the time and location of a train in longitude and latitude signals are sent. By subtracting the time, the signal was transmitted from the time it was received; the GPS can tell how far it is from each satellite. The GPS receiver also knows the exact position in the sky of the satellites, at the moment they sent their signals. So, given the travel time of the GPS signals from three satellites and their exact position in the sky, the GPS receiver can determine train position in three dimensions – east, north and altitude. Receiver gets the current GPS coordinates in terms of Latitude and Longitude of the train. Also calculates the speed with which the train is moving and to transmit the computed information to the central server using GPRS. Using location-based services GPS location of the train and the speed with which the train moving is calculated.

### **3.1.3 GSM Modem**

The next data collection unit of passenger information system is a recipient onboard and stationary GSM modem. GSM is a cellular network, which means that mobile phones connect to it by searching for cells in the immediate vicinity. Passenger information from GPS receiver module means location of both train and passenger in longitude and latitude including train speed is transferred to server using this GSM network. Our system is wireless so in wireless data transporting, GSM, SMS technology is common feature with all mobile network service provider. GSM modem is specialized type of modem which accepts a SIM card, operates on a subscriber's mobile number over a network. A GSM modem can be an external unit or a PCMCIA card (also called PC Card). An external GSM modem is connected to a PC through a serial cable, a USB cable, Bluetooth or Infrared. Like a GSM mobile phone, a GSM modem requires a SIM card from a wireless carrier in order to operate. PC's use AT commands to control a GSM modems. GSM modems support an extended set of AT commands. These extended AT commands are defined in the GSM standards. So, signal conditioning will convert its output in to Logic Zero and Lo. GSM modems support an extended set of AT commands. These extended AT commands are defined in the GSM standards. With the extended AT commands, you can do things like: Read, write and delete SMS messages, Send SMS messages. A GPRS modem is a GSM modem with additional support for GPRS technology for data transmission. GPRS means: 'General Packet Radio Service'. It is based on a packet-switched technology, as an extension to GSM, it should be noted that GSM is a circuit-switched. An advantage of GPRS over GSM is that GPRS has a much higher data transmission speed. GPRS can be used as the bearer of SMS. If SMS over GPRS is used, an SMS transmission speed of about 30 SMS messages per minute may be achieved. This is much faster than SMS over GSM. A GPRS modem is required to send and receive SMS via GPRS.

### **3.1.4 Server Module**

The server is at the center of our real time passenger information center it takes information data from both GPS receiver module mounted on the train and user android Module to find train and Passenger's exact location on the route.

The functions of the server are listed below:

- ❖ To maintain a database of all the routes, the train that ply on a route, the stops along each route
- ❖ To continuously receive location and speed from the vehicle units of all the train.
- ❖ To calculate the arrival time of all the train at their next and subsequent station stops.
- ❖ To reply to SMS-based queries requesting arrival time at specific stops from users; a GSM modem connected to the server transfers these queries to the server which processes them and sends the reply message.
- ❖ To host Internet web pages, which allow users to track trains in real time and get the arrival time for any route-stop pair and plan trips from any source to any destination stop at any time. Also new users such as foreigners or anybody who need the information of AALRT services can get this from the web page.

#### ***3.1.4.1 Distance Calculation***

The distance between two points is calculated as the sum of the distance between the two coordinating points and the distance between the train and its nearest stopping train station coordinating point. Distance between the two coordinating points is easily and efficiently calculated from database, while the distance between the train and it's nearest matched coordinating point is calculated using Haversine formula. Haversine formula calculates the distance between the two pairs of longitude and latitude considering in mind the spherical shape of the earth (ignoring any hills they fly over, of course!). [18]

*Mobile Enabled passenger information system: case of AALRT*

Haversine formula is illustrated as following ..... Eq3.1

Distance = Radius \* C

Radius = radius of earth, 6371 km

$$\Delta Lat = lat_2 - lat_1 \qquad \text{Lat=Latitude}$$

$$\Delta Long = long_2 - long_1 \qquad \text{Long=Longitude}$$

**Haversine formula:**  $a = \sin^2 \frac{\Delta Lat}{2} + \cos(lat_1) \cos(lat_2) \sin^2 \frac{\Delta Long}{2}$

$$C = 2 \times a \tan^{-1} ([\sqrt{a}, \sqrt{(1-a)}])$$

$$D = R.C$$

Where R is earth's radius (mean radius=6,371km):

Note that angles need to be in radians to pass to trig functions! Java Script is given here.

JavaScript:

```
Var R = 6371e3; // meters

var φ1 = lat1.toRadians();
var φ2 = lat2.toRadians();
var Δφ = (lat2-lat1).toRadians();
var Δλ = (lon2-lon1).toRadians();
var a = Math.sin(Δφ/2) * Math.sin(Δφ/2) +
        Math.cos(φ1) * Math.cos(φ2) *
        Math.sin(Δλ/2) * Math.sin(Δλ/2);
var c = 2 * Math.atan2(Math.sqrt(a), Math.sqrt(1-a));
var d = R * c;
```

**3.1.4.2. Estimation Time of Arrival**

Using GPS satellite after location and speed of the train is received by GPS receiver module mounted on train, the information is transferred to server using GPRS that periodically updates the data. The Server calculates estimation time of arrival (ETA) of train and using different delivery mode, it sends this information to the users which requests ETA. in which our case, the system is designed for AALRT passenger to access ETA using android, web and SMS.

After getting speed and location of train from GPS receiver, we can calculate ETA by the following Algorithm:

- A) Firstly, get the location and the speed of requested train of AALRT to be tracked using longitude and latitude location value received from GPS, then calculate distance using haversine formula eq (3.1)
- B) Get the location coordinates of the passenger using GPS.
- C) Find the location of nearest stopping station place for train and calculate distance using also haversine formula eq (3.1)
- D) Calculate the distance between the two coordinating points using entries in the database.

$$\sum_{i=T}^S Distance_i \dots\dots\dots (3.2)$$

where T= nearest coordinating point to the train:

S= nearest coordinating point to the Station

- E) Add the distance between the train and its nearest coordinating point. This is calculated

$$D = \sum_{i=T}^S Distance_i + Distance_{TS} \dots\dots\dots (3.3)$$

Where  $Distance_{TS}$  is distance between the train and its nearest coordinating station point.

- F) calculate the arrival time Train or ETA

From previously done research review, presented dwell time and passenger flow theory.

$$Dwell\ time_{Total} = \sum Dwell\ time_{particular\ station}$$

1. Train stop
2. Checking procedures (Entire train check etc.)
3. Doors opening
4. Alighting and boarding
5. Waiting for the timetable departure time
6. Check procedures (No blocked doors check etc.)
7. Doors closing
8. Final clearance and departure

Of the components above number 1-3 and 6-8 can be seen as system constants that depend on the technical systems and equipment. The time it takes for these components to happen can be seen as a technical time which is more or less the same every time for a particular case, therefore called constant in this study. [19]

Adding this Dwell time, we can EAT to be

$$\text{Using } T = \frac{\text{Distance}}{\text{speed}} + Dwell\ time_{Total} \dots\dots\dots (3.4)$$

### **3.1.5 Android Mobile Platforms**

Android is a new mobile device operating system first developed by Android Inc., a firm purchased by Google in 2005. The Android Operating System (OS) is built on top of a Linux kernel, all Android applications, including core applications, are developed in Java. Each Java program is run as a single process on Linux with its own instance of a Dalvik Virtual Machines (DVM). A DVM is an optimized java virtual machine for Mobile Devices [20]. A set of native C/C++ libraries is provided to support various additional functions for software developers. Android supports many features required for mobile software development. The application framework is so flexible that it allows all software components and extensions to be reused or customized. Another reason behind selecting android as a platform for developing this system is that android based mobile phones are used on a very large scale among people. Android is easy to handle and is user friendly, and hence the application will be used by maximum number of citizens. Android is an open source operating system for mobile phones. The primary idea here is to provide train timings to AALRT users in real time. The system will get the input from the driver at the start of the journey. The input provided by the driver is train code and start the GPS of the train at start of the journey. The train code is mapped with the GPS location of the train. In every 30 seconds the GPS location of the train is stored in server. When user search with respect to train code, the server searches the requested train code With the help of the database the location of the train is mapped on the map and the real time location of the train is display to the user, as the application is based on android, it will be easy enough for the user to understand. The Mobile phone GPS will find the current location of the user in terms of Latitude and Longitude and sends request to the server so that approximate time required by the train to reach the stop will be calculated.

### **3.2 Real Time Location Information Delivery modes**

Many real time tracking systems have been developed. Delivery mode of this Real time location varies in different countries, system may display the arrival time of the transit vehicle on the LCD screens that are placed on every stop, through SMS facility, web-based application or by using android application using GPRS (General Packet Radio Service).

Real time location-based models for AALRT designed in this thesis are illustrated below

- **SMS (Short message service):** is used over the GSM networks to transfer the train location coordinates. The location information is sent to the central server over the GSM networks using SMS and stored in the database. User sends the request and receives the information through SMS.
- **Through the web-based application:** users of AALRT can track the train using the web-based systems, it also enables users with different operating systems platforms to easily reach the details with the help of internet access. The most beneficent of this web design is that, foreigners and new passenger can get information relating to AALRT service information like of, where to be found of stations location and place of ticket purchasing location. This can be achieved by describing reference location to each station, so that passenger can easily access the locations without wondering around searching for the places. This reduces effort, time and energy wasted with lack of information, also foreigner doesn't need to know Amharic to get the whole service information.
- **Android application:** that enables AALRT passengers to access ETA using their android smart phone. Android has inbuilt GPS service provided by the Smartphone to get its GPS coordinates. These coordinates are transferred to the central server. Users can retrieve information through android application.

### **3.2.1 SMS Based Train Tracking System**

SMS stands for short message service. It is a method of communication that sends text between cell phones. The maximum size of the text messages is: 160 characters (letters, numbers or symbols in the Latin alphabet). For other alphabets, such as Chinese, the maximum SMS size is 70 characters. It is well known that SMS service is a cell phone feature but indeed, SMS can also work on other computing devices now a days, such as PC, Laptop, or Tablet PC as long as they can accept SIM Card. SIM Card is needed because SMS service needs SMS center client which is built in on the SIM Card. The alternative solution of using SMS connection reduces both the cost and the round-trip time of message delivery.

- The SMS facility is a convenient way of delivering limited amount of information (no need for dialing and negotiating a radio link data connection).
- With predetermined traffic network layout, the length of the SMS message is sufficient to include all data needed to deliver a request and its response in two single transactions

#### **3.2.1.1 Major Components of the SMS based Tracking System**

##### **i) GPS module:**

GPS receiver device is installed over the tracking train. GPS works in any weather condition, anywhere in the world, 24 hours a day. GPS antenna is connected to the right jack to make sure it receives the signal. GPS device is turned on and tuned to receive signals from satellite. Device is capable of receiving longitude and latitude coordinates of the location. GPS data is regularly sent to the server using SMS service over the GSM networks. For this purpose, SIM card is inserted to connect to the GSM networks. Server analyses the data.

**ii) Server module:**

It manages and analysis the GPS data and manage the queries from the users. Initially a setup message is sent from the server to the train module to set the time interval at which GPS data is sent. SMS enabler acts as the interface between the train module and the server. PC suit is installed at server side. SMS enabler receives the SMS from the train module and extracts the longitude, latitude values from the entire message and stores in the database.

**iii) User module:**

Users send SMS to the server to retrieve real time location of the train. Server-side application processes the user's request and fetches the location name from the database for the corresponding train code number and sends a reply to the user. On server side initially, user message is validated if it's in a valid format with valid train code number. After this, request is processed and receives a reply from the server. There are two ways to send SMS from a computer to a phone. Firstly, by connecting a GSM modem to a computer, by applying AT commands GSM modem sends the SMS. Secondly, by connecting computer to the SMS center (SMSC) to send SMS using a protocol. SIM (Subscriber identity module) card is inserted in the GSM modem that is connected to the computer.

### **3.2.2 Web System**

Web designed here queries about the service of AART. It also enables users with different operating system platforms to easily query the real time location of train using the internet accesses. The coordinates of the train to be tracked is acquisitioned from the satellite using GPS receiver. Coordinates are sent to the server using GPRS service over GSM networks as a HTTP packet.

### **3.2.2.1 Major components of the system**

- **Web design:** Overall functionality of the system is eased using various web application languages. Objectives of the web application are to define and manage all users' accounts and train account. Receiving the GPS coordinates and storing them in database corresponding to the train code number. Replying to the client's request to get real time location of a particular Train code number and displaying the information on web. Designing of this web page the real time location are reviewed in text form, making this web having additional benefit of added feature of basic AALRT service information and stations location is listed for new users.
  
- **II) Communication – train and server:** GPS device on tracked train queries the coordinates, satellite responds with string of data with longitude and latitude coordinates, speed, and time. Details are separated from the received data and the set of data along with the train code number are sent through the GPRS to the path defined in the URL using HTTP get method.

### **3.2.3 Mobile Application**

Getting real time location of the vehicle has become very easy with the rapid increase in the use of programmable smart phones with built in GPS facility. The massive influx in the use of Android has made the system very user friendly. It has been observed that by using Android phones the acceptance to the system has increased enormously. Location Based services (LBS) are used by Android to get the real time location of the train. Android Mobile Application system is installed by the users on their programmable Smartphone from the Android market.

## *Mobile Enabled passenger information system: case of AALRT*

The application provides with user friendly interface where user selects the route numbers and the train number from the map. With this request at web service is generated along with GPS location of the passenger. Location of passenger is tracked using Location Based Services. In response to this web service sends the detailed information in which the passenger might be interested in. The detailed information includes the train's current speed, Distance between the current locations of the train and the stations and the time it will take for the train to reach the desired stations by passenger, the estimated time to reach those destinations is reviewed on the Google Map.

## **CHAPTER 4 PIS SIMULATION**

PIS are designed for AALRT passengers, simulation analysis that three delivery modes is completed successfully, those information delivery system are described below:

- Android App,
- Web based
- SMS based

### **4.1 Android Application Development**

As explained above Android Smartphone consists of GPS system, with the use of which the current location of the train and the passenger can easily be tracked since most AALRT passenger use Android smart phone, Android application is proposed for the system. One of the major features provided by Android is the set of Android applications (apps) provided through Android market that enables the users to extend the functionality. Android Software Development Kit (SDK) is a set of tools and APIs (Application Program Interface) that facilitates the development of Android application. To develop Android apps, set of tools are provided in Android SDK. after downloading and installing Android SDK these tools can be used from Eclipse IDE (Integrated development environment) through ADT plugin (Android development tools) or from command line. Google developed ADT plugin with a purpose of developing Android applications in Eclipse. Eclipse makes it very easy for all the Android developers to create the android projects and debug the programs whenever needed.

## *Mobile Enabled passenger information system: case of AALRT*

Android was preferred because of its open source nature, ease of development, zero barrier to entry and pervasiveness. It quickly gained attractions by developers because of its fully developed features to exploit the cloud-computing model offered by web resources and it also enhances that experience with local data stores on the handset itself. Android studio version 3.2.1, programming Language Java and Google Maps API is used to development of AALRT passenger information system application. Local XAMPP server is used as storage and database device.

What goes on in an Android application is mainly divided into two, including the visual part, which the user interacts with and the non-visual, which runs in the background. The visual part is termed an activity. An activity is usually a single screen that the user sees on the device at one time. An application typically has multiple activities, and the user flips back and forth among them. Figure 6 below shows an activity life-cycle and the different states which it can exist in. The non-visual parts are called Services. They run in the background and do not have any user interface components. They perform actions such as polling data from a server on the Internet, playing music in the background etc. They are responsible for those actions that must continue while the user flips between activities of the same or different applications. Broadcast Receiver is also an important component of Android development. It is the system's way of alerting applications of certain events in the systems when they occur. For example, the system broadcasts an SMS arrival event when a new SMS arrives on the mobile phone. Subsequently all application that are registered to receive this events will be notified and each of them will respond accordingly.

Figure 6 shows activity life cycle of each android application

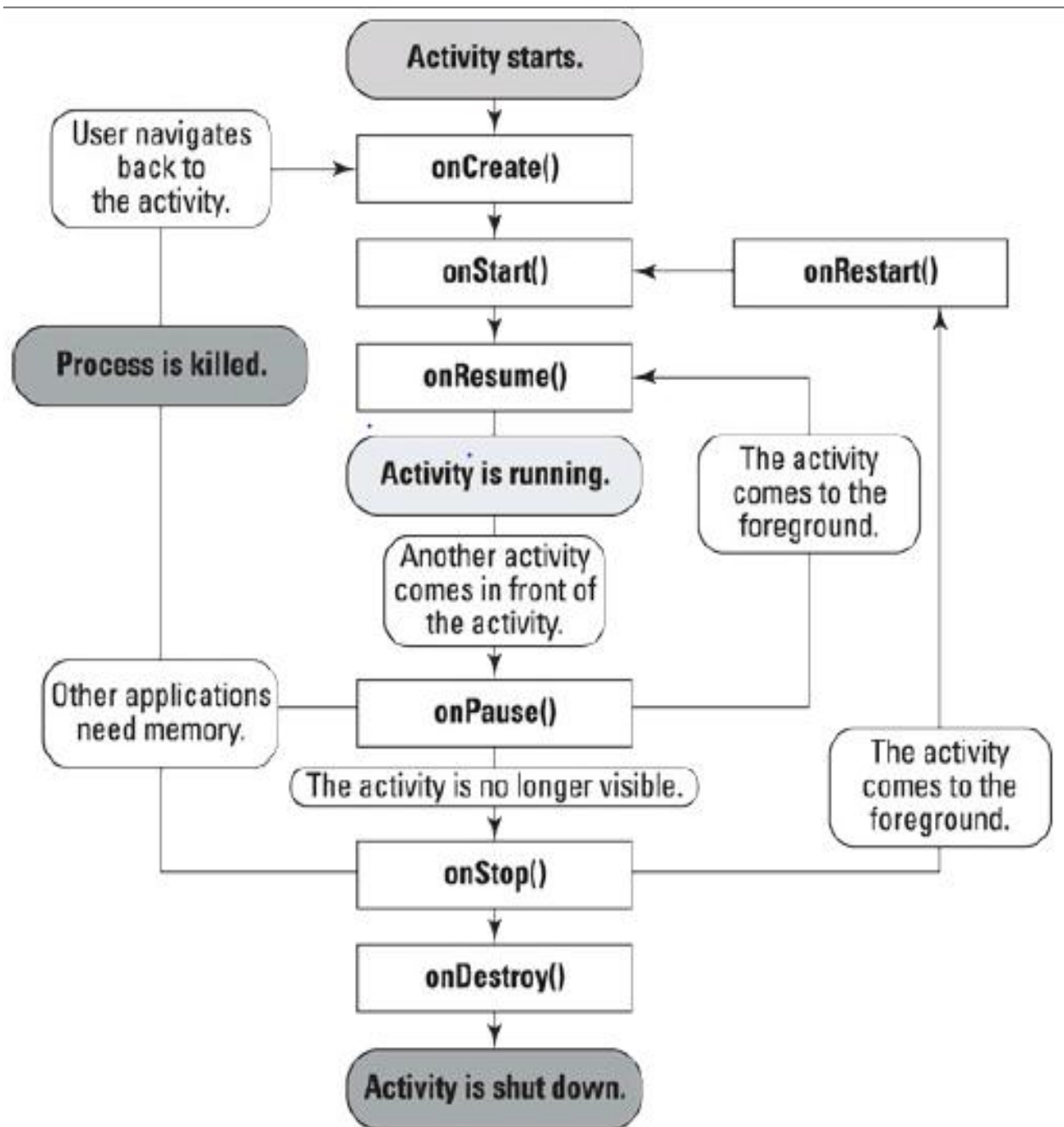
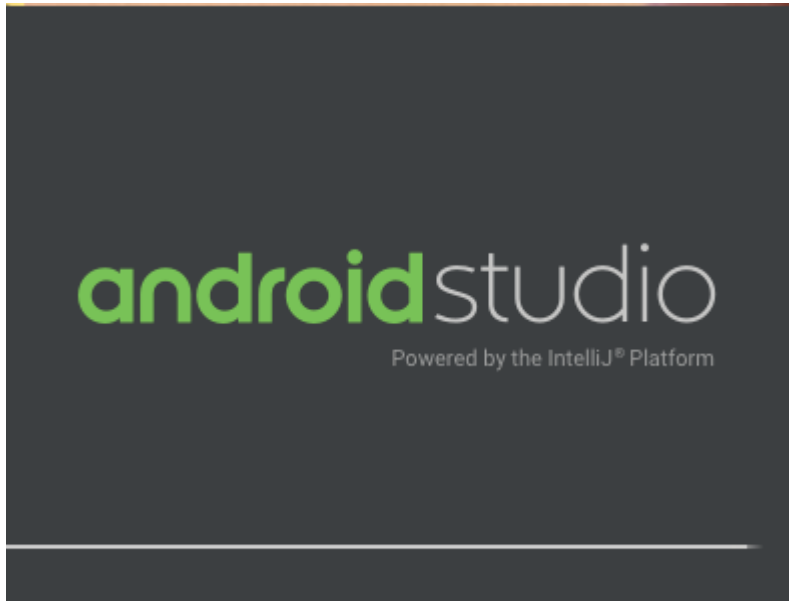


Figure 6: Activity Life cycle [21]

The rectangles represent callback methods you can implement to respond to events in the activity. The shaded ovals represent the major states of the activity. All activities implement *onCreate()* for initialization and may also implement *onPause()* for cleanup

- ❖ **The entire lifetime** takes place between the first call to *onCreate()* and the final call to *onDestroy()*. The activity performs all global setup in *onCreate()* and releases all remaining resources in *onDestroy()*. For example, if you create a thread to download a file from the Internet in the background, it may be initialized in the *onCreate()* method. That thread can be stopped in the *onDestroy()* method.
- ❖ **The visible lifetime** of the activity takes place between the *onStart()* and *onStop()* methods. During this time, the user can see the activity onscreen (though it may not be in the foreground interacting with the user, which can happen when the user is interacting with a dialog box). Between these two methods, you can maintain the resources that are needed to show and run your activity. For example, you can create an event handler to monitor the state of the phone. The phone state can change, and this event handler can inform the activity of the phone entering Airplane mode and react accordingly. You would set up the event handler in *onStart()* and tear down any resources you're accessing in *onStop()*. The *onStart()* and *onStop()* methods can be called multiple times as the activity becomes visible or hidden to the user.
- ❖ **The foreground lifetime** of the activity begins at the call to *onResume()* and ends at the call to *onPause()*. During this time, the activity is in front of all other activities and is interacting with the user. An activity normally toggles between *onResume()* and *onPause()* multiple times, for example, when the device goes to sleep or when a new activity handles a particular event — therefore, the code in these methods must be fairly lightweight.

Through the development of this AALRT passenger information system application, installing Android studio on computer is first steps, Android Studio 3.4 is used

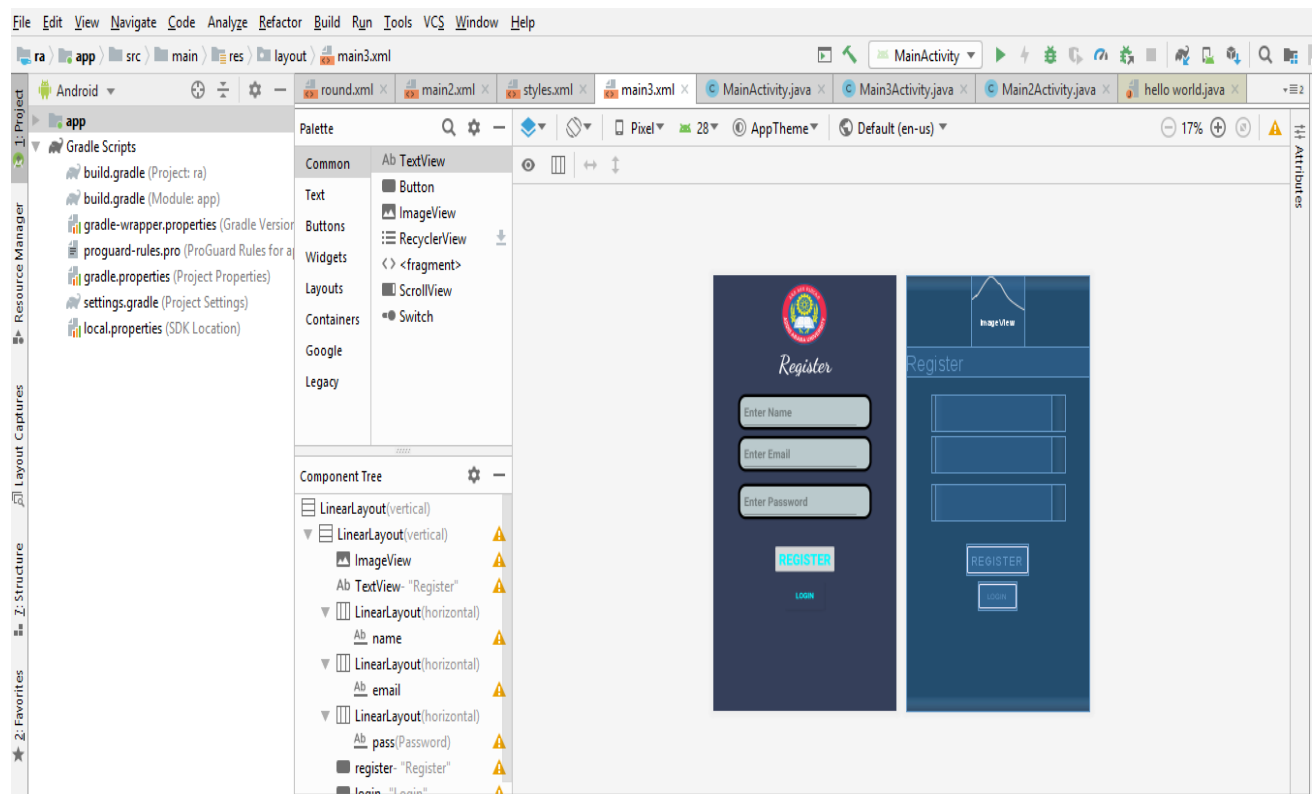


*Figure 7:Android Studios 3.4*

The Android Software development kit (SDK) provides the tools and application programming interfaces (APIs) necessary to begin developing applications on the Android platform using the Java programming language

## *Mobile Enabled passenger information system: case of AALRT*

The Android SDK provides an extensive set of application programming interfaces (APIs) that is both modern and robust. Android applications are written in Java programming language. Although the Android operating system consists primarily of Java code, some of the framework isn't written in Java. Android apps use small amounts of XML in addition to Java. Using the Google Maps API provided by Android studio AALRT train tracking application is created.



*Figure 8: Screenshot of developed AALRT Passenger information application user interface layout on Android studio*

## Mobile Enabled passenger information system: case of AALRT

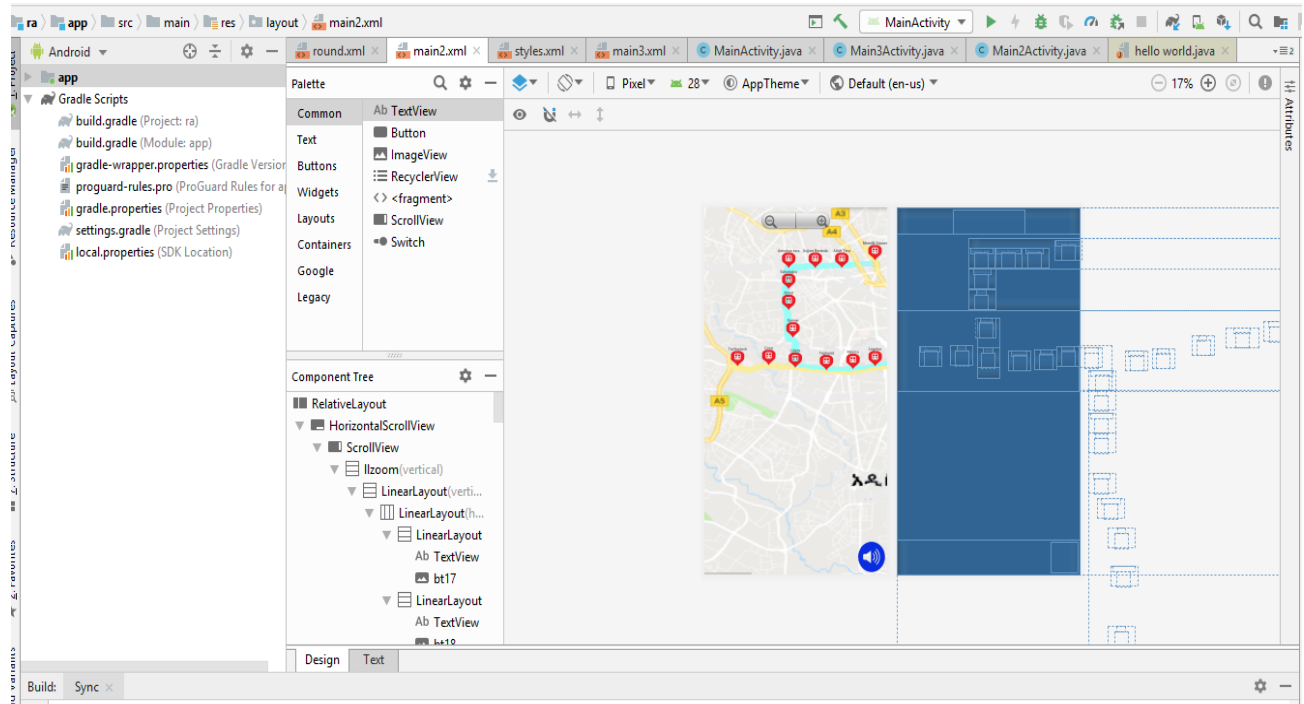


Figure 9: Screenshot of developed AALRT Passenger information application layer on Map using Android studio

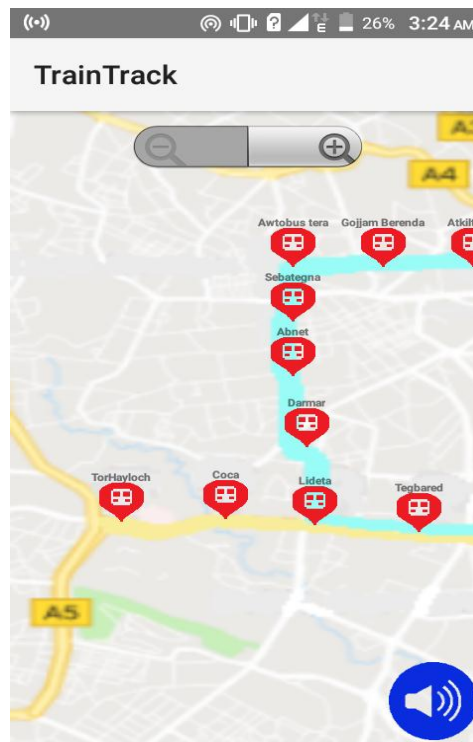
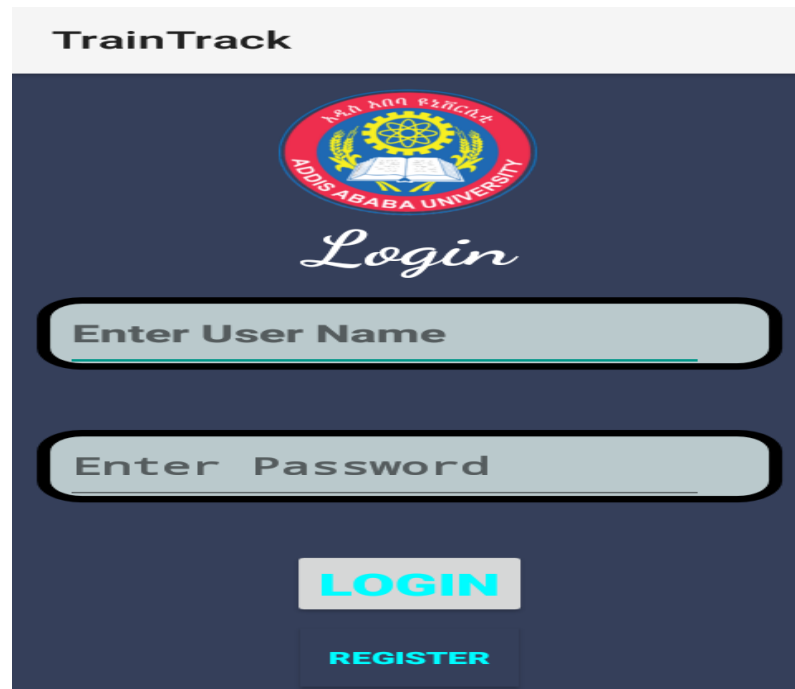


Figure 10: Screenshot of AALRT stations on Map

After writing necessary java and Xml source code for this tracking system next we can install this application on our device using Emulator, An emulator (also known as an AVD) is an Android Virtual Device that looks, acts, walks, like a physical Android

device. AVDs can be configured to run just about any particular version of Android. Appendix A shows Main Activity Java source code used for developing this application.

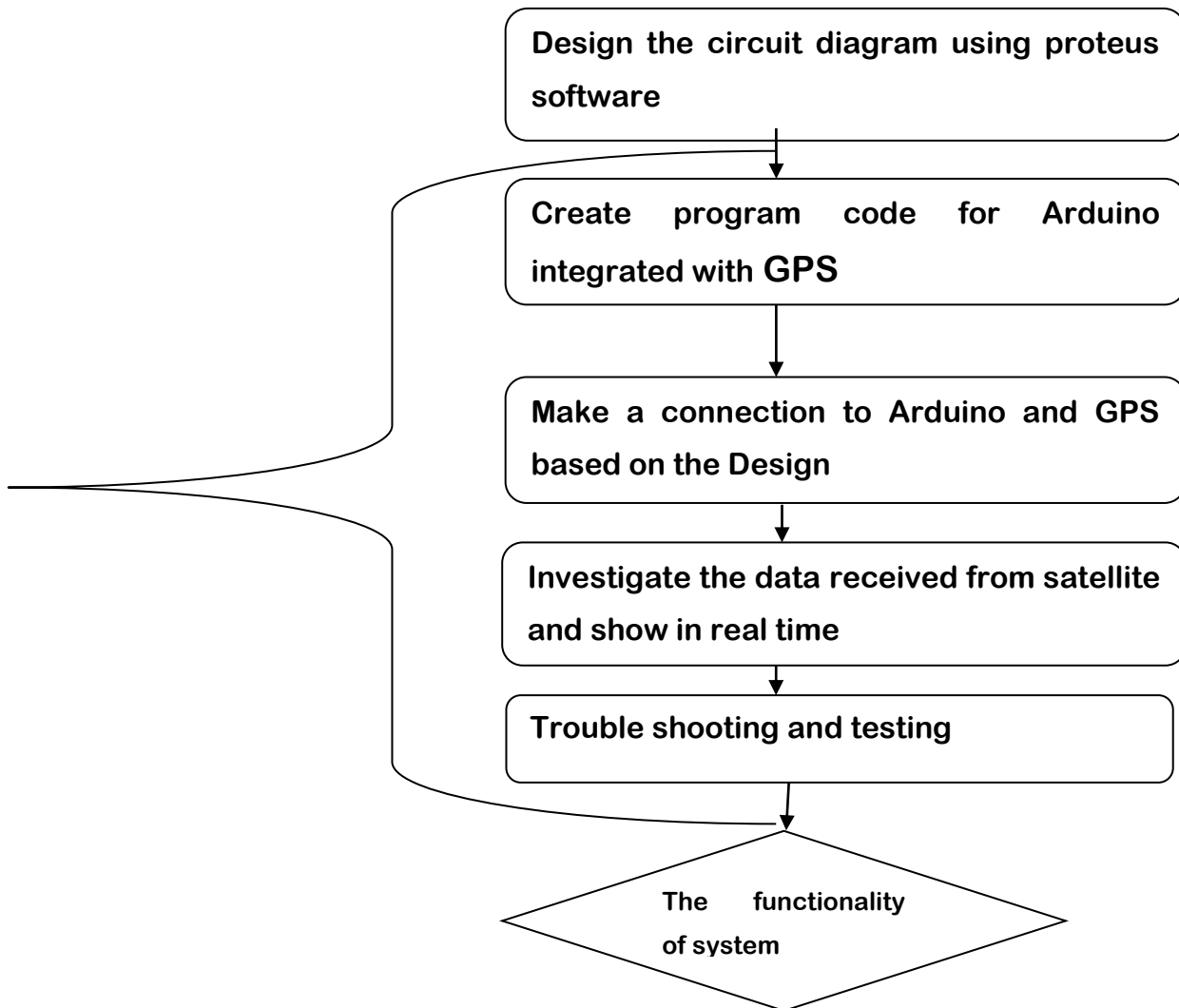


*Figure 11: screenshot of AALRT Passenger Information Android App outlook*

#### **4.2. Tracking of location simulation by PROTEUS Software**

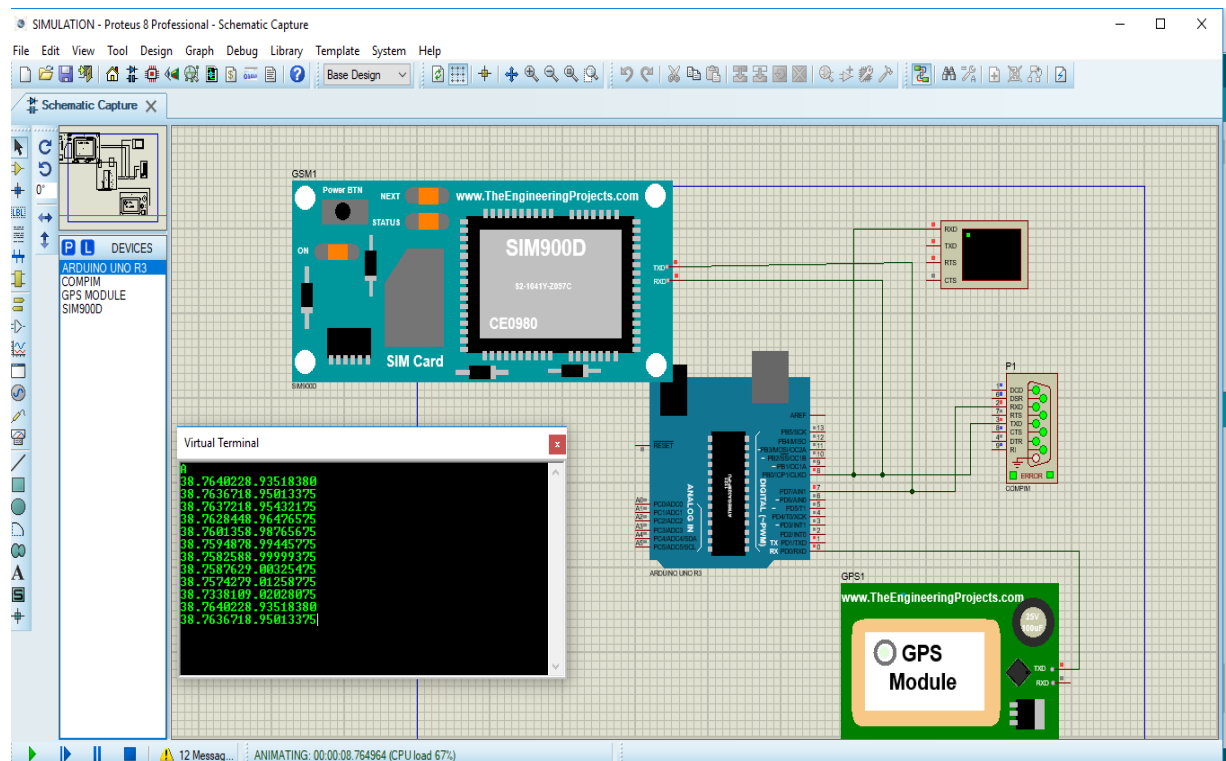
In order to get the exact train location in terms of latitude and longitude there is no real GPS for tracking system; therefore trains are assigned virtually for simulation purpose. Proteus software is used for location tracking in order to get latitude and longitude of the train. The Proteus Design Suite is electronic circuit design software which includes a schematic capture, simulation and PCB (Printed Circuit Board) Layout modules. Virtual Prototyping enables system testing before the first physical PCB is ordered. Arduino and GPS module connected using serial communication is used as libraries to simulate the tracking system.

Below *figure 12* shows steps in order to simulate tracking system using proteus software.



*Figure 12: Flow chart for Tracking System Simulation using Proteus Software*

Here circuit diagram which is created using proteus simulates to get coordinate values of the train location, system has arduino module. GPS module connected using serial connection.



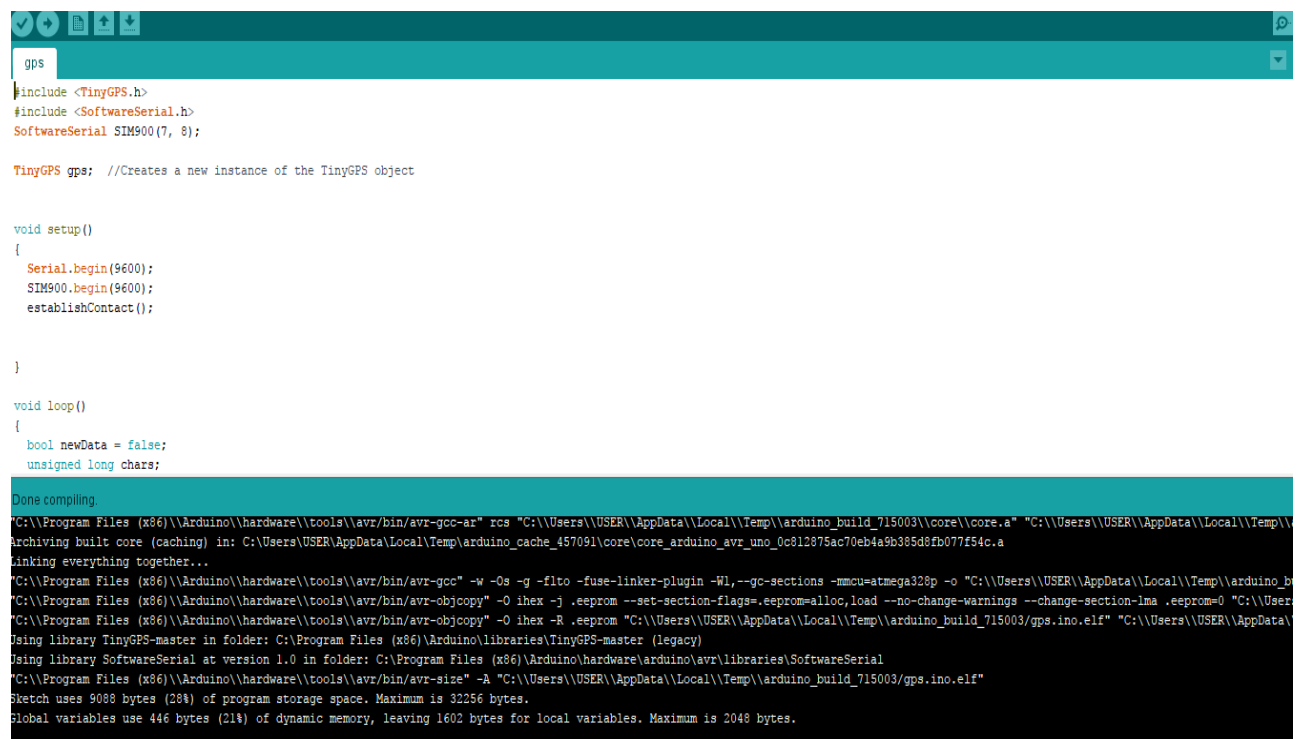
*Figure 13: proteus simulation of AALRT train tracking system*

To implement this in real hardware design, we just have to install GPS receiver and GSM modem in our train to get the latitude and longitude values of train location. GPS receiver calculates its position by precisely timing the signals sent by GPS satellites high above the Earth. Each satellite continually transmits messages that include the time message was transmitted, train moving speed and Satellite position at time of message transmission. This makes the system to be less complex in hardware implementation.

### 4.3 Arduino Module

Hardware tracking system simulation is tested using Arduino. Arduino is a tool for making computers that can sense and control more of the physical world than your desktop computer. It's an open-source physical computing platform based on a simple microcontroller board, and a development environment for writing software for the board. Arduino can be used to develop interactive objects, taking inputs from a variety of switches or sensors, and controlling a variety of lights, motors, and other physical outputs.

As shown above, successfully the desired latitude and longitude location of the train T from GPS module which has its own libraries using serial connection to Arduino is generated within few seconds from proteus.



```
gps
#include <TinyGPS.h>
#include <SoftwareSerial.h>
SoftwareSerial SIM900(7, 8);

TinyGPS gps; //Creates a new instance of the TinyGPS object

void setup()
{
  Serial.begin(9600);
  SIM900.begin(9600);
  establishContact();
}

void loop()
{
  bool newData = false;
  unsigned long chars;
```

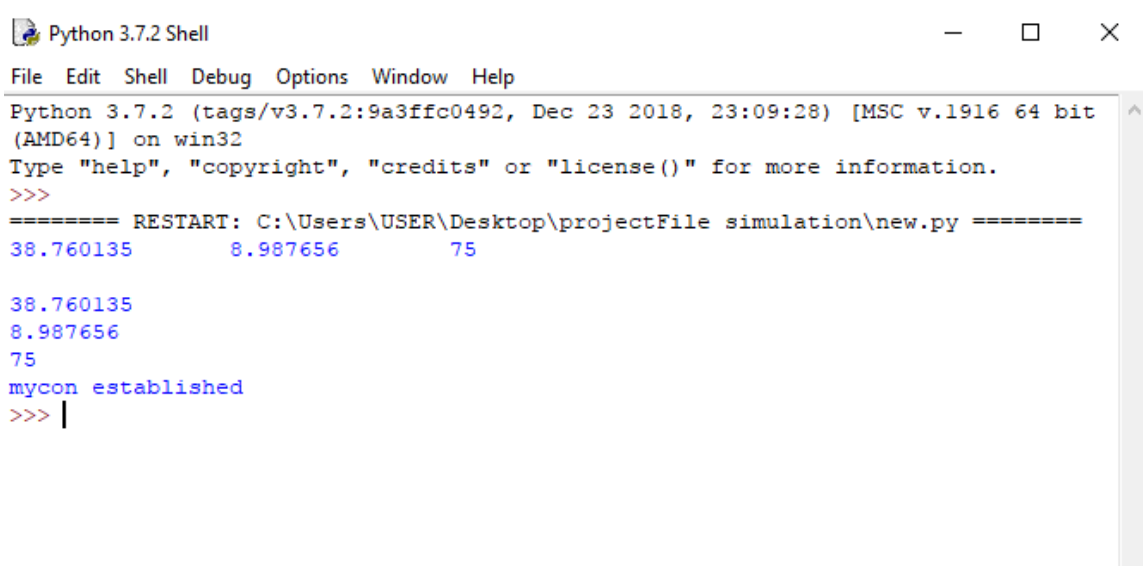
```
Done compiling.
C:\Program Files (x86)\Arduino\hardware\ttools\avr\bin\avr-gcc-ar" xcs "C:\Users\USER\AppData\Local\Temp\arduino_build_715003\core\core.a" "C:\Users\USER\AppData\Local\Temp\
Archiving built core (caching) in: C:\Users\USER\AppData\Local\Temp\arduino_cache_457091\core\core_arduino_avr_uno_0c812875ac70eb4a5b385d8fb077f54c.a
Linking everything together...
C:\Program Files (x86)\Arduino\hardware\ttools\avr\bin\avr-gcc" -w -Os -g -flto -fuse-linker-plugin -Wl,--gc-sections -mmcu=atmega328p -o "C:\Users\USER\AppData\Local\Temp\arduino_b
C:\Program Files (x86)\Arduino\hardware\ttools\avr\bin\avr-objcopy" -O ihex -j .eeprom --set-section-flags=.eeprom=alloc,load --no-change-warnings --change-section-lma .eeprom=0 "C:\User
C:\Program Files (x86)\Arduino\hardware\ttools\avr\bin\avr-objcopy" -O ihex -R .eeprom "C:\Users\USER\AppData\Local\Temp\arduino_build_715003/gps.ino.elf" "C:\Users\USER\AppData\
Using library TinyGPS-master in folder: C:\Program Files (x86)\Arduino\libraries\TinyGPS-master (Legacy)
Using library SoftwareSerial at version 1.0 in folder: C:\Program Files (x86)\Arduino\hardware\arduino\avr\libraries\SoftwareSerial
C:\Program Files (x86)\Arduino\hardware\ttools\avr\bin\avr-size" -A "C:\Users\USER\AppData\Local\Temp\arduino_build_715003/gps.ino.elf"
Sketch uses 9088 bytes (28%) of program storage space. Maximum is 32256 bytes.
Global variables use 446 bytes (21%) of dynamic memory, leaving 1602 bytes for local variables. Maximum is 2048 bytes.
```

*Figure 14: Screenshot of arduino software used for proteus simulation*

After extracting the train location in terms of latitude and longitude value, the next step is to transfer this value to the local server XAMPP using Python and VSPE.

## **Python MySQL**

Python provides interfaces to all major commercial databases. Python is an interpreter, high-level, general-purpose programming language. Created by Guido van Rossum and first released in 1991, Python's design philosophy emphasizes code readability with its notable use of significant whitespace. Its language constructs and object-oriented approach aims to help programmers write clear, logical code for small and large-scale projects.



```
Python 3.7.2 Shell
File Edit Shell Debug Options Window Help
Python 3.7.2 (tags/v3.7.2:9a3ffc0492, Dec 23 2018, 23:09:28) [MSC v.1916 64 bit
(AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:\Users\USER\Desktop\projectFile simulation\new.py =====
38.760135      8.987656      75

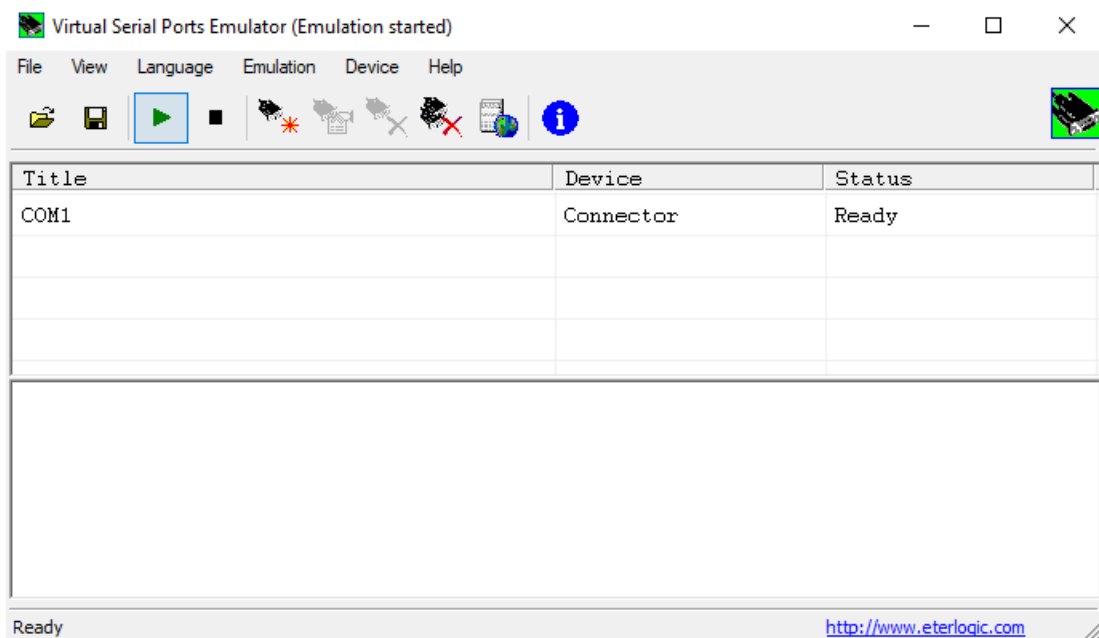
38.760135
8.987656
75
mycon established
>>> |
```

*Figure 15: Screenshot of python*

## **Serial Data Transmission**

The digital data can be transmitted between any two devices in two ways, parallel or serial. Serial data transmission means information is transmitted from source to destination over a single pathway and one bit is transmitted at a time.

- ❖ **Virtual Serial Port Emulation (VSPE):** VSPE is used to help developers to create applications that use serial ports. It is able to create various virtual devices to transmit/receive data. One Serial port can be opened into many different applications and use their different functionality. With VSPE you are able to share physical serial port data for several applications, create virtual serial port device pairs and so on. Here file named COM1 is created in order to make the software communicate with each other and transfer data.



*Figure 16: Screenshot of VSPE*

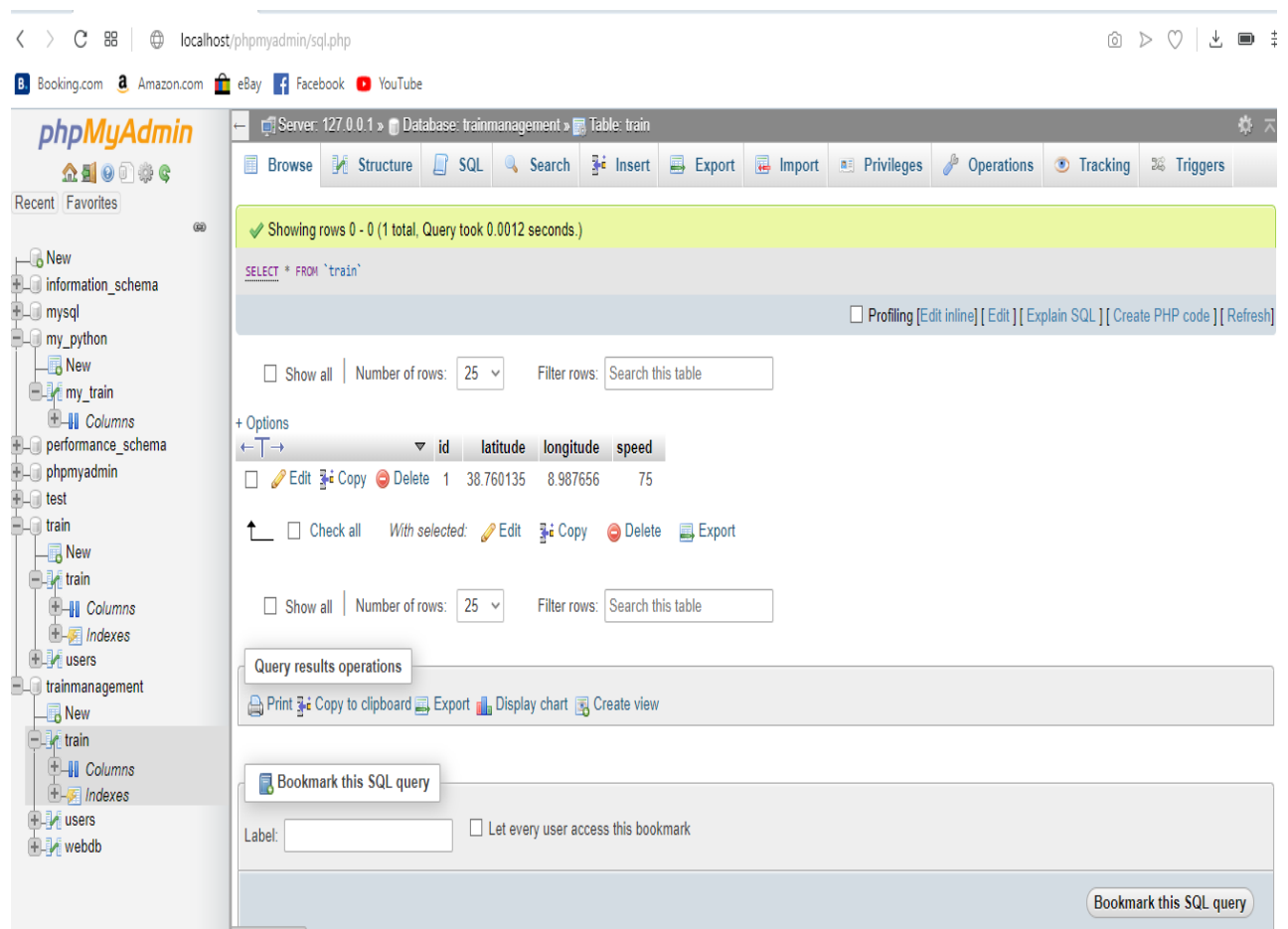
#### **4.4 Server Design**

MySQL server3.2.2 is used where tables for all the routes are present. Each table comprises the Coordinating station point, Latitude, Longitude, Distance and Direction. Points are set up along the route at the distance of 100 meters or at the train stops, whichever is closest. These coordinating points are essential for easy calculation of the distance between the train and stations and results in increase in accuracy. A Google map is used to plot the location of the train.

## Database Connection

I am using XAMPP for my database. PHP program to fetch data from localhost server database using XAMPP. XAMPP is a free and open source cross platform web server solution stack package developed by Apache which allow a web application to be easily tested on local web server. Here, we can manually create a relational database and store data in tabular form by going to this link. But to operate on local host or for storing data first we have to start Apache and MySQL from XAMPP control panel. Here I created database name Train management, table name that have user info having column name as ID, First Name, User Email and Password and we have to fetch the data stored there. So, below is the PHP program whose task is to fetch data.

Below is database named train management that is created to store latitude and longitude value received from proteus using python connectivity system



*Figure 17: Server Database*

## *Mobile Enabled passenger information system: case of AALRT*

This Database is designed to store all received train information, user information and the location of train information. Information to be stored in the database includes

- ❖ Information about users of the system
- ❖ Information about train
- ❖ Information received from train

This database is accessible from internet to authorized users. Information is stored in table form as follows in Table 1.

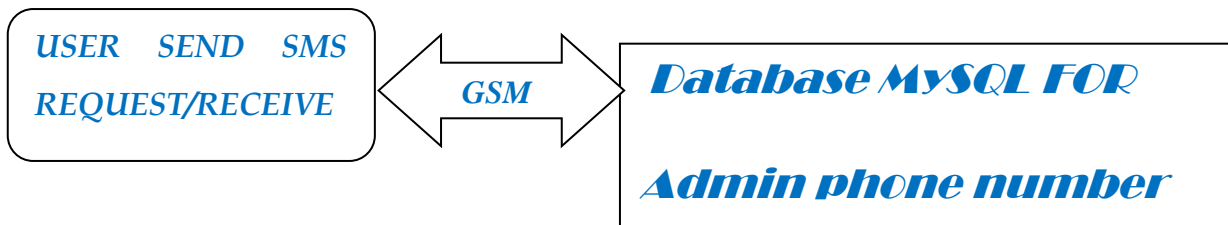
Train ID	Route	Train Stops	Latitude	Longitude	Date	Time	Distance	Speed
T	Kality to Menillik Square	Saris	30.0320932	30.0320932	16.6.2019	5.08AM	23km	45km/hr.

***Table 1: Database storage of tracking system***

#### **4.5 SMS Delivery Development**

The other information delivery mode is using SMS; Short Messaging Service (SMS) is a feature available on all mobile phones which allows a small amount of text to be sent between one user and another. Passengers can send request to server that stores real time of the AALRT train location using SMS request using specified train code, if the code is right, immediately the server will response the train info through again using SMS. To do this we have to connect GSM modem both inside the train and at server side.

For the simulation purpose one Mobile phone is used as server to reply to the other person ETA request. During the design of the android application additional java code was added to activities so that the system reply using SMS, example if a person sends SMS “T” to Admin phone number that is already registered on XAMPP local database admin phone number that is being used as server immediately replies an SMS containing of current location of train ”T”.



*Figure 18: Flow chart for simulation of SMS based system*

#### **4.6 Web Design Development**

Coordinates of the train to be tracked is acquisitioned from the satellite using GPS receiver. Coordinates are sent to the server using GPRS service over GSM networks as a HTTP packet. Following are the major components of the system. HTTP stands for Hyper Text Transfer Protocol. This is a basis for data communication in the internet. The data communication starts with a request sent from a client and ends with the response received from a web server. A website URL starting with “http://” is entered in a web browser from a computer (client). URL is an acronym for Uniform Resource Locator and is a reference (an address) to a resource on the Internet. The browser can be a Chrome, Firefox, Opera or anything else. Browser sends a request sent to the web server that hosts the website. The web server then returns a response as a HTML page or any other document format to the browser. Browser displays the response from the server to the user.

## *Mobile Enabled passenger information system: case of AALRT*

Using my own computer as server local host web page representing AALRT passenger information website is created for simulation purpose. XAMPP stands for Cross-Platform (X), Apache (A), MySQL (M), PHP (P) and Perl (P). It is a simple, lightweight Apache distribution that makes it extremely easy for developers to create a local web server for testing purposes



*Figure 19: Screenshot of AALRT passenger information Wen page*

XAMPP has four primary components. These are:

1. Apache: Apache is the actual web server application that processes and delivers web content to a computer. Apache is the most popular web server online, powering nearly 54% of all websites.

## *Mobile Enabled passenger information system: case of AALRT*

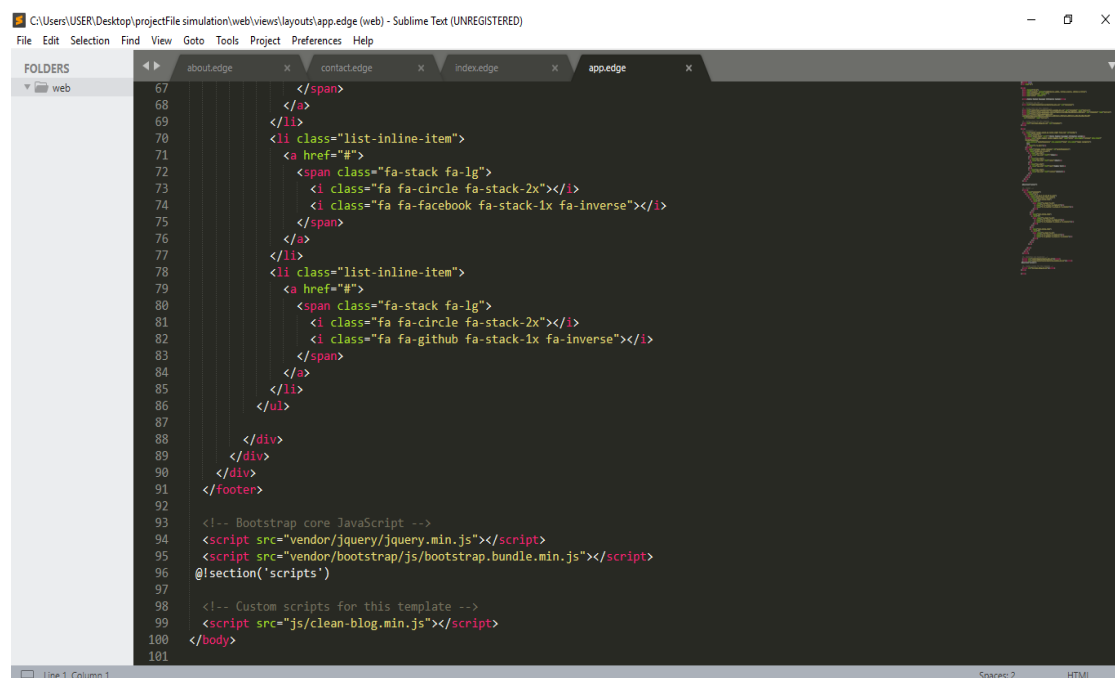
2. MySQL: Every web application, however simple or complicated, requires a database for storing collected data. MySQL, which is open source, is the world's most popular database management system

3. PHP: PHP stands for Hypertext Preprocessor. It is a server-side scripting language that powers some of the most popular websites in the world. It is open source, relatively easy to learn, and works perfectly with MySQL, making it a popular choice for web developers.

4. Perl: Perl is a high-level, dynamic programming language used extensively in network programming, system admin, etc. Although less popular for web development purposes, Perl has a lot of niche applications

Steps taken developing this AALRT web page are as follows:

After XAMPP has installed PHP successfully, fire up Notepad or sublime and type the source code into a new document: Save this file as '4000' in c:\xampp\htdocs\ (or whichever directory you installed XAMPP in). Navigate to localhost: 4000 to the open web page. Source code used to design AALRT web id given in Appendix B



```
C:\Users\USER\Desktop\projectFile simulation\web\views\layouts\app.edge (web) - Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help

FOLDERS
web

67 </span>
68 </a>
69 </li>
70 <li class="list-inline-item">
71 <a href="#">
72 <span class="fa-stack fa-lg">
73 <i class="fa fa-circle fa-stack-2x"></i>
74 <i class="fa fa-facebook fa-stack-1x fa-inverse"></i>
75 </span>
76 </a>
77 </li>
78 <li class="list-inline-item">
79 <a href="#">
80 <span class="fa-stack fa-lg">
81 <i class="fa fa-circle fa-stack-2x"></i>
82 <i class="fa fa-github fa-stack-1x fa-inverse"></i>
83 </span>
84 </a>
85 </li>
86 </ul>
87
88 </div>
89 </div>
90 </div>
91 </footer>
92
93 <!-- Bootstrap core JavaScript -->
94 <script src="vendor/jquery/jquery.min.js"></script>
95 <script src="vendor/bootstrap/js/bootstrap.bundle.min.js"></script>
96 @section("scripts")
97
98 <!-- Custom scripts for this template -->
99 <script src="js/clean-blog.min.js"></script>
100 </body>
101
```

*Figure 20: Sublime Text windows; being used for AALRT web development*

## CHAPTER 5 RESULT AND DISCUSSION

### 5.1 Android Application

This thesis aims at providing real time information of AALRT train status using Android smartphones. The whole system is divided into four models; The android application, the website which has the information of the database for passengers which doesn't get access to an internet-connected android device, SMS information system that works for passenger which don't have internet connection on their phone and database which holds all of the information in a MYSQL database .

After successful implementation of AALRT train Tracking System the following results is obtained: Steps that should have to be taken in order to use this AALRT Passenger information application is discussed here below as follows:

Firstly, after AALRT users install this application on their android smartphone they have to activate this app by registering with their name, email and password.

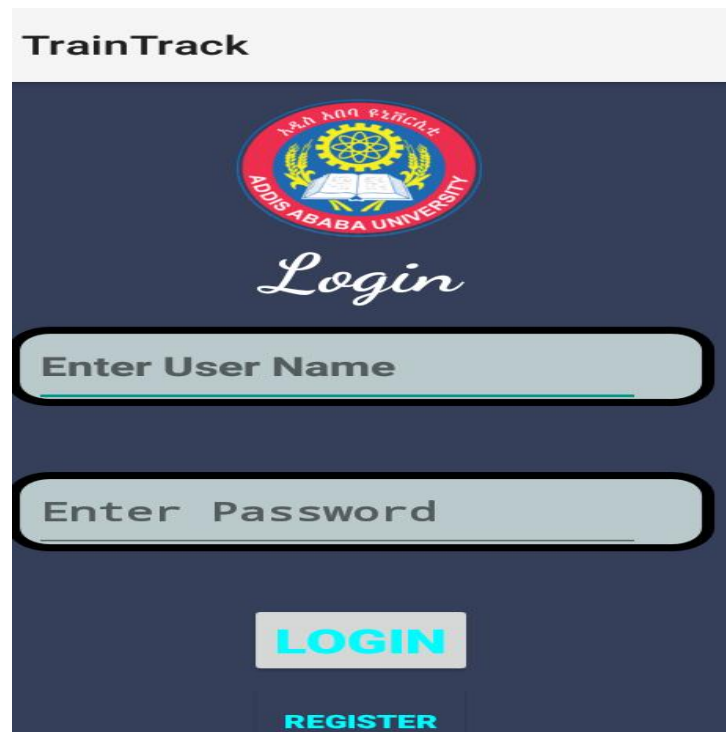
Step 1: New Users have to register using their Name, Phone Number and Password. New registration layout is shown in the figure below



*Figure 21: AALRT Application Registration layout*

## *Mobile Enabled passenger information system: case of AALRT*

Step 2: Next Log in using their Phone Number and Password. : At monitoring side, initially user needs to perform Login activity. Login page is shown in It provides Login interface to the user

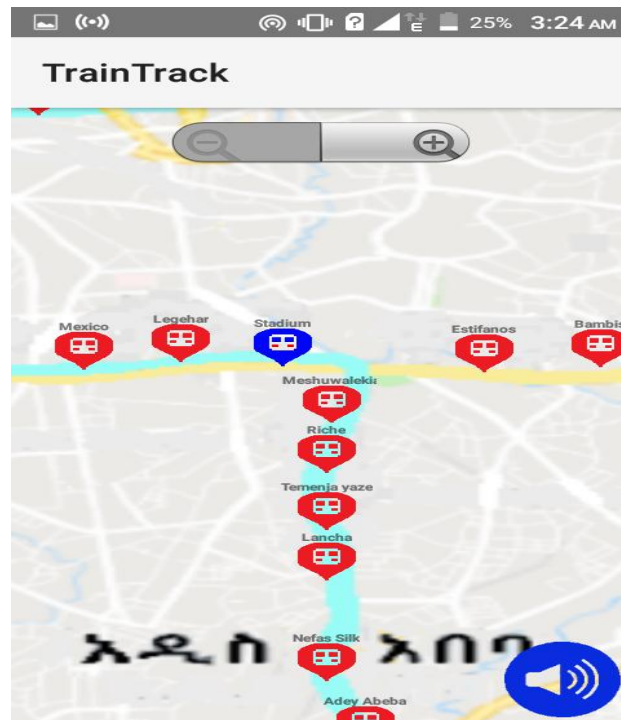


*Figure 22: AALRT Application Log in layout*

Step 3: When user will enter user name and password then system will do validation to check whether the entered username and password is correct or not. If the entered username or password is wrong then system gives an error message.

## *Mobile Enabled passenger information system: case of AALRT*

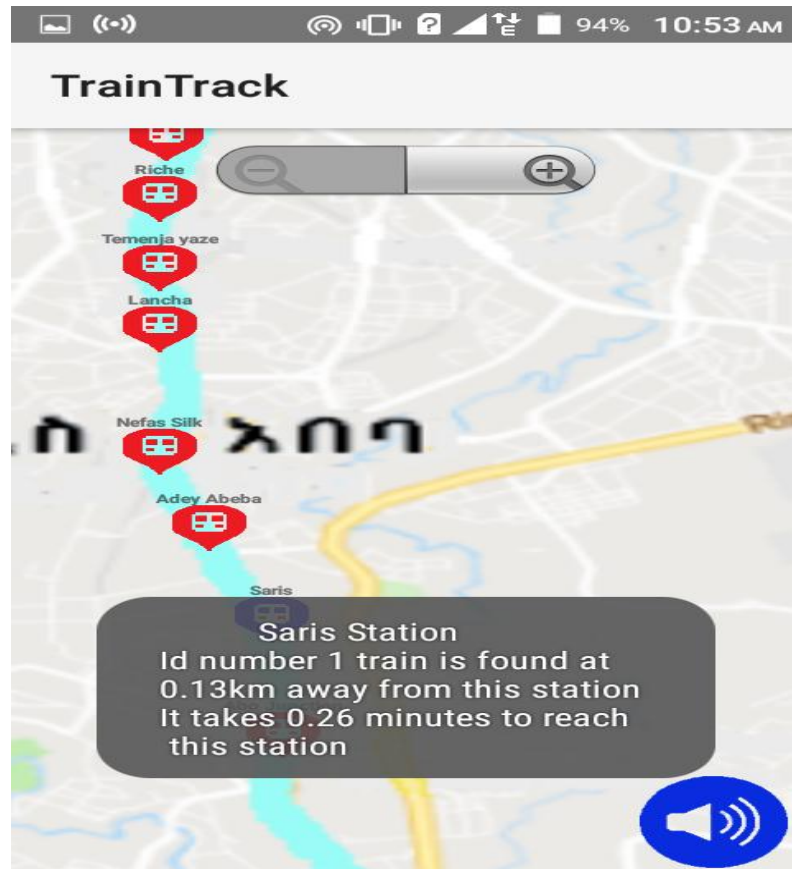
After the system checks the validity of entered user name and password, then user gets directed to next page with successful login.



*Figure 23: Screen capture of AALRT info app showing trains information*

Step 4: There are 39 icons on the map showing all stations residing along either side of the rail roads in the East-West and North-South direction. All icons are responsive and upon touching one can read the station names including the details of information coming to that station. The red icon shows the station located in East-West and corridor South-North corridor.

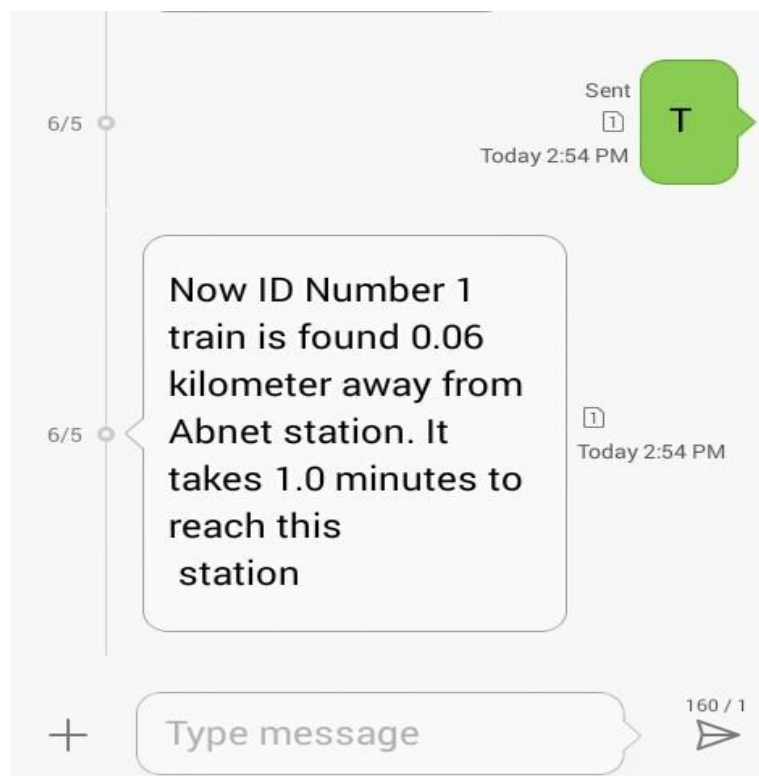
Simulation is done for one train that has ID name "T". The Icon will show blue color to the nearest station of the train where it is found. Upon touching of the icon, popup detailing on the next train with details of arrival time and kilometer will be shown to the users on their mobile screen. The system has benefit of hearing this train info by voice with easy access to the button which is very useful for blind people. Figure shows this detail



*Figure 24: Saris station icon popup detailing on the next train*

## 5.2 SMS

As I tried to describe before SMS has advantage facility being convenient way of delivering limited amount of information (no need for dialing and negotiating a radio link data connection)., With predetermined traffic network layout, the length of the SMS message is sufficient to include all data needed to deliver a request and its response in two small transactions. One Admin mobile phone number is taken as server which is connected to GSM modem to reply to each user's request. When the user sends SMS request containing train ID'T' to Admin number , immediately the system will send an SMS of exact train location and estimation time of arrival to the users. Figure 4.18shows the process.



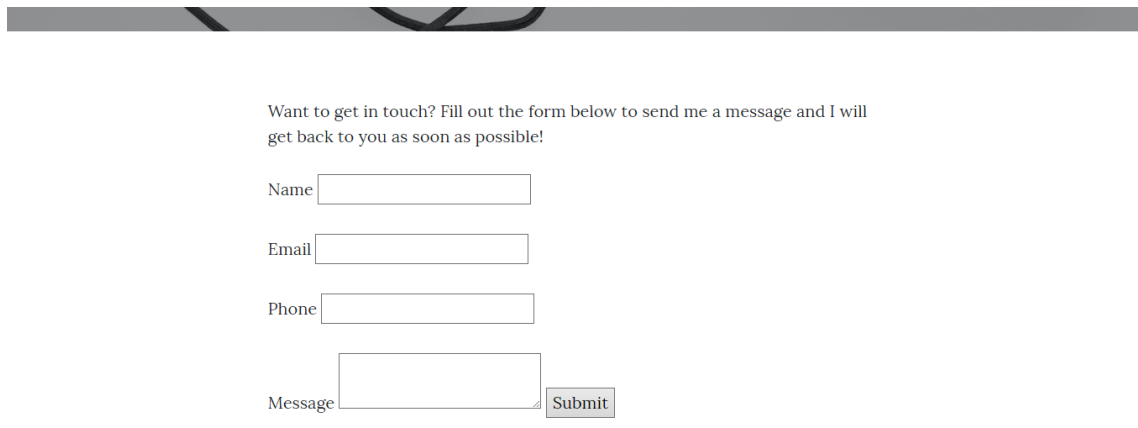
*Figure 25: capture of SMS delivery mode process*

### 5.3 Web

Web access has its own advantage of providing better communication between passengers and AALRT Company. some of the benefits are, it allows passengers to get more details of the service on the web page ,comment on the service, know the details of where about of the stations from the main road and also the company can release any update information about service such as price of ticket and etc. Below local web is designed on personal computer. Source code used to design this web is given in Appendix



*Figure 26: AALRT web*



*Figure 27: capture of comment giving space for AALRT passenger on the web page*

## **CHAPTER 6 CONCLUSION AND RECOMMENDATION**

### **6.1 Conclusion**

This thesis aim was to put away lack of passenger information system of AALRT. Information like of, real time location of the coming train and basic information about transit agency is not provided currently for the users. Implementation and description Mobile enabled Passenger Information system is stated. The PIS monitors the current location of all the AALRT train and estimates their arrival time at different stops in their respective routes. This computed arrival times are updated time to time as soon as every train sends new updates. It distributes all this information to passengers using Android application, SMS facility and Web that consist of the details information of AALRT service. This research serves the needs of passengers and administrators of the transit system. With the advent of GPS and the ubiquitous cellular network, GPS receiver that collects latitude and longitude value of the train is installed in every AALRT train, this value is then transferred to the server using GSM network, from there the server calculates the Estimation time of the train and respond to all passengers request using three delivery modes. Those delivery modes are Android app, SMS and web .

Simulation tracking is done for one train, the designed system shows position of this train on map in every 30 seconds as the value of location is changed there is easily accessible button on the app so that visually enable people can hear this information in audio, SMS that can be access using any mobile phones is also designed successfully. Web that has most of information about AALRT is designed so that new users and foreign people can get serviced information easily without wondering around. At end this PIS system has great contribution in such that passenger can make decision regarding the travel wait time based on provided information and better communication will be formed between passenger and the AALRT agency

## **6.2 Recommendation**

Presently there is no standard passenger information system for AALRT service. Getting enough information about the service is most important thing for better passenger satisfactory and reliable service. Train tracking system provides the passengers with the current location of the train and gives the details of the estimated arrival time of the train to the passenger. This system helps the passengers to make better travelling decision as passengers can decide if he/she should wait for the train or not in accordance to the estimated arrival time (ETA). From this system passenger can also determine if the train has been missed or yet to come. The system provides the basic information which is necessary to make day to day life more efficient. Therefore, it is recommended that AALRT should implement passenger information system as soon as possible.

For researchers who want study more of this passenger information area there is many improvements they can implement for example Sometimes the current GPS coordinates of the train does not match with the coordinates on the same route of the Google Map. In result of this when the train's current location is mapped on Google Map it might be shown on the road but near it. To overcome this problem reference points can be plotted along the route in corresponds to the Google Maps. With this GPS locations generated by the mobile devices are associated with its nearest reference point.

During design implementation of this thesis hardware implementation could have been developed but Because of time, budget, and lack of experience constraints, the PIS could not be designed wider than the one design on this thesis.

## REFERENCES

- [1] Corporation discloses/Addis light rail project detail”. The Ethiopian Herald 10 March 2013. Archived from the original, on 9 November 2013 Retrieved 25 October 2014
- [2] Asrat, Amare (10 November 2015). "Addis light rail Hayat - Torhailoch route begins operation”. Ethiopian News Agency Addis Ababa Archived from the original on 17 November 2015. Retrieved 2015-11-13.
- [3] [http://www.waltainfo.com/news/editors\\_pick/detail?cid=3760](http://www.waltainfo.com/news/editors_pick/detail?cid=3760)
- [4] Akpoyomare et al (2014). Factor influencing customer satisfaction in telecommunication
- [5] Design and implementing of Passenger Information System at stations of AALRT by Natan jima Addis Ababa University July 2017
- [6] Ovuezireie. D., Marinov. B., Dairo.P., & Iliana. G. (2014) Customer Satisfaction Factors for Light Rail: What can we learn from a successful case? vol 9
- [7] Vincent. M. (2008) Measurement valuation of public transport reliability
- [8] Nandan .S. (2010). Determinants of customer satisfaction on service quality: A study of railway platforms in India. Journal of public transportation, Vol 13 (1).
- [9] Van. O., & Van. N. (2010, March) the impacts of rail terminal design on transit service.
- [10] Brendan Kidwell, “Predicting Transit Vehicle Arrival Times”, GeoGraphics Laboratory, Bridgewater State College, August 2001.
- [11] <https://arxiv.org/pdf/1206.0447>
- [12] R. Manikandan, S. Niranjani, “Implementation on real time transportation information using GSM query response system,” Contemporary Engineering Sciences, Vol. 7, No.11, pp: 509-514, 2014
- [13] R.Maruthi, C.Jayakumari “SMS based Bus Tracking System using Open Source Technologies,” International Journal of Computer Applications (0975 – 8887) Volume

86 – No 9, January 2014 College of Engineering, Chennai SSN College of Engineering, Chennai

[14] Lam, Ka Ho. Bus Arrival Predictor on the Platform of Google Android. Hong Kong:s.n., 2008-2009

[15] R. Manikandan, S. Niranjani, “Implementation on real time transportation information using GSM query response system,” Contemporary Engineering Sciences, Vol. 7, No.11, pp: 509-514, 2014

[16] Sund, H., Thurfjell, F., 2009. Passenger information at the railway station

[17] <https://latitude.to/map/et/ethiopia/cities/addis-ababa>

[18] Micro System Center, <http://technet.microsoft.com/enus/library/cc181234.asp>

[19] Heinz, W., 2003. Passenger service times on trains: Theory, measurements and Mode

[20] Bornstein, Dan. Dalvik VM Internals. [Online] 2008  
<http://sites.google.com/site/io/dalvik-vm-internals>

[21] <http://d.android.com/reference/android/app/Activity.html>

## Appendix Source Code

### Source code for Main Activity of PIS App

```
package com.track.train;

import android.Manifest;
import android.app.Service;
import android.content.Context;
import android.content.DialogInterface;
import android.content.Intent;
import android.content.pm.PackageManager;
import android.location.Location;
import android.location.LocationListener;
import android.location.LocationManager;
import android.os.Bundle;
import android.os.IBinder;
import android.provider.Settings;
import android.widget.Toast;
import android.app.*;

public class LocationTrack extends Service implements LocationListener {

    private final Context mContext;
    boolean checkGPS = false;
    boolean checkNetwork = false;
    boolean canGetLocation = false;
    Location loc;
    double latitude;
    double longitude;
    private static final long MIN_DISTANCE_CHANGE_FOR_UPDATES = 10;
    private static final long MIN_TIME_BW_UPDATES = 1000 * 60 * 1;
    protected LocationManager locationManager;
    public LocationTrack(Context mContext) {
        this.mContext = mContext;
        getLocation();
    }
    private Location getLocation() {
        try
        {
            locationManager = (LocationManager) mContext
                .getSystemService(LOCATION_SERVICE);
            // get GPS status
            checkGPS = locationManager
                .isProviderEnabled(LocationManager.GPS_PROVIDER);
```

```
//          get          network          provider          status
checkNetwork          =          locationManager
.isProviderEnabled(LocationManager.NETWORK_PROVIDER);

if          (!checkGPS          &&          !checkNetwork)          {
    Toast.makeText(mContext, "No Service Provider is available",
Toast.LENGTH_SHORT).show();
}          else          {
    this.canGetLocation          =          true;

// if GPS Enabled get lat/long using GPS Services
if          (checkGPS)          {

//          if          (ActivityCompat.checkSelfPermission(mContext,
Manifest.permission.ACCESS_FINE_LOCATION)          !=
PackageManager.PERMISSION_GRANTED          &&
ActivityCompat.checkSelfPermission(mContext,
Manifest.permission.ACCESS_COARSE_LOCATION)          !=
PackageManager.PERMISSION_GRANTED)          {
//          TODO:          Consider          calling
//          ActivityCompat#requestPermissions
// here to request the missing permissions, and then overriding
// public void onRequestPermissionsResult(int requestCode, String[]
permissions,
//          int[] grantResults)
// to handle the case where the user grants the permission. See the
documentation
// for ActivityCompat#requestPermissions for more details.
//          }
    locationManager.requestLocationUpdates(
LocationManager.GPS_PROVIDER,
MIN_TIME_BW_UPDATES,
MIN_DISTANCE_CHANGE_FOR_UPDATES,          this);
    if          (locationManager          !=          null)          {
        loc          =          locationManager
.getLastKnownLocation(LocationManager.GPS_PROVIDER);
        if          (loc          !=          null)          {
            latitude          =          loc.getLatitude();
            longitude          =          loc.getLongitude();
        }
    }
}

if          (checkNetwork)          {
```

## Mobile Enabled passenger information system: case of AALRT

```
        /*if (ActivityCompat.checkSelfPermission(mContext,
Manifest.permission.ACCESS_FINE_LOCATION) !=
PackageManager.PERMISSION_GRANTED
&&
ActivityCompat.checkSelfPermission(mContext,
Manifest.permission.ACCESS_COARSE_LOCATION) !=
PackageManager.PERMISSION_GRANTED) {
// TODO: Consider calling
// ActivityCompat#requestPermissions
// here to request the missing permissions, and then overriding
// public void onRequestPermissionsResult(int requestCode, String[]
permissions,
// int[] grantResults)
// to handle the case where the user grants the permission. See the
documentation
// for ActivityCompat#requestPermissions for more details.
}*/
locationManager.requestLocationUpdates(
LocationManager.NETWORK_PROVIDER,
MIN_TIME_BW_UPDATES,
MIN_DISTANCE_CHANGE_FOR_UPDATES, this);

if (locationManager != null) {
loc = locationManager
.getLastKnownLocation(LocationManager.NETWORK_PROVIDER);
}

if (loc != null) {
latitude = loc.getLatitude();
longitude = loc.getLongitude();
}
}

} catch (Exception e) {
e.printStackTrace();
}

return loc;
}

public double getLongitude() {
if (loc != null) {
longitude = loc.getLongitude();
}
return longitude;
}
```

```
public double getLatitude() {
    if (loc != null) {
        latitude = loc.getLatitude();
    }
    return latitude;
}

public boolean canGetLocation() {
    return this.canGetLocation;
}

public void showSettingsAlert() {
    AlertDialog.Builder alertDialog = new AlertDialog.Builder(mContext);

    alertDialog.setTitle("GPS is not Enabled!");
    alertDialog.setMessage("Do you want to turn on GPS?");

    alertDialog.setPositiveButton("Yes", new DialogInterface.OnClickListener() {
        public void onClick(DialogInterface dialog, int which) {
            Intent intent = new
Intent(Settings.ACTION_LOCATION_SOURCE_SETTINGS);
            mContext.startActivity(intent);
        }
    });

    alertDialog.setNegativeButton("No", new DialogInterface.OnClickListener() {
        public void onClick(DialogInterface dialog, int which) {
            dialog.cancel();
        }
    });

    alertDialog.show();
}

public void stopListener() {
    if (locationManager != null) {
        /* if (ActivityCompat.checkSelfPermission(mContext,
Manifest.permission.ACCESS_FINE_LOCATION) !=
PackageManager.PERMISSION_GRANTED &&
ActivityCompat.checkSelfPermission(mContext,
Manifest.permission.ACCESS_COARSE_LOCATION) !=
```

## Mobile Enabled passenger information system: case of AALRT

```
PackageManager.PERMISSION_GRANTED) {
    //      TODO:      Consider      calling
    //      ActivityCompat#requestPermissions
    // here to request the missing permissions, and then overriding
    // public void onRequestPermissionsResult(int requestCode, String[]
permissions,
    //      int[] grantResults)
    // to handle the case where the user grants the permission. See the
documentation
    // for ActivityCompat#requestPermissions for more details.
    return;
}*/
locationManager.removeUpdates(LocationTrack.this);
}
}

@Override
public IBinder onBind(Intent intent) {
    return null;
}

@Override
public void onLocationChanged(Location location) {
}

@Override
public void onStatusChanged(String s, int i, Bundle bundle) {
}

@Override
public void onProviderEnabled(String s) {
}

@Override
public void onProviderDisabled(String s) {
}
}
```

**Source Code for Web app**

```
<!DOCTYPE html>

<html lang='en'>

<head>

  <meta charset='utf-8'>

  <meta name='viewport' content='width=device-width, initial-scale=1, shrink-to-fit=no'>

  <meta name='description' content=''>

  <meta name='author' content=''>

  <title>Mobile Enabled Passenger Information System</title>

  <!-- Bootstrap core CSS -->

  <link href='vendor/bootstrap/css/bootstrap.min.css' rel='stylesheet'>

  <!-- Custom fonts for this template -->

  <link href='vendor/font-awesome/css/font-awesome.min.css' rel='stylesheet' type='text/css'>

  <link href='https://fonts.googleapis.com/css?family=Lora:400,700,400italic,700italic' rel='stylesheet' type='text/css'>
```

```
<link
href='https://fonts.googleapis.com/css?family=Open+Sans:300italic,400italic,600italic,700italic,800italic,400,300,600,700,800'
rel='stylesheet' type='text/css'>

<!-- Custom styles for this template -->

<link href="css/clean-blog.min.css" rel="stylesheet">

</head>

<body>

<!-- Navigation -->

<nav class="navbar navbar-expand-lg navbar-light fixed-top" id="mainNav">

  <div class="container">

    <a class="navbar-brand" href="/">Mobile Enabled Passenger information
system</a>

    <button class="navbar-toggler navbar-toggler-right" type="button" data-
toggle="collapse" data-target="#navbarResponsive"

      aria-controls="navbarResponsive" aria-expanded="false" aria-label="Toggle
navigation">

      Menu

      <i class="fa fa-bars"></i>

    </button>

    <div class="collapse navbar-collapse" id="navbarResponsive">
```

```
<ul class="navbar-nav ml-auto">  
  <li class="nav-item">  
    <a class="nav-link" href="/">Home</a>  
  </li>  
  <li class="nav-item">  
    <a class="nav-link" href="/about">About</a>  
  </li>  
  <li class="nav-item">  
    <a class="nav-link" href="/post">Sample Post</a>  
  </li>  
  <li class="nav-item">  
    <a class="nav-link" href="/contact">Contact</a>  
  </li>  
</ul>  
</div>  
</div>  
</nav>  
  
@!section('content')  
  
<!-- Footer -->
```

```
<footer>
```

```
<div class="container">
```

```
<div class="row">
```

```
<div class="col-lg-8 col-md-10 mx-auto">
```

```
<ul class="list-inline text-center">
```

```
<li class="list-inline-item">
```

```
<a href="#">
```

```
<span class="fa-stack fa-lg">
```

```
<i class="fa fa-circle fa-stack-2x"></i>
```

```
<i class="fa fa-twitter fa-stack-1x fa-inverse"></i>
```

```
</span>
```

```
</a>
```

```
</li>
```

```
<li class="list-inline-item">
```

```
<a href="#">
```

```
<span class="fa-stack fa-lg">
```

```
<i class="fa fa-circle fa-stack-2x"></i>
```

```
<i class="fa fa-facebook fa-stack-1x fa-inverse"></i>
```

```
</span>
```

```
</a>
```

```
</li>
```

```
<li class="list-inline-item">
  <a href="#">
    <span class="fa-stack fa-lg">
      <i class="fa fa-circle fa-stack-2x"></i>
      <i class="fa fa-github fa-stack-1x fa-inverse"></i>
    </span>
  </a>
</li>
</ul>

</div>

</div>

</div>

</footer>

<!-- Bootstrap core JavaScript -->
<script src="vendor/jquery/jquery.min.js"></script>
<script src="vendor/bootstrap/js/bootstrap.bundle.min.js"></script>

@!section('scripts')

<!-- Custom scripts for this template -->
```

```
<script src="js/clean-blog.min.js"></script>
```

```
</body>
```

```
</html>
```