# Vehicle tracking using Raspberry Pi using GPS receiver, cell tower location and signal strength triangulation of cell tower locations and arrival time prediction

[1]Ashok Philip Verghese, [2]Joby John
[1]Mtech Student, [2]Associate Professor
Electronics and communication,
Mangalam College of Engineering, Kottayam, India

***Abstract*— This paper discusses vehicle tracking using raspberry pi as the vehicle component. The location of the vehicle is obtained by three different methods. One of the methods of obtaining the location is by GPS receiver. The second method is by cell tower location. The third method is by triangulation of nearby cell tower location based on the signal strength. In addition to this a method of comparing the path to be followed and the actual path and alert being sent in case of deviation from actual path is also implemented. There is also facility for estimating the time of arrival of the vehicle at a particular point based on the geodic distance and the average speed. All these facilities are very useful in any kind of vehicle tracking system like one implemented for a school bus. Also an intelligent alert message having the current location is made using text to speech converter library in python.**

***IndexTerms*— vehicle tracking; raspberry pi; python; flask; GSM module.**

## I. INTRODUCTION

This paper describes a vehicle tracking system. The vehicle tracking system shows the movement of a vehicle on a map. A normal vehicle tracking system consists of a vehicle component usually a GPS or location tracking system. It collects and sends geographical information to a server at periodic intervals. The server stores the geographical information. Then there is the display system usually a website or desktop graphical user interface which shows the location of the vehicle on the map. There are mainly two types of vehicle tracking systems .They are passive and active vehicle tracking systems. A passive vehicle tracking system does not show the location of the vehicle in real time where as an active system shows the location of the vehicle in real time. Our purposed system is almost real time and the location of the vehicle gets updated periodically so it is an active vehicle tracking system. This vehicle tracking system can be used in tracking a school bus which will be useful to the students and parents and staff for indentifying the location of the bus in real time so they do not wait for long time at the bus stop. Also prevents theft from occurring as the location of the vehicle can be monitored on the map in case of theft. There are provisions for monitoring the speed of the vehicle, temperature inside the vehicle and for sending alert in case of fire inside the vehicle. All these are useful for improving the ease of use of the school bus and for improving the comfort of the passengers. In our proposed system the location of the vehicle at any instant is provided in three different ways. One is using the latitude and longitude from the GPS receiver. The next one is using the cell tower location of the GSM module of the vehicle component. The third one is a triangulation method of nearby cell tower locations of the GSM module of the vehicle component based on the signal strength. In addition to these we have an algorithm to alert in case the vehicle goes off the specified path. The specified path is to be traced on a map and the information is transferred to the raspberry pi and stored there. During the actual journey the location is compared to the stored data and in case of large deviations a wrong path message is announced and also a message is sent to the vehicle owner regarding this with the current location co-ordinates. In the proposed system instead of using an online server for data storage we are using an online store for data. So this system can be considered an IoT system. A method of prediction of time of arrival of the vehicle based on the geodic distance between the points and the average current speed also has been made. The alert messages in case of wrong path are made by using text to speech python libraries.

## II. MAIN COMPONENTS OF THE SYSTEM

*Raspberry PI*

The Raspberry Pi is a series of credit card–sized single-board computers developed in Wales, United Kingdom by the Raspberry Pi Foundation with the intention of promoting the teaching of basic computer science in schools and developing countries.

*Fig 1. Raspberry Pi 2 Model B+*

*GSM Module*

GSM/GPRS Modem-RS232 is built with Dual Band GSM/GPRS engine- SIM900A, works on frequencies 900/ 1800 MHz. The baud rate is configurable from 9600-115200 through AT command. The GSM/GPRS Modem is having internal TCP/IP stack to enable you to connect with internet via GPRS. It is suitable for SMS, Voice as well as DATA transfer application in M2M interface. The onboard Regulated Power supply allows you to connect wide range unregulated power supply. Using this modem ,you can make audio calls, SMS, Read SMS, attend the incoming calls and internet through simple AT commands.

General features

- Dual-Band 900/ 1800 MHz
- GPRS multi-slot class 10/8
- GPRS mobile station class B
- Dimensions: 24*24*3 mm
- Weight: 3.4g
- Control via AT commands (GSM 07.07 ,07.05 and SIMCOM enhanced AT Commands)
- Operation temperature: -40°C to +85 °C

*Fig 2 SIM900A GSM /GPRS module*

*GPS RECEIVER*

TF-172 is a complete GPS receiver. Built-in satellite receiver antenna, satellite receiver core is using the most advanced U BLOX positioning core, provide highly accurate positioning information, therefore it is not only can meet personal use needs, also can meet the strict requirement of industrial positioning. Wide scope of application, such as automobile navigation, security system, map making, all kinds of investigation, agricultural use, etc. Built-in button battery for storing satellite data, such as the status of satellite signal, the last positioning location and time, to speed up first time positioning when device is switched on.


*Fig 3 Ublox gps receiver*

*Ultra-Mini Bluetooth CSR 4.0 USB Dongle Adapter*
Model BT-401
Bluetooth Standard V4.0
USB Standard USB 2.0
Operation Range 20 m
Data Transmission Rate +6dBm
Sensitivity -86dBm
Dimensions: 0.91 in x 0.55 in x 0.20 in (2.3 cm x 1.4 cm x 0.5 cm)
Weight: 0.18 oz (5 g)


*Fig 4 Bluetooth Dongle*

*Sensor:*

a) Temperature Sensor:
Specifications:
- Unique 1 wire interface requires only one port pin for communication.
- Each device has unique 64 bit serial code stored in an On Board ROM.
- Requires no External components.
- Can be powered from Data line; power supply range is 3.0v to 5.5v.
- Measures temperature from -55 to +125 Degree C.
- User definable non volatile (NV) alarm settings.
- Thermometer resolution is user selectable

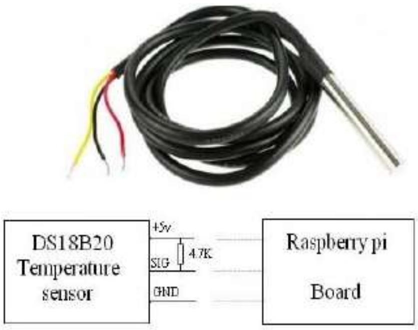


*Fig 5 The temperature sensor DS18B20*

b) Smoke or fire sensor:
Specifications:
• High sensitive to LPG Gas, iso-butane, propane.
• Small sensitive to alcohol, smoke.
• Fast response.
• Stable and long life.
• Simple drive circuit.


*Fig 6 The smoke sensor MQ6*

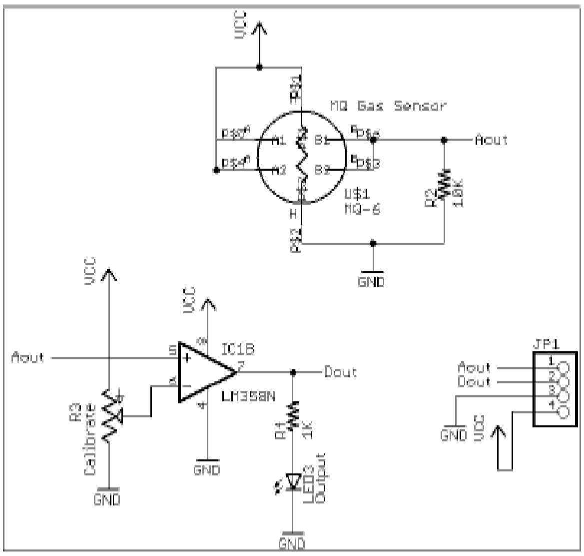


*Fig 7 Circuit diagram of Smoke sensor MQ6*

## III. DESIGN OF THE SYSTEM

The below diagram shows the basic design of the system. It consists of basically the vehicle component, the online data store and the website. The website can also be accessed on a Smartphone by an app that accesses the website. The app has been tested using android studio emulator.

The vehicle component consists of the raspberry pi which is connected to the GPS/GLONASS module via USB port. Using the gpsd process we get the latitude and longitude of the vehicle at regular intervals and the python scripts transmits these to the online data store via GPRS of the GSM module. The GSM module is also connected to another USB port and is accessed by python script and the required AT commands are given to control the GSM module via the python script. The vehicle owners component consist of the browser or smartphone app which requests the websever of the required information and the locations are showed on a map.

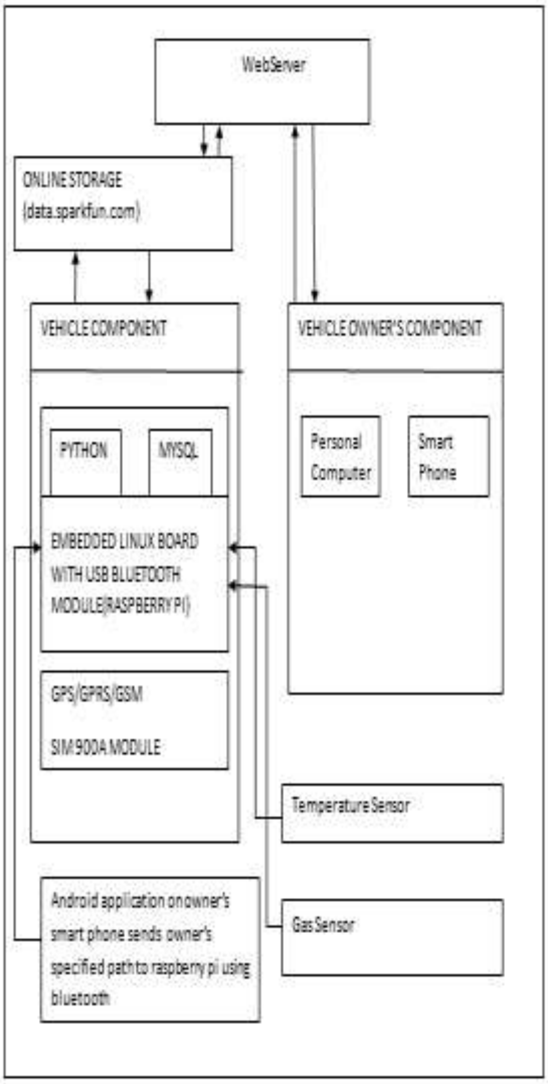


*Fig 8 Block diagram of the system*

The above is an exsiting system design based on the first reference paper[1].The only change made to this is the usage of an online storage instead of a online server.Here we have use a free online datastore stream on data.sparkfun.com.This eliminates the requirement of a public ip online server incase we install the website on our own pc.In case we need to access the website through the internet and we do not have the website installed in our machine then only we need a public server to host the website.

In addition to the above we are showing the location based on two other ways which do not depend on the GPS/GLONASS module.So in case we do not get fix of the location on the GPS servers we still have the location based on the other two methods.The other two methods employed for this are mentioned below

*Wrong path algorithm*

The specified path to be followed by the vehicle is traced by the vehicle owner using google maps. Then he needs to export it into kml. The kml files can be uploaded in the website from where they will be transferred to the raspberry pi. A python script in raspberry pi moves the latitudes and longitudes in the kml file to a table. During the actual journey the actual latitudes and longitudes are compared with the latitudes and longitudes in the raspberry pi table and if the difference is greater than a particular value then an alert message and sms is sent.

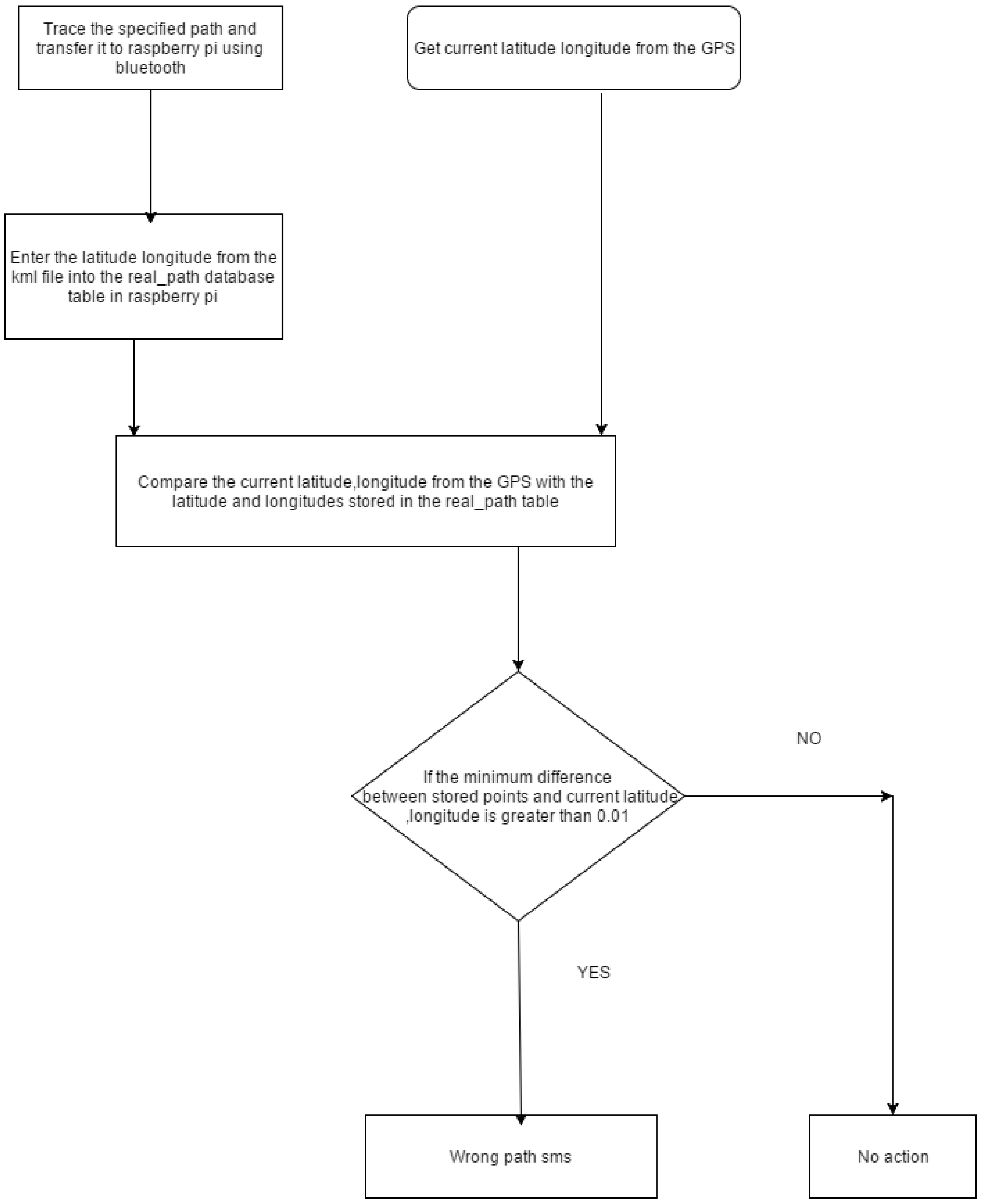


*Fig 9 Flow chart of wrong path algorithm*

*Finding the latitude and longitude based on cell tower location*

In this method the latitude and longitude are obtained by AT command "AT+CREG?" which gives the lacid and cellid to which the gsm module sim is registered currently. The lacid and cellid are send to the online datastore using gprs. From the online storage the lacid and cellid are reterived by the website (vehicle owners component).Then the lacid and cellid are converted into latitude and longitude using the google maps api. And using these latitude and longitudes the locations are mapped on the maps.

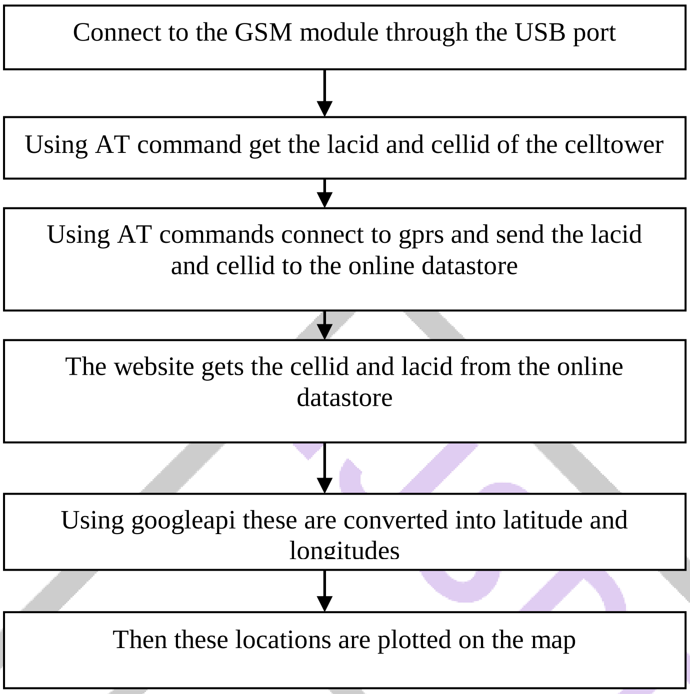


*Fig 10 Flow chart of finding location based on cell tower*

*Finding the latitude and longitude based on triangulation of the three nearest cell tower locations based on signal strength*

In this method the latitude and longitude are obtained based on a triangulation method based on the three nearest cell tower locations and their signal strengths. The nearest signal towers are obtained from the AT command AT+CENG=1,1 followed by AT+CENG?.This cell tower information and corresponding signal strengths are sent to the online datastore by gprs. The website retrieves this information and finds the corresponding latitude, longitudes of the nearest three cell tower locations. Then the final location is obtained by triangulation method involving taking the weighted average of the latitude and longitudes based on their respective signal strengths.

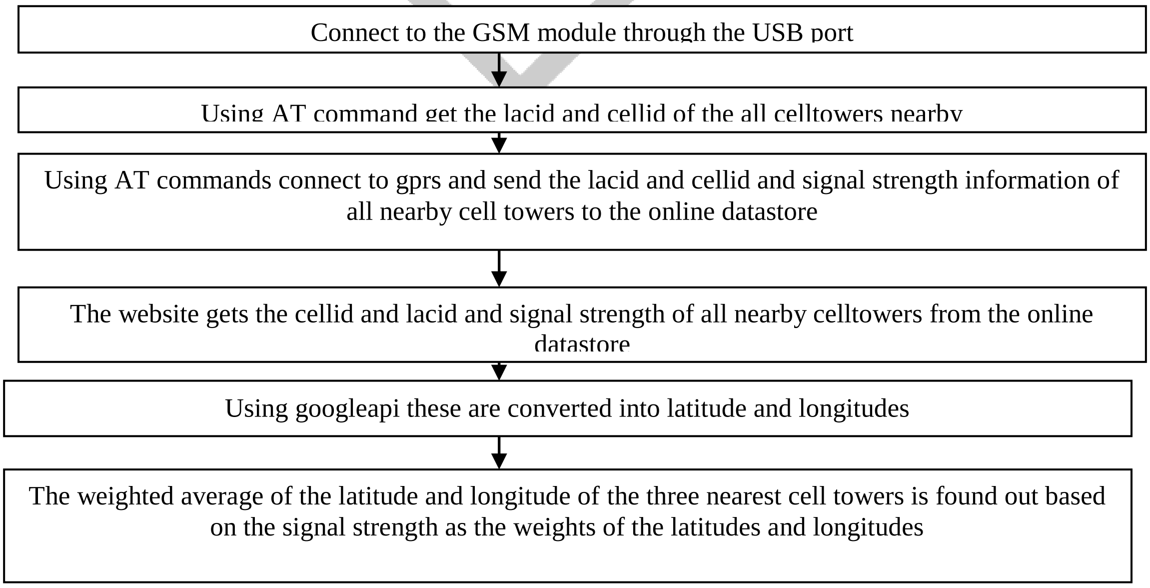


*Fig 11 Flow chart of location based on signal strength and cell tower location*

*Prediction of arrival time at a particular location*

A prediction of the arrival time at a particular location is made by using the average speed of the vehicle. The geodic distance between the current latitude and longitude and the target location latitude and longitude is calculated using the api and the time required to cover that distance is measured using the average speed of travel.

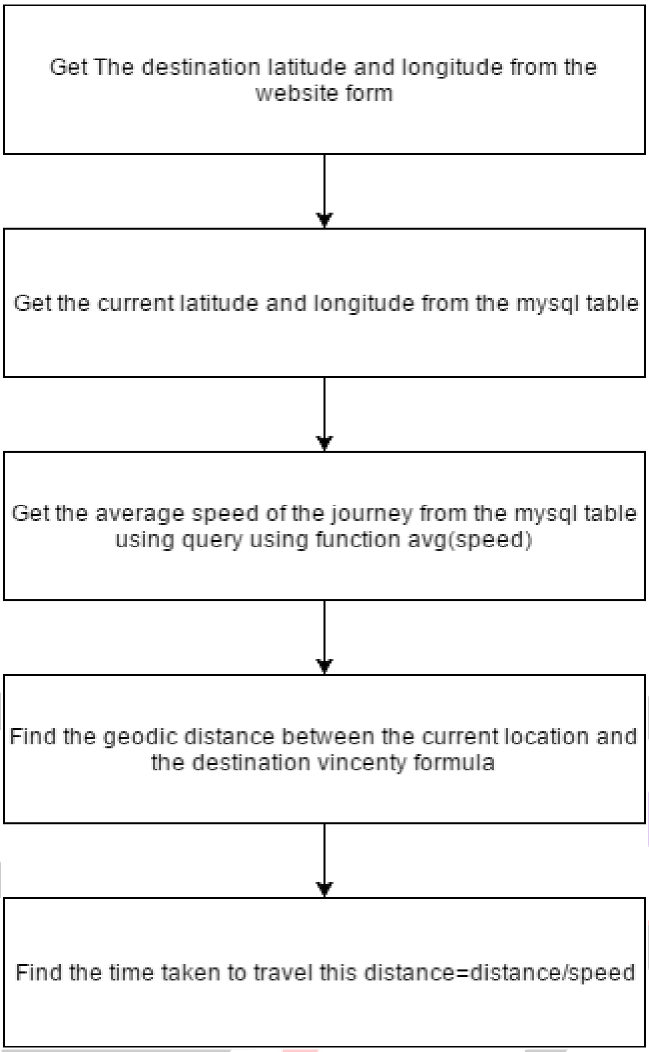


*Fig 12 Flow chart of prediction of time required*

## IV. TOOL AND LANGUAGES USED

*Flask(web Framework)*

Flask is a micro web application framework written in Python and based on the Werkzeug toolkit and Jinja2 template engine. It is BSD licensed. As of 2015, the latest stable version of Flask is 0.10.1.Examples of applications that make use of the Flask framework are Pinterest, LinkedIn, as well as the community web page for Flask itself. Flask is called a micro framework because it does not presume or force a developer to use a particular tool or library.

*Mysql*

MySQL is an open-source relational database management system (RDBMS);in July 2013, it was the world's second most widely used RDBMS, and the most widely used open-source client–server model RDBMS Python

*Python*

Python is a widely used general-purpose, high-level programming language. Its design philosophy emphasizes code readability, and its syntax allows programmers to express concepts in fewer lines of code than would be possible in languages such as C++ or Java. The language provides constructs intended to enable clear programs on both a small and large scale. Python supports multiple programming paradigms, including object-oriented, imperative and functional programming or procedural styles. It features a dynamic type system and automatic memory management and has a large and comprehensive standard library

*Android studio*

Android Studio provides the fastest tools for building apps on every type of Android device. World-class code editing, debugging, performance tooling, a flexible build system, and an instant build/deploy system all allow you to focus on building unique and high quality apps.

## V. Results

This the login screen for the website which shows the tracking of the vehicle on the map. There are two types of login Admin and Non Admin as mentioned earlier


*Fig 13Login screen*

When the admin user logs in with his username and password this page is presented.The admin user will be able to track the vehicle and also use the admin facilities of uploading the kml file which gives the actual path to be followed by the vehicle and also he has the facility for authorising the signed up non admin users.


*Fig 14 Screen after logging in for Admin user*

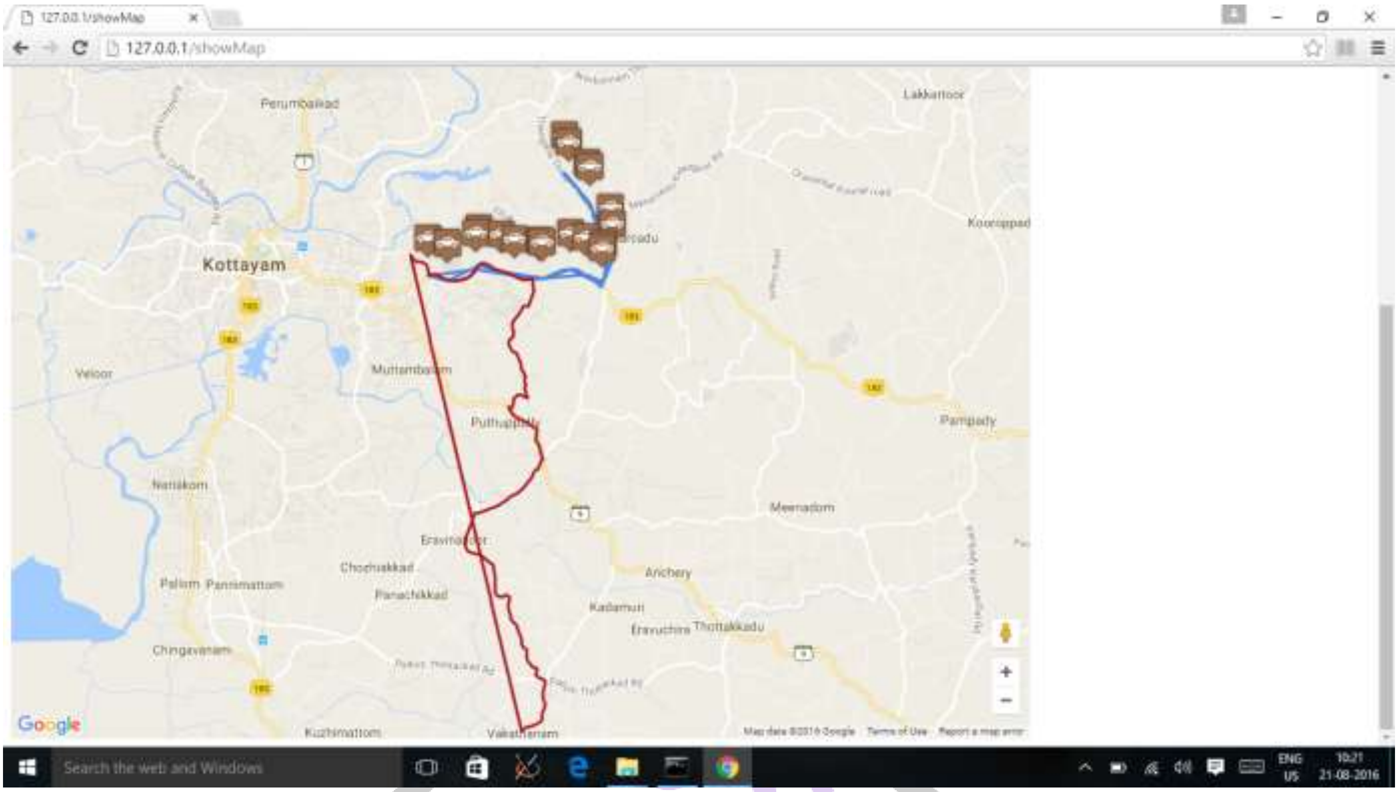


*Fig 15 Map showing vehicle location using GPS*

The above figure shows the path followed by the vehicle using the car icon and blue lines. The specified path to be followed is given by the red line. The tracking done using the other two methods of tracking location as mentioned above are also given by two other maps. So if location is not available via GPS we can check the location based on the other two methods.

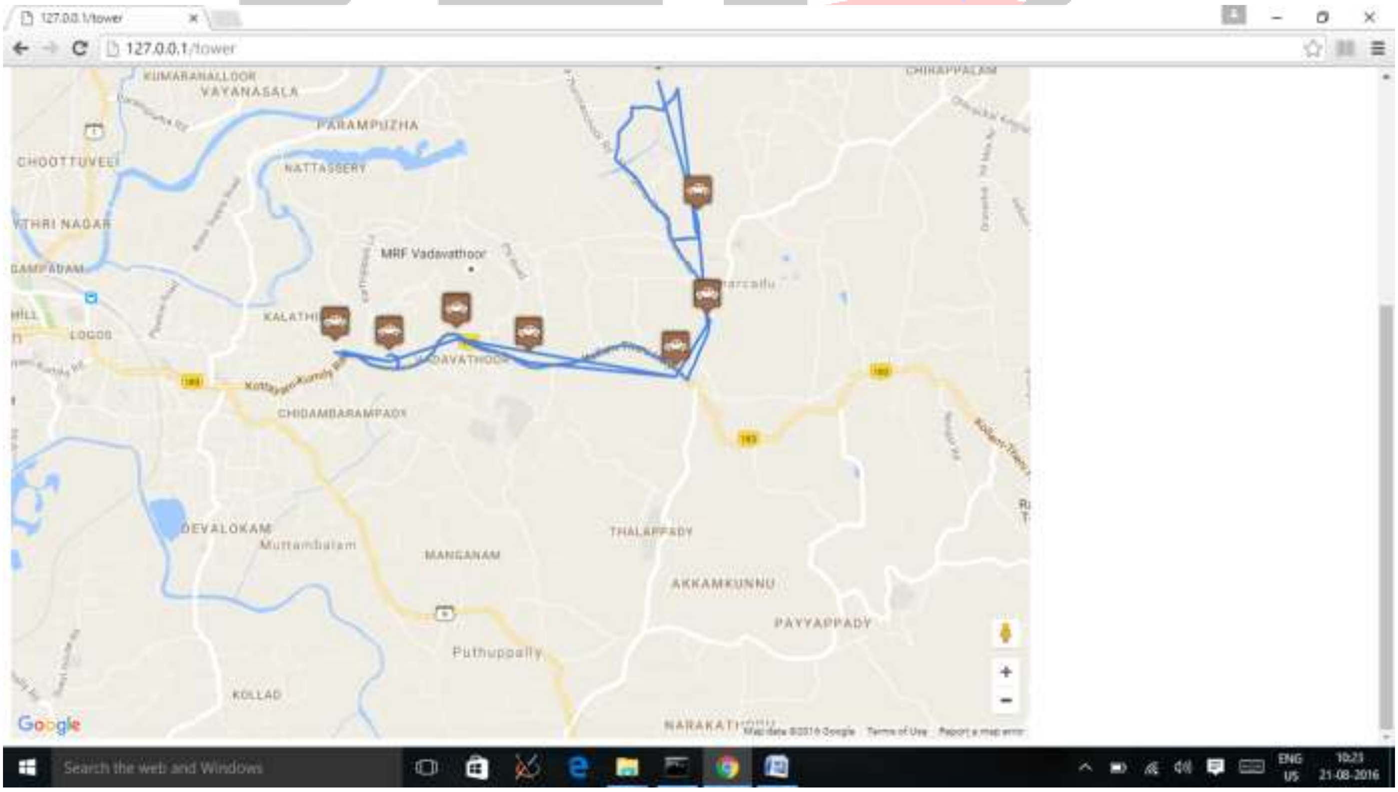


*Fig 16 Map showing vehicle location using cell tower location*

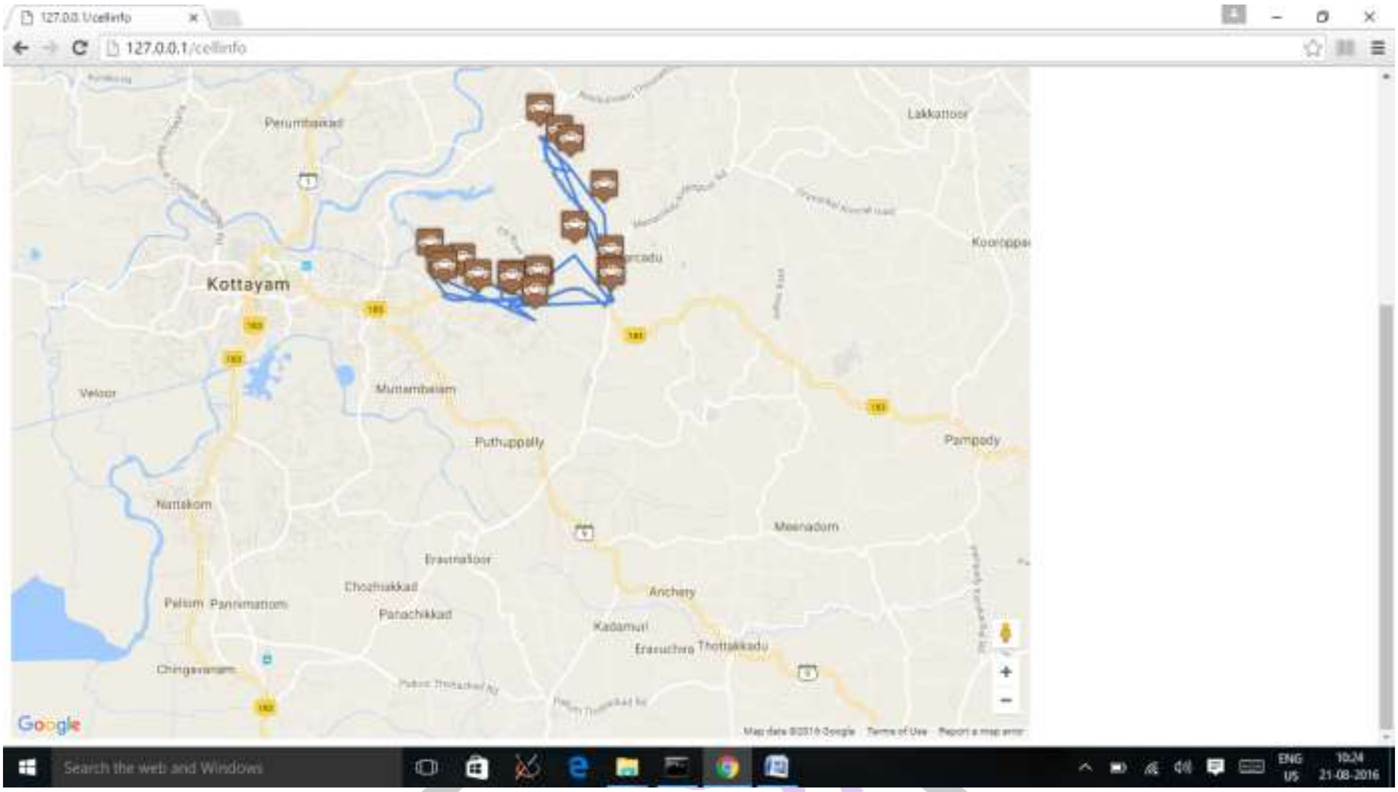


*Fig 17 Map showing vehicle location using cell tower triangulation*

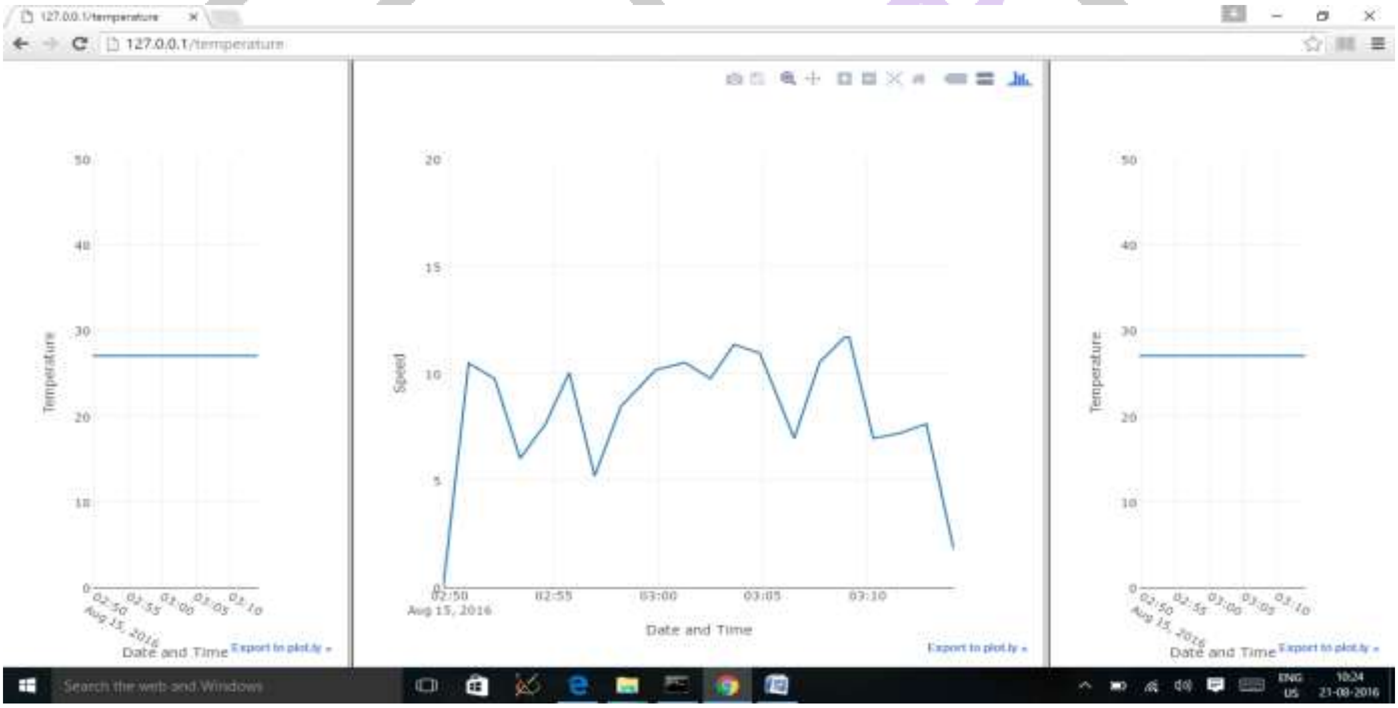


*Fig 18 Graph of variation in temperature and speed*


*Fig 19 form to get latitude and longitude*


*Fig 20 prediction of time required to reach particular destination*


*Fig 21 App showing website for vehicle tracking in android emulator*

A simple app was developed and tested on android studio emulator and above shows the screenshot of the same.

## VI. CONCLUSION

The proposed system used raspberry pi for a website based vehicle tracking system that can be used for all type of vehicles. Ease of travel for the travelers could be enhanced by locating the current position of the vehicle and safety of the travel could be enhanced by the fire and temperature alerts. The proposed system plays an important role in real time tracking and monitoring of vehicle by updating vehicle real time information on the server side after certain interval of time in order to monitor vehicle continuously. Whenever driver drives vehicle on the wrong path or in case of vehicle's accident situation occurs, the proposed system provides the vehicle's current location, speed to the vehicle owner's mobile. Hence this benefits to track the vehicle as early as possible. Student's safety mechanism also gets provided using temperature and smoke sensors. Two other methods of tracking the vehicle using cell tower location and triangulation of nearby cell tower locations was also provided in case of issue of not getting location information from the GPS receiver. Prediction of arrival time at a particular location is also provided based on the average speed and geodic distance. Also by using the online datastore we can we avoid the need of a public ip server and can do vehicle tracking local system on which the website and database are installed.

## REFERENCES

[1] Shinde, P.A. ; Mane, Y.B." Real time vehicle monitoring and tracking system based on embedded Linux board and android application", Circuit, Power and Computing Technologies (ICCPCT), 2015 International Conference on 19-20 March 2015.

[2] Tarapiah, S.; Atalla, S.; Alsayid, B., "Smart on-board transportation management system Geo-Casting featured," Computer Applications and Information Systems (WCCAIS), 2014 World Congress on , vol., no.,pp.1,6, 17-19 Jan. 2014.

[3] Kumar, R.; Kumar, H., "Availability and handling of data received through GPS device: In tracking a vehicle," Advance Computing Conference (IACC), 2014 IEEE International, vol., no., pp.245, 249, 21-22 Feb. 2014.

[4] SeokJu Lee; Tewolde, G.; Jaerock Kwon, "Design and implementation of vehicle tracking system using GPS/GSM/GPRS technology and smartphone application," Internet of Things (WF-IoT), 2014 IEEE World Forum on , vol., no., pp.353,358, 6-8 March 2014.

[5] Pengfei Zhou; Yuanqing Zheng; Mo Li, "How Long to Wait? Predicting Bus Arrival Time with Mobile Phone Based Participatory Sensing,"Mobile Computing, IEEE Transactions on, vol.13, no.6, pp.1228, 1241,June 2014.

[6] Liu; Anqi Zhang; Shaojun Li, "Vehicle anti-theft tracking system based on Internet of things," Vehicular Electronics and Safety (ICVES), 2013 IEEE International Conference on, vol., no., pp.48, 52, 28-30 July 2013.