IoT Based 3-Dimensional GPS Tracking System

Shreyans Patel¹, Ghansyam Rathod²

¹²EC Department, Birla Vishvakarma Mahavidyalaya Engineering College Anand, Gujarat, India

e-mail: ¹ shreyans_contact@gmail.com
e-mail: ² ghansyam.rathod@bvmengineering.ac.in

Abstract. This study solves the problem of identifying the exact location of the device in multi-story buildings by displaying the latitude, longitude and even the altitude data of the device with the use of a simple prototype. By simple mapping of buildings, this prototype can be programmed to give altitude output as the floor of the building on which the device is located. This paper addresses some opportunities brought by smart tracking methods. The prototype aims to locate the device and send its geolocation coordinates via email to the user so that they can recover their precious device or use the system for tracking applications. This study is based on the Internet of Things (IoT) so that the prototype can be used for locating any Android or iOS device connected to the internet with a GPS sensor available, directly through communication between the device and the controller through a Web Server or Cloud. In addition to the services of retrieving the device location that is already provided by many companies, the innovation done is that the prototype can also detect the altitude at which the object is located. This makes it very easy to find in multi-story buildings where just latitude and longitude data is not enough to find it quickly.

Keywords: IoT (Internet of Things), Geo-location, GPS (Global Positioning System), Android, iOS, altitude

1. INTRODUCTION

The method of figuring out the exact location of something is known as the Global Positioning System (GPS). Nowadays, this method finds itself very useful in various applications like having a tracking system integrated into a vehicle, cell phone, and many other distinct GPS enabled devices, those which are either stationary or mobile elements. The GPS functions by providing latitude, longitude, speed and altitude information which tells the exact location of the device that is being tracked. Hence, these parameters can also help in tracking the motion of a vehicle or a moving person. Therefore, for an example, such a tracking system integrated with GPS can be used by many companies to keep a track of the delivery progress as well as keep an eye on the route to monitor valuable items in transit at any time needed. Such a system for tracking vehicles uses Global Positioning System (GPS) [1] to detect the position. The General Packet Radio System (GPRS) [2] is profoundly used for transmitting the valuable data and a service provider like Google Maps which is based on the Google Earth software for showing the device being tracked to the user by locating it on the map [3]. Tracking systems like this can also be very helpful for some smart transportation systems like the Intelligent Transportation System (ITS) [4]. Parents can also monitor the location of their child by using such tracking systems.
Communication technology has made a significant development ever since it surpassed its ability to allow others to communicate with one another when they are in motion. Nowadays, cell phones are turning very advanced and provide more features than just carrying a conversation. Mobile phone GPS tracking is one of these features.

With the integration of GPS technology in new smart phones being very common these days, we can track the location of anyone carrying a GPS enabled mobile phone accurately and at any time. Tracking cell phones using GPS technology, therefore, is a very helpful feature for entrepreneurs, colleagues, friends, and parents to get monitoring facilities according to their needs.

The system that we have suggested in this paper provides us with a tracking system using a user and attendant model when a request is made by the user. The user is an embedded system that is interfaced with a switch that is used to make a tracking request whenever needed. An IoT platform called Blynk which comprises of a program based on a Blynk server to receive the location information, that is later converted into a mail format, one that can display the data as well as provide the end-user with a link to locate the device by using Google Maps serves as the attendant in the model. First, we have talked about our study to detect exact location of a GPS enabled device and later we have implemented our study to develop a prototype.

In the remaining parts of the paper, we have explained the technology involved in the prototype in Section II. We have described the prototype’s system planning and explained its implementation. Section IV displays the prototype for the system along with some screenshot of our projected work. Section V gives our conclusion and suggestions for future implementation. It also shows a study of different similar systems and their accuracies to compare them with our prototype.

2. Background

Global Positioning System (GPS) Technology

The GPS or the Global Positioning System is a technology that integrates a network of 24 satellites that belong to the United States of America (USA). These satellites were earlier used for armed services and later were allowed to be used for commercial use. The system works by emitting radio signals of pulses that are very short, to the GPS enabled retrievers at regular intervals. To compute the two-dimensional (latitude and longitude) position of the device, a triangulation technique is used which uses the signal from at least three satellites for calculation. To compute the three-dimensional (latitude, longitude, and altitude) position, at least four satellites are used. The average speed and direction of travel can be calculated after the location is computed. Hence, the Global Positioning System (GPS) is a significant technology for tracking the position of a device [3].

General Packet Radio System (GPRS) Technology

The development of GSM networks to enhance and support existing GSM networks leads to the creation of General Packet Radio Service (GPRS). Furthermore, it not only provides data services like the Short Messaging Service (SMS) along with the Circuit Switched Data (CSD) for faxes, but it also provides switch data services such as email and web browsers. During the transmission of each frame, the GPRS works on a prevailing GSM lattice infrastructure by with the accessible time slots are used. Therefore, the existing GSM link traffic is not overburdened and the data services are delivered efficiently. We can get a peak data transfer rate of 115.2 kbps in addition to eight accessible slots per frame by using the GPRS technology. Because of the large average service area of the GSM network structure worldwide, GPRS is the prime data service system available which is always operating. This makes it better suited for a real-time tracing management system [3].

Google Maps and Google Earth Software

Google Earth software is one of the most commonly used software available for free, that offers maps using satellite-captured images worldwide [5]. Google Earth has a version of its own that displays the maps online with the help of a browser and a web server. It is popularly known as Google Maps. The objects in the program are displayed with the use of various plug-ins offered to the community. An example of such kind of objects can be 3-Dimensional objects like skyscrapers which can be created using drafting software. The objects can be pin objects to display a part on a map called
the Point of Interest (POI) and we can display a path or a road by using the line objects [3]. To display such objects on a map, Google Earth uses a programming language of its own called the KML or the Keyhole Markup Language [6]. KML is a customized Markup Language (XML) with addons and it is used to define how we render the objects for the map. These objects based on the Keyhole Markup Language (KML), can be used with the Maps app provided by Google for Android/iOS or simply Google Maps to display the line and pin objects. For the system projected by us, we have used Google Maps as our choice to display the location of the device worldwide.

Applications of the Idea

There can be numerous applications of the idea proposed in the paper. One of the applications of such a system is proposed by the author of [7] which demonstrates an application to track running for fitness and data. The system enables the user to see both the location on the map as well as the altitude at which the user was moving as shown in Fig. 1. This can be very helpful for people concerned about tracking their fitness data and history. Another marvellous application is given by the authors of [3] where they have designed a system using the same concept for Real-Time Tracking application. This system can be very useful to business owners to check the status while the products are in transit. This helps in tracking information like the speed of the truck, location, driver status and many more. Fig. 2 and Fig. 3 demonstrates how the user can use the system and see the data like track trail and real-time tracking data respectively.

Figure 1. Visualized Running Track (left) and Altitude Differences (right) [7]

Figure 2. The tracking trail of Google Map-Server [3]
3. Projected System and Implementation

The prototype that we have designed is an open-source software-based embedded system. The hardware which is used to build the prototype consists of easy to find components that are very economical for building prototypes [3]. Our prototype can be categorized into a collection comprising three major units, a Location Requesting Device, a Handling Server and the Device to be tracked. The device to be tracked can be any Android/iOS device that supports GPS and GPRS technology so that it can transmit the location information to the Blynk server over a GSM network or the internet. The Handling Server is a service handler that receives the GPS data of the device to be tracked and it sends the data to the user via an email when a tracking request is made by the user from the Location Requesting Device. The Handling Server manipulates the GPS data from the device in a special form that can display the device location on Google Maps and the floor where the device is located is mentioned in the email. The block diagram of the projected system’s functioning is given in Fig. 4 and it gives an overview of the major processes involved. The inbuilt GPS sensor of the smart devices can measure latitude, longitude, altitude and speed. We use this data for the system.

Location Requesting Device

The Location Requesting Device is built using NodeMCU. NodeMCU is an open-source Internet of Things (IoT) Platform. A NodeMCU controller based on ESP-12 module hardware which includes a firmware that runs on the SoC produced by Espressif Systems called ESP8266 Wi-Fi is used to build this device. Generally, the term “NodeMCU” is used for the firmware instead of the development kits that are used. A Lua scripting language is used by the firmware for the device operation.

Handling Server

The handling server is based on a popular open-source IoT platform Blynk [8] which comprises free cloud and iOS and Android Mobile Apps. In this prototype, this is the handling server that is used to retrieve GPS data from the device to be sent to the user on request. The device to be tracked must have Blynk Mobile Application installed. The device must have access to the internet for the Blynk application to be able to send its GPS data on its server. The response time of the email is within a maximum of 5 seconds after the request is made.

Device to be Tracked

The device that is to be tracked by the user can be any Android or iOS-based device which has a GPS sensor built-in and Blynk app installed for providing tracking functionality. This built-in sensor is the same one which is used for the tracking purpose. This allows us to use the system without incorporating a dedicated GPS sensor. This also reduces the cost of the system significantly and makes it very economical. Since all the smart devices nowadays have these features by default, this system...
can be implemented easily for all the existing users. Fig. 5 shows the flow chart of the system to explain its working process. If the internet fails during operation, the system would stop working. In such a scenario, the system will be back online as soon as the device gets connected to the internet. Being able to track the device is as difficult as it is in all the other available tracking services.

4. Prototype of the System

The design projected by us in Section III is further explained in this section with its implementation shown. The prototype of our Location Requesting Device is shown on the right-hand side of Fig. 6 and the device to be tracked on its left. Fig. 7 shows the screenshot of the main screen of the software used for the demonstration. Fig. 8 shows the prototype operating when the request button is pressed, to demonstrate its working, we have added a virtual LED module in the Blynk app to make it easy for the user to see. Fig. 9 displays the screenshot of the prototype under demonstration confirming the user on sending the mail. Fig. 10 shows the mail sent by the server when the system starts up to notify the user. Fig. 11 displays the screenshot of the mail sent by the server after pressing the request button. It shows the Latitude, Longitude, and Floor where the device is located. It has also provided a link to Google Maps to show the device on the map which is shown in the screenshot in Fig. 12.

![Flowchart of the System](image)

**Figure 5.** Flowchart of the System

![GPS Device Tracking Module and Device to be tracked](image)

**Figure 6.** GPS Device Tracking Module and Device to be tracked
Figure 7. Screenshot of the Software Main Screen

Figure 8. Prototype Operation on Pressing Request Button

Figure 9. System giving Email Sent Notification to the User
5. Future Work

In this paper, we have proposed an embedded system that uses open-source software. The hardware which is used to build the prototype consists of easy to find components that are very economical for building prototypes. This projected system gives an idea to bring an innovative change in the fields of device tracking by adding altitude tracking features. It can be used for multi-story buildings where the user might lose their device while working and changing floors throughout the day.

We also observed inaccuracies in the operation of GPS sensors in low-cost mobile phones which compromises the system reliability. We have also used the inaccuracies data of some standard devices given in [7] to unify a reading. This data is given in Table 1. Fig. 13 demonstrates the graphical representation of Elevation Inaccuracies. Now we will discuss the accuracy of our prototype to draw a line between its performance and the performance of these standard devices presented in our study.

| Application              | Total ascent in meters | Total descent in meters | Total deviation in meters | Rank |
|--------------------------|------------------------|-------------------------|---------------------------|------|
| Adidas miCoach           | 5                      | 5                       | 10                        | 4    |
| Endomondo                | 10                     | 23                      | 33                        | 8    |
| Noom Cardio Trainer      | 0                      | 0                       | 0                         | 1    |
| Actual Reading | Device Average Output | Actual Reading | Device Average Output |
|----------------|----------------------|----------------|----------------------|
| 1              | 0.7934156            | 26             | 21.37055             |
| 2              | 1.616007             | 27             | 22.19363             |
| 3              | 2.439586             | 28             | 23.01672             |
| 4              | 3.2626712            | 29             | 23.8398              |
| 5              | 4.085763             | 30             | 24.66289             |
| 6              | 4.908841             | 31             | 25.48597             |
| 7              | 5.7319264            | 32             | 26.30906             |
| 8              | 6.5550117            | 33             | 27.13214             |
| 9              | 7.378097             | 34             | 27.95523             |
| 10             | 8.201182             | 35             | 28.77831             |
| 11             | 9.024267             | 36             | 29.6014              |
| 12             | 9.847352             | 37             | 30.42448             |
| 13             | 10.670438            | 38             | 31.24757             |
| 14             | 11.493523            | 39             | 32.07065             |
| 15             | 12.316608            | 40             | 32.89374             |
| 16             | 13.139693            | 41             | 33.71682             |
| 17             | 13.962778            | 42             | 34.53991             |
| 18             | 14.785864            | 43             | 35.36299             |
| 19             | 15.608949            | 44             | 36.18608             |
| 20             | 16.432035            | 45             | 37.00916             |
| 21             | 17.25512             | 46             | 37.83225             |
| 22             | 18.078205            | 47             | 38.65533             |
| 23             | 18.90129             | 48             | 39.47842             |
| 24             | 19.724375            | 49             | 40.3015              |
| 25             | 20.54746             | 50             | 41.12459             |

**TABLE II. Calculated Predictions by Machine Learning Algorithms**

**ELEVATION INACCURACIES IN METERS**

![Elevation Inaccuracies Graph](image)

*Fig. 13. Elevation Inaccuracies [7]*
Fig. 14 shows the relation between the actual floor the device was tested on with the average readings given by the prototype from the device sensor readings to demonstrate its accuracy. We have used machine learning algorithms with linear regression to calculate the output for other floors based on system behavior with an accuracy of 99.91%. The calculated predictions are listed in Table 2. From the results, we observed that the accuracy of the system is similar to devices from rank 1 to rank 7 from Fig. 13. Since we are only using relative measurements of the sensor to differentiate between floors, we get a very high accuracy even if the true GPS readings of the sensor may not be accurate enough for other purposes.

6. Conclusion

We conclude that if the budget-friendly devices are integrated with a better quality of sensors, this system would find a wide application in the smart phone world. Due to the usage of relative readings of the sensor at each floor in contrast to its true readings, the system works quite efficiently for our purpose of tracking in a building. There are several applications where this idea for tracking can be applied like Briefcase Security, Real-Time Tracking, Smart Device Tracking, Military Tracking System. This will make it much easy for the user to track not only in latitude and longitude but also in terms of altitude for a much accurate tracking experience in 3-D.

REFERENCES

[1] E. D. Kaplan, Understanding GPS: Principles and Applications, Artech House Publishers, ISBN 0890067937, February 1996
[2] R. J. Bates, GPRS: General Packet Radio Service, McGraw-Hill Professional, 1st Edition, ISBN 0071381880, November 12, 2001.
[3] N. Chadil, A. Russamesawang, and P. Keeratiwintakorn, “Real-time tracking management system using GPS, GPRS and Google Earth,” 5th Int. Conf. Electr. Eng. Comput. Telecommun. Inf. Technol. ECTI-CON 2008, vol. 1, pp. 393–396, 2008, DOI: 10.1109/ECTICON.2008.4600454.
[4] M. Mcdonald, H. Keller, J. Klijnhout, and V. Mauro, Intelligent Transport Systems in Europe: Opportunities for Future Research, World
[5] Google, Inc., Google Earth software, http://earth.google.com/ [last accessed on August 12, 2020].
[6] Google, Inc, Keyhole Markup Language Documentation Introduction, http://code.google.com/apis/kml/documentation/ [last accessed on March 12, 2020].
[7] C. Bauer, “On the (in-)accuracy of GPS measures of smartphones: A study of running tracking applications,” ACM Int. Conf. Proceeding Ser., no. December 2013, pp. 335–340, 2013, DOI: 10.1145/2536853.2536893.
[8] Blynk, Inc, Blynk - IoT platform for business and developers. https://blynk.io/ [last accessed on August 12, 2020].