



Designing an Intelligent Real-Time Public Transportation Monitoring System Based on IoT

Thair A. Salih, Noor K. Younis

Technical Engineering College, Northern Technical University, Mosul, Iraq

Email: Thairali59@ntu.edu.iq, noorky@ntu.edu.iq

How to cite this paper: Salih, T.A. and Younis, N.K. (2021) Designing an Intelligent Real-Time Public Transportation Monitoring System Based on IoT. *Open Access Library Journal*, 8: e7985.

<https://doi.org/10.4236/oalib.1107985>

Received: September 20, 2021

Accepted: October 26, 2021

Published: October 29, 2021

Copyright © 2021 by author(s) and Open Access Library Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

Abstract

In recent years, smart public transportation has become one of the most important things for developing cities and improving people's quality of life. Public transportation users face many problems, the most important of which is the long wait at the bus station. The proposed system in this paper helps users of public transportation to find public transportation, arrival times, and other information from any place and at any time using a mobile application. The main objective of the prototype is to reduce the wait time at the bus station by knowing the nearest buses to a user, the real-time location of buses on the Google map to help passengers track buses in real-time, the arrival time of buses, and speed. The system was implemented based on Internet of Things (IoT) technology, by using the Global Positioning System (GPS), a microcontroller with a built-in Wi-Fi module (ESP32), and a mobile user interface by the Blynk IoT platform. The proposed system has been implemented and tested in real-time, where all data obtained is displayed by the GPS sensors for bus locations (longitude and latitude) and speed on the smartphone application. The distance between the bus location and the passenger that will appear in the mobile app was calculated by using the Haversine formula. It measured the accuracy of the distance obtained based on a study of several days at different times on multiple roads in Mosul city and compared it with the actual distance. The average difference between the calculated values computed by the Haversine equation compared to the data obtained from the actual distance is (177 meters) with a minimum error of 8 meters. The arrival time of each bus was calculated based on the distance and average speed of the bus registered along the road.

Subject Areas

Wireless Communication, Computer Engineering

Keywords

ESP32, Global Position System (GPS), Internet of Things(IoT), Blynk, Haversine

1. Introduction

Public transportation refers to shared passenger transportation services such as buses, trolleybuses, trains, ferries, and expresses transportation like the metro [1]. Intelligent Transportation Systems (ITS) has a subsystem called Smart Public Transportation (SPT). It can intelligently monitor public transit networks to ensure their operation and to provide clients with information on excursions and system operating conditions [2]. Smart public Design of an intelligent real-time Public Transportation Monitoring System based on IoT transport systems, depending on several technologies, allows SPTS to retrieve data from multiple sensor systems and to manage and control the transportation network. There are many innovative technologies, which promoted the development and implementation of smart public transport systems, such as Geographical Information Systems (GIS), Automatic Vehicle Location Systems (AVLS) and Traveler Information Systems (TIS) [3]. The Internet of Things (IoT) is a network of interconnected physical objects that can be accessed through the internet. As countries aim to improve their citizens' standard of living, they begin to improve the infrastructure of their cities, towns, and villages. Improving public infrastructure would also require bettering city transit services. In cities, bus services are the backbone of the public transportation system (PTS). Because of many issues such as wait times, traffic congestion, etc., the reliability of public transport is being diminished; SPTS is using IoT to avoid any of these issues [4]. Wireless communication is the transmission of data between two or more points without the use of wires or cables. There are several technologies used to manage and control public transportation systems [5].

Through combining information technology, advanced techniques, and smart sensing systems, previous works show that different wireless technologies manage and monitor smart public transportation systems such as RFID, IR, GPS, Bluetooth, and Wi-Fi, etc.

Parmeshwar *et al.* (2017) [6] create and introduce a smart bus transport system using GSM, GPS and Arduino. GSM, GPS sends a message to passengers who are waiting for the next bus to arrive. Passengers counted automatically in this scheme.

Kiran *et al.* (2017) [7] smart bus tracking system based on Wi-Fi and a real-time bus tracking mobile application. The app uses the bus terminal's address to send data to the cloud, where it is retrieved and shown to the user in a mobile app. Users are also told when the bus is approaching the bus station.

An Android application for vehicle tracking and monitoring [8] has the capa-

bility of current passenger count and that count is automatically updated, as is the location of the passengers. Passengers' wait times and other issues will be reduced, using smartphones and technologies like GPS, public transportation, Android, MySQL, and RFID chips.

As an Android App for tracking buses and calculating distances to stations along their routes [9], the tracking system includes placement of GPS, RTC, and Arduino UNO in a bus, and an Android App installed on any smartphone to track the bus location.

The system includes a GPS and a sensor built into the bus that counts passengers [10]. Obtain this information via Wi-Fi or the Internet at the bus station. A RaspberryPi3 is used to process GPS and sensor data before sending it to a server.

The prototype collects data at the bus station, uploads the updated information to a server via the internet, and displays the data to customers [11]. RFID, GPS, and Wi-Fi built-in into the controller are among the main technologies employed in this work.

A paper suggests a system for tracking public buses using GPS (Global Positioning System), GPRS, IR, and an Android application [12]. The passengers can monitor the location of the bus by using an app installed on their smartphone to track position, bus numbers, routes, bus stops, and bus timings.

An Arduino Uno, Wi-Fi Module, Router, and GPS are used to build and develop a Smart Bus Tracking and Management System that can be controlled and monitored from anywhere using an internet connection and a mobile phone [13].

The system consists of GPS and the Internet of things (IoT) cloud to store data [14]. Monitoring is done by an Android application installed on the officials' phones. If there is a delay, an alarm will give to the driver and officials regarding the delay.

The biggest problem facing users of public transport is related to the schedule and the availability of seats on the bus [15]. Passengers can track the bus's real-time location at any time using GPS (Sim808), Arduino UNO, ESP8266, and RFID.

It is a prototype for developing a real-time system capable of monitoring arrivals, departures, and bus data at all bus intersections [16]. The RFID module can be used with the GSM module to design this system. RFID will assist in getting data for a specific bus and storing it at the designated bus station.

A vehicle monitoring and tracking system was created using the Blynk platform as a data transport [17]. Nod MCU ESP8266, Ultrasonic sensors, gas sensors, infrared sensors, temperature sensors, and GPS modules are used. The smartphone application notifies the driver or other passengers in the vehicle.

The system can book tickets automatically and track buses in real-time [18]. RFID (Radio Frequency Identification), GPS will help with the location of the bus. The system will calculate the distance traveled and deduct the money from

the passenger's E-wallet, which can be managed through the mobile app.

The system uses Global Positioning System (GPS), Arduino, and IR (infrared) sensors [19]. An Android app is used to track the bus's position. The app includes details about the bus, such as its number, routes, bus stops, and arrival and departure times.

The suggested system offers the actual position of the desired bus as well as numerous specifics, such as access time, number of passengers on the bus, bus speed, and departure time [20]. By using the Node MCU, GPS module, Wi-Fi IoT module, Arduino UNO, and Android application.

The suggested proposed system in this paper is a low-cost, IoT-based, real-time, and easy-to-use system. It consists of a GPS module and an ESP32 (microcontroller & Wi-Fi module) that will be installed inside the buses. GPS is used to obtain a real-time bus location, as well as speed, direction, etc. All data collected by ESP32 will be transferred to the Blynk server through Wi-Fi, then displayed on the mobile application. The system aims to reduce the wait time at the bus station.

The paper is organized as follows. Section 2 describes the proposed system methodology. Section 3 presents the implement of the proposed system. Section 4 discusses the results obtained through implementing the proposed system, and finally, the last section provides the conclusion.

2. Proposed System Block Diagram

The prototype of the proposed system is in **Figure 1**. It consists of an Android application designed for users who want real-time information about the buses. The app will display information about buses such as real-time location on Google Maps, speed, distance, and arrival time of each bus. The proposed system includes an ESP32 with a Wi-Fi built-in module, a GPS module, and an Android app connected to the server. The proposed system is operated by GPS and

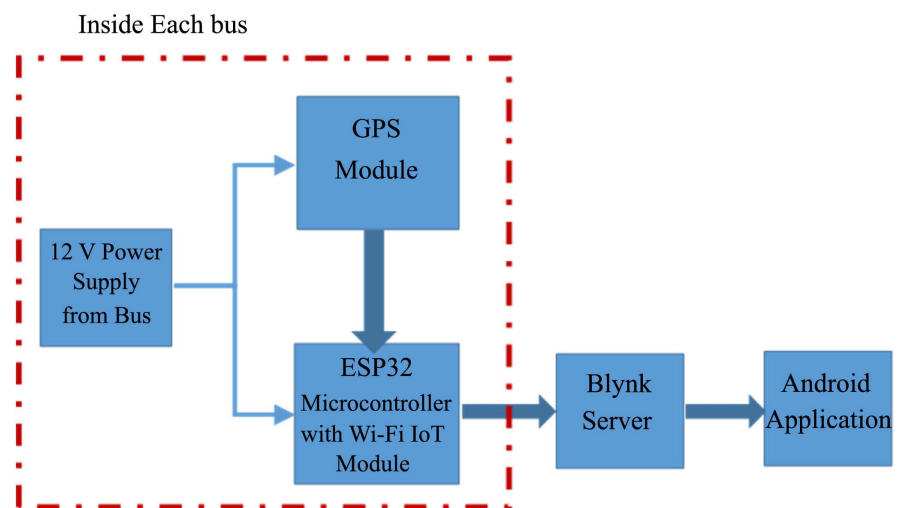


Figure 1. Block diagram of prototype of smart public transportation system.

ESP32, which are installed in each bus with a power supply that may be obtained from the bus. GPS receives the satellite signals and then the position coordinates with latitude, longitude and speed are determined for moving buses. After receiving the data, the tracking data can be transmitted using wireless communications systems. In this system, the ESP32 is a microcontroller with a Wi-Fi module. All the information collected by ESP32, such as location (latitude and longitude), speed, etc., will be uploaded to the Blynk Server. Based on IoT, the user can access this information on a bus through the Android application. The Haversine formula is used to calculate the distance and estimated arrival time that will be shown on an Android application.

3. Implement of the Proposed System

The prototype of smart public transportation is in **Figure 2**. It consists of two parts: The first part is the communication unit, which includes a GPS module and an ESP32 micro-controller with a Wi-Fi built-in module. This unit is used in the public transportation system for vehicle monitoring and tracking. With the help of the GPS module, it can determine the current position and calculate the speed of buses. The GPS data is transferred to the Blynk server with the help of the Wi-Fi module for storage and analysis. Then it is displayed on the mobile phone application. The second part is the mobile application. The android application gets data from the Blynk server and provides the required data to the user based on the information provided in the android application.

3.1. ESP32 Microcontroller

The ESP32 is a microcontroller with a Wi-Fi module, an open-source IoT platform that is characterized by low-cost and low-power system-on-a-chip (SOC). An ESP32 has a dual-core structure and internal modules such as Wi-Fi, Bluetooth, and many Peripheral Interfaces such as IR, SPI, CAN, Ethernet, and temperature sensors [21]. The specifications of the ESP32 are given in **Table 1**.

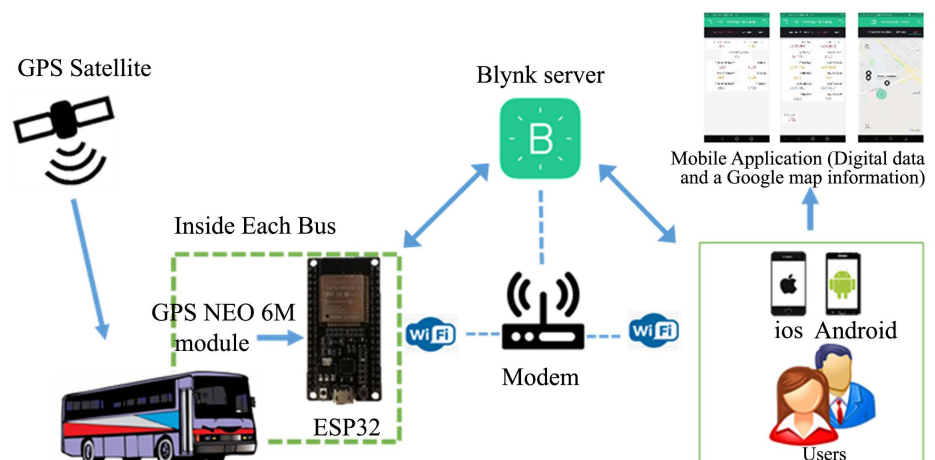


Figure 2. Prototype of smart public transportation architecture.

Table 1. ESP32 specification.

CPU	Tensilica Xtensa LX6 32 bit Dual Core at 160/240 MHz
SRAM	520 KB
FLASH	2 MB (max. 64 MB)
Voltage	2.2 V to 3.6 V
Operating Current	80 mA average Free
Programmable	Free (C, C++, Lua, etc.)
Open Source	Yes
Wi-Fi	802.11 b/g/n
Bluetooth	4.2 BR/EDR + BLE
UART	3
GPIO	32 pins

3.2. GPS Module

GPS (Global Position System) used for positioning and tracking buses based on satellite communication. GPS satellites cover the entire earth at all times. To get accurate GPS location data, there should be a minimum of three satellites. The NEO-6M GPS module used in the proposed system is small and works on very low power, making it ideal for tracking applications. The GPS module operates at 3.3 V, as a result, powered by connecting the GPS module to the 3.3 V pin of the ESP32, respectively shown in **Figure 3**.

3.3. Mobile Application

The Android application is critical to this system's success. It assists passengers by providing information about the bus they need to take. It works as a link between the user and the server. In this system, the android application is designed by using the Blynk platform. The Blynk platform is an IoT platform, offering a way to design wonderful applications in minutes for Android and iOS smartphones. It can be used to manage different controllers, such as Raspberry Pi, ESP8266, ESP32, etc. The Blynk platform's architecture includes Blynk libraries, Blynk server, and Blynk apps [22]. The Android app shown in **Figure 5** includes a map that will show the passenger's location, as well as track the present location of the bus. It also provides the estimated time for the bus to arrive, the speed of the moving bus, and displays the nearest bus to the user by calculating the distance between the passenger and the bus location.

3.3.1. Distance Calculation

The Haversine formula was adopted to calculate the distance that will appear in the Android app. It calculates the distance between the passenger and the bus location using the latitude and longitude of the bus and the passenger who is at home, work, or at the bus stop. The following equations can be used to calculate the distance [23].

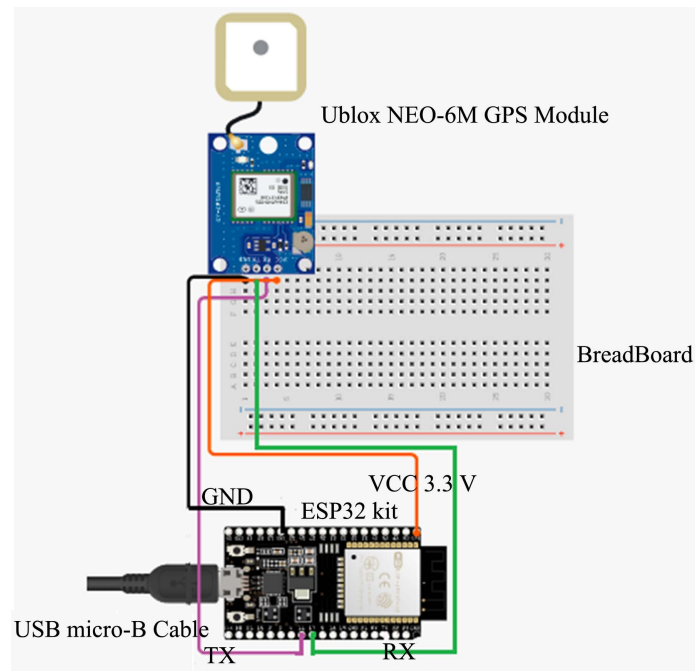


Figure 3. GPS module to ESP32 connection.

$$a = \sin^2(\Delta\varphi/2) + \cos\varphi_1 \cdot \cos\varphi_2 \cdot \sin^2(\Delta\lambda/2) \quad (1)$$

$$c = 2 \cdot \text{atan2}(\sqrt{a} \cdot \sqrt{1-a}) \quad (2)$$

$$d = R \cdot c \quad (3)$$

where: (“ φ ”) = latitude;

(“ λ ”) = longitude;

(“ $\Delta\varphi$ ”) = latitude 2 – latitude 1;

(“ $\Delta\lambda$ ”) = longitude 2 – longitude 1;

R = earth radius (6371 meters);

d = distance between the two location.

3.3.2. Arrival Time Calculation

The arrival time of each bus, which will appear in the Android app, is calculated based on the following equation.

$$t = d/s * 60 \quad (4)$$

where: t = arrival time of bus.

d = distance between the passenger and bus location that get from Equation (3).

s = average speed of bus along route.

To convert the arrival time (t) that will be obtained from Equation (3) from an hour into a minute, it is multiplied by 60 (one hour into a minute).

3.4. Work the Proposed System

The proposed system method is working procedure first; the user must deter-

mine his location by activating the location feature in a smartphone. To get the information entered the application will provide the details about buses, bus location, bus speed, bus arrival time, nearest bus from a user by offering the distance between user location and bus. This information will assist the passenger to select their suitable bus. In the flowchart that describes the proposed system work in **Figure 4**. A GPS module connected to an ESP32 Microcontroller with a built-in Wi-Fi module is placed inside each bus. When the power supply is on, the GPS module communicates continuously with the satellite to get coordinates. The GPS module will initialize itself, then the module will get the coordinates, but if the coordinates are not received, then the module will initialize again. Once the GPS obtains the coordinates, it sends the data, including latitude and longitude, and speed to the IoT Blynk server through the ESP32. At the Blynk server, the latitude and longitude are extracted and used on the visual map in the Blynk application. The live location of the bus can be seen on the Google map. Continuous data digital updates such as speed, distance, and the arrival time of the bus are displayed on the mobile application.

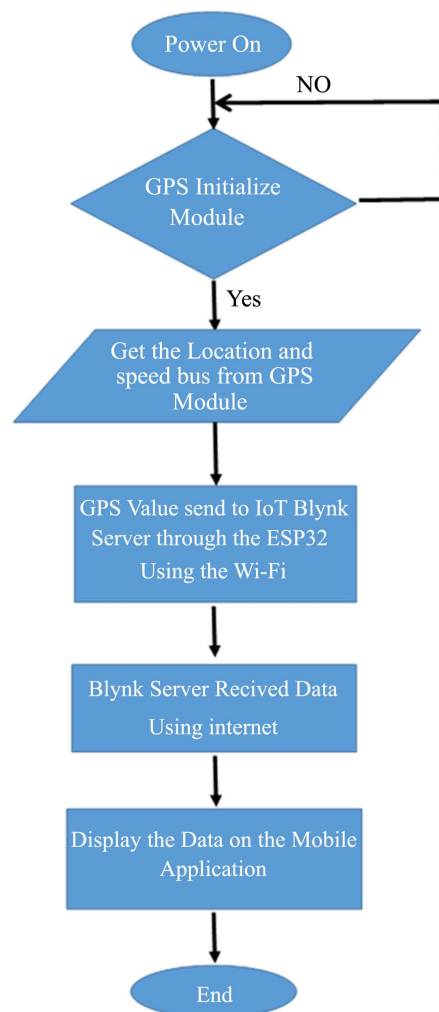


Figure 4. Smart public transportation system flowchart.

4. Results and Discussion

For testing the efficiency of the proposed system, the prototype has been installed (GPS unit and ESP32) inside a vehicle with supplied internet to use the possibilities offered by the Internet of Things. That vehicle roamed through multiple roads in Mosul city for several days and at different times, for collecting and recording data (latitude, longitude, speed, distance, and time of arrival). Depending on the system model, this information will be transmitted via a Wi-Fi internet connection to the Blynk server and then to the Android mobile application. **Figure 5** represents the result displayed in the end-users mobile application for the smart public transportation system.

4.1. Location Accuracy Analysis

The proposed system has been tested and **Table 2**. Display data obtained from several streets in the city of Mosul to determine the level of accuracy of the data obtained through GPS on several days and at different times.

4.2. Distance Accuracy Analysis

Figure 6 presents a graphic comparison of the distance level, the accuracy obtained through the calculation of the great circle distance equation, and the comparison between the total distance obtained from each of the test data and the total distance through real distance.

The Haversine equation used to measure the accuracy of the obtained distance. Based on a study of 4 days with different times, the average difference between the calculated values computed by the Haversine equation compared to the data obtained from a real distance is (177 meters) with an error ratio between 8 to 225 meters. This error is due to the fact that the Haversine equation

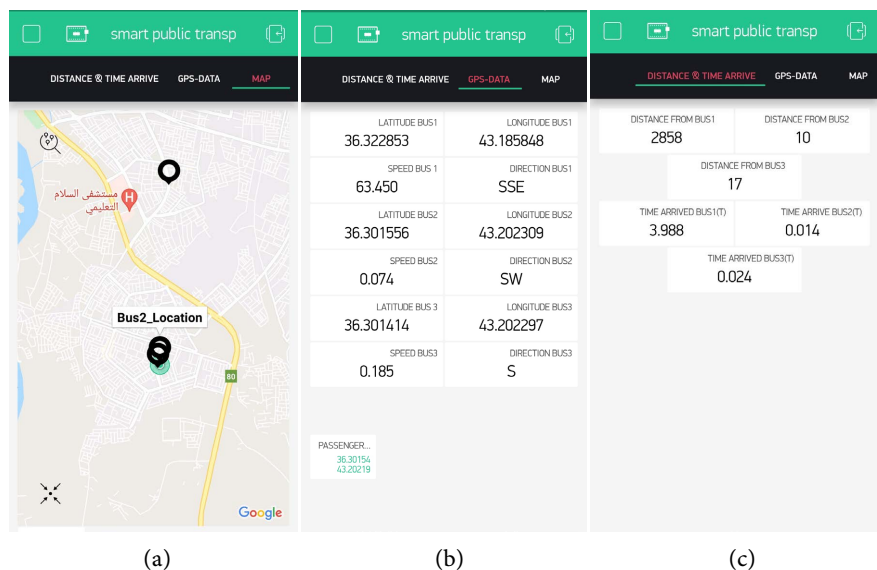


Figure 5. Mobile app for prototype of smart public transportation (a) Buses and passenger location on map, (b) GPS data, (c) Distance and arrival time of each bus.

Table 2. Several locations from the different streets in mosul city.

No.	Date	Time	Latitude	Longitude
1	17/2/2021	10:58 am	36.316739	43.191141
2	17/2/2021	11:04 am	36.350198	43.183983
3	17/7/2021	11:10 am	36.343765	43.182983
4	17/7/2021	11:13 am	36.337387	43.162712
5	17/2/2021	11:19 am	36.310638	43.207447
6	6/3/2021	12:30 am	36.3015221	43.202267
7	8/3/2021	11:38 pm	36.30159	43.20216
8	8/3/2021	11:39 pm	36.301559	43.202206
9	8/3/2021	11:40 pm	36.310688	43.203171
10	8/3/2021	11:43 pm	36.295567	43.195332
11	8/3/2021	11:55 pm	36.389160	43.187370
12	9/3/2021	10:05 am	36.377155	43.149384
13	9/3/2021	10:07 am	36.350117	43.238834
14	9/3/2021	10:15 am	36.376621	43.153072
15	9/3/2021	10:16 am	36.379650	43.153046
16	9/3/2021	10:17 am	36.381821	43.150658
17	9/3/2021	10:19 am	36.382019	43.156296
18	10/3/2021	10:07 am	36.335922	43.202955
19	10/3/2021	10:15 am	36.310688	43.203171
20	10/3/2021	10:24 am	36.301514	43.202290
21	11/3/2021	11:32 am	36.374424	43.147087
22	11/3/2021	11:33 am	36.371513	43.136124
23	11/3/2021	11:34 am	36.370644	43.133099
24	15/3/2021	7:15 pm	36.301514	43.202446
25	15/3/2021	7:16 pm	36.302582	43.201664
26	24/3/2021	1:25 pm	36.370796	43.133259
27	24/3/2021	1:27 pm	36.365589	43.120064
28	24/3/2021	1:28 pm	36.358574	43.111202

calculates the distance between two points on the surface of the Earth, assuming that the geometry of the earth is completely spherical (ignoring ellipsoidal effects) with a mean radius of 6.371 Km. Since the earth is not spherical, the earth in reality is elliptical with a radius between 6.378 Km (equatorial) and 6.357 Km (polar). This means the errors from assuming spherical geometry may reach 0.55%.

4.3. Arrival Time Accuracy Analysis

The arrival time between two points calculated based on Equation (4) as shown in Figure 7, where time accuracy depends on the accuracy of the distance in Table 3.

Figure 7 shows the average speed registered along route during 13 minute.

Table 3 displays the some overall results obtained applying the Equations (1)-(4) of the proposed system.

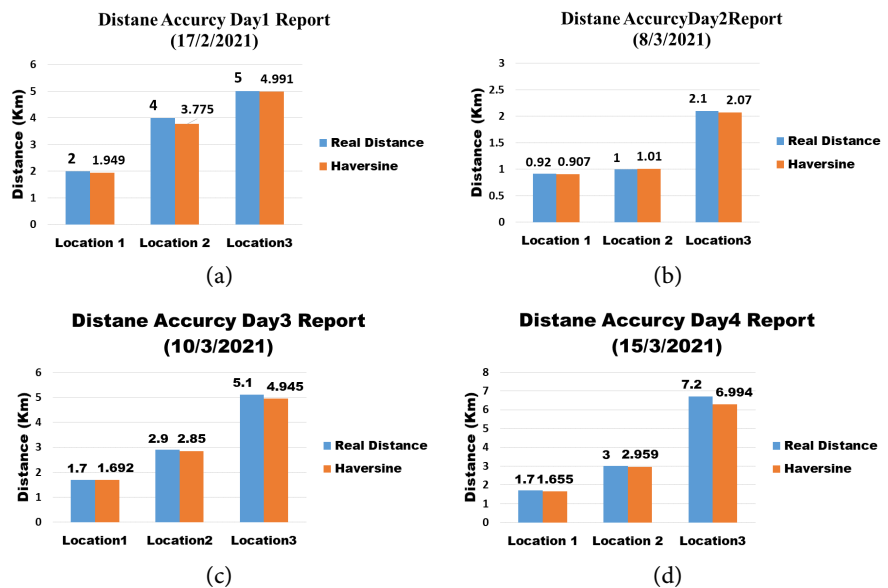


Figure 6. Graphic comparison of distance (a) Day 1, (b) Day 2, (c) Day 3, (d) Day 4.

Table 3. Results obtained applying of the proposed System.

No.	Date	Time	Distance (Km)	Average Speed (Km/h)	Arrival Time (minute)
1	15/3/2021	10:58:12 am	3.775 Km	49 Km/h	4.5 minute
2	15/3/2021	11:15:20 am	1.494 Km	59 Km/h	1.98 minute
3	15/3/2021	11:20:33 am	4.949 Km	60 Km/h	4.99 minute
4	21/3/2021	3:06:12 pm	9.450 Km	42 Km/h	11.340 minute
5	21/3/2021	3:12 :30 pm	6.618 Km	40 Km/h	7.94 minute
6	21/3/2021	3:16:33 pm	5.229 Km	31 Km/h	6.275 minute
7	21/3/2021	3:19:50 pm	2.237 Km	40 Km/h	3.63 minute
8	21/3/2021	3:23:13 pm	1.928 Km	49 Km/h	2.314 minute
9	24/3/2021	1:25:15 pm	10.500 Km	59 Km/h	12.6 minute
10	24/3/2021	1:25:50 pm	10.442 Km	64 Km/h	12.503 minute
11	24/3/2021	1:26::18 pm	10.398 Km	47 Km/h	12.478 minute
12	27/4/2021	11:35:24 am	2.229 Km	37 Km/h	3.6 minute
13	27/4/2021	11:36:26 am	2.130 Km	37 Km/h	2.972 minute
14	27/4/2021	11::37:17 am	1.432 Km	45 Km/h	1.998 minute
15	27/4/2021	11:37:58 am	0.938 Km	47 Km/h	1.309 minute

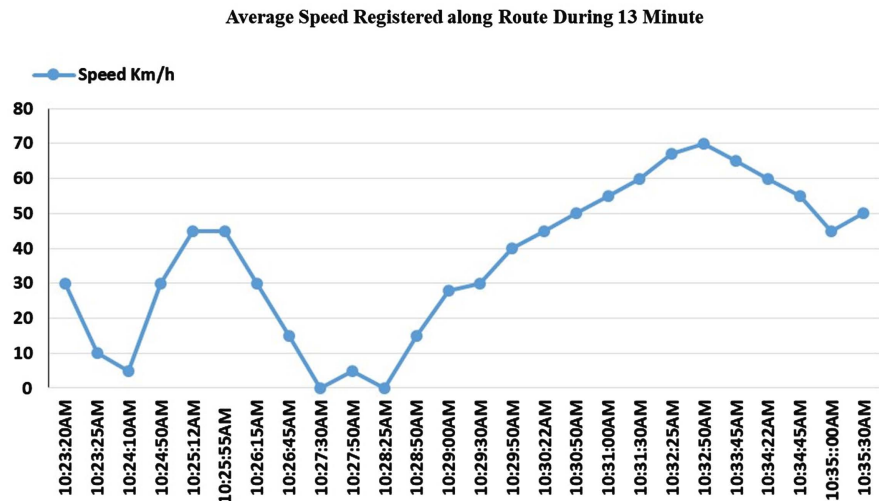


Figure 7. Graphic of average speed over the time the GPS-Data.

The proposed system is easy to use than other systems and more cost-efficient. It also provides greater performance than RFID and GPS-GSM systems. The application of this proposed system showed that using three GPS-Data only: latitude, longitude, and speed, it is possible to find information about buses, including arrival time, distance. This proposed system could be a valuable guide for those people involved in the study of GPS-Data gathered from moving buses. In the proposed system, the present location of the bus can be known by the latitude and longitude of the bus. The distance between the location of the bus and the location of the passenger is calculated based on the Haversine equation. The bus's arrival time is calculated based on Equation (4). All information is loaded onto the Blynk server and displayed on the mobile application.

5. Conclusion and Future Work

This paper introduces a prototype of a smart public transport system that is designed and implemented using IoT technology, GPS, and ESP32. This system shows that it can use GPS-Data only to get real-time information about buses, such as the current location of the bus, speed, arrival time, and distance. This system offers solutions for users of public transport who take a bus to reach their destinations. This system will help them to reduce the waiting time at the bus station because it provides passengers with necessary information about buses, like the current location of buses, the speed of the bus, arrival time, and the distance. The distance between the user and the bus location, which is calculated using the Haversine equation with a minimum error of 8 meters, and the arrival time of each bus is calculated based on the distance and the average speed of the bus registered along the road. The system is practical and reliable for transmitting data using Wi-Fi, compared with other systems that are based on GPS-GSM technology. Future work: The mobile application will be developed by adding additional features, using an IR sensor to count the number of passengers on the bus, and using RFID tags to implement an electronic ticketing system.

Conflicts of Interest

The authors declare no conflicts of interest.

References

- [1] Sutar, S.H., Koul, R. and Suryavanshi, R. (2016) Integration of Smart Phone and IOT for Development of Smart Public Transportation System. *International Conference on Internet of Things and Applications*, Pune, 22-24 January 2016, 73-78. <https://doi.org/10.1109/IOTA.2016.7562698>
- [2] Jananey, J., *et al.* (2019) Review on IoT Based Architecture for Smart Public Transport System. *International Journal of Applied Engineering Research*, **14**, 466-471. <http://www.ripublication.com>
- [3] Elkosantini, S. and Darmoul, S. (2013) Intelligent Public Transportation Systems: A Review of Architectures and Enabling Technologies. 2013 *International Conference on Advanced Logistics and Transport*, Sousse, 29-31 May 2013, 233-238. <https://doi.org/10.1109/ICAadLT.2013.6568465>
- [4] Apsara, S., Rashmi, G.A., Mohan Kumar, K.V., Anitha, L. and Jyothi, B. (2017) Internet-of-Things (IOT), Navigation System (NS), Micro Navigation (MN), Crowd-Aware Route Recommendation (CARR), Bus Arrival Time Estimation (Bate). *International Journal of Advance Research, Ideas and Innovations in Technology*, **3**, 1163-1170. <https://www.IJARIIT.com>
- [5] Divekar, S.N., Patil, S.R. and Shelke, S.A. (2018) Smart Bus System. *International Journal of Scientific Research in Science, Engineering and Technology*, **4**, 585-588.
- [6] Farooq, M.U., Shakoor, A. and Siddique, A.B. (2017) GPS Based Public Transport Arrival Time Prediction. <https://doi.org/10.1109/FIT.2017.00021>
- [7] Kiran, R., *et al.* (2017) Implementation of Smart Bus Tracking System Using Wi-Fi. *International Journal of Innovative Research in Science, Engineering and Technology*, **6**, 12940-12946.
- [8] Raut, P.P., Mahato, A., Nagane, A., Kuwar, A.M. and Chip, R. (2017) Public Transportation System Using IoT. *International Journal for Scientific Research and Development*, **5**, 240-242. <https://www.ijserd.com>
- [9] Sridevi, K. (2017) Smart Bus Tracking and Management System Using IoT. *International Journal for Research in Applied Science and Engineering Technology*, **5**, 372-374.
- [10] Vakula, D. and Raviteja, B. (2018) Smart Public Transport for Smart Cities. *Proceedings of the 2017 International Conference on Intelligent Sustainable Systems (ICISS)*, Palladam, 7-8 December 2017, 805-810. <https://doi.org/10.1109/ISS1.2017.8389288>
- [11] Geetha, S. and Cicilia, D. (2018) IoT Enabled Intelligent Bus Transportation System. *Proceedings of the 2nd International Conference on Communication and Electronics Systems*, Vol. 2018, 7-11. <https://doi.org/10.1109/CESYS.2017.8321235>
- [12] Kadam, A.J., Patil, V., Kaith, K., Patil, M.D. and Bendre, M.S. (2018) Smart Bus for Smart City Using IOT Technology. 180-184. <https://doi.org/10.1109/ICECA.2018.8474819>
- [13] Madhur, I., *et al.* (2018) Smart Bus Tracking and Management System. *Iconic Research and Engineering Journals*, **1**, 80-84.
- [14] Karthikeyan, G. and Jawahar, M.L. (2018) Smart Bus Management System. *International Journal of Engineering Research and Technology*, **6**, 1-4.

<https://www.ijert.org>

- [15] Sawant, Y.V., et al. (2019) Smart Bus System Using IoT. *International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering*, 7, 105-112. <https://doi.org/10.17148/IJIREEICE.2019.7418>
- [16] Godge, P., Gore, K., Gore, A., Jadhav, A. and Nawathe, A. (2019) Smart Bus Management and Tracking System. *International Journal of Latest Engineering Science*, 2, 20-25. <https://www.ijlesjournal.org>
- [17] Padmaja, B.V.S., Kolluru, V.R. and Kota, S.S. (2019) IoT Based Implementation of Vehicle Monitoring and Tracking System Using Node MCU. *The International Journal of Innovative Technology and Exploring Engineering*, 8, 446-450.
- [18] Shah, M.J., Prasad, R.P. and Singh, A.S. (2020) IOT Based Smart Bus System. *3rd International Conference on Communication System, Computing and IT Applications (CSCITA)*, Mumbai, 3-4 April 2020, 130-134. <https://doi.org/10.1109/CSCITA47329.2020.9137816>
- [19] Harini, B.K., Parkavi, A., Supriya, M., Kruthika, B.C. and Navya, K.M. (2020) Increasing Efficient Usage of Real-Time Public Transportation Using IOT, Cloud and Customized Mobile App. *SN Computer Science*, 1, Article No. 159. <https://doi.org/10.1007/s42979-020-00161-8>
- [20] Kumar, V.M., Shobana, N.C., Kumar, P.P., Valentine, N.K., et al. (2020) Smart Bus Tracking System. *Journal of Critical Reviews*, 7, 1672-1677. <https://www.researchgate.net>
- [21] Maier, A., Sharp, A. and Vagapov, Y. (2017) Comparative Analysis and Practical Implementation of the ESP32 Microcontroller Module for the Internet of Things. *Internet Technologies and Applications (ITA)*, Wrexham, 12-15 September 2017, 143-148. <https://doi.org/10.1109/ITECHA.2017.8101926>
- [22] Htwe, H.N., Mon, Z.Z., Mya, A. and Aung, M. (2019) Design and Implementation of IOT Based Vehicle Tracking System. *International Journal of Science, Engineering and Technology Research*, 8, 374-379. https://www.academia.edu/42909622/Design_and_Implementation_of_IOT_Based_Vehicle_Tracking_System
- [23] Prasetya, D.A., Nguyen, P.T., Faizullin, R., Iswanto, I. and Armay, E.F. (2020) Resolving the Shortest Path Problem Using the Haversine Algorithm. *Journal of Critical Reviews*, 7, 62-64.