Malaysian Journal of Computing, 8 (1): 1250-1263, 2023

eISSN: 2600-8238

MOBILE APPLICATION LUGGAGE TRACKING SYSTEM USING GPS MODULE AND NODEMCU

Nurul Fitriah binti Abdulham[1] **Noorhayati Mohamed Noor**[2*]

[1,2*]*Faculty of Computer and Mathematical Sciences, Universiti Teknologi MARA, 40450 Shah Alam, Selangor Malaysia*

[1]fitriahabdulham@gmail.com, [2*]noorhayati@tmsk.uitm.edu.my

ABSTRACT

In this paper, a mobile application for luggage tracking system to track a misplaced or stolen luggage is designed. The application is to help traveller to track their misplaced luggage or stolen in public places. The luggage tracking system is a prototype of device tracker using a GPS and NodeMCU which will be kept inside a luggage. The device will be tracked via GPS satellites and send the data to the GPS module, and the luggage's owner will be informed of its whereabouts through the mobile application. The luggage tracking system uses wireless communication, which is Wi-Fi with the assistance of a microcontroller for location transmission. This project system also includes a mobile application to track and trace the luggage's position using Google Map with a marker on it.

Keywords: *Google Map, GPS, Mobile Application, NodeMCU, Real-time, Tracking System.*

Received for review: 17-06-2022; Accepted: 18-02-2023; Published: 10-04-2023

DOI: 10.24191/mjoc.v8i1.17698

1. Introduction

Nowadays, there are many people who travel from one place to another place. The Statistics Malaysia Department reported that the number of travellers increased from 7.7% in 2019 to 8.1% in 2019. This illustrates that many travellers love to carry important stuff such as jewellery, documents, camera, and other valuable stuff. They'll keep all the valuable stuff in their luggage. We have an instinct, as human beings, to protect our valuable items from burglary. People wants to secure their important things in the luggage but there might be a chance of losing the luggage or stolen in the public places like airport, bus station and railway.

Furthermore, from time to time, mishandled baggage at the airport has increased. Traveler did not know that their luggage was missing until they had not able to retrieve their luggage despite waiting for a long time. So, it's necessary to track down the luggage in case of loss or burglary. There are a number of tracking systems, as we know, and devices have already been developed, such as car tracking systems and person tracking systems.

When people travel around the world, in public places such as the airport, bus station, and train station, they will not notice the loss of their luggage because they were occupied, luggage is stored in the allocate with their boarding ticket or transportation status. Besides, there are some possibilities of people get robbed at the airport or accidentally exchanged their baggage without noticing it. According to Goldsten (2017), the Daily Mail stated that there

were more than 200 baggage robberies a day at NY's JFK airport. The airport workers were behind robbery, Daily Mail also said, and the airline company merely announced the luggage as missing. Therefore, this incident lead people to lose their luggage with the important stuff inside the luggage in airport and other places and may not be able to retrieve it back.

In this paper, we propose a prototype to track the location of the misplaced or lost luggage. This project aims to develop a prototype of a personal device tracker utilizing GPS and NodeMCU. The device will be placed in the baggage and will be used to track the luggage using a GPS module. The precise position of the luggage will be displayed on a map using the Tracking system application, which is already installed on the user's mobile phone. The following is a breakdown of the paper's structure: The related work is presented in Section 2. The prototype's system architecture is explained in Section 3. The testing technique is then described in Section 4, followed by a discussion of the results in Section 5. The paper concludes with a discussion of the limitation of the prototype.

2. Related Work

A tracking system is a system used to track objects or significant items. A tracking system is now an important tool for industries who want real-time information such as luggage, car, an individual, or merchandise about their stuff. The system helps the organization to operate smoothly and efficiently. For example, the oil and gas company are implementing the system because it contributes to future operating cost savings. An ideal system regularly updates the location, elevation, and range of the target.

The luggage tracking system is a tracker for personal devices that can be used by anyone to locate and monitor their luggage. Mostly, people in public places such as airports, trains, and station buses will lose or misplace their luggage. There may even be a risk of stealing objects or luggage. This device used GPS technology to control the items and GSM technology to alert the luggage owner where the luggage was placed.

Traditionally, police or civilians will check for missing or lost luggage by asking others near them or monitoring and looking at CCTV footage. Also, airlines use RFID or a barcode to monitor each luggage at the airport. It has been used to identify automated toll collection, access control, surveillance, equipment monitoring, and other applications. RFID stands for Radio-Frequency Identification. Although the airport has also introduced barcodes to sort, register, and classify the luggage of people. However, there are also drawbacks to the barcode system, such as a significant volume of mishandled baggage at the airport. After the RFID device is used by airline companies, the amount of mishandled luggage will dramatically decrease.

In addition, some organizations use Bluetooth beacons, QR codes, RFID and Wi-Fi to track and monitor the luggage of people at airports. For example, Senthilkumar *et al.* (2017) have designed a luggage tracking system by using a GPS shield and GSM module and has an alarm to detect the lost luggage. This project has one specific feature: if the bags go out of a specific range or move far away from the owner, the alarm will ring loudly and the owner will be alerted where the bags are placed.

One project based on Radio Frequency Identification (RFID) technology to track objects Devaki *et al.* (2019) because of its uniqueness which is can be used everywhere either at the airport or smart city. The smart RFID tag stored certain amount of data that can be used in identification to prove that is the owner of the luggage. The developer of this project is used a passive RFID for luggage identification. The RFID reader will read the smart RFID card with its unique number and owner details that attached to the luggage.

Another researcher has been developed a tracking system by using the combination of RFID and GPS module. Karthick *et al.* (2017) developed a tracking device with the aid of the

GPS Module and RFID tag attached to the bags luggage. The author uses a fingerprint that is better than biometrics such as voice recognition, which is not quite reliable, to lock their luggage. The owner must scan the fingerprint on the scanner to open the luggage and the scanner will align the fingerprint with the registered fingerprint. If an unauthorized person accesses the luggage, the warning sounds and the device immediately sends the location of their luggage via SMS to their cell phone. An RFID tag with a special tag ID and an ID saved in the cloud is available for each luggage.

Research used Wi-Fi and finger fingerprint technology to develop a smart wearable real-time airport tracking. For example, a project Ghazal *et al.* (2016) designed a tracking system that has a smart tag that can be charged and conveniently placed in the luggage of consumers. This project interacts over Wi-Fi between the computers and the database server. Notification will be sent to smartphones and smartwatches of customers, if their baggage has yet arrived, the information will consist of location and expected arrival time (ETA). If the location suits the traveller's final destination, the smart tag will reflect its current location. The device would align the two positions with k-Nearest Neighbour (k-NN) (Ghazal *et al.,* 2016). For localization, the RSS of the smart tag is sent every second to the server. Until the tag is switched off as the owner gets the luggage, this radical pattern persists. This method is referred to as Wi-Fi fingerprinting and is used to localize the proposed system indoors.

Another research related to short range tracking system Adjei *et al.* (2020) has created a low power consumption real-time system which uses only a 5V battery and is very portable. Using a Bluetooth module and a GPS module, this real-time gadget can detect the missing stuff in real-time. At first, the device's Bluetooth are combined with the user smartphone's Bluetooth, if the connections get disconnected the device will send the user a call that is something like a warning; meaning that if the user does not see the message containing the location for some reason, this call draws attention of the user to their smartphones.

3. Methodology

3.1 Proposed Design

This segment would emphasis on the designing of the circuit diagram for the Arduino microcontroller, the proposed system block diagram, the proposed system flowchart, and the mobile application interface.

3.1.1 Circuit Diagram

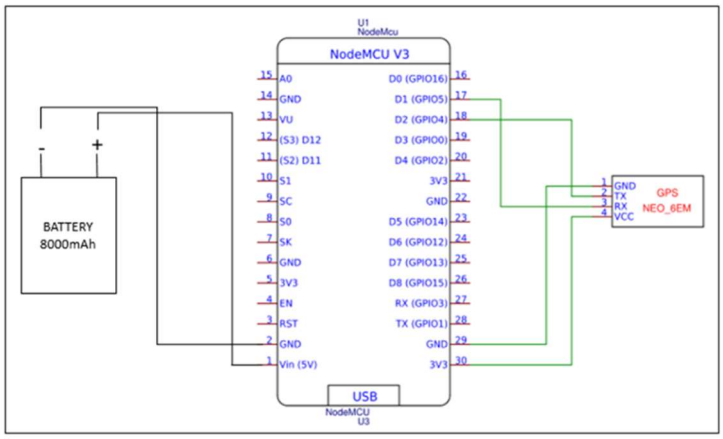


Figure 1. Luggage Tracking Device Circuit.

3.1.2 Block Diagram

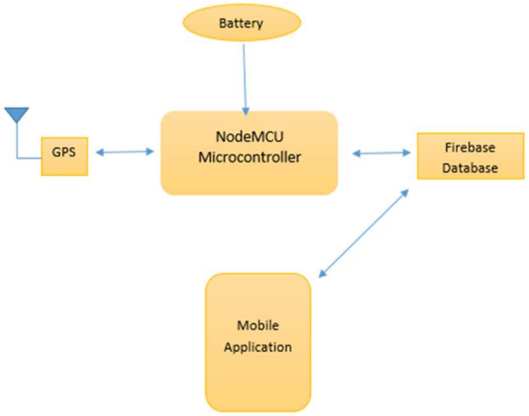


Figure 2. Block Diagram of System.

The proposed block diagram where the real-time GPS device tracker will be kept in the luggage. Figure 2 represents the system block diagram. The proposed system includes a GPS module, a microcontroller and a battery.

3.1.3 System Flowchart

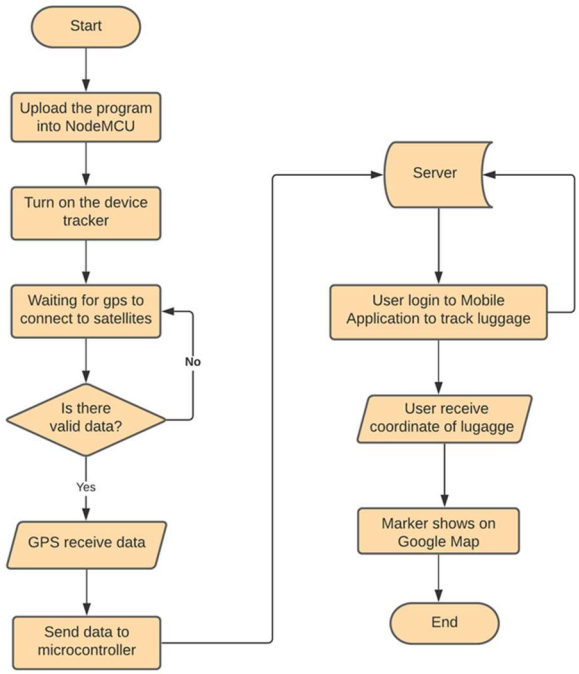


Figure 3. System Flowchart.

This project begins with uploading the application code to the NodeMCU v2 after it has established a Wi-Fi connection. The device tracker requires at least four satellites to measure the location. Satellites provide location information in the form of radio waves to the GPS receiver. It will send the coordinates to the GPS receiver if there is a valid location data. If there no valid location, it means that the GPS have not get the signal from satellites yet. After that, the GPS module will send the data to the microcontroller, which will automatically store the GPS coordinates in a database for used.

In order to track their luggage, the user must first log onto the mobile application. The database will contain all user information, such as email and passwords. When users click the Location button, a Google Map with a marker appears.

3.2 Proposed Design

3.2.1 NodeMCU V2

The ESP-12E module has an ESP8266 chip with a Tensilica Xtensa 32-bit LX106 RISC CPU. This microprocessor is RTOS compatible and has a clock frequency range of 80MHz to 160MHz. NodeMCU is an open-source Lua-based firmware and development board designed specifically for Internet of Things (IoT) applications.

Figure 4. NodeMCU V2.

3.2.2 GPS Modem (Neo-6m GPS)

NEO-6 module series is a family of stand-alone GPS receivers using the high-performance u-blox 6 positioning engines. It comes with tiny battery that can be easily integrated with microcontrollers for board selection. Built-in EEPROM and hot-start. A high-performance GPS module with patch antenna and an on-board memory chip also provided.

Figure 5. Neo-6M GPS.

3.3 Software Requirements

3.3.1 Arduino IDE

The Arduino IDE is an open software that, like the Arduino Uno board, is used to compile the program and load it into the microcontroller. The Arduino IDE is a cross-platform framework written in both C and Java programming.

3.3.2 Firebase Real-time Database

Firebase is the development platform for mobile apps from Google that will assist the developer to store, enhance and expand their product. The real-time database of Firebase is often referred to as cloud-hosted NoSQL. Firebase has a special function where data can be saved and synchronized in real-time between users.

3.3.3 Android Studio

Android Studio is Google's Android Operating System Optimized Development Environment (IDE). The Android studio can operate with any operating system, such as Windows, Apple and Linux. Android Studio offers developers a well-stocked toolkit that supports the Google Cloud Platform for designing Android software apps or other projects.

3.3.4 Flutter Framework

Flutter as an open-source platform for scripting and developing native Android and iOS apps. Flutter offers simplicity of programming with native-like performance, all while retaining visual consistency across platforms. Dart is the programming language used by Flutter. Dart was created to be a substitute for JavaScript.

4. Result and Findings

4.1.1 System Prototype Model

A prototype has been developed to illustrate how this Mobile Application Luggage Tracking System using GPS module and NodeMCU.

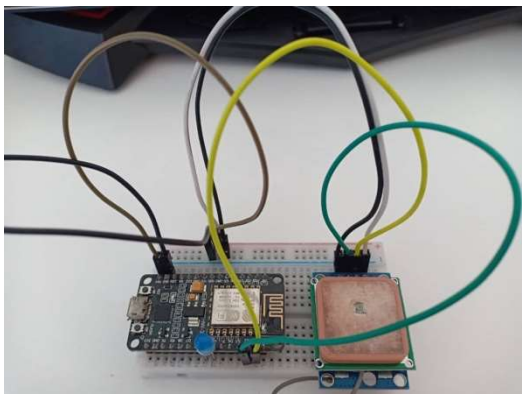


Figure 6. Device Tracker Prototype.

A tangible prototype model based on the luggage size has been constructed, as shown in figure 6. The NodeMCU, GPS module, and LED are used in this prototype. The rationale for using NodeMCU for this device tracker is that it has 128KB RAM, whereas the Arduino Uno only has 2KB RAM. Furthermore, while NodeMCU has a micro-USB port and UNO has a USB type B connection, micro-USB cables are more readily accessible than USB type B cables. Since the NodeMCU development board is smaller than the Arduino Uno, it suitable for the device tracker and packed in a box and carried in a suitcase.

4.1.2 System Prototype Model

This Mobile Application Luggage Tracking System's mobile application UI design. The system interface, the registration interface, the login interface, the profile interface, the location interface, and the main interface have all been successfully developed. This user interface was designed to make the application as easy to use as possible. This project's interfaces are depicted in the diagram below.

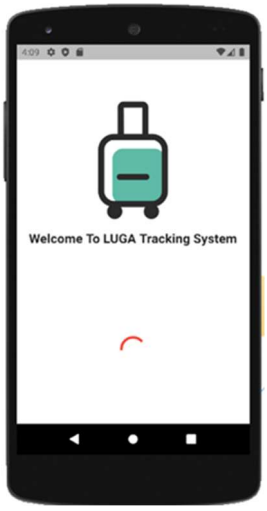


Figure 7. Welcome Screen Interface.

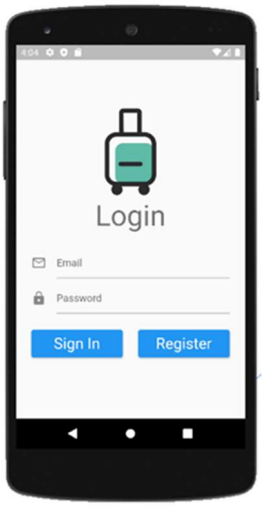


Figure 8. Welcome Screen Interface.

The mobile application, LUGA Tracking System, appears on the welcome screen, which is the first thing a user sees when they start the application. This screen will be displayed for five seconds before entering the application's login interface.

The login screen for users to access this application is shown in Figure 8 above. To access the main interface of this mobile application, the user must first input their email address and password, then click the sign in button. To use this application, the user must have a valid email address and a password of at least 6 characters. If the user does not meet the email and password validation requirements, a warning message will appear advising them to use appropriate email and password.

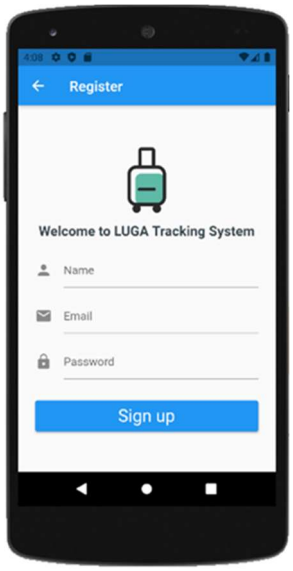


Figure 9. Register Screen Interface.

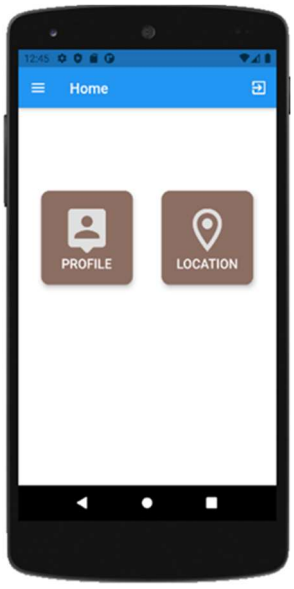


Figure 10. Home Screen Interface.

The register interface is shown in Figure 9. The user must first provide their full name, email address, and password before clicking the register button, which will register all of the user's information for this application. To register in this system, the user must use a valid email address and a password with at least 6 characters. All user information will be saved in firebase authentication so that the user may continue to utilize the LUGA tracking system.

Figure 10 show the main interface for this mobile application interface design. Once the user login in to this application the main page or home page will appear. This main screen shows the current data of the user profile and location of the luggage. The user can choose the menu by clicking the card and it will lead the user to another page. In this system only two buttons will work which is Profile and Location button.

Figure 11 below shows the profile screen of the user. The user's data that have been stored in firebase authentication is fetched and will be displayed in this interface. If the user wants to go to other interfaces or pages, they can click the home button at the app bar above, it will bring the user to the home page. If they click the sign-out button, the user will log out from this application.

The position of the luggage is depicted in Figure 12 using real-time data. In this part, a map has been created utilizing Google Services, specifically the Geolocation API, to assist users in locating their luggage. The map depicts the position of the device tracker that is installed in the luggage and allows the user to locate their belongings using this app. The map will display the position of the luggage on the map along with the phrase "Find your Luggage here," allowing the user to go to the indicated area to recover their luggage. The user may modify the camera location or the direction in which a vertical line appears on the map by clicking the green button at the bottom of the application. The red button, on the other hand, will display

the position of the luggage on a large-scale map. For example, it will display the current status of the location so that the user can see where their luggage is.

Figure 11. Profile Screen Interface.

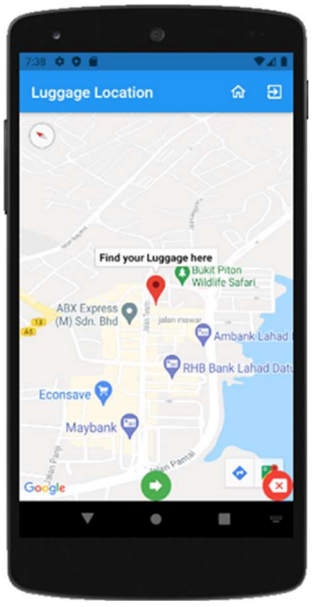


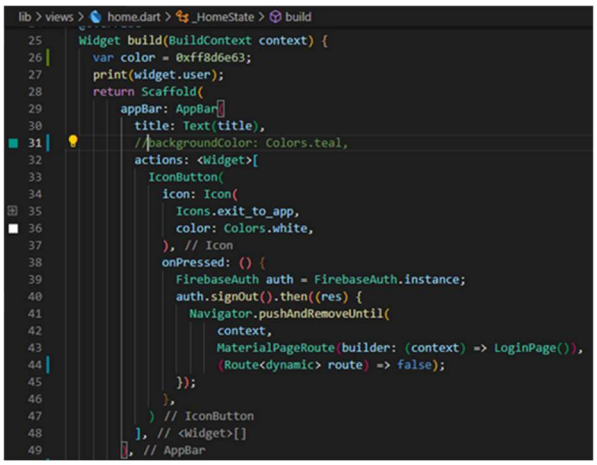
Figure 12. Tracking Interface.

4.2 Testing Phase Result

Functionality testing is used to ensure that a software programme meets its design criteria and works as expected. Inserting the input and checking the output data of the device tracker will be used to evaluate the project's functionalities. This test is performed to check that the application meets all of the criteria. This sort of testing is required and should be performed regardless of the type of application.

4.2.1 Mobile Application

Figure 13 below shows that one of the coding for buttons and app bar or navbar that contain home button, back button and sign out button. Also same goes to other screens, such as the profile screen, location screen, and home screen, which all include a card widget that may be clicked. As a result, when the user hits any of the mobile application buttons, such as the sign-in, sign-out, register, and other buttons, they all operate properly.



```
lib > views > home.dart > _HomeState > build
25  Widget build(BuildContext context) {
26    var color = 0xff8d6e63;
27    print(widget.user);
28    return Scaffold(
29        appBar: AppBar(
30          title: Text(title),
31          //backgroundColor: Colors.teal,
32          actions: <Widget>[
33            IconButton(
34              icon: Icon(
35                Icons.exit_to_app,
36                color: Colors.white,
37              ), // Icon
38              onPressed: () {
39                FirebaseAuth auth = FirebaseAuth.instance;
40                auth.signOut().then((res) {
41                  Navigator.pushAndRemoveUntil(
42                      context,
43                      MaterialPageRoute(builder: (context) => LoginPage()),
44                      (Route<dynamic> route) => false);
45                });
46              },
47            ) // IconButton
48          ], // <Widget>[]
49        ), // AppBar
```

Figure 13. Coding for Mobile Application Feature.

4.2.2 GPS Module and Location Detection

```
41 String path = "/gps-data/coordinates";
42 void loop(){
43    if (ss.available() > 0){
44     if(gps.encode(ss.read()))
45     {
46       Serial.print(F("Location: "));
47       if (gps.location.isValid())
48       {
49
50         latitude = gps.location.lat();
51         longitude = gps.location.lng();
52       }
53       Serial.print(latitude,6);
54       Serial.print ("   ");
55       Serial.print(longitude,6);
56       Serial.println();
57       Firebase.setFloat(firebaseData, path + "/latitude", latitude);
58       Firebase.setFloat(firebaseData, path + "/longitude", longitude);
59
60       delay(100);
61     }
62    }
63 }
```

Figure 14. Coding for GPS.

On the Arduino IDE Software, Figure 14 illustrates the code for a GPS module to identify the location of the luggage. The GPS capability is programmed using the C programming language. If valid location data is received, it will show the coordinates or a GPS reading message on the serial monitor. Converts decimal degrees to degrees, minutes, and seconds, and displays position information in latitude and longitude on a serial monitor.

The loop () function will provide enough time for the location data to be read. While data is accessible on a serial connection, encode data read from GPS. Encode is used to parse the string received by the GPS and store it in the buffer in order to retrieve information from it. The data from NodeMCU will be push to Firebase Realtime Database.

Figure 15 below shows the output of the device tracker after the program code for this project has been successfully uploaded. It shows the coordinates of the device tracker in 6 decimal places. But before that, the microcontroller needs to connect to Wi-Fi first and it will be able to detect location. The GPS locate or measure the position and display the location in one second before it updates new coordinates.

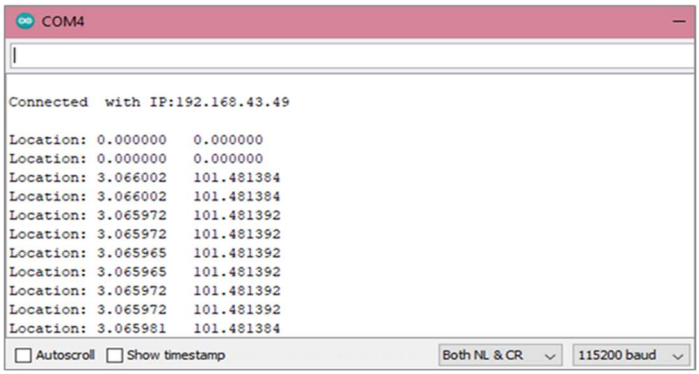


Figure 15. Coordinates of the Detected Location on Serial Monitor.

4.2.3 Firebase Realtime Database

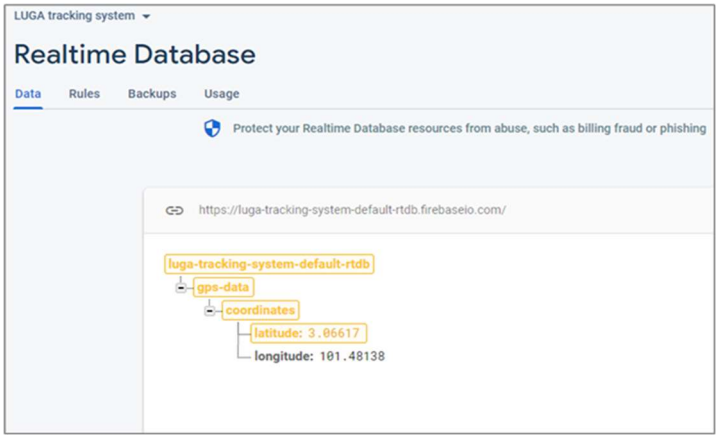


Figure 16. Firebase Realtime Database.

When the device tracker is turned on and the GPS receiver is successfully connected to satellites, the device tracker is ready to use. As illustrated in Figure 16, it will be able to identify the device's position and process the data into the Firebase Realtime Database. The information is based on real-time latitude and longitude data received from satellites. Until the device tracker is turned off, the data will continue to update its latitude and longitude. Even if the user goes to a different place, the latest location can still be updated. This information will be displayed in the Location Screen mobile app.

4.3 Project Limitation

The limitation of this project is as below:

- The device tracker can only send data to a database that is connected to the internet or via Wi-Fi.
- Because the GPS can only identify locations to a 3-meter precision, it is not particularly accurate.
- The system can only identify location when an internet connection is available.

5. Conclusion and Future Recommendation

5.1 Conclusion

The development of luggage tracking Mobile application system using flutter framework has successfully completed in order to achieve its objective. This proposed project presented a device tracking system with and Android application via GPS module and NodeMCU. The purpose of this project is to develop a tracking device which is use to for people who get frustrated while losing or misplacing their luggage.

The proposed system's tracking device is a basic device model that describes about tracking a luggage that interface with Android mobile devices to locate the luggage using android application with the help of microcontroller and Global Positioning System (GPS) module. NodeMCU helps to transmit the latitude and longitude coordinated that tracked via GPS module and acts as an input medium. The received signals will be transmitted to the android application through input message and the luggage will be tracked. Experimentation has been done and map have been created with Google Map APIs in order to track the location of the bags which are misplaced and lost. The results show that the Mobile Application Luggage Tracking System with GPS and NodeMCU is a success

5.2 Future Recommendations

In the future, the study should broaden its research scope to include the GPS module in order to improve accuracy while attempting to locate luggage. Although this system has evolved and provided some functionality to the user, it still needs to enhance the functionality of the mobile application and the device tracker. There are a few recommendations for future works, including specifying the address of the location on the Google Map so that the user can easily find their luggage and recording a location history where the user can monitor their pervious history of luggage location.

Acknowledgement

The authors wish to thank College of Computing, Informatics and Media (KPPIM) of UiTM Shah Alam to the giving the avenue to publish this final year project.

Funding

The authors received no specific funding for this work.

Author Contribution

Author1 provided the development of the LUGA system and draft the paper. Author2 supervised the development and oversaw the overall article writing.

Conflict of Interest

The authors have no conflicts of interest to declare.

References

Adjei, H. A., Oduro-Gyimah, F. K., Shunhua, T., Agordzo, G. K., & Musariri, M. (2020). Developing a Bluetooth Based Tracking System for Tracking Devices Using Arduino. In 2020 5th International Conference on Computing, Communication and Security (ICCCS) (pp. 1-5). IEEE.

Demir, M. Ö., Kurt, G. K., & Pusane, A. E. (2020). On the Limitations of GPS Time-Spoofing Attacks. In 2020 43rd International Conference on Telecommunications and Signal Processing (TSP) (pp. 313-316). IEEE.

Devaki, Poonkodi & Karthika, M. (2019). IoT Based Smart Object Tracking System. International Research Journal of Engineering and Technology (IRJET), 929-932.

Ghazal, M., Ali, S., Haneefa, F., & Sweleh, A. (2016). Towards Smart Wearable Real-Time Airport Luggage Tracking. In 2016 International Conference on Industrial Informatics and Computer Systems (CIICS) (pp. 1-6). IEEE.

Goldsten, M. (2017, July 6). How To Protect Yourself from Rising Airport Luggage Theft. A post at Forbes available at https://www.forbes.com/sites/michaelgoldstein/2017/07/06/how-to-protect-yourself-from-rising-airport-luggage-theft

Karthick, S., Joel, J., Balaji, S., & Anish, T. P. (2020). Smart Luggage Tracking and Alert System using Arduino. International Research Journal of Modernization in Engineering, Technology and Science, 2(05), 811-816.

Karvinsky. (2020, April 16). Tracking Based on GPS and the "Internet of Things". A post at IoT Business news available at https://iotbusinessnews.com/2020/04/16/54510-tracking-based-on-gps-and-the-internet-of-things/

Newton, E. (2021, December 21). Airways Magazine: Airports. A post at Airways Magazine available at https://airwaysmag.com/airports/iot-airport-baggage-systems/

NOAA. (2020, September 4). What is GPS? A post at National Ocean Service available at https://oceanservice.noaa.gov/facts/gps.html

Purwar, A., Divya, J., & Chaubey, V. K. (2016). GPS Signal Jamming and Anti-Jamming Strategy - A Theoretical Analysis. In 2016 IEEE Annual India Conference (INDICON) (pp. 1-6). IEEE.

Safdar, S., Zeb, A., Khan, A., & Zeeshan, K. (2018). Android Based Vehicle Tracking System. EAI Endorsed Transactions on Energy Web, 5(17), 1-6. https://doi.org/10.4108/eai.10-4-2018.154447.

Senthilkumar, S., & Jain, M. (2017). Luggage Tracking System using IoT. International Journal of Pure and Applied Mathematics, 117(1), 49-55.

Shobayo, O., Olajube, A., Okoyeigbo, O., & Ogbonna, J. (2021). Design and Implementation of an IoT Based Baggage Tracking System. In Information and Communication Technology and Applications: Third International Conference, ICTA 2020 (pp. 618-631). Springer International Publishing.

Sinton, D. (2018, September 19). GPS Jamming and Spoofing: When Good Signals Go Bad. A post at Directions Magazine available at https://www.directionsmag.com/article/8107

Technology, S. S. (2021, June 8). Syrma SGS Technology. A post at Syrma SGS Corporate available at https://www.syrma.com/luggage/

Venkatesan, M., Anbarasu, A., Balajikrishnan, G. S., Harishbabu, A., & M., J. (2021). An Intelligent Luggage Tracking System using GSM and GPS. International Research Journal of Engineering and Technology (IRJET), 08, 39-41.

Wikipedia. (2020, December 15). Global Positioning System. A post at Wikipedia available at https://en.wikipedia.org/wiki/Global_Positioning_System