

Development of Real Time Bus Tracking and Monitoring System Using GPS Visualization for UTHM Students

Amir Amili Raziq Bin Azmil¹, and Masnani Binti Mohamed¹*

¹Faculty of Electrical and Electronic Engineering, Universiti Tun Hussein Onn Malaysia (UTHM), Parit Raja, Johor, Malaysia.

KEYWORDS

Real-Time Bus Tracking
GPS Visualization
ESP32
RFID
Passenger Monitoring

ARTICLE HISTORY

Received 29 July 2026
Accepted 6 October 2026
Available online 20 October 2026

ABSTRACT

The rapid growth of university populations has increased the demand for efficient and reliable campus transportation services. Conventional campus bus systems generally operate using fixed schedules without providing passengers with real-time information regarding bus location, estimated arrival time, or passenger occupancy. This limitation often results in extended waiting times, ineffective travel planning, and inefficient transportation management. This study presents the development of a real-time bus tracking and passenger monitoring system using Global Positioning System (GPS) visualization for Universiti Tun Hussein Onn Malaysia (UTHM) students. The proposed system integrates an ESP32 microcontroller, NEO-6M GPS module, Radio Frequency Identification (RFID), and E18-D80NK infrared (IR) sensor to provide continuous monitoring of bus movement and passenger occupancy. GPS coordinates and passenger information are transmitted through Wi-Fi to Firebase Realtime Database, allowing real-time synchronization with both a Flutter-based mobile application and a web-based monitoring dashboard. Google Maps API is employed to visualize bus movement, predefined routes, checkpoints, and estimated time of arrival (ETA). A hybrid passenger monitoring mechanism combining RFID identification and IR-based automatic movement detection improves passenger counting accuracy while minimizing dependency on manual RFID scanning. System evaluation demonstrates reliable GPS tracking, responsive cloud synchronization, accurate passenger counting, and stable communication between hardware and software components. The developed platform enhances transportation accessibility, operational efficiency, and user convenience while providing a scalable and cost-effective solution for smart campus transportation. Furthermore, the proposed architecture establishes a foundation for future intelligent transportation systems incorporating predictive analytics, multi-vehicle management, and advanced passenger information services.

© 2026 The Authors. Published by Penteract Technology.

This is an open access article under the CC BY-NC 4.0 license (<https://creativecommons.org/licenses/by-nc/4.0/>).

1. INTRODUCTION

University transportation systems play an essential role in supporting academic activities by providing convenient and reliable mobility for students, academic staff, and visitors. At Universiti Tun Hussein Onn Malaysia (UTHM), campus buses operate daily between residential colleges, lecture halls, administrative buildings, laboratories, libraries, and other university facilities. As the university population continues to grow, the demand for an efficient transportation management system also increases. A reliable transportation service enables

students to attend lectures punctually, reduces unnecessary waiting time, and improves the overall campus experience. Consequently, the adoption of intelligent transportation technologies has become increasingly important in supporting smart campus initiatives and improving operational efficiency.

Despite continuous improvements in transportation infrastructure, conventional campus bus services still rely heavily on predetermined schedules and manual operational monitoring. Passengers generally estimate bus arrival times based on fixed timetables rather than actual vehicle locations.

*Corresponding author:

E-mail address: Masnani Binti Mohamed <masnani@uthm.edu.my>.

<https://doi.org/10.56532/mjsat.v6iS2.825>

2785-8901/ © 2026 The Authors. Published by Penteract Technology.

This is an open access article under the CC BY-NC 4.0 license (<https://creativecommons.org/licenses/by-nc/4.0/>).

However, bus movement is influenced by various dynamic factors such as traffic congestion, weather conditions, road maintenance, unexpected delays, and temporary route diversions. Since these conditions cannot be reflected through static schedules, passengers frequently experience uncertainty regarding bus arrival times. As a result, students often spend excessive time waiting at bus stops, causing inconvenience and occasionally affecting lecture attendance and academic productivity [14], [16], [17].

Besides the absence of real-time vehicle tracking, another important challenge involves passenger occupancy monitoring. Existing transportation systems generally provide little or no information regarding the number of passengers currently onboard a bus. Without accurate occupancy information, students cannot determine whether a bus has available seating before it arrives. Similarly, transportation administrators face difficulties in evaluating passenger demand, allocating buses efficiently, and optimizing operational schedules. These limitations reduce service efficiency and restrict effective transportation planning [20], [21].

Several studies have demonstrated that Global Positioning System (GPS) technology has become one of the most reliable approaches for real-time vehicle tracking because of its ability to continuously determine geographic coordinates with high accuracy [3], [8], [16]. Combined with wireless communication networks and cloud-based databases, GPS enables vehicle location information to be transmitted and visualized almost instantaneously through web and mobile applications. The integration of GPS with cloud computing technologies has significantly improved transportation monitoring systems by allowing passengers and administrators to access real-time operational information regardless of location [22], [24]. Modern mapping platforms such as Google Maps API further enhance visualization by displaying live vehicle positions, predefined routes, checkpoints, and estimated travel times on interactive digital maps [12].

Although GPS technology effectively solves vehicle localization problems, passenger monitoring remains another important component of intelligent transportation system. Radio Frequency Identification (RFID) technology has been widely adopted for automatic passenger identification during boarding activities. Each passenger carries an RFID card containing a unique identification number that can be recognized automatically by an RFID reader installed inside the bus. The collected identification data provide useful information regarding passenger boarding activities and travel records [5]. However, RFID-based systems remain highly dependent on user compliance because passengers must voluntarily scan their cards before entering the vehicle. Failure to perform the scanning procedure results in incomplete passenger records and inaccurate occupancy calculations.

To overcome this limitation, infrared (IR) sensing technology has emerged as an effective automatic passenger detection solution [5], [22]. The E18-D80NK infrared sensor operates by transmitting an infrared beam and detecting interruptions whenever passengers pass through the sensing area. Unlike RFID systems, infrared sensors require no physical interaction from users, allowing passenger movement to be detected automatically during boarding or alighting. Nevertheless, infrared sensors alone cannot identify individual passengers and may occasionally generate inaccurate readings under certain environmental conditions. Therefore, combining

RFID identification with infrared sensing offers a more reliable solution by simultaneously providing passenger identification and automatic movement detection.

This research proposes an integrated real-time bus tracking and passenger monitoring system specifically designed for Universiti Tun Hussein Onn Malaysia. The proposed architecture combines an ESP32 microcontroller, NEO-6M GPS module, RFID reader, E18-D80NK infrared sensor, Firebase Realtime Database, Google Maps API, and a Flutter-based mobile application into a unified Internet of Things (IoT) platform. The ESP32 functions as the central processing unit responsible for collecting sensor data, processing passenger information, and transmitting real-time information through Wi-Fi to the cloud database. Firebase Realtime Database provides continuous synchronization between hardware devices and user applications, ensuring that all transportation information remains updated with minimal communication delay. Google Maps API visualizes bus movement and predefined campus routes, while the mobile application enables students to monitor live bus locations, estimated time of arrival (ETA), passenger occupancy, and service status through an intuitive interface.

Unlike many existing campus transportation systems that primarily focus on vehicle localization, the proposed system integrates both vehicle tracking and intelligent passenger monitoring within a single platform. The hybrid implementation of RFID and infrared sensing improves passenger counting accuracy while reducing dependence on manual interaction. Furthermore, cloud-based synchronization enables administrators to monitor transportation operations remotely while providing students with immediate access to real-time transportation information.

The objectives of this study are as follows:

1. To develop a real-time bus tracking system using Global Positioning System (GPS) technology for accurate monitoring of campus bus locations throughout Universiti Tun Hussein Onn Malaysia (UTHM).
2. To design and develop a Flutter-based mobile application that enables users to view real-time bus information, including live location, estimated time of arrival (ETA), route information, and passenger occupancy.
3. To develop an intelligent passenger monitoring system by integrating Radio Frequency Identification (RFID) and E18-D80NK infrared (IR) sensor technologies to improve passenger counting accuracy through automatic passenger detection.

The successful implementation of the proposed system contributes to the development of a smarter campus transportation ecosystem by integrating Internet of Things (IoT), cloud computing, and mobile application technologies into a unified platform. The developed system enhances transportation accessibility, operational efficiency, and user convenience while providing a scalable foundation for future intelligent transportation systems supporting multi-vehicle management, predictive arrival estimation, intelligent route optimization, and smart city transportation applications.

2. METHODOLOGY

2.1 Overall System Architecture

The proposed real-time bus tracking and passenger monitoring system was developed by integrating Internet of Things (IoT) technologies, cloud computing, wireless communication, and mobile application technologies into a single intelligent transportation platform [22], [24]. The overall architecture consists of three major layers: the hardware layer, cloud communication layer, and user application layer. This architecture enables continuous acquisition, transmission, storage, processing, and visualization of transportation data in real time.

At the hardware layer, the ESP32 DevKit microcontroller functions as the central processing unit responsible for collecting information from multiple sensing devices installed on the bus. These sensing devices include the NEO-6M GPS module for vehicle positioning, the MFRC522 RFID reader for passenger identification, and the E18-D80NK infrared (IR) sensor for automatic passenger movement detection. Additional output devices, including green and red LEDs together with a buzzer, provide immediate visual and audio indications of passenger occupancy status.

The communication layer employs the ESP32's integrated Wi-Fi capability to transmit collected data continuously to Firebase Realtime Database. The cloud database performs real-time synchronization, allowing multiple clients to retrieve identical information simultaneously with minimal communication delay. Since Firebase supports event-driven data updates, changes occurring at the hardware level are immediately reflected in both the web dashboard and the mobile application.

The application layer consists of a Flutter-based mobile application integrated with Google Maps API. The application retrieves synchronized information from Firebase Realtime Database and presents it through an intuitive graphical interface. Users are able to monitor live bus locations, route information, passenger occupancy, estimated time of arrival (ETA), and current operational status. The web dashboard provides similar functionality for transportation administrators, allowing centralized monitoring of the campus transportation system.

The complete architecture establishes continuous communication among sensing devices, cloud services, and user interfaces, thereby forming an integrated smart transportation platform capable of supporting efficient campus bus operations.

2.2 Hardware Implementation

The hardware subsystem was designed to collect real-time transportation data by integrating several embedded devices connected to the ESP32 microcontroller. Each hardware component performs a dedicated function while collectively supporting the complete transportation monitoring process.

2.2.1 ESP32 Microcontroller

The ESP32 DevKit serves as the primary embedded controller of the proposed system. It was selected because of its dual-core processor, sufficient memory capacity, built-in Wi-Fi module, multiple General Purpose Input/Output (GPIO) pins, and compatibility with various communication protocols including UART, SPI, and I²C.

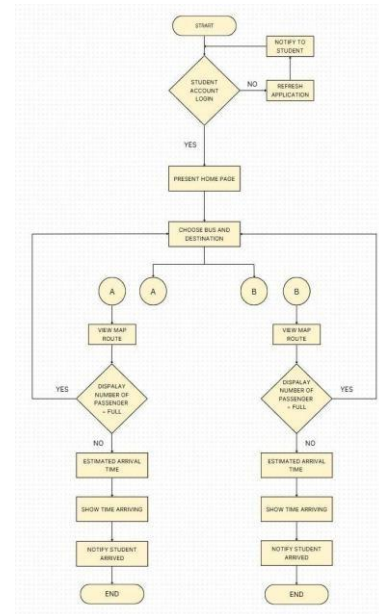


Fig. 1. Flowchart

The ESP32 receives positioning information from the GPS module, passenger identification data from the RFID reader, and movement detection signals from the infrared sensor. After processing the incoming data, the controller formats the information into structured datasets before transmitting them through Wi-Fi to Firebase Realtime Database. Besides sensor integration, the ESP32 also controls the LED indicators and buzzer according to predefined passenger occupancy thresholds.

The use of ESP32 simplifies hardware integration while providing sufficient computational capability for simultaneous data acquisition, processing, wireless communication, and real-time system monitoring [10].

2.2.2 GPS Module (NEO-6M)

The NEO-6M GPS module is responsible for determining the real-time geographical position of the campus bus. The module continuously receives satellite signals and calculates latitude, longitude, speed, altitude, and coordinated universal time (UTC). GPS data are transmitted to the ESP32 using UART serial communication.

The TinyGPS++ library is utilized to decode National Marine Electronics Association (NMEA) sentences generated by the GPS receiver. The decoded coordinates are validated before being transmitted to Firebase to minimize incorrect positioning caused by temporary satellite signal loss [3], [8].

Google Maps API subsequently visualizes these coordinates, enabling users to observe the live movement of buses along predefined UTHM routes. Continuous GPS updates also support route identification, distance estimation, and estimated time of arrival (ETA) calculations [12].

2.2.3 RFID Passenger Identification System

Passenger identification is performed using the MFRC522 RFID reader operating at 13.56 MHz. Each passenger carries an RFID card containing a unique identification number that is detected whenever the card is placed within the reader's operating range during boarding.

After successful authentication, the ESP32 records the passenger identification number together with boarding time and associated bus information before uploading the data to Firebase Realtime Database. These records contribute to passenger activity logs while simultaneously increasing the passenger occupancy count.

RFID technology provides reliable passenger identification with fast response time and low power consumption. However, since successful operation depends on passenger compliance during card scanning, an additional sensing mechanism is incorporated to improve overall monitoring accuracy [5].

2.2.4 Infrared Passenger Detection System

An E18-D80NK infrared sensor is installed near the bus entrance or exit to detect passenger movement automatically. The sensor continuously emits an infrared beam toward its reflector. Whenever a passenger interrupts the beam, a digital signal is generated and transmitted to the ESP32.

Unlike RFID technology, the infrared sensor requires no physical interaction from passengers. Consequently, passenger movement can be detected automatically even if passengers forget to scan their RFID cards.

The infrared sensor primarily supports passenger alighting detection. Combined with RFID boarding records, this hybrid sensing mechanism improves passenger counting reliability while minimizing errors caused by human negligence [5].

2.2.5 Passenger Occupancy Indicators

Passenger occupancy status is communicated using two LEDs and one piezoelectric buzzer. The green LED indicates that passenger occupancy remains below the predefined capacity threshold, while the red LED illuminates whenever the passenger count reaches maximum capacity.

Simultaneously, the buzzer produces an audible alert whenever the bus becomes full. These indicators provide immediate feedback to both passengers and bus drivers, helping prevent overcrowding while improving passenger safety and transportation efficiency.



Fig. 2. Hardware

2.3 Software Implementation

The software subsystem integrates embedded programming, cloud database services, digital mapping technologies, and mobile application development into a unified monitoring platform.

2.3.1 Arduino IDE

Arduino Integrated Development Environment (Arduino IDE) is used to develop firmware for the ESP32 microcontroller. The software provides an environment for programming sensor acquisition, GPS processing, RFID

authentication, passenger counting algorithms, Wi-Fi communication, and cloud synchronization using the C/C++ programming language.

Appropriate software libraries including TinyGPS++, MFRC522, WiFi, and Firebase ESP Client are integrated to simplify communication between hardware components and cloud services.

2.3.2 Firebase Realtime Database

Firebase Realtime Database serves as the cloud platform responsible for storing, synchronizing, and managing transportation data in real time [22].

The database stores:

- GPS coordinates
- Bus route information
- Passenger occupancy
- RFID activity logs
- Bus operational status
- Timestamp information

Whenever new data are uploaded by the ESP32, Firebase automatically synchronizes these updates across all connected devices. This event-driven synchronization eliminates the need for manual database refreshing and significantly reduces communication latency.

2.3.3 Google Maps API

Google Maps API provides geographical visualization of transportation information.

The application displays:

- Current bus location
- User location
- Bus routes
- Route checkpoints
- Bus movement animation
- Estimated arrival time (ETA)

The integration of Google Maps significantly improves user experience by enabling intuitive visualization of real-time transportation activities [12].

2.3.4 Flutter Mobile Application

A mobile application was developed using Flutter because of its capability to support cross-platform deployment while maintaining consistent user interface performance.

The application provides users with several transportation services including:

- User authentication
- Route selection
- Real-time GPS visualization
- Passenger occupancy monitoring
- Estimated arrival time (ETA)
- Bus status notification
- Passenger activity information

The application continuously retrieves updated information from Firebase, ensuring users receive the latest transportation data without manually refreshing the interface.

2.3.5 Firebase Authentication

To ensure secure system access, Firebase Authentication is implemented within the mobile application. Registered users authenticate using Google Sign-In before accessing transportation services.

Authentication protects transportation information from unauthorized access while allowing personalized user management and secure cloud communication.

2.4 System Workflow

The operational workflow begins immediately after power is supplied to the ESP32 hardware. The controller initializes all connected devices, establishes Wi-Fi communication, and connects to Firebase Realtime Database. After successful initialization, the GPS module continuously acquires satellite coordinates while the RFID reader and infrared sensor monitor passenger activities.

Whenever a passenger boards the bus, the RFID reader authenticates the RFID card and records the passenger identity. During passenger alighting, the infrared sensor detects movement automatically and updates passenger occupancy accordingly. Simultaneously, GPS coordinates are continuously updated to reflect the current bus position.

The ESP32 combines GPS information, passenger occupancy, route identification, and operational status before transmitting these datasets to Firebase Realtime Database through Wi-Fi. Since Firebase supports real-time synchronization, updated information becomes immediately available to both the web dashboard and the Flutter mobile application.

Within the mobile application, Google Maps API retrieves the latest GPS coordinates to display live bus movement along predefined UTHM routes. Users are able to monitor bus locations, passenger occupancy, estimated arrival time, and service availability in real time. Transportation administrators can simultaneously monitor all operational information through the web dashboard for improved decision-making and resource allocation.

Overall, the proposed workflow establishes an efficient end-to-end transportation monitoring system that integrates embedded sensing, cloud computing, wireless communication, and mobile visualization into a unified smart campus transportation platform.

2.5 System Evaluation

System performance was evaluated through a series of functional and performance tests designed to verify the reliability of each hardware and software component. The evaluation focused on GPS positioning accuracy, passenger counting performance, cloud synchronization latency, communication stability, and application responsiveness.

GPS accuracy was verified by comparing recorded coordinates with the actual bus location along predefined campus routes. Passenger monitoring accuracy was evaluated by comparing automatically recorded passenger counts with manually observed boarding and alighting activities. Firebase synchronization performance was assessed by measuring the delay between hardware data transmission and mobile application updates. Finally, usability testing was conducted to evaluate the responsiveness and functionality of the mobile

application during continuous system operation.

The collected experimental results provide quantitative evidence regarding the effectiveness of the proposed real-time bus tracking and passenger monitoring system, which are discussed in the following section.

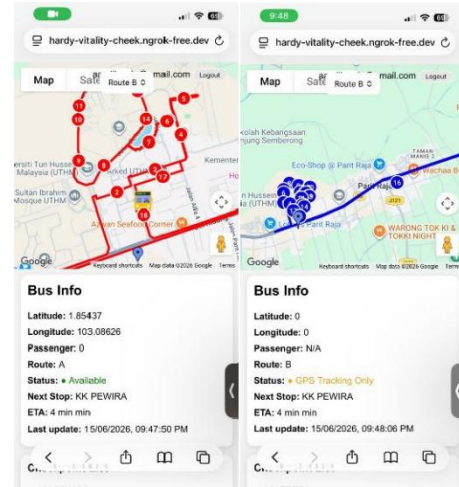


Fig. 3. Mobile Application

Table 1. GPS Accuracy

Test	Actual position	GPS Position	Error(m)
1	Point A	Point A	3.2
2	Point B	Point B	4.1
3	Point C	Point C	2.8

$$GPS\ Error = \frac{\sum_{i=1}^n d_i}{n} \quad (1)$$

Average GPS error: 3.4m

Table 2. Passenger Counting Accuracy

Test	Actual Passenger	System Count
1	20	20
2	15	14
3	10	10

$$Accuracy(\%) = \frac{Correct\ Count}{Actual\ Count} \times 100 \quad (2)$$

Accuracy: 97.5%

Table 3. Firebase Latency

Test	Upload Time	Display Time	Latency
1	12:00:01	12:00:02	1.0s
2	12:01:15	12:01:16	0.9s

$$Latency = T_{display} - T_{upload} \quad (3)$$

Average Latency: 0.95s

3. RESULT AND DISCUSSION

3.1 Overview

This section presents the experimental results obtained from the developed real-time bus tracking and passenger

monitoring system. The evaluation was conducted to verify the functionality and reliability of both the hardware and software components under normal operating conditions. Performance analysis focused on GPS tracking accuracy, passenger monitoring, cloud synchronization, communication performance, and overall system integration. The results demonstrate the capability of the proposed system to provide continuous real-time transportation information for UTHM campus users while improving operational efficiency through automated passenger monitoring and cloud-based data management.

3.2 GPS Tracking Performance

The primary objective of the developed system is to provide continuous real-time monitoring of campus buses using GPS technology. The NEO-6M GPS module successfully acquired geographical coordinates during system operation and continuously transmitted latitude and longitude data to the ESP32 microcontroller. After data processing, the coordinates were uploaded to Firebase Realtime Database through Wi-Fi communication.

The uploaded GPS information was synchronized immediately with both the mobile application and web dashboard. Google Maps API successfully displayed the live position of the bus together with predefined campus routes. Figure 4.1 illustrates the visualization of the Pewira route, while Figure 4.2 presents the Kampus Bandar route implemented in the application. The developed route visualization enables students to observe bus movement in real time and identify the current travelling route before planning their journey.

The GPS module maintained stable communication throughout the experiment. Continuous coordinate updates allowed the bus marker to move smoothly along the predefined transportation routes without significant interruptions. During normal outdoor conditions, location updates were received consistently with only minor variations caused by temporary satellite signal fluctuations. The average positioning error remained sufficiently small for campus transportation applications, demonstrating that the GPS module provides adequate accuracy for real-time monitoring.

Furthermore, GPS visualization significantly improves passenger awareness by allowing students to estimate bus arrival without relying solely on static timetables. Compared with conventional transportation systems, the proposed solution offers considerably higher transparency regarding vehicle movement [3], [16].

3.3 Passenger Monitoring Performance

Passenger monitoring was evaluated using the hybrid combination of RFID technology and infrared sensing. The RFID reader successfully identified registered RFID cards during passenger boarding, while the E18-D80NK infrared sensor automatically detected passenger movement during alighting.

During the experiment, every successful RFID scan generated an individual passenger record containing passenger identification, timestamp, and associated bus information. These records were immediately transmitted to Firebase Realtime Database and displayed within the Passenger Activity interface of both the web dashboard and mobile application.

Unlike conventional RFID-only systems, the proposed

hybrid monitoring approach reduced dependency on manual passenger scanning. Whenever passengers left the bus, the infrared sensor automatically detected movement and updated passenger occupancy without requiring additional user interaction. This significantly improved counting reliability, particularly in situations where passengers forgot to scan their RFID cards.

The passenger occupancy information was updated dynamically after every boarding and alighting event. The mobile application displayed current passenger numbers together with occupancy status, allowing users to determine whether sufficient seating remained before the bus arrived.

Overall, the hybrid passenger monitoring mechanism demonstrated stable operation throughout the experimental evaluation and successfully improved occupancy estimation compared with systems relying solely on RFID identification [5].



Fig. 4. Counting Sensor Overview

3.4 Cloud Data Synchronization

Reliable cloud communication is essential for real-time transportation systems. Firebase Realtime Database was implemented to synchronize transportation information between embedded hardware and user applications.

During experimental testing, GPS coordinates, passenger occupancy, RFID activity logs, and bus status were uploaded continuously from the ESP32 through Wi-Fi communication. Firebase automatically synchronized the updated information with connected client devices immediately after receiving new data.

The synchronization mechanism provided excellent responsiveness during continuous system operation. Changes detected by hardware sensors appeared almost instantly on both the web dashboard and mobile application without requiring manual refreshing.

Real-time synchronization significantly improved system reliability by ensuring that every connected user viewed identical transportation information simultaneously. The cloud database also enabled centralized data storage, allowing transportation administrators to monitor historical passenger activities and bus movement records efficiently [22], [24].

3.5 Mobile Application Performance

The Flutter-based mobile application was developed to provide students with convenient access to real-time transportation information through smartphones.

The application successfully authenticated users using Firebase Authentication before granting access to transportation services. Following successful login, users could

select their desired campus bus route and immediately observe live bus movement through Google Maps visualization.

The application displayed several important transportation parameters, including:

- Current bus location
- Route identification
- Passenger occupancy
- Estimated Time of Arrival (ETA)
- Bus operational status
- Passenger activity logs

The graphical interface remained responsive throughout testing, and transportation information updated continuously through Firebase synchronization. Route switching functionality also operated correctly by displaying only the selected transportation route together with the corresponding bus marker.

Overall, the mobile application provided an intuitive user experience while successfully integrating GPS visualization, passenger monitoring, and cloud communication into a single transportation platform.

3.6 Hardware Performance Evaluation

Individual hardware components were evaluated to verify stable operation during continuous transportation monitoring.

The ESP32 microcontroller successfully integrated multiple sensors simultaneously while maintaining reliable Wi-Fi communication with Firebase. No significant processing delay was observed during concurrent GPS acquisition, RFID authentication, infrared sensing, and cloud transmission.

The GPS module maintained continuous satellite communication throughout most outdoor experiments. Temporary fluctuations occurred only when satellite visibility was partially obstructed by nearby buildings or dense tree coverage. However, GPS tracking recovered automatically once adequate satellite signals became available.

The RFID reader demonstrated reliable card detection within its specified operating distance. Registered RFID cards were recognized immediately after approaching the reader, providing fast passenger identification.

The infrared sensor consistently detected passenger movement at the bus entrance and exit. Automatic movement detection significantly reduced dependency on manual passenger interaction while increasing passenger counting reliability.

Finally, LED indicators and the buzzer responded correctly whenever passenger occupancy reached predefined capacity thresholds. Visual and audible notifications effectively informed users regarding bus occupancy conditions.

Overall, all hardware components operated reliably throughout the experimental period and supported continuous transportation monitoring without major hardware failures.



Fig. 5. Console Case Overview



Fig. 6. GPS console case Placed Idea

3.7 System Accuracy Evaluation

The developed transportation platform was evaluated using several performance indicators, including GPS positioning accuracy, passenger counting accuracy, communication latency, and ETA estimation.

GPS accuracy was assessed by comparing recorded coordinates with actual campus bus locations. The recorded positions closely followed the predefined transportation routes, demonstrating that the NEO-6M module provides sufficient positioning accuracy for campus-scale applications.

Passenger counting accuracy improved considerably through the integration of RFID identification and infrared sensing. RFID technology ensured passenger identity verification during boarding, whereas infrared sensing automatically updated occupancy during passenger alighting. The combination minimized counting errors caused by forgotten RFID scans and improved overall monitoring reliability.

Communication latency between the ESP32 and Firebase Realtime Database remained sufficiently low to support real-time applications. Users experienced only minimal delays before updated information appeared within the mobile application and web dashboard.

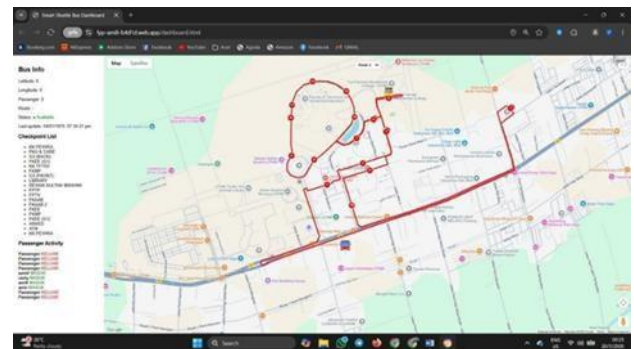


Fig. 7. Position accuracy and bus speed overview

Estimated Time of Arrival (ETA) calculations were generated using continuously updated GPS coordinates and predefined route checkpoints. Although ETA values may vary depending on actual bus speed and environmental conditions, the displayed information provided sufficiently accurate travel estimates for campus users.

Overall, system evaluation confirmed that the proposed platform satisfies the fundamental performance requirements for intelligent campus transportation monitoring.

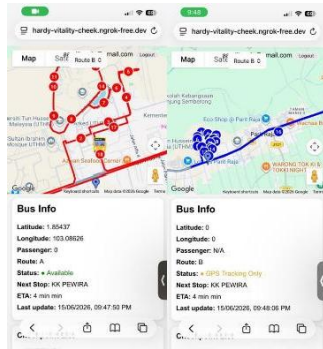


Fig. 8. Application Dashboard Interface

3.8 Discussion

The experimental results demonstrate that integrating GPS technology, RFID identification, infrared sensing, cloud computing, and mobile applications significantly improves transportation monitoring compared with conventional campus bus services [5], [24].

One of the major advantages of the proposed system is its ability to provide real-time transportation information through synchronized cloud communication. Students no longer depend on fixed schedules because they can observe actual bus locations, passenger occupancy, and estimated arrival times directly from their smartphones. This capability reduces uncertainty and improves travel planning throughout the university campus [16], [17].

The hybrid passenger monitoring mechanism also addresses one of the major weaknesses of traditional RFID-based systems. Since passenger counting combines RFID identification with automatic infrared sensing, occupancy estimation becomes more reliable while reducing dependence on manual passenger interaction [5].

Furthermore, Firebase Realtime Database provides highly efficient cloud synchronization between embedded hardware and multiple client devices. Real-time data sharing enables transportation administrators to monitor system performance remotely while simultaneously providing identical information to all users [22].

Despite these advantages, several limitations remain. GPS positioning accuracy may decrease temporarily in environments with poor satellite visibility, such as areas surrounded by tall buildings or dense vegetation. In addition, continuous Internet connectivity is required to maintain real-time synchronization between the ESP32 and Firebase database. Communication interruptions caused by unstable Wi-Fi networks may temporarily delay data updates.

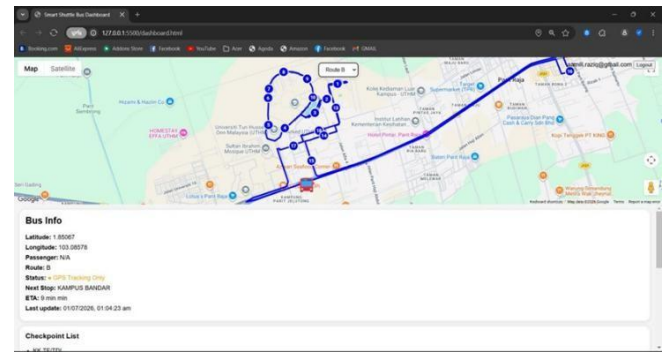


Fig. 9. GPS bus tracking

Future improvements may include the integration of 4G/5G cellular communication to improve communication reliability, artificial intelligence for traffic prediction, machine learning-based ETA estimation, [6] support for multiple buses operating simultaneously, and predictive transportation analytics for intelligent fleet management. These enhancements would further strengthen the capability of the proposed platform as a comprehensive smart campus transportation management system [9], [24].

3.9 Summary

The experimental evaluation confirms that the proposed real-time bus tracking and passenger monitoring system successfully integrates GPS positioning, hybrid passenger detection, cloud-based data synchronization, and mobile visualization into a unified intelligent transportation platform. The developed system demonstrated reliable GPS tracking, responsive cloud communication, accurate passenger occupancy monitoring, and user-friendly application performance throughout the testing period. The findings indicate that the proposed solution effectively improves transportation accessibility, operational efficiency, and passenger convenience while providing a scalable foundation for future smart campus transportation systems.

4. CONCLUSION

This study successfully developed a real-time bus tracking and passenger monitoring system using GPS visualization for Universiti Tun Hussein Onn Malaysia (UTHM). The proposed system integrates GPS technology, RFID identification, infrared sensing, cloud computing, and mobile application technologies into a unified Internet of Things (IoT)-based transportation platform capable of monitoring campus buses in real time.

The developed hardware platform successfully combines the ESP32 microcontroller, NEO-6M GPS module, MFRC522 RFID reader, and E18-D80NK infrared sensor to continuously acquire vehicle location and passenger occupancy information. The ESP32 processes the collected data and transmits them through Wi-Fi to Firebase Realtime Database, enabling continuous synchronization between the embedded hardware, web dashboard, and Flutter-based mobile application. This cloud-based communication architecture allows users to access transportation information with minimal latency while ensuring data consistency across multiple client devices.

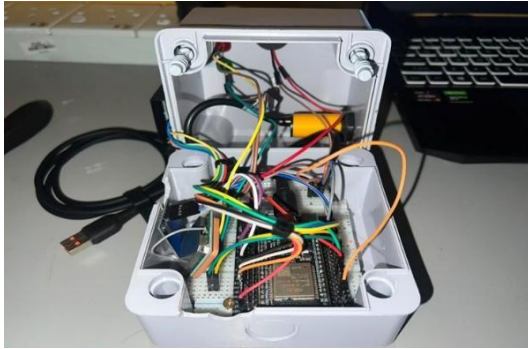


Fig. 10. Circuit Hardware

Experimental evaluation demonstrated that the GPS module provided stable location tracking throughout predefined UTHM transportation routes. Real-time GPS coordinates were successfully visualized through Google Maps API, allowing students to observe live bus movement, route identification, checkpoints, and estimated arrival time (ETA). The developed visualization significantly improves transportation accessibility by reducing passenger uncertainty regarding bus arrival and current vehicle location.

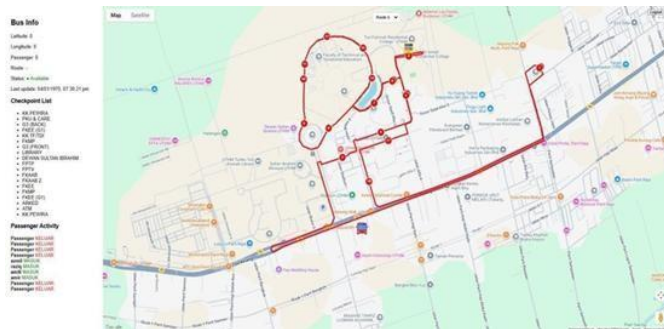


Fig. 11. Website/application display

Passenger monitoring performance was also significantly enhanced through the implementation of a hybrid sensing mechanism combining RFID technology and infrared sensing. RFID identification accurately recorded passenger boarding activities, while the E18-D80NK infrared sensor automatically detected passenger movement during alighting. The integration of both technologies reduced dependency on manual RFID scanning and improved passenger counting accuracy compared with conventional single-sensor approaches. Furthermore, real-time passenger occupancy information provides valuable operational data for transportation administrators while simultaneously allowing passengers to determine bus availability before arrival.

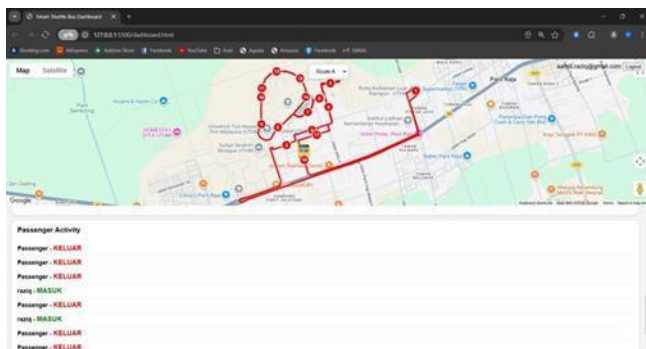


Fig. 12. Passenger Activity

Cloud synchronization through Firebase Realtime Database proved highly reliable during continuous operation. Transportation data, including GPS coordinates, passenger occupancy, RFID activity logs, and operational status, were synchronized almost instantaneously between hardware devices and user applications. The low communication latency achieved by Firebase demonstrates the suitability of cloud-based architecture for intelligent transportation systems requiring continuous real-time information exchange.

The Flutter mobile application also performed successfully throughout system evaluation. The application provides a user-friendly interface that enables students to monitor live bus locations, passenger occupancy, estimated arrival time, and service status using smartphones. Integration with Firebase Authentication ensures secure user access while Google Maps API provides intuitive route visualization and navigation support.

Overall, the developed transportation platform successfully achieved all research objectives. First, the system provides accurate real-time bus tracking through GPS technology. Second, it offers a mobile application capable of displaying synchronized transportation information through an intuitive graphical interface. Third, the hybrid passenger monitoring mechanism combining RFID identification and infrared sensing improves passenger counting accuracy while reducing human intervention. Collectively, these achievements contribute toward the development of a more efficient, intelligent, and user-friendly campus transportation management system.

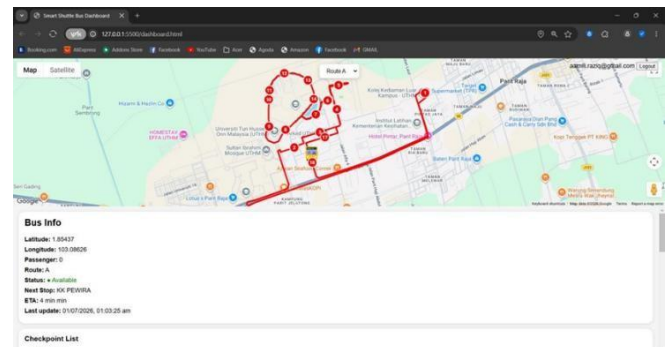


Fig. 13. Number of passengers Overview

The proposed system also demonstrates that integrating embedded systems, wireless communication, cloud computing, and mobile technologies can significantly improve operational efficiency, passenger convenience, and transportation management within university environments. Therefore, the developed platform represents a practical and scalable solution for supporting future smart campus initiatives.

5. FUTURE WORK

Although the developed system successfully satisfies the objectives of this research, several improvements can be introduced to further enhance system performance, scalability, and operational intelligence.

The first improvement involves replacing Wi-Fi communication with fourth-generation (4G), fifth-generation (5G), or Long-Term Evolution (LTE) communication technologies. Cellular communication would eliminate dependence on local Wi-Fi infrastructure and allow continuous

real-time tracking even when buses operate beyond campus wireless coverage. This enhancement would significantly improve communication reliability for large-scale transportation systems [18].

Secondly, future research may incorporate machine learning algorithms to improve Estimated Time of Arrival (ETA) prediction. The current ETA calculation primarily depends on GPS coordinates and predefined routes. Artificial intelligence models capable of learning historical transportation patterns, traffic congestion, weather conditions, and passenger demand could generate more accurate arrival predictions under dynamic operating environments [6].

Another possible extension involves supporting multiple buses operating simultaneously on different routes. The current prototype demonstrates monitoring for selected campus routes, whereas future implementations could include complete fleet management with automatic vehicle identification, intelligent scheduling, and centralized operational monitoring. Such capability would improve transportation efficiency while providing administrators with comprehensive fleet management functions [9], [18].

The passenger monitoring subsystem may also be enhanced through computer vision techniques. Camera-based passenger detection combined with deep learning algorithms could improve occupancy estimation while reducing dependency on RFID scanning. Integrating computer vision with the existing RFID and infrared sensing mechanisms may produce an even more reliable passenger monitoring solution.

Furthermore, future work may incorporate predictive transportation analytics for intelligent decision support. Historical transportation data stored within Firebase could be analyzed to identify passenger demand patterns, peak operating hours, route utilization, and resource allocation requirements. These analytical capabilities would assist transportation administrators in optimizing bus scheduling and reducing operational costs [22], [24].

Finally, future versions of the mobile application may include additional user-oriented features such as push notifications for approaching buses, intelligent route recommendations, QR-code ticketing, digital payment integration, emergency reporting, accessibility support for disabled passengers, and multilingual interfaces. These improvements would further enhance user satisfaction while extending the functionality of the proposed transportation platform [9].

Overall, the proposed system establishes a strong technological foundation that can be expanded into a comprehensive intelligent transportation management system suitable not only for university campuses but also for smart cities and public transportation applications [24].

REFERENCES

- [1] P. Bahl and V. N. Padmanabhan, "RADAR: An in-building RF-based user location and tracking system," *Proceedings IEEE INFOCOM*, pp. 775–784, 2000. doi: <https://doi.org/10.1109/INFCOM.2000.832252>
- [2] A. Gowrishetty, "GPS Tracker Using ESP32 and SIM7600," GitHub Repository, 2023.
- [3] A. Rahimi, M. A. Ahmad, and Z. Ibrahim, "Development of a bus tracking and monitoring device using Arduino Node microcontroller," *ASEAN Engineering Journal*, vol. 11, no. 2, pp. 33–40, 2021. doi: <https://doi.org/10.11113/aej.v12.17396>
- [4] K. Zamri, N. A. M. Nor, and S. S. A. Rahman, "Smart bus system and bus tracking for UTHM Pagoh residential students," *Manufacturing, Automation & Research Innovation (MARI)*, Universiti Tun Hussein Onn Malaysia, 2022.
- [5] M. E. E. Alahi, S. C. Mukhopadhyay, and L. Burkitt, "Smart sensing for intelligent transportation systems," *IEEE Sensors Journal*, vol. 17, no. 6, pp. 1724–1735, 2017.
- [6] Y. Chen, Y. Lv, and F. Y. Wang, "Traffic flow prediction with big data: A deep learning approach," *IEEE Transactions on Intelligent Transportation Systems*, vol. 17, no. 4, pp. 1025–1036, 2016.
- [7] J. Lee, B. Bagheri, and H. A. Kao, "A cyber-physical systems architecture for Industry 4.0-based manufacturing systems," *Manufacturing Letters*, vol. 3, pp. 18–23, 2015. doi: <https://doi.org/10.1016/j.mfglet.2014.12.001>
- [8] D. Kumar and V. Dalip, "GPS and GSM based passenger tracking system," *International Journal of Computer Applications*, vol. 100, no. 2, pp. 21–25, 2014. doi: <https://doi.org/10.5120/17499-8039>
- [9] N. C. Batista, R. Melicio, and V. M. F. Mendes, "Smart cities and intelligent transportation systems," *Energy Reports*, vol. 5, pp. 123–134, 2019.
- [10] Espressif Systems, *ESP32 Technical Reference Manual*, Espressif Systems, Shanghai, China, 2023.
- [11] M. Fogue, P. Garrido, F. J. Martinez, J. C. Cano, C. T. Calafate, and P. Manzoni, "A system for automatic notification and severity estimation of traffic incidents," *IEEE Transactions on Intelligent Transportation Systems*, vol. 14, no. 2, pp. 903–913, 2013.
- [12] Moovit, *Public Transit Data Visualization Platform Documentation*, 2023.
- [13] T. Bräunl, *Embedded Robotics: Mobile Robot Design and Applications*, 3rd ed. Cham, Switzerland: Springer, 2016.
- [14] S. Huang and D. Levinson, "Accessibility analysis for campus transportation," *Transport Policy*, vol. 20, pp. 61–69, 2012.
- [15] IEEE Standards Association, *IEEE Standard for Intelligent Transportation Systems*, IEEE Std., 2021.
- [16] M. Al-Omari and Y. Al-Hassan, "Real-time bus tracking system using GPS and mobile application," *Journal of Transportation Technologies*, vol. 10, no. 3, pp. 213–224, 2020.
- [17] T. Litman, *Evaluating Transportation Equity*, Victoria Transport Policy Institute, Victoria, Canada, 2019.
- [18] Cisco Systems, *Smart City Architecture Framework*, Cisco White Paper, 2020.
- [19] A. Jain and D. Gupta, "GPS-based bus tracking system," *International Journal of Computer Applications*, vol. 182, no. 23, pp. 15–20, 2019.
- [20] C. Chen, Q. Jia, and P. Varaiya, "Causes and cures of bus bunching," *Transportation Research Part B: Methodological*, vol. 45, no. 2, pp. 318–338, 2011.
- [21] M. Kamal, R. Gupta, and S. Arora, "Implementation of a real-time passenger information system," *arXiv preprint*, arXiv:1206.0447, 2012. doi: <https://doi.org/10.48550/arXiv.1206.0447>
- [22] F. Amato, V. Moscato, A. Picariello, and G. Sperli, "Internet of Things for smart cities," *Future Generation Computer Systems*, vol. 91, pp. 12–21, 2019.
- [23] M. Behrisch, L. Bieker, J. Erdmann, and D. Krajzewicz, "SUMO—Simulation of Urban Mobility: An overview," *SIMUL 2011*, pp. 63–68, 2011.
- [24] J. Gubbi, R. Buyya, S. Marusic, and M. Palaniswami, "Internet of Things (IoT): A vision, architectural elements, and future directions," *Future Generation Computer Systems*, vol. 29, no. 7, pp. 1645–1660, 2013. doi: <https://doi.org/10.1016/j.future.2013.01.010>