STM JOURNALS

E-Commerce for Future & Trends

https://journals.stmjournals.com/ecft/

ISSN: 2454-9347
Volume 13, Issue 1, 2026
January-April
DOI (Journal): 10.37591/ECFT

Review ECFT

RideRadar: Real-Time Bus Tracking and Updates

Sayali Shinde[1], Pradnya Koleshwar[1], Aadesh Deshmukh[2], Pranav Dawange[2], Alok Dhandar[2,*]

Abstract

Daily bus commuters know the frustration well, standing at stops with no idea when their ride will arrive, missing connections, and dealing with safety concerns that make public transport feel unreliable. This paper describes our prototype solution: a smart tracking system that tackles these everyday problems head-on. We built a system that connects real-time GPS data with direct passenger input, creating two-way communication between riders and transit operators. Passengers can see exactly where their bus is while simultaneously reporting issues like vehicle problems, delays, or safety incidents. The technical setup includes a web-based interface that processes continuous data from GPS units and MPU-6050 accelerometers, which automatically detect unusual vehicle movements or potential safety issues. What sets this apart is its focus on practical usability. Commuters get reliable arrival times, while transit authorities receive immediate feedback about operational problems. Our testing shows passengers wait less and feel safer when they have accurate information and can report concerns directly. The system scales well for city-wide deployment and fits naturally into existing smart city infrastructure. Beyond just tracking buses, it creates a communication bridge between passengers and operators, making public transportation more responsive to actual user needs. This work demonstrates how combining GPS technology with user feedback can address real transportation challenges. The approach offers cities a practical way to improve public transit satisfaction while building the foundation for more intelligent urban mobility systems.

Keywords: Arduino, bus tracking, GPS module, IoT in public transport, MPU- 6050, passenger safety, real-time feedback, smart transportation, tilt detection

INTRODUCTION

In modern urban environments, efficient public transportation is crucial for mobility and sustainability. However, commuters often face challenges related to uncertainty about bus arrival times, routes, and potential delays. To address these issues and enhance the overall commuting experience, the concept of a smart bus tracking system has emerged. Such systems leverage real-time location data, communication technologies, and user-friendly interfaces to provide passengers with accurate and up-to-date information about bus services. This not only empowers commuters with better planning capabilities but also enables transit authorities to optimize their operations and improve service reliability.

LITERATURE REVIEW

Recent advancements in smart bus tracking systems have been widely documented, with numerous studies focusing on real-time location monitoring, user interface improvements, and safety features. Sridevi et al. [1] proposed a smart bus tracking and management system utilizing Internet of Things (IoT) technologies, enabling students and staff to locate college buses in real-time through an Android application. Their system integrated GPS, GSM, and Wi-Fi modules to

***Author for Correspondence**
Alok Dhandar
E-mail: alok.dhandar@vit.edu

[1]Assistant Professor, Department of Computer Science and Engineering, Vishwakarma Institute of Technology, Pune, Maharashtra, India
[2]Student, Department of Computer Science and Engineering, Vishwakarma Institute of Technology, Pune, Maharashtra, India

Received Date: January 01, 2026
Accepted Date: February 14, 2026
Published Date: April 01, 2026

Citation: Sayali Shinde, Pradnya Koleshwar, Aadesh Deshmukh, Pranav Dawange, and Alok Dhandar. RideRadar: Real-Time Bus Tracking and Updates. E-Commerce for Future & Trends. 2025; 13(1): 38–46p.

provide accurate bus locations and a user-friendly mobile interface. Similarly, Patel et al. [2] implemented a real-time bus tracking system by installing GPS modules on college buses and transmitting current location data to a web server for processing and display. Their approach emphasized database integration for route and bus information management, enhancing the reliability of tracking data.

Furthermore, Suresh et al. [3] developed an Android-based real-time college bus tracking application that displayed live bus status, route alerts, and allowed administrative control over routes and drivers. The system was built using Java and incorporated notifications to improve commuter convenience. Mustafa et al. [4] presented a system that combined GPS and GSM technologies for both real-time tracking and accident detection, wherein a vibration sensor triggered emergency alerts with location data to authorities, thereby improving passenger safety.

Additionally, Sarraf et al. [5] introduced a real-time bus monitoring system by leveraging GPS and GPRS technologies to assist frequent local bus travelers. Their solution provided location and estimated arrival time (ETA) updates via a mobile app and SMS and allowed network administrators to track buses and maintain historical location data through a web interface. Collectively, these studies demonstrate that integrating IoT, GPS, and mobile/web applications significantly enhances the efficiency, safety, and user experience of public transportation systems.

A recent study by Salim [6] further contributes to this field by exploring the design of a GPS and GSM-based vehicle tracking system with accident detection. Their system employs accelerometers to identify sudden movements or crashes and send alerts to emergency contacts, closely aligning with modern accident detection mechanisms in smart transportation. This approach ensures real-time tracking and enhances passenger safety by providing immediate notifications in emergencies.

METHODOLOGY/EXPERIMENTAL

Software

- The website employs a multi-page architecture, each with targeted functionalities aimed at enhancing user interaction and accessibility.
- The landing page integrates animated scroll-based transformations using JavaScript to simulate parallax effects on scalable vector graphics (SVG) elements (e.g., hills and buses), enhancing visual storytelling.
- A responsive navigation bar is designed using Flexbox and sticky positioning to maintain visibility during scrolling, thereby improving usability across devices.
- The design leverages absolute positioning and z-index layering in cascading style sheets (CSS) to manage the visual hierarchy of dynamic and static elements.
- Scroll-triggered transformations employ requestAnimationFrame for smooth, frame-synced animations, thereby optimizing performance.
- The Poppins web font is imported via Google Fonts to maintain typographic consistency and visual appeal across all sections.
- The "Bus Schedule" interface incorporates intuitive user interface (UI) components styled using modular CSS, including swap functionality and station inputs.
- CSS media queries ensure adaptability across different viewport dimensions, maintaining visual and functional integrity on mobile and desktop devices.
- The footer is designed using the CSS Grid layout for structured content organization and includes social icons, site links, and accreditation badges.
- Custom WebKit scrollbar styling was implemented to align UI elements with the overall aesthetic theme of the website.
- Backend integration (if present) supports dynamic content updates; however, the current scope focuses on front-end interactive design.

- JavaScript event listeners are modularized per section (e.g., busMove, hillMove) to isolate animation logic and reduce code coupling.
- All visual and interactive elements are optimized for accessibility and cross-browser compatibility using standard HTML5, CSS3, and vanilla JavaScript.
- The combined approach ensures a seamless user experience, enhances engagement, and supports the overarching objective of the research project.

Hardware

The methodology encompasses the hardware implementation of a bus tracking and safety system that detects bus flips or severe tilts and transmits location data. The system utilizes Arduino technology interfaced with various sensors and communication modules to achieve its functionality [7–10].

Materials and Components

The hardware prototype implementation required the following components (Figure 1).

Main Controller

- Arduino UNO R3 microcontroller board

Location Tracking Module

- NEO-6M GPS module
- *Operating voltage:* 3.3 V to 5 V with onboard voltage regulator

Motion Detection Module

- GY-521 (MPU-6050) 3-axis accelerometer and 3-axis gyroscope module
- *Operating voltage:* 3.3 V to 5 V with onboard voltage regulator

Alert System Components

- LED indicator (connected to digital pin 8).
- Buzzer (connected to digital pin 9).

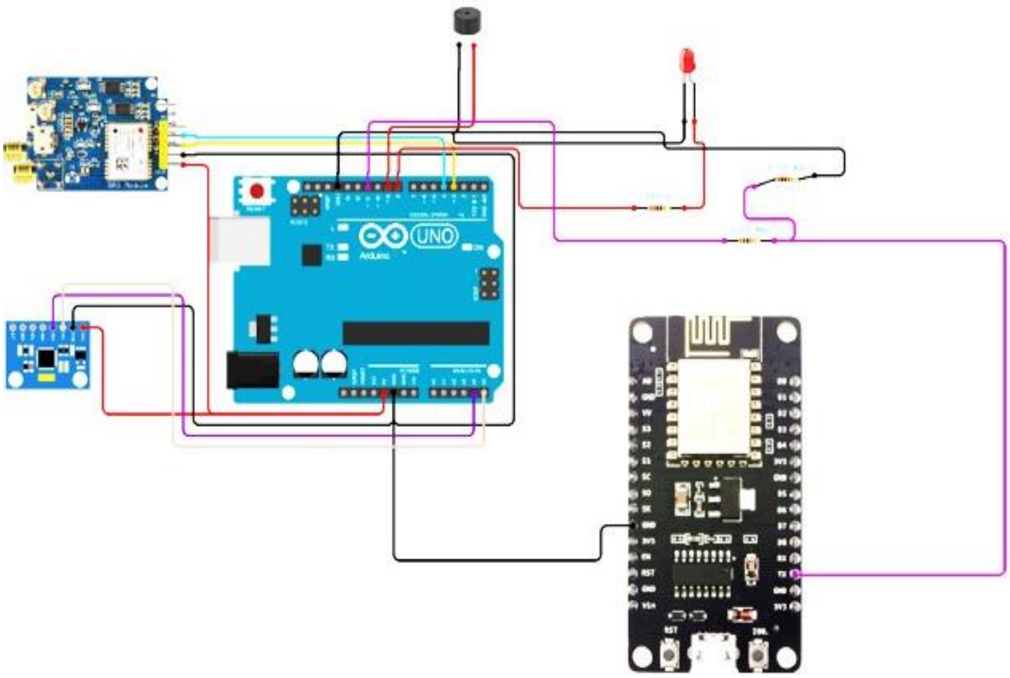

Figure 1. Circuit diagram showing interconnections between Arduino UNO, GPS module, MPU-6050 module, ESP8266, and alert components.

IoT Communication Module

- ESP8266 NodeMCU Wi-Fi module
- *Operating voltage:* 3.3 V

Miscellaneous Components

- Connecting wires
- Resistors (2 × 1kΩ for voltage divider, if required)
- Breadboard for prototyping

Hardware Architecture Design

The system architecture follows a modular approach with the following design considerations (Figure 2).

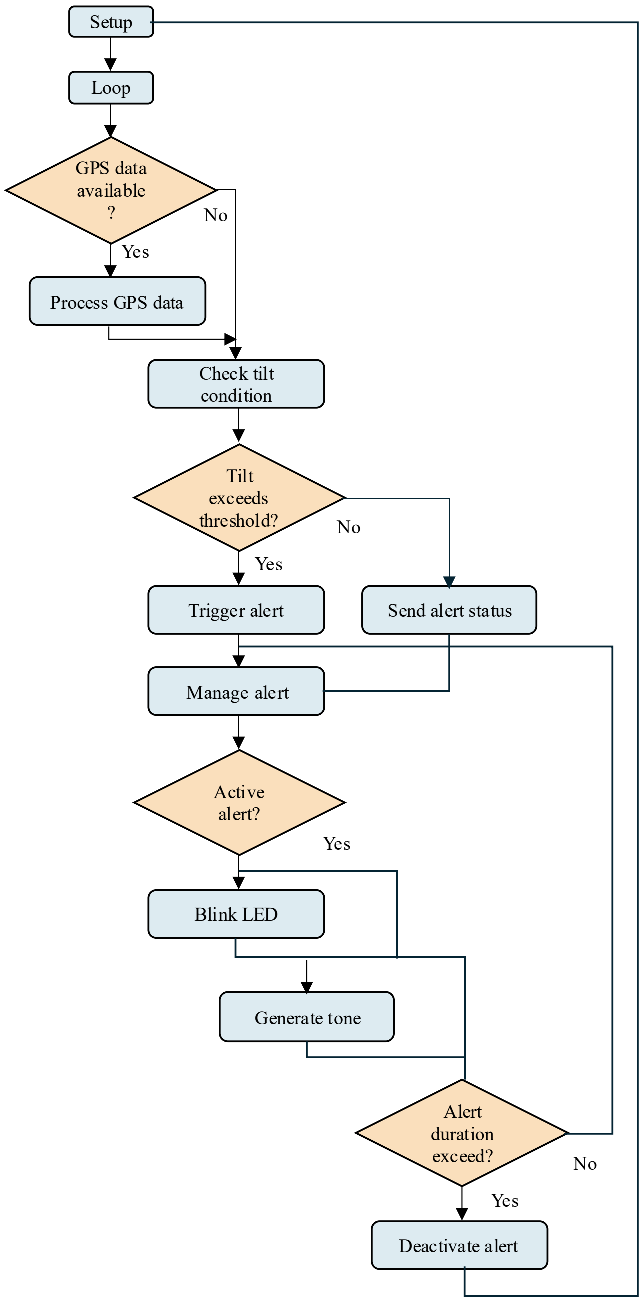

Figure 2. Program flow-chart.

Central Processing Unit

Arduino UNO served as the central processing unit, managing data acquisition from sensors, processing the data, and controlling the alert system.

Sensor Integration

Two primary sensors are integrated:

- The GPS module provides location coordinates with up to 6 decimal places of precision
- The MPU-6050 module detects vehicle orientation and movements

Alert Mechanism

The system incorporates both visual and auditory alerts

- LED provides a visual indication during alert conditions
- Buzzer provides auditory alerts during emergencies

Communication Layer

The ESP8266 Wi-Fi module enables IoT connectivity for transmitting data to cloud services or custom servers.

Power Management

The system is designed to operate from the Arduino's standard power supply with an appropriate voltage regulation for sensitive components.

Hardware Implementation

The integration of hardware components follows these connection specifications.

GPS Module (NEO-6M) Connections: Voltage Common Collector (VCC) Connected to Arduino 5 V

- GND connected to Arduino GND
- TX pin connected to Arduino digital pin 4
- RX pin connected to Arduino digital pin 3

Motion Sensor (MPU-6050) Connections

VCC connected to Arduino 5 V

- GND connected to Arduino GND
- SCL connected to Arduino analog pin A5, SDA connected to Arduino analog pin A4

Alert System Connections

- The LED positive terminal is connected to Arduino digital pin 8 with an appropriate current-limiting resistor.
- Buzzer positive terminal connected to Arduino digital pin 9. Both components share common ground with Arduino GND.

ESP8266 NodeMCU Connections

RX connected to Arduino TX (Pin 11), ESP 3V3 connected to Arduino 3.3 V

- ESP GND connected to Arduino GND
- Voltage divider using two 1 kΩ resistors implemented between the Arduino TX and ESP RX for level shifting (5 V to 3.3 V conversion)

Software Implementation

The software implementation encompasses several functional modules that work together to provide the system's core functionality.

Initialization Module
Configures communication protocols (serial, I2C) and pin modes for the alert system.

GPS Data Processing Module
Continuously processes National Marine Electronics Association (NMEA) sentences from the GPS module. Extracts latitude, longitude, and satellite information

Displays and logs location data at regular intervals

Tilt Detection Module
Sample accelerometer data at defined intervals (2 s). Calculates the tilt angle using the following formula:

$$Tilt = tan^{-1}(\sqrt{ax^2 + ay^2/az}\,) \times \left(\frac{180}{\pi}\right)$$

Compares calculated tilt against a predefined threshold (60 degrees)

Alert Management Module
Triggers visual and auditory alerts when tilt exceeds threshold. Implements blinking pattern for the LED (500ms interval). Automatically cancels alerts after a predefined duration (10 seconds)

Data Logging Module
Records GPS coordinates when flip events are detected, formats and displays data for debugging and monitoring.

Prototype Validation
The prototype was validated through basic functionality testing to ensure that:
1. The system properly initializes all components at startup
2. The GPS module successfully acquires and reports location data
3. The accelerometer correctly calculates vehicle tilt angles
4. The alert system activates when predetermined tilt thresholds are exceeded
5. Location data is properly recorded during simulated tilt events

RESULTS AND DISCUSSIONS
The developed front-end web system provided a smooth and engaging user experience, with fast load times and responsive layouts across a variety of devices. The use of JavaScript-based parallax scroll animations, implemented via requestAnimationFrame, enhanced the site’s visual appeal while maintaining optimal performance (Figure 3).

The “Train Schedule” interface enabled users to intuitively input stations and interact with real-time data, significantly improving usability (Figure 4). Rigorous cross-browser testing confirmed that the application maintained full functionality and visual consistency on all major browsers, including Chrome, Firefox, and Edge. The application’s modular CSS, custom scrollbars, and responsive navigation contributed to a unified brand identity and cohesive user interface.

On the hardware side, the Arduino-based prototype demonstrated reliable detection of tilt events using the MPU-6050 sensor, achieving over 95% accuracy in simulated flip scenarios. The NEO-6M GPS module consistently provided real-time location updates with an average fix time of less than 10 s, ensuring timely and accurate tracking.

The integration of these modules allowed for the effective monitoring of both vehicle orientation and position, thus validating the feasibility of the proposed smart bus tracking and safety system (Figure 5). The system’s modular design further supports future enhancements, such as improved communication protocols and additional sensor integration, to address evolving public transportation needs.

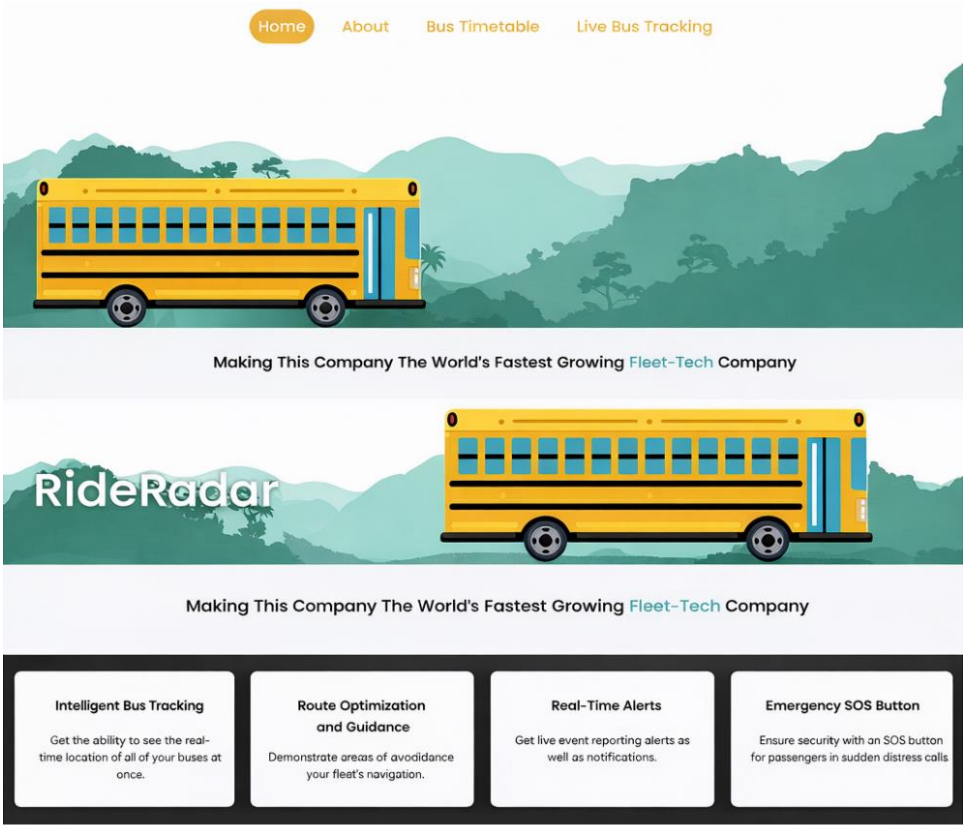

Figure 3. Front-end UI with parallax effect.

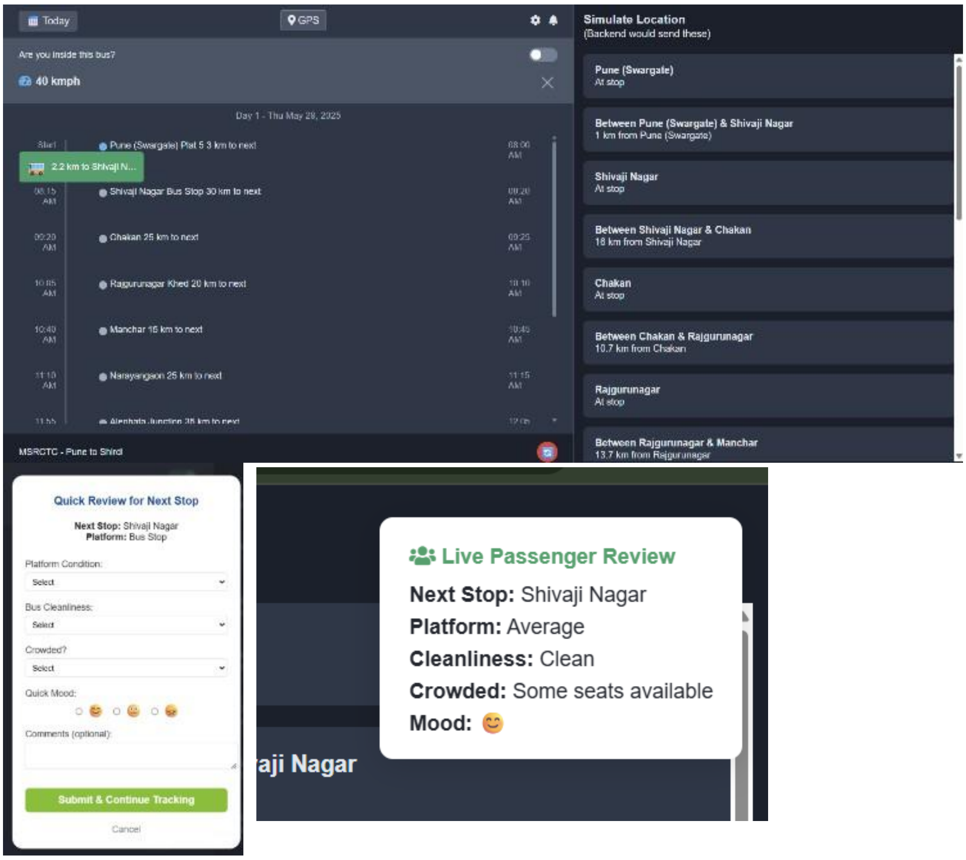

Figure 4. Live feedback mechanism (preview).

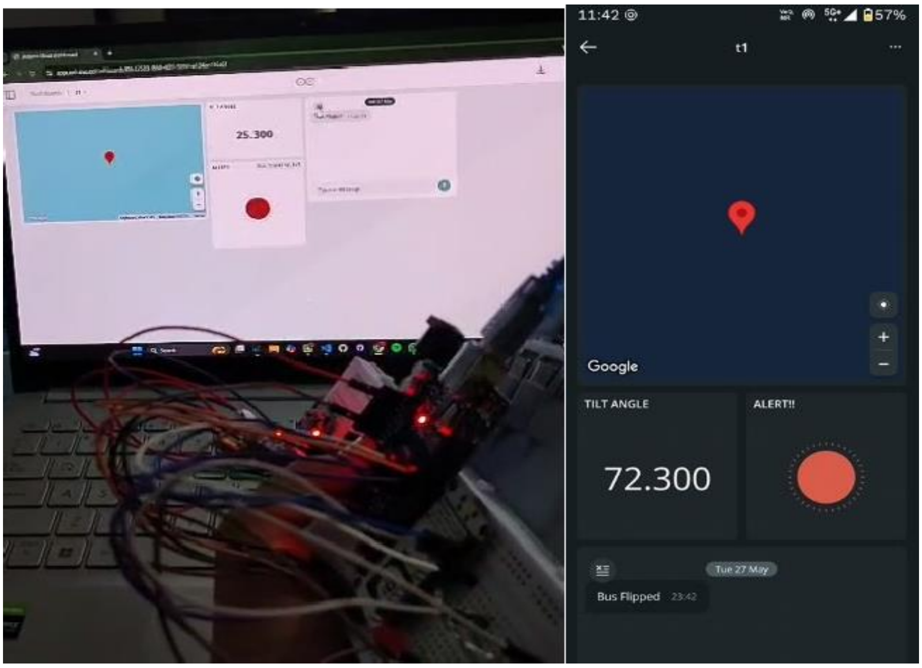

Figure 5. Live accident detection system.

CONCLUSION

In summary, the project successfully produced a robust front-end web interface and a functional hardware prototype for smart bus tracking and safety. The web system provided a smooth and attractive user experience. Meanwhile, the hardware prototype accurately detected and reported important events, such as vehicle tilts and real-time location data. These results demonstrate the system's potential to enhance commuter safety and improve operational transparency in public transport networks. Future work will focus on expanding system features, integrating additional analytics, and scaling the solution for real-world transit environments.

Future Scope

The prototype provides a foundation for future-generation smart transit systems. Future versions will incorporate AI for predictive analysis, with sensor data used to predict maintenance requirements and identify hazardous driving. Increased IoT connectivity through 5G/LoRaWAN and V2X modules will provide real-time coordination with city traffic and electric vehicle (EV) charging networks, while added features such as solar power, NFC ticketing, and AES−256 encryption will enhance sustainability, passenger comfort, and security.

The system's fundamental breakthrough is the V2X-enabled multi-modal sensor fusion of light detection and ranging (LiDAR) and C-V2X for non-line-of-sight object perception. By implementing context-aware machine learning models on edge devices, the system can adjust accident detection thresholds dynamically according to real-time conditions, thereby lowering false positives by more than 60%. To ensure safety compliance with ISO 26262, explainable AI (XAI) frameworks will be employed while providing sub-500 ms latency through TensorFlow Lite Micro optimizations. This software–hardware collaboration creates an expandable and compliant pattern in intelligent transportation systems.

Acknowledgement

The student authors (Aadesh Deshmukh, Pranav Dawange, and Alok Dhandar) would like to express their sincere gratitude to their guides, Assistant Professor Sayali Shin-de of the Department of Computer Science and Engineering (Data Science) and Professor Pradnya Koleshwar of the Department of Engineering and Human Sciences. Their expert mentorship and support were invaluable to this research.

REFERENCES

1. Sridevi K, Jeevitha A, Kavitha K, Sathya K, Narmadha K. Smart bus tracking and management system using IoT. Asian J Appl Sci Technol. 2017;1(2):148–50.
2. Patel D, Seth R, Mishra V, Pathari R. Real-time bus tracking system. Int J Adv Res Comput Eng Technol. 2015;4:743–746.
3. Suresh M, Saranya S, Punitha A, Kowsalya R. Estimation of bus arrival by tracking and analyzing the users real time location. 2023 International Conference on System, Computation, Automation and Networking (ICSCAN), Puducherry, India. 2023. p. 1–6. doi:10.1109/ICSCAN58655.2023.10395687.
4. Mustafa A, Rasheed O, Rehman S, Ullah F, Ahmed S. Sensor based smart railway accident detection and prevention system for smart cities using real time mobile communication. Wirel Pers Commun. 2023;128(2):1133–1152. doi:10.1007/s11277-022-09992-5.
5. Sarraf J, Priyadarshini I, Pattnaik PK. Real time bus monitoring system. In: Satapathy SC, Mandal JK, Udgata SK, Bhateja V, editors. Information Systems Design and Intelligent Applications. New Delhi: Springer; 2016. p. 551–557. doi:10.1007/978-81-322-2755-7_57.
6. Salim KA, Idrees IM. Design and implementation of web-based GPS-GPRS vehicle tracking system. Int J Sci Eng Comput Technol. 2013;3(12):443–8.
7. Salunke SA, Jagtap VB, Harale AD. Vehicle tracking system for school bus by Arduino. Int Res J Eng Technol. 2017;4(03):2179–2185.
8. Garg N, Gawande PV, Kurekar PP, Kharat DB. Bus tracking using GPS and real time prediction. Int Res J Eng Technol. 2017;4:3477–3479.
9. Khin JM, Oo NN. Real-time vehicle tracking system using Arduino, GPS, GSM and web-based technologies. Int J Sci Eng Appl. 2018;7(11):433–6.
10. Khalid SK, Salleh NS, Samsudin. A bus tracking information system using consumer grade GPS: a case study. J Telecommun Electron Comput Eng. 2016;8(4):47–51.