International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 09 Issue: 07 | July 2022 www.irjet.net p-ISSN: 2395-0072

A novel method for Smart school bus tracking system using Machine learning and IoT

Aiswarya Kannan[1], Luckneshwaran.E [2], Ranjith.P[3], VijayaKumar.S[4]

[1]Assistant Professor, Department of ECE, SRM TRP Engineering College, Tiruchirappalli, India
[2,3,4] Third year SRM TRP Engineering College, Department of ECE, SRM TRP Engineering College, Tiruchirappalli, India

---***---

Abstract

Thousands of students use school transportation around the world. We require a transportation system that is effective, reliable, safe, and smart. The suggested system depicts a transportation concept that provides real-time tracking, calculates optimal routes to destinations, detects intrusion, and assists in the maintenance of transportation system statistical data. IR sensors and RFID tags are used to create an IoT network. The detection of intrusion is done via facial recognition. Google Maps, GPS, and accelerometer data are used to detect live location. A bus-mounted Raspberry Pi microcontroller interfaces with a centralised Firebase cloud platform. Admins and parents can access the mapped data via a mobile application. The system saves important data such as driving abilities, attendance analysis, and the optimal routes. In the ML optimizer, the data is effectively used.

Key Words: IoT, IR sensor, RFID, Raspberry Pi, GPS, Cloud, Google Maps, ML

1. INTRODUCTION

For parents, getting their children to and from school is a major problem. Several concerns arise, including extended waits for delayed buses, kidnapping, and kids deboarding at incorrect stations. Other concerns include a lack of knowledge regarding the routes used by the school bus to reach its destinations, as well as stopping unauthorised individuals from accessing the bus. Student safety has long been a top priority, especially for working parents, whose solutions must be continually enhanced [1]. Drivers may not be able to identify all students on time if one goes missing. Children are frequently prohibited from using cell phones on school grounds. It is difficult for school officials to contact each parent about their child's safety. [2]. If the motorist is involved in rash driving, the driver should be assisted and the caretakers should be informed [3]. By tracking and monitoring students, the Internet of Things provides a much better approach for ensuring their safety.

IoT is implanted as the primary concept to solve and overcome these challenges. Web-based bus tracking is provided by the proposed solution. Students entering the bus are detected by an IR sensor at the entry, which is verified by an RFID reader. The RFID device is scanned for identification and information retrieval. When students board the bus, they must wear their unique tag to be scanned by the RFID scanner. Parents and administrators are notified when their child boards the bus using confirmed scanning. The Raspberry Pi microcontroller is used to update attendance over Firebase. If a scan mismatch occurs, a Firebase notice is issued to detect an intruder and capture the intruder's photo. When boarding or disembarking from the bus, each child is identified.

The accelerometer keeps track of the bus's condition and analyses how it drives. In real-time, the technology monitors the child's location and bus route. The car is followed by the GPS module.The GPS position, driver behaviour, and emergency conditions are used to monitor the overall system. Caretakers would be contacted right away if an accident or emergency occurred. All preceding sections would be saved in Firebase for future use.

1.1 Literature Survey

F Judy et al. [4] proposed a school bus monitoring system that uses RFID and GPS to communicate to a remote server using Wi-Fi and an ESP8266 microcontroller. Caretakers could access information utilising the cloud-based Firebase messaging service. Jisha et al. [5] developed a car monitoring system for schoolchildren that employed GPS and GPRS/GSM technology to ensure the safety of the students. The system comprised of an Internet-enabled android application that communicated with a server.

W. Pattanusorn et al. [6] devised a system that automatically registered children's information at the entry-level when they passed through the scanner. SMS alerted parents of their children's school bus arrival and departure times. H Eren et al. [7] proposed a sensory-based driving behaviour analysis technique. The tracking was done using a smartphone to simulate a car-independent system without the need for vehicle-mounted sensors. Muneer et al. proposed an android-based framework that combined GPS data with Google maps data to accurately locate the misplaced mobile phone. [8]

The need for the development of a real-time transportation-based information system for users, which might aid in improved trip planning and reduce bus waiting time, sparked interest. Real-time data processing can help commuters get at their destination faster by reducing waiting time.

The proposed work uses IoT with RFID, Raspberry Pi, IR sensor, GPS, Firebase, and Google Maps to achieve features such as attendance analysis, notifying end-users with alighting and boarding alerts, location tracking through GPS by integrating Google Maps, intrusion and accident detection, and cloud storage to expand the security framework and traceability aspects of the child. The mobile application would make this information available to the administrator, parents, and driver.

2. COMPONENTS

The proposed system utilizes the following hardware and software components:

2.1 Hardware Specifications

The bus's hardware is the most crucial component. The hardware utilised for prototype development is described in this section, which comprises the following:

A. IR Sensor

The primary function of an IR Sensor is to measure and detect infrared radiation in the surrounding environment. The proposed system uses an IR sensor as the first step in verifying the student's entry into the bus. With a pair of infrared transmitters and a receiver tube, an IR Sensor module has an adaptable potentiality of atmospheric light. Infrared technology is used in a variety of wireless applications.

B. RFID Reader RC522

The RFID RC522 Card Reader Module, based on the MFRC522 controller, is a low-cost 13.56 MHz RFID reader module. The module necessitates a 3.3V power supply. It can communicate with any CPU board directly using the SPI protocol, and it also supports I2C and UART. It is utilised for attendance analysis and person identification in the proposed system.

C. Raspberry Pi 3 B+ Micro-controller

The Raspberry Pi foundation microcomputer that was created to promote programming and computing principles. It has a 64-bit quad-core processor with a clock speed of 1.4GHz and dual-band 2.4GHz. It has 5GHz wireless LAN and Bluetooth connectivity, making it an ideal alternative for highly networked designs. Its high processing power and on-board connectivity make it ideal for IoT applications.

D. Camera Module

The 5-megapixel Camera Module Rev 1.3 is a specially developed Raspberry Pi add-on. A unique CSI interface is utilised for camera interaction. The CSI bus provides extremely high data speeds and consistently transports pixel data. It is used to capture a snapshot of the intruder in the proposed system.

E. GPS Module –Neo 6M

The Neo-6M GPS module is a reliable GPS receiver with a 25 x 25 x 4mm ceramic antenna built in. It has a good satellite search capacity. The power and signal indicators can be used to check on the module's status. It is used to gather information about geographical parameters.

F. MPU6050 Accelerometer

The MPU6050 is a single-chip 3-axis accelerometer and gyroscope. It is also known as a six-axis motion tracking or six Degrees of Freedom (DoF) device because of the three accelerometer and three gyroscope outputs. Hardware components used in the prototype development are shown in Figure 1.

Fig -1: Hardware Components

2.2 Specifications for Software

HTML, JS, CSS, jQuery, and Bootstrap were used to create the android application. In order to view the current location, the Google Maps API has been integrated into the programme. Using Firebase fire-store, we were able to get real-time changes. Apache Cordova was used to encapsulate this view in an app.

A. Firebase

The Firebase database is a cloud-based database that allows for real-time data syncing and storage across users around the world. It promotes user participation and serverless application development. JSON is the format in which the NoSQL database is stored. When developing cross-platform programmes, a Realtime Database instance can be shared among all clients. Even if the app goes down, the data is still accessible.

B. API for Google Maps

Google has created application programming interfaces (APIs) that allow users to communicate with Google Services and integrate them with other services. Analytics, machine learning as a service, and user data access are among the features. Google Maps can also be integrated into a website or application. It is possible to provide users with relevant content and to customise their map view according to the site. It is used in the suggested model to add a map to the Android app to detect bus routes and deliver the best way to destination with real-time traffic updates.

C. Node-RED

Hardware components, APIs, and other web services are used to collaborate. Node-RED is a popular visual programming tool that comes with a web-based editor. JSON can be used to save the flows and share them with others. They can be used at any moment with a simple click. It is an event-driven and non-blocking approach that is based on Node.js. The Raspberry Pi is being used in the current system.

D. jQuery and Bootstrap

Bootstrap is a web application framework. Consistency is maintained across browsers and device screen sizes thanks to HTML and CSS. It has a large number of plugins and themes. Because jQuery is an open-source tool, it is used to create client-side scripting on HTML, which makes websites more responsive.

E. Apache Cordova

Apache Cordova is a mobile development framework that is free and open-source. By combining HTML5, CSS3, and JavaScript, we can create hybrid web applications. It's in charge of bridging the gap between web-based apps and native mobile capabilities. It serves as a link between web apps and mobile devices.

3. PROPOSED SYSTEM ARCHITECTURE

The suggested system seeks to provide effective services through the integration of various technologies and the Internet of Things. The suggested system's diagrammatic representation is shown in Figure 2.

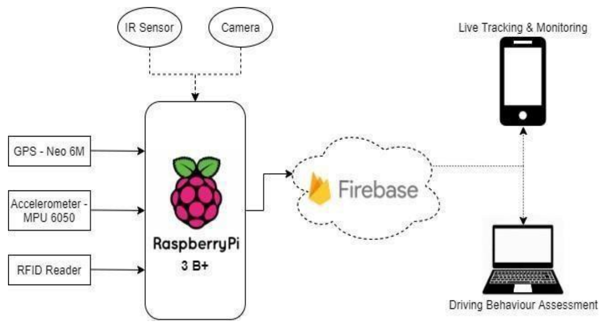


Fig -2: Representation of the System Proposed

In Figure 3, you can see a hardware prototype of the system. It emphasises the various components that were used to construct the prototype. The Raspberry Pi serves as the system's brain, connecting to several sensors like as GPS, Accelerometer, RFID, IR, and Camera. For database management, it's also linked to Google Firebase. The Raspberry Pi and the Firebase have most of the data in sync.

The administrative, driver, and parent mobile apps are designed to keep everyone up to date on the latest information, such as routes, live monitoring, attendance, and notifications.

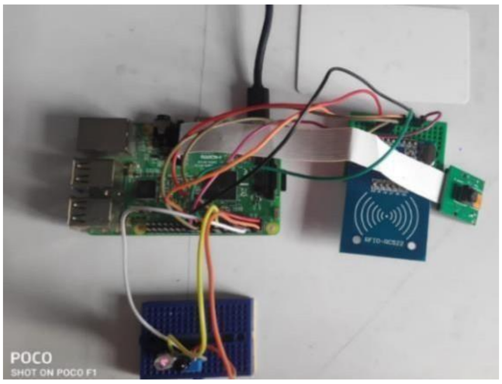
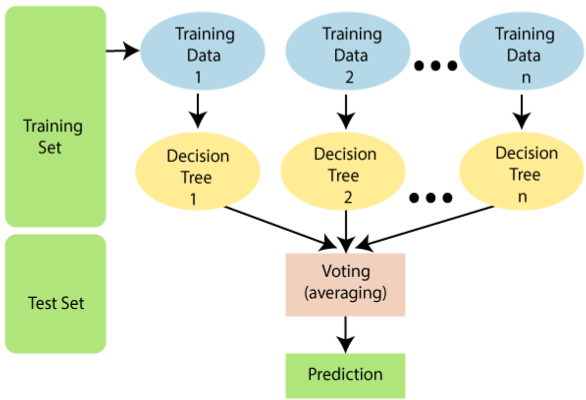


Fig -3: Prototype Hardware

The accident detection method is based on the Random Forest Machine Learning Classifier (Fig. 4). Random Forest is a popular machine learning algorithm that belongs to the supervised learning technique. This algorithm solves classification and regression problems in machine learning. It is based on the notion of ensemble learning, which entails merging numerous classifiers to solve a complex problem and improve the model's performance.



Random Forest Algorithm in Action (Fig. 4)

"Random Forest is a classifier that contains numerous decision trees on various subsets of a given data set and takes the average to increase the predicted accuracy of that data set," as the name suggests. Rather than relying on a single decision tree with a majority of votes, the random forest classifier collects predictions from all trees to forecast the final result. The greater the number of trees in the forest, the higher the accuracy and the less likely it is to overfit.

The Random Forest algorithm, as shown in Figure 4, works in two phases: first, it creates a random forest by mixing N decision trees, and then it makes predictions for each tree formed in the first phase.

The steps and diagram below show how the working process works:

Step 1: Picking K data points at random from the training set.

Step 2: Create decision trees for the data points you've chosen (Subsets).

Step 3: Determine the number of decision trees to be constructed (N).

Steps 4: steps 1 & 2 are repeated.

Step 5: Find the forecasts of each decision tree for new data points, and assign the new data points to the category with the most votes.

In Section 4, Fig-8 depicts the implemented architecture algorithm that outlines the entire flow. It emphasizes how the various elements (hardware and software) interact with one another. Students enter the bus through an IR sensor, which is verified by an RFID reader.

Following successful verification, the tag data is transferred to the cloud via a Raspberry Pi microcontroller, where it can be accessed by admins, parents, and drivers via a mobile application. This aids parents and school officials in collecting attendance reports. If an RFID reader is not discovered, the camera is used to identify a suspected breach. At that point, a photograph of the person is taken.

Throughout the journey, data from the accelerometer and GPS module is collected and sent to the cloud.

The physical coordinates of the vehicle's location as well as the speed can be tracked using GPS-based information combined with Google maps data. Machine Learning information from the driving behaviour analysis is used to detect accidents, assuring student safety. In the event of an emergency, the administration can contact the driver.

4. RESULTS

This phase of the pilot study develops the findings and conclusions. The bus route has three stops in the design execution. The data was retrieved for two months (February and March 2020). The traffic pattern, driving behaviour, and identification of possible bottlenecks in that route are among the conclusions drawn from the study and testing. Data for the proposed system was gathered using GPS units installed in school buses.

The 3.6-kilometer experimental route was chosen from Vidyalankar Institute of Technology in Mumbai to Matunga Railway Station in Mumbai. On this route, there are nine bus stops. GPS data from the devices installed in the buses was relayed to a server every 60 seconds. Latitude, longitude, speed, and time stamps were all included in the GPS data.

The data received in the server was immediately processed in order to make a real-time prediction of the bus's arrival time. The time of arrival of the next bus stop was predicted using data from previous days and real-time data from a GPS receiver at the current bus stop. The mobile app enabled for attendance tracking as well as real bus location tracking, as well as warnings (for intrusion detection and accident detection). The outputs for RFID scanning, student detection, and intrusion detection are depicted in Figures -5, 6 and 7.

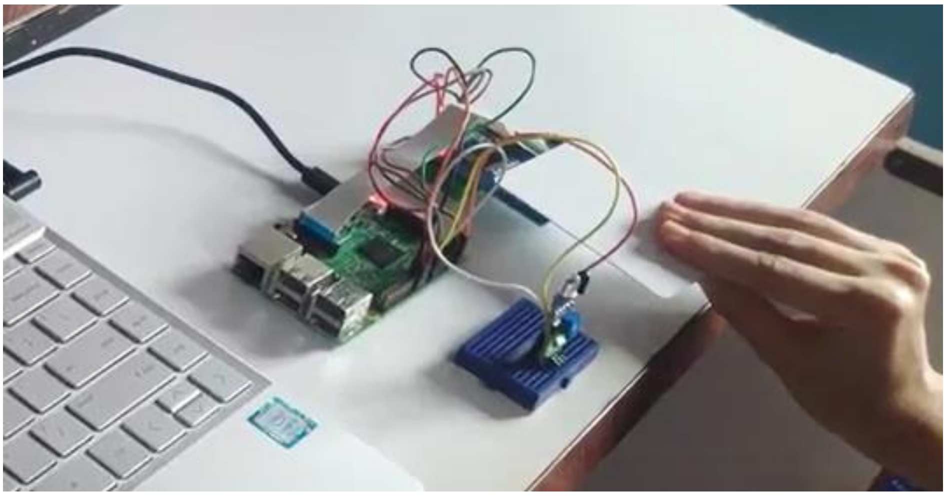


Fig -5: RFID Scanning process setup

Fig -6: Student detected on successful scan

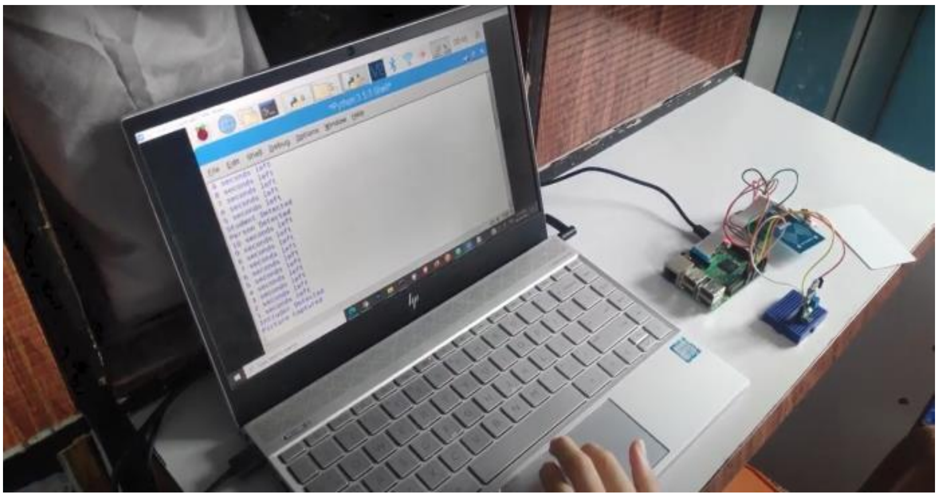


Fig -7: Intruder detected on unsuccessful scan

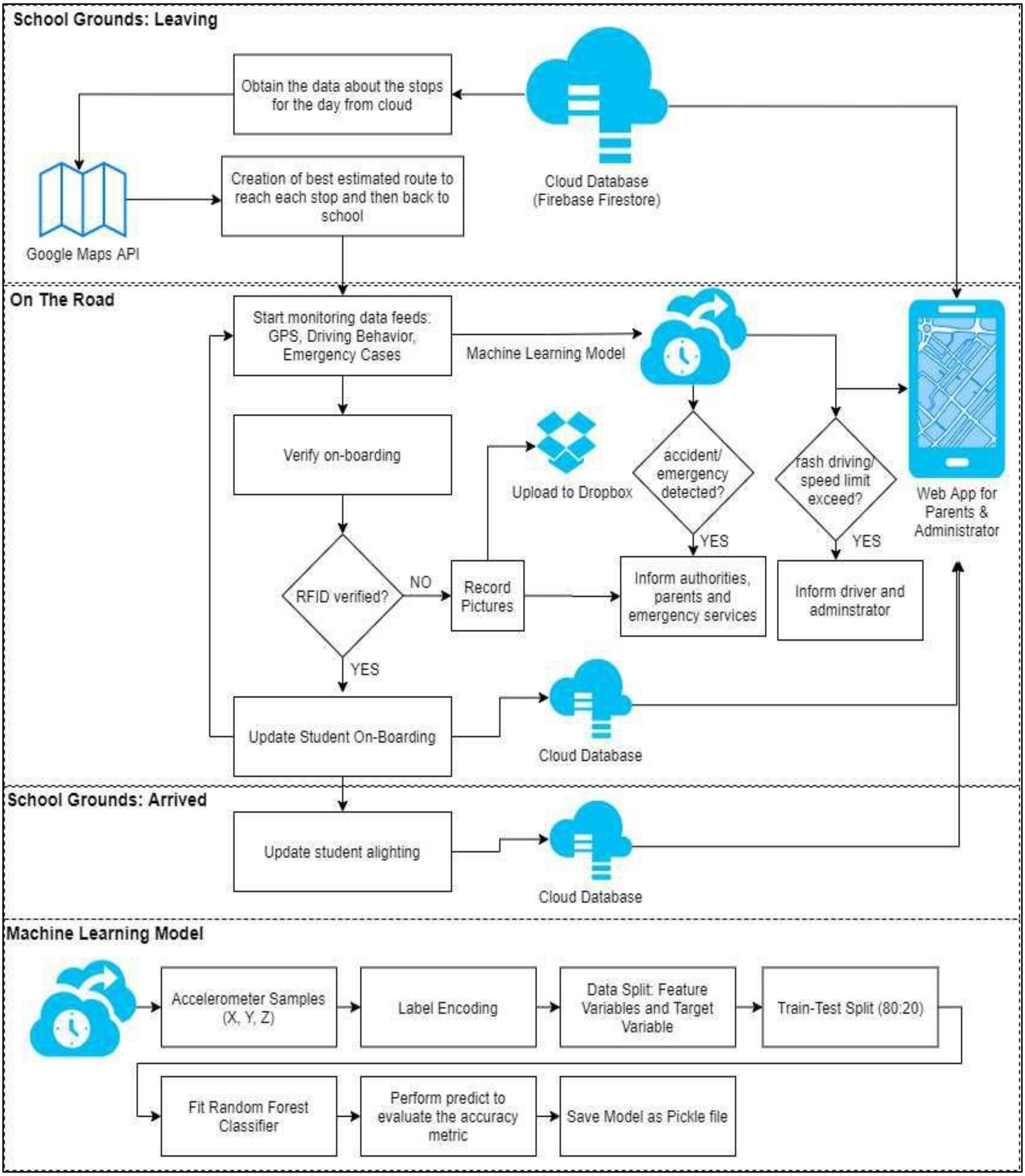


Fig-8: Algorithm model

The implementation algorithm has been broken into three components, as shown in Fig-8, to make it easier to understand. The data about the stops and best-estimated routes is examined and stored on the cloud database starting at the top. RFID is set up to detect students and intruders. The image of the invader is captured, saved, and sent to the cloud. Using a Machine Learning model, an accelerometer is utilised to calculate and evaluate driving behaviour for accident detection.

Figure 9 depicts various views of the proposed implementation of the mobile application. The graphs below exhibit accelerometer data for accident detection for several samples (X-Axis – Timeline & Y-Axis – G Values). The samples were gathered in order to better understand how people behave in various situations. The data was then entered into the ML model to make predictions after it had been wrangled. A 91.8 percent accuracy rate was obtained.

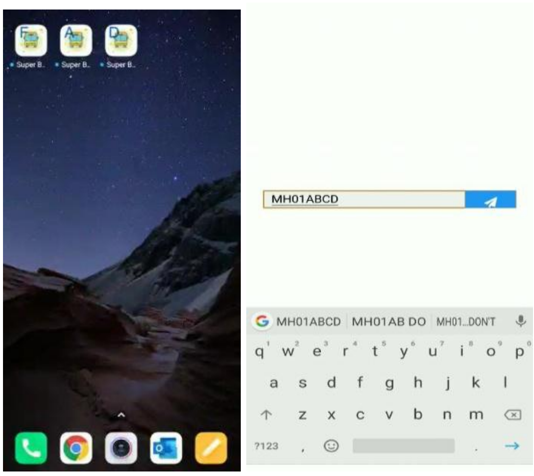


Fig-9: 3 Apps and a snap entering bus details Tabs

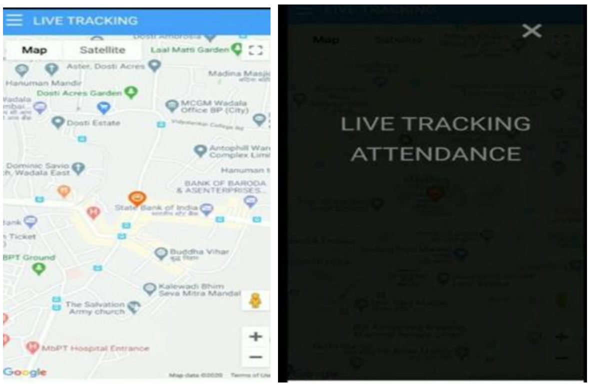


Fig-10: Live Tracking Tabs

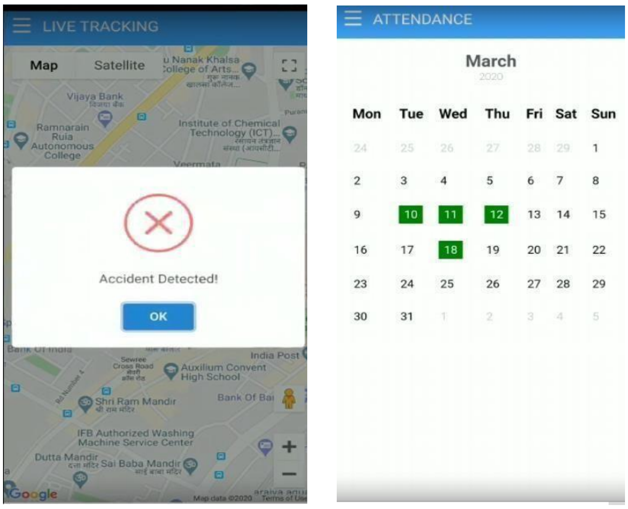


Fig -11: Accident detection and Attendance view

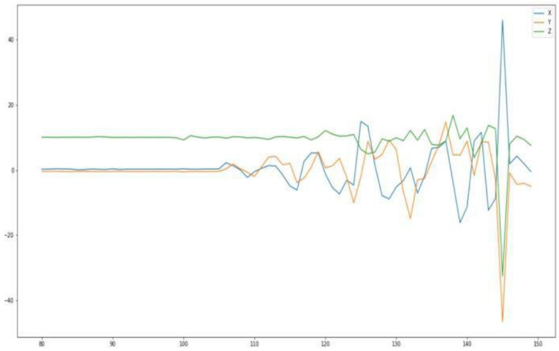


Fig-12a: Sample Data 1

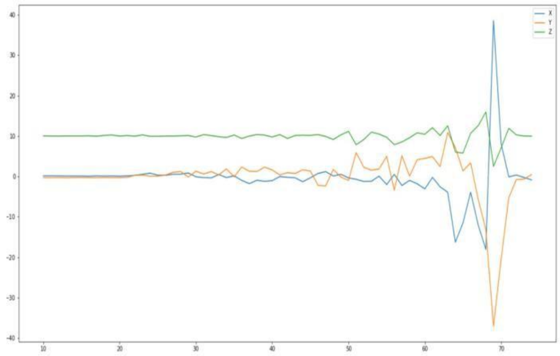


Fig-12b: Sample Data 2

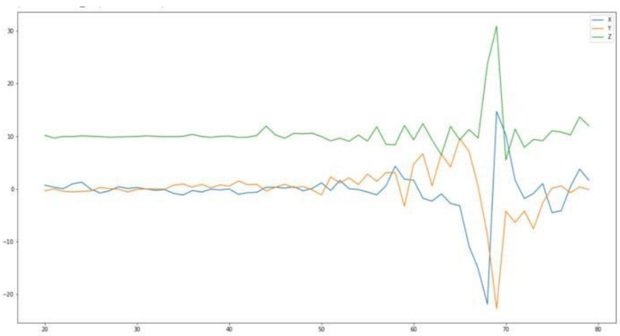


Fig-12c: Sample Data 3

CONCLUSIONS

The prototype was successfully implemented, and the key functionalities of student detection via RFID, intrusion detection, location tracking, and accident detection were all confirmed to be correct. Secondary capabilities such as attendance analysis, route estimates, cloud-storage of essential data, and custom alerts were also tested and found to work as expected. The prototype outperforms GSM and Arduino-based systems by combining GPS and Google Maps APIs to give live location and real-time tracking of children. Overall, this was a positive step toward a brighter future for Smart School buses.

REFERENCES

[1] R. Bandal and A. Oak, "Managing Location Identification and Chain SMS for Smart School Transport System using IoT," IEEE 2019 International Conference on Computer Communication and Informatics (ICCCI - 2019), Jan. 23-25 2019, Coimbatore, INDIA.

[2] B. Pavithra, S. Suchitra, P. Subbalakshmi, J. Mercy Faustina, "RFID based Smart Automatic Vehicle Management System for Healthcare Applications," Third International Conference on Electronics Communication and Aerospace Technology [ICECA 2019].

[3] N. Akhtar, K. Pandey, S. Gupta, "Mobile Application For Safe Driving," 2014 Fourth International Conference on Communication Systems and Network Technologies.

[4] J. T. Raj, J. Sankar, "IoT Based Smart School Bus Monitoring and Notification System," 2017 IEEE Region 10 Humanitarian Technology Conference (R10-HTC) 21-23 Dec 2017, Dhaka, Bangladesh.

[5] Jisha R C et al., "An Android Application for School Bus Tracking and Student Monitoring System," 2018 IEEE International Conference on Computational Intelligence and Computing Research.

[6] W. Pattanusorn and I. Nilkhamhang, "Real-Time Monitoring System for University Buses using Available Wi-Fi Networks and Travel Time Prediction," 2018 15th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology.

[7] H. Eren, S. Makinist, E. Akin, and A. Yilmaz, "2012 Intelligent Vehicles Symposium Alcalá de Henares, Spain, June 3-7, 2012.

[8] M. Ahmad Dar and J. Parvez, "A Live-Tracking Framework for Smartphones," IEEE Sponsored 2nd International Conference on Innovations in Information Embedded and Communication Systems ICIIECS'15.