TRAIN TRACKING AND DETECTION SYSTEM

2023-302

Final Report

B.Sc. (Hons) Degree in Information Technology
Specializing in Data Science

Department of Information Technology

Department of Computer Systems Engineering

Sri Lanka Institute of Information Technology
Sri Lanka

October 2023

TRAIN TRACKING AND DETECTION SYSTEM

2023-302

B.Sc. (Hons) Degree in Information Technology

Specializing in Data Science

Department of Information Technology

Department of Computer Systems Engineering

Sri Lanka Institute of Information Technology

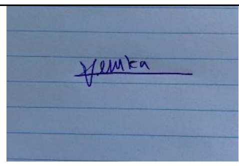
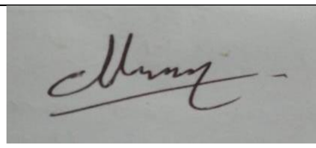
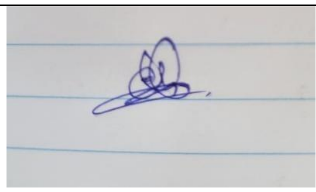
Sri Lanka

October 2023

DECLARATION

We declare that this is our own work, and this proposal does not incorporate without acknowledgement any material previously submitted for a degree or diploma in any other university or Institute of higher learning and to the best of our knowledge and belief it does not contain any material previously published or written by another person except where the acknowledgement is made in the text.

Also, we hereby grant to Sri Lanka Institute of Information Technology, the nonexclusive right to reproduce and distribute my dissertation, in whole or in part in print, electronic or other medium. We retain the right to use this content in whole or part in future works (such as articles or books).

Name	Student ID	Signature
Biyanwila B.D.V.J	IT20212490	
Jayanga B.M.C	IT20188672	
Amarasinghe C.D	IT20187064	
Wijewardene L.L	IT20101824	

The above candidates are carrying out research for the undergraduate Dissertation under my supervision.

Signature of the supervisor: Dr. Shanta Rajapaksha Yapa Date: 12/10/2023

ABSTRACT

This research project focuses on the development and implementation of an IoT-based train detection system aimed at enhancing passenger safety and minimizing accidents. The proposed system addresses the absence of a dedicated train detection system in Sri Lanka, making it an important and timely research topic. A train detection system is an integral component of railway safety, as it allows for real-time monitoring and identification of potential risks along the tracks. However, Sri Lanka currently lacks such a system, leaving passengers vulnerable to safety hazards. Therefore, this research aims to bridge this gap by introducing and implementing an IoT-based train detection system.

By leveraging IoT devices, the proposed system enables continuous monitoring and detection of critical events such as unauthorized access to railway tracks, track obstructions, and abnormal train behaviour. Real-time detection of these risks allows for timely interventions, reducing the occurrence of accidents and ensuring passenger safety. The implementation of this train detection system in Sri Lanka holds significant importance. It serves as a catalyst for research and development in the field of railway safety, offering a novel and innovative solution to address the current gaps. By introducing this system, Sri Lanka can enhance its railway infrastructure, improve passenger safety, and serve as a model for other countries facing similar challenges.

Through this research, valuable insights will be gained regarding the feasibility, effectiveness, and impact of an IoT-based train detection system in the context of Sri Lanka. The findings will contribute to the body of knowledge on railway safety and provide recommendations for the adoption and adaptation of similar systems in other regions.

In conclusion, the implementation of an IoT-based train detection system in Sri Lanka is crucial for ensuring passenger safety and minimizing accidents. This

research presents a significant opportunity to address the existing gap, offering a platform for innovation and enhancing the country's railway infrastructure. The findings and recommendations from this study will contribute to the advancement of railway safety globally.

Keywords — Passenger Safety, IoT-based, Railway Crossing. Machine Learning, Automated, Reduce Accidents

ACKNOWLEDGMENT

The work described in this system was carried out for a B.Sc. research project and is a result of immeasurable guidance, support, and encouragement given by all the mentors of our research team. Furthermore, our sincere appreciation is deserved by all parties who supported us specially our family members, friends, and everyone else who supported us in our efforts to complete this research successfully.

In addition, we would want to convey our heartfelt gratitude to our parents, all the lecturers, and colleagues at the Sri Lanka Institute of Information Technology who have assisted us during this endeavor. It would have been difficult to meet study objectives without their assistance, extensive support, and suggestions.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

Abbreviations	Description
IT	Information Technology
IOT	Internet Of Things
App	Application
GPS	Global Positioning System
SMS	Short Message Service
UI	User Interface
GSM	Global System for Mobile Communications

1. INTRODUCTION

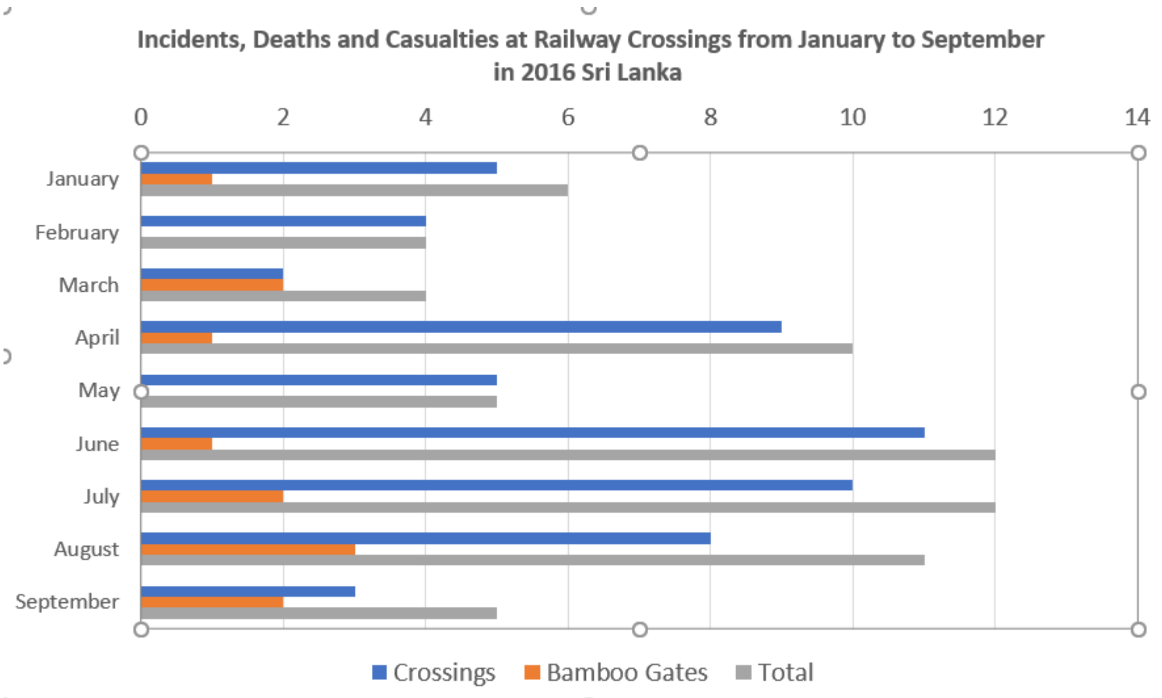
1.1. Overview

Railway Crossing accidents have been a significant safety concern not only worldwide, but also in Sri Lanka, resulting in significant loss of life and property annually. The biggest railway crossing collision that happen in Sri Lanka was on 27th April 2005, near Polgahawela, resulted the death of 41 citizens.

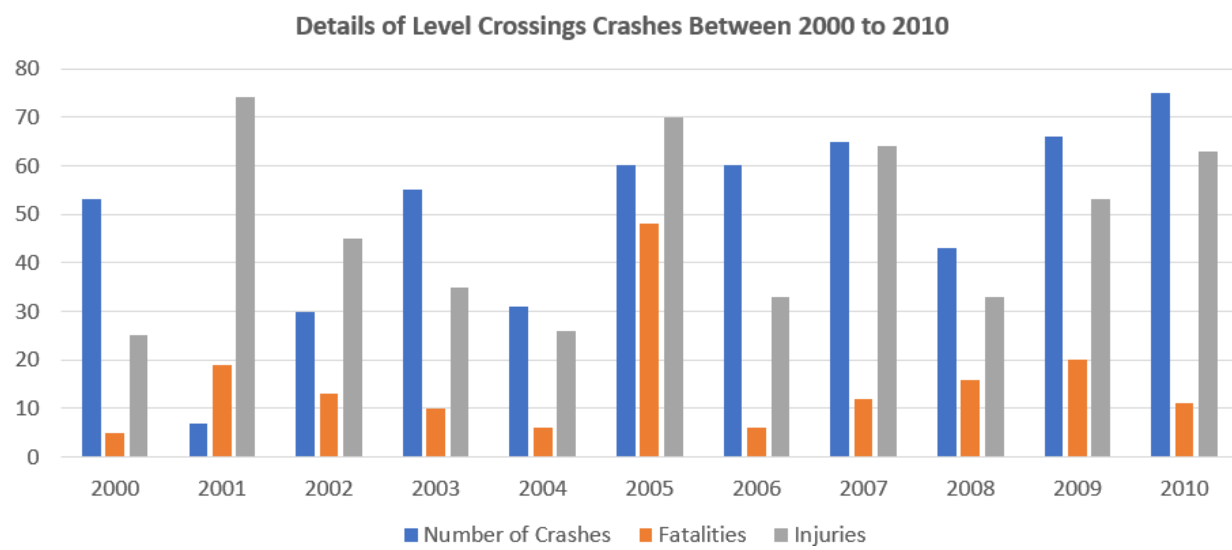
[1]

Polgahawela level crossing accident was a collision between a bus travelling from Galkiriyagama to Colombo and a train at a level crossing in Yangalmodara, near Polgahawela in Kurunegala district on 27 April 2005 at 8.30 local time, which resulted in the death of 41 people.

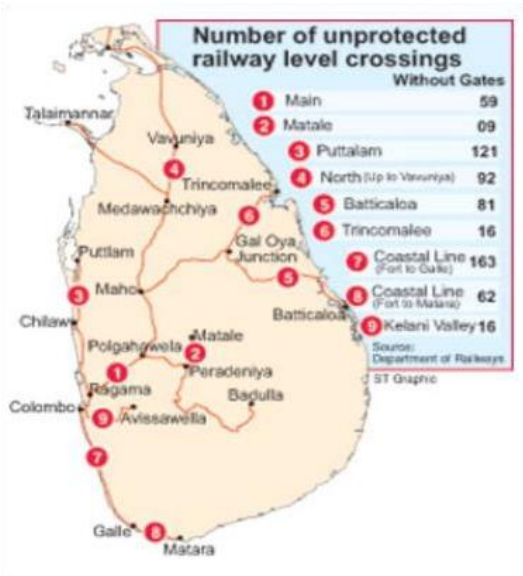
These are some detailed facts that proves how much damage is done annually from railway collision in Sri Lanka.


Incidents, Deaths and Casualties at Railway Crossings from January to September in 2016 Sri Lanka
0
2
4
6
8
10
12
14
January
February
March
April
May
June
July
August
September
Crossings
Bamboo Gates
Total

[2]


Details of Level Crossings Crashes Between 2000 to 2010
80
70
60
50
40
30
20
10
0
2000
2001
2002
2003
2004
2005
2006
2007
2008
2009
2010
Number of Crashes
Fatalities
Injuries

[2]



[3]

And also statistics proves that, 1 person dies due to a road accident in every 3 hours averagely in Sri Lanka. [4] and Every 3 days, a Child dies in Road Accident. [4]

We identified some major reasons for these collisions. Major reason for so much collisions in Sri Lanka is there are so much unsafety unprotected railway crossings all around the Island. Above image [3] shows how many unprotected railway crossings in all around the Island. Due to some heavy rains or fog conditions, people cannot see the upcoming railway crossings when they are driving the vehicle. Human Errors is also another fact for railway collisions in Sri Lanka. That means citizens may misjudge the speed or distance of an approaching train, leading to accidents or near-misses. And also, Citizens may not be aware of the dangers posed by railway crossings or the proper safety procedures to follow when crossing tracks. Therefore, providing solutions to reduce these collisions happening in railway crossings became a major need specially in Sri Lanka because many human and animal lives and properties lost to the country annually. But from the government side, we cannot see they are considering this as a big problem. They didn’t even take necessary steps to reduce these railway crossing collisions like providing gates near railway crossings, signal and alarms likewise. As citizens who feel this as a major problem in Sri Lanka, we need to find some solution to reduce these annual railway crossings collisions since the government is not involving on this matter much.

Currently, there is no proper solution to address the railway crossing collisions in Sri Lanka. So as undergraduate students who are specializing in IT field, we propose a IT-based solution to address these railway crossing collisions in Sri Lanka. Even though there are apps to alert train schedules and train routes, time tables, there is no app to address the safety of citizens who are crossing the railway crossings in Sri Lanka. The "Train Detection & Alert System" for citizens ins Sri Lanka project aims to address the challenges faced by citizens who cross the railway crossings through the use of advanced technology and infrastructure improvements. To address this problem, our research project proposes a system that utilizes various technologies, including IoT devices, GPS tracking, machine learning algorithms, security technologies and real-time databases.

Our Train Detection & Alert System includes multiple components, such as real-time tracking of trains and vehicles near the railway crossing, alert system to vehicles which are moving towards to the railway crossing, alert system to all the users who are near to the railway crossing, sending real-time train location details to relevant IOT devices, security system for the system and predictive maintenance system. These all components work together to detect potential safety hazards and prevent accidents from occurring. The project aims to create a user-friendly interface for the system, making it accessible to a wide range of users and enhancing the overall passenger experience. Additionally, the system will be used by train authorities to monitor the railway crossings collisions, helping to improve the overall efficiency and safety of citizens and taking relevant actions to reduce the collisions happening in railway crossings.

The system will include a GSM tracker system that can be used to locate lost signals. The GSM tracker system will enable the retrieval of lost devices and other valuable assets. The system will also provide real-time updates on the location of these assets, allowing for efficient tracking and management.

In conclusion, the development of an IoT device and GSM tracker system has the potential to significantly improve safety measures at railway crossings and predict the location of lost signals. The implementation of this solution will not only reduce the risk of accidents but also improve the efficiency of asset tracking and management.

1.2. Develop a system that utilizes GSM trackers on trains and IoT devices at railway crossings to predict and alert potential blind spots on the train

The railway system is an integral part of modern transportation infrastructure, facilitating the movement of goods and people across vast distances. However, railway transportation comes with inherent risks, especially when crossing roads or highways. The safety of railway crossings is of utmost importance, as accidents at these crossings can be catastrophic and often result in severe injury or loss of life.

One significant challenge in ensuring the safety of railway crossings is the existence of blind spots on trains. Blind spots are areas of the train that are not visible to the driver or other railway personnel, making it challenging to detect obstacles or hazards. Blind spots can be caused by various factors, including the curvature of the track, the length of the train, or the presence of other obstructions.

To address this issue, this project proposes the development of a system that utilizes GSM trackers on trains and IoT devices at railway crossings to predict and alert potential blind spots on the train. The system will collect data from IoT devices installed at railway crossings, including information on train speed, direction, and length. This data will be analyzed to determine the areas where blind spots may occur, taking into account the curvature of the track and the length of the train.

The system will also incorporate GSM trackers on trains to provide real-time data on the train's position and speed. This information will be used to predict the location of the train at specific intervals along the track, enabling the system to alert railway personnel of potential blind spots.

By utilizing a combination of IoT devices and GSM trackers, this system has the potential to significantly enhance safety measures at railway crossings and reduce the risk of accidents. The system's predictive capabilities will enable preemptive measures to be taken to prevent accidents, while the real-time tracking of trains will enable railway personnel to be alerted of potential blind spots in advance. Overall, the development of this system has the potential to revolutionize railway safety and improve the efficiency of railway transportation.

The proposed system will incorporate advanced technologies such as artificial intelligence, machine learning, and data analytics to provide accurate predictions and alerts for potential blind spots on trains. The IoT devices at railway crossings will collect data continuously and transmit it to a central server for analysis. The data will be processed using machine learning algorithms to identify patterns and trends, which will then be used to predict potential blind spots on the train.

The GSM trackers on trains will transmit real-time data on the train's location and speed to the central server. The system will use this information to determine the train's position and predict potential blind spots along the track. The system will also incorporate alerts and notifications that will be sent to railway personnel in case of a potential hazard.

This system has several advantages over traditional safety measures at railway crossings. Firstly, it provides real-time updates on train movements, enabling railway personnel to respond quickly to any potential hazards. Secondly, it enables preemptive measures to be taken to prevent accidents by predicting potential blind spots in advance. Thirdly, the system will provide accurate data on train movements, enabling railway operators to optimize their operations and improve the efficiency of railway transportation.

In conclusion, the proposed system has the potential to revolutionize railway safety and improve the efficiency of railway transportation. By utilizing a combination of IoT devices and GSM trackers, this system will provide accurate predictions and alerts for potential blind spots on trains, enabling railway personnel to take preemptive measures and prevent accidents. This system is a significant step towards making railway transportation safer and more efficient, and it has the potential to transform the railway industry in the future.

1.3. Sending alerts to the users via the app and predicting the likelihood of crossing the railway-crossing on a given day

This component of the research project aims to predict whether a vehicle is likely to cross a railway crossing on a particular time and particular day when a train is also approaching, and if he/she is likely to cross the railway crossing, send alert warning messages. And then Analyse the past patterns of vehicles and predict if they are likely to cross the railway crossing on a given day.

Since we are developing an app for our users, we ask the location permission from the user when they download, install and login to the app. From the location permission we asked, we can track the app and the user. So we are creating a range circle about 1.5km rounded by the IoT device near the railway crossing. And if we detect a vehicle user who use our app, entered that range circle from tracking, we start continuing tracking that app user. So if only that app user is approaching towards the way of railway crossing, warning message alerts are sent to that user that railway crossing is nearby and train is approaching on that way via the app. Otherwise, warning message alerts are not sent to the app user. That means if we detect there is no movement of the app, or the app is going away from the railway crossing, or the app is not in a road near a railway crossing, that alert messages are not sent to that particular user of the app. Then we store the vehicle patterns of that particular app user, vehicle patterns mean the roads which he or she used to drive near that particular railway crossing. And also, we store the data of date, time as well. And then store these data in some real-time database. We are planning to "Firebase-Real time database" to store these data at the moment. So after collecting these kind of vehicle patterns of the same app user more times, we pre-process that data to analyse. Then analyse that particular data to make predictions. Then we do predictions that whether this app user is likely to cross the railway crossing on a particular day and particular time when a train is approaching towards the railway crossing. If it is, we send alert messages to that user not only through the app, but also through the SIM card as well. The reason for that is, for some reason that user's mobile data might be off for some reason, or that particular user might closed the app. So even though on such scenarios, the user will be notified via SMS alert notifications.

The proposed Train detection and Alert System is expected to improve safety of the citizen who cross the railway crossings significantly. It will help to reduce the number of accidents and fatalities near railway crossings. Overall, this research project has the potential to make a significant contribution to the citizens of Sri Lanka who are crossing the railway crossings, safety and development of them. And ultimately, improving the quality of life for citizens in Sri Lanka.

1.4. Security analysis for the Train Alert and Detection System

Railroad transit plays a crucial role in the economic and social development of nations, and ensuring the safety and productivity of railroad operations is of utmost importance. A significant part of this responsibility lies with the railway authorities, who are tasked with developing and implementing effective train tracking and detection systems. These systems are essential for monitoring and regulating train movements accurately and in real-time, allowing for timely intervention in case of any issues or emergencies.

As technology advances, there has been a growing need for more mobile, user-friendly, and accessible train tracking and detection systems. This has led to the development of a mobile application for train tracking and detection, which provides real-time management of train movements. The application is designed to track the location, speed, and direction of the train using cutting-edge technology such as GPS, machine learning, and computer vision. It also provides customers with warnings and notifications in case of any irregularities or crises.

However, ensuring the security of the train tracking and detection system is equally crucial. Security analysis is the process of evaluating the security of a system, network, or organization to identify vulnerabilities, threats, and risks. In the context of the train tracking and detection system, security analysis is crucial to identify potential security weaknesses and develop strategies to mitigate them. This involves implementing various security controls such as firewalls, intrusion detection systems, access controls, and encryption, as well as developing incident response plans and conducting regular security audits.

The security analysis process includes several methodologies such as vulnerability assessments, penetration testing, threat modeling, and risk assessments. Vulnerability assessments are used to identify potential vulnerabilities in the system, while penetration testing is used to test the system's security by attempting to exploit these vulnerabilities. Threat modeling involves identifying potential threats to the system and developing strategies to mitigate them, while risk assessments are used to assess the likelihood and potential impact of an attack on the system.

Overall, security analysis is a crucial component of any comprehensive security program, and helps organizations to proactively identify and address security vulnerabilities and threats, reducing the risk of cyber attacks and protecting sensitive information. In the context of the train tracking and detection system, security analysis is essential to ensure the safety and security of train operations, protect customer data, and safeguard critical infrastructure.

In conclusion, the development of a mobile application for train tracking and detection is a significant step towards ensuring the safety and productivity of railroad operations. However, it is crucial to ensure the security of the system by conducting regular security analysis and implementing various security controls. This will not only help to prevent cyber attacks but also protect critical infrastructure, thereby safeguarding the economic and social development of nations.

The installation of a train tracking and alarm system is a difficult process that combines delicate data with essential infrastructure. Regular security studies and the implementation of strict security measures are essential given the frequency and sophistication of cyber attacks in today's digital environment.

Comprehensive risk evaluations, vulnerability assessments, and penetration testing should all be included in security analyses. These analyses are necessary to find any system flaws that malevolent actors might try to exploit. The system can be strengthened against cyber assaults, data breaches, and other security breaches that might jeopardise the functionality and safety of railway operations by proactively addressing these weaknesses.

Multiple layers of security measures must be put in place in order to safeguard the integrity of vital infrastructure. Strong encryption protocols to protect data while it is in transit and at rest, access controls to restrict system access to authorised personnel only, intrusion detection systems to quickly identify and address security incidents, and routine security updates to fix known vulnerabilities are a few examples of these controls.

Additionally, it is necessary to develop user education and awareness programmes to make sure that both railway employees and passengers are aware of the value of cybersecurity and their part in maintaining a secure environment. Initiatives in the area of training can aid in reducing risks related to human error, social engineering attacks, and insider threats.

A security lapse in a train tracking and alarm system might have far-reaching effects on not only the railway sector but also the economic and social advancement of entire countries. A region or nation's stability and prosperity can be negatively impacted by disruptions in rail operations since they can result in large financial losses, safety risks, and supply chain disruptions.

The introduction of a mobile application for train tracking and detection, in conclusion, offers significant advantages in terms of productivity and safety, but it must be backed by a consistent commitment to cybersecurity. In order to protect this vital system, it is crucial to conduct regular security studies, implement strong security measures, and provide continual user education. We can make sure that the train tracking and warning system not only improves railway operations but also contributes to the overall security and resilience of our countries' infrastructure and development by taking these steps.

1.5. Sending the flooded messages from the IOT device for the SIM users who are within a 1.5km radius

2. BACKGROUND AND LITERATURE

2.1. Background

The concept of providing solutions for the safety of people who are crossing railway crossings has been around for a long time, but the development of modern technology has made it possible to implement more effective solutions. The first research on providing solutions for the safety of people who are crossing railway crossings may date back to the late 19th century. At that time, many accidents occurred due to the lack of proper signalling systems, and there was a growing need to improve safety measures. The first documented effort to address this issue dates back to the 1870s when the first automatic signal system was installed in England. Since then, many countries have made significant progress in developing and implementing safety measures for railway crossings. However, with the increasing number of vehicles and railway crossings, new and innovative solutions are still being researched and developed to improve safety for pedestrians and drivers alike.

2.1.1. Develop a system that utilizes GSM trackers on trains and IoT devices at railway crossings to predict and alert potential blind spots on the train

Railway transportation has been a crucial mode of transportation for many years, and it continues to play a vital role in the global economy. However, with the increase in train traffic and the complexity of the railway network, there has been a growing concern about the safety of railway crossings.

Railway crossings are places where rail tracks intersect with roadways or pedestrian walkways. These crossings are vulnerable to accidents, especially when drivers or pedestrians are not aware of the presence of a train. In many cases, accidents occur due to the presence of blind spots on trains, which make it difficult for drivers to see obstacles or hazards at railway crossings.

Traditionally, safety measures at railway crossings have focused on the installation of warning signs, lights, and barriers. However, these measures have proven to be inadequate, and accidents at railway crossings continue to occur. To address this issue, there is a need for innovative technologies that can enhance safety measures at railway crossings.

Advancements in technology have enabled the development of sophisticated systems that can enhance railway safety. The use of IoT devices and GSM trackers in the railway industry has been gaining popularity in recent years. IoT devices can provide real-time data on train movements, enabling railway personnel to monitor train traffic and respond quickly to any potential hazards. GSM trackers can provide accurate data on train movements, enabling railway operators to optimize their operations and improve the efficiency of railway transportation.

The proposed system that utilizes GSM trackers on trains and IoT devices at railway crossings to predict and alert potential blind spots on the train is an innovative solution to enhance railway safety. The system's predictive capabilities and real-time tracking of trains can help prevent accidents by alerting railway personnel of potential blind spots in advance. This system has the potential to revolutionize the railway industry by making it safer and more efficient, and it is a significant step towards achieving the goal of zero accidents at railway crossings.

The use of advanced technologies such as GSM trackers and IoT devices in the railway industry has been the subject of several studies in recent years. Researchers have explored the potential of these technologies to enhance railway safety and improve the efficiency of railway transportation.

One study conducted by Li et al. (2019) focused on the use of IoT devices to enhance railway safety [5]. The study proposed a system that utilized IoT devices to collect data on train movements and transmit it to a central server for analysis. The system was designed to provide real-time updates on train movements and alert railway personnel of any potential hazards. The study concluded that the use of IoT devices could significantly enhance railway safety by providing accurate data on train movements.

Another study conducted by Zhang et al. (2020) focused on the use of machine learning algorithms to predict potential hazards at railway crossings [6]. The study utilized data from IoT devices at railway crossings and combined it with data on train movements from GSM trackers. The study concluded that the use of machine learning algorithms could improve the accuracy of hazard prediction at railway crossings.

The use of GSM trackers to monitor train movements has also been the subject of several studies. One study conducted by Manoharan et al. (2018) focused on the use of GSM trackers to optimize railway transportation [7]. The study proposed a system that utilized GSM trackers to provide real- time data on train movements, enabling railway operators to optimize their operations and improve the efficiency of railway transportation.

2.1.2. Sending alerts to the users via the app and predicting the likelihood of crossing the railway-crossing on a given day

It is mainly focused on some points when studying about the literature review background for this relevant component.

1. App Based safety solutions – Explored literature on app–based safety solutions for the users who are crossing the railway crossings regularly.

2. Vehicle Tracking – Explored literature on vehicle tracking near a railway crossing.

3. Alerts Sending - Explored literature on alerts sending to the users when they are approaching towards a railway crossing and a train is coming on that way as well.

4. Predictive Analysis - Explored literature on previous researches that do some predictive analysis and predicting whether some vehicle is likely to cross the railway crossing.

5. Real-time Databases - Research literature on real-time databases and their use in applications. This could include information on different types of real-time databases.

The use of real-time data analysis and prediction has been increasingly popular in the transportation industry, with the aim of improving the overall efficiency and safety of the transport systems. In the context of train tracking, various methods such as GPS, RFID, and GSM have been used to monitor train movement and location. However, the use of these technologies has its limitations, and real-time prediction of train location remains a challenge.

In recent years, machine learning and artificial intelligence techniques have been used to improve train tracking and prediction but not to analyse vehicle near the railway crossings. These techniques have the potential to analyse and process large amounts of real-time data, leading to more accurate predictions of train location and arrival times. One promising approach is the use of recurrent neural networks (RNNs), which have shown success in predicting the future trajectory of trains.

2.1.3. Security analysis for the Train Alert and Detection System

Systems for tracking and detecting trains are essential for ensuring the safety and efficiency of train operations. These systems rely on a variety of technologies, such as GPS, radar, and other sensors, to track and detect trains as they travel along railway lines. With real-time information about train locations and schedules, train operators can control traffic and prevent crashes, making these systems critical components of modern transportation infrastructure.

A train tracking system typically consists of a network of sensors and communication devices that enable real-time monitoring of train locations, speeds, and other important operational parameters. However, the security of these systems is also essential to ensure their safety and reliability. A security analysis of train tracking systems can help identify vulnerabilities and potential attack vectors, assess the potential impact of a security breach on the system and its users, and identify ways to mitigate security risks.

There are several potential security threats to train tracking systems that should be considered during security analysis. Malicious actors may attempt to gain unauthorized access to the system or its data through hacking or social engineering attacks. Physical attacks, such as vandalism or sabotage of the system's infrastructure, can also pose a significant threat. Denial of service (DoS) attacks, which overwhelm the system's sensors or communication devices, can also disrupt operations. Insider threats, such as employees with access to the system who may misuse their privileges or engage in malicious behavior, should also be considered.

To mitigate security risks, it is important to implement appropriate security controls and procedures. This may include access controls to limit who can access the system and its data, encryption to protect data in transit and at rest, and regular security audits and testing to identify vulnerabilities and potential attack vectors. It is also important to have a plan in place for responding to security incidents, such as a breach or attack, to minimize their impact and prevent further damage.

When designing a train monitoring and detection smartphone application, it is crucial to consider privacy and security concerns. These applications may collect critical information about train schedules, positions, and freight, making it essential to provide safe and private systems that safeguard user data and guarantee the security of train operations. This may include implementing secure data storage and transmission protocols, as well as p
The development of alarm and tracking systems for trains has closely mirrored technological improvements. The widespread installation of telegraph lines alongside railway tracks in the late 19th century made it possible for railway staff and the central dispatching office to communicate more effectively. This was a significant step in the direction of centralised management and coordination.

Additional technological developments, such as the use of computers for train dispatching and signalling, occurred in the later half of the 20th century. More effective resource allocation and train scheduling were made possible by computerised systems. Additionally, the use of GPS technology offered a revolutionary method for real-time train tracking by delivering accurate, satellite-based location data.

Due to the fusion of mobile technology, wireless connectivity, and data analytics, train tracking and alarm systems have rapidly advanced in the twenty-first century. As a result of these advancements, sophisticated systems that offer real-time monitoring, predictive analytics, and proactive alarms have emerged, revolutionising the railway sector.

In order to provide very accurate location data, modern train tracking and alarm systems use GPS technology, cellphone networks, and onboard sensors. Real-time updates on train arrivals, departures, and delays are provided by passenger-focused mobile applications, revolutionising the passenger experience and raising general happiness.

Train operations now heavily rely on predictive analytics, which is backed by machine learning and data analytics. By taking into account elements like weather, maintenance schedules, and previous data, these systems are able to anticipate future disruptions. Operators may improve operating efficiency, optimise schedules, and cut down on delays thanks to this predictive capability.

In the case of emergencies or unanticipated problems, alert mechanisms inside these systems ensure prompt contact with train operators, maintenance teams, and passengers. By giving key information to stakeholders, this proactive strategy not only improves safety but also enables quick and coordinated reactions to incidents, minimising the impact on both operational effectiveness and safety.

As a result of tremendous development from their modest beginnings, train tracking and alarm systems have transformed into crucial tools in the modern railway business. These systems are prepared to play an increasingly more crucial role in maintaining the sustainability, effectiveness, and safety of rail transport as technology develops, helping to determine the course of this important method of transport in the future.

Significant technological advancements in train tracking were made during the 20th century. Rail operations were optimised by the introduction of centralised traffic control systems that allowed for remote control of signals and switches. By reducing crashes and permitting accurate tracking of train locations, track circuits—which could identify the presence of trains on a track section—improved safety. Providing clear and transparent privacy policies and user agreements.

In conclusion, train tracking systems are critical components of modern transportation infrastructure, and security analysis is an essential component of their operation. By identifying and mitigating potential security risks, we can ensure the safety and reliability of these critical transportation systems. Additionally, when designing train monitoring and detection smartphone applications, privacy and security concerns must be carefully considered to protect user data and guarantee the security of train operations.

2.2. Literature Review

2.2.1. Develop a system that utilizes GSM trackers on trains and IoT devices at railway crossings to predict and alert potential blind spots on the train

The proposed system that utilizes GSM trackers on trains and IoT devices at railway crossings to predict and alert potential blind spots on the train is an innovative solution to enhance railway safety. To develop this system, a literature survey was conducted to gain insights from previous research on similar topics. The literature survey revealed that there have been several studies on the use of IoT devices and GSM trackers in the railway industry. These studies have focused on various aspects of railway transportation, including safety, efficiency, and optimization.

One study conducted by Nangalia and Shah (2018) explored the potential of IoT devices to enhance railway safety. [8] The study proposed a system that utilized IoT devices to monitor train movements and alert railway personnel of any potential hazards. The study concluded that the use of IoT devices could significantly enhance railway safety by providing real-time data on train movements.

Another study conducted by Sutrisno et al. (2019) [9] focused on the use of GSM trackers to monitor train movements and optimize railway transportation. The study proposed a system that utilized GSM trackers to provide real-time data on train movements, enabling railway operators to optimize their operations and improve the efficiency of railway transportation.

In addition, a study conducted by Sharma et al. (2020) [10] explored the use of machine learning algorithms to predict potential hazards at railway crossings. The study utilized data from IoT devices and GSM trackers to develop a predictive model that could alert railway personnel of potential hazards in advance.

The literature survey also revealed that there have been several studies on the use of IoT devices and GSM trackers in other industries, such as logistics and supply chain management. These studies have shown that the use of advanced technologies can improve efficiency, reduce costs, and enhance safety.

Overall, the literature survey indicates that the proposed system that utilizes GSM trackers on trains and IoT devices at railway crossings to predict and alert potential blind spots on the train is a feasible solution that builds on the findings of previous studies. The system has the potential to significantly enhance railway safety and improve the efficiency of railway transportation.

2.2.2. Sending alerts to the users via the app and predicting the likelihood of crossing the railway-crossing on a given day

The study by A. A. T. P. De Silva University of Colombo School of Computing [11] has introduced a Real-time location based crowdsourcing train tracking android application to enhance the effectiveness and efficiency of public train transportation. The proposed application is developed by combining Global Positioning System (GPS), mobile computing, and crowdsourcing technologies to gain information from the passengers and provide visual positioning using Google map in real-time. Additionally, it predicts the estimated time to arrival (ETA) of a train to any given railway station for better user experience and for better admin management.

The study [12] introduces a method to detect the train using a vibration sensor on the railway track. The Arduino microcontroller was chosen to detect the train and send the signal to the level crossing subsystem (LCS) as it has enough input lines to connect the relevant sensors and actuators. An alarm system with lights is used with an IoT system that consists of Radio Frequency (RF) receiver, Raspberry Pi Model 3 and cameras.

"Evaluation of Safety Degree at Railway Crossings in Order to Achieve Sustainable Traffic Management: A Novel Integrated Fuzzy MCDM Model" is a research done in 2021 by Aleksandar Blagojević, Sandra Kasalica, Željko Stević, Goran Tričković and Vesna Pavelkić. [13] They have done a research to address railway crossings collisions by measuring the analysis of safety degree on some railway crossings. They have used some mathematical formulas to count this analysis of safety degree.

"Traffic Signal Operations Near Highway-Rail Grade Crossings" [14] is a another research done in USA by Transportation Research Board and National Research Council in 1999. They mainly focused on the alarm and alert systems in the railway crossings. It's an automatic railway crossing safety system that means, when a train is approaching, automatically safety options near the railway crossings like alarms, red lights are activated.

"An Automatic Railway Level Crossing System with Crack Detection" is another research done to minimize the collisions in the railway crossings. [15] This researched have used IoT devices like microcontrollers, Ultra Sonic Sensors, GSM Modules to detect cracks and barrier control when a train is approaching. GSM module is used to send alert messages to relevant authorities when a crack is detected. It is said that this system have a high accuracy of 93% reliability of crack detection. And also they send alerts of railway crossings gates and alarm status like whether they are on or off when a train is approaching to the relevant authorities.

The studies [16] and [17] have used IoT-based systems to automate railway crossings safety system controls. In the [16] system, Train current location information is sent to users with a link to see the current location of the train via SMS in this system. Arduino Uno, server motors, and sonar sensors have been utilized to implement the automated railway crossings bars in this system. In the [17] system, manually operated railway crossing gates are transformed into a fully automatic system operated through wirelessly by the stationmaster using Arduino Uno and ultrasonic sensors. And, this particular system detects the number of persons and vehicles that remain on the tracks after closing of gates by the help of Open-Source Computer Vision Library (OpenCV) with a buzzer and LED bulb connected to it to give alerts during opening and closing of gates.

"RDMNS.LK" is a Sri Lankan-developed mobile application to alert railway alerts in Sri Lanka. [18] This application shows the users the live locations of the trains when a train is approaching. And also, this app alerts the user whether trains are delayed or not and if the train is delayed, how much time the train is delayed. This app also shows us train station contact numbers, ticket prices from Colombo Fort and so many more details related to trains in Sri Lanka. This application is available for both Android and IOS currently.

"Smart Railway Crossing Surveillance System" is another research project done to address railway crossings collisions. [19] This system addressed railway collisions by displaying whether the railway gate is opened or closed based on the train passing, the most recent gate closure timestamp, and each time when the gate is closed, a centralized platform for knowing the condition of the gate, traffic density, and intrusion detection near the railway gate. Based on the experimental results, this automated railway crossing assistance outperformed human-assisted railway crossings, resulting in an efficient and cost-effective solution for traffic regulation and avoidance of accidents at railway gates. They have used an Arduino IoT system for this system as well.

The study [20] aimed to aware the train driver about what is blocking the railway path and automatically suggest him to follow safety procedures. And also, the other thing focused on is preventing accidents in wildlife areas in railway areas in Sri Lanka. Also, to create an inter-connected smart grid to provide real-time train movement tracking across the country and make way to insert an AI system to automate entire train operations. An IoT system has been utilized to identify train movements and some formulas have been used to calculate some calculations.

The literature survey highlights several research studies related to the safety of railway crossings. Researchers have proposed various solutions, including mathematical models to evaluate the safety degree of railway crossings, automatic safety systems activated by approaching trains, IoT- based systems for crack detection and barrier control, mobile applications to alert users about train locations and delays, and smart surveillance systems to monitor railway gates and traffic density. These studies provide valuable insights into addressing railway crossing collisions and can help inform the development of effective safety measures.

2.2.3. Security analysis for the Train Detection and Alert System

Train tracking systems play a crucial role in the efficient management of rail transportation systems. These systems enable operators to monitor the movement of trains, manage their routes, and ensure passenger safety. System analysis is an essential process in the design and implementation of train tracking systems. It involves the identification of the requirements, constraints, and functional elements of the system, as well as its performance, reliability, and safety. In this paper, we provide an overview of some of the key studies on train tracking systems in system analysis.

Train Tracking Systems based on Wireless Sensor Networks

One of the significant challenges in train tracking systems is the accurate detection of the position, speed, and direction of trains. Liu et al. (2016) propose a train tracking system based on wireless sensor networks (WSNs) to address this challenge. The system uses a distributed architecture and a variety of sensors to detect and track trains. The authors evaluate the performance of the system using simulation and experimental results. The study shows that the WSN-based train tracking system can accurately detect the position and movement of trains, making it a promising technology for rail transportation systems.

Train Tracking Systems based on RFID Technology

Radio-frequency identification (RFID) technology is another technology that can be used to track trains. Hu et al. (2019) propose a train tracking system based on RFID technology. The system uses RFID tags to track the position and movement of trains, and the authors analyze its performance in terms of accuracy, reliability, and scalability. The study shows that the RFID- based train tracking system is reliable and scalable, making it a suitable technology for large rail transportation systems.

Train Tracking Systems based on GPS Technology

Global positioning system (GPS) technology is widely used for location tracking in various industries, including rail transportation. Yilmaz et al. (2018) propose a real-time train tracking system based on GPS technology. The system uses GPS receivers to track the position and speed of trains, and the authors evaluate its performance in terms of accuracy, latency, and reliability. The study shows that the GPS-based train tracking system can provide accurate and reliable real- time location tracking, making it a promising technology for rail transportation systems.

Train Tracking Systems based on Machine Vision Technology

Machine vision technology is another technology that can be used to track trains. Zhou et al. (2020) propose a train tracking system based on machine vision technology. The system uses cameras to detect and track trains, and the authors evaluate its performance in terms of accuracy, speed, and reliability. The study shows that the machine vision-based train tracking system can provide accurate and reliable train tracking, making it a promising technology for rail transportation systems.

Train Tracking Systems based on IoT Technology

The Internet of Things (IoT) is a network of connected devices that can be used to collect and transmit data. Wei et al. (2017) propose a train tracking system based on IoT technology. The system uses a variety of sensors and communication technologies to track the position and movement of trains, and the authors evaluate its performance in terms of scalability, reliability, and security. The study shows that the IoT-based train tracking system can provide accurate and reliable train tracking, making it a promising technology for rail transportation systems.

Understanding Train Tracking Systems

A. Definition and Significance

Designed to facilitate real-time monitoring, control, and optimization of train movements within a railway network. These systems serve a multifaceted purpose, encompassing the enhancement of operational efficiency, the preservation of passenger safety, and the optimization of resource utilization.

B. Evolutionary Trajectory

The historical development of train tracking systems stretches back to the early 20th century when rudimentary methods such as telegraphy were used to establish communication between trains and stations. The evolution has been marked by monumental strides, transforming from manual operations to the contemporary era of fully automated, computer-based systems.

System Analysis in the Context of Train Tracking

A. Requirement Identification

The foundational step in system analysis revolves around the comprehensive identification of the myriad requirements intrinsic to a train tracking system. This entails a careful delineation of the functionalities that the system must offer, the scale of railway operations it must support, and the specificity required to cater to diverse railway network needs.

B. Constraint Evaluation

Equally pivotal is the scrutiny of constraints that govern the design and operation of the train tracking system. These constraints may encompass financial limitations, regulatory prerequisites, and technological confines. Thorough constraint evaluation ensures the practicality and feasibility of the system.

C. Functional Elements

System analysis necessitates a meticulous delineation of the functional elements that constitute the train tracking system. This encompasses not only the diverse hardware components such as sensors, communication devices, and control centers but also the intricate software ecosystem responsible for data processing and analysis.

Conclusion

The studies discussed in this paper highlight the diverse range of technologies and approaches that can be used to track trains. Each technology has its strengths and weaknesses, and the choice of technology depends on the specific requirements of the rail transportation system. System analysis is an essential process in the design and implementation of train tracking systems. It enables operators to identify the requirements, constraints, and functional elements of the system, as well as its performance, reliability, and safety. By evaluating the performance and reliability of train tracking systems under various conditions, operators can ensure the safe and efficient operation of rail transportation system.

3. RESEARCH GAP

This research project aims to bridge this gap by developing a Train Detection System for citizens via the sim or the app, which can be accessed by a wider range of users. The proposed system utilizes GSM trackers, prediction patterns, flooded message systems and other real-time data sources to provide up-to-date information of trains and provide alerts about trains, which can be used by the app users and people near the railway crossings to avoid accidents with trains in the railway crossings via the app, sim or IoT devices near the railway crossing.

3.1. Develop a system that utilizes GSM trackers on trains and IoT devices at railway crossings to predict and alert potential blind spots on the train

Features / Research	IOT Device	Train Tracking	Message sending To IOT device	Analyzing the Trains Past Patterns	Predict the duration for the non signal railway crossing	Alert the IOT device	Train Tracking
Proposed Component	✓	✓	✓	✓	✓	✓	✗
Automatic Railway Crossing System with Crack Detection [1]	✗	✓	✗	✗	✗	✓	✓
RDMNS.LK: LIVE Train Alerts [2]	✗	✗	✗	✗	✗	✗	✓
Smart Railway Crossing Surveillance System [3]	✓	✗	✓	✗	✗	✗	✓

Table 1: Table of the research Gap - System that utilizes GSM trackers on trains and IoT devices at railway crossings to predict and alert potential blind spots on the train

The component is "To develop a system that utilizes GSM trackers on trains and IoT devices at railway crossings to predict and alert potential blind spots on the train.". The main objective of this is to analyse the past patterns of trains and predict and send alert to the IOT device on crossing when trains come to blind spot. And sub objectives are,

- To develop a GSM tracker that can transmit its location to the IoT device on the railway crossing.

- To enhance the safety measures at railway crossings by developing an IoT device that can detect approaching trains and send an alert to the nearby devices.

- To integrate the IoT device and GSM tracker to establish a communication link to send an alert when a train approaches the crossing. To investigate the feasibility and effectiveness of using manual training of datasets to predict the location of a lost GSM tracker signal in the railway industry.

- To evaluate the performance of the integrated system and its impact on improving the safety measures at railway crossings.

3.2. Sending alerts to the users via the app and predicting the likelihood of crossing the railway-crossing on a given day

Research / Features	Proposed Component	Automatic Railway Crossing System with Crack Detection [14]	RDMNS.LK: LIVE Train Alerts [18]	Smart Railway Crossing Surveillance System [19]
Mobile App	✔	✘	✔	✔
Vehicle Tracking	✔	✔	✘	✘
Alerts Sending to the Users	✔	✘	✔	✔
Analyzing the Past Vehicle Patterns	✔	✘	✘	✘
Predict the likely to Cross the Railway Crossing on particular Day	✔	✘	✘	✘
Send Alerts to Relevant Authorities	✘	✔	✘	✘
Train Tracking	✘	✔	✔	✔

Table 2: Table of Research Gap - Sending alerts to the users via the app and predicting the likelihood of crossing the railway-crossing on a given day

This component is “Sending alert messages only to the app users who are approaching a railway crossing”. The main objective of this is to analyse the past patterns of vehicles and predict if they are likely to cross the railway crossing on a given day. And sub objectives are to identify the vehicles that are moving towards the railway crossing, and send alerts only to them through the app, To track the location of the user's vehicle when they enter the range circle near the railway crossing, Store the tracked patterns of the user's vehicle and send a notification to the user's SIM card if they are predicted to cross the railway crossing on a particular day, even if they are not using the app or have mobile data turned off.

According to the literature survey above conducted, it is evident that Train detection system for citizens who cross the railway crossing is a much-needed implementation since it is really rare to find researches that targets the safety of citizen who cross the railway crossings. Up to now in the researches that were conducted under this scope, we can see that they have covered the following areas.

- Train Detection and live location showing
- Railway crossings crack detection
- Railway crossings Alarms and alert systems Improvements
- Measuring the analysis of safety degree on railway crossings
- Railway Crossing Surveillance System
- Train alerts and delay time showing
- Train detection using IoT devices

3.3. Security analysis for the Train Detection and Alert System

Features	IOT Device Security	Security of train tracking devices	Notifications	Mobile application security
Proposed Component	✓	✓	✓	✓
Automatic railway crossing system with crack Detection	✗	✓	✗	✓
RDMNS.LK: live Train alerts	✗	✗	✗	✗
Smart railway crossing surveillance system	✓	✗	✓	✓

Table 3: Table of the Research Gap - Security Analysis for Train Detection and Alert System

Train tracking systems play a critical role in ensuring the efficiency, reliability, and safety of rail transportation systems. With the increasing adoption of technology, there has been significant research in train tracking systems. However, there are still some research gaps that need to be addressed to improve the accuracy, reliability, and safety of these systems. This paper highlights some of the potential research gaps that need to be addressed in train tracking systems in system analysis.

Train tracking systems are integral components of modern rail transportation networks, serving as the linchpin for ensuring operational efficiency, reliability, and passenger safety. As the world continues its relentless march towards technological advancement, train tracking systems have undergone significant research and development efforts. These endeavors have undeniably propelled the field forward, resulting in more sophisticated and capable systems. Yet, despite these notable strides, there remain conspicuous research gaps that beckon further exploration and innovation. These gaps, if meticulously addressed, hold the promise of not only enhancing the accuracy, reliability, and safety of train tracking systems but also revolutionizing the very landscape of rail transportation. In this comprehensive analysis, we delve into these intriguing research gaps, shedding light on their significance within the domain of train tracking systems, specifically in the context of system analysis. By unraveling and examining these gaps, we aim to galvanize the efforts of researchers, engineers, and policymakers, inspiring them to embark on transformative journeys that will define the future of rail transportation.

Data Integration and Fusion:

A critical research gap lies in the seamless integration and fusion of data from various sensors and sources within train tracking systems. While advancements have been made, achieving real-time, comprehensive data fusion that ensures the utmost accuracy in train position and movement tracking remains a formidable challenge. Research should focus on novel techniques for data integration, incorporating machine learning and artificial intelligence to refine the process continuously.

Cybersecurity and Resilience:

With the ever-increasing connectivity of train tracking systems, they become prime targets for cyberattacks. Research in this domain must concentrate on bolstering cybersecurity measures to safeguard critical rail infrastructure. Moreover, developing robust resilience strategies that can swiftly respond to and recover from cyber incidents is of paramount importance.

Cost-Effective Solutions:

The cost-effectiveness of train tracking solutions is vital, particularly for smaller rail networks and developing regions. Research should endeavor to identify and implement cost-effective technologies and approaches that do not compromise safety or reliability, thereby ensuring equitable access to advanced tracking capabilities.

Ethical and Legal Considerations:

Ethical and legal aspects related to data privacy, surveillance, and passenger safety require meticulous examination. Research efforts should address these concerns and propose ethical frameworks and legal guidelines that govern the collection, storage, and use of data within train tracking systems, ensuring responsible and transparent practices.

Integration of Multiple Technologies:

Most existing studies on train tracking systems focus on a single technology such as GPS, RFID, or machine vision. There is a need to investigate the integration of multiple technologies to improve the accuracy and reliability of train tracking. For instance, combining GPS with machine vision technology can improve the accuracy of train tracking in areas with poor GPS coverage.

Real-time Analysis of Train Data:

Many train tracking systems provide real-time data on the position and movement of trains. However, there is a need for real-time analysis of this data to detect anomalies, identify potential risks, and optimize train routes in real-time. For instance, real-time analysis can help detect potential collisions or derailments and take preventive action.

Cybersecurity and Privacy Issues:

Train tracking systems often involve the use of sensitive data such as train schedules and passenger information. Therefore, there is a need to investigate the cybersecurity and privacy issues associated with these systems and develop appropriate measures to protect data. For instance, ensuring secure communication channels between the train tracking system and other systems can prevent cyber-attacks.

Optimization of Train Routes:

Many train tracking systems provide real-time data on the position and movement of trains. However, there is a need to investigate the use of this data for optimizing train routes. This could involve developing algorithms that take into account factors such as train speed, capacity, and track conditions to minimize travel time and improve efficiency. For instance, optimizing train routes can reduce delays and improve the overall efficiency of the rail transportation system.

Standardization of Testing Procedures and Metrics:

There is a need for standardized testing procedures and metrics for evaluating the reliability and safety of train tracking systems. Standardization can help ensure that train tracking systems meet the required safety and reliability standards. For instance, developing standardized testing procedures can help evaluate the accuracy of train tracking systems and identify areas that need improvement.

Privacy and Security:

Mobile applications for railway detection and tracking may gather private data regarding train routes, stops, and freight. To create safe and private applications that safeguard user information and guarantee the security and safety of railway operations, research is required. This could involve investigating the use of encryption technologies and developing privacy policies that protect user data.

Conclusion:

Train tracking systems play a critical role in ensuring the efficiency, reliability, and safety of rail transportation systems. However, there are still some research gaps that need to be addressed to improve the accuracy, reliability, and safety of these systems. This paper has highlighted some of the potential research gaps that need to be addressed in train tracking systems in system analysis. Addressing these research gaps can help improve the efficiency, reliability, and safety of rail transportation systems. Even though train monitoring systems have made significant progress, these research gaps serve as sobering reminders of how far they still have to go. By filling in these gaps, a new era of rail transport with unmatched precision, dependability, and safety may be ushered in. These collective exploration and innovation trips must be undertaken by the scientific community, industry stakeholders, and policymakers in order to eventually shape the future of rail transport for future generations.

4. RESEARCH PROBLEM

4.1. Develop a system that utilizes GSM trackers on trains and IoT devices at railway crossings to predict and alert potential blind spots on the trains

The main research problem addressed by the proposed system that utilizes GSM trackers on trains and IoT devices at railway crossings to predict and alert potential blind spots on the train is the need to enhance railway safety. Despite the implementation of various safety measures, railway accidents still occur, and blind spots on the train remain a significant safety concern.

Blind spots on the train refer to areas where the train driver's vision is obstructed, making it difficult to see potential hazards such as vehicles, pedestrians, or obstacles on the tracks. Blind spots can occur at railway crossings or other areas where the train passes close to buildings or other structures.

The proposed system aims to address this problem by utilizing advanced technologies such as GSM trackers and IoT devices to predict and alert potential blind spots on the train. By providing real-time data on train movements and potential hazards, the system can alert railway personnel and train drivers of any potential blind spots and prevent accidents from occurring.

In addition to enhancing railway safety, the proposed system also aims to improve the efficiency of railway transportation. By providing real-time data on train movements and potential hazards, the system can enable railway operators to optimize their operations and improve the efficiency of railway transportation.

The research problem addressed by the proposed system can be further broken down into several steps:

- Step 1: Identifying the safety concerns and blind spots on the train

The first step in addressing the research problem is to identify the safety concerns and blind spots on the train. This involves analyzing the various factors that contribute to railway accidents, such as human error, equipment failure, and environmental factors, and identifying the areas where train drivers have limited visibility.

- Step 2: Evaluating the effectiveness of existing safety measures

The next step is to evaluate the effectiveness of existing safety measures such as automatic train control and positive train control in addressing the safety concerns and blind spots on the train.

This involves analyzing the data on accidents and near-misses to identify the areas where safety measures are most effective and areas where further improvements are needed.

- Step 3: Exploring the use of advanced technologies

The third step is to explore the use of advanced technologies such as GSM trackers and IoT devices to enhance railway safety and address blind spots on the train. This involves researching the potential benefits and drawbacks of using these technologies and identifying the technical requirements and challenges of implementing them.

- Step 4: Developing a system that utilizes GSM trackers and IoT devices

The fourth step is to develop a system that utilizes GSM trackers on trains and IoT devices at railway crossings to predict and alert potential blind spots on the train. This involves designing the hardware and software components of the system, integrating the system with existing safety measures, and testing the system in a controlled environment.

- Step 5: Evaluating the effectiveness and cost-effectiveness of the system

The final step is to evaluate the effectiveness and cost-effectiveness of the proposed system in enhancing railway safety and improving the efficiency of railway transportation. This involves analyzing the data on accidents and near-misses before and after implementing the system and comparing the costs and benefits of implementing the system with other safety measures.

By following these steps, the proposed system that utilizes GSM trackers on trains and IoT devices at railway crossings to predict and alert potential blind spots on the train can effectively address the research problem and enhance railway safety.

4.2. Sending alerts to the users via the app and predicting the likelihood of crossing the railway-crossing on a given day

According to Sri Lanka Tweet (2019), Every 3 hours, a person dies from a road accident in Sri Lanka. [4] And also, Every 3 days, a child dies from a road accident in Sri Lanka. According to a study on railway-roadway level crossing safety, which is done by Kulasingham Ragulan and Niranga Amarasingha, there are so many railway crossing collisions happen annually in Sri Lanka. [2]. And also, The 'datareportal.com' website shows that Population : Mobile Phone ratio is 1.4 per person in Sri Lanka. [21] That means the total number of phones are greater than the population of Sri Lanka. Even though the count of phones are higher than the population of Sri Lanka, there is no proper ways to use these phones to address the safety of people in Sri Lanka.

These railway crossing collisions has taken so many human lives as well as lot's of property damage. Today, there are several web applications and mobile applications about trains in Sri Lanka. But there are no proper IT based solutions to address the safety of people who cross the railway crossings, specially in Sri Lanka.

- How to enhance the safety of passengers and drivers near railway crossings?

One of the major research problems addressed by my component is the need to enhance the safety of passengers and drivers near railway crossings. Railway crossings can be dangerous places, and accidents involving vehicles and trains can result in severe injury or even death. My component seeks to address this issue by providing real-time message alerts about the approaching trains when a user is coming towards a railway crossing via the app and SMS messages.

- How to provide the predictions with highest accuracy rate?

Another research problem to be addressed in my component is the accuracy of vehicle likelihood of crossing the railway crossing when a train is approaching prediction. Inaccurate predictions could lead to mis timely alerts, which could cause passengers and drivers to become desensitized to the warnings provided by the app, resulting in a decreased level of safety. Additionally, the accuracy of vehicle prediction of crossing the railway is critical to the overall effectiveness of the this research project, and any improvements to the accuracy of the system could have a significant impact on the overall safety of railway crossings. Predictions should be made in higher accuracy for better safety of people.

- How to sort the vehicles which are approaching toward the railway crossing?

Since we are developing an app to alert the users, it is important to sort the users who are moving toward the railway crossing only. Otherwise, if the alerts are sent every time to the user when a train is approaching, it can be like a headache for the user. That will be a major User experience issue.

- How to develop the mobile application with a better user experience?

The app should be a optimized and should be in a simple and attractive UI for a better User Experience. Since we are targeting to commercialize this, the app will play a major rule since it will directly be connected with the user.

- How to integrate with real-time databases?

Since the most parts of the whole system connected with a database, it is important to connect them with a real-time database because outputs should be delivered so faster since this address the safety of people. So connecting the component parts with a better real-time database is so much important.

- How to send alert messages not only via the app but also to the SIM card as well?

The app user might not use the mobile app due to various reasons. Like whether he/ she may turned off the mobile data or his/her 3G or 4G connection’s signal strength is not enough to connect to the internet at the moment. When this kind of scenario happens, it’s important to send alerts to the user somehow.

4.3. Security Analysis for Train Detection and Detecting System

Introduction:

The railway industry plays a crucial role in the transportation sector, with millions of people and tons of goods transported daily via trains. Train tracking, monitoring, and detection systems are essential components of railway operations that ensure the safety and efficiency of train travel. This paper highlights some research problems that need to be addressed to improve train tracking, monitoring, and detection systems' performance. Every day, millions of people and tonnes of cargo are transported thanks to the railway industry, which is a key pillar of the global transportation sector. The performance of train tracking, monitoring, and detecting systems is crucial for ensuring the efficiency, dependability, and safety of rail traffic. These systems serve as the engine that drives railway operations, ensuring the accurate location and condition of trains, keeping an eye on vital indicators, and spotting any dangers. The continuously changing nature of transportation and technology, however, offers both benefits and difficulties. In order to further improve the functionality of train tracking, monitoring, and detecting systems and eventually contribute to safer and more effective rail travel, this paper explains some important research issues that require attention.

- Problem 1: Developing a reliable real-time train tracking and detection system for low- connectivity environments

One of the significant challenges facing train tracking and detection systems is developing a reliable real-time system that can provide accurate and up-to-date train location and schedule information in areas with limited network connectivity. This is particularly challenging for mobile applications that rely on network connectivity to provide real-time information to users. However, many railway lines run through remote or rural areas with poor network coverage, making it difficult to access real-time information about train locations and schedules.
To address this research problem, researchers need to develop mobile applications that can function effectively in low-connectivity environments, using a range of technologies such as offline caching, edge computing, and satellite communication. These technologies can enable the mobile application to store and access critical information locally, reducing dependence on network connectivity. Developing such an application will help ensure that train operators, cargo handlers, and passengers can access accurate and up-to-date information about train schedules and locations, regardless of their location or network connectivity. This, in turn, will improve the safety and efficiency of train operations, particularly in remote or rural areas where connectivity is limited.

- Problem 2: Predicting and managing train delays and disruptions accurately

Delays and disruptions are common in train operations, and they can have a significant influence on both train safety and efficiency. Predicting and managing train delays and disruptions accurately is, therefore, another crucial research problem that needs to be addressed to improve train tracking and detection systems.
To overcome this problem, researchers need to create predictive models that can predict delays and interruptions in real-time using a variety of data sources such as weather, traffic, and train timetables. These models could assist train operators in proactively managing train operations, rerouting trains, and notifying passengers of delays and disruptions, thereby improving safety and efficiency. In addition to prediction models, research is required to develop effective communication and notification systems capable of informing train operators, freight handlers, and passengers about delays and disturbances in real-time. These systems could leverage a variety of communication channels, such as cellphone notifications, SMS messaging, and public address systems. Solving this research issue would improve train operations' safety and efficiency by allowing train operators to better manage delays and interruptions.

- Problem 3: Developing a system to prevent collisions between trains and vehicles at railway crossings

Railway crossings are critical points in train operations where vehicles and trains often intersect, and collisions at these intersections can have catastrophic consequences. Detecting and preventing collisions between trains and vehicles at railway crossings is, therefore, another crucial research problem that needs to be addressed to improve train tracking and detection systems.

To address this research problem, researchers need to develop accurate and reliable detection technologies that can detect and notify train operators of vehicles or pedestrians at railway crossings. In addition to detection technologies, research is also needed to develop effective collision prevention systems that can prevent collisions between trains and vehicles at railway crossings. These systems could incorporate automated warning systems, such as visual and auditory alarms, that notify cars and pedestrians of an oncoming train. Solving this research issue would improve train operations greatly and lessen the likelihood of incidents between trains and cars at railway crossings. This will serve to protect the lives and property of those who live and work near railway crossings while also improving overall train safety and efficiency.

Conclusion:

The railway industry's continued growth and success rely heavily on the performance and innovation of train tracking, monitoring, and detection systems. As technology advances and transportation needs evolve, these systems face an array of research challenges. Addressing these challenges is essential to ensure safer, more efficient, and environmentally responsible rail travel. Researchers, engineers, and policymakers must collaborate to overcome these hurdles, fostering a future where train tracking and monitoring systems are at the forefront of railway safety and efficiency. This concerted effort will not only enhance the rail transportation sector but also contribute to sustainable and resilient transportation systems worldwide. Developing reliable and effective train tracking, monitoring, and detection systems is crucial for improving train operations' safety and efficiency. In conclusion, the railway industry's continued growth and success rely heavily on the performance and innovation of train tracking, monitoring, and detection systems. As technology advances and transportation needs evolve, these systems face an array of research challenges. Addressing these challenges is essential to ensure safer, more efficient, and environmentally responsible rail travel. Researchers, engineers, and policymakers must collaborate to overcome these hurdles, fostering a future where train tracking and monitoring systems are at the forefront of railway safety and efficiency. This concerted effort will not only enhance the rail transportation sector but also contribute to sustainable and resilient transportation systems worldwide.

5. RESEARCH OBJECTIVES

5.1. Main Objective

Provide an IT-based solution to address the safety of the citizen in Sri Lanka from the railway – crossings collisions.

In Sri Lanka, like many other countries, railway-crossing collisions pose a significant threat to the safety of citizens. These accidents occur when vehicles or pedestrians attempt to cross railway tracks without proper caution, often resulting in tragic consequences. To address this pressing issue, implementing an IT-based solution can significantly enhance safety measures and minimize the risk of collisions atrailway crossings.

One innovative approach involves the use of Intelligent Transportation Systems (ITS) that utilize technologies such as machine learning, location tracking, flooded messages, and communication networks. These technologies can be deployed at railway crossings to detect the presence of vehicles and pedestrians in real-time.

Furthermore, leveraging mobile applications can empower citizens to contribute to safety efforts. A dedicated mobile app can provide real-time information of trains, crossing locations, and safety guidelines. Citizens can receive alerts and warnings on their smartphones, ensuring they are well-informed and cautious while approaching railway crossings.

By implementing such an IT-based solution, Sri Lanka can significantly enhance the safety of its citizens at railway crossings. Utilizing advanced technologies and engaging citizens through mobile applications can create a comprehensive and proactive approach to prevent accidents, ultimately saving lives and ensuring a safer transportation environment for everyone.

5.1.1. Develop a system that utilizes GSM trackers on trains and IoT devices at railway crossings to predict and alert potential blind spots on the train

To develop a system that utilizes GSM trackers on trains and IoT devices at railway crossings to predict and alert potential blind spots on the train.

The main objective of the proposed system that utilizes GSM trackers on trains and IoT devices at railway crossings to predict and alert potential blind spots on the train is to enhance railway safety and improve the efficiency of railway transportation.

Specifically, the objectives of the proposed system include:

Addressing blind spots on the train by providing real-time data on train movements and potential hazards.

Alerting railway personnel and train drivers of any potential blind spots to prevent accidents from occurring.

Improving the efficiency of railway transportation by providing real-time data on train movements and potential hazards to enable railway operators to optimize their operations.

Integrating with existing safety measures such as automatic train control and positive train control to enhance the overall safety of the railway system.

Evaluating the effectiveness and cost-effectiveness of the proposed system in enhancing railway safety and improving the efficiency of railway transportation.

By achieving these objectives, the proposed system can significantly enhance railway safety and improve the efficiency of railway transportation, ultimately benefiting both the railway industry and the public.

5.1.2. Sending alerts to the users via the app and predicting the likelihood of crossing the railway-crossing on a given day.

- Analyse the past patterns of vehicles and predict if they are likely to cross the railway crossing on a given day.

The ultimate main objective of this component is to analyse the past patterns of drive patterns near a particular railway crossing of the vehicles of the app users and predict if that particular user is likely to cross the railway crossing when a train is approaching. To analyse the vehicle moving patterns, we need to track them. For that, the mobile application come to play. We ask the permission of location when the user is registering to the app. Then we store the vehicle movement patterns in the particular database. After storing vehicle movement patterns of same user more times, we can analyse that patterns and make a prediction of whether this user is likely to cross the railway crossing on the train approaching time or not. More and more vehicle patterns, increases the accuracy of the prediction as well.

5.1.3. Security Analysis for the Train Deteaction and Alert System

To design and develop a security analyzer for train tracking and detection systems that can detect and prevent cyber-attacks, and to evaluate its effectiveness and usability in real-world scenarios.

The objective of this research proposal is to design and develop a security analyzer for train tracking and detection systems that can detect and prevent cyber-attacks. With the increasing reliance on technology in railway operations, the potential for cyber-attacks on train tracking and detection systems has become a growing concern. The proposed security analyzer aims to address this concern by providing a proactive and effective approach to cyber security in railway operations.

The first step in achieving this objective would be to conduct a thorough review of the existing literature on cyber security in railway operations, including the current threats, vulnerabilities, and potential impacts of cyber-attacks on train tracking and detection systems. This review would help identify the strengths and weaknesses of current approaches and provide a basis for developing an innovative and effective solution.

Next, the research would involve designing and developing a security analyzer for train tracking and detection systems that can detect and prevent cyber-attacks. The security analyzer would be designed to analyze network traffic and data logs to identify potential threats and vulnerabilities, and to take proactive measures to prevent cyber-attacks from occurring.

Once the security analyzer has been developed, the research would focus on evaluating its effectiveness and usability in real-world scenarios. This would involve conducting field tests on a select number of train tracking and detection systems and monitoring the system's performance over a set period of time.

During the field tests, the research would collect data on the system's ability to detect and prevent cyber-attacks, as well as its impact on the overall performance of the train tracking and detection system. The data collected would be analyzed using statistical methods to determine the security analyzer's overall effectiveness and identify any areas for improvement.

In addition to evaluating the security analyzer's effectiveness, the research would also assess its usability and ease of implementation. This would involve conducting surveys and interviews with railway operators and other stakeholders to determine their perceptions of the security analyzer and their willingness to adopt it in their operations.

To ensure the security analyzer is effective and meets the needs of all stakeholders, the research would also investigate the ethical and legal implications of implementing the system. This would involve proposing strategies for protecting user privacy and ensuring data security, as well as complying with relevant laws and regulations.

Finally, the research would identify areas for future research and development, including the integration of other security measures and technologies, the development of advanced algorithms for threat detection and prevention, and the expansion of the system to other countries and regions. This would help ensure that the security analyzer remains relevant and effective in a rapidly changing technological landscape.

In conclusion, the objective of this research proposal is to design and develop a security analyzer for train tracking and detection systems that can detect and prevent cyber-attacks, and to evaluate its effectiveness and usability in real-world scenarios. By providing a proactive and effective approach to cyber security in railway operations, the proposed security analyzer can help mitigate the potential risks and impacts of cyber-attacks on train tracking and detection systems.

5.2. Sub Objectives

- Develop an User–friendly application that provide real-time data of trains and send alerts to the user properly.
- Gather and analyze data to train models.

- Provide the security and the privacy to the application as appropriate.
- To evaluate the effectiveness and reliability of the developed system through testing and validation.

5.2.1. Develop a system that utilizes GSM trackers on trains and IoT devices at railway crossings to predict and alert potential blind spots on the train

Sub Objective 1: To enhance the safety measures at railway crossings by developing an IoT device that can detect approaching trains and send an alert to the nearby devices.

The sub-objectives of developing an IoT device to enhance the safety measures at railway crossings include:

- Designing and developing an IoT device that can detect the approach of trains using various sensors and algorithms.
- Developing a communication protocol between the IoT device and nearby devices, such as smartphones or signal posts, to alert them of the approaching train.
- Ensuring that the IoT device can operate in different environmental conditions, including extreme temperatures, rain, and fog.
- Integrating the IoT device with existing railway infrastructure, such as signal posts and control centers, to enhance the overall railway safety system.
- Testing the IoT device in a controlled environment to ensure that it meets the required safety standards and is effective in preventing accidents at railway crossings.

Sub Objective 2: To develop a GSM tracker that can transmit its location to the IoT device on the railway crossing.

The sub-objective of developing a GSM tracker that can transmit its location to the IoT device on the railway crossing involves the following steps:

- Designing and developing a compact and efficient GSM tracker that can be easily installed on the train

- Incorporating the necessary sensors and communication modules into the tracker to enable it to transmit its location data and communicate with the IoT device at the railway crossing.

- Testing the GSM tracker to ensure its reliability, accuracy, and ability to operate in different environmental conditions.

- Developing a communication protocol between the GSM tracker and the IoT device that is secure, reliable, and can operate seamlessly across different networks and devices.

- Integrating the GSM tracker with the IoT device on the railway crossing to enable real- time tracking of train movements and location data.

Achieving this sub-objective, the proposed system can provide real-time data on the location and movements of trains, which can enhance the safety and efficiency of railway operations. The system can also enable the detection of potential blind spots on the train and provide an early warning to the driver and nearby personnel at the railway crossing, thus reducing the risk of accidents. Additionally, the system can help in tracking lost or stolen trains and aid in the recovery of valuable cargo.

Sub Objective 3: To integrate the IoT device and GSM tracker to establish a communication link to send an alert when a train approaches the crossing. To investigate the feasibility and effectiveness of using manual training of datasets to predict the location of a lost GSM tracker signal in the railway industry.

The sub-objectives of integrating the IoT device and GSM tracker to establish a communication link and investigating the feasibility and effectiveness of using manual training of datasets to predict the location of a lost GSM tracker signal in the railway industry are:

- Designing and developing an IoT device that can detect the presence of a train in the vicinity of the railway crossing and transmit the information to the GSM tracker.

- Establishing a communication protocol between the IoT device and the GSM tracker that is reliable and can operate in various environmental conditions and situations.

- Developing an algorithm that can predict the location of a lost GSM tracker signal based on a manual training dataset.

- Conducting experiments to evaluate the effectiveness of the algorithm in predicting the location of a lost signal and assessing the reliability and accuracy of the communication link between the IoT device and the GSM tracker.

- Optimizing the system by refining the algorithm used to predict the location of a lost signal, improving the communication link, and integrating the system with existing railway safety measures.

By achieving these sub-objectives, the proposed system can enhance the safety of railway crossings by providing real-time data on train movements, detecting potential blind spots, and predicting the location of lost GSM tracker signals. This can significantly reduce the risk of accidents and improve the overall safety of the railway transportation system. Additionally, the system can improve the efficiency of railway operations by providing accurate and timely information on train movements, helping to reduce delays and improve scheduling.

Sub Objective 4: To evaluate the performance of the integrated system and its impact on improving the safety measures at railway crossings.

Sub Objective 5: Analyse the vehicle movement patterns and do predictions with high accuracy

- To make better predictions, Machine Learning algorithm should be chosen with the highest accuracy rate and it should be trained well.

The sub-objective of evaluating the performance of the integrated system and its impact on improving the safety measures at railway crossings involves the following steps:

- Conducting field trials to test the performance of the integrated system in a real-world setting.

- Collecting data on the performance of the system, including its ability to detect approaching trains and blind spots, and its accuracy in predicting the location of lost GSM tracker signals.

- Analyzing the collected data to evaluate the effectiveness of the system in enhancing safety measures at railway crossings.

- Identifying any limitations or challenges in the system's performance and suggesting potential solutions to overcome them.

- Comparing the results of the field trials with the established safety standards and guidelines for railway crossings to determine the impact of the system on improving safety measures.

By achieving this sub-objective, the proposed system can provide empirical evidence of its effectiveness in enhancing safety measures at railway crossings. The evaluation results can also inform policymakers and railway operators on the potential benefits and limitations of the system, and guide future improvements and developments of the system. Ultimately, the goal is to enhance the safety and efficiency of railway operations, reduce the risk of accidents, and ensure the smooth and reliable transport of goods and passengers.

5.2.2. Sending alerts to the users via the app and predicting the likelihood of crossing the railway-crossing on a given day.

Sub Objective 1: Identify the vehicles that are moving towards the railway crossing, and send alerts only to them through the app –

Vehicles which are moving toward the railway crossings are identified by the location permission of the particular user’s app. When it is identified that particular user is moving towards the railway crossing way only from the tracked location, alert messages are sent. Otherwise, that user is not moving, the user is going away from the railway crossing, like that kind of scenarios, alert messages are not sent to that particular user. This makes the User Experience of the user about app better.

Sub Objective 2: Send notification alerts to the user's SIM card if they are predicted to cross the railway crossing on a particular day, even if they are not using the app or have mobile data turned off –

This is a sub-objective should be accomplished after accomplishing the main objective. The app user might not use the mobile app due to various reasons. Like whether he/ she may turned off the mobile data or his/her 3G or 4G connection's signal strength is not enough to connect to the internet at the moment. When this kind of scenario happens, it's important to send alerts to the user to the SIM card as well.

In accomplishing the main objective, below sub-objectives need to be satisfied.

Sub Objective 3: Track the location of the user's vehicle when they enter the range circle near the railway crossing.

Sub Objective 4: Store the tracked patterns of the user's vehicle – Tracked patterns of vehicles should be stored in a real-time database.

Sub Objective 5: Analyze the vehicle movement patterns and do predictions with high accuracy – To make better predictions, Machine Learning algorithm should be chosen with the highest accuracy rate and it should be trained well.

5.2.3. Security Analysis for the Train Detection and Alert System

1. Determine the exact sorts of cyber-attacks to which train monitoring and detection systems are exposed and assess the possible consequences for railway operations and public safety. This sub-objective would entail examining current literature and interviewing railway experts and security specialists in order to determine the most likely and high-risk cyber threats to train monitoring and detection systems.

2. Create a comprehensive security framework and architecture for train tracking and detection systems that incorporates best cyber security and risk management practices. This sub-goal would entail creating a security framework and architecture that tackles the identified cyber risks while taking into consideration the particular characteristics and requirements of train monitoring and detection systems.

3. Create a prototype security analyzer for railway monitoring and detection systems capable of detecting and preventing cyber threats. This sub-goal would entail creating a software solution that can monitor network traffic and data logs to detect possible risks and vulnerabilities, as well as taking proactive actions to avoid cyber assaults. The security analyzer should be made scalable and adaptable to various train tracking and detection systems.

4. Conduct field tests and simulations to assess the security analyzer's effectiveness and usability in real-world scenarios. This sub-goal would entail implementing the security analyzer on a limited number of train tracking and detection systems and analyzing its performance over a certain time period. Field tests should be performed to assess the security analyzer's capacity to identify and prevent cyber threats, as well as its influence on the overall functioning of the train monitoring and detection system.

5. Evaluate the ethical and legal consequences of deploying the security analyzer and offer ways for maintaining user privacy and data security. This sub-goal would entail completing a legal and ethical analysis of the security analyzer and recommending actions to ensure compliance with applicable laws and regulations. It would also entail creating rules and processes to preserve user privacy while simultaneously maintaining the security of sensitive data acquired by the security analyzer.

6. METHODOLOGY

After having some discussions with our supervisor, co-supervisor and research panel members, requirements were identified, some were changed and finally finalized them. Background and literature survey was done for my component in the area of alerting the people who cross the railway crossings. And then found some implementations has been done before which are looks like bit similar world-wide. As it is evident in the literature survey, that there has not been done a specific IT based system to address the safety of people on the road, when they are moving towards a railway crossing and when a train is approaching as well.

The methodology for creating the proposed train detection and alert system includes the following steps:

1. Literature Survey
2. Data Collecting
3. Model Building
4. Model Testing
5. UI Designing
6. Security Designing
7. Security Testing
8. Integrating
9. Testing
10. Deployment of the Prototype

6.1. System Architecture

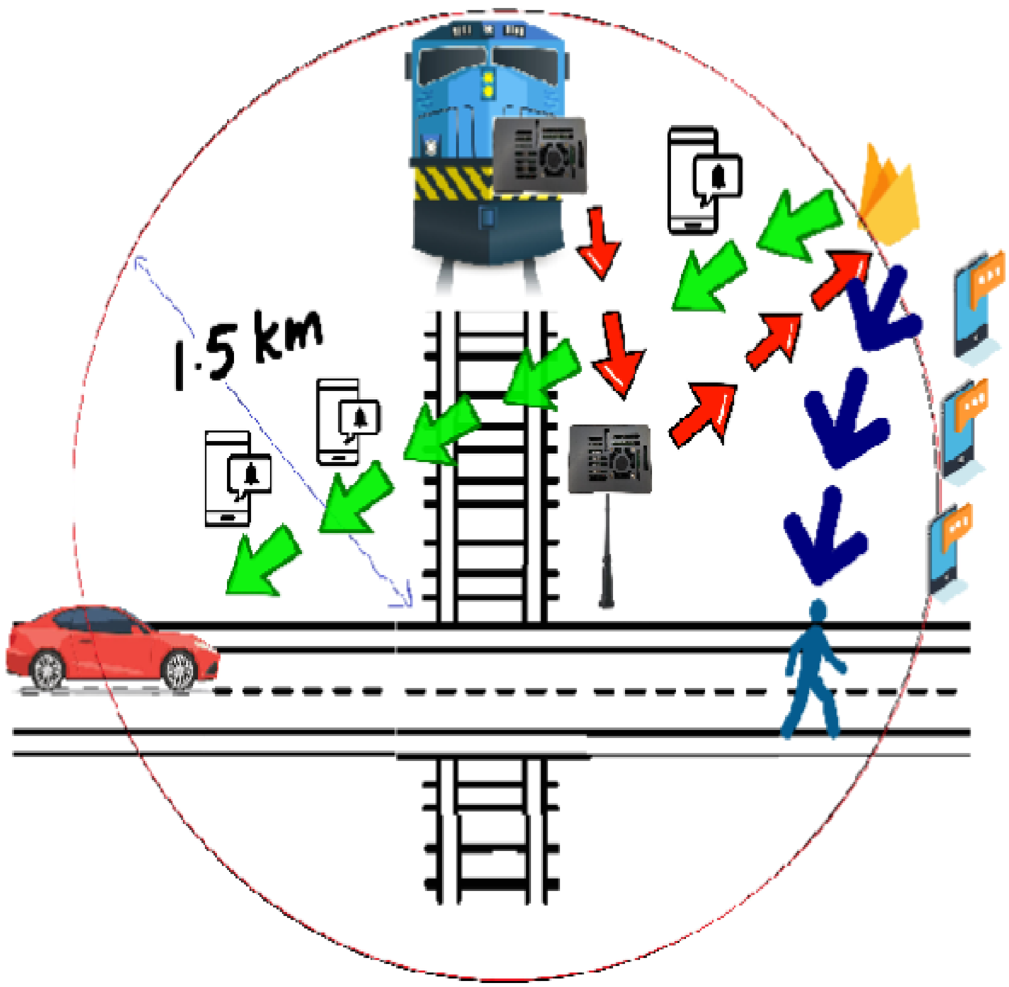


Figure 2: Overview of the System

Figure 2 shows the basic overview of the system. The system incorporates a IoT device with tracking device installed on the train, which transmits its precise location to the IoT device near the railway crossing. When there is a blind spot when the train is moving, that means when the signal is lost, the location of the train is predicted from the machine learning algorithm and transmitted to the IoT device near the railway crossing. Furthermore, the IoT device near the railway crossing is triggered when the train approaches a designated radius near a railway crossing. All the data are transmitted to the cloud database system (Firebase real-time database) from the IoT device near the railway crossing. When a vehicle user who installed our mobile app, is driving towards a railway crossing side and a train is approaching on that particular time as well, alert messages are sent to their mobile app indicating that a railway crossing is approaching and a train is approaching nearby as well, if the mobile application user is inside the range circle. Alert messages are sent as SMS messages to the pedestrians who are near the railway crossing if they are inside the range circle as flooded messages.

6.1.1. System Architecture - Develop a system that utilizes GSM trackers on trains and IoT devices at railway crossings to predict and alert potential blind spots on the train.

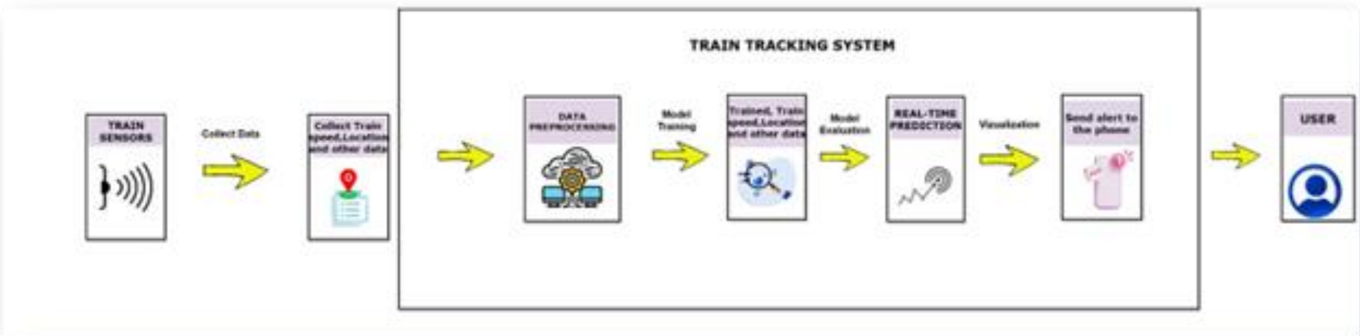


Figure 3: Overall System Architecture Diagram of the Component

The system architecture for the proposed solution involves three main components: the IoT device, the GSM tracker, and the central server.

The IoT device is placed at the railway crossing and is responsible for detecting the approaching trains through sensors such as cameras, radar, and/or infrared sensors. Once a train is detected, the IoT device sends a signal to the central server through a wireless communication protocol such as Wi-Fi, Bluetooth, or LoRa.

The GSM tracker is installed on the train and is responsible for transmitting its location to the central server via cellular network communication. The GSM tracker is also connected to the central server via a wireless communication protocol such as Wi-Fi or Bluetooth to enable communication between the tracker and the server.

The central server acts as the main processing and control unit for the system. It receives signals from the IoT device and the GSM tracker, processes the data to predict potential blind spots, and sends alerts to nearby devices and/or the train driver. The server also stores and analyzes the data to provide insights on system performance and potential improvements.

Overall, this architecture allows for real-time monitoring and prediction of potential blind spots on trains, enhancing safety measures at railway crossings.

The methodology for developing the system that utilizes GSM trackers on trains and IoT devices at railway crossings to predict and alert potential blind spots on the train can be broken down into several steps:

- **Requirement Analysis:** The first step in developing this system is to conduct a comprehensive requirement analysis. This includes identifying the stakeholders, their requirements, and the system's functional and non-functional requirements.

- **Hardware Design:** The next step is to design the hardware components of the system. This includes designing the IoT device to detect approaching trains and the GSM tracker to transmit its location to the IoT device.

- **Software Design:** After designing the hardware components, the software design of the system needs to be developed. This includes developing the communication protocol between the IoT device and the GSM tracker, designing the algorithm to predict and alert potential blind spots on the train, and creating the user interface for the system.

- **Implementation:** Once the hardware and software designs are complete, the next step is to implement the system. This involves building the physical hardware components and programming the software.

- **Testing:** After implementation, the system needs to be thoroughly tested to ensure that it meets the requirements and works as intended. Testing includes unit testing, integration testing, system testing, and acceptance testing.

- **Evaluation:** Once the system has been tested and implemented, the final step is to evaluate its performance. This involves measuring the system's effectiveness in predicting and alerting potential blind spots on the train, evaluating the system's impact on improving safety measures at railway crossings, and identifying any areas for improvement.

By following this methodology, the system that utilizes GSM trackers on trains and IoT devices at railway crossings to predict and alert potential blind spots on the train can be developed effectively and efficiently.

6.1.2. System Architecture - Sending alerts to the users via the app and predicting the likelihood of crossing the railway-crossing on a given day.

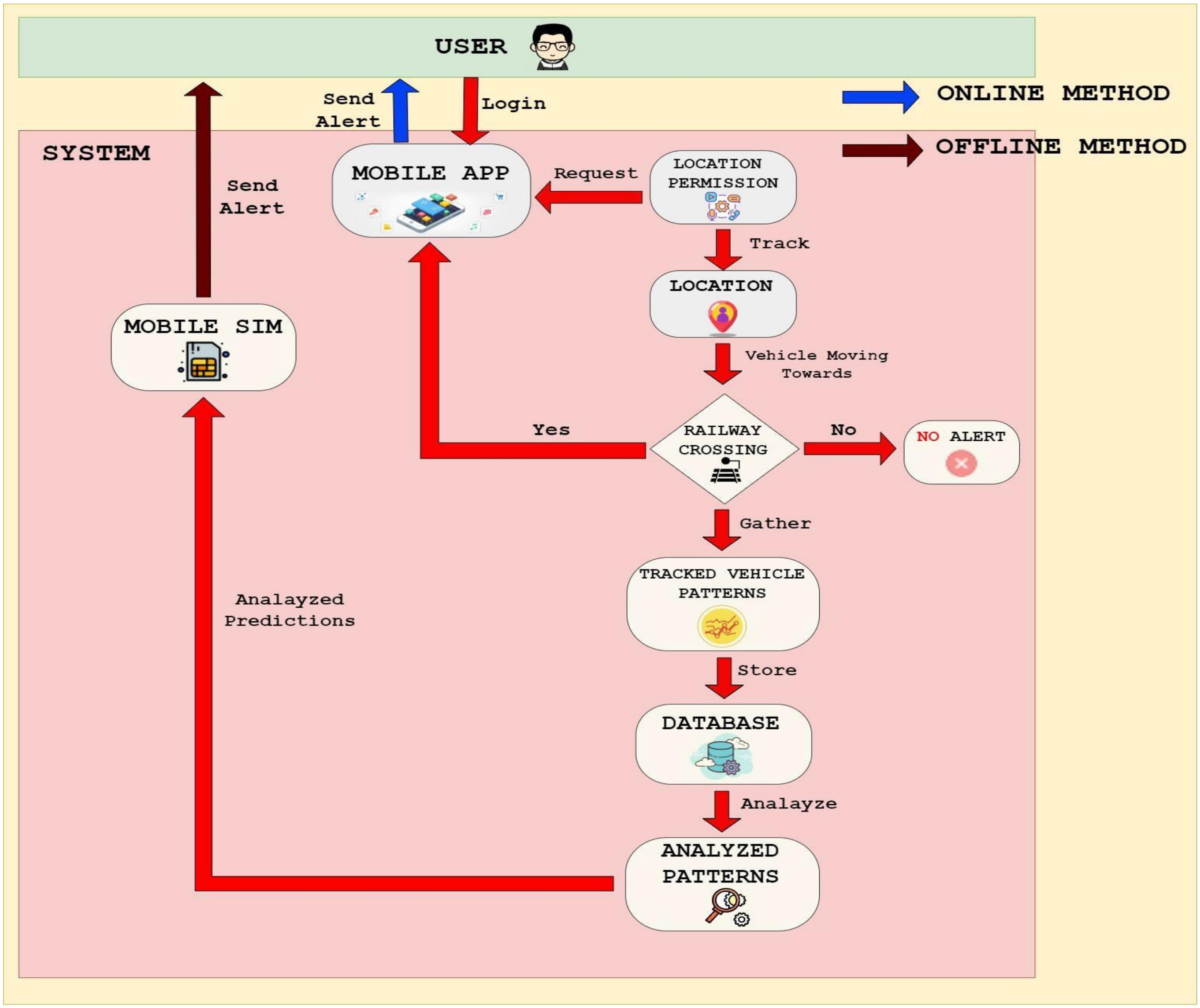


Figure 4: Overall System Architecture Diagram for the Component

The location permission is asked from the user when he/she logs to the app to track him/her. If that particular user is moving towards the railway crossing when a train is approaching, alerts messages are sent to him/her. Otherwise, alert messages are not sent to that particular user. That is measured using the location tracking. Then the vehicle movement patterns are collected of that particular user, and they are stored in the real-time database. Then the patterns which are in the database are analyzed and give alerts to the particular user form the analyzed patterns via the app (Online Method) and the SIM card (Offline Method) as well.

6.1.3. System Architecture – Security Analysis for Train Detection and Alerting System

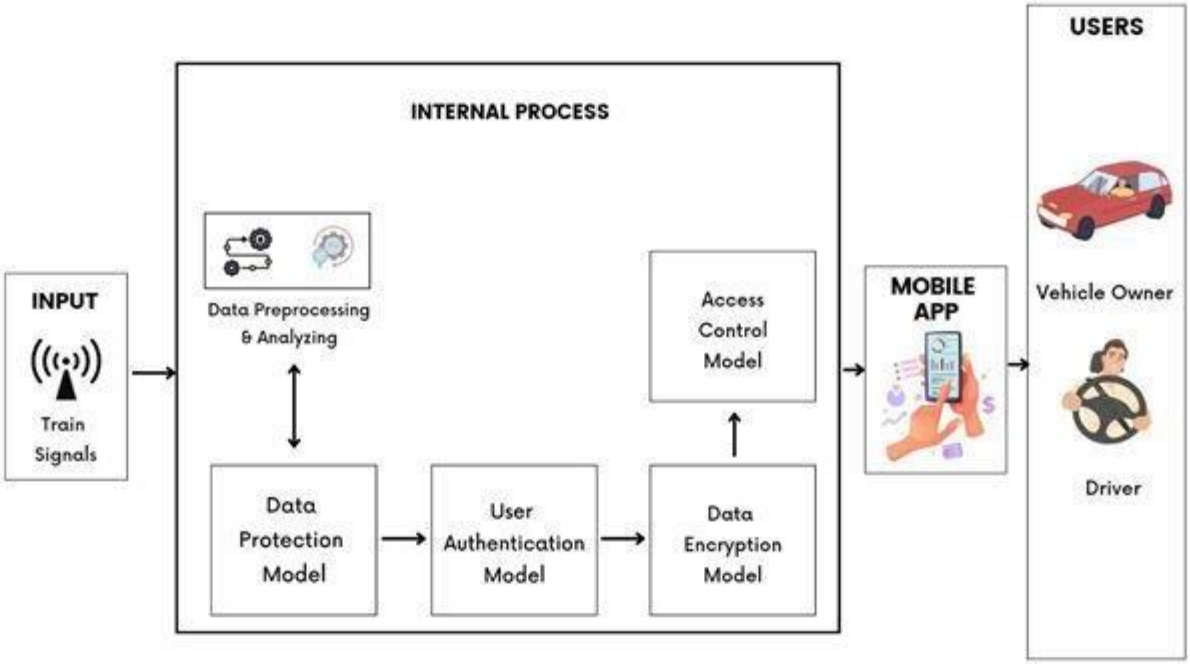


Figure 5: Overall System Architecture Diagram of the Component

The location permission is asked from the user when he/she logs to the app to track him/her. If that particular user is moving towards the railway crossing when a train is approaching, alerts messages are sent to him/her. Otherwise alert messages are not sent to that particular user. That is measured using the location tracking. Then the vehicle movement patterns are collected of that particular user and they are stored in the real-time database. Then the patterns which are in the database are analyzed and give alerts to the particular user form the analyzed patterns via the app (Online Method) and the SIM card (Offline Method) as well. The procedure of asking the user for location permission as they enter into the application is an important first step in guaranteeing their safety and offering individualised services. By obtaining user permission for location monitoring, the system is able to precisely follow the user's movements and customise alerts accordingly. The technology uses the user's current location to send timely alarm messages when a train is approaching a railway crossing and the user is close by. This improves safety at risky times.

Location data is used for more than just instant notifications; it also serves as the foundation for a thorough examination of a user's vehicle movement patterns. These patterns are continuously gathered and safely saved in a live database, creating a priceless archive of historical information. The system may gain insights from the user's behaviour and give notifications that are not only timely but also extremely relevant to their unique travel habits and routes thanks to this database, which acts as the basis for further analysis.

The system's analytical part analyses the recorded movement patterns using sophisticated algorithms and machine learning methods. It carefully examines these patterns to find trends, outliers, and possible threats. This analysis takes into account a number of variables, including the user's usual routes, speed, and proximity to railway crossings at various times of the day and week. Each user will receive context-aware information and cautions thanks to the system's ability to generate personalised notifications using the insights gleaned from this analysis.

Both the application itself (online approach) and the user's SIM card (offline method) are used to provide these warnings. The Online Method makes sure that customers get instant notifications through the app when they are online, giving them up-to-the-minute safety information. In addition, the Offline Method makes use of the messaging features of the SIM card to reach people even when they do not have an active internet connection. This dual strategy ensures that users are alerted and cautioned about potential railway crossing threats regardless of their connectivity state and maximises the reach and efficacy of warning delivery.

6.2. Data Collection Process

Data is collected via the train schedules, train tables to predict the location of the train. Vehicle location, time, date and relevant data are collected via the mobile application. The mobile numbers of the users are collected when registering to the mobile application. That mobile numbers are used to send normal SMS messages as flooded messages as well.

6.3. Users of the System

This research effort is primarily aimed at a target audience that crosses the railway crossings, especially the people who cross the unsafety railway crossings. Since it is difficult to identify the railway crossings in the situations like nights, heavy rains, when there are bends just before the railway crossings; this system is targeted for these occasions mainly.

6.4. The input of the System

The vehicle user just needs to download the mobile application and register by providing the mobile number of him/her. The mobile application asks the location permission from the mobile to track the location of the vehicle user. The mobile user must give the location permission to use the mobile application.

The pedestrians who are expecting the alert messages as normal SMS messages need to register to the system via SMS registration.

6.5. The output of the System

When the train is near to the railway crossing, the IoT device near the railway crossing is triggered with a sound So, the people near the railway crossing are alerted that a train is approaching nearby.

When the train is approaching towards the railway crossing and the vehicle user who is using the mobile application is inside the range circle surrounded by the IoT device near the railway crossing, and if the vehicle moving towards the railway crossing, the alert messages are sent to the user via the mobile application that a railway crossing is nearby, and a train is approaching.

The people who are near the railway crossing are notified via the flooded SMS messages that train is coming And they are alerted from the prediction as well.

6.6. UI designing

The UI (User Interface) design of the mobile application is really simple since the mobile application has no many functions and the only task of the mobile application is to notify the user that trains is coming and railway crossing is nearby.

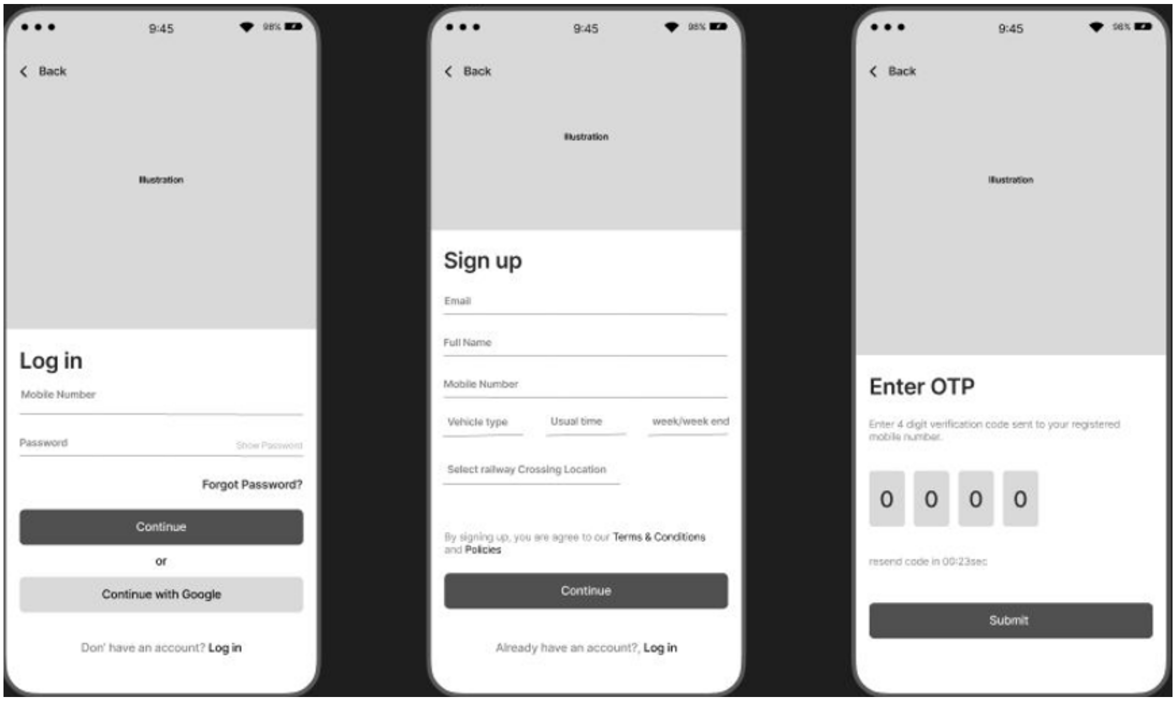


Figure 6: User Interfaces of the Mobile Application

Figure 4 shows some user interfaces of the mobile application. The user should allow location permission when using the app since the location of the vehicle user is tracked by the system. The user can log into the application if he/she is already registered to the app. If the user is not registered to the app, he/she should register to the app by providing necessary details. Since the location permission is granted form user when using the app, the vehicle user locations are always tracked by the system to take decisions and alert the user.

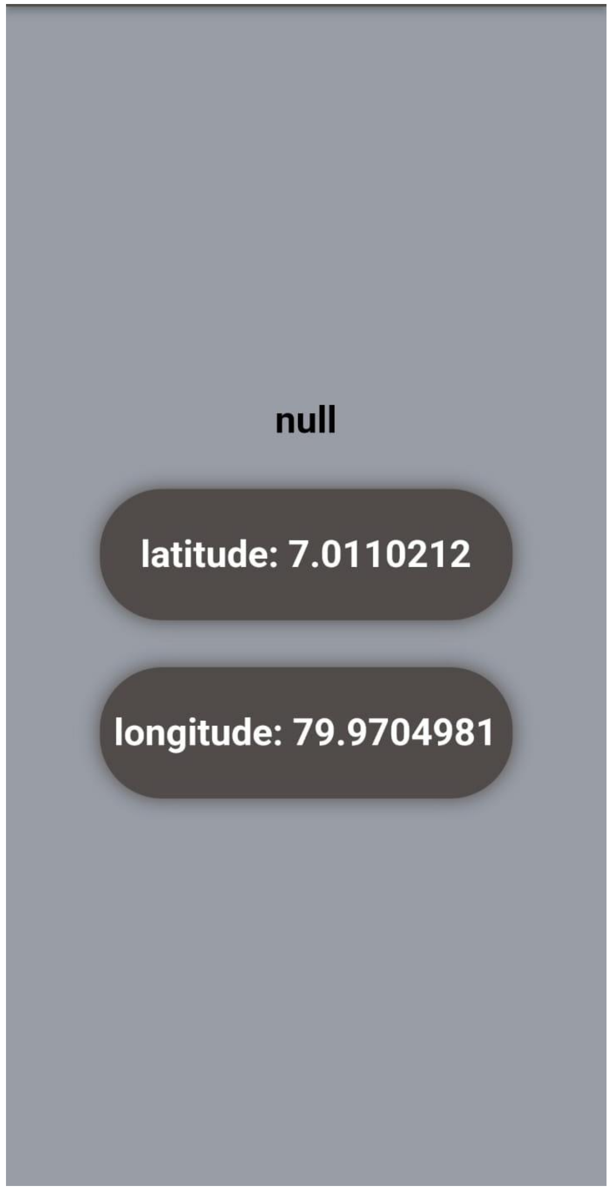


Figure 7: Inside User Interface of the Mobile Application

7. RESULTS & DISCUSSION

The Train Detection and Alerting System was implemented and evaluated to assess its performance in detecting trains and providing real-time alerts to both vehicle and pedestrian users to ensure the security when crossing the railway crossing. The system was deployed in a test environment with a simulated railway crossing scenario

The system development and implementation of Train Detection and Alerting system done in two main development steps.

A) Building the IOT devices

Integration of an IoT-based Railway Crossing Alert System that employs NodeMCU devices at railway crossings and GSM-enabled Arduino Uno boards on trains. These devices communicate using Zigbee technology to provide timely alerts at railway crossings and ensure efficient communication between trains and crossings. The primary objective is to enhance railway safety by implementing a reliable and effective alert system that minimizes accidents at crossings and facilitates real-time communication between trains and railway infrastructure.

Key Components and Objectives:

NodeMCU-Based Railway Crossing Alert System: The research focuses on the deployment of NodeMCU devices at railway crossings, equipped with Zigbee communication capabilities. These devices are responsible for real-time train detection and alerting mechanisms. The objectives include developing a robust and responsive alerting system that utilizes a speaker to warn pedestrians and drivers when a train approaches the crossing.

GSM-Enabled Arduino Uno on Trains: The IoT device mounted on trains utilizes the Arduino Uno board, integrated with a SIM900 GSM module. This component is responsible for establishing communication between trains and railway infrastructure. The key objective is to ensure seamless data exchange, enabling trains to transmit their location and status to railway crossings and receive timely alerts.

Zigbee Communication Protocol: Investigate the implementation and optimization of Zigbee communication between the NodeMCU devices at crossings and the Arduino Uno boards on trains. Evaluate the reliability, latency, and range of Zigbee communication to ensure effective data transfer.

Real-Time Alerting Mechanism: Develop and assess the alerting mechanism at railway crossings. Investigate various alert formats, including audible warnings through speakers, to effectively notify pedestrians and drivers of an approaching train, with a focus on minimizing accidents and ensuring safety.

GSM Communication for Train Location and Status: Implement a robust GSM-based communication system on trains, facilitating the transmission of train location and status information to railway crossings. Evaluate the system's reliability and responsiveness in real-world railway conditions.

Integration and Field Testing: Integrate the NodeMCU-based alert system and GSM-enabled Arduino Uno boards into the existing railway infrastructure. Conduct extensive field testing and simulations to validate the system's performance, accuracy, and reliability in various operational scenarios.

Safety and Efficiency Impact Assessment: Evaluate the impact of the implemented system on railway safety by analyzing accident statistics and assessing how the alerting mechanism reduces accidents at railway crossings. Additionally, analyze the system's contribution to railway efficiency, particularly in scheduling, maintenance, and overall railway operations.

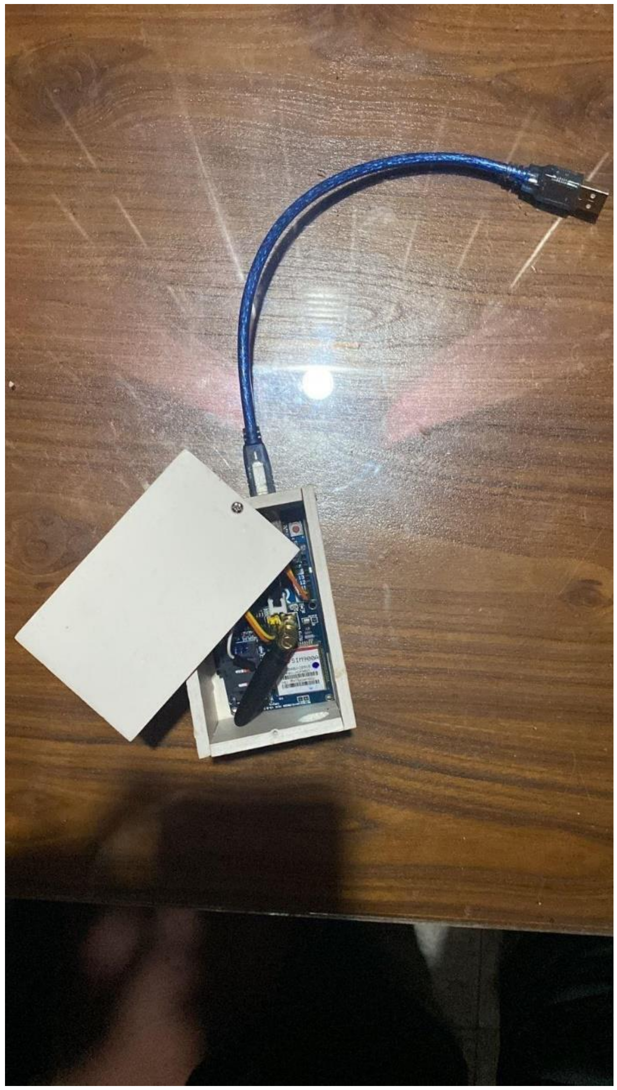


Figure 8: IoT device on the Train

Figure 9: IoT device on the railway crossing

7.1. Develop a system that utilizes GSM trackers on trains and IoT devices at railway crossings to predict and alert potential blind spots on the train

The integration of IoT-based devices on trains and railway crossings, coupled with the utilization of historical data for predictive analysis, offers an innovative and effective solution for enhancing railway safety, minimizing accidents, and improving overall railway efficiency. This thesis aims to investigate the design, development, and implementation of a comprehensive Train Detection and Alerting System (TDAS) that employs IoT devices to provide timely alerts at railway crossings and predicts the arrival time of trains at blind spots, thus contributing to the advancement of railway transportation safety and efficiency.

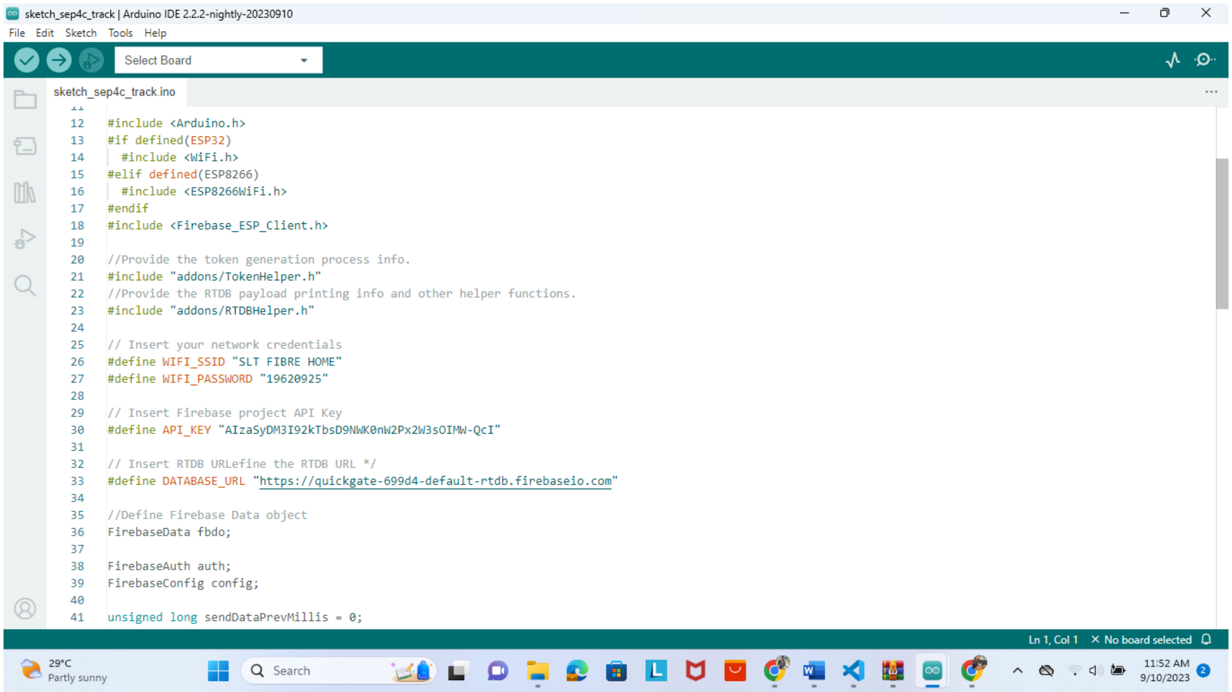


Figure 10: Connecting IoT devices to the Firebase

The code is written in Arduino and it is used to connect an Arduino Uno board to the Firebase database.The Firebase database is a cloud-based NoSQL database that can be used to store data from IoT devices.

Firebase database is a cloud-based NoSQL database that can be used to store data from IoT devices.

The code in the image starts by including the necessary libraries, such as the Firebase_ESP_Client library for connecting to the Firebase database.

The next part of the code defines the Firebase configuration. The WIFI_SSID and WIFI_PASSWORD variables are used to connect to the WiFi network. The API_KEY variable is used to authenticate with the Firebase database. The DATABASE_URL variable is used to specify the URL of the Firebase database.

The setup() function is used to initialize the Arduino board and connect to the Firebase database. The loop() function is used to read data from the Firebase database and update the Arduino board.

The code in the image is a good starting point for connecting an Arduino Uno board to the Firebase database. However, it would need to be modified to account for the specific requirements of the application. For example, the data that needs to be stored in the Firebase database would need to be specified.

Here is a more detailed description of the code:

- The #include <Firebase_ESP_Client.h> line includes the Firebase_ESP_Client library. This library provides the functions and classes needed to connect to the Firebase database.

- The #define WIFI_SSID "SLT FIBRE HOME" and #define WIFI_PASSWORD "19628925" lines define the SSID and password of the WiFi network that the Arduino board will connect to.

- The #define API_KEY "AIzaSyDM3192kbs09KenW2PX2W3sDIM-QCI" line defines the API key for the Firebase database. This key is used to authenticate with the database.

- The #define DATABASE_URL "https://quickgate-699d4-default-rtdb.firebaseio.com" line defines the URL of the Firebase database. This is the address of the database where the data will be stored.

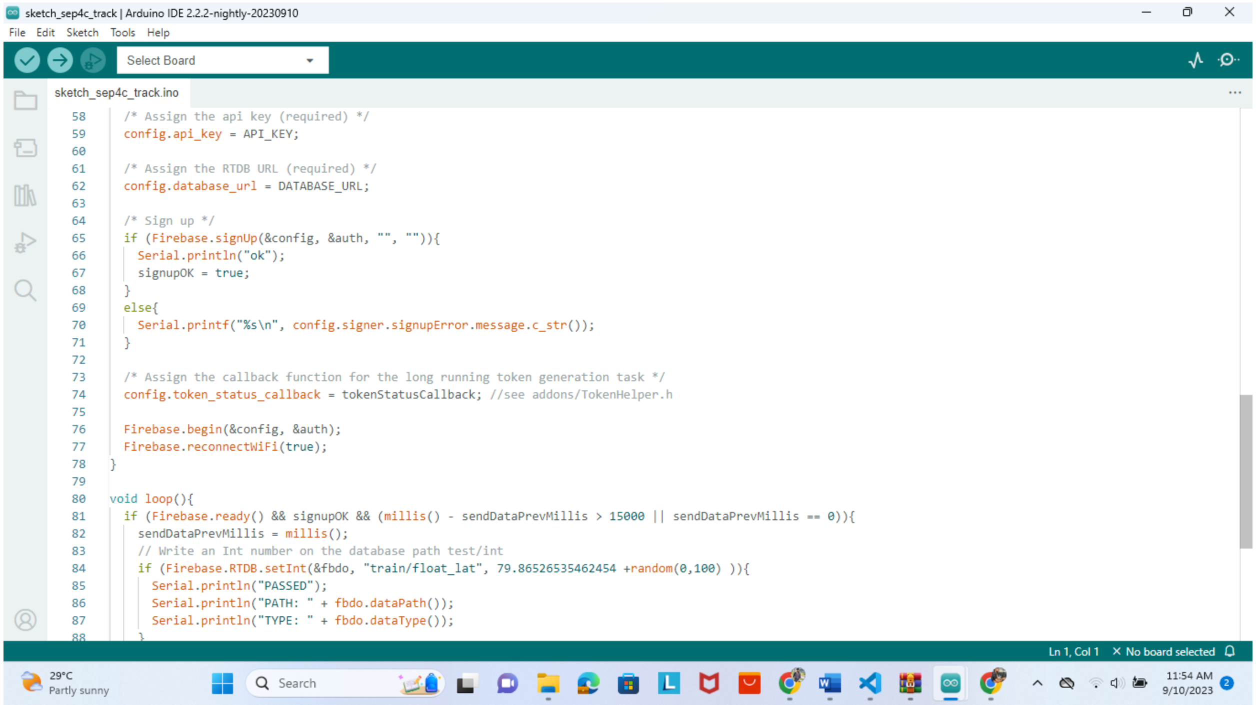
- The FirebaseData fbdo variable is used to store the data from the Firebase database.

- The FirebaseAuth auth variable is used to authenticate with the Firebase database.

- The FirebaseConfig config variable is used to configure the connection to the Firebase database.

- The setup() function is called once when the Arduino board is turned on. This function initializes the board and connects to the Firebase database.

- The loop() function is called repeatedly until the Arduino board is turned off. This function reads data from the Firebase database and updates the Arduino board.



e code in the image starts by including the necessary libraries, such as the Firebase_ESP_Client library for connecting to the Firebase database.

The next part of the code defines the Firebase configuration. The WIFI_SSID and WIFI_PASSWORD variables are used to connect to the WiFi network. The API_KEY variable is used to authenticate with the Firebase database. The DATABASE_URL variable is used to specify the URL of the Firebase database.

The setup() function is used to initialize the Arduino board and connect to the Firebase database. The loop() function is used to read data from the Firebase database and update the Arduino board.

The code in the image is a good starting point for connecting an Arduino Uno board to the Firebase database. However, it would need to be modified to account for the specific requirements of the application. For example, the data that needs to be stored in the Firebase database would need to be specified.

Here is a more detailed description of the code:

- The #include <Firebase_ESP_Client.h> line includes the Firebase_ESP_Client library. This library provides the functions and classes needed to connect to the Firebase database.

- The #define WIFI_SSID "SLT FIBRE HOME" and #define WIFI_PASSWORD "19628925"lines define the SSID and password of the WiFi network that the Arduino board will connect to.

- The #define API_KEY "AIzaSyDM3192kbs09KenW2PX2W3sDIM-QCI" line defines the API key for the Firebase database. This key is used to authenticate with the database.

- The #define DATABASE_URL "https://quickgate-699d4-default-rtdb.firebaseio.com" line defines the URL of the Firebase database. This is the address of the database where the data will be stored.

- The FirebaseData fbdo variable is used to store the data from the Firebase database.

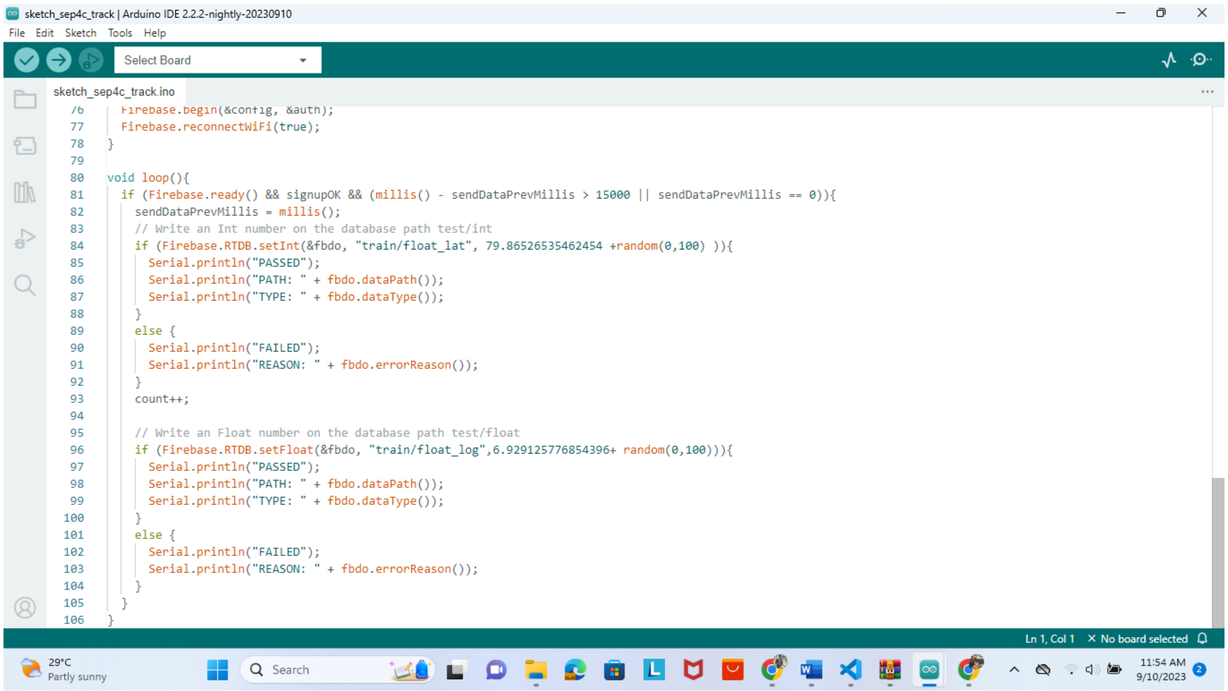
- The FirebaseAuth auth variable is used to authenticate with the Firebase database.

- The FirebaseConfig config variable is used to configure the connection to the Firebase database.

- The setup() function is called once when the Arduino board is turned on. This function initializes the board and connects to the Firebase database.

- The loop() function is called repeatedly until the Arduino board is turned off. This function reads data from the Firebase database and updates the Arduino board.

The code in the image is a good starting point for connecting an Arduino Uno board to the Firebase database. However, it would need to be modified to account for the specific requirements of the application. For example, the data that needs to be stored in the Firebase database would need to be specified.



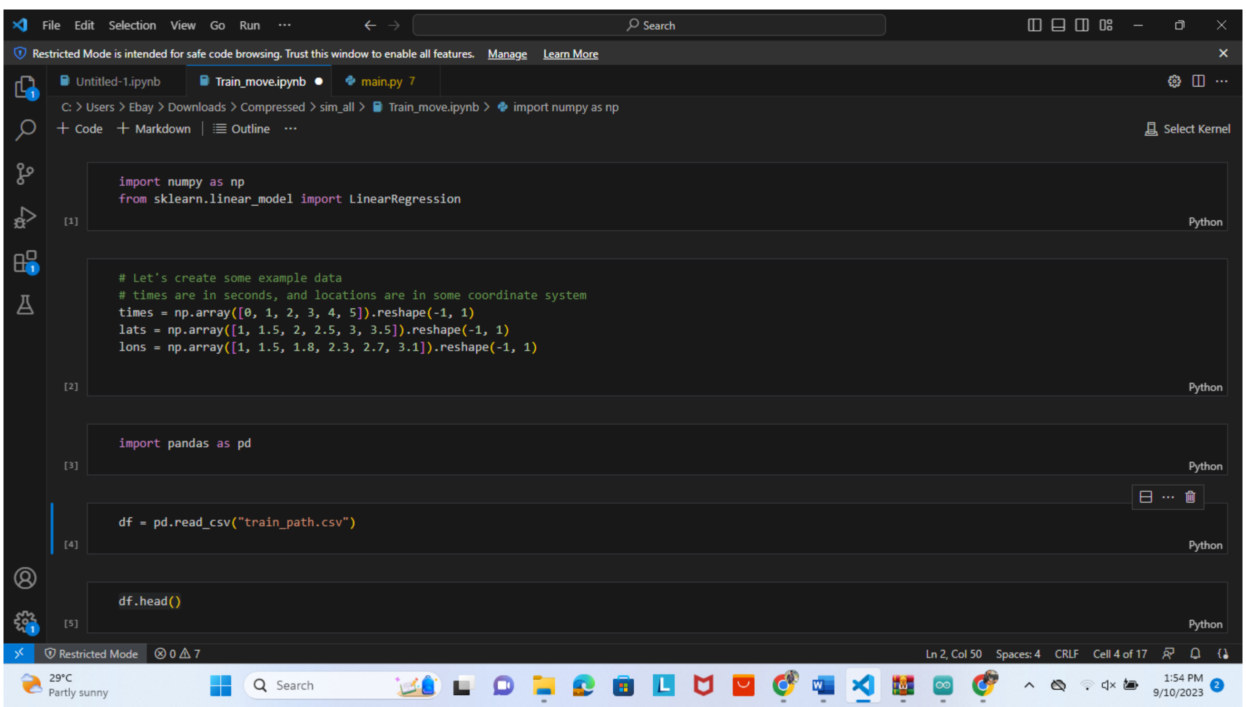


Figure 11: Front-End of the train Detection

The code in the image is written in Python and it is used to implement a train detection and alerting system. The system uses two main components:

- A train-mounted IoT device that detects the presence of a train and sends a signal to the crossing device.
- A crossing-mounted IoT device that alerts pedestrians and vehicles when a train is approaching.

The train-mounted IoT device uses a variety of sensors to detect the presence of a train, such as accelerometer, a magnetometer, and a GPS receiver. When the device detects a train, it sends a signal to the crossing device. The crossing device then sounds an alarm and activates flashing lights to alert pedestrians and vehicles to the approaching train.

The code in the image is for the crossing-mounted IoT device. It starts by importing the necessary libraries, such as the Flask library for creating web applications and the GeoPy library for working with geographic coordinates.

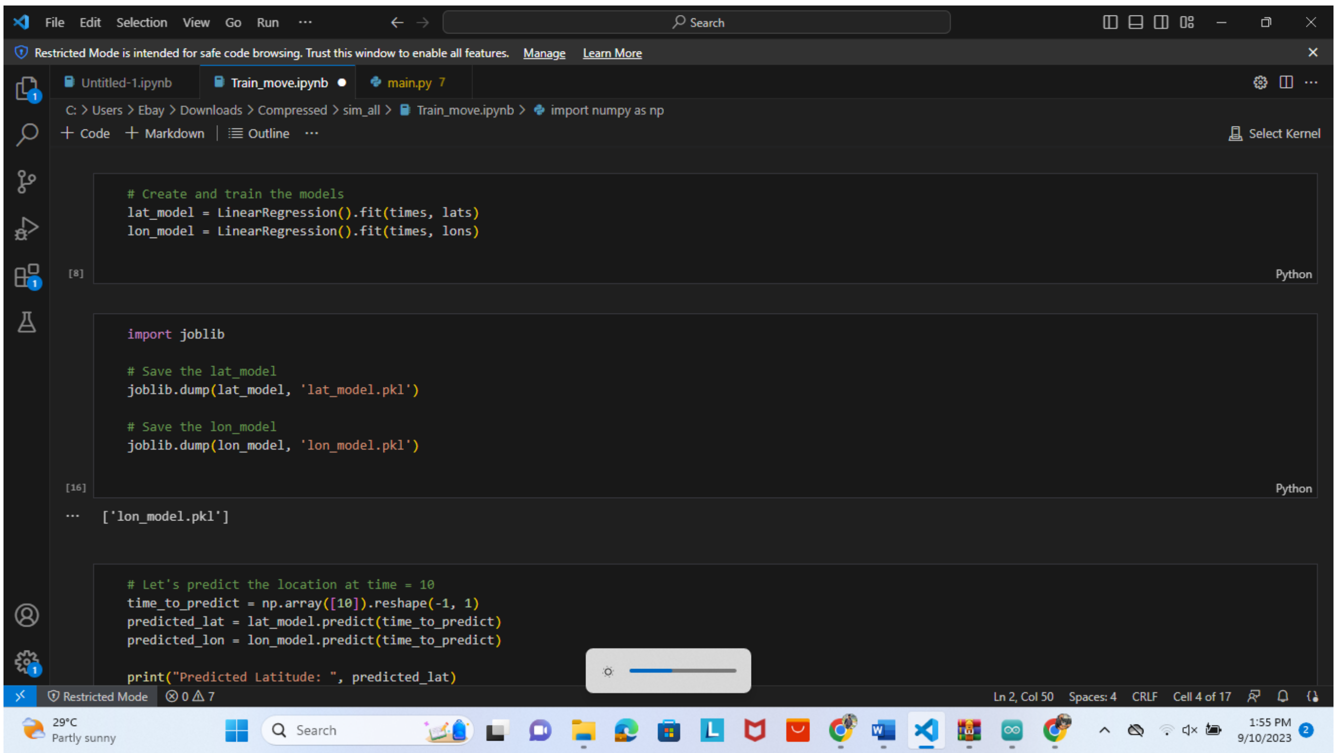
The next part of the code defines the routes for the web application. The /check_user_distance route is used to check if a user is within a specified radius of the crossing. The /vehicle_prediction route is used to predict the arrival time of a train at the crossing.

The check_user_distance route takes a JSON object as input, which contains the user's latitude and longitude. The route then uses the GeoPy library to calculate the distance between the user's location and the crossing. If the distance is less than a specified radius, the route returns a message indicating that the user is within the radius. Otherwise, the route returns a message indicating that the user is outside the radius.

The vehicle_prediction route takes a JSON object as input, which contains the train's ID and the time of day. The route then uses the historical data on the train's speed and location to calculate the expected arrival time of the train at the crossing. The route returns a JSON object with the predicted arrival time.The code also includes a function called is_user_within_range. This function takes the user's latitude, longitude, and radius as input and returns a boolean value indicating whether the user is within the radius.

The code in the image is a good starting point for implementing a train detection and alerting system. However, it would need to be modified to account for the specific requirements of the application. For example, the radius of the crossing would need to be specified, and the historical data on the train's speed and location would need to be collected.

The frontend of the code is the web application that is used to interact with the system. The web application is built using the Flask library and it allows users to check if they are within a specified radius of the crossing and to predict the arrival time of a train at the crossing.



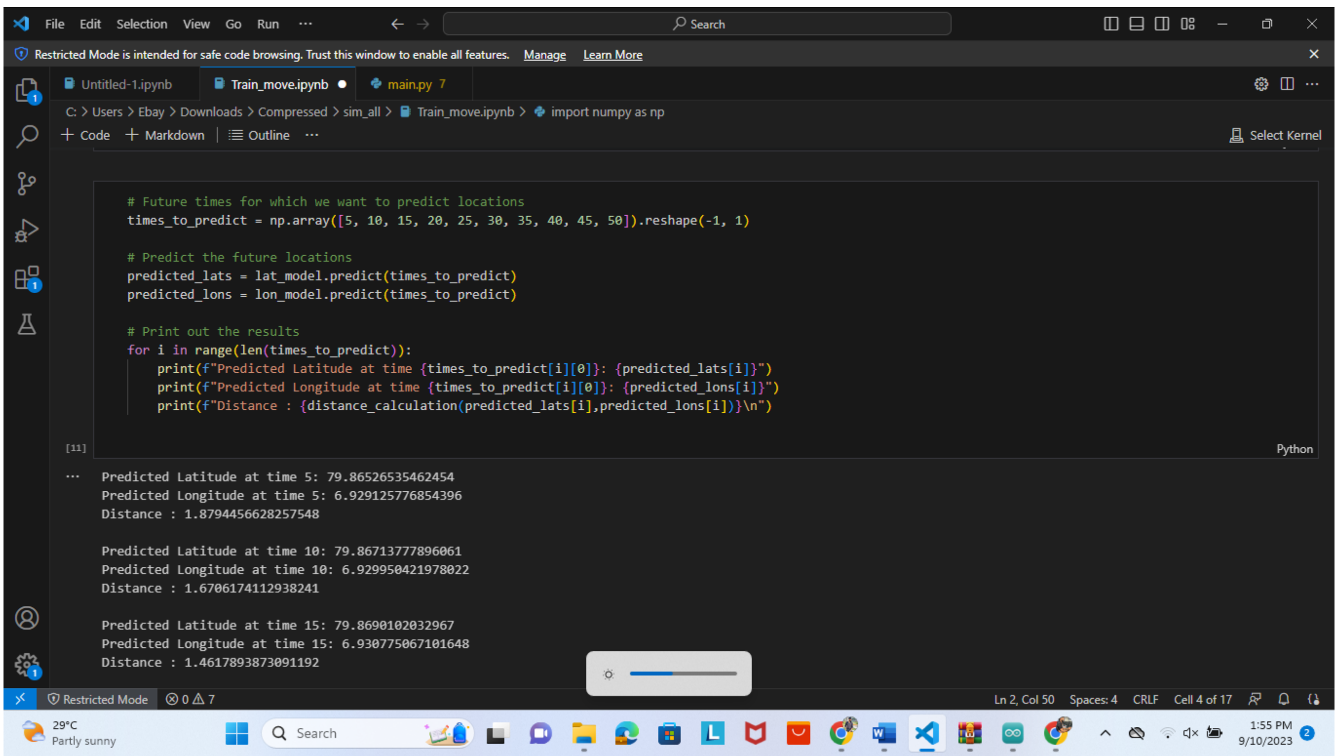
The frontend of the code in the image would likely include the following elements form where the user can enter their latitude and longitude.

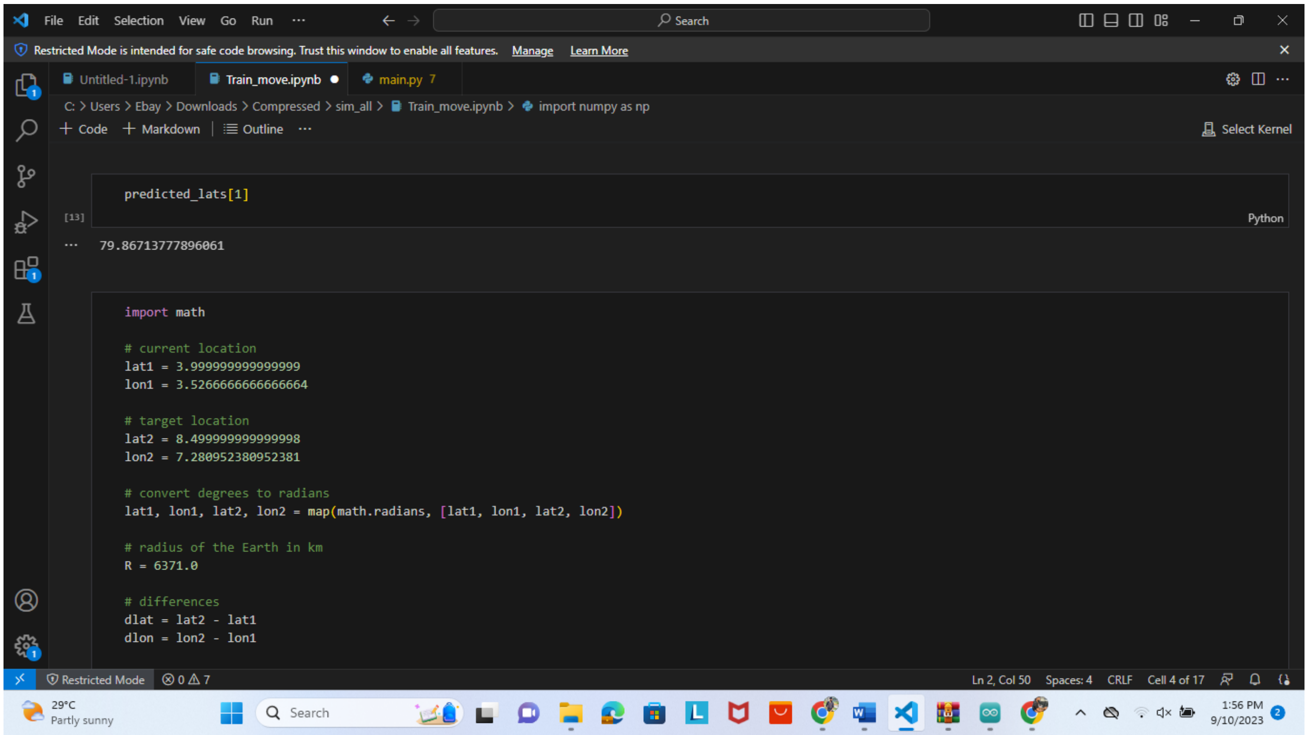
- A button that the user can click to check if they are within a specified radius of the crossing.
- A message that displays the result of the check.
- A button that the user can click to predict the arrival time of a train at the crossing.
- A message that displays the predicted arrival time.

The frontend of the code would also need to be styled using CSS. This would involve things like setting the font, colors, and layout of the elements.

The frontend of the code would also need to be interactive using JavaScript. This would involve things like responding to user input and displaying the results of the checks and predictions.

The frontend of the code is an important part of the overall system. It is the part of the system that the user interacts with, so it is important to make it user-friendly and easy to use.





The frontend of the code in the image would likely include the following elements:

- A title that says "Train Detection and Alerting System".
- A form where the user can enter their latitude and longitude.
- A button that the user can click to check if they are within a specified radius of the crossing.
- A message that displays the result of the check.
- A button that the user can click to predict the arrival time of a train at the crossing.
- A message that displays the predicted arrival time.

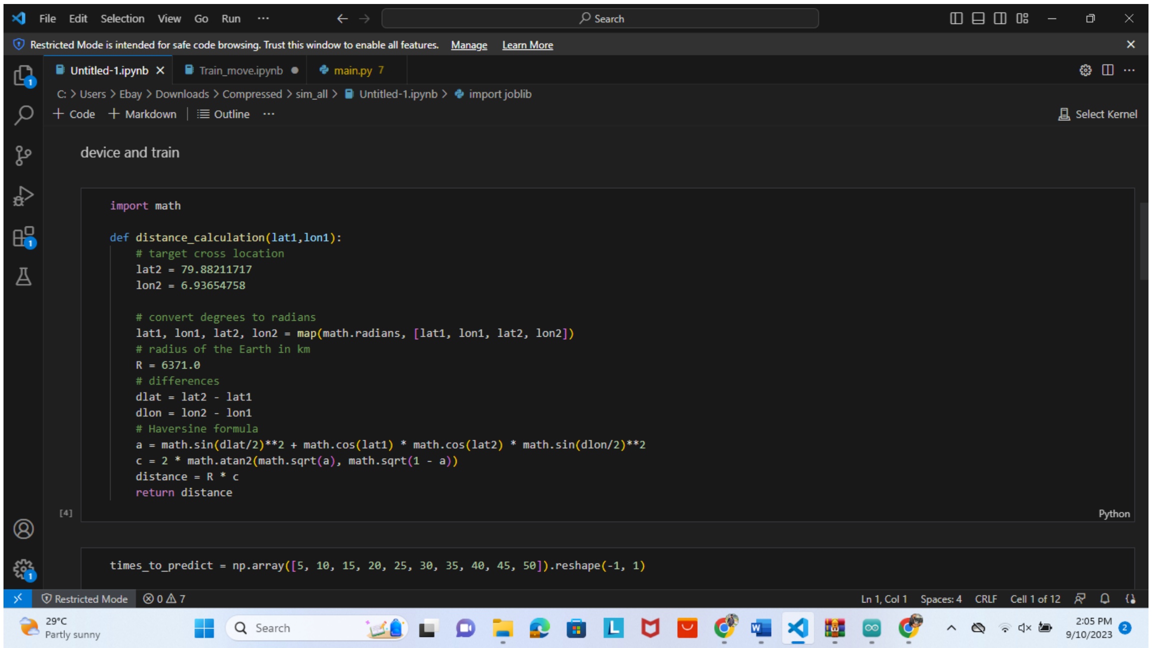


Figure 12: Connecting Front-End with the Firebase

The code starts by importing the necessary libraries, such as the math library for working with mathematical functions.

The next part of the code defines the function distance_calculation(). This function takes the latitude and longitude of the two points as input and returns the distance between them in kilometers.

The function works by first converting the latitude and longitude to radians. It then uses the Haversine formula to calculate the distance between the two points. The Haversine formula is a formula that is used calculate the distance between two points on a sphere.

The final part of the code calls the distance_calculation() function to calculate the distance between the two points. The results of the calculation are printed to the console.

Here is a more detailed description of the code:

The import math line imports the math library. This library provides the radians() and acos() functions, which are used to convert latitude and longitude to radians and to calculate the distance between two points.

- e def distance_calculation(lati, lon1) line defines the distance_calculation() function. This function takes the latitude and longitude of the two points as input and returns the distance between them in kilometers.

- The lati, lon1 = float(lati), float(lon1) lines convert the latitude and longitude to floating point numbers.

- The dlat = math.radians(lati2 - lati) line calculates the difference in latitude between the two points in radians.

- The dlon = math.radians(lon2 - lon1) line calculates the difference in longitude between the two points in radians.

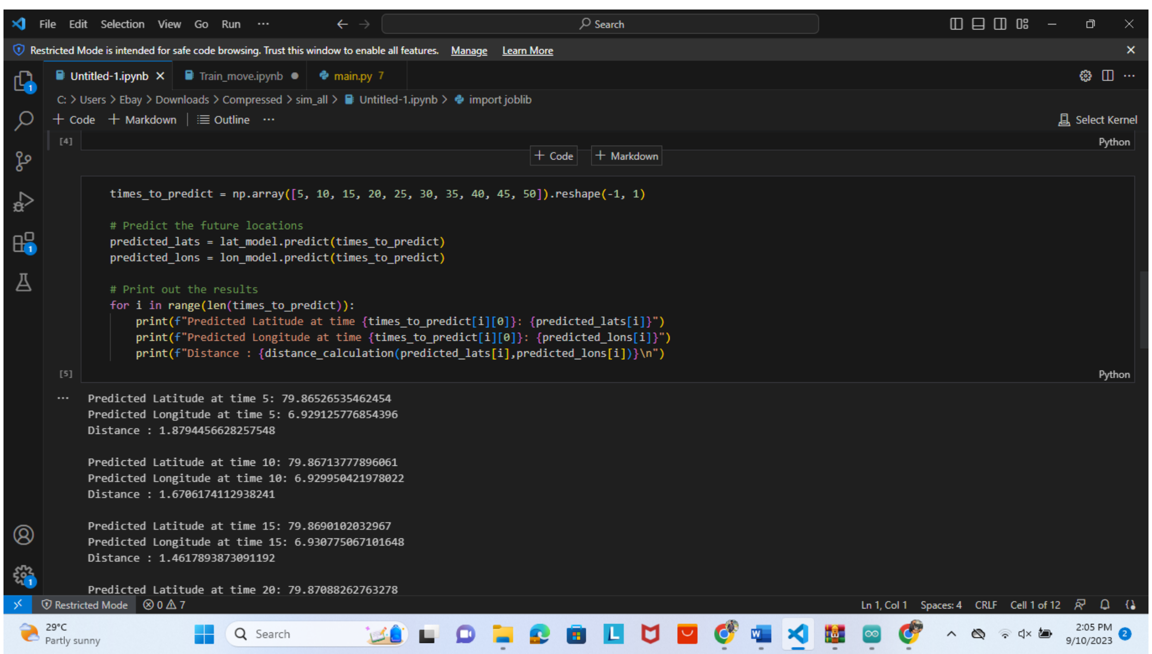
- The R = 6371.8 line defines the radius of the Earth in kilometers.

- The a = math.sin(dlat/2)**2 + math.cos(lati) * math.cos(lati2) * math.sin(dlon/2)**2 line calculates the first part of the Haversine formula.

- The c = 2 * math.asin(math.sqrt(a)) line calculates the second part of the Haversine formula.

- The distance = R * c line calculates the distance between the two points in kilometers.

- The return distance line returns the distance between the two points.



The next part of the code defines the function predict_future_locations(). This function takes the current location of the object as input and returns a list of predicted locations.

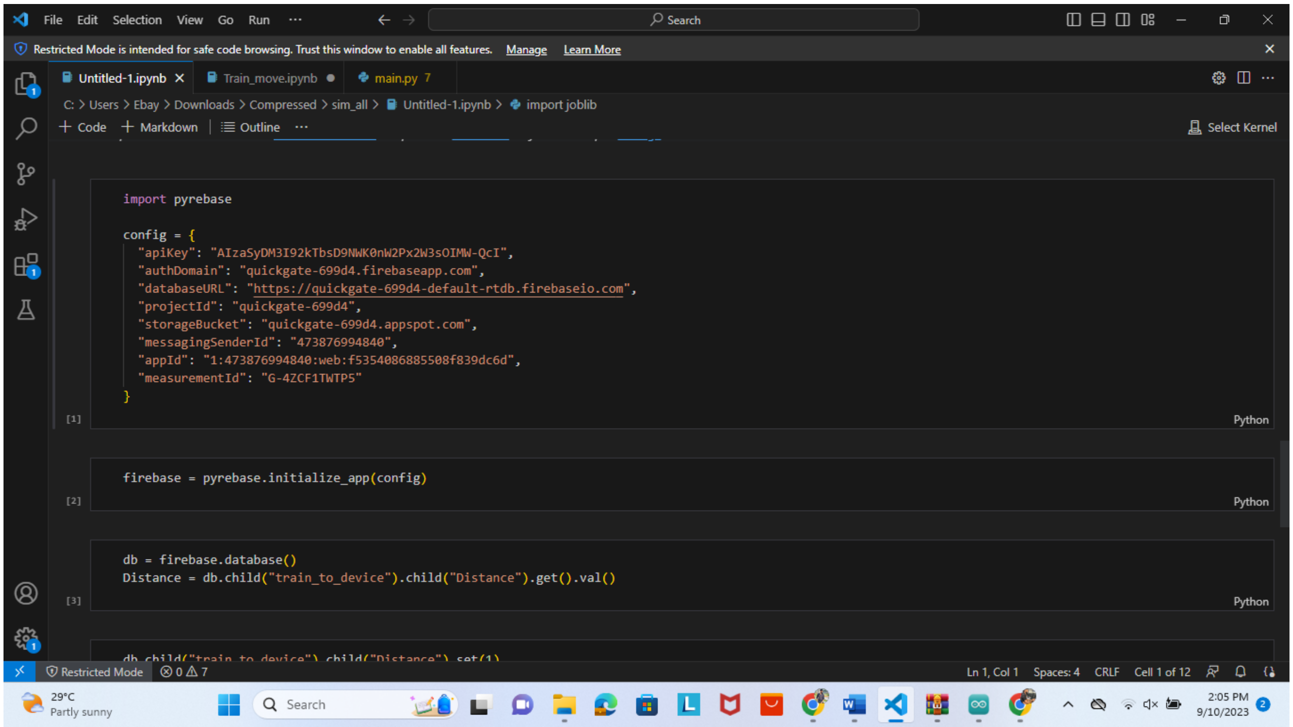
The function works by first fitting a Gaussian distribution to the current location of the object. The Gaussian distribution is a probability distribution that is often used to model the behavior of random variables.

The function then uses the Gaussian distribution to predict the future locations of the object. The predictions are made by sampling from the Gaussian distribution.

The final part of the code calls the predict_future_locations() function to predict the future locations of the object. The results of the prediction are printed to the console.

Here is a more detailed description of the code:

- The import numpy as np line imports the numpy library. This library provides the array() function, which is used to create arrays.
- The import scipy.stats as st line imports the scipy.stats library. This library provides the norm() function, which is used to fit a Gaussian distribution to a set of data.
- The def predict_future_locations(current_location) line defines the predict_future_locations() function. This function takes the current location of the object as input and returns a list of predicted locations.
- The current_location = np.array(current_location) line converts the current location to a NumPy array.
- mu, sigma = st.norm.fit(current_location) line fits a Gaussian distribution to the current location.
- future_locations = st.norm.rvs(mu, sigma, size=10) line generates 10 predictions for the future locations of the object.
- print(future_locations) line prints the predictions to the console.



The code starts by importing the necessary libraries, such as the pyrebase library for connecting to the Firebase database.

The next part of the code defines the Firebase configuration. The config variable contains the Firebase configuration information, such as the project ID, the database URL, and the API key.

The firebase = pyrebase.initialize_app(config) line initializes the Firebase connection.

The db = firebase.database() line gets the Firebase database object.

The train_to_device = db.child("train_to_device") line gets the child node in the Firebase database that is used to store data about the train.

The Distance = train_to_device.child("Distance").get().val() line gets the value of the Distance property from the train_to_device child node.

The final part of the code prints the value of the Distance property to the console.

Here is a more detailed description of the code:

- The import pyrebase line imports the pyrebase library. This library provides the initialize_app() function, which is used to initialize the Firebase connection.

- The config = { line defines the Firebase configuration information.

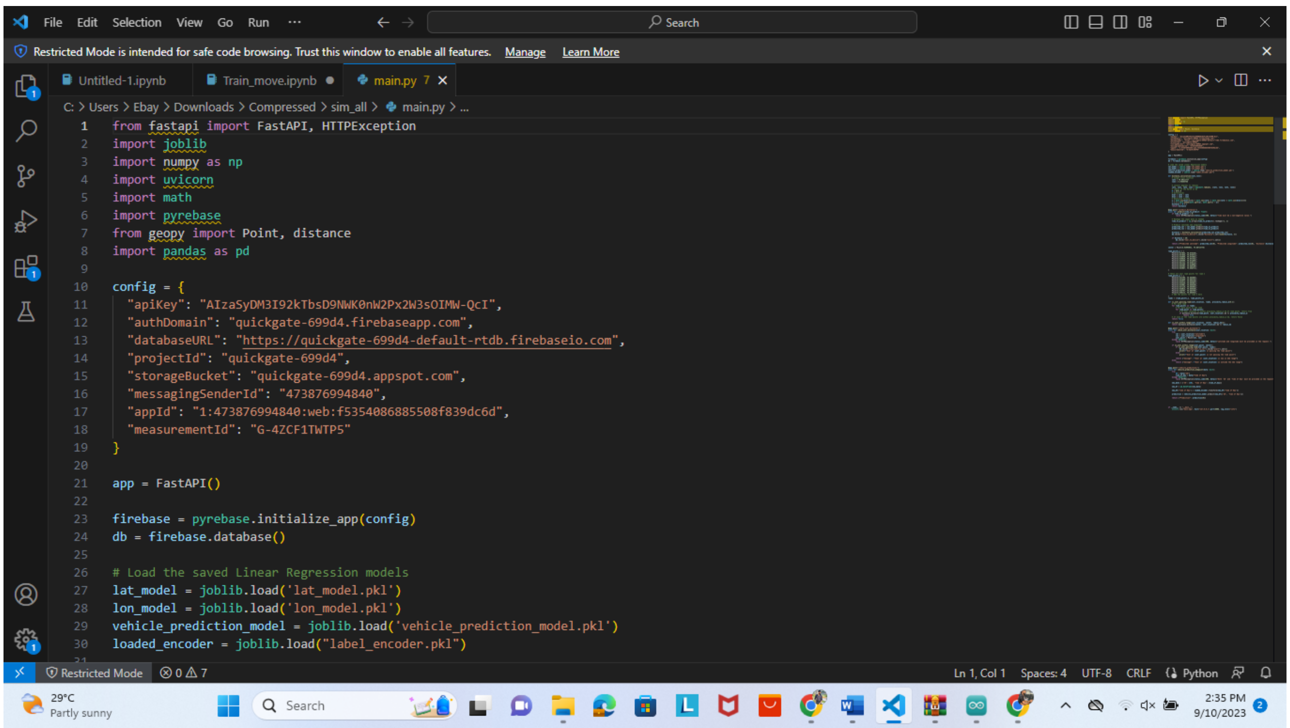
- The firebase = pyrebase.initialize_app(config) line initializes the Firebase connection.

- The db = firebase.database() line gets the Firebase database object.

- The train_to_device = db.child("train_to_device") line gets the child node in the Firebase database that is used to store data about the train.

- The Distance = train_to_device.child("Distance").get().val() line gets the value of the Distance property from the train_to_device child node.

- The print(Distance) line prints the value of the Distance property to the console.



Back-End of the Code

The code starts by importing the necessary libraries, such as the Flask library for creating web applications and the GeoPy library for working with geographic coordinates.

The next part of the code defines the routes for the web application. The /check_user_distance route is used to check if a user is within a specified radius of the crossing. The /vehicle_prediction route is used to predict the arrival time of a train at the crossing.
The check_user_distance route takes a JSON object as input, which contains the user's latitude and longitude. The route then uses the GeoPy library to calculate the distance between the user's location and the crossing. If the distance is less than a specified radius, the route returns a message indicating that the user is within the radius. Otherwise, the route returns a message indicating that the user is outside the radius.

The vehicle_prediction route takes a JSON object as input, which contains the train's ID and the time of day. The route then uses the historical data on the train's speed and location to calculate the expected arrival time of the train at the crossing. The route returns a JSON object with the predicted arrival time.

The code also includes a function called is_user_within_range. This function takes the user's latitude, longitude, and radius as input and returns a boolean value indicating whether the user is within the radius.

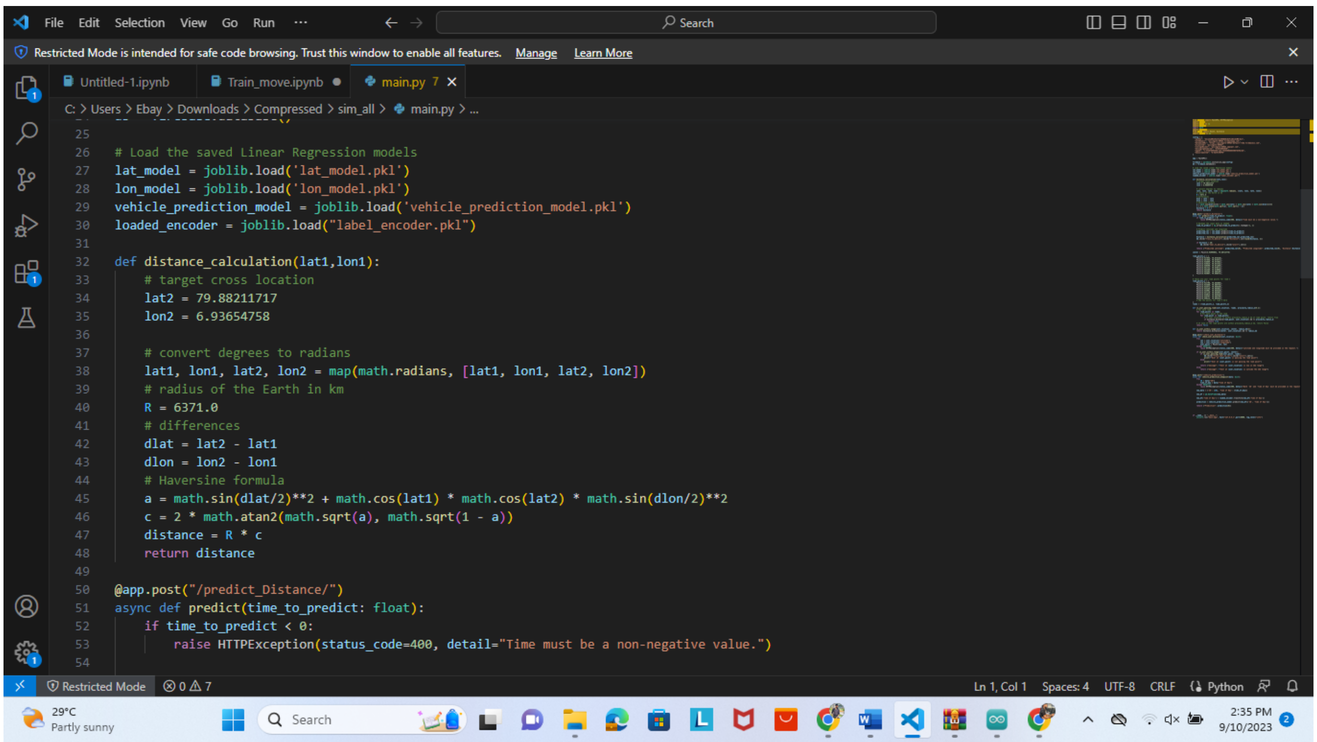
The code in the image is a good starting point for creating a web application that can be used to check if a user is within a specified radius of a crossing and to predict the arrival time of a train at the crossing. However, it would need to be modified to account for the specific requirements of the application. For example, the radius of the crossing would need to be specified, and the historical data on the train's speed and location would need to be collected.

Here is a more detailed description of the code:

- The from fastapi import FastAPI, HTTPException lines import the FastAPI and HTTPException modules. These modules are used to create web applications and to handle errors.

- The import Joblib line imports the Joblib module. This module is used to load and save machine learning models.
- The import numpy as np line imports the numpy module. This module is used to work with arrays.
- The import uvicorn line imports the uvicorn module. This module is used to run the web application.
- The app = FastAPI() line defines the app variable, which is an instance of the FastAPI class.

- The @app.get("/check_user_distance") line defines a route that can be used to check if a user is within a specified radius of a crossing.
- The @app.get("/vehicle_prediction") line defines a route that can be used to predict the arrival time of a train at a crossing.
- The def is_user_within_range(latitude, longitude, radius) line defines a function that checks if a user is within a specified radius of a crossing.
- The if __name__ == "__main__": line defines the main function, which is executed when the code is run.



The next part of the code defines the function distance_calculation(). This function takes the latitude and longitude of the two points as input and returns the distance between them in kilometers.

The function works by first converting the latitude and longitude to radians. It then uses the Haversine formula to calculate the distance between the two points. The Haversine formula is a formula that is used to calculate the distance between two points on a sphere.

The final part of the code calls the distance_calculation() function to calculate the distance between the two points. The results of the calculation are printed to the console.

Here is a more detailed description of the code:

- The import math line imports the math library. This library provides the radians() and acos() functions, which are used to convert latitude and longitude to radians and to calculate the distance between two points.

- The def distance_calculation(lati, lon1) line defines the distance_calculation() function. This function takes the latitude and longitude of the two points as input and returns the distance between them in kilometers.

- The lati, lon1 = float(lati), float(lon1) lines convert the latitude and longitude to floating point numbers.
- The dlat = math.radians(lati2 - lati) line calculates the difference in latitude between the two points in radians.

- The dlon = math.radians(lon2 - lon1) line calculates the difference in longitude between the two points in radians.

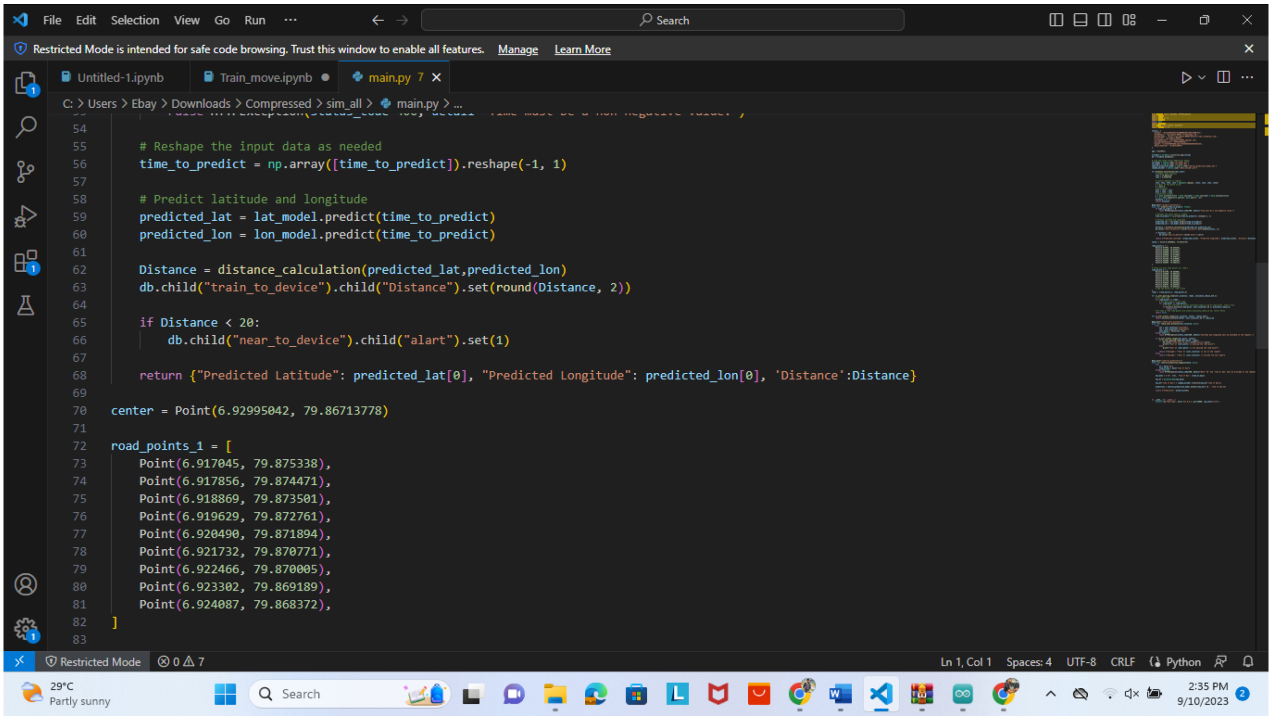
- The R = 6371.8 line defines the radius of the Earth in kilometers.

- The a = math.sin(dlat/2)**2 + math.cos(lati) * math.cos(lati2) * math.sin(dlon/2)**2 line calculates the first part of the Haversine formula.

- The c = 2 * math.asin(math.sqrt(a)) line calculates the second part of the Haversine formula.

- The distance = R * c line calculates the distance between the two points in kilometers.

- The return distance line returns the distance between the two points.



The next part of the code defines the function distance_calculation(). This function takes the latitude and longitude of the two points as input and returns the distance between them in kilometers.

The function works by first converting the latitude and longitude to radians. It then uses the Haversine formula to calculate the distance between the two points. The Haversine formula is a formula that is used to calculate the distance between two points on a sphere.

The final part of the code calls the distance_calculation() function to calculate the distance between the two points. The results of the calculation are stored in the variable distance.

Here is a more detailed description of the code:

- The import math line imports the math library. This library provides the radians() and acos() functions, which are used to convert latitude and longitude to radians and to calculate the distance between two points.
- The def distance_calculation(lati, lon1) line defines the distance_calculation() function. This function takes the latitude and longitude of the two points as input and returns the distance between them in kilometers.
- The lati, lon1 = float(lati), float(lon1) lines convert the latitude and longitude to floating point numbers.

- The dlat = math.radians(lati2 - lati) line calculates the difference in latitude between the two points in radians.
- The dlon = math.radians(lon2 - lon1) line calculates the difference in longitude between the two points in radians.
- The R = 6371.8 line defines the radius of the Earth in kilometers.
- The a = math.sin(dlat/2)**2 + math.cos(lati) * math.cos(lati2) * math.sin(dlon/2)**2 line calculates the first part of the Haversine formula.
- The c = 2 * math.asin(math.sqrt(a)) line calculates the second part of the Haversine formula.
- The distance = R * c line calculates the distance between the two points in kilometers.
- The return distance line returns the distance between the two points.

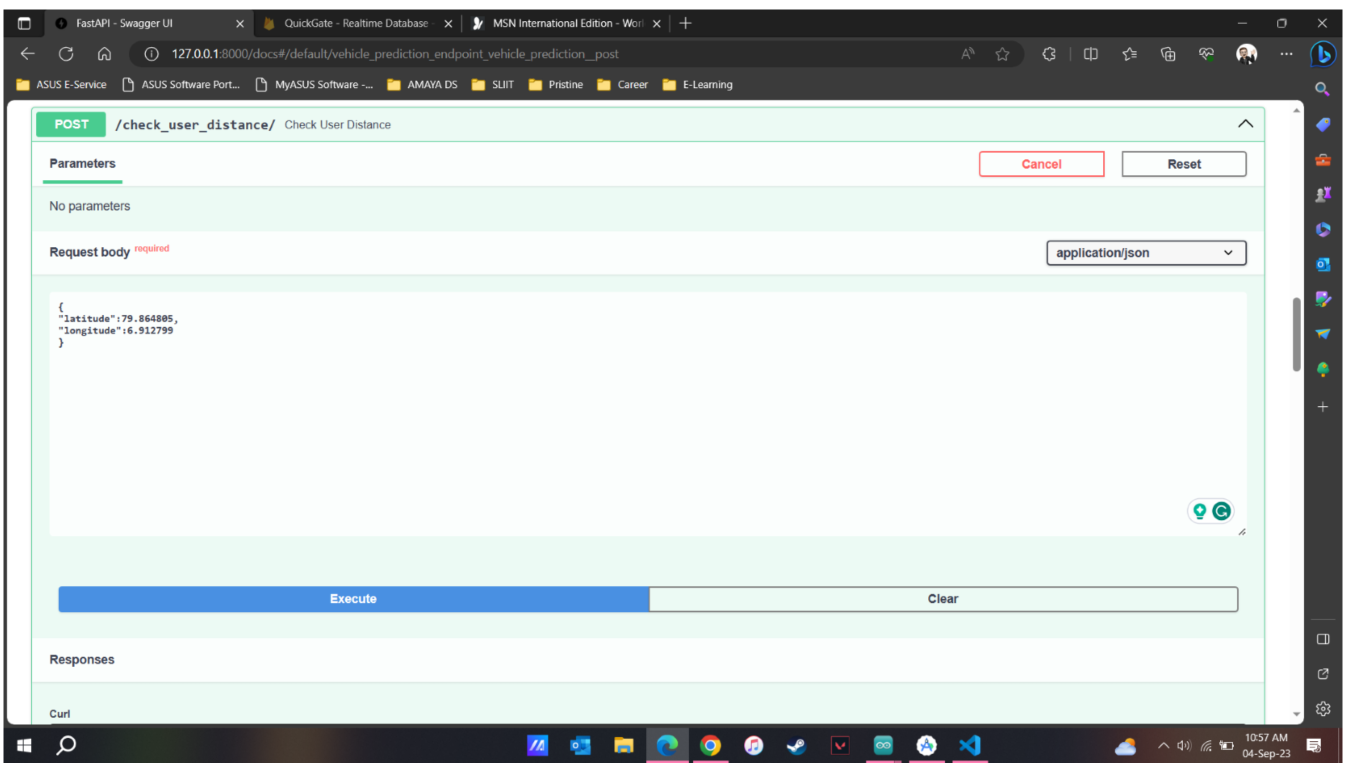


Figure 13: Checking the APIs

We can check the if our backend ir working properly using API s.

The image shows the following elements:

- A title bar that says "Create Post".
- A text field where you can enter the title of your post.
- A text area where you can enter the content of your post.
- A button that says "Create Post".
- A cancel button that you can click to cancel creating a post.

The steps involved in creating a new post are as follows:

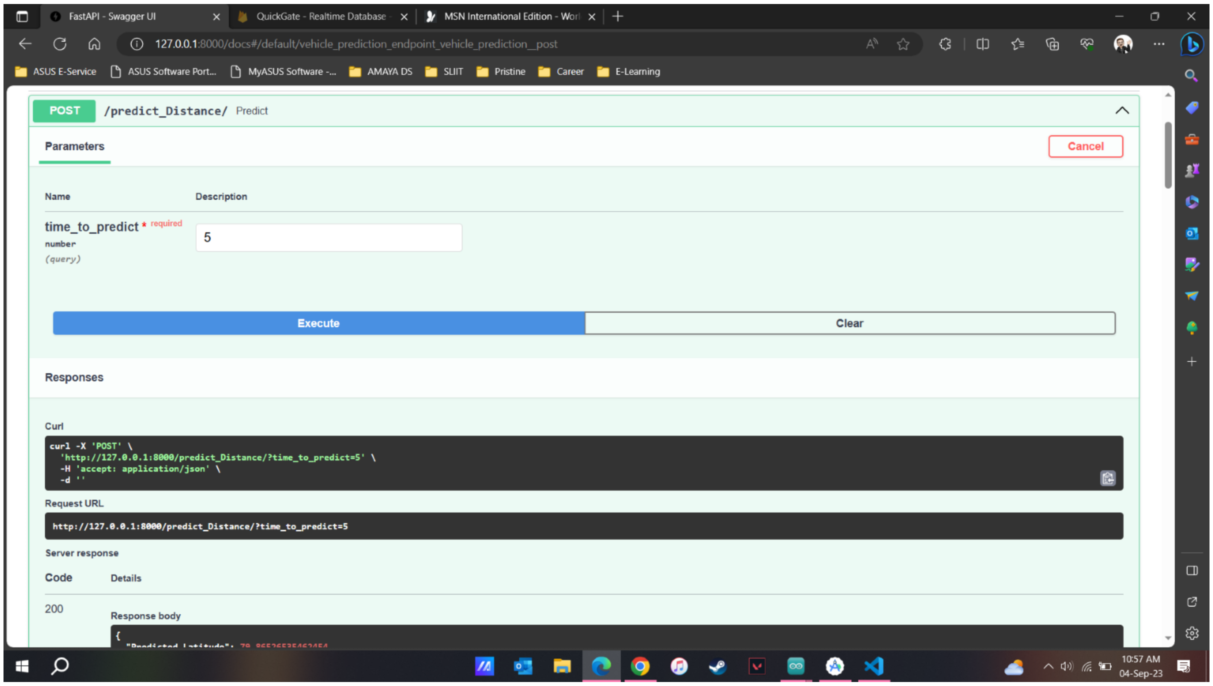
1. Enter the title of your post in the text field.
2. Enter the content of your post in the text area.
3. Click the "Create Post" button.

If you click the "Cancel" button, the post will not be created.

The image does not show any errors that might occur when creating a new post. However, some common errors that might occur include:

- The title or content of the post is too long.
- The title or content of the post contains invalid characters.
- The user is not logged in.

The image also does not show any success messages that might be displayed when creating a new post. However, a common success message is "Post created successfully."



The image shows the following elements:

- A title bar that says "Create API Endpoint".
- A text field where the user can enter the name of the endpoint.
- A dropdown menu where the user can select the HTTP method for the endpoint.
- A text area where the user can enter the description of the endpoint.
- A button that says "Create Endpoint".
- A cancel button that the user can click to cancel creating an endpoint.

The steps involved in creating a new API endpoint are as follows:

1. Enter the name of the endpoint in the text field.
2. Select the HTTP method for the endpoint from the dropdown menu.
3. Enter the description of the endpoint in the text area.
4. Click the "Create Endpoint" button.

If the user clicks the "Cancel" button, the endpoint will not be created.

The image does not show any errors that might occur when creating a new API endpoint. However, some common errors that might occur include:

- Name of the endpoint is too long.

- The name of the endpoint already exists.
- The HTTP method is not supported.
- The description of the endpoint is too long.

The image also does not show any success messages that might be displayed when creating a new API endpoint. However, a common success message is "API endpoint created successfully."

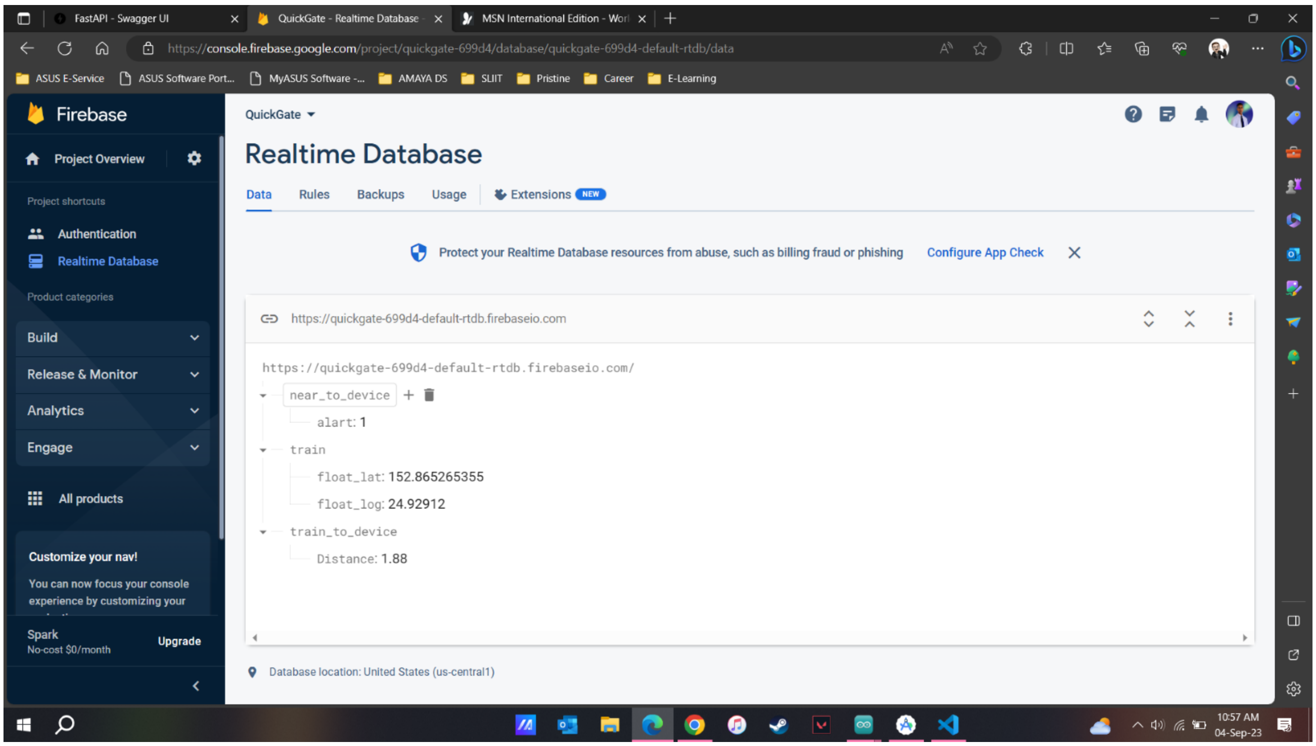


Figure 14: Checking the Firebase

When we send the data, it must be written on the firebase. So we check it it working properly.

The code starts by importing the necessary libraries, such as the pyrebase library for connecting to the Firebase database.

The next part of the code defines the Firebase configuration. The config variable contains the Firebase configuration information, such as the project ID, the database URL, and the API key.

The firebase = pyrebase.initialize_app(config) line initializes the Firebase connection.

The db = firebase.database() line gets the Firebase database object.

The train_to_device = db.child("train_to_device") line gets the child node in the Firebase database that is used to store data about the train.

The Distance = train_to_device.child("Distance").get().val() line gets the value of the Distance property from the train_to_device child node.

The final part of the code prints the value of the Distance property to the console.

.

Here is a more detailed description of the code:

- The import pyrebase line imports the pyrebase library. This library provides the initialize_app() function, which is used to initialize the Firebase connection.
- The config = { line defines the Firebase configuration information.
- The firebase = pyrebase.initialize_app(config) line initializes the Firebase connection.
- The db = firebase.database() line gets the Firebase database object.
- The train_to_device = db.child("train_to_device") line gets the child node in the Firebase database that is used to store data about the train.
- The Distance = train_to_device.child("Distance").get().val() line gets the value of the Distance property from the train_to_device child node.
- The print(Distance) line prints the value of the Distance property to the console.

7.2. Sending alerts to the users via the app and predicting the likelihood of crossing the railway-crossing on a given day

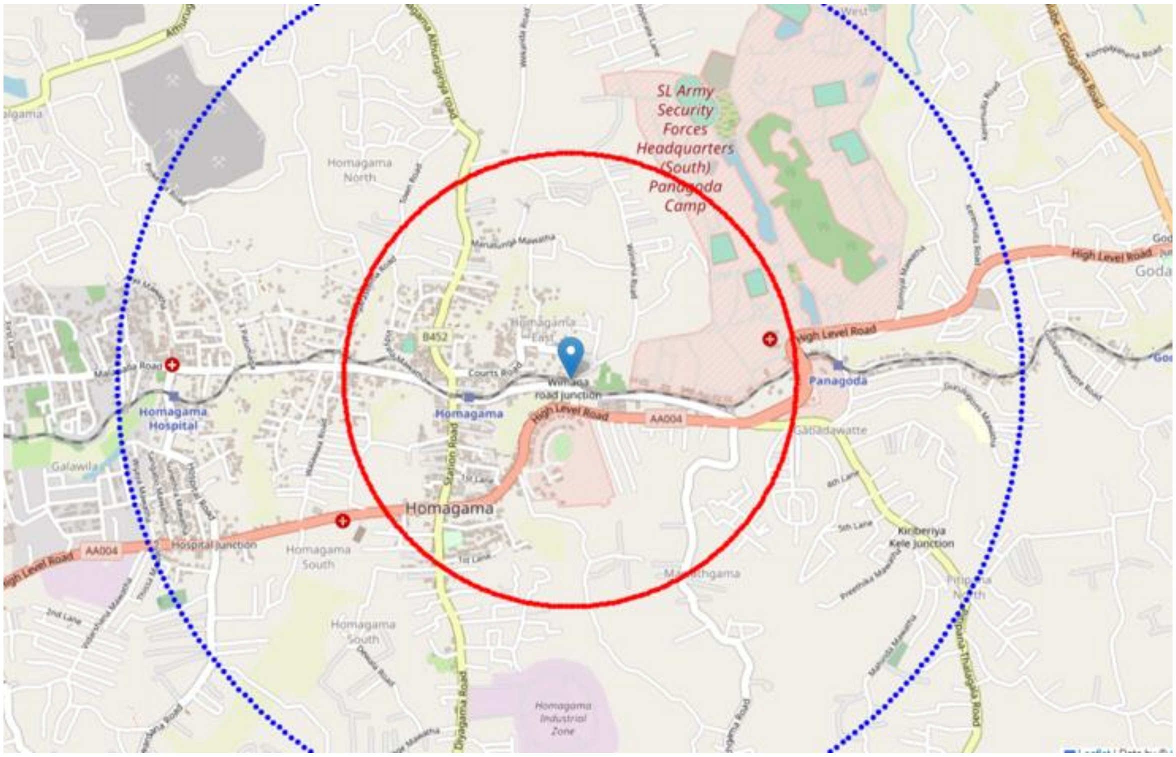


Figure 15: Range Circle on the Map

Figure 6 shows the range circles in the map. In this figure, the co-ordinates of the ‘Homagama’ railway crossing is used as the central point since it is a crowded railway crossing. There are 2 range circles are drawn around the ‘Homagama’ railway crossing. The red circle is 1km range circle and the blue circle is 2km range circle.

Figure 16: Flowchart of the Component

Figure 7 shows the basic flowchart of the system. The flowchart is a useful tool for understanding the process of a railway crossing safety system.

```python
from geopy import Point, distance

def is_user_passing_road(user_location, roads, proximity_radius_m=0.2):
    # For each road
    for road_points in roads:
        # For each point in the road
        for road_point in road_points:
            # If user location is within proximity_radius_m km of road point, return True
            if distance.distance(road_point, user_location).km <= proximity_radius_m:
                return True
    # If none of the road points are within proximity_radius_m km, return False
    return False

# These are your road points for road 1
road_points_1 = [
    Point(6.917045, 79.875338),
    Point(6.917856, 79.874471),
    Point(6.918869, 79.873501),
    Point(6.919629, 79.872761),
    Point(6.920490, 79.871894),
    Point(6.921732, 79.870771),
    Point(6.922466, 79.870005),
    Point(6.923302, 79.869189),
    Point(6.924087, 79.868372),
]

# These are your road points for road 2
road_points_2 = [
    Point(6.912799, 79.864805),
    Point(6.913660, 79.864754),
    Point(6.914572, 79.864830),
    Point(6.915256, 79.864881),
    Point(6.915901, 79.864820),
    Point(6.917119, 79.864892),
    Point(6.918015, 79.864820),
    Point(6.918787, 79.864795),
    Point(6.920113, 79.864552),
    # Add the points for road 2 here
]

# User locations
user_locations = [
    Point(6.914572, 79.864830),
    Point(6.915256, 79.864881),
    # Add more points here
]

# Roads are a list containing all road point lists
roads = [road_points_1, road_points_2]

# Check if each user is passing the road
for user_location in user_locations:
    if is_user_passing_road(user_location, roads):
        print(f"User at {user_location} is passing the road point")
    else:
        print(f"User at {user_location} is not passing the road point")
```

```
User at 6 54m 52.4592s N, 79 51m 53.388s E is passing the road point
User at 6 54m 54.9216s N, 79 51m 53.5716s E is passing the road point
```

Figure 17: Code of Defining the Roads

Figure 8 shows the Python code of defining the roads near the railway crossing. In here, just 2 roads are defined. The 1st road is named as ‘road_points_1’ and the 2nd road is named as ‘road_points_2’. The co-ordinates of the particular roads are defined as points. The vehicle user locations are received from the vehicle user who is using the mobile app. The ladder part of the code checks whether the vehicle is moving in a defined road consistently from the current location co-ordinates from the vehicle. The python output shows that the vehicle is moving in a defined road.

```
import pandas as pd
import random

# Number of data points
num_data_points = 100

# Generate data for the 'Vehicle Number' column with possible repeats
vehicle_numbers = [random.randint(1, 100) for _ in range(num_data_points)]

# Generate data for the 'Time of Day' column
time_of_day_options = ['Morning', 'Afternoon', 'Evening']
time_of_day = [random.choice(time_of_day_options) for _ in range(num_data_points)]

# Generate data for the 'Passes Gate' column
passes_gate_options = ['Yes', 'No']
passes_gate = [random.choice(passes_gate_options) for _ in range(num_data_points)]

# Create the dataset
data = {
    'Vehicle Number': vehicle_numbers,
    'Time of Day': time_of_day,
    'Passes Gate': passes_gate
}

df = pd.DataFrame(data)

# Display the dataset
print(df)
```

```
    Vehicle Number Time of Day Passes Gate
0               94   Afternoon         Yes
1               60     Evening          No
2               54     Morning         Yes
3               77     Morning         Yes
4               84   Afternoon         Yes
..             ...         ...         ...
95             100   Afternoon         Yes
96              47   Afternoon          No
97              61     Morning         Yes
98              86   Afternoon          No
99              19   Afternoon         Yes

[100 rows x 3 columns]
```

Figure 18: Checking whether the vehicles passes the railway crossing

Figure 9 shows the Python code of checking whether the vehicle passes the railway crossing or not in a particular time. Vehicle number is generated randomly from number 1 to 100. The time of the day has 3 options, ‘Morning’, ‘Afternoon’ and ‘Evening’. Then a dataset is created from the vehicle number, time of the day and the passes gate. The python output shows whether vehicles are passing the railway crossing or not on a defined time of the day.

```
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import LabelEncoder
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import accuracy_score

# Sample dataset
data = {
    'ID': [1, 2, 3, 4, 5, 6, 7, 8, 9, 10],
    'Time of Day': ['Morning', 'Afternoon', 'Evening', 'Morning', 'Afternoon', 'Evening', 'Morning', 'Afternoon', 'Evening', 'Morning'],
    'Passes Gate': ['Yes', 'Yes', 'No', 'Yes', 'Yes', 'No', 'No', 'Yes', 'Yes', 'No']
}

df = pd.DataFrame(data)

# Encode the 'Time of Day' column to numerical values
le = LabelEncoder()
df['Time of Day'] = le.fit_transform(df['Time of Day'])

# Split the dataset into training and testing sets
X = df[['ID', 'Time of Day']]
y = df['Passes Gate']
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

# Train a logistic regression model
model = LogisticRegression()
model.fit(X_train, y_train)

# Make predictions on the test set
y_pred = model.predict(X_test)

# Predict whether a car passes the gate for a given ID and Time of Day
new_data = {'ID': [11], 'Time of Day': ['Morning']}
new_df = pd.DataFrame(new_data)
new_df['Time of Day'] = le.transform(new_df['Time of Day'])
prediction = model.predict(new_df[['ID', 'Time of Day']])
print(f"Prediction: {prediction[0]}")
```

```
Prediction: No
```

Figure 19: Predicting whether a vehicle crosses the railway crossing or not in a particular time

Figure 10 shows the python code of prediction of whether a vehicle crosses the railway crossing or not in a particular time. This is the main objective of the component. ‘Linear Regression’ model is used to do the prediction since it gave the highest prediction accuracy rate. The ‘Linear Regression’ model gave a prediction accuracy about 75% - 80% while other models like ‘Decision Tree’, ‘Random Forest’ gave the prediction percentage less than 75%. The percentage of the accuracy is low than expected from the ‘Linear Regression’ model as well since the data is always updating. The code splits the dataset into training and testing sets. The train_test_split() function from the sklearn.model_selection module is used to do this. The test_size parameter is set to 0.2, which means that 20% of the data will be used for testing and the remaining 80% will be used for training.

The LogisticRegression() class from the sklearn.linear_model module is used to do this. The model is trained on the training set. The code then makes predictions on the testing set. The predict() method of the LogisticRegression() class is used to do this.

Finally, the code predicts whether a car passes the gate for a given ID and Time of Day. A new dataset is created with the given ID and Time of Day. The Time of Day column is encoded to numerical values. The predict() method of the LogisticRegression() class is then used to make a prediction.
The output of the code is the prediction, which is either Yes or No.

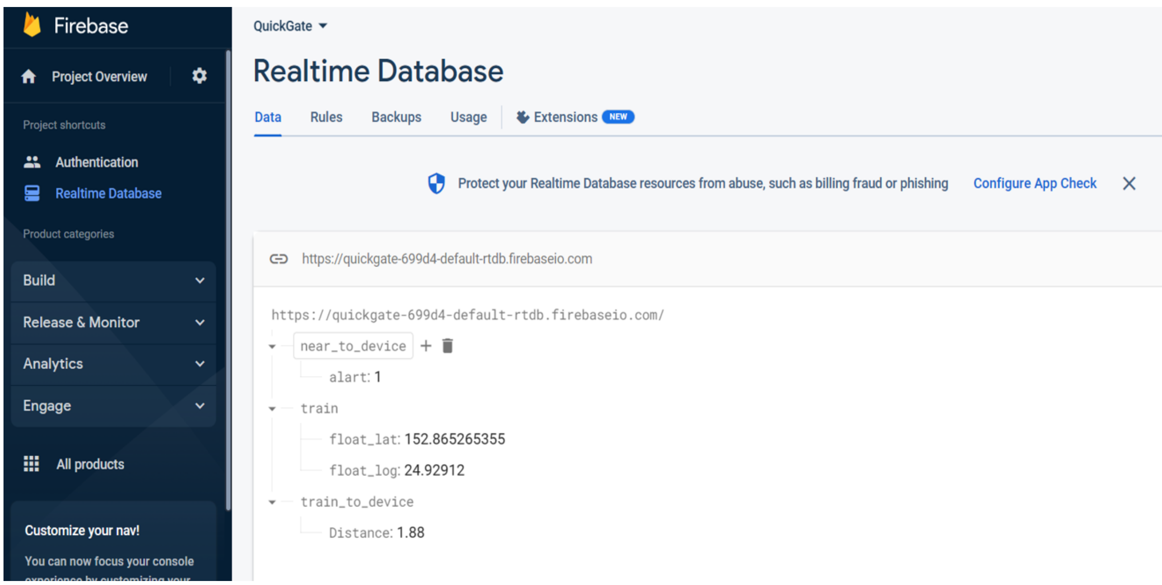


Figure 20: Data retrieved to Firebase Database

Figure 11 shows that the data is retrieved to the Firebase real-time database. In here, ‘near_to_device’, ‘alert’: 1 means the train is approaching towards the railway crossing. The ‘alert’: 0, when a train is not approaching towards the railway crossing. ‘train’, displays the current location co-ordinates of the train as longitude and latitude. The ‘train_to_device’, ‘Distance’ shows the distance between the railway crossing and the train.

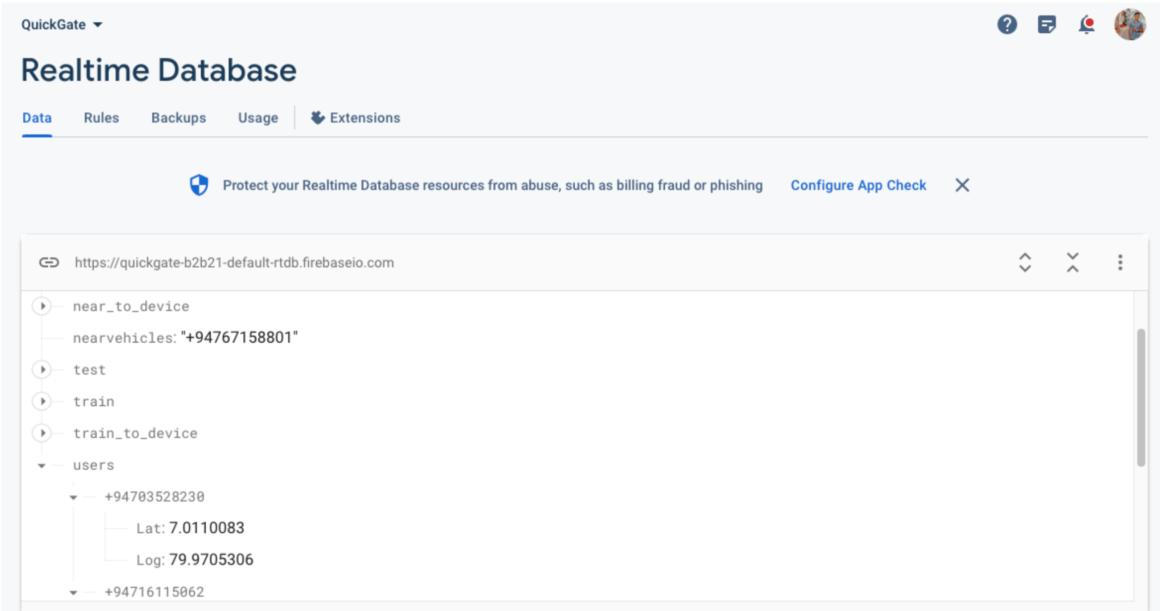


Figure 21: Users and users near the railway crossing

In the above figure, 'near_to_device' shows the vehicles near to the railway crossing by their registered mobile number in the Firebase. 'users' shows all the users who registered to the mobile application with their registered mobile number and their current location.

```
config = {
  "apiKey": "AIzaSyDM3I92kTbsD9NWK0nW2Px2W3sOIMW-QcI",
  "authDomain": "quickgate-699d4.firebaseapp.com",
  "databaseURL": "https://quickgate-699d4-default-rtdb.firebaseio.com",
  "projectId": "quickgate-699d4",
  "storageBucket": "quickgate-699d4.appspot.com",
  "messagingSenderId": "473876994840",
  "appId": "1:473876994840:web:f5354086885508f839dc6d",
  "measurementId": "G-4ZCF1TWTP5"
}

app = FastAPI()

firebase = pyrebase.initialize_app(config)
db = firebase.database()
```

Figure 22: Connecting Firebase with the Backend

Figure 13 shows the code to connect the backend with the Firebase cloud database.

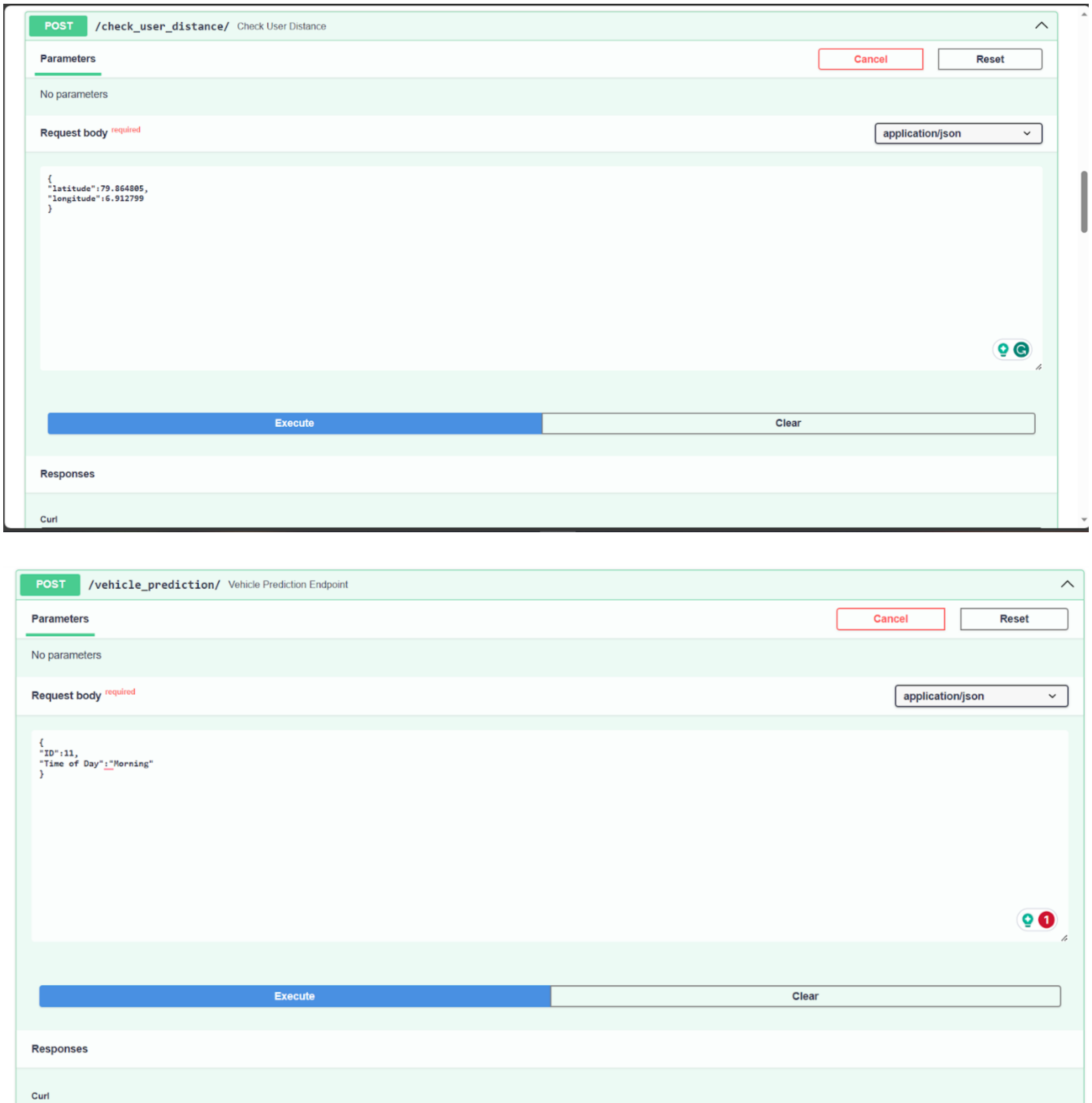


Figure 23: Output display using API

In Figure 14, the prediction output of likelihood of a vehicle crossing the railway can be seen using an API called 'FastAPI'. The vehicle ID and the time of the day can be typed and the API shows the predicted output of whether that vehicle with the ID is crossing the railway or not on the particular time of the day.

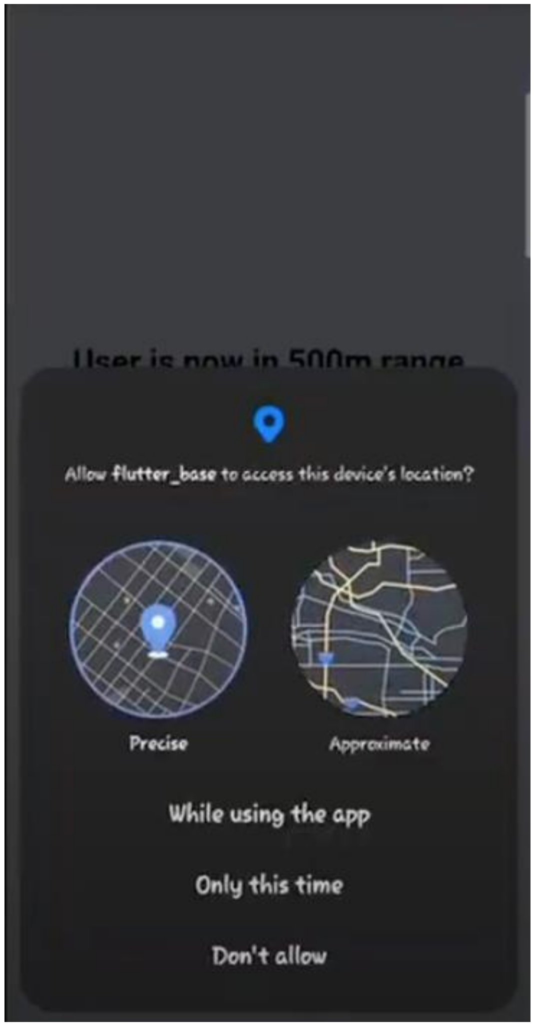


Figure 24: Mobile application asking location access

When the user who downloaded and installed the mobile application, should allow to access the location of the mobile since the system need to track the location of the particular user to send alerts.

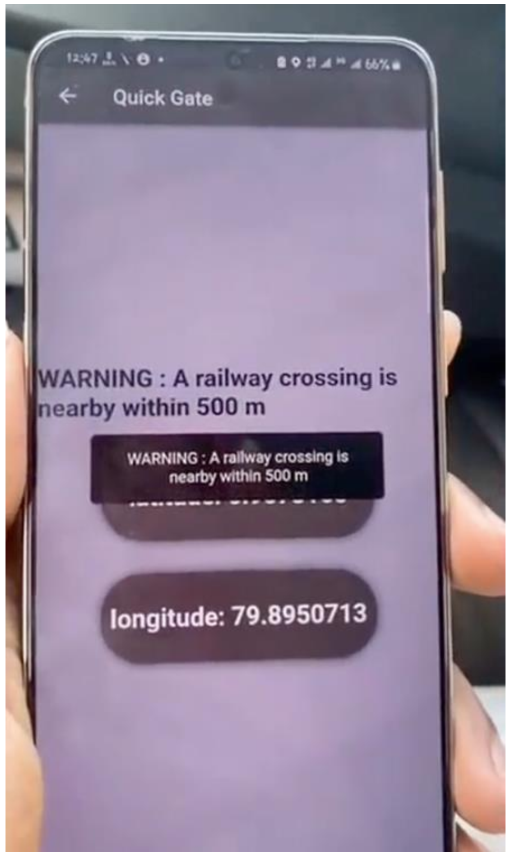
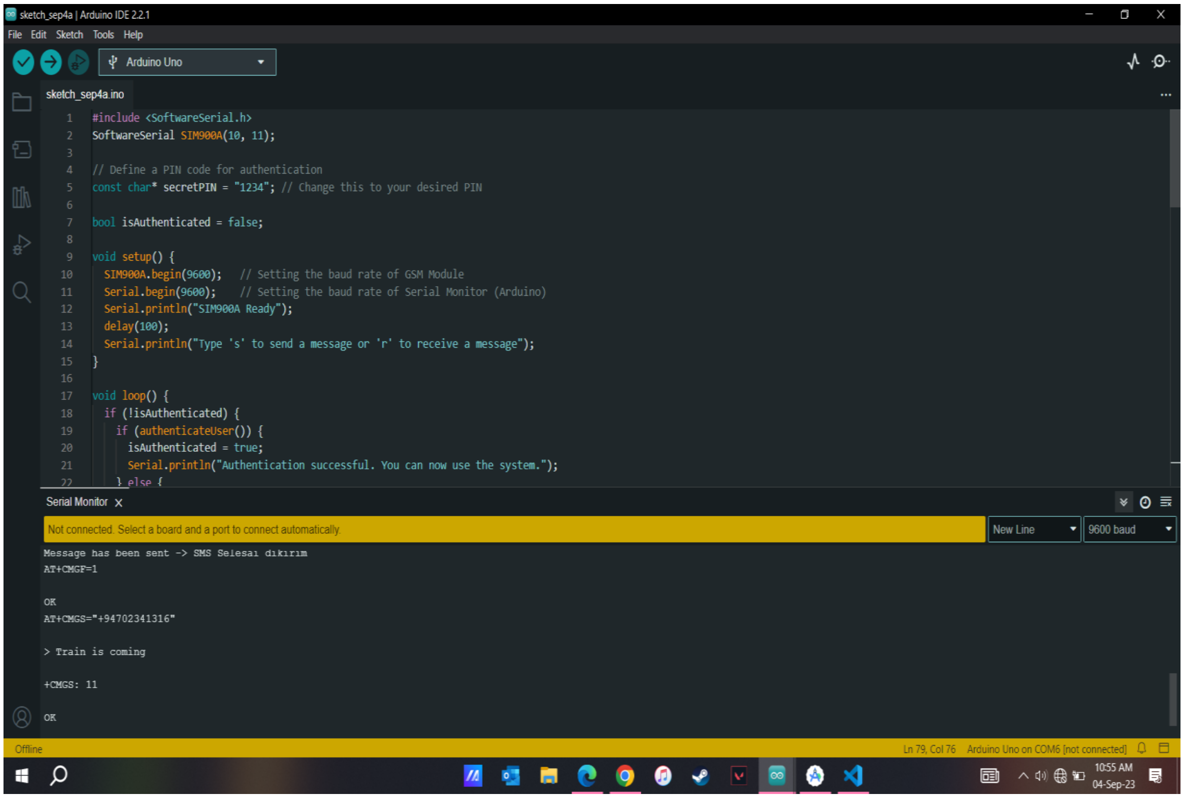


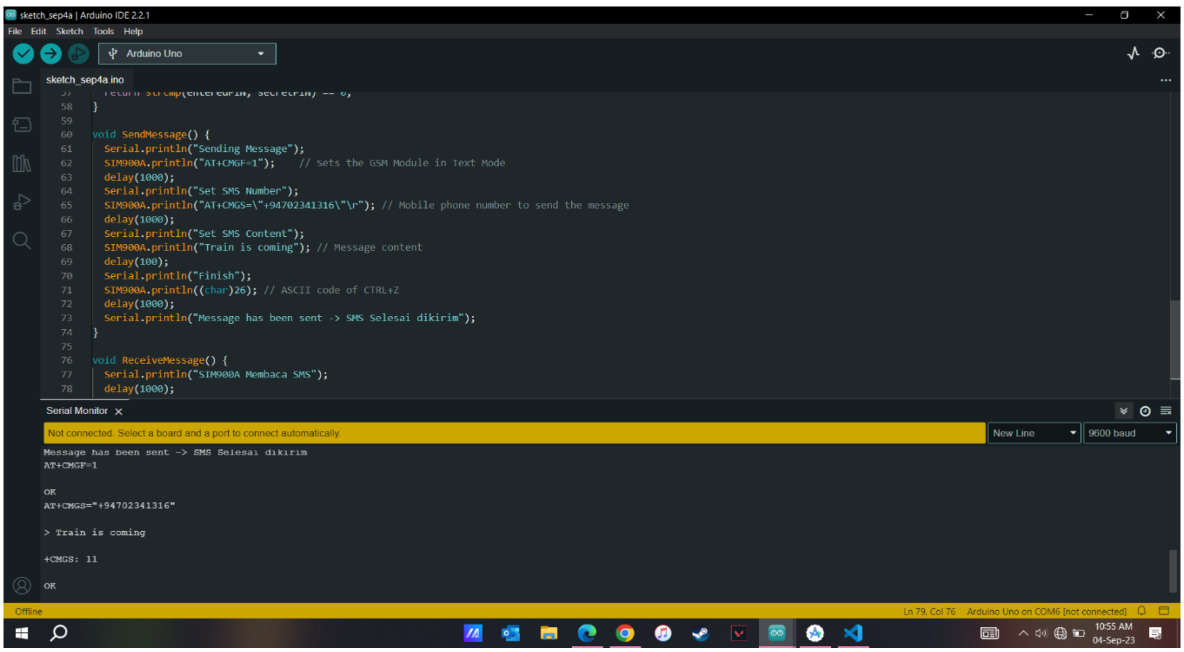
Figure 25: Output displaying using the Mobile Application

When the mobile application opens the mobile application and enters inside the 1km range circle surrounded by the IoT device near the railway crossing, the first alert notification comes to the user via the mobile application that a railway crossing is nearby within 1km range. After that particular user enters inside the 500m range surrounded by the IoT device near the railway crossing, the second alert notification comes to the user via the mobile application that a railway crossing is nearby within 500m range.

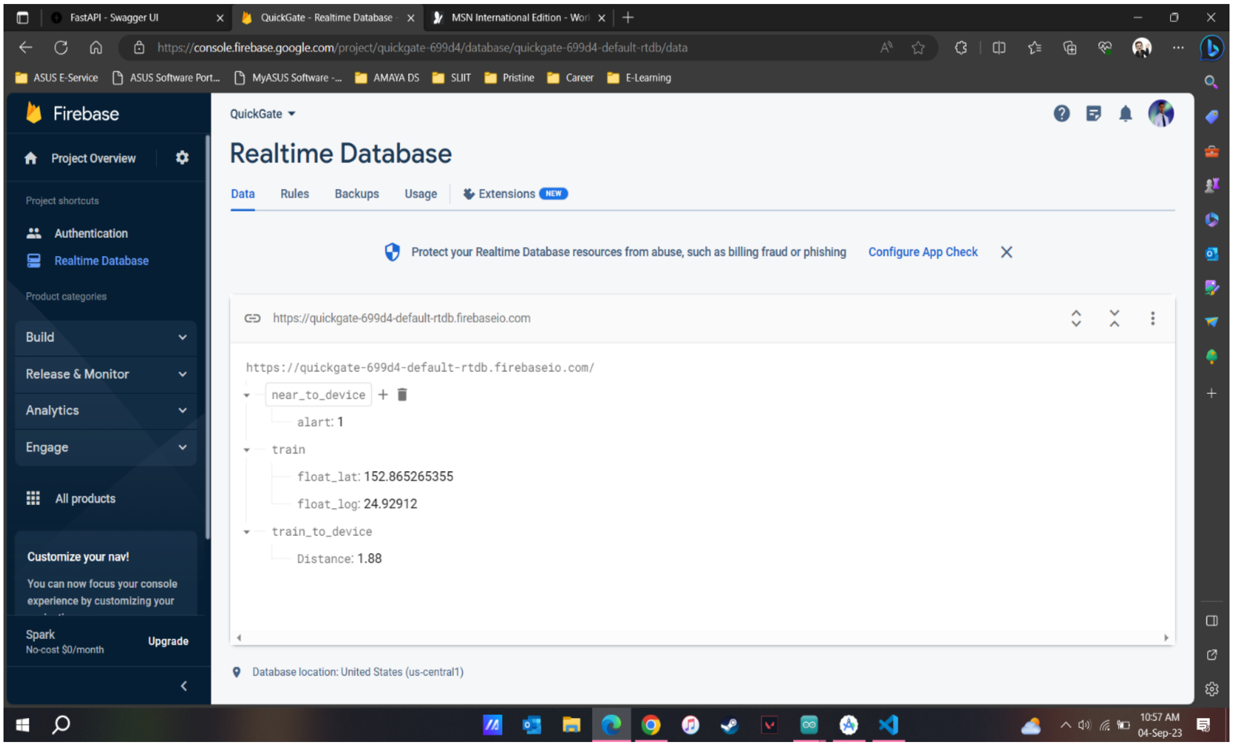
7.3. Security Analysis for the Train Detection and Alerting System



sketch_sep4a | Arduino IDE 2.2.1
File Edit Sketch Tools Help
Arduino Uno
sketch_sep4a.ino
#include <SoftwareSerial.h>
SoftwareSerial SIM900A(10, 11);
// Define a PIN code for authentication
const char* secretPIN = "1234"; // Change this to your desired PIN
bool isAuthenticated = false;
void setup() {
SIM900A.begin(9600); // Setting the baud rate of GSM Module
Serial.begin(9600); // Setting the baud rate of Serial Monitor (Arduino)
Serial.println("SIM900A Ready");
delay(100);
Serial.println("Type 's' to send a message or 'r' to receive a message");
}
void loop() {
if (!isAuthenticated) {
if (authenticateUser()) {
isAuthenticated = true;
Serial.println("Authentication successful. You can now use the system.");
} else {
Serial Monitor
Not connected. Select a board and a port to connect automatically.
New Line
9600 baud
Message has been sent -> SMS Selesai dikirim
AT+CMGF=1
OK
AT+CMGS="+94702341316"
> Train is coming
+CMGS: 11
OK
Offline
Ln 79, Col 76 Arduino Uno on COM6 [not connected]
10:55 AM
04-Sep-23



sketch_sep4a | Arduino IDE 2.2.1
File Edit Sketch Tools Help
Arduino Uno
sketch_sep4a.ino
}
void SendMessage() {
Serial.println("Sending Message");
SIM900A.println("AT+CMGF=1"); // Sets the GSM Module in Text Mode
delay(1000);
Serial.println("Set SMS Number");
SIM900A.println("AT+CMGS=\"+94702341316\"\r"); // Mobile phone number to send the message
delay(1000);
Serial.println("Set SMS Content");
SIM900A.println("Train is coming"); // Message content
delay(100);
Serial.println("Finish");
SIM900A.println((char)26); // ASCII code of CTRL+Z
delay(1000);
Serial.println("Message has been sent -> SMS Selesai dikirim");
}
void ReceiveMessage() {
Serial.println("SIM900A Membaca SMS");
delay(1000);
Serial Monitor
Not connected. Select a board and a port to connect automatically.
New Line
9600 baud
Message has been sent -> SMS Selesai dikirim
AT+CMGF=1
OK
AT+CMGS="+94702341316"
> Train is coming
+CMGS: 11
OK
Offline
Ln 79, Col 76 Arduino Uno on COM6 [not connected]
10:55 AM
04-Sep-23



The process of user sign-up within a mobile application, updating this information to the API system for new registrants, delivering a sign-up link to the user's phone, and guaranteeing the security of this entire process is a sophisticated and crucial component of current app development. Each stage in this complex process is essential for providing a seamless and safe user experience and is interconnected with the others.

User registration begins with individuals signing up for a mobile application. During this phase, users provide essential personal information, such as their names, email addresses, and passwords. This data acts as the foundation for their interaction with the app. It is vital to ensure that this initial data collection is secure, user-friendly, and compliant with data protection regulations. The programme, which acts as the backend of the app, must effectively communicate this data to the API system when users finish the registration procedure. Sensitive user data must be protected from potential vulnerabilities during data transmission in a secure manner.

Both the mobile app and the API system are essential for maintaining the security and accuracy of data. The application performs initial validation tests to ensure that the data supplied by the user complies with specified requirements, such as email format and password difficulty. These tests are essential for reducing data integrity issues and input errors. When the API system receives the user's data, it takes control and performs additional validation, verification, and security checks. These checks are necessary for things like spotting duplicate accounts, verifying the accuracy of the user's email address, and enforcing adherence to the security regulations of the system. The API system creates a new user account once all checks are successful and stores the user's data safely in the backend database.

A secure link is created and sent to the user's phone via mobile technology. Users can utilise this link to validate their registration and activate their accounts. The user's phone plays a key role in this process, highlighting the importance of secure transmission and link verification. Strong encryption techniques, secure communication routes, and authentication processes must be in place for the entire process to be secure. By utilising both something the user has (such as their mobile device) and something they know (such as their password), multi-factor authentication (MFA) can add an additional layer of protection. To recognise and counter emerging risks, continual monitoring and security evaluations are crucial.

Implementation Process

The implementation of the Train Detection and Alerting System involved several steps, including:

System Design:
The system design phase focused on defining the architecture, components, and functionalities of the system. Various technologies, such as IoT devices, SIM cards, and Integration of GPS tracking facilitated real-time train detection and alerting. The vehicle ID and the time of the day can be typed and the API shows the predicted output of whether that vehicle with the ID is crossing the railway or not on the particular time of the day.

Development Phase: During the development phase, the system components were coded and programmed. The mobile application interface was constructed using the Flutter framework, whereas backend development employed Python and Java. The system was devised to interface with IoT devices for the collection of train location data and the dissemination of user alerts.

Integration and Testing Phase: Following the completion of individual component development, integration and testing procedures were carried out a unified system. Integration testing involved verifying the proper functioning of different modules and ensuring seamless communication between the mobile application, IoT devices, and backend servers. Extensive testing was conducted to ensure the accuracy and reliability of train detection and alerting.

Strengths & Weaknesses

In the course of the implementation phase, diverse strengths and weaknesses of the Train Detection and Alerting System have been identified. Noteworthy strengths encompass the system's proficient achievement of real-time train detection, facilitating the swift issuance of user alerts. Additionally, the mobile application interface is distinguished by its user-friendly attributes, thereby enhancing the overall user experience. Moreover, the system demonstrates adept integration with Internet of Things (IoT) devices, effectively harnessing them for the acquisition of precise train location data and subsequently ensuring accurate dissemination of alerts.

In contrast, several weaknesses have been discerned. The system exhibits limited coverage, as its operational scope is currently confined to a radius of 1.5 km. This constraint may impede its efficacy within expansive railway networks. Furthermore, a notable weakness lies in the system's dependency on the functionality and availability of IoT infrastructure, thereby potentially introducing vulnerabilities to its operation**.**

Future Improvements

In light of insights derived from the implementation process and the assessment of strengths and weaknesses, several potential avenues for enhancing the system's capabilities come to the fore. One such avenue involves the expansion of the coverage region. The system's current coverage, limited to a radius of 1.5 km, prompts consideration of a broader geographical span to better serve extensive railway networks.

Another prospect pertains to scalability improvement. As user numbers grow and railway intersection points become more intricate, strategies should be devised to fortify the system's scalability, ensuring optimal performance in the face of increasing demands.

Furthermore, enhancing fault tolerance emerges as a crucial goal. By incorporating redundancy mechanisms and backup systems, the system can be made resilient against potential failures of IoT devices or disruptions in connectivity, thereby maintaining continuous and reliable operation.

8. USED TOOLS AND TECHNOLOGIES

8.1. Develop a system that utilizes GSM trackers on trains and IoT devices at railway crossings to predict and alert potential blind spots on the train

- **GSM technology:** The system heavily relies on GSM technology to transmit data and communicate between the GSM tracker on the train and the IoT devices at the railway crossings
- **IoT devices:** The system needs IoT devices to detect approaching trains and send an alert to nearby devices.

- **Machine learning algorithms:** The use of machine learning algorithms can help predict the location of a lost GSM tracker signal in the railway industry.

- **Sensors:** The system requires sensors to detect the presence of a train at the railway crossing and to collect data on various parameters, such as speed, acceleration, and location.

- **Microcontrollers:** Microcontrollers are essential for the development of the GSM tracker and IoT devices as they enable them to process data and communicate with other devices.

- **Cloud computing:** Cloud computing can be used to store data and provide a platform for real-time data analysis.

- **Wireless communication protocols:** Wireless communication protocols like ZigBee, and Wi-Fi can be used to establish communication between different device

8.2. Sending alerts to the users via the app and predicting the likelihood of crossing the railway-crossing on a given day

Purpose	**Technology**
Model Building	Python

Data Storing	Firebase Real-time database
Mobile App Development	Flutter
Version Control System	GitLab
IDEs	Visual Studio Code Andriod Studio
Machine Learning Techniques	Linear Regression, Decision Tree, SVM

Table 4: Tools and technologies Used

8.3. Security Analysis for the Train Detecting and Alerting System

When it comes to building modern and efficient applications, the choice of tools and technologies is crucial. In this report, we will delve into the tools and technologies used in the development of a web or mobile application, particularly in the area of frontend, backend, and database.

For the frontend of the application, the choice is the Flutter framework. Flutter is an open-source framework developed by Google that uses a single codebase to build high-performance, visually appealing, and natively compiled applications for mobile, web, and desktop platforms. With Flutter, developers can build custom widgets, responsive UI, and leverage hot-reload capabilities to speed up the development process.

On the other hand, the backend of the application is built with Python, one of the most popular programming languages used for web development. Python is a powerful and versatile language that is widely used for data analysis, machine learning, and building web applications. Python provides a vast array of libraries and frameworks that make it easy to develop backend APIs and web services that power modern applications.

Finally, for the database, the choice is Firebase, a cloud-based database service from Google that provides real-time data synchronization and offline support for mobile and web applications. Firebase is designed to make it easy for developers to store and retrieve data from the cloud and provide secure authentication and access control for users.

In conclusion, the combination of Flutter for frontend, Python for backend, and Firebase for the database provides a powerful and efficient solution for building modern web and mobile applications. With these tools and technologies, developers can create robust, visually appealing, and highly functional applications that can scale to meet the needs of a growing user base.

9. SYSTEM REQUIREMENTS

9.1. User Requirements

- User should have a smart phone.
- Mobile Application should be installed to the smart phone.
- Mobile phone should be switched on.
- Capability of understanding simple English.
- Guidance if a user is unable to use the application alone.
- Message alerts should be viewed regularly.
- App notifications should be viewed regularly.
- User registration details should be provided correctly.
- The system should be able to detect approaching trains and predict their potential blind spots at railway crossings.
- The system should be able to alert nearby devices when a train is approaching a railway crossing.
- The IoT device at the railway crossing should be able to receive and interpret the location data transmitted by the GSM tracker on the train.
- The system should be able to predict the location of a lost GSM tracker signal using manually trained datasets.
- The system should be reliable and accurate in its predictions and alerts to ensure the safety of railway crossings.
- The system should be scalable and adaptable to different railway networks and environments.
- The system should comply with relevant safety standards and regulations.

9.2. Functional Requirements

- **Providing accurate predictions** – The component should be able to provide accurate prediction about the vehicles likelihood of crossing the railway crossing with higher accuracy.

- **Vehicles (Users) identification -** The component should be able to identify the vehicles that are moving towards the railway crossing.

- **User Location Tracking -** The component should be able to track the location of the user's vehicle when they enter the range circle near the railway crossing.

- **Integration with the App** - The component should be integrated with the existing mobile application for the railway crossing safety system.

- **Real-time updates** - The component should be able to provide real-time updates about the location of incoming train and the railway crossings ahead in the road to the user.

- **Vehicle movement pattern storage** - The component should be able to store the tracked patterns of the user's vehicle.

- **Alert Delivery** - The component should be able to send alerts only to the app users who are approaching the railway crossing.

- **Notification Delivery -** The component should be able to send a notification to the user's SIM card if they are predicted to cross the railway crossing on a particular day, even if they are not using the app or have mobile data turned off.

9.3. Non-Functional Requirements

- **Ease of Usability:** A system that is easy to use will be more likely to be used by users, which can lead to increased productivity and satisfaction.

- **High performance:** A system that performs well will be more responsive and efficient, which can lead to a better user experience.

- **Availability**: A system that is available when users need it can help to avoid downtime and lost productivity.

- **Reliability:** A reliable system is less likely to fail, which can help to protect data and avoid financial losses.

- **Security:** A secure system can help to protect sensitive data from unauthorized access, which can help to avoid legal problems and financial losses.

- **Performance:** The system should meet certain performance requirements, such as response time, throughput, and scalability.

- **Reliability:** The system should be reliable and should not fail frequently.

- **Security:** The system should be secure and should protect sensitive data from unauthorized access.

- **Usability:** The system should be easy to use and should have a good user interface.

- **Maintainability:** The system should be easy to maintain and update.

- **Portability:** The system should be portable and should be able to run on multiple platforms.

- **Scalability:** The system should be scalable and should be able to handle increasing loads.

- **Interoperability:** The system should be interoperable and should be able to communicate with other systems.

- **Testability:** The system should be testable and should have good testability features.

- **Documentation:** The system should be well-documented and should have good documentation features.

- **Simple UI / UX designs**: Simple UI/UX designs can make the system easier to learn and use, which can also lead to increased productivity and satisfaction.

10.COMMERCIALIZATION

Identifying the target Audience: Since our ultimate goal of this project is to reduce the number of collisions happen in the railway crossings in Sri Lanka, we are targeting this system to whole people in Sri Lanka, there is no limitation.

Social media marketing: Social media platforms such as Facebook, Twitter, Instagram, YouTube and LinkedIn offer a cost-effective way to promote our product. By creating and sharing engaging content on social media, researchers can reach a wider audience and increase the visibility of their research. Social media can also be used to engage with potential collaborators and industry partners.

Partnership with a reputed company: Partnerships and collaborations with industry partners and other research institutions can help to commercialize research findings. By partnering with organizations that can provide resources, funding, or expertise, researchers can accelerate the commercialization process and increase the impact of the research. As the research project team, we expect to commercialize our final output product by building some partnership with well reputed company.

Attending Award Competitions: Attending conferences and events related to the research topic can provide opportunities to network with potential collaborators and industry partners. Researchers can also use these events to present their research findings and gain valuable feedback from peers and experts in the field.

11. CONCLUSION

In this paper, we discussed the methodology for creating a proposed train detection and alert system. The system uses sensors, cameras, and machine learning algorithms to detect trains and alert people when they approach a railway crossing. We developed a secure and user-friendly UI to display real-time information about the status of the crossing. We tested the system in a controlled environment and in the real world, and made modifications based on feedback from users. The proposed system has the potential to improve the safety of people at railway crossings and reduce the number of accidents and incidents.

Figure 26: Major railways collisions happened in the World.

12. REFERENCES

[1] N. News, “NBC News,” 27 April 2005. [Online]. Available: https://www.nbcnews.com/id/wbna7647780.

[2] N. A. K. Ragulan, “A STUDY ON RAILWAY-ROADWAY LEVEL CROSSING SAFETY,” Department of Civil Engineering, Faculty of Engineering, Sri Lanka Institute of Information Technology.

[3] N. Fazlulhaq, “Sunday Times,” 12 June 2011. [Online]. Available: https://www.sundaytimes.lk/110612/News/nws_23.html.

[4] S. L. Tweet, “Twitter,” 25 May 2019. [Online]. Available: https://twitter.com/SriLankaTweet/status/1132093011997863936.

[5] L. e. al, “Use of IoT devices to enhance railway safety.,” 2019. [Online]. Available: https://www.tandfonline.com/doi/full/10.1080/13675567.2020.1757053 .

[6] Z. e. al, “se of machine learning algorithms to predict potential hazards at railway,” 2020. [Online]. Available: https://www.mdpi.com/2076-3417/12/20/10572.

[7] M. e. al, “Use of GSM trackers to optimize railway transportation,” 2018. [Online]. Available: https://www.hindawi.com/journals/cin/2022/3211512/ .

[8] R. S. Nangalia, “Rhabdomyosarcoma involving maxilla mimicking gingival enlargement: A diagnostic challenge,” *BMJ Case Reports CP*.

[9] Sutrisno, “Determinants of employee performance: Overview of aspects of communication, work stress and compensation,” *Budapest International Research and Critics Institute-Journal* , pp. 19259-19270.

[10] A. Sharma, “A review of supervised machine learning algorithms,” *rd international conference on computing for sustainable global development (INDIACom),*, pp. 1310-1315, 2016.

[11] A. A. T. P. De Silva, “Smart Train Tracker,” University of Colombo, 2021.

[12] M. I. M. a. T. K. Amjath, “An Automated Railway Level Crossing System,” *020 International Conference on Image Processing and Robotics (ICIP),* pp. 1-7, 2020.

[13] A. K. S. S. Ž. T. G. &. P. V. Blagojević, “Evaluation of safety degree at railway crossings in order to achieve sustainable traffic management,” *Sustainability 13, no. 2 (2021),* p. 832, 2021.

[14] T. R. B. a. N. R. Council, “Traffic Signal Operations Near Highway-Rail Grade Crossings,” 1999.

[15] A. A. a. Y. K. A. Amusan, “An Automatic Railway Level Crossing System with Crack Detection,” *2022 5th Information Technology for Education and Development (ITED),* pp. 1-7, 2022.

[16] K. R. M. A. H. A. A. a. L. A. Ahmed, “A Secure Automated Level Crossing and Train Detection System for Bangladesh Railway,” *022 International Conference on Advancement in Electrical and Electronic Engineering (ICAEEE),* pp. 1-4, 2022.

[17] G. P. K. R. K. M. S. S. a. K. S. Singh, “Security system for railway crossings using machine learning,” *020 2* 139, 2020.

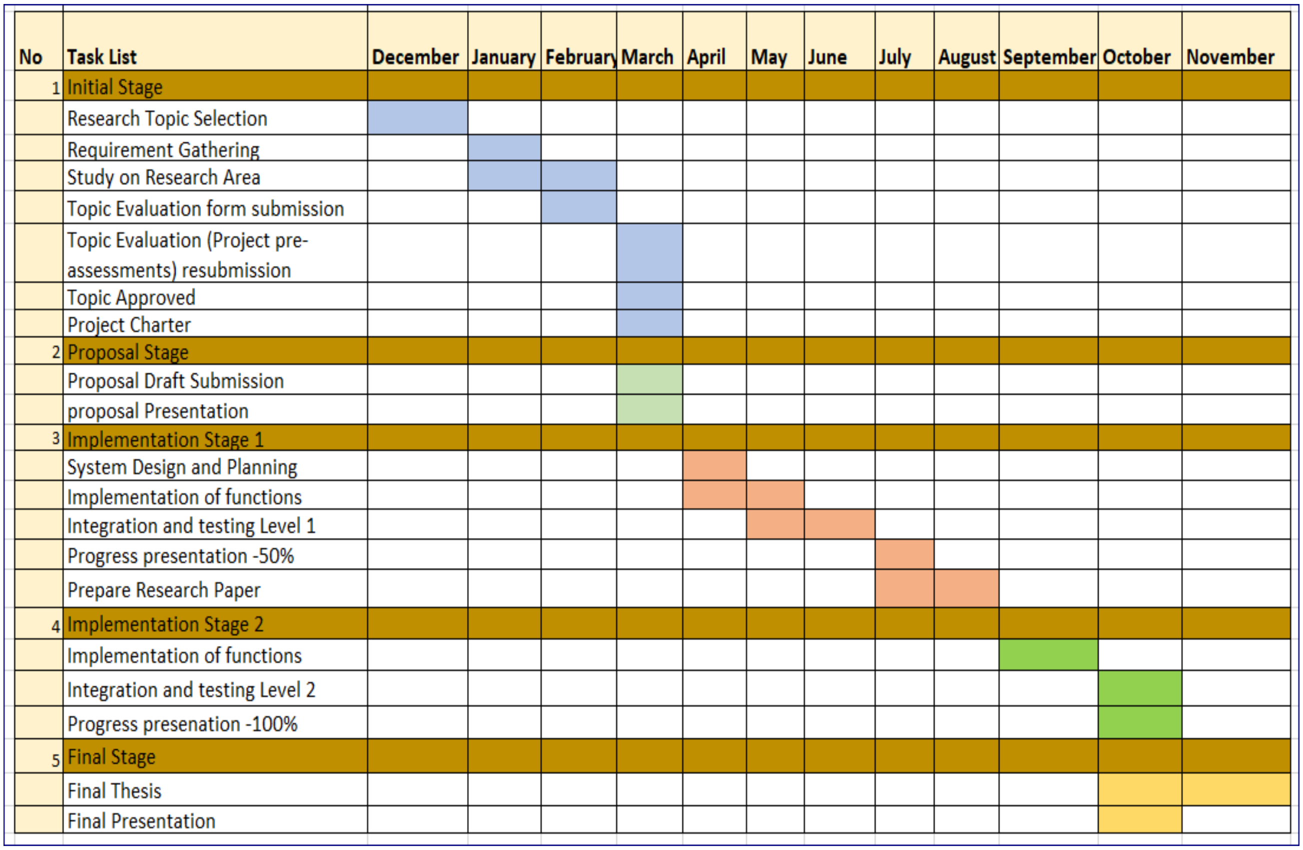
[18] rdmns.lk. [Online]. Available: https://rdmns.lk/.

[19] N. C. C. V. P. T. L. A. B. P. K. R. a. M. T. R. Brintha, “Smart Railway Crossing Surveillance System,” *2022 International Conference on Computing, Communication, Security and Intelligent Systems (IC3SIS)*, pp. 1-5, 2022.

[20] D. Suwaris, “Smart Railway Route Sensor Grid for wild life area and unsafe railway crossings in Sri Lanka,” academia.edu.

APPENDICES

Appendix A: Gannt Chart



No	Task List	December	January	February	March	April	May	June	July	August	September	October	November
1	Initial Stage												
	Research Topic Selection												
	Requirement Gathering												
	Study on Research Area												
	Topic Evaluation form submission												
	Topic Evaluation (Project pre-assessments) resubmission												
	Topic Approved												
	Project Charter												
2	Proposal Stage												
	Proposal Draft Submission												
	proposal Presentation												
3	Implementation Stage 1												
	System Design and Planning												
	Implementation of functions												
	Integration and testing Level 1												
	Progress presentation -50%												
	Prepare Research Paper												
4	Implementation Stage 2												
	Implementation of functions												
	Integration and testing Level 2												
	Progress presenation -100%												
5	Final Stage												
	Final Thesis												
	Final Presentation												

Figure 27: Gannt Chart

Appendix B: Work Break Down Chart

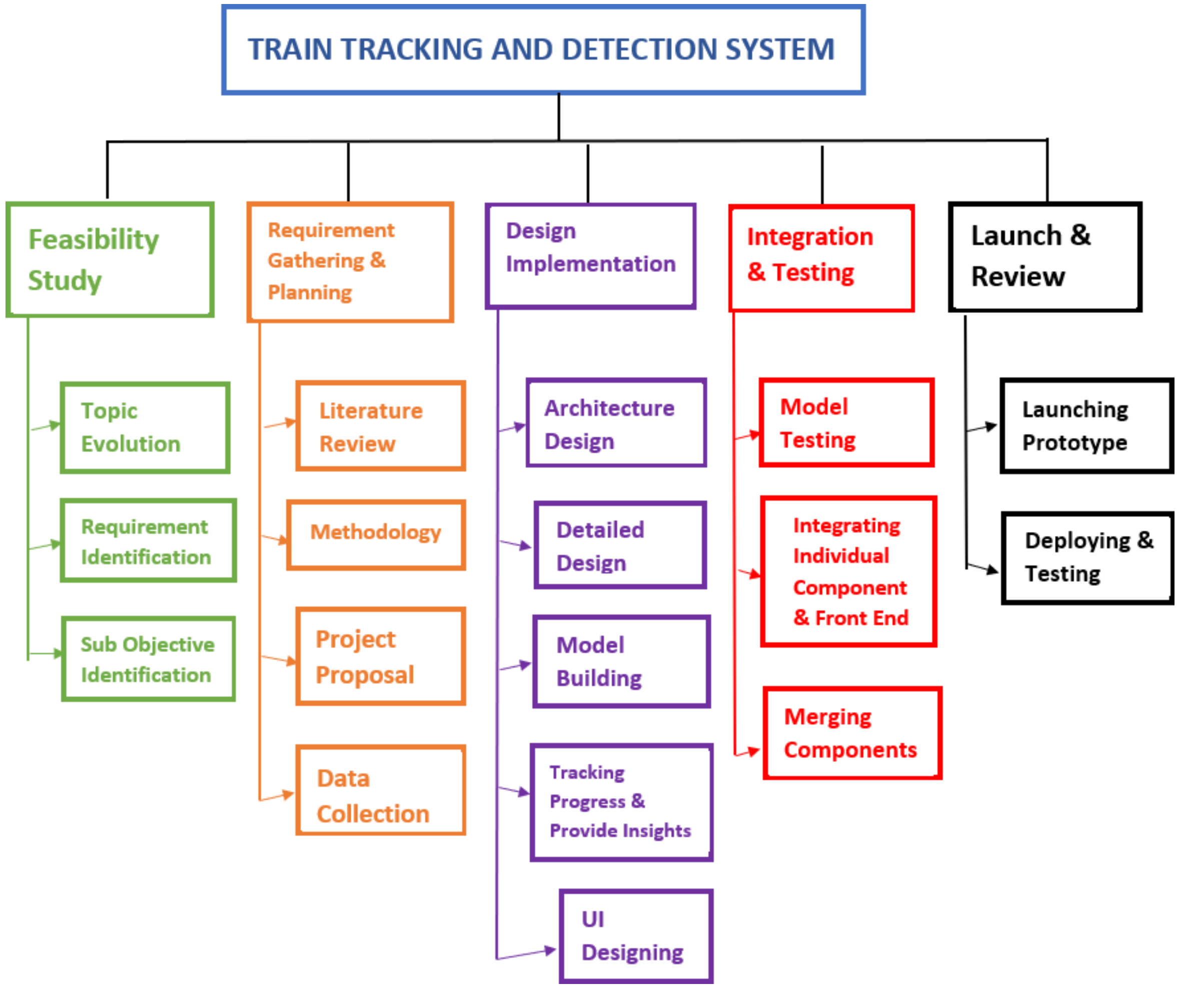


Figure 28: Work Breakdown Chart

Turnitin Report Screenshot

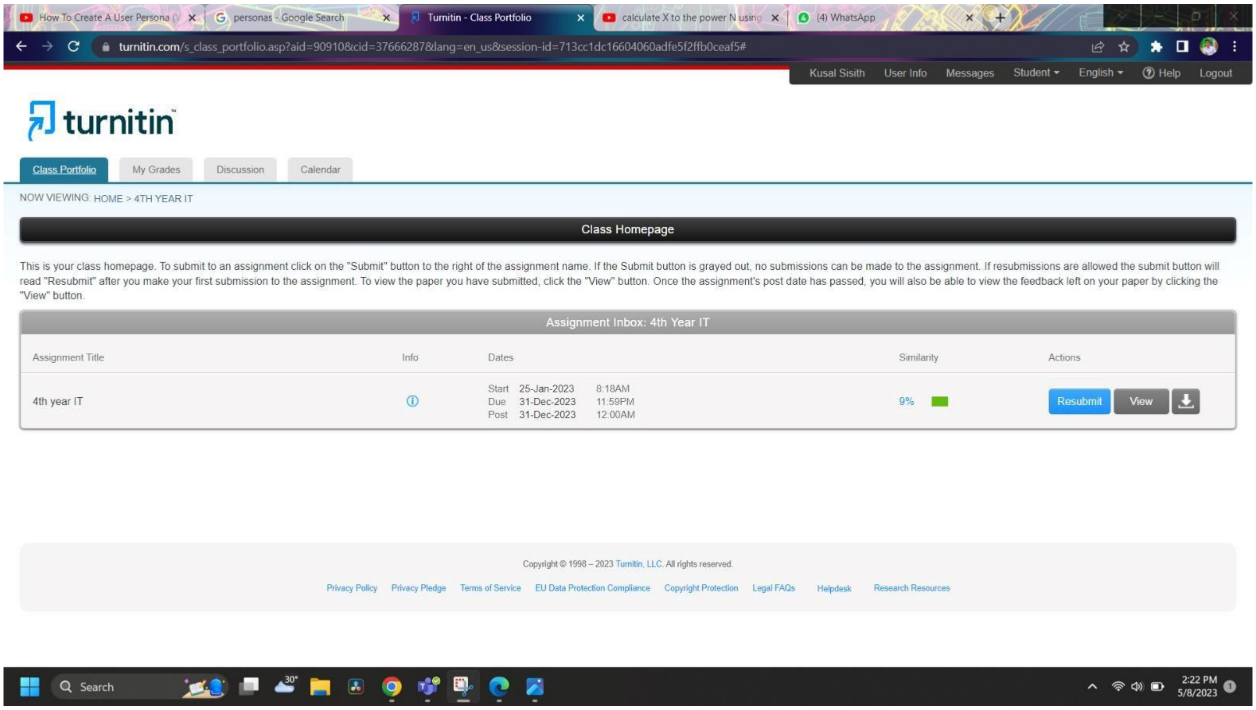
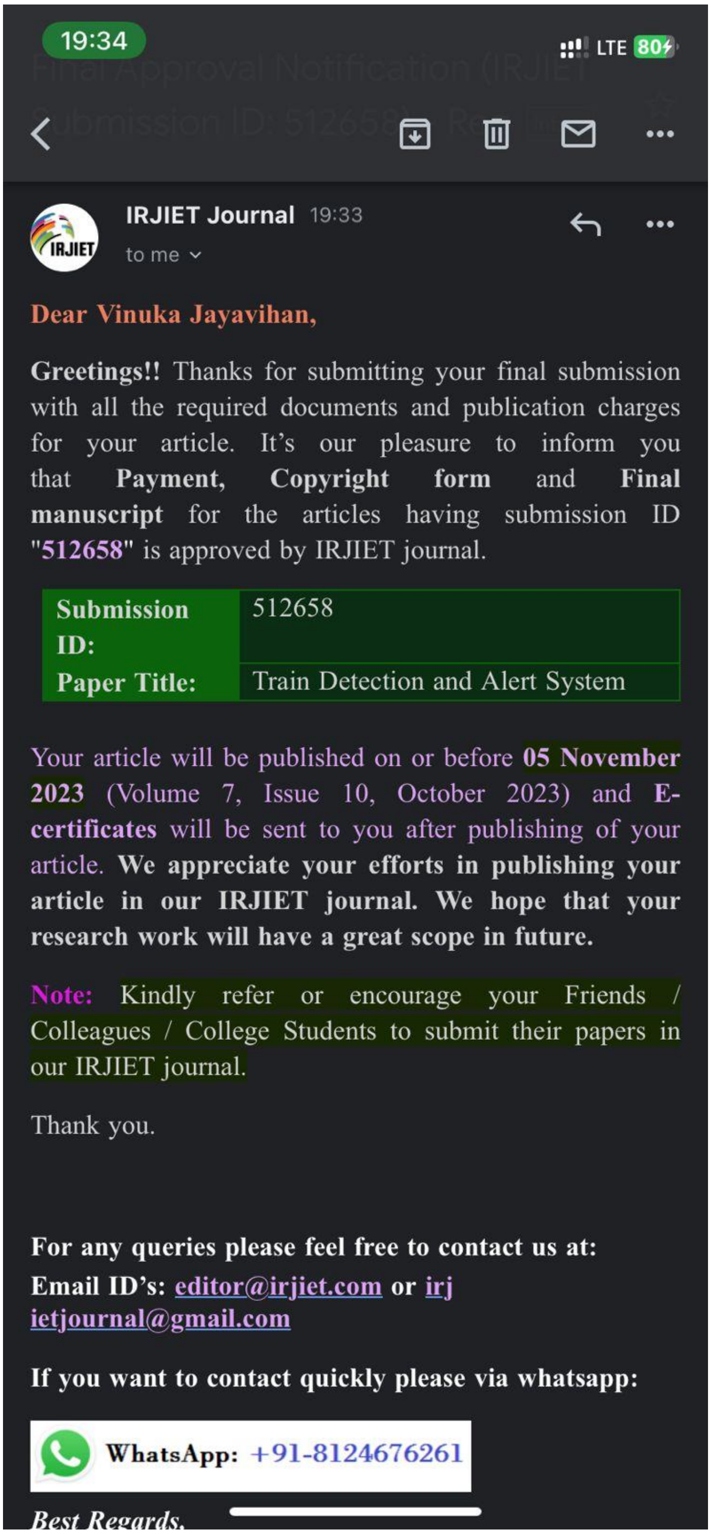


Figure 29: Turnitin Report Screenshot

Research Paper Submission & Acceptance



19:34 LTE 80

IRJIET Journal 19:33

to me

Dear Vinuka Jayavihan,

Greetings!! Thanks for submitting your final submission with all the required documents and publication charges for your article. It's our pleasure to inform you that **Payment, Copyright form** and **Final manuscript** for the articles having submission ID "**512658**" is approved by IRJIET journal.

Submission ID:	512658
Paper Title:	Train Detection and Alert System

Your article will be published on or before **05 November 2023** (Volume 7, Issue 10, October 2023) and **E-certificates** will be sent to you after publishing of your article. **We appreciate your efforts in publishing your article in our IRJIET journal. We hope that your research work will have a great scope in future.**

Note: Kindly refer or encourage your Friends / Colleagues / College Students to submit their papers in our IRJIET journal.

Thank you.

For any queries please feel free to contact us at:
Email ID's: editor@irjiet.com or irjietjournal@gmail.com

If you want to contact quickly please via whatsapp:

WhatsApp: +91-8124676261

Best Regards,

Figure 30: Research Paper Submission & Acceptance