iJRASET
International Journal For Research in Applied Science and Engineering Technology



INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 11 **Issue:** VI **Month of publication:** June 2023

DOI: https://doi.org/10.22214/ijraset.2023.54177

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 11 Issue VI Jun 2023- Available at www.ijraset.com

Easy Bus (Virtual Bus Pass and Bus Tracking System): Design and Implementation Using Machine Learning Algorithms

Nagababu Garigipati[1], C. Sharika Reddy[2], Challa Sai Charitha[3], Kalwa Pujitha[4], Rajoli Lakshmi Krithika[5]
[1]*Assistant Professor,* [2, 3, 4, 5]*Studen, G. Narayanamma Institute of Technology and Science, Hyderabad, Telangana, India*

Abstract: ***Unforeseen and atypical road conditions have an impact on the functioning of the bus movement. Daily issues such as traffic congestion, unanticipated delays, fluctuations in student demand, and regular bus scheduling occur, leading to disruptions in students' schedules and necessitating their inevitable wait for their respective buses to arrive. To alleviate this inconvenience for students, a system that offers real-time information on bus locations can be introduced.***
EASY BUS is an interactive application that simplifies the process of using college buses, making it convenient and accessible. Common issues faced by bus-riding students, such as uncertainty about bus arrival times, missed buses, or operational status on a given day, are effectively resolved by this app. To make the most of this solution, students are required to install the app and regularly update their bus passes.
This app boasts two prominent features: GPS tracking of bus locations and a virtual bus pass. The bus's whereabouts are tracked by a sensor every thirty seconds, and the collected data is transmitted to the server, which subsequently displays it on a map for the user (student). The app also offers a bus pass feature, allowing users to create new passes or renew their existing ones, with the associated charges processed through online payment methods.
Keywords: GPS, Location, Tracking, Mobile, Bus-pass, API, Cloud, Firebase

I. INTRODUCTION

Bus transportation, known as Omnibus, was first introduced in Nantes on August 10, 1826. An Omnibus consisted of two horses and could accommodate up to 16 passengers. The term "bus" is derived from the Latin word "Omnibus," meaning a carriage for all. Following the development of the Omnibus, various similar transportation systems such as horsecars, cable cars, streetcars, and modern buses were invented. In Malaysia, the development of public transportation began during the British era and has steadily improved over time. However, a number of problems have developed over time, including a decline in bus transportation's time management as a result of environmental conditions. This has caused inconveniences, particularly for students and staff members in universities who rely on buses. Their tardiness to classes can be attributed to waiting for buses without knowing their exact locations. In a fast-paced and evolving environment, time is of great importance, and any delay can lead to significant incidents, especially during exam days when students may need that extra time for last-minute revisions. One of the main challenges faced by students is uncertainty about the bus's location. They lack real-time information regarding the current whereabouts of the bus, making it difficult for them to estimate its arrival time and plan accordingly. Moreover, the absence of unique identifications for buses further complicates matters, as students are unsure which bus they should take.

II. RELATED WORK

"Real-Time Bus Position and Time Monitoring System": To address the issue of delays in public transportation, this system aims to provide real-time information about bus arrival times at specific stops. By utilizing GPS technology and online platforms, passengers can access accurate information about bus positions and arrival times, allowing them to plan their journeys effectively and reduce waiting times. This system not only improves the passenger experience but also promotes the use of public transport, leading to reduced traffic congestion and pollution.
"Real-Time GPS/GPRS Based Vehicle Tracking System": This system focuses on displaying real-time bus locations in Mumbai city through a standalone tracking device. By utilizing transmitter and receiver modules, along with LED-embedded route maps, the device provides accurate bus locations to pedestrians at bus stops. The system operates using GPS data from buses and does not require an external power source, making it cost-effective and sustainable.

©IJRASET: All Rights are Reserved

"Real-Time Availability System": This survey proposes an application that goes beyond existing bus routes and schedule information by providing real-time data on available seats and the current location of buses. The aim is to improve time management and create an efficient public transport system for daily commuters. In order to create a flexible, comfortable, easily accessible, and reliable bus service that promotes a switch from private automobiles to public transport, the survey investigates techniques to fulfil the capacity requirements of variously sized cities.

III. PROBLEM STATEMENT

The proposed system aims to detect and track the location of a bus, providing this information through a user-friendly application. The application will have a simple and clear interface, designed to be easily navigable. The system's functionality will be based on the fee status of the student [1].

IV. EXISTING SYSTEMS

Globally, numerous vehicle tracking systems have been created, each with unique characteristics and functionalities [3]. The EVO GPS Tracker is one illustration; it was created specially to improve vehicle security and offer GPS location services for vehicles [2]. While providing simple and practical methods for car management and monitoring, this device makes use of cutting-edge technology and safety features to assure the security of automobiles on their travels.

Another option is LiveViewGPS, which primarily caters to fleet vehicle owners, providing vehicle security and GPS location services. They offer a range of options for personal vehicles, buses, and professional vehicles, meeting diverse tracking needs [4]. LiveViewGPS devices incorporate advanced technology to maintain vehicle safety at all times.

The GPS tracking system for vehicles, the XSSecure XTS Tracking System, is made by Conjoinix, a global maker of security technology. A long range of capabilities are available, including distance calculation, historical tracking data auditing, transit tracking, asset tracking, video and audio surveillance, fuel monitoring, and stolen vehicle recovery. The XSSecure XTS Tracking System is a flexible and effective solution for managing and tracking vehicles thanks to these capabilities.

V. PROPOSED SYSTEM

The entire project is divided into three different modules. “Authentication” is the first module which verifies a user's identity from the database, before allowing access to the application. An SMTP server is set and it sends the verification and change password 3 emails to the registered email address [6]. “Sign-in” and “Sign-up” are the two sub-modules in the Authentication module. In the Sign-in module, the user enters the registered email address and password and accesses the application.

In the Sign-up module, the user registers using the GNITS student email address and sets a password. Each new user information is stored and updated accordingly in the database.

The second module is the “Tracking” module. In this module, the real-time location of the bus is tracked using a sensor. This location is marked on the map. The location of the bus is updated on the map, every time the position is changed. The user’s location is displayed on the map. This feature is controlled by the user. If the user wishes not to share the location, then the user's location is not displayed on the map but the bus position is displayed regardless. The initial location and the final location of the bus are displayed.

The third module is the “Bus Pass” module. After the user signs in to the application, the bus location is displayed on the map only if the student pays the bus fare. In this module, a virtual bass pass is displayed on the screen. If the student has paid the bus fare then the bus location is displayed and the student can board the bus, but if the student hasn’t paid the bus fare then a note is popped up implying the student to pay the bus fees to access the location of the bus and to board the bus.

VI. OBJECTIVES

1) To get real-time location coordinates of the bus and reduce the waiting time.
2) Generating a Bus Pass if the student's bus fee is cleared.

VII. METHODOLOGY

The EASY BUS mobile application offers real-time bus tracking and a virtual bus pass feature. It utilizes a "real-time bus tracking system" that continuously tracks the movement and locations of buses along various routes, providing live data displayed on a map. Students can access information about the initial location, final location, and current location of the bus, ensuring the uniqueness of each bus's location on the map.

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |

To track the buses, sensors are installed in each bus, and their coordinates are stored in a Google Sheet. The latitude and longitude data from the Google Sheet are then stored in Cloud Firestore, a cloud-hosted NoSQL database. Cloud Firestore follows a document-based data model, where data is stored in documents that contain fields mapping to values. These documents are organized into collections, facilitating data organization and query building. When a user selects a specific Bus ID, the bus's location is retrieved from Cloud Firestore and displayed on the map. As the coordinates change, the location on the map updates in real time, allowing users to track the bus's live location.

The application is developed using the Dart language in Android Studio, providing a simple yet efficient user interface. User information is stored in Google Firebase, a software development platform that supports iOS, Android, and web app development. Firebase offers tools for analytics, crash reporting, and conducting marketing and product experiments. The Google Maps API is a key component of the application, enabling easy exploration and navigation of maps through intuitive gestures. This integration simplifies bus tracking for users. The generation of the virtual bus pass is determined based on the student's fee status, which is stored in Google Firebase [9].

The architectural style of the system follows a three-tier layer structure, as depicted in Figure 6.1. The mobile application operates as the client layer, interacting with a server through an internet connection. The core functionality of the mobile application involves updating the user interface with requested information based on user input and retrieving data from a data store.

This system has a few modules which can break into three partitions. Those modules have their own characteristics. Following are the modules:

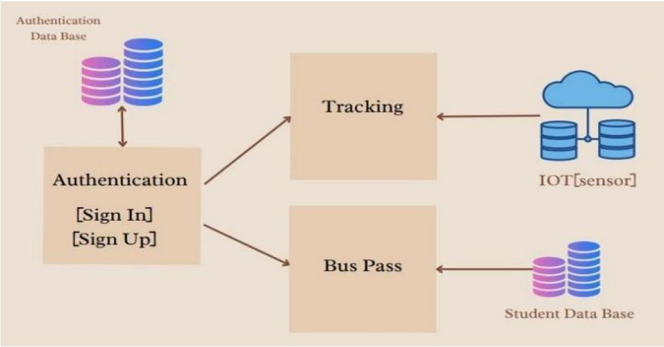


Figure 6.1: Architecture of EASY BUS application

The module design is as follows:

A. Module 1: Authentication

The Real-Time Bus Tracking Mobile Application's authentication module explains how a user interacts with its functionalities. The module specifies all of the user interfaces for functions that are offered to remote end users. Before granting access to the application, the authentication module uses the database to confirm the user's identity. Only when the user's credentials are legitimate may they access the bus's location and a virtual bus pass. This module consists of two submodules namely Sign In and Sign Up.

1) Sign-in: If the user has already created an account on EASY BUS, they can just provide their credentials to log in to the app and find where the bus currently is. The data of the users is stored in the Firebase real-time database which is used to authenticate the user details when they log in. The Firebase Realtime Database is a cloud-hosted database. Data is stored as JSON and synchronized in real time to every connected client. An SMTP server is set which sends a mail to the registered mail id to change or reset the password.

2) Sign-up: If a new user wants to know the location of the bus, they're first supposed to sign up by providing the required information. While registering, a verification link is sent to the provided mail id by the SMTP server to verify the user credentials. Later, these details are added to the Firebase real-time database.

B. Module 2: Tracking

The tracking module employs a GPS Tracker device installed on the bus to monitor its location. The GPS Tracker receives microwave signals transmitted by GPS satellites, enabling it to gather crucial information like the bus's precise location, speed, estimated time of arrival, and direction. This data is continuously updated in real-time, ensuring accuracy. The collected GPS data is then transferred to the GSM or cellular network, which facilitates communication with the mapping system.

©IJRASET: All Rights are Reserved

The mapping system processes the received data and displays the bus's location on a map, allowing users to track its movement. Users can access the historic navigation data through the Bus Tracking Mobile Application. When a user requests to check the current bus location, the application retrieves the relevant data from the mapping system and presents it to the user.

In summary, the tracking module utilizes GPS technology, satellite signals, and cellular networks to track the bus's location, speed, estimated time of arrival, and direction. This information is made available to users through the Bus Tracking Mobile Application, providing a convenient and real-time bus tracking experience.

The main functions of the tracking module are:

1) The real-time data, i.eThe current bus location fetched from the sensor is presented on the map.
2) Displays the user location.
3) Put on view the initial location of the bus and the destination.

C. Module 3: Bus Pass

Bus pass module checks if a user has paid the bus fee and generates a virtual bus pass depending on the status, i.e., paid or unpaid. A database from the administration is used to check if the user has paid the fee or not when the user credentials are entered. A pass is provided if the user has paid the fee, else a message is displayed that the fee is due.

VIII. IMPLEMENTATION

A. User Verification

From the EASY BUS application When the user signs up with their email ID their details along with the unique User ID created is recorded in the firebase authentication table as shown in Figure 8.1. This User ID is used for the Authentication Token.

Internet users can access apps, services, websites, and application programming interfaces (APIs) without having to enter their login information each time they visit thanks to authentication tokens. Instead, the user only needs to log in once, after which a special token is created and sent to any websites or applications that the user connects to.

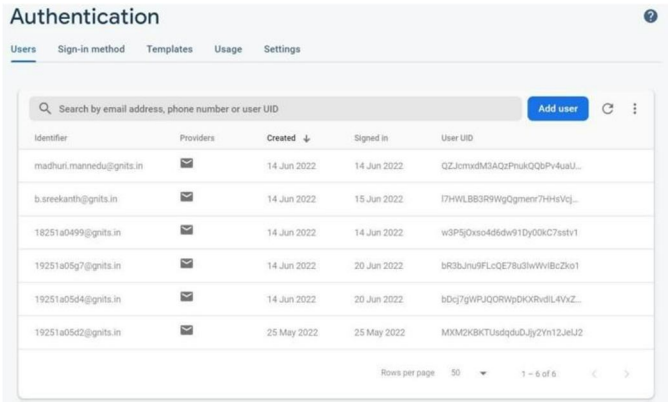


Figure 8.1: Firebase Authentication Table

The particular bus locations from the sensor in the bus are sent into the google sheets, the locations of the bus are stored in the Historic google sheet (as shown in Figure 8.3) for bus maintenance. The location of the bus is updated in the Update Google sheet as shown in Figure 8.2.

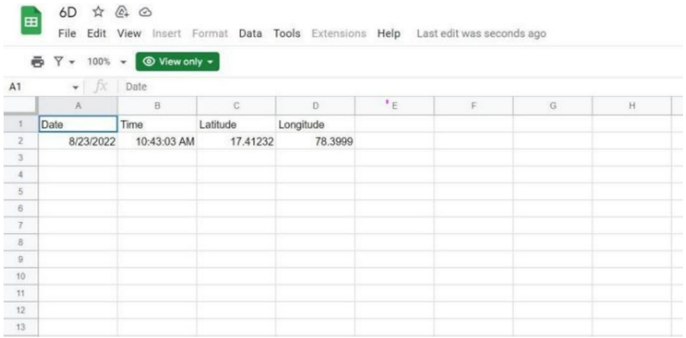
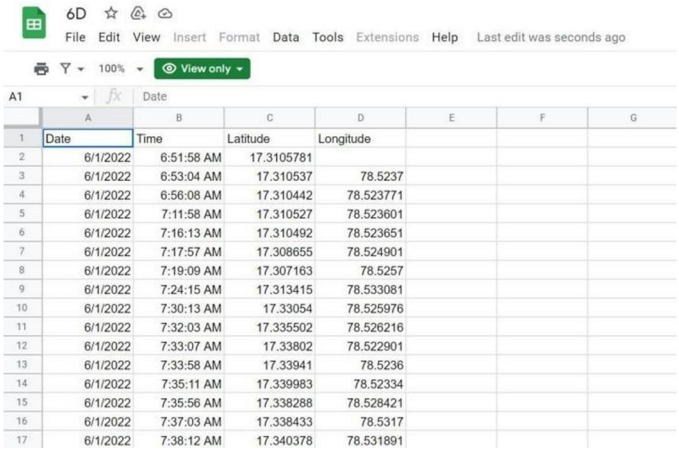


Figure 8.2: Google Sheets – Update



6D

File Edit View Insert Format Data Tools Extensions Help Last edit was seconds ago

100% View only

A1 fx Date

	A	B	C	D	E	F	G
1	Date	Time	Latitude	Longitude			
2	6/1/2022	6:51:58 AM	17.3105781				
3	6/1/2022	6:53:04 AM	17.310537	78.5237			
4	6/1/2022	6:56:08 AM	17.310442	78.523771			
5	6/1/2022	7:11:58 AM	17.310527	78.523601			
6	6/1/2022	7:16:13 AM	17.310492	78.523651			
7	6/1/2022	7:17:57 AM	17.308655	78.524901			
8	6/1/2022	7:19:09 AM	17.307163	78.5257			
9	6/1/2022	7:24:15 AM	17.313415	78.533081			
10	6/1/2022	7:30:13 AM	17.33054	78.525976			
11	6/1/2022	7:32:03 AM	17.335502	78.526216			
12	6/1/2022	7:33:07 AM	17.33802	78.522901			
13	6/1/2022	7:33:58 AM	17.33941	78.5236			
14	6/1/2022	7:35:11 AM	17.339983	78.52334			
15	6/1/2022	7:35:56 AM	17.338288	78.528421			
16	6/1/2022	7:37:03 AM	17.338433	78.5317			
17	6/1/2022	7:38:12 AM	17.340378	78.531891			

Figure 8.3: Google Sheets – Historic

B. Bus Pass Verification

The bus details are taken from the google sheets into the Cloud Firestore under 'Location' collection (shown in Figure 8.4) with track document which has the current Latitude, Latitude, Date and Time sent from the sensor.

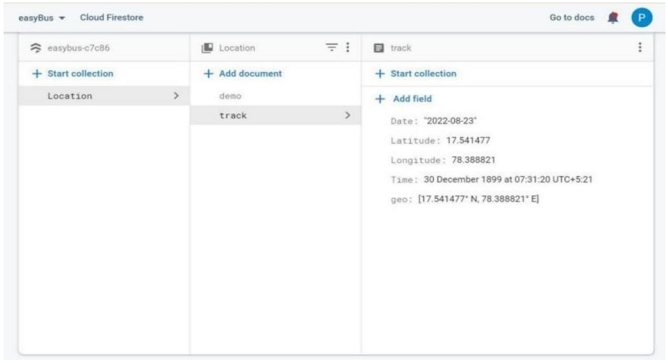


Figure 8.4: Firestore - Live Tracking

The fee details of the users are stored in the firebase-firestore and this collection is linked to the user collection via the unique student roll number as shown in Figure 8.5 (a) and Figure 8.5 (b).



Figure 8.5 (a) Fee Paid



Figure 4.5 (b) Fee Not Paid Figure 4.5: User Fee Status

C. *Technologies Used*

1) *Android Studios:* An official integrated development environment (IDE) offered by Google for building applications for the Android OS is called Android Studio. It is created exclusively for Android development and is based on the IntelliJ IDEA software from JetBrains. Android Studio, which is available for Windows, macOS, and Linux, provides a variety of tools for creating apps that work with various Android devices. It offers features that are effective and optimised for developers to speed up the process of creating apps.
2) *Flutter:* Flutter is an open-source software development kit (SDK) developed by Google for creating user interfaces (UI). It allows developers to build cross-platform applications that can run on multiple operating systems, including Android, iOS, Linux, macOS, Windows, Google Fuchsia, and the web, using a single codebase. The core engine of Flutter, written in C++, provides low-level rendering support through Google's Skia graphics library. Additionally, it seamlessly integrates with platform-specific software development kits (SDKs) offered by Android and iOS. Flutter apps are built using the Dart programming language, taking advantage of its advanced features to create high-quality user interfaces and application logic.
3) *Firebase:* Google Firebase is a comprehensive application development platform supported by Google. It offers a wide range of tools and services that empower developers to create iOS, Android, and web applications. Firebase provides various functionalities, including analytics tracking to gather insights on user behavior and app performance, crash reporting to identify and resolve application crashes, and tools for conducting marketing and product experiments to optimize user engagement. By utilizing Firebase, developers can streamline the development process and enhance the overall quality and functionality of their applications.
4) *Cloud Firestore:* Cloud Firestore, also known as Google Firestore, is a crucial component of the Google Firebase platform. It is a cloud-based NoSQL database server designed to store and synchronize data efficiently. Firestore allows web and mobile applications to interact directly with the database using native SDKs, making data management seamless. The database is known for its high performance and can automatically scale to accommodate growing data needs. Firestore is user-friendly and reliable, offering developers a range of technologies and SDKs to work with, including Java, C++, Unity, Go, Node.js SDKs, as well as REST and RPC APIs. This flexibility enables developers to leverage Firestore's capabilities across various programming languages and frameworks.
5) *Firebase Realtime Database:* The Realtime Database feature of Firebase serves the purpose of facilitating data sharing among multiple clients, including Android and iOS mobile apps, as well as web applications using JavaScript. The primary objective of this system is to provide developers with a secure, reliable, and efficient way to synchronize data with minimal coding efforts. The database is designed to scale and handle a large number of users effectively. The term "realtime" is used to describe the database because it achieves synchronization of data across clients with remarkable speed, approaching real-time performance. Although there are inherent limitations in transmitting data over the internet and wireless connections, the system ensures that the elapsed time for data changes to propagate from one client to another is visually imperceptible. This near-instantaneous synchronization enhances the real-time nature of the database. Furthermore, the Realtime Database offers data persistence by storing data locally on devices. This enables offline access to the data, ensuring that it remains accessible even when a device is not connected to the network. When connectivity is restored, the local data is automatically synchronized and merged with the remote data, ensuring data consistency across devices.
6) *Google Maps API:* Google Maps APIs are prebuilt code components that enable developers to easily integrate maps into their websites or mobile applications. These APIs provide a straightforward way to include interactive maps and offer additional functionalities to enhance the user experience. Google Maps APIs are available for various platforms, including Android, iOS, and web browsers. This allows developers to incorporate maps seamlessly into their applications across different devices and platforms. The APIs also offer HTTP web services, which enable the exchange of information between different systems. By leveraging Google Maps APIs, developers can access a wide range of features and capabilities, such as displaying maps, adding markers and overlays, calculating routes and distances, geocoding addresses, and performing spatial queries. These APIs provide a convenient and efficient solution for integrating maps and location-based services into applications, enhancing their functionality and usability.
7) SMTP Server: SMTP stands for Simple Mail Transfer Protocol, and it serves as an essential communication protocol for sending and receiving emails. It enables mail servers to handle the transmission of outgoing emails between senders and recipients. In simple terms, an SMTP server functions as a digital postman responsible for delivering emails to their intended destinations.

An SMTP server is typically a computer or a software application that implements the SMTP protocol. It acts as a mediator between the sender's email client or application and the recipient's email server. When an email is composed and sent, the SMTP server receives it from the sender and relays it to the appropriate destination server. To configure an email client or application, you need to specify the SMTP server address. Each email service provider or hosting service has its own SMTP server address. For example, Gmail uses smtp.gmail.com as its SMTP server, while Twilio SendGrid utilizes smtp.sendgrid.com. By using the appropriate SMTP server address, your email client or application knows where to send outgoing messages for delivery. The SMTP server ensures the reliable and efficient transmission of emails across the internet, enabling seamless communication between email senders and recipients.

IX. RESULTS AND DISCUSSIONS

A. User Registration

A new user can access the application by registering themselves with GNITS email and password. The email and password fields are provided on the registration page as shown in Figure 10.1. Once the password is set for the new account created, a message pops up suggesting the user to verify the registered email (implying a verification link has been sent to the registered email) as shown in Figure 10.2.



Figure 10.1 Registration form



Figure 10.2 Verification alert

B. Email Verification

Once the registration form is filled, a verification mail is sent to the email address by the SMTP server set. The mail consists of a verification link as shown in Figure 10.3, which directs you to another page Figure 10.4 after the completion of the verification process and updated into the database accordingly.

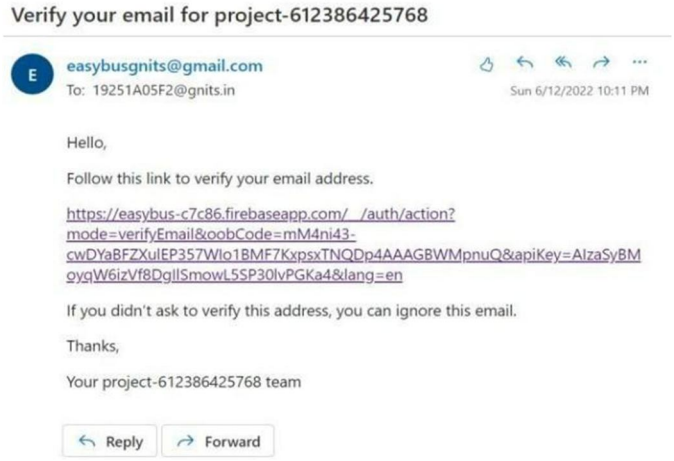


Figure 10.3 Verification mail sent by the SMTP server

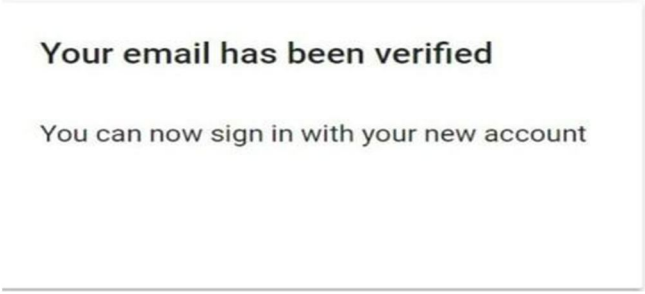


Figure 10.4 Successful verification of the registered mail id

C. Sign-in

Users can get access to the application features in Fig-6 using their login credentials, i.e, by entering their GNITS email id and password in the login form as shown in Figure 10.5. To reset the password, a reset password link is sent to their maid id using the SMTP server as shown in case of email verification during registration process in Figure 10.3. After signing into the application, the user can either view the location of the bus or their virtual bus pass as shown in Figure 10.6.



Figure 10.5 Login form



Figure 5.6 Features of EASY BUS

D. Bus Tracking

After successful login, the user is given access to view the location of the bus. Users are prompted to allow access to their location, in case of denial only three markers are displayed on the map. The three markers denote the current location of the bus, initial location of the bus and the destination, i.e, the college location as depicted in Figure 10.7.

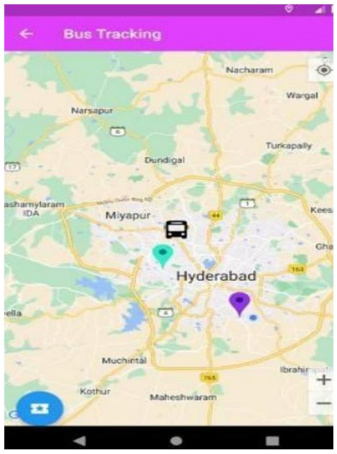


Figure 10.7 Real Time Bus Tracking

E. Bus Pass

A virtual bus pass is generated for each user who has cleared the bus fare. The status of the bus fee is communicated to the user after comparing the user details with the administration database. A note is popped up that discloses the status of the fare to the user as in Figure 10.8 (a) and Figure 10.8 (b).



Figure 10.8 (a) No fee dues



Figure 10.8 (b) Fee dues

Figure 10.8 Note stating the status of the bus fare

X. CONCLUSION

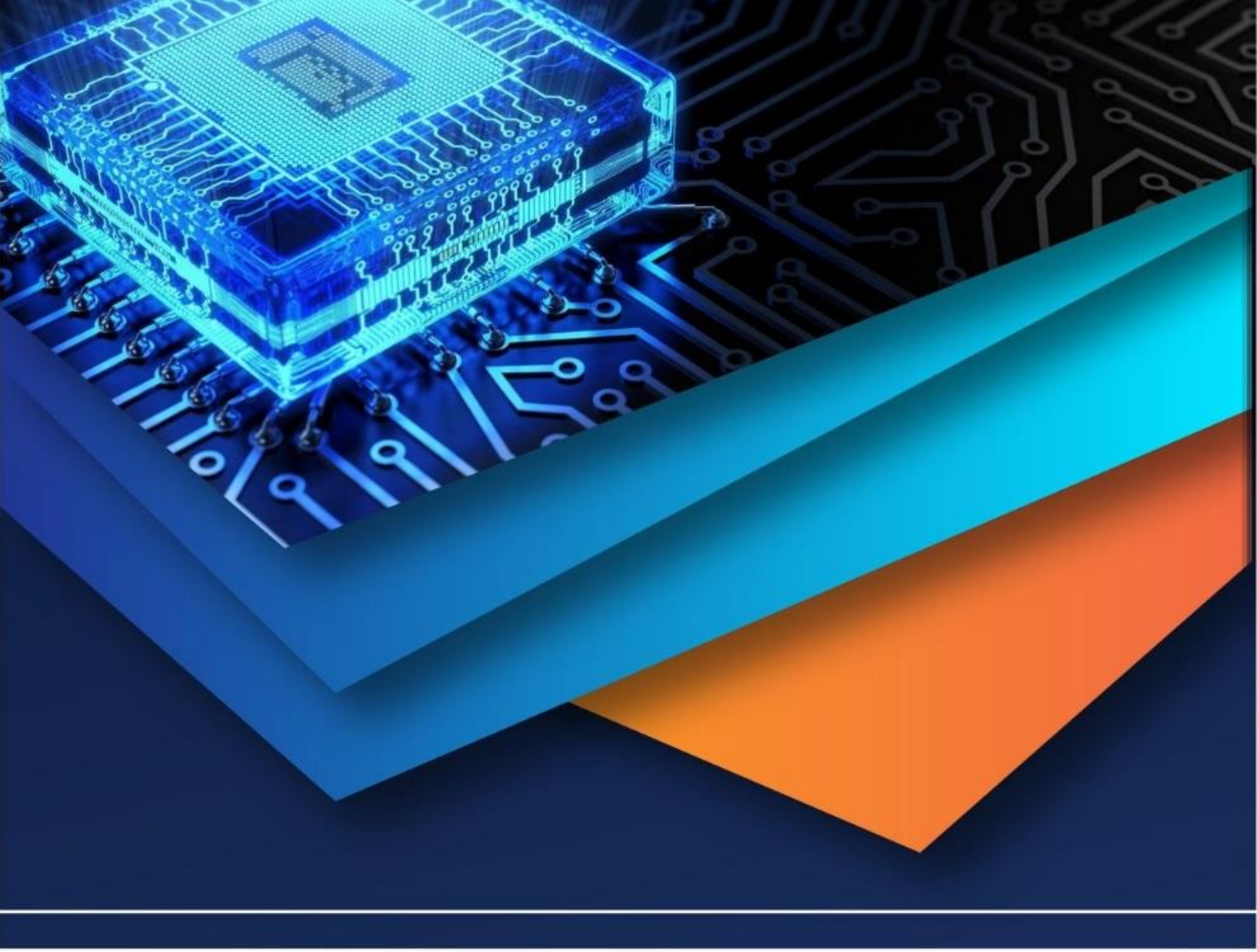
The Real-Time Bus Tracking Mobile Application aims to provide a solution for students who rely on bus transportation within the campus. With the increasing use of mobile applications by students for educational purposes, having an app that helps manage their trips to class is highly beneficial. The application requires an internet connection to access real-time information, allowing students to stay updated on bus schedules and plan their journeys accordingly. One of the key features of the app is the validation of student bus passes, enabling the management to ensure that only authorized students can avail of the bus services.

Additionally, the app serves as a platform for students to voice their concerns or make complaints, providing an avenue for communication between students and the management. The primary objective of the project is to address the transportation issues faced by students on campus. Further enhancements can be made to the system, such as incorporating the functionality to determine the total available seats on the bus in advance. There is also potential for expansion to other campus facilities like college canteens and libraries, allowing students to easily access various services through the app. To ensure widespread accessibility, the project will be deployed on a cloud platform, making it available to all mobile users. Further improvements can include displaying the estimated arrival time of the bus at the user's stop, along with the predetermined route of the bus.

The app can also send alert messages to users, notifying them when the bus is expected to arrive within the next five minutes. In situations where the user's usual bus is unavailable on a particular day, the app can provide information about alternate buses and their respective routes for the user's convenience.

REFERENCES

[1] Dr. Saylee Gharge, Manal Chhaya, Gaurav Chheda, Jitesh Deshpande, “Real time bus monitoring system using GPS,” An International Journal of Engineering Science and Technology, Vol.2, Issue3, June-2019.
[2] M. B. M. Kamel, “Real-time GPS/GPRS based vehicle tracking system”, International Journal Of Engineering And Computer Science, Aug-2020.
[3] “Real time vehicle monitoring and tracking system for school bus via Beagle bone,” International Journal of Science and Research (IJSR), vol. 5, no. 5, pp. 918–921, May2019.
[4] Abid Khan, Ravi Mishra, “GPS-GSM based tracking system,” International Journal of Engineering Trends and Technology, Vol.3, Issue 2, pp: 161-164, 2019.
[5] Muhammad Irsyad Kamil Riadz, Nurul Nazihah Jamal, Mohammad Shafri Amir Mohd Sharif, Mohd Sufi Mohd Suhaimi, 2017. A real-time bus tracking System, Undergraduate Project Report, Universiti Teknologi Malaysia: Faculty of Computing.
[6] Manini Kumbhar, Meghana Survase, Pratibha Mastud, Avdhut Salunke, 2016. Real-Time Web Based Bus Tracking System, International Research Journal of Engineering and Technology.
[7] S. Halim, 2016. Core Asset Variability Modeling Approach for Software Product Line Development (Doctoral dissertation). [15] S. N. M. Ekssan, 2017. Enhanced Educational Robotics Feature Model with Pedagogical Elements In Software Product Line, Universiti Teknologi Malaysia: Faculty of Computing.



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089 (24*7 Support on Whatsapp)