IEC
Institute of Electronics and Computer

Journal of Artificial Intelligence and Systems, 2024, 6, 188-200
https://iecscience.org/journals/AIS
ISSN Online: 2642-2859

Live Bus Tracking Using Greedy-Based Approach

Sanjukta Mohanty[1,*], Aditya Aswajeet Nayak[2], Prassannajeet Pradhan[3], Subrat Kumar Sahoo[4], Sonali Beura[5]

[1] Odisha University of Technology and Research, Computer Science and Engineering, Bhubaneswar, Odisha, and India
mailtorani.sanjukta@gmail.com
[2] Odisha University of Technology and Research, Computer Science and Engineering, Bhubaneswar, Odisha, and India
adityaaswajeetnayak@gmail.com
[3] Odisha University of Technology and Research, Computer Science and Engineering, Bhubaneswar, Odisha, and India
pratapprasannajeet2009@gmail.com
[4] Odisha University of Technology and Research, Computer Science and Engineering, Bhubaneswar, Odisha, and India
subratsahoo0796@gmail.com
[5] Odisha University of Technology and Research, Computer Science and Engineering, Bhubaneswar, Odisha, and India
saisonali75@gmail.com
*Corresponding Author: Sanjukta Mohanty, Email: mailtorani.sanjukta@gmail.com

How to cite this paper: Sanjukta Mohanty, Aditya Aswajeet Nayak, Prassannajeet Pradhan, Subrat Kumar Sahoo, Sonali Beura (2024). Live Bus Tracking Using Greedy-Based Approach. Journal of Artificial Intelligence and Systems, 6, 188–200. https://doi.org/10.33969/AIS.2024060113.

Received: March 16, 2024
Accepted: May 15, 2024
Published: May 17, 2024

Abstract

In the realm of modern transportation, the ability to access real-time information has revolutionized the way people navigate their daily commutes. This thesis delves into the realm of live tracking and its pivotal role in enhancing the public bus transportation experience. The core of this research centers around the development and implementation of a real-time bus tracking and arrival time estimation system. Leveraging cutting-edge technologies such as GPS, Google Maps API, and real-time analytics, this system provides users with dynamic, on-the-spot information about bus locations and estimated arrival times. The algorithms employed in this system are designed with inspiration from greedy algorithms, focusing on making locally optimal decisions to ensure real-time effectiveness. The user-centric web-based application seamlessly interfaces with the central control unit and driver's smartphones, offering a holistic solution for both end-users and administrators. This research unfolds the intricacies of heuristic algorithms in the context of real-time transportation systems, ultimately enhancing the efficiency and convenience of public bus services.

Keywords

Greedy Approach, Bus Tracking, GPS, Google Maps, Internet of Things.

1. Introduction

In the realm of college campuses, efficient bus transportation systems are indispensable, catering to the mobility needs of students. College bus services offer a cost-effective and essential mode of transport within the campus premises. However, a significant challenge faced by users of these services is the uncertainty surrounding bus schedules and arrival times. It is not uncommon for students to be waiting at bus stops without knowing when the next bus will arrive. Unexpected delays that arise from different sources, like heavy traffic or construction, exacerbate the issue even more. Here is where our suggested method comes into play. It aims to simplify college bus transit and give commuters important information regarding bus departures and arrivals, as well as real-time bus positions and anticipated wait times.

We have created a methodical tracking system that uses GPS technology to provide precise and instantaneous tracking results to meet this challenge. GPS technology offers the features required for efficient college bus tracking. Buses may communicate their positions to a central controller thanks to our system. Following an upload to a server, commuters can obtain this location data via an easy-to-use web application. Students can easily access this information from any location with internet connectivity, including their residences. Our system also has an integrated web application interface with Google Maps. End users can view all transmitted data through this interface, including the college bus's current location on a map. The web application makes sure the tracking data is updated regularly so that users always know where the bus is and when it's expected to arrive. By providing students and faculty members with a reliable way to track the college buses in real time. Users can make informed decisions about their travel, thereby optimizing their time and enhancing their overall college experience.

In summary, our system utilizes GPS technology to track college bus locations and transmit this information via wireless communication, ensuring that students have access to real-time data. By improving the efficiency of college bus transportation, our solution contributes to a smoother campus experience, allowing everyone to focus on their academic and extracurricular pursuits without the hassle of uncertain bus schedules.

2. Related Work

Various designs and implementations have been suggested for bus tracking systems. When it comes to putting these designs into practice or crafting the actual system, each of the proposed methods and implementations tends to have its distinct characteristics and features.

Karkare et al. developed a live tracking system that utilizes GPS technology to send location updates to users at a high frequency, providing real-time information [1]. This system, operational on Android platforms, serves multiple safety and security purposes, making it suitable for tracking the live locations of children or loved ones. By continuously relaying GPS signals, this system offers versatile applications for safety and security needs, providing peace of mind to users concerned about the well-being of their family members. In [2], the authors have developed a similar application that monitors employee locations using smartphone apps and GPS devices, either integrated into their personal phones or company vehicles. However,

this article outlines distinct guidelines for employers when employing location tracking for two primary purposes: individual employee misconduct investigations and non-investigatory group tracking. Specifically, for individual misconduct investigations, the article recommends GPS tracking be employed only to substantiate instances of serious employee misconduct. Shibghatullah et al. developed a vehicle tracking application tailored for monitoring buses, trains, and taxis [3]. One key challenge in vehicle tracking systems is accurately estimating arrival times. Leveraging technology, this paper focused on enhancing the precision of estimated arrival times, enabling users to anticipate vehicle arrivals more reliably. The proposed application utilizes GPS on Android smartphones to determine vehicle locations, storing coordinates in a Firebase Real-time Database for real-time data synchronization. Moreover, a real-time bus tracking system for students has been designed by Hasan et. al., which relies on GPS technology to track and locate buses in real-time [4]. This Android application helps students determine the precise location of buses, ensuring they arrive at the bus stop neither too early nor too late. It provides additional information such as bus details, driver information, contact numbers, and routes. This application offers a valuable time-saving solution, keeping students informed about their bus's real-time location, updated with latitude and longitude coordinates on Google Maps. Jain et al. researched the development of a precise and efficient GPS tracking application for school buses [5]. The primary aim is to address the issue of students waiting for buses and the lack of information for those who have missed their rides. The study comprises three main components. First, satellite data is acquired, refined through the Kalman filter, and transmitted to dedicated web servers via cellular networks. In the second part, coordinates are processed using the Google Maps API. The limitation of this approach is that an external GPS device is embedded in the vehicle to track the location. In [6], the authors have developed a model to automate public transport services, offering real-time tracking via RFID-tagged buses and readers at stops. An Arduino serves as the central controller, with GSM modules delivering tracking updates. GPS provides bus locations, and users receive notifications through IoT, while RFID reader inputs are processed by Arduino. The processed data is transmitted to the cloud, facilitating user-system interaction. Gull et. al developed a model which empowers administrators to add new bus drivers and students to the driver list [7]. It generates unique QR codes for students, featuring their details. Additionally, the system tracks bus locations via the driver's mobile device. Parents can access a real-time map displaying the current bus position, with frequent updates for safety. Various notifications are implemented, such as alerts for schedule deviations, and parents receive notifications for any changes in the daily bus schedule, ensuring efficient school bus management and safety. Shabli et. al proposed a campus bus tracking system, utilizing IoT technology with LoRa technology to address intermittent internet connectivity on campus [8]. The paper primarily concentrates on hardware and user testing. According to the survey, users expressed satisfaction with the system's performance, usability, and effectiveness. Sobhana et. al designed a model in which the school bus's location is tracked through the Global Positioning System (GPS) [9]. It uses Java language and relies on Firebase as a database to enable real-time location synchronization with registered parents. Parents receive advance notifications when their child crosses a previous bus stop, along with the address of that stop, as well as updates when their child arrives at a specific drop-off location. Moreover, a real-time bus tracking system for students has been designed by Vincent et. al using ReactJS and GPS technology, that tracks school bus locations, provides

attendance reports, estimated arrival times, and live video monitoring for student safety [10]. It also allows route and database management. Accessible through web browsers and Android apps, the system comprises three key modules: Server for location monitoring, Bus Unit for real-time tracking and attendance recording, and End User Application for user interfaces.

Table 1. Summary of Literature Review

References	Techniques Used	Key Findings	Performance
[1]	IoT-based GPS to track location,	The application continuously relays GPS data to track the location of children,	Sends data accurately and continuously.
[2]	GPS device embedded in the smartphone.	Tracks the location of employees through a GPS device in a smartphone.	It gives accurate data as a GPS device is used.
[3]	Firebase server, google distance matrix API, MySQL database, PHP.	It allows for tracking the real-time location of the vehicle and shows the departure time	Both driver and user need to be connected to the internet.
[4]	Firebase server, Google Maps, android platform, GPS technology.	Shows the real-time location of the bus by using only a smartphone and server.	Cost-effective and efficient.
[5]	Java framework, PHP, Python framework, Google Maps API.	It's an Android-based application that provides an estimated time for the client.	It can also track the lost vehicle.
[6]	IoT application, RFID, ARDUINO, GSM.	It's an IoT-based bus tracking system that can locate and track the buses.	Users can track the buses easily via an Android application.
[7]	IoT Technology, QR	The system sends	It provides a

	code, Javascript.	notifications to the parents about the status of the trip and their kid's location.	QR code, an SOS feature in the case of any emergencies during the bus trip.
[8]	IoT using LoRa technology.	It enables students to observe the available buses, routes, and locations of buses.	It uses IoT to make the system more efficient.
[9]	Java, Firebase DB, GPS, RFID.	Parents can get the location of their child through SMS.	It's more efficient as it also uses SMS.
[10]	HTML, CSS, JavaScript, GPS, GSM, Google Firestore DB, REST API.	College administration, parents, and students can easily track the location of the bus.	Positional accuracy of 2.5m.

3. Overview of the Proposed Model

This section provides a detailed description of the proposed methodology to track the live location of buses. The proposed methodology consists of three components driver module, admin module, and client module.

3.1. Problem Statement

The management of college buses presents a significant contemporary challenge. The existing system lacks an efficient solution for offering comprehensive information regarding college buses, including their estimated arrival times, expected waiting durations, and real-time bus location updates.

3.2. Solution

Our college bus information system offers users comprehensive details on all available bus routes, including real-time location updates. The system primarily relies on GPS technology, which is integrated into the driver's mobile phone. The GPS in the driver's phone receives satellite signals and interfaces with the Google Maps API to pinpoint the exact latitude and longitude coordinates. This location data is then transmitted via Google's infrastructure, typically using wireless communication systems like GSM/GPRS.

The system is designed to provide remote users with access to real-time bus location information based on their source and destination preferences. Our proposed system's

ability to provide precise, up-to-the-minute bus location data is particularly valuable for efficient bus tracking and monitoring.

3.3. Architecture of Proposed System

The proposed model comprises a multifaceted approach, skillfully interweaving the functionalities of three core modules Central Control Unit (Admin Module), Bus Module, and Client Side Application (User Module) to create a robust and efficient framework for optimizing college bus transportation services.

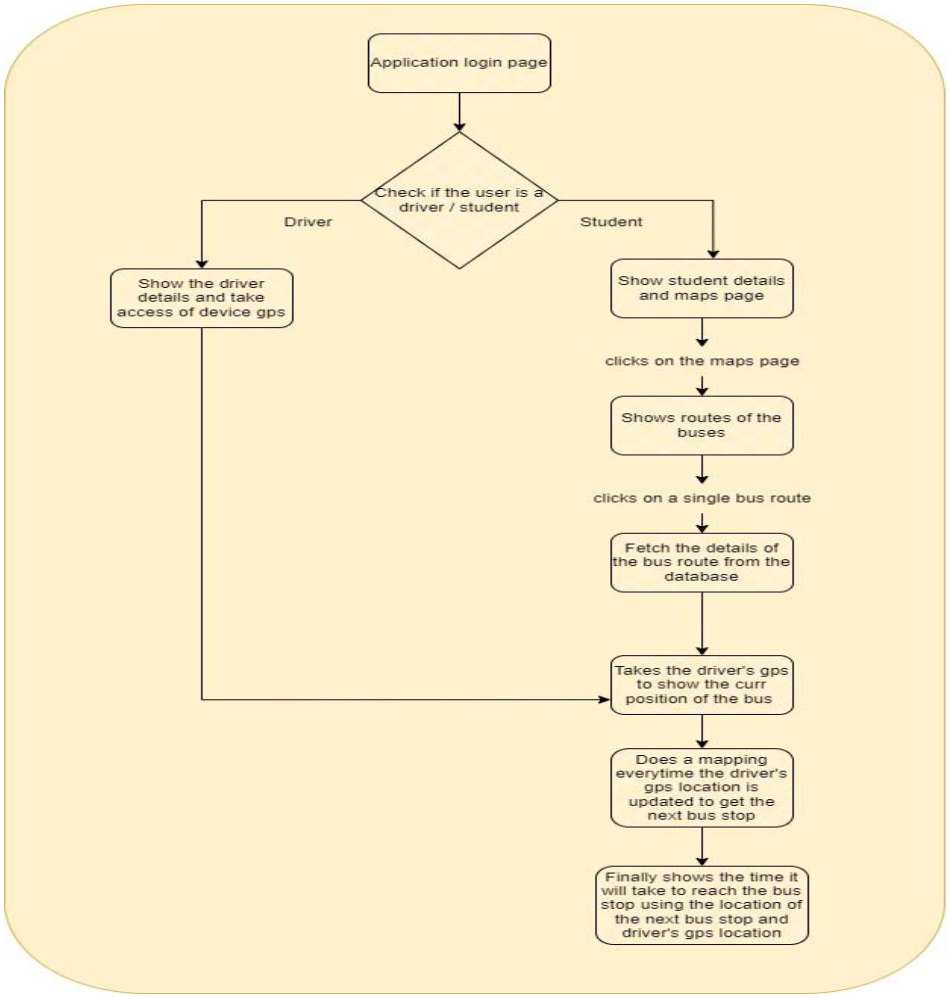


Figure 1. Flow of the Application.

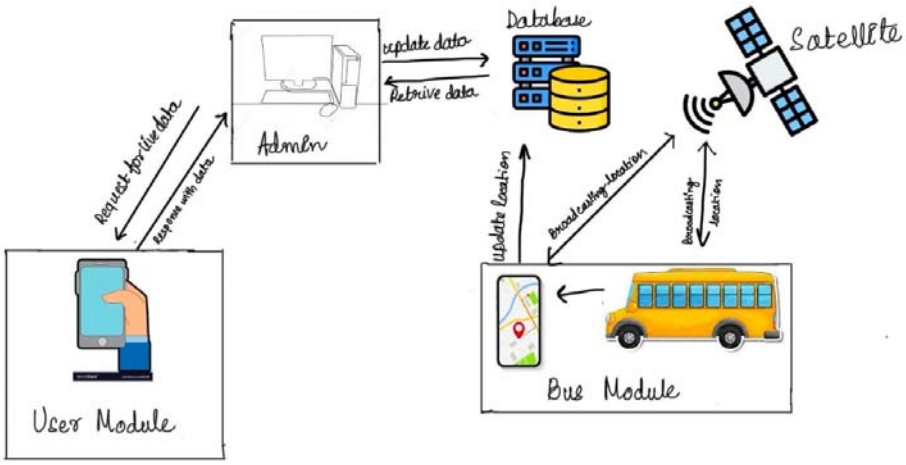


Figure 2. Architecture of the proposed system

3.3.1 Central Control Unit (Admin Module)

At the heart of the system is the Central Control Unit, serving as the administrative core [11]. This module takes on a pivotal role in orchestrating and overseeing the entire system. Its responsibilities encompass an array of critical functions, including user management, dynamic bus route updates, and seamless addition of new routes to the central database. It acts as the vital link between administrative activities and end-user interactions, ensuring the system's smooth operation. The Central Control Unit, by effectively managing user credentials and route information, lays the groundwork for the entire system's functionality. It acts as the gatekeeper for user authentication and authorization while simultaneously providing a central repository for all system-related data.

3.3.2 Bus Module

The Bus Module harnesses the power of advanced technology to offer a precise and reliable bus-tracking solution [12]. It involves the use of specialized devices that as the smartphone of the user and GPS transceivers within the smartphones, creating a comprehensive tracking infrastructure. These devices operate seamlessly in diverse weather conditions and geographical locations, guaranteeing 24/7 tracking capabilities. Also, another feature of these is they can track location even if the smartphone is not connected to the internet. One of the notable features of this module is the absence of subscription fees or setup charges for GPS usage. The GPS receivers installed in the mobile phone capture signals from at least three satellites, resulting in highly accurate position calculations. Depending on the specific application, these GPS transceivers can function as Data Loggers, Data Pullers, or Data Pushers. Data received from the buses is continuously transmitted to the central server. This data, which includes real-time bus coordinates, latitude, longitude, timestamps, and more, is processed and securely stored in the central repository. Each bus is uniquely identified through its bus license card number and the track they are going on, ensuring precise tracking. This module plays a critical role in real-time tracking,

enabling users to access up-to-the-minute bus locations and ensuring that the central server is continuously fed with live data for further analysis and distribution.

3.3.3 Client-Side Application (User Module)

The User Module offers an intuitive, web-based interface for remote users, facilitating their interactions with the system [13]. Users gain access by logging in using their unique credentials, granting them personalized access to the system. Upon successful login, they are directed to their dashboard, where they can view and manage their account details. One of the most interactive features of this module is the Maps page, which integrates the power of Google Maps. Here, users can browse a list of available bus routes, each associated with specific bus numbers. When users select a particular bus number to view route details, the system initiates a request to the Central Control Unit. The response is structured as an array of objects, each containing three key properties:

1. "stop," indicating the stop number on the bus route

2."long," representing the longitude of the stop

3." lat," displaying the latitude of the stop.

The object looks like -

```javascript
[{ stop: 1, long: 12.13, lat: 11.12 },
{ stop: 2, long: 13.13, lat: 12.12 },
{ stop: 3, long: 14.13, lat: 13.12 }]
```

This response is then seamlessly integrated into the client-side application. To ensure that users receive continuous real-time updates on the bus's current location, a socket connection is established, delivering a constant stream of data from the Bus Module. This real-time data, denoted as "X," provides users with the most current information about the bus's whereabouts.

Algorithms used in the setup

The algorithms provided in this context are inspired by the concept of greedy algorithms. Greedy algorithms make locally optimal choices at each step, aiming to find a global solution. The provided algorithms use a similar approach by making local decisions for real-time bus tracking and stop identification.

Algorithm 1: Determine Next Bus Stop

Input:

- X - Current location of the bus.

- Bus_Route - Array representing the bus route with stops, where each stop is an object with properties stop, long, and lat.

- Google_Maps_API - The API used to calculate distances.

Output:

- Next_Stop - The next bus stop.

1. Initialize Next_Stop as null.

2. For i from 0 to Bus_Route.length - 2 (loop through the stops except the last one):

a. Calculate d1 using Google_Maps_API as the distance between (X.long, X.lat) and (Bus_Route[i].long, Bus_Route[i].lat).

b. Calculate d2 using Google_Maps_API as the distance between (X.long, X.lat) and (Bus_Route[i + 1].long, Bus_Route[i + 1].lat).

c. Calculate d using Google_Maps_API as the distance between (Bus_Route[i].long, Bus_Route[i].lat) and (Bus_Route[i + 1].long, Bus_Route[i + 1].lat).

d. If d1 + d2 ≈ d (approximately equal):

- Set Next_Stop to Bus_Route[i + 1].

- Exit the loop.

3. If Next_Stop is still null after the loop:

a. Set Next_Stop to the last stop in the Bus_Route (end of the route).

Output:

- Next_Stop now contains the identified next bus stop based on the distances calculated.

Algorithm 2: Real-Time Bus Location and Estimated Time of Arrival

Input:

- Bus_Route - Array representing the bus route with stops, where each stop is an object with properties stop, long, and lat.

- Google_Maps_API - The API used to calculate distances and speed.

- Bus_Speed - The current speed of the bus.

- Next_Stop - The next bus stop.

Output:

- Live_Location - The live location of the bus.

- Estimated_Arrival_Time - The estimated time of arrival at the next stop.

- Scheduled_Arrival_Time - The scheduled time of arrival at the next stop.

1. Obtain the live location Live_Location of the bus using the Google Maps API.

2. Determine the current speed Bus_Speed of the bus.

3. Calculate the remaining distance Remaining_Distance from the Live_Location to the Next_Stop using the Google Maps API.

4. Calculate the estimated time of arrival Estimated_Arrival_Time as follows:

a. Calculate the time Time_To_Arrival by dividing Remaining_Distance by Bus_Speed.

b. Add Time_To_Arrival to the current time to get the estimated time of arrival.

5. Retrieve the scheduled arrival time Scheduled_Arrival_Time for the bus at the Next_Stop from the bus schedule or database.

6. Return Live_Location, Estimated_Arrival_Time, and Scheduled_Arrival_Time as the output.

Output:

- Live_Location represents the real-time location of the bus.

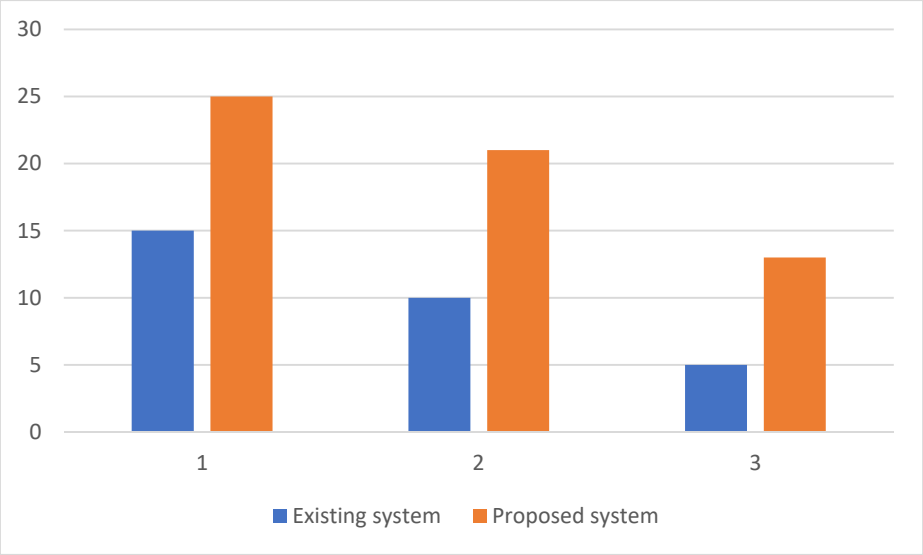
- Estimated_Arrival_Time provides the estimated time of arrival at the next stop based on the live location and speed.

- Scheduled_Arrival_Time indicates the scheduled time of arrival at the next stop as per the bus schedule.

This comprehensive methodology underscores the intricate workings of our multi-module framework, demonstrating how it synergizes the roles of each module to create an innovative system for bus transportation optimization. It addresses user management, bus route updates, real-time location tracking, and interactive access to vital information, marking a significant step towards enhancing the overall bus transportation experience.

4. Result Analysis

The proposed system can give better performance than the existing system based on the number of users trying to access the results. The y-axis here is the performance index. The x-axis here is several users * 1000.



5. Conclusion & Future Scope

Our system significantly reduces the waiting time for remote bus users, offering a more efficient and user-centric public transportation experience. The system's real-time bus tracking feature is a key asset, allowing users to monitor bus locations at any given time and place, providing unparalleled accessibility and convenience. Current bus-related data is stored securely on a centralized server and is readily accessible to remote users via a web-based application, ensuring that they have access to up-to-date information. The user-friendly interface, featuring visual representations on Google Maps, simplifies the process of obtaining essential information. Moreover, the open accessibility of the web-based application allows users to access it from any device with an internet connection, accommodating a wide range of user preferences. In addition to these advantages, the system empowers users to make dynamic decisions about their journeys by offering real-time information on bus availability, enabling them to either wait or reschedule their travel plans as needed. In summary, the system's successful implementation has significantly improved the user experience, offering real-time data and interactivity that aids in making informed decisions about bus travel.

Acknowledgments

We show our gratitude towards the authors for their genuine efforts and assistance in carrying out this research project.

Conflicts of Interest

The authors declared that they have no conflict of interest to report regarding the present study.

References

[1] Ankur Ganorkar , Sagar Karkare, Ashwini Andhale, Pranali Rokade, Santosh Bansode, 2020, Live Tracking System, INTERNATIONAL JOURNAL OF ENGINEERING RESEARCH & TECHNOLOGY (IJERT) Volume 09, Issue 06 (June 2020).

[2] Marc Chase McAliister, GPS and Cell Phone Tracking of Employees, 70 Fla. L. Rev. 1265 (2019). Available at: https://scholarship.law.ufl.edu/flr/vol70/iss6/3.

[3] Abdul S. Shibghatullah, Abdurrahman Jalil, Mohd H. Abd Wahab, Joseph Ng Poh Soon, Kasthuri Subaramaniam, and Tillal Eldabi, "Vehicle Tracking Application Based on Real Time Traffic," International Journal of Electrical and Electronic Engineering & Telecommunications, Vol. 11, No. 1, pp. 67-73, January 2022. Doi: 10.18178/ijeetc.11.1.67-73.

[4] M. N. Hasan and M. Sharif Hossen, "Development of An Android Based Real Time Bus Tracking System," 2019 1st International Conference on Advances in Science, Engineering and Robotics Technology (ICASERT), Dhaka, Bangladesh, 2019, pp. 1-5, doi: 10.1109/ICASERT.2019.8934621.

[5] S. Jain, A. Trivedi and S. Sharma, "Application Based Bus Tracking System," 2019 International Conference on Machine Learning, Big Data, Cloud and Parallel Computing (COMITCon), Faridabad, India, 2019, pp. 152-154, doi: 10.1109/COMITCon.2019.8862254.

[6] A. Deebika Shree, J. Anusuya and S. Malathy, "Real Time Bus Tracking and Location Updation System," 2019 5th International Conference on Advanced Computing & Communication Systems (ICACCS), Coimbatore, India, 2019, pp. 242-245, doi: 10.1109/ICACCS.2019.8728353.

[7] H. Gull, D. Aljohar, R. Alutaibi, D. Alqahtani, M. Alarfaj and R. Alqahtani, "Smart School Bus Tracking: Requirements and Design of an IoT based School Bus Tracking System," 2021 5th International Conference on Trends in Electronics and Informatics (ICOEI), Tirunelveli, India, 2021, pp. 388-394, doi: 10.1109/ICOEI51242.2021.9452818.

[8] Ahmad Hanis Mohd Shabli, Mawarny Md. Rejab, Nur Hani Zulkifli Abai, and Suwannit Chareen Chit. 2023. Campus Bus Tracking System using LoRa Technology: IoT System Testing. In Proceedings of the 2023 12th International Conference on Software and Computer Applications (ICSCA '23). Association for Computing Machinery, New York, NY, USA, 234–239. https://doi.org/10.1145/3587828.3587863.

[9] M. Sobhana, T. R. Chowdary, M. G. S. S. Venkatesh and K. S. Devendra, "Smart Campus Bus Tracking Alert System Using Real-Time GPS," 2023 9th International Conference on Advanced Computing and Communication Systems (ICACCS), Coimbatore, India, 2023, pp. 1777-1781, doi: 10.1109/ICACCS57279.2023.10112757.

[10] B. Vincent, J. Sabu, C. Mathew, S. S. Nair, S. B. George and S. D, "Live College Bus Tracking and Route Mapping Using Internet of Things," 2023 2nd International Conference on Computational Systems and Communication (ICCSC), Thiruvananthapuram, India, 2023, pp. 1-7, doi: 10.1109/ICCSC56913.2023.10143028.

[11] [online] Available:
https://www.fleetroot.com/blog/how-does-real-time-bus-tracking-work.
[12] [online] Available:
https://www.armia.com/blog/how-to-build-a-school-bus-tracking-system.
[13] [online] Available:
https://www.busboss.com/blog/topic/school-bus-tracking-app.