

# Areal Time Bus Monitoring System using Mobile Location

**Sathishkumar T M, Magibalan R, Mohanprasath A, Ragul T, Selva Praveen S**

*Associate. Prof. in Department of ECE, KSRCE TIRUNCHENGODE, TamilNadu, India*

*Student in Department of ECE KSRCE TIRUCHENGODE, Tamil Nadu, India*

**Abstract** -Using Public Transportation Networks (PTNs) might be challenging for users who are unfamiliar with the destination location. This is accurate for both infrequent and recurring customers who must travel to unfamiliar places. While using public transit in these circumstances, having sufficient on-trip navigational information can be quite helpful. By improving the use of public transportation through effective transit traveler information systems, we can assist current users and inspire new users. For use not just on the device itself but also for distant apps that need to track the device. When using the app to plan their route and choose when to board the bus, users have flexibility. The suggested technique is straightforward to use and guarantees minimal upkeep.

**Keywords**—GlobalPositioningSystem(GPS);StudentMonitoring;AndroidStudio;BusTracking; Hypertext Markup Language (HTML);CascadingStyleSheets(CSS);jQuery;Laravel(PHPframework);MySQLDatabase;JAVA;ExtensibleMarkupLanguage(XML);GoogleMapsAPI.

## I. INTRODUCTION

Bus service is the most popular mode of public transit out of all of them. Traffic jams, high parking costs, and a lack of parking spaces at the destination are just a few of the many reasons individuals choose to take the bus over driving their own vehicle. Unfortunately, the current transportation information system for bus transit is severely inadequate. Bus passengers only know the scheduled arrival time for their bus, not the precise arrival time. The bus transportation system does not have a proper mechanism to track all buses' positions and the real arrival times at each bus stop, unlike the train or flight transportation systems. These issues arise technology to monitor every bus on the road and does not have a platform for updating users on the most recent bus traffic information. Towards a solution. Bus timing and locating systems must be developed and put into use to address these issues and improve the current bus service system. Bus service is the most popular mode of public transit out of all of them.

### I. LITERATURE SURVEY

#### a. STRUCTURED QUERY LANGUAGE (SQL) DATABASE

*Author Name:* ElGouhary, Amany, RichardWells–Richard andAnthonyThatcher (2006).

The information from the Global Positioning System (GPS) devices is sent to web servers to be received, interpreted, and then saved in a MySQL database. This will be used to offer a dynamic web page that will take information about the user's present position and previous position from a database and overlay those values on an appropriate map from an online database.

#### a. APPLICATION DEVELOPMENT

*Author Name:* ManiniKumbhar, MeghanaSurvase,PratibhaMavdhutSalunk (2016).

The bus may be followed at any time or place using a system. The server stores all of the most recent data, which remote users can access through web-based software. This method is web-based, but as Android applications are more portable and smart phones are more extensively used nowadays, they are more preferred by users. A user would find it difficult to regularly utilise a web-based system while standing at a bus stop waiting for a bus.

#### b. GOOGLE MAPS API (APPLICATION PROGRAMMING INTERFACE)

*Author Name:* Md.MarufiRahman, JannatulRobaiatMou, Kusum Tara, Md.Ismail Sarkar (2016).

Using GSM and coordinates given by mobile devices, a real-time Google Map and location-based vehicle tracking system are shown on Google Maps. For various features of its service, the Google Maps Platform provides a number of APIs. To name a few, there is a Maps Static API for straightforward. Google Map embeds, a Maps JavaScript API for interactive and configurable maps, a Locations API for accessing information about sites of interest, and a Directions API for providing driving directions.

#### c. ALERTING SYSTEM

*Author Name:* P.A. Shinde and Y.B. Mane (2015).

A piece of technology that helps with school bus issues; it keeps track of the bus's whereabouts and alerts the driver if it decides to take a wrong turn. If the bus travels faster than the allowed speed, the system sends a warning message.

a. *PROBLEMSTATEMENT*

Bus travel is a fantastic approach to introduce the advantages of carpooling because it has decreased traffic on the roads. Also, it offers low-cost transportation options for families with modest incomes to go to other locations. The biggest disadvantage of taking a bus is the unpredictable arrival time, which may be caused by unanticipated events. There are a lot of reasons why a bus could not come when we expect it to, even though we are well familiar with the timetable.

- *Traffic jams-* When vehicles are fully stopped for periods of time, this is known as a traffic. Traffic congestion can lead to drivers becoming frustrated and engaging in road rage.
- *Severe rain-* As a matter of fact, rain often causes severe traffic congestion, especially when there is a heavy downpour.
- *Bus malfunctions-* The vehicle stops outside of a designated stop, either in the middle of the road or on the roadside with a (potentially) discernible cause. This could occur due to congestion, roadwork, road damage, accidents involving other traffic participants, objects on the road that cannot be easily evaded, and similar situations. While people wait at the bus stop, a dangerous scenario exists. They have to wait a long time at the bus stop because the bus is late or doesn't arrive. At the same time, the individuals might end up as robbery victims. Based on the typical travel time between two bus stops, the estimated arrival time is determined.

II. PROPOSEDSYSTEM

Buses can be tracked in real time, and the results of this tracking are subsequently provided to remote users who request this information. The pertinent information is provided by our system.

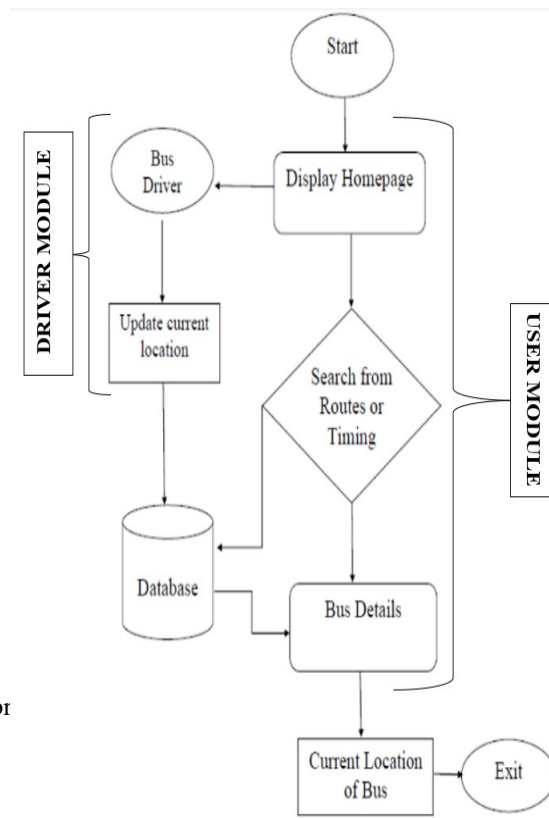
- Real-time location: This will show the bus's exact location at the moment.
- Route information: The system will provide the route information along with the stations' names.
- Driver's Contact Information: The system will show the driver's name along with their phone number.
- Average wait time and anticipated arrival time: The system will provide the ETA (estimated time of arrival) between two stations on Google Maps.
- Real-time traffic to many routes in the event of excessive congestion: The system will show the driver traffic information in the event of heavy congestion.

The suggested technique enables the user to determine the precise location of the bus from their current location. The user interface displays the bus routes so that users may choose the one they want to take. In the Google map, the bus's location is shown. The suggested method offers the following benefits:

A Google map showing the exact position is provided. Each person has access to the bus's details. This also enhances security because it is continually known where the bus is.

a. *FLOWCHART*

Fig1. Flowchart for our overall operation



- *USER INTERFACE DESIGN*

In order for bus users to conveniently access information, the interface design for bus information must be as straightforward as feasible. The programme is broken down into two primary sections: a map with bus location and congestion information.

The first phase involves pulling data from the API, the second involves using Google Maps to navigate through traffic, and the third involves collecting the current position and sending SMS. These two classes, MainActivity.java and Routes.java, carry out the fundamental functions of the entire system.

The user module is the first module in the application. Two buttons, one for bus 1 and the other for bus 2, are present on the user module. The corresponding buttons can be clicked by users. When they do, a red marker showing the latitude and longitude of the precise location of the bus as well as the estimated arrival time appear (ETA). The app also uses a blue circle to denote the user's location. The name of the place is also displayed. The precise distance between the user and the bus is displayed when the user clicks on the instructions button.

The driver module comes in second. Moreover, this module features two buttons that match the two paths. The driver can access traffic information for his travels by clicking on the corresponding routes. Blue signifies no traffic, orange indicates medium traffic, and red indicates strong traffic.

A navigation drawer with three options is also included in the app, including Route map, Know your driver, and Feedback. The stops along each route are listed on the route map. For this, there are two radio buttons included. The Know Your Driver option can be selected by the user if he wishes to learn any details about the driver.

- *CONCEPT FOR USER CONNECTION*

Bus users need to be able to access information simply, thus the interface design for bus information must be as straightforward as feasible. The application is separated into two primary sections: a map with bus location information and details on traffic congestion. The first section involves extracting data from the API, followed by traffic navigation using Google Maps and, finally, by obtaining the current location and sending SMS. These two classes, MainActivity.java and Routes.java, carry out the system's fundamental functions. The user module is the app's first module. Two separate and the other is for bus 2. On the corresponding buttons, users can click. In the shape of a red marker and the Expected Arrival Time, users can click this to obtain the latitude and longitude details about the precise location of the bus (ETA). The user's current position is also highlighted in the app as a blue circle. Moreover, the place's name is shown. The precise distance between the user and the bus is displayed when the directions button is clicked.

The driver module is the second module. Two buttons that correlate to the two routes are also present on this module. The driver can view traffic information for his journeys by selecting the appropriate routes. Blue signifies no traffic, while red and orange indicate high and medium traffic, respectively.

In addition, the app offers a navigation drawer with three options: Route map, Know your driver, and Feedback. Each stop along the designated itinerary is listed on the route map. For this, there are two radio buttons that have been added. The Know Your Driver option allows the user to get any information he wants to know about the driver.

- *MONITORING SERVER*

The database server and the web server are the two primary software components hosted by the tracking server. The user and the on-board unit continue to communicate with each other through the server. Using a web-based application on an independent platform, it gives a high-level application. The physical computer that serves as both a web server and a database server is the one where both servers are physically located.

The database server and the web server are the two primary software components hosted by the tracking server. The user and the on-board unit continue to communicate with each other through the server. Using a web-based application on an independent platform, it gives a high-level application. The physical machine that serves as both a web server and a database server is the one where both servers are physically located.

Approaching the drop-off/pick-up zone for home

With the use of this service, parents can be alerted before the bus arrives at their kids' usual stop. The bus's location is compared to the matching zones that are kept in the system database using this function.

The purpose of this service is to alert parents before the bus arrives at their children's regular stop. This function does a comparison between the bus's current position and the zones that correspond to it in the system database.

## Pipeline of Different Roles

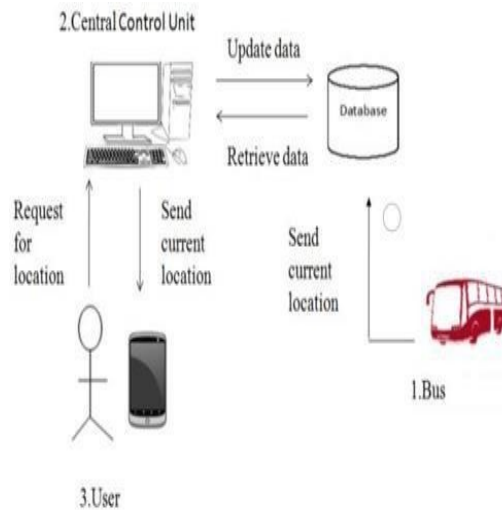


Fig2.Pipelineforouroveralloperation

**Bus User** - People is the main user of the proposed system because the main objective of bus tracking system is to provide estimated bus arrival time for people. People must able to retrieve real time estimated bus arrival time for every bus stop. While waiting in bus stop, people able to access bus tracking system with mobile device instead of using a computer. This is the major reason a mobile application for bus tracking was created. People may observe the bus position on a map thanks to the system's real-time bus tracking capability and mapping feature. By using this mapping technology, users may instantly determine the location of a bus based on a map.

**Bus Driver** - Bus drivers are the system's second most significant user. In order to update passengers on the current situation, the bus driver is able to do so. When driver can give permission in the app then only the students can view the current location of the bus. Because drivers are the system's second most significant user.

## f. IMPLEMENTATION

User Module, Driver Module, and Admin Module are the 3 modules we'll make. Both the web application and the Android app will be put into practice. Student module and driver module for Android application. Web application for Admin Module on the Server. We are utilizing a tool called Android Studio to make Android applications. The official Integrated Development Environment (IDE) for Google's Android operating system is called android studio.

- **Google Maps API** - A number of APIs (Application Programming Interfaces) are available for the Google Maps Platform's various services. On this system, the ETA (Estimated Time of Arrival) between two stations will show the average waiting time and anticipated time to arrive. A few of the available APIs are the Directions API for displaying directions to a location, the Locations API for accessing information about sites of interest, and the Maps Static API for straightforward Google Map embeds.
- **Frontend Development - HTML** - The coding that organizes a web page's content is called HTML (Hyper Text Markup Language). This method of linking documents is what has made the World Wide Web the international phenomenon that it is today.  
**CSS** - Frontend developers style HTML elements using CSS (Cascading Style Sheets), which stands for Cascading Style Sheets. CSS is used to manage the website's design and appearance.  
**JQUERY** - A JavaScript (JS) library with the tagline "Write Less, Do More," jQuery is free and open-source. Developers only need to call their function; jQuery will handle the rest, saving them from having to repeatedly write each job block!
- **Back-end Development - LARAVEL** - The framework Laravel is open-source. It has a fairly extensive feature set. It will save you a tone of time. It shields websites against the many threats that could occur. Laravel is an MVC framework with bundles, migrations, and Artisan CLI. Laravel offers a robust set of tools and an application architecture.  
**MySQL Database** - A separate programmed called a database is used to store a group of data. For creating, accessing, managing, searching, and duplicating the data it holds, each database has one or more unique APIs.
- **Mobile App Development - JAVA** - Applications for Android are created using the Java programming language. You can create amazing Android applications with the help of the many common Java libraries and

unique Android libraries. The Android SDK includes many standard Java as well as special Android libraries that will help you develop awesome Android applications.

*XML* - Extensible Markup Language is known as XML. Extraction of the GPS coordinates from the server is the first stage. It is a markup language like HTML, except there are no built-in tags. In order to extract the latitude and longitude from the XML, we use the XmlPullParser interface. XML Pull Parser is an interface that defines parsing functionality provided in XMLPULL V1 API.

g. *USER CASE DIAGRAM*

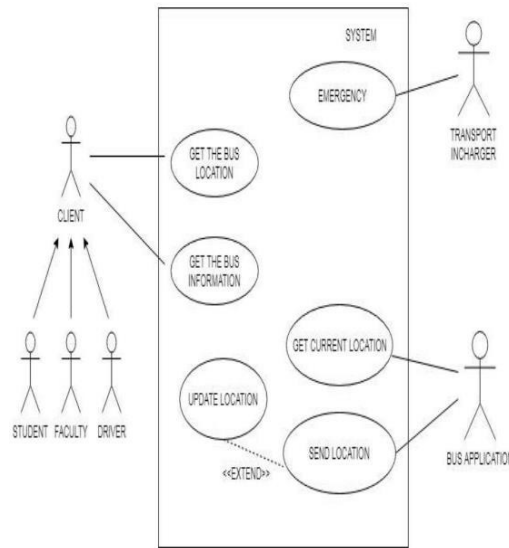


Fig.3.usercasediagram

*Actor* -User, Driver, Admin&Busapplication.

*Use-case* -Getthebuslocation, Getthebusinformation, Getthe currentlocation, Sendlocation, Updatelocation

*Pre-condition* -The bus driver's present location should be operational and send the coordinates to the server.

*Post-condition* -Bus position & time should be sent from server to mobile application.

h. *ACTIVITYDIAGRAM*

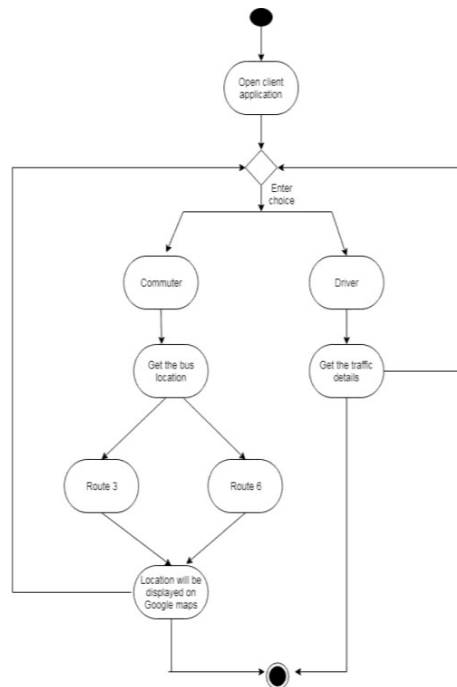


Fig.4.activitydiagram

**USER INTERFACE DESIGN** - The interface design of bus information must be as simple as possible so that bus users able to get information easily. The application is divided into two main parts, which is map with bus position & congestion details. Route map specifies each of the stops along the respective route. The first module in the app is the user module. The second module is the driver module. Two radio buttons have been incorporated for this purpose. If the user wants to get to know any information regarding the driver, he can click on the Know Your Driver option.

**LOCATION MANAGER** - The system location services are accessible through this class. These services allow apps to get updates on the device's position on a regular basis or to trigger an intent that they specify when the device gets close to a certain area.

**LOCATION LISTENER**:Used for receiving notifications from the Location Manager when the location has changed. These methods are called if the Location Listener has been registered with the location manager service using the request Location Updates(String, long, float, Location Listener) method.

The methods of Location Listener are:

- onLocationChanged: void onLocationChanged (Location location) Called when the location has changed.
- onProviderDisabled: void onProviderDisabled (String provider) Called when the provider is disabled by the user. If request Location Updates is called on an already disabled provider, this method is called immediately.
- onProviderEnabled: void onProviderEnabled (String provider) Called when the provider is enabled by the user.
- onStatusChanged: void onStatusChanged (String provider, int status, Bundle extras) Called when the provider status changes.
- getLatitude(): Get the latitude in degrees.
- getLongitude(): Get the longitude in degrees.

### III. RESULT

a. Web-Application

*Dashboard –*

This is the page that contains Total no. of buses, students, and drivers.



Fig5. Dashboard

*List of the user –*

This is the page that contains the no. of user's and their details. Here we can see the how many users were accessed.

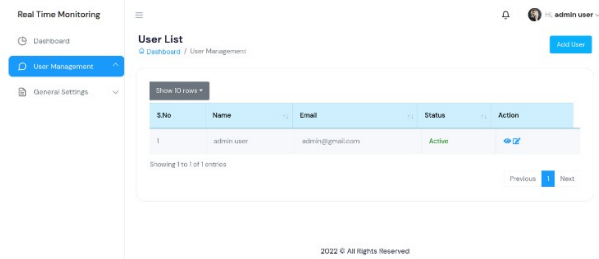


Fig 6. User list

*Login page –*

This is the page that contains the login details for admin & driver. This login page allows a user to gain access to an application by entering their username and password.

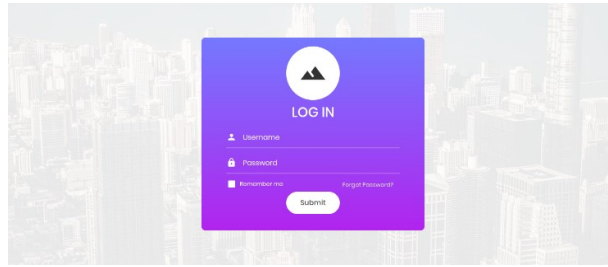


Fig 7. Login page

*Forgotpassword –*

This page includes instructions for replacing your current password with a new one. It allows users who have forgotten their password to unlock, retrieve, or reset it, usually by answering account security questions or sending them an e-mail.

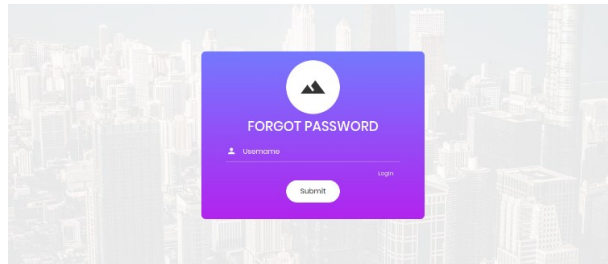


Fig8.Forgotpassword page

*UserAdd –*

This is the page that contains the details for adding users. Here we can add the new user and allow to access the portal.

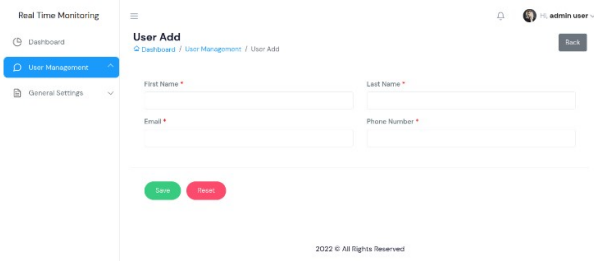


Fig 9. User Add

*Userdetails –*

This is the page that contains the details of the users. Here we can show the details of the user.

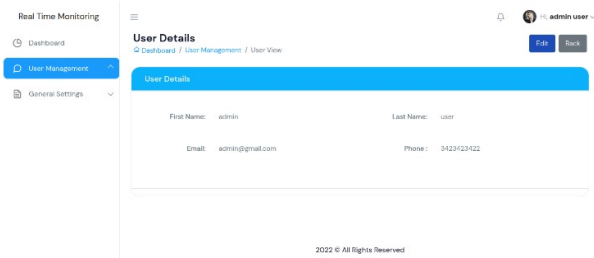


Fig.10. User details

*Bus details –*

The information on the buses is on this page. Here, we may display information on the buses and the drivers who will be driving them.

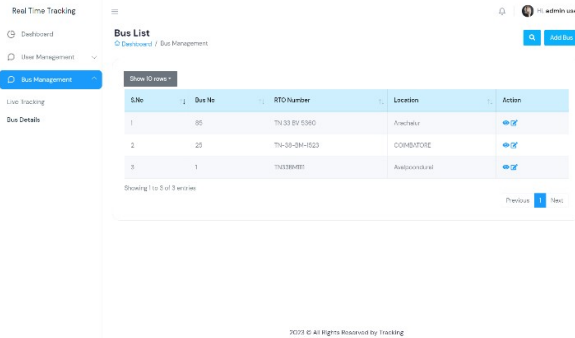


Fig.11.Busdetails

*Live tracking –*

Here, the administrator may use Google Maps to see the bus’s present location. When an administrator clicks on the bus icon, the bus number, registration number, and driver’s number are displayed.

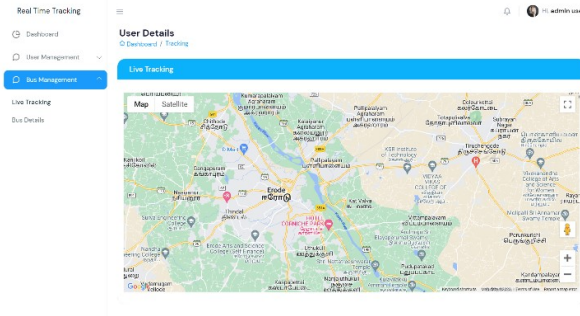


Fig.12. live tracking of the bu

b. MobileApplication

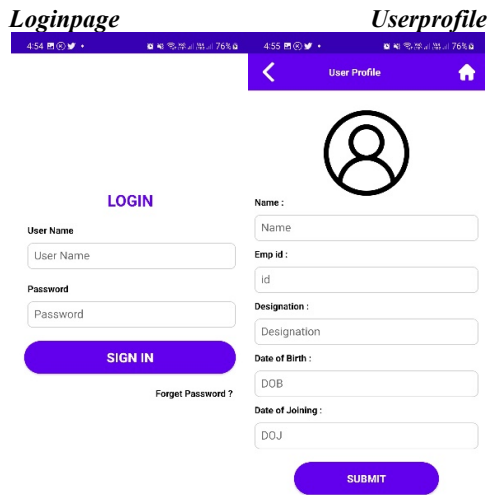


Fig 13.Login page

Fig14. Userprofile

*Routeselector*

*TrackingPage*



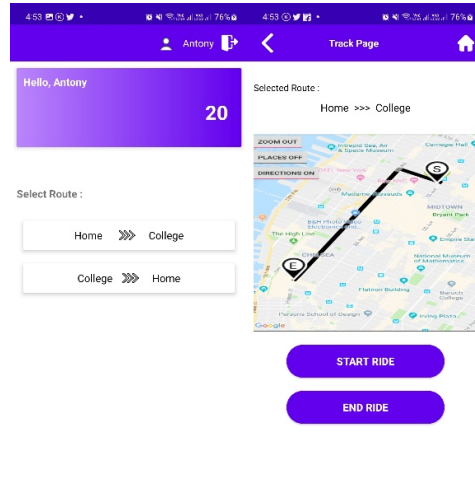


Fig 15. Routeselector

Fig 16. Trackingpage

### LIMITATIONS

If the mobile location feature is not enabled on the user's and driver's phone, our program will not function. The specifics of the bus breakdown are only sent to the admin. Bus delays are not communicated to passengers beforehand. The public won't receive any alerts regarding bus crossings from individual stops.

### IV. CONCLUSION & FUTURE ENHANCEMENT

People currently have a lot of problems because of traffic and other factors. One of the main issues is that people frequently miss their buses by a split second, which can cause problems like being late for appointments, tests, and other events. Bus Locating System was introduced to address this type of issue.

As of now we have implemented our project to track the bus of limited route. In future, it can be implemented on remaining every route. Our application is not so secured; it can be accessed by everyone. In future, the login details should be provided so that only the people who are travelling in bus and also who have paid the transportation fee can access the location of the bus. An alert message or notification should be sent to the user whenever the bus arrives to the stop or crosses the stop.

### REFERENCES

- [1] Abid khan, Ravi Sharma, "GPS-GSM based tracking system," International Journal of Engineering Trends and Technology, Vol.3, Issue2, and pp: 161-164, 2012.
- [2] Pankaj Verma, J.S.Bhatia, "Design and development of GPS-GSM based tracking system with Google map based monitoring," International Journal of Computer Science, Engineering and Applications, Vol.3, No.3, June 2013.
- [3] Nagarajan and M.Madheswaran - 'Experimental Study and steady state stability analysis of CLL-T Series Parallel Resonant Converter with Fuzzy controller using State Space Analysis' - Iranian Journal of Electrical & Electronic Engineering, Vol.8 (3), pp.259-267, September 2012.
- [4] G.Raja, D.Naveen Kumar, G.Dhanateja, G.V.Karthik, Y.Vijay Kumar, "Bus Position Monitoring system to facilitate the passenger," IEEE and Advanced technology (IJESAT), Volume -3, Issue-3 pp:132-135, 2014.
- [5] C.Nagarajan and M.Madheswaran - 'Stability Analysis of Series Parallel Resonant Converter with Fuzzy Logic Controller Using State Space Techniques' - Taylor & Francis, Electric Power Components and Systems, Vol.39 (8), pp.780-793, May 2011.
- [6] Shekhar Shinde, Vijay Kumar Nagalwar, Nikhil Shinde, B.V.Pawar, "Design of City Bus Tracking System," Int. journal of Engineering Research and application, ISSN: 2278 9622, Vol 4, Issue 4 (Version 9), pp: 114-117, April 2014.
- [7] C.Nagarajan and M.Madheswaran - 'Performance Analysis of LCL-T Resonant Converter with Fuzzy/PID Using State Space Analysis' - Springer, Electrical Engineering, Vol.93 (3), pp.167-178, September 2011.
- [8] Sabira Khanam, Maliha Mahbub, Anuradha Mandal, M. Shamim Kaiser and Shamim Al Mamun, "Improvement of RFID Tag Detection Using Smart Antenna For Tag Based School Monitoring System", International Conference on Electrical Engineering and Information & Communication Technology (ICEEICT) 2014.
- [9] C.Nagarajan and M.Madheswaran - 'Experimental verification and stability state space analysis of CLL-Series Parallel Resonant Converter' - Journal of ELECTRICAL ENGINEERING, Vol.63 (6), pp.365-372 Dec.2012.
- [10] G.Neelakrishnan, K.Anandhakumar, A.Prathap, S.Prakash "Performance Estimation of cascaded h-bridge MLI for HEV using SVPWM" Suraj Punj Journal for Multidisciplinary Research, 2021, Volume 11, Issue 4, pp:750-756