**A Project On**

**Bus tracking system using a mobile application**



A project paper is submitted to the **Faculty of Engineering & Technology** of **Islamic University** in partial fulfillment of the requirements for the Degree of B.Sc(Engg.) in **Information and Communication Technology**.

Under the Supervision of

**Dr. MD. ALAMGIR HOSSAIN**

Assistant Professor,

Department of Information & Communication Technology

Islamic University, Kushtia, Bangladesh

Submitted by

**MD Taufique A Elahi**

Roll No: 1718005, Reg. No: 1316

Session: B.Sc. (2017-18)

**DEPARTMENT OF INFORMATION AND COMMUNICATION TECHNOLOGY**

**ISLAMIC UNIVERSITY, KUSHTIA, BANGLADESH**

**Dedicated**

**To**

**My Beloved Parents**

**&**

**Teachers**

***CERTIFICATE***

I am pleased to certify that **Taufique A Elahi**, Examination Roll No-1718005, and Registration No-1316, has performed a project work **“Bus tracking system using a mobile application.”** under my supervision in the academic year 2017-18 for the fulfillment of the partial requirement for Degree of Bachelor of Science in Information and Communication Technology. So far as I concern this is an original project work that he carried out for four years in the Department of Information and Communication Technology, Islamic University, Kushtia, Bangladesh.

I strongly declare that this dissertation has not been copied from any other project or submitted to elsewhere prior to submission to this department.

Supervisor

**Dr. MD. ALAMGIR HOSSAIN**

Assistant Professor,

Department of Information and Communication Technology

Islamic University, Kushtia, Bangladesh.

***Acknowledgment***

At first, thanks to Almighty Allah, the creator, and sustainer who has given me strength and opportunity to complete the Project “Bus tracking application for mobile.”

Regarding the outcome of my project, I would like to thank my honorable supervisor **Dr. MD. ALAMGIR HOSSAIN**, Assistant Professor, Department of Information and Communication Technology, Islamic University, Kushtia, Bangladesh for his commented, guidance and kind help in many ways throughout the lifetime of this work.

I am also grateful to other teachers of the Department of Information and Communication Technology, Islamic University in making valuable suggestions and inspiration from time to time.

I would like to express the deepest appreciation for their co-operation and those who gave me the possibility to complete this project. Also, I will not forget friends, seniors, juniors and relatives who supported me in any respect during the completion of the project.

**Taufique A Elahi**

***Abstract***

The main aim of this Android application is to track the University buses of Islamic University in Kushtia-Jenaidha city which would give the exact location of buses with the help of Google map and help the users to plan their way to reach their university on time. This application may be greatly used by university students and staffs since Android mobiles has become common and spread everywhere. In addition, this will also enhance the security since the movement of the university buses is always available.

A GPS tracking device is a portable unit that allows users to monitor and track its location. These devices are most commonly used in vehicles as cars, buses tracking system. This project helps to locate the routes in which buses are travelling and displays the bus position in Google map. I will complete this project using Dart programming language with flutter framework.

*Contents*

**Certificate iii**

**Acknowledgment iv**

**Abstract v**

[Chapter 1 9](#_Toc137245806)

[Introduction 9](#_Toc137245807)

[1.1 General Introduction 9](#_Toc137245808)

[1.2 Objectives 10](#_Toc137245809)

[1.3 Motivation 10](#_Toc137245810)

[1.4 Project Overview 11](#_Toc137245811)

[Chapter 2 12](#_Toc137245812)

[Literature Review 12](#_Toc137245813)

[2.1 Transit Apps 12](#_Toc137245814)

[2.1.1 Problem statement 12](#_Toc137245815)

[2.2 Public Transportation Agencies 12](#_Toc137245816)

[2.2.1 Problem statement 12](#_Toc137245817)

[2.3 Research Projects 13](#_Toc137245818)

[2.4 IoT based Intelligent Bus Monitoring System 13](#_Toc137245819)

[2.4.1 Problem statement 13](#_Toc137245820)

[Chapter 3 15](#_Toc137245821)

[Methodology 15](#_Toc137245822)

[3.1 Software Development Life Cycle (SDLC) 15](#_Toc137245823)

[3.2 E-R diagram 17](#_Toc137245824)

[3.3 Class Diagram 18](#_Toc137245825)

[3.4 System Architecture 19](#_Toc137245826)

[Chapter 4 21](#_Toc137245827)

[Design And Implementation 21](#_Toc137245828)

[4.1 Overview 21](#_Toc137245829)

[4.2 Components 21](#_Toc137245830)

[4.2.1 Location Manger 21](#_Toc137245831)

[4.2.2 Fragments 21](#_Toc137245832)

[4.2.3 Async Task and services 22](#_Toc137245833)

[4.3 Block Diagram 23](#_Toc137245834)

[4.4 Data flow Diagram: 23](#_Toc137245835)

[4.5 Data Dictionary: 24](#_Toc137245836)

[4.7 Workflow Diagram 26](#_Toc137245837)

[Chapter 5 27](#_Toc137245848)

[Result and discussion 27](#_Toc137245849)

[5.1 Authentication: 27](#_Toc137245850)

[5.1.1 Welcome page: 27](#_Toc137245851)

[5.1.2 Registration page: 28](#_Toc137245852)

[5.1.3 Login and Password control: 28](#_Toc137245853)

[5.2 Bus Details: 29](#_Toc137245854)

[5.3 Tracking: 30](#_Toc137245855)

[Chapter 6 31](#_Toc137245856)

[Conclusion And Future Works 31](#_Toc137245857)

[6.1 Conclusion 31](#_Toc137245858)

[6.2 Future Work 31](#_Toc137245859)

[6.2.1 Integration with Other Transport Modes 31](#_Toc137245860)

[6.2.2 Predictive Analytics 31](#_Toc137245861)

[6.2.3 Enhanced User Experience 32](#_Toc137245862)

[6.2.4 Real-time Traffic Updates 32](#_Toc137245863)

[6.2.5 Integration with Payment Systems 32](#_Toc137245864)

[6.2.6 Continuous Data Analysis 32](#_Toc137245865)

[6.2.7 Integration with Smart City Initiatives 32](#_Toc137245866)

[References: 33](#_Toc137245867)

***Figure List***

Fig3.1 E-R diagram of bus tracking system

Fig:3.3 System architecture

Fig:3.2 Class diagram

Fig:4.1 System block Diagram

Fig:4.2 Data flow diagram

Fig 4.3 Set a discretions

Fig4.4 Work flow diagram

Fig:5.1 Welcome page

Fig:5.2 Register

Fig:5.3 Sign in and Password Controller

Fig:5.4 Bus Details

Fig:5.5 Tracking Buses

# 

# Chapter 1

# Introduction

## 1.1 General Introduction

The Bus Tracking System was created on the Android platform using the Dart programming language. It is built on client-server technology and makes use of a database. One Android user (the bus drivers) shares the bus's current location as well as additional information to the server. The data entered by the user is saved in the server's database. The data entered by the user is saved in the server's database. The administrator can access the registration page via the user app. The administrator can retain a record of the bus, including the bus number, schedule, route information, driver contract, and so on. The administrator has the authority to modify the bus record as needed. So, in summary, our system manages all data regarding the present location of the bus, and using this data, real-time tracking of the bus can be done, and this information is then delivered to distant users who want to know the real-time bus information. GPS (Global Positioning System) and Google Maps are used for development purposes. The solution includes a server-client application that displays the real-time location of the bus on Google Maps.

The mobile industry is one of the fastest and most dynamic businesses today. The need for efficient and immediate communication is everlasting. Market sectors and growing demanding consumers are always demanding better than ever. Having a mobile phone makes life easier. Communication is always part of our daily life and we cannot avoid it. The invention of the mobile phone has enabled huge leaps in technological innovation and new applications. It was originally intended as a mobile phone that could be carried wirelessly over long distances. Advances in communications, improvements in radio frequencies, and the development of the Internet have made mobile phones more sophisticated and easier to use in applications. Mobile phones are now equipped with navigation systems such as GPS and global positioning satellites to help travelers, tourists, adventurers and navigators get the most accurate overview of their current location. Plus, it helps you find the nearest facility you want to search for. Such as restaurants, hotels, shopping malls, banks, police stations and other places of interest. Vehicle tracking system is one of the most commonly used applications for vehicle tracking, it is also used to prevent vehicle theft. Nowadays, Android apps are a very good source for tracking vehicles. It provides real-time data on vehicle movements. Android phones are widely used for this purpose because they are equipped with GPS devices. It serves as both transmitter and receiver. The vehicle tracking system combines the use of automated vehicle location in individual vehicles with this group data collection software for a complete picture of vehicle location. Modern vehicle tracking systems typically use GPS to locate vehicles, but other types of automated vehicle positioning technology may also be used. Vehicle information can be viewed on an electronic map via the Internet. Urban transit agencies are increasingly common users of vehicle tracking systems, especially in large cities.

## 1.2 Objectives

In short, the objectives of the project are:

• The system will reduce the paperwork done by the bus management team.

• The system will be able to improve the efficiency and performance of bus service.

• The system will reduce the work done by the bus management team by automatically calculating the estimated bus time and displaying the bus's location in real-time to the user.

• The system must be able to show the estimated arrival time of each bus at each bus stop

For University:

* To make the university transportation system more efficient.
* To eliminate paper work and increase the level of accuracy
* To increase speed of service with the use of technology

## 1.3 Motivation

The proposed system provides the exact location of the bus to users from their location via a mobile application. In this project to receive signals from satellites, the driver has to enable GPS on his Android smartphone. The device receives GPS data and sends the latitude and longitude values ​​of the bus location to the server periodically.

Advantage of university students

-Makes students to avoid unexpected delays.

-Real time monitoring of college buses.

## 1.4 Project Overview

Bus Tracking System is a project that aims to provide real-time bus tracking and monitoring in the traffic network. It uses modern technologies such as GPS (Global Positioning System), mobile networks, and web applications to enable efficient management of bus fleets and improve the overall public transport experience. The system benefits both bus operators and passengers by providing accurate information on bus locations, estimated arrival times, and route updates.

Key Feature and component

**GPS Tracking Devices**: Each bus is equipped with a GPS tracking device that continuously broadcasts the bus's location coordinates. These devices use satellite signals to determine the bus's precise position.

**Central Server**: The central server is the core component of the bus tracking system. It receives the location data from the GPS devices and processes it to generate useful information. The server manages the data storage, handles communication with the mobile devices, and performs necessary calculations for route optimization and scheduling.

**Mobile Applications**: Mobile applications are developed for both bus operators and passengers. The bus operator app allows the fleet managers to monitor the buses, track their movements, and analyze performance metrics. The passenger app provides real-time information about bus locations, estimated arrival times, and any changes in the bus routes or schedules.

# Chapter 2

# Literature Review

## 2.1 Transit Apps

There are several popular transit apps available today that provide real-time bus tracking and information. Examples include Transit App, Moovit, and Citymapper. These apps use GPS data from buses to display real-time bus locations, estimated arrival times, and route information to users. They often integrate with mapping services like Google Maps or provide their own mapping functionalities.

### 2.1.1 Problem statement

1. Inaccurate or unreliable bus arrival predictions
2. Limited coverage and availability
3. User interface complexity and usability
4. Lack of integration with other transportation modes
5. Limited real-time updates and service alerts

## 2.2 Public Transportation Agencies

Many public transportation agencies have developed their own bus tracking applications to provide real-time information to passengers. These apps are often tailored to specific cities or regions and offer features such as live bus locations, arrival predictions, route maps, and service alerts. Examples include the MTA's "MTA Bus Time" app in New York City and the "Trans link" app in Vancouver.

### 2.2.1 Problem statement

1. Limited visibility and control over the bus fleet
2. Inefficient route planning and optimization
3. Lack of real-time information for passengers
4. Limited integration with other transportation modes
5. Data management and analysis challenges

## 2.3 Research Projects

Academic and research institutions have also conducted studies and developed bus tracking applications. These projects often focus on optimizing bus routes, reducing waiting times, or improving overall transportation efficiency. For example, researchers have explored machine learning algorithms to predict bus arrival times more accurately or used data analytics to identify congestion patterns and optimize bus scheduling.

## 2.4 IoT based Intelligent Bus Monitoring System

On reviewing the past work of college bus tracking, monitoring and alerting system, there is a possibility to categorize various methodologies and identify new trends. One among them is a challenge for vehicle tracking, monitoring and alerting system. Now- a-days with the increase in the crime rate and accidents, parents worry about their wards when they are going to colleges. And many Students find themselves locked in a college bus in the bus parking lot after falling asleep on their way to college, miss the bus, or leave at the wrong station. This project makes use of the applicability of radio frequency identification (RFID) technology for tracking and monitoring Student during their trip to and from college-on-college busses. And it has the advantage of efficient tracking capabilities, low cost and easy maintenance. The individual RFID tags are effective and it is used for tracking and monitoring Student. Fire sensor is also used in this project to detect any fire accidents. Speed of the bus also can be calculated and send a message to the parents through GSM.

### 2.4.1 Problem statement

**Lack of real-time visibility**: Existing bus monitoring systems often rely on manual reporting or scheduled updates, which can result in delays and inaccurate information. There is a need for a system that provides real-time visibility into the location, speed, and status of each bus in the fleet, enabling transportation authorities to track buses accurately and make informed decisions in real-time.

**Inefficient resource allocation**: Without a comprehensive monitoring system, it can be challenging to optimize the allocation of buses and drivers. A lack of insights into passenger demand, route performance, and vehicle conditions hinders effective resource planning. An intelligent bus monitoring system should provide data-driven analytics and predictive models to optimize bus deployment, manage driver schedules, and ensure efficient resource allocation based on demand patterns and operational requirements.

**Passenger experience and satisfaction**: Passengers often face uncertainty regarding bus arrival times, crowded buses, and service disruptions. This leads to dissatisfaction and affects their overall experience with public transportation. An IoT-based bus monitoring system should aim to provide accurate and real-time information to passengers, including estimated arrival times, crowd density, and service updates, enhancing their experience and improving satisfaction levels.

**Vehicle maintenance and health monitoring**: Proactive vehicle maintenance is crucial to ensure safe and reliable operations. Traditional maintenance approaches may rely on periodic inspections or reactive repairs, leading to unexpected breakdowns and service interruptions. An intelligent bus monitoring system should incorporate IoT sensors to monitor key vehicle parameters such as engine health, tire pressure, fuel consumption, and other critical systems. Real-time monitoring and predictive analytics can help identify potential issues early, enabling timely maintenance and reducing downtime.

**Data management and security**: IoT-based bus monitoring systems generate large volumes of data from multiple sources, including GPS trackers, sensors, and passenger feedback. Ensuring efficient data collection, storage, and analysis is vital for extracting meaningful insights. Additionally, data security and privacy must be addressed to protect sensitive information and prevent unauthorized access to the system.

# Chapter 3

# Methodology

## 3.1 Software Development Life Cycle (SDLC)

As the mobile applications have complex functionality the following Mobile Application Development Lifecycle model is proposed to enable a systematic approach in development.

**1. Implementation Phase**

In the first phase, ideas are collected and categorized. The main objective of this phase is to come out with a new idea or improvements to the existing application. The ideas can come from the user or from the developers. If the user himself comes out with the idea, the idea is further detailed and analyzed. Developers can brainstorm to generate ideas for new applications. The filtered list of ideas is discussed by the mobile application idea team comprising of the business and IT representatives for the feasibility to launch a project around the idea. If a similar application exists in the market, the popularity of the application and the features supported are studied and compared. The differences with the existing application(s) are documented. If no similar application exists on any mobile platform, then the idea with its core functionality should be documented. The other important task in this phase is to define the time required to develop the application. The initial requirement gathering should also be completed. The work done by the mobile application idea team should then be documented and forwarded to the design team.

**2.Design Phase**

In this phase, the idea from the mobile application team is developed into an initial design of the application. The feasibility of developing the application on all mobile platforms is determined. Alternatively, the specific target mobile platform is identified. A decision has to be made on whether the developed application is to be released as a free version or trial version with limited features or released only as a premium version. The application functionality is broken down into modules and into prototypes i.e., combination of modules which are to be released in the prototype fashion. The functional requirements are defined. The software architecture of the application is created. Then the prototypes and associated modules are defined. A very important part of the design phase is to create the story board

for the user interface interaction: this storyboard describes the flow of the application. The

design team’s work is documented and forwarded to the development team for coding.

**3. Development Phase**

In this phase, the application is coded. Coding for different modules of the same prototype can proceed in parallel. The development process can be in two stages: Coding for Functional Requirement and Coding for UI requirements. The code is developed first for the core functionalities. Parallel development can be done for modules of the same prototype that are independent of each other. Subsequently, these modules can be integrated. In the second stage, user interface is designed so that it can be supported on as many mobile operating system platforms as possible; it is not good practice to have a different look and feel for the same application on different platforms. The minimum set of interface components present in all mobile OS platforms should be used in the design. Finally, the documentation of the development phase is then forwarded to the prototyping phase.

**4. Prototyping Phase**

In this phase, the functional requirements of each prototype are analyzed; the prototypes are tested and sent to the client for feedback. After feedback is received from the client, the required changes are implemented through the development phase. When the second prototype is ready, it is integrated with the first prototype, tested and then sent to the client. The development, prototyping and testing phases are repeated until the final prototype is ready. The final prototype is sent to the client for final feedback. The work done in this prototyping phase is documented and then forwarded to the testing phase.

**5. Testing Phase**

Testing is one of the most important phases of any development lifecycle model. The testing of the prototype types is performed on an emulator/simulator followed by testing on the real device. The emulator/simulator is often provided in the SDK. The testing on the real device, for example in the case of Android operating system development, should be performed on multiple operating system versions, multiple models of handsets with variable screen size. The test cases are documented and forwarded to the client for feedback.

**6. Deployment Phase**

Deployment is the final phase of the development process. After the testing is completed and the final feedback is obtained from the client, the application is ready for the deployment. The application is uploaded to the appropriate application store/market for user consumption.

Before the application is deployed, the following steps are to be checked.

* Register as a developer on the respective application developer’s website by paying the annual fee, if necessary, for respective OS.
* Check the rules and regulations of the application store for the deployment of an application.
* Refine the application, for example, by removing all the log files and comments.
* Design the icon and wallpaper to be used on the application store.
* Create the file format required on operating system platform.

**7. Maintenance Phase**

The maintenance is the final phase of this model and this maintenance is a continuous process. Feedback is collected from users and required changes are made in the form of bug fixes or improvements. Appropriate security patches, performances improvements, additional functionality, new user interfaces should be provided at regular intervals in the form of updates to the application. The maintenance phase also includes the marketing of the application: advertising and highlighting its unique features. If any application requires a backend server: this server and related operating system must be maintained as well.

## 3.2 E-R diagram

We can express overall logical structure of database graphically with an E-R diagram

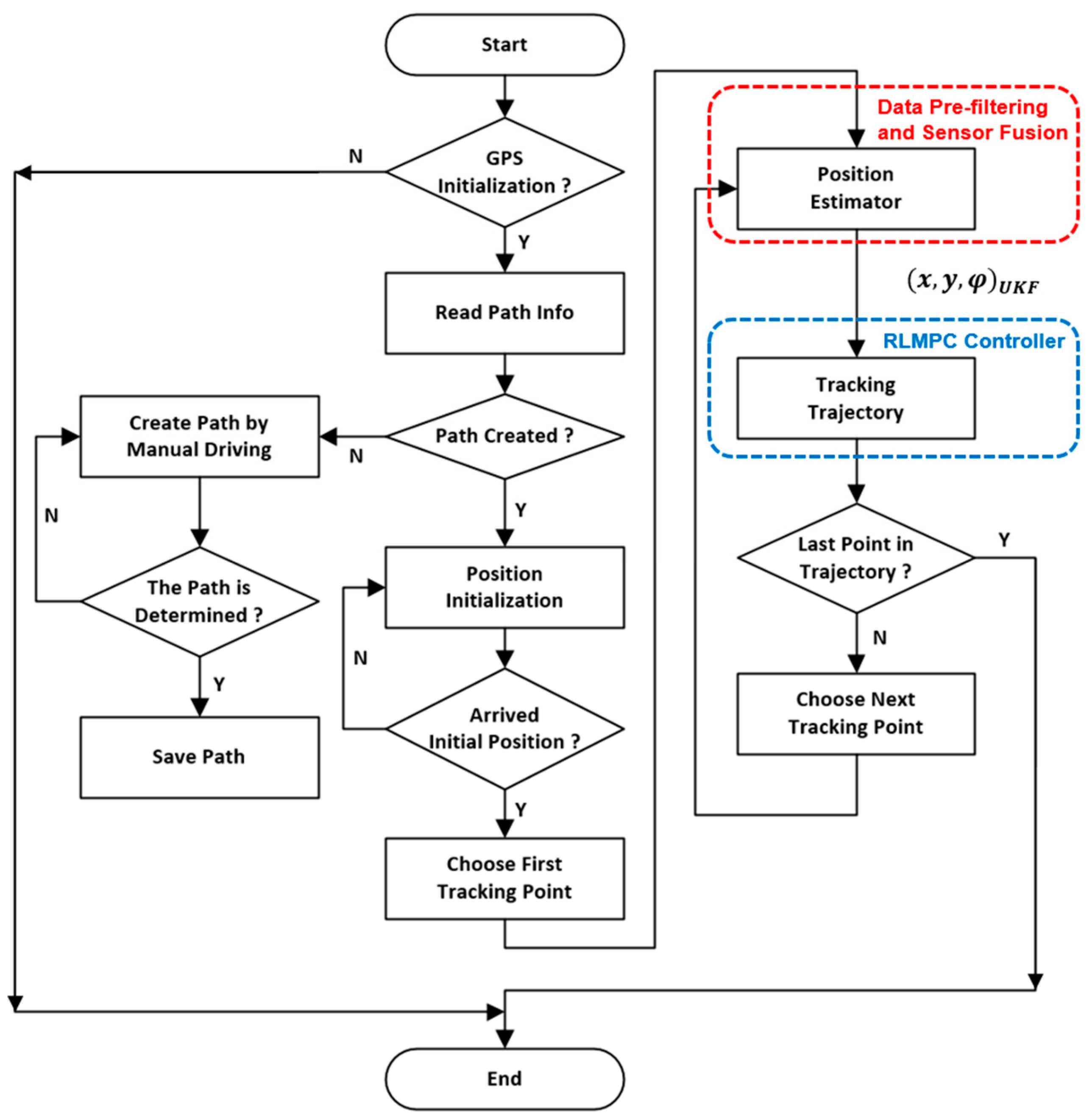


Fig3.1 E-R diagram of bus tracking system

## 3.3 Class Diagram

Server

+fatchInfo

+response to the client

Bus Application

+latitude

+longitude

+date

+time

Client Applications

+getBusInfo

+getBusLocation

Database

+Name

+Size

+Insert

+Update

Bus

+BusNumber

+Bus Route

Fig:3.2 Class diagram

## 3.4 System Architecture

Based on the chosen technology, design a system architecture that outlines the components, infrastructure, and data flow. Determine how the buses will be equipped with tracking devices, how the data will be collected and transmitted, and how it will be processed and made accessible to relevant parties

Admin/Transport Authority

Bus operator 3

Bus operator 2

Bus operator 1

Bus tracking device

Bus tracking device

Bus tracking device

Bus Tracking Mobile Application using google map

Data Storage and processing system

Data Storage and processing system

Data Storage and processing system

Fig:3.3 System architecture

# Chapter 4

# Design And Implementation

This chapter describes features, fragments, classes, architecture and the application itself by providing necessary information of major components. First, overall information is given along with project’s components and classes. Subsequently, the architecture details of the application are discussed

## 4.1 Overview

**﻿**Application starts with instantiating Location Manager. This is needed to track user location. Detailed description regarding Location Manager is provided in this section. Next, UI and user interaction handling sets up all necessary selections.

## 4.2 Components

In order to provide a detailed view concerning system mechanism, project can be grouped in three segments. These are

* Location Manager,
* Fragments
* AsyncTask & Services.

### 4.2.1 Location Manger

This class provides access to the system location services. These services allow applications to obtain periodic updates of the device's geographical location, or to fire an application specified Intent when the device enters the proximity of a given geographical location.

### 4.2.2 Fragments

Fragments handle core operations in this project such as Tracking, establishing connection, opening sockets, showing details of buses. They are briefly responsible for managing all session and configuring operations. Fragments also include asynchronous methods with respect to file & text transfer as well as socket connection.

A Fragment is a piece of an application's user interface or behavior that can be placed in an Activity and it represents a particular operation or interface that is running within a larger Activity. A Fragment is closely tied to the Activity it is in, and cannot be used apart from one. Though Fragment defines its own life cycle, that life cycle is dependent on its activity: if the activity is stopped, no fragments inside of it can be started; when the activity is destroyed, all fragments will be destroyed.

### 4.2.3 Async Task and services

To perform an asynchronous task for retrieving the current location using the Google Maps API in a Flutter application, you can utilize the geolocator package along with the async and await keywords. Here's an example of how you can implement it:

Firstly, add geolocation in pubspc.ymal file then import that package

import 'package: cloud\_firestore/cloud\_firestore.dart';  
import 'package: flutter/material.dart';  
import ‘package: google\_maps\_flutter/google\_maps\_flutter.dart';  
import 'package: location/location.dart' as loc;

Some methods are:

1. cameraPosition()
2. getCurrentlocation ()
3. \_getLocation()
4. \_listenLocation ()
5. stopListening ()

## 4.3 Block Diagram

Network

Mobile application for user

Mobile application for Driver

Admin

Server

database

Fig:4.1 System block Diagram

## 4.4 Data flow Diagram:

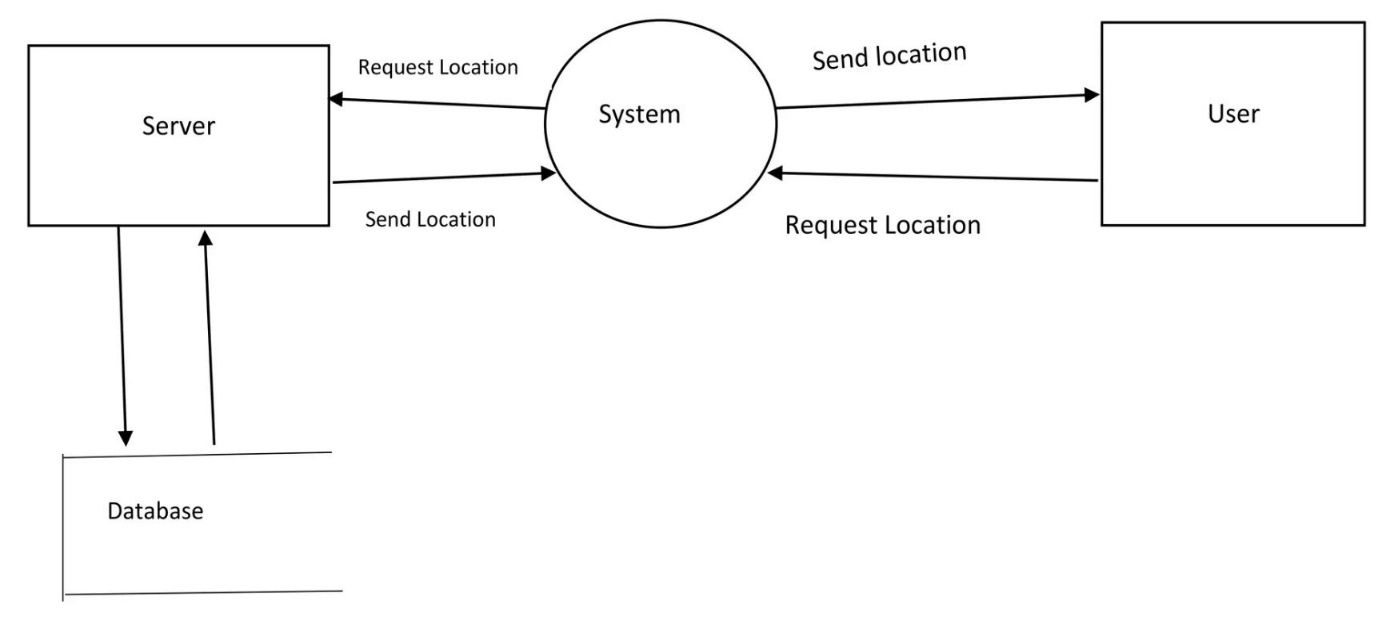
****

Fig:4.2 Data flow diagram

## 4.5 Data Dictionary:

**Table-1 Showing Data**

|  |  |  |
| --- | --- | --- |
| Field Name | Type | Descriptions |
| Driver name | String | Primary key |
| Bus number | Integer | Primary key |
| Latitude | Integer |  |
| Longitude | Integer |  |

**Table-2 Collecting data**

|  |  |
| --- | --- |
| Field name | Type |
| User Name | Var |
| Phone number | Integer |
| Email | String |

To collect data from users and save it to a Firebase Realtime Database in a Flutter application, you can follow these steps:

1.Set up Firebase:

Create a new Flutter project or open an existing one.

2.Add the required dependencies:

Open your pubspec.yaml file and add the following dependencies:

cloud\_firestore: ^4.7.1

firebase\_core: ^2.13.0

3.Import the necessary packages:

In your Dart file, import the required packages

import 'package: cloud\_firestore/cloud\_firestore.dart';

import 'package: firebase\_core/firebase\_core. Dart';

**4.6 Set direction and Marker:**

For direction use polyline package in flutter pub dev and import package

import 'package: flutter\_polyline\_points/flutter\_polyline\_points. Dart';  
import 'package: google\_maps\_flutter/google\_maps\_flutter.dart';  
import 'package: location/location.dart';

Package polyline helps to set direction user current location to destination

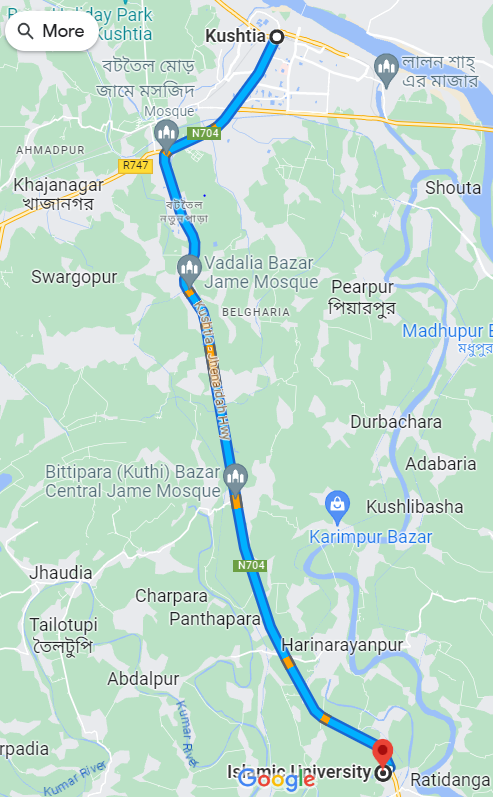


Fig 4.3 Set a discretions

## 

## 4**.7 Workflow Diagram**

## 

# 

Capture bus location

Update bus location on the map

# 

Checking bus root change

# 

Update bus route information and notify user

True

Condition

false

# 

Continue monitoring bus

Fig 4.4 Workflow Diagram

# Chapter 5

# Result and discussion

The bus tracking project has been successfully implemented, providing real-time tracking and monitoring of buses in the transportation network. Here is the result and discussion of the project:

## 5.1 Authentication:

Authentication is an important part of many applications, including those that deal with user data and sensitive information. It ensures that only authorized users can access some features or data of the application. As part of a bus monitoring project, authentication can be implemented to ensure access to administrative functions, user profiles, and other sensitive areas of the system.

### 5.1.1 Welcome page:

When a user opens the app, a welcoming interface will first be there to welcome and engage the user. A motion picture of a bus would be on the screen and there will be two buttons for registering and login purposes.



Fig:5.1 Welcome page

### 5.1.2 Registration page:

When the user clicked on the registration button, A registration form will pop up and the form contains required information query including Full name, E-mail Id, password for security and another field for confirming the entered password Only after filling the registration form, user can sign up to the application.

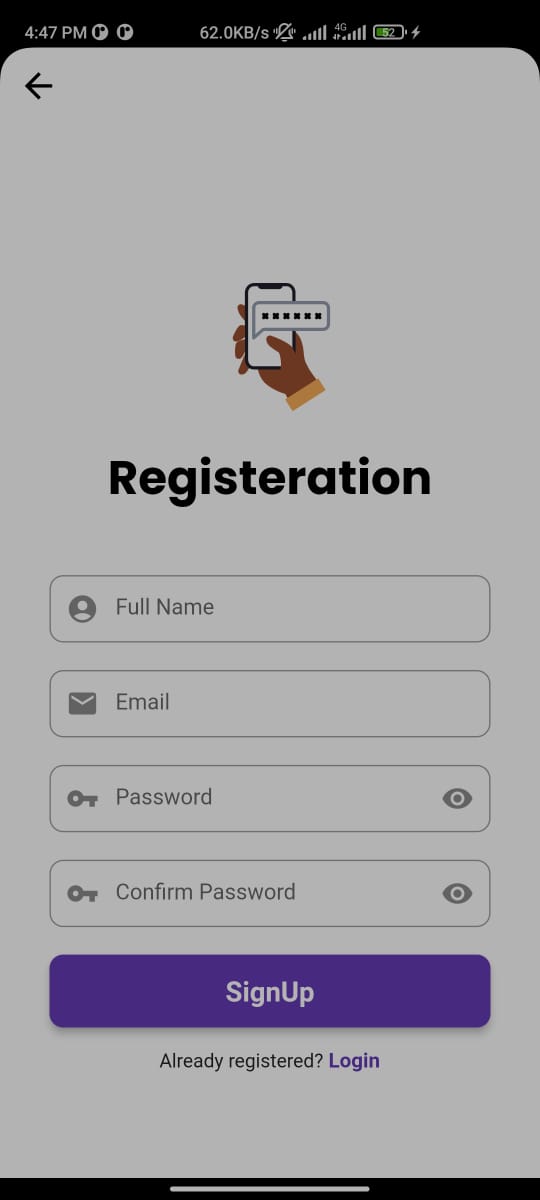


Fig:5.2 Register

**Note:** And if the user has already an account, he can skip the sign in part and click the login button

### 5.1.3 Login and Password control:

In order to log in to the application the user only need to fill two field and those are Email Id and password. And if the user forgot the password, he can login to the account by clicking "forgot password "

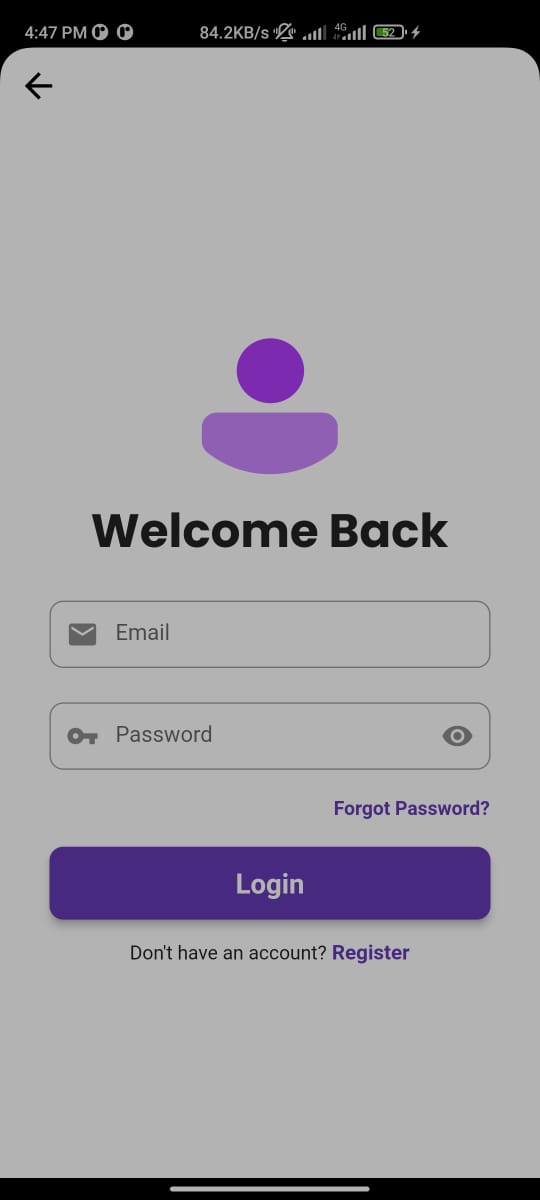
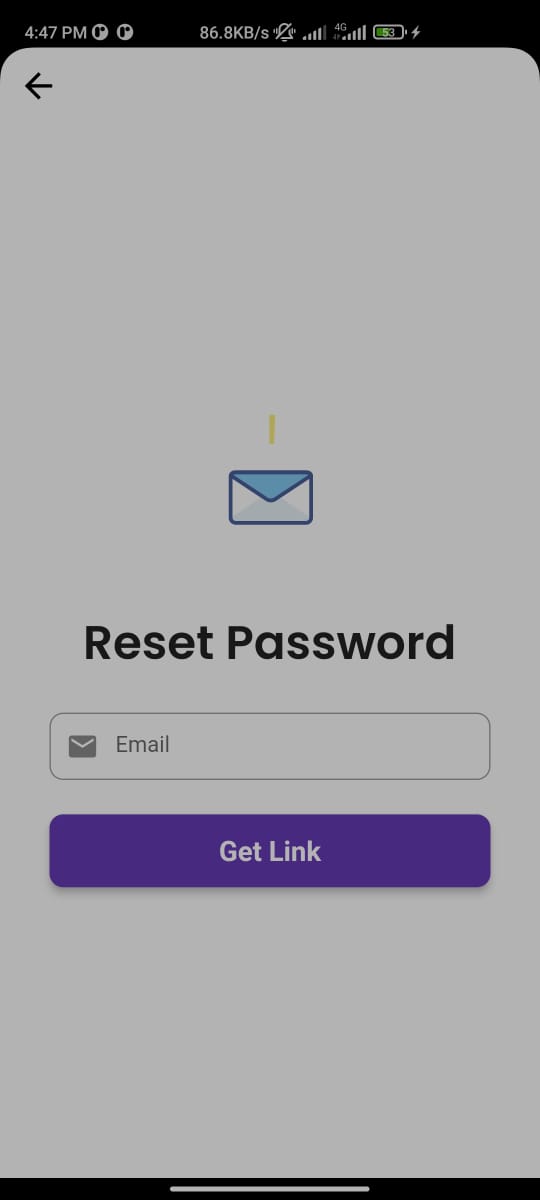
 

Fig:5.3 Sign In and Password Controller

## 5.2 Bus Details:

After login to the account this page will be there on the screen. Here user can enable live location to locate the bus location. And he can also stop the location sharing if he wants.

Bellow that, there will be information about the drivers and about which bus they are driving



Fig:5.4 Bus Details

## 5.3 Tracking:

On the right side of the driver’s name, there is a direction sign that enables user to examine where the bus is, and by clicking enable live location, user's current location will be linked up to the bus location.

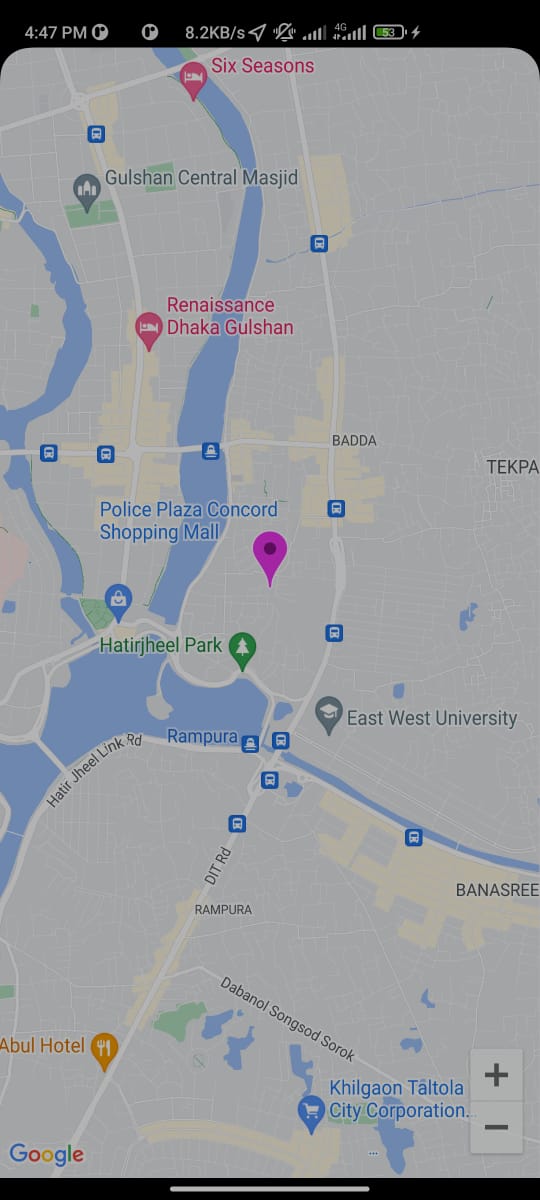
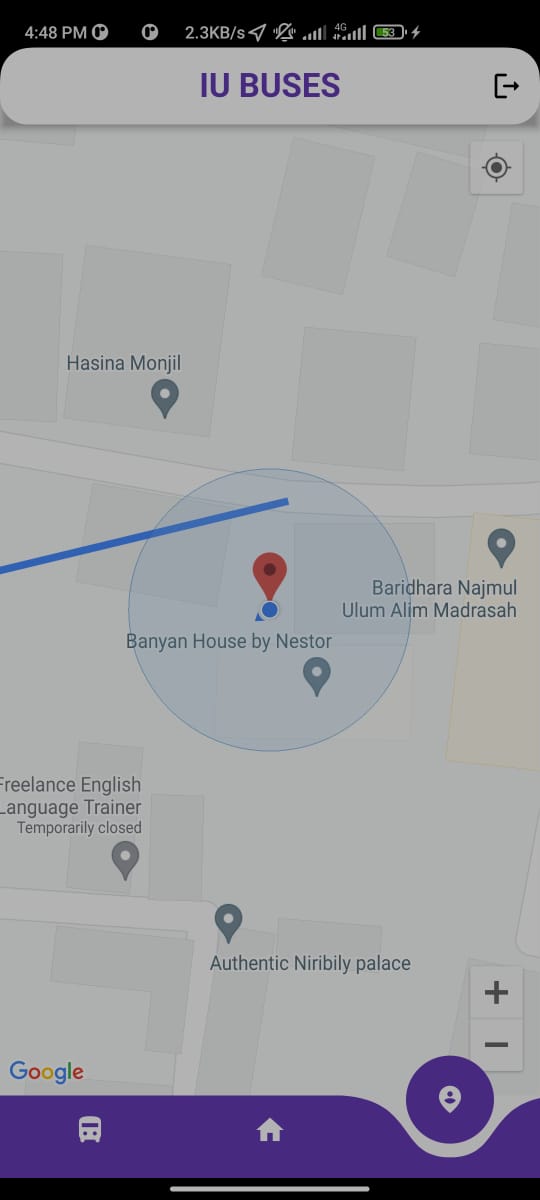
 

Fig:5.5 Tracking Buses

# Chapter 6

# Conclusion And Future Works

## 6.1 Conclusion

In summary, the Google GPS-based bus tracking application project has successfully achieved the goals of real-time bus tracking and monitoring, improving operational efficiency, and improving passenger experience. The project demonstrated the efficient integration of Google Maps API, GPS technology, and mobile apps to create a complete bus tracking and management solution.

With the implementation of real-time tracking, passengers can access accurate information on bus location, estimated time of arrival, and any change in route or schedule. This has improved the overall passenger experience by reducing wait times and allowing for better trip planning. Bus operators have benefited from the project by better understanding fleet performance, punctuality, and route optimization to improve operational efficiency. The project shows the potential to leverage technology to improve the public transport system. By providing real-time tracking and monitoring, the app increases the safety, security, and reliability of bus services. It also enables operators to make data-driven decisions, optimize resources, and react quickly to emergencies or incidents.

## 6.2 Future Work

There are several areas of future work that can further enhance the Google GPS-based bus tracking application:

6.2.1 Integration with Other Transport Modes

Expand the application to include integration with other transport modes, such as trains or taxis, to provide a seamless multi-modal transportation experience for users.

### 6.2.2 Predictive Analytics

Utilize historical data and machine learning algorithms to develop predictive analytics models that can forecast bus arrival times more accurately, accounting for various factors like traffic conditions, weather, and historical patterns.

### 6.2.3 Enhanced User Experience

Continuously improve the user interface and user experience of the mobile application to make it more intuitive, user-friendly, and accessible for a wider range of users

### 6.2.4 Real-time Traffic Updates

Integrate real-time traffic information into the application to provide users with updates on traffic conditions and suggest alternative routes in case of congestion or accidents.

### 6.2.5 Integration with Payment Systems

Integrate payment systems within the application to allow passengers to conveniently purchase tickets or make fare payments directly through the app.

### 6.2.6 Continuous Data Analysis

Regularly analyze the collected data to identify trends, patterns, and areas for improvement. This can help optimize bus routes, identify areas of high demand, and enhance operational efficiency.

### 6.2.7 Integration with Smart City Initiatives

Collaborate with smart city initiatives to leverage data from various sources, such as traffic management systems or weather data, to further optimize bus operations and improve transportation services.

By focusing on these areas of future work, the Google GPS-based bus tracking application can evolve into a comprehensive and intelligent transportation management system that provides even greater benefits to both operators and passengers, contributing to a more efficient and sustainable public transportation network.

## References:

[1] T.P. Liang, et al.,” Adoption of mobile technology in business-a fit viability model”, Industrial Management & data systems, vol. 107, pp. 1154-1169, 2007.

[2] Yadav ,B “ Collage Bus tracking application”

<https://www.academia.edu/27965619/KANTIPUR_ENGINEERING_COLLEGE_College_Bus_Tracking_System_>

[3] <http://javapapers.com/android/android-location-tracker-with-google-maps/>

Accessed on-15th January

[4]<http://www.codeproject.com/Articles/665527/A-GPS-Location-Plotting-Android-Application> Accessed on-15th January

[5]<https://developer.android.com/guide/topics/location/strategies.html>

Accessed on-16th January

[6]<http://www.todroid.com/how-to-create-a-google-map-application-using-androidstudio/>

Accessed on-17th February

[7] N. Dhanasekar(2019) “IOT based smart bus tracking system” <https://www.ijert.org/iot-based-intelligent-bus-monitoring-system>

[8] <https://console.cloud.google.com/welcome/new?project=learning-maps-388218>

Access on – 13th march

[9] Dart <https://en.wikipedia.org/wiki/Dart_(programming_language)>

[10] Flutter Widget <https://medium.com/flutter-community/an-introduction-to-flutter-its-all-widgets-aabd0f86ca1c>

[11] Firebase Documentation <https://firebase.google.com/docs>