A
PROJECT REPORT
ON

"Public transport application for providing offline timetables of buses and train in remote areas"

Submitted in Partial Fulfillment of Requirement for the Award of
Degree of
BACHELOR OF TECHNOLOGY
COMPUTER SCIENCE AND ENGINEERING
of
DR. BABASAHEB AMBEDKAR TECHNOLOGICAL UNIVERSITY, LONERE
Submitted By

Names of Students	Exam Seat No.
Mr. Prasad S. Junghare.	1967971242007
Mr. Yash V. Kadam.	1967971242010
Miss. Shreya. S. Katte.	1967971242011
Mr. Saras S. Shinde	1967971242039

UNDER THE GUIDANCE OF
Dr. V. K. Bhosale
(Head of Department)
Computer Science and Engineering

Raosaheb Wangde Master Charitable Trust's

Dnyanshree Institute of Engineering and Technology

Sajjangad Road, Tal. Dist. Satara, Maharashtra State, 415 013. 2022-2023

CERTIFICATE

This is to certify that the project report entitled, “Public transport application for providing offline timetables of buses and train in remote areas”

Submitted by

Names of Students	PRN No.	Roll No.
Mr. Prasad S. Junghare.	1967971242007	4307
Mr. Yash V. Kadam.	1967971242010	4310
Miss. Shreya S. Katte.	1967971242011	4311
Mr. Saras S. Shinde.	1967971242039	4339

It is a bonafide work carried out by these students under guidance of Dr. V. K. Bhosale. It has been accepted and approved for the partial fulfillment of the requirement of Dr. Babasaheb Ambedkar Technology Lonere, for the award of the degree of Bachelor of Technology (Computer Science and Engineering). This work and project report has not been earlier submitted to any other Institute or University for the award of any degree or diploma.

Dr. V. K. Bhosale.	Dr. V. K. Bhosale	Dr. A. D. Jadhav
(Project Guide)	(Head of Dept.)	(Principal)
	Computer Science and Engineering	

Prof.
(External Examiner)
Place: Satara.
Date:

ACKNOWLEDGEMENT

The satisfaction and euphoria that accompany the successful completion of any task would be incomplete without the mention of people who made it possible. So, we acknowledged all those whose guidance and encouragement served as a beacon of light and crowned our efforts with success.

We have immense pleasure in expressing thanks to the principal **Dr. A. D. Jadhav**, for providing all the facilities for the successful completion of the project. Our profound sense of gratitude is due to our Project guide and Head of the department **Dr. V.K. Bhosale**, for constant encouragement, Valuable guidance and also for the inspiration to pursue the project and guided us in this Endeavour.

We also thanks to **Prof. P.M. Pondkule**. Coordinator, for their guidance and for being a constant source of support.

Last but not the least we are thankful to all faculty members and lab instructors without whose support at various stages, this project would not have materialized.

"Thanking You"

Mr. Prasad S. Junghare
Mr. Yash V. Kadam
Miss. Shreya S. Katte
Mr. Saras S. Shinde

Place:
Date:

List of Figures

Abstract

Today the transport system has been kind of a mess, especially in remote areas, bus arriving timing, depart timing has been a major issue. Also, if you are in a new area which is completely new, and you don’t have clue of the transport system, then you don’t have many choices in the market for doing the same. This application will solve the same problem, help travelers to find the nearest bus, train station also checks time of bus, train, live track it and check the ticket fare and also available to book tickets.

Keywords: Android / IOS application, Transport system, timetables, book tickets.

Contents

Chapter 1

Introduction

React Native is a popular framework for building mobile applications that run natively on both Android and iOS platforms. In this project, we propose to develop a public transport system using React Native.

Public transportation plays a vital role in the transportation sector and greatly impacts the daily lives of people. It helps in reducing traffic congestion, improving air quality, and providing an affordable mode of transportation for people.

The proposed public transport system aims to provide an easy and convenient way for people to access and use public transport services. It will provide real-time information about the arrival and departure times of buses, trains, and other modes of transport. Users will also be able to purchase tickets and plan their journeys using the app.

Overall, the goal of this project is to create a user-friendly public transport system that improves the experience of using public transportation and helps to encourage more people to use it.

.Chapter 2

Literature Review

In year 2008 authors Noppadol Chadil, Apirak Russameesawang, etc.all , [1] real-Time Tracking Management System Using GPS, GPRS and Google Earth[IEEE]
Due to the high cost of fossil-based energy, several methods are proposed to reduce the usage of the energy in logistics and fleet management to be even more. GPS tracking system is a common approach to get vehicle location information in real-time for fleet planning.

In year 2011 authors Mohammad A. Al-Khedher, [2] Dec 2011 an integrated GPS-GSM system is proposed to track vehicles using Google Earth application[IEEE]
An integrated GPS-GSM system is proposed to track vehicles using Google Earth application. The remote module has a GPS mounted on the moving vehicle to identify its current position, and to be transferred by GSM with other parameters acquired by the automobile's data port as an SMS to a recipient station.

In year 2018 authors Sanam Kazi , Murtuza BagasrawalaFarheen Shaikh, Anamta Sayyed [3] proposed Smart E-Ticketing System for Public Transport Bus[IEEE]
Bengaluru Metropolitan Transport Corporation (BMTC) is a prominent public transport service provider in Bengaluru. It makes commuting favourable and cheap compared to other modes of transport within the city.

In year 2019 authors Shubham Jain, Adarsh Trivedi, Shweta Sharma, [4] proposed Application Based Bus Tracking System This research is based on Bus Tracking System[IEEE]
This research is based on Bus Tracking System, The main objective of this work is to make a GPS tracking application which would be able to track school buses more accurately and efficiently than present bus-tracking systems.

In year 2020 authors Ayman R. Mohammed, [5] propsed UML Modeling of Online Public Bus Reservation System in Egypt.

Designing a proper public transportation system is a main concern for many countries with large population size like Egypt. Therefore, the online based transportation business for individuals is highly growing in Egypt and the concept of online reservation in the transportation field has arisen.

In year 2021 Authors Prof. M. Chavan, Navarange Prajwal, More Vishal, Nagargoje Shubham [6] proposed Online Bus Reservation System.

Bus reservation system deals with maintenance of records of details of each passenger. It also includes maintenance of information like schedule and details of each bus. We observed the working of the Bus reservation system and after going through it, we get to know that there are many operations, which they have to do manually. It takes a lot of time and causing many errors while data entry. Due to this, sometimes a lot of problems occur and they were facing many disputes with customers. To solve the above problem, and further maintaining records of passenger details, seat availability, price per seat, bill generation and other things, we are offering this proposal of computerized reservation system.

Chapter 3

Design, Development and Drawing

3.1 Proposed Work:

3.1.1 Problem Definition:

We will develop a React Native application that will provide offline timetables, online ticket booking capabilities, and other basic functions for the public transportation system.

3.1.2 Project Objectives

Main objectives of our proposed system are as follows

1.Developing a React Native application is the objective of this project.
2.The purpose of this service is to provide users with online and offline timetables for public Transport buses and trains. .
3.Provide users with the ability to book tickets using the application

3.2 Design

3.2.1 System Architecture

The system architecture of React Native based public transport application is shown in fig.

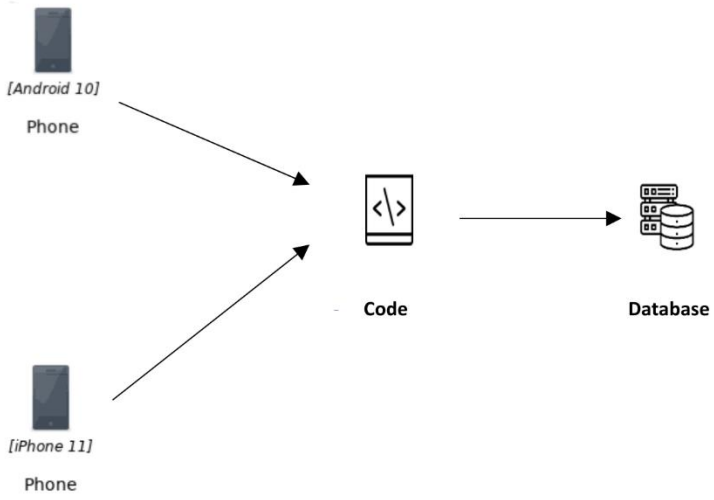


Figure 3.1: System Architecture

3.3 Development

3.3.1 Software Requirement Specifications

1 Visual studio code 5.6.2
2 Android studio 7.4.4
3 Expo Go 2.26.6
4 JavaScript ES6
5 React Native 0.70
6 Git Desktop 2.39.0
7 NodeJS Python Jdk8 18.0.0
8 GitHub Desktop 3.7.2
9 HTML5, CSS 2.1
10 Operating System – Windows 10 or higher
11 MySQL 5.7
12 php 8.1
13 Firebase 10.1.0
14 Xampp Server

3.3.2 Hardware Requirement Specifications

1 RAM : 8/12 GB
2 Processor : i5 ,i7 / RYZEN 5, RYZEN 7
3 Graphics card: 2 GB

3.4 Methodology

In this project, we are developing a React Native based public transport application. We are using React Native, React Js, HTML and CSS for Frontend design and Firebase, php, Xampp, MySQL, for the backend part.

3.4.1 Backend

In a react native app, the back end part of a public transport system would be responsible for handling the data and logic required to support the functionality of the app. This could include tasks such as:

1. Storing and retrieving information about routes, schedules, fares, and other details of the public transport system
2. Processing requests from the app's front end to retrieve this information and present it to the user
3. Handling user requests to purchase tickets or make reservations
4. Handling requests from the app to update or modify the data in the system, such as when a user adds a new route or updates their ticketing information
5. Overall, the back end of a public transport system in a react native app would be responsible for handling the data and logic required to support the functionality of the app and provide a seamless experience for the user.

3.4.2 Front-end

The front-end part of the public transport system in a react native app will be responsible for the visual and interactive elements of the app that the user interacts with. This includes the layout and design of the app, the buttons and controls for the user to input their desired destination and preferences, and the display of information such as real-time bus or train schedules, route maps, and ticket prices. The front end part of the app will also handle any user input and communication with the back end part of the app, which is responsible for storing and processing data and making API calls to external systems.

3.5 Drawing

3.5.1 System Flow

The system flow of React Native based public transportation application. When users interact with the system then flow gets started.

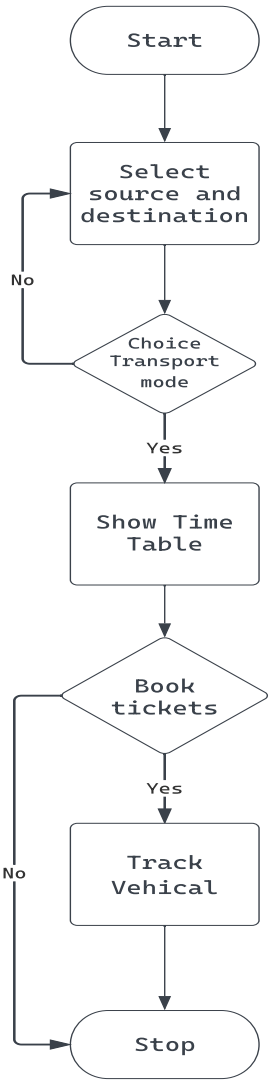


Figure 3.2: System Flow

3.5.2 Data Flow Diagram Level 0

DFD shows what kind of information will be input to and output from the system, where the data will come from and go to and where the data will be stored. Systems Data Flow Diagram Level 0 is shown in fig...

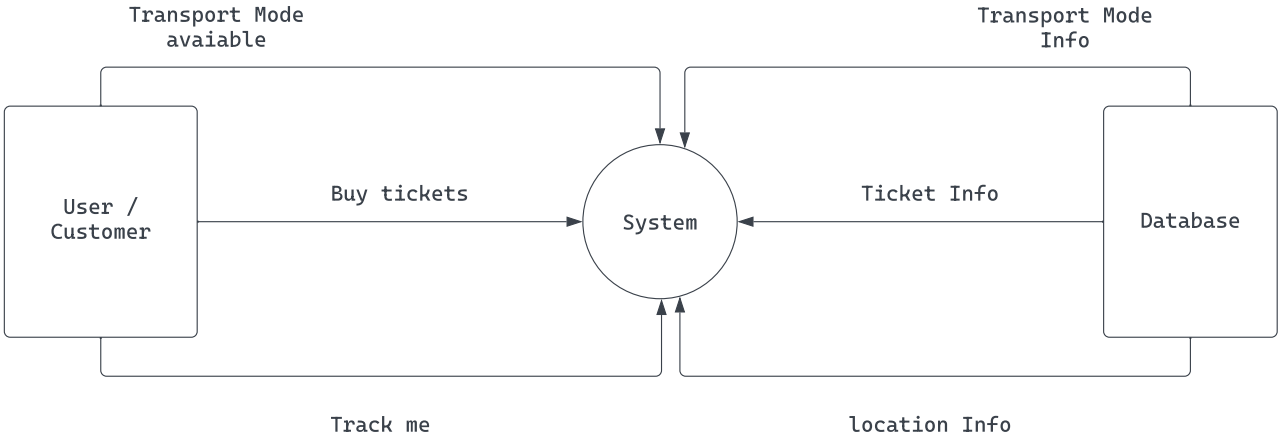


Figure 3.3: Data flow diagram Level 0

3.5.3 Data Flow Diagram Level 1

Level 1 DFD. The next stage is to create the Level 1 Data Flow Diagram. This highlights the main functions carried out by the system. As a rule, we try to describe the system using between two and seven functions - two being a simple system and seven being a complicated system.

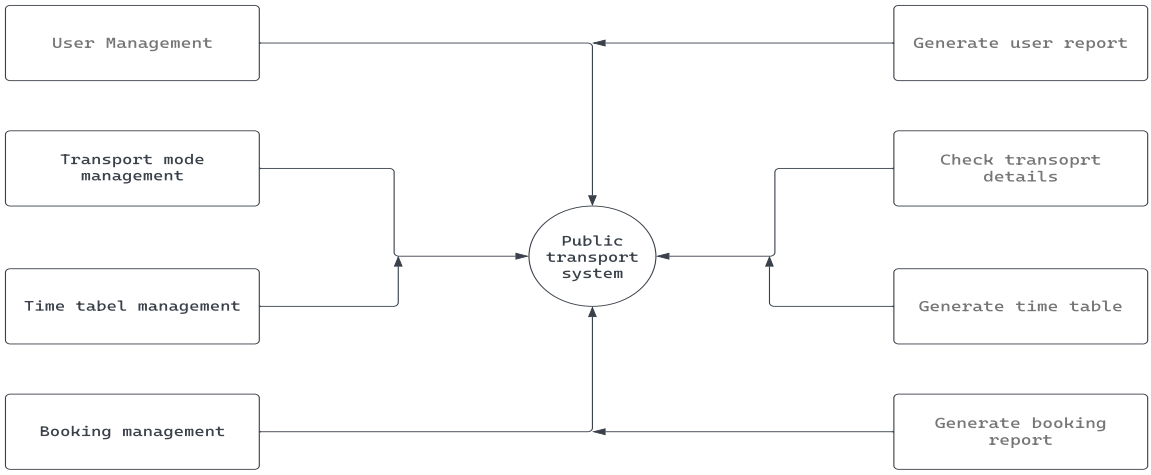


Figure 3.4: Data flow diagram Level 1

3.5.3 Data Flow Diagram Level 2

Data flow diagram (DFD): Level 2 This level two data flow diagram (DFD) template can map out information flow, visualize an entire system

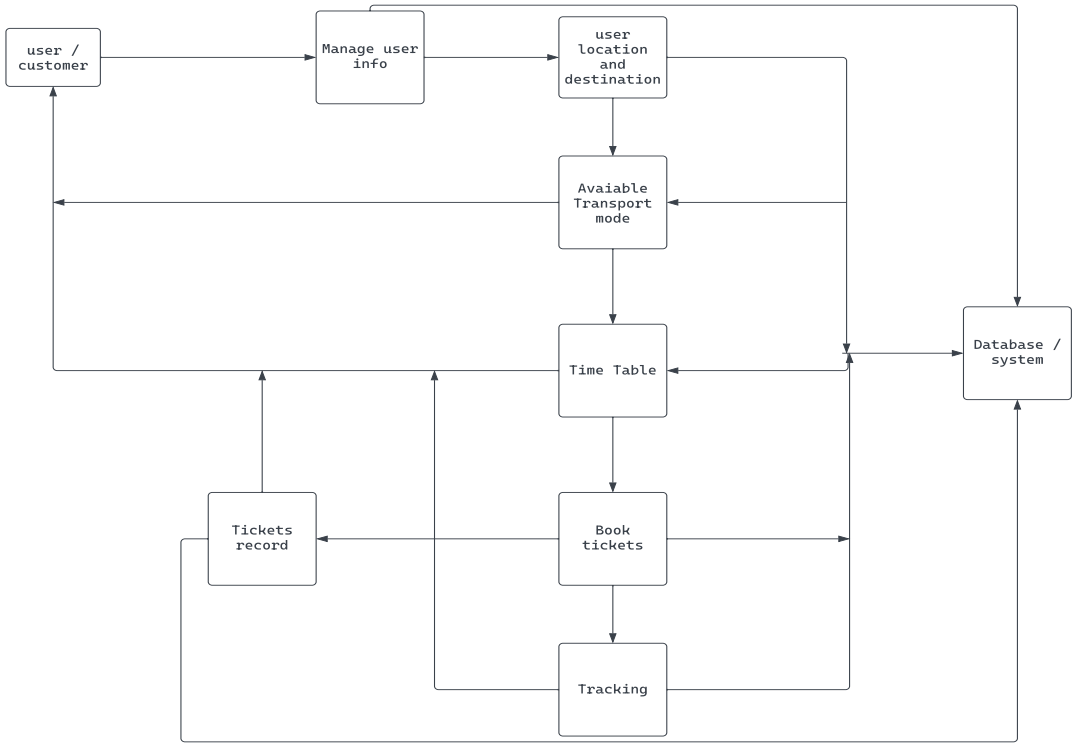


Figure 3.5: Data flow diagram Level 2

3.6 UML Diagrams

3.6.1 Class Diagram

A class diagram in the Unified Modeling Language (UML) is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations (or methods), and the relationships among classes. It explains which class contains information.

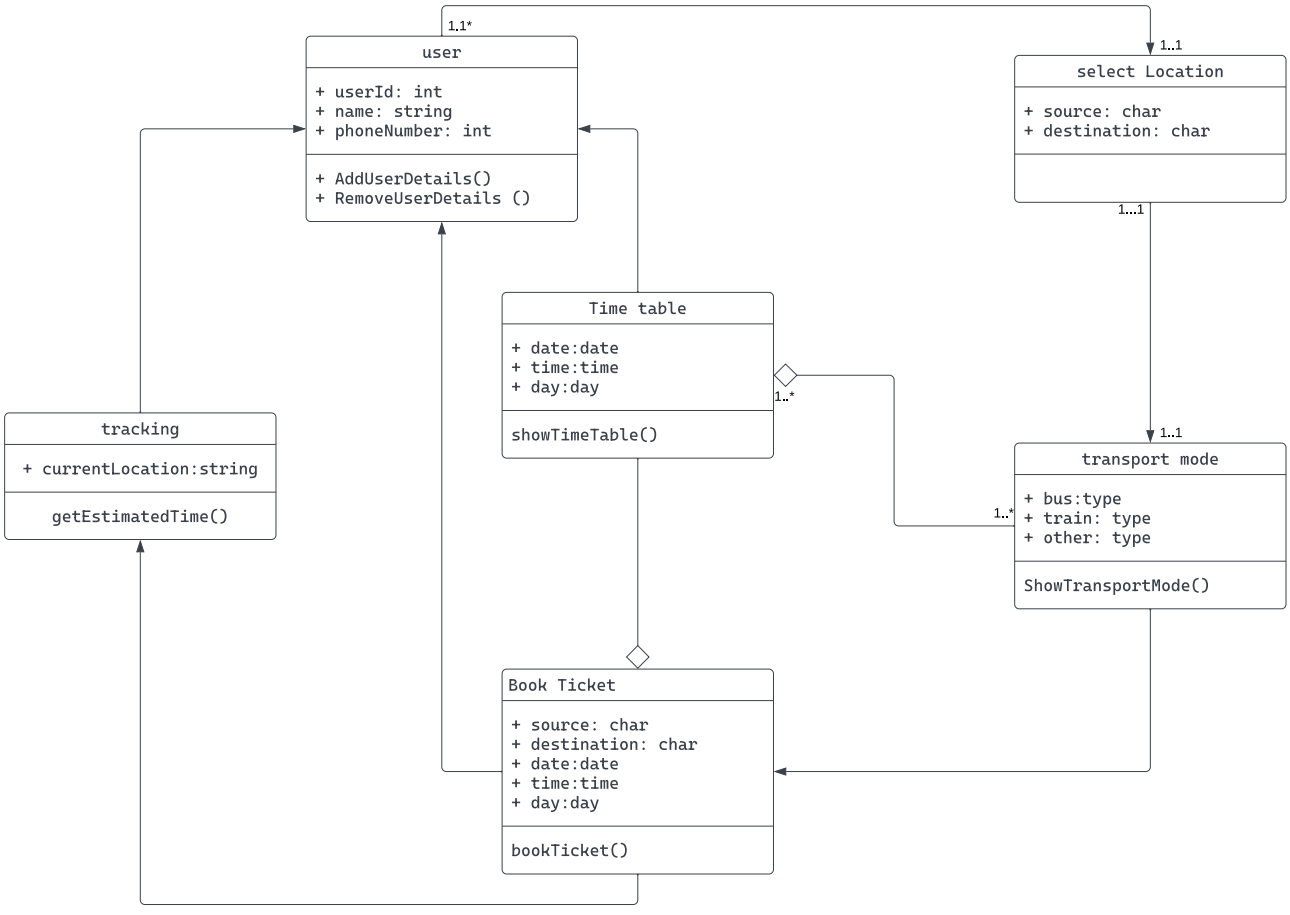


Figure 3.6: Class Diagram

3.6.2 Object Diagram

Object Diagram show the runtime entity or real world entity with respect to class diagram.

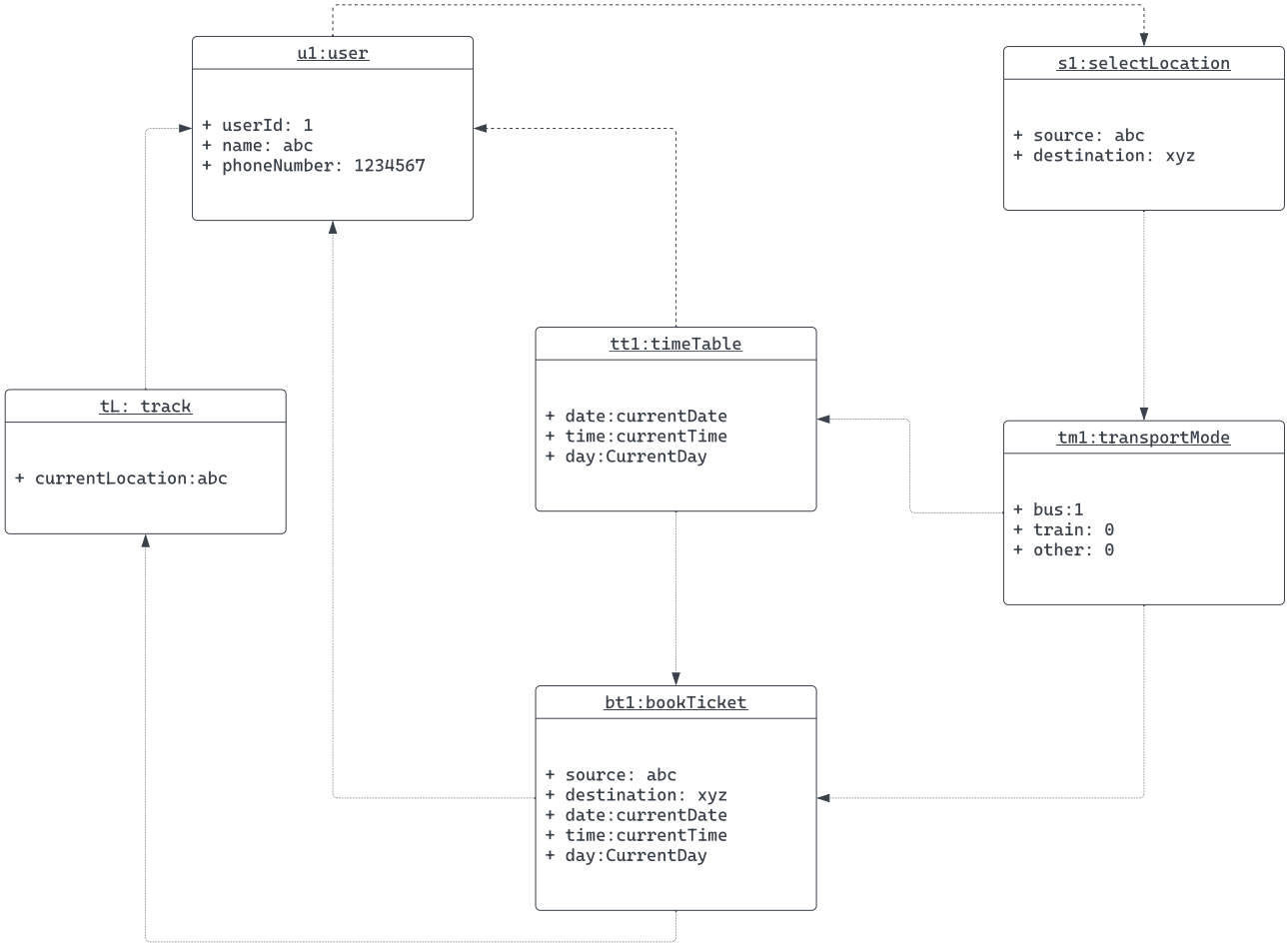


Figure 3.7: Object Diagram

3.6.3 Component Diagram

Component diagrams are used in modeling the physical aspects of object-oriented systems that are used for visualizing, specifying, and documenting component-based systems and also for constructing executable systems through forward and reverse engineering. Component diagrams are essentially class diagrams that focus on a system's components that often used to model the static implementation view of a system.

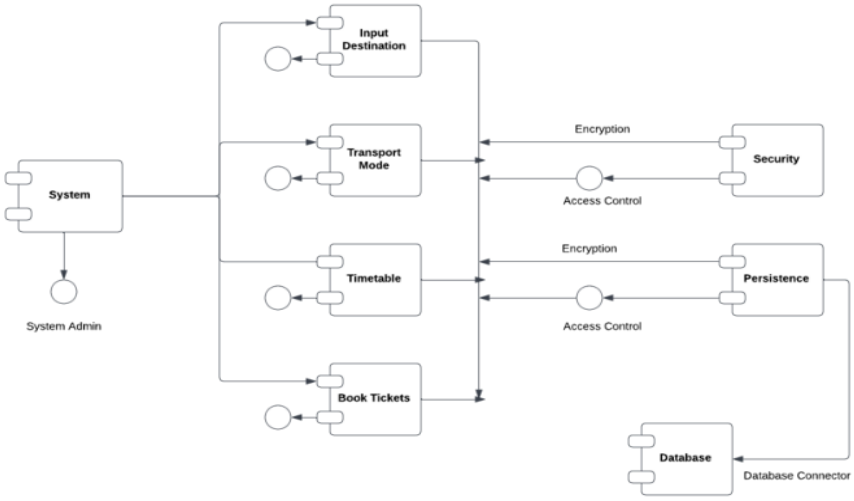
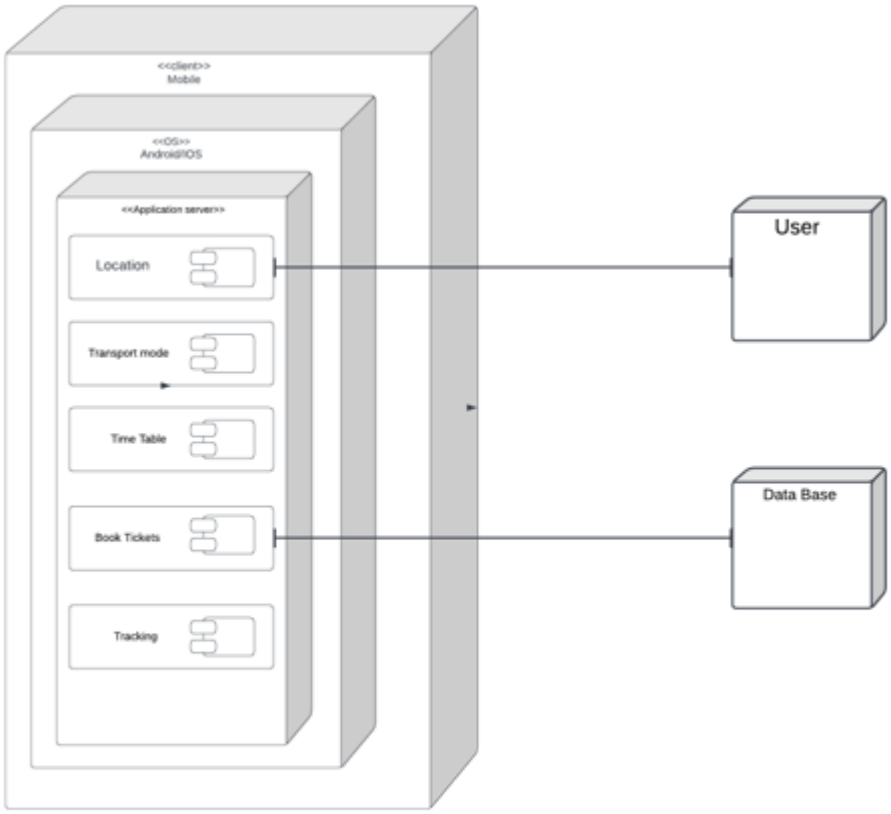


Figure 3.8: Component Diagram

3.6.4 Deployment Diagram

The Deployment Diagram also helps to model the physical aspect of an Object-Oriented software system. It models the run-time configuration in a static view and visualizes the distribution of components in an application.



DEPLOYMENT DIAGRAM

Figure 3.9: Deployment Diagram

3.6.5 Package Diagram

A package diagram in the Unified Modeling Language depicts the dependencies between the packages that make up a model.

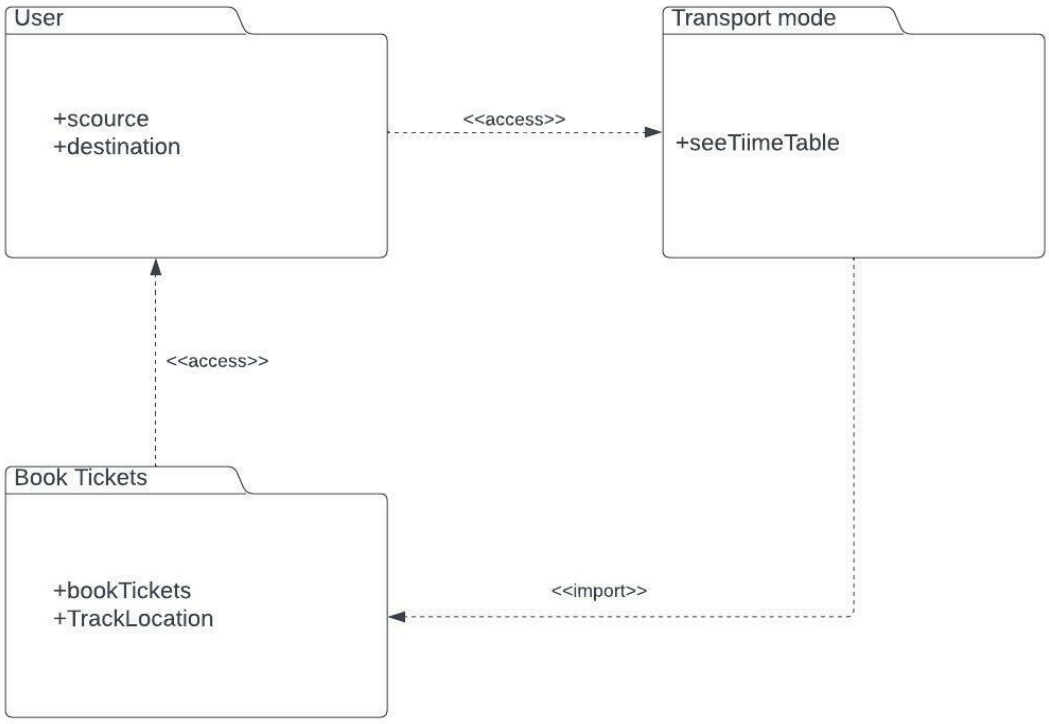


Figure 3.10: Package diagram

3.6.6 State Diagram

A state diagram, also known as a state machine diagram or state chart diagram, is an illustration of the states an object can attain as well as the transitions between those states in the Unified Modeling Language (UML). In this context, a state defines a stage in the evolution or behavior of an object, which is a specific entity in a program or the unit of code representing that entity.

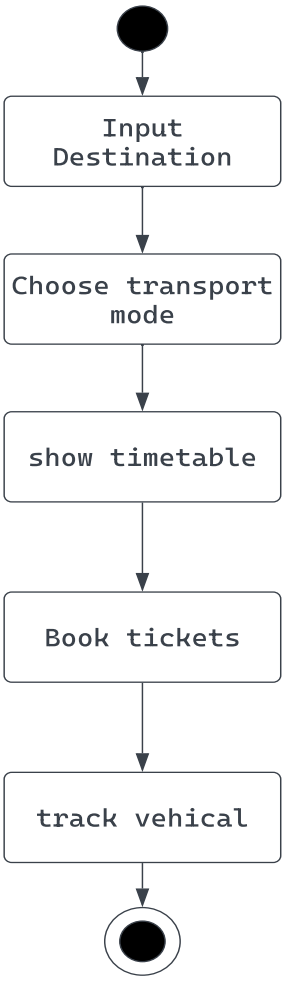


Figure 3.11: State Diagram

3.6.7 Use-Case Diagram

A use case diagram is a type of behavioral diagram by the Unified Mod- eling Language (UML) and created from a Use-case analysis. Its purpose is to present a graphical overview of the functionality provided by a system in terms of actors, their goals (represented as use cases), and any dependencies between those use cases.

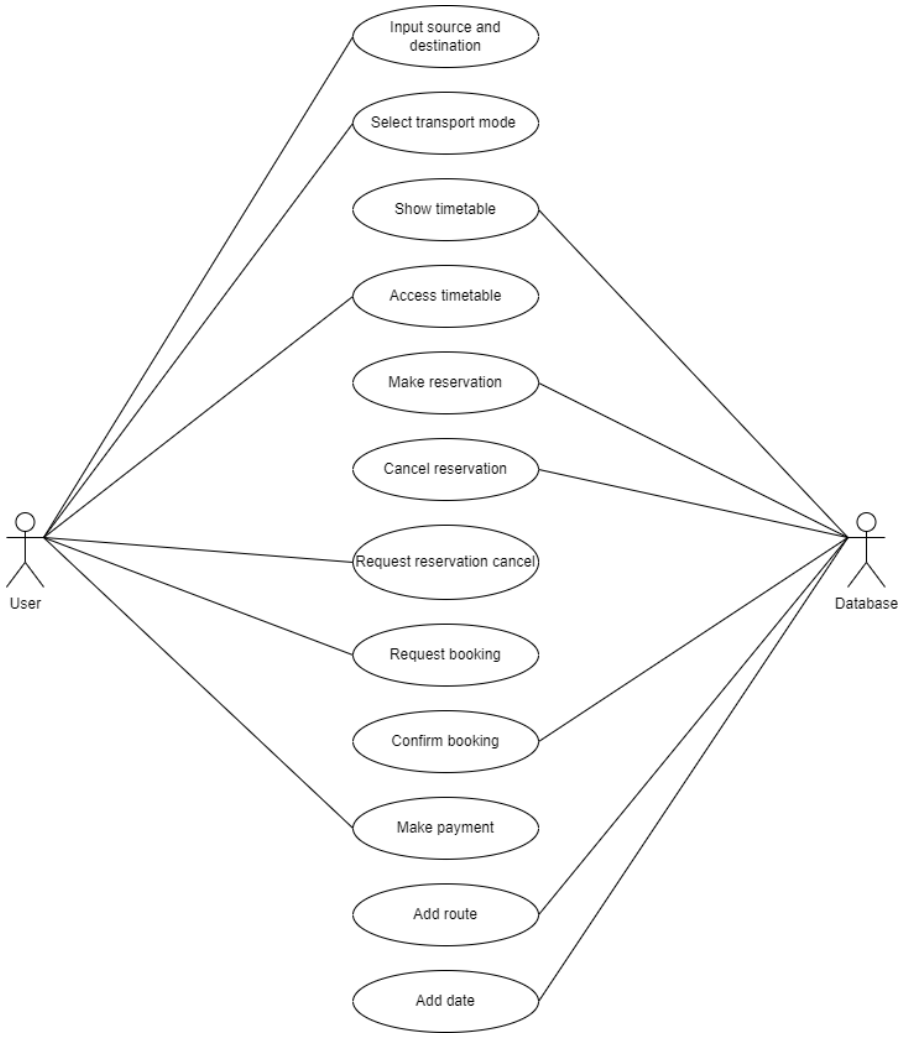


Figure 3.12: Use-Case Diagram

3.6.8 Sequence Diagram

A sequence diagram is an interaction diagram that shows how processes operate with one another and in what order. It is a construct of a Message Sequence Chart. A sequence diagram shows object interactions arranged in a sequence.

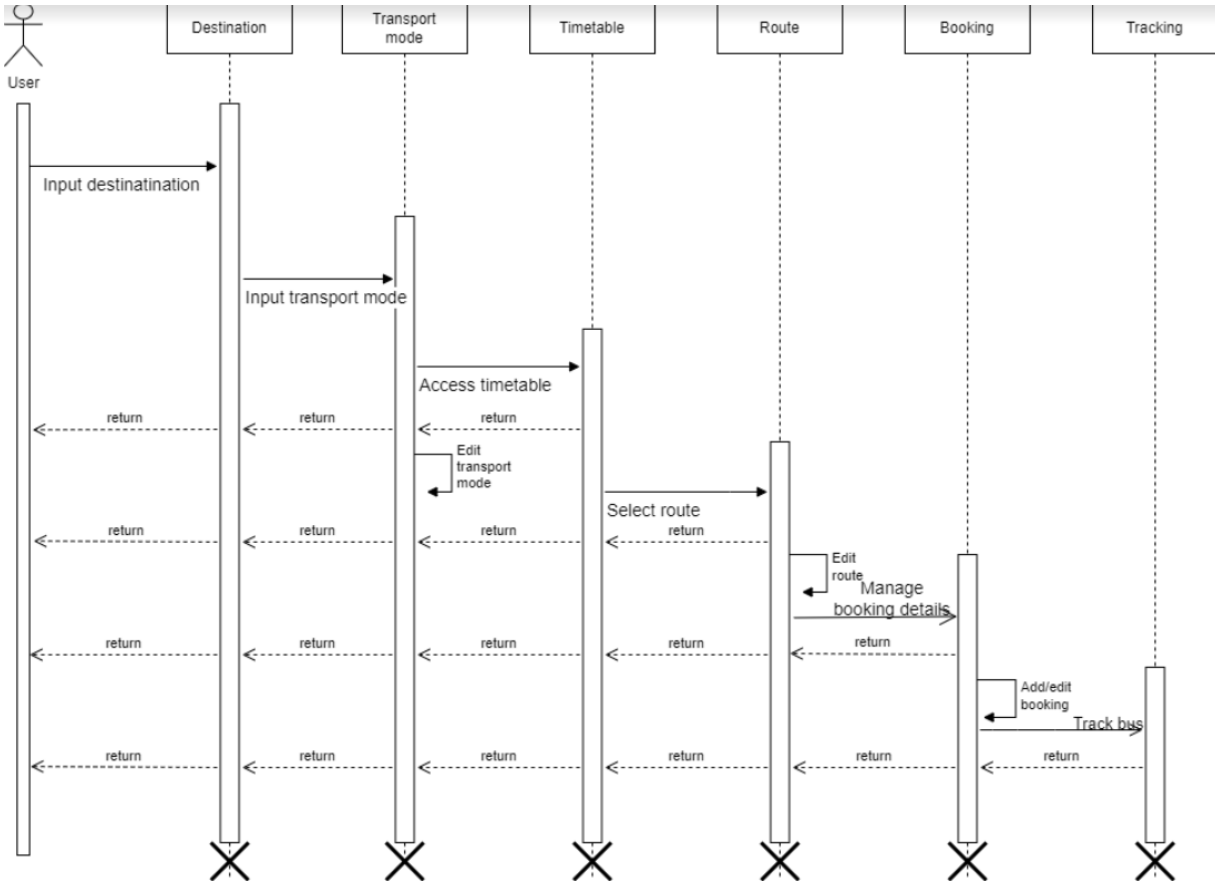


Figure 3.13: Sequence Diagram

Chapter 4

Timeline of Project

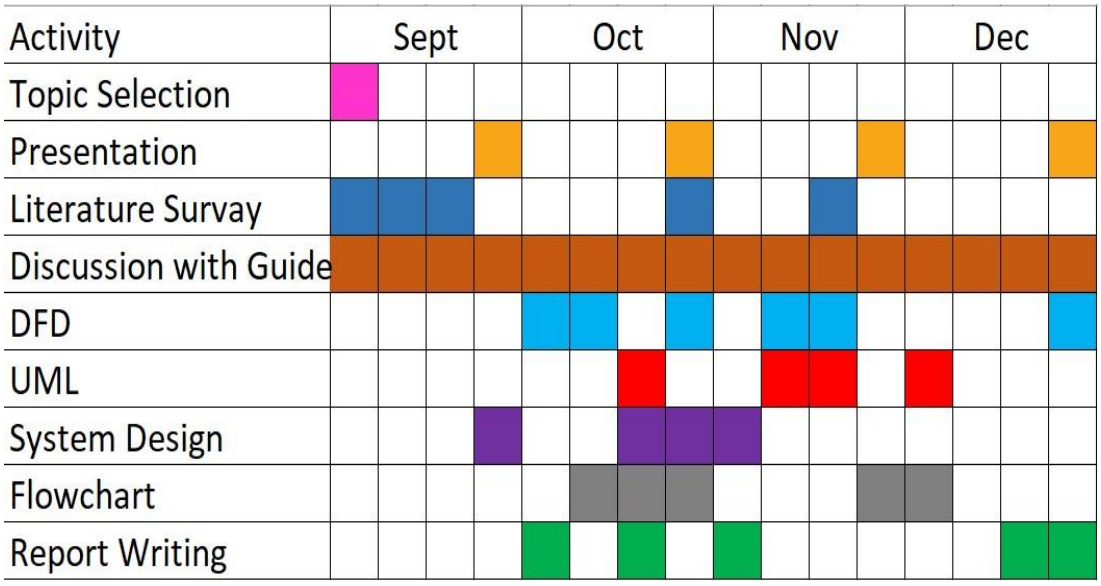


Figure 4.1: Gantt Chart

Chapter 5

Mathematical Model

Mathematical model for Improved Web Scraper System For Smart Online Shopping

$$S = (I, O, F, Success, Failure)$$

Where,
I = Set of Inputs
I = (i1,i2)
i1 = Source
i2 = Destination

O = Set of Outputs

F = Set of Functions F
= (f1,f2,f3)
f1 = Fetch Data Function.
f2 = Display available option
f3 = Ask for other option

Success = Success Case
Success = System gives accurate information.

Failure = Failure Case
Failure = System doesn't respond.

Chapter 6

Testing and Evaluations

Testing is an investigation conducted to provide stakeholders with information about the quality of the product or service under test. Software testing also provides an objective, independent view of the software to allow the business to appreciate and understand the risks of software implementation. Test techniques include, but are not limited to, the process of executing a program or application with the intent of finding software bugs. Software testing can also be stated as the process of validating and verifying that a software program, application, or product meets the business and technical requirements that guided its design and development.

7.1 White Box Testing

White-box test used to test the inner part of the "box", and it focuses on utilizing inside learning of the product to direct the choice of test information. This tests include: structural test, glass-box and clear-box. White box testing is expensive than black box. It takes the source code, before the tests can be. This is signi cantly more like in the assurance of exact information and the assurance if the product is or is not right. This testing is connected just with the product item; it can’t guarantees that the entire speci cation has been executed.

The clear box or WhiteBox name symbolizes the ability to see through the software ’s outer shell (or "box") into its inner workings. Likewise, the "black box" in "Black Box Testing" symbolizes not being able to see the inner workings of the software so that only the end-user experience can be tested.

7.2 Black Box Testing

Black Box Testing is a software testing method in which the functionalities of software applications are tested without having knowledge of internal code structure, implementation details and internal paths. Black Box Testing mainly focuses on input and output of software applications and it is entirely based on software requirements and specifications. It is also known as Behavioral Testing. For Example, an operating system like Windows, a website like Google, a database like Oracle or even your own custom application. Under Black Box Testing, you can test these applications by just focusing on the inputs and outputs without knowing their internal code implementation.

7.3 Test Case Generation

Testing aims at nding errors in a system or program. A set of tests is also called a test suite. Test case generation is the process of generating test suites for a particular system. Model-based Testing (MBT) is a technique to generate test suites for a system from a model describing the system. One usually tries to generate test suite which satis es a given coverage criterion.

7.3.1 Positive Test Cases

Positive testing is the type of testing that can be perform ed on the system by providing the valid data as input. It checks whether system behaves as expected with positive inputs. This test is done to check the system that does what it is supposed to do.

7.3.2 Negative Test Cases

Negative Testing is a variant of testing that can be performed on the system by providing invalid data as input. It checks whether an system behaves as expected with the negative inputs. This is to test the system does not do anything that it is not supposed to do so.

Chapter 8

Conclusion and Future Scope

8.1 Conclusion

As a result of our React Native based system, we conclude our system is better most of the application present currently since it doesn't require connectivity to use the basic features of the application users are provided with a platform where they can locate local bus and train stations, as well as offline timetables and transport tracking systems. This system is intended to help people from remote locations and save them time.

8.2 Future Scope

There are several potential avenues for further development and improvement of a React Native public transport application:

1. Integration with real-time transport data: Currently, most public transport apps rely on static schedule information. By integrating with real-time data feeds from transport agencies, the app could provide more accurate arrival and departure times for buses, trains, and other modes of transport.
2. Personalization and recommendations: The app could use machine learning algorithms to learn the user's travel patterns and preferences, and provide personalized recommendations for the fastest or most convenient routes.
3. Integration with other transportation options: The app could be expanded to include options such as ride-sharing or bike-sharing services, allowing users to compare and choose the best mode of transportation for their needs.
4. Multi-language support: The app could be translated and localized for use in different countries and regions, making it accessible to a wider audience.
5. Integration with smart city infrastructure: The app could be integrated with smart city infrastructure such as traffic lights or parking garages, providing users with real-time information about traffic conditions and available parking spaces.
6. Accessibility features: The app could include features such as voiceover support for users with visual impairments, or support for alternative input methods for users with physical disabilities.
7. Offline support: The app could include offline support, allowing users to access schedule and route information even when they don't have an internet connection

References

.

[1] Chadil, Noppadol, Apirak Russameesawang, and Keeratiwintakorn. Real-Time Tracking Management System Using GPS, GPRS and Google Earth, In IEEE 5th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology, vol. 1,2008 [IEEE]

[2] Mohammad A. Al-Khedher, “Hybrid GPS-GSM Localization of Automobile Tracking system”, International Journal of Computer Science & Information Technology (IJCSIT) Vol 3, No 6, Dec 2011. [IEEE]

[3] Sanam Kazi , Murtuza Bagasrawala, Farheen Shaikh, Anamta Sayyed Smart E-Ticketing System for Public Transport Bus - ©2018 IEEE]

[4] Shubham Jain, Adarsh Trivedi, Shweta Sharma, Bus Tracking System – ©2019.[MDPI]

[5] Ayman R. Mohammed: UML Modeling of Online Public Bus Reservation System in Egypt - ©2020

[6] Prof. M. Chavan, Navarange Prajwal, More Vishal, Nagargoje Shubham: Online Bus reservation System - ©2021