



**NEW HORIZON
COLLEGE OF ENGINEERING**

Autonomous College Permanently Affiliated to VTU, Approved by AICTE & UGC
Accredited by NAAC with 'A' Grade, Accredited by NBA

A MINI PROJECT REPORT

for

Mini Project in Mobile Application Development (20CSE77A)

on

BMTC WALLET

Submitted by

SOLLETI GOWTHAM KUMAR

USN: 1NH18CS747, Sem-Sec: 7-D

In partial fulfillment for the award of the degree of

**BACHLOR OF ENGINEERING
IN
COMPUTER SCIENCE AND ENGINEERING**



**NEW HORIZON
COLLEGE OF ENGINEERING**



Certificate

This is to certify that the mini project work titled

BMTC WALLET

*Submitted in partial fulfillment of the degree of Bachelor of Engineering
in Computer Science and Engineering by*

**SOLLETI GOWTHAM KUMAR
1NH18CS747**

DURING

ODD SEMESTER 2021-2022

for

*Course: Mini Project in Mobile Application
Development-
20CSE77A*

Signature of Reviewer

Signature of HOD

SEMESTER END EXAMINATION

Name of the Examiner

Signature with date

1. _____

2. _____

ABSTRACT

Since the rise of information technology, manual has turned out to be exercise in futility and vitality particularly in the created nations. In Bangalore, the manual system is still functional and needs to be improved. The primary reason for not constructing a robotized system is the projected lack of learning and creativity in the field of information technology. The most important aspect of completing this project is to design and plan a BMTC wallet for an organisation in Bangalore, with the goal of replacing all of their manual tasks or routine operations with a computerized system that will allow the organization to serve its clients to their full satisfaction. The method enables consumers to book tickets with the touch of a button, eliminating the need to be stressed or walk to a counter to purchase a ticket. Once you've completed the registration process and entered all of the required information, you'll be a user with full access to the system. The system has a lot of features, for example, the bottom displays on the application by knowing the routes, knowing the price and discount while booking, and payment is also offered in the system. The system includes an admin section that is responsible for updating the majority of the information needed for the benefit of consumers, as well as allowing users to see all of the information about where each bus is heading and the destination. This project is totally based on digitalizing local transportation. This application is used to freely transfer to the conductor as it also has options like advance booking and QR scan and QR code. The advance booking is used to book the bus in advance and a ticket will generate and the ticket has to show to the conductor once the booking is confirmed. The QR scan is used to scan the QR to transfer the money via google pay and phone pay and others. The QR code is mainly integrated here because there is a chance that a person or passenger can transfer the wrong amount of money. So this QR code is scanned by the conductor and the money will transfer from passenger to conductor, in this way there is no chance of a person can breach the bus. This application is also integrated with chatbot assistance to guide the person or passenger to go through this application. This application is user-friendly and even rural or uneducated people also are able to use it to prevent the contact of covid-19. This application is also integrated with payment gateways which will be transparent to people and the government.

ACKNOWLEDGEMENT

The satisfaction and euphoria that accompany the successful completion of any task would be impossible without the mention of the people who made it possible, whose constant guidance and encouragement crowned our efforts with success.

I have great pleasure in expressing gratitude to **Dr. Mohan Manghnani**, Chairman of New Horizon Educational Institutions for providing necessary infrastructure and creating good environment.

I take this opportunity to express my profound gratitude to **Dr. Manjunatha**, Principal NHCE, for his constant support and encouragement.

I would also like to thank **Dr. B. Rajalakshmi**, Professor and Head, Department of Computer Science and Engineering, for her constant support.

I also express my gratitude to **Manjunatha Swamy**, senior assistant professor Department of Computer Science and Engineering, my mini project reviewer, for constantly monitoring the development of the project and setting up precise deadlines. Her valuable suggestions were the motivating factors in completing the work.

1NH18CS747
SOLLETI GOWTHAM KUMAR

TABLE OF CONTENTS

CHAPTER NO	TITLE	PAGE NO
	ABSTRACT	I
	ACKNOWLEDGEMENT	II
	LIST OF FIGURES	V
1	INTRODUCTION	
1.1	About Android	1
1.2	Android Architecture	2
1.3	About Mini Project	4
1.4	Objective of the mini-project	4
1.5	Advantages of the mini-project	5
2	REQUIREMENT SPECIFICATION	
2.1	Mobile Requirements	6
2.2	Computer Requirements	6
3	ANALYSIS AND DESIGN	
3.1	Algorithm / Pseudo code	7
3.2	Flow chart	8
4	IMPLEMENTATION	
4.1	Android concepts used	9
4.2	Functionality of the project	11
4.3	SQLite Databaseconnectivity	17
5	SAMPLE OUTPUT	19
	CONCLUSION	31
	REFERENCES	32

Table of figures

<i>Fig no</i>	<i>name</i>	<i>Pg no</i>
1.2.1	Android architecture	2
3.2.1	Flow chart	8
4.2.1	Code snippet	11
4.2.2	Code snippet	11
4.2.3	Code snippet	11
4.2.4	Code snippet	12
4.2.5	Code snippet	12
4.2.6	Code snippet	12
4.2.7	Code snippet	13
4.2.8	Code snippet	13
4.2.9	Code snippet	13
4.2.10	Code snippet	14
4.2.11	Code snippet	14
4.2.12	Code snippet	15
4.2.13	Code snippet	15
4.2.14	Code snippet	16
4.2.15	Code snippet	16
4.2.16	Code snippet	17
4.2.17	Code snippet	17
5.1	Output screenshot	18
5.2	Output screenshot	19
5.3	Output screenshot	20
5.4	Output screenshot	21
5.5	Output screenshot	22
5.6	Output screenshot	23
5.7	Output screenshot	24
5.8	Output screenshot	25
5.9	Output screenshot	26
5.10	Output screenshot	27
5.11	Output screenshot	28
5.12	Output screenshot	29

CHAPTER 1

INTRODUCTION

1.1 ANDROID

The Android working framework is a versatile working framework that was created by Google (GOOGL) to be essentially utilized for contact screen gadgets, PDAs, and tablets. Its configuration allows clients to control the cell phones naturally, with finger developments that reflect normal movements, for example, squeezing, swiping, and tapping. Google likewise utilizes Android programming in TVs, vehicles, and wristwatches—every one of which is fitted with a novel UI. The Android working framework is a versatile working framework that was created by Google (GOOGL) to be essentially utilized for contact screen gadgets, phones, and tablets. Its configuration allows clients to control the cell phones instinctively, with finger developments that reflect normal movements, for example, squeezing, swiping, and tapping. Google likewise utilizes Android programming in TVs, vehicles, and wristwatches—every one of which is fitted with a special UI. The versions on android are as follows:

- Android 1.0: Android Alpha
- Android 1.1: Android Beta
- Android 1.5: Android Cupcake
- Android 1.6: Android Donut
- Android 2.0 - 2.1: Android Éclair
- Android 2.2 – 2.2.3: Android Froyo
- Android 2.3.3 – 2.3.7: Android Gingerbread
- Android 3.0 – 3.2.6: Android Honeycomb
- Android 4.0 – 4.0.4: Android Ice Cream Sandwich
- Android 4.1 – 4.3.1: Android Jelly bean
- Android 4.4 – 4.4.4: Android KitKat
- Android 5.0 – 5.1.1: Lollipop
- Android 6.0 – 6.0.1: Android Marshmallow
- Android 7.0 – 7.1.2: Android Nougat
- Android 8.0 – 8.1.0: Android Oreo
- Android 9.0: Android Pie
- Android 10

1.2 ANDROID ARCHITECTURE

Android is architected in the form of a software stack comprising applications, an OS, run-time environment, middleware, services and libraries.

Each layer of the stack, and the corresponding elements within each layer, are tightly integrated and carefully tuned to provide the optimal application development and execution environment for mobile devices.

The Android software stack is a Linux kernel and a collection of C/C++ libraries exposed through an application framework that provides services for, and management of, the run time and applications.

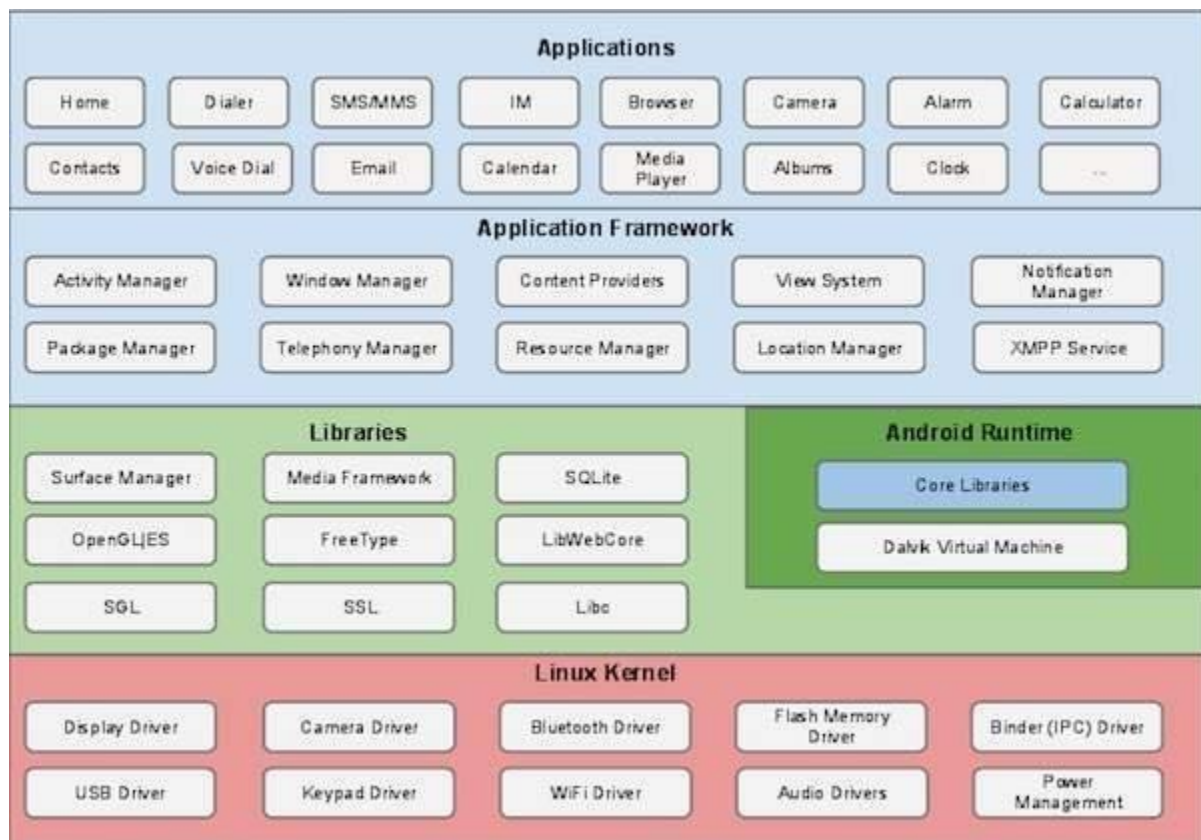


Fig no 1.2.1

The Android software stack is composed of the following: -

- Linux kernel
- Libraries
- Android run time
- Application framework
- Application layer

LINUX KERNEL

At the lower part of the layers is Linux - Linux 3.6 with roughly 115 patches. This gives a degree of reflection between the gadget equipment and it contains all the fundamental equipment drivers like camera, keypad, show and so forth. Additionally, the part handles every one of the things that Linux is great at, for example, organizing and a huge range of gadget drivers, which remove the aggravation from connecting to fringe equipment.

LIBRARIES

On top of Linux portion there is a bunch of libraries including open-source Web program motor WebKit, notable library libc, SQLite information base which is a valuable archive for capacity and sharing of use information, libraries to play and record sound and video, SSL libraries answerable for Internet security and so on.

ANDROID LIBRARIES

This classification envelops those Java-based libraries that are explicit to Android improvement. Instances of libraries in this classification remember the application structure libraries for expansion to those that work with UI building, designs drawing and information base access.

ANDROID RUNTIME

This is the third part of the design and accessible on the second layer from the base. This part gives a key part called Dalvik Virtual Machine which is a sort of Java Virtual Machine uniquely planned and improved for Android.

The Dalvik VM utilizes Linux center highlights like memory the board and multi-stringing, which is natural in the Java language. The Dalvik VM empowers each Android application to run in its own cycle, with its own occasion of the Dalvik virtual machine.

The Android runtime additionally gives a bunch of center libraries which empower Android application designers to compose Android applications utilizing standard Java programming language.

Application Framework

The Application Framework layer gives numerous more significant level administrations to applications as Java classes. Application engineers are permitted to utilize these administrations in their applications.

APPLICATIONS

The Android application is at the top layer. we will write our application to be installed on this layer only. Application components are the essential building blocks of an Android application. These components are loosely coupled by the application manifest file `AndroidManifest.xml` that describes each component of the application and how they interact.

1.3 ABOUT MINI-PROJECT

Nowadays covid is spreading too fast through the exchange of money. So everywhere it is digitalized but some places are not digitalized still now they are local buses. In local transport, they are still using paper money. This project is totally based on digitalizing local transportation. This application is used to freely transfer to the conductor a it also has options like advance booking and QR scan and QR code. The advance booking is used to book the bus in advance and a ticket will generate and the ticket has to show to the conductor once the booking is confirmed. The QR scan is used to scan the QR to transfer the money via google pay and phone pay and others. The QR code is mainly integrated here because there is a chance that a person or passenger can transfer the wrong amount of money. So this QR code is scanned by the conductor and the money will transfer from passenger to conductor, in this way there is no chance of a person can breach the bus. This application is also integrated with chatbot assistance to guide the person or passenger to go through this application. This application is user-friendly and even rural or uneducated people also are able to use it to prevent the contact of covid-19. This application is also integrated with payment gateways which will be transparent to people and the government.

1.4 OBJECTIVES OF THE MINI PROJECT

The primary goal of this project is to develop and design an Android mobile application for local bus booking in order to consolidate all of their everyday activities into one application that will allow them to serve consumers to their full satisfaction. For Bangalore, the suggested system will function as a worldwide network interaction application.

The following are the project's objectives:

To study and evaluate current system difficulties and propose a solution to customers and

management by establishing a stable system that allows individuals to operate in a short amount of time.

To define the important aspects of various components and techniques required for the Android Mobile Application of the BMTC wallet, as well as to allow consumers the power to pick which local bus services to utilise over the internet and to avoid traffic in Bangalore.

The following are some of the fundamental components of a BMTC wallet Android mobile application that delivers increased service to bus operators and customers:

Customer information such as name, address, phone number, and e-mail is collected.

- Advance booking for local buses.
- Chat bot integration.
- Seating chart.
- Payment gateway integrations.
- QR scanner
- QR code for conductors to scan and transfer the money.
- Payment information/credit card authorization.

1.5 ADVANTAGES OF MINIPROJECT

- This project is totally for public use and mainly for local bus passengers.
- This theme of this project is to digitalize everywhere where the paper money is used and another main advantage is that to stop spreading the corona virus through exchange of money.
- They is no use of carrying money in your pockets any more, if you have mobile phone you can transfer the money to anybody using this the application is developed.
- This application is more secured through payment gateways why because it is matter of money.
- This application is user friendly any one can easily understand the UI and its also integrated with chatbot.
- This project is authenticated because this application has a feature that the conductor can directly transfer money from passenger account through qr scan as same as we see in paytm.
- This app also integrated with qr scan to scan the qr code and transfer money through phonepe, google pay.

CHAPTER 2

REQUIREMENT SPECIFICATIONS

2.1 MOBILE REQUIREMENTS

- Device : Android OS Devices (API level 5 & above)
- Display : 320 x 480, 240 x 320 & 480
- OS: Android OS 2.1 & above.

2.2 COMPUTER REQUIREMENTS

- **HARDWARE REQUIREMENTS**
 - RAM: 4GB
 - Processor: Intel core i5
- **SOFTWARE REQUIREMENTS**
 - Programming language: Java
 - Back end: SQL Lite
 - Front end: Android Studio
 - Operating system: Windows 10

CHAPTER 3

ANALYSIS AND DESIGN

3.1 ALGORITHM

Step 1: the main page of the application will display with three options.

Step 2: the three options are admin login, customer login and pay through the conductor.

Step 3: if the user press login to admin, it will directed to admin page where a place to login and signup for admin.

Step 4: the signup for admin is only done when admin will know admin code which is used to signup.

Step 5: After admin login into its account, the various options will display like add upcoming buses, delete, update and view upcoming buses where admin will manage buses.

Step 6: when a customer press login to customer, its display login and signup options where customer will create and login into its account.

Step 7: After customer will login into his account, the page will display view buses option where customer will able to book the tickets in advance it will display all the upcoming local buses.

Step 8: After customer chooses his bus, it will redirect to seat selection page where customer will select its seats.

Step 9: After customer select the seat number it will redirect to payment page.

Step 10: the payment page will display card details and phone number and its is integrated with gateway modules.

Step 11: After the payment the ticket will generate.

Step 12: If user select pay to conductor, it will open QR scan where user can scan phonepe QR code.

3.2 FLOWCHART

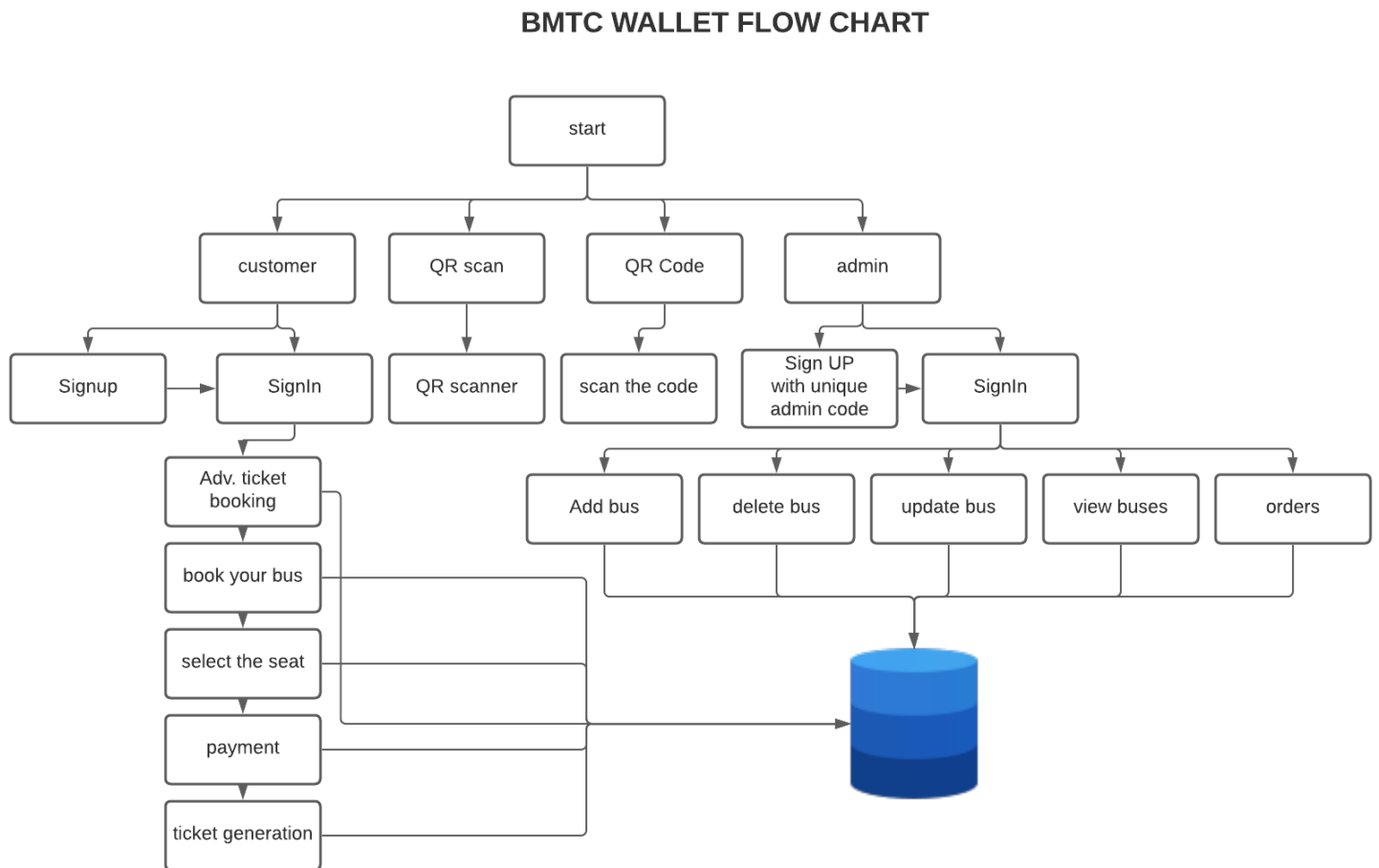


Fig 3.2 : Flow chart

CHAPTER 4

IMPLEMENTATION

4.1 Android concepts used

Apps and APK Files

APK is a file that contains the java code and other resources. An app is nothing but a package in APK. APK stands for Android application package.

Activities

An Activity is a GUI Component which is basically a window in a desktop application. It takes up the whole screen since mobile phone screens are small. Arranging activities side by side is not possible.

Services :

It is the invisible workers of your application. It Runs in the background. It does updating your data sources and Activities, triggering Notifications and broadcasting Intents. It is used to perform long running tasks ,or those that require no user interaction

Views and ViewGroups

A View usually draws something the user can see and interact with.

A ViewGroup is an invisible container that defines the layout structure for View and other ViewGroup objects.

Android GUI elements fall under three categories: Activities, Views and ViewGroups. Activities are the screens / windows. Views are the individual GUI elements, sort of a TextView displaying a text, a Button that users can click on etc. ViewGroups are containers for Views. A ViewGroup groups a collection of Views together

.

Views and ViewGroups are often nested inside an activity or inside a fraction (which is again nested inside an activity)

.

Layouts :

A layout defines what a screen looks like. Layout defines the structure for a user interface. They are used to define the user interface for an Activity. Layouts usually contain GUI components such as buttons and text fields. They are built using a hierarchy of View and ViewGroup objects.

Intents

Android intents are small objects that an activity can pass to the Android operating system, to tell the operating system that some other action or activity is required.

For instance, a photograph app may send an intent to the OS when the user has chosen to share a photo. The intent describes the "sharing action". Other applications that know the way to perform the "sharing action" can then be opened, and therefore the sharing is often performed via another application.

Shared preferences

Shared Preferences are a simple, lightweight key/value pair mechanism for saving primitive application data, most commonly a user's application preferences

Content provider

A content provider manages access to a central repository of data. A provider is part of an Android application, which often provides its own UI for working with the data.

Toast :

Toasts provide simple feedback about an operation in a small popup. It only fills the amount of space required for the message. The current activity remains visible and interactive. It automatically disappears after a timeout.

4.2 Functionality

Home page

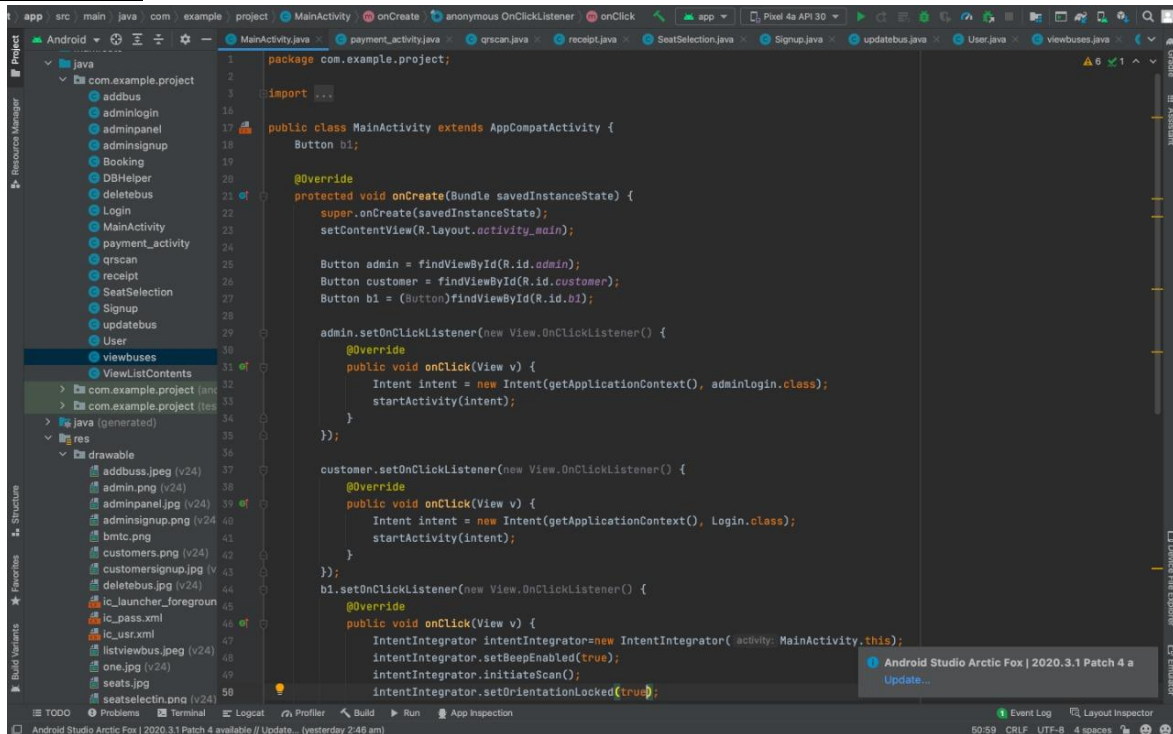


Fig 4.2.1: Main Activity

➔ This is main activity java code where it has buttons and there functions to access to their corresponding functionality, Using intents this page is programmed.

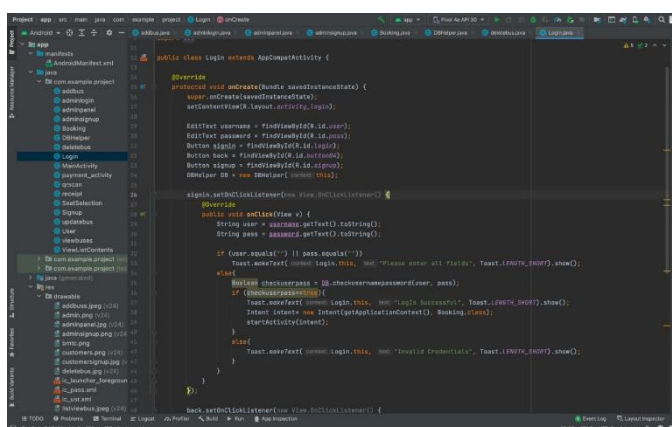


Fig 4.2.2: Login page

➔ This is login functionality code for admin login and customer login.

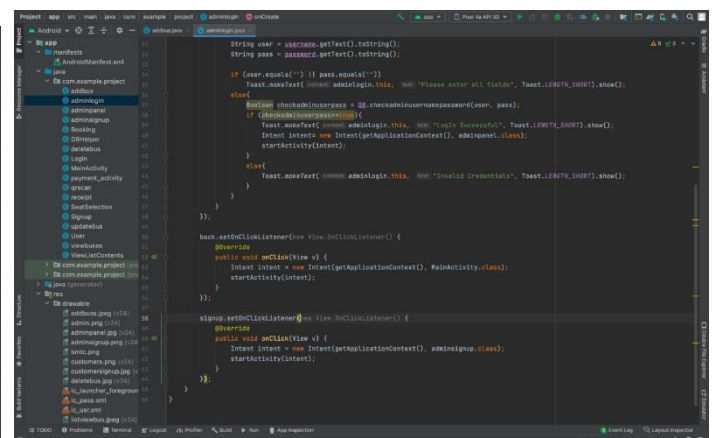


Fig 4.2.3: admin login

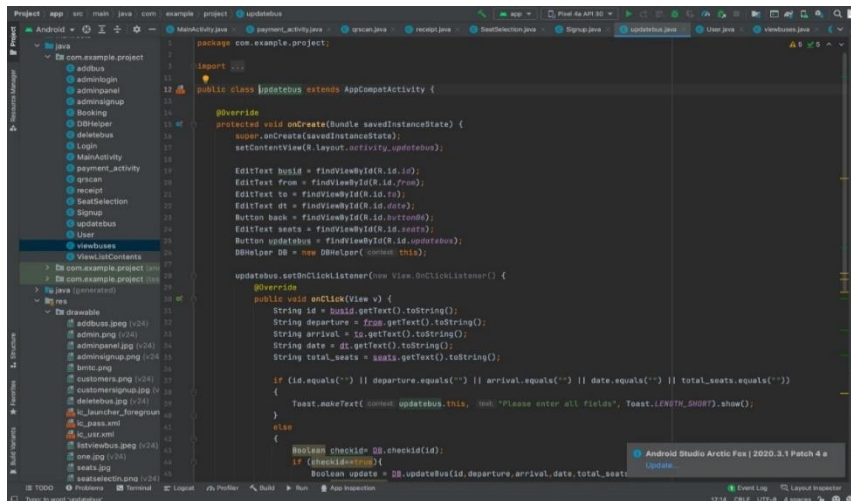


Fig 4.3.4: update page

→ This code of java is for update of bus details and it is only accessed by admin of application.

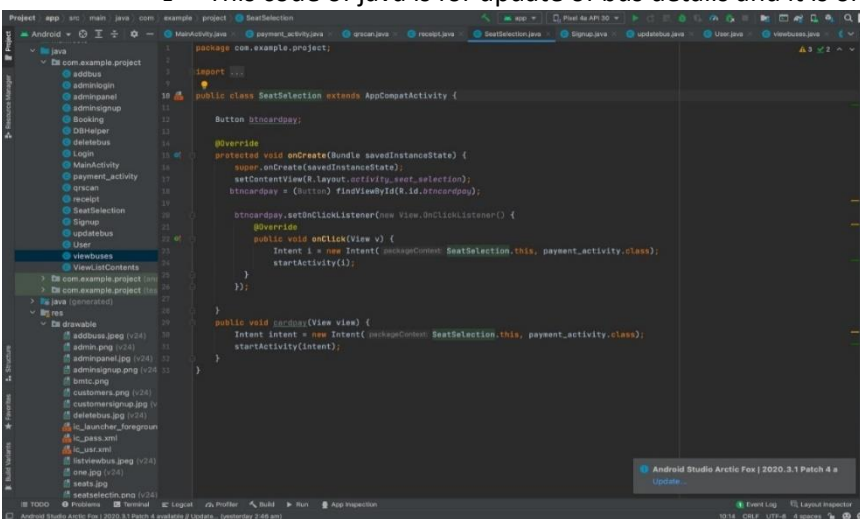


Fig 4.2.5: Seat Selection page

→ This page of java code is for seat selection in bus.

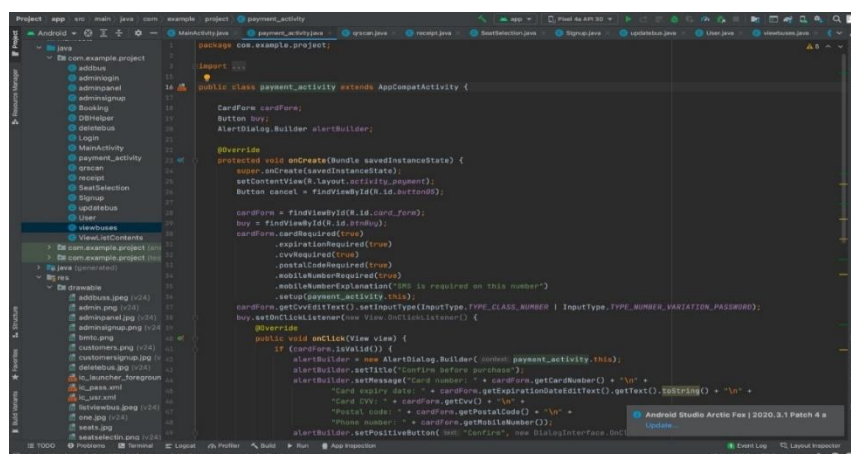


Fig 4.2.6 Payment page

→ this part of java code is used for authenticate payment mode.

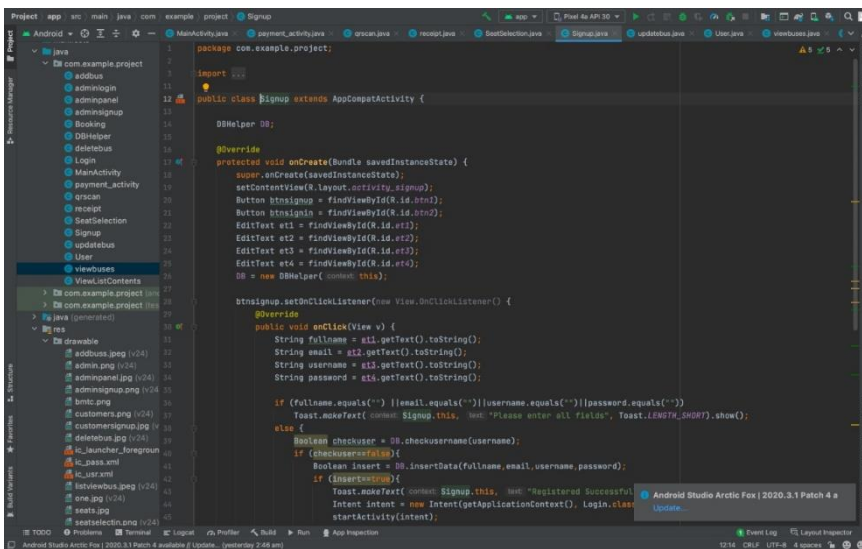


Fig 4.2.7: signup page

➔ this part of java code is used for signup the customer and it has fields like name and age etc.

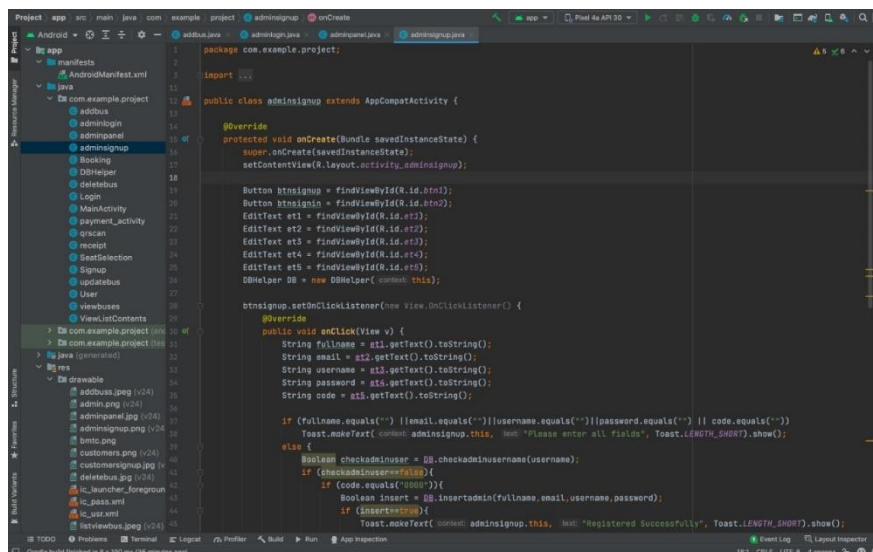


Fig 4.2.8: admin signup

➔ this part of java code is used for signup the admin and it has fields like name and age etc, and It also have admin code which is used to authenticate admin.

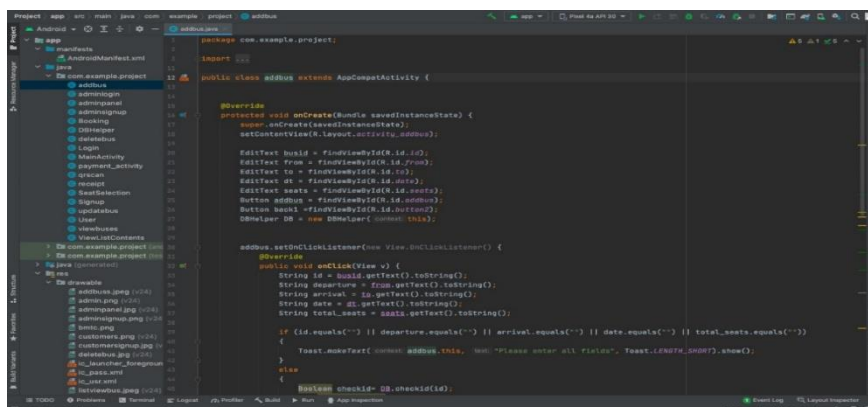


Fig 4.2.9: add bus page

➔ this part of java code is used for adding buses into database.

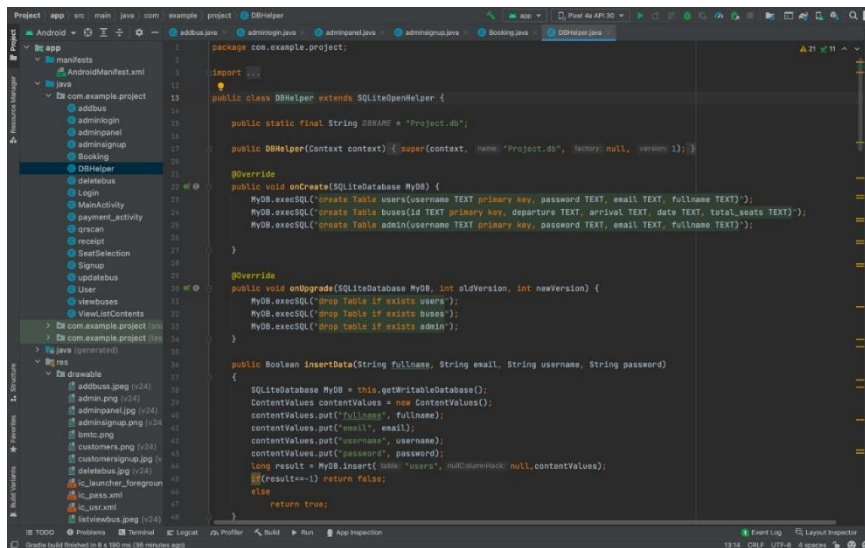


Fig 4.2.10: DB helper

➔ This part of the java code is used to book the tickets in advance .

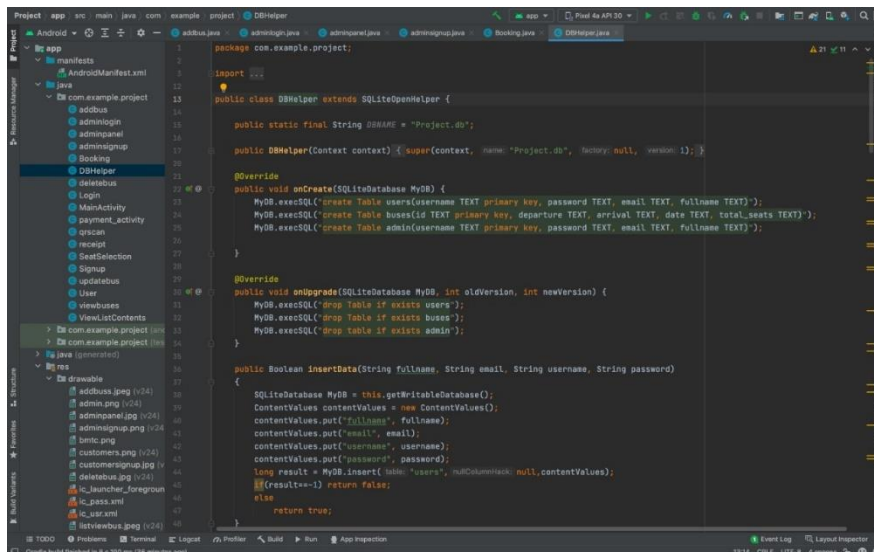


Fig 4.2.11 : sqlite Database

➔ the code is mainly for integration of database to the application.

XML Code:

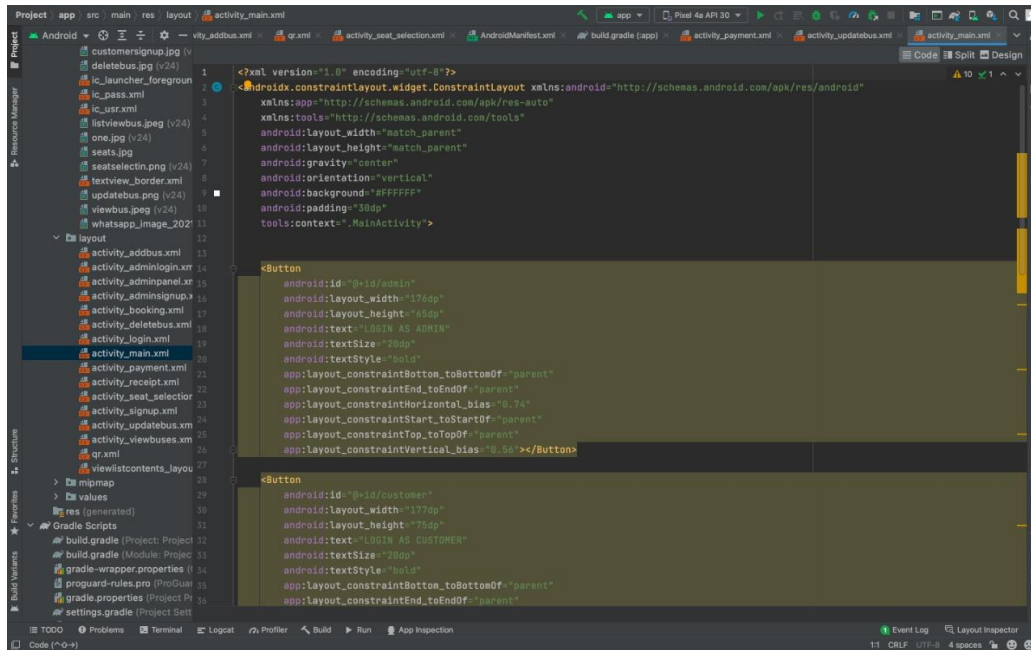


Fig 4.2.12: activity main

➔ This xml code is for starting page of the application where it consist of buttons and images.

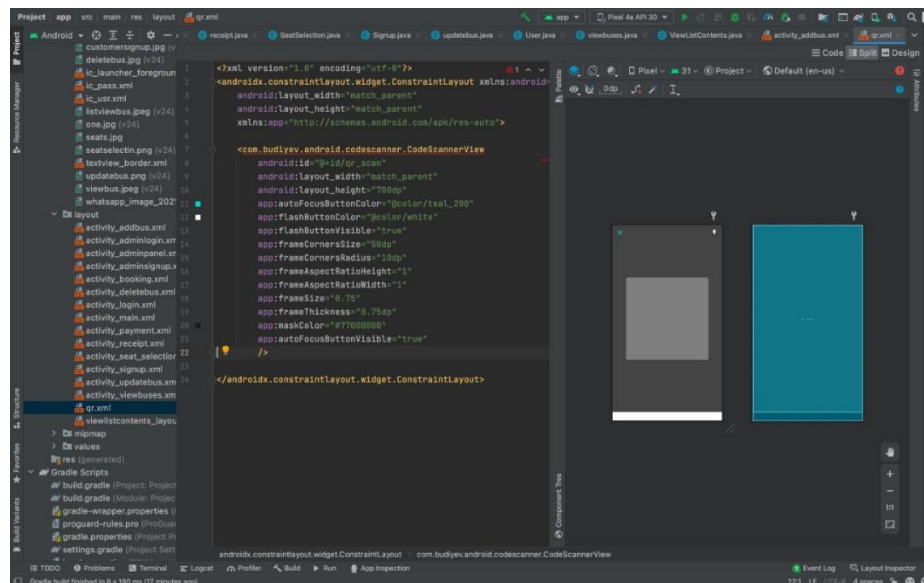


Fig 4.2.13: qr scan

➔ This part of code is for integration of QR scanner.

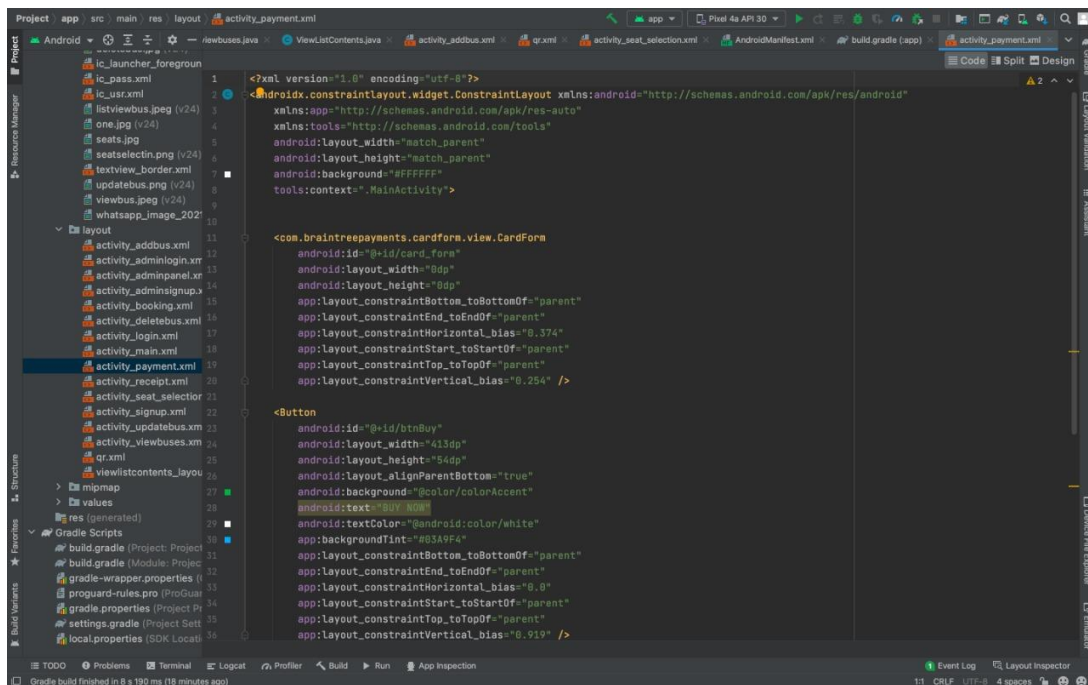


Fig 4.2.14: payment activity

➔ This part of XML code is for integration of payments for advance ticket booking.

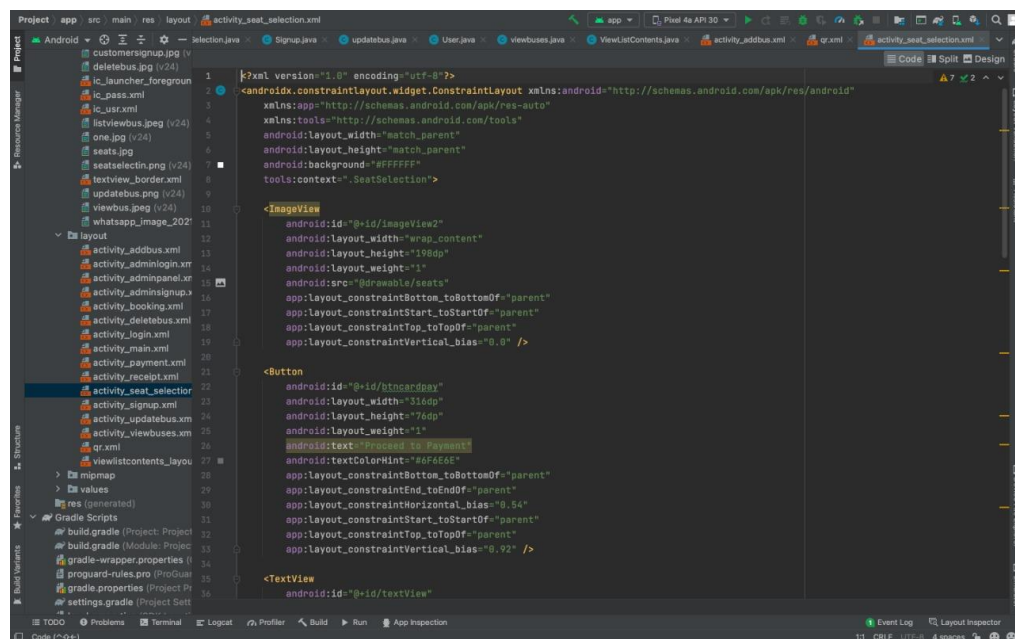


Fig 4.2.15: seat selection activity

➔ This part of XML code is for allocating seats to the customer.

Permissions:

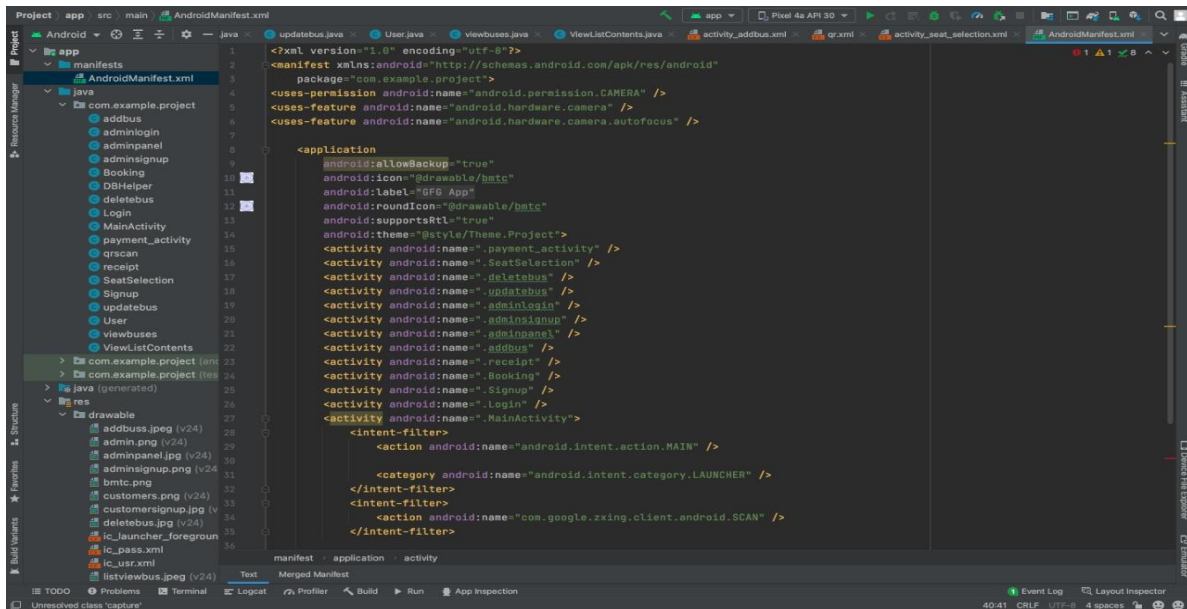


Fig 4.2.16: Android manifest file

➔ This part of code is to add user permissions to access camera into this application to scan the qr code.

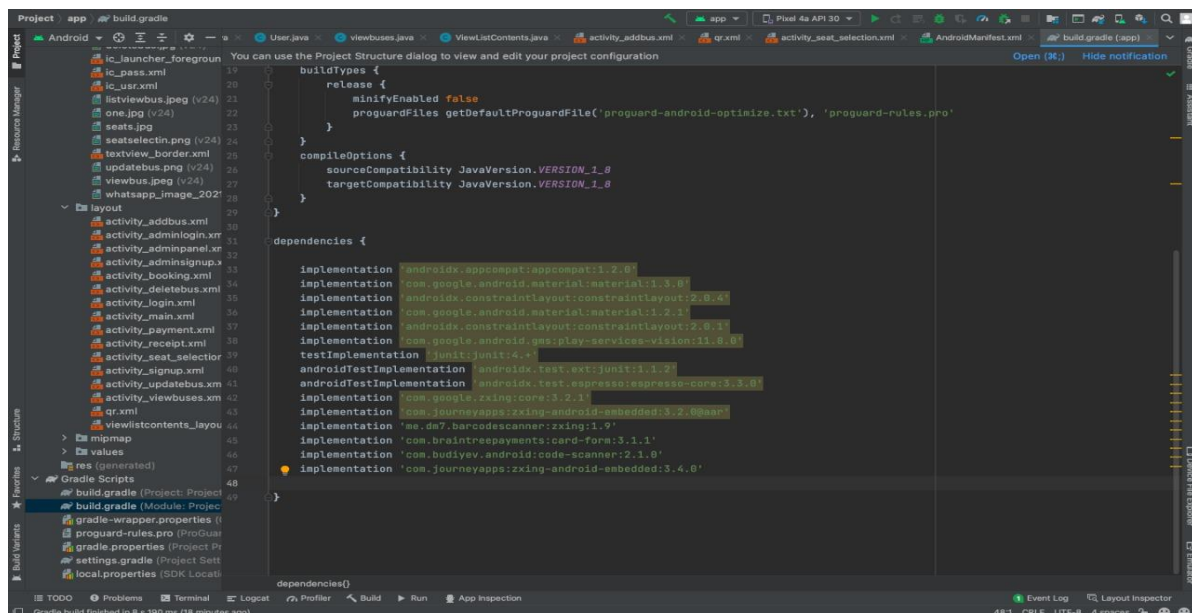


Fig 4.2.17: build gradle

➔ This part of code is gradle file which is used to add dependencies to import qr scanner into the application.

4.3 SQLite Database connectivity

- Using SQLite you can create independent relational databases for your applications.
- SQLite is a well regarded relational database management system (RDBMS)
- Use them to store and manage complex, structured application data
- Android databases are stored in the device manager
/data/data/<package_name>/databases folder on your device (or emulator)
- By default all databases are private, accessible only by the application that created them

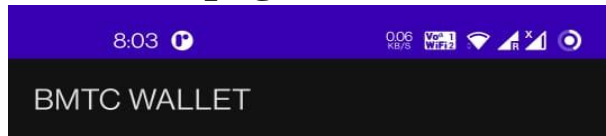
Simple steps to create a database and handle are as follows.

1. Create "SQLite Database" object.
2. Open or Create a database and create a connection.
3. Perform insert, update or delete operation.
4. Create a Cursor to display data from the table of the database.
5. Close the database connectivity.

CHAPTER 5

SAMPLE OUTPUTS

5.1 Home page



→ This is the main home of the application and it is the page where admin and customer can navigate to their pages.



Fig no 5.1 Home page

5.2 Customer pages



8:03

0.00 KB/s

VoWiFi

BMTC WALLET

BACK

SignUp



FullName

Email

Username

Password

SIGNUP

- This page is for registering new user. The user should enter the details asked and login again. The name, age , phone number, address, email, username, password and an emergency contact have to be given . Using this user name and password user have to login

Fig no 5.2 registration form

5.3 AdminPanel page



→ This is the main page of admin in this app . where we will be performing actions like add buses update bus, delete and view buses.

Fig no 5.3 admin panel page

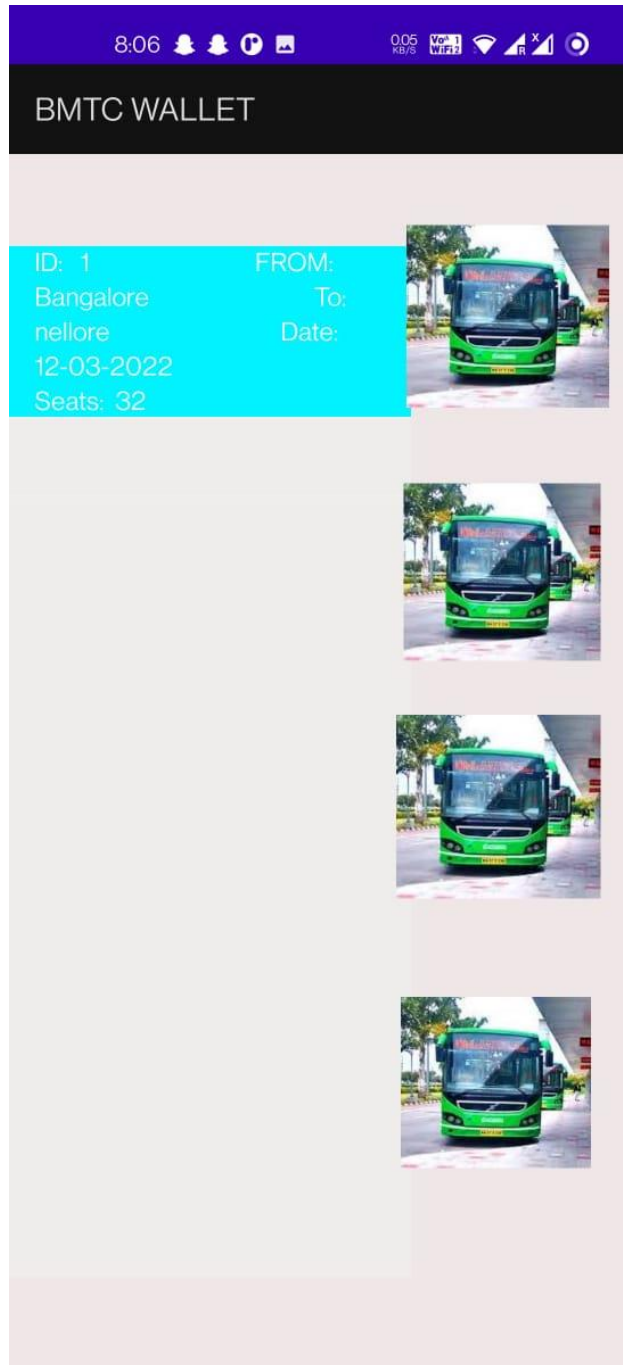
5.4 Customer page



→ This is the customer page after the user login. Here the admin can see view buses and book buses.

Fig no 5.4 Customer page

5.5 View bus page



➔ This is the information page. This gives all information about all the upcoming local buses.

Fig no 5.5 view buses page

5.6 Payment page:

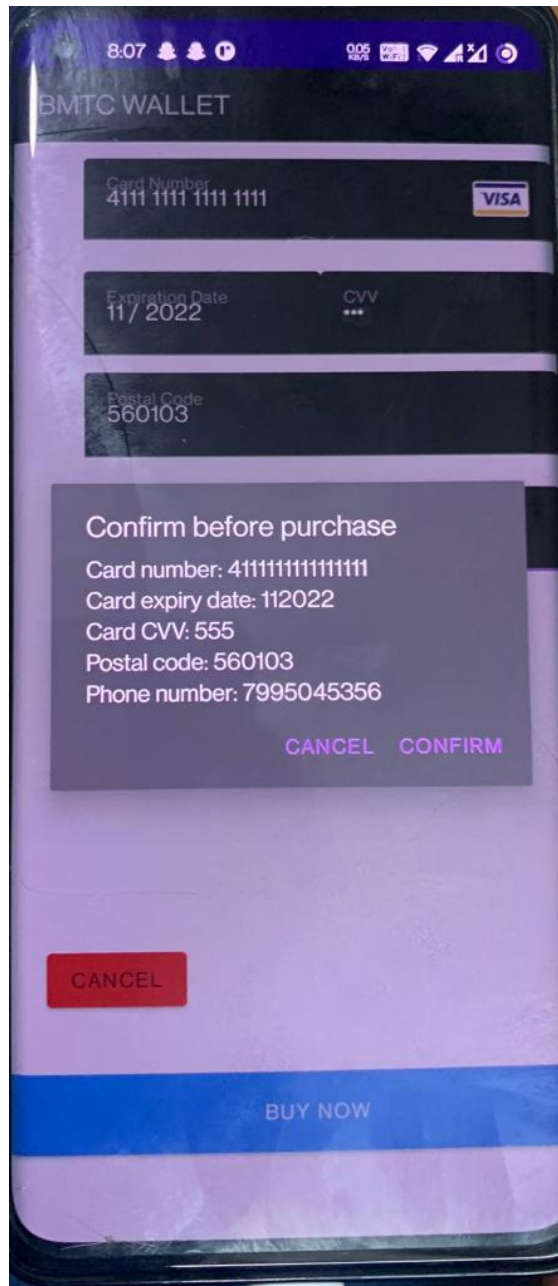
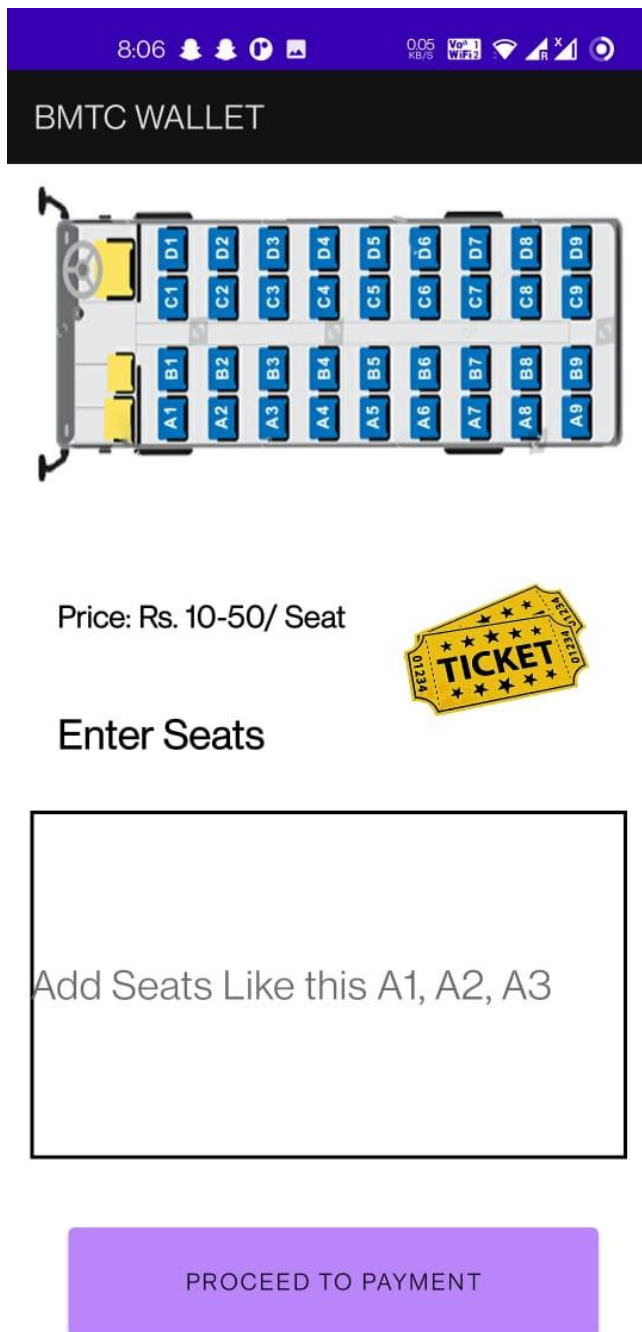


Fig 5.6 payment page

- ➔ This page will display the fields for payment for a ticket and its is authenticated with gateways by using dependency.

5.7 seat information page



- This is page gives information about seats and numbers and their prices and has a button where its show the payment.

Fig no 5.7 Seats information page

5.8 Admin login and SignUp:



BACK

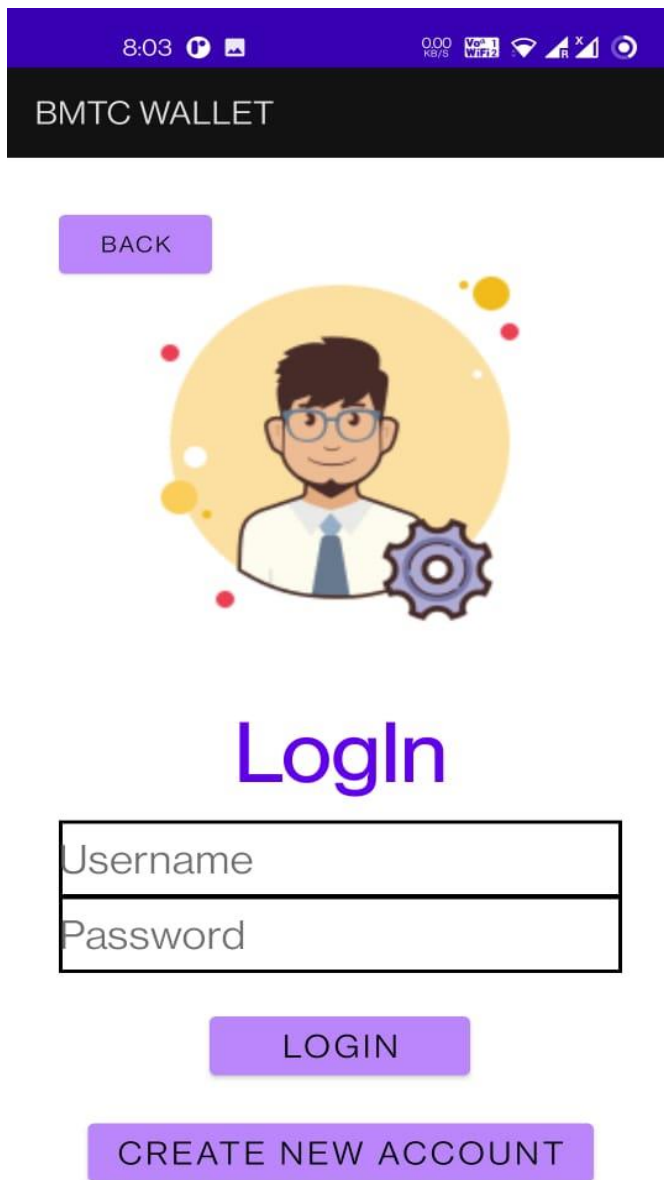


→ this the page where admin can signUp his account.

SignUp

SIGNUP

Fig 5.8 SignUp page



8:03 0.00 KB/s VoWiFi

BMTC WALLET

BACK

Username

Password

LOGIN

CREATE NEW ACCOUNT

Fig 5.9 admin login

➔ This is the page where admin can login to his account.



BACK



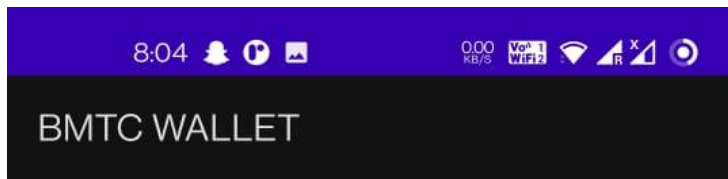
➔ this is page where users can login in to their account.

Login

LOGIN

CREATE NEW ACCOUNT

Fig 5.10: customer login



BACK

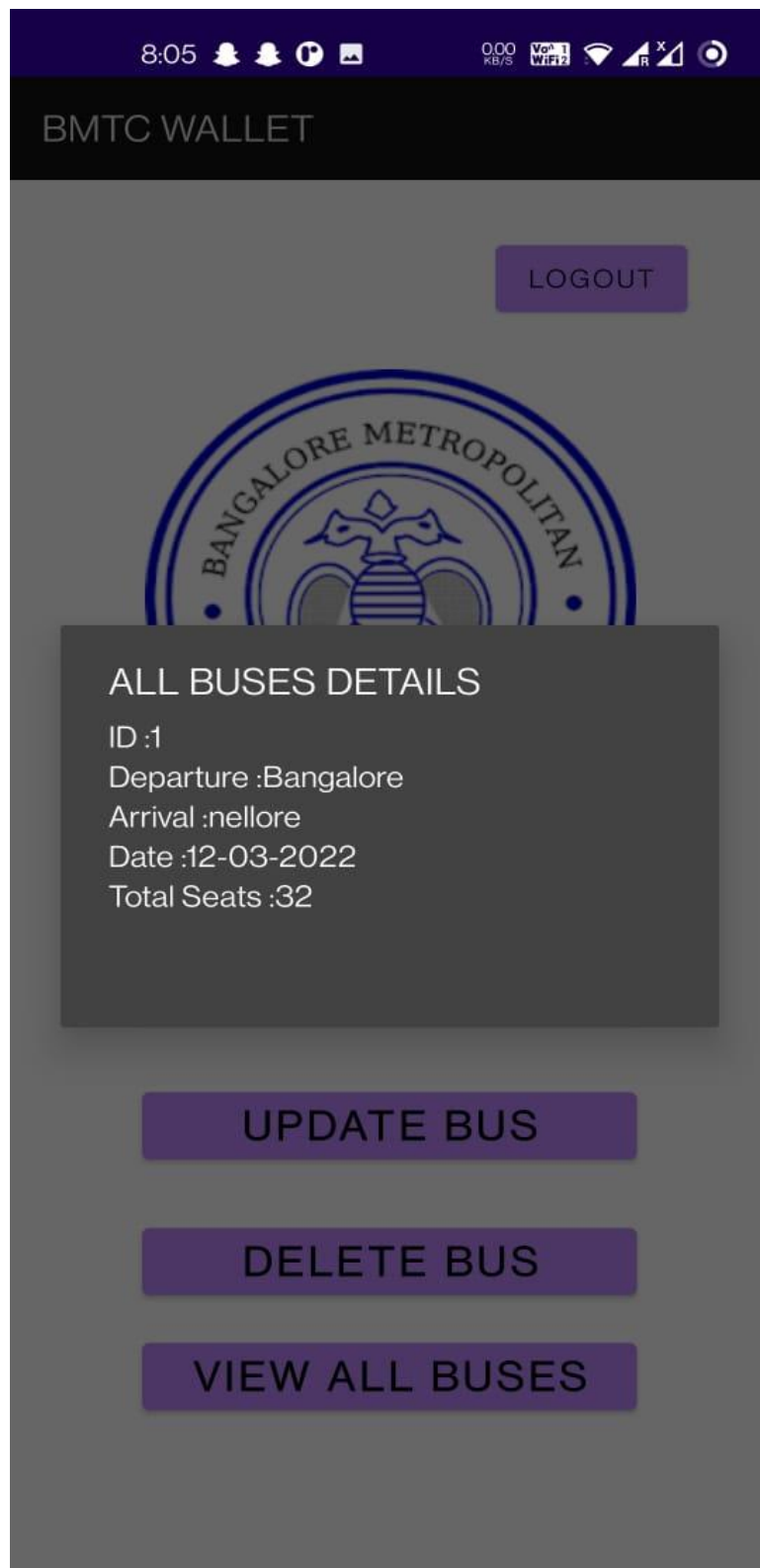


➔ This is the page where admin will add new buses into the upcoming list.

ID
Date
Arrival city
Departure city
Total Seats

ADD BUS

Fig 5.11: add bus page



➔ This is page where admin will view their upcoming buses.

Fig 5.12: View bus page

CHAPTER 5

CONCLUSION

We can finish up by saying that the BMTC Wallet consists of an online bus booking software that was created with the goal of creating an automated system for ticketing purchases through the use of an easy-to-use online booking system. An BMTC Wallet for your mobile application would make it easier to handle reservations, ticket availability, and client data for those who use online bus reservation services. Route scheduling, display of data on seat availability, availability of the seat map for the travellers, and the opportunity for the travellers to pick their seat from the seat map are some of the unique features that a travel management business may add to a bus reservation system. The demand for a well-equipped bus service and huge traffic conditions has grown as the number of travellers has increased with each passing year. When it comes to travelling within the country, bus travel has risen to prominence. The development of travel technology and travel portals has included the implementation of a bmtc wallet. With the integration of The QR scan is used to scan the QR to transfer the money via google pay and phone pay and others. The QR code is mainly integrated here because there is a chance that a person or passenger can transfer the wrong amount of money. So this QR code is scanned by the conductor and the money will transfer from passenger to conductor, in this way there is no chance of a person can breach the bus. This application is also integrated with chatbot assistance to guide the person or passenger to go through this application. This make this application to stand of the crowd.

REFERENCE

- [1] Wikipedia.org. (2019). Wikipedia. [online] Available at: <https://www.wikipedia.org> [Accessed 11 Nov. 2019].
- [2] GeeksforGeeks. (2019). GeeksforGeeks | A computer science portal for geeks. [online] Available at: <https://www.geeksforgeeks.org> [Accessed 11 Nov. 2019].
- [3] Tutorialspoint.com. (2019). H2O, Colab, Theano, Flutter, KNime, Mean.js, Weka, Solidity, Org.Json, AWS QuickSight, JSON.Simple, Jackson Annotations, Passay, Boon, MuleSoft, Nagios, Matplotlib, Java NIO, PyTorch, SLF4J, Parallax Scrolling, Java Cryptography. [online] Available at: <https://www.tutorialspoint.com> [Accessed 11 Nov. 2019].
- [4] Stack Overflow. (2019). Stack Overflow - Where Developers Learn, Share, & Build Careers. [online] Available at: <https://stackoverflow.com> [Accessed 11 Nov. 2019].
- [5] Udemy. (2019). Online Courses - Learn Anything, On Your Schedule | Udemy. [online] Available at: <https://www.udemy.com> [Accessed 11 Nov. 2019].