

Smart Public Transportation System in the prospective of
Bangladesh using Android Application and RFID
Technology



A Thesis Submitted in Partial Fulfillment of the Requirements
for the Degree of

Bachelor of Science in Electrical and Electronic Engineering
By

Mohammed Adnan Uddin-12121100

Asif Faiyaz Bithil -12221045

Abed Al Shihab-13121059

Mashrah Bin Hyder-12121089

Supervised by

Dr. Md. Khalilur Rhaman

Associate Professor
Department of Computer Science and Engineering

Spring 2017

BRAC University, Dhaka

DECLARATION

We hereby declare that, the thesis titled “Smart Public Transportation System In The prospective Of Bangladesh Using Android Application and RFID Technology” is our work. The work has not been presented elsewhere for assessment. Where material has been used from other sources it has been properly acknowledged.

Signature of Author

\

.....
Mohammed Adnan Uddin
Student ID: 12121100

.....
Asif Faiyaz Bithil
Student ID: 12221045

.....
Abed Al Shihab
Student ID: 13121059

.....
Mashrah Bin Hyder
Student ID: 12121089

Signature of Supervisor

.....

Dr. Md. Khalilur Rhaman

Associate Professor

Department of Computer Science and Engineering

BRAC University, Dhaka.

ACKNOWLEDGEMENTS

Firstly, We would like to thank **Dr. Md. Khalilur Rhaman**, Associate Professor, Department of Computer Science And Engineering, BRAC University, Dhaka; for his supportive guidance and feedbacks for completion of the thesis.

ABSTRACT

In this project we have implemented the model of a Smart Public Bus Transportation System that will ease the hassle of women and senior citizens of our country, as we observed they face so many difficulties in their day to day life while using the Public Bus transportation. So we came up with a unique idea to ease their problem by technology. In this project, we tried to implement a model for Smart Public Transportation System In the prospective Of Bangladesh Using Android Application and RFID Technology. Moreover, we will give the customers a very easy way to book the seat from the home and trace the bus where is it coming from. Every will be provided the RFID tag to enter the bus and administrator also can check their revenue has been cut from RFID TAGS.

Our aim is to make a smart bus communication system which will be very easier to buy a ticket from home using our android app. In addition, this project focuses on a GPS tracking which will trace the bus location. We also used radio frequency identification (RFID) technology for user identification for entering the bus here. The proposed RFID system uses tags that are mounted on the door of the bus, through which information embedded on the tags are read by RFID readers; the proposed system will take the user identification and the administration to manually perform to enter into the bus and confirmation of the ticket, respectively. Data information is also easily exchanged between the users and toll administrator for giving the virtual money to the user.

Contents

ACKNOWLEDGEMENTS	3
ABSTRACT.....	4
LIST OF FIGURES	8
List of Table	9
ABBREVIATIONS	10
Introduction	11
1.1 Introduction to development of inter-city smart bus transportation system:.....	11
1.2 Problems with the Traditional system:.....	12
1.3. Advantages of Smart Public Transportation system:	14
1.4 Motivation and Objectives.....	15
1.4.1 Motivation:.....	15
1.4.2 Objective:.....	16
1.5 Background of the Project:	17
1.6 Different types of Bus Communication System:	18
1.6.1 Computer Based Bus Communication System:	18
1.6.2 Android Based Bus Communication System:.....	19
1.7 Different types of payment:	20
1.7.1App Based Payment:.....	20
1.7.2 RFID Based Payment:.....	21
1.7.3 Mobile Operator Based:	22
1.8 Summary of Our Project:	23
Project Overview	24
2.1 Application Area of the Project.....	24
2.2 Bus Ticket System	26
2.3 Working principle of our project.....	27
Hardware Components.....	30
3.1 Introduction	30
3.2 Hardware components	31
3.2.1Arduino UNO R3.....	32
3.2.2 Bluetooth Module HC-05	33

3.2.3 RC-522 13.56 MHz RFID Reader	34
3.2.4 GPS enabled Smartphone	36
3.2.5 JUMPER WIRE.....	37
3.2.6 9v Battery	37
3.3 Software Components	38
3.3.1 Arduino IDE	38
3.3.2 Android Studio	39
Proto type Mechanism and Algorithm.....	41
4.1 Mechanism and Algorithm.....	41
4.1.1 System Design:	41
4.1.2 BUS LOCATOR APPLICATION	43
4.1.3 FOR USER END APPLICATION	45
4.1.4 Algorithm.....	48
4.2 Working procedure of proto type:.....	50
4.2.1 For BUS Locator Application	50
4.2.2 For BUS Ticket Application	50
4.3 Display results:	51
4.3.1 For BUS Application	51
4.3.2 For User End Application	52
Economic Overview	53
5.1. Economic losses in present public bus transportation system:.....	53
5.2. Proposed transportation system economic benefits:.....	54
Plan and Budget	55
6.1. Plan for proto type.....	55
6.1.1. Timeline of proto type	55
6.1.2. Budget for proto type	56
Conclusion and Future Ideas.....	57
7.1 Conclusion:.....	57
7.2 Future Ideas:	59
7.2.1 Implementing speed meter:	59
7.2.2 Face detectors:.....	60
7.2.3 Using Counters:.....	60

Reference	62
Other Reference Links.....	63
Appendix	64
Appendix A.....	64
Arduino Code:	64
Appendix B	66
BUS LOCATOR.....	66
Appendix C	74
BUS TICKET BOOKING	74

LIST OF FIGURES

Figure 1.1: People are hanging onto bus door at a very risky way	12
Figure 1.2: Huge crowd, without any proper seat for women.	13
Figure 1.3: Huge crowd, without any proper seat for women.	13
Figure 1.4: Troubling in normal buses	15
Figure 1.5: No privacy for women	16
Figure 1.6: Bus Tracking App	18
Figure 1.7: GPS tracking	19
Figure 1.8: Vehicle Communication Interface	19
Figure 1.9: App baased payment	20
Figure 1.10: RFID Based Payment	21
Figure 1.11: Payment System	22
Figure 1.12: Search BUS From Map	23
Figure 2.1: Inconvenience of Women to get up in bus	25
Figure 2.2: No privacy for women	27
Figure 2.3: Components of our project	28
Figure 2.4: Seat plan & Bus Position	29
Figure 3.0: Schematic Diagram	30
Figure 3.1: Arduino UNO R3	32
Figure 3.2: Bluetooth Module HC-05	33
Figure 3.3: RC-522 13.56 MHz RFID Reader	34
Figure 3.4: RC-522 13.56 MHz RFID Reader flow chart	35
Figure 3.5 : GPS Enabled Smart Android Phone	36
Figure 3.6: Jumper Wire	37
Figure 3.7: 9v Battery	37
Figure 4.1: Bus locator Device	42
Figure 4.2: Bus Locator Application and Location Service Started	43
Figure 4.3: For RFID POP up Window & INPUT RFID TAG & Ticket Confirmation	44
Figure 4.4: BUY Bus Ticket & User Login	45
Figure 4.5: USER Account Balance & Search for Bus	46
Figure 4.6: Bus Position & Seat Plan	47
Figure 4.7: Bus Seat Booking Application Flow Chart	48
Figure 4.8: For Bus End Application	49
Figure 4.9: BUS Locating	50
Figure 4.10: RFID detected	51
Figure 4.11: SEAT BOOKED	52
Figure 7.1: Public transportation for women	58
Figure 7.2: Speed Meter	59
Figure 7.3: Face detection System	60
Figure 7.4: weight sensor	61

List of Table

Table 6.1: Timeline of proto type.....	57
Table 6.2: Budget for proto type.....	58

ABBREVIATIONS

- RFID: Radio Frequency Identification
- GPS: The Global Positioning System
- IDE: Integrated Development Environment
- APK: Android Application Package File
- BRTA: Bangladesh Road Transport Authority
- VCI: Vehicle Communication Interface
- ICSP: Internet And Computer Service Provider
- PWM: Pulse Width Modulation
- EDR: Enhance Data Rate
- AFH: Adaptive Frequency Hopping Feature
- PIO: Programmed Input /Output
- UART: Universal Asynchronous Receiver Transmitter
- SPI: Serial Peripheral Interface
- UNFPA: United Nations Fund for Population Activities

Chapter 1

Introduction

1.1 Introduction to development of inter-city smart bus transportation system:

Dhaka City presently is one of the 10th largest mega-cities of the world with a population of about 14.0 million and having the highest annual growth rate and is expected to be the second largest city of the world with a population of 22.8 million by 2015 (UNFPA, 2001). The city represents less than 1% of the country's total land area supporting about 10% of the population. There has been a phenomenal growth in terms of population and area in the last four decades. Annual growth rate of Bangladesh's population was 1.8 percent per year between 1998 and 2005 but Dhaka's population has increased at an astonishing average annual rate of 9.1 percent since 1999 (World Bank, 2007). Such a huge population is expected to generate a commensurable number of trips each day. For example, population of greater Dhaka (urban areas of Dhaka) is expected to be 36.0 million by 2024 with estimated total 70 million person trips a day (STP, 2005).

The public transportation system is one of the most important factor yet the biggest problems in our densely populated Bangladesh. People find it very hard to find using the public transportation especially the metro buses. They have to stand in a long queue to get into their desired bus without having a confirmation that will they a have a seat to sit or not. Moreover, this is a great hassle especially for the working women or senior citizens in our country.

Thus, these problems actually help us to ignite an idea how to solve these and how to make the public transports more convenient especially for our working women and senior citizens of our country. We have targeted the Metro Bus services as large number of people use this public transport for their inter-city communication.

The main idea is that, we build an android application from where people can book tickets from anywhere, don't have to stand in long queue and they can pay the fair from that application and from the RIFD card that will be provided to them. From this project, especially women and

senior citizens will be benefited in a great way. They will have a hassle less journey to their destination.

1.2 Problems with the Traditional system:

The traditional is very much hassle for the people, they have to stand in long queue for desired bus to come. And it's very much tough to get a seat in the bus.

- More Time Consuming.
- It's very much hard for senior citizens to stand in a long queue.
- It's very risky to get up in bus in rush.
- Chances of accidents are high.
- No proper seat management for women and senior citizens



1.1 Figure: People are hanging onto bus door at a very risky way



1.2 Figure: Huge crowd, without any proper seat for women.



1.3 Figure: Huge crowd, without any proper seat for women.

1.3. Advantages of Smart Public Transportation system:

As our project carries some advantages in transportation system, some of them are described below;

- **Proper seat management and convenient:** This system ensures a proper seat management for people, especially for women and senior citizens. They can book their seat from their home and can see the location of the bus, can get out from the home accordingly. Thus this system ensures seats without doing any rush.
- **Efficiency in time:** This system is more efficient than the traditional way, people already have booked their from tickets from the system, that's why the bus do not have to wait for the people to get in and people don't have wait in long queue to get in.
- **Revenue Collection will be easier:** This integrated system has a revenue collection system through virtual money and RFID card. So the administrator can be benefitted by this system because there is no chance of fraud in revenue collection.
- **Less accidental issues:** By using this system, people don't have to get in bus in a rush or in risk; they will just book their seat from the android application. If they are no seats available they have to wait for the next bus or they will choose the next bus according to their destination.

1.4 Motivation and Objectives:

Surely, we had some motivations and clear objectives to build this system. They are described below:

1.4.1 Motivation:

The struggling of the people to catch a bus to ride for reaching their destiny motivated us strongly. In our country which is the most densely populated in the world and in where **17.6%** people of the population were found to be under the poverty line and almost 40% of the population are classified as poor class. To travel they use the cheapest way what they find is buses but transportations are not that much of the population. A rush of competition occurs to catch a bus falling others apart. This is not the end. Another misery starts when they are to pay the fare. Bus authority takes the fare according to their wishes. Moreover, women who are a major part of our nation face the most miserable situation while catching a bus as they cannot compete with the men. As a result they have to wait a long time waiting for another bus's arrival. Considering all of these scenarios of our daily life we decided to implement something unique that can eradicate all these odd stories so that people do not hurry to get a bus, women do not feel embarrassment, establishment of accountability and most importantly a comfortable journey.



Figure 1.4: Troubling in normal buses



Figure 1.5: No privacy for women

1.4.2 Objective:

- **Introducing an smart system to travel by bus:**

We made up our mind to innovate such an app that is much smart to be accessed. Since all the sectors of our country are becoming digitalized therefore, we also wanted our project to be something like digital so that who uses it feel comfortable and easy. We made up all the features of the app like this way so that user can cope up with that in the first use.

- **To save the valuable time of the people what they waste while waiting for buses:**

Wasting time means the wastage of money. In our country, due to the shortage of the transports people have to wait a long time to get the entrance of a bus. This took a lot of time but nothing to do unless a smart system is built up. So we think that the system that we tried to implement will save the time in a large scale. Here, people do not need to go to the stoppage for the bus to catch. He can easily monitor the latest location of the bus he wished to travel by using the app. Thus the time is saved which was supposed to be wasted if he had to wait at a stoppage for the bus.

- **To ensure the accountability:**

It is a very common scenario to witness the bargaining between the passenger and the payment collector. Sometimes, payment collector demands more money from the Passengers even if it is illegal or sometimes passengers do not want to pay the exact money. Hence, bargaining starts

which ends with a fixed maximum time. In this regard, we tried to ensure the accountability of both of the passengers and the payment collectors.

1.5 Background of the Project:

Transport infrastructure is one of the most important factors for a country's progress. It is said that the progress of a country is dependent largely on the developed transportation system of that particular country. Transportation is important because it facilitates trade, exchange and travel. Without effective transportation, regions are largely isolated from each other. Effective, affordable transportation also plays a role in letting people move to new areas. Because of its intensive use of infrastructures, the transport sector is an important component of the economy and a common tool used for development. This is even more so in a global economy where economic opportunities have been increasingly related to the mobility of people, goods and information. A relation between the quantity and quality of transport infrastructure and the level of economic development is apparent. Development is related to improving the welfare of a society through improving the system of transportation smoothly. Without improving it Government cannot ensure a better standard of living to the nation. In addition, the economic condition of a country closely related with it. It is apparent to us that the importance of having a good transport system or developing it in a good way is much more important than many of things that the government does for the welfare of the people. If we look at the developed countries like Germany, USA, Japan, they first developed their systems of transportation then they went for the next. There are almost 250000 vehicles for the large population in our country. From this it can be clearly described the miserable situation of the people. That is why we come out with our ideas and tried to implement it using our learnt methods and concepts. In our project we innovated an app that is designed with bus selection, choosing the route and also booking the seat as well. We designed two ways of procedures for the payment. The first one is the RFID based and the second one is the app based using database. We tried to give the best efforts we could to develop the project.



Figure 1.6: Bus Tracking App

1.6 Different types of Bus Communication System:

Before going to description of our project, here we are giving some insights about other digital system for the public transportation that are used in the around the world

1.6.1 Computer Based Bus Communication System:

Various buses can communicate with other buses and of course with the customers by using computers. It is to mention that all the computer has to have internet connection at all time. Customers can see the location of the buses they desire to choose and the vacant seats of that particular bus at the same time. At the same time bus can monitor the passengers if they book it or not. This is how the communication occurs by using computers. The computers that are used here are connected to each other and every seconds each of them is sending and receiving data from database.

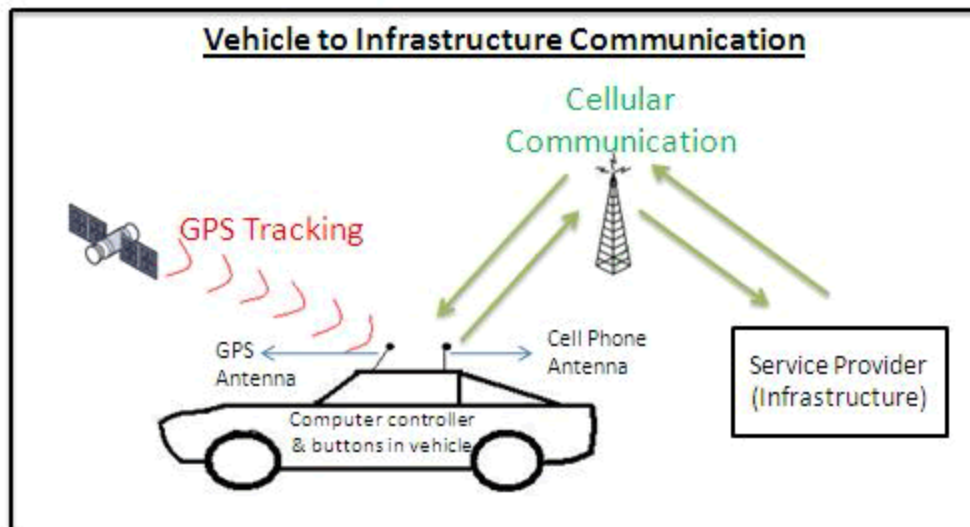


Figure 1.7: GPS tracking

1.6.2 Android Based Bus Communication System:

This is another way of communicating with the buses and the customers. In this system customer opens the app that is customized for the bus booking system on his smartphone ensuring that it has net connection. Then he can view the overall charts of the routes the bus provides which makes him easy to select the route. After that users book the seat and waits at

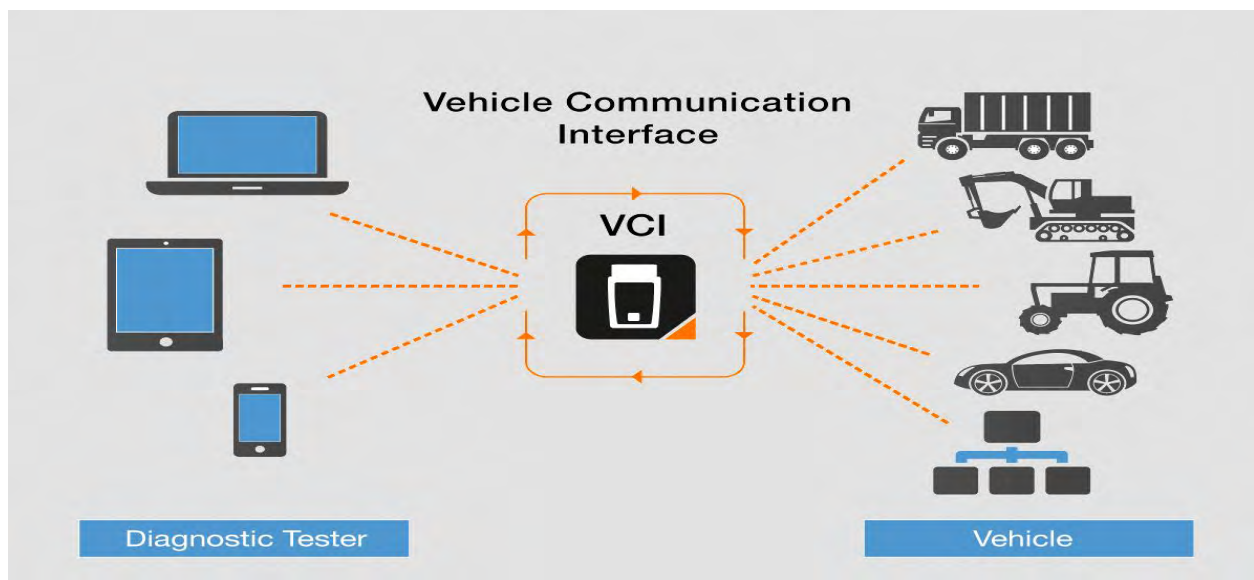


Figure 1.8: Vehicle Communication Interface

1.7 Different types of payment:

There are different types of payment systems that are being used around the world:

1.7.1 App Based Payment:

Payment can be collected through the app. The condition is that the users has to have an account for that and it is to noted that in their account sufficient amount of money must remain in order to pay the payment for while booking the seats. In many countries the apps that are used to communicate with buses are much more developed. Sometimes, through app people do everything with relax. They do not need to deal with the tickets as smart app came up to their hands.

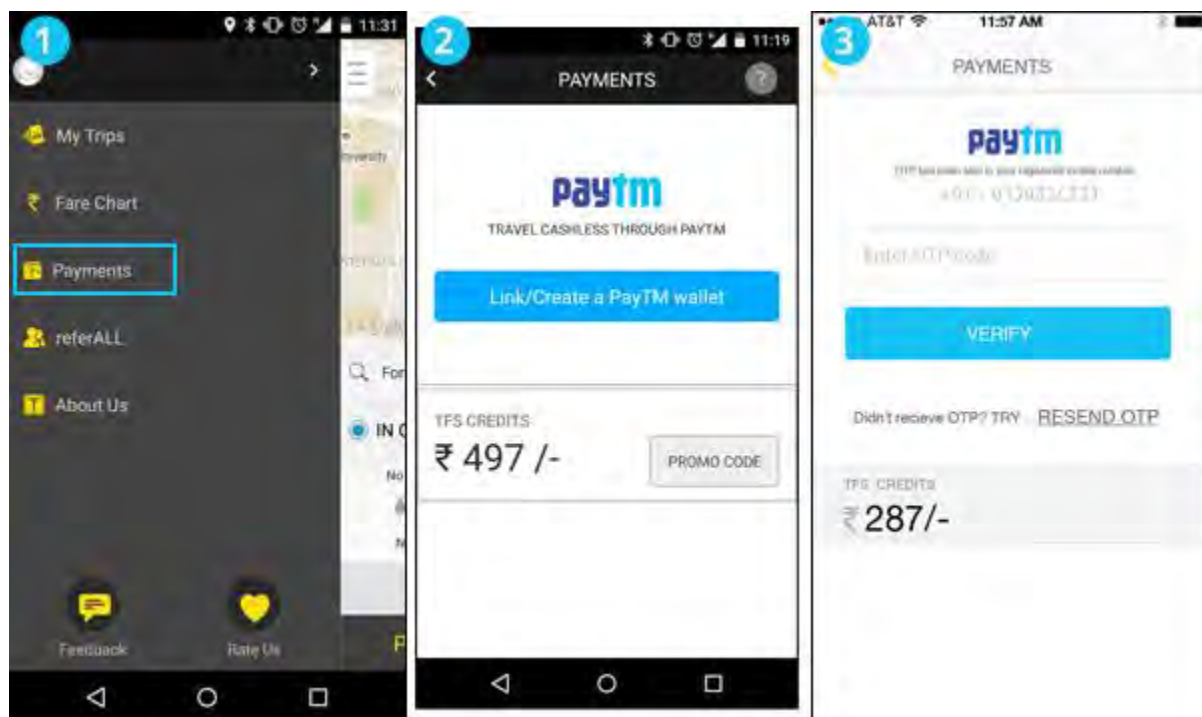


Figure 1.9: App based payment

1.7.2 RFID Based Payment:

Radio-frequency identification (**RFID**) uses electromagnetic fields to automatically identify and track tags attached to objects. The tags contain electronically stored information. Passive tags collect energy from a nearby reader's interrogating radio waves. Here, every user requires to have a RFID tag card that is used to punch on RFID device to pay the payment.



Figure 1.10: RFID Based Payment

1.7.3 Mobile Operator Based:

Payment can be collected in this way also. Users can pay the payment that is designed for the distance he will travel by using mobile money system. At present around the world there are many ways of mobile payment system called 'Mobile Banking System'. As soon as the Bus authority receives the payment from the user, authority then adds virtual money to the account of the user who paid by using mobile payment system. Then whenever the user confirms any bus to travel, fixed virtual money is deducted from his account.

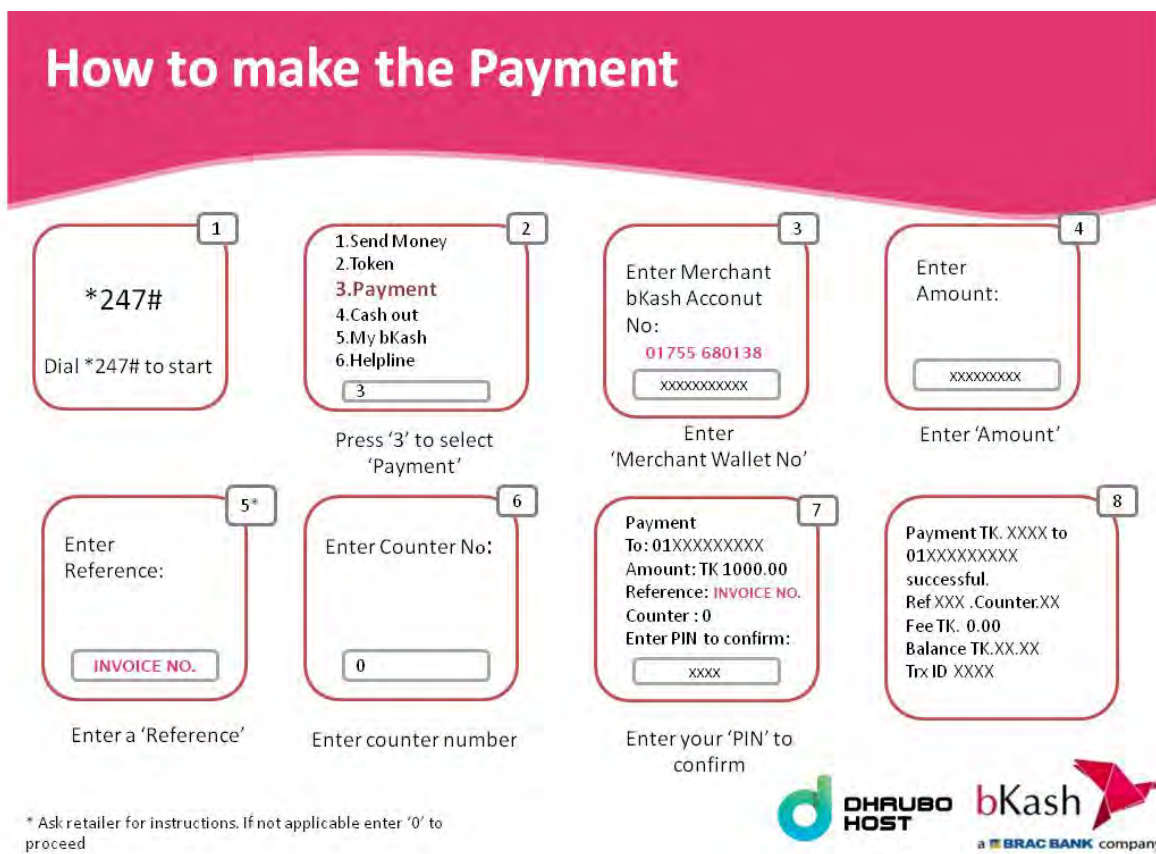


Figure 1.11: Payment System

1.8 Summary of Our Project:

Here, we created two apps for bus communication. One is for the customers for their access to the see the positions of the buses and to book seats. Another is for the bus to provide latest information of the position of the bus and the status of seats to the customers. From the customer app users can select his desired route with the bus that is available and can book his desired seat conditioning if that is available. Then users have to pay the payment in order to confirm their tickets. Unless they pay, the bus authority will not count him as the passenger of the bus. After finishing all of these very flexible steps, users have to wait at the bus stoppage for the selected bus to ride. Lastly to say, we tried to implement this project as much as flexible as we targeted all classes of people in our country by making all the features as simple and easy as we can.



Figure 1.12: Search BUS from Map

Chapter 2

Project Overview

2.1 Application Area of the Project

The women who are travelling by bus facing many problems in Bangladesh. There is no privacy for them. And some of the people arguing with them for the seat which is already reserved for them. Women's Safety in Public Transport is one of the key challenges across the globe. The authorities are taking various measures to make Public Transport safe and comfortable for women. But we think its not enough still the women of our country faces many obstacles here. It is very important to understand the travel needs of men and women are very different. In developing countries, lots of women forced to remain at home owing to lack of safe transportation option. Women are more likely to travel shorter distances and to stop more frequently than men during their journey. For this kind of issues we have designed a design for the women of our country and it will only serve the women. Women can cut their ticket via mobile app inside from home. It will be very easier to cut the ticket. They do not have to bargaining with the contractor.

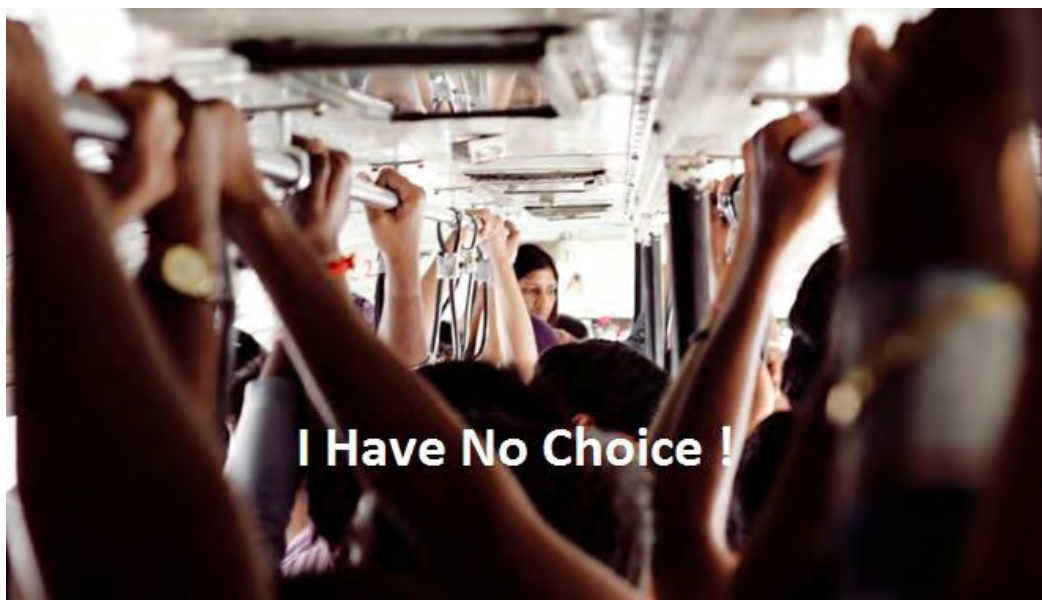


Figure: women have no Choices

We are constantly receiving complaints of women being abused, leered at or followed, Again they face antisocial elements ,abusive behavior in public transport so it's another reason for what we r trying to launch a women-only bus service to protect female passengers from groping and verbal abuse common on its packed public transportation system.



Figure 2.1: Inconvenience of Women to get up in bus

2.2 Bus Ticket System

Bus ticket booking during the offline era posed various difficulties to the customers as well as the bus operators. Offline ticket booking reduced the scope of customers to choose different options based on their travel criterion (Gayathry, 2013). It also increased the franchising cost for the bus operators. At the same time, the bus operators were also finding it difficult to monitor their bus seat filling information. Many small and medium bus service organizations do not have their own online bus ticket booking system. Online Bus ticketing system web portal is a total internet ticketing operations offering the benefit of total in-house ticket bookings, ticket sales. It also offers the power of decision making to customers to make a ticket booking through bus operators 'popularity, performance and ranking. This powerful Internet based ticket booking system that allows a full control of the ticketing inventory. According to Melisa (2007), stated the basic components of an Online Bus Ticketing System web portal that provides enhanced service to the bus operators and customers consist of the following:

- Capture of customer information such as name, address, phone number and e-mail address
- Price list
- Bus operators ranking
- Seating chart
- Search Bus
- Payment information
- Reports



Figure 2.2: No privacy for women

2.3 Working principle of our project

First of all the driver of the bus press start button from the application then go to the RFID input page from the application. Which will take the RFID input from the user.

Secondly, user trace the bus by smart phone application. Then book a ticket. Then he/she have to travel with the bus. Then he has to give the RFID Tag in the bus door to confirm the ticket. Then he/she can enter into the bus

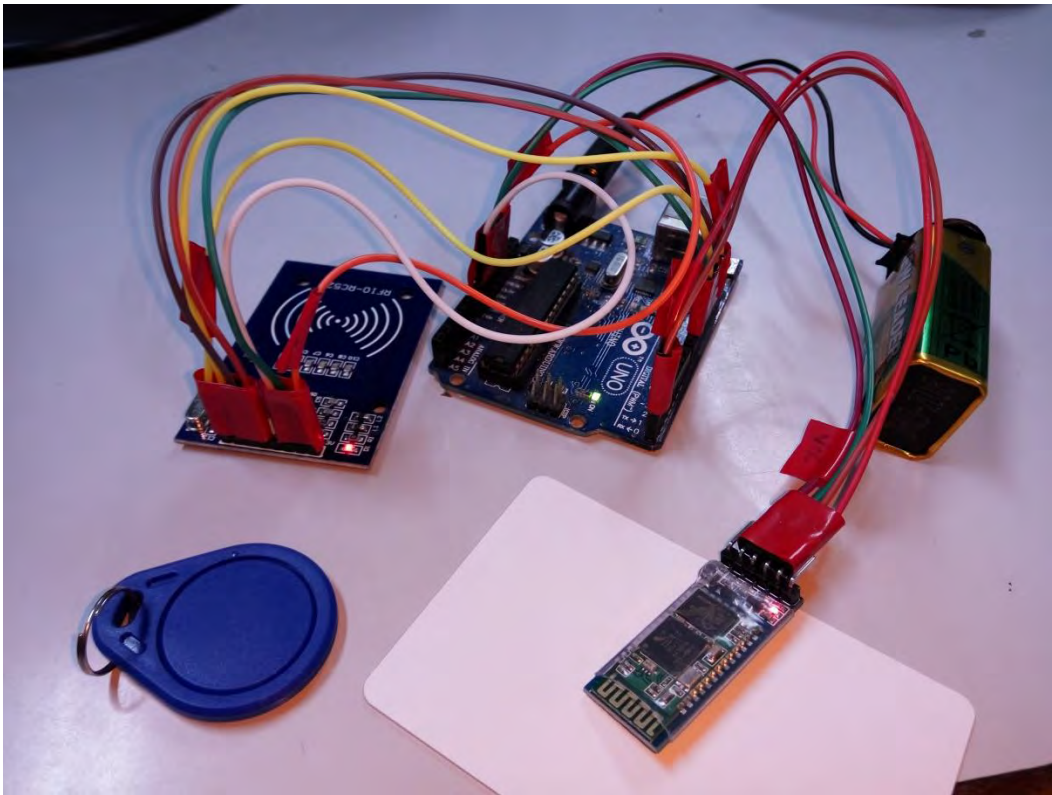


Figure 2.3: Components of our project

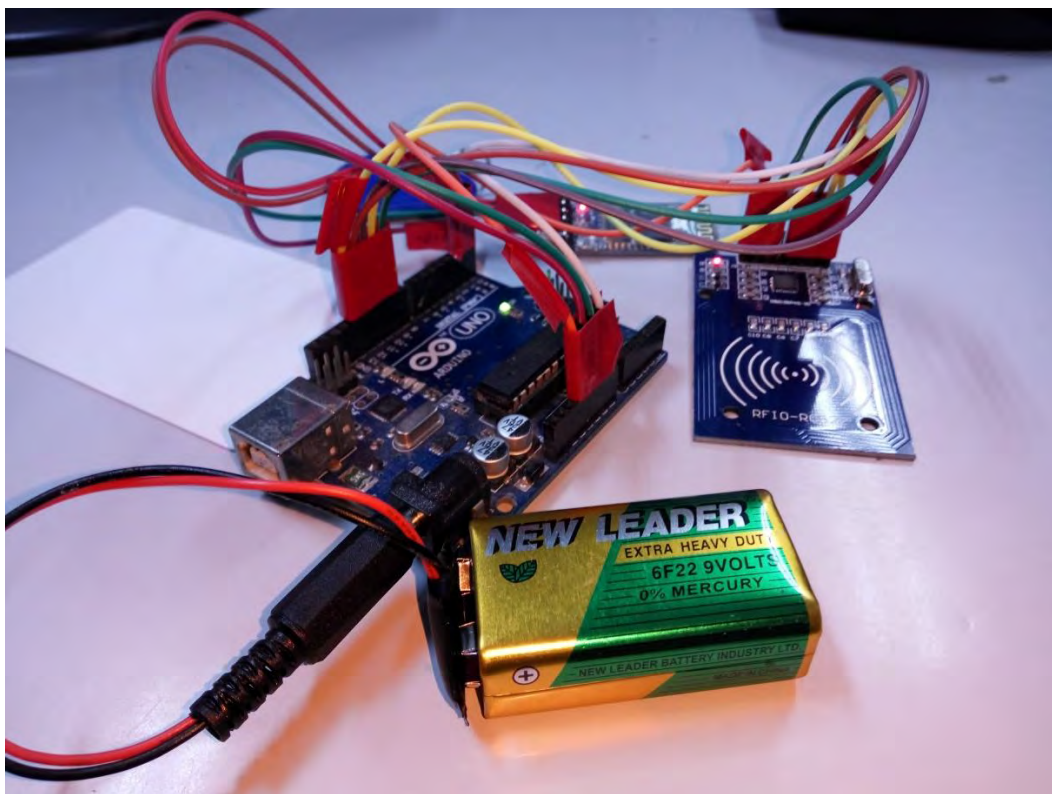


Figure: The Components we have used

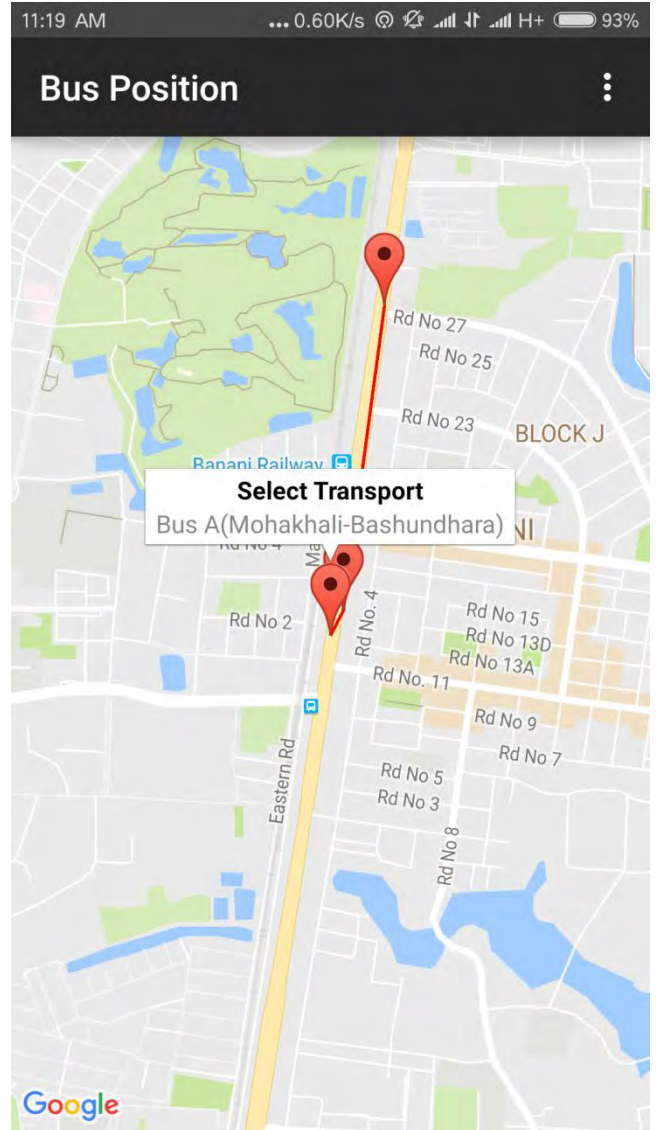
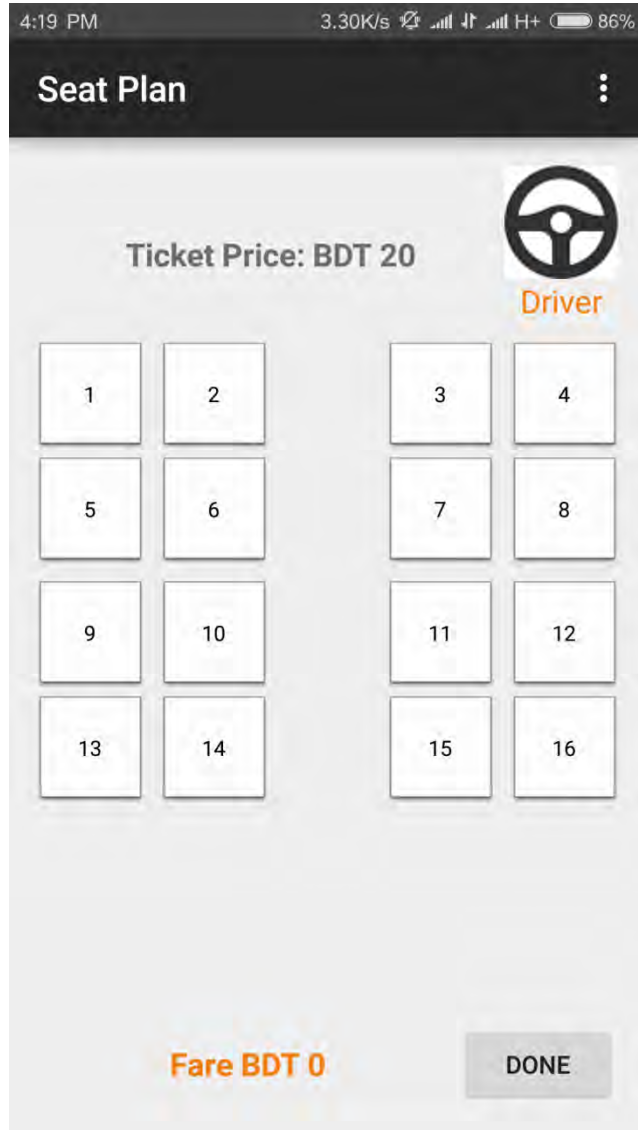


Figure 2.4: Seat plan & Bus Position

Chapter 3

Hardware Components

3.1 Introduction

It is always important to have proper knowledge about all the hardware and software components of the project. We have used lots of components to make our project perfect. In our project the main and most important part is Arduino UNO R3 for the user identification and interfacing the locator application. We have also used Bluetooth Module HC-05 for interfacing the RFID RC-522 with application. For Bus location tracing we have used GPS enabled android based smartphone whether user can trace the bus.

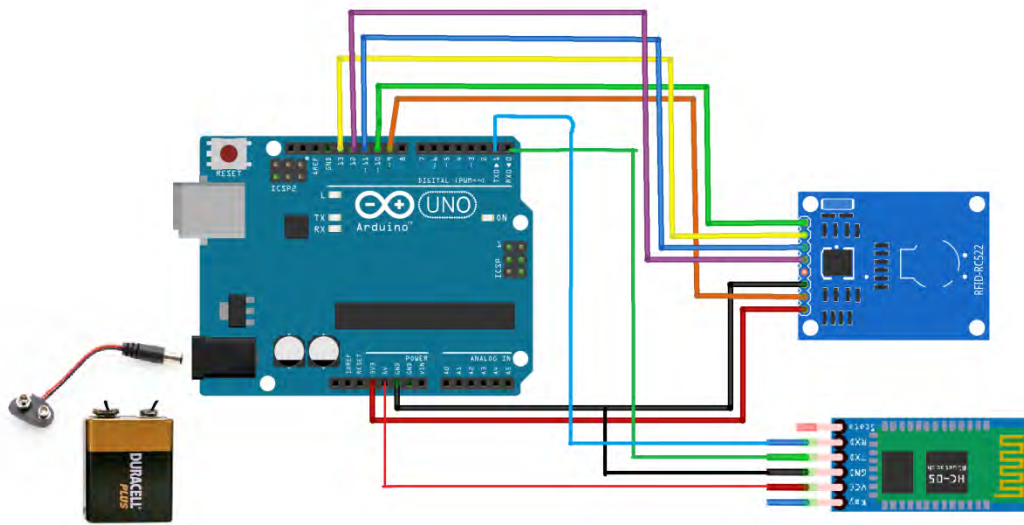


Figure 3.0: Schematic Diagram

3.2 Hardware components

We have used lots of hardware and software components in our project. Now let's discuss about those components in below

- Arduino Uno R3
- Bluetooth Module HC-05
- RC-522 13.56 MHz RFID Reader
- GPS enabled Smartphone
- Jumper Wires (Male-Male, Female-Male)
- 9V DC Battery

3.2.1 Arduino UNO R3

Arduino/Genuino Uno is a microcontroller board based on the ATmega328P .It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz quartz crystal, a USB connection, a power jack, an ICSP header and a reset button. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with a AC-to-DC adapter or battery to get started. You can tinker with your UNO without working too much about doing something wrong, worst case scenario you can replace the chip for a few dollars and start over again.

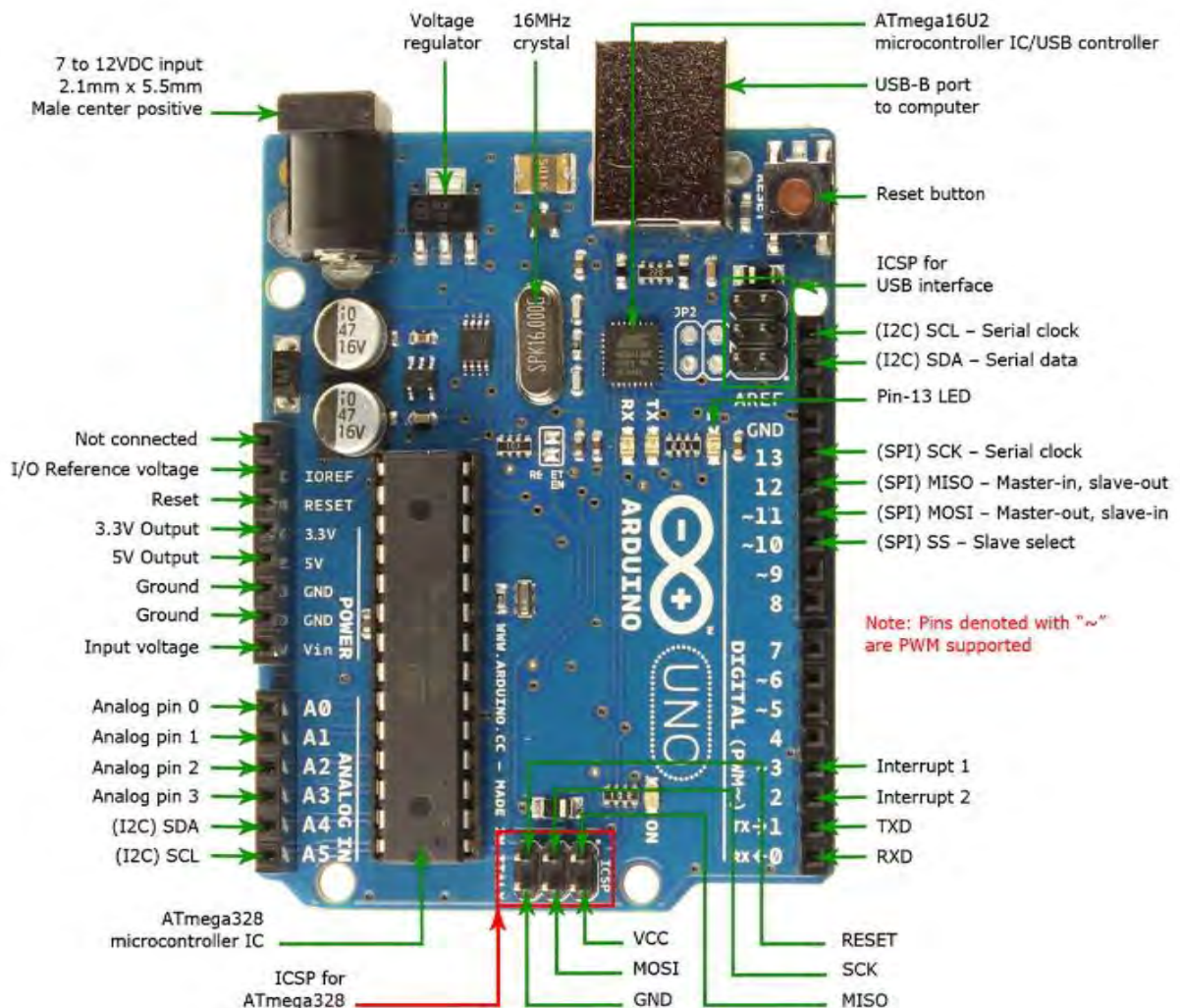


Figure 3.1: Arduino UNO R3

3.2.2 Bluetooth Module HC-05

HC-05 module is an easy to use Bluetooth SPP (Serial Port Protocol) module, designed for transparent wireless serial connection setup. Serial port Bluetooth module is fully qualified Bluetooth V2.0+EDR (Enhanced Data Rate) 3Mbps Modulation with complete 2.4GHz radio transceiver and baseband. It uses CSR Bluecore 04-External single chip Bluetooth system with CMOS technology and with AFH(Adaptive Frequency Hopping Feature). It has the footprint as small as 12.7mmx27mm. Hope it will simplify your overall design/development cycle.

Hardware features

- Typical -80dBm sensitivity
- Up to +4dBm RF transmit power
- Low Power 1.8V Operation ,1.8 to 3.6V I/O
- PIO control
- UART interface with programmable baud rate
- With integrated antenna
- With edge connector



Figure 3.2: Bluetooth Module HC-05

3.2.3 RC-522 13.56 MHz RFID Reader

This low cost MFRC522 based RFID Reader Module is easy to use and can be used in a wide range of applications. RC522 is a highly integrated transmission module for contactless communication at 13.56 MHz. This transmission module utilizes an outstanding modulation and demodulation concept completely integrated for different kinds of contactless communication methods and protocols at 13.56 MHz.

The MFRC522 is a highly integrated reader/writer IC for contactless communication at 13.56 MHz.

Features:

- MFRC522 chip based board
- Operating frequency: 13.56MHz
- Supply Voltage: 3.3V
- Current: 13-26mA
- Read Range: Approx. 3cm with supplied card and fob
- SPI Interface
- Max Data Transfer Rate: 10Mbit / s
- Dimensions: 60mm × 39mm



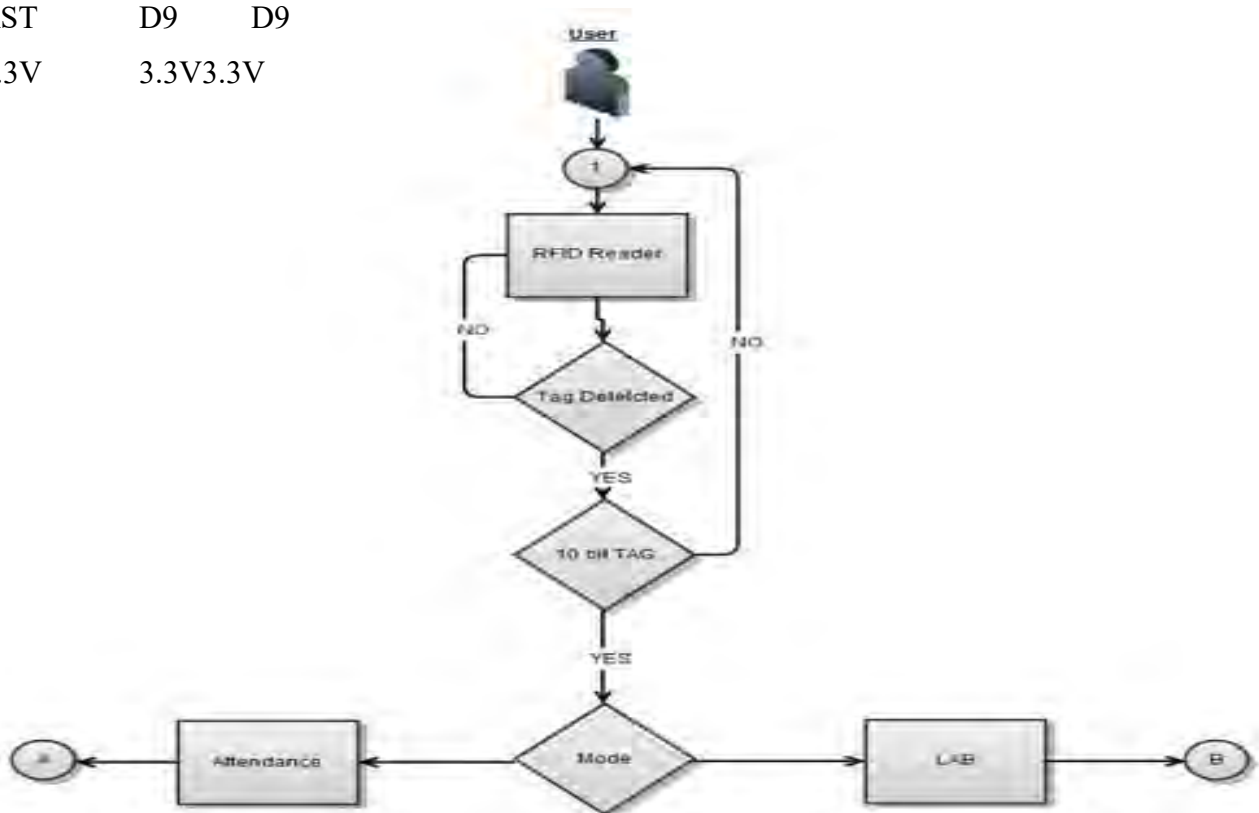
Figure 3.3: RC-522 13.56 MHz RFID Reader

Package contents

- 1 x RC522 RFID Module
- 1x RFID plain white Card
- 1x RFID FOB
- 1x 8pin right angle header pins
- 1x 8pin straight header pins

Arduino Wiring

RC522 MODULE	UNO	MEGA
SDA(SS)	D10	D53
SCK	D13	D52
MOSI	D11	D51
MISO	D12	D50
PQ	Not Connected	
GND	GNDGND	
RST	D9	D9
3.3V	3.3V3.3V	



3.2.4 GPS enabled Smartphone

In this project we use smart phone as GPS tracker and for booking ticket from android application.



© BECL 2013. ALL RIGHTS RESERVED.

Figure 3.5 : GPS Enabled Smart Android Phone

3.2.5 JUMPER WIRE

Here are some things that may make it easier for you. Here are some of the cables and pins that are available which is named as jumper wire. Here are a few places you can jump to, if you're in a hurry. Because of that we are using here jumper wire.

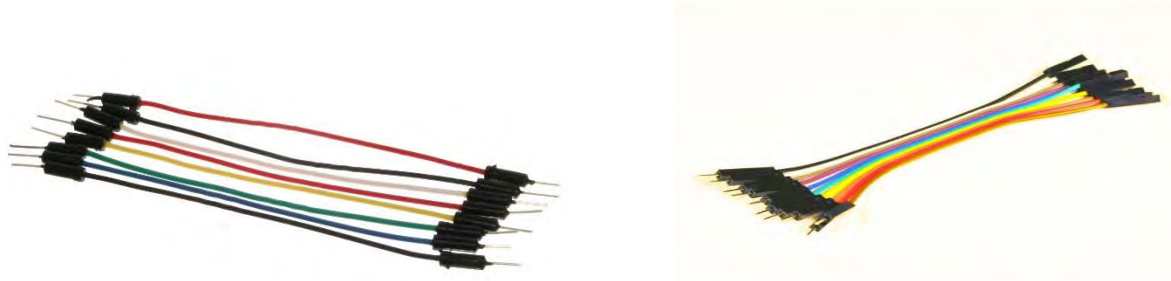


Figure 3.6: Jumper Wire

3.2.6 9v Battery

We are using here 9v Battery for power up the Arduino uno.



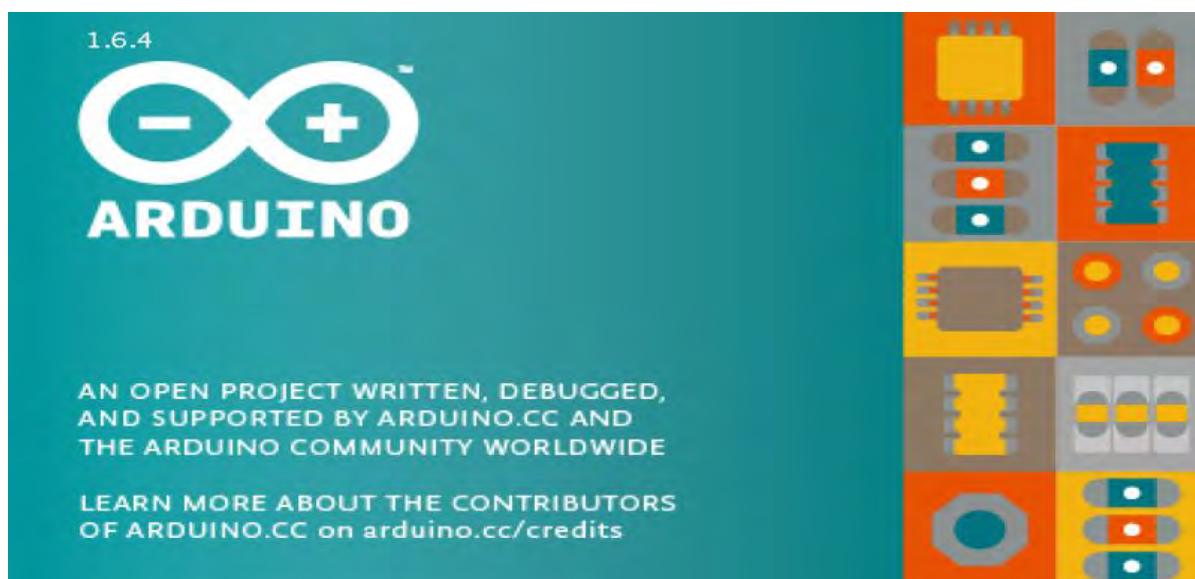
Figure 3.7: 9v Battery

3.3 Software Components

As we want to design Smart Public Transportation System, we need to have some knowledge in different types of programming languages by which we can easily build a connection between microcontroller and Android app and also can store and retrieve data in memory. To receive data from the RFID and send it to Android Application, Arduino microcontroller and HC-05 kit module had been programmed by the Arduino IDE software.

3.3.1 Arduino IDE

It is known that Arduino IDE is open source software. It is used to compile the program into the microcontroller. C- Programming language is used for coding in this software. There is two parts in this code mainly. Void setup () is known as preparation for the program and it runs only once, void loop () is known as execution for the program. In this software we have written some function to get the authentication of RFID tags And Bluetooth Module to send data to android Application .



3.3.2 Android Studio

Android Studio is Android's official IDE. It is purpose built for Android to accelerate your development and help you build the highest-quality apps for every Android device.

It offers tools custom-tailored for Android developers, including rich code editing, debugging, testing, and profiling tools.



Android Studio's Instant Run feature pushes code and resource changes to your running app. It intelligently understands the changes and often delivers them without restarting your app or rebuilding your APK, so you can see the effects immediately.

The code editor helps you write better code, work faster, and be more productive by offering advanced code completion, refactoring, and code analysis. As you type, Android Studio provides suggestions in a dropdown list. Simply press Tab to insert the code.

The Android Emulator installs and starts your apps faster than a real device and allows you to prototype and test your app on various Android device configurations: phones, tablets, Android Wear, and Android TV devices. You can also simulate a variety of hardware features such as GPS location, network latency, motion sensors, and multi-touch input.

We use this software to write our code for GPS tracker , RFID Input, Online ticket Booking, GPS location.

Chapter 4

Proto type Mechanism and Algorithm

4.1 Mechanism and Algorithm

4.1.1 System Design:

Smart Public Transportation System is made up with Android Application, Arduino UNO R3, RFID Reader And Bluetooth Module. The core part of tracking system is Smart phone. Most of the coding of our system is design in the Android Studio. We use Android studio programming language to run the whole system for our Smart Public Transportation System.

We made up a application named Bus Locator. There we build up a code for trace the bus. For this we have to press the button start button to give the current location of the bus to the server. In this application we also connect with the Arduino Uno with the interface of the Bluetooth module to take the read of RFID tags. It will give the information of the user. Here we have used RC-522 13.56 MHz RFID Reader because RC522 is a highly integrated transmission module for contactless communication at 13.56 MHz this transmission module utilizes an outstanding modulation and demodulation concept completely integrated for different kinds of contactless communication methods and protocols at 13.56 MHz

We also made up another application named Bus Ticket. In this application we can trace the location of the current location of the bus by selecting the route of journey. After selecting the bus from the map there will be a pop up window to select the seat he/she wanted. After selecting the seat there is a book option to book the seat. After booking the seat, there pop up window that will show the current balance of the user. Than to confirm the ticket user must have to ride the bus which he/ she have bought the ticket. To confirm it user will give the RFID tag in the door where the RFID reader is located. After confirming with the RFID tag the user can enter in the bus.

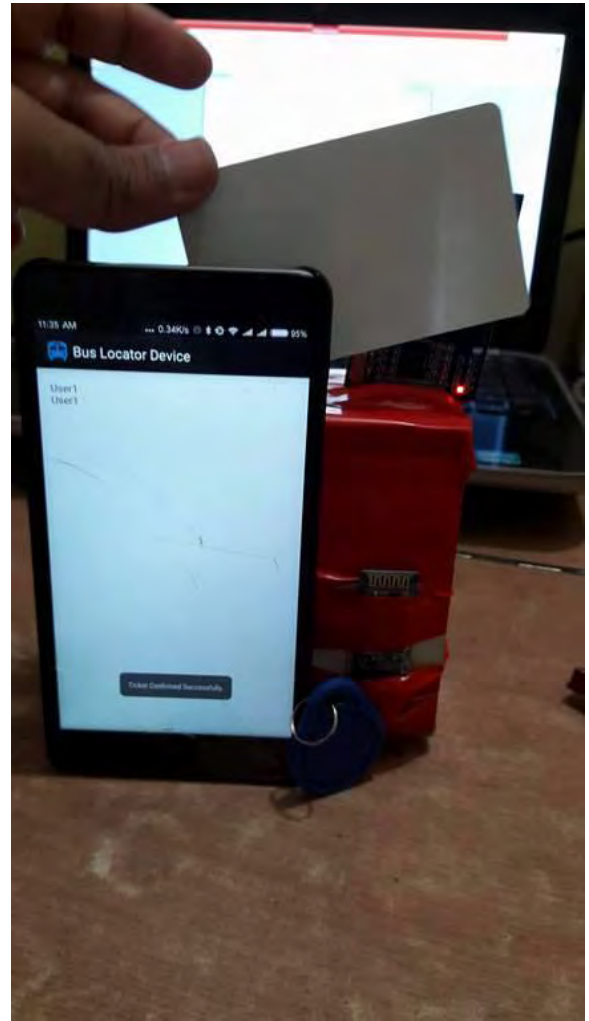


Figure 4.1: Bus locator Device

4.1.2 BUS LOCATOR APPLICATION

First of all driver have to open the application from the device. Then he have to prees the button to strt the location service.

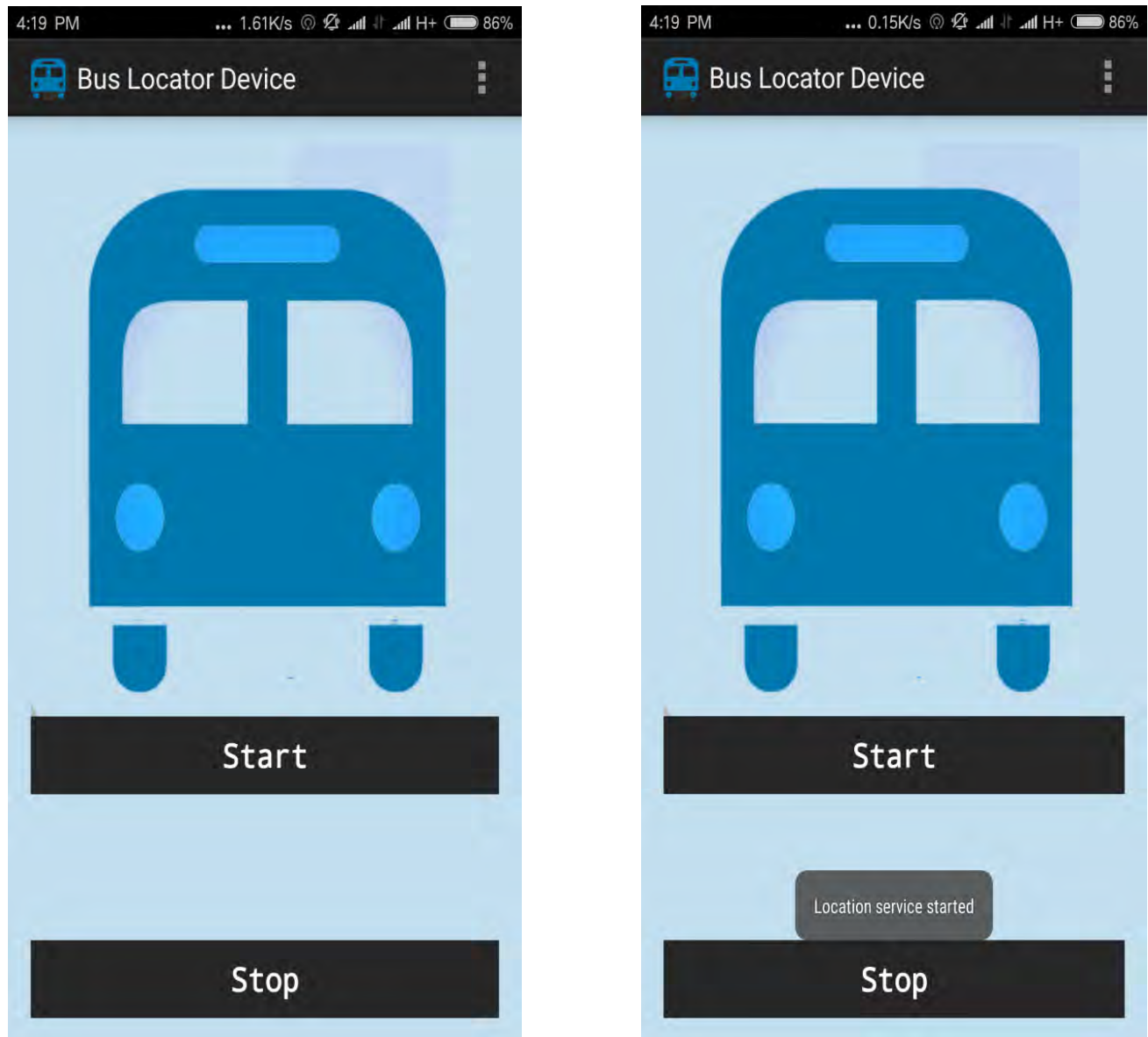


Figure 4.2: Bus Locator Application and Location Service Started

After starting the location he has to go to the RFID window to take the input of RFID tags of the user. Then the application will automatically send the information of the user to the server.

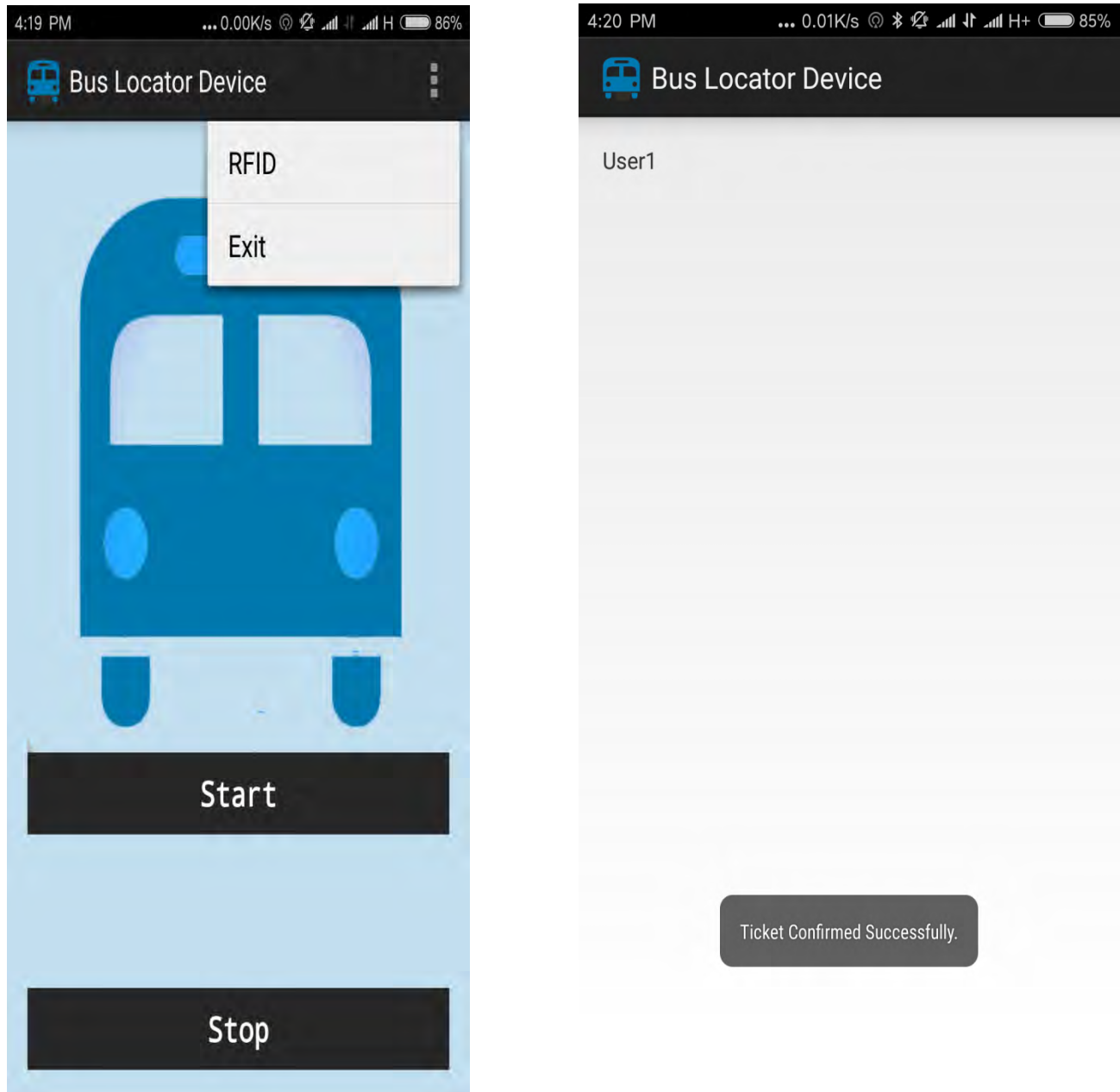


Figure 4.3: For RFID POP up Window & INPUT RFID TAG & Ticket Confirmation

4.1.3 FOR USER END APPLICATION

First of all User have to install this app and open an account here. Here she will find 2 boxes she has to insert her user name through which she can access this account then she has to insert her password. Then she can login here

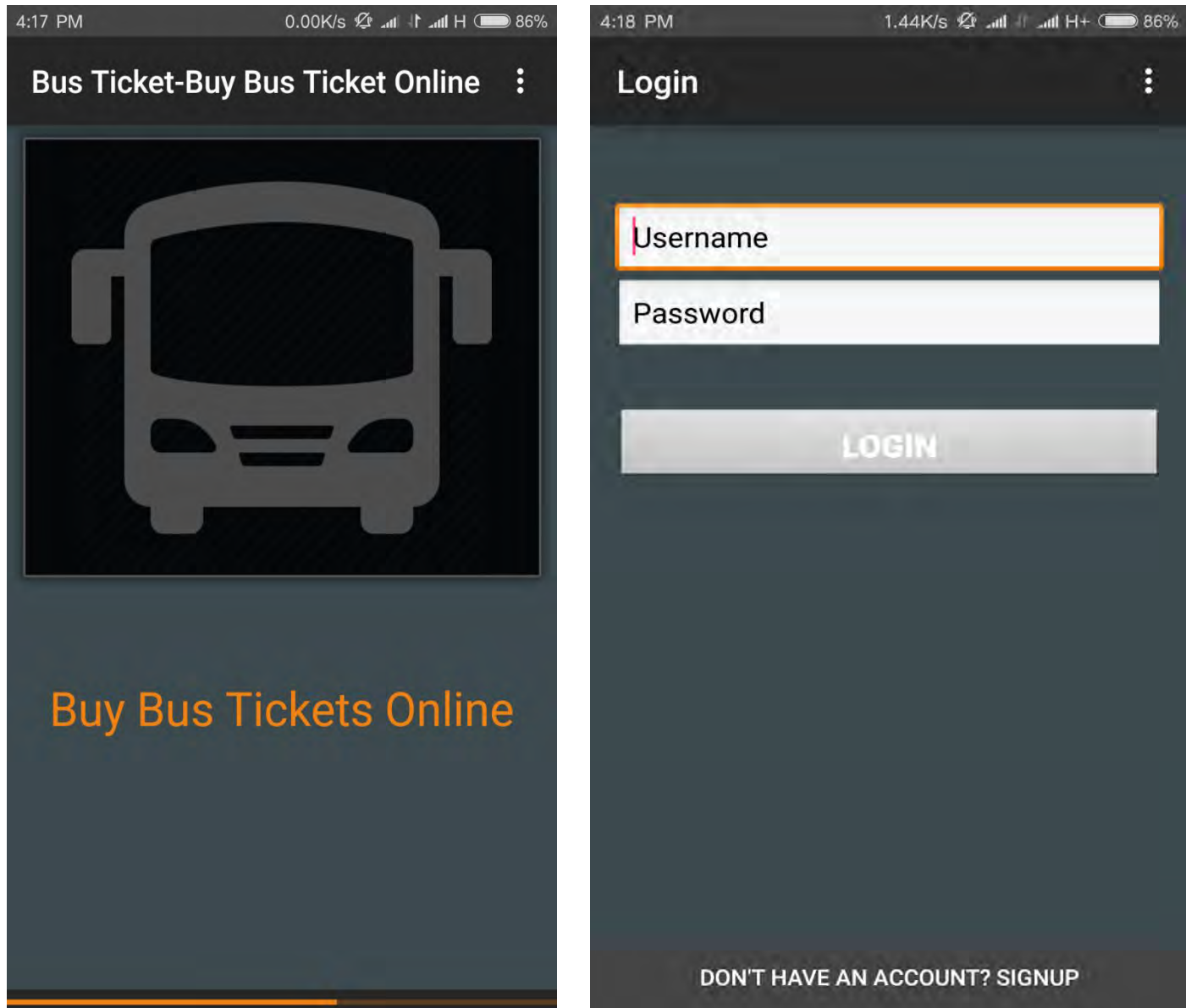


Figure 4.4: BUY Bus Ticket & User Login

She will tap a button to get a ride .then she has to set her current location from where she will set off and desired destination. She will see vehicle details and can track their arrival on the map

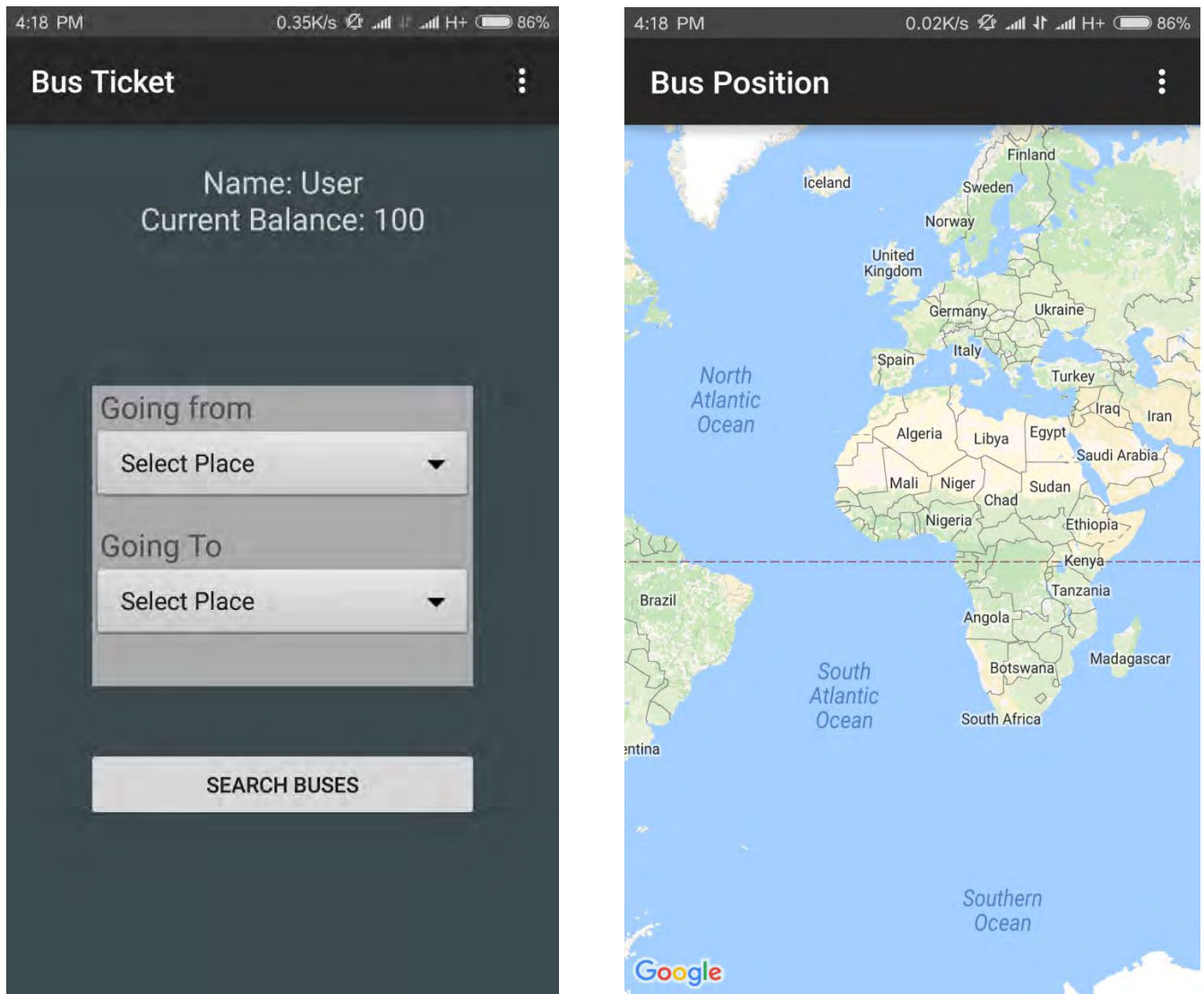


Figure 4.5: USER Account Balance & Search for Bus

When the location will be detected she will book the seat after booking the seat automatically 20 taka will be deducted from her balance

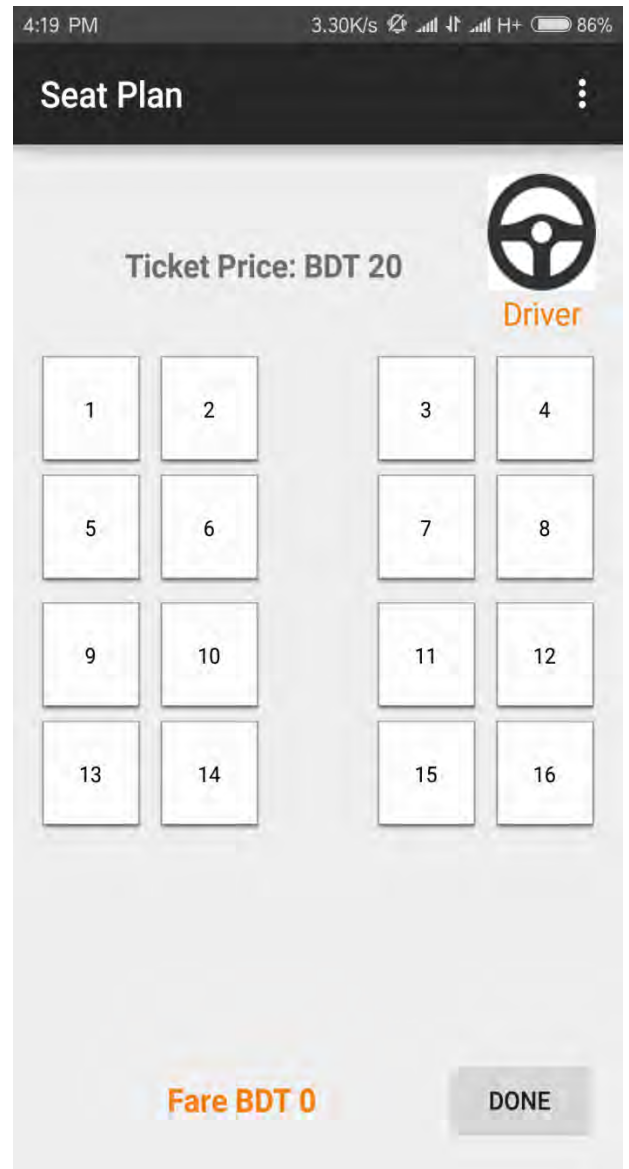
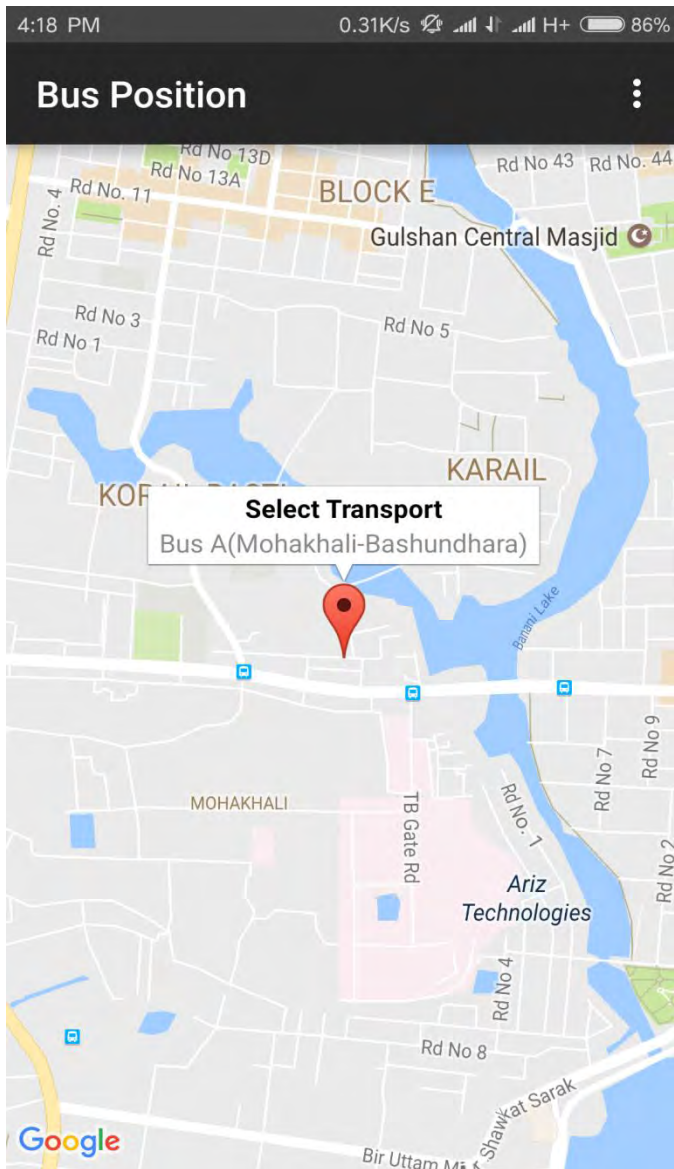


Figure 4.6: Bus Position & Seat Plan

4.1.4 Algorithm



Figure 4.7: Bus Seat Booking Application Flow Chart

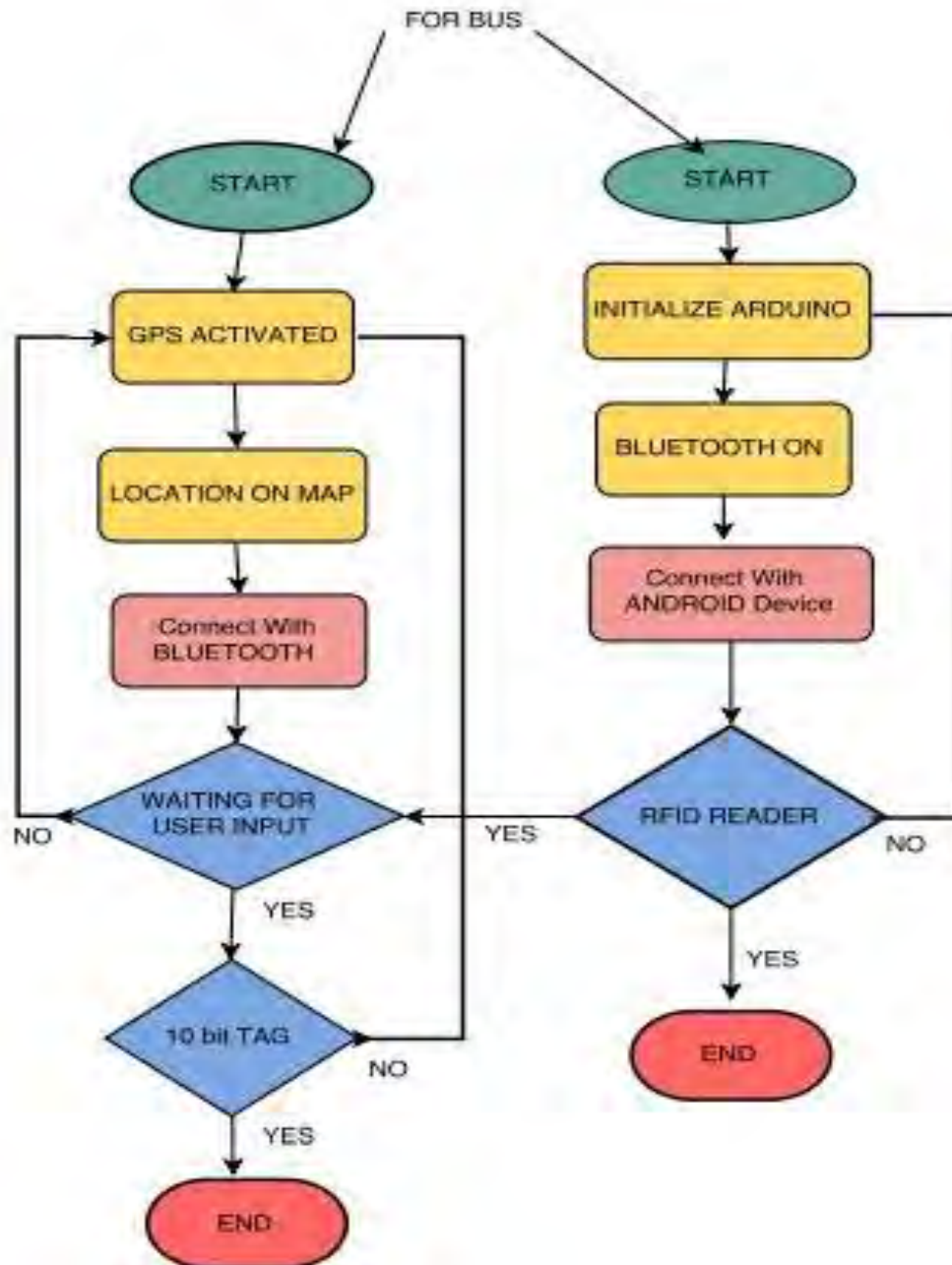


Figure 4.8: For Bus End Application

4.2 Working procedure of prototype:

Below we have describe the our whole system step by step

4.2.1 For BUS Locator Application

First of all the driver of the bus press start button from the application then go to the RFID input page from the application. Which will take the RFID input from the user.

4.2.2 For BUS Ticket Application

First of all, users trace the bus by smart phone application. Then book a ticket. Then he/she have to travel with the bus. Then he has to give the RFID Tag in the bus door to confirm the ticket. Then he/she can enter into the bus

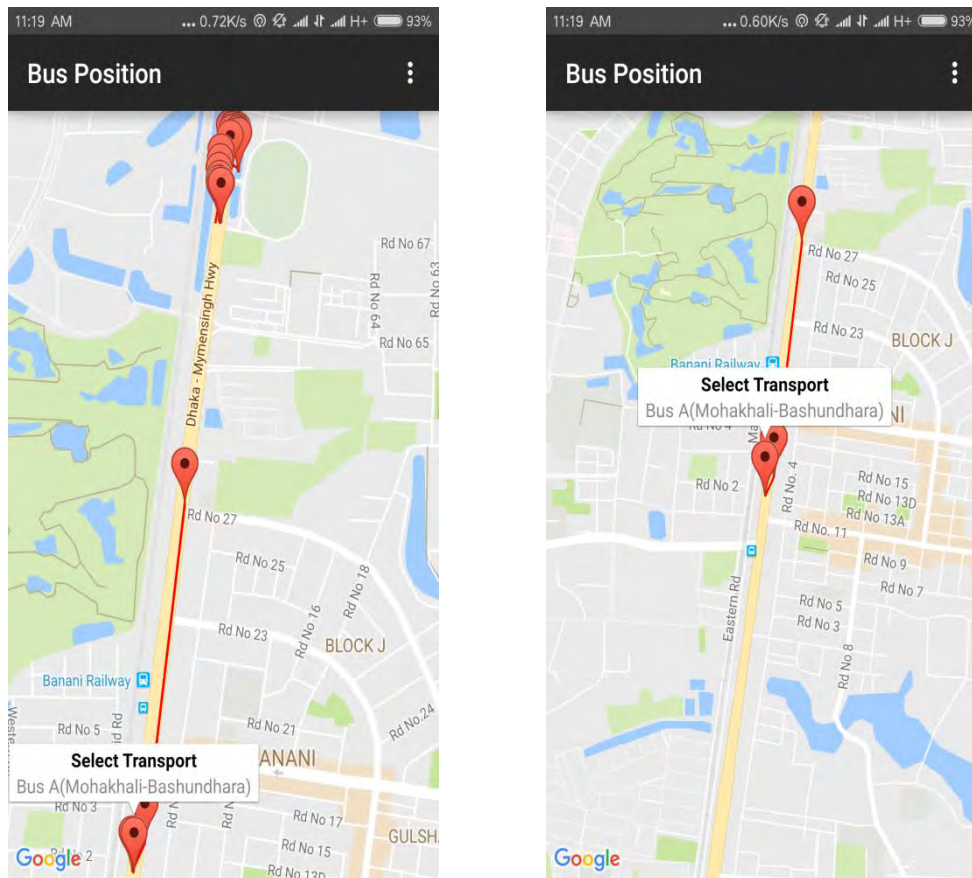


Figure 4.9: BUS Locating

4.3 Display results:

4.3.1 For BUS Application

For getting the facilities from this app the user must have owned RFID tag. When the user get into the bus she has to give her RFID tag to RFID reader so that it can be ensured that the ticket has been confirmed

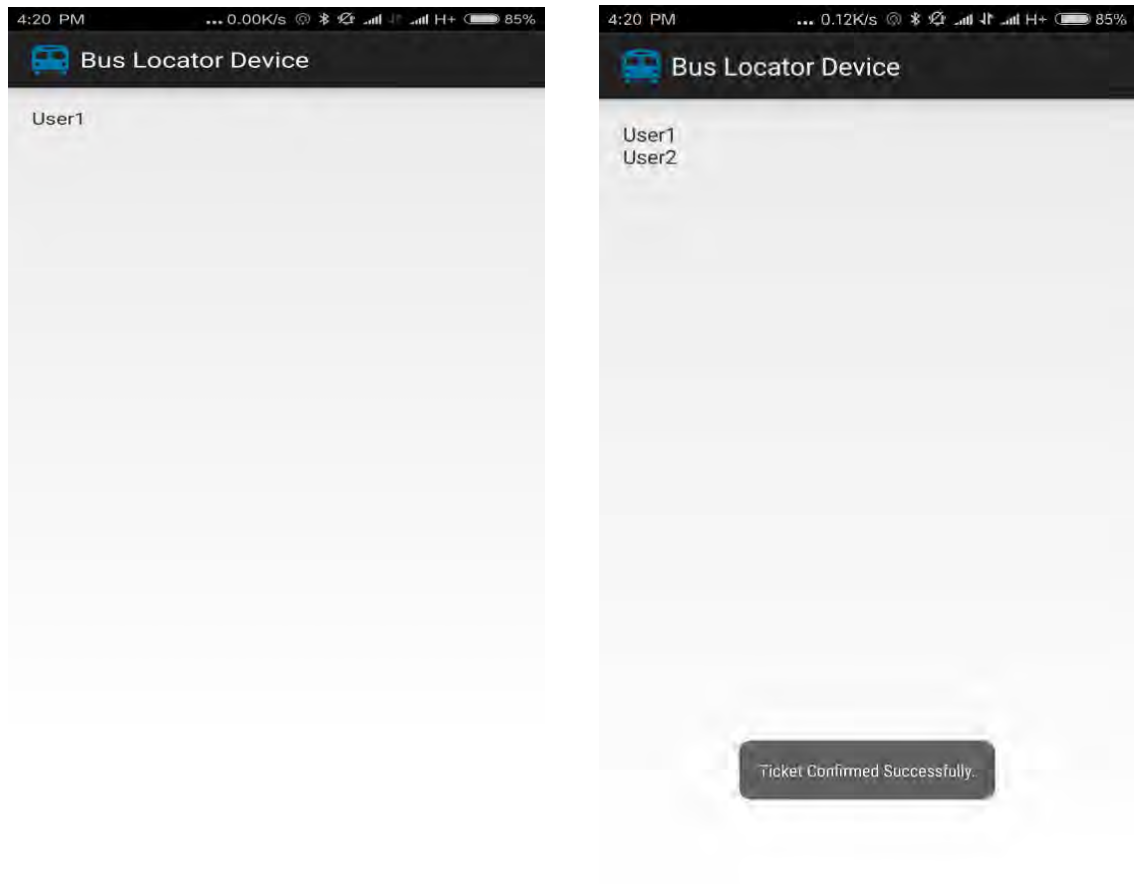


Figure 4.10: RFID detected

4.3.2 For User End Application

If the ticket has been booked then the app will confirm that. And she will be notified it through the app as you can see the notification “you have successfully purchased the ticket” .And there will be another popup window that will show that 20 taka is deducted from her balance and it will also show her current balance. To confirm da ticket she has to tap the button “DONE”.

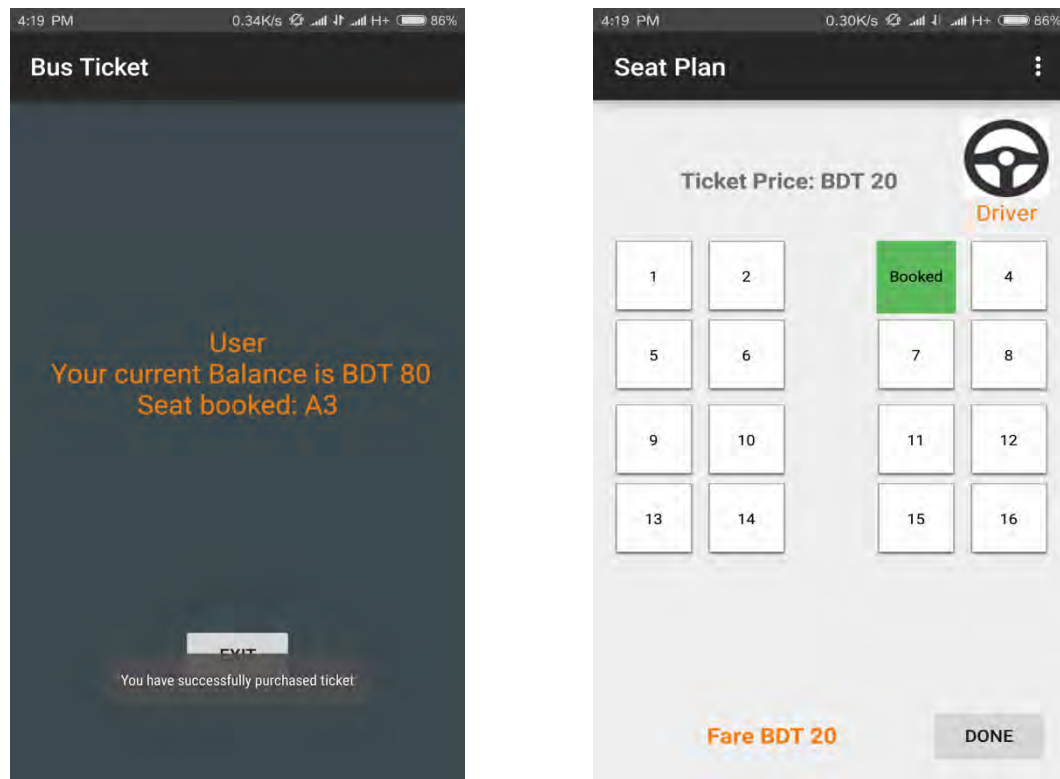


Figure 4.11: SEAT BOOKED

Chapter 5

Economic Overview

5.1. Economic losses in present public bus transportation system:

The traditional system has some demerits that can lead to some economic losses for the concerned authorities and public.

- **Wastage of paper:** The traditional system has paper printed tickets for the seat, but after that the tickets just been torn away and this literally creates some unwanted wastages for environment and also it is also wasting money for printing the tickets again.
- **Flaws in revenue collection:** In traditional system the revenue are being collected manually by the bus conductor. So there are high chances of flaws in collecting revenue for human error. So the bus admins can face an economic loss.
- **Tax Invasion:** Traditional System does not provide any proper revenue collection data to the government authorities, therefore; there are high chances of tax invasion and government can be deprived from a great tax collections.

5.2. Proposed transportation system economic benefits:

Smart Public Transportation System is also has some economic benefits beside other benefits it will provide to the publics. Some of the significant benefits are described below

- **Environmental Benefits:** It is saving a huge amount of papers that are used for printing tickets, the system we have designed that provides an electronic ticket to the user thus it will save a huge amount of money that were needed for the ticket printing as well the papers.
 - **Benefits of bus administration:** The system also has internal database the will provide the total revenue collection to the administrators that is why the administrator do not have to assign a person for revenue collection thus it also benefitting the administrators economically.
 - **People will use public transportation:** This designed system is way more convenient than conventional transportation system; people will be encouraged to take public transportation over the private transportation. As a result, the traffic jam will be less because the public buses can accommodate more people and the people can save their money that they have been spending for their private transportation.
 - **Government Tax collection:** Moreover, it will also benefit the government; government can have an access to the database of the administrators for invigilate the revenue collection and can impose proper tax in it.
 - **Foreign Investments:** Last but not the least, the proposed transportation system can also have an impact on our country's ICT Sector, more investors will come to our country for investing in information and technology sector, thus our government will have get benefitted by these such kinds of ideas.
-

Chapter 6

Plan and Budget

6.1. Plan for proto type

6.1.1. Timeline of proto type

Calendar 2016-2017											
MAY	JUNE	JULY	AUG	SEPT	OCT	NOV	DEC	JAN	FEB	MAR	APR
1. Read and Search information about RFID											
2. Identifying project objectives											
3. Searching for materials											
4. Gathering ideas for designing											
				1. Full design of project							
				2. Working on Arduino							
				3. Writing Code (Android)							
				4. Components Gathering							
								1. Completing full proto type			
								2. Simulating all codes and merging with hardware			
								3. Writing the book			
								4. Presentation			

Table 6.1: Timeline of proto type

6.1.2. Budget for proto type

COMPONENTS (FOR ONE BUS ONLY)	PRICE (IN TAKA)
GPS AND BLUETOOTH ENABLED SMART PHONE	6000/-
RIFD READER	750/-
BLUETOOTH MODULE	600/-
AURDUINO UNO	600/-
TOTAL	7950/-

Table 6.2: Budget for proto type

Chapter 7

Conclusion and Future Ideas

7.1 Conclusion:

The women, who are travelling by bus facing many problems in Bangladesh. There is no privacy for them. And some of the people arguing with them for the seat which is already reserved for them. Women's Safety in Public Transport is one of the key challenges across the globe. The authorities are taking various measures to make Public Transport safe and comfortable for women. But we think it's not enough still the women of our country face many obstacles here. It is very important to understand the travel needs of men and women are very different. In developing countries, lots of women forced to remain at home owing to lack of safe transportation option. Women are more likely to travel shorter distances and to stop more frequently than men during their journey. For this kind of issues we have designed a design for the women of our country and it will only serve the women. Women can cut their ticket via mobile app inside from home. It will be very easier to cut the ticket. They do not have to bargaining with the contractor.

We are constantly receiving complaints of women being abused, leered at or followed, Again they face antisocial elements ,abusive behavior in public transport so it's another reason for what we r trying to launch a women-only bus service to protect female passengers from groping and verbal abuse common on its packed public transportation system.

We did not stop finishing our project only but also we tried to bring out the feedback of this project from the mass of the people. We experimented this project upon a selected people and what we got in respond really satisfied us. Finally, we could see that this project of us has been very beneficiary to the normal class of people as well as the working people. Now people can not also save their time but also can utilize the time that was supposed to be wasted while waiting for the bus. The working class of women would not become sufferer anymore to get a seat as we fixed some women physically disabled quotes among the seats of a bus. We tried to ensure the safety of the buses while the driver is driving by using GPS so that the driver can be monitored by the authority all the time. We made an ending line for the collectors of payment and the passengers who unwell to pay full money by using online payment system and RFID system. By

allocating the chart of seats on online while someone is going to book is another beneficiary as we removed the possibility of clashes of the passengers as they can now know their allocated seat before he enters the bus. In addition we think that this project will play an important role to reduce traffic jam as we designed the system for the bus to stop at only particular stoppages. Since the driver cannot park it here and there as he wishes to take on any passengers, thus the possibility of creating a traffic jam becomes less. Moreover, to cope up with the digital world we tried to digitalize the transportation system of our country by designing this project so that people find an smart solution of transportation problem. To conclude we would like to say that we tried our best to make this project fruitful and beneficiary for the society and for the betterment of the transportation system of our country.



Figure 7.1: Public transportation for women

7.2 Future Ideas:

7.2.1 Implementing speed meter:

In our daily lives a common scenario is the high number of abnormal deaths by road accidents. For the last 8 years 20,971 people died on road accidents(source BRTA). So Implementing speed meter in every buses of our system is highly on our consideration in future. We will work to set speed meters in the buses to measure the speed and sending it to the database of our control system. This will caution the drivers all the time and he will drive carefully as monitoring him is the thing that will be in remind of him.



Figure 7.2: Speed Meter

7.2.2 Face detectors:

Hacking is not a common scenario at present but we will not leave it away from happening. The accounts of our passengers could be hacked at any time. So we will build a system that is one user can upload a profile photo once in every 1 year. Therefore, if even the account is hacked but the photo will not be changed immediately and whenever the hacker will be on the way to get in the bus he will be caught as we will work on image processing system while getting on the buses. It is to note that we will also work on the system so that the accounts do not get affected by other users.

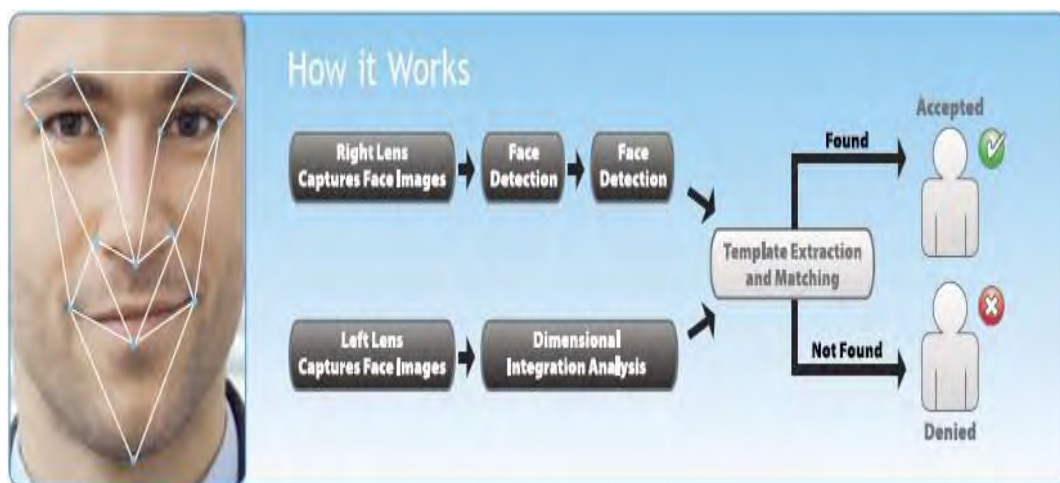


Figure 7.3: Face detection System

7.2.3 Using Counters:

Though there are many seating bus services in our country but none of them follow the criteria properly. As this project is named with smart so we will keep the buses smart by ensuring seating service accurately. Therefore, we will establish motion sensors on the door of the bus or we will use weight sensors on every seats so that it counts the passengers. Whenever the bus gets the passengers filling up all the seats the door will get locked automatically and no other can get the access to the bus. It is to state that due to the large population our country we will set this system only in a few buses but the other buses will not be used for this system as a large number of people will remain out of the service of our system if we use counter system in every buses.



Figure7.4: weight sensor

Reference

1. Khadijah Kamarulazizi and DR. Wadad Ismail, “Electronic toll collection system using passive RFID technology”, School of Electrical and Electronic, University Science Malaysia, Journal of Theoretical and Applied Information Technology, 2005-2010
2. Ganesh K. Andurkar and Vidya R. Ramteke, “Smart highway Electronic Toll Collection system”, International Journal of Innovative Research in Computer and Communication Engineering, Vol. 3, Issue 5, May 2015
3. Md. Saidul Hoque, Md. Jawed Alam, Md. Akbar Hossain Patwary, A.S.M. Elias Shah, A.H.M. Javed Hossain Talukdar, Mobashera Sadia Haque and Md. Shahriar Rahman Khan, Roads and Highways Department, Dhaka-1000, Bangladesh, Masaaki Tatsumi and Prosenjit Kumar Ghosh, Oriental Consultants Global Co. Ltd., Tokyo-151-0071, Japan, “Three 2nd bridges construction and existing bridges rehabilitation under KMG project, IABSE-JSCE Joint conference on Advances in Bridge Engineering-III, August 21-22, 2015
4. David Levinson and Elva Chang, “A Model for Optimizing Electronic Toll Collection Systems”, Department of Civil Engineering, University of Minnesota, Transportation Research Part A 37 (2003) 293-314, Received 6 September 1999; received in revised form 12 February 2002; accepted 19 February 2002
5. Priyanka Sharma and Vivek Sharma, “Electronic toll collection technologies: A state of art review”, an International Journal of Advanced Research in Computer Science and Software Engineering, Volume 4, Issue 7, July 2014, pp. 621-625
6. Abdalla Mohammed Mohammed, Maryse Mamdouh Naguib Fahmy Nashed, Michael Essam Amin Nasr, Mohammed Gamal El Prince and Nour El Din Mohammed Nour El Din, “Information and Communication Technologies Engineering Smart Road System”, Uninettuno University – Helwan University, 3 August 2011
7. Mohmoud Saffarzadeh and Abdolreza Rezaee-Arjroody, “Cost-Benefit Analysis of Electronic Toll Collection (ETC) System in Iranian Freeways”, Case Study: Tehran-QOM Freeway, PIARC International Seminar on Intelligent Transport System (ITS) In Road Network Operations, August 16, 2016
8. Nigar Sultana, “Efficiency Analysis of Public Transit systems in Bangladesh: A Case Study of Dhaka City” A thesis presented to the University of Waterloo in fulfilment of the thesis requirement for the degree of Master of Arts in Planning, 2013
9. Minhaj Mahmud and Atonu Rabbani, “Travel Mode Choice Preferences of Urban Commuters in Dhaka: A Pilot Study”, March 25, 2012

Other Reference Links

<http://www.iiste.org/Journals/index.php/CEIS/article/download/18128/18516>

<http://www.dwatch-bd.org/rassul/reports/transportation.doc>

<https://www.fundprovo.com/project/women-only-bus-service-in-paksitan>

https://www.thestar.com/news/2008/01/23/womenonly_bus_service_makes_first_stop_in_mexico.html

Appendix

Appendix A

Arduino Code:

```
RFID#include <SPI.h>
#include <MFRC522.h>

#define SS_PIN 10
#define RST_PIN 9
MFRC522 rfid(SS_PIN, RST_PIN);
MFRC522::MIFARE_Key key;
unsigned long previousMillis = 0;
byte nuidPICC[4];

void setup() {
  pinMode(13,OUTPUT);
  Serial.begin(9600);
  SPI.begin();
  rfid.PCD_Init();
}

void loop() {
  if ( ! rfid.PICC_IsNewCardPresent())
    return;

  if ( ! rfid.PICC_ReadCardSerial())
    return;

  MFRC522::PICC_Type piccType = rfid.PICC_GetType(rfid.uid.sak);
  if (piccType != MFRC522::PICC_TYPE_MIFARE_MINI &&
      piccType != MFRC522::PICC_TYPE_MIFARE_1K &&
```

```
piccType != MFRC522::PICC_TYPE_MIFARE_4K) {  
    return;  
}
```

```
if (rfid.uid.uidByte[0] != nuidPICC[0] ||  
    rfid.uid.uidByte[1] != nuidPICC[1] ||  
    rfid.uid.uidByte[2] != nuidPICC[2] ||  
    rfid.uid.uidByte[3] != nuidPICC[3] )  
{  
    for (byte i = 0; i < 4; i++) {  
        nuidPICC[i] = rfid.uid.uidByte[i];  
    }  
}  
else  
{  
    for (byte i = 0; i < 4; i++) {  
        nuidPICC[i] = rfid.uid.uidByte[i];  
    }  
    printID(rfid.uid.uidByte, rfid.uid.size);  
    Serial.println();  
    rfid.PICC_HaltA();  
    rfid.PCD_StopCrypto1();  
}  
}
```

```
void printID(byte *buffer, byte bufferSize) {  
    for (byte i = 0; i < bufferSize; i++) {  
        Serial.print(buffer[i] < 0x10 ? " 0" : " ");  
        Serial.print(buffer[i]);  
    }  
}
```

Appendix B

BUS LOCATOR

Manifest

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.example.lenovo.gpstracker">

    <uses-permission android:name="android.permission.ACCESS_NETWORK_STATE"
/>
    <uses-permission android:name="android.permission.BLUETOOTH" />
    <uses-permission android:name="android.permission.BLUETOOTH_ADMIN" />
    <uses-permission android:name="android.permission.INTERNET" />
    <uses-permission android:name="android.permission.ACCESS_FINE_LOCATION" />
    <uses-permission
android:name="android.permission.ACCESS_COARSE_LOCATION" />

    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:supportsRtl="true"
        android:theme="@style/AppTheme">
        <activity
            android:name=".MainActivity"
            android:label="@string/app_name">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
        <activity android:name=".Main2Activity"></activity>
    </application>

</manifest>
```

```
import java.io.IOException;
import java.io.InputStream;
import java.io.OutputStream;
import java.lang.reflect.Method;
import java.util.UUID;
import android.app.Activity;
import android.bluetooth.BluetoothAdapter;
```

```
import android.bluetooth.BluetoothDevice;
import android.bluetooth.BluetoothSocket;
import android.content.Intent;
import android.os.Build;
import android.os.Bundle;
import android.os.Handler;
import android.util.Log;
import android.widget.TextView;
import android.widget.Toast;

import com.loopj.android.http.AsyncHttpClient;
import com.loopj.android.http.AsyncHttpResponseHandler;
import com.loopj.android.http.JsonHttpResponseHandler;
import com.loopj.android.http.RequestParams;

import org.json.JSONException;
import org.json.JSONObject;

import cz.msebera.android.httpclient.Header;

public class Main2Activity extends Activity {

    private static final String TAG = "Bus Locator";
    TextView txtArduino;
    Handler h;

    final int RECIEVE_MESSAGE = 1;
    private BluetoothAdapter btAdapter = null;
    private BluetoothSocket btSocket = null;
    private StringBuilder sb = new StringBuilder();

    private ConnectedThread mConnectedThread;

    RequestParams params = new RequestParams();

    private static final UUID MY_UUID = UUID.fromString("00001101-0000-1000-8000-00805F9B34FB");

    private static String address = "98:D3:31:FC:17:2B";

    private static String rfid_1 = "241 110 38 59";
    private static String rfid_2 = "58 49 125 126";

    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);

        setContentView(R.layout.activity_main2);
        txtArduino = (TextView) findViewById(R.id.txtArduino);
```

```

h = new Handler() {
    public void handleMessage(android.os.Message msg) {
        switch (msg.what) {
            case RECIEVE_MESSAGE:
                byte[] readBuf = (byte[]) msg.obj;
                String strIncom = new String(readBuf, 0, msg.arg1);
                sb.append(strIncom);
                int endOfLineIndex = sb.indexOf("\r\n");
                if (endOfLineIndex > 0) {
                    String sbprint = sb.substring(0, endOfLineIndex);
                    sb.delete(0, sb.length());
                    if(sbprint.contains("241 110 38 59")) {
                        txtArduino.append("User1" + "\n");
                        params.add("rfid", "2411103859");
                        ticketConfirmation();
                    }
                    else if(sbprint.contains("58 49 125 126")) {
                        txtArduino.append("User2" + "\n");           // update TextView
                        params.add("rfid", "5849125126");
                        ticketConfirmation();
                    }
                    else{
                        txtArduino.append("No user found. \n");
                    }
                }
                break;
            }
        }
    };
};

btAdapter = BluetoothAdapter.getDefaultAdapter();
checkBTState();
}

private BluetoothSocket createBluetoothSocket(BluetoothDevice device) throws
IOException {
    if(Build.VERSION.SDK_INT >= 10){
        try {
            final Method m =
device.getClass().getMethod("createInsecureRfcommSocketToServiceRecord", new Class[]
{ UUID.class });
            return (BluetoothSocket) m.invoke(device, MY_UUID);
        } catch (Exception e) {
            Log.e(TAG, "Could not create Insecure RFCOMM Connection",e);
        }
    }
    return device.createRfcommSocketToServiceRecord(MY_UUID);
}

```

@Override

```

public void onResume() {
    super.onResume();

    Log.d(TAG, "...onResume - try connect...");

    BluetoothDevice device = btAdapter.getRemoteDevice(address);
    try {
        btSocket = createBluetoothSocket(device);
    } catch (IOException e) {
        errorExit("Fatal Error", "In onResume() and socket create failed: " + e.getMessage()
+ ".");
    }
    btAdapter.cancelDiscovery();
    Log.d(TAG, "...Connecting...");
    try {
        btSocket.connect();
        Log.d(TAG, "...Connection ok...");
    } catch (IOException e) {
        try {
            btSocket.close();
        } catch (IOException e2) {
            errorExit("Fatal Error", "In onResume() and unable to close socket during
connection failure" + e2.getMessage() + ".");
        }
    }

    Log.d(TAG, "...Create Socket...");

    mConnectedThread = new ConnectedThread(btSocket);
    mConnectedThread.start();
}

@Override
public void onPause() {
    super.onPause();

    Log.d(TAG, "...In onPause()...");

    try {
        btSocket.close();
    } catch (IOException e2) {
        errorExit("Fatal Error", "In onPause() and failed to close socket." + e2.getMessage()
+ ".");
    }
}

private void checkBTState() {
    if(btAdapter==null) {
        errorExit("Fatal Error", "Bluetooth not support");
    } else {

```

```

    if (btAdapter.isEnabled()) {
        Log.d(TAG, "...Bluetooth ON...");
    } else {
        Intent enableBtIntent = new
Intent(BluetoothAdapter.ACTION_REQUEST_ENABLE);
        startActivityForResult(enableBtIntent, 1);
    }
}
}

private void errorExit(String title, String message){
    Toast.makeText(getBaseContext(), title + " - " + message,
Toast.LENGTH_LONG).show();
    finish();
}

private class ConnectedThread extends Thread {
    private final InputStream mmInStream;

    public ConnectedThread(BluetoothSocket socket) {
        InputStream tmpIn = null;

        try {
            tmpIn = socket.getInputStream();
        } catch (IOException e) {}

        mmInStream = tmpIn;
    }

    public void run() {
        byte[] buffer = new byte[256];
        int bytes;

        while (true) {
            try {
                // Read from the InputStream
                bytes = mmInStream.read(buffer); // Get number of bytes and message in
"buffer"
                h.obtainMessage(RECIEVE_MESSAGE, bytes, -1, buffer).sendToTarget();
            } catch (IOException e) {
                break;
            }
        }
    }
}

public void ticketConfirmation()
{
    AsyncHttpClient client = new AsyncHttpClient();

```

```

        client.post("http://api.spectrumtechbd.com/gpstracker/confirmation.php", params,
new AsyncHttpResponseHandler() {
    @Override
    public void onSuccess(int statusCode, Header[] headers, byte[] responseBody) {
        Toast.makeText(getApplicationContext(), "Ticket Confirmed Successfully.",
Toast.LENGTH_LONG).show();
    }

    @Override
    public void onFailure(int statusCode, Header[] headers, byte[] responseBody,
Throwable error) {
        Toast.makeText(getApplicationContext(), "THERE ARE SOME TROUBLE
CONNECTING", Toast.LENGTH_LONG).show();
    }
});
}
}
}

```

```

import android.app.AlertDialog;
import android.content.DialogInterface;
import android.content.Intent;
import android.net.ConnectivityManager;
import android.net.NetworkInfo;
import android.os.Bundle;
import android.provider.Settings;
import android.support.v7.app.ActionBarActivity;
import android.view.Menu;
import android.view.MenuItem;
import android.view.View;
import android.widget.Button;
import android.content.Context;
import android.location.Location;
import android.location.LocationListener;
import android.location.LocationManager;
import android.widget.Toast;

import com.loopj.android.http.*;
import cz.msebera.android.httpclient.Header;

public class MainActivity extends ActionBarActivity {
    String lat, lon;
    Button btn_start, btn_stop;
    private LocationManager locationManager;

```



```
private LocationListener locationlistener;
```

```
@Override
```

```
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);
    if (!isOnline()) {
        AlertDialog.Builder builder = new AlertDialog.Builder(this);
        builder.setCancelable(false);
        builder.setMessage("You don't have an active internet connection.");
        builder.setPositiveButton("Settings", new DialogInterface.OnClickListener() {
            @Override
            public void onClick(DialogInterface dialog, int which) {
                startActivity(new Intent(Settings.ACTION_SETTINGS));
            }
        });
        builder.setNegativeButton("Close", new DialogInterface.OnClickListener() {
            @Override
            public void onClick(DialogInterface dialog, int which) {
                finish();
                System.exit(0);
            }
        });
        AlertDialog alert = builder.create();
        alert.show();
    }
    btn_start = (Button) findViewById(R.id.button);
    btn_stop = (Button) findViewById(R.id.button2);
```

```
locationmanger = (LocationManager) getSystemService(LOCATION_SERVICE);
```

```
locationlistener = new LocationListener() {
```

```
    @Override
```

```
    public void onLocationChanged(Location location) {
```

```
        lat = location.getLatitude()+"";
```

```
        lon = location.getLongitude()+"";
```

```
        sendData(lat, lon);
```

```
    }
```

```
    @Override
```

```
    public void onStatusChanged(String provider, int status, Bundle extras) {
```

```
    }
```

```
    @Override
```

```
    public void onProviderEnabled(String provider) {
```

```
    }
```

```
    @Override
```

```

    public void onProviderDisabled(String provider) {
        startActivity(new Intent(Settings.ACTION_LOCATION_SOURCE_SETTINGS));
    }
};
btn_start.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        locationManager.requestLocationUpdates(locationmanger.GPS_PROVIDER, 3000,
0, locationlistener);
        locationManager.requestLocationUpdates(locationmanger.NETWORK_PROVIDER,
3000, 0, locationlistener);
        Toast.makeText(getApplicationContext(), "Location service started",
Toast.LENGTH_SHORT).show();
    }
});
btn_stop.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        locationManager = null;
        locationlistener = null;
        startActivity(new Intent(Settings.ACTION_LOCATION_SOURCE_SETTINGS));
        finish();
        System.exit(0);
    }
});
}

public void sendData(String lat, String lon)
{

}

public boolean isOnline() {
    ConnectivityManager cm =
        (ConnectivityManager) getSystemService(Context.CONNECTIVITY_SERVICE);
    NetworkInfo netInfo = cm.getActiveNetworkInfo();
    return netInfo != null && netInfo.isConnectedOrConnecting();
}

@Override
public boolean onCreateOptionsMenu(Menu menu) {
    // Inflate the menu; this adds items to the action bar if it is present.
    getMenuInflater().inflate(R.menu.menu_main, menu);
    return true;
}

@Override
public boolean onOptionsItemSelected(MenuItem item) {
    super.onOptionsItemSelected(item);
}

```

```
switch (item.getItemId()) {
    case R.id.item1:
        startActivity(new Intent(getApplicationContext(), Main2Activity.class));
        return true;
    case R.id.item2:
        finish();
        System.exit(0);
        return true;
    default:
        return super.onOptionsItemSelected(item);
}
}
```

Appendix C

BUS TICKET BOOKING

```
<?xml version="1.0" encoding="utf-8"?>
<manifest
xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.example.lenovo.gpstracker2">

    <!--
        The ACCESS_COARSE/FINE_LOCATION permissions are not
        required to use
        Google Maps Android API v2, but you must specify either
        coarse or fine
        location permissions for the 'MyLocation'
        functionality.
    -->
    <uses-permission
android:name="android.permission.ACCESS_NETWORK_STATE" />
    <uses-permission
android:name="android.permission.ACCESS_FINE_LOCATION" />

    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:supportsRtl="true"
        android:theme="@style/AppTheme">

        <!--
            The API key for Google Maps-based APIs is defined
```

as a string resource.

(See the file "res/values/google_maps_api.xml").

Note that the API key is linked to the encryption key used to sign the APK.

You need a different API key for each encryption key, including the release key that is used to sign the APK for publishing.

You can define the keys for the debug and release targets in src/debug/ and src/release/.

```
-->
<meta-data
    android:name="com.google.android.geo.API_KEY"
    android:value="@string/google_maps_key" />

<activity
    android:name=".MapsActivity"
    android:label="@string/title_activity_maps" />
<activity
    android:name=".FinishActivity"
    android:label="@string/title_activity_finsh" />
<activity
    android:name=".LoginActivity"
    android:label="@string/title_activity_login" />
<activity
    android:name=".MainActivity"
    android:label="@string/title_activity_main" />
<activity android:name=".StartActivity">
    <intent-filter>
        <action
            android:name="android.intent.action.MAIN" />

            <category
                android:name="android.intent.category.LAUNCHER" />
        </intent-filter>
    </activity>
<activity
    android:name=".TicketActivity"
    android:label="@string/title_activity_ticket"/>
</application>

</manifest>

import java.util.ArrayList;
import java.util.HashMap;
import java.util.Hashtable;
import android.content.ContentValues;
import android.content.Context;
import android.database.Cursor;
import android.database.DatabaseUtils;
```

```
import android.database.sqlite.SQLiteOpenHelper;
import android.database.sqlite.SQLiteDatabase;

public class DBHelper extends SQLiteOpenHelper{
    public static final String DATABASE_NAME = "MyDBName.db";
    public static final String USER_TABLE_NAME = "user";
    public static final String USER_COLUMN_ID = "id";
    public static final String USER_COLUMN_NAME = "name";
    public static final String USER_COLUMN_USERNAME =
"username";
    public static final String USER_COLUMN_BALANCE = "balance";

    private HashMap hp;

    public DBHelper(Context context)
    {
        super(context, DATABASE_NAME , null, 1);
    }

    @Override
    public void onCreate(SQLiteDatabase db) {
        // TODO Auto-generated method stub
        db.execSQL(
            "create table user " +
            "(id integer primary key, name text,
username text, balance integer)"
        );
    }

    @Override
    public void onUpgrade(SQLiteDatabase db, int oldVersion, int
newVersion) {
        // TODO Auto-generated method stub
        db.execSQL("DROP TABLE IF EXISTS user");
        onCreate(db);
    }

    public boolean insertUser(String name, String username,
Integer balance)
    {
        SQLiteDatabase db = this.getWritableDatabase();
        ContentValues contentValues = new ContentValues();
        contentValues.put("name", name);
        contentValues.put("username", username);
        contentValues.put("balance", balance);
        db.insert("user", null, contentValues);
        return true;
    }

    public Cursor getUser(int id){
```

```

        SQLiteDatabase db = this.getReadableDatabase();
        Cursor res = db.rawQuery( "select * from user where
id="+id+"", null );
        return res;
    }

    public boolean updateUser(Integer id, Integer balance)
    {
        SQLiteDatabase db = this.getWritableDatabase();
        ContentValues contentValues = new ContentValues();
        contentValues.put("balance", balance);
        db.update("user", contentValues, "id = ? ", new String[]
{ Integer.toString(id) } );
        return true;
    }

    public boolean deleteUser()
    {
        SQLiteDatabase db = this.getWritableDatabase();
        db.delete("user", null, null);
        return true;
    }
}

```

```

public class MainActivity extends ActionBarActivity {
    String lat, lon;
    Button btn_start, btn_stop;
    private LocationManager locationManager;
    private LocationListener locationlistener;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        if (!isOnline()) {
            AlertDialog.Builder builder = new
AlertDialog.Builder(this);
            builder.setCancelable(false);
            builder.setMessage("You don't have an active
internet connection.");
            builder.setPositiveButton("Settings", new
DialogInterface.OnClickListener() {
                @Override
                public void onClick(DialogInterface dialog, int
which) {
                    startActivity(new
Intent(Settings.ACTION_SETTINGS));

```

```

    }
    });
    builder.setNegativeButton("Close", new
DialogInterface.OnClickListener() {
    @Override
    public void onClick(DialogInterface dialog, int
which) {
        finish();
        System.exit(0);
    }
    });
    AlertDialog alert = builder.create();
    alert.show();
}
btn_start = (Button) findViewById(R.id.button);
btn_stop = (Button) findViewById(R.id.button2);

locationmanger = (LocationManager)
getSystemService(LOCATION_SERVICE);
locationlistener = new LocationListener() {
    @Override
    public void onLocationChanged(Location location) {
        lat = location.getLatitude()+"";
        lon = location.getLongitude()+"";
        sendData(lat, lon);
    }

    @Override
    public void onStatusChanged(String provider, int
status, Bundle extras) {

    }

    @Override
    public void onProviderEnabled(String provider) {

    }

    @Override
    public void onProviderDisabled(String provider) {
        startActivity(new
Intent(Settings.ACTION_LOCATION_SOURCE_SETTINGS));
    }
};
btn_start.setOnClickListener(new View.OnClickListener()
{
    @Override
    public void onClick(View v) {

locationmanger.requestLocationUpdates(locationmanger.GPS_PROVIDE

```

```
R, 3000, 0, locationlistener);

locationmanger.requestLocationUpdates(locationmanger.NETWORK_PRO
VIDER, 3000, 0, locationlistener);
    Toast.makeText(getApplicationContext(),
"Location service started", Toast.LENGTH_SHORT).show();
    }
});
btn_stop.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        locationmanger = null;
        locationlistener = null;
        startActivity(new
Intent(Settings.ACTION_LOCATION_SOURCE_SETTINGS));
        finish();
        System.exit(0);
    }
});
}

public void sendData(String lat, String lon)
{

}

public boolean isOnline() {
    ConnectivityManager cm =
        (ConnectivityManager)
getSystemService(Context.CONNECTIVITY_SERVICE);
    NetworkInfo netInfo = cm.getActiveNetworkInfo();
    return netInfo != null &&
netInfo.isConnectedOrConnecting();
}

@Override
public boolean onCreateOptionsMenu(Menu menu) {
    // Inflate the menu; this adds items to the action bar
if it is present.
    getMenuInflater().inflate(R.menu.menu_main, menu);
    return true;
}

@Override
public boolean onOptionsItemSelected(MenuItem item) {
    super.onOptionsItemSelected(item);

    switch (item.getItemId()) {
        case R.id.item1:
```



```

        startActivity(new
Intent(getApplicationContext(), Main2Activity.class));
        return true;
    case R.id.item2:
        finish();
        System.exit(0);
        return true;
    default:
        return super.onOptionsItemSelected(item);
    }
}
}

```

```

import android.database.Cursor;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.TextView;
import android.widget.Toast;
import com.loopj.android.http.AsyncHttpClient;
import com.loopj.android.http.JsonHttpResponseHandler;
import org.json.JSONException;
import org.json.JSONObject;
import cz.msebera.android.httpclient.Header;

public class FinishActivity extends AppCompatActivity {
    Button btnExit;
    TextView tvStatus;
    DBHelper mydb;
    private static final String TAG_SEAT = "seat";
    private static final String TAG_USER = "user";
    String user;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_finish);
        mydb = new DBHelper(this);
        Toast.makeText(getApplicationContext(), "You have
successfully purchased ticket", Toast.LENGTH_LONG).show();
        btnExit = (Button) findViewById(R.id.button3);
        tvStatus = (TextView) findViewById(R.id.textView3);
        Cursor rs = mydb.getUser(1);
        rs.moveToFirst();
        final String name =
rs.getString(rs.getColumnIndex(DBHelper.USER_COLUMN_NAME));
        final String username =
rs.getString(rs.getColumnIndex(DBHelper.USER_COLUMN_USERNAME));
        final int cbalance =

```

```
rs.getInt(rs.getColumnIndex(DBHelper.USER_COLUMN_BALANCE));
    if(!rs.isClosed()){
        rs.close();
    }
    tvStatus.setText(name+"\n Your current Balance is BDT "+
cbalance+ "\n Seat booked: ");
    getBookedSeat();
    btnExit.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View view) {
            Toast.makeText(getApplicationContext(), "Thank
you for purchasing ticket.", Toast.LENGTH_LONG).show();
            finish();
            System.exit(0);
        }
    });
}

private void getBookedSeat(){
}
}
```

```
import android.content.Intent;
import android.support.v7.app.ActionBarActivity;
import android.os.Bundle;
import android.view.Menu;
import android.view.MenuItem;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Toast;

import com.loopj.android.http.AsyncHttpClient;
import com.loopj.android.http.JsonHttpResponseHandler;
import com.loopj.android.http.RequestParams;

import org.json.JSONException;
import org.json.JSONObject;

import cz.msebera.android.httpclient.Header;

public class LoginActivity extends ActionBarActivity {

    EditText etUser, etPass;
    Button btnLogin, btnSignup;
    SessionManager session;
    DBHelper mydb;
```

```

private static final String TAG_SUCCESS = "success";
private static final String TAG_MESSAGE = "message";
private static final String TAG_USERNAME = "username";
private static final String TAG_NAME = "name";
private static final String TAG_BALANCE = "balance";

@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_login);
    mydb = new DBHelper(this);
    session = new SessionManager(getApplicationContext());

    etUser = (EditText) findViewById(R.id.et_username);
    etPass = (EditText) findViewById(R.id.et_password);
    btnLogin = (Button) findViewById(R.id.btn_login);
    btnSignup = (Button) findViewById(R.id.btn_signup);
    btnLogin.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            checkLogin();
        }
    });

    btnSignup.setOnClickListener(new View.OnClickListener()
{
    @Override
    public void onClick(View v) {
        //Intent intent = new
Intent(getApplicationContext(), RegistrationActivity.class);
        //startActivity(intent);
        //finish();
    }
});

}
private void checkLogin() {
    return true;
}

@Override
public boolean onCreateOptionsMenu(Menu menu) {
    // Inflate the menu; this adds items to the action bar
if it is present.
    getMenuInflater().inflate(R.menu.menu_normal, menu);
    return true;
}

```

```
@Override
public boolean onOptionsItemSelected(MenuItem item) {
    super.onOptionsItemSelected(item);

    switch (item.getItemId()) {
        case R.id.item1:
            return true;
        case R.id.item2:
            finish();
            System.exit(0);
            return true;
        default:
            return super.onOptionsItemSelected(item);
    }
}

import android.content.Context;
import android.content.Intent;
import android.database.Cursor;
import android.net.ConnectivityManager;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.Menu;
import android.view.MenuItem;
import android.view.View;
import android.widget.ArrayAdapter;
import android.widget.Button;
import android.widget.Spinner;
import android.widget.TextView;
import android.widget.Toast;

import com.loopj.android.http.AsyncHttpClient;
import com.loopj.android.http.JsonHttpResponseHandler;
import com.loopj.android.http.RequestParams;

import org.json.JSONArray;
import org.json.JSONException;
import org.json.JSONObject;

import java.util.HashMap;

import cz.msebera.android.httpclient.Header;

public class MainActivity extends AppCompatActivity {
    SessionManager session;
```

```

DBHelper mydb;
Spinner spFrom, spTo;
Button btnSearch;
TextView tvInfo;
RequestParams params = new RequestParams();
private static final String TAG_HISTORY = "history";

@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);
    session = new SessionManager(getApplicationContext());
    spFrom = (Spinner)findViewById(R.id.spinner);
    spTo = (Spinner)findViewById(R.id.spinner2);
    btnSearch = (Button)findViewById(R.id.button);
    tvInfo = (TextView)findViewById(R.id.textView);

    mydb = new DBHelper(this);
    Cursor rs = mydb.getUser(1);
    rs.moveToFirst();
    final String username =
rs.getString(rs.getColumnIndex(DBHelper.USER_COLUMN_USERNAME));
    params.add("username", username);
    final String name =
rs.getString(rs.getColumnIndex(DBHelper.USER_COLUMN_NAME));
    final int balance =
rs.getInt(rs.getColumnIndex(DBHelper.USER_COLUMN_BALANCE));
    tvInfo.setText("\n"+"Name: "+name+"\n"+"Current Balance:
"+balance);

    final ArrayAdapter<CharSequence> start_pointAdapter =
ArrayAdapter.createFromResource(
        this, R.array.starting_point,
        android.R.layout.simple_spinner_item);

    start_pointAdapter.setDropDownViewResource(android.R.layout.simple_spinner_dropdown_item);
    spFrom.setAdapter(start_pointAdapter);

    final ArrayAdapter<CharSequence> end_pointAdapter =
ArrayAdapter.createFromResource(
        this, R.array.ending_point,
        android.R.layout.simple_spinner_item);

    start_pointAdapter.setDropDownViewResource(android.R.layout.simple_spinner_dropdown_item);
    spTo.setAdapter(end_pointAdapter);

    btnSearch.setOnClickListener(new View.OnClickListener()
{

```

```
        @Override
        public void onClick(View view) {
            if(spFrom.getSelectedItem().equals("Select
Place") || spTo.getSelectedItem().equals("Select Place"))
            {
                Toast.makeText(getApplicationContext(),"Please select
destination",Toast.LENGTH_LONG).show();
            }
            else if(spFrom.getSelectedItem() ==
spTo.getSelectedItem())
            {
                Toast.makeText(getApplicationContext(),"You
select the same place",Toast.LENGTH_LONG).show();
            }
            else {
                startActivity(new
Intent(getApplicationContext(), MapsActivity.class));
                finish();
            }
        }
    });
}

private void clearHistory() {
    return true;
}

@Override
public boolean onCreateOptionsMenu(Menu menu) {
    // Inflate the menu; this adds items to the action bar
if it is present.
    getMenuInflater().inflate(R.menu.menu_main, menu);
    return true;
}

@Override
public boolean onOptionsItemSelected(MenuItem item) {
    super.onOptionsItemSelected(item);

    switch (item.getItemId()) {
        case R.id.item1:
            return true;
        case R.id.item2:
            mydb.deleteUser();
            session.logoutUser();
            finish();
    }
}
```

```
        return true;
    case R.id.item3:
        mydb.deleteUser();
        session.logoutUser();
        clearHistory();
        finish();
        return true;
    case R.id.item4:
        finish();
        System.exit(0);
        return true;
    default:
        return super.onOptionsItemSelected(item);
    }
}
}
```

```
import android.content.Intent;
import android.graphics.Color;
import android.graphics.drawable.BitmapDrawable;
import android.location.Location;
import android.support.v4.app.FragmentActivity;
import android.os.Bundle;
import android.support.v7.app.AppCompatActivity;
import android.view.Menu;
import android.view.MenuItem;
import android.widget.Toast;

import com.google.android.gms.maps.CameraUpdateFactory;
import com.google.android.gms.maps.GoogleMap;
import com.google.android.gms.maps.OnMapReadyCallback;
import com.google.android.gms.maps.SupportMapFragment;
import
com.google.android.gms.maps.model.BitmapDescriptorFactory;
import com.google.android.gms.maps.model.LatLng;
import com.google.android.gms.maps.model.Marker;
import com.google.android.gms.maps.model.MarkerOptions;

import com.google.android.gms.maps.model.Polyline;
import com.google.android.gms.maps.model.PolylineOptions;
import com.loopj.android.http.*;
import org.json.JSONArray;
import org.json.JSONException;
import org.json.JSONObject;

import java.util.ArrayList;
```

```
import cz.msebera.android.httpclient.Header;

public class MapsActivity extends AppCompatActivity implements
    OnMapReadyCallback {

    private GoogleMap mMap;
    private static final String TAG_LATITUDE = "latitude";
    private static final String TAG_LONGITUDE = "longitude";

    String coordinate;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_maps);
        // Obtain the SupportMapFragment and get notified when
        the map is ready to be used.
        SupportMapFragment mapFragment = (SupportMapFragment)
            getSupportFragmentManager()
                .findFragmentById(R.id.map);
        mapFragment.getMapAsync(this);
    }

    /**
     * Manipulates the map once available.
     * This callback is triggered when the map is ready to be
     used.
     * This is where we can add markers or lines, add listeners
     or move the camera. In this case,
     * we just add a marker near Sydney, Australia.
     * If Google Play services is not installed on the device,
     the user will be prompted to install
     * it inside the SupportMapFragment. This method will only
     be triggered once the user has
     * installed Google Play services and returned to the app.
     */

    @Override
    public void onMapReady(GoogleMap googleMap) {
        mMap = googleMap;
        String coordinate = "23.45463,90.46732";
        String [] coordinate_array = coordinate.split(",");
        double lat = Double.parseDouble(coordinate_array[0]);
        double lng = Double.parseDouble(coordinate_array[1]);
        LatLng loc = new LatLng(lat, lng);
        mMap.addMarker(new
MarkerOptions().position(loc).title("Car Tracker"));
        mMap.moveCamera(CameraUpdateFactory.newLatLng(loc));
        mMap.moveCamera(CameraUpdateFactory.newLatLngZoom(loc,
13));
```



```
Thread t = new Thread() {

    @Override
    public void run() {
        try {
            while (!isInterrupted()) {
                Thread.sleep(3000);
                runOnUiThread(new Runnable() {
                    @Override
                    public void run() {
                        getBus();
                    }
                });
            }
        } catch (InterruptedException e) {
        }
    }
};

t.start();
}

private void getBus() {

}

public void onBackPressed() {
    startActivity(new Intent(getApplicationContext(),
MainActivity.class));
    finish();
}

@Override
public boolean onCreateOptionsMenu(Menu menu) {
    // Inflate the menu; this adds items to the action bar
if it is present.
    getMenuInflater().inflate(R.menu.menu_normal, menu);
    return true;
}

@Override
public boolean onOptionsItemSelected(MenuItem item) {
    super.onOptionsItemSelected(item);

    switch (item.getItemId()) {
        case R.id.item1:
            return true;
    }
}
```

```
        case R.id.item2:
            finish();
            System.exit(0);
            return true;
        default:
            return super.onOptionsItemSelected(item);
    }
}

import android.content.Context;
import android.content.Intent;
import android.content.SharedPreferences;
import android.content.SharedPreferences.Editor;

import java.util.HashMap;

/**
 * Created by lenovo on 27/12/2016.
 */
public class SessionManager {
    SharedPreferences pref;
    Editor editor;
    Context _context;

    int PRIVATE_MODE = 0;

    // Shared Preferences file name
    private static final String PREF_NAME = "busticket";
    private static final String IS_LOGIN = "IsLoggedIn";
    public static final String KEY_USERNAME = "uname";

    public SessionManager(Context context) {
        this._context = context;
        pref = _context.getSharedPreferences(PREF_NAME,
PRIVATE_MODE);
        editor = pref.edit();
    }

    public void createLoginSession(String username) {
        // Storing login value as TRUE
        editor.putBoolean(IS_LOGIN, true);
        // Storing username in pref
        editor.putString(KEY_USERNAME, username);
        // commit changes
        editor.commit();
    }
}
```

```
public boolean checkLogin() {

    if (!this.isLoggedIn()) {
        return true;
    }
    return false;
}

public HashMap<String, String> getUserDetails() {
    HashMap<String, String> user = new HashMap<String,
String>();
    // user name
    user.put(KEY_USERNAME, pref.getString(KEY_USERNAME,
null));
    // return user
    return user;
}

/**
 * Clear session details
 */
public void logoutUser() {
    // Clearing all data from Shared Preferences
    editor.clear();
    editor.commit();

    _context.startActivity(new Intent(_context,
LoginActivity.class)
        .addFlags(Intent.FLAG_ACTIVITY_CLEAR_TOP)
        .setFlags(Intent.FLAG_ACTIVITY_NEW_TASK));
}

public boolean isLoggedIn() {
    return pref.getBoolean(IS_LOGIN, false);
}
}

import android.app.ProgressDialog;
import android.content.Context;
import android.content.DialogInterface;
import android.content.Intent;
import android.graphics.Color;
import android.net.ConnectivityManager;
import android.provider.Settings;
import android.support.v7.app.AlertDialog;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
```

```

import android.view.Menu;
import android.view.MenuItem;
import android.widget.ProgressBar;

public class StartActivity extends AppCompatActivity {
    SessionManager session;
    private ProgressBar mProgress;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_start);
        session = new SessionManager(getApplicationContext());
        if (!isOnline()) {
            AlertDialog.Builder builder = new
AlertDialog.Builder(this);
            builder.setCancelable(false);
            builder.setMessage("You don't have an active
internet connection.");
            builder.setPositiveButton("Settings", new
DialogInterface.OnClickListener() {
                @Override
                public void onClick(DialogInterface dialog, int
which) {
                    startActivity(new
Intent(Settings.ACTION_SETTINGS));
                }
            });
            builder.setNegativeButton("Close", new
DialogInterface.OnClickListener() {
                @Override
                public void onClick(DialogInterface dialog, int
which) {
                    finish();
                    System.exit(0);
                }
            });
            AlertDialog alert = builder.create();
            alert.show();
        } else {
            mProgress = (ProgressBar)
findViewById(R.id.splash_screen_progress_bar);
            mProgress.getProgressDrawable().setColorFilter(
                Color.argb(255, 240, 131, 22),
                android.graphics.PorterDuff.Mode.SRC_IN);

            new Thread(new Runnable() {
                public void run() {
                    doWork();
                    startApp();
                    finish();
                }
            }).start();
        }
    }
}

```

```
        }
    }).start();
}

private void doWork() {
    for (int progress=0; progress<100; progress+=20) {
        try {
            Thread.sleep(1000);
            mProgress.setProgress(progress);
        } catch (Exception e) {
            e.printStackTrace();
        }
    }
}

private void startApp() {
    if(session.checkLogin()){
        startActivity(new Intent(getApplicationContext(),
LoginActivity.class));
    }
    else {
        startActivity(new Intent(getApplicationContext(),
MainActivity.class));
    }
}

public boolean isOnline() {
    ConnectivityManager cm =
        (ConnectivityManager)
getSystemService(Context.CONNECTIVITY_SERVICE);

    return cm.getActiveNetworkInfo() != null &&
cm.getActiveNetworkInfo().isConnectedOrConnecting();
}

@Override
public boolean onCreateOptionsMenu(Menu menu) {
    // Inflate the menu; this adds items to the action bar
if it is present.
    getMenuInflater().inflate(R.menu.menu_normal, menu);
    return true;
}

@Override
public boolean onOptionsItemSelected(MenuItem item) {
    super.onOptionsItemSelected(item);

    switch (item.getItemId()) {
```

```
        case R.id.item1:
            return true;
        case R.id.item2:
            finish();
            System.exit(0);
            return true;
        default:
            return super.onOptionsItemSelected(item);
    }
}

import android.content.Intent;
import android.database.Cursor;
import android.graphics.Color;
import android.os.Handler;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.Menu;
import android.view.MenuItem;
import android.view.View;
import android.view.animation.Animation;
import android.view.animation.AnimationUtils;
import android.widget.Button;
import android.widget.CheckBox;
import android.widget.EditText;
import android.widget.TextView;
import android.widget.Toast;

import com.loopj.android.http.AsyncHttpClient;
import com.loopj.android.http.JsonHttpResponseHandler;
import com.loopj.android.http.RequestParams;

import org.json.JSONException;
import org.json.JSONObject;

import java.util.ArrayList;
import java.util.Arrays;

import cz.msebera.android.httpclient.Header;

import static android.R.attr.delay;

public class TicketActivity extends AppCompatActivity {
    Button btnDone;
    CheckBox ch1, ch2, ch3, ch4, ch5, ch6, ch7, ch8, ch9, ch10,
ch11, ch12, ch13, ch14, ch15, ch16;
    EditText et1;
    TextView tv1, tv2;
```

```

int tk;
DBHelper mydb;
RequestParams params = new RequestParams();
private static final String TAG_SUCCESS = "success";
private static final String TAG_SEAT = "seat";
private static final String TAG_USER = "user";
private int c = 0;
private int [] seat = new int[16];
String username, user;
@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_ticket);
    getBookedSeat();
    mydb = new DBHelper(this);
    Cursor rs = mydb.getUser(1);
    rs.moveToFirst();
    username =
rs.getString(rs.getColumnIndex(DBHelper.USER_COLUMN_USERNAME));
    params.put("username", (CharSequence)username);
    final int balance =
rs.getInt(rs.getColumnIndex(DBHelper.USER_COLUMN_BALANCE));
    if (!rs.isClosed()) {
        rs.close();
    }

    tv1 = (TextView) findViewById(R.id.tv_log);
    tv2 = (TextView) findViewById(R.id.tv_log2);

    ArrayList<TextView> answerViewCollection = new
ArrayList<TextView>(2);
    answerViewCollection.add(tv2);
    answerViewCollection.add(tv1);

    for (int i = 0; i < answerViewCollection.size(); i++) {
        final View auxView = answerViewCollection.get(i);

        final Animation exitAnim =
AnimationUtils.makeOutAnimation(this,
            true);
        exitAnim.setDuration(3000);
        exitAnim.setStartOffset(delay * i);
        auxView.startAnimation(exitAnim);
        auxView.setVisibility(View.INVISIBLE);
    }

    ch1 = (CheckBox) findViewById(R.id.checkBox);
    ch2 = (CheckBox) findViewById(R.id.checkBox2);
    ch3 = (CheckBox) findViewById(R.id.checkBox3);
    ch4 = (CheckBox) findViewById(R.id.checkBox4);

```

```
ch5 = (CheckBox) findViewById(R.id.checkBox5);
ch6 = (CheckBox) findViewById(R.id.checkBox6);
ch7 = (CheckBox) findViewById(R.id.checkBox7);
ch8 = (CheckBox) findViewById(R.id.checkBox8);
ch9 = (CheckBox) findViewById(R.id.checkBox9);
ch10 = (CheckBox) findViewById(R.id.checkBox10);
ch11 = (CheckBox) findViewById(R.id.checkBox11);
ch12 = (CheckBox) findViewById(R.id.checkBox12);
ch13 = (CheckBox) findViewById(R.id.checkBox13);
ch14 = (CheckBox) findViewById(R.id.checkBox14);
ch15 = (CheckBox) findViewById(R.id.checkBox15);
ch16 = (CheckBox) findViewById(R.id.checkBox16);
et1 = (EditText) findViewById(R.id.et_fare);
btnDone = (Button) findViewById(R.id.button2);
et1.setEnabled(false);

ch1.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        if (ch1.isChecked()) {
            ch1.setText("Booked");
            ch1.setBackgroundColor(Color.rgb(255,
92,184,92)); // selected
            tk += 20; c++;
            seat[0]=1;
        } else {
            ch1.setText("1");
            tk -= 20; c--;
            seat[0]=0;
            ch1.setBackgroundColor(Color.rgb(255,
211,211,211));
        }
        et1.setText("Fare BDT " + tk);
    }
});
ch2.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        if (ch2.isChecked()) {
            ch2.setText("Booked");
            ch2.setBackgroundColor(Color.rgb(255,
92,184,92)); // selected
            tk += 20; c++;
            seat[1]=2;
        } else {
            ch2.setText("2");
            tk -= 20; c--;
            seat[1]=0;
            ch2.setBackgroundColor(Color.rgb(255,
211,211,211));
        }
    }
});
```



```
    }
    et1.setText("Fare BDT " + tk);
}
});
ch3.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        if (ch3.isChecked()) {
            ch3.setText("Booked");
            ch3.setBackgroundColor(Color.rgb(255,
92,184,92)); // selected
            tk += 20; c++;
            seat[2]=3;
        } else {
            ch3.setText("3");
            tk -= 20; c--;
            seat[2]=0;
            ch3.setBackgroundColor(Color.rgb(255,
211,211,211));
        }
        et1.setText("Fare BDT " + tk);
    }
});
ch4.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        if (ch4.isChecked()) {
            ch4.setText("Booked");
            ch4.setBackgroundColor(Color.rgb(255,
92,184,92)); // selected
            tk += 20; c++;
            seat[3]=4;
        } else {
            ch4.setText("4");
            tk -= 20; c--;
            seat[3]=0;
            ch4.setBackgroundColor(Color.rgb(255,
211,211,211));
        }
        et1.setText("Fare BDT " + tk);
    }
});
ch5.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        if (ch5.isChecked()) {
            ch5.setText("Booked");
            ch5.setBackgroundColor(Color.rgb(255,
92,184,92)); // selected
            tk += 20; c++;
```

```
        seat[4]=5;
    } else {
        ch5.setText("5");
        tk -= 20; c--;
        seat[4]=0;
        ch5.setBackgroundColor(Color.rgb(255,
211,211,211));
    }
    et1.setText("Fare BDT " + tk);
}
});
ch6.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        if (ch6.isChecked()) {
            ch6.setText("Booked");
            ch6.setBackgroundColor(Color.rgb(255,
92,184,92)); // selected
            tk += 20; c++;
            seat[5]=6;
        } else {
            ch6.setText("6");
            tk -= 20; c--;
            seat[5]=0;
            ch6.setBackgroundColor(Color.rgb(255,
211,211,211));
        }
        et1.setText("Fare BDT " + tk);
    }
});
ch7.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        if (ch7.isChecked()) {
            ch7.setText("Booked");
            ch7.setBackgroundColor(Color.rgb(255,
92,184,92)); // selected
            tk += 20; c++;
            seat[6]=7;
        } else {
            ch7.setText("7");
            tk -= 20; c--;
            seat[6]=0;
            ch7.setBackgroundColor(Color.rgb(255,
211,211,211));
        }
        et1.setText("Fare BDT " + tk);
    }
});
ch8.setOnClickListener(new View.OnClickListener() {
```

```
@Override
public void onClick(View view) {
    if (ch8.isChecked()) {
        ch8.setText("Booked");
        ch8.setBackgroundColor(Color.rgb(255,
92,184,92)); // selected
        tk += 20; c++;
        seat[7]=8;
    } else {
        ch8.setText("8");
        tk -= 20; c--;
        seat[7]=0;
        ch8.setBackgroundColor(Color.rgb(255,
211,211,211));
    }
    et1.setText("Fare BDT " + tk);
}
});
ch9.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        if (ch9.isChecked()) {
            ch9.setText("Booked");
            ch9.setBackgroundColor(Color.rgb(255,
92,184,92)); // selected
            tk += 20; c++;
            seat[8]=9;
        } else {
            ch9.setText("9");
            tk -= 20; c--;
            seat[8]=0;
            ch9.setBackgroundColor(Color.rgb(255,
211,211,211));
        }
        et1.setText("Fare BDT " + tk);
    }
});
ch10.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        if (ch10.isChecked()) {
            ch10.setText("Booked");
            ch10.setBackgroundColor(Color.rgb(255,
92,184,92)); // selected
            tk += 20; c++;
            seat[9]=10;
        } else {
            ch10.setText("10");
            tk -= 20; c--;
            seat[9]=0;
        }
    }
});
```

```
        ch10.setBackgroundColor(Color.rgb(255,
211,211,211));
    }
    et1.setText("Fare BDT " + tk);
}
});
ch11.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        if (ch11.isChecked()) {
            ch11.setText("Booked");
            ch11.setBackgroundColor(Color.rgb(255,
92,184,92)); // selected
            tk += 20; c++;
            seat[10]=11;
        } else {
            ch11.setText("11");
            tk -= 20; c--;
            seat[10]=0;
            ch11.setBackgroundColor(Color.rgb(255,
211,211,211));
        }
        et1.setText("Fare BDT " + tk);
    }
});
ch12.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        if (ch12.isChecked()) {
            ch12.setText("Booked");
            ch12.setBackgroundColor(Color.rgb(255,
92,184,92)); // selected
            tk += 20; c++;
            seat[11]=12;
        } else {
            ch12.setText("12");
            tk -= 20; c--;
            seat[11]=0;
            ch12.setBackgroundColor(Color.rgb(255,
211,211,211));
        }
        et1.setText("Fare BDT " + tk);
    }
});
ch13.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        if (ch13.isChecked()) {
            ch13.setText("Booked");
            ch13.setBackgroundColor(Color.rgb(255,
```

```
92,184,92)); // selected
        tk += 20; c++;
        seat[12]=13;
    } else {
        ch13.setText("13");
        tk -= 20; c--;
        seat[12]=0;
        ch13.setBackgroundColor(Color.rgb(255,
211,211,211));
    }
    et1.setText("Fare BDT " + tk);
}
});
ch14.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        if (ch14.isChecked()) {
            ch14.setText("Booked");
            ch14.setBackgroundColor(Color.rgb(255,
92,184,92)); // selected
            tk += 20; c++;
            seat[13]=14;
        } else {
            ch14.setText("14");
            tk -= 20; c--;
            seat[13]=0;
            ch14.setBackgroundColor(Color.rgb(255,
211,211,211));
        }
        et1.setText("Fare BDT " + tk);
    }
});
ch15.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        if (ch15.isChecked()) {
            ch15.setText("Booked");
            ch15.setBackgroundColor(Color.rgb(255,
92,184,92)); // selected
            tk += 20; c++;
            seat[14]=15;
        } else {
            ch15.setText("15");
            tk -= 20; c--;
            seat[14]=0;
            ch15.setBackgroundColor(Color.rgb(255,
211,211,211));
        }
        et1.setText("Fare BDT " + tk);
    }
}
```

```

});
ch16.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        if (ch16.isChecked()) {
            ch16.setText("Booked");
            ch16.setBackgroundColor(Color.rgb(255,
92,184,92)); // selected
            tk += 20; c++;
            seat[15]=16;
        } else {
            ch16.setText("16");
            tk -= 20; c--;
            seat[15]=0;
            ch16.setBackgroundColor(Color.rgb(255,
211,211,211));
        }
        et1.setText("Fare BDT " + tk);
    }
});

btnDone.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        if (c == 0) {
            Toast.makeText(getApplicationContext(), "You
didn't select any seat.", Toast.LENGTH_LONG).show();
        }
        else if(user.equals(username)){
            Toast.makeText(getApplicationContext(),
"Sorry, you've already booked two seats.",
Toast.LENGTH_LONG).show();
        }
        else if(c > 2){
            Toast.makeText(getApplicationContext(), "You
can't book more than two seats.", Toast.LENGTH_LONG).show();
        }
        else if(tk > balance){
            Toast.makeText(getApplicationContext(), "You
don't have sufficient money.", Toast.LENGTH_LONG).show();
        }
        else{
            int tbalance = balance - tk;
            mydb.updateUser(1, tbalance);
            bookSeat();
        }
    }
});

```

```
}

private void getBookedSeat () {

}

private void seatBooked (CheckBox x) {
    x.setEnabled (false);
    x.setText ("Booked");
    x.setBackgroundColor (Color.argb (255, 221, 55, 55));
}

private void bookSeat () {

}

@Override
public boolean onCreateOptionsMenu (Menu menu) {
    // Inflate the menu; this adds items to the action bar
    if it is present.
    getMenuInflater ().inflate (R.menu.menu_normal, menu);
    return true;
}

@Override
public boolean onOptionsItemSelected (MenuItem item) {
    super.onOptionsItemSelected (item);

    switch (item.getItemId ()) {
        case R.id.item1:
            return true;
        case R.id.item2:
            finish ();
            System.exit (0);
            return true;
        default:
            return super.onOptionsItemSelected (item);
    }
}

public void onBackPressed () {
    startActivity (new Intent (getApplicationContext (),
MapsActivity.class));
    finish ();
}
```

}

}

