SPL-2 Project Report (User Manual), 2020

BUS AUTOMATION SYSTEM

Course: Software Project Lab II

Course No: SE 505

Submitted by

Sadman Sakib, BSSE 1005

Bashirul Islam, BSSE 1022

BSSE Session: 2017-2018

Supervised by

Professor Dr. Zerina Begum

Designation: Professor

Institute of Information Technology

University of Dhaka

28-08-2021

Table of Contents

Index of Figures

Acknowledgement

We are highly indebted for getting such a wonderful opportunity to prepare the user manual report of the project ‘Bus Automation System- DU CHAKKA’. We would like to thank whole-heartedly to our supervisor, Dr. Zerina Begum, Professor, Institute of Information Technology, University of Dhaka for giving us guidelines on preparing this report.

Abstract

Our objective of the project is to make an automated University Bus System. It will be a web application. The university students and the officers will be the users of the system. They have to create a user account to use this app. This application will provide us the location of the bus. It will also bear information about the schedule of up-trips and the down-trips of each bus. This schedule will be updated dynamically. It will also dynamically show the current location of the bus. There will be a notice part which will give the user the present information about the bus. The users can also give feedback.

1. Introduction

Our objective of the project is to make an automated University Bus System. It will be a web application. The university students and the officers will be the users of the system. They have to create a user account to use this app. This application will provide us the location of the bus. It will also bear information about the schedule of up-trips and the down-trips of each bus. This schedule will be updated dynamically. It will also dynamically show the current location of the bus. There will be a notice part which will give the user the present information about the bus. The users can also give feedback.

1.1 Background study

- ❖ The university has **16** routes to commute its non-residential students.
- ❖ There are **75** regular buses on these routes.
- ❖ About **10**,**000** users are using the university bus.

What we have learnt from the students and office goers-

- **60% users** of the bus miss the university bus as they do not know the summer and winter schedule correctly
- **30%** knows the schedule
- **10% students** (non-resident) do not use the university bus

1.2 Challenges

During this project, we have to face a lot of challenges. First of all, we have to learn flutter as an android application. Implementing sending notifications to the users was very difficult for us. Besides, I have to handle large code together.

1.3. Project Overview

We have divided my whole project into different parts. They are-

- Getting longitude and latitude from the google map
- Providing schedule of different bus route
- Profile creation and edit
- Database handling
- Sending notification to the users
- Storing feedback from the users
- Showing bus location

CHAPTER 2: SETTING UP THE APP

Step 1:
Download the apk file from the following link Google drive link:

https://drive.google.com/file/d/1Qw1xQPAipgkP7wnR1vhzkVepk1cglDfw/view

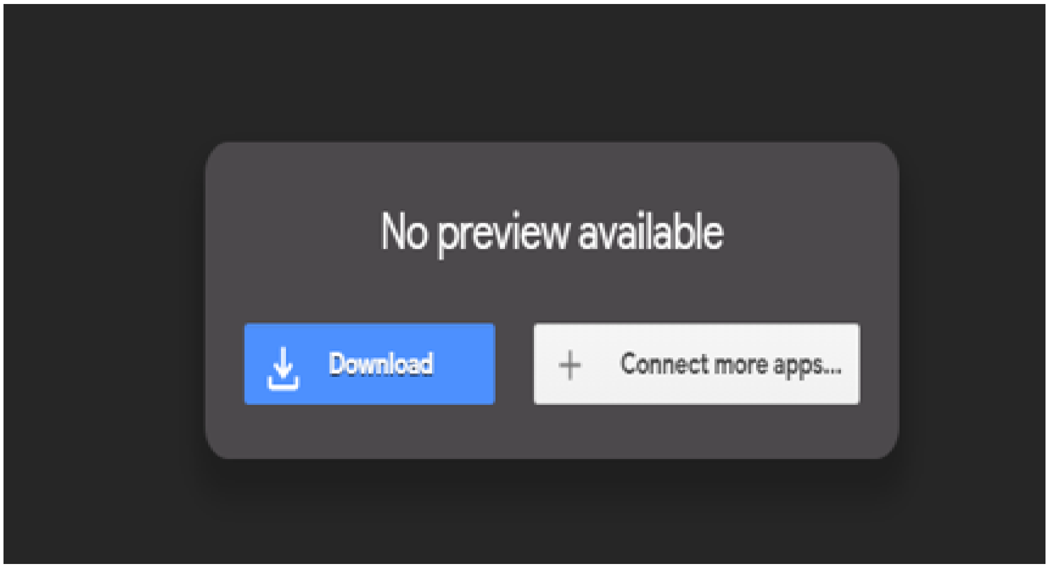

No preview available

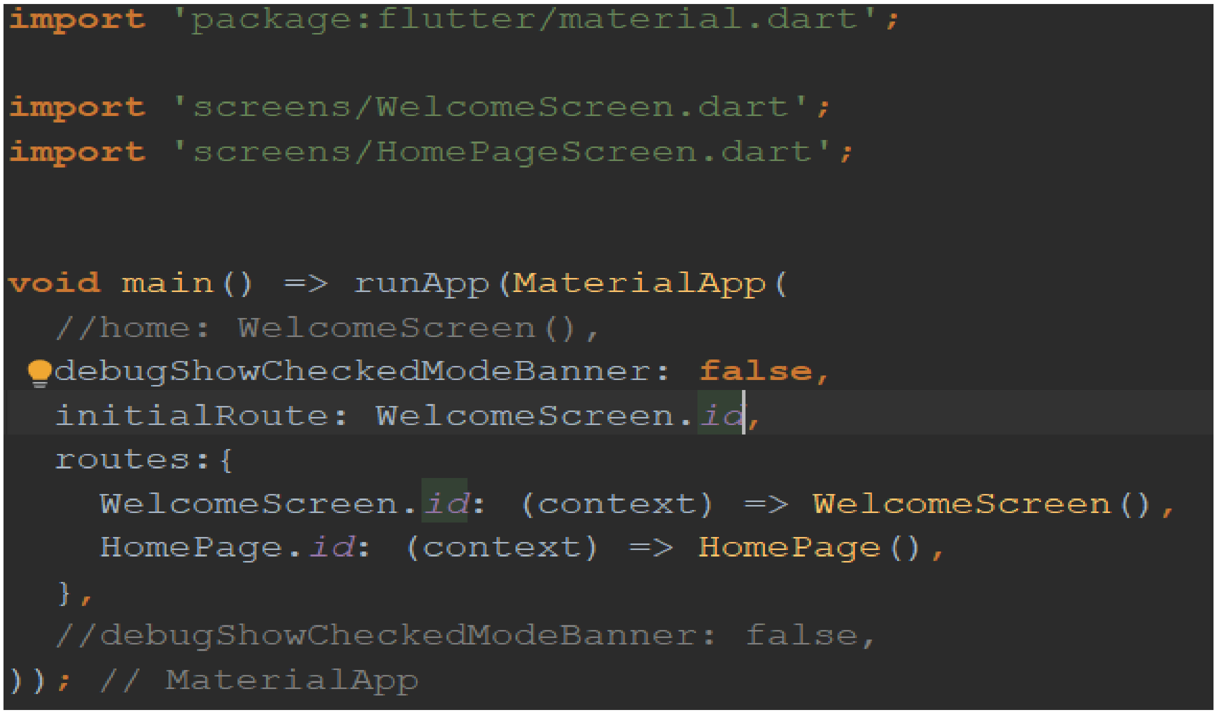
Figure 1: Installation

Step 2:
Install the app in your mobile.

CHAPTER 3: SOURCE CODE DESCRIPTION

3.1 Main class

In the main class, we initialize the route to welcome screen.

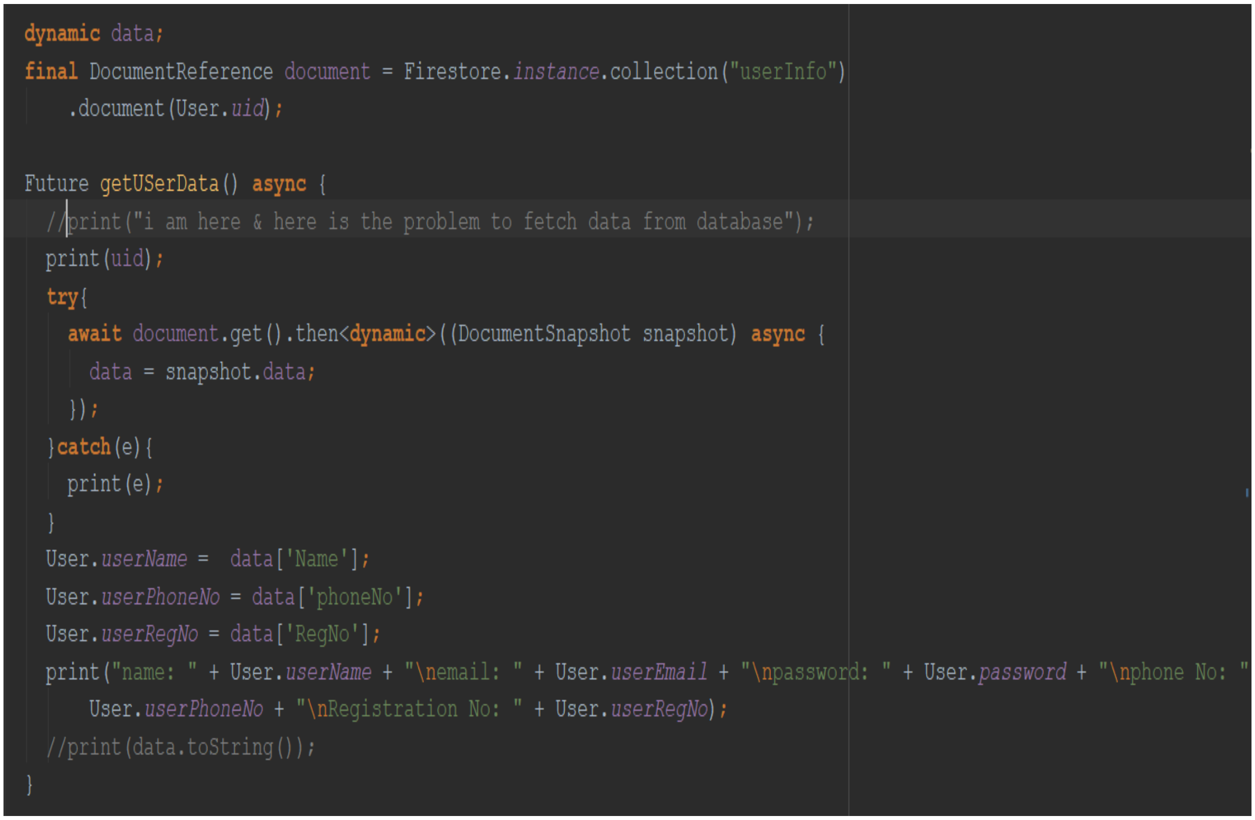


```dart
import 'package:flutter/material.dart';

import 'screens/WelcomeScreen.dart';
import 'screens/HomePageScreen.dart';


void main() => runApp(MaterialApp(
  //home: WelcomeScreen(),
  debugShowCheckedModeBanner: false,
  initialRoute: WelcomeScreen.id,
  routes:{
    WelcomeScreen.id: (context) => WelcomeScreen(),
    HomePage.id: (context) => HomePage(),
  },
  //debugShowCheckedModeBanner: false,
)); // MaterialApp
```

Figure 2: main class code snippet

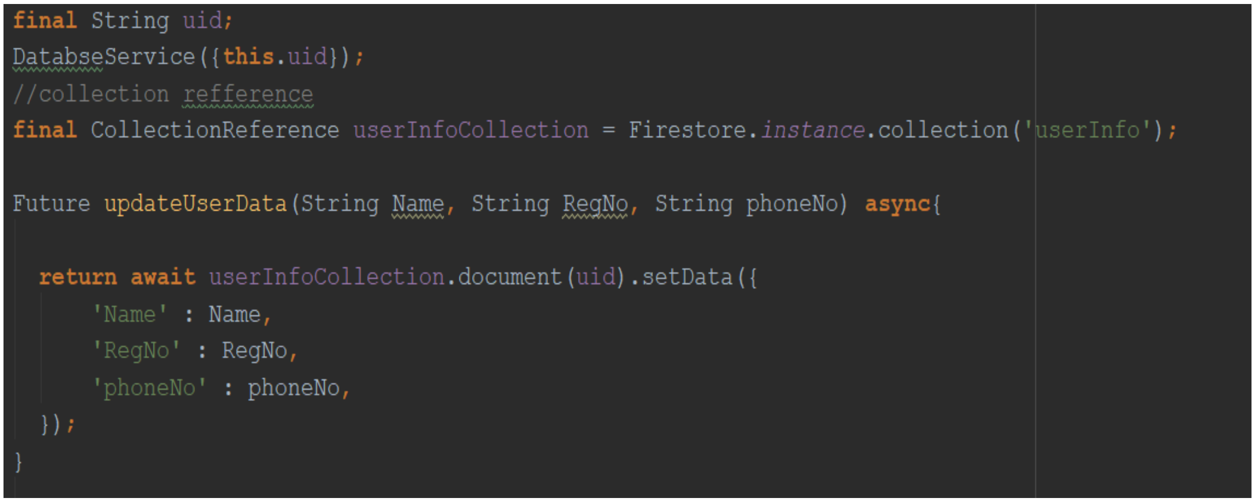
3.2 Database Service Class

In this class we mainly fetch data from users and store it in the firebase database.



```
dynamic data;
final DocumentReference document = Firestore.instance.collection("userInfo")
    .document(User.uid);

Future getUSerData() async {
  //print("i am here & here is the problem to fetch data from database");
  print(uid);
  try{
    await document.get().then<dynamic>((DocumentSnapshot snapshot) async {
      data = snapshot.data;
    });
  }catch(e){
    print(e);
  }
  User.userName =  data['Name'];
  User.userPhoneNo = data['phoneNo'];
  User.userRegNo = data['RegNo'];
  print("name: " + User.userName + "\nemail: " + User.userEmail + "\npassword: " + User.password + "\nphone No: "
      User.userPhoneNo + "\nRegistration No: " + User.userRegNo);
  //print(data.toString());
}
```

Figure 3: fetch data and store in the database



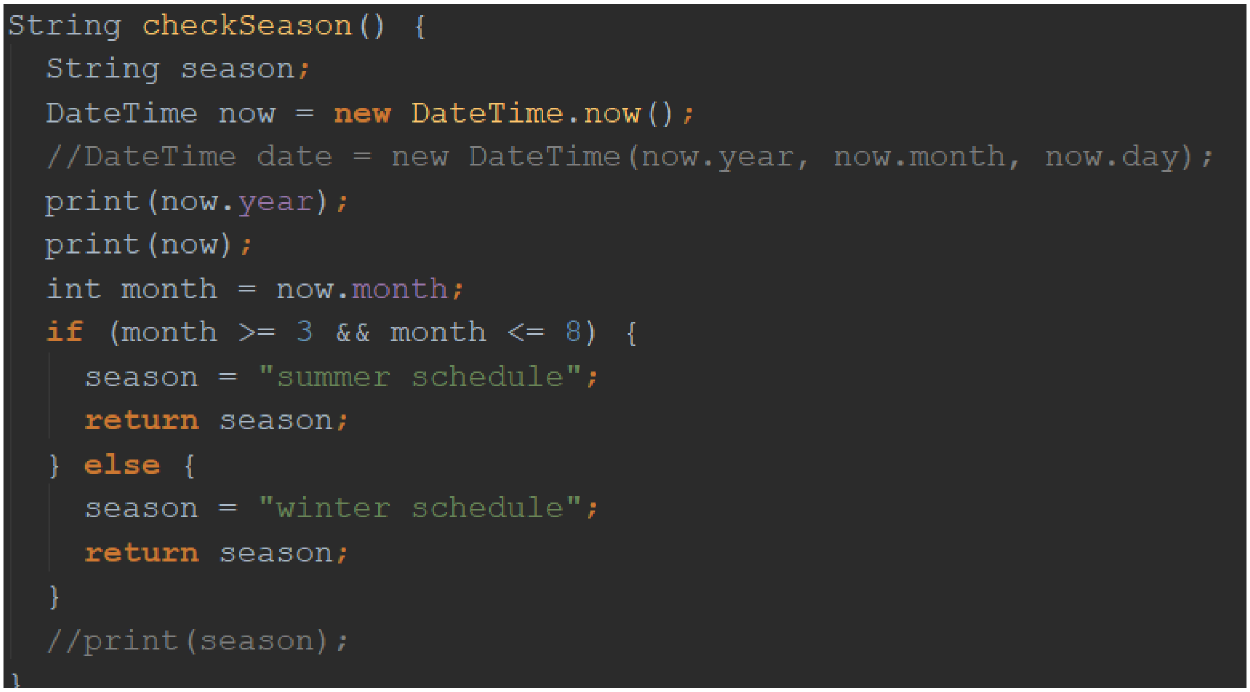
```
final String uid;
DatabseService({this.uid});
//collection refference
final CollectionReference userInfoCollection = Firestore.instance.collection('userInfo');

Future updateUserData(String Name, String RegNo, String phoneNo) async{

  return await userInfoCollection.document(uid).setData({
      'Name' : Name,
      'RegNo' : RegNo,
      'phoneNo' : phoneNo,
  });
}
```

Figure 4: Updating user data in database

Here, we update user data in the database.

3.3 Schedule Class (Dynamic Scheduling)

In the schedule class, if it is March to August month, the system will show the summer schedule. Otherwise it will show the winter schedule.

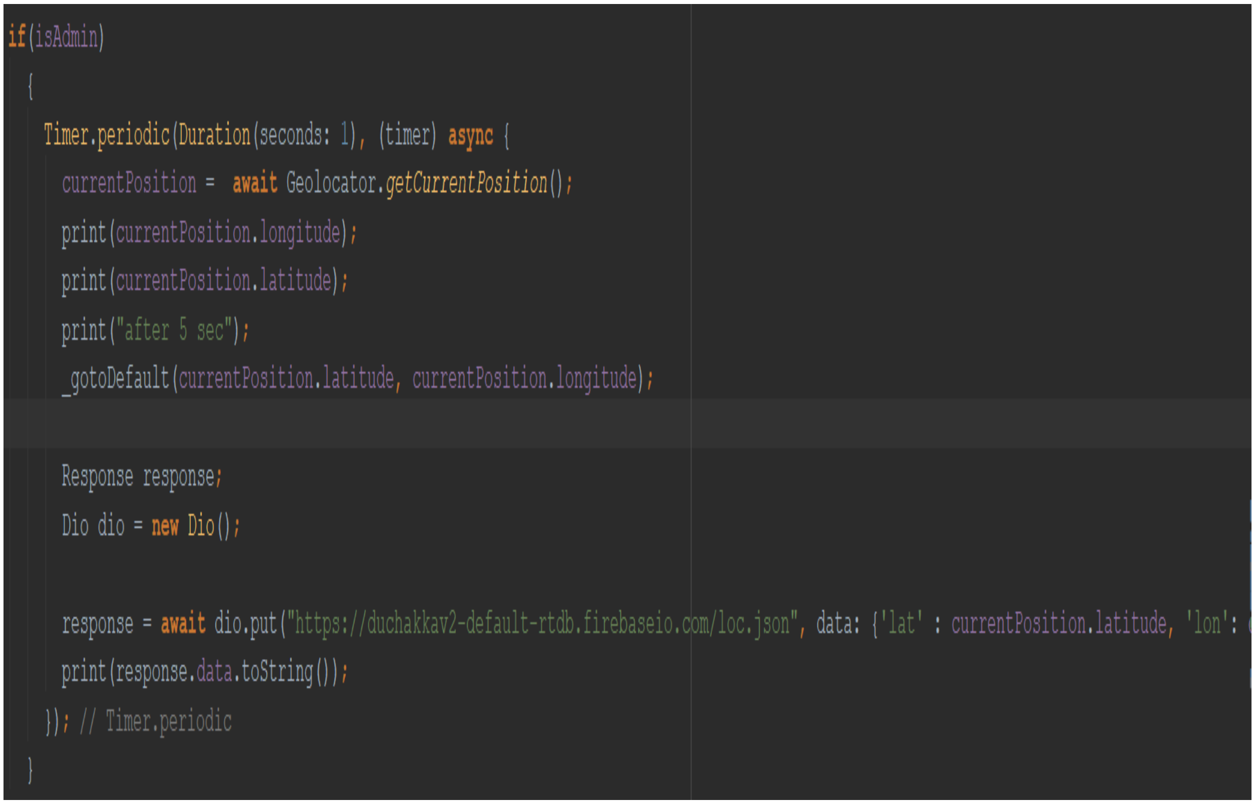


```
String checkSeason() {
  String season;
  DateTime now = new DateTime.now();
  //DateTime date = new DateTime(now.year, now.month, now.day);
  print(now.year);
  print(now);
  int month = now.month;
  if (month >= 3 && month <= 8) {
    season = "summer schedule";
    return season;
  } else {
    season = "winter schedule";
    return season;
  }
  //print(season);
}
```

Figure 5: dynamic scheduling

3.4 Locate Admin

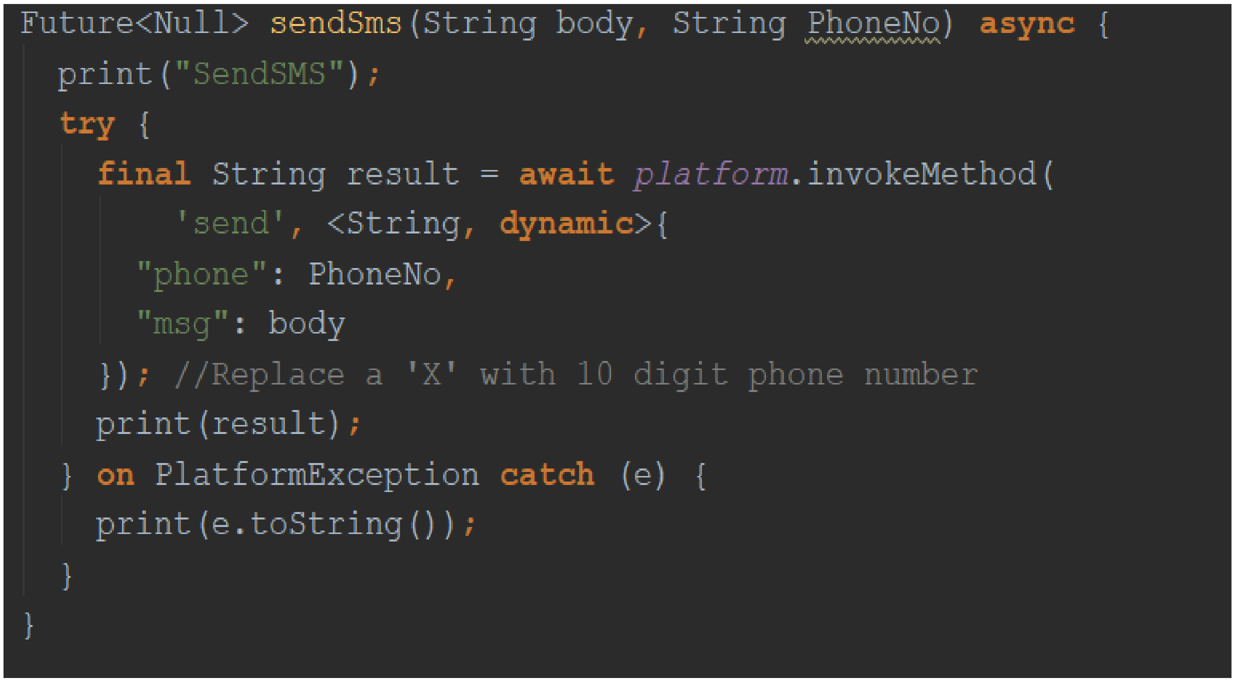
To show the current location of the bus, we fetch the longitude and latitude of the admin and store it in the database.



```
if(isAdmin)
 {
   Timer.periodic(Duration(seconds: 1), (timer) async {
     currentPosition =  await Geolocator.getCurrentPosition();
     print(currentPosition.longitude);
     print(currentPosition.latitude);
     print("after 5 sec");
     _gotoDefault(currentPosition.latitude, currentPosition.longitude);

     Response response;
     Dio dio = new Dio();

     response = await dio.put("https://duchakkav2-default-rtdb.firebaseio.com/loc.json", data: {'lat' : currentPosition.latitude, 'lon':
     print(response.data.toString());
   }); // Timer.periodic
 }
```

Figure 6: locating admin

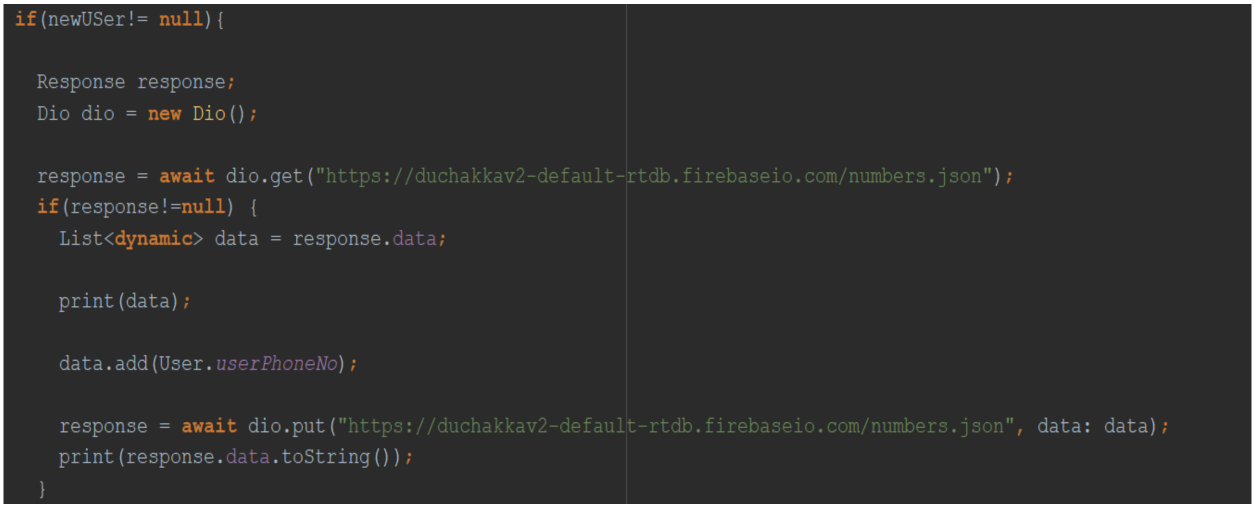
3.5 Send notification through SMS in notification class



```
Future<Null> sendSms(String body, String PhoneNo) async {
  print("SendSMS");
  try {
    final String result = await platform.invokeMethod(
        'send', <String, dynamic>{
      "phone": PhoneNo,
      "msg": body
    }); //Replace a 'X' with 10 digit phone number
    print(result);
  } on PlatformException catch (e) {
    print(e.toString());
  }
}
```

Figure 6: SMS notification

3.6 Store Phone Number for future Use

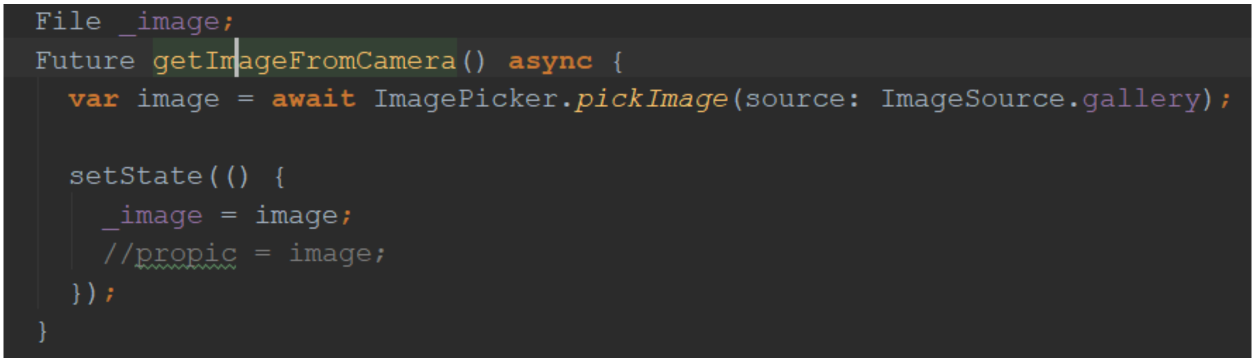


```
if(newUSer!= null){

  Response response;
  Dio dio = new Dio();

  response = await dio.get("https://duchakkav2-default-rtdb.firebaseio.com/numbers.json");
  if(response!=null) {
    List<dynamic> data = response.data;

    print(data);

    data.add(User.userPhoneNo);

    response = await dio.put("https://duchakkav2-default-rtdb.firebaseio.com/numbers.json", data: data);
    print(response.data.toString());
  }
```

Figure 7: store Phone Number

3.7 Get Image from Camera

This code snippet shows how to collect pic from pic and add it as profile information.



```
File _image;
Future getImageFromCamera() async {
  var image = await ImagePicker.pickImage(source: ImageSource.gallery);

  setState(() {
    _image = image;
    //propic = image;
  });
}
```

Figure 8: figure get image

CHAPTER 4: USER MANUAL

4.1 Registration

After clicking on our app icon, the user or admin will get into the welcome screen. If he/she is not registered yet, he has to click the Signup button. Then he/she will get the 2nd image. By entering those credentials (Name, Email, password, phone Number, Registration Number) a user or an admin can register into our system. All information will be stored in the firebase database.

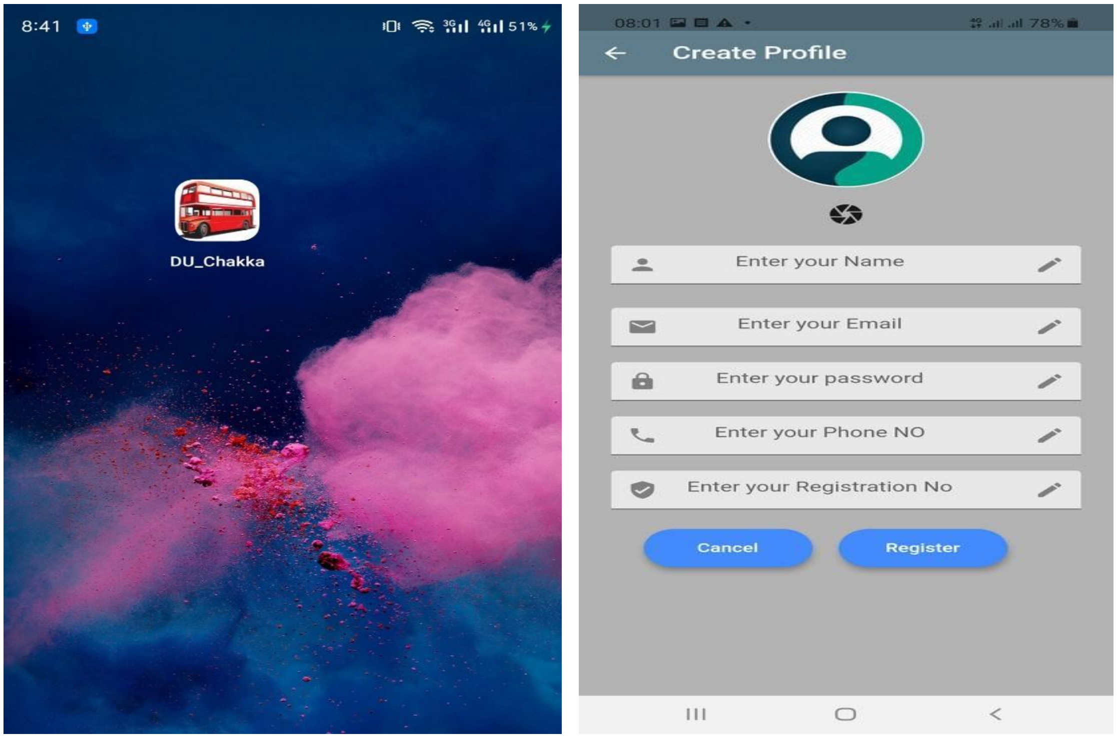
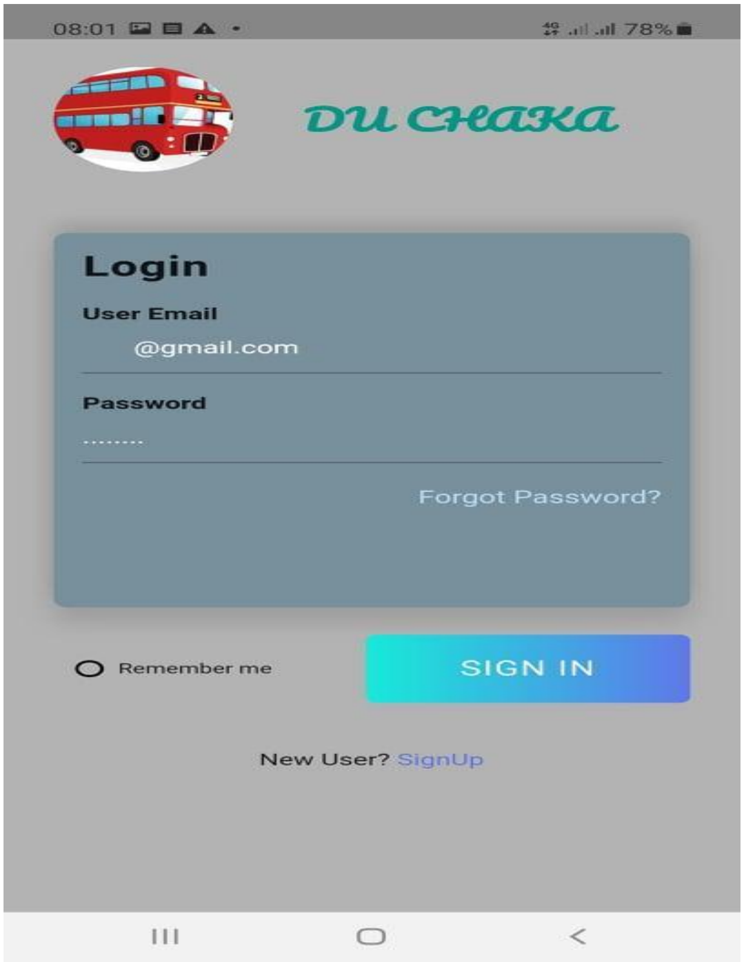


Figure 9: manual for registration

4.2 Login

After registration, the user or admin must login into the system. He/she has to provide his email and password to get into the system.

After successful login, the page will route to the welcome screen.



Login

User Email

@gmail.com

Password

....

Figure 10: manual for login

4.3 Home Screen and Bus Location

The user can see the current bus location in two ways. After successful login, he/she will see the home screen with a blue background. If he/she taps on the screen, the system will show the location of the nearest bus.
If he/she clicks on location on the navigation drawer, the page will route to the location section.

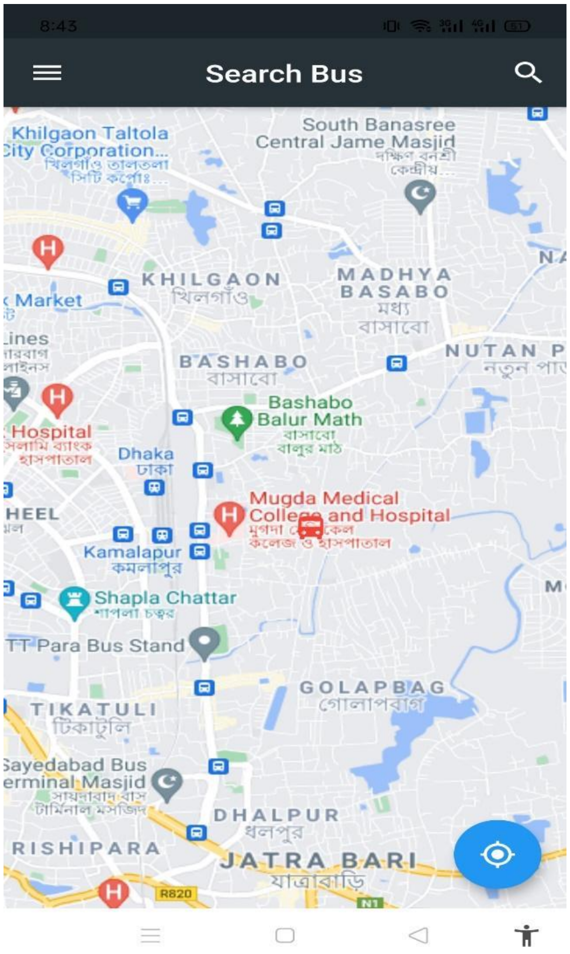


Figure 11: manual for bus location

4.4 Schedule

If the user clicks on schedule on the navigation drawer, the page will route to the schedule section. On the schedule section, there are the names of the buses. If the user click on the name of the bus, the system will show him the schedule of the bus.

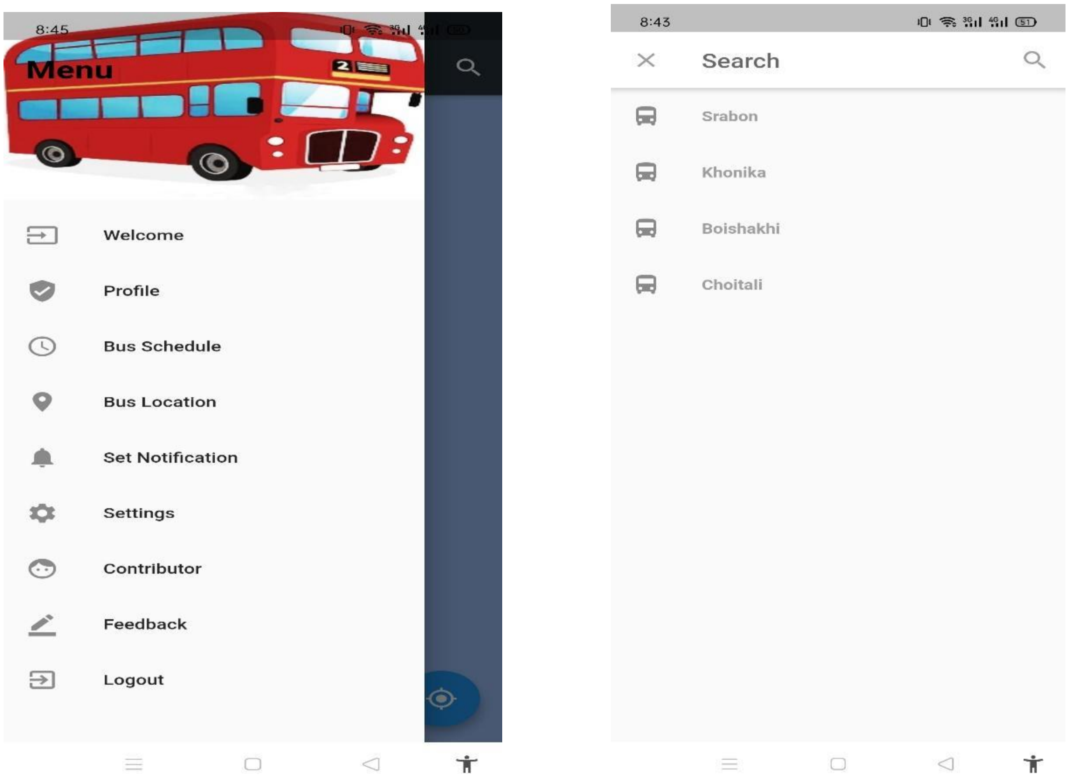
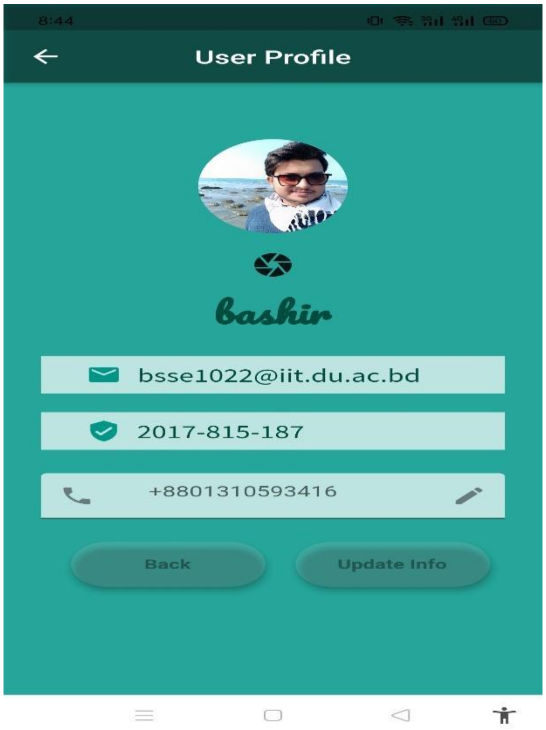


Figure 12: manual for bus search for schedule

4.5 Profile

After registration a profile is created by the system. After clicking on profile on the navigation drawer, the page will route to the profile section. The user can see his profile and he/she can change his phone number.

To change his/her phone number he/she has to click on update info, then he/she can change the phone number.



User Profile

bashir

bsse1022@iit.du.ac.bd

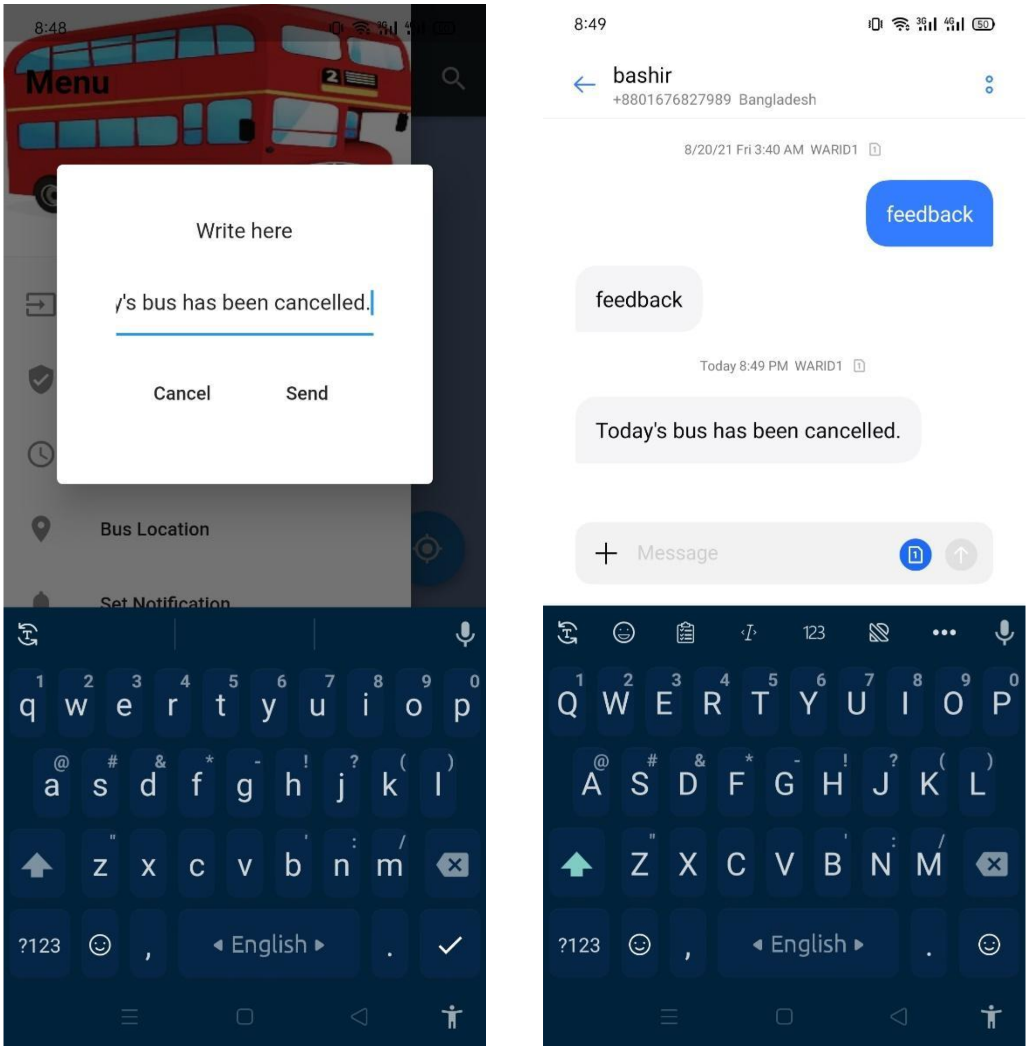
2017-815-187

+8801310593416

Figure 13: manual for profile

4.6 Notification

After clicking on notification on the navigation drawer, the page will route to the notification section. If the user is a normal user, this click will not work because normal users cannot create a notification. If the user is an admin, a dialogue box will pop. Then the admin can create a post and the system will send SMS to all users.



Write here

/'s bus has been cancelled.

bashir
+8801676827989 Bangladesh

8/20/21 Fri 3:40 AM WARID1

feedback

feedback

Today 8:49 PM WARID1

Today's bus has been cancelled.

Figure 14: manual for notification

4.7 Feedback

After clicking on feedback on the navigation drawer, the page will route to the feedback section. A dialogue box will pop. Both user and admin can give feedback.

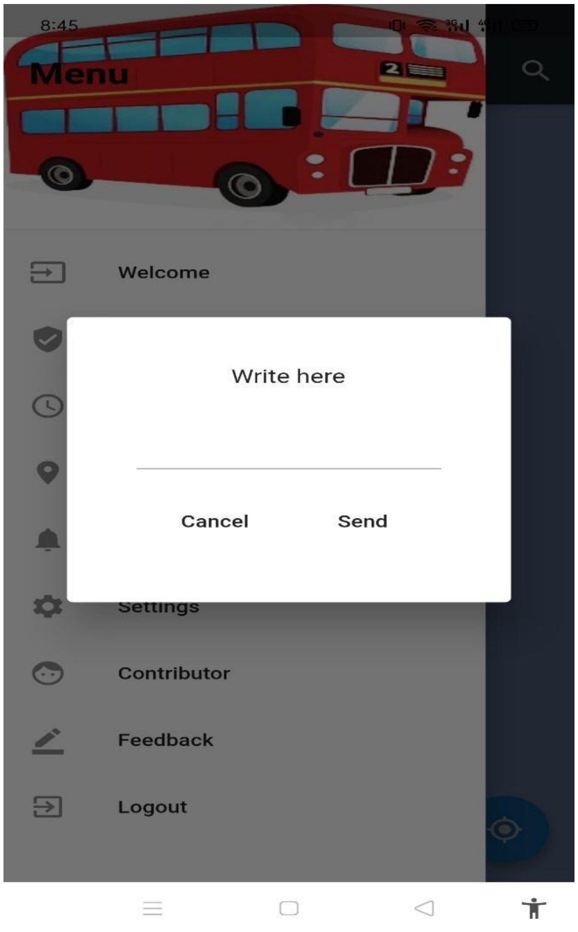
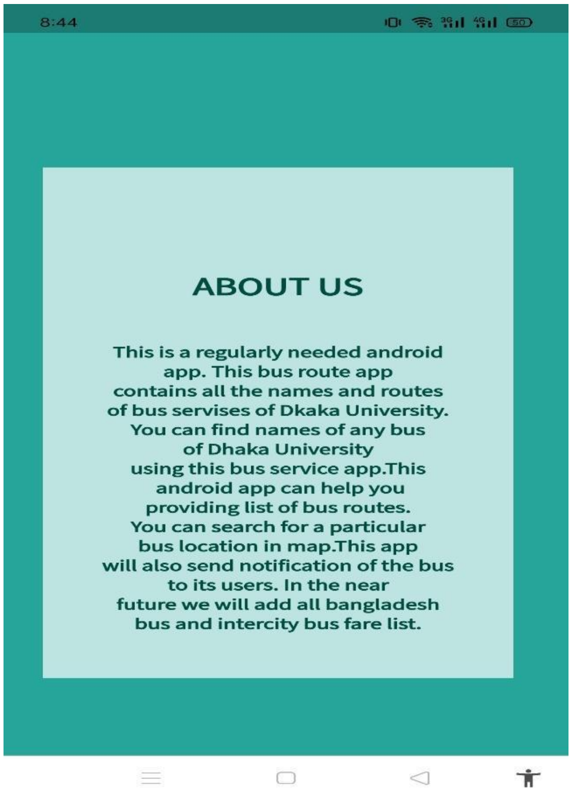


Figure 15: manual for feedback

4.8 About

After clicking on about on the navigation drawer, the page will route to about section.



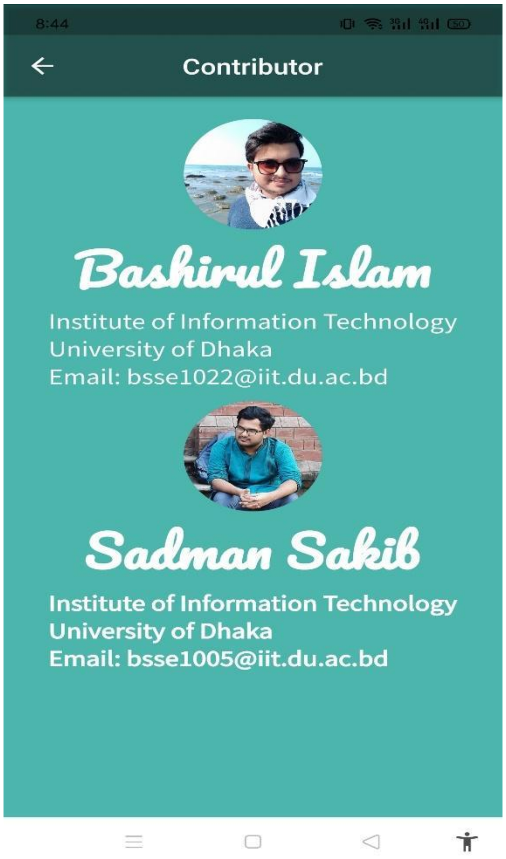
ABOUT US

This is a regularly needed android app. This bus route app contains all the names and routes of bus servises of Dkaka University. You can find names of any bus of Dhaka University using this bus service app.This android app can help you providing list of bus routes. You can search for a particular bus location in map.This app will also send notification of the bus to its users. In the near future we will add all bangladesh bus and intercity bus fare list.

Figure 16: manual for about section

4.9 Contributor

After clicking on contributor on the navigation drawer, the page will route to the contributor section.



Bashirul Islam

Institute of Information Technology
University of Dhaka
Email: bsse1022@iit.du.ac.bd

Sadman Sakib

Institute of Information Technology
University of Dhaka
Email: bsse1005@iit.du.ac.bd

Figure 17: manual for contributor section

4.10 Log out

When the user clicks on the log out button, he/she will log out of the system.

.

5.1 Conclusion

Implementing this helps to understand this new topic and how to use flutter in different scenarios. Now we have gathered knowledge about sending notification and gathering notification. The project also helps us to improve our coding skill and I have learned to handle large code. This project was quiet challenging and we gained a lot of experience from it. We want to thank my supervisor and other respectable teachers for guiding me a lot during this project.

5.2 Appendix

- the app does not show location of the bus without internet connection
- admin to admin communication is not possible

6. Reference

1. https://tariqul09tmgmail.wordpress.com/2017/04/25/khonika-a-story-of-a-dhaka-university-bus-route

   (bus schedule, date: 20/ 01/2019)

2 https://ieeexplore.ieee.org/abstract/document/6950068

(smart bus service, date: 2/ 02 /2018)