SMART MESS MANAGER

BY

MD SHAH PARAN
ID: 161-15-7020

MD SHAIDUL HAQUE TIPU
ID: 161-15-6906

AND
ZUBAYIR AHMED
ID: 161-15-6907

This Report Presented in Partial Fulfillment of the Requirements for the Degree of Bachelor of Science in Computer Science and Engineering

Supervised By

DR. SHEAK RASHED HAIDER NOORI
Associate Professor and Associate Head
Department of CSE
Daffodil International University

DAFFODIL INTERNATIONAL UNIVERSITY

DHAKA, BANGLADESH

SEPTEMBER 2019

APPROVAL

This Project titled **"SMART MEAL MANAGEMENT"**, Submitted by **MD SHAH PARAN, ID: 161-15-7020, MD SHAIDUL HAQUE TIPU, ID: 161-15-6906, ZUBAYIR AHMED, ID:161-15-6907** to the Department of Computer Science and Engineering, Daffodil International University has been accepted as satisfactory for the partial fulfillment of the requirements for the degree of B.Sc. in Computer Science and Engineering and approved as to its style and contents. The presentation has been held on 14th September 2019.

BOARD OF EXAMINERS

Dr. Syed Akhter Hossain — **Chairman**
Professor and Head
Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

Md. Tarek Habib — **Internal Examiner**
Assistant Professor
Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

Abdus Sattar — **Internal Examiner**
Assistant Professor
Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

Dr. Dewan Md. Farid — **External Examiner**
Associate Professor
Department of Computer Science and Engineering
United International University

DECLARATION

We hereby declare that, this project has been done by us under the supervision of **Dr. Sheak Rashed Haider Noori, Associate Professor and Associate Head** Daffodil International University. We also declare that neither this project nor any part of this project has been submitted elsewhere for award of any degree or diploma.

Supervised by:

Dr. Sheak Rashed Haider Noori
Associate Professor and Associate Head
Department of CSE
Daffodil International University

Submitted by:

Shah Paran

(Md Shah Paran)
ID: 161-15-7020
Department of CSE
Daffodil International University

(Md Shaidul Haque Tipu)
ID: 161-15-6906
Department of CSE
Daffodil International University

(Zubayir Ahmed)
ID: 161-15-6907
Department of CSE
Daffodil International University

ACKNOWLEDGEMENT

First we express our heartiest thanks and gratefulness to almighty God for His divine blessing makes us possible to complete the final year project/internship successfully.

We really grateful and wish our profound our indebtedness to **Dr. Sheak Rashed Haider Noori, Associate Professor and Associate Head**, Department of CSE Daffodil International University, Dhaka. Deep Knowledge & keen interest of our supervisor in the field of "web and apps" to carry out this project. His endless patience ,scholarly guidance ,continual encouragement , constant and energetic supervision, constructive criticism , valuable advice ,reading many inferior draft and correcting them at all stage have made it possible to complete this project.

We would like to express our heartiest gratitude Dr. Sheak Rashed Haider Noori, Associate Professor and Associate Head**,** Department of CSE, for his kind help to finish our project and also to other faculty member and the staff of CSE department of Daffodil International University.

We would like to thank our entire course mate in Daffodil International University, who took part in this discuss while completing the course work.

Finally, we must acknowledge with due respect the constant support and patients of our parents.

ABSTRACT

This report is about a meal management system. Here people of a mess or a sublet will be able to control their various type of calculation easily. In Dhaka there have various type of mess where students, jobholder, male and female included. We are creating the apps thinking about the bachelor. Day by day these type of people are increasing. So we are creating our apps think about the situation. In meal management system apps will make bachelor life easier. Here one person can be admin in a mess. And the admin will invite others person. At the same time person can be added by himself as a member. By this apps

People of a mess can calculate food cost, members, who is giving money in which day, his of previous month. By this app people of mess can calculate their all type of food related problems. Which day which people would gone to market, all can be seen here. This apps will make our life more comfortable and trustable. In our growing development sector, especially for young middleclass people this apps will works a lot.

TABLE OF CONTENTS

LIST OF FIGURES

CHAPTER 1

Introduction

1.1 Introduction

Meal management system is an apps based service where people will get all type of meal management process in one apps. Here mess member can calculate all type food related things.by this app mess member can find out who will go to market or who have to give money. People can find out mess cost from this apps. It will help to solve the calculation problem of mess members.

1.2 Motivation

Bangladesh is a developing country. Everyday many students and various type of person come to Dhaka and city area for many reason. Almost people stay in mess or sublet. They have to go to market sharing their time and money. Sometime we feel difficulties to find out who go for marketing, who give money for marketing and what we are eating like that. Our team also felt the calculation problem, find out proper meal rate instantly on current month and expense change problem. As we felt these problem we wanted to solve these for us at 1st stage. So our practical problem make us motivated for the apps.

1.3 Objectives

1. We will create safe and comfortable life for mess livers.
2. People from several place can be able to know about mess and its cost.
3. Giving a proper calculation about mess daily life.
4. To make mess life easier, friendly and well organized.

1.4 Expected Outcome

1. We want to run this project in physical market.
2. In 1st year we want to work in country level to spread it in full Bangladesh.
3. Within 1 year we want to make minimum 50,000 users.

1.5 Report Layout

Create our report with all the details and suitable information. We have completed our report as follows:

Chapter 1: In this chapter, we discuss about motivation, objectives and expected outcomes of our project.

Chapter 2: Here we talk about introduction, related works, comparative studies, scope of the prblems and challenges .From this chapter we discuss about background of our project.

Chapter 3: We composed here about business process modeling, requirement collection and analysis. Here we have shown use case modeling and description, logical data model and design requirement of our project.

Chapter 4: Here we give off font end design, back end design and implementation requirement of our project.

Chapter 5: We specified database implementation, font end design implementation, implementation of interaction and also testing implementation of our project.

Chapter 6: We conversed about discussion and conclusion. We also provided our scope for future development our project.

CHAPTER 2

Background

2.1 Introduction

Before started our project we completed many research in field level. We talk to many type of people for this project. They give a positive review. After completing our research on field level we started our work. At the same time we went to many mess for research.

2.2 Related Works

In the present time there have some similar work in the market. Few of them are working well like meal manager, mess manager, hostel meal manager, meal manager, mess expense, mess app etc. Few of them are doing well. But these are simple and not well organized. We see that their apps are look like poor UI design and also their user experience not good. That's why we try to provide a better user experience and also UI design from them.

Meal Manager

Meal manager presented by rayhandroid. This apps calculate about meal, other expense, deposit money and messaging system [1]. Anybody can create account via email address. It's basically lunched in Bangladesh targeting on the middle class mess livers.

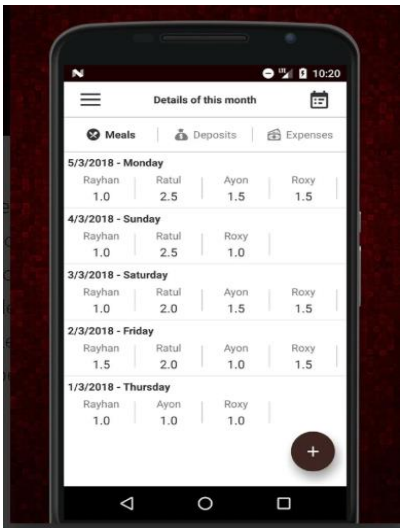


Figure 2.1: Meal Manager

Mess Manager

This app will automatically show every member's cost, about meal number, rate of meal and how much each member should pay more or will get back [2].

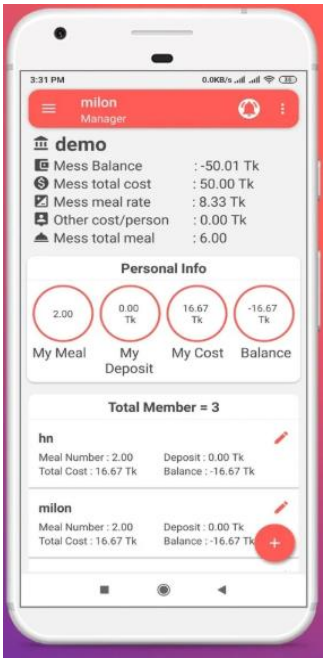


Figure 2.2: Mess Manager

Hostel Meal Manager

This is very simple apps. Anybody can join to the apps and can entry name, deposit, mill. And it will be added in database [3].This apps look like very unprofessional UI design and also poor user experience.

Figure 2.3: Hostel Meal Manager

There are few more apps like these but they are below standard like mealist, messapp. They provide simple entry to the database.

2.3 Comparative Studies

Most of the meal management system apps are very simple .They calculate about the total meal, total expense etc. Comparison among these apps become studied well. Even we see that some apps are provide bad user experience and poor UI design also. But in our apps provide user friendly UI design. We have unique features which is not available in any other existent apps such as phone number authentication, graphical view of user cost and deposit, automatic bazar schedule.

2.4 Scope of the Problem

Building such type of apps we need to do many field work. At the same time we need to invest for marketing. At the 1st stage investment is huge problem for us. Proper information collection is another problem which we face. We have to ensure have people will use it, know it properly, But in 1st stage its really difficult for to ensure the use of the apps.

2.5 Challenges

In this project we have to face problem in several sector. When we work on Admin panel, we have to face a lot. In flatten data structure section [4] we have to face most difficulties. An apps quality and representation depend on UX design and related to others graphics design. So we felt enough challenge on design section. We focused on user friendly design for customer. In notification section we have to face huge problem to control the structure.

CHAPTER 3

Requirement Specification

3.1 Business Process Modeling

This is the main structure of smart meal management where almost everything can be seen here. Customer can make id via email and contact number. When anybody will give request for an account a code will go to the address for confirmation. In this apps we can find out home, activities, schedule, profile etc. In the profile customer or user will see advertise related to meal and various type of offer.

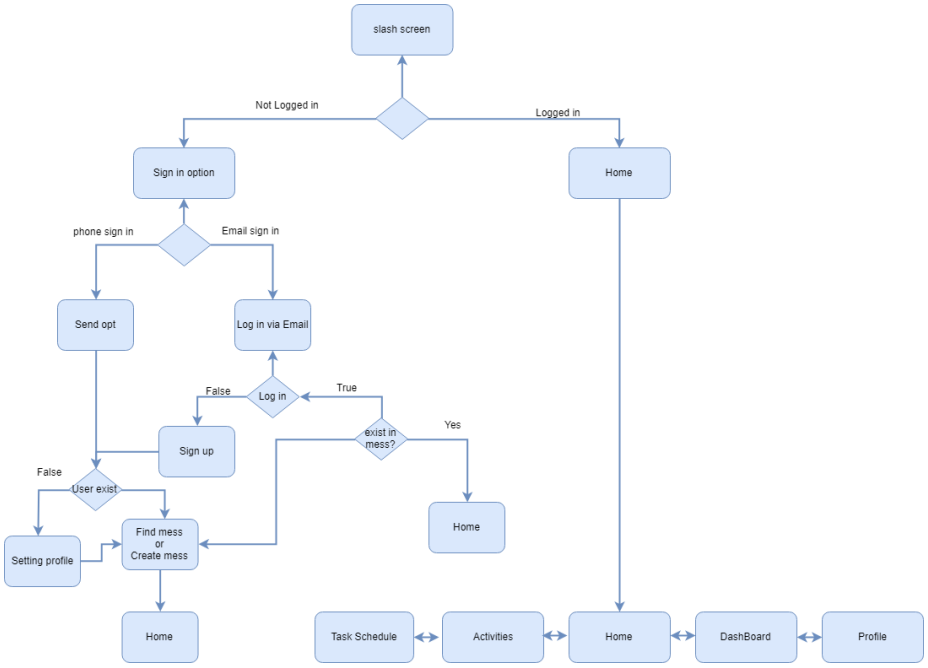


Figure 3.1: Business Process Modeling

3.2 Requirement Collection and Analysis

To build the project we needed android studio software. At the same time we have to use firebase for secure database and server. In language section we have to use java. Moreover we uses all of these in a mobile. After completing our apps we analyze that via customer. Based on customer review we changed few think like notification section and few UX design part.

3.3 Use Case Modeling and Description

This model is used for our app. Here have two type of actor. One is manager and another is general member. Here manager can create an account where he will get all type of access like add member, delete member, update member. A manager can accept member request where general member can't do that. Except that work a member can sign up to the apps and with the excess of manager they can see update of bill, expense, deposit, history etc. So in this apps member have all excess without member adding, deleting, updating, removing member and promoting member to manager section.

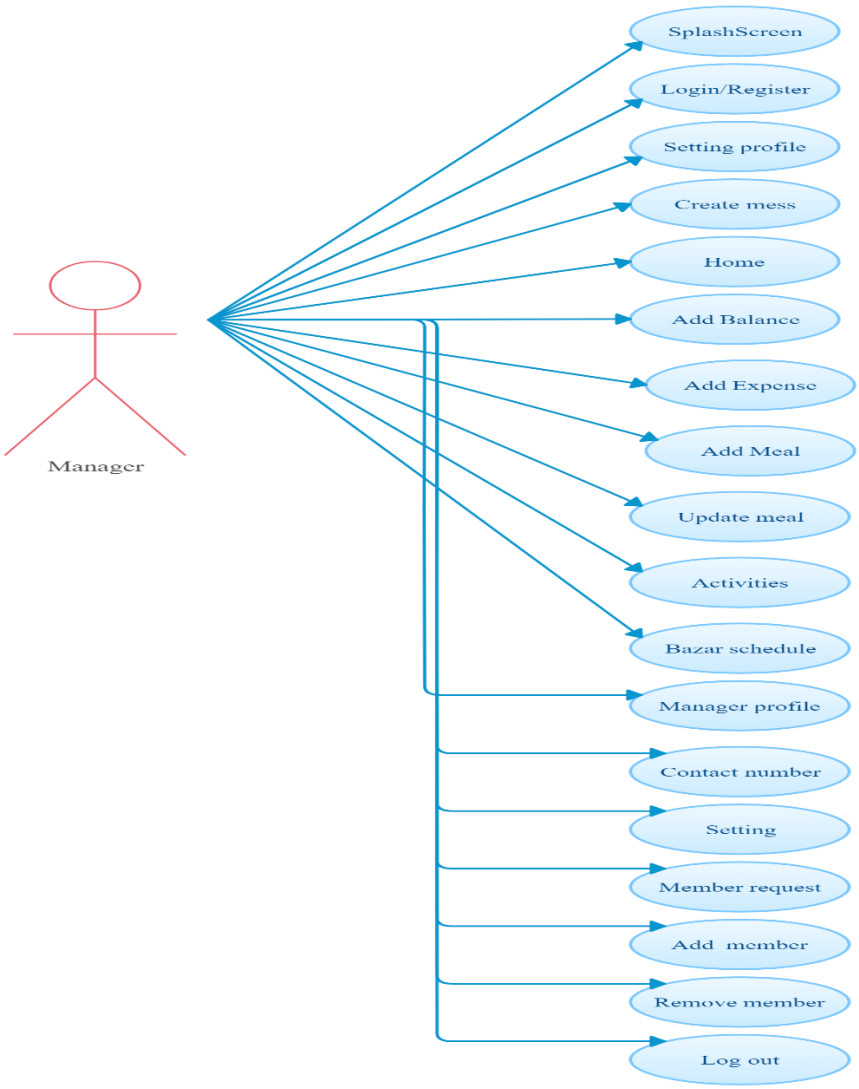


Figure 3.2: Use Case Modeling and Description

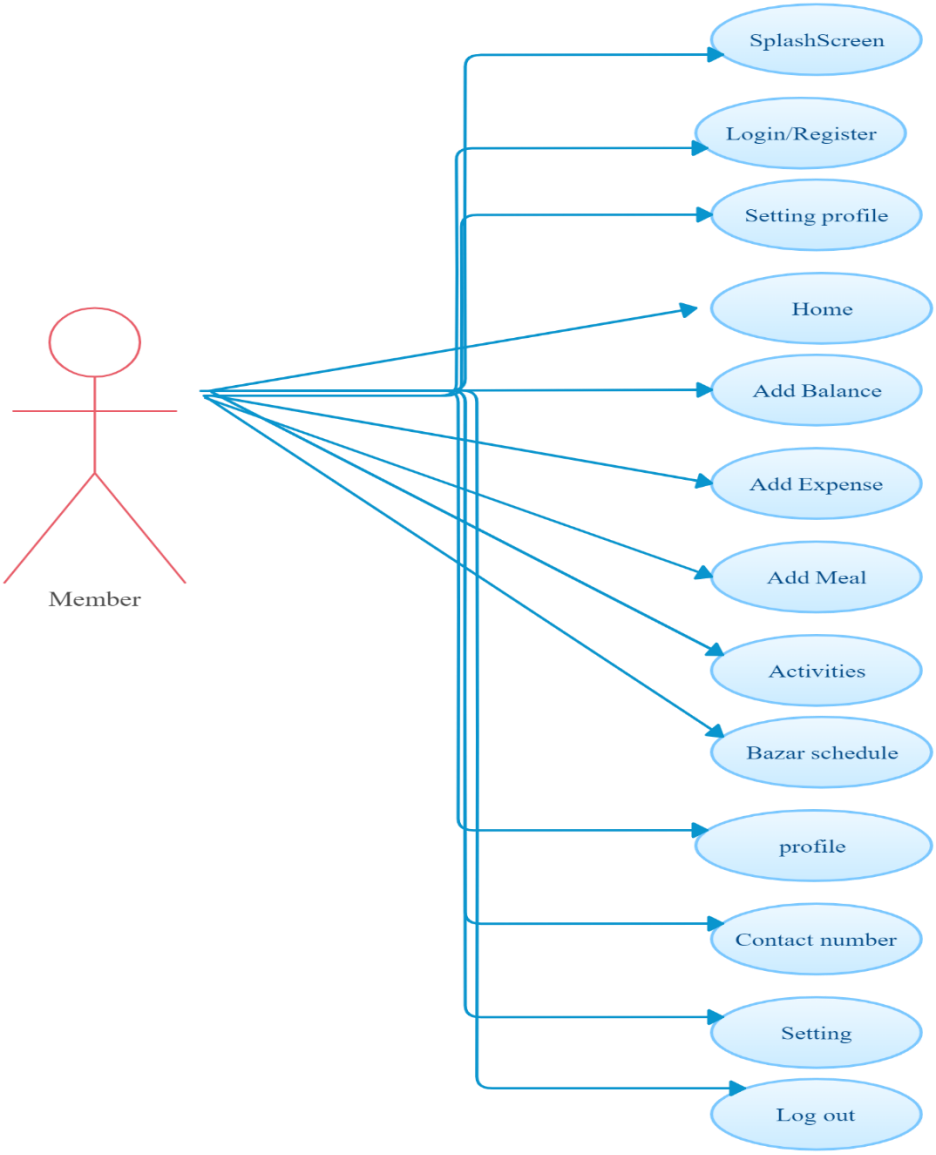


Figure 3.3: Use Case Modeling and Description

3.4 Logical Data Model

In logical data model part all data will be connected to the database. Data will transfer from database to android version. Apps will take all data from database. Apps security system is good enough for customer.

3.5 Design Requirements

To complete the design we required user interface design skill. We need proper database design and modeling. We need to take proper step in design sector. Design should be user friendly. We use XML to design the user interface. Adobe XD also used to design logo and icon design.

3.6 JSON Data Model

Firebase database is used as Json data model structure. Json data model is essential part of our project . Here key and value become paired to store data. Firebase is a non SQL database which make the apps faster and more secure [5].

Figure 3.4: JSON Data Model

Chapter 4

Design Specification

4.1 Front-end Design

We make a front end design which is easier for customers to handle. Here customer will get various type of taste. We make the apps for all type of customer who will be able to control the apps easily. We use XML for front-end design.

Splash screen

This is the splash screen of smart Meal manager apps. When member will open the apps they will see it.

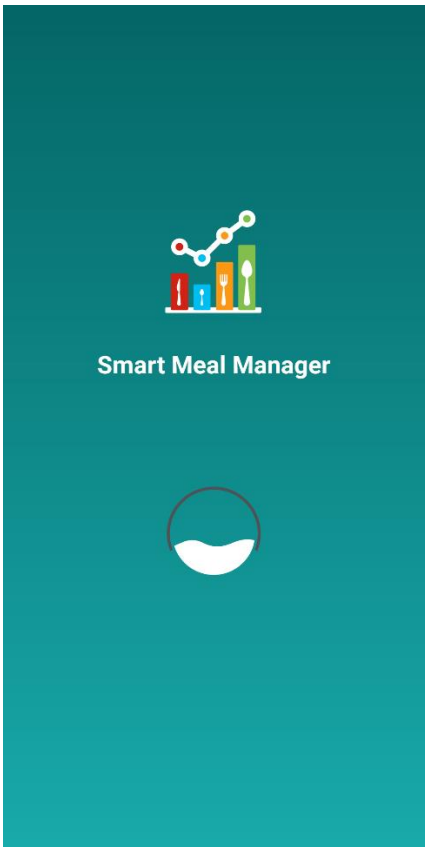


Figure 4.1: Splash screen

Sign In method

User can sign in to smart meal manager app via email address or mobile number. Here we provide two types of sign up system if any user have no email address then he can sign in by phone number or he want to sign in by email then he can do it.

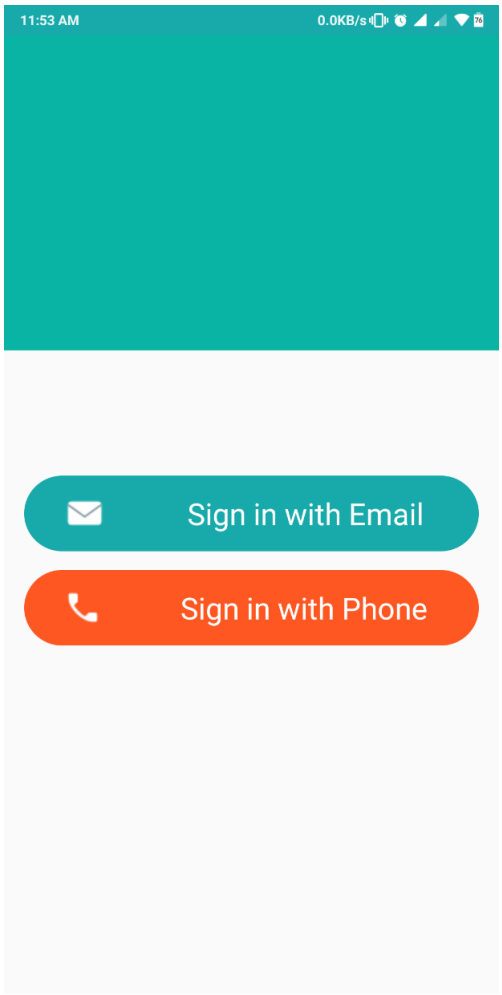


Figure 4.2: Sign In method

Sign up

User can sign up using email address, password and confirm password. If user have already a created account then he go to the sign in activity.

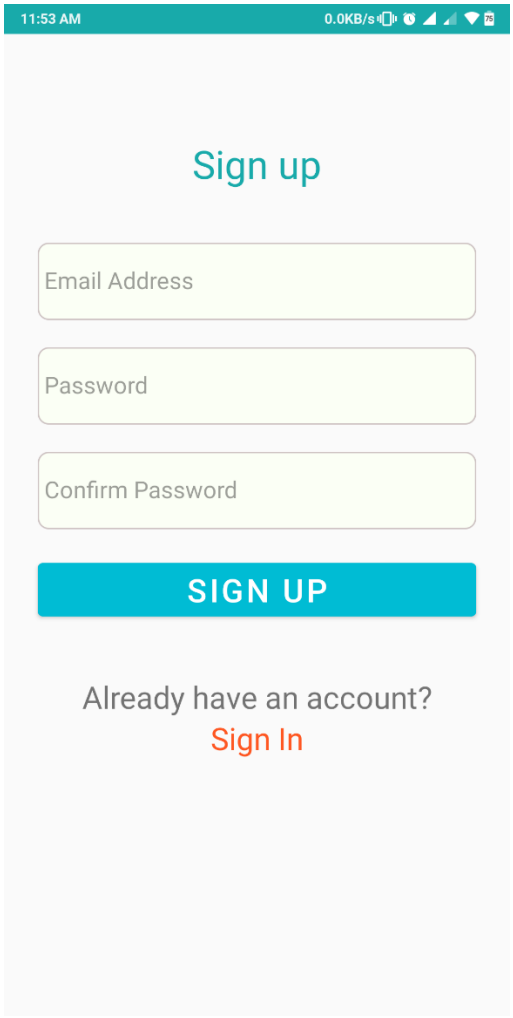


Figure 4.3: Sign up

Log In

User can login to the app using authenticated email address and password. After doing a successful log in then he ready go the next activity.

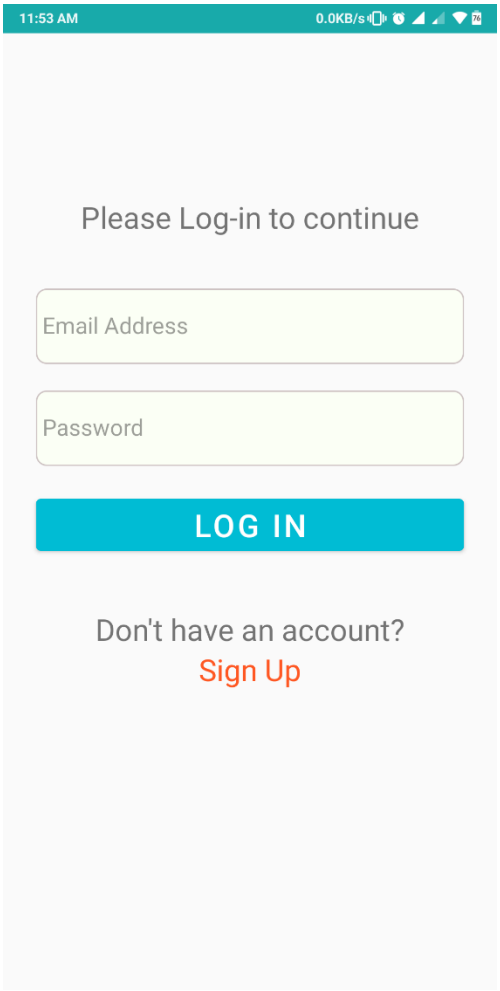


Figure 4.4: Log In

Phone number verification

First of all user give his active phone number and then system will provide a verification code after that it will be verified. By this way app will secure and protect user privacy.

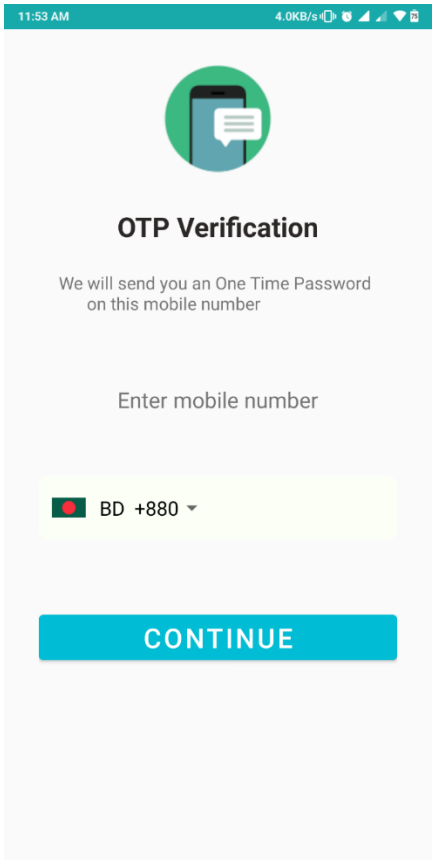


Figure 4.5: Phone number verification

Create/Exist

User can create a new mass or if he want to add other existing mess then he go to the exist mess. When they will go to exit option people will find the search option.

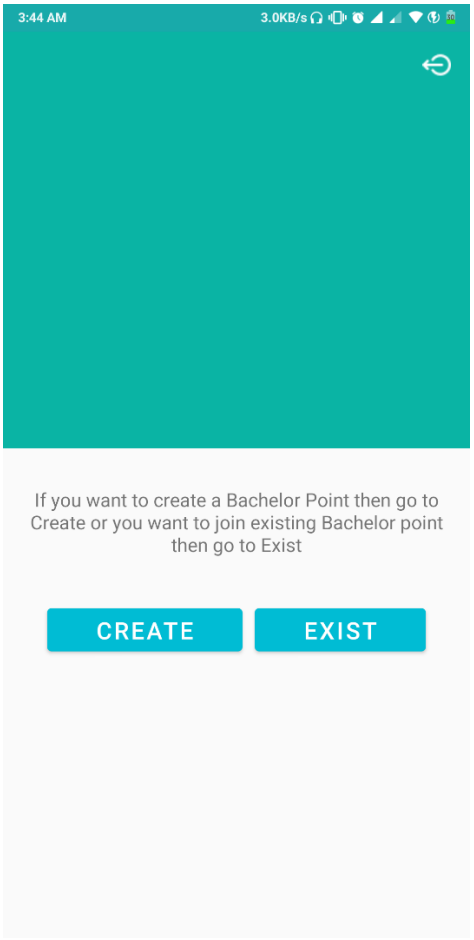


Figure 4.6: Create/Exist

Create mess

User can create a mess as a manager. After creating a successful mess then he can go to the home activity and also he will access everything.

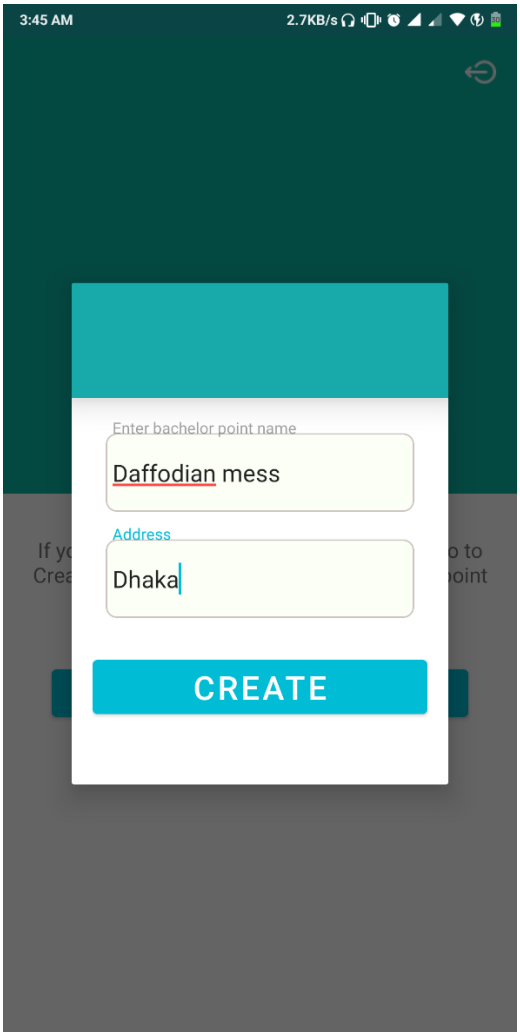


Figure 4.7: Create mess

Search Mess

If user go to the exist option, he will find the search option. Then user can search by mess name and he also can send a join request.

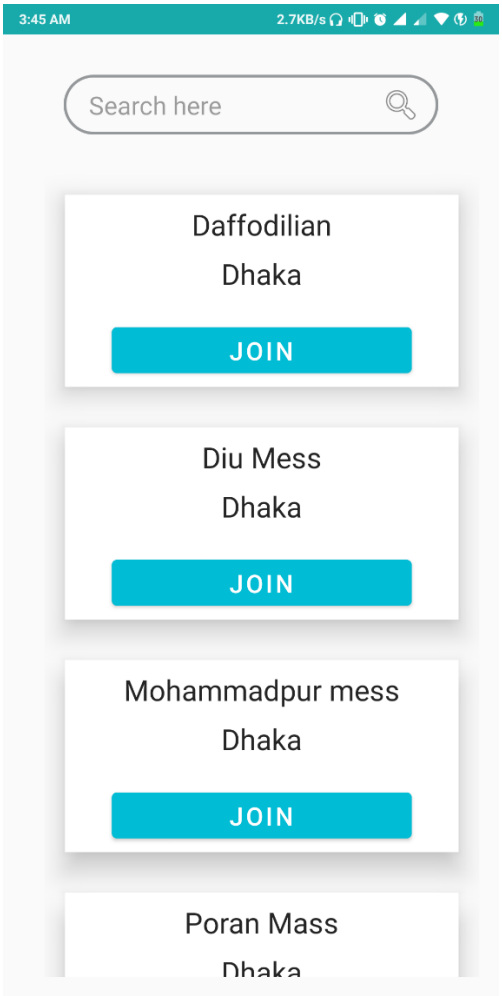


Figure 4.8: Search Mess

Home Activity

Here is the home activity of the apps. Here user can see details about himself and others member. Being joined to mess, member can see mess.

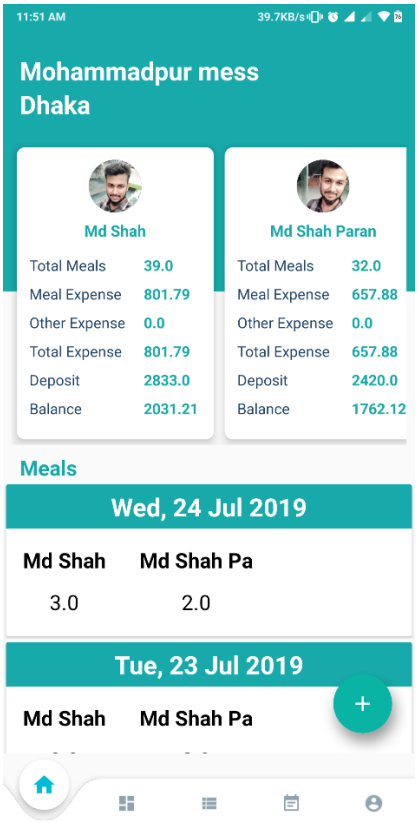


Figure 4.9: Home activity

Add meal

Member can add meal every day. They can add meal once time a day. Even anyone add meal then every member will get notification about it.

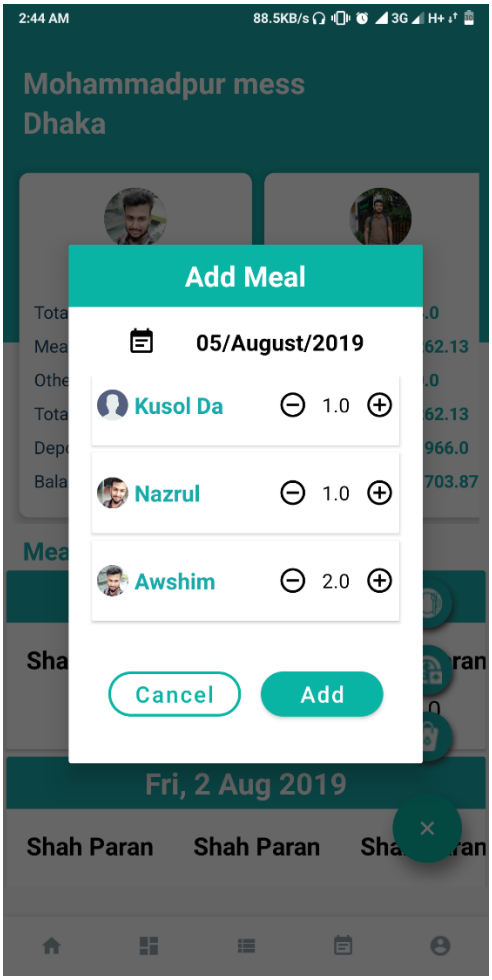


Figure 4.10: Add meal

Add Balance

Every mess member can deposit money anytime for mess expense. This money will add his particular account. Even anyone deposit money then every member will get notification about it.

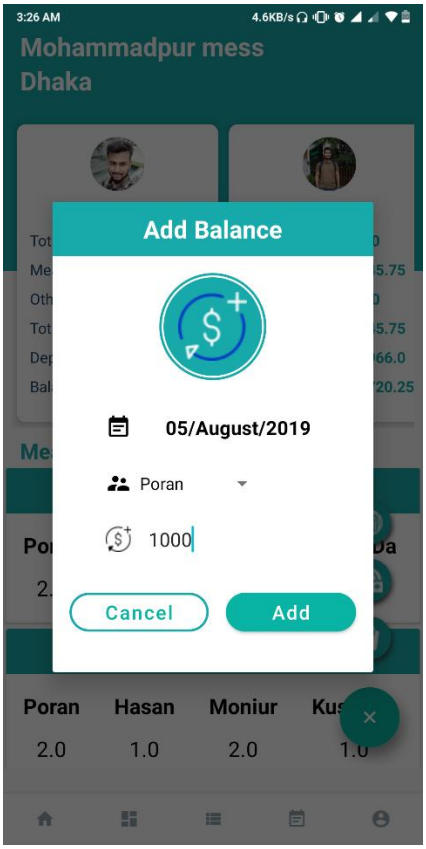


Figure 4.11: Add Balance

Add Expense

Everyday expense will add by add expense such as bazar, utility cost and others. In add expense have two types of system one has regular bazar and another one is utility. Regular bazar can be changed meal rate on the other hand utility are always distributed equally for all the mess member.

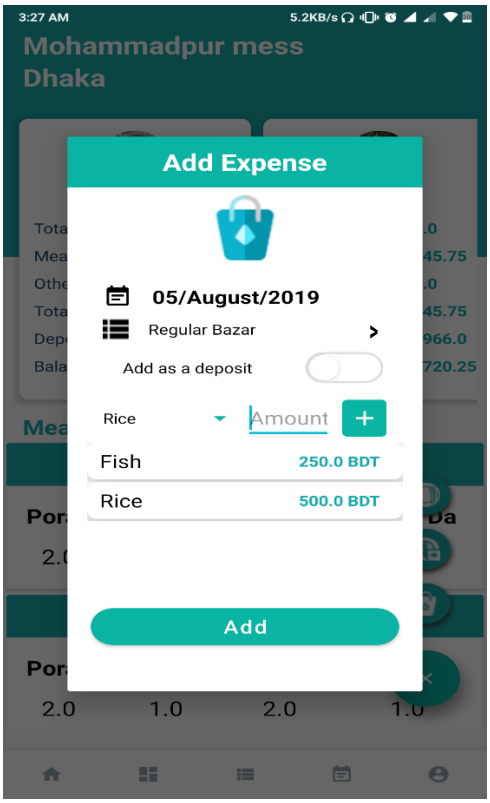


Figure 4.12: Add Balance

Dashboard

In dashboard members can see balance total meal and meal rate with graphical format. Here member and manager will see deposited money and expense of every person. Member can see real time, meal rate, total mess balance and total mess meal.

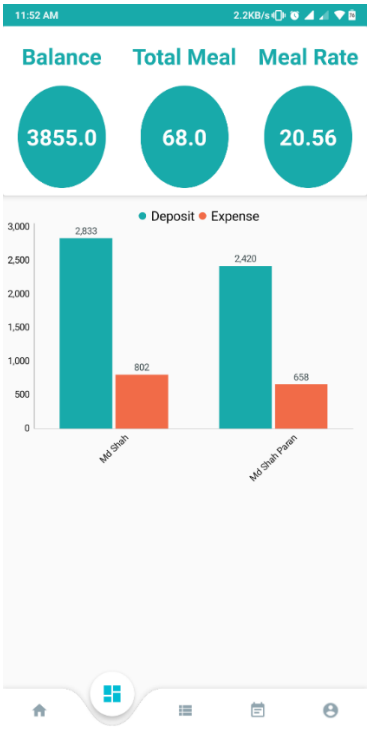


Figure 4.13: Dashboard

Expense

In this apps member can see expense in activities part. This part will represent member present expense and time. For this reason member will be able to find out their expense activities easily.

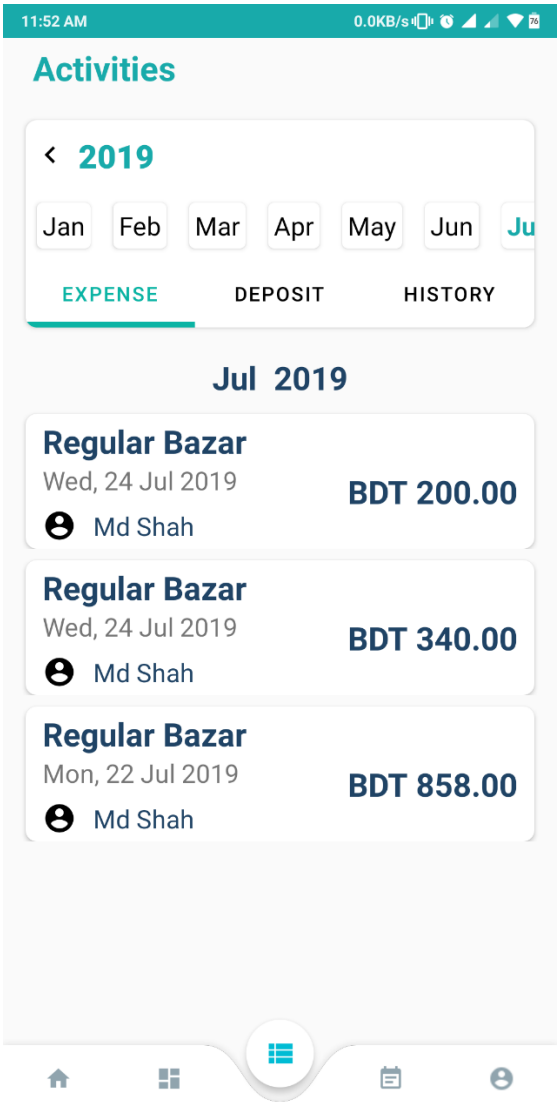


Figure 4.14: Expense

Deposit

In the deposit part member will be able to see how much cost to be deposited to the manager. At the same time how much money already deposited to the manager, member can see that.

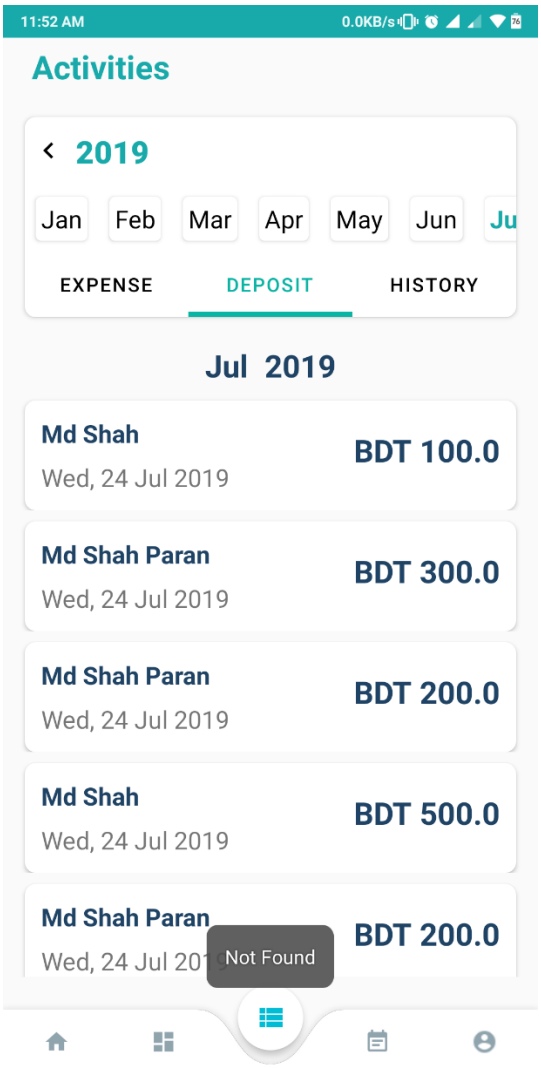


Figure 4.15: Deposit

History

In history part member will be able to see all transaction history of previous month. Here deposit, Bazar cost and meal cost and total cost will be able to see. History part will help members and manager to calculate all things properly.

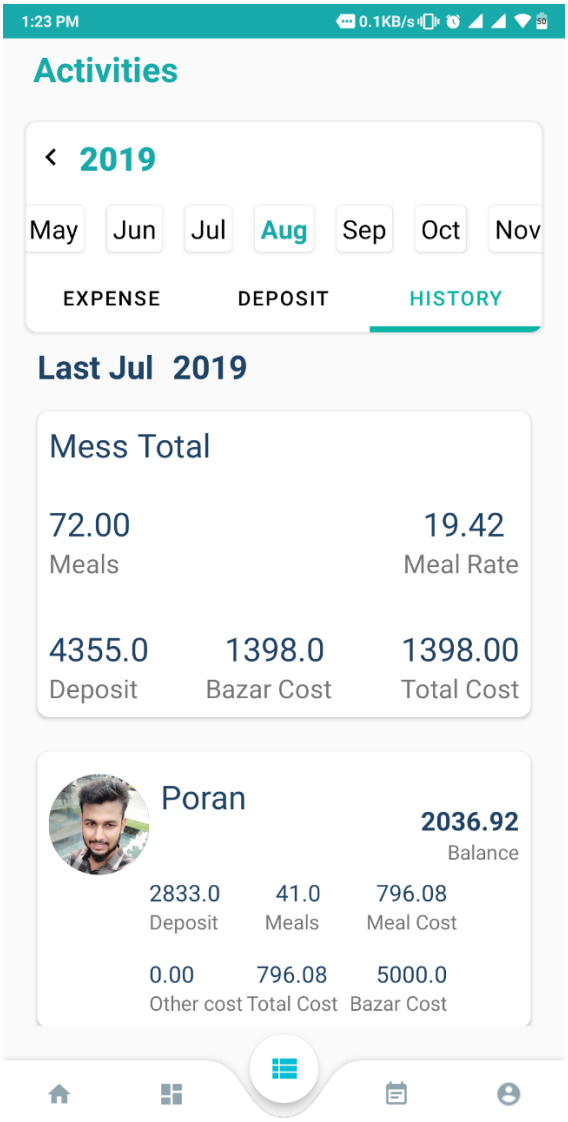


Figure 4.16: History

Bazar schedule

Automatic scheduled bazar based on total member and time to time notify scheduled members. In this apps member can see the market schedule so that member will be able to know about their time management and responsibility. Apps will automatically schedule the market management.

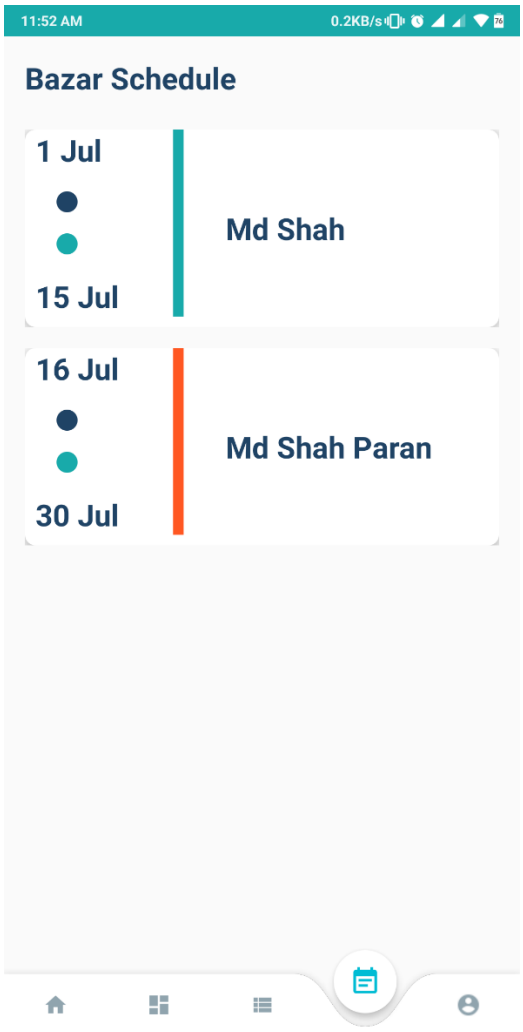


Figure 4.17: Bazar schedule

Manager profile

Manager can add member, accept member request , remove member and other stuff. Manager can control members of a mess. He have the power to add and delete any member from his group. At the same time he can invite members under his management system.

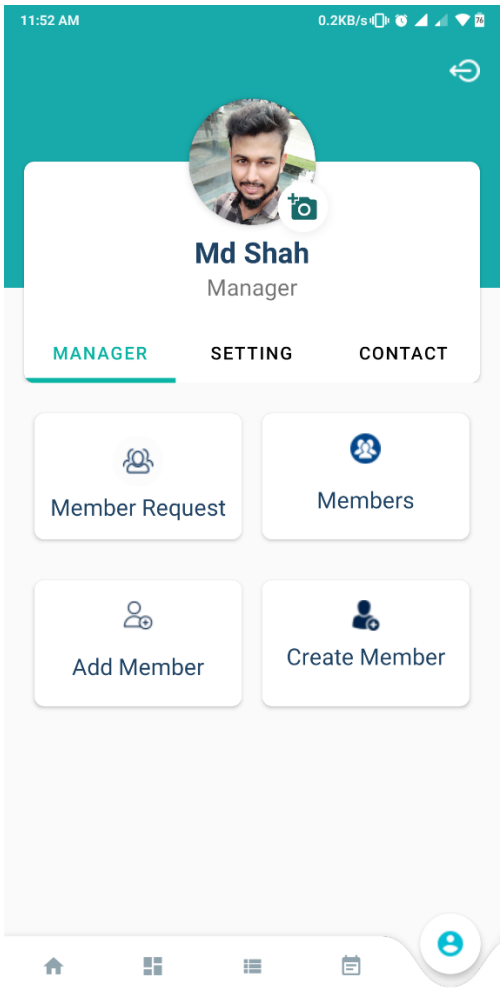


Figure 4.18: Manager Profile

Settings

Customer can set their name and contact information after the registration. In some cases they may need to change their information that why we pick the option.

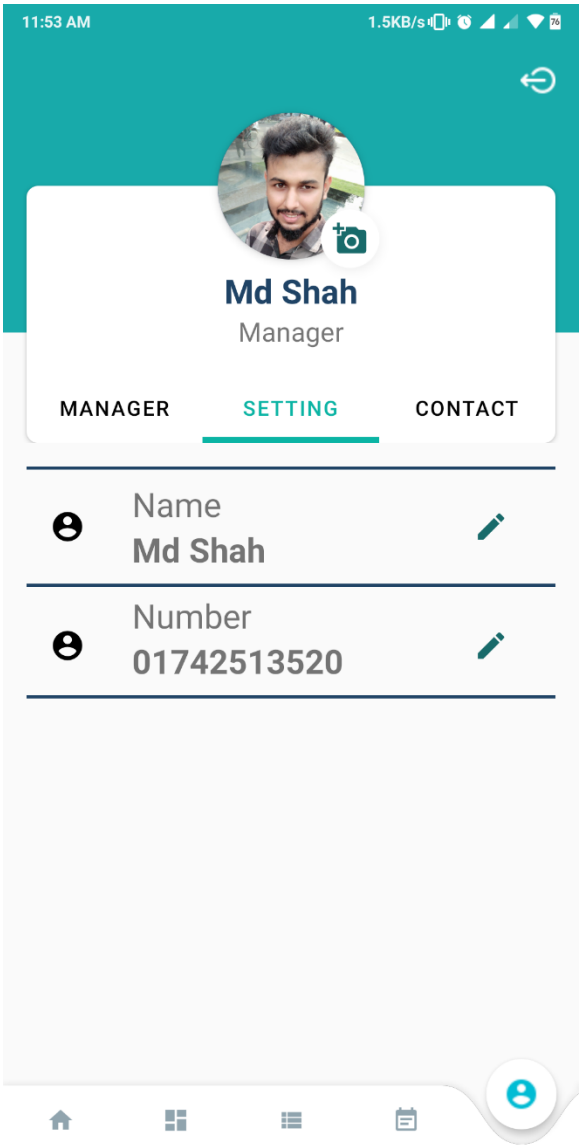


Figure 4.19: Settings

Contact

Member of meal management system can contact each other via this apps. So that there have free calling system in virtual world. It will save money at the same time it will make easier for communication between mess members.

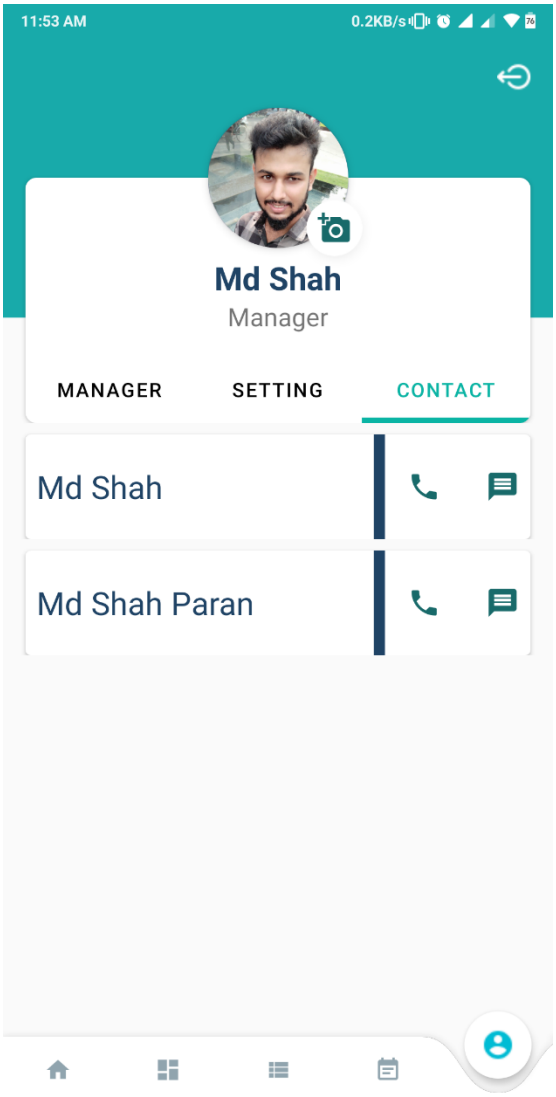


Figure 4.20: Contact

4.2 Back-end Design

We use java for back-end design. All of our back-end code is written by java. We can ensure that our back-end become enough strong which will secure customer data. Firebase real time database become used for stored data which is Cloud hosted database become use for store data. We have also use firebase [6] for user authentication.

4.3 Interaction Design and UX

We made our UX design as simple as we can. When we research on various site we could found out that more engagement depend on simple UX design. So we made it simple thinking about people requirement and mind. Customer satisfaction is important to us.

4.4 Implementation Requirements

For proper implementation we had to work a lot. We had to learn various type of programing language java. We had to think about the UX design. We needed huge time to make a proper and simple design. We use JDK, android studio, firebase real-time database and firebase server.

Chapter 5

Implementation and Testing

5.1 Implementation of Database

We gave our best effort on database because an apps security and handling depend on its database. We make our database using Google firebase database which is fast to response. We hope that we make a secure database. Gradually we will make it more secure.

Figure 5.1: Implementation of Database

5.2 Implementation of Front-end Design

The frond-end has been designed using XML. In front end design we basically focused on UI design. Here we use UI design tools. We use SDKs to access device characters. We also use Cross-platform accommodations.

5.2.1 Manifest

Each android should have manifest file [7]. It's give necessary information to android system. It's works as a unique identifier and it describe the component of the application. The manifest declare the permission.

```
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:dist="http://schemas.android.com/apk/distribution"
    xmlns:tools="http://schemas.android.com/tools"
    package="com.poran.bechalormanagement">

    <uses-permission android:name="android.permission.INTERNET" />
    <uses-permission android:name="com.google.android.c2dm.permission.RECEIVE" />
    <uses-permission android:name="android.permission.READ_EXTERNAL_STORAGE" />
    <uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE" />
    <uses-permission android:name="android.permission.CALL_PHONE" />
    <uses-permission android:name="android.permission.RECEIVE_BOOT_COMPLETED"/>

    <dist:module dist:instant="true" />

    <application
        android:allowBackup="true"
        android:hardwareAccelerated="true"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportsRtl="true"
        android:theme="@style/AppTheme"
        tools:ignore="GoogleAppIndexingWarning">



        <receiver android:name=".notifyService.LocalNotificationService">
            <intent-filter>
                <action android:name="com.poran.bechalormanagement" />
            </intent-filter>
        </receiver>
        <receiver android:name=".notifyService.MyBootLoader"
            android:enabled="true"
            >
            <intent-filter>
                <action android:name="android.intent.action.BOOT_COMPLETED"/>
                <action android:name="android.intent.action.REBOOT"/>
            </intent-filter>
        </receiver>

        <service
            android:name=".notifyService.MyfirebaseMessaginService"
            android:permission="com.google.android.c2dm.permission.SEND"
            >
            <intent-filter>
                <action android:name="com.google.firebase.MESSAGING_EVENT"/>
                <action android:name="com.google.android.c2dm.intent.RECEIVE"/>

            </intent-filter>
        </service>



        <activity
            android:name=".activity.SettingUserProfile"
            android:configChanges="orientation"
            android:screenOrientation="portrait" />
        <activity
            android:name=".activity.ExistanceBachelorPointList"
            android:configChanges="orientation"
            android:screenOrientation="portrait" />
        <activity
            android:name=".activity.SignInMethod"
            android:configChanges="orientation"
            android:screenOrientation="portrait" />
        <activity
            android:name=".activity.EmailSignUp"
            android:configChanges="orientation"
            android:screenOrientation="portrait" />
        <activity
            android:name=".activity.EmailLogIn"
            android:configChanges="orientation"
            android:screenOrientation="portrait" />
        <activity
            android:name=".activity.Navigation"
            android:configChanges="orientation"
            android:screenOrientation="portrait">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
        <activity
            android:name=".activity.OPTVarifyActivity"
            android:configChanges="orientation"
            android:screenOrientation="portrait" />
        <activity
            android:name=".activity.MainActivity"
            android:configChanges="orientation"
            android:screenOrientation="portrait" />
        <activity
            android:name=".activity.LoginActivity"
            android:configChanges="orientation"
            android:screenOrientation="portrait" />
        <activity
            android:name="com.theartofdev.edmodo.cropper.CropImageActivity"
            android:theme="@style/Theme.MaterialComponents.Light.Bridge" />


    </application>
```

Figure 5.2: Manifest

5.2.2 Permission overview

Permission overview represent the privacy. Permission of system must grand to operant for apps. Permission become granted when customer install an apps. Overall its protect the privacy. [8]

```
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:dist="http://schemas.android.com/apk/distribution"
    xmlns:tools="http://schemas.android.com/tools"
    package="com.poran.bechalormanagement">

    <uses-permission android:name="android.permission.INTERNET" />
    <uses-permission android:name="com.google.android.c2dm.permission.RECEIVE" />
    <uses-permission android:name="android.permission.READ_EXTERNAL_STORAGE" />
    <uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE" />
    <uses-permission android:name="android.permission.CALL_PHONE" />
    <uses-permission android:name="android.permission.RECEIVE_BOOT_COMPLETED"/>
```

Figure 5.3: Permission overview

5.2.3 Build gradle file

Build gradle file use for Building tool, it also work as build source code. At the same time works as a dependency set [9]

```
apply plugin: 'com.android.application'
apply plugin: 'com.google.gms.google-services'

android {
    compileSdkVersion 28
    defaultConfig {
        applicationId "com.poran.bechalormanagement"
        minSdkVersion 21
        targetSdkVersion 28
        versionCode 1
        versionName "1.0"
        testInstrumentationRunner "androidx.test.runner.AndroidJUnitRunner"
    }
    buildTypes {
        release {
            minifyEnabled false
            proguardFiles getDefaultProguardFile('proguard-android-optimize.txt'), 'proguard-rules.pro'
        }
    }
}

dependencies {
    implementation fileTree(dir: 'libs', include: ['*.jar'])
    implementation 'com.android.support:appcompat-v7:28.0.0'
    implementation 'com.android.support.constraint:constraint-layout:1.1.3'
    implementation 'com.google.firebase:firebase-core:17.0.0'
    implementation 'com.google.firebase:firebase-auth:18.0.0'
    implementation 'com.google.firebase:firebase-database:18.0.0'
    implementation 'com.google.firebase:firebase-storage:18.0.0'
    implementation 'com.firebaseui:firebase-ui-database:4.3.1'
    implementation 'com.google.firebase:firebase-messaging:19.0.1'




    testImplementation 'junit:junit:4.12'
    androidTestImplementation 'com.android.support.test:runner:1.0.2'
    androidTestImplementation 'com.android.support.test.espresso:espresso-core:3.0.2'
    implementation 'com.hbb20:ccp:2.2.4'
    implementation 'com.google.android.material:material:1.0.0'
    implementation 'com.android.support:recyclerview-v7:28.0.0'
    implementation 'com.airbnb.android:lottie:2.5.0-rc1'

    implementation 'com.android.support:cardview-v7:28.0.0'

    //glide
    implementation 'com.github.bumptech.glide:glide:4.4.0'
    annotationProcessor 'com.github.bumptech.glide:compiler:4.4.0'
    api 'com.theartofdev.edmodo:android-image-cropper:2.8.+'
    //circle imageview
    implementation 'de.hdodenhof:circleimageview:2.2.0'
    implementation 'com.android.support:design:28.0.0'
    implementation 'com.github.clans:fab:1.6.4'
    implementation 'com.etebarian:meow-bottom-navigation:1.0.1'
    implementation 'com.github.zcweng:switch-button:0.0.3@aar'
    implementation 'com.github.PhilJay:MPAndroidChart:v3.1.0'
    implementation 'org.greenrobot:eventbus:3.1.1'
    implementation 'com.google.code.gson:gson:2.8.5'
    implementation 'id.zelory:compressor:2.1.0'
    implementation 'io.reactivex.rxjava2:rxandroid:2.1.0'
    implementation 'com.android.volley:volley:1.1.1'
    implementation 'com.jaredrummler:material-spinner:1.3.1'


}
apply plugin: 'com.google.gms.google-services'
```

Figure 5.4: Build gradle file

5.3 Implementation of Interactions

In our project we call all data from database than we transfer it to the apps. For this reason our apps become lighter. In smart meal management customer can get service from apps with is basically controlled by mess manager and overall apps controlled by admin panel.

5.4 Testing Implementation

After completing at task, we tasted our project and it's successful. It transfer data properly. Our testing result is satisfactory but we will working hard to make it more accurate.

5.5 Test Results and Reports

Test result is 97% which is satisfactory to us. We are hoping that we will make in more satisfactory.

CHAPTER 6

Discussion and conclusion

6.1 Discussion and Conclusion

This project will make our life comfortable in mess. We have to face many problem in mess because of calculation problem. As all most people of a mess is a students or bachelor job holder so that all most people become busy on their work. At the same time there have some financial crisis in Bangladesh so that many mess member tries to avoid go to the market. In this situation relationship between mess members become worsen .Our apps can solve all the problem of a mess. Even they can call each other via our apps. They can calculate the nutrition system of a mess. People can find out a proper mess via our website. As day by day mess member is increasing so we hope that our apps will run well. We have few marking policy which can spread our project so quickly. At 1st stage we want to work in our country especially in Dhaka city gradually we want to spread it all over world.

6.2 Scope for Further Developments

We have many scope to develop our project in future. Scope of the project is written below:

1. We will work on international level.
2. We will provide our own security system.
3. We will create our own database system for customers.
4. We will provide nutrition information.
5. Members of our apps will get opportunity to chat via our apps.
6. We will provide cost details analyzing data of mess and meals.

REFERENCE:

[1] Meal Manager is a cloud base, https://play.google.com/store/apps/details?id=com.self.rayhan.mealmanagementfirestore [Accessed March 1st 2019]

[2]Mess Manager, https://play.google.com/store/apps/details?id=com.m27lab.messmanager.app [Accessed March 1st 2019]

[3]Hostel Meal Manager, https://play.google.com/store/apps/details?id=com.aktarulahsan.hostel_meal_manager [Accessed March 1st 2019]

[4] Flatten data structures, https://firebase.google.com/docs/database/web/structure-data [Accessed May17th 2019]

[5] Json data modeling, https://blog.couchbase.com/json-data-modeling-rdbms-users/ [Accessed June 23th 2019]

[6] Firebase Realtime, Database https://firebase.google.com/products/realtime-database [Accessed April 7th 2019]

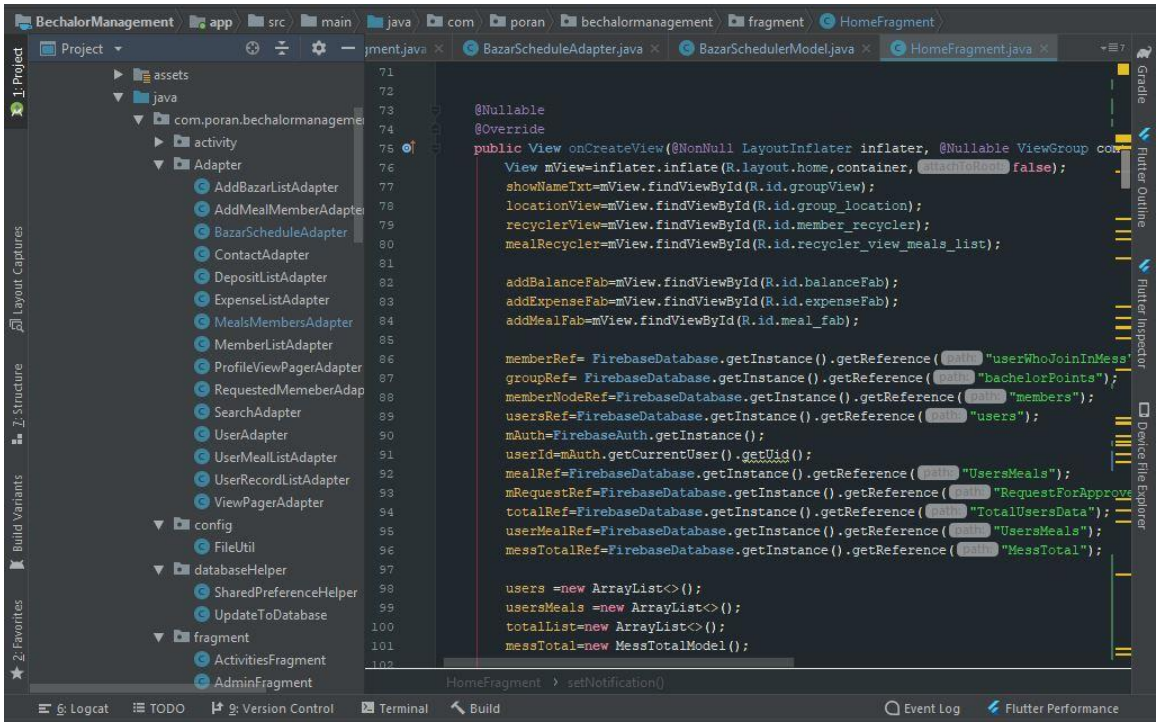
[7] App menifest overview, https://developer.android.com/guide/topics/manifest/manifest-intro [Accessed May 13th 2019]

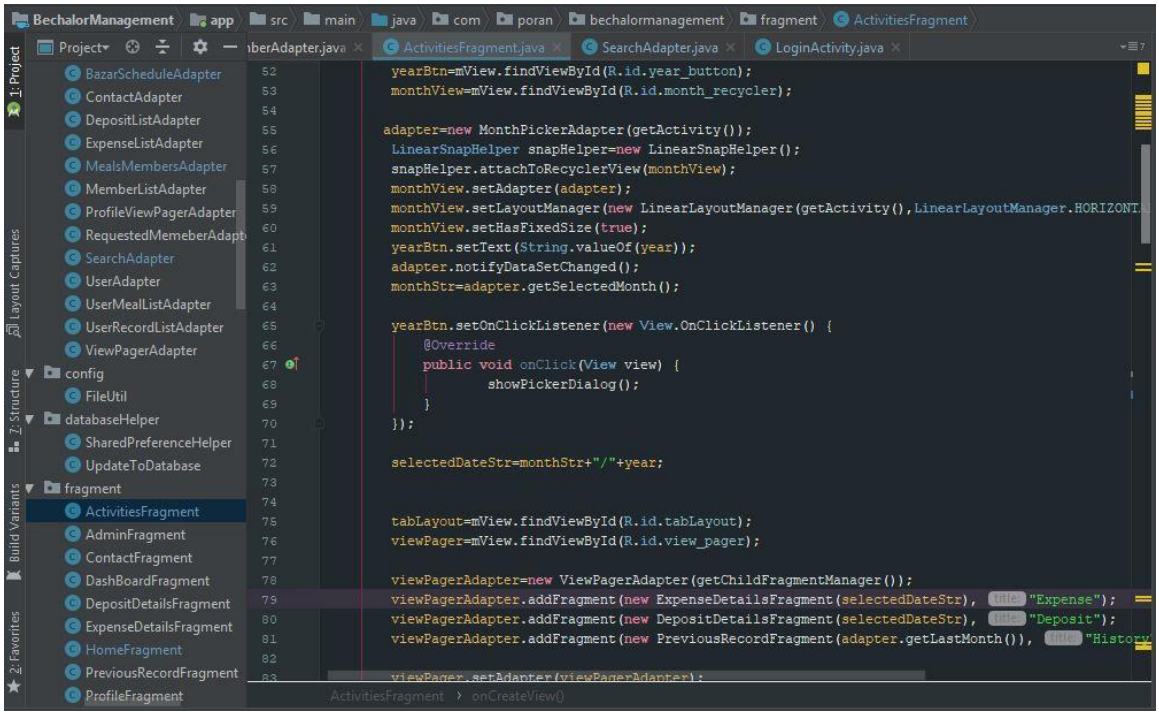
[8] Android permission overview, https://developer.android.com/guide/topics/permissions/overview [Accessed May 8th 2019]

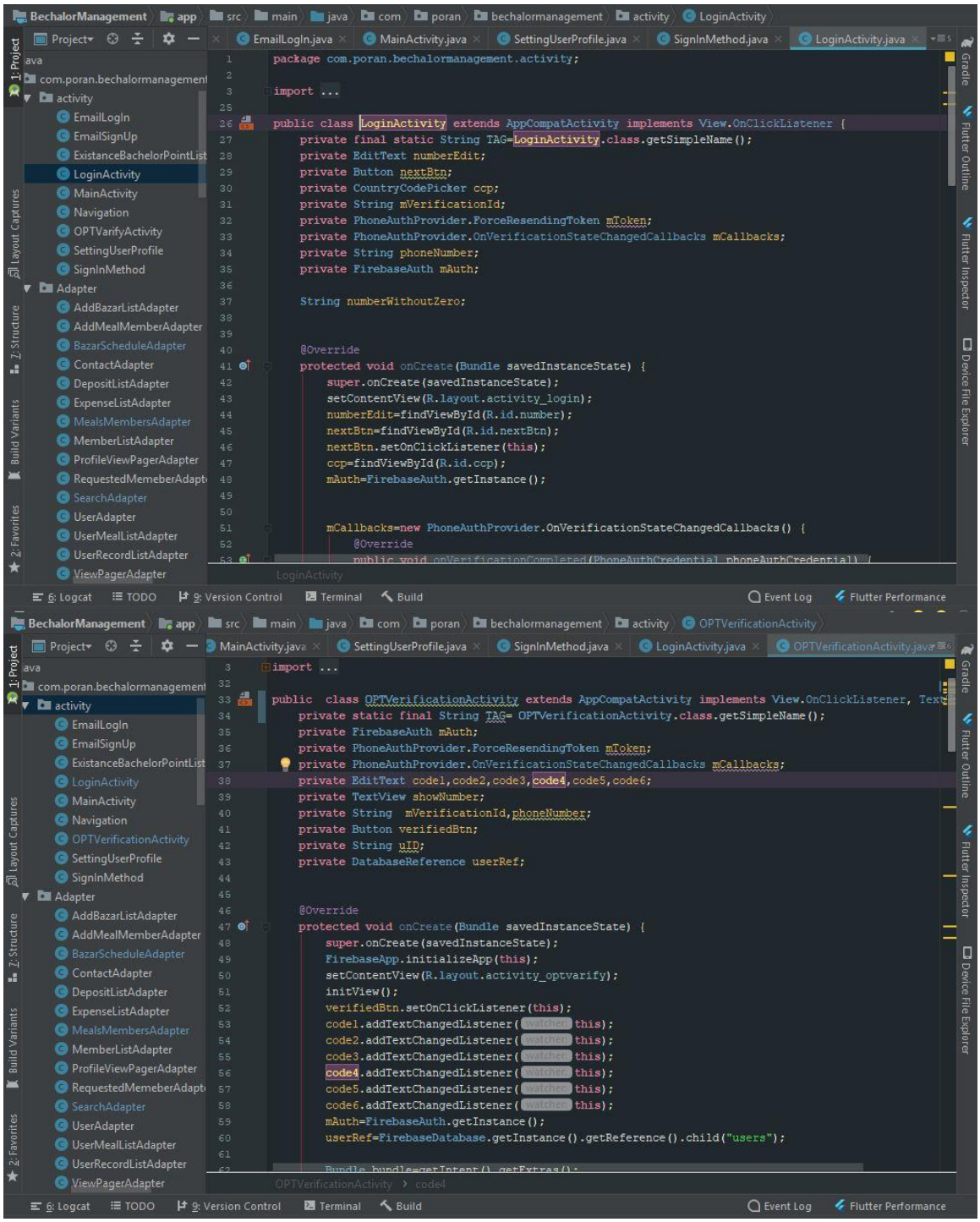
[9] Using gradle for android apps, https://www.vogella.com/tutorials/AndroidBuild/article.html [Accessed May 8th 2019]

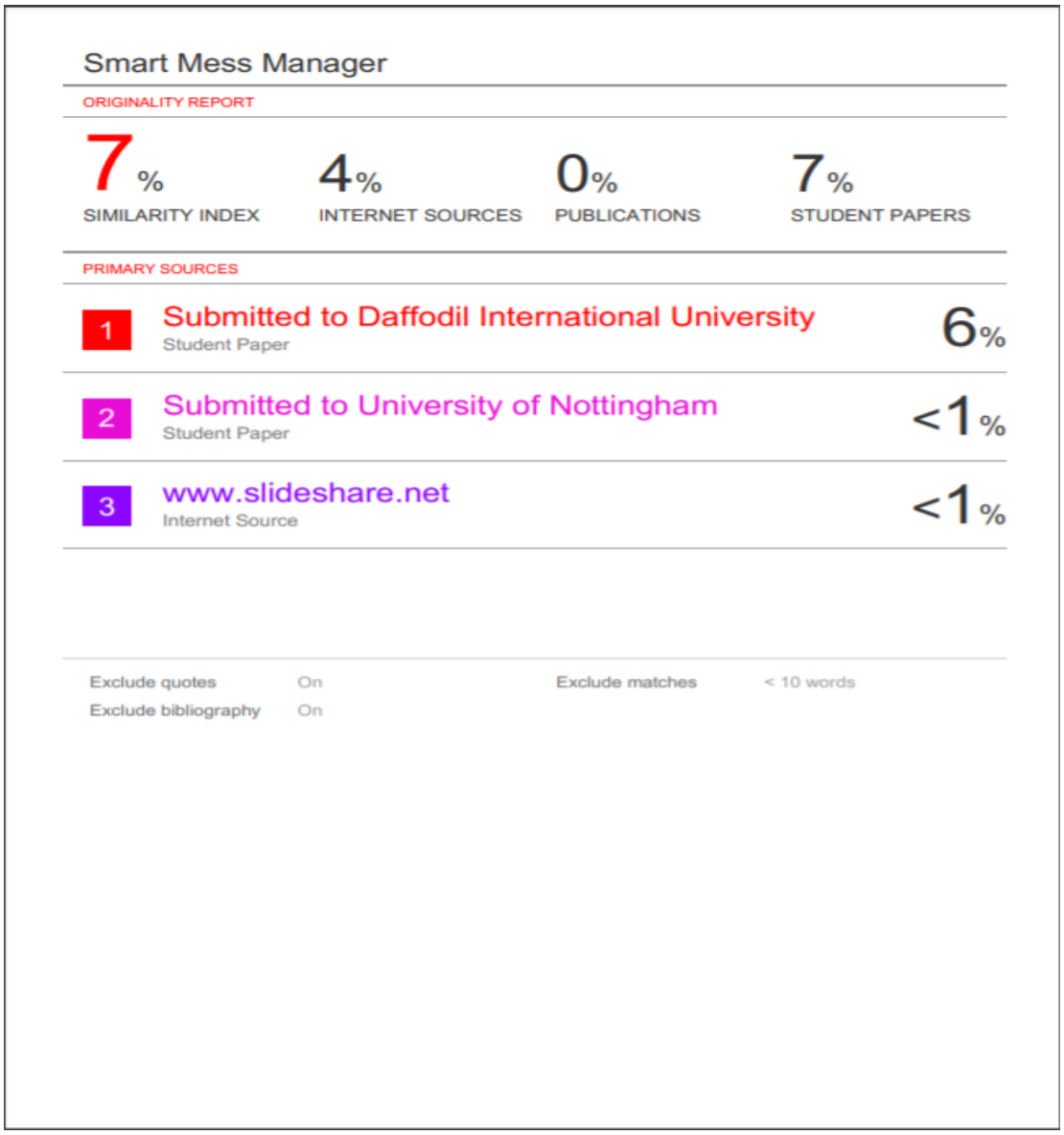
Appendices

Appendix A: Project Reflection









Smart Mess Manager

ORIGINALITY REPORT

7%	4%	0%	7%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	Submitted to Daffodil International University Student Paper	6%
2	Submitted to University of Nottingham Student Paper	<1%
3	www.slideshare.net Internet Source	<1%

Exclude quotes	On	Exclude matches	< 10 words
Exclude bibliography	On		