

**ANDROID BASED COMMUNITY CENTER MANAGEMENT SYSTEM**

**BY**

**MEHEDI HASAN SAIKAT**

**ID: 162-15-7746**

**MOHAMMED MAHMUDUL HASAN**

**ID: 162-15-7760**

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

Supervised By

**NAZMUN NESSA MOON**

Assistant Professor

Department of CSE

Daffodil International University

Co-Supervised By

**Gazi Zahirul Islam**

Assistant Professor

Department of CSE

Daffodil International University



**DAFFODIL INTERNATIONAL UNIVERSITY**

**DHAKA, BANGLADESH**

**JULY 2020**

## APPROVAL

This Project titled “**Android Based Community Center Management System**”, submitted by **Mehedi Hasan Saikat and Mohammed Mahmudul Hasan** 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 July 2020.

### BOARD OF EXAMINERS



**Dr. Syed Akhter Hossain**  
**Professor and Head**

Department of Computer Science and Engineering  
Faculty of Science & Information Technology  
Daffodil International University

**Chairman**

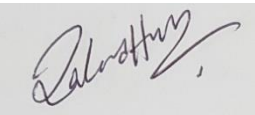


**Dr. Sheak Rashed Haider Noori**

**Associate professor & Associate Head**

Department of Computer Science and Engineering  
Faculty of Science & Information Technology  
Daffodil International University

**Internal Examiner**

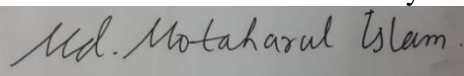


**Md. Zahid Hasan**

**Assistant Professor**

Department of Computer Science and Engineering  
Faculty of Science & Information Technology  
Daffodil International University

**Internal Examiner**



**Dr. Md. Motaharul Islam**

**Professor**

Department of Computer Science and Engineering  
United International University

**External Examiner**

## DECLARATION

We hereby declare that, this project has been done by us under the supervision of **Nazmun Nessa Moon, Assistant Professor, Department of CSE** 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:



---

**Nazmun Nessa Moon**  
Assistant Professor  
Department of CSE  
Daffodil International University

### Co-Supervised by:



---

**Gazi Zahirul Islam**  
Assistant Professor  
Department of CSE  
Daffodil International University

### Submitted by:



---

**Mehedi Hasan Saikat**  
ID: 162-15-7746  
Department of CSE  
Daffodil International University



---

**Mohammed Mahmudul Hasan**  
ID: 162-15-7760  
Department of CSE  
Daffodil International University

## ACKNOWLEDGEMENT

Firstly, 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 **Nazmun Nessa Moon, Assistant Professor**, Department of CSE, Daffodil International University, Dhaka. Deep Knowledge & keen interest of our supervisor in the field of “*android application*” to carry out this project. Her endless patience, scholarly guidance, continual encouragement, constant and energetic supervision, constructive criticism, valuable advice, reading many inferior drafts and correcting them at all stage have made it possible to complete this project.

We would like to express our heartiest gratitude to Dr. Syed Akhter Hossain, Professor and 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 project “**Android Based Community Center Management System**” is a system for reducing the complexities of booking process of a community center. This project is for a community center so that people can book this using an android application. Now-a-days there is a plenty of community centers in Bangladesh but very few has their own online based management system. Most of the online platforms just provide addresses of the community centers. For these reasons, we’ve developed a mobile application for a community center with many features. This application will be helpful for both users and owner of the community center. To develop this project, we have used Firebase Firestore database and Android Studio as IDE. This system is tested in many ways after implementation and it works successfully.

# TABLE OF CONTENTS

<b>CONTENTS</b>	<b>PAGE</b>
Board of examiners	i
Declaration	ii
Acknowledgements	iii
Abstract	iv
List of Figures	viii
List of Tables	x
<b>CHAPTER 1: INTRODUCTION</b>	<b>1-3</b>
1.1 Introduction	1
1.2 Motivation	1
1.3 Objectives	2
1.4 Expected Outcome	2
1.5 Report Layout	3
<b>CHAPTER 2: BACKGROUND</b>	<b>4-7</b>
2.1 Introduction	4
2.2 Related Works	4
2.3 Comparative Studies	6
2.4 Scope of the Problem	7
2.5 Challenges	7

<b>CHAPTER 3: REQUIREMENT SPECIFICATION</b>	<b>8-13</b>
3.1 Business Process Modeling	8
3.2 Requirement Collection and Analysis	9
3.3 Use Case Modeling and Description	9
3.4 Logical Data Model	12
3.5 Design Requirements	13
<b>CHAPTER 4: DESIGN SPECIFICATION</b>	<b>14-21</b>
4.1 Front-End Design	14
4.2 Back-End-Design	16
4.3 Interaction Design and UX	19
4.4 Implementation Requirements	21
<b>CHAPTER 5: IMPLEMENTATION AND TESTING</b>	<b>22-38</b>
5.1 Implementation of Database	22
5.2 Implementation of Front-end Design	23
5.2.1 User Module	24
5.2.2 Admin Module	31
5.3 Implementation of Interactions	35
5.4 Testing Implementation	35
5.5 Test Results and Reports	38
<b>CHAPTER 6: CONCLUSION AND FUTRUE SCOPE</b>	<b>39</b>
6.1 Discussion and Conclusion	39
6.2 Scope for Future Developments	39

<b>REFERENCES</b>	<b>40</b>
<b>APPENDIX</b>	<b>41</b>
Appendix A: Project Reflection	41
<b>PLAGIARISM</b>	<b>42</b>
Plagiarism Report	42



## LIST OF FIGURES

<b>FIGURES</b>	<b>PAGE NO</b>
Figure 2.1: website of Senamalancha community center	5
Figure 3.1: Business Process Model for booking process	8
Figure 3.2: Use Case Model for Users (customers)	10
Figure 3.3: Use Case Model for Admin	11
Figure 3.4: ER diagram of the system	12
Figure 4.1: User Registration Page in Android Studio	14
Figure 4.2: Booking Page from Android Studio	15
Figure 4.3: Admin Home Page from Android Studio	16
Figure 4.4: Firebase Authentication Page	17
Figure 4.5: Firebase Storage Page	18
Figure 4.6: Firebase Database (Cloud Firestore) Page	19
Figure 5.1: Registration using Firebase Authentication and Cloud Firestore with JAVA code	22
Figure 5.2: Firebase Database	23
Figure 5.3: User Home Page	24
Figure 5.4: Admin Home Page	24
Figure 5.5: User Log In Page	25
Figure 5.6: User Registration Page	25
Figure 5.7: After Log In	25
Figure 5.8: User Profile Update	25
Figure 5.9: Gallery	26
Figure 5.10: Food Packages	26
Figure 5.11: Booking Information	27
Figure 5.12: FAQ	27
Figure 5.13: Booking Page	28
Figure 5.14: Booking History	28
Figure 5.15: Confirmed Booking Details	29

Figure 5.16: Pending Booking Details	29
Figure 5.17: Message	30
Figure 5.18: Service Rating	30
Figure 5.19: Admin Check Booking	31
Figure 5.20: Admin Confirm/Cancel Booking	31
Figure 5.21: Admin Confirmed Booking list	32
Figure 5.22: Admin Pending Booking list	32
Figure 5.23: Admin Edit Packages	33
Figure 5.24: Admin Manage FAQ	33
Figure 5.25: Admin Photo Upload	34
Figure 5.26: Admin Reply message	34

## LIST OF TABLES

### TABLES

### PAGE NO

Table 1: Testing Information and Date

35

# CHAPTER 1

## INTRODUCTION

### 1.1 Introduction

Mobile applications especially android apps are becoming very popular day by day. Many people in Bangladesh are now interested to arrange programs in community centers. There are a lot of community centers in Bangladesh and many are launching their business. But none of them has their own android applications. So, we have developed an android application for a community center. Our application has features like checking availability of the community center for reservation, checking many food packages offered by the community center, user account creation, booking a day or evening slot etc.

There are two types of users in this application, admin and public users. Public users need to register through registration or sign up process of the application. After successful log in, users will be able to book the community center and the features related to database will be available to them. For data security, only authenticated users are allowed to read the database. Without logging in, people will be able to use this app but only limited features will be available to them. Features related to database will not be available to them. There is a separate logging system for the admin. Only admin will be able to log in through this, others will be denied. Admin features like booking confirmation, cancellation, booking history deletion, updating rental information, photo upload in gallery etc. will be available after successful log in by an admin. Admin is able to book any date.

### 1.2 Motivation

Number of community centers is increasing day by day in Bangladesh. Most of the online platforms provide us only the address of the community centers. That's why we want to build a system for a community center. People will have the opportunity to reserve the community center checking availability and check offered food packages with price.

People are so busy. Sometimes we don't have enough time to visit any community center and get information asking them one by one. One of the major targets of technology is to make everything easy and reduce complexities. An online platform is the best solution for

both users and community center owner to reduce complexities of the booking process. As android mobile phones are the most popular devices in the recent time, so android application will be convenient for the users.

### **1.3 Objectives**

- To reserve community center
- To check availability or vacancy on any specific date
- To get an assumption about the cost checking packages
- To reduce community center owners and members complexities and run center more efficiently
- To check available facilities of the community center
- To save time and money

### **1.4 Expected Outcome**

1. Community Center's owner or admin will be able to manage everything through this application.
2. People will have the opportunity to check facilities provided by the community center.
3. People will be able to check packages, reserve community center being at home just by an android phone.
4. Communicating with the community center through the application is possible using message option.
5. They will not need to go to the community center physically.
6. It will save their time and other complexities.

### **1.5 Report Layout**

#### **Chapter 1: Introduction**

Motivation, objectives and the expected outcome of the project have been discussed in this chapter. The report layout has been mentioned in the last part of this chapter.

## **Chapter 2: Background**

Background circumstances of our project have been discussed here. Related work, comparing with other systems, the scope of the problem and challenges of the project are explained here.

## **Chapter 3: Requirement Specification**

This chapter discusses about the requirements. Business process modeling, the requirement collection and analysis, the use case model of the project with description, the logical relational database model and the design requirements are the topics of this chapter.

## **Chapter 4: Design Specification**

This chapter consists of all the designs of the project. These are Front-end design, back-end design, Interaction design and UX and the implementation requirements.

## **Chapter 5: Implementation and Testing**

Implementation of database, front-end designs and interactions, testing implementation, test results and reports are the contents of this chapter.

## **Chapter 6: Conclusion and Future Scope**

It is the last chapter where conclusion and future scope of the project have been discussed.

# **CHAPTER 2**

## **BACKGROUND**

### **2.1 Introduction**

For starting a development project, we need to research the related projects done previously. The chapter background is very important for completing a project properly. We can get relevant information needed to complete the project from previous works which are related to the project we want to work. We need to check whether anyone did this type of project or not. If we get related projects, then we need to identify their limitations. We can work on the limitations to remove these and improve or develop the system and finally build a new system. Without researching related works, we will not get the ideas on existing features of the related projects. As a result, we may have a very narrow scope to improve existing features and adding new ones. For these reasons, background of the project plays a vital role to build a successful project.

## **2.2 Related Work**

There are some websites which provides the address of community centers in Bangladesh. Some others have very limited features. Some platforms for finding community centers in Bangladesh are:

- Parjatanbd.com [1]
- Mediabangladesh.net [2]
- Biyebazaar.com [3]
- Thebride.com [4]
- Communitycenterbd.com [5]

**Features:** These websites are basically common platforms to find community center. They are not managed by any specific one community center.

- Provides only address of the center
- Some provides only facilities of the centers
- Can search community center in an area

### **Problems:**

- Third party media
- Can't book directly

- Can't communicate with the authentic authorities of any community center through the website

There is almost no mobile application in Bangladesh where one can book a center which is run by the center's authentic authority. Most of the community centers don't have their own online based management system. Senamalancha [6] is a community center in Dhaka, Bangladesh which runs through a website shown in figure 2.1 but they have no mobile application.

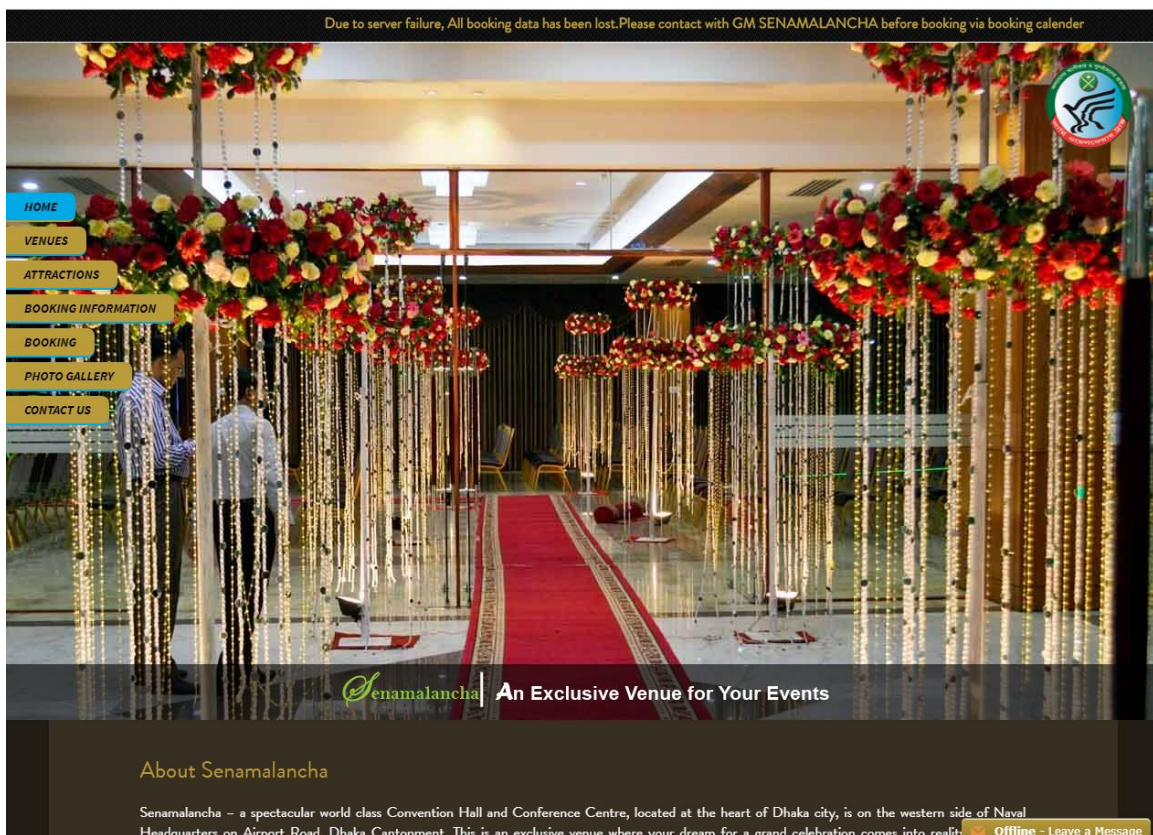


Figure 2.1: website of Senamalancha community center

**Features:**

- Online booking is possible
- Booking information is available
- Photo gallery



- Contact option

### **Problems:**

- Doesn't provide enough information about their offered food packages
- There is no service rating option
- No user registration option
- Security issues

## **2.3 Comparative Studies**

Our app contains many features. Some are similar and some different comparing to the mentioned web applications. Moreover, we have developed this in mobile platform which is a new development because the existing related projects are all websites.

- People can book the center directly checking available date. In some websites people need to send request of booking, then the authority replies them ensuring availability and booking.
- Checking offered food packages with price by the center is not available in other projects, we've included this feature in our app.
- People can give rating after taking a service from the community center in our app. Service rating is also a new feature in our app comparing the existing websites.
- Security is an important issue. We have used Firebase authentication system for users which is managed by google.
- Users can check their booking history (upcoming, current and previous) and cancel if it is not confirmed. It's also a developed and different type of feature of our app.

## **2.4 Scope of the problem**

We've faced many problems to complete this project. We overcame these difficulties. We needed to know about the booking process of the community centers. We collected information about the community centers in Bangladesh. Database choice is also a crucial decision. We selected Firebase to use as database. Android studio, the IDE, we have used

in our project have two options to choose a programming language either JAVA or KOTLIN. We've chosen JAVA as our programming language. Android mobiles are available and popular nowadays. As this project is based on android, I hope many people will use this type of app which will reduce their complexities.

## **2.5 Challenges**

Some challenges we have faced are mentioned below:

- Firebase Firestore database was a new tool for us. As we have used this database in our project, so we needed to learn a lot about this database.
- Beautiful design and unique features.
- Developing user friendly interface.
- Handling two parts, common users and admin in one app
- Ensuring data security
- Blocking unauthorized users from reading, writing and deleting data from database.

## **CHAPTER 3**

### **REQUIREMENT SPECIFICATION**

#### **3.1 Business Processing Model**

Business Process Model is a very important tool to build a flowing diagram or model. It's a mapping concept. It defines actual flow of data. It is process of creating a structural view of a process or system. Everyone needs to develop a business process model before starting a project.

From the booking page of our app, a user can book the community center checking the availability on any specific date. Then admin can cancel or confirm the booking request. The following figure 3.1 shows the business process model for this process.

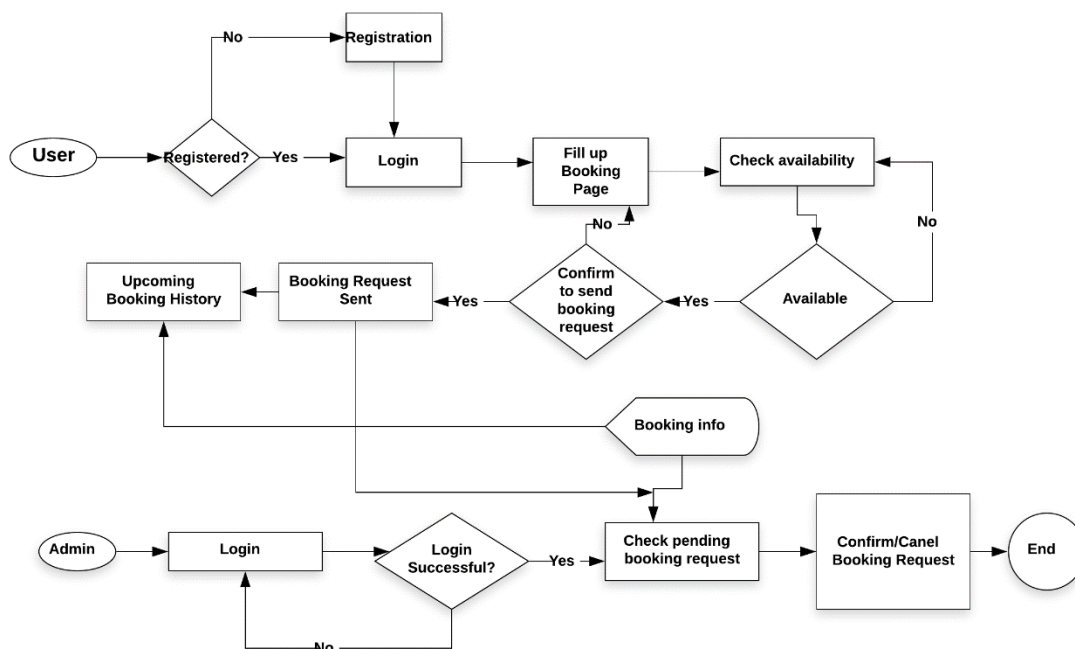


Figure 3.1: Business Process Model for booking process

### 3.2 Requirement Collection & Analysis

Specifying and collecting requirements are the initial work for any project. We need requirements to draw a graphical view of a project. It is difficult to implement a project without graphical view. Requirement collection is a very big challenge for a project because requirement is not available everywhere. Sometimes there is a chance to get

wrong information too. We must have to collect information from proper sources. So, we need to be very careful about requirement collection. For our project named community center management system, we collected requirements before starting to write codes. We collected information from many sources. We searched websites of community centers and websites that provide information about community centers located in Bangladesh. We collected some required information from many websites and we needed to contact with many community centers' manager to get more specific and accurate information. We knew about the booking process and services that a community center provides. Collecting requirements, we analyzed these. After analysis, we prepared many methods models to represent the system. For doing a successful project, we need to collect requirement, then we need to analyze the collected requirements and specify the method for the project.

### **3.3 Use Case Modeling and Description**

Use case diagram is a behavioral UML diagram type. This diagram is used to do analysis on many systems. These diagrams are useful to realize the different types of roles in a system and to detect how those roles interact with system. One of the purposes of a use case diagram is to identify functions and how roles interact with them. Use case diagram is important for a high level view of the system. Internal and external factors can be identified by this diagram [7].

We have created two use case models; one is for admin and the other one is for user(customer). Figure 3.2 shows use case model for user(customer) and figure 3.3 shows use case model for admin.

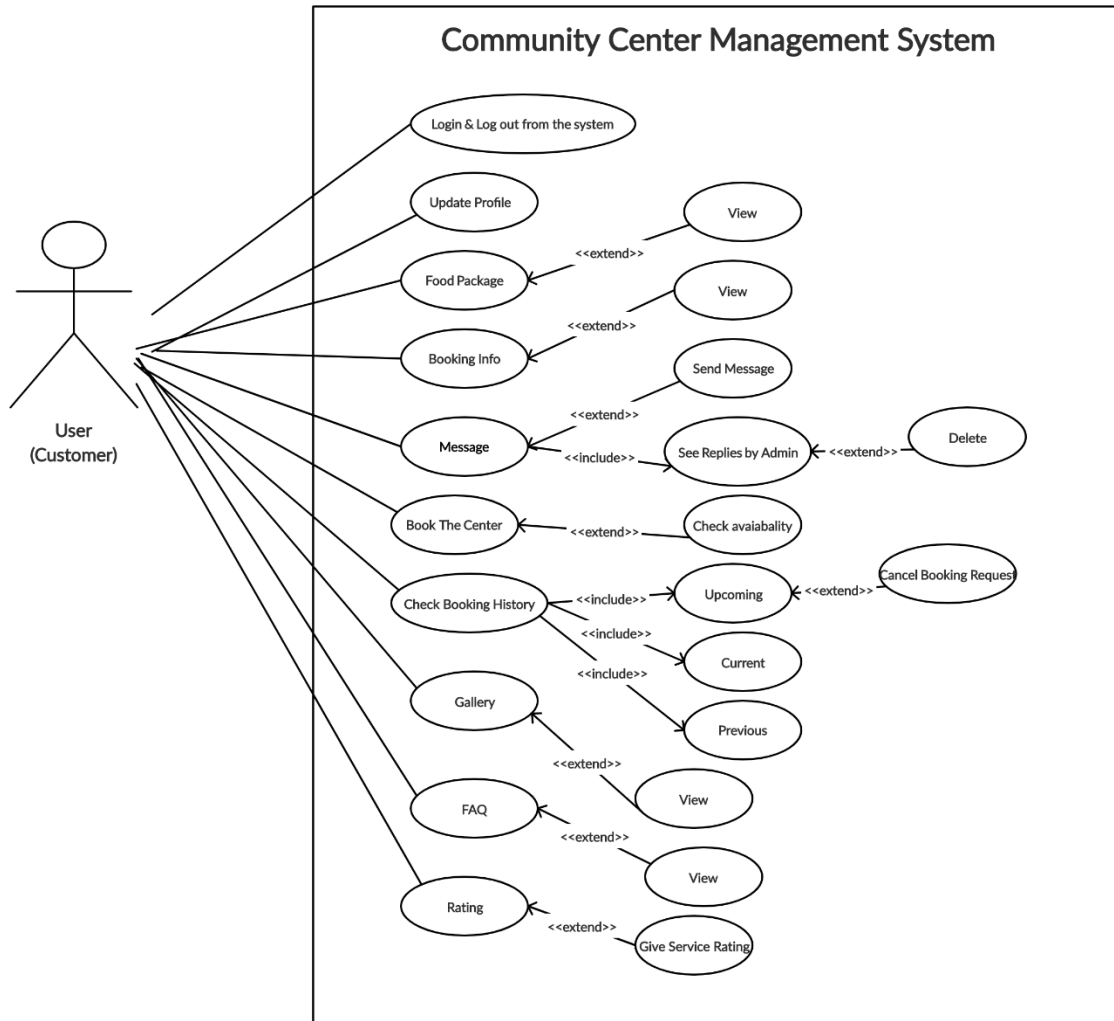


Figure 3.2: Use Case Model for Users (customers)

From the use case diagram shown in figure 3.2, we can see many functions of users. There is sign up and sign in option for users. They can update their profile information after logging in to the system. From packages, they can view many types of food package. They can view booking information from booking info page. There is message option from where they are able to send messages and see the replies of their messages by admin, and delete messaging history option is included there. The use case named book the center has a condition that the center has to be vacant. There are also some use cases named check booking history, gallery, FAQ, rating. Users have only the view option for gallery and

FAQ. They can give rating if they've taken a service from the community center. They can cancel a booking request if the booking status is Not Confirmed.

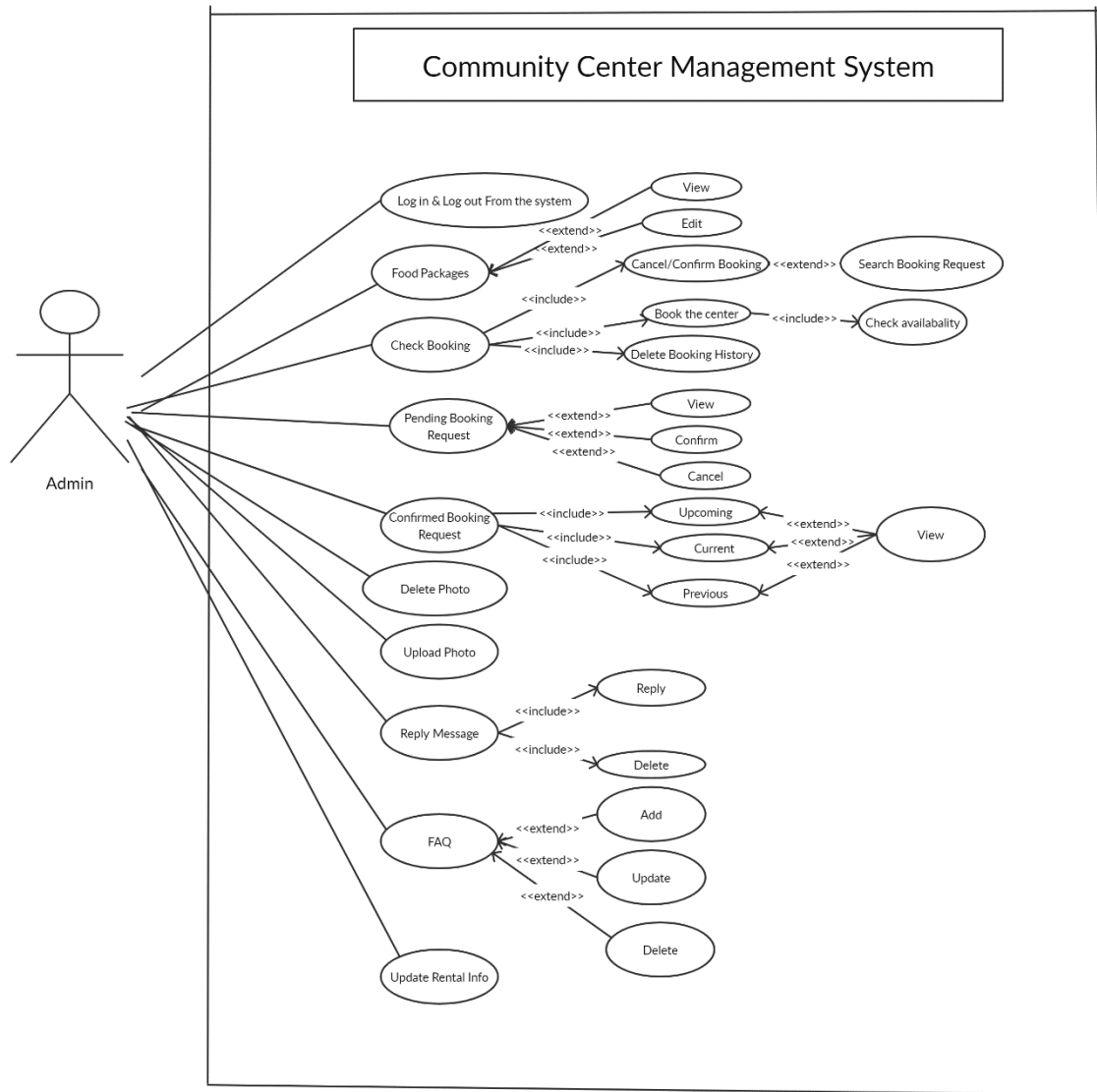


Figure 3.3: Use Case Model for Admin

From this use case model, we can see the activities of admin. Admin can view and edit food packages. Admin has many other options which we can see from the use case model in figure 3.3.

### 3.4 Logical Data Model

The logical data model is used for reporting the database elements of a business sector. The entities and the relationships shared by them are the center elements of ER models [8].

The Entity Relationship Diagram (ERD) is a logical data model. ER diagram of our project is shown in figure 3.4.

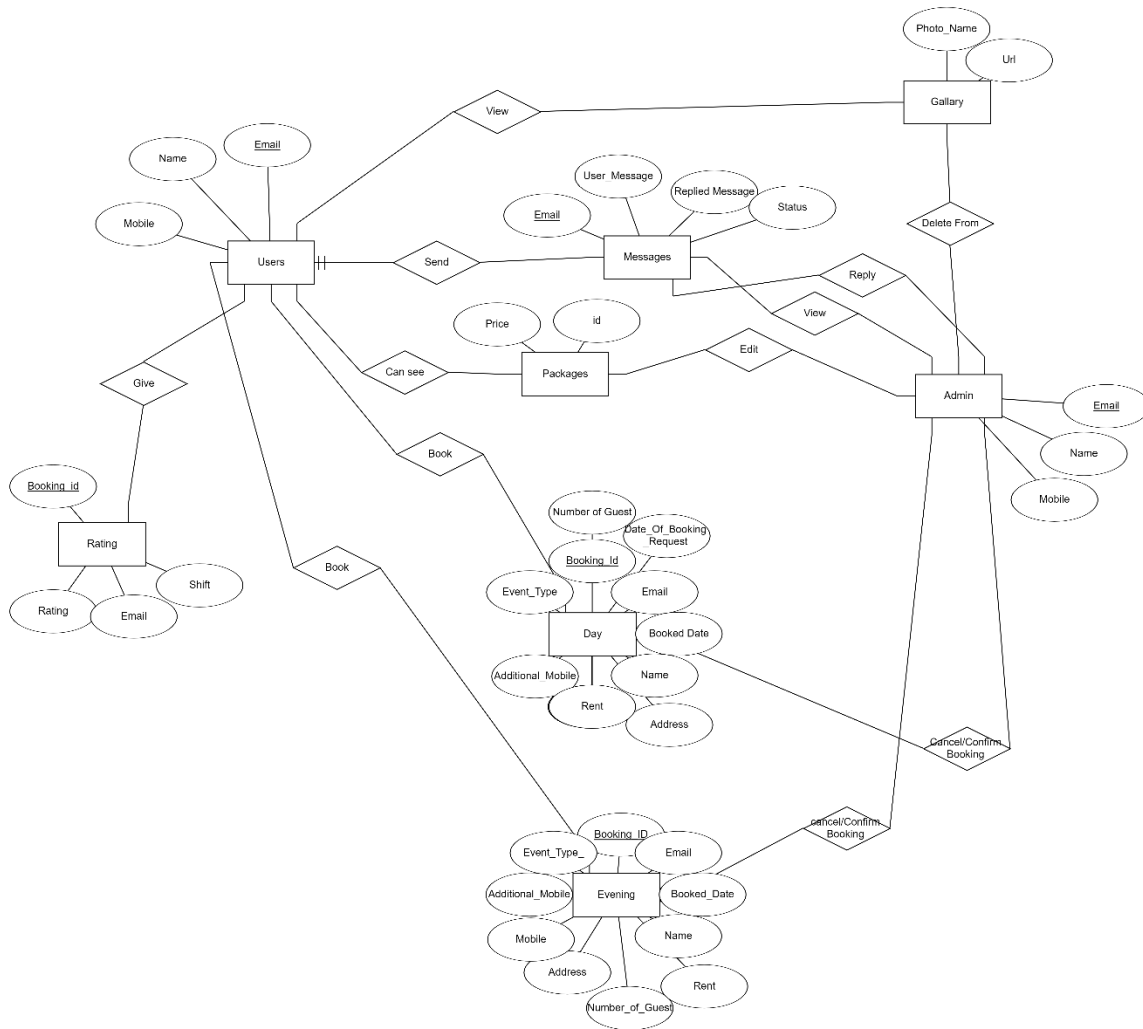


Figure 3.4: ER diagram of the system

### 3.5 Design Requirements

- This application has two type of user; admin and general users(customers).
- General users have registration or sign up option.
- After sign in, users can update their profile information.
- People can book the community center for any shift checking availability on any specific date.
- After booking, a user is able to check the booking information from upcoming option of booking history.
- After taking a service from the community center, a user can give rating for the service.
- Users can check food packages with price, see galley, frequently asked questions.
- User can send message and see the replies of their message by admin.
- Admin can edit food packages.
- Admin can cancel or confirm any pending booking request, search booking information.
- Admin can delete booking history from database.
- Admin can also book the center just choosing shift and available date.
- Admin can check pending and confirmed booking list.
- All the messages sent by users are displayed in the reply message option in admin home page, admin can reply or delete any message from there.
- Admin can upload photos to the gallery and delete from gallery.
- Admin can add, update, delete an FAQ.
- Admin can update rental information.

While designing the system, we tried to make the system simple and user friendly for the users. Efficiency is important. We keep it in mind so that our system can run in the most of the android devices efficiently and smoothly.

## **CHAPTER 4**



# DESIGN SPECIFICATION

## 4.1 Front-End Design

In any application, front-end-design is very important because users interact with the application by this. We've tried to make it user friendly and easily understandable by the users.

To make the user interface, we have used XML (mark-up-language). Our application has many pages, we needed to design every single page. Let's see some pages with their design and code.

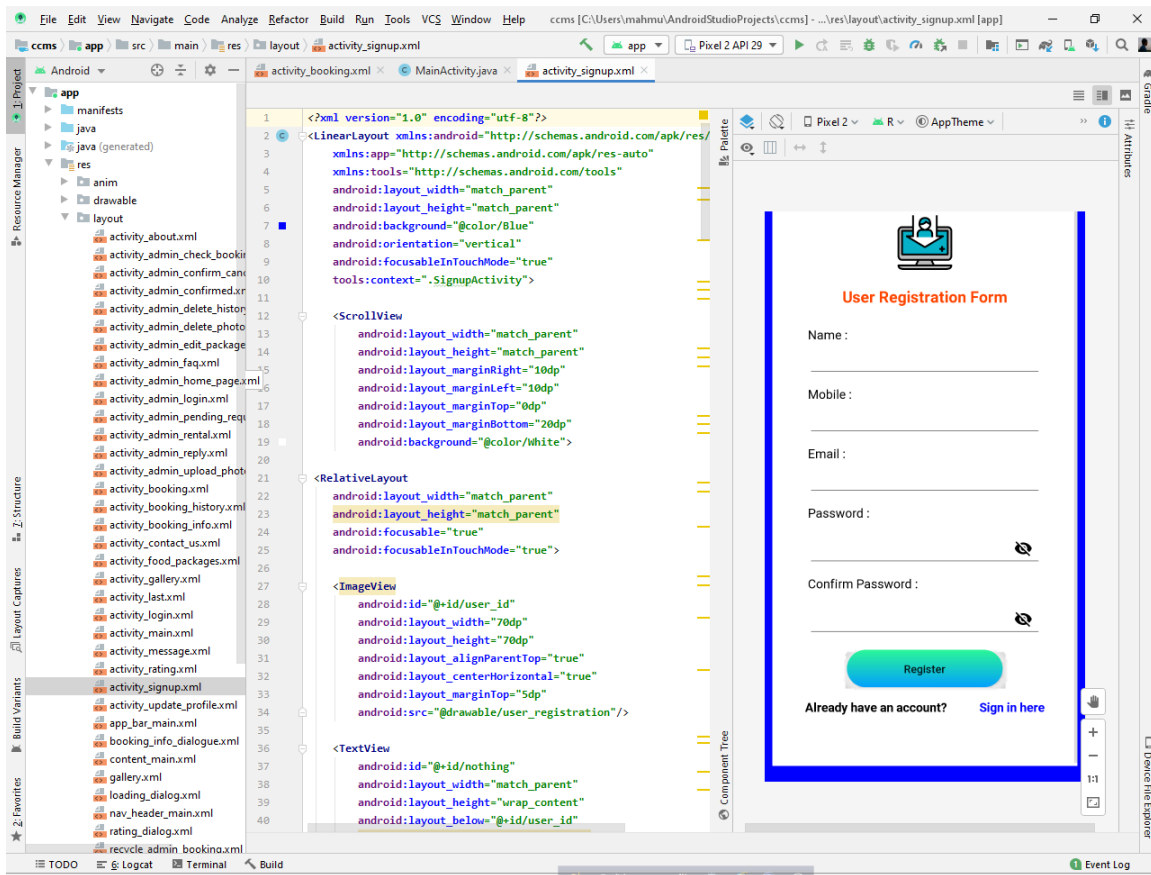


Figure 4.1: User Registration Page in Android Studio

This is the design of user registration page with XML code of our app shown in figure 4.1. But it is better to mention that the shown XML code is not the full code for this design, it's just a part of full code. And there is also a JAVA code for the page to perform the actions.

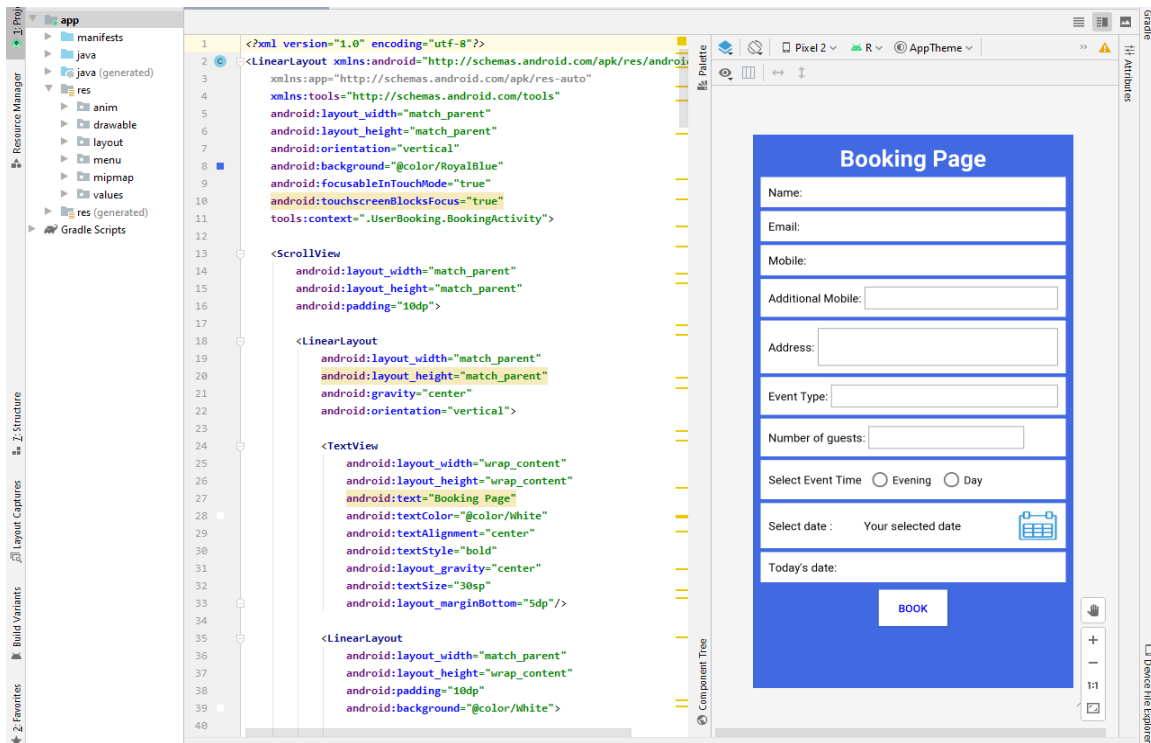


Figure 4.2: Booking Page from Android Studio

This is the design of our Booking Page. Part of XML code is also shown in the figure 4.2.

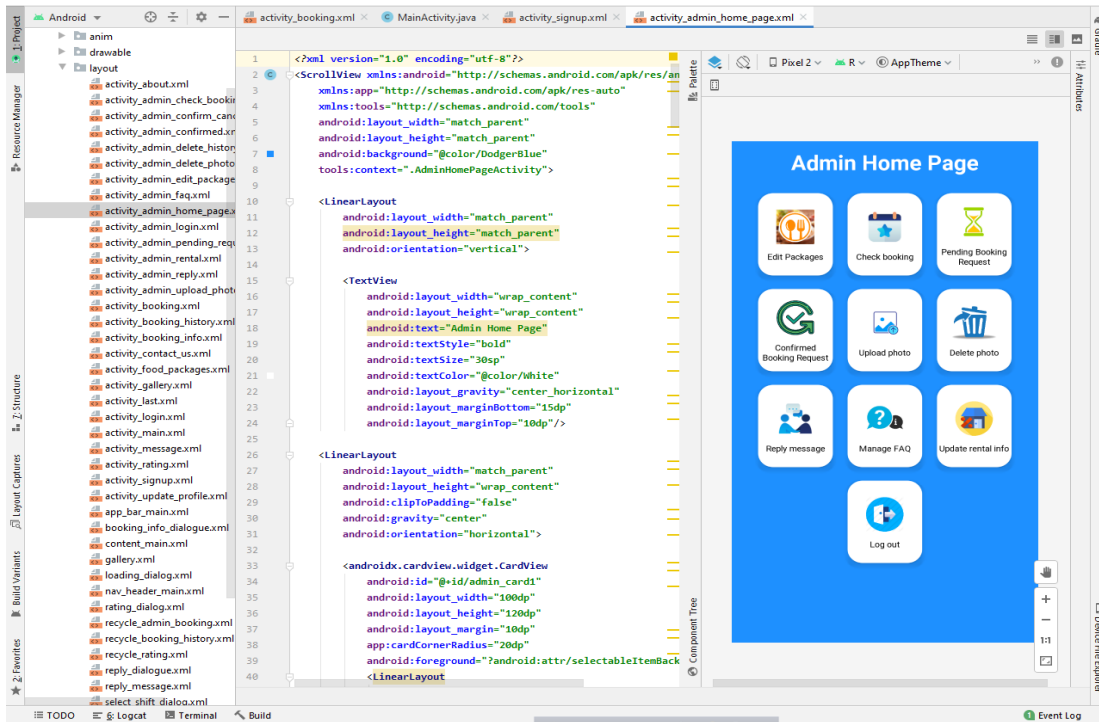


Figure 4.3: Admin Home Page from Android Studio

Figure 4.3 shows the design of admin home page of our app with some part of its XML code. There are so many pages in our app, every page has its XML code. But it's important to keep in mind that these pages are not workable without JAVA code which I've not shown here.

## 4.2 Back-End Design

All logics of a project are performed in the back-end part. Database connection, using database, database maintaining etc. all these database related codes are done in back-end part. In our project we have used JAVA as the programming language for the back-end coding.

For maintaining the back-end part, we have used Firebase database for our project. Firebase provides us user authentication system, real time online server service for any database system.

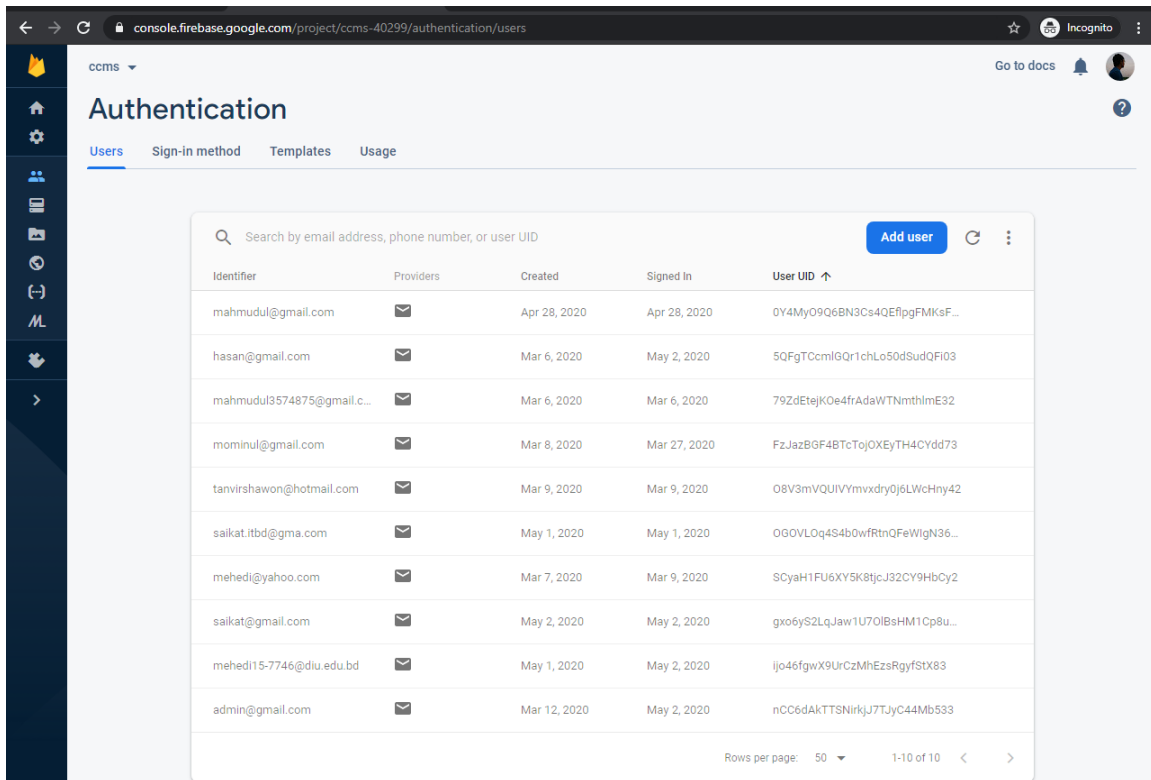


Figure 4.4: Firebase Authentication Page

This is the user authentication method in Firebase. All the users of our app will be added here after registration. A database admin is able to add new user, remove user from this authentication page of firebase.

We've used Firebase storage for keeping the image files of Gallery from our app. Let's have a look on this.

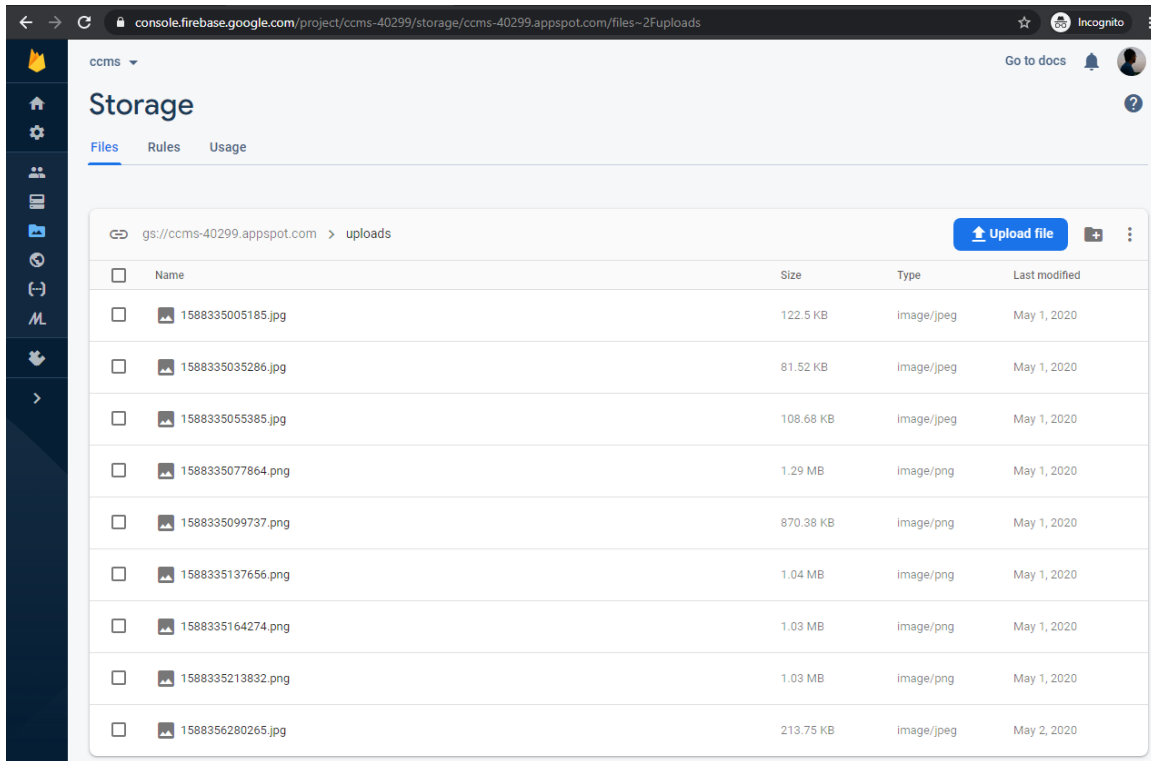


Figure 4.5: Firebase Storage Page

In firebase database system, we need to keep a file to the storage and then it's URL is kept in the Cloud Firestore. That's how we can access the files using their URL.

The main part of our database is Cloud Firestore. Data is stores in the Cloud Firestore in a document and a collection is made with a number of documents. In simple words, there are some collections in a cloud firestore database and a collection has many documents under it. We can keep variety type of data in a document. Our database is shown below:

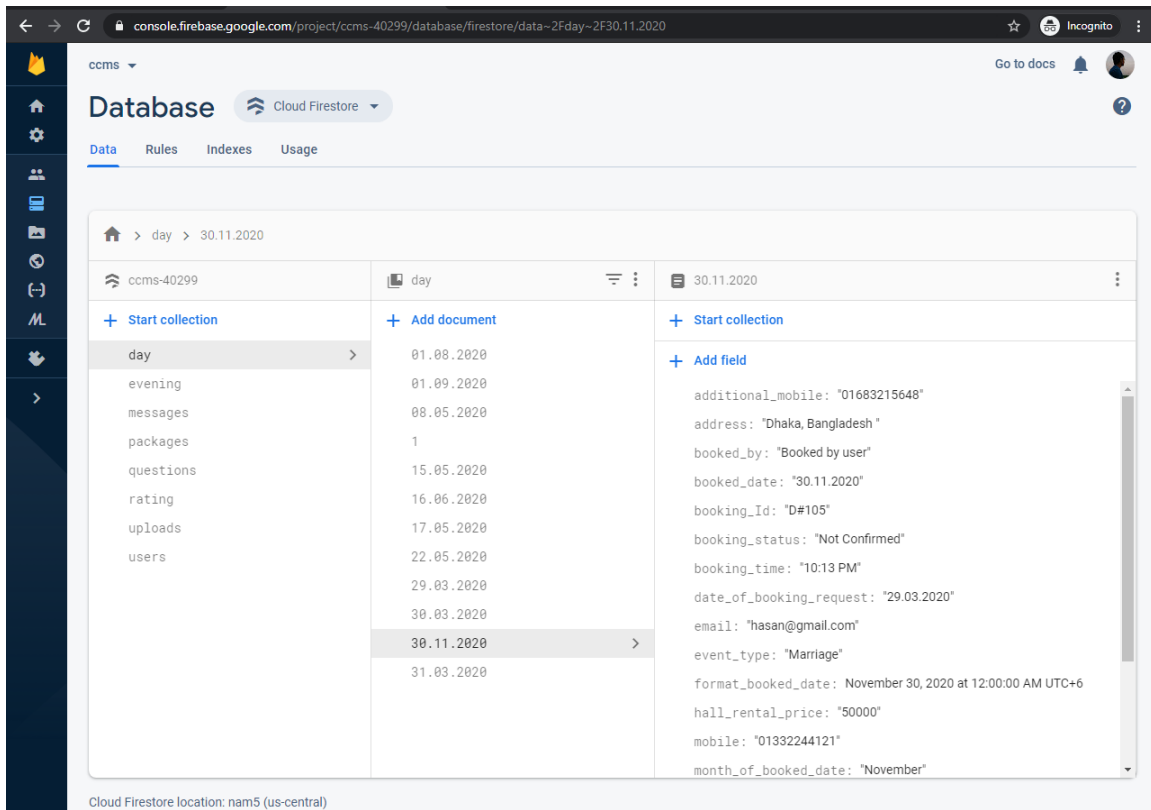


Figure 4.6: Firebase Database (Cloud Firestore) Page

Our database has total 8 collections. There are many documents under these collections. Document is added to a specific collection of the database according to the use of our mobile application.

### 4.3 Interaction Design and UX

#### User Home Page Design:

- Food Packages (Can see the food packages with price offered by the community center)
- Booking (Booking process is done by this page)
- Booking Info. (Booking Information is available here)
- Message (Users can send message. Admin's reply is shown below and users can delete the messaging history)

- Rating (Service Rating can be given through this page only after a service is completed successfully by a user)
- Contact Us (Contact information of the center)
- FAQ (Display frequently asked questions set by admin)
- Sign In, Sign Up
- Menu Bar (Gallery, Booking History, Update Profile, log out, share, rate app, about)

### **Admin Home Page Design:**

- Packages (Update price)
- Check booking (Book a date by admin, confirm and cancel booking, delete booking history)
- Pending Booking Request (Cancel, Confirm, details show; sort by booked date, sort by request date)
- Confirmed Booking Status (Display Upcoming, Current, Previous confirmed booking status list)
- Upload Photo (upload photo to display in gallery)
- Delete photo (Delete photo from gallery)
- Reply message (Users' messages are displayed; can reply with message, can delete without reply)
- Manage FAQ (Add, Update, Delete)
- Update rental info (booking cost update)
- Log out (log out from app)

### **UX**

User Experience is the abbreviation of UX. We have tried to make this app user friendly. We've used this app among some people for trial. We've found everything okay. But as we haven't launched this app publicly, so we don't have enough information on user experience at this moment.

## 4.4 Implementation Requirements

People can use different types of IDE to do an android project. We used Android Studio as it seemed to be more convenient to us. There are a lot of built-in classes in Android Studio. So, it reduces your coding complexities. Design is also very easy in Android Studio. Requirements for the implementation of our project are discussed below:

- Android Studio plays an important role as we've used it as IDE. It is a full package to develop an android application. We've used the current latest version of android Studio 3.6.1. Though we started our project with a old version, we then upgraded it to the latest.
- Android SDK is a kind of software development kit which helps to create mobile applications on android platform. In our app, minSdkVersion is 16 and targetSdkVersion is 29.
- We don't need to worry about server. Because firebase gives us the facility of server. We can use storage, database, authentication feature of firebase very smoothly. Firebase manages server, hosting etc. We just need to connect our project with firebase.
- Emulator, a feature in android studio, is a virtual operating system which can be created in android studio to debug, test and run an application on different versions of android. In AVD manager, we can build virtual devices which can be started by emulator to run our application. We've used virtual device to debug and run our application



## CHAPTER 5

### IMPLEMENTATION AND TESTING

#### 5.1 Implementation of Database

Database is a basic requirement for any management system project. We need to store various data in our database. We keep data and retrieve from database according to our need. As, we are developing a mobile application, so the database needs to be real time. Firebase is a real time database. If we use firebase database, then we need not to worry about the server and hosting. Firebase manages all these.

First of all, we connected our project with firebase following some steps which are available in firebase website. We need to set different dependency in Gradle to get services from firebase. Dependencies like ‘com.google.firebase:firebase-auth:19.2.0’ for authentication, ‘com.google.firebase:firebase-firestore:21.4.1’ for cloud firestore, ‘com.google.firebase:firebase-storage:19.1.1’ for storage are used in our project.



```
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161

mAuth.createUserWithEmailAndPassword(email, pass).addOnCompleteListener(new OnCompleteListener<AuthResult>() {
    @Override
    public void onComplete(@NonNull Task<AuthResult> task) {
        progressBar.setVisibility(View.GONE);

        if (task.isSuccessful()) {
            userID = mAuth.getCurrentUser().getUid();
            DocumentReference documentReference = db.collection( collectionPath: "users").document(userID);
            Map<String, Object> user = new HashMap<>();
            user.put("Mobile", mobile);
            user.put("Email", email);
            user.put("Name", name);
            documentReference.set(user).addOnCompleteListener((task) -> {
                Toast.makeText(getApplicationContext(), text: "Registration Successful", Toast.LENGTH_SHORT).show();
            });
            Intent i = new Intent(getApplicationContext(), MainActivity.class);
            startActivity(i);
            finish();
        }
        else {
            String e = task.getException().getMessage();
            Toast.makeText(getApplicationContext(), text: "Error: " + e, Toast.LENGTH_SHORT).show();
        }
    }
});
```

Figure 5.1: Registration using Firebase Authentication and Cloud Firestore with JAVA code

Figure 5.1 shows JAVA code for user registration using Firebase Authentication. Users are created in Firebase Authentication module and we stored data from the user’s registration form into the cloud firestore.

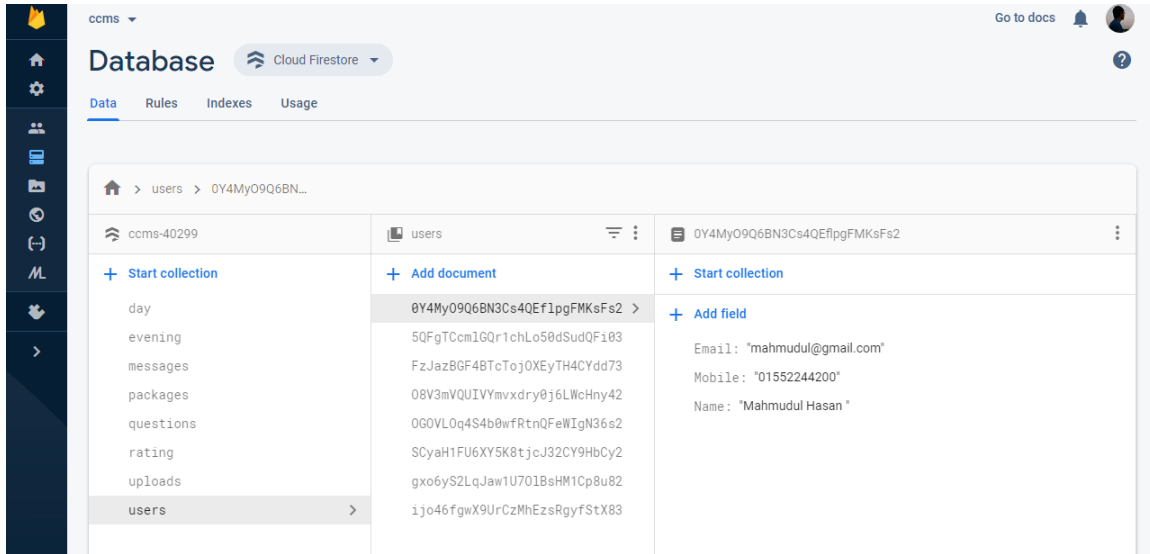


Figure 5.2: Firebase Database

From figure 5.2, we can see the collections in the database. In the collection named “users”, all the users’ information is stored. Others data are also stored in the database writing proper JAVA code.

## 5.2 Implementation of Front-End Design

Making a simple and easily understandable UI is very important for the success of a mobile application. Because if there are complexities to understand the application, people will not be interested to use this. We tried our best to make an attractive UI for the users. There are two types of module in our project; user module and admin module. In other words, we can say that there will be two type of users; common users and admin.

First of all, let’s see two home pages, one is for common users and the other one for admin.

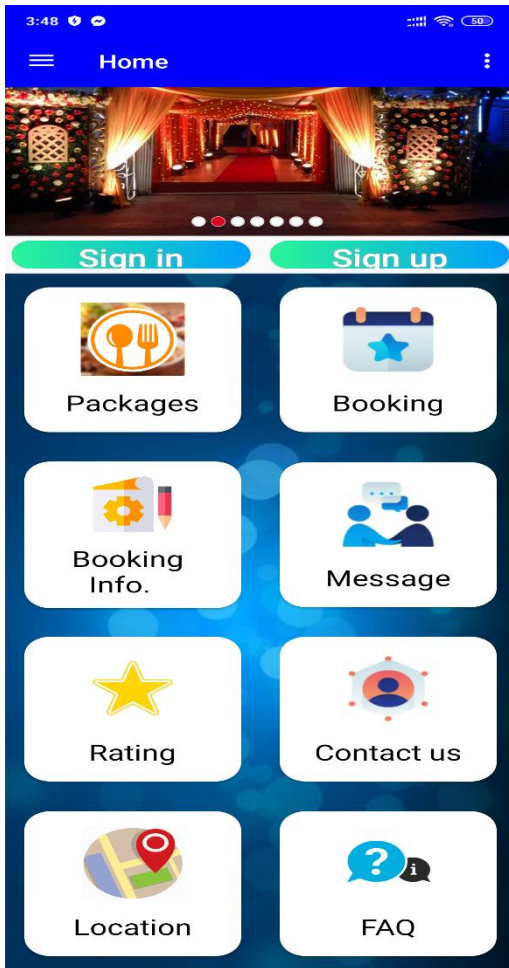


Figure 5.3: User Home Page

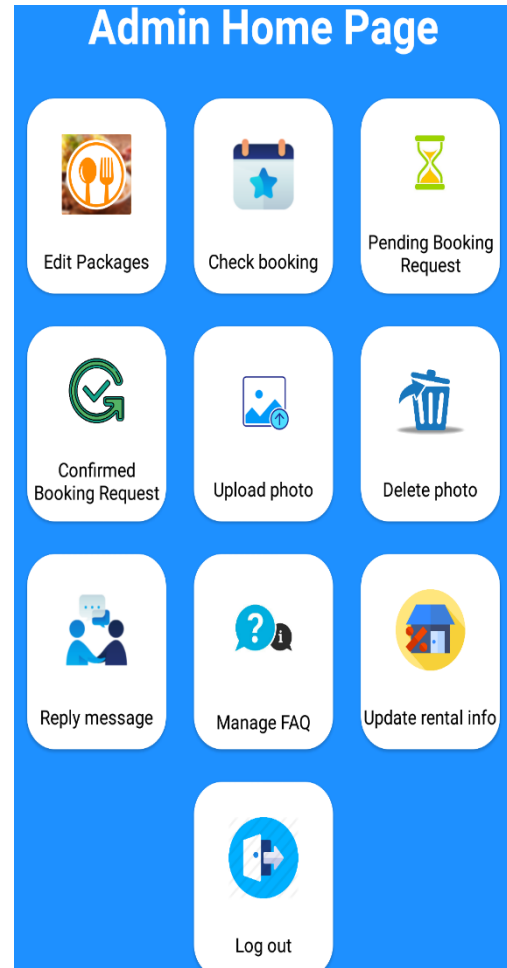


Figure 5.4: Admin Home Page

Figure 5.3 and figure 5.4 show two different home pages. There are many features available for the two different users. First of all, I'll mention the users features, then the admin features.

### 5.2.1 User Module

User module has the features shown in figure 5.3 and also has some other features too which are available after log in, shown in figure 5.7.

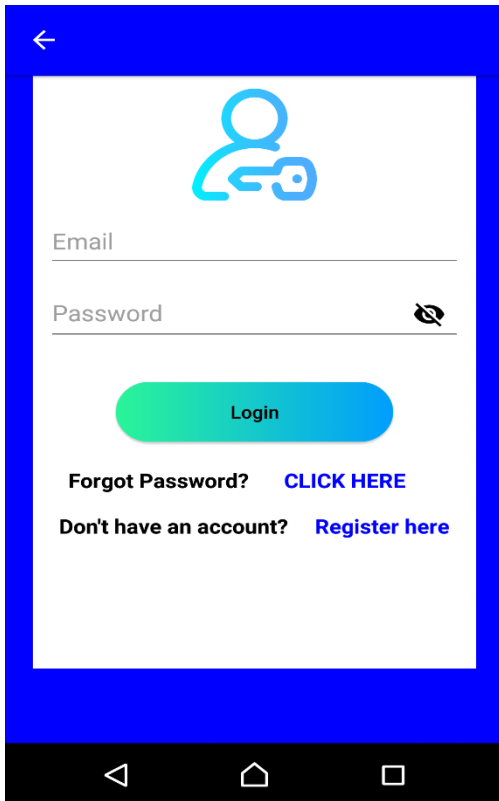


Figure 5.5: User Log In Page

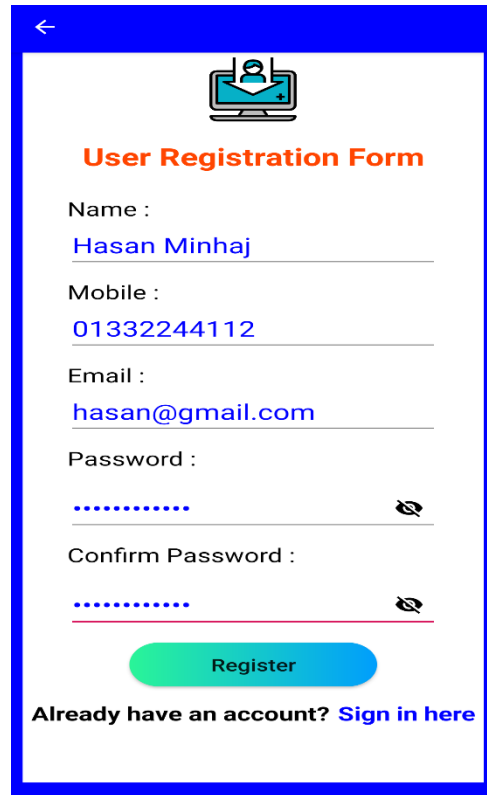


Figure 5.6: User Registration Page

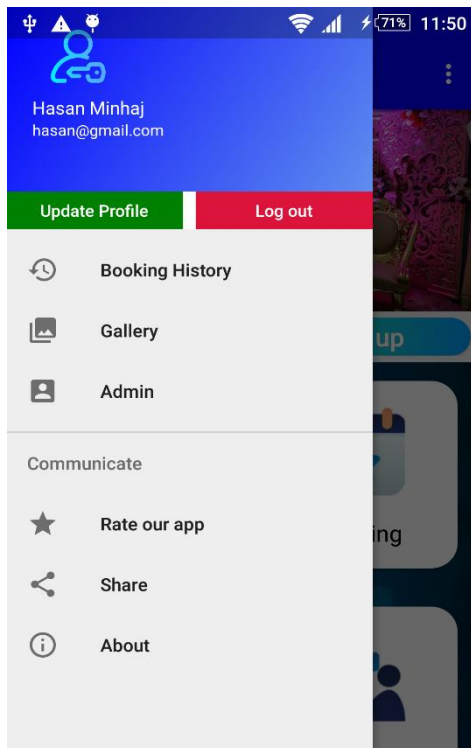


Figure 5.7: After Log In

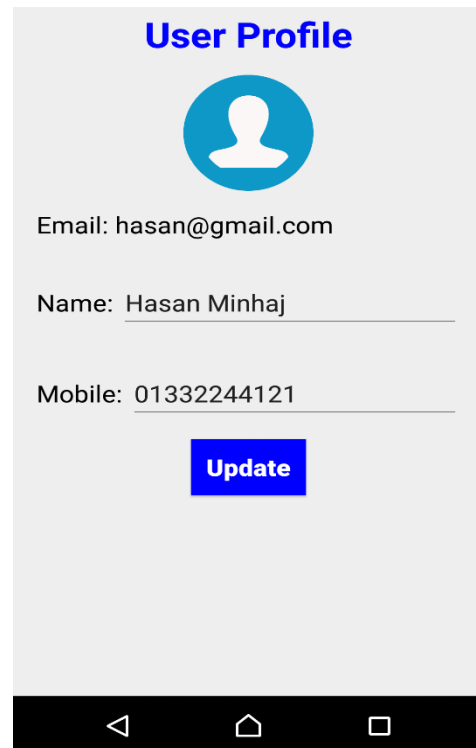


Figure 5.8: User Profile Update

After giving all the valid information, a user can register in this app. Using proper and correct email and password, a user can log in to the system. If any wrong information is provided, log in is denied with toasting a warning message. After successful log in, user's information is be shown in the navigation bar and two new options; update profile, log out become visible. Choosing 'update profile' option a user can update his information shown in figure 5.8.



Figure 5.9: Gallery



Figure 5.10: Food Packages

Figure 5.9 shows the gallery of our app. The uploaded photo with name by admin of the community center is shown in this option. Figure 5.10 is the offered food packages by community center with price. Users can get an idea about the food price.

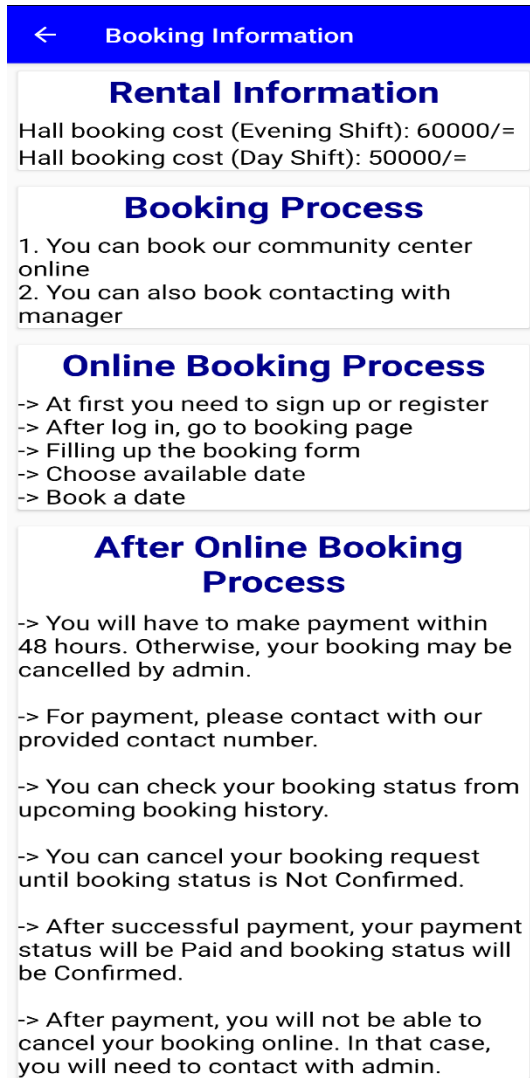


Figure 5.11: Booking Information



Figure 5.12: FAQ

Figure 5.11 shows the booking information including booking guideline. It's in the Booking Info. Option of our option. FAQ option of our app is shown in the figure 5.12.

←

**Slot is already booked, choose another date**

Name: Mehedi Hasan Saikat

Email: mehedi15-7746@diu.edu.bd

Mobile: 01961264463

Additional Mobile: 01961264463

Address: Dhaka

Event Type: Marriage

Number of guests: 300

Select Event Time  Evening  Day

Select date :

Today's date: 01.05.2020

**BOOK**

Figure 5.13: Booking Page

← Booking History

**D#108**

**Booked Date:** 16.06.2020

**Shift:** Day

**Booking Status:** Confirmed

**D#102**

**Select Shift**

**EVENING** **DAY**

**D#104**

**Booked Date:** 01.09.2020

**Shift:** Day

**Booking Status:** Confirmed

Upcoming Current Booking Previous History

Figure 5.14: Booking History

Figure 5.13 shows the page where a user can book a date checking vacancy. After book a date, a user can check details in his upcoming booking history (figure 5.14) option. Users can check their upcoming, current and previous booking details from Booking History which is showed in figure 5.14.



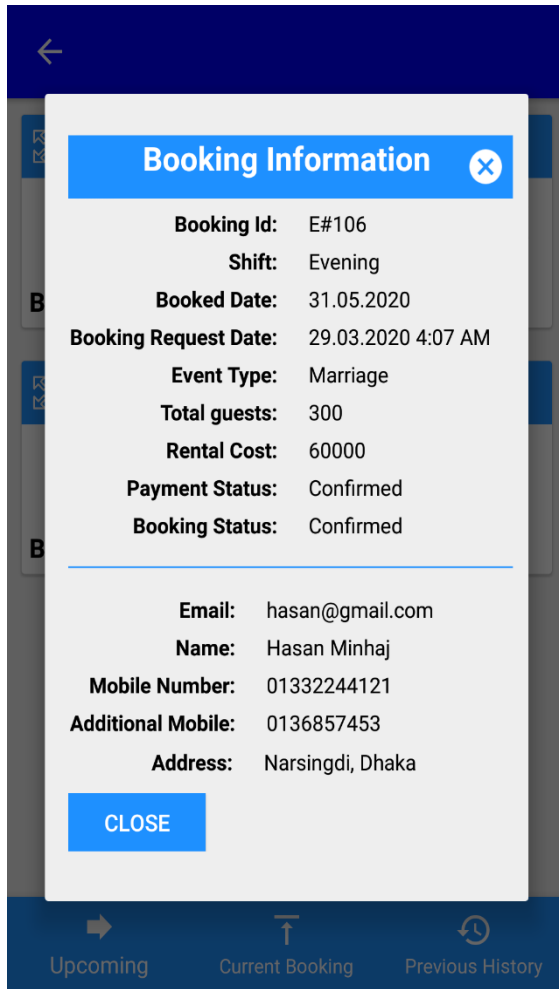


Figure 5.15: Confirmed Booking Details

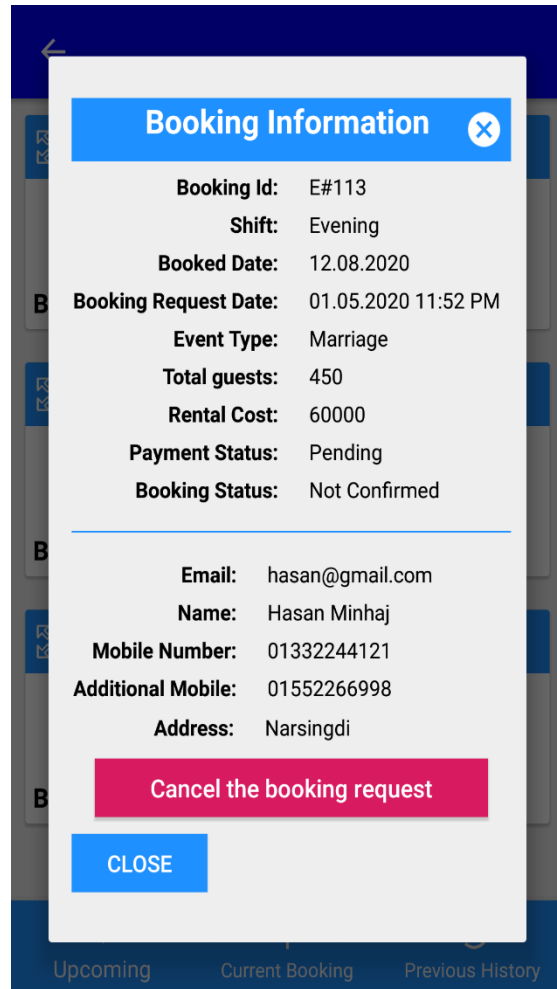


Figure 5.16: Pending Booking Details

If a user touch or press the booking icons, booking detail is showed like figure 5.15 or figure 5.16. Users can cancel their booking request until it is not confirmed (figure 5.16). They can't cancel booking if it's confirmed. But an admin can do it.



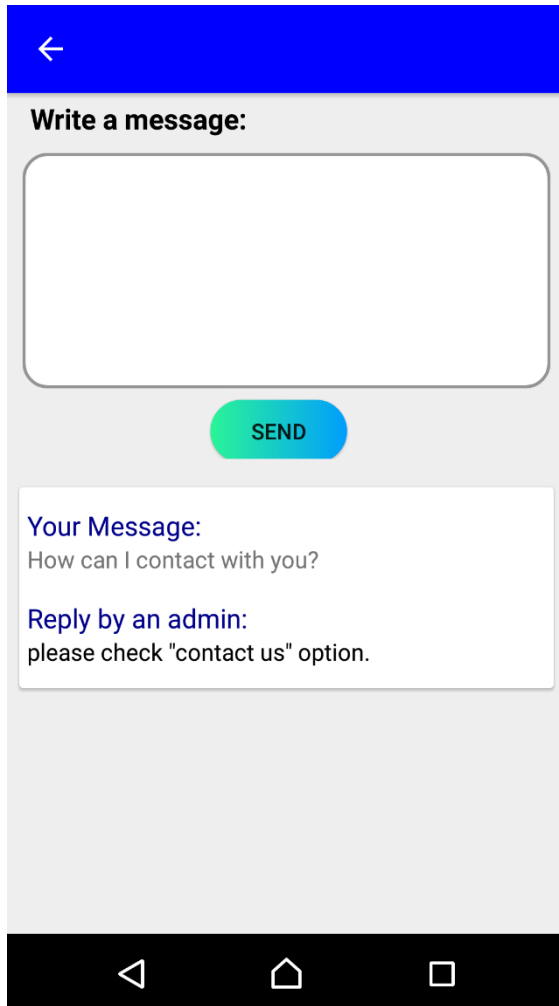


Figure 5.17: Message

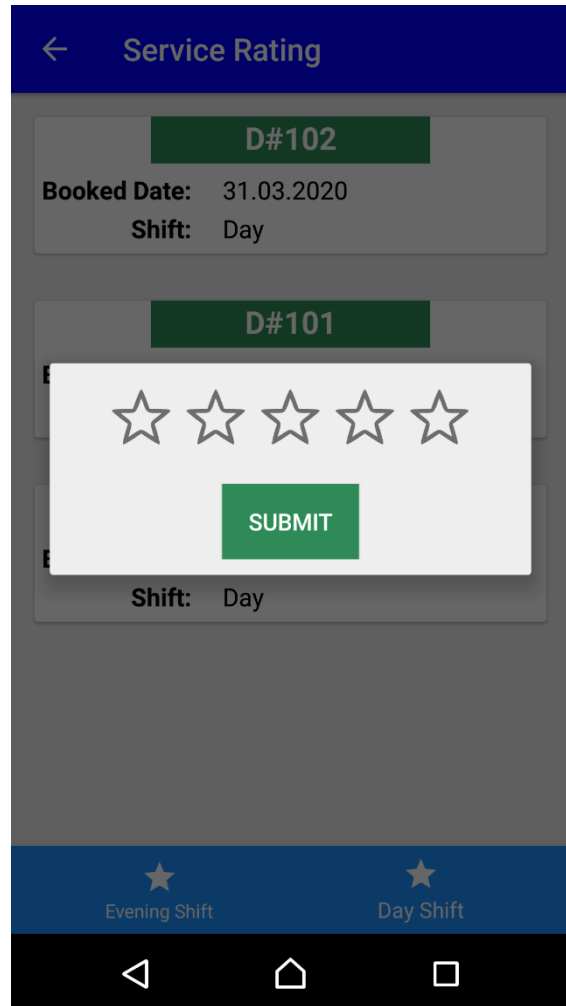


Figure 5.18: Service Rating

Users can give rating after taking a service from the community center shown in figure 5.18. In this page, list of all services taken by the logged in user is showed. Rating can be given by pressing any booking id.

Figure 5.18 shows the message option. A user can send message. If an admin replies his message, it will appear below the message box one by one. Users get a delete option for deleting any message just long pressing it.

## 5.2.2 Admin Module

Admin Module has features showed in figure 5.4 Some pages are discussed below.

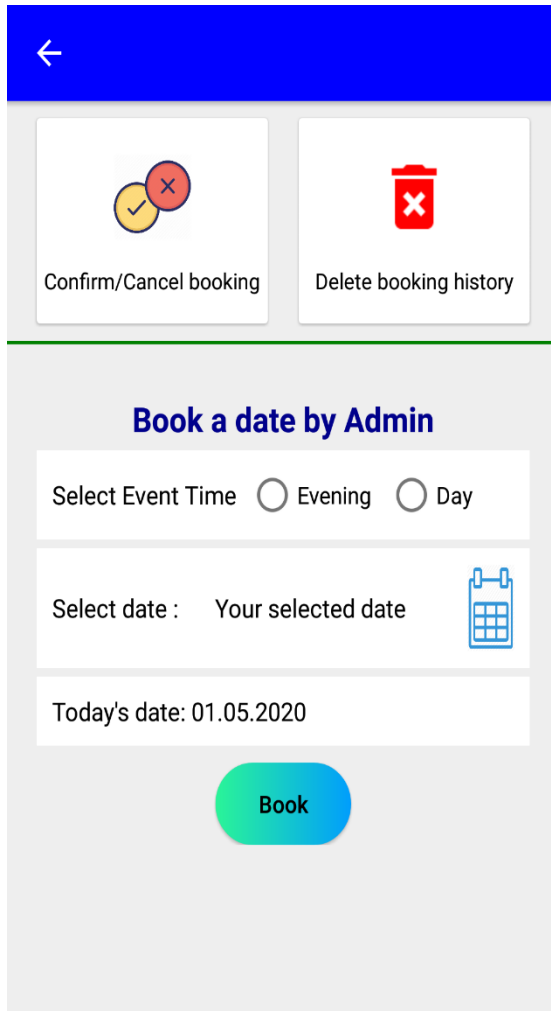


Figure 5.19: Admin Check Booking

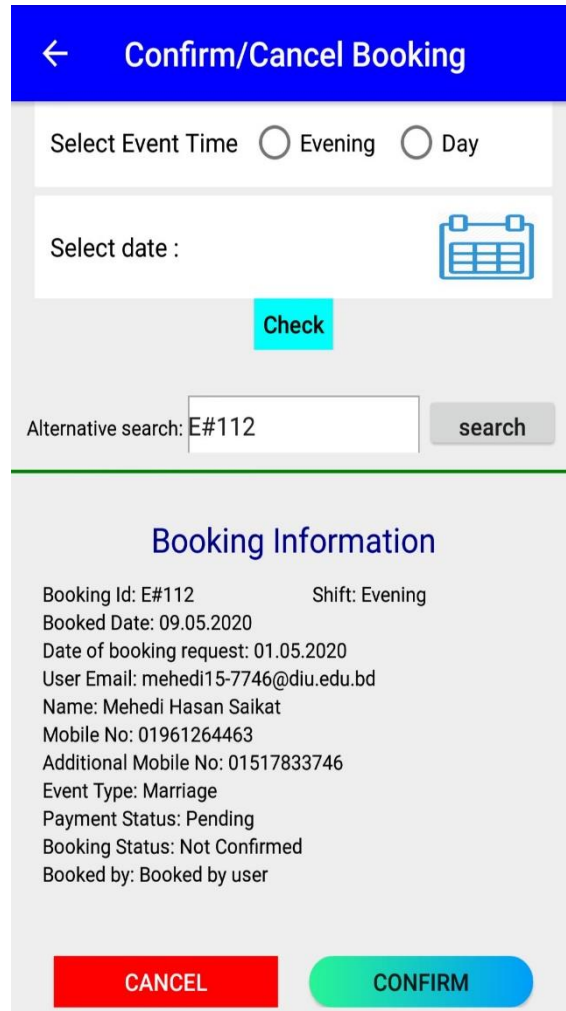


Figure 5.20: Admin Confirm/Cancel Booking

There are three options, Confirm/Cancel booking (figure 5.20), Delete booking history and book a date, included in Check Booking showed in figure 5.19. Admin can see booking information either selecting event time with date or choosing alternative search. Any kind of booking can be cancelled here by an admin. I am not attaching the delete booking history page here. But admin can delete any booking history from there, there are two options delete single history and delete by month.

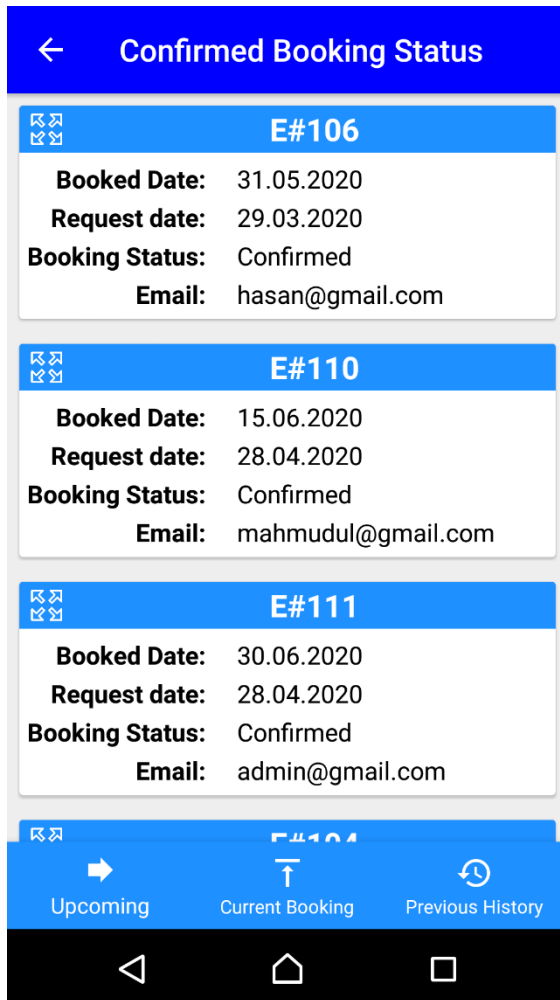


Figure 5:21: Admin Confirmed Booking list

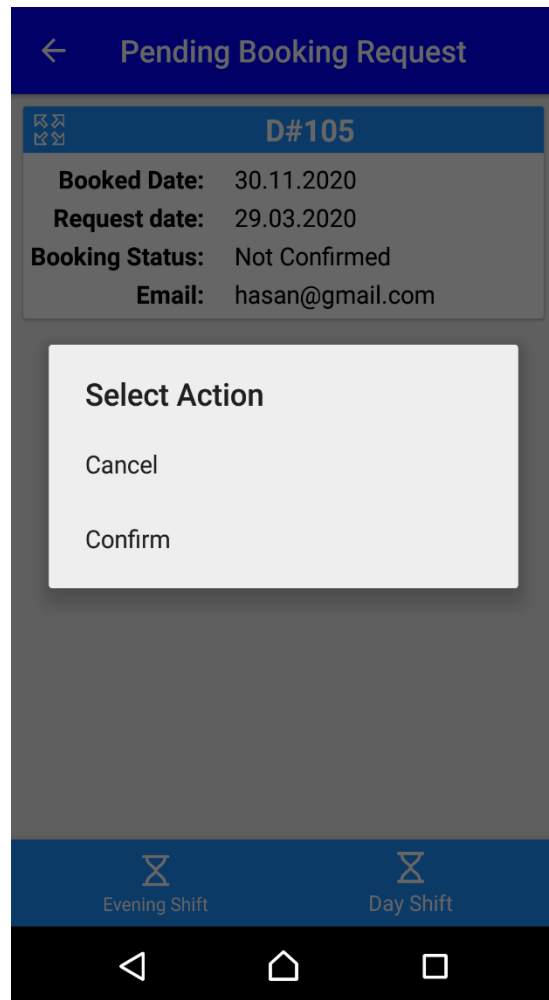


Figure 5:22: Admin Pending Booking list

Admin can check the confirmed booking list showed in figure 5.21. Figure 5.22 shows the pending booking request page. Here, an admin can check the pending booking request list sorted by booked date or request date. Admin is able to cancel or confirm any booking request from here also.



Figure 5.23: Admin Edit Packages

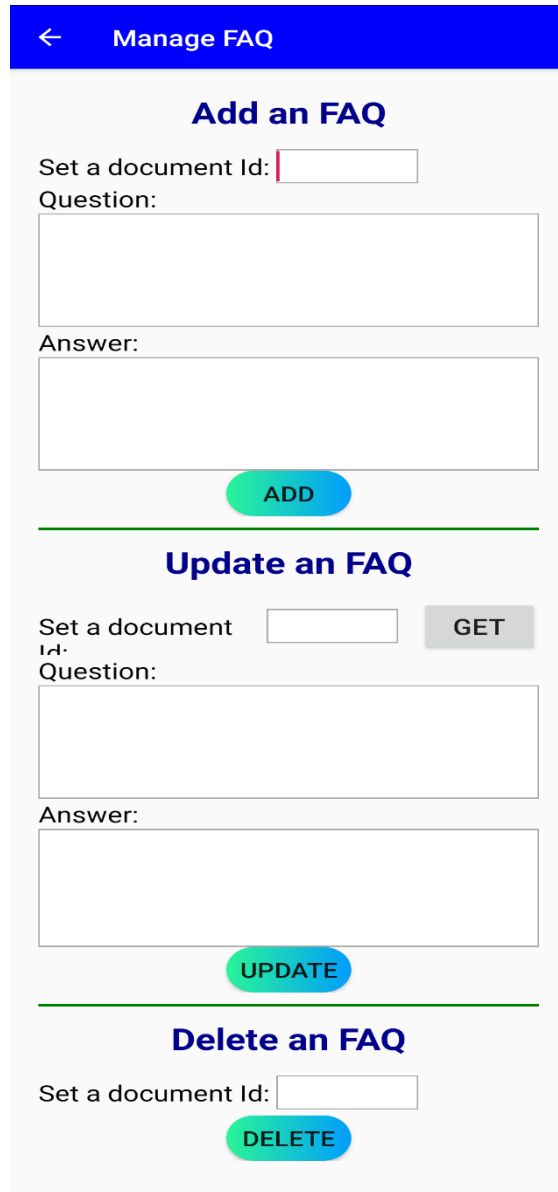


Figure 5.24: Admin Manage FAQ

Admin can edit any food package; this page is showed in figure 5.23. Admin can add an FAQ, update an FAQ getting any particular selecting its document id, delete an FAQ; this page is showed in figure 5.24.

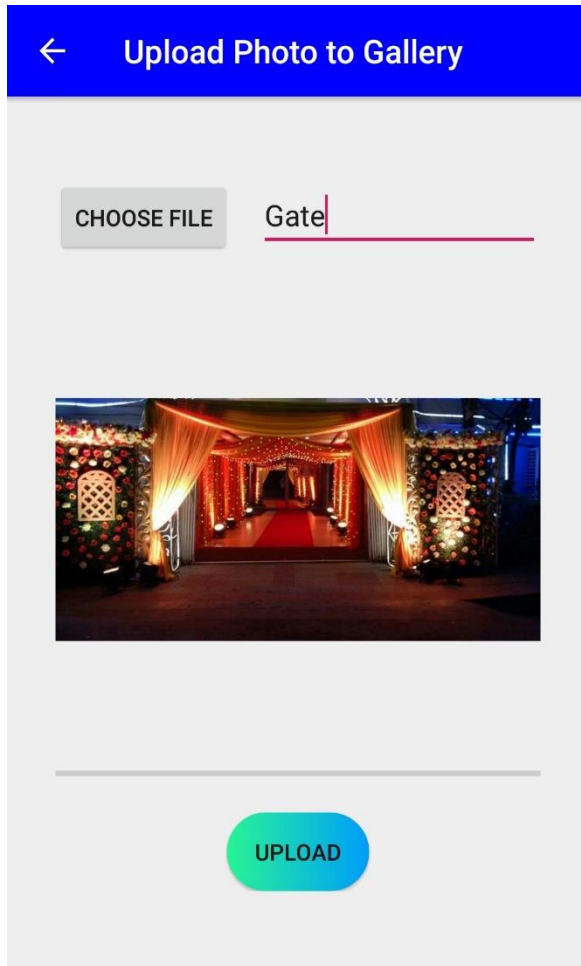


Figure 5.25: Admin Photo Upload

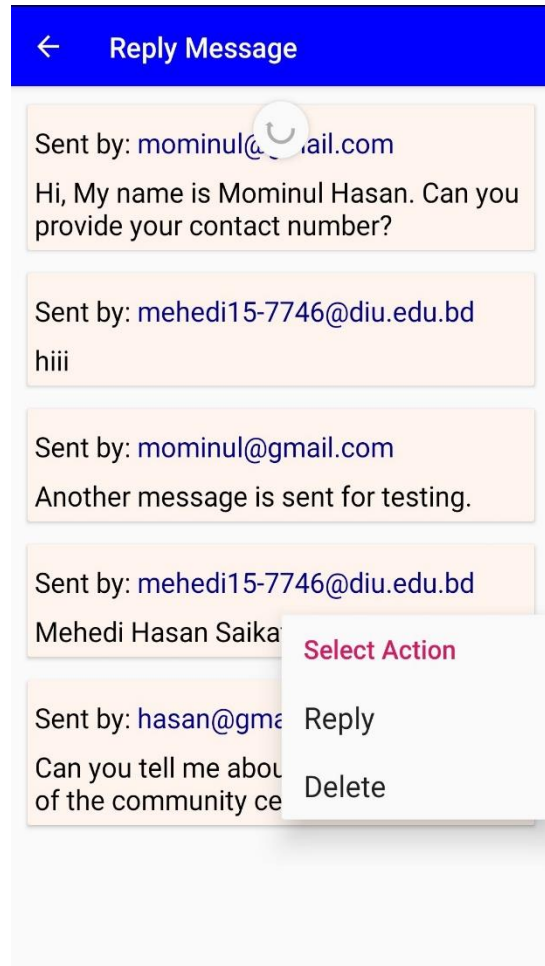


Figure 5.26: Admin Reply Message

Figure 5.25 shows the page where an admin can upload a photo to gallery. Figure 5.26 shows message reply option. Admin can reply a message and delete without any reply. Once a message is replied or deleted, it will disappear from the list.

There are two more pages 'delete photo' and 'update rental info' for admin which are not attached in this report. From there, admin is able to delete any photo from the gallery and another one is for updating rent for the community center.

### 5.3 Implementation of Interaction

Making an application interactive is very important for the success of any project. People like to use a user-friendly application. If an application is not interactive, it may not be attractive for the users. Users can easily use this app without any complexity We've tried our best to make a user friendly and interactive application.

### 5.4 Testing Implementation

It's a process of testing an application in many ways. We tested our whole project. As we have two modules; user module and admin module, so we've tested the both modules.

Table 5.1: Testing Information and Date

Test Case	Test Input	Expected Outcome	Obtained Outcome	Result	Tested on
1.User Registration	Give all required information correctly	Shows registration successful	Registration Successful	Passed	02/02/2020
2.User Registration	Click register button without entering any data	Shows error message	Registration denied with error message "Please fill all fields"	Passed	02/02/2020
3.User Sign In	Try to log in with incorrect password	Shows warning message	Log In denied with error message	Passed	27/02/2020

3.User Sign In	Enter correct information	Shows the success message	Log In with message “Log In Successfully”	Passed	27/02/2020
4.Update user profile	No empty fields	Updates profile information	Updated and showed “Successfully Updated”	Passed	01/03/2020
5.Booking by user	Enter data remaining without address	Shows warning message for address	Showed warning as “Enter your address”	Passed	01/04/2020
5.Booking by user	Enter all correct data	Successful booking	Booked with message “Booking Successful”	Passed	05/04/2020
6.Booking by user	Enter a date already booked	No vacancy message	Showed warning that already booked	Passed	05/04/2020
7.Sending message by user	Press send button writing a message	Successfully send the message	Successfully sent	Passed	07/04/2020
8.Admin Log In	Incorrect information	Access deny	Access denied	Passed	07/04/2020

9.Admin Log In	Enter all correct information	Permit the access	Accessed showing Admin Log In Successful	Passed	08/04/2020
10.Edit packages	No empty fields	Information update	Updated with message	Passed	08/04/2020
11.Booking by admin	Select shift and available date	Booking successful	Booked with message "Booking Successful"	Passed	08/04/2020
12.Confirm booking by admin	Press confirm and ok selecting a booking id	Shows booking confirmation	Showed Booking Confirmed message	Passed	03/05/2020
12.Cancel booking by admin	Press cancel and ok selecting a booking id	Booking should be cancelled	Showed booking cancelled	Passed	03/05/2020
13.Photo upload to gallery by admin	Select image with a name	Photo upload successful	Showed successfully uploaded	Passed	05/05/2020
14.Message reply by admin	Long press any message details and write a message	Send message and update the list	Showed successfully replied and removed this from list	Passed	05/05/2020



15.Delete photo from gallery by admin	Select any photo and press delete option	Photo should be deleted and update gallery	Showed deleted and remove this from gallery	Passed	05/05/2020
16.Add a FAQ by admin	Write a question, answer and set an id	Add FAQ	Showed successfully added	Passed	10/05/2020
17.Log Out	Click log out button	Log out the account	Log out successful	Passed	10/05/2020

## 5.5 Testing Result and Reports

Testing is essential to find out bugs in an application. We've fixed almost all the bugs we got. After fixing bugs, we tested our application again and then got the testing results which are showed in table 5.1.

We have tested our app with more queries like remaining empty fields, invalid information. In the table, we have showed some of the queries we applied but not all of them. Our app was successful for all the quires. We have installed our app in many android devices and it works perfectly in every device.

## **CHAPTER 6**

### **CONCLUSION AND FUTURE SCOPE**

#### **6.1 Discussion and Conclusion**

This project will help to run a community center smoothly. Users will be benefitted through this online based booking system. Community center's owner will also be benefitted and this system will reduce his complexities to run the business. More upgrade is possible in future with new features according to the center's requirements.

#### **6.2 Scope for Further Developments**

Development of any project has no end. We have also plan to develop it more in future. Some future plans for this project are given below:

- We will also build a web version of this project
- Online payment method for the booking process will be introduced.
- Notification option can be added for booking updates.
- Booking information download option will be added.
- Some other features like food order and choosing event management can be added in the same app.
- Admin portal and general users' portal can be divided into two different app with same database for more security purpose.

## REFERENCES

- [1] Parjatanbd website, <http://parjatanbd.com/>, last accessed on 05 May, 2020
- [2] Mediabangladesh website, <http://mediabangladesh.net/>, last accessed on 05 May, 2020
- [3] Biyebazaar website, <http://biyebazaar.com/>, last accessed on 05 May, 2020
- [4] Thebride website, <http://thebride.com/>, last accessed on 05 May, 2020
- [5] Communitycenterbd website, <http://communitycenterbd.com>, last accessed on 05 May,2020
- [6] Senamalancha website, <http://senamalancha.com/>, last accessed on 05 May, 2020.
- [7] Use case diagram tutorial, <http://creately.com/blog/diagrams/use-case-diagram-tutorial/amp/>, last accessed on 14 May, 2020
- [8] Logical Data Model, <http://www.sciencedirect.com/topics/computer-science/logical-data-model>, last accessed on 14 May, 2020
- [9] Android Emulator, <https://developer.android.com/studio/releases/emulator>, last accessed on 10 September, 2019.
- [10] Cloud Firestore, <https://firebase.google.com/docs/firestore/quickstart>, last accessed on 07 May, 2020.
- [11] Android SDK, <https://developer.android.com/studio/releases/platforms>, last accessed on 12 August, 2019.
- [12] How to add Firebase to a project, <https://firebase.google.com/docs/android/setup>, last accessed on 15 October, 2019.
- [13] Image Slider, <https://demonuts.com/image-slider-slideshow/>, last accessed on 29 January, 2020
- [14] Why Firebase is the best mobile backend as a service?, <https://www.tristatetechnology.com/blog/firebase-backend-mobile-app/>, last accessed on 05 July, 2019
- [15] Connect firebase to application, <https://console.firebase.google.com/>, last accesses on 06 May, 2020

## **APPENDIX**

### **Appendix A: Project Reflection**

We began our journey to build this project from Fall,2018. For better understanding, we did a lot of research on community centers in Bangladesh. We collected information from many websites. Then we took decision to use Firebase as our database. For this, we needed to study a lot on this topic as Cloud Firestore was a very new database system for us. Firebase has two types of database; one is Real Time Database and the other one is Cloud Firestore. We chose Cloud Firestore as it is the latest and recommended edition by Firebase. We've tried to build a user-friendly community center management system application specially for the booking purpose. It can also be a communication medium between users and community centers' officials. I hope this application will be very helpful for the users. We will keep upgrading our system adding new facilities and features.

# PLAGIARISM

## Plagiarism Report

community center			
ORIGINALITY REPORT			
6%	5%	1%	%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS
PRIMARY SOURCES			
1	<a href="https://dspace.daffodilvarsity.edu.bd:8080">dspace.daffodilvarsity.edu.bd:8080</a> Internet Source	4%	
2	<a href="http://www.slideshare.net">www.slideshare.net</a> Internet Source	1%	
3	<a href="https://firebase.google.com">firebase.google.com</a> Internet Source	<1%	
4	Sugato Bagchi, Xue Bai, Jayant Kalagnanam. "Data Quality Management using Business Process Modeling", 2006 IEEE International Conference on Services Computing (SCC'06), 2006 Publication	<1%	
5	<a href="http://www.fatihun.edu.tr">www.fatihun.edu.tr</a> Internet Source	<1%	
6	Florian Heidenreich, Pablo Sánchez, João Santos, Steffen Zschaler et al. "Chapter 3 Relating Feature Models to Other Models of a Software Product Line", Springer Science and Business Media LLC, 2010 Publication	<1%	