COUNSEL-PRO:AN ANDROID BASED MOBILE APPLICATION FOR BOOKING & INTERACTING WITH TEACHER FOR COUNSELLING TIME

SUBMITTED BY

Masum Billah
ID: 172-15-9821

AND

Abrar Alvi Nirab
ID: 172-15-9800

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

Supervised By

Ms. Nishat Sultana
Lecturer
Department of CSE
Daffodil International University

DAFFODIL INTERNATIONAL UNIVERSITY

DHAKA, BANGLADESH APRIL, 2021

APPROVAL

This Project titled "COUNSEL-PRO:AN ANDROID BASED MOBILE APPLICATION FOR BOOKING & INTERACTING WITH TEACHER FOR COUNSELLING TIME", submitted by Masum Billah, ID No: 172-15-9821; Abrar Alvi Nirab, ID No: 172-15-9800; 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 1 June 2021.

BOARD OF EXAMINERS

Dr. Touhid Bhuiyan **Chairman**

Professor and Head

Department of Computer Science and Engineering

Faculty of Science & Information Technology

Daffodil International University

THabib

Md. Tarek Habib **Internal Examiner**

Assistant Professor

Department of CSE

Faculty of Science & Information Technology

Daffodil International University

Saiful Islam **Internal Examiner**

Senior Lecturer

Department of Computer Science and Engineering

Faculty of Science & Information Technology

Daffodil International University

Dr. Md Arshad Ali **External Examiner**

Associate Professor

Department of Computer Science and Engineering

Hajee Mohammad Danesh Science and Technology University

DECLARATION

We hereby declare that, this project has been done by us under the supervision of **Ms. Nishat Sultana, Lecturer, 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:

Ms. Nishat Sultana
Lecturer
Department of CSE
Daffodil International University

Submitted by:

masum Billah

Masum Billah
ID: 172-15-9821
Department of CSE
Daffodil International University

Nirab

Abrar Alvi Nirab
ID: 172-15-9800
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 **Ms. Nishat Sultana**, **Lecturer**, Department of CSE Daffodil International University, Dhaka. Deep Knowledge & keen interest of our supervisor in the field of "*Field name*" to carry out this project. His 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 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

The title of our project is “**AN ANDROID BASED MOBILE APPLICATION FOR BOOKING & INTERACTING WITH TEACHER FOR COUNSELLING TIME**”. In this project we focused to take appointment from faculty member of an educational institution like University or College. This appointment is taken by students. Both student and faculty member use only one application. In this project both student and faculty members of an institution have to register in this system to communicate with each other. Student can search their specific faculty member by searching their name. And can sent appointment with the proper reason. A student also can get the details information of this faculty member like (email, phone, office location, designation, appointment schedule, department etc.). After sent request a notification received by faculty member and can see the details (student name, department, ID, email, reason of request etc.) of this appointment request. Then faculty member can accept or deny of this request with a short message. After accept/deny student notify this response. Both student and faculty member can see their appointment in their dashboard of the application. The faculty member can take note about his/her giving appointment in “Note Section”. Which will save in this system.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1: Introduction

1.1 Introduction

Appointment system is very common a topic in our daily life. But in modern era this appointment system is executed by modern technology like phone call, sending email or any other system. We can see that there are many appointment systems in our country like doctor-patient appointment. In some case they make some application for getting appointment. But for educational purpose there is no application system in which a student can get appointment from faculty. Generally, there a lots amount of student has in an institution. And every department of an institution contain a big number of students. In our project we make an android application in which a student can get direct appointment from teacher according to his/her schedule. This is one to one communication system in which student can directly reach to teacher. In this application student can also get all information of the faculty like phone, email, office, department etc. But before both student and faculty have to register in this system. It helps to both student and faculty to maintain appointment with smart way and hassle free.

1.2 Motivation

To communicate with faculties in an institution like university, college is very important for a student. We keep communication via email, phone, message. But sometime it is not possible to reach the faculty by this way. Sometime a student doesn't know when a faculty is available in his/her office. Because their counseling schedule updating every semester or every month. When a student visit faculty office then doesn't meet with faculty because they don't know about their available time. From our real-life observation, we think that if there is any platform which a student can get every information of a faculty such as: email, phone, counseling schedule, office location and other information and student can make appointment to faculty for counseling. And faculty can reply of this appointment. And this platform helps both student and faculty to communicate in one platform. From this kind of think we take plan to make this project.

1.3 Objective

- ➢ To make the communication between student and faculty is easier.
- ➢ In one application student get every information of faculty.
- ➢ Get real time appointment schedule update of faculty.21

1.4 Expected Outcomes

The main expected outcome will make a user-friendly application in which student can communicate with faculty in easy way.

1.5 Project Management and Finance

In our projects there are two types of cost. One is database cost and another is developer cost. As we use Google firebase database so it has some cost variation. We can use this database free, but when user will increase and the amount of data will increase then need to purchased to its plan. The developer cost should be 30000 – 35000k to build this project. And the maintain cost based on developer. We include a chart which are describe about firebase plan price with free and without free based on user and amount of data. We use firebase authentication and Cloud Firestore service. The cost is need when the free limit is fulfilled.

TABLE 1.1: Free and cost plan of firebase database

Service Name	Free Plan	Cost Plan
Phone Authentication	10k/month	$0.06/verification
Other Authentication services(Email, Facebook etc.)	No limit	No Cost
Stored data	1 GiB total	$0.18/GiB
Document writes	20K/day	$0.18/100K
Document reads	50K/day	$0.06/100K
Document deletes	20K/day	$0.02/100K

1.6 Report Layout

Chapter 1: Introduction

In this chapter we will describe about out project introduction, project motivation, objective, expected outcomes, project managements and finance and the report layout.

Chapter 2: Background

In this chapter we will discuss about some project with is some similar with our project. And we try to compare our project with other project and try to say why our project is better then others.

Chapter 3: Requirement Specification

In this chapter we will discuss about business process model of our project, Requirement collection and the use case model diagram.

Chapter 4: Design Specification

In this chapter we will discuss about frond end design and back-end design of our project. We will show the all-design screen and back-end file structure functionality. And also discuss about the implement requirement.

Chapter 5: Implementation and Testing

In this chapter we will discuss about database design and implementation, how to implement the front-end design and discuss the different types of testing result.

Chapter 6: Conclusion and Future Scope

In this chapter we will discuss the final result, limitation of our application. And also informed about the future scope of our project.

Chapter 2: Background

2.1 Introduction

In this part we represent some work in this field or summarize some application of appointment or teacher- student communication way in our country or another country.

2.2 Related Works

[1] The authors implement a web-based application which is made with .NET language for appointment management system in which help the student and teachers to arrange meeting in an efficient and appropriate way in university or college. [2] A web and mobile application-based system where tutor can make appointment with GPS system. Is the tutor execute their job or not, they can track by this system. [3] A web-based system in which student and teacher can introduce with both, and they can get appointment from teacher. In this system student and teacher can drop message for communication.

2.3 Comparative Analysis

Here the first work is web-based application in which teacher and student arrange meeting but, in our project, we are going to build a mobile based application which is more user friendly than web application.

The second is a GPS tracking system in which tutor can book their appointment and it can track of the tutor location to check is the done their job properly. But in our system the teacher adds his/her office location/room number at the time of registration. So that we not need to add GPS system.

The third is some similar with our project but there is some different with our project. In our teacher can take note of student appointment and teacher get the real time appointment notification and when teacher response the notification the student also gets the real time notification.

2.4 Scope of the Problem

Most of the system which we describe in previous section is web based. But a mobile based application is more user friendly then a web-based application. In our application we used firebase data base in which give the real time update and user (teacher and student) will receive notification. The other projects have some complexity but in our project is very easy to use and it is user friendly. Another thing is that user receive notification when the application is closed.

2.5 Challenges

At the time of developing this project we faced some problem. There are two types of users in our application. To keep the track of user type we faced problem. After finishing the project, we tried to fix all bugs of this project. We used some advance features of android development area like MVVM coding architecture pattern, Dagger for dependency Injection, Android X version and other some library. Which was more challenging for us.

Chapter 3: Requirement Specification

3.1 Business Process Modeling

Figure is known as data flow diagram. DataStream chart used in a dare to address the data stream of an information. DFD refine the procedure of information stream in the business data framework from contribution to record stockpiling. Data Flow Diagram (DFD) is divided into two part. They are sensible and physical. The sensible DFD describe all information stream through a framework and find out the usefulness of business. The physical DFD illustrate the usage of legitimate information stream.

We combined our DFD and business process model, which is given below:

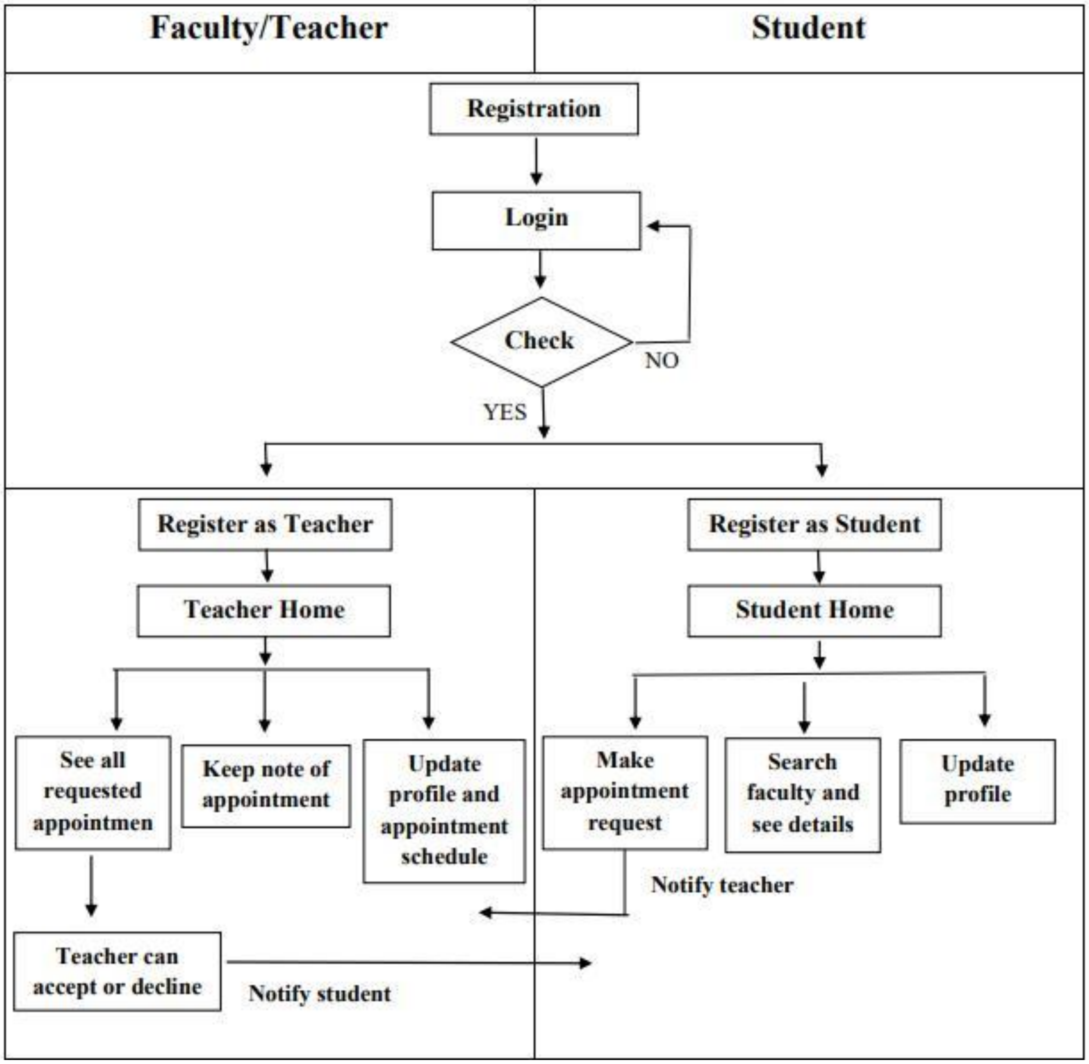


Figure 3.1: Data Flow Diagram

3.2 Requirement Collection and Analysis

This application is android based application so it has some Requirement. They are:

- **Hardware Configuration:**
  - Android smart phone for both teacher and student.
  - A computer for make this software.

- **Software Configuration:**
  - Operating System: Android (API level 21 or higher)
  - Tools: Android Studio IDE (Version: 4.0.1)
  - Database: Firebase storage, Firebase Firestore
  - Server: Firebase

- **Features:**
  - Email & password registration
  - User login
  - Separate registration teacher & student with details information
  - See all teacher list
  - Student make appointment with reason
  - Teacher can accept or decline with a reply
  - Teacher make note of appointment
  - Both teacher and student can see appointment list in home page
  - Update profile
  - Search
  - Get notification (both teacher & student)
  - Logout

3.3 Use Case Modeling and Description

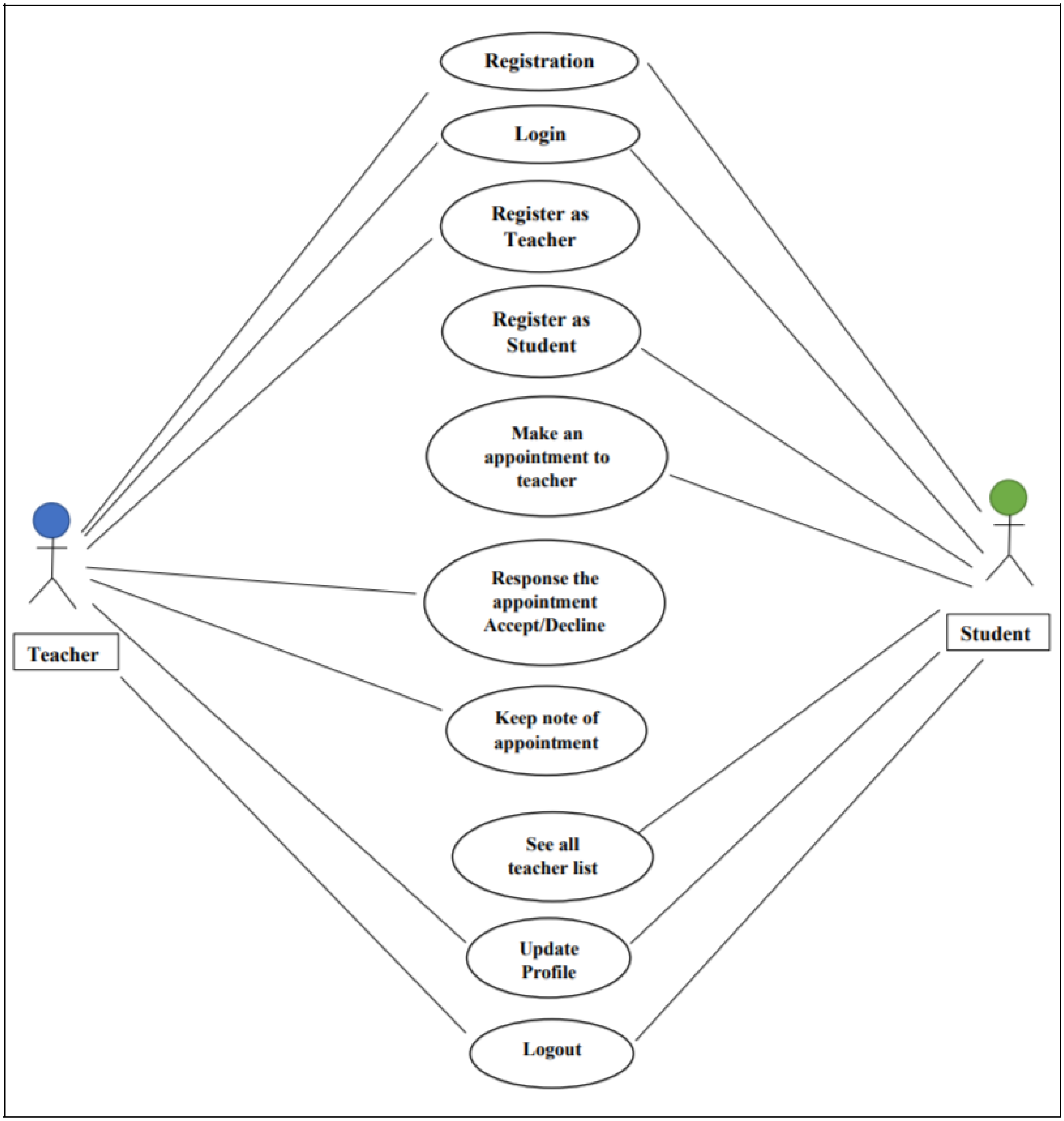


Figure 3.2: Use case diagram

In this use case model diagram, there are two actors. One is teacher another is student. Both teacher and student have to register with email and password. After register they receive a confirmation email in their mail address. After confirm they have to login. For first time both teacher and student must register again separate like as a student and as a teacher. Student can see all teacher list with details information who are already registered in this system. Then student cam make appointment with the reason of appointment. After make appointment teacher receive a notification message. The student can see all appointment which he/she made in their home page with status like pending, accept, decline. Same as teacher can see all request list of appointment with the details of student. When teacher make any response against the appointment like accept or decline with a reply message then student also receive a notification from teacher. The teacher can keep note of appointment in note section. Both teacher and student can update their profile and logout from this system.

Chapter 4: Design Specification

4.1 Front-end Design

- ### 4.1.1 Splash Screen:

  This is the splash screen of our application. It shows at the starting time of our application.

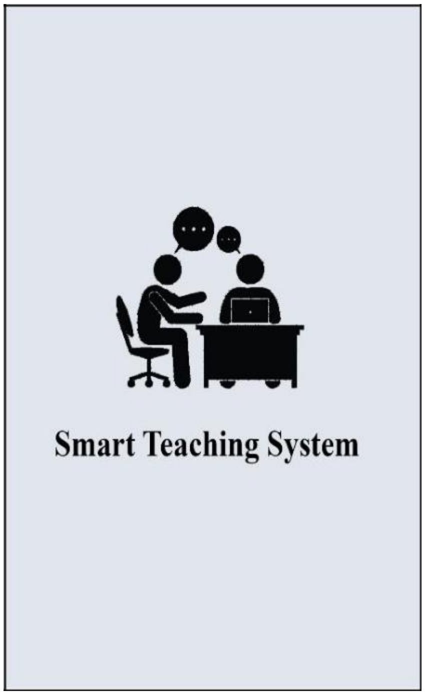
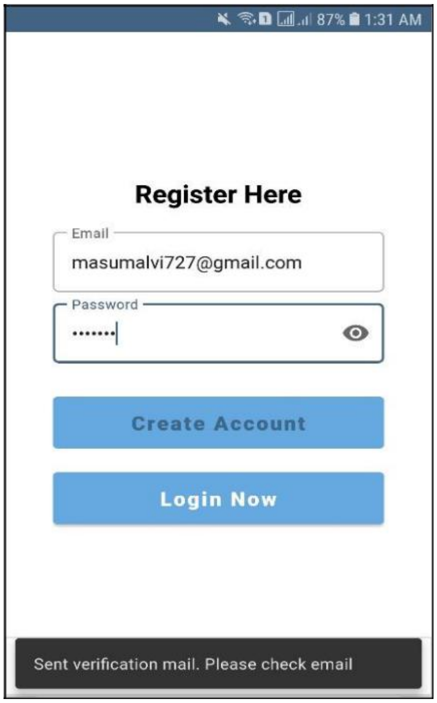


Figure 4.1.1.1: Splash screen

➢

4.1.2 Authentication Screen:

At first both type of user has to register in this system with email and password. After register a confirmation email sent in mail. Then your have to confirm his/her registration by click this mail link. Without confirmation user can not login.



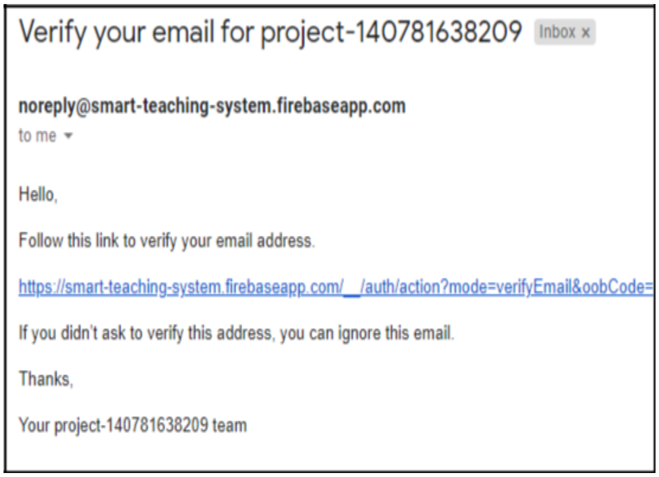
Register Here

Email: masumalvi727@gmail.com

Password: •••••••

Sent verification mail. Please check email

Figure 4.1.2.1: Register screen


Verify your email for project-140781638209 Inbox ×

noreply@smart-teaching-system.firebaseapp.com
to me

Hello,

Follow this link to verify your email address.

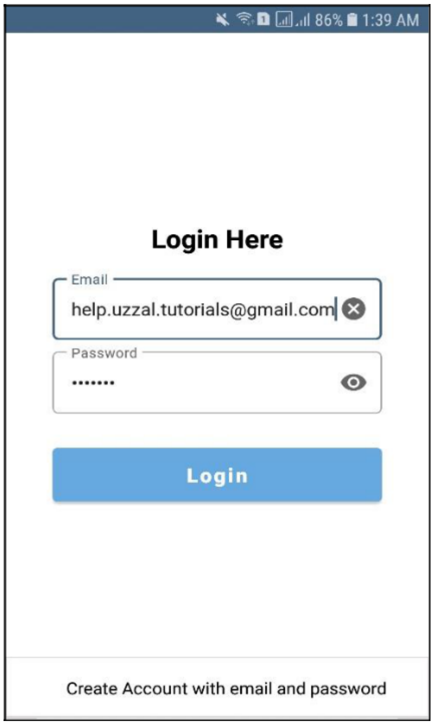
https://smart-teaching-system.firebaseapp.com/__/auth/action?mode=verifyEmail&oobCode=

If you didn't ask to verify this address, you can ignore this email.

Thanks,

Your project-140781638209 team

Figure 4.1.2.2: Registration verification mail



1:39 AM 86%

Login Here

Email

help.uzzal.tutorials@gmail.com

Password

•••••••

Figure 4.1.2.2: User login screen

- ### 4.1.3 Registration Screen:

After user registration and login user have to specific registration again. Register as a teacher or as a student. In this registration both teacher and student enter their details information. And by this registration we keep track the user role.

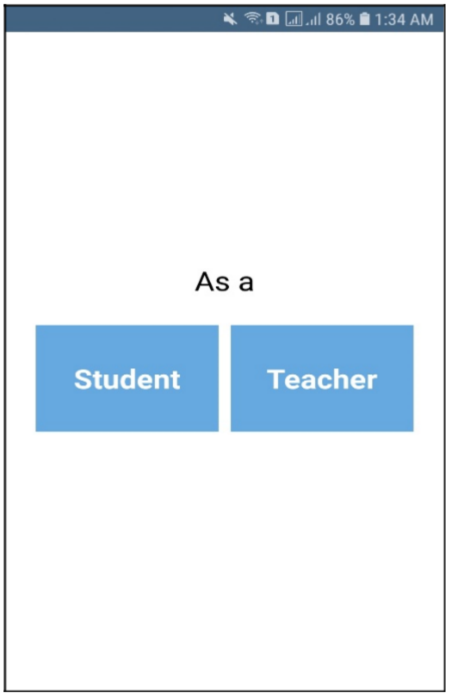
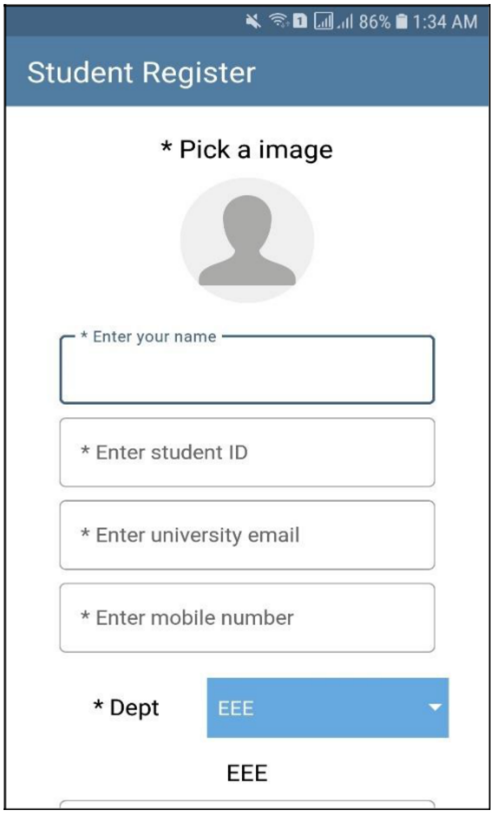


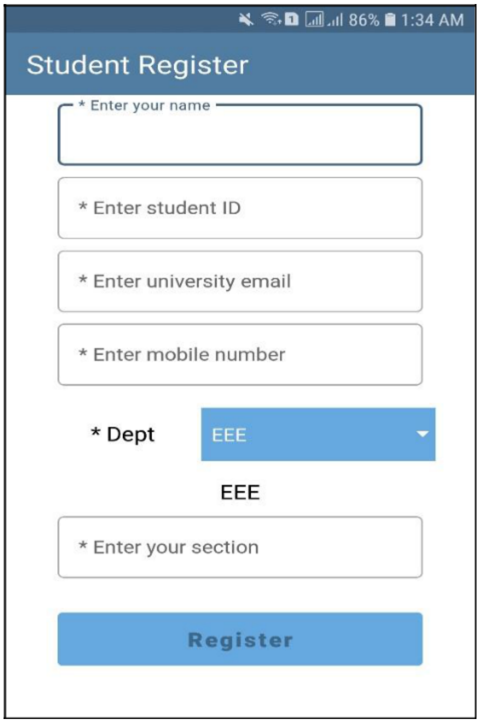
Figure 4.1.3.1: User type choose screen



* Dept EEE

EEE

Figure 4.1.3.2: Student register screen-1



* Dept EEE

EEE

Figure 4.1.3.3: Student register screen-2

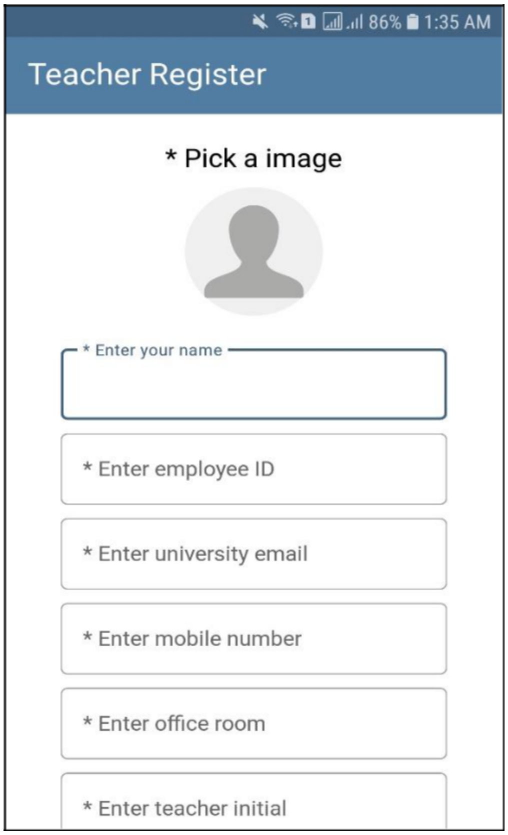
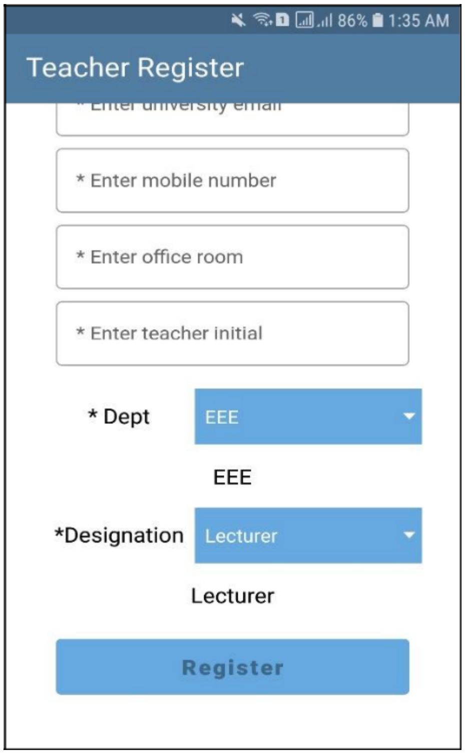


Figure 4.1.3.4: Teacher register screen-1



Teacher Register

* Enter university email

* Enter mobile number

* Enter office room

* Enter teacher initial

* Dept EEE

EEE

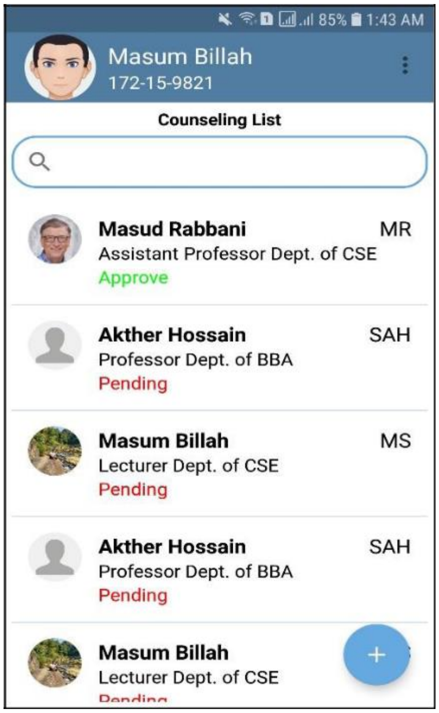
*Designation Lecturer

Lecturer

Figure 4.1.3.5: Teacher register screen-2

➢ 4.1.4 Student activities:

In Student home page he/she can see all of appointment list which they already request with appointment status. And student can find any appointment by search teacher name or teacher initial. Student also can delete the appointment when a appointment is accept or decline. But pending appointment cannot delete.



85% 1:43 AM

Masum Billah
172-15-9821

Counseling List

Masud Rabbani MR
Assistant Professor Dept. of CSE
Approve

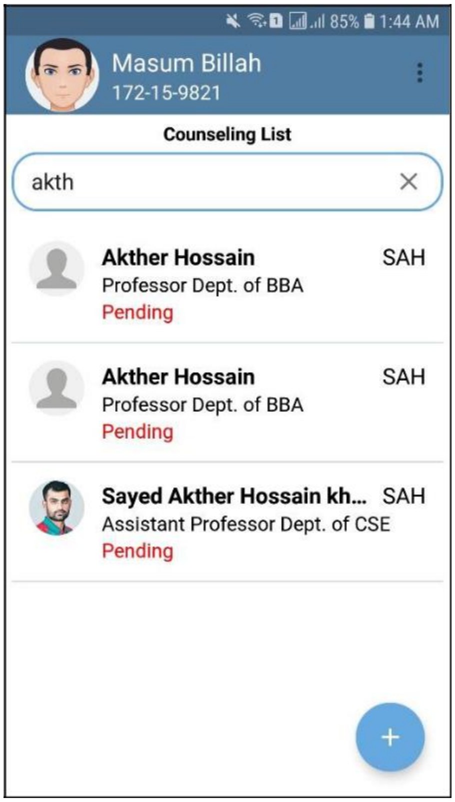
Akther Hossain SAH
Professor Dept. of BBA
Pending

Masum Billah MS
Lecturer Dept. of CSE
Pending

Akther Hossain SAH
Professor Dept. of BBA
Pending

Masum Billah
Lecturer Dept. of CSE
Pending

Figure 4.1.4.1: Student home screen



85% 1:44 AM

Masum Billah
172-15-9821

Counseling List

akth

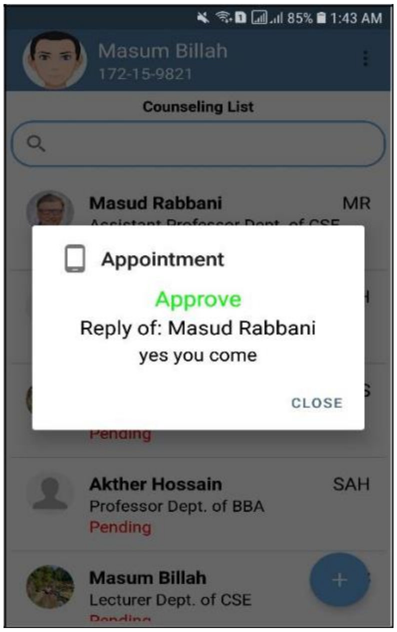
Akther Hossain SAH
Professor Dept. of BBA
Pending

Akther Hossain SAH
Professor Dept. of BBA
Pending

Sayed Akther Hossain kh... SAH
Assistant Professor Dept. of CSE
Pending

Figure 4.1.4.2: Search from home screen by teacher name

Student also can see the details of this appointment with status. When an appointment is pending, student can see his/her message in dialogue box with status “Pending”. But when an appointment is approved or decline, student can see the teacher message with the status of “Accept” or “Decline” like below:



Masum Billah
172-15-9821

Masud Rabbani MR
Assistant Professor Dept. of CSE

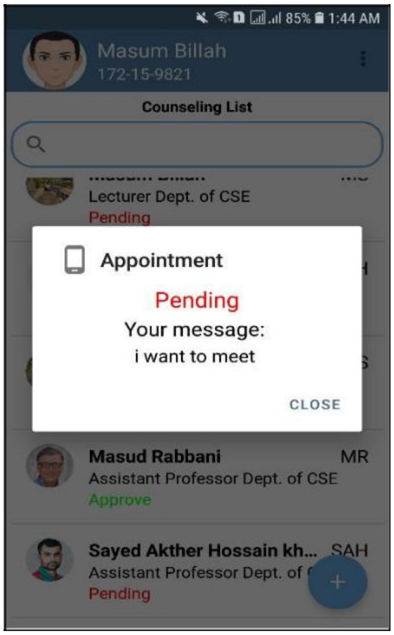
Appointment
Approve
Reply of: Masud Rabbani
yes you come

Pending

Akther Hossain SAH
Professor Dept. of BBA
Pending

Masum Billah
Lecturer Dept. of CSE
Pending

Figure 4.1.4.3: Approve/Decline appointment dialogue



Masum Billah
172-15-9821

Lecturer Dept. of CSE
Pending

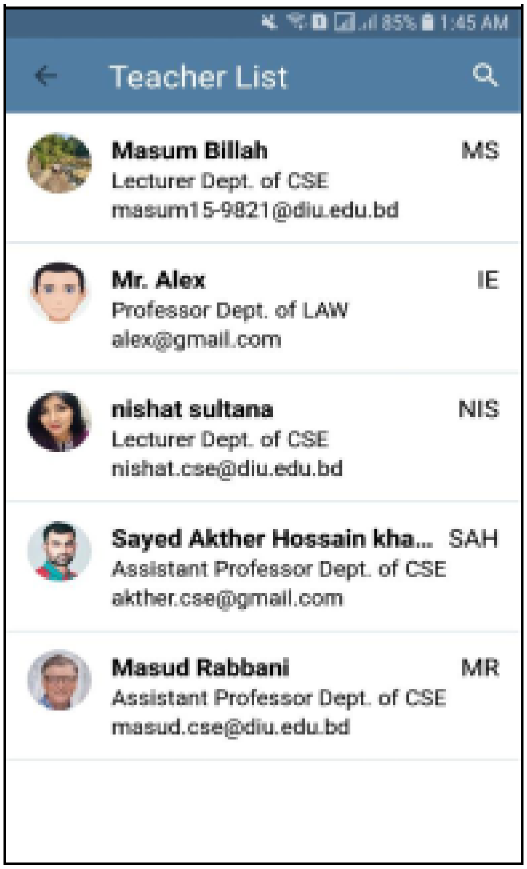
Appointment
Pending
Your message:
i want to meet

Masud Rabbani MR
Assistant Professor Dept. of CSE
Approve

Sayed Akther Hossain kh... SAH
Assistant Professor Dept. of
Pending

Figure 4.1.4.4: Pending appointment dialogue

Student can see the all-teacher list who are already register in this system. They can find the teacher by searching the teacher's name or initial and see the details information about teacher like name, email, phone, office location, counseling time. And student can apply for appointment with the reason. After that teacher receive a notification.



Teacher List

Masum Billah MS
Lecturer Dept. of CSE
masum15-9821@diu.edu.bd

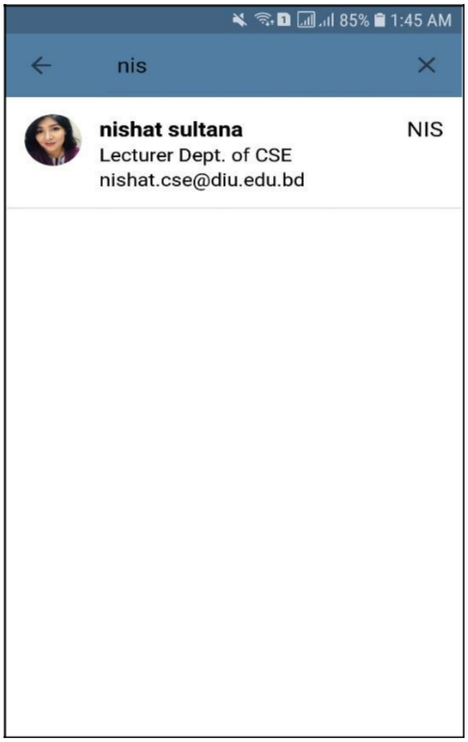
Mr. Alex IE
Professor Dept. of LAW
alex@gmail.com

nishat sultana NIS
Lecturer Dept. of CSE
nishat.cse@diu.edu.bd

Sayed Akther Hossain kha... SAH
Assistant Professor Dept. of CSE
akther.cse@gmail.com

Masud Rabbani MR
Assistant Professor Dept. of CSE
masud.cse@diu.edu.bd

Figure 4.1.4.5: All teacher list screen

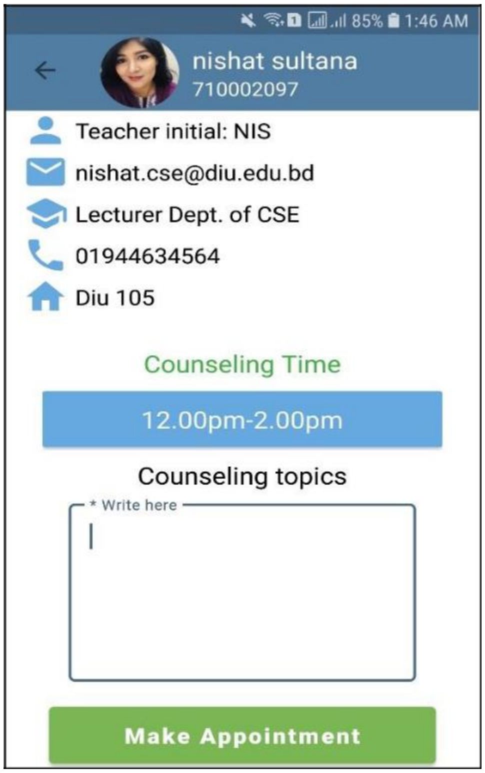


nishat sultana NIS

Lecturer Dept. of CSE

nishat.cse@diu.edu.bd

Figure 4.1.4.6: Teacher search by name or initial screen



nishat sultana

710002097

Teacher initial: NIS

nishat.cse@diu.edu.bd

Lecturer Dept. of CSE

01944634564

Diu 105

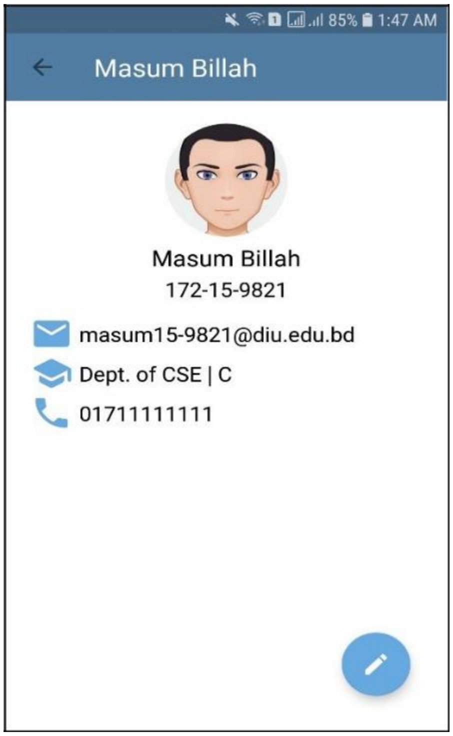
Counseling Time

12.00pm-2.00pm

Counseling topics

Figure 4.1.4.7: Teacher details and make appointment screen

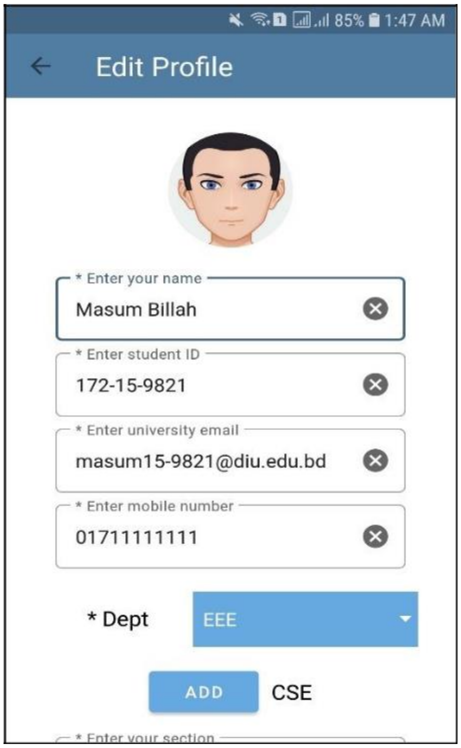
Student can see their profile and update their profile



Masum Billah
172-15-9821

masum15-9821@diu.edu.bd
Dept. of CSE | C
01711111111

Figure 4.1.4.8: Student profile screen



* Enter your name
Masum Billah

* Enter student ID
172-15-9821

* Enter university email
masum15-9821@diu.edu.bd

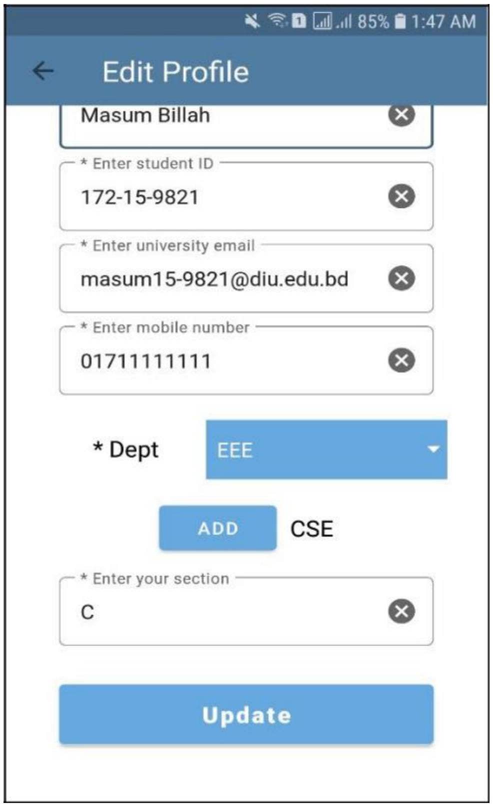
* Enter mobile number
01711111111

* Dept EEE

CSE

* Enter your section

Figure 4.1.4.9: Student edit profile screen-1



Masum Billah

* Enter student ID
172-15-9821

* Enter university email
masum15-9821@diu.edu.bd

* Enter mobile number
01711111111

* Dept EEE

CSE

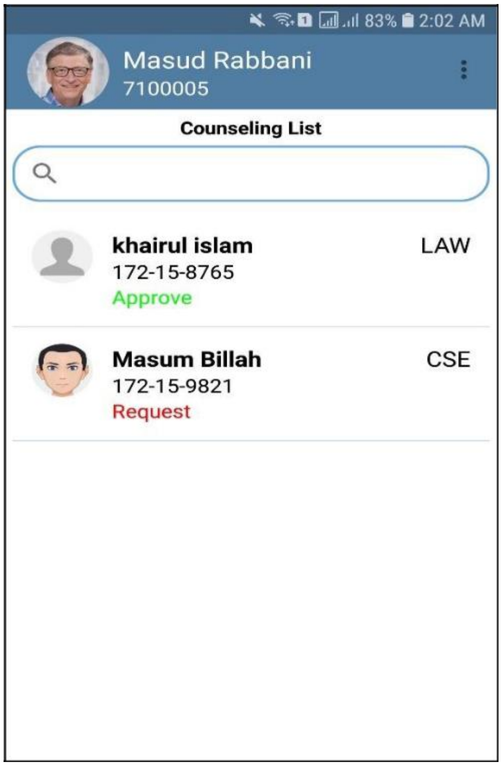
* Enter your section
C

Figure 4.1.4.10: Student edit profile screen-2

➢

4.1.5 Teacher activities:

In teacher home page he/she can see all student appointment list. And find out any specific name by search by student name. And teacher can delete the appointment only when an appointment is accept or decline. It is similar with the student home page.



83% 2:02 AM

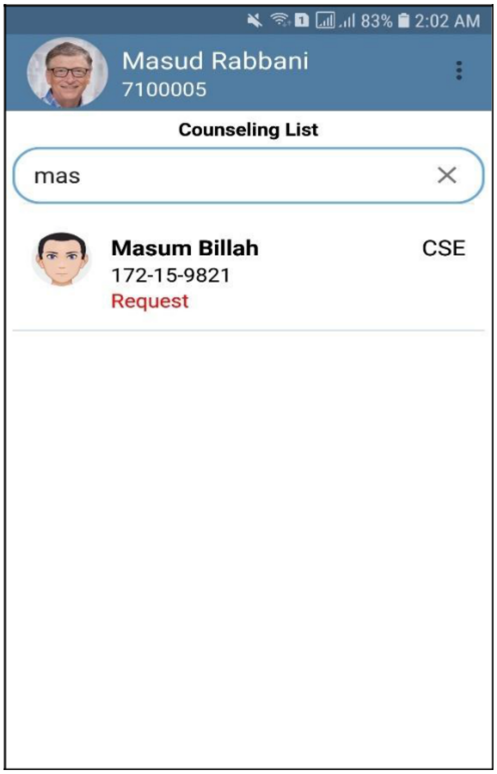
Masud Rabbani
7100005

Counseling List

khairul islam LAW
172-15-8765
Approve

Masum Billah CSE
172-15-9821
Request

Figure 4.1.5.1: Teacher home screen



83% 2:02 AM

Masud Rabbani
7100005

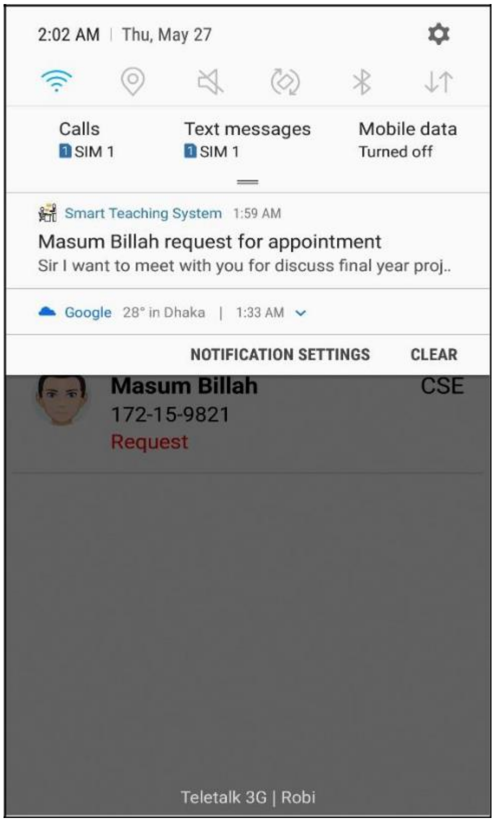
Counseling List

mas

Masum Billah CSE
172-15-9821
Request

Figure 4.1.5.2: Teacher home screen search

When a student makes an appointment, teacher received a notification. It is same with student. When teacher response a request then student receive a notification. Teacher can see the appointment details with the information of student. And response with a message. When an appointment already approved/decline, then the approve/decline button is disable.



Smart Teaching System 1:59 AM

Masum Billah request for appointment

Sir I want to meet with you for discuss final year proj..

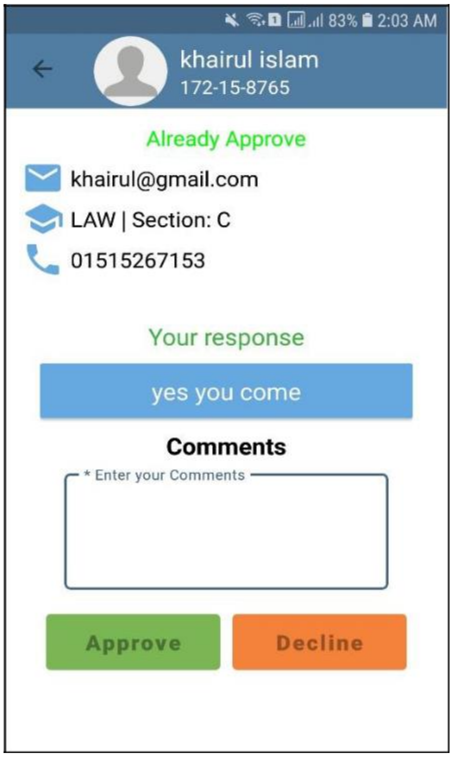
Google 28° in Dhaka | 1:33 AM

Masum Billah CSE

172-15-9821

Request

Figure 4.1.5.3: Teacher notification receive



khairul islam
172-15-8765

Already Approve

khairul@gmail.com

LAW | Section: C

01515267153

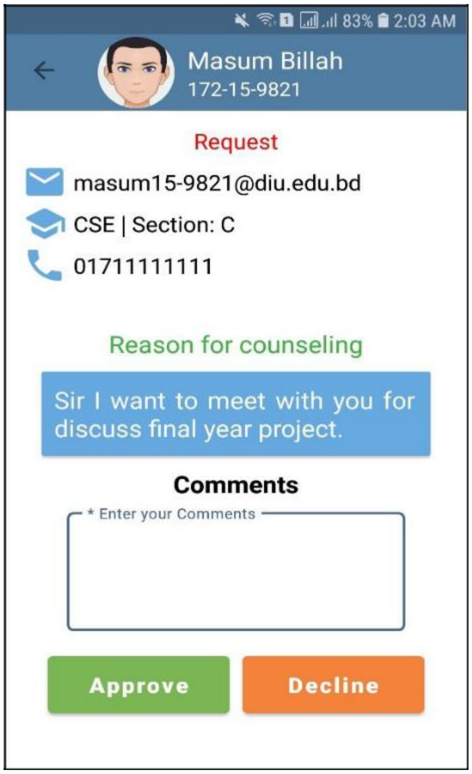
Your response

yes you come

Comments

* Enter your Comments

Figure 4.1.5.4: Approve appointment screen



Masum Billah
172-15-9821

Request

masum15-9821@diu.edu.bd

CSE | Section: C

01711111111

Reason for counseling

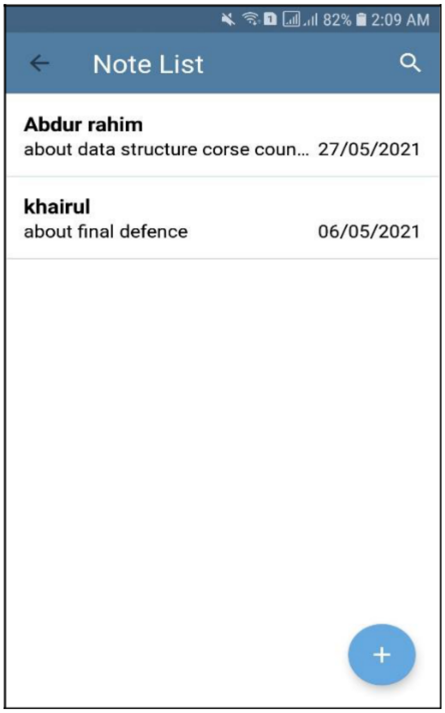
Sir I want to meet with you for discuss final year project.

Comments

* Enter your Comments

Figure 4.1.5.5: Request appointment screen

Teacher can take note of the appointment in Note item page.



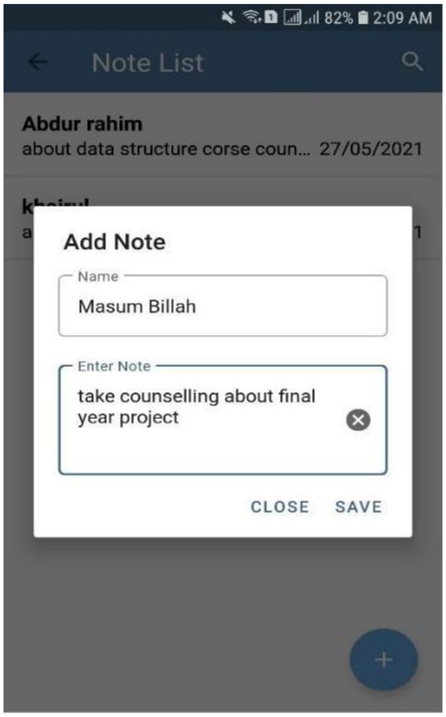
2:09 AM 82%

Note List

Abdur rahim
about data structure corse coun... 27/05/2021

khairul
about final defence 06/05/2021

Figure 4.1.5.5: Teacher note list



2:09 AM 82%

Note List

Abdur rahim
about data structure corse coun... 27/05/2021

Add Note

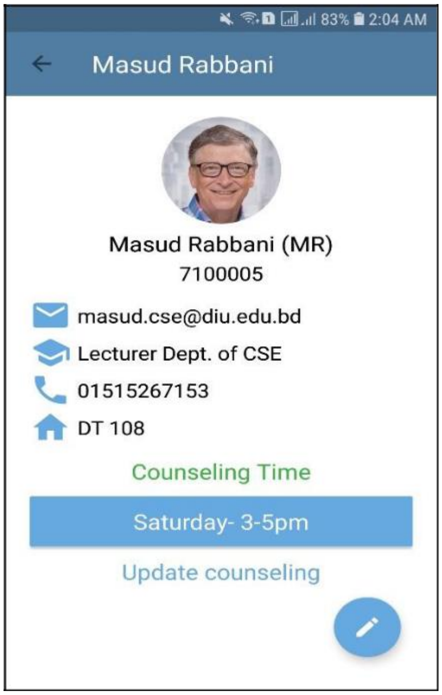
Name: Masum Billah

Enter Note: take counselling about final year project

CLOSE SAVE

Figure 4.1.5.6: Add note screen

Teacher can see his/her profile and in profile screen. Teacher can update his/her counseling time in profile. And teacher can also update his/her profile.



Masud Rabbani

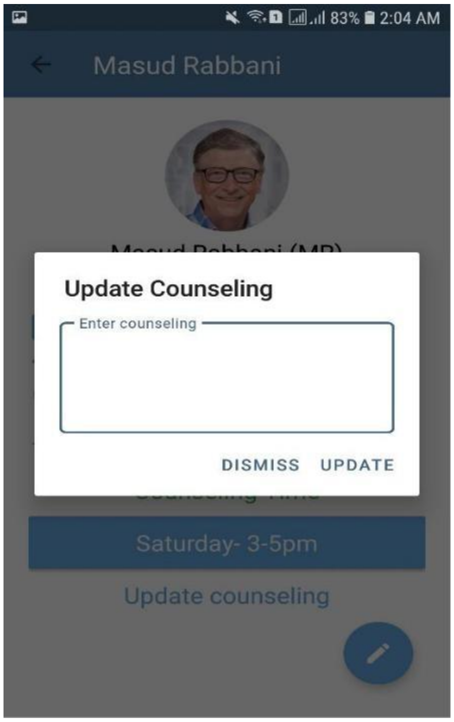
Masud Rabbani (MR)
7100005

masud.cse@diu.edu.bd
Lecturer Dept. of CSE
01515267153
DT 108

Counseling Time

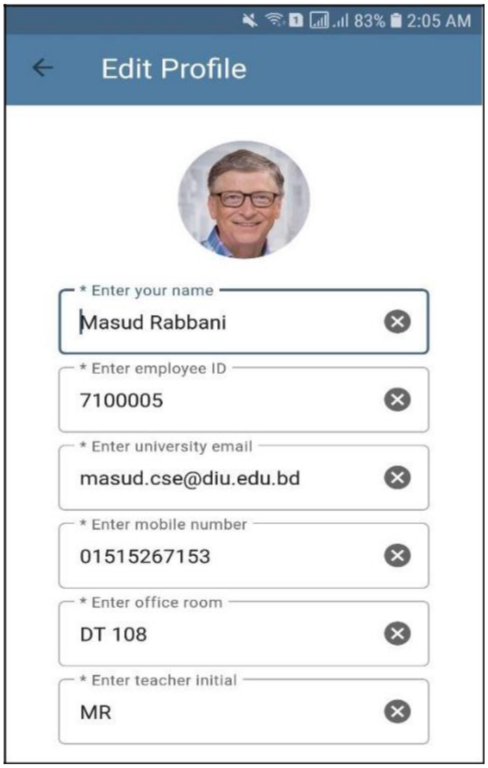
Saturday- 3-5pm

Figure 4.1.5.7: Teacher profile screen



Update Counseling

Figure 4.1.5.8: Teacher update counseling screen



* Enter your name
Masud Rabbani

* Enter employee ID
7100005

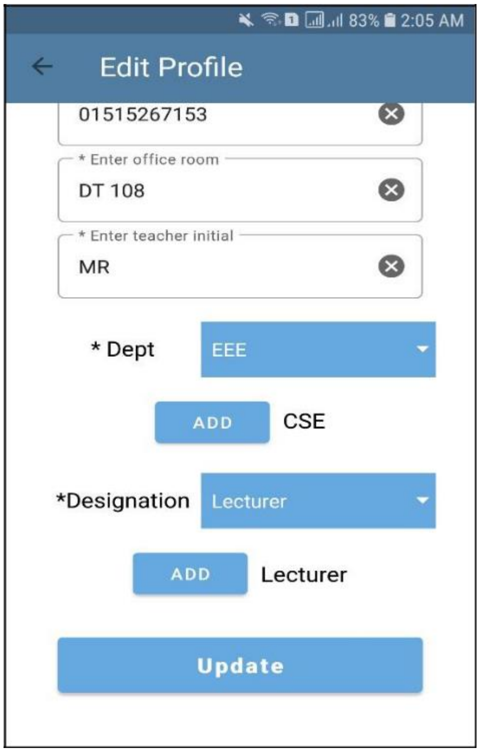
* Enter university email
masud.cse@diu.edu.bd

* Enter mobile number
01515267153

* Enter office room
DT 108

* Enter teacher initial
MR

Figure 4.1.5.9: Teacher update profile screen-1



01515267153

* Enter office room
DT 108

* Enter teacher initial
MR

* Dept: EEE

ADD CSE

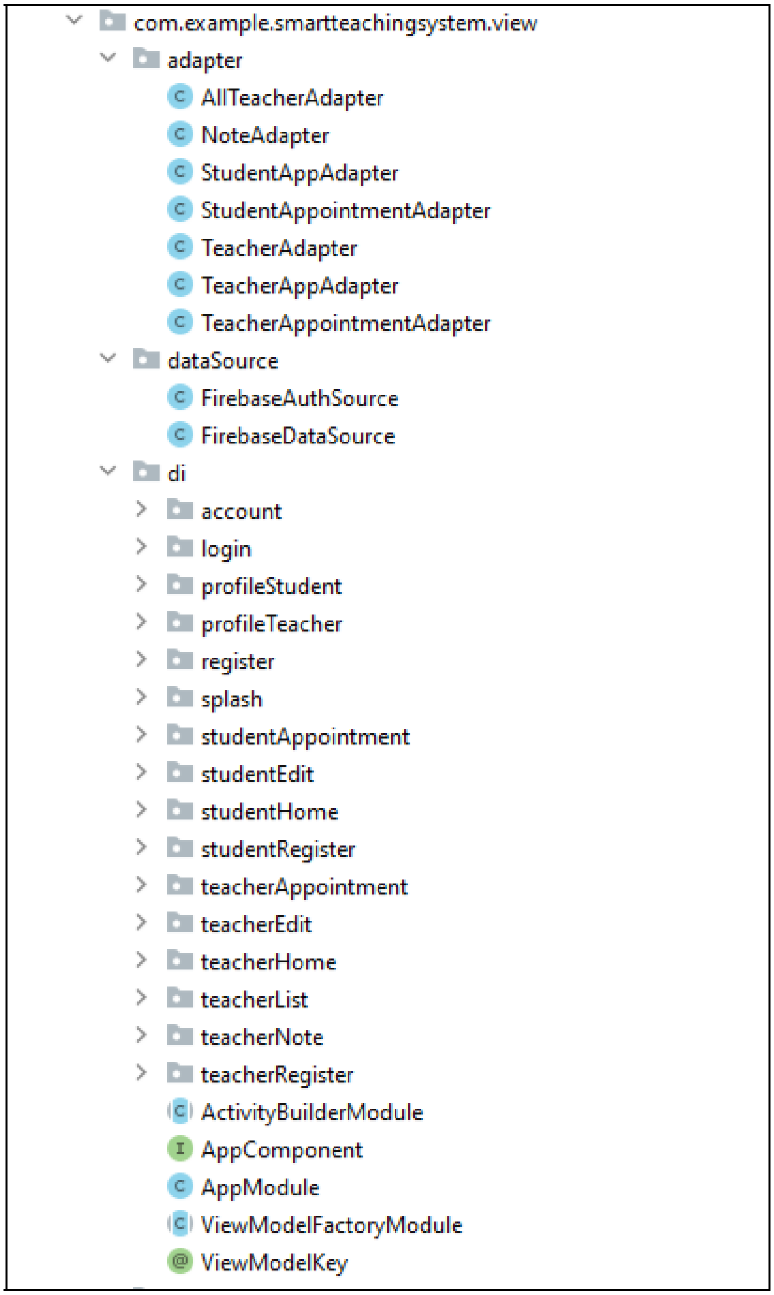
*Designation: Lecturer

ADD Lecturer

Figure 4.1.5.10: Teacher update profile screen-2

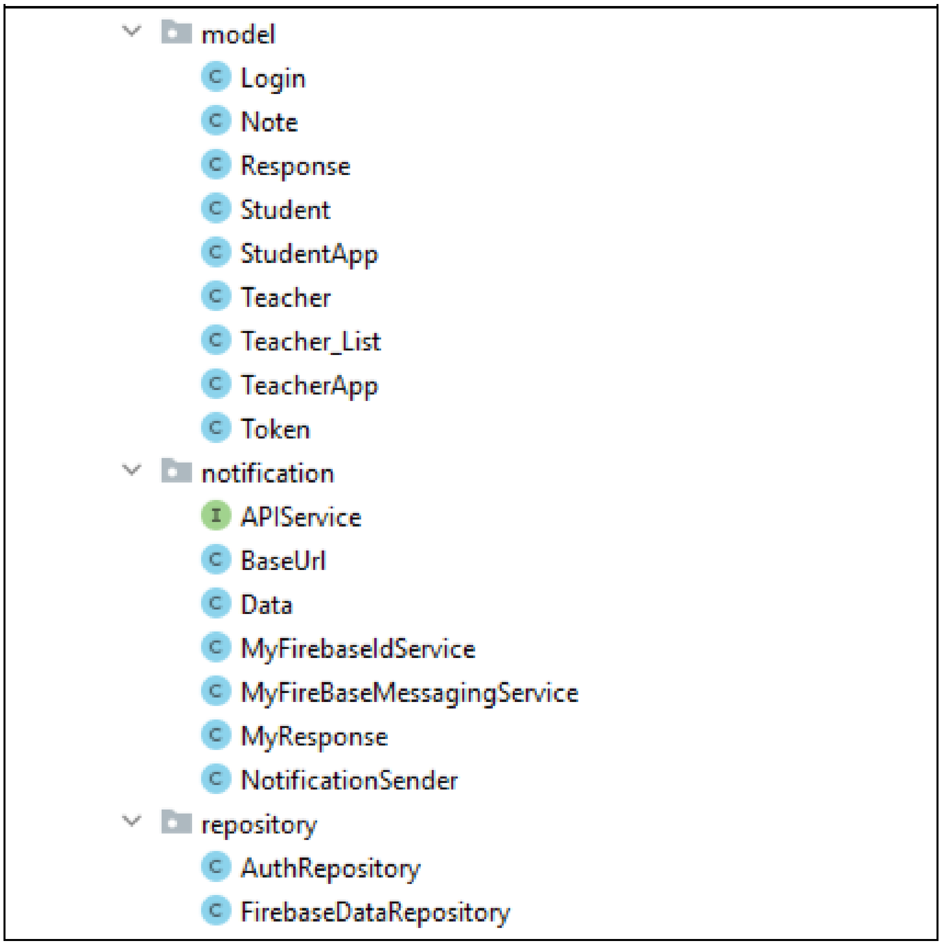
4.2 Back-end Design

There are some file structures in back end of a android project. They are Manifest, Java/Kotlin, Resource and Gradle Script. And resource file has layout, drawable, mipmap, values. The backend file structure added in below:



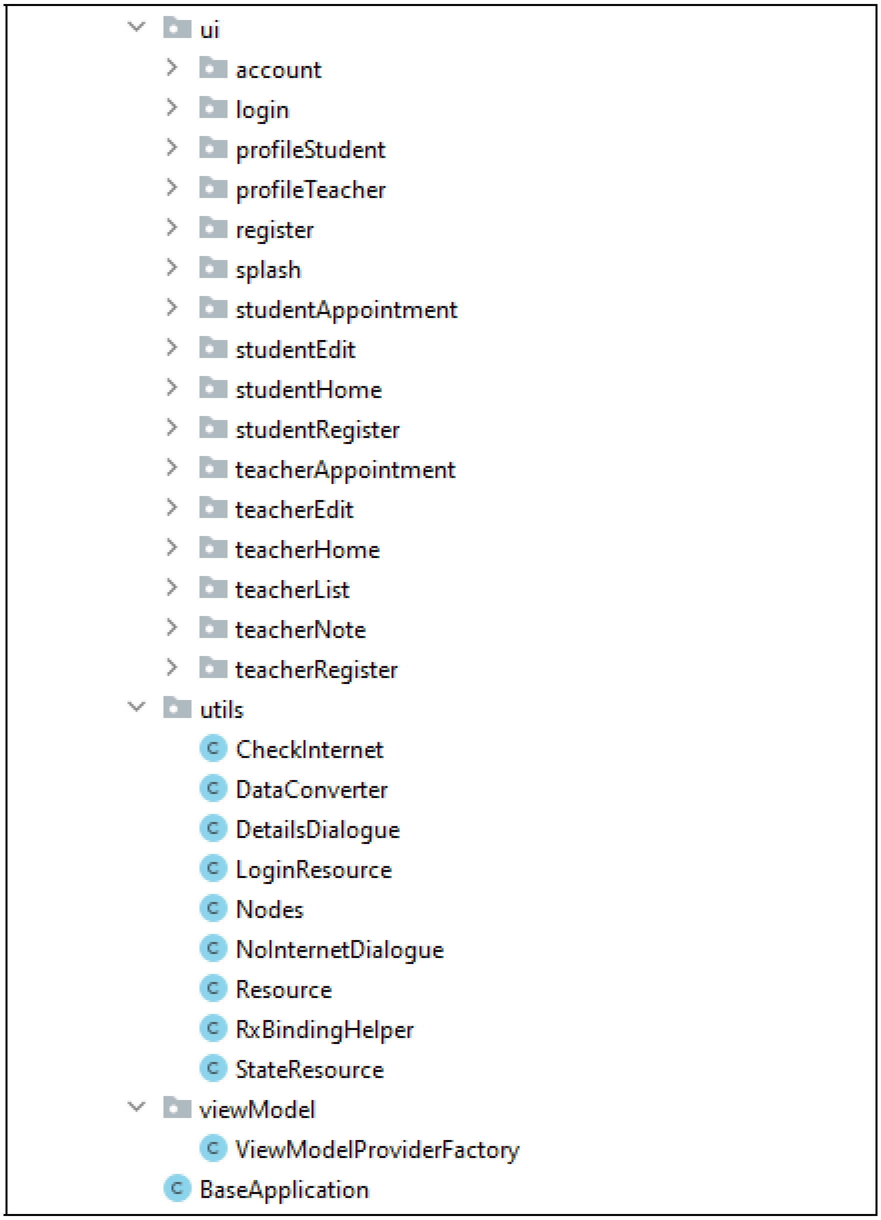
```
com.example.smartteachingsystem.view
    adapter
        AllTeacherAdapter
        NoteAdapter
        StudentAppAdapter
        StudentAppointmentAdapter
        TeacherAdapter
        TeacherAppAdapter
        TeacherAppointmentAdapter
    dataSource
        FirebaseAuthSource
        FirebaseDataSource
    di
        account
        login
        profileStudent
        profileTeacher
        register
        splash
        studentAppointment
        studentEdit
        studentHome
        studentRegister
        teacherAppointment
        teacherEdit
        teacherHome
        teacherList
        teacherNote
        teacherRegister
        ActivityBuilderModule
        AppComponent
        AppModule
        ViewModelFactoryModule
        ViewModelKey
```

Figure 4.2.1: Back-end design which shown that java file divided some part. Here adapter, data source, and di means the dependency injection package.



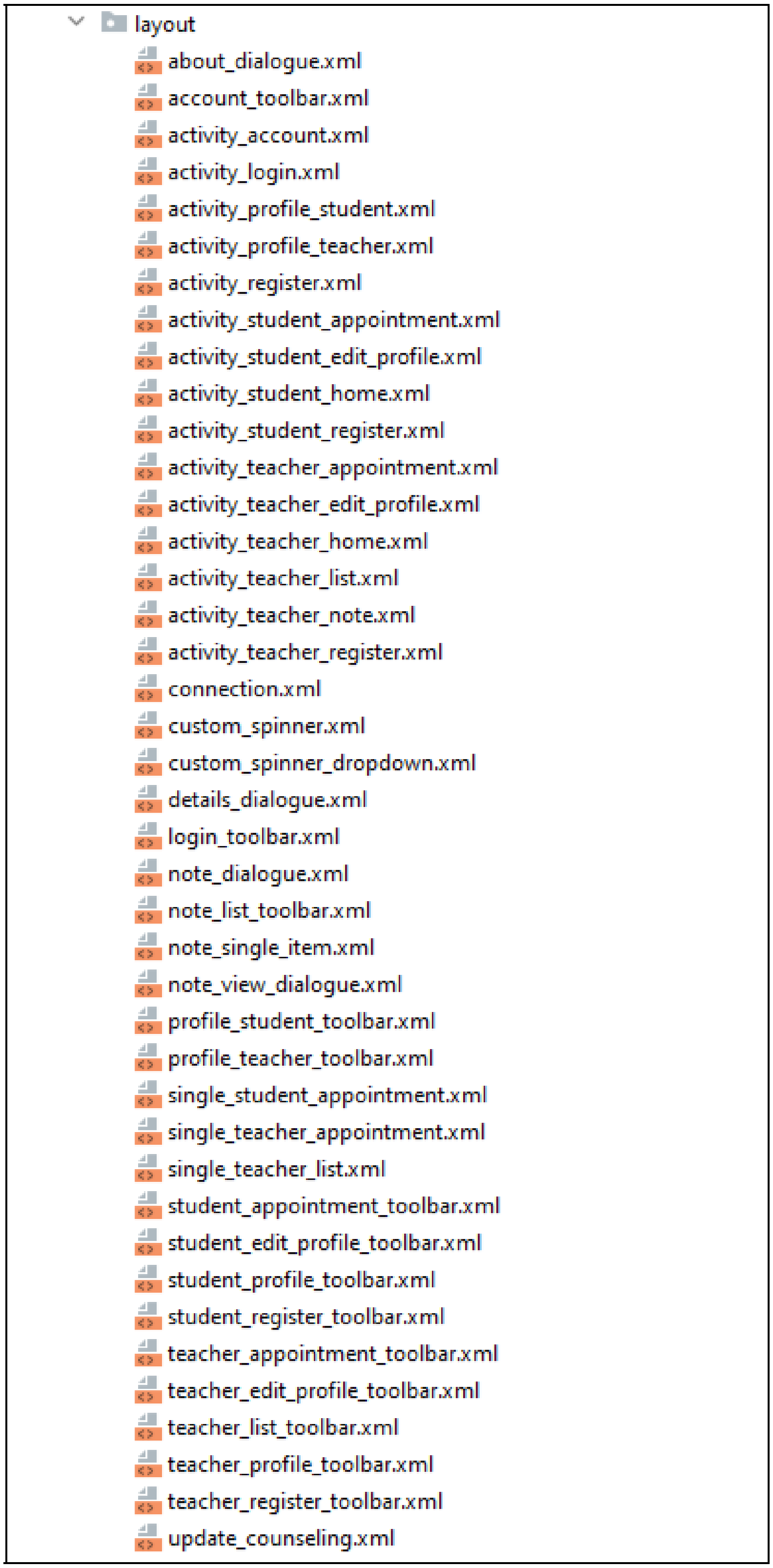
```
model
    Login
    Note
    Response
    Student
    StudentApp
    Teacher
    Teacher_List
    TeacherApp
    Token
notification
    APIService
    BaseUrl
    Data
    MyFirebaseIdService
    MyFireBaseMessagingService
    MyResponse
    NotificationSender
repository
    AuthRepository
    FirebaseDataRepository
```

Figure 4.2.2: Back-end design which shown that java file divided some part. Here model, notification and repository package.



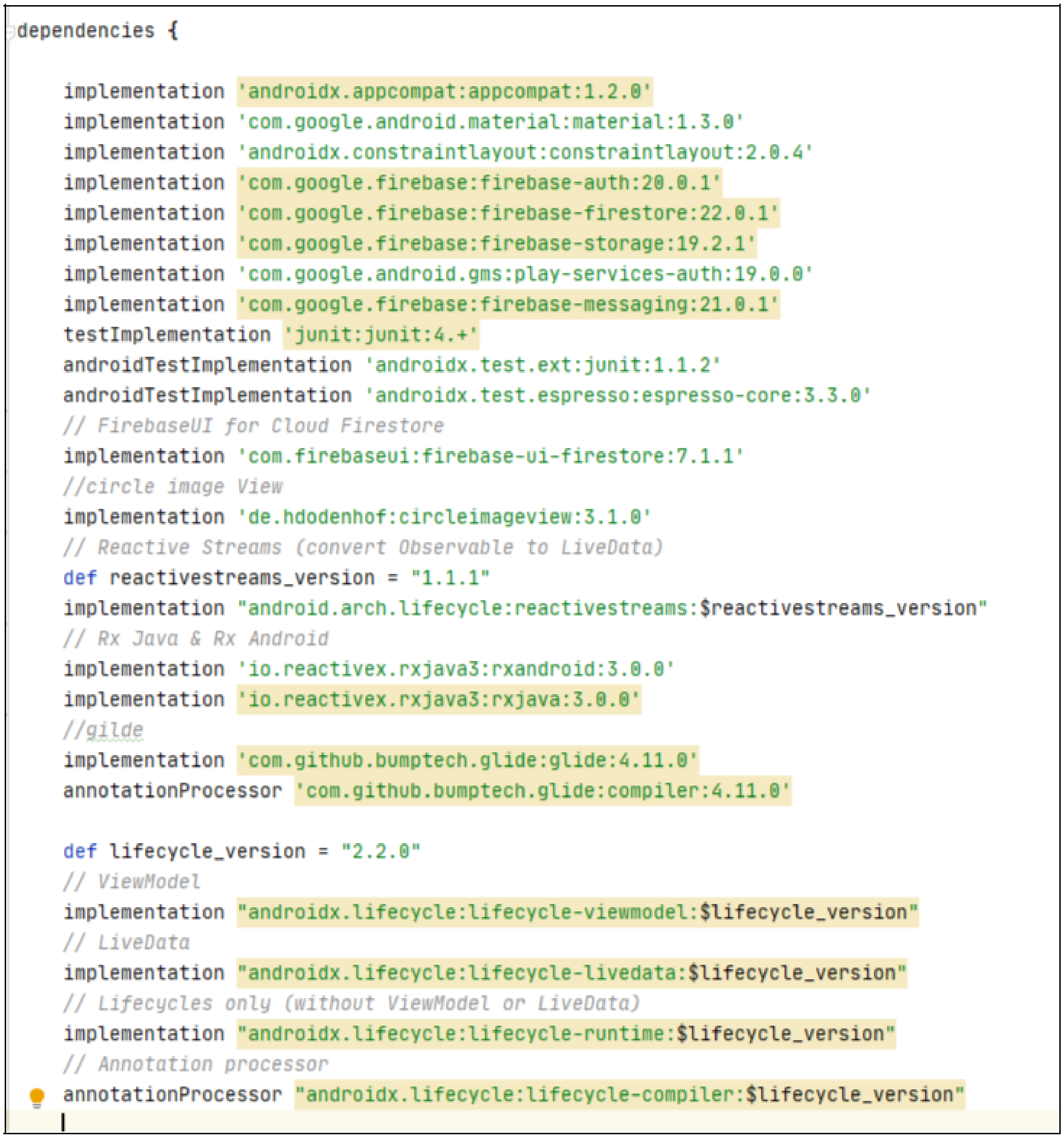
```
ui
  account
  login
  profileStudent
  profileTeacher
  register
  splash
  studentAppointment
  studentEdit
  studentHome
  studentRegister
  teacherAppointment
  teacherEdit
  teacherHome
  teacherList
  teacherNote
  teacherRegister
utils
  CheckInternet
  DataConverter
  DetailsDialogue
  LoginResource
  Nodes
  NoInternetDialogue
  Resource
  RxBindingHelper
  StateResource
viewModel
  ViewModelProviderFactory
BaseApplication
```

Figure 4.2.3: This is also a part of java file. Which show UI, Utils, and view model package



```
layout
    about_dialogue.xml
    account_toolbar.xml
    activity_account.xml
    activity_login.xml
    activity_profile_student.xml
    activity_profile_teacher.xml
    activity_register.xml
    activity_student_appointment.xml
    activity_student_edit_profile.xml
    activity_student_home.xml
    activity_student_register.xml
    activity_teacher_appointment.xml
    activity_teacher_edit_profile.xml
    activity_teacher_home.xml
    activity_teacher_list.xml
    activity_teacher_note.xml
    activity_teacher_register.xml
    connection.xml
    custom_spinner.xml
    custom_spinner_dropdown.xml
    details_dialogue.xml
    login_toolbar.xml
    note_dialogue.xml
    note_list_toolbar.xml
    note_single_item.xml
    note_view_dialogue.xml
    profile_student_toolbar.xml
    profile_teacher_toolbar.xml
    single_student_appointment.xml
    single_teacher_appointment.xml
    single_teacher_list.xml
    student_appointment_toolbar.xml
    student_edit_profile_toolbar.xml
    student_profile_toolbar.xml
    student_register_toolbar.xml
    teacher_appointment_toolbar.xml
    teacher_edit_profile_toolbar.xml
    teacher_list_toolbar.xml
    teacher_profile_toolbar.xml
    teacher_register_toolbar.xml
    update_counseling.xml
```

Figure 4.2.4: This all-layout file. Layout file contain all front-end design of a android application

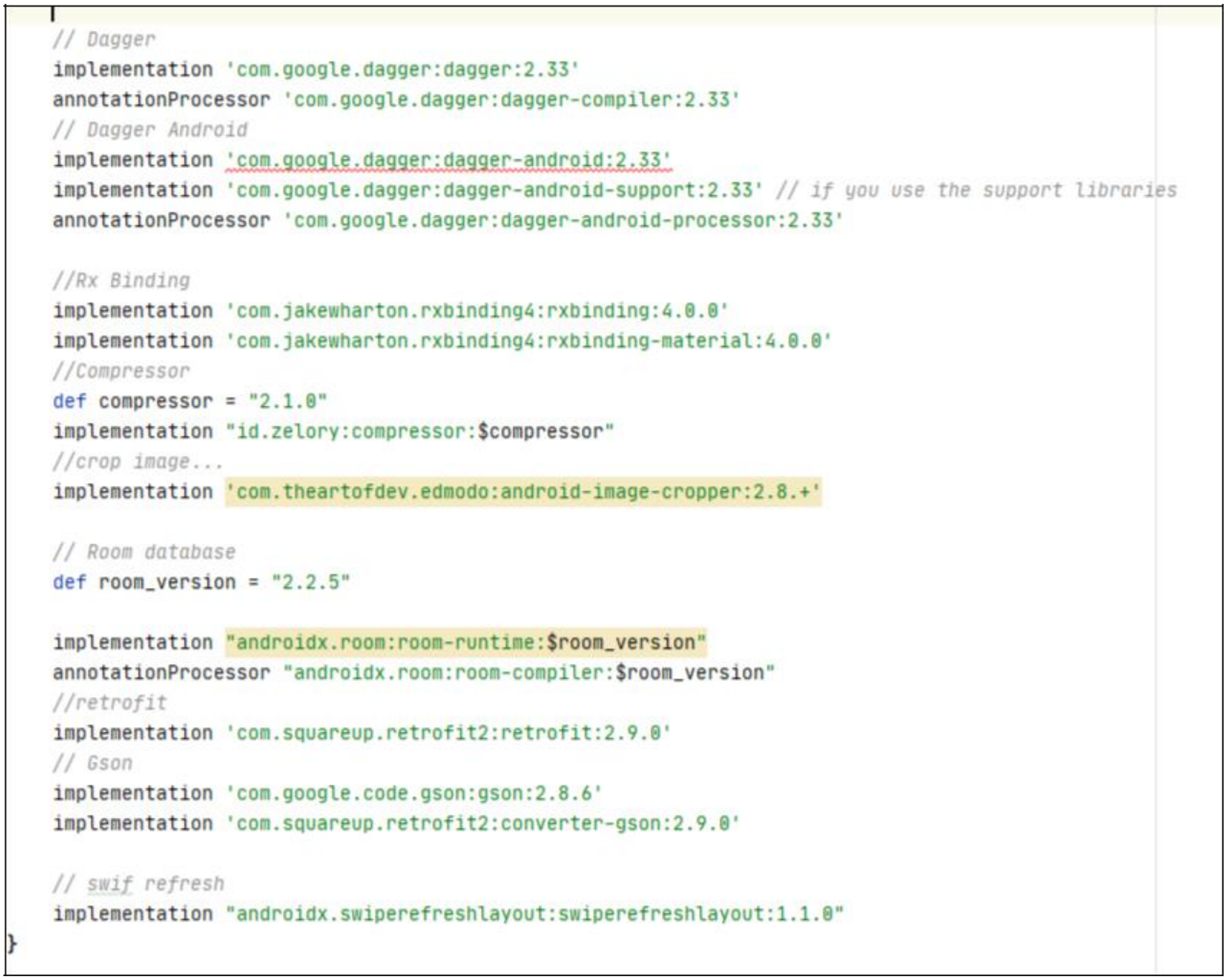


```
dependencies {

    implementation 'androidx.appcompat:appcompat:1.2.0'
    implementation 'com.google.android.material:material:1.3.0'
    implementation 'androidx.constraintlayout:constraintlayout:2.0.4'
    implementation 'com.google.firebase:firebase-auth:20.0.1'
    implementation 'com.google.firebase:firebase-firestore:22.0.1'
    implementation 'com.google.firebase:firebase-storage:19.2.1'
    implementation 'com.google.android.gms:play-services-auth:19.0.0'
    implementation 'com.google.firebase:firebase-messaging:21.0.1'
    testImplementation 'junit:junit:4.+'
    androidTestImplementation 'androidx.test.ext:junit:1.1.2'
    androidTestImplementation 'androidx.test.espresso:espresso-core:3.3.0'
    // FirebaseUI for Cloud Firestore
    implementation 'com.firebaseui:firebase-ui-firestore:7.1.1'
    //circle image View
    implementation 'de.hdodenhof:circleimageview:3.1.0'
    // Reactive Streams (convert Observable to LiveData)
    def reactivestreams_version = "1.1.1"
    implementation "android.arch.lifecycle:reactivestreams:$reactivestreams_version"
    // Rx Java & Rx Android
    implementation 'io.reactivex.rxjava3:rxandroid:3.0.0'
    implementation 'io.reactivex.rxjava3:rxjava:3.0.0'
    //gilde
    implementation 'com.github.bumptech.glide:glide:4.11.0'
    annotationProcessor 'com.github.bumptech.glide:compiler:4.11.0'

    def lifecycle_version = "2.2.0"
    // ViewModel
    implementation "androidx.lifecycle:lifecycle-viewmodel:$lifecycle_version"
    // LiveData
    implementation "androidx.lifecycle:lifecycle-livedata:$lifecycle_version"
    // Lifecycles only (without ViewModel or LiveData)
    implementation "androidx.lifecycle:lifecycle-runtime:$lifecycle_version"
    // Annotation processor
    annotationProcessor "androidx.lifecycle:lifecycle-compiler:$lifecycle_version"
```

Figure 4.2.5: Gradle Script-1 which added all necessity libraries and third-party library.



```
    // Dagger
    implementation 'com.google.dagger:dagger:2.33'
    annotationProcessor 'com.google.dagger:dagger-compiler:2.33'
    // Dagger Android
    implementation 'com.google.dagger:dagger-android:2.33'
    implementation 'com.google.dagger:dagger-android-support:2.33' // if you use the support libraries
    annotationProcessor 'com.google.dagger:dagger-android-processor:2.33'

    //Rx Binding
    implementation 'com.jakewharton.rxbinding4:rxbinding:4.0.0'
    implementation 'com.jakewharton.rxbinding4:rxbinding-material:4.0.0'
    //Compressor
    def compressor = "2.1.0"
    implementation "id.zelory:compressor:$compressor"
    //crop image...
    implementation 'com.theartofdev.edmodo:android-image-cropper:2.8.+'

    // Room database
    def room_version = "2.2.5"

    implementation "androidx.room:room-runtime:$room_version"
    annotationProcessor "androidx.room:room-compiler:$room_version"
    //retrofit
    implementation 'com.squareup.retrofit2:retrofit:2.9.0'
    // Gson
    implementation 'com.google.code.gson:gson:2.8.6'
    implementation 'com.squareup.retrofit2:converter-gson:2.9.0'

    // swif refresh
    implementation "androidx.swiperefreshlayout:swiperefreshlayout:1.1.0"
}
```

Figure 4.2.6: Gradle Script-2 which added all necessity libraries and third-party library.

4.3 Implementation Requirements

- Database: Firebase Cloud Firestore, Firebase Storage
- Dependency Injection: Dagger (2,3)
- Retrofit: for notification response
- Rx Java
- MVVM Architecture Pattern
- View Model & Live Data
- AndroidX
- Language: Java

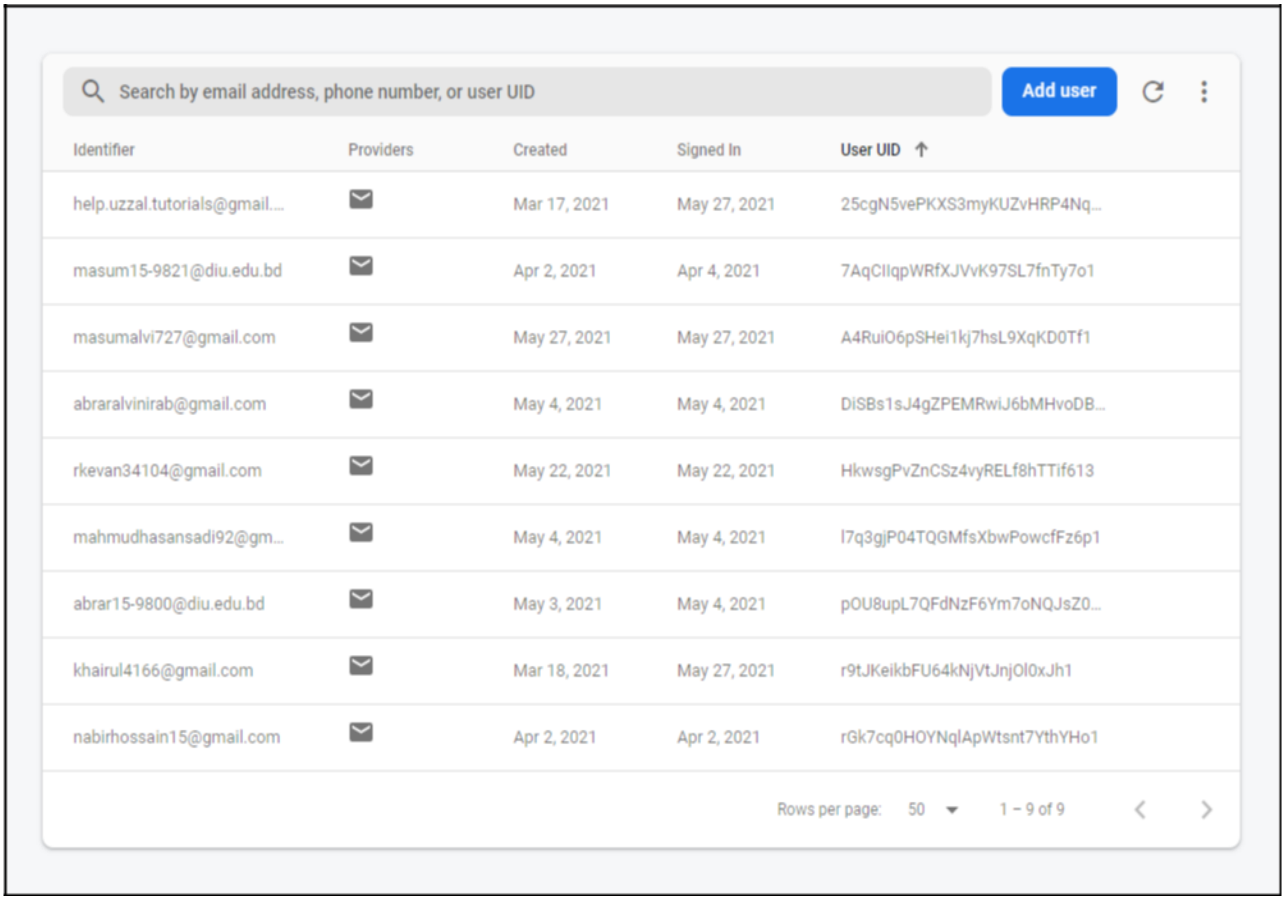
Chapter 5: Implementation and Testing

5.1 Implementation of Database

In our project we used Firebase database. It is easy to control user and easy to manage the user. There are multiple authentication system of firebase like google account, phone, email password, facebook, yahoo etc. We use the email password authentication system.

➢ 5.1.1 User authentication

When a user register in this system a unique user id generated by this database. For that reason, it is easy to handle and uniquely identified every user.



Identifier	Providers	Created	Signed In	User UID ↑
help.uzzal.tutorials@gmail....	✉	Mar 17, 2021	May 27, 2021	25cgN5vePKXS3myKUZvHRP4Nq...
masum15-9821@diu.edu.bd	✉	Apr 2, 2021	Apr 4, 2021	7AqClIqpWRfXJVvK97SL7fnTy7o1
masumalvi727@gmail.com	✉	May 27, 2021	May 27, 2021	A4RuiO6pSHei1kj7hsL9XqKD0Tf1
abraralvinirab@gmail.com	✉	May 4, 2021	May 4, 2021	DiSBs1sJ4gZPEMRwiJ6bMHvoDB...
rkevan34104@gmail.com	✉	May 22, 2021	May 22, 2021	HkwsgPvZnCSz4vyRELf8hTTif613
mahmudhasansadi92@gm...	✉	May 4, 2021	May 4, 2021	l7q3gjP04TQGMfsXbwPowcfFz6p1
abrar15-9800@diu.edu.bd	✉	May 3, 2021	May 4, 2021	pOU8upL7QFdNzF6Ym7oNQJsZ0...
khairul4166@gmail.com	✉	Mar 18, 2021	May 27, 2021	r9tJKeikbFU64kNjVtJnjOl0xJh1
nabirhossain15@gmail.com	✉	Apr 2, 2021	Apr 2, 2021	rGk7cq0HOYNqlApWtsnt7YthYHo1

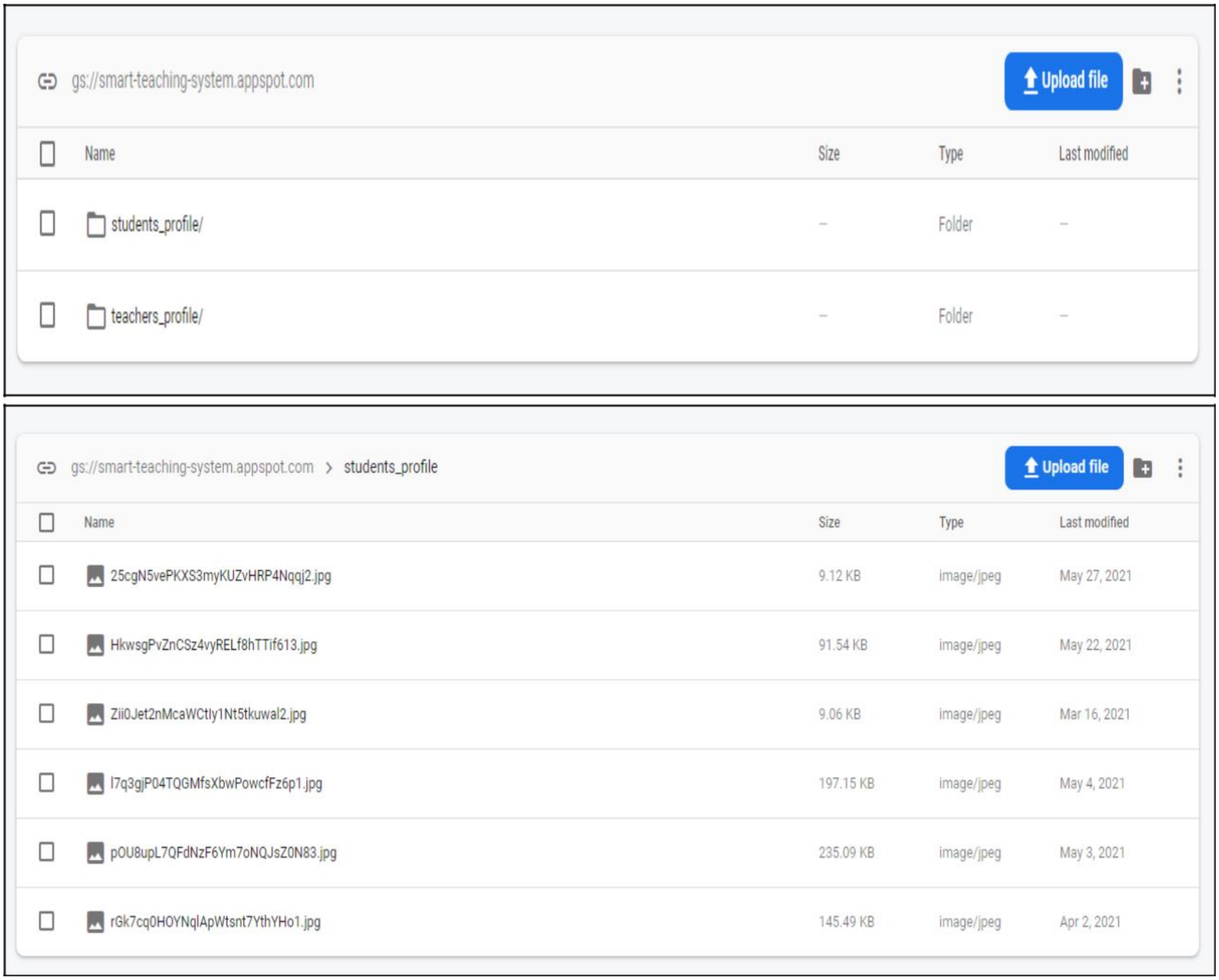
Rows per page: 50 ▾ 1 – 9 of 9

Figure 5.1.1.1: Firebase authenticate users

➢

5.1.2 Firebase Storage

Firebase storage used for save user profile images. This image is stored against every user unique id.



gs://smart-teaching-system.appspot.com

□	Name	Size	Type	Last modified
□	students_profile/	–	Folder	–
□	teachers_profile/	–	Folder	–

gs://smart-teaching-system.appspot.com > students_profile

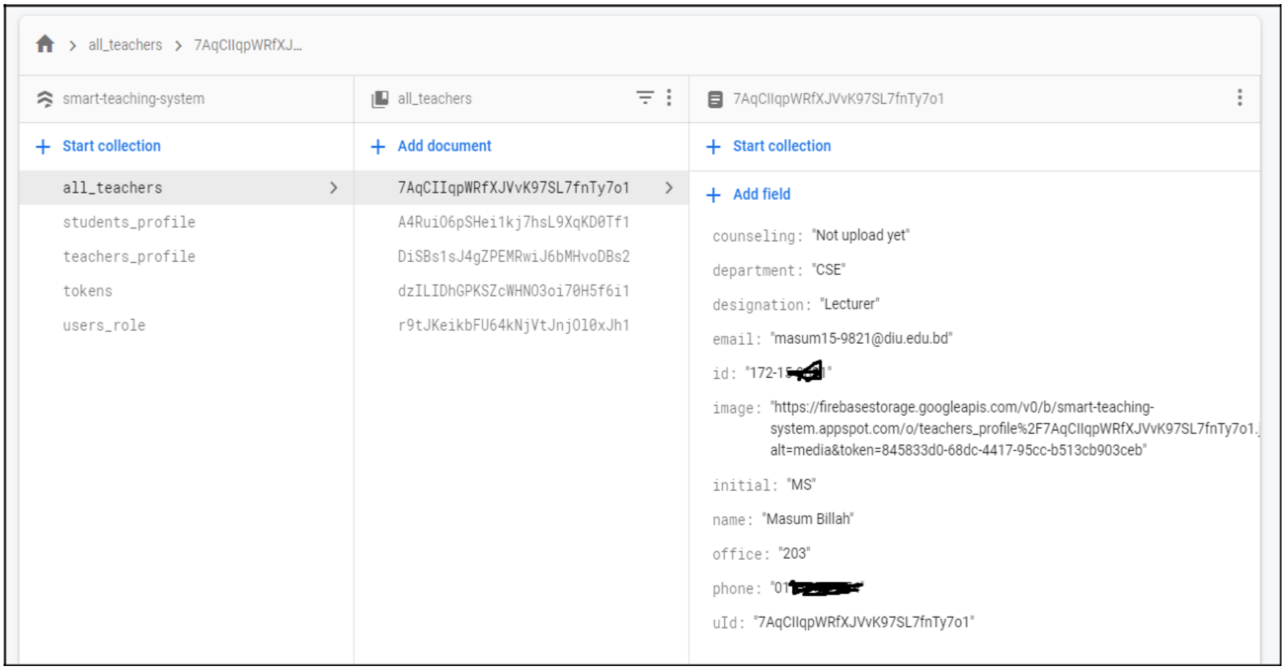
□	Name	Size	Type	Last modified
□	25cgN5vePKXS3myKUZvHRP4Nqqj2.jpg	9.12 KB	image/jpeg	May 27, 2021
□	HkwsgPvZnCSz4vyRELf8hTTif613.jpg	91.54 KB	image/jpeg	May 22, 2021
□	Zii0Jet2nMcaWCtly1Nt5tkuwal2.jpg	9.06 KB	image/jpeg	Mar 16, 2021
□	l7q3gjP04TQGMfsXbwPowcfFz6p1.jpg	197.15 KB	image/jpeg	May 4, 2021
□	pOU8upL7QFdNzF6Ym7oNQJsZ0N83.jpg	235.09 KB	image/jpeg	May 3, 2021
□	rGk7cq0HOYNqlApWtsnt7YthYHo1.jpg	145.49 KB	image/jpeg	Apr 2, 2021

Figure 5.1.2.1: Firebase Storage

➢

5.1.3 Firebase Cloud Firestore:

We used Firebase cloud firestore which is the upgrade version of firebase real time database. It is easy the execute complex query. We created different directory for manage our all data in this application.



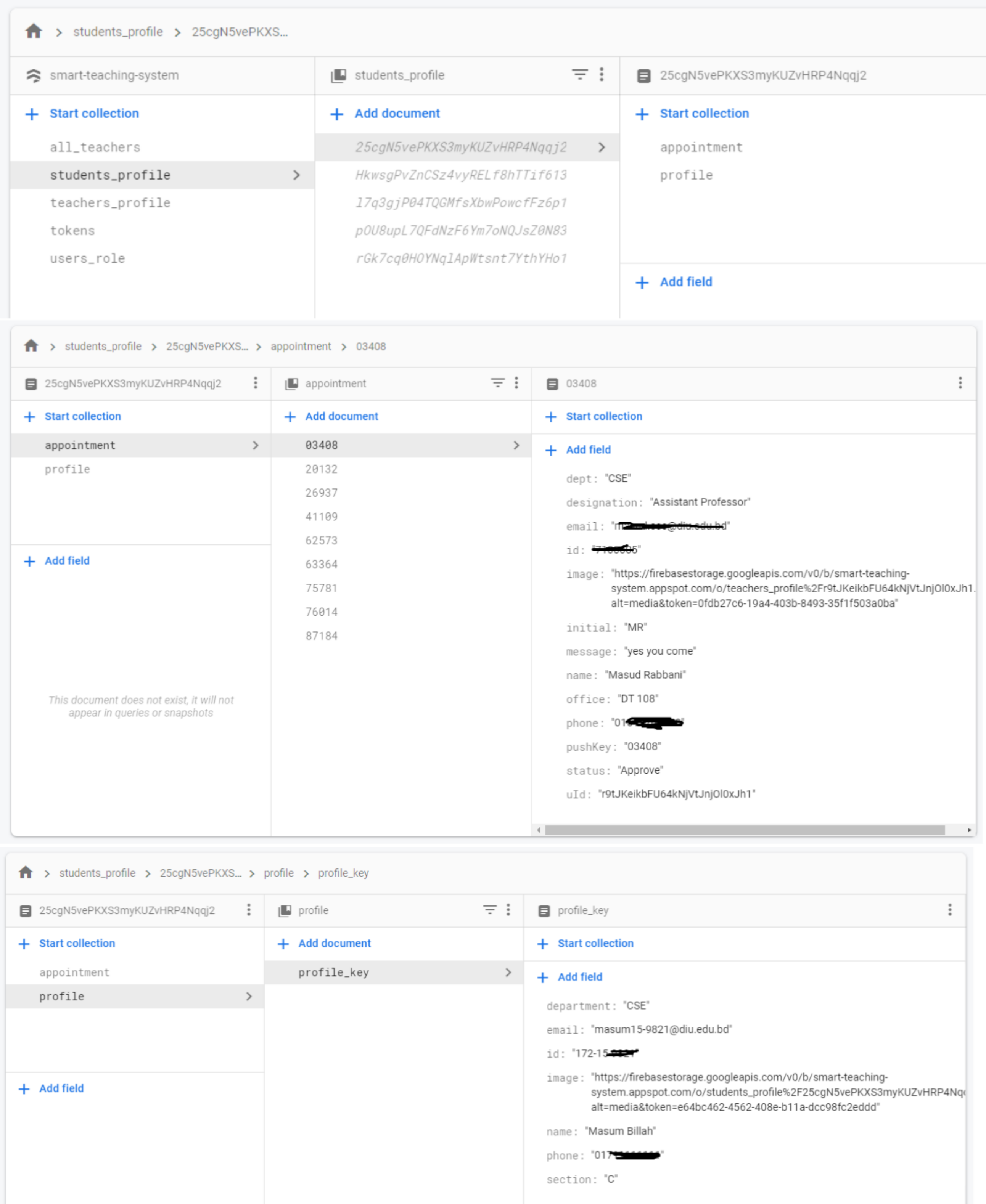
```
all_teachers
students_profile
teachers_profile
tokens
users_role
```

```
7AqCIIqpWRfXJVvK97SL7fnTy7o1
A4Rui06pSHei1kj7hsL9XqKD0Tf1
DiSBs1sJ4gZPEMRwiJ6bMHvoDBs2
dzILIDhGPKSZcWHNO3oi70H5f6i1
r9tJKeikbFU64kNjVtJnj0l0xJh1
```

```
counseling: "Not upload yet"
department: "CSE"
designation: "Lecturer"
email: "masum15-9821@diu.edu.bd"
id: "172-15[redacted]"
image: "https://firebasestorage.googleapis.com/v0/b/smart-teaching-
        system.appspot.com/o/teachers_profile%2F7AqCIlqpWRfXJVvK97SL7fnTy7o1.
        alt=media&token=845833d0-68dc-4417-95cc-b513cb903ceb"
initial: "MS"
name: "Masum Billah"
office: "203"
phone: "01[redacted]
uId: "7AqCIlqpWRfXJVvK97SL7fnTy7o1"
```

Figure 5.1.3.1: Firestore directory “all_teacher” where store all teacher information.

Here “student_profile” and “teacher_profile” directories data storing process are same



students_profile > 25cgN5vePKXS...

smart-teaching-system
+ Start collection
all_teachers
students_profile
teachers_profile
tokens
users_role

students_profile
+ Add document
25cgN5vePKXS3myKUZvHRP4Nqqj2
HkwsgPvZnCSz4vyRELf8hTTif613
l7q3gjP04TQGMfsXbwPowcfFz6p1
pOU8upL7QFdNzF6Ym7oNQJsZ0N83
rGk7cq0HOYNqlApWtsnt7YthYHo1

25cgN5vePKXS3myKUZvHRP4Nqqj2
+ Start collection
appointment
profile
+ Add field

students_profile > 25cgN5vePKXS... > appointment > 03408

25cgN5vePKXS3myKUZvHRP4Nqqj2
+ Start collection
appointment
profile
+ Add field
This document does not exist, it will not appear in queries or snapshots

appointment
+ Add document
03408
20132
26937
41109
62573
63364
75781
76014
87184

03408
+ Start collection
+ Add field
dept: "CSE"
designation: "Assistant Professor"
email: "[redacted]"
id: [redacted]5"
image: "https://firebasestorage.googleapis.com/v0/b/smart-teaching-system.appspot.com/o/teachers_profile%2Fr9tJKeikbFU64kNjVtJnjOl0xJh1.
alt=media&token=0fdb27c6-19a4-403b-8493-35f1f503a0ba"
initial: "MR"
message: "yes you come"
name: "Masud Rabbani"
office: "DT 108"
phone: "01[redacted]
pushKey: "03408"
status: "Approve"
uId: "r9tJKeikbFU64kNjVtJnjOl0xJh1"

students_profile > 25cgN5vePKXS... > profile > profile_key

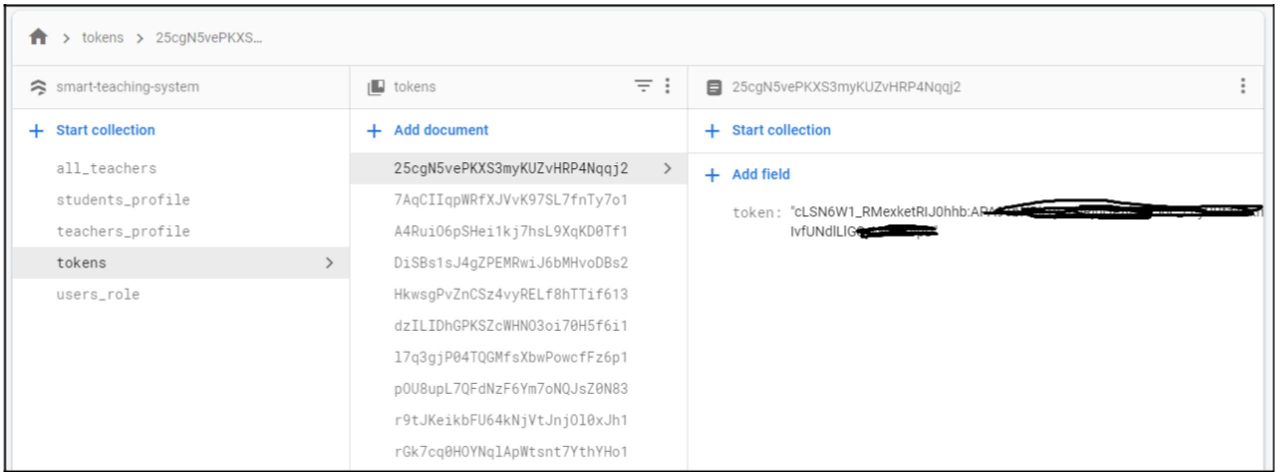
25cgN5vePKXS3myKUZvHRP4Nqqj2
+ Start collection
appointment
profile
+ Add field

profile
+ Add document
profile_key

profile_key
+ Start collection
+ Add field
department: "CSE"
email: "masum15-9821@diu.edu.bd"
id: "172-15[redacted]
image: "https://firebasestorage.googleapis.com/v0/b/smart-teaching-system.appspot.com/o/students_profile%2F25cgN5vePKXS3myKUZvHRP4Nq
alt=media&token=e64bc462-4562-408e-b11a-dcc98fc2eddd"
name: "Masum Billah"
phone: "017[redacted]"
section: "C"

Figure 5.1.3.2: Firestore directory “student_profile” where store all student profile information and appointment details.

In the “tokens” directory store all user device access token. It is updating all time based on multiple devices. Which helps to sent notification for both user teacher and student. And the “user_role” directory store every user type; a user is teacher or student.



tokens > 25cgN5vePKXS...

smart-teaching-system

+ Start collection

all_teachers
students_profile
teachers_profile
tokens
users_role

tokens

+ Add document

25cgN5vePKXS3myKUZvHRP4Nqqj2
7AqCIIqpWRfXJVvK97SL7fnTy7o1
A4Rui06pSHei1kj7hsL9XqKD0Tf1
DiSBs1sJ4gZPEMRwiJ6bMHvoDBs2
HkwsgPvZnCSz4vyRELf8hTTif613
dzILIDhGPKSZcWHNO3oi70H5f6i1
17q3gjP04TQGMfsXbwPowcfFz6p1
pOU8upL7QFdNzF6Ym7oNQJsZ0N83
r9tJKeikbFU64kNjVtJnj010xJh1
rGk7cq0HOYNqlApWtsnt7YthYHo1

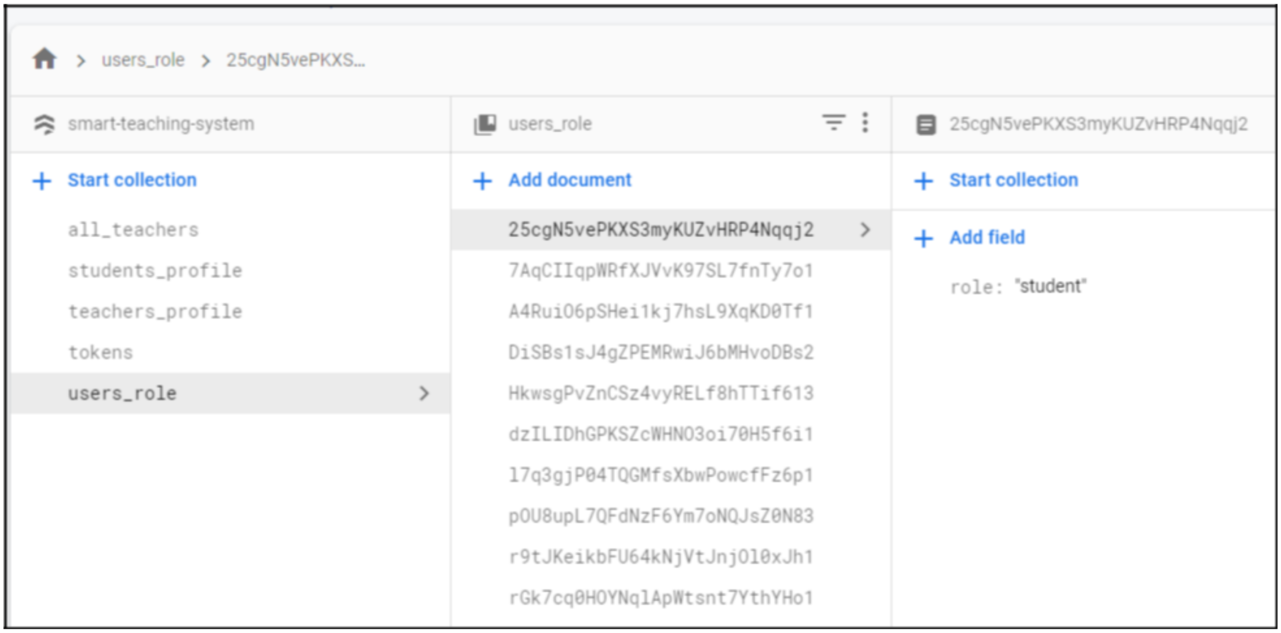
25cgN5vePKXS3myKUZvHRP4Nqqj2

+ Start collection

+ Add field

token: "cLSN6W1_RMexketRIJ0hhb:AP [redacted]
IvfUNdILIG [redacted]

Figure 5.1.3.3: Firestore directory “tokens” where store all users device token



users_role > 25cgN5vePKXS...

smart-teaching-system

+ Start collection

all_teachers
students_profile
teachers_profile
tokens
users_role

users_role

+ Add document

25cgN5vePKXS3myKUZvHRP4Nqqj2
7AqCIIqpWRfXJVvK97SL7fnTy7o1
A4Rui06pSHei1kj7hsL9XqKD0Tf1
DiSBs1sJ4gZPEMRwiJ6bMHvoDBs2
HkwsgPvZnCSz4vyRELf8hTTif613
dzILIDhGPKSZcWHNO3oi70H5f6i1
17q3gjP04TQGMfsXbwPowcfFz6p1
pOU8upL7QFdNzF6Ym7oNQJsZ0N83
r9tJKeikbFU64kNjVtJnj010xJh1
rGk7cq0HOYNqlApWtsnt7YthYHo1

25cgN5vePKXS3myKUZvHRP4Nqqj2

+ Start collection

+ Add field

role: "student"

Figure 5.1.3.4: Firestore directory “user_role” where store all users role (teacher/student)

5.2 Implementation of Front-end Design

An attractive user interface is the first priority for an application. First us focused on the UI design. We maintain our design with proper way so that in every size of device it will remain same. We handle different screen size device.

5.3 Testing Implementation, Results and Report

We test our project with different device and different case of input. Most of the case out application give the success result.

TABLE 5.3.1: Testing result

Test case	Test input	Expected outcome	Obtained outcome	Passed/ Failed	Testing Period
Install	Test in different version of android phone	Installed	Installed	Passed	20/05/21
Login	Not login without registration	Not login	Not login	Passed	20/05/21
Register	Enter invalid email	Not register	Not register	Passed	20/05/21
Login	Login without mail verification	Not login	Not login	Passed	20/05/21
Password	Enter wrong password	Invalid password	Invalid password	Passed	20/05/21
Login	After all registration	Go to home page	Go to home page	Passed	20/05/21
Registration	Register without upload image	Forced to upload image	Say to upload image	Passed	20/05/21
Submit appointment	Submit appointment to teacher without reason	Forced to enter reason	Say to enter reason	Passed	20/05/21

Delete	Delete the pending appointment for student	Not delete	Not delete	Passed	20/05/21
Delete	Delete the request appointment without accept/decline	Not delete	Not delete	Passed	20/05/21
Authentication	If user already logged in not need to login when open this application again	Go to home page without login	Go to home page without login	Passed	20/05/21
Logout	After logout action	Go to login page	Go to login page	Passed	20/05/21
Internet Connection	If there is no internet connection	No Internet connection message	No Internet connection message	Passed	20/05/21

Chapter 6: Conclusion and Future Scope

6.1 Discussion and Conclusion

Finally, our application has been completed. We hope it will help the student and teacher a lot to maintain their counseling communication. Now a days mobile phone application going to popular more than another platform application. For that reason, we believed this application will granted to all student and teacher. We hope this application will be helpful for student to find information about their expected teacher. And easy to reach the teacher which they want.

6.2 Limitation

There are some limitations in our application. They are:

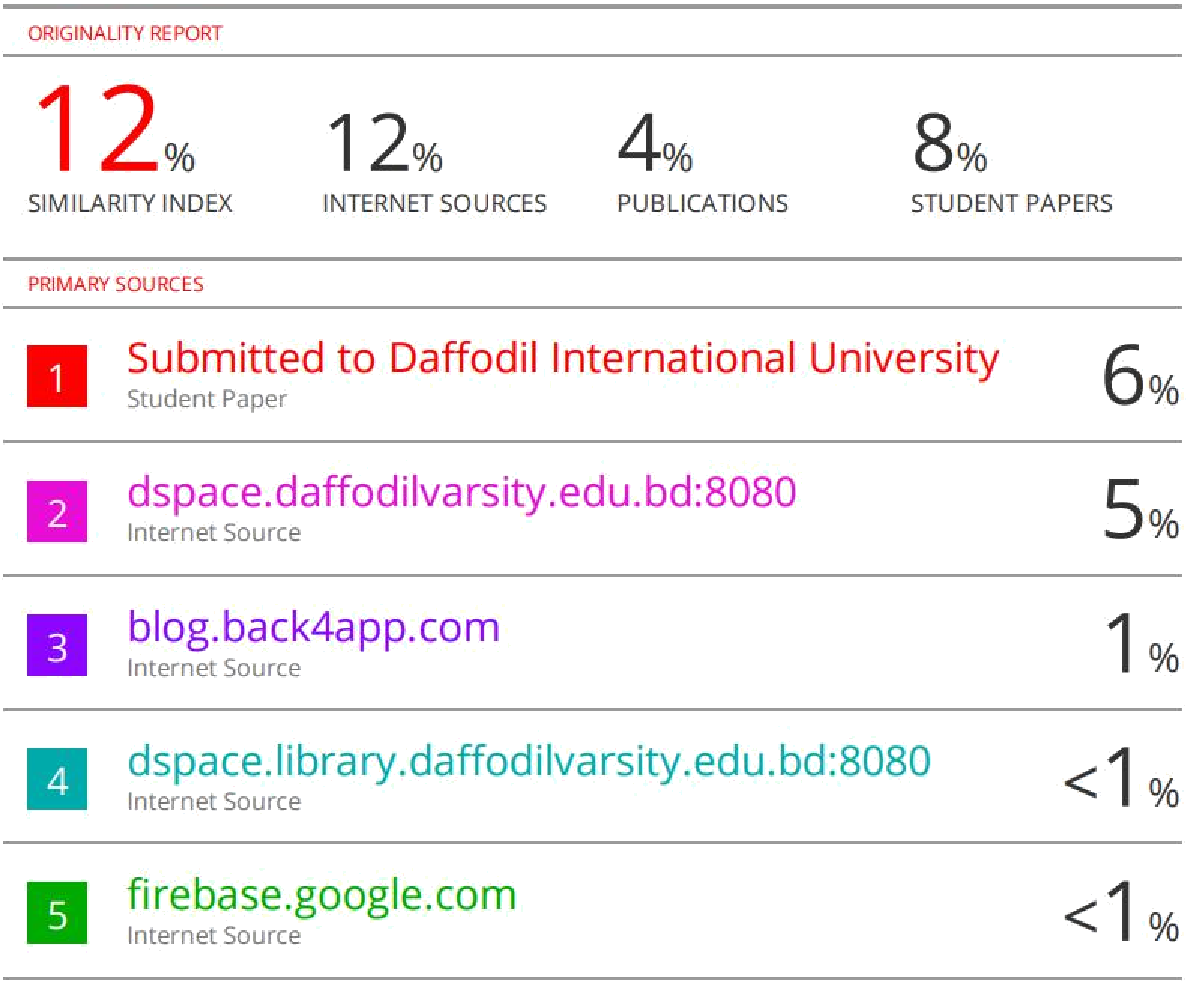
- After appointment request if a student change their information, it will not be updated in teacher dashboard
- Same as after make appointment request if teacher change his/her information it will be changed in teacher list but not changed in student home appointment list.
- Some time the notification not sent for device token related problem.

6.3 Scope for Further Developments

- Add real time chatting option between teacher and student.
- Add file share option between teacher and student.
- To make a web portal and admin panel system for more dynamic of this application.

References

1. M.H.N.M Nasir, S.H. Hamid and H. Hasan, “WebML and .NET Architecture for Developing Students Appointment Management System”, Journal of Applied Science 9(8): 1432- 1440, 2009 ISSN 1812-5654 © 2009 Asian Network for Scientific Information.
2. Al-Monte Vince M Calo, Jocelyn B Barbosa , Amparo V Dinagsao, “GrabTutor: A web and mobile application for tutor appointment system with GPS security feature”, Calo et al. / Indian Journal of Science and Technology 2020;13(19):1955–1964
3. BELLO Ridwan Oluwaseun, OLUGBEBI Muyiwa, BABATUNDE Abdulrauph Olanrewaju, BELLO Bashir Omolaran , BELLO Shakirat Iyabo, “Student-Teacher Online Booking Appointment System in Academic Institutions”, Journal of Computer Science and Control Systems Volume 9, Number 2, October 2016



ORIGINALITY REPORT

12%	12%	4%	8%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	Submitted to Daffodil International University Student Paper	6%
2	dspace.daffodilvarsity.edu.bd:8080 Internet Source	5%
3	blog.back4app.com Internet Source	1%
4	dspace.library.daffodilvarsity.edu.bd:8080 Internet Source	<1%
5	firebase.google.com Internet Source	<1%