GLA University, Mathura - 2020

Mini Project - 1

MID - TERM REPORT

Institute of Engineering and Technology

SUBMITTED BY :

Tushar Gupta (181500760)

Kartik Agrawal (181500311)

Jalanki Nayak (181500291)

Yash Vardhan Gautam (181500830)

SUPERVISED BY :

Mr. Vaibhav Diwan (Assisstant Professor)

Department of Computer Engineering & Applications

CONTENT

ABSTRACT

In this, we will build an android application that will serve both functionalities of a chatting application and an educational app. The inspiration for the chatting app has been taken from Whatsapp as it being the most popular and primary daily chat application. The inspiration for the educational part hasn't been taken from a single platform but many, primarily being the Google classroom platform and the rest is the QnA section of various online educators such as the likes of Udemy, Coursera, Udacity, etc. The building blocks of the project being Java, Android, XML, Google Firebase, and the IDE is Android Studio. The app has two categories of user viz. Teacher and Student. Where a Teacher can create classrooms, add students to it, discuss doubts and give an assignment, a Student can join a classroom, download various resources provided by the teacher, submit his assignments, and can also chat with his fellow students. The choice of android as the developing environment is solely because of its wide reach and its ease of use.

INTRODUCTION

1. GENERAL INTRODUCTION

There is numerous active chating app worldwide, among those WhatsApp scores the highest rank around the globe. WhatsApp was founded by Jan Koum and Brian Acton who had previously spent 20 years combined at Yahoo. WhatsApp joined Facebook in 2014, but continues to operate as a separate app with a laser focus on building a messaging service that works fast and reliably anywhere in the world. As per the record it serves 2 billion users worldwide. WhatsApp is available in 180 countries and 60 diferent languages. Due to its many features it is covering large users.

It provides us many features to communicate and share stuffs to friends and family easily and cost efficiently. WhatsApp has many features but Groups in these Chatting Applicatons are used for informal chats and fun. Using these groups for educational purposes is not preferable and has many limitatons. To overcome the limitatons discussed above, an applicaton need to be made which contains the features of both chating applicaton and classroom. StuFun is an Chatting Application which incorporate the features of Classroom which makes it convinient for both fun and educational purpose.

2. AREA OF COMPUTER SCIENCE

This project explores the region of android app development in the area of Computer Science. The primary tool for developing is Java for android and though newer languages like Kotlin have arrived we have still gone for Java because of its proven results. The IDE for the project is Android Studio and we have used Github as our version source control.

For our database and real-time authentication needs, we have gone for the Google Firebase system as in the beginning it will provide us with a 2gb storage for each user and will process a maximum of 100 users at any moment simultaneously.

As a lot of documentation and designing comes with app development we have used Lucidchart for creating diagrams and graphs and Figma has been used to design the user interface.

3. HARDWARE AND SOFTWARE REQUIREMENTS

Hardware Requirements

- At least 8gb RAM
- Intel i5 g7 processor or equivalent

Software Requirements

- Android Studio(v3.6 or higher)
- Java JDK
- Android SDK
- Figma
- Wireframes

PROBLEM STATEMENT

INTRODUCTION AND MOTIVATION

In today's world social messaging apps like **Whatsapp, Hike, Messenaer** etc are the most popular way to connect with friends ,family and businesses. Apps like these allow us to do personal chats with a particular person or we can create groups where we can chat with a lot many people. Many businesses are now being run on WhatsApp as a result many official and professional work is now being done on such apps.

Education doesn't remain aloof from such apps; now a days such groups are also used for educational purposes where teachers create groups of their classes and can share information like class activities, pdf notes, video notes, e-books etc.

But using these groups for educatonal purposes has some limitatons. There are various cases like –

• Does the apps comply with GDPR guidelines and especially in case of student.

• Students can be added only with their mobile numbers which are generally of their parents and moreover they quickly become a pit of unknown digits and faceless profiles.

• Whenever student is having some doubt regarding a topic and he/she want to discuss it with class they share their doubts in the group and a discussion starts which most of the time goes in informal direction.

• Whenever Teacher wants to provide any information or wants to do any announcement, they post it on group and a discussion starts   which most of the time goes in informal direction.

PROPOSED SOLUTION

Groups in these Chatng Applicatons like **WhatsApp** should be used for informal chats and fun. Using these groups for educational purposes is not preferable and has many limitatons. To overcome the limitatons discussed above, an applicaton need to be made which contains the features of both chatting applicaton and classroom. This applicaton would incorporate the following features which would overcome the limitatons in groups of chating application -

➢ For fun and communicaton with friends and family, you can chat with them personally. Two users can exchange text messages, photos, voice notes and documents.

➢ A separate classroom panel will be there in the App which will overcome the limitations of Groups in chatting applications in educational fields.

➢ If the User(Teacher) have created a classroom, he/she can launch the announcements, answer the queries of Students inside Discussion Panel and can monitor the list of Users(Students) who have joined the classroom.

➢ If the User(Student) have joined an existing classroom, he/she can view the announcements and ask query from Teacher inside Discussion Panel.

Such an Applicaton must be used as digital educaton is increasing day by day. Furthermore, the applicaton must be well tested so that the user of the applicaton will have confidence while using it. he success of the project is measured by the number of features completed(implemented, tested, documented and delivered).

OBJECTIVE

We want to create anandroidapplication that will function both as a Chatting App with a professional approach and as a Classroom App.

PROJECT LIFECYCLE

- Identifying the problem and writing a problem statement

- Carry out A literature review of the project

- Making an SRS for the project

- In the SRS identifying the major subparts of the app

- Start documenting the major subparts of the app

- Design the various charts and diagrams for the documentation of the major subparts, like the following

- Use case diagrams and their respective ADRs

- Activity diagrams and their respective ADRs

- Class diagrams and their respective ADRs

- Entity-Relationship diagrams and their respective ADRs

- Dataflow diagrams and their respective ADRs

- Construct User Interface prototypes with the help of Wireframes

- Start implementing the design with one subpart at a time, the likes of which are

- Defining, Registering, and Log-in of the two user types viz. Teacher & Student

- Designing the personal chat system for the Student user type

- Designing the classroom chat and resource sharing system for the Teacher user type

- Designing and joining a classroom chat system for the Student user type

- Designing the various subparts of the classroom chat system such as the Assignment panel, the Discussion panel, and the notification panel.

- Testing the various subparts of the project that had been implemented

IMPLEMENTATION DETAILS

Our Project named StuFun is divided into 4 Sections :

1. AUTHENTICATION

Authentication section deals with Registration and Login process into the Application. It consist of User Selection Panel, Teacher Registration/Login and Student Registration/Login. For identity verification, email verification would be there before login.

2. PERSONAL CHAT

Personal Chat Section has same functionality as of other chating Application. Users would be identified by their phone number. Users can chat with each other and can send photos, documents, links via message. All the previous chats would be saved and user can see their previous chats.

3. CLASSROOM PANEL (STUDENT)

Classroom panel have different functionalities for Teacher and Student. Student can only join the classroom by entering the Class ID generated during creating Classroom. Inside the classroom, Students can view announcements and ask queries inside discussion panel.

4. CLASSROOM PANEL (TEACHER)

Teacher can only create the classroom by entering the details. A unique ID of class will be generated which will be its identity. Inside the classroom, teacher can post, delete edit announcements, answer the queries of students and can monitor the list of students inside the classroom.

Each section will be developed in the sequential manner and the project will be considered to be completed iff all 4 sections of the project are successfully developed.

PROGRESS TILL DATE & REMAINING WORK

- **PROGRESS TILL DATE**

  Till now, we have completed our documentation work (Literature Review, SRS, ADR, Wireframes) and 1st section i.e. AUTHENTICATION.

  Both Front End and Back End Portion of the Authentication Panel has been completed. We have used Google Firebase as our Data Storage (Database).

- **REMAINING WORK**

  Authentication Panel has been completed. Now we have to complete Front End and Back End portion of remaining 3 sections i.e., Personal Chat, Classroom panel for Student, Classroom panel for Teacher.

  At the completion of these remaining sections, the project will be considered as completed.

SOME SCREENSHOTS

File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help

```java
package com.example.stufun;

import ...

public class UserSelection extends AppCompatActivity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_user_selection);

        findViewById(R.id.student_select).setOnClickListener((view) -> {
            Intent intent = new Intent( packageContext: UserSelection.this,LoginActivity.class);
            intent.putExtra( name: "type", value: "Students");
            startActivity(intent);
        });
        findViewById(R.id.teacher_select).setOnClickListener((view) -> {
            Intent intent = new Intent( packageContext: UserSelection.this,LoginActivity.class);
            intent.putExtra( name: "type", value: "Teachers");
            startActivity(intent);
        });
    }
}
```

File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help

```xml
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".LoginActivity">

    <View
        android:layout_width="match_parent"
        android:layout_height="500dp"
        android:background="@drawable/oval_green"
        android:layout_marginTop="-270dp"
        />
    <ImageView
        android:layout_width="120dp"
        android:layout_height="120dp"
        android:background="@drawable/oval_white"
        android:src="@drawable/student"
        android:padding="25dp"
        android:elevation="10dp"
        android:layout_centerHorizontal="true"
        android:layout_marginTop="70dp" />
    <com.google.android.material.textfield.TextInputLayout
        android:id="@+id/user_email"
        android:layout_width="match_parent"
        android:layout_marginStart="20dp"
        android:layout_marginEnd="20dp"
```

Email address

Password

LOGIN

Not a Account Yet? Sign Up

```java
package com.example.stufun;

import ...

public class LoginActivity extends AppCompatActivity {

    private TextInputLayout login,password;
    private String type = "";

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_login);

        type = Objects.requireNonNull(getIntent().getExtras()).getString( key: "type");
        login = findViewById(R.id.user_email);
        password = findViewById(R.id.user_pass);

        findViewById(R.id.lg_btn).setOnClickListener((view) -> { validateData(); });

        findViewById(R.id.signup_txt).setOnClickListener((view) -> {
                if(type.equals("Students"))
                    startActivity(new Intent( packageContext: LoginActivity.this,StudentRegister.class));
                else
                    startActivity(new Intent( packageContext: LoginActivity.this,TeacherRegister.class));
        });
    }
```



```java
    private void validateData(){

        String e = Objects.requireNonNull(login.getEditText()).getText().toString();
        String p = Objects.requireNonNull(password.getEditText()).getText().toString();

        if(TextUtils.isEmpty(e) || TextUtils.isEmpty(p)){
            Toast.makeText( context: this, text: "Empty Entry", Toast.LENGTH_SHORT).show();
        }
        else {
            trylogin(e,p);
        }

    }

    private void trylogin(final String em, final String pass){

        DatabaseReference reference = FirebaseDatabase.getInstance().getReference().child(type);

        reference.addValueEventListener(new ValueEventListener() {
            @Override
            public void onDataChange(@NonNull DataSnapshot dataSnapshot) {
                if(dataSnapshot.child(em).exists()){
                    if(Objects.equals(dataSnapshot.child(em).child("password").getValue(), pass))
                    {
                        Intent intent = new Intent( packageContext: LoginActivity.this,HomeActivity.class);
                        intent.setFlags(Intent.FLAG_ACTIVITY_CLEAR_TASK|Intent.FLAG_ACTIVITY_NEW_TASK);
```

Android Studio — activity_teacher_register.xml

```xml
            android:layout_height="wrap_content"
            android:hint="Password"
            >
            <com.google.android.material.textfield.TextInputEditText
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                />
        </com.google.android.material.textfield.TextInputLayout>

        <com.google.android.material.textfield.TextInputLayout
            android:id="@+id/teacher_reg_cpass"
            android:layout_width="match_parent"
            android:layout_marginStart="20dp"
            android:layout_marginEnd="20dp"
            android:layout_below="@+id/teacher_reg_pass"
            android:layout_height="wrap_content"
            android:hint="Confirm Password"
            >
            <com.google.android.material.textfield.TextInputEditText
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                />
        </com.google.android.material.textfield.TextInputLayout>

        <Button
            android:layout_width="200dp"
            android:layout_height="wrap_content"
```


Android Studio — TeacherRegister.java

```java
package com.example.stufun;

import ...

public class TeacherRegister extends AppCompatActivity {
    private TextInputLayout name,email,phone,pass,cpass;
    private DatabaseReference reference;
    private String uname,uemail,uphone,upass,ucpass;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_teacher_register);

        reference = FirebaseDatabase.getInstance().getReference().child("Teachers");

        name = findViewById(R.id.teacher_reg_name);
        email = findViewById(R.id.teacher_reg_email);
        phone = findViewById(R.id.teacher_reg_phone);
        pass = findViewById(R.id.teacher_reg_pass);
        cpass = findViewById(R.id.teacher_reg_cpass);

        findViewById(R.id.teacher_registration_btn).setOnClickListener((view) -> {
                validateData();
        });
    }

    private void validateData(){
        uname = String.valueOf(Objects.requireNonNull(name.getEditText()).getText());
```

First window — StudentRegister.java:

```java
package com.example.stufun;

import ...

public class StudentRegister extends AppCompatActivity {

    private TextInputLayout name,email,phone,course,branch,section,year,pass,cpass;
    private DatabaseReference reference;
    private String uname,uemail,uphone,ucourse,uyear,usection,upass,ucpass,ubranch;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_student_register);

        reference = FirebaseDatabase.getInstance().getReference().child("Students");

        name = findViewById(R.id.reg_name);
        email = findViewById(R.id.reg_email);
        phone = findViewById(R.id.reg_phone);
        course = findViewById(R.id.reg_course);
        branch = findViewById(R.id.reg_branch);
        section = findViewById(R.id.reg_sec);
        year = findViewById(R.id.reg_year);
        pass = findViewById(R.id.reg_password);
        cpass = findViewById(R.id.reg_con_password);
```


Second window — StudentRegister.java:

```java
    private void validateData(){
        uname = String.valueOf(Objects.requireNonNull(name.getEditText()).getText());
        uemail = String.valueOf(Objects.requireNonNull(email.getEditText()).getText());
        usection = String.valueOf(Objects.requireNonNull(section.getEditText()).getText());
        uphone = String.valueOf(Objects.requireNonNull(phone.getEditText()).getText());
        ucourse = String.valueOf(Objects.requireNonNull(course.getEditText()).getText());
        ubranch = String.valueOf(Objects.requireNonNull(branch.getEditText()).getText());
        uyear = String.valueOf(Objects.requireNonNull(year.getEditText()).getText());
        upass = String.valueOf(Objects.requireNonNull(pass.getEditText()).getText());
        ucpass = String.valueOf(Objects.requireNonNull(cpass.getEditText()).getText());

        if(upass.length()<6 || !upass.equals(ucpass)){
            Toast.makeText(
                    context: StudentRegister.this, text: "Password Missmatch",Toast.LENGTH_SHORT)
                    .show();
        }
        else if(TextUtils.isEmpty(uname) || TextUtils.isEmpty(uemail) || TextUtils.isEmpty(uphone) ||
                TextUtils.isEmpty(ucpass)|| TextUtils.isEmpty(uyear)|| TextUtils.isEmpty(usection)
                || TextUtils.isEmpty(ucourse)|| TextUtils.isEmpty(upass)){
            Toast.makeText(
                    context: StudentRegister.this, text: "Password Missmatch",Toast.LENGTH_SHORT)
                    .show();
        }
        else{
            RegisterUser();
        }
    }
```

```java
        }

    }

    private void RegisterUser(){

        HashMap<String,Object> map = new HashMap<>();
        map.put("name",uname);
        map.put("email",uemail);
        map.put("phone",uphone);
        map.put("password",upass);
        map.put("year",uyear);
        map.put("section",usection);
        map.put("branch",ubranch);
        map.put("course",ucourse);

        reference.child("email").updateChildren(map).addOnCompleteListener((task) -> {
            Toast.makeText( context: StudentRegister.this,  text: "Successfully Registered", Toast.LENGTH_SHORT).show();
            Intent intent = new Intent( packageContext: StudentRegister.this,LoginActivity.class);
            intent.setFlags(Intent.FLAG_ACTIVITY_CLEAR_TASK|Intent.FLAG_ACTIVITY_NEW_TASK);
            startActivity(intent);
        });
    }
}
```