A Project Report on

Connect – Mobile Application

Submitted by

T Shoba Devi R170755
D Swetha R170338
SK Tanheera R171216

Submitted to

IIIT RK Valley
Idupulapaya, Vempalli, YSR Kadapa
Andhra Pradesh, India PIN 516330.

Under the guidance of

Ms. M.HimaBindu
Assistant Professor

as a part of
Partial fulfillment of the degree of Bachelor of Technology in
Computer Science and Engineering.

Date: 20-09-2022.

CERTIFICATE

This is to certify that the report entitled "**Connect**" submitted by T Shoba (R170755), D Swetha(R170338), SK Tanheera(R171216) in partial fulfillment of the requirements for the award of Bachelor of Technology in Computer Science and Engineering is a bonafide work carried out by them under my supervision and guidance.

The report has not been submitted previously in part or in full to this or any other University or Institution for the award of any degree or diploma.

M.HimaBindu,
Project Internal Guide,
Computer Science and Engineering,
R.K Valley, RGUKT.

P .Harinadha,
Head of the Departmet,
Computer Science andEngineering,
R.K.Valley, RGUKT.

Acknowledgement

I would like to express my sincere gratitude to ***Ms. M. HimaBindu,*** *my project internal guide for valuable suggestions and keen interest throughout the progress of my course of research.*

I am grateful to ***P Harinadha HOD CSE****, for providing excellent computing facilities and a congenial atmosphere for progressing with my project.*

At the outset, I would like to thank ***Rajiv Gandhi University of Knowledge Technologies****, for providing all the necessary resources for the successful completion of my course work.*

Table of Contents

Abstract

Our Application name is **CONNECT** which helps us to communicate with each other who are registered in this app. In this app we have three features as

1. Profile update : Here we update the profile.
2. Chat :In chats,we can send messages and receive the messages.
3. Users :It shows the registered users.

First user need to register into the application ,then it creates an account for the user. The username will be displayed on the profile tab. And there will be option to change profile image which is visible for everyone. It can be updatable anytime by the user.

Another feature which is available in the app is users. Here all the users who are registered into the application. And they can chat with any of the user available on the list just by clicking on the user.

And another main feature is Chat Feature. Here we can chat with any other user. Here we can send and recieve messages from one another. If message is delivered it shows delived message. If message is seen by the user it shows seen. We can chat with any number of people who are registered in the app. It also shows the status of the user as offline or online. If user is online is shows online as green symbol. And if user is offline it shows offline symbol as round shape attached to the profile picture.

The logout option enables users to exit from the app. It helps users to log in to another account.

1. Introduction

Our Application name is **CONNECT** which helps us to communicate with each other who are registered in this app. In this app we have three features as

1. Profile update : Here we update the profile.
2. Chat :In chats,we can send messages and receive the messages.
3. Users :It shows the registered users.

First user need to register into the application ,then it creates an account for the user. The username will be displayed on the profile tab. And there will be option to change profile image which is visible for everyone. It can be updatable anytime by the user.

Another feature which is available in the app is users. Here all the users who are registered into the application. And they can chat with any of the user available on the list just by clicking on the user.

And another main feature is Chat Feature. Here we can chat with any other user. Here we can send and recieve messages from one another. If message is delivered it shows delived message. If message is seen by the user it shows seen. We can chat with any number of people who are registered in the app. It also shows the status of the user as offline or online. If user is online is shows online as green symbol. And if user is offline it shows offline symbol as round shape attached to the profile picture.

The logout option enables users to exit from the app. It helps users to log in to another account. A native application is created for use on a platform like mobile and tablets.

The mobile app development industry is going through a transformative phase. With the advancement in micro-processing technologies, you will be able to run mobile applications on multiple platforms. For example, apps built for mobiles will run seamlessly on desktops in the coming years. Moreover, mobile apps built with Flutter or React Native will work on Android Phones, iPhones, Macs, as well as PCs. But before we get more in-depth, let’s dig into the mobile app usage statistics to understand how users are spending their time using mobile phones, and what are the opportunities in building your own mobile application.

In social messaging applications, the reliable and robust chat feature is the essential part. In this post, you’ll learn about how to build your own real-time Android WhatsApp project with Jetpack Compose and Stream SDK.

1.1 Android App Development

Mobile app development is a process for building mobile applications that run on mobile devices. These applications can either be pre-installed or downloaded and installed by the user later. They use the network capabilities of the device to work computing resources remotely. Hence, the mobile app development process requires creating software that can be installed on the device, and enabling backend services for data access through APIs, and testing the application on target devices.

To develop scalable mobile apps, you also need to consider screen sizes, hardware requirements, and many other aspects of the app development process. With an increasing number of jobs in the mobile app development industry, it is essential that the process is well defined and understood by entrepreneurs, startups, and especially developers.

1.1.1 Mobile App Development Lifecycle 2021

There are over 3.5 Billion Software Users worldwide, so there is no doubt that the industry is healthy and thriving. Stats are growing steadily, without any indications of slowing down. And studies show that an average American checks their phone at least once every twelve minutes, and over 10% of these people check their phone about every four minutes. There are some more statistics to keep in mind.

1.1.2 Mobile App Development Platforms

The two most important mobile app platforms are iOS from Apple Inc. and Android from Google. iOS is Apple's proprietary mobile operating system built specifically for iPhones. Android, however, runs on mobile devices manufactured by various OEMs, including Google.

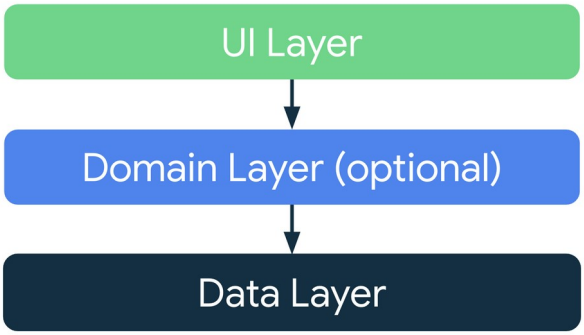
While there are many similarities between the two, however different software development kits (SDKs) are used for different platforms. Apple uses iOS exclusively for their own devices, while Google has made Android available for other companies that meet specific requirements. Developers have built over 1.5 million applications for both platforms to date.

1.2 App Architecture

Fundamentally, the architecture is composed of two layers: **UI Layer** and a **Data Layer.**

As Android apps grow in size, it's important to define an architecture that allows the app to scale, increases the app's robustness, and makes the app easier to test.

An app architecture defines the boundaries between parts of the app and the responsibilities each part should have. In order to meet the needs mentioned above, you should design your app architecture to follow a few specific principles.



1.2.1 UI Layer

Connect is built with 100% Jetpack Compose to draw UI elements, and it configures screens by observing UI states, which come from the ViewModel that holds UI states and restores data when configuration changes.

The ViewModel transforms the business data from the data layer into UI states, and UI elements configure screens following the UI states for success, loading, or error.

UI states represent the business data or exception following the single source of truth principle, so you can focus on how to draw UI elements depending on the UI states in the UI layer.

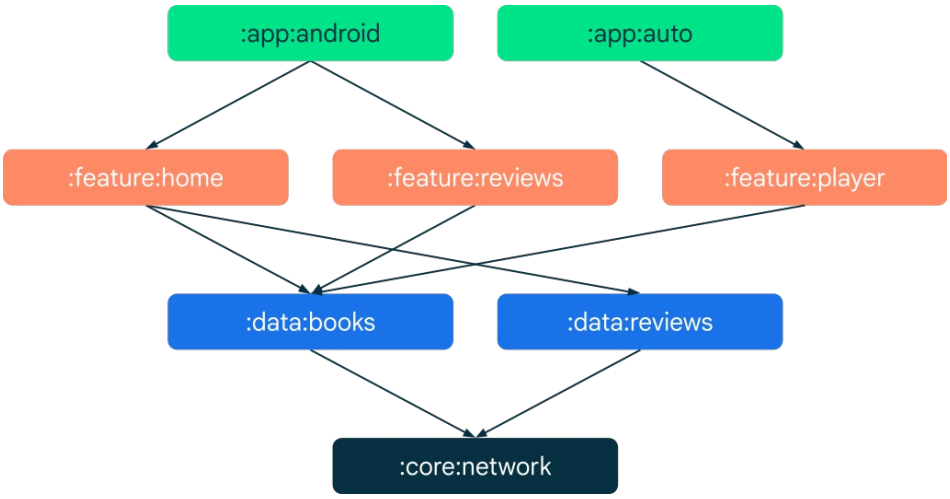
1.2.2 Data Layer

The data layer consists of repositories, which are exposed to other layers as public interfaces. Repositories include business logic that is mostly run on the background thread such as querying data from local databases and fetching remote data from the network.

1.3 Modularization

Connect App is built with multi-module strategies to improve app development.

Modularization is a practice of organizing a codebase into loosely coupled and self contained parts. Each part is a module. Each module is independent and serves a clear purpose. By dividing a problem into smaller and easier to solve subproblems, you reduce the complexity of designing and maintaining a large system.



Reusability: Modularizing reusable code properly enable opportunities for code sharing and restrict code access from other modules.

Parallel Building: Each module can be run in parallel and it reduces the build time.

Decentralize Team Focusing: Each developer team can assign their dedicated module and they can focus on their own modules.

Profile Feature:

A user profile is a collection of settings and information associated with a user. It contains critical information that is used to identify an individual, such as their name, age, portrait photograph and individual characteristics such as knowledge or expertise

Chat Feature:

Online chat may refer to any kind of communication over the Internet that offers a real-time transmission of text messages from sender to receiver. Chat messages are generally short in order to enable other participants to respond quickly

Users:

A user is a person who utilizes a computer or network service. A user often has a user account and is identified to the system by a username.

1.4 Adding Firebase

There are various services offered online such as storage, online processing, realtime database, authorisation of user etc. Google developed a platform called **Firebase** that provide all these online services. It also gives a daily analysis of usage of these services along with the details of user using it.

There are **two ways to add firebase** to any Android app:

1.4.1 Using Firebase Assistant

1. Update the android studio (>= 2.2)

2. Create a **new project** in the firebase by clicking on the Add project.

3. Now open the android studio and click on Tools in the upper left corner.

4. Now click on the Firebase option in the drop down menu.

5. A menu will appear on the right side of screen. It will show various **services** that Firebase offers. Choose the desired service.

6. Now Click on the **Connect to Firebase** option in the menu of desired service.

7. Add the dependencies of your service by clicking on **Add [YOUR SERVICE NAME] to the app option**.

1.4.2 Manually adding firebase

Create a firebase project

- Create a project by clicking on **create project** in the firebase console.
- Fill the necessary details in the pop up window about the project. Edit the project ID if required.
- Click on create project to finally create it.

Now add this project to the android app

Click on the **Add firebase to your android app** option on the starting window.

A prompt will open where to enter the package name of the app.

- Now the app is connected to the Firebase. Now all the cloud based as well server based services can be easily used in the app.
- Now the app will be registered with firebase.

Also, the SHA1 certificate, gradle>root folder>tasks>Android>signing report>copy SHA1 from console

Now download the **google-services.json** file and place it in the root directory of the android app.

Adding the sdk in the project.

PROJECT-LEVEL *build.gradle*

classpath 'com.google.gms:google-services:4.0.0'

APP-LEVEL *build.gradle*

compile 'com.google.firebase:firebase-core:16.0.0'

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

Now **Sync** the gradle by clicking on sync now.

After adding the above code(sdk), run the app to send the verification to the Firebase console.

1.5 Configure your build

The Android build system compiles app resources and source code, and packages them into APKs or Android App Bundles that you can test, deploy, sign, and distribute. Android Studio uses Gradle an advanced build toolkit, to automate and manage the build process, while allowing you to define flexible custom build configurations. Each build configuration can define its own set of code and resources, while reusing the parts common to all versions of your app. The Android plugin for Gradle works with the build toolkit to provide processes and configurable settings that are specific to building and testing Android applications.

1.5.1 Syncing project with Gradle files

When you make changes to the build configuration files in your project, Android Studio requires that you sync your project files so that it can import your build configuration changes and run some checks to make sure your configuration won't create build errors.

1.6 Testing the Developed App

After successfully developing an application, it is necessary that the quality of the application is ensured to be on-point. Quality assurance is a crucial phase in the mobile application development process as it determines the reliability, stability, and usability of the developed application. In order to ensure an all-inclusive testing process, there are a number of aspects that need to be addressed by following a complete testing cycle subjective to each application.

Testing can be broadly classified into two categories, manual testing, and automated testing. It depends on the type of application whether it needs manual testing or automated testing can give accurate results.

Any application must pass through a myriad of testing methods to come up with a perfect application. Some major testing methods that are a must-do for all the mobile applications are,

Functional Testing

- Installation & initialization of the application on all the distribution channels
- Business features and functionality testing
- Testing fields, parameters, and user feedback fields
- Testing any possible interruptions
- Testing necessary device resources
- Testing possible updates for each distribution channel

Performance Testing

- Volume testing to check the app's performance under a high volume of data
- Load testing to check the speed of the app when subjected to normal and extreme load
- Stability testing to see whether the app behaves normally under all conditions
- Testing the response time of your application in all conditions

Recovery Testing

Testing the application's failure mechanisms in case of software issues

Usability testing– To make sure that the mobile app is easy to use and provides a satisfactory user experience to the customers

- **Compatibility testing**– Testing of the application in different mobiles devices, browsers, screen sizes and OS versions according to the requirements.
- **Interface testing**– Testing of menu options, buttons, bookmarks, history, settings, and navigation flow of the application.
- **Services testing**– Testing the services of the application online and offline.
- **Low-level resource testing**: Testing of memory usage, auto-deletion of temporary files, local database growing issues known as low-level resource testing.
- **Performance testing**– Testing the performance of the application by changing the connection from 2G, 3G to WIFI, sharing the documents, battery consumption, etc.
- **Operational testing**– Testing of backups and recovery plan if a battery goes down, or data loss while upgrading the application from a store.
- **Installation tests–** Validation of the application by installing /uninstalling it on the devices.
- **Security Testing**– Testing an application to validate if the information system protects data or not.

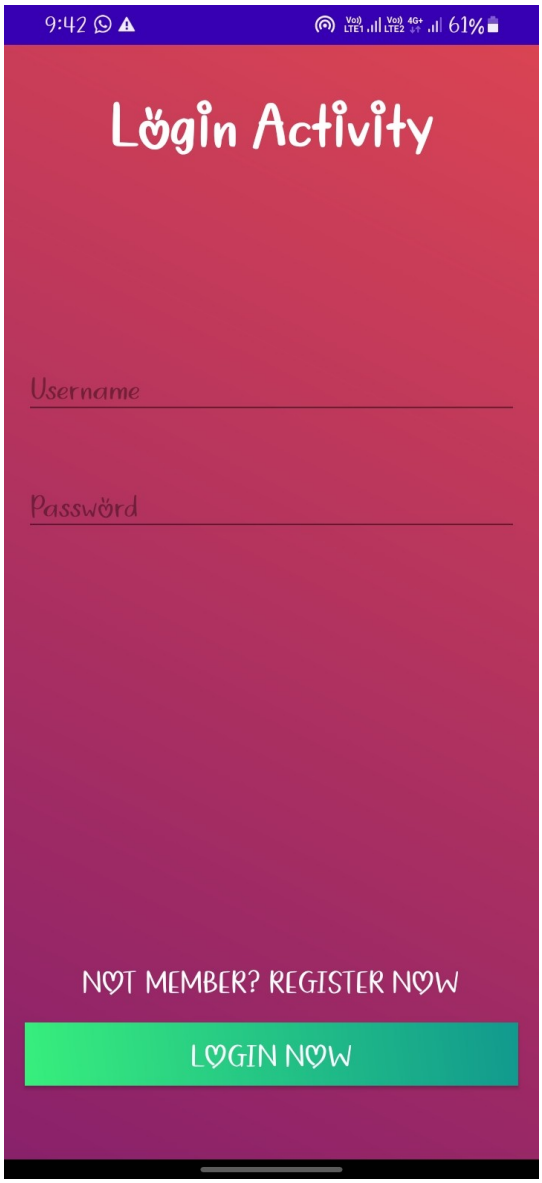
1.7 Deployment and Maintenance

Once you have built and tested the mobile application, it is time to deploy it, and maintain it for further development.

2. Working Process

2.1 Starting of the application

When user opens the app the first page which appears is as follows.



2.2 Registration Process

If User is new to the application then the user needs to register into the app.

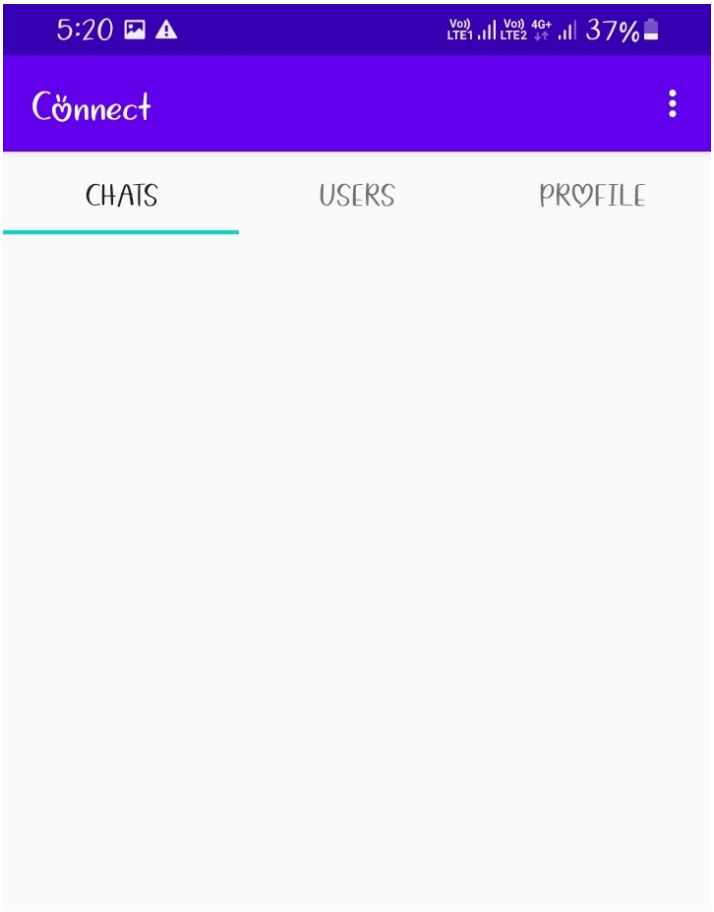
Registration Page:

It is Registration Page.Here the user register into the account with their details.

Here in this page it asks Username,Password and Personal Email Id's.User need to enter valid Username,Password and Email Id .Otherwise, it displays incorrect details.

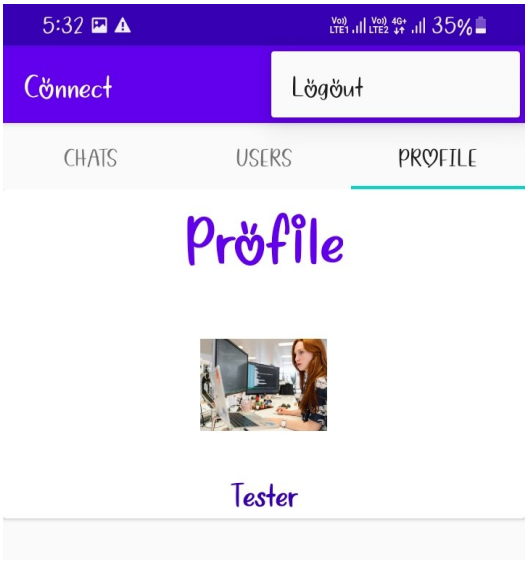
2.3 Home Page

When user logins to the app,Home Page is displayed as follows



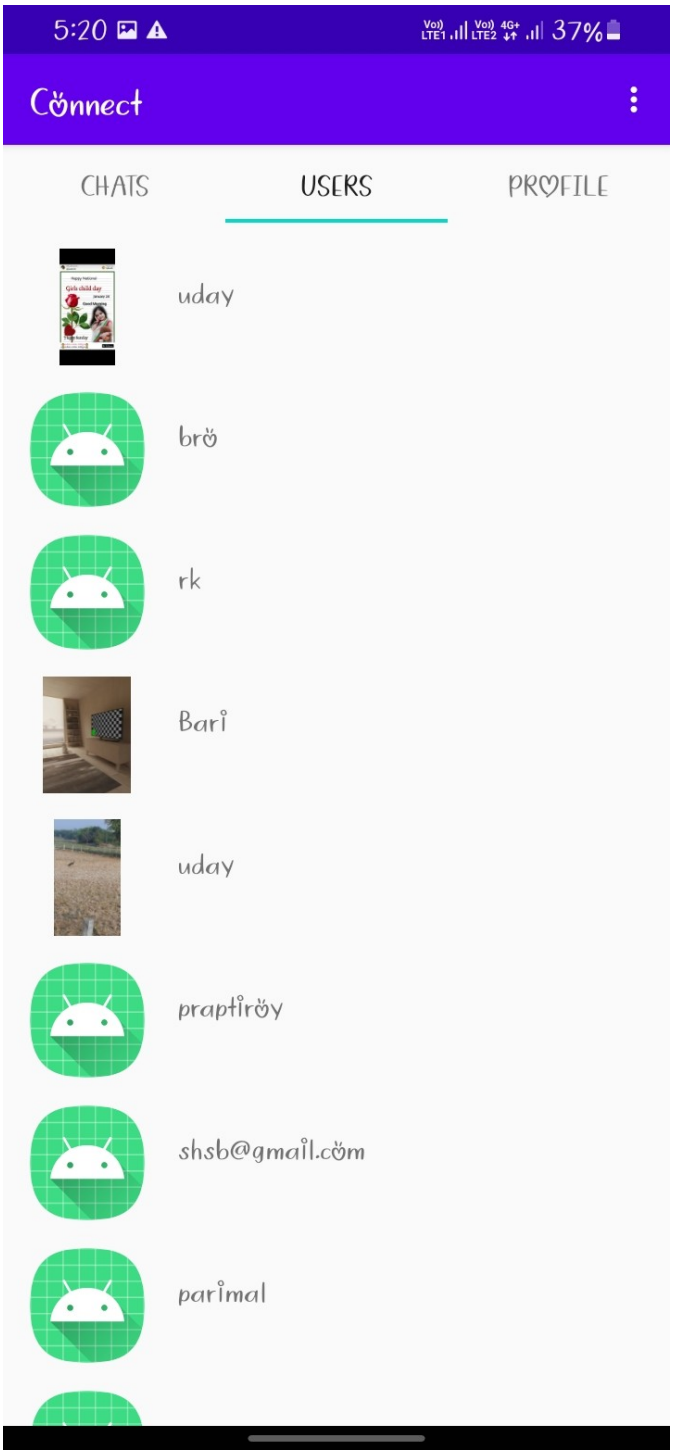
2.4 Profile

It is Profile Page.Here we can update profile picture.After updating the profile picture,it is displayed on the top of the chat page.



2.5 Users

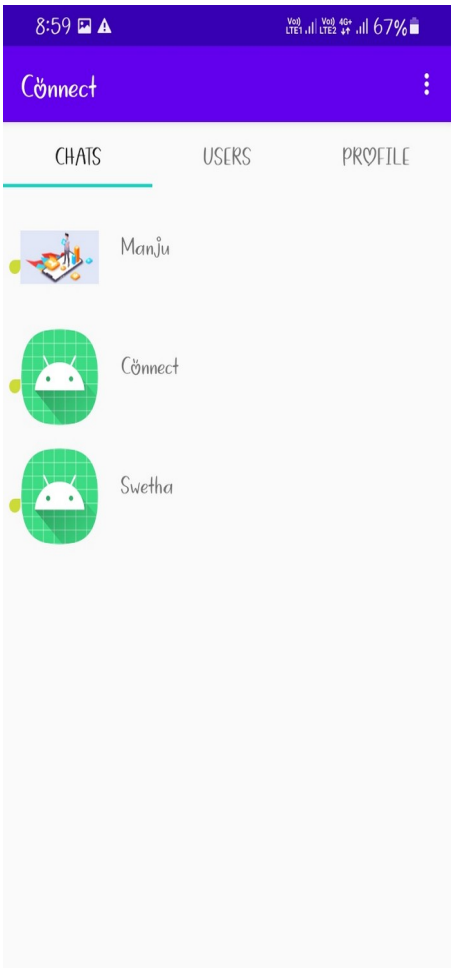
Here are all the list of users who are registered in the application. Any user can click on the name of other users to chat with them. It makes communication easy anytime anywhere.

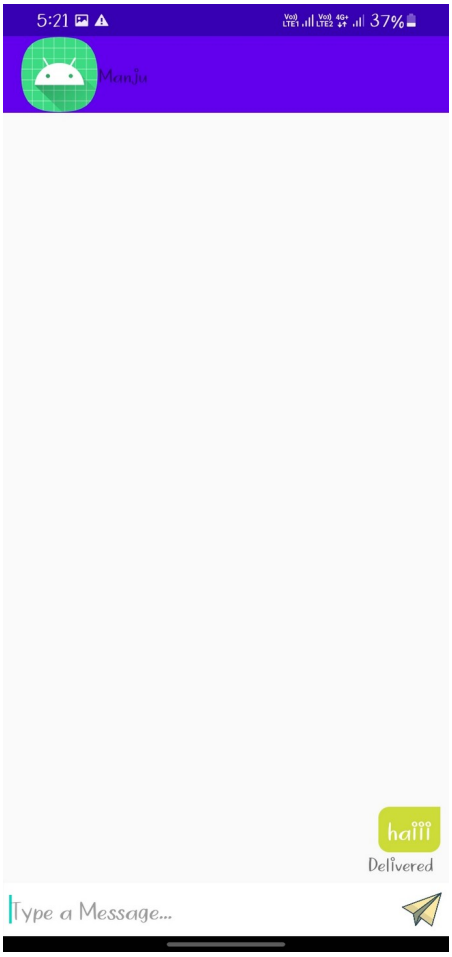

5:20
37%
Connect
CHATS
USERS
PROFILE
uday
brö
rk
Bari
uday
praptiröy
shsb@gmail.com
parimal

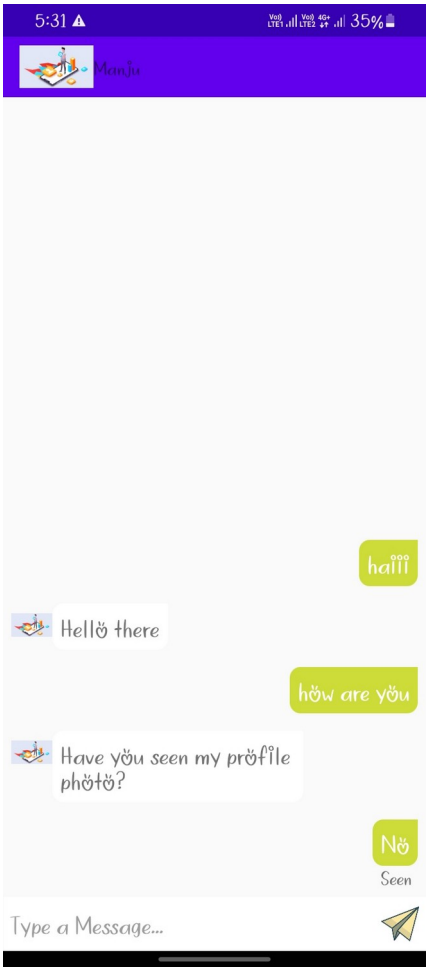
2.6 Chats

Here all the recent chats will be displayed.

Chats **Delivered** **Seen**







2.7 logout

If user wants to logout logout option is available. If user wants to login into another account also. And for the safety and security user can also be logout from his/her account.

3. Tools

3.1 Android Studio

Android is a complete set of software for mobile devices such as tablet computers, notebooks, smartphones, electronic book readers, set-top boxes etc.

It contains a **linux-based Operating System**, **middleware** and **key mobile applications**.

It can be thought of as a mobile operating system. But it is not limited to mobile only. It is currently used in various devices such as mobiles, tablets, televisions etc.

Android Studio is a new and fully integrated development environment, which has been recently launched by Google for the Android operating system. It has been designed to provide new tools for app development and to provide an alternative to Eclipse, currently the most widely used IDE.

When you begin a new project in Android studio, the project's structure will appear with almost all the files held within the SDK directory, this switch to a Gradle based management system offers an even greater flexibility to the build process.

Android Studio allows you to see any visual changes you make to your app in real-time, and you can also see how it will look on a number of different Android devices, each with different configurations and resolutions, simultaneously.

Another feature in Android Studio are the new tools for the packing and labelling of code. These let you keep on top of your project when dealing with large amounts of code. The programme also uses a drag & drop system to move the components throughout the user interface.

3.2 Android studio offers

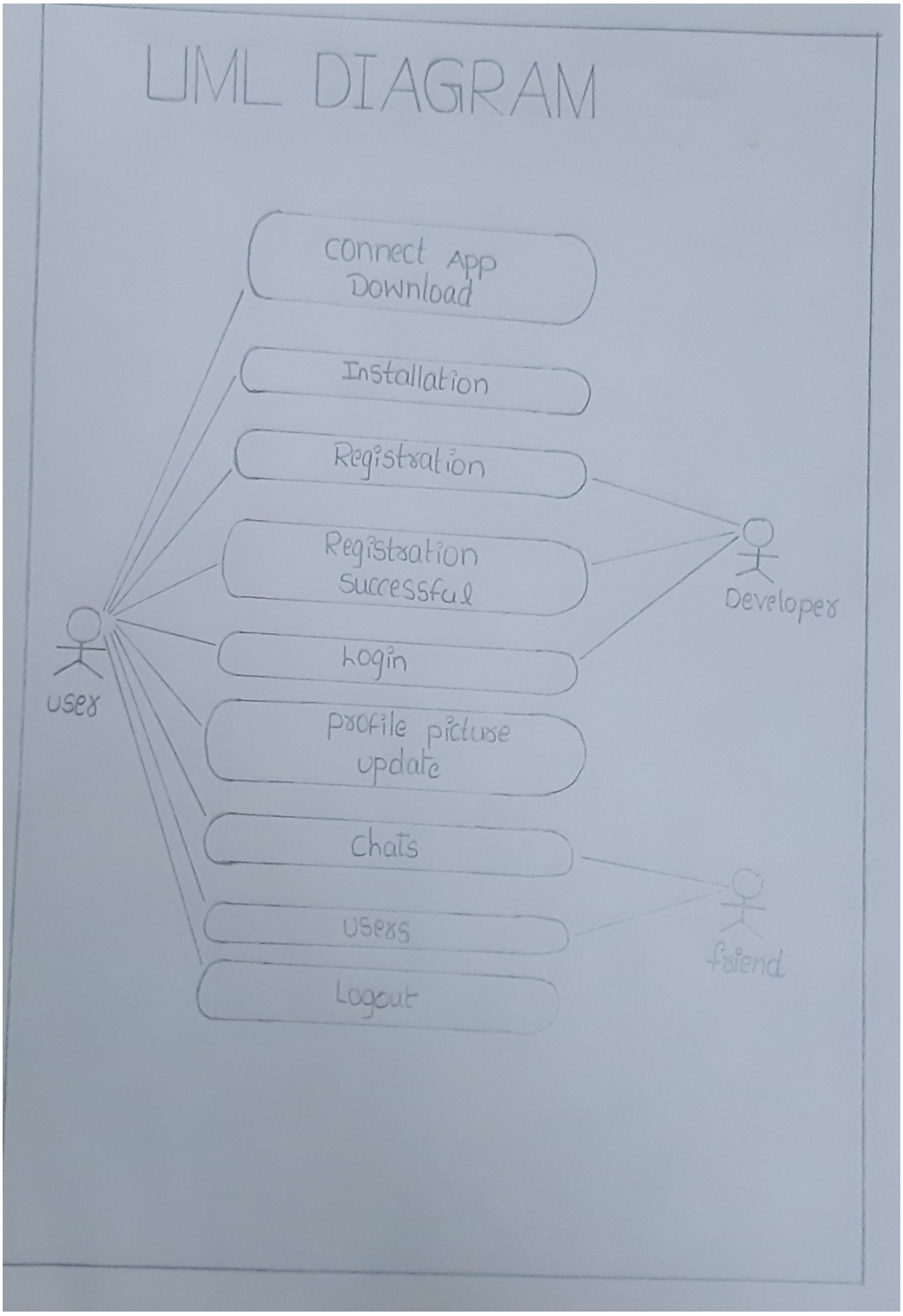
-A robust and straight forward development environment.
- An easy way to test performance on other types of device.
- Wizards and templates for common elements found in all Android programming.
- A full-featured editor with lots of extra tools to speed up the development of your applications.

To install Android Studio, it is necessary to have Android's Software Developer Kit (SDK), along with Java Developer Kit (JDK), included in this pack.

4. Implementation

4.1 UML Diagram:

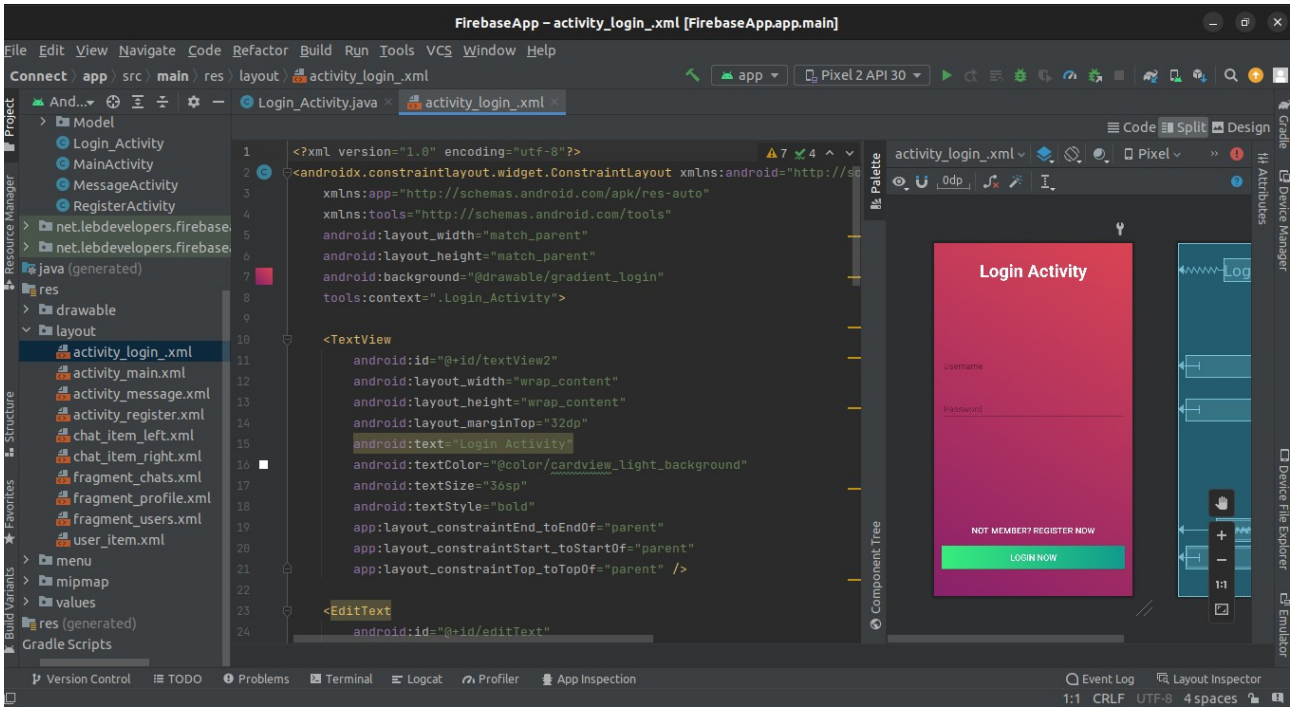
A UML diagram is a diagram based on the UML (Unified Modeling Language) with the purpose of **visually representing a system along with its main actors, roles, actions, artifacts or classes**, in order to better understand, alter, maintain, or document information about the system.

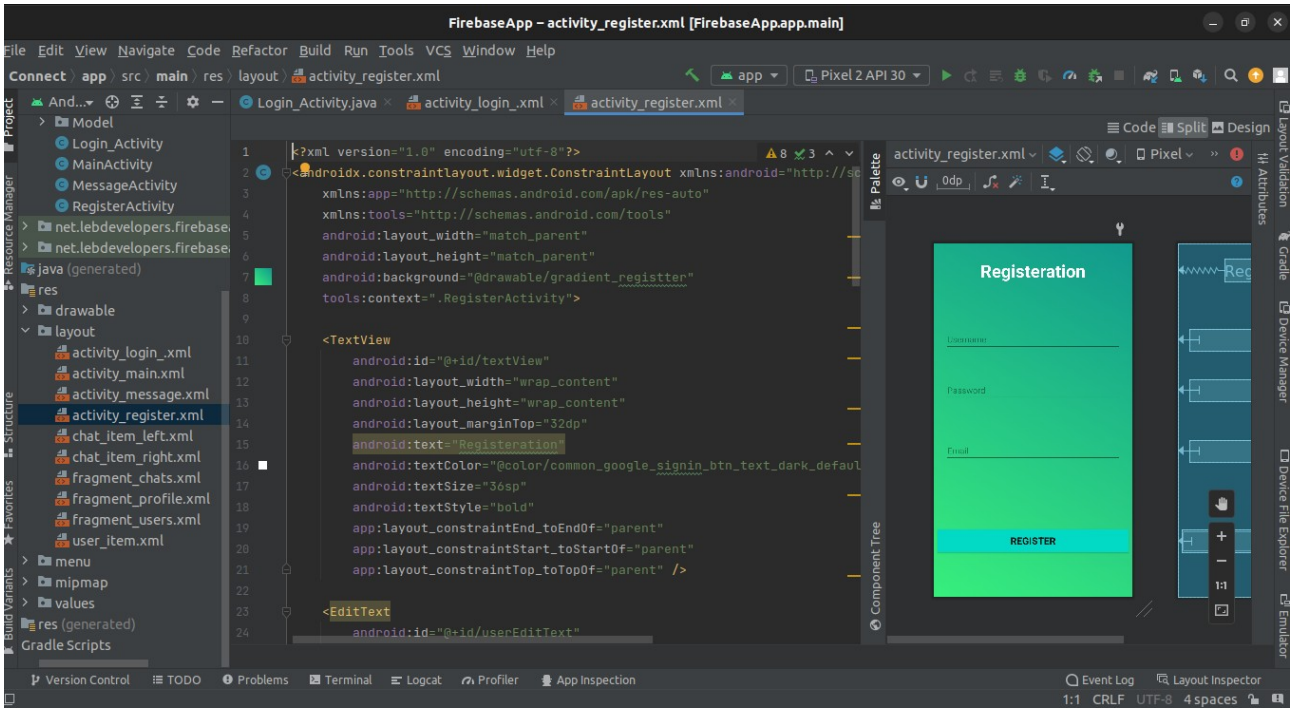


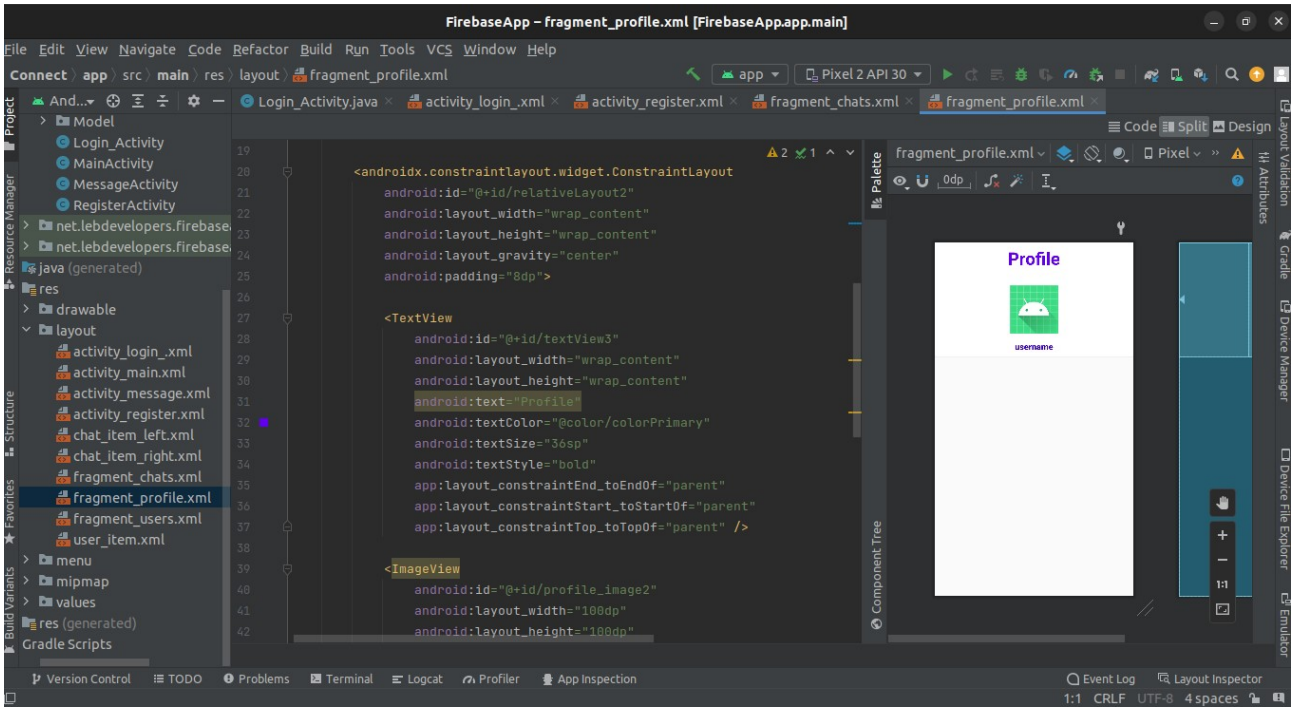
4.2 Design & Implementation

4.2.1 UI Code

In the industrial design field of human–computer interaction, a user interface is the space where interactions between humans and machines occur.

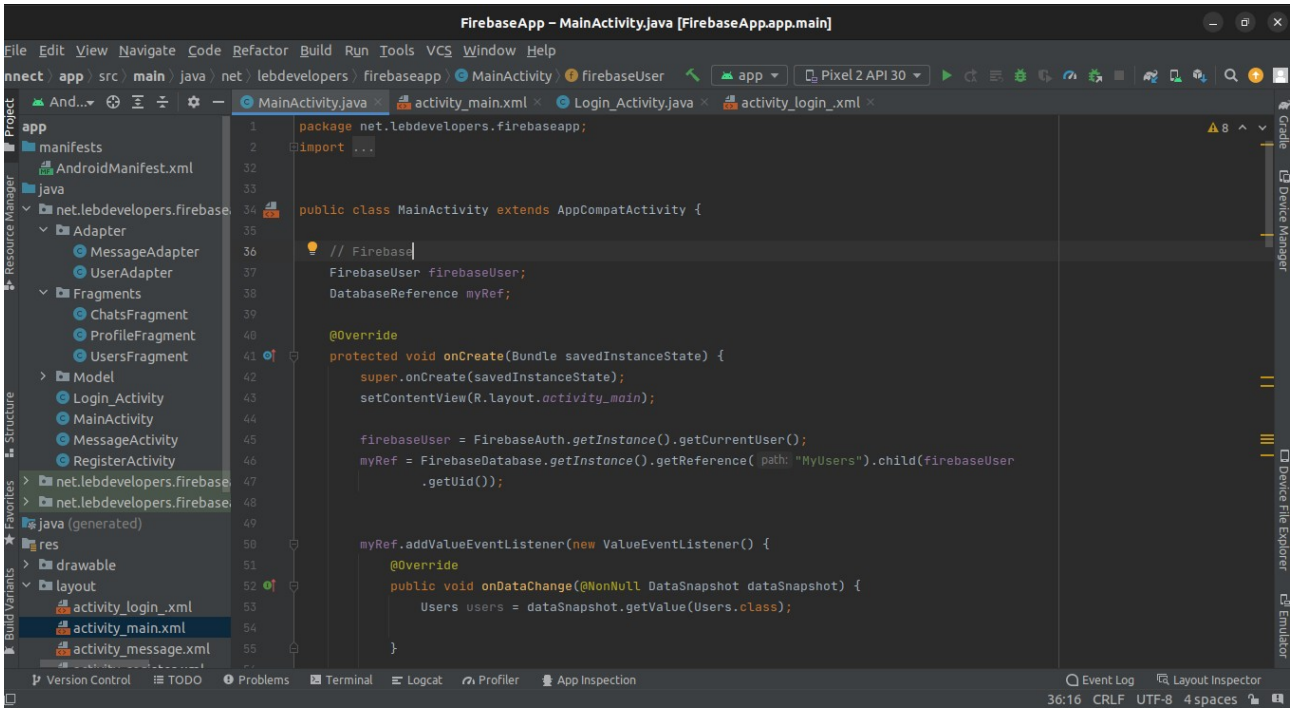


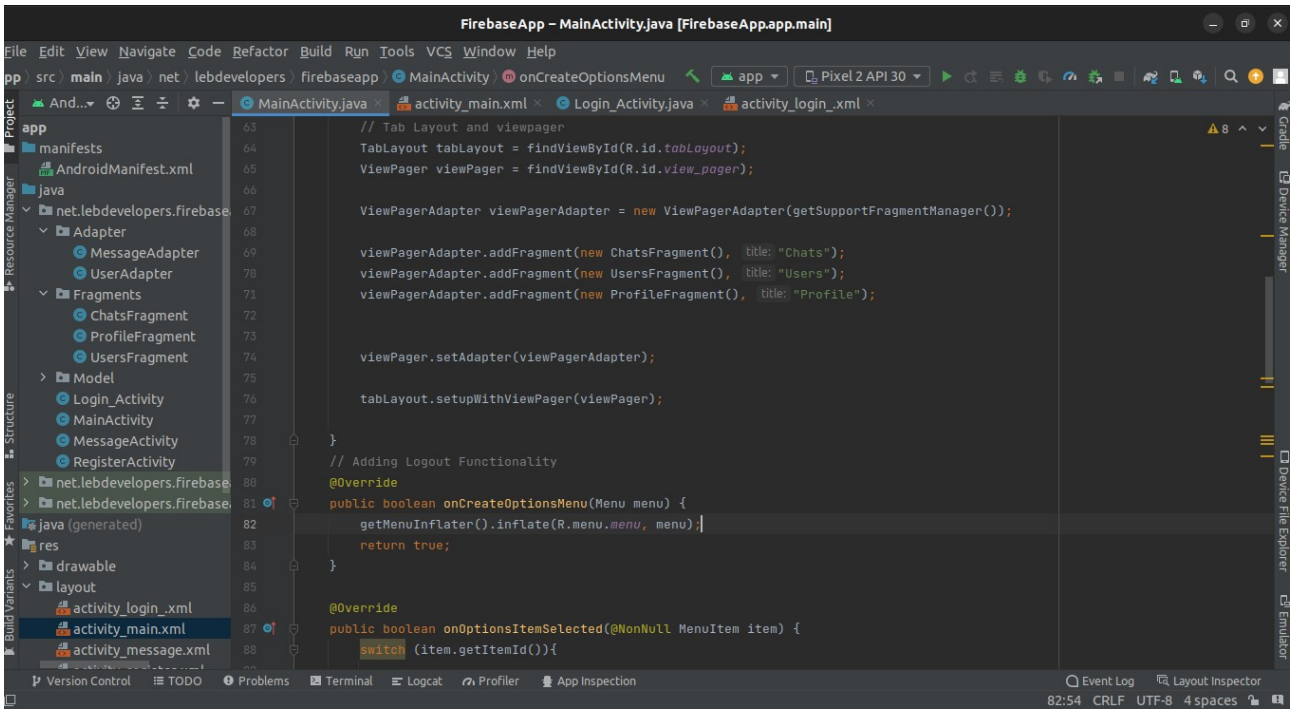


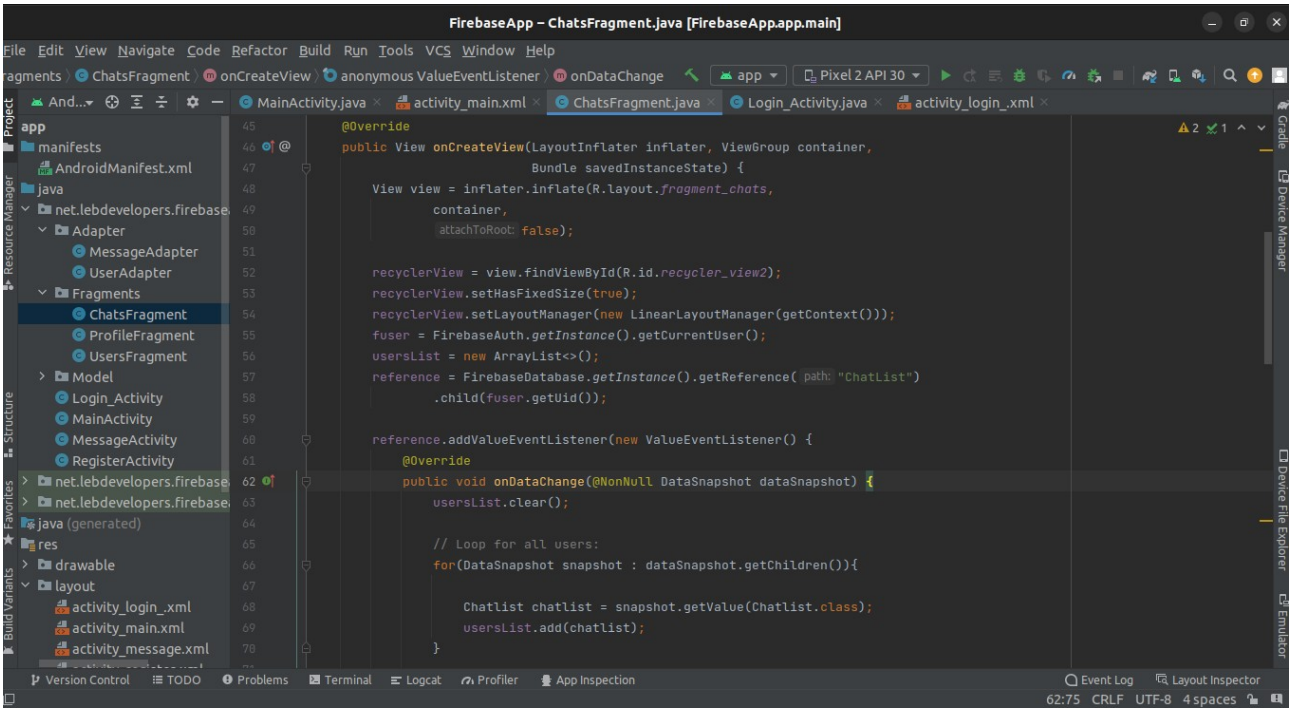


4.2.2 Java Code

Java is **the technology of choice for building applications using managed code that can execute on mobile devices**. Android is an open source software platform and Linux-based operating system for mobile devices.




FirebaseApp – MainActivity.java [FirebaseApp.app.main]
File Edit View Navigate Code Refactor Build Run Tools VCS Window Help
MainActivity.java
activity_main.xml
Login_Activity.java
activity_login_.xml
// Tab Layout and viewpager
TabLayout tabLayout = findViewById(R.id.tabLayout);
ViewPager viewPager = findViewById(R.id.view_pager);
ViewPagerAdapter viewPagerAdapter = new ViewPagerAdapter(getSupportFragmentManager());
viewPagerAdapter.addFragment(new ChatsFragment(), title: "Chats");
viewPagerAdapter.addFragment(new UsersFragment(), title: "Users");
viewPagerAdapter.addFragment(new ProfileFragment(), title: "Profile");
viewPager.setAdapter(viewPagerAdapter);
tabLayout.setupWithViewPager(viewPager);
}
// Adding Logout Functionality
@Override
public boolean onCreateOptionsMenu(Menu menu) {
getMenuInflater().inflate(R.menu.menu, menu);
return true;
}
@Override
public boolean onOptionsItemSelected(@NonNull MenuItem item) {
switch (item.getItemId()){
Version Control
TODO
Problems
Terminal
Logcat
Profiler
App Inspection
Event Log
Layout Inspector
82:54 CRLF UTF-8 4 spaces


FirebaseApp – ChatsFragment.java [FirebaseApp.app.main]
File Edit View Navigate Code Refactor Build Run Tools VCS Window Help
MainActivity.java
activity_main.xml
ChatsFragment.java
Login_Activity.java
activity_login_.xml
@Override
public View onCreateView(LayoutInflater inflater, ViewGroup container,
Bundle savedInstanceState) {
View view = inflater.inflate(R.layout.fragment_chats,
container,
attachToRoot: false);
recyclerView = view.findViewById(R.id.recycler_view2);
recyclerView.setHasFixedSize(true);
recyclerView.setLayoutManager(new LinearLayoutManager(getContext()));
fuser = FirebaseAuth.getInstance().getCurrentUser();
usersList = new ArrayList<>();
reference = FirebaseDatabase.getInstance().getReference(path: "ChatList")
.child(fuser.getUid());
reference.addValueEventListener(new ValueEventListener() {
@Override
public void onDataChange(@NonNull DataSnapshot dataSnapshot) {
usersList.clear();
// Loop for all users:
for(DataSnapshot snapshot : dataSnapshot.getChildren()){
Chatlist chatlist = snapshot.getValue(Chatlist.class);
usersList.add(chatlist);
}
Version Control
TODO
Problems
Terminal
Logcat
Profiler
App Inspection
Event Log
Layout Inspector
62:75 CRLF UTF-8 4 spaces

5. Future Scope

- We can add Search option in the User’s Fragment.
- We can add Video and Audio Calls.
- We can add more information about User.