



(Established under Galgotias University Uttar Pradesh Act No. 14 of 2011)

ANDROID CHAT SYSTEM USING FIREBASE

A Project Report of Capstone Project – 2

Submitted by

NISHANT RANJAN

(17SCSE121007)

*in partial fulfilment for the award of the
degree of*

BACHELOR OF COMPUTER APPLICATION

SCHOOL OF COMPUTING SCIENCE AND ENGINEERING

**Under the Supervision of
Ravindra Kumar Chahar,
Associate Professor**

APRIL / MAY- 2020



SCHOOL OF COMPUTING AND SCIENCE AND ENGINEERING

BONAFIDE CERTIFICATE

Certified that this project report **“ANDROID CHAT SYSTEM USING
FIREBASE”** is the bonafide work of **“NISHANT RANJAN
(17SCSE121007)”** who carried out the project work under my supervision.

SIGNATURE OF HEAD

Dr. MUNISH SHABARWAL,
PhD (Management), PhD (CS)
Professor & Dean,
**School of Computing Science &
Engineering**

SIGNATURE OF SUPERVISOR

RAVINDRA KUMAR CHAHAR,
Associate Professor
**School of Computing Science &
Engineering**

Title of the Project

Android Chat System using Firebase

Area/Concepts Involved

Android App Development, RDBMS

Abstract

The main objective of the Online Chat Application is to manage the details of Chat Application, Online Chat, Smiles Chat, Users, Chat History.

It manages all the information about Chat Application, Chat Profile, Chat History, Chat Application. The project is totally built at administrative end and thus only the administrator is guaranteed the access.

The purpose of the project is to build an application program to reduce the manual work for managing the Chat Application, Online Chat, Chat Profile, Smiles Chat.

It tracks all the details about the Smiles Chat, Users, Chat History.

Features-

- Application also provides excel export for Online Chat, Smiles Chat, Chat History
- You can also export the report into csv format for Chat Application, Online Chat, Chat History
- To increase efficiency of managing the Chat Application, Online Chat
- It deals with monitoring the information and transactions of Users.
- Manage the information of Chat Application
- Editing, adding and updating of Records is improved which results in proper resource management of Chat Application data.
- Manage the information of Users
- Integration of all records of Chat History

Literature Survey:

Types of mobile apps

There are three main types of mobile apps including native apps, web-based mobile apps and hybrid apps. Hybrid mobile apps combine elements of native and web-based apps. Native apps include Android, Windows Phone, and iOS. Hybrid apps are designed for platforms including Xamarin, Angular Mobile Sencha Touch, React Native, Ionic and others. On the other hand, web-based apps are responsive versions of websites designed in order to work on any mobile device.

Native Mobile Apps

Native apps are developed for a certain mobile device operating system like Windows Phone or Android. Therefore, they are native for a certain device or platform. Apps built for Android, Windows Phone, Blackberry, Symbian cannot be used on any other platform except on their own. Therefore, a mobile app designed for Android can only be used on an Android device. Main advantages of native apps are good user experience and high performance. In addition, an access to broad range of APIs puts no limits on app usage. Native mobile apps are accessible from app stores of their kind and have that very clear tendency of reaching target customers. Some disadvantages of native mobile apps include higher costs in comparison to other types of mobile apps. Creating a native mobile app duplicates costs since maintenance and separate supports for different apps are required that result in greater product price.

Hybrid Mobile Apps

Hybrid mobile apps are specifically built using different multi-platform web technologies like JavaScript and HTML5. Hybrid apps are website applications created in a native wrapper that means they use elements of both native and web-based apps. Hybrid apps also possess common cons and pros of both web mobile and native mobile applications. Hybrid multi-platform mobile apps are relatively easy to develop which is a clear advantage. Since code base for hybrid apps ensures that low-cost maintenance alongside smooth updates. On the other hand, hybrid application lack in speed, performance and overall optimization compared to native mobile apps. There are also specific design issues thanks to apps inability to look in the same way on different platforms.

Web-Based Apps

Web-based applications behave in very similar fashion to those native mobile apps. Web apps use a certain browser in order to run and they are commonly written in CSS, JavaScript or HTML5. Web apps redirect users to URL and further offer install options by creating a bookmark on their browser. The greatest advantage of web apps is that they require a minimum of device memory. Users can access web apps from any device that is connected to the Internet. All personal databases are saved on a certain server, so the use of web

applications with poor internet connection commonly results in very bad user experience. Another drawback on web apps is access to not so many APIs, with exception of geolocation and several others. In addition, a performance of web-based apps is inextricably linked to network connection and browser work. Therefore, only around fourteen percent of time users spend on web-based apps as only some of device APIs can be used like geolocation.

Categories of Apps

There are thirty-three categories of application in Google Play and twenty-to categories in Apple's App Store. However, there are only seven app categories which have managed to reach more than three percent of users according to latest studies. Commonly used seven categories of apps include gaming apps, business, education, lifestyle, entertainment, utilities, travel, book, health and fitness and food and drink. Around twenty-four percent of users download gaming apps. Gaming apps are followed by ten-percent of users who download business, education and lifestyle apps. By far the most popular category is gaming apps. This is not surprising since the average personal time spent playing mobile games is constantly increasing according to the latest research. Mobile gaming and gaming apps have always been thriving, so developers spend more time and money into creating new games on regular basis. Gaming apps in popularity are followed by business app commonly referred to as productivity apps. Modern-day portable devices are capable of performing various complex tasks, so there is no wonder business apps used for billing, sending emails and checking working progress are on the second place. The most commonly used business apps include Adobe Acrobat Reader, Voxer Walkie-Talkie Messenger, and Indeed Job Search. On the third place are educational apps. Leading apps in education category are Duolingo, Photomath, and Quizlet. Educational apps are followed by popular lifestyle apps like Tinder, Text Free, and AroundMe. Entertainment apps are also very popular like Netflix, Dubsmash, Talking Tom Cat, Amazon Prime Video and other. Utility apps closely follow entertainment apps in a number of downloads. The most popular utility apps include Bitmoji, Flashlight, QR Reader, Speed Test and My Verizon. Other popular categories of apps include travel apps like Google Earth and Uber.




Steps for developing android application

React Redux


Official React bindings for Redux

GET STARTED




Official

React Redux is maintained by the Redux team, and kept up-to-date with the latest APIs from Redux and React.




Predictable

Designed to work with React's component model. You define how to extract the values your component needs from Redux, and your component receives them as props.



Encapsulated

Creates wrapper components that manage the store interaction logic for you, so you don't have to write it yourself.



Optimized

Automatically implements complex performance optimizations, so that your own component only re-renders when the data it needs has actually changed.

Other Libraries from the Redux Team

Redux

A predictable state container for JavaScript applications

Redux Starter Kit

A simple batteries-included toolset to make using Redux easier

Docs Community More

Firestore console

Welcome to Firebase!

Tools from Google for developing great apps, engaging with your users and earning more through mobile ads.

[Learn more](#) [Documentation](#) [Support](#)

Recent projects

+


Add project

[Explore a demo project](#)

socialape

socialape-0081e

Terms · Privacy Policy



Languages that may be used:

Java/Kotlin, XML, Firebase

Software/Tools:

- Android Studio
- API
- Canva
- Firebase
- GenyMotion
- XML

Introduction-

a. Overall description

The main objective of the Online Chat Application is to manage the details of Chat Application, Online Chat, Smiles Chat, Users, Chat History. It manages all the information about Chat Application, Chat Profile, Chat History, Chat Application. The project is totally built at administrative end and thus only the administrator is guaranteed the access. The purpose of the project is to build an application program to reduce the manual work for managing the Chat Application, Online Chat, Chat Profile, Smiles Chat. It tracks all the details about the Smiles Chat, Users, Chat History.

b. Motivation and Scope

- My Project is only a humble venture to satisfy the needs to manage their project works. Several user-friendly coding has been adopted.
- This package shall be a powerful package in satisfying all the requirements of the college project.
- Made statement of aims and objectives of the project.
- The description of Purpose, Scope and applicability.
- Included features and operations in detail, including screen layouts.
- We designed user interface and security issues related to system.

- Finally, the system is implemented and tested according to test cases

Problem Statement

- No simplified user to user chatting application.
- Dependency on the internet for the service.
- Security issues.
- Privacy issues.
- Simplified User Interface.

Proposed Model

- Provides the searching facilities based on various factors. Such as Chat Application, Smiles Chat, Users, Chat History
- The transactions are executed in off-line mode, hence on-line data for Chat Application, Online Chat capture and modification is not possible.
- It tracks all the information of Online Chat, Chat Profile, Users etc.
- Manage the information of Online Chat
- Shows the information and description of the Chat Application, Smiles Chat
- All the fields such as Chat Application, Smiles Chat, Chat History are validated and does not take invalid values
- It generates the report on Chat Application, Online Chat, Chat Profile
- Provide filter reports on Smiles Chat, Users, Chat History
- You can easily export PDF for the Chat Application, Chat Profile, Users.

Implementation

a. Goals

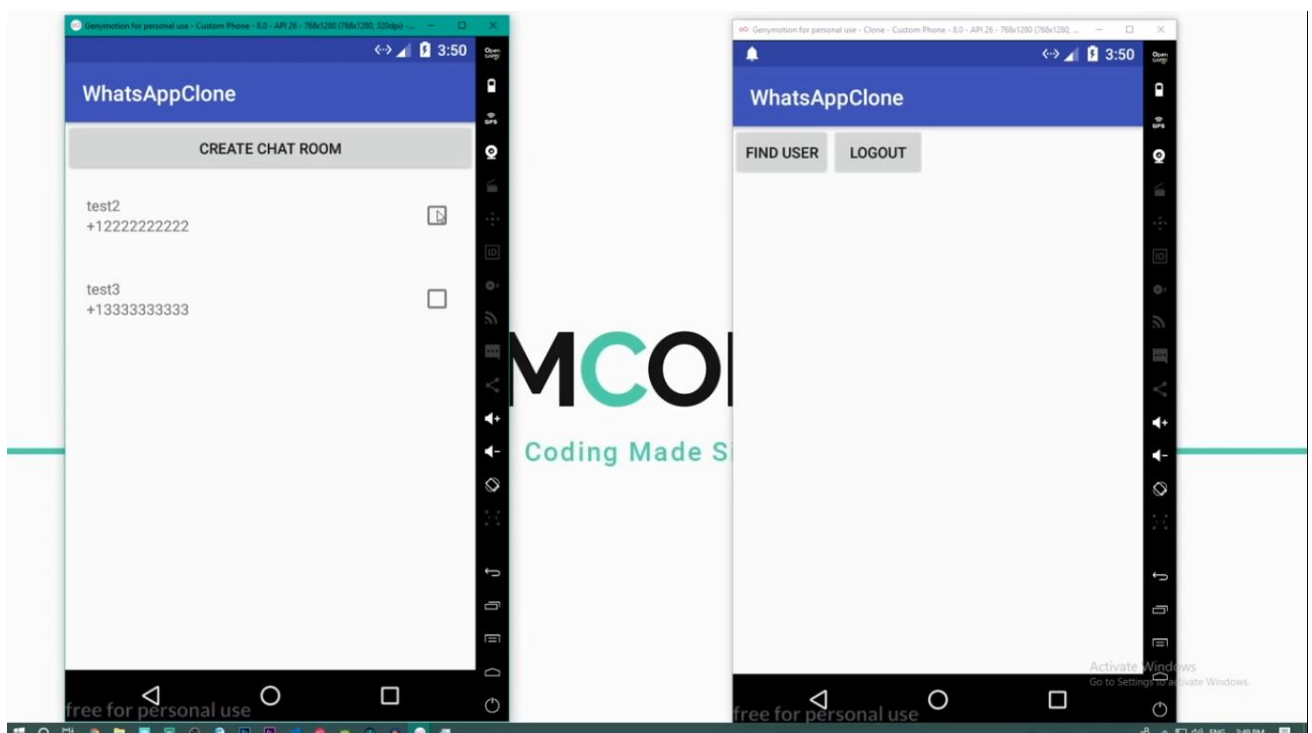
1. Login with Phone Number
2. Simple Text Chat
3. Send Images
4. Group Chat
5. Assorted Features

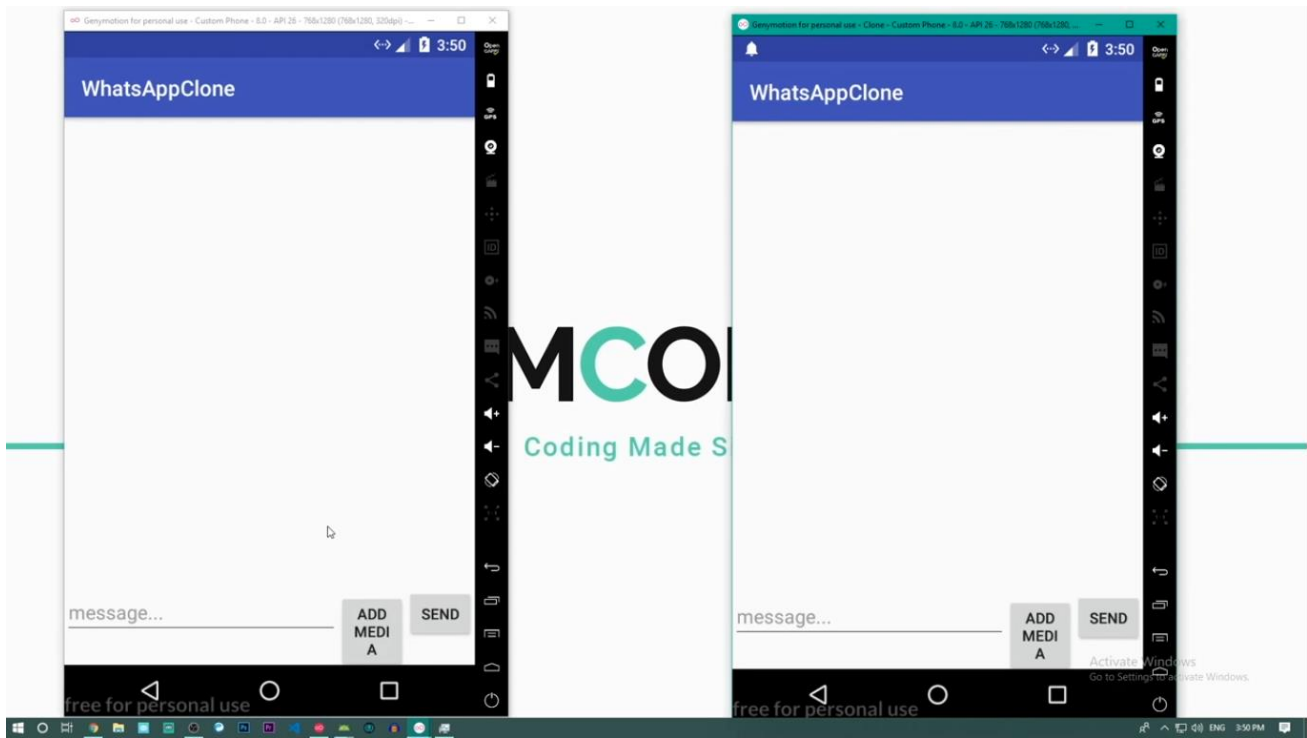
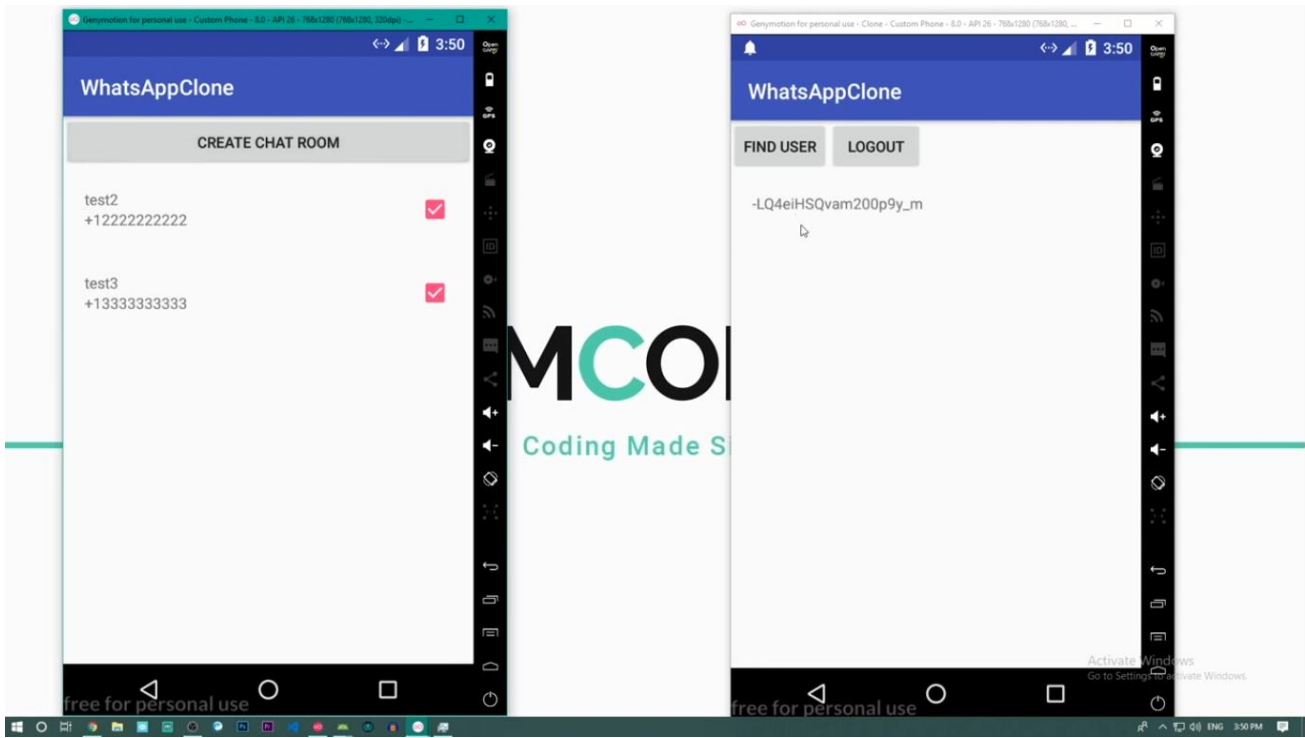
b. Modules

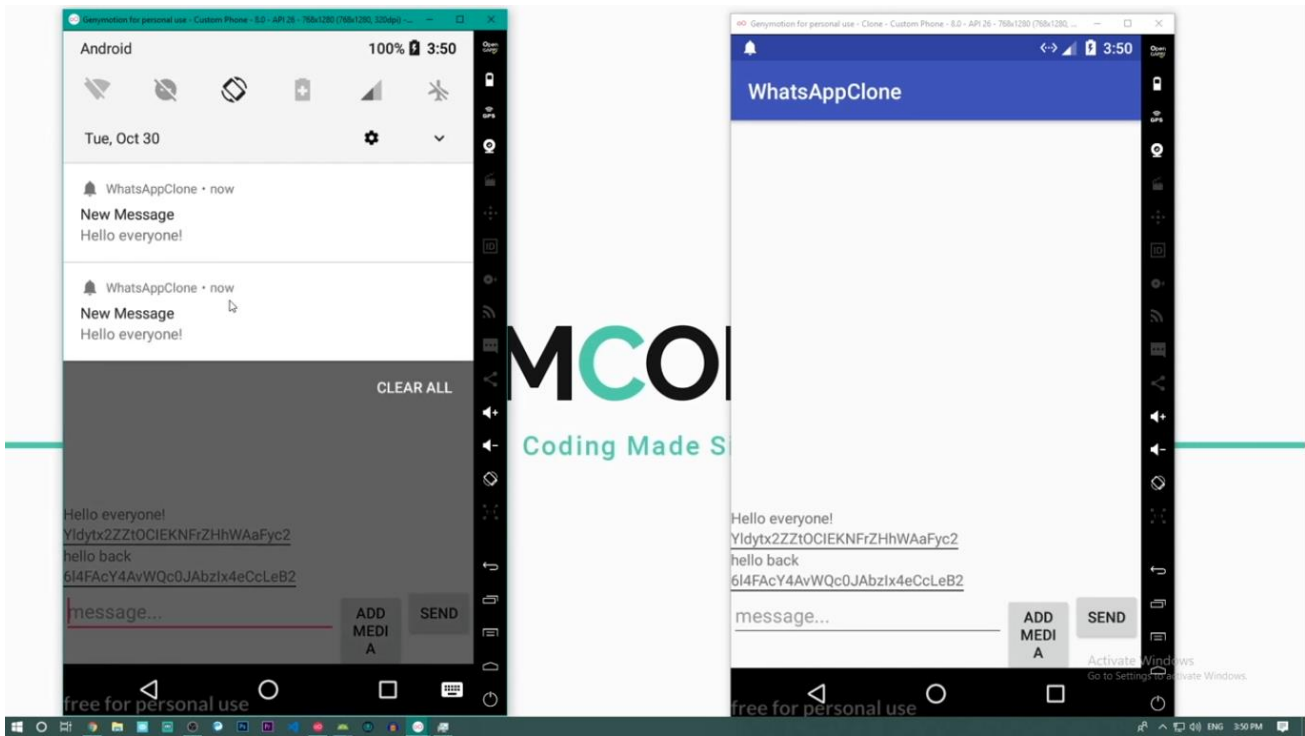
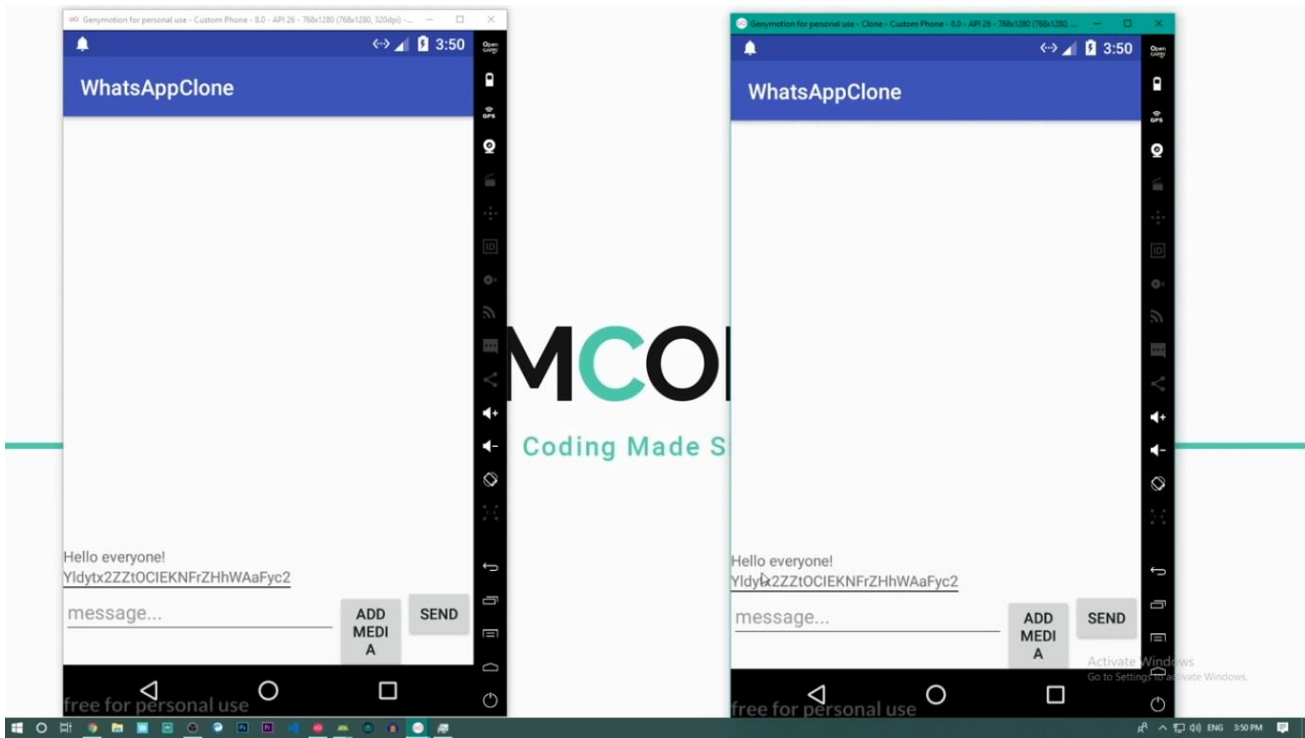
- Online Chat Management Module: Used for managing the Online Chat details.
- User Module: Used for managing the details of user.
- Chat Application Management Module: Used for managing the information and details of the Chat Application.
- Chat History Module: Used for managing the Chat History Details.
- Chat Profile Module: Used for managing the Chat profile information.
- Login Module: Used for managing the login details.
- User Module: Used for managing the users of the system

Design View

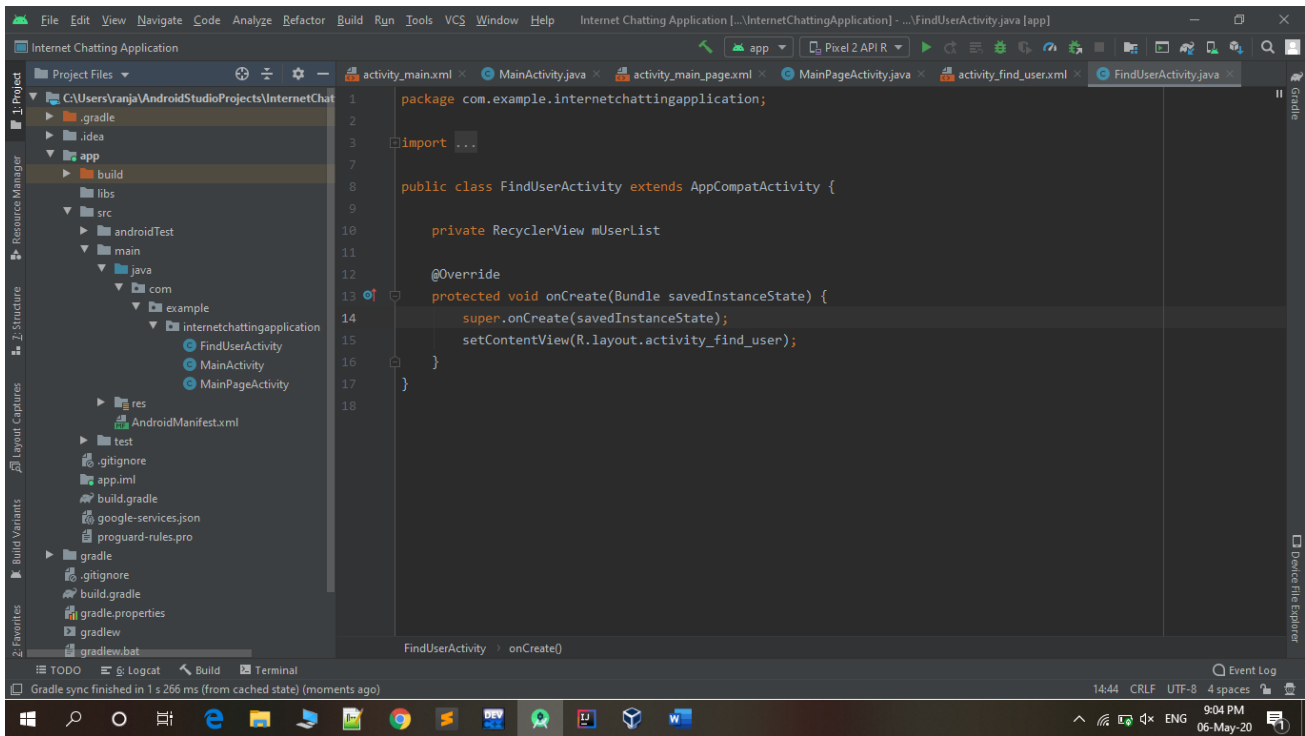
- System needs to store information about new entry of online chat.
- System also needs a search area for older chats.
- System need to keep the record of chat history.
- System need to update and delete the record.
- Find the way to chat without internet connection.





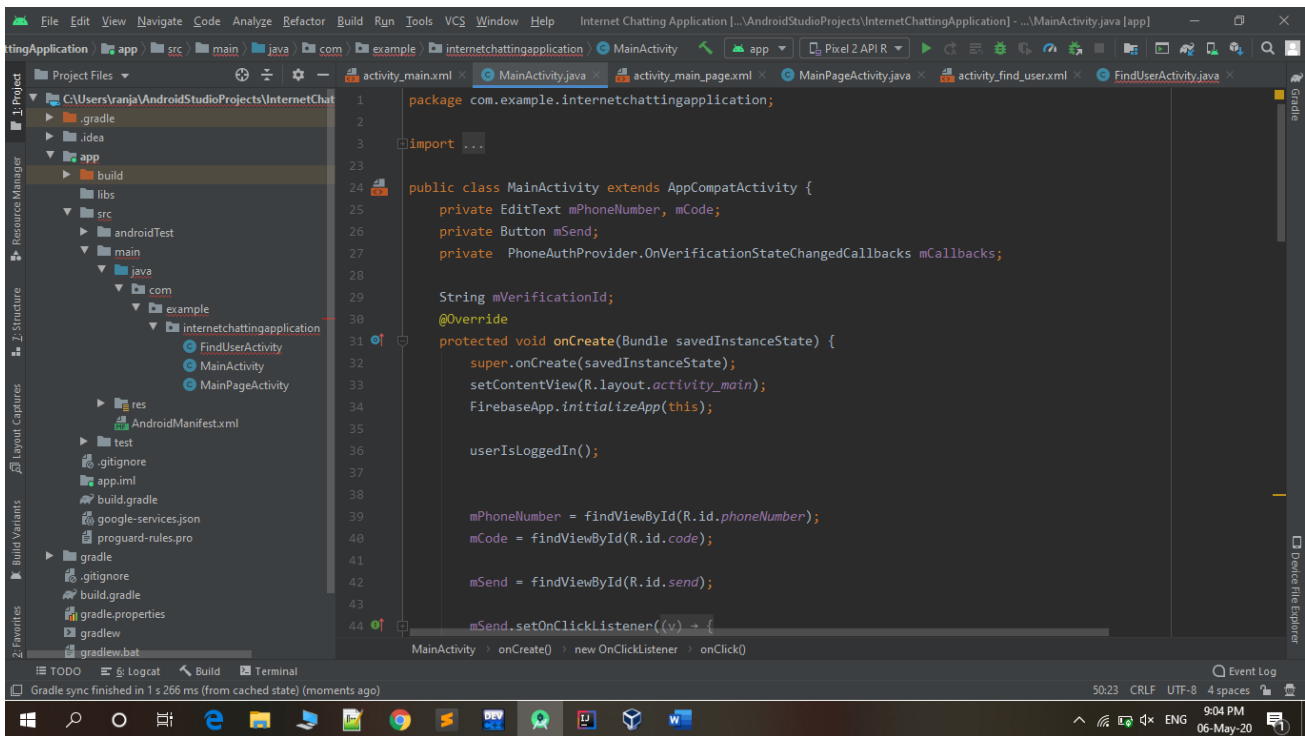


CODE



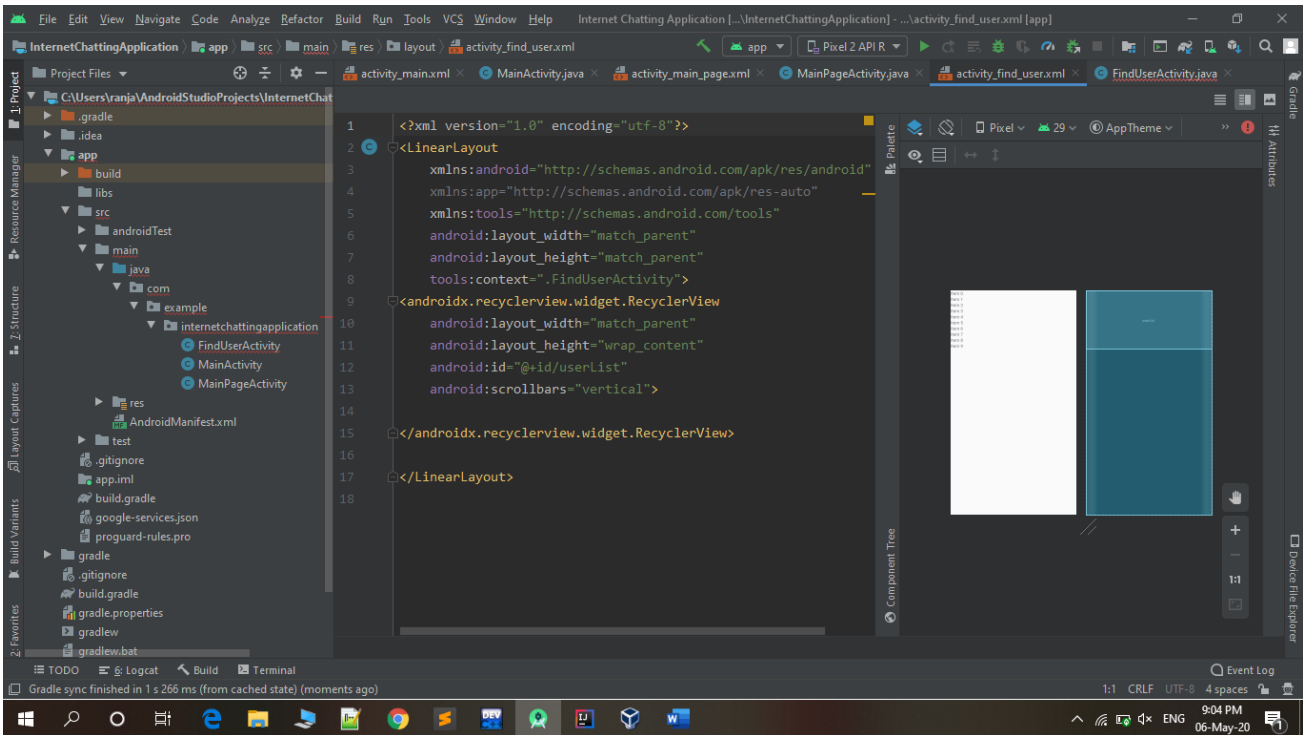
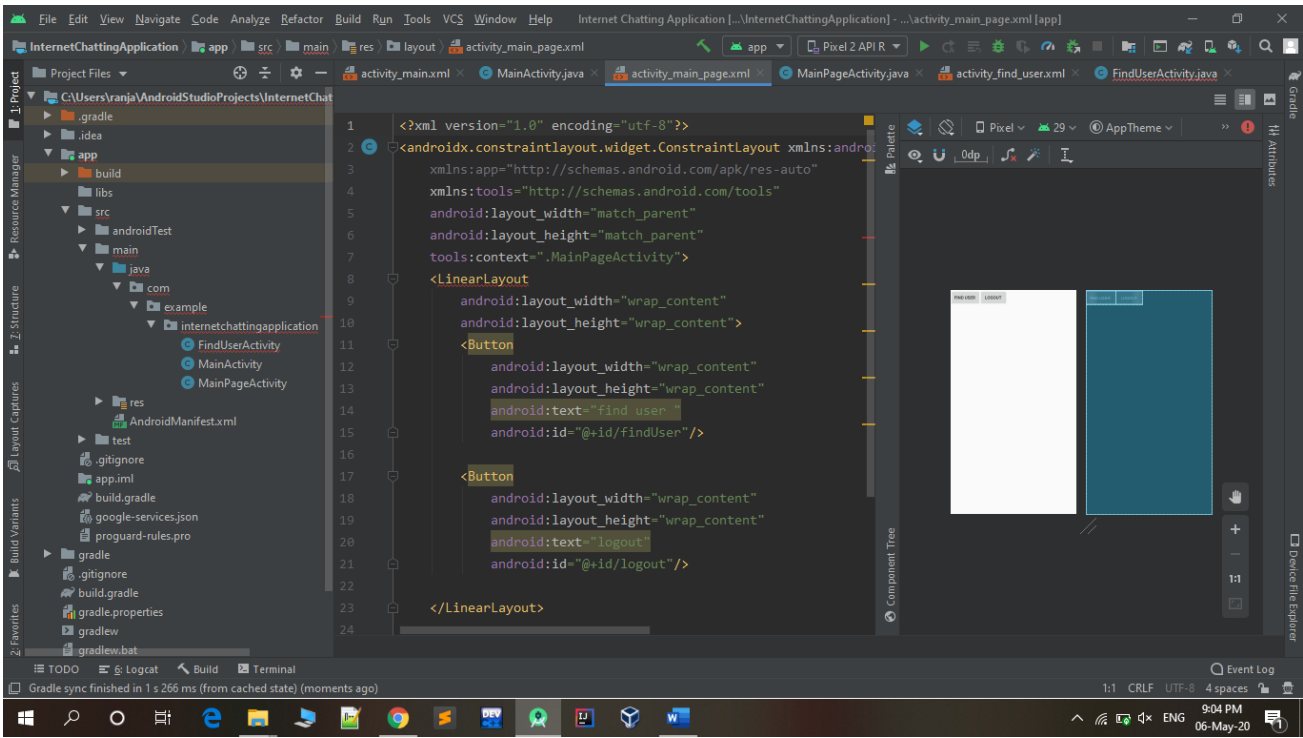
The screenshot shows the Android Studio IDE with the 'FindUserActivity.java' file open. The code defines a class that extends AppCompatActivity and implements the onCreate method to initialize a RecyclerView.

```
1 package com.example.internetchattingapplication;
2
3 import ...
4
5 public class FindUserActivity extends AppCompatActivity {
6
7     private RecyclerView mList;
8
9     @Override
10    protected void onCreate(Bundle savedInstanceState) {
11        super.onCreate(savedInstanceState);
12        setContentView(R.layout.activity_find_user);
13    }
14 }
15
16
17
18
```



The screenshot shows the Android Studio IDE with the 'MainActivity.java' file open. The code defines a class that extends AppCompatActivity and implements the onCreate method to initialize the app and find UI elements.

```
1 package com.example.internetchattingapplication;
2
3 import ...
4
5 public class MainActivity extends AppCompatActivity {
6     private EditText mPhoneNumber, mCode;
7     private Button mSend;
8     private FirebaseAuthProvider.OnVerificationStateChangedCallbacks mCallbacks;
9
10    String mVerificationId;
11
12    @Override
13    protected void onCreate(Bundle savedInstanceState) {
14        super.onCreate(savedInstanceState);
15        setContentView(R.layout.activity_main);
16        FirebaseApp.initializeApp(this);
17
18        userIsLoggedIn();
19
20        mPhoneNumber = findViewById(R.id.phoneNumber);
21        mCode = findViewById(R.id.code);
22
23        mSend = findViewById(R.id.send);
24        mSend.setOnClickListener((v) -> {
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
```



```
File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help Internet Chatting Application [...InternetChattingApplication] - ...FindUserActivity.java [app]
Project Files
C:\Users\vranja\AndroidStudioProjects\InternetChat
  .gradle
  .idea
  app
    build
    libs
    src
      androidTest
      main
        java
          com
            example
              internetchattingapplication
                FindUserActivity
                MainActivity
                MainPageActivity
          res
            AndroidManifest.xml
            test
            .gitignore
            app.iml
            build.gradle
            google-services.json
            proguard-rules.pro
          gradle
            .gitignore
            build.gradle
            gradle.properties
            gradlew
            gradlew.bat
Terminal
Gradle sync finished in 1 s 266 ms (from cached state) (moments ago)
14:44 CRLF UTF-8 4 spaces
9:04 PM
06-May-20
```

```
package com.example.internetchattingapplication;

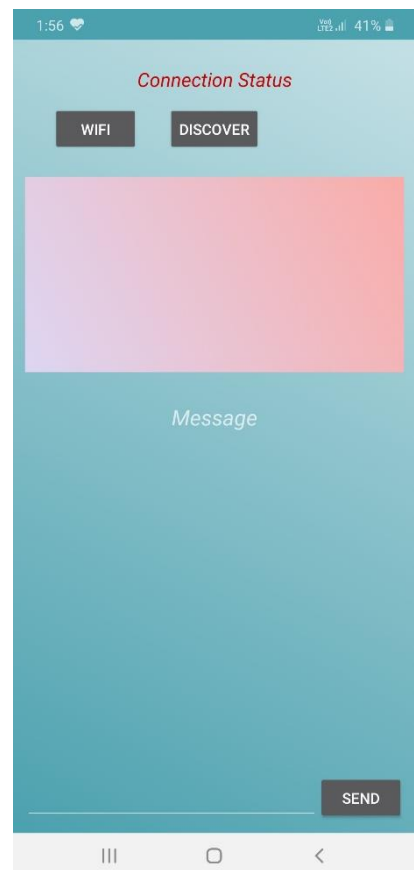
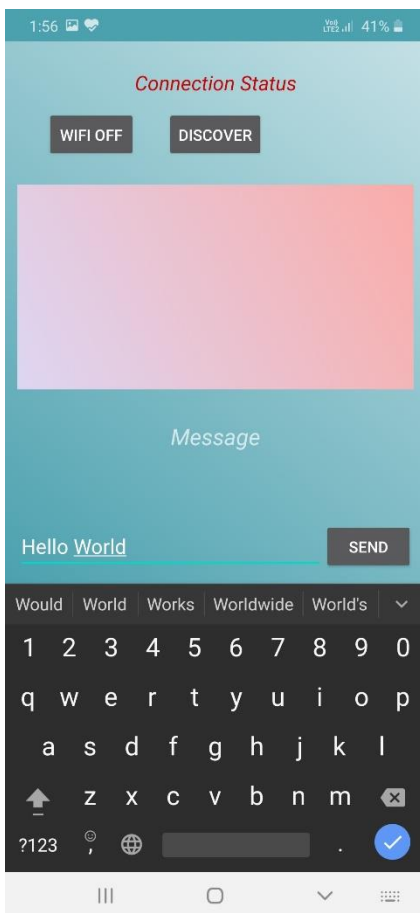
import ...

public class FindUserActivity extends AppCompatActivity {

    private RecyclerView userList;

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

Implementation:



System Requirements

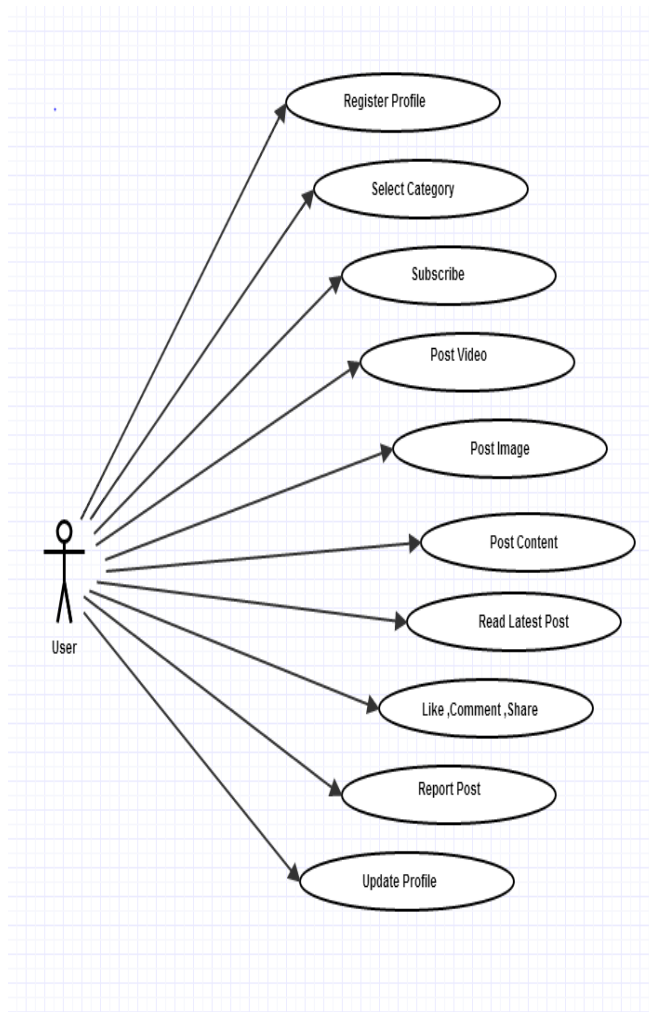
	Linux
Language	Java 2 Runtime Environment
Database	MySQL Server
Browser	Any of Mozilla, Opera, Chrome etc
Web Server	Tomcat 7
Software Development Kit	Java JDK 1.7 or Above
Scripting Language Enable	JSP (Java Server Pages)
Database JDBC Driver	MySQL Jconnector

Hardware Requirements:

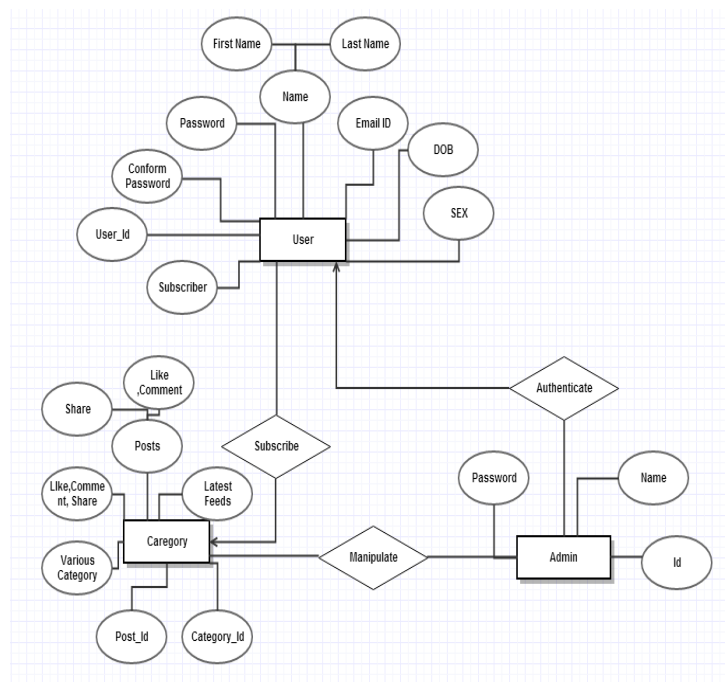
Name of component	Specification
Processor	Pentium III 630MHz
RAM	128 MB
Hard disk	20 GB
Monitor	15" color monitor
Keyboard	122 keys

Diagram

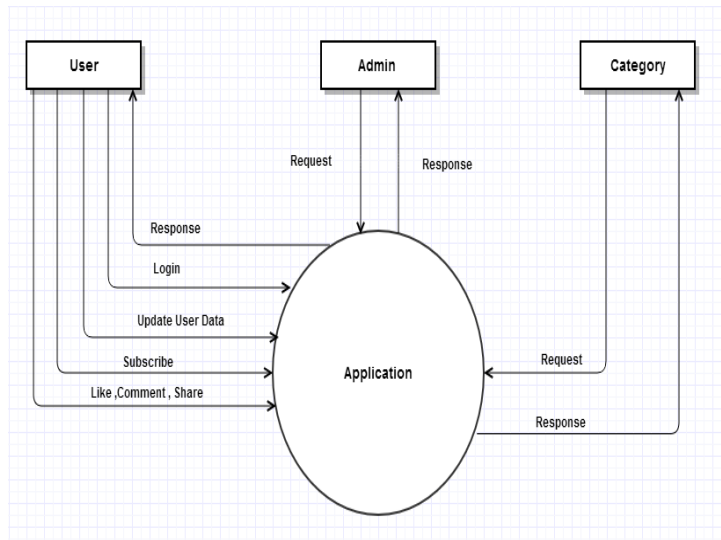
Use Case Diagram



ER Diagram



DFD Diagram



References-

- Google for problem solving
- <https://www.javatpoint.com/java-tutorial>
- <https://www.developerfusion.com/article/84479/jsp-for-netdevelopers/>
- <https://docs.oracle.com/javase/tutorial/>
- https://www.tutorialspoint.com/the_full_stack_web_development/in_dex.asp
- <https://developer.android.com/>
- <https://flutter.dev/docs/get-started/install/windows>