

Abstract

Social media is a way of living these days. As the popularity of this medium is on the rise, so are the risks associated with them. The possibility of your data leak is huge. Hackers can add up to your misery by either scamming you or ultimately hacking your social media account. Fake accounts also play their part in here. Therefore, you have to be very vigilant while using social media apps for your personal and business interests.

Many of these FB users are unaware of the risks involved. You may not know that you have been under the observation of a stalker, and the probability of your privacy breach is high.

Only a few Facebook users know that the privacy settings that your account was set on will revert to the default settings every time Facebook redesigns its website. Now here you stand again behind an unlocked door. The rising risks need you to level-up your security mechanisms and to focus on this area attentively.

CONTENTS

LIST

Page No.

1. Certificate	2
2. Abstract	3
3. Acknowledgement	4
4. Table of Contents	5
5. List of figures	7
6. Chapters	
i) Introduction	8
ii) Motivation	9
iii) Problem Statement	10
iv) Declared Problem	12
v) Aim	
• Objective	13
• Social Media cloning	14
• Authentication	15
vi) Cloning	
• Attacks of Cloning	17
• Precautions	19
viii) Authentication with AES encryption	
• Working	21
• AES 128 vs AES 256	22
• AES vs RSA	23
• AES vs DES	24

ix) Tools

- Android Studio** **25**
- Firebase** **26**
- Eclipse IDE** **27**

x) Process Flow **30**

xi) Firebase console **42**

7. Future Scope **43**

8. References **44**

9. Checklist **45**

10. Declaration **46**

LIST OF FIGURES

- **Figure Number 1 → The World most used social platforms**
- **Figure Number 2 → Gender base users**
- **Figure Number 3 → Scams of social media**
- **Figure Number 4 → Attack of the Clones**
- **Figure Number 5 → Gb WhatsApp**
- **Figure Number 6 → Basic Authentication**
- **Figure Number 7 → Clone App**
- **Figure Number 8 → Multi Parallel App**
- **Figure Number 9 → 2A Account**
- **Figure Number 10 → Dr. Clone**
- **Figure Number 11 → Parallel U**
- **Figure Number 12 → Dual Space**
- **Figure Number 13 → AES Design**
- **Figure Number 14 → Android project view**
- **Figure Number 15 → User view in android studio**
- **Figure Number 16 → Tool Window**
- **Figure Number 17 → Eclipse window**
- **Figure Number 18 → Menu window of eclipse**
- **Figure Number 19 → Activities of App**
- **Figure Number 20 → Drawable layout of App**
- **Figure Number 21 → Drawable resources of App**
- **Figure Number 22 → Gradle Build of App**
- **Figure Number 23 → Firebase Console output**
- **Figure Number 24 → Firebase User hierarchy**

Introduction

Social media user numbers have surged in the past 12 months too, with 521 million new users joining social media in the year to April 2021.

That equates to annualised growth of 13.7 percent, or an average 16½ new users every single second.

It's also worth noting that comparisons of social media users to total population may under-represent the full extent of social media use, because most social media companies restrict use of their platforms to people aged 13 and above.

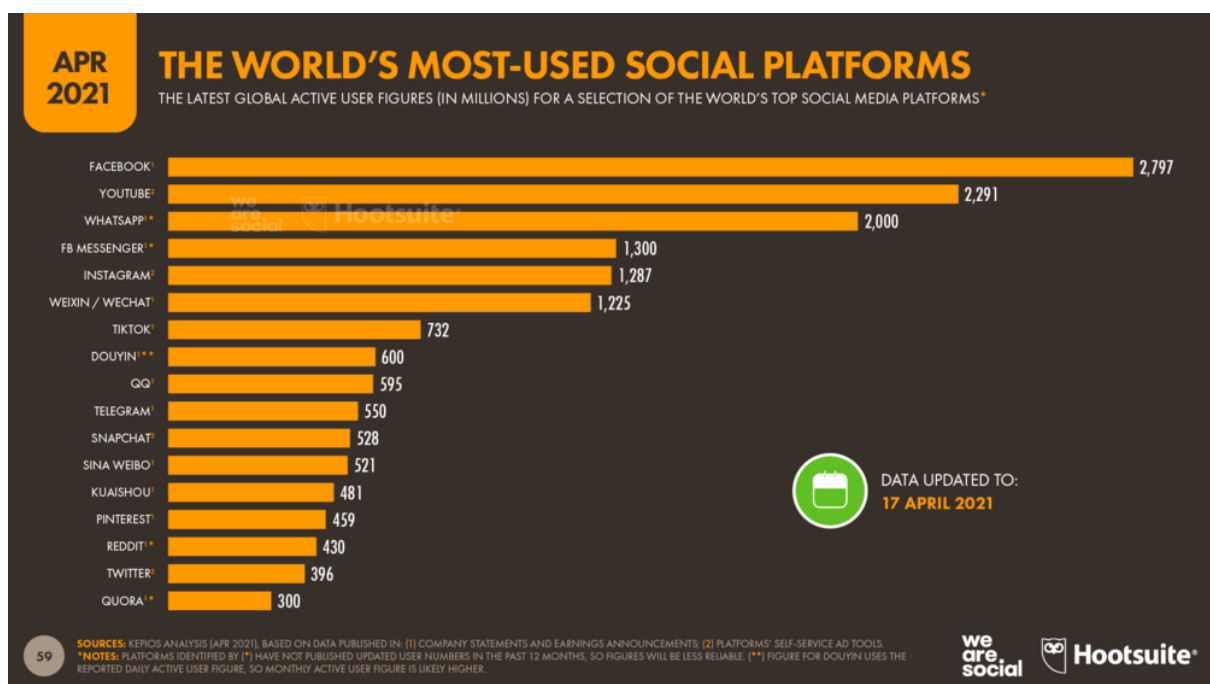


Fig no. 1

As you can analyse around 2000 foreign country are outspreaded their feet in connectivity with a traumatic potential.

Basic Goals of Connectivity –

- 1) Reduction of Physical documents example – Tables and notary papers.
- 2) Fast data sharing anywhere.
- 3) Secure data and file transmission without loss.
- 4) Reach to tough areas

The digital wars begin between crackers, grabbers and malicious data injectors for Money or anything important likewise money example – Crypto Currency (Bitcoin).

Motivation

In addition to the well-known importance of perceived threat, results show that privacy concerns are also an important factor in explaining the intention of social network users to engage in self-protective behaviour on social networks. But a common man as a soft user only knows basic operations and arranges his/her small works with palmtop, desktop, laptop or mobile phones.

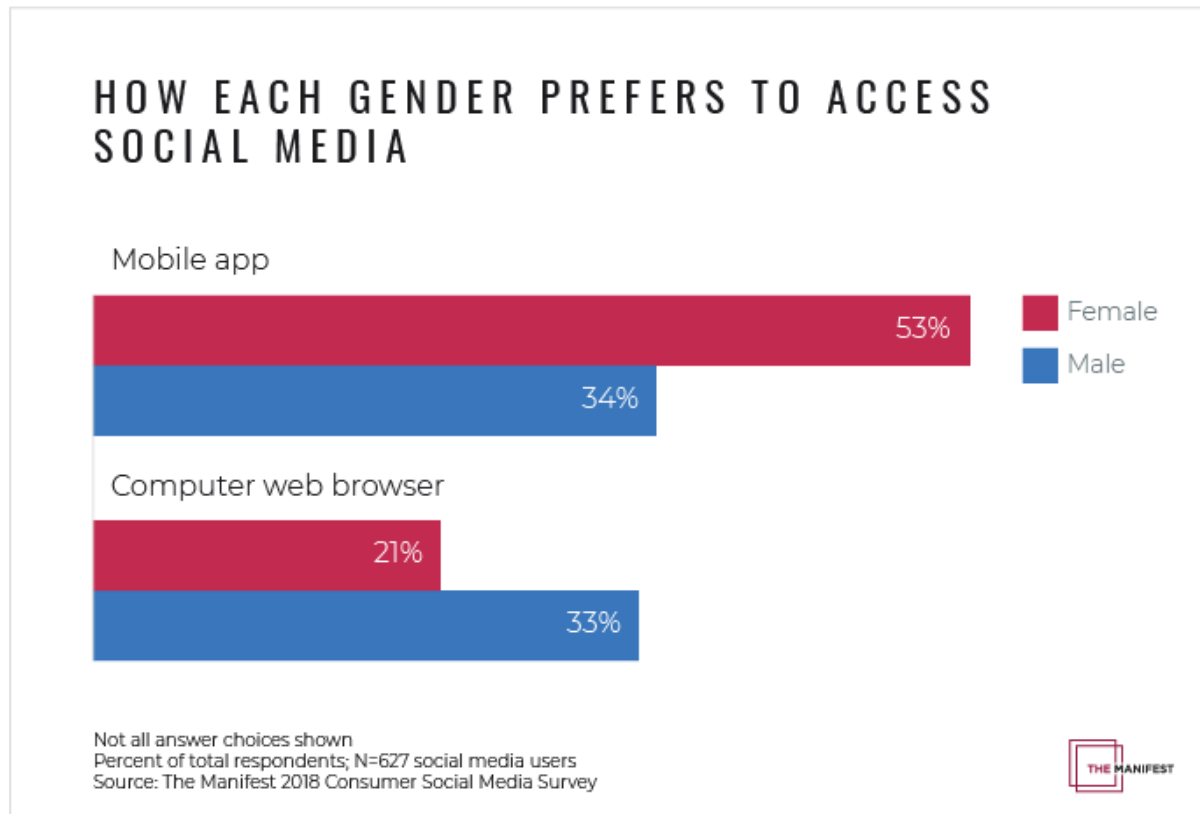


Fig no. 2

Above statistics shows distribution of users in virtual world.

Problem Statement

App clone essentially refers to taking suggestion from every other internet site or app subject which could be very particular and special and copying that idea to create any other App clone.

Cloning doesn't imply copying precisely similar to the authentic one, it manner taking a few thoughts of the App and including your personal particular functions together with it to create your personal new app. App cloning could be very famous these days and plenty of app improvement businesses have cloned famous app script like Uber, Ola Lyft and many others, reskinned it together with their personal new functions to create than App clone.

This is App cloning approach could be very beneficial and useful for budding marketers who want to begin their personal enterprise with an app. Designing an app from scratch takes a variety of time and funding which might not be affordable.

Even India is among largest platform of social media apps such as Telegram, WhatsApp etc.

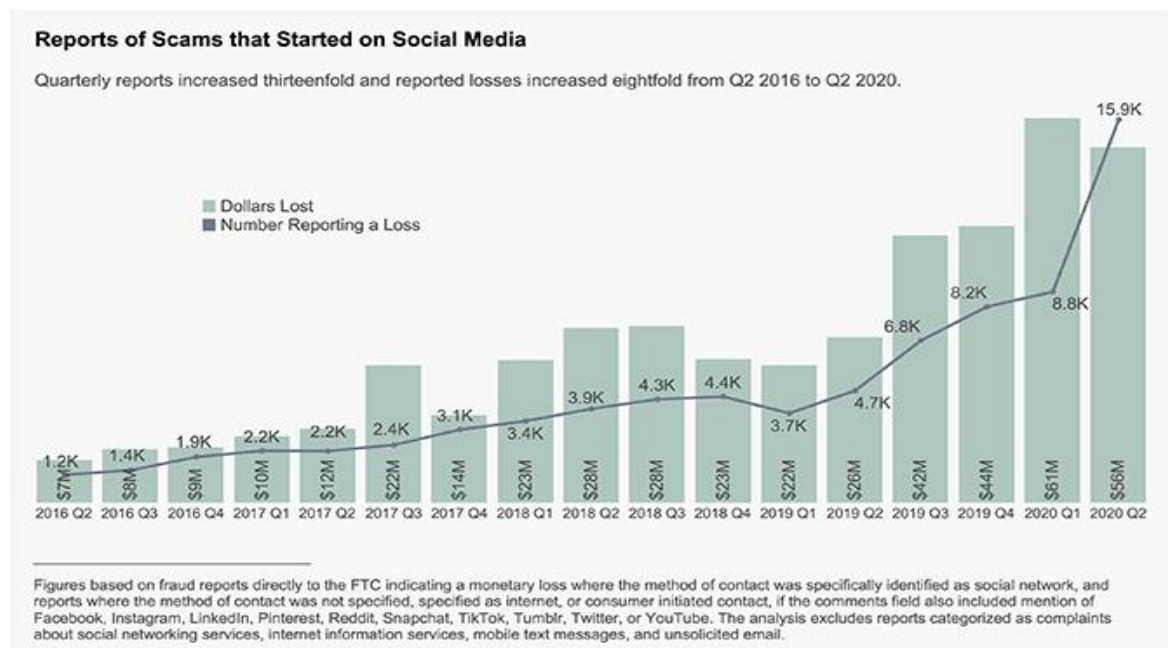


Fig no. 3

As you can see with a large population with less sophisticated users, India is an international hub to attract attackers and the attackers probably uses Bots to retrieve their credential data. Being online is easy but to maintain yourself in safe zone is conditionally tough for soft users.

A cloned app might not be able to handle the increased load, and the app might crash. In other terms, there might be issues with scalability in case you choose to adopt a cloned script.

Below is a depiction of attacking of several major Apps.

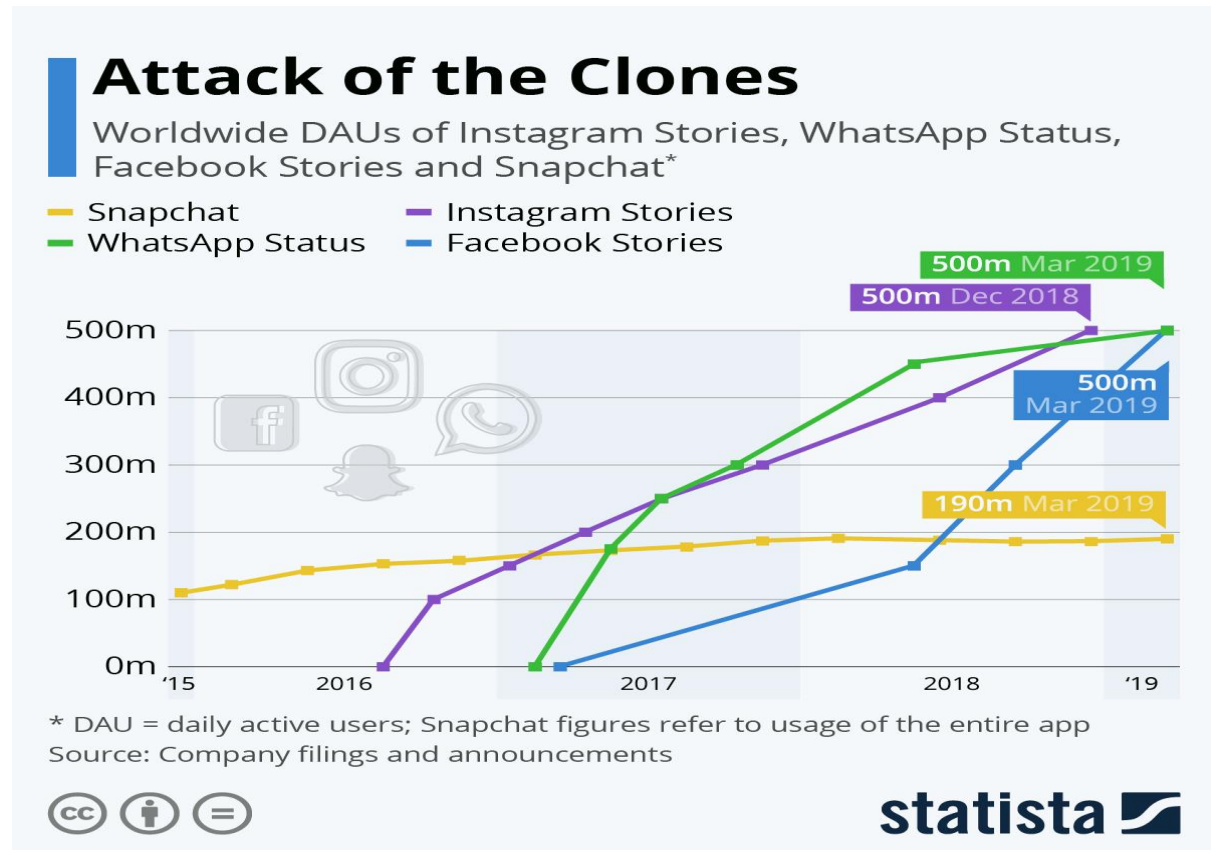


Fig no. 4

This chart shows daily active users of Instagram Stories, WhatsApp Status, Facebook Stories and Snapchat.

Declared Problem: -

The central root of individuals loss in digitalization is majorly due to app phishing but almost every case is recoverable except some encrypted attacks.

App cloner basically provides extra features to lure customers and of course they do not fall under copyright issues.

You can hack someone WhatsApp account and see all message without touching their phone and letting them know.

A very generous disaster had been seen modded apk of WhatsApp so called Gb WhatsApp.

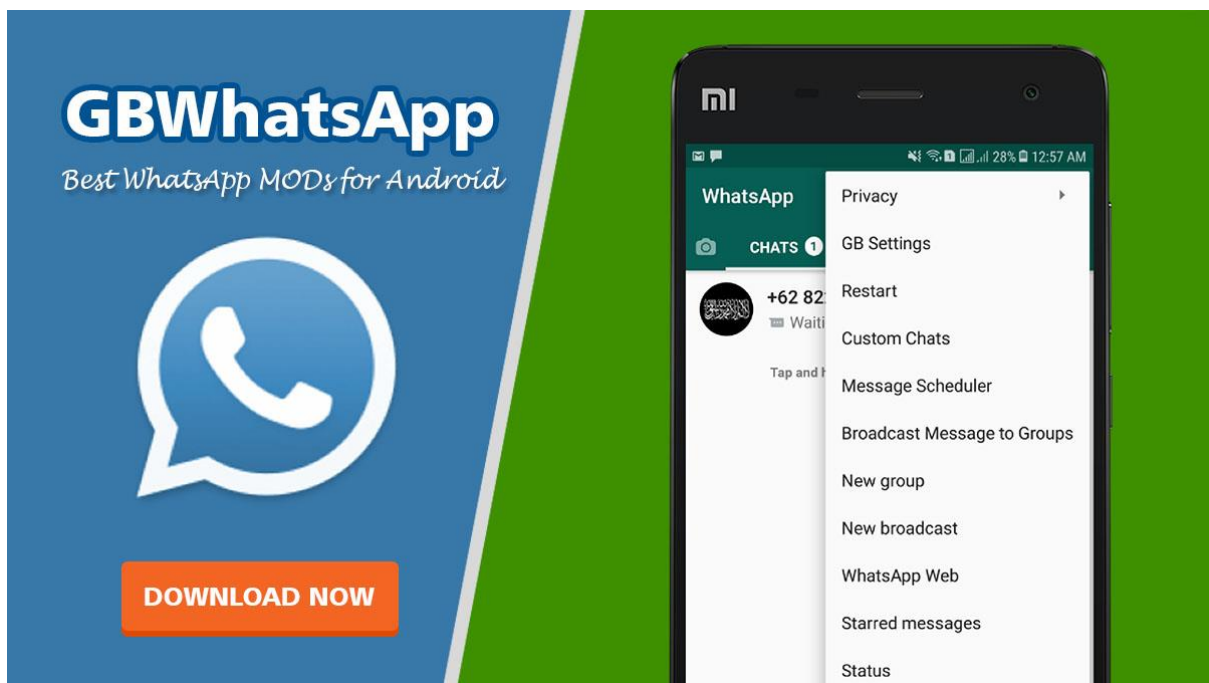


Fig no. 6

GB WhatsApp provides the same WhatsApp like experience to its users with additional features that includes using dual WhatsApp accounts on the same device, hide read receipts, send high-resolution images, Auto-reply feature, more characters in a status update and more.

WhatsApp warns against the use

Using a third-party app like GB WhatsApp can get your original WhatsApp account banned permanently. Additionally, these apps can also compromise your privacy as there are no well-known sources for downloading the app.

Goals and Milestones

To provide more understanding about the major problem and the key role of app cloning and their server working. we tried a small snippet to resolve such problems on machine level.

Aim → Nowadays there are many online databases available with different deploying packages in context of almost every language. Our main aim is to study the internal processing of a client server request in a social media app on different platforms and subject them for dummy data to grasp the functioning.

Key features →

- 1) Creating a chat app similar to WhatsApp
- 2) Subjecting to encryption less keyframing and accessing using **Java** module **Math.java**.

Objective: -

To get a thorough knowledge of access server, their key mechanism and encryption decryption techniques.

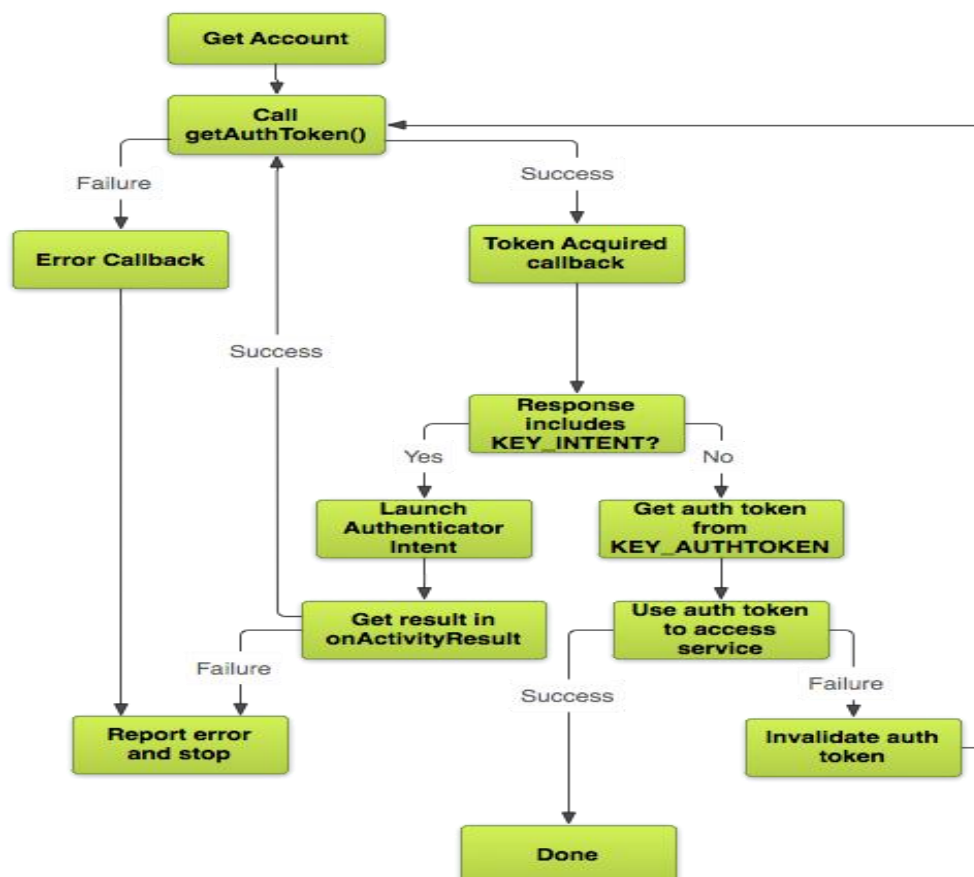
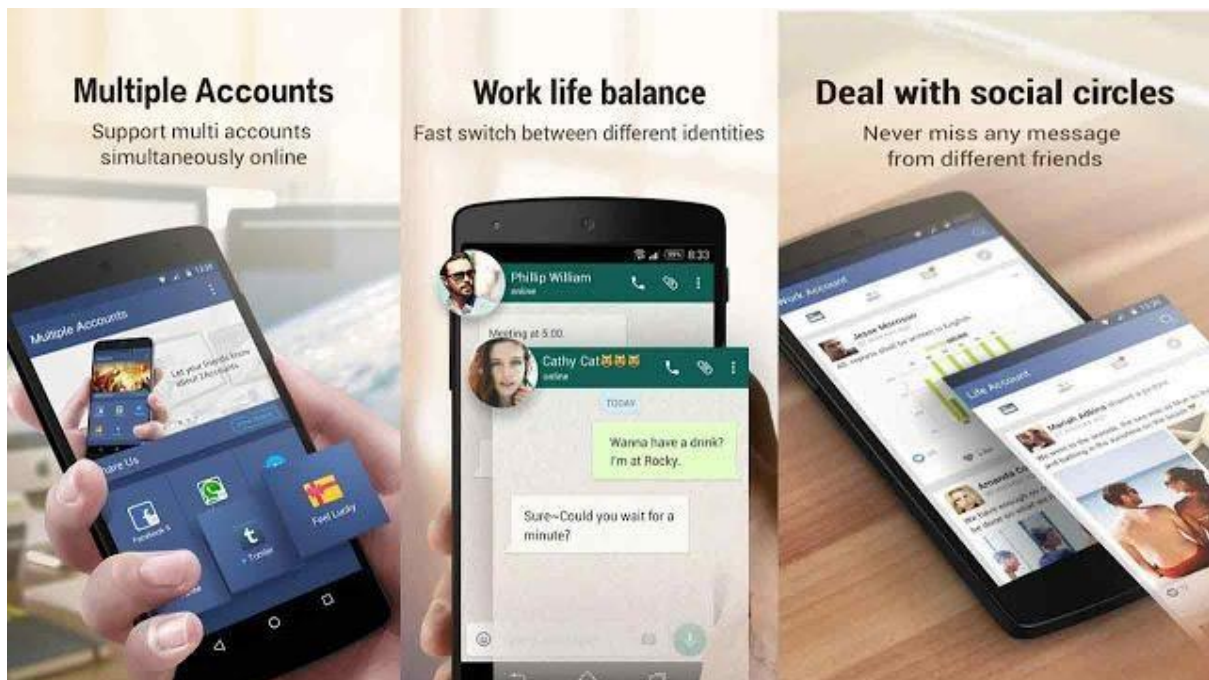


Fig no .9 Basic App Authentication Architecture

SOME CLONER APPS

Let's dive into the list of **best Clone Apps for Android**. That helps you but always your security on risk.

MULTIPLE ACCOUNTS



This is one of the best clone apps for android users. Because Multiple Accounts is a very good alternative to Parallel Space. And it also supports most of the apps like social media, games, and some pre-installed or default apps. It means you're allowed to create clones of any app without any bugs and errors. Also, create different Gmail accounts for your games and easily bifurcate your professional and personal life with the Multiple Accounts clone app.

The best part of this app is, that it consumes very little memory to operate. It means you'll get good performance from this app without compromising anything. And this app is trusted by five million people so don't worry it'll not steal your data. One last thing about this app is, that it only takes 6MB of storage from your device which is a pretty cool thing.

CLONE APP



People like it because it has dark mode enabled. Clone App is popular because features like dark mode, and zero ads. It means you can use this app without watching annoying ads with a premium feel.

MULTI PARALLEL



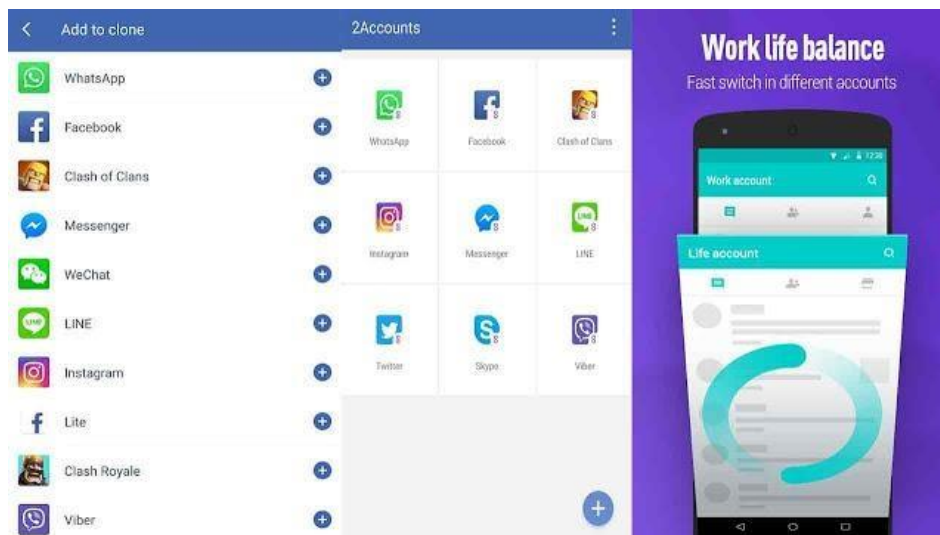
Multi Parallel is an app that allows you to make many clones of a single app. It means if you're a person like me who manage different websites with their social media accounts. Then this clone app is made for you.

DO MULTIPLE ACCOUNTS



This app duplicator is known for its simplistic user interface with a smooth design. Like the Multi Parallel app which is on our list, this clone creator also allows you to create so many duplicates of a single app. And you'd easily use different accounts simultaneously with the help of switch app shortcuts. Also, hide or lock your extra apps from people.

2ACCOUNTS



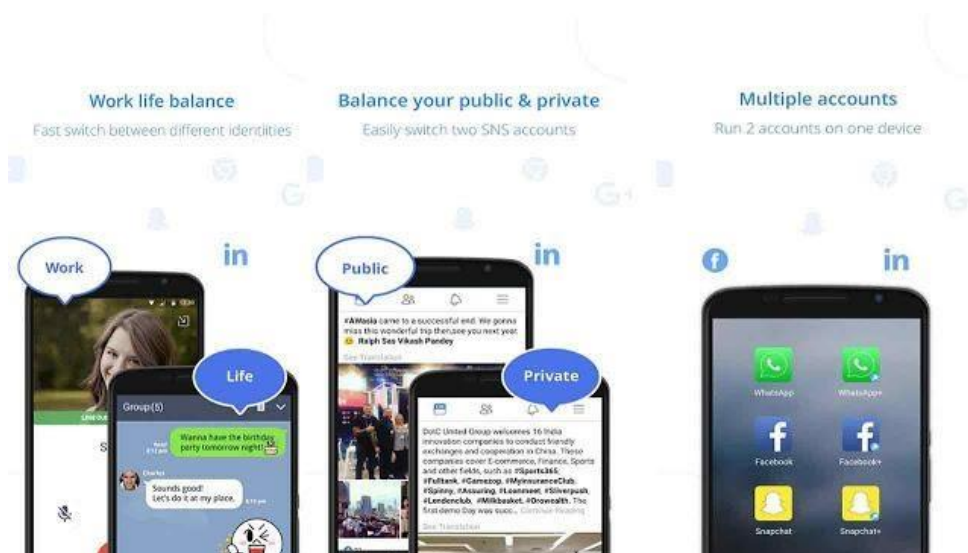
The 2Accounts is one of the best clone apps for android available in the market right now. The interface and design are very smooth with good speed. This app is good for those who want to balance their personal and professional lives. Because it helps them to identify their work and personal notifications.

DR. CLONE



Dr. Clone is a very famous and well-known app for creating multiple duplicates of your games and social media handles. In Dr. Clone you'd see two tabs that differentiate your hidden or unhidden apps which helps you to find them quickly.

PARALLEL U



Parallel U is an app that creates the first copy of your lovely games and social media handles. And this app allows you to hide your duplicates with its incognito mode or lock them with a password to protect them from unwanted people.

CLONE APP – RUN MULTIPLE ACCOUNTS

In this app, you can create multiple accounts of most of your social media handles. But unfortunately, the Clone app doesn't support so many games. So, if you're a person who wants to play games from multiple accounts then this app is not for you. But if you just want to use different social media handles then you should definitely try it.

DUAL SPACE



Dual Space is an app that is not so popular but works very well and fine. It supports many apps and games which is a pretty good thing. And also, a pretty good interface is here to give you a nice user experience. The app size is 11MB and it also consumes a very small portion of your ram and storage.

Basic Idea of Authentication (Firebase)

You can use Firebase Authentication to let your users authenticate with Firebase using their email addresses and passwords, and to manage your app's password-based accounts.

Before you begin

1. If you haven't already, add Firebase to your Android project.
2. If you haven't yet connected your app to your Firebase project, do so from the Firebase console.
3. Enable Email/Password sign-in:
 - a. In the Firebase console, open the **Auth** section.
 - b. On the **Sign in method** tab, enable the **Email/password** sign-in method and click **Save**.
4. Using the Firebase Android BoM, declare the dependency for the Firebase Authentication Android library in your **module (app-level) Gradle file** (usually app/build.gradle).

Steps

1.

```
private FirebaseAuth mAuth;
// ...
// Initialize Firebase Auth
mAuth = FirebaseAuth.getInstance();
```
2.

```
@Override
public void onStart() {
    super.onStart();
    // Check if user is signed in (non-null) and update UI
    accordingly.
    FirebaseUser currentUser = mAuth.getCurrentUser();
    if(currentUser != null){
        reload();
    }
}
```
3.

```
mAuth.createUserWithEmailAndPassword(email, password)
    .addOnCompleteListener(this, new OnCompleteListener<AuthResult>() {
        @Override
        public void onComplete(@NonNull Task<AuthResult> task) {
            if (task.isSuccessful()) {
                // Sign in success, update UI with the signed-in user's information
                Log.d(TAG, "createUserWithEmail:success");
                FirebaseUser user = mAuth.getCurrentUser();
                updateUI(user);
            } else {
```



```

        // If sign in fails, display a message to the user.
        Log.w(TAG, "createUserWithEmail:failure", task.getException());
        Toast.makeText(EmailPasswordActivity.this, "Authentication failed.",
            Toast.LENGTH_SHORT).show();
        updateUI(null);
    }
}
});

```

If the new account was created, the user is also signed in. In the call-back, you can use the `getCurrentUser` method to get the user's account data.

Authentication Encryption

The **cryptography algorithms** include the following.

In this IoT domain, security matters the most. Though there are many security mechanisms in practice, they do not hold the ability to come up with current day smart applications mainly for the software operating with resource-constraint equipment. In a consequence of this, cryptography algorithms came into practice ensuring enhanced security. So, few of the cryptographic algorithms are as follows:

Triple DES

Taking over from the conventional DES mechanism, triple DES was currently implemented in the security approaches. These algorithms permit hackers to ultimately gained the knowledge to overcome in an easy approach. This was the extensively implemented approach by many of the enterprises. Triple DES operates with 3 keys having 56 bits per each key. The entire key length is a maximum of bits, whereas experts would contend that 112-bits in key intensity is more probable. This algorithm handles to make a reliable hardware encryption answer for banking facilities and also for other industries.

RSA

One of the public-key encryption algorithms used to encrypt information transmitted through the internet. It was a widely used algorithm in GPG and PGP methodologies. RSA is classified under symmetric type of

algorithms as it performs its operation using a couple of keys. One of the keys is used for encryption and the other for decryption purposes.

AES (Advanced Encryption Standard)

The Advanced Encryption Standard (AES) is a symmetric block cipher chosen by the U.S. government to protect classified information. AES is implemented in software and hardware throughout the world to encrypt sensitive data. It is essential for government computer security, cybersecurity and electronic data protection.

The National Institute of Standards and Technology (NIST) started development of AES in 1997 when it announced the need for an alternative to the Data Encryption Standard (DES), which was starting to become vulnerable to brute-force attacks.

AES was created for the U.S. government with additional voluntary, free use in public or private, commercial or non-commercial programs that provide encryption services. However, nongovernmental organizations choosing to use AES are subject to limitations created by U.S. export control.

How AES encryption works

AES includes three block ciphers: AES-128, AES-192 and AES-256.

AES-128 uses a 128-bit key length to encrypt and decrypt a block of messages, while AES-192 uses a 192-bit key length and AES-256 a 256-bit key length to encrypt and decrypt messages. Each cipher encrypts and decrypts data in blocks of 128 bits using cryptographic keys of 128, 192 and 256 bits, respectively.

Symmetric, also known as *secret key*, ciphers use the same key for encrypting and decrypting, so the sender and the receiver must both know -- and use -- the same secret key. The government classifies information in three categories: Confidential, Secret or Top Secret. All key lengths can be used to protect the Confidential and Secret level. Top Secret information requires either 192- or 256-bit key lengths.

There are 10 rounds for 128-bit keys, 12 rounds for 192-bit keys and 14 rounds for 256-bit keys. A round consists of several processing steps

that include substitution, transposition and mixing of the input plaintext to transform it into the final output of ciphertext.

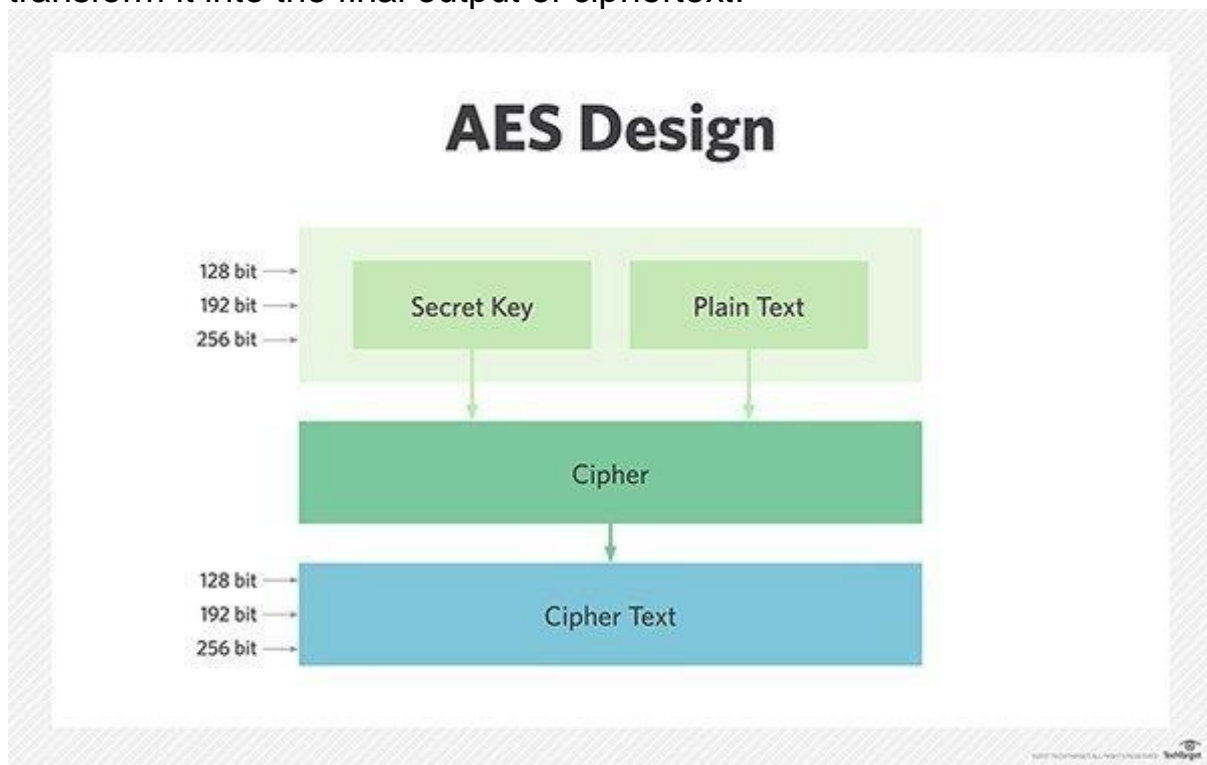


Fig No. 13

The AES encryption algorithm defines numerous transformations that are to be performed on data stored in an array. The first step of the cipher is to put the data into an array – after which, the cipher transformations are repeated over multiple encryption rounds.

The first transformation in the AES encryption cipher is substitution of data using a substitution table; the second transformation shifts data rows, and the third mixes columns.

The last transformation is performed on each column using a different part of the encryption key. Longer keys need more rounds to complete.

Difference between AES-128 and AES-256

Overall, security experts consider AES safe against brute-force attacks, in which all possible key combinations are checked until the correct key is found. However, the key size employed for encryption needs to be large enough so that it cannot be cracked by modern computers, even considering advancements in processor speeds based on Moore's law.

A 256-bit encryption key is significantly more difficult for brute-force attacks to guess than a 128-bit key; however, because the latter takes

so long to guess, even with a huge amount of computing power, it is unlikely to be an issue for the foreseeable future, as a hacker would need to use quantum computing to generate the necessary brute force.

Still, 256-bit keys also require more processing power and can take longer to execute. When power is an issue -- particularly on small devices -- or where latency is likely to be a concern, 128-bit keys are likely to be a better option.

When hackers want to access a system, they will aim for the weakest point, which is typically not the encryption, regardless of whether it's a 128-bit key or a 256-bit key. Users should make sure the software under consideration does what they want it to do, that it protects user data in the way it's expected to and that the overall process has no weak points.

Additionally, there should be no Gray areas or uncertainty about data storage and handling. For example, if data resides in the cloud, users should know the location of the cloud. Most importantly, the security software that has been selected should be easy to use to ensure those users do not need to perform unsecure workarounds to do their jobs.

AES vs. RSA

AES is used widely for protecting data at rest. Applications for AES include self-encrypting disk drives, database encryption and storage encryption. On the other hand, the RSA (Rivest-Shamir-Adleman) algorithm is often used in web browsers to connect to websites, in virtual private network (VPN) connections and in many other applications.

Unlike AES, which employs symmetric encryption, RSA is the base of asymmetric cryptography. Symmetric encryption involves converting plaintext to ciphertext using the same key, or secret key, to encrypt and decrypt it.

On the other hand, the term *asymmetric* comes from the fact that there are two related keys used for encryption: a public and a private key.

AES vs. DES

The U.S. government developed **DES** algorithms more than 40 years ago to ensure government systems all used the same, secure standard

to facilitate interconnectivity. DES served as the linchpin of government cryptography for years until 1999, when researchers broke the algorithm's 56-bit key using a distributed computer system. In 2000, the U.S. government chose to use AES to protect classified information. DES is still used in some instances for backward compatibility.

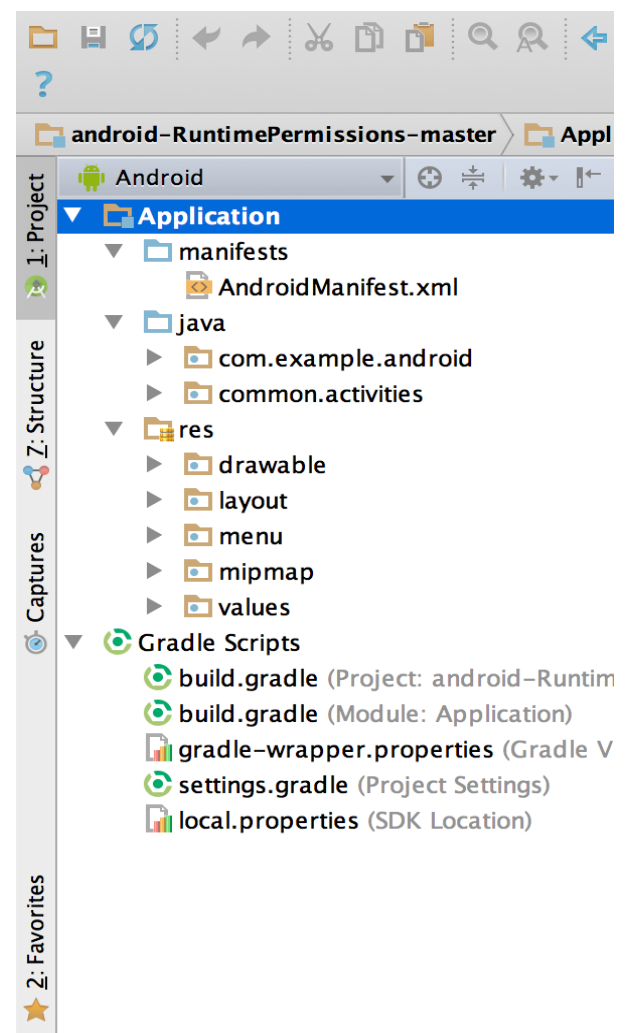
The two standards are both symmetric block ciphers, but AES is more mathematically efficient. The main benefit of AES lies in its key length options. The time required to crack an encryption algorithm is directly related to the length of the key used to secure the communication -- 128-bit, 192-bit or 256-bit keys. Therefore, AES is exponentially stronger than the 56-bit key of DES. AES encryption is also significantly faster, so it is ideal for applications, firmware and hardware that require low latency or high throughput.

Protruding Project Tools →

- 1) Android Studio**
- 2) Eclipse Ide**
- 3) Firebase**

Android Studio → Android Studio is the official Integrated Development Environment (IDE) for Android app development, based on IntelliJ IDEA. On top of IntelliJ's powerful code editor and developer tools, Android Studio offers even more features that enhance your productivity when building Android apps, such as:

- A flexible Gradle-based build system
- A fast and feature-rich emulator
- A unified environment where you can develop for all Android devices
- Apply Changes to push code and resource changes to your running app without restarting your app
- Code templates and GitHub integration to help you build common app features and import sample code
- Extensive testing tools and frameworks
- Lint tools to catch performance, usability, version compatibility, and other problems
- C++ and NDK support
- Built-in support for Google Cloud Platform, making it easy to integrate Google Cloud Messaging and App Engine.



Project structure

Each project in Android Studio contains one or more modules with source code files and resource files. Types of modules include:

- Android app modules
- Library modules
- Google App Engine modules

By default, Android Studio displays your project files in the Android project view, as shown in figure 1. This view is organized by modules to provide quick access to your project's key source files.

All the build files are visible at the top level under **Gradle Scripts** and each app module contains the following folders:

- **manifests**: Contains the AndroidManifest.xml file.
- **java**: Contains the Java source code files, including JUnit test code.
- **res**: Contains all non-code resources, such as XML layouts, UI strings, and bitmap images.

The Android project structure on disk differs from this flattened representation. To see the actual file structure of the project, select **Project** from the **Project** dropdown (in figure 1, it's showing as **Android**).

You can also customize the view of the project files to focus on specific aspects of your app development. For example, selecting the **Problems** view of your project displays links to the source files containing any recognized coding and syntax errors, such as a missing XML element closing tag in a layout file.

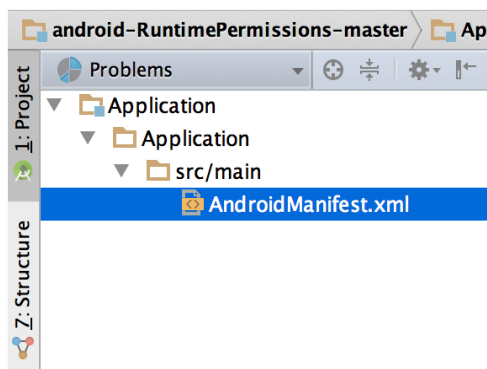
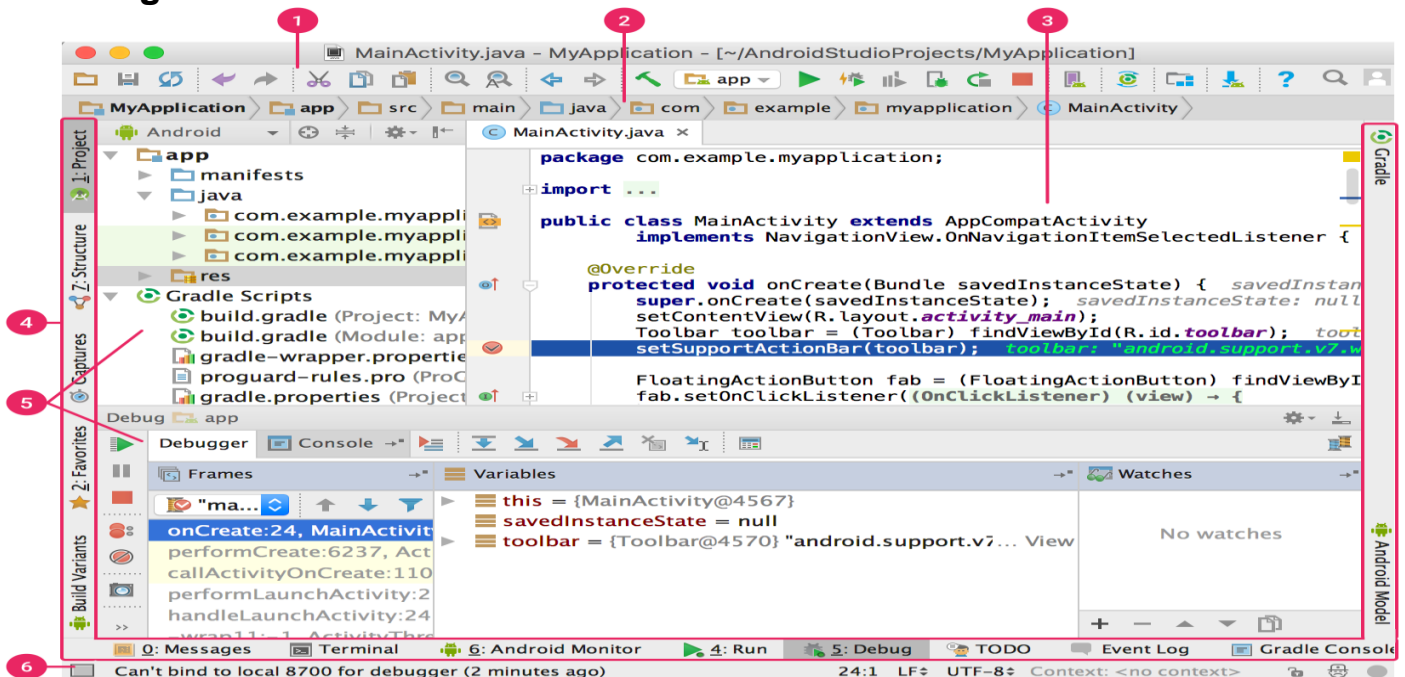


Figure 2. The project files in Problem's view, showing a layout file with a problem.

The user interface

Figure 3. The Android Studio main window.



1. The toolbar lets you carry out a wide range of actions, including running your app and launching Android tools.
2. The navigation bar helps you navigate through your project and open files for editing. It provides a more compact view of the structure visible in the Project window.
3. The editor window is where you create and modify code. Depending on the current file type, the editor can change. For example, when viewing a layout file, the editor displays the Layout Editor.
4. The tool window bar runs around the outside of the IDE window and contains the buttons that allow you to expand or collapse individual tool windows.
5. The tool windows give you access to specific tasks like project management, search, version control, and more. You can expand them and collapse them.


You can organize the main window to give yourself more screen space by hiding or moving toolbars and tool windows. You can also use keyboard shortcuts to access most IDE features.

At any time, you can search across your source code, databases, actions, elements of the user interface, and so on, by double-pressing

the Shift key, or clicking the magnifying glass in the upper right-hand corner of the Android Studio window. This can be very useful if, for example, you are trying to locate a particular IDE action that you have forgotten how to trigger.

Tool windows

Instead of using pre-set perspectives, Android Studio follows your context and automatically brings up relevant tool windows as you work. By default, the most commonly used tool windows are pinned to the tool window bar at the edges of the application window.

- To expand or collapse a tool window, click the tool's name in the tool window bar. You can also drag, pin, unpin, attach, and detach tool windows.
- To return to the current default tool window layout, click **Window > Restore Default Layout** or customize your default layout by clicking **Window > Store Current Layout as Default**.
- To show or hide the entire tool window bar, click the window icon  in the bottom left-hand corner of the Android Studio window.
- To locate a specific tool window, hover over the window icon and select the tool window from the menu.

2) Eclipse In the context of computing, Eclipse is an integrated development environment (IDE) for developing applications using the Java programming language and other programming languages such as C/C++, Python, PERL, Ruby etc.

The Eclipse platform which provides the foundation for the Eclipse IDE is composed of plug-ins and is designed to be extensible using additional plug-ins. Developed using Java, the Eclipse platform can be used to develop rich client applications, integrated development environments and other tools. Eclipse can be used as an IDE for any programming language for which a plug-in is available.

The Java Development Tools (JDT) project provides a plug-in that allows Eclipse to be used as a Java IDE, PyDev is a plugin that allows Eclipse to be used as a Python IDE, C/C++ Development Tools (CDT) is a plug-in that allows Eclipse to be used for developing application using C/C++, the Eclipse Scala plug-in allows Eclipse to be used an

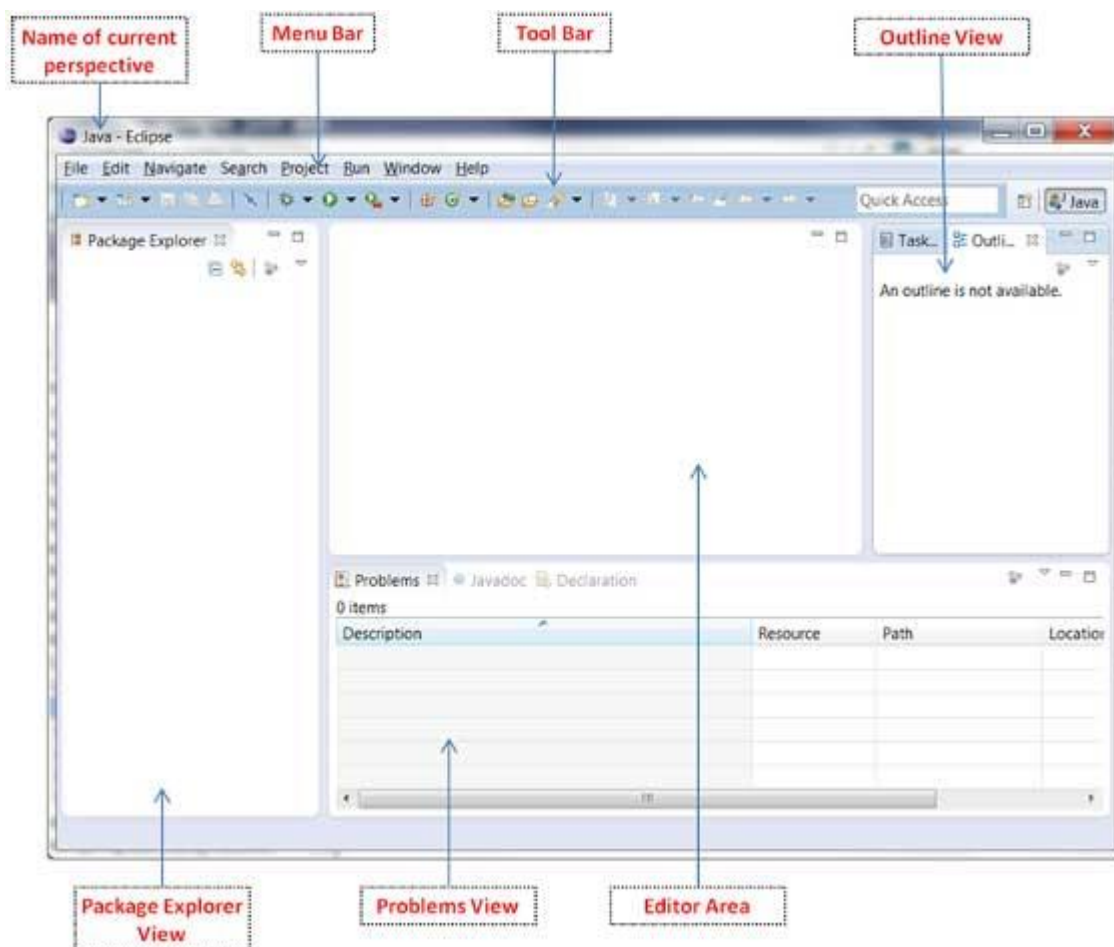
IDE to develop Scala applications and PHP eclipse is a plug-in to eclipse that provides complete development tool for PHP.

Parts of an Eclipse Window

The major visible parts of an eclipse window are –

- Views
- Editors (all appear in one editor area)
- Menu Bar
- Toolbar

An eclipse perspective is the name given to an initial collection and arrangement of views and an editor area. The default perspective is called java. An eclipse window can have multiple perspectives open in it but only one perspective can be active at any point of time. A user can switch between open perspectives or open a new perspective. A perspective controls what appears in some menus and tool bars.



A perspective has only one editor area in which multiple editors can be open. The editor area is usually surrounded by multiple views. In general, editors are used to edit the project data and views are used to view the project metadata. For example, the package explorer shows the java files in the project and the java editor is used to edit a java file.

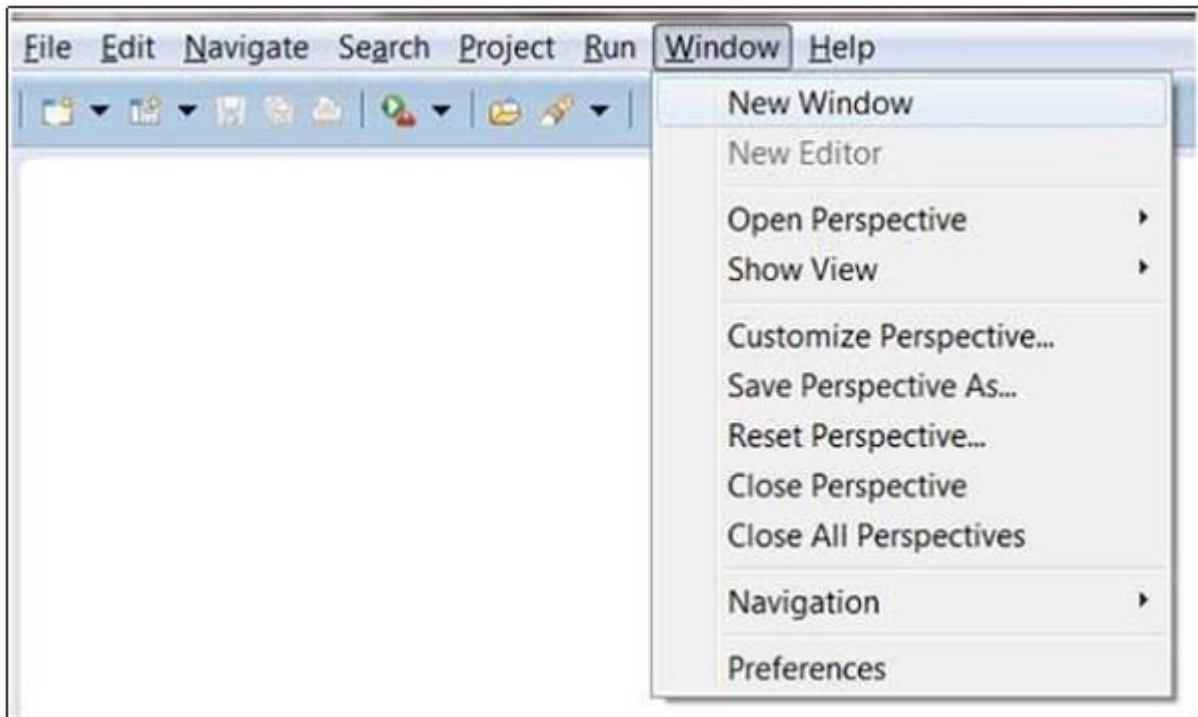
The eclipse window can contain multiple editors and views but only one of them is active at any given point of time. The title bar of the active editor or view looks different from all the others.

The UI elements on the menu bar and tool bar represent commands that can be triggered by an end user.

Typical Eclipse Menus

The typical menus available on the menu bar of an Eclipse window are-

- File menu
- Edit menu
- Navigate menu
- Search menu
- Project menu
- Run menu
- Window menu
- Help menu



Plug-ins can add new menus and menu items. For example, when the java editor is open you will see the Source menu and when the XML editor is open, you will see the Design menu.

Brief Description of Menus

Sr.No	Menu Name & Description
1	File The File menu allows you to open files for editing, close editors, save editor content and rename files. Among the other things, it also allows you to import and export workspace content and shutdown Eclipse.
2	Edit The Edit menu presents items like copy & paste.

3	Source The Source menu is visible only when a java editor is open. It presents a number of useful menu items related to editing java source code.
4	Navigate The Navigate menu allows you to quickly locate resources and navigate to them.
5	Search The Search menu presents items that allow you to search the workspace for files that contain specific data.
6	Project The menu items related to building a project can be found on the Project menu.
7	Run The menu items on the Run menu allow you to start a program in the run mode or debug mode. It also presents menu items that allow you to debug the code.
8	Window The Window menu allows you to open and close views and perspectives. It also allows you to bring up the Preferences dialog.

4) **Firebase** →

Google Firebase is a Google-backed application development software that enables developers to develop iOS, Android and Web apps. Firebase provides tools for tracking analytics, reporting and fixing app crashes, creating marketing and product experiment.

Firebase offers a number of services, including:

- Analytics – Google Analytics for Firebase offers free, unlimited reporting on as many as 500 separates events. Analytics

presents data about user behaviour in iOS and Android apps, enabling better decision-making about improving performance and app marketing.

- Authentication – Firebase Authentication makes it easy for developers to build secure authentication systems and enhances the sign-in and onboarding experience for users. This feature offers a complete identity solution, supporting email and password accounts, phone auth, as well as Google, Facebook, GitHub, Twitter login and more.
- Cloud messaging – Firebase Cloud Messaging (FCM) is a cross-platform messaging tool that lets companies reliably receive and deliver messages on iOS, Android and the web at no cost.
- Realtime database – the Firebase Realtime Database is a cloud-hosted NoSQL database that enables data to be stored and synced between users in real time. The data is synced across all clients in real time and is still available when an app goes offline.
- Crashlytics – Firebase Crashlytics is a real-time crash reporter that helps developers track, prioritize and fix stability issues that reduce the quality of their apps.
- Performance – Firebase Performance Monitoring service gives developers insight into the performance characteristics of their iOS and Android apps to help them determine where and when the performance of their apps can be improved.
- Test lab – Firebase Test Lab is a cloud-based app-testing infrastructure. With one operation, developers can test their iOS or Android apps across a variety of devices and device configurations. They can see the results, including videos, screenshots and logs, in the Firebase console.

Use cases

Firebase use cases include:

- Create onboarding flows – developers can give users a quick, intuitive sign-in process using Firebase Authentication. They allow users to sign into their apps via their Google, Twitter, Facebook or GitHub accounts in less than five minutes. Developers can also track each step of their onboarding flows to enhance the user experience. Additionally, developers can use Google Analytics for Firebase to log events at each step of their onboarding flows, create funnels to determine where users are dropping off and use remote configuration to make changes to their apps to see how those changes affect conversions.
- Customize a “welcome back” screen – developers can use personalization to give every user the best experience by customizing the initial screen based on a user’s preferences, usage history, location or language. Developers can define audiences based, in part, on user behaviours and show targeted content to each audience.
- Progressively roll out new features – developers can launch new features with minimal risk by first testing those features on a few users to see how they work and how users respond.

Security leak

In June 2018, mobile security firm App authority reported that thousands of iOS and Android mobile apps were exposing more than 113 GBs of data via 2,271 misconfigured Firebase databases.

Beginning in January 2018, App authority researchers scanned mobile apps that used Firebase systems to store user data and analysed communications pattern for requests made to Firebase domains.

After scanning over 2.7 million Android and iOS apps, researchers identified 28,502 mobile apps (1,275 iOS and 27,227 Android) that connected and stored data inside Firebase backends.

Of these, 3,046 apps (600 iOS and 2,446 Android) saved data inside 2,271 misconfigured Firebase databases that enabled anyone to view their content.

PROCESS FLOW CHART OF THE PROJECT

1. Getting Authentication points

```
import java.text.*;
class GFG
{

    // Function to find the circle on
    // which the given three points lie
    static void findCircle(int x1, int y1,
                          int x2, int y2,
```



```

        int x3, int y3)
{
    int x12 = x1 - x2;
    int x13 = x1 - x3;

    int y12 = y1 - y2;
    int y13 = y1 - y3;

    int y31 = y3 - y1;
    int y21 = y2 - y1;

    int x31 = x3 - x1;
    int x21 = x2 - x1;

    //  $x_1^2 - x_3^2$ 
    int sx13 = (int)(Math.pow(x1, 2) -
        Math.pow(x3, 2));

    //  $y_1^2 - y_3^2$ 
    int sy13 = (int)(Math.pow(y1, 2) -
        Math.pow(y3, 2));

    int sx21 = (int)(Math.pow(x2, 2) -
        Math.pow(x1, 2));

    int sy21 = (int)(Math.pow(y2, 2) -
        Math.pow(y1, 2));

    int f = ((sx13) * (x12)
        + (sy13) * (x12)
        + (sx21) * (x13)
        + (sy21) * (x13))
        / (2 * ((y31) * (x12) - (y21) * (x13)));
    int g = ((sx13) * (y12)
        + (sy13) * (y12)
        + (sx21) * (y13)
        + (sy21) * (y13))
        / (2 * ((x31) * (y12) - (x21) * (y13)));
    int c = -(int)Math.pow(x1, 2) - (int)Math.pow(y1, 2) -
        2 * g * x1 - 2 * f * y1;

    // eqn of circle be  $x^2 + y^2 + 2gx + 2fy + c = 0$ 
    // where centre is (h = -g, k = -f) and radius r
    // as  $r^2 = h^2 + k^2 - c$ 
    int h = -g;
    int k = -f;
    int sqr_of_r = h * h + k * k - c;

```

```

// r is the radius
double r = Math.sqrt(sqr_of_r);
DecimalFormat df = new DecimalFormat("#.#####");
System.out.println("Centre = (" + h + "," + k + ")");
System.out.println("Radius = " + df.format(r));
}
public static void getp1(){ return x1+y1);
public static void getp2(){ return x2+y2);

// Driver code
public static void main (String[] args)
{
int x1 = Integer.parseInt(singin.mail.getText().toString());
int y1 = Integer.parseInt(singin.mail.getText().toString());
int x2 = Integer.parseInt(singin.pswd.getText().toString());
int y2 = Integer.parseInt(singin.pswd.getText().toString());
int x3 = auth.reference.getData().child("un_id");
int y3 = auth.reference.getData().child("un_id");
    findCircle(x1, y1, x2, y2, x3, y3);
}
}

```

Given three coordinates that lie on a circle, (x_1, y_1) , (x_2, y_2) , and (x_3, y_3) . The task is to find the equation of the circle and then print the centre and the radius of the circle. Equation of circle in general form is $x^2 + y^2 + 2gx + 2fy + c = 0$ and in radius form is $(x - h)^2 + (y - k)^2 = r^2$, where (h, k) is the centre of the circle and r is the radius.

2. Passing arguments into Firebase database

```

auth.signInWithEmailAndPassword
    mail = cir.getp1();
    pswd = cir.getp2();
(binding.mail.getText().toString(),binding.pswd.getText().toString()).ad
dOnCompleteListener
    (new OnCompleteListener<AuthResult>() {
        @Override
        public void onComplete(@NonNull
@org.jetbrains.annotations.NotNull Task<AuthResult> task) {
            pd.dismiss();
            if (task.isSuccessful()) {

```

```

        Intent intent = new
Intent(signin.this,MainActivity.class);
        startActivity(intent);
    }

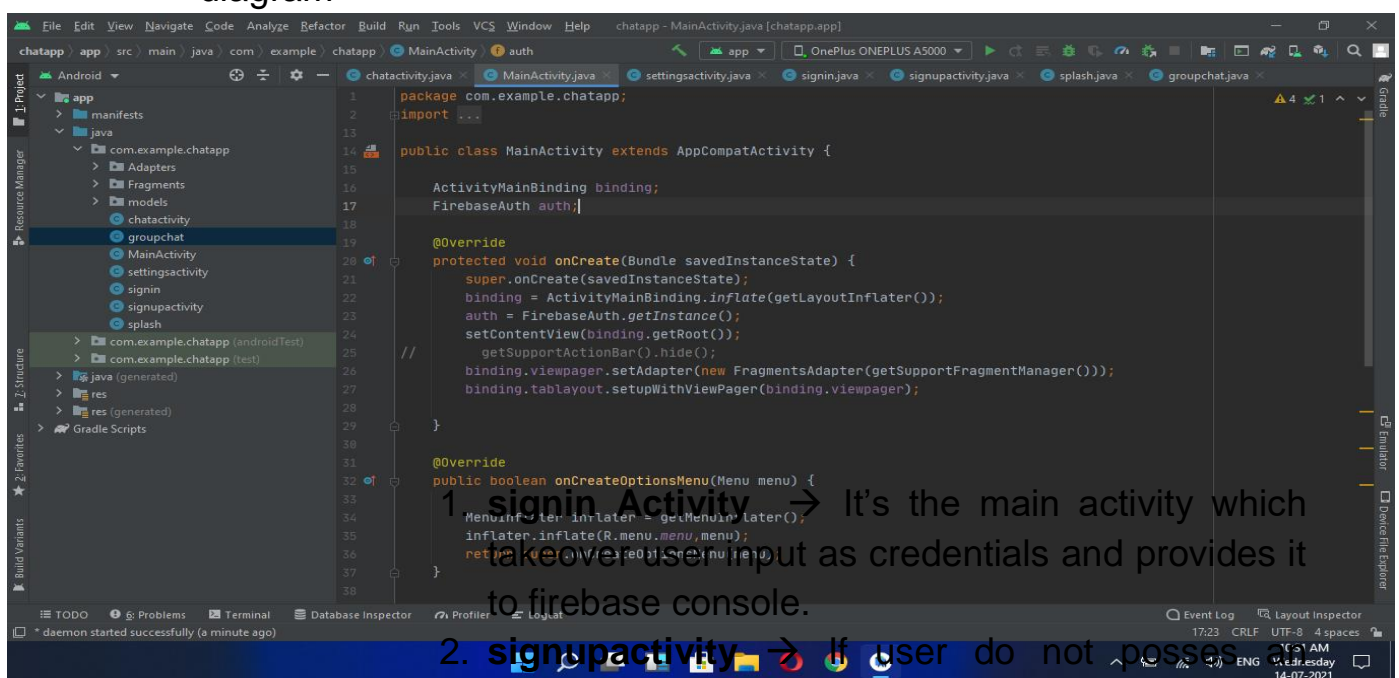
    else{

        Toast.makeText(signin.this,
task.getException().getMessage(), Toast.LENGTH_SHORT).show();
    }
});

```

3. App Creation (Project Structure) →

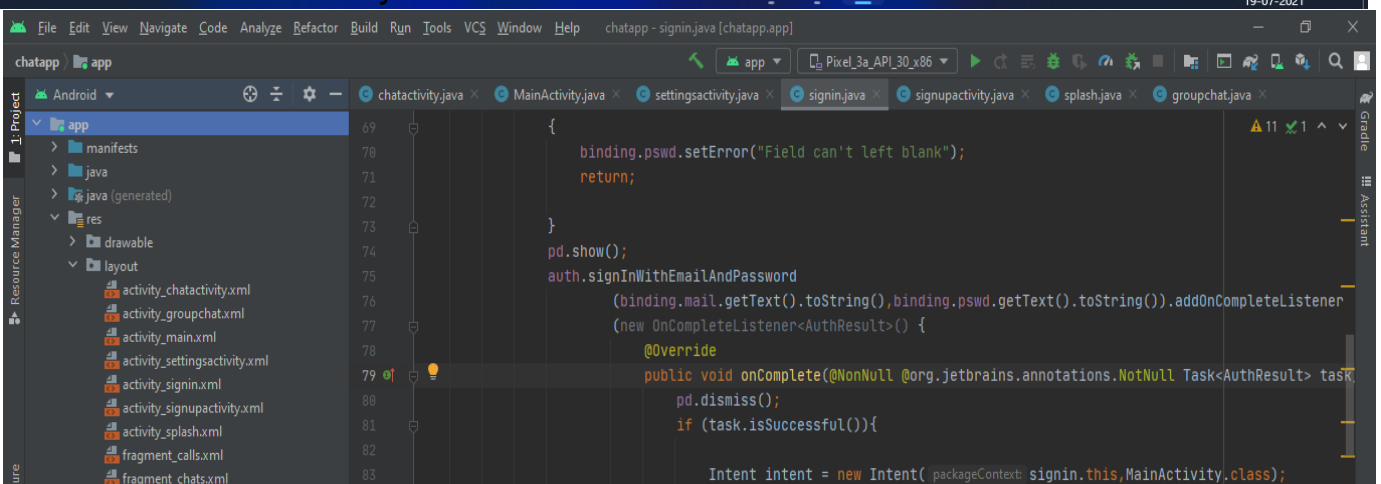
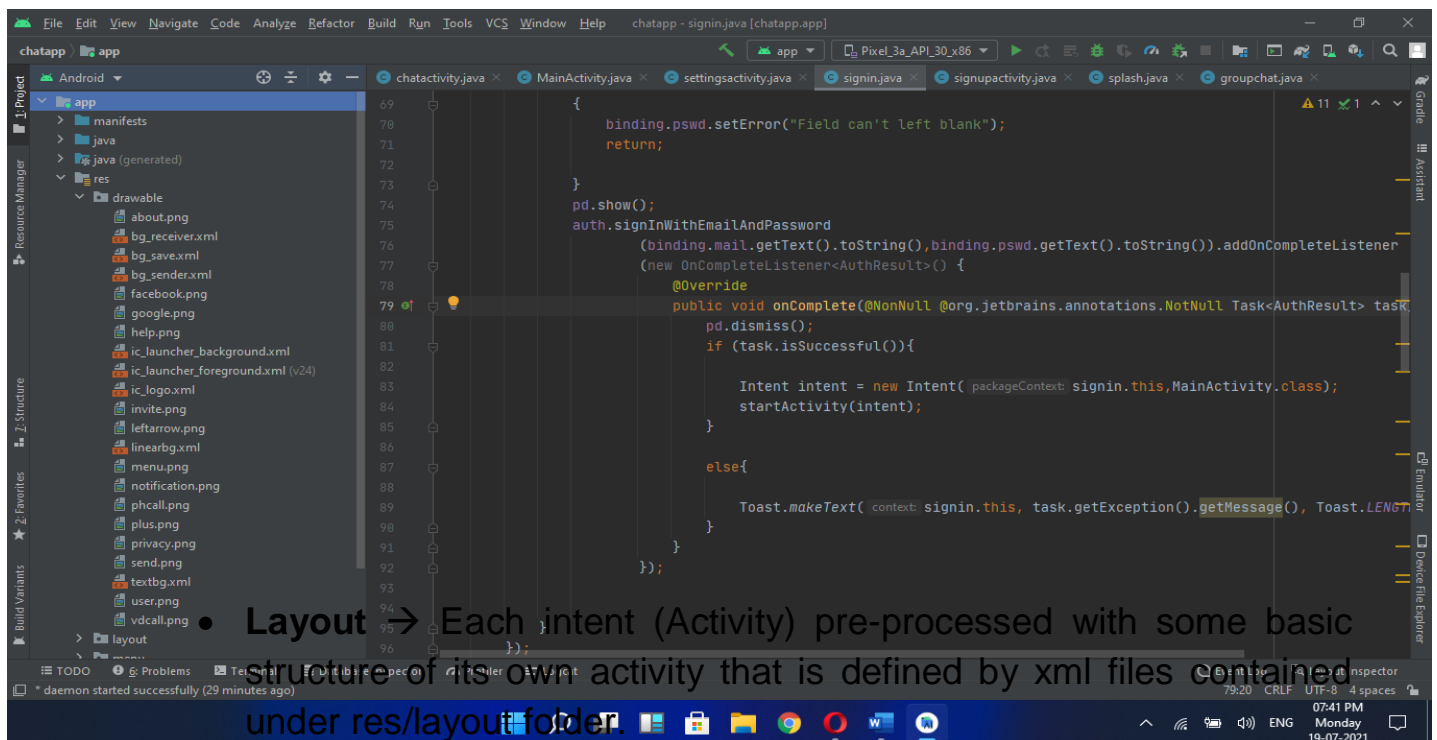
- **Activities** → Activity is a basic view structure made of combination of views and view groups.
We used 7 activities with 3 fragments and 3 adapter providing essential setters and getters in model class.
The basic hierarchy of activity class as shown in the following diagram



2. **signuactivity** → If user do not posses an account it has to create its account first and this task holds by this activity.
3. **splash** → This activity basically provides initial anime screen for android app. Using Lottie json object animations.

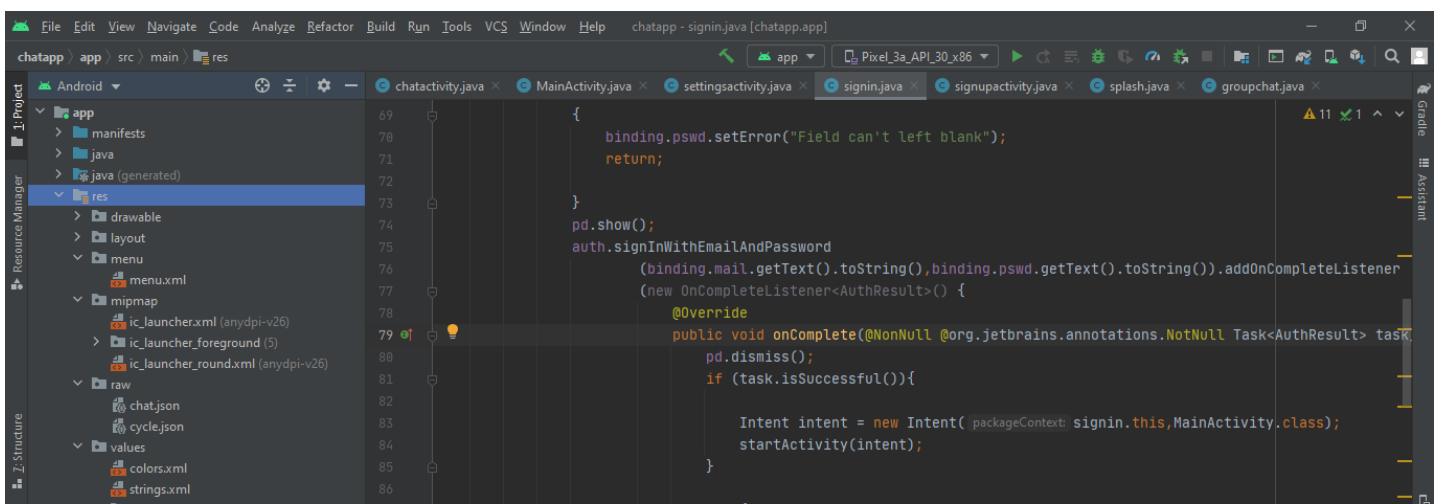
4. **groupchat** → An activity provides a platform to chat with multiple users. In this activity not a single user required to verify identity again and again because here relation of chat is one to many.
5. **MainActivity** → This activity comes into play when user sing in with credentials and firebase plus attribute model allows the task.
6. **settingsactivity** → In this activity the basic of user profile has been updated.
7. **chatactivity** → The conversation and tagging each text with parent ID done by in this activity.

- **drawable** → the drawable under res folder used for app decoration, button designing and content making. Most of the drawable are free to use and vectored into image of 24px or 64px.



- 1) **samplesender** → It's a resource file which deals with sender chat box in chat activity with the usage of chat adapter.
- 2) **samlereceiver** → Similar to samplesender it deals with receiver chat box in chat activity with the usage of chat adapter.
- 3) **Fragments** → all the fragments lie on MainActivity by the help of recyclerview.

- Other → This structure represents colour value used in project. Named string used for various views. Icon used for app on display and MainActivity menu.



- Gradle → These are dependencies used by us during the project development.

Major one are as follows →

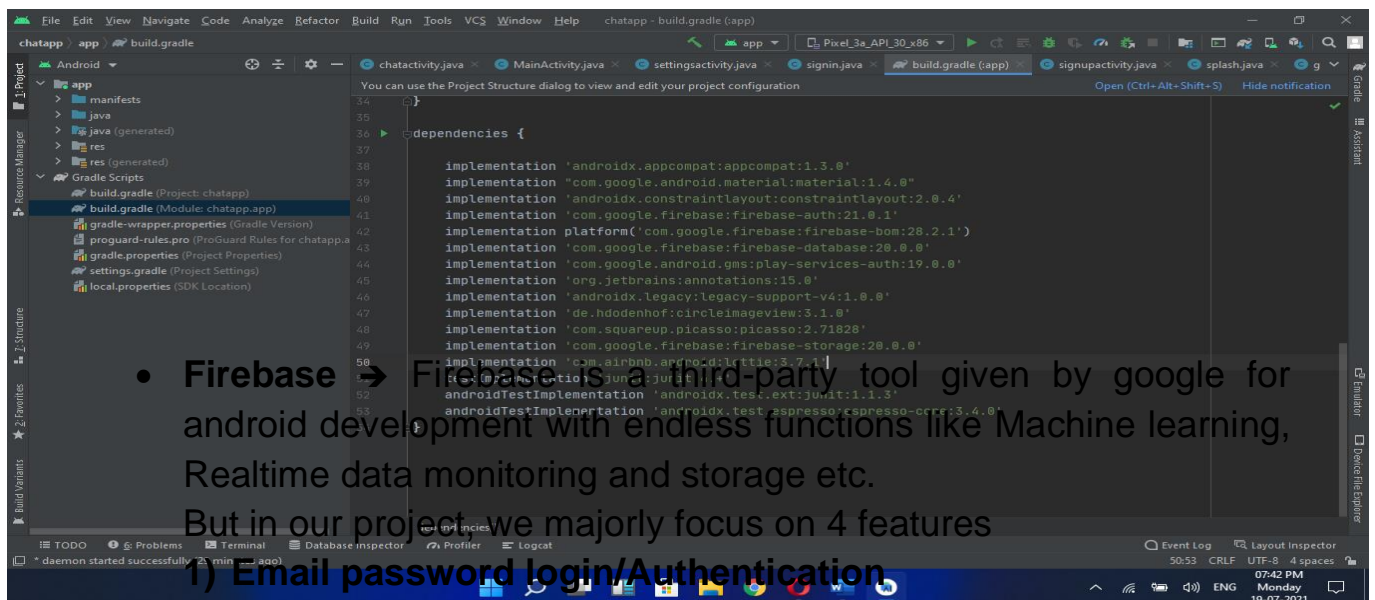
Firestore authentication implementation

Firestore Realtime database implementation

Picasso implementation

Lottie implementation

BOM implementation



Each of above step require some common algo to be perform first

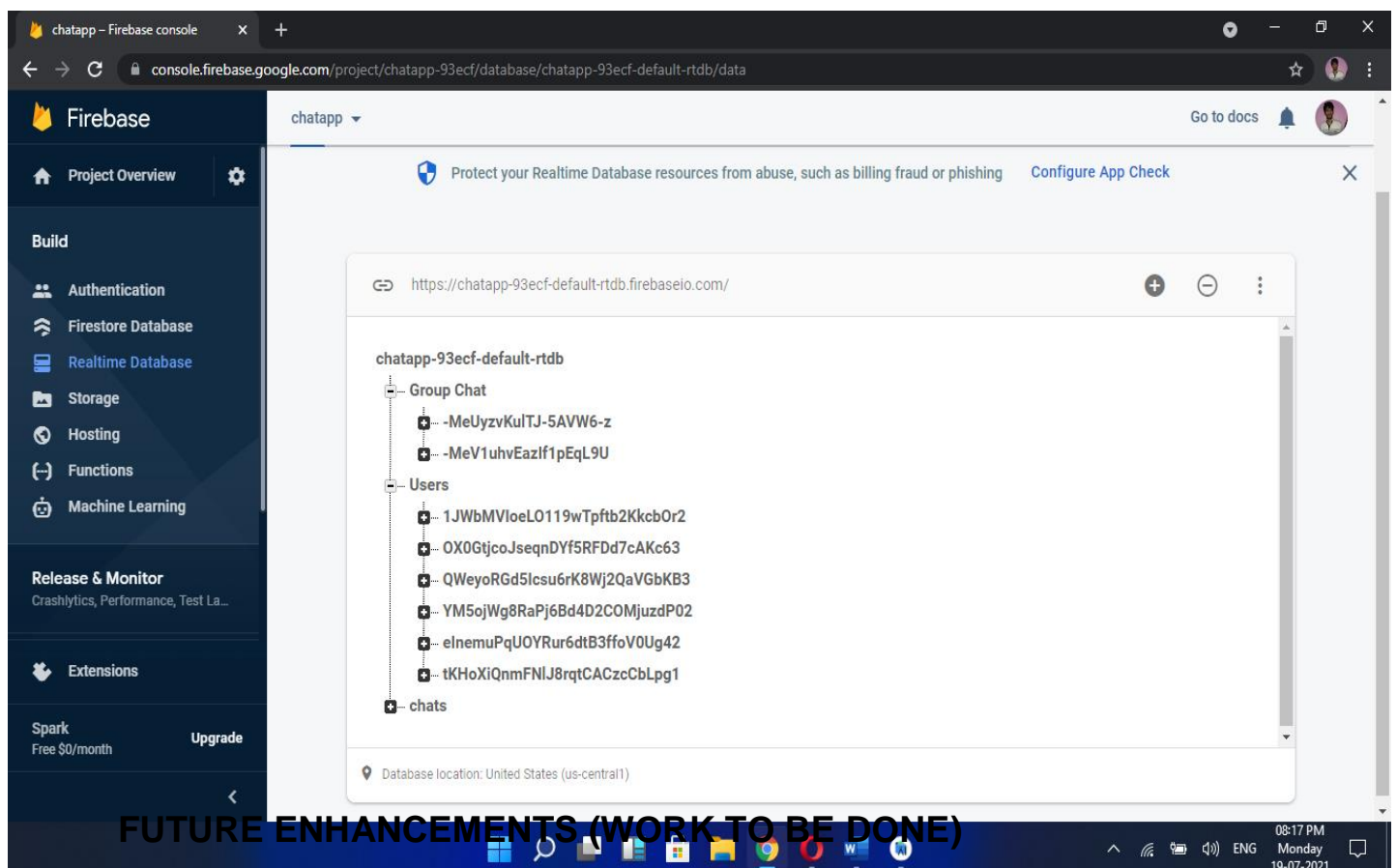
Firebase token generation using SHA1

Google authentication using mail

Adding database storage and authentication dependencies into Gradle build.

Creating Childs/nodes in database.

Firebase database works on tree hierarchy mechanism.



- In future work of this project, we will try to make it a well-developed complete app based on new features.
- We will also implement it as user friendly software
- May turn up these modules in collaboration as a client-side testing.
- We will try to add even more functionalities to make it more user benefiting.

Bibliography

Web-sites

- 1) ILETS Mentor**
- 2) vpnmentor**
- 3) Helpnetsecurity**
- 4)The Economic Times**
- 5)Business Line**
- 6)Business Help**