

# Soft Expo Chats

## Documentation 2021



# Instruction:1

Import the project into your Android studio.

To import the code into your Android studio do the following steps:

- i)Unzip the Project File
- ii)Open Android Studio exe
- iii)Click on Open an Existing Project on Android Studio
- iv)Select our Android Project

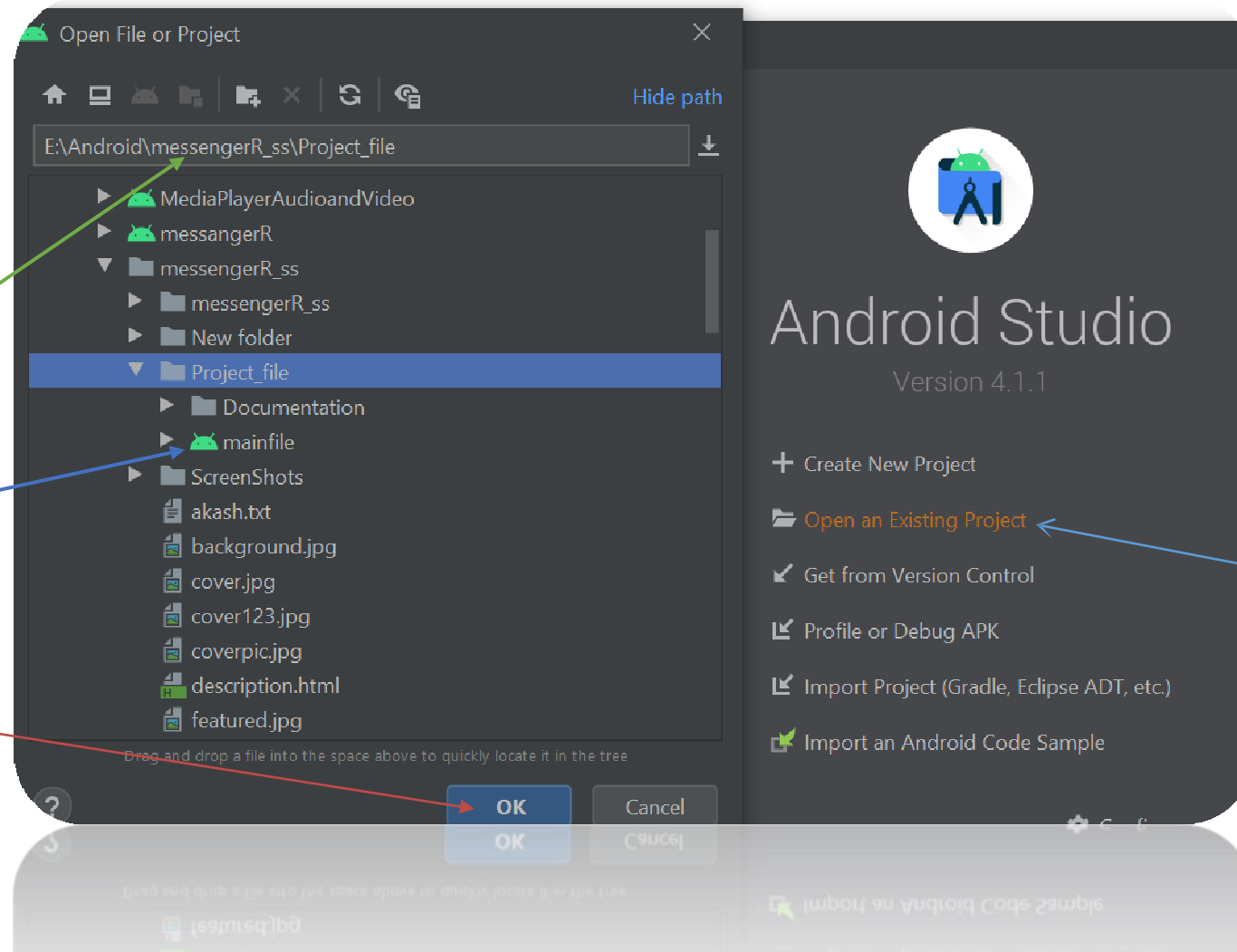
[In Android Studio Opening dialog,when you choose the option "Open an Existing Project" ,You have to put the path of project root path. Then Select our project file ... See Examples]

# Examples

step1

Unzip the Project file

Step 3  
put the Root location of  
the unzip file  
and Select our main  
project  
Then  
click ok. Files will be  
imported automatically  
on Android Studio within  
less than one minute



Step 2

After Open Android Studio  
Click on Open an Existing  
Project

## Instruction 2:

Changing the basics of project like app name, package name, icon, color, splashscreen, ui designs as you preferences...

=>You can use ours also but you have to change package name must for creating your own network....then go to Instruction 3...

=>if you don't change packagename and want to use our package name and our network just skip Instruction 3. Jump to Instruction 4.

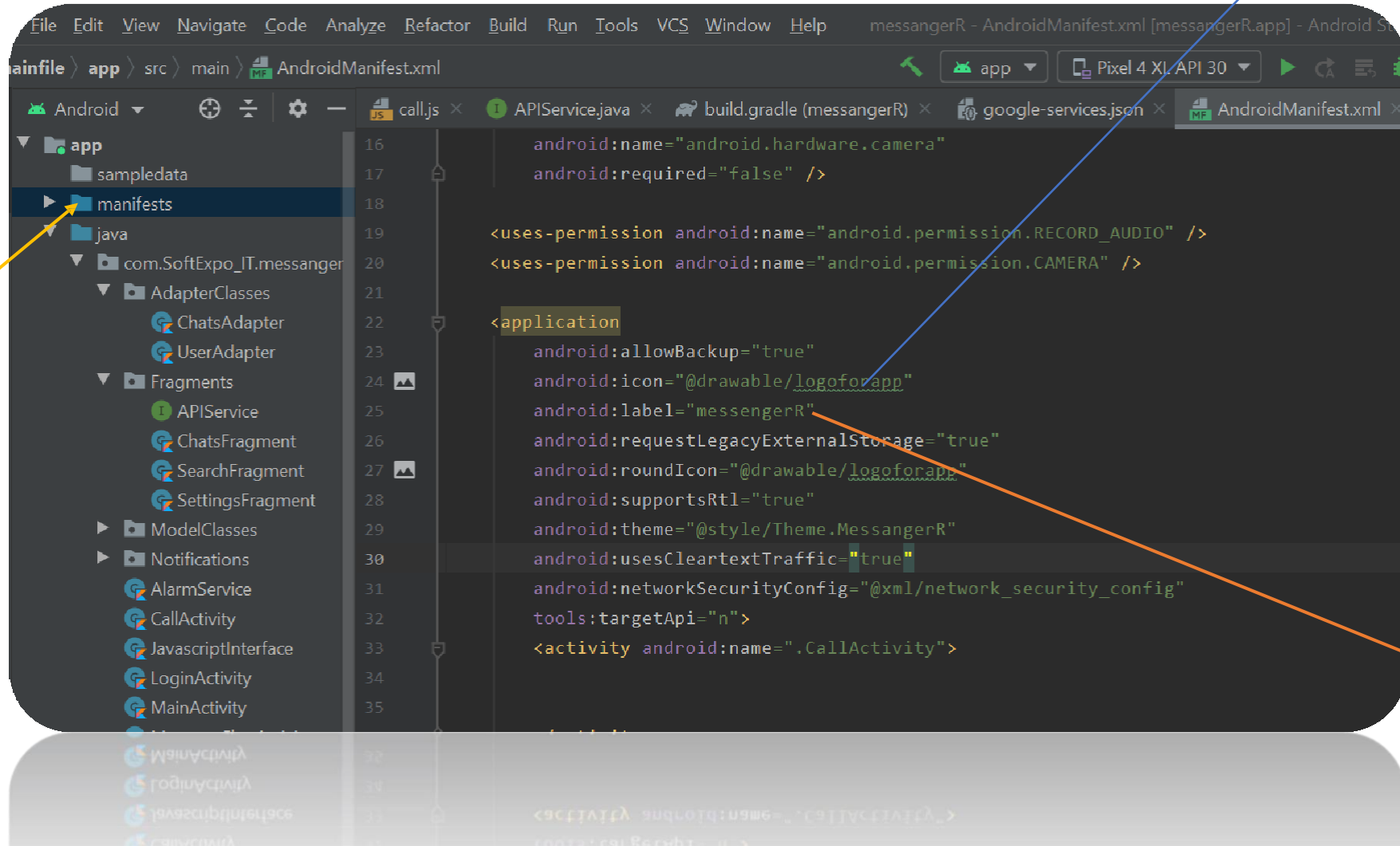
⇒You can change the app\_name and icon from manifest file...if you want to change colors and ui designs, just go to XML files and change these...

⇒import your designs at drawable folder and set these designs from xml

File...Our Designs are open source...You can also use these. Want to Change Splashscreen...just replace the splashscreen image with your image with the same name at drawable folder...That's it...

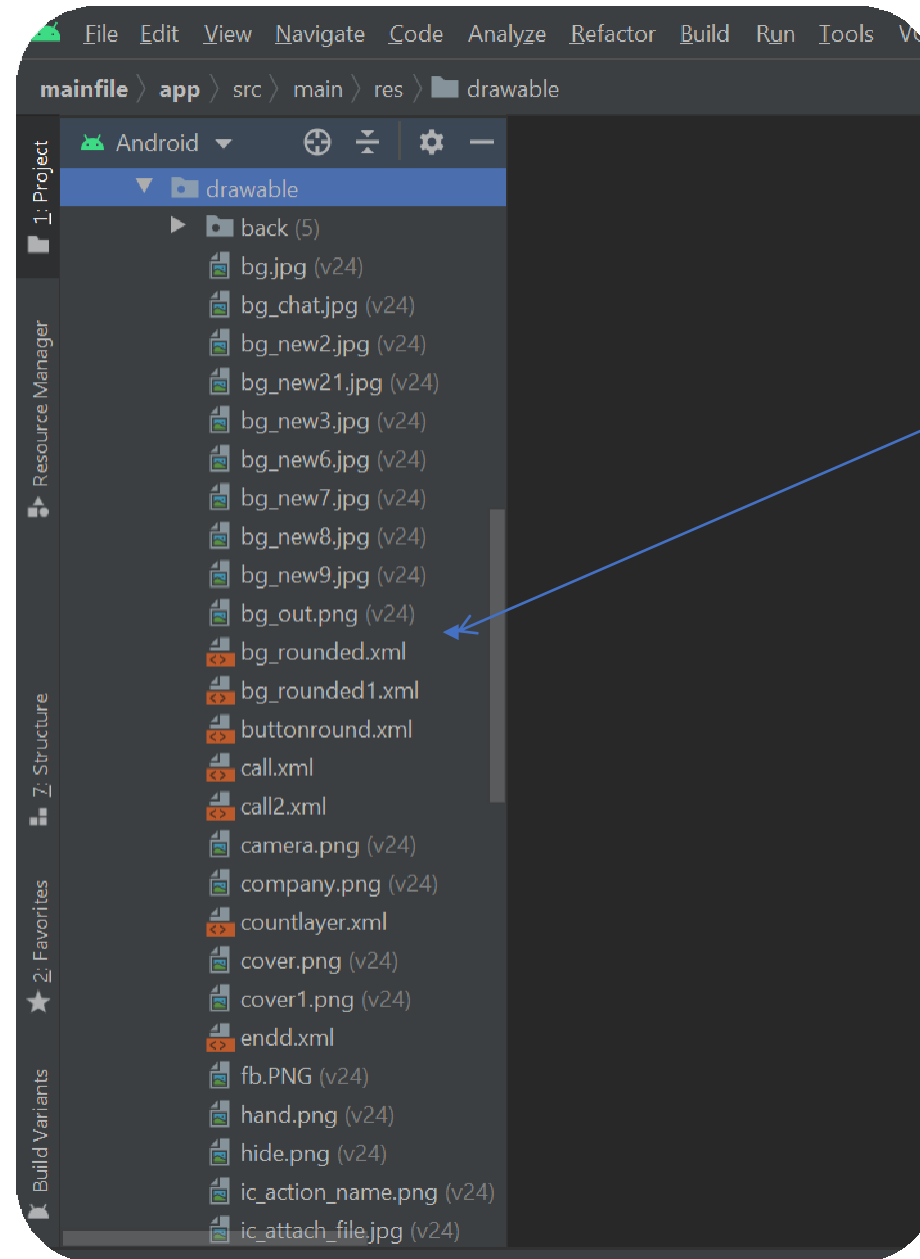
# Samplaes

Import your icon in drawable folder and set your app icon here



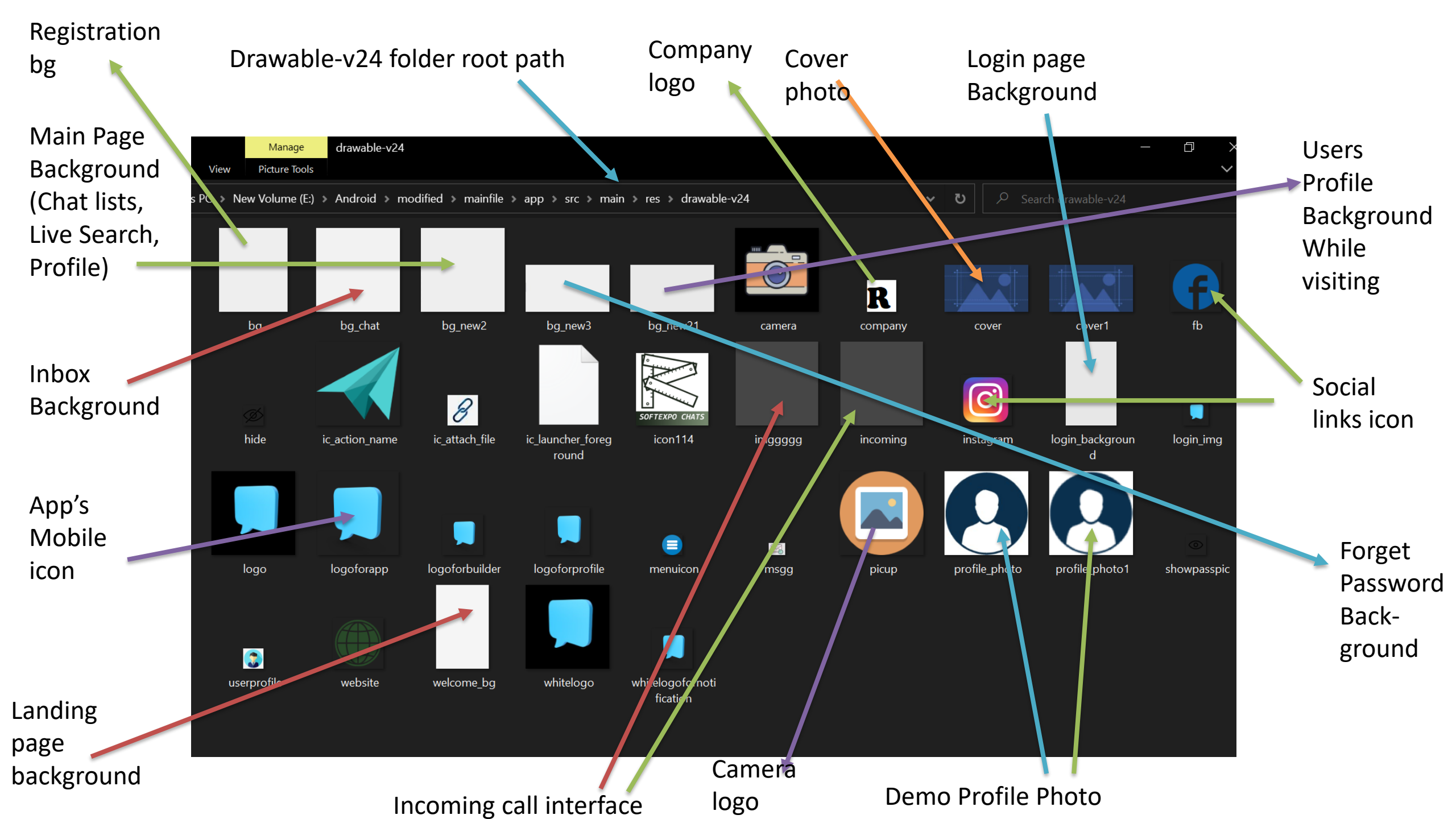
Step 1  
Go to  
manifests file

You can set your  
App name here



This is our Drawable Folder.  
You can copy your image and just  
paste it here in Drawable Folder.  
The image will be imported  
Automatically. Then Go to xml files  
and you can set your Designs  
there. Icons are also imported  
here in Drawable File

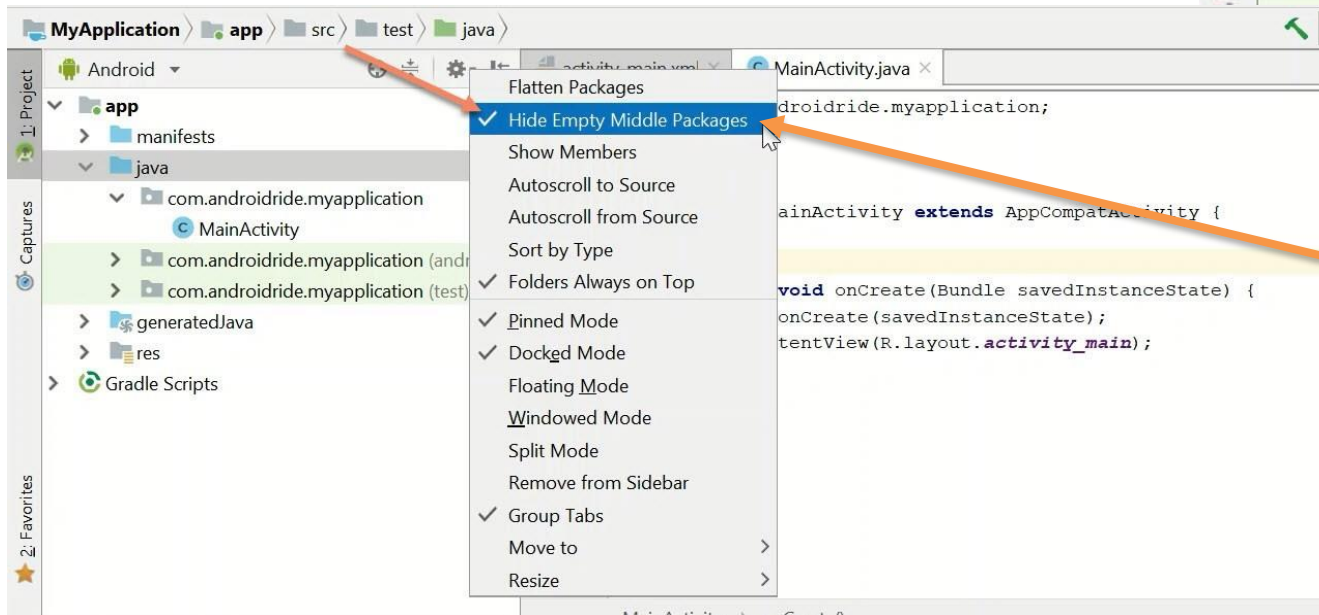
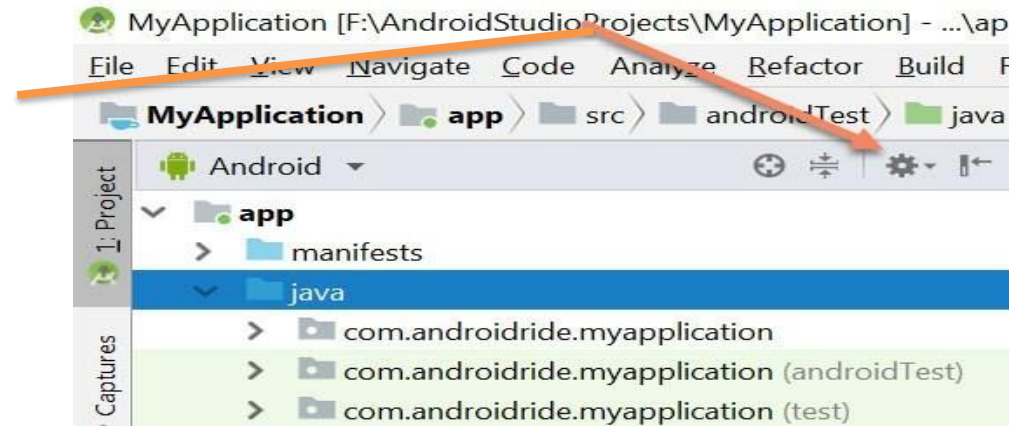
**You can Replace our photo with  
your selected photo with the  
same name. EX:**  
**Our landing page's image name is  
bg.jpg... You can copy your photo  
And paste it into drawable folder  
then delete our bg.jpg picture  
and rename your photo as bg.jpg  
...Your app's landing image will  
be changed and set with your  
picture...take a look on our  
drawable folder...**



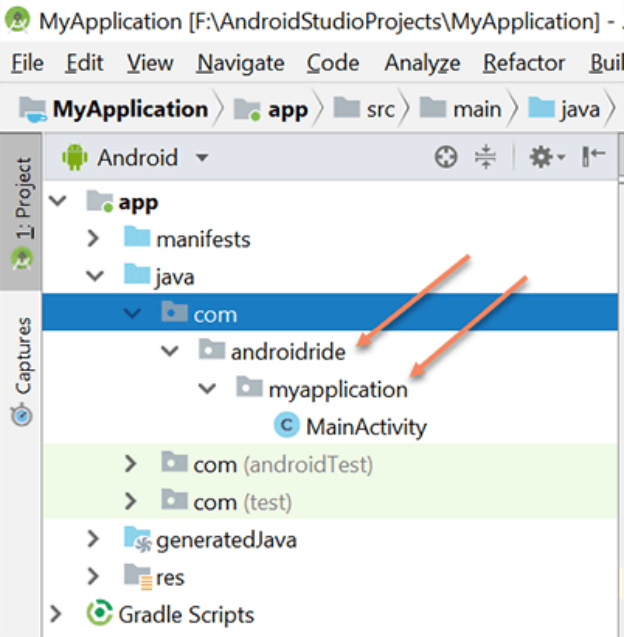
# Methods To Change Or Rename Android App Package Name in Android Studio

In this example, we will change package name **com.androidride.myapplication** to **com.xyz.yourapplication**.

Step 1: Open Android Studio project, Select package name or Java, Click the gear icon.

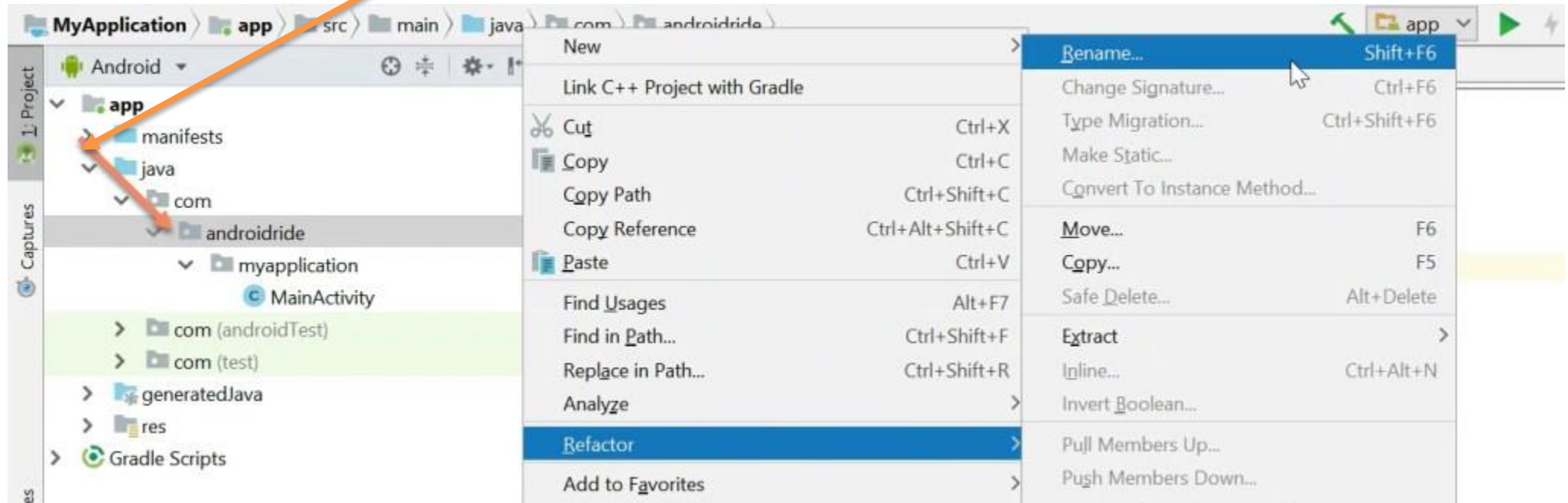


Step 2: Deselect Compact Middle Packages. This breaks your folder structure into parts.

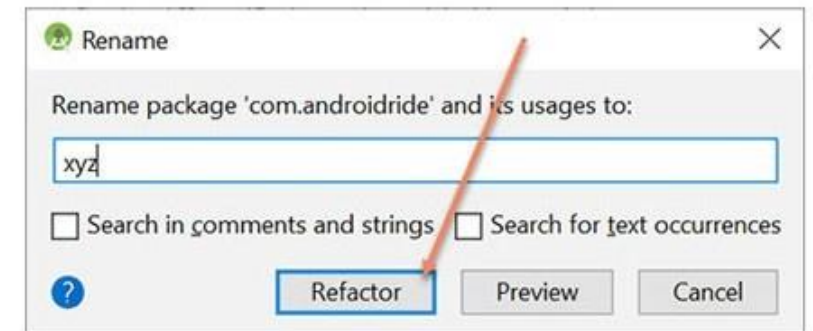
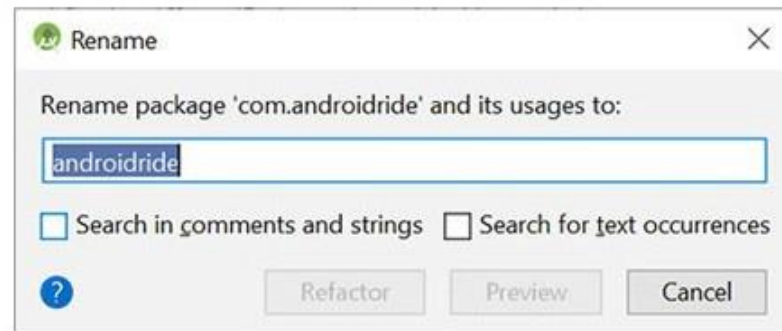
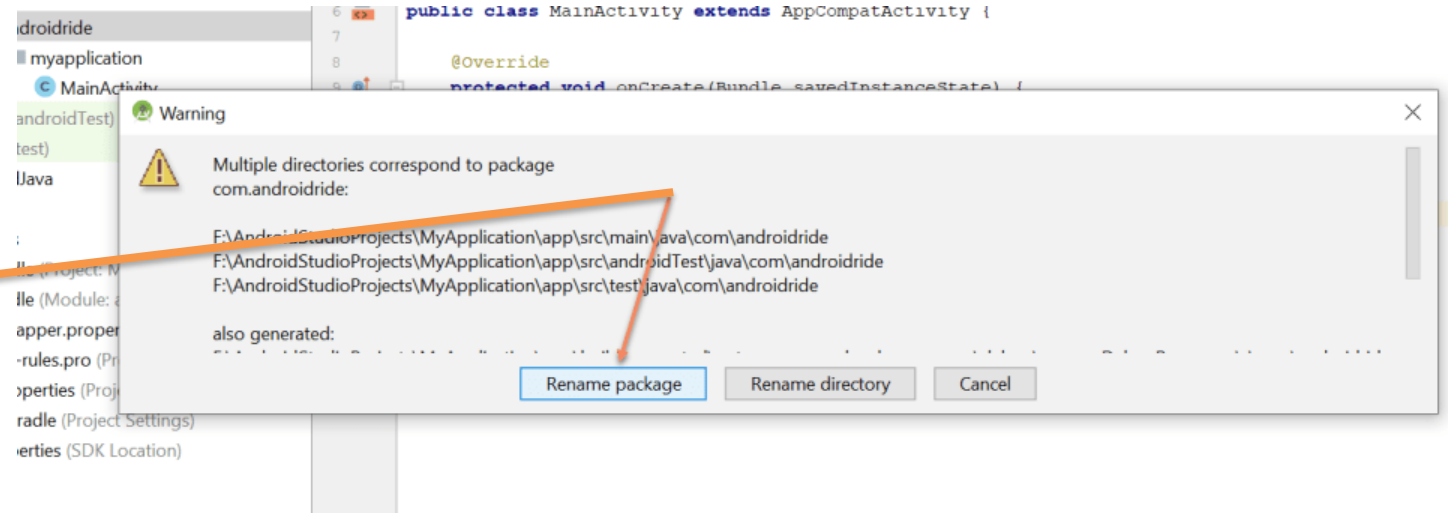


The package is broken down.

Step 3 : Right Click on AndroidRide, **Refactor -> Rename** . You can also use **Shift + F6**.



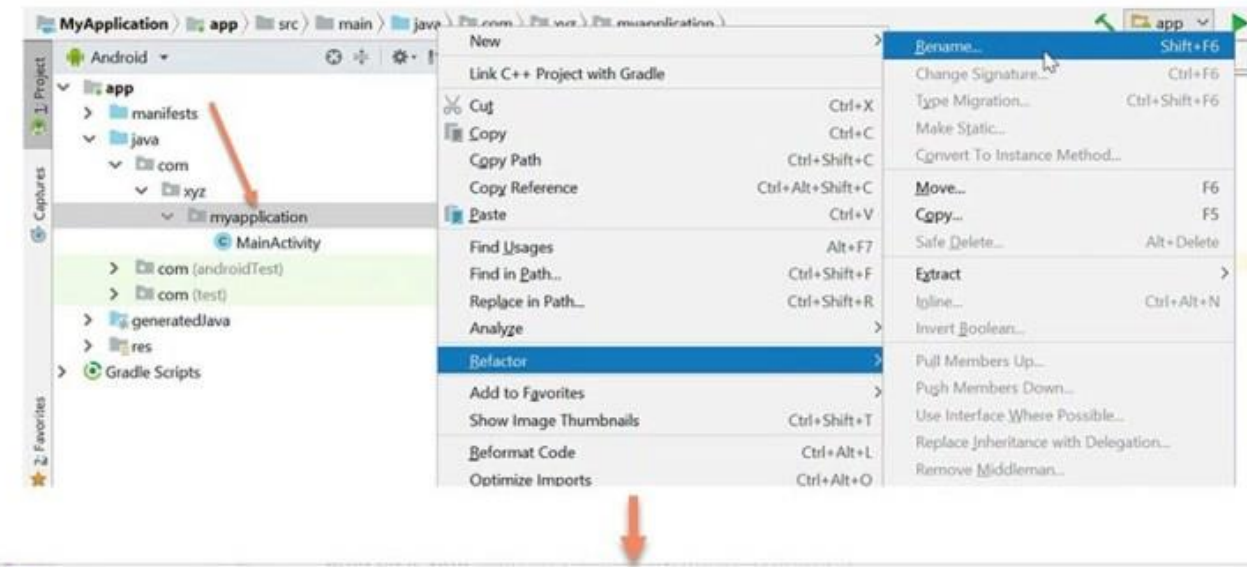
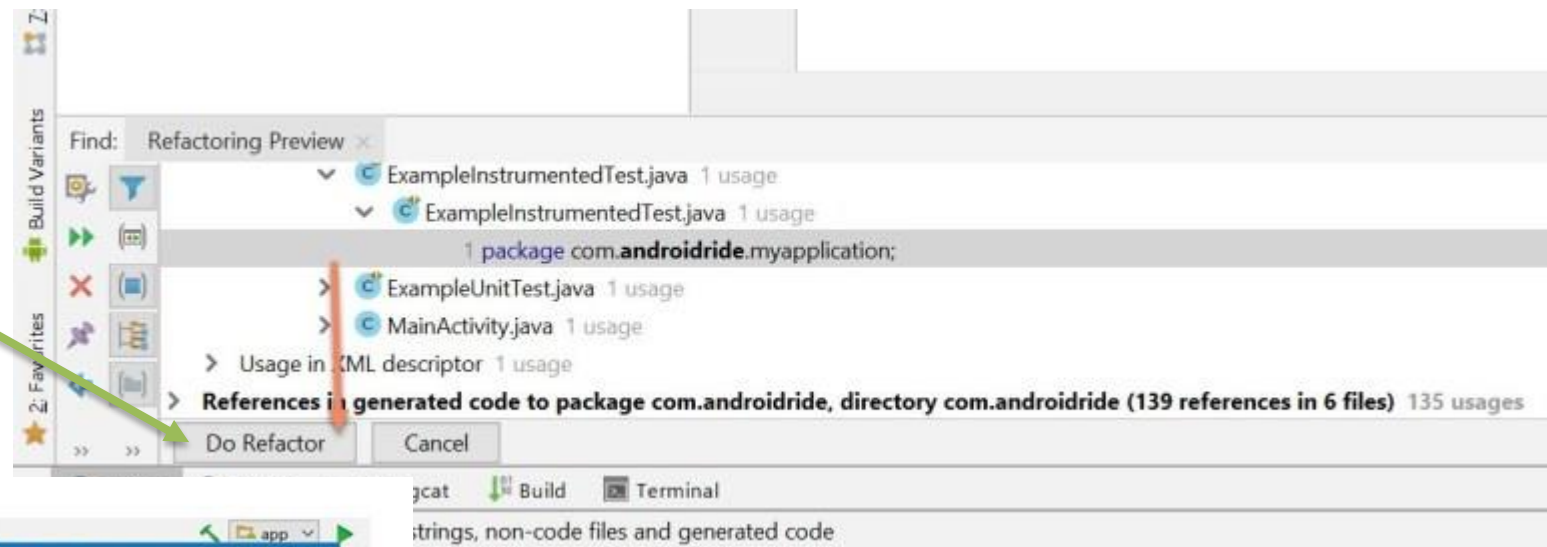
Step 4 : Click on **Rename package**



Step 5 : Clear AndroidRide and put new Name in it.  
Here **"xyz"**

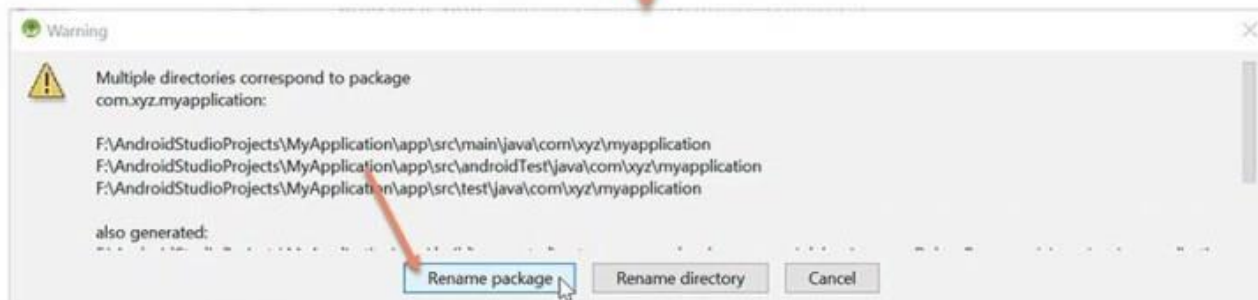
## Step 6:

If This dialog will come, click on do Refactor otherwise not need

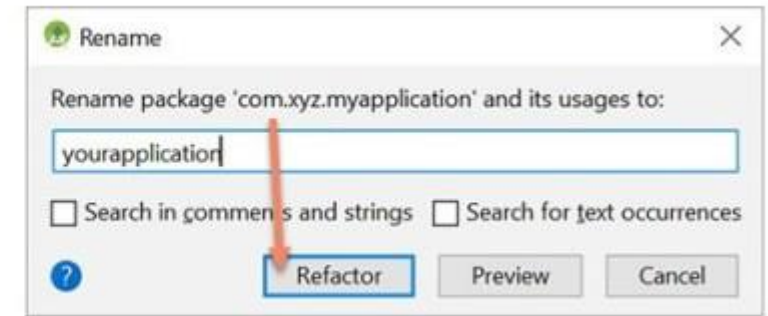
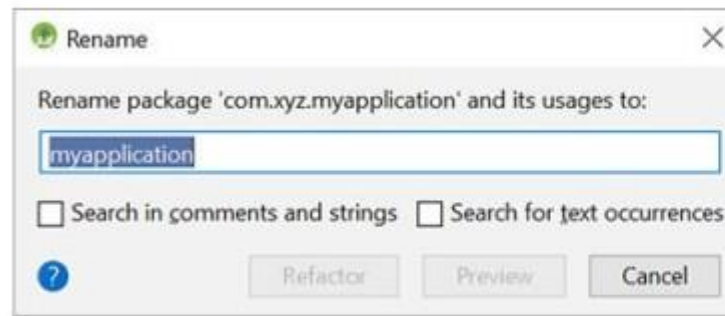


## Step:7

- Repeat the same process.
- Right click on **myapplication**, Refactor -> **Rename** and **Rename package**.

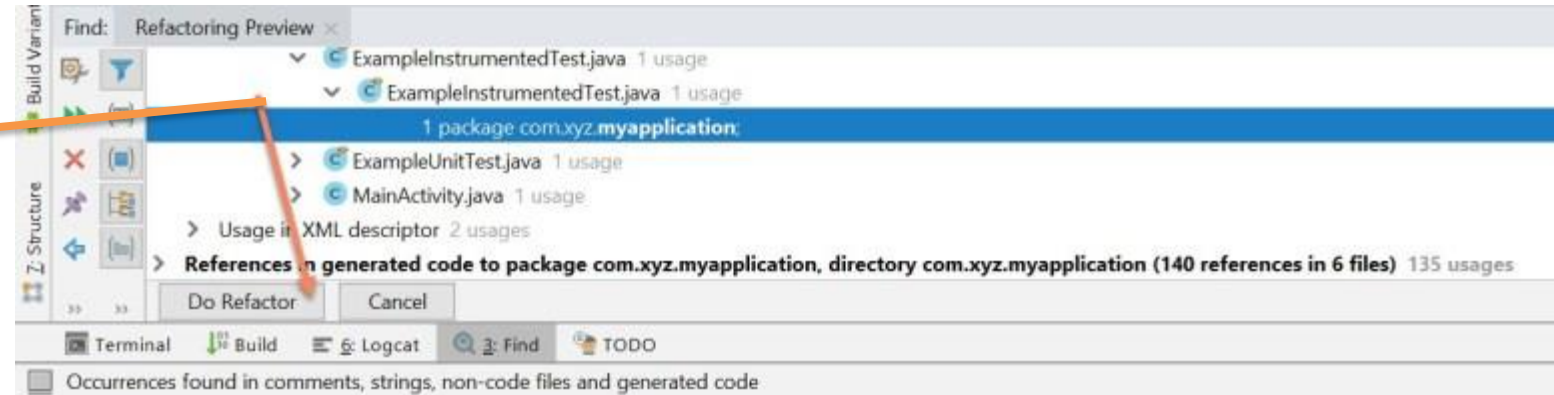
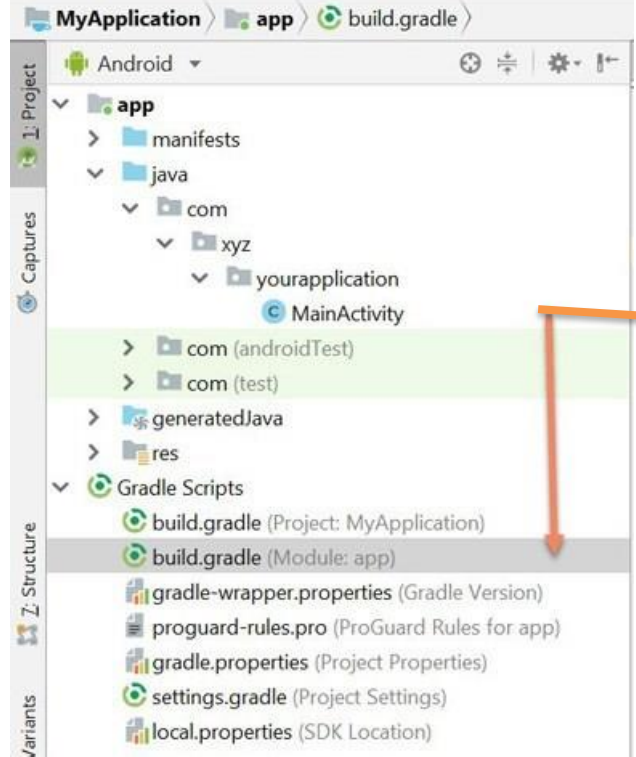


Step 8:  
Change **myapplication** to  
**yourapplication**

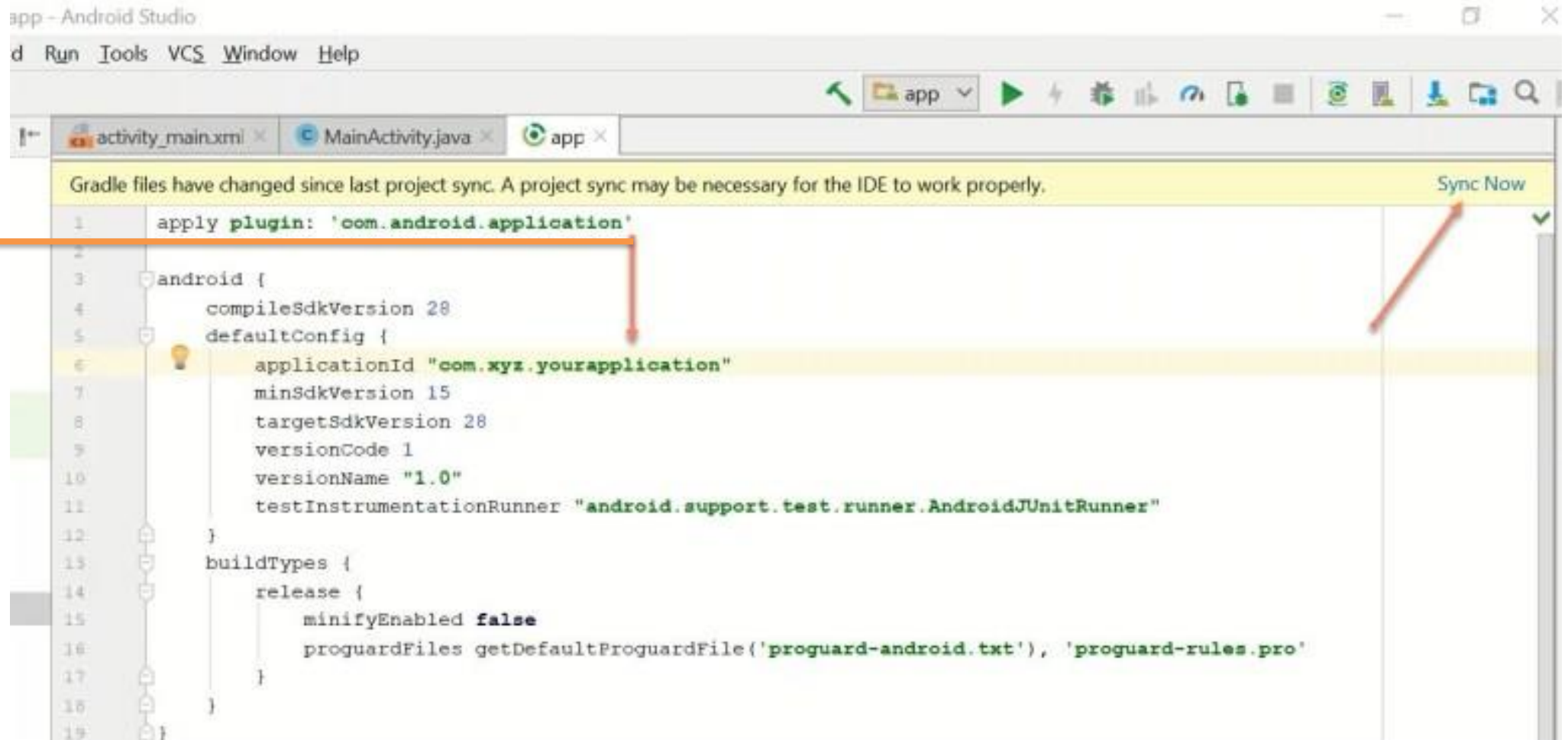


•Step 9: Click on **Do Refactor**.

MyApplication [F:\AndroidStudioProjects\MyApplication] - app -  
File Edit View Navigate Code Analyze Refactor Build R



Step 10: After changing the package name,  
open **build.gradle(module)** and change **Application Id**



change

applicationid **"com.androidride.myapplication"** to **"com.xyz.yourapplication"**.

-After changing applicationid, click on **sync now**.

Congrats. Package name successfully changed.

## Instruction 3:

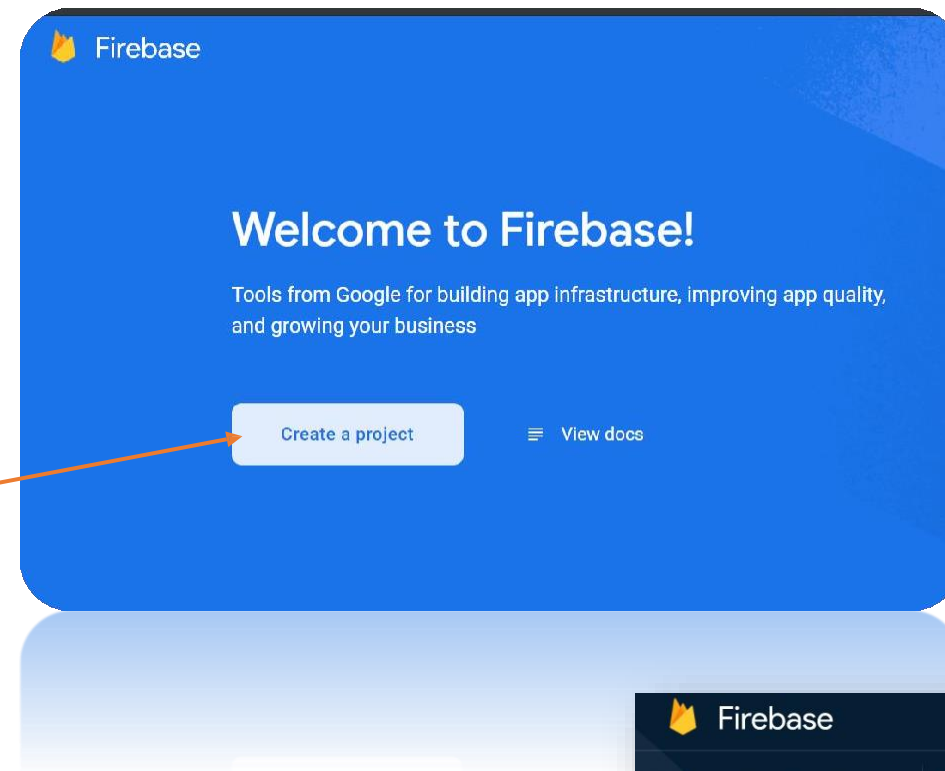
This project is completely built for Private Use. If you want to create Your own Network, Just you have create an account on "<https://console.firebase.google.com/> "and then create a project in It , Then, setup your android app inside that project ...For setup:

- ⇒ After Opening the project and Click on add app Android icon option then give your package name and download the config file from firebase . Following firebase instructions delete our existing config file from the project and in that position paste your downloaded config file. That's it. You can find the location of our config file from the firebase's instruction . After Clicking Continue to console your network is Created Successfully....
- ⇒ For Push Notifications System go to Firebase's Project overview->Project Settings
  - >Cloud Messaging->copy the server key token and then go to our project file->java
  - >package name file->Fragments->APIService.java file
  - >"Authorization : key = 'paste the token here'" ...Your push Notification Service is Build Completely...That's it. Network is Setup with Push Notifications

- Samples

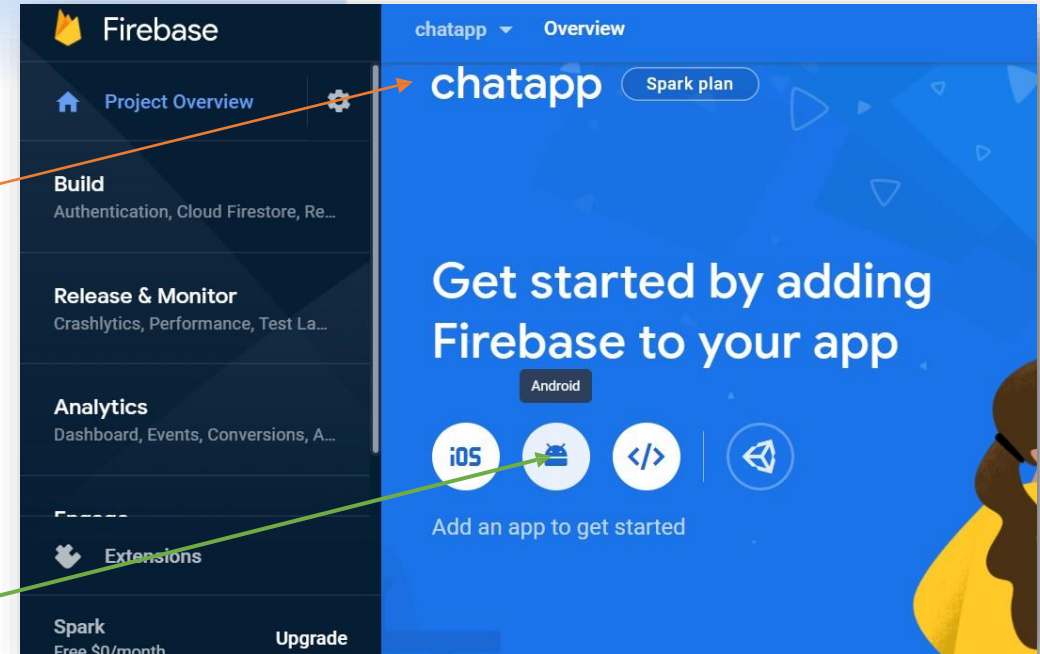
You can login to  
firebase account  
with your email id  
Then,  
Create a Project here

Firebase is  
Completely Free and  
Fast Server



We create a  
project named  
Chatapp

Click Here to Setup Your  
Network



[illegible]

Enter your app's Package Name  
here

## Download the config file And Follow Their Instructions

If You Don't Understand Their Instructions  
Follow ours on nxt slide

✓

Register app

Android package name: com.chatapp.sss

2

Download config file

Instructions for Android Studio below | [Unity](#) [C++](#)

Download google-services.json

Switch to the **Project** view in Android Studio to see your project root directory.

Move the google-services.json file you just downloaded into your Android app module root directory.

google-services.json

Project

MyApplication (~/.Desktop/MyApplication)

- .gradle
- .idea
- app
  - build
  - libs
  - src
  - .gitignore
  - app.iml
  - build.gradle
  - google-services.json
  - proguard-rules.pro
- gradle

Previous

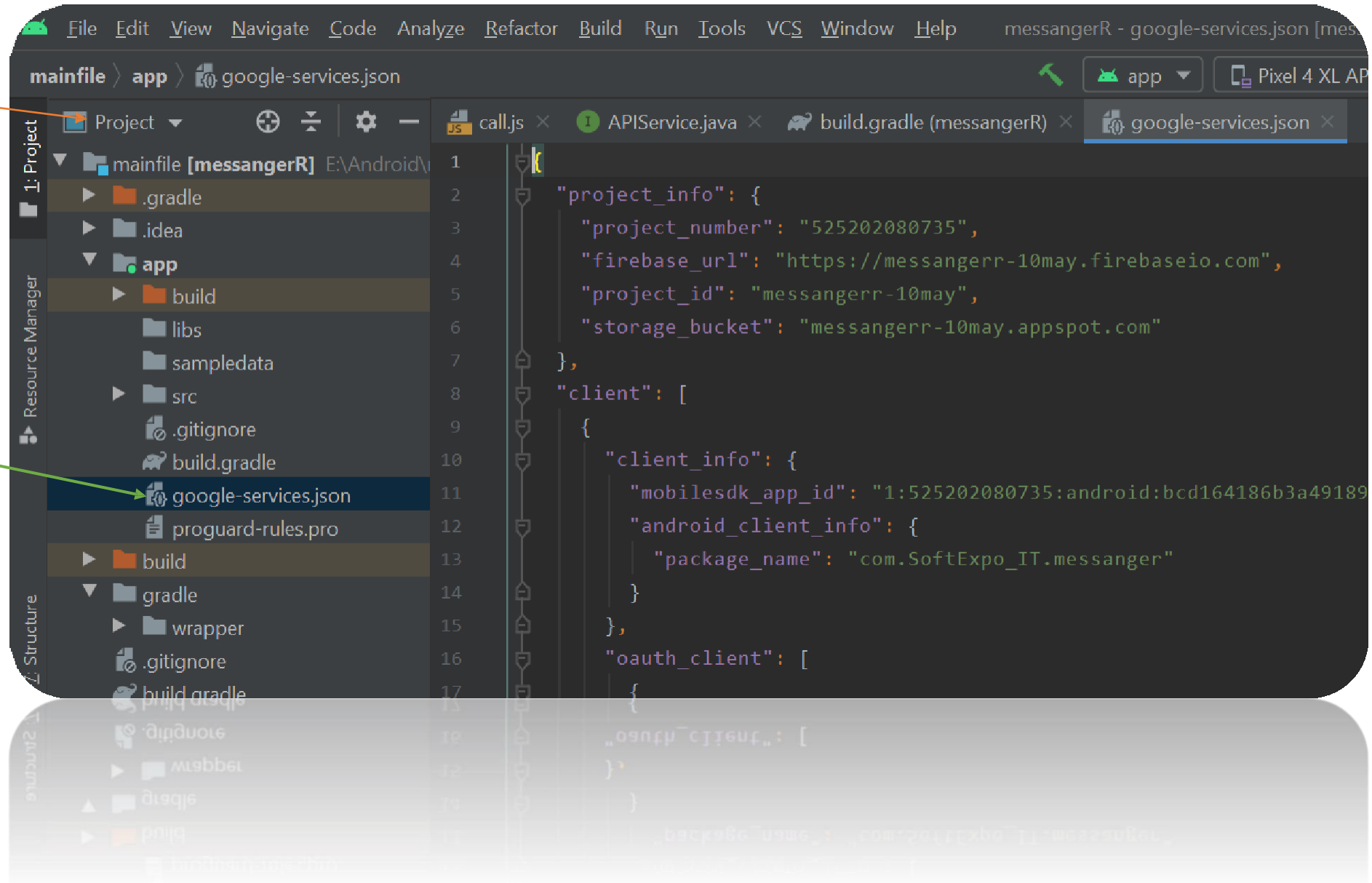
Next

Change android to  
Project Here

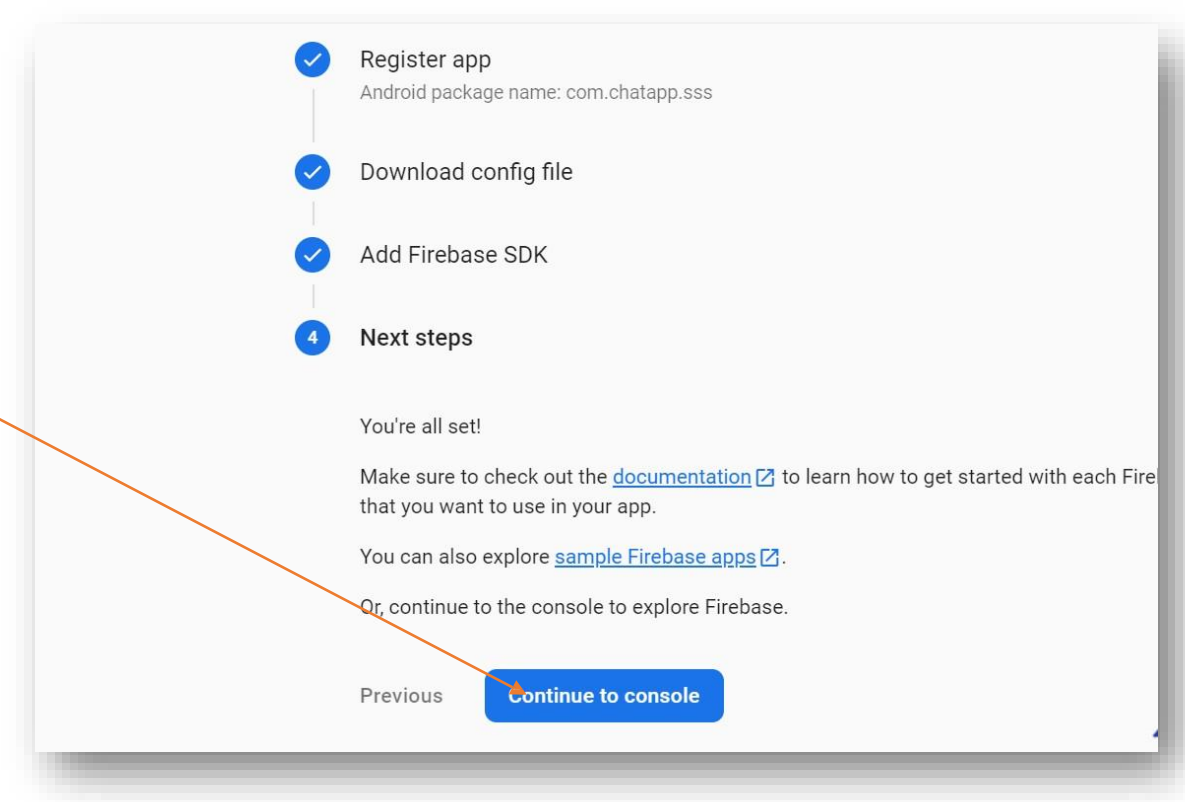
Delete our google-  
services.json file from  
here

And paste your new  
Copied services.json file  
here  
At the same position

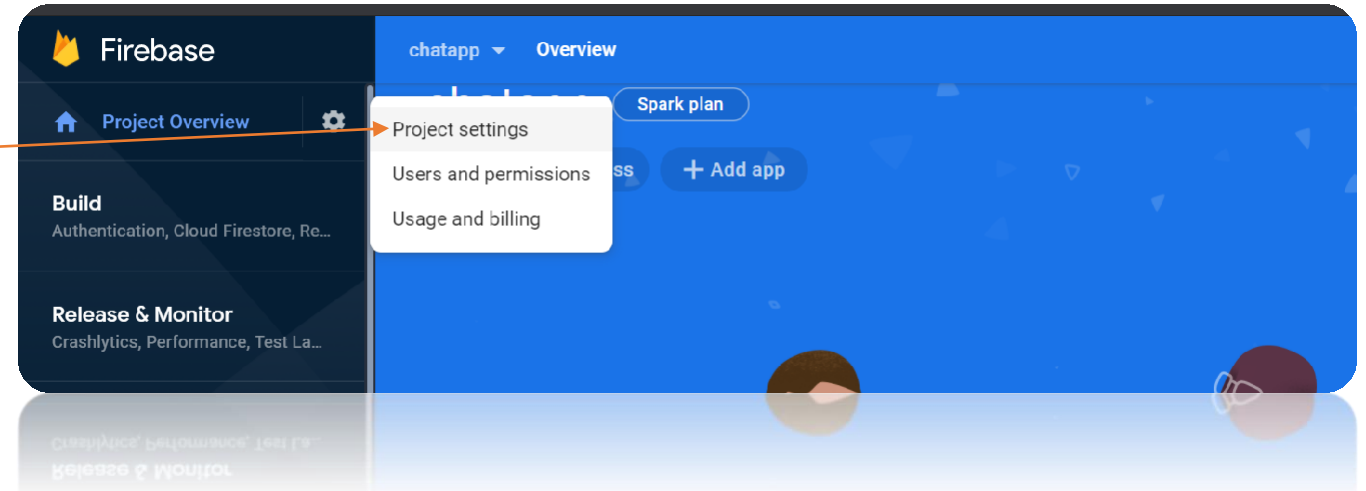
Be Careful  
It is most Important part



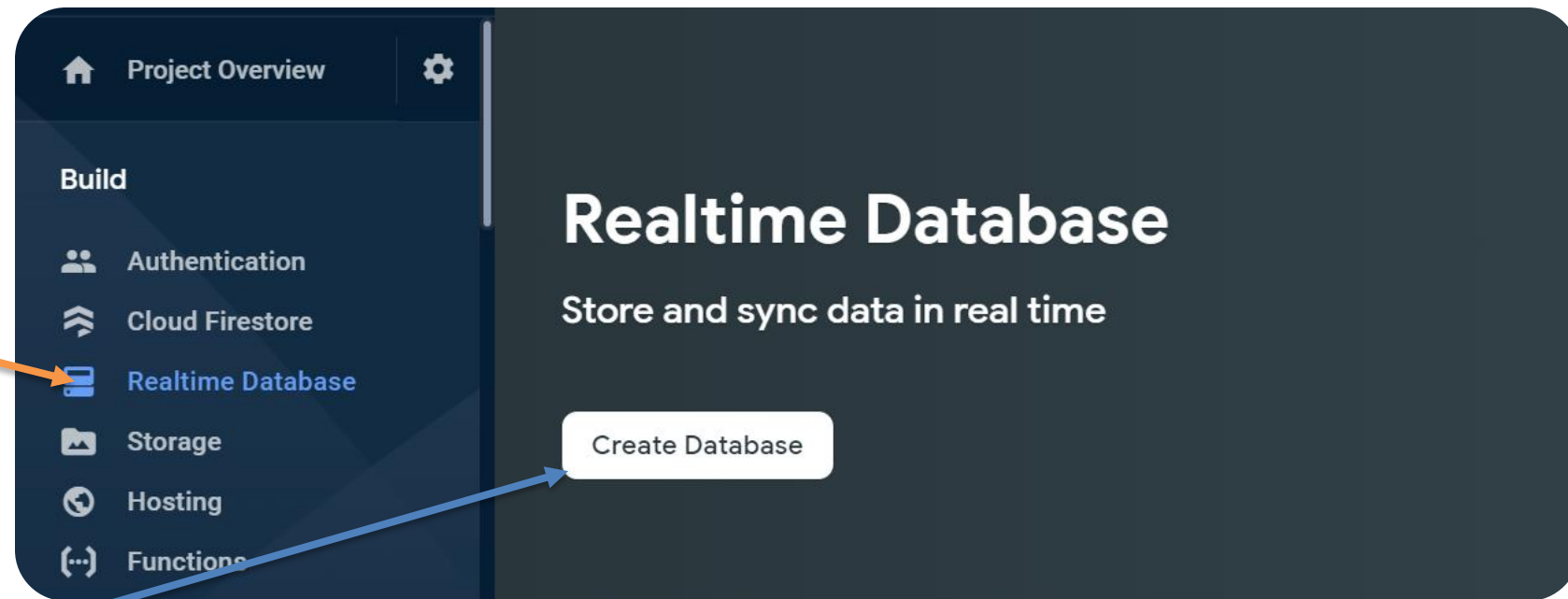
Then Click Continue the Console  
Your Network Setup Finished completely



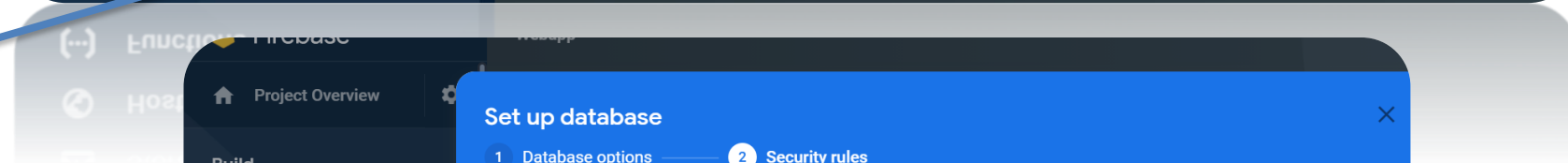
For Push notifications go to  
Project Settings from Project Overview



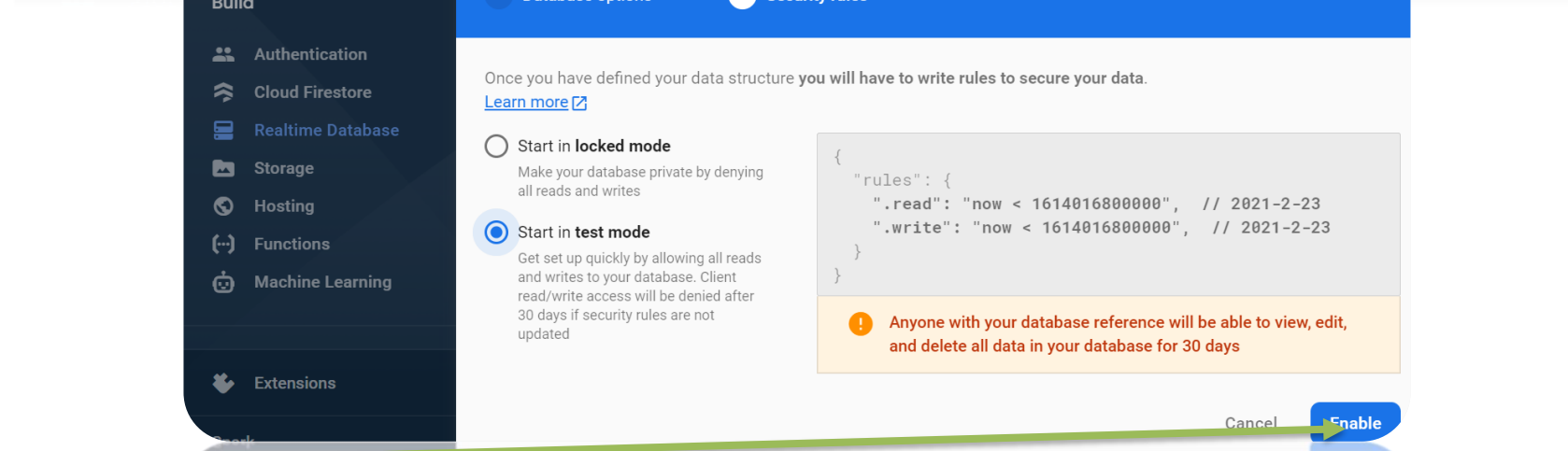
Then You have to create a database just click here



Then Click on Create Database



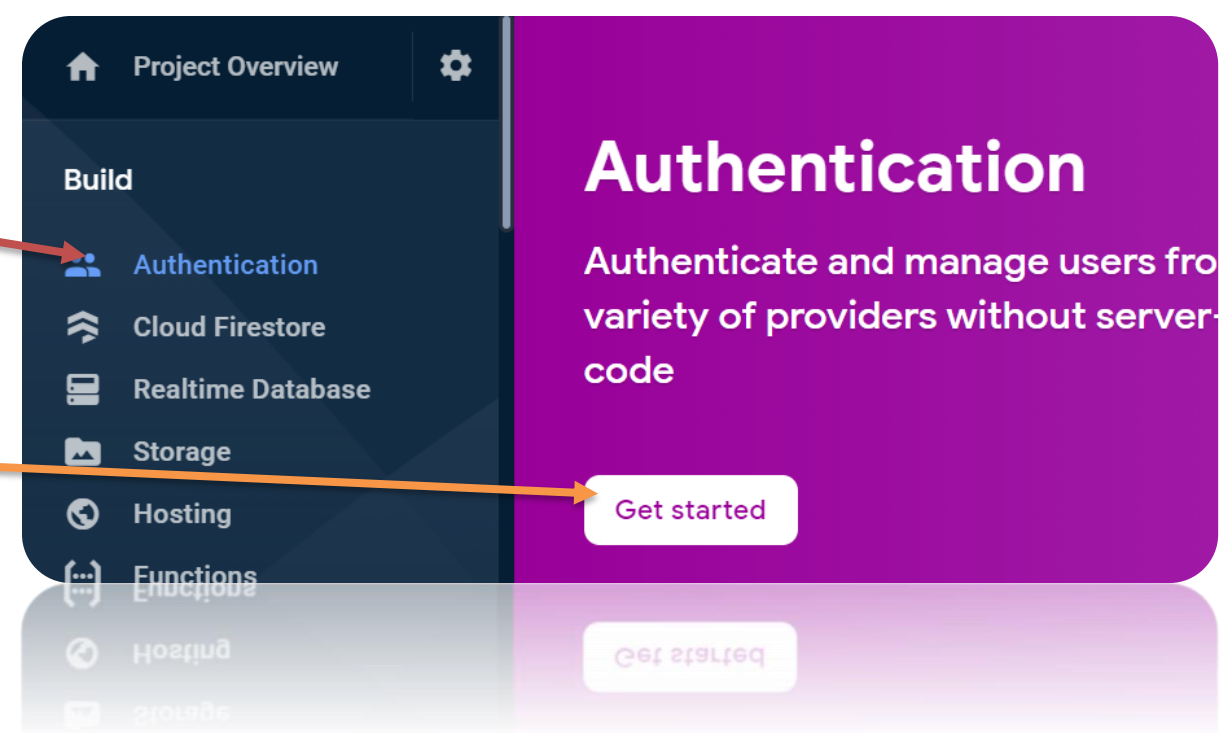
Then Enable it  
Your Database will be  
created automatically



...

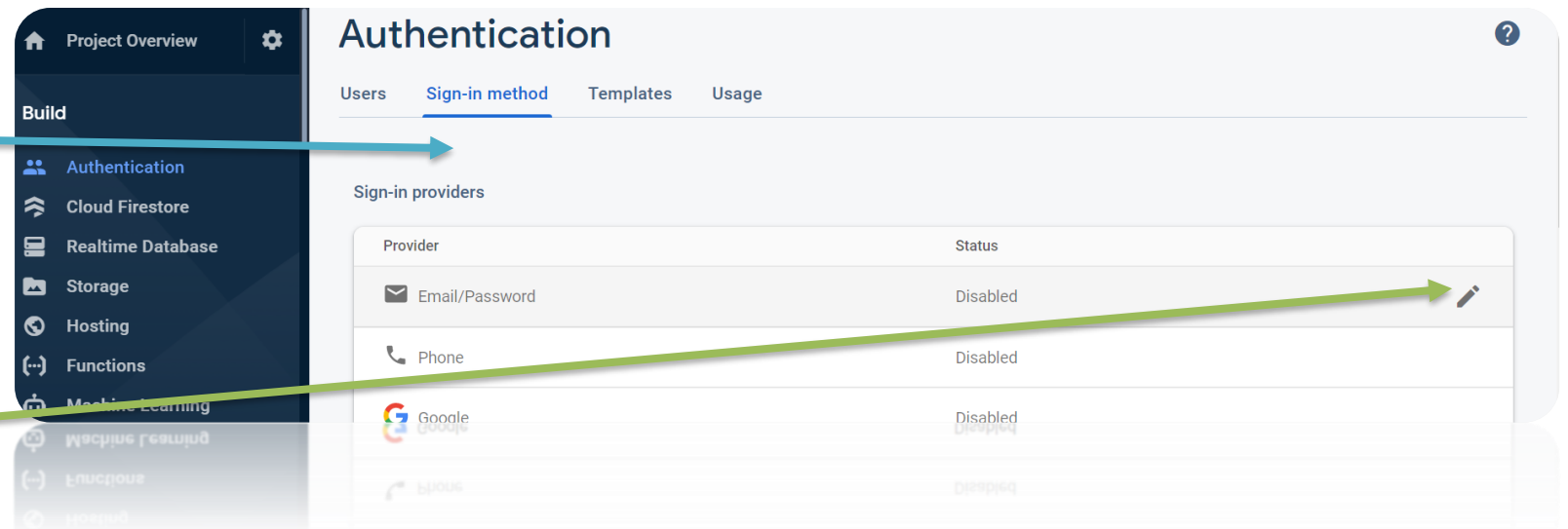
Then Click to Authentication

Click on get started



You can See a page like this

Click here

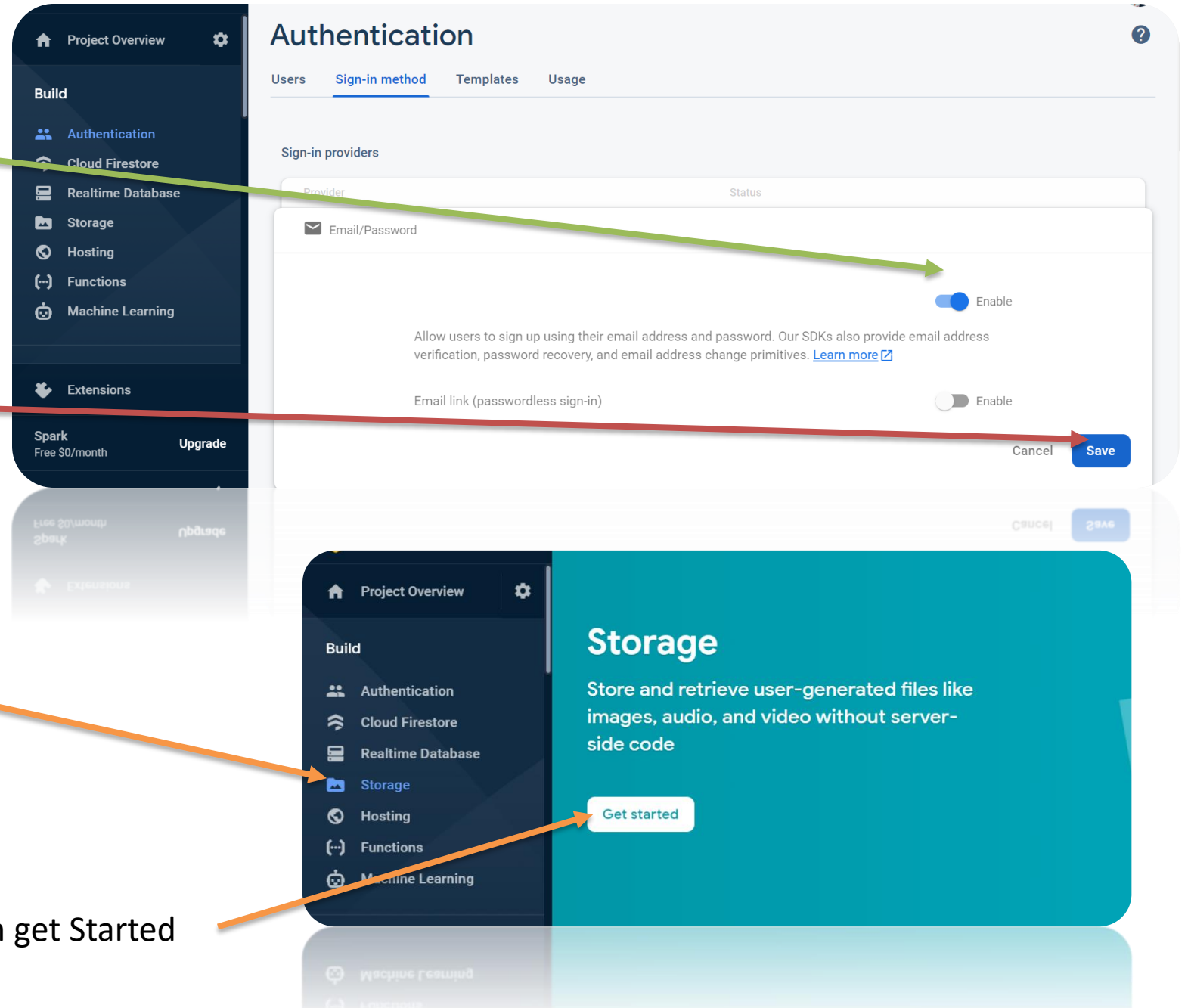


Enable Email/password

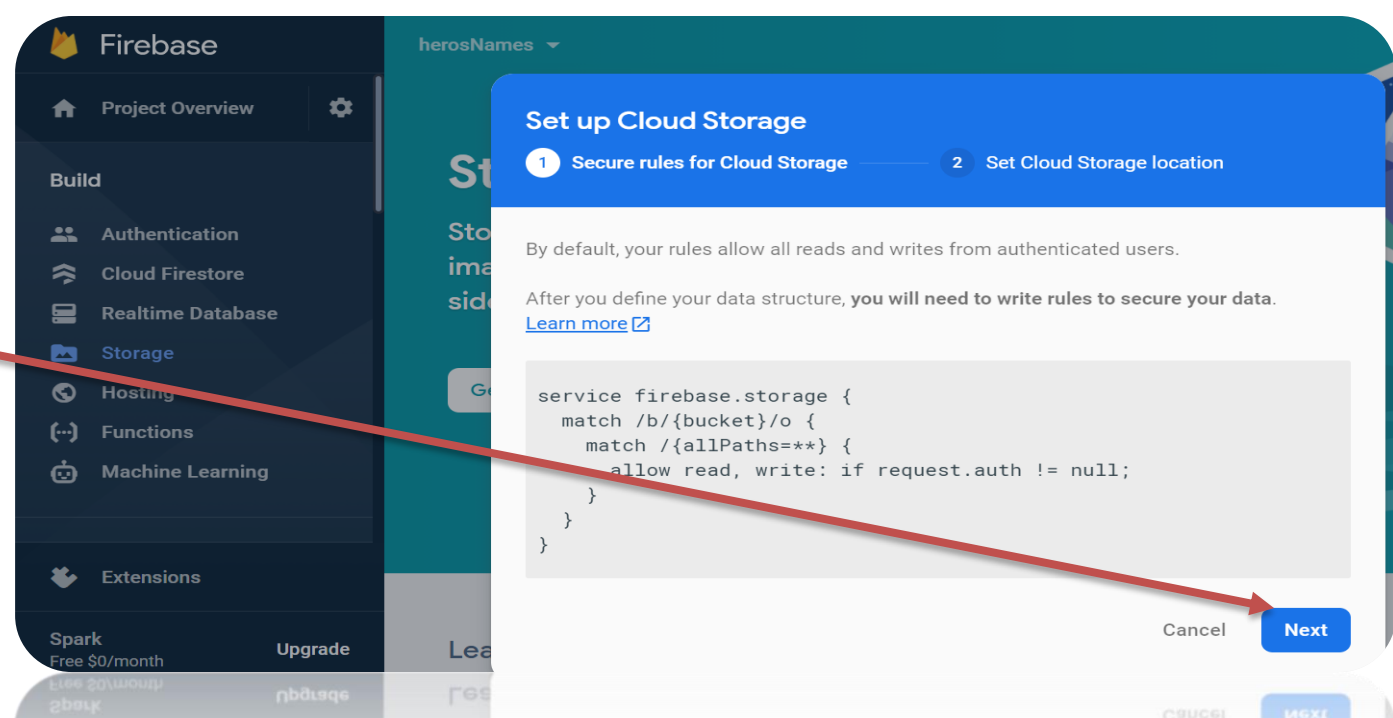
Click on save

Click On Storage

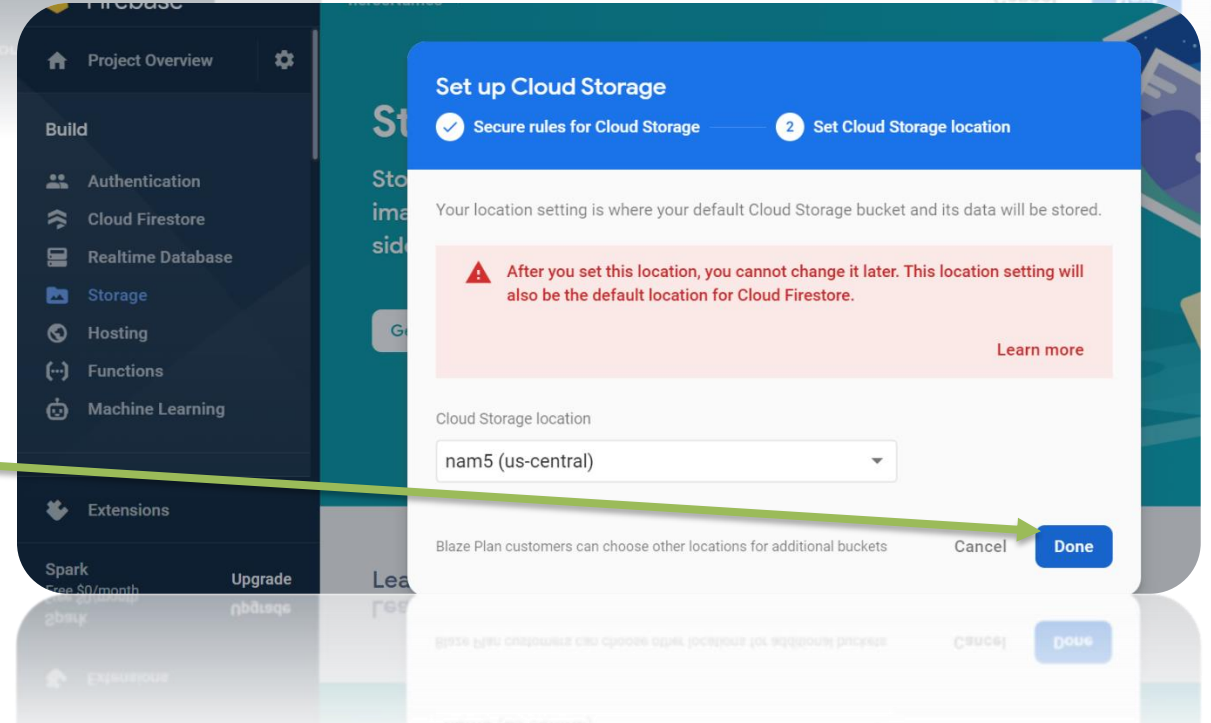
Then Click on get Started



Then click on Next



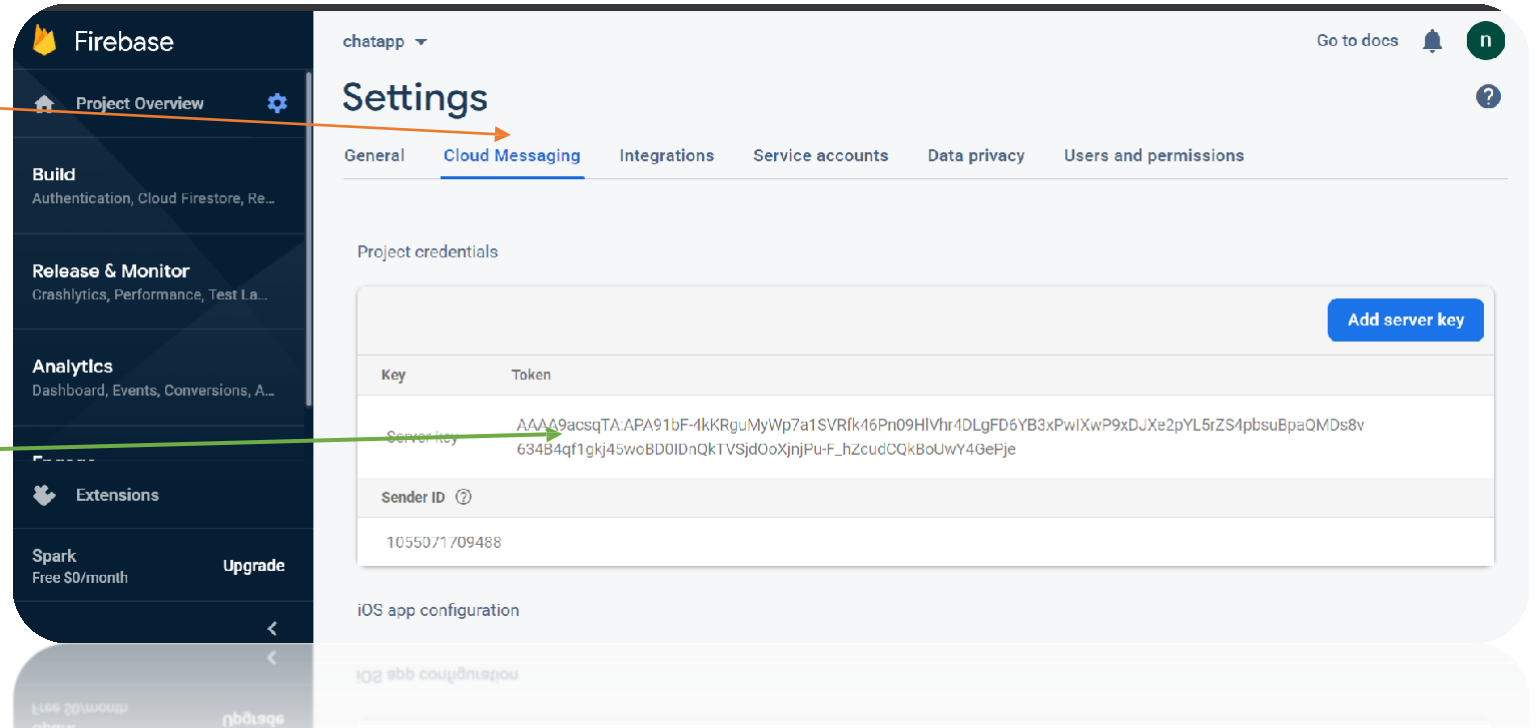
Click on Done



Your Private Network Creation  
is fully completed

Click on Cloud Messaging

Copy Server Key Token

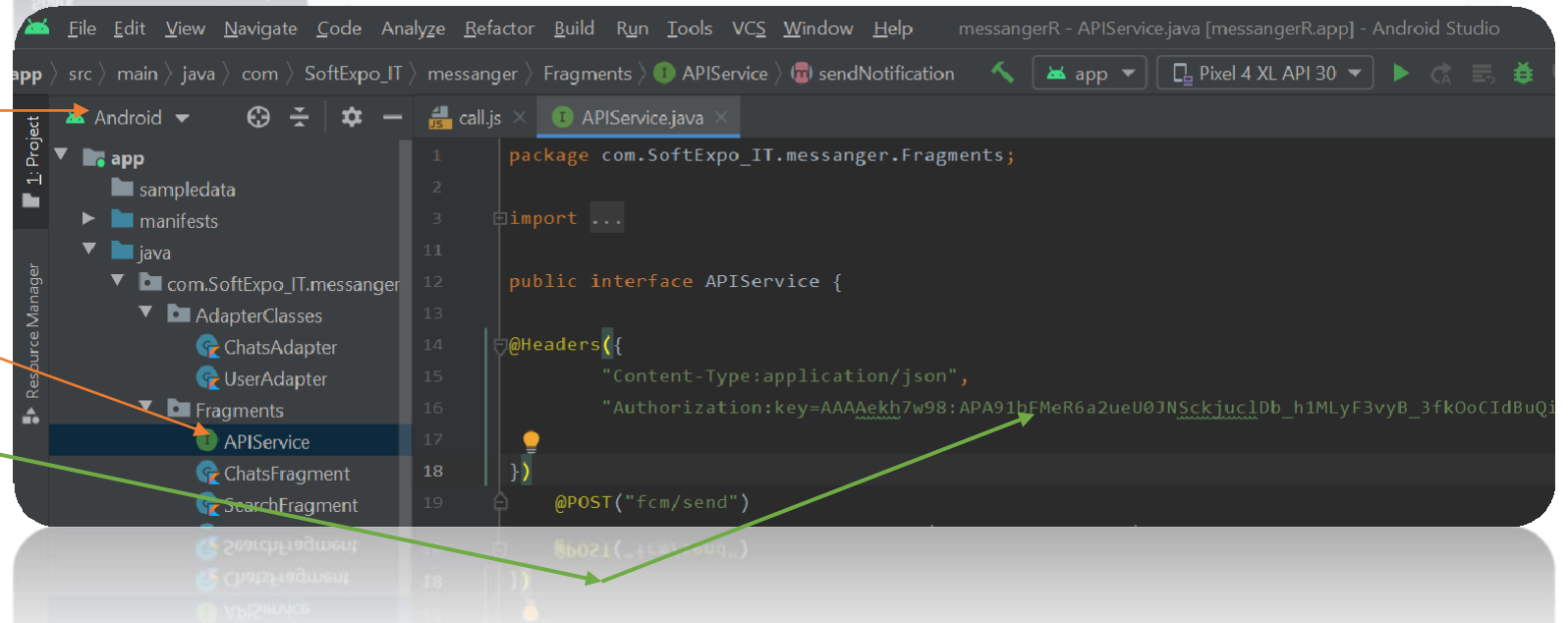


Select Android here

Go to ApiService class

Then delete the existing key value  
And paste your newly copied  
Token here

Done. Push Notification is setup 100%



- **Instruction 4:**

- You can add more turn/stun servers if you want...We use turn servers for World-Wide Video calling.
- Go to Android->app->assets->call.js->init function->config->you can add more turn/stun servers according to your location...
- if you don't want to add any server additionally , skip Instruction 4...Use our Turn/Stun Servers...
- our turn/stun servers are international and can handle your video calling Network Properly

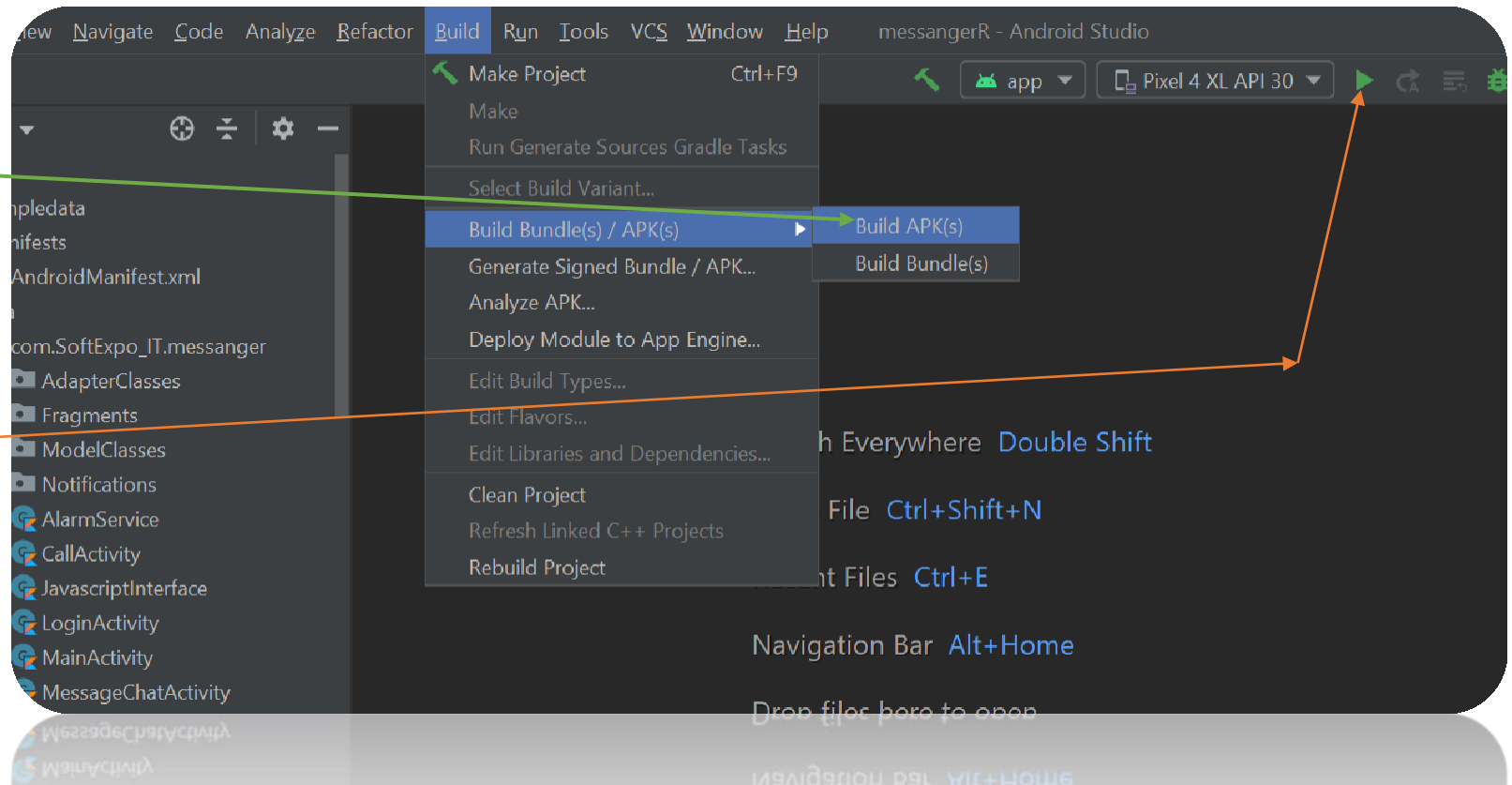


# Instruction 5

- Build your Project to get your app's APK file...The process is :  
Build->Build Bundle(s)/APK(s)->Build APK(s)

Build APK File

Before Build You can run  
your file in emulator with  
this



# Instruction 6

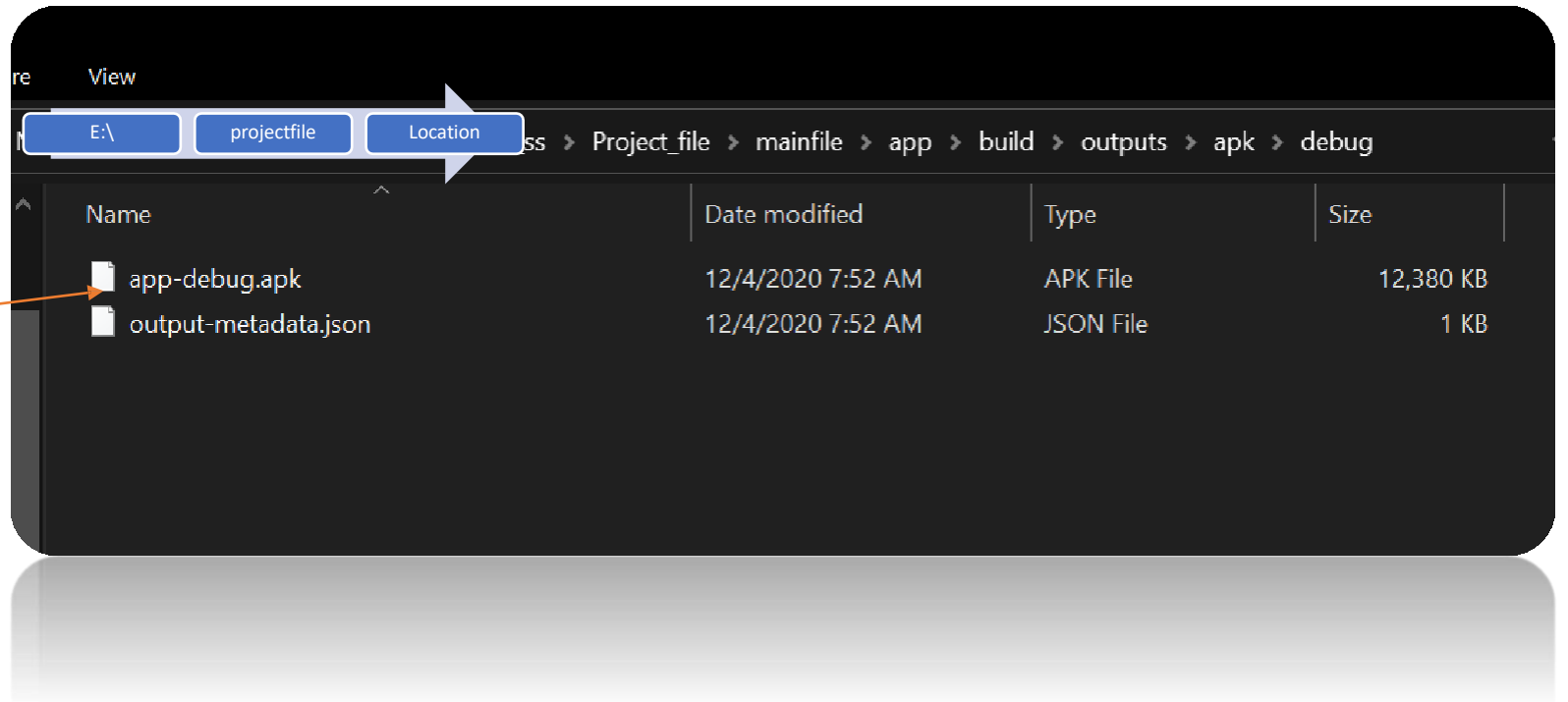
- Deploy The APK File in the play store/Any where you want.
- You can also Distribute the APK File with any Link

That's it . These are the only steps you need to Follow . Follow these steps to create your own network...If you want to use Our network, just build the project You can also get the apk file from `Projectfile_Location\mainfile\app\build\outputs\apk\debug`

Our APK APP

So if Anyone don't want to build  
APKs

They will Directly use our Released  
APK File from Here



By Mahir (CEO,SoftExploit Limited)  
Created on 24 January 2021  
Email: [softexploitlimited@gmail.com](mailto:softexploitlimited@gmail.com)