-Tools:

Android studio

Fire base

Retrofit

-Splash activity

First we start with splash activity the purpose of it is to check if the phone connected to the internet or not with beautiful design and logo this the entrance of our app.

It has only one button called start.

The main purpose of this screen is to check the phone is connected to a network or not so:

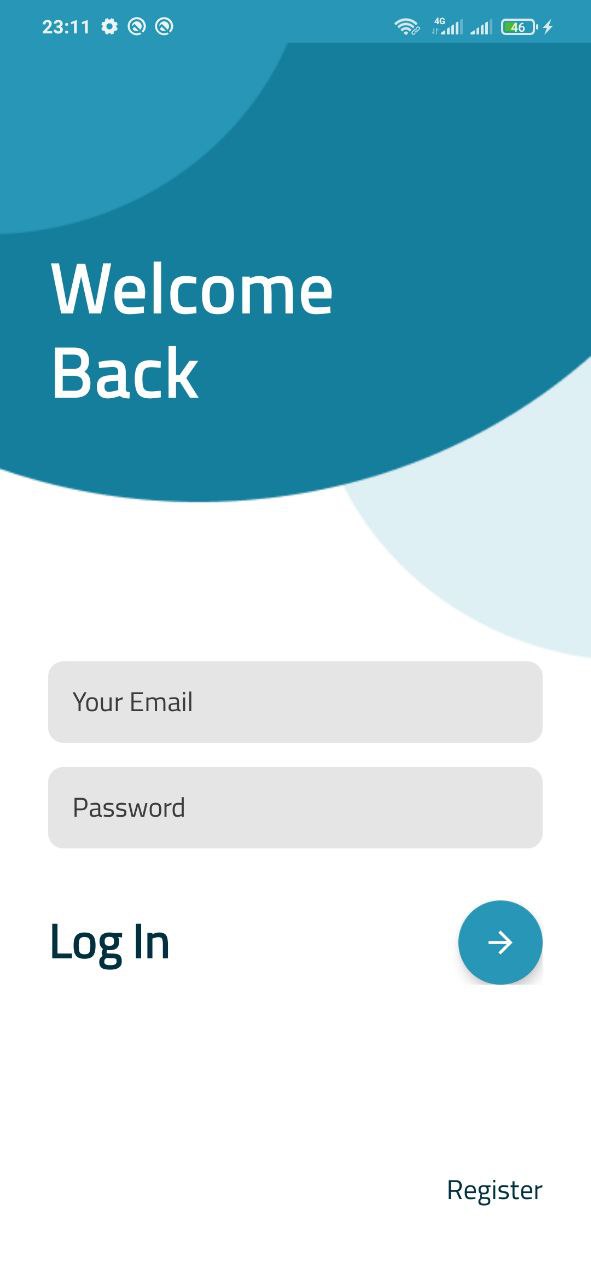
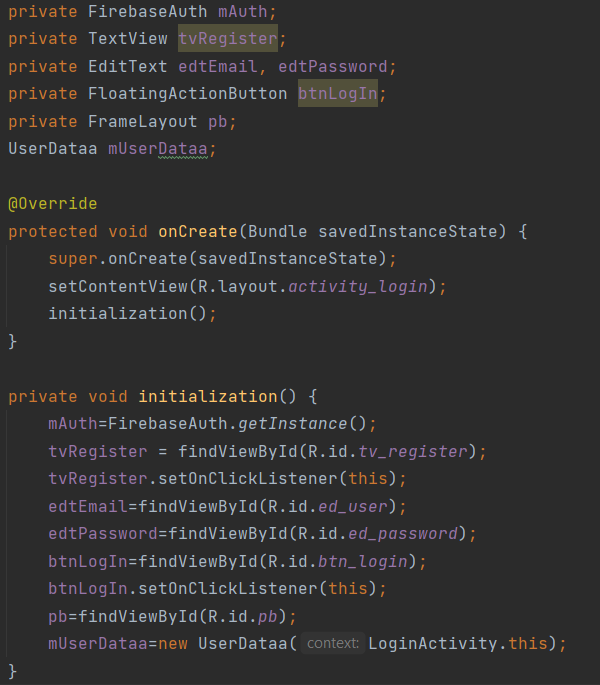
1. We intiate the views
2. Check the network by using isNetworkConnected function
3. If connected navigate to cam activity
4. If not print toast

Camera Activity دي ناصر هيشرحها

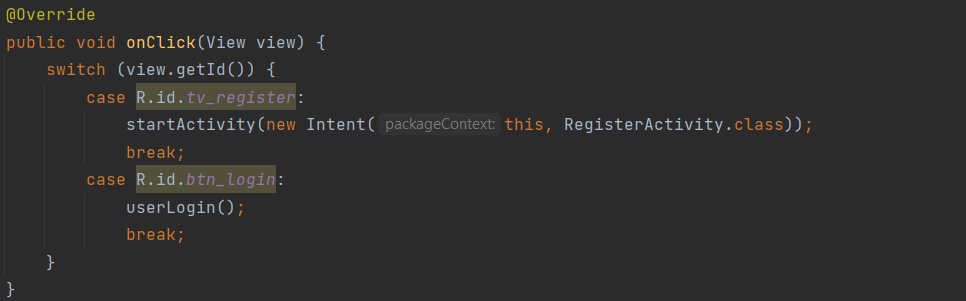
Authentication:

-log in activity

We use firebase authentication in our app for manage users and registration log in activity has only two edit texts for email and password, log in button and register button for new users after check if users wear masks or not the app automatic navigate to log in. user has two options:

1. Already a user so he will just log in with his email and password
2. New user so he will register and enter his data.

The action occurred when the floating button is pressed so we write the code in the button.

After initializing views in log in activity we can access, control and handle it like getting, saving data and validate it and so on…

This is on click method we handle all click actins within log in activity on it we have the most important function in it which is userLogin(); this function doing all process to log in the user.

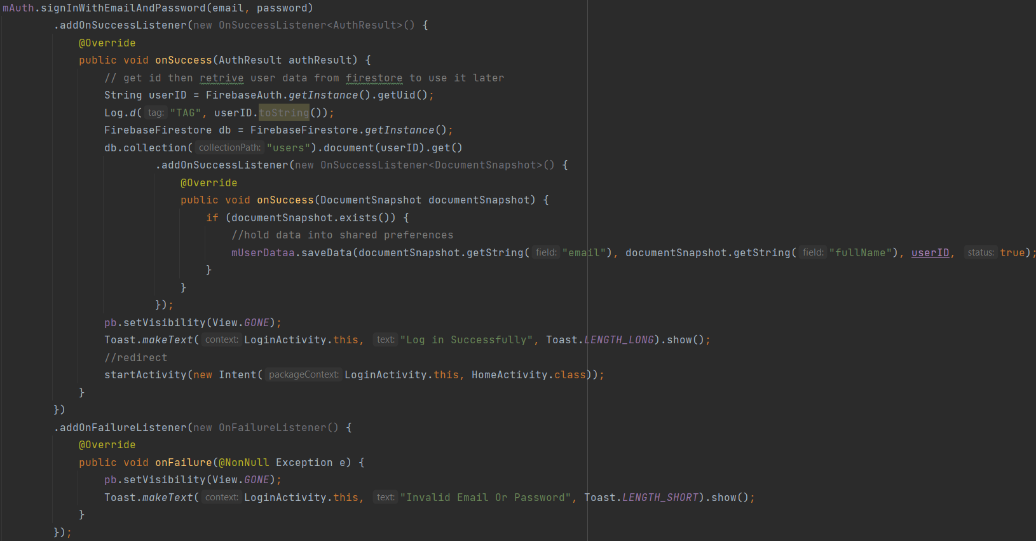


First we get the email and password from EditTexts by using getText() function, convert them into string and finally delete any spaces after or before the text by using trim() function.

Second we have some checks on the email and password that user enters it:

* is the email empty?
* is the user entre invalid email?
* Is the password empty?
* Is the length of the password less than 6 characters?

After all that checks we have an valid email, password next and final check is the email registerd , user entre the correct password and firebase will check that.

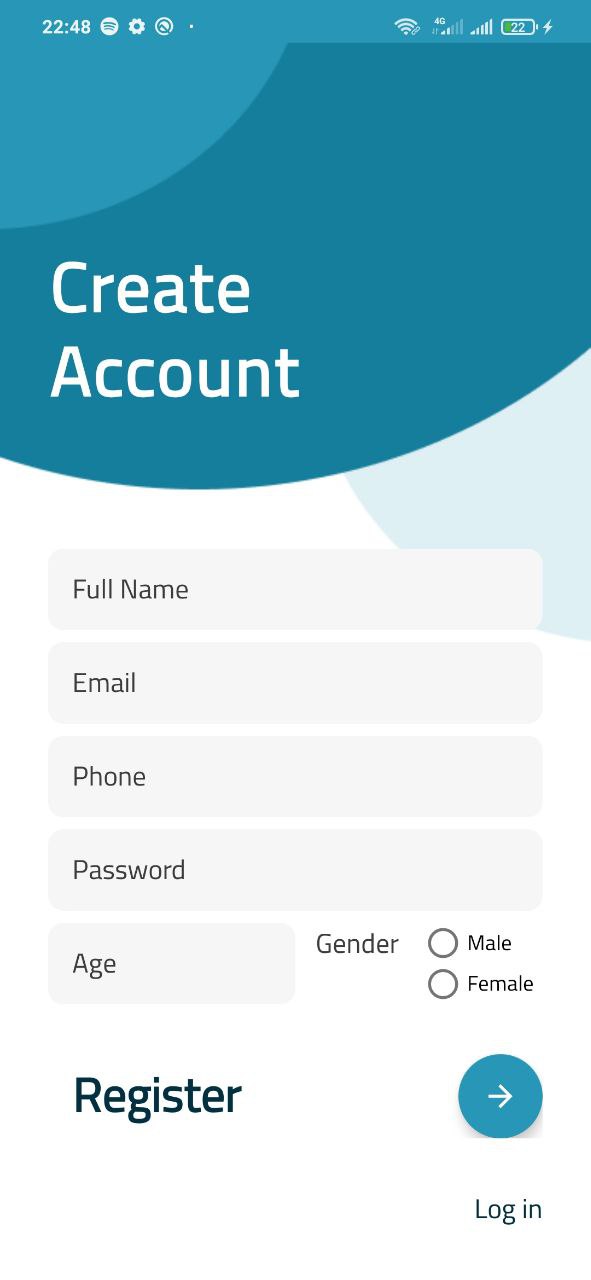
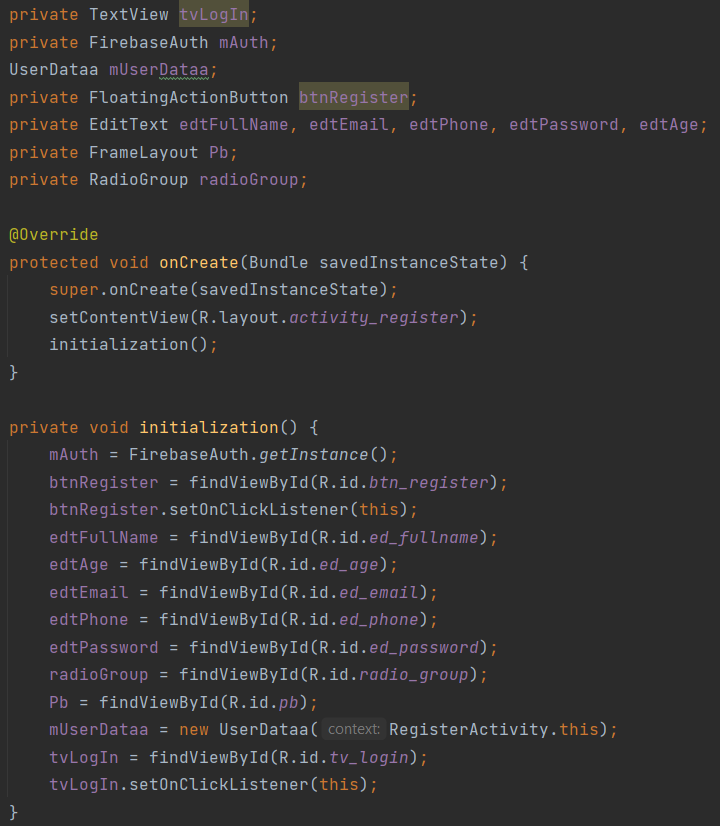
we use firebase signInWithEmailAndPassword function and pass email, password to it, then it connects with firebase authentication to check the data, if the user exists and the password is correct we can log in but first we need to store some info about user so we access fire store and get some data that can be used later and store it locally in shared preferences.

* First check email and password if invalid print toast with the error
* If valid
  1. Get user id
  2. Access fire store data
  3. Get the data of the user and store it in shared preferences.
  4. Data like email, full name and id
* If not valid we get the error code from firebase by using exception

After all that steps we log in and navigate to home screen.

If the new user need to register he will just go to register activity by pressing register button it will navigate to register screen.

Register Activity

The new user can easily create a new account by entre his data like his name, email, phone, password, age and finally select his gender.

The screen consists of some edit texts, floating button and log in button that navigate back to login screen.

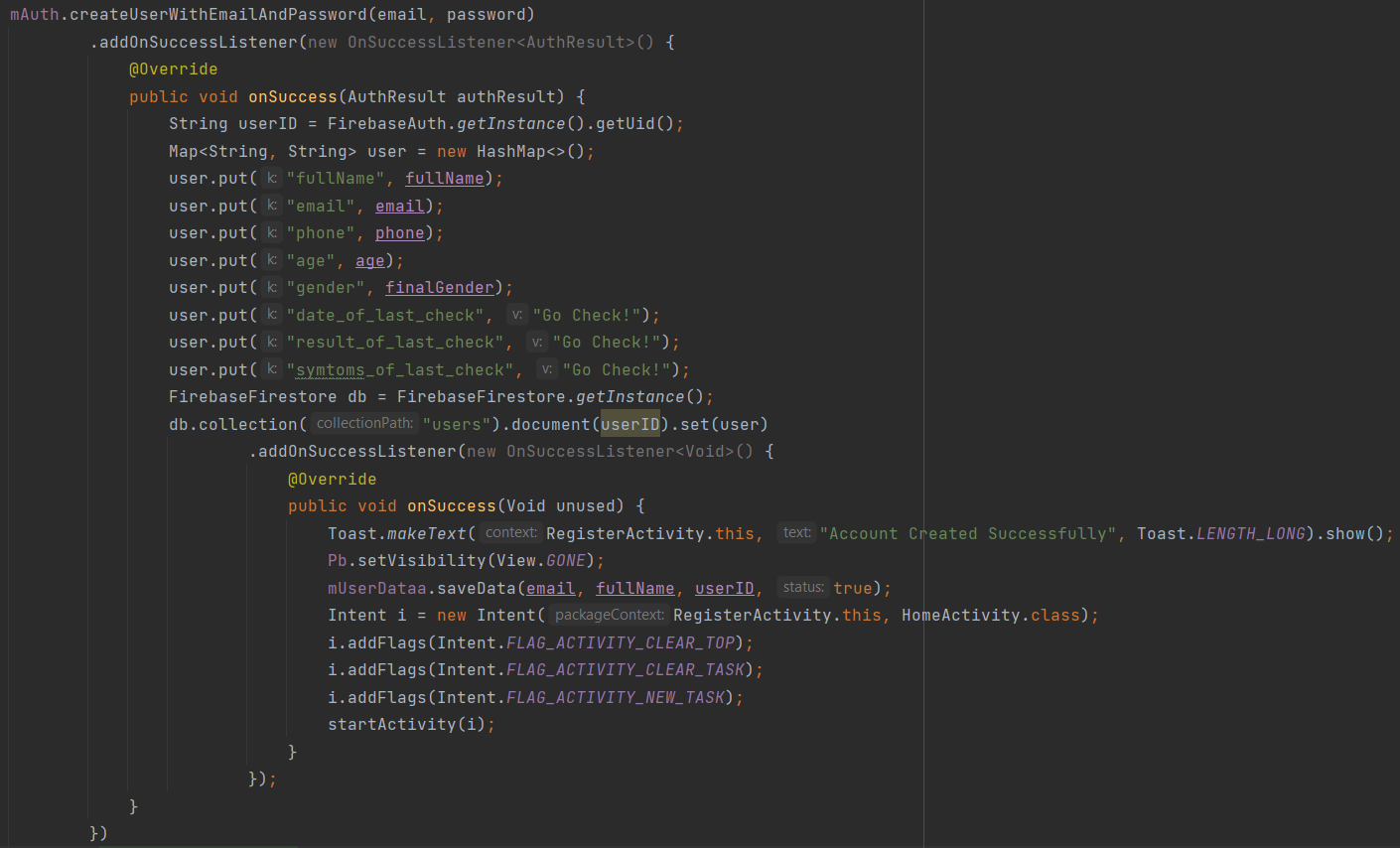
Like the login activity we use firebase authentication and firestore to save the user data

We first start with intiate views in initialization() function and then we validate inputs like we did before.

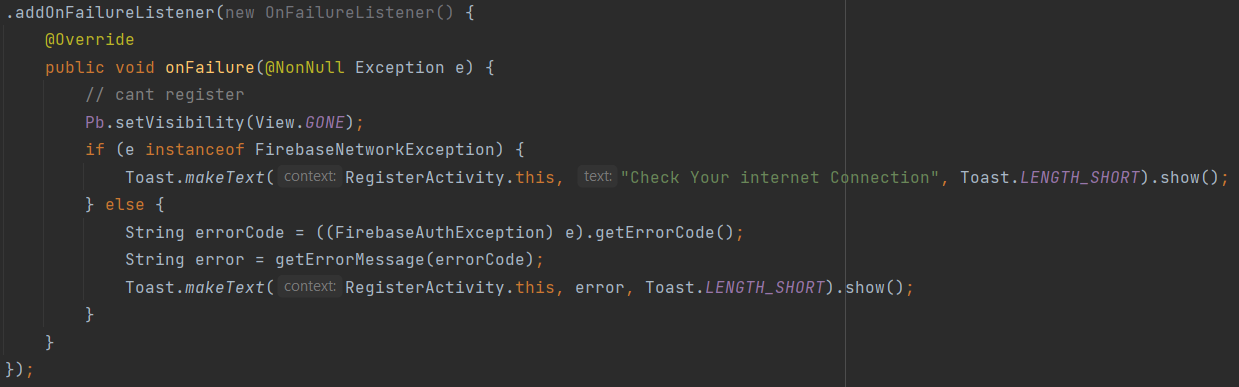
Validation for:

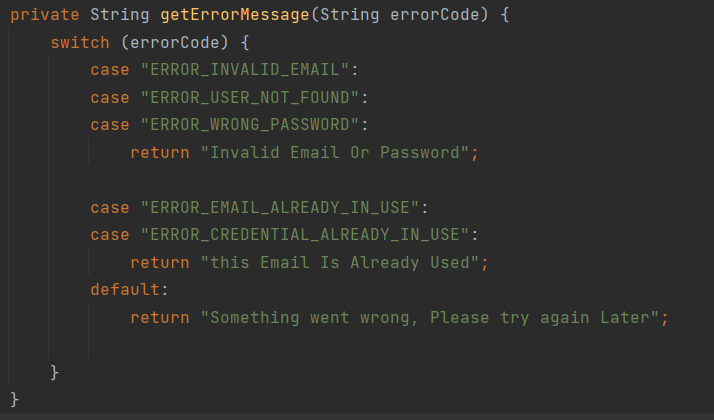
1. Name if the user didn’t entre anything
2. Email if user entre invalid email
3. Phone if user entre invalid phone
4. Password if user entre short password
5. Age if user entre invalid age
6. Gender if user didn’t choose anything



After validating all data we create user using firebase createUserWithEmailAndPassword() function and it takes only the email and password so..

After creating account we save all data into firestore each user has a unique document with his id hold his data.

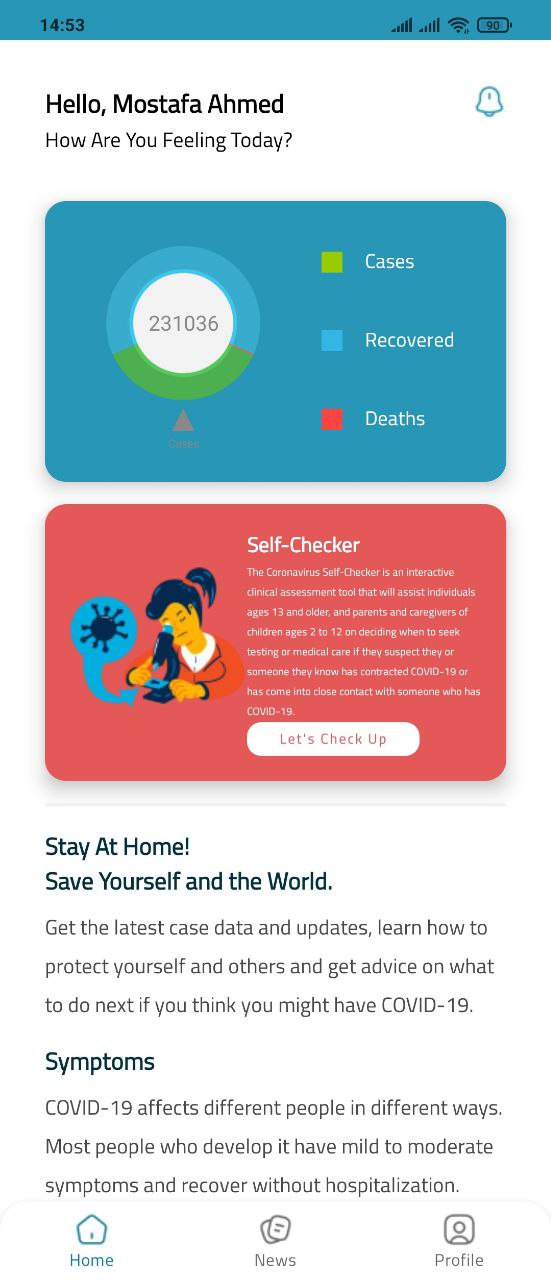
If an error occurred like no connection or error in the data base we received it in onFailure method

We have many cases so we handle it in getErrorMessage method

After register user we navigate into Home screen that holds 3 fragments and bottom navigation view..

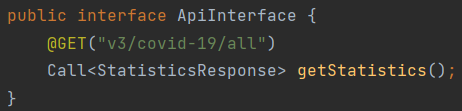
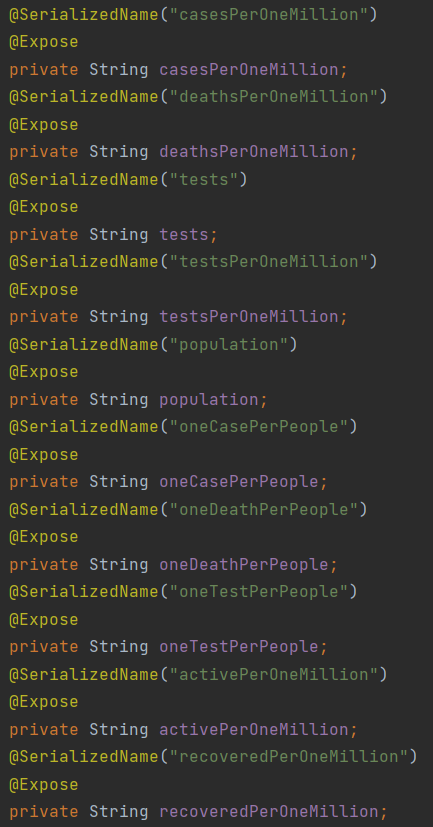
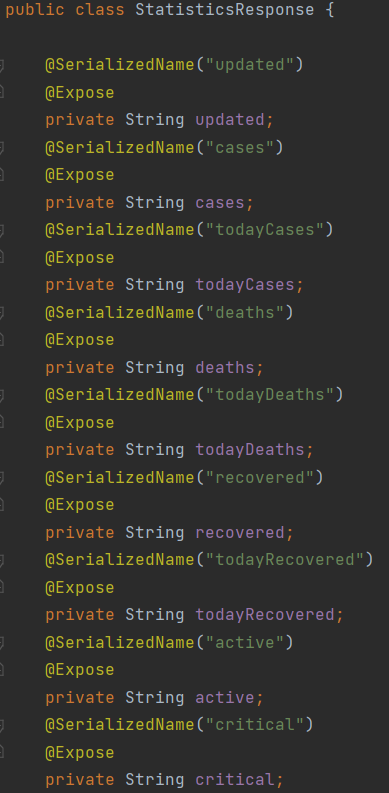
Home Activity

Home screen has multi options

* first we have statistics about corona cases around the world updated every day
* self-checker we can access it by clicking on let’s check-up.
* News button
* Profile Button
* Some advices

The default container in this activity is home fragment that show the statistics.

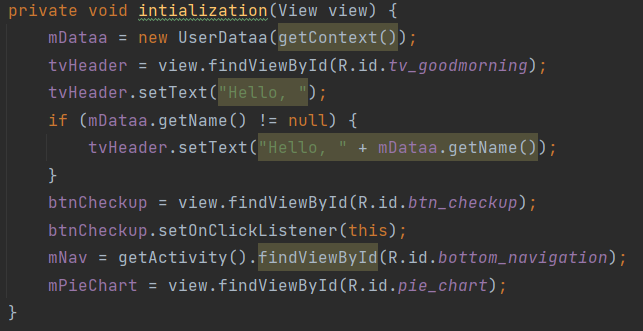
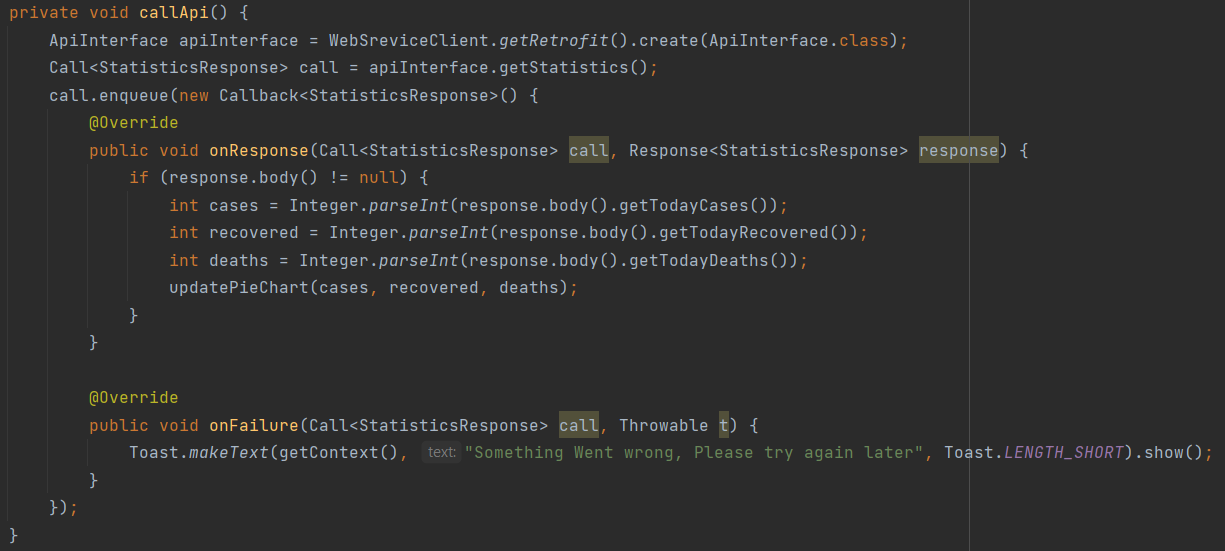
We get statistics from an public api that cover cases and updated every day. We use retrofit library to get this data. We received data as a query object and translate it into java object.

* First we have 3 classes that helps to get data from the api
  1. An interface ApiInterface that have the get function.
  2. StatisticsResponse class that form the data in the api. 
  3. WebSreviceClient class that make the call. 

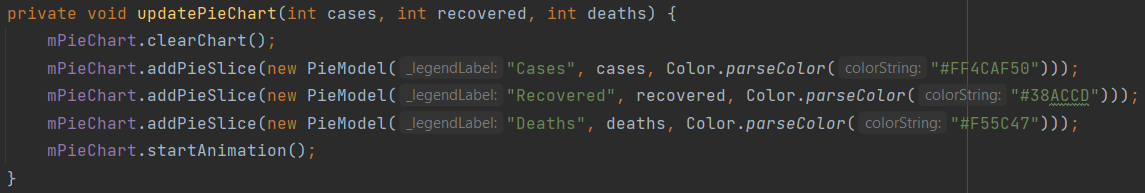
After intiating these helper classes we can easily control the data in the entire app

The home fragment holds the statistics pie chart that show 3 values

1. cases
2. recovered
3. Deaths

After intiating views in home fragment we called a method for make the connection and retrieve the data from api.

After calling callApi() function first we make instance of the interface and we call the function that get all statistics and after get the data we parse it into integers and pass it into updatePieChart function it draw the pie chart in the screen if the connection is bad or the phone not connected a toast messege will be shown.



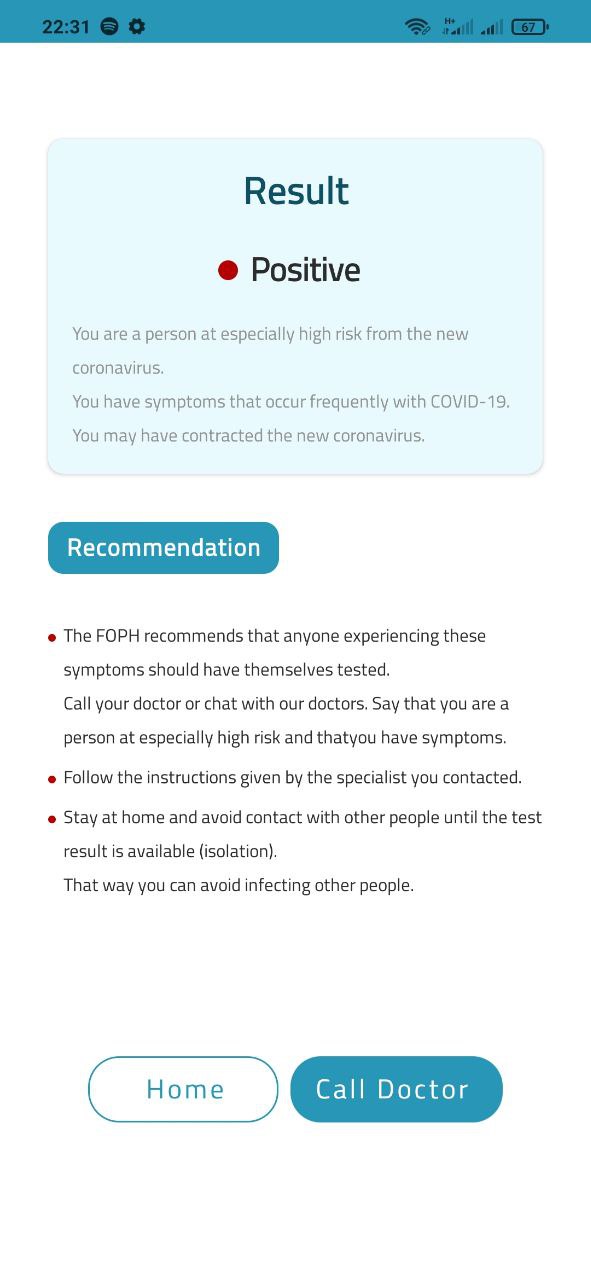
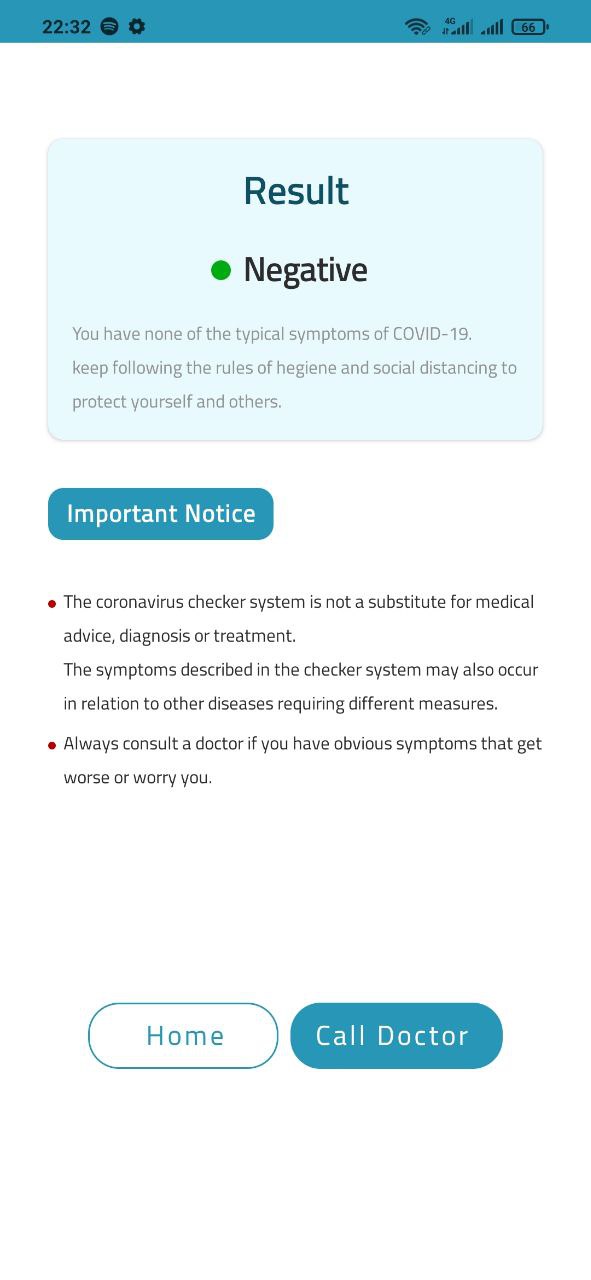
This function is only draw the results that we got from the api.

Next we have checkup button that start a few questions that determined if the user have corona or not..

Questions Screen

Result screen

After the user answer the qustions we have two ways:

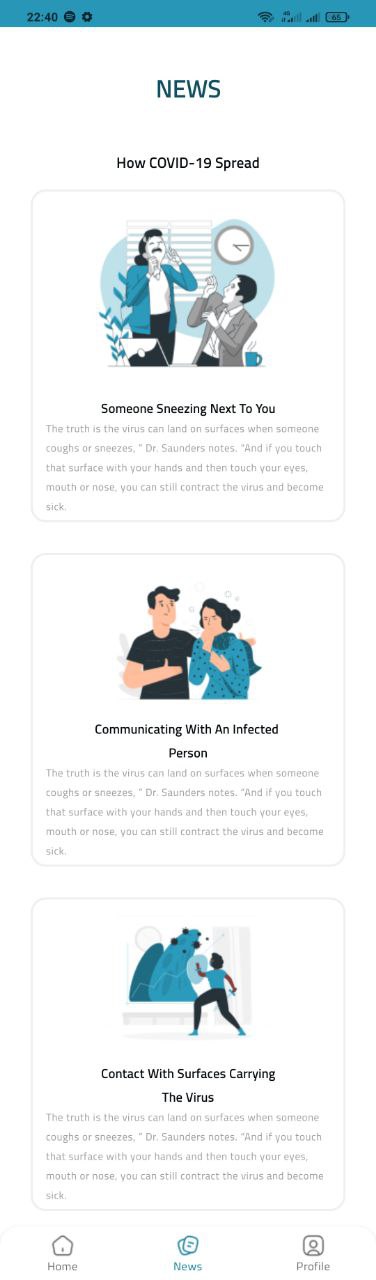
* Positive means that user have the virus
* Negative means that user didn’t have the virus.

After check user have to ways:

1. Back to home screen
2. Call the doctor

Call doctor

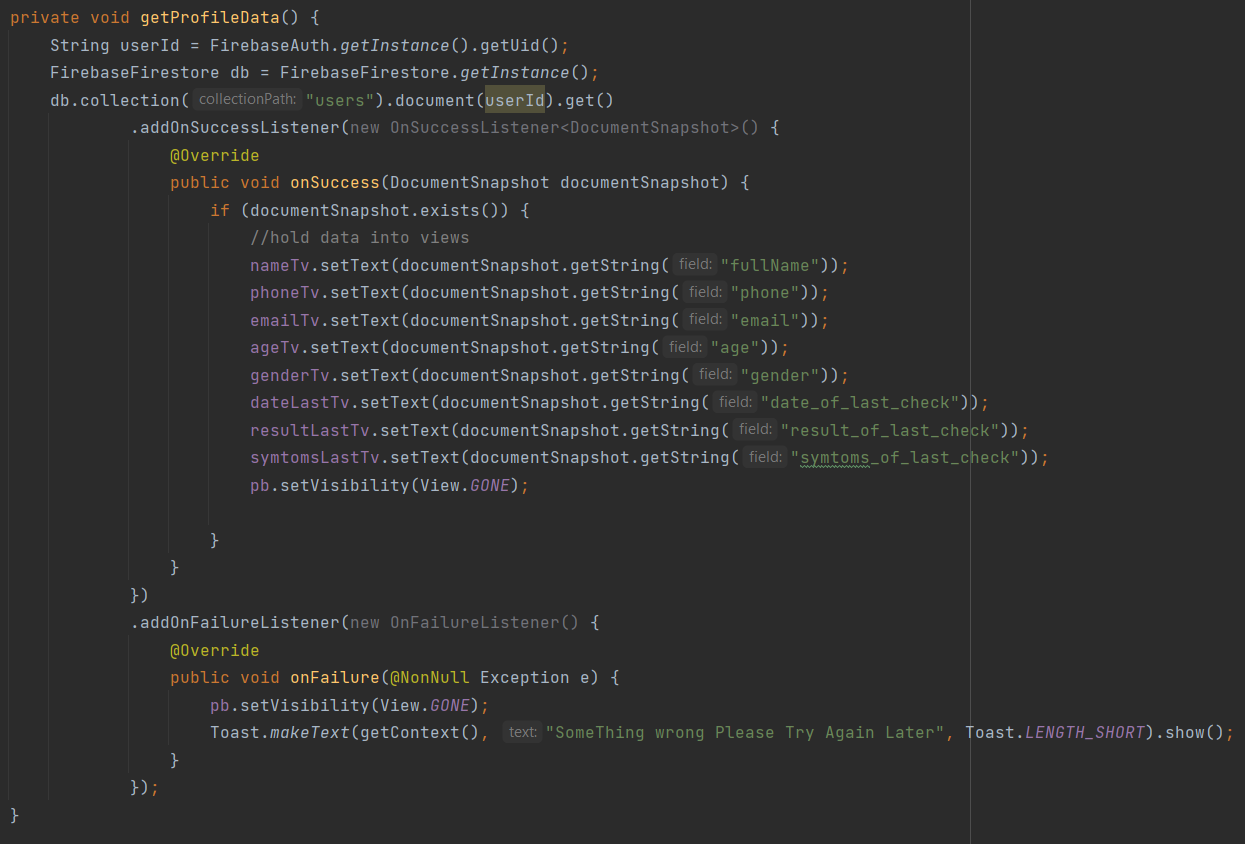
News screen

Nothing great it’s just a design screen designed by el designer el fager nafe3

Profile screen

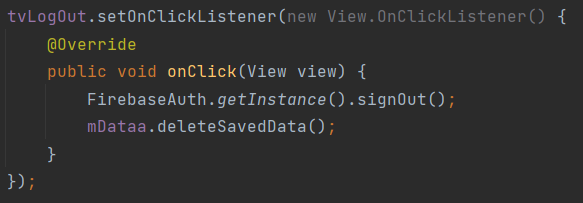
In this screen we show the data that the user entered in sign up screen we grab it from firestore and we grab the result of last check

Screen here

After intiating views we use getProfileData() function to get the data and hold it into views

* First we get the user id from firebase and store it
* Then we access the document of the current user and get all data that stored
* And finally we view it in some view
* If were problem the app will show a toast with the error which mostly bad connection

The log out button

It’s a normal button that log out the user from the app and clear all data saved in shared preferences.

We call fire base signout function and deleteSavedData() function that clear all data saved in shared preferences.