Name : Mujahid Ullah

Subject: Android Development

Submitted To Sir Junaid

Date : 2-Jan-2021

Assignment No 11

Firebase database

Firebase Realtime database is a cloud hosted database that supports multiple platforms Android, iOS and Web. All the data is stored in JSON format and any changes in data, reflects immediately by performing a sync across all the platforms & devices. This allows us to build more flexible realtime apps easily with minimal effort.

This article covers basics integration of firebase realtime database. The other concepts like performs CRUD operations, data validations, firebase access rules also covered.

1. How the Data is Stored – JSON Structured

Firebase realtime database is a schemaless database in which the data is stored in JSON format. Basically the entire database is a big JSON tree with multiple nodes. So when you plan your database, you need to prepare the json structure in way that the data is accessible in easier way by avoiding nesting of child nodes.

```
{
  "users": [
    {
      "name": "Ravi Tamada",
      "email": "ravi@androidhive.info",
      "address": "XXX, XXXX, 1234"
    }
  ],
  "posts": [
  {
   "id": 100,
   "author": "Ravi Tamada",
   "content": "This is awesome firebase realtime database...",
   "timestamp": "13892733894"
  }
 ]
}
```

2. Offline Data

Firebase provides great support when comes to offline data. It automatically stores the data offline when there is no internet connection. When the device connects to internet, all the data will be pushed to realtime database. However enabling disk persistence stores the data offline even though app restarts. Disk persistence can be enabled by calling below one line code.

```
FirebaseDatabase.getInstance().setPersistenceEnabled(true);
```

3. Performing CRUD Operations

Before getting into the android app, I would like to give you basic information about performing CRUD operations on to realtime database. Later we'll combine all these concepts together to build a simple app with firebase realtime database as backend.

In order to perform any operation on to database whether it can be read or write, you need to get the reference to database first. The below code gives you reference to database JSON top node. From here you need to use the child node names to traverse further.

```
private DatabaseReference mDatabase;


mDatabase = FirebaseDatabase.getInstance().getReference();
```

3.1 Inserting Data

To insert data, you can use **setValue()** method on to database reference path. This will create or update the value on path provided. For an example below code inserts a node called “copyright” in json top level.

```
DatabaseReference mRef = mDatabase.getReference("copyright");
```

```
mRef.setValue(Hi I m Mujahid");
```

The realtime database accepts multiple data types String, Long, Double, Boolean, Map<String, Object>, List<Object> to store the data. You can also use custom java objects to store the data which is very helpful when storing model class directly in database.

Let's say you want to store user profile in the database. First you need to create User model with an empty constructor and other properties.

```
@IgnoreExtraProperties

public class User {



  public String name;

  public String email;



  // Default constructor required for calls to

  // DataSnapshot.getValue(User.class)

  public User() {

  }



  public User(String name, String email) {

    this.name = name;

    this.email = email;

  }

}
```

As every user needs a unique Id, you can generate one by calling push() method which creates an empty node with unique key. Then get the reference to 'users' node using child() method. Finally use setValue() method to store the user data.

```
DatabaseReference mDatabase = FirebaseDatabase.getInstance().getReference("users");



// Creating new user node, which returns the unique key value

// new user node would be /users/$userid/
```

```
String userId = mDatabase.push().getKey();


// creating user object
User user = new User("Mujahid", "mujahidmskhan112@gmail.com");


// pushing user to 'users' node using the userId
mDatabase.child(userId).setValue(user);
```

By running the above code, a new user node will be inserted in database with a unique key value. In general, the user id should be acquired by implementing Firebase Auth in your app which gives you authId that acts as user id.

```
{
  "users": [
    "-KTYWvZG4Qn9ZYTc47O6" : {
      "email" : "ravi@androidhive.info",
      "name" : "Ravi Tamada"
    },
    {
      ...
    }
  ]
}
```

3.2 Reading Data

To read the data, you need to attach the ValueEventListener() to the database reference. This event will be triggered whenever there is a change in data in realtime. In onDataChange() you can perform the desired operations onto new data.

Below is the event listener that is triggered whenever there is a change in user profile data that we created earlier.

```
mDatabase.child(userId).addValueEventListener(new ValueEventListener() {
    @Override
```

```
  public void onDataChange(DataSnapshot dataSnapshot) {


    User user = dataSnapshot.getValue(User.class);


    Log.d(TAG, "User name: " + user.getName() + ", email " + user.getEmail());
  }


  @Override
  public void onCancelled(DatabaseError error) {
    // Failed to read value
    Log.w(TAG, "Failed to read value.", error.toException());
  }
});
```

3.3 Updating Data

To update data, you can use the same setValue() method by passing new value. You can also use updateChildren() by passing the path to update data without disturbing other child nodes data.

For example if you want to update only the user email, you can use below code block.

```
String newEmail = "mujahidmskhan!2@gmail.com";


mDatabase.child(userId).child("email").setValue(newEmail);
```

3.4 Deleting Data

To delete data, you can simply call removeValue() method on to database reference. You can also pass null to setValue() method which do the same delete operation.

You can learn more about performing CRUD operations onto more advanced data like Lists of data here.

4. Security & Rules

Firebase rules provides a way to identify user role while performing read and write operations. These rules will acts a security layer on the server before perform any CRUD operation. By default the rules allows user to perform read & write operation only after authentication.

The below rules allow authenticated users only to read or write data.

```
{
  "rules": {
    ".read": "auth != null",
    ".write": "auth != null"
  }
}
```

Below rules allows everyone to read & write data without authentication.

```
{
  "rules": {
    ".read": true,
    ".write": true
  }
}
```

You can also use these rules to validate data before inserting into database. For example below rules validates the name to be less than 50 chars and email to be valid using email regular expression.

```
{
    "rules": {
        ".read": true,
        ".write": true,
        "users": {
            "$user": {
                "name": {
                    ".validate": "newData.isString() && newData.val().length < 50"
                },
                "email": {
                    ".validate": "newData.isString() && newData.val().matches(/^[A-Z0-9._%+-]+@[A-Z0-9.-
]+\\.[A-Z]{2,4}$/i)"
```

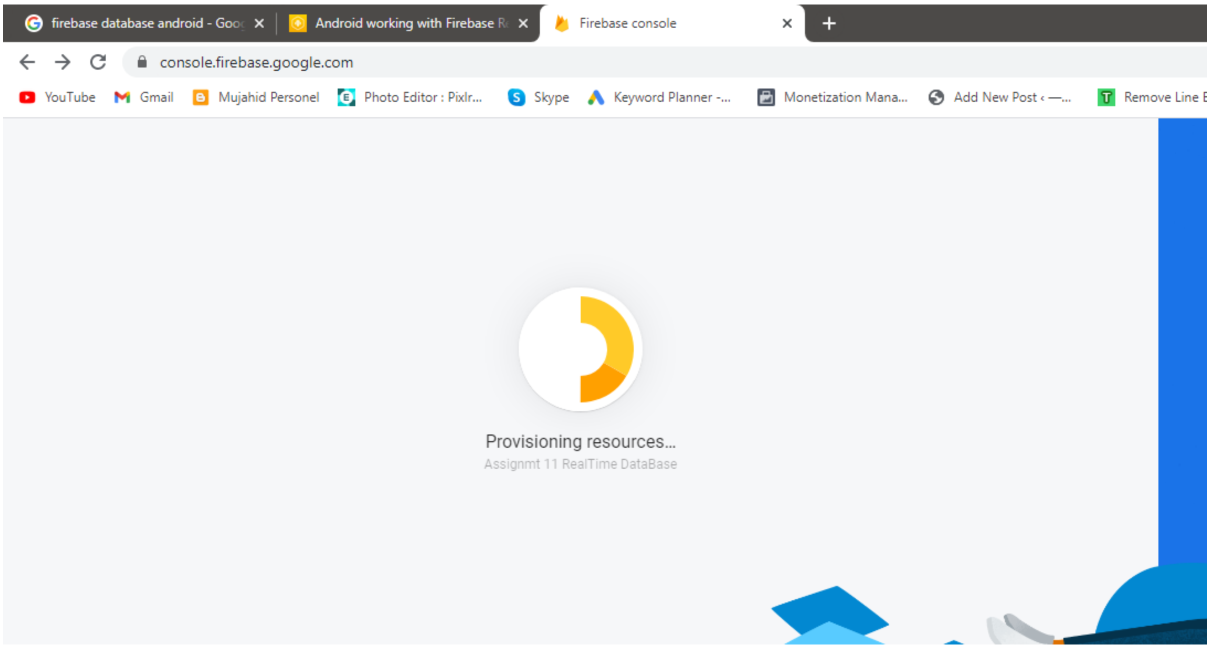
```
                }
            }
        }
    }
}
```

Go through firebase security & rules guide to learn more about the security concepts.

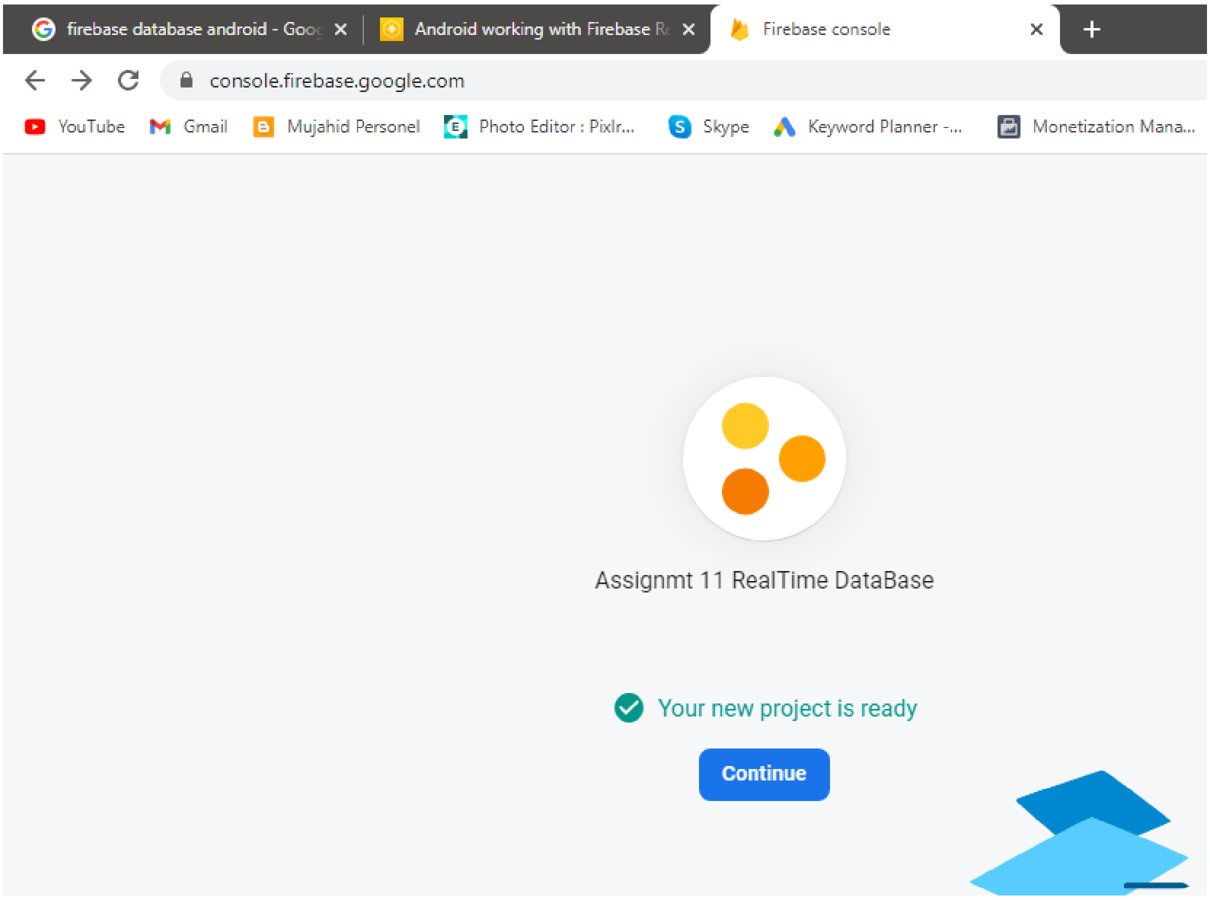
Now we have enough knowledge to get started with an android project. Let's create one and see how to integrate the realtime database with an example app.

5. Creating Android Project

1. First thing you need to do is go to https://firebase.google.com/ and make an account to gain access to their console. After you gain access to the console you can start by creating your first project.



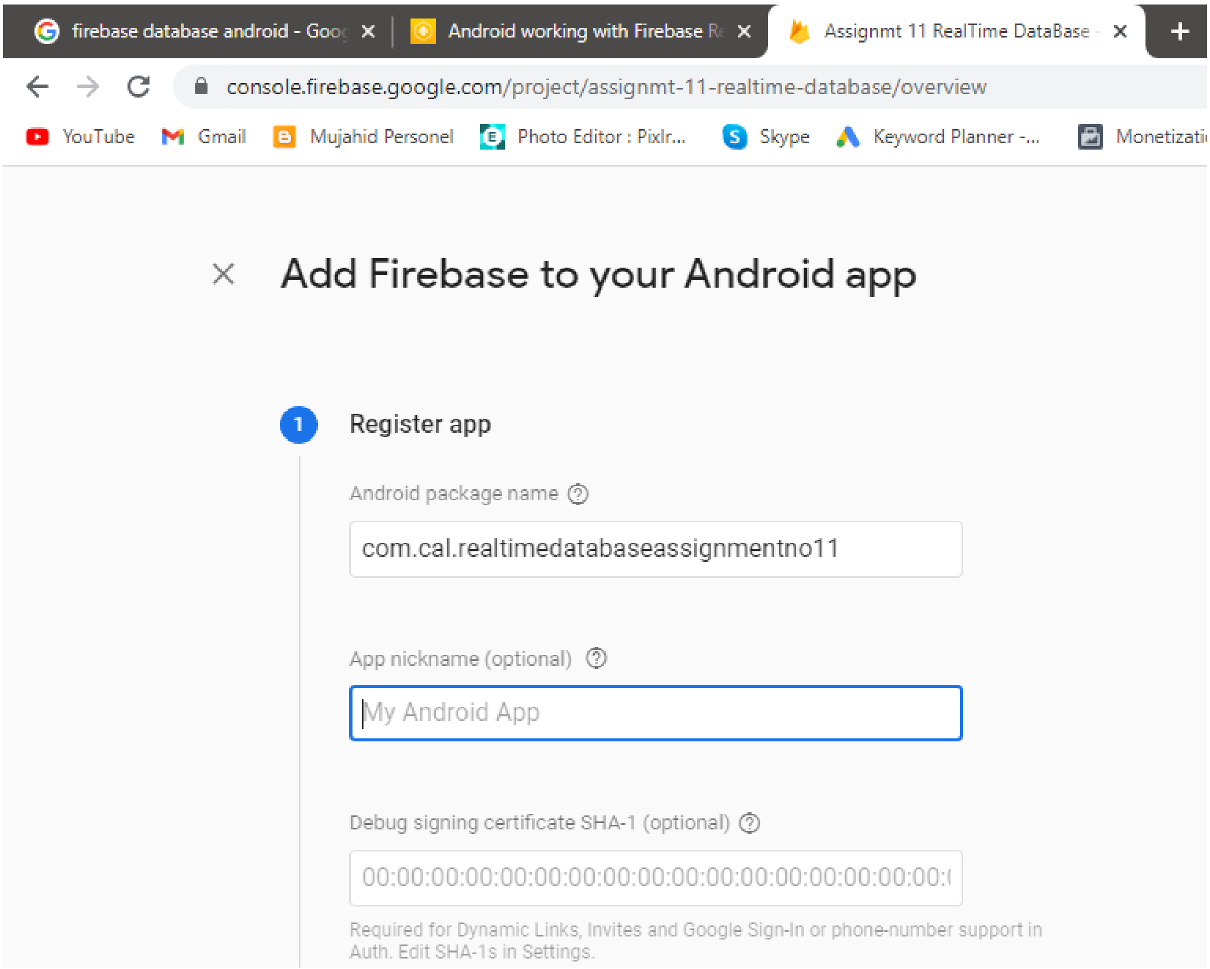
Provisioning resources...
Assignmt 11 RealTime DataBase



firebase database android - Goo
Android working with Firebase R
Firebase console
console.firebase.google.com
YouTube
Gmail
Mujahid Personel
Photo Editor : Pixlr...
Skype
Keyword Planner -...
Monetization Mana...
Assignmt 11 RealTime DataBase
Your new project is ready
Continue

2. Give the package name of project
(com.cal.realtimedatabaseassignmentno11)
in which you are going to integrate the Firebase. Here the google-services.json file will be downloaded when you press add app button.

Firebase firestore


firebase database android - Goo
Android working with Firebase R
Assignmt 11 RealTime DataBase
console.firebase.google.com/project/assignmt-11-realtime-database/overview
YouTube
Gmail
Mujahid Personel
Photo Editor : Pixlr...
Skype
Keyword Planner -...
Monetizati

Add Firebase to your Android app

1 Register app

Android package name

com.cal.realtimedatabaseassignmentno11

App nickname (optional)

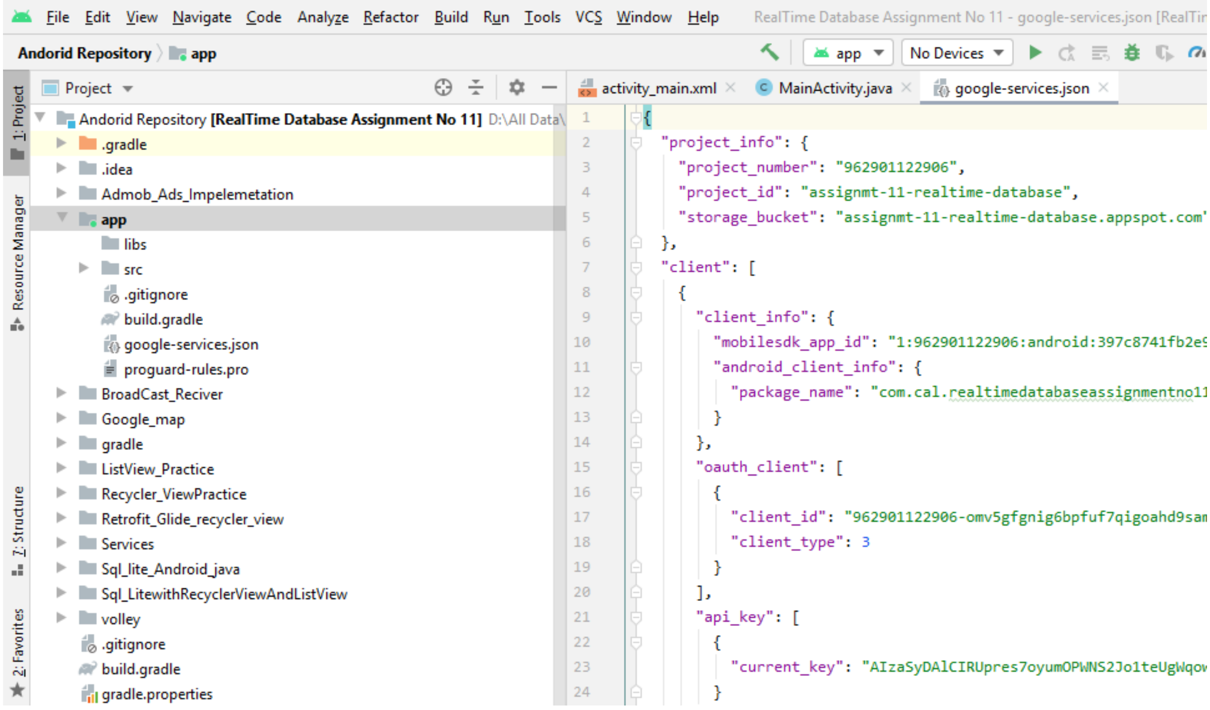
My Android App

Debug signing certificate SHA-1 (optional)

00:00:00:00:00:00:00:00:00:00:00:00:00:00:00:00:00:0

Required for Dynamic Links, Invites and Google Sign-In or phone-number support in Auth. Edit SHA-1s in Settings.

Add Json File in Ap Directory



```
{
  "project_info": {
    "project_number": "962901122906",
    "project_id": "assignmt-11-realtime-database",
    "storage_bucket": "assignmt-11-realtime-database.appspot.com'
  },
  "client": [
    {
      "client_info": {
        "mobilesdk_app_id": "1:962901122906:android:397c8741fb2e
        "android_client_info": {
          "package_name": "com.cal.realtimedatabaseassignmentno11
        }
      },
      "oauth_client": [
        {
          "client_id": "962901122906-omv5gfgnig6bpfuf7qigoahd9sam
          "client_type": 3
        }
      ],
      "api_key": [
        {
          "current_key": "AIzaSyDAlCIRUpres7oyumOPWNS2Jo1teUgWqow
        }
```

Add Class Path: on project Build.gradle

classpath 'com.google.gms:google-services:4.3.4'

Add the Following Dependency on Module Build.gradle.

```
// Import the Firebase BoM
  implementation platform('com.google.firebase:firebase-bom:26.2.0')

  // Add the dependency for the Firebase SDK for Google Analytics
  // When using the BoM, don't specify versions in Firebase
dependencies
  implementation 'com.google.firebase:firebase-analytics'

  // Add the dependencies for any other desired Firebase products
  // https://firebase.google.com/docs/android/setup#available-libraries
```

Now Create Model_User Class

```
package com.example.ecommerce_app.Models;

public class PlateModel {

    public  PlateModel(){
        ///////empty constructor///////////////
    }
    private int plate_imag;

    public PlateModel(int plate_imag) {
        this.plate_imag = plate_imag;
    }

    public int getPlate_imag() {
        return plate_imag;
    }

    public void setPlate_imag(int plate_imag) {
        this.plate_imag = plate_imag;
    }
}
```

activity_main.xml Code:

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
```

```
android:layout_height="match_parent"
android:orientation="vertical"
android:background="@drawable/bg"
tools:context=".Admin.AdminDashboardActivity">

<ScrollView
    android:layout_width="match_parent"
    android:layout_height="wrap_content">

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:orientation="vertical">
        <LinearLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:gravity="center"
            android:orientation="horizontal">
        </LinearLayout>

<de.hdodenhof.circleimageview.CircleImageView
    android:id="@+id/id_getImage"
    android:padding="10dp"
    android:src="@drawable/ic_baseline_image_24"
    android:layout_gravity="center"
    android:layout_width="200dp"
    android:layout_height="200dp"
    />

        <Button
            android:id="@+id/button_select_Image"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:background="@color/colorAccent"
            android:drawableRight="@drawable/right_arrow"
            android:drawableTint="@color/colorWhite"
            android:elevation="20dp"
            android:text="Select Image"
            android:textColor="@color/colorWhite"
            android:textSize="15sp"
            android:layout_marginBottom="6dp"
            android:textStyle="bold" />
<EditText
    android:id="@+id/edit_product"
    android:layout_width="match_parent"
    android:layout_height="50dp"
    android:layout_marginLeft="16dp"
    android:layout_marginTop="25dp"
    android:layout_marginRight="16dp"
    android:background="@drawable/skip_button"
    android:drawableLeft="@drawable/orders"
    android:drawablePadding="8dp"
    android:drawableTint="@color/colorAccent"
    android:elevation="20dp"
    android:hint="Products"
    android:inputType="text"
    android:padding="16dp" />
<EditText
    android:id="@+id/edit_texture"
```

```xml
        android:layout_width="match_parent"
        android:layout_height="50dp"
        android:layout_marginLeft="16dp"
        android:layout_marginTop="25dp"
        android:layout_marginRight="16dp"
        android:background="@drawable/skip_button"
        android:drawableLeft="@drawable/orders"
        android:drawablePadding="8dp"
        android:drawableTint="@color/colorAccent"
        android:elevation="20dp"
        android:hint="Texture"
        android:inputType="text"
        android:padding="16dp" />
    <EditText
        android:id="@+id/edit_colour"
        android:layout_width="match_parent"
        android:layout_height="50dp"
        android:layout_marginLeft="16dp"
        android:layout_marginTop="25dp"
        android:layout_marginRight="16dp"
        android:background="@drawable/skip_button"
        android:drawableLeft="@drawable/orders"
        android:drawablePadding="8dp"
        android:drawableTint="@color/colorAccent"
        android:elevation="20dp"
        android:hint="Couleur"
        android:inputType="text"
        android:padding="16dp" />
    <EditText
        android:id="@+id/edit_quantity"
        android:layout_width="match_parent"
        android:layout_height="50dp"
        android:layout_marginLeft="16dp"
        android:layout_marginTop="25dp"
        android:layout_marginRight="16dp"
        android:background="@drawable/skip_button"
        android:drawableLeft="@drawable/orders"
        android:drawablePadding="8dp"
        android:drawableTint="@color/colorAccent"
        android:elevation="20dp"
        android:hint="Quantite"
        android:inputType="text"
        android:padding="16dp" />
    <EditText
        android:id="@+id/edit_taille"
        android:layout_width="match_parent"
        android:layout_height="50dp"
        android:layout_marginLeft="16dp"
        android:layout_marginTop="25dp"
        android:layout_marginRight="16dp"
        android:background="@drawable/skip_button"
        android:drawableLeft="@drawable/name_icon"
        android:drawablePadding="8dp"
        android:drawableTint="@color/colorAccent"
        android:elevation="20dp"
        android:hint="Taille"
        android:inputType="text"
        android:padding="16dp" />
    <EditText
```

```
        android:id="@+id/edit_prex"
        android:layout_width="match_parent"
        android:layout_height="50dp"
        android:layout_marginLeft="16dp"
        android:layout_marginTop="25dp"
        android:layout_marginRight="16dp"
        android:background="@drawable/skip_button"
        android:drawableLeft="@drawable/name_icon"
        android:drawablePadding="8dp"
        android:drawableTint="@color/colorAccent"
        android:elevation="20dp"
        android:hint="Prix de vente"
        android:inputType="text"
        android:padding="16dp" />

    <Button
        android:id="@+id/buttonAdd_Items"
        android:layout_width="343dp"
        android:layout_height="wrap_content"
        android:layout_marginStart="16dp"
        android:layout_marginLeft="10dp"
        android:layout_marginTop="10dp"
        android:layout_marginEnd="18dp"
        android:layout_marginRight="18dp"
        android:background="@color/colorAccent"
        android:drawableRight="@drawable/right_arrow"
        android:drawableTint="@color/colorWhite"
        android:elevation="20dp"
        android:text="Add"
        android:textColor="@color/colorWhite"
        android:textSize="15sp"
        android:layout_marginBottom="16dp"
        android:textStyle="bold" />
    <ProgressBar
        android:id="@+id/progress_circular"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="center"
        android:visibility="invisible"
        />
        </LinearLayout>
    </ScrollView>
</LinearLayout>
```

> getReference(“app_title”) create a node named app_title which stores the toolbar title.

> getReference(“users”) gets reference to users node.

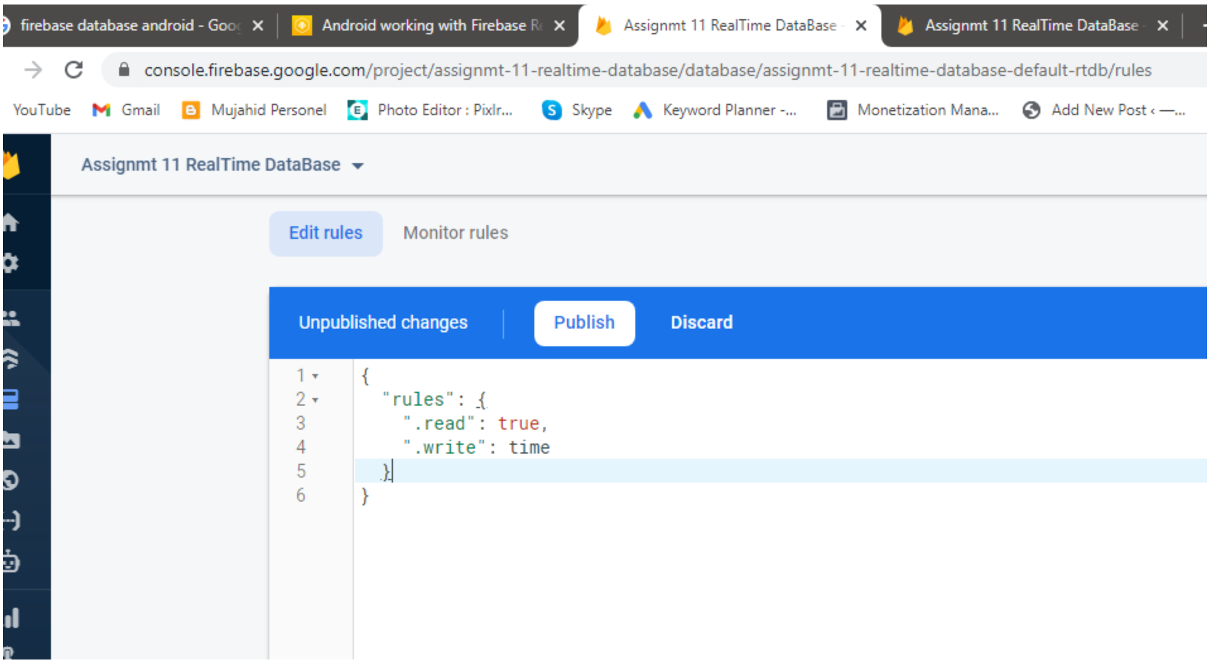
> createUser() method stores a new user in realtime database

> updateUser() method updates user information like name and email.

MainActive.java Code

```
package com.example.ecommerce_app.Admin;

import androidx.annotation.NonNull;
import androidx.annotation.Nullable;
import androidx.appcompat.app.AppCompatActivity;

import android.content.ContentResolver;
import android.content.Intent;
import android.graphics.Bitmap;
import android.graphics.BitmapFactory;
import android.net.Uri;
import android.os.Bundle;
import android.view.View;
import android.webkit.MimeTypeMap;
import android.widget.Button;
import android.widget.EditText;
import android.widget.ImageView;
import android.widget.ProgressBar;
import android.widget.Toast;

import com.example.ecommerce_app.MainActivity;
import com.example.ecommerce_app.R;
import com.google.android.gms.tasks.Continuation;
import com.google.android.gms.tasks.OnCompleteListener;
import com.google.android.gms.tasks.Task;
import com.google.firebase.database.DataSnapshot;
import com.google.firebase.database.DatabaseError;
import com.google.firebase.database.DatabaseReference;
import com.google.firebase.database.FirebaseDatabase;
import com.google.firebase.database.ValueEventListener;
import com.google.firebase.storage.FirebaseStorage;
import com.google.firebase.storage.StorageReference;
import com.google.firebase.storage.StorageTask;
import com.google.firebase.storage.UploadTask;
import com.mukesh.image_processing.ImageProcessor;

import java.io.InputStream;

public class AdminDashboardActivity extends AppCompatActivity {
    EditText products,texture,colour,quantity,taile,prixde;

    ImageView get_image;
    private DatabaseReference mdataref;
    private StorageReference mstorageref;
    private StorageTask mUploadtask;
    private Uri imageUri;
    ProgressBar progressBar;
    Bitmap bitmap;
    private static final int PICK_IMAGE_REQUEST=1;
    Button
btn_add_items,btn_select_image,btn_filter1,btn_filter2,btn_filter3,btn_filter4,btn
_filter5;
```

```
    long max_id;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_admin_dashboard);
        products = findViewById(R.id.edit_product);
        texture = findViewById(R.id.edit_texture);
        colour = findViewById(R.id.edit_colour);
        quantity = findViewById(R.id.edit_quantity);
        taile = findViewById(R.id.edit_taille);
        prixde = findViewById(R.id.edit_prex);
        btn_add_items = findViewById(R.id.buttonAdd_Items);
        get_image=findViewById(R.id.id_getImage);
        progressBar=findViewById(R.id.progress_circular);

        btn_select_image=findViewById(R.id.button_select_Image);

        btn_select_image.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                openFileChooser();
            }
        });

        mstorageref= FirebaseStorage.getInstance().getReference("ecomerce");
        mdataref= FirebaseDatabase.getInstance().getReference("ecomerce");


        mdataref.addValueEventListener(new ValueEventListener() {
            @Override
            public void onDataChange(@NonNull DataSnapshot snapshot) {
                if (snapshot.exists())
                    max_id=(snapshot.getChildrenCount());
            }

            @Override
            public void onCancelled(@NonNull DatabaseError error) {

            }
        });



        btn_add_items.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                if (mUploadtask!=null && mUploadtask.isInProgress()){
                    Toast.makeText(AdminDashboardActivity.this, "Upload in
Progress", Toast.LENGTH_SHORT).show();

                }else {
                    UploadData();
                    progressBar.setVisibility(View.VISIBLE);
                }


            }
        });
    }

    private String getFileExtension(Uri uri) {
        ContentResolver cR = getContentResolver();
        MimeTypeMap mime = MimeTypeMap.getSingleton();
        return mime.getExtensionFromMimeType(cR.getType(uri));
    }
    private void UploadData() {
        if (imageUri != null)
        {
            final StorageReference fileReference =
mstorageref.child(System.currentTimeMillis()
                    + "." + getFileExtension(imageUri));

            fileReference.putFile(imageUri).continueWithTask(new
Continuation<UploadTask.TaskSnapshot, Task<Uri>>()
            {
                @Override
                public Task<Uri> then(@NonNull Task<UploadTask.TaskSnapshot> task)
throws Exception
                {


                    if (!task.isSuccessful())
                    {
                        throw task.getException();
                    }
                    return fileReference.getDownloadUrl();
                }
            }).addOnCompleteListener(new OnCompleteListener<Uri>()
            {
                @Override
                public void onComplete(@NonNull Task<Uri> task)
                {

                    if (task.isSuccessful())
                    {
                        Uri downloadUri = task.getResult();
//                        Log.e(TAG, "then: " + downloadUri.toString());


                        Product_add_pojo upload = new
Product_add_pojo(products.getText().toString().trim(),texture.getText().toString()
.trim(),downloadUri.toString()

,colour.getText().toString().trim(),quantity.getText().toString().trim(),taile.get
Text().toString().trim(),prixde.getText().toString().trim());


                        mdataref.child(String.valueOf(max_id+1)).setValue(upload);
                        Toast.makeText(AdminDashboardActivity.this, "Uploaded",
Toast.LENGTH_SHORT).show();
                        progressBar.setVisibility(View.GONE);
                        startActivity(new Intent(AdminDashboardActivity.this,
MainActivity.class));
                        finish();
                    } else
                    {
                        Toast.makeText(AdminDashboardActivity.this, "upload 
failed: " + task.getException().getMessage(), Toast.LENGTH_SHORT).show();
                    }
                }
            });
        }
    }

    private void openFileChooser() {
        Intent intent = new Intent();
        intent.setType("image/*");
        intent.setAction(Intent.ACTION_GET_CONTENT);
        startActivityForResult(intent, PICK_IMAGE_REQUEST );
    }
    @Override
    protected void onActivityResult(int requestCode, int resultCode, @Nullable 
Intent data) {
        if (requestCode==PICK_IMAGE_REQUEST && resultCode==RESULT_OK  && data != 
null && data.getData() != null){

            imageUri=data.getData();
            try {
                InputStream 
inputStream=getContentResolver().openInputStream(imageUri);
                bitmap= BitmapFactory.decodeStream(inputStream);


                get_image.setImageBitmap(bitmap);

            }catch (Exception ex){}
        }
        super.onActivityResult(requestCode, resultCode, data);

    }
}
```


firebase database android - Goo
Android working with Firebase R
Assignmt 11 RealTime DataBase
Assignmt 11 RealTime DataBase
console.firebase.google.com/project/assignmt-11-realtime-database/database/assignmt-11-realtime-database-default-rtdb/rules
YouTube
Gmail
Mujahid Personel
Photo Editor : Pixlr...
Skype
Keyword Planner -...
Monetization Mana...
Add New Post ‹ —...
Assignmt 11 RealTime DataBase
Edit rules
Monitor rules
Unpublished changes
Publish
Discard

```
{
  "rules": {
    ".read": true,
    ".write": time
  }
}
```

Add SearchView Recycler View in Product Activity

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    android:background="@drawable/bg">

    <LinearLayout
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="30dp"
        android:background="@color/colorWhite"
        android:orientation="horizontal">
        <ImageView
            android:id="@+id/navigationBar"
            android:layout_width="40dp"
            android:layout_height="40dp"
            android:layout_marginLeft="10dp"
            android:layout_gravity="center_vertical"
            android:layout_marginRight="6dp"
            android:contentDescription="Navigation Bar"
            android:src="@drawable/ic_baseline_menu_24" />
        <androidx.appcompat.widget.SearchView
            android:id="@+id/id_edt_search"
            android:layout_width="250dp"
            android:layout_height="44dp"
            android:ems="10"
            android:background="@drawable/search_layout"
            android:hint="Search Here"
            android:inputType="textPersonName"
            android:layout_gravity="center_vertical"
            android:textSize="16sp"
            android:textColor="#999999"
            android:paddingBottom="10dp"
            android:layout_marginLeft="4dp"
            android:layout_marginRight="4dp"
            android:paddingRight="15dp"
            android:paddingLeft="20dp"
            android:paddingTop="6dp"

            />

    </LinearLayout>

    <androidx.recyclerview.widget.RecyclerView
        android:id="@+id/id_recyclerview"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:layout_marginTop="2dp"
        android:padding="22dp"
        />

</LinearLayout>
```

Product Fragment.Java Code:

```
package com.example.ecommerce_app.Fragments;

import android.app.Dialog;
import android.app.ProgressDialog;
import android.content.Intent;
import android.net.Uri;
import android.os.Bundle;

import androidx.annotation.NonNull;
import androidx.drawerlayout.widget.DrawerLayout;
import androidx.fragment.app.Fragment;
import androidx.recyclerview.widget.LinearLayoutManager;
import androidx.recyclerview.widget.RecyclerView;

import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.ImageView;
import android.widget.RelativeLayout;
import android.widget.SearchView;
import android.widget.TextView;
import android.widget.Toast;

import com.example.ecommerce_app.Adapters.Products_List_Adapter;
import com.example.ecommerce_app.Admin.AdminDashboardActivity;
import com.example.ecommerce_app.Admin.Product_add_pojo;
import com.example.ecommerce_app.EmailLoginRegister.EmailRegisterActivity;
import com.example.ecommerce_app.HomeActivity;
import com.example.ecommerce_app.R;
import com.example.ecommerce_app.Show_User_Feedback;
import com.example.ecommerce_app.User_Feed_Back_Activity;
import com.google.android.material.navigation.NavigationView;
import com.google.firebase.database.DataSnapshot;
import com.google.firebase.database.DatabaseError;
import com.google.firebase.database.DatabaseReference;
import com.google.firebase.database.FirebaseDatabase;
import com.google.firebase.database.ValueEventListener;

import java.util.ArrayList;
import java.util.List;

import de.hdodenhof.circleimageview.CircleImageView;


public class ProductFragment extends Fragment implements View.OnClickListener{
    androidx.appcompat.widget.SearchView search_etd;
    CircleImageView circle_btn, admin_section;
    private RecyclerView recyclerView;
    SearchView searchView;
    Products_List_Adapter adapterClass;
    DatabaseReference mDatabaseRef;

    private List<Product_add_pojo> mUser;
    DrawerLayout drawerLayout;
    ImageView navigationBar,addtocart;
    NavigationView navigationView;
    private View view;
    TextView textView;
    private RelativeLayout loginSingUp,bookmark,gold;
    private TextView
your_orders,favorites_orders,address_book,online_ordering_help,send_feedback,repor
t_safety_emergency,rate_playstore;

    @Override
    public View onCreateView(LayoutInflater inflater, ViewGroup container,
                             Bundle savedInstanceState) {
        // Inflate the layout for this fragment
        view = inflater.inflate(R.layout.fragment_products, container, false);


        recyclerView = view.findViewById(R.id.id_recyclerview);
        search_etd=view.findViewById(R.id.id_edt_search);
        recyclerView.setHasFixedSize(true);
        recyclerView.setLayoutManager(new LinearLayoutManager(getContext()));

        mDatabaseRef = FirebaseDatabase.getInstance().getReference("ecomerce");



        onSetNavigationDrawerEvents();
        return view;
    }

    private void onSetNavigationDrawerEvents() {
        drawerLayout =  view.findViewById(R.id.drawerLayout);
        navigationView = view.findViewById(R.id.navigationView);
//        drawerLayout.openDrawer(GravityCompat.START);
        navigationBar = (ImageView) view.findViewById(R.id.navigationBar);
        // id of layouts
        loginSingUp = view.findViewById(R.id.relativeLayout);
        bookmark = view.findViewById(R.id.relativeLayout3);
        gold = view.findViewById(R.id.relativeLayout4);
        //id of textviews
        your_orders = view.findViewById(R.id.your_orders);
        favorites_orders = view.findViewById(R.id.favorites_orders);
        address_book = view.findViewById(R.id.address_book);
        online_ordering_help = view.findViewById(R.id.online_ordering_help);
        send_feedback = view.findViewById(R.id.send_feedback);
        report_safety_emergency = view.findViewById(R.id.report_safety_emergency);
        rate_playstore = view.findViewById(R.id.rate_playstore);
        // set clicklistner
        navigationBar.setOnClickListener(this);
        //set click listner on relative layouts
        loginSingUp.setOnClickListener(this);
        bookmark.setOnClickListener(this);
        gold.setOnClickListener(this);
        //set click listner on textviews
        your_orders.setOnClickListener(this);
        favorites_orders.setOnClickListener(this);
        address_book.setOnClickListener(this);
        online_ordering_help.setOnClickListener(this);
        send_feedback.setOnClickListener(this);
        report_safety_emergency.setOnClickListener(this);


    }

    @Override
    public void onClick(View v) {
        switch (v.getId()){
            case R.id.navigationBar:
                drawerLayout.openDrawer(navigationView, true);
                break;
            case R.id.relativeLayout:
                Intent intent = new Intent(getContext(),
EmailRegisterActivity.class);
                startActivity(intent);
                Toast.makeText(getContext(), "Wellcome To The KinkyAirbean",
Toast.LENGTH_SHORT).show();
                break;
            case R.id.relativeLayout3:
                Intent intent1 = new Intent(getContext(), HomeActivity.class);
                startActivity(intent1);
                Toast.makeText(getContext(), " Wellcome to Products of
Kinkyairbean", Toast.LENGTH_SHORT).show();
                break;
            case R.id.your_orders:
                Intent intent2 = new Intent(getContext(),
Show_User_Feedback.class);
                startActivity(intent2);
                Toast.makeText(getContext(), "Clients After products of
kinkyairbaen", Toast.LENGTH_SHORT).show();
                break;
            case R.id.favorites_orders:
                Intent intent3 = new Intent(getContext(),
User_Feed_Back_Activity.class);
                startActivity(intent3);
                Toast.makeText(getContext(), "Feedback and send images with
products", Toast.LENGTH_SHORT).show();
                break;
            case R.id.address_book:
                Toast.makeText(getContext(), "address_book",
Toast.LENGTH_SHORT).show();
                break;
            case R.id.online_ordering_help:
                Intent gmail = new Intent(Intent.ACTION_VIEW , Uri.parse("mailto:"
+ "kinkyaribbean@gmail.com"));
                gmail.putExtra(Intent.EXTRA_SUBJECT,"what Can I Help You");
                gmail.putExtra(Intent.EXTRA_TEXT,"This is Exceed Solution");
                startActivity(gmail);
                Toast.makeText(getContext(), "Your online_ordering_help",
Toast.LENGTH_SHORT).show();
                break;
            case R.id.send_feedback:
                Intent gmail1 = new Intent(Intent.ACTION_VIEW ,
Uri.parse("mailto:" + "kinkyaribbean@gmail.com"));
                gmail1.putExtra(Intent.EXTRA_SUBJECT,"what Can I Help You");
                gmail1.putExtra(Intent.EXTRA_TEXT,"This is Exceed Solution");
                startActivity(gmail1);
                Toast.makeText(getContext(), "send_feedback", 
Toast.LENGTH_SHORT).show();
                break;
            case R.id.report_safety_emergency:
                Toast.makeText(getContext(), "report_safety_emergency", 
Toast.LENGTH_SHORT).show();
                Intent gmail2 = new Intent(Intent.ACTION_VIEW , 
Uri.parse("mailto:" + "kinkyaribbean@gmail.com"));
                gmail2.putExtra(Intent.EXTRA_SUBJECT,"what Can I Help You");
                gmail2.putExtra(Intent.EXTRA_TEXT,"This is Exceed Solution");
                startActivity(gmail2);
                break;
            case R.id.rate_playstore:
                Toast.makeText(getContext(), "rate_playstore", 
Toast.LENGTH_SHORT).show();
                startActivity(new 
Intent(Intent.ACTION_VIEW,Uri.parse("https://play.google.com/store/apps/details?id
=com.freeblinkingapps.issb_test_preparation")));
                break;
        }
    }

    @Override
    public void onStart() {
        final ProgressDialog Dialog = new ProgressDialog(getActivity());
        Dialog.setMessage("Wait Loading In Just a Seconds...");

        Dialog.show();
        super.onStart();
        if (mDatabaseRef != null){
            mDatabaseRef.addValueEventListener(new ValueEventListener() {
                @Override
                public void onDataChange(@NonNull DataSnapshot snapshot) {
                    if (snapshot.exists()){
                        mUser=new ArrayList<>();
                        for (DataSnapshot ds : snapshot.getChildren()){
                            mUser.add(ds.getValue(Product_add_pojo.class));
//                            User user=snapshot.getValue(User.class);
//                            user.setUid(snapshot.getKey());
                            Products_List_Adapter mAdapter=new 
Products_List_Adapter(mUser,getContext());
                            recyclerView.setAdapter(mAdapter);

                        }
                    }

                    Dialog.dismiss();

                }

                @Override
                public void onCancelled(@NonNull DatabaseError error) {
                    Toast.makeText(getContext(), error.getMessage(), 
Toast.LENGTH_SHORT).show();
```

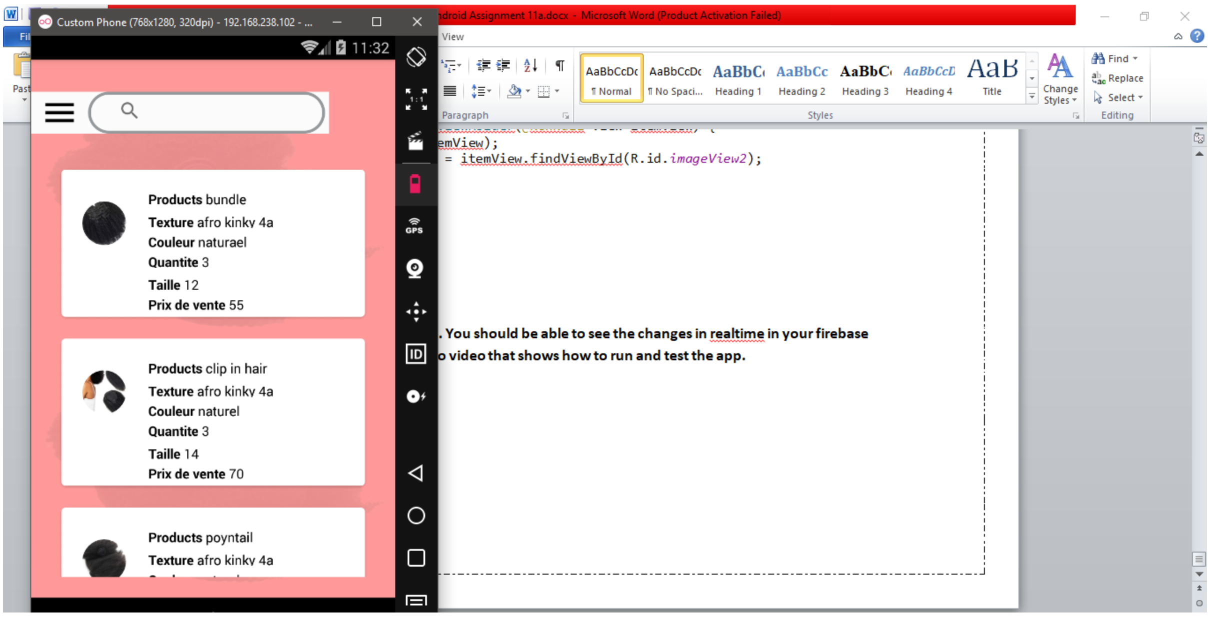
```
                }
            });
        }
        if (search_etd !=null){
            search_etd.setOnQueryTextListener(new
androidx.appcompat.widget.SearchView.OnQueryTextListener() {
                @Override
                public boolean onQueryTextSubmit(String query) {
                    return false;
                }

                @Override
                public boolean onQueryTextChange(String s) {
                    search(s);
                    return true;
                }
            });
        }
    }
    private void search(String str){

        ArrayList<Product_add_pojo> mylist=new ArrayList<>();
        for (Product_add_pojo object : mUser){
            if (object.getName().toLowerCase().contains(str.toLowerCase())){
                mylist.add(object);
            }
        }

        Products_List_Adapter adapterClass =new
Products_List_Adapter(mylist,getContext());
        recyclerView.setAdapter(adapterClass);


    }

}
```

Adapter Class:

```
package com.example.ecommerce_app.Adapters;

import android.content.Context;
import android.text.Layout;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.ImageView;

import androidx.annotation.NonNull;
import androidx.recyclerview.widget.RecyclerView;

import com.bumptech.glide.Glide;
import com.example.ecommerce_app.Models.PlateModel;
import com.example.ecommerce_app.R;

import java.util.List;
```

```
public class PlatesAdapter extends
RecyclerView.Adapter<PlatesAdapter.plateViewHolder> {
    private List<PlateModel> plateModelList;
    private Context context;

    public PlatesAdapter(List<PlateModel> plateModelList, Context context) {
        this.plateModelList = plateModelList;
        this.context = context;
    }

    @NonNull
    @Override
    public plateViewHolder onCreateViewHolder(@NonNull ViewGroup viewGroup, int
viewType) {
        View view =
LayoutInflater.from(viewGroup.getContext()).inflate(R.layout.layout_plates,viewGro
up,false);
        return new plateViewHolder(view);
    }

    @Override
    public void onBindViewHolder(@NonNull plateViewHolder holder, int position) {
    PlateModel plateModel = plateModelList.get(position);

Glide.with(context.getApplicationContext()).load(plateModel.getPlate_imag()).into(
holder.plateimg);
    }

    @Override
    public int getItemCount() {
        return plateModelList.size();
    }

    public class plateViewHolder extends RecyclerView.ViewHolder {
        private ImageView plateimg;
        public plateViewHolder(@NonNull View itemView) {
            super(itemView);
            plateimg = itemView.findViewById(R.id.imageView2);
        }
    }
}
```

Run & test the app once. You should be able to see the changes in realtime in your firebase console. Check the Demo video that shows how to run and test the app.



Products bundle
Texture afro kinky 4a
Couleur naturael
Quantite 3
Taille 12
Prix de vente 55

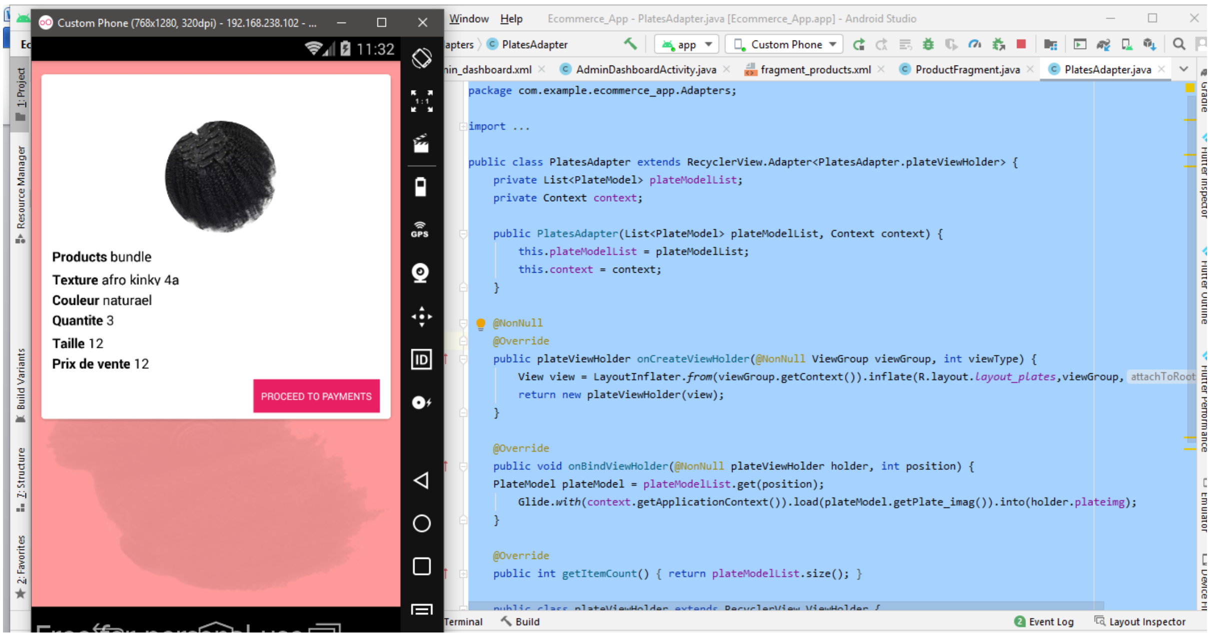
Products clip in hair
Texture afro kinky 4a
Couleur naturel
Quantite 3
Taille 14
Prix de vente 70

Products poyntail
Texture afro kinky 4a

```
emView);
 = itemView.findViewById(R.id.imageView2);
```

. You should be able to see the changes in realtime in your firebase
o video that shows how to run and test the app.

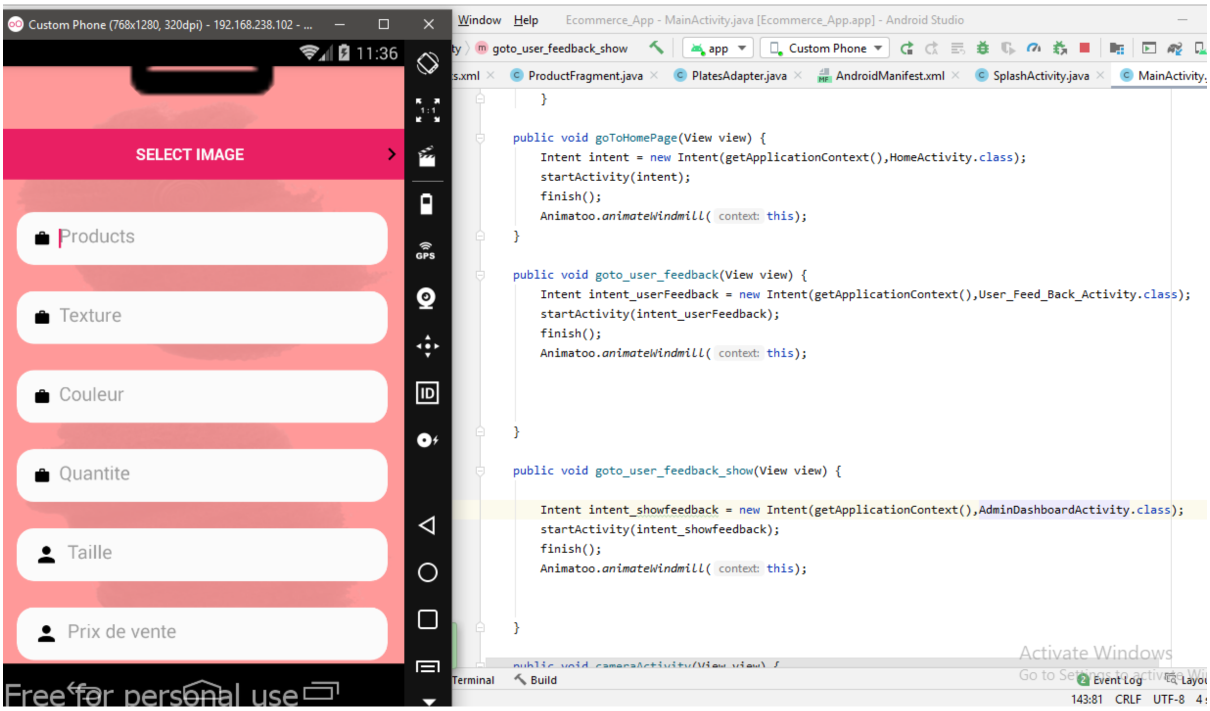
When Click On Item Goto Product Detail:



Products bundle
Texture afro kinky 4a
Couleur naturael
Quantite 3
Taille 12
Prix de vente 12

```
package com.example.ecommerce_app.Adapters;

import ...

public class PlatesAdapter extends RecyclerView.Adapter<PlatesAdapter.plateViewHolder> {
    private List<PlateModel> plateModelList;
    private Context context;

    public PlatesAdapter(List<PlateModel> plateModelList, Context context) {
        this.plateModelList = plateModelList;
        this.context = context;
    }

    @NonNull
    @Override
    public plateViewHolder onCreateViewHolder(@NonNull ViewGroup viewGroup, int viewType) {
        View view = LayoutInflater.from(viewGroup.getContext()).inflate(R.layout.layout_plates,viewGroup, attachToRoot
        return new plateViewHolder(view);
    }

    @Override
    public void onBindViewHolder(@NonNull plateViewHolder holder, int position) {
    PlateModel plateModel = plateModelList.get(position);
        Glide.with(context.getApplicationContext()).load(plateModel.getPlate_imag()).into(holder.plateimg);
    }

    @Override
    public int getItemCount() { return plateModelList.size(); }

    public class plateViewHolder extends RecyclerView.ViewHolder {
```

Add Products



```java
    }

public void goToHomePage(View view) {
    Intent intent = new Intent(getApplicationContext(),HomeActivity.class);
    startActivity(intent);
    finish();
    Animatoo.animateWindmill( context: this);
}

public void goto_user_feedback(View view) {
    Intent intent_userFeedback = new Intent(getApplicationContext(),User_Feed_Back_Activity.class);
    startActivity(intent_userFeedback);
    finish();
    Animatoo.animateWindmill( context: this);



}

public void goto_user_feedback_show(View view) {

    Intent intent_showfeedback = new Intent(getApplicationContext(),AdminDashboardActivity.class);
    startActivity(intent_showfeedback);
    finish();
    Animatoo.animateWindmill( context: this);


}

public void cameraActivity(View view) {
```