**Note:** .xml file formats lack the functionality to include comments and consequently do not contain any comments.

**File Name:** Recommendation\_Algorithm.java

**Purpose:** The algorithm which determines the order of products

––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––

package com.example.usshop;  
  
import android.util.Log;  
  
import androidx.annotation.NonNull;  
  
import com.google.firebase.auth.FirebaseAuth;  
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;  
  
public class **Recommendation\_Algorithm** {  
 private static List<BackendDataStorage> AllProducts = new ArrayList<>();  
 List<String> AllUsers = new ArrayList<>();  
 private static List<Integer> AllUnratedRows = new ArrayList<>();  
  
 DatabaseReference DR;  
  
 private static int TempUserAmount, TempProductAmount, UserIndex, TempCounter = 0;  
 private static double TempDotProduct, UserLength = 0, TempOtherUserLength = 0, SimilarityDotProduct = 0, SimilaritySum;  
  
 private static double[][] Ratings;  
  
 public **Recommendation\_Algorithm** (List<BackendDataStorage> TempProducts) {  
 // Setting Variables  
 AllProducts = TempProducts;  
 TempProductAmount = AllProducts.size();  
 DR = FirebaseDatabase.getInstance().getReference("Recommendation\_Algorithm");  
  
 // Fill other necessary variables  
 DR.addValueEventListener(new ValueEventListener() {  
 @Override  
 public void **onDataChange**(@NonNull DataSnapshot snapshot) {  
 //Empty future used variables to reset  
 AllUsers.clear();  
 TempCounter = 0;  
  
 for (DataSnapshot UsersSnapshot : snapshot.getChildren()) {  
 // Adding all users  
 AllUsers.add(UsersSnapshot.getKey());  
  
 // Getting UserIndex  
 if (UsersSnapshot.getKey().equals(FirebaseAuth.getInstance().getCurrentUser().getUid())) {  
 UserIndex = TempCounter;  
 }  
 TempCounter++;  
 }  
 TempUserAmount = AllUsers.size();  
 Ratings = new double[TempUserAmount][TempProductAmount + 1]; // Row 0 is for Similarity  
  
 // Now that Ratings' boundaries are set, filling it with its values is necessary  
 Ratings = AllZeros(Ratings);  
 for (DataSnapshot Users : snapshot.getChildren()) {  
 for (DataSnapshot Products : Users.getChildren()) {  
 // For each product, look through the 'AllUsers' & 'AllProducts' 2D arrays  
 // to get the index of the rating  
 for (int user = 0; user < AllUsers.size(); user++) {  
 for (int product = 0; product < AllProducts.size(); product++) {  
 if (Users.getKey().equals(AllUsers.get(user)) && Products.getKey().equals(AllProducts.get(product).getTitle())) {  
 Ratings[user][product + 1] = Products.getValue(Double.class);  
 }  
 }  
 }  
 }  
 }  
  
 // Employing Collaborative Filtering  
 Get\_Cosine\_Similarity();  
 Predictions();  
  
 // Getting all of current users' Ratings & Predicted Ratings  
 double[] TempRatings = new double[TempProductAmount];  
 for (int i = 0; i < (Ratings[UserIndex].length - 1); i++) {  
 TempRatings[i] = Ratings[UserIndex][i + 1];  
 }  
 Selection\_Sort(TempRatings);  
  
 // Finally display the items  
 Shop.shop.CreateItems(AllProducts);  
 }  
  
 @Override  
 public void **onCancelled**(@NonNull DatabaseError error) {  
 Log.d("Error", "Failed to read value: " + error.getMessage());  
 }  
 });  
 }  
  
 // Getting similarity of current user to all other users using cosine similarity  
 public static void **Get\_Cosine\_Similarity** () {  
 // Cosine Similarity is (x . y) / (||x|| \* ||y||) where  
 // x . y is dot product of vectors x and y  
 // ||x|| \* ||y|| is the cross product of the length of vectors x and y  
  
 // Getting ||x||  
 for (int i = 1; i < Ratings[UserIndex].length; i++) {  
 UserLength += Math.pow(Ratings[UserIndex][i], 2);  
  
 if (Ratings[UserIndex][i] == 0) {  
 AllUnratedRows.add(i);  
 }  
 }  
 UserLength = Math.sqrt(UserLength);  
  
 boolean Exclude = false;  
 // Getting similarity for each user  
 for (int User = 0; User < Ratings.length; User++) {  
 // Resetting variables for each new user  
 TempDotProduct = 0; TempOtherUserLength = 0;  
  
 // Going through each users ratings & getting cosine similarity  
 for (int Rating = 1; Rating < Ratings[User].length; Rating++) {  
 if (User != UserIndex && Ratings[User][Rating] != 0) {  
 // Dot Product for each other user  
 TempDotProduct += Ratings[UserIndex][Rating] \* Ratings[User][Rating];  
  
 // Step 1 of getting ||y||  
 TempOtherUserLength += Math.pow(Ratings[User][Rating], 2);  
 Exclude = false;  
 }  
 }  
 // Step 2 of getting ||y||  
 TempOtherUserLength = Math.sqrt(TempOtherUserLength);  
  
 // ||x|| \* ||y||  
 double CrossProduct = UserLength \* TempOtherUserLength;  
  
 // Getting cosine similarity for current indexed user & setting it  
 double Cosine\_Similarity = TempDotProduct / CrossProduct;  
 Ratings[User][0] = Cosine\_Similarity;  
 }  
 }  
  
 // Predicting scores for unknown values  
 public static void **Predictions** () {  
 for (int i = 1; i < Ratings[UserIndex].length; i++) {  
 if (Ratings[UserIndex][i] == 0) {  
 // Resetting values for each new empty rating  
 SimilaritySum = 0;  
 SimilarityDotProduct = 0;  
  
 // Get dot product of all other users with similarity score & rating in respective column  
 // Also getting Sum of similarities  
 for (int j = 0; j < Ratings.length; j++) {  
 if (j != UserIndex || Ratings[j][i] != 0) {  
 SimilaritySum += Ratings[j][0];  
 SimilarityDotProduct += Ratings[j][0] \* Ratings[j][i];  
 }  
 }  
 Ratings[UserIndex][i] = SimilarityDotProduct / SimilaritySum;  
 }  
 }  
 }  
  
 public static void **Selection\_Sort** (double[] arr) {  
 for (int i = 0; i < arr.length - 1; i++)  
 {  
 int index = i;  
 for (int j = i + 1; j < arr.length; j++){  
 if (arr[j] > arr[index]){  
 index = j;//searching for greatest index  
 }  
 }  
 // Swapping temporary array  
 double biggerNumber = arr[index];  
 arr[index] = arr[i];  
 arr[i] = biggerNumber;  
  
 // Swapping Products  
 BackendDataStorage tempProduct = AllProducts.get(index);  
 AllProducts.set(index, AllProducts.get(i));  
 AllProducts.set(i, tempProduct);  
 }  
 }  
  
 //For testing purposes only  
 public static void **Print2DArray** (double[][] array) {  
 for (int i = 0; i < array.length; i++) { //this equals to the row in our 2D array.  
 for (int j = 0; j < array[i].length; j++) { //this equals to the column in each row.  
 System.out.print(array[i][j] + " ");  
 }  
 System.out.println(); //change line on console as row comes to end in the matrix.  
 }  
 }  
  
 public static double[][] AllZeros (double[][] array) {  
 for (int i = 0; i < array.length; i++) {  
 for (int j = 0; j < array[i].length; j++) {  
 array[i][j] = 0;  
 }  
 }  
 return array;  
 }  
}

**File Name:** MainActivity.java

**Purpose:** Backend that lets the user decide whether to register or login.

––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––

| package com.example.usshop;  import androidx.appcompat.app.AppCompatActivity;  import android.content.Intent; import android.os.Bundle; import android.util.Log; import android.view.View;  import com.google.firebase.auth.FirebaseAuth;  public class **MainActivity** extends **AppCompatActivity** {   public static final FirebaseAuth fAuth = FirebaseAuth.getInstance();   @Override  protected void **onCreate**(Bundle savedInstanceState) {  super.onCreate(savedInstanceState);  setContentView(R.layout.activity\_main);  setTitle("Setup");  // Load User if they had logged in before  if (fAuth.getCurrentUser() != null) {  if (fAuth.getCurrentUser().getEmail().equalsIgnoreCase("special.user@gmail.com")) {  startActivity(new Intent(getApplicationContext(), Backend.class));  finish();  return;  }  startActivity(new Intent(getApplicationContext(), Shop.class));  finish();  }  }   public void **LoadRegister** (View view) {  Intent i = new Intent(this, Register.class);  startActivity(i);  }   public void **LoadLogin** (View view) {  Intent i = new Intent(this, Login.class);  startActivity(i);  } } |
| --- |
|  |

**File Name:** activity\_main.xml

**Purpose:** Frontend of the screen that lets the user decide whether to register or login.

––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––

<?xml version="1.0" encoding="utf-8"?>  
<androidx.constraintlayout.widget.ConstraintLayout 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"  
 tools:context=".MainActivity">  
  
 <Button  
 android:id="@+id/button"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginStart="216dp"  
 android:layout\_marginTop="252dp"  
 android:onClick="LoadRegister"  
 android:text="Register"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent" />  
  
 <Button  
 android:id="@+id/button2"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginStart="56dp"  
 android:layout\_marginTop="252dp"  
 android:onClick="LoadLogin"  
 android:text="Login"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent" />  
</androidx.constraintlayout.widget.ConstraintLayout>

**File Name:** Register.java

**Purpose:** Backend that creates an account of the users inputted details in FA.

––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––

package com.example.usshop;  
  
import androidx.annotation.NonNull;  
import androidx.appcompat.app.AppCompatActivity;  
  
import android.content.Intent;  
import android.os.Bundle;  
import android.view.View;  
import android.widget.EditText;  
import android.widget.Toast;  
  
import com.google.android.gms.tasks.OnCompleteListener;  
import com.google.android.gms.tasks.Task;  
import com.google.firebase.auth.AuthResult;  
  
public class **Register** extends **AppCompatActivity** {  
 String FirstName, LastName;  
 EditText Password, PIN;  
  
 @Override  
 protected void **onCreate**(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.activity\_register);  
  
 Password = findViewById(R.id.EnteredPassword);  
 PIN = findViewById(R.id.PIN);  
 }  
  
 public void **CreateAccount** (View view) {  
 // If not in U's location, do not create account  
 String PINInput = PIN.getText().toString();  
  
 if (!PINInput.equalsIgnoreCase("12345")) {  
 findViewById(R.id.Warning).setVisibility(View.VISIBLE);  
 return;  
 }  
  
 FirstName = ((EditText)findViewById(R.id.EnteredFirstName)).getText().toString().trim();  
 LastName = ((EditText)findViewById(R.id.EnterLastName)).getText().toString().trim();  
  
 // Ensure '.' or '@' is not used anywhere in first or last name  
 if (FirstName.contains(".") || LastName.contains(".") || FirstName.contains("@") || LastName.contains("@")) {  
 Toast.makeText(Register.this, "The first nor the last name can contain '.' or '@'", Toast.LENGTH\_SHORT).show();  
 return;  
 }  
  
 String email = FirstName + "." + LastName + "@gmail.com";  
 String password = Password.getText().toString().trim();  
  
 // Creating Account & checking if it was successful  
 MainActivity.fAuth.createUserWithEmailAndPassword(email, password).addOnCompleteListener(new OnCompleteListener<AuthResult>() {  
 @Override  
 public void **onComplete**(@NonNull Task<AuthResult> task) {  
 if (task.isSuccessful()) {  
 Toast.makeText(Register.this, "Account Created!", Toast.LENGTH\_SHORT).show();  
 startActivity(new Intent(getApplicationContext(), Shop.class));  
 }  
 else {  
 Toast.makeText(Register.this, "Error: " + task.getException().getMessage(), Toast.LENGTH\_SHORT).show();  
 }  
 }  
 });  
 }  
}

**File Name:** activity\_register.xml

**Purpose:** Frontend of the screen that creates an account of the users inputted details in FA.

––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––

<?xml version="1.0" encoding="utf-8"?>  
<androidx.constraintlayout.widget.ConstraintLayout 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"  
 tools:context=".Register">  
  
 <EditText  
 android:id="@+id/PIN"  
 android:layout\_width="307dp"  
 android:layout\_height="48dp"  
 android:ems="10"  
 android:inputType="textPersonName"  
 android:text="Enter PIN"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent"  
 app:layout\_constraintVertical\_bias="0.374" />  
  
 <EditText  
 android:id="@+id/EnteredPassword"  
 android:layout\_width="307dp"  
 android:layout\_height="48dp"  
 android:ems="10"  
 android:inputType="textPersonName"  
 android:text="Create Password"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent"  
 app:layout\_constraintVertical\_bias="0.281" />  
  
 <EditText  
 android:id="@+id/EnteredFirstName"  
 android:layout\_width="307dp"  
 android:layout\_height="48dp"  
 android:ems="10"  
 android:inputType="textPersonName"  
 android:text="Enter First Name"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent"  
 app:layout\_constraintVertical\_bias="0.103" />  
  
 <EditText  
 android:id="@+id/EnterLastName"  
 android:layout\_width="307dp"  
 android:layout\_height="48dp"  
 android:ems="10"  
 android:inputType="textPersonName"  
 android:text="Enter Last Name"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent"  
 app:layout\_constraintVertical\_bias="0.193" />  
  
 <Button  
 android:id="@+id/AccountCreatorButton"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginStart="88dp"  
 android:layout\_marginTop="340dp"  
 android:onClick="CreateAccount"  
 android:text="Create Account"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent" />  
  
 <TextView  
 android:id="@+id/Note"  
 android:layout\_width="286dp"  
 android:layout\_height="57dp"  
 android:layout\_marginStart="60dp"  
 android:layout\_marginTop="404dp"  
 android:text="Note: Please type your first name, last name, and PIN truthfully. Otherwise, the services can not be provided to you."  
 android:textColor="#FF0000"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent" />  
  
 <TextView  
 android:id="@+id/Warning"  
 android:layout\_width="272dp"  
 android:layout\_height="25dp"  
 android:layout\_marginStart="68dp"  
 android:layout\_marginTop="312dp"  
 android:text="Services are not provided in your area!"  
 android:textColor="#FF0000"  
 android:visibility="invisible"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent"  
 tools:visibility="visible" />  
</androidx.constraintlayout.widget.ConstraintLayout>

**File Name:** Login.java

**Purpose:** Backend that verifies whether the inputted account details exist and send to the “Shop” screen accordingly.

––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––

package com.example.usshop;  
  
import androidx.annotation.NonNull;  
import androidx.appcompat.app.AppCompatActivity;  
  
import android.content.Intent;  
import android.os.Bundle;  
import android.view.View;  
import android.widget.EditText;  
import android.widget.Toast;  
  
import com.google.android.gms.tasks.OnCompleteListener;  
import com.google.android.gms.tasks.Task;  
import com.google.firebase.auth.AuthResult;  
  
public class **Login** extends **AppCompatActivity** {  
 String FirstName, LastName, Email, Password;  
  
 @Override  
 protected void **onCreate**(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.activity\_login);  
 }  
  
 public void **LoggingIn** (View view) {  
 FirstName = ((EditText)findViewById(R.id.EnteredFirstName)).getText().toString().trim();  
 LastName = ((EditText)findViewById(R.id.EnteredLastName)).getText().toString().trim();  
  
 Email = FirstName + "." + LastName + "@gmail.com";  
 Password = ((EditText)findViewById(R.id.EnteredPassword)).getText().toString().trim();  
  
 // Signing In  
 // Note: U's Account Details; Email: special.user@gmail.com. Password: 1a9+7\*  
 // Note: Test User Account Details; Email: random.user@gmail.com. Password: ranuser  
 MainActivity.fAuth.signInWithEmailAndPassword(Email, Password).addOnCompleteListener(new OnCompleteListener<AuthResult>() {  
 @Override  
 public void **onComplete**(@NonNull Task<AuthResult> task) {  
 if (task.isSuccessful()) {  
 // Checking if user is U. Else loads screen for the Shop  
 if (Email.equalsIgnoreCase("special.user@gmail.com") && Password.equals("1a9+7\*")) {  
 Toast.makeText(Login.this, "Special Account Login Successful!", Toast.LENGTH\_SHORT).show();  
 startActivity(new Intent(getApplicationContext(), Backend.class));  
 return;  
 }  
  
 Toast.makeText(Login.this, "Login Successful!", Toast.LENGTH\_SHORT).show();  
 startActivity(new Intent(getApplicationContext(), Shop.class));  
 }  
 else {  
 Toast.makeText(Login.this, "Error: " + task.getException().getMessage(), Toast.LENGTH\_SHORT).show();  
 }  
 }  
 });  
 }  
}

**File Name:** activity\_login.xml

**Purpose:** Frontend of the screen that verifies whether the inputted account details exist and send to the “Shop” screen accordingly.

––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––

<?xml version="1.0" encoding="utf-8"?>  
<androidx.constraintlayout.widget.ConstraintLayout 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"  
 tools:context=".Register">  
  
 <EditText  
 android:id="@+id/EnteredPassword"  
 android:layout\_width="307dp"  
 android:layout\_height="48dp"  
 android:ems="10"  
 android:inputType="textPersonName"  
 android:text="Enter Password"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent"  
 app:layout\_constraintVertical\_bias="0.368" />  
  
 <EditText  
 android:id="@+id/EnteredFirstName"  
 android:layout\_width="307dp"  
 android:layout\_height="48dp"  
 android:ems="10"  
 android:inputType="textPersonName"  
 android:text="Enter First Name"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent"  
 app:layout\_constraintVertical\_bias="0.196" />  
  
 <EditText  
 android:id="@+id/EnteredLastName"  
 android:layout\_width="307dp"  
 android:layout\_height="48dp"  
 android:ems="10"  
 android:inputType="textPersonName"  
 android:text="Enter Last Name"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent"  
 app:layout\_constraintVertical\_bias="0.284" />  
  
 <Button  
 android:id="@+id/Login"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginStart="132dp"  
 android:layout\_marginTop="340dp"  
 android:onClick="LoggingIn"  
 android:text="Login"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent" />  
  
</androidx.constraintlayout.widget.ConstraintLayout>

**File Name:** Backend.java

**Purpose:** Backend that allows U to create products with the corresponding inputted fields

––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––

package com.example.usshop;  
  
import androidx.activity.result.ActivityResult;  
import androidx.activity.result.ActivityResultCallback;  
import androidx.activity.result.ActivityResultLauncher;  
import androidx.activity.result.contract.ActivityResultContracts;  
import androidx.annotation.NonNull;  
import androidx.appcompat.app.AppCompatActivity;  
  
import android.Manifest;  
import android.app.Activity;  
import android.content.ContentResolver;  
import android.content.Intent;  
import android.content.pm.PackageManager;  
import android.database.Cursor;  
import android.graphics.Bitmap;  
import android.graphics.BitmapFactory;  
import android.graphics.Picture;  
import android.icu.text.CaseMap;  
import android.net.Uri;  
import android.os.Build;  
import android.os.Bundle;  
import android.provider.MediaStore;  
import android.util.Log;  
import android.view.View;  
import android.webkit.MimeTypeMap;  
import android.widget.Button;  
import android.widget.EditText;  
import android.widget.ImageView;  
import android.widget.Toast;  
  
import com.google.android.gms.tasks.Continuation;  
import com.google.android.gms.tasks.OnCompleteListener;  
import com.google.android.gms.tasks.OnFailureListener;  
import com.google.android.gms.tasks.OnSuccessListener;  
import com.google.android.gms.tasks.Task;  
import com.google.firebase.auth.FirebaseAuth;  
import com.google.firebase.database.DatabaseReference;  
import com.google.firebase.database.FirebaseDatabase;  
import com.google.firebase.storage.FirebaseStorage;  
import com.google.firebase.storage.OnProgressListener;  
import com.google.firebase.storage.StorageReference;  
import com.google.firebase.storage.UploadTask;  
  
public class **Backend** extends **AppCompatActivity** {  
 private static final int PERMISSION\_REQUEST = 0;  
 private static final int RESULT\_LOAD\_IMAGE = 1;  
  
 ImageView imageView;  
 Button button;  
  
 ActivityResultLauncher<Intent> ActivityResultLauncher;  
 Bitmap BF;  
  
 Uri SelectedImage;  
 StorageReference StorageRef;  
 DatabaseReference DatabaseRef;  
  
 FirebaseDatabase rootNode;  
 DatabaseReference reference;  
  
 @Override  
 protected void **onCreate**(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.activity\_backend);  
  
 imageView = (ImageView) findViewById(R.id.Image);  
 button = (Button) findViewById(R.id.AddImage);  
  
 StorageRef = FirebaseStorage.getInstance().getReference("Items");  
 DatabaseRef = FirebaseDatabase.getInstance().getReference("Items");  
  
 ActivityResultLauncher = registerForActivityResult(  
 new ActivityResultContracts.StartActivityForResult(),  
 result -> {  
 if (result.getResultCode() == Activity.RESULT\_OK) {  
 // Stores Image Data  
 SelectedImage = result.getData().getData();  
  
 // Stores rough location of Image (like in Camera, WhatsApp Images, etc) in Index 0 of String array  
 String[] FilePathColumn = {MediaStore.Images.Media.DATA};  
  
 // Stores contents and location of image  
 Cursor cursor = getContentResolver().query(SelectedImage, FilePathColumn, null, null, null);  
 cursor.moveToFirst();  
  
 // Store Picture's exact Path  
 int ColumnIndex = cursor.getColumnIndex(FilePathColumn[0]);  
 String PicturePath = cursor.getString(ColumnIndex);  
 cursor.close();  
  
 // Setting image  
 BF = BitmapFactory.decodeFile(PicturePath);  
 ((ImageView)findViewById(R.id.Image)).setImageBitmap(BF);  
 }  
 });  
 }  
  
 public void **InsertImage** (View view) {  
 // Request permission to use user's media if not already asked before.  
 if (Build.VERSION.SDK\_INT >= Build.VERSION\_CODES.M && checkSelfPermission(Manifest.permission.READ\_EXTERNAL\_STORAGE) != PackageManager.PERMISSION\_GRANTED) {  
 requestPermissions(new String[] {Manifest.permission.READ\_EXTERNAL\_STORAGE}, PERMISSION\_REQUEST);  
 }  
  
 Intent i = new Intent(Intent.ACTION\_PICK, MediaStore.Images.Media.EXTERNAL\_CONTENT\_URI);  
  
 // Run method inside ActivityResultLauncher  
 ActivityResultLauncher.launch(i);  
 }  
  
 public void **SpecialLogout** (View view) {  
 FirebaseAuth.getInstance().signOut(); //logout  
 startActivity(new Intent(getApplicationContext(), Login.class));  
 finish();  
 }  
  
 // Sends item to database  
 public void **SendData** (View view) {  
 // Getting & converting values  
 String title = ((EditText)findViewById(R.id.Title)).getText().toString().trim();  
 String description = ((EditText)findViewById(R.id.Description)).getText().toString().trim();  
 String priceStr = ((EditText)findViewById(R.id.Price)).getText().toString();  
  
 // If image not selected, do not continue  
 if (BF == null || title.isEmpty() || description.isEmpty() || priceStr.isEmpty()) {  
 Toast.makeText(Backend.this, "Must include image, title, description, and price!", Toast.LENGTH\_LONG).show();  
 return;  
}  
 int price = Integer.parseInt(priceStr);

// Top node of the database  
 rootNode = FirebaseDatabase.getInstance();  
 reference = rootNode.getReference("Items");  
  
 // Storing image on Firebase Cloud  
 StorageReference fileReference = StorageRef.child(title  
 + "." + getFileExtension(SelectedImage));  
  
 // Putting image in Firebase Storage & Firebase Realtime Database  
 fileReference.putFile(SelectedImage).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()) {  
 Toast.makeText(Backend.this, "File Uploaded!", Toast.LENGTH\_SHORT).show();  
 BackendDataStorage BDS = new BackendDataStorage(title, description, price, task.getResult().toString());  
 DatabaseRef.child(title).setValue(BDS);  
 }  
 else {  
 Toast.makeText(Backend.this, "Task failed: " + task.getException().getMessage(), Toast.LENGTH\_LONG).show();  
 }  
 }  
 });  
 }  
  
 public String **getFileExtension** (Uri uri) {  
 ContentResolver CR = getContentResolver();  
 MimeTypeMap mime = MimeTypeMap.getSingleton();  
 return mime.getExtensionFromMimeType(CR.getType(uri));  
 }  
  
 // To add functionality to remove products, follow following video: https://www.youtube.com/watch?v=68vp1O7Tukg.  
 //  
}

**File Name:** activity\_backend.xml

**Purpose:** Frontend of the screen that allows U to create products with the corresponding inputted fields

––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––

<?xml version="1.0" encoding="utf-8"?>  
<androidx.constraintlayout.widget.ConstraintLayout 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"  
 tools:context=".Backend">  
  
 <Button  
 android:id="@+id/SpecialLogout"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginStart="264dp"  
 android:layout\_marginTop="0dp"  
 android:onClick="SpecialLogout"  
 android:text="Logout"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent" />  
  
 <EditText  
 android:id="@+id/Description"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginStart="4dp"  
 android:layout\_marginTop="300dp"  
 android:ems="10"  
 android:inputType="textPersonName"  
 android:minHeight="48dp"  
 android:text="Insert Description"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent" />  
  
 <EditText  
 android:id="@+id/Title"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginStart="4dp"  
 android:layout\_marginTop="236dp"  
 android:ems="10"  
 android:inputType="textPersonName"  
 android:minHeight="48dp"  
 android:text="Insert Title"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent" />  
  
 <TextView  
 android:id="@+id/PriceText"  
 android:layout\_width="185dp"  
 android:layout\_height="30dp"  
 android:layout\_marginStart="4dp"  
 android:layout\_marginTop="368dp"  
 android:text="Price:"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent" />  
  
 <EditText  
 android:id="@+id/Price"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginStart="16dp"  
 android:layout\_marginTop="388dp"  
 android:ems="10"  
 android:inputType="number"  
 android:minHeight="48dp"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent"  
 tools:text="Insert Price" />  
  
 <TextView  
 android:id="@+id/Ruppee"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginStart="4dp"  
 android:layout\_marginTop="392dp"  
 android:text="₹"  
 android:textSize="25dp"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent" />  
  
 <Button  
 android:id="@+id/Send"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginStart="132dp"  
 android:layout\_marginTop="464dp"  
 android:onClick="SendData"  
 android:text="Sell Product"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent" />  
  
 <ImageView  
 android:id="@+id/Image"  
 android:layout\_width="405dp"  
 android:layout\_height="150dp"  
 android:layout\_marginStart="4dp"  
 android:layout\_marginTop="64dp"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent" />  
  
 <Button  
 android:id="@+id/AddImage"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginStart="40dp"  
 android:layout\_marginTop="4dp"  
 android:onClick="InsertImage"  
 android:text="Insert Image"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent" />  
  
</androidx.constraintlayout.widget.ConstraintLayout>

**File Name:** Shop.java

**Purpose:** Backend that uses MyAdapter.java and BackendDataStorage.java to display all of the products, and provide the functionality to search for products

––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––

package com.example.usshop;  
  
import androidx.annotation.NonNull;  
import androidx.appcompat.app.AppCompatActivity;  
import androidx.recyclerview.widget.LinearLayoutManager;  
import androidx.recyclerview.widget.RecyclerView;  
  
import android.content.Intent;  
import android.graphics.Bitmap;  
import android.os.Bundle;  
import android.util.Log;  
import android.view.Menu;  
import android.view.MenuItem;  
import android.view.View;  
import android.view.inputmethod.EditorInfo;  
import android.widget.ArrayAdapter;  
import android.widget.ProgressBar;  
import android.widget.SearchView;  
  
import com.google.firebase.auth.FirebaseAuth;  
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.LinkedList;  
import java.util.List;  
  
public class **Shop** extends **AppCompatActivity** {  
 List<BackendDataStorage> TempData = new ArrayList<>();  
  
 RecyclerView rv;  
  
 DatabaseReference ItemsReference = FirebaseDatabase.getInstance().getReference("Items");  
  
 MyAdapter myAdapter;  
  
 ProgressBar progressBar;  
  
 public static Shop shop;  
  
 @Override  
 protected void **onCreate**(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.activity\_shop);  
 shop = this;  
  
 progressBar = findViewById(R.id.progressBar);  
  
 //Whenever the value of ItemsReference changes, it is stored  
 ItemsReference.addValueEventListener(new ValueEventListener() {  
 @Override  
 public void **onDataChange**(@NonNull DataSnapshot snapshot) {  
 // Storing all of the data  
 for (DataSnapshot DS : snapshot.getChildren()) {  
 TempData.add(DS.getValue(BackendDataStorage.class));  
 }  
 // Run recommendation algorithm to predict users preferences &  
 // display predicted preferences  
 Recommendation\_Algorithm RA = new Recommendation\_Algorithm(TempData);  
 }  
  
 @Override  
 public void **onCancelled**(@NonNull DatabaseError error) {  
 Log.d("Error", "Failed to read value: " + error.getMessage());  
 progressBar.setVisibility(View.INVISIBLE);  
 }  
 });  
 }  
  
 public void **CreateItems** (List<BackendDataStorage> TempData) {  
 rv = findViewById(R.id.Recycler\_View);  
  
 // Create & put items on screen  
 myAdapter = new MyAdapter(this, TempData);  
 rv.setAdapter(myAdapter);  
 rv.setLayoutManager(new LinearLayoutManager(this));  
 progressBar.setVisibility(View.INVISIBLE);  
 }  
  
 public void **Logout** (View view) {  
 FirebaseAuth.getInstance().signOut(); //logout  
 startActivity(new Intent(getApplicationContext(), Login.class));  
 finish();  
 }  
  
 // Navbar stuff  
 @Override  
 public boolean **onCreateOptionsMenu**(Menu menu) {  
 // Adding NavMenu to Shop  
 getMenuInflater().inflate(R.menu.nav\_menu, menu);  
  
 // Specifying details of Navbar  
 MenuItem menuItem = menu.findItem(R.id.nav\_search);  
 SearchView searchView = (SearchView) menuItem.getActionView();  
 searchView.setQueryHint("Search Product");  
 searchView.setImeOptions(EditorInfo.IME\_ACTION\_DONE);  
  
 // Filtering items if somewhere inside the item's text is the same  
 searchView.setOnQueryTextListener(new SearchView.OnQueryTextListener() {  
 @Override  
 public boolean **onQueryTextSubmit**(String s) {  
 return false;  
 }  
  
 @Override  
 public boolean **onQueryTextChange**(String s) {  
 myAdapter.getFilter().filter(s);  
 return false;  
 }  
 });  
  
 return true;  
 }  
}

**File Name:** activity\_shop.xml

**Purpose:** Frontend of the screen that uses MyAdapter.java and BackendDataStorage.java to display all of the products, and provide the functionality to search for products

––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––

<?xml version="1.0" encoding="utf-8"?>  
<androidx.constraintlayout.widget.ConstraintLayout 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:id="@+id/loadPanel"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 tools:context=".Shop">  
  
 <Button  
 android:id="@+id/Logout"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginStart="264dp"  
 android:layout\_marginTop="0dp"  
 android:onClick="Logout"  
 android:text="Logout"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent" />  
  
 <androidx.recyclerview.widget.RecyclerView  
 android:id="@+id/Recycler\_View"  
 android:layout\_width="409dp"  
 android:layout\_height="681dp"  
 android:layout\_marginTop="48dp"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent" />  
  
 <ProgressBar  
 android:id="@+id/progressBar"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginStart="152dp"  
 android:layout\_marginTop="300dp"  
 android:foregroundGravity="center"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent" />  
  
</androidx.constraintlayout.widget.ConstraintLayout>

**File Name:** MyAdapter.java

**Purpose:** Backend focusing on displaying all products and making the search bar function with the appropriate search algorithm

––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––

package com.example.usshop;  
  
import android.annotation.SuppressLint;  
import android.content.Context;  
import android.content.Intent;  
import android.graphics.Bitmap;  
import android.graphics.BitmapFactory;  
import android.os.AsyncTask;  
import android.os.BaseBundle;  
import android.os.Bundle;  
import android.util.Log;  
import android.view.LayoutInflater;  
import android.view.View;  
import android.view.ViewGroup;  
import android.widget.Filter;  
import android.widget.Filterable;  
import android.widget.ImageView;  
import android.widget.TextView;  
  
import androidx.annotation.NonNull;  
import androidx.constraintlayout.widget.ConstraintLayout;  
import androidx.recyclerview.widget.RecyclerView;  
  
import com.google.firebase.analytics.FirebaseAnalytics;  
import com.squareup.picasso.Picasso;  
  
import java.io.IOException;  
import java.io.InputStream;  
import java.net.MalformedURLException;  
import java.net.URL;  
import java.util.ArrayList;  
import java.util.LinkedList;  
import java.util.List;  
import java.util.Locale;  
  
public class **MyAdapter** extends **RecyclerView**.**Adapter**<**MyAdapter**.**MyViewHolder**> implements **Filterable** {  
 private List<BackendDataStorage> Data = new ArrayList<>();  
 private List<BackendDataStorage> FullData;  
  
 private Context context;  
  
 private FirebaseAnalytics mFirebaseAnalytics;  
  
 public **MyAdapter** (Context ct, List<BackendDataStorage> Data) {  
 context = ct;  
 mFirebaseAnalytics = FirebaseAnalytics.getInstance(ct);  
  
 this.Data = Data;  
 FullData = new ArrayList<>(Data);  
 }  
  
 // Creates a ViewHolder with all of the content  
 @NonNull  
 @Override  
 public MyViewHolder **onCreateViewHolder**(@NonNull ViewGroup parent, int viewType) {  
 LayoutInflater inflater = LayoutInflater.from(context);  
 View view = inflater.inflate(R.layout.my\_row, parent,false);  
 return new MyViewHolder(view);  
 }  
  
 // For each item in ViewHolder, run the following method  
 @Override  
 public void **onBindViewHolder**(@NonNull MyViewHolder holder, @SuppressLint("RecyclerView") int position) {  
 holder.text1.setText(Data.get(position).getTitle());  
 holder.text2.setText(Data.get(position).getDescription());  
 holder.text3.setText("₹" + Data.get(position).getPrice());  
  
 //Setting image  
 Picasso.get()  
 .load(Data.get(position).getPictureUrl())  
 .placeholder(R.mipmap.ic\_launcher)  
 .fit()  
 .centerCrop()  
 .into(holder.image);  
  
 holder.cl.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void **onClick**(View view) {  
 //Sending clicked product to ProductDisplay  
 Intent i = new Intent(context, ProductDisplay.class);  
 i.putExtra("data1", Data.get(position).getTitle());  
 i.putExtra("data2", Data.get(position).getDescription());  
 i.putExtra("price", String.valueOf(Data.get(position).getPrice()));  
 i.putExtra("image", Data.get(position).getPictureUrl());  
 context.startActivity(i);  
 }  
 });  
 }  
  
 @Override  
 public int **getItemCount**() {  
 return Data.size();  
 }  
  
 public class **MyViewHolder** extends **RecyclerView**.**ViewHolder** {  
 TextView text1, text2, text3;  
 ImageView image;  
  
 ConstraintLayout cl;  
  
 public **MyViewHolder** (@NonNull View viewItem) {  
 super(viewItem);  
 text1 = viewItem.findViewById(R.id.Title\_txt);  
 text2 = viewItem.findViewById(R.id.Description\_txt);  
 text3 = viewItem.findViewById(R.id.PriceDisplay);  
 image = viewItem.findViewById(R.id.TempImage);  
 cl = viewItem.findViewById(R.id.mainLayout);  
 }  
 }  
  
 // Filtering Logic Below  
 @Override  
 public Filter **getFilter**() {  
 return ExampleFilter;  
 }  
  
 private Filter ExampleFilter = new Filter() {  
 @Override  
 // When typed "MyAdapterVariable.getFilter().filter("some string");", the string runs & filters here  
 protected FilterResults **performFiltering**(CharSequence charSequence) {  
 List<BackendDataStorage> FilteredResults = new ArrayList<>();  
  
 //If not filtered, include everything  
 if (charSequence == null || charSequence.length() == 0) {  
 FilteredResults.addAll(FullData);  
 }  
 else {  
 //Ignoring capitalization, if the searched sequence of strings are identical in the  
 //product & searched string, then keep displaying  
 String FilterPattern = charSequence.toString().toLowerCase().trim();  
  
 for (BackendDataStorage item : FullData) {  
 String PatternbeingFilledto = item.getTitle().toLowerCase() + " " + item.getDescription().toLowerCase() + " " + String.valueOf(item.getPrice());  
 if (PatternbeingFilledto.contains(FilterPattern)) {  
 FilteredResults.add(item);  
 }  
 }  
 }  
  
 FilterResults results = new FilterResults();  
 results.values = FilteredResults;  
  
 return results;  
 }  
 // Other better search algorithms such as Full-text search (https://blog.sebastian-daschner.com/entries/full-text-search-neoj4-java) could be implemented  
  
 //When done button is clicked, all filtered items are now displayed  
 @Override  
 protected void **publishResults**(CharSequence charSequence, FilterResults filterResults) {  
 Data.clear();  
 Data.addAll((List) filterResults.values);  
 notifyDataSetChanged();  
 }  
 };  
}

**File Name:** my\_row.xml

**Purpose:** A temporary frontend display of where the details of each product should be positioned for MyAdapter.java to use to display products appropriately.

––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––

<?xml version="1.0" encoding="utf-8"?>  
<androidx.constraintlayout.widget.ConstraintLayout 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="wrap\_content"  
 android:layout\_margin="0dp"  
 android:id="@+id/mainLayout">  
  
 <androidx.cardview.widget.CardView  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 app:layout\_constraintBottom\_toBottomOf="parent">  
  
 </androidx.cardview.widget.CardView>  
  
 <androidx.constraintlayout.widget.ConstraintLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 tools:layout\_editor\_absoluteX="0dp"  
 tools:layout\_editor\_absoluteY="45dp">  
  
 <TextView  
 android:id="@+id/PriceDisplay"  
 android:layout\_width="110dp"  
 android:layout\_height="18dp"  
 android:layout\_marginStart="8dp"  
 android:layout\_marginTop="4dp"  
 android:text="Insert Price"  
 app:layout\_constraintStart\_toEndOf="@+id/TempImage"  
 app:layout\_constraintTop\_toBottomOf="@+id/Description\_txt" />  
  
 <TextView  
 android:id="@+id/Description\_txt"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginStart="8dp"  
 android:layout\_marginTop="4dp"  
 android:text="Insert Description"  
 app:layout\_constraintStart\_toEndOf="@+id/TempImage"  
 app:layout\_constraintTop\_toBottomOf="@+id/Title\_txt" />  
  
 <ImageView  
 android:id="@+id/TempImage"  
 android:layout\_width="102dp"  
 android:layout\_height="99dp"  
 android:layout\_marginEnd="300dp"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent"  
 tools:srcCompat="@tools:sample/avatars" />  
  
 <TextView  
 android:id="@+id/Title\_txt"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginStart="8dp"  
 android:text="Insert Title"  
 android:textSize="20sp"  
 android:textStyle="bold"  
 app:layout\_constraintBottom\_toBottomOf="@+id/TempImage"  
 app:layout\_constraintStart\_toEndOf="@+id/TempImage"  
 app:layout\_constraintTop\_toTopOf="parent"  
 app:layout\_constraintVertical\_bias="0.222" />  
 </androidx.constraintlayout.widget.ConstraintLayout>  
  
</androidx.constraintlayout.widget.ConstraintLayout>

**File Name:** BackendDataStorage.java

**Purpose:** Class that is used as an object to store and allow accessibility to the details of each products details

––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––

package com.example.usshop;  
  
public class **BackendDataStorage** {  
 private String Title, Description;  
 private int Price;  
 private String PictureUrl;  
  
 public **BackendDataStorage** () {  
  
 }  
  
 public **BackendDataStorage**(String title, String description, int price, String pictureUrl) {  
 Title = title;  
 Description = description;  
 Price = price;  
 PictureUrl = pictureUrl;  
 }  
  
 public String **getTitle**() {  
 return Title;  
 }  
  
 public String **getDescription**() {  
 return Description;  
 }  
  
 public int **getPrice**() {  
 return Price;  
 }  
  
 public String **getPictureUrl**() {  
 return PictureUrl;  
 }  
  
 public void **setTitle**(String title) {  
 Title = title;  
 }  
  
 public void **setDescription**(String description) {  
 Description = description;  
 }  
  
 public void **setPrice**(int price) {  
 Price = price;  
 }  
  
 public void **setPictureUrl**(String pictureUrl) {  
 PictureUrl = pictureUrl;  
 }  
}

**File Name:** ProductDisplay.java

**Purpose:** Backend displaying the product selected in the Shop. It also sends non-0 ratings to FRD and runs JavaMailAPI.java when a product is purchased

––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––

package com.example.usshop;  
  
import androidx.activity.OnBackPressedCallback;  
import androidx.appcompat.app.ActionBar;  
import androidx.appcompat.app.AppCompatActivity;  
  
import android.app.ProgressDialog;  
import android.content.Intent;  
import android.graphics.Bitmap;  
import android.net.Uri;  
import android.os.Bundle;  
import android.util.Log;  
import android.view.View;  
import android.widget.Button;  
import android.widget.ImageView;  
import android.widget.ProgressBar;  
import android.widget.RatingBar;  
import android.widget.TextView;  
import android.widget.Toast;  
  
import com.google.firebase.auth.FirebaseAuth;  
import com.google.firebase.database.DatabaseReference;  
import com.google.firebase.database.FirebaseDatabase;  
import com.squareup.picasso.Picasso;  
  
import java.util.Calendar;  
import java.util.Date;  
  
public class **ProductDisplay** extends **AppCompatActivity** {  
 ImageView mainImageView;  
 TextView Title, Description, PriceText;  
 private RatingBar RB;  
  
 private boolean PurchasedBefore = false;  
 String data1, data2, Price;  
 String imageURL;  
  
 DatabaseReference DatabaseRef;  
  
 @Override  
 protected void **onCreate**(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.activity\_product\_display);  
  
 RB = (RatingBar) findViewById(R.id.rating);  
 DatabaseRef = FirebaseDatabase.getInstance().getReference("Recommendation\_Algorithm");  
  
 mainImageView = findViewById(R.id.IV);  
 Title = findViewById(R.id.DisplayTitle);  
 Description = findViewById(R.id.DisplayDescription);  
 PriceText = findViewById(R.id.Cash);  
 getData();  
 setData();  
 }  
  
 private void **getData** () {  
 if (getIntent().hasExtra("image") && getIntent().hasExtra("data1")  
 && getIntent().hasExtra("data2") && getIntent().hasExtra("price")) {  
 data1 = getIntent().getStringExtra("data1");  
 data2 = getIntent().getStringExtra("data2");  
 Price = getIntent().getStringExtra("price");  
 imageURL = getIntent().getStringExtra("image");  
 }  
 else {  
 Toast.makeText(this, "Insufficient Data!", Toast.LENGTH\_LONG).show();  
 }  
 }  
  
 private void **setData** () {  
 Title.setText(data1);  
 Description.setText(data2);  
 PriceText.setText("₹" + Price);  
  
 //Setting image  
 Picasso.get()  
 .load(imageURL)  
 .placeholder(R.mipmap.ic\_launcher)  
 .fit()  
 .centerCrop()  
 .into(mainImageView);  
 }  
  
 //Runs if "Buy" button is clicked  
 public void **Purchased** (View view) {  
 //Getting Users First & Last Name  
 String NameMail = MainActivity.fAuth.getCurrentUser().getEmail(), FirstName = "", LastName = "";  
  
 // Getting First & Last Name from the users mail  
 boolean ReachedDot = false, ReachedAtSymbol = false;  
 Date CurrentTime = Calendar.getInstance().getTime();  
 for (int MailIndex = 0; MailIndex < NameMail.length(); MailIndex++) {  
 if (!ReachedDot) {  
 if (NameMail.charAt(MailIndex) != '.') {  
 FirstName += NameMail.charAt(MailIndex);  
 }  
 else {  
 ReachedDot = true;  
 }  
 }  
 else if (!ReachedAtSymbol) {  
 if (NameMail.charAt(MailIndex) != '@') {  
 LastName += NameMail.charAt(MailIndex);  
 }  
 else {  
 ReachedAtSymbol = true;  
 }  
 }  
 }  
  
 // Sending Mail  
 String SendMailto = "arnav.kumarltp@gmail.com", Subject = "<<BUSINESS>> " + FirstName + " bought " + Title.getText().toString(),  
 Message = FirstName + " " + LastName + " on " + CurrentTime.toString() + " bought " + Title.getText().toString() + " for ₹" + Price;  
  
 JavaMailAPI JMAPI = new JavaMailAPI(this, SendMailto, Subject, Message);  
 JMAPI.execute();  
 }  
  
 public void **Send\_Rating** (View view) {  
 if (RB.getRating() != 0) {  
 // Add rating to firebase  
 String UserUID = FirebaseAuth.getInstance().getCurrentUser().getUid();  
 DatabaseRef.child(UserUID).child(data1).setValue(RB.getRating());  
 Toast.makeText(this, "Rating Sent!", Toast.LENGTH\_SHORT).show();  
 }  
 else {  
 Toast.makeText(this, "Rating cannot be 0", Toast.LENGTH\_SHORT).show();  
 }  
 }  
}

**File Name:** activity\_product\_display.xml

**Purpose:** Frontend of the screen that displays the product selected in the Shop with a rating bar and a “Buy” and “Send Rating” button.

––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––

<?xml version="1.0" encoding="utf-8"?>  
<androidx.constraintlayout.widget.ConstraintLayout 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"  
 tools:context=".ProductDisplay">  
  
 <ImageView  
 android:id="@+id/IV"  
 android:layout\_width="172dp"  
 android:layout\_height="160dp"  
 android:layout\_marginTop="16dp"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent"  
 tools:srcCompat="@tools:sample/avatars" />  
  
 <TextView  
 android:id="@+id/DisplayTitle"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginTop="28dp"  
 android:text="Title"  
 android:textSize="32dp"  
 android:textStyle="bold"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toBottomOf="@+id/IV" />  
  
 <TextView  
 android:id="@+id/DisplayDescription"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginTop="28dp"  
 android:text="Description"  
 android:textSize="20dp"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintHorizontal\_bias="0.498"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toBottomOf="@+id/DisplayTitle" />  
  
 <TextView  
 android:id="@+id/Cash"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginStart="168dp"  
 android:layout\_marginBottom="8dp"  
 android:text="Insert Price"  
 app:layout\_constraintBottom\_toTopOf="@+id/Purchase"  
 app:layout\_constraintStart\_toStartOf="parent" />  
  
 <Button  
 android:id="@+id/Purchase"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginStart="148dp"  
 android:layout\_marginBottom="28dp"  
 android:onClick="Purchased"  
 android:text="Buy"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent" />  
  
 <RatingBar  
 android:id="@+id/rating"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginStart="60dp"  
 android:layout\_marginTop="356dp"  
 android:contentDescription="Rating"  
 android:visibility="visible"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent" />  
  
 <Button  
 android:id="@+id/SendRating"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginStart="120dp"  
 android:layout\_marginTop="416dp"  
 android:onClick="Send\_Rating"  
 android:text="Send Rating"  
 android:visibility="visible"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent" />  
  
</androidx.constraintlayout.widget.ConstraintLayout>

**File Name:** JavaMailAPI.java

**Purpose:** Backend which sends U an email informing him of a purchase using Utils.java

––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––

package com.example.usshop;  
import android.app.ProgressDialog;  
import android.content.Context;  
import android.os.AsyncTask;  
import android.widget.Toast;  
import java.util.Properties;  
import javax.mail.Message;  
import javax.mail.MessagingException;  
import javax.mail.PasswordAuthentication;  
import javax.mail.Session;  
import javax.mail.Transport;  
import javax.mail.internet.InternetAddress;  
import javax.mail.internet.MimeMessage;  
  
public class **JavaMailAPI** extends **AsyncTask**<**Void**,**Void**,**Void**> {  
  
 //Variables  
 private Context mContext;  
 private Session mSession;  
  
 private String mEmail;  
 private String mSubject;  
 private String mMessage;  
  
 private ProgressDialog mProgressDialog;  
  
 //Constructor  
 public **JavaMailAPI**(Context mContext, String mEmail, String mSubject, String mMessage) {  
 this.mContext = mContext;  
 this.mEmail = mEmail;  
 this.mSubject = mSubject;  
 this.mMessage = mMessage;  
 }  
  
 @Override  
 protected void **onPreExecute**() {  
 super.onPreExecute();  
 //Show progress dialog while sending email  
 mProgressDialog = ProgressDialog.show(mContext,"Confirming Product Delivery", "Please wait...",false,false);  
 }  
  
 @Override  
 protected void **onPostExecute**(Void aVoid) {  
 super.onPostExecute(aVoid);  
 //Dismiss progress dialog when message successfully send  
 mProgressDialog.dismiss();  
  
 //Show success toast  
 Toast.makeText(mContext,"Product will be delivered soon!",Toast.LENGTH\_SHORT).show();  
 }  
  
 @Override  
 protected Void **doInBackground**(Void... params) {  
 //Creating properties  
 Properties props = new Properties();  
  
 //Configuring properties for gmail  
 props.put("mail.smtp.host", "smtp.gmail.com");  
 props.put("mail.smtp.socketFactory.port", "465");  
 props.put("mail.smtp.socketFactory.class", "javax.net.ssl.SSLSocketFactory");  
 props.put("mail.smtp.auth", "true");  
 props.put("mail.smtp.port", "465");  
  
 //Creating a new session  
 mSession = Session.getDefaultInstance(props,  
 new javax.mail.Authenticator() {  
 //Authenticating the password  
 protected PasswordAuthentication **getPasswordAuthentication**() {  
 return new PasswordAuthentication(Utils.EMAIL, Utils.PASSWORD);  
 }  
 });  
  
 try {  
 //Creating MimeMessage object  
 MimeMessage mm = new MimeMessage(mSession);  
  
 //Setting sender address  
 mm.setFrom(new InternetAddress(Utils.EMAIL));  
 //Adding receiver  
 mm.addRecipient(Message.RecipientType.TO, new InternetAddress(mEmail));  
 //Adding subject  
 mm.setSubject(mSubject);  
 //Adding message  
 mm.setText(mMessage);  
 //Sending email  
 Transport.send(mm);  
 } catch (MessagingException e) {  
 Toast.makeText(mContext,"Product can not be delivered",Toast.LENGTH\_SHORT).show();  
 }  
 return null;  
 }  
}

**File Name:** Utils.java

**Purpose:** Backend containing the email and password of a temporary account which is needed to simulate sending an email to U.

––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––––

package com.example.usshop;  
  
public class **Utils** {  
 public static final String EMAIL = "testing31.email4314@gmail.com", PASSWORD = "zoikswfipvoemmrn";  
}