

NAME :- prashant kumar mishra

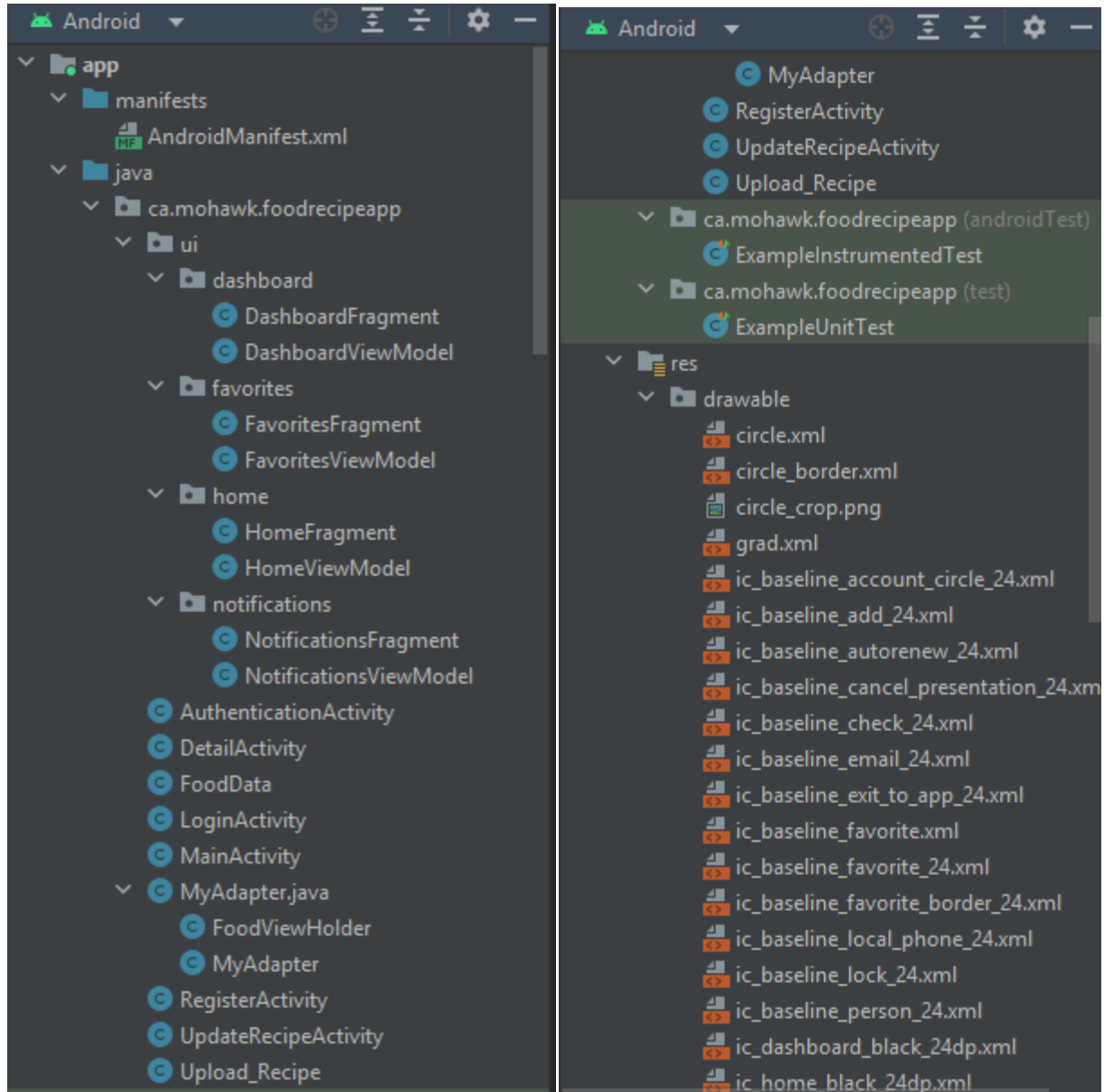
Course:-bsc(hons)computer science 4B

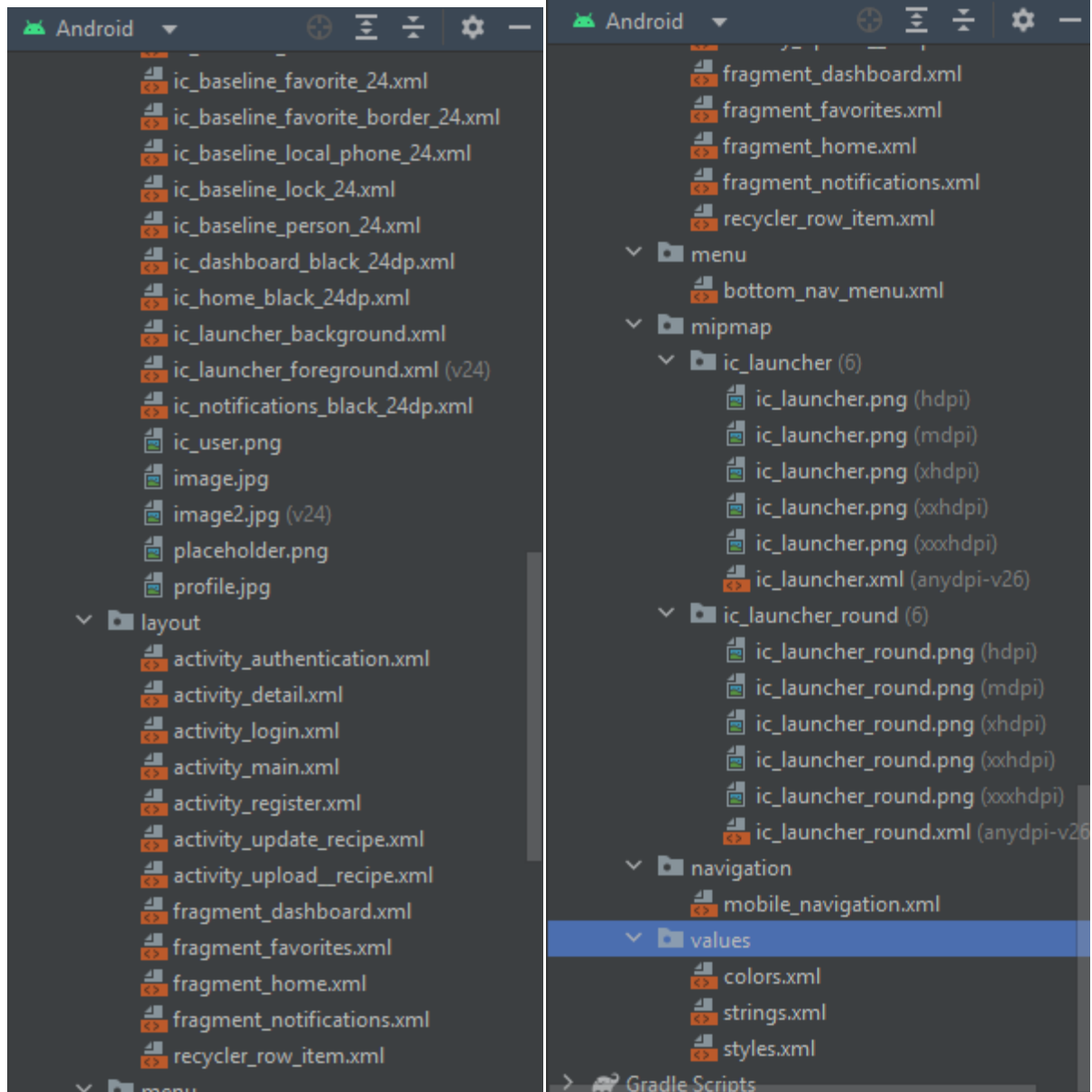
Roll no:- 10894

Subjet :- sec android development assignment

The app is based on recipe make rand recipe finder ,especially it includes support for Indian origin cusians and dishes

The file structure of the app is as follows





Codes :-

App/manifest/androidmanifest.xml

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="ca.mohawk.foodrecipeapp">

    <uses-permission android:name="android.permission.READ_EXTERNAL_STORAGE"
/>
    <uses-permission android:name="android.permission.INTERNET"/>

    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportsRtl="true"
        android:theme="@style/AppTheme">
        <activity android:name=".LoginActivity">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
        <activity android:name=".RegisterActivity" >

        </activity>
        <activity android:name=".AuthenticationActivity" />
        <activity android:name=".UpdateRecipeActivity" />
        <activity android:name=".Upload_Recipe" />
        <activity android:name=".DetailActivity" />
        <activity android:name=".MainActivity">

        </activity>
    </application>
</manifest>
```

Ui/dashboard/dashboardfragment

```
package ca.mohawk.foodrecipeapp.ui.dashboard;

import android.app.ProgressDialog;
import android.content.Intent;
import android.os.Bundle;
import android.text.Editable;
```

```

import android.text.TextWatcher;
import android.util.Log;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.view.WindowManager;
import android.widget.EditText;
import android.widget.TextView;

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.firestore.DocumentReference;
import com.google.firebase.firestore.DocumentSnapshot;
import com.google.firebase.firestore.EventListener;
import com.google.firebase.firestore.FirebaseFirestore;
import com.google.firebase.firestore.FirebaseFirestoreException;
import com.google.firebase.firestore.QueryDocumentSnapshot;
import com.google.firebase.firestore.QuerySnapshot;
import com.google.firebase.firestore.model.DocumentCollections;

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

import androidx.annotation.NonNull;
import androidx.annotation.Nullable;
import androidx.fragment.app.Fragment;
import androidx.lifecycle.Observer;
import androidx.lifecycle.ViewModelProviders;
import androidx.recyclerview.widget.GridLayoutManager;
import androidx.recyclerview.widget.RecyclerView;
import ca.mohawk.foodrecipeapp.FoodData;
import ca.mohawk.foodrecipeapp.LoginActivity;
import ca.mohawk.foodrecipeapp.MyAdapter;
import ca.mohawk.foodrecipeapp.R;

public class DashboardFragment extends Fragment {

    private DashboardViewModel dashboardViewModel;
    FirebaseAuth firebaseAuth;
    FirebaseFirestore firebaseFirestore;
    String userID;
    RecyclerView mRecyclerView;
    TextView txt_no_result_found;
    EditText txt_search;
    ProgressDialog progressDialog;
    List<FoodData> myFoodList;
    MyAdapter myAdapter;

    public View onCreateView(@NonNull LayoutInflater inflater,
                             ViewGroup container, Bundle savedInstanceState)
    {
        dashboardViewModel =
            ViewModelProviders.of(this).get(DashboardViewModel.class);
        View root = inflater.inflate(R.layout.fragment_dashboard, container,
            false);
    }

```

```

getActivity().getWindow().setSoftInputMode(WindowManager.LayoutParams.SOFT_INPUT_STATE_ALWAYS_HIDDEN);

        mRecyclerView = root.findViewById(R.id.recyclerView);
        GridLayoutManager gridLayoutManager = new
GridLayoutManager(root.getContext(), 1);
        mRecyclerView.setLayoutManager(gridLayoutManager);
        txt_no_result_found = root.findViewById(R.id.txt_no_result_found);
        txt_search = root.findViewById(R.id.txt_searchtext);
        progressDialog = new ProgressDialog(root.getContext());
        progressDialog.setMessage("Loading Recipes...");

        firebaseAuth = FirebaseAuth.getInstance();
        firebaseFirestore = FirebaseFirestore.getInstance();
        if (firebaseAuth.getCurrentUser() == null){
            Intent intent = new Intent(getContext(), LoginActivity.class);
            String warning = "You need to Login First to unlock this
feature!!";
            intent.putExtra("warning", warning);
            startActivity(intent);
        } else {
            userID = firebaseAuth.getCurrentUser().getUid();

            myFoodList = new ArrayList<FoodData>();

            myAdapter = new MyAdapter(root.getContext(), myFoodList);
            mRecyclerView.setAdapter(myAdapter);

            progressDialog.show();

            final DocumentReference documentReference =
firebaseFirestore.collection("users").document(userID);

documentReference.collection("Recipes").get().addOnSuccessListener(new
OnSuccessListener<QuerySnapshot>() {
    @Override
    public void onSuccess(QuerySnapshot queryDocumentSnapshots) {
        myFoodList.clear();
        for (QueryDocumentSnapshot documentSnapshot :
queryDocumentSnapshots) {
            FoodData foodData =
documentSnapshot.toObject(FoodData.class);
            foodData.setKey(foodData.getKey());
            foodData.setRecipeKey(foodData.getRecipeKey());
            myFoodList.add(foodData);
        }
        myAdapter.notifyDataSetChanged();
        progressDialog.dismiss();
    }
}).addOnFailureListener(new OnFailureListener() {
    @Override
    public void onFailure(@NonNull Exception e) {
        progressDialog.dismiss();
    }
}

```

```

        });

        txt_search.addTextChangedListener(new TextWatcher() {
            @Override
            public void beforeTextChanged(CharSequence charSequence, int
i, int i1, int i2) {

            }

            @Override
            public void onTextChanged(CharSequence charSequence, int i,
int i1, int i2) {

            }

            @Override
            public void afterTextChanged(Editable editable) {
                filter(editable.toString());
            }

        });
    }
    return root;
}

private void filter(String text) {
    ArrayList<FoodData> filterList = new ArrayList<>();

    for (FoodData item: myFoodList){
        if(item.getItemName().toLowerCase().contains(text.toLowerCase())){
            filterList.add(item);
        }
    }

    myAdapter.filteredList(filterList);
}
}

```

Ui/dashboard/dashboardviewmodel

```

package ca.mohawk.foodrecipeapp.ui.dashboard;

import androidx.lifecycle.LiveData;
import androidx.lifecycle.MutableLiveData;
import androidx.lifecycle.ViewModel;

public class DashboardViewModel extends ViewModel {

    private MutableLiveData<String> mText;
}

```



```
public DashboardViewModel() {  
    mText = new MutableLiveData<>();  
    mText.setValue("This is dashboard fragment");  
}  
  
public LiveData<String> getText() {  
    return mText;  
}  
}
```

favoritesfragment

```
package ca.mohawk.foodrecipeapp.ui.favorites;  
  
import android.app.ProgressDialog;  
import android.content.Intent;  
import android.os.Bundle;  
import android.text.Editable;  
import android.text.TextWatcher;  
import android.util.Log;  
import android.view.LayoutInflater;  
import android.view.View;  
import android.view.ViewGroup;  
import android.view.WindowManager;  
import android.widget.EditText;  
import android.widget.TextView;  
  
import com.google.android.gms.tasks.OnFailureListener;  
import com.google.android.gms.tasks.OnSuccessListener;  
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 com.google.firebase.firestore.DocumentReference;  
import com.google.firebase.firestore.FirebaseFirestore;  
import com.google.firebase.firestore.QueryDocumentSnapshot;  
import com.google.firebase.firestore.QuerySnapshot;  
  
import java.util.ArrayList;  
import java.util.List;  
  
import androidx.annotation.NonNull;  
import androidx.fragment.app.Fragment;  
import androidx.lifecycle.ViewModelProviders;  
import androidx.recyclerview.widget.GridLayoutManager;  
import androidx.recyclerview.widget.RecyclerView;  
import ca.mohawk.foodrecipeapp.FoodData;  
import ca.mohawk.foodrecipeapp.LoginActivity;  
import ca.mohawk.foodrecipeapp.MyAdapter;  
import ca.mohawk.foodrecipeapp.R;
```

```

import ca.mohawk.foodrecipeapp.ui.home.HomeViewModel;

public class FavoritesFragment extends Fragment {

    private FavoritesViewModel favoritesViewModel;
    RecyclerView mRecyclerView;
    List<FoodData> myFoodList;
    FoodData mFoodData;
    TextView txt_empty;

    private DatabaseReference databaseReference;
    private ValueEventListener eventListener;
    ProgressDialog progressDialog;
    EditText txt_search;
    MyAdapter myAdapter;
    String userId;
    String TAG = "tag";
    FirebaseAuth firebaseAuth;
    FirebaseFirestore firebaseFirestore;

    public View onCreateView(@NonNull LayoutInflater inflater,
                             ViewGroup container, Bundle savedInstanceState)
    {
        favoritesViewModel =
        ViewModelProviders.of(this).get(FavoritesViewModel.class);
        View root = inflater.inflate(R.layout.fragment_favorites, container,
        false);

        getActivity().getWindow().setSoftInputMode(WindowManager.LayoutParams.SOFT_IN
        PUT_STATE_ALWAYS_HIDDEN);

        mRecyclerView = root.findViewById(R.id.recyclerView);
        txt_empty = root.findViewById(R.id.txt_empty);
        GridLayoutManager gridLayoutManager = new
        GridLayoutManager(root.getContext(), 1);
        mRecyclerView.setLayoutManager(gridLayoutManager);
        txt_search = root.findViewById(R.id.txt_searchtext);

        firebaseAuth = FirebaseAuth.getInstance();
        firebaseFirestore = FirebaseFirestore.getInstance();
        if (firebaseAuth.getCurrentUser() == null){
            Intent intent = new Intent(getContext(), LoginActivity.class);
            String warning = "You need to Login First to unlock this
feature!!";
            intent.putExtra("warning", warning);
            startActivity(intent);
        } else {
            userId = firebaseAuth.getCurrentUser().getUid();

            progressDialog = new ProgressDialog(root.getContext());
            progressDialog.setMessage("Loading Recipes...");

            myFoodList = new ArrayList<FoodData>();

            progressDialog.show();

```

```

        final DocumentReference documentReference =
firebaseFirestore.collection("users").document(userId);

documentReference.collection("Favorites").get().addOnSuccessListener(new
OnSuccessListener<QuerySnapshot>() {
    @Override
    public void onSuccess(QuerySnapshot queryDocumentSnapshots) {
        myFoodList.clear();
        for (QueryDocumentSnapshot documentSnapshot :
queryDocumentSnapshots) {
            FoodData foodData =
documentSnapshot.toObject(FoodData.class);
            foodData.setKey(foodData.getKey());
            foodData.setRecipeKey(documentSnapshot.getId());
            myFoodList.add(foodData);
        }
        myAdapter.notifyDataSetChanged();
        progressDialog.dismiss();
    }
}).addOnFailureListener(new OnFailureListener() {
    @Override
    public void onFailure(@NonNull Exception e) {
        progressDialog.dismiss();
    }
});

txt_search.addTextChangedListener(new TextWatcher() {
    @Override
    public void beforeTextChanged(CharSequence charSequence, int
i, int i1, int i2) {

    }

    @Override
    public void onTextChanged(CharSequence charSequence, int i,
int i1, int i2) {

    }

    @Override
    public void afterTextChanged(Editable editable) {
        filter(editable.toString());
    }
});

myAdapter = new MyAdapter(root.getContext(), myFoodList);
mRecyclerView.setAdapter(myAdapter);
}
return root;
}

private void filter(String text) {
    ArrayList<FoodData> filterList = new ArrayList<>();

    for (FoodData item: myFoodList){

```

```

if (item.getItemName().toLowerCase().contains(text.toLowerCase())) {
    filterList.add(item);
}

myAdapter.filteredList(filterList);
}
}

```

faviouretesviewmodel

```

package ca.mohawk.foodrecipeapp.ui.favorites;

import androidx.lifecycle.LiveData;
import androidx.lifecycle.MutableLiveData;
import androidx.lifecycle.ViewModel;

public class FavoritesViewModel extends ViewModel {

    private MutableLiveData<String> mText;

    public FavoritesViewModel() {
        mText = new MutableLiveData<>();
        mText.setValue("This is Favorites fragment");
    }

    public LiveData<String> getText() {
        return mText;
    }
}

```

homefragment

```

package ca.mohawk.foodrecipeapp.ui.home;

import android.app.ProgressDialog;
import android.content.Intent;
import android.os.Bundle;
import android.text.Editable;
import android.text.TextWatcher;
import android.util.Log;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.view.WindowManager;
import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;

```

```

import android.widget.Toast;

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 com.google.firebase.firestore.FirebaseFirestore;

import java.util.ArrayList;
import java.util.List;
import androidx.annotation.NonNull;
import androidx.fragment.app.Fragment;
import androidx.lifecycle.ViewModelProviders;
import androidx.recyclerview.widget.GridLayoutManager;
import androidx.recyclerview.widget.RecyclerView;
import ca.mohawk.foodrecipeapp.FoodData;
import ca.mohawk.foodrecipeapp.MyAdapter;
import ca.mohawk.foodrecipeapp.R;

public class HomeFragment extends Fragment {

    private HomeViewModel homeViewModel;
    RecyclerView mRecyclerView;
    List<FoodData> myFoodList;
    FoodData mFoodData;
    TextView txt_no_results_found;
    private DatabaseReference databaseReference;
    private ValueEventListener eventListener;
    ProgressDialog progressDialog;
    EditText txt_search;
    MyAdapter myAdapter;
    String userId;
    String TAG = "tag";
    FirebaseAuth firebaseAuth;
    FirebaseFirestore firebaseFirestore;

    public View onCreateView(@NonNull LayoutInflater inflater,
                             ViewGroup container, Bundle savedInstanceState)
    {
        homeViewModel = ViewModelProviders.of(this).get(HomeViewModel.class);
        View root = inflater.inflate(R.layout.fragment_home, container,
false);

        getActivity().getWindow().setSoftInputMode(WindowManager.LayoutParams.SOFT_IN
PUT_STATE_ALWAYS_HIDDEN);

        mRecyclerView = root.findViewById(R.id.recyclerView);
        GridLayoutManager gridLayoutManager = new
GridLayoutManager(root.getContext(), 1);
        mRecyclerView.setLayoutManager(gridLayoutManager);
        txt_search = root.findViewById(R.id.txt_searchtext);
        txt_no_results_found = root.findViewById(R.id.txt_no_results_found);
        firebaseAuth = FirebaseAuth.getInstance();

```

```

        firebaseFirestore = FirebaseFirestore.getInstance();
        //        userId = firebaseAuth.getCurrentUser().getUid();

        progressDialog = new ProgressDialog(root.getContext());
        progressDialog.setMessage("Loading Recipes...");

        myFoodList = new ArrayList<FoodData>();

        myAdapter = new MyAdapter(root.getContext(), myFoodList);
        mRecyclerView.setAdapter(myAdapter);

        databaseReference =
        FirebaseDatabase.getInstance().getReference("Recipe");
        progressDialog.show();
        eventListener = databaseReference.addValueEventListener(new
        ValueEventListener() {
            @Override
            public void onDataChange(@NonNull DataSnapshot snapshot) {
                myFoodList.clear();
                for (DataSnapshot itemSnapshot : snapshot.getChildren()) {
                    FoodData foodData =
                    itemSnapshot.getValue(FoodData.class);
                    foodData.setKey(itemSnapshot.getKey());
                    myFoodList.add(foodData);
                }

                myAdapter.notifyDataSetChanged();
                progressDialog.dismiss();
            }

            @Override
            public void onCancelled(@NonNull DatabaseError error) {
                progressDialog.dismiss();
            }
        });

        txt_search.addTextChangedListener(new TextWatcher() {
            @Override
            public void beforeTextChanged(CharSequence charSequence, int i,
            int i1, int i2) {
                }

            @Override
            public void onTextChanged(CharSequence charSequence, int i, int
            i1, int i2) {
                }

            @Override
            public void afterTextChanged(Editable editable) {
                filter(editable.toString());
            }
        });
        return root;
    }

```

```

        private void filter(String text) {
            ArrayList<FoodData> filterList = new ArrayList<>();

            for (FoodData item: myFoodList){
                if(item.getItemName().toLowerCase().contains(text.toLowerCase())){
                    filterList.add(item);
                }
            }

            myAdapter.filteredList(filterList);
        }
    }
}

```

homeviewmodel

```

package ca.mohawk.foodrecipeapp.ui.home;

import androidx.lifecycle.LiveData;
import androidx.lifecycle.MutableLiveData;
import androidx.lifecycle.ViewModel;

public class HomeViewModel extends ViewModel {

    private MutableLiveData<String> mText;

    public HomeViewModel() {
        mText = new MutableLiveData<>();
        mText.setValue("This is home fragment");
    }

    public LiveData<String> getText() {
        return mText;
    }
}

```

notificationsfragment

```

package ca.mohawk.foodrecipeapp.ui.notifications;

import android.content.Intent;
import android.os.Bundle;
import android.util.Log;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.ImageView;
import android.widget.LinearLayout;
import android.widget.TextView;
import android.widget.Toast;

import com.google.firebase.auth.FirebaseAuth;
import com.google.firebase.auth.FirebaseUser;
import com.google.firebase.firestore.DocumentReference;
import com.google.firebase.firestore.DocumentSnapshot;
import com.google.firebase.firestore.EventListener;
import com.google.firebase.firestore.FirebaseFirestore;
import com.google.firebase.firestore.FirebaseFirestoreException;

import java.util.concurrent.Executor;

import androidx.annotation.NonNull;
import androidx.annotation.Nullable;
import androidx.fragment.app.Fragment;
import androidx.lifecycle.Observer;
import androidx.lifecycle.ViewModelProviders;
import ca.mohawk.foodrecipeapp.LoginActivity;
import ca.mohawk.foodrecipeapp.R;

public class NotificationsFragment extends Fragment {

    private NotificationsViewModel notificationsViewModel;
    TextView txt_name,txt_email,txt_phone, verifyMsg, txt_verify,
txt_verified;
    ImageView photo_imageView, verify_imageview, verified_imageview;
    LinearLayout btnVerifyEmail;
    FirebaseAuth firebaseAuth;
    FirebaseFirestore firebaseFirestore;
    String userID;

    public View onCreateView(@NonNull LayoutInflater inflater, ViewGroup
container, Bundle savedInstanceState) {
        notificationsViewModel =
ViewModelProviders.of(this).get(NotificationsViewModel.class);
        View root = inflater.inflate(R.layout.fragment_notifications,
container, false);

        txt_name = root.findViewById(R.id.txt_name);
        txt_email = root.findViewById(R.id.txt_email);
        txt_phone = root.findViewById(R.id.txt_phone);
        photo_imageView = root.findViewById(R.id.photo_imageView);
        verify_imageview = root.findViewById(R.id.verify_imageview);
        verified_imageview = root.findViewById(R.id.verified_imageview);
        txt_verify = root.findViewById(R.id.txt_verify);
        txt_verified = root.findViewById(R.id.txt_verified);

```



```

        btnVerifyEmail = root.findViewById(R.id.btnVerifyEmail);

        firebaseAuth = FirebaseAuth.getInstance();
        firebaseFirestore = FirebaseFirestore.getInstance();
        if (firebaseAuth.getCurrentUser() == null) {
            Intent intent = new Intent(getContext(), LoginActivity.class);
            String warning = "You need to Login First to unlock this
feature!!";
            intent.putExtra("warning", warning);
            startActivity(intent);
        } else {
            userID = firebaseAuth.getCurrentUser().getUid();
            final FirebaseUser firebaseUser = firebaseAuth.getCurrentUser();

            if (!firebaseUser.isEmailVerified()) {
                verified_imageview.setVisibility(View.GONE);
                txt_verified.setVisibility(View.GONE);

                verify_imageview.setVisibility(View.VISIBLE);
                txt_verify.setVisibility(View.VISIBLE);
            } else {
                verify_imageview.setVisibility(View.GONE);
                txt_verify.setVisibility(View.GONE);

                verified_imageview.setVisibility(View.VISIBLE);
                txt_verified.setVisibility(View.VISIBLE);
                btnVerifyEmail.setOnClickListener(null);
            }

            DatabaseReference documentReference =
firebaseFirestore.collection("users").document(userID);
            documentReference.addSnapshotListener(getActivity(), new
EventListener<DocumentSnapshot>() {
                @Override
                public void onEvent(@Nullable DocumentSnapshot value,
@Nullable FirebaseFirestoreException error) {
                    if (error != null) {
                        Log.d("TAG", "onEvent:" + error.getMessage());
                    } else {
                        txt_name.setText(value.getString("fullName"));
                        txt_email.setText(value.getString("email"));
                        txt_phone.setText(value.getString("phone"));
                    }
                }
            });
        }
        return root;
    }
}

```

notificationsviewmodel

```
package ca.mohawk.foodrecipeapp.ui.notifications;

import androidx.lifecycle.LiveData;
import androidx.lifecycle.MutableLiveData;
import androidx.lifecycle.ViewModel;

public class NotificationsViewModel extends ViewModel {

    private MutableLiveData<String> mText;

    public NotificationsViewModel() {
        mText = new MutableLiveData<>();
        mText.setValue("This is notifications fragment");
    }

    public LiveData<String> getText() {
        return mText;
    }
}
```

authenticationactivity

```
package ca.mohawk.foodrecipeapp;

import androidx.appcompat.app.AppCompatActivity;
import android.os.Bundle;

public class AuthenticationActivity extends AppCompatActivity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_authentication);
    }
}
```

detailactivity

```
package ca.mohawk.foodrecipeapp;

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

import android.content.Intent;
import android.os.Bundle;
import android.util.Log;
import android.view.View;
import android.widget.Button;
import android.widget.ImageView;
import android.widget.TextView;
import android.widget.Toast;

import com.bumptech.glide.Glide;
import com.google.android.gms.tasks.OnFailureListener;
import com.google.android.gms.tasks.OnSuccessListener;
import com.google.android.material.snackbar.BaseTransientBottomBar;
import com.google.android.material.snackbar.Snackbar;
import com.google.firebase.auth.FirebaseAuth;
import com.google.firebase.database.DatabaseReference;
import com.google.firebase.database.FirebaseDatabase;
import com.google.firebase.firestore.CollectionReference;
import com.google.firebase.firestore.DocumentReference;
import com.google.firebase.firestore.DocumentSnapshot;
import com.google.firebase.firestore.EventListener;
import com.google.firebase.firestore.FirebaseFirestore;
import com.google.firebase.firestore.FirebaseFirestoreException;
import com.google.firebase.firestore.QueryDocumentSnapshot;
import com.google.firebase.firestore.QuerySnapshot;
import com.google.firebase.storage.FirebaseStorage;
import com.google.firebase.storage.StorageReference;

public class DetailActivity extends AppCompatActivity {

    TextView foodDescription, recipeName, recipePrice;
    ImageView foodImage, like_imageView, dislike_imageView;
    Button btnUpdate, btnDelete;
    String recipeItemName, recipeItemPrice, recipeDescription;
    String key = "";
    String compareKey = "";
    String recipeKey = "";
    String imageUrl = "";
    String favoriteKey = "";
    String userID;
    FirebaseAuth firebaseAuth;
    FirebaseFirestore firebaseFirestore;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_detail);

        recipeName = findViewById(R.id.txtRecipeName);
        recipePrice = findViewById(R.id.txtPrice);
        foodDescription = findViewById(R.id.txtDescription);
```

```

        foodImage = findViewById(R.id.ivImage2);
        btnUpdate = findViewById(R.id.btnUpdate);
        btnDelete = findViewById(R.id.btnDelete);
        like_imageView = findViewById(R.id.like_imageView);
        dislike_imagView = findViewById(R.id.dislike_imagView);

        firebaseAuth = FirebaseAuth.getInstance();
        firebaseFirestore = FirebaseFirestore.getInstance();
        if (firebaseAuth.getCurrentUser() == null){
            finish();
            Intent intent = new Intent(this, LoginActivity.class);
            String warning = "You need to Login First to unlock this
feature!!";
            intent.putExtra("warning", warning);
            startActivity(intent);
        } else {
            userID = firebaseAuth.getCurrentUser().getUid();

            Bundle mBundle = getIntent().getExtras();

            if (mBundle != null) {
                recipeItemName = mBundle.getString("RecipeName");
                recipeName.setText(recipeItemName);
                recipeItemPrice = mBundle.getString("price");
                recipePrice.setText(recipeItemPrice);
                recipeDescription = mBundle.getString("Description");
                foodDescription.setText(recipeDescription);
                key = mBundle.getString("keyValue");
                recipeKey = mBundle.getString("recipeKeyValue");
                imageUrl = mBundle.getString("Image");

                Glide.with(this).load(mBundle.getString("Image")).into(foodImage);
            }

            Log.d("TAG DetailActivity", "Recipe Key " + recipeKey);

            final DocumentReference favoriteDocumentReference =
firebaseFirestore.collection("users").document(userID);

favoriteDocumentReference.collection("Favorites").get().addOnSuccessListener(
new OnSuccessListener<QuerySnapshot>() {
    @Override
    public void onSuccess(QuerySnapshot queryDocumentSnapshots) {
        for (QueryDocumentSnapshot documentSnapshot :
queryDocumentSnapshots) {
            FoodData foodData =
documentSnapshot.toObject(FoodData.class);
            if (key.equals(foodData.getKey())) {
                like_imageView.setVisibility(View.VISIBLE);
                dislike_imagView.setVisibility(View.GONE);
                favoriteKey = documentSnapshot.getId();
                Log.d("TAG", "favoriteKey : " + favoriteKey);
            }
        }
    }
}
}

```

```

    });

    if (recipeKey != null) {
        DocumentReference documentReference =
firebaseFirestore.collection("users").document(userID).collection("Recipes").
document(recipeKey);
        documentReference.addSnapshotListener(this, new
EventListener<DocumentSnapshot>() {
            @Override
            public void onEvent(@Nullable DocumentSnapshot value,
@Nullable FirebaseFirestoreException error) {
                if (error != null) {
                    Log.d("TAG", "onEvent:" + error.getMessage());
                } else {
                    compareKey = value.getString("key");
                    if (key.equals(compareKey)) {
                        btnUpdate.setVisibility(View.VISIBLE);
                        btnDelete.setVisibility(View.VISIBLE);
                    }
                }
            }
        });
    }
}

}

}

public void btnDeleteRecipe(View view) {
    final DatabaseReference reference =
FirebaseDatabase.getInstance().getReference("Recipe");
    FirebaseStorage storage = FirebaseStorage.getInstance();
    StorageReference storageReference =
storage.getReferenceFromUrl(imageUrl);

    storageReference.delete().addOnSuccessListener(new
OnSuccessListener<Void>() {
        @Override
        public void onSuccess(Void aVoid) {
            reference.child(key).removeValue();
        }
    });

    DocumentReference documentReference =
firebaseFirestore.collection("users").document(userID).collection("Recipes").
document(recipeKey);
    documentReference.delete()
        .addOnSuccessListener(new OnSuccessListener<Void>() {
            @Override
            public void onSuccess(Void aVoid) {
                Log.d("TAG", "Recipe deleted from Firestore");
                Snackbar.
make(findViewById(R.id.txtRecipeName), "Recipe
Deleted", BaseTransientBottomBar.LENGTH_SHORT).show();
                startActivity(new
Intent(getApplicationContext(), MainActivity.class));
                finish();
            }
        }).addOnFailureListener(new OnFailureListener() {

```

```

        @Override
        public void onFailure(@NonNull Exception e) {
            Log.d("TAG", " Recipe not deleted from Firestore " +
e.getMessage());
            Snackbar.make(findViewById(R.id.txtRecipeName), "Failed to
Delete Recipe" + e.getMessage(), BaseTransientBottomBar.LENGTH_SHORT).show();
        }
    });
}

    public void btnUpdateRecipe(View view) {

        startActivity(new
Intent(getApplicationContext(), UpdateRecipeActivity.class)
            .putExtra("recipeNameKey", recipeName.getText().toString())

.putExtra("descriptionKey", foodDescription.getText().toString())
            .putExtra("priceKey", recipePrice.getText().toString())
            .putExtra("oldImageUrl", imageUrl)
            .putExtra("key", key)
            .putExtra("recipeKey", recipeKey));
    }

    public void onBackPressed() {
        Intent intent=new Intent(this, MainActivity.class);
        startActivity(intent);
        finish();
    }

    public void btnLike(View view) {
        // Delete Saved Recipe

        DocumentReference documentReference =
firebaseFirestore.collection("users").document(userID).collection("Favorites"
).document(favoriteKey);
        documentReference.delete()
            .addOnSuccessListener(new OnSuccessListener() {
                @Override
                public void onSuccess(Void aVoid) {
                    Log.d("TAG", " Recipe deleted from Firestore");
                }
            }).addOnFailureListener(new OnFailureListener() {
                @Override
                public void onFailure(@NonNull Exception e) {
                    Log.d("TAG", " Recipe not deleted from Firestore " +
e.getMessage());
                    Snackbar.make(findViewById(R.id.txtRecipeName), "Failed to
Delete Recipe" + e.getMessage(), BaseTransientBottomBar.LENGTH_SHORT).show();
                }
            });

        dislike_imagView.setVisibility(View.VISIBLE);
        like_imageView.setVisibility(View.GONE);
    }

    public void btnDislike(final View view) {
        // Add to Save Recipe

```

```

        favoriteKey =
firebaseFirestore.collection("users").document(userID).collection("Favorites"
).document().getId();
        FoodData fireStoreFoodData = new
FoodData(recipeItemName,recipeDescription,recipeItemPrice,imageUrl,key,favori
teKey);

        DocumentReference documentReference =
firebaseFirestore.collection("users").document(userID).collection("Favorites"
).document(favoriteKey);
        documentReference.set(fireStoreFoodData).addOnSuccessListener(new
OnSuccessListener<Void>() {
            @Override
            public void onSuccess(Void aVoid) {
                Log.d("TAG", " Recipe added to Firestore :");
            }
        }).addOnFailureListener(new OnFailureListener() {
            @Override
            public void onFailure(@NonNull Exception e) {
                Log.d("TAG", " Recipe not added to Firestore " +
e.getMessage());
                Snackbar.make(view,"Failed to Upload Recipe" +
e.getMessage(), BaseTransientBottomBar.LENGTH_SHORT).show();
            }
        });

        like_imageView.setVisibility(View.VISIBLE);
        dislike_imagView.setVisibility(View.GONE);
    }
}

```

fooddata

```

package ca.mohawk.foodrecipeapp;

public class FoodData {

    private String itemName;
    private String itemDescription;
    private String itemPrice;
    private String itemImage;
    private String key;
    private String recipeKey;

    public FoodData() {

```

```

    }

    public FoodData(String itemName, String itemDescription, String
itemPrice, String itemImage) {
        this.itemName = itemName;
        this.itemDescription = itemDescription;
        this.itemPrice = itemPrice;
        this.itemImage = itemImage;
    }

    public FoodData(String itemName, String itemDescription, String
itemPrice, String itemImage, String key, String recipeKey){
        this.itemName = itemName;
        this.itemDescription = itemDescription;
        this.itemPrice = itemPrice;
        this.itemImage = itemImage;
        this.key = key;
        this.recipeKey = recipeKey;
    }

    public String getItemName() {
        return itemName;
    }

    public String getItemDescription() {
        return itemDescription;
    }

    public String getItemPrice() {
        return itemPrice;
    }

    public String getItemImage() {
        return itemImage;
    }

    public String getKey() {
        return key;
    }

    public void setKey(String key) {
        this.key = key;
    }

    public String getRecipeKey() {
        return recipeKey;
    }

    public void setRecipeKey(String recipeKey) {
        this.recipeKey = recipeKey;
    }
}

```


loginactivity

```
package ca.mohawk.foodrecipeapp;

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

import android.content.DialogInterface;
import android.content.Intent;
import android.os.Bundle;
import android.text.TextUtils;
import android.view.View;
import android.view.WindowManager;
import android.widget.EditText;
import android.widget.ProgressBar;
import android.widget.TextView;
import android.widget.Toast;

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.android.material.snackbar.BaseTransientBottomBar;
import com.google.android.material.snackbar.Snackbar;
import com.google.firebase.auth.AuthResult;
import com.google.firebase.auth.FirebaseAuth;

public class LoginActivity extends AppCompatActivity {

    EditText txt_login, txt_password;
    ProgressBar loginProgressBar;
    FirebaseAuth firebaseAuth;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_login);

        getWindow().setSoftInputMode(WindowManager.LayoutParams.SOFT_INPUT_STATE_HIDDEN);

        txt_login = findViewById(R.id.txt_login);
        txt_password = findViewById(R.id.txt_password);
        loginProgressBar = findViewById(R.id.loginProgressBar);
        firebaseAuth = FirebaseAuth.getInstance();

        if (firebaseAuth.getCurrentUser() != null) {
            startActivity(new
```

```

Intent(getApplicationContext(),MainActivity.class));
    finish();
}

Intent intent = getIntent();
intent.getExtras();

if(intent.hasExtra("warning"))
{
    Snackbar.
make(findViewById(R.id.txt_login),intent.getStringExtra("warning"),BaseTransi
entBottomBar.LENGTH_LONG).show();
}
}

public void btnLogin(final View view) {
    String email = txt_login.getText().toString().trim();
    String password = txt_password.getText().toString().trim();

    if (TextUtils.isEmpty(email)) {
        txt_login.setError("Email is Required.");
        return;
    }

    if (TextUtils.isEmpty(password)) {
        txt_password.setError("Password is Required.");
        return;
    }

    if (password.length() < 6) {
        txt_password.setError("Password Must be >= 6 Characters");
        return;
    }

    loginProgressBar.setVisibility(View.VISIBLE);

    firebaseAuth.signInWithEmailAndPassword(email,password).addOnCompleteListener
(new OnCompleteListener<AuthResult>() {
        @Override
        public void onComplete(@NonNull Task<AuthResult> task) {
            if (task.isSuccessful()) {
                finish();
                startActivity(new
Intent(getApplicationContext(),MainActivity.class));
            }
            else {
                Snackbar.make(view,"Error ! " +
task.getException().getMessage(), BaseTransientBottomBar.LENGTH_LONG).show();
                loginProgressBar.setVisibility(View.INVISIBLE);
            }
        }
    });
}

public void btnRegister(View view) {
    startActivity(new Intent(this,RegisterActivity.class));
}

```

```

    }

    public void btnForgotPassword(final View view) {

        final EditText resetMail = new EditText(view.getContext());
        final AlertDialog.Builder passwordResetDialog = new
AlertDialog.Builder(view.getContext());
        passwordResetDialog.setTitle("Reset Password");
        passwordResetDialog.setMessage("Enter Your Email to receive reset
link.");
        passwordResetDialog.setView(resetMail);

        passwordResetDialog.setPositiveButton("Yes", new
DialogInterface.OnClickListener() {
            @Override
            public void onClick(DialogInterface dialogInterface, int i) {

                String mail = resetMail.getText().toString();

                firebaseAuth.sendPasswordResetEmail(mail).addOnSuccessListener(new
OnSuccessListener<Void>() {
                    @Override
                    public void onSuccess(Void aVoid) {
                        Snackbar.make(view, "Reset Link Sent to your
Email", BaseTransientBottomBar.LENGTH_SHORT).show();
                    }
                }).addOnFailureListener(new OnFailureListener() {
                    @Override
                    public void onFailure(@NonNull Exception e) {
                        Snackbar.make(view, "Error Sending Link
!!", BaseTransientBottomBar.LENGTH_SHORT).show();
                    }
                });
            }
        });

        passwordResetDialog.setNegativeButton("No", new
DialogInterface.OnClickListener() {
            @Override
            public void onClick(DialogInterface dialogInterface, int i) {

            }
        });

        passwordResetDialog.create().show();
    }

    public void btnGuestLogin(View view) {
        startActivity(new
Intent(getApplicationContext(), MainActivity.class));
    }
}

```

mainactivity

```

package ca.mohawk.foodrecipeapp;

import androidx.annotation.NonNull;
import androidx.appcompat.app.AlertDialog;
import androidx.appcompat.app.AppCompatActivity;
import androidx.navigation.NavController;
import androidx.navigation.Navigation;
import androidx.navigation.ui.AppBarConfiguration;
import androidx.navigation.ui.NavigationUI;
import android.content.DialogInterface;
import android.content.Intent;
import android.os.Bundle;
import android.util.Log;
import android.view.View;
import android.widget.EditText;

import com.google.android.gms.tasks.OnFailureListener;
import com.google.android.gms.tasks.OnSuccessListener;
import com.google.android.material.bottomnavigation.BottomNavigationView;
import com.google.android.material.snackbar.BaseTransientBottomBar;
import com.google.android.material.snackbar.Snackbar;
import com.google.firebase.auth.FirebaseAuth;
import com.google.firebase.auth.FirebaseUser;

public class MainActivity extends AppCompatActivity {

    FirebaseAuth firebaseAuth;
    FirebaseUser firebaseUser;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        BottomNavigationView navView = findViewById(R.id.nav_view);
        // Passing each menu ID as a set of Ids because each
        // menu should be considered as top level destinations.
        AppBarConfiguration appBarConfiguration = new
AppBarConfiguration.Builder(
            R.id.navigation_home, R.id.navigation_dashboard,
R.id.favourite_dashboard ,R.id.navigation_notifications)
                .build();
        NavController navController = Navigation.findNavController(this,
R.id.nav_host_fragment);
        NavigationUI.setupActionBarWithNavController(this, navController,
appBarConfiguration);
        NavigationUI.setupWithNavController(navView, navController);

        firebaseAuth = FirebaseAuth.getInstance();
        firebaseUser = firebaseAuth.getCurrentUser();
    }
}

```

```

//      if (firebaseAuth.getCurrentUser() == null){
//          finish();
//          startActivity(new Intent(this,LoginActivity.class));
//      }

    }

    public void btn_uploadActivity(View view) {
        startActivity(new Intent(this, Upload_Recipe.class));
    }

    public void btnLogout(View view) {
        FirebaseAuth.getInstance().signOut();
        startActivity(new
Intent(getApplicationContext(),LoginActivity.class));
        finish();
    }

    public void btnVerifyEmail(final View view) {
        firebaseUser.sendEmailVerification().addOnSuccessListener(new
OnSuccessListener<Void>() {
            @Override
            public void onSuccess(Void aVoid) {
                Snackbar.make(view,"Verification Email has been Sent,
remember to Logout and Login again after verification!!",
BaseTransientBottomBar.LENGTH_LONG).show();

            }
        }).addOnFailureListener(new OnFailureListener() {
            @Override
            public void onFailure(@NonNull Exception e) {
                Log.d("TAG","onFailure: Email not sent " + e.getMessage());
            }
        });
    }

    public void btnResetPassword(final View view) {

        final EditText resetPassword = new EditText(view.getContext());
        final AlertDialog.Builder passwordResetDialog = new
AlertDialog.Builder(view.getContext());
        passwordResetDialog.setTitle("Reset Password ?");
        passwordResetDialog.setMessage("Enter New Password greater than 6
characters long.");
        passwordResetDialog.setView(resetPassword);

        passwordResetDialog.setPositiveButton("Yes", new
DialogInterface.OnClickListener() {
            @Override
            public void onClick(DialogInterface dialogInterface, int i) {

                String newPassword = resetPassword.getText().toString();

                firebaseUser.updatePassword(newPassword).addOnSuccessListener(new
OnSuccessListener<Void>() {
                    @Override

```

```

        public void onSuccess(Void aVoid) {
            Snackbar.make(view, "Password Reset Successfully.",
BaseTransientBottomBar.LENGTH_LONG).show();
        }
    }).addOnFailureListener(new OnFailureListener() {
        @Override
        public void onFailure(@NonNull Exception e) {
            Snackbar.make(view, "Password Reset Failed!!",
BaseTransientBottomBar.LENGTH_LONG).show();
        }
    });
}

passwordResetDialog.setNegativeButton("No", new
DialogInterface.OnClickListener() {
    @Override
    public void onClick(DialogInterface dialogInterface, int i) {

    }
});

passwordResetDialog.create().show();
}

public void onBackPressed() {
    finish();
}
}

```

myadapter.java

```

package ca.mohawk.foodrecipeapp;

import android.content.Context;
import android.content.Intent;
import android.util.Log;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.view.animation.Animation;
import android.view.animation.ScaleAnimation;
import android.widget.ImageView;
import android.widget.TextView;
import android.widget.Toast;

import com.bumptech.glide.Glide;
import com.google.android.gms.tasks.OnSuccessListener;
import com.google.firebase.auth.FirebaseAuth;
import com.google.firebase.firestore.DocumentReference;
import com.google.firebase.firestore.FirebaseFirestore;

```

```

import com.google.firebase.firestore.QueryDocumentSnapshot;
import com.google.firebase.firestore.QuerySnapshot;

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

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

public class MyAdapter extends RecyclerView.Adapter<FoodViewHolder>{

    private Context mContext;
    private List<FoodData> myFoodList;
    private int lastPosition = -1;
    String userID;
    String recipeKey = "";
    FirebaseAuth firebaseAuth;
    FirebaseFirestore firebaseFirestore;

    public MyAdapter(Context mContext, List<FoodData> myFoodList) {
        this.mContext = mContext;
        this.myFoodList = myFoodList;
        firebaseFirestore = FirebaseFirestore.getInstance();
        firebaseAuth = FirebaseAuth.getInstance();
    }

    @NonNull
    @Override
    public FoodViewHolder onCreateViewHolder(@NonNull ViewGroup parent, int
viewType) {
        View itemView = LayoutInflater.from(mContext)
            .inflate(R.layout.recycler_row_item, parent, false);
        return new FoodViewHolder(itemView);
    }

    @NonNull
    @Override
    public void onBindViewHolder(@NonNull final FoodViewHolder holder, int
position) {

        Glide.with(mContext).load(myFoodList.get(position).getItemImage()).into(holder.imageView);

        holder.mTitle.setText(myFoodList.get(position).getItemName());

        holder.mDescription.setText(myFoodList.get(position).getItemDescription());
        holder.mPrice.setText(myFoodList.get(position).getItemPrice());

        holder.mCardView.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                final Intent intent = new
Intent(mContext, DetailActivity.class);
                intent.putExtra("RecipeName", myFoodList.get(holder.getAdapterPosition()).getI

```

```

temName());

intent.putExtra("price",myFoodList.get(holder.getAdapterPosition()).getItemPrice());

intent.putExtra("Image",myFoodList.get(holder.getAdapterPosition()).getItemImage());

intent.putExtra("Description",myFoodList.get(holder.getAdapterPosition()).getItemDescription());
        intent.putExtra("keyValue",
myFoodList.get(holder.getAdapterPosition()).getKey());

        if (firebaseAuth.getCurrentUser() != null){
            userID = firebaseAuth.getCurrentUser().getUid();

if(myFoodList.get(holder.getAdapterPosition()).getRecipeKey() == null) {
            final DocumentReference documentReference =
firebaseFirestore.collection("users").document(userID);

documentReference.collection("Recipes").get().addOnSuccessListener(new
OnSuccessListener<QuerySnapshot>() {
            @Override
            public void onSuccess(QuerySnapshot
queryDocumentSnapshots) {
                for (QueryDocumentSnapshot documentSnapshot :
queryDocumentSnapshots) {
                    FoodData foodData =
documentSnapshot.toObject(FoodData.class);
                    if
(foodData.getKey().equals(myFoodList.get(holder.getAdapterPosition()).getKey(
))) {
                        //
                        Log.d("TAG inside RecipeKey : ",
foodData.getRecipeKey());
                        recipeKey = foodData.getRecipeKey();
                        intent.putExtra("recipeKeyValue",
recipeKey);
                        mContext.startActivity(intent);
                    }
                }
            }

            if (recipeKey.equals("")){
                Log.d("TAG outside ",recipeKey);
                intent.putExtra("recipeKeyValue",
myFoodList.get(holder.getAdapterPosition()).getRecipeKey());
                mContext.startActivity(intent);
            }
        }
    });
}
else{
    Log.d("TAG myAdapter",recipeKey + " myAdapter");
    intent.putExtra("recipeKeyValue",
myFoodList.get(holder.getAdapterPosition()).getRecipeKey());
    mContext.startActivity(intent);
}
//
}
}

```



```

        } else {
            intent.putExtra("recipeKeyValue",
myFoodList.get(holder.getAdapterPosition()).getRecipeKey());
            mContext.startActivity(intent);
        }
    }
});

setAnimation(holder.itemView, position);
}

public void setAnimation(View viewToAnimate, int position) {
    if (position > lastPosition) {
        ScaleAnimation animation = new
ScaleAnimation(0.0f, 1.0f, 0.0f, 1.0f,
Animation.RELATIVE_TO_SELF, 0.5f, Animation.RELATIVE_TO_SELF, 0.5f);
        animation.setDuration(1500);
        viewToAnimate.startAnimation(animation);
        lastPosition = position;
    }
}

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

public void filteredList(ArrayList<FoodData> filterList) {
    myFoodList = filterList;
    notifyDataSetChanged();
}
}

class FoodViewHolder extends RecyclerView.ViewHolder {

    ImageView imageView;
    TextView mTitle, mDescription, mPrice;
    CardView mCardView;

    public FoodViewHolder(@NonNull View itemView) {
        super(itemView);

        imageView = itemView.findViewById(R.id.ivImage);
        mTitle = itemView.findViewById(R.id.tvTitle);
        mDescription = itemView.findViewById(R.id.tvDescription);
        mPrice = itemView.findViewById(R.id.tvPrice);

        mCardView = itemView.findViewById(R.id.myCardView);
    }
}

```

registeractivity

```
package ca.mohawk.foodrecipeapp;

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

import android.content.Intent;
import android.os.Bundle;
import android.text.TextUtils;
import android.util.Log;
import android.view.View;
import android.view.WindowManager;
import android.widget.EditText;
import android.widget.ProgressBar;
import android.widget.TextView;
import android.widget.Toast;

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.android.material.snackbar.BaseTransientBottomBar;
import com.google.android.material.snackbar.Snackbar;
import com.google.firebase.auth.AuthResult;
import com.google.firebase.auth.FirebaseAuth;
import com.google.firebase.auth.FirebaseUser;
import com.google.firebase.firestore.DocumentReference;
import com.google.firebase.firestore.FirebaseFirestore;

import java.util.HashMap;
import java.util.Map;

public class RegisterActivity extends AppCompatActivity {

    EditText mName, mPhone , mEmail, mPassword;
    TextView mLogin;
    ProgressBar mProgressBar;
    FirebaseAuth firebaseAuth;
    String TAG = "TAG";
    FirebaseFirestore firebaseFirestore;
    String userID;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_register);

        getWindow().setSoftInputMode(WindowManager.LayoutParams.SOFT_INPUT_STATE_HIDDEN);

        mName = findViewById(R.id.txt_name);
```

```

        mPhone = findViewById(R.id.txt_phone);
        mEmail = findViewById(R.id.txt_email);
        mPassword = findViewById(R.id.txt_password);
        mLogin = findViewById(R.id.txt_login);
        mProgressBar = findViewById(R.id.registerProgressBar);

        firebaseAuth = FirebaseAuth.getInstance();
        firebaseFirestore = FirebaseFirestore.getInstance();

        if (firebaseAuth.getCurrentUser() != null){
            finish();
            startActivity(new
Intent(getApplicationContext(),MainActivity.class));
        }

    }

    public void register(final View view) {
        final String email = mEmail.getText().toString().trim();
        String password = mPassword.getText().toString().trim();
        final String fullName = mName.getText().toString().trim();
        final String phone = mPhone.getText().toString().trim();

        if (TextUtils.isEmpty(email)){
            mEmail.setError("Email is Required.");
            return;
        }

        if (TextUtils.isEmpty(password)){
            mPassword.setError("Password is Required.");
            return;
        }

        if (password.length() < 6){
            mPassword.setError("Password Must be >= 6 Characters");
            return;
        }

        mProgressBar.setVisibility(View.VISIBLE);

        // register the user in Firebase

        firebaseAuth.createUserWithEmailAndPassword(email,password).addOnCompleteListener(
new OnCompleteListener<AuthResult>() {
            @Override
            public void onComplete(@NonNull Task<AuthResult> task) {
                if (task.isSuccessful()){

                    userID = firebaseAuth.getCurrentUser().getUid();
                    DocumentReference documentReference =
firebaseFirestore.collection("users").document(userID);
                    final Map<String,Object> mapUser = new HashMap<>();
                    mapUser.put("fullName", fullName);
                    mapUser.put("email",email);
                    mapUser.put("phone",phone);
                    documentReference.set(mapUser).addOnSuccessListener(new
OnSuccessListener<Void>() {

```

```

        @Override
        public void onSuccess(Void aVoid) {
            Log.d("TAG", "onSuccess: user Profile is created
for "+ mapUser);
        }
    }).addOnFailureListener(new OnFailureListener() {
        @Override
        public void onFailure(@NonNull Exception e) {
            Log.d("TAG", "onFailure: " + e.toString());
        }
    });

    // Sending Verification Link
    FirebaseUser user = firebaseAuth.getCurrentUser();
    user.sendEmailVerification().addOnSuccessListener(new
OnSuccessListener<Void>() {
        @Override
        public void onSuccess(Void aVoid) {
            Snackbar.make(view, "Verification Email has been
Sent", BaseTransientBottomBar.LENGTH_LONG).show();
        }
    }).addOnFailureListener(new OnFailureListener() {
        @Override
        public void onFailure(@NonNull Exception e) {

            Log.d(TAG, "onFailure: Email not sent " +
e.getMessage());
        }
    });

    Snackbar.make(view, "User Created",
BaseTransientBottomBar.LENGTH_LONG).show();
    finish();
    startActivity(new
Intent(getApplicationContext(), MainActivity.class));
}
else {
    Snackbar.make(view, "Error ! " +
task.getException().getMessage(), BaseTransientBottomBar.LENGTH_LONG).show();
    mProgressBar.setVisibility(View.INVISIBLE);
}
}
});
}

public void login(View view) {
    startActivity(new Intent(this, LoginActivity.class));
}
}

```

updaterecipeactivity

```

package ca.mohawk.foodrecipeapp;

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

import android.app.ProgressDialog;
import android.content.Intent;
import android.net.Uri;
import android.os.Bundle;
import android.util.Log;
import android.view.View;
import android.view.WindowManager;
import android.widget.EditText;
import android.widget.ImageView;
import android.widget.Toast;

import com.bumptech.glide.Glide;
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.android.material.snackbar.BaseTransientBottomBar;
import com.google.android.material.snackbar.Snackbar;
import com.google.firebase.auth.FirebaseAuth;
import com.google.firebase.database.DatabaseReference;
import com.google.firebase.database.FirebaseDatabase;
import com.google.firebase.firestore.CollectionReference;
import com.google.firebase.firestore.DocumentReference;
import com.google.firebase.firestore.FirebaseFirestore;
import com.google.firebase.storage.FirebaseStorage;
import com.google.firebase.storage.StorageReference;
import com.google.firebase.storage.UploadTask;

import java.text.DateFormat;
import java.util.Calendar;

public class UpdateRecipeActivity extends AppCompatActivity {

    ImageView recipeImage;
    Uri uri;
    EditText txt_name,txt_description, txt_price;
    String imageUrl, userID;
    String key, oldImageUrl, recipeKey;
    String recipename, recipeDescription, recipePrice;
    DatabaseReference databaseReference;
    StorageReference storageReference;
    FirebaseAuth firebaseAuth;
    FirebaseFirestore firebaseFirestore;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);

```

```

        setContentView(R.layout.activity_update_recipe);

        recipeImage = findViewById(R.id.iv_foodImage);
        txt_name = findViewById(R.id.txt_recipe_name);
        txt_description = findViewById(R.id.text_description);
        txt_price = findViewById(R.id.text_price);

getWindow().setSoftInputMode(WindowManager.LayoutParams.SOFT_INPUT_STATE_HIDDEN);

        firebaseAuth = FirebaseAuth.getInstance();
        firebaseFirestore = FirebaseFirestore.getInstance();
        userID = firebaseAuth.getCurrentUser().getUid();

        Bundle bundle = getIntent().getExtras();
        if(bundle != null){

Glide.with(UpdateRecipeActivity.this).load(bundle.getString("oldImageUrl")).into(recipeImage);
            txt_name.setText(bundle.getString("recipeNameKey"));
            txt_description.setText(bundle.getString("descriptionKey"));
            txt_price.setText(bundle.getString("priceKey"));
            key = bundle.getString("key");
            recipeKey = bundle.getString("recipeKey");
            oldImageUrl = bundle.getString("oldImageUrl");
        }

        databaseReference =
FirebaseDatabase.getInstance().getReference("Recipe").child(key);

    }

    public void uploadRecipe() {

        FoodData foodData = new FoodData(
            recipename, recipeDescription, recipePrice, imageUrl);

        databaseReference.setValue(foodData).addOnCompleteListener(new
OnCompleteListener<Void>() {
            @Override
            public void onComplete(@NonNull Task<Void> task) {
                if(imageUrl != oldImageUrl ){
                    StorageReference storageReferenceNew =
FirebaseStorage.getInstance().getReferenceFromUrl(oldImageUrl);
                    storageReferenceNew.delete();
                }
            }
        });

        DocumentReference documentReference =
firebaseFirestore.collection("users").document(userID).collection("Recipes").
document(recipeKey);

documentReference.update("itemName", recipename, "itemDescription", recipeDescri

```

```

ption, "itemPrice", recipePrice, "itemImage", imageUrl)
    .addOnSuccessListener(new OnSuccessListener<Void>() {
        @Override
        public void onSuccess(Void aVoid) {
            Log.d("TAG", " Recipe updated to Firestore");
            Snackbar.
make(findViewById(R.id.txt_recipe_name), "Recipe
Uploaded", BaseTransientBottomBar.LENGTH_SHORT).show();
            finish();
        }
    }).addOnFailureListener(new OnFailureListener() {
        @Override
        public void onFailure(@NonNull Exception e) {
            Log.d("TAG", " Recipe not updated to Firestore " +
e.getMessage());
            Snackbar.make(findViewById(R.id.txt_recipe_name), "Failed to
Upload Recipe" + e.getMessage(), BaseTransientBottomBar.LENGTH_SHORT).show();
        }
    });

    finish();
    Intent intent = new Intent(this, DetailActivity.class);
    intent.putExtra("RecipeName", recipename);
    intent.putExtra("price", recipePrice);
    intent.putExtra("Image", imageUrl);
    intent.putExtra("Description", recipeDescription);
    intent.putExtra("keyValue", key);
    intent.putExtra("recipeKeyValue", recipeKey);

    startActivity(intent);
}

public void btnSelectImage(View view) {

    Intent photoPicker = new Intent(Intent.ACTION_PICK);
    photoPicker.setType("image/*");
    startActivityForResult(photoPicker, 1);

}

@Override
protected void onActivityResult(int requestCode, int resultCode,
@Nullable Intent data) {
    super.onActivityResult(requestCode, resultCode, data);

    if (resultCode == RESULT_OK) {
        uri = data.getData();
        recipeImage.setImageURI(uri);
    }
    else {
        Snackbar.make(findViewById(R.id.txt_recipe_name), "You haven't
picked image", BaseTransientBottomBar.LENGTH_LONG).show();
    }
}

public void btnUpdateRecipe(View view) {

```

```

        recipename = txt_name.getText().toString().trim();
        recipeDescription = txt_description.getText().toString().trim();
        recipePrice = txt_price.getText().toString().trim();

        final ProgressDialog progressDialog = new ProgressDialog(this);
        progressDialog.setMessage("Recipe Uploading...");
        progressDialog.show();

        if(uri == null){
            imageUrl = oldImageUrl;
            uploadRecipe();
            progressDialog.dismiss();
        }
        else{
            storageReference =
                FirebaseStorage.getInstance().getReference().child("RecipeImage").child(uri.g
                etLastPathSegment());
            storageReference.putFile(uri).addOnSuccessListener(new
            OnSuccessListener<UploadTask.TaskSnapshot>() {
                @Override
                public void onSuccess(UploadTask.TaskSnapshot taskSnapshot) {

                    Task<Uri> uriTask =
                    taskSnapshot.getStorage().getDownloadUrl();
                    while(!uriTask.isComplete());
                    Uri urlImage = uriTask.getResult();
                    imageUrl = urlImage.toString();
                    uploadRecipe();
                    progressDialog.dismiss();
                }
            }).addOnFailureListener(new OnFailureListener() {
                @Override
                public void onFailure(@NonNull Exception e) {
                    progressDialog.dismiss();
                }
            });
        }
    }
}

```

upload_recipe

```

package ca.mohawk.foodrecipeapp;

import androidx.annotation.NonNull;

```



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

import android.app.ProgressDialog;
import android.content.Intent;
import android.net.Uri;
import android.os.Bundle;
import android.util.Log;
import android.view.View;
import android.view.WindowManager;
import android.widget.EditText;
import android.widget.ImageView;

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.android.material.snackbar.BaseTransientBottomBar;
import com.google.android.material.snackbar.Snackbar;
import com.google.firebase.auth.FirebaseAuth;
import com.google.firebase.database.FirebaseDatabase;
import com.google.firebase.firestore.CollectionReference;
import com.google.firebase.firestore.DocumentReference;
import com.google.firebase.firestore.FirebaseFirestore;
import com.google.firebase.storage.FirebaseStorage;
import com.google.firebase.storage.StorageReference;
import com.google.firebase.storage.UploadTask;

import java.text.DateFormat;
import java.util.Calendar;
import java.util.HashMap;
import java.util.Map;

public class Upload_Recipe extends AppCompatActivity {

    ImageView recipeImage;
    Uri uri;
    EditText txt_name,txt_description, txt_price;
    String imageUrl, userID;
    FirebaseAuth firebaseAuth;
    FirebaseFirestore firebaseFirestore;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_upload_recipe);

        recipeImage = findViewById(R.id.iv_foodImage);
        txt_name = findViewById(R.id.txt_recipe_name);
        txt_description = findViewById(R.id.text_description);
        txt_price = findViewById(R.id.text_price);

        getWindow().setSoftInputMode(WindowManager.LayoutParams.SOFT_INPUT_STATE_HIDDEN);

        firebaseAuth = FirebaseAuth.getInstance();
```

```

        firebaseFirestore = FirebaseFirestore.getInstance();

        userID = firebaseAuth.getCurrentUser().getUid();

    }

    public void btnSelectImage(View view) {

        Intent photoPicker = new Intent(Intent.ACTION_PICK);
        photoPicker.setType("image/*");
        startActivityForResult(photoPicker, 1);

    }

    @Override
    protected void onActivityResult(int requestCode, int resultCode,
    @Nullable Intent data) {
        super.onActivityResult(requestCode, resultCode, data);

        if (resultCode == RESULT_OK) {
            uri = data.getData();
            recipeImage.setImageURI(uri);
        }
        else {
            Snackbar.make(findViewById(R.id.txt_recipe_name), "You haven't
            picked image", BaseTransientBottomBar.LENGTH_LONG).show();
        }
    }

    public void uploadImage(final View view) {

        StorageReference storageReference =
        FirebaseStorage.getInstance().getReference().child("RecipeImage").child(uri.g
        etLastPathSegment());

        final ProgressDialog progressDialog = new ProgressDialog(this);
        progressDialog.setMessage("Recipe Uploading...");
        progressDialog.show();

        storageReference.putFile(uri).addOnSuccessListener(new
        OnSuccessListener<UploadTask.TaskSnapshot>() {
            @Override
            public void onSuccess(UploadTask.TaskSnapshot taskSnapshot) {

                Task<Uri> uriTask =
                taskSnapshot.getStorage().getDownloadUrl();
                while (!uriTask.isComplete());
                Uri urlImage = uriTask.getResult();
                imageUrl = urlImage.toString();
                uploadRecipe(view);
                progressDialog.dismiss();

            }
        }).addOnFailureListener(new OnFailureListener() {
            @Override
            public void onFailure(@NonNull Exception e) {
                progressDialog.dismiss();
            }
        });
    }

```

```

    }

    public void btnUploadRecipe(View view) {

        if(uri == null ){
            Snackbar.make(view,"Please Upload
Image",Snackbar.LENGTH_LONG).show();
        }
        else if(txt_name.getText().toString().isEmpty()){
            Snackbar.make(view,"Please Write Recipe
Name",Snackbar.LENGTH_LONG).show();
        }
        else if(txt_description.getText().toString().isEmpty()){
            Snackbar.make(view,"Please Write Recipe
Description",Snackbar.LENGTH_LONG).show();
        }
        else {
            uploadImage(view);
        }
    }

    public void uploadRecipe(final View view) {

        FoodData foodData = new
FoodData(txt_name.getText().toString(),txt_description.getText().toString(),t
xt_price.getText().toString(),imageUrl);

        String myCurrentDateTime =
DateFormat.getDateTimeInstance().format(Calendar.getInstance().getTime());

        FirebaseDatabase.getInstance().getReference("Recipe").child(myCurrentDateTime
).setValue(foodData).addOnCompleteListener(new OnCompleteListener<Void>() {
            @Override
            public void onComplete(@NonNull Task<Void> task) {

                if(task.isSuccessful()){
                    Log.d("TAG", " Recipe added to Database");
                }
            }
        }).addOnFailureListener(new OnFailureListener() {
            @Override
            public void onFailure(@NonNull Exception e) {
                Log.d("TAG", " Recipe not added to Database " +
e.getMessage());
            }
        });

        String recipeID =
firebaseFirestore.collection("users").document(userID).collection("Recipes").
document().getId();
        FoodData fireStoreFoodData = new
FoodData(txt_name.getText().toString(),txt_description.getText().toString(),t
xt_price.getText().toString(),imageUrl,myCurrentDateTime,recipeID);

        DocumentReference documentReference =
firebaseFirestore.collection("users").document(userID).collection("Recipes").

```

```

document(recipeID);
    documentReference.set(fireStoreFoodData).addOnSuccessListener(new
OnSuccessListener<Void>() {
        @Override
        public void onSuccess(Void aVoid) {
            Log.d("TAG", " Recipe added to Firestore");
            Snackbar.make(view,"Recipe
Uploaded",BaseTransientBottomBar.LENGTH_SHORT).show();
            finish();
        }
    }).addOnFailureListener(new OnFailureListener() {
        @Override
        public void onFailure(@NonNull Exception e) {
            Log.d("TAG", " Recipe not added to Firestore " +
e.getMessage());
            Snackbar.make(view,"Failed to Upload Recipe" +
e.getMessage(), BaseTransientBottomBar.LENGTH_SHORT).show();
        }
    });
}
}

```

activity_authentication.xml

```

<?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=".AuthenticationActivity">

</androidx.constraintlayout.widget.ConstraintLayout>

```

activity_detail.xml

```

<?xml version="1.0" encoding="utf-8"?>

<ScrollView 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"
    xmlns:android="http://schemas.android.com/apk/res/android">

```

```
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="#F9F6ED"
    android:orientation="vertical"
    android:scrollbarSize="19sp"
    tools:context=".DetailActivity">

    <ImageView
        android:id="@+id/ivImage2"
        android:layout_width="match_parent"
        android:layout_height="200dp"
        android:scaleType="centerCrop"
        app:srcCompat="@drawable/image" />

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:orientation="horizontal"
        android:weightSum="2">

        <TextView
            android:id="@+id/txtRecipeName"
            android:layout_width="0dp"
            android:layout_height="wrap_content"
            android:layout_marginTop="10dp"
            android:layout_weight="1.7"
            android:padding="20dp"
            android:text="Recipe"
            android:textSize="22sp"
            android:textStyle="bold" />

        <ImageView
            android:id="@+id/like_imageView"
            android:layout_width="0dp"
            android:layout_height="30dp"
            android:layout_marginTop="35dp"
            android:layout_weight="0.3"
            android:onClick="btnLike"
            android:visibility="gone"
            app:srcCompat="@drawable/ic_baseline_favorite_24" />

        <ImageView
            android:id="@+id/dislike_imagView"
            android:layout_width="0dp"
            android:layout_height="30dp"
            android:layout_marginTop="35dp"
            android:layout_weight="0.3"
            android:onClick="btnDislike"
            android:visibility="visible"
            app:srcCompat="@drawable/ic_baseline_favorite_border_24" />

    </LinearLayout>

    <TextView
```

```
        android:id="@+id/txtDescription"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginTop="10dp"
        android:padding="20dp"
        android:text="Description"
        android:textSize="19sp" />

    <TextView
        android:id="@+id/txtPrice"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginTop="10dp"
        android:gravity="center"
        android:padding="20dp"
        android:text="20 mins"
        android:textColor="@color/colorPrimary"
        android:textSize="19sp"
        android:textStyle="bold" />

    <RelativeLayout
        android:layout_width="match_parent"
        android:layout_height="match_parent">

        <Button
            android:id="@+id/btnUpdate"
            android:layout_width="140dp"
            android:layout_height="60dp"
            android:layout_alignParentBottom="true"
            android:layout_marginStart="20dp"
            android:layout_marginTop="20dp"
            android:layout_marginEnd="20dp"
            android:layout_marginBottom="20dp"
            android:backgroundTint="@color/colorPrimary"
            android:onClick="btnUpdateRecipe"
            android:text="Update"
            android:textColor="#FFFFFF"
            android:textSize="22sp"
            android:visibility="gone" />

        <Button
            android:id="@+id/btnDelete"
            android:layout_width="140dp"
            android:layout_height="60dp"
            android:layout_alignParentRight="true"
            android:layout_alignParentBottom="true"
            android:layout_marginStart="20dp"
            android:layout_marginTop="20dp"
            android:layout_marginEnd="20dp"
            android:layout_marginBottom="20dp"
            android:backgroundTint="@color/colorPrimary"
            android:onClick="btnDeleteRecipe"
            android:text="Delete"
            android:textColor="#FFFFFF"
            android:textSize="22sp"
            android:visibility="gone" />
```

```
        </RelativeLayout>

    </LinearLayout>
</ScrollView>
```

activity_login.xml

```
<?xml version="1.0" encoding="utf-8"?>
<ScrollView 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:background="#F9F6ED"
    android:orientation="vertical"
    tools:context=".LoginActivity">

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:orientation="vertical"
        android:padding="30dp">

        <ImageView
            android:id="@+id/iv_foodImage"
            android:layout_width="match_parent"
            android:layout_height="200dp"
            android:scaleType="centerCrop"
            app:srcCompat="@drawable/image2" />

        <ProgressBar
            android:id="@+id/loginProgressBar"
            style="?android:attr/progressBarStyle"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:visibility="invisible" />

        <LinearLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_marginTop="25dp"
            android:orientation="horizontal"
            android:weightSum="2">

            <ImageView
                android:layout_width="wrap_content"
                android:layout_height="25dp"
                android:layout_gravity="center"
                android:layout_weight="0.3"
```

```
        app:srcCompat="@drawable/ic_baseline_account_circle_24" />

    <EditText
        android:id="@+id/txt_login"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="center_horizontal"
        android:layout_weight="1.7"
        android:backgroundTint="#F9F6ED"
        android:hint="Username"
        android:inputType="textEmailAddress"
        android:textSize="18sp" />
</LinearLayout>

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginTop="10dp"
    android:layout_marginBottom="50dp"
    android:orientation="horizontal"
    android:weightSum="2">

    <ImageView
        android:layout_width="wrap_content"
        android:layout_height="25dp"
        android:layout_gravity="center"
        android:layout_weight="0.3"
        app:srcCompat="@drawable/ic_baseline_lock_24" />

    <EditText
        android:id="@+id/txt_password"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="center_horizontal"
        android:layout_weight="1.7"
        android:backgroundTint="#F9F6ED"
        android:hint="Password"
        android:inputType="textWebPassword"
        android:textSize="18sp" />
</LinearLayout>

<Button
    android:id="@+id/btnLogin"
    android:layout_width="120dp"
    android:layout_height="50dp"
    android:layout_gravity="center"
    android:layout_marginBottom="20dp"
    android:backgroundTint="@color/colorPrimary"
    android:onClick="btnLogin"
    android:text="Login"
    android:textColor="#ffffff"
    android:textSize="18sp" />

<TextView
    android:id="@+id/txt_forgotPassword"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
```



```

        android:layout_marginBottom="20dp"
        android:gravity="center"
        android:onClick="btnForgotPassword"
        android:text="Forgot Password?"
        android:textColor="@color/colorPrimary"
        android:textSize="16sp"
        android:textStyle="bold" />

        <TextView
            android:id="@+id/txt_register"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:gravity="center"
            android:text="Don't have an account yet? SIGN UP"
            android:textColor="@color/colorPrimary"
            android:textSize="16sp"
            android:textStyle="bold"
            android:onClick="btnRegister"/>

        <Button
            android:id="@+id/btnGuestLogin"
            android:layout_width="wrap_content"
            android:layout_height="50dp"
            android:layout_gravity="center"
            android:layout_marginTop="20dp"
            android:layout_marginBottom="20dp"
            android:backgroundTint="@color/colorPrimary"
            android:onClick="btnGuestLogin"
            android:text="Guest Login"
            android:textColor="#ffffff"
            android:textSize="18sp" />

    </LinearLayout>

</ScrollView>

```

activity_main.xml

```

<?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"
    android:id="@+id/container"
    android:layout_width="match_parent"
    android:layout_height="match_parent">

    <com.google.android.material.bottomnavigation.BottomNavigationView
        android:id="@+id/nav_view"
        android:layout_width="0dp"
        android:layout_height="wrap_content"
        android:layout_marginStart="0dp"

```

```

        android:layout_marginEnd="0dp"
        android:background="?android:attr/windowBackground"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintLeft_toLeftOf="parent"
        app:layout_constraintRight_toRightOf="parent"
        app:menu="@menu/bottom_nav_menu" />

<fragment
    android:id="@+id/nav_host_fragment"
    android:name="androidx.navigation.fragment.NavHostFragment"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    app:defaultNavHost="true"
    app:layout_constraintBottom_toTopOf="@id/nav_view"
    app:layout_constraintLeft_toLeftOf="parent"
    app:layout_constraintRight_toRightOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:navGraph="@navigation/mobile_navigation" />

</androidx.constraintlayout.widget.ConstraintLayout>

```

activity_register.xml

```

<?xml version="1.0" encoding="utf-8"?>
<ScrollView 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:background="#F9F6ED"
    android:orientation="vertical"
    tools:context=".RegisterActivity">

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:orientation="vertical"
        android:padding="30dp">

        <ImageView
            android:id="@+id/iv_foodImage"
            android:layout_width="match_parent"
            android:layout_height="200dp"
            android:scaleType="centerCrop"
            app:srcCompat="@drawable/image2" />

        <ProgressBar
            android:id="@+id/registerProgressBar"
            style="?android:attr/progressBarStyle"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"

```

```
        android:visibility="invisible" />

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginTop="10dp"
    android:orientation="horizontal"
    android:weightSum="2">

    <ImageView
        android:layout_width="30dp"
        android:layout_height="25dp"
        android:layout_gravity="center"
        android:layout_weight="0.3"
        app:srcCompat="@drawable/ic_baseline_person_24" />

    <EditText
        android:id="@+id/txt_name"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="center_horizontal"
        android:layout_weight="1.7"
        android:backgroundTint="#F9F6ED"
        android:hint="Full Name"
        android:inputType="textPersonName"
        android:textSize="18sp" />
</LinearLayout>

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginTop="10dp"
    android:orientation="horizontal"
    android:weightSum="2">

    <ImageView
        android:layout_width="wrap_content"
        android:layout_height="25dp"
        android:layout_gravity="center"
        android:layout_weight="0.3"
        app:srcCompat="@drawable/ic_baseline_local_phone_24" />

    <EditText
        android:id="@+id/txt_phone"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="center_horizontal"
        android:layout_weight="1.7"
        android:backgroundTint="#F9F6ED"
        android:hint="Phone"
        android:textSize="18sp" />
</LinearLayout>

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginTop="10dp"
```

```
        android:orientation="horizontal"
        android:weightSum="2">

        <ImageView
            android:layout_width="wrap_content"
            android:layout_height="25dp"
            android:layout_gravity="center"
            android:layout_weight="0.3"
            app:srcCompat="@drawable/ic_baseline_email_24" />

        <EditText
            android:id="@+id/txt_email"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_gravity="center_horizontal"
            android:layout_weight="1.7"
            android:backgroundTint="#F9F6ED"
            android:hint="Email"
            android:inputType="textEmailAddress"
            android:textSize="18sp" />
    </LinearLayout>

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginTop="10dp"
        android:layout_marginBottom="50dp"
        android:orientation="horizontal"
        android:weightSum="2">

        <ImageView
            android:layout_width="30dp"
            android:layout_height="25dp"
            android:layout_gravity="center"
            android:layout_weight="0.3"
            app:srcCompat="@drawable/ic_baseline_lock_24" />

        <EditText
            android:id="@+id/txt_password"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_gravity="center_horizontal"
            android:layout_weight="1.7"
            android:backgroundTint="#F9F6ED"
            android:hint="Password"
            android:inputType="textWebPassword"
            android:textSize="18sp" />
    </LinearLayout>

    <Button
        android:id="@+id/registerButton"
        android:layout_width="120dp"
        android:layout_height="50dp"
        android:layout_gravity="center"
        android:layout_marginBottom="30dp"
        android:backgroundTint="@color/colorPrimary"
        android:onClick="register"
```

```

        android:text="Register"
        android:textColor="#ffffff"
        android:textSize="18sp" />

        <TextView
            android:id="@+id/txt_login"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:gravity="center"
            android:onClick="login"
            android:text="Already have an account? SIGN IN"
            android:textColor="@color/colorPrimary"
            android:textSize="16sp"
            android:textStyle="bold" />

    </LinearLayout>

</ScrollView>

```

activity_update_recipe.xml

```

<?xml version="1.0" encoding="utf-8"?>
<ScrollView 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:background="#F9F6ED"
    android:orientation="vertical"
    tools:context=".UpdateRecipeActivity">

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:orientation="vertical"
        android:padding="30dp">

        <ImageView
            android:id="@+id/iv_foodImage"
            android:layout_width="match_parent"
            android:layout_height="200dp"
            android:scaleType="centerCrop"
            app:srcCompat="@drawable/placeholder" />

        <EditText
            android:id="@+id/txt_recipe_name"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_marginTop="20dp"
            android:hint="Recipe Name"
            android:textSize="19sp" />
    
```

```

        <EditText
            android:id="@+id/text_description"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_marginTop="10dp"
            android:hint="Description"
            android:textSize="19sp" />

        <EditText
            android:id="@+id/text_price"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_marginTop="10dp"
            android:layout_marginBottom="10dp"
            android:hint="Cook Time"
            android:textSize="19sp" />

        <Button
            android:layout_width="match_parent"
            android:layout_height="60dp"
            android:backgroundTint="@color/colorPrimary"
            android:onClick="btnSelectImage"
            android:text="Select Image"
            android:textColor="#ffffff"
            android:textSize="22sp" />

        <Button
            android:layout_width="match_parent"
            android:layout_height="60dp"
            android:backgroundTint="@color/colorPrimary"
            android:onClick="btnUpdateRecipe"
            android:text="Update Recipe"
            android:textColor="#ffffff"
            android:textSize="22sp" />

    </LinearLayout>
</ScrollView>

```

activity_upload_recipe.xml

```

<?xml version="1.0" encoding="utf-8"?>
<ScrollView 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:background="#F9F6ED"
    android:orientation="vertical"
    tools:context=".Upload Recipe">

```

```
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="vertical"
    android:padding="30dp">

    <ImageView
        android:id="@+id/iv_foodImage"
        android:layout_width="match_parent"
        android:layout_height="200dp"
        android:scaleType="centerCrop"
        app:srcCompat="@drawable/placeholder" />

    <EditText
        android:id="@+id/txt_recipe_name"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginTop="20dp"
        android:hint="Recipe Name"
        android:textSize="19sp" />

    <EditText
        android:id="@+id/text_description"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginTop="10dp"
        android:hint="Description"
        android:textSize="19sp" />

    <EditText
        android:id="@+id/text_price"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginTop="10dp"
        android:layout_marginBottom="10dp"
        android:hint="Cook Time"
        android:textSize="19sp" />

    <Button
        android:layout_width="match_parent"
        android:layout_height="60dp"
        android:backgroundTint="@color/colorPrimary"
        android:onClick="btnSelectImage"
        android:text="Select Image"
        android:textColor="#ffffff"
        android:textSize="22sp" />

    <Button
        android:layout_width="match_parent"
        android:layout_height="60dp"
        android:backgroundTint="@color/colorPrimary"
        android:onClick="btnUploadRecipe"
        android:text="Upload Recipe"
        android:textColor="#ffffff"
        android:textSize="22sp" />
```

```

        </LinearLayout>
    </ScrollView>

```

fragment_dashboard.xml

```

<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout 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:layout_marginBottom="60dp"
    android:background="#F9F6ED"
    android:orientation="vertical"
    tools:context=".MainActivity">

    <EditText
        android:id="@+id/txt_searchtext"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_margin="15dp"
        android:hint="Search"
        android:textSize="22sp" />

    <androidx.recyclerview.widget.RecyclerView
        android:id="@+id/recyclerView"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:layout_below="@+id/txt_searchtext"
        android:scrollbars="vertical" />

    <TextView
        android:id="@+id/txt_no_result_found"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_below="@+id/txt_searchtext"
        android:gravity="center"
        android:text="No Results Found"
        android:textSize="24sp"
        android:visibility="gone" />

    <com.google.android.material.floatingactionbutton.FloatingActionButton
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignParentRight="true"
        android:layout_alignParentBottom="true"
        android:layout_marginStart="40dp"
        android:layout_marginTop="40dp"
        android:layout_marginEnd="40dp"
        android:layout_marginBottom="20dp"
    >

```



```
        android:backgroundTint="@color/colorAccent"
        android:onClick="btn_uploadActivity"
        android:src="@drawable/ic_baseline_add_24" />

</RelativeLayout>
```

fragment_faviourets.xml

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout 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:layout_marginBottom="60dp"
    android:background="#F9F6ED"
    android:orientation="vertical"
    tools:context=".MainActivity">

    <EditText
        android:id="@+id/txt_searchtext"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_margin="15dp"
        android:hint="Search"
        android:textSize="22sp" />

    <androidx.recyclerview.widget.RecyclerView
        android:id="@+id/recyclerView"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:layout_below="@+id/txt_searchtext"
        android:scrollbars="vertical">

    </androidx.recyclerview.widget.RecyclerView>

    <TextView
        android:id="@+id/txt_empty"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_below="@+id/txt_searchtext"
        android:gravity="center"
        android:text="No Saved Recipes Found"
        android:textSize="24sp"
        android:visibility="gone" />

</RelativeLayout>
```

fragment_home.xml

```

<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout 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:layout_marginBottom="60dp"
    android:background="#F9F6ED"
    android:orientation="vertical"
    tools:context=".MainActivity">

    <EditText
        android:id="@+id/txt_searchtext"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_margin="15dp"
        android:hint="Search"
        android:textSize="22sp" />

    <androidx.recyclerview.widget.RecyclerView
        android:id="@+id/recyclerView"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:layout_below="@+id/txt_searchtext"
        android:scrollbars="vertical" />

    <TextView
        android:id="@+id/txt_no_results_found"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_below="@+id/txt_searchtext"
        android:gravity="center"
        android:text="No Results Found"
        android:textSize="24sp"
        android:visibility="gone" />

</RelativeLayout>

```

fragment_notifications.xml

```

<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout 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">

<RelativeLayout
    android:id="@+id/rellay1"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:background="@drawable/grad"
    android:paddingBottom="20dp">

    <RelativeLayout
        android:id="@+id/imgUser"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_centerHorizontal="true"
        android:layout_marginTop="20dp"
        android:background="@drawable/circle_border">

        <androidx.cardview.widget.CardView
            android:id="@+id/myCardView"
            android:layout_width="100dp"
            android:layout_height="100dp"
            app:cardCornerRadius="50dp"
            android:layout_centerHorizontal="true"
            android:layout_centerVertical="true"
            >
            <ImageView
                android:id="@+id/photo_imageView"
                android:layout_width="match_parent"
                android:layout_height="match_parent"
                android:scaleType="centerCrop"
                android:src="@drawable/profile"
            />
        </androidx.cardview.widget.CardView>
    </RelativeLayout>

    <TextView
        android:id="@+id/txt_name"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_below="@+id/imgUser"
        android:layout_centerHorizontal="true"
        android:layout_marginTop="15sp"
        android:fontFamily="sans-serif-light"
        android:text="Vivek Patel"
        android:textColor="@color/design_default_color_background"
        android:textSize="32sp" />

</RelativeLayout>

<LinearLayout
    android:id="@+id/linlay1"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_below="@id/rellay1"
    android:orientation="horizontal"/>

```

```

<RelativeLayout
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:layout_below="@+id/linlay1">

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginLeft="30dp"
        android:layout_marginTop="40dp"
        android:layout_marginRight="30dp"
        android:orientation="vertical">

        <LinearLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:orientation="vertical">

            <TextView
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:text="Email"
                android:textColor="@color/colorPrimaryDark"
                android:textSize="13sp" />

            <TextView
                android:id="@+id/txt_email"
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:fontFamily="sans-serif-light"
                android:textColor="#000000"
                android:textSize="18sp"
                tools:text="vivek.patel@mohawkcollege.ca" />

            <View
                android:layout_width="match_parent"
                android:layout_height="1dp"
                android:layout_marginTop="15dp"
                android:background="@color/colorPrimary" />

        </LinearLayout>

        <LinearLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_marginTop="15dp"
            android:orientation="vertical">

            <TextView
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:text="Phone"
                android:textColor="@color/colorPrimaryDark"
                android:textSize="13sp" />

            <TextView

```

```

        android:id="@+id/txt_phone"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:fontFamily="sans-serif-light"
        android:textColor="#000000"
        android:textSize="18sp"
        tools:text="1111111111" />

    <View
        android:layout_width="match_parent"
        android:layout_height="1dp"
        android:layout_marginTop="15dp"
        android:background="@color/colorPrimary" />

</LinearLayout>

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginTop="15dp"
    android:orientation="vertical">

    <TextView
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Verify Email"
        android:textColor="@color/colorPrimaryDark"
        android:textSize="13sp" />

    <LinearLayout
        android:id="@+id/btnVerifyEmail"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:onClick="btnVerifyEmail"
        android:orientation="horizontal">

        <ImageView
            android:id="@+id/verify_imageview"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_marginTop="5dp"
            android:layout_marginEnd="10dp"
            app:layout_constraintEnd_toEndOf="parent"
            app:layout_constraintHorizontal_bias="0.126"
            app:layout_constraintStart_toStartOf="parent"

app:layout_constraintTop_toBottomOf="@+id/email_imageview"

app:srcCompat="@drawable/ic_baseline_cancel_presentation_24" />

        <ImageView
            android:id="@+id/verified_imageview"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_marginTop="5dp"
            android:layout_marginEnd="10dp"
            app:layout_constraintEnd_toEndOf="parent"

```

```

        app:layout_constraintHorizontal_bias="0.126"
        app:layout_constraintStart_toStartOf="parent"

app:layout_constraintTop_toBottomOf="@+id/email_imageview"
        app:srcCompat="@drawable/ic_baseline_check_24" />

        <TextView
            android:id="@+id/txt_verify"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_marginTop="5dp"
            android:text="Verify Now"
            android:textColor="#E9000F"
            android:textSize="18sp"
            app:layout_constraintEnd_toEndOf="parent"
            app:layout_constraintHorizontal_bias="0.031"

app:layout_constraintStart_toEndOf="@+id/phone_imageview"
app:layout_constraintTop_toBottomOf="@+id/email_textview" />

        <TextView
            android:id="@+id/txt_verified"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_marginTop="5dp"
            android:text="Verified"
            android:textColor="#249E00"
            android:textSize="18sp"
            app:layout_constraintEnd_toEndOf="parent"
            app:layout_constraintHorizontal_bias="0.031"

app:layout_constraintStart_toEndOf="@+id/phone_imageview"
app:layout_constraintTop_toBottomOf="@+id/email_textview" />
    </LinearLayout>

    <View
        android:layout_width="match_parent"
        android:layout_height="1dp"
        android:layout_marginTop="15dp"
        android:background="@color/colorPrimary" />

    </LinearLayout>

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:onClick="btnResetPassword"
        android:orientation="horizontal">

        <ImageView
            android:id="@+id/reset_imageview"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_marginTop="20dp"

```

```

        android:layout_marginEnd="10dp"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.126"
        app:layout_constraintStart_toStartOf="parent"

app:layout_constraintTop_toBottomOf="@+id/email_imageview"
    app:srcCompat="@drawable/ic_baseline_autorenew_24" />

    <TextView
        android:id="@+id/rest_textview"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="20dp"
        android:text="Reset Password"
        android:textSize="18sp"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.031"
        app:layout_constraintStart_toEndOf="@+id/phone_imageview"
        app:layout_constraintTop_toBottomOf="@+id/email_textview"
    />

</LinearLayout>

<View
    android:layout_width="match_parent"
    android:layout_height="1dp"
    android:layout_marginTop="15dp"
    android:layout_weight="1"
    android:background="@color/colorPrimary" />

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:onClick="btnLogout"
    android:orientation="horizontal">

    <ImageView
        android:id="@+id/logout_imageview"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="20dp"
        android:layout_marginEnd="10dp"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.126"
        app:layout_constraintStart_toStartOf="parent"

app:layout_constraintTop_toBottomOf="@+id/email_imageview"
    app:srcCompat="@drawable/ic_baseline_exit_to_app_24" />

    <TextView
        android:id="@+id/logout_textview"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="20dp"
        android:text="Logout"
        android:textSize="18sp"

```

```

        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.031"
        app:layout_constraintStart_toEndOf="@+id/phone_imageview"
        app:layout_constraintTop_toBottomOf="@+id/email_textview"
    />

    </LinearLayout>

    <View
        android:layout_width="match_parent"
        android:layout_height="1dp"
        android:layout_marginTop="15dp"
        android:layout_weight="1"
        android:background="@color/colorPrimary" />

    </LinearLayout>

</RelativeLayout>

</RelativeLayout>

```

recycler_row_item.xml

```

<?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="wrap_content"
    android:orientation="vertical"
    android:padding="10dp">

    <androidx.cardview.widget.CardView
        android:id="@+id/myCardView"
        android:layout_width="wrap_content"
        android:layout_height="250dp"
        android:layout_margin="4dp"
        app:cardElevation="3dp"
        app:cardUseCompatPadding="true">

        <LinearLayout
            android:layout_width="match_parent"
            android:layout_height="match_parent"
            android:orientation="vertical"
            android:weightSum="5.5">

            <ImageView
                android:id="@+id/ivImage"
                android:layout_width="match_parent"

```



```

        android:layout_height="0dp"
        android:layout_weight="4"
        android:scaleType="centerCrop"
        app:srcCompat="@drawable/image" />

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="0dp"
    android:layout_weight="1.5"
    android:orientation="horizontal"
    android:weightSum="2">

    <LinearLayout
        android:layout_width="0dp"
        android:layout_height="match_parent"
        android:layout_weight="1.3"
        android:padding="10dp"
        android:orientation="vertical">

        <TextView
            android:id="@+id/tvTitle"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:maxLines="1"
            android:text="Title"
            android:textSize="18sp"
            android:textStyle="bold" />

        <TextView
            android:id="@+id/tvDescription"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:maxLines="1"
            android:text="Description"
            android:textSize="18sp" />
    </LinearLayout>

    <LinearLayout
        android:layout_width="0dp"
        android:layout_height="match_parent"
        android:layout_weight=".7"
        android:gravity="center">

        <TextView
            android:id="@+id/tvPrice"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:text="$ 30"
            android:textAlignment="center"
            android:textColor="@color/colorPrimary"
            android:textSize="25sp"
            android:textStyle="bold" />
    </LinearLayout>
</LinearLayout>

```

```
</androidx.cardview.widget.CardView>  
</LinearLayout>
```


bottom_nav_menu.xml


```
<?xml version="1.0" encoding="utf-8"?>  
<menu xmlns:android="http://schemas.android.com/apk/res/android">  
  
    <item  
        android:id="@+id/navigation_home"  
        android:icon="@drawable/ic_home_black_24dp"  
        android:title="@string/title_home" />  
  
    <item  
        android:id="@+id/navigation_dashboard"  
        android:icon="@drawable/ic_dashboard_black_24dp"  
        android:title="@string/title_dashboard" />  
  
    <item  
        android:id="@+id/favourite_dashboard"  
        android:icon="@drawable/ic_baseline_favorite"  
        android:title="@string/favorites" />  
  
    <item  
        android:id="@+id/navigation_notifications"  
        android:icon="@drawable/ic_baseline_person_24"  
        android:title="@string/title_notifications" />  
  
</menu>
```

Some output images

FoodRecipeApp



 Username

 Password

LOGIN

[Forgot Password?](#)

Don't have an account yet? [SIGN UP](#)


GUEST LOGIN

FoodRecipeApp



 Full Name

 Phone

 Email

 Password

REGISTER

Already have an account? [SIGN IN](#)

Home

Search

**Samosa**

A samosa (/sə'mousə/) is

15 mins**Chaat - Dahi Puri**

Chaat is a broad term

20 mins

Home



FoodRecipeApp

**Samosa**

A samosa (/sə'mousə/) is a fried or baked pastry with a savoury filling, such as spiced potatoes, onions, peas, cheese, beef and other meats, or lentils.

Ingredients:

- 1 tablespoon vegetable oil
- ½ cup chopped onion
- 3 (19 ounce) cans garbanzo beans, drained
- 2 tablespoons curry paste
- ½ cup apple juice
- 3 sheets frozen puff pastry, thawed
- ¼ cup all-purpose flour for dusting

Directions:**Step 1**

Preheat oven to 350 degrees F (175

degrees C).

Step 2

Heat oil in a large skillet over medium-high heat. Saute onion until browned, about 5 minutes. Reduce heat and add garbanzo beans. Mix curry paste with apple juice until smooth, then add to skillet. Simmer mixture 10 minutes, stirring occasionally and adding more apple juice or water as needed to keep mixture moist.

Step 3

Cut each sheet of pastry into 3 equal-size rectangles, then cut each rectangle in half. (You will have 18 pieces total.) Dust a clean, flat surface lightly with flour, and roll each piece of pastry until it is double in size. Dust pastry with flour as needed, to prevent it from sticking to rolling pin. Spoon an equal amount of filling into center of each pastry square. Fold in half and press edges to seal. Place on nonstick baking sheets.

Step 4

Bake in preheated oven for 25 minutes, until golden brown.

15 mins

Favorites

Search



Chaat - Dahi Puri

Chaat is a broad term

20 mins



Cucumber Raita

This easy cucumber raita

10 mins

FoodRecipeApp



Cucumber Raita



This easy cucumber raita is flavored with fresh mint, fresh cilantro, and a touch of ground cumin. You'll be making this often!

Ingredients:

2 (7 ounce) containers Greek yogurt
1 cucumber, peeled and diced
2 tablespoons chopped fresh mint
2 tablespoons chopped fresh cilantro
½ teaspoon ground cumin
salt and ground black pepper to taste

Directions:


Step 1

Combine Greek yogurt, cucumber, mint, cilantro, cumin, salt, and pepper in a bowl; mix well.



Favorites



My Recipes	FoodRecipeApp
<p>Search</p> <hr/>	 <p>Recipe Name</p> <hr/> <p>Description</p> <hr/> <p>Cook Time</p> <hr/>
<p>Home icon, My Recipes icon, Heart icon, User icon</p>	<p>SELECT IMAGE</p> <p>UPLOAD RECIPE</p>

Home

cha



Chaat - Dahi Puri

Chaat is a broad term

20 mins

