APP DEVELOPMENT

Lab Programs WEEK(10-12)

A.BINDU PRIYANKA

18131A0501

CSE-1

10.Write a Mobile Application that use fragmentation.

JAVA File:

```
package com.example.fragment;

import android.app.Fragment;
import android.app.FragmentManager;
import android.app.FragmentTransaction;
import androidx.appcompat.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;

public class MainActivity extends AppCompatActivity {
   Button firstFragment, secondFragment;

   @Override
   protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        // get the reference of Button's
        firstFragment = (Button) findViewById(R.id.firstFragment);
        secondFragment = (Button) findViewById(R.id.secondFragment);

// perform setOnClickListener event on First Button
        firstFragment.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
// load First Fragment
                loadFragment(new FirstFragment());
            }
        });
// perform setOnClickListener event on Second Button
        secondFragment.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
```

```
// load Second Fragment
                loadFragment(new FirstFragment());
            }
        });

    }

    private void loadFragment(Fragment fragment) {
// create a FragmentManager
        FragmentManager fm = getFragmentManager();
// create a FragmentTransaction to begin the transaction and replace the
Fragment
        FragmentTransaction fragmentTransaction = fm.beginTransaction();
// replace the FrameLayout with new Fragment
        fragmentTransaction.replace(R.id.frameLayout, fragment);
        fragmentTransaction.commit(); // save the changes
    }
}
```

ANDROID Manifest XML File:

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

    <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/Theme.Fragment">
        <activity android:name=".MainActivity">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

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

</manifest>
```

XML File:

```
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
```

```
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    tools:context=".MainActivity">
    <!-- display two Button's and a FrameLayout to replace the Fragment's  -->
    <Button
        android:id="@+id/firstFragment"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="First Fragment"
        android:textColor="@color/white"
        android:textSize="20sp" />

    <Button
        android:id="@+id/secondFragment"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginTop="10dp"
        android:text="Second Fragment"
        android:textColor="@color/white"
        android:textSize="20sp" />
    <TextView
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_margin="20dp"
        android:gravity="center"
        android:text="18131A0501"
        android:textSize="25sp" />
    <FrameLayout
        android:id="@+id/frameLayout"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:layout_marginTop="10dp" />
</LinearLayout>
```

First Fragment JAVA File:

```
package com.example.fragment;

import android.app.Fragment;
import android.os.Bundle;
import android.widget.Button;
import android.widget.Toast;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
```

```
public class FirstFragment extends Fragment {
   View view;
   Button firstButton;

   @Override
   public View onCreateView(LayoutInflater inflater, ViewGroup container,
                            Bundle savedInstanceState) {
// Inflate the layout for this fragment
        view = inflater.inflate(R.layout.fragment_first, container, false);
// get the reference of Button
        firstButton = (Button) view.findViewById(R.id.firstButton);
// perform setOnClickListener on first Button
        firstButton.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
// display a message by using a Toast
                Toast.makeText(getActivity(), "First Fragment",
Toast.LENGTH_LONG).show();
            }
        });
        return view;
   }
}
```

First Fragment XML File:

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
   xmlns:tools="http://schemas.android.com/tools"
   android:layout_width="match_parent"
   android:layout_height="match_parent"
   tools:context=".FirstFragment">

   <!--TextView and Button displayed in First Fragment -->
   <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_centerHorizontal="true"
        android:layout_marginTop="100dp"
        android:text="This is First Fragment"
        android:textColor="@color/black"
        android:textSize="25sp" />

   <Button
        android:id="@+id/firstButton"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:layout_centerInParent="true"
        android:layout_marginLeft="20dp"
```

```
        android:layout_marginRight="20dp"
        android:text="First Fragment"
        android:textColor="@color/white"
        android:textSize="20sp"
        android:textStyle="bold" />
</RelativeLayout>
```

Fragment_Second JAVA File:

```
package com.example.fragment;

import android.os.Bundle;
import androidx.fragment.app.Fragment;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.Button;
import android.widget.Toast;

class SecondFragment extends Fragment {
   View view;
   Button secondButton;

   @Override
   public View onCreateView(LayoutInflater inflater, ViewGroup container,
                            Bundle savedInstanceState) {
// Inflate the layout for this fragment
        view = inflater.inflate(R.layout.fragment_second, container, false);
// get the reference of Button
        secondButton = (Button) view.findViewById(R.id.secondButton);
// perform setOnClickListener on second Button
        secondButton.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
// display a message by using a Toast
                Toast.makeText(getActivity(), "Second Fragment",
Toast.LENGTH_LONG).show();
            }
        });
        return view;
   }
}
```

Fragment_Second XML File:

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

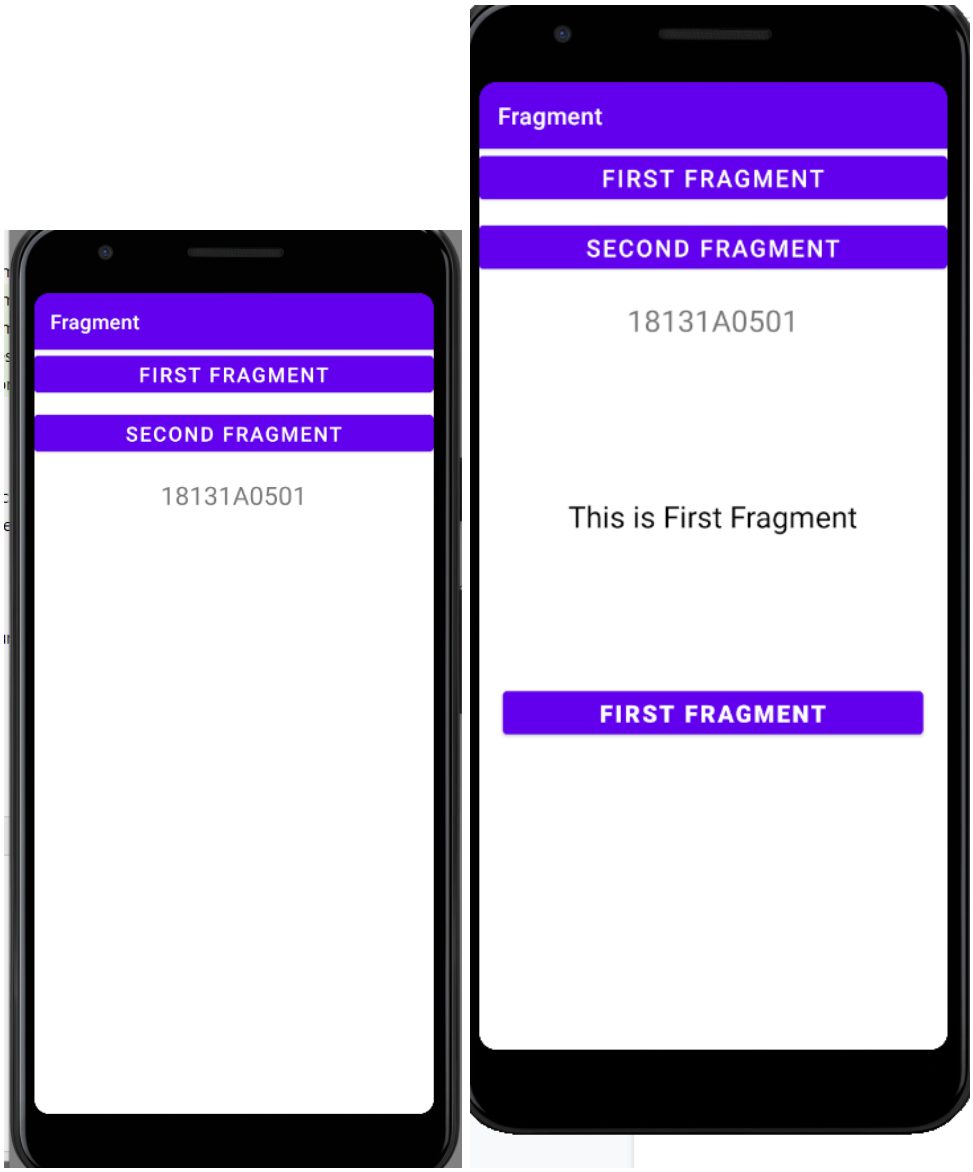
```
    android:layout_height="match_parent"
    tools:context=".SecondFragment">

    <!--TextView and Button displayed in Second Fragment -->
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_centerHorizontal="true"
        android:layout_marginTop="100dp"
        android:text="This is Second Fragment"
        android:textColor="@color/black"
        android:textSize="25sp" />

    <Button
        android:id="@+id/secondButton"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:layout_centerInParent="true"
        android:layout_marginLeft="20dp"
        android:layout_marginRight="20dp"
        android:text="Second Fragment"
        android:textColor="@color/white"
        android:textSize="20sp"
        android:textStyle="bold" />

</RelativeLayout>
```

Output:



11.Develop a native application that uses GPS Location Information.

GPS Tracker JAVA File:

```
package com.example.exp11;

import android.annotation.SuppressLint;
```

```
import android.app.AlertDialog;
import android.app.Service;
import android.content.Context;
import android.content.DialogInterface;
import android.content.Intent;
import android.location.Location;
import android.location.LocationListener;
import android.location.LocationManager;
import android.os.Bundle;
import android.os.IBinder;
import android.provider.Settings;
import android.util.Log;

public class GPSTracker extends Service implements LocationListener {

    private final Context mContext;

    // flag for GPS status
    boolean isGPSEnabled = false;

    // flag for network status
    boolean isNetworkEnabled = false;

    // flag for GPS status
    boolean canGetLocation = false;

    Location location; // location
    double latitude; // latitude
    double longitude; // longitude

    // The minimum distance to change Updates in meters
    private static final long MIN_DISTANCE_CHANGE_FOR_UPDATES = 10; // 10
meters

    // The minimum time between updates in milliseconds
    private static final long MIN_TIME_BW_UPDATES = 1000 * 60 * 1; // 1 minute

    // Declaring a Location Manager
    protected LocationManager locationManager;

    public GPSTracker(Context context) {
        this.mContext = context;
        getLocation();
    }

    @SuppressLint("MissingPermission")
    public Location getLocation() {
        try {
            locationManager = (LocationManager)
mContext.getSystemService(LOCATION_SERVICE);

            // getting GPS status
            isGPSEnabled =
locationManager.isProviderEnabled(LocationManager.GPS_PROVIDER);

            // getting network status
            isNetworkEnabled = locationManager
```

```
                    .isProviderEnabled(LocationManager.NETWORK_PROVIDER);

            if (!isGPSEnabled && !isNetworkEnabled) {
                // no network provider is enabled
            } else {
                this.canGetLocation = true;
                // First get location from Network Provider
                if (isNetworkEnabled) {
                    locationManager.requestLocationUpdates(
                            LocationManager.NETWORK_PROVIDER,
                            MIN_TIME_BW_UPDATES,
                            MIN_DISTANCE_CHANGE_FOR_UPDATES, this);

                    Log.d("Network", "Network");
                    if (locationManager != null) {
                        location = locationManager

.getLastKnownLocation(LocationManager.NETWORK_PROVIDER);

                        if (location != null) {
                            latitude = location.getLatitude();
                            longitude = location.getLongitude();
                        }
                    }
                }

                // if GPS Enabled get lat/long using GPS Services
                if (isGPSEnabled) {
                    if (location == null) {
                        locationManager.requestLocationUpdates(
                                LocationManager.GPS_PROVIDER,
                                MIN_TIME_BW_UPDATES,
                                MIN_DISTANCE_CHANGE_FOR_UPDATES, this);

                        Log.d("GPS Enabled", "GPS Enabled");
                        if (locationManager != null) {
                            location = locationManager

.getLastKnownLocation(LocationManager.GPS_PROVIDER);

                            if (location != null) {
                                latitude = location.getLatitude();
                                longitude = location.getLongitude();
                            }
                        }
                    }
                }
            }

        } catch (Exception e) {
            e.printStackTrace();
        }

        return location;
    }

    /**
```

```
     * Stop using GPS listener
     * Calling this function will stop using GPS in your app
     * */

    public void stopUsingGPS(){
        if(locationManager != null){
            locationManager.removeUpdates(GPSTracker.this);
        }
    }

    /**
     * Function to get latitude
     * */

    public double getLatitude(){
        if(location != null){
            latitude = location.getLatitude();
        }

        // return latitude
        return latitude;
    }

    /**
     * Function to get longitude
     * */

    public double getLongitude(){
        if(location != null){
            longitude = location.getLongitude();
        }

        // return longitude
        return longitude;
    }

    /**
     * Function to check GPS/wifi enabled
     * @return boolean
     * */

    public boolean canGetLocation() {
        return this.canGetLocation;
    }

    /**
     * Function to show settings alert dialog
     * On pressing Settings button will lauch Settings Options
     * */

    public void showSettingsAlert(){
        AlertDialog.Builder alertDialog = new AlertDialog.Builder(mContext);

        // Setting Dialog Title
        alertDialog.setTitle("GPS is settings");

        // Setting Dialog Message
```

```
        alertDialog.setMessage("GPS is not enabled. Do you want to go to
settings menu?");

        // On pressing Settings button
        alertDialog.setPositiveButton("Settings", new
DialogInterface.OnClickListener() {
            public void onClick(DialogInterface dialog,int which) {
                Intent intent = new
Intent(Settings.ACTION_LOCATION_SOURCE_SETTINGS);
                mContext.startActivity(intent);
            }
        });

        // on pressing cancel button
        alertDialog.setNegativeButton("Cancel", new
DialogInterface.OnClickListener() {
            public void onClick(DialogInterface dialog, int which) {
                dialog.cancel();
            }
        });

        // Showing Alert Message
        alertDialog.show();
    }

    @Override
    public void onLocationChanged(Location location) {
    }

    @Override
    public void onProviderDisabled(String provider) {
    }

    @Override
    public void onProviderEnabled(String provider) {
    }

    @Override
    public void onStatusChanged(String provider, int status, Bundle extras) {
    }

    @Override
    public IBinder onBind(Intent arg0) {
        return null;
    }
```

Android Manifest File:

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.example.exp11">

    <uses-permission android:name="android.permission.ACCESS_COARSE_LOCATION"
/>
```

```
    <uses-permission android:name="android.permission.ACCESS_FINE_LOCATION" />
    <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/Theme.Exp11">

        <activity android:name=".MainActivity">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

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

</manifest>
```

MainActivity JAVA File:

```
package com.example.exp11;
import android.Manifest;
import android.app.Activity;
import android.content.pm.PackageManager;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.Toast;

import androidx.core.app.ActivityCompat;

public class MainActivity extends Activity {

    Button btnShowLocation;
    private static final int REQUEST_CODE_PERMISSION = 2;
    String mPermission = Manifest.permission.ACCESS_FINE_LOCATION;

    // GPSTracker class
    GPSTracker gps;
    private PackageManager MockPackageManager;

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

        try {
            if (ActivityCompat.checkSelfPermission(this, mPermission)
                    != MockPackageManager.PERMISSION_GRANTED) {

                ActivityCompat.requestPermissions(this, new
String[]{mPermission},
                        REQUEST_CODE_PERMISSION);
```

```
            }
        } catch (Exception e) {
            e.printStackTrace();
        }

        btnShowLocation = (Button) findViewById(R.id.button);

        // show location button click event
        btnShowLocation.setOnClickListener(new View.OnClickListener() {

            @Override
            public void onClick(View arg0) {
                // create class object
                gps = new GPSTracker(MainActivity.this);

                // check if GPS enabled
                if(gps.canGetLocation()){

                    double latitude = gps.getLatitude();
                    double longitude = gps.getLongitude();

                    // \n is for new line
                    Toast.makeText(getApplicationContext(), "Your Location is
- \nLat: "
                            + latitude + "\nLong: " + longitude,
Toast.LENGTH_LONG).show();
                }else{
                    // can't get location
                    // GPS or Network is not enabled
                    // Ask user to enable GPS/network in settings
                    gps.showSettingsAlert();
                }

            }
        });
    }
}
```

XML File:

```
<?xml version = "1.0" encoding = "utf-8"?>
<LinearLayout xmlns:android = "http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width = "fill_parent"
    android:layout_height = "fill_parent"
    android:orientation = "vertical" >


    <Button
        android:id = "@+id/button"
        android:layout_width = "fill_parent"
        android:layout_height = "wrap_content"
        android:text = "getlocation"/>
```

```
    <TextView
        android:layout_width="131dp"
        android:layout_height="wrap_content"
        android:layout_x="37dp"
        android:layout_y="614dp"
        android:text="18131A0501"
        android:textAlignment="center"
        android:textColorHighlight="#FF9800"
        android:textSize="20sp"
        tools:ignore="DuplicateIds"
        android:gravity="center_horizontal" />
    <TextView
        android:layout_width="131dp"
        android:layout_height="wrap_content"
        android:textAlignment="center"
        android:layout_x="37dp"
        android:layout_y="614dp"
        android:text="Bindu Priyanka"
        android:textColorHighlight="#ff0000"
        android:textSize="20sp"
        android:gravity="center_horizontal" />


</LinearLayout>
```

Output:

12.Develop a native application that uses Google Map Services.

JAVA File:

```
package com.example.ex12;

import androidx.core.app.ActivityCompat;
import androidx.fragment.app.FragmentActivity;
import android.Manifest;
import android.content.pm.PackageManager;
import android.location.Location;
import android.location.LocationListener;
import android.location.LocationManager;
import android.os.Bundle;
import com.google.android.gms.maps.CameraUpdateFactory;
import com.google.android.gms.maps.GoogleMap;
import com.google.android.gms.maps.OnMapReadyCallback;
```

```
import com.google.android.gms.maps.SupportMapFragment;
import com.google.android.gms.maps.model.LatLng;
import com.google.android.gms.maps.model.MarkerOptions;


public class MapsActivity extends FragmentActivity implements
OnMapReadyCallback {

    private GoogleMap mMap;

    private LocationListener locationListener;
    private LocationManager locationManager;


    private final long MIN_TIME = 1000;// 1 second
    private final long MIN_DIST = 5;// 5 Meters

    private LatLng latLng;


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

        // Obtain the SupportMapFragment and get notified when the map is
ready to be used.
        SupportMapFragment mapFragment = (SupportMapFragment)
getSupportFragmentManager()
                .findFragmentById(R.id.map);
        mapFragment.getMapAsync(this);
        ActivityCompat.requestPermissions(this, new
String[]{Manifest.permission.ACCESS_FINE_LOCATION},
PackageManager.PERMISSION_GRANTED);
        ActivityCompat.requestPermissions(this, new
String[]{Manifest.permission.ACCESS_COARSE_LOCATION},
PackageManager.PERMISSION_GRANTED);
    }

    /**
     * Manipulates the map once available.
     * This callback is triggered when the map is ready to be used.
     * This is where we can add markers or lines, add listeners or move the
camera. In this case,
     * we just add a marker near Sydney, Australia.
     * If Google Play services is not installed on the device, the user will
be prompted to install
     * it inside the SupportMapFragment. This method will only be triggered
once the user has
     * installed Google Play services and returned to the app.
     */
    @Override
    public void onMapReady(GoogleMap googleMap) {
        mMap = googleMap;

        // Add a marker in Sydney and move the camera
```

```
            LatLng sydney = new LatLng(-34, 151);
            mMap.addMarker(new MarkerOptions().position(sydney).title("Marker
in Sydney"));
            mMap.moveCamera(CameraUpdateFactory.newLatLng(sydney));


        locationListener = new LocationListener() {

            @Override
            public void onLocationChanged(Location location) {
                try {
                    latLng = new
LatLng(location.getLatitude(),location.getLongitude());
                    mMap.addMarker(new
MarkerOptions().position(latLng).title("My Position"));
                    mMap.moveCamera(CameraUpdateFactory.newLatLng(latLng));
                }
                catch(SecurityException e){
                    e.printStackTrace();
                }

            }

            public void onStatusChanged(String s, int i, Bundle bundle) {

            }

            @Override
            public void onProviderEnabled(String s) {

            }

            @Override
            public void onProviderDisabled(String s) {

            }
        };

        locationManager = (LocationManager) getSystemService(LOCATION_SERVICE);
        try {

locationManager.requestLocationUpdates(LocationManager.GPS_PROVIDER,MIN_TIME,MI
N_DIST,locationListener);
        }
        catch (SecurityException e)
        {
            e.printStackTrace();
        }

    }
}
```

API XML File:

```
<resources>
    <string name="google_maps_key" templateMergeStrategy="preserve"
```

```
translatable="false">AIzaSyA8BPMoJ_B3AjuAEQWaCN1uvgVxM5ueVJo</string>
</resources>
```

Android Manifest File:

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.example.ex12">

       <uses-permission android:name="android.permission.ACCESS_FINE_LOCATION"
/>
    <uses-permission
android:name="android.permission.ACCESS_COARSE_LOCATION"/>

    <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/Theme.Ex12">


        <meta-data
            android:name="com.google.android.geo.API_KEY"
            android:value="@string/google_maps_key" />

        <activity
            android:name=".MapsActivity"
            android:label="@string/title_activity_maps">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

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

</manifest>
```

XML File:

Output: