Java Executable Code:

1. Splash Screen Activity

package com.pkg.smartdrive;  
  
import androidx.appcompat.app.AppCompatActivity;  
  
import android.content.Intent;  
import android.os.Bundle;  
import android.os.Handler;  
import android.view.WindowManager;  
  
public class SplashScreenActivity extends AppCompatActivity {  
 private static int *SPLASH\_SCREEN*=4000;//Delay to be added to the Splash Screen  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 getWindow().setFlags(WindowManager.LayoutParams.*FLAG\_FULLSCREEN*,WindowManager.LayoutParams.*FLAG\_FULLSCREEN*);//hiding status bar  
 setContentView(R.layout.*activity\_splashscreen*);  
 //Handler Method to add delay between Splash Screen and Main Screen  
 new Handler().postDelayed(new Runnable() {  
 @Override  
 public void run() {  
 Intent intent=new Intent(SplashScreenActivity.this, SignUpActivity.class);//To move from Splash Screen to Main Screen  
 startActivity(intent);  
 finish();  
 }  
 },*SPLASH\_SCREEN*);  
 }  
}

1. Signup Activity

package com.pkg.smartdrive;  
  
import androidx.appcompat.app.AppCompatActivity;  
  
import android.content.Intent;  
import android.os.Bundle;  
import android.view.View;  
import android.widget.Button;  
import android.widget.EditText;  
  
import com.hbb20.CountryCodePicker;  
  
public class SignUpActivity extends AppCompatActivity {  
  
 CountryCodePicker ccp\_input;  
 EditText editText\_Name,editText\_Phone;  
 Button signUp\_Button;  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_signup*);  
  
 //typecasting into variables  
 editText\_Name=(EditText)findViewById(R.id.*editText\_Name*);  
 editText\_Phone=(EditText)findViewById(R.id.*editText\_Phone*);  
 ccp\_input=(CountryCodePicker)findViewById(R.id.*ccp\_input*);  
 ccp\_input.registerCarrierNumberEditText(editText\_Phone);  
 signUp\_Button=(Button)findViewById(R.id.*signUp\_Button*);  
  
 //On Button Click  
 signUp\_Button.setOnClickListener(new View.OnClickListener() {  
 //Otp calling  
 @Override  
 public void onClick(View v) {  
 //TO-DO : Permissions for contact-access, geolocation, messageing,etc.  
  
  
 //For moving to next Activity (GetOTPActivity)  
 Intent intent=new Intent(SignUpActivity.this, GetOTPActivity.class);  
 intent.putExtra("mobile\_no", ccp\_input.getFullNumberWithPlus().replace(" ",""));  
 startActivity(intent);  
 }  
 });  
 }  
}

1. Otp Activity

package com.pkg.smartdrive;  
  
import androidx.annotation.NonNull;  
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.EditText;  
import android.widget.TextView;  
import android.widget.Toast;  
  
import com.google.android.gms.tasks.OnCompleteListener;  
import com.google.android.gms.tasks.Task;  
import com.google.firebase.FirebaseException;  
import com.google.firebase.FirebaseTooManyRequestsException;  
import com.google.firebase.auth.AuthResult;  
import com.google.firebase.auth.FirebaseAuth;  
import com.google.firebase.auth.FirebaseAuthInvalidCredentialsException;  
import com.google.firebase.auth.PhoneAuthCredential;  
import com.google.firebase.auth.PhoneAuthProvider;  
  
import java.util.concurrent.TimeUnit;  
  
import java.util.concurrent.TimeUnit;  
  
public class GetOTPActivity extends AppCompatActivity {  
  
 EditText Otp\_EditText;  
 Button button\_VerifyOTP;  
 String phone\_number;  
 String Otp\_Id;  
 FirebaseAuth mAuth;  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_getotp*);  
  
 //typecasting into variables  
 phone\_number=getIntent().getStringExtra("mobile\_no").toString();  
 Otp\_EditText=(EditText)findViewById(R.id.*Otp\_EditText*);  
 button\_VerifyOTP=(Button)findViewById(R.id.*button\_VerifyOTP*);  
 mAuth=FirebaseAuth.*getInstance*();  
 initiate\_OTP();  
  
 button\_VerifyOTP.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View v) {  
 if (Otp\_EditText.getText().toString().isEmpty())  
 Toast.*makeText*(getApplicationContext(), "Fields are Empty", Toast.*LENGTH\_LONG*).show();  
 else if (Otp\_EditText.getText().toString().length() != 6)  
 Toast.*makeText*(getApplicationContext(), "OTP is Invalid!!!", Toast.*LENGTH\_LONG*).show();  
 else {  
 PhoneAuthCredential credential = PhoneAuthProvider.*getCredential*(Otp\_Id, Otp\_EditText.getText().toString());  
 signInWithPhoneAuthCredential(credential);  
 }  
 }  
 });  
 }  
  
 private void initiate\_OTP() {  
 PhoneAuthProvider.*getInstance*().verifyPhoneNumber(  
 phone\_number, // Phone number to verify  
 60, // Time Out Duration  
 TimeUnit.*SECONDS*, // Time Out unit  
 this, // Activity (for callback binding)  
 new PhoneAuthProvider.OnVerificationStateChangedCallbacks() {  
 @Override  
 public void onCodeSent(@NonNull String s, @NonNull PhoneAuthProvider.ForceResendingToken forceResendingToken) {  
 Otp\_Id=s;  
 }  
  
 @Override  
 public void onVerificationCompleted(@NonNull PhoneAuthCredential phoneAuthCredential) {  
 signInWithPhoneAuthCredential(phoneAuthCredential);  
  
 }  
  
 @Override  
 public void onVerificationFailed(@NonNull FirebaseException e) {  
 Toast.*makeText*(getApplicationContext(),e.getMessage(),Toast.*LENGTH\_LONG*).show();  
 }  
 }  
  
 );  
 }  
 private void signInWithPhoneAuthCredential(PhoneAuthCredential credential) {  
 mAuth.signInWithCredential(credential)  
 .addOnCompleteListener(this, new OnCompleteListener<AuthResult>() {  
 @Override  
 public void onComplete(@NonNull Task<AuthResult> task) {  
 if (task.isSuccessful()) {  
 startActivity(new Intent(GetOTPActivity.this,ArdiunoIPActivity.class));  
 finish();  
 }  
 else {  
 Toast.*makeText*(getApplicationContext(),"Sign In error",Toast.*LENGTH\_LONG*).show();  
 }  
 }  
 });  
 }  
}

1. Adriuno Activity

package com.pkg.smartdrive;  
  
import androidx.appcompat.app.AppCompatActivity;  
  
import android.app.Dialog;  
import android.content.Intent;  
import android.os.Bundle;  
import android.util.Log;  
import android.view.View;  
import android.widget.Button;  
import android.widget.Toast;  
  
import com.google.android.gms.common.ConnectionResult;  
import com.google.android.gms.common.GoogleApiAvailability;  
  
public class ArdiunoIPActivity extends AppCompatActivity {  
  
 private static final String *TAG* = "MainActivity";  
 private static final int *ERROR\_DIALOG\_REQUEST*=9001;  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_ardiunoip*);  
 if(isServiceOK()){  
 init();  
 }  
 }  
  
 private void init(){  
 Button btn=(Button)findViewById(R.id.*btn*);  
 btn.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View v) {  
 Intent intent=new Intent(ArdiunoIPActivity.this,MapActivity.class);  
 startActivity(intent);  
 }  
 });  
 }  
  
  
 public boolean isServiceOK(){  
 Log.*d*(*TAG*,"isServicesOK : checking google services version");  
 int avaiable= GoogleApiAvailability.*getInstance*().isGooglePlayServicesAvailable(ArdiunoIPActivity.this);  
 if(avaiable== ConnectionResult.*SUCCESS*){  
 //Everything is ok  
 Log.*d*(*TAG*,"google play services are up to date");  
 return true;  
 }  
 else if(GoogleApiAvailability.*getInstance*().isUserResolvableError(avaiable)){  
 Log.*d*(*TAG*,"Fixable Error");  
 Dialog dialog=GoogleApiAvailability.*getInstance*().getErrorDialog(ArdiunoIPActivity.this,avaiable,*ERROR\_DIALOG\_REQUEST*);  
 dialog.show();  
 }  
 else{  
 Toast.*makeText*(this,"Unsolvable Error",Toast.*LENGTH\_SHORT*).show();  
 }  
 return false;  
 }  
}

1. Map Activity

package com.pkg.smartdrive;  
  
import androidx.annotation.NonNull;  
import androidx.appcompat.app.AppCompatActivity;  
import androidx.core.app.ActivityCompat;  
import androidx.core.content.ContextCompat;  
  
import android.Manifest;  
import android.content.Intent;  
import android.content.pm.PackageManager;  
import android.location.Address;  
import android.location.Geocoder;  
import android.location.Location;  
import android.net.Uri;  
import android.os.Bundle;  
import android.util.Log;  
import android.widget.Toast;  
  
import com.google.android.gms.location.FusedLocationProviderClient;  
import com.google.android.gms.location.LocationServices;  
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;  
import com.google.android.gms.tasks.OnCompleteListener;  
import com.google.android.gms.tasks.Task;  
  
import java.io.IOException;  
import java.util.ArrayList;  
import java.util.List;  
  
public class MapActivity extends AppCompatActivity implements OnMapReadyCallback {  
 private static final String *TAG* = "MapAcitivity";  
 private static final String *FINE\_LOCATION* = Manifest.permission.*ACCESS\_FINE\_LOCATION*;  
 private static final String *COARSE\_LOCATION* = Manifest.permission.*ACCESS\_COARSE\_LOCATION*;  
 private static final int *LOCATION\_PERMISSION\_REQ\_CODE* = 1234;  
 private static final float *DEFAULT\_ZOOM* = 15f;  
 //variables  
 private String to\_be\_searched="Solapur";  
 private Boolean mLocationPermissionGranted = false;  
 private GoogleMap mMap;  
 private FusedLocationProviderClient mFusedLocationProviderClient;  
  
 @Override  
 public void onMapReady(GoogleMap googleMap) {  
 Toast.*makeText*(this, "Map is ready", Toast.*LENGTH\_SHORT*).show();  
 Log.*d*(*TAG*, "onMapReady:Map is Ready ");  
 mMap = googleMap;  
 if (mLocationPermissionGranted) {  
 getDeviceLocation();  
  
 if (ActivityCompat.*checkSelfPermission*(this, Manifest.permission.*ACCESS\_FINE\_LOCATION*) != PackageManager.*PERMISSION\_GRANTED* && ActivityCompat.*checkSelfPermission*(this, Manifest.permission.*ACCESS\_COARSE\_LOCATION*) != PackageManager.*PERMISSION\_GRANTED*) {  
 return;  
 }  
 mMap.setMyLocationEnabled(true);  
 }  
 }  
  
 public void map\_Call\_function(String url)  
 {  
 Intent intent=new Intent(Intent.*ACTION\_VIEW*);  
 intent.setData(Uri.*parse*(url));  
 startActivity(intent);  
  
 }  
  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_map*);  
 getLocationPermission();  
 init\_Petrol\_Pump\_info();  
 map\_Call\_function("https://www.google.com/maps/search/petrol+pump+near+me");  
 }  
  
  
 private void init\_Petrol\_Pump\_info() {  
 Log.*d*(*TAG*, "init\_Petrol\_Pump\_info: Initializing search for petrol pumps ");  
 geoLocate();  
 }  
  
 private void geoLocate(){  
 String search\_String=to\_be\_searched;  
 Log.*d*(*TAG*, "geoLocate: geolocating");  
 Geocoder geocoder=new Geocoder(MapActivity.this);  
 List<Address> list=new ArrayList<>();  
 try {  
 //currently returns 1 results for testing  
 list=geocoder.getFromLocationName(search\_String,1);  
  
 }catch (IOException e){  
 Log.*e*(*TAG*, "geoLocate: IO Exception : "+ e.getMessage() );  
 }  
 if (list.size()>0){  
 Address address=list.get(0);  
 Log.*d*(*TAG*, "geoLocate: found at location:\n" +address.toString());  
 //moveCamera(new LatLng(address.getLatitude(),address.getLongitude()),DEFAULT\_ZOOM,"none");  
 }  
  
 }  
  
  
 private void getDeviceLocation(){  
 Log.*d*(*TAG*,"getDeviceLocation: getting the current Devices current Location");  
 mFusedLocationProviderClient= LocationServices.*getFusedLocationProviderClient*(this);  
 try {  
 if(mLocationPermissionGranted){  
 Task location=mFusedLocationProviderClient.getLastLocation();  
 location.addOnCompleteListener(new OnCompleteListener() {  
 @Override  
 public void onComplete(@NonNull Task task) {  
 if(task.isSuccessful()) {  
 Log.*d*(*TAG*, "getDeviceLocation: found Location!");  
 Location currentLocation=(Location) task.getResult();  
 moveCamera(new LatLng(currentLocation.getLatitude(),currentLocation.getLongitude()),*DEFAULT\_ZOOM*,"My Location");  
  
 }else{  
 Log.*d*(*TAG*, "onComplete: Current Location is NULL");  
 Toast.*makeText*(MapActivity.this,"Unable to get Current Location",Toast.*LENGTH\_SHORT*).show();  
 }  
 }  
 });  
 }  
  
 }catch (SecurityException e){  
 Log.*e*(*TAG*,"getDeviceLocation: SecurityException: "+e.getMessage());  
 }  
 }  
  
 public void moveCamera(LatLng latLng,float zoom,String title){  
 Log.*d*(*TAG*, "moveCamera: moving camera to: lat: "+latLng.latitude+", lng: "+latLng.longitude+" , latlngobj: "+latLng);  
 mMap.moveCamera(CameraUpdateFactory.*newLatLngZoom*(latLng,zoom));  
 MarkerOptions options=new MarkerOptions().position(latLng).title(title);  
 mMap.addMarker(options);  
  
 }  
  
  
 private void initMap(){  
 Log.*d*(*TAG*, "initMap:initializing Map");  
 SupportMapFragment mapFragment=(SupportMapFragment) getSupportFragmentManager().findFragmentById(R.id.*map*);  
 mapFragment.getMapAsync(MapActivity.this);  
  
 }  
  
  
 private void getLocationPermission(){  
 Log.*d*(*TAG*, "getLocationPermission:Getting Permissions");  
 String[] permissions={Manifest.permission.*ACCESS\_FINE\_LOCATION*,Manifest.permission.*ACCESS\_COARSE\_LOCATION*};  
 if(ContextCompat.*checkSelfPermission*(this.getApplicationContext(),*FINE\_LOCATION*)== PackageManager.*PERMISSION\_GRANTED*){  
 if(ContextCompat.*checkSelfPermission*(this.getApplicationContext(),*COARSE\_LOCATION*)== PackageManager.*PERMISSION\_GRANTED*){  
 mLocationPermissionGranted=true;  
 initMap();  
 }  
 else{  
 ActivityCompat.*requestPermissions*(this,permissions,*LOCATION\_PERMISSION\_REQ\_CODE*);  
 }  
 }else{  
 ActivityCompat.*requestPermissions*(this,permissions,*LOCATION\_PERMISSION\_REQ\_CODE*);  
 }  
 }  
  
 @Override  
 public void onRequestPermissionsResult(int requestCode, @NonNull String[] permissions, @NonNull int[] grantResults) {  
 Log.*d*(*TAG*, "onRequestPermissionsResult:function called");  
 mLocationPermissionGranted=false;  
 switch(requestCode){  
 case *LOCATION\_PERMISSION\_REQ\_CODE*:{  
 if(grantResults.length>0){  
 for (int i=0;i<grantResults.length;i++){  
 if (grantResults[i]!=PackageManager.*PERMISSION\_GRANTED*) {  
 mLocationPermissionGranted = false;  
 Log.*d*(*TAG*, "onRequestPermissionsResult:Permission Failed");  
 return;  
 }  
 }  
 Log.*d*(*TAG*, "onRequestPermissionsResult:Permission Granted");  
 mLocationPermissionGranted=true;  
 //initialize map  
 initMap();  
 }  
 }  
 }  
 }  
  
}

Layout Files

1. Splash Screen Layout

<?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"  
 android:background="@drawable/smartdrive\_background"  
 tools:context=".SplashScreenActivity">  
  
  
 <TextView  
 android:id="@+id/Copyright\_text"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginBottom="80dp"  
 android:fontFamily="@font/roboto\_medium"  
 android:text="SmartDrive © 2021, All Right Reserved"  
 android:textColor="@color/white\_shade"  
 android:textSize="16sp"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintHorizontal\_bias="0.496"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toBottomOf="@+id/imageView" />  
  
 <ImageView  
 android:id="@+id/imageView"  
 android:layout\_width="685dp"  
 android:layout\_height="463dp"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent"  
 app:srcCompat="@drawable/smartdrive\_logo" />  
</androidx.constraintlayout.widget.ConstraintLayout>

1. SignUp Layout

<?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"  
 android:background="@drawable/signuppage\_smartdrive"  
 android:textAlignment="center"  
 app:flow\_horizontalAlign="center"  
 app:flow\_verticalAlign="center"  
 tools:context=".SignUpActivity">  
  
 <TextView  
 android:id="@+id/Welcome\_text"  
 android:layout\_width="276dp"  
 android:layout\_height="122dp"  
 android:fontFamily="@font/cabin\_semibold"  
 android:text="Welcome SmartDrivers ,"  
 android:textAlignment="viewStart"  
 android:textColor="@color/white\_shade"  
 android:textSize="40sp"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintHorizontal\_bias="0.162"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent"  
 app:layout\_constraintVertical\_bias="0.026" />  
  
 <TextView  
 android:id="@+id/Message\_text"  
 android:layout\_width="317dp"  
 android:layout\_height="55dp"  
 android:fontFamily="@font/roboto\_bold"  
 android:text="Sign Up !!"  
 android:textAlignment="center"  
 android:textColor="@color/grey\_shade"  
 android:textSize="28sp"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toBottomOf="@+id/Welcome\_text"  
 app:layout\_constraintVertical\_bias="0.11" />  
  
 <EditText  
 android:id="@+id/editText\_Name"  
 android:layout\_width="300dp"  
 android:layout\_height="50dp"  
 android:background="@drawable/edittext\_layout"  
 android:ems="10"  
 android:fontFamily="@font/roboto\_medium"  
 android:hint=" Enter Your Name"  
 android:inputType="textPersonName"  
 android:paddingLeft="5dp"  
 android:textAlignment="viewStart"  
 android:textColor="@color/black\_shade"  
 android:textSize="20sp"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintHorizontal\_bias="0.405"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toBottomOf="@+id/Message\_text"  
 app:layout\_constraintVertical\_bias="0.091" />  
  
 <EditText  
 android:id="@+id/editText\_Phone"  
 android:layout\_width="185dp"  
 android:layout\_height="50dp"  
 android:background="@drawable/edittext\_layout"  
 android:ems="10"  
 android:fontFamily="@font/roboto\_medium"  
 android:hint="Enter Mobile No."  
 android:inputType="phone"  
 android:textAlignment="center"  
 android:textColor="@color/black\_shade"  
 android:textSize="20sp"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintHorizontal\_bias="0.0"  
 app:layout\_constraintStart\_toEndOf="@+id/ccp\_input"  
 app:layout\_constraintTop\_toBottomOf="@+id/editText\_Name"  
 app:layout\_constraintVertical\_bias="0.172" />  
  
 <com.hbb20.CountryCodePicker  
 android:id="@+id/ccp\_input"  
 android:layout\_width="114dp"  
 android:layout\_height="55dp"  
 android:textAlignment="center"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintHorizontal\_bias="0.144"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toBottomOf="@+id/editText\_Name"  
 app:layout\_constraintVertical\_bias="0.172" />  
  
 <Button  
 android:id="@+id/signUp\_Button"  
 android:layout\_width="250dp"  
 android:layout\_height="55dp"  
 android:background="@drawable/button\_layout"  
 android:fontFamily="@font/roboto\_medium"  
 android:text="Sign Up"  
 android:textSize="16sp"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toBottomOf="@+id/editText\_Phone"  
 app:layout\_constraintVertical\_bias="0.2" />  
</androidx.constraintlayout.widget.ConstraintLayout>

1. Otp Layout

<?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"  
 android:textAlignment="center"  
 android:background="@drawable/signuppage\_smartdrive"  
 tools:context=".GetOTPActivity">  
  
 <TextView  
 android:id="@+id/Heading1"  
 android:layout\_width="127dp"  
 android:layout\_height="44dp"  
 android:fontFamily="@font/roboto\_medium"  
 android:letterSpacing=".3"  
 android:text="GET"  
 android:textAlignment="center"  
 android:textColor="@color/black\_shade"  
 android:textSize="30sp"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent"  
 app:layout\_constraintVertical\_bias="0.264" />  
  
 <TextView  
 android:id="@+id/Heading2"  
 android:layout\_width="115dp"  
 android:layout\_height="144dp"  
 android:fontFamily="@font/roboto\_bold"  
 android:letterSpacing=".2"  
 android:text="CODE"  
 android:textAlignment="center"  
 android:textColor="@color/grey\_shade"  
 android:textSize="55sp"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toBottomOf="@+id/Heading1"  
 app:layout\_constraintVertical\_bias="0.0" />  
  
 <EditText  
 android:id="@+id/Otp\_EditText"  
 android:layout\_width="295dp"  
 android:layout\_height="50dp"  
 android:background="@drawable/edittext\_layout"  
 android:ems="10"  
 android:fontFamily="@font/roboto\_medium"  
 android:hint="Enter OTP !!"  
 android:inputType="numberPassword"  
 android:textAlignment="center"  
 android:textColor="@color/black\_shade"  
 android:textSize="20sp"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toBottomOf="@+id/Heading2"  
 app:layout\_constraintVertical\_bias="0.158" />  
  
 <Button  
 android:id="@+id/button\_VerifyOTP"  
 android:layout\_width="250dp"  
 android:layout\_height="55dp"  
 android:background="@drawable/button\_layout"  
 android:fontFamily="@font/roboto\_medium"  
 android:text="Verify OTP"  
 android:textSize="16sp"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toBottomOf="@+id/Otp\_EditText"  
 app:layout\_constraintVertical\_bias="0.14" />  
  
 <TextView  
 android:id="@+id/Main\_heading"  
 android:layout\_width="300dp"  
 android:layout\_height="64dp"  
 android:fontFamily="@font/cabin\_semibold"  
 android:text="Just One More Step ..."  
 android:textAlignment="viewStart"  
 android:textColor="@color/white\_shade"  
 android:textSize="30sp"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintHorizontal\_bias="0.477"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent"  
 app:layout\_constraintVertical\_bias="0.128" />  
  
</androidx.constraintlayout.widget.ConstraintLayout>

1. ArdiunoIP Layout

<?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"  
 android:background="@drawable/signuppage\_smartdrive"  
 tools:context=".ArdiunoIPActivity">  
  
 <Button  
 android:id="@+id/btn"  
 android:layout\_width="250dp"  
 android:layout\_height="55dp"  
 android:layout\_marginBottom="172dp"  
 android:textSize="14sp"  
 android:background="@drawable/button\_layout"  
 android:text="Calculate Fuel and Map"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintHorizontal\_bias="0.496"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toBottomOf="@+id/Litres\_conversion"  
 app:layout\_constraintVertical\_bias="0.886" />  
  
 <TextView  
 android:id="@+id/waitingText"  
 android:layout\_width="240dp"  
 android:layout\_height="100dp"  
 android:fontFamily="@font/cabin\_semibold"  
 android:text="Getting input Please Wait ..."  
 android:background="@drawable/edittext\_layout"  
 android:textColor="@color/white\_shade"  
 android:textSize="35sp"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintHorizontal\_bias="0.257"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent"  
 app:layout\_constraintVertical\_bias="0.085" />  
  
 <EditText  
 android:id="@+id/editText\_ardiunoIP"  
 android:layout\_width="270dp"  
 android:layout\_height="50dp"  
 android:background="@drawable/edittext\_layout"  
 android:ems="10"  
 android:hint="Input from Ardiuno"  
 android:inputType="textPersonName"  
 android:textAlignment="center"  
 android:textSize="20sp"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintHorizontal\_bias="0.496"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toBottomOf="@+id/waitingText"  
 app:layout\_constraintVertical\_bias="0.153" />  
  
 <TextView  
 android:id="@+id/Litres\_conversion"  
 android:layout\_width="270dp"  
 android:layout\_height="50dp"  
 android:background="@drawable/edittext\_layout"  
 android:text="Coversions in Litres"  
 android:textAlignment="center"  
 android:textSize="20sp"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintHorizontal\_bias="0.496"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toBottomOf="@+id/editText\_ardiunoIP"  
 app:layout\_constraintVertical\_bias="0.138" />  
  
</androidx.constraintlayout.widget.ConstraintLayout>

1. Map Layout

<?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=".MapActivity">  
  
 <fragment  
 android:id="@+id/map"  
 android:name="com.google.android.gms.maps.SupportMapFragment"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 tools:layout\_editor\_absoluteX="285dp"  
 tools:layout\_editor\_absoluteY="286dp" />  
  
</androidx.constraintlayout.widget.ConstraintLayout>

1. Button Layout

<?xml version="1.0" encoding="utf-8"?>  
<shape xmlns:android="http://schemas.android.com/apk/res/android"  
 android:shape="rectangle">  
 <stroke  
 android:width="2dp"  
 android:color="@color/red"/>  
 <corners  
 android:radius="50dp"/>  
  
</shape>

1. EditText Layout

<?xml version="1.0" encoding="utf-8"?>  
<shape xmlns:android="http://schemas.android.com/apk/res/android"  
 android:shape="rectangle">  
 <stroke  
 android:color="@color/red"  
 android:width="1dp"/>  
 <corners  
 android:radius="50dp"/>  
</shape>

Android Manifest

<?xml version="1.0" encoding="utf-8"?>  
<manifest xmlns:android="http://schemas.android.com/apk/res/android"  
 package="com.pkg.smartdrive">  
  
 <uses-permission android:name="android.permission.INTERNET"/>  
 <uses-permission android:name="android.permission.ACCESS\_FINE\_LOCATION"/>  
 <uses-permission android:name="android.permission.ACCESS\_COARSE\_LOCATION"/>  
 <uses-permission android:name="android.permission.ACCESS\_NETWORK\_STATE"/>  
  
 <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.SmartDrive">  
 <meta-data  
 android:name="com.google.android.gms.version"  
 android:value="@integer/google\_play\_services\_version" />  
  
 <meta-data  
 android:name="com.google.android.geo.API\_KEY"  
 android:value="@string/google\_maps\_api\_key" />  
  
 <activity android:name=".MapActivity"></activity>  
 <activity android:name=".ArdiunoIPActivity" />  
 <activity android:name=".GetOTPActivity" />  
 <activity android:name=".SignUpActivity" />  
 <activity android:name=".SplashScreenActivity">  
 <intent-filter>  
 <action android:name="android.intent.action.MAIN" />  
  
 <category android:name="android.intent.category.LAUNCHER" />  
 </intent-filter>  
 </activity>  
  
 <meta-data  
 android:name="preloaded\_fonts"  
 android:resource="@array/preloaded\_fonts" />  
 </application>  
  
</manifest>

Gradle Scripts

// Top-level build file where you can add configuration options common to all sub-projects/modules.  
buildscript {  
 repositories {  
 google()  
 jcenter()  
 }  
 dependencies {  
 classpath "com.android.tools.build:gradle:4.1.2"  
 classpath 'com.google.gms:google-services:4.2.0'  
  
 // NOTE: Do not place your application dependencies here; they belong  
 // in the individual module build.gradle files  
 }  
}  
  
allprojects {  
 repositories {  
 google()  
 jcenter()  
 }  
}  
  
task clean(type: Delete) {  
 delete rootProject.buildDir  
}

plugins **{** id 'com.android.application'  
 id 'com.google.gms.google-services'  
**}**android {  
 compileSdkVersion 30  
 buildToolsVersion "30.0.3"  
  
 defaultConfig {  
 applicationId "com.pkg.smartdrive"  
 minSdkVersion 21  
 targetSdkVersion 30  
 versionCode 1  
 versionName "1.0"  
  
 testInstrumentationRunner "androidx.test.runner.AndroidJUnitRunner"  
 }  
  
 buildTypes {  
 release {  
 minifyEnabled false  
 proguardFiles getDefaultProguardFile('proguard-android-optimize.txt'), 'proguard-rules.pro'  
 }  
 }  
 compileOptions {  
 sourceCompatibility JavaVersion.VERSION\_1\_8  
 targetCompatibility JavaVersion.VERSION\_1\_8  
 }  
}  
  
dependencies {  
  
 implementation 'androidx.appcompat:appcompat:1.2.0'  
 implementation 'com.google.android.material:material:1.3.0'  
 implementation 'androidx.constraintlayout:constraintlayout:2.0.4'  
 //Country Code Picker from Github  
 implementation 'com.hbb20:ccp:2.4.5'  
 testImplementation 'junit:junit:4.+'  
 androidTestImplementation 'androidx.test.ext:junit:1.1.2'  
 androidTestImplementation 'androidx.test.espresso:espresso-core:3.3.0'  
 //Firebase  
 implementation 'com.google.firebase:firebase-auth:16.0.5'  
 //google service API  
 implementation 'com.google.android.gms:play-services-location:18.0.0'  
 //Google maps  
 implementation 'com.google.android.gms:play-services-maps:17.0.0'  
  
  
}

Colors Layout

<?xml version="1.0" encoding="utf-8"?>  
<resources>  
 <color name="red">#E60000</color>  
 <color name="white\_shade">#F4F6F7</color>  
 <color name="grey\_shade">#3B3B3B</color>  
 <color name="black\_shade">#1f1f1f</color>  
 <color name="black">#000000</color>  
 <color name="white">#FFFFFFFF</color>  
</resources>

API Key info

<resources>  
 <string name="app\_name">SmartDrive</string>  
 <string name="google\_maps\_api\_key">AIzaSyBXg3wcezL5pPyZUfjz7m4x4RpYxNQB7jU</string>  
</resources>