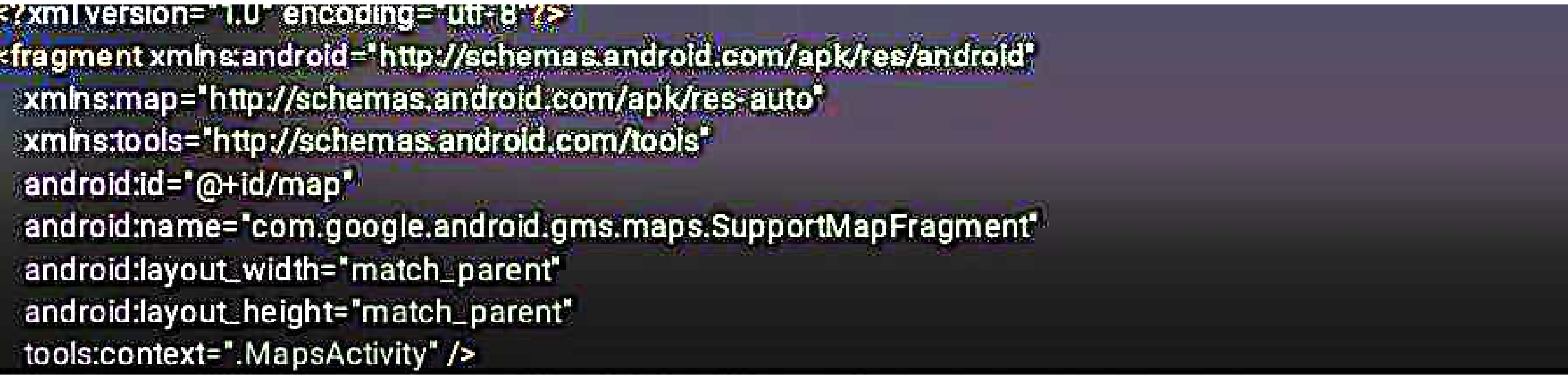
Ganpat University

॥ विद्यया समाजोत्कर्षः ॥

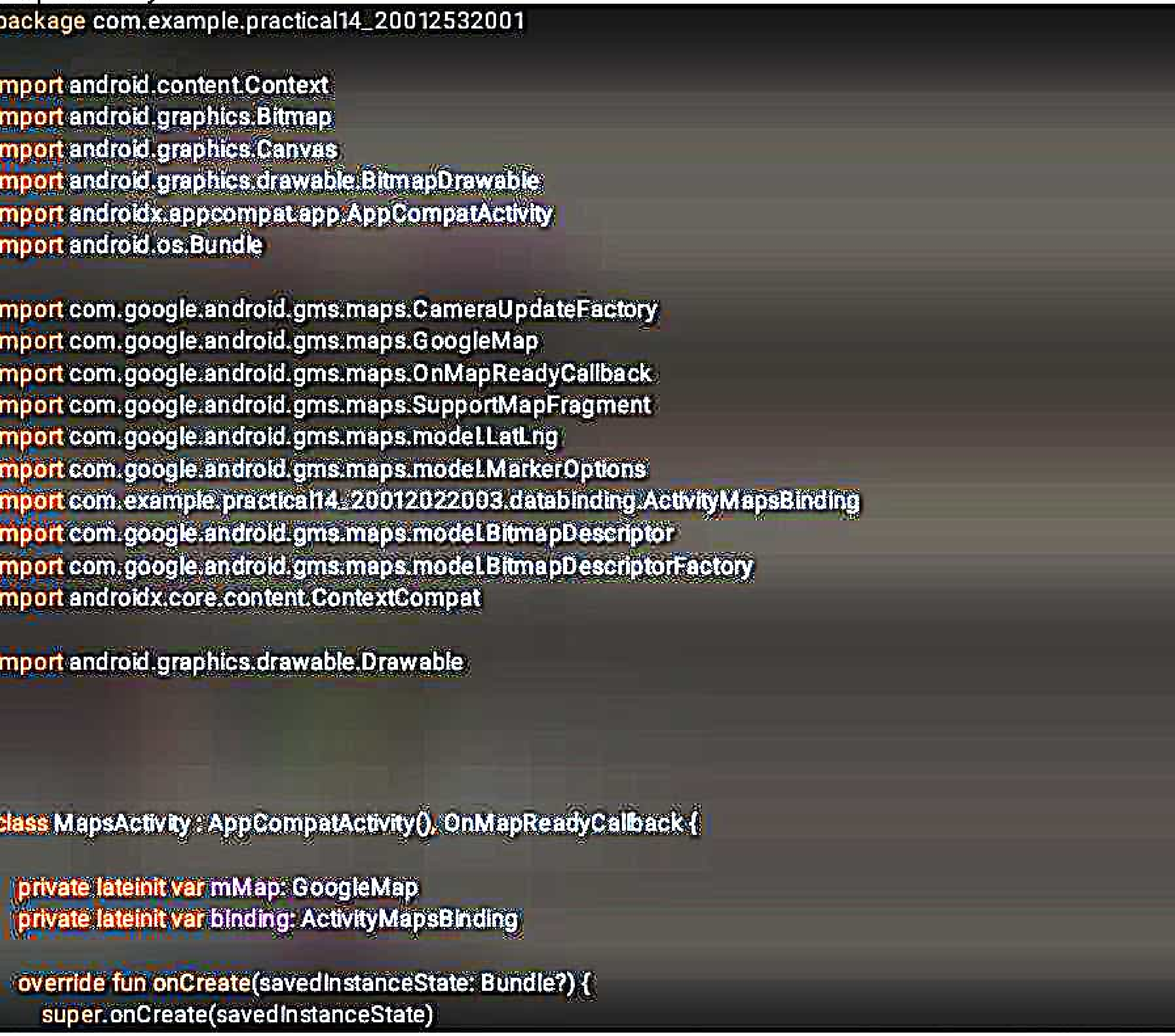
U.V. Patel College of Engineering

Department of Computer Engineering/Information Technology

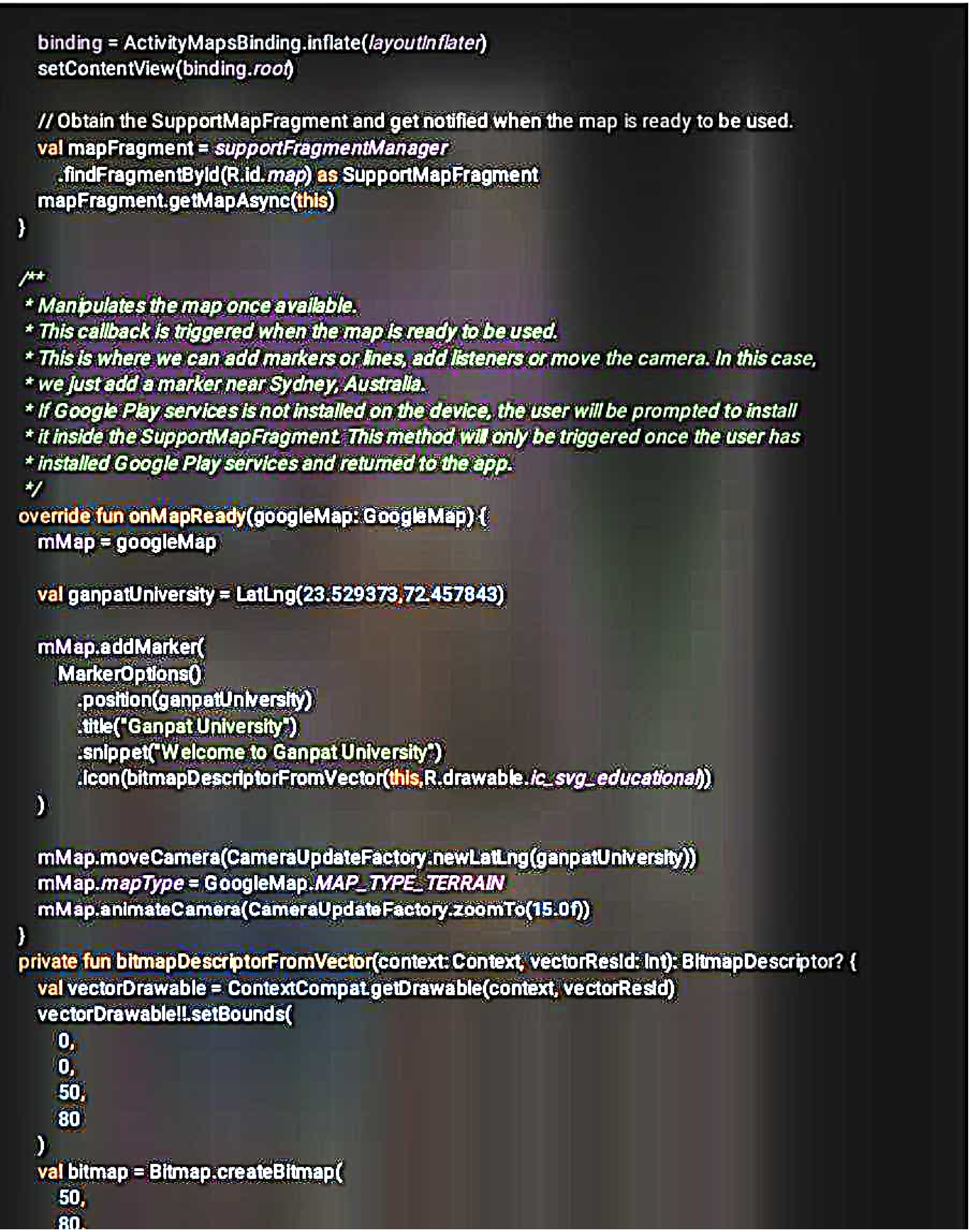


```
<?xml version="1.0" encoding="utf-8"?>
<fragment xmlns:android="http://schemas.android.com/apk/res/android"
  xmlns:map="http://schemas.android.com/apk/res-auto"
  xmlns:tools="http://schemas.android.com/tools"
  android:id="@+id/map"
  android:name="com.google.android.gms.maps.SupportMapFragment"
  android:layout_width="match_parent"
  android:layout_height="match_parent"
  tools:context=".MapsActivity" />
```

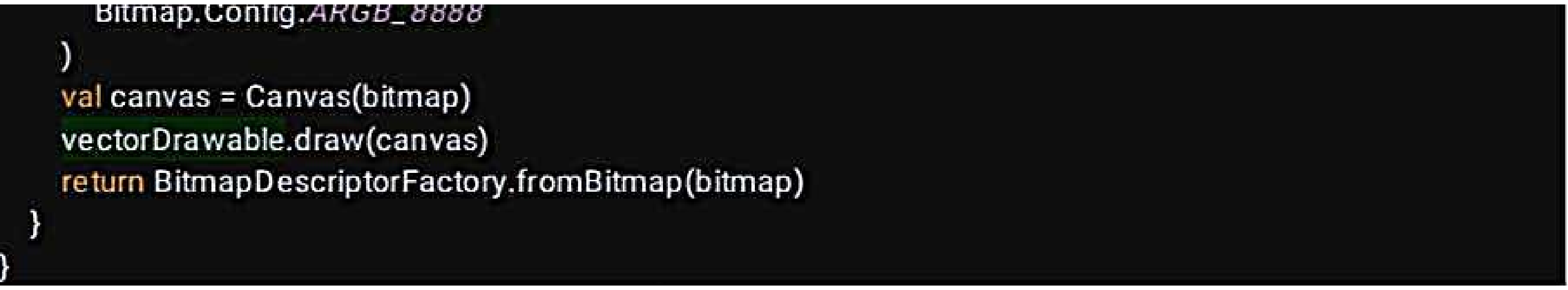
MapsActivity.kt:



```
package com.example.practical14_20012532001

mport android.content.Context
mport android.graphics.Bitmap
mport android.graphics.Canvas
mport android.graphics.drawable.BitmapDrawable
mport androidx.appcompat.app.AppCompatActivity
mport android.os.Bundle

mport com.google.android.gms.maps.CameraUpdateFactory
mport com.google.android.gms.maps.GoogleMap
mport com.google.android.gms.maps.OnMapReadyCallback
mport com.google.android.gms.maps.SupportMapFragment
mport com.google.android.gms.maps.model.LatLng
mport com.google.android.gms.maps.model.MarkerOptions
mport com.example.practical14_20012022003.databinding.ActivityMapsBinding
mport com.google.android.gms.maps.model.BitmapDescriptor
mport com.google.android.gms.maps.model.BitmapDescriptorFactory
mport androidx.core.content.ContextCompat

mport android.graphics.drawable.Drawable




class MapsActivity : AppCompatActivity(), OnMapReadyCallback {

  private lateinit var mMap: GoogleMap
  private lateinit var binding: ActivityMapsBinding

  override fun onCreate(savedInstanceState: Bundle?) {
    super.onCreate(savedInstanceState)
```

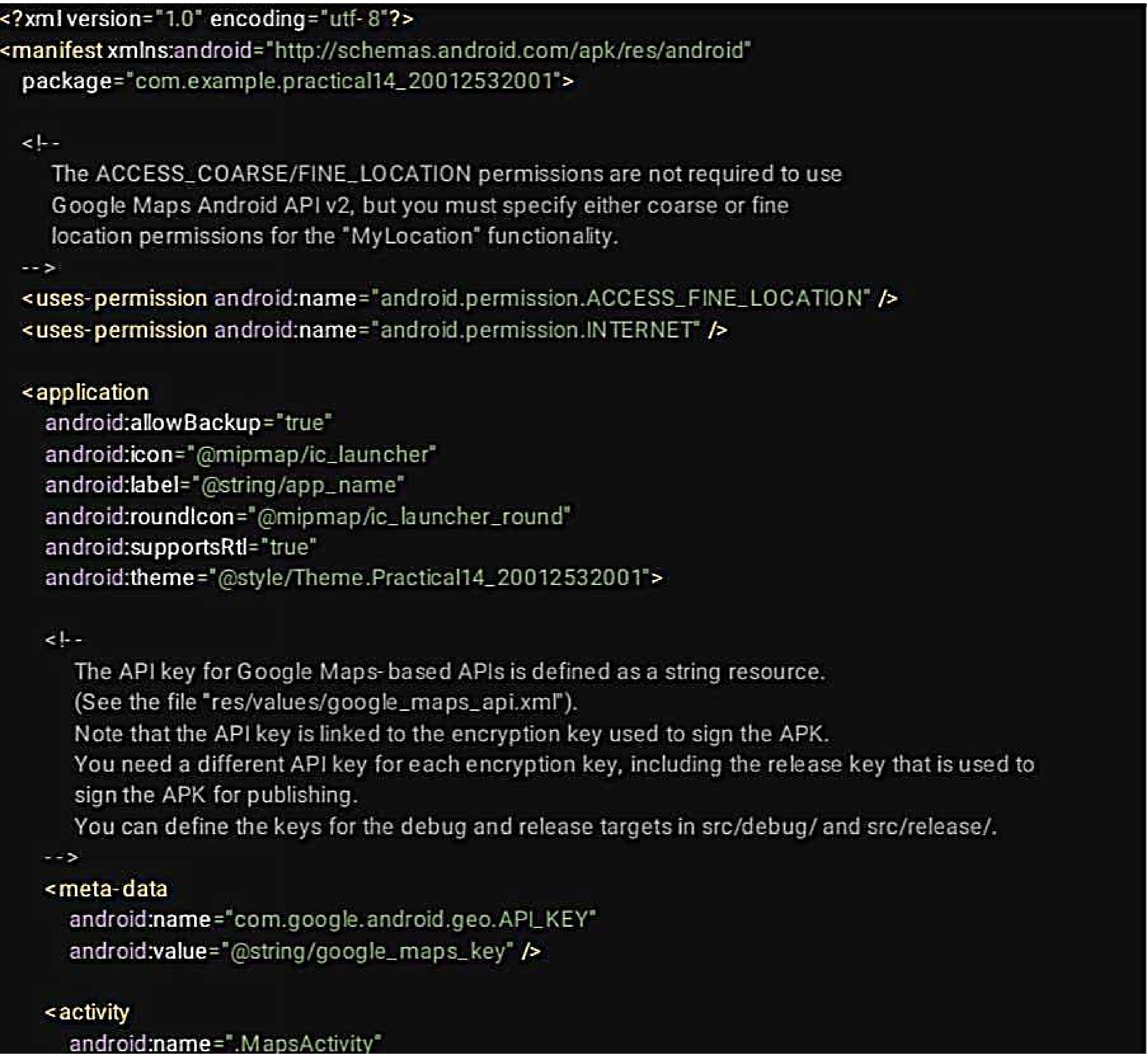


```
    binding = ActivityMapsBinding.inflate(layoutInflater)
    setContentView(binding.root)

    // Obtain the SupportMapFragment and get notified when the map is ready to be used.
    val mapFragment = supportFragmentManager
        .findFragmentById(R.id.map) as SupportMapFragment
    mapFragment.getMapAsync(this)
}

/**
 * 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 fun onMapReady(googleMap: GoogleMap) {
    mMap = googleMap

    val ganpatUniversity = LatLng(23.529373,72.457843)

    mMap.addMarker(
        MarkerOptions()
            .position(ganpatUniversity)
            .title("Ganpat University")
            .snippet("Welcome to Ganpat University")
            .icon(bitmapDescriptorFromVector(this,R.drawable.ic_svg_educational))
    )

    mMap.moveCamera(CameraUpdateFactory.newLatLng(ganpatUniversity))
    mMap.mapType = GoogleMap.MAP_TYPE_TERRAIN
    mMap.animateCamera(CameraUpdateFactory.zoomTo(15.0f))
}
private fun bitmapDescriptorFromVector(context: Context, vectorResId: Int): BitmapDescriptor? {
    val vectorDrawable = ContextCompat.getDrawable(context, vectorResId)
    vectorDrawable!!.setBounds(
        0,
        0,
        50,
        80
    )
    val bitmap = Bitmap.createBitmap(
        50,
        80,
```

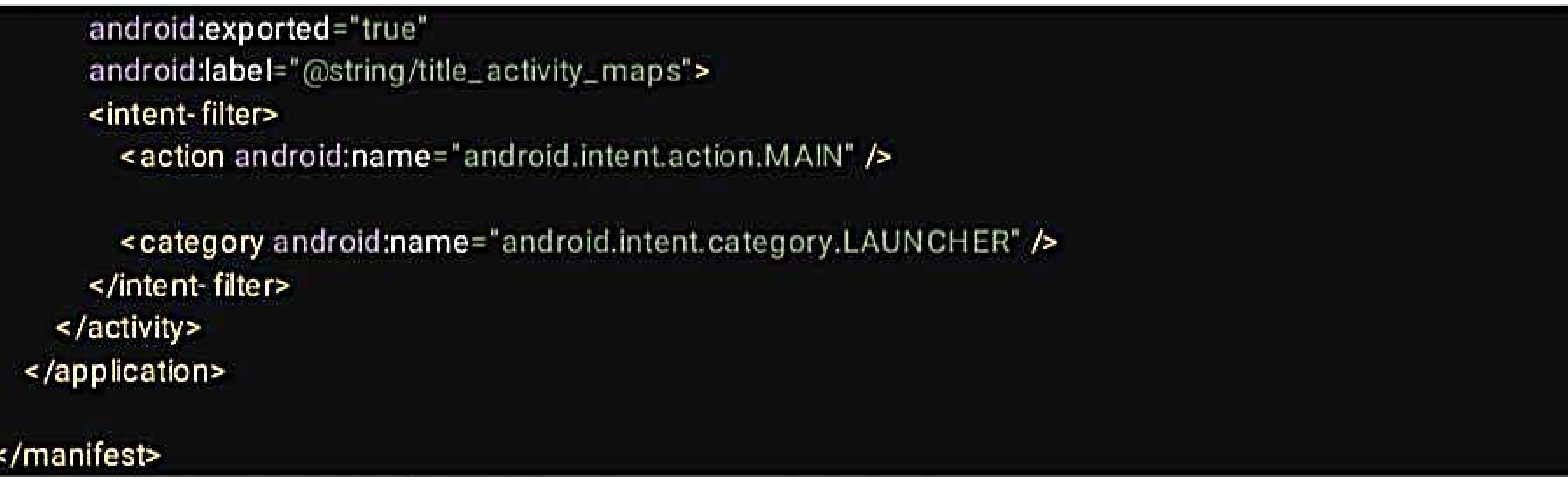


```
      Bitmap.Config.ARGB_8888
    )
    val canvas = Canvas(bitmap)
    vectorDrawable.draw(canvas)
    return BitmapDescriptorFactory.fromBitmap(bitmap)
  }
}
```

AndroidManifest.xml:

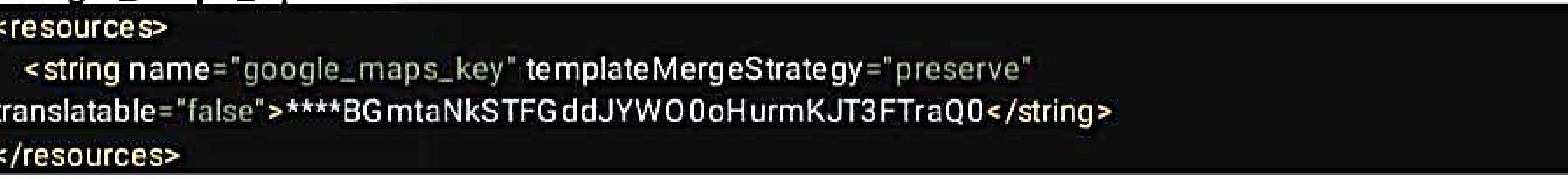


```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
  package="com.example.practical14_20012532001">

  <!--
    The ACCESS_COARSE/FINE_LOCATION permissions are not required to use
    Google Maps Android API v2, but you must specify either coarse or fine
    location permissions for the "MyLocation" functionality.
  -->
  <uses-permission android:name="android.permission.ACCESS_FINE_LOCATION" />
  <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/Theme.Practical14_20012532001">

    <!--
      The API key for Google Maps-based APIs is defined as a string resource.
      (See the file "res/values/google_maps_api.xml").
      Note that the API key is linked to the encryption key used to sign the APK.
      You need a different API key for each encryption key, including the release key that is used to
      sign the APK for publishing.
      You can define the keys for the debug and release targets in src/debug/ and src/release/.
    -->
    <meta-data
      android:name="com.google.android.geo.API_KEY"
      android:value="@string/google_maps_key" />

    <activity
      android:name=".MapsActivity"
```

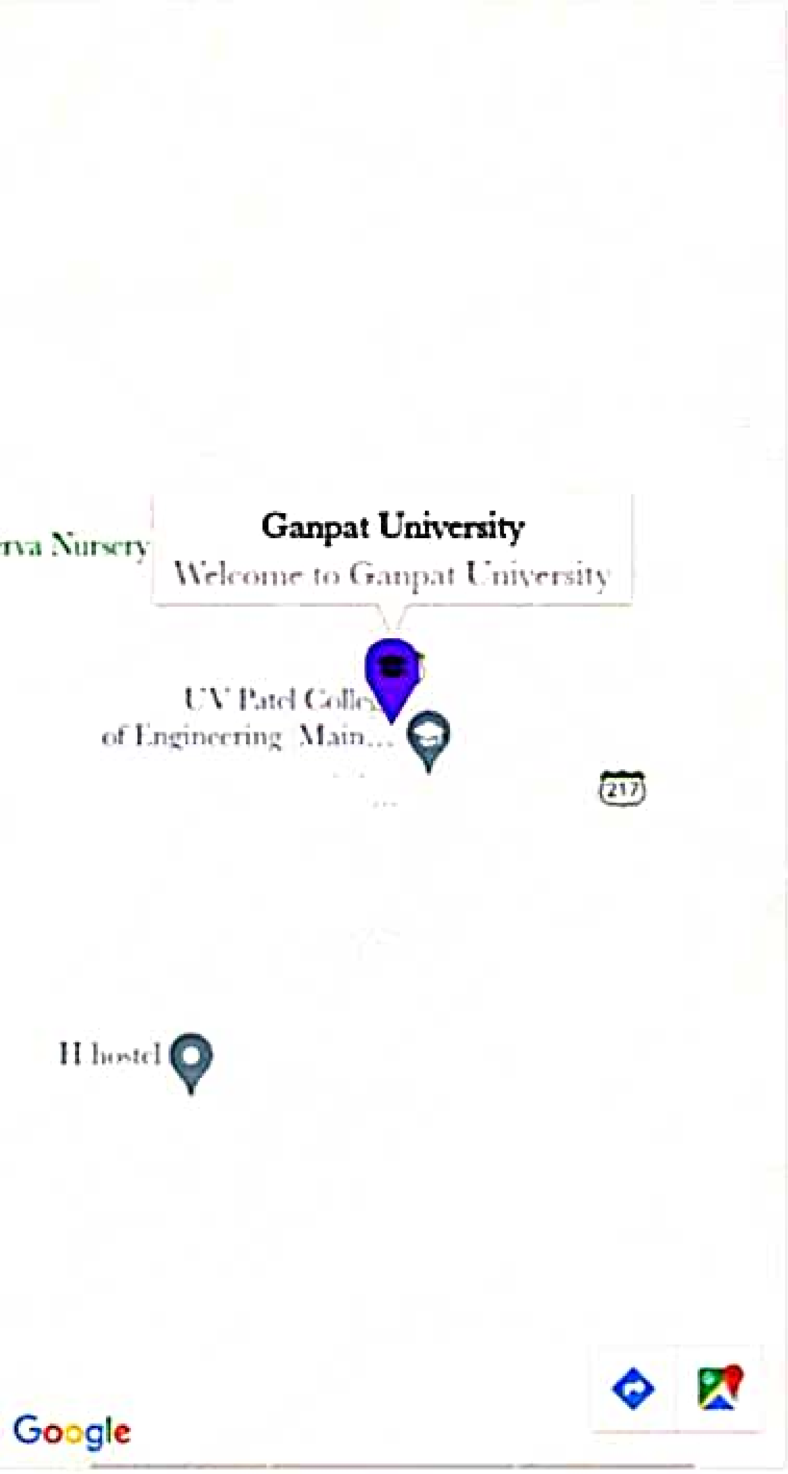


```
        android:exported="true"
        android:label="@string/title_activity_maps">
        <intent-filter>
          <action android:name="android.intent.action.MAIN" />

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

</manifest>
```

Google_maps_api.xml:



```
<resources>
  <string name="google_maps_key" templateMergeStrategy="preserve"
translatable="false">****BGmtaNkSTFGddJYWO0oHurmKJT3FTraQ0</string>
</resources>
```


Ganpat University
Welcome to Ganpat University
rva Nursery
UV Patel Colleg
of Engineering (Main...
217
H hostel
Google