MOBILE APPLICATION & DEVELOPMENT ASSIGNMENT

EX-07 TELEPHONY SERVICES

AIM:

To implement an application to get Telephony services.

PROCEDURE:

Step 1: Create a New Project

- Open Android Studio → Click New Project → Enter project name (e.g., *TelephonyApp*).
- Select minimum SDK (for example, API 21 or higher) → Click Next → Choose Empty Activity → Click Finish.

Step 2: Add Required Permissions

- Open the AndroidManifest.xml file.
- Add permissions to access phone state and network information, like READ_PHONE_STATE, READ_PHONE_NUMBERS, and ACCESS_NETWORK_STATE.

Step 3: Design the User Interface

- Open activity_main.xml.
- Add a Button (to fetch details) and a TextView (to display the telephony information).

Step 4: Initialize Telephony Services

- In MainActivity.kt, initialize the TelephonyManager service.
- Set up the button so that when clicked, it fetches device information like:
    - Mobile number,
    - Network operator name,
    - SIM operator,
    - IMEI number.

Step 5: Handle Runtime Permissions

MOBILE APPLICATION & DEVELOPMENT ASSIGNMENT

EX-07 TELEPHONY SERVICES

- For devices running Android 6.0 (Marshmallow) and above, ask for permissions during runtime.
- Ensure permission is granted before trying to access telephony data.

Step 6: Display the Information

- Once the button is clicked and permission is granted, display the collected telephony information inside the TextView.

Step 7: Run the Application

- You can run the app either on a real device (recommended) or emulator (limited info in emulator).

CODE:

AndroidManifest.xml :

```
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
package="com.example.telephonyapp">


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


   <application android:allowBackup="true" android:label="Telephony &
Location Info" android:theme="@style/Theme.AppCompat.DayNight">
```

MOBILE APPLICATION & DEVELOPMENT ASSIGNMENT

EX-07 TELEPHONY SERVICES

```
    <activity android:name=".MainActivity" android:exported="true">
      <intent-filter>
        <action android:name="android.intent.action.MAIN"/>
        <category android:name="android.intent.category.LAUNCHER"/>
      </intent-filter>
    </activity>
  </application>
</manifest>
```

Activity_main.xml:

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

  <LinearLayout
    android:orientation="vertical"
    android:padding="20dp"
    android:layout_width="match_parent"
    android:layout_height="wrap_content">

    <TextView
      android:id="@+id/tvTelephonyInfo"
      android:layout_width="match_parent"
      android:layout_height="wrap_content"
      android:text="Telephony Information"
      android:textStyle="bold"
```

```
    android:textSize="18sp"
    android:layout_marginBottom="10dp"/>

<TextView
    android:id="@+id/tvLocation"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="Location Details"
    android:textStyle="bold"
    android:textSize="18sp"
    android:layout_marginTop="10dp"/>

<TextView
    android:id="@+id/tvAddress"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="Address Information"
    android:textStyle="bold"
    android:textSize="18sp"
    android:layout_marginTop="10dp"/>

<Button
    android:id="@+id/btnFetch"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="Fetch Info"
```

```
        android:layout_marginTop="20dp"/>

    </LinearLayout>
</ScrollView>
```

MainActivity.kt :

```
package com.example.telephonyapp

import android.Manifest
import android.content.pm.PackageManager
import android.location.Geocoder
import android.location.Location
import android.location.LocationManager
import android.os.Bundle
import android.telephony.TelephonyManager
import android.widget.Button
import android.widget.TextView
import androidx.appcompat.app.AppCompatActivity
import androidx.core.app.ActivityCompat
import java.util.*

class MainActivity : AppCompatActivity() {

    private lateinit var tvTelephonyInfo: TextView
    private lateinit var tvLocation: TextView
    private lateinit var tvAddress: TextView
```

MOBILE APPLICATION & DEVELOPMENT ASSIGNMENT

EX-07 TELEPHONY SERVICES

```
    private lateinit var btnFetch: Button
    private val LOCATION_PERMISSION = 101
    private lateinit var locationManager: LocationManager

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_main)

        tvTelephonyInfo = findViewById(R.id.tvTelephonyInfo)
        tvLocation = findViewById(R.id.tvLocation)
        tvAddress = findViewById(R.id.tvAddress)
        btnFetch = findViewById(R.id.btnFetch)

        btnFetch.setOnClickListener {
            if (checkPermissions()) {
                displayTelephonyInfo()
                fetchLocation()
            } else {
                requestPermissions()
            }
        }
    }

    private fun checkPermissions(): Boolean {
        return ActivityCompat.checkSelfPermission(this,
Manifest.permission.ACCESS_FINE_LOCATION) ==
PackageManager.PERMISSION_GRANTED &&
```

MOBILE APPLICATION & DEVELOPMENT ASSIGNMENT

EX-07 TELEPHONY SERVICES

```
            ActivityCompat.checkSelfPermission(this,
Manifest.permission.READ_PHONE_STATE) ==
PackageManager.PERMISSION_GRANTED
    }

    private fun requestPermissions() {
        ActivityCompat.requestPermissions(this,
            arrayOf(Manifest.permission.ACCESS_FINE_LOCATION,
Manifest.permission.READ_PHONE_STATE),
            LOCATION_PERMISSION)
    }

    override fun onRequestPermissionsResult(requestCode: Int, permissions:
Array<out String>, grantResults: IntArray) {
        if (requestCode == LOCATION_PERMISSION &&
grantResults.isNotEmpty() && grantResults[0] ==
PackageManager.PERMISSION_GRANTED) {
            displayTelephonyInfo()
            fetchLocation()
        } else {
            tvTelephonyInfo.text = "Permission Denied"
            tvLocation.text = "Permission Denied"
        }
    }

    private fun displayTelephonyInfo() {
        val telephonyManager = getSystemService(TELEPHONY_SERVICE) as
TelephonyManager
```

MOBILE APPLICATION & DEVELOPMENT ASSIGNMENT

EX-07 TELEPHONY SERVICES

```
        val info = """
            Network Operator: ${telephonyManager.networkOperatorName}
            SIM Country: ${telephonyManager.simCountryIso}
            SIM Operator: ${telephonyManager.simOperatorName}
            Phone Type: ${when (telephonyManager.phoneType) {
                TelephonyManager.PHONE_TYPE_GSM -> "GSM"
                TelephonyManager.PHONE_TYPE_CDMA -> "CDMA"
                else -> "Unknown"
            }}
        """.trimIndent()

        tvTelephonyInfo.text = info
    }

    private fun fetchLocation() {
        locationManager = getSystemService(LOCATION_SERVICE) as
LocationManager

        if (ActivityCompat.checkSelfPermission(this,
Manifest.permission.ACCESS_FINE_LOCATION) !=
PackageManager.PERMISSION_GRANTED) return

        val location: Location? =
locationManager.getLastKnownLocation(LocationManager.GPS_PROVIDER)
            ?:
locationManager.getLastKnownLocation(LocationManager.NETWORK_PROV
IDER)
```

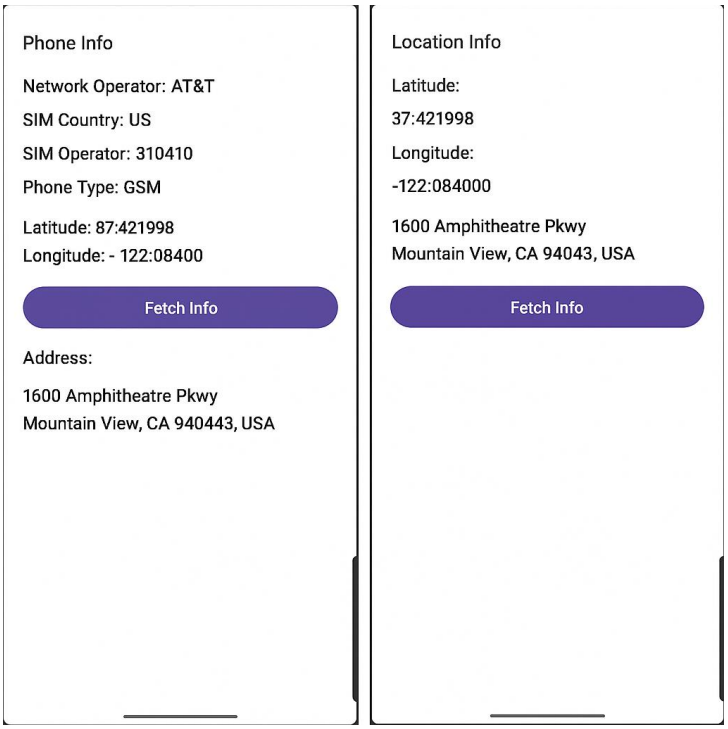
```
    if (location != null) {
        val lat = location.latitude
        val lon = location.longitude
        tvLocation.text = "Latitude: $lat\nLongitude: $lon"
        getAddress(lat, lon)
    } else {
        tvLocation.text = "Unable to get location."
    }
}

private fun getAddress(lat: Double, lon: Double) {
    val geocoder = Geocoder(this, Locale.getDefault())
    try {
        val addressList = geocoder.getFromLocation(lat, lon, 1)
        if (!addressList.isNullOrEmpty()) {
            val address = addressList[0]
            val fullAddress = address.getAddressLine(0)


            tvAddress.text = "Address:\n$fullAddress"
        } else {
            tvAddress.text = "Unable to get address."
        }
    } catch (e: Exception) {
        e.printStackTrace()
        tvAddress.text = "Geocoder error: ${e.localizedMessage}"
```

```
        }
    }
}
```

OUTPUT IMAGE:



RESULT:

The application has been successfully developed using Kotlin and android studio.