**EXPERIMENT-7 TELEPHONY SERVICES**

**AIM:**

To implement an application to get Telephony services.

**PROCEDURE:**

* Go to File → New Project. f
* Provide the Application Name and click Next.
* Select the Target Android Devices.
* Choose the Minimum SDK required.
* Choose the activity (Blank Activity by default).
* Enter the Activity Name.
* Click Finish.
* Edit the program (design layout and write Kotlin code).
* Run the Application in 2 ways:
  1. Using an Emulator.
  2. Using a Real Mobile Device (via USB debugging).

**PROGRAM:**

## **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">

<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

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 &&

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

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\_PROVIDER)

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}"

}

}

}