Ex. No. : 08 - A **Date :**

Register No. : 221701051 **Name : SELVAHARIBALAN S**

Send SMS

Aim

Develop an application to send SMS.

Procedure:

Step 1 : File → New Project

Provide the application name (e.g.,"SMS") and click “Next”.

Step 2 : Select the target Android devices

Select the minimum SDK to run the application. Click “Next”.

Step 3 : Choose the activity for the application

By default, choose “Blank Activity”. Click “Next”.

Step 4 : Enter activity name and click “Finish”.

Step 5 : Edit the program

Add SMS permissions (SEND_SMS) in the manifest file.

Use SMSMANAGER in MainActivity.kt to send messages from the app to a provided phone number

.

Step 6 : Run the application

Two ways to run the application:

1. Running through emulator (with Sim Card Configured)
2. Running through mobile device

AndroidManifest.xml

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

    <!-- Required permission for sending SMS -->
    <uses-permission android:name="android.permission.SEND_SMS" />

    <!-- Declare that the app uses telephony features but doesn't require them
-->
    <uses-feature
        android:name="android.hardware.telephony"
        android:required="false" />

    <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.SMS">
        <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"?>
<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="@color/white"
    tools:context=".MainActivity">

    <TextView
        android:id="@+id/titleTextView"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:background="@color/purple_500"
        android:padding="16dp"
        android:text="Send SMS"
        android:textColor="@color/white"
        android:textSize="20sp"
        android:textStyle="bold"
        app:layout_constraintTop_toTopOf="parent" />

    <EditText
        android:id="@+id/phoneNumberEditText"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginStart="16dp"
        android:layout_marginTop="16dp"
        android:layout_marginEnd="16dp"
```

```
        android:background="@drawable/edit_text_background"
        android:hint="Enter the phone number...!"
        android:inputType="phone"
        android:padding="12dp"
        android:textSize="16sp"
        app:layout_constraintTop_toBottomOf="@id/titleTextView" />

    <EditText
        android:id="@+id/messageEditText"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginStart="16dp"
        android:layout_marginTop="16dp"
        android:layout_marginEnd="16dp"
        android:background="@drawable/edit_text_background"
        android:hint="Enter the message...!"
        android:inputType="textMultiLine"
        android:minLines="3"
        android:padding="12dp"
        android:textSize="16sp"
        app:layout_constraintTop_toBottomOf="@id/phoneNumberEditText" />

    <Button
        android:id="@+id/sendButton"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginStart="16dp"
        android:layout_marginTop="16dp"
        android:layout_marginEnd="16dp"
        android:background="@drawable/button_background"
        android:padding="12dp"
        android:text="Send"
        android:textColor="@color/white"
        android:textSize="16sp"
        app:layout_constraintTop_toBottomOf="@id/messageEditText" />

</androidx.constraintlayout.widget.ConstraintLayout>
```

MainActivity.kt

```
package com.example.sms

import android.Manifest
import android.content.pm.PackageManager
import android.os.Build
import android.os.Bundle
import android.telephony.SmsManager
import android.widget.Button
import android.widget.EditText
import android.widget.Toast
import androidx.appcompat.app.AlertDialog
import android.appcompat.app.AppCompatActivity
import android.core.app.ActivityCompat
import androidx.core.content.ContextCompat

class MainActivity : AppCompatActivity() {
   private lateinit var phoneNumberEditText: EditText
   private lateinit var messageEditText: EditText
   private lateinit var sendButton: Button

   private val SMS_PERMISSION_CODE = 100

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

       // Initialize views
       phoneNumberEditText = findViewById(R.id.phoneNumberEditText)
       messageEditText = findViewById(R.id.messageEditText)
```

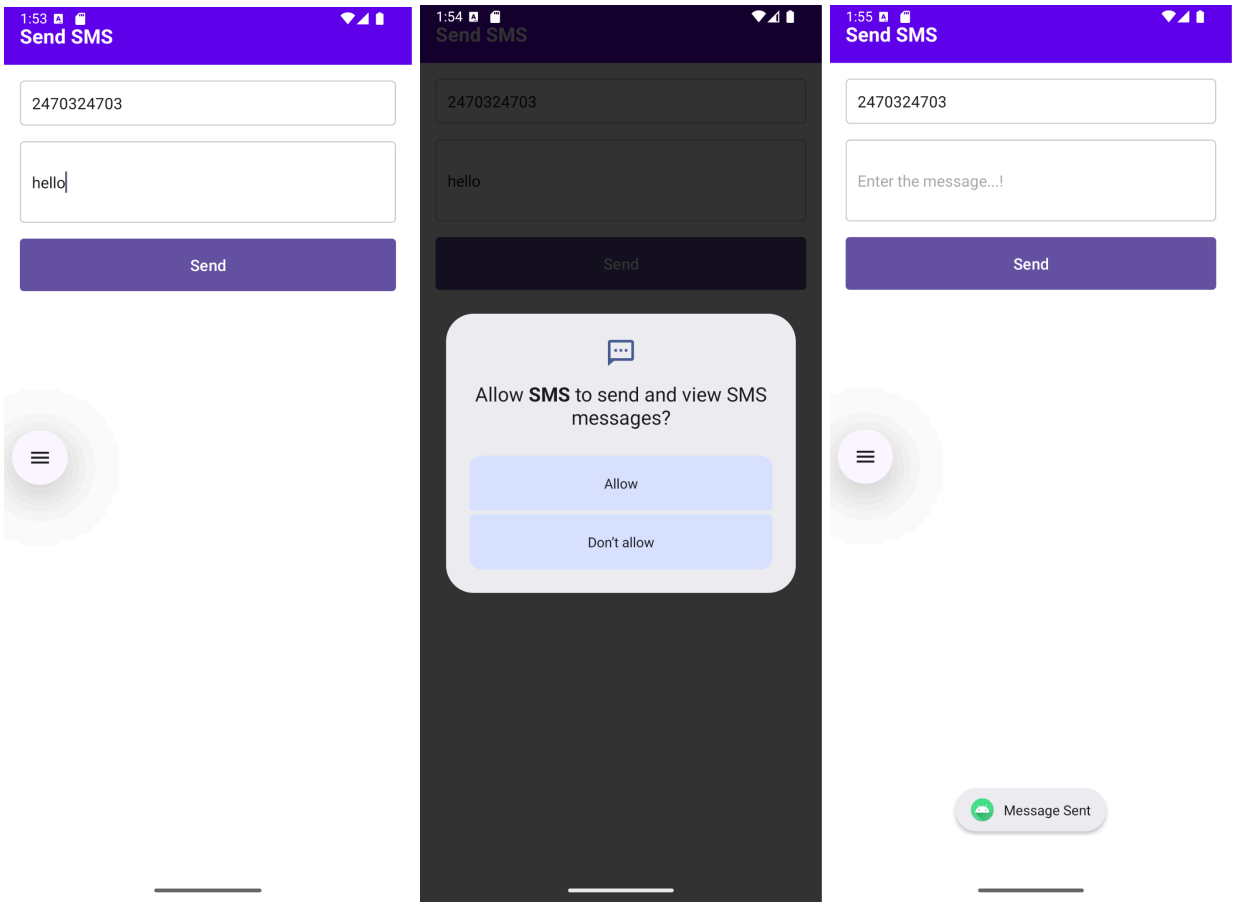
```
        sendButton = findViewById(R.id.sendButton)

        // Set click listener for the send button
        sendButton.setOnClickListener {
            if (checkSmsPermission()) {
                sendSMS()
            } else {
                requestSmsPermission()
            }
        }
    }

    private fun checkSmsPermission(): Boolean {
        return ContextCompat.checkSelfPermission(
            this,
            Manifest.permission.SEND_SMS
        ) == PackageManager.PERMISSION_GRANTED
    }

    private fun requestSmsPermission() {
        if (ActivityCompat.shouldShowRequestPermissionRationale(
                this,
                Manifest.permission.SEND_SMS
            )
        ) {
            // Show custom dialog explaining why we need this permission
            AlertDialog.Builder(this)
                .setTitle("Permission needed")
                .setMessage("Allow Send SMS to send and view SMS messages?")
                .setPositiveButton("ALLOW") { _, _ ->
                    ActivityCompat.requestPermissions(
```

```
                        this,
                        arrayOf(Manifest.permission.SEND_SMS),
                        SMS_PERMISSION_CODE
                    )
                }
                .setNegativeButton("DENY") { dialog, _ ->
                    dialog.dismiss()
                }
                .create()
                .show()
        } else {
            // Request permission directly
            ActivityCompat.requestPermissions(
                this,
                arrayOf(Manifest.permission.SEND_SMS),
                SMS_PERMISSION_CODE
            )
        }
    }

    override fun onRequestPermissionsResult(
        requestCode: Int,
        permissions: Array<out String>,
        grantResults: IntArray
    ) {
        super.onRequestPermissionsResult(requestCode, permissions,
grantResults)
        if (requestCode == SMS_PERMISSION_CODE) {
            if (grantResults.isNotEmpty() && grantResults[0] ==
PackageManager.PERMISSION_GRANTED) {
                // Permission granted, send the SMS
                sendSMS()
```

```
            } else {
                // Permission denied
                Toast.makeText(
                    this,
                    "SMS permission denied. Cannot send messages.",
                    Toast.LENGTH_SHORT
                ).show()
            }
        }
    }

    private fun sendSMS() {
        val phoneNumber = phoneNumberEditText.text.toString().trim()
        val message = messageEditText.text.toString().trim()

        if (phoneNumber.isEmpty() || message.isEmpty()) {
            Toast.makeText(this, "Please enter both phone number and message",
Toast.LENGTH_SHORT).show()
            return
        }

        try {
            val smsManager = if (Build.VERSION.SDK_INT >=
Build.VERSION_CODES.S) {
                this.getSystemService(SmsManager::class.java)
            } else {
                @Suppress("DEPRECATION")
                SmsManager.getDefault()
            }

            smsManager.sendTextMessage(phoneNumber, null, message, null, null)
            Toast.makeText(this, "Message Sent", Toast.LENGTH_SHORT).show()
```

```
            // Clear fields after sending
            messageEditText.text.clear()
        } catch (e: Exception) {
            Toast.makeText(this, "Failed to send SMS: ${e.message}",
Toast.LENGTH_SHORT).show()
            e.printStackTrace()
        }
    }
}
```

Output



Result:

The SMS application successfully sends text messages to specified phone numbers using Kotlin code on a mobile device with SMS permission granted.