

Department of Computer Science and Engineering

Shivanirudh S G, 185001146, Semester VII

18 October 2021

UCS1711 - Mobile Application Development Lab

Exercise 8: Application that send a SMS and creates an alert upon receiving the SMS

Aim:

Implement an application that send a SMS and creates an alert upon receiving the SMS.

Code:

Main Activity:

Design:

```

1 <?xml version="1.0" encoding="utf-8"?>
2 <androidx.constraintlayout.widget.ConstraintLayout
   xmlns:android="http://schemas.android.com/apk/res/android"
3   xmlns:app="http://schemas.android.com/apk/res-auto"
4   xmlns:tools="http://schemas.android.com/tools"
5   android:layout_width="match_parent"
6   android:layout_height="match_parent"
7   tools:context=".MainActivity">
8
9   <TextView
10     android:id="@+id/textView1"
11     android:layout_width="wrap_content"
12     android:layout_height="wrap_content"
13     android:layout_alignParentTop="true"
14     android:layout_centerHorizontal="true"
15     android:layout_marginTop="88dp"
16     android:text="SMS"
17     android:textSize="30dp"
18     app:layout_constraintEnd_toEndOf="parent"
19     app:layout_constraintHorizontal_bias="0.498"
20     app:layout_constraintStart_toStartOf="parent"
21     app:layout_constraintTop_toTopOf="parent" />
22
23   <EditText
24     android:id="@+id/editTextNumber"
25     android:layout_width="wrap_content"
26     android:layout_height="wrap_content"
27     android:layout_below="@+id/imageButton"
28     android:layout_centerHorizontal="true"
29     android:layout_marginTop="216dp"
30     android:hint="Enter Phone Number"
31     android:phoneNumber="true"
32     android:textColorHint="@color/
abc_primary_text_material_dark"
33     app:layout_constraintEnd_toEndOf="parent"
34     app:layout_constraintStart_toStartOf="parent"
35     app:layout_constraintTop_toTopOf="parent" />
36
37   <EditText
38     android:id="@+id/editTextMsg"
39     android:layout_width="224dp"
40     android:layout_height="87dp"
41     android:layout_below="@+id/editTextNumber"
42     android:layout_alignStart="@+id/editTextNumber"
43     android:layout_alignLeft="@+id/editTextNumber"

```

```

44         android:layout_marginTop="328dp"
45         android:hint="Enter message"
46         android:textColorHint="@color/
abc_primary_text_material_dark"
47         app:layout_constraintEnd_toEndOf="parent"
48         app:layout_constraintStart_toStartOf="parent"
49         app:layout_constraintTop_toTopOf="parent" />
50
51     <Button
52         android:id="@+id/btnSend"
53         android:layout_width="wrap_content"
54         android:layout_height="wrap_content"
55         android:layout_below="@+id/editTextMsg"
56         android:layout_centerHorizontal="true"
57         android:layout_marginTop="472dp"
58         android:onClick="sendSMSMessage"
59         android:text="Send"
60         app:layout_constraintEnd_toEndOf="parent"
61         app:layout_constraintHorizontal_bias="0.498"
62         app:layout_constraintStart_toStartOf="parent"
63         app:layout_constraintTop_toTopOf="parent" />
64
65
66 </androidx.constraintlayout.widget.ConstraintLayout>

```

Behaviour:

```

1 package com.example.sms;
2
3 import androidx.appcompat.app.AppCompatActivity;
4 import androidx.core.app.ActivityCompat;
5 import androidx.core.content.ContextCompat;
6
7 import android.Manifest;
8 import android.content.pm.PackageManager;
9 import android.os.Bundle;
10
11 import android.telephony.SmsManager;
12
13 import android.util.Log;
14 import android.view.Menu;
15 import android.view.View;
16
17 import android.widget.Button;
18 import android.widget.EditText;

```

```

19 import android.widget.Toast;
20
21 public class MainActivity extends AppCompatActivity {
22
23     private static final int REQUEST_SEND_SMS_PERMISSION =1 ;
24     Button sendBtn;
25     EditText phoneNo;
26     EditText message;
27     String number;
28     String msg_content;
29
30     @Override
31     protected void onCreate(Bundle savedInstanceState) {
32         super.onCreate(savedInstanceState);
33         setContentView(R.layout.activity_main);
34
35         sendBtn = (Button) findViewById(R.id.btnSend);
36         phoneNo = (EditText) findViewById(R.id.editTextNumber
37 );
38         message = (EditText) findViewById(R.id.editTextMsg);
39     }
40
41     public void sendSMSMessage(View view) {
42         number = phoneNo.getText().toString();
43         msg_content = message.getText().toString();
44
45         if (ContextCompat.checkSelfPermission(this, Manifest.
46 permission.SEND_SMS)
47 != PackageManager.PERMISSION_GRANTED) {
48             if (ActivityCompat.
49 shouldShowRequestPermissionRationale(this,
50 Manifest.permission.SEND_SMS)) {
51                 Toast.makeText(getApplicationContext(), "Need
52 permission.",
53                     Toast.LENGTH_LONG).show();
54             } else {
55                 ActivityCompat.requestPermissions(this,
56                     new String[]{Manifest.permission.
57 SEND_SMS},
58                     REQUEST_SEND_SMS_PERMISSION);
59             }
60         }
61     }
62
63     @Override

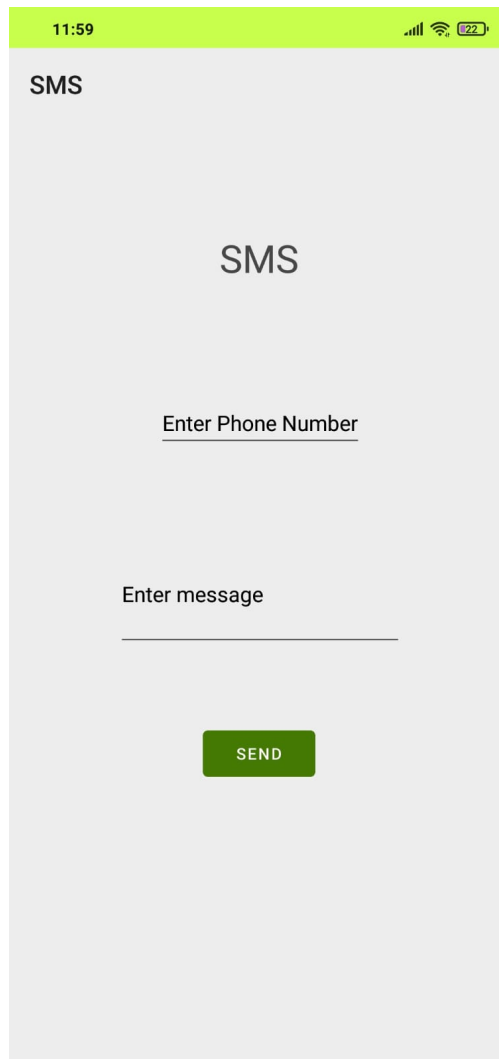
```

```

59     public void onRequestPermissionsResult(int requestCode,
String permissions[], int[] grantResults) {
60         Toast.makeText(getApplicationContext(), "control here
.",
61             Toast.LENGTH_LONG).show();
62         // super.onRequestPermissionsResult(requestCode,
permissions, grantResults);
63         switch (requestCode) {
64             case REQUEST_SEND_SMS_PERMISSION: {
65                 if (permissions[0].equalsIgnoreCase(Manifest.
permission.SEND_SMS)
66                     && grantResults[0] == PackageManager.
PERMISSION_GRANTED) {
67                     try {
68                         SmsManager smsManager = SmsManager.
getDefault();
69                         smsManager.sendTextMessage(number,
null, msg_content, null, null);
70                         Toast.makeText(getApplicationContext
(), "SMS sent.",
71                             Toast.LENGTH_LONG).show();
72                     } catch (Exception e) {
73                         Toast.makeText(getApplicationContext
(), e.toString(),
74                             Toast.LENGTH_LONG).show();
75                     }
76                 } else {
77                     Toast.makeText(getApplicationContext(),
78                         "SMS failed, please try again.",
79                         Toast.LENGTH_LONG).show();
80                     return;
81                 }
82             }
83         }
84     }
85 }
86 }

```

Output:



The image shows a mobile application interface for sending SMS. At the top, there is a green status bar with the time '11:59' and icons for cellular signal, Wi-Fi, and battery level at '22%'. Below the status bar, the word 'SMS' is displayed in the top left corner. In the center of the screen, the word 'SMS' is shown in a large, bold font. Below this, there are two input fields: the first is labeled 'Enter Phone Number' and the second is labeled 'Enter message'. Both labels are underlined. At the bottom center, there is a green button with the text 'SEND' in white capital letters.



