

ASSIGNMENT 8

SMS Messenger

SHASHANKA VENKATESH - 185001145

18th October, 2021

Learning Objective

Implement an application that sends an SMS

XML (activity_main.xml)

```
1 <?xml version="1.0" encoding="utf-8"?>
2 <RelativeLayout 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     <EditText
10         android:id="@+id/contactEditText"
11         android:layout_width="match_parent"
12         android:layout_height="wrap_content"
13         android:layout_marginStart="20dp"
14         android:layout_marginTop="20dp"
15         android:layout_marginEnd="20dp"
16         android:layout_marginBottom="20dp"
17         android:backgroundTint="@color/teal_700"
18         android:hint="Contact Number"
19         android:minHeight="48dp"
20         android:inputType="number"
21         android:maxLength="10"/>
22
23     <ScrollView
24         android:id="@+id/scrollView"
25         android:layout_width="match_parent"
```

```

26     android:layout_height="match_parent"
27     android:layout_above="@+id/messageEditText"
28     android:layout_below="@id/contactEditText">
29
30     <LinearLayout
31         android:layout_width="match_parent"
32         android:layout_height="wrap_content"
33         android:orientation="vertical">
34
35         <TextView
36             android:id="@+id/messageTxt"
37             android:layout_width="match_parent"
38             android:layout_height="wrap_content"
39             android:layout_margin="50px" />
40     </LinearLayout>
41 </ScrollView>
42
43 <EditText
44     android:id="@+id/messageEditText"
45     android:layout_width="match_parent"
46     android:layout_height="wrap_content"
47     android:layout_alignParentBottom="true"
48     android:layout_marginHorizontal="5dp"
49     android:layout_marginBottom="10dp"
50     android:layout_toLeftOf="@+id/sendBtn"
51     android:hint="Enter message"
52     android:minHeight="48dp" />
53
54 <Button
55     android:layout_width="wrap_content"
56     android:layout_height="wrap_content"
57     android:text="SEND"
58     android:id="@+id/sendBtn"
59     android:layout_alignParentBottom="true"
60     android:layout_alignParentRight="true"
61     android:layout_marginHorizontal="5dp"
62     android:layout_marginBottom="10dp"/>
63 </RelativeLayout>

```

Java (MainActivity.java)

```
1 package com.madlab.smsmessenger;
2
3 import androidx.appcompat.app.AppCompatActivity;
4
5 import androidx.core.app.ActivityCompat;
6 import android.app.PendingIntent;
7 import android.content.BroadcastReceiver;
8 import android.content.Context;
9 import android.content.Intent;
10 import android.content.IntentFilter;
11 import android.telephony.SmsManager;
12 import android.view.View;
13 import android.widget.Button;
14 import android.widget.EditText;
15 import android.widget.TextView;
16 import static android.Manifest.permission.READ_PHONE_STATE;
17 import static android.Manifest.permission.READ_SMS;
18 import static android.Manifest.permission.RECEIVE_SMS;
19 import static android.Manifest.permission.SEND_SMS;
20
21 import android.os.Bundle;
22 import android.widget.Toast;
23
24 public class MainActivity extends AppCompatActivity {
25     EditText contactEditText, messageEditText;
26     Button sendButton;
27     TextView messageText;
28     IntentFilter intentFilter;
29     private BroadcastReceiver intentReceiver = new BroadcastReceiver() {
30         @Override
31         public void onReceive(Context context, Intent intent) {
32             messageText.setText(intent.getExtras().getString("message"));
33         }
34     };
35
36     @Override
37     protected void onCreate(Bundle savedInstanceState) {
38         super.onCreate(savedInstanceState);
39         setContentView(R.layout.activity_main);
40
41         ActivityCompat.requestPermissions(MainActivity.this, new String[]
42 {
43             READ_SMS, RECEIVE_SMS, SEND_SMS, READ_PHONE_STATE
44 }, 1);
45
46         contactEditText = findViewById(R.id.contactEditTxt);
47         messageEditText = findViewById(R.id.messageEditTxt);
48         sendButton = findViewById(R.id.sendBtn);
49         messageText = findViewById(R.id.messageTxt);
```

```

49         intentFilter = new IntentFilter();
50         intentFilter.addAction("SMS_RECEIVED_ACTION");
51
52         sendButton.setOnClickListener(new View.OnClickListener() {
53             @Override
54             public void onClick(View view) {
55                 sendMsg(contactEditText.getText().toString(),
56 messageEditText.getText().toString());
57                 Toast.makeText(MainActivity.this, "Message Sent!", Toast.
LENGTH_LONG).show();
58                 messageEditText.setText("");
59             }
60         });
61     }
62
63     void sendMsg(String num,String myMsg) {
64         String SENT = "Message Sent";
65         String DELIVERED = "Message Delivered";
66         PendingIntent sentPI = PendingIntent.getBroadcast(this, 0, new
Intent(SENT), 0);
67         PendingIntent deliveredPI = PendingIntent.getBroadcast(this,0,new
Intent(DELIVERED),0);
68         SmsManager smsManager = SmsManager.getDefault();
69         smsManager.sendTextMessage(num,null, myMsg, sentPI, deliveredPI);
70     }
71
72     @Override
73     protected void onPause() {
74         unregisterReceiver(intentReceiver);
75         super.onPause();
76     }
77
78     @Override
79     protected void onResume() {
80         registerReceiver(intentReceiver,intentFilter);
81         super.onResume();
82     }
83 }

```

App Screenshots



