



**Ganpat
University**
॥ विद्यया समाजोत्कर्षः ॥

**U.V. Patel
College of
Engineering**

Practical - 13

AIM: Create an application to receive sms, send sms and display all inbox message into listview.

B.Tech. Semester - V
Computer Engineering/ Information Technology
Academic Year: 2021 (Odd Sem.)

Prepared By:

HINGRAJIYA ACHAL A.: 19012021012

Department of Information Technology Engineering

UVPCE - Ganpat University

Table Of Contents

| | |
|--|----|
| Aim: | 1 |
| Project structure: | 2 |
| Source Code: | 3 |
| Project Files | 3 |
| src\main\AndroidManifest.xml | 3 |
| res\menu\main_menu.xml | 4 |
| res\menu\top_app_bar.xml | 4 |
| Inbox Activity | 5 |
| practical12_19012021012\InboxActivity.kt | 5 |
| practical13_19012021012\sms_info\InboxMessages.kt..... | 8 |
| practical13_19012021012\adapters\InboxSmsAdapter.kt | 9 |
| res\layout\sms_card.xml | 11 |
| res\layout\activity_inbox.xml..... | 14 |
| Outbox Activity | 15 |
| practical13_19012021012\OutboxActivity.kt | 15 |
| practical13_19012021012\sms_info\OutboxMessages.kt..... | 18 |
| practical13_19012021012\adapters\OutboxSmsAdapter.kt | 19 |
| res\layout\activity_outbox.xml | 21 |
| Output: | 23 |

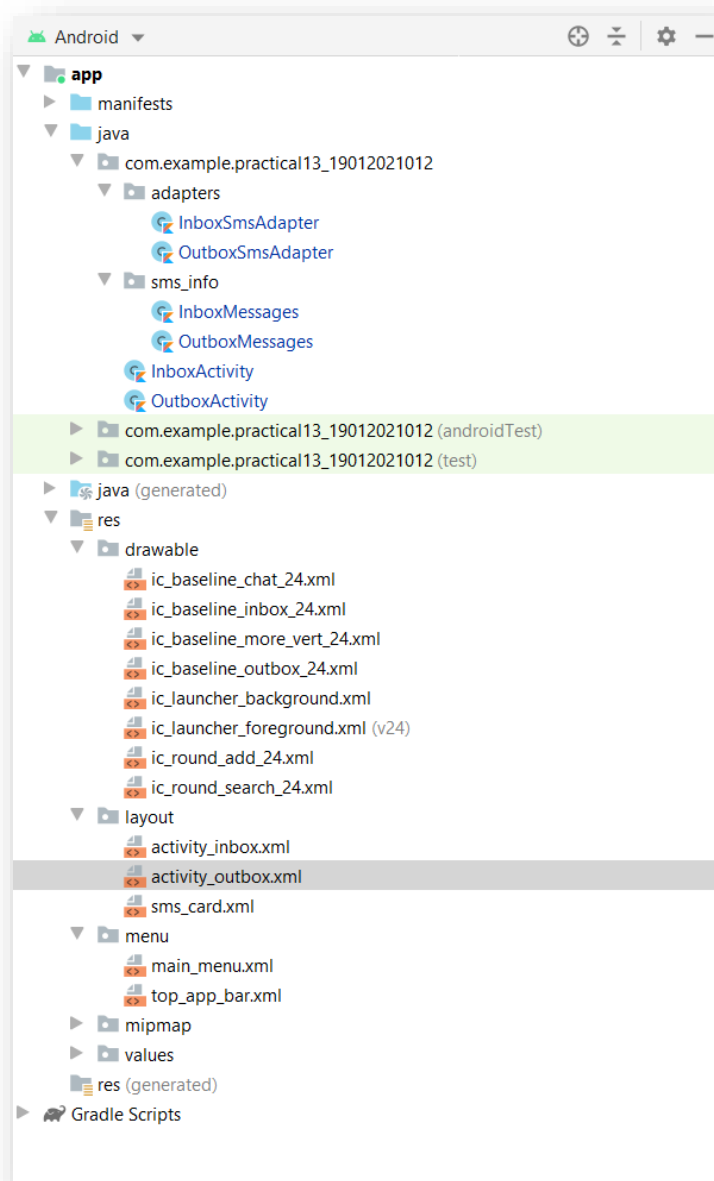
Table of figures

- 1. Project Structure.....2
- 2. SMS Card Layout.....11
- 3. Inbox Activity Layout14
- 4. Outbox Activity Layout.....21
- 5. Inbox Activity Output.....23
- 6. Outbox Activity Output.....23
- 7. Inbox Activity New SMS Received (Which is sent by extended controls).....23
- 8. First received SMS.....24
- 9. Confirming from Messages App.....24
- 10. List of received SMSs24
- 11. Outbox Activity: Creating new SMS.....24
- 12. New SMS Sent successfully.....25
- 13. Confirming that all SMSs are sent.....25

PRACTICAL – 13

Aim: Create an application to receive sms, send sms and display all inbox message into listview.

Project structure:



1. Project Structure

Source Code:

Project Files

src\main\AndroidManifest.xml

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.example.practical13_19012021012">

    <uses-permission android:name="android.permission.SEND_SMS"/>
    <uses-permission android:name="android.permission.READ_SMS"/>
    <uses-permission android:name="android.permission.RECEIVE_SMS"/>

    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportRtl="true"
        android:theme="@style/Theme.Practical13_19012021012">
        <activity android:name=".InboxActivity" android:noHistory="true">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />
                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
        <activity android:name=".OutboxActivity" android:noHistory="true"/>

    </application>

</manifest>
```

res\menu\main_menu.xml

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

    <item
        android:id="@+id/inbox"
        android:title="Inbox"
        android:icon="@drawable/ic_baseline_inbox_24"/>
    <item
        android:id="@+id/outbox"
        android:title="Outbox"
        android:icon="@drawable/ic_baseline_outbox_24"/>
</menu>
```

res\menu\top_app_bar.xml

```
<menu xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto">

    <item
        android:id="@+id/search"
        android:icon="@drawable/ic_round_search_24"
        android:title="Search"
        android:contentDescription="Search"
        app:showAsAction="ifRoom" />

    <item
        android:id="@+id/more"
        android:title="More"
        android:icon="@drawable/ic_baseline_more_vert_24"
        android:contentDescription="More"
        app:showAsAction="ifRoom" />

</menu>
```

Inbox Activity

Used to display received messages.

practical12_19012021012\InboxActivity.kt

```
package com.example.practical13_19012021012

import android.Manifest
import android.app.AlertDialog
import android.content.*
import android.content.pm.PackageManager
import android.os.Build
import androidx.appcompat.app.AppCompatActivity
import android.os.Bundle
import android.provider.Telephony
import android.view.LayoutInflater
import android.view.View
import android.widget.EditText
import android.widget.ListView
import androidx.annotation.RequiresApi
import androidx.core.app.ActivityCompat
import com.example.practical13_19012021012.adapters.InboxSmsAdapter
import com.example.practical13_19012021012.sms_info.InboxMessages
import com.google.android.material.bottomnavigation.BottomNavigationView

class InboxActivity : AppCompatActivity() {

    lateinit var lvAdapter : InboxSmsAdapter

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

        val bnav = findViewById<BottomNavigationView>(R.id.bnav)
        bnav.setOnItemSelectedListener {
            when (it.itemId) {
                R.id.outbox -> {
                    Intent(this, OutboxActivity::class.java).apply {
                        startActivity(this)
                    }
                    return@setOnItemSelectedListener true
                }
            }
            else ->
                return@setOnItemSelectedListener true
        }
    }
}
```



```

val lvInboxSms = findViewById<ListView>(R.id.lv_inbox_sms)

lvAdapter = InboxSmsAdapter(this, InboxMessages.inboxSmsArr)
lvInboxSms.adapter = lvAdapter

if (ActivityCompat.checkSelfPermission(
    this,
    Manifest.permission.RECEIVE_SMS
) != PackageManager.PERMISSION_GRANTED
){
    ActivityCompat.requestPermissions(
        this,
        arrayOf(
            Manifest.permission.RECEIVE_SMS,
            Manifest.permission.SEND_SMS,
            Manifest.permission.READ_SMS
        ),
        111
    )
} else {
    receiveMsg()
}
}

override fun onRequestPermissionsResult(
    requestCode: Int,
    permissions: Array<out String>,
    grantResults: IntArray
){
    super.onRequestPermissionsResult(requestCode, permissions, grantResults)

    if (requestCode == 111
        && grantResults[0] == PackageManager.PERMISSION_GRANTED
    ){
        receiveMsg()
    }
}

private fun receiveMsg() {
    val br = object : BroadcastReceiver() {
        @RequiresApi(Build.VERSION_CODES.N)
        override fun onReceive(context: Context?, intent: Intent?) {

            if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.KITKAT) {
                for (sms in Telephony.Sms.Intents.getMessagesFromIntent(intent)) {

                    val contactNo = sms.originatingAddress.toString()
                    val body = sms.displayMessageBody.toString()

                    showSmsDialog(contactNo, body)
                }
            }
        }
    }
}

```

```

        val newSms = InboxMessages(contactNo, body)
        InboxMessages.addSMS(newSms)
        lvAdapter.notifyDataSetChanged()
    }
}
}
}
registerReceiver(br, IntentFilter("android.provider.Telephony.SMS_RECEIVED"))
}

@RequiresApi(Build.VERSION_CODES.N)
private fun showSmsDialog(contactNo : String, smsBody : String) {
    val dialogTitle = "New SMS Received"
    val builder: AlertDialog.Builder = AlertDialog.Builder(this)
    builder.setTitle(dialogTitle)

    val customLayout: View = LayoutInflater.from(this).inflate(R.layout.sms_card, null)

    val tvContactNo = customLayout.findViewById<EditText>(R.id.tv_contact_no)
    val tvSmsBody = customLayout.findViewById<EditText>(R.id.tv_sms_body)

    if (android.os.Build.VERSION.SDK_INT >= android.os.Build.VERSION_CODES.O) {
        tvContactNo.focusable = View.NOT_FOCUSABLE
        tvSmsBody.focusable = View.NOT_FOCUSABLE
    }

    tvContactNo.isFocusableInTouchMode = false
    tvContactNo.setBackgroundResource(android.R.color.transparent)
    tvSmsBody.isFocusableInTouchMode = false
    tvSmsBody.setBackgroundResource(android.R.color.transparent)

    tvContactNo.setText(contactNo)
    tvSmsBody.setText(smsBody)

    builder.setView(customLayout)
    builder.setPositiveButton(
        "Close",
        DialogInterface.OnClickListener { _, _ ->

    })

    val dialog: AlertDialog = builder.create()

    dialog.show()

}
}
}

```

practical13_19012021012\sms_info\InboxMessages.kt

```
package com.example.practical13_19012021012.sms_info

class InboxMessages (var contactNo : String, var body : String){

    var id = idGeneration()

    companion object {

        var id : Long = 0

        fun idGeneration(): Long {
            id += 1
            return id
        }

        fun addSMS(sms : InboxMessages){
            inboxSmsArr.add(sms)
        }
        var inboxSmsArr: ArrayList<InboxMessages> = ArrayList()
    }
}
```

practical13_19012021012\adapters\InboxSmsAdapter.kt

```
package com.example.practical13_19012021012.adapters

import android.annotation.SuppressLint
import android.content.Context
import android.os.Build
import android.view.LayoutInflater
import android.view.View
import android.view.ViewGroup
import android.widget.BaseAdapter
import android.widget.EditText
import androidx.annotation.RequiresApi
import com.example.practical13_19012021012.R
import com.example.practical13_19012021012.sms_info.InboxMessages

class InboxSmsAdapter(private val context : Context, private val dataSource : ArrayList<InboxMessages>) :
BaseAdapter() {
    override fun getCount(): Int {
        return dataSource.size
    }

    override fun getItem(position: Int): Any {
        return dataSource[position]
    }

    override fun getItemId(position: Int): Long {
        return dataSource[position].id
    }

    @RequiresApi(Build.VERSION_CODES.O)
    @SuppressLint("ViewHolder")
    override fun getView(position: Int, convertView: View?, parent: ViewGroup?): View {

        val inflater = context.getSystemService(Context.LAYOUT_INFLATER_SERVICE) as LayoutInflater

        val rowView = inflater.inflate(R.layout.sms_card, parent, false)
        val tvContactNo = rowView.findViewById<EditText>(R.id.tv_contact_no)
        val tvSmsBody = rowView.findViewById<EditText>(R.id.tv_sms_body)
        tvContactNo.focusable = View.NOT_FOCUSABLE
        tvContactNo.isFocusableInTouchMode = false
        tvContactNo.setBackgroundResource(android.R.color.transparent)
        tvSmsBody.focusable = View.NOT_FOCUSABLE
        tvSmsBody.isFocusableInTouchMode = false
        tvSmsBody.setBackgroundResource(android.R.color.transparent)

        val sms = getItem(position) as InboxMessages
```

```
        tvContactNo.setText(sms.contactNo)
        tvSmsBody.setText(sms.body)

        return rowView
    }
}
```

res\layout\sms_card.xml

```

<?xml version="1.0" encoding="utf-8"?>

<androidx.cardview.widget.CardView
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_alignParentRight="true"
    android:layout_alignParentLeft="true"
    android:layout_alignParentEnd="true"
    android:layout_alignParentStart="true"
    android:layout_marginHorizontal="20dp"
    android:layout_centerInParent="true"
    app:cardCornerRadius="10dp">

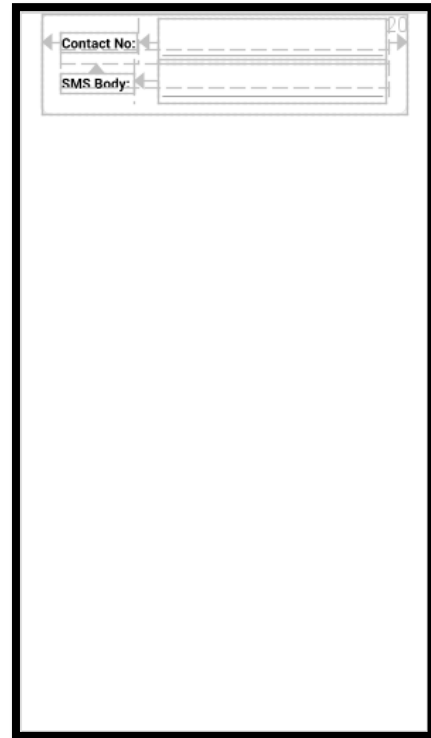
    <RelativeLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:paddingVertical="10dp">

        <TextView
            android:id="@+id/lbl_contact_no"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_alignParentStart="true"
            android:layout_alignParentLeft="true"
            android:layout_marginHorizontal="20dp"
            android:layout_marginVertical="10dp"
            android:textColor="@color/black"
            android:textStyle="bold"
            android:text="Contact No: " />

        <EditText
            android:id="@+id/tv_contact_no"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_alignParentTop="true"
            android:layout_alignBaseline="@+id/lbl_contact_no"
            android:layout_toEndOf="@id/lbl_contact_no"
            android:layout_toRightOf="@id/lbl_contact_no"
            android:layout_alignParentRight="true"
            android:layout_marginRight="20dp"
            android:layout_alignParentEnd="true"
            android:layout_marginEnd="20dp" />

        <TextView

```



2. SMS Card Layout

```
android:id="@+id/lbl_sms_body"  
android:layout_width="wrap_content"  
android:layout_height="wrap_content"  
android:layout_alignLeft="@+id/lbl_contact_no"  
android:layout_marginVertical="10dp"  
android:layout_below="@+id/tv_contact_no"  
android:textColor="@color/black"  
android:textStyle="bold"  
android:text="SMS Body: "  
android:layout_alignStart="@+id/lbl_contact_no" />
```

```
<EditText  
android:id="@+id/tv_sms_body"  
android:layout_width="wrap_content"  
android:layout_height="wrap_content"  
android:layout_alignParentTop="true"  
android:layout_alignBaseline="@+id/lbl_sms_body"  
android:layout_alignStart="@id/tv_contact_no"  
android:layout_toEndOf="@id/lbl_sms_body"  
android:layout_toRightOf="@id/lbl_sms_body"  
android:layout_alignRight="@+id/tv_contact_no"  
android:layout_alignLeft="@id/tv_contact_no"  
android:layout_alignEnd="@+id/tv_contact_no" />
```

```
</RelativeLayout>  
</androidx.cardview.widget.CardView>
```


res\layout\activity_inbox.xml

```

<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout
    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"
    tools:context=".InboxActivity"
    android:fitsSystemWindows="true"
    >
    <com.google.android.material.appbar.AppBarLayout
        android:id="@+id/topAppBarLayout"
        android:layout_width="match_parent"
        android:layout_height="wrap_content">

        <com.google.android.material.appbar.MaterialToolbar
            android:id="@+id/topAppBar"
            android:layout_width="match_parent"
            android:layout_height="?attr/actionBarSize"
            app:title="Messages"
            app:menu="@menu/top_app_bar"
            app:titleCentered="true"
            style="@style/Widget.MaterialComponents.Toolbar.Primary"
            />

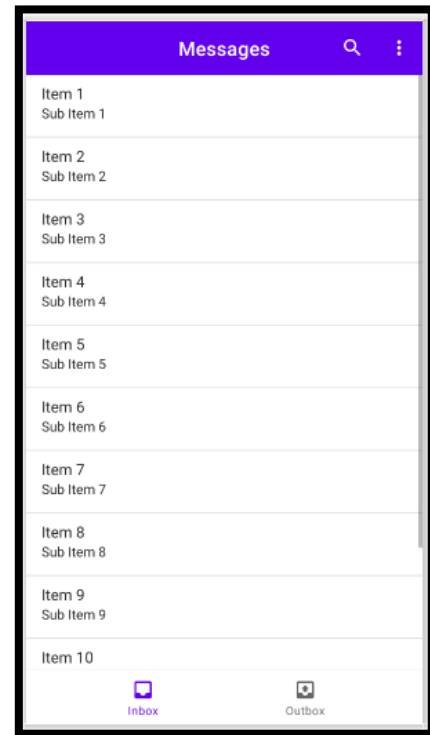
    </com.google.android.material.appbar.AppBarLayout>

    <ListView
        android:id="@+id/lv_inbox_sms"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:layout_alignParentTop="false"
        android:layout_above="@+id/bnav"
        android:layout_below="@+id/topAppBarLayout"/>

    <com.google.android.material.bottomnavigation.BottomNavigationView
        android:id="@+id/bnav"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        app:menu="@menu/main_menu"
        android:layout_alignParentBottom="true"
        android:layout_alignParentRight="true"
        android:layout_alignParentLeft="true"
        android:layout_alignParentEnd="true"
        android:layout_alignParentStart="true" />

</RelativeLayout>

```



3. Inbox Activity Layout

Outbox Activity

Used to display sent messages and send new sms.

practical13_19012021012\OutboxActivity.kt

```
package com.example.practical13_19012021012

import android.Manifest
import android.app.AlertDialog
import android.content.DialogInterface
import android.content.Intent
import android.content.pm.PackageManager
import android.os.Build
import androidx.appcompat.app.AppCompatActivity
import android.os.Bundle
import android.telephony.SmsManager
import android.view.LayoutInflater
import android.view.View
import android.widget.Button
import android.widget.EditText
import android.widget.ListView
import android.widget.Toast
import androidx.annotation.RequiresApi
import androidx.core.app.ActivityCompat
import com.example.practical13_19012021012.adapters.OutboxSmsAdapter
import com.example.practical13_19012021012.sms_info.OutboxMessages
import com.google.android.material.bottomnavigation.BottomNavigationView

class OutboxActivity : AppCompatActivity() {

    lateinit var lvAdapter : OutboxSmsAdapter

    @RequiresApi(Build.VERSION_CODES.N)
    override fun onCreate(savedInstanceState: Bundle?) {

        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_outbox)

        val bnav = findViewById<BottomNavigationView>(R.id.bnav)

        bnav.selectedItemId = R.id.outbox

        bnav.setOnItemSelectedListener {
```

```
        when (it.itemId) {

            R.id.inbox -> {
                Intent(this, InboxActivity::class.java).apply {

                    startActivity(this)

                }

                return@setOnItemSelectedListener true
            }
            else ->
                return@setOnItemSelectedListener true
        }
    }

    val lvInboxSms = findViewById<ListView>(R.id.lv_outbox_sms)

    lvAdapter = OutboxSmsAdapter(this, OutboxMessages.outboxSmsArr)

    lvInboxSms.adapter = lvAdapter

    if (ActivityCompat.checkSelfPermission(
        this,
        Manifest.permission.RECEIVE_SMS
    ) != PackageManager.PERMISSION_GRANTED
    ){

        ActivityCompat.requestPermissions(
            this,
            arrayOf(
                Manifest.permission.RECEIVE_SMS,
                Manifest.permission.SEND_SMS,
                Manifest.permission.READ_SMS
            ),
            111
        )
    }

    val btnNewSms = findViewById<Button>(R.id.btn_new_sms)

    btnNewSms.setOnClickListener {
        showNewSmsDialog()
    }
}
```

```
@RequiresApi(Build.VERSION_CODES.N)
private fun showNewSmsDialog() {

    val dialogTitle = "Send SMS"
    val builder: AlertDialog.Builder = AlertDialog.Builder(this)
    builder.setTitle(dialogTitle)

    val customLayout: View = LayoutInflater.from(this).inflate(R.layout.sms_card, null)

    builder.setView(customLayout)
    builder.setPositiveButton(
        "Send",
        DialogInterface.OnClickListener { dialog, which ->

            val tvContactNo = customLayout.findViewById<EditText>(R.id.tv_contact_no)
            val tvBody = customLayout.findViewById<EditText>(R.id.tv_sms_body)
            val contactNo = tvContactNo.text.toString()
            val body = tvBody.text.toString()

            val sms = SmsManager.getDefault()

            sms.sendTextMessage(contactNo, "Achal Hingrajiya", body, null, null)

            Toast.makeText(this, "SMS Sent", Toast.LENGTH_SHORT).show()

            val newSms = OutboxMessages(contactNo, body)
            OutboxMessages.addSMS(newSms)

            lvAdapter.notifyDataSetChanged()

        })

    val dialog: AlertDialog = builder.create()

    dialog.show()

}
}
```

practical13_19012021012\sms_info\OutboxMessages.kt

```
package com.example.practical13_19012021012.sms_info

class OutboxMessages (var contactNo : String, var body : String){

    var id = idGeneration()

    companion object {

        var id : Long = 0

        fun idGeneration(): Long {
            id += 1
            return id
        }

        fun addSMS(sms : OutboxMessages){
            outboxSmsArr.add(sms)
        }
        var outboxSmsArr: ArrayList<OutboxMessages> = ArrayList()
    }
}
```

practical13_19012021012\adapters\OutboxSmsAdapter.kt

```
package com.example.practical13_19012021012.adapters

import android.annotation.SuppressLint
import android.content.Context
import android.os.Build
import android.view.LayoutInflater
import android.view.View
import android.view.ViewGroup
import android.widget.BaseAdapter
import android.widget.EditText
import androidx.annotation.RequiresApi
import com.example.practical13_19012021012.R
import com.example.practical13_19012021012.sms_info.OutboxMessages

class OutboxSmsAdapter(private val context : Context, private val dataSource : ArrayList<OutboxMessages>) :
BaseAdapter() {
    override fun getCount(): Int {
        return dataSource.size
    }

    override fun getItem(position: Int): Any {
        return dataSource[position]
    }

    override fun getItemId(position: Int): Long {
        return dataSource[position].id
    }

    @RequiresApi(Build.VERSION_CODES.O)
    @SuppressLint("ViewHolder")
    override fun getView(position: Int, convertView: View?, parent: ViewGroup?): View {

        val inflater = context.getSystemService(Context.LAYOUT_INFLATER_SERVICE) as LayoutInflater

        val rowView = inflater.inflate(R.layout.sms_card, parent, false)
        val tvContactNo = rowView.findViewById<EditText>(R.id.tv_contact_no)
        val tvSmsBody = rowView.findViewById<EditText>(R.id.tv_sms_body)
        tvContactNo.focusable = View.NOT_FOCUSABLE
        tvContactNo.isFocusableInTouchMode = false
        tvContactNo.setBackgroundResource(android.R.color.transparent)
        tvSmsBody.focusable = View.NOT_FOCUSABLE
        tvSmsBody.isFocusableInTouchMode = false
        tvSmsBody.setBackgroundResource(android.R.color.transparent)
```

```
    val sms = getItem(position) as OutboxMessages

    tvContactNo.setText(sms.contactNo)
    tvSmsBody.setText(sms.body)

    return rowView
}
}
```

res\layout\activity_outbox.xml

```

<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout
    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"
    tools:context=".OutboxActivity"
    android:fitsSystemWindows="true"
    >

    <com.google.android.material.appbar.AppBarLayout
        android:id="@+id/topAppBarLayout"
        android:layout_width="match_parent"
        android:layout_height="wrap_content">

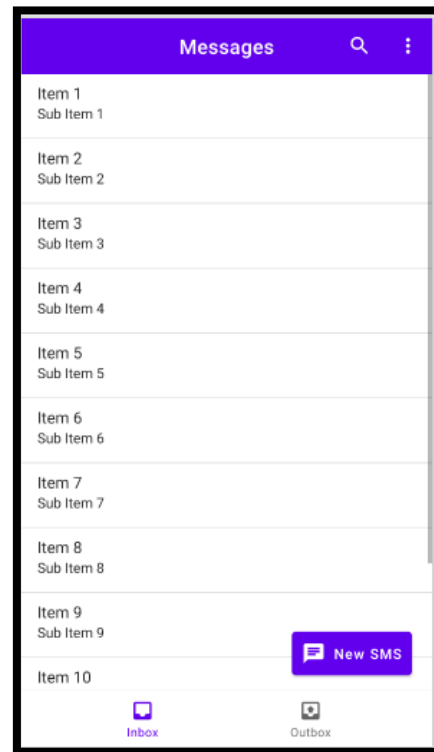
        <com.google.android.material.appbar.MaterialToolbar
            android:id="@+id/topAppBar"
            android:layout_width="match_parent"
            android:layout_height="?attr/actionBarSize"
            app:title="Messages"
            app:menu="@menu/top_app_bar"
            app:titleCentered="true"
            style="@style/Widget.MaterialComponents.Toolbar.Primary"
            />

    </com.google.android.material.appbar.AppBarLayout>

    <ListView
        android:id="@+id/lv_outbox_sms"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:layout_alignParentTop="false"
        android:layout_above="@+id/bnav"
        android:layout_below="@+id/topAppBarLayout"/>

    <Button
        android:id="@+id/btn_new_sms"
        style="@style/Widget.MaterialComponents.Button.Icon"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignParentRight="true"
        android:layout_alignParentEnd="true"
        android:layout_above="@+id/bnav"
        android:layout_marginBottom="10dp"
        android:layout_marginRight="20dp"
        app:icon="@drawable/ic_baseline_chat_24"

```



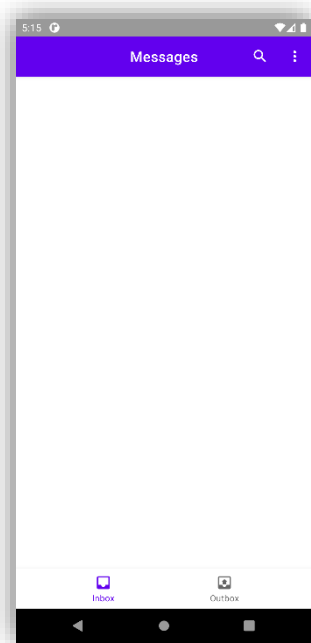
4. Outbox Activity Layout


```
android:text="New SMS"  
android:textAllCaps="false"  
android:padding="10dp"  
android:layout_marginEnd="20dp" />
```

```
<com.google.android.material.bottomnavigation.BottomNavigationView  
  android:id="@+id/bnav"  
  android:layout_width="match_parent"  
  android:layout_height="wrap_content"  
  app:menu="@menu/main_menu"  
  android:layout_alignParentBottom="true"  
  android:layout_alignParentRight="true"  
  android:layout_alignParentLeft="true"  
  android:layout_alignParentEnd="true"  
  android:layout_alignParentStart="true" />
```

```
</RelativeLayout>
```

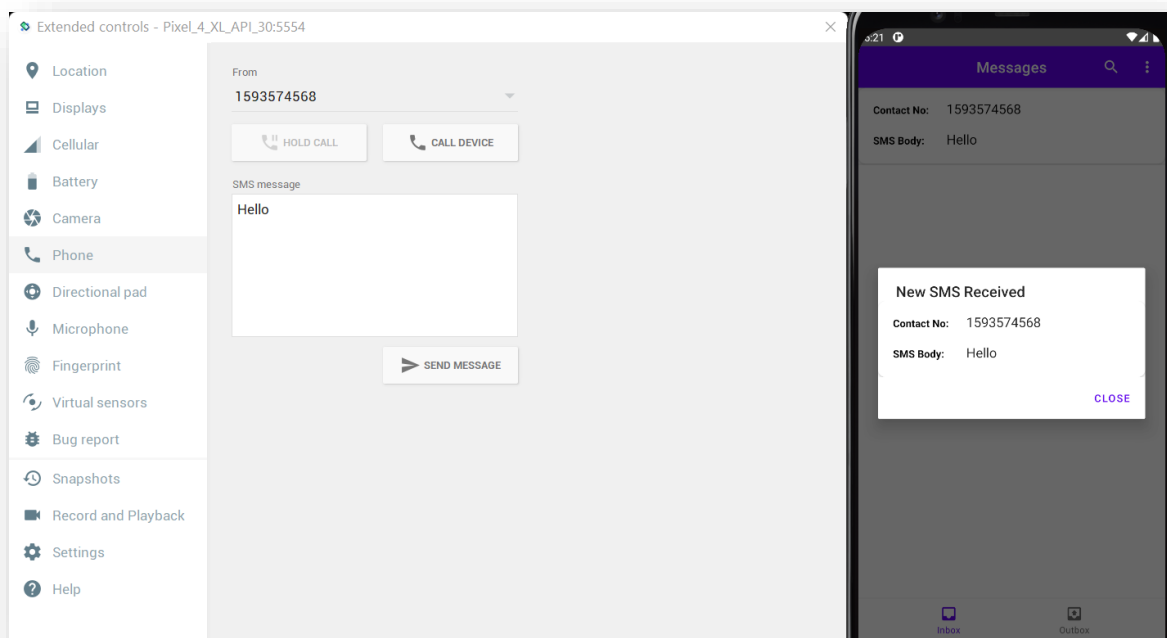
Output:



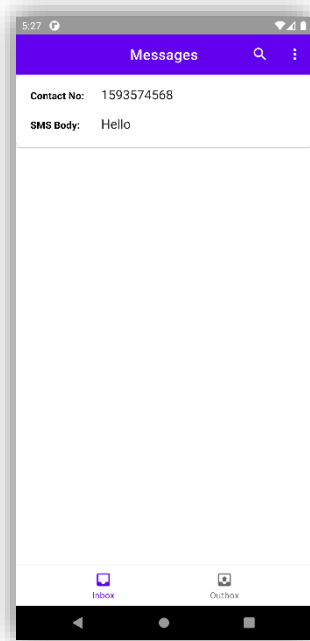
5. Inbox Activity Output



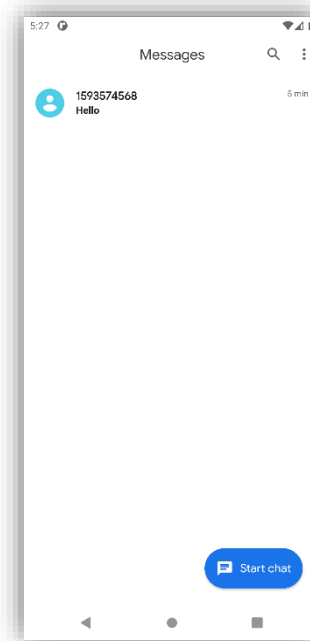
6. Outbox Activity Output



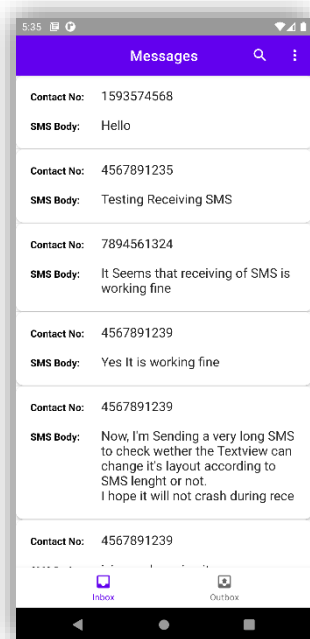
7. Inbox Activity New SMS Received (Which is sent by extended controls)



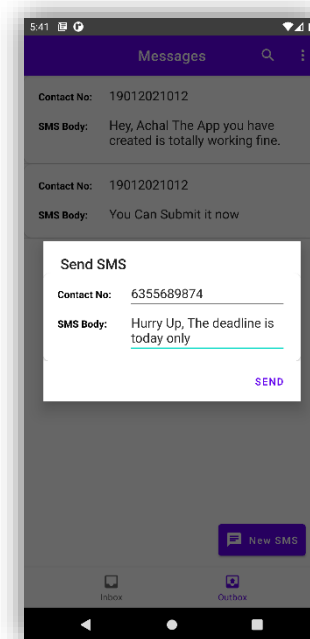
8. First received SMS



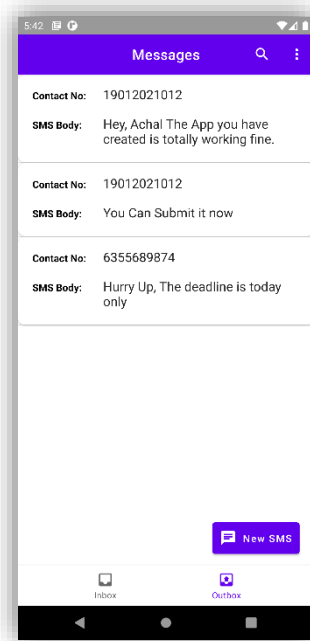
9. Confirming from Messages App



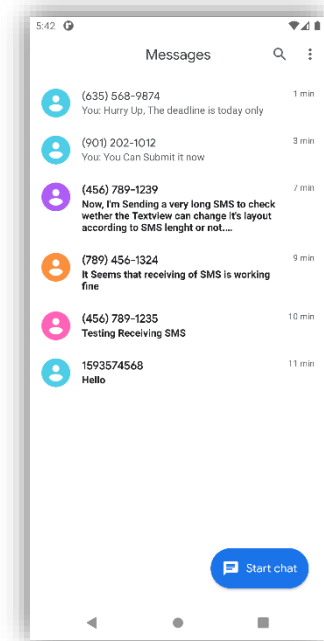
10. List of received SMSs



11. Outbox Activity: Creating new SMS



12. New SMS Sent successfully



13. Confirming that all SMSs are sent