

# Modul 6 – Broadcast Receiver

*Lihat riwayat »*

---

## Tujuan

---

Pada materi kali ini, Anda akan belajar menerapkan dua mekanisme umum dalam memanfaatkan komponen broadcast receiver yaitu :

- Implementasi broadcast receiver untuk menerima *event* yang di-*broadcast* oleh sistem Android. Kali ini kita akan menerima *broadcast* ketika ada sms yang masuk.
- Implementasi broadcast receiver untuk komunikasi antar aplikasi

Ayo kita mulai,

1. Buat project aplikasi dengan nama **SmsListenerApp**. Tetap gunakan **EmptyActivity**.
2. Pada **activity\_main.xml**, ubah komponen **TextView** menjadi :

```
1. <?xml version="1.0" encoding="utf-8"?>
2. <RelativeLayout xmlns:android="http://schemas.android.com/apk/res/andro
   id"
3.     xmlns:tools="http://schemas.android.com/tools"
4.     android:layout_width="match_parent"
5.     android:layout_height="match_parent"
6.     android:paddingBottom="@dimen/activity_vertical_margin"
7.     android:paddingLeft="@dimen/activity_horizontal_margin"
8.     android:paddingRight="@dimen/activity_horizontal_margin"
9.     android:paddingTop="@dimen/activity_vertical_margin"
10.    tools:context="com.dicoding.mybroadcastreceiver.MainActivity">
11.    <TextView
12.        android:layout_width="match_parent"
13.        android:layout_height="wrap_content"
14.        android:text="Ini Aplikasi Sms Receiver"/>
15. </RelativeLayout>
```

3. Kemudian buatlah sebuah **Activity** baru bernama **SmsReceiverActivity**. Pada **activity\_sms\_receiver.xml** sempurnakan kode yang ada menjadi seperti berikut :

```

1. <LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
2.     xmlns:tools="http://schemas.android.com/tools"
3.     android:layout_width="320dp"
4.     android:layout_height="match_parent"
5.     android:paddingBottom="@dimen/activity_vertical_margin"
6.     android:paddingLeft="@dimen/activity_horizontal_margin"
7.     android:paddingRight="@dimen/activity_horizontal_margin"
8.     android:paddingTop="@dimen/activity_vertical_margin"
9.     android:orientation="vertical"
10.    tools:context="com.dicoding.mybroadcastreceiver.SmsReceiverActivity">
11.    <TextView
12.        android:id="@+id/tv_no"
13.        android:layout_width="match_parent"
14.        android:layout_height="wrap_content"
15.        android:text="Coming From"
16.        android:layout_marginBottom="@dimen/activity_vertical_margin"/>
17.    <TextView
18.        android:id="@+id/tv_message"
19.        android:layout_width="match_parent"
20.        android:layout_height="wrap_content"
21.        android:text="Woy, main kesini jadi ngga? sekarang yak."
22.        android:textSize="18sp"
23.        android:layout_marginBottom="@dimen/activity_vertical_margin"/>
24.    <Button
25.        android:id="@+id/btn_close"
26.        android:layout_width="wrap_content"
27.        android:layout_height="wrap_content"
28.        android:padding="8dp"
29.        android:layout_gravity="right"
30.        android:text="Tutup"/>
31. </LinearLayout>
  
```

**Catatan :** Jika anda menambahkan activity secara manual maka perlu di registrasikan terlebih dahulu di dalam manifest. Tambahkan baris kode berikut.

```

32. <activity
33.     android:name=".SmsReceiverActivity"
34.     android:theme="@style/SmsReceiverTheme" />
  
```

4. Lalu buka berkas **styles.xml** pada direktori resource **res → values** dan tambahkan style baru seperti ini :

```

1. <style name="SmsReceiverTheme" parent="Theme.AppCompat.Light.Dialog"></style>
  
```

5. Sehingga kode dari `styles.xml` secara keseluruhan akan menjadi seperti ini :

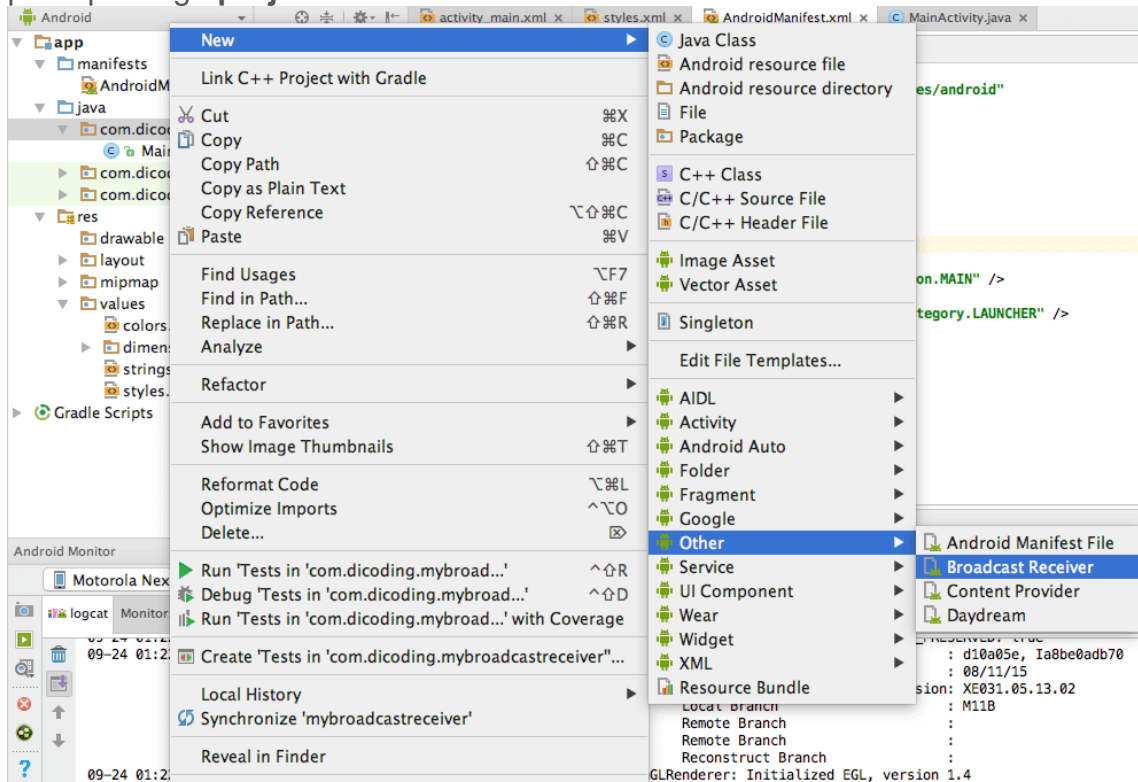
```
1. <resources>
2.     <style name="AppTheme" parent="Theme.AppCompat.Light.DarkActionBar"
3.         >
4.         <item name="colorPrimary">@color/colorPrimary</item>
5.         <item name="colorPrimaryDark">@color/colorPrimaryDark</item>
6.         <item name="colorAccent">@color/colorAccent</item>
7.     </style>
8.     <style name="SmsReceiverTheme" parent="Theme.AppCompat.Light.Dialog"
9.         "></style>
10. </resources>
```

Ide nya adalah kita akan menampilkan sebuah activity yang seolah-olah menjadi sebuah dialog.

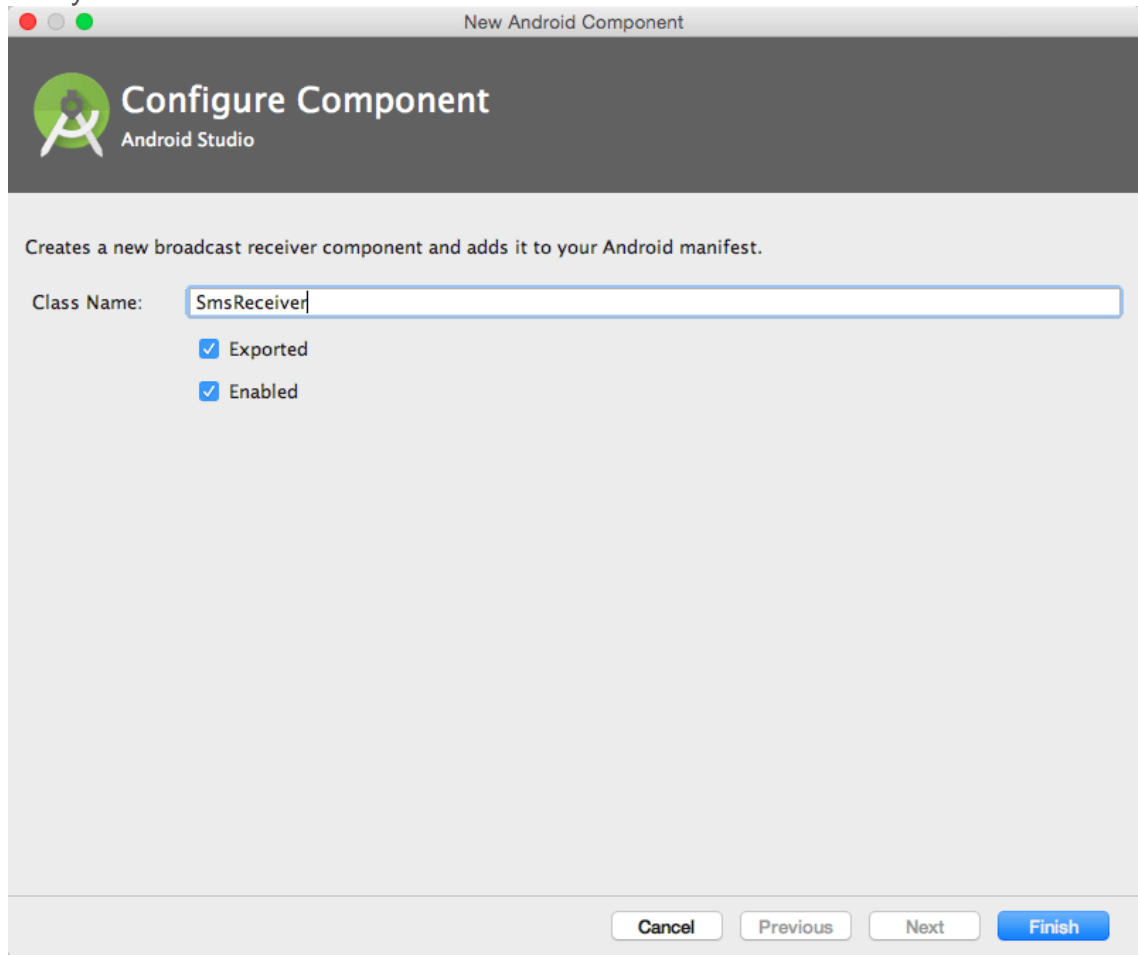
6. Langkah selanjutnya adalah dengan menambahkan kode di `SmsReceiverActivity` menjadi :

```
1. public class SmsReceiverActivity extends AppCompatActivity implements V
2.     iew.OnClickListener{
3.     private TextView tvSmsFrom;
4.     private TextView tvSmsMessage;
5.     private Button btnClose;
6.     public static final String EXTRA_SMS_NO = "extra_sms_no";
7.     public static final String EXTRA_SMS_MESSAGE = "extra_sms_message";
8.     @Override
9.     protected void onCreate(Bundle savedInstanceState) {
10.         super.onCreate(savedInstanceState);
11.         setContentView(R.layout.activity_sms_receiver);
12.         tvSmsFrom = (TextView)findViewById(R.id.tv_no);
13.         tvSmsMessage = (TextView)findViewById(R.id.tv_message);
14.         btnClose = (Button)findViewById(R.id.btn_close);
15.         btnClose.setOnClickListener(this);
16.         String senderNo = getIntent().getStringExtra(EXTRA_SMS_NO);
17.         String senderMessage = getIntent().getStringExtra(EXTRA_SMS_MESS
18.             AGE);
19.         tvSmsFrom.setText("from : "+senderNo);
20.         tvSmsMessage.setText(senderMessage);
21.     }
22.     @Override
23.     public void onClick(View v) {
24.         if (v.getId() == R.id.btn_close){
25.             finish();
26.         }
27.     }
28. }
```

7. Selanjutnya kita akan membuat sebuah kelas **Receiver** dengan cara klik kanan pada package **project utama** → **New** → **Other** → **BroadcastReceiver**.



8. Pada dialog yang tampil isikan **SmsReceiver** dan klik **Finish** untuk menyelesaikan.



9. Setelah tercipta, secara otomatis kelas **BroadcastReceiver** yang kita buat sudah teregistrasi di berkas **AndroidManifest.xml**.
10. Setelah selesai, silakan lengkapi kodenya menjadi sebagai berikut.

```
1. public class SmsReceiver extends BroadcastReceiver {
2.     final SmsManager sms = SmsManager.getDefault();
3.     public SmsReceiver() {
4.     }
5.     @Override
6.     public void onReceive(Context context, Intent intent) {
7.         // Retrieves a map of extended data from the intent.
8.         final Bundle bundle = intent.getExtras();
9.         try {
10.            if (bundle != null) {
11.                final Object[] pduObj = (Object[]) bundle.get("pdu");
12.                for (int i = 0; i < pduObj.length; i++) {
13.                    SmsMessage currentMessage = getSmsMessage(pduObj
14.                        bj[i], bundle);
15.                    String phoneNumber = currentMessage.getDisplayOrigin
16.                        atingAddress();
17.                    String senderNum = phoneNumber;
18.                    String message = currentMessage.getDisplayMessageBod
19.                        y();
20.                }
21.            }
22.        } catch (Exception e) {
23.            // Handle the exception
24.        }
25.    }
26. }
```

```
17.         Log.i("SmsReceiver", "senderNum: "+ senderNum + "; m
    message: " + message);
18.         Intent showSmsIntent = new Intent(context, SmsReceiv
    erActivity.class);
19.         showSmsIntent.setFlags(Intent.FLAG_ACTIVITY_NEW_TAS
    K);
20.         showSmsIntent.putExtra(SmsReceiverActivity.EXTRA_SMS
    _NO, phoneNumber);
21.         showSmsIntent.putExtra(SmsReceiverActivity.EXTRA_SMS
    _MESSAGE, message);
22.         context.startActivity(showSmsIntent);
23.     }
24. }
25. } catch (Exception e) {
26.     Log.e("SmsReceiver", "Exception smsReceiver" +e);
27. }
28. }
29. private SmsMessage getIncomingMessage(Object aObject, Bundle bundle)
    {
30.     SmsMessage currentSMS;
31.     if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.M) {
32.         String format = bundle.getString("format");
33.         currentSMS = SmsMessage.createFromPdu((byte[]) aObject, form
    at);
34.     } else {
35.         currentSMS = SmsMessage.createFromPdu((byte[]) aObject);
36.     }
37.     return currentSMS;
38. }
39. }
```

11. Selanjutnya pada berkas **AndroidManifest.xml** tambahkan permission diatas tag seperti ini :

```
1. <uses-permission android:name="android.permission.RECEIVE_SMS"/>
2. <uses-permission android:name="android.permission.READ_SMS" />
```

12. Dan tambahkan baris intent-filter di dalam tag yang kita punya seperti dibawah ini :

```
1. <intent-filter>
2.     <action android:name="android.provider.Telephony.SMS_RECEIVED" />
3. </intent-filter>
```

13. Sehingga berkas AndroidManifest.xml akan menjadi seperti ini :

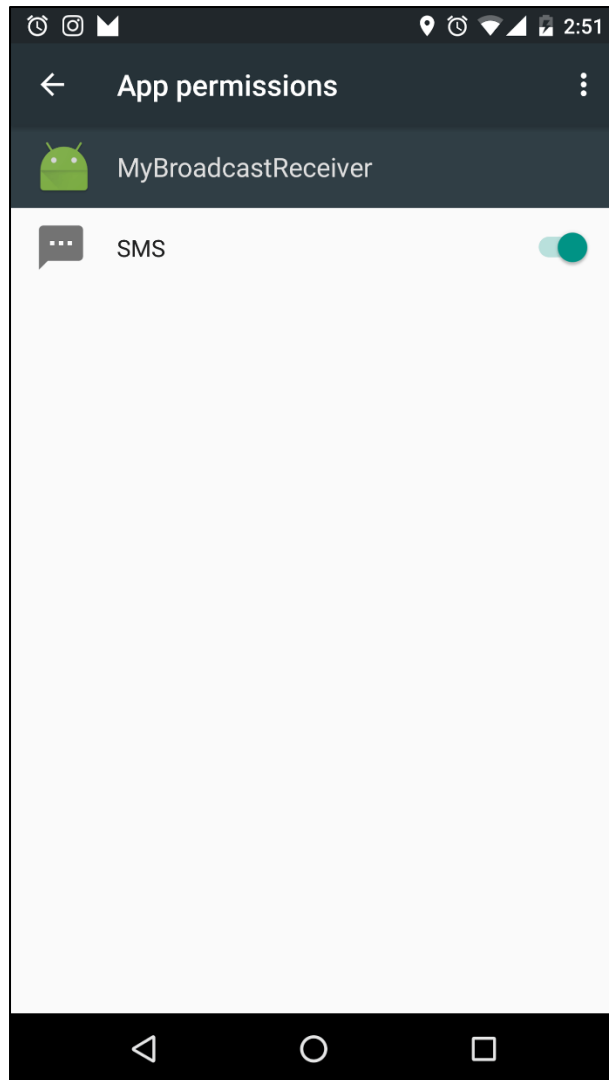
```
1. <?xml version="1.0" encoding="utf-8"?>
2. <manifest xmlns:android="http://schemas.android.com/apk/res/android"
3.     package="com.dicoding.mybroadcastreceiver">
4.     <uses-permission android:name="android.permission.RECEIVE_SMS" />
5.     <uses-permission android:name="android.permission.READ_SMS" />
6.     <application
7.         android:allowBackup="true"
8.         android:icon="@mipmap/ic_launcher"
9.         android:label="@string/app_name"
10.        android:supportRtl="true"
11.        android:theme="@style/AppTheme">
12.        <activity android:name=".MainActivity">
13.            <intent-filter>
14.                <action android:name="android.intent.action.MAIN" />
15.                <category android:name="android.intent.category.LAUNCHER"
16.            />
17.            </intent-filter>
18.        </activity>
19.        <activity android:name=".SmsReceiverActivity"
20.            android:theme="@style/SmsReceiverTheme"/>
21.        <receiver
22.            android:name=".SmsReceiver"
23.            android:enabled="true"
24.            android:exported="true">
25.            <intent-filter>
26.                <action android:name="android.provider.Telephony.SMS_RECEIVED" />
27.            </intent-filter>
28.        </receiver>
29.    </application>
30.</manifest>
```

14. Selesai sudah bagian pertama dalam memahami broadcast receiver.

15. Silakan jalankan aplikasi Anda. Cobalah mengirim SMS ke device Android Anda dan perhatikan hasilnya. Setiap SMS yang masuk akan ditampilkan langsung di layar. Keren bukan?!

## Permission Marshmallow

Bila Anda memiliki peranti Android dengan versi Marshmallow keatas, maka Anda perlu memberikan permission pada aplikasi untuk mengirimkan SMS. Caranya, buka **settings** → **Application** → **MyBroadcastReceiver** → **Permissions** → **SMS** (diaktifkan)





# Lembar Kerja Praktikum

NPM:	Nama Asisten:
Nama:	Nilai:
Kelas:	

Form ini digunakan untuk menilai kerja mahasiswa selama proses praktikum berlangsung. Selama praktikum mahasiswa diminta untuk membuat Aplikasi Broad Recevers. Semua tombol harus dapat berfungsi sesuai dengan materi yang telah disampaikan.

