

# Android Application Components

---

1. Activity

**1. Broadcast Receiver**

1. Content Provider

1. Service

---

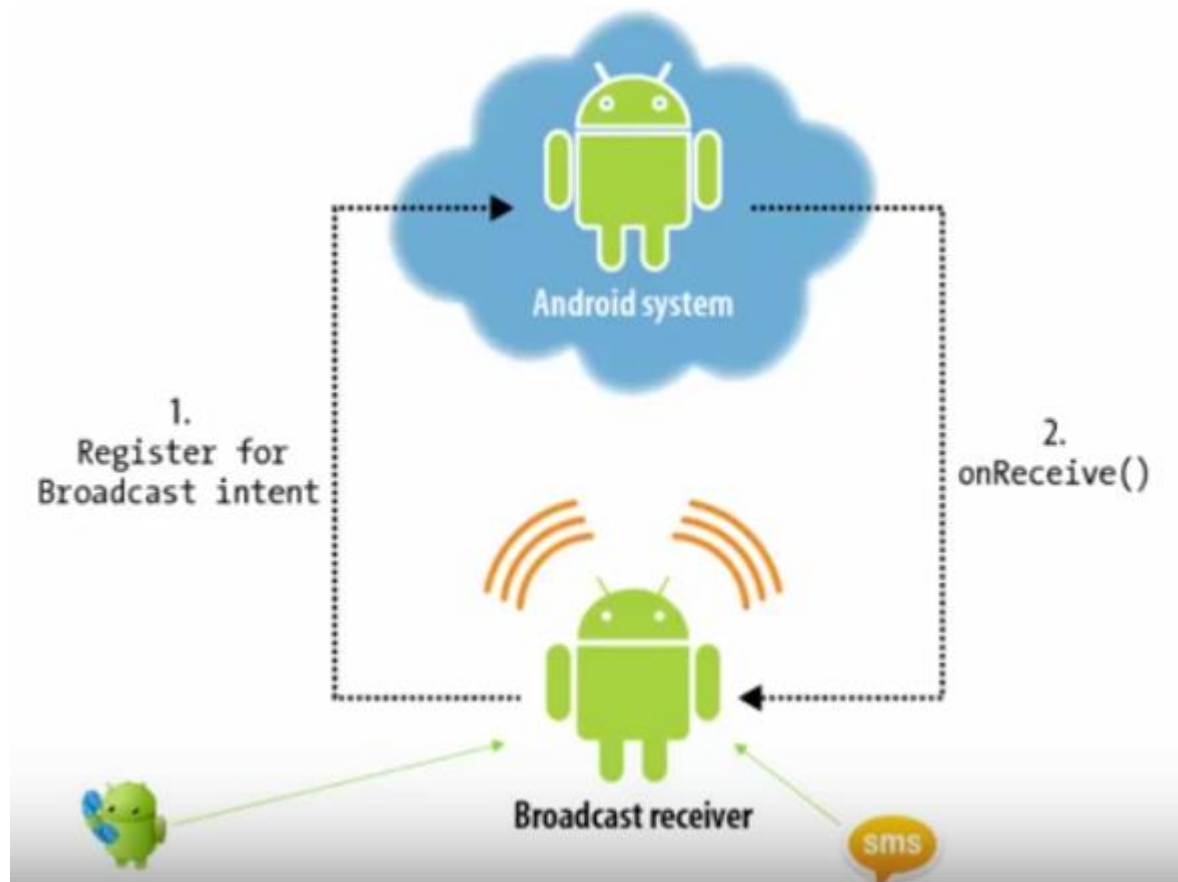
# Broadcast Receiver

---

- Lắng nghe các thông điệp từ các ứng dụng bên ngoài hoặc từ các intent
- Cách thức hoạt động:
  - Bắt các thông điệp

---

- Hoạt động



- 
- Hai bước thiết lập
    - Khởi tạo Broadcast Receiver
    - Đăng ký Broadcast Receiver
  - Một Broadcast Receiver phải được implement từ class Broadcast Receiver và phải được Override phương thức onReceive()

```
public class MyReceiver extends BroadcastReceiver {  
  
    @Override  
    public void onReceive(Context context, Intent intent) {  
        Toast.makeText(context, "Intent Detected.", Toast.LENGTH_LONG).show();  
    }  
  
}
```

---

- Tạo Broadcast Receiver

Sự kiện	Mô tả
<code>android.intent.action.BATTERY_CHANGED</code>	Bao gồm tất cả các thông tin về pin
<code>android.intent.action.BATTERY_LOW</code>	Xảy ra khi pin yếu
<code>android.intent.action.BATTERY_OKAY</code>	Xảy ra khi pin được sạc
<code>android.intent.action.BOOT_COMPLETED</code>	Khi thiết bị khởi động thành công
<code>android.intent.action.BUG_REPORT</code>	Khi ứng dụng bị lỗi
<code>android.intent.action.CALL</code>	Gọi đến một người nào đó bằng data
<code>android.intent.action.CALL_BUTTON</code>	Phát sinh khi nhấn nút gọi
<code>android.intent.action.DATE_CHANGED</code>	Phát sinh khi ngày được thay đổi
<code>android.intent.action.REBOOT</code>	Phát sinh khi khởi động lại máy

---

# Broadcast Receiver

---

- a component that responds to system wide broadcast announcements
    - Incoming SMS, MMS, Email, phone call
    - Battery low, Screen has turned off, or a picture was captured.
  - think of a Broadcast receiver as a “gateway” to the other components.
  - intended to very little work
    - Depending on the incoming event to the receiver, it could start a Service or Activity.
-

# Broadcast Receiver

---

## To add a new Broadcast Receiver

- open the AndroidManifest.xml file
  - click on the Application tab at the bottom of the window
  - under the Application Nodes section, click Add ...
  - select Receiver
  - in the Attributes for Receiver section, click on the [Name\\*](#) link
  - Enter the name of your Broadcast Receiver
    - e.g. "MyBroadcastReceiver"
  - Finish
-

# Broadcast Receiver - Manifest File

---

If you view AndroidManifest.xml in the XML editor, you should see something like this

```
<receiver  
    android:name="MyBroadcastReceiver"></receiver>
```

---



# Broadcast Receiver

---

Your new Broadcast Receiver file should look something like this

```
public class MyBroadcastReceiver extends BroadcastReceiver {  
  
    @Override  
    public void onReceive(Context context, Intent intent) {  
  
    }  
}
```

---

# Broadcast Receiver

---

```
public class M extends BroadcastReceiver {  
    @Override  
    public void onReceive(Context context, Intent intent) {  
    }  
}
```

Event listener for when a broadcast has been received.

# Broadcast Receiver

---

```
public class M extends BroadcastReceiver {  
    @Override  
    public void onReceive(Context context, Intent intent) {  
  
    }  
}
```

How do you know which broadcasts to listen for?

# Registering BroadcastReceiver

---

Hai cách để đăng ký BroadcastReceiver lắng nghe broadcast

- 1 . Thêm một <intent-filter> vào file Manifest
  2. Bằng chương trình
-

# Registering BroadcastReceiver

---

1. To add an <intent-filter> to the Manifest and listen for a Broadcast ...
    1. Open AndroidManifest.xml and click on the **Application** Tab at the bottom of the Window
    2. Select your **BroadcastReceiver**
    3. Click **Add ...**
    4. Make sure "Create new element in the selected element ... " is selected, then choose **Intent Filter**
    5. Click **Add ...**
    6. Choose **Action**
    7. Select the appropriate permission in the **Name** drop-down list
      - a. e.g. android.intent.action.TIME\_SET
    8. Save AndroidManifest.xml
-

# Registering BroadcastReceiver

---

**Alternatively**, if you view AndroidManifest.xml in the XML editor, you should see something like this

```
<receiver android:name="MyBroadcastReceiver">
```

```
</receiver>
```

---

# Registering BroadcastReceiver

---

Phải thông báo cho hệ thống biết rằng ta muốn nghe broadcast(s) nào. Thêm thẻ <intent-filter>

```
<receiver android:name="MyBroadcastReceiver">
```

```
</receiver>
```

---

# Registering BroadcastReceiver

---

Phải thông báo cho hệ thống biết rằng ta muốn nghe broadcast(s) nào. Thêm thẻ `<intent-filter>`

Place the cursor here  
and press  
Ctrl + Space

```
name="MyBroadcastReceiver">
```

```
</receiver>
```

---



# Registering BroadcastReceiver

---

Phải thông báo cho hệ thống biết rằng ta muốn nghe broadcast(s) nào. Thêm thẻ `<intent-filter>`

Select `<intent-filter>`

```
name="MyBroadcastReceiver">
```

```
</receiver>
```

---

# Registering BroadcastReceiver

---

Phải thông báo cho hệ thống biết rằng ta muốn nghe broadcast(s) nào. Thêm thẻ `<intent-filter>`

```
<receiver android:name="MyBroadcastReceiver">  
  <intent-filter>  
  
  </intent-filter>  
</receiver>
```

---

# Registering BroadcastReceiver

---

Phải thông báo cho hệ thống biết rằng ta muốn nghe broadcast(s) nào. Thêm thẻ `<intent-filter>`

an `<intent-filter>`  
describes what this  
receiver should listen for

```
android:name="MyBroadcastReceiver">
```

```
</intent-filter>
```

```
</receiver>
```

---

# Registering BroadcastReceiver

---

Phải thông báo cho hệ thống biết rằng ta muốn nghe broadcast(s) nào. Thêm thẻ <intent-filter>

Place the cursor here  
and press  
Ctrl + Space

```
me="MyBroadcastReceiver">
```

```
</intent-filter>
```

```
</receiver>
```

---

# Registering BroadcastReceiver

---

Phải thông báo cho hệ thống biết rằng ta muốn nghe broadcast(s) nào. Thêm thẻ `<intent-filter>`



Select `<action/>`

`android:name="MyBroadcastReceiver">`

`</intent-filter>`

`</receiver>`


---

# Registering BroadcastReceiver

---

Phải thông báo cho hệ thống biết rằng ta muốn nghe broadcast(s) nào. Thêm thẻ <intent-filter>

```
<receiver android:name="BroadcastReceiver">  
  <intent-filter>  
    <action android:name="android.intent.action.MAIN" />  
  </intent-filter>  
</receiver>
```



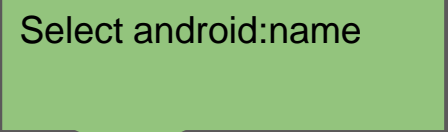
Press Ctrl + Space  
again

# Registering BroadcastReceiver

---

Phải thông báo cho hệ thống biết rằng ta muốn nghe broadcast(s) nào. Thêm thẻ <intent-filter>

```
<receiver android:name="BroadcastReceiver">  
  <intent-filter>  
    <action android:name="android.intent.action.MAIN" />  
  </intent-filter>  
</receiver>
```

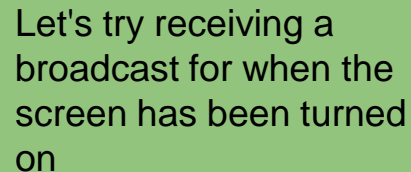


# Registering BroadcastReceiver

---

Phải thông báo cho hệ thống biết rằng ta muốn nghe broadcast(s) nào. Thêm thẻ <intent-filter>

```
<receiver android:name="...">  
  <intent-filter>  
    <action android:name="..." />  
  </intent-filter>  
</receiver>
```



Let's try receiving a broadcast for when the screen has been turned on




# Registering BroadcastReceiver

---

Phải thông báo cho hệ thống biết rằng ta muốn nghe broadcast(s) nào. Thêm thẻ `<intent-filter>`

```
<receiver android:name="I  
  <intent-filter>  
    <action android:name=" "/>  
  </intent-filter>  
</receiver>
```




Press Ctrl + Space  
again

# Registering BroadcastReceiver

---

Phải thông báo cho hệ thống biết rằng ta muốn nghe broadcast(s) nào. Thêm thẻ `<intent-filter>`

```
<receiver android:name="Select android.intent.action.TIME_SET"
  <intent-filter>
    <action android:name="" />
  </intent-filter>
</receiver>
```



# Registering BroadcastReceiver

---

Phải thông báo cho hệ thống biết rằng ta muốn nghe broadcast(s) nào. Thêm thẻ <intent-filter>

```
<receiver android:name="MyBroadcastReceiver">  
  <intent-filter>  
    <action android:name="android.intent.action.TIME_SET"/>  
  </intent-filter>  
</receiver>
```

---

# Registering BroadcastReceiver

---

Phải thông báo cho hệ thống biết rằng ta muốn nghe broadcast(s) nào. Thêm thẻ `<intent-filter>`

```
<receiver android:name="MyBroadcastReceiver">
  <intent-filter>
    <action android:name="android.intent.action.TIME_SET"/>
  </intent-filter>
</receiver>
```

---

# Registering BroadcastReceiver

---

2. Đăng ký BroadcastReceiver bằng chương trình ...

---

# Registering BroadcastReceiver

---

```
public class MainActivity extends Activity {
```

```
    MyBroadcastReceiver receiver;
```

```
    @Override public void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
        setContentView(R.layout.main);
```

```
        receiver = new MyBroadcastReceiver();
```

```
        IntentFilter filter = new IntentFilter(Intent.ACTION_SCREEN_ON);
```

```
        registerReceiver(receiver, filter);
```

```
    }
```

```
}
```

---

# Registering BroadcastReceiver

---

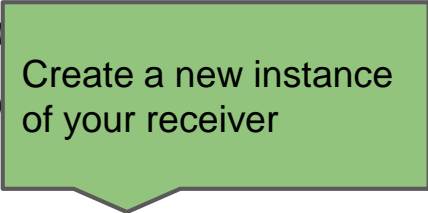
```
public class MainActivity extends Activity {
```

```
    MyBroadcastReceiver receiver;
```

```
    @Override public void onCreate(Bundle savedInstanceState) {
```

```
        super.onCreate(savedInstanceState);
```

```
        setContentView(R.layout.activity_main);
```



Create a new instance  
of your receiver

```
        receiver = new MyBroadcastReceiver();
```

```
        IntentFilter filter = new IntentFilter(Intent.ACTION_SCREEN_ON);
```

```
        registerReceiver(receiver, filter);
```

```
    }
```

```
}
```

---

# Registering BroadcastReceiver

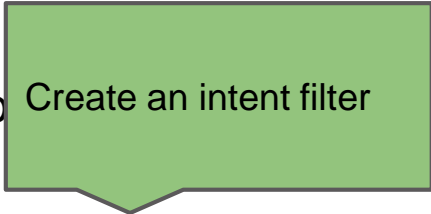
---

```
public class MainActivity extends Activity {
```

```
    MyBroadcastReceiver receiver;
```

```
    @Override public void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
        setContentView(R.layout.main);
```

```
        receiver = new MyBro
```



Create an intent filter

```
        IntentFilter filter = new IntentFilter(Intent.ACTION_SCREEN_ON);
```

```
        registerReceiver(receiver, filter);
```

```
    }
```

```
}
```

---



# Registering BroadcastReceiver

---

```
public class MainActivity extends Activity {
```

```
    MyBroadcastReceiver receiver;
```

```
    @Override public void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
        setContentView(R.layout.main);
```

```
        receiver = new MyBroadcastReceiver();
```

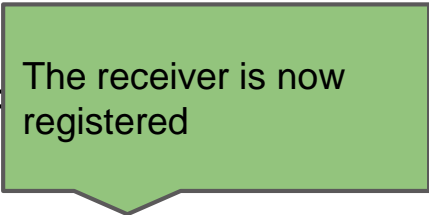
```
        IntentFilter intentFilter = new IntentFilter(Intent.ACTION_SCREEN_ON);
```

```
        registerReceiver(receiver, filter);
```

```
    }
```

```
}
```

---



The receiver is now registered

# BroadcastReceiver

Now let's look at our  
BroadcastReceiver

```
public class MyBroadcastReceiver extends BroadcastReceiver {  
    @Override  
    public void onReceive(Context context, Intent intent) {  
  
        Intent myIntent = new Intent(context, ActivityToBeLaunched.class);  
  
        Toast.makeText(  
            context,  
            "Broadcast Received!",  
            Toast.LENGTH_LONG).show();  
  
        myIntent.addFlags(Intent.FLAG_ACTIVITY_NEW_TASK);  
        context.startActivity(myIntent);  
    }  
}
```

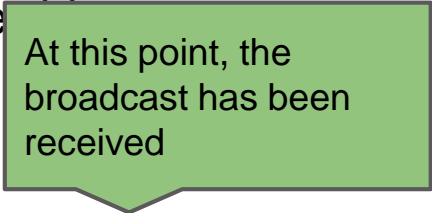
# BroadcastReceiver - onReceive()

---

```
public class MyBroadcastReceiver extends BroadcastReceiver {
```

```
    @Override
```

```
    public
```



At this point, the broadcast has been received

```
        onReceive(context context, Intent intent) {
```

```
            Intent myIntent = new Intent(context,
                ActivityToBeLaunched.class);
```

```
            Toast.makeText(
```

```
                context,
```

```
                "Broadcast Received!",
```

```
                Toast.LENGTH_LONG).show();
```

```
            myIntent.addFlags(Intent.FLAG_ACTIVITY_NEW_TASK);
```

```
            context.startActivity(myIntent);
```

```
        }
```

---

# BroadcastReceiver - onReceive()

---

```
public class MyBroadcastReceiver extends BroadcastReceiver {
```

```
@Override
```

```
public
```

Let's say we want to launch a new Activity when we receive the broadcast

```
onReceive(Context context, Intent intent) {
```

```
    Intent myIntent = new Intent(context,
    ActivityToBeLaunched.class);
```

```
    Toast.makeText(
```

```
        context,
```

```
        "Broadcast Received!",
```

```
        Toast.LENGTH_LONG).show();
```

```
    myIntent.addFlags(Intent.FLAG_ACTIVITY_NEW_TASK);
```

```
    context.startActivity(myIntent);
```

```
}
```

---

# BroadcastReceiver - onReceive()

---

```
public class MyBroadcastReceiver extends BroadcastReceiver {
```

```
    @Override
```

```
    public void onReceive(Context context, Intent intent) {
```

And we want to show a  
Toast

```
        myIntent = Intent(context, ActivityToBeLaunched.class);
```

```
        Toast.makeText(
```

```
            context,
```

```
            "Broadcast Received!",
```

```
            Toast.LENGTH_LONG).show();
```

```
        myIntent.addFlags(Intent.FLAG_ACTIVITY_NEW_TASK);
```

```
        context.startActivity(myIntent);
```

```
    }
```

```
}
```

---



# BroadcastReceiver - onReceive()

---

```
public class MyBroadcastReceiver extends BroadcastReceiver {  
    @Override  
    public void onReceive(Context context, Intent intent) {  
  
        Intent myIntent = new Intent(context, ActivityToBeLaunched.class);  
  
        Toast.makeText(  
            context,  
            "Broadcast",  
            Toast.LENGTH_SHORT).show();  
  
        myIntent.addFlags(Intent.FLAG_ACTIVITY_NEW_TASK);  
        context.startActivity(myIntent);  
    }  
}
```

Calling startActivity() from outside of an Activity context requires the FLAG\_ACTIVITY\_NEW\_TASK flag

# Unregistering BroadcastReceiver

---

Sometimes after you start listening for a Broadcast, you may want to STOP listening for it at some point

To UNregister a BroadcastReceiver programmatically ...

---



# Unregistering BroadcastReceiver

---

```
public class MainActivity extends Activity {  
  
    @Override  
    public void onPause() {  
  
        unregisterReceiver(receiver);  
        super.onPause();  
  
    }  
}
```

---

# Unregistering BroadcastReceiver

---

```
public class MainActivity extends Activity {
```

@ You will want to unregister before  
calling a super.onSomething()  
method, otherwise you may get  
Force Close

```
    unregisterReceiver(receiver);
```

```
    super.onPause();
```

```
}
```

```
}
```

---

# Unregistering BroadcastReceiver

---

**If you register a BroadcastReceiver programmatically, you should unregister it before your Activity is destroyed!**

- otherwise you will have a "leaked" receiver
- Please call unregisterReceiver in
  - onPause()
  - onStop()
  - or onDestroy()

**This is important**

---

# Code example

---

- Bắt cuộc gọi

- Tạo Project
- Tạo lớp IncomingCall (Java Class)
- Override phương thức onReceive
- Lấy trạng thái hiện tại

```
String state = intent.getStringExtra(TelephonyManager.EXTRA_STATE);
```

- Kiểm tra xem có phải đang có cuộc gọi đến?

```
if (TelephonyManager.EXTRA_STATE_RINGING.equals(state))
```

- Nếu đúng → thông báo

```
Toast.makeText(context, "a calling come  
from"+intent.getStringExtra(TelephonyManager.EXTRA_INCOMING_NUMBER),  
Toast.LENGTH_LONG).show();
```

Lấy số gọi đến



# Code example

- Khai báo Receiver trong file Manifest

```
<?xml version="1.0" encoding="utf-8" ?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.example.kenzo.broadcastreceiverdemo">
    <uses-permission android:name="android.permission.READ_PHONE_STATE"/>
    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="Broadcast receiver demo"
        android:supportsRtl="true"
        android:theme="@style/AppTheme">
        <activity android:name=".MainActivity">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
        <receiver android:name=".IncomingCall">
            <intent-filter>
                <action android:name="android.intent.action.PHONE_STATE"/>
            </intent-filter>
        </receiver>
    </application>
</manifest>
```

Xin quyền đọc thông tin trạng thái điện thoại

Khai báo thông tin Receiver

Android:name → class khai báo Broadcast Receiver

<intent-filter action> → bắt các thay đổi trạng thái của điện thoại

# Code example

---

- Bật command window
  - telnet localhost 5554
  - Thực hiện cuộc gọi với lệnh
    - Gms call <số đt>
-

# Code example 2

---

- Kiểm tra mức pin dưới 10% thì cảnh báo
- (Đăng ký Broadcast Receiver bằng chương trình)
- Tạo đối tượng Broadcast Receiver
- Override phương thức onReceive

```
public BroadcastReceiver broadcastReceiver = new BroadcastReceiver() {  
    @Override  
    public void onReceive(Context context, Intent intent) {  
  
    }  
}
```

---

# Code example 2

---

- Kiểm tra nếu pin < 10% thì thông báo

```
int battery = intent.getIntExtra("level",0);
    if (battery<=10)
    {
        Toast.makeText(context,"Low battery. Battery is "+battery+" plug your
chager",Toast.LENGTH_LONG).show();
    }
```

- Đăng ký Receiver với phương thức registerReceiver()

```
registerReceiver(broadcastReceiver, new
IntentFilter(Intent.ACTION_BATTERY_CHANGED));
```

---



# Code example 2

---

- Thiết lập giá trị pin
  - Power capacity <%pin còn lại>

# Nhận tin nhắn

---

- SmsManager: hỗ trợ gửi SMS
- Intent: chứa dữ liệu thông qua Bundle
  - Bundle bundle = intent.getExtras();

```
SmsMessage currentMessage = SmsMessage.createFromPdu((byte[]) pduObj[i]);  
String phoneNumber = currentMessage.getDisplayOriginatingAddress();
```

```
String senderNum = phoneNumber;  
String message = currentMessage.getDisplayMessageBody();
```

```
Log.i("SmsReceiver", "senderNum: " + senderNum + "; message: " + message);
```

---

# Code Examples

---

- [BroadcastReceiverExample](#)
  - [SimpleFragmentUsingXml](#)
  - [SimpleFragmentProgrammatically](#)
  - [FragmentActivityReferenceExample](#)
  - [FragmentTransitionExample](#)
  - [LayoutInflaterExample](#)
  - [PermissionsExample](#)
  - [ComplexFragmentsExample](#)
  - [DynamicUiExample](#)
-

# References

---

- [The Busy Coder's Guide to Android Development - Mark Murphy](#)
  - [Android Developers](#)
  - [Code Examples](#)
-