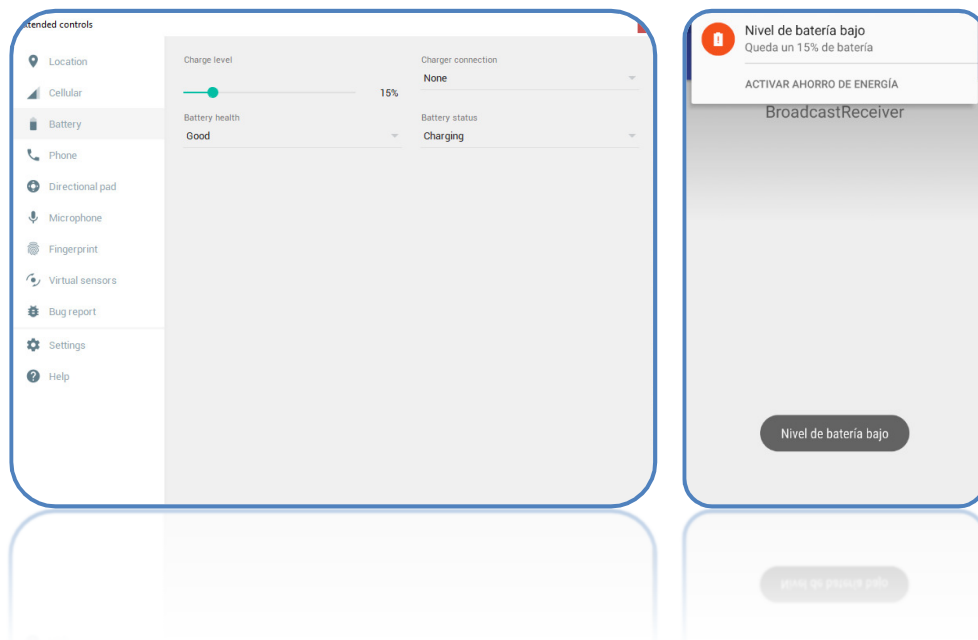


## UNIDAD 4: Broadcast Receiver

Vamos a ver el funcionamiento de un *Broadcast Receiver* creando una aplicación que muestre una alerta cuando el nivel de batería sea bajo y cuando se active o desactive el modo avión.

Aunque lo habitual es crear un *Broadcast Receiver* para cada tipo de evento, para esta práctica vamos a crear uno que responda ante los dos eventos.



## CREACIÓN DE LA INTERFAZ

Esta aplicación no requiere ningún elemento con el que el usuario interactúe. De hecho, los *Broadcast Receiver* no tienen interfaz.

Aquí lo vamos a agregar a una actividad vacía, pero se podría utilizar este mismo *Broadcast Receiver* u otro similar en cualquiera de nuestras aplicaciones.

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
    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"
    tools:context="com.example.oscar.exbroadcastreceiver.MainActivity"
    android:layout_width="match_parent"
    android:layout_height="match_parent">

    <TextView
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="BroadcastReceiver"
        android:textSize="20sp"
        android:gravity="center"
        android:layout_margin="24dp" />

</LinearLayout>
```

## CLASE RECEPTOR

El primer paso es crear una clase que extienda a *BroadcastReceiver* y sobrecargar el método *onReceive()*, que recibe como parámetros el contexto y un *intent* a partir del cual podremos obtener toda la información del evento.

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

A partir del *intent* podremos saber el evento que se ha producido utilizando *getAction()*. En esta aplicación nos interesan dos tipos de eventos:

- Nivel de batería bajo: *android.intent.action.BATTERY\_LOW*
- Cambio de estado en el modo avión: *android.intent.action.AIRPLANE\_MODE*

```
switch (intent.getAction()) {  
    case "android.intent.action.BATTERY_LOW":  
        ...  
        break;  
    case "android.intent.action.AIRPLANE_MODE":  
        ...  
        break;
```

En el caso del nivel de batería bajo, simplemente generaremos una alerta en modo de *Toast*.

```
int duracion = Toast.LENGTH_SHORT;  
String mensa = "Nivel de batería bajo";  
Toast.makeText(context, mensa, duracion).show();
```

En el caso de cambio de estado del modo avión, comprobaremos si el nuevo estado es activado o desactivado utilizando un extra de tipo booleano del *intent* denominado "*state*", un valor *true* indica que está activado, y *false* que está desactivado. Una vez conocido el estado, generamos la alerta adecuada para cada caso.

```
Bundle extras = intent.getExtras();  
if (extras != null) {  
    if (extras.getBoolean("state") == true) {  
        mensa = "Modo Avión activado";  
    } else {  
        mensa = "Modo Avión desactivado";  
    }  
}  
Toast.makeText(context, mensa, duracion).show();
```

La clase "Receptor" completa quedaría así:

```
public class Receptor extends BroadcastReceiver {
    @Override
    public void onReceive(Context context, Intent intent) {
        int duracion = Toast.LENGTH_SHORT;
        String mensa = "";
        switch (intent.getAction()) {
            case "android.intent.action.BATTERY_LOW":
                /* Este código permite obtener el nivel de batería en el momento en
                 * que cambia el estado
                 */
                IntentFilter filtro = new IntentFilter(Intent.ACTION_BATTERY_CHANGED);
                Intent estadoBateria = context.registerReceiver(null, filtro);
                int nivel = estadoBateria.getIntExtra(BatteryManager.EXTRA_LEVEL, -1);
                int escala = estadoBateria.getIntExtra(BatteryManager.EXTRA_SCALE, -1);
                int valor = (int)(nivel / (float)escala * 100);
                mensa = "Nivel de batería bajo (" + valor + "%)";
                /*

                mensa = "Nivel de batería bajo";
                Toast.makeText(context, mensa, duracion).show();
                break;

            case "android.intent.action.AIRPLANE_MODE":
                Bundle extras = intent.getExtras();
                if (extras != null) {
                    if (extras.getBoolean("state") == true) {
                        mensa = "Modo Avión activado";
                    } else {
                        mensa = "Modo Avión desactivado";
                    }
                }
                Toast.makeText(context, mensa, duracion).show();
                break;
        }
    }
}
```

## REGISTRAR LA CLASE RECEPTOR

Una vez creado la clase "Receptor" hemos de registrarla dentro de un elemento receiver de nuestro archivo manifiesto, y agregar los filtros de *intent* con los eventos a los que queremos responder.

```
<receiver android:name=".Receptor" android:exported="false">
  <intent-filter>
    <action android:name="android.intent.action.BATTERY_LOW"/>
    <action android:name="android.intent.action.AIRPLANE_MODE"/>
  </intent-filter>
</receiver>
```

Este sería nuestro archivo de manifiesto:

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

  <application
    android:allowBackup="true"
    android:icon="@mipmap/ic_launcher"
    android:label="@string/app_name"
    android:roundIcon="@mipmap/ic_launcher_round"
    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>

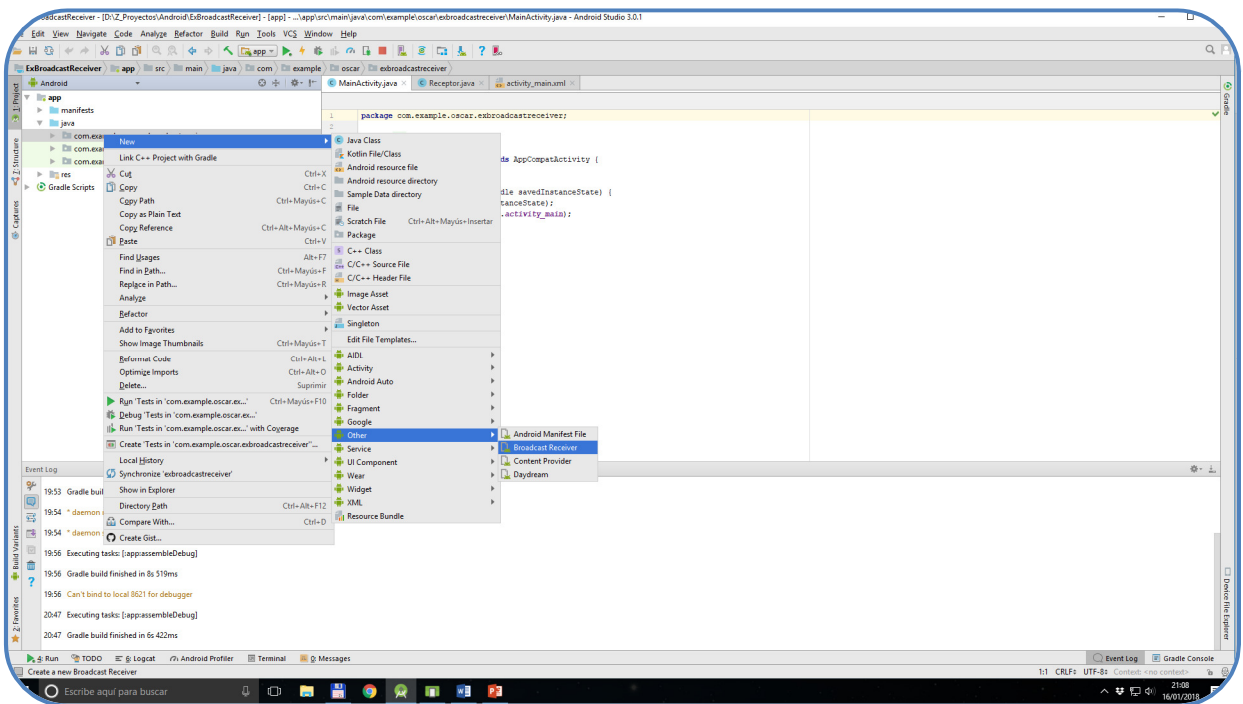
    <receiver android:name=".Receptor" android:exported="false">
      <intent-filter>
        <action android:name="android.intent.action.BATTERY_LOW"/>
        <action android:name="android.intent.action.AIRPLANE_MODE"/>
      </intent-filter>
    </receiver>

  </application>
</manifest>
```

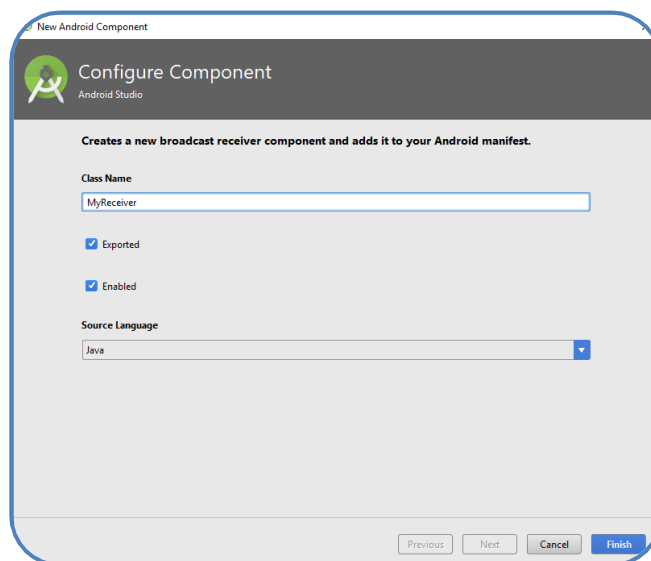
## BROADCAST RECEIVER EN ANDROID STUDIO

Android Studio incluye una herramienta para la creación de un *BroadcastReceiver*.

Podemos acceder a esta herramienta pulsando el botón derecho del ratón sobre la carpeta de nuestras clases java y seleccionando: *New, Other, BroadcastReceiver*:



A continuación, establecemos el nombre de la clase, sus propiedades y el lenguaje en que queremos implementarla, y pulsamos finalizar:



Automáticamente se creará la clase que extiende a *BroadcastReceiver* con el método *onCreate* que tendremos que implementar:

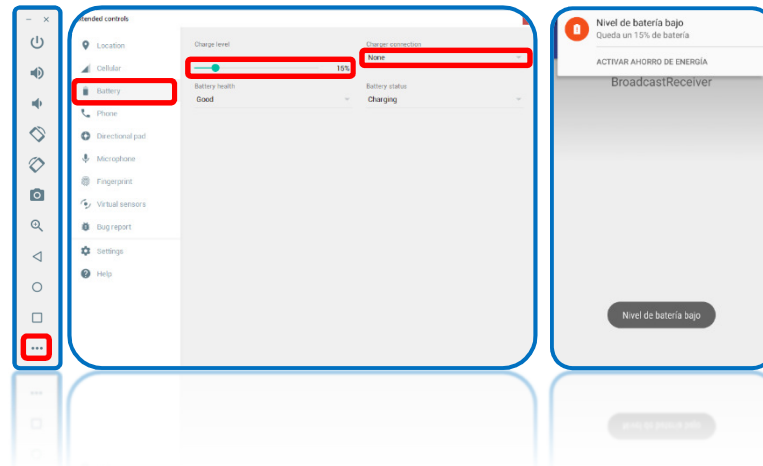
```
public class MyReceiver extends BroadcastReceiver {  
  
    @Override  
    public void onReceive(Context context, Intent intent) {  
        // TODO: This method is called when the BroadcastReceiver is receiving  
        // an Intent broadcast.  
        throw new UnsupportedOperationException("Not yet implemented");  
    }  
}
```

Y también se creará el receptor en nuestro archivo de manifiesto, aunque nos faltará añadir los eventos que queremos gestionar:

```
<receiver  
    android:name=".MyReceiver"  
    android:enabled="true"  
    android:exported="true">  
</receiver>
```

## PRUEBAS

Para probar que realmente nuestra aplicación responde ante un nivel de batería bajo, deberemos utilizar las herramientas de nuestro emulador.



En cuanto al modo avión, tendremos que activarlo y desactivarlo en los ajustes de nuestro dispositivo.

