

# **Broadcast receivers, content providers, services, async tasks**

**Mateusz Nowotyński   Marcin Moskal   Kamil Osuch**

**12.12.2017**

## Do czego to służy?

BroadcastReceiver pozwala nam na odbieranie powiadomień (Systemu bądź innej aplikacji) wewnątrz naszej aplikacji. Takim powiadomieniem może być na przykład informacja o nowej wiadomości SMS bądź rozładowanej baterii.

Żeby zbudować nasz własny BroadcastReceiver musimy wykonać dwie czynności:

- 1 Stworzyć podklasę klasy BroadcastReceiver
- 2 Wyszpecyfikować receiver w manifeście aplikacji

## Przykład klasy Broadcast Receiver'a

```
public class MyBroadcastReceiver extends BroadcastReceiver {
    private static final String TAG = "MyBroadcastReceiver";

    @Override
    public void onReceive(Context context, Intent intent) {
        StringBuilder sb = new StringBuilder();
        sb.append("Action: " + intent.getAction() + "\n");
        sb.append("URI: " + intent.toUri(Intent.URI_INTENT_SCHEME).toString() + "\n");
        String log = sb.toString();
        Log.d(TAG, log);
        Toast.makeText(context, log, Toast.LENGTH_LONG).show();
    }
}
```

## Dodanie do manifestu

```
<receiver android:name=".MyBroadcastReceiver" android:exported="true">
    <intent-filter>
        <action android:name="android.intent.action.BOOT_COMPLETED"/>
        <action android:name="android.intent.action.INPUT_METHOD_CHANGED" />
    </intent-filter>
</receiver>
```