

Intents - III

Responder a intents implícitos

Ya hemos visto que es posible realizar solicitudes de una determinada acción al sistema: el *intent implícito*. Pero también es posible que sea nuestra app la que pueda dar ese servicio y por tanto, sea elegible por **chooser**. Para ello, primero tenemos que registrar nuestra app como manejadora potencial de un determinado tipo de *intent*. Y todo eso se hace a través del fichero **AndroidManifest.xml** de nuestra app. Al añadir un intent-filter estaremos anunciando al sistema Android que nuestra app es capaz de responder a ciertas **acciones**.

```
<activity
    android:name=".Activity2"
    android:exported="true">

    <intent-filter>
        <action android:name="es.ua.eps.intent.action.HAZ_ALGO" />
        <category android:name="android.intent.category.DEFAULT" />
        <category android:name="android.intent.category.SELECTED_ALTERNATIVE" />
        <data android:mimeType = "vnd.myApp.cursor.item/*" />
    </intent-filter>

    <meta-data
        android:name="android.app.lib_name"
        android:value="" />
</activity>
```

Cada intent-filter consta de los siguientes elementos:

- **Action:** especifica el nombre de la acción o acciones que la actividad es capaz de gestionar. Es posible utilizar el nombre de una acción que ya existe, o inventarte uno. Obviamente, el nombre debería ser largo y descriptivo, marcando la ruta de paquetes.
- **Category:** especifica bajo qué circunstancias la app atenderá a la petición. Se pueden indicar varias. Cuando se lance un *intent implícito* para que lo atienda tu app, se le añadirán las categorías que deseemos, y solamente responderán las apps que contengan todas las categorías indicadas.
- **Data:** indica sobre qué tipos de datos actuará nuestra *actividad*. Es posible indicar varios. Es habitual usar una URI para identificar el dato de forma implícita, aunque podemos usar también un tipo MIME.

Categorías del intent

La forma de añadir una categoría a un intent es simple y se puede apreciar en este ejemplo. Recuerda que únicamente responderán al intent aquellas apps cuyas categorías contengan todas las categorías indicadas.

```
buttonToAc2.setOnClickListener{ it: View!
    fun onClick(v: View?) {
        val intent = Intent(applicationContext, Activity2::class.java)
        intent.putExtra(name: "Veces", finalVeces)
        intent.addCategory(Intent.CATEGORY_DEFAULT)
        startActivityForResult(intent, SUBACTIVITY_1)
    }
}
```

Algunos de los principales valores estándar para las categorías son:

- **ALTERNATIVE:** Nuestra app se presenta como alternativa, aunque quien vaya a hacer la llamada no conozca la acción que realiza. Útil cuando tenemos acciones propias que no son los estándares de Android y que otras apps no conocerán.
- **SELECTED_ALTERNATIVE:** Igual que la anterior, pero cuando se le presente al usuario la app a elegir, la nuestra aparecerá seleccionada por defecto.
- **BROWSABLE:** La acción está disponible para ser lanzada sobre un navegador web. Cuando un *intent* se lance desde un navegador, siempre incluirá esta categoría, luego solo encontrará apps **BROWSABLE**.
- **DEFAULT:** Esta acción siempre se añade por defecto al realizar un *intent*.
- **GADGET:** Nuestra *actividad* puede ser ejecutada empotrada dentro de otra *actividad*.
- **HOME:** Si no especificamos acción, esta actividad es una alternativa a la pantalla de inicio nativa.
- **LAUNCHER:** Nuestra app aparecerá en el lanzador de aplicación de Android. Esta categoría aparece por defecto en la *actividad principal* de nuestro proyecto.

Tipo de dato del intent

En los intent implícitos, el tipo de dato se especifica mediante una URI.

```
val viewIntent = Intent (Intent.ACTION_VIEW)
viewIntent.data = Uri.parse ( uriString: "http://www.google.com");
```

A partir de esa URI el sistema infiere el tipo MIME de los datos, aunque también podríamos indicárselo explícitamente:

```
val viewIntent = Intent (Intent.ACTION_VIEW)
viewIntent.setType("text/html")
```