

29. Internacionalización

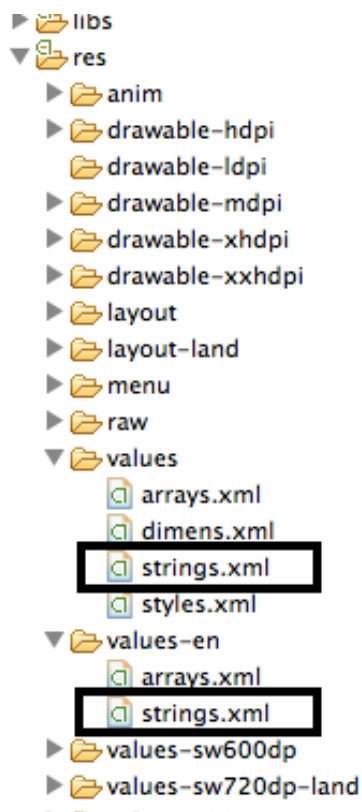
Hemos visto que Android permite adaptar la interfaz de las actividades a la posición del dispositivo (landscape o portrait) mediante ficheros de diseño alternativos. El mismo mecanismo se utiliza para adaptar la aplicación a otras características concretas del dispositivo como la versión del sistema operativo, el idioma, el tamaño de la pantalla, la densidad, etc. En esta unidad vamos a ver cómo crear un recurso alternativo para el idioma inglés en nuestro juego.

Igual que en la unidad 21 añadimos un fichero de diseño `activity_main.xml` alternativo a la carpeta `/res/layout-land/`, ahora añadiremos versiones en inglés de `strings.xml` y `arrays.xml` dentro de una nueva carpeta de recursos alternativa de nombre `/res/values-en`. Android se encargará de seleccionar el fichero de recursos mas adecuado dada la configuración del dispositivo:

```
/res/values-en/strings.xml
```

```
<?xml version="1.0" encoding="utf-8"?>
<resources>
    <string name="app_name">CCC29</string>
    <string name="action_settings">Settings</string>
    <string name="gameOverTitle">Game over</string>
    <string name="gameOverMessage">Do you want to play again?</string>
    <string name="initialImage">Imagen inicial del juego</string>
    <string name="aboutText">About</string>
    <string name="sendMessageText">Send message</string>
    <string name="aboutMessage">Cha cha cha is a solitaire game
        in which we have to get rid of the pieces
        jumping over them, vertically or
        horizontally, towards the adjacent position which
        must be empty. The aim is to leave
        the last piece in the center of the board, the position
        that is initially empty.</string>
    <string name="firstCategoryPrefTitle">Personal data</string>
    <string name="editTextPrefTitle">Player name</string>
    <string name="editTextPrefSummary">Click to type the player name</string>
    <string name="secondCategoryPrefTitle">Game properties</string>
    <string name="listPrefTitle">Type of figure</string>
    <string name="listPrefSummary">Click to pop up a list to choose from</string>
    <string name="checkBoxPrefTitle">Background music</string>
    <string name="checkBoxPrefSummary">Mark if you want to listen to music</string>
    <string name="listPrefDialogTitle">Choose the initial figure</string>
    <string name="preferencesText">Preferences</string>
</resources>
```

Como puedes ver, el nombre del archivo de recursos es el mismo (`strings.xml`) así como el de las cadenas en su interior (`app_name`, `aboutText`, ...). Lo que cambia es su valor y el nombre de la carpeta en que se encuentra el archivo (`values-en` en lugar de `values`):



De igual manera también proporcionamos una versión en inglés de `arrays.xml`:

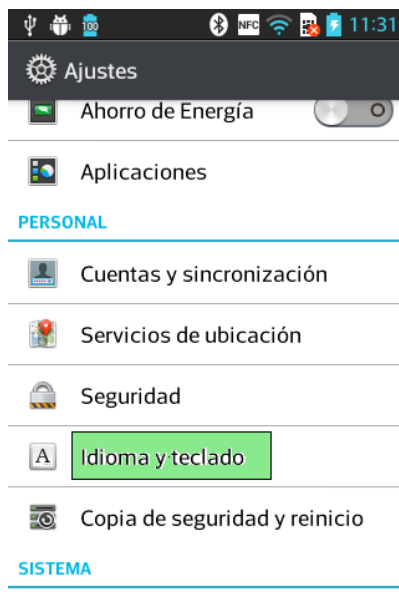
`/res/values-en/arrays.xml`

```
<?xml version="1.0" encoding="utf-8"?>
<resources>
    <string-array name="figures">
        <item>Complete</item>
        <item>Circle</item>
        <item>Ring</item>
    </string-array>
    <string-array name="figureCodes">
        <item>0011100001110011111111111110111111111111100111000011100</item>
        <item>0000000001110001111100110110011111000111000000000</item>
        <item>0000000001110001101100100010011011000111000000000</item>
    </string-array>
</resources>
```

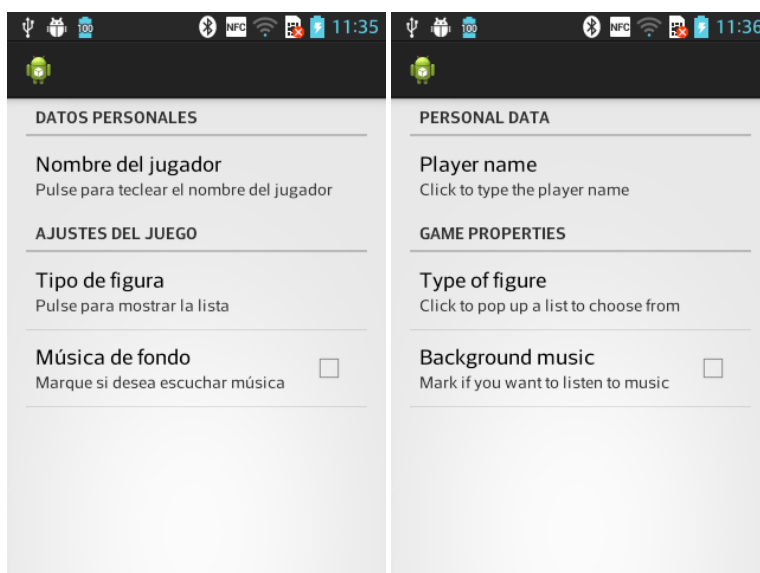
Para dar soporte al idioma francés tendríamos que añadir una carpeta de nombre `values-fr`, así como un nuevo archivo `strings.xml` con versiones en francés de los recursos correspondientes.

Podemos localizar aún más los recursos: por ejemplo, podríamos añadir una carpeta de nombre `values-en-rUS` para proporcionar una alternativa para el inglés de EEUU. Android resuelve la elección de recursos yendo de lo particular a lo general.

Puedes comprobar el efecto cambiando el idioma de tu dispositivo en `Ajustes -> Idioma y teclado`:



Lo que sigue es el menú de preferencias en castellano y en inglés:



Es muy conveniente dotar a la app de recursos por omisión, es decir, recursos en la carpeta `/res/values`, sin calificadores. Si no existen recursos por omisión, la app terminará bruscamente si Android necesita un recurso y no encuentra el que encaja con la configuración del dispositivo. Nuestro juego, si se fija el italiano como lenguaje del dispositivo, utilizará los recursos en castellano de la carpeta `/res/values`.

Debido al gran número de calificadores de configuración que existen en Android, puede que la configuración del dispositivo encaje con mas de un recurso alternativo. Es por ello que los calificadores tienen asignado un orden o precedencia. Así, por ejemplo, el calificador de lenguaje va antes que el de la orientación del dispositivo. Este orden se ha de respetar al crear recursos alternativos: `/res/values-en-land` en lugar de `/res/values-land-en`.

Finalmente, no olvides que los recursos deben situarse en subdirectorios de `res`. No es posible colocar recursos en la raíz del proyecto, por ejemplo. Los nombres de los subdirectorios están determinados por Android. Si creas uno nuevo de nombre `principal`, por poner un ejemplo, no será tenido en cuenta. Además, no se puede anidar estructura en los directorios estándar, es decir, no podemos crear subdirectorios como, por ejemplo, `fragments` dentro de `/res/layout`. Dada esta limitación, Android recomienda utilizar prefijos y barras bajas, como en `activity_main.xml`, para mantener ordenados los archivos.