

Servicios Basados en Localización (LBS)

Tema 6

Desarrollo de aplicaciones en Android

Miguel Ángel Manso
Ramón Alcarria



1

Multilinguaje

- Android tiene características de soporte de multilinguaje muy fáciles de utilizar.
- Hacemos uso de la carpeta values para escribir Strings en varios idiomas
 - values-es, values-en, values-fr
- No solo podemos cambiar los Strings sino también cualquier propiedad del layout, como el tipo de fuente, el tamaño, los colores, etc.

Ejercicio Multilenguaje

- Aplicación que muestre “Hola Mundo” en varios idiomas (dependiendo de la configuración del móvil)



LBS 2014-2015

3

Ejercicio Multilenguaje

- Crear xml para cada lenguaje y cada tamaño de Fuente.
 - Crear carpetas: values-es, values-en, values-it
 - Replicar strings.xml dentro de esas carpetas
- Crear en cada carpeta values-xx un archivo *xml* que se llame *dimens.xml*

```
<?xml version="1.0" encoding="utf-8"?>
<resources>
<dimen name="tamano">50sp</dimen>
</resources>
```

- En main.xml añadimos:

```
android:textSize="@dimen/tamano">
```

LBS 2014-2015

4

Ejercicio Multilinguaje

- Para español:

- strings.xml

```
<?xml version="1.0" encoding="utf-8"?>
<resources>
<string name="base_text">Español</string>
<string name="app_name">Multi-lenguaje</string>
</resources>
```

- dimens.xml

```
<?xml version="1.0" encoding="utf-8"?>
<resources>
<dimen name="tamano">80sp</dimen>
</resources>
```

Ejercicio Multilinguaje

- Para inglés:

- strings.xml

```
<?xml version="1.0" encoding="utf-8"?>
<resources>
<string name="base_text">English</string>
<string name="app_name">Multi-language</string>
</resources>
```

- dimens.xml

```
<?xml version="1.0" encoding="utf-8"?>
<resources>
<dimen name="tamano">50sp</dimen>
</resources>
```

Ejercicio Multilenguaje

- Para italiano:

- strings.xml

```
<?xml version="1.0" encoding="utf-8"?>
<resources>
<string name="base_text">Italiano</string>
<string name="app_name">Multi-language</string>
</resources>
```

- dimens.xml

```
<?xml version="1.0" encoding="utf-8"?>
<resources>
<dimen name="tamano">30sp</dimen>
</resources>
```

Ejercicio Multilenguaje

- El layout:

```
<TextView android:text="@string/base_text"
android:id="@+id/TextView01"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:textSize="@dimen/tamano">
</TextView>
```

Multipantalla

- Las pantallas se diferencian en tamaño y densidad.
 - Tamaño: small, normal, large, xlarge
 - Densidad: low (ldpi), medium (mdpi), high (hdpi), extra high (xhdpi)
 - La orientación de la pantalla también se considera un cambio en el tamaño de la pantalla.

```
MyProject/
res/
  layout/           # default (portrait)
    main.xml
  layout-land/     # landscape
    main.xml
  layout-large/    # large (portrait)
    main.xml
  layout-large-land/ # large landscape
    main.xml
```

LBS 2014-2015

9

Multipantalla

- Provisión de imágenes según densidad
 - xhdpi: 200 %
 - hdpi: 150 %
 - mdpi: 100 %
 - ldpi: 75 %
- *@drawable/miImagen* seleccionará la imagen con la resolución correcta dependiendo de la densidad de pantalla.

```
MyProject/
res/
  drawable-xhdpi/
    awesomeimage.png
  drawable-hdpi/
    awesomeimage.png
  drawable-mdpi/
    awesomeimage.png
  drawable-ldpi/
    awesomeimage.png
```

LBS 2014-2015

10

Multiversión

- ¿Cómo hacer nuestra app compatible con versiones antiguas y nuevas de Android?
 - Recomendado soportar al menos el 90% de los dispositivos
 - `minSdkVersion`: Determina la mínima versión de android compatible con tu app
 - `targetSdkVersion`: Determina la máxima versión de Android que has tenido en cuenta.

Multiversión

- ¿Cómo hacer nuestra app compatible con versiones antiguas y nuevas de Android?
 - Recomendado soportar al menos el 90% de los dispositivos
 - `minSdkVersion`: Determina la mínima versión de android compatible con tu app
 - `targetSdkVersion`: Determina la máxima versión de Android que has tenido en cuenta.

Multiversión

- Determinar la version de Android del dispositivo

```
private void setUpActionBar() {  
    // Make sure we're running on Honeycomb or higher to use ActionBar APIs  
    if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.HONEYCOMB) {  
        ActionBar actionBar = getActionBar();  
        actionBar.setDisplayHomeAsUpEnabled(true);  
    }  
}
```

- Uso de estilos según version de Android

```
<activity android:theme="@android:style/Theme.Dialog">
```

```
<activity android:theme="@android:style/Theme.Translucent">
```

```
<activity android:theme="@style/CustomTheme">
```