


Open · Opened 3 weeks ago by  **Rubén Montero**

Imágenes

Resumen

- Aprenderemos sobre el componente `ImageView`
- Mostraremos una imagen en `MonstersActivity`

Descripción

Hasta ahora hemos usado botones y textos.

Son dos componentes fundamentales en desarrollo móvil, pero a los usuarios también le gusta observar recursos gráficos en las aplicaciones.

`ImageView` es un componente que sirve para mostrar una imagen. Nosotros vamos a usar una imagen que *ya viene* con la aplicación.

¿Ya viene con la aplicación?

Sí. En la carpeta `res/` podemos añadir distintos tipos de *resources* (recursos). Algunos, como ya sabes, son archivos XML que describen las interfaces.

Pues bien, dentro de `res/drawables/` es común añadir *imágenes* (`.png`, `.jpg`) que mostraremos dentro de nuestra aplicación.

¡Ojo! No queremos que estas imágenes ocupen demasiado, pues *aumentarán el tamaño de la aplicación*. ¿Alguna vez has ido a bajar una aplicación de la PlayStore y después de ver que pesaba 200Mb has cancelado la descarga?

La tarea

En primer lugar, en `activity_monsters.xml`, **añade** un `android:id` al `TextView` que ya habíamos creado en una tarea anterior:

```
<TextView
    android:id="@+id/monstersTitle"
```

Luego, **añade** un `ImageView` :

```
<ImageView
    android:layout_width=""
    android:layout_height=""
```

Es *raro* que se use `wrap_content` con un `ImageView` . Pantallas de diferentes tamaños mostrarán la misma imagen más grande o más pequeña, pues los píxeles de la imagen son los mismos.

En su lugar, suele concebirse el `ImageView` como un *lienzo*. Ajustamos dicho *lienzo* empleando *constraints*. Para ello, **usa** `0dp` como *ancho* y *alto*:

```
<ImageView
    android:layout_width="0dp"
    android:layout_height="0dp"
```

A continuación, **añade** *constraints* para posicionarlo *debajo* del título, y *pegado* al borde de la pantalla por izquierda, derecha y abajo. **Así**:

```
<ImageView
    android:layout_width="0dp"
    android:layout_height="0dp"
    app:layout_constraintTop_toBottomOf="@id/monstersTitle"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintBottom_toBottomOf="parent"/>
```

`android:src`

Lo único que nos falta es indicar *qué* imagen debe mostrar. Nosotros vamos a mostrar la imagen de *togemon*.

Añade el siguiente atributo a tu `ImageView` :

```
android:src="@drawable/togemon"
```

Y... una buena práctica

También es *recomendable* añadir el atributo `contentDescription` a las imágenes.

Este atributo sirve para *describir* una imagen por cuestiones de [accesibilidad](#). Es decir, una persona ciega o con visión reducida usará el *lector de pantalla* y agradecerá disponer de una ayuda para comprender el contenido visual de la imagen.

Para terminar, **añade** este otro atributo a tu `ImageView`:

```
android:contentDescription="Un Digimon muy poderoso"
```

¡Listo! Ya hemos mostrado nuestra primera imagen en una aplicación. ¿Qué tal se ve?

Por último

Sube tus cambios al repositorio en un nuevo *commit*.



Rubén Montero [@ruben.montero](#) changed milestone to [%Sprint 1](#) 3 weeks ago



Ania Blanco [@ania.blanco](#) mentioned in commit [5d9c4e55](#) 1 week ago