




<https://github.com/jrodriguezballesster/GamesExplosion.git>

PRÁCTICA 4: Gestionar una pequeña colección de juegos

Vamos a crear una App con 3 actividades para la gestión de una pequeña colección de videojuegos.


Resultado final

 Vease captf1  Vease captf2  Vease captf3


Version 1

Ejercicio 1: (30%)


- En la actividad principal cambiaremos el icono de la app por uno parecido o igual al de la figura y le pondremos como nombre PMDM y nuestro Nombre y apellido

 Vease capt1

- Implementamos un ListView simple como el de la figura

 Vease capt2

- El modelo de datos incluirá la clase Game con los atributos que creas conveniente. Carga 4 o 5 juegos al iniciar la aplicación para probar. La imagen de cada juego cárgala como un mipmap en tu aplicación. Si cargas 4 juegos ponle a cada juego la imagen que toque.

 Vease capt3


Ejercicio 2: (20%)

Al hacer click sobre cualquier elemento de la lista se abrirá la actividad Game Details. En ella se muestra la información de cada juego. Hay muchas formas de diseñar el layout, puedes hacerlo con un Relative Layout con la estructura siguiente:


Para posicionar los elementos puedes ayudarte de propiedades para ajustar márgenes y padding y otras tales como:

`android:layout_below="@+id/textViewSummary"` Esta propiedad sitúa al elemento justo debajo del elemento con id `textViewSummary`

`android:layout_toRightOf="@id/imageView"` Esta propiedad sitúa al elemento justo a la derecha del elemento con id `imageView`

 Vease capt6


Hay que implementar también la opción de volver a la actividad principal con:
`getSupportActionBar().setDisplayHomeAsUpEnabled(true);`

 Vease capt1

Ejercicio 3: (25%)

Crea un adaptador propio con un layout propio para visualizar cada elemento en la ListView.

El layout propio debe mostrar un icono de la plataforma: PC, SWITCH, PS4 y XONE; el título, la nota y la fecha de publicación. El diseño del layout es libre.


 Vease capt7

Si quieres puedes hacer que el layout de cada elemento sea de color blanco para los elementos pares y de color gris para los impares. Lo puedes hacer cargando 2 layouts diferentes por ejemplo.

Version 2


Mejorar Actividad 2

- * Se ha renombrado la activity a GameDetails como se pedía.
- * Se ha implementado la puntuación como un sistema de mostrar estrellas.
- * Se muestran las plataformas como iconos.
- * Se ha mejorado el diseño de la activity

 Vease capt6

Completar actividad 3

El layout propio debe mostrar un icono de la plataforma: PC, SWITCH, PS4 y XONE; el título, la nota y la fecha de publicación. El diseño del layout es libre.


 Vease capt8

Se han añadido la fecha, la puntuación y los iconos de las plataformas como mipmap. Las imágenes de los juegos se han guardado directamente en la carpeta drawable. Por esto

- * Se ha remodelado la clase Juego.
- * Se ha mejorado la lectura de las Plataformas y no hay que poner valores vacíos.
- * Se ha rediseñado el layout
- * Se han añadido los iconos como mipmaps desde New->Image Asset

Otras mejoras


- * Se ha modificado el Style del ActionBar
- + Se han dejado comentado lineas para hacer el texto responsivo

 Vease capt9

Version 3

Finalizar actividad 3

Si quieres puedes hacer que el layout de cada elemento sea de color blanco para los elementos pares y de color gris para los impares. Lo puedes hacer cargando 2 layouts diferentes por ejemplo.

 Vease capt10

Se ha realizado con un solo layout cambiando la propiedad background dentro de un if

mas mejoras

Se ha añadido una activity inicial de la aplicacion; Se accede al siguiente Activity clickando en cualquier lugar

Dificultades de realizacion

- * Diseño de las activitys, layouts debido a que me gusta poco diseñar
- * La dificultad de comprender y usar los distintos tipos layouts
- * Intentar mantener valores responsivos.
- * Poder acceder a los archivos imagenes desde codigo

al final los guardo como int en la clase juego el int se obtiene tan sencillo como poner `R.carpeta.nombreimagen`

- * La puntuacion la marco haciendo visibles iconos Star_on.

He intentado hacerlo creando iconos Star_on/Star_off pero se queda pendiente para una proxima mejora.