

# GALERIA DE ARTE

Iván Señarís Veiras – Grupo B

Centro: IES San Clemente

## 1 – Título

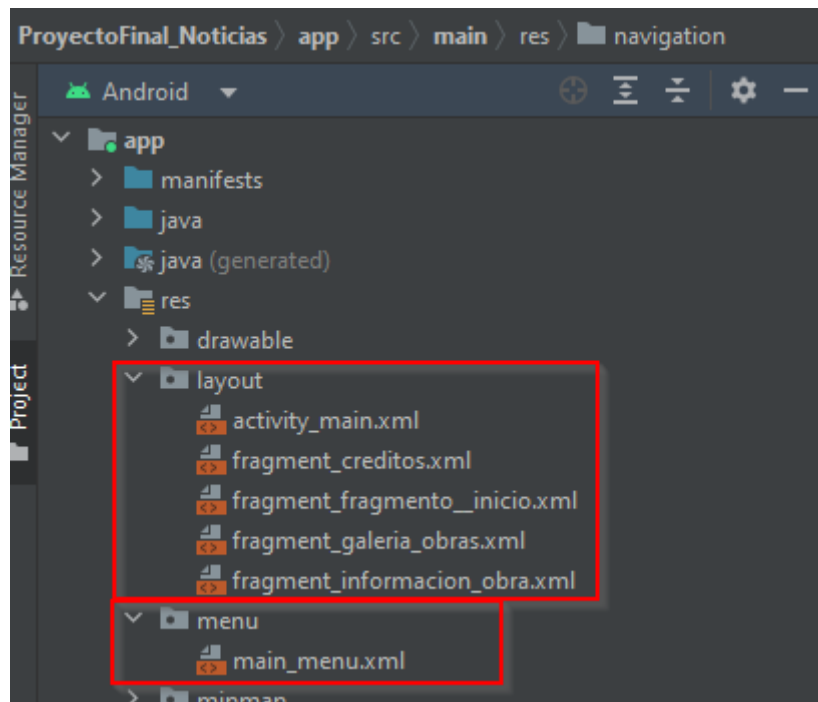
Mi aplicación se llamará **Galería de Arte**, ya que es un nombre representativo y que encapsula muy bien la idea y finalidad de mi aplicación.

## 2 – Concepto

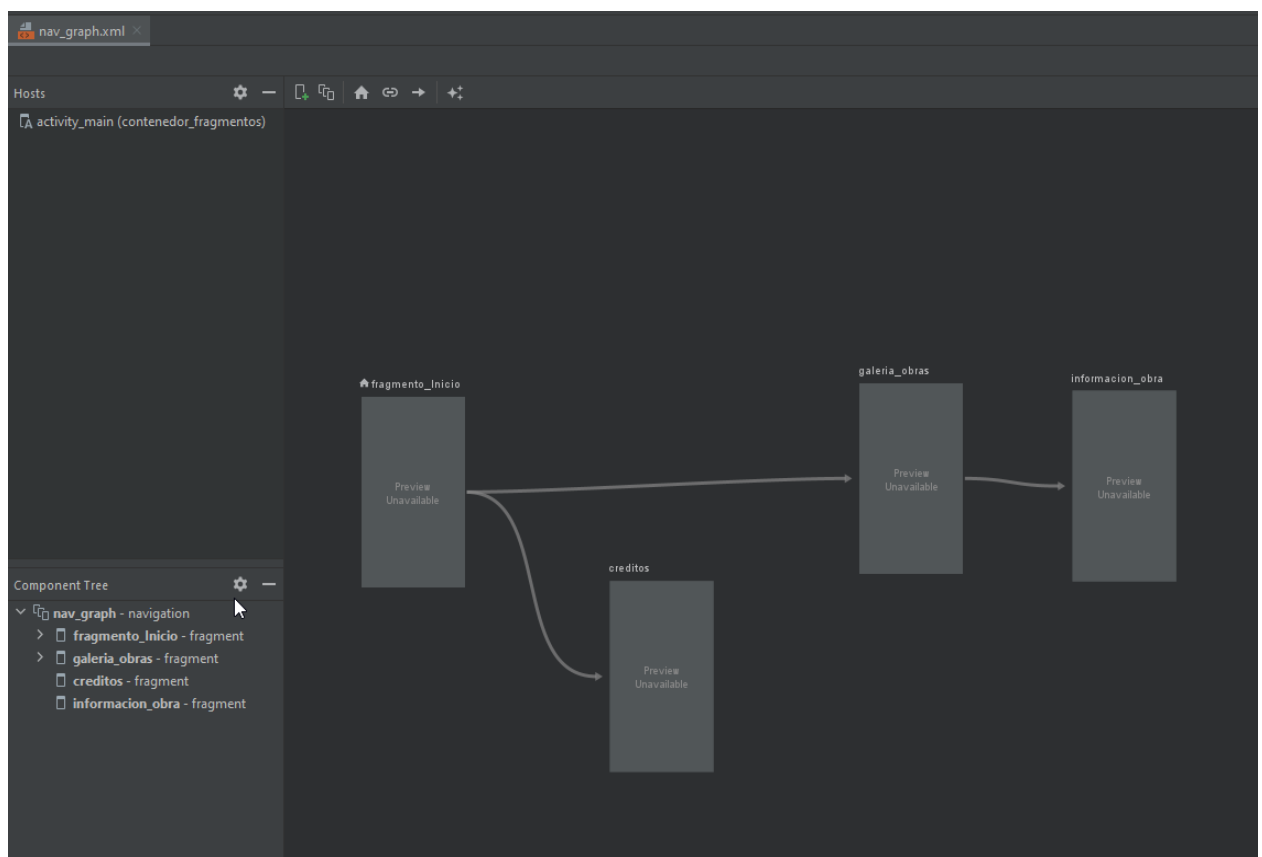
La idea principal de la aplicación es tener una galería de obras de arte online (en una aplicación de móvil), de forma que sea accesible para todo el mundo en cualquier lugar y cualquier momento. Esto surge de que muchas de las obras (pinturas y esculturas) se encuentran localizadas en distintos museos o galerías de arte alrededor del mundo y visitarlas y conocer información sobre ellas puede ser muy costoso y complicado (no está al alcance de todos), lo que pretendo con esta aplicación es poder hacer llegar el arte todo el mundo desde cualquier lugar.

## 3 – Navegación

La aplicación cuenta con 4 fragmentos (layouts) y un menú en la parte superior:



En cuanto a la navegación entre fragmentos usamos el componente Navigation, se puede ver de forma visual mediante el archivo `nav_graph.xml` :



## Fragmento de Inicio

Este es el fragmento inicial que se carga desde el archivo nav\_graph.xml (es el fragmento que cargamos al iniciar la aplicación)

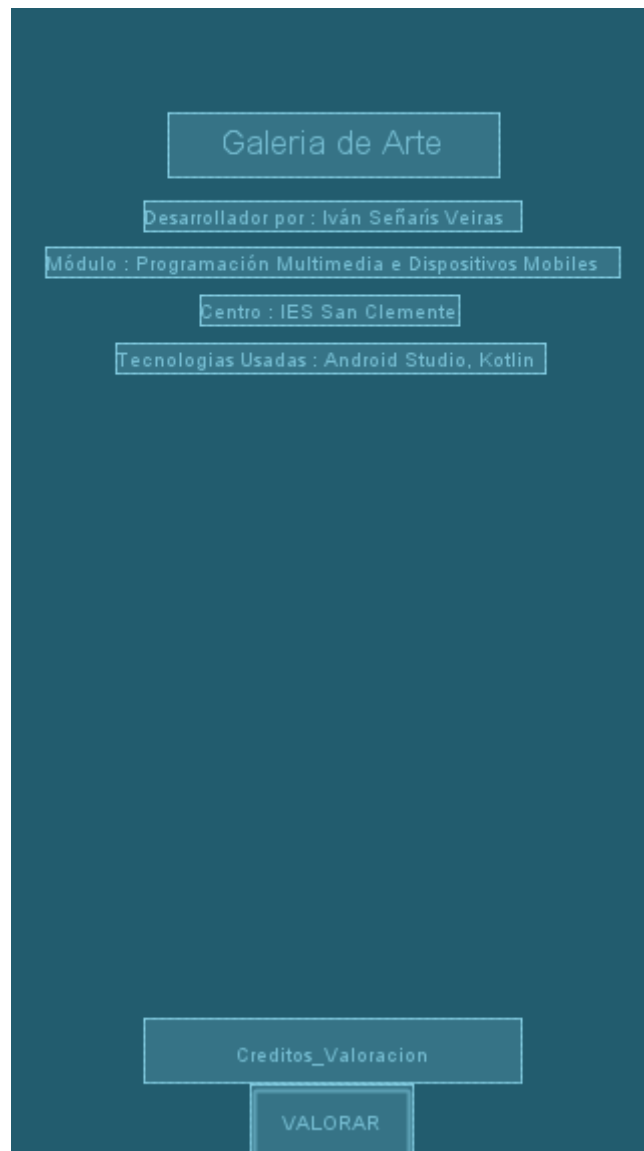


Como se puede apreciar en la imagen el fragmento de inicio cuenta con un constraint layout que engloba un elemento de `<ImageView>` y dos elementos de tipo `<Button>`, dichos elementos cuentan con constraint que hacen que su distribución se mantenga con una relación de aspecto en función de si la pantalla está en vertical u horizontal.

Los botones de dicho fragmento permiten navegar a otros dos fragmentos que veremos a continuación.

## Fragmento Créditos

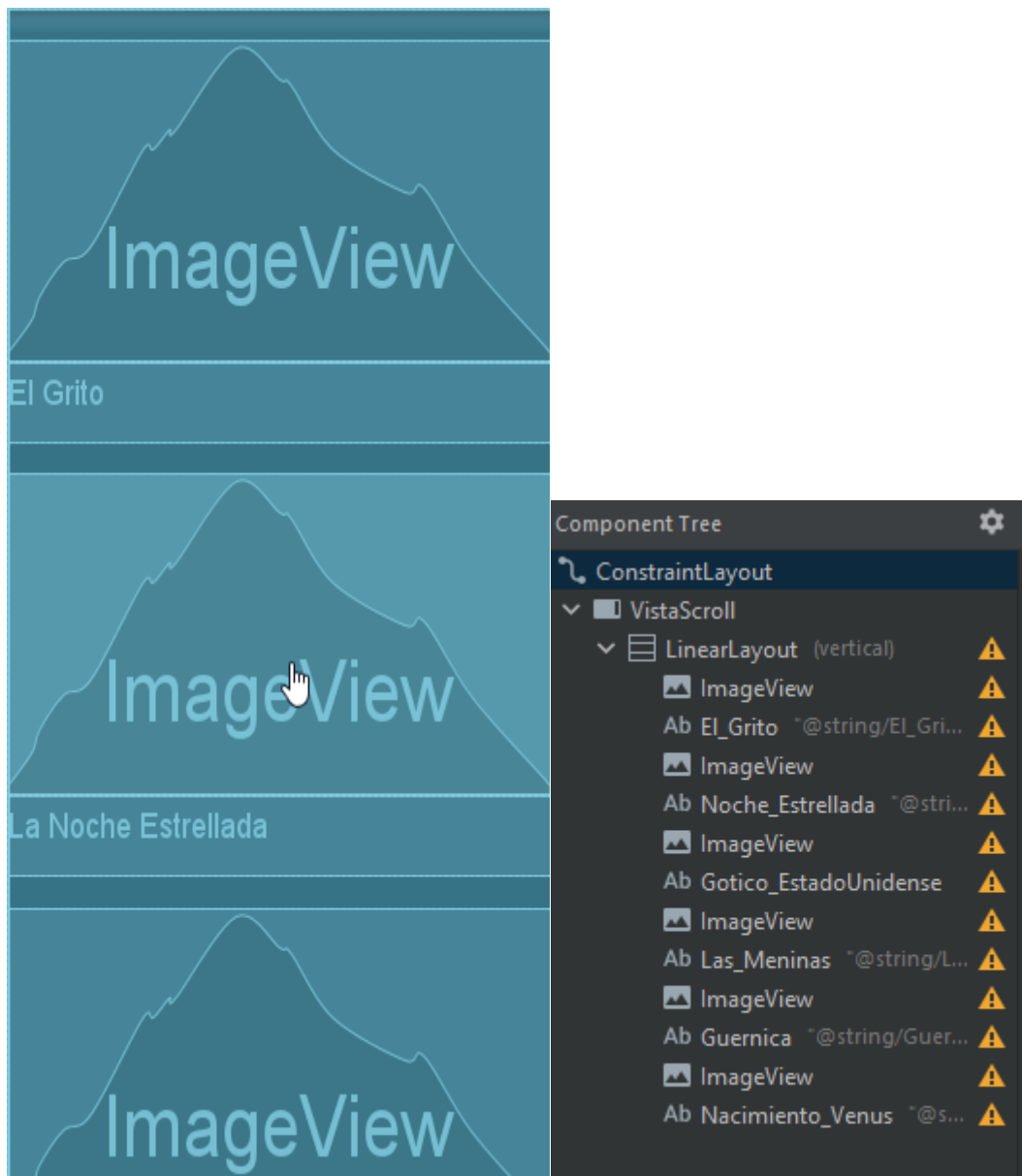
Este fragmento contiene información / créditos sobre la aplicación, dicho fragmento se muestra / carga al pinchar en el botón Información del fragmento anterior.



Cuenta con un constraint layout que contiene elementos de tipo TextView cuyo contenido se obtiene mediante @id/name buscando en el archivo string.xml de la carpeta values (es recomendable no tener los textos en la aplicación mediante hardcoding)

## Fragmento Galería de Obras

Este fragmento contiene una “galería” con las obras que le hemos añadido, la galería está formada por una imagen y su correspondiente título. Dicho fragmento cuenta con un ScrollView que nos permite hacer scroll de la galería (una limitación del ScrollView es que solo permite un elemento hijo), es por esto que tenemos un LinearLayout con disposición vertical para generar la galería, en ese LinearLayout incluimos elementos ImageView y TextView



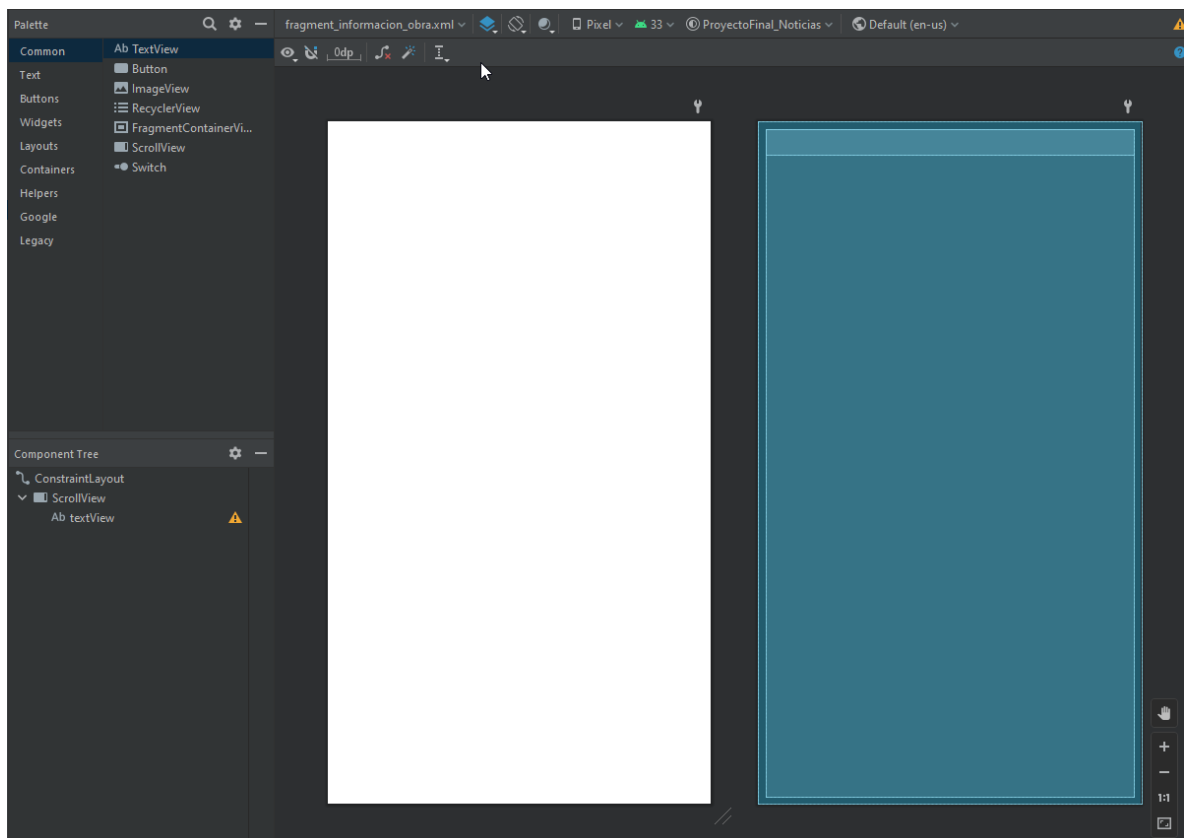
Para navegar hasta este fragmento deberemos de pinchar en el botón Visualizar el fragmento inicial. Todos los títulos de las obras de esta galería se almacenan en ul

fichero strings.xml y es en el fragmento donde indicamos que el texto del textview se debe obtener de dicho xml (strings.xml) en lugar de meterlo en hardcoding.

```
<TextView
    android:id="@+id/El_Grito"
    android:layout_width="match_parent"
    android:layout_height="50dp"
    android:text="@string/El_Grito"
    android:textSize="25dp"
    android:gravity="center_horizontal">
</TextView>
```

### Fragmento Información de la Obra

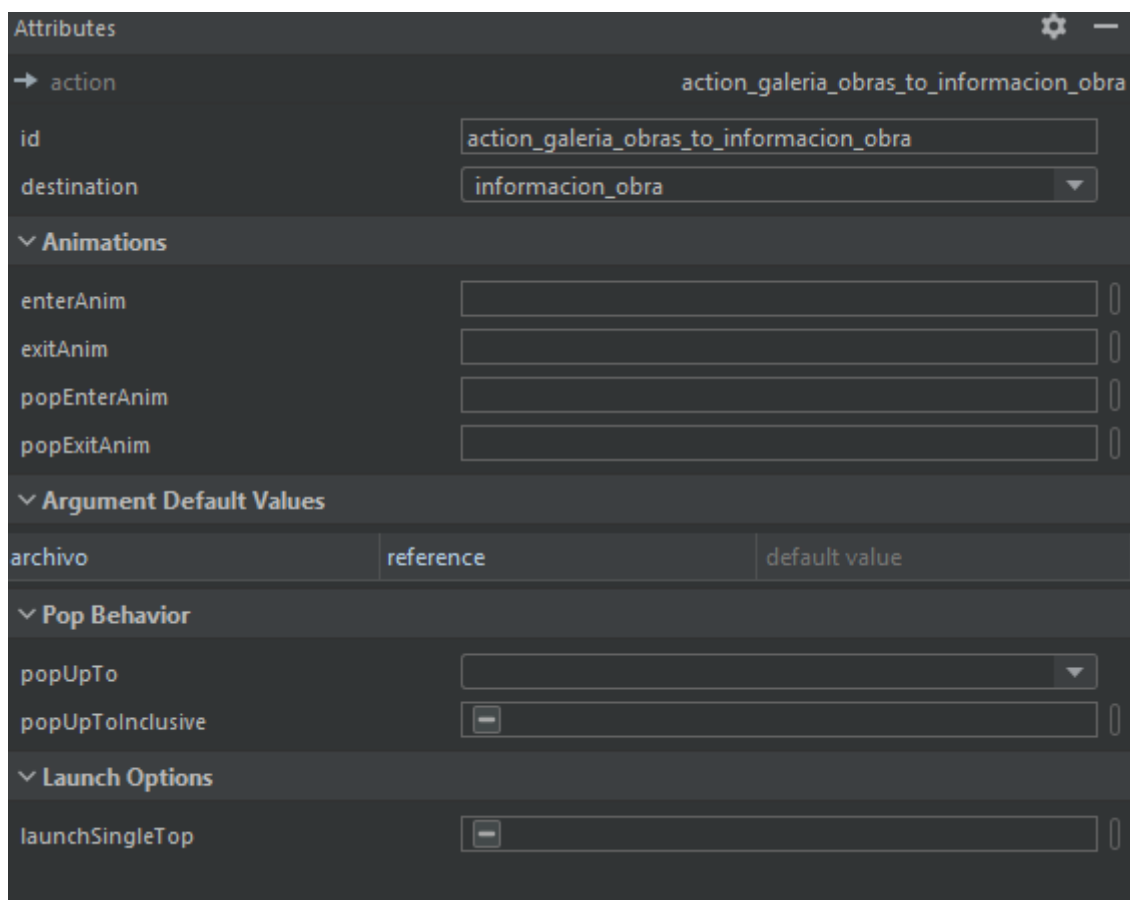
Este fragmento contiene información de la obra seleccionada en la galería, se accede/navega hasta él pinchando en el título de la obra que deseemos en la galería.



Como se puede apreciar este fragmento únicamente cuenta con un ScrollView y su único elemento hijo un textView. En dicho textView es donde completaremos/añadiremos la documentación/información acerca de la obra seleccionada.

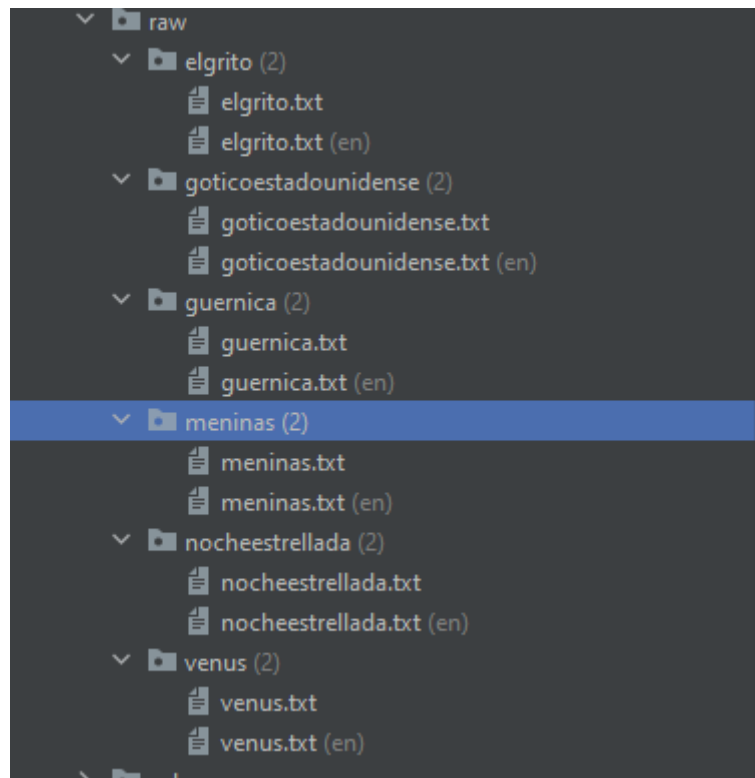
¿Pero como sabemos que información mostrar en función de la obra seleccionada?

Pues bien cuando navegamos desde el fragmento galería de obras al fragmento información de la obra, estamos pasando un parámetro de tipo referenciado:



Este parámetro contiene el nombre del archivo de la carpeta raw (donde se encuentran los archivos de texto plano que contiene la información de cada obra):





Por lo tanto al pulsar en el título de la obra en la galería estamos pasando al fragmento de información el nombre del archivo al que debe de leer para obtener los datos, una vez tenemos en el fragmento información de la obra dicho archivo simplemente lo leemos y mostramos la información obtenida de leerlo en el textview :

```

16
17 class informacion_obra : Fragment() {
18
19
20
21 override fun onCreateView(
22     inflater: LayoutInflater, container: ViewGroup?,
23     savedInstanceState: Bundle?
24 ): View? {
25     // Inflate the layout for this fragment
26     val view = inflater.inflate(R.layout.fragment_informacion_obra, container, attachToRoot: false)
27
28     //Obtengo referencia a los textviews
29     val textView = view.findViewById<TextView>(R.id.textView)
30     val parametro = informacion_obraArgs.fromBundle(requireArguments()).archivo
31
32     var string: String? = ""
33     val stringBuilder = StringBuilder()
34     val inputStream = this.resources.openRawResource(parametro)
35     val reader = BufferedReader(InputStreamReader(inputStream))
36
37     string = reader.readLine()
38     var valores = string.split(...delimiters: "\n")
39
40     //Añado el texto correspondiente al textview
41
42     textView.text = string
43
44     return view;
45
46
47
48

```

## Menú superior

Aunque no se trata de un fragmento, también tenemos un menú superior que contiene ciertos elementos que son fijos en la aplicación y que no varían en función del fragmento en el que estamos:



## Grafo de Navegación

Finalmente tenemos el fichero de navegación que nos permite indicar las rutas de navegación, así como los parámetros que pasamos entre un fragmento y otro:

```
1 <?xml version="1.0" encoding="utf-8"?>
2 <navigation xmlns:android="http://schemas.android.com/apk/res/android"
3   xmlns:app="http://schemas.android.com/apk/res-auto"
4   android:id="@+id/nav_graph"
5   app:startDestination="@id/fragmento_inicio">
6   <fragment
7     android:id="@+id/fragmento_inicio"
8     android:name="com.example.proyectofinal_noticias.Fragmento_inicio"
9     android:label="Fragmento_inicio" >
10    <action
11      android:id="@+id/action_fragmento_inicio_to_galeria_obras"
12      app:destination="@id/galeria_obras" />
13    <action
14      android:id="@+id/action_fragmento_inicio_to_creditos"
15      app:destination="@id/creditos" />
16  </fragment>
17  <fragment
18    android:id="@+id/galeria_obras"
19    android:name="com.example.proyectofinal_noticias.galeria_obras"
20    android:label="galeria_obras" >
21    <action
22      android:id="@+id/action_galeria_obras_to_informacion_obra"
23      app:destination="@id/informacion_obra" />
24  </fragment>
25  <fragment
26    android:id="@+id/creditos"
27    android:name="com.example.proyectofinal_noticias.creditos"
28    android:label="creditos" />
29  <fragment
30    android:id="@+id/informacion_obra"
31    android:name="com.example.proyectofinal_noticias.informacion_obra"
32    android:label="informacion_obra" >
33    <argument
34      android:name="archivo"
35      app:argType="reference" />
36  </fragment>
37 </navigation>
```

En él se indica cual es el fragmento que se carga al inicio de la aplicación, las rutas que siguen los fragmentos y los argumentos/parámetros que se pasan.

## 4 – Opinión Personal

En lo personal estoy muy contento con el resultado creo que para el periodo de tiempo dedicado he conseguido completar una aplicación en la que se demuestran los conocimientos adquiridos durante el curso junto con algún comportamiento más avanzado.

Por otro lado, la aplicación tiene un gran potencial de mejora implementar un Login de usuario y su perfil, implementar un calendario con eventos de arte, implementar un

buscador en la galería que te permita buscar las obras por Museo o por tipo de obra y se me pueden ocurrir un montón de mejoras y extras que añadirle.

En resumen, creo que la aplicación es sencilla y está completa (dentro de lo establecido sin extendernos en las mejoras) y lo más importante es funcional y puede ayudar a cubrir una necesidad.