



**UNIVERSIDAD AUTÓNOMA  
DE AGUASCALIENTES**

**Materia:**

**Diseño para Móviles**

**Maestra:**

**Margarita Mondragón Arellano**

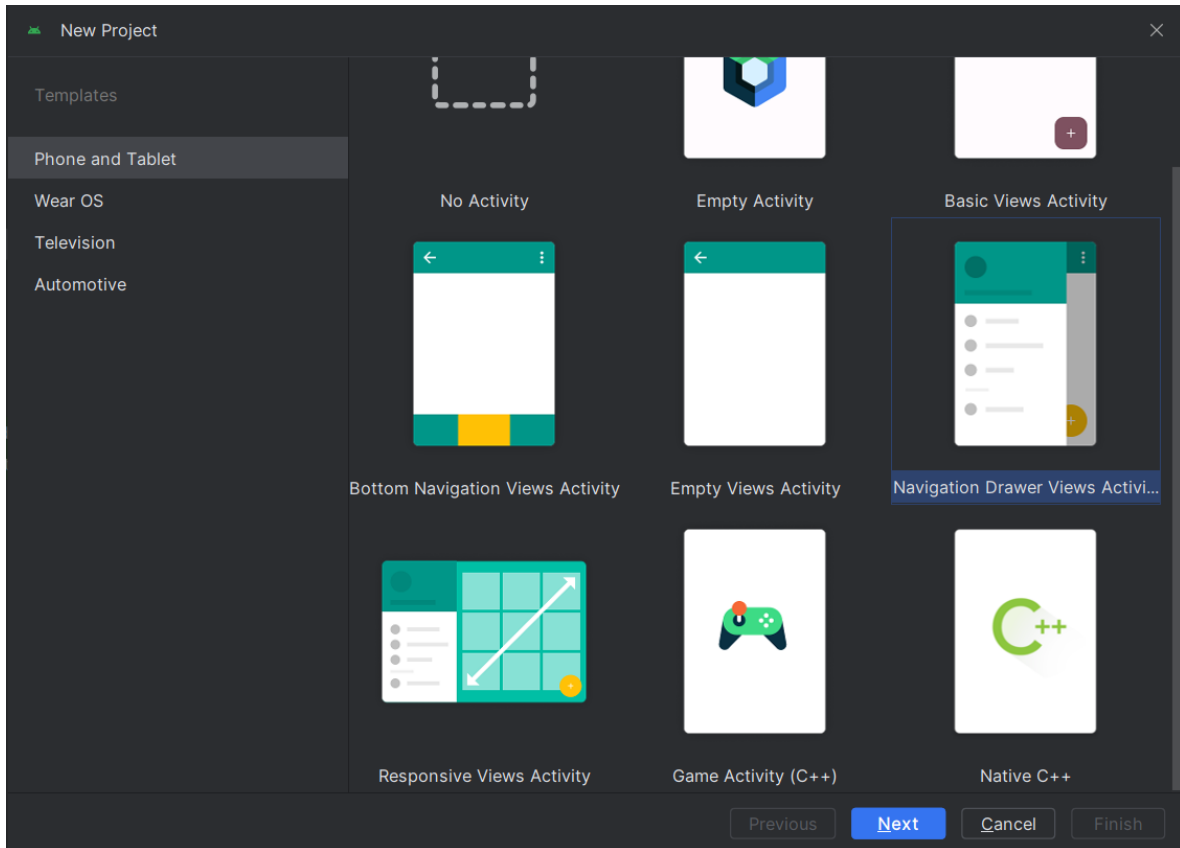
**Examen parcial 2:**

**Dulcería**

**Alumna:**

**Andrea Sotelo Barajas.**

Se crea una nueva aplicación con navegation drawer que crea varios archivos que se explicaran abajo



Código para controlar fragments de la pantalla primer pantalla

```
1 package com.example.ex_2p_andy.ui.home
2
3 > import ...
4
11
12 class HomeFragment : Fragment() {
13
14     private var _binding: FragmentHomeBinding? = null
15
16     // This property is only valid between onCreateView and
17     // onDestroyView.
18     private val binding get() = _binding!!
19
20     override fun onCreateView(
21         inflater: LayoutInflater,
22         container: ViewGroup?,
23         savedInstanceState: Bundle?
24     ): View {
25         val homeViewModel =
26             ViewModelProvider( owner, this).get(HomeViewModel::class.java)
27
28         _binding = FragmentHomeBinding.inflate(inflater, container, attachToParent: false)
29         val root: View = binding.root
30
31         val textView: TextView = binding.textHome
32         homeViewModel.text.observe(viewLifecycleOwner) { (it:String?)
33             textView.text = it
34         }
35         return root
36     }
37
38     override fun onDestroyView() {
39         super.onDestroyView()
40     }
41 }
```

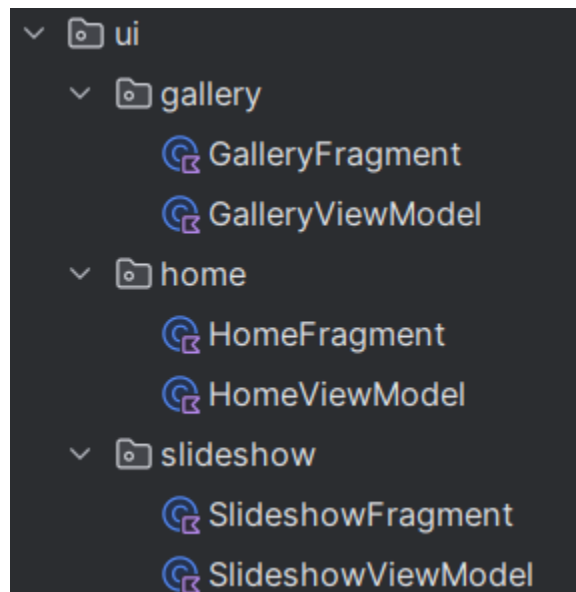
> example > ex\_2p\_andy > ui > home > @ HomeFragment

12:7 LF UTF-8 4 spaces

Mas código para controlar la primer pantalla

```
1 package com.example.ex_2p_andy.ui.home
2
3 import androidx.lifecycle.LiveData
4 import androidx.lifecycle.MutableLiveData
5 import androidx.lifecycle.ViewModel
6
7 class HomeViewModel : ViewModel() {
8
9     private val _text = MutableLiveData<String>().apply { (this: MutableLiveData<String>)
10         value = "This is home Fragment"
11     }
12     val text: LiveData<String> = _text
13 }
```

Se repite el código para las otras dos pantallas con mas fragments y demás como se ve en la siguiente imagen



En el main activity se agrego la siguiente función para el paso de mensaje

```
49
50     fun sendMessage(view: View) {
51         val editText = findViewById<EditText>(R.id.editText)
52         val message = editText.text.toString()
53         val intent = Intent(packageContext: this, Pantalla_info::class.java).apply { this.intent
54             | putExtra(EXTRA_MESSAGE, message)
55         }
56
57         startActivity(intent)
58     }
59 }
```

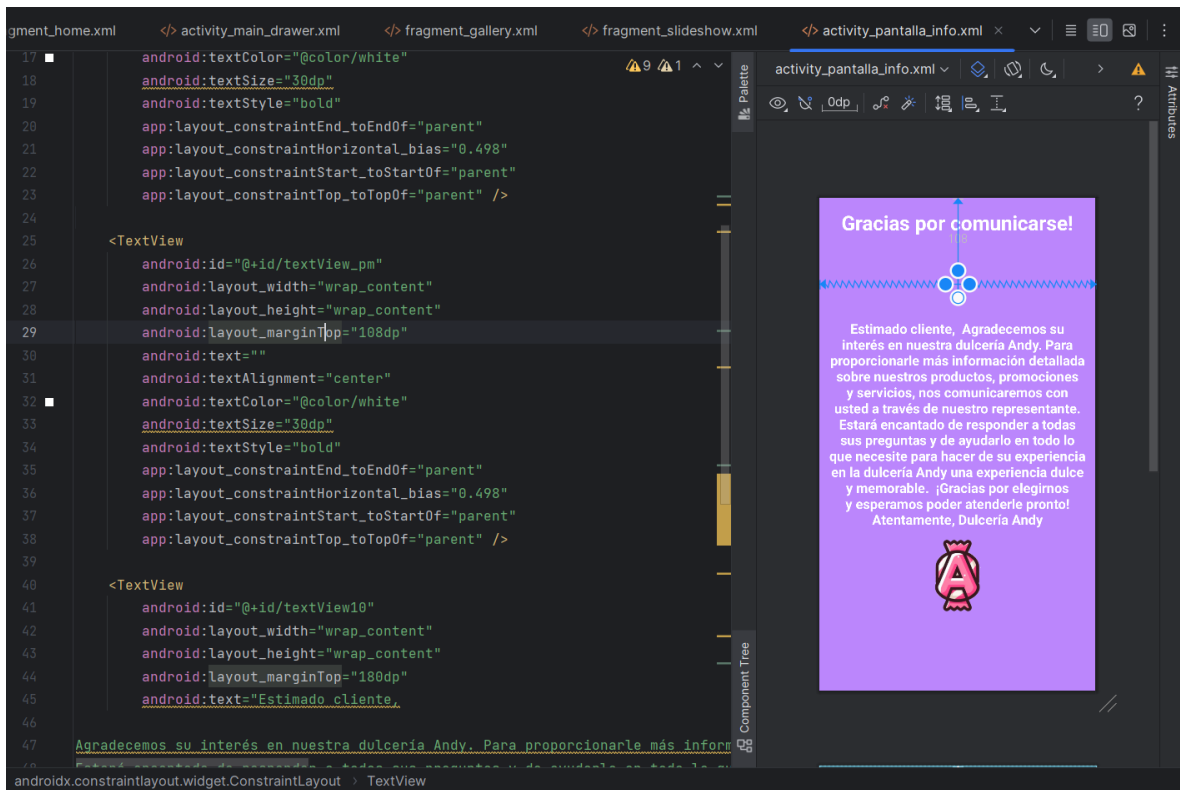
Se creo una clase con el siguiente código

```
c_pasar_mensaje.kt x GalleryFragment.kt <> fragment_home.xml <> activity_main_drawer.xml <> fragment_gallery.xml <> fragment_slidesh...
1 package com.example.ex_2p_andy
2
3 import android.content.Intent
4 import androidx.appcompat.app.AppCompatActivity
5 import android.os.Bundle
6 import android.view.View
7 import android.widget.EditText
8
9 // Constante para identificar el mensaje extra enviado a la siguiente actividad
10 const val EXTRA_MESSAGE = "com.example.myfirstapp.MESSAGE"
11
12 <> class CodigoPasarMensaje : AppCompatActivity() {
13     override fun onCreate(savedInstanceState: Bundle?) {
14         super.onCreate(savedInstanceState)
15         setContentView(R.layout.fragment_slideshow) // Establece el diseño de la actividad como activity_sp.xml
16     }
17
18     // Método invocado al hacer clic en el botón de enviar mensaje
19     fun sendMessage(view: View) {
20         // Obtiene el texto ingresado en el EditText
21         val editText = findViewById<EditText>(R.id.editText)
22         val message = editText.text.toString()
23
24         // Crea un Intent para abrir la siguiente actividad (SP) y le pasa el mensaje como extra
25         val intent = Intent(packageContext: this, Pantalla_info::class.java).apply { this.intent
26             | putExtra(EXTRA_MESSAGE, message) // Agrega el mensaje como extra al intent
27         }
28
29         // Inicia la siguiente actividad con el intent creado
30         startActivity(intent)
31     }
32 }
```

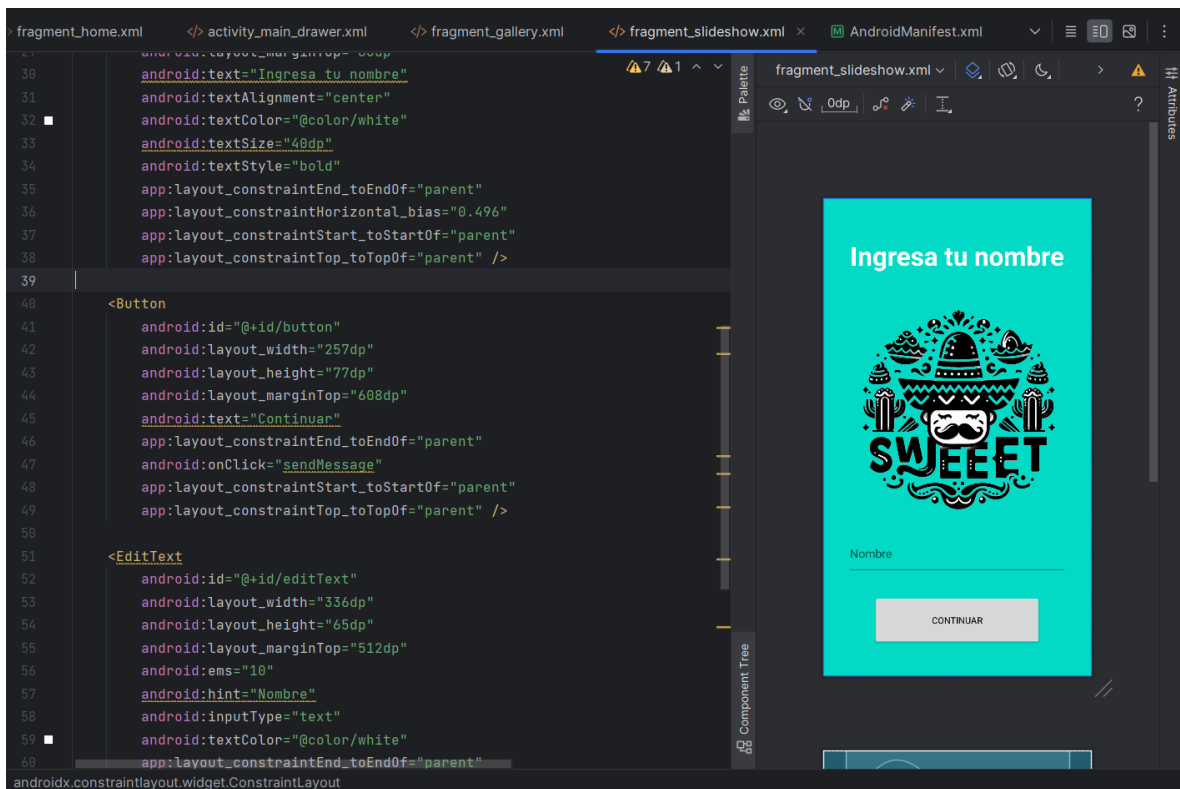
Se creo una nueva pantalla para el paso del mensaje con el siguiente codigo

```
/Fragment.kt <> fragment_home.xml <> activity_main_drawer.xml <> fragment_gallery.xml <> fragment_slideshow.xml Pantalla_info.kt x
1 package com.example.ex_2p_andy
2
3 import androidx.appcompat.app.AppCompatActivity
4 import android.os.Bundle
5 import android.widget.TextView
6 import android.widget.Toast
7
8 <> class Pantalla_info : AppCompatActivity() {
9     override fun onCreate(savedInstanceState: Bundle?) {
10         super.onCreate(savedInstanceState)
11         setContentView(R.layout.activity_pantalla_info)
12
13         Toast.makeText(context: this, text: "Se dio mas info correctamente", Toast.LENGTH_SHORT).show()
14
15         // Obtener el mensaje enviado desde MainActivity
16         val message = intent.getStringExtra(EXTRA_MESSAGE)
17
18         // Encontrar el TextView y establecer el mensaje
19         val textView = findViewById<TextView>(R.id.textView_pm)
20         textView.text = message
21     }
22 }
```

Y el siguiente xml donde el mensaje se pasará a un textview



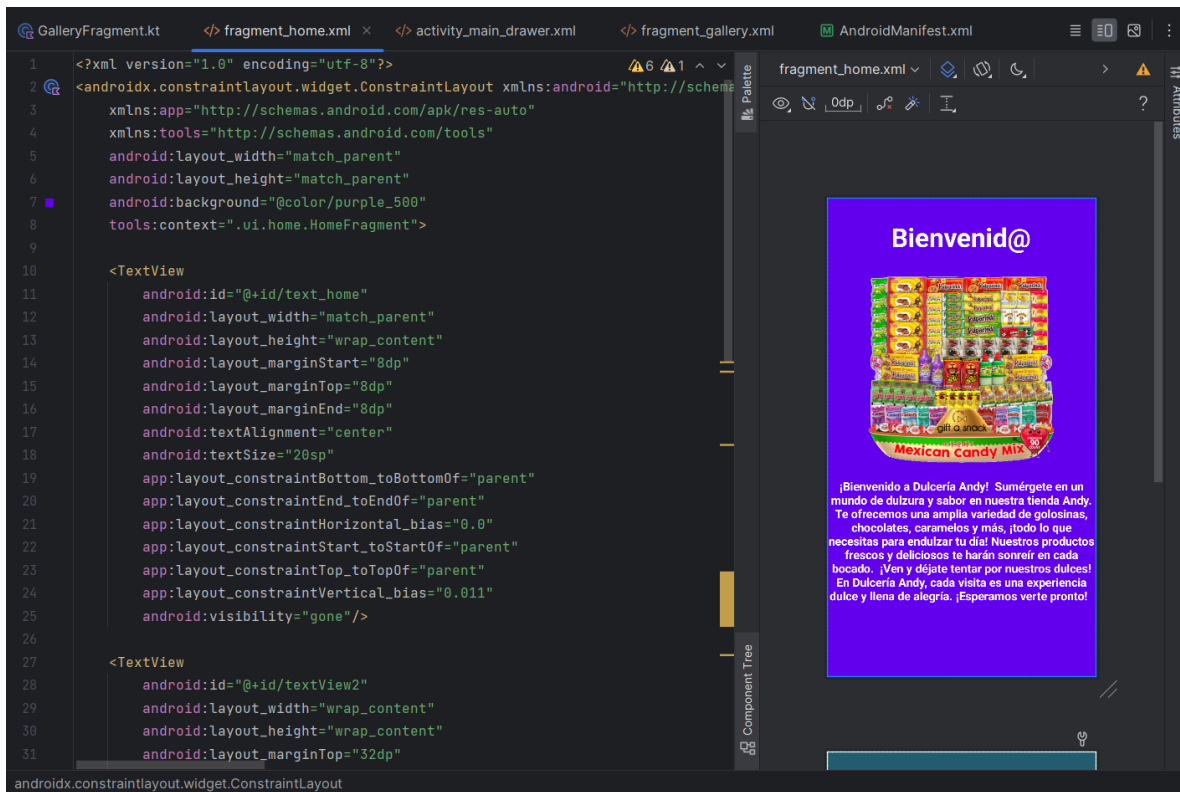
Se crea el xml donde se manda el mensaje



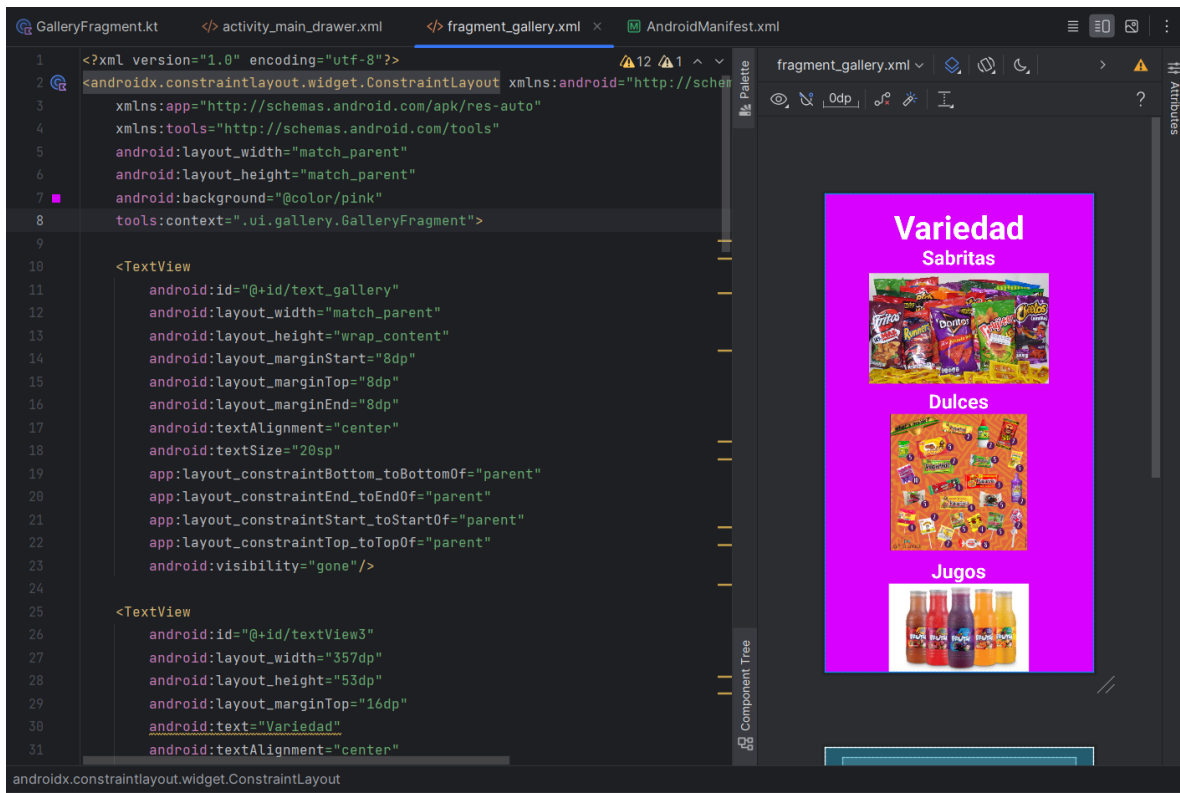
Y en el xml donde se tiene el botón se agregó lo siguiente para que funcione

```
<Button  
    android:id="@+id/button"  
    android:layout_width="257dp"  
    android:layout_height="77dp"  
    android:layout_marginTop="608dp"  
    android:text="Continuar"  
    app:layout_constraintEnd_toEndOf="parent"  
    android:onClick="sendMessage"  
    app:layout_constraintStart_toStartOf="parent"  
    app:layout_constraintTop_toTopOf="parent" />
```

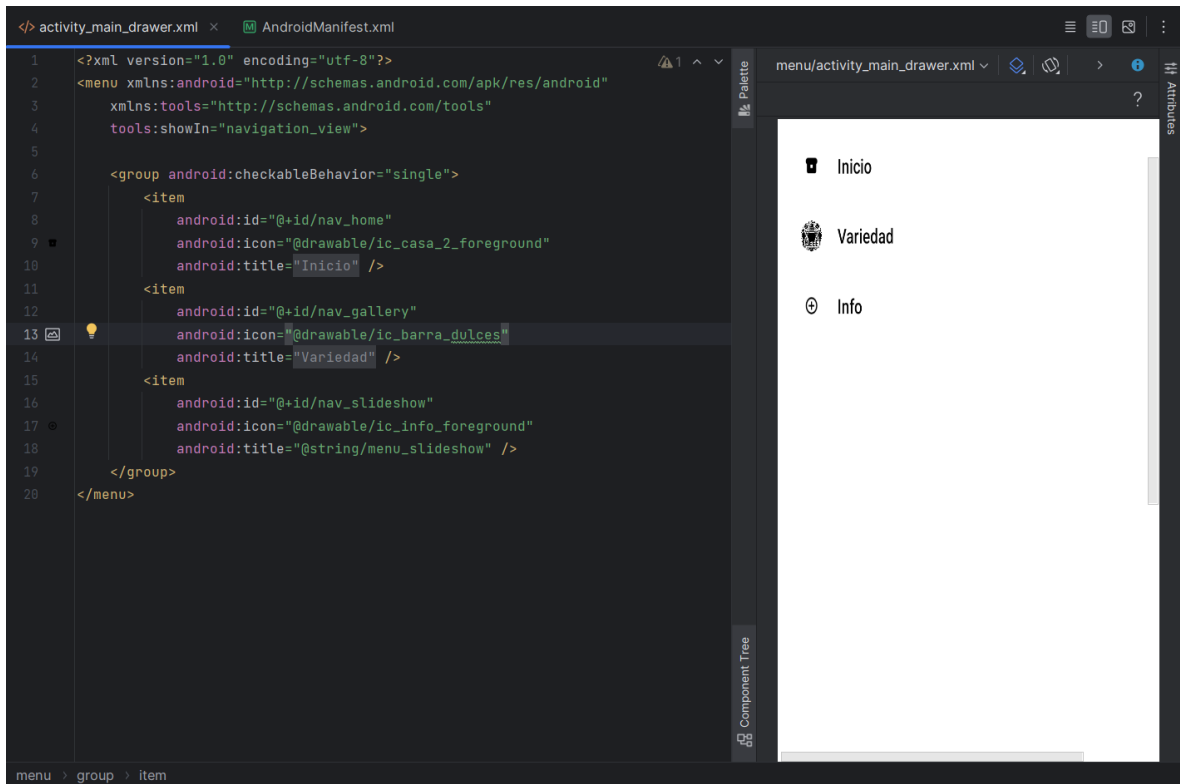
Se crea el xml de bienvenida



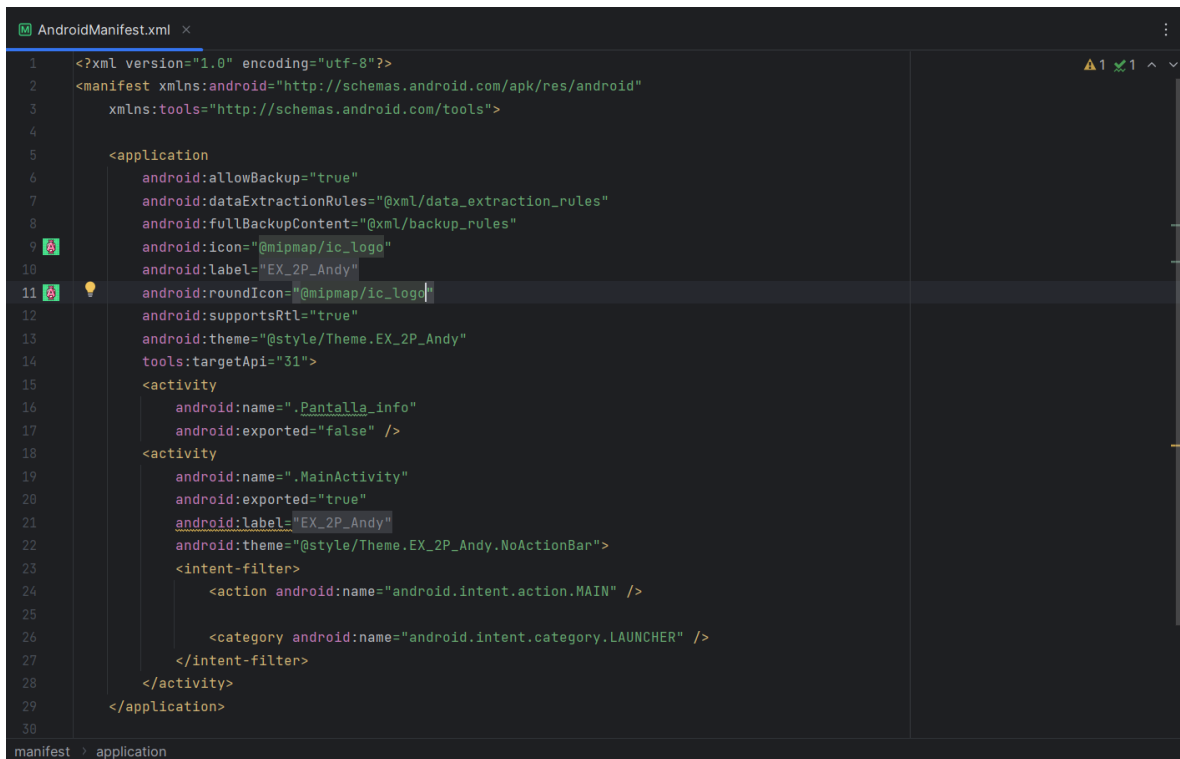
Se crea el Segundo xml donde se muestran los productos



Se modifican iconos y nombres

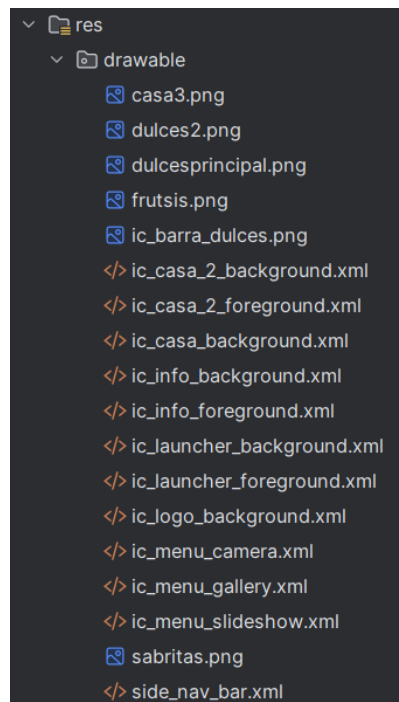


Se crea en la carpeta drawable y modifica el logo en el manifest de la aplicación












```
1 <?xml version="1.0" encoding="utf-8"?>
2 <manifest xmlns:android="http://schemas.android.com/apk/res/android"
3     xmlns:tools="http://schemas.android.com/tools">
4
5     <application
6         android:allowBackup="true"
7         android:dataExtractionRules="@xml/data_extraction_rules"
8         android:fullBackupContent="@xml/backup_rules"
9         android:icon="@mipmap/ic_logo"
10        android:label="EX_2P_Andy"
11        android:roundIcon="@mipmap/ic_logo"
12        android:supportRtl="true"
13        android:theme="@style/Theme.EX_2P_Andy"
14        tools:targetApi="31">
15        <activity
16            android:name=".Pantalla_info"
17            android:exported="false" />
18        <activity
19            android:name=".MainActivity"
20            android:exported="true"
21            android:label="EX_2P_Andy"
22            android:theme="@style/Theme.EX_2P_Andy.NoActionBar">
23            <intent-filter>
24                <action android:name="android.intent.action.MAIN" />
25
26                <category android:name="android.intent.category.LAUNCHER" />
27            </intent-filter>
28        </activity>
29    </application>
30
```

Se agregaron imágenes en el proceso de creación






Se agregaron colores para usar en el proyecto

```
</> mobile_navigation.xml  </> colors.xml  x
1  <?xml version="1.0" encoding="utf-8"?>
2  <resources>
3      <color name="purple_200">#FFBB86FC</color>
4      <color name="purple_500">#FF6200EE</color>
5      <color name="purple_700">#FF3700B3</color>
6      <color name="teal_200">#FF03DAC5</color>
7      <color name="teal_700">#FF018786</color>
8      <color name="black">#FF000000</color>
9      <color name="white">#FFFFFFFF</color>
10   <color name="pink">#D800FF</color>
11 </resources>
```

En el xml strings se agregaron varios títulos de otros xml

```
</> mobile_navigation.xml  </> strings.xml  x
i Edit translations for all locales in the translations editor.
1  <resources>
2  <string name="app_name">EX_2P_Andy</string>
3  <string name="navigation_drawer_open">Open navigation drawer</string>
4  <string name="navigation_drawer_close">Close navigation drawer</string>
5  <string name="nav_header_title">Dulceria Andy</string>
6  <string name="nav_header_subtitle">Los mejores dulces</string>
7  <string name="nav_header_desc">Navigation header</string>
8  <string name="action_settings">Settings</string>
9
10 <string name="menu_home">Inicio</string>
11 <string name="menu_gallery">Variedad</string>
12  <string name="menu_slideshow">Info</string>
13 </resources>
```

Acá se muestra un mensaje al usuario por medio de un toast

