

# Actividad | 3# | Diseño de Aplicación(Parte Final)

## Desarrollo de Aplicaciones Móviles II

---

Ingeniería en Desarrollo de Software



TUTOR: Eduardo Israel Castillo García

ALUMNO: Casandra Montserrat Ortiz Cortes

FECHA:30/11/2025

## Índice

**Introducción.....1**

**Descripción.....2**

**Justificación.....3**

**Desarrollo.....4**

- **Interfaz**
- **Codificación**
- **Prueba de la aplicación**

**Conclusión.....5**

**Referencia**

## **Introducción**

El diseño de apps es un proceso esencial para garantizar que la aplicación sea funcional y atractiva para los usuarios. Se trata de una disciplina que involucra la estructuración de la navegación de la aplicación, la definición de las directrices visuales y la creación de una experiencia de usuario satisfactoria.

En otras palabras, el diseño de aplicaciones móviles implica la planificación y el diseño de cada elemento visual y funcional de la aplicación. Esto incluye desde la disposición de los botones hasta la forma en que se muestra la información en pantalla. El diseño de apps también tiene como objetivo asegurarse de que la aplicación sea fácil de usar y proporcione una experiencia de usuario agradable.

Estudiar al usuario modelo: En primer lugar, debemos entender a quién va dirigida la aplicación. Debemos conocer las necesidades y preferencias de los usuarios para que el diseño se adapte a ellas, UI aplicada.

## **Descripción**

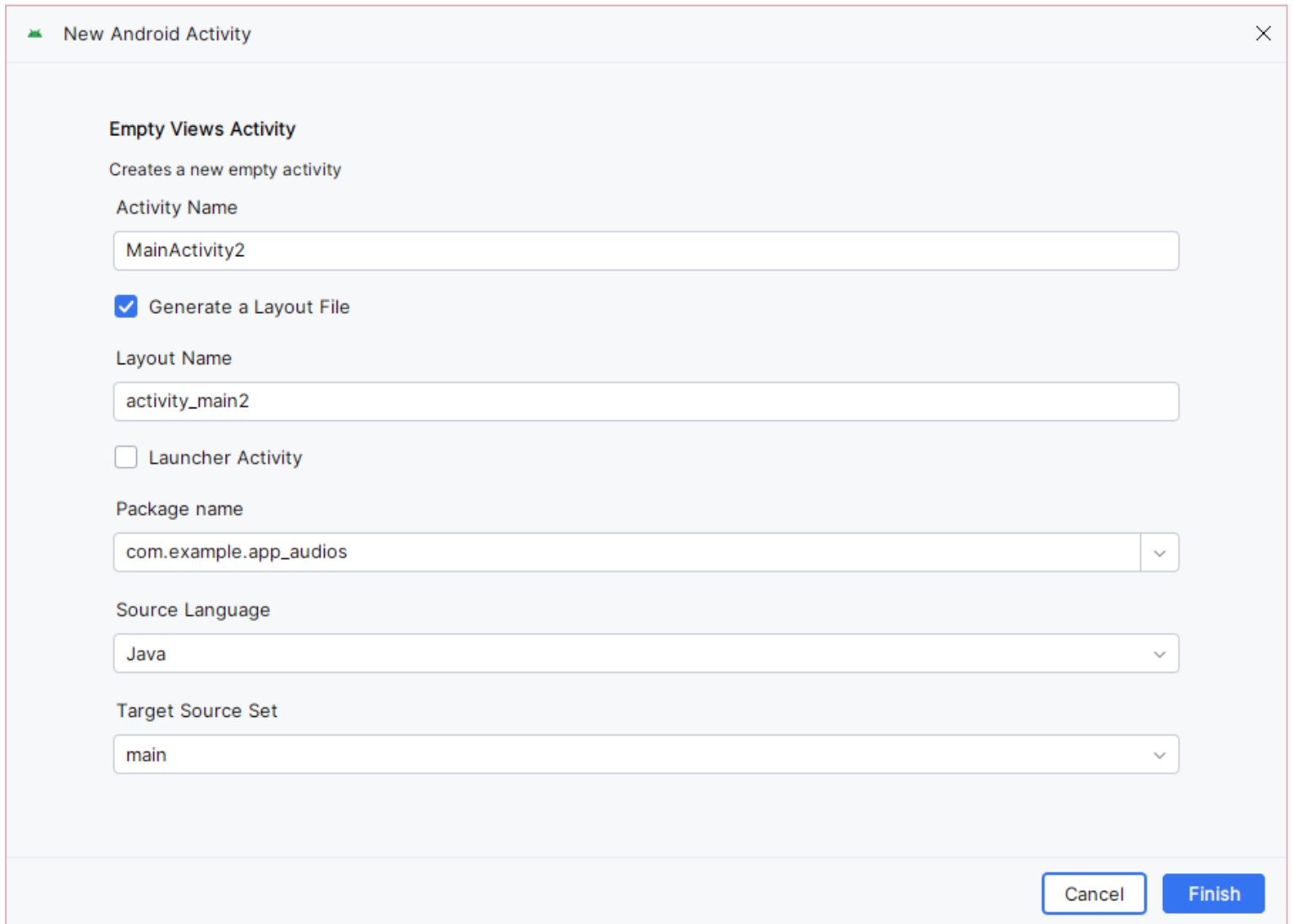
Aprendimos a crear una aplicación básica en Android Studio, utilizando el lenguaje Java y el sistema de diseño con Layout. Durante el desarrollo, configuramos el entorno, creamos el archivo XML para la interfaz y utilizamos componentes como TextView, Button y otras vistas que nos permitieron realizar acciones. También exploramos cómo funciona la actividad principal o MainActivity, que es la encargada de controlar lo que pasa cuando el usuario interactúa con la pantalla. Aprendimos a solucionar errores comunes, como los problemas para ejecutar la app con el botón Run y cómo elegir correctamente la actividad desde Run/Debug Configuration. Además, comprendimos la importancia del emulador y dispositivo de como el mismo android puede ayudarte hacer las ejecucaiones para probar nuestro proyecto. Este trabajo nos ayudó a conocer mejor cómo se inicia una app desde cero, conectando diseño, código y ejecución, y fue una experiencia útil para entender el funcionamiento básico de Android.

## **Justificación**

**Las ventajas de las aplicaciones móviles para empresas son muchas, entre ellas la mejora de la productividad, la calidad de la producción y la comunicación entre los mandos intermedios y los responsables, entre otras. Además, los mejores diseños de aplicaciones móviles son aquellos que cuidan los pequeños detalles para hacer de la aplicación algo realmente atractivo y fácil de manejar. En este sentido, es importante trabajar la línea gráfica para que sea apta para todos los públicos.**

**El diseño de aplicaciones móviles es un proceso crucial para el éxito de cualquier app. Es valioso tener en cuenta los diferentes tipos de diseño y las ventajas y desventajas de cada uno para poder adaptarse a las necesidades específicas de cada proyecto. Además, el diseño debe enfocarse en ofrecer la mejor experiencia de usuario posible y en mantener coherencia con la imagen corporativa de la empresa. Otro tipo de diseño de aplicaciones.**

## ○ Interfaz



**New Android Activity**

**Empty Views Activity**  
Creates a new empty activity

Activity Name  
MainActivity2

☒ Generate a Layout File

Layout Name  
activity\_main2

☐ Launcher Activity

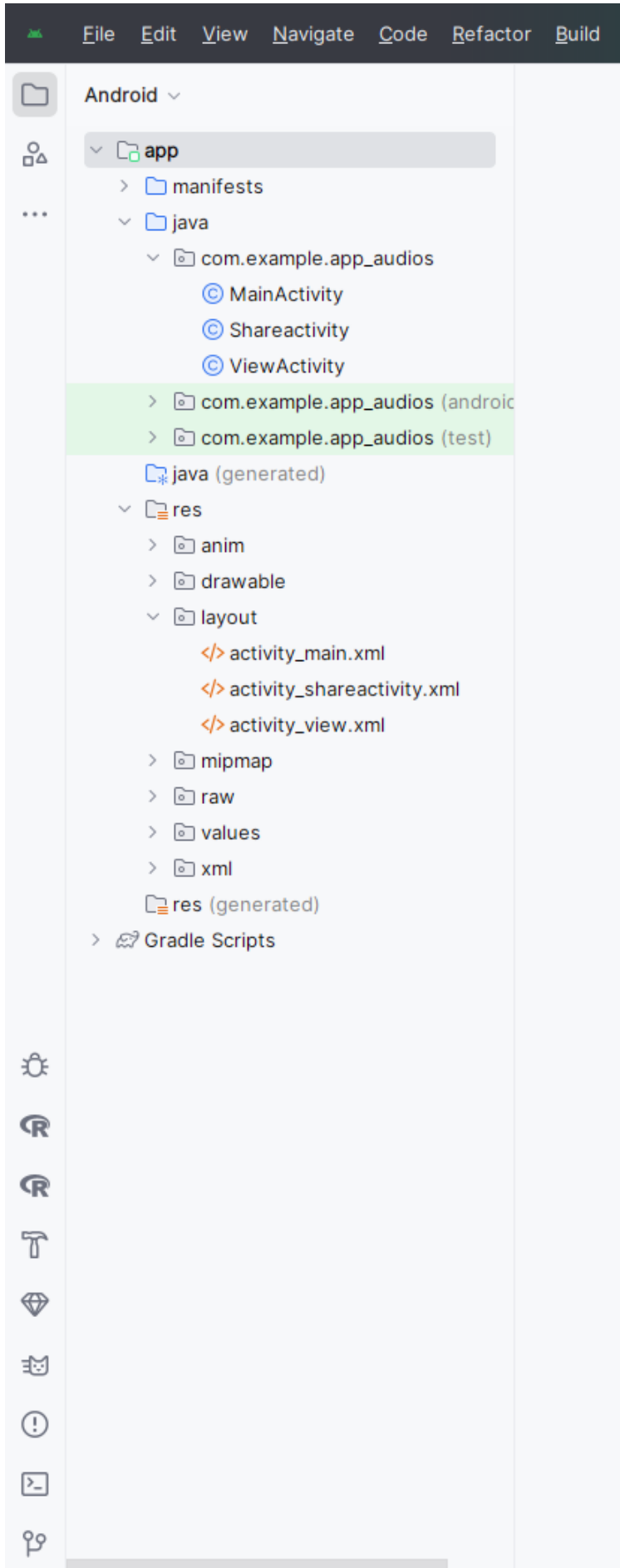
Package name  
com.example.app\_audios

Source Language  
Java

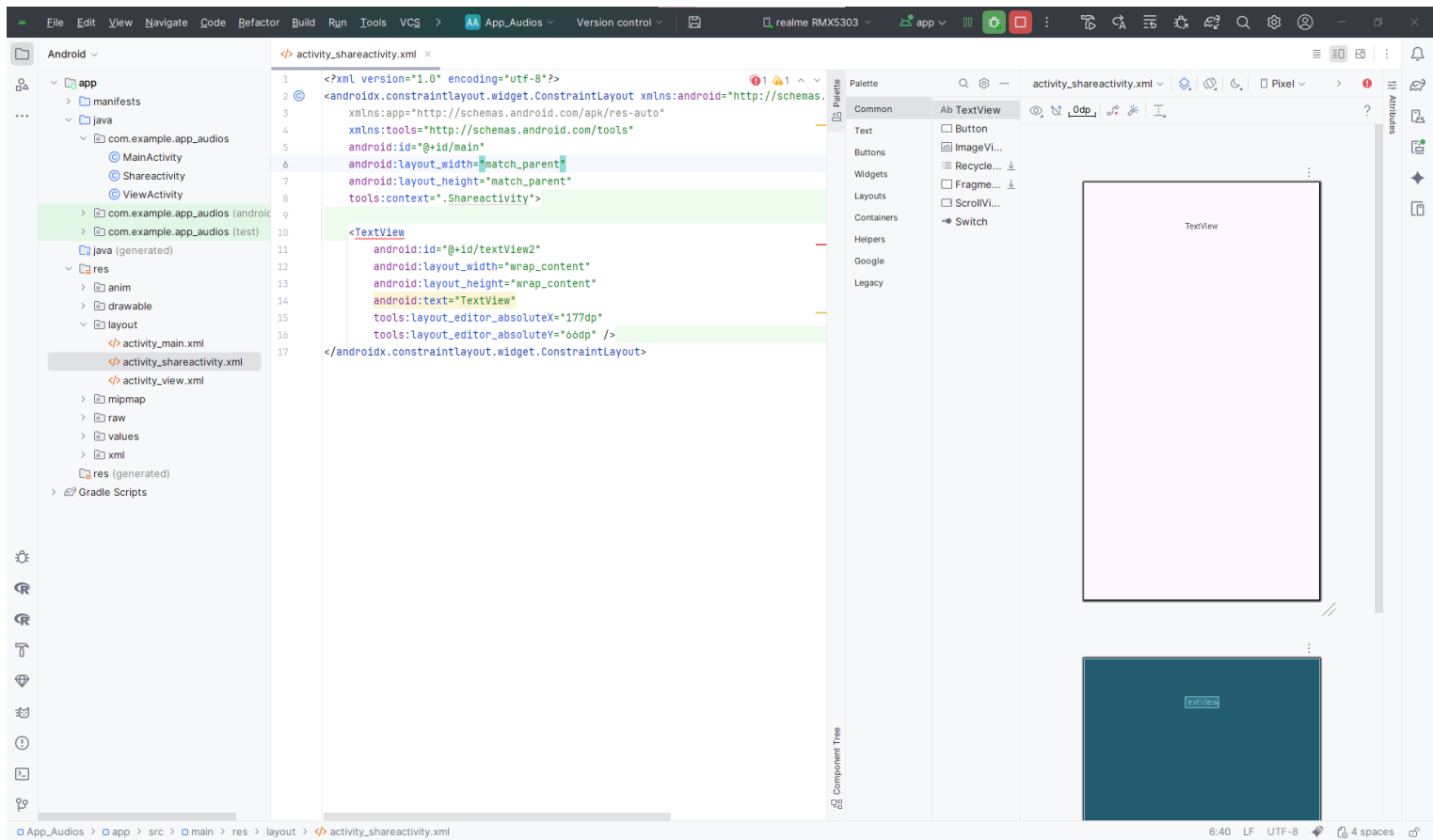
Target Source Set  
main

Cancel Finish

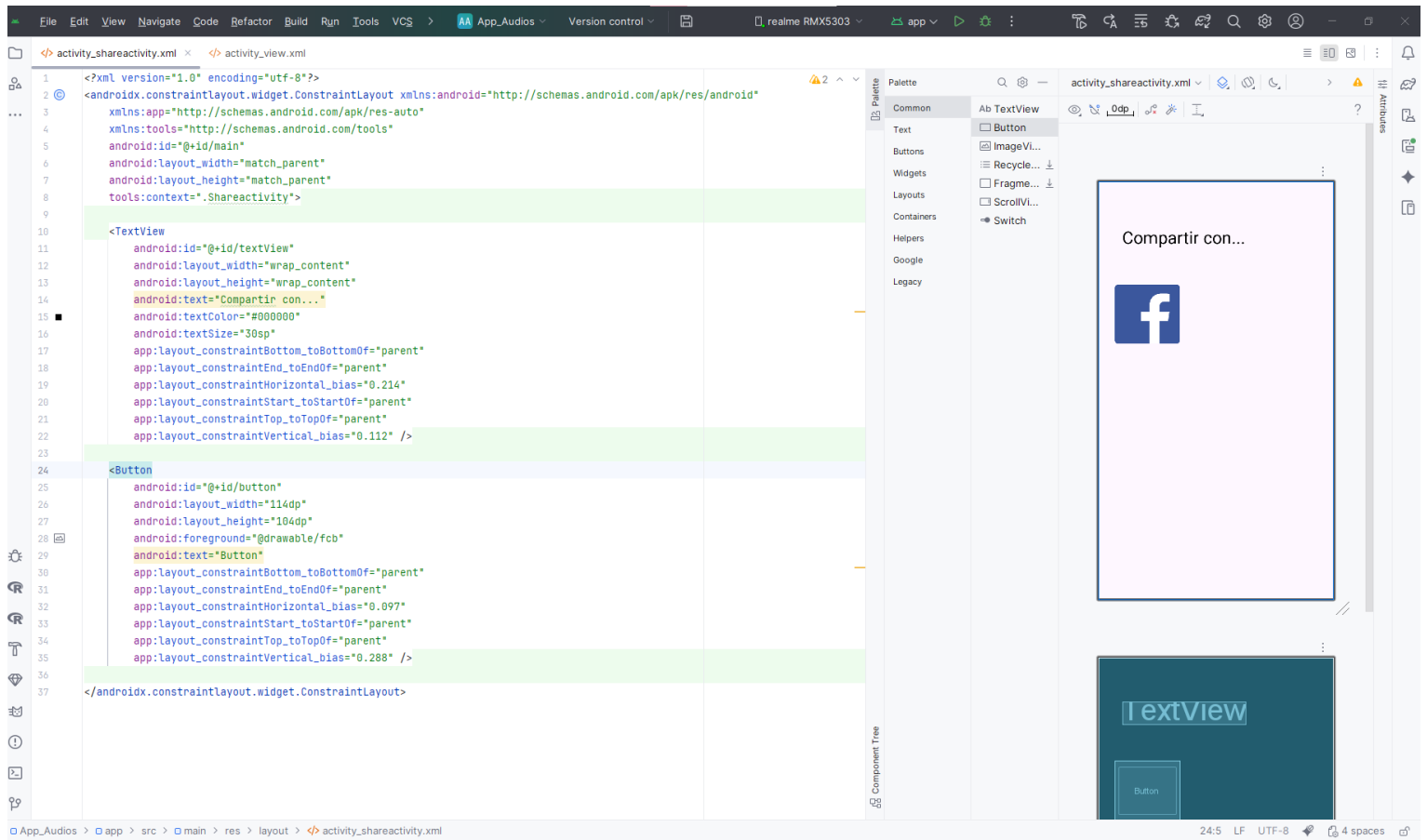
Crearemos una nueva actividad, en java, que se llamara Shareactivity.



**Lado Izquierdo:** Se muestra otro detalle en la imagen de java, Shareactivity agregado y en layout activity\_view.xml.

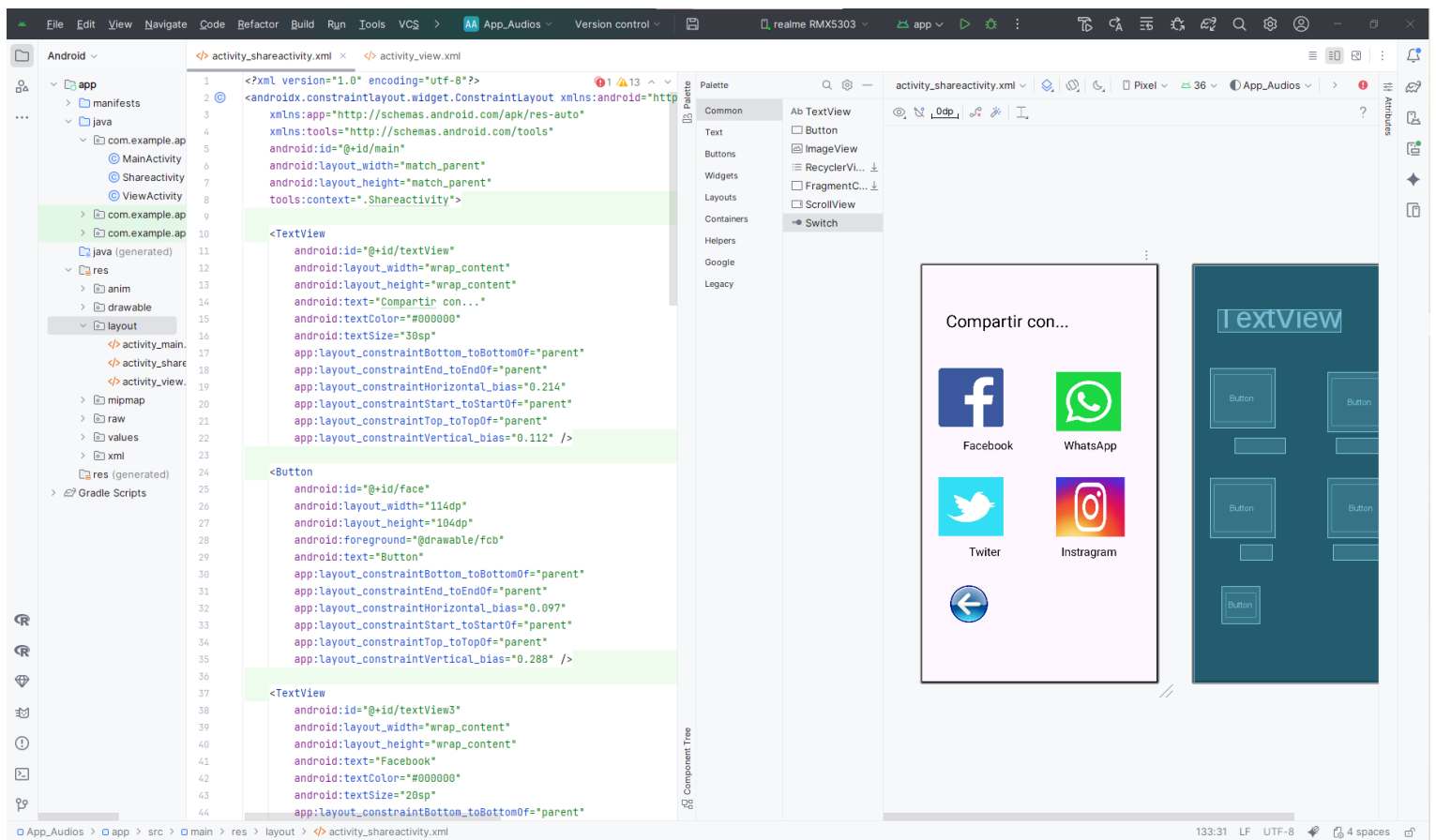


**Lado de Enmedio:** Se abre la nueva ventana de `activity_shareactivity.xml`, donde muestra los códigos del Text view del cuadro de lado derecho, donde comenzaremos agregar más detalles.



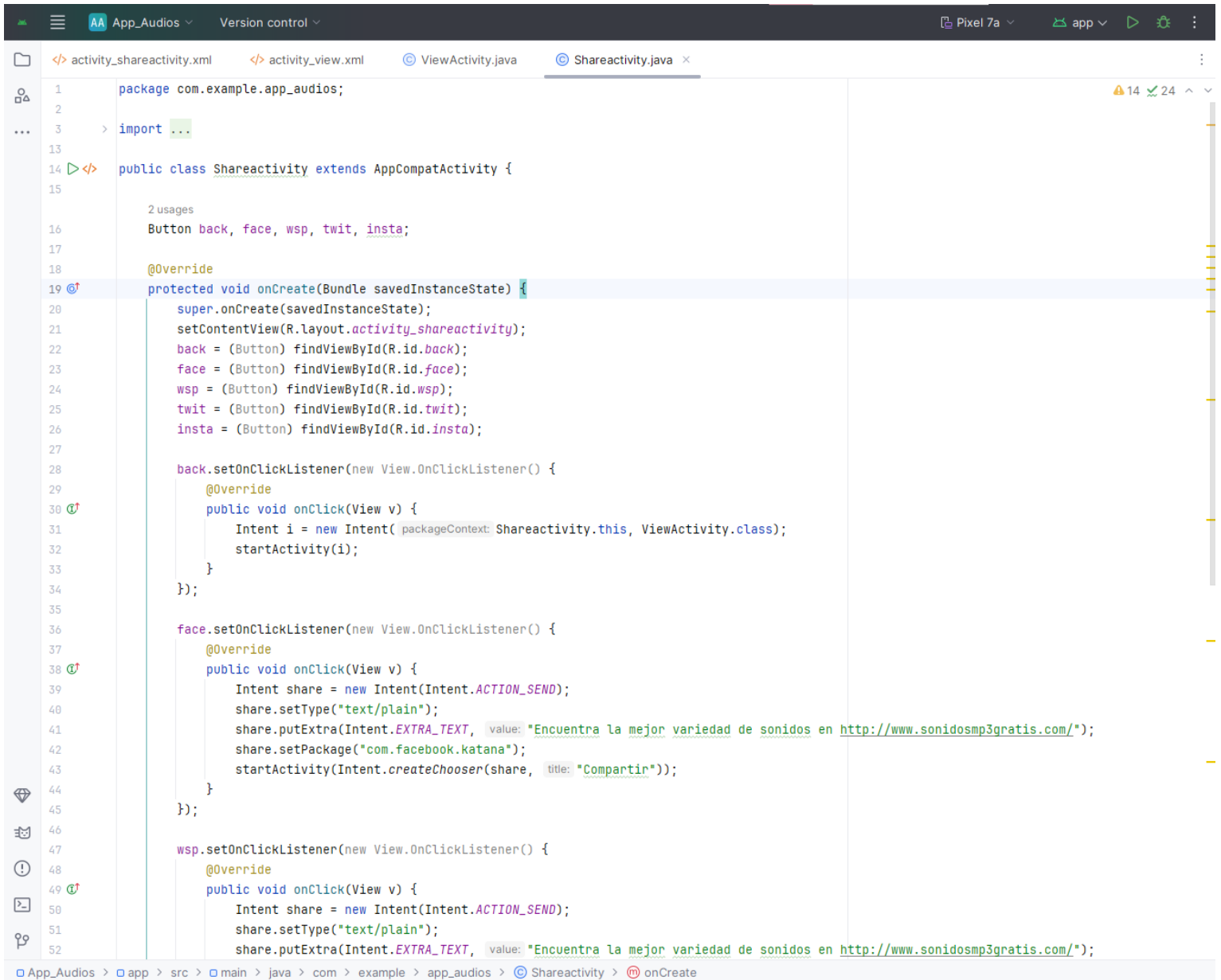
**Aquí vemos donde agregaremos Button con la imagen que tenemos en drawable, de Facebook y agregaremos los códigos de los datos.**





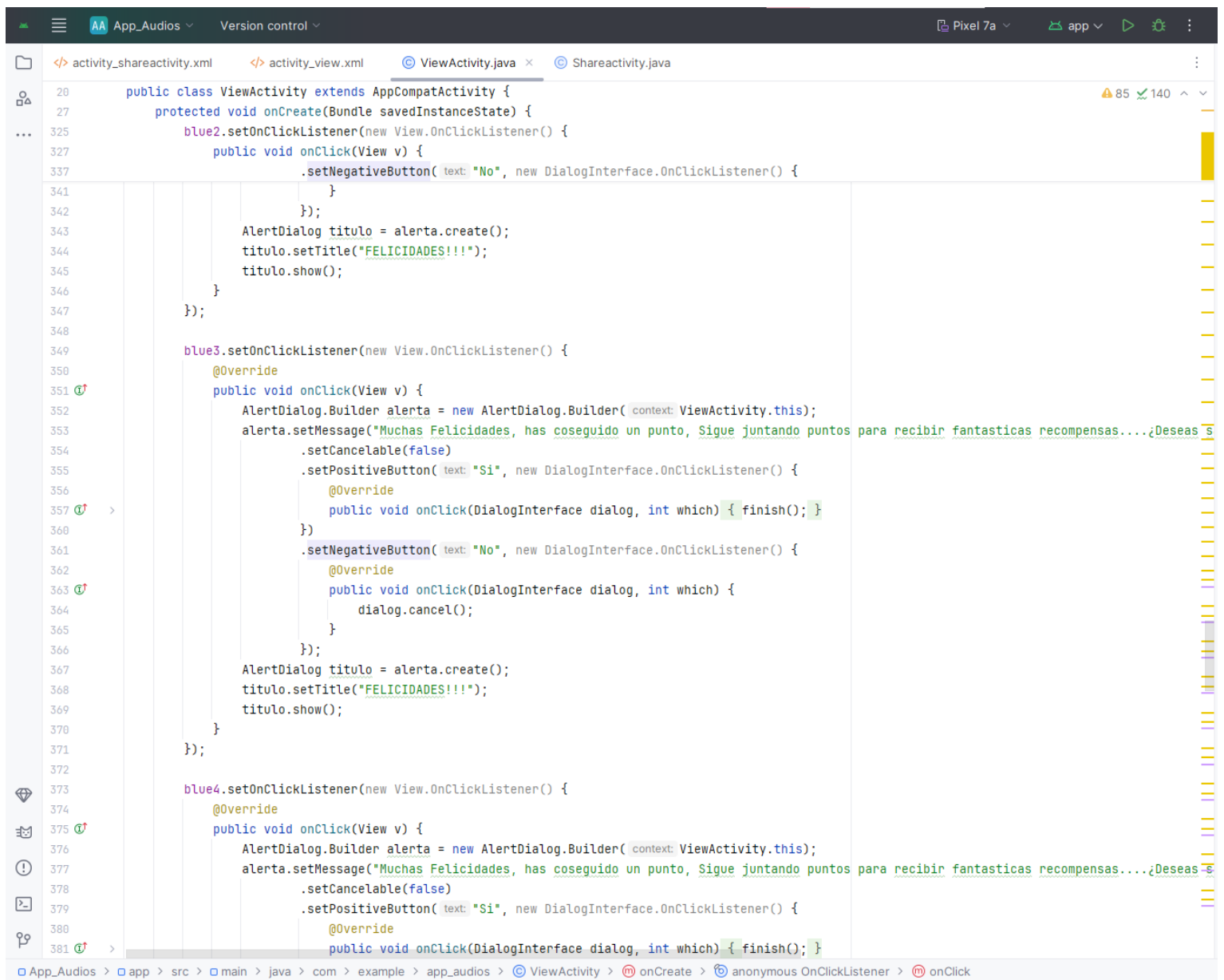
**Nos muestra ya mas a detalle el termino que se vera reflejado en nuestra app , como Facebook , WhatsApp, Twitter y Instagram, con su nombre abajo, y en la parte de arriba muestra donde lleva cuando se comparte.**

## ○ Codificación



```
1 package com.example.app_audios;
2
3 import ...
4
5
6
7
8
9
10
11
12
13
14 public class Shareactivity extends AppCompatActivity {
15
16     2 usages
17     Button back, face, wsp, twit, insta;
18
19     @Override
20     protected void onCreate(Bundle savedInstanceState) {
21         super.onCreate(savedInstanceState);
22         setContentView(R.layout.activity_shareactivity);
23         back = (Button) findViewById(R.id.back);
24         face = (Button) findViewById(R.id.face);
25         wsp = (Button) findViewById(R.id.wsp);
26         twit = (Button) findViewById(R.id.twit);
27         insta = (Button) findViewById(R.id.insta);
28
29         back.setOnClickListener(new View.OnClickListener() {
30             @Override
31             public void onClick(View v) {
32                 Intent i = new Intent(packageContext: Shareactivity.this, ViewActivity.class);
33                 startActivity(i);
34             }
35         });
36
37         face.setOnClickListener(new View.OnClickListener() {
38             @Override
39             public void onClick(View v) {
40                 Intent share = new Intent(Intent.ACTION_SEND);
41                 share.setType("text/plain");
42                 share.putExtra(Intent.EXTRA_TEXT, value: "Encuentra la mejor variedad de sonidos en http://www.sonidosmp3gratis.com/");
43                 share.setPackage("com.facebook.katana");
44                 startActivity(Intent.createChooser(share, title: "Compartir"));
45             }
46         });
47
48         wsp.setOnClickListener(new View.OnClickListener() {
49             @Override
50             public void onClick(View v) {
51                 Intent share = new Intent(Intent.ACTION_SEND);
52                 share.setType("text/plain");
53                 share.putExtra(Intent.EXTRA_TEXT, value: "Encuentra la mejor variedad de sonidos en http://www.sonidosmp3gratis.com/");
54             }
55         });
56     }
57 }
```

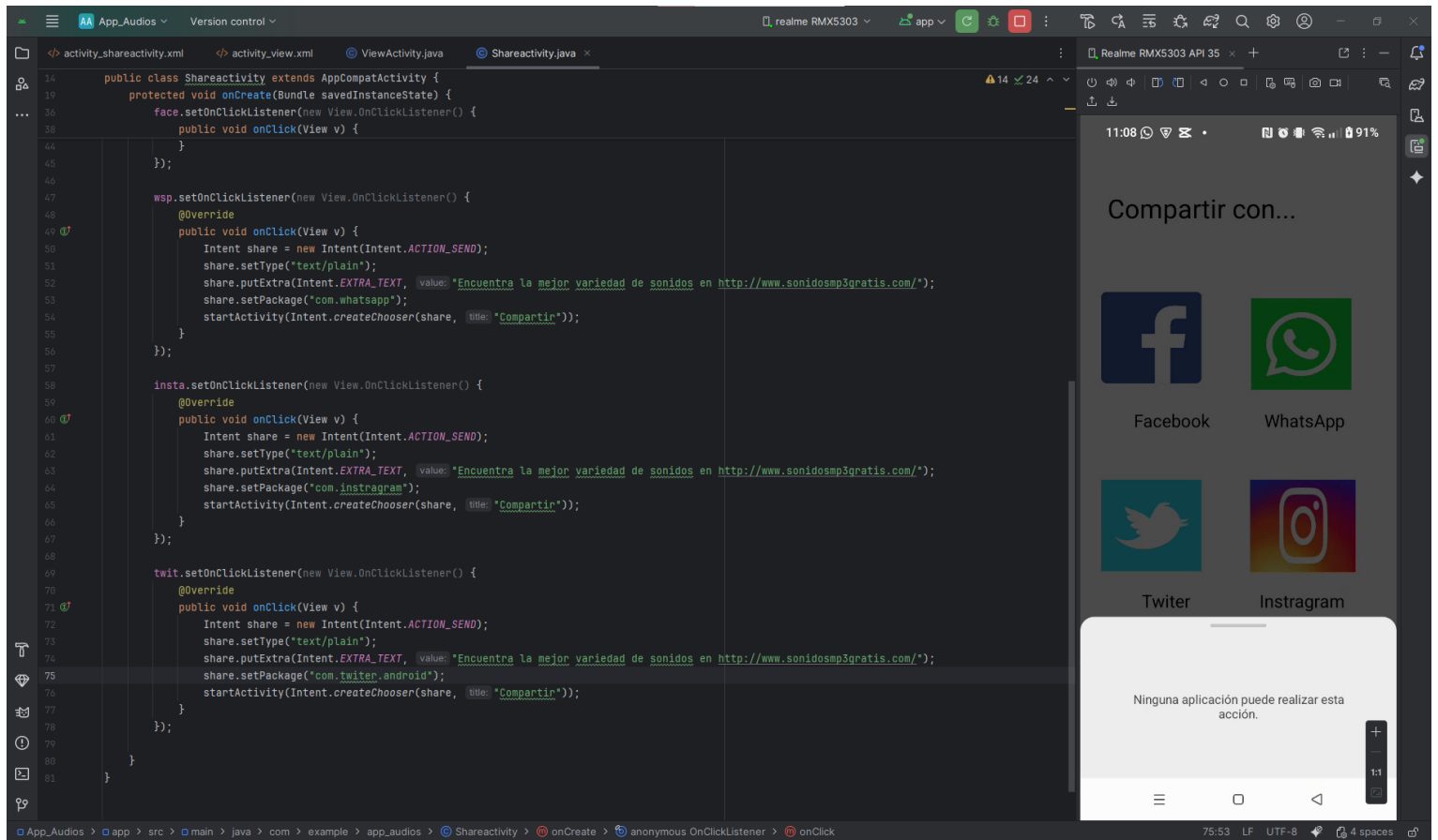
Aquí vemos las funciones que tendrá cada código de los sitios web al momento que demos clic y esta tenga una acción al momento de que se comparta.



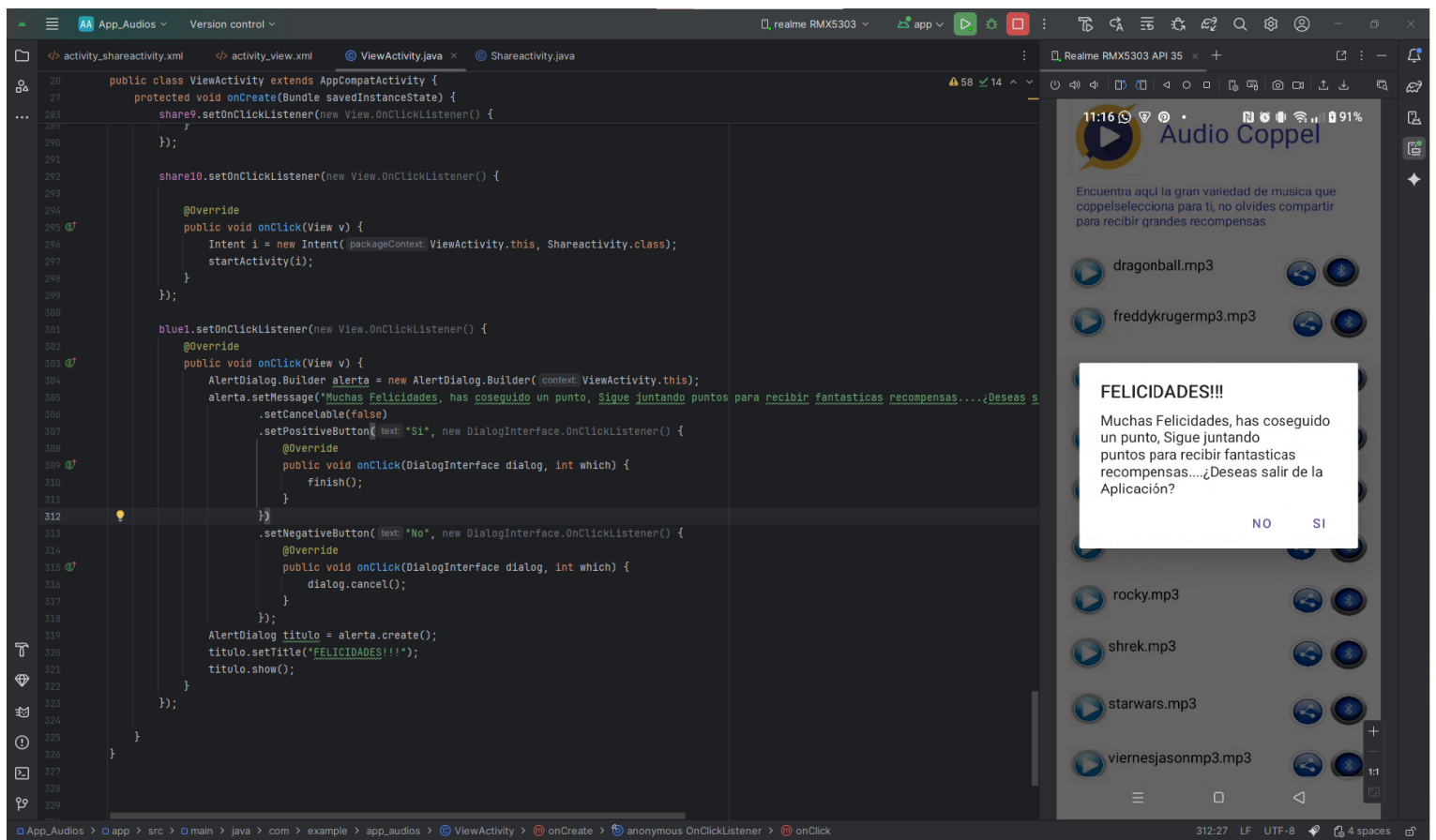
```
20 public class ViewActivity extends AppCompatActivity {
21     protected void onCreate(Bundle savedInstanceState) {
325         blue2.setOnClickListener(new View.OnClickListener() {
327             public void onClick(View v) {
337                 .setNegativeButton( text: "No", new DialogInterface.OnClickListener() {
341                     }
342                 });
343                 AlertDialog titulo = alerta.create();
344                 titulo.setTitle("FELICIDADES!!!");
345                 titulo.show();
346             }
347         });
348
349         blue3.setOnClickListener(new View.OnClickListener() {
350             @Override
351             public void onClick(View v) {
352                 AlertDialog.Builder alerta = new AlertDialog.Builder( context: ViewActivity.this);
353                 alerta.setMessage("Muchas Felicidades, has coseguido un punto, Sigue juntando puntos para recibir fantasticas recompensas....¿Deseas S
354                 .setCancelable(false)
355                 .setPositiveButton( text: "Si", new DialogInterface.OnClickListener() {
356                     @Override
357                     public void onClick(DialogInterface dialog, int which) { finish(); }
358                 })
359                 .setNegativeButton( text: "No", new DialogInterface.OnClickListener() {
360                     @Override
361                     public void onClick(DialogInterface dialog, int which) {
362                         dialog.cancel();
363                     }
364                 });
365                 AlertDialog titulo = alerta.create();
366                 titulo.setTitle("FELICIDADES!!!");
367                 titulo.show();
368             }
369         });
370
371         blue4.setOnClickListener(new View.OnClickListener() {
372             @Override
373             public void onClick(View v) {
374                 AlertDialog.Builder alerta = new AlertDialog.Builder( context: ViewActivity.this);
375                 alerta.setMessage("Muchas Felicidades, has coseguido un punto, Sigue juntando puntos para recibir fantasticas recompensas....¿Deseas S
376                 .setCancelable(false)
377                 .setPositiveButton( text: "Si", new DialogInterface.OnClickListener() {
378                     @Override
379                     public void onClick(DialogInterface dialog, int which) { finish(); }
380                 })
381                 .setNegativeButton( text: "No", new DialogInterface.OnClickListener() {
382                     @Override
383                     public void onClick(DialogInterface dialog, int which) {
384                         dialog.cancel();
385                     }
386                 });
387                 AlertDialog titulo = alerta.create();
388                 titulo.setTitle("FELICIDADES!!!");
389                 titulo.show();
390             }
391         });
392     }
393 }
```

Aquí nos muestra que cada blue tiene la misma función ya que fueron diez en total y al momento que el código muestra la acción de dar un mensaje.

## ○ Prueba de la aplicación



Vemos aquí un mensaje al dar click en uno de los sitios que nos muestra este mensaje ya que no tiene ninguna acción.



Donde podemos apreciar que al darle click blue el mensaje donde muestra su acción, si salimos de la aplicación si o no.

## Conclusión

Considera las diferentes relaciones de aspecto, clases de tamaño y resoluciones que los usuarios pueden encontrar. Verifica que tu app proporcione una buena experiencia del usuario tanto en orientación horizontal como vertical, así como en diferentes tamaños de pantalla y factores de forma.

Para obtener más información, consulta la guía para adaptar tu diseño y los diseños canónicos.

Respetar las zonas seguras del dispositivo, que incluyen partes de la IU, como cortes de pantalla, inserciones de borde a borde, pantallas de borde, teclados de software y barras del sistema.

Proporciona un diseño flexible para que los usuarios interactúen con el teclado. Advertencia:

Ten cuidado cuando cubras contenido con el teclado.

Mantén las interacciones esenciales, como la navegación principal, en un área de la pantalla a la que se pueda llegar. Los botones de acción flotantes (BAF) proporcionan un punto de interacción destacado y accesible. Usa la contención para agrupar contenido.

## REFERENCIA:

Gianotti, R. (s. f.). *Diseño de apps | blog | Agencia de marketing digital Dos setenta* |. Agencia de Marketing Digital Dos Setenta. <https://dossetenta.com/los-diferentes-tipos-de-diseno-de-apps-y-cual-es-el-mejor-para-tu-empresa/#:~:text=En%20otras%20palabras%2C%20el%20dise%C3%B1o,una%20experiencia%20de%20usuario%20agradable>.

*Conceptos básicos de diseño*. (s. f.). Android Developers.

<https://developer.android.com/design/ui/mobile/guides/layout-and-content/layout-basics?hl=es-419>

## Link Zip:

[https://github.com/casandraortiz31/actividadesCasandranuevo/blob/9c160c4ef028e409903b186b46d87b09d6bda647/Casandra%20Ortiz\\_act3\\_DAM2.zip](https://github.com/casandraortiz31/actividadesCasandranuevo/blob/9c160c4ef028e409903b186b46d87b09d6bda647/Casandra%20Ortiz_act3_DAM2.zip)