

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

Desarrollo de Aplicaciones Móviles II

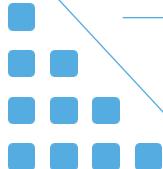
Ingeniería en Desarrollo de Software



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

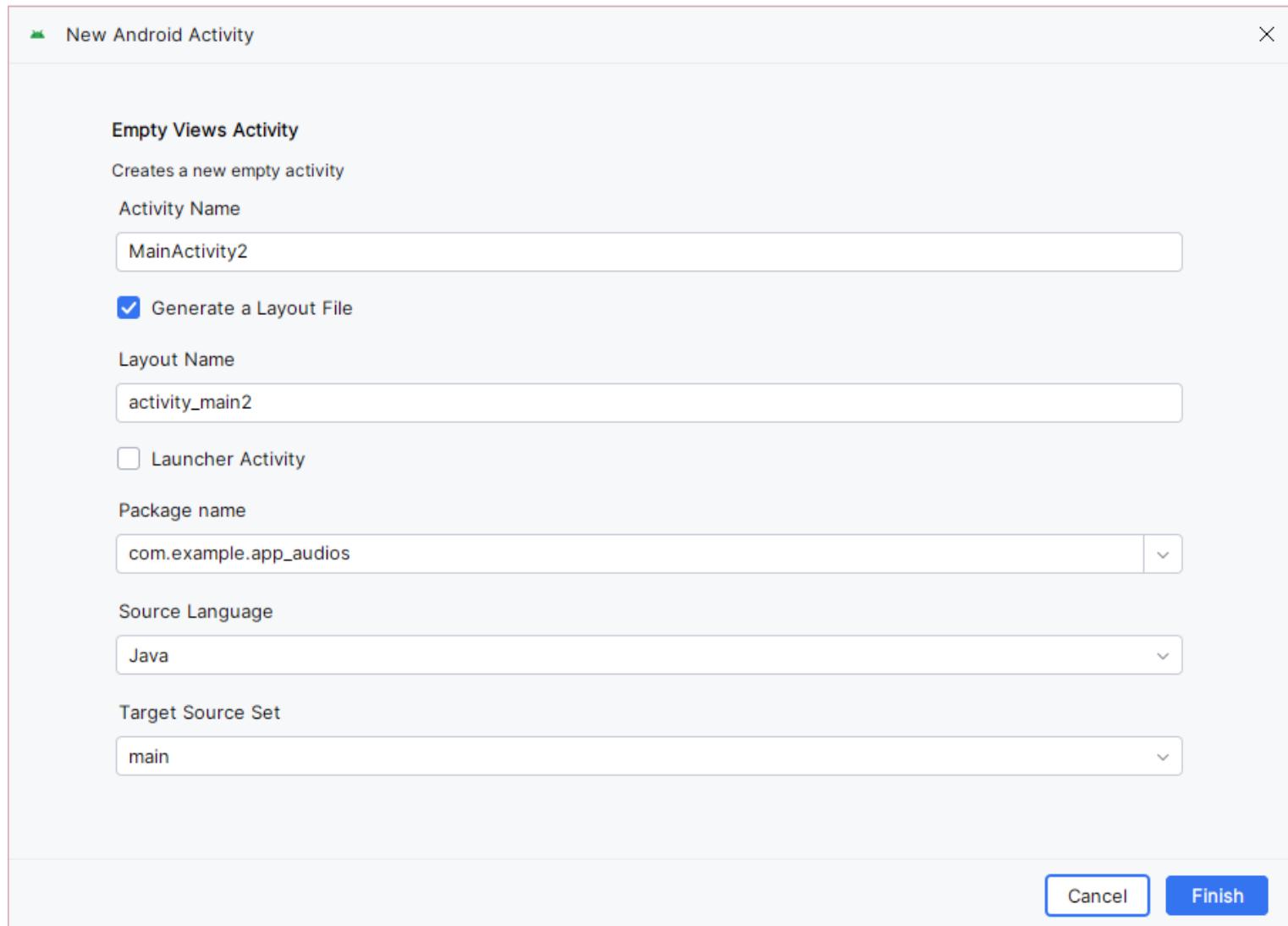
Aprendimos a crear una aplicación básica en Android Studio, utilizando el lenguaje Java y el sistema de diseño con Layout. Durante el desarrollo, configuramos el entorno, creamos el archivo XML para la interfaz y utilizamos componentes como TextView, Button y otras vistas que nos permitieron realizar 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 ejecuciones 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

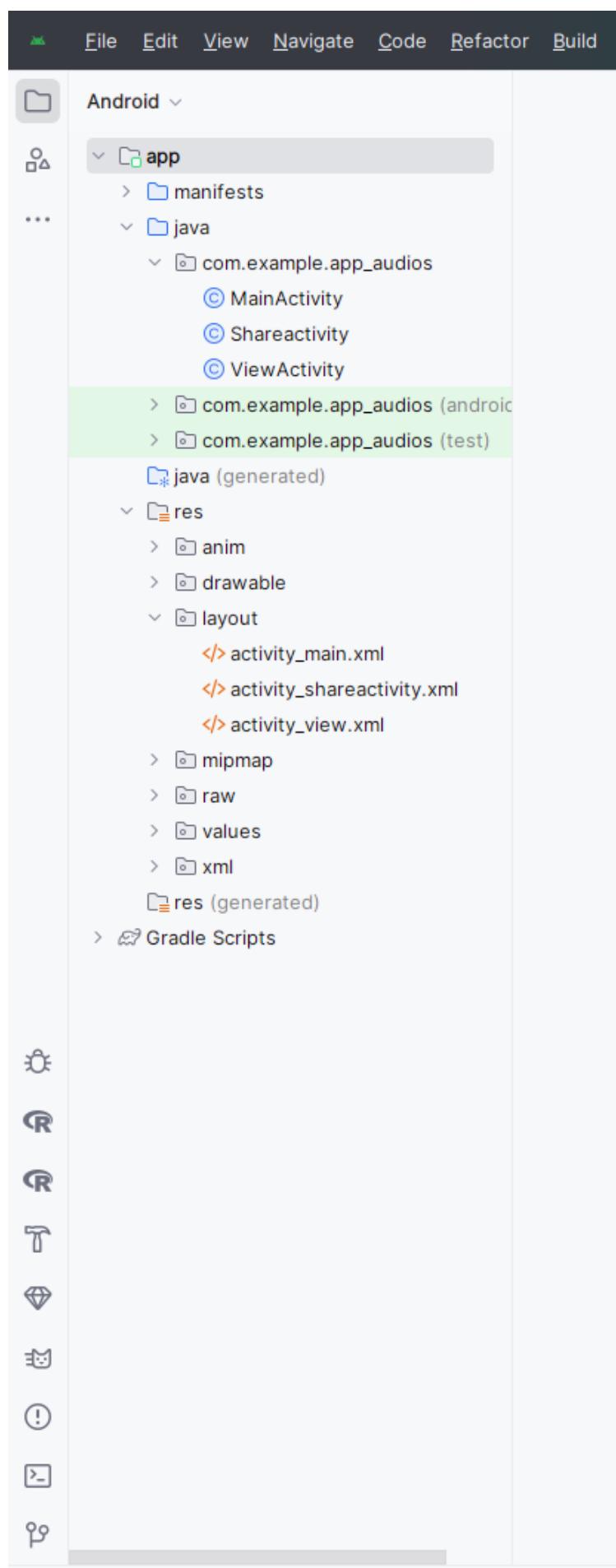
Las ventajas de las aplicaciones móviles para empresas son muchas, entre ellas la mejora de la productividad, la calidad de la producción y la comunicación entre los mandos intermedios y los responsables, entre otras. Además, los mejores diseños de aplicaciones móviles son aquellos que cuidan los pequeños 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



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.

The screenshot shows the Android Studio interface. On the left is the Project Structure sidebar with the file tree. The main area displays the XML code for `activity_shareactivity.xml`. The code defines a `ConstraintLayout` containing a `TextView` with specific attributes like `id`, `width`, and `height`. To the right of the code editor is a preview window showing a single `TextView` component. Below the preview is a component tree panel. The bottom status bar shows the file path and some system information.

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:id="@+id/main"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".Shareactivity">

    <TextView
        android:id="@+id/textView2"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="TextView"
        tools:layout_editor_absoluteX="177dp"
        tools:layout_editor_absoluteY="66dp" />

</androidx.constraintlayout.widget.ConstraintLayout>
```

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.

The screenshot shows the Android Studio interface with the XML code for `activity_shareactivity.xml` on the left and the layout preview on the right.

XML Code:

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/tools"
    android:id="@+id/main"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".Shareactivity">

    <TextView
        android:id="@+id/textView"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Compartir con..."
        android:textColor="#000000"
        android:textSize="30sp"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.214"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.112" />

    <Button
        android:id="@+id/button"
        android:layout_width="114dp"
        android:layout_height="104dp"
        android:foreground="@drawable/fcb"
        android:text="Button"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.097"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.288" />

</androidx.constraintlayout.widget.ConstraintLayout>
```

Layout Preview:

The layout preview shows a blue rectangular container labeled "Compartir con...". Inside this container is a blue square button with a white "f" icon, representing the Facebook logo. Below the button is a blue rectangular area labeled "TextView".

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

The screenshot shows the Android Studio interface. On the left is the Project Navigational Bar. In the center-left is the code editor with the XML file `activity_shareactivity.xml`. The code defines a layout with a constraint layout containing several text views and buttons for sharing to various platforms. On the right is the Layout Preview window, which displays a pink-themed share interface with icons for Facebook, WhatsApp, Twitter, and Instagram, along with a back arrow button.

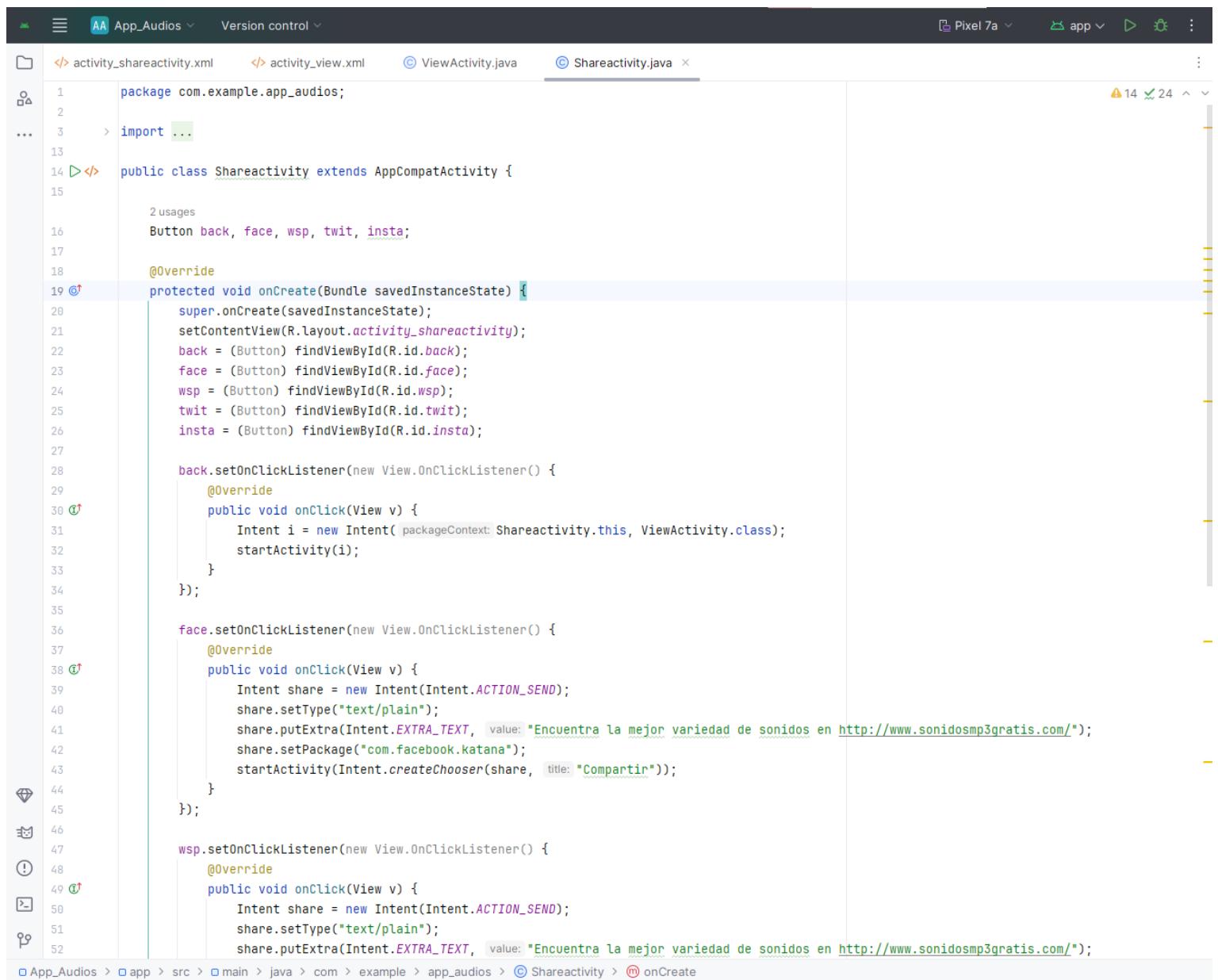
```

<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:id="@+id/main"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".Shareactivity">
    <textView
        android:id="@+id/textView"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Compartir con..."
        android:textColor="#000000"
        android:textSize="30sp"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.214"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.112" />
    <Button
        android:id="@+id/face"
        android:layout_width="114dp"
        android:layout_height="104dp"
        android:foreground="@drawable/fcb"
        android:text="Button"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.097"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.288" />
    <textView
        android:id="@+id/textView3"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Facebook"
        android:textColor="#000000"
        android:textSize="20sp"
        app:layout_constraintBottom_toBottomOf="parent" />

```

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



The screenshot shows the Android Studio interface with the code editor open to the Shareactivity.java file. The code is written in Java and defines a class that extends AppCompatActivity. It contains methods for button click listeners and intent creation for sharing content.

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

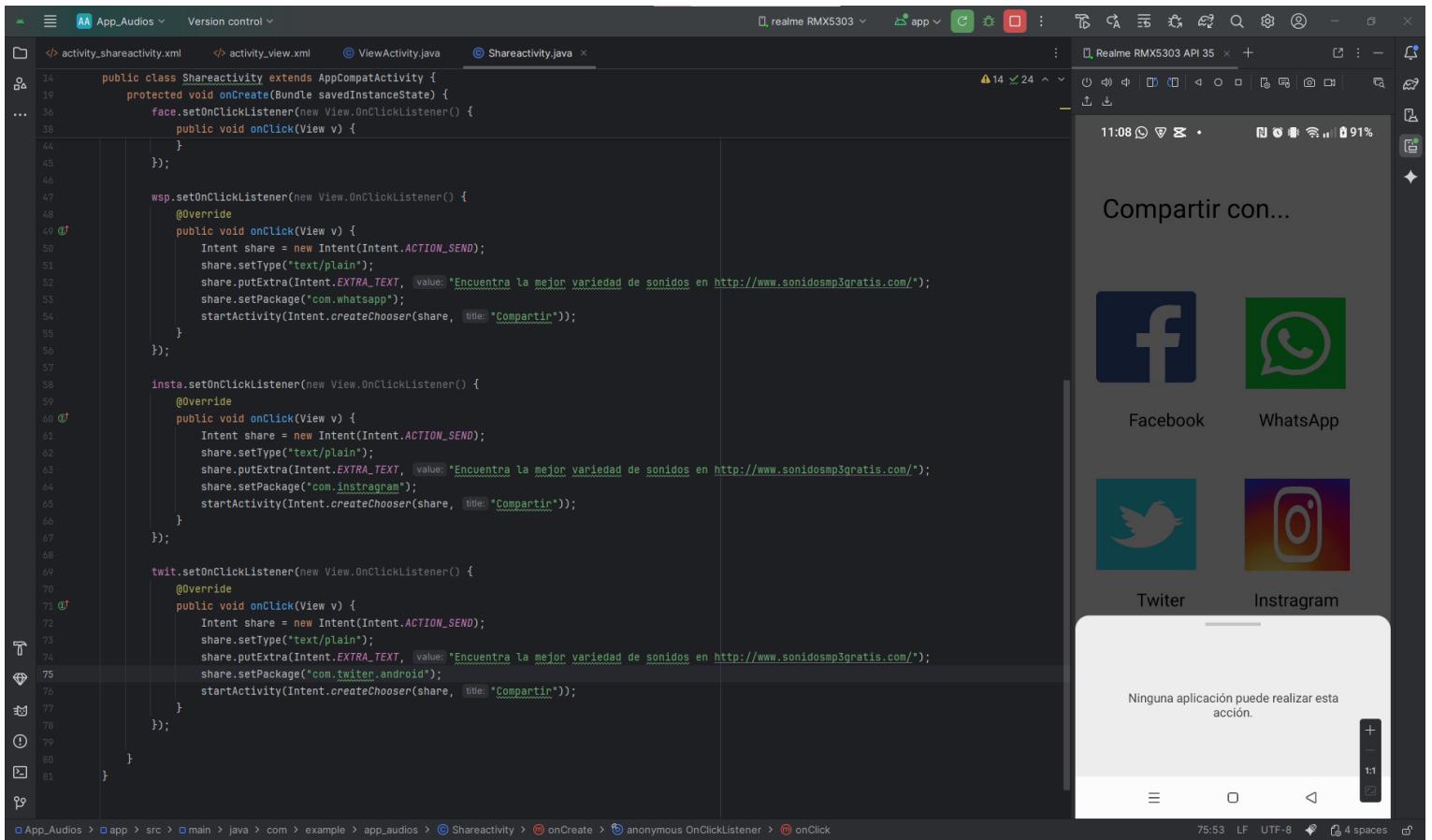
The screenshot shows the Android Studio interface with the following details:

- Project Structure:** The project is named "App_Audios".
- File List:** The current file is "ViewActivity.java", which is open. Other files listed include "activity_shareactivity.xml", "activity_view.xml", and "Shareactivity.java".
- Code Editor:** The code for "ViewActivity.java" is displayed. It contains logic for handling button clicks on a view. Specifically, it creates an AlertDialog titled "FELICIDADES!!!" with two buttons: "Si" (positive) and "No" (negative). The "Si" button triggers a finish() call, while the "No" button triggers a cancel() call.
- Toolbars and Status Bar:** The top bar shows "Pixel 7a" as the device, "app" as the configuration, and various icons for file operations. The status bar on the right shows "85 140" with a downward arrow.
- Side Panels:** The left side has standard Android Studio navigation icons for file, recent projects, and search.

```
20     public class ViewActivity extends AppCompatActivity {
21         protected void onCreate(Bundle savedInstanceState) {
22             blue2.setOnClickListener(new View.OnClickListener() {
23                 public void onClick(View v) {
24                     .setNegativeButton(text: "No", new DialogInterface.OnClickListener() {
25                         ...
26                     });
27                     AlertDialog titulo = alerta.create();
28                     titulo.setTitle("FELICIDADES!!!");
29                     titulo.show();
30                 }
31             });
32             blue3.setOnClickListener(new View.OnClickListener() {
33                 @Override
34                 public void onClick(View v) {
35                     AlertDialog.Builder alerta = new AlertDialog.Builder(context: ViewActivity.this);
36                     alerta.setMessage("Muchas Felicidades, has cosegado un punto, Sigue juntando puntos para recibir fantaticas recompensas....¿Deseas s");
37                     .setCancelable(false)
38                     .setPositiveButton(text: "Si", new DialogInterface.OnClickListener() {
39                         @Override
40                         public void onClick(DialogInterface dialog, int which) { finish(); }
41                     })
42                     .setNegativeButton(text: "No", new DialogInterface.OnClickListener() {
43                         @Override
44                         public void onClick(DialogInterface dialog, int which) {
45                             dialog.cancel();
46                         }
47                     });
48                     AlertDialog titulo = alerta.create();
49                     titulo.setTitle("FELICIDADES!!!");
50                     titulo.show();
51                 }
52             });
53             blue4.setOnClickListener(new View.OnClickListener() {
54                 @Override
55                 public void onClick(View v) {
56                     AlertDialog.Builder alerta = new AlertDialog.Builder(context: ViewActivity.this);
57                     alerta.setMessage("Muchas Felicidades, has cosegado un punto, Sigue juntando puntos para recibir fantaticas recompensas....¿Deseas s");
58                     .setCancelable(false)
59                     .setPositiveButton(text: "Si", new DialogInterface.OnClickListener() {
60                         @Override
61                         public void onClick(DialogInterface dialog, int which) { finish(); }
62                     });
63                 }
64             });
65         }
66     }
```

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



The screenshot shows the Android Studio interface with the code editor on the left and the emulator window on the right. The code editor displays the `Shareactivity.java` file, which contains Java code for handling sharing intents. The emulator window shows a `realme RMX5303 API 35` device with a `Compartir con...` dialog open. The dialog lists four sharing options: Facebook, WhatsApp, Twitter, and Instagram. Below these options, a message says `Ninguna aplicación puede realizar esta acción.` (No application can perform this action.).

```
14 public class Shareactivity extends AppCompatActivity {
15     protected void onCreate(Bundle savedInstanceState) {
16         face.setOnClickListener(new View.OnClickListener() {
17             public void onClick(View v) {
18                 ...
19             }
20         });
21
22         wsp.setOnClickListener(new View.OnClickListener() {
23             @Override
24             public void onClick(View v) {
25                 Intent share = new Intent(Intent.ACTION_SEND);
26                 share.setType("text/plain");
27                 share.putExtra(Intent.EXTRA_TEXT, "Encuentra la mejor variedad de sonidos en http://www.sonidosmp3gratis.com/");
28                 share.setPackage("com.whatsapp");
29                 startActivity(Intent.createChooser(share, "Compartir"));
30             }
31         });
32
33         insta.setOnClickListener(new View.OnClickListener() {
34             @Override
35             public void onClick(View v) {
36                 Intent share = new Intent(Intent.ACTION_SEND);
37                 share.setType("text/plain");
38                 share.putExtra(Intent.EXTRA_TEXT, "Encuentra la mejor variedad de sonidos en http://www.sonidosmp3gratis.com/");
39                 share.setPackage("com.instagram");
40                 startActivity(Intent.createChooser(share, "Compartir"));
41             }
42         });
43
44         twit.setOnClickListener(new View.OnClickListener() {
45             @Override
46             public void onClick(View v) {
47                 Intent share = new Intent(Intent.ACTION_SEND);
48                 share.setType("text/plain");
49                 share.putExtra(Intent.EXTRA_TEXT, "Encuentra la mejor variedad de sonidos en http://www.sonidosmp3gratis.com/");
50                 share.setPackage("com.twitter.android");
51                 startActivity(Intent.createChooser(share, "Compartir"));
52             }
53         });
54     }
55 }
```

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

```
public class ViewActivity extends AppCompatActivity {
    protected void onCreate(Bundle savedInstanceState) {
        share9.setOnClickListener(new View.OnClickListener() {
            ...
        });
        share10.setOnClickListener(new View.OnClickListener() {
            ...
        });
        blue1.setOnClickListener(new View.OnClickListener() {
            ...
        });
    }
}
```

11:16 91% Audio Coppel
Encuentra aqui la gran variedad de musica que coppel selecciona para ti, no olvides compartir para recibir grandes recompensas

dragonball.mp3

freddykrugermp3.mp3

FELICIDADES!!
Muchas Felicidades, has cosegado un punto, Sigue juntando puntos para recibir fantasicas recompensas....¿Deseas salir de la Aplicación?
NO SI

rocky.mp3

shrek.mp3

starwars.mp3

viernesjasonmp3.mp3

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.

Respeta 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/cassandraortiz31/actividadesCasandranuevo/blob/9c160c4ef028e409903b186b46d87b09d6bda647/Cassandra%20Ortiz_act3_DAM2.zip