

## Explicación paso a paso

### 1. Creación de Splash (pantalla de carga)

#### a) Activity\_splash.xml

Este es un archivo XML que define el diseño de la pantalla de inicio (Splash) de la aplicación.

Utilizé un CoordinatorLayout, que es un diseño especial que se utiliza para coordinar la disposición de varias vistas en la pantalla.

Dentro del CoordinatorLayout, hay un LinearLayout que contiene dos elementos: un ImageView que muestra una imagen de piedra, papel o tijeras, y un TextView que muestra el título "PIEDRA, PAPEL O TIJERAS".

Los atributos android:layout\_width y android:layout\_height se utilizan para establecer el ancho y alto de los elementos en la pantalla. También hay otros atributos, como android:src para establecer la imagen del ImageView y android:text para establecer el texto del TextView.

#### b) Splash.kt

Aquí está la implementación de la actividad Splash. La actividad Splash se muestra brevemente cuando se inicia la aplicación y luego se cierra automáticamente después de un período de tiempo.

En el código se define la duración de la actividad Splash en 3000 milisegundos (3 segundos) y se utiliza un controlador para crear un retardo antes de iniciar la actividad Login y cerrar la actividad Splash.

## 2. Creación de Login

### a) Activity\_login.xml

Este es el archivo de diseño XML para la pantalla de inicio de sesión (Login). En el archivo hice el diseño de la pantalla, principalmente con los siguientes elementos:

- CoordinatorLayout: es el contenedor principal que se utiliza para coordinar las interacciones entre diferentes vistas y componentes en una pantalla.
- LinearLayout: es un contenedor que se utiliza para organizar las vistas de la pantalla en una disposición lineal.
- TextView: muestra el título de la pantalla ("Piedra, papel o tijera").
- ImageView: muestra una imagen (en este caso, el logo del juego).
- TextInputLayout y TextInputEditText: se utilizan para recopilar el nombre del jugador a través de una entrada de texto y mandarlo a la siguiente vista.
- Button: se utiliza para iniciar el juego, y tiene un onClick atributo que hace referencia a un método onClick(), y manda a la siguiente vista.

### b) Login.kt

El archivo Login.kt controla la lógica detrás de la pantalla de inicio de sesión, asegurándose de que el usuario ingrese un nombre antes de permitir que se inicie el juego.

### 3. Creación de MainActivity

#### a) Activity\_main.xml

La primera línea indica la versión de XML y la codificación utilizada.

La siguiente línea indica el nombre completo de la clase CoordinatorLayout de la biblioteca AndroidX que se utilizará para organizar los elementos de la interfaz gráfica.

En las siguientes líneas, se definen los espacios de nombres que se utilizarán para la aplicación.

El atributo layout\_width y layout\_height están establecidos en match\_parent, lo que significa que el tamaño de la vista será igual al tamaño de su contenedor.

El atributo fitsSystemWindows se establece en true para asegurarse de que la aplicación ocupe todo el espacio disponible en la pantalla del dispositivo, incluyendo la barra de estado.

En las siguientes líneas, se definen varios elementos de la interfaz de usuario de Android, como LinearLayout, TextView, ImageView e ImageButton, junto con sus atributos para establecer su apariencia y posición en la pantalla.

Los elementos LinearLayout se utilizan para organizar otros elementos dentro de ellos, ya sea horizontal o verticalmente.

Los elementos TextView se utilizan para mostrar texto en la pantalla.

Los elementos ImageView se utilizan para mostrar imágenes en la pantalla.

Los elementos ImageButton se utilizan para permitir al usuario hacer clic en una imagen para realizar una acción determinada en la aplicación.

Los atributos id se utilizan para identificar los elementos en la interfaz de usuario y permitir su manipulación programática.

Los atributos onClick se utilizan para asociar los botones de la interfaz gráfica con los métodos en el código Java de la aplicación que se ejecutarán cuando se haga clic en los botones.

El atributo tools:context se utiliza para especificar el nombre de la actividad de la aplicación en la que se utiliza esta interfaz de usuario.

#### b) MainActivity.kt

Este es el código de la actividad principal de un juego de piedra, papel y tijera. La actividad extiende la clase AppCompatActivity y tiene algunos campos privados para almacenar información sobre el jugador, los puntos, y los datos seleccionados por el jugador y la computadora.

El método onCreate() es el primer método que se ejecuta cuando la actividad se inicia y es donde se configura la vista de la actividad utilizando el método setContentView() y se recuperan los widgets de la interfaz de usuario como TextViews, ImageViews, etc. utilizando el método findViewById(). También se

intenta obtener el nombre del jugador que fue pasado desde la actividad anterior a través del objeto Intent.

La actividad también tiene un método llamado `onClick()` que se llama cuando el usuario hace clic en uno de los botones de la interfaz de usuario. Este método utiliza una estructura de control de flujo "when" para realizar una acción específica en función del botón que fue presionado.

También hay tres métodos privados en esta actividad: `resultadoJugador()`, `aleatorioDataComputadora()` y `resultado()`. `resultadoJugador()` actualiza la imagen que representa la selección del jugador y llama al método `resultado()`, que compara la selección del jugador con la selección de la computadora para determinar el ganador. `aleatorioDataComputadora()` genera una selección aleatoria para la computadora y llama a `resultadoJugador()`.

El método `Fin()` verifica si algún jugador ha alcanzado los 10 puntos y muestra un mensaje según el ganador, y finaliza la actividad en consecuencia.