21-12-2020

Lucía Colás Vico – 1007 – Gis Marcos Sánchez Hernández – 981 – GIS Sara Rodríguez Alarcón – 781 – GII

https://github.com/srodriguezal/MarioConga

**MARIO CONGA**

Práctica 3 – Laboratorio de dispositivos móviles

**ÍNDICE**

[**INTRODUCCIÓN** 1](#_Toc59490392)

[**EVOLUCIÓN** 2](#_Toc59490393)

[**Mejoras e implementaciones** 2](#_Toc59490394)

[**DIFICULTADES ENCONTRADAS** 4](#_Toc59490395)

[**CONCLUSIONES** 5](#_Toc59490396)

[**BIBLIOGRAFÍA** 6](#_Toc59490397)

# **INTRODUCCIÓN**

En esta práctica perteneciente al tercer bloque de la asignatura de Laboratorio de dispositivos móviles se ha pedido realizar un juego en Android siguiendo la temática del clásico juego de la serpiente denominado Snake.

En dicha práctica se pedía realizar una serie de modificaciones al juego base proporcionado en la asignatura.

A continuación, se expondrá cómo ha sido la evolución del juego a lo largo del trabajo hasta su entrega, los detalles más relevantes que se han implementado y las dificultades encontradas en ciertos momentos de su realización.

# **EVOLUCIÓN**

El código proporcionado en clase consistía en un juego llamado “*Piratas de los siete mares”* cuya temática era similar a la del popular juego “*Snake”*.

En esta versión se debe coger tesoros y por cada tesoro se añade un miembro a la tripulación hasta que ya no entren más tripulantes en la pantalla, en cuyo caso habremos ganado el juego. En el caso contrario, se pierde el juego si el jugador hace chocar su barco con la tripulación.

La primera parte de la evolución del juego ha sido cambiar todos los *assets* de piratas del juego original por *assets* de otro popular juego llamado “*Mario Bros*” para dotar al juego de una nueva identidad. Se han usado diferentes elementos tanto para los personajes como para los tesoros que tienen que recoger, pero siempre manteniendo la escala original.

Posteriormente, se han introducido dos nuevas pantallas para la selección de los distintos personajes, los cuales tienen sus propios tesoros y enemigos.

Por último, se han añadido efectos de sonido en la selección de los personajes, fin del juego y durante la partida.

## **Mejoras e implementaciones**

A continuación, se expondrán las mejoras, implementaciones y cambios que se han realizado en esta práctica con detalle y comentando los aspectos más importantes de cada una de ellas:

* **Terreno de juego:** Se ha cambiado el terreno de juego por una temática basada en el césped de un jardín debido a que encajaba más con la temática de los juegos de “*Mario Bros*” con la que se quería dotar al juego. Al igual que el terreno de juego, también se han modificado las letras y números de todas las pantallas del juego por otras más coloridas que encajaran mejor con la temática escogida.
* **Personajes:** Se han cambiado los personajes originales por personajes de los juegos de “*Mario Bros*”. A cada personaje se le ha asignado unos tesoros y unos enemigos personalizados para cada uno.
* **Puntuaciones y tesoros:** Se ha dividido los tesoros en tres tipos donde cada tipo tiene asignada una puntuación. Los tesoros son personalizados para cada personaje.
* **Aparición de enemigos:** En el juego original solo se podía perder si el barco chocaba con su tripulación, pero en “*Mario Conga*” se decidió añadir un enemigo en una posición aleatoria para que, si se el jugador choca con el enemigo, pierda. Dicho enemigo irá modificando su posición según el número de tesoros que haya conseguido. En este caso también hay tres tipos de enemigos distintos para cada personaje los cuales alternarán su aparición.
* **Música:** Se ha modificado la música que llevaba el juego original y cada vez que se seleccione un personaje sonará un tono característico de este.
* **Pantalla de selección de personajes:** Se han añadido dos pantallas adicionales a las que ya se encontraban en el juego original. En dichas pantallas, el usuario puede seleccionar y confirmar la elección del personaje con el que desee jugar.
* **Seguidores:** Al igual que ocurre con los enemigos y los tesoros, los seguidores también son personalizados para cada personaje.
* **Botones:** Además de las imágenes que representan los botones, también se ha modificado la posición del botón de pausa para que no obstaculizara el campo de juego.

# **DIFICULTADES ENCONTRADAS**

Al intentar hacer un juego con muchos personajes donde cada uno tiene unos elementos específicos, se ha modificado el código de la aplicación original, debido a que estaba únicamente realizado para un personaje concreto, por lo que se ha tenido que cambiar el método de drawWorld entero. En un principio este método pintaba todo a la vez al ser un solo personaje por lo que se tomó la decisión de que pintara según el personaje escogido para evitar fallos *NullPointerException* o *AssetsNullReference.*

Para añadir puntuaciones diferentes según el premio hubo que modificar la clase **Mundo.java.** En un principio se pensó que con eso era suficiente, pero la idea no podía realizarse si no se obtenía el tipo de premio. Por ello, en vez de hacerlo individual, se decidió pasar el tipo de premio a esta clase a través de un método “*get*” de la clase **Premio.java** y de esta forma solucionar el problema. Una vez resuelto dicho fallo, se usó este planteamiento como base para personalizar a cada personaje con sus elementos correspondientes obteniendo el resultado esperado.

A nivel visual se ha modificado la posición del botón de pausa debido a que tapaba una parte de la pantalla y a veces los enemigos o los premios se posicionaban debajo de este. Se barajaron dos opciones: mapear el botón para evitar que la conga, los tesoros y los enemigos pudieran pasar o aparecer debajo del botón o cambiar su ubicación. La decisión tomada fue el cambio de ubicación a la parte de los botones intentando que intercediera lo mínimo posible con la puntuación y los botones de dirección.

Por último, se intentó que apareciera un número aleatorio de enemigos durante las partidas, pero hubo que descartar la idea dado que habría que modificar los parámetros desde varias clases y almacenar todas las coordenadas en un array. No se encontró la manera de poder llevarlo a cabo ni de forma eficiente ni de forma satisfactoria, por lo que se decidió únicamente que apareciera un enemigo y que este cambiara la posición según el número de seguidores.

# **CONCLUSIONES**

Desde el inicio del bloque 3 de la asignatura de Laboratorios de Dispositivos móviles se ha podido aprender a realizar un juego sin necesidad de *Activities*, solo usando las bibliotecas de Android Studio, por lo que realizar mejoras y nuevas pantallas en esta práctica nos ha servido para aprender más acerca de ellas y cómo se trabaja mediante coordenadas para colocar todo en el lugar deseado y que al visualizarlo se comporte de forma dinámica.

También se puede decir que se ha aprendido a hacer un poco de evolución en proyectos de Android Studio lo cual es bueno porque no siempre se trabajará con códigos propios, sino que es bastante más probable que se trabaje con códigos de otras personas y este tipo de prácticas ayuda a los alumnos mejorar como programadores.

En líneas generales hemos aprendido mucho con esta práctica en todos los sentidos y al tener la libertad de modificarla a nuestro gusto, pero siguiendo siempre las indicaciones, nos permite dar rienda suelta a nuestra imaginación y aprender mientras divertimos desarrollándola.

# **BIBLIOGRAFÍA**

Developers, G. (s.f.). *developer.android.com*. Obtenido de https://developer.android.com/

Smyth, N. (2020). *Android Studio 4.0 Development Essentials Java Edition.* Payload Media, Inc.