

Hero's Path

I.E.S San José
Cortegana (Huelva)



Resumen

Hero's Path es un videojuego tipo *Role Play Game* 2D para dispositivos Android. Funciona a través de combates por turno, exploración lineal de un mapa aleatorio que varía en cada partida, decisiones aplicables del jugador y mejora de personaje. Con este proyecto se busca demostrar conocimientos técnicos en desarrollo con Jetpack Compose, Kotlin y Diseño Gráfico.

Palabras Clave

Tabla de contenido

Resumen	2
Palabras Clave	2
1. Introducción	3
2. Estado del arte	3
3. Estudio de viabilidad.	3
DAFO	3
4. Requisitos	4
5. Diseño.....	4
6. Codificación	4
7. Documentación Técnica	5
8. Pruebas	5
Mapa	5
Batallas.....	6
Tiendas	7
Conclusiones	8



1. Introducción

"Hero's Path" es un videojuego de rol clásico basado en combates por turnos y progresión de personaje. El objetivo principal es crear una experiencia fácil de entender, pero completa, sirviendo como demostración de capacidades en diseño, programación y desarrollo de interfaces gráficas. Se plantea como juego offline, modular, fácil de expandir y con una estructura clara.

2. Estado del arte

Teniendo en cuenta RPGs clásicos como "Pokémon" y otros como "Loop Hero". La novedad que se trae es un sistema veloz, apenas sin diálogos, aportando a la experiencia *casual*, costando poco tiempo y siendo un ejercicio constante de progresión.

3. Estudio de viabilidad.

DAFO

- **Fortalezas:** Desarrollo modular, interfaz moderna, buen rendimiento.
- **Debilidades:** No tiene motor de físicas.
- **Amenazas:** Dificultad para destacar en un mercado tan saturado.
- **Oportunidades:** Portabilidad y base para ampliar el juego.

Estudio de mercado: El mercado de juegos móviles ha crecido exponencialmente. Los RPGs por turnos siguen teniendo una comunidad fiel. Este tipo de juego puede atraer tanto a nostálgicos como a jugadores casuales por su simplicidad y progresión.



4. Requisitos

Requisitos funcionales:

- Sistema de combate por turnos.
- Tienda y herrería.
- Exploración por nodos.

Requisitos no funcionales:

- UI moderna y fluida.
- Rendimiento aceptable en gama media.
- Arquitectura limpia y mantenible.

5. Diseño

La aplicación está dividida en las siguientes pantallas (mapa, combate, tienda y herrería) con un gestor de estados que las intercambia según las acciones del jugador. Se ha empleado una arquitectura MVVM con ViewModels para separar la lógica de la presentación. Los assets están diseñados en pixel art desde Photoshop.

6. Codificación

Las partes más destacables de la codificación han sido la implementación de “animaciones” al mover al personaje de un nodo a otro, haciendo una transición suave *Ease-In-Out*.

Para mantener la nitidez de las imágenes Pixel Art se ha implementado el uso de `BitmapPaiter` con `FilterQuality.None`.

El clase *GameScreen* centraliza los cambios de estado y gestiona las acciones de la partida.



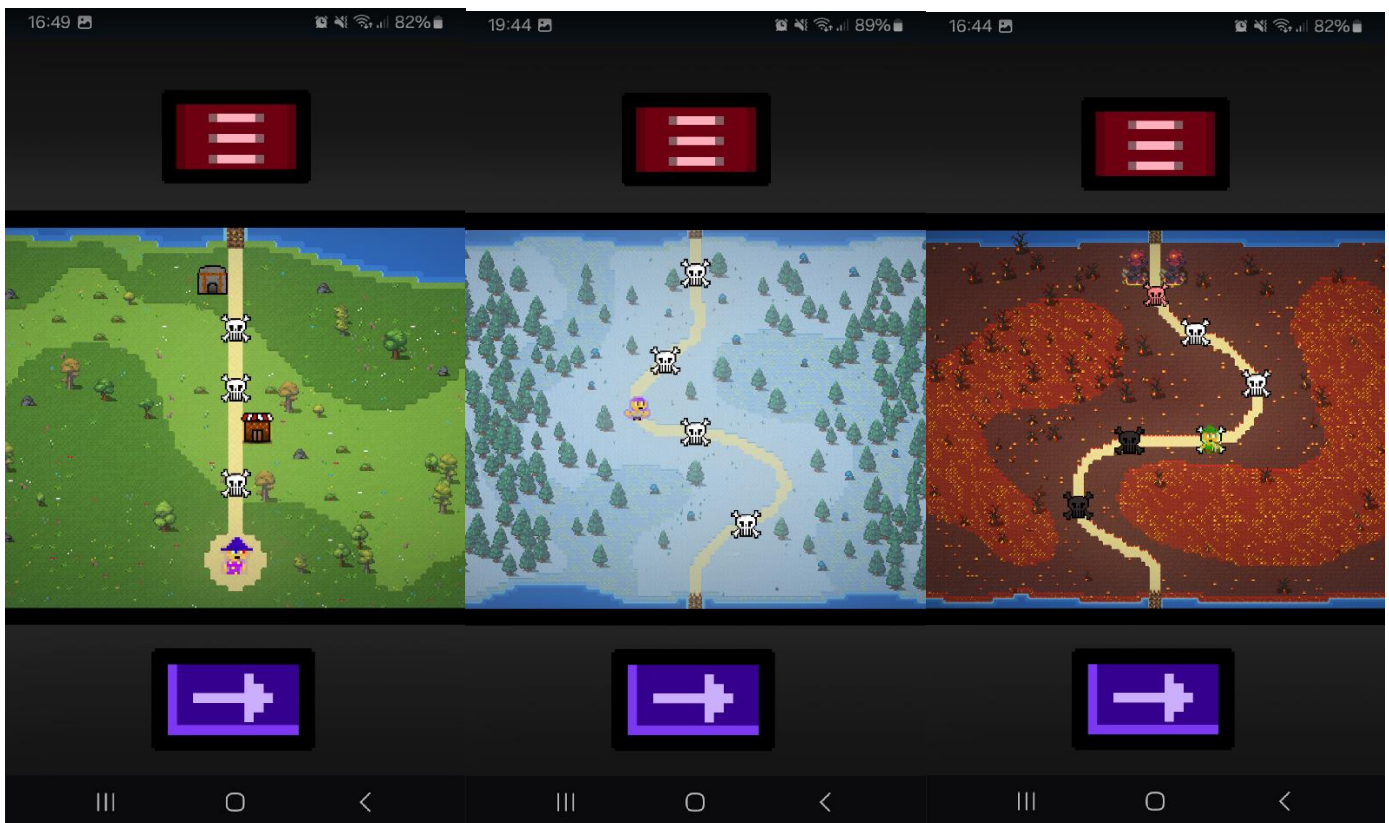
7. Documentación Técnica

- Lenguaje: Kotlin.
- UI: Jetpack Compose.
- Arquitectura: MVVM.
- Librerías: AndroidX.
- Recursos: Drawables y bitmap.

8. Pruebas

Las pruebas se han realizado en el PC del alumno, debido a que el teléfono estaba dando muchos problemas de rendimiento.

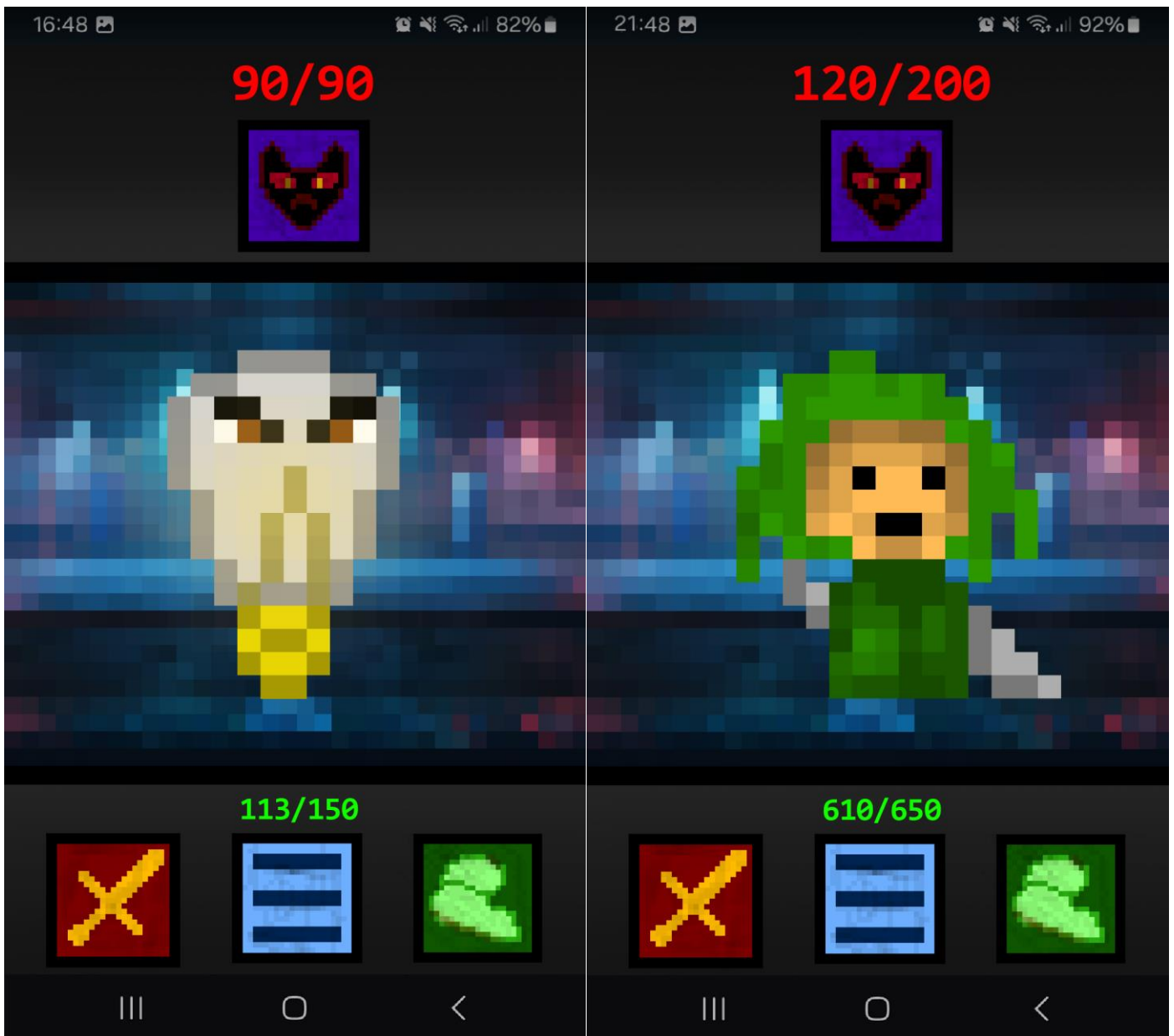
Mapa



Estas 3 imágenes muestran 3 momentos del juego diferentes, la primera muestra el inicio, de la odisea, con la herrería, tienda y 3 enemigos, la segunda muestra una parte más avanzada donde no aparece ninguna tienda. En la última tenemos la zona final, que tampoco tiene tiendas.



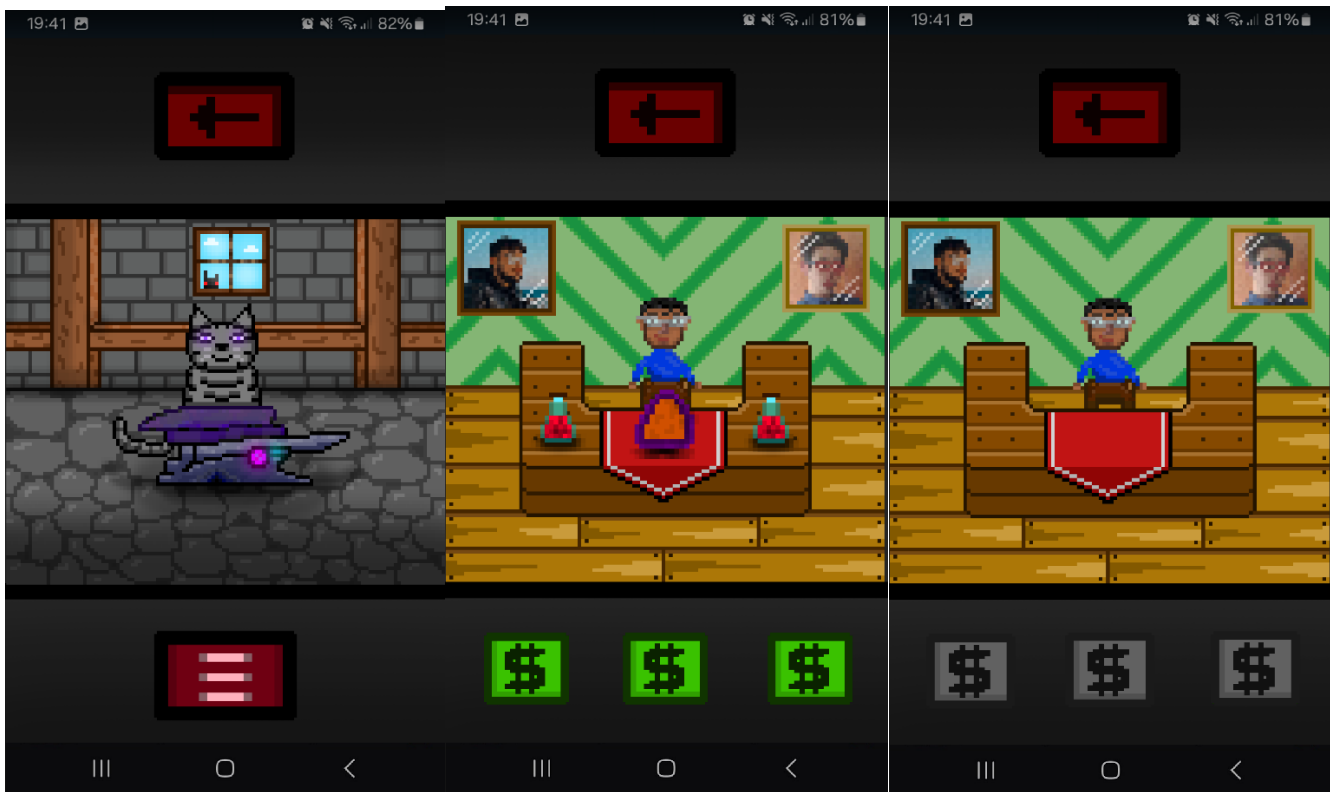
Batallas



En estas dos imágenes se puede apreciar a dos enemigos especiales, el de la izquierda es Marcos Bombi, un guerrero maldito que está obligado a vivir una eternidad iluminando las calles, a la derecha está Shiyo, un caballero hueco que ha viajado mucho para enfrentarse a el jugador.



Tiendas



A la izquierda del todo tenemos a el herrero, un gato extraterrestre llamado Carlillo, su yunque podrá mejorar sin límite tu equipamiento, mientras tengas dinero claro. A la derecha tenemos la tienda de Jack, con dos estados, uno de ellos tiene todos los ítems disponibles a la venta y la otra los tiene todos comprados.



Conclusiones

Este proyecto ha permitido explorar en profundidad el desarrollo moderno en Android, demostrando la capacidad de construir un videojuego sin necesidad de motores externos. La experiencia ha sido enriquecedora tanto a nivel técnico como personal, y deja una base clara para futuras mejoras o expansiones del juego.

