

Memoria del Proyecto – Battle Clicker RPG



Integrantes:

Juan Miguel Sarria Orozco
Alejandro Estebanez Moreno
Eduardo Garcia Rivas

Repositorio: <https://github.com/kazukigd2/CookieClickerRPG>

ÍNDICE

ÍNDICE	2
1. Introducción	3
2. Finalidad y motivación del proyecto	3
3. Arquitectura general de la aplicación	4
3.1 Inspiración en motores gráficos profesionales	4
3.2 GameManager y patrón Singleton	4
3.3 Sistema de combate	5
3.4 Sistema de actualización por ticks	5
4. Diseño de personajes y componentes	6
4.1 Clase base Personaje	6
4.2 Componentes modulares	6
5. Base de datos y persistencia	7
6. Contenido del juego	8
7. Recursos audiovisuales y optimización	9
8. Rendimiento y experiencia de usuario	9
9. Guía de uso	10
9.1. Pantalla de portada	10
9.2. Pantalla de ajustes	11
9.3. Selección de partida	12
9.4. Selección de nivel / Mapa	13
9.5. Estadísticas del jugador	14
9.6. Tienda de estadísticas y habilidades	15
9.7. Almacén / Equipamiento	16
9.8. Tienda	17
9.9. Pantalla de combate	18
9.10. Recompensas tras el combate	19
9.11. Resumen de uso	20
10. Conclusión	20

1. Introducción

Battle Clicker RPG es una aplicación Android desarrollada como **proyecto universitario** dentro de la asignatura Sistemas Empotrados y Dispositivos Móviles. Se trata de un **juego de rol incremental (idle RPG)** que combina mecánicas clásicas de *clicker* con un **sistema de combate RPG completo**, progresión de personaje, equipamiento, habilidades y enfrentamientos contra enemigos y jefes.

El objetivo principal de la aplicación es **el entretenimiento**, ofreciendo al usuario una experiencia de juego progresiva, accesible y optimizada para dispositivos móviles. El jugador puede avanzar incluso en sesiones cortas, pero al mismo tiempo profundizar en la gestión de estadísticas, habilidades y estrategia de combate.

Aunque el proyecto nace como una práctica académica, desde el inicio se ha planteado con una **visión de producto real**, siguiendo buenas prácticas de arquitectura, rendimiento y diseño, con la intención de que pueda ser publicado en la **tienda oficial de Android** una vez finalizada la asignatura.

Con el fin de facilitar la evaluación y prueba de la aplicación, en la raíz del proyecto se incluye una carpeta denominada **APK**, que contiene el archivo instalable de la aplicación listo para su ejecución en dispositivos Android compatibles, sin necesidad de compilación adicional.

2. Finalidad y motivación del proyecto

La finalidad de Battle Clicker RPG es demostrar que una aplicación Android puede ir más allá de un prototipo sencillo, implementando:

- Una **arquitectura modular y escalable**
- Lógica de juego compleja y desacoplada de la interfaz
- Persistencia de datos robusta mediante base de datos sqlite
- Soporte completo para diferentes dispositivos, orientaciones y estados del sistema

El proyecto busca también **resolver uno de los problemas habituales en aplicaciones móviles de juegos**: el alto consumo de batería y recursos. Para ello se ha diseñado un sistema de actualización basado en *ticks* optimizados y una gestión eficiente de recursos gráficos y de audio.

3. Arquitectura general de la aplicación

3.1 Inspiración en motores gráficos profesionales

Debido a la experiencia previa del grupo con **motores gráficos como Unreal Engine**, se decidió diseñar el juego siguiendo una arquitectura similar a la de un **motor de juego**, separando claramente:

- Lógica de juego
- Estado del combate
- Interfaz gráfica
- Persistencia de datos

Este enfoque permite que la aplicación sea **mantenible, escalable y fácil de ampliar** en el futuro.

3.2 GameManager y patrón Singleton

El núcleo de la aplicación es la clase **GameManager**, implementada como **Singleton**, que actúa como punto central del estado del juego. Sus responsabilidades principales incluyen:

- Gestión del progreso del jugador
- Control de sesiones de combate
- Comunicación entre la lógica del juego y la interfaz
- Acceso controlado a la base de datos

Este patrón evita duplicidades de estado y facilita la sincronización entre actividades.

3.3 Sistema de combate

El sistema de combate está dividido en varias clases claramente diferenciadas:

- **CombatSystem**
Contiene toda la lógica del combate: cálculo de daño, críticos, esquiva, efectos, habilidades, ataques en área y pasivas.
- **CombatSession**
Representa una partida o encuentro concreto, gestionando enemigos, oleadas y el estado actual del combate.
- **CombatListener (Interfaz)**
Actúa como puente entre la lógica y la interfaz gráfica, permitiendo que la UI reaccione a eventos como daño, muerte de enemigos, uso de habilidades o efectos visuales, sin acoplar directamente la lógica al diseño visual.

Este diseño facilita que la interfaz pueda cambiarse sin modificar la lógica interna del juego.

3.4 Sistema de actualización por ticks

El juego utiliza un **sistema propio de ticks** para actualizar la lógica:

- **Tick de lógica:** 100 ms
Se encarga del combate, efectos, ataques automáticos y temporizadores.
Este valor se ha probado para asegurar fluidez sin un consumo excesivo de batería.
- **Tick de interfaz:** 200 ms
Actualiza textos, barras de vida y estados visuales, ya que la UI no requiere refrescos tan rápidos como la lógica.

Este sistema garantiza un **equilibrio entre rendimiento, precisión y consumo energético**, especialmente importante en dispositivos móviles.

4. Diseño de personajes y componentes

4.1 Clase base Personaje

Se ha diseñado una clase genérica **Personaje**, que contiene los elementos comunes a cualquier entidad del juego:

- Estadísticas (vida, ataque, defensa, crítico, esquiva, etc.)
- Sistema de habilidades
- Sistema de efectos y modificadores
- Gestión de daño y curación

A partir de esta clase heredan:

- **Jugador**
- **Enemigo**

Esto evita duplicación de código y permite añadir nuevos tipos de personajes fácilmente.

4.2 Componentes modulares

Tanto el jugador como los enemigos utilizan un sistema de **componentes**, entre ellos:

- Estadísticas (Stats)
- Habilidades activas
- Habilidades pasivas
- Items
- Equipo

Cada personaje solo utiliza los componentes que necesita, lo que hace el sistema flexible y extensible.

5. Base de datos y persistencia

La aplicación utiliza una base de datos **SQLite** que se encuentra en la carpeta del proyecto en **src/main/assets/game.db** la cual se copia al proyecto en cuanto se instala la aplicación, dicha base de datos está diseñada con una arquitectura clara basada en:

- **Contract**: definición de tablas y columnas
- **DAO**: acceso seguro a datos
- **Model**: clases Java para trabajar siempre con objetos

La base de datos se divide conceptualmente en:

- **Tablas estáticas**

Contienen datos del juego como:

- Enemigos
- Habilidades
- Items
- Equipo
- Estadísticas base
- etc...

	<i>enemigo_id</i>	vida_max	mana_max	ataque	defensa	vel_ataque_ms	critico	danio_critico	punteria	esquiva	vida_regen	mana_regen
	Filtro	Filtro	Filtro	Filtro	Filtro	Filtro	Filtro	Filtro	Filtro	Filtro	Filtro	Filtro
1	1	325	100	15	7	1585	0.053	1.5	0.8	0.032	1.1	0.0
2	2	364	100	17	8	1570	0.056	1.5	0.8	0.034	1.2	0.0
3	3	408	100	19	9	1555	0.059	1.5	0.8	0.036	1.3	0.0
4	4	457	100	21	10	1540	0.062	1.5	0.8	0.038	1.4	0.0
5	5	539	100	25	13	1525	0.065	1.5	0.8	0.04	1.5	0.0

- Esto permite ampliar el juego simplemente añadiendo nuevas filas a la base de datos.

- **Tablas dinámicas (prefijo Jugador_XXX)**

Almacenan el progreso del jugador:

- Equipamiento actual
- Habilidades desbloqueadas
- Inventario
- Estadísticas mejoradas
- Partidas guardadas

<i>ataque_nivel</i>	<i>baseStats_def</i>	<i>bola_fuego_nivel</i>	<i>critico_nivel</i>	<i>curacion_nivel</i>	<i>defensa_nivel</i>	<i>enemigo_def</i>	<i>enemigo_stats</i>	<i>equipment_def</i>	<i>fase</i>	<i>fase_equipment_drop</i>	<i>fase_exp</i>	<i>fase_item_drop</i>	<i>habilidad_def</i>	<i>item_def</i>	<i>jugador</i>	<i>jugador_equipment</i>	<i>jugador_equipment_ent_equipado</i>	<i>jugador_habilidad</i>	<i>jugador_habilidad_equipada</i>	<i>jugador_item</i>	<i>jugador_item_equipado</i>	<i>jugador_pasivas</i>	<i>jugador_stat_nivel</i>	<i>nivel_def</i>	<i>oleada</i>	<i>oleada_enemigo</i>	<i>pasiva_auto_area</i>	<i>pasiva_doble_golpe</i>	<i>pasiva_reducción_cd</i>	
Filtro	Filtro	Filtro	Filtro	Filtro	Filtro	Filtro	Filtro	Filtro	Filtro	Filtro	Filtro	Filtro	Filtro	Filtro	Filtro	Filtro	Filtro	Filtro	Filtro	Filtro	Filtro	Filtro	Filtro	Filtro	Filtro	Filtro	Filtro	Filtro	Filtro	Filtro
1	Slime	1	normal	enemy_slime	slime																									
2	Slime Corrupto	2	normal	enemy_slimecorrupto	slime																									
3	Slime Venenoso	3	normal	enemy_slimevenenoso	slime																									
4	Slime Cristal	4	normal	enemy_slimecristal	slime																									
5	Slime Rey	5	normal	enemy_slimerey	slime																									
6	Goblin	6	normal	enemy_goblin	goblin																									

Este enfoque facilita la expansión futura sin modificar la lógica del juego.

6. Contenido del juego

Actualmente, Battle Clicker RPG cuenta con:

- Más de **270 commits** en el repositorio
- Aproximadamente **2 horas de duración de juego**
- **50 niveles**
- Más de **200 oleadas de enemigos**
- **50 enemigos distintos**, cada uno con:
 - Sprite de idle
 - Ataque
 - Recibir daño
 - Muerte
 - Sus sonidos propios
- Más de **400 sprites** en total
- **32 piezas de equipamiento**
- **4 habilidades activas**
- **4 habilidades pasivas**
- Sistema completo de estadísticas:
 - Ataque
 - Defensa
 - Crítico
 - Daño crítico
 - Esquiva
 - Puntería
 - Vida
 - Maná
 - Regen Vida
 - Regen Mana

7. Recursos audiovisuales y optimización

Para los recursos gráficos y sonoros se han utilizado herramientas de IA y edición:

- Sprites generados con **IA (Nano Banana – Google)** y retocados manualmente
- Música generada con **SUNO AI**
- Efectos de sonido creados con **ElevenLabs**
- Edición manual para:
 - Ajustar tamaños
 - Eliminar fondos
 - Optimizar imágenes

Todas las imágenes se han convertido a **WebP**, y los audios se han comprimido en .ogg para minimizar el tamaño final de la aplicación y reducir consumo de memoria, el objetivo que aun teniendo más de 400 imágenes, 11 canciones y efectos de sonidos varios la aplicación ocupe alrededor de 60 megas para que no sea penalizada en la store.

8. Rendimiento y experiencia de usuario

La aplicación ha sido diseñada teniendo en cuenta:

- Consumo reducido de batería
- Correcta gestión del ciclo de vida de Android
- Soporte completo para:
 - Rotación de pantalla
 - Segundo plano
 - Pantalla completa (ocultando notificaciones cuando es necesario)
- Interfaz totalmente **responsive**

Además, se han añadido múltiples efectos visuales y sonoros para mejorar la experiencia:

- Shake de enemigos
- Parpadeos de pantalla
- Efectos de habilidades
- Sonidos diferenciados por enemigo

9. Guía de uso

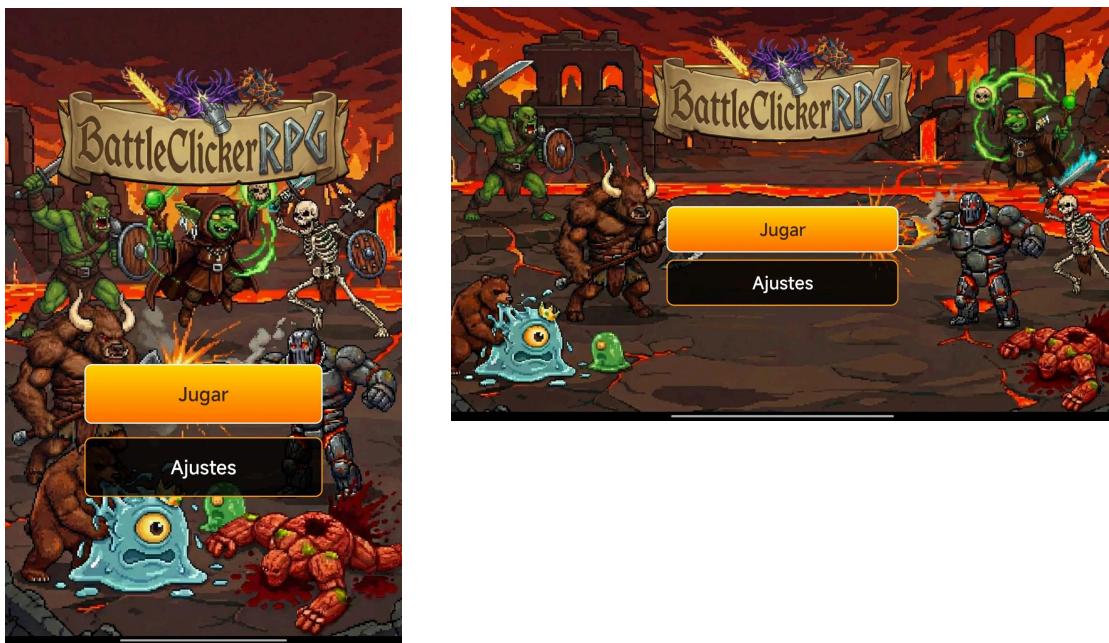
En este apartado se describe el funcionamiento de la aplicación desde el punto de vista del usuario, explicando las diferentes pantallas y cómo interactuar con ellas para jugar correctamente.

Las imágenes proporcionadas han sido sacadas de una tablet, en móvil se verá más compacto.

9.1. Pantalla de portada

Al iniciar la aplicación se muestra la **pantalla principal (portada)** del juego. En esta pantalla el usuario dispone de dos opciones principales:

- **Jugar**: permite acceder al juego y continuar o iniciar una partida.
- **Ajustes**: permite configurar distintos parámetros de la aplicación.



9.2. Pantalla de ajustes

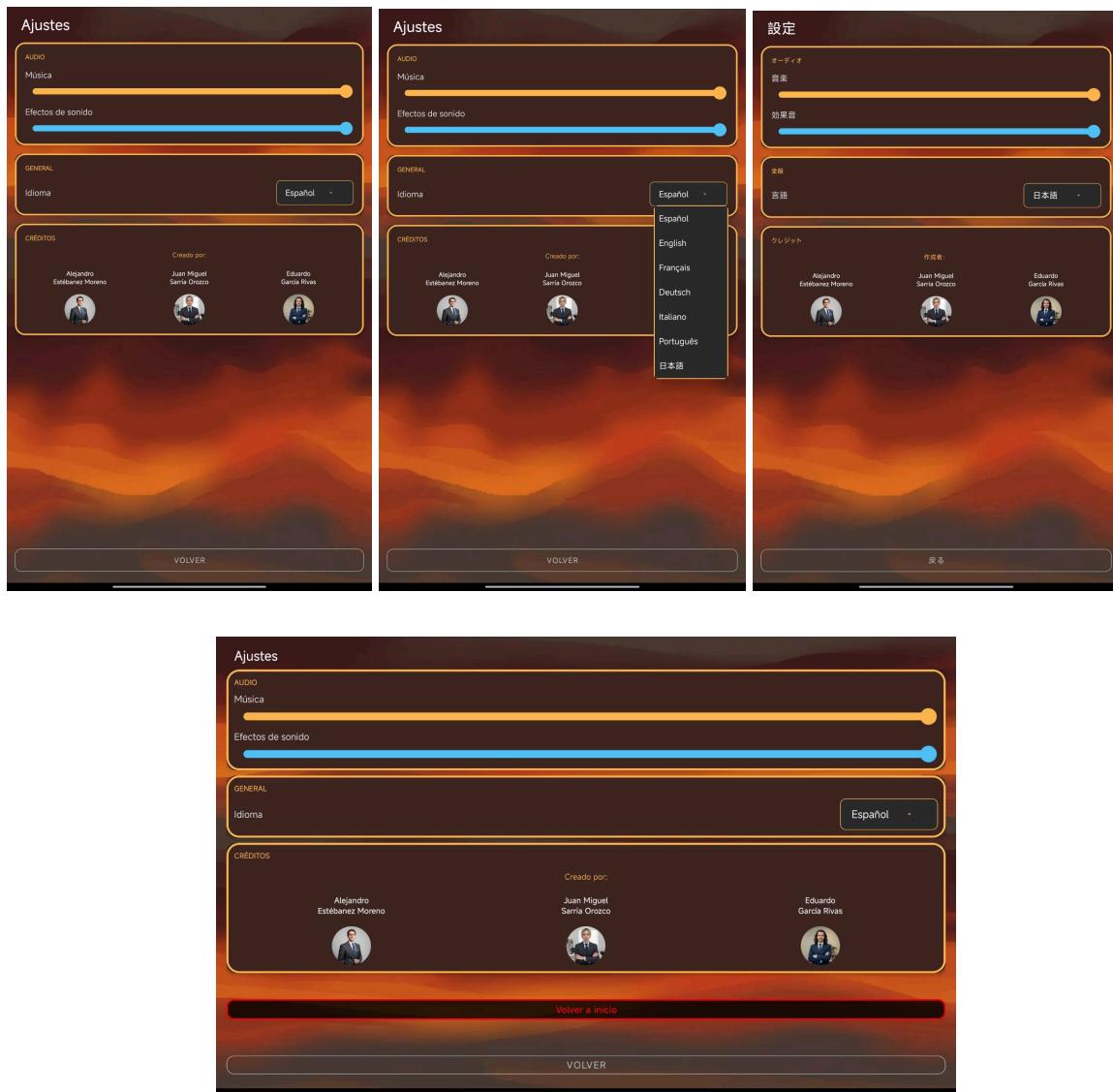
Desde la pantalla de ajustes el usuario puede personalizar la experiencia de juego. Las opciones disponibles son:

- Activar o desactivar **efectos de sonido**
- Activar o desactivar **música**
- Cambiar el **idioma de la aplicación**

La aplicación cuenta con **soporte multi-idioma**, incluyendo entre otros:

- Español, Ingles, Japones, Francés, Alemán, Portugues e Italiano

Además, en esta pantalla se muestra un apartado de **créditos**, donde se indican los integrantes del grupo desarrollador del proyecto.



9.3. Selección de partida

Al pulsar el botón **Jugar**, el usuario accede a la pantalla de **selección de partida**.

En esta pantalla es posible:

- **Cargar una partida existente**, conservando el progreso anterior
- **Crear una nueva partida**, comenzando el juego desde el inicio
- **Borrar** una partida existente

El juego dispone de **4 ranuras de guardado**, permitiendo mantener distintos progresos de forma independiente.



9.4. Selección de nivel / Mapa

Una vez cargada o creada una partida, el jugador accede a la **pantalla principal de juego**, donde se muestra el **mapa de niveles**.

Desde esta pantalla se puede:

- Pulsar sobre el mapa en el que quieras **iniciar el siguiente combate**
- Acceder a distintos paneles mediante botones de navegación

Controles disponibles:

- **Parte superior:**
 - **Parte superior derecha:** botón de ajustes (acceso a la misma pantalla de ajustes descrita anteriormente)
 - **Parte superior centro:** botón que lleva a una pantalla con el resumen de las estadísticas del jugador
- **Parte inferior de la pantalla:** botones de acceso a distintas secciones del juego



9.5. Estadísticas del jugador

Al pulsar sobre el nivel/experiencia/oro de la pantalla de selección de nivel / Mapa se abrirá una pantalla que mostrará las **estadísticas del jugador** en ese instante, incluye también el equipamiento equipado y las habilidades equipadas.

FICHA DEL JUGADOR

NIVEL JUGADOR: 35			
ATRIBUTO	BASE	EQUIPO	TOTAL
Vida (Nv.35)	18857	+4793	23650
Maná (Nv.35)	100	+35	135
Ataque (Nv.35)	943	+240	1183
Defensa (Nv.35)	882	+84	966
Velocidad (Nv.35)	443	-90	353
Crítico	15,0%	+6,0%	21,0%
Daño Crit.	x1,5	+0,4	x1,9
Puntería	90%	+30%	120%
Esquiva	5%	+12%	17%
Reg. Vida	2,0 /s	+9,6 /s	11,6 /s
Reg. Maná	2,0 /s	+1,0 /s	3,0 /s
PASIVAS			
Auto área	60%		
Doble golpe	60%		
Reducción CD	3 s		

FICHA DEL JUGADOR

NIVEL JUGADOR: 60			
ATRIBUTO	BASE	EQUIPO	TOTAL
Auto área	60%		
Doble golpe	60%		
Reducción CD	3 s		
Reducción maná	30%		
EQUIPAMIENTO			
ARMA	Filo del Caos		
PECHERA	Armadura del Caos		
PANTALÓN	Calzas del Vacío		
ANILLO	Sello del Caos		
HABILIDADES			
PRIMARIA	Bola de Fuego		
SECUNDARIA	Veneno		

FICHA DEL JUGADOR

NIVEL JUGADOR: 35			
ATRIBUTO	BASE	EQUIPO	TOTAL
Vida (Nv.35)	18857	+4793	23650
Maná (Nv.35)	100	+35	135
Ataque (Nv.35)	943	+240	1183
Defensa (Nv.35)	882	+84	966
Velocidad (Nv.35)	443	-90	353
Crítico	15,0%	+6,0%	21,0%
Daño Crit.	x1,5	+0,4	x1,9
Puntería	90%	+30%	120%

9.6. Tienda de estadísticas y habilidades

Desde el botón correspondiente se accede a la **zona de mejora de estadísticas y habilidades**.

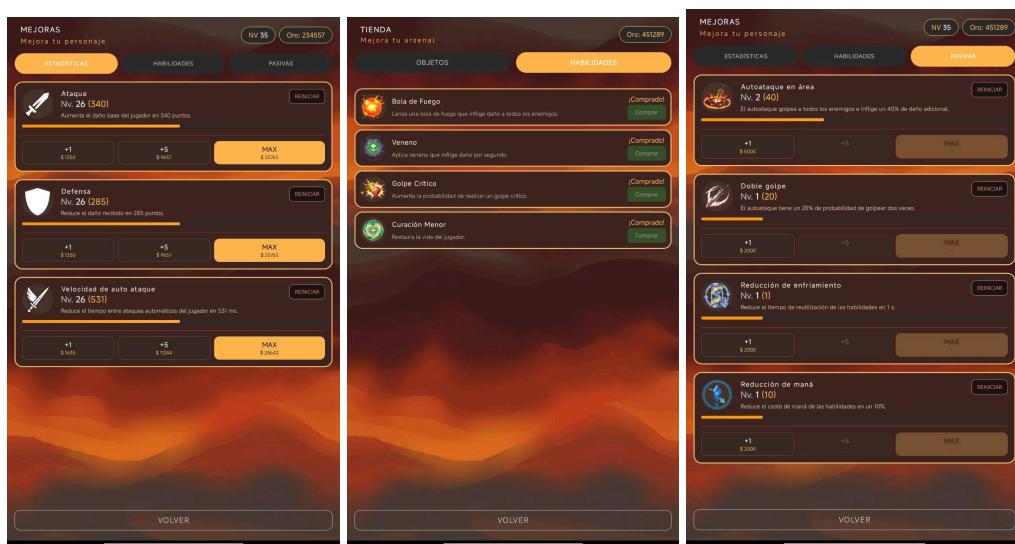
En esta sección el jugador puede:

- Mejorar estadísticas básicas como:
 - Ataque
 - Defensa
 - Velocidad de ataque
- Mejorar habilidades activas
- Mejorar habilidades pasivas

Las mejoras pueden realizarse:

- De **1 en 1**
- De **5 en 5**
- Al **máximo posible**, según el oro disponible

Esto permite al jugador adaptar su progresión a su estilo de juego.



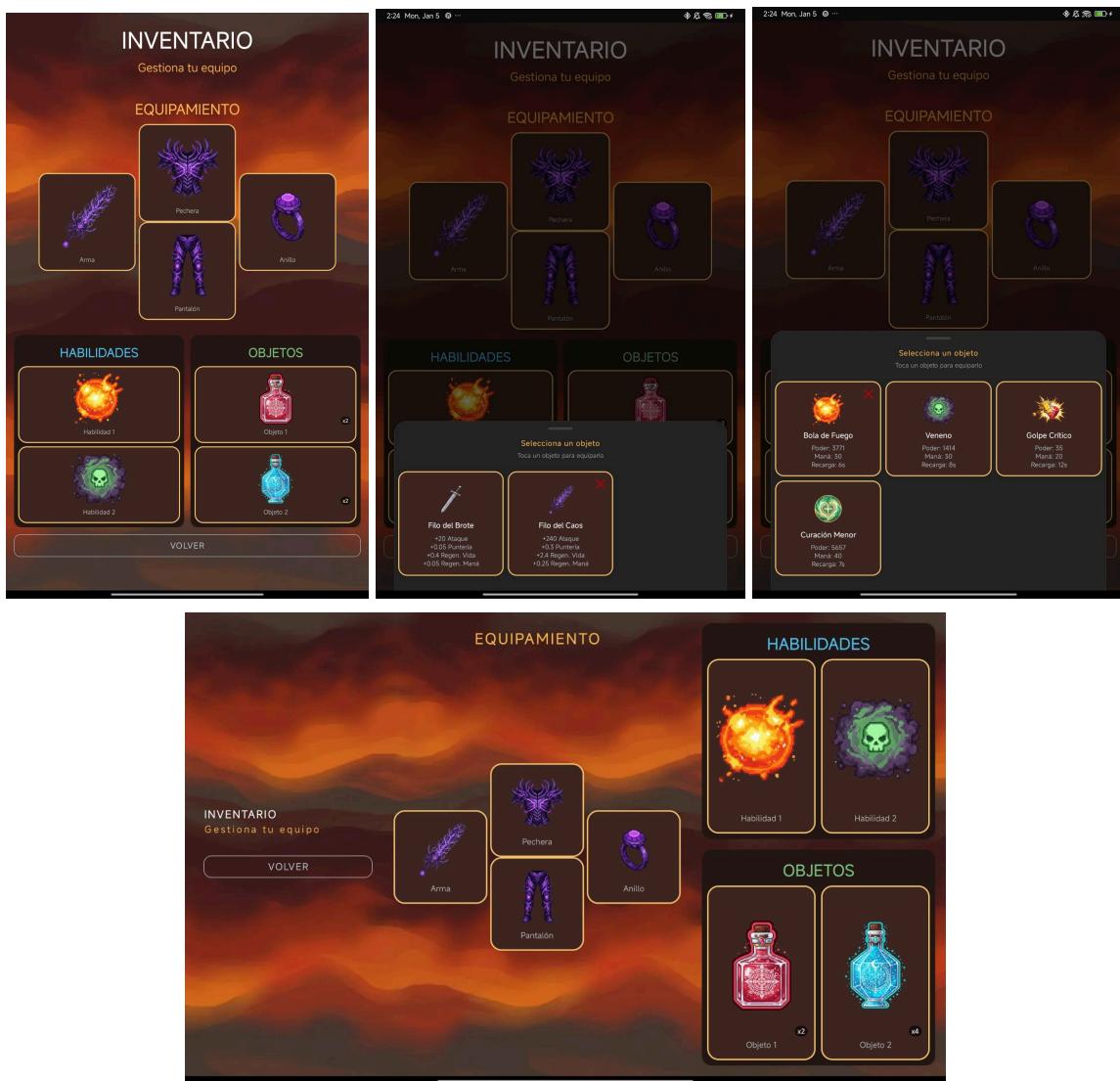
9.7. Almacén / Equipamiento

En la sección de **almacén**, el jugador puede gestionar su equipamiento y habilidades.

Desde aquí es posible:

- Equipar piezas de equipo:
 - Casco, Pechera, Pantalones, Anillo
- Equipar:
 - Hasta **2 habilidades activas**
 - Hasta **2 objetos (items)** para usar en combate

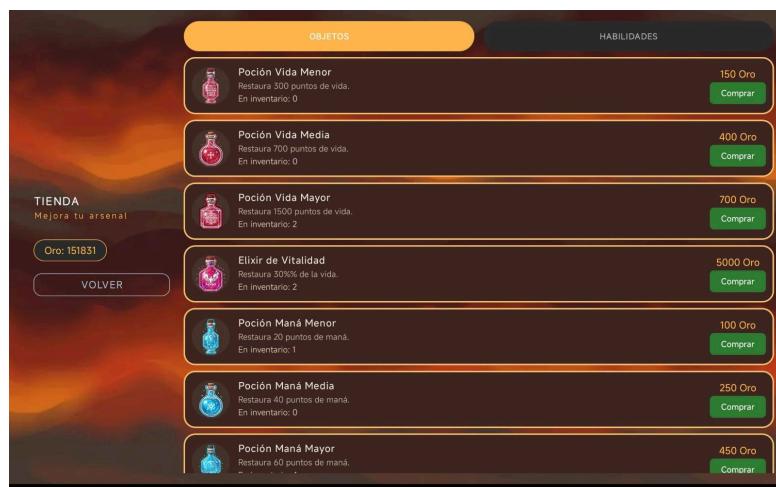
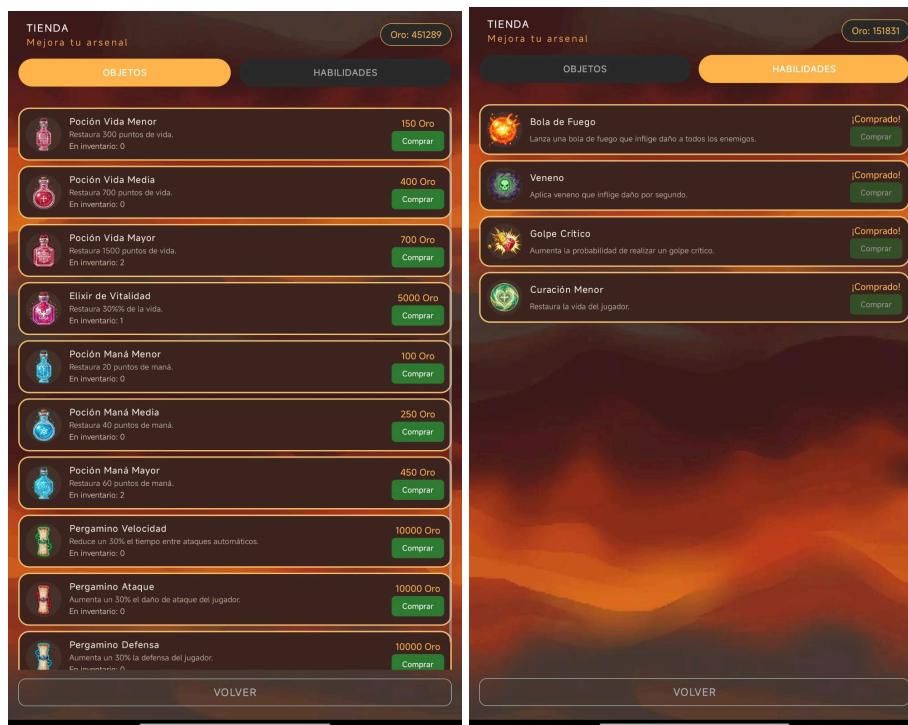
El equipamiento modifica las estadísticas del personaje, permitiendo crear diferentes combinaciones y estrategias.



9.8. Tienda

En la tienda el jugador puede gastar el oro obtenido para adquirir nuevos recursos, como:

- **Items de apoyo**, por ejemplo:
 - Pociones de vida
 - Pociones de maná
- **Objetos temporales** como pergaminos que aumentan estadísticas durante un tiempo limitado
- **Habilidades** que se pueden desbloquear y usar en combate



9.9. Pantalla de combate

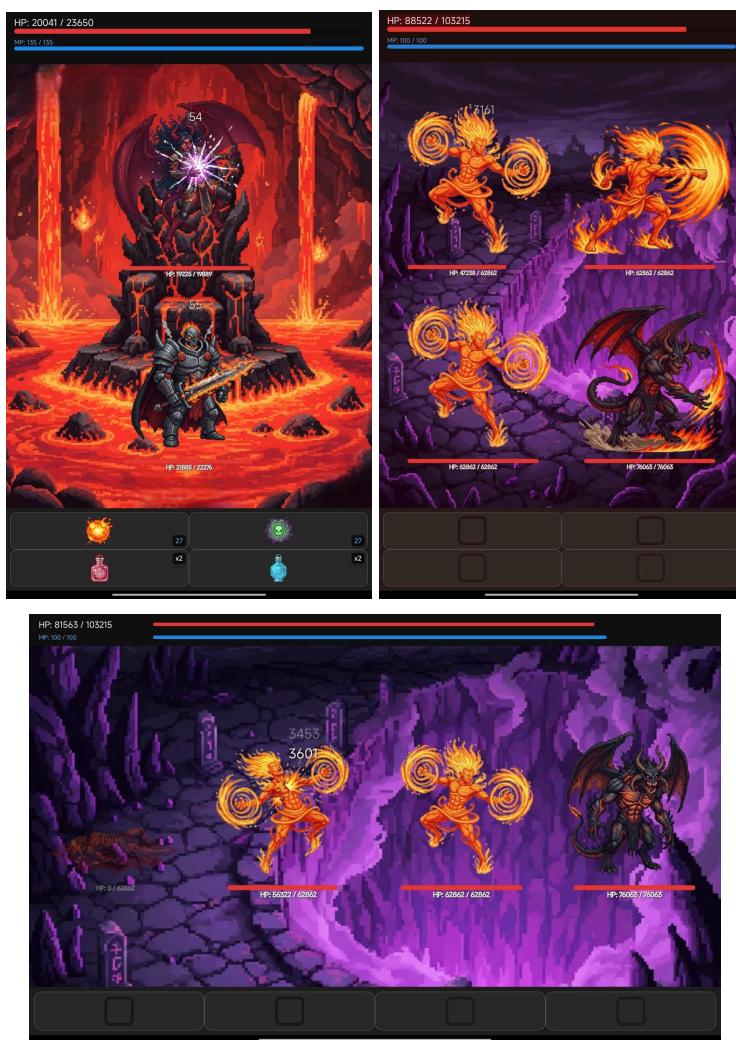
Al iniciar un nivel, el jugador entra en la **pantalla de combate**.

Durante el combate:

- El jugador puede **tocar (clicar)** a los enemigos para infilir daño
- Existe un sistema de **autoataque**, que ataca automáticamente según la velocidad de ataque del personaje
- Las **habilidades pasivas** pueden modificar el comportamiento del combate, por ejemplo:
 - Autoataque en área o Doble golpe automático

En la parte inferior de la pantalla se muestran:

- **4 botones** correspondientes a las habilidades activas equipadas
- Los **items equipados**, que pueden utilizarse durante el combate

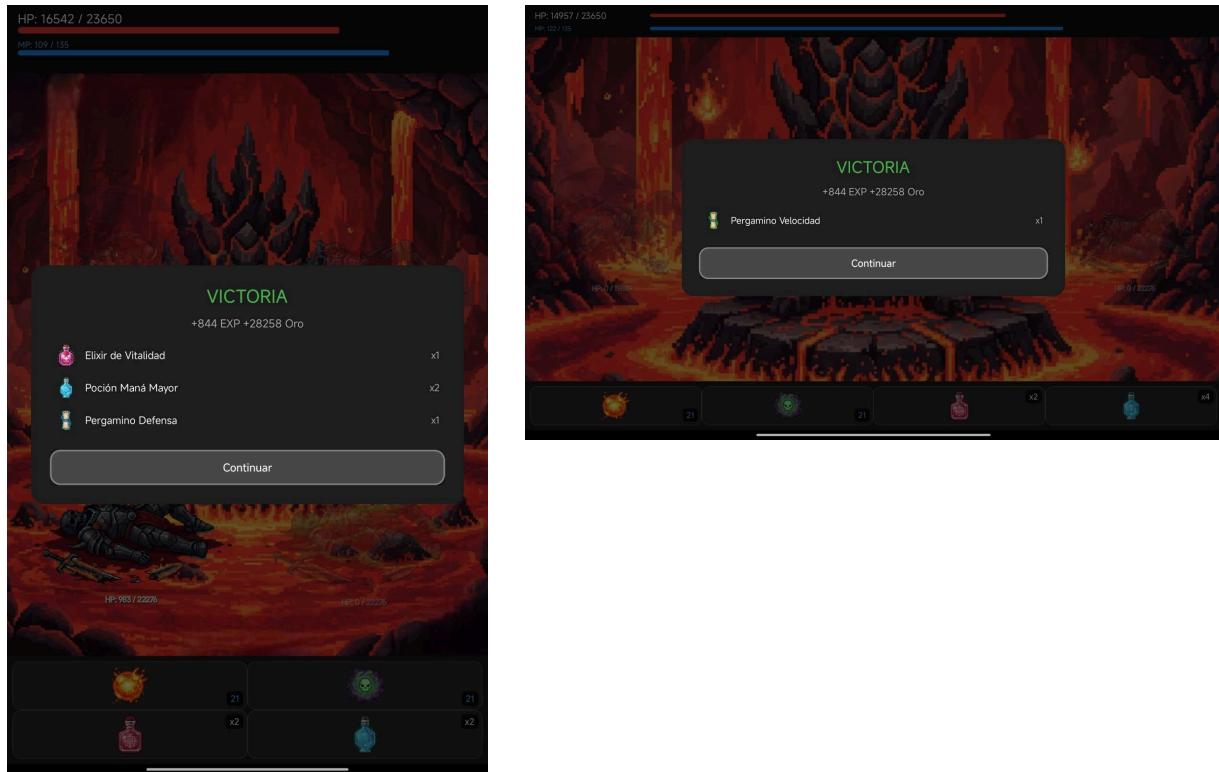


9.10. Recompensas tras el combate

Una vez finalizado un combate con éxito, se muestra una **ventana emergente (popup)** con las recompensas obtenidas, que pueden incluir:

- Oro
- Experiencia
- Equipamiento
- Items

Estas recompensas permiten continuar progresando en el juego y mejorar al personaje.



9.11. Resumen de uso

El flujo normal de uso de la aplicación es el siguiente:

1. Iniciar la aplicación
2. Configurar ajustes (opcional)
3. Seleccionar o crear partida
4. Elegir nivel desde el mapa
5. Gestionar estadísticas, equipamiento y habilidades
6. Combatir contra enemigos
7. Obtener recompensas y progresar

Este diseño permite una experiencia de juego clara, accesible y progresiva.

10. Conclusión

Battle Clicker RPG no es únicamente una práctica académica, sino un proyecto desarrollado con una mentalidad profesional, aplicando conceptos avanzados de arquitectura, diseño y optimización. El resultado es una aplicación Android sólida, escalable y preparada para su publicación, que cumple sobradamente los requisitos técnicos exigidos y demuestra un alto nivel de implicación por parte del equipo.