

Proyecto de Fin de Ciclo

1 Resumen.

El proyecto consistirá en el desarrollo de un videojuego de estilo RPG con perspectiva en dos dimensiones (2D) ambientado en el folklore y la mitología ibérica medieval.

Debido a la limitación temporal para realizar el proyecto, se presentará tan solo una versión limitada (demo jugable) que constará de cinco escenas:

Tres de estas escenas se desarrollarán en escenarios bidimensionales con desplazamiento vertical y horizontal. El objetivo de las mismas será presentar al jugador la posibilidad de explorar el escenario.

Las otras dos consistirán en combates por turnos contra diferentes enemigos.

2 Introducción.

Siendo nuestro hobby los videojuegos, vimos en la propuesta de desarrollo de una aplicación la oportunidad de explorar dicha afición desde el otro lado de la pantalla, es decir, siendo nosotros mismos quienes creen un mundo fantástico en el que otras personas pudieran vivir aventuras a través de los personajes que inventemos.

Esto nos permitirá explorar nuestra creatividad al mismo tiempo que aprendemos a usar nuevas tecnologías y sentamos una base para futuros proyectos más complejos dentro del campo del desarrollo de videojuegos.

AQUI FALTA LA SOLUCIÓN

3 Estado del arte.

Debido a esta afición que compartimos, no nos faltan referencias que nos sirvan de guía a la hora de darle forma a nuestro proyecto.

Nos basamos en títulos como The Legend of Zelda (1986) y Dragon Quest V para las escenas de exploración del mundo o en los títulos clásicos de la saga Pokémon y Final Fantasy para las de combate.

Los roles de los personajes y las diferentes mecánicas de combates están inspiradas en algunas propias de juegos como League of Legends o World of Warcraft.

Para las estadísticas de habilidades, items y personajes hemos recurrido a juegos de rol clásicos como Dungeons & Dragons o Rolemaster.

Respecto a la ambientación hemos tomado como referentes el juego de rol Akelarre y el videojuego Blasphemous.

4 Estudio de viabilidad. Método DAFO.

Debilidades	Fortalezas
<p>Al tener un producto ambientado en la cultura local, podría no ser tan atractivo para público internacional.</p> <p>No tener un sistema de combate convencional implica una curva de aprendizaje mas lenta.</p> <p>La temática del juego puede no ser atractiva para todo tipo de público.</p>	<p>Originalidad: el sistema de combate, la ambientación y los personajes no son los convencionales dentro del género.</p> <p>Bajos requisitos gráficos: Al ser un proyecto de pixel-art no requiere de un motor gráfico excesivamente potente.</p> <p>Ponemos en valor nuestra cultura al tomar como inspiración el folklore y leyendas Ibéricas.</p>
Amenazas	Oportunidades
<p>El mercado de los videojuegos está muy explotado, por lo que puede hacerse complicado situar en él un nuevo producto.</p>	<p>Nuestro target objetivo es muy amplio, actualmente el consumo de videojuegos está ampliamente extendido entre grandes sectores de población.</p> <p>A día de hoy, los canales de distribución de software no necesitan soporte físico, por lo que es relativamente sencillo hacer llegar a los consumidores nuestro producto.</p>

4.1 Estudio de mercado.

4.1.1 Viabilidad técnica/económica del proyecto.

A pesar de que podría considerarse que el mercado del videojuego actual se encuentra saturado consideramos que nuestro producto podría encontrar un lugar en el mismo debido a su planteamiento original en cuanto a temática. Estamos acostumbrados a videojuegos que se inspiran en culturas de tradición anglo-sajona u oriental pero nosotros consideramos que el folklore y cultura española es tan rica como éstas y está mucho menos explotada. Es por esto que creemos que éste es uno de los puntos fuertes de nuestro producto.

El público al que va específicamente dirigido se centra en la franja de edad comprendida entre los 25-40 años dado que el diseño y apariencia del mismo podría considerarse “retro” (pixel-art, escenarios simples, sistema de control sencillo, etc.).

4.1.2 Recursos HW.

Por otro lado, estimamos que los recursos necesarios para poder ejecutar el videojuego y jugar a él de forma fluida serían:

Procesador de 64 bits.

Procesador: Intel(R) Core (TM) i5.

Memoria: 8 GB de RAM.

Almacenamiento: 2 GB de espacio disponible.

4.1.3 Recursos SW.

Para desarrollar la demo que presentaremos vamos a utilizar los siguientes recursos:

Sistema operativo Windows 10.

Motor de creación de videojuegos Unity.

Entorno de Desarrollo Integrado Rider.

Software de gestión de efectos sonoros y música.

En cuanto a los recursos necesarios para correr el juego, estimamos que serán:

Sistema operativo de 64 bits.

4.1.4 Recursos humanos.

El equipo de desarrollo estará integrado por los dos desarrolladores de videojuegos con mayor proyección de futuro de toda la comarca del valle de Ricote:

Alfonso Ruiz Cava.

Daniel Ortuño Murcia.

4.1.5 Viabilidad temporal.

Planificación 5%

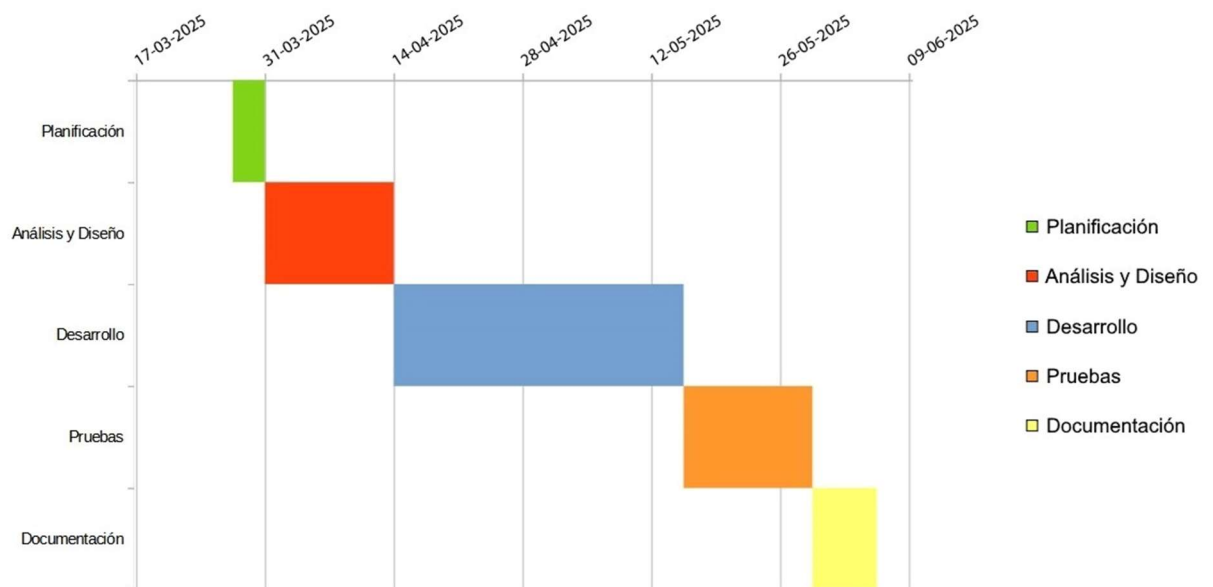
Análisis y Diseño 20%

Desarrollo 45%

Pruebas 20%

Documentación 10%

4.2 Planificación temporal o agenda de trabajo.



5 Análisis de requisitos.

5.1 Descripción de requisitos.

En nuestro videojuego el jugador controlará a un grupo de 4 personajes con los que podrá explorar un mundo de fantasía y enfrentarse a diversos enemigos mediante escenas de combate por turnos.

5.1.1 Texto explicativo.

Descripción del mundo:

La historia tiene lugar en un mundo imaginario basado en los reinos medievales de la Península Ibérica.

Descripción del guion:

Trascurre a través de 5 diferentes escenas.

- Escena 1ª: Poblado en el que los personajes podrán interactuar con diferentes NPCs y prepararse para la aventura obteniendo diversos objetos.
- Escena 2ª: Escenario que proporciona un entorno que explorar a los personajes jugables (PJ). En el pueden aparecer enemigos que darán paso a la siguiente escena de combate.
- Escena 3ª: Combate contra un grupo de personajes enemigos (PE). Será un enfrentamiento de baja dificultad orientado a que el jugador pueda familiarizarse con los controles y sistemas de combate.
- Escena 4ª: Transición visual en la que al jugador se le muestra una serie de diálogos entre los personajes jugables (PJ) y los personajes enemigos (PE) que desemboca en la escena final.
- Escena 5ª: Combate contra el jefe final. Nueva escena de combate en la que el jugador deberá aplicar las técnicas aprendidas en la tercera escena para derrotar al líder de los PEs de más bajo nivel.
- Escena 6ª: Conclusión de la historia y créditos.

Descripción de la exploración:

Los personajes se desplazarán por el mundo a través de tres escenarios principales:

- Comenzarán en una zona segura, en la que no hay personajes hostiles, el pueblo. Aquí se podrán familiarizar con los controles de movimiento e interactuarán con diversos personajes de los que obtendrán información y/o items que les podrán ayudar en posteriores eventos.
- Una vez el jugador haya hecho todo lo que quiera en el pueblo, saldrá al mundo exterior. Éste es un nuevo escenario que podrá explorar y en el que se podrá encontrar con personajes enemigos con los que se entablará un combate.
- Tras derrotar al primer grupo de PEs, se desbloqueará el paso hacia el tercer y último escenario en el que se librará un combate contra un jefe final. La transición entre estas dos partes del juego se hará mediante una escena en la que el jugador no controlará a los personajes si no que estos se de forma automática por el escenario.

Descripción del combate:

El combate se desarrolla siguiendo el clásico sistema de turnos.

Los personajes irán interviniendo en el combate uno tras otro de forma secuencial siguiendo un orden que se determina mediante el siguiente sistema: Por cada personaje se crea un número aleatorio entre 0 y 10 que se modifica sumando el valor de agilidad de cada personaje, el resultado obtenido determina el valor de iniciativa de cada personaje para esta ronda del combate. Los personajes con mas iniciativa actúan primero.

Durante el turno de cada PJ, el jugador seleccionará una de las cuatro posibles acciones, esta tendrá el efecto correspondiente y se pasará al turno del siguiente personaje.

Esta secuencia se repetirá hasta que todos los integrantes de uno de los dos equipos hayan sido eliminados, momento en el que se dará por finalizado el combate.

Descripción de personajes:

El grupo de personajes jugables (PJ) estará conformado por 4 personajes con diferentes características, roles y apariencia:

- Guerrero:
 - Apariencia: Hombre alto y corpulento equipado con una armadura de placas metálicas, gran escudo y espada.
 - Rol: Tanque, es el encargado de recibir el daño proveniente de los enemigos.
 - Estadísticas:

PV	21
PM	8
Fuerza física	4
Resistencia física	6
Poder mágico	2
Resistencia mágica	5
Agilidad	4

- Habilidades:
 - Primera: Ataque básico. Aplica daño moderado y aumenta agro de forma moderada.
 - Segunda: Defensa con escudo. Mitiga el daño recibido.
 - Tercera: Ataque especial. Efecto de área que aumenta drásticamente el agro.
 - Cuarta: Utilización de objetos.

- Hechicera:

- Apariencia: Es una mujer de edad avanzada ataviada con túnica y libro.
- Rol: DPS mágico.
- Estadísticas:

PV	11
PM	21
Fuerza física	2
Resistencia física	3
Poder mágico	7
Resistencia mágica	5
Agilidad	6

- Habilidades:
 - Primera: Ataque básico. Aplica daño mágico a un solo objetivo. Baja ligeramente la defensa del enemigo.
 - Segunda: Meditación. Recupera energía.
 - Tercera: Ataque especial. Efecto de área causa gran daño mágico.
 - Cuarta: Utilización de objetos.

- Arquera:

- Apariencia: Mujer alta de complexión delgada con protecciones ligeras de cuero.
- Rol: Dps físico.
- Estadísticas:

PV	17
PM	12
Fuerza física	7
Resistencia física	4
Poder mágico	2
Resistencia mágica	3
Agilidad	8

- Habilidades:

- Primera: Ataque básico. Daño físico a un solo objetivo. Aumenta ligeramente la fuerza física de forma progresiva.
- Segunda: Ataque de un solo objetivo que inutiliza al enemigo durante un tiempo determinado.
- Tercera: Ataque especial. Efecto de área causa gran daño físico.
- Cuarta: Utilización de objetos.

- Sacerdote:

- Apariencia: Hombre de mediana edad vestido con un hábito de monje y un bastón.
- Rol: Soporte.
- Estadísticas:

PV	14
PM	18
Fuerza física	3
Resistencia física	3
Poder mágico	7
Resistencia mágica	4
Agilidad	5

- Habilidades:

- Primera: Curación a un solo objetivo. Reduce efectos negativos presentes en el objetivo curado (venenos, quemaduras, etc.)
- Segunda: Potenciación de las estadísticas específicas de un compañero del equipo.
- Tercera: Curación especial. Efecto de área que cura una gran cantidad de vida a los integrantes del equipo.
- Cuarta: Utilización de objetos.

Los personajes enemigos (PE) a los que se enfrentará el jugador son los siguientes:

- Jefe final (ocupa 2 espacios de equipo)
 - Apariencia:
 - Habilidades:
 - Primera: Ataque físico. Un solo objetivo. Baja la resistencia física.
 - Segunda: Ataque mágico. Un solo objetivo. Baja la resistencia mágica.
 - Tercera: Ataque de área mágico. Baja el la fuerza física y mágica del grupo.
 - Cuarta: Invocación de esbirros (1 ó 2, según los huecos. El Jefe ocupa 2 huecos). Sube la defensa física y mágica del jefe final y la velocidad de los esbirros.
 - Estadísticas:

PV	40
PM	28
Fuerza física	8
Resistencia física	6
Poder mágico	7
Resistencia mágica	6
Agilidad	9

- Esbirro 1:
 - Apariencia:
 - Habilidades:
 - Estadísticas:

PV	14
PM	6
Fuerza física	6
Resistencia física	5
Poder mágico	1
Resistencia mágica	2
Agilidad	5

- Esbirro 2:
 - Apariencia:
 - Habilidades:
 - Estadísticas:

PV	9
PM	12
Fuerza física	2
Resistencia física	2
Poder mágico	6
Resistencia mágica	5
Agilidad	5

- Cultista 1:
 - Apariencia:
 - Habilidades:
 - Estadísticas:

PV	6
PM	2
Fuerza física	2
Resistencia física	3
Poder mágico	2
Resistencia mágica	1
Agilidad	3

- Cultista 2:
 - Apariencia:
 - Habilidades:
 - Estadísticas:

PV	4
PM	5
Fuerza física	1
Resistencia física	2
Poder mágico	3
Resistencia mágica	2
Agilidad	2

Los personajes de presentes en el pueblo con los que se relacionarán los PJs serán de tres “tipos”:

- Ciudadanos: Son personajes con los que solamente se mantienen diálogos. Servirán para dar contexto y ambientar la historia. Serán los habitantes del pueblo.
- Comerciantes: Los PJs podrán obtener diferentes items (objetos utilizables y/o equipo) de estos personajes. Estará en un puesto y al interactuar con él se desplegará un menú a través del cual, el jugador podrá comprar objetos para equipar a los PJs y objetos fungibles (pociones, comida, bebida, etc).
- Guía: Este personaje será único y planteará a los PJs una misión que deberán cumplir fuera de los límites del pueblo. Se presenta ante los PJs como un ángel, les informa de la amenaza que se cierne sobre el pueblo y sus habitantes y les solicita que investiguen en el exterior para encontrar una forma de eliminarla.

Descripción de las principales interfaces del juego:

- Interfaz de inicio: Al iniciar el juego se nos presenta una interfaz clásica de menú en la que habrá una imagen estática de fondo y varias opciones:
 - Iniciar partida.
 - Menú de opciones.
 - Salir del juego.
- Interfaz de opciones: en el que podremos modificar el sonido (control de volumen), velocidad del texto, brillo/contraste, etc.
- Interfaz de mapa: Al iniciar el juego los personajes aparecerán en escenario 2D que permitirá desarrollar actividades de exploración. En él se podrán desplazar en vertical/horizontal e interactuar con diferentes elementos del entorno.
- Interfaz de combate: En él se mostrará a los Personajes Jugables (PJ) combatiendo contra los enemigos por turnos y de forma controlada. Se mostrará una tarjeta para el personaje activo con detalles como el nombre, la vida, la energía, una miniatura de la apariencia del personaje y un menú de las cuatro opciones (habilidades) disponibles para ese personaje. Como con los protagonistas. Los equipos enemigos serán de 4 miembros como máximo.

- Interfaz de menú de opciones In-Game: Se mostrará un listado con distintas opciones: menú de GRUPO, GUARDAR, menú de OPCIONES del juego (sonido, brillo, etc) y SALIR del juego (cerrarlo).
- Interfaz del menú del Grupo. Se mostrará una ventana que estará dividida en dos secciones de forma vertical:
 - La sección izquierda de la pantalla contendrá el listado de los miembros del grupo. De cada miembro se mostrará la imagen de su avatar (como se ve en el mapa) junto a su nombre, puntos de vida y puntos de energía. Esta lista se podrá recorrer, utilizando un cursor para indicar cuál es el personaje que vamos a seleccionar. La imagen del PJ sobre el que esté el cursor se agrandará un poco para resaltar que ese es el personaje activo.
 - La sección derecha se dividirá a su vez en dos subsecciones de forma horizontal. Mostrará los datos a consultar/modificar siguiendo la siguiente estructura:
 - La subsección superior muestra una ficha con la información del PJ: se mostrarán nuevamente la cantidad de vida y energía que tenga el personaje en ese momento, una imagen mas detallada de la apariencia del personaje, sus estadísticas y un maniquí en el que se podrán cambiar los objetos de tipo equipo (arma principal, armadura, yelmo y perneras).
 - La subsección inferior muestra el Inventario: consistirá en una ventana con dos pestañas (una para el equipo y otra para los fungibles). Cada objeto dispondrá de distintas opciones de acción en función de en qué pestaña nos encontremos (los objetos de equipo se “equipan” y los objetos fungibles se “usan”).

El dinero será una información común, ya que el dinero es del grupo en general.

- Interfaz de Comercio. Se mostrará una ventana que estará dividida en dos secciones de forma vertical:
 - La sección izquierda mostrará un listado de los items presentes en el inventario del grupo.
 - La sección derecha mostrará una lista con los items que ofrece el vendedor.

El proceso de compra/venta se realizará de la siguiente manera: Utilizando las teclas direccionales izquierda/derecha nos desplazaremos entre la lista del grupo

y la del vendedor, aquella en la que nos encontremos será la lista activa y se mostrará resaltada con colores más brillantes. Utilizando las teclas direccionales arriba/abajo nos desplazaremos a lo largo de la lista activa. El objeto sobre el que nos encontremos será el objeto activo, la fila de la lista se expandirá mostrando las estadísticas del objeto y una pequeña descripción del mismo y pulsando la tecla Enter se realizará la acción de compra/venta.

Descripción de objetos

Los objetos tendrán diferentes tipos y subtipos en función de su uso y sus efectos:

- Tipo “Equipo”. Serán los objetos destinados a mejorar las estadísticas base de los personajes de cara al combate. Pueden ser equipados o desequipados del personaje:
 - Subtipo “Armamento”: el arma principal de cada PJ, que variará en función de a qué personaje esté destinado (Ej: el arco para la arquera, bastón para el mago o el sacerdote, etc).
 - Subtipo “Cabeza”: equipo que el PJ llevará puesto como indumentaria para la cabeza (cascos, coronas, capuchas, etc).
 - Subtipo “Cuerpo”: equipo que el PJ llevará puesto como indumentaria para el combate en la parte superior del cuerpo (armadura de placas, manto místico, etc).
 - Subtipo “Piernas/pies”: equipo que el PJ llevará puesto como indumentaria para el combate en la parte inferior del cuerpo (perneras de hierro, de cuero, etc).
- Tipo “Fungible”. Objetos que se consumen una sola vez y se gastan:
 - Subtipo “Pociones”: Objetos destinados a la recuperación de salud, energía y/o eliminación de estados perjudiciales que podrán ser usados durante el combate. Recuperan una cantidad limitada de puntos de vida o energía. Serán caros. (Ej: poción roja, poción azul, panacea, etc).
 - Subtipo “Comida/bebida”: Objetos destinados a la recuperación de salud, energía que solo podrán ser utilizados fuera del combate. Recuperan una gran cantidad de puntos de vida o energía. Serán baratos.

5.1.2 Diagramas de casos de uso relevantes.

Casos de uso de la Pantalla de Inicio

- El jugador se desplaza entre las opciones de la Pantalla de Inicio:
Utilizando las teclas direccionales (arriba/abajo) el jugador puede desplazarse entre las diferentes opciones del menú. La opción activa es aquella sobre la que se sitúa en cada momento y queda resaltada mediante un pequeño aumento del tamaño de la fuente de texto.
- El jugador Acepta la opción activa:
Pulsando la tecla "?" el jugador entra al submenú correspondiente a la Opción Activa
- El jugador Cancela la opción activa:
Pulsando la tecla "?" el jugador retrocede al menú anterior. Si el menú actual es el menú raíz se lanza una ventana emergente preguntando al usuario si desea salir del juego.
- El jugador selecciona la opción Nueva Partida:
Se lanza la Interfaz de Mapa para que el jugador pueda empezar a jugar.
- El jugador selecciona la opción Cargar Partida:
Se lanza la Interfaz de Mapa desde el último punto en el que se realizó un guardado de partida.
- El jugador selecciona la opción Menú de Configuración:
Se lanza la Interfaz del Menú de Opciones.
- El jugador selecciona la opción Salir del Juego:

Se lanza una ventana emergente pidiendo al jugador que confirme si realmente desea salir del juego. En caso de respuesta afirmativa, se cierra la aplicación.

Casos de uso de menú de Configuración

- El jugador se desplaza entre las opciones del Menú:
Utilizando las teclas direccionales (arriba/abajo) el jugador puede desplazarse entre las diferentes opciones del menú. La opción activa es aquella sobre la que se sitúa en cada momento y queda resaltada mediante un pequeño aumento del tamaño de la fuente del texto.
- El jugador modifica la opción activa:
Utilizando las teclas direccionales (izquierda/derecha) el jugador puede modificar el valor de la opción activa.
- El jugador Sale del Menú de Configuración:
Pulsando la tecla "?" el jugador retrocede al menú anterior. Si el jugador ha realizado cambios en la configuración, se lanza una ventana emergente preguntando si desea o no guardar los cambios y se retrocede al menú anterior.

Casos de uso del Menú de Opciones In-Game

- El jugador se desplaza entre las opciones del Menú:
Utilizando las teclas direccionales (arriba/abajo) el jugador puede desplazarse entre las diferentes opciones del menú. La opción activa es aquella sobre la que se sitúa en cada momento y queda resaltada mediante un pequeño aumento del tamaño de la fuente del texto.
- El jugador Acepta la opción activa:
Pulsando la tecla "?", el jugador entra al submenú correspondiente a la Opción Activa.

- El jugador Cancela la opción activa:
Pulsando la tecla "?" el jugador vuelve a la Interfaz de Mapa.
- El jugador selecciona la opción Menú de Grupo:
Se lanza la Interfaz del Menú de Grupo.
- El jugador selecciona la opción Guardar Partida:
Se almacena el estado del juego para, posteriormente, poder seleccionar la opción "Cargar Partida".
- El jugador selecciona la opción Cargar Partida:
Se carga el último estado del juego que se guardó. Se lanza la Interfaz de Mapa desde el último punto en el que se realizó un guardado de partida.
- El jugador selecciona la opción Menú de Configuración:
Se lanza la Interfaz de Menú de Configuración.
- El jugador selecciona la opción Salir del Juego:
Se lanza una ventana emergente pidiendo al jugador que confirme si realmente desea salir del juego. En caso de respuesta afirmativa, se cierra la aplicación.

Casos de uso de Menú de Grupo

- El jugador se desplaza entre los diferentes PJs:
Utilizando las teclas direccionales (arriba/abajo) el jugador puede desplazarse entre los diferentes personajes del grupo. El Pj activo es aquel sobre el que se sitúa en cada momento y queda resaltado mediante un pequeño aumento del tamaño de la miniatura del personaje.
- El jugador selecciona un PJ Activo.:

Pulsando la tecla "?", el jugador selecciona al PJ Activo pudiendo utilizar objetos sobre él. El control se desplaza a la parte inferior derecha de la interfaz situándose sobre el listado de los objetos de tipo Equipo.

- El jugador selecciona la pestaña de tipo de objetos que quiere usar:

Utilizando las teclas direccionales (derecha/izquierda) el jugador puede alternar entre las dos pestañas del inventario, la de objetos equipables o la de objetos consumibles. La Pestaña Activa es aquella sobre la que se sitúa en cada momento y queda resaltada mediante un pequeño aumento del tamaño de la fuente de texto del título la misma y/o mediante un marco de otro color.
- El jugador selecciona un Objeto Equipable para Equipar.

Utilizando las teclas direccionales (arriba/abajo) el jugador puede desplazarse entre los objetos Equipables. El Objeto Activo es aquel sobre el que se sitúa en cada momento y queda resaltado mediante un pequeño aumento del tamaño de la fuente de texto del mismo.

Pulsando la tecla "?", el jugador selecciona equipar el Objeto Activo, se lanza una ventana emergente que pide confirmación de la acción, si la respuesta es afirmativa el objeto es equipado: sustituye al objeto correspondiente de la figura de maniquí del PJ Activo y aplica las modificaciones sobre las estadísticas previstas para dicho objeto y dicho PJ. También queda indicado con una "E" bajo el nombre del objeto, quitando dicha letra del objeto que estaba equipado anteriormente.
- El jugador selecciona un Objeto Consumible para Consumir.

Utilizando las teclas direccionales (arriba/abajo) el jugador puede desplazarse entre los Objetos Consumibles. El Objeto Activo es aquel sobre el que se sitúa en cada momento y queda resaltado mediante un pequeño aumento del tamaño de la fuente de texto del mismo.

Pulsando la tecla "?", el jugador selecciona consumir el Objeto Activo, se lanza una ventana emergente que pide confirmación de la acción, si la respuesta es afirmativa el objeto es consumido: aplica sus efectos sobre el PJ Activo y se reduce en uno su número (se gasta) desapareciendo del listado del inventario en caso de que fuera la última unidad.
- El jugador Cancela el PJ activo:

Pulsando la tecla "?", el jugador vuelve a la sección de selección de PJ.

- El jugador selecciona la opción Continuar/Seguir Jugando.
Pulsando la tecla “?”, el jugador sale del Menú de Grupo y vuelve a la Interfaz de Mapa.

Casos de uso de mapa:

- El jugador se desplaza hacia arriba, abajo, derecha y/o izquierda
Pulsando las teclas direccionales (arriba/abajo/derecha/izquierda), el PJ se mueve en la dirección deseada. Sólo se puede mover en una dirección, es decir, no se puede mover de forma diagonal. Por tanto, en caso de que el jugador mantenga pulsada más de una tecla de dirección, el PJ irá únicamente en la dirección de la primera tecla que se haya pulsado. El resto de personajes del grupo sigue el “patrón” de movimiento del PJ que tenga delante.
- El jugador interacciona con un npc/elemento de la escena.
Pulsando la tecla “?”, el PJ interactúa con el npc/elemento que esté situado junto y frente a él. Se abre un cuadro de diálogo con la información que ese npc/elemento tiene para ofrecer o se abre el menú de Comercio, en caso de hablar con un npc de tipo “Mercader”.
- El jugador abre el Menú de Opciones In-Game.
Pulsando la tecla “Esc” el jugador abre el Menú de Opciones In-Game (“Dentro del Juego”), pausando la partida.
- El juego lanza un Trigger. Al situarse el PJ en un punto/zona en concreto se lanza un evento asociado a dicha zona del mapa. En el caso de ésta demo tan solo se contemplan dos tipos de evento:
 - X Evento de combate: Se inicia un combate presentando al jugador la interfaz de combate.
 - X Evento de transición pre-Jefe Final: tiene lugar una transición visual controlada por el sistema, teniendo el jugador únicamente el control de la sucesión de los diálogos hasta el final de dicha transición (pulsando la tecla “?” se pasa al siguiente diálogo hasta que la escena termine).

Casos de uso de combate:

- El jugador se desplaza entre las opciones del Menú:

Utilizando las teclas direccionales (arriba/abajo) el jugador puede desplazarse entre las diferentes opciones del menú. La opción activa es aquella sobre la que se sitúa en cada momento y queda resaltada mediante un pequeño aumento del tamaño de la fuente del texto.

- El jugador acepta la opción activa.

Pulsando la tecla “?”, el jugador selecciona la opción activa, pasando a la selección de PJ/PE/Objeto, en función de la opción seleccionada.
- El jugador cancela la opción activa.

Pulsando la tecla “?”, el jugador retrocede al sistema de selección/menú anterior. En caso de encontrarse en el menú raíz, no sucede nada.
- El jugador selecciona Habilidad del personaje de turno.

Pulsando la tecla “?”, el jugador selecciona la Habilidad activa. En caso de ser una habilidad mono-objetivo, se pasa a la selección de PE o de PJ objetivo, según el tipo de habilidad que sea y según el tipo de PJ al que pertenezca.

Tras elegir al PJ/PE objetivo, se ejecuta la habilidad en cuestión sobre dicho personaje (o personajes, en caso de Habilidades de Área) aplicando los efectos correspondientes (daño, estados, etc). Tras esto, termina el turno del PJ activo.
- El jugador selecciona la habilidad Usar Objeto.

En caso de seleccionar Usar Objeto, se abre una ventana emergente en la que el jugador puede seleccionar un consumible para utilizar (pociones). La opción activa es aquella sobre la que se sitúa en cada momento y queda resaltada mediante un pequeño aumento del tamaño de la fuente del texto. La forma de “consumo” es análoga a la del menú in-game.
- Un personaje ataca.

Cuando el personaje utiliza una habilidad física/mágica, se genera un número aleatorio entre 1 y “X”, al cuál se le suma la estadística correspondiente (fuerza física/mágica) del personaje.

Si dicha habilidad es una habilidad de área, se genera otro número adicional entre 1 y “X”, el cual también se suma.

Si es un ataque, dicho resultado es el daño que ese personaje le hace al(los) personaje(s) objetivo.

Si es una “sanación”, el personaje objetivo recupera una cantidad de puntos de salud igual al resultado obtenido.

Tras esto, se aplican los efectos correspondientes a esa habilidad.

- Un personaje (PJ/PE) recibe daño.

Cuando el personaje recibe un ataque físico/mágico, se genera un número aleatorio entre 1 y “X”, al cuál se le suma la estadística correspondiente (resistencia física/mágica) del personaje. Dicho resultado se le resta al daño recibido, y el valor obtenido son los puntos de salud que se le restan al PE.

- Un personaje obtiene un “estado”:

Tras realizar daño a un personaje, en función de la habilidad que se haya utilizado, ese personaje obtiene un estado perjudicial u otro. El resultado obtenido corresponde al número de turnos durante los que ese personaje tendrá asignado el estado.

Los estados y sus efectos son:

- Una estadística es aumentada/reducida

En función de la Habilidad utilizada, una o varias estadísticas pueden aumentar/reducirse hasta el final del combate en un “X”% o “Y”%. Dicho porcentaje siempre se realizará en función al valor base de esa estadística hasta un máximo del 200%.

- El juego determina que el combate ha terminado:

Al finalizar cada turno, se comprueba la cantidad de personajes “vivos” de cada equipo (o del equipo afectado). Si esta es igual a 0, se da por finalizado el combate y se vuelve a la interfaz de mapa.

- El grupo pierde el combate:

Si al finalizar el combate, el grupo derrotado es el de los PJs, se abre una ventana emergente preguntando si se desea cargar la partida desde el último punto de guardado. Si la respuesta es negativa, se abre la interfaz de mapa en el pueblo. Los objetos utilizados no se recuperan.

Casos de uso de comercio:

- El jugador selecciona Lista de Compra o Lista de Venta.

Usando las teclas direccionales (izquierda/derecha), el jugador puede alternar entre las dos listas de la Interfaz de Comercio (compra o venta). La lista activa queda resaltada mediante un marco de un color llamativo, mientras que la otra lista pierde "brillo".
- El jugador selecciona un Objeto de la Lista Activa.

Utilizando las teclas direccionales (arriba/abajo) el jugador puede desplazarse entre los Objetos de la Lista Activa. El Objeto Activo es aquel sobre el que se sitúa en cada momento y queda resaltado mediante un pequeño aumento del tamaño de la fuente de texto del mismo.
- El jugador selecciona confirmar Operación Compra/Venta.

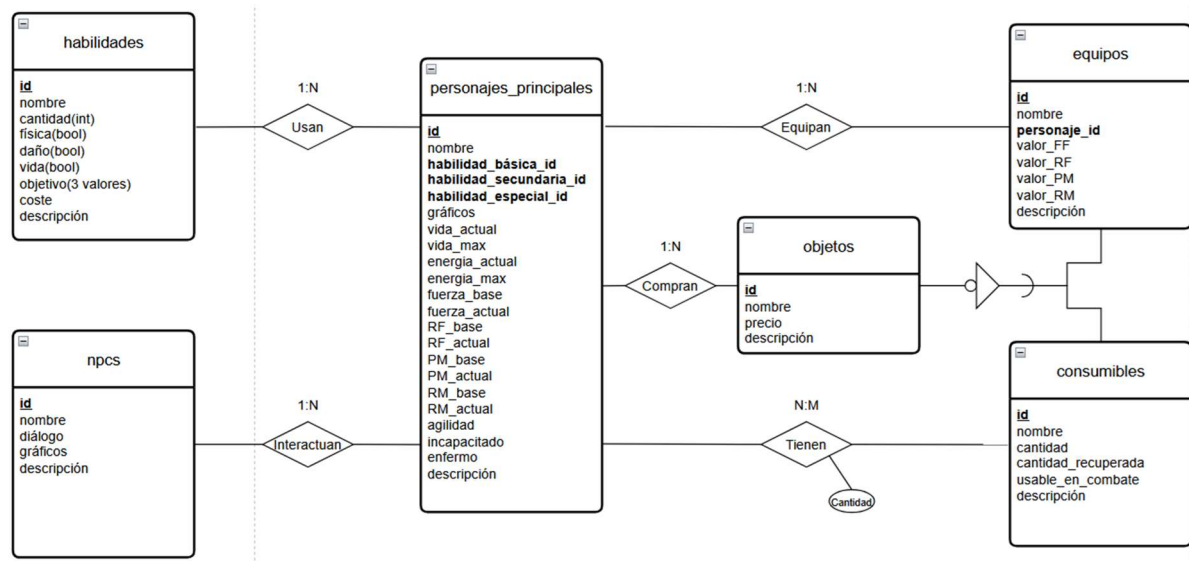
Pulsando la tecla "?", el jugador selecciona el Objeto Activo. Se abre una ventana emergente para confirmar la Compra/Venta. Si la respuesta es afirmativa, el Objeto activo cambia de Lista, así como su precio.

Cada objeto tiene un precio fijo. Si dicho objeto se encuentra en la Lista de Compra, no se modifica. Pero si, por el contrario, se encuentra en la Lista de Venta, su precio se reduce al "X"%.
- El jugador selecciona Salir/Terminar.

Pulsando la tecla "?", el jugador sale del Menú de Comercio y vuelve a la Interfaz de Mapa.

6 Diseño.

6.1 Diseño Conceptual Entidad Relación.



6.2 Diseño Lógico Relacional o Paso a tablas.

habilidades (id, nombre, cantidad(int), física(bool), daño(bool), vida(bool), objetivo (3 valores), coste, descripción)

personajes_principales (id, nombre, **hab_basica_id**, **hab_secundaria_id**, **hab_especial_id**, **usar_objeto_id**, gráficos, vida_actual, vida_max, energia_actual, energia_max, fuerza_base, fuerza_actual, RF_base, RF_actual, PM_base, PM_actual, RM_base, RM_actual, agilidad, incapacitado, enfermo, descripción)

npcs (id, nombre, diálogo, gráficos, descripción)

objetos (id, nombre, precio, descripción)

equipos (id, nombre, **personaje_id**, valor_FF, valor_RF, valor_PM, valor_RM, descripción)

consumibles (id, nombre, cantidad, cantidad_recuperada, usable_en_combate, descripción)

AUXILIARES:

~~estadísticas (id, nombre, descripción) (creo que "valor" queda mejor en la tabla de la relación "personajes"-"estadísticas" ("poseen")), porque si la dejamos aquí van a haber muchos valores repetidos: van a tener los mismos valores 9 veces (personajes) por cada estadística}~~

RELACIONES:

tienen_consumibles (id, personaje_id, consumible_id, cantidad)

6.3 Diseño Físico (paso a tablas, optimizaciones).

-- Tabla: habilidades

```
CREATE TABLE habilidades (  
id INT PRIMARY KEY AUTO_INCREMENT,  
nombre VARCHAR(100) NOT NULL,  
cantidad INT DEFAULT 0,  
fisica BOOLEAN DEFAULT true,  
dano BOOLEAN DEFAULT true,  
vida BOOLEAN DEFAULT true,  
objetivo ENUM('mono', 'area', 'auto') NOT NULL,  
coste INT DEFAULT 0,  
descripcion TEXT NOT NULL  
);
```

-- Tabla: personajes_principales

```
CREATE TABLE personajes_principales (  
id INT PRIMARY KEY AUTO_INCREMENT,  
nombre VARCHAR(100) NOT NULL,  
hab_basica_id INT NOT NULL,  
hab_secundaria_id INT,  
hab_especial_id INT,  
usar_objeto_id INT,
```

```

graficos VARCHAR(255) NOT NULL,

vida_max INT NOT NULL CHECK (vida_max >= 1),

vida_actual INT NOT NULL CHECK (vida_actual >= 0 AND vida_actual <= vida_max),

energia_max INT NOT NULL CHECK (energia_max >= 1),

energia_actual INT NOT NULL CHECK (energia_actual >= 0 AND energia_actual <= energia_max),

fuerza_base INT NOT NULL CHECK (fuerza_base >= 1),

fuerza_actual INT NOT NULL CHECK (fuerza_actual >= 0 AND fuerza_actual <= fuerza_base),

RF_base INT NOT NULL CHECK (RF_base >= 1),

RF_actual INT NOT NULL CHECK (RF_actual >= 0 AND RF_actual <= RF_base),

PM_base INT NOT NULL CHECK (PM_base >= 1),

PM_actual INT NOT NULL CHECK (PM_actual >= 0 AND PM_actual <= PM_base),

RM_base INT NOT NULL CHECK (RM_base >= 1),

RM_actual INT NOT NULL CHECK (RM_actual >= 0 AND RM_actual <= RM_base),

agilidad INT NOT NULL CHECK (agilidad >= 1),

aggro INT DEFAULT 1 CHECK (aggro = 1),

incapacitado BOOLEAN DEFAULT false,

enfermo BOOLEAN DEFAULT false,

descripcion TEXT NOT NULL,

FOREIGN KEY (hab_basica_id) REFERENCES habilidades(id),

FOREIGN KEY (hab_secundaria_id) REFERENCES habilidades(id),

FOREIGN KEY (hab_especial_id) REFERENCES habilidades(id),

FOREIGN KEY (usar_objeto_id) REFERENCES habilidades(id)

);

```

-- Tabla: npcs

```

CREATE TABLE npcs (

id INT PRIMARY KEY AUTO_INCREMENT,

nombre VARCHAR(100) NOT NULL,

dialogo TEXT NOT NULL,

graficos VARCHAR(255) NOT NULL,

```

descripcion TEXT NOT NULL

);

-- Tabla: objetos

CREATE TABLE objetos (

id INT PRIMARY KEY AUTO_INCREMENT,

nombre VARCHAR(100) NOT NULL,

precio INT NOT NULL CHECK (precio >= 10),

descripcion TEXT NOT NULL

);

-- Tabla: equipos

CREATE TABLE equipos (

id INT PRIMARY KEY AUTO_INCREMENT,

nombre VARCHAR(100) NOT NULL,

personaje_id INT,

comprado BOOLEAN DEFAULT false, --

valor_FF INT NOT NULL CHECK ((valor_RF = 0 AND valor_PM = 0 AND valor_RM = 0 AND valor_FF >= 1) OR valor_FF >= 0),

valor_RF INT NOT NULL CHECK ((valor_FF = 0 AND valor_PM = 0 AND valor_RM = 0 AND valor_RF >= 1) OR valor_RF >= 0),

valor_PM INT NOT NULL CHECK ((valor_RF = 0 AND valor_FF = 0 AND valor_RM = 0 AND valor_PM >= 1) OR valor_PM >= 0),

valor_RM INT NOT NULL CHECK ((valor_RF = 0 AND valor_PM = 0 AND valor_FF = 0 AND valor_RM >= 1) OR valor_RM >= 0),

descripcion TEXT NOT NULL,

FOREIGN KEY (personaje_id) REFERENCES personajes_principales(id)

);

-- Tabla: consumibles

CREATE TABLE consumibles (

id INT PRIMARY KEY AUTO_INCREMENT,

```
nombre VARCHAR(100) NOT NULL,  
cantidad INT CHECK (cantidad >= 0),  
cantidad_recuperada INT CHECK (cantidad_recuperada >= 10),  
usable_en_combate BOOLEAN DEFAULT true,  
descripcion TEXT NOT NULL  
);
```

-- Tabla: tienen_consumibles (relación personaje-consumible-cantidad)

-- Si el campo "cantidad" está en la tabla "consumibles", esta tabla es innecesaria y/o redundante

```
CREATE TABLE tienen_consumibles (  
id INT PRIMARY KEY AUTO_INCREMENT,  
personaje_id INT NOT NULL,  
consumible_id INT NOT NULL,  
cantidad INT CHECK (cantidad >= 0),  
FOREIGN KEY (personaje_id) REFERENCES personajes_principales(id),  
FOREIGN KEY (consumible_id) REFERENCES consumibles(id)  
);
```

6.4 Descripción de las tablas y campos

HABILIDADES

Campo	Tipo de Dato	Restricciones	Descripción
id	INT	PRIMARY KEY, AUTO_INCREMENT	Identificador único de la habilidad.
nombre	VARCHAR(100)	NOT NULL	Nombre de la habilidad.
cantidad	INT	DEFAULT 0	Cantidad de veces que se genera un número aleatorio cuya suma será la salud afectada.
fisica	BOOLEAN	DEFAULT true	Indica si la habilidad afecta o no a la salud/energía.
dano	BOOLEAN	DEFAULT true	Indica si la habilidad resta o suma salud/energía.
vida	BOOLEAN	DEFAULT true	Indica si la habilidad afecta a la vida o a la energía.
objetivo	ENUM	NOT NULL ('mono', 'area', 'auto')	Determina el tipo de objetivo de la habilidad (único, múltiple o a sí mismo).
coste	INT	DEFAULT 0	Coste de energía que la habilidad necesita para poder ser utilizada..
descripcion	TEXT	NOT NULL	Descripción narrativa o funcional de la habilidad.

PERSONAJES_PRINCIPALES

Campo	Tipo de Dato	Restricciones	Descripción
id	INT	PRIMARY KEY, AUTO_INCREMENT	Identificador único del personaje.
nombre	VAR-CHAR(100)	NOT NULL	Nombre del personaje.
hab_basica_id	INT	NOT NULL, FOREIGN KEY → habilidades(id)	ID de la habilidad básica.
hab_secundaria_id	INT	FOREIGN KEY → habilidades(id)	ID de la habilidad secundaria.
hab_especial_id	INT	FOREIGN KEY → habilidades(id)	ID de la habilidad especial.
usar_objeto_id	INT	FOREIGN KEY → habilidades(id)	ID de la habilidad para usar objetos.
graficos	VAR-CHAR(255)	NOT NULL	Ruta o referencia gráfica del personaje (sprites).
vida_max	INT	NOT NULL, CHECK (≥ 1)	Vida máxima del personaje.
vida_actual	INT	NOT NULL, CHECK (≥ 0 y \leq vida_max)	Vida actual del personaje.
energia_max	INT	NOT NULL, CHECK (≥ 1)	Energía máxima del personaje.
energia_actual	INT	NOT NULL, CHECK (≥ 0 y \leq energia_max)	Energía actual del personaje.
fuerza_base	INT	NOT NULL, CHECK (≥ 1)	Valor base de la fuerza física.
fuerza_actual	INT	NOT NULL, CHECK (≥ 0 y \leq fuerza_base)	Valor actual de la fuerza física.
RF_base	INT	NOT NULL, CHECK (≥ 1)	Valor base de la resistencia física.

RF_actual	INT	NOT NULL, CHECK (≥ 0 y \leq RF_base)	Valor base de la resistencia física.
PM_base	INT	NOT NULL, CHECK (≥ 1)	Valor base de la potencia mágica.
PM_actual	INT	NOT NULL, CHECK (≥ 0 y \leq PM_base)	Valor base de la potencia mágica.
RM_base	INT	NOT NULL, CHECK (≥ 1)	Valor base de la resistencia mágica.
RM_actual	INT	NOT NULL, CHECK (≥ 0 y \leq RM_base)	Valor base de la resistencia mágica.
agilidad	INT	NOT NULL, CHECK (≥ 1)	Valor de la agilidad.
aggro	INT	DEFAULT 1, CHECK (≥ 1)	Valor del “aggro” (atención que un personaje tiene sobre los enemigos).
incapacitado	BOOLEAN	DEFAULT false	Estado de “incapacidad” (no puede actuar).
enfermo	BOOLEAN	DEFAULT false	Estado de “enfermedad” (el personaje pierde salud gradualmente).
descripcion	TEXT	NOT NULL	Descripción narrativa del personaje.

NPCS

Campo	Tipo de Dato	Restricciones	Descripción
id	INT	PRIMARY KEY, AUTO_INCREMENT	Identificador único del NPC.
nombre	VAR-CHAR(100)	NOT NULL	Nombre del NPC.

dialogo	TEXT	NOT NULL	Diálogo del NPC cuando se interactúa con él/ella.
graficos	VAR-CHAR(255)	NOT NULL	Ruta o referencia gráfica del personaje (sprites).
descripcion	TEXT	NOT NULL	Descripción narrativa del NPC.

EQUIPOS

Campo	Tipo de Dato	Restricciones	Descripción
id	INT	PRIMARY KEY, FOREIGN KEY → objetos(id)	Identificador único del equipo.
nombre	VAR-CHAR(100)	NOT NULL	Nombre del equipo.
personaje_id	INT	FOREIGN KEY → personajes_principales(id)	Personaje que tiene equipado el objeto (si aplica).
comprado	BOOLEAN	DEFAULT false	Indica si ha sido adquirido y debe mostrarse en el inventario.
valor_FF	INT	NOT NULL, CHECK complejo (ver SQL original)	Bonificación en fuerza física.
valor_RF	INT	NOT NULL, CHECK complejo (ver SQL original)	Bonificación en resistencia física.
valor_PM	INT	NOT NULL, CHECK complejo (ver SQL original)	Bonificación en poder mágico.
valor_RM	INT	NOT NULL, CHECK complejo (ver SQL original)	Bonificación en resistencia mágica.
descripcion	TEXT	NOT NULL	Descripción del objeto equipado.

CONSUMIBLES

Campo	Tipo de Dato	Restricciones	Descripción
id	INT	PRIMARY KEY, FOREIGN KEY → objetos(id)	Identificador único del consumible.
nombre	VAR-CHAR(100)	NOT NULL	Nombre del consumible.
cantidad_recuperada	INT	CHECK (≥ 10)	Valor que se recupera al usarlo (vida/energía).
usable_en_combate	BOOLEAN	DEFAULT true	Indica si el objeto puede usarse durante un combate.
descripcion	TEXT	NOT NULL	Descripción narrativa o funcional del consumible.

6.5 Orientación a objetos:

6.5.1 Diagramas de clases. Descripción de clases y atributos.

```
public class Habilidad
{
    public int Id { get; set; }
    public string Nombre { get; set; }
    public int Cantidad { get; set; } = 0;
    public bool Fisica { get; set; } = true;
    public bool Dano { get; set; } = true;
    public bool Vida { get; set; } = true;
```

```
public string Objetivo { get; set; } // "mono", "area", "auto"  
public int Coste { get; set; } = 0;  
public string Descripcion { get; set; }  
}
```

```
public class PersonajePrincipal  
{  
    public int Id { get; set; }  
    public string Nombre { get; set; }  
    public int HabBasicald { get; set; }  
    public int? HabSecundariald { get; set; }  
    public int? HabEspecialld { get; set; }  
    public int? UsarObjetold { get; set; }  
    public string Graficos { get; set; }
```

```
    public int VidaMax { get; set; }  
    public int VidaActual { get; set; }  
    public int EnergiaMax { get; set; }  
    public int EnergiaActual { get; set; }
```

```
    public int FuerzaBase { get; set; }  
    public int FuerzaActual { get; set; }
```

```
    public int RFBase { get; set; }  
    public int RFActual { get; set; }
```

```
    public int PMBase { get; set; }  
    public int PMActual { get; set; }
```

```
    public int RMBase { get; set; }  
    public int RMActual { get; set; }
```

```
public int Agilidad { get; set; }  
public int Aggro { get; set; } = 1;  
public bool Incapacitado { get; set; } = false;  
public bool Enfermo { get; set; } = false;  
public string Descripcion { get; set; }
```

```
public Habilidad HabBasica { get; set; }  
public Habilidad HabSecundaria { get; set; }  
public Habilidad HabEspecial { get; set; }  
public Habilidad UsarObjeto { get; set; }  
}
```

```
public class Npc  
{  
    public int Id { get; set; }  
    public string Nombre { get; set; }  
    public string Dialogo { get; set; }  
    public string Graficos { get; set; }  
    public string Descripcion { get; set; }  
}
```

```
public class Objeto  
{  
    public int Id { get; set; }  
    public string Nombre { get; set; }  
    public int Precio { get; set; }  
    public string Descripcion { get; set; }  
}
```

```
public class Equipo
```

```

{
    public int Id { get; set; }
    public string Nombre { get; set; }
    public int? PersonajeId { get; set; }
    public bool Comprado { get; set; } = false;

    public int ValorFF { get; set; }
    public int ValorRF { get; set; }
    public int ValorPM { get; set; }
    public int ValorRM { get; set; }
    public string Descripcion { get; set; }

    public PersonajePrincipal Personaje { get; set; }
}

public class Consumible
{
    public int Id { get; set; }
    public string Nombre { get; set; }
    public int Cantidad { get; set; }
    public int CantidadRecuperada { get; set; }
    public bool UsableEnCombate { get; set; } = true
    public string Descripcion { get; set; }
}

public class TienenConsumible
{
    public int Id { get; set; }
    public int PersonajeId { get; set; }
    public int ConsumibleId { get; set; }
    public int Cantidad { get; set; }
}

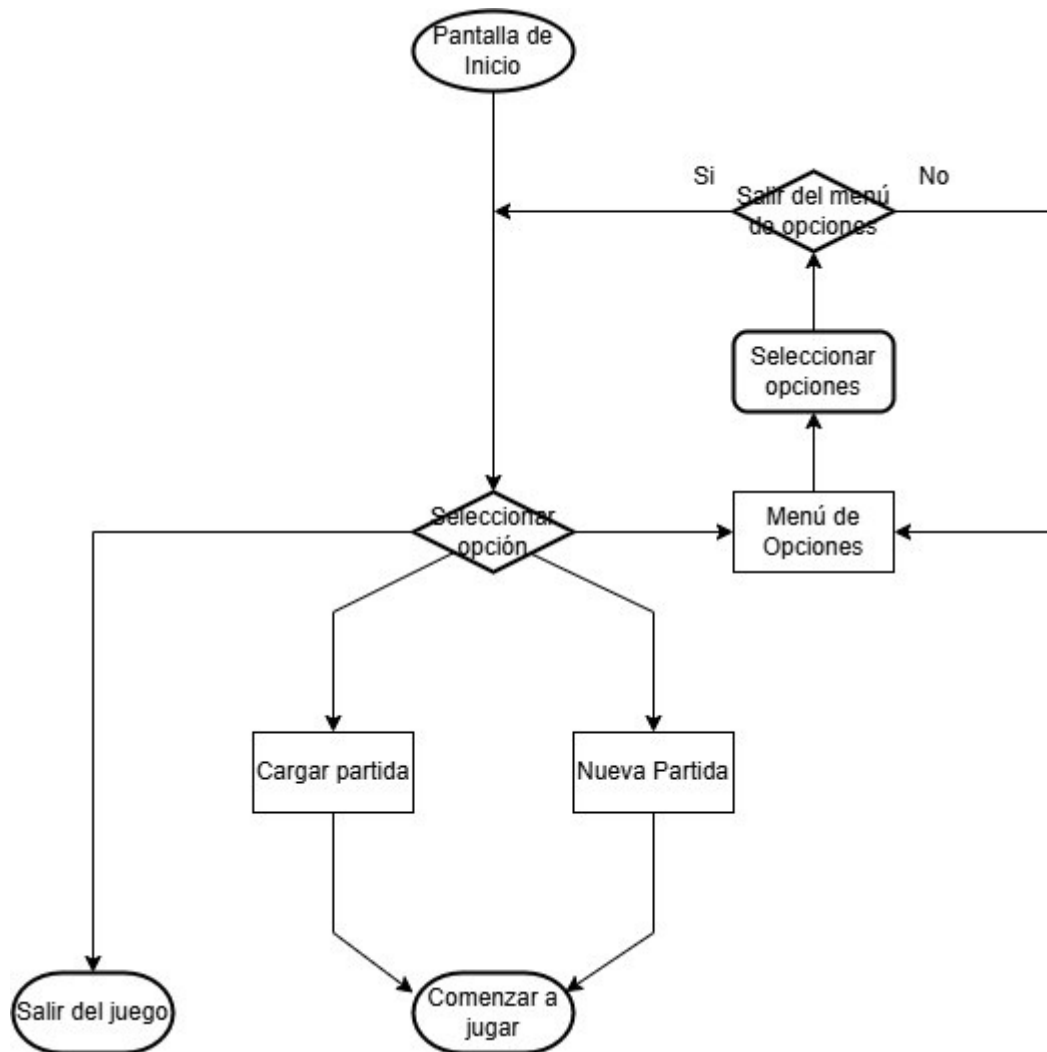
```

```
public PersonajePrincipal Personaje { get; set; }  
public Consumible Consumible { get; set; }  
}
```

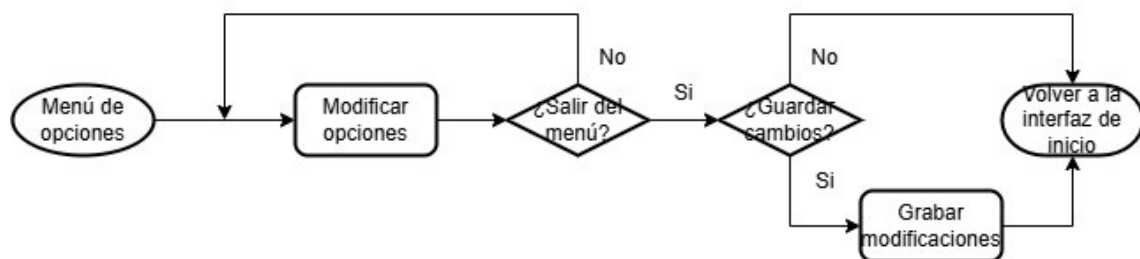
6.5.2 Diagrama de secuencias. De lo más relevante.

6.5.3 Diagrama de actividad. De lo más relevante.

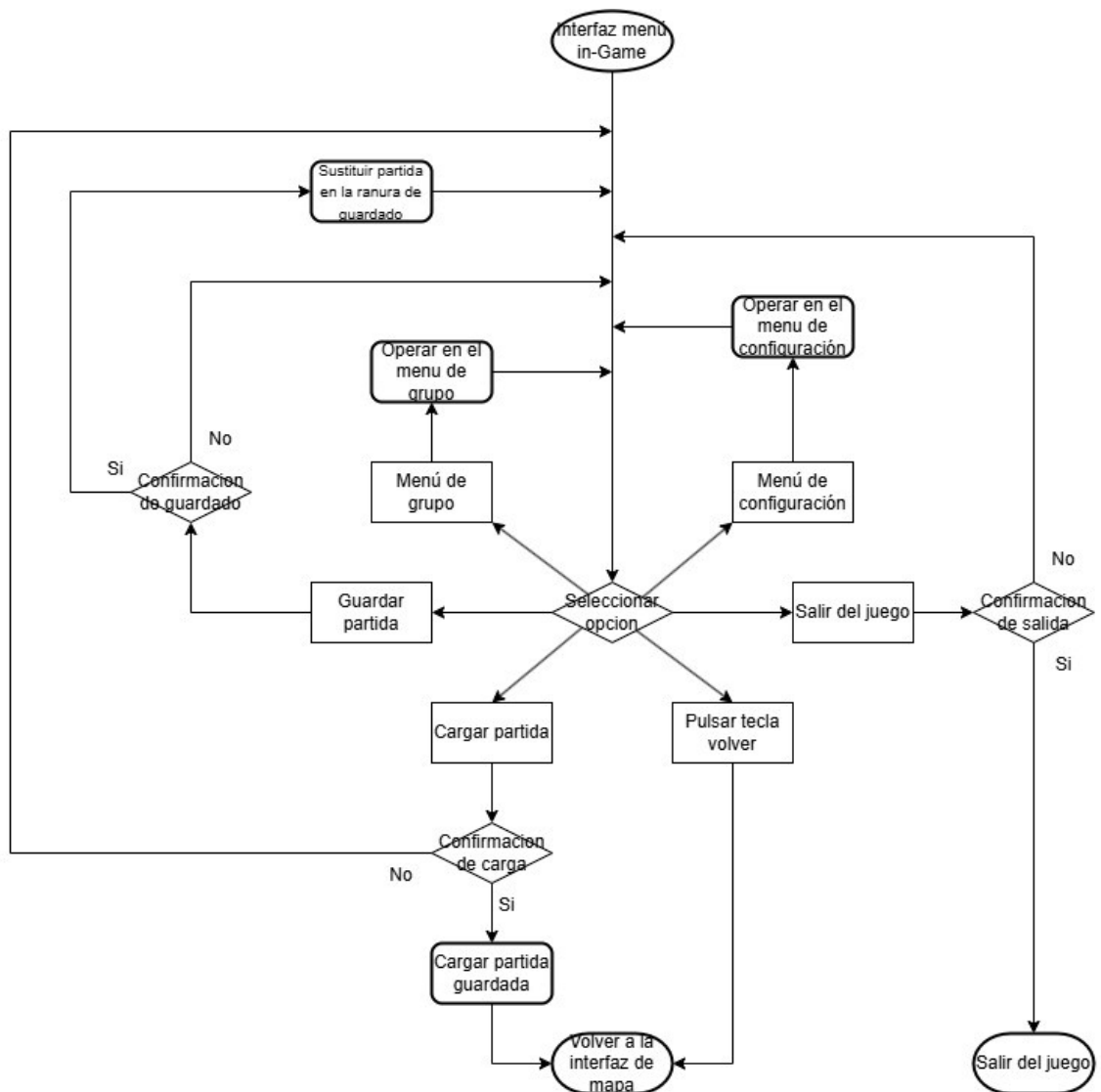
Caso interfaz inicial:



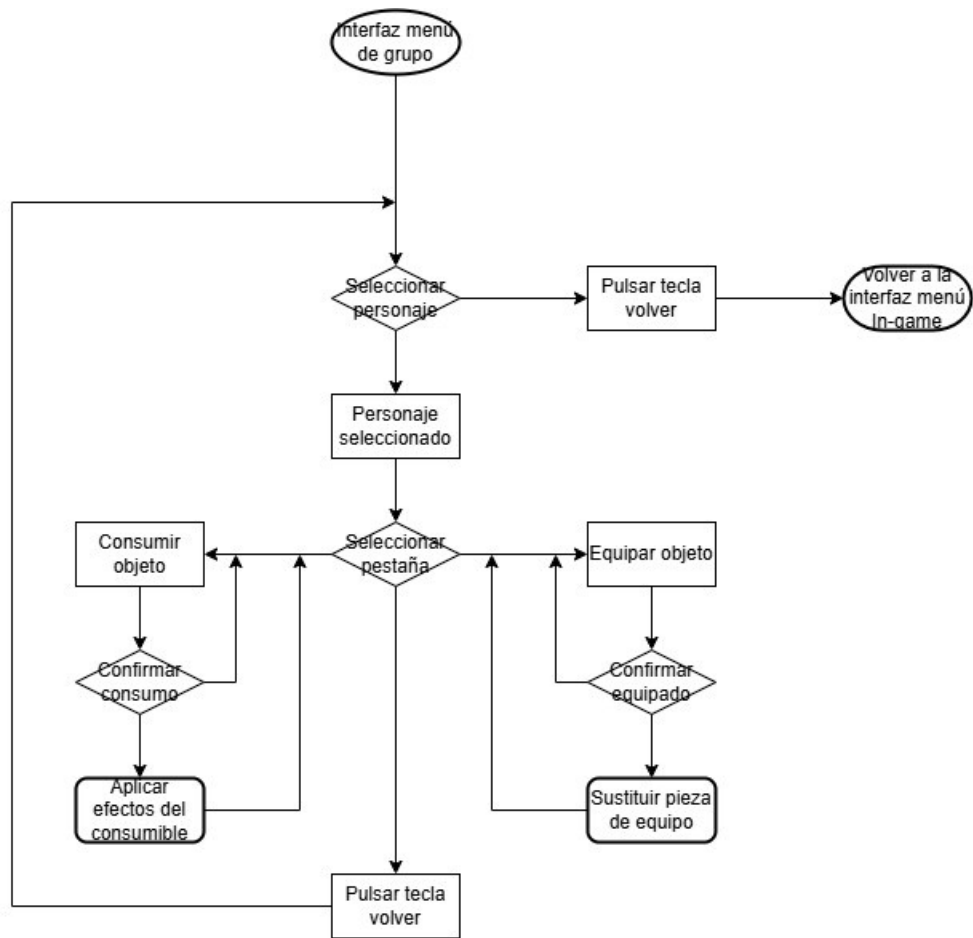
Caso menú de opciones:



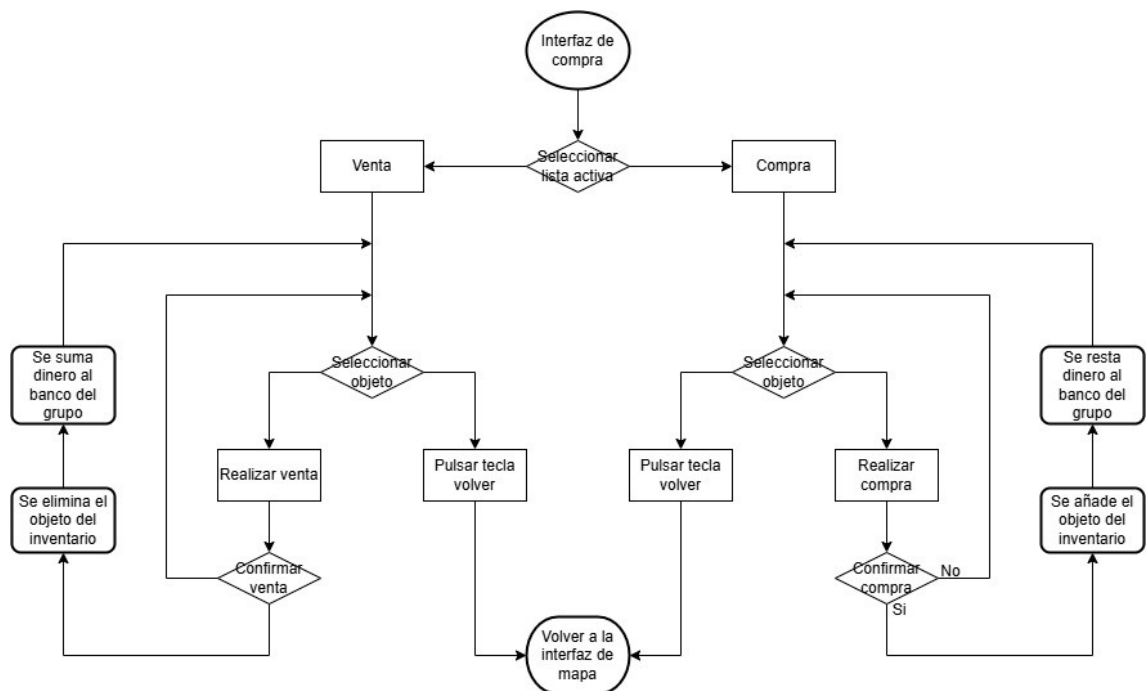
Caso menú in-game:



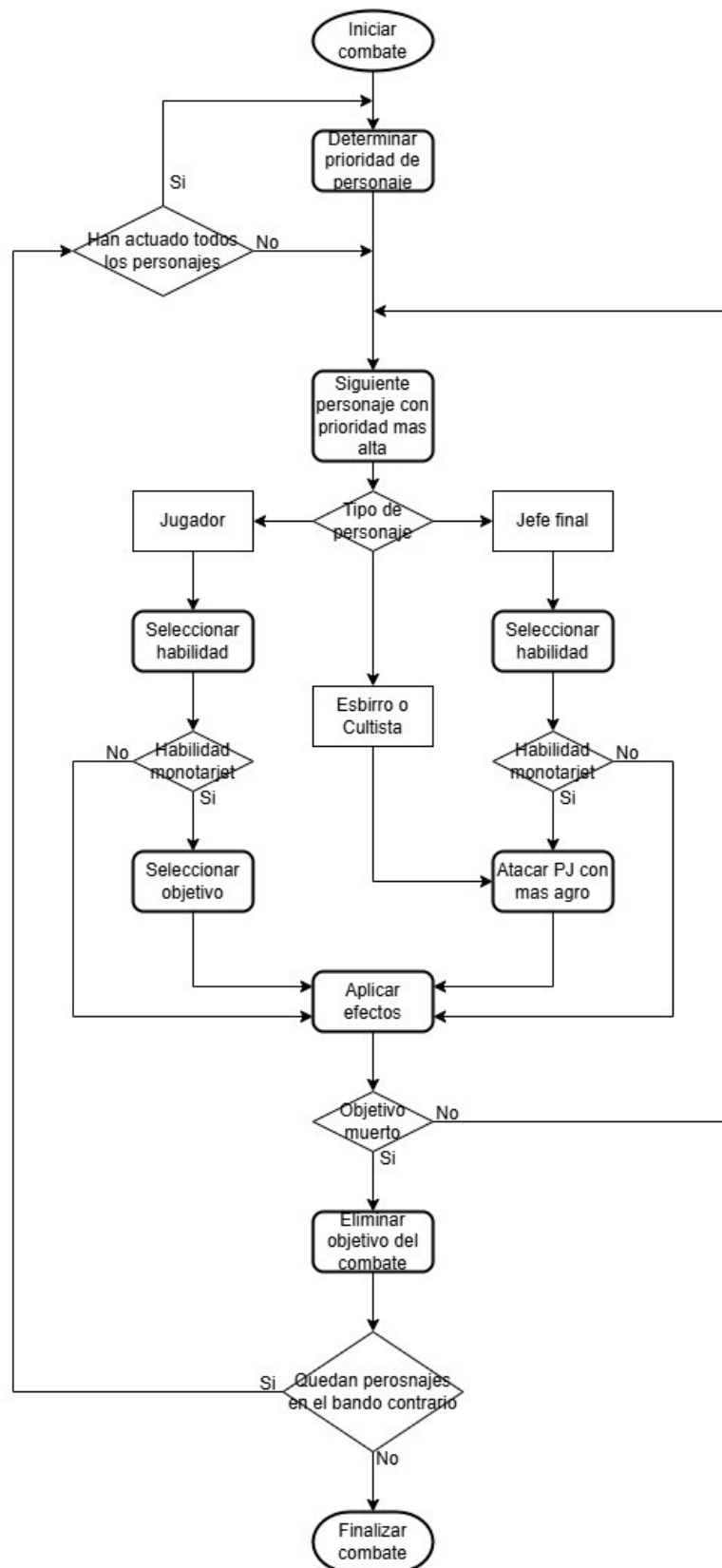
Caso menú de grupo:



Caso menú de ventas:

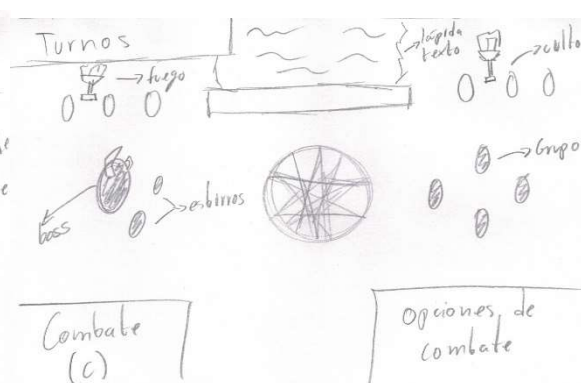
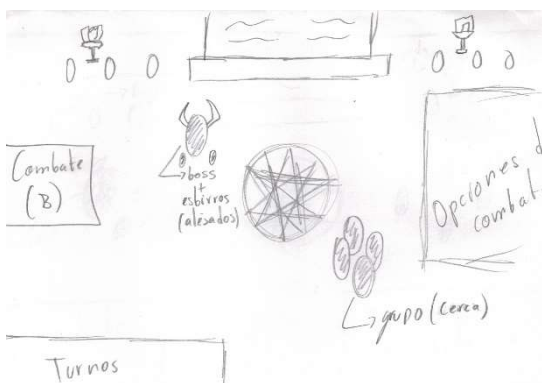
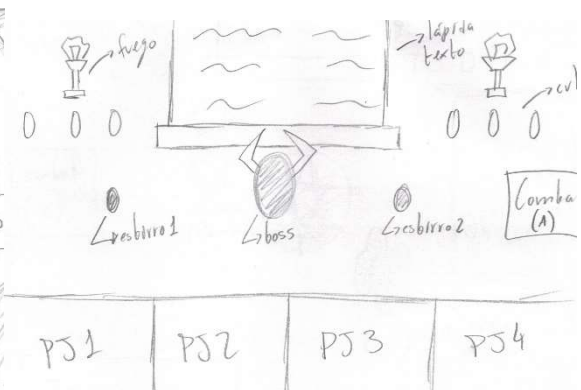
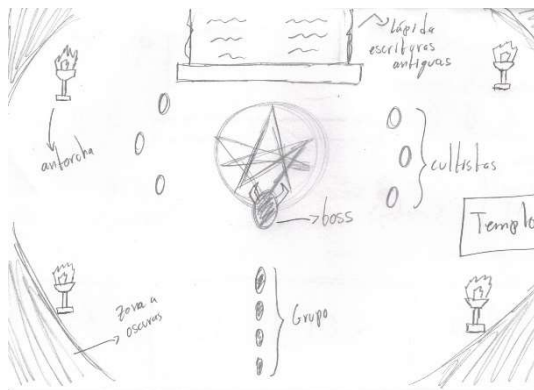
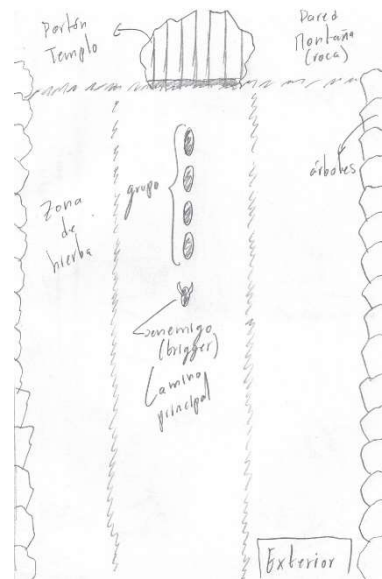
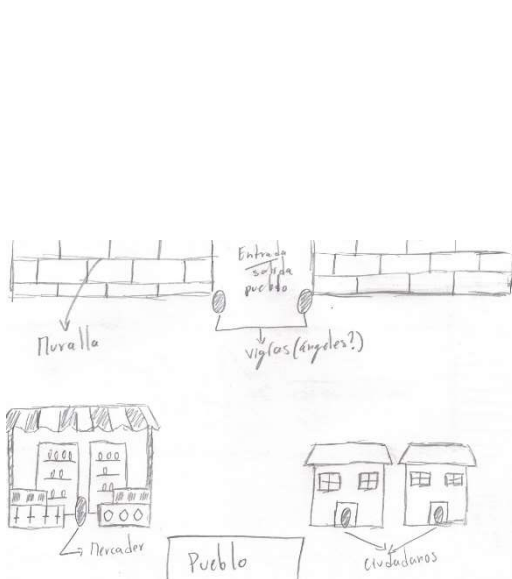
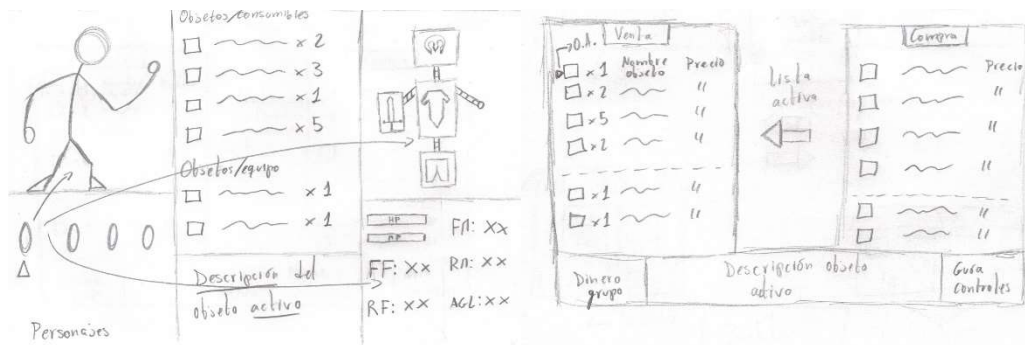


Caso combate:



6.6 Diseño UX.

6.7 Mockups (Penpot, Figma)



7 Codificación.

7.1 Tecnologías elegidas y su justificación (lenguajes, frameworks, librerías, etc.)

7.1.1 Desarrollo de servicios.

7.1.1.1 Descripción general.

7.1.1.2 Seguridad.

7.1.2 Desarrollo multiplataforma.

7.1.2.1 Descripción general.

7.1.2.2 Asegurar la funcionalidad en distintos dispositivos.

7.2 Documentación interna de código (puede ir en un anexo).

7.2.1 Descripción de cada fichero. Autor, función y fecha de creación

7.2.2 Descripción de cada función. Autor, función y fecha de creación.

7.2.3 Documentación externa (puede ir en un anexo).

7.2.4 Manual del usuario.

8 Despliegue.

8.1 Diagramas de despliegue

8.2 Descripción de la instalación o despliegue

9 Herramientas de apoyo.

10 Control de versiones.

11 Sistemas de integración continua.

12 Gestión de pruebas.

13 Conclusiones.

13.1 Conclusiones sobre el trabajo realizado

13.2 Conclusiones personales

13.3 Posibles ampliaciones y mejoras

14 Bibliografía

15 Libros, artículos y apuntes.

16 Direcciones web.