

**Desarrollo de un videojuego 2d tipo roguelike con temática medieval, para adolescentes de entre 13 a 25 años, para ayudar a personas con tdah (trastorno de déficit de atención e hiperactividad) a fomentar la concentración y atención sostenida.**

Martínez Daza Sergio Daniel

Reyes Parra Jonathan David

UNIVERSITARIA UNIAGUSTNIANA

Tecnología en Desarrollo de Software

BOGOTÁ DC

2023

**Desarrollo de un videojuego 2d tipo roguelike con temática medieval, para adolescentes de entre 13 a 25 años, para ayudar a personas con tdah (trastorno de déficit de atención e hiperactividad) a fomentar la concentración y atención sostenida.**

Martínez Daza Sergio Daniel

Reyes Parra Jonathan David

Trabajo de grado para optar al título de Tecnólogo en Desarrollo de Software

UNIVERSITARIA UNIAGUSTNIANA

Tecnología en Desarrollo de Software

BOGOTÁ DC

2023

## **Resumen**

“Becoming to Berserk” es un innovador videojuego de estilo roguelike específicamente diseñado para ayudar a jóvenes de entre 13 y 20 años que padecen Trastorno por Déficit de Atención e Hiperactividad (TDAH). Este proyecto busca proporcionar una experiencia de juego que, además de ser entretenida, ayude a mejorar las habilidades de concentración y organización de sus jugadores. El juego será desarrollado utilizando el avanzado motor de desarrollo de videojuegos GameMaker, conocido por su capacidad para crear juegos de alta calidad. Además, estará completamente en idioma español, facilitando su accesibilidad para la audiencia hispanohablante. Se ejecutará exclusivamente en la plataforma Windows 10 o versiones posteriores, garantizando su compatibilidad con la mayoría de los sistemas actuales. “Becoming to Berserk” incluirá una emocionante mecánica de combate en tiempo real, niveles interactivos cuidadosamente diseñados, y una amplia variedad de equipamiento que los jugadores podrán utilizar. La interfaz de usuario será intuitiva y fácil de navegar, asegurando que incluso los jugadores más jóvenes puedan disfrutar del juego sin dificultades. Dado su estilo roguelike, no se implementará un sistema de guardado de partidas, lo que añadirá un nivel adicional de desafío y emoción al juego.

*Palabras Clave: Videojuego, GameMaker, TDAH, Roguelike*

## **Abstract**

"Becoming to Berserk" is an innovative roguelike video game specifically designed to assist young people aged 13 to 20 who suffer from Attention Deficit Hyperactivity Disorder (ADHD). This project aims to provide a gaming experience that, in addition to being entertaining, helps improve the concentration and organizational skills of its players. The game will be developed using the advanced GameMaker game development engine, known for its ability to create high-quality games. Furthermore, it will be entirely in Spanish, making it accessible to Spanish-speaking audiences. It will run exclusively on the Windows 10 platform or later versions, ensuring compatibility with most current systems. "Becoming to Berserk" will feature an exciting real-time combat mechanic, carefully designed interactive levels, and a wide variety of equipment that players can utilize. The user interface will be intuitive and easy to navigate, ensuring that even younger players can enjoy the game without difficulties. Given its roguelike style, a save game system will not be implemented, adding an additional level of challenge and excitement to the game.

*Keywords: Video game, GameMaker, ADHD, Roguelike*

## Tabla de contenido

Tabla de contenido .....	5
Introducción .....	7
1 Título del proyecto .....	8
2 Planeación del proyecto .....	8
2.1 Objetivos del proyecto.....	8
2.1.1 Objetivo general .....	8
2.1.2 Objetivos específicos .....	8
2.2 Planteamientos del problema y/o necesidad .....	9
2.3 Alcance del proyecto .....	10
2.4 Metodología de desarrollo de software .....	13
3 Marco Teórico y Estado del Arte. ....	18
3.1 Marco teórico .....	18
3.2 Estado del arte.....	20
4 Especificación de requisitos de software (IEEE 830) .....	23
4.1 Perspectiva del producto.....	23
4.2 Funcionalidad del producto .....	23
4.3 Características de los usuarios .....	24
4.4 Restricciones .....	24
4.5 Dependencias y suposiciones .....	25
4.6 Requisitos Específicos.....	25
4.6.1 Actores y Roles.....	25
4.6.2 Requisitos Funcionales .....	26
4.6.3 Diagrama de caso de uso.....	27
4.6.4 Especificaciones de caso de uso .....	28
4.7 Requisitos de Rendimiento .....	30
4.8 Restricciones de Diseño .....	31
4.9 Atributos del Software del Sistema.....	32
5. Diseño del software (ISO -12207-1) .....	34
5.1 Diseño de la Arquitectura de software .....	35
5.2 Diseño detallado del software.....	36

5.2.1 Diagrama de clases .....	37
5.2.2 Diagrama de paquetes .....	38
5.2.3 Diagrama de despliegue .....	39
5.3 Diseño de la Interfaz .....	39
5.3.1 Interfaz Gráfica de Usuario .....	40
5.3.2 Interfaces de Entrada .....	42
5.3.3 Interfaces de Salida .....	44
6 implementación .....	44
6.1 Plataformas de desarrollo .....	45
6.2 Base de datos .....	45
6.3 Infraestructura de hardware y redes .....	45
7. Pruebas del software .....	46
7.1 Pruebas del software .....	46
7.2 Pruebas de Usabilidad .....	52
Bibliografía .....	55
Anexos .....	57

*Tabla 1 Contenido*

### **Lista de Tablas**

Tabla 1 Contenido .....	5
Tabla 2 Figuras .....	7

### **Tabla de Figuras**

Figura 1 Metodología en V. Clara Cera, (2021) .....	13
Figura 2 Diagrama de caso de uso de inicio y jugabilidad. Autoridad propia, (2023) .....	28
Figura 3 Diagrama de clases. Autoridad propia, (2023) .....	37
Figura 4 Diagrama de paquetes. Autoridad propia, (2023) .....	38

Figura 5 Diagrama de despliegue. Autoridad propia, (2023) .....	39
Figura 6. Sprite menú. Autoridad propia, (2023).....	40
Figura 7. Sprite Tutorial. Autoridad propia, (2023).....	41
Figura 8. Sprite niveles. Autoridad propia, (2023) .....	41
Figura 9. Imagen juego. Autoridad propia, (2023) .....	42
Figura 10. Imagen Pausa. Autoridad propia, (2023).....	42
Figura 11. Vector gratuito dibujos animados de teclado. Juicy_fish .....	43
Figura 12. Usar Microsoft Bluetooth Mouse. Microsoft.....	43
Figura 13. Imagen player muerto. Autoridad propia, (2023).....	44
Figura 14. Escudo y vida. Autoridad propia, (2023).....	44
Figura 15. Niveles sala 2. Autoridad propia, (2023).....	47
Figura 16. Nivel desbloqueado. Autoridad propia, (2023).....	48
Figura 17. Juego Pausado. Autoridad propia, (2023).....	49
Figura 18. Menu Play. Autoridad propia, (2023).....	50
Figura 19. Niveles sala 2. Autoridad propia, (2023).....	51
Figura 20. Juego Pausado. Autoridad propia, (2023).....	51
Figura 21. Diagrama Gantt. Autoridad propia, (2023).....	57

*Tabla 2 Figuras*

## Introducción

En la era digital, la convergencia entre la tecnología y la salud mental se ha convertido en un campo de exploración fascinante y lleno de posibilidades. Este proyecto, titulado "Becoming to Berserk", se sitúa en la intersección de la innovación tecnológica y el bienestar emocional, con el objetivo de ofrecer una solución única para un desafío contemporáneo. Nos sumergimos en el universo de los videojuegos 2D tipo Roguelike con una temática medieval, pero con un propósito más allá del entretenimiento: apoyar a adolescentes y jóvenes adultos (de 13 a 20 años) que enfrentan el trastorno de déficit de atención e hiperactividad (TDAH).

La realidad del TDAH plantea desafíos significativos en la capacidad de concentración y atención sostenida, afectando la calidad de vida de quienes lo experimentan. "Becoming to Berserk" no solo aspira a ser un juego cautivador, sino también a actuar como una herramienta terapéutica diseñada para mejorar estas habilidades cognitivas. En un contexto en el que la tecnología se convierte en aliada en la búsqueda del bienestar, este proyecto se destaca como una iniciativa que aprovecha la narrativa y la mecánica del juego para contribuir positivamente a la vida de quienes enfrentan este trastorno.

A lo largo de este documento, exploraremos el desarrollo detallado de "Becoming to Berserk", desde sus objetivos y metodología hasta su diseño y prueba. Este proyecto representa la unión de la creatividad y la tecnología con un propósito claro: proporcionar una experiencia de juego única que va más allá del entretenimiento tradicional. Al fusionar elementos medievales y mecánicas de juego innovadoras, "Becoming to Berserk" busca marcar un hito no solo en el mundo de los videojuegos, sino también en la aplicación de la tecnología para abordar desafíos de salud mental.

En resumen, "Becoming to Berserk" no es solo un juego; es un paso hacia adelante en la utilización de la tecnología como una fuerza positiva para el bienestar mental. A medida que



exploramos los detalles de su creación, descubriremos cómo este proyecto aspira a cambiar la forma en que comprendemos la relación entre la tecnología y la salud mental.

## **1 Título del proyecto**

Desarrollo de un videojuego 2D tipo roguelike con temática medieval, para adolescentes entre 13 a 20 años, para ayudar a personas con TDAH (trastorno de déficit de atención e hiperactividad) a fomentar la concentración y atención sostenida.

## **2 Planeación del proyecto**

### **2.1 Objetivos del proyecto**

#### **2.1.1 Objetivo general**

Desarrollo de un videojuego 2D Roguelike con temática medieval y estilo de combate “en tiempo real” dedicado para personas jóvenes entre 13 a 20 años con TDAH (trastorno de déficit de atención e hiperactividad) para fomentar la concentración y atención sostenida.

#### **2.1.2 Objetivos específicos**

1. Implementar un conjunto de mecánicas de juego que fomenten la concentración y la atención sostenida en el videojuego en 2D, utilizando principios de programación y diseño de software para asegurar su funcionamiento adecuado y coherencia con la temática del juego.
2. Crear un conjunto de niveles para el videojuego en 2D, utilizando herramientas de diseño de software y jugabilidad para asegurar un diseño de niveles efectivo y atractivo para los jugadores.

3. Desarrollar un conjunto de niveles en 2D que sea interesante de explorar y que tenga un equilibrio adecuado entre la dificultad y la recompensa. Para lograr este objetivo, implementaremos la metodología en V, que según Clara Cera (2021), ‘es una metodología que se emplea en varios ámbitos y puede llegar a ser de gran ayuda a la hora de nuestro desarrollo’.

4. Apoyar a personas con TDAH (trastorno de déficit de atención e hiperactividad) para mejorar su concentración y análisis por medio de las mecánicas que se emplean en el videojuego

## **2.2 Planteamientos del problema y/o necesidad**

El problema que este proyecto busca abordar es el desafío que enfrentan los adolescentes y jóvenes adultos (de 13 a 20 años) con trastorno de déficit de atención e hiperactividad (TDAH). El TDAH puede afectar significativamente la capacidad de concentración y atención sostenida de una persona, lo que a su vez puede afectar su calidad de vida.

“Becoming to Berserk” es un videojuego 2D tipo Roguelike con temática medieval que no solo busca proporcionar entretenimiento, sino también actuar como una herramienta terapéutica para mejorar estas habilidades cognitivas. A través de su narrativa y mecánica de juego, este proyecto aspira a contribuir positivamente a la vida de quienes enfrentan este trastorno.

Sin embargo, el desarrollo de tal juego presenta varios desafíos. Estos incluyen la implementación de mecánicas de juego efectivas, la creación de niveles atractivos y equilibrados, y la garantía de que el juego sea a la vez entretenido y terapéuticamente útil. Además, el proyecto debe llevarse a cabo dentro de un marco de tiempo específico y con recursos limitados.

Por lo tanto, el problema que este proyecto aborda es doble: ¿Cómo se puede desarrollar un videojuego que sea a la vez entretenido y beneficioso para los jóvenes con TDAH? Y, ¿cómo se puede lograr esto de manera eficiente y efectiva? Estas son las preguntas que “Becoming to Berserk” busca responder.

### 2.3 Alcance del proyecto

El proyecto "Becoming to Berserk" tiene como objetivo crear un videojuego de mazmorras en 2D utilizando la plataforma GameMaker. El juego se desarrollará en un entorno de fantasía medieval y ofrecerá una buena experiencia como un juego de acción con elementos roguelike.

Componentes del Alcance:

Diseño del Mundo:

Creación de un mundo de fantasía medieval en el cual se representarán varios niveles y en ellos se encargará el usuario de completarlos para avanzar la aventura.

Diseño de mapas y niveles detallados con elementos interactivos.

Historia y Personajes:

Desarrollo de una trama que involucra al jugador en una búsqueda para convertirse en un guerrero legendario "Berserk".

Creación de personajes memorables.

Sistema de Juego:

Implementación de un sistema de combate con armas y magia y curación.  
Diseño de mecánicas de juego para la exploración.

Arte y Gráficos:

Creación de ilustraciones y sprites detallados para personajes, enemigos y entornos.  
Diseño de animaciones fluidas para movimientos, ataques y habilidades.  
Diseño de efectos visuales para hechizos, objetos mágicos y otros elementos.

Música y Sonido:

Composición de una banda sonora original que se adapte a la atmósfera y la acción del juego.  
Incorporación de efectos de sonido para realzar la experiencia de juego.

Interfaz de Usuario:

Diseño de una interfaz intuitiva y atractiva que facilite la navegación, gestión de inventario y acceso a información importante.  
Creación de pantallas de menú, pantalla de inicio y otros elementos de la interfaz.

Pruebas y Ajustes:

Realización de pruebas para identificar y solucionar errores, desequilibrios y problemas de jugabilidad.

Entregables:

Versión completa y jugable del videojuego "Becoming to Berserk"  
documentación detallada del diseño del juego, incluidos personajes, y mecánicas

Arte, sprites, animaciones y efectos visuales utilizados en el juego  
Banda sonora original y efectos de sonido  
Archivos fuente del proyecto en GameMaker  
Manual de usuario para los jugadores.

#### Restricciones:

- El juego se desarrollará exclusivamente en 2D utilizando las capacidades de Game Maker.
- En cuanto a los recursos externos, se utilizarán assets gratuitos disponibles en la web para elementos como gráficos y sonidos, siempre y cuando se ajusten a las necesidades del proyecto.
- En caso de ser necesario, se empleará inteligencia artificial para generar ciertos elementos del juego, como imágenes y música. Esto permitirá una mayor personalización y adaptabilidad del juego a las necesidades específicas de los usuarios.
- El equipo de desarrollo incluye solo a los creadores de este documento ya indicados en la portada.

## 2.4 Metodología de desarrollo de software

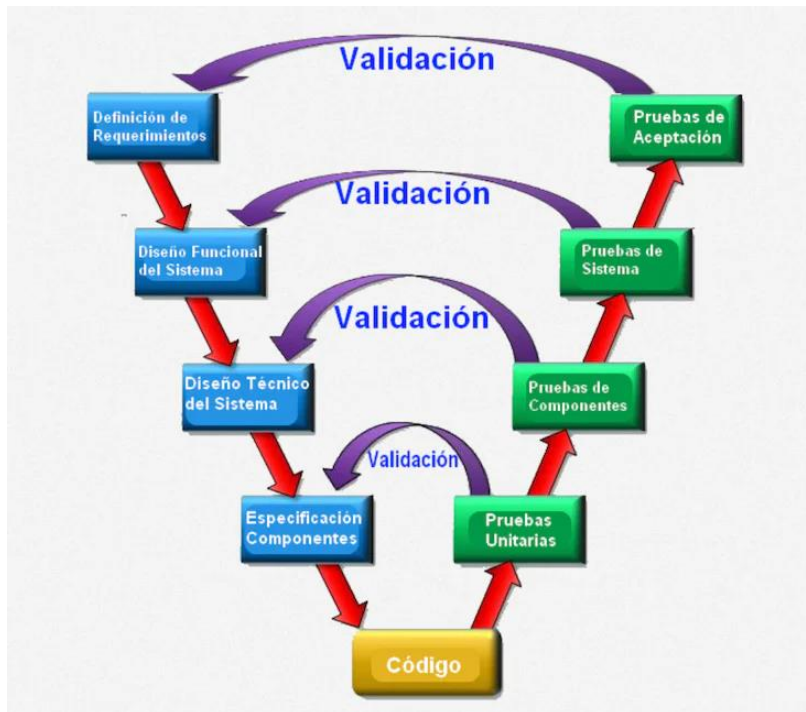


Figura 1 Metodología en V. Clara Cera, (2021)

Modelo V es una metodología muy útil que empleamos en el desarrollo del videojuego en 2D. “muchos proyectos usan el modelo en V para gestiones en lo que respecta a metodologías ágiles” (Clara Cera, 2021), qué es lo que buscamos en nuestro proyecto por más de ser una metodología clásica.

Según el proyecto es viable por lo que ayuda a estructurar de manera clara y sencilla la implementación de todos los módulos que requiere para el desarrollo del proyecto.

### Fase de Diseño y Planificación:

Definición de requerimientos. Según la autora en este se resaltarán los puntos de partida del proyecto con el cual se analizaron los puntos necesarios para revisar que requiere el proyecto no solo en la parte técnica sino también el análisis en este caso del mercado y las tendencias que están de moda en estos tiempos.

Identificar el objetivo del juego: En esta etapa, se define el objetivo del videojuego, en este caso, mejorar la capacidad de concentración y fomentar la creatividad, entre otros.

Investigar y definir el mercado: Realizando una investigación con respecto a las compras recientes y valoraciones de videojuegos en plataformas de venta de este tipo de productos para un mercado orientado al PC (computador personal) se identificó que los juegos con la temática Roguelike que empleamos, regularmente usan ambientes medievales o de fantasía antiguos usando mecánicas, a su vez validando la mayoría del público que está interesado en estos juegos son jóvenes con edades entre 15 a 25 años los cuales disfrutan más este tipo de títulos actualmente

Definir el concepto del juego: Luego de haber analizado, se estableció que el videojuego va a tener como idea principal que el protagonista tendrá que sortear ciertos obstáculos para salir de una mazmorra en la que fue encerrado por el hecho de que se presagio que en algún momento una persona se encargaría de destruir toda la civilización tal y como se conoce por mero capricho, pero qué sería específicamente de una villa alejada de toda civilización. Esto al asustar al gobernante de la tierra se encarga de asediar todos los pueblos que están en las lejanías en los calabozos que han usado para enviar cualquier traidor que se quiera enfrentar al reino, no obstante, el protagonista recibe cierta ayuda para conseguir unos objetos que le ayudaran a salir de este calabozo y recuperar la tranquilidad de su pueblo.

Este es el comienzo de la historia en la cual el escenario es un mundo medieval, con un género de acción Roguelike en el que el jugador se ve embarcado en esta historia para recuperar dicha paz.

En cuanto a las mecánicas se implementa un combate de tiempo real en cual el jugador se enfrenta a los enemigos sin tener que entrar en procesos como combates por turnos para un mayor dinamismo a la hora de jugar.

En cuanto a la estética como se basa en épocas medievales se implementan paletas de colores en contraste con los escenarios, al inicio se usan colores marrones y así sucesivamente llegando a colores vivos como el dorado y el morado que en aquellas épocas eran pigmentos difíciles de conseguir

Crear el plan del proyecto: Al ser dos personas se delegan las tareas según lo correspondiente al proceso básico.

Inicialmente la creación de los sprites (las animaciones del personaje) y a su vez las mecánicas de ataque en esto se requieren tareas de diseño primeramente para luego codificar y poder emplear dichas animaciones en los entornos deseados.

La creación de escenarios y de mecánicas en las que el personaje se irá desplazando por los diferentes niveles.

Se implementarán los diseños de los enemigos y las mecánicas de ataque para suponer parte de los retos al usuario.

Se incorporan ciertos objetos de colisión en los cuales este pueda llegar a interactuar así como los cuadros de texto y los puzzles que serán el otro desafío para el jugador, a su vez revisar las mecánicas que se implementaran en el entorno para hacer una simbiosis con el entorno.

Diseño funcional del sistema:

Crear el documento de diseño del juego: Se debe crear el documento de diseño del juego, que incluya detalles de la historia, personajes, mecánica de juego y niveles.

Diseñar la interfaz de usuario: Se debe diseñar la interfaz de usuario, incluyendo la pantalla de inicio, los menús y los controles del juego.

Definir las mecánicas de juego: En esta etapa, se deben definir las mecánicas de juego, incluyendo la inteligencia artificial, la física, el control del jugador, entre otros.



#### Diseño del sistema:

Crear los assets del juego: Se deben crear los assets del juego, incluyendo personajes, escenarios, efectos visuales, efectos de sonido, entre otros.

Programar las mecánicas de juego: En esta etapa, se debe programar las mecánicas de juego, incluyendo la inteligencia artificial, la física, el control del jugador, entre otros.

Crear los niveles del juego: Se deben crear los niveles del juego, incluyendo el diseño de los escenarios, la colocación de objetos, enemigos y otros elementos necesarios.

#### Codificación:

Implementar el código del juego: En esta etapa, se debe programar el código del juego utilizando el motor de desarrollo de videojuegos seleccionado.

Realizar pruebas y ajustes: Se deben realizar pruebas en el juego para encontrar y corregir errores y ajustar los elementos del juego en función de la retroalimentación obtenida.

#### Fase de Implementación y Pruebas:

Pruebas de validación de requerimientos: En esta etapa, se podrían realizar pruebas para validar que los requerimientos establecidos para el videojuego sean claros, medibles y alcanzables. Por ejemplo, se podría verificar que el juego tenga un tiempo de carga máximo de 10 segundos, que los controles del juego sean intuitivos y fáciles de usar, entre otros.

#### Diseño funcional del sistema:

Pruebas de validación de diseño: En esta etapa, se podrían realizar pruebas para validar el diseño funcional del videojuego. Por ejemplo, se podría revisar que los diagramas de flujo, los wireframes o los prototipos se correspondan con los requerimientos establecidos para el juego y se ajusten a las expectativas del usuario final.

#### Diseño del sistema:

**Pruebas de integración:** En esta etapa, se podrían realizar pruebas para verificar que las distintas partes del sistema funcionen correctamente juntas. Por ejemplo, se podría probar los elementos gráficos del juego (personajes, objetos, fondos, etc.) se integren correctamente con la lógica del juego y la física del motor de Game Maker.

#### Codificación:

En esta etapa, se programará el código del juego utilizando el motor de desarrollo de videojuegos GameMaker y su lenguaje de programación nativo, GML (GameMaker Language). Este lenguaje es especialmente adecuado para el desarrollo de videojuegos 2D y permite una gran flexibilidad y control sobre los elementos del juego. Durante esta etapa, se implementarán las mecánicas de juego, se definirán los comportamientos de los personajes y enemigos, y se programarán las interacciones entre los diferentes elementos del juego.

**Pruebas unitarias:** En esta etapa, se podrían realizar pruebas para verificar que cada unidad de código funcione correctamente y de acuerdo con los requerimientos establecidos. Por ejemplo, se podría probar que la mecánica de salto del personaje funcione correctamente, que los enemigos se muevan de acuerdo con los patrones establecidos, o que los objetos se colisionan y se destruyan correctamente.

**Pruebas de aceptación:** En esta etapa, se realizarán pruebas para verificar que el juego cumpla con los requerimientos y especificaciones establecidos y sea funcional y usable para el usuario final. Por ejemplo, se podría probar que el juego se inicie correctamente, que los gráficos y la música sean de alta calidad, que la dificultad del juego sea adecuada, y que las mecánicas del juego funcionen como se esperaba

En resumen, el modelo en V es una metodología de desarrollo que puede ser adaptada para el desarrollo de un videojuego en 2D. Este modelo permite una planificación y diseño detallado del juego antes de la implementación y pruebas, lo que puede aumentar la eficiencia y calidad del proceso de desarrollo.

### **3 Marco Teórico y Estado del Arte.**

#### **3.1 Marco teórico**

##### **Videojuegos y TDAH**

Los videojuegos pueden ser beneficiosos para los jóvenes con Trastorno por Déficit de Atención e Hiperactividad (TDAH) de varias maneras. Según investigaciones recientes, los videojuegos pueden ayudar a mejorar las habilidades espaciales y visuales, lo que puede ser especialmente útil para los jóvenes con TDAH. Los videojuegos a menudo requieren una buena coordinación mano-ojo, y jugarlos puede ayudar a mejorar esta habilidad. Muchos videojuegos requieren que los jugadores piensen estratégicamente y resuelvan problemas, lo que puede ayudar a mejorar estas habilidades. Algunos videojuegos pueden ayudar a mejorar las habilidades sociales y de lenguaje.

Además, un estudio de 2021 encontró que los niños con TDAH tienden a jugar videojuegos con más frecuencia que otros niños de su edad. El estudio también encontró que la gravedad de la compulsión por los videojuegos se correlacionaba con la gravedad de los síntomas del TDAH: impulsividad, falta de atención e hiperactividad. Sin embargo, los autores aclararon que el TDAH no parece causar adicción a los videojuegos, pero el TDAH debería verse como un factor contribuyente al trastorno del juego.

##### **Diseño de juegos 2D**

Según la plataforma Tokio School (2023), el diseño de este videojuego 2D incluirá elementos visuales como una paleta de colores vibrante, elementos de animación para mejorar la efectividad de los ataques de los personajes y una perspectiva isométrica para ofrecer más profundidad. La inclusión de elementos del ambiente como barriles, cajas y vegetación permitirá

que el jugador interactúe con el mundo del juego y brindará un sentimiento de exploración y aventura.

### **GameMaker**

GameMaker (2013) es una herramienta de creación de videojuegos. Se basa en una interfaz amigable e intuitiva, con características útiles como la capacidad de desarrollar para múltiples plataformas. La utilización de GameMaker permitirá el uso de scripts, que serán esenciales para la creación de la inteligencia artificial de enemigos, el control de los efectos especiales y la creación de las mecánicas de juego.

### **Roguelike.**

El género Roguelike es un subgénero de los videojuegos de rol que se caracteriza por niveles generados de manera procedimental, jugabilidad basada en turnos y la muerte permanente del personaje del jugador. Los juegos Roguelike son típicamente “dungeon crawlers”, en los cuales el jugador controla a un único personaje que debe navegar por una serie de habitaciones o corredores, derrotando enemigos y recolectando objetos en el camino.

### **Controles de Teclado y Mouse:**

En este tipo de juego, los jugadores utilizan el teclado y el mouse de su computadora para interactuar con el juego. El teclado se usa para mover al personaje, y activar acciones, mientras que el mouse se utiliza para apuntar, hacer clic en objetos o interactuar con elementos en pantalla.

### **Personaje Jugable:**

El personaje jugable es el avatar controlado por el jugador en el juego. Puede ser un héroe, un personaje personalizado o un personaje predefinido, dependiendo de la historia y la mecánica del juego. Los jugadores controlarán al personaje jugable utilizando el teclado y el mouse, lo

que les permitirá moverse por el mundo del juego, interactuar con objetos y enemigos, y realizar acciones como atacar, o usar objetos.

### **Mundo del Juego:**

Según Kochi, T (2021), el mundo del juego se refiere al entorno y el escenario en el que se desarrolla la acción del juego. Puede incluir diferentes ubicaciones, escenarios y elementos interactivos, en el cual este se verá inmerso para ejecutar y explotar todas las acciones posibles en este.

## **3.2 Estado del arte**

### **Desarrollo de un videojuego**

**Universidad de Bogotá Jorge Tadeo Lozano**

**“PRISON ESCAPIST”**

**Autor:**

**William Alejandro Martinez Mora**

En el proyecto "Prison Scapist", se desarrolló un videojuego utilizando herramientas como Unity3D y Blender. El enfoque fue entender el proceso de creación de videojuegos, desde las mecánicas hasta la programación. Se adquirieron conocimientos en gestión de proyectos y programación, logrando un prototipo funcional. El objetivo era comprender cómo se hacen los juegos y adquirir habilidades para ingresar a la industria desde una perspectiva de tecnólogo en desarrollo de software.

### **Desarrollo de un videojuego y documentación de su diseño**

**Universitat Politècnica de Catalunya (UPC) - Barcelona Tech**

**"Transcendental Journey"**

**Autor:**

**Rubén González López**

En el proyecto "Desarrollo de un videojuego y documentación de su diseño", Rubén González López creó un juego 2D de plataformas en Unity. El juego aborda la experiencia de transición de género para sensibilizar y normalizar la transexualidad. El trabajo es parte de sus estudios de Ingeniería Informática y se basa en la especialidad de Ingeniería del Software. El juego, llamado "Transcendental Journey", se construyó con conocimientos previos y se planea ampliar en el futuro para añadir más mecánicas y narrativa personal. Este proyecto muestra cómo los videojuegos pueden ser herramientas educativas y creativas en temas sensibles.

**Diseño de un videojuego educativo como material didáctico en la clase de tecnología e informática para ciclo cuatro**

**Trabajo de grado**

**Universidad Pedagógica Nacional**

**"Enigmatronic"**

**Autores:**

**Diego Gantiva & Erika Gantiva**

En el proyecto "Enigmatronic", se creó un videojuego educativo 2D para la clase de tecnología e informática en ciclo cuatro. Diseñado por Diego Gantiva y Erika Gantiva, el juego aborda temas de física mecánica y eléctrica, como máquinas simples y componentes eléctricos.

Los estudiantes resuelven acertijos basados en estos conceptos, fomentando la comprensión y la resolución de problemas. "Enigmatronic" se presenta como una herramienta didáctica que enriquece la enseñanza al permitir la experimentación y estimular la investigación. El juego también podría mejorar en el futuro al agregar un editor de niveles y expandir su alcance a dispositivos móviles.

### **Desarrollo de Prototipo de Videojuego "Les Catacombes"**

**Universidad Autonoma de Bucaramanga**

**Autores:**

**Kevin Castellanos Tovar, Fabian Andres Delgado Serrano**

Este proyecto involucra la creación de un prototipo de videojuego multijugador llamado "Les Catacombes" para la empresa "Life Is The Game". Se utiliza la metodología ágil SUM y el motor Unity 3D. Los jugadores enfrentan desafíos en catacumbas, moviendo fichas y recolectando tokens. El objetivo principal es atraer inversores, sin intención de comercialización directa.

### **Proyecto de desarrollo de videojuego**

**"Boom Away"**

**Universidad de los Andes**

**Autores:**

**Daniel Andrés Pardo Chaves, Andrés Felipe Losada y Pablo Andrés Suárez Murillo**

Desarrollo de un juego llamado "Boom Away" por estudiantes de la Universidad de los Andes. El juego se basa en el uso innovador de explosivos para el movimiento en niveles diseñados alrededor de esta idea. Se emplearon principios de diseño de software y soluciones de ingeniería para crear una aplicación escalable. Se recopilaron datos analíticos del juego para mejorar la experiencia del usuario.

## **4 Especificación de requisitos de software (IEEE 830)**

### **4.1 Perspectiva del producto**

“Becoming to Berserk” es un juego con ambiente medieval de tipo de plataforma el cual se emplea para ayudar a jóvenes de entre 13 y 20 años con TDAH. Este tiene como objetivo apoyarlos y, por medio de las mecánicas del videojuego, mejorar su concentración activa y fomentar la concentración usando tanto las mecánicas del juego como su historia para que al interactuar tanto de manera activa y/o pasiva vayan adquiriendo conocimientos para desenvolverse en diferentes entornos.

### **4.2 Funcionalidad del producto**

Este es un videojuego de plataforma, esto quiere decir que el usuario manejará un personaje el cual tendrá que controlar para superar ciertos niveles los cuales se reflejan en el mapa donde está el personaje. Los desafíos incluirán resolver acertijos, como encontrar la salida hacia el siguiente nivel, y enfrentarse a combates en salas repletas de enemigos. De esta manera, al interactuar, el usuario se encargará de superar estos obstáculos para completar la historia.



### 4.3 Características de los usuarios

Tipo de usuario	Jugador
Formación	No se requiere experiencia previa en videojuegos. El usuario solo necesita estar interesado en los juegos tipo roguelike de plataforma.
Actividades	Control y manejo del personaje, a su vez manejo de inventario, resolver acertijos y combates por medio de las mecánicas del juego

*Tabla 1. Características de usuario. Autoridad propia*

### 4.4 Restricciones

#### Requisitos mínimos de hardware

- Procesador de 2GHz
- 3 GB de Ram
- 512 MB de memoria de video
- 2 GB de almacenamiento

#### Requisitos mínimos de software

- Windows 10 o versiones posteriores

## 4.5 Dependencias y suposiciones

- Se eligió el motor de desarrollo de videojuegos Game Maker por recomendación de un docente y porque es una herramienta muy buena para empezar en los videojuegos.
- El rendimiento del juego debería ser óptimo siempre que se cumplan los requisitos mínimos de software y hardware.
- Solo se podrá ejecutar en entornos de Windows 7 o versiones posteriores.

## 4.6 Requisitos Específicos

### 4.6.1 Actores y Roles

#### **Jugador**

**Descripción:** El jugador es el usuario final del videojuego. Controla al personaje principal y se sumerge en el mundo medieval del juego. Su objetivo principal es completar los niveles, enfrentar desafíos y avanzar en la historia.

#### **Interacción con el Sistema:**

Controla al personaje principal durante el juego.

Participa en combates en tiempo real con enemigos.

#### **Administrador del Juego**

**Descripción:** El administrador del juego es responsable de supervisar y mantener el funcionamiento general del videojuego. Puede ser un desarrollador o un moderador encargado de gestionar aspectos técnicos.

**Interacción con el Sistema:**

Realiza actualizaciones y parches para corregir errores y mejorar la experiencia del juego.

Proporciona soporte técnico y responde a preguntas y problemas de los jugadores.

**Enemigos**

**Descripción:** Los enemigos son personajes controlados por la inteligencia artificial del juego que representan desafíos y obstáculos para el jugador. Pueden variar en tipo y dificultad a lo largo del juego.

**Interacción con el Sistema:**

Atacan al jugador cuando se encuentran en su rango.

Siguen patrones de movimiento y ataque predefinidos.

Pueden soltar recompensas o botín cuando son derrotados.

**4.6.2 Requisitos Funcionales****RF1: Mecánica de Combate en Tiempo Real**

El videojuego debe proporcionar una mecánica de combate en tiempo real que permita a los jugadores controlar al personaje y enfrentarse a enemigos de manera dinámica. Los jugadores deben poder atacar, esquivar y usar habilidades características de las armas durante el combate.

**RF2: Diseño de Niveles Interactivos**

Los niveles del juego deben ser interactivos y desafiantes. Deben incluir elementos como obstáculos, salas repletas de enemigos y salas con recompensas.

### **RF3: Sistema de Progresión y Equipamiento**

El juego debe tener un sistema de equipamiento que permita a los jugadores adquirir diferentes tipos de armas, armaduras y objetos mágicos.

### **RF4: Sin Sistema de Guardado de Partidas**

Dado el género Roguelike del juego, no se implementará un sistema de guardado de partidas. Esto garantiza una experiencia de juego única en cada partida.

### **RF5: Inteligencia Artificial de los Enemigos**

Los enemigos en el juego deben estar controlados por una inteligencia artificial que los haga reaccionar de manera realista a las acciones del jugador. Deben ser capaces de atacar y moverse por el entorno.

### **RF6: Interfaz de Usuario Intuitiva**

El videojuego debe contar con una interfaz de usuario intuitiva que permita a los jugadores navegar por los menús, gestionar su inventario y acceder a información importante de manera sencilla y eficiente.

#### **4.6.3 Diagrama de caso de uso**

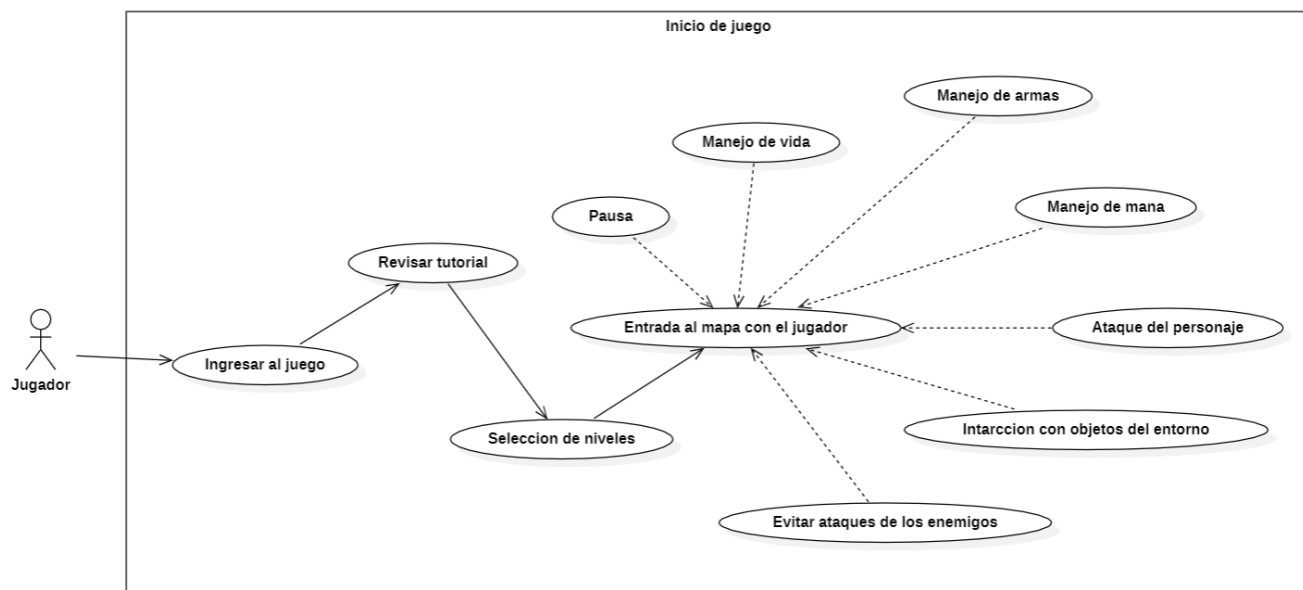


Figura 2 Diagrama de caso de uso de inicio y jugabilidad. Autoridad propia, (2023)

#### 4.6.4 Especificaciones de caso de uso

Nombre:	Inicio del juego
Autor:	Sergio Martinez, Jonathan Reyes

Fecha:	09/09/2023	
Descripción: Se ejecuta el juego en el ordenador para poder comenzar a jugar		
Actores: Jugador		
Precondiciones: Tener los requisitos tanto de hardware como de software para poder ejecutar el juego.		
Flujo normal:	Flujo Alternativo:	
<ol style="list-style-type: none"> <li>1. Instalar el juego en el ordenador</li> <li>2. Al iniciar el acceso directo comenzara a visualizar el tutorial.</li> <li>3. Seleccionara el nivel que este sin un candado para comenzar</li> <li>4. Aparecerá en el primer nivel y con las indicaciones del tutorial podrá comenzar a jugar.</li> <li>5. En dado caso de querer salir o cancelar el sonido oprimirá la letra “p” en el teclado para estas opciones</li> <li>6. Ya con estas indicaciones, la persona será capaz de navegar por la interfaz de usuario</li> </ol>	<ol style="list-style-type: none"> <li>3A Una vez desbloqueados más niveles será capaz de ingresar en estos</li> </ol>	
Postcondiciones: Al ser un juego lineal no tendrá punto de guardado, esto resultara en que, si el jugador cierra el juego, el proceso se perderá, a no ser que este lo finalice lo cual le dará un logro y dejara todo desbloqueado		
Prioridad: Alta		

Tabla 2. Especificaciones de caso de uso. Autoridad propia.

## Actores

Jugador

### **Iniciar juego**

Se ejecuta el videojuego en el Pc para entrar a este

### **Dar permisos**

Esto se hace por normativas de calidad para evitar inconvenientes con que la historia se relacione con alguien real u pensar que se esté haciendo copyright de algún otro software parecido

Comenzar partida

Se selecciona este botón para comenzar el juego

Colocar nombre del personaje

Empezar historia

Una vez ingresado el nombre se podrá comenzar a jugar y con esto empezar la historia y que el jugador empiece a interactuar con nuestro videojuego y de esta manera empezar a ayudar a las personas a las cuales va dirigido nuestro título, una vez hecho esto el jugador tendrá que interactuar con el inventario, y las mecánicas de ataque.

Menú de pausa

En este se podrán manejar temas no relacionados con la historia como es salir al escritorio, maneras de ajustar del volumen.

## **4.7 Requisitos de Rendimiento**

En esta sección, se definen los requisitos de rendimiento del videojuego "Becoming to Berserk". Estos requisitos son fundamentales para garantizar una experiencia de juego fluida y satisfactoria para los jugadores.

### **Velocidad de Respuesta del Juego:**

**Requisito:** El juego debe responder a las acciones del jugador con un retraso máximo de 60 milisegundos.

**Justificación:** La respuesta rápida a las acciones del jugador es crucial para la jugabilidad. Cualquier retraso notable puede afectar negativamente la experiencia del usuario y provocar frustración.

### **Requisitos de Hardware:**

**Requisito:** El juego debe ejecutarse sin problemas en una computadora con las siguientes especificaciones mínimas: procesador de 2GHz, 3 GB de RAM, 512 MB de memoria de video y 2 GB de espacio de almacenamiento.

**Justificación:** Establecer requisitos mínimos de hardware garantiza que el juego sea accesible para una amplia audiencia. Los jugadores con computadoras que cumplen con estas especificaciones deberían poder disfrutar del juego sin problemas de rendimiento.

## **4.8 Restricciones de Diseño**

**Diseño Gráfico en 2D:** El juego se desarrollará exclusivamente en 2D, lo que significa que todos los elementos visuales, personajes, entornos y efectos serán en dos dimensiones. No se permitirá el uso de elementos en 3D en ninguna parte del juego.

**Estética Medieval:** El diseño artístico del juego se basará en una estética medieval, utilizando paletas de colores y elementos visuales que sean coherentes con esta temática. Los colores y estilos de arte deberán reflejar el período medieval.



**Uso de GameMaker:** El juego se desarrollará utilizando el motor de desarrollo de videojuegos GameMaker. Todas las mecánicas, y programación se realizarán en este entorno.

**Presupuesto Limitado:** Se seguirá un presupuesto limitado para recursos externos, como música y arte. En la medida de lo posible, se utilizarán recursos gratuitos y, en caso necesario, se emplearán recursos externos de manera muy selectiva y económica.

**Plataforma Windows:** El juego se desarrollará exclusivamente para la plataforma Windows 10 o versiones posteriores. No se contempla el soporte para otras plataformas como macOS o sistemas operativos móviles.

**Equipo de Desarrollo Limitado:** El equipo de desarrollo incluye únicamente a los creadores del proyecto, Martínez Daza Sergio Daniel y Reyes Parra Jonathan David, tal como se menciona en la portada del documento. No se prevé la incorporación de otros miembros al equipo.

#### **4.9 Atributos del Software del Sistema**

En esta sección, se definen los atributos del software del sistema que son esenciales para el videojuego "Becoming to Berserk". Estos atributos son fundamentales para garantizar la calidad, seguridad y funcionalidad del juego.

##### **Optimización para PC:**

**Descripción:** El juego debe estar completamente optimizado para plataformas de PC, aprovechando el hardware y el software de computadoras personales. Debe tener en cuenta las

especificaciones comunes de PC y garantizar un rendimiento suave en una amplia variedad de configuraciones de hardware.

**Importancia:** La optimización específica para PC es fundamental para ofrecer una experiencia de juego fluida y sin problemas en las computadoras de los jugadores. Garantiza que el juego funcione eficientemente en diversas configuraciones de PC.

#### **Interfaz Intuitiva y Control del Teclado/Ratón:**

**Descripción:** La interfaz de usuario del juego debe ser intuitiva y fácil de usar con un teclado y ratón. Los controles deben ser adaptables y personalizables para permitir que los jugadores ajusten la configuración según sus preferencias.

**Importancia:** Una interfaz fácil de usar y controles adaptativos son esenciales para que los jugadores de PC puedan disfrutar del juego sin dificultades. Permite una experiencia de juego cómoda y personalizada para cada jugador.

#### **Compatibilidad con Versiones de PC y Actualizaciones:**

**Descripción:** El juego debe ser compatible con las versiones actuales de sistemas operativos de PC y estar preparado para futuras actualizaciones. Debe seguir los estándares y tecnologías más recientes para garantizar la compatibilidad con las configuraciones de PC modernas.

**Importancia:** La compatibilidad y las actualizaciones son esenciales para asegurar que el juego siga siendo jugable en las computadoras de los jugadores a medida que evolucionan las tecnologías y los sistemas operativos. Garantiza una experiencia continua y sin problemas para los jugadores de PC.

## **5. Diseño del software (ISO -12207-1)**

El diseño del software para nuestro videojuego 2D Roguelike, “Becoming to Berserk”, se realizará siguiendo el estándar ISO 12207. Este estándar proporciona un marco para el ciclo de vida del software, incluyendo el diseño. En esta etapa, tomaremos decisiones sobre cómo se implementará el sistema y crearemos los planos para la construcción del software.

Nuestro enfoque se centrará en la creación de un videojuego que sea atractivo y beneficioso para los jóvenes con TDAH. Para lograr esto, nos centraremos en los siguientes aspectos durante la fase de diseño:

**Mecánicas de juego:** Diseñaremos mecánicas de juego que sean atractivas y desafiantes, pero también que ayuden a fomentar la concentración y la atención sostenida.

**Diseño de niveles:** Crearemos niveles que sean interesantes y divertidos de explorar, pero que también presenten desafíos que requieran concentración y atención al detalle.

**Interfaz de usuario:** Diseñaremos una interfaz de usuario que sea intuitiva y fácil de usar, para que los jugadores puedan concentrarse en el juego en sí en lugar de luchar con los controles.

**Arte y gráficos:** Crearemos arte y gráficos que sean atractivos y que se ajusten a la temática medieval del juego.

Durante esta etapa, también tomaremos decisiones sobre qué herramientas y tecnologías utilizaremos para implementar nuestro diseño. Esto incluirá la elección de un motor de juego

(GameMaker), así como cualquier otra herramienta o tecnología necesaria para crear los gráficos, sonidos y otras características del juego.

Finalmente, crearemos planos detallados para la construcción del software. Estos planos incluirán diagramas de clases, diagramas de paquetes y otros artefactos de diseño que nos ayudarán a implementar nuestro diseño de manera eficiente y efectiva.

### **5.1 Diseño de la Arquitectura de software**

La arquitectura de software para nuestro videojuego 2D Roguelike, “Becoming to Berserk”, se diseñará con un enfoque modular para facilitar el desarrollo y el mantenimiento. La arquitectura se dividirá en varios componentes clave que interactuarán entre sí para crear la experiencia completa del juego.

- **Motor del juego:** Este será el núcleo de nuestro videojuego, encargado de manejar la lógica del juego, las físicas y las interacciones del jugador. Utilizaremos GameMaker como nuestro motor de juego debido a su facilidad de uso y su potente conjunto de características para el desarrollo de juegos 2D.

- **Interfaz de usuario:** Este componente se encargará de todas las interacciones del jugador con el juego, incluyendo los menús, los controles y la visualización de información al jugador. Nuestro objetivo es hacer que la interfaz de usuario sea intuitiva y fácil de usar, permitiendo a los jugadores concentrarse en el juego en sí.

- **Gestión de niveles:** Este componente se encargará de cargar y gestionar los diferentes niveles del juego. Cada nivel será diseñado con cuidado para proporcionar un desafío atractivo y ayudar a fomentar la concentración y la atención sostenida.

- **Sistema de combate:** Este componente manejará todas las interacciones de combate en el juego, incluyendo los ataques del jugador, las respuestas de los enemigos y la gestión de la salud y el maná.

Cada uno de estos componentes será diseñado e implementado con cuidado para asegurar que trabajen juntos de manera eficiente y efectiva. Nuestro objetivo es crear un videojuego que sea atractivo, divertido y beneficioso para nuestros jugadores objetivo.

## **5.2 Diseño detallado del software**

El diseño detallado del software para nuestro videojuego 2D Roguelike, “Becoming to Berserk”, se centrará en cómo se implementarán los componentes individuales identificados en el diseño de la arquitectura.

- **Motor del juego:** Utilizaremos GameMaker como nuestro motor de juego. Este motor proporciona una serie de características y funcionalidades que nos permitirán implementar la lógica del juego, las físicas y las interacciones del jugador. Los algoritmos específicos para la lógica del juego y las físicas se desarrollarán utilizando GML (GameMaker Language), el lenguaje de programación nativo de GameMaker.

- **Interfaz de usuario:** La interfaz de usuario se implementará utilizando las funcionalidades de UI proporcionadas por GameMaker. Esto incluirá la creación de menús, botones e indicadores de estado (como barras de salud y maná) utilizando los objetos y scripts de GameMaker.

- **Gestión de niveles:** La gestión de niveles se implementará utilizando las funcionalidades de gestión de habitaciones y niveles proporcionadas por GameMaker. Cada nivel será un objeto que contendrá su propio conjunto de entidades (como personajes, enemigos y objetos) y su propia lógica.

- **Sistema de combate:** El sistema de combate se implementará utilizando una combinación de scripts y objetos en GameMaker. Los scripts definirán las acciones que pueden realizar los personajes (como atacar, defenderse y usar objetos), mientras que los objetos representarán a los personajes, enemigos y objetos en el juego

### 5.2.1 Diagrama de clases

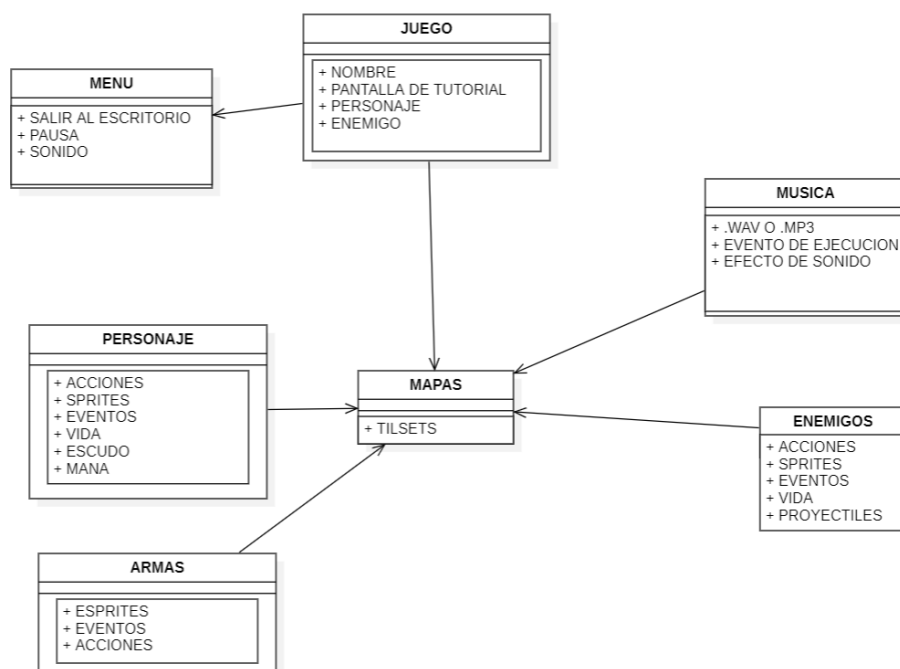


Figura 3 Diagrama de clases. Autoridad propia, (2023)

### 5.2.2 Diagrama de paquetes

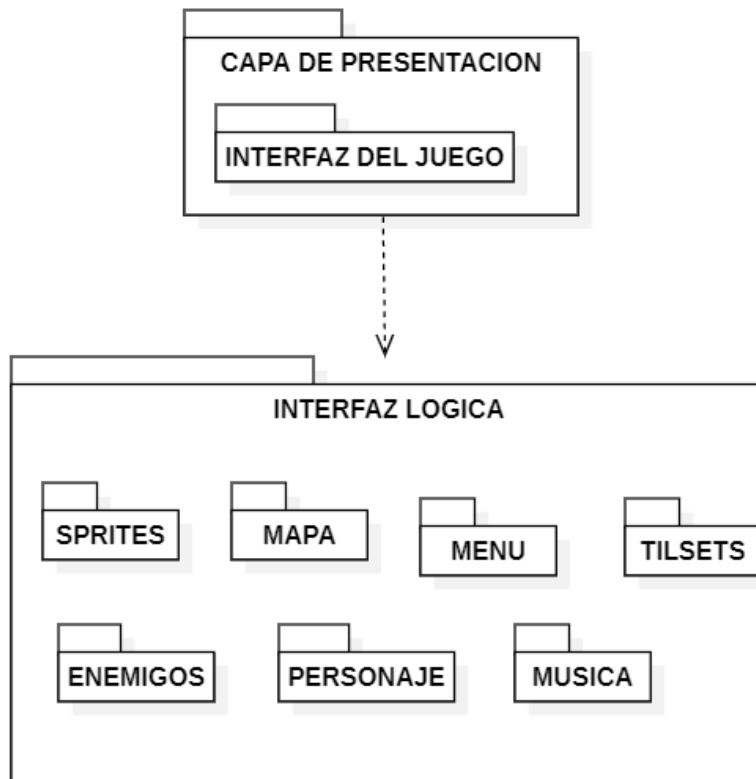


Figura 4 Diagrama de paquetes. Autoridad propia, (2023)

### 5.2.3 Diagrama de despliegue

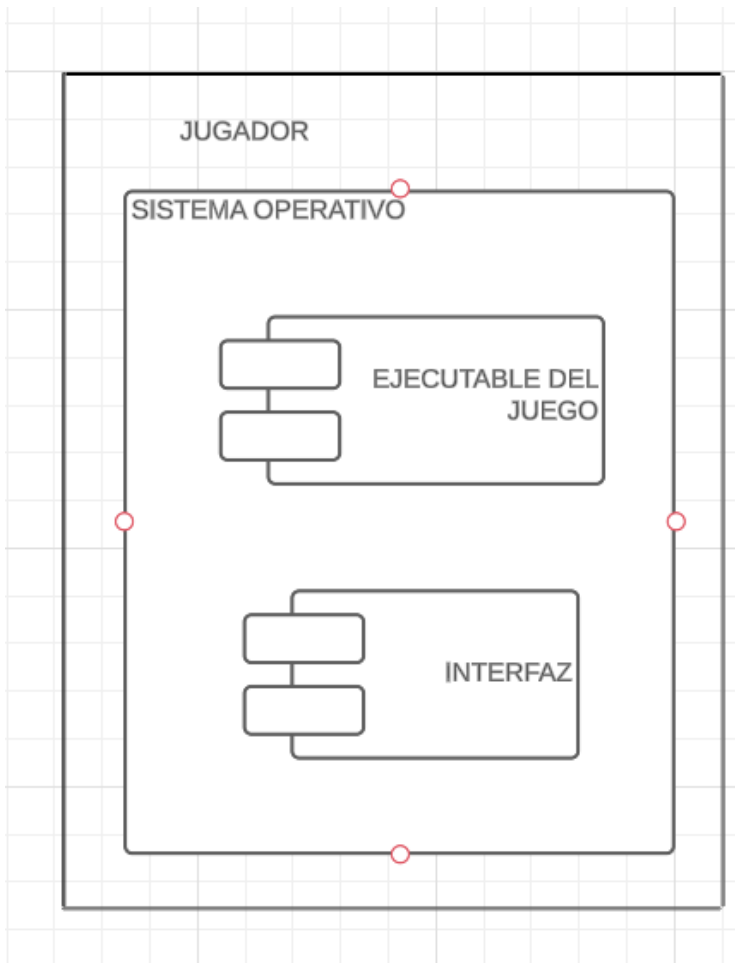


Figura 5 Diagrama de despliegue. Autoridad propia, (2023)

### 5.3 Diseño de la Interfaz



El diseño de la interfaz se centrará en proporcionar una experiencia de usuario intuitiva y atractiva, que permita a los jugadores con TDAH concentrarse en el juego sin distracciones innecesarias.

### 5.3.1 Interfaz Gráfica de Usuario

La Interfaz Gráfica de Usuario (GUI) presentará un diseño medieval con elementos visuales que reflejen la temática del juego. Los menús serán claros y fáciles de navegar, con iconos y texto legibles. La barra de salud y maná estará ubicada en la esquina superior izquierda, mientras que el mapa estará en la esquina superior derecha. El inventario será accesible a través de un menú que los jugadores pueden abrir y cerrar a voluntad.

Interfaz de inicio



Figura 6. Sprite menú. Autoridad propia, (2023)

Sprite de tutorial



Figura 7.Sprite Tutorial. Autoridad propia, (2023)

## Menú de niveles

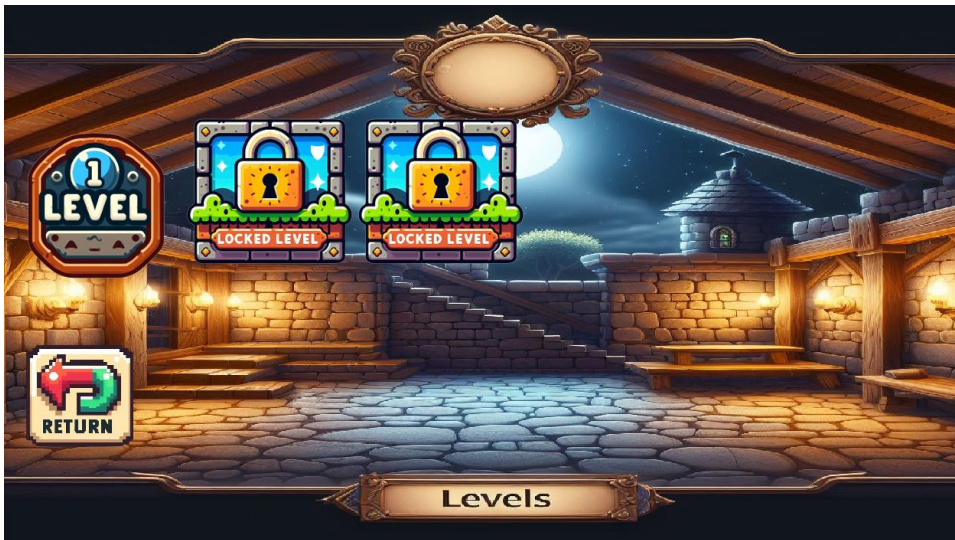


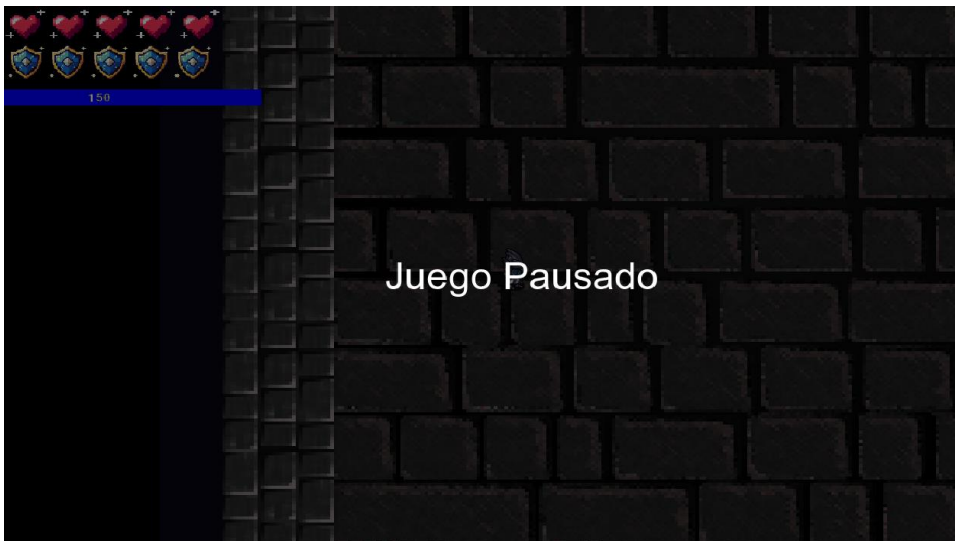
Figura 8.Sprite niveles. Autoridad propia, (2023)

## Entorno



*Figura 9. Imagen juego. Autoridad propia, (2023)*

Menú de pausa

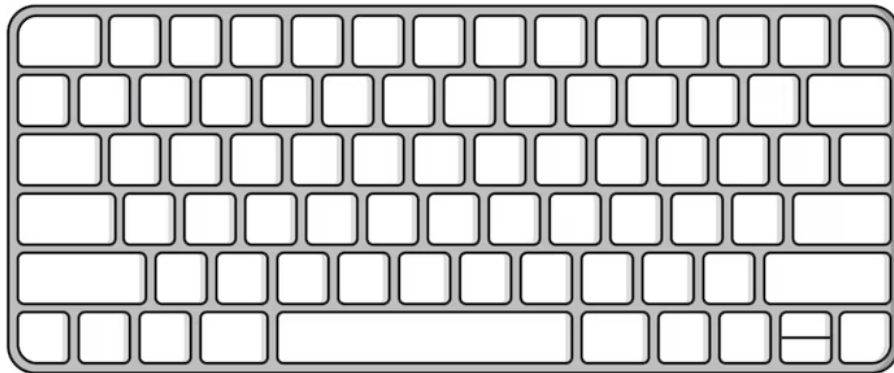


*Figura 10. Imagen Pausa. Autoridad propia, (2023)*

### **5.3.2 Interfaces de Entrada**

Las Interfaces de Entrada incluirán controles WASD para mover al personaje y el ratón para mirar alrededor y seleccionar objetos. Las teclas numéricas se utilizarán para usar habilidades o

items. Además, se implementará un sistema de “hotkeys” para permitir a los jugadores acceder rápidamente a ciertas acciones o elementos del juego.



*Figura 11. Vector gratuito dibujos animados de teclado. Juicy\_fish*



*Figura 12. Usar Microsoft Bluetooth Mouse. Microsoft*

### 5.3.3 Interfaces de Salida

Las Interfaces de Salida proporcionarán retroalimentación visual y auditiva a las acciones del jugador. Por ejemplo, cuando un jugador ataca a un enemigo, habrá una animación de ataque para el personaje del jugador, una animación de daño para el enemigo, un sonido de espada golpeando y una disminución en la barra de salud del enemigo.



Figura 13. Imagen player muerto. Autoridad propia, (2023)

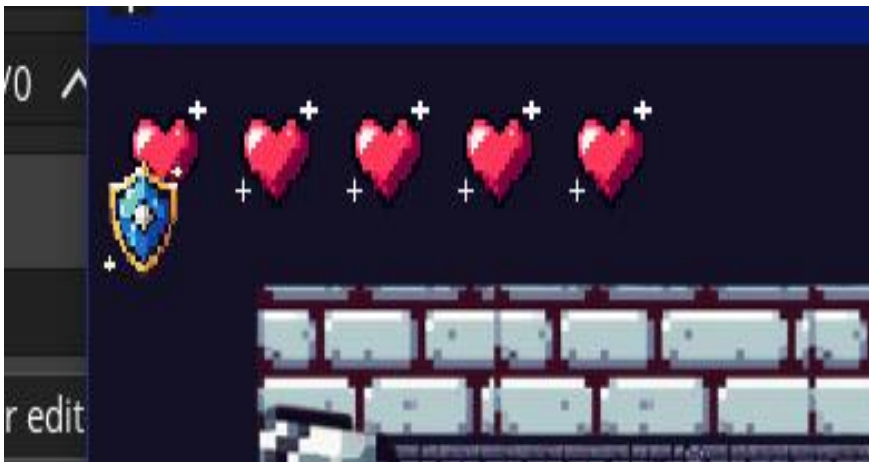


Figura 14. Escudo y vida. Autoridad propia, (2023)

## 6 implementación

## **6.1 Plataformas de desarrollo**

Gamemaker:

Version 2023.8.1.102

Lenguaje de programación Star GML

Se está empleando para el desarrollo windows 10 home

Requisitos de uso:

- 1GB de RAM
- Procesador de 1 GHZ
- GPU de 512 MB
- Almacenamiento de 1 GB

## **6.2 Base de datos**

La plataforma que nosotros estamos empleando no viene integrada con una base de datos, por tanto, nos toca por medio de código implementar nuestra propia estructura de datos para almacenar los datos empleando el lenguaje de programación ya sea usando arreglos, matrices, etc.

## **6.3 Infraestructura de hardware y redes**

Hardware empleado

Se emplearon dos portátiles para el desarrollo, cada uno con 8 GB de RAM, uno con Intel Core I5 de 12ava generación, y otro con Ryzen 5 2500, y grafica integrada respectiva a cada procesador. Ambos contaban con Windows 10 home versión 22h2.

Se empleo la herramienta de git y git hub para el manejo de versiones y avance.

El juego al no requerir internet no fue necesario emplear o integrar nada relacionado a redes.

## **7. Pruebas del software**

### **7.1 Pruebas del software**

• **Nombre de la prueba:** Prueba de la mecánica de combate con enfoque en el daño de las armas y la vida del enemigo

**Descripción de la prueba:** Esta prueba de unidad verifica la función que maneja el daño que un arma puede infligir a un enemigo. En esta prueba, se crea un personaje con un arma específica y se le hace atacar a un enemigo. La prueba verifica que el daño infligido se calcula correctamente basándose en el daño del arma y se resta correctamente de la vida total del enemigo.

**Resultado esperado:** El daño infligido por el personaje debe ser igual al daño del arma. La vida del enemigo después del ataque debe ser igual a la vida inicial del enemigo menos el daño del arma.

**Resultado obtenido:** El daño infligido por el personaje fue el esperado. La vida del enemigo después del ataque fue igual a la vida inicial del enemigo menos el daño del arma.

**Estado de la prueba:** Exitosa

- **Nombre de la prueba:** Prueba del diseño de niveles creados y desbloqueo de niveles

**Descripción de la prueba:** Esta prueba de unidad verifica la función que carga los niveles creados en tu juego y la funcionalidad de desbloqueo de niveles. En esta prueba, se carga un nivel, se simula que el jugador lo completa y se verifica que el siguiente nivel se desbloquee correctamente.

**Resultado esperado:** Al cargar un nivel, este debe corresponder al nivel precreado correspondiente. Al completar un nivel, el siguiente nivel debe desbloquearse.

**Resultado obtenido:** Al cargar un nivel, este correspondía al nivel precreado correspondiente. Al completar un nivel, el siguiente nivel se desbloqueó correctamente.

**Estado de la prueba:** Exitosa

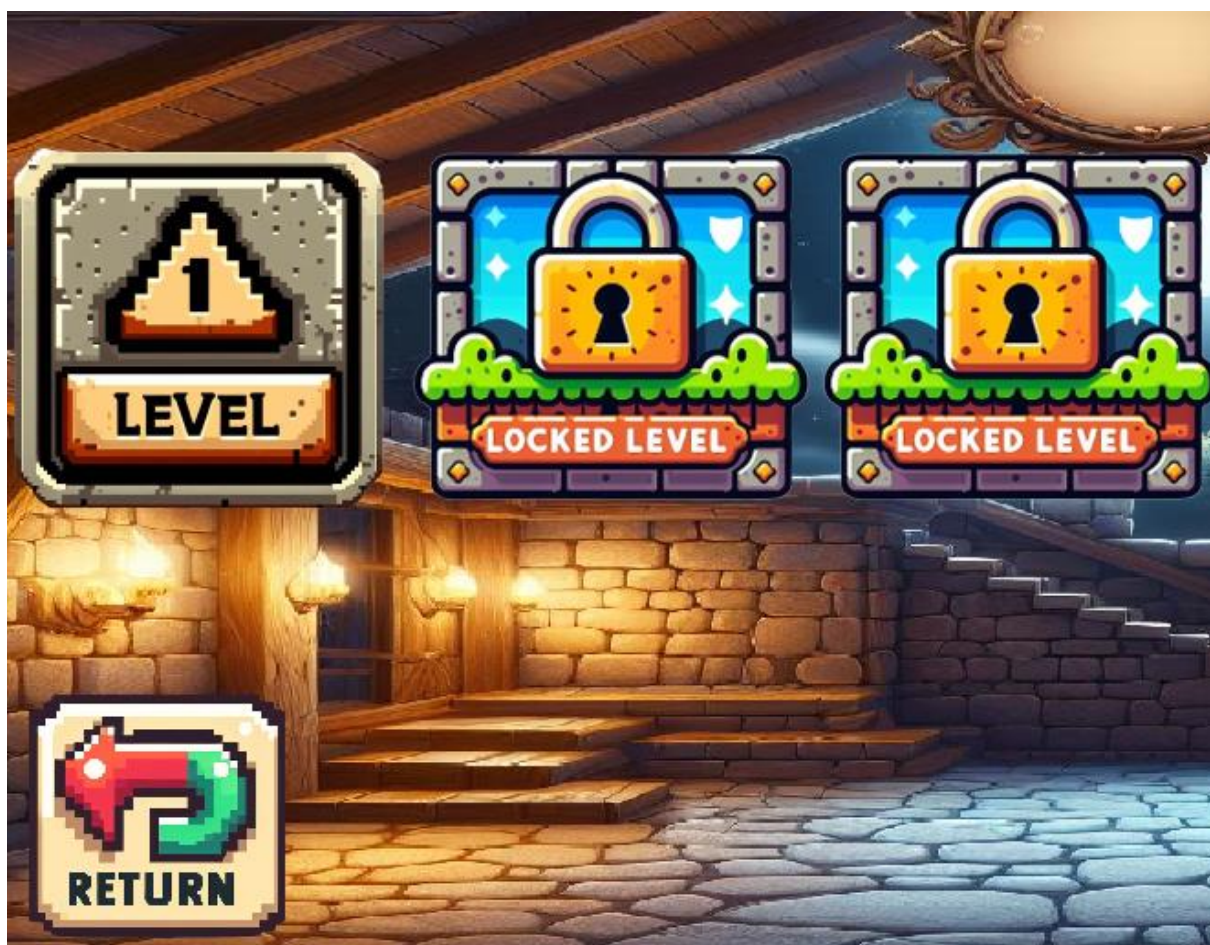


Figura 15. Niveles sala 2. Autoridad propia, (2023)



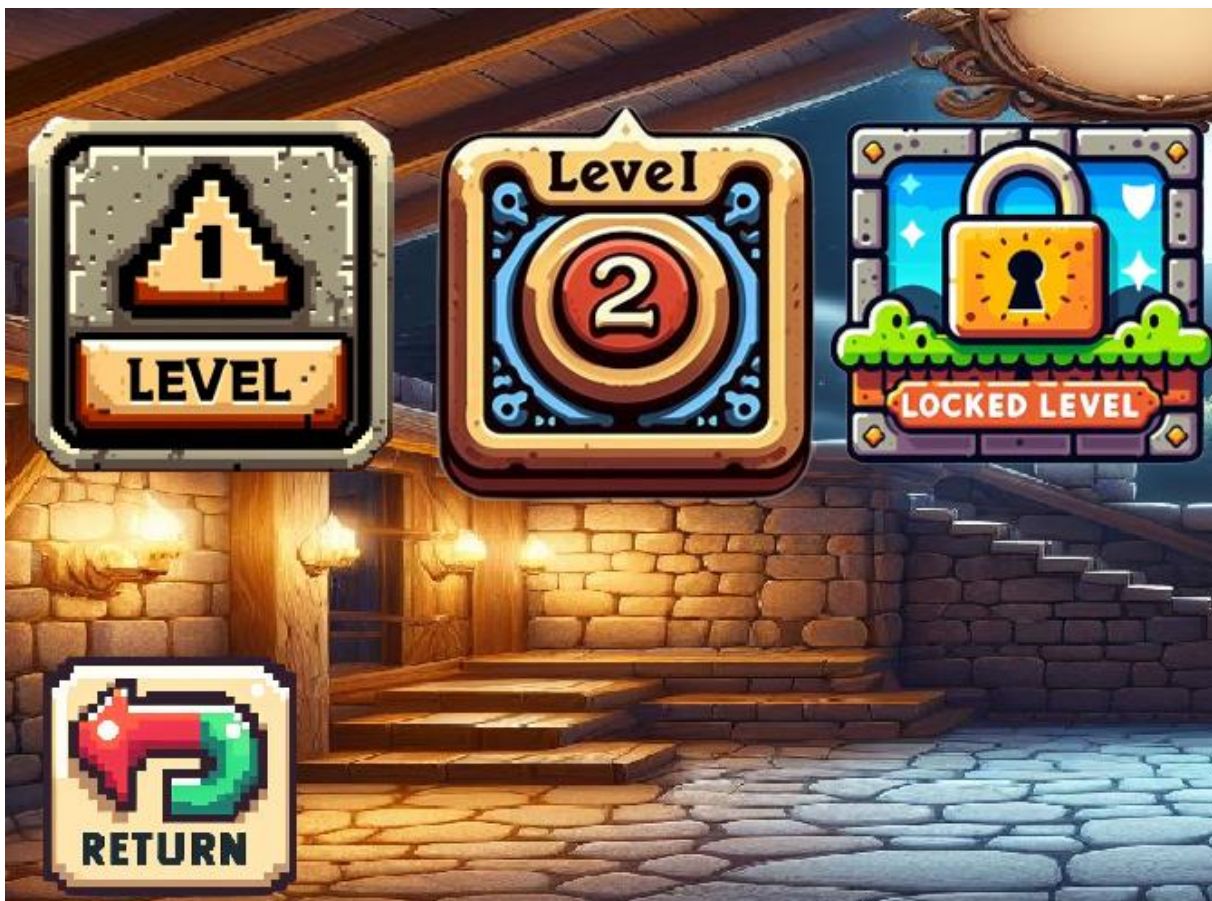


Figura 16. Nivel desbloqueado. Autoridad propia, (2023)

- **Nombre de la prueba:** Prueba de recoger y usar armas

**Descripción de la prueba:** Esta prueba de unidad verifica la función que permite al personaje recoger y usar armas en tu juego. En esta prueba, se simula que el personaje recoge un arma y luego la usa para atacar a un enemigo. La prueba verifica que el daño infligido al enemigo se calcula correctamente basándose en el daño del arma recogida.

**Resultado esperado:** Al recoger un arma, esta debe añadirse al inventario del personaje. Al usar el arma para atacar a un enemigo, el daño infligido debe ser igual al daño del arma.

**Resultado obtenido:** Al recoger un arma, esta se añadió al inventario del personaje. Al usar el arma para atacar a un enemigo, el daño infligido fue igual al daño del arma.

**Estado de la prueba:** Exitosa



Figura 17. Juego Pausado. Autoridad propia, (2023)

- **Nombre de la prueba:** Prueba de la interfaz de usuario

**Descripción de la prueba:** Esta prueba de unidad verifica que la interfaz de usuario de tu juego funcione correctamente. En esta prueba, se simula la interacción del jugador con los diferentes elementos de la interfaz de usuario, como los menús, los botones y las pantallas de inventario.

**Resultado esperado:** Todos los elementos de la interfaz de usuario deben responder correctamente a la interacción del jugador. Por ejemplo, al hacer clic en un botón, debe llevar a cabo la acción correspondiente. Además, la interfaz de usuario debe mostrar la información correcta en todo momento, como la vida del personaje y el inventario actual.

**Resultado obtenido:** Todos los elementos de la interfaz de usuario respondieron correctamente a la interacción del jugador. Al hacer clic en un botón, llevó a cabo la acción correspondiente. Además, la interfaz de usuario mostró la información correcta en todo momento, como la vida del personaje y el inventario actual.

**Estado de la prueba:** Exitosa



Figura 18. Menu Play. Autoridad propia, (2023)

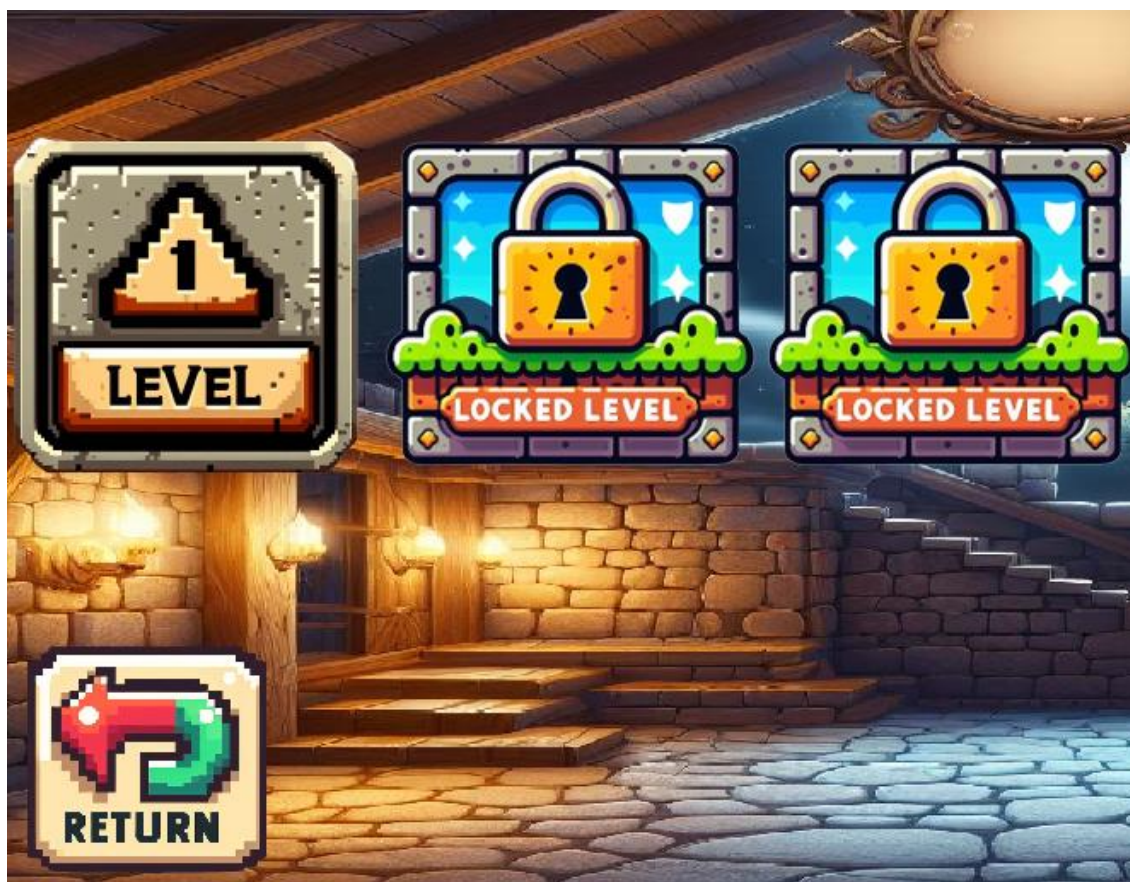


Figura 19. Niveles sala 2. Autoridad propia, (2023)

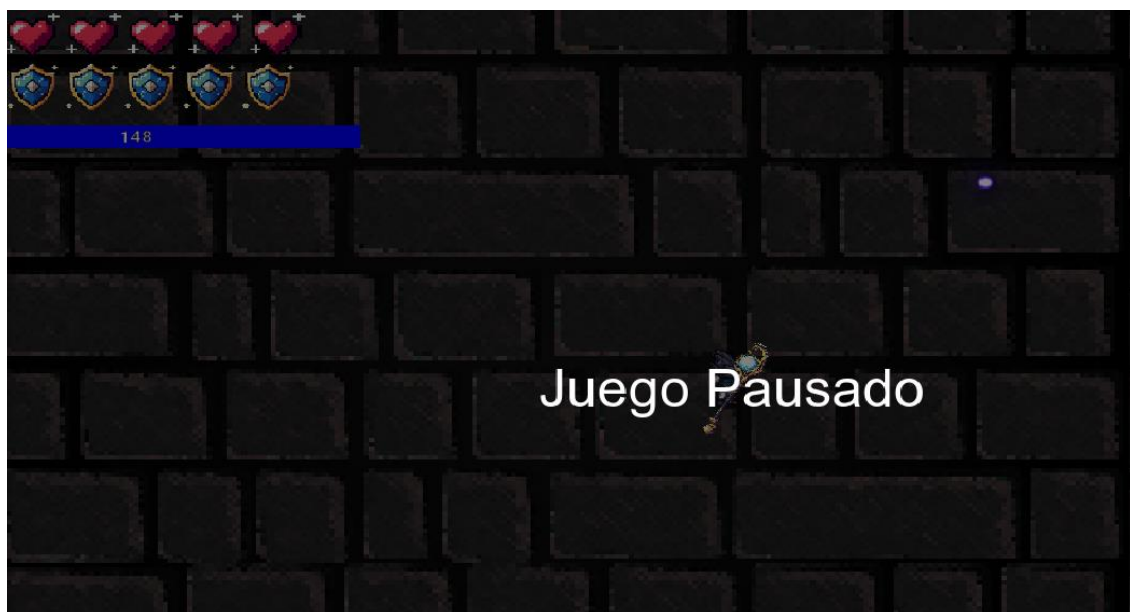


Figura 20. Juego Pausado. Autoridad propia, (2023)

## 7.2 Pruebas de Usabilidad

- **Nombre de la prueba:** Prueba de la interfaz de usuario

**Descripción de la prueba:** Esta prueba de usabilidad se realiza para evaluar la eficacia de la interfaz de usuario del juego. En esta prueba, se invita a un grupo de jugadores a interactuar con la interfaz de usuario del juego, incluyendo los menús, las pantallas de inventario y las indicaciones en pantalla durante el juego.

**Resultado esperado:** Los jugadores deberían ser capaces de navegar por la interfaz de usuario de manera intuitiva y eficiente. Deberían ser capaces de acceder a los menús, ver su inventario y entender las indicaciones en pantalla sin dificultad.

**Resultado obtenido:** Los jugadores fueron capaces de navegar por la interfaz de usuario de manera intuitiva y eficiente. Pudieron acceder a los menús, ver su inventario y entender las indicaciones en pantalla sin dificultad.

**Estado de la prueba:** Exitosa

- **Nombre de la prueba:** Prueba de accesibilidad

**Descripción de la prueba:** Esta prueba de usabilidad se realiza para evaluar la accesibilidad del juego. En esta prueba, se invita a un grupo de jugadores con diversas habilidades y necesidades a jugar el juego y a proporcionar feedback sobre su experiencia.

**Resultado esperado:** Todos los jugadores, independientemente de sus habilidades y necesidades, deberían ser capaces de jugar y disfrutar del juego. Los colores y las fuentes utilizadas en el juego deberían ser legibles para todos los jugadores. Además, los jugadores deberían ser capaces de navegar por la interfaz de usuario y de entender las indicaciones en pantalla sin dificultad.

**Resultado obtenido:** Todos los jugadores, independientemente de sus habilidades y necesidades, fueron capaces de jugar y disfrutar del juego. Los colores y las fuentes utilizadas

en el juego fueron legibles para todos los jugadores. Además, los jugadores fueron capaces de navegar por la interfaz de usuario y de entender las indicaciones en pantalla sin dificultad.

**Estado de la prueba:** Exitosa

### **Conclusiones.**

El desarrollo del videojuego “Becoming to Berserk” ha sido un proceso desafiante pero gratificante. A lo largo de este proyecto, hemos aplicado nuestros conocimientos en programación, diseño de videojuegos, y pruebas de software para crear un producto que no solo es entretenido, sino que también tiene el potencial de ayudar a jóvenes con Trastorno por Déficit de Atención e Hiperactividad (TDAH).

Nuestro videojuego, desarrollado con el motor GameMaker, presenta una mecánica de combate en tiempo real, diseño de niveles interactivos, equipamiento, y una interfaz de usuario intuitiva. A pesar de los desafíos que enfrentamos durante el desarrollo, hemos logrado implementar todas estas características de manera exitosa.

Las pruebas de software y usabilidad que realizamos fueron fundamentales para asegurar la calidad de nuestro juego. A través de las pruebas de unidad e integración, pudimos verificar que cada componente de nuestro juego funcionara correctamente tanto de manera individual como en conjunto. Las pruebas de usabilidad nos permitieron entender cómo los jugadores interactúan con nuestro juego y hacer mejoras basadas en sus comentarios.

En resumen, estamos orgullosos del trabajo que hemos realizado en “Becoming to Berserk”. Creemos que hemos creado un videojuego que no solo es divertido de jugar, sino que también puede tener un impacto positivo en la vida de los jóvenes con TDAH.

Este documento de grado no solo sirve como una descripción detallada de nuestro proyecto, sino también como una reflexión de nuestro aprendizaje y crecimiento a lo largo del desarrollo del videojuego. Esperamos que este documento pueda servir como una guía útil para otros estudiantes que estén interesados en el desarrollo de videojuegos y en la creación de productos que puedan tener un impacto positivo en la sociedad.

## Bibliografía

- Andronico, M. (2012). The Evolution of Video Game Graphics. Recuperado de: <https://www.gamedesigning.org/gaming/video-game-graphics/>
- Boyle, R. (2022). ADHD and Video Games: Is There a Link? Recuperado de: <https://www.webmd.com/add-adhd/childhood-adhd/adhd-and-video-games-is-there-a-link>
- Castellanos, K., & Fabian, D. (2019). DESARROLLO DE UN PROTOTIPO DE VIDEOJUEGO PARA LA EMPRESA LIFE IS THE GAME, EMPLEANDO UNA METODOLOGÍA ÁGIL Y UN MOTOR DE JUEGOS DE ÚLTIMA GENERACIÓN. Universidad autónoma de Bucaramanga. Recuperada de: [https://repository.unab.edu.co/bitstream/handle/20.500.12749/7269/2019\\_Tesis\\_Kevin\\_Castellanos\\_Tovar.pdf?sequence=1](https://repository.unab.edu.co/bitstream/handle/20.500.12749/7269/2019_Tesis_Kevin_Castellanos_Tovar.pdf?sequence=1)
- Desarrollo Web. (2020). ¿Qué es el modelo V? Recuperado de: Modelo V: definición, ventajas y áreas de aplicación - IONOS
- Edwards, B. (2015). The History of Video Game Consoles. How-To Geek. Recuperado de: <https://www.howtogeek.com/330346/the-history-of-video-game-consoles/>
- Gantiva, D., & Erika, G. (2016). Diseño de un videojuego educativo como material didáctico en la clase de tecnología e informática para ciclo cuatro. Universidad Pedagógica Nacional. Recuperado de: <http://repository.pedagogica.edu.co/bitstream/handle/20.500.12209/2017/TE-18996.pdf?sequence=1&isAllowed=y>
- Game maker. (2013). Características de Game maker. Recuperado de: <https://gamemaker.io/es>
- Gonzales, R. (2020). Desarrollo de un videojuego y documentación de su diseño. Universitat Politècnica de Catalunya (UPC). Recuperado de: <https://upcommons.upc.edu/bitstream/handle/2117/339364/152455.pdf?sequence=1&isAllowed=y>
- Gurwin, G. (2015). The Evolution of Video Game Music: From 8-bit to orchestral scores. Digital Trends. Recuperado de: <https://www.digitaltrends.com/gaming/evolution-of-video-game-music/>
- Hovde, M. (2022). The Link Between ADHD and Video Games. Recuperado de: <https://psychcentral.com/adhd/adhd-video-games>



- Martinez, W. (2020). DESARROLLO DE UN VIDEOJUEGO PRISON SCAPIST. UTADEO. Recuperado de: <https://expeditiorepositorio.utadeo.edu.co/bitstream/handle/20.500.12010/16381/Trabajo%20de%20grado.pdf?sequence=1&isAllowed=y>
- Matulef, J. (2018). The Evolution of Video Game Storytelling. Eurogamer. Recuperado de: <https://www.eurogamer.net/articles/2016-02-06-the-evolution-of-video-game-storytelling>
- North, J. (2020). Research Shows That Some Video Games Can Be Used as Therapy for ADHD. Recuperado de: <https://adhd.newlifeoutlook.com/games-for-adhd/>
- Pardo, D., Losada, A., & Suárez, P. (2019). Proyecto de desarrollo de videojuego, Boom Away. Universidad de los Andes. Recuperado de: <https://repositorio.uniandes.edu.co/bitstream/handle/1992/50760/22650.pdf?sequence=1>
- Pedraz, P. (2021). 10 tipos de videojuegos, sus mecánicas habituales y sus enfoques narrativos. Recuperado de: <https://www.alaluzdeunabombilla.com/2021/07/06/10-tipos-de-videojuegos-sus-mecanicas-habituales-y-sus-enfoques-narrativos/>
- Romero, G. (2014). GM - Como Hacer que la Barra Siga a la Pantalla. Recuperado de: <https://www.youtube.com/watch?v=F2VP4KjowKE>
- Sisternas, P. (2023). ¿Modelo en V sí o no para organizar proyectos? Recuperado de: Modelo en V: la clave para una gestión de proyectos eficaz (emprendepyme.net)
- Tokio School. (2020). Diseño de videojuegos. Diseño y desarrollo de videojuegos. Recuperado de: <https://www.tokioschool.com/noticias/disenio-desarrollo-videojuegos/>
- Torres, N. (2023). What is a Roguelike Game? 3 Components of Roguelikes. Recuperado de: <https://www.betterbestgaming.com/what-is-roguelike-game/>
- Universal LPC Spritesheet Generator. (2021) Recuperado de: [https://sanderfrenken.github.io/Universal-LPC-Spritesheet-Character-Generator/#?body=Body\\_color\\_fur\\_black&head=Human\\_male\\_fur\\_black&shadow=Shadow\\_shadow&sex=muscular&wound\\_arm=none&hat=Horned\\_helmet\\_iron&arms=Armour\\_steel&shoulders=Legion\\_gold&baldron=Baldron\\_slate&bracers=Bracers\\_steel&cape=Tattered\\_bluegray&legs=Pants\\_slate&shoes=Armour\\_iron&weapon=none](https://sanderfrenken.github.io/Universal-LPC-Spritesheet-Character-Generator/#?body=Body_color_fur_black&head=Human_male_fur_black&shadow=Shadow_shadow&sex=muscular&wound_arm=none&hat=Horned_helmet_iron&arms=Armour_steel&shoulders=Legion_gold&baldron=Baldron_slate&bracers=Bracers_steel&cape=Tattered_bluegray&legs=Pants_slate&shoes=Armour_iron&weapon=none)

## Anexos

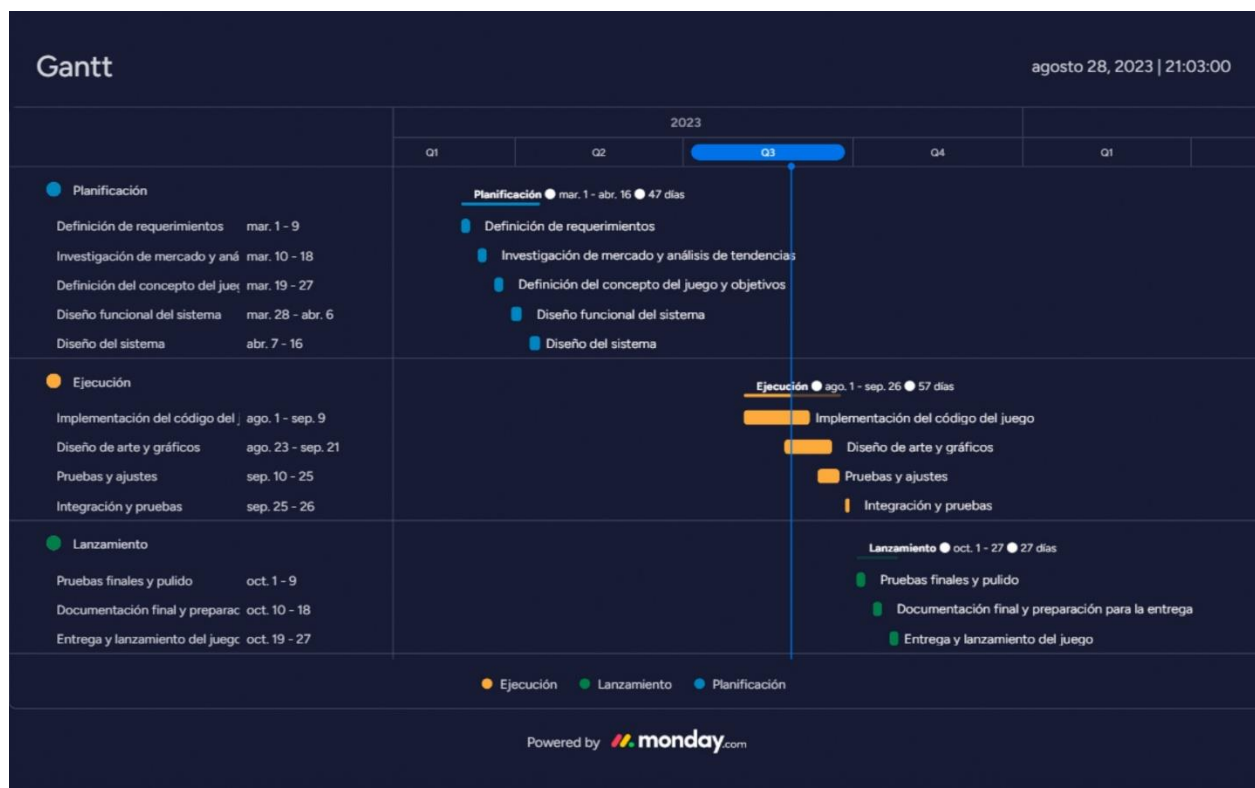


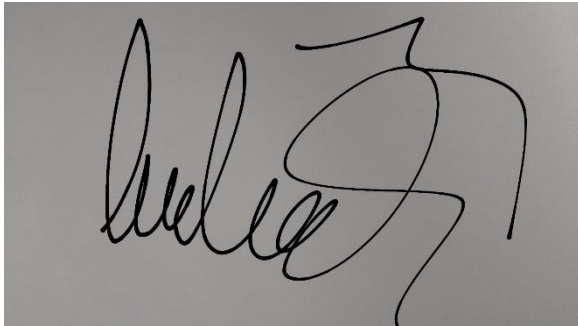
Figura 21. Diagrama Gantt. Autoridad propia, (2023)

Entrevista a persona con TDAH

Nombre: Natalia Montenegro Valencia

Edad: 19

Autorización por firma para usar sus datos:

A handwritten signature in black ink on a light gray background. The signature is stylized and appears to read 'Natalia Montenegro Valencia'.

¿Como te sientes al comenzar el juego?

Es un poco extraña la sensación por el ambiente que da el primer nivel sin mucho que ver al inicio pero me siento tenso que me hace sentirme alerta

¿Encuentras fácil de entender la interfaz del juego?

Se ve bastante interesante solo que las instrucciones quedan un poco confusas, pero por lo demás me agrada

¿Como describirías la dificultad del juego?

Al inicio me es fácil por la cantidad de enemigos sin embargo a medida que voy subiendo de nivel me cuesta coordinarme a la hora de pasar el nivel

¿Crees que el nivel de dificultad es adecuado para ti?

De momento me resulta complicado, pero sé que si le doy un poco más de tiempo al juego me resultara más fácil jugarlo

¿Puedes mantenerte concentrado/a durante un periodo prolongado de tiempo?

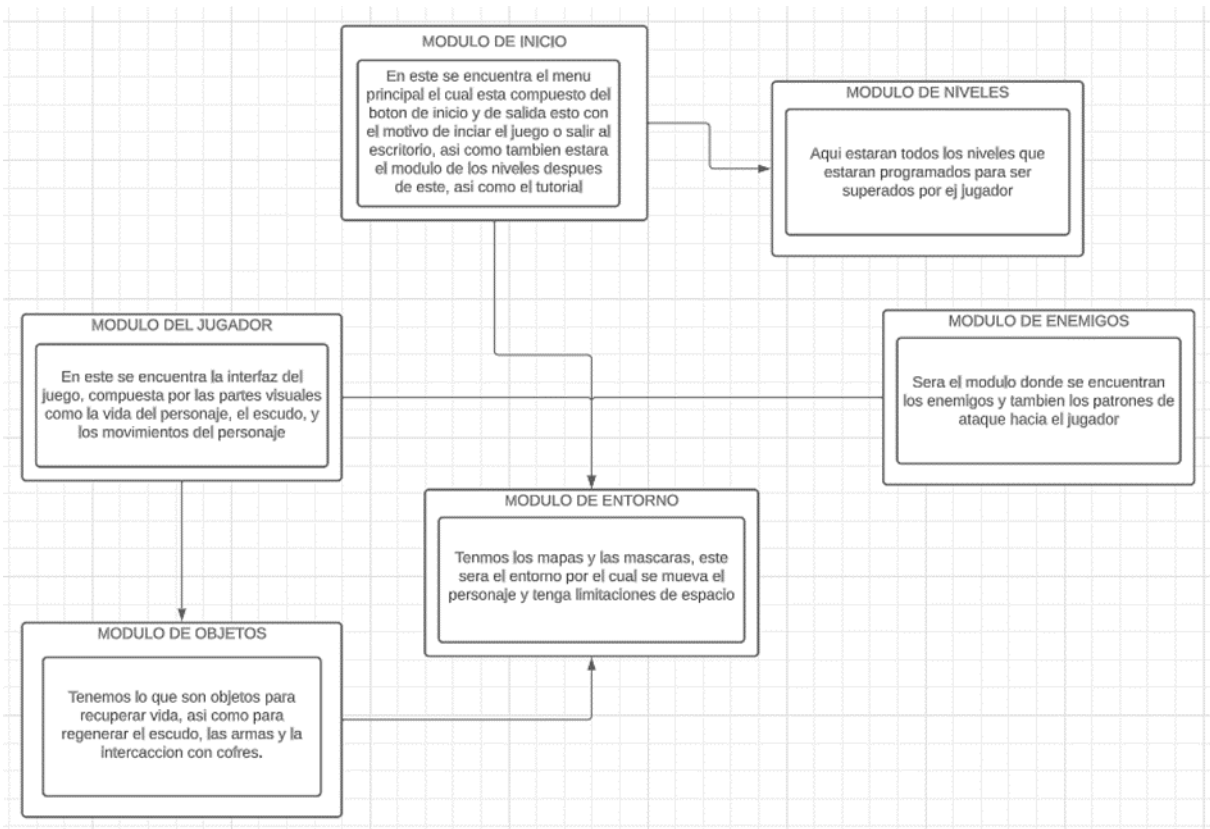
La verdad por el tema de que me mantiene tensa si, resulta entretenido por el tema de mantenerse expectante de lo que pueda suceder

¿Como te hace sentir la ambientación de las salas para la concentración?

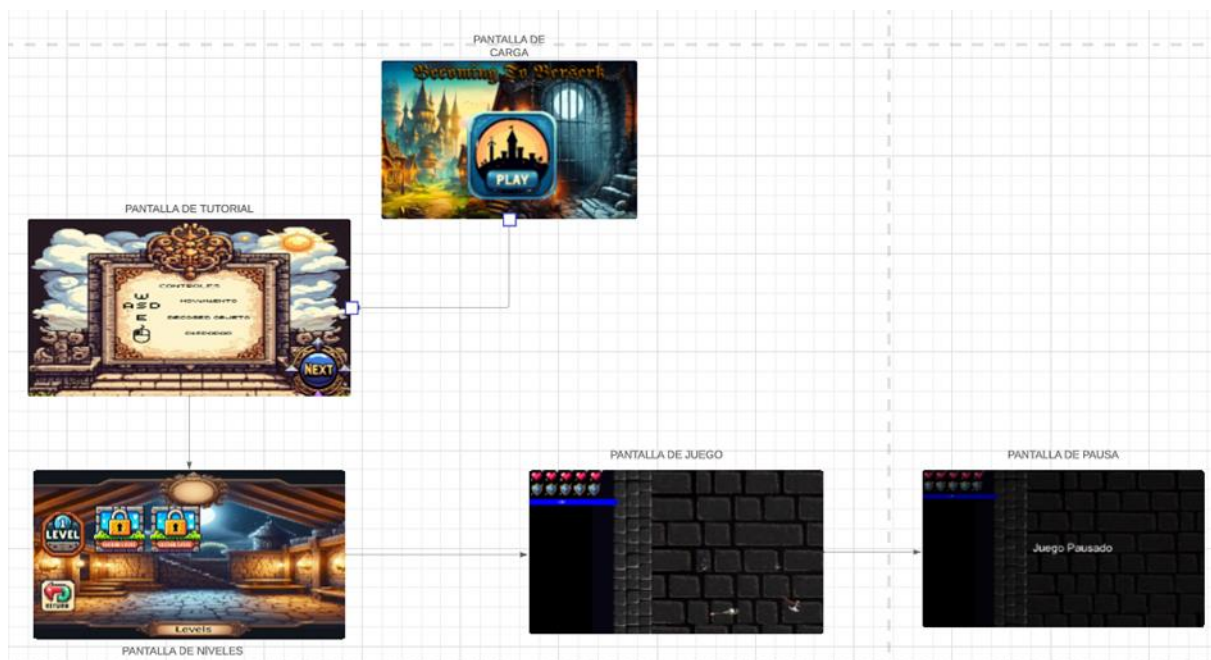
No sé muy bien cómo explicarlo, pero en parte ayuda que las salas tengan cierta temática para tener una mayor inmersión

## 2 MAPA DEL SISTEMA

### 2.1 Modelo Lógico



## 2.2 Navegación



### 3 DESCRIPCIÓN DEL SISTEMA

Nuestro videojuego maneja de manera sencilla el ingreso de inputs a su vez como el manejo de la información, de esta manera será bastante sencillo familiarizarse con estos. De esta forma se manejarán entornos sencillos para no complicar su uso.

#### 3.1 Subsistemas

Subsistema de inicio: En este se almacena el proceso para iniciar el juego

Subsistema de niveles: En este a medida que se completa cada nivel se irán desbloqueando progresivamente cada uno

Subsistema de interacciones: Con este se manejarán cada uno de los movimientos del jugador interactuando con los inputs por el teclado y el ratón

Subsistema de enemigos: Se manejan los ataques y las colisiones con el enemigo

##### 3.1.1 Pantallas

Pantalla de inicio:

En este tenemos a lo que se refiere con la primera interacción del jugador, en el cual le podrá dar inicio al juego

Pantalla del tutorial:

Se encuentran las instrucciones de los controles del jugador, con este podrá interactuar directamente con el software

Pantalla de niveles:

Con este podrá seleccionar el nivel del juego y con este avanzar en la historia

Pantalla de juego:

Aquí se encuentran todas las interacciones que tiene el jugador directamente con el entorno del juego, entre estas relaciones están las interacciones con los enemigos, las armas para avanzar en el juego, a su vez como lo son los ítems que siguen al jugador como los son las vidas, el escudo y a su vez la barra de mana de la cual se extraerán los disparos de las armas

### 3.1.2 Mensajes de error

Este al ser un videojuego no habrá pantallas de errores, esto debido a que los únicos errores de ejecución serán por el hardware o software incompatible, esto ya esta especificado en preguntas frecuentes cuales serán los requisitos mínimos de ejecución

### 3.1.3 Ayudas contextuales

Para estas se podrán basar en diferentes juegos del estilo de RPG o role play en los cuales se podrá ir familiarizando con este tipo de videojuegos, en los que se manejan estilos de 2D a su vez como temáticas sencillas con mecánicas fáciles de entender

#### 4 FAQ

¿En qué sistemas operativos se podrá usar?

En Windows, por temas de implementación será de uso único en Windows en sus versiones de Windows 10 o mayor

¿Hay contenido adicional?

No, nuestro juego tiene un lineamiento muy dirigido, a lo que nos referimos es que tiene una finalidad fija y no habrá secuelas u algo parecido

¿Qué estrategias son buenas para el juego?

De esto ya dependerá como el jugador quiere pasar la experiencia, ya sea intentando hacer partidas de no hits, no usar armas etc. Ya será como el jugador presienta que le sea la manera más cómoda de jugarlo.



La efectividad de este proyecto depende en gran medida de la atención, paciencia, concentración y capacidad de manejar varios puntos simultáneamente. Si la demanda de estos aspectos es elevada, es posible que el proyecto no sea adecuado. Sin embargo, si la exigencia es mínima y el proyecto cuenta con una guía paso a paso que utiliza un lenguaje concreto, claro y sencillo, además de contar con un flujo constante y un tiempo bien estructurado sin muchos distractores, podría ser una elección apropiada. Es crucial tener en cuenta estos factores, ya que un proyecto con demasiados distractores o una longitud excesiva podría no ser apropiado para estas personas, llevándolas a la frustración en lugar del éxito.

Aleyda león Sánchez

Psicóloga y directora del área de recursos humanos

*Aleyda León Sánchez*  
Psicóloga  
+569 95475648  
aleyda.leon.s@gmail.com