WIZRAT

Imagen que contiene exterior, tabla, vivo, colorido

Descripción generada automáticamente

DAM 2º Año IES Emilio Jimeno

02/05/2024 Riquelmer Añez Joffre

Tutor: Joaquín

###### 2. Índice

[3. Descripción del Proyecto 1](#_Toc1634893476)

[3.1 Resumen – Abstract (español e inglés) 1](#_Toc1877046394)

[3.2 Contexto del Proyecto 1](#_Toc399948793)

[3.2.1 Ámbito y Entorno 1](#_Toc663181191)

[3.2.2 Análisis de la Realidad 2](#_Toc1829756350)

[3.2.3 Solución y Justificación de la Solución Propuesta 3](#_Toc695413800)

[3.2.4 Destinatarios 3](#_Toc1084218172)

[3.3 Objetivo del Proyecto 3](#_Toc520246389)

[4. Documento de Acuerdo del Proyecto 4](#_Toc2140786039)

[4.1 Tareas 4](#_Toc1824326504)

[4.2 Planificación Temporal de Tareas 4](#_Toc321135201)

[4.3 Metodología de Desarrollo 4](#_Toc1959591081)

[4.4 Historias de Usuario 5](#_Toc110789689)

[Historia de Usuario 01: Movimiento y Ataque del Personaje 5](#_Toc1608293402)

[Historia de Usuario 02: Uso del Escudo 5](#_Toc1277119379)

[Historia de Usuario 03: Recolección de Ítems y Corazones Azules 5](#_Toc788811997)

[Historia de Usuario 04: Puntuación 5](#_Toc1660791903)

[Historia de Usuario 05: Control de Volumen 5](#_Toc893681625)

[Historia de Usuario 06: Compatibilidad y Rendimiento 6](#_Toc401581677)

[Historia de Usuario 07: Animación 6](#_Toc1768658269)

[Historia de Usuario 08: Escalada 6](#_Toc398807896)

[4.5 Presupuesto 6](#_Toc496205803)

[4.6 Contrato/Pliego de Condiciones 7](#_Toc1901868964)

[4.7 Análisis de Riesgos (DAFO) 7](#_Toc793041929)

[5. Documento de Análisis y Diseño 7](#_Toc1779495632)

[5.1 Modelado de Datos 7](#_Toc1797558586)

[5.2 Análisis y Diseño del Sistema Funcional 8](#_Toc1444372849)

[5.2.1 Diagrama de Casos de Uso 8](#_Toc1283592508)

[5.2.2 Diagrama de Clases 9](#_Toc1501697612)

[5.2.3 Diagrama de Secuencia 10](#_Toc1957323525)

[5.2.4 Diagrama de Estados 12](#_Toc331521392)

[Diagrama de estados 12](#_Toc378156543)

[5.3 Análisis y Diseño de la Interfaz de Usuario 13](#_Toc875517597)

[5.4 Diseño de la Arquitectura de la Aplicación 15](#_Toc1837665426)

[5.4.1 Tecnologías/Herramientas Usadas 15](#_Toc970746716)

[5.4.2 Arquitectura de Componentes de la Aplicación 15](#_Toc2012409226)

[5.4.3 Diagrama de Componentes: 16](#_Toc1500834727)

[6. Documento de Implementación e Implantación del Sistema 16](#_Toc2132817223)

[6.1 Implementación 17](#_Toc1824016073)

[6.1.2 Configuración del Proyecto en Unity 17](#_Toc1226826394)

[6.1.3 Explicación del funcionamiento de los scripts. 17](#_Toc1164338847)

[6.1.4 Implementación del Personaje Principal 18](#_Toc789562435)

[6.1.5 Ataque con la Espada: 18](#_Toc1866147756)

[6.2 Pruebas 19](#_Toc1104129135)

[Pruebas Unitarias 19](#_Toc2100974138)

[Pruebas Integrales 19](#_Toc124632593)

[Debugging y Optimización 19](#_Toc546669607)

[7. Documento de Cierre 20](#_Toc1877484322)

[7.1 Instalación y Configuración 20](#_Toc725568292)

[7.2 Manual de Usuario 20](#_Toc168061166)

[7.3 Resultados Obtenidos y Conclusiones 22](#_Toc483857785)

[7.4 Diario de Bitácora 22](#_Toc621140371)

[8. Bibliografía 23](#_Toc479096548)

[9. Anexos 24](#_Toc1495289654)

[Contrato de desarrollo de software 24](#_Toc935514524)

[PRIMERA. Objeto del Contrato 25](#_Toc945631670)

[SEGUNDA. Funcionalidades del Software 26](#_Toc2064073061)

[TERCERA. Precio y forma de pago 26](#_Toc160756936)

[CUARTA. Intereses de demora 26](#_Toc1856613787)

[QUINTA. Originalidad del Software 27](#_Toc2097579271)

[SEXTA. Plazo de ejecución 27](#_Toc100582506)

[SÉPTIMA. Modificación del Software 28](#_Toc1422131739)

[OCTAVA. Propiedad del Software 28](#_Toc2009355797)

[NOVENA. Cesión del Contrato. Subcontratación 29](#_Toc2115757068)

[DÉCIMA. Inexistencia de relación laboral 29](#_Toc459054626)

[DECIMOPRIMERA. Cumplimiento normativo aplicable 30](#_Toc125422578)

[DECIMOSEGUNDA. Entrega del Software 30](#_Toc481942209)

[DECIMOTERCERA. Garantía del Software 30](#_Toc1012757260)

[DECIMOCUARTA. Gastos 31](#_Toc1028714464)

[DECIMOQUINTA. Elevación a público del Contrato 31](#_Toc1660199821)

[DECIMOSEXTA. Fuerza mayor 31](#_Toc58361413)

[DECIMOSÉPTIMA. Obligación de secreto y confidencialidad 31](#_Toc101612829)

[DECIMOCTAVA. Inexistencia de renuncia 32](#_Toc265843061)

[DECIMONOVENA. Terminación anticipada del Contrato 32](#_Toc1228936023)

[VIGÉSIMA. Incumplimiento del Contrato 33](#_Toc280532053)

[VIGESIMOPRIMERA. Exigibilidad 33](#_Toc1852220358)

[VIGESIMOSEGUNDA. Integridad del Contrato 34](#_Toc1213674202)

[VIGESIMOTERCERA. Protección de datos 34](#_Toc2124572905)

[VIGESIMOCUARTA. Ley aplicable y jurisdicción competente 34](#_Toc2039170929)

## 3. Descripción del Proyecto

### 3.1 Resumen – Abstract (español e inglés)

**Resumen en español:**

Este proyecto consiste en el desarrollo de un videojuego en 2D utilizando Unity. El protagonista del juego es un ratón mágico que debe derrotar enemigos mágicos para prevenir la corrupción del mundo. El juego se organiza por niveles y hay ítems recogibles que aumentan la potencia de salto y velocidad del personaje, además de corazones azules que permiten curarse. El combate se basa en el uso de un escudo, y una espada que realiza un tajo vertical. La mayoría de los elementos del juego han sido creados por el autor, excepto el Sprite del jefe final y algunos bloques y fondos.

**Abstract in English:**

This project involves the development of a 2D video game using Unity. The game's protagonist is a magical mouse who must defeat magical enemies to prevent world corruption. The game is level-based and there are collectible items that increase the character's jumping power and speed, as well as blue hearts for healing. Combat revolves around using a shield and a sword that performs a vertical slash. Most of the game's elements were created by the author, except for the sprite of the final boss and some blocks and backgrounds.

### 3.2 Contexto del Proyecto

#### 3.2.1 Ámbito y Entorno

**Introducción al videojuego:**

El videojuego, titulado "Wizrat", es una aventura en 2D desarrollada en Unity. La trama se centra en un ratón mágico que debe enfrentar y vencer a diversos enemigos mágicos para prevenir la corrupción del mundo. El juego está ambientado en un mundo fantástico con gráficos coloridos y animaciones fluidas, diseñado para atraer tanto a jugadores jóvenes como adultos que disfrutan de juegos de plataformas y acción.

**Plataformas:** Es uno de los géneros de videojuegos más antiguo, para cumplir los requisitos de este género tiene que ser un juego en el que te puedas mover en todas las direcciones dentro de las dimensiones del juego, saltar y combatir enemigos mientras se recogen objetos para completar el juego. Un videojuego de plataformas no necesariamente tiene que ser 2D, como ejemplo Celeste3D, o Super Mario 64.

**Acción:** Es un género de videojuegos bastante amplio, engloba todos los que obligan al jugador a usar toda su destreza y reflejos para poder avanzar en el juego. En este tipo de juego es muy importante la curva de dificultad porque es sencillo que el jugador se canse si el desafío es desproporcionado.

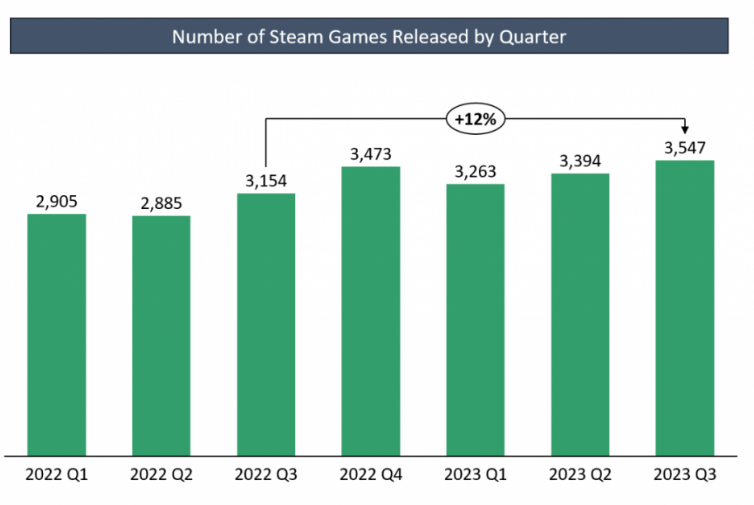
**Ambiente y espacio para el que se crea:** El juego está diseñado para ser jugado en computadoras personales. Los niveles están ambientados en diversos escenarios fantásticos, incluyendo bosques encantados. Cada escenario está cuidadosamente diseñado para proporcionar una experiencia inmersiva y desafiante, con obstáculos únicos y enemigos específicos de cada nivel.

**Cámara:** El juego está diseñado con una cámara de desplazamiento horizontal hacia la derecha, la usada por videojuegos 2D de plataformas.

#### 3.2.2 Análisis de la Realidad

**Necesidades actuales que llevan a la creación del juego:**

Hoy hay un [crecimiento en el mercado](https://fungies.io/q3-2023-steam-market-report-record-breaking-q3-performance/) de videojuegos indie. " Wizrat " responde a esta oportunidad al ofrecer una narrativa envolvente y un sistema de juego innovador que combina elementos de acción y aventura. Además, la creación del juego proporciona una plataforma para explorar y aplicar conocimientos de desarrollo de videojuegos, diseño gráfico y programación.



[[1]](#footnote-2)Gráfica del crecimiento indie en los videojuegos

**Descripción de aplicaciones similares:**

En el mercado existen varios juegos que comparten características con " Wizrat ", como "Hollow Knight" y "Ori and the Blind Forest", que también presentan protagonistas en mundos mágicos enfrentándose a diversos enemigos. Sin embargo, " Wizrat " se diferencia por su protagonista único y su combinación específica de mecánicas de juego, como el uso de un escudo y una espada, así como los ítems que modifican las habilidades del personaje.

#### 3.2.3 Solución y Justificación de la Solución Propuesta

**Propósito del videojuego:**

El propósito principal de " Wizrat " es proporcionar una experiencia de juego entretenida y desafiante, mientras se exploran conceptos de programación y diseño en Unity. El juego está diseñado para ser accesible pero también desafiante, con una curva de aprendizaje que recompensa la habilidad y la estrategia.

**Razones por las cuales el juego es mejor que las alternativas actuales:**

" Wizrat " ofrece una experiencia de juego única con su combinación de mecánicas de combate y recolección de ítems. El enfoque en un protagonista mágico y el diseño de niveles detallados proporcionan una experiencia fresca y atractiva que no se encuentra en otros juegos. Además, la capacidad de personalizar y mejorar las habilidades del personaje añade una capa adicional de profundidad al juego.

#### 3.2.4 Destinatarios

**Usuarios finales:**

El juego está dirigido a una audiencia amplia, incluyendo tanto a jugadores casuales como a aquellos que buscan un desafío más profundo. Es particularmente atractivo para los fans de plataformas y aventuras, y para los interesados en la estética y narrativa de mundos fantásticos.

### 3.3 Objetivo del Proyecto

**Razones educativas, sociales, personales e informáticas para realizar el juego:**

* **Educativas:** El desarrollo del juego proporciona una oportunidad de aplicar y expandir conocimientos en desarrollo de software, diseño gráfico, y uso del motor de videojuegos Unity.
* **Sociales:** El juego fomenta la creatividad y el pensamiento estratégico en los jugadores, además de proporcionar un producto que puede ser disfrutado por una amplia comunidad de jugadores.
* **Personales:** El proyecto permite al autor expresar su creatividad y habilidades técnicas, además de proporcionar una plataforma para demostrar su capacidad en el desarrollo de videojuegos.
* **Informáticas:** El proyecto implica usar diversas tecnologías y metodologías de desarrollo de software, proporcionando una valiosa experiencia práctica en informática y desarrollo de videojuegos.

## 4. Documento de Acuerdo del Proyecto

### 4.1 Tareas

Listado de actividades y tareas necesarias para desarrollar el videojuego " Wizrat ":

1. **Diseño del Juego**
   * a) Conceptualización de la historia y personajes.
   * b) Diseño de niveles y escenarios.
   * c) Diseño de mecánicas de juego (movimiento, combate, recolección de ítems).
2. **Desarrollo del Juego**
   * a) Configuración del proyecto en Unity.
   * b) Implementación del sistema de movimiento del personaje.
   * c) Implementación del sistema de combate (escudo y espada).
   * d) Programación de enemigos y su comportamiento.
   * e) Desarrollo de los ítems recogibles y sus efectos.
   * f) Implementación de los niveles y su progresión.
3. **Diseño Gráfico**
   * a) Creación de sprites para el personaje principal y enemigos.
   * b) Diseño de fondos y bloques para los niveles.
   * c) Integración de sprites y gráficos proporcionados por terceros (jefe final, algunos bloques y fondos).
4. **Pruebas y Debugging**
   * a) Pruebas unitarias de las mecánicas de juego.
   * b) Pruebas integrales de los niveles.
   * c) Debugging y optimización del código.
   * d) Feedback y ajustes según pruebas de usuarios.
5. **Documentación**
   * a) Redacción del manual de usuario.
   * b) Creación del documento de instalación y configuración.
   * c) Registro del desarrollo en el diario de bitácora.

### 4.2 Planificación Temporal de Tareas

La planificación temporal de las tareas se organizará en un diagrama de Gantt para visualizar la distribución del trabajo a lo largo del tiempo. Aquí se proporciona un esquema básico:

Gráfico

Descripción generada automáticamente[[2]](#footnote-3)Diagrama de Gantz

### 4.3 Metodología de Desarrollo

Para el desarrollo de " Wizrat ", se utilizará la metodología ágil Scrum. La elección de Scrum se debe a su flexibilidad y capacidad de adaptación a los cambios y necesidades del proyecto. Las principales características de esta metodología son:

* **Sprints:** El proyecto se dividirá en ciclos de desarrollo cortos, denominados sprints, que permitirán entregar partes funcionales del juego de manera incremental.
* **Reuniones Diarias:** Se realizarán reuniones diarias para evaluar el progreso y ajustar las tareas según sea necesario.
* **Revisión y Retrospectiva:** Al final de cada sprint, se revisarán los resultados obtenidos y se realizarán ajustes para el siguiente sprint.

#### **4.4 Historias de Usuario**

##### Historia de Usuario 01: Movimiento y Ataque del Personaje

**Como usuario, quiero que el personaje principal pueda moverse ágilmente y atacar para enfrentar enemigos y obstáculos.**

* **Criterio de Aceptación 01:** El personaje principal se mueve en las cuatro direcciones con fluidez.
* **Criterio de Aceptación 02:** El personaje principal puede atacar con la espada utilizando un tajo vertical.

##### Historia de Usuario 02: Uso del Escudo

**Como usuario, quiero poder usar un escudo para protegerme de ataques enemigos.**

* **Criterio de Aceptación 01:** El escudo bloquea ataques.

##### Historia de Usuario 03: Recolección de Ítems y Corazones Azules

**Como usuario, quiero recoger ítems que mejoren mis habilidades y corazones azules que curen al personaje para progresar en el juego.**

* **Criterio de Aceptación 01:** Los ítems recogidos aumentan la potencia de salto y la velocidad del personaje.
* **Criterio de Aceptación 02:** Los corazones azules curan al personaje.

##### Historia de Usuario 04: Puntuación

**Como usuario, quiero ganar puntos al derrotar enemigos y completar niveles para aumentar mi score principal.**

* **Criterio de Aceptación 01:** Los enemigos proporcionan puntos al ser derrotados.
* **Criterio de Aceptación 02:** Cada nivel completado suma puntos al score principal.

##### Historia de Usuario 05: Control de Volumen

**Como usuario, quiero poder controlar el volumen del juego para una experiencia de audio personalizada.**

* **Criterio de Aceptación 01:** La interfaz permite controlar el volumen del videojuego.

##### Historia de Usuario 06: Compatibilidad y Rendimiento

**Como usuario, quiero que el juego sea compatible con mis dispositivos y tenga un rendimiento óptimo para disfrutar de una mejor experiencia de juego.**

* **Criterio de Aceptación 01:** El juego es compatible con dispositivos Windows y HTML5.
* **Criterio de Aceptación 02:** El juego carga en menos de 10 segundos.
* **Criterio de Aceptación 03:** La tasa de frames por segundo (FPS) es constante y no inferior a 30 FPS.
* **Criterio de Aceptación 04:** El juego tiene una interfaz de usuario intuitiva y fácil de usar.

##### Historia de Usuario 07: Animación

**Como usuario, quiero que el movimiento y el combate sean perceptibles mediante animación.**

* **Criterio de Aceptación 01:** Las animaciones de movimiento son fluidas y coherentes.
* **Criterio de Aceptación 02:** Las animaciones de combate son fluidas y coherentes.
* **Criterio de Aceptación 03:** Las animaciones de enemigos hacen que entiendas que tipo de patrón de movimiento/ataque van a seguir.
* **Criterio de Aceptación 04:** Las animaciones de elementos son óptimas y dan a entender la función de cada elemento.

##### Historia de Usuario 08: Escalada

**Como usuario, quiero que el personaje pueda escalar paredes necesarias para pasar de nivel.**

* **Criterio de Aceptación 01:** El personaje puede escalar paredes específicas.

#### **4.5 Presupuesto**

|  |  |  |  |
| --- | --- | --- | --- |
| **Concepto** | **Horas** | **Valor por Hora (€)** | **Costo Total (€)** |
| Desarrollo y programación | 200 | 8,7 | 1740 |
| Diseño gráfico | 100 | 8,7 | 870 |
| Pruebas y debugging | 50 | 8,7 | 435 |
| Documentación | 20 | 8,7 | 174 |
| **Total Esfuerzo Humano** | 0 | 0 | **3219** |
| Licencias de software | 0 | 0 | 20 |
| Publicación en plataformas | 0 | 0 | 0 |
| **Total Herramientas y Recursos** | 0 | 0 | **20** |
| **Costo Total del Proyecto** | 0 | 0 | **3239** |

#### **4.6 Contrato/Pliego de Condiciones**

El pliego de condiciones incluirá los términos y condiciones de uso del juego, derechos de autor, y las responsabilidades del desarrollador y los usuarios. Este documento será adaptado de un modelo encontrado en línea, ajustado específicamente a " Wizrat ".

#### **4.7 Análisis de Riesgos (DAFO)**

**Debilidades:**

* Limitaciones técnicas del desarrollador.
* Recursos gráficos limitados.

**Amenazas:**

* Alta competencia en el mercado de juegos independientes.
* Posibles problemas técnicos no previstos.

**Fortalezas:**

* Jugabilidad casual entretenida a la par que desafiante.
* Flexibilidad y creatividad en el diseño.

**Oportunidades:**

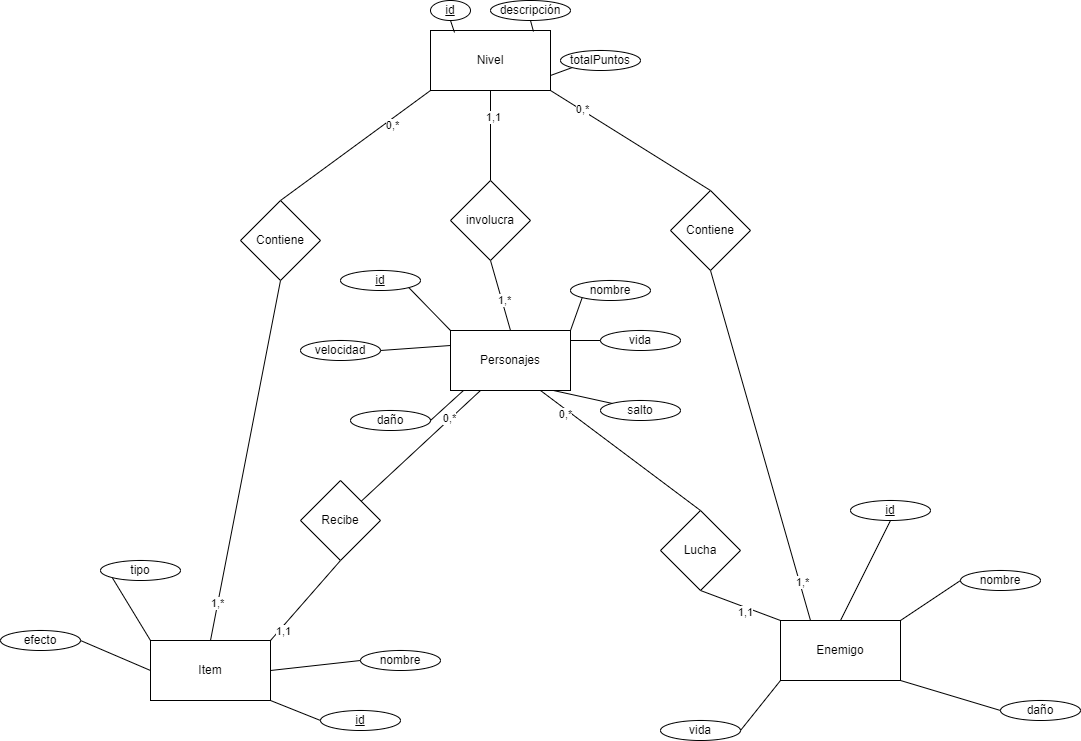
* Creciente demanda de juegos independientes.
* Posibilidad de expansión y mejoras futuras.

## 5. Documento de Análisis y Diseño

### 5.1 Modelado de Datos

Para el modelado de datos del videojuego " Wizrat ", utilizaremos un diagrama de entidad-relación (ER) que describa las principales entidades y sus relaciones. Las entidades clave son el Personaje, los Enemigos, los Ítems y los Niveles.

**Diagrama ER:**

[[3]](#footnote-4)Diagrama Entidad Relación

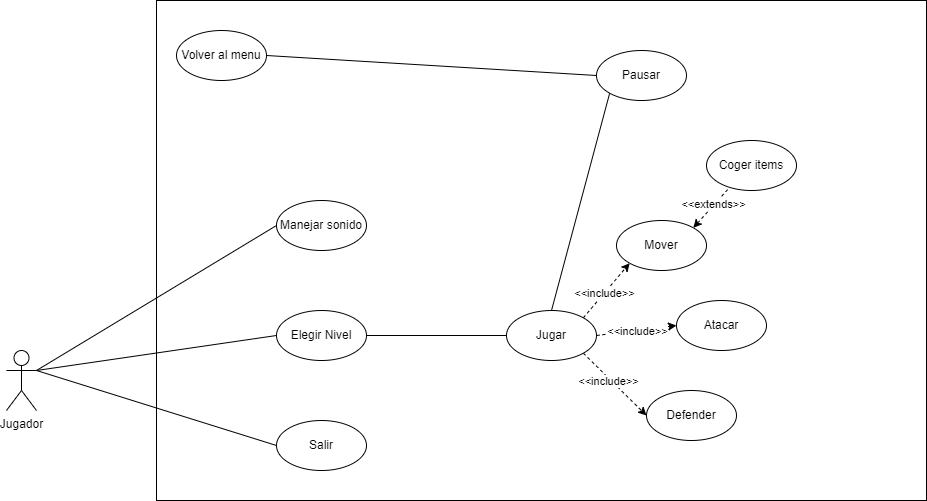
### 5.2 Análisis y Diseño del Sistema Funcional

Para el diseño del sistema funcional, utilizaremos varios diagramas UML que ayudarán a visualizar las diferentes partes del sistema. Incluiremos diagramas de casos de uso, clases y secuencia.

#### 5.2.1 Diagrama de Casos de Uso

**Actores:**

* Jugador

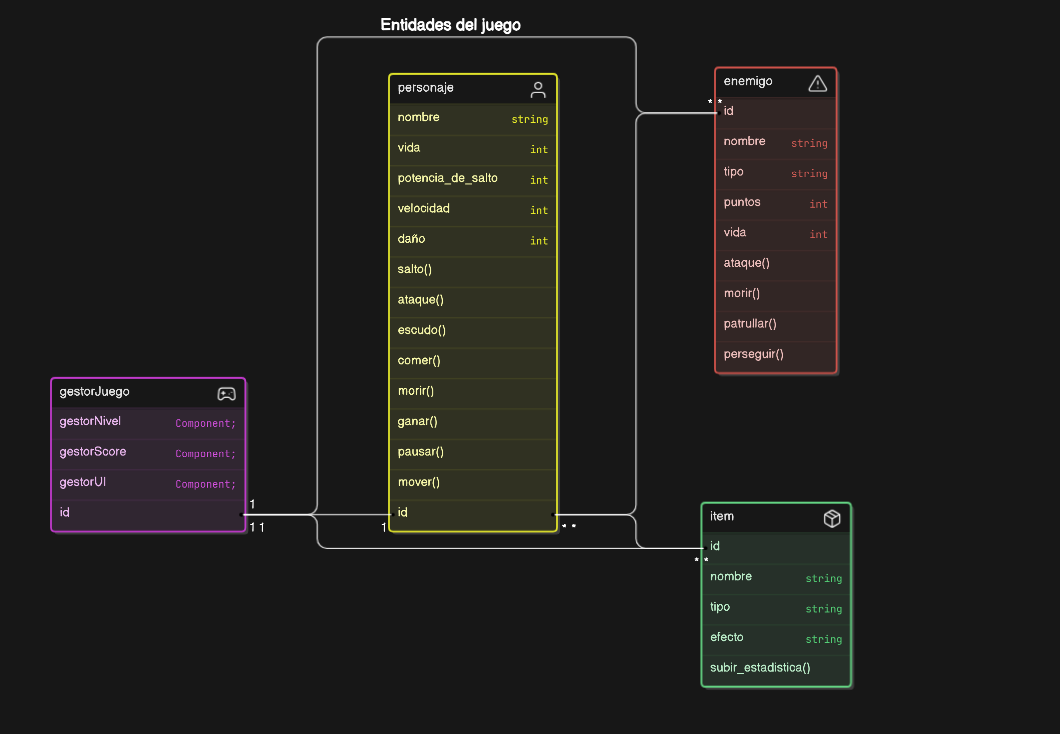
[[4]](#footnote-5)Diagrama de casos de Uso.

El jugador inicialmente puede manejar el sonido, salir y elegir nivel. Elegir nivel abrirá la escena de un nivel para poder jugar, dentro de jugar incluye acciones como atacar, moverse y defenderse. Coger items se extiende de moverse porque no se podría coger items si no te mueves. También puedes pausar el juego y volver al menú.

**Casos de Uso:**

* **Manejar sonido:** el usuario puede subir y bajar el sonido mediante una barra deslizante.
* **Salir:** el usuario puede salir de la aplicación pulsando un botón.
* **Elegir nivel:** el usuario puede elegir un nivel del juego mediante el menú.
* **Jugar:** el usuario puede jugar, lo que engloba acciones como mover, atacar, defender...
* **Mover:** el usuario puede desplazarse en hacía todas las direcciones incluyendo saltar.
* **Atacar:** el usuario puede invocar un ataque con un botón o el ratón.
* **Defender:** el usuario puede invocar un escudo que lo defiende de los ataques mediante un botón.
* **Coger items:** el usuario puede coger items dentro del juego al desplazarse hacia ellos.
* **Pausar:** el flujo del juego se detiene y se abre un menú con opciones.
* **Volver al menú:** se vuelve al menú principal mediante un botón.

#### 5.2.2 Diagrama de Clases

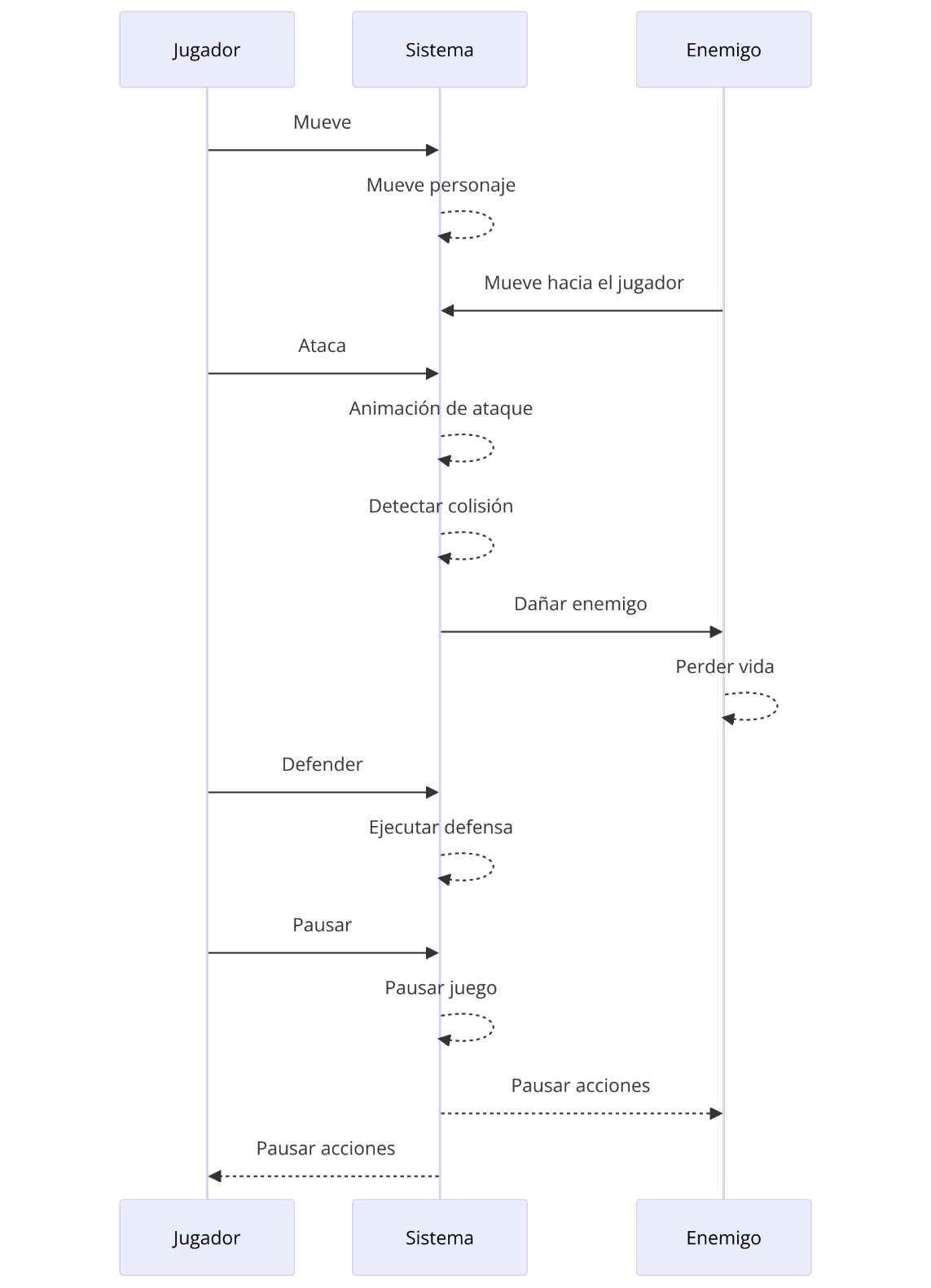
[[5]](#footnote-6).Diagrama de clases

El gestor de juego gestiona los items y los enemigos, sus componentes gestorNivel,gestorScore y gestorUI responden a las acciones del jugador, el jugador ataca, el gestorJuego utiliza gestorNivel y mata al enemigo que ha golpeado, el jugador defiende, el gestorJuego utiliza gestorNivel para aplicar la defensa y gestorUI para mostrar el cooldown del escudo. El enemigo también responde a los movimientos del usuario, cuando muere llama al gestorJuego para que elimine su entidad y añada puntos al score, además de ejecutar sus propias acciones como atacar. Los items reaccionan a la función comer() del usuario, llamando al gestorJuego para eliminar el item y dar el beneficio al personaje.

#### 5.2.3 Diagrama de Secuencia

**Secuencia de Moverse y Atacar:**

* + El jugador pulsa la tecla de movimiento.
  + El sistema mueve al personaje en la dirección especificada.
  + El jugador pulsa la tecla de ataque.
  + El sistema ejecuta la animación de ataque y detecta colisión con enemigos.
  + Si hay colisión, el enemigo recibe daño.
  + Si el enemigo recibe daño, pierde vida.
  + El jugador pulsa la tecla de defensa.
  + Si hay colisión, el jugador no recibe daño.
  + El jugador pausa el juego, el sistema y el enemigo se pausan con él.

[[6]](#footnote-7).Diagrama secuencial

#### 5.2.4 Diagrama de Estados

### [[7]](#footnote-8)Diagrama de estados

El juego inicia y estamos en la animación estática, dependiendo de las condiciones que cumpla el usuario, se ejecutaran una serie de acciones.

-Click: El personaje activará la animación de ataque.

-E: El personaje activará la animación de escudo.

-Die: Cuando se ataca al personaje, se reproducirá la animación de ser golpeado y si se queda sin vida, reproduce la animación de muerte.

-SPACE: Al presionar SPACE, el personaje reproducirá la animación de salto.

-Velocity: Al pasar cierta velocidad, el personaje reproducirá la animación de salto

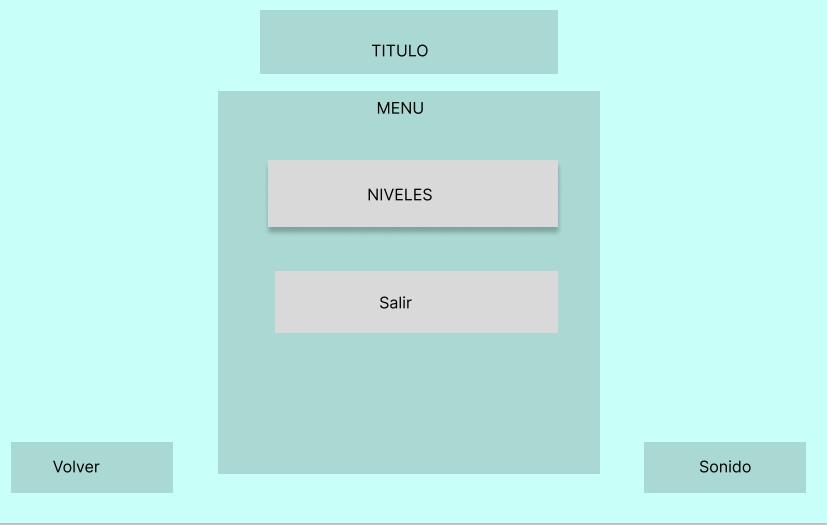
### 5.3 Análisis y Diseño de la Interfaz de Usuario

La interfaz de usuario (UI) debe ser intuitiva y fácil de usar, proporcionando al jugador toda la información necesaria sin sobrecargar la pantalla. Los elementos clave de la UI incluyen:

* **Barra de Vida:** Muestra la vida restante del personaje.
* **Puntuación:** Muestra los puntos acumulados.
* **Indicador de Ítems:** Muestra los ítems recogidos y sus efectos.

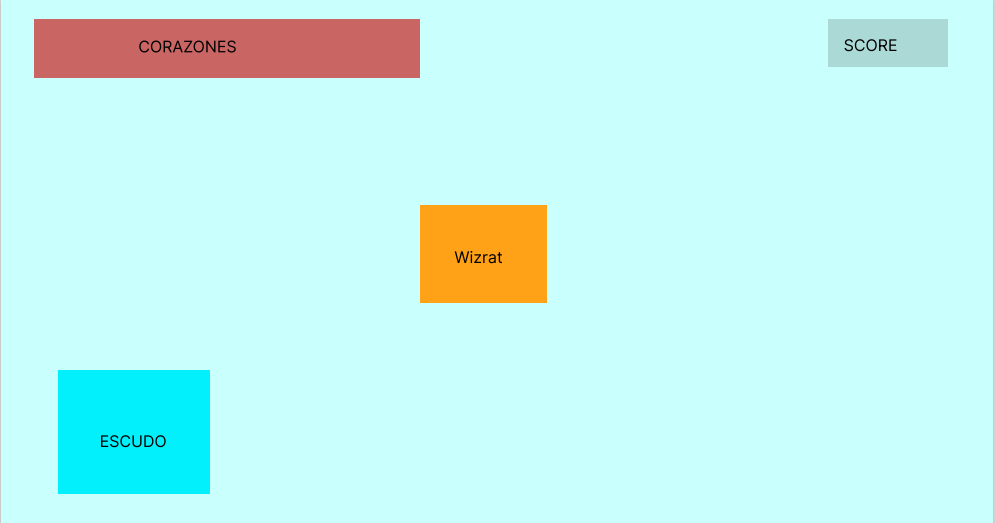
Pantalla Principal:

[[8]](#footnote-9)Pantalla del juego

[[9]](#footnote-10)Mockup Menú

El menú principal se compone de 4 botones:

* **Niveles:** Dándole al botón niveles te dejarán elegir entre varios niveles.
* **Salir:** Dándole al botón salir la aplicación se cerrará.
* **Volver:** Cuando le das al botón Niveles, este botón aparecerá por si quieres volver al menú inicial.
* **Sonido:** Deslizando esta slider se sube y baja el volumen del juego.

[[10]](#footnote-11)Mockup pantalla principal

La pantalla principal se compone 4 elementos:

* **Corazones:** representa la vida del jugador.
* **Score:** representa numéricamente la puntuación del jugador.
* **Escudo:** Representa gráficamente el cooldown del escudo.
* **Wizrat:** Es el personaje jugable.

### 5.4 Diseño de la Arquitectura de la Aplicación

#### 5.4.1 Tecnologías/Herramientas Usadas

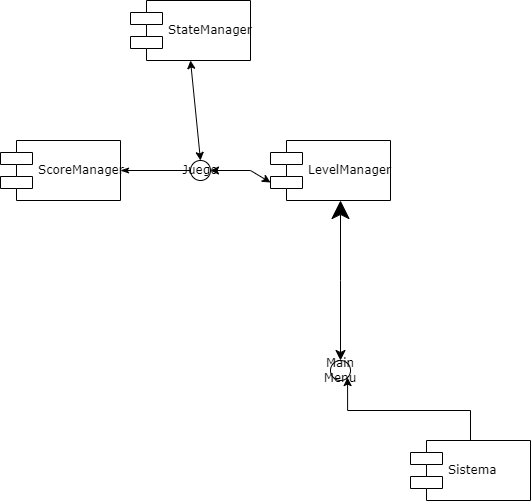
* **Unity:** Motor utilizado para el desarrollo del juego (Versión 2021.3).
* **C#:** Lenguaje de programación para los scripts del videojuego.
* **Libresprite:** Herramientas para la creación y edición de gráficos.
* **Draw.io:** Herramienta de creación de diagramas UML.
* **Designer AI**: Herramienta de generación de imágenes.
* **Suno:** Herramienta de generación de audio por IA.

#### **5.4.2 Arquitectura de Componentes de la Aplicación**

La arquitectura del juego seguirá un modelo de componentes para garantizar la modularidad y facilidad de mantenimiento. Los principales componentes son:

* **Gestor de Juego:** Controla el flujo del juego y maneja la lógica principal.
* **Controlador de Personaje:** Gestiona el movimiento y acciones del personaje principal.
* **Controlador de Enemigos:** Gestiona el comportamiento de los enemigos.
* **Controlador de Ítems:** Maneja la lógica de los ítems.
* **Sistema de UI:** Controla la interfaz de usuario y actualiza los elementos visuales.

#### 5.4.3 Diagrama de Componentes:



[[11]](#footnote-12)Diagrama de componentes.

El sistema se sirve de la interfaz Main Menu para que el usuario ejecute sus acciones, esto lo llevara a otro componente, LevelManager, que como su nombre indica, maneja los niveles. LevelManager interactuará con StateManager por medio del Juego, que permite cambiar de escena, el ScoreManager se encarga de pasar el score como dato para que quede constancia de él.

## 6. Documento de Implementación e Implantación del Sistema

### 6.1 Implementación

La implementación del videojuego " Wizrat " se realizará en Unity, utilizando el lenguaje de programación C#. A continuación, se describe el proceso de implementación de las principales funcionalidades del juego.

#### 6.1.2 Configuración del Proyecto en Unity

**Creación del Proyecto:**

* + Abrir Unity y crear un nuevo proyecto en 2D.
  + Configurar las propiedades básicas del proyecto (nombre, ubicación, etc.).

**Importación de Recursos:**

* + Importar todos los sprites, fondos y demás recursos gráficos necesarios.
  + Organizar los recursos en carpetas para mantener el proyecto ordenado.

#### 6.1.3 Explicación del funcionamiento de los scripts.

En Unity, MonoBehaviour es una clase base fundamental para la creación de scripts que definen el comportamiento de los objetos en tu juego. Cada objeto en tu escena de Unity puede tener uno o más componentes adjuntos a él, y estos componentes son scripts que heredan de MonoBehaviour.

Aquí hay algunos puntos clave sobre MonoBehaviour y su uso en Unity:

**Clase Base**: MonoBehaviour es una clase base en Unity que te permite crear scripts para definir el comportamiento de los objetos en tu juego.

**Lifecycle Methods**: MonoBehaviour proporciona varios métodos que Unity llama automáticamente en diferentes etapas del ciclo de vida del objeto al que está adjunto el script. Algunos de estos métodos incluyen:

* + Start(): Se llama una vez cuando el objeto se inicializa.
  + Update(): Se llama en cada frame de la ejecución del juego.
  + FixedUpdate(): Se llama en intervalos de tiempo fijos, útil para físicas.
  + OnCollisionEnter(), OnTriggerEnter(): Se llaman cuando el objeto colisiona con otro objeto o entra en un trigger, respectivamente.
  + Entre otros.

**Acceso a Componentes**: Los scripts MonoBehaviour pueden acceder y modificar los componentes adjuntos al mismo objeto, así como a otros objetos en la escena. Esto permite la interacción entre diferentes elementos del juego.

**Comunicación entre Objetos**: Los scripts MonoBehaviour también permiten la comunicación entre objetos. Por ejemplo, un script adjunto a un objeto puede llamar métodos en otro objeto para activar ciertos comportamientos o interacciones.

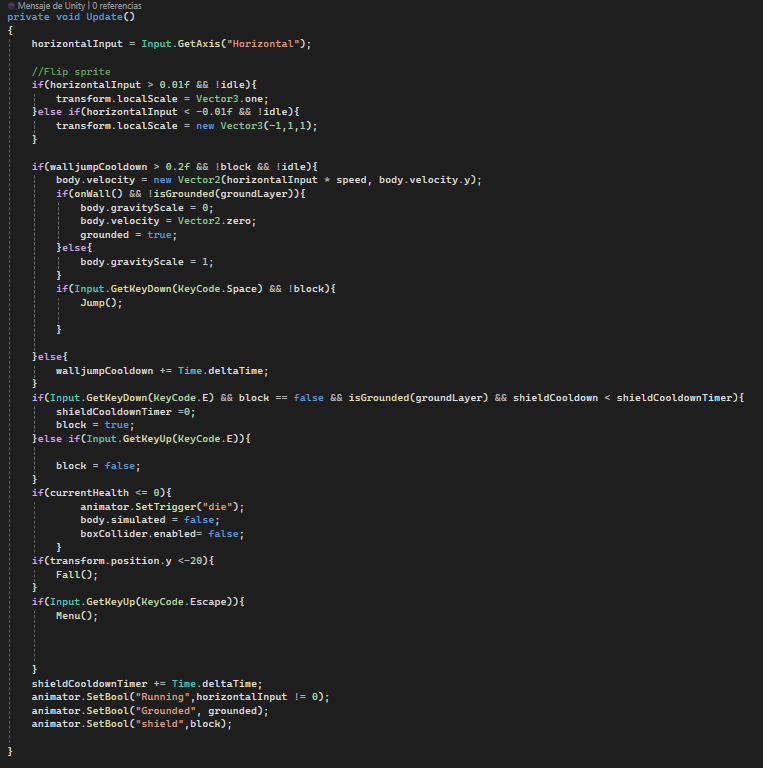
**Personalización del Editor**: Los scripts MonoBehaviour se pueden personalizar en el Editor de Unity utilizando atributos como [SerializeField], que permite que las variables privadas sean visibles en el Inspector de Unity, o [HideInInspector], que oculta variables en el Inspector.

**Creación de Comportamientos Personalizados**: Con MonoBehaviour, puedes crear comportamientos personalizados para tus objetos en el juego. Esto incluye movimientos, interacciones, efectos visuales, lógica de juego, entre otros.

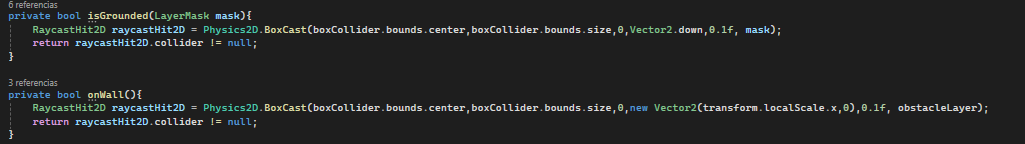
#### 6.1.4 Implementación del Personaje Principal

**Movimiento del Personaje:**

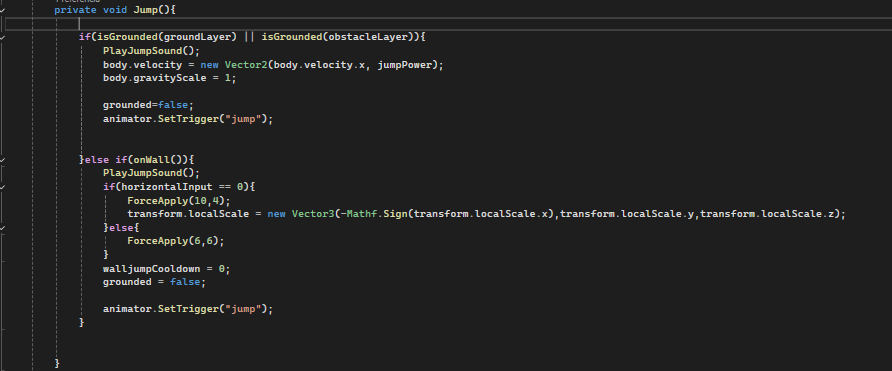
* + He implementado el movimiento utilizando el ribidbody2d como base, que es más o menos el núcleo de simulación de físicas del personaje.
  + Las teclas de movimiento vienen a ser el típico WASD para las direcciones, el espacio para saltar, clic izquierdo para atacar y E para invocar un escudo.
  + El movimiento también dispone de algunas funcionalidades como quedarse anclado a un obstáculo y saltar paredes, simulando una escalada.
  + La orientación en el eje x del personaje depende del input que el jugador este pasándole al videojuego.

[[12]](#footnote-13) Función Update()

En este caso he puesto todo el script de movimiento dentro de la función Update(), aunque también se podría usar la función FixedUpdate(), dentro de Update() también se llama a otras funciones como onWall(), que detecta si esta colisionando con un bloque desde los lados, o isGrounded(), que detecta si esta colisionando por encima con un bloque.

[[13]](#footnote-14) Funciones de comprobación posicional

También se llama a la función Jump(), que controla el salto del jugador, siendo posible el salto de paredes cuando te encuentras con un obstáculo, pero no con paredes marcadas como suelo. La función Jump() llama a su vez al animador, así como Update(), para animar el salto.



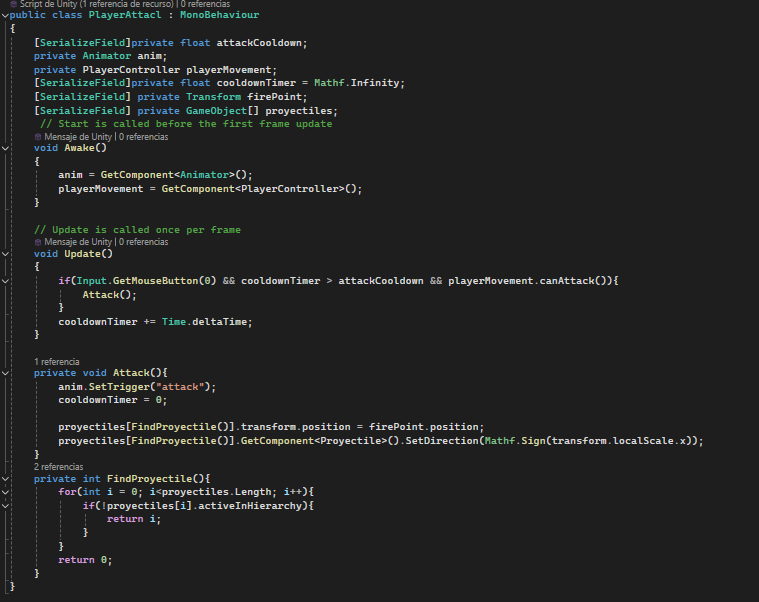
[[14]](#footnote-15)Función de salto



[[15]](#footnote-16)Función auxiliar salto

También llama a ForceApply, que es una función genérica para aplicar una fuerza sobre el jugador, usada tanto para saltar como para ser empujado por enemigos.

#### 6.1.5 Ataque con la Espada:

[[16]](#footnote-17)Script de ataque

El script que se encarga del ataque del personaje utiliza el clic izquierdo como input, llama a la función Attack(), esta llama al animador para ejecutar la animación, reinicia el cooldown, llama a un objeto de tipo proyectiles para coger su posición y cambiarla a la del punto de fuego que está pegado al personaje. Proyectiles es un pool de objetos para una fácil implementación posterior en caso de querer mejorar el juego. FindProyectile() es llamado para buscar el proyectil en cuestión y lanzarlo.

### 6.2 Pruebas

### Pruebas Unitarias

#### **Caso de Prueba 1: Movimiento del Personaje**

* **Título o identificador:** UT-Movimiento-Personaje
* **Descripción:** Prueba el movimiento del personaje principal hacia la izquierda y la derecha.
* **Características:** Método o funcionalidad donde se realiza la prueba: personaje.mover()
* **Datos de entrada:**
* Dirección: izquierda, derecha
* **Valores de prueba:**
* izquierda: -1
* derecha: 1
* **Datos de salida:**
* Posición del personaje en el eje X
* **Valores esperados:**
* izquierda: Disminución de la posición en X en 1 unidad
* derecha: Aumento de la posición en X en 1 unidad
* **Resultado:** Describir el resultado obtenido: La posición del personaje en X se actualiza correctamente según la dirección del movimiento.

#### **Caso de Prueba 2: Saltar del Personaje**

* **Título o identificador:** UT-Saltar-Personaje
* **Descripción:** Prueba el salto del personaje principal.
* **Características:** Método o funcionalidad donde se realiza la prueba: personaje.saltar()
* **Datos de entrada:**
* Fuerza de salto: 5
* **Valores de prueba:**
* Fuerza de salto: 5
* **Datos de salida:**
* Posición del personaje en el eje Y
* **Valores esperados:**
* Aumento de la posición en Y en 5 unidades
* **Resultado:** La posición del personaje en Y se incrementa correctamente al saltar.

#### **Caso de Prueba 3: Ataque del Personaje**

* **Título o identificador:** UT-Ataque-Personaje
* **Descripción:** Prueba el ataque del personaje principal a un enemigo.
* **Características:** Método o funcionalidad donde se realiza la prueba: personaje.atacar()
* **Datos de entrada:**
* Enemigo: enemigo
* **Valores de prueba:**
* Enemigo con salud de 20 unidades
* **Datos de salida:**
* Salud del enemigo
* **Valores esperados:**
* Reducción en la salud del enemigo según el poder de ataque del personaje (por ejemplo, 10 unidades de daño)
* **Resultado:** La salud del enemigo se reduce correctamente al ser atacado por el personaje.

#### **Caso de Prueba 4: Uso del Escudo**

* **Título o identificador:** UT-Uso-Escudo
* **Descripción:** Prueba el uso del escudo por el personaje principal.
* **Características:** personaje.usar\_escudo()
* **Datos de entrada:**
* Ataque recibido de un enemigo: 1 unidades de daño
* **Valores de prueba:**
* Activación del escudo
* **Datos de salida:**
* Salud del personaje
* **Valores esperados:**
* Anulación del daño recibido
* **Resultado:** La salud del personaje no se reduce

### **Pruebas Integrales**

#### **Caso de Prueba 5: Recolección de Ítem**

* **Título o identificador:** IT-Recolección-Ítem
* **Descripción:** Prueba la recolección de un ítem por el personaje principal.
* **Características:** Método o funcionalidad donde se realiza la prueba: personaje.recolectar\_item()
* **Datos de entrada:**
* Ítem: moneda
* **Valores de prueba:**
* Ítem: moneda
* **Datos de salida:**
* Puntuación aumentada
* **Valores esperados:**
* Incremento en el número de puntuación
* **Resultado:** La puntuación se actualiza al coger el item.

#### **Caso de Prueba 6: Colisión con Enemigo**

* **Título o identificador:** IT-Colisión-Enemigo
* **Descripción:** Prueba la colisión del personaje con un enemigo.
* **Características:** personaje.colisionar()
* **Datos de entrada:**
* Enemigo: enemigo
* **Valores de prueba:**
* Enemigo: enemigo
* **Datos de salida:**
* Estado de salud del personaje
* **Valores esperados:**
* Reducción en los puntos de salud del personaje
* **Resultado:** Los puntos de salud del personaje se reducen correctamente al colisionar con un enemigo.

##### **Debugging y Optimización**

**Debugging:**

* + Utilizar las herramientas de depuración de Unity para identificar y corregir errores en el código.

**Optimización:**

* + Asegurar que el juego se ejecuta suavemente, manteniendo una tasa de FPS (frames por segundo) constante y reduciendo tiempos de carga.

## 7. Documento de Cierre

### 7.1 Instalación y Configuración

**Requisitos del Sistema:**

* **Sistema Operativo:** Windows 10 o superior, macOS 10.14 o superior
* **Procesador:** Intel Core i5 o equivalente
* **Memoria RAM:** 8 GB
* **Almacenamiento:** 500 MB de espacio disponible
* **Tarjeta Gráfica:** Compatible con DirectX 11 (Windows) o Metal (macOS)

**Instrucciones de Instalación:**

* 1. **Descarga del Juego o juégalo online:**

Accede a la página de descarga del juego y descarga el archivo comprimido (Wizrat.zip) o juega el navegador [Wizrat by Reemolke (itch.io)](https://reemolke.itch.io/wizrat).

* 1. **Extracción de Archivos:**

Extrae el contenido del archivo comprimido en la ubicación deseada de tu disco duro.

* 1. **Ejecución del Juego:**
  + Navega a la carpeta donde extrajiste los archivos y ejecuta el archivo Wizrat.exe (Windows).

**Configuración del Juego:**

* 1. **Ajustes de Gráficos:**

Accede al menú de opciones del juego.

Ajusta la resolución y calidad gráfica según las capacidades de tu hardware.

* 1. **Guardar y Cargar Partidas:**

El juego guarda automáticamente tu progreso al completar cada nivel.

Puedes cargar tu partida desde el menú principal del juego.

### 7.2 Manual de Usuario

**Introducción:**

"Wizrat" es un juego de aventura en 2D donde controlas a un ratón mágico que debe derrotar enemigos mágicos para prevenir la corrupción del mundo. A lo largo del juego, recogerás ítems que mejorarán tus habilidades y te ayudarán en tu misión.

**Controles del Juego:**

* **Movimiento:** Usa las flechas de dirección o las teclas WASD para mover al ratón.



* **Atacar:** Haz clic izquierdo para realizar un tajo vertical con la espada, solo se puede ejecutar pisando el suelo.



* **Usar Escudo:** Mantén presionada la tecla E para activar el escudo y bloquear ataques, solo se puede ejecutar pisando el suelo. Tarda 10 segundos en poder volver a activarse.



* **Saltar:** Presiona SPACE para saltar.



**Elementos del Juego:**

* **Enemigos:** Derrota a los enemigos para ganar puntos y avanzar en el juego.
  + Rana: salta para atacarte.



* + Cerdo: te persigue para atacarte, si se acerca lo suficiente te dañará.



* + Fuego: vuela hacia ti para atacarte, si te toca el también muere.



* **Ítems:**
  + **Potencia de Salto:** Aumenta la capacidad de salto del ratón.



* + **Velocidad:** Incrementa la velocidad de movimiento.



* + **Corazón:** Restaura la salud del ratón.



* **Niveles:** Completa cada nivel derrotando enemigos y recogiendo ítems para progresar en el juego.
* **Objetos:**
  + **Pinchos:** salen y se esconden haciendo daño al pisarlos.



* + **Bandera:** al tocarla completas el nivel.



**Consejos para Jugar:**

* Usa el escudo estratégicamente para bloquear ataques mientras te acercas a los enemigos.
* Recoge tantos ítems como puedas para mejorar tus habilidades y facilitar tu avance.
* Explora cada nivel a fondo para encontrar ítems ocultos y derrotar a todos los enemigos.

### 7.3 Resultados Obtenidos y Conclusiones

**Resultados Obtenidos:**

* **Desarrollo Completo del Juego:**
  + Se completó el desarrollo del juego "Wizrat" según las historias de usuario.
  + El juego incluye todos los niveles, enemigos, ítems y mecánicas de combate planificadas.
* **Pruebas Exitosas:**
  + Se realizaron pruebas unitarias e integrales, asegurando que todas las partes del juego funcionan correctamente.
  + Las pruebas de usuarios proporcionaron feedback positivo y se realizaron ajustes basados en sus sugerencias.

**Conclusiones:**

El desarrollo de "Wizrat" ha sido una experiencia enriquecedora que ha permitido aplicar conocimientos en desarrollo de videojuegos, diseño gráfico y programación. El juego ofrece una experiencia única y desafiante, con una narrativa atractiva y mecánicas de juego innovadoras. La metodología ágil utilizada facilitó la gestión del proyecto y permitió realizar ajustes según las necesidades del desarrollo. En resumen, el proyecto ha cumplido con sus objetivos y ha proporcionado una plataforma valiosa para el aprendizaje y la creatividad.

### 7.4 Diario de Bitácora

|  |  |  |
| --- | --- | --- |
| Fecha | Tarea | Descripción |
| 06-04 | Pensar el concepto del juego e historia. | No tardé mucho en pensar el concepto, se me vino a la mente la rata mágica y eso hice. |
| 12-04 | Hacer tutoriales de creación de videojuegos. | Antes de meterme con el juego hice el tutorial de Unity para hacer videojuegos. |
| 14-04 | Hacer los sprites iniciales del personaje principal | Hice sprites de idle y de movimiento para testear la jugabilidad. |
| 20-04 | Empezar el proyecto de Unity | No hice más que meter las animaciones y hacer el script básico de movimiento |
| 24-04 | Poner el proyecto en GitHub | Joaquín me dijo de poner el proyecto en GitHub, así que hice eso. |
| 25-04 | Desarrollar el movimiento e introducir proyectiles. | Acabé de implementar el movimiento e introduje la espada mágica como proyectil. |
| 26-04 | Implementación de items | Introduje los items que sumarán vida o capacidades de salto y velocidad. |
| 27-04 | Corrección items y salto | El salto daba problemas en algunas partes y los items decidí cambiarlos a unos quesos. |
| 28-04 | Corrección de la espada mágica y el walljump | Las correcciones del salto normal limitaban el walljump así que tuve que arreglarlo un poco, la espada se chocaba con cosas que no debía así que corregí eso también. |
| 29-04 | Introducción de enemigos | Los primeros enemigos con su movimiento y sus funciones primarias. |
| 30-04 | Empuje del jugador y enemigo | Los enemigos ahora empujan al jugador y viceversa. |
| 01-05 | Mejora de la IA del enemigo | Introduje la función onRadius() para que el enemigo solo persiguiera cuando el jugador estaba cerca. También animé estas interacciones. |
| 02-05 | Introducción de defensa del jugador. | Cree un escudo que bloquea un daño e hice su respectiva animación. |
| 03-05 | Muerte y cooldown del escudo. | Introduje un sistema de muerte y le puse cooldown al escudo. |
| 04-05 | Introducción de la puntuación y mejora de los coleccionables. | Cree el sistema de puntuación por enemigos y quesos. |
| 05-05 | Rediseño de sprite | Rehíce todas las animaciones y el personaje. |
| 06-05 | Rediseño de la espada | Rehíce todas las animaciones y la espada. |
| 09-05 | Introducción enemiga II | Hice un enemigo nuevo volador |
| 10-05 | Tilemap | Cree el tilemap del suelo desde cero. |
| 11-05 | Elementos dañinos | Cree unos pinchos que salen y se esconden para añadir dificultad. |
| 12-05 | Mejora menú y enemigo II | La UI del juego la modifiqué porque la introducida era muy rudimentaria. |
| 13-05 | Bandera de victoria | Introduje una bandera de victoria que te pasa al siguiente nivel al menú dependiendo de lo que elijas. |
| 14-05 | Documentación | Avancé mucho en la documentación |
| 18-05 | Enemigo III | Introduje una rana saltarina que hace 2 de daño y solo te persigue saltando. |
| 19-05 | Introducción de barra de vida y el panel de muerte | Cree unos corazones en Libresprite y los utilice como una barra de vida, después añadí un panel de muerte con opción para reintentar o volver al menú principal. |
| 24-05 | Introducción de SFX | Cree unos efectos de sonidos y los añadi a cada acción del personaje/enemigo. |
| 25-05 | Control audio | Cree una barrita de volumen para modificarlo. |
| 26-05 | Introducción OST | Cree una canción instrumental en Suno y la introduje a los niveles. |
| 27-05 | Arreglo GitHub | Al hacer un push me dio errores el merge, intenté arreglarlos, pero tardé un rato largo. |
| 28-05 | Documentación | Hice la mayoría de los diagramas UML. |
| 05-06 | Mejora UI | Cree botones más visualmente atractivos |
| 10-06 | Mejora corazones | Cambié los colores de los corazones para mejor visibilidad. |
| 11-06 | Nivel tutorial | Cree un nivel tutorial |
| 12-06 | Icono del escudo | Cree un icono del escudo para representar su cooldown |

### **8. Bibliografía**

* **Sitios Web:**
  + **Documentación de Unity:** <https://docs.unity3d.com/>
  + **Stack Overflow:** <https://stackoverflow.com/>
  + **Unity Learn:** <https://learn.unity.com/>
* **Recursos Gráficos:**
  + **OpenGameArt:** <https://opengameart.org/>
  + **Itch.io:** [itch.io](https://itch.io/)
* **Explicaciones auxiliares**
  + **Tutoriales de Unity2d:** [YouTube](https://www.youtube.com/)
* **Explicaciones auxiliares**
  + **Suno:** [Suno](https://suno.com/)

### **9. Anexos**

* **Contrato de desarrollo.**

# Contrato de desarrollo de software

En Calatayud, a 14 de junio 2024

**REUNIDOS**

**De una parte,**

Riquelmer Añez Joffre, mayor de edad, con domicilio en Calle Sixto Celorrio N3 Piso 3A, DNI/NIF núm. 17465687Q, y en su propio nombre y representación.

En adelante, el "**Desarrollador**".

**De otra parte,**

\_\_\_\_\_\_\_\_, mayor de edad, con domicilio en \_\_\_\_\_\_\_\_, DNI/NIF núm. \_\_\_\_\_\_\_\_, y en su propio nombre y representación.

En adelante, el "**Cliente**".

El Desarrollador y el Cliente que, en adelante, podrán denominarse, individualmente, la "Parte" y las "Partes", reconociéndose la capacidad jurídica necesaria para contratar y obligarse, y en especial, para otorgar este CONTRATO DE DESARROLLO DE SOFTWARE. En adelante, el "**Contrato**".

**EXPONEN**

**I.** Que el Desarrollador se dedica a la siguiente actividad:

Desarrollo de aplicaciones para PC

**II.** Que, en virtud de lo anterior, el Desarrollador dispone de los conocimientos y medios necesarios para el diseño y codificación de programas informáticos o aplicaciones;

**III.** Que el Cliente desea el desarrollo de un programa informático denominado Wizrat (en adelante, el "**Software**"), con una serie de características y funcionalidades determinadas que han sido comunicadas al Desarrollador;

**IV.** Que, con el fin de desarrollar el Software de acuerdo con las especificidades y funcionalidades solicitadas por el Cliente, las Partes han negociado y aceptado un presupuesto y un plan de trabajo para el desarrollo del Software;

**V.** Que, en virtud de lo anterior, el Desarrollador desea, libre y espontáneamente, comprometerse a diseñar, estructurar y codificar el Software atendiendo a las instrucciones del Cliente, operación que las Partes desean formalizar a través del presente Contrato, que se regirá por las siguientes,

**ESTIPULACIONES**

### PRIMERA. Objeto del Contrato

Mediante el presente Contrato, el Desarrollador se compromete a diseñar, estructurar y codificar el Software en favor del Cliente dentro del plazo fijado en el presente Contrato.

De esta forma, el Desarrollador creará un Software personalizado según el presupuesto o los documentos técnicos negociados entre las Partes de forma previa a la firma del presente Contrato, cuya titularidad corresponderá en su integridad al Cliente.

Por último, el diseño, la estructuración y codificación del Software se ceñirá a lo establecido en las estipulaciones de este Contrato y a lo dispuesto en el Real Decreto Legislativo 1/1996, de 12 de abril, por el que se aprueba el texto refundido de la Ley de Propiedad Intelectual, así como a la restante legislación aplicable.

### SEGUNDA. Funcionalidades del Software

Siguiendo las peticiones del Cliente, y de acuerdo con la negociación previa llevada a cabo entre las Partes, el Desarrollador queda obligado a diseñar y codificar el Software cumpliendo con las siguientes funcionalidades o características:

Un videojuego 2D utilizando el motor de Unity, de género plataforma con elementos de acción y aventura

El Desarrollador diseñará y codificará el Software siguiendo lo establecido en esta Estipulación, así como lo recogido en el propio Contrato y en todos sus Anexos.

### TERCERA. Precio y forma de pago

Las Partes acuerdan el pago de una cantidad ascendiente a **3239 euros** como remuneración del Software, sin incluir los impuestos que se pudieran derivar de esta operación.

El Cliente satisfará el pago del precio fijo el 20 de junio 2024 mediante transferencia bancaria a la siguiente cuenta bancaria facilitada por el Desarrollador:

Entidad bancaria: Caixa Bank

IBAN: ES48328428348294294293

Por último, el Desarrollador emitirá una factura al Cliente cumpliendo con los requisitos legales necesarios y dentro de los plazos previstos en la legislación actual.

### CUARTA. Intereses de demora

Siguiendo lo recogido en el artículo 1.101 del Código Civil, cualquier retraso en el pago de la remuneración establecida en el presente Contrato dará lugar a un incremento del precio equivalente a los intereses de demora generados por el retraso en el pago.

El tipo de interés de demora será igual al tipo de interés de referencia o de refinanciación semestral del Banco Central Europeo en vigor a 1 de enero para el primer semestre del año correspondiente, y a 1 de julio para el segundo semestre del año correspondiente.

Los intereses de demora se exigirán automáticamente desde la fecha de pago fijada en la Estipulación anterior, sin necesidad de aviso del vencimiento ni intimación alguna por parte del Desarrollador. El devengo de dichos intereses no afectará al ejercicio de cualquier acción que pueda corresponderle al Desarrollador derivada del incumplimiento del pago.

### QUINTA. Originalidad del Software

El Desarrollador manifiesta de forma expresa que el Software será íntegramente diseñado y codificado por el Desarrollador, por su personal cualificado, o por colaboradores en línea con lo dispuesto en este Contrato, respetando, en su integridad, la legislación sobre la propiedad intelectual aplicable.

En virtud de lo anterior, el Desarrollador manifiesta que no utilizará ningún código u cualquier otro tipo de propiedad intelectual de terceros distintos a los citados en el párrafo anterior, ya sea un código fuente o un código de carácter complementario, para el desarrollo del Software. El Software será totalmente codificado de forma original por el Desarrollador y sus colaboradores, en su caso.

Siguiendo estas manifestaciones, el Desarrollador se compromete a:

**a.** Exonerar al Cliente de toda responsabilidad frente a terceros que aleguen una violación de sus derechos de propiedad intelectual que puedan tener sobre el Software.

**b.** Garantizar el uso del Software, debiendo indemnizar al Cliente en el caso de que por resolución judicial se impida su uso por causas o motivos generados con anterioridad a la entrega del Software en favor del Cliente.

Además, las Partes acuerdan que, por razones de equidad, y en la medida que lo justifiquen las circunstancias, el Cliente podrá reclamar la restitución de las sumas pagadas o gastos derivados del Contrato hasta la fecha en la que tiene lugar el pronunciamiento judicial en favor de un tercero o en la que se produce la cancelación o denegación de la inscripción del Software en el registro de la propiedad intelectual correspondiente.

**c.** Mantener informado al Cliente de todos los posibles usos fraudulentos o violaciones del Software que hayan podido realizar terceros durante el desarrollo del Software, comprometiéndose a adoptar todas las medidas necesarias para garantizar su protección y permitir el correcto uso del Software.

**d.** Garantizar que, en caso de ser necesario, el Cliente cuenta con el consentimiento de todos desarrolladores, incluidos sus posibles colaboradores, para la cesión de su posible propiedad sobre el Software en favor del Cliente, recayendo bajo su responsabilidad los perjuicios que se puedan derivar de la ausencia de este.

### SEXTA. Plazo de ejecución

El Contrato entrará en vigor en la fecha señalada en el encabezado del presente Contrato. El Desarrollador deberá llevar a cabo el diseño y codificación del Software siguiendo los plazos establecidos a continuación:

- 14 de junio 2024, se deberá completar lo siguiente:

Se deberá de tener una demo funcional

El Cliente se compromete a colaborar y aportar la información necesaria requerida por el Desarrollador para poder codificar el Software según los plazos establecidos en esta Estipulación. Si el Cliente no facilita esta información o no colabora adecuadamente, el Desarrollador podrá comunicar por escrito al Cliente su imposibilidad de cumplir con los plazos establecidos en esta Estipulación.

### SÉPTIMA. Modificación del Software

El Cliente podrá únicamente solicitar la introducción de cambios, modificaciones o mejoras en el Software si así accede el Desarrollador.

En el caso de que el Desarrollador acepte la introducción de estos cambios, se deberá negociar y recoger por escrito los nuevos plazos de entrega y precio según las modificaciones solicitadas.

### OCTAVA. Propiedad del Software

El Desarrollador reconoce de forma expresa que el Cliente será el titular de pleno dominio de la propiedad intelectual del Software. De esta forma, el Desarrollador renuncia, expresa y totalmente, a cuantos derechos de propiedad intelectual pudieran generarse como consecuencia del desarrollo del Software.

En consecuencia, el Desarrollador se obliga a no hacer uso de estos derechos para fines distintos a los del cumplimiento del Contrato. El resultado de los trabajos realizados, en su totalidad o en cualquiera de sus fases, será propiedad del Cliente y este, en consecuencia, podrá solicitar en cualquier momento la entrega de los documentos o materiales que la integren, con todos sus antecedentes, borradores, datos o procedimientos.

No obstante, la cesión de los derechos de explotación de software o programas informáticos preexistentes de propiedad del Desarrollador que el Cliente estuviera interesado en incorporar como parte del futuro Software serían negociados expresamente, y caso por caso, entre el Desarrollador y el Cliente. El Desarrollador informaría previamente al Cliente de forma clara y concisa sobre las condiciones de adquisición y/o explotación, para que éste pueda decidir libremente sobre los mismos.

El Cliente podrá libremente decidir llevar a cabo la inscripción del Software o no en el Registro de la Propiedad Intelectual correspondiente, así como realizar todas las renovaciones de dicha inscripción que considere necesarias.

La cesión de los derechos de propiedad intelectual en favor del Cliente prevista en el presente Contrato no tiene ámbito geográfico determinado; es decir, se extiende a todos los países del mundo sin limitación geográfica de ninguna clase.

Por último, el Desarrollador no mantendrá derecho de propiedad alguno, ni tendrá ningún derecho de compensación más allá de lo establecido en este Contrato, sobre todas aquellas partes del Software o de sus módulos desarrollados.

### NOVENA. Cesión del Contrato. Subcontratación

Las Partes no podrán ceder su posición en el presente Contrato, ni tampoco los derechos u obligaciones que de este mismo emanasen a su favor o a su cargo, sin el consentimiento previo, expreso y por escrito de la otra Parte.

En particular esta Estipulación regirá de forma que el Desarrollador se compromete a comunicar al Cliente por escrito, y de forma previa a la celebración de un acuerdo de subcontratación, su intención de contratar a una parte subcontratista o colaboradora, la identidad de esta, el tipo de servicios y trabajos a realizar por esta en referencia al Software y las condiciones económicas y legales, de la relación de subcontratación. Todo ello en orden de facilitar que el Cliente pueda aprobar dicha subcontratación, sin que dicha autorización suponga la asunción de responsabilidad alguna por parte del Cliente o la aprobación del resultado de los servicios y trabajos que la parte subcontratista o colaboradora provea.

Igualmente, será responsabilidad del Desarrollador comprobar que la parte subcontratista está autorizada para la prestación de los servicios o actividades objeto de subcontratación, así como regular por escrito la relación contractual con la misma, incorporando o anexando los acuerdos establecidos en el presente Contrato. Además, el Desarrollador se obliga a entregar al Cliente una copia de dicho contrato en los 5 (cinco) días siguientes a su firma.

La parte subcontratista actuará en todo momento bajo la dirección y control del Desarrollador, quien se obliga y responsabiliza de hacer cumplir todas las obligaciones asumidas en el presente Contrato.

El Desarrollador responderá solidariamente de las obligaciones que asuma la parte subcontratista, incluso cuando el Cliente hubiera autorizado dicha subcontratación, incluyendo los daños y perjuicios que pudiese sufrir directa o indirectamente por la actuación de dicha parte subcontratista. Del mismo modo, cualquier acto, error o negligencia en el cumplimiento de las obligaciones laborales o de Seguridad Social de la parte subcontratista, de sus representantes, o trabajadores, no serán, en ningún caso, responsabilidad del Cliente.

El incumplimiento de esta Estipulación por el Desarrollador será motivo suficiente para resolver el presente Contrato.

### DÉCIMA. Inexistencia de relación laboral

Las Partes declaran de forma expresa que, a todos los efectos legales, el Desarrollador desempeña su actividad de forma totalmente independiente, siguiendo su propia organización y con sus medios personales y técnicos.

De esta forma, las Partes reconocen la inexistencia de relación laboral alguna entre ellas según lo establecido en el artículo 1 y siguientes del Real Decreto Legislativo 2/2015, de 23 de octubre, por el que se aprueba el texto refundido de la Ley del Estatuto de los Trabajadores.

En virtud de lo anterior, no será de aplicación la normativa laboral ni cabrá la posibilidad por ninguna de las Partes de establecer reclamación alguna en esta materia.

### DECIMOPRIMERA. Cumplimiento normativo aplicable

El Desarrollador se compromete a desarrollar el Software cumpliendo de forma diligente con toda la normativa aplicable y, en particular, con todas las obligaciones laborales, de la Seguridad Social, fiscales y de protección de datos que le sean aplicables en relación con el desarrollo del Software.

### DECIMOSEGUNDA. Entrega del Software

Una vez cumplido el plazo fijado en la Estipulación Sexta del presente Contrato, se realizará la entrega del Software en favor del Cliente. La entrega del Software se deberá llevar a cabo de la siguiente forma:

Se entregará telemáticamente en un archivo

Por otro lado, el Cliente se compromete a la recepción del Software, pudiendo en todo caso expresar las reservas o incidencias que estime oportunas sobre el mismo en el caso de que no satisfaga sus expectativas.

Si las Partes así lo acuerdan, se podrá aprobar la entrega parcial antes del vencimiento del plazo antes mencionado de partes o módulos del Software que puedan funcionar de forma autónoma.

### DECIMOTERCERA. Garantía del Software

Tras la realización de la entrega del Software según lo dispuesto en la Estipulación anterior, serán de aplicación al mismo las garantías previstas en el Código Civil, así como lo dispuesto en el texto refundido de la Ley General para la Defensa de Consumidores y Usuarios y en el Código de Comercio en su caso.

En concreto, las Partes acuerdan un periodo de garantía con una duración de: 1 mes desde la entrega del Software.

Por último, la garantía legal no se aplicará en el caso de que el Cliente incumpla con su obligación de pago establecida en la Estipulación tercera o decida terminar de forma anticipada este Contrato por decisión voluntaria.

### DECIMOCUARTA. Gastos

El Desarrollador se hará cargo de todos los gastos derivados en relación con el diseño y codificación del Software. Deberá hacerse cargo del pago de los medios e instrumentos necesarios para poder ejecutar el Contrato correctamente, así como de todos los impuestos o tasas que se devenguen en relación con la creación del Software, quedando el Cliente completamente indemne del pago de todos estos gastos.

### DECIMOQUINTA. Elevación a público del Contrato

Cualquiera de las Partes podrá solicitar, mediante requerimiento fehaciente, la elevación a público del presente Contrato.

En ese caso, las Partes elegirán por mutuo acuerdo el Notario o Notaria Público ante el cual se otorgará la escritura pública y la parte solicitante se hará cargo de los correspondientes gastos notariales.

### DECIMOSEXTA. Fuerza mayor

El retraso en el cumplimiento de cualquier obligación de las Partes no será considerado una omisión o un incumplimiento del Contrato en el caso de que tenga su origen en causas imprevisibles o inevitables (en adelante, "**Fuerza Mayor**"), siempre que se haya informado debidamente a la otra Parte sobre esta situación.

Se entenderá como Fuerza Mayor, entre otras: inundación, incendio, explosión, avería en la planta de producción, cierre patronal, huelga, disturbio civil, bloqueo, embargo, mandato, ley, orden, regulación, ordenanza, demanda o petición del gobierno, o cualquier otra causa que se encuentre fuera de control de la Parte involucrada, sin que pueda entenderse que la falta de fondos constituye una causa de Fuerza Mayor.

La Parte afectada por la Fuerza Mayor hará todo lo posible por eliminar su causa. La exigibilidad de la obligación cuyo cumplimiento se haya visto afectado por la situación de Fuerza Mayor se suspenderá hasta diez (10) días después de que la situación de Fuerza Mayor deje de impedir o retrasar el cumplimiento. Si la causa de Fuerza Mayor no desaparece tras treinta (30) días, o un plazo inferior que justifique la resolución por la imposibilidad de ejecutar la obligación, las Partes podrán acordar la modificación del Contrato o su resolución.

### DECIMOSÉPTIMA. Obligación de secreto y confidencialidad

Las Partes reconocen que toda la información a la que se pueda tener acceso en el marco del Contrato ya sea relacionada con el Software objeto de desarrollo o relacionada con la actividad u organización de alguna de las Partes (en adelante, la "**Información**"), tiene carácter confidencial. De esta forma, las Partes acuerdan no divulgarla y mantener la más estricta confidencialidad respecto de dicha Información, advirtiendo, en su caso, de dicho deber de confidencialidad y secreto a sus empleados, asociados y a cualquier persona que, por su cargo o relación personal o sentimental deba o pueda tener acceso a la misma.

Ninguna de las Partes podrá reproducir, modificar, hacer pública o divulgar a terceros la Información sin previa autorización escrita y expresa de la otra Parte.

Las Partes se comprometen a poner los medios necesarios para que la Información no sea divulgada ni cedida. Adoptarán las mismas medidas de seguridad que adoptarían respecto a la información confidencial de su propiedad, evitando su pérdida, robo o sustracción.

El receptor de la Información se compromete, en su caso, a advertir sobre la existencia del deber de confidencialidad a sus empleados, asociados, y a toda persona a la cual se le facilite la Información, haciéndose responsable del uso indebido que estos puedan hacer de la Información relacionada con el Contrato.

Asimismo, la Parte que recibe la Información se compromete a poner en conocimiento de la otra Parte cualquier acción o incidente por parte de terceros que pueda atentar contra la confidencialidad de la Información.

Ambas Partes se comprometen a que la utilización de la Información solo estará dirigida a alcanzar los objetivos del Contrato y no otros, y que, así, solo estará en conocimiento de aquellas personas estrictamente necesarias para cumplir con aquellos.

Las disposiciones relativas a la confidencialidad previstas en este Contrato se aplicarán durante la vigencia de este y, prevalecerán durante el siguiente período: 1 año tras su terminación. Este plazo de tiempo es inmediato a la terminación del Contrato.

### DECIMOCTAVA. Inexistencia de renuncia

La renuncia de una de las Partes a exigir el cumplimiento de alguna de las obligaciones previstas en el Contrato, o a ejercer alguno de los derechos o acciones que le asisten en virtud de este, **(a)** no liberará a la otra Parte del cumplimiento íntegro de las restantes obligaciones contenidas en el Contrato; y, **(b)** no se entenderá como una renuncia a exigir en un futuro el cumplimiento de cualquier obligación o a ejercer derechos o acciones previstos en el Contrato.

La dispensa, aplazamiento o renuncia de alguno de los derechos contemplados en el Contrato, o a una parte de estos, será vinculante si consta por escrito, pudiendo quedar sujeta a las condiciones que el otorgante de dicha dispensa considere oportuno, limitándose al caso concreto en el que se produjo, y no restringirá su exigibilidad en otros supuestos del derecho al que afecta.

### DECIMONOVENA. Terminación anticipada del Contrato

El presente Contrato podrá ser resuelto por el mutuo acuerdo de las Partes, con los efectos que ellas determinen. En todo caso, la terminación del Contrato se deberá formular por escrito.

Por otro lado, el Contrato terminará en el periodo de duración inicial, o de cualquiera de sus prórrogas, si cualquiera de las Partes lo denuncia según lo establecido en la Estipulación de Duración del Contrato.

Igualmente, podrá ser resuelto en cualquier momento por cada una de las Partes, a su elección, sin necesidad de intervención judicial, y sin perjuicio de la responsabilidad en la que incurra la otra Parte por su incumplimiento contractual, siempre que existan "causas justificadas", tal y como se expone a continuación:

**a.** el incumplimiento total o parcial por la otra Parte de alguna de las condiciones u obligaciones esenciales de este Contrato que no sea corregido en el plazo de diez (10) días a partir de la notificación escrita y fehaciente para que así lo haga; y,

**b.** las demás establecidas en el articulado del presente Contrato o las que se recojan en la ley, y en concreto, en el Código Civil y el Código de Comercio.

Ante la terminación anticipada del Contrato por cualquier causa, el Desarrollador cesará en su preparación y el Cliente en el uso y comercialización de los módulos o partes del Software que hayan recibido, debiendo cumplirse, en todo caso, lo dispuesto en la Estipulación "Obligación de secreto y confidencialidad" sobre la obligación de confidencialidad", las "Obligaciones de no competencia" y las restantes obligaciones aplicables.

### VIGÉSIMA. Incumplimiento del Contrato

El incumplimiento por cualquiera de las Partes de las obligaciones recogidas en el presente Contrato facultará a la otra Parte para, o bien exigir su cumplimiento más el correspondiente pago de intereses derivados del retraso en el cumplimiento, o bien resolver el Contrato en el caso de que no se rectifique o subsane el incumplimiento por parte de la Parte incumplidora en el plazo de diez (10) días naturales desde la fecha en la que se verifique el incumplimiento, con la consiguiente indemnización de daños y perjuicios más el pago de intereses por el retraso en el cumplimiento siguiendo lo dispuesto en el artículo 1.124 del Código Civil.

Nadie podrá eximirse del cumplimiento de las obligaciones del presente Contrato mediante el pago de la correspondiente indemnización de daños y perjuicios, pudiendo exigirse el cumplimiento de las obligaciones o prestaciones debidas junto a la satisfacción de la correspondiente indemnización.

### VIGESIMOPRIMERA. Exigibilidad

La falta por cualquier Parte de la exigencia del cumplimiento de cualquiera de las obligaciones recogidas en el presente Contrato no afectará al derecho de dicha Parte a hacer valer la misma. La renuncia por cualquier Parte de una estipulación de este Contrato no podrá interpretarse ni como una renuncia a denunciar cualquier incumplimiento posterior de dicha estipulación, ni como una renuncia de esta.

### VIGESIMOSEGUNDA. Integridad del Contrato

Las Partes reconocen que todos los documentos Anexos y/o adjuntados al presente Contrato forman parte integrante del mismo a todos los efectos, y, por tanto, son totalmente vinculantes para las Partes.

### VIGESIMOTERCERA. Protección de datos

Las Partes de este Contrato conocen y se obligan a cumplir el Reglamento (UE) 2016/679 del Parlamento Europeo y del Consejo, de 27 de abril de 2016, relativo a la protección de las personas físicas en lo que respecta al tratamiento de datos personales y a la libre circulación de estos datos (RGPD), así como la Ley Orgánica 3/2018, de Protección de Datos Personales y garantía de los derechos digitales y su normativa de desarrollo, y/o aquellas que las pudieran sustituir o actualizar en el futuro.

De esta forma, las Partes son conscientes de que mediante la firma de este Contrato consienten que sus datos personales recogidos en el presente Contrato, así como aquellos que se pudiesen recoger en el futuro para poder dar cumplimiento o una correcta ejecución de este mismo, podrían ser incorporados por la otra Parte a su propio fichero automatizado o no de recogida de datos con el fin de ejecutar correctamente la relación contractual y, eventualmente, para una gestión administrativa y/o comercial.

Las Partes se comprometen a que estos datos personales no se comunicarán a terceros, aunque, si se hace algún tipo de comunicación personal, se comprometen siempre y de forma previa, a solicitar el consentimiento expreso, informado e inequívoco de la Parte titular de dichos datos, indicando la finalidad concreta para la que se realizará la comunicación.

Para las Partes no se limita el ejercicio de los derechos de acceso, rectificación, supresión, limitación del tratamiento, portabilidad u oposición con los que pudieran contar.

### VIGESIMOCUARTA. Ley aplicable y jurisdicción competente

El Contrato se regirá e interpretará conforme a la legislación española y, en particular, al Real Decreto Legislativo 1/1996, de 12 de abril, por el que se aprueba el texto refundido de la Ley de Propiedad Intelectual.

Las Partes se someten para la resolución de cualesquiera disputas o reclamaciones derivadas de la interpretación o ejecución del Contrato, incluyendo todas aquellas obligaciones no contractuales derivadas o relativas al Contrato, a la jurisdicción de los Juzgados y Tribunales competentes conforme a derecho.

**EN VIRTUD DE LO CUAL**, las Partes reconocen haber leído en su totalidad el Contrato, manifiestan comprenderlo, y aceptan obligarse por sus términos y condiciones, constituyendo el completo y el total acuerdo de las Partes. Y, en prueba de conformidad, las Partes firman el presente Contrato en todas sus hojas, y en tantas copias originales como Partes participen en el Contrato, constituyendo todas esas copias un único acuerdo, en el lugar y fechas indicados en el encabezamiento.

**EL DESARROLLADOR**

Riquelmer Añez Joffre

**EL CLIENTE**

DNI/NIF: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. Gráfico del crecimiento Indie en los videojuegos [↑](#footnote-ref-2)
2. Diagrama de Gantz [↑](#footnote-ref-3)
3. Diagrama Entidad Relación [↑](#footnote-ref-4)
4. 3 Diagrama de casos de Uso [↑](#footnote-ref-5)
5. Diagrama de clases [↑](#footnote-ref-6)
6. Diagrama secuencial [↑](#footnote-ref-7)
7. Diamagra de Estados [↑](#footnote-ref-8)
8. Pantalla de juego [↑](#footnote-ref-9)
9. Mock up menú [↑](#footnote-ref-10)
10. Mockup pantalla principal [↑](#footnote-ref-11)
11. Diagrama de componentes [↑](#footnote-ref-12)
12. Función Update() [↑](#footnote-ref-13)
13. Funciones de comprobación posicional [↑](#footnote-ref-14)
14. Función de salto [↑](#footnote-ref-15)
15. Función auxiliar Salto [↑](#footnote-ref-16)
16. Script de ataque [↑](#footnote-ref-17)