

La leyenda de Masamune

Carlos Santos Vilchez

Grado de ingeniería informática

Videojuegos

Guillermo García Romero

Joan Arnedo Moreno

03/01/2021



Esta obra está sujeta a una licencia de Reconocimiento-NoComercial-SinObraDerivada [3.0 España de Creative Commons](https://creativecommons.org/licenses/by-nc-nd/3.0/es/)

FICHA DEL TRABAJO FINAL

Título del trabajo:	<i>La Leyenda de Masamune</i>
Nombre del autor:	<i>Carlos Santos Vílchez</i>
Nombre del consultor/a:	<i>Guillermo García Romero</i>
Nombre del PRA:	<i>Joan Arnedo Moreno</i>
Fecha de entrega (mm/aaaa):	01/2021
Titulación:	<i>Plan de estudios del estudiante</i>
Área del Trabajo Final:	<i>Videojuegos</i>
Idioma del trabajo:	<i>Castellano</i>
Palabras clave	<i>Action-RPG, 2D, Videojuego</i>
<p>Resumen del Trabajo (máximo 250 palabras): <i>Con la finalidad, contexto de aplicación, metodología, resultados i conclusiones del trabajo.</i></p>	
<p>La finalidad de este proyecto es la de crear un videojuego, ya que es el tema central del área elegida para la realización del TFG. Todo ello utilizando metodologías y técnicas aprendidas durante el transcurso del grado, además de otros conocimientos adquiridos de manera externa para poder desarrollar el videojuego de una manera satisfactoria.</p> <p>El proyecto “<i>La leyenda de Masamune</i>” se basa en el desarrollo de un videojuego 2D de género “<i>Action-RPG</i>”, utilizando el motor gráfico <i>Unity</i> e intentando ser fiel a los juegos de este género que solían lanzarse para dispositivos de 8 ó 16 bits, por lo que muchos de los elementos que salen en la versión final del proyecto están inspirados en dichos juegos.</p> <p>La metodología a utilizar para la creación del proyecto es simple, se trata de avanzar el desarrollo en una serie de etapas bien definidas, dichas etapas encuadran tareas como la búsqueda de recursos, la disposición de dichos recursos en el juego, la creación de animaciones, el desarrollo de <i>scripts</i>, el diseño de niveles, etc.</p> <p>Como conclusión, puedo decir que la creación de un videojuego desde cero y con un límite de tiempo para ello ha supuesto un verdadero reto, pero muy gratificante por la cantidad de conocimientos adquiridos, ya no solo sobre cómo se crea un videojuego, sino por poder comprobar lo que cuesta y el mérito que tienen muchos estudios que lanzan juegos verdaderamente trabajados.</p>	

Abstract (in English, 250 words or less):

The objective of this project is to make a videogame from scratch, since it is the main subject of the chosen area of the final degree project. To be able to do that it will be necessary to make use of all the methodologies and skills learned throughout the degree, besides some other knowledge acquired on the net about creating games.

The project "*La leyenda de Masamune*" is a 2D *Action-RPG* game made with Unity, in which some of its features will be inspired in old 8/16 bit games that made this genre shine some years ago. So you can expect to recognize such things as movements, menus, items, etc from older games.

The methodology used to create this project is quite simple, it was all about designing well defined stages that allowed me to develop the game step by step in a linear way. These stages let me develop the game in separate "components", such as looking for resources, creating animations, script developing...

To conclude, I would like to say that making a game from scratch with a limited time has taught me a lot and it has been a real challenge for me. But I also really enjoyed it because this journey has made me learn a lot about this world, not only about how to create games but also about how hard it is to create a well made videogame.

Índice

1. Introducción.....	3
1.1 Contexto y justificación del Trabajo	3
1.2 Objetivos del Trabajo.....	3
1.3 Enfoque y método seguido.....	4
1.4 Planificación del Trabajo	5
1.5 Breve resumen de productos obtenidos	6
1.6 Breve descripción de los otros capítulos de la memoria.....	6
2. Estado del arte	7
2.1 Género del juego.....	7
2.2 Referencias del género	8
2.2 Plataformas de desarrollo.....	9
3. Definición del juego	12
3.1 Historia y ambientación	12
3.2 Tipo de interacción juego-jugador	12
3.3 Definición de los personajes / elementos	13
3.4 Interacción entre los actores del juego	16
3.5 Objetivos planteados al jugador	17
3.6 Concept Art: <i>screenshots</i> , animación... ..	18
3.7 Plataforma de destino.....	19
3.8 Cuantificación de tiempo y recursos por objetivo	20
4. Diseño técnico.....	21
4.1 Entorno de desarrollo elegido.....	21
4.2 Requerimientos técnicos del entorno	21
4.3 Inventario y descripción de las herramientas utilizadas.....	22
4.4 Assets y recursos del juego.....	24
4.5 Esquema de la arquitectura del juego	35
4.6 IA de los enemigos	35
5. Diseño de niveles	37
5.1 Diseño del mapa	37
5.2 Sistema de misiones	39
5.3 Características de los enemigos	39
6. Manual de usuario	40
6.1 Requisitos mínimos	40
6.2 Instrucciones de juego.....	40
7. Conclusiones.....	43
7.1 Conclusiones del trabajo	43
7.2 Logro de los objetivos planteados	43
7.3 Seguimiento de la planificación y metodología.....	44
7.4 Líneas de trabajo futuro.....	44
8. Glosario	45
9. Bibliografía	47
10. Anexos	48

Lista de figuras

- Ilustración 1. Diagrama de *Gantt*.
- Ilustración 2. *The Legend of Zelda*.
- Ilustración 3. *Final Fantasy VI*.
- Ilustración 4. *Alundra*.
- Ilustración 5. Motores gráficos.
- Ilustración 6. Logo de *Unity*.
- Ilustración 7. Logo de *Unreal*.
- Ilustración 8. Logo de *GameMaker Studio*.
- Ilustración 9. Logo de *RPG Maker*.
- Ilustración 10. *Sprite* de personaje principal.
- Ilustración 11. *Sprite* de *NPC*.
- Ilustración 12. *Sprite* de *NPC*.
- Ilustración 13. *Sprite* de *NPC*.
- Ilustración 14. *Sprite* de *NPC*.
- Ilustración 15. *Sprite* de *NPC*.
- Ilustración 16. *Sprite* de *NPC*.
- Ilustración 17. *Sprite* de *NPC*.
- Ilustración 18. *Sprite* de enemigo.
- Ilustración 19. *Sprite* de enemigo.
- Ilustración 20. *Sprite* de enemigo.
- Ilustración 21. *Sprite* de enemigo.
- Ilustración 22. *Sprite* de enemigo.
- Ilustración 23. *Sprite* de enemigo.
- Ilustración 24. *Sprite* de enemigo.
- Ilustración 25. *Sprite* de poción.
- Ilustración 26. *Sprite* de animal.
- Ilustración 27. *Sprite* de animal.
- Ilustración 28. *Sprite* de animal.
- Ilustración 29. *Sprite* de animal.
- Ilustración 30. *Sprite* de animal.
- Ilustración 31. *The Legend of Zelda*.
- Ilustración 32. Animación de *Link*.
- Ilustración 33. Museo de Tokio.
- Ilustración 34. Logo de *Unity*.
- Ilustración 35. Requerimientos mínimos de *Unity*.
- Ilustración 36. *Hardware* personal.
- Ilustración 37. Logo de *Unity*.
- Ilustración 38. Logo de *Adobe Photoshop*.
- Ilustración 39. Logo de *Visual Studio*.
- Ilustración 40. Logo de *Audacity*.
- Ilustración 41. Logo de *Windows*.
- Ilustración 42. Logo de *DaVinci Resolve*.
- Ilustración 43. *Tileset* 1.
- Ilustración 44. *Tileset* 2.
- Ilustración 45. *Tileset* 3.
- Ilustración 46. *Tileset* 4.
- Ilustración 47. *Tileset* 5.

Ilustración 48. *Tileset* 6.
Ilustración 49. *Tileset* 7.
Ilustración 50. *Tileset* 8.
Ilustración 51. *Sprites* 1.
Ilustración 52. *Sprites* 2.
Ilustración 53. *Sprites* 3.
Ilustración 54. *Sprites* 4.
Ilustración 55. *Sprites* 5.
Ilustración 56. *Sprites* 6.
Ilustración 57. *Sprites* 7.
Ilustración 58. *Sprites* 8.
Ilustración 59. *Sprites* 9.
Ilustración 60. *Sprites* 10.
Ilustración 61. *Sprites* 11.
Ilustración 62. Panel central del inventario.
Ilustración 63. Panel lateral del inventario 1.
Ilustración 64. Panel lateral del inventario 2.
Ilustración 65. Cuadro de diálogo.
Ilustración 66. Barra de vitalidad.
Ilustración 67. Efectos sonoros.
Ilustración 68. Música del juego.
Ilustración 69. Fuente.
Ilustración 70. Arquitectura del juego.
Ilustración 71. Estados de animación de samuráis.
Ilustración 72. Estados de animación de lobos.
Ilustración 73. Mapa del juego.
Ilustración 74. Aldea.
Ilustración 75. Bosque 1.
Ilustración 76. Bosque 2.
Ilustración 77. Castillo.
Ilustración 78. Pantalla de título.
Ilustración 79. Pantalla de controles.
Ilustración 80. Pantalla de trama.
Ilustración 81. Pantalla de juego.
Ilustración 82. Pantalla de diálogo.
Ilustración 83. Pantalla de inventario.

1. Introducción

1.1 Contexto y justificación del Trabajo

El sector de los videojuegos ha sufrido numerosos cambios y avances a lo largo de los años desde su invención hace unas décadas. Estos avances han permitido que la calidad de los juegos haya aumentado a límites casi fotorrealistas, por lo que su popularidad ha ido aumentando exponencialmente, siendo una de las industrias más rentables en la actualidad.

Todo ello ha supuesto la creación de miles de estudios que se dedican, en exclusiva, al desarrollo de videojuegos, trayéndonos creaciones de todo tipo, desde historias fantásticas en épocas medievales a títulos deportivos. Es más, hoy en día cualquiera puede crear su propio videojuego, solo se necesita un ordenador, tiempo y ganas.

Por todo ello, además de por ser una de mis mayores aficiones desde que tengo uso de razón, he decidido realizar para el proyecto de fin de grado mi propio juego, pues me parece una forma entretenida, amena y divertida de finalizar mis estudios del grado de ingeniería informática.

1.2 Objetivos del Trabajo

Los objetivos marcados para este proyecto están relacionados con la creación de un videojuego, desde sus fases más tempranas hasta su versión final. Se pueden clasificar de la siguiente manera:

- Conceptualización del juego
- Familiarización de uso del motor *Unity*
- Búsqueda de recursos
- Diseño de la estructura del juego
- Diseño de menús
- Creación del entorno
- Creación de animaciones de personajes
- Creación de scripts para añadir funcionalidades
- Diseño del sistema de diálogos
- Diseño del sistema de misiones

- Diseño narrativo de la historia
- Diseño de un nivel completamente jugable
- Documentación de todo el desarrollo
- Creación de un tráiler promocional
- Creación de la defensa del proyecto

1.3 Enfoque y método seguido

El proyecto se basa en el desarrollo de un videojuego desde cero, por lo que no se adapta ningún producto existente, ya que mi idea es la de pasar por todas las fases que se necesitan para la creación de un videojuego, como si de un estudio se tratase.

Lo que sí que he utilizado han sido recursos gratuitos de internet, tanto gráficos como sonoros, ya que el tiempo que tenemos para el desarrollo del proyecto es muy limitado, por lo que, si tuviera que diseñar también el apartado gráfico seguramente no acabaría el juego a tiempo o, si lo acabara, el resultado sería mucho más pobre, ya que tendría menos tiempo para dedicar al resto de fases de desarrollo.

En cuanto a la metodología, he aplicado el modelo "Desarrollo en cascada", pues como he comentado anteriormente mi desarrollo está fijado en diversas etapas, por lo que hasta que no se acaba una etapa no se avanza a la siguiente. La elección de esta metodología se debe a que permite crear un producto sólido, desde los cimientos hasta el final.

Esta metodología cuenta, normalmente, con las siguientes fases:

- Análisis
- Diseño
- Implementación
- Pruebas
- Mantenimiento

Los objetivos establecidos en el apartado anterior irán encuadrados en estas fases, de manera que la creación del juego pueda seguir un ciclo de vida adecuado y fácil de mantener y escalar.

1.4 Planificación del Trabajo

Las herramientas necesarias para la realización de este proyecto son mi ordenador personal y una conexión a internet. En mi ordenador se encuentra todo el *software* a utilizar para el desarrollo del juego y en internet todos los recursos y conocimientos necesarios para crear un título de cierta calidad.

El *software* utilizado para el proyecto es el siguiente:

- **Unity 2020.1.6f1**
- **Visual Studio 2019**
- **Audacity**
- **Adobe Photoshop CC 2019**
- **Sistema de grabación de pantalla de Windows 10**
- **DaVinci Resolve 16**

Las tareas a realizar y la línea temporal se encuentran encuadradas en el siguiente diagrama de Gantt:

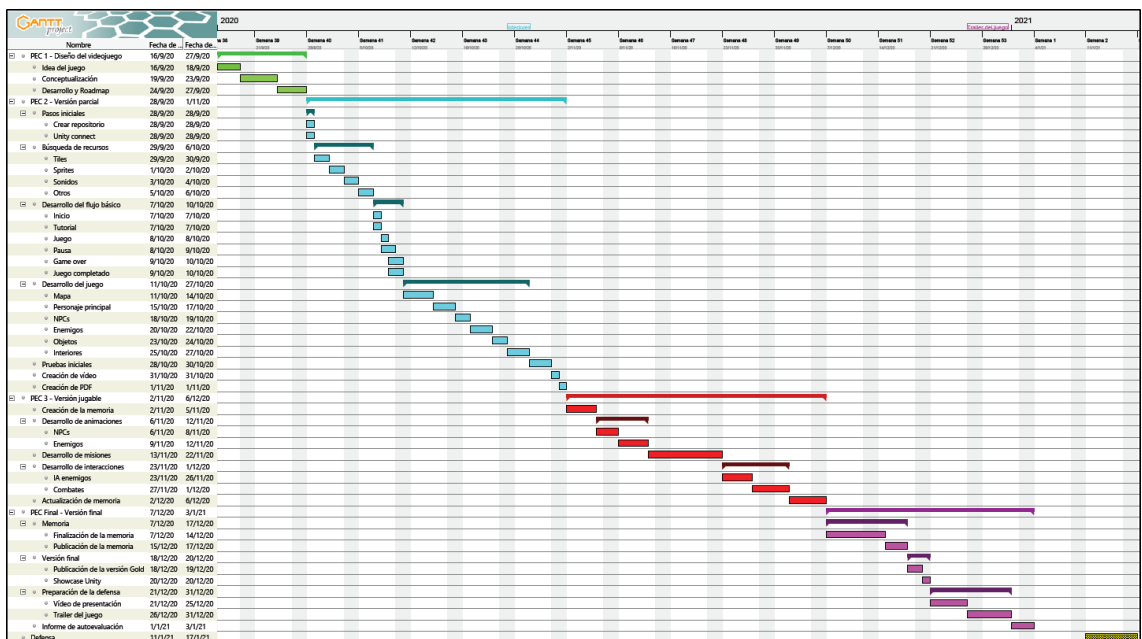


Ilustración 1. Diagrama de Gantt

1.5 Breve resumen de productos obtenidos

Los productos obtenidos durante el semestre son los siguientes:

- Vídeo PEC 2
- Vídeo PEC 3
- Ejecutable de la versión parcial
- Ejecutable de la versión jugable
- Ejecutable de la versión final
- Tráiler promocional
- Repositorio *GitHub* con el proyecto entero
- Memoria
- *Showcase* de *Unity Connect*
- Defensa del proyecto
- Informe de autoevaluación

1.6 Breve descripción de los otros capítulos de la memoria

La descripción del resto de capítulos de la memoria del proyecto es la siguiente:

- Capítulo 2: Conceptualización del videojuego.
- Capítulo 3: Trama, personajes y mecánicas del juego.
- Capítulo 4: Elementos técnicos del proyecto.
- Capítulo 5: Diseño del nivel del juego en detalle.
- Capítulo 6: Instrucciones de uso del juego.
- Capítulo 7: Conclusiones personales del proyecto.
- Capítulo 8: Glosario de términos poco comunes.
- Capítulo 9: Bibliografía con enlaces de interés.
- Capítulo 10: Anexos del proyecto.

2. Estado del arte

2.1 Género del juego

El género elegido para mi proyecto es el de **Action-RPG** en 2D. La elección de este género se debe a que siempre me han gustado los juegos que mezclan el rol con la acción, pues permiten explorar y vivir grandes aventuras sin que el juego llegue a aburrir al jugador, ya que los combates se desarrollan de una manera rápida y dinámica, siempre teniendo la estrategia como fondo.

Podemos decir que este género, que ha ido eclipsando a los juegos RPG tradicionales por turnos con el paso de los años, mezcla elementos roleros, como la personalización de personajes, subida de niveles, equipación de armas y armaduras... con un sistema de combate más enfocado a la acción.

La elección de hacerlo en 2D es básicamente debido al tiempo del que disponemos para realizar el proyecto, pues facilita el desarrollo del juego al no tener que crear o buscar *assets* en 3D que puedan representar la idea del juego, entre otras cosas. Además, los juegos de este tipo en 2D siguen siendo bastante populares, pues se crearon grandes obras en generaciones pasadas que nunca pasan de moda.

Los videojuegos del género *Action-RPG* suelen tener las siguientes características:

- **Único personaje:** se controla a un único personaje, tanto para moverse por el mapa como para combatir.
- **Desarrollo del personaje:** conforme avanza la historia y se derrotan enemigos el personaje sufre una serie de *power-ups* que pueden ser como los juegos de rol clásicos, a través de niveles y subida de *stats*, o simplemente aumentando la vida y obteniendo armas más potentes.
- **Sistemas de misiones:** la trama se desarrolla a través de un sistema de misiones, donde se debe superar una para desbloquear la siguiente. También pueden existir misiones secundarias para alargar el juego u obtener objetos especiales.
- **Ítems:** objetos repartidos por el mapa u obtenidos al derrotar enemigos que ayudan al personaje. Estos objetos pueden ser curativos, monetarios, ofensivos, defensivos, necesarios para una misión, etc.

2.2 Referencias del género

Para la realización de mi proyecto he tomado otros títulos como referencia a la hora de implementar funcionalidades, como mecánicas, diálogos, movimientos, animaciones etc. El motivo es que son patrones que suelen repetirse en este tipo de juegos, por lo que, un usuario cuando va a probar una obra de este género espera encontrarse con unas mecánicas de juego con las que ya se encuentra familiarizado.

Por ejemplo, títulos como los siguientes:

- ***The legend of Zelda***

La principal inspiración para mi proyecto ha sido un clásico de Nintendo, como lo es la saga *The legend of Zelda* de la era de los 8 y 16 bits, que mezclaba elementos de rol con puzzles y acción, siendo juegos realmente entretenidos y que gustaban a personas de todas las edades, pues llegaban a suponer un auténtico desafío en determinadas partes de la historia.



Ilustración 2. *The Legend of Zelda*

- ***Final Fantasy***

Aunque no es un *Action-RPG* es una de mis sagas favoritas, por lo que no podían faltar referencias a ella en mi proyecto, de hecho, los menús están inspirados en los del *Final Fantasy 7*, además de algunos diálogos y colores.



Ilustración 3. *Final Fantasy VI*

- **Alundra**

Otro de los grandes clásicos del género, con la particularidad de que era muy difícil de completar debido a la dificultad, aun así, no podía faltar en mi repertorio a modo de inspiración.



Ilustración 4. Alundra

2.2 Plataformas de desarrollo

Años atrás, cuando no existían todas las herramientas de las que disponemos en la actualidad, los desarrolladores tenían que romperse realmente la cabeza para crear sus motores gráficos y/o videojuegos, aplicando fórmulas matemáticas y físicas para que los personajes pudieran moverse, atacar, etc.

Todo eso pasó a la historia, pues actualmente tenemos motores gráficos gratuitos o de pago a nuestra disposición, por lo que cualquier persona puede crear su propio videojuego si se lo propone, ya que dichos motores te ahorran muchísimo trabajo, permitiéndote que te dediques a lo realmente importante, el diseño de tu juego.

Como he comentado, existen diversos motores que pueden ser utilizados por cualquier persona. Si bien es cierto que las grandes desarrolladoras utilizan sus propios motores, como pueden ser *Rockstar*, *CD Projekt Red* o *Square Enix*, la mayoría de estudios y desarrolladores utilizan motores de empresas externas, ya que no todos pueden permitirse la creación de su propio motor y es una manera de ahorrar costes.



Ilustración 5. Motores gráficos

Los más populares son los siguientes:

- **Unity**

Unity es un motor que permite crear videojuegos tanto en 2D como en 3D, además de ser gratuito (solo se paga al conseguir unas ganancias superiores a 100.000\$ o para obtener más características). Es un motor muy popular debido a su sencillez de uso, aunque requiere de conocimientos de programación para la realización de *scripts*, utilizando el lenguaje *C#*, lenguaje orientado a objetos.

Además, es un motor muy completo, pues tiene como características la creación de animaciones, implementación avanzada de audio y efectos sonoros... y, lo más importante, la exportación del juego a cualquier plataforma que deseemos de forma automática (*Android, IOs, Windows, Mac...*), por lo que solo debemos preocuparnos de la creación del videojuego.



Ilustración 6. Logo de *Unity*

- **Unreal Engine**

Unreal Engine es un potente motor para la creación de videojuegos o animaciones, además de ser gratuito desde hace poco tiempo (solo se paga un pequeño porcentaje en función a las ganancias obtenidas con la venta del juego). Requiere conocimientos de programación, pues es necesario la escritura de *scripts* para la creación de videojuegos.

Además, permite el uso de dos formas de crear scripts: *C++* y *Blueprints*. *C++* es un lenguaje de programación basado en *C* orientado a objetos de bajo nivel, mientras que *Blueprints* se basa en un sistema de nodos. Con este motor se han creado grandes éxitos mundiales como *Fortnite*.



Ilustración 7. Logo de *Unreal*

- **GameMaker Studio**

Motor de pago enfocado a usuarios con pocos conocimientos de programación, ya que permite la opción de 'arrastrar y soltar' para ir creando el videojuego sin tener que programar, aunque también permite el uso de *scripts* para los usuarios más avanzados a través del lenguaje de programación propio del motor, llamado *Game Maker Language*.

Es un motor utilizado generalmente para la creación de juegos 2D, aunque a través de *scripts* es posible la creación de juegos en 3D. También, el motor trae una serie de bibliotecas que permiten tales cosas como el movimiento de personajes u otras funciones avanzadas.



Ilustración 8. Logo de GameMaker Studio

- **RPG Maker**

RPG Maker es un motor o conjunto de herramientas enfocada a la creación de juegos RPG. Tiene la ventaja de que es muy fácil de usar, ya que abstrae bastante al usuario de lo que es la programación, por lo que se puede ir creando el juego, simplemente, a base de ir arrastrando elementos.

Como contras podemos decir que es un producto de pago, a diferencia de otros motores comentados anteriormente, además de no haber grandes títulos hechos con esta herramienta en el mercado, debido a que no es tan potente como otros motores.



Ilustración 9. Logo de RPG Maker

3. Definición del juego

3.1 Historia y ambientación

Como he comentado en la descripción del juego, este estará ambientado en el Japón feudal de la era Edo, donde el *Shogun* era el gobernador del país y los samuráis los guerreros, que podían estar de su parte o en su contra, propiciando diversas guerras por el poder a lo largo de los años.

En mi juego controlaremos a un *Ronin* (samurái sin dueño) que, vagando por las tierras del país, llega a un pequeño poblado. Al llegar a dicho poblado nuestro personaje se encontrará envuelto en una serie de acontecimientos, que tendrán que ver con los continuos saqueos y ataques por parte de una banda de forajidos a los aldeanos.

Los aldeanos, aterrorizados, le ofrecen a nuestro personaje una gran recompensa a cambio de que acabe con ellos y, sobre todo, que recupere la legendaria espada *Masamune*, robada y custodiada por el líder de los bandidos. Además, ya que atacar directamente sería un suicidio deberemos, a través de una serie de misiones, eliminar a los forajidos uno por uno, hasta llegar al líder de la banda, que será el jefe final.

3.2 Tipo de interacción juego-jugador


El jugador solo podrá controlar al personaje protagonista, con el cual podrá realizar una serie de acciones, que son las siguientes:

- **Movimiento:** Teclas *WASD*
- **Correr:** *Shift* izquierdo
- **Ataque:** *Click* derecho del ratón
- **Hablar con *NPCs* o interactuar con objetos:** Botón *Q*
- **Consumir objetos curativos:** Tecla *C* o *click* en el icono
- **Abrir menú de pausa:** Tecla *E* o *click* en el icono
- **Interactuar con elementos de los menús:** *Click* izquierdo


3.3 Definición de los personajes / elementos


El juego cuenta con una serie de personajes/elementos repartidos por el mapa, cada uno con sus diálogos, movimientos, recorridos, aspectos, etc. Consiguiendo, de esta manera, crear un mundo 2D vivo, donde el jugador pueda experimentar que no está solo mientras recorre el mapa para ir completando las diversas misiones. Los personajes/elementos de los que dispone el juego son los siguientes:


Personaje principal:


Nombre	Ryu
Vitalidad	100
Katana	Kazan
Descripción	Protagonista y personaje jugable
Aspecto	 Ilustración 10. Sprite de personaje principal


NPCs:

Nombre	Aldeano 1
Descripción	Aldeano anciano
Aspecto	 Ilustración 11. Sprite de NPC


Nombre	Aldeano 2
Descripción	Aldeana anciana
Aspecto	 Ilustración 12. Sprite de NPC

Nombre	Aldeano 3
Descripción	Aldeano de mediana edad
Aspecto	 Ilustración 13. Sprite de NPC


Nombre	Aldeano 4
Descripción	Aldeana de mediana edad
Aspecto	 Ilustración 14. Sprite de NPC


Nombre	Aldeano 5
Descripción	Aldeano joven
Aspecto	 Ilustración 15. Sprite de NPC


Nombre	Aldeano 6
Descripción	Aldeana joven
Aspecto	 Ilustración 16. Sprite de NPC


Nombre	Aldeano 7
Descripción	Aldeana niña
Aspecto	 Ilustración 17. Sprite de NPC


Forajidos:


Nombre	Mitsuhide
Vitalidad	150
Daño	4
Velocidad	1.3
Katana	Umi
Descripción	Primer enemigo del juego, cuenta con unos <i>stats</i> bajos para que sea fácilmente derrotable
Aspecto	 Ilustración 18. Sprite de enemigo


Nombre	Ujimasa
Vitalidad	170
Daño	6
Velocidad	1.6
Katana	Kaji
Descripción	Segundo enemigo, un poco más rápido y fuerte que el anterior
Aspecto	 Ilustración 19. Sprite de enemigo

Nombre	Yoritomo
Vitalidad	200
Daño	10
Velocidad	1.4
Katana	Mizu
Descripción	Tercer enemigo, lento, pero con un gran daño
Aspecto	 Ilustración 20. Sprite de enemigo


Nombre	Yukimura
Vitalidad	190
Daño	3
Velocidad	1.9
Katana	Doku
Descripción	Enemigo muy veloz, pero provoca poco daño
Aspecto	 <p>Ilustración 21. Sprite de enemigo</p>

Nombre	Hideyoshi
Vitalidad	220
Daño	8
Velocidad	1.6
Katana	Chi
Descripción	Enemigo ligeramente veloz, a la vez que fuerte
Aspecto	 <p>Ilustración 22. Sprite de enemigo</p>


Nombre	Kojiro
Vitalidad	300
Daño	8
Velocidad	1.7
Katana	Jigoku
Descripción	Enemigo con <i>stats</i> altas y balanceadas
Aspecto	 <p>Ilustración 23. Sprite de enemigo</p>


Nombre	Musashi
Vitalidad	600
Daño	10
Velocidad	1.8
Katana	Masamune
Descripción	Jefe final del juego, rápido y fuerte
Aspecto	 <p>Ilustración 24. Sprite de enemigo</p>


Objetos:


Nombre	Poción
Descripción	Objeto curativo
Puntos de vitalidad	40
Aspecto	 <p>Ilustración 25. Sprite de poción</p>


Animales:

Nombre	Lobo
Vitalidad	50
Daño	6
Velocidad	1
Descripción	Enemigo débil que merodea por zonas recónditas del bosque
Aspecto	 Ilustración 26. Sprite de animal

Nombre	Perro
Descripción	Animal inofensivo que anda por la aldea
Aspecto	 Ilustración 27. Sprite de animal

Nombre	Gato
Descripción	Animal inofensivo que anda por la aldea
Aspecto	 Ilustración 28. Sprite de animal

Nombre	Gallina
Descripción	Animal inofensivo que anda por las afueras de la aldea
Aspecto	 Ilustración 29. Sprite de animal

Nombre	Mariposa
Descripción	Insecto inofensivo que vuela por las afueras de la aldea
Aspecto	 Ilustración 30. Sprite de animal

3.4 Interacción entre los actores del juego

La interacción entre los actores y elementos del juego se producen siempre entre el personaje principal y el resto.

- **Con NPCs:** nuestro personaje podrá hablar con los *NPCs* a través de unos diálogos al encontrarse cerca de ellos (dentro de su radio de acción) y pulsar el botón de hablar (también se producen diálogos automáticos al acceder a una zona determinada), además de poder recibir misiones de alguno de ellos.

- **Con enemigos:** nuestro personaje entablará combate con los enemigos al entrar en su radio de acción, dónde los enemigos perseguirán a nuestro personaje una vez éste entre dentro de dicho radio. Cada vez que un enemigo toque al personaje perderá puntos de vida, mientras que, si nuestro personaje consigue acertar en el enemigo, pulsando el botón de ataque, hará lo propio con dicho enemigo.
- **Con objetos:** nuestro personaje podrá recuperar vida a través del uso de pociones, con un botón designado a ello, que se encontrarán repartidas por el mapa, y para recogerlas habrá que ponerse encima de ellas.
- **Con otros:** nuestro personaje podrá leer carteles, interactuar con objetos o entrar en casas que tengan diseñada un interior. Para leer un cartel o interactuar con un objeto bastará con pulsar el botón de hablar, y para entrar en una casa habrá que ponerse encima de la puerta.

3.5 Objetivos planteados al jugador

El objetivo del jugador es el de completar todas las misiones que le sean proporcionadas por los *NPCs*, misiones que seguirán un orden lógico para la narrativa de la historia y que obligarán al jugador a moverse a determinadas zonas del mapa.

El reto para completar las misiones consiste en tener que derrotar a una serie de enemigos sin llegar a morir, enemigos que podrán ser no importantes (lobos), jefes (forajidos) o el jefe final (el líder de los forajidos), que estará reservado para la última misión.

Las misiones disponibles son las siguientes:

Nº de misión	Primera misión
Objetivo	Reunir información en la aldea

Nº de misión	Segunda misión
Objetivo	Adentrarse en el bosque

Nº de misión	Tercera misión
Objetivo	Derrotar al primer enemigo

Nº de misión	Cuarta misión
Objetivo	Derrotar al segundo enemigo

Nº de misión	Quinta misión
Objetivo	Derrotar al tercer enemigo

Nº de misión	Sexta misión
Objetivo	Derrotar al cuarto enemigo

Nº de misión	Séptima misión
Objetivo	Derrotar al quinto enemigo

Nº de misión	Octava misión
Objetivo	Derrotar al sexto enemigo

Nº de misión	Novena misión
Objetivo	Derrotar al jefe final

Nº de misión	Última misión
Objetivo	Regresar a la aldea

3.6 Concept Art: *screenshots*, animación...

Debido a mis nulas dotes de dibujo utilizaré capturas del juego **Zelda: Minish cup** para explicar cómo se ve el juego:

- **Gameplay:**



Ilustración 31. The legend of Zelda

El juego luce, básicamente, como en esta captura, es decir, el personaje está centrado en la pantalla, por lo que al moverlo la pantalla lo va siguiendo. La *UI* no es igual, pues sólo se muestra una barra de vida, el botón de inventario y el botón de pociones (con el número de pociones que poseemos).

Lo que sí que cambia drásticamente es el estilo de los *sprites* y de los *tiles*, pues al ambientarse en el Japón feudal, las casas, árboles y personajes lucen como en esa época. Por lo que, los personajes utilizan ropa de la época, además de armas japonesas en lugar de espadas convencionales.

- **Animaciones:**

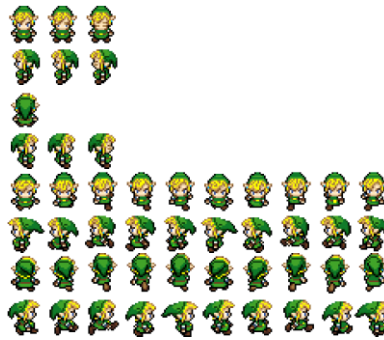


Ilustración 32. Animación de Link

Los únicos elementos animados del juego son el personaje principal, los *NPCs*, los animales y los enemigos, y se hace tal y como se muestra en la imagen, donde cada *sprite* simula un movimiento. El personaje principal y los enemigos son los que más animaciones contienen, pues aparte de las animaciones de movimiento también disponen de animaciones de ataque.

- **Estilo japonés:**

Extendiendo la información sobre que los *sprites* y los *tiles* están ambientados en el Japón feudal, decir que se han tomado como referencia gráfica para el diseño de la aldea imágenes de este estilo, ya que representan la vida tal y como era en el periodo Edo de Japón:



Ilustración 33. Museo de Tokio

3.7 Plataforma de destino

La plataforma de destino es la del *PC (Windows)*, ya que es el sistema operativo más extendido entre los jugadores, aunque es cierto que el juego podría encajar también en dispositivos móviles, pero no se hace por falta de tiempo y para poder dedicar más tiempo a otros apartados del desarrollo.

3.8 Cuantificación de tiempo y recursos por objetivo

La cuantificación de tiempo ya se encuentra reflejada en el diagrama de Gantt por lo que me centraré en los recursos utilizados para cada objetivo:

- **Recursos para la creación del videojuego:** Como ya he comentado, se ha utilizado *Unity* como motor para la creación del proyecto, además del entorno de programación *Visual Studio* para el desarrollo de *scripts*.
- **Recursos gráficos:** Para el apartado gráfico del juego se han utilizado *tiles* y *sprites* gratuitos sacados de repositorios de internet. Para el diseño de los menús o pequeñas ediciones a los *sprites* se ha utilizado el *software Adobe Photoshop CC 2019*.
- **Recursos sonoros:** Los sonidos también han sido sacados de repositorios gratuitos de internet o producidos por mí mismo, haciendo uso del *software Audacity* cuando ha sido necesario.
- **Recursos audiovisuales:** Para la creación y edición de vídeos (el tráiler o las presentaciones) se ha utilizado el *software DaVinci Resolve*.
- **Recursos de hardware:** He utilizado mi propio ordenador personal para realizar todas las tareas.
- **Recursos humanos:** Todo el proyecto ha sido desarrollado por mí, a excepción del material gráfico, debido a que no poseo conocimientos de diseño gráfico.

4. Diseño técnico

4.1 Entorno de desarrollo elegido

El motor que he elegido para la creación de mi proyecto es **Unity**, ya que es gratuito y una gran elección para la creación de videojuegos 2D como el que voy a hacer, debido a su sistema de *tiles* y *sprites*. También posee una gran comunidad en caso de dudas y una tienda donde poder conseguir *scripts* o elementos gráficos y sonoros que hagan que el juego tenga más calidad.

Por otro lado, ya estoy familiarizado con dicho motor, pues he creado varios proyectos personales con él con buenos resultados, aunque la mayoría inacabados por falta de tiempo.



Ilustración 34. Logo de Unity

4.2 Requerimientos técnicos del entorno

Unity no requiere de un gran equipo para poder ejecutar y utilizar el motor de manera correcta, por lo que prácticamente cualquier persona con un ordenador más o menos moderno puede hacer uso de él. Los requisitos mínimos para poder utilizar **Unity** (versión **2019.1**) son los siguientes:

System	Minimum requirements
Operating system	Windows: 7 SP1+, 8, 10, 64-bit versions only macOS: 10.12+ Linux: Fixed at: Ubuntu 16.04, 18.04 and CentOS 7 Server versions of Windows and OS X are untested.
CPU	SSE2 instruction set support.
GPU	Graphics card with DX10 (shader model 4.0) capabilities.
Devices	
iOS	Mac computer running minimum macOS 10.12.6 and Xcode 9.4 or higher.
Android	Android SDK and Java Development Kit (JDK). IL2CPP scripting backend requires Android NDK.
Universal Windows Platform	Windows 10 (64-bit), Visual Studio 2015 with C++ Tools component or later and Windows 10 SDK.

Ilustración 35. Requerimientos mínimos de Unity

Mi equipo supera ampliamente los requisitos mínimos, por lo que no me he encontrado ningún problema a la hora de desarrollar el proyecto durante todo el semestre.

El *hardware* de mi equipo es el siguiente:

Nombre del equipo:	CARLOS-GS73VR7R
Sistema operativo:	Windows 10 Home 64 bits (10.0, compilación 19041)
Idioma:	español (configuración regional: español)
Fabricante del sistema:	Micro-Star International Co., Ltd.
Modelo del sistema:	GS73VR 7RF
BIOS:	E17B1IMS.31B
Procesador:	Intel(R) Core(TM) i7-7700HQ CPU @ 2.80GHz (8 CPUs), ~2.8GHz
Memoria:	16384MB RAM
Archivo de paginación:	12398MB usados, 6434MB disponibles
Versión de DirectX:	DirectX 12

Ilustración 36. *Hardware* personal

4.3 Inventario y descripción de las herramientas utilizadas

Para el desarrollo de mi proyecto he hecho uso de diversos programas que me han ayudado a poder realizar el juego tal y como tenía en mente en el momento de la conceptualización. Las herramientas utilizadas han sido las siguientes:

- **Unity**

Como he comentado al principio de esta sección, el motor utilizado para la creación de mi proyecto ha sido **Unity**, en su versión **2019.1.6f1**.



Ilustración 37. Logo de Unity

- **Photoshop CC**

Para la edición y creación de elementos gráficos he hecho uso del programa **Adobe Photoshop CC** en su versión de **2019**. Ha sido especialmente útil para la creación de los cuadros de diálogo y menús.



Ilustración 38. Logo de Adobe Photoshop

- **Visual Studio**

Para la creación de scripts he hecho uso del *IDE Visual Studio 2019*, ya que es el que usa por defecto Unity, además de ser un IDE rápido y práctico.



Ilustración 39. Logo de *Visual Studio*

- **Audacity**

He utilizado **Audacity** en su versión **2.4.2** para realizar ediciones de sonido, tanto del juego como del tráiler promocional del proyecto.



Ilustración 40. Logo de *Audacity*

- **Capturadora de Windows 10**

Para la grabación de los vídeos de las PEC 1 y 2, además de los trozos de *gameplay* del tráiler he utilizado la capturadora de pantalla del propio Windows, que se puede abrir utilizando el atajo *Windows + G*.



Ilustración 41. Logo de *Windows*

- **DaVinci Resolve**

Para la edición de vídeo del tráiler y de la defensa he hecho uso del editor de vídeos **DaVinci Resolve** en su versión **16**.



Ilustración 42. Logo de *DaVinci Resolve*

4.4 Assets y recursos del juego

Debido a que mi proyecto se basaba en un RPG necesitaba una gran cantidad de recursos, tanto gráficos como sonoros, por lo que recurrí a elementos gratuitos sacados de internet, excepto por los menús y diálogos que han sido diseñados por mí. Los recursos utilizados son los siguientes, clasificados por categorías:

- **Tiles de entorno**

Un juego de género RPG requiere de un gran número de elementos gráficos para tener un entorno rico y diverso, ya sean casas, tipos de suelo, árboles, plantas... Por ello, hice acopio de una serie de *tilemaps* sacados de internet, de los cuales utilicé las partes que más me interesaban de cada uno.

Los *tiles* utilizados son los siguientes:



Ilustración 43. *Tileset 1*

Del primer *tileset* hice uso de algunos de los árboles, las rocas, las hierbas y la farola. Especial mención a los árboles rosas que simulan los cerezos japoneses.

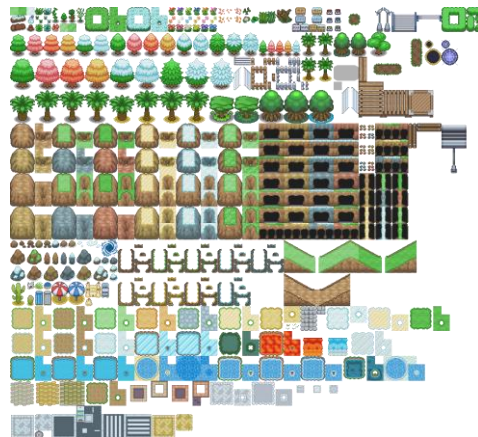


Ilustración 44. *Tileset 2*

Del segundo *tileset* saqué los tipos de suelos utilizados en el juego, el agua de los ríos y acequias, los carteles, hierbas y algunos tipos de piedras.



Ilustración 45. *Tileset 3*

Del *tileset* de casas hice uso de prácticamente todo, pues es lo que le daba el toque de estilo japonés al juego. Utilicé las casas para la aldea, los templos para el bosque y los elementos de castillo para la zona donde se encuentra el jefe final.



Ilustración 46. *Tileset 4*

Del primer *tileset* de interiores hice uso de los suelos, tatamis y paredes, además de utilizar los monumentos por el mapa y el árbol sin hojas para recrear el bosque.



Ilustración 47. *Tileset 5*

Del segundo *tileset* de interiores utilicé la parte que recrea la cocina, además de una de las mesas de madera.



Ilustración 48. *Tileset 6*

Del tercer *tileset* de interiores utilicé elementos decorativos como la mesa, los muebles y armarios, las mamparas, los cojines, las lámparas, libros y pergaminos...

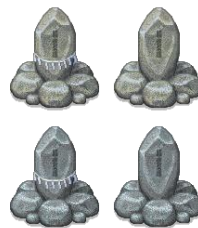


Ilustración 49. *Tileset 7*

También hice uso de tiles puramente decorativos, como el de las piedras, situadas por el bosque y en el castillo. Además de un último *tileset* de “lámparas” japonesas, utilizadas para decorar alguna las casas.



Ilustración 50. *Tileset 8*

- **NPCs**

Para los *NPCs* necesitaba encontrar unos personajes que tuvieran una estética completamente japonesa de la era feudal, por lo que no era una tarea fácil, ya que no es una temática que se suele emplear en este tipo de juegos, que tienden a ser más fantasioso o medievales, aun así, encontré los *sprites* perfectos para mi proyecto:

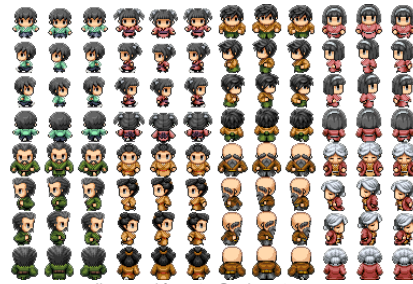


Ilustración 51. *Sprites 1*

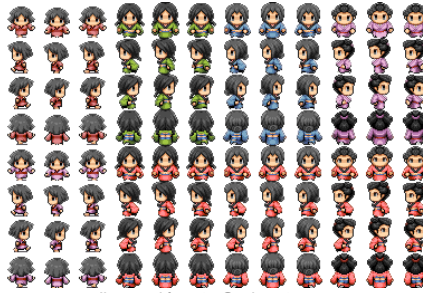


Ilustración 52. *Sprites 2*



Ilustración 53. *Sprites 3*



Ilustración 54. *Sprites 4*



Ilustración 55. Sprites 5

Solo utilicé los *sprites* que me interesaban de cada *tileset*, ya que no era viable la utilización de todos ellos, además de ser un trabajo bastante tedioso la parte de crear las animaciones para cada uno de los personajes.

Los *NPCs* quedaron repartidos por el mapa, quedando unos cuantos, en la aldea, con los que se puede interactuar a modo de conversaciones. Los otros *NPCs* se quedaron esparcidos por el bosque a modo de personas muertas, para dar al juego un toque más macabro y peligroso.

Por ejemplo, los *sprites* de monjes sólo aparecen en la zona de templos, mientras que los *sprites* con pinta de policías aparecen en la zona del castillo.

• Enemigos

En cuanto a los enemigos, forman parte de la misma colección del creador de los *NPCs* anteriores, son los siguientes:



Ilustración 56. Sprites 6

Los elegí como enemigos porque la historia gira en torno a una serie de samurái saqueadores asesinos, y estos *sprites* encajaban perfectamente, además, al tener cada uno un kimono diferente se puede diferenciar bien los unos de los otros, de manera que no resulte monótono para el jugador.

• Animales

También hay animales repartidos por el mapa, para darle un toque más vivo al juego, además de añadir más diversidad de enemigos, pues los lobos atacarán al personaje si este entra en su radio de acción:

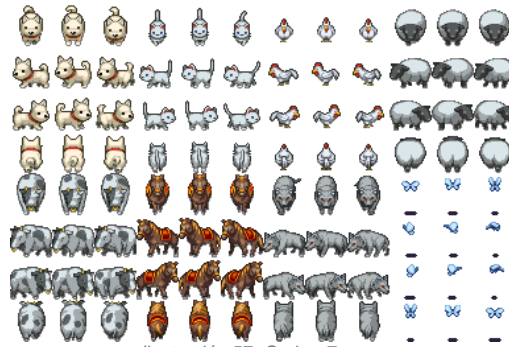


Ilustración 57. Sprites 7

Los animales utilizados en el juego son el perro, el gato, la gallina, la mariposa y el lobo. El perro y el gato se encuentran en la aldea y son interactuables a través de diálogo, mientras que las gallinas y mariposas merodean por la zona exterior a la aldea y pueden ser eliminadas a base de un solo ataque. Los lobos, por otro lado, se encuentran en el bosque y se cuentan como enemigos del juego.

• Personaje principal

El personaje principal es realmente uno de los *sprites* de los *tileset* de *NPCs*, pero me gustó su estética y pensé que sería buena idea dejarlo como personaje jugable:



Ilustración 58. Sprites 8

- **Katanas**

Al ser un juego de samuráis no podían faltar las katanas, además de ser los objetos que realizan daño, tanto a nuestro personaje como a los enemigos:



Ilustración 59. Sprites 9

Se han utilizado un total de ocho katanas, la del personaje y las siete de los enemigos.

- **Items**

También se ha hecho uso de un *tileset* de objetos y estados, de los cuales se han utilizado dos, relativos a pociones y otro con el dibujo de un engranaje para situarlo como botón de abrir el inventario:



Ilustración 60. Sprites 10

- **Sangre**

Para darle al juego un aspecto más macabro y peligroso, que ponga en alerta al jugador, también se han utilizados *tiles* que representan manchas de sangre, que se han repartido por todo el bosque:

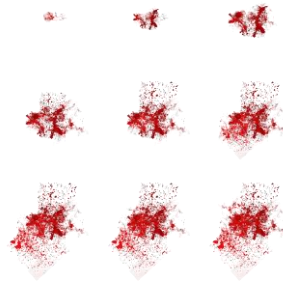


Ilustración 61. Sprites 11

- **Menú**

Para la creación del menú de inventario me inspiré en el juego Final Fantasy VII, tanto en los colores como en la distribución de los cuadros:

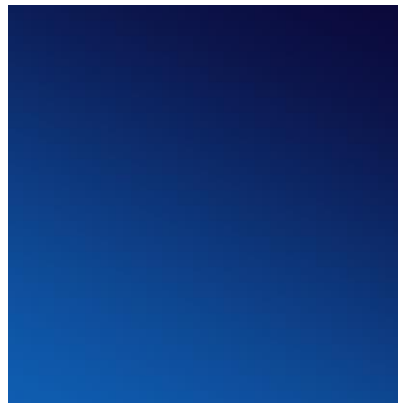


Ilustración 62. Panel central del inventario



Ilustración 63. Panel lateral del inventario 1



Ilustración 64. Panel lateral del inventario 2

• Cuadro de diálogo

El cuadro de diálogo también está inspirado en *Final Fantasy VII*, por lo que tiene la misma tonalidad que el menú de inventario:



Ilustración 65. Cuadro de diálogo

• Barra de vida

En un principio, la vida iba a mostrarse en forma de corazones, como en *Zelda*, pero al final cambié de opinión, pues una barra de vida tradicional da más margen al jugador para sobrevivir:



Ilustración 66. Barra de vitalidad

• Efectos de sonido

Los efectos de sonido reproducen acciones que el personaje puede realizar de manera sonora, por lo que el juego resulta más llamativo para el jugador. Los efectos utilizados son los siguientes:

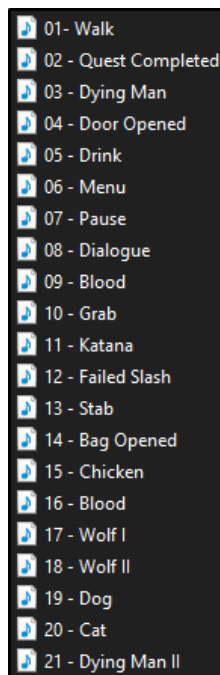


Ilustración 67. Efectos sonoros

Los títulos de los efectos son bastante auto explicativos por lo que no entraré en detalles.

• Música

También añadí tres pistas de audio de temática japonesa feudal, ya que la música es uno de los elementos más importantes de un RPG:

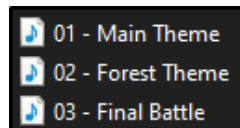


Ilustración 68. Música del juego

Las tres pistas tratan de representar tres estados, la primera la tranquilidad de la aldea, donde no hay ningún peligro y el jugador puede sentirse a salvo, la segunda es más cañera, se reproduce al entrar en el bosque y hace que el jugador entre en alerta y la última tiene un toque de epicidad, pues se reproduce al llegar a la zona donde se encuentra el jefe final.

• Fuente

Otro elemento importante es el tipo de fuente de los textos, por lo que utilicé una que replica la fuente utilizada en *Final Fantasy VII* llamada **Reactor7**:

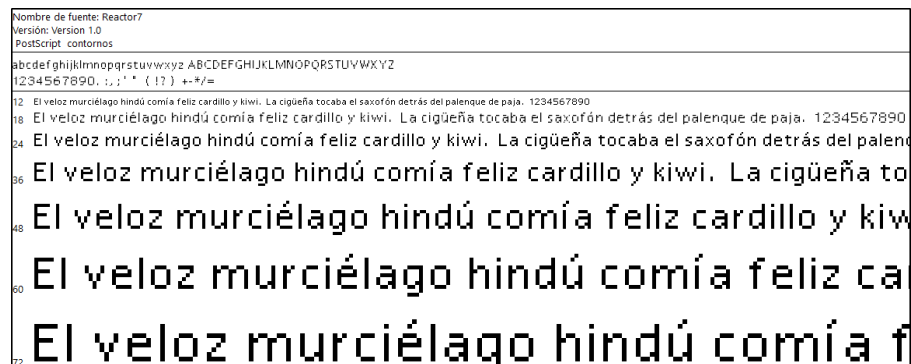


Ilustración 69. Fuente

• Créditos

Los recursos utilizados que han sido sacados de internet son de libre uso en su totalidad, sin derechos de autor o, si los tiene, utilizables siempre que no sea para uso comercial, como es mi caso. A continuación, adjunto un enlace con el origen de cada uno de los recursos:

Tilesets

- 01 - [Tileset I](#)
- 02 - [Japanese houses](#)
- 03 - [Tileset II](#)
- 04 - [House Interior I](#)
- 05 - [House Interior II](#)
- 06 - [House Interior III](#)
- 07 - [Japanese Lamp](#)
- 08 - [Tomb Stones](#)

Animales

- 01 - [Animals](#)

Sangre

- 01 - [Blood](#)

Katanas

- 01 - [Katanas](#)

Items

- 01 - [Items](#)

NPCs y enemigos

- 01 - [Characters](#)

Efectos de sonido

Todos sacados de freesound.org

Música

- 01 - [Main Theme](#)
- 02 - [Forest Theme](#)
- 03 - [Final battle](#)

Fuente

- 01 - [Reactor7](#)

Menú de inventario y cuadro de diálogo

Hechos por mí

4.5 Esquema de la arquitectura del juego

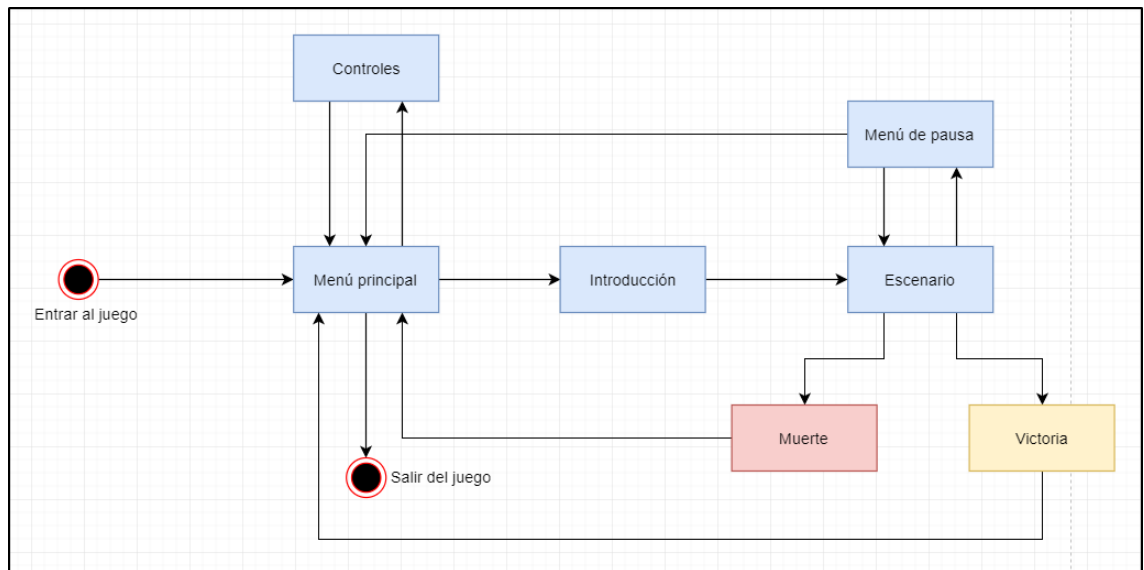


Ilustración 70. Arquitectura del juego

4.6 IA de los enemigos

Existen dos tipos de enemigos en el juego, que son los samuráis y los lobos, por lo que, aunque comparten el mismo *script* de movimiento, la manera de atacar es diferente, ya que los samuráis requieren de una animación para atacar y los lobos no, pues solo con tocar al personaje con el morro ya es suficiente para hacerle daño (simulando un mordisco).

Por lo tanto, para la explicación de la IA de los enemigos los voy a clasificar en esas dos categorías:

• Samuráis

Lo primero que hice fue crear las animaciones de cada uno de los samuráis enemigos, repartidas en tres estados: *Idle*, *Walking* y *Attacking*:

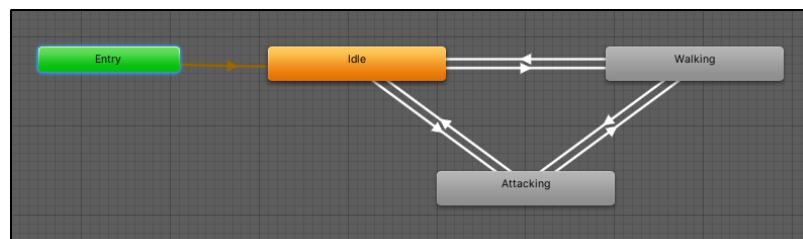


Ilustración 71. Estados de animación de samuráis

La lógica de la IA es simple, el samurái enemigo permanece quieto en un lugar predeterminado (estado *Idle*), al meternos dentro de su radio de acción se activa un diálogo con el enemigo, al finalizar el diálogo el enemigo comienza a andar hacia nosotros (estado *Walking*) y, una vez el enemigo se

encuentra lo suficientemente cerca de nuestro personaje es cuando se activa la animación de ataque (estado *Attacking*).

Así pues, la IA se basa en la implementación de una serie de métodos de *Unity* (como el que hace que se mueva constantemente hacia las coordenadas del jugador, simulando una persecución) y de unos cuantos *if* anidados que permiten que la animación de ataque se active y desactive dependiendo de lo cerca que se encuentre el enemigo del jugador.

También, las animaciones del estado *Attacking* tienen la peculiaridad de que cuando el enemigo da el espadazo se activa un *collider*, que es lo que produce que, cuando ese *collider* colisiona con nuestro jugador, el *script* de vida del jugador recoja el evento y le reduzca la vida. Siendo igual en sentido inverso, cuando nuestro personaje ataca al enemigo.

• Lobos

La IA de los lobos es ligeramente diferente, pues no requieren de animación para producir el ataque, por lo que solo cuentan con dos estados, que son *Idle* y *Walking*:



Ilustración 72. Estados de animación de lobos

Otra de las diferencias radica en el *collider*, pues se encuentra situado en la zona de la cabeza, estando siempre activo, por lo que cuando la cabeza hace contacto con el jugador es cuando se produce el daño.

A diferencia de los samuráis, los lobos se encuentran en constante movimiento, siempre dentro de una zona restringida de la que no pueden salir. Lo que no cambia es el sistema de combate, cuando el personaje entre en el radio de acción del lobo, éste comenzará a perseguirlo.

Otra de las características de la IA, compartida por ambos tipos de enemigos, es que se utiliza una zona restringida por la que pueden moverse, por lo que hay una condición que comprueba constantemente que los enemigos no salgan de dicha zona y, si lo hacen, les hace dar la vuelta para volver dentro, evitando de esta forma que vayan a zonas del mapa no deseadas.

5. Diseño de niveles

5.1 Diseño del mapa

Como suele ser típico en los RPG en 2D el transcurso de la historia y misiones se desarrolla a lo largo y ancho de un mapa de mundo abierto, soliendo ser, también, ricos en contenido con zonas bien diferenciadas entre ellas (montañas, bosques, playas, ciudades...). En mi proyecto el mapa cuenta con tres zonas diferenciadas entre sí, que son la aldea e interiores, el bosque y el castillo:



Ilustración 73. Mapa del juego

Para entrar en más detalle explicaré las zonas una por una, pues cada zona tiene unas características propias:

- **Aldea**



Ilustración 74. Aldea

La aldea es donde comienza la historia, en esta zona no hay peligro de ningún tipo, solo está poblada por *NPCs* y animales inofensivos, además de algunas pociones repartidas por la zona. Esta parte del mapa sirve para que el jugador se familiarice con la trama, los controles y se empape de la esencia de Japón, además de poder visitar algunos interiores.

- **Bosque**



Ilustración 75. Bosque 1



Ilustración 76. Bosque 2

El bosque es donde el jugador pasará la mayor parte del tiempo, además de ser donde se desarrollan la mayoría de los combates. Está poblada por samuráis y lobos que el jugador deberá derrotar para poder avanzar. La decoración es bastante sangrienta, hecho adrede para poner en alerta al usuario.

Contiene caminos opcionales, que desembocan en zonas con pociones, aunque custodiadas por diversos lobos, por lo que no será gratis la obtención de los objetos curativos.

- **Castillo**

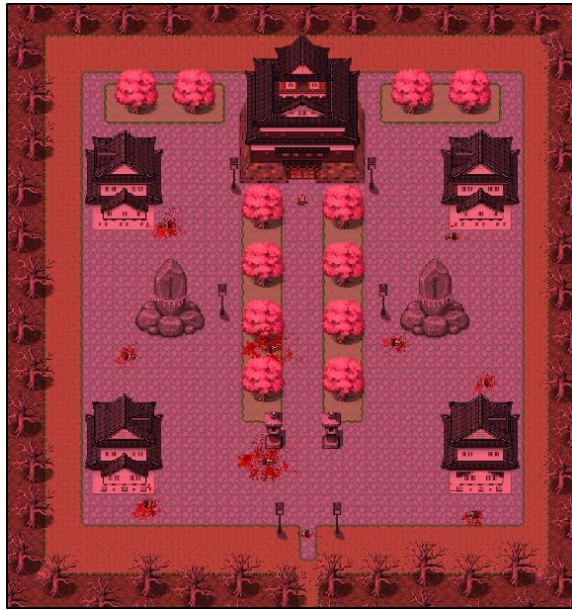


Ilustración 77. Castillo

Esta zona es donde se desarrolla el combate final contra el líder de la banda, decorado también de manera sangrienta, además de la música épica que suena en la zona, harán que el jugador sepa que se encuentra en la parte final del juego.

5.2 Sistema de misiones

El juego cuenta realmente con un solo nivel, subdivido en una serie de misiones, que se suceden una detrás de otra hasta la finalización del juego. Las misiones ya han sido descritas en uno de los apartados anteriores.

5.3 Características de los enemigos

Para hacerlo más rolero y desafiante cada enemigo tiene unas características únicas de daño, vida y velocidad, de manera que conforme avancemos por el bosque más difícil será derrotar a los forajidos. Para compensarlo, nuestro personaje obtendrá la katana del enemigo que derrote, por lo que nosotros también nos iremos haciendo más fuertes por cada enemigo eliminado.

6. Manual de usuario

6.1 Requisitos mínimos

- Sistema operativo: Windows 7
- Procesador: Intel Core i3
- Memoria RAM: 1 GB
- Tarjeta gráfica: Compatible con DX10, DX11 y DX12
- Almacenamiento: 80 MB

6.2 Instrucciones de juego

El juego es bastante auto explicativo, al ejecutarlo entraremos en el menú inicial donde tenemos tres opciones, que son cerrar el juego, ver los controles o iniciar la partida:



Ilustración 78. Pantalla de título

Si pulsamos en “Salir del juego” el juego se cerrará, devolviendo al usuario al escritorio. Si pulsamos en “Controles” veremos la siguiente pantalla:

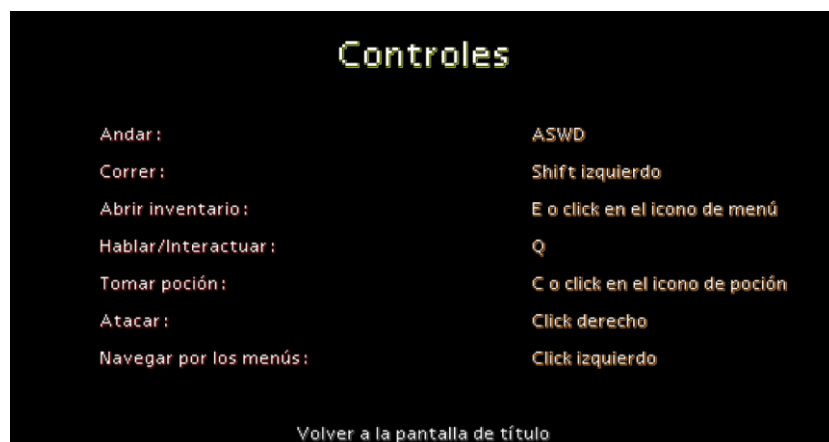


Ilustración 79. Pantalla de controles

En esta pantalla podemos ver los controles del juego, además de poder pulsar en “Volver a la pantalla de título” para hacer lo propio. Una vez en el menú de inicio, si pulsamos en “Iniciar partida” veremos una pequeña introducción de la trama del juego:

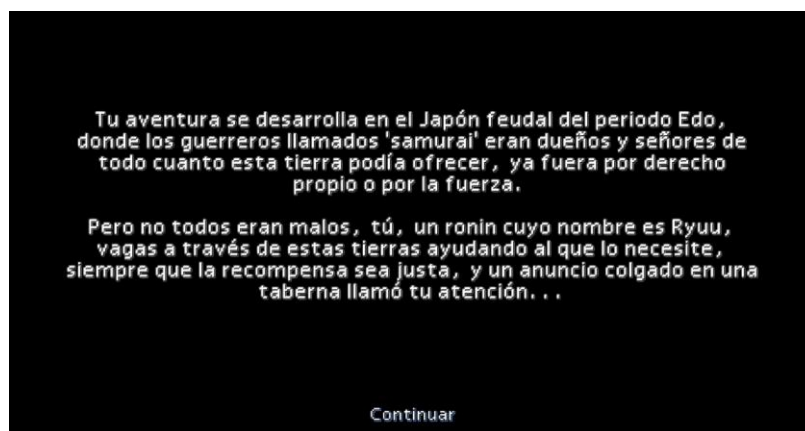


Ilustración 80. Pantalla de trama

Pulsando el botón “Continuar” dará comienzo la partida:

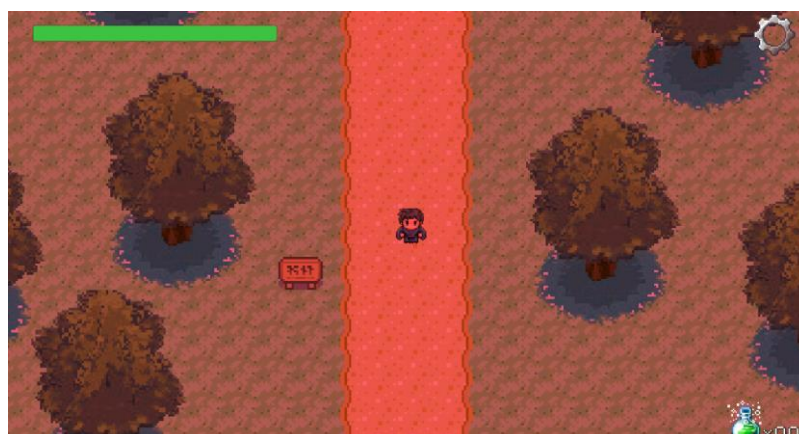


Ilustración 81. Pantalla de juego

Cuando aparezca un diálogo es suficiente con pulsar en la esquina inferior derecha para avanzar:



Ilustración 82. Pantalla de diálogo

El menú de pausa cuenta con diversas opciones, como podemos ver a continuación:



Ilustración 83. Pantalla de inventario

Pulsando en el botón “Inventario” se nos mostrarán las katanas que poseemos en ese momento, pudiendo equiparnos cualquiera de ellas pulsando sobre la katana deseada.

El botón “Estado” muestra información sobre el jugador, además del daño que hace la espada equipada, mientras que el botón “Misiones” nos muestra información sobre la misión que estamos cursando actualmente.

Por último, el botón “Salir del juego” nos devuelve al menú de título inicial, desde el cual podremos cerrar el juego perdiendo, eso sí, todos los progresos que se hayan realizado en el juego.

7. Conclusiones

7.1 Conclusiones del trabajo

Por un lado, en cuanto al juego, estoy bastante contento con el resultado final, ya que ha quedado prácticamente tal y como lo había imaginado en el momento en el que pensé en la temática que tendría mi proyecto. Además, he aprendido técnicas nuevas en cuanto al desarrollo de videojuegos ya que, aunque solo había hecho pequeños proyectos personales, ya tenía algo de experiencia con *Unity*, por lo que lo aprendido durante este semestre me será útil para desarrollos futuros.

Por otro lado, las conclusiones que saco de la realización del trabajo final de grado con la temática de videojuegos son bastante positivas, ya que he tenido total libertad para desarrollar el proyecto, algo que choca bastante con el resto de asignaturas, donde siempre tienes que seguir unas reglas o pautas ya definidas. Puedo decir con total franqueza que he disfrutado de la asignatura, aunque es verdad que me ha requerido bastantes horas, por lo que la fuerza de voluntad y dedicación han sido muy importantes.

7.2 Logro de los objetivos planteados

Considero que he cumplido prácticamente todos los objetivos que me había planteado, incluso he podido realizar mejoras, siguiendo los consejos del consultor, que han potenciado el resultado final, como poner los interiores de las casas en la misma escena que el resto del mapa, evitando periodos de cargas prolongados.

Los objetivos no cumplidos, ya sea por falta de tiempo o conocimiento, diría que han sido los siguientes:

- Mecánicas de combate más variadas
- Más tipos de enemigos
- Más tipos de armas
- Implementación de puzles
- Mayor duración de juego
- Pruebas con usuarios

7.3 Seguimiento de la planificación y metodología

En el diagrama de Gantt plasmé la planificación que tenía en mente para todo el semestre, tanto para el desarrollo del juego como para el resto de entregables de la asignatura (memoria, vídeos, defensa...), y considero que he seguido bastante bien dicha planificación, siendo constante e intentando implementar todos los puntos conforme avanzaban los días para no descuadrar la planificación.

Aun así, no todo ha ido tal y como estaba planeado, pues debido al trabajo y a algunos eventos sociales ha habido días en los que no he podido avanzar el proyecto. Otro de los puntos que no ha ido conforme estaba plasmado en la planificación ha sido la creación de la memoria durante la PEC 3, pues estaba centrado en conseguir una versión prácticamente final del juego para dicha entrega.

En cuanto a la metodología, me ha venido bastante bien implementar el desarrollo en cascada, pues me ha permitido avanzar el proyecto en una serie de etapas bien definidas, empezando por la conceptualización del proyecto y terminando con la versión final completamente desarrollada y funcional.

7.4 Líneas de trabajo futuro

Siempre hay cosas que mejorar, y mi proyecto tiene bastante recorrido todavía, es decir, implementando una serie de funcionalidades extra y mejorando algunas de las mecánicas actuales podría convertirse en una obra de mucha más calidad. Los elementos que implementaría más adelante serían los siguientes, (además de los objetivos no cumplidos mencionados anteriormente):

- Sistema de guardado de partida
- Sistema de *stats* del jugador más profundo
- Elección de respuestas en las conversaciones
- Poder interactuar con más objetos
- Sistema de misiones secundarias
- Lanzamiento del juego en más plataformas

8. Glosario

RPG

Género de videojuego que utiliza los juegos de mesa de rol como mecánicas.

Action-RPG

Género de videojuego que mezcla *RPG* con acción, eliminando los turnos en los combates.

Script

Trozo de código que realiza alguna funcionalidad.

Software

Programa informático.

Hardware

Parte física de un equipo informático.

Assets

Recursos utilizados en un juego o proyecto (gráficos o sonoros).

Power-Up

Personaje que se hace más fuerte por algún motivo.

Stats

Estadísticas del jugador (fuerza, vida, velocidad...).

Edo

Periodo de Japón que comprende los años 1603-1868.

Shogun

Líder militar japonés, título que se concedía en la antigüedad en Japón al gobernador.

Ronin

Samurái sin amo.

NPC

Personaje no utilizable.

Sprite

Mapa de bits que simula gráficamente un personaje o elemento.

Screenshot

Captura de pantalla.

Tile

Tesela que se utiliza para rellenar el “fondo” de un videojuego, por donde andan los personajes (*sprites*).

Tilemap

Imagen que contiene una serie de *tiles*, normalmente relacionados en una temática.

IDE

Entorno de desarrollo integrado, utilizado para la escritura, compilación y ejecución de código.

Gameplay

Parte jugable de un videojuego.

IA

Inteligencia artificial, asignada normalmente a personajes no jugables.

Collider

En *Unity*, un *collider* es un elemento colisionador que permite que otro *collider* no pueda atravesarlo, o que se accione un disparador al entrar en él.

9. Bibliografía

Unity – [Página web](#)

Unity – [Documentación](#)

Unity – [Requisitos mínimos](#)

Visual Studio 2019 – [Página web](#)

Audacity – [Página web](#)

Adobe Photoshop CC – [Página web](#)

DaVinci Resolve – [Página web](#)

The legend of Zelda – [Wikipedia](#)

Final Fantasy – [Wikipedia](#)

Alundra – [Wikipedia](#)

Rockstar Games – [Wikipedia](#)

CD Projekt Red – [Wikipedia](#)

Square Enix – [Wikipedia](#)

Unreal Engine – [Wikipedia](#)

GameMaker Studio – [Wikipedia](#)

RPG Maker – [Wikipedia](#)

Museo de Tokio – [Wikipedia](#)

10. Anexos

Repositorio de GitHub - [Enlace](#)

Vídeo PEC 2 - [Enlace](#)

Vídeo PEC 3 - [Enlace](#)

Unity Showcase - [Enlace](#)

Tráiler - [Enlace](#)

Defensa - [Enlace](#)