



Universidad
de La Laguna

Escuela Superior de
Ingeniería y Tecnología
Sección de Ingeniería Informática

Trabajo de Fin de Grado

Videojuego para la promoción de una alimentación saludable

A Food Story

José Carlos Rodríguez Cortés

La Laguna, 5 de Julio de 2016

Dña. **Carina Soledad González González**, con N.I.F. 54.064.251-Z profesor Titular de Universidad adscrito al Departamento de Ingeniería Informática y de Sistemas, como tutora

C E R T I F I C A

Que la presente memoria titulada:

“Videojuego para la promoción de una alimentación saludable”

ha sido realizada bajo su dirección por D. **José Carlos Rodríguez Cortés**, con N.I.F. 42.210.614-W.

Y para que así conste, en cumplimiento de la legislación vigente y a los efectos oportunos firman la presente en La Laguna a 5 de Julio de 2016.

Agradecimientos

A la Universidad de La Laguna por ofrecerme una educación de calidad y el mejor trato posible así como la posibilidad de emprender este proyecto con todo lo que ello conlleva.

A la profesora Carina por su apoyo en la realización de este trabajo.

A Ruth, por estar siempre ahí para lo que fuera.

Licencia



© Esta obra está bajo una licencia de Creative Commons Reconocimiento-NoComercial-CompartirIgual 4.0 Internacional.

Liberación de responsabilidades

For the American territory:

COPYRIGHT FAIR USE NOTICE, Title 17, US Code (Sections 107-118 of the copyright law): All media in this document is used for educational purpose under terms of fair use. All images used belong to their respective companies.



Para el territorio español:

Real Decreto Legislativo 1/1996, de 12 de abril, por el que se aprueba el texto refundido de la Ley de Propiedad Intelectual, regularizando, aclarando y armonizando las disposiciones legales vigentes sobre la materia.

- **Publicado en:** «BOE» núm. 97, de 22 de abril de 1996, páginas 14369 a 14396 (28 págs.)
- **Sección:** I. Disposiciones generales
- **Departamento:** Ministerio de cultura
- **Referencia:** BOE-A-1996-8930

Capítulo II: Límites, Artículo 32. Citas y reseñas:

Es lícita la inclusión en una obra propia de fragmentos de otras ajenas de naturaleza escrita, sonora o audiovisual, así como la de obras aisladas de carácter plástico, fotográfico figurativo o análogo, siempre que se trate de obras ya divulgadas y su inclusión se realice a título de cita o para su análisis, comentario o juicio crítico. Tal utilización sólo podrá realizarse con fines docentes o de investigación, en la medida justificada por el fin de esa incorporación e indicando la fuente y el nombre del autor de la obra utilizada.

Las recopilaciones periódicas efectuadas en forma de reseñas o revistas de prensa tendrán la consideración de citas.

En cuanto a las citas me ceñiré a la norma **ISO 690**.

Resumen

El objetivo de este trabajo ha sido realizar un videojuego que se ejecutará principalmente en PC, pero podrá ser extendido a otras plataformas, que trate sobre la temática de la salud. Se ha optado por tratar el problema de la mala alimentación y la obesidad infantil mediante los videojuegos para crear conciencia sobre la alimentación saludable en la población infantil.

La motivación de desarrollar este proyecto en forma de videojuego es la de intentar llegar de forma más directa al público infantil, ya que el ámbito lúdico siempre ha sido una buena forma de transmitir conocimiento a los más pequeños.

Para el desarrollo de este videojuego hemos optado por utilizar el motor gráfico Unity 3D[®] debido a la flexibilidad y potencia que ofrece así como a su agradable curva de aprendizaje. Esta herramienta fue seleccionada también debido a que ya se poseía experiencia en su manejo.

Para terminar cabe señalar las aportaciones que se realizan al mundo de los juegos orientados a la salud. Se ha optado por centrarse en la región de la salud dedicada a los hábitos alimenticios, creando para ello un juego que trata de educar a los niños en hábitos de vida saludable. El principal aporte que se realiza a este campo es el de crear un juego del género de plataformas cuando la gran mayoría de los juegos de este tipo se centran en aventuras conversacionales o juegos de gestión.

Palabras clave: Videojuego, PC, Plataformas, Salud, Hábitos alimenticios.

Abstract

The main goal of this project is to make a videogame that could be played in PC and, if the time permits, in another platforms, like smartphones and tablets, or in web browsers maybe.

The main theme of the videogame that we want to make is the healthy life and healthier habits. A videogame that we can use for ludic and educational purpose. Both of them including in a game which main theme is the alimentary healthy habits or rather the unhealthier habits that we need to avoid. Both of them including in a game which main theme is the alimentary healthy habits or rather the unhealthier habits that we need to avoid.

The motivation to develop this project in a videogame form is for trying a more direct way to teach the kids, because the ludic way always have been a nice way to teach kids who have some problems with another ways.

For the development of this game, we chose the option of use the game engine Unity 3D[®] because of its flexibility and potential as well as its nice learning curve. This tool was selected too because we have some experience using it.

To finish this abstract, we need to point the contribution that we provide to the world of the healthy games. We took the option of chose the region of the healthy alimentary habits, creating for this a game which main objective is to teach the kids healthy life habits. The main contribution that we make to this section is the creation of a platform game because the vast majority of the games of this type are conversational adventures or Management Games and it's a bit difficult to find a platform healthy game.

Keywords: Videogame, PC, Platforms, Healthy life, Healthier habits.

Índice General

Capítulo 1. Introducción	5
1.1 Idea principal.....	5
1.2 Creación de videojuegos: Fundamentos	5
1.3 Metodología	7
1.4 Hitos y alcance.....	8
Capítulo 2. Antecedentes	9
2.1 Estudio de otros videojuegos	9
2.2 Soluciones propuestas a este problema.....	10
Capítulo 3. Objetivos, Fases y Desarrollo	11
3.1 Objetivos.....	11
3.2 Fases	11
3.3 Desarrollo.....	12
3.4 Problemas y soluciones	14
Capítulo 4. Proyecto: A Food Story	16
4.1 Personajes.....	16
4.1.1 Personajes principales	16
4.1.2 Personajes secundarios	17
4.1.3 Enemigos	17
4.2 Historia.....	21
4.3 Mecánicas.....	22
4.4 Apartado Gráfico	24
4.5 Aspectos Técnicos	25
4.6 Errores conocidos	28
Capítulo 5. Conclusiones y líneas futuras	29
5.1 Lo que se ha hecho.....	29
5.2 Líneas Futuras	30
Capítulo 6. Summary and Conclusions	31
6.1 Things we have made	31
6.2 The future of the project	31

Capítulo 7. Presupuesto	33
--------------------------------	-----------

Bibliografía	34
---------------------	-----------

Índice de figuras

Ilustración 1: Ori and the Blind Forest (Versión desconocida, 2015). Moon Studios.	6
Ilustración 2: Pong (Versión desconocida, 1975). Atari.	6
Ilustración 3: Megaman (Versión desconocida, 1987). Capcom.	13
Ilustración 4: Robi (Personaje Principal).	16
Ilustración 5: Princesa Ruth. (Personaje secundario).	17
Ilustración 6: Enemigo Caminante.	18
Ilustración 7: Enemigo Saltador.	18
Ilustración 8: Enemigo Seguidor.	19
Ilustración 9: Array Circular.	19
Ilustración 10: Enemigo Estacionario.	19
Ilustración 11: General “Sin Brazos”.	20
Ilustración 12: General “Sin Brazos” disparando un proyectil.	20
Ilustración 13: Chef Robot [13].	21
Ilustración 14: Coleccionable Naranja.	23
Ilustración 15: Tiles de mapa.	24
Ilustración 16: Frames del enemigo caminante.	24
Ilustración 17: Partes del cuerpo del General “Sin Brazos” *.	25

Ilustración 18: Código del movimiento del personaje.	25
Ilustración 19: Código del salto.	26
Ilustración 20: Código para la situación de “Escalera”.	26
Ilustración 21: Código para generar un disparo.	26
Ilustración 22: Código para la recolección de coleccionables.	27
Ilustración 23: Singleton que representa la información del juego.	27

Índice de tablas

Tabla 7.1. Tabla de presupuestos.....33

Capítulo 1.

Introducción

En este capítulo veremos una pequeña introducción con la idea principal del proyecto, la problemática inicial que surge cuando comenzamos a realizar un proyecto de estas características, una idea general del desarrollo concebida en un inicio, es decir, sin conocimiento de lo que implica realmente el desarrollo de un videojuego y, finalmente, los hitos y alcance que se plantean para el proyecto.

1.1 Idea principal

Como ya se ha introducido en el resumen la idea principal de este TFG es la de hacer un videojuego que trate sobre algún tema relacionado con la salud. Más concretamente se ha optado por la opción de la temática relacionada con los hábitos de alimentación, y las enfermedades relacionadas con los malos hábitos alimenticios.

De esta forma vamos a tratar temas como la obesidad, la diabetes o el exceso de colesterol en sangre desde un punto de vista lúdico pero que no perderá la esencia educativa.

1.2 Creación de videojuegos: Fundamentos

El desarrollo de un videojuego, al ser algo completamente distinto a lo estudiado en la universidad, contempla una serie de problemáticas que no se habían planteado en un principio. Algunos de estos problemas son los siguientes:

- **Narrativa:** Cualquier videojuego que se precie tiene algún componente narrativo por lo tanto surge el primer problema. ¿Cómo se desarrolla la narrativa en un videojuego? ¿Qué tipos de narrativa existen?

Diseño gráfico y arte: Si pensamos en algún videojuego nos vienen a la mente imágenes en movimiento o por lo menos algún tipo de diseño que requiere de buena mano en cuanto a diseño gráfico se refiere (obviando claro está, algunos juegos antiguos que se basaban en texto o que trabajaban solamente con algunos píxeles coloreados). P.Ej.: No podemos comparar la carga gráfica de un juego como Ori and the Blind Forest® con otros como por ejemplo el Pong®.



Ilustración 1: Ori and the Blind Forest (Versión desconocida, 2015). Moon Studios.

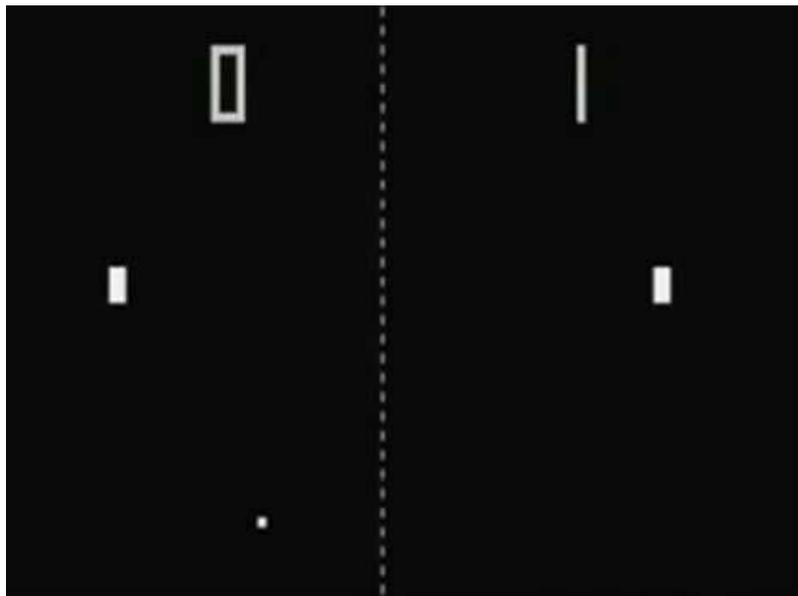


Ilustración 2: Pong (Versión desconocida, 1975). Atari.

- **Programación con nuevos enfoques:** Aunque un videojuego se trate de un producto de software más lo cierto es que la programación ligada a su desarrollo difiere enormemente de la aplicada a cualquier otro programa. Más concretamente, para este proyecto, se tendrá en cuenta aspectos físicos como la gravedad, el rozamiento con el suelo, las fuerzas aplicadas a objetos tangibles, etc.
- **Diseño del videojuego:** El diseño del propio videojuego en sí supone una problemática a tener en cuenta ya que se trata de un

tema harto extenso, lo suficiente como para hacer un trabajo solamente hablando de cómo diseñar un videojuego.

- **Otros problemas:** Algunos problemas menores pero no por ello menos importantes son aquellos relacionados con la puesta en escena del videojuego, es decir, temas de iluminación, planos de grabación, etc. y otros como, por ejemplo, las características propias del uso de un motor de videojuegos (P.E.: Unity 3D[®] o Unreal Engine[®]).

1.3 Metodología

El proyecto se dividirá en varios apartados, a saber:

- **Estudio de precedentes y proyectos similares:** Se hará un estudio de las tendencias de desarrollo en cuanto a los videojuegos en general y se centrará la atención en los juegos dedicados a la temática saludable.
- **Concepción y diseño del videojuego:** Una vez realizado el estudio de otros proyectos similares se procederá a la concepción de un primer “boceto” de videojuego y el posterior diseño del mismo, (mecánicas, personajes, historia, ambientación, etc.). Para más información acerca del diseño del videojuego diríjase a [\[1\]](#).
- **Historia:** Una vez diseñado el juego se procederá a escribir una historia adecuada, así como a seleccionar el tipo de narrativa a emplear.
- **Estilo artístico:** En esta fase se seleccionará el estilo artístico que tendrá el juego.
- **Estructuración de la programación:** Se realizará un estudio de que estructuras se van a emplear para la programación del videojuego (clases, atributos, patrones de diseño, etc.).
- **Programación:** El apartado de la programación se dividirá en tres sub-apartados:
 - **Programación de la física:** Se implementarán todas las interacciones físicas que se darán durante la ejecución del juego.
 - **Programación gráfica:** Se implementarán todas las interacciones gráficas que se ejecutarán en pantalla.
 - **Programación de la lógica del juego y de los personajes:** En este apartado se procederá a la programación de las mecánicas jugables así como de las inteligencias artificiales de

los enemigos y la lógica ligada al comportamiento del escenario.

- **Documentación:** En paralelo a todo lo anteriormente mencionado se realizará la debida documentación para un proyecto de videojuegos. Esto es, documentos tales como: GDD, EDT, etc. Nos centraremos sobre todo en desarrollar un buen GDD para nuestro juego.

1.4 Hitos y alcance

El desarrollo de este juego se efectuará intentando alcanzar los siguientes hitos:

- Concepción inicial del juego.
- Diseño del juego y la jugabilidad.
- Diseño de las mecánicas y físicas.
- Diseño de personajes.
- Diseño de la historia.
- Diseño de niveles.
- Escritura del GDD.
- Programación de las bases físicas del juego.
- Programación de las mecánicas del juego.
- Programación de los “prefabs” necesarios para construir los niveles.
- Programación de eventos de juego.
- Animación de personajes y creación de cinemáticas.

En cuanto al alcance de este proyecto, podemos definir que se pretende conseguir una “demo” jugable para utilizar como muestra del proyecto que dure al menos cinco minutos y que presente la idea principal del juego así como una introducción a la historia que plantea.

Capítulo 2.

Antecedentes

En este capítulo hablaremos de los antecedentes que se estudiaron para la consecución de este proyecto, la problemática actual de los juegos orientados a la salud y también cómo se plantea solucionar esta problemática mediante este proyecto.

2.1 Estudio de otros videojuegos

Como primer paso para este proyecto se estudiarán proyectos similares al que se pretende realizar. Tras un estudio superficial de la tendencia general de este tipo de juegos nos encontramos con una serie de patrones que todos, o casi todos, los juegos orientados a la salud siguen:

- Juegos basados en ver un video o leer un texto y responder una serie de preguntas.
- Juegos basados en texto en el que “tomamos decisiones” (y entrecomillo porque realmente no afectan para nada a la historia del juego, son decisiones vacías y sin ningún fin) a través de una historia.
- Juegos del género de gestión, es decir, juegos en los que poseemos un negocio (cafetería, restaurante, hospital, clínica, etc.) y tenemos que realizar una serie de acciones para llevar a buen puerto a nuestro negocio, en este caso claro está, intercalando las acciones de gestión con acciones saludables y recomendaciones acerca de higiene, salud bucodental, hábitos alimenticios, etc.

En general podemos afirmar que, algunos más que otros, todos son en general bastante aburridos ya que, “Básicamente, es divertido ejercitar el cerebro” y “Diversión es solo otra palabra para aprendizaje” [\[2\]](#) como diría Koster y estos juegos no plantean ningún reto para el jugador.

Y como se puede comprobar después de probar muchos de estos juegos durante horas, ninguno supone un gran reto y por tanto se hacen tediosos y aburridos, con la salvedad de algunos juegos de gestión muy bien llevados que ofrecían una jugabilidad entretenida y que ofrecía diversión durante horas.

Al darnos cuenta de esto podemos apreciar el problema directo que nos surge: ¿Para qué hacer un juego que enseñe hábitos alimentarios saludables si nadie lo va a jugar o va a jugar dos minutos y lo va a desechar

inmediatamente?, está claro que si no podemos atraer la atención de un usuario durante más de cinco minutos no vamos a poder enseñarle nada.

De esta forma se estableció un requisito más para la culminación del proyecto, esto es, crear un videojuego orientado a la salud que transmita enseñanzas de hábitos de vida saludables y que al mismo tiempo resulte divertido y que innove algo en el campo de los juegos de salud que tan encasillado está en los juegos narrativos y de gestión.

2.2 Soluciones propuestas a este problema

Para solucionar este problema se decidió orientar el proyecto a un juego del género de las plataformas por ser un género dinámico y que proporciona ciertas facilidades en cuanto a su diseño e implementación. Además, se planteó la posibilidad de expresar el apartado saludable mediante la narrativa y las mecánicas de juego.

Capítulo 3.

Objetivos, Fases y Desarrollo

A continuación presentaremos los objetivos que se plantearon para el proyecto, las fases que se plantearon para alcanzar estos objetivos y el desarrollo de cada una de estas fases, incluyendo en este último las características a destacar de cada fase así como los problemas que surgieron al intentar completarlas.

3.1 Objetivos

Los objetivos a cumplir en este proyecto serán los siguientes:

- Crear un GDD apropiado para el juego que se desea desarrollar.
- Crear un videojuego orientado a la salud que sea funcional y “jugable”.
- Desarrollar una historia que vaya acorde con el juego y disminuir lo máximo posible la llamada “disonancia ludo-narrativa” aunque en un juego con estas características sea bastante difícil conseguir esto.
- Programar todas las mecánicas de juego tal que sean cómodas, dinámicas y sin fallos.
- Desarrollar una demostración del juego que sirva para mostrar las mecánicas del juego así como los aspectos fundamentales de la historia.

3.2 Fases

Las fases necesarias para conseguir los objetivos planteados anteriormente serán las siguientes:

- Recopilación de información acerca de la situación del mercado de los juegos orientados a la salud.
- Recopilación de información sobre “¿Cómo diseñar y desarrollar un juego?”
- Planteamiento inicial del videojuego.
- Diseño del videojuego, personajes, mecánicas e historia.
- Programación de mecánicas, física y eventos del juego.
- Corrección de errores.

3.3 Desarrollo

A continuación se procederá a especificar como se desarrollaron las fases anteriormente nombradas y más adelante hablaremos de los problemas encontrados en cada una de estas fases así como de las soluciones propuestas para cada uno de estos problemas.

Para comenzar vamos a hablar de una cuestión que surge al comenzar con este proyecto ¿Por qué un videojuego?

La respuesta a esta pregunta podemos verla reflejada en el artículo “5 Razones para aprender jugando” [\[10\]](#). Y estas razones son:

- **Los juegos funcionan para enseñar:** Las investigaciones demuestran de forma inequívoca los beneficios del aprendizaje basado en el juego.
- **Los juegos enganchan (hablando de forma positiva):** Un buen juego es capaz de mantenernos pegados a él durante bastante tiempo, nos “engancha”, esto, a nivel educativo, es bastante positivo ya que así, el jugador aprenderá más, debido a que le dedica más tiempo.
- **En los juegos podemos fallar:** La mecánica principal de los juegos es la de “Intento algo nuevo, fracaso, aprendo, lo vuelvo a intentar...” y así se genera un bucle de fracaso y aprendizaje que sirve al alumno para experimentar y tomar decisiones, lo que aporta un refuerzo constante del contenido en el alumno.
- **Los juegos tienen sus raíces en la teoría educativa:** Esto es así, todos los juegos parten de las teorías pedagógicas del “Constructivismo” [\[11\]](#) y de la “Teoría de Aprendizaje Situacional” (o Aprendizaje Situado) [\[12\]](#).
- Como se nos explica de forma más detallada en el artículo antes mencionado, **los juegos aprovechan los procesos afectivos y cognitivos**, es decir, los juegos utilizan las emociones del jugador para intentar transmitirle algo o simplemente para hacerle más interesante el aprendizaje.

Para la realización del proyecto nos hemos planteado utilizar la herramienta Unity 3D[®]. Esto es debido principalmente a la facilidad de uso y potencia que proporciona Unity 3D[®]. Unity 3D[®] es una herramienta muy flexible que proporciona facilidades a la hora de trabajar con temas relacionados con la iluminación, los sprites 2D, posicionamiento en el espacio, transformaciones, fórmulas matemáticas para trabajar con vectores, animaciones y un largo etcétera. Debido a todo esto y a la experiencia previa que se poseía en la herramienta antes de comenzar el proyecto se ha seleccionado Unity 3D[®] para la confección de este juego.

A continuación vamos a hablar del desarrollo de las fases que componen este proyecto.

En cuanto al proceso de recopilación de información acerca del mercado de los juegos orientados a la salud. Para esta tarea se utilizó principalmente el recurso de los buscadores de internet. Muchas de las páginas visitadas estaban orientadas casi exclusivamente a los videojuegos orientados a la salud y a las guías de orientación para las personas que padecen la enfermedad que dichas páginas web tratan.

En cuanto a la recopilación de información acerca del diseño de videojuegos propiamente dicho, nos hemos centrado sobre todo en las obras “Game Programming Patterns” [4] y “The Art of Game Design: A Book of Lenses” [7].

En el diseño del juego nos hemos centrado en crear un juego con unas mecánicas atractivas por lo que optamos por diseñar unas mecánicas muy similares a las del juego Megaman[®] (conocido como Rockman[®] en Japón), es decir, mecánicas de salto, movimiento lateral con una cámara ortográfica y lanzamiento de proyectiles de distintas características. También se cogió la idea de conseguir diferentes habilidades del juego antes mencionado. Más adelante se especificarán mejor las mecánicas del juego. Además, estas se pueden consultar también en el GDD [1].



Ilustración 3: Megaman (Versión desconocida, 1987). Capcom.

Toda la programación se realizó en el lenguaje principal que acepta Unity 3D[®], es decir C#. Toda la carga programática se basa principalmente en pequeños (y no tan pequeños) scripts que regulan el comportamiento de todos los “gameobjects” del juego.

Además de esto, la programación fue realizada siguiendo el paradigma orientado a objetos.

3.4 Problemas y soluciones

En un principio no se encontraron mayores problemas ya que se trataba solamente de búsqueda de información acerca de los juegos orientados a la salud y se encontró bastante información relacionada con el tema.

Los problemas se empezaron a plantear en cuanto se comenzó con la búsqueda de información en todo lo relacionado con el diseño de juegos. Algunos de estos problemas fueron:

- Toda, o casi toda, la información se encontraba en inglés. Esto no supone un problema por sí mismo pero plantea una dificultad añadida el hecho de no poder encontrar información de calidad en el idioma español en cuanto al tema de diseño de videojuegos.
- La información acerca de estos temas es bastante dispersa y se centra en áreas de conocimiento muy amplias, entre ellas pudimos comprobar que el diseño de videojuegos se enfocaba desde muchos puntos de vista, algunos de ellos eran:

Videojuegos enfocados desde el punto de vista de la **psicología**. [3]

Videojuegos enfocados desde el punto de vista de un **programador**. [4]

Videojuegos enfocados desde el punto de vista de un **diseñador de juegos clásicos** (juegos de mesa, juguetes, puzzles, etc.). [5]

Videojuegos enfocados desde un punto de vista de un **diseñador gráfico o grafista**. [6]

Videojuegos enfocados desde un punto de vista de un **diseñador de videojuegos “puro”**. [7]

A pesar de esto se pudo recopilar bastante información acerca del diseño de videojuegos como para continuar, centrándonos sobre todo en las obras que hablan del desarrollo de videojuegos enfocados desde el punto de vista de un programador y el diseño de videojuegos “puro” que es aquel que cuenta con la idea de que no estamos centrados en un área en concreto del desarrollo de un juego y nos da una idea general de en qué se basa el diseño de un juego.

En el planteamiento inicial del juego no hubieron muchos problemas ya que se tenía una idea desde la que se partía y era la de hacer un juego que no fuera lo “típico” dentro del mundo de los videojuegos orientados a la salud. Lo más complicado de la cuestión vino de la mano de la selección de un género en el que enfocar el juego. Una vez se decidió que el género sería el de plataformas, se plantearon las mecánicas clásicas para los juegos de este género: correr, saltar, disparar proyectiles, etc.

A la hora de diseñar el juego se plantearon una serie de problemas que no se habían tenido en cuenta. El diseño de juego incluye una serie de

competencias de las que no se posee ninguna habilidad, tales como: música, diseño gráfico, iluminación, escritura (para los guiones), etc.

Debido a lo anteriormente mencionado se tuvo que realizar una segunda investigación para aclarar cada uno de los puntos que se habían visto comprometidos.

A continuación se expondrá como se solucionó cada uno de los puntos:

- **Apartado musical:** En cuanto al apartado musical se estudió un poco de teoría musical y se compuso una pequeña melodía que se repetiría en un bucle continuo durante el juego. Esta melodía se compuso basándose en los Modos Griegos [8], más concretamente se emplearon los Modos:
 - Jónico: Por expresar sentimientos como la alegría y la belleza.
 - Lidio: Por poseer un aire fantástico y soñador.
 - Mixolidio: Por poseer sonidos juveniles y despreocupados.
- Como se puede apreciar, estos modos fueron seleccionados para que el juego sea de agrado de los niños al incluir solamente aquellos Modos de tono alegre y jovial y descartando los Modos Dórico, Frigio, Eólico y Locrio por sus connotaciones melancólicas y tristes. Las melodías se compusieron a base de unos pocos acordes en piano y algunos arpeggios sencillos de violín. También se incluyó una línea de bajo para darle “cuerpo” a la melodía y un apartado de percusión muy sencillo.
- **Apartado gráfico:** Aunque en un principio se planteó la posibilidad de trabajar en conjunto con un diseñador gráfico. Esto resultó imposible por lo que el apartado gráfico tuvo que solucionarse gracias a diseños gratuitos y sin licencia extraídos de la web “spriters-resource” [9].
- **Historia:** Este problema va intrínsecamente ligado con los problemas en cuanto al apartado gráfico ya que, se debía adaptar la historia a los diseños gratuitos que se habían obtenido de internet. Un ejemplo de este problema viene de la mano de los protagonistas. En un principio los protagonistas principales iban a ser dos hermanos, un niño y una niña; no obstante, al carecer de un diseñador gráfico que pudiera suplir estas necesidades y en vista de que en internet no se podían conseguir diseños de calidad con las características requeridas para la historia que se había construido, la historia tuvo que ser reescrita para adaptarse a la representación gráfica de los personajes y el mundo.

En cuanto a la programación, surgieron los problemas habituales cuando se trabajan con herramientas nuevas: período de aprendizaje y adaptación al motor gráfico Unity 3D[®]. Comprensión del funcionamiento de las físicas del motor así como de las animaciones, iluminación y demás elementos gráficos característicos del motor que no encontramos en la programación de aplicaciones “normales”.

Capítulo 4.

Proyecto: A Food Story

En este capítulo vamos a ver más cosas relacionadas con el diseño del juego en sí, así como el estado actual del mismo.

4.1 Personajes

En nuestro juego llamado “A Food Story” nos encontramos con una serie de personajes que influirán en el desarrollo del mismo.

Vamos a estudiarlos según su impacto en el juego.

4.1.1 Personajes principales

Dentro de esta sección podemos encasillar al personaje principal, de nombre Robi.

Robi es un robot venido de un planeta lejano. Su nave espacial se averió en mitad de un viaje rutinario y calló en el planeta donde se desarrolla el juego.



Ilustración 4: Robi (Personaje Principal).

Una característica de Robi es que necesita comer sano para mantenerse funcional ya que su mecanismo principal funciona mediante hidratos de carbono, por lo tanto deberemos alimentar a Robi a lo largo de todo el juego pero eso sí, deberemos tener cuidado con la ingesta de hidratos de Robi ya que un exceso (o defecto) de los mismos supondrá perder el juego.

4.1.2 Personajes secundarios

Como único personaje secundario de este juego nos encontramos con la “Princesa Ruth”, la princesa del reino “Saludable”. Lleva gobernando el reino de sus padres desde que estos desaparecieron hace unas semanas. Es la encargada de hacer que el reino “Saludable” coma de forma equilibrada en ausencia de sus padres.



Ilustración 5: Princesa Ruth. (Personaje secundario).

La princesa Ruth es expulsada de su palacio por el Chef Robot el cual planea instaurar en reinado de terror haciendo que toda la gente del reino “Saludable” coma comida basura.

La Princesa Ruth acompañará a Robi en sus viajes por el reino “Saludable” en busca de restaurar la paz con la que el Chef Robot amenaza con acabar.

4.1.3 Enemigos

En cuanto a los enemigos tendremos un plantel un tanto más grande ya que aquí se incluirán los enemigos pertenecientes al ejercito del Chef Robot además del propio Chef.

A continuación vamos a analizar cada uno de los enemigos.

I.4.1.3.1 Enemigo Caminante

Se trata de un enemigo básico que deambula de un lado a otro. Su comportamiento es muy sencillo: avanza sin detenerse a no ser que choque con algún objeto o le golpeemos con un proyectil.



Ilustración 6: Enemigo Caminante.

I.4.1.3.2 Enemigo Saltador

Se trata de un enemigo que va saltando a todas partes. El comportamiento de este enemigo implica un desafío mayor que el anterior nombrado ya que salta muy rápido y se abalanza sobre nosotros en poco tiempo por lo que deberemos tenerlo controlado en todo momento. El comportamiento de este enemigo es también bastante sencillo, ya que se basará solamente en saltos diagonales que realizará sin parar.

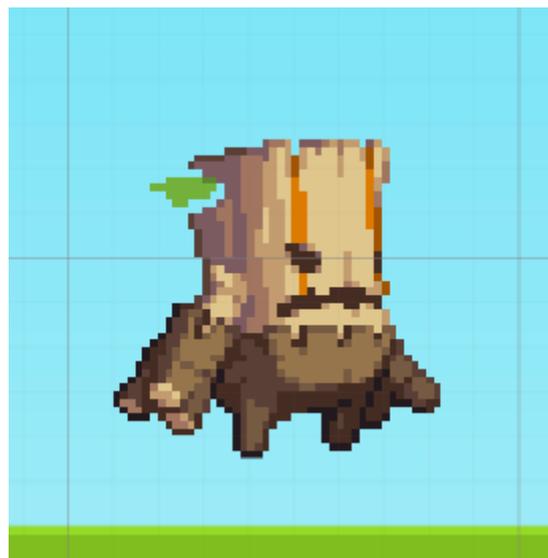


Ilustración 7: Enemigo Saltador.

I.4.1.3.3 Enemigo Seguidor

Este enemigo nos perseguirá allá donde vayamos. Debido a su perseverancia y a su altura, nos será difícil evadirlo saltándolo (aunque podrá ser una opción siempre que las condiciones adecuadas se reúnan) por lo que deberemos acabar con él con nuestros proyectiles.



Ilustración 8: Enemigo Seguidor.

I.4.1.3.4 Enemigo Estacionario

Este enemigo se desplazará de un punto A a un punto B. Una vez llegue al punto B, cambiará de destino y volverá a dirigirse al punto A. En la trayectoria que recorrerá este enemigo pueden definirse todos los puntos que sean necesarios y este los recorrerá uno a uno y cuando llegué al último volverá al principio, es decir, las posiciones estarán dispuestas en forma de un “array circular” y el enemigo se limitará a recorrer este array.

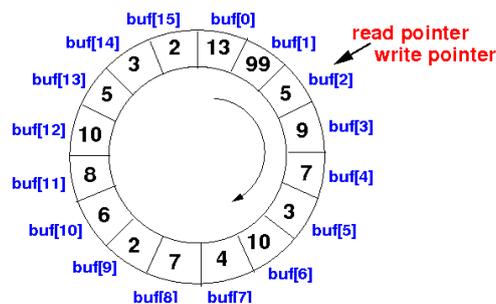


Ilustración 9: Array Circular.

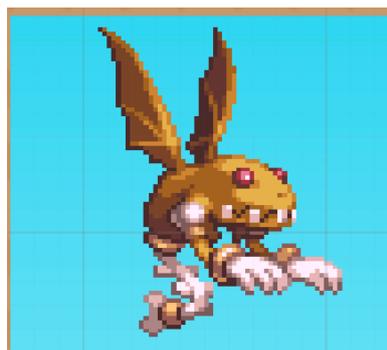


Ilustración 10: Enemigo Estacionario.

I.4.1.3.5 General “Sin Brazos” (Mini Jefe)

Este es el primer “Mini Jefe” que nos encontramos en el juego. Se trata de un robot gigante que perdió los brazos mientras remaba en una barca. Tiene un comportamiento sencillo aunque un tanto más complicado de evitar que el de los enemigos anteriormente mencionados.

Sus patrones de ataque son sencillos: salta cada cierto tiempo y dispara un proyectil cada cierto tiempo también, no obstante, al no estar estos tiempos sincronizados hacemos que el jugador tenga que estar atento en todo momento ya que los proyectiles, que salen siempre del mismo sitio y siempre se dirigen al jugador, se generan a distintas alturas debido a los saltos del General.

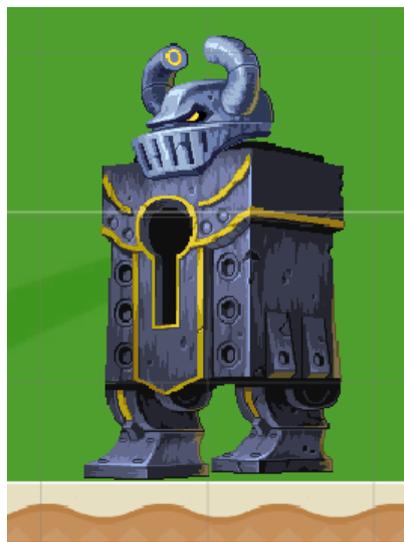


Ilustración 11: General “Sin Brazos”.



Ilustración 12: General “Sin Brazos” disparando un proyectil.

I.4.1.3.6 El Chef Robot

El Chef Robot es el enemigo principal de la historia. Es el chef oficial del palacio real el cual, un día, cansado de pelar cebollas para el cocido real, decide revelarse contra los reyes y la princesa Ruth, secuestrando primero a los reyes para, después de un tiempo, expulsar a Ruth del palacio real.

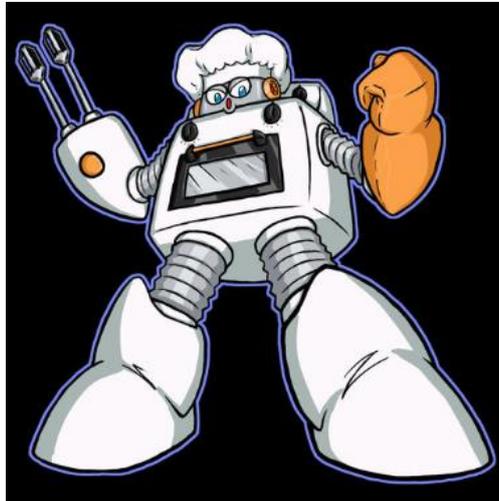


Ilustración 13: Chef Robot [\[13\]](#).

4.2 Historia

La historia de este juego estará centrada en las aventuras de Robi y la Princesa Ruth mientras, los dos, intentan expulsar al Chef Robot para recuperar el trono del reino “Saludable”.

La historia comienza con el Chef Robot, que, mientras pelaba unas cebollas para el cocido real, harto de que los ojos le lloraran por culpa de la cebolla, decide rebelarse contra el reino. Para ello, el Chef Robot primero secuestra a los reyes, dejando a la Princesa Ruth sola en el trono, para más tarde, dar un golpe de estado, derrocando así al último eslabón de la realeza (la Princesa Ruth).

Tras expulsar a la princesa del reino “Saludable”, el Chef Robot se propone acabar con la buena salud de todos los ciudadanos del reino “Saludable” haciéndoles comer solamente comida basura.

Después de varios días, la Princesa Ruth, tras múltiples intentos fallidos de recuperar el palacio, vaga desesperada por el mundo buscando a alguien que pueda ayudarla a expulsar al Chef Robot del reino “Saludable”. Una noche, mientras la Princesa Ruth camina por un prado, esta ve una gran estrella fugaz. Ella le pide un deseo a esta estrella “Ojalá alguien pudiera ayudarme”. En cuanto terminó de pensar esto la estrella fugaz cayó del cielo y se estrelló con un gran estruendo contra la tierra.

La princesa Ruth se acercó entre curiosa y temerosa al lugar del impacto y, cuál no sería su sorpresa al ver a un pequeño robot surgir del lugar del impacto. Este se presentó como Robi y alegó que venía de un planeta muy lejano y que sus reservas de hidratos se encontraban bastante mermadas y, además, le faltaba una pieza para poder arreglar su nave. Una gran vara metálica.

Ruth le ofrece el báculo de la reina a cambio de que la ayude a reconquistar el reino Saludable. A partir de aquí se desarrollan todos los eventos del juego en los que la princesa Ruth se va haciendo cada vez más amiga de Robi y juntos emprenden una aventura para conseguir liberar al reino Saludable del malvado Chef Robot y conseguir que todos en el reino coman saludablemente.

Finalmente, tras derrotar al Chef Robot, este comprende que las cebollas, aunque hacen que los ojos escuezan, son buenas para la salud por lo que acepta volver a trabajar como el chef del reino y promete no volver a intentar ninguna “trastada” más.

Los padres de Ruth, que habían sido secuestrados por el Chef Robot vuelven a ser los gobernantes del reino y en agradecimiento le regalaron a Robi el báculo de la reina para que este pudiera arreglar su nave.

Robi parte con su nave a nuevos planetas con la esperanza de vivir nuevas aventuras y conocer gente tan maravillosa como la que había conocido en aquel planeta.

4.3 Mecánicas

Las mecánicas a implementar en este juego serán las clásicas para un juego de plataformas, incluyendo algunas mecánicas de disparo para acercarnos un poco a las mecánicas del juego Megaman[®].

Las mecánicas serán las siguientes:

- **Caminar:** El personaje podrá caminar de un lado a otro en el eje X (es decir, caminará de forma horizontal). Eso se efectuará aplicando una fuerza en el eje horizontal. Esta mecánica se utilizará para hacer que el personaje se desplace por el mapa.
- **Saltar:** El personaje podrá saltar, tanto vertical como diagonalmente. Esto se conseguirá aplicando fuerzas hacia arriba, para el salto, y hacia los lados, para el movimiento. Esta mecánica se utilizará para salvar obstáculos y saltar huecos.
- **Subir/Bajar escaleras:** El personaje podrá subir y bajar escaleras. Las escaleras funcionarán eliminando la gravedad y aumentando el rozamiento, de esta forma dará la sensación de que el personaje se

agarra a la escalera. Esta mecánica se utilizará para llegar a lugares que son inaccesibles mediante la mecánica **Saltar**.

- **Disparar proyectiles:** El personaje principal podrá disparar proyectiles desde el cañón de su brazo. El proyectil funcionará avanzando en línea recta hasta que se choca con algún objeto “tangible” (que no se pueda atravesar, técnicamente hablando, que tenga un hitbox asociado) o hasta que se alcance la distancia máxima que puede recorrer el proyectil.
- **Cambio de habilidad:** Una mecánica interesante será el cambio de habilidad, esto es, el cambio del tipo de proyectil que podemos utilizar. Cuando el personaje principal vence a un mini-jefe (o mini-boss), situados al final de cada nivel, obtendrá una habilidad nueva y podrá alternar entre todas las habilidades que posea.
- **Recolectar coleccionables:** Esta es una mecánica menor que consistirá en que podemos recoger coleccionables (comida) que nos permitirá aumentar nuestros niveles de azúcar.

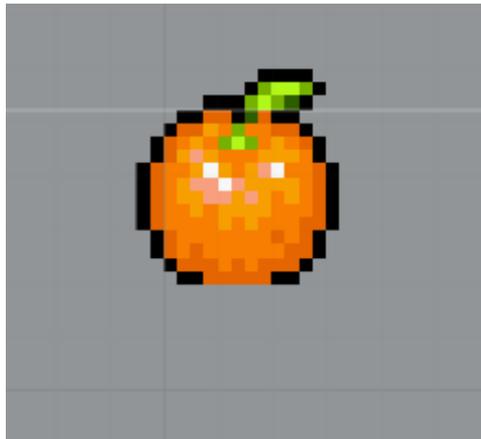


Ilustración 14: Coleccionable Naranja.

4.4 Apartado Gráfico

En cuanto al apartado gráfico cabe destacar que todos los personajes y mapas son diseños 2D basados en sprites.

Los mapas, más concretamente, están basados en tiles, es decir, pequeños cuadrados que, juntos, conforman los mapas.

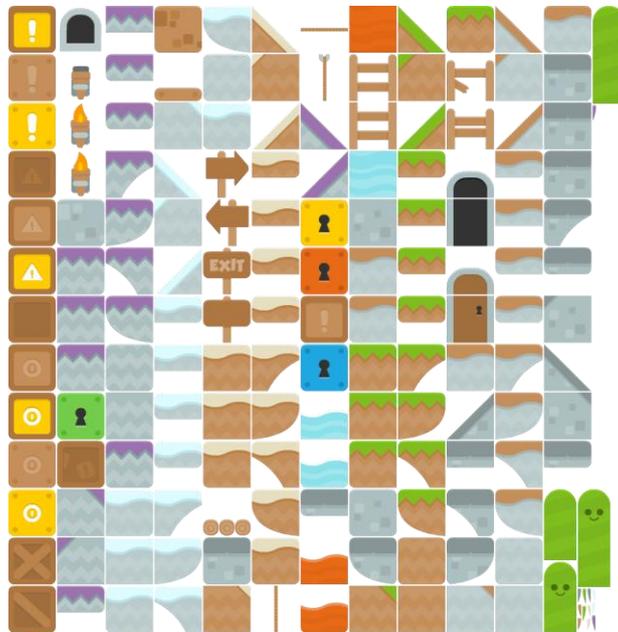


Ilustración 15: Tiles de mapa.

Para la construcción de estos mapas se ha utilizado la aplicación Tiled [\[14\]](#), un programa gratuito que nos permite dibujar mapas basados en tiles de forma sencilla.

Los personajes, por otra parte, pueden ser divididos en dos tipos dependiendo del tipo de animación que hemos utilizado para representarlos.

Los personajes del primer tipo son aquellos cuyas animaciones se basan en frames, es decir, imágenes del mismo personaje en distintas posiciones que, reproducidas una detrás de otra a cierta velocidad, dan una sensación de movimiento.



Ilustración 16: Frames del enemigo caminante.

El segundo tipo de personajes son aquellos cuyas animaciones se basan en curvas de animación y que han tenido que ser animados a mano. Estos personajes se caracterizan por no poseer frames con distintas posiciones sino mostrar un “esqueleto” con las distintas partes del cuerpo del personaje para que nosotros podamos animarlos a placer.

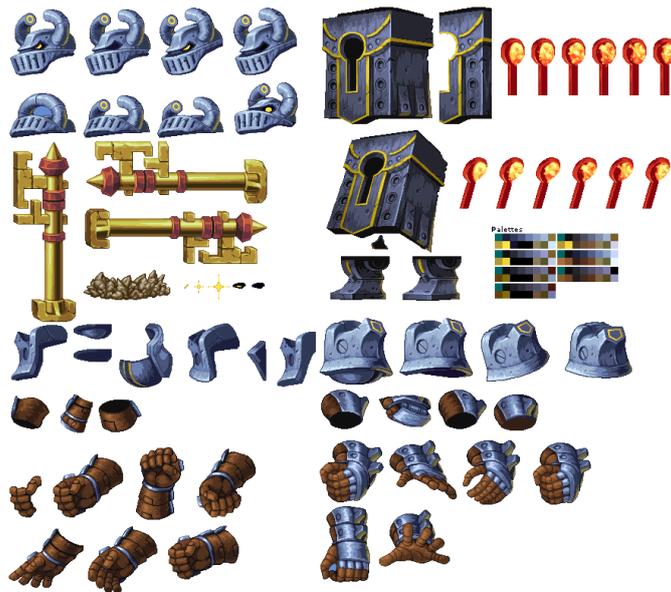


Ilustración 17: Partes del cuerpo del General “Sin Brazos” *.

4.5 Aspectos Técnicos

En esta sección vamos a hablar de todos los aspectos técnicos que tienen que ver con el juego.

Empezaremos hablando de las mecánicas:

- **Caminar:** Como ya se ha comentado antes, los movimientos horizontales se ejecutarán aplicando fuerzas horizontales al personaje principal.

```

/// <summary>
/// Caminar hacia la izquierda
/// </summary>
private void CaminarIzquierda ()
{
    animaciones.SetBool (WALKING, true);
    if (face) {
        face = false;
        RotarObjeto (transform.gameObject);
    }
    body.AddForce (Vector2.left * Velocidad * CONSTANTE_VELOCIDAD);
}

/// <summary>
/// Caminar hacia la derecha
/// </summary>
private void CaminarDerecha ()
{
    animaciones.SetBool (WALKING, true);
    if (!face) {
        face = true;
        RotarObjeto (transform.gameObject);
    }
    body.AddForce (Vector2.right * Velocidad * CONSTANTE_VELOCIDAD);
}

```

Ilustración 18: Código del movimiento del personaje.

- **Saltar:** El salto se implementará también aplicando una fuerza, esta vez en el eje vertical. Para que el salto sea diagonal aprovecharemos las fuerzas aplicadas en el eje horizontal que empleamos para caminar.

```

/// <summary>
/// Saltar
/// </summary>
private void Saltar ()
{
    body.velocity = new Vector2 (body.velocity.x, FuerzaSalto);
}

```

Ilustración 19: Código del salto.

- **Subir/Bajar escaleras:** Para subir y bajar escaleras se elimina la gravedad y se aumenta el rozamiento en una proporción fija para dar la sensación de que el personaje se agarra a la escalera. Una vez ahí, utilizamos las fuerzas aplicadas para movernos para desplazarnos por la escalera. La gravedad y el rozamiento se eliminan cuando el personaje entra en la zona designada como escalera (en el trigger con la etiqueta escalera).

```

void OnTriggerStay2D (Collider2D other)
{
    if (other.tag == "Escalera") {
        body.gravityScale = 0;
        frenteAEscalera = true;
        body.drag = agarre * CONSTANTE_AGARRE;
        SetIDLE ();
    }
}

```

Ilustración 20: Código para la situación de “Escalera”.

- **Disparar proyectiles:** Para disparar proyectiles utilizaremos un objeto invisible como “creador de proyectiles” y lo utilizaremos para posicionar el punto en el que nacerán los proyectiles. En cuanto al evento de disparar vamos a efectuarlo solamente cuando termine la animación de disparar y, por lo tanto, el sistema de animaciones será el que se encargue de generar el proyectil.

```

/// <summary>
/// Disparar
/// </summary>
public void Disparar ()
{
    Transform naceBala = GameObject.Find ("NaceBalaPlayer").transform;
    GameObject balaClone = Instantiate (prefabBala, naceBala.position, Quaternion.identity) as GameObject;
    InformacionBala datosBala = balaClone.GetComponent<InformacionBala> ();
    balaClone.GetComponent<InformacionBala> ().posicionOrigen = naceBala.position;
    Rigidbody2D cuerpoBalaClone = balaClone.GetComponent<Rigidbody2D> ();
    if (PlayerController.face) {
        cuerpoBalaClone.velocity = new Vector2 (datosBala.velocidadBala, cuerpoBalaClone.velocity.y);
    } else {
        cuerpoBalaClone.velocity = new Vector2 (-datosBala.velocidadBala, cuerpoBalaClone.velocity.y);
        PlayerController.RotarObjeto (balaClone);
    }
    balaClone.transform.parent = GameObject.Find (PlayerController.BALAS).transform;
}

```

Ilustración 21: Código para generar un disparo.

- **Recolectar coleccionables:** El sistema de recolección de coleccionables es bastante sencillo y se basa únicamente en hacer desaparecer el coleccionable en cuestión y sumar al personaje (o restar) el azúcar que proporciona. Los coleccionables serán objetos intangibles para que no interfieran con los enemigos ni con los proyectiles.

```
void OnTriggerEnter2D (Collider2D other)
{
    if (other.tag == "Player") {
        PlayerController controlador = GameObject.Find (PlayerController.PLAYER).GetComponent<PlayerController> ();
        controlador.azucar += AzucarProporcionado;
        Destroy (this.gameObject);
    }
}
```

Ilustración 22: Código para la recolección de coleccionables.

En cuanto al azúcar proporcionado (o hidratos de carbono de cada alimento) cabe decir que se emplea una clase que sigue el patrón singleton donde se almacena la información estática del juego y que contiene un “diccionario” que contiene los alimentos y los hidratos de carbono que contiene cada alimento. Los datos de los hidratos de carbono se extraerán de un fichero de texto de extensión “.info”.

```
using UnityEngine;
using System.Collections.Generic;

public class InformacionJuego : MonoBehaviour
{
    private static InformacionJuego instancia = null;

    protected InformacionJuego ()
    {
    }

    public static InformacionJuego Instancia {
        get {
            if (instancia == null) {
                instancia = new InformacionJuego ();
            }
            instancia.Inicializar ();
            return instancia;
        }
    }

    public Dictionary<string, float> HidratosDeCarbono;

    public void Inicializar ()
    {
        string line;
        string[] aux;

        HidratosDeCarbono = new Dictionary<string, float> ();
        System.IO.StreamReader file = new System.IO.StreamReader ("Assets/FicheroHidratos.info");
        while ((line = file.ReadLine ()) != null) {
            aux = line.Split (',');
            HidratosDeCarbono [aux [0].ToLower ()] = float.Parse (aux [1]);
        }

        file.Close ();
    }
}
```

Ilustración 23: Singleton que representa la información del juego.

4.6 Errores conocidos

Se conoce un “bug” por el cual, cuando Robi salta muy cerca de una pared alta y se mantiene el botón de salto, es capaz de seguir subiendo la pared de forma indefinida.

Se sospecha que es debido a un problema con las físicas del motor gráfico o quizás con el sistema de colisiones. Para evitar que el personaje se salga del mapa se ha establecido un “techo” mientras se intenta solucionar este error.

Capítulo 5.

Conclusiones y líneas futuras

En este capítulo vamos a hablar acerca de las conclusiones que hemos podido extraer de este proyecto así como las líneas futuras del mismo, es decir, comentaremos lo que se ha realizado del proyecto, lo que faltaría por desarrollar y, en caso de que el proyecto quisiera retomarse en un futuro, cómo se podría continuar con él.

5.1 Lo que se ha hecho

Hasta el momento se ha conseguido implementar la mayor parte de las mecánicas, aunque quedan pendientes las mecánicas de cambio de habilidades, la demostración implementada es perfectamente jugable.

La historia está desarrollada casi por completo faltando algunos detalles intermedios pero estando clara ya toda su estructura básica.

La música, desarrollada en el programa de edición musical FL Studdio[®], está completa a falta de crear quizás alguna melodía nueva o distinta dependiendo de la situación.

Los diseños están completos, por lo menos en lo que a la demostración se refiere.

El juego presenta ahora mismo dos niveles. El primer nivel sirve de tutorial para introducir las mecánicas de juego básicas así como introducir también la historia que se plantea. De esta forma se crea rápidamente un vínculo con el jugador para que mantenga interés en el juego. El segundo nivel plantea ya un acercamiento más serio a lo que sería el juego en sí, presentando un nivel mucho más largo que el nivel de tutorial, con nuevos enemigos que plantean nuevos retos, secciones de plataformas más complejas y desafiantes que las presentadas en el tutorial y, además, el primer enfrentamiento con un “mini-jefe”, es decir, un enemigo con más resistencia que un enemigo normal y patrón de comportamiento un tanto más complicado que los demás.

5.2 Líneas Futuras

Falta aumentar la cantidad de niveles del juego para crear un juego completo en el que se desarrolle la historia de forma completa así como todas las mecánicas planteadas para el mismo.

Ligado de forma intrínseca a la creación de nuevos niveles viene la creación de nuevas melodías para el juego así como nuevos diseños para personajes y mapas.

En cuanto a la continuación del proyecto, este sería bastante sencillo de continuar ya que se ha construido un “creador de juegos de plataformas” más que un juego en sí. Por lo tanto solamente habría que crear nuevos elementos de juego y aprovechar los que ya están creados.

Además de esto habría que perfeccionar las mecánicas de juego así como optimizar el código para alcanzar un mejor rendimiento.

Capítulo 6.

Summary and Conclusions

In this chapter we'll talk about of the conclusions that we have extracted of this project and the future of the project too.

With more concretion, we'll talk about the things we have made of the project, the things we need to develop and how we can continue with the project in the future.

6.1 Things we have made

At the moment, we achieve to develop the most part of the mechanics of the game, though we need to develop the "change ability" mechanic yet.

The history of the game is written almost completely.

The music, that we have made with FL Studio®, is almost complete too, but we need to compose some new music for the new levels of the game.

The designs are complete too, at least for the "demo".

The game have now two levels. The first of them is a tutorial which introduces the main mechanics and the history. In this way we can construct a link with the player in a direct way in the beginning of the game.

The second level is a most complex level. In this level we can find new challenges because of the new enemies, the new, and more complex, platform sections and, also, the first battle with a "mini-boss", a bigger enemy than the normal ones who has behaviour patterns more complex than the normal enemies.

6.2 The future of the project

We need to develop the rest of the levels of the game to create a full game instead of a demonstration.

Linked to this, we need to create some new melodies for the game and make new designs for the new levels.

The project is quite easy to increment and expand because we made a "platform game maker" instead of a simple game, so, we can make some new levels in an easy way. In case we want to expand the game, we need to

make some new objects for the game and use the objects that we already made.

Capítulo 7.

Presupuesto

En este capítulo se estudiará el presupuesto necesario para producir la demo inicial del juego así como el tiempo invertido por cada profesional que ha intervenido en el proyecto.

Puesto	Precio	Tiempo Invertido
Diseñador gráfico	600€	30 h
Programador	4500€	300 h
Música	1200€	5 h
TOTAL	6300€	335 h

Tabla 7.1. Tabla de presupuestos.

Bibliografía

- [1] GDD: Game Design Document.
- [2] KOSTER, R. (2004). A Theory of Fun for Game Design, Estados Unidos: O'REILLY.
- [3] CSIKSZENTMIHALYI, MIHALY (1990). Flow: The Psychology of Optimal Experience, New York: Harper and Row.
- [4] NYSTROM, ROBERT(2014). Game Programming Patterns, Desconocido: GB.
- [5] BRATHWAITE, BRENDA y SCHREIBER, IAN (2009). Challenges for Game Designers, Estados Unidos: Course Technology (CENGAGE Learning).
- [6] GONZALEZ JIMENEZ, DANIEL (2014). Arte de Videojuegos: Da forma a tus sueños. España: Ra-Ma EDITORIAL.
- [7] SCHELL, JESSE (2008). The Art of Game Design: A Book of Lenses. Estados Unidos: MORGAN KAUFMANN PUBLISHERS.
- [8] Wikipedia. Modos Griegos.
<https://es.wikipedia.org/wiki/Modos_griegos> [Consulta: 12 de Mayo de 2016]
- [9] Página de “sprites” gratuitos y sin licencia.
<<http://www.spritters-resource.com/>> [Consulta: 20 de Junio de 2016]
- [10] Blog La imaginadora.
<<http://www.laimaginadora.com/blog/5-razones-para-aprender-jugando/>> [Consulta: 29 de Junio de 2016]
- [11] Wikipedia. Constructivismo (pedagogía).
<[https://es.wikipedia.org/wiki/Constructivismo_\(pedagog%C3%ADa\)](https://es.wikipedia.org/wiki/Constructivismo_(pedagog%C3%ADa))> [Consulta: 29 de Junio de 2016]
- [12] Wikiterms. Aprendizaje Situado.
<<https://wikiterms.wikispaces.com/Aprendizaje+situado>> [Consulta: 19 de Junio de 2016]
- [13] Página de la que se extrae la imagen del Chef Robot.
<<http://erik-red.deviantart.com/art/Chef-Man-157720613>> [Consulta: 19 de Junio de 2016]
Todo lo relacionado con las ilustraciones del Chef Robot son propiedad intelectual del usuario Erik-Red de la página <http://www.deviantart.com/>

[14] Página de Tiled.
<<http://www.mapeditor.org/>> [Consulta: 23 de Abril de 2016]