

Universidad Carlos III Madrid

Proyecto fin de grado

Grado en Ingeniería Informática



Happy Grow: Videojuego desarrollado con Unity3D para apoyo en la educación infantil frente a problemas alimentarios en personas

Autor: Jorge Lucena Rodríguez

Tutor: Alejandro Calderón Mateos

Agradecimientos

Aunque soy un hombre de pocas palabras para estos temas que no son de mi agrado personal, redactaré las suficientes líneas para aquellos que las merecen.

En primer lugar a mi familia, quienes de forma directa e indirecta han vivido y sobrellevado a lo largo de muchos años conmigo esta experiencia que es la de sobrevivir a una carrera universitaria en la Universidad Carlos III de Madrid.

En segundo lugar a mi novia, quien día a día ha compartido y comparte el final de mi carrera y estará presente en lo que a partir de ahora nos depare el futuro.

En tercer lugar a un profesor que ha destacado en mis estudios, Alejandro Calderón. Esta persona, entre todos los que he vivido compartiendo conocimientos mediante la docencia, se ha ganado un hueco en las personalidades a recordar.

En cuarto lugar a mi tutor, Alejandro Calderón que aunque inicialmente no iba a ser quien me tutorizase el proyecto pero ya se había ganado una mención aquí, ganó la segunda al acogerme y brindarme la ayuda necesaria para acabar mi obra. Aquí también he de añadir a M^aCarmen Fernández que en los inicios del proyecto, aunque fue breve el contacto, me ayudó mucho más de lo que se haya podido imaginar.

En penúltimo lugar, a mis compañeros de carrera Diego, Pablo y César que también han sufrido conmigo esta experiencia y han luchado, realizado peripecias y hecho lo imposible para sobrevivir juntos a esta aventura inolvidable.

Por último pero no por ello menos importante, a todos aquellos creadores de videojuegos, que con su magia me han hecho crecer en miles de mundos donde todo es posible y me han enseñado que siempre merece la pena luchar hasta el final.

Resumen

La industria de los videojuegos es, a hoy de hoy, un mercado que continua en auge en el sector del ocio en España y es consumido por todo tipo de personas a todo tipo de edades. Esto significa que se puede utilizar no sólo como ocio, sino también para poder influenciar de forma positiva en la sociedad motivando ciertos hábitos e inculcando conocimientos al consumidor. El alto tiempo que muchos niños pequeños dedican a los videojuegos no debe ser ignorado, ya que mientras un niño disfruta, también puede formarse como persona en esta sociedad tan tecnológica.

Este proyecto tiene como objetivo el desarrollo de un videojuego 3D en para PC utilizando el motor gráfico Unity3D y con él ayudar a resolver un problema existente en la alimentación de pequeños infantes. El juego se desarrolla en distintos escenarios de un *Tower Defense* en el que los niños mientras deciden y aprenden estrategias, adquieren conocimientos educativos. Durante el juego se les motiva con excelentes hábitos alimenticios para su dieta diaria enfocado bajo la idea de la “Exposición de un igual”.

Este videojuego no sólo sirve para ocupar el tiempo libre de los niños, ya que la historia y el breve argumento del juego hace que el niño se adentre del todo en el mundo de fantasía preparado para él. Una vez el niño se familiariza con este entorno, queda preparado para aprender conocimientos educativos y encariñar al niño con todo tipo de verduras y frutas disminuyendo el rechazo que el niño sienta a primera, segunda o tercera vista frente a un alimento. Debido a la idea principal utilizada, el niño fácilmente puede extrapolar los sentimientos y conocimientos recibidos tras haber utilizado este videojuego el cual cada vez que lo juegue, será ligeramente diferente debido a variantes aleatorias internas.

Gracias a este juego educativo para PC, los padres tendrán otra alternativa llena de ventajas para que sus niños ingieran todo tipo de verduras a la hora de la comida. Ésta también es perfectamente compatible con clásicas alternativas como decorar la comida para hacerla más apetecible o contar historietas sobre héroes que comen comida sana y salvan al mundo.

Abstract

The video game industry today is, a market that continues to boom in the leisure sector in Spain and is consumed by all kinds of people from all ages. This means that can be used not only as entertainment but also to positively influence society motivating certain habits and instilling consumer knowledge. The high time that many young children spend on video games should not be ignored, because while a child enjoys, can also be formed as a person in this society as technology.

This project aims to develop a 3D game on PC using the Unity3D game engine and with it help solve an existing problem in feeding young infants. The game takes place in different scenarios of a Tower Defense in which children learn strategies as they decide and acquire educational knowledge. During the game are encouraged with excellent eating habits to your daily diet focused on the idea of "Exposure of an equal".

This game not only serves to fill the free time of children, since history and the short story of the game makes the child to delve completely into a fantasy world prepared for him. Once the child becomes familiar with this environment is ready to learn educational skills and become attached to the child with all kinds of vegetables and fruits decreasing the rejection that the child feels at first, second or third time to a food. Because of the main idea used, the child can easily extrapolate the feelings and knowledge received after using this game where every time you play it, it will be slightly different due to internal random variations.

Thanks to this educational game for PC, parents have other choice full of advantages for their children in order to eat all kinds of vegetables to lunch. This is also perfectly compatible with conventional alternatives such as decorating the food to make it more palatable or tell stories about heroes who eat healthy food and save the world.

Índice de contenidos

1.	Introducción general	1
1.1	Motivación	1
1.2	Objetivos	3
1.3	Estructura del trabajo	4
2.	Estado de la cuestión	3
2.1	Camuflar la comida	3
2.2	Premiar la correcta alimentación	6
2.3	Uso de una distracción.....	8
2.4	Uso de cuentos o historias o similar	11
2.5	Medicación.....	14
2.6	Videojuegos	16
2.6.1	Videojuegos en PC	17
2.6.2	Videojuegos en Smartphone	25
2.7	Solución propuesta	29
2.7.1	Cómo la solución afronta el problema.....	29
2.7.2	Datos extra obtenidos del estudio	40
2.8	Comparativa de las distintas soluciones y la propuesta.....	45
3	Análisis, diseño, implementación e implantación.....	47
3.1	Metodología, ciclo de vida y marco regulador	47
3.2	Análisis	50
3.2.1	Requisitos de usuario	51
3.2.2	Requisitos de sistema	53
3.2.3	Problemas varios del análisis.....	55
3.3	Diseño	57
3.3.1	Comparativa de herramientas.....	57
3.4	Diseño de la propuesta	71
3.4.1	Casos de uso	71
3.4.2	Matriz de trazabilidad.....	72
3.4.3	Diagrama de navegación del videojuego	73
3.4.4	Diagramas de secuencia	74

3.4.5	Diagrama entidad-relación	75
3.4.6	Componentes del motor gráfico	76
3.5	Implementación	88
3.6	Implantación	89
4	Planificación y presupuesto	91
4.1	Planificación	91
4.2	Presupuesto	93
5	Conclusiones y trabajos futuros	96
5.1	Conclusiones	96
5.1.1	Sistema desarrollado	96
5.1.2	Proceso software elegido	98
5.1.3	Conclusiones personales	99
5.2	Trabajos futuros	103
5.2.1	Tutorial	103
5.2.2	Escenarios	103
5.2.3	Modo historia	103
5.2.4	Visualización de efectos al superar o no un nivel	104
5.2.5	Multiplataforma	105
5.2.6	Otros problemas	105
	Apéndice I: Manual de uso para usuario genérico	106
	Bibliografía	110

Índice de imágenes

EJEMPLO 1 DE “CAMUFLAR LA COMIDA”	4
EJEMPLO 2 DE “CAMUFLAR LA COMIDA”	4
EJEMPLO 3 DE “CAMUFLAR LA COMIDA”	5
EJEMPLO 1 DE “PREMIAR LA ALIMENTACIÓN”	7
EJEMPLO 1 DE “USO DE DISTRACCIÓN”	9
EJEMPLO 2 DE “USO DE DISTRACCIÓN”	9
EJEMPLO 3 DE “USO DE DISTRACCIÓN”	10
EJEMPLO 1 DE “USO DE CUENTOS O SIMILAR”	12
EJEMPLO 2 DE “USO DE CUENTOS O SIMILAR”	12
EJEMPLO 3 DE “USO DE CUENTOS O SIMILAR”	12
EJEMPLO 1 DE “USO DE MEDICACIÓN”	14
EJEMPLO 2 DE “USO DE MEDICACIÓN”	14
EJEMPLO 1 DE WEB DE JUEGOS EDUCATIVOS.....	18
EJEMPLO 2 DE WEB DE JUEGOS EDUCATIVOS.....	18
EJEMPLO 3 DE WEB DE JUEGOS EDUCATIVOS.....	19
EJEMPLO 1 DE BANCO DE RECURSOS	19
EJEMPLO 1 DE WEB ESPECIALIZADA.....	20
EJEMPLO 1-1 DE “VIDEOJUEGO EN PC”	21
EJEMPLO 1-2 DE “VIDEOJUEGO EN PC”	22
EJEMPLO 1-3 DE “VIDEOJUEGO EN PC”	22
EJEMPLO 2-1 DE “VIDEOJUEGO EN PC”	23
EJEMPLO 2-2 DE “VIDEOJUEGO EN PC”	24
EJEMPLO 2-3 DE “VIDEOJUEGO EN PC”	24
EJEMPLO 3 DE “VIDEOJUEGO EN SMARTPHONE”	26
EJEMPLO 4 DE “VIDEOJUEGO EN SMARTPHONE”	27
MENÚ PRINCIPAL DE “MI SOLUCIÓN”	29
NIÑO Y NIÑA EN “MI SOLUCIÓN”	31
DUENDE VERDURIANO EN “MI SOLUCIÓN”	31
IMAGEN DE DERROTA DE “MI SOLUCIÓN”	32
PLATO BRÓCOLIS DE “MI SOLUCIÓN”	33
MONTÓN DE PIRULETAS DE “MI SOLUCIÓN”	34
CONSEJOS SALUDABLES DE “MI SOLUCIÓN”	35
ESCENARIO1 TD DE “MI SOLUCIÓN”	36
BOCETO INICIAL DE INTERFAZ 2D DE TD DE “MI SOLUCIÓN”	36
HERRAMIENTA AUDIO: AUDACITY	57
HERRAMIENTA AUDIO: ADOBE AUDITION CC	58
HERRAMIENTA “AUDIO”: CAMTASIA STUDIO.....	59
HERRAMIENTA “AUDIO”: ADOBE PREMIERE PRO CC	60
HERRAMIENTA MODELADO3D: SCULPTRIS.....	61

HERRAMIENTA MODELADO3D: CINEMA4D	62
HERRAMIENTA MODELADO3D: BLENDER.....	63
HERRAMIENTA MODELADO3D 3DSMAX.....	63
HERRAMIENTA MODELADO3D: MAYA	64
HERRAMIENTA IMAGEN: PAINT.....	64
HERRAMIENTA IMAGEN: GIMP	65
HERRAMIENTA IMAGEN: ADOBE PHOTOSHOP CC.....	66
HERRAMIENTA MOTOR GRÁFICO: UNITY3D	67
HERRAMIENTA MOTOR GRÁFICO: ADOBE FLASH PROFESSIONAL CC.....	68
HERRAMIENTA MOTOR GRÁFICO: WIMI5.....	69
HERRAMIENTA MOTOR GRÁFICO: UNREAL ENGINE4	70
DIAGRAMA DE CASOS DE USO	71
DIAGRAMA DE NAVEGACIÓN DEL VIDEOJUEGO	73
DIAGRAMA DE SECUENCIA 1, CONFIGURACIÓN.....	74
DIAGRAMA DE SECUENCIA 2, SALIR DE LA APLICACIÓN	74
DIAGRAMA DE SECUENCIA 3, MECÁNICA DE JUEGO	75
DIAGRAMA ENTIDAD-RELACIÓN	76
SELECCIÓN DE ESCENAS A COMPILAR Y MONTAR	78
MÓDULO TRANSFORM EN UN ELEMENTO	79
MÓDULOS GUITEXT Y GUITEXTURE EN UN ELEMENTO	80
MÓDULO SCRIPT EN UN ELEMENTO	81
MÓDULO ANIMATOR EN UN ELEMENTO	81
MÓDULO MESH RENDER DE UN ELEMENTO	82
MÓDULO X COLLIDER DE UN ELEMENTO.....	82
MÓDULO CHARACTER CONTROLLER DE UN ELEMENTO	83
MÓDULO RIGIDBODY DE UN ELEMENTO	83
MÓDULOS AUDIOLISTENER Y AUDIOSOURCE DE UN ELEMENTO	84
JERARQUÍA PRINCIPAL DE ALMACENADO DE DATOS EN UNITY3D	85
ESTRUCTURA DEL DIRECTORIO MODELOS.....	86
ESTRUCTURA DEL DIRECTORIO SCRIPTS	86
MENÚ DE APERTURA DEL VIDEOJUEGO	89
DIAGRAMA DEL GANTT DEL PROYECTO	92
BOTONES 1, 2 Y 3 DEL MENÚ PRINCIPAL.....	106
ESQUINA INFERIOR IZQUIERDA DE LA INTERFAZ DE JUEGO	107
ESQUINA INFERIOR DERECHA DE LA INTERFAZ DE JUEGO	107
BOTONES COMENZAR OLEADA Y MEJORAR DUENDE DE LA INTERFAZ DE JUEGO.....	107

Índice de tablas

TABLA 1 CONTRASTE DE ASPECTOS DEL JUEGO CON SUS ELEMENTOS, MECÁNICAS Y LA EDUCACIÓN	40
TABLA 2 PROPIEDADES POSITIVAS DE ALIMENTOS SANOS DE “MI SOLUCIÓN”	44
TABLA 3 COMPARATIVA DE SOLUCIONES Y LA PROPUESTA	45
TABLA 4 EJEMPLO DE REQUISITO	51
TABLA 5 REQUISITO DE USUARIO 01	51
TABLA 6 REQUISITO DE USUARIO 02	52
TABLA 7 REQUISITO DE USUARIO 03	52
TABLA 8 REQUISITO DE USUARIO 04	52
TABLA 9 REQUISITO DE USUARIO 05	52
TABLA 10 REQUISITO DE USUARIO 06	52
TABLA 11 REQUISITO DE USUARIO 07	52
TABLA 12 REQUISITO DE USUARIO 08	53
TABLA 13 REQUISITO DE USUARIO 09	53
TABLA 14 REQUISITO DE USUARIO 10	53
TABLA 15 REQUISITO DE USUARIO 11	53
TABLA 16 REQUISITO DE USUARIO 12	53
TABLA 17 REQUISITO DE SISTEMA 01	54
TABLA 18 REQUISITO DE SISTEMA 02	54
TABLA 19 REQUISITO DE SISTEMA 03	54
TABLA 20 REQUISITO DE SISTEMA 04	54
TABLA 21 REQUISITO DE SISTEMA 05	54
TABLA 22 REQUISITO DE SISTEMA 06	55
TABLA 23 REQUISITO DE SISTEMA 07	55
TABLA 24 REQUISITO DE SISTEMA 08	55
TABLA 25 MATRIZ DE TRAZABILIDAD.....	73
TABLA 26 ESTADÍSTICAS DEL CÓDIGO	88
TABLA 27 ESTADÍSTICAS DE INTERFACES	88
TABLA 28 ESTADÍSTICAS DE MODELOS3D	88
TABLA 29 ESTADÍSTICA DE AUDIO.....	88
TABLA 30 COSTES DEL PROYECTO POR PERSONAL HUMANO.....	93
TABLA 31 COSTES DEL PROYECTO POR LICENCIAS DE SOFTWARE	94
TABLA 32 COSTES DEL PROYECTO POR ADQUISICIÓN Y USO DE HARDWARE	94
TABLA 33 COSTES DEL PROYECTO POR GASTOS INDIRECTOS	95
TABLA 34 COSTES TOTALES DEL PROYECTO.....	95

1. Introducción general

Este capítulo dividido en tres partes, pretende, en primer lugar, expresar el porqué de este proyecto repasando brevemente el estado de la sociedad y el hueco en que encaja. En segundo lugar, se listarán los objetivos que se han querido cumplir durante la realización de este proyecto. En tercer lugar, se hará un pequeño esquema acerca de cómo se organizan el resto de capítulos y qué va a tratar cada uno de ellos.

1.1 Motivación

En los tiempos que corren, se vive en una sociedad en la que el peso de los videojuegos es cada vez mayor en todo tipo de ámbitos y para todo tipo de público. A día de hoy, es bastante común viajar en el tren, metro y ver a todo el mundo con la cabeza baja observando su Smartphone mientras habla o bien juega algún videojuego sin ninguna finalidad más que el ocio en los ratos libres.

Yo soy una persona a la que le encantan los videojuegos, los cuales llevo disfrutando desde bien pequeños y me han ayudado a mejorar mi nivel de inglés, ampliar vocabulario de la lengua materna y, entre muchos otros, aprender todo tipo de culturas externas reflejadas en dichos juegos. Al igual que yo he disfrutado de ellos, espero que las futuras generaciones, que cada vez empiezan antes, no sólo aprendan y disfruten, sino que mejoren como personas motivando e incentivando conductas saludables para ellos mismos y la sociedad.

En una persona adulta, los videojuegos es improbable que modifiquen tus conductas o te puedan ayudar a formarte como persona. Como mucho, puedas aprender cultura, historia o conocimientos de la sociedad o actualidad. Sin embargo, difícilmente puedas jugar algo que te ayude con tu vida diaria creando una mejor conducta en tu hogar que es lo que este proyecto pretende.

¿Dónde está enfocado este proyecto? En crear un videojuego que mediante su uso pueda ayudar a los públicos más jóvenes, en concreto infantes de 5 a 7 años con malos hábitos alimenticios. A esta edad, los niños ya deben haber probado por primera vez la inmensa mayoría de los alimentos de una dieta sana y equilibrada y con ello surge el problema que este proyecto quiere combatir: “Los niños que han probado ciertas verduras, hortalizas, legumbres y frutas, debido ya sea al sabor o a la textura, se niegan rotundamente a volver a consumirlas”. Este problema tanto para los niños, que puedan tener problemas con el crecimiento y problemas de salud por carencia de vitaminas, tanto para los padres que, evidentemente se preocupan por sus hijos trayéndoles quebraderos de cabeza también a ellos.

Siendo este el problema a intentar solucionar mediante el uso de un videojuego como herramienta he puesto mi empeño y dedicación en este proyecto a hacerlo realidad. Inicialmente este trabajo se ha realizado para poder ejecutarse en un PC para

los niños en casa, aunque para un futuro se podría realizar para Smartphone y otras plataformas para poder disfrutar sus beneficios en cualquier lugar de una forma más cómoda y sencilla.

1.2 Objetivos

Los objetivos principales de este proyecto son:

- Facilitar a los padres una alternativa novedosa y llamativa para sus hijos, con la que puedan aprender divirtiéndose ocupando de esta forma parte de su tiempo libre.
- Concienciar a los niños sobre los beneficios de una dieta saludable, y los problemas que pueden surgir al no llevarla a cabo.

Los objetivos principales para que este proyecto tenga éxito:

- La realización de un videojuego para PC, teniendo como objetivo un público muy joven pero que resulta muy atrayente para dicho público.
- Fomentar intensamente a los jugadores como una dieta sana y equilibrada influye completamente en el crecimiento de los niños a lo largo del tiempo.

Los objetivos personales de este proyecto son:

- Comprobar que un alumno de grado es capaz de llevar a cabo un proyecto en solitario de principio a fin en un entorno de trabajo nuevo y manejando tecnologías nuevas.
- Aportar algo que merezca la pena a la industria del videojuego.

1.3 Estructura del trabajo

A continuación, se puede leer la descripción de los siguientes capítulos de esta memoria:

- **Capítulo 2: Estado de la cuestión:**
En esta sección del documento, se analizarán las características de otras posibles soluciones a este problema, se contrastarán con la aquí propuesta y se ilustrará mediante una tabla comparativa por qué esta solución es mejor que las actuales.
- **Capítulo 3: Análisis, diseño, implementación e implantación:**
En este capítulo se tratará sobre la metodología usada, el marco regulador, requisitos, herramientas utilizadas, diagramas de diseño, así como también se verán los apartados que más dificultad han tenido y el cómo se puede usar este proyecto.
- **Capítulo 4: Planificación y presupuesto:**
Aquí estará el diagrama de Gantt seguido de principio a fin junto al precio que equivale el tiempo asociado al diagrama tanto del alumno tanto del tutor.
- **Capítulo 5: Conclusiones y trabajos futuros:**
En este apartado estarán contenidas las conclusiones del proyecto se repasará brevemente el proceso de creación y se mostrará el estudio de líneas futuras.
- **Apéndices y bibliografía:**
Apéndices del documento y la bibliografía utilizada.

2. Estado de la cuestión

Este capítulo está dividido en ocho apartados donde los seis primeros tratan de unas posibles soluciones al problema ya existente en nuestra sociedad pero tiene alguna carencia. A continuación se podrán contemplar dos apartados más, la solución aquí propuesta y finalmente una tabla comparativa de las características de estas soluciones para sacar una conclusión visual del tema.

Remarcando como problema “Los niños que han probado ciertas verduras, hortalizas, legumbres y frutas, debido ya sea al sabor o a la textura, se niegan rotundamente a volver a consumirlas”, he aquí las respuestas.

2.1 Camuflar la comida

Con el fin de conseguir que el niño coma todo tipo de comida, una práctica muy común es la de esconder/maquillar/camuflar la comida para que el niño “sin darse cuenta” piense que está comiendo sólo un alimento. La finalidad consiste en que cuando se quiera ingerir un alimento, en realidad está ingiriendo una porción de ese alimento y de otro alimento sano pero indeseado al mismo tiempo.

Esta es una de las técnicas más utilizadas por los padres para conseguir que sus hijos coman de todo. Aunque en muchas ocasiones funcione ya que al engañar la mente del niño que automáticamente rechaza un alimento sólo por saber que es “eso que no gusta” conlleva una elaborada preparación de la forma de los alimentos para cada comida. Esta preparación significa tiempo para los padres y un esfuerzo mental ya que requiere innovación e imaginación. La parte positiva es que normalmente cualquier comida se puede moldear con las herramientas adecuadas por lo que no requiere ningún sobre coste económico fuera del pensado y si se moldea con formas llamativas para el niño, será más fácil captar su atención para una ingesta directa.

Principales características positivas:

- El niño ingiere verduras.
- Coste económico nulo.
- Puede captar fácilmente la atención del niño.

A continuación se pueden ver unas ilustraciones de la praxis de esta solución:



Imagen 1 Ejemplo 1 de “camuflar la comida”



Imagen 2 Ejemplo 2 de “camuflar la comida”



Imagen 3 Ejemplo 3 de “camuflar la comida”

Esta práctica es una tradición que lleva repitiéndose durante generaciones por lo que no puede considerarse que tenga un autor como tal o una entidad bibliográfica a la que referirse. Igualmente, a continuación existe un enlace de un blog popular que trata esta solución.

<http://tallersmariavictrix.blogspot.com.es/2014/07/ideas-creativas-para-decorar-la-comida.html>

2.2 Premiar la correcta alimentación

Al igual que el nombre de este apartado, esta alternativa a la solución 1 consiste simplemente en ofrecer una recompensa por el esfuerzo que los niños hacen si “por propia voluntad” deciden ingerir esos alimentos que tanto les disgustan. Los premios pueden ser de todo tipo, desde poder tomar dos postres, no tener que recoger los juguetes, poder ver la televisión más tiempo o hasta ofrecer juguetes nuevos, como ejemplos clásicos. La alternativa opuesta sería la de castigar al niño en caso de no comer verduras o posponer la comida hasta la siguiente hora de comer, sin embargo eso no es considerado una solución ya que trae más efectos negativos que positivos.

Esta técnica es otra de las más utilizadas al igual que la anterior. Aunque esta solución puede dar resultados positivos durante un largo periodo de tiempo, lo cual resolvería el problema, ofrecer premios como “doble postre” a la larga igualmente puede producir el efecto “contrario” el cual podría ser el sobrepeso. Esto significaría añadir otro problema más que combatir junto al actual siendo más un retroceso que un avance. Otro punto negativo es que si al niño se le enseña que siempre que haga algo que no le guste pero debe hacerlo, se le va a recompensar por ello, puede producir graves problemas de disciplina que repercutirán en su futuro. Esto significa que en principio, esta solución para el problema, si se utiliza a la larga lo más seguro es que provoque más problemas de los que solucione.

Para poder premiar a los niños, previamente hay que conocer sus gustos y asignar el premio correspondiente a sus gustos, lo cual no debería ser tarea fácil para los padres. En muchos casos se suele aplicar la resolución de ofrecer doble postre como recompensa, lo que significa igualmente un pequeño sobrecoste económico que a la larga sea más significativo.

Principales características positivas:

- El niño ingiere verduras.

A continuación se puede ver una ilustración de la praxis de esta solución:

LA COMIDA COMO PREMIO



Imagen 4 Ejemplo 1 de "premiar la alimentación"

De forma paralela a la solución1, esta solución también es algo cada vez más común hoy en día pero también es una tradición generacional. Siendo esta una tradición sin autor ni entidad bibliográfica, como último dato se muestra un enlace de una web muy utilizada de consejos médicos para la salud de los hijos.

<http://www.webconsultas.com/bebes-y-ninos/educacion-infantil/que-hacer-si-tu-hijo-se-niega-comer-verduras-7459>

2.3 Uso de una distracción

Esta solución, que también se suele usar junto a la 1, trata de “engañar” al niño consiguiendo que centre la mayor parte de su atención en una distracción. Con esto se consigue que mientras el niño no esté atento de qué es la comida, simplemente la ingiera ignorando de qué se trata pues lo que le importa, lo está viendo con los ojos y no lo que saborea. De ahora a hace unas tres décadas, lo más común ha sido el uso de un televisor retransmitiendo dibujos animados o algún otro programa infantil pensado adrede para llamar la atención de los niños. Hoy en día, debido a los avances tecnológicos, se están empezando a usar teléfonos inteligentes o tabletas con el mismo uso, ya que debido a su tamaño y movilidad es más fácil trasladarlos a cualquier lugar en el que el niño coma (siempre que estén sus padres ahí).

Cada vez esta técnica está siendo más utilizada en niños con 6 meses en adelante, por lo que los efectos positivos y negativos hay que valorarlos con mucho ojo ya que, como el problema que se quiere solucionar se da en niños de 5 a 7 años de edad, puede tener resultados excelentes o nefastos. Esto depende del abuso de esta técnica ya que el primer y más efecto nefasto que tiene, es que si la rutina del niño es siempre que se va a comer, tiene que haber un dispositivo que reproduzca un clip con dibujos animados u otro contenido audiovisual, el niño asocia comer con ver dibujos. Si un niño hace esta asociación debido al abuso, es muy posible que se niegue a comer cualquier tipo de alimento si no hay un dispositivo al lado que reproduzca algo cerca de él. Si igualmente el niño asocia “comer = masticar + observar algo” puede crear problemas de sociabilidad ya que lo normal es que se aislen con su distracción ignorando al resto de comensales generando un problema no deseado también difícil de erradicar. Esta es una buena solución pero, como con cualquier cosa, nunca es bueno abusar.

Aunque a edades tempranas, se pueda “engañar” a un niño siempre con el mismo clip, un niño de 5 años fácilmente puede reconocer el mismo clip si se reproduce muchas veces en poco tiempo. Los padres en caso de usar esta posible solución con frecuencia, deberían vigilar qué es lo que ven sus hijos, pues no todos los contenidos que pueda mostrar una televisión son aptos para ellos. En caso de usar una tableta o teléfono inteligente, son los padres quienes tienen que estar buscando vídeos en alguna web, o haber tenido que dedicar un tiempo extra a seleccionar qué sí y qué no deben ver sus hijos. Un punto muy positivo, es que si se escoge selectamente el contenido a visualizar, el niño inconscientemente puede aprender conocimientos útiles en su vida o incluso para fomentarle a comer verduras por su cuenta. Cierto es que ahora mismo no todas las familias con hijos pequeños tienen tabletas o un teléfono inteligente con un tamaño aceptable para poder dejar mirándolo a un niño pequeño sin que fuerce su vista, por lo que es muy posible que esto significa un sobre coste económico para los padres.

Principales características positivas:

- El niño ingiere verduras.
- Puede captar fácilmente la atención del niño.
- Puede contener material educativo.

A continuación se pueden ver unas ilustraciones de la praxis de esta solución:

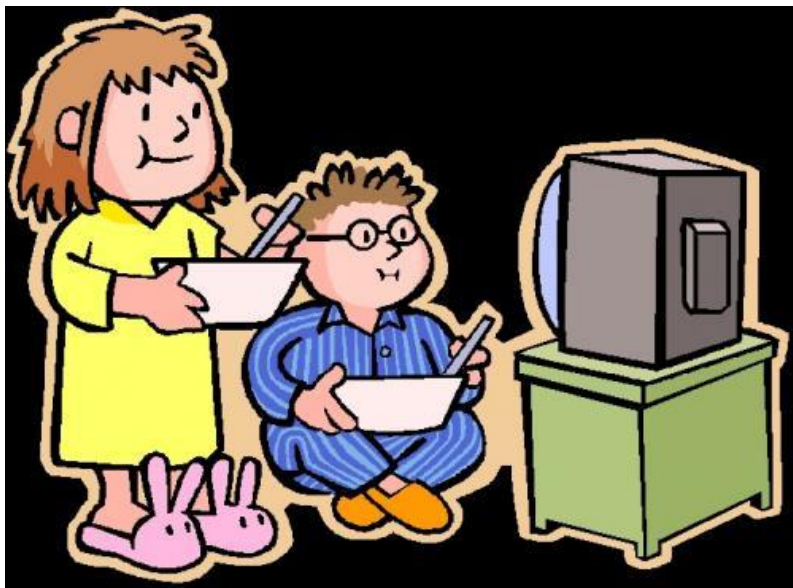


Imagen 5 Ejemplo 1 de “uso de distracción”



Imagen 6 Ejemplo 2 de “uso de distracción”



Imagen 7 Ejemplo 3 de “uso de distracción”

Esta solución cada día más utilizada en edades iniciales ha ido cambiando a lo largo de los tiempos y ahora, en la era tecnológica, tiene matices positivos y negativos. Esta información y mucha más se ha sacado de la sección VII del siguiente libro: *Educación a niños y niñas de 0 a 6 años* [\[1\]](#)

2.4 Uso de cuentos o historias o similar

Una alternativa menos usada debido a la complejidad que tiene, es la del uso de cuentos o historias a los niños para convencerles de que comer verduras es lo mejor que pueden hacer para crecer grandes y fuertes. Para cumplir el objetivo, no vale cualquier cuento, debe ser uno con elementos normalmente de fantasía para que la imaginación del niño pueda darle el empujón para hacer caso a las enseñanzas de dicho cuento. Los cuentos no importa de lo que traten, siempre y cuando transmitan que es bueno comer verduras y que es lo mejor que pueden hacer, ya que sino hasta “les podrían pasar cosas malas”.

Esta solución se suele utilizar en los centros educativos como refuerzo a la educación de los padres ya que el efecto positivo se transmite de un niño a otro mientras lo escuchan. Se hablará con más detenimiento de este efecto en el apartado 2.7. Como la transmisión de la idea no es $1 \rightarrow 1$, sino $1 \rightarrow m$ existen una gran cantidad de formas espaciales de llevar a cabo esta praxis. Como bien dice el título, se puede contar un cuento o una fábula de un libro, alguna historia inventada de un niño que no quería comer verduras, frutas u hortalizas, también se pueden llevar a cabo talleres de marionetas donde estas sean alimentos donde éstas te recomiendan que las comas debido a que son muy buenas para los niños. Existe una gran cantidad de formas de manifestar este concepto que no son las formas aquí dichas.

Un punto negativo que se puede encontrar, es que, ya sean sesiones con una gran cantidad de niños o una particular, requiere un alto tiempo de preparación y un equipamiento (ya sean libros o marionetas o lo que sea). El equipamiento tampoco es gratuito y hay que renovarlo con el tiempo, o ampliarlo para mejorar las historias pues siempre con los mismos personajes se puede llegar a la monotonía y esto dejará de ser atractivo para los niños y se volverá algo pesado que querrán evitar. Otro detalle que puede ser negativo, es que el efecto que esto produce es mayor con una gran cantidad de niños (más de 5) por lo que, si por la razón que sea, las sesiones que normalmente no suelen ser diarias, tienen poca audiencia, el efecto que se emite será menor del deseado dando paso a mejorar utilizar otras soluciones y esta como sólo un refuerzo.

Principales características positivas:

- El niño entiende que debe ingerir verduras.
- El tiempo empleado por los padres puede ser nulo si lo realiza su centro educativo.
- El coste económico puede ser nulo para los padres si lo realiza su centro educativo.
- Puede contener material educativo.
- El niño puede involucrarse con el método según cómo se realiza la actividad.

A continuación se pueden ver unas ilustraciones de la praxis de esta solución:

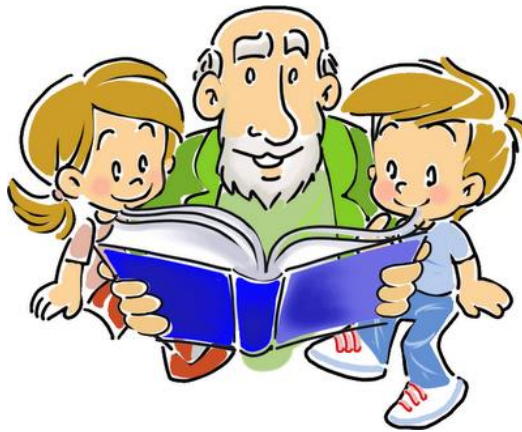


Imagen 8 Ejemplo 1 de “uso de cuentos o similar”



Imagen 9 Ejemplo 2 de “uso de cuentos o similar”



Imagen 10 Ejemplo 3 de “uso de cuentos o similar”

Toda la información de esta solución también es clásica y tradicional pero está muy documentada en la web y por eso mismo a modo de referencia y de “ejemplo 4” se pueden consultar las siguientes referencias:

Información:

<http://www.educapeques.com/escuela-de-padres/el-valor-de-los-cuentos.html>

Ejemplo de cuento:

<http://cuentosparadormir.com/infantiles/cuento/el-nino-y-el-repollo>

2.5 Medicación

Aunque esta es la peor solución posible, en diversas ocasiones se puede tener que llegar a recurrir a ella. Esta consiste en, tras localizar qué carencia de vitaminas tiene el niño por dichos alimentos que no consume, suministrar dichas vitaminas mediante cápsulas o jarabe al niño para que esto no suponga un grave problema al organismo del niño y siga creciendo con normalidad. Esto es, al mismo tiempo, una solución eventual y también una solución a las consecuencias de no haber paliado el problema inicial.

Por mala suerte, esta opción no tiene ninguna parte positiva salvo que el niño acaba tomando las vitaminas que el cuerpo necesita. Términos negativos son, entre otros, el alto precio de las medicinas, el rechazo cada vez mayor que el niño mostrará a tomar esta opción, el tiempo que habrá que emplear hasta conseguir que el niño forzado tome los medicamentos o la falta de disciplina del niño cuando vea que existen alternativas si no desea hacer algo.

Principal característica positiva:

- El niño recibe las vitaminas de las verduras de forma artificial.

A continuación se pueden ver unas ilustraciones de la praxis de esta solución:



Imagen 11 Ejemplo 1 de “uso de medicación”



Imagen 12 Ejemplo 2 de “uso de medicación”

Debido a la escasa antigüedad de la penicilina esta solución, en comparación con las anteriores, es bastante más moderna y si uso en niños recomendado o no y las conclusiones anteriores han sido extraídas de la siguiente web:

<http://blog.hola.com/farmaciameritxell/2013/12/vitaminas-para-ninos.html>

2.6 Videojuegos

Dentro del nuevo gran abanico de las soluciones tecnológicas a problemas cotidianos, se puede entrar de lleno en el mundo de los videojuegos. Este es un mundo que, aunque aún es joven, ya posee una amplia gama de formas de juego, plataformas y géneros de juego. Este método de resolución de problemas consiste en, mientras el niño se sumerge en la historia del videojuego y es él el protagonista de una historia guiada, diversos factores del propio juego introducidos adrede inyectan información al niño que juega, pudiendo extrapolar dichos conocimientos al mundo real. Es decir, si un niño juega a un juego que trata de un niño que tiene que ir salvando animales enjaulados y estos animales se le acercan y juegan con ese niño virtual, esto puede ayudar a que jóvenes con miedo a animales puedan entender que, si en ese juego no pasa nada y está bien acariciar animales de compañía, en la vida real tampoco pasa nada malo.

Esta solución puede ser extremadamente útil para los padres pues, independientemente del dispositivo, hay una gran cantidad material gratuito para que los niños se puedan beneficiar de ella. También son muy beneficiosos porque si los niños aprenden el manejo de cada juego, son ellos mismos los que dedican tiempo sin necesidad de supervisión adulta constante. Debido al aspecto caricaturesco de la mayoría de los videojuegos y sus voces, llaman extremadamente fácil la atención de los niños recibiendo una aceptación plena por su parte. Un punto muy a favor es que, está comprobado empíricamente que si una persona es la que vive una aventura, aunque sea virtualmente, todo lo que se haya aprendido en ella se graba a fuego en la memoria, por lo que esta solución es sublime para enseñar información nueva.

El lado oscuro de los videojuegos es que sin un control paterno, es muy posible que causen adicción a edades tan tempranas, por lo que la cantidad de horas de juego sí debe ser supervisada por los padres o tutores. De forma similar a la del apartado 2.3, sí hay que prestar atención a los juegos que puedan recomendar y no se haya supervisado su contenido ya que los niños absorben información muy fácilmente, tanto positiva tanto negativa. Para evitar este último punto, existen webs con multitud de juegos educativos ya supervisados, de las que se hablarán más adelante con más detenimiento.

Principales características positivas:

- El niño entiende que debe ingerir verduras.
- El tiempo empleado por los padres puede ser nulo si lo realiza su centro educativo o si no es la primera vez que prueba ese videojuego.
- El coste económico puede ser nulo para los padres según dónde busque el videojuego ya que generalmente son gratuitos.
- Puede contener material altamente educativo.

- El niño se involucra con este método en mayor o menor medida según la motivación del propio juego.

Para poder tratar mejor este marco tan amplio, de cara a enfrentar el problema propuesto a resolver en este proyecto, el resto de este sub-apartado se dividirá en los dos más cercanos a los niños de 5 a 7 años. Videojuegos en plataforma fija PC y videojuegos en plataforma móvil Smartphone.

2.6.1 Videojuegos en PC

Los videojuegos en PC están caracterizados principalmente por su manejo con el ratón y el teclado que envían información al ordenador y muestra el entorno de posibilidades por la pantalla. Todo aquello que se quiera hacer en ordenador, en principio está limitado a las funciones del ratón y teclado aunque también se pueden añadir periféricos aunque no suele ser normal. Dentro de los juegos de PC, este documento se centra en los educativos relacionados con el problema principal ya que, como dice la introducción, el abanico de posibilidades a tratar es prácticamente ilimitado.

Existen una gran cantidad de videojuegos educativos para esta plataforma enfocados a esta edad, sin embargo, sólo una limitada gama se ofrece a tratar de lejos este problema de alimentación. Las temáticas más comunes de estos videojuegos suelen ser aquellas que sirvan para potenciar la expresividad de los niños mediante el juego, otras pueden ayudar a ejercitar la memoria, resolver puzles con algún cálculo mental simple para agilizar la mente o bien revivir una historia del pasado del hombre y aprender de ella. Todas estas temáticas se pueden conjugar y mezclar bajo la idea de “fomentar la correcta alimentación infantil”.

Una vez se tiene una idea para hacer el videojuego, se procede a realizarlo mediante las 4 fases de producción del videojuego de las que no se va a tratar ya que no viene al caso. Hace una década el método de propagar estos videojuegos para su uso y poder disfrutar de sus beneficios, era compartirlos/venderlos en formato CD usualmente sólo en los centros educativos, ya que era donde más ordenadores solía haber para poder ejecutarlos y no en la vivienda. A día de hoy es extremadamente más fácil difundir este material educativo pues, aunque igualmente se distribuye en los centros educativos igualmente en CDs/DVDs/pen drives, la flexibilidad que aporta internet a la hora de poder movilizar datos implica el mejor medio posible para propagar el material.

Navegando por la web, se pueden encontrar numerosas webs especializadas en videojuegos especializados en temáticas para niños con objetivo de mejorar distintas facetas y también mejorar sus aptitudes artísticas. En estos portales web, se pueden encontrar videojuegos con la misma idea que se quiere transmitir, entre muchas ideas,

ya que tienen multitud de secciones divididas por géneros y temáticas. Algunos portales web destacables de esta índole son:

- <http://www.elbuhoboo.com/juegos-educativos.php>



Imagen 13 Ejemplo 1 de web de juegos educativos

- <http://www.educapeques.com/juegos-educativos>

Juegos Educativos



Disfruta de nuestros Juegos educativos y didácticos para niños.

Gracias a estos juegos educativos los niños se divierten aprendiendo.

Imagen 14 Ejemplo 2 de web de juegos educativos

- <http://childtopia.com/index.php>



Imagen 15 Ejemplo 3 de web de juegos educativos

También existen bancos de recursos con una sección de juegos educativos como:

- http://www.juntadeandalucia.es/averroes/eltejar/misitio_rafa/banco_de_recursos.htm

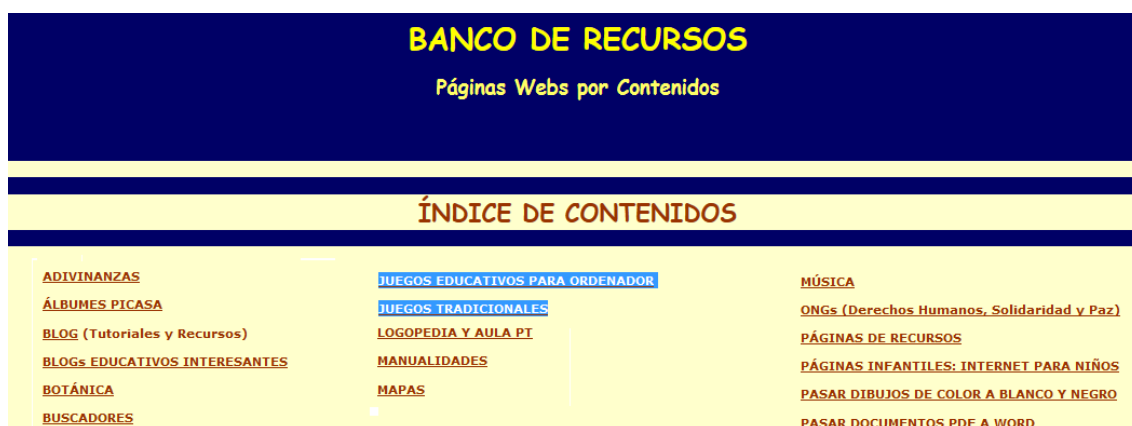


Imagen 16 Ejemplo 1 de banco de recursos

A continuación, tras ver sitios web con videojuegos educativos genéricos, se puede pasar a una web con un problema específico aunque cabe decir que no hay ninguna web específica con juegos adrede para tratar este problema. Como última web introductoria, la siguiente es una especializada con toda la información necesaria para poder abordar la obesidad infantil:

- <http://obesidadinfantil.consumer.es/web/es/6-10/desayuno/index.php>



Imagen 17 Ejemplo 1 de web especializada

Finalmente llega el momento de mostrar y tratar algunos juegos de navegador, ya que instalables no he encontrado, que tratan el tema de la familiarización del niño con las verduras para fomentar la ingestión de estas. Ahora, aunque hay más videojuegos, se van a mostrar brevemente dos juegos de navegador, ejemplos de esta solución, con ciertas carencias que se podrían mejorar.

- *La batidora*



Imagen 18 Ejemplo 1-1 de "videojuego en PC"

Como bien dice el propio enunciado del juego, éste trata de seleccionar de 2 a 3 frutas y verduras, según el nivel de dificultad, para seleccionarlas y según vaya saliendo el color de la mezcla poder saber si estamos escogiendo bien o mal las verduras. Se cuenta con la ayuda de un pajarito que te informará de si se está haciendo bien o mal. Si se combinan bien los alimentos se mostrará una pantalla con información de los alimentos de la mezcla pero, si se hace mal, se solicitará que se reintente de nuevo.

Este juego está muy bien pero está enfocado a, al menos, niños que sepan leer o tengan a alguien al lado que les esté explicando el significado de todo, ya que la enseñanza de este juego se basa en todo lo que se lee. Aunque este juego sea gratuito, se use en centros educativos, el niño sea autodidacta usándolo y quiera enseñar información útil al niño no puede cumplir su propósito adecuadamente. La ausencia de una voz que lea los textos, la carencia de una "historia" o más bien una excusa que le insta a sentirse que debe combinar las frutas y verdura con un fin mayor y sentir la autorrealización o algún detalle que le haga más familiarizarle con aquellas verduras que el internamente odia hacen que este juego no pueda ser exitoso con su cometido.



Imagen 19 Ejemplo 1-2 de “videojuego en PC”



Imagen 20 Ejemplo 1-3 de “videojuego en PC”

- Fruity-Fruit Match

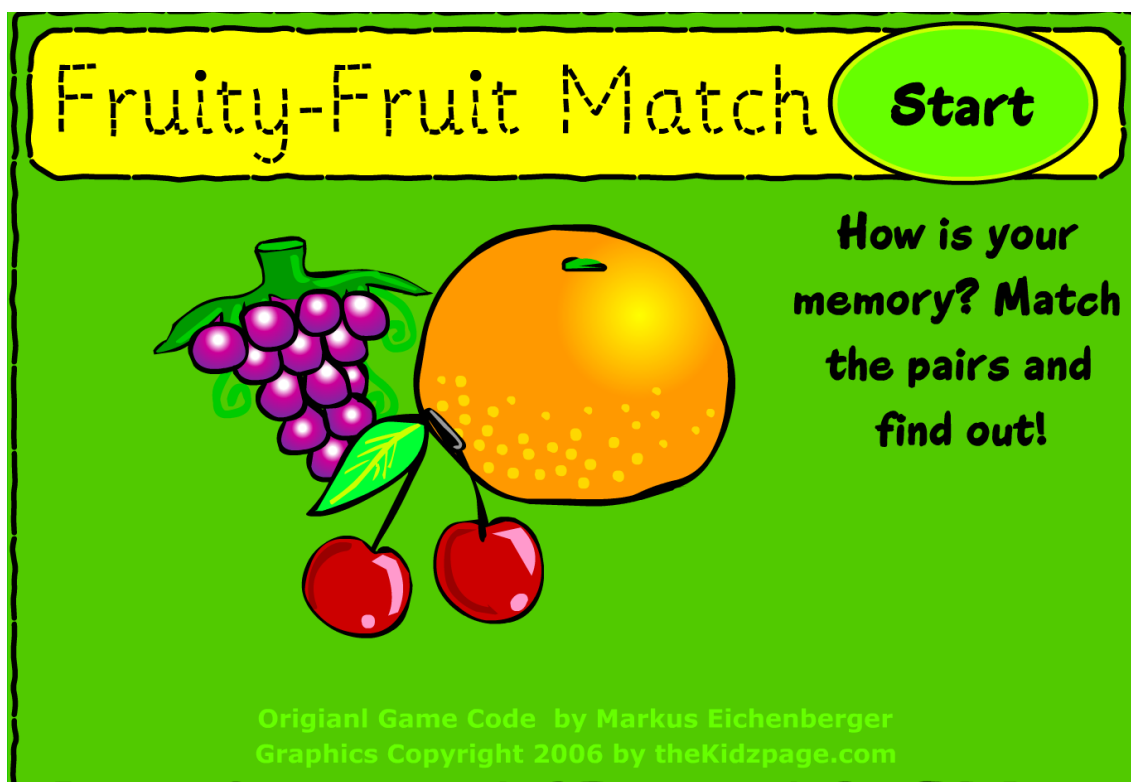


Imagen 21 Ejemplo 2-1 de "videojuego en PC"

Este juego consiste en, pulsar en "Start" para comenzar y, a partir de ahí, buscar las parejas de frutas. Existe una matriz de 4x4 elementos ocultos iguales de dos en dos y el jugador debe seleccionar un elemento oculto, del cual se mostrará el contenido tras seleccionarlo, para después seleccionar otro que también será mostrado. Si al comparar los objetos son iguales, ambos quedarán descubiertos permanentemente. Si dichos objetos son distintos, los dos se volverán a dar la vuelta y no volverá a ser visible el contenido de esa posición. El juego finaliza al encontrar todas las parejas y al usuario se le mostrará un resultado final según el tiempo que haya tardado y el número de intentos. Este último detalle está muy bien pues anima al niño a rejugar el videojuego, pudiendo absorber más información de él.

Ignorando el idioma de este videojuego, éste también posee numerosas carencias que lo hacen menos atractivo respecto a otros juegos. Aunque este juego prácticamente no precisa leer, una introducción oral con los nombres de las frutas sería muy importante para que el niño aprenda algo que no solo sean formas. Una música de fondo alegre, sonidos o caritas alegres en cada fruta también pueden relajar al niño y hacer que el juego le caiga en gracia haciendo que "se encariñe" de dichas frutas. Las frutas utilizadas son las más comunes y normalmente queridas, por lo que añadiendo frutas menos gustosas como ciruelas podría hacer que el niño mejore su relación con aquello que menos agrado tiene.

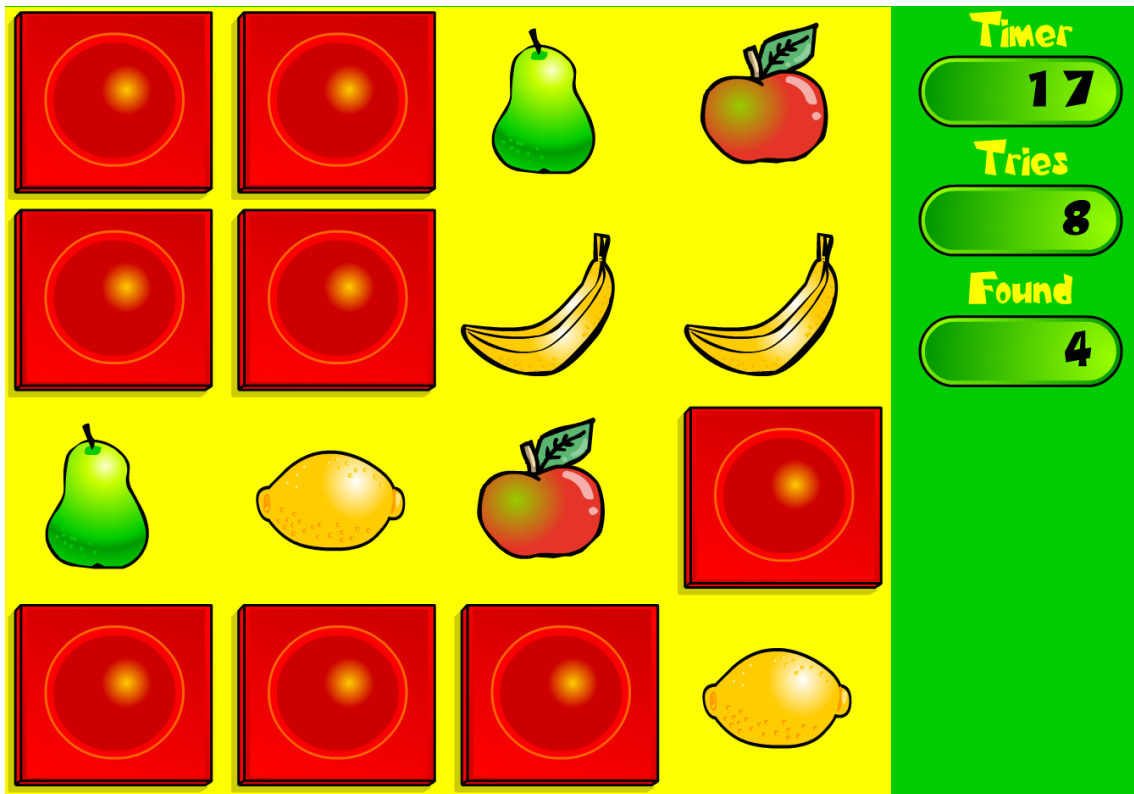


Imagen 22 Ejemplo 2-2 de "videojuego en PC"

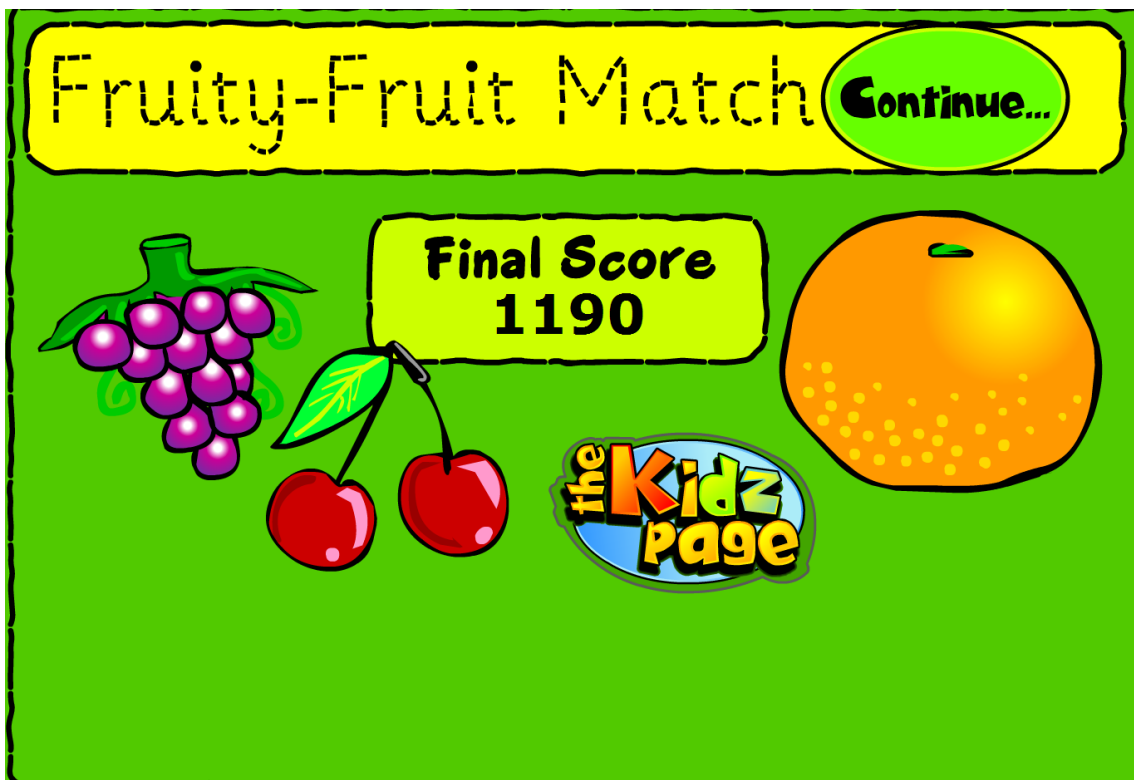


Imagen 23 Ejemplo 2-3 de "videojuego en PC"

2.6.2 Videojuegos en Smartphone

Los videojuegos en Smartphone están caracterizados principalmente por su simple manejo pulsando diferentes secciones de la pantalla a la cual se han atribuido distintas funciones. Todo aquello que se quiera hacer con el dispositivo, en principio está limitado a la función de pulsar y a los botones home, atrás, opciones, volumen+, volumen-. Dentro de los juegos de Smartphone, este documento se centra en los educativos relacionados con el problema principal a continuación del apartado 2.6.1, ya que el abanico de posibilidades a tratar es prácticamente ilimitado.

Existen una gran cantidad de videojuegos educativos también para esta plataforma enfocados a esta edad, sin embargo, sólo una limitada gama se ofrece a tratar de lejos este problema de alimentación ya que la mayoría se limitan al ocio. Las temáticas más comunes de estos videojuegos educativos suelen ser aquellas que sirvan para potenciar la expresividad de los niños mediante el juego, otras pueden ayudar a ejercitar la memoria buscando parejas, resolver puzles con algún cálculo mental simple para agilizar la mente o bien revivir una historia del pasado del hombre y aprender de ella. Todas estas temáticas se pueden conjugar igualmente que los videojuegos de PC y mezclar bajo la idea de “fomentar la correcta alimentación infantil”.

Debido a la creación reciente de este mercado día a día emergente, que del mismo modo tiene las 4 fases principales de creación de un videojuego, tiene un modo de transmisión y propagación igualmente amplio aunque limitado al SO del artefacto. Si el usuario tiene un *Android*, podrá descargar a través de internet, de *Play Store*, por pago o de forma gratuito cualquier juego subido previamente y buscándolo por palabras clave a través de su buscador. Si el usuario tiene un dispositivo de *iOS*, acudirá a la *Apple Store* así como si se tiene un *Blackberry* se acudirá a la correspondiente y así con cada SO del dispositivo. Una vez descargado el software de la tienda online correspondiente sea de pago o no, la instalación es automática y el paso siguiente no es más que jugar. Cabe decir, que lo ideal es primero el padre o tutor comprobar las opciones básicas del juego por si tiene algún filtro parental y ya a continuación dejarle el dispositivo al niño para que disfrute aprendiendo.

A continuación, tras explicar cómo conseguir videojuegos en el dispositivo Smartphone (para el caso de las tabletas es el mismo proceso) se procede a mostrar dos ejemplos de esta otra vía, uno en un dispositivo *Android* y otro de *iOS*. Se han escogido estos dos por tener por tener una gran cantidad de descargas.

- *Juego de cocina para niños*



Imagen 24 Ejemplo 3 de “videojuego en smartphone”

Para jugar este juego, hay que pulsar el botón cocinar del menú principal y, como segundo paso, elegir qué plato cocinar. A partir de aquí, paso a paso de guía al niño a cocinar virtualmente un plato sano, cortando alimentos, pelando, cocinando y finalmente agrupándolos en el mismo plato listo para comer. Es un juego gratuito muy bueno con el que niño presta más atención a la hora de cocinar, fuera del juego el niño prestará atención a cómo se cocina en casa e intentará involucrarse con los alimentos por lo que resulta muy útil para acercar al niño con todo tipo de verduras, frutas y hortalizas.

Como puntos negativos, se puede comentar que, aunque para tratar los alimentos se usan fotos y el resto son elementos diseñados virtualmente con un aspecto más infantil, para ellos, esto hace que aunque lo reconozcan con más facilidad fuera del juego, se trate como un elemento áspero difícil de tratar, un elemento fuera de lugar. Este juego carece de una historia que introduzca al niño dentro de ese mundo y el usuario sienta la necesidad de que debe cocinar más porque así les puede ayudar y de este modo extrapolar ese vínculo de niño-cocinar-verduras al mundo real. Algo negativo es que la finalidad del juego y de lo que trata, es de cocinar y no de comer, por lo que lo que se le concienza al niño es que está bien cocinar alimentos, aprenda nombres de alimentos pero nada de comer.

- *Fruta y verduras*



Imagen 25 Ejemplo 4 de "videojuego en Smartphone"

Este juego consiste en, como resulta bastante evidente según la imagen correspondiente, plantar en tu propio huerto grandes cantidades de un amplio repertorio de verduras, hortalizas y árboles frutales. El niño comienza con unas pocas semillas y va cultivando verduras hasta que maduran y las puede vender y obtener un dinero. Con ese dinero, según la cantidad que tenga, el niño puede o bien comprar más semillas de las mismas o nuevas plantas o por otro lado puede invertir el dinero en hacer en ampliar la parcela de su huerto para poder cultivar más plantas a la vez. Es un juego muy interesante para niños con un mínimo de edad como 5 años que poco a poco van entendiendo el concepto de realizar un esfuerzo en conseguir algo, sacar un beneficio por ese esfuerzo y finalmente reinvertir ese beneficio. Esta idea es la misma que enseña el cuento de la lechera, principio básico del capitalismo, salvo que al ser un mundo virtual se puede llevar a cabo y ser rico virtualmente. Del mismo modo que enseña eso este juego, también ilustra gran cantidad de frutas, verduras a lo largo del proceso de sembrado, crecimiento, floración y recogida, para que el niño sepa cuál es el proceso de "creación" de las verduras que come en casa, pues no aparecen "por arte de magia".

Sin una idea y argumento que avance con el tiempo, como lo que sería el cuento de la lechera como tal, una narración que avance, el juego resulta bastante monótono y aburrido para un niño de esta edad, ya que no es demasiado animado. Para que este juego pudiese intentar resolver del todo el problema propuesto en este proyecto, debería tratar por comentarios o cambiar un poco la finalidad del juego para que fomente que los niños ingieran frutas y verduras de todo tipo debido a lo buenas y saludables que son.

Las referencias de toda la información aquí residente, se ha basado e inspirado en el siguiente enlace de la universidad de salamanca y experiencia personal:

http://campus.usal.es/~teoriaeducacion/rev_numero_02/n2_art_etxeberria.htm

2.7 Solución propuesta

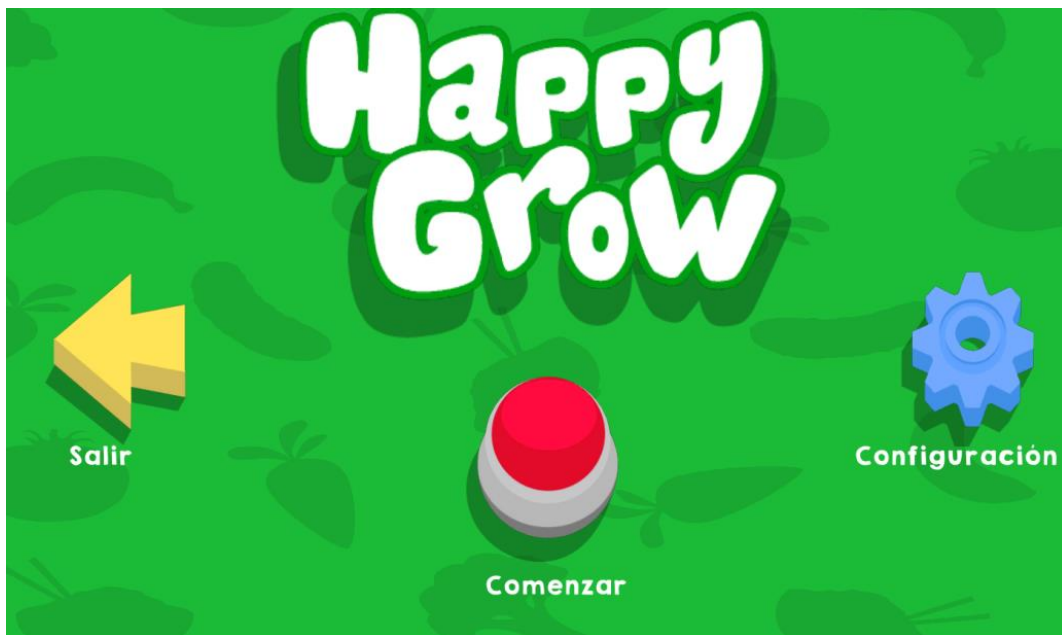


Imagen 26 Menú principal de “mi solución”

Finalmente llega el momento de tratar la solución propuesta, distinta y a la vez igual a las anteriores que intenta tomar lo bueno de cada una de ellas para poder satisfacer las necesidades de la sociedad. Esta solución consiste en un videojuego de ordenador basado sobre la idea de la **Exposición de un igual**, del mismo modo que la solución 4 hace para influenciar a algunos niños, y estos convencerán de la misma idea al resto. Este videojuego está pensado para jugarse en casa de forma autodidacta, adjunto con un simple manual de usuario, recibido de forma gratuita, una microhistoria motivante para el niño en su interior y una amplia información educativa que refuerza la idea de la ingesta de verduras. A continuación se va a explicar detalladamente cómo y por qué se cumple lo anterior.

Como paréntesis antes de explicar todo, se debe decir que aunque han existido muchas fuentes con datos que me han sido de utilidad, pero el documental estrella fue uno que emitió Telemadrid[2] hace tiempo y fue subido a YouTube donde fui visualizado y analizado en conciencia. Como información relevante secundario, también fueron de utilidad los consejos de los capítulos de la serie Supernanny[3].

2.7.1 Cómo la solución afronta el problema

En primer lugar, se va a explicar qué es exactamente la idea sobre la que se basa el juego y cómo se cumple, la exposición de un igual. Se define, “exposición de un igual” al comportamiento de persona frente un objeto, ser vivo, acto, idea, interés, del cual dicha persona está en **desagrado** pero existe un ente cercano físicamente, un **igual**, el cual muestra efusivamente su agrado por el mismo objeto, ser vivo, acto, idea, interés y contagia con su efusividad a la primera persona. Utilizando el ejemplo de la solución4 del taller de marionetas... si la historia consiste en un niño que ha crecido

sano y fuerte gracias a comer brócolis, ha cometido una gran hazaña y ha solucionado un gran problema debido a haber crecido así de sano y la moraleja de la historia es que es muy bueno comer brócolis, suponiendo que había un gran público de 20 niños en la sala seguramente haya dos posibles pensamientos al respecto. El primero, apodado conjunto A, el de la mayoría de los niños, es que digan su mala opinión y expongan que los brócolis saben mal, tienen una textura desagradable y que desea el exterminio de todos los brócolis del planeta. La segunda, apodado conjunto B, el de la minoría de los niños, aunque les pueda costar por vergüenza (la cual se ve disminuida porque comparte la opinión del héroe de la historia), dirán que ellos están orgullosos de comer brócolis, que les encantan y que de mayor ellos serán igual o más fuertes que el héroe de esa historia. Primero se suele manifestar la opinión mayoritaria del conjunto A y escasos segundos después, el conjunto B. Una vez B se ha manifestado, A, aunque este esté en mayoría casi absoluta empieza a sufrir los efectos de la exposición de un igual, esto es, al interno debate de... "Si a B, le encanta el brócoli como al héroe, ¿Por qué a mí no me gusta?" que suele acabar con la siguiente resolución emocional: "Yo voy a ser menos que B, así que a mí también me gusta el brócoli a partir de ahora". En personas mayores esta idea suele surtir menos efectos a medida que los ideales de las personas son más maduros, pero en niños pequeños, hasta el marco de los 12-13 años, siempre que se comparen con un igual, funciona bastante bien. Trátese de igual a una persona con características físicas/edad/creencia religiosa/etnia similar a la del niño que se quiere influenciar positivamente. A mayor público en una función de marionetas, más fácil es encontrar un pequeño conjunto de niños B que les encante el brócoli y puedan influenciar positivamente a la mayoría A de la idea. En un conjunto pequeño de 4-5 niños, si únicamente existen niños del conjunto A, la probabilidad de que un niño al llegar a casa y se enfrente a su terrible archienemigo, el brócoli, y el niño decide ingerirlo sin rechistar es bastante más baja. Los niños A influenciados por B, al llegar a casa, al menos lo intentarán y a algunos les acabará gustando o simplemente no les disgustará.

¿Cómo se aplica esto mismo al videojuego? De una forma, puédase llamar trucada, ya que se coloca al niño que juega enfrente de la pantalla, dentro del juego verá que tiene un avatar que lo identifique (cambia según el género, por lo que trata de ser su **igual**) y está rodeado de individuos con la mentalidad B, es decir, pseudoiguales (no son humanoides, por lo que no son iguales, pero tienen aspecto de niño). Estos están continuamente expresando su agrado por las verduras, por lo que le están influyendo durante todo el tiempo que el niño esté jugando, por lo que por *feedback* activo y pasivo, el niño recibe la idea de que comer verduras es muy importante para el crecimiento y puedes tener problemas si no lo haces.

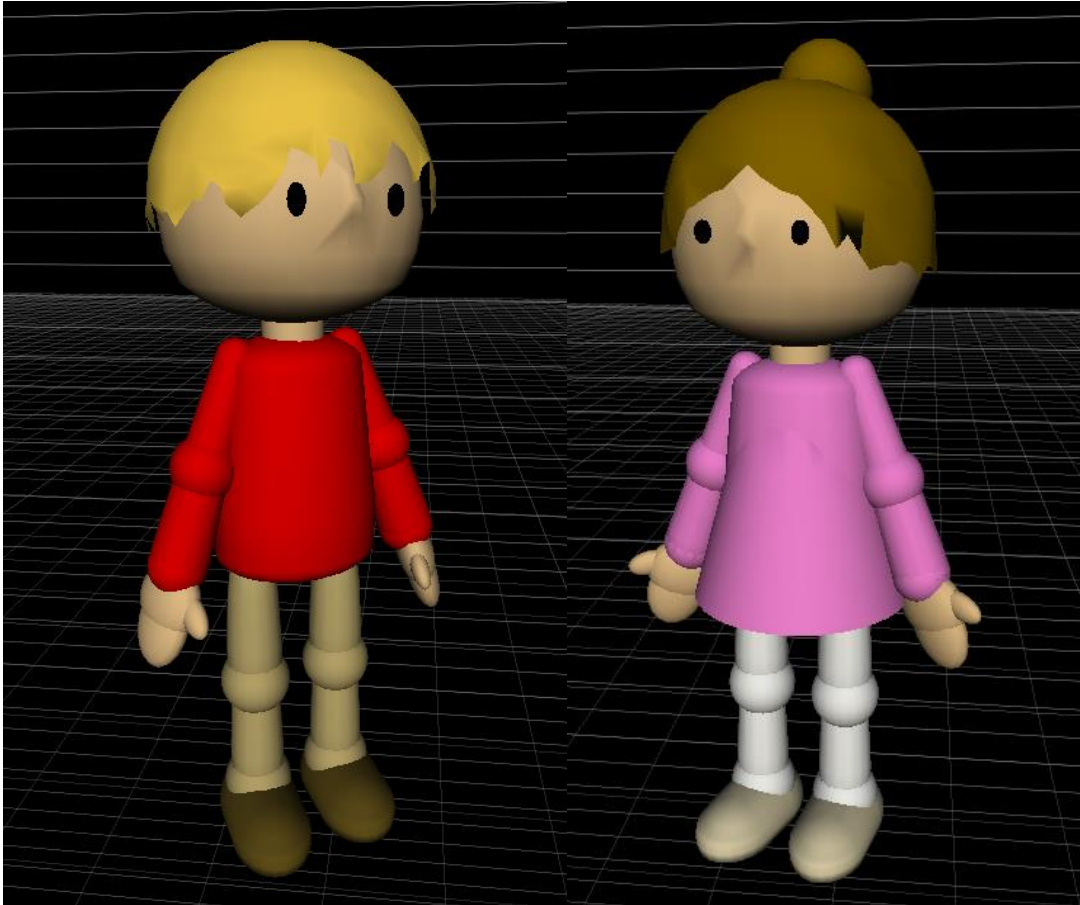


Imagen 27 Niño y niña en “mi solución”



Imagen 28 Duende verduriano en “mi solución”

En segundo lugar, tratando de la historia motivante, cabe decir que es bastante simple, corta pero que, sin embargo, cumple el propósito que tiene. El propósito es adentrar al niño en un mundo alternativo de fantasía en el que se sienta libre de poder hacer todo lo que quiera, donde puede jugar, aprender y disfrutar sin límites. Un lugar donde cada pieza se ha colocado estratégicamente para que el niño se fije en todas y en ninguna a la vez y le transmitan toda la información que deben, ya sea de motivación personal, motivación alimentaria, sentir agrado frente algo o desagrado frente otro punto. La historia trata de un niño que se adentra en un mundo de fantasía lleno de duendes amantes de las verduras y duendes amantes de los dulces donde estos últimos quieren atiborrar al avatar del niño de chucherías para que se le caigan los dientes y así no pueda volver a comer para que no pueda crecer grande y fuerte. Por otro lado, con la ayuda de los duendes verdurianos y todo su arsenal de comida saludable deben hacer frente a este nuevo enemigo para poder salvar tanto a su avatar tanto ese mundo fantástico hecho adrede para que le guste. Tanto la historia tanto el juego tratan de incentivar y reforzar el vínculo del niño con la comida saludable que, guste o no, es muy importante para el niño y debe comerla ya que sino no podrá crecer adecuadamente. En caso de fallar en la tarea asignada al héroe de fantasía, se insta a seguir intentándolo con un refuerzo positivo y, en caso de salir victorioso, de premia con una congratulación. Es muy importante que nunca haya un elemento que haga sentir incómodo al niño para no desincentivarle a seguir jugando.



Imagen 29 Imagen de derrota de "mi solución"

En tercer lugar, este juego tiene una gran cantidad de información útil que transmite al usuario que lo juega de forma visual, auditiva y la mecánica del propio juego. Se transmite de forma visual al ver a los distintos duendes verdurianos. Cada duende verduriano, se ha especializado en alguna verdura en concreto, se ha atiborrado de ella y ha crecido algo obteniendo todos los beneficios positivos de ello. Esto mismo refleja que el niño puede visualizar cómo podría ser él "dentro de poco" según qué verdura coma y puede imaginarse las mezclas de comer de todo pues lo

está viendo todo de manera individual. Del mismo modo puede ver que los duendes amantes de los dulces tienen un aspecto insano, con malos hábitos que no se recomiendan seguir. Se transmite de forma auditiva pues, durante el transcurso del propio juego, al inicio y al final, existen diferentes audios que, o bien narran la historia y motivan al niño, o bien redactan lo buenas que son las verduras e importantes que son en el día a día o bien explican detalles del juego como el porqué de la forma y atributos de los duendes verdurianos, como se acaba de comentar. Se transmite en la mecánica, pues se sugiere al niño que, todo lo que vaya a llegar a él, entonado como alimentos, es una lucha entre verduras y chucherías en la que, por el bien del propio niño, deben ganar las verduras para que él pueda crecer sano, sin ningún tipo de problemas sin tener que llegar a recurrir a la tediosa solución 5 del uso de medicamentos.

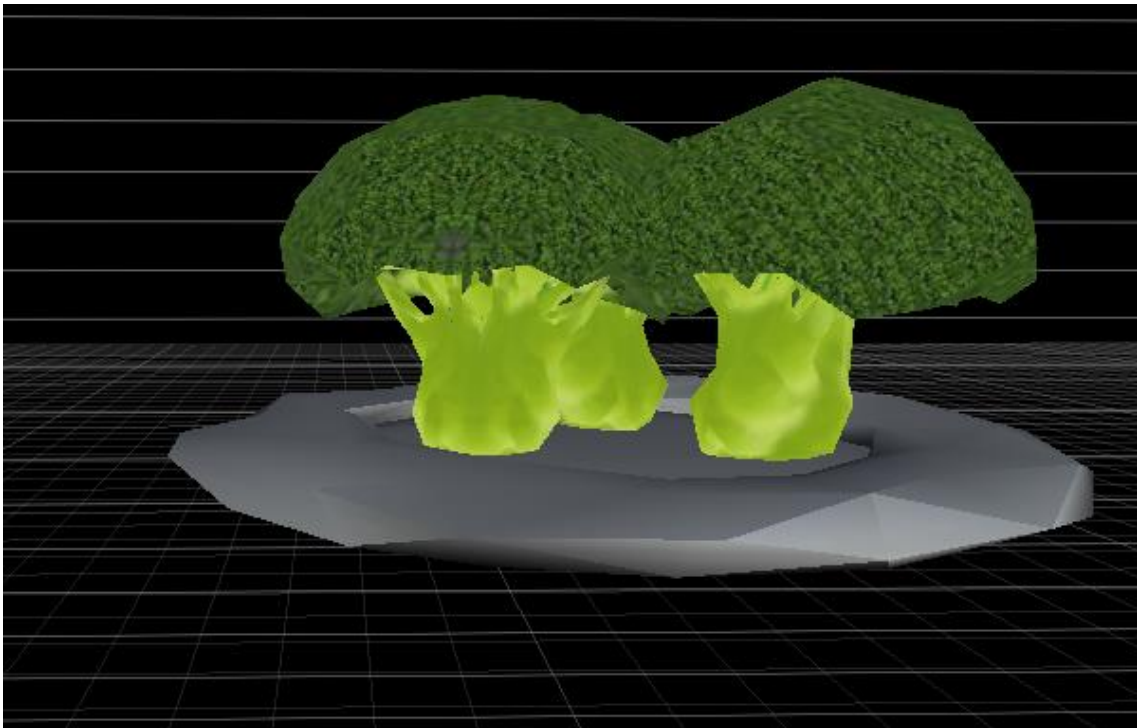


Imagen 30 Plato brócolis de "mi solución"

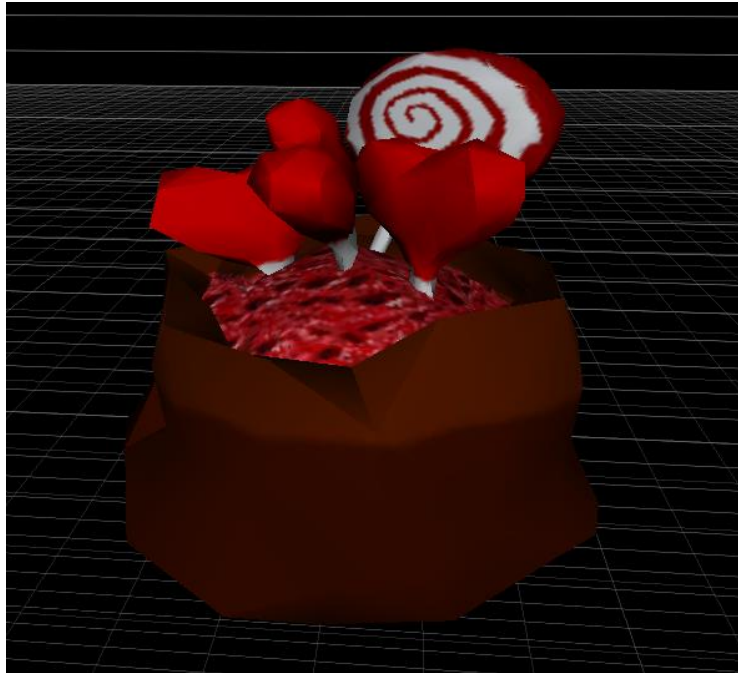


Imagen 31 Montón de piruletas de “mi solución”

El propio juego se explica a medida que se juega con los propios comentarios, pero aun así, existe un pequeño manual, un apéndice de este presente documento que explica tanto la instalación como el cómo se juega sin añadir ninguna complicación adicional.

En principio este videojuego es gratuito por lo que ninguna familia que lo pueda necesitar, debiera de sufrir ningún problema para poder adquirirlo. El formato es para PC, aunque en algún futuro puede que esté para Smartphone por lo que la difusión se facilitará más.

Para motivar a los niños a seguir jugando aunque se lo hayan pasado una vez, cada vez que juegan, los audios son reproducidos de forma aleatoria por lo que hasta que hayan escuchado todos seguramente tengan que jugar bastante tiempo. Los enemigos aparecen, dentro de las posiciones predefinidas, aleatoriamente en unas u otras, por lo que cada vez que juegan tienen otro factor en cuenta que nunca es el mismo.

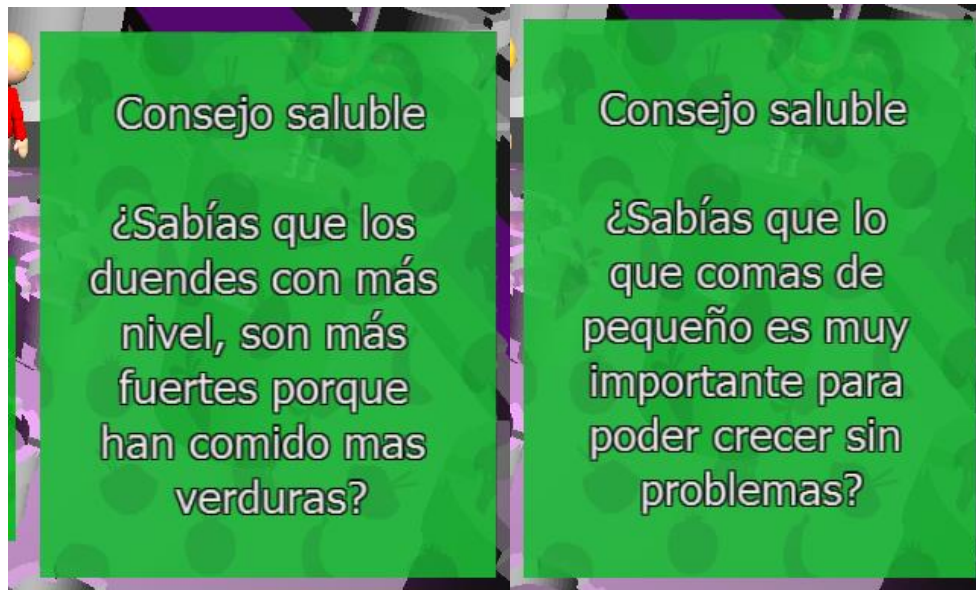


Imagen 32 Consejos saludables de “mi solución”

Una vez explicados los detalles exteriores del juego, se puede tratar la mecánica principal que los une, mecánica de un juego del género *Tower Defense* (a partir de ahora TD). Un TD consiste en un juego que el usuario ve normalmente desde arriba, como un señor viendo un tablero de ajedrez, donde ese terreno de juego tiene tres partes diferenciadas: Espacio de torres, recorrido y decorado. El género consiste en colocar torres en sus espacios correspondientes para que éstas puedan atacar a los enemigos, los cuales aparecerán en una punta del laberinto e intentarán llegar a la salida, e impidan que éstos lleguen a salir. También es muy común que no sólo sea una oleada, sino que a lo largo de entre 5 y 10 oleadas, el usuario cada vez vaya colocando más torres en lugares estratégicos mientras recibe un dinero virtual por aniquilar enemigos durante las oleadas, ya que las oleadas poco a poco son más complejas con enemigos más rápidos y resistentes. Se ha escogido este género porque a mí, cuando tenía esa edad, me fascinaban los juegos de estrategia de este tipo. Principalmente es por motivación personal y he querido transmitir la misma sensación de hallar la mejor estrategia con unos recursos limitados combinado con elementos que ayuden a fomentar intensivamente la solución de un problema de nuestros días.

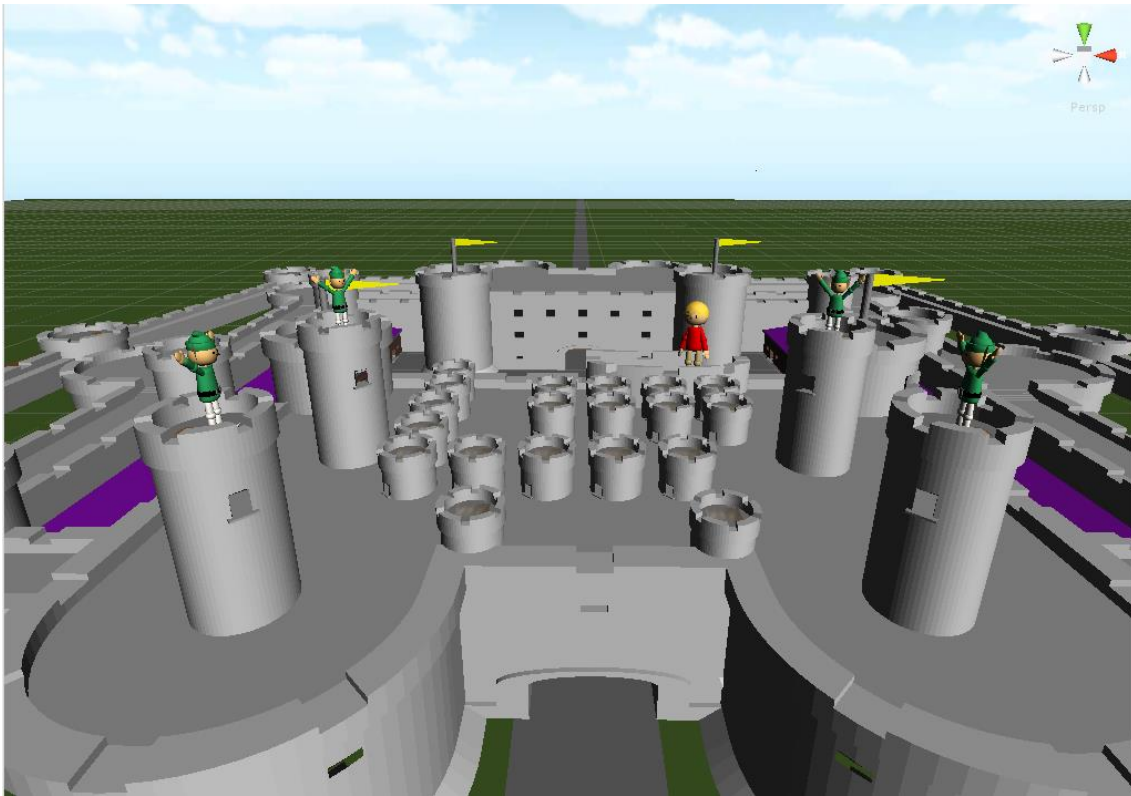
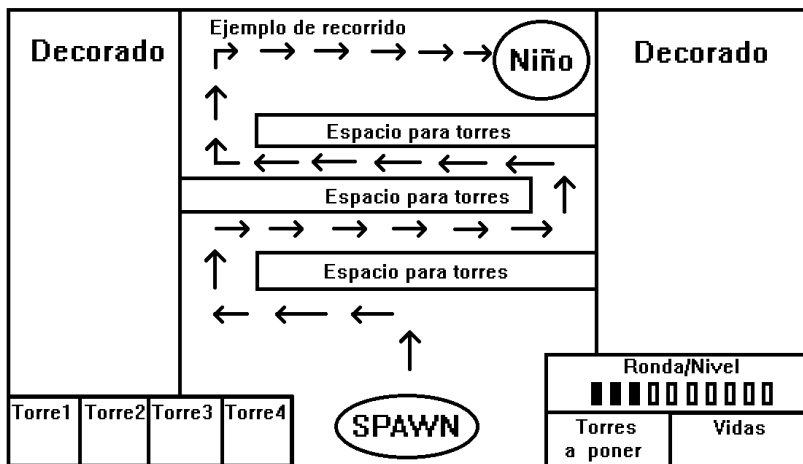


Imagen 33 Escenario1 TD de "mi solución"



- Spawn es el/los puntos por donde los enemigos aparecerán para iniciar el recorrido.
- Decorado no es más que el escenario, ninguna funcionalidad GUI. Cuando salgan mensajes emergentes aparecerán aquí.
- Niño es el lugar a proteger donde estará el niño 3D, el final del recorrido. Al acabar el nivel, este niño 3D se comerá todos las torres (platos con comida sana) y según lo que coma, pasarán unas cosas u otras.
- Ejemplo de recorrido no es más que un laberinto(uno de varios) simple a usar y recorrer por los enemigos.
- Espacio para torres son espacios donde el niño colocará las torres con las que defenderse.
- Torre1/2/3/4 es la parte de la interfaz donde están las torres usables en ese escenario. En cada escenario habrá distintas torres.
- Ronda/Nivel es un indicador visual con el que el niño puede saber lo que lleva de progreso en el escenario hasta que éste finalice. En este ejemplo el estado es ronda3 de 10.
- Torres a poner es un indicador visual para que el niño sepa cuando y cuantas torres puede poner. Como ejemplo, es un plato. Si tiene "dinero" para poder colocar 2 torres, se visualizarán 2 platos. Si tiene para 3, 3. Si lo gasta, no tendrá nada hasta que vuelva a ganar "dinero". El dinero será invisible para el niño, sólo verá cuando tiene para poner una torre. Todas las torres valen igual.
- Vidas es el indicador visual de la salud actual del niño. El indicador es una boca con 20 dientes(10 arriba 10 abajo, 1 vidas) y por cada enemigo que llegue al final 2 dientes (1 arriba, 1 abajo, 1 vida) se volverán amarillos. Cuando todos los dientes estén amarillos, con otro más se le caerán todos los dientes y perderá.

Imagen 34 Boceto inicial de interfaz 2D de TD de "mi solución"

A continuación se puede ver una tabla que contrasta los distintos aspectos del juego con los elementos del juego, su unión con un TD y su temática educativa con la alimentación:

	Elementos de juego	TD resuelve	Alimentación
Personajes	<u>Protagonista</u> El niño	El punto final del recorrido.	Según cómo juegues, todo afecta a este niño.
	<u>Antagonista</u> Rey de los duendes del azúcar	El jefe final de cada escenario.	Duende que lleva consigo una cantidad atroz de azúcar.
	<u>Enemigos</u> Duendes del azúcar	Los súbditos a eliminar que recorren el recorrido.	Duende que llevan consigo productos con altas cantidades de azúcar.
	<u>Amigos</u> Duendes verdurianos	Las torres a colocar para eliminar a los enemigos.	Duendes que llevan consigo productos saludables.
	<u>Consejera</u> Duende de consejos	La duende que dará consejos a lo largo de la partida.	Duende que comenta buenos hábitos alimenticios relacionados con el material del juego.
Escenarios	<u>Escenario1</u> Castillo fantasía	Escenario que recorrerán los duendes del azúcar para llegar al niño que tendrá huecos donde se podrán colocar los duendes verdurianos.	Nada.
	<u>Escenario2</u> Bosque de árboles frutales		
	<u>Escenario3</u> Desierto con pirámides(alimenticias)		
Mecánicas de juego	Seleccionar torre	Se selecciona un alimento del margen inferior izquierdo para posteriormente colocar un duende verduriano con él.	Al seleccionar el dibujo del alimento, se oye la pronunciación del mismo alimento.
	Duende verduriano	Se pueden colocar en las zonas	Estos duendes portan un

		permitidas.	alimento sano.
		Se puede mejorar el alimento que portan y será más poderoso.	El alimento aumenta en proporción o bien se cocina obteniendo así un “nuevo” alimento encima del plato.
	Empezar oleada	Un botón que sólo estará disponible al inicio y al final de cada oleada para empezar la siguiente.	Nada.
Sistema de puntuación	Vidas	Parte de la interfaz inferior derecha que se muestra con un dibujo que va cambiando según el número de vidas.	El dibujo es una sonrisa que representa de forma visual las vidas que quedan. Todas las vidas, los dientes blancos. A medida que se pierden vidas los dientes se quedan amarillentos. Al quedarte sin vidas toda su dentadura estará llena de caries.
	Torres	Parte de la interfaz inferior derecha que muestra con uno/varios dibujos para poder saber el número de torres que puedes colocar o mejorar. Por cada dibujo se puede o colocar un duende verduriano o cocinar/aumentar la comida de uno existente(mejorar)	El dibujo es el de un gorro con un número ya que los duendes verdurianos portan en sus cabezas un flamante gorro característico.
	Ronda	Parte de la interfaz inferior derecha que se muestra con	El dibujo de una pila que se va cargando. Esto

		diferentes barras verticales.	indica que si plantas cara a las chucherías comiendo verduras también puedes acabar lleno de energía.
	Resultado positivo	Este caso se visualizará en caso de haber superado correctamente todas las oleadas. El niño absorberá todos los alimentos que los duendes verdurianos lleven consigo y luego comerá dichos alimentos. Más tarde aparecerá un botón para volver al menú principal.	Se escuchará la frase: "¡Muy bien! ¡Has conseguido derrotar a los duendes malvados!". La escena es un niño debido a haber llevado una alimentación adecuada ha crecido grande, fuerte y da saltos de alegría por no haber sucumbido a la tentación.
	Resultado negativo	Este caso se visualizará en caso de haber perdido todas las vidas. Tras escuchar una frase con el mismo escenario pero el niño cambiado, aparecerá un botón para reintentar ese escenario y otro para volver al menú principal	Se escuchará la frase: "¡Oh vaya! ¡Los terribles duendes han podido contigo! Así no podrás crecer sano y fuerte ¡Prueba otra vez!". La escena es un niño que no ha podido aguantar a la tentación de comer chuches hasta no poder más y le pasa factura, i lustrado de una forma cómica para el niño.
Guión	En este juego tendrás que utilizar y mejorar alimentos saludables para poder comértelos y conseguir crecer fuerte y sano. Para ello tendrás que luchar con los duendes del azúcar dirigidos		

	<p>por su malvado rey pero con la ayuda de los duendes verdurianos en distintos lugares como un castillo de fantasía, un bosque frutal o hasta un desierto de pirámides podrás defenderte de ellos antes de que te obliguen a comer demasiados dulces y no puedas crecer. Todo esto será posible gracias al poder de los duendes sobre las verduras, frutas, hortalizas y demás alimentos para poder destruir las chucherías antes de que las lleven desde su guarida hasta tu avatar en el juego.</p>
--	--

Tabla 1 Contraste de aspectos del juego con sus elementos, mecánicas y la educación

Junto a la idea de resolver el problema de comer verduras, este videojuego también quiere transmitir las siguientes ideas enfocadas:

- No es bueno tomar muchas chucherías (dulce en general), se le ponen amarillos los dientes, le pueden salir caries y hasta se le pueden caer y provocar mucho dolor.
- Tomar verduras/frutas/hortalizas/legumbres es muy bueno, forman parte de la base de la pirámide alimenticia y ayudan al crecimiento y desarrollo del niño, alcanzando una vida saludable.
- Las verduras/frutas/hortalizas/legumbres no tienen por qué saber mal, si no lo prueban no lo pueden saber. Los alimentos que no destacan por ser dulces tienen sabores diferentes que hay que saber aceptarlos e integrarlos en la alimentación.

2.7.2 Datos extra obtenidos del estudio

En este pequeño apartado se mostrarán, en primer lugar, un conjunto de frases sacadas de documentales con los elementos más importantes a utilizar en cualquier solución para este problema. En segundo lugar, una tabla con los alimentos más comunes que debieran comer los niños para tener una dieta sana y equilibrada, sacados de los mismos documentales y qué propiedades buenas tienen cada uno. A continuación reside el conjunto de frases:

- Hay que recompensar las buenas acciones como poco con palabras de ánimo.
- Se debe usar la exposición de un igual: Si un niño, está junto a otro niño que sí come verduras, este tenderá a comerlas también.
- Hay que usar frases de ánimo: "Si sigues comiendo verduras y esas cosas, cuando crezcas tendrás unos músculos enormes".
- Es importante la exposición repetida: Probar poca cantidad pero con regularidad del alimento (de 8 a 10 veces). Al final puede que les guste o que simplemente lo odien menos.
- ¿Es útil racionar las chucherías? Cuanto más se restringa algo, es más posible que surja el "morbo por lo prohibido" y empeore las cosas. *En este juego se lucha contra las cantidades ingentes de chucherías pues en pequeñas cantidades no hacen daño.
- El azúcar es aquello que más llama la atención de los niños.

- ¿El azúcar te vuelve hiperactivo? No, el cerebro tiene una velocidad máxima de funcionamiento, por lo que al aumentar la cantidad de azúcar, en forma de glucosa únicamente prolongaría el tiempo de esa velocidad, la cual está lejos del hiperactivismo. Lo más importante es el ambiente en el que esté el niño.
- Aunque el azúcar no induzca al hiperactivismo, es un factor que daña los dientes y engorda a los niños por lo que hay que tratarlo con cuidado.
- ¿Los niños paran de comer cuando están llenos? No. Si se les carga el plato pueden comer de un 20% a un 70% que su ración correspondiente. Suele ser mejor que se lo sirvan ellos la comida saludable y pequeñas proporciones de postre o dulce.
- ¿Comer delante de la televisión fomenta comer más? Sí, porque la mente está distraída y tarda más en detectar el "saciado" por lo que aumenta la dosis de ingesta. Esto implica que tampoco es recomendable jugar al videojuego mientras se come, sino antes o después y durante la comida simplemente comer.
- Hay que conseguir que el niño siempre pruebe el menú que haya.
- No se deben poner raciones excesivas. Si los niños quieren repetir podrán hacerlo después.
- No llevar el postre a la mesa antes de terminar la comida. Así evitamos que los niños dejen de comer antes de tiempo y pidan el postre. Primero lo importante.
- Se debe establecer un tiempo limitado de comida (40min). A lo largo de la comida, merienda o cena no pedimos a los niños que se den prisa en comer, basta con que sepan que hay un tiempo para hacerlo.
- Cuando se esté terminando el tiempo de comer se les avisa para que sean conscientes de que si no terminan se retirará la comida.
- Si queremos que los más pequeños comiencen a comer lo mismo que el resto de la familia, hay que hacerles ver que ese momento es especial porque significa que se hacen mayores.
- Cuando el niño coma hay que decirle lo bien que lo ha hecho. De esta manera reforzamos el comportamiento para que lo repita.
- Aunque los niños no se lo coman todo desde el principio, cuando se inicia un hábito es más importante valorar el esfuerzo que los resultados, entre otras cosas, se empieza poco a poco.
- El niño debe comer por hambre, no por apetito, si al niño no le apetece comer verduras, su propio estómago rugirá y le pedirá que coma algo por hambre, instinto natural.
- Es importante a temprana edad utilizar castigos negativos y castigos positivos:
 - Castigo negativo: si el niño no se come la comida, el niño no verá la tele.
 - Castigo positivo: si el niño no se come la comida, entonces se tendrá que quedar sentado hasta que se lo coma.
- Lejos de los castigos, también son buenos los refuerzos, tanto positivos como negativos:
 - Refuerzo positivo: si el niño se come toda la comida, entonces tendrá helado de postre.
 - Refuerzo negativo: si el niño se come toda la comida, entonces no tendrá por qué comer verdura por la noche.

Para acabar este apartado, aquí está la tabla con la información de qué es lo bueno que tienen estos alimentos más comunes de una dieta sana y equilibrada para el crecimiento de los niños:

Alimento	Tipo	Propiedades	Bueno para	Referencias inmediatas
Zanahoria	Hortaliza	Calcio, hierro, potasio, fósforo y muy rica en vitamina A. También se encuentran en menos cantidad vitaminas B, C, D.	La vista.	http://www.culturizando.com/2012/10/10-beneficios-de-las-zanahorias.html http://www.clubplaneta.com.mx/cocina/zana.htm http://alimentos.org.es/zanahoria
Tomate	Verdura-fruta	Zinc, potasio, fósforo y rica en vitamina A. También se encuentran en menos cantidad vitaminas B, C, E.	El corazón.	http://verduras.consumer.es/documentos/hortalizas/tomate/intro.php http://www.culturizando.com/2014/06/confirmado-el-tomate-es-bueno-para-tu.html http://www.complejob.net/2011/06/beneficios-del-tomate.html http://alimentos.org.es/tomate
Champiñón	Hongo	Hierro, potasio, fósforo y vitamina B.	Crecimiento del pelo y uñas.	http://alimentosparacurar.com/n/2854/champinon-propiedades-y-beneficios.html http://www.natursan.net/champinon-es-beneficios-y-propiedades/ http://alimentos.org.es/champinones
Pepino	Verdura	Magnesio, potasio y rico en vitaminas B y C. También se encuentran en menos cantidad vitaminas A, E y K.	La piel.	http://www.culturizando.com/2014/01/10-beneficios-del-pepino.html http://salud.ellasabe.com/plantas-medicinales/133-pepino-propiedades-y-beneficios http://alimentos.org.es/pepino
Arroz	Cereal	Fibra, zinc, potasio, fósforo y rico en vitamina B. También se encuentran en menos cantidad vitaminas D y K.	Dar mucha energía.	http://www.lineayforma.com/nutricion/los-beneficios-de-comer-arroz.html http://ejerciciosencasa.es/beneficios-del-arroz-y-sus-propiedades/ http://alimentos.org.es/arroz
Apio	Verdura	Fibra, hierro,	Facilitar la	http://www.nutricion.pro/alimentos/

		potasio, fósforo y rico en vitaminas A, B, E y C en menos cantidad.	digestión Ir al baño. La piel.	el-apio-y-sus-beneficios/ http://alimentosparacurar.com/n/114/ beneficios-del-apio-para-la- salud.html http://alimentos.org.es/apio
Plátano	Fruta	Magnesio, potasio, rico en vitamina C y algo menos de B.	Crecimiento del pelo	http://salud.uncomo.com/articulo/cuales-son-los-beneficios-del-platano-6335.html http://www.paritarios.cl/especial_propiedades_ventajas_platano.htm http://alimentos.org.es/platano
Maíz	Cereal	Magnesio, potasio, fósforo y rico en vitaminas A, B y E.	Dar mucha energía	http://www.natursan.net/maiz-beneficios-y-propiedades/ http://lpcdedios.wordpress.com/2013/11/14/maiz-y-sus-beneficios/ http://alimentos.org.es/mazorca-maiz
Judías verdes	Legumbre	Fibra, potasio y rico en vitaminas B y E.	Ir al baño Crecimiento del organismo	http://www.mapfre.es/salud/es/cinformativo/alimentos-judias-verdes.shtml http://www.botanical-online.com/judiaspropiedadesalimentarias.htm http://alimentos.org.es/judias-verdes
Brécol	Verdura	Calcio, potasio, fósforo y rico en vitaminas A, B, C y sobre todo E, típica de alimentos verdes.	Crecimiento del organismo y del pelo. Para la piel.	http://ecolife.co/index.php?option=com_content&view=article&id=116:brocoli-16-razones-de-peso-para-consumirlo&catid=44:ecobienestar&Itemid=225 http://vivirsalud.imujer.com/2011/01/21/beneficios-del-brocoli-para-la-salud http://alimentos.org.es/brecol
Espinacas	Verdura	Hierro, potasio y vitaminas A, B, C, E y K.	Obtener fuerza.	http://www.zonadiet.com/comida/espinaca.htm http://salud.uncomo.com/articulo/cuales-son-los-beneficios-de-las-espinacas-10152.html http://www.culturizando.com/2012/11/10-beneficios-de-las-espinacas.html http://alimentos.org.es/espinacas
Guisantes	Legumbre	Fibra, magnesio, fósforo,	Para el crecimiento	http://www.buenasalud.net/2012/09/19/beneficios-de-los-guisantes.html

		vitaminas A, C y sobretodo B.	del cuerpo.	http://larifa.org/2013/05/beneficios-para-la-salud-de-los-guisantes-verdes-enteros.html http://alimentos.org.es/guisante-verde
--	--	-------------------------------	-------------	--

Tabla 2 Propiedades positivas de alimentos sanos de “mi solución”

2.8 Comparativa de las distintas soluciones y la propuesta

A continuación se puede observar una tabla que compara todas las soluciones con las características más notorias de cada una para poder analizar mejor la situación. Con una X están marcados los campos que no se cumplen mientras que con una V se marcan los campos que sí se cumplen.

Soluciones	Característica1 No requiere el tiempo de los padres	Característica2 No requiere un coste económico	Característica3 Capta fácilmente la atención del niño	Característica4 Puede tener un fin educativo específico	Característica5 El niño se involucra con la actividad
Solución 1	X	V	V	X	X
Solución 2	X	X	X	X	X
Solución 3	X	X	V	V	X
Solución 4	X	X	V	V	X
Solución 5	V	X	X	X	X
Solución 6	V	V	V	X	V
Propuesta	V	V	V	V	V

Tabla 3 Comparativa de soluciones y la propuesta

3 Análisis, diseño, implementación e implantación

Este capítulo está dividido en cinco apartados donde se desarrolla la parte técnica del proyecto, habiendo dejado atrás la parte teórica. A continuación se trata con detenimiento tanto la metodología y circundantes, luego en el análisis una cantidad moderada de requisitos del sistema para llegar al diseño estructural con los casos de uso y diagramas de navegación para acabar con la implementación del sistema y su implantación.

Antes de comenzar, se debe decir que el aprendizaje de toda la herramienta Unity se realizó mediante el estudio de los video tutoriales de su propia web[4] y más en profundidad de la web de Unity Cookie[5].

3.1 Metodología, ciclo de vida y marco regulador

- Metodología

Debido al tema inicial del proyecto, poco definido que poco a poco se fue matizando con el tiempo hasta concretarse hasta la idea que trata esta memoria, se optó por llevar a cabo el uso de una metodología ágil[6], pues se fue avanzando con pequeñas tareas con una duración de entre 1 y 2 semanas. Dentro de la metodología ágil, la utilizada ha sido la programación extrema, dado que el proyecto ha ido cambiando sobre la marcha. Inicialmente la idea era únicamente un videojuego como herramienta para solventar un problema. A medida que pasó el tiempo se fueron creando y modificando los requisitos a medida que la idea final se iba acotando, siempre de cara a satisfacer al usuario final, que en este caso sería el cliente que fuese a utilizar el producto. La opinión del cliente, ya que éste forma parte del equipo es el centro de la metodología ya que asegura que en la recta final, éste está satisfecho con lo realizado.

- Ciclo de vida

En este micro apartado se detallan las clásicas fases del ciclo de vida del proyecto de un videojuego[7]. Aunque no se han seguido muy fielmente estas fases a lo largo del proyecto, es debido a que no el ciclo de vida de un videojuego no es unívoco para planificar y desarrollar un videojuego, lo que no quita que sean muy recomendables. Éstas son genéricas por lo que sirven de guía para todo tipo de proyecto de un videojuego.

Conceptualización: Esta fase tiene por objetivo definir el concepto del juego empezando por una base sacada de una lluvia de ideas. A partir de ahí se empieza a maquetar y definir la idea del juego junto a los bocetos 2D básicos teniendo muy presente el estado de la cuestión. El siguiente punto de la conceptualización, es la decisión del género y la ambientación. Una vez definidos estos primeros parámetros, se puede pasar a la siguiente fase.

Pre-Producción: Esta siguiente fase consiste en estudiar la viabilidad del proyecto y se decidirá si puede ser o no rentable y si se llevará o no a cabo. Se debe definir completamente la fase de producción con la ayuda de todos los integrantes del equipo. Para ello debe definirse para producción, un estudio del juego completo incluyendo datos más detallados como personajes principales, la ambientación o las interfaces iniciales para el usuario.

Producción: A lo largo del desarrollo de esta fase se acabará de construir el esqueleto principal del juego con todas sus ramas. Durante este proceso, se van necesitando no solo desarrolladores sino también gente que pruebe y dé el visto bueno a todo el material realizado, a esta gente se les llama los tester, o testeadores. Una vez se tiene un esqueleto entero montado mínimamente sostenible, se procede a avanzar a la fase alfa.

Alfa: Durante esta fase de un videojuego, en principio se pueden probar todas las funcionalidades que debe cumplir. Es posible que muchos detalles de la interfaz, o modelos no estén completos y que existan numerosos bugs de sistema. Según el juego que se desarrolle, pueden existir también peligros físicos ya que aún puede mantener bucles infinitos sin control y algún equipo pueda sufrir daños.

Beta: La penúltima fase queda alcanzada cuando el esqueleto está completamente vestido y sólo faltan algunos pequeños agujeros o balanceos en las mecánicas internas. Se considera el juego acabado y listo para pasar a última fase cuando ya no se encuentran bugs y se cumplen todos los requisitos que el cliente final pueda solicitar.

Liberación: Para acabar un juego, el último paso es congelar la versión del código dado que se considera acabado y esta misma versión empieza a ser distribuida al público de forma comercial o gratuita.

Actualizaciones: Desde que las plataformas de los juegos están unidas a la web, las fases alfa y beta se aceleran dejando en muchas ocasiones bugs sin solucionar o funcionalidades poco probadas con la finalidad de acabar antes el trabajo y lucrarse antes de él. Para solucionar estos problemas los creadores implementan parches para solucionar estas incidencias. Cuando no sólo se tapan agujeros sino que también se quiere (o si sólo se quiere) ampliar el contenido del propio juego, se lanza al mercado una actualización con esos pequeños datos que debían estar incluidos al principio, pero no se hizo por haber querido salir antes.

- Marco regulador

En este punto hay varias normas que se deben atender. En primer lugar, la Ley Orgánica 15/1999, de 13 de diciembre, de Protección de Datos de Carácter Personal^[8]. Esta norma regula el tratamiento de los datos recogidos, con la

necesidad de obtener el consentimiento expreso del afectado informando de los datos recogidos y del uso que van a tener.

Respecto a la propiedad de todos los contenidos y creaciones utilizadas en el proyecto, debemos referirnos al Real Decreto Legislativo 1/1996, de 12 de abril, en el que se aprueba el texto refundido de la Ley de Propiedad Intelectual[9]. En concreto, se regula la protección de los programas de ordenador a partir del artículo 95 de esta norma. En este sentido no sólo se protegen las funciones o normas creadas con dicho fin si no la documentación añadida a este trabajo. Se debe decir en este sentido que tanto la documentación como la programación propiamente dicha es totalmente original.

En la misma norma se indica la obligatoriedad de que el contenido audiovisual emitido goza de una protección determinada. Siendo cociente de esto, todos los contenidos audiovisuales empleados han sido obtenidos originalmente con sus correspondientes contratos de cesión de la titularidad o, si como en varios casos; se han recogido de bancos de muestras gratuitas o con licencia Creative Common. Abundando en la propiedad intelectual, se debe decir que el software utilizado para la generación del código cuenta con la preceptiva versión licenciada para la creación del trabajo presentado.

Por otro lado, es importante señalar que se ha elegido prescindir de publicidad y otros contenidos por lo dispuesto en la Ley 34/1988, de 11 de noviembre, General de Publicidad[10]. Se considera publicidad ilícita, entre otras, la que incita a un menor a la compra explotando su inexperiencia o incredulidad. Estando el juego dirigido a menores en edad temprana, la publicidad a la que se les somete debe estar muy controlada, motivo por el que se ha eliminado.

3.2 Análisis

El análisis del proyecto se realizó durante la fase de pre-producción del videojuego como se acaba de comentar, donde se sacaron las principales funcionalidades y se empezaron a generar los requisitos de usuario y requisitos de sistema.

Estas funcionalidades principales, que debido a la metodología han ido modificándose interiormente hasta llegar al fin del proyecto, son las siguientes:

- **Iniciar el juego:** Se podrá usar ejecutar en un PC.
- **Navegar por el menú:** Tendrá un menú principal con opciones de opciones, salida y comenzar.
- **Jugar en un escenario:** Deberá tener toda la mecánica de un TD adaptada con temática educativa.
- **Ganar:** Tendrá un escenario en caso de victoria que te deje volver al menú principal o salir.
- **Perder:** Tendrá un escenario en caso de derrota que te deje volver al menú principal, reintentar o salir.

A continuación se pueden encontrar los requisitos de estas funcionalidades. El formato utilizado para la obtención y representación de dichos requisitos ha sido inspirado por documento del gobierno *Proceso de Análisis del Sistema de la Información de Métrica v3*[\[11\]](#) cuyo enlace se puede encontrar en la bibliografía. Antes de mostrarlos, primero se va a proceder a explicar el formato utilizado junto a un ejemplo de tabla de requisito y acto seguido se mostrarán primero los requisitos de usuario para continuar con los requisitos del sistema y acabar esta sección con la matriz de trazabilidad de los requisitos.

Para cada tabla de requisito, existirán los siguientes campos con el siguiente significado:

- **Código de requisito:** Para poder identificar unívocamente el tipo de requisito y a qué hace referencia es necesario usar un código antes del propio nombre. El código comenzará por las letras REQ, seguido de un guión, continuado por las letras HG que hacen referencia al videojuego, seguido de otro guión, una letra para diferenciar usuario de sistema, otro guión y dos letras para diferenciar el tipo de requisito y acabado en un número para indicar el orden. Las letras para diferenciar el requisito serían SF para requisitos funcionales, NF para requisitos no funcionales, U para requisitos de usuario, S para requisitos de sistema. Los números finales comenzarán por 01 incrementándose en una unidad por cada requisito de ese tipo.
- **Nombre de requisito:** Un nombre que sirva de resumen, simple e identificativo acerca de qué trata dicho requisito.

- Descripción: Aquí residen los datos principales del requisito y lo que se debe o no se debe hacer.
- Prioridad: En aras de la correcta realización en orden de funcionalidades, hay unos pasos que se deben hacer antes y otros después, por lo que los primeros pasos siempre tendrán una mayor prioridad que los últimos detalles. Los posibles valores para este campo son: Alta, media, baja.
- Necesidad: Este campo indica si el requisito es referente a algo vital para resolver el problema del proyecto o si no es necesario para el problema pero es una parte de la mecánica del TD. Los posibles valores para este campo son: Sí, no.
- Fuente: Este campo indica el origen de la idea del requisito.
- Modificable: Aquí se indica si este requisito es modificable debido a algo que se pueda mejorar, o si es parte de la idea inquebrantable principal. Los posibles valores para este campo son: Sí, no.

Finalmente aquí está la tabla de ejemplo de requisito:

REQ-HG-S-SF 01: Ejemplo de requisito.			
Descripción	Todos los requisitos de sistema, ya sean funcionales, no funcionales o negativos tendrán que tener una estructura idéntica a esta aquí residente.		
Prioridad	Alta	Necesidad	Si
Fuente	Métrica V3	Modificable	No

Tabla 4 Ejemplo de requisito

3.2.1 Requisitos de usuario

Los requisitos de usuario son aquellos creados por el cliente y que se encargan de definir todo aquello que el usuario final ve. Dentro de cada requisito de usuario en este documento encontraremos requisitos funcionales, que dicen qué se ha analizado, y requisitos no funcionales, que restringen el cómo.

REQ-HG-U-SF 01: Arranque del juego.			
Descripción	El juego debe tener un archivo ejecutable que al iniciarse ejecute la aplicación.		
Prioridad	Alta	Necesidad	Si
Fuente	Cliente	Modificable	No

Tabla 5 Requisito de usuario 01

REQ-HG-U-SF 02: Comenzar a jugar.

Descripción	Desde el menú principal se debe poder acceder a todos los escenarios de juego del TD.		
Prioridad	Alta	Necesidad	Si
Fuente	Cliente	Modificable	No

Tabla 6 Requisito de usuario 02

REQ-HG-U-SF 03: Control de opciones.

Descripción	Desde el menú principal, se debe de poder desactivar y reactivas toda fuente de audio para el usuario.		
Prioridad	Baja	Necesidad	No
Fuente	Usuario	Modificable	No

Tabla 7 Requisito de usuario 03

REQ-HG-U-SF 04: Salir del juego.

Descripción	Desde el dentro del juego se debe poder salir de la aplicación.		
Prioridad	Alta	Necesidad	Si
Fuente	Cliente	Modificable	No

Tabla 8 Requisito de usuario 04

REQ-HG-U-SF 05: Elección de género.

Descripción	Se debe poder elegir el género del infante antes de empezar a jugar en un escenario.		
Prioridad	Media	Necesidad	Si
Fuente	Usuario	Modificable	No

Tabla 9 Requisito de usuario 05

REQ-HG-U-SF 06: Decisiones en el juego.

Descripción	Dentro de cada escenario, el usuario podrá decidir las torres que quiere usar dentro de un repertorio existente.		
Prioridad	Media	Necesidad	Si
Fuente	Cliente	Modificable	Si

Tabla 10 Requisito de usuario 06

REQ-HG-U-SF 07: Visionado de interfaz.

Descripción	Dentro de cada escenario, el usuario podrá ver todos los datos necesarios para poder jugar correctamente.		
Prioridad	Media	Necesidad	Si
Fuente	Cliente	Modificable	Si

Tabla 11 Requisito de usuario 07

REQ-HG-U-SF 08: Consejos educativos.

Descripción	Dentro de cada escenario, el usuario a medida que juega, debe poder escuchar consejos educativos referidos a una dieta sana y equilibrada que favorezca el correcto crecimiento de niños en su etapa de crecimiento.		
Prioridad	Media	Necesidad	Si
Fuente	Cliente	Modificable	no

Tabla 12 Requisito de usuario 08

REQ-HG-U-SF 09: Resolución tras el escenario.

Descripción	Al usuario que esté jugando se le debe informar al momento en caso de que llegue a superar exitosamente o de no superar el escenario al que juegue.		
Prioridad	Alta	Necesidad	Si
Fuente	Cliente	Modificable	No

Tabla 13 Requisito de usuario 09

REQ-HG-U-NF 01: Tamaño de interfaz adecuado.

Descripción	Dentro de cada escenario, la interfaz permanente se posicionará en los bordes de la pantalla y nunca ocupará más del 30% de la pantalla.		
Prioridad	Media	Necesidad	Si
Fuente	Cliente	Modificable	No

Tabla 14 Requisito de usuario 10

REQ-HG-U-NF 02: Elección de la resolución.

Descripción	La resolución del juego debe ser de 800x600 para el correcto visionado de la interfaz en su marco establecido.		
Prioridad	Media	Necesidad	Si
Fuente	Usuario	Modificable	Si

Tabla 15 Requisito de usuario 11

REQ-HG-U-NF 03: Profundidad en las decisiones del usuario.

Descripción	Todas las torres situadas en el espacio para torres tendrán únicamente una posible evolución con los atributos principales un 50% más alto.		
Prioridad	Baja	Necesidad	No
Fuente	Usuario	Modificable	No

Tabla 16 Requisito de usuario 12

3.2.2 Requisitos de sistema

Los requisitos de sistema son aquellos creados por el analista y que se encargan de definir todo aquello que el sistema necesita. Dentro de cada requisito de sistema en este documento encontraremos requisitos funcionales, que dicen qué se ha analizado, y requisitos no funcionales, que restringen el cómo.

REQ-HG-S-SF 01: Reconocimiento de periféricos.

Descripción	Cuando la aplicación arranque, ésta debe reconocer la existencia y señales de entrada y salida tales como ratón, teclado y pantalla principal.		
Prioridad	Alta	Necesidad	Si
Fuente	Desarrollador	Modificable	No

Tabla 17 Requisito de sistema 01

REQ-HG-S-SF 02 Lenguaje de programación.

Descripción	Todo el lenguaje de programación del proyecto excepto librerías externas debe estar programado en JavaScript de Unity3D.		
Prioridad	Alta	Necesidad	Si
Fuente	Desarrollador	Modificable	No

Tabla 18 Requisito de sistema 02

REQ-HG-S-SF 03: Formato de audios.

Descripción	Todos los audios utilizados deben tener el formato .wav para el correcto funcionamiento del código.		
Prioridad	Baja	Necesidad	Si
Fuente	Desarrollador	Modificable	Si

Tabla 19 Requisito de sistema 03

REQ-HG-S-SF 04: Formato de modelos 3D.

Descripción	Todos los modelos 3D utilizados deben tener formato .fbx o .obj tengan o no animaciones en su interior.		
Prioridad	Alta	Necesidad	Si
Fuente	Desarrollador	Modificable	Si

Tabla 20 Requisito de sistema 04

REQ-HG-S-SF 05: Formato de texturas e interfaz.

Descripción	Todas las texturas de modelos como las de las botoneras o de fondo de interfaz deben tener el formato de .png.		
Prioridad	Baja	Necesidad	Si
Fuente	Desarrollador	Modificable	Si

Tabla 21 Requisito de sistema 05

REQ-HG-S-NF 01: Tamaño de archivos.			
Descripción	Ningún archivo que se importe a Unity3D debe ocupar más de 3Mb para agilizar la carga de datos.		
Prioridad	Media	Necesidad	Si
Fuente	Desarrollador	Modificable	Si

Tabla 22 Requisito de sistema 06

REQ-HG-S-NF 02: Refresco de interfaz.			
Descripción	La interfaz debe refrescarse a cada frame para así evitar irregularidades fuera de lugar en caso de una acumulación de acciones consecutivas.		
Prioridad	Alta	Necesidad	Si
Fuente	Desarrollador	Modificable	Si

Tabla 23 Requisito de sistema 07

REQ-HG-S-NF 03: Sistema operativo.			
Descripción	La aplicación debe funcionar perfectamente para equipos con Windows7 o superior.		
Prioridad	Alta	Necesidad	Si
Fuente	Desarrollador	Modificable	No

Tabla 24 Requisito de sistema 08

3.2.3 Problemas varios del análisis

Las funcionalidades que más ha costado implementar han sido principalmente las de la propia mecánica del juego ya que se ha partido de tutoriales sin conocimiento previo de la materia. De todos los problemas, los más llamativos han sido:

- **Marcado de objetivo de una torre:** Esto prácticamente es el principio básico de un TD, es decir, tener un objeto torre fijo que cuando se acerca un enemigo automáticamente la torre lo detecte al entrar en su rango de acción, se gire hacia el enemigo y se prepare para disparar. Es algo relativamente sencillo, pero aunque este es el primer paso, no es fácil sin experiencia previa.
- **Unificado de sistema-interfaz:** Esta ha sido prácticamente la parte más difícil de realizar, pues cuando pasas de tener objetos sueltos que realizan objetos por un lado, una interfaz desplegable por otro y llega el momento de hacer que al pulsar la interfaz, se generen objetos en las coordenadas oportunas, y los recursos se modifiquen correctamente. Desde luego que esto no fue así, ya que en un inicio no se crearon objetos y cuando lo hicieron, tampoco lo hacían donde debían. Esta fue la parte más difícil de todas de encajar.
- **Sistema de pathfinding:** Debido a que toda la parte de inteligencia artificial ha sido descargado de un módulo extra, la incorporación fue demasiado complicada porque los posibles recorridos generados eran nulos. La división del

terreno en nodos de un tamaño suficiente como para poder ser atravesado fue la parte más importante para poder solucionar este problema y los enemigos no se empezasen a bloquear en el mismo punto porque atravesaban los nodos en diagonal, sin llegar al punto central de cada nodo, lo que provocada la colisión con el escenario.

3.3 Diseño

A lo largo de esta sección se van a exponer las distintas herramientas que se han usado para la construcción de este proyecto y aquellas que han sido valoradas junto a la argumentación de por qué han sido rechazadas. Tras la comparativa, está incluido el diagrama de casos de uso, escenarios y el diagrama de navegación de la aplicación.

3.3.1 Comparativa de herramientas

Para poder nombrar las herramientas utilizadas, primero se ha de dividir este tema en, apartado de audio, modelos 3D, imagen y motor gráfico, pues son temas completamente distintos entre sí con alternativas distintas dentro de cada sub-apartado.

- Audio

Se hace referencia a “Audio” a todo archivo de audio o vídeo con audio usado en el proyecto, que se ha generado, ha sido alterado mediante el uso de un software. A continuación se pueden ver las herramientas utilizadas y sus alternativas contempladas, si las hubiese:

- Audacity[12]



Imagen 35 Herramienta audio: Audacity

Este software libre de código abierto desarrollado por un grupo de voluntarios, es un editor de audio libre, fácil de usar para múltiples plataformas en varios idiomas.

Las principales funcionalidades son: grabación de audio, importación y exportación de audios a distintos formatos, edición, inserción de efectos y análisis de espectro.

Este programa ha sido utilizado únicamente para recortar y añadir pequeños efectos a audios simples. Ya se tenían nociones básicas de este software.

- Adobe Audition CC[13]

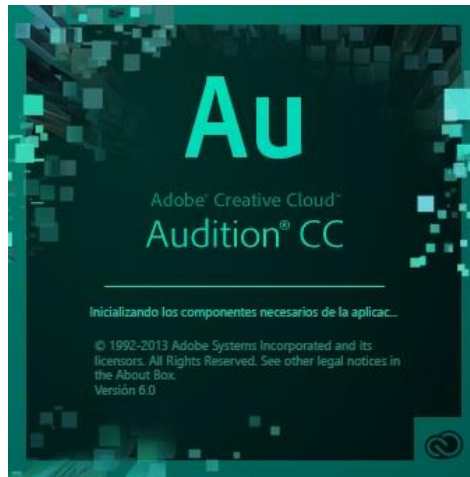


Imagen 36 Herramienta audio: Adobe Audition CC

Este software de pago (24,19€/mes o a partir de 300€ la licencia) desarrollado por Adobe es un grabador y editor de audios, similar a Audacity aunque mucho más potente. Es un programa multiplataforma para combinar con el resto de la familia Adobe CC.

Las principales funcionalidades son: grabación mono/estéreo desde un dispositivo externo, edición de audio con funciones de forma de onda, visualización espectral y multipista. También tiene la sincronización de configuraciones de Adobe con su nube junto a la de dispositivos móviles.

Este programa no ha sido utilizado debido a que, aunque era el programa que juntaba todas las funcionalidades que se necesitan y se han acabado cumpliendo con varios programas, era el único que no se sabía usar. Ante la gran cantidad de tiempo que se podría perder (dejando a un lado el coste económico) para unas funciones tan básicas, se decidió utilizar otras herramientas similares, incluyendo otra de la familia Adobe que se verá a continuación.

- Camtasia Studio[14]



Imagen 37 Herramienta “audio”: Camtasia Studio

Este software de pago (135€ la licencia) con versión trial disponible desarrollado por TechSmith es un creador y editor de vídeos (audio e imagen en el tiempo) multiplataforma que destaca por permitir la funcionalidad de grabar la propia pantalla del ordenador vía screencast o plug-fin.

Las principales funcionalidades son: grabación de todo lo mostrado en la pantalla, visualizado y edición ligera de videos en dispositivos móviles y, principalmente, edición de vídeo con todo tipo de efectos manipulando independientemente vídeo de audio.

Este programa ha sido utilizado en pequeña escala para grabación de la pantalla para vídeos. Principalmente se han utilizado únicamente los audios de dichos vídeos. Ya se tenían nociones básicas de este software.

- Adobe Premiere Pro CC[15]



Imagen 38 Herramienta “audio”: Adobe Premiere Pro CC

Este software de pago (24,19€/mes a partir de 300€ la licencia) desarrollado por Adobe es un grabador y editor de vídeos, similar a Camtasia aunque mucho más potente, también multiplataforma con posibilidad de aumentar más aún sus posibilidades con programadas de Adobe adjuntos como Adobe After Effects CC.

Las principales funcionalidades son: grabación desde un dispositivo externo, edición de vídeo con un extenso repertorio de efectos de audio y vídeo, sincronización de configuraciones con la nube junto a la de dispositivos móviles.

Este programa ha sido utilizado únicamente para modificar efectos de audio de los vídeos grabación con Camtasia debido al gran repertorio y potencial del software. Ya se tenían nociones avanzadas de este software y debido a eso su utilización.

- Modelado 3D

Se hace referencia a “Modelado 3D” a todo archivo con datos de un objeto 3D, texturas de un objeto 3D, materiales del objeto y animaciones de objetos generadas o modificadas mediante el uso de un software. A continuación se pueden ver las herramientas utilizadas y sus alternativas contempladas, si las hubiese:

- Sculptris[\[16\]](#)



Imagen 39 Herramienta modelado3D: Sculptris

Este software desarrollado por Pixologic es una herramienta multiplataforma de modelado gráfico 3D para crear todo tipo de modelos a partir de una esfera. Es un software muy sencillo que gran potencial.

Las principales características son: creación de esferas para modificar, modelar en modo simétrico o asimétrico, importar/exportar datos de/a otros formatos, introducir texturas a estas esferas modificadas, importar/exportar textura de/a otros formatos.

Este programa se ha utilizado para modelar todos los elementos no humanoides y los no lisos, es decir, todas las frutas, verduras, chucherías y caramelos. Aunque no se tenía ninguna noción de cómo usar este programa, es realmente sencillo y por intuición puedes llegar a hacer grandes proezas.

- Cinema4D[17]



Imagen 40 Herramienta modelado3D: Cinema4D

Este software de pago (695\$ la licencia) con versión trial accesible desarrollado por Maxon es una herramienta multiplataforma de modelado y animación 3D para crear, modelar y animar modelos a partir de formas básicas. Este software no es algo más complicado pero sólo se han usado muchas opciones básicas.

Las principales características son: creación de formas básicas para modificar, modelador es modo simétrico o asimétrico, importar/exportar datos de/a otros formatos, introducir texturas a estas formas modificadas, importar/exportar textura de/a otros formatos, introducir esqueletos a las formas y añadir estos esqueletos.

Este programa se ha utilizado para modelar y animar a todos los humanoides y elementos lisos, es decir, todos los duendes, el niño, la niña y el decorado. Aunque no se tenía ninguna noción básica de este software, por una recomendación personal se aprendieron a crear y animar formas muy básicas, suficientes para obtener los resultados esperados.

- Blender[\[18\]](#)



Imagen 41 Herramienta modelado3D: Blender

Este software gratuito multiplataforma desarrollado por Blender Foundation es una herramienta de modelado y animación 3D para crear, modelar y animar modelos 3D. Este software tiene una curva de aprendizaje notoria, siendo bastante arisco al inicio.

Las principales características son: modelado, introducir esqueletos, animación, simulación y renderizado de modelos/animaciones. De todo esto, se pueden importar/exportar el material de/a formatos básicos.

Este programa no se ha utilizado debido a su complejidad aunque su potencial es excelente.

- 3DSMax[\[19\]](#)



Imagen 42 Herramienta modelado3D 3DSMax

Este software de pago (195€/mes o a partir de 3900€ la licencia permanente) propiedad por Autodesk es una herramienta multiplataforma de modelado y animación 3D para crear, modelar y animar en 3D. Este software de pago tiene una curva de aprendizaje alta, siendo complicado el inicio.

Las principales características son: modelado, animación, simulación y renderización de modelos/animaciones. De todo esto, se pueden importar/exportar el material de/a formatos básicos.

Este programa no se ha utilizado debido a su complejidad y precio aunque su potencial es excelente.

- Maya[20]



Imagen 43 Herramienta modelado3D: Maya

Este software de pago (195€/mes o a partir de 3900€ la licencia permanente) propiedad de Autodesk es una herramienta extremadamente similar a 3DS MAX, ya que en un inicio fueron competencia.

Las principales diferencias son la distinta interfaz gráfica que han mantenido y los métodos para concluir un proyecto.

Este programa no se ha utilizado por las mismas razones que su hermano de Autodesk.

- Imagen

Se hace referencia a “Imagen” a todo archivo con datos de una imagen sea bitmap o vectorial, sus efectos y datos en distintas capas generadas o modificadas mediante el uso de un software. A continuación se pueden ver las herramientas utilizadas y sus alternativas contempladas, si las hubiese:

- Paint[21]



Imagen 44 Herramienta imagen: Paint

Este software gratuito desarrollado por Microsoft es una herramienta básica de diseño gráfico de bitmap preinstalada en cualquier sistema operativo Windows. La dificultad de uso de este programa es terriblemente sencilla, aunque su potencial también lo es.

Las principales funcionalidades son: abrir cualquier formato básico de imagen bitmap, aplicar efectos básicos, dibujar sobre dicha imagen y exportar a formatos básicos.

Este software se ha utilizado principalmente para recortar imágenes grandes, capturas de pantalla y añadir pequeños retoques simples a algunas imágenes.

○ Gimp[22]

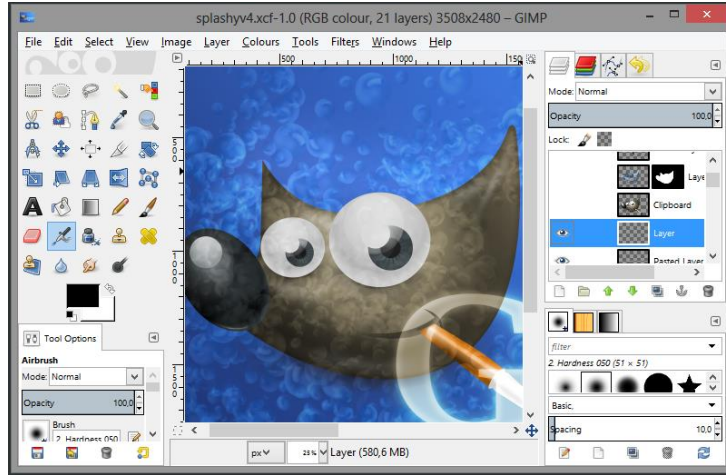


Imagen 45 Herramienta imagen: Gimp

Este software gratuito multiplataforma, parte del proyecto GNU, es una excelente herramienta de diseño gráfico para prácticamente cualquier idea que tengas en mente y quieras pintar o bien quieras añadir efectos o modificar imágenes.

Las principales funcionalidades son: abrir cualquier formato básico de imagen, aplicar una amplia gama de efectos a la imagen, dibujar sobre la imagen y exportar a formatos básicos o formato propio de Gimp.

Este software se ha utilizado para todas las interfaces 2D, botoneras, mensajes, imágenes de fondo, es decir, todo lo que no era tan fácil para hacerlo con Paint. Ya se tenían nociones básicas del uso de este software, por lo que ese factor ha sido decisivo para su uso.

- Adobe Photoshop CC[23]



Imagen 46 Herramienta imagen: Adobe Photoshop CC

Este software de pago (24,19€/mes a partir de 300€ la licencia) desarrollado por Adobe es una herramienta multiplataforma de diseño gráfico extremadamente potente, con posibilidad de aumentar más aún sus posibilidades con programadas de Adobe adjuntos como Adobe Bridge CC.

Las principales funcionalidades son: importar/exportar cualquier formato básico de imagen principalmente bitmap, aplicar una extensa gama de efectos a la imagen, dibujar sobre la imagen sincronización de configuraciones con la nube junto a la de dispositivos móviles.

Este software no se ha utilizado pese a ser una gran alternativa debido a su complejidad inicial y a que ya se conocía una herramienta similar, en concreto, Gimp.

- Motor gráfico

Se hace referencia a Motor gráfico[24] a corazón de una aplicación, la capa software de más bajo nivel que mueve todos los procesos superiores dando la sensación de ser el “motor” de esta máquina, que tiene resultados “gráficos”. Como explicación menos técnica, este motor es el encargado de tener asociados todos los modelos, interfaces, sonidos, clases, objetos, funciones y poder unir los elementos para que trabajen juntos en un terreno común. Cada motor suele tener sus propias reglas básicas de físicas (como la gravedad, colisiones), paralelismo de hilos y gestiones internas que facilitan a los desarrolladores el poder crear juegos de una forma más sencilla sin tener que gestionar la memoria directamente como sería un juego hecho directamente en C. A continuación se pueden ver las herramientas utilizadas y sus alternativas contempladas:

- Unity3D[4]



Imagen 47 Herramienta motor gráfico: Unity3D

Este software con versión gratuita y versión de pago (1500€ la licencia) desarrollado por Unity Technologies es una herramienta multiplataforma considerada un motor gráfico extremadamente potente en auge que tiene una multitud de reglas básicas ya establecidas para los desarrolladores y sus propios lenguajes de diseño C# y Javascript.

Tiene una infinidad de funcionalidades, pero lo más importante a destacar sería que permite importar datos de una amplia gama de formatos de audio, imagen, modelos, animaciones siendo extremadamente versátil con los datos a trabajar. Esto hace que trabajar con la herramienta que trabajes, seguramente Unity3D aceptará el formato que se le intente importar. La segunda más importante, paralela a la anterior, es la cantidad de formatos que permite exportar. Se pueden exportar juegos para Windows, navegador, iOS, MacOS, Linux, Android, BlackberryOS, WindowsPhone, Wii, Playstation y Xbox. Esta característica hace que, si ya era versátil de entrada, tener tantas posibles salidas lo hace un software excelente. No se va a entrar en más características específicas de cómo trata objetos, el paralelismo de hilos, procesos y otras gestiones, pero sí resumirlo en que es una herramienta que cuesta aprender al principio, pero que tiene un gran potencial. El trabajo en Unity3D está enfocado tanto a videojuegos 2D, simuladores 2D, sólo interfaces 2D, videojuegos 3D y simuladores 3D.

Este software ha sido el utilizado durante todo el proyecto, por lo que más detalles se mostrarán en el apartado de implementación. No se tenía ninguna noción de este software, pero se decidió utilizar debido a su versatilidad y que grandes titanes de la industria del videojuego habían decidido usarlo para algunas de sus franquicias.

- Adobe Flash Professional[25]



Imagen 48 Herramienta motor gráfico: Adobe Flash Professional CC

Este software de pago (24,19€/mes a partir de 300€ la licencia) desarrollado por Adobe es una aplicación de creación y modificación de gráficos vectoriales que pueden interactuar con su lenguaje propio ActionScript. Esta aplicación, no es considerada un motor gráfico ya que no reúne los principales requisitos como sean las reglas sobre física, pero es una base sobre la que se pueden montar videojuegos simples para navegador (aunque también tiene otras finalidades esta aplicación). Todos los datos producidos por Flash son tratados como un flujo de video tratado frame a frame que interactúa con código para poder sacar toda la funcionalidad. El uso más común de este software es para ser reproducido como si fuese vídeo en navegador bajo el Adobe Flash Player, herramienta extremadamente utilizada en todo tipo de navegador web. Flash enfocado únicamente a 2D.

Este software no ha sido utilizado durante el proyecto. Se planteó su uso debido a su fácil aprendizaje y su gran extensión por los navegadores, pero con la llegada de HTML5 y su extensión por la web, la tecnología Flash poco a poco irá dejando de usarse. Realizar un videojuego sobre HTML5 sería una mejor alternativa.

- Wimi5[26]



Imagen 49 Herramienta motor gráfico: Wimi5

Este software gratuito desarrollado por el propio equipo de Wimi5 es una plataforma cloud que permite crear, publicar y monetizar juegos sencillos apodados “casuales” en HTML5 (de ahí el 5 de wimi5). Esta plataforma tampoco se considera un motor gráfico, ya que consiste principalmente en el uso de una interfaz sobre la cual se van colocando elementos, con mecánicas ya predefinidas que se pueden conjugar entre sí y así poco a poco elaborar un juego simple/juego casual. *Se considera un juego casual, al videojuego que únicamente se va a utilizar durante una duración de 5 minutos 1 hora, sin intención de volver a rejugarse en otra ocasión, sólo para jugar a algo aleatorio en un pequeño tiempo libre.

Wimi5 es una herramienta muy útil ya que los conocimientos de programación que requiere son muy bajos, con mecánicas muy sencillas para que prácticamente cualquiera pueda utilizarlo. Se enfoca exclusivamente en entorno web, lo cual es un buen punto al ser especializado pero cierra demasiado el marco de opciones salida. Wimi5 enfocado únicamente a 2D.

Este software no ha sido utilizado durante el proyecto. Se planteó su uso debido a la sencillez de su interfaz, su gran facilidad de combinar opciones simples y crear un videojuego en un pequeño periodo de tiempo. Comparado con flash, esta tecnología es mejor ya que viene integrada en el propio navegador y no necesita de un plug-in externo para su funcionamiento. Sin embargo, esto comparado con un gigante como Unity3D, si se tienen expectativas de futuro como programador en un gran proyecto, esta plataforma se queda corta si el desarrollador quiere dedicar su carrera por este rumbo. Si es crear videojuegos como pasatiempo, es una opción genial, pero no es el caso.

○ Unreal Engine 4[27]

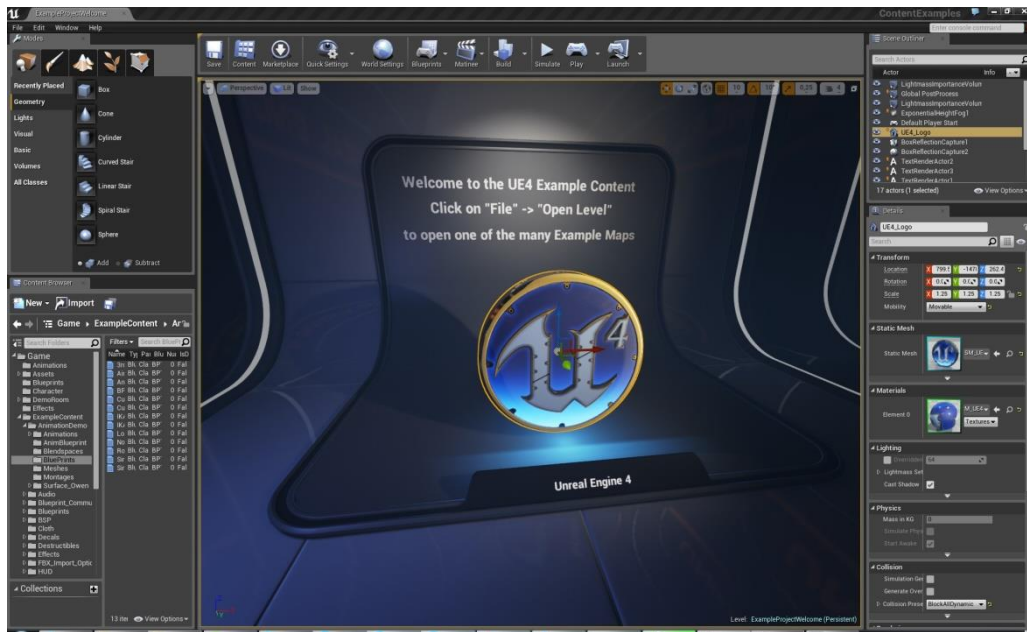


Imagen 50 Herramienta motor gráfico: Unreal Engine4

Este software de pago (19\$/mes) desarrollado por Epic Games es un motor gráfico con una larga, inicialmente dedicado a juego del género *Shooter* en PC aunque actualmente es usado para todo tipo de género en múltiples plataformas.

Este motor tiene aproximadamente las mismas funcionalidades que Unity3D, salvo que tiene diferencias características. Hablando más técnicamente, también es muy versátil con datos de entrada y formatos de salida, pero la interfaz, el trato de objetos, gestión de memoria, hilos, reglas, clases, código son diferentes. Unreal Engine 4 es igual o más potente que Unity3D a la hora de renderizar más polígonos en la misma escena respecto a lo que sería tratar escenas pesadas. Unreal Engine puede enfocar sus herramientas frente a todo tipo de aplicaciones/juegos en 2D y 3D.

Esta herramienta no ha sido utilizada durante el proyecto. Era la mejor alternativa, pues tiene grandes salidas laborales, permite una amplia gama de formatos de entrada, usable en todo tipo de plataformas de salida, aunque con una interfaz algo áspera al principio, como todo motor gráfico. Debido a que no existía una versión gratuita al inicio del proyecto, se eliminó esa alternativa dejando paso a Unity3D como vencedor.

3.3.2 Diseño de la propuesta

Una vez analizados los requisitos recogidos y analizados enmarcados en el contexto con el que se van a conseguir, es preciso profundizar más aún en la parte de análisis del sistema tocando plenamente la ingeniería del software. De los múltiples ámbitos de la ingeniería del software que aún no se han tocado, a continuación se va a hacer énfasis en los casos de uso, los escenarios, los diagramas de navegación como en la matriz de trazabilidad.

Este apartado no debe pasar desapercibido en ningún proyecto, ya que consiste en la base más sólida sobre la que se empieza a construir todo el código que soportará la aplicación de este o cualquier otro proyecto. Sin una concienzuda revisión de estos puntos, el sistema final puede tener inconsistencias, carencias, evitar repetición de errores, hasta incluso haber perdido la esencia que trata de resolver. Por esto y una lista casi innumerable de razones, es vital la existencia de la ingeniería del software en todo aquel software que no quiera ser un fracaso.

Es de importancia citar el trabajo de Mario Velázquez de un juego para PC mediante Ogre 3D[28] debido a la ayuda de sus diagramas de secuencia y diagrama de actividad que me ayudaron a plantear correctamente los míos.

3.3.2.1 Casos de uso

Los casos de uso son la descripción de todos los pasos necesarios que deben llevarse a cabo para poder realizar un proceso. Los personajes que participan en los casos de uso son denominados actores. Los casos de uso sirven para unir mediante una interacción activa al usuario con el sistema.

Este es el diagrama que une todos los casos de uso del proyecto:

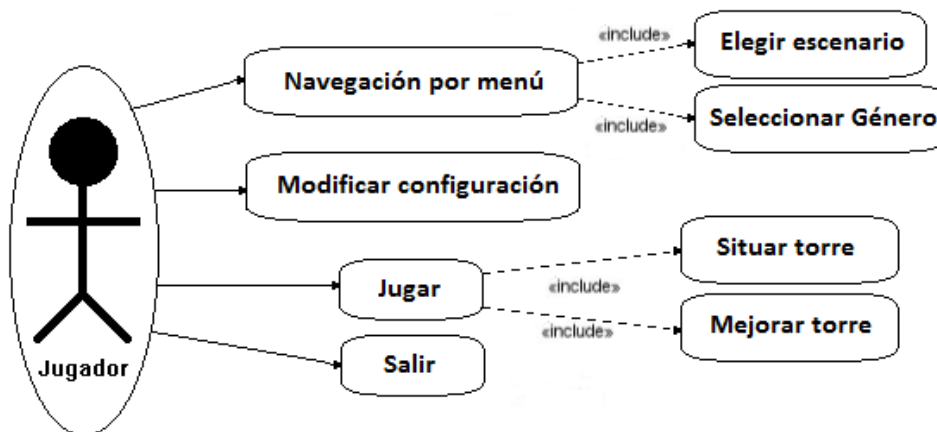


Imagen 51 Diagrama de casos de uso

Ahora que se pueden tener como referencia en el diagrama, se van a explicar con un mínimo de detalle en qué consiste cada uno de ellos, teniendo en cuenta que el primer caso de uso es el primer de la lista, el segundo es el caso de uso nº2, etc. He aquí los casos de uso explicados:

- Navegación por menú. Permitir al usuario navegar por las distintas interfaces del menú principal mediante el uso de botones.
- Elegir escenario. Permite al jugador uno entre varios escenarios posibles.
- Seleccionar género. Permite al jugador elegir su género dentro del juego.
- Modificar configuración. Se permite puede modificar el audio a gusto del jugador.
- Jugar. Parte fundamental donde el jugador estratégicamente decide el cuándo comenzar las oleadas permitiendo así que se ejecuten las mecánicas del juego.
- Situar torre. Dentro de un escenario, el jugador puede seleccionar una entre varias torres posibles y situarlas en distintas ubicaciones prefijadas.
- Mejorar torre. Una vez existentes distintas torres en el escenario, seleccionando una torre específica, ésta se puede mejorar si se reúnen los debidos requisitos.
- Salir. Salir de la aplicación.

3.3.2.2 Matriz de trazabilidad

A medida que se desarrolla el ciclo de vida del desarrollo del software de un proyecto, se llevan a cabo dos fases muy importantes ya explicadas en este documento: Los requisitos y los casos de uso. Cuando el proyecto tiene un mínimo de envergadura con su dificultad y complejidad adjunta, es común llegar al punto de no saber qué test se han ejecutado para comprobar los requisitos y qué casos de uso se relacionan con qué. Para cumplir esta necesidad, ya que este proyecto, aunque no es muy grande, sí lo es como para necesitarlo, se utiliza una matriz de trazabilidad. Esta matriz sirve para corroborar la unión de las pruebas o casos de uso existentes y qué relación tienen con los requisitos para poder saber que todo lo exigido se ha cumplido. La nomenclatura para la matriz es, de cara a los casos de uso en el eje X, CU-N donde CU significa 'caso de uso' y N es el número de del caso de uso. Para los requisitos en el eje Y, se usa la misma nomenclatura que en la sección 3.2, 'Análisis'. Un aspa 'X' marca que ese caso de uso corrobora ese requisito mientras que un espacio en blanco ' ' indica que no tiene referencia. A continuación se puede ver la matriz respectiva al proyecto:

	CU-1	CU-2	CU-3	CU-4	CU-5	CU-6	CU-7	CU-8
REQ-HG-U-SF 01	X							
REQ-HG-U-SF 02		X						
REQ-HG-U-SF 03				X				
REQ-HG-U-SF 04								X
REQ-HG-U-SF 05			X					
REQ-HG-U-SF 06						X		
REQ-HG-U-SF 07					X			
REQ-HG-U-SF 08					X			
REQ-HG-U-SF 09					X			
REQ-HG-U-NF 01					X			
REQ-HG-U-NF 02	X							
REQ-HG-U-NF 03					X		X	
REQ-HG-S-SF 01	X				X			
REQ-HG-S-SF 02	X				X			
REQ-HG-S-SF 03					X			
REQ-HG-S-SF 04					X			
REQ-HG-S-SF 05	X				X			
REQ-HG-S-NF 01	X				X			
REQ-HG-S-NF 02	X				X			
REQ-HG-S-NF 03	X							

Tabla 25 Matriz de trazabilidad

3.3.2.3 Diagrama de navegación del videojuego

Partiendo como base la matriz que se acaba de mostrar para corroborar que todo está comprobado, se va a exponer en esta sección un diagrama de navegación del propio juego, dentro de la parte teórica. De esta forma se puede visualizar de una forma más clara el ordenada la actividad del sistema desde el inicio hasta su cierre.

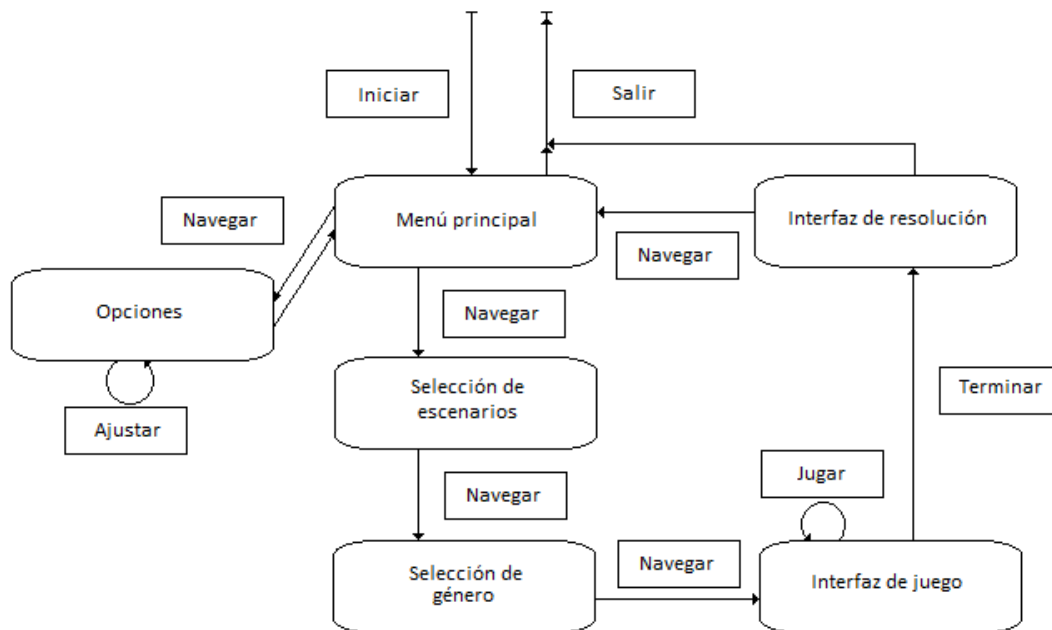


Imagen 52 Diagrama de navegación del videojuego

3.3.2.4 Diagramas de secuencia

Los diagramas de secuencia se utilizan para poder entender de una forma cómoda y clara como distintos objetos o partes del sistema interaccionan entre sí a lo largo del tiempo. Junto al diagrama de la sección anterior, con este se puede acabar de entender perfectamente el flujo de información en el sistema de una forma completamente teórica.

El primer diagrama de secuencia hace alusión al cambio de la configuración en su menú asociado.

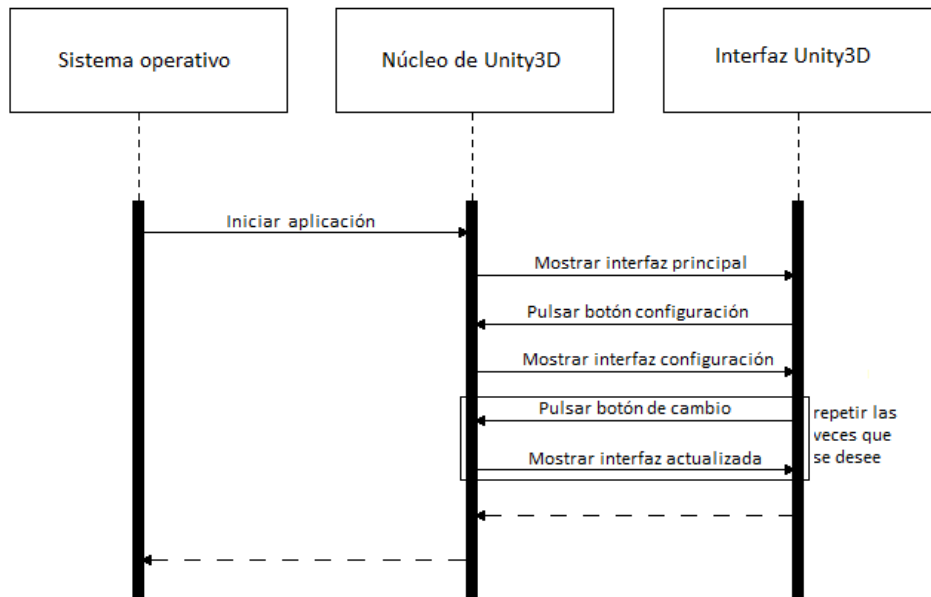


Imagen 53 Diagrama de secuencia 1, configuración

El segundo diagrama de secuencia transmite los pasos de lo que es salir de la aplicación desde el menú principal como pudiese haber sucedido tras la secuencia 1 o la resolución de la secuencia 3.

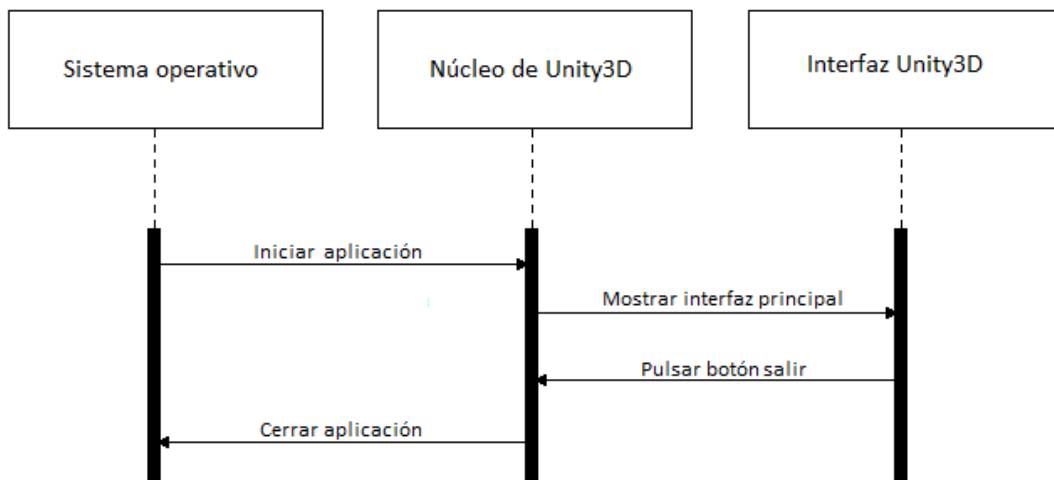


Imagen 54 Diagrama de secuencia 2, salir de la aplicación

El tercer diagrama de secuencia muestra el paso intermedio de las secuencias 1 y 2, es decir, tras configurar, si se configuran las opciones y si se quiere o no salir de la aplicación. Trata la mecánica principal por parte de la interacción del usuario.

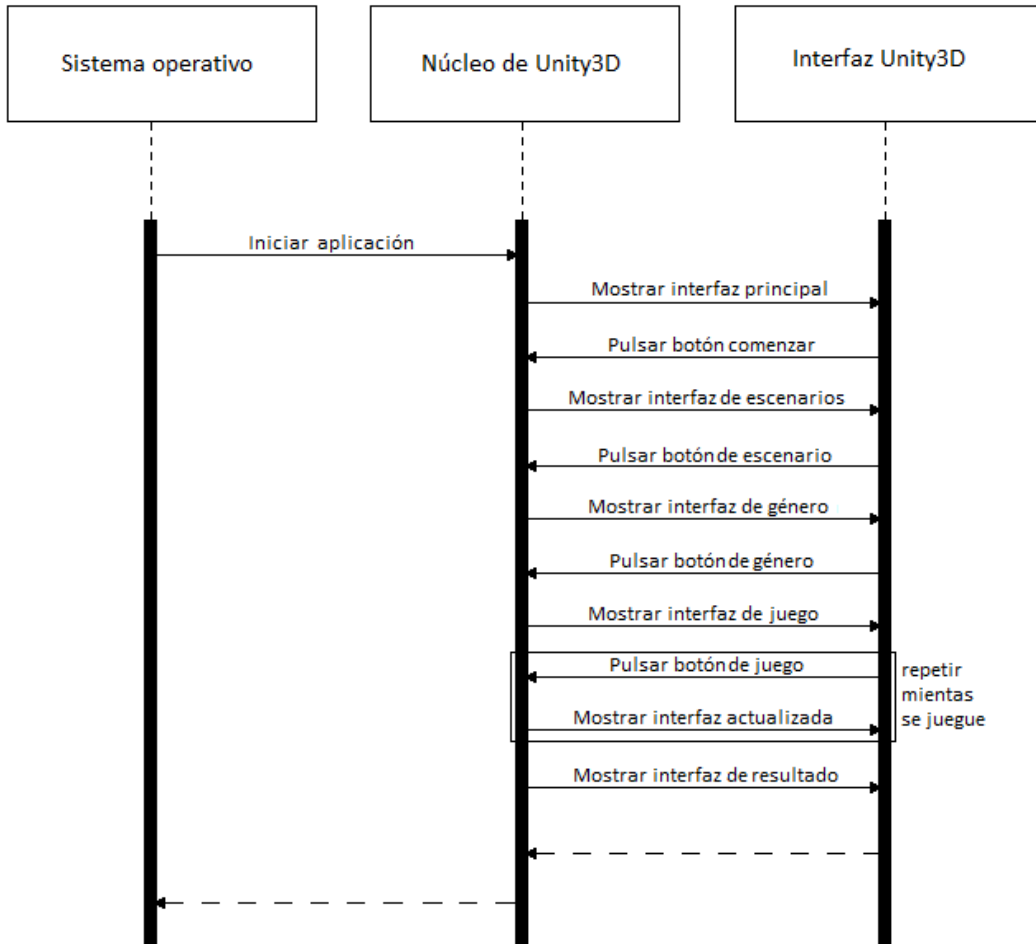


Imagen 55 Diagrama de secuencia 3, mecánica de juego

3.3.2.5 Diagrama entidad-relación

Este diagrama sirve de herramienta para controlar fácilmente los datos de una base de datos mediante la representación de las entidades que lo componen y la relación existente entre entidades.

Se debe decir que este juego no tiene una base de datos permanente en la que se almacenen puntuaciones o el jugador tenga un progreso con datos que se deban ir guardando permanentemente. Sin embargo, a lo largo del procedimiento de la mecánica principal del TD, se crean una alta cantidad de entidades temporales que son las que interaccionan y dan sentido a todo el desarrollo.

El diagrama de la parte inicial, menú principal del juego, no merece la pena ser descrito ya que no son más que menús cargados dinámicamente en función de lo que

se pulse. Los escenarios de juego contienen toda la información que sí merece la pena describir a conciencia. Como explicación general, la entidad LevelMaster es aquella parte que coordina absolutamente todo en el juego almacenando tanto la información del jugador, de los enemigos y todos los elementos que interaccionan entre sí excepto la interfaz que es la que el usuario usa para controlar los datos de LevelMaster y poder jugar como tal.

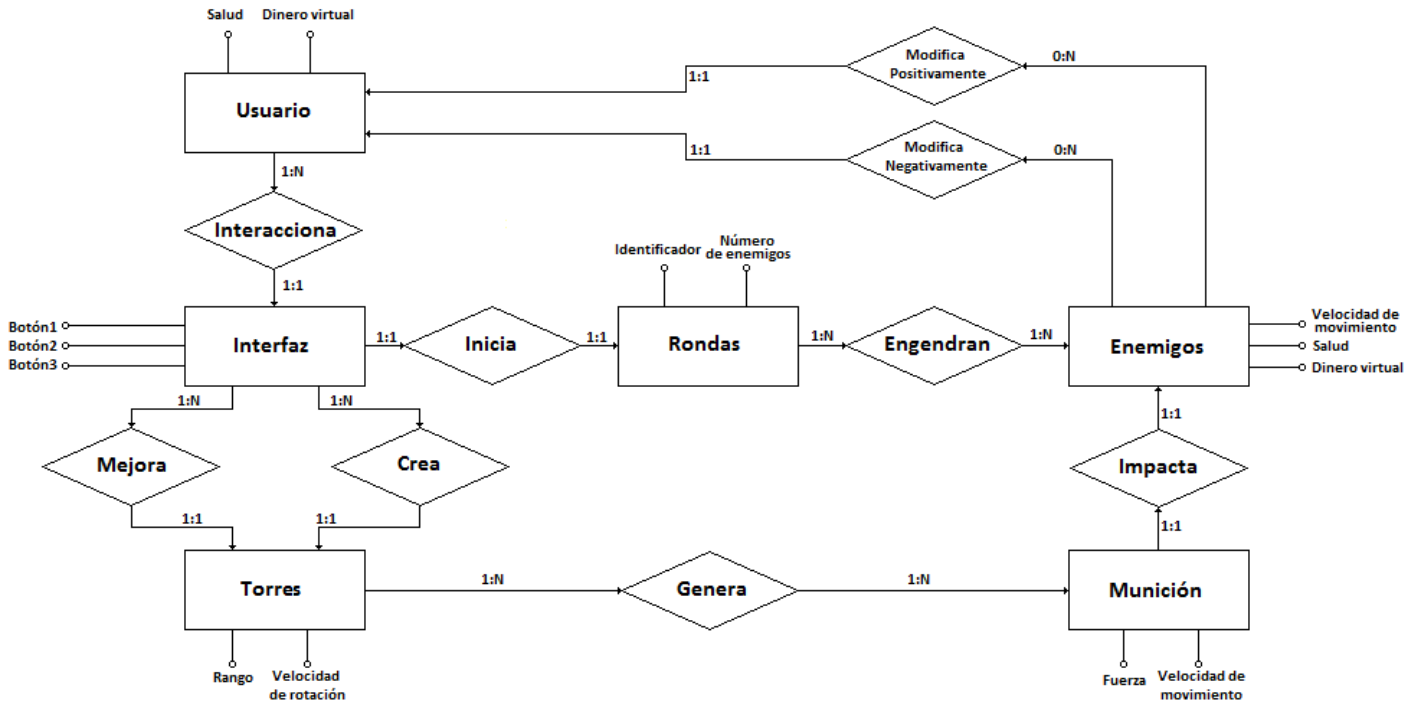


Imagen 56 Diagrama entidad-relación

3.3.2.6 Componentes del motor gráfico

Como se ha ido comentando a lo largo del documento, la herramienta principal para la perfecta elaboración y nexos de unión de todos los elementos que componen el videojuego ha sido Unity. Unity es un editor de espacios virtuales que permite importar una gran cantidad de formatos de archivos de imagen, audio, modelos 3D con o sin animaciones y maniobrar con ellos al antojo del desarrollador.

A la hora de operar de correctamente todos los elementos que se importen a Unity3D se ha de hacer mediante líneas de código en alguno de los dos lenguajes de programación propios de la herramienta. Para poder realizar esta tarea, *Unity3D* posee otra herramienta paralela, un entorno de programación para los lenguajes del software principal, llamada MonoDevelop-Unity.

Para poder desarrollar todo un videojuego, la programación y agrupación de elementos se divide en “escenas”, siendo cada una de ellas un “mundo” diferente al anterior con distintos elementos situados en él. Esto se puede usar para no saturar al

procesador y hacerle cargar todos los elementos a la vez sino dividir y colocar cada elemento en su lugar o bien, como el caso de este juego, para separar ámbitos como los menús de navegación principal, victoria y derrota de los escenarios de juego. Mediante código es como al usuario se le redirige de una escena a otra según active un evento o desee el programador, en este juego es pulsando botones del menú o finalizando un escenario.

Cada elemento de las escenas está dividido en módulos para poder definir con precisión qué debe y qué no debe hacer. El primer módulo básicos es la definición de parámetros base como nombre, cada y etiqueta. El segundo es el Transform que hace referencia a cómo y dónde está un elemento. El tercer y último elemento básico es el de tener un script a ejecutar. Nota, un elemento sin script, puede interactuar con otro elemento con script que haga referencia a él.

Para abordar mejor esta sección, tras la introducción básica al funcionamiento de la herramienta, primero se va a explicar con mayor profundidad las distintas escenas, luego con más detenimiento los principales módulos utilizados de los elementos básicos y no básicos y finalmente la estructura de almacenamiento de ficheros del proyecto. Esta organización ha sido basada en la utilizada en el TFE de David Gómez, Java'S Purgatory[29].

- Definición y movimiento en escenas

De forma muy similar al diagrama de navegación de la sección 3.4.3, existen un total de 6 escenas claramente diferenciables y divididas adrede.

La primera que resulta evidente, es la del menú principal. Esta escena consta de un total de 4menús. Menú principal, menú de opciones, menú de selección de escenario y menú de selección de género. Los botones mantienen una estructura idéntica entre sí, aunque se explicarán más adelante. Esta es una escena ligera con muy pocos elementos para que el juego cargue rápidamente tras arrancarlo desde el sistema operativo. Una vez seleccionado el género, según qué escenario se haya escogido, al usuario se le redirigirá a la escena del escenario en cuestión.

La segunda, tercera y cuarta escenas son idénticas entre sí, salvo por la ambientación con todos los elementos activos y pasivos que ello necesita. Dado que aquí es donde más elementos activos existen, se decidió separarlos del menú principal y de la victoria y derrota. Al cargar estas escenas se puede ver como el tiempo de carga no es tan reducido como para el menú principal y el usuario experimentará un pequeño retardo desde la selección de género en la primera escena a alguna de estas. Si no se hubieran dividido estas escenas y se hubiera creado un único mundo grande pero con la condición de que según el escenario la cámara enfoque a un lugar o a otro, los tiempos de carga serían considerablemente más altos aumentando el coste computacional para el ordenador. Asimismo, esto

habría disminuido la cantidad de usuarios finales que puedan utilizar este producto debido al material necesitado por lo que la utilidad del juego sería menor. Si el usuario completa adecuadamente la mecánica de la escena, se le redirigirá inmediatamente a la quinta escena. En caso contrario, será redirigido a la sexta.

La quinta y sexta escenas actualmente son extremadamente escuetas y no merece la pena que sean dos distintas. Se ha decidido dividir las, porque como ya se visualizará en la sección 6.2.4 Trabajos futuros, en orden de ser capaz de realizar largas animaciones en lugar de imágenes, es importante que estas escenas estén divididas para agilizar al máximo el periodo de carga de estas. Desde aquí también se podrá volver a la primera escena o bien salir de la aplicación.

En la siguiente imagen se puede ver el menú de compilar y montar el juego seleccionando las distintas escenas que se quieran usar. Para hacer pruebas se montan y prueban individualmente. En este caso es del juego completo. También se puede ver la zona de selección de la plataforma destino.

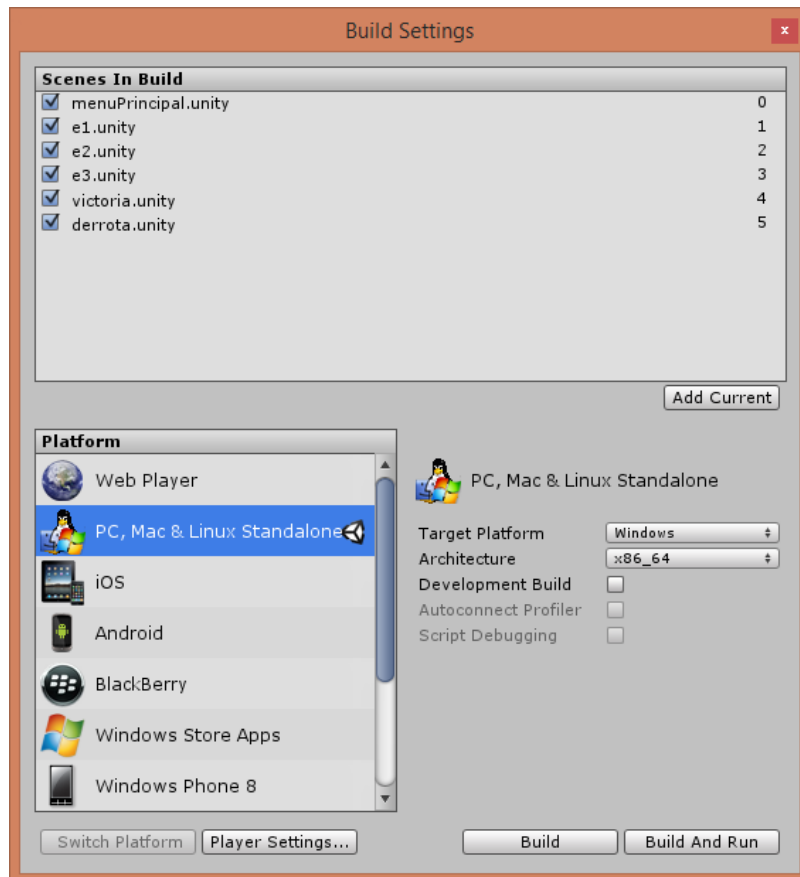


Imagen 57 Selección de escenas a compilar y montar

- Principales módulos utilizados en elementos

Se considera un elemento de una escena como un objeto puesto en la jerarquía de la escena y mediante el uso del inspector se pueden ver detallados los módulos que tienen asociados.

El inspector es la sección de la interfaz de Unity3D para mostrar información detallada de un elemento seleccionado. A partir de ahora, cuando se expliquen estos módulos, se hará referencia a una imagen de un ejemplo de uso de una de ellos.

- Transform

Este módulo inicial pseudo-obligatorio especifica la situación espacial de los elementos. Comienza con las coordenadas XYZ, luego define la rotación en ejes XYZ de la posición original del elemento (si tiene un componente de modelo 2 o 3D esto es más llamativo) y finalmente la escala del elemento (si no es un modelo 2 o 3D carecen de sentir el valor de estos campos).

*Nota, Lo primero que se ve debajo de la palabra Inspector es la parte sí obligatoria de cada elemento con su nombre, etiqueta, capa y forma de representación.

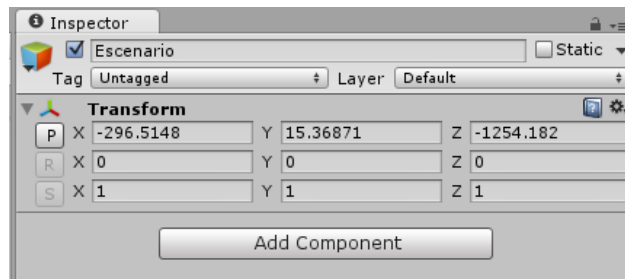


Imagen 58 Módulo Transform en un elemento

- GUITexture/GUIText

Estos dos módulos hermanos son los que se han usado para las interfaces del menú principal, menú victoria y menú derrota.

GUIText es el encargado de escribir texto en la pantalla. Los propios nombres de los parámetros son auto-explicativos. Anchor se refiere a qué esquina de margen de la pantalla está anclado el texto. El material es un parámetro que no tiene por qué usarse, sería para utilizar una foto o una textura/material como color. Pixel Correct y Rich Text son opciones para dar más calidad al texto. Texto, alineación, márgenes, espaciado, tabulado, fuente, tamaño de fuente, estilo de fuente, color no merecen ser explicados.

GuiTexture es más simple aún que GUIText. Texture es la imagen que se quiere mostrar. El color por defecto es un gris neutro, ya que el brillo se le daría con color blanco y oscuridad con color negro. Dentro de color, la línea negra que tiene en su inferior hace referencia a la capa alpha de la imagen, la que se usa para la transparencia de la imagen. Pixel Inset se usa para mover la imagen en píxeles sobre ejes XY y para definir la escala XY. Los cuatro Left/Right/Top/Bottom Border sirven para indicar los márgenes en píxeles a los que no les afecta el escalado de la imagen en pantalla.

El pequeño problema que tiene este componente es que utiliza pixeles por lo que con distintas resoluciones de pantalla los resultados finales pueden ser muy distintos según el usuario. Para evitar este problema, en los escenarios se ha utilizado una librería externa para su elaboración.

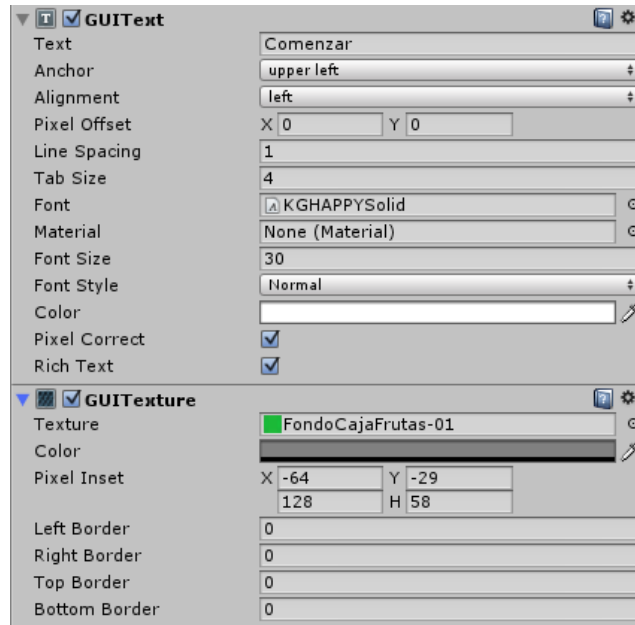


Imagen 59 Módulos GUIText y GUITexture en un elemento

- Script

En este módulo todos los parámetros dependen de las variables globales definidas en el propio código, por lo que depende principalmente del lenguaje de programación, algo que no se va a tratar aquí. En el siguiente ejemplo, el del script de cualquier torre rellena con los parámetros del duende tomate. Los parámetros son de todo tipo, desde el propio script en Script, un GameObject, que es otro elemento, en My Upgrade, integer en My Upgrade Cost, String en My Upgrade Name, o arrays de módulos Transform como los Muzzle Position.

Todo depende del propio código como tal.

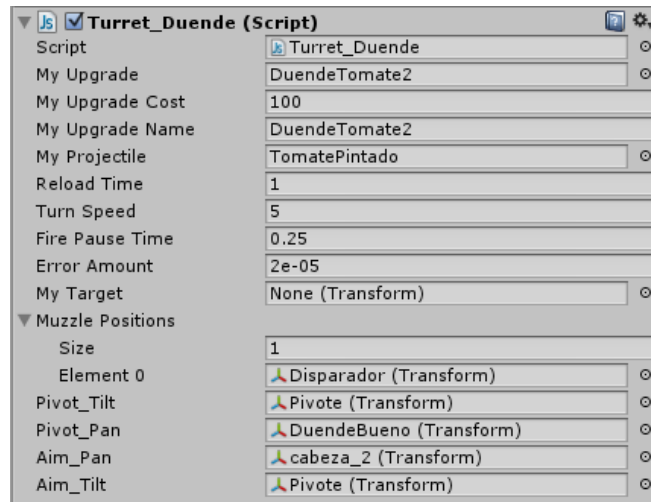


Imagen 60 Módulo Script en un elemento

○ Animator

Este módulo únicamente lo tienen los modelos 3D no planos que tienen el modelo y texturas resumido en un único archivo reconocido por Unity. El parámetro Controller tiene valor según el movimiento que vaya a tener el elemento, o si el propio jugador mediante un first person controller lo va a utilizar. El avatar es el propio modelo con o sin animaciones (esto depende del programa que lo creó). Los otros parámetros son para las animaciones mediante un checkbox y el Culling Mode también depende del programa de exportación.

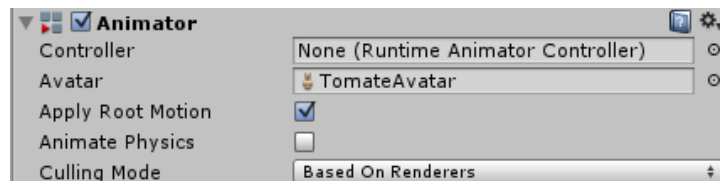


Imagen 61 Módulo Animator en un elemento

○ Mesh Renderer

Como alternativa al anterior módulo anterior, éste se usa al importar modelos cuyo formato Unity no puede interpretarlo del mismo modo. En esta ocasión las texturas se deben introducir manualmente habiéndose exportado también desde el otro software y situado en el array Materials. Los checkboxes de Cast Shadows son para reflejar las sombras en las superficies cercanas si existe un foco de luz mientras que Receive Shadows es para reflejar las sombras de la luz sobre sí mismo. Para cada material del array de materiales se pueden aplicar las texturas correspondientes. En el caso de ejemplo mostrado de las banderas del escenario 1 sólo son dos colores ya que es una bandera de dos colores.

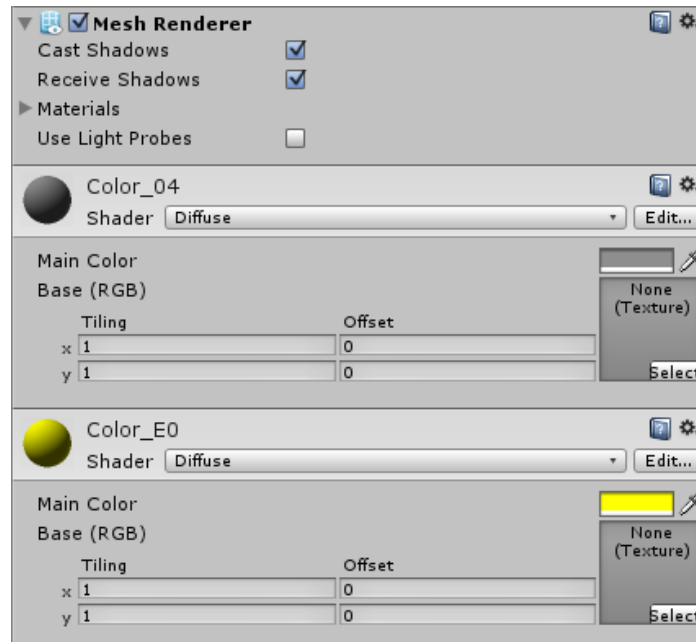


Imagen 62 Módulo Mesh Render de un elemento

o X Collider

Este módulo, a título personal, lo considero el más importante junto al de script para poder desarrollar juegos. Consiste en una forma geométrica que rodea al elemento tenga un modelo asociado o no. El módulo no se llama X Collider, aquí se usa la X debido a que existen diversos tipos según la forma geométrica. Los más comunes son: Sphere Collider, Box Collider y Capsule Collider.

El primer checkbox significa si el Collider está activo o no. El material sería si no se quisiese que fuese invisible, aunque lo normal es que sea invisible. El centro suele ser el centro del elemento y, según la forma geométrica, tendrá un radio, tamaño de lado o diámetro. Este módulo, de cara al código javascript utilizado es el que se ha utilizado para que los enemigos tengan su esfera de presencia, las torres tengan un radio de visión y detecten la presencia del enemigo y disparen proyectiles que al cruzar su Collider con el del enemigo se produzca el impacto.

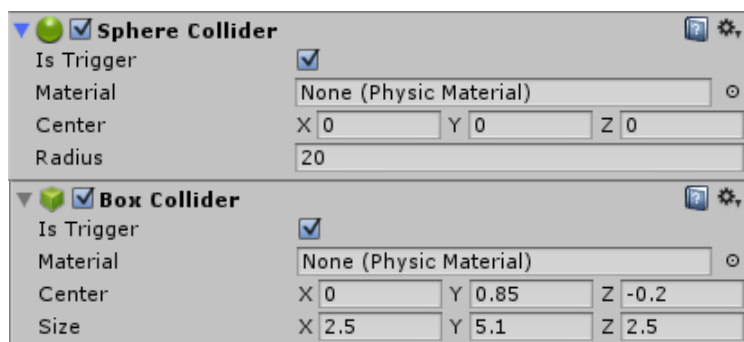


Imagen 63 Módulo X Collider de un elemento

○ Character Controller

Este módulo únicamente lo poseen aquellos elementos que se vayan a desplazar por la escena y representa el centro desde el cual se va a mover. También existen otras formas de mover o rotar elementos en una escena, pero esta es la más sencilla y uniforme ya que internamente está programada para interactuar con las físicas si también tiene un Rigidbody y ser más real. Los parámetros son evidentes como el ángulo límite de una pendiente del cual no puedes trepar más, el tamaño de los pasos, grosor de la peana del elemento, distancia mínima de avance independiente a los pasos, centro de la peana, el radio y el tamaño.

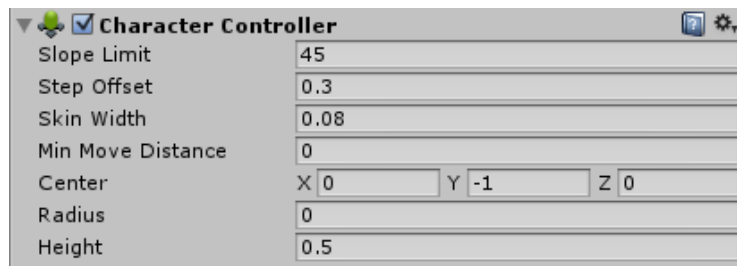


Imagen 64 Módulo Character Controller de un elemento

○ Rigidbody

Este módulo sirve para activar la simulación de físicas establecidas por Unity3D. Físicas tales como la gravedad, colisión entre elementos aplicada al elemento entero, o si el elemento se ha dividido en partes, sólo aplicada a los subelementos que tengan interacción con el terreno. Los parámetros básicos son la masa, el arrastre, el ángulo de arrastre, si afecta o no la gravedad, el control de físicas que no sean gravedad, interpolación, colisiones o fijar el posible ángulo de position y rotación del elemento según ejes XYZ.

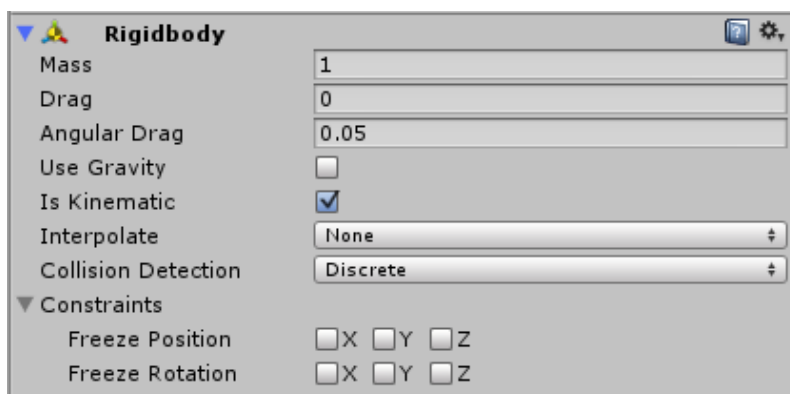


Imagen 65 Módulo Rigidbody de un elemento

○ Audio Listener/Audio Source

Estos dos módulos se necesitan mutuamente en relación 1:N por escena. Para que una fuente de audio pueda ser percibida, debe haber un oyente activo, pero no más de uno activo a la vez. Normalmente el oyente está

situado en la cámara, para que los sonidos que lleguen a lo que ve el usuario, también se oiga. Si por alguna razón se cambia a otra cámara o la cámara se mueve, habría que activar otro Audio Listener o bien hacer que el elemento cámara también tenga el listener y se mueva a la par.

Dado que el oyente sólo es un oyente sin parámetros, simplemente escucha todo mientras esté activo ahora se explica el emisor de audio.

Audio Clip hace referencia al archivo de audio a reproducir. Los siguientes checkbox tienen la finalidad de ser activados o desactivados durante la ejecución del audio. Mute por si se quiere silenciar (no pausar) el clip, Bypass para activar efectos predefinidos, Play On Awake para que este clip se escuche desde el inicio de la escena y Loop por si va a ser la música de fondo y se desea que se repita indefinidamente o hasta ser silenciado. Prioridad, volumen y tono se usa para destacar un sonido dentro de otros. La música de fondo suele tener menos volumen y prioridad mientras que los efectos de sonido, de disparos en este caso o los consejos, tienen un volumen y prioridad mayor para poder escucharse claramente.

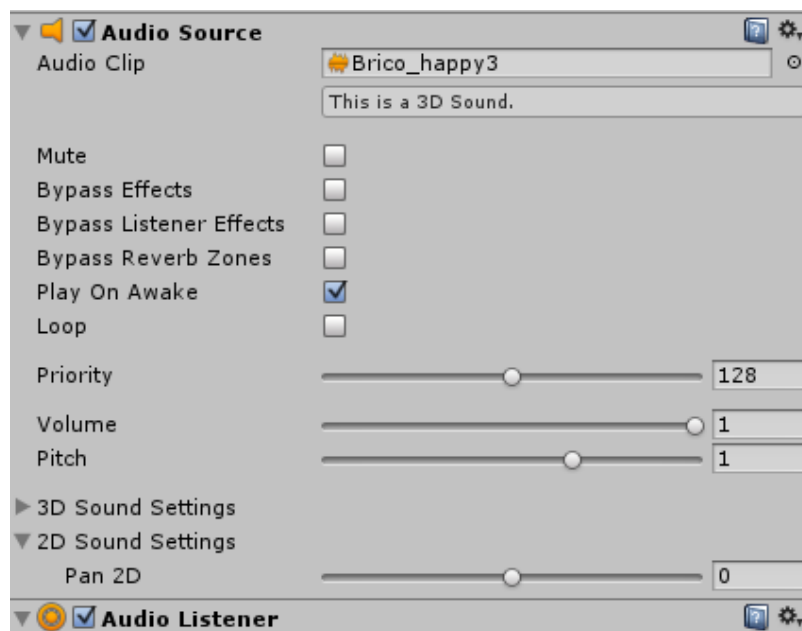


Imagen 66 Módulos AudioListener y AudioSource de un elemento

- Otros

Únicamente por el hecho de nombrarlos, existen otros módulos utilizados para la IA o para la interfaz de los escenarios sacados de librerías externas programados en C#. UIRoot, UICamera, UIPanel, UIAnchor, Tween Position, UIButton, UISound, UIMessage, UILabel y UISlice Sprite para la interfaz. Seeker, Asthar Path y Compass para la IA que hace que los enemigos recorran el laberinto.

- Estructura de almacenamiento de ficheros

Para concluir con la información básica sobre Unity3D y su uso para realizar este proyecto respecto al diseño, se va a concluir con la estructura de ficheros y su organización.

Ya se ha comentado que no existe una base de datos como tal y que tampoco se registran máximas puntuaciones, por lo que cada partida es independiente de la anterior y el único almacenamiento existente es el de los propios datos.

Se va a explicar la estructura empezando por el exterior hacia el interior con la siguiente imagen de referencia global:

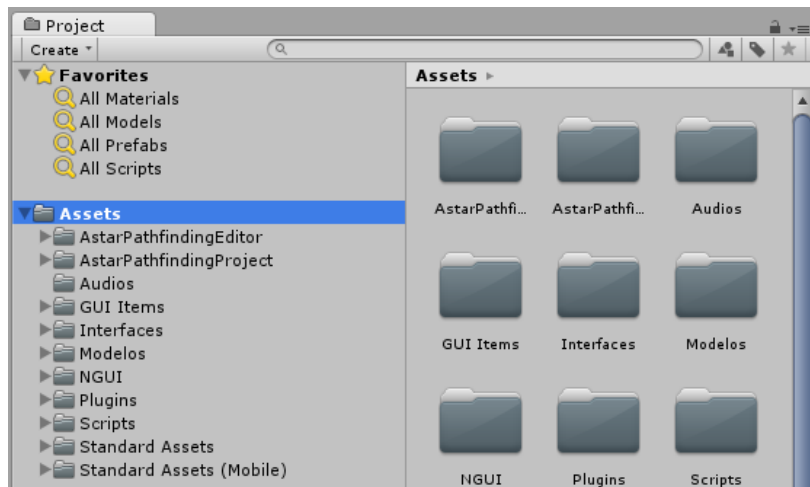


Imagen 67 Jerarquía principal de almacenamiento de datos en Unity3D

Por encima de Assets, están los directorios de librerías, metadatos del proyecto y ficheros para definir el proyecto como proyecto de Unity3D. Se puede considerar que el directorio Assets es todo aquello con lo que se trabaja desde el editor. AstarPathfindingEditor/Project son la librería externa de la IA.

GUI Items, NGUI y Plugins son la librería y el repositorio del atlas externo para la interfaz de los escenarios.

En Audios se han almacenado todas las músicas y efectos de sonido del juego.

En Interfaces se han almacenado todos los modelos 2D como fondos de menú, vidas, rondas, botones y texturas para interfaz. El subdirectorío que tiene simplemente es para almacenar las fuentes usadas en las interfaces.

En Modelos se abre bastante el abanico pues aquí residen todas las torres de duendes con su modelo inicial, modelo con texturas, su munición, munición con texturas, su prefab (elemento para hacer copias idénticas) con el código y sus parámetros ya rellenos y lo mismo para todos los enemigos. Al ser el directorio del proyecto, este apartado no distingue de escenas y almacena todos los datos de todo. El desarrollador es el que decide cómo gestionar todos los directorios, por lo que la estructura de este directorio y los anteriores no externos explicados también han sido decididos así adrede.

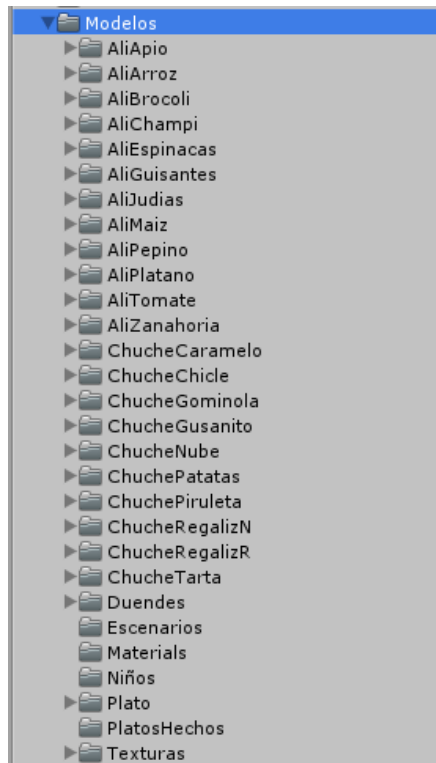


Imagen 68 Estructura del directorio Modelos

Para almacenar los scripts se ha creado su propio directorio para tenerlos organizados, aunque como se ha citado recientemente, los prefab también tienen el código introducido en como módulo por lo que no importa la ubicación. Para organizar mejor los scripts, se han copiado aquí también los de la librería externa NGUI para tenerlos mejor controlados, ya que MonoDevelop-Unity usa los directorios del proyecto que hayas escogido, así que lo óptimo es tener los ficheros relativamente cercanos. A continuación se pueden ver todos los ficheros de código creados de 0 para el proyecto.

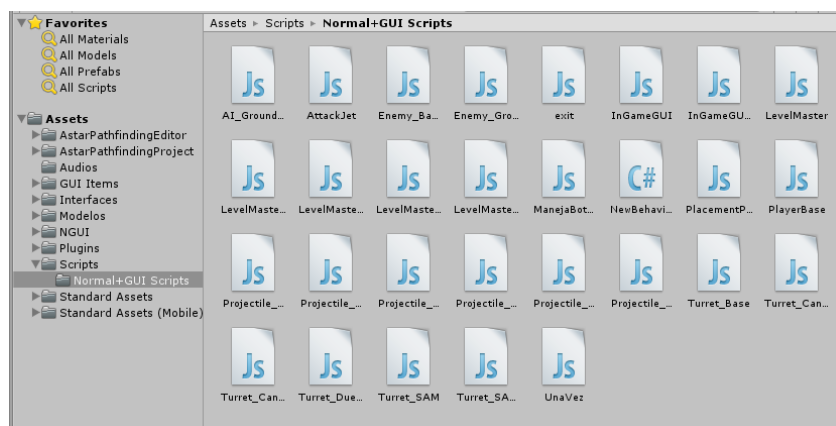


Imagen 69 Estructura del directorio Scripts

Los directorios Stander Assets y Estándar Assets(Mobile) se usan según la plataforma de destino del juego, aunque inicialmente son iguales por lo que explicando

una se explican las dos. Aquí están los datos que el desarrollador deciden importar de los repositorios base de Unity3D, por ejemplo, el generador de terreno, el generador de árboles, generador de agua, efectos especiales, las físicas, los character controller, etc.

3.4 Implementación

Con todo el diseño detallado, la puesta a punto del proyecto supuso un largo periodo de trabajo el cual se va a resumir en estadísticas para poder visualizar de forma ágil y eficiente el trabajo realizado. Se va a expresar toda la información en forma de tablas para representar datos del código, datos de la interfaz, datos de los modelos y datos del audio.

Datos de código	Valor
Cantidad de archivos	29
Número total de líneas de código	72+17+62+90+6+160+161+364+647+438+568+36+54+4+58+25+12+11+55+55+23+8+90+94+132+117+132+11= 3502
Media de líneas por archivo	120

Tabla 26 Estadísticas del código

Datos de la interfaz	Valor
Cantidad de modelos 2D distintos	53
Cantidad de botones existentes	42
Cantidad de tipografías usadas	3

Tabla 27 Estadísticas de interfaces

Datos de los modelos3D	Valor
Cantidad de modelos 3D distintos	62
Cantidad de modelos montados	26
Animaciones usadas	4

Tabla 28 Estadísticas de modelos3D

Datos del audio	Valor
Cantidad de audios distintos	28
Cantidad de melodías usadas	2

Tabla 29 Estadística de audio

3.5 Implantación

Para llevar a la práctica este proyecto, el proceso de instalación y uso es extremadamente sencillo. Una vez el usuario final ha recibido el juego, este consta de dos partes: un directorio de datos y un ejecutable (varía según el SO). Es muy importante dejar ambas partes en el mismo directorio antes de la ejecución.

Una vez estén preparadas las dos partes, únicamente hay que activar el archivo ejecutable (en Windows) y el último para antes de jugar, es elegir la resolución de la pantalla con la que se quiere jugar. Una vez se pulsa en “Play!” comenzará el videojuego a funcionar.

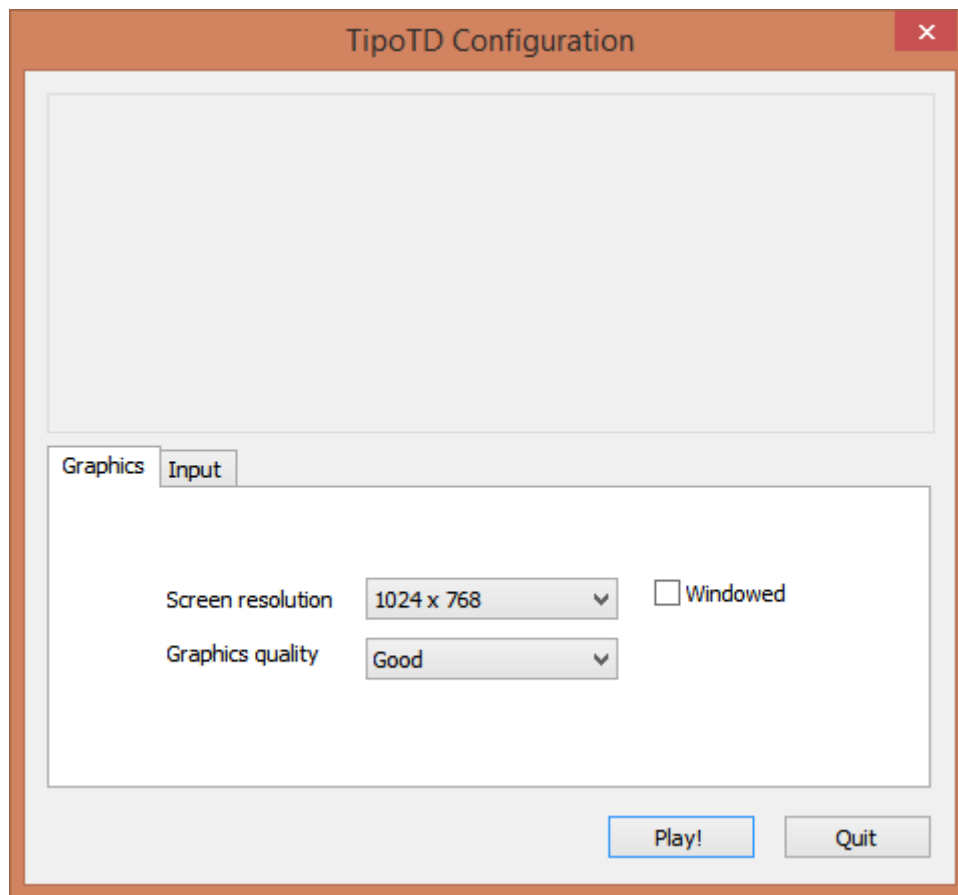


Imagen 70 Menú de apertura del videojuego

Una vez dentro del juego, se empezará con el menú principal, el cual tiene múltiples opciones con opciones para llegar a las pantallas de juego. Esta información está redactada con claridad junto a la del resto de juego en el apéndice de manual de usuario.

4 Planificación y presupuesto

Este capítulo está dividido en dos apartados donde se desarrolla la parte de ocupación temporal y la gestión de recursos, asumiendo los precios que estos llevan consigo. Primero, mediante un diagrama de Gantt se puede ver el diseño y desarrollo temporal de las distintas partes del proyecto. En segundo y último lugar, se pueden visualizar todos los gastos y recursos consumidos de forma directa o indirecta con los costes asociados aproximados que equivalen.

4.1 Planificación

En esta pequeña sección únicamente se muestra una ilustración del diagrama de Gantt, desarrollada con el software Gantt Project^[30] y que muestra las tareas asignadas y los tiempos de realización de cada una de ellas desde el inicio del proyecto, hasta el final. Todos los nombres de las tareas son auto explicativos por sí mismos.

En el siguiente folio con vista apaisada se puede visualizar a la perfección el diagrama.

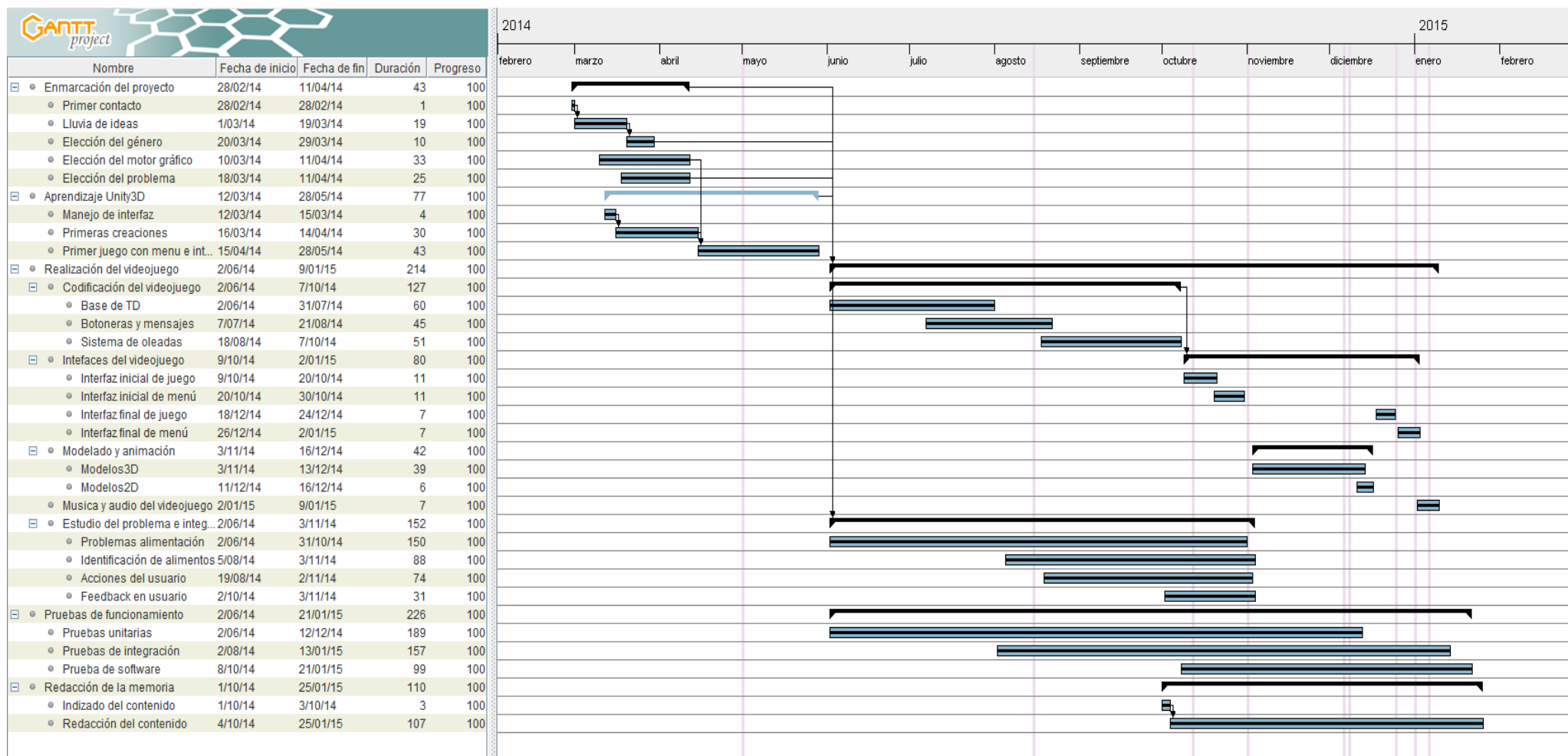


Imagen 71 Diagrama del Gantt del proyecto

4.2 Presupuesto

Una vez mostrado el tiempo real que ha llevado realizar este gran trabajo, se puede calcular fácilmente el coste total real que significa y del presupuesto que se debe disponer para poder realizar uno similar. A continuación se van a mostrar los costes desglosados de los distintos tipos de costes para concluir con el coste total de realización. Cabe decir que para el formato de las tablas, me he inspirado en el TFE de David Gómez, Java'S Purgatory[29].

- Costes por personal humano

Aquí se pueden ver los costes representativos de las horas empleadas por el alumno y por el tutor, asumiendo el coste de horas del alumno como el de un ingeniero informático junior y el tutor un ingeniero informático senior en el mismo ámbito.

*Para calcular estos costes, mirando un ejemplo en <http://www.boe.es/boe/dias/2013/02/27/pdfs/BOE-A-2013-2203.pdf> lo que cobraría el ingeniero anualmente serían 31.212,66€ brutos anuales. Si la jornada laboral anual es de 52semanas a 8 horas por semana, con el cálculo de "52semanas*5días/semana*8horas/día" sacamos unas 2080horas anuales de trabajo. Si dividimos el salario anual entre las horas de trabajo, sacamos que el ingeniero informático titulado cobra 15,00€ la hora. El ingeniero senior que tendrá mucha más experiencia, aproximadamente cobrará un 50% más que el junior, 22,50€ la hora. Las horas del Ingeniero Junior, incluye todas las horas de análisis, codificación, pruebas y redacción de documentación.

Personal	Rol	Coste(€) por hora	Total de horas	Coste imputable(€)
Jorge Lucena Rodríguez	Ingeniero Junior	15,00	1005	15075,00
Alejandro Calderón Mateos	Tutor/Ingeniero Senior	22,50	30	675,00
Total				15750,00€

Tabla 30 Costes del proyecto por personal humano

- Costes por licencias de software

En la siguiente tabla residen los costes por adquisición del software utilizado, teniendo en cuenta que algunos eran gratuitos, otros tienen coste mensual y también hay licencias permanentes. Para calcular el coste imputable, se divide el coste absoluto entre el tiempo de vida útil y se multiplica por la duración del proyecto. Se estima la vida útil de un PC antes de ser obsoleto o retrasado en

tecnología a 5 años. Para el software sería aproximadamente lo mismo antes de necesitar la siguiente versión.

Licencia	Coste(€) absoluto	Vida útil	Tiempo usado	Coste(€) imputable
Windows 8	103,60	5años	11meses	19,00
Microsoft Office	79,00	5años	11meses	14,50
Gantt Project	0,00	5años	11meses	0,00
Audacity	0,00	5años	1mes	0,00
Camtasia Studio	135	5años	1mes	2,25
Adobe Premiere CC	24,19/mes- >1mes	1mes	1mes	24,19
Sculptris	0,00	5años	2meses	0,00
Cinema4D	695,00	5años	2meses	23,17
Gimp	0,00	5años	1mes	0,00
Unity3D normal	0,00	5años	11meses	0,00
Mozilla Firefox	0,00	5años	11meses	0,00
Total				83,11€

Tabla 31 Costes del proyecto por licencias de software

- Costes de hardware

A continuación se pueden ver distintos dispositivos físicos que han sido utilizados en la realización de este proyecto. Para calcular los costes imputables, se tiene en cuenta la vida útil del material físico y el tiempo utilizado.

Descripción	Coste(€) absoluto	Vida útil	Tiempo usado	Coste(€) imputable
Ordenador de sobremesa	1200,00€	5años	11meses	220,00€
Microfonía	15,00€	3años	1mes	0,40€
Total				220,40€

Tabla 32 Costes del proyecto por adquisición y uso de hardware

- Costes indirectos

No solo han existido costes tangibles y de software, también ha habido otros gastos indirectos como los siguientes.

Descripción	Coste(€) absoluto/mes	Tiempo total	Coste(€) imputable
Luz	10,50€/mes	11	115,50
Internet	60,00€/mes	11	660,00
Desplazamiento	89,50€/mes	11	984,50
Total			1760€

Tabla 33 Costes del proyecto por gastos indirectos

- Coste total

Finalmente aquí está la suma de los costes totales anteriores.

Descripción	Coste(€) Total
Personal	15750
Software	83,11
Hardware	220,40
Indirecto	1760,00
Total	17813,51€

Tabla 34 Costes totales del proyecto

Asimismo, una vez sumados los costes obtenemos que el presupuesto total necesario para este Trabajo de fin de grado ha sido de **Diecisiete mil ocho cientos trece con cincuenta y un Euros**, IVA del 21% incluido.

5 Conclusiones y trabajos futuros

Este capítulo está dividido en dos apartados principales con subdivisiones internas. En primer lugar, como el nombre del capítulo indica, se tratarán las conclusiones tras realizar el proyecto fijándose prioritariamente en si se han cumplido o no los objetivos. Dentro de las conclusiones, primero se repasarán los objetivos, luego los problemas más importantes para cumplir dichos objetivos y acabar citando aquellos escenarios de la carrera que han sido punto de apoyo para poder realizar el proyecto como tal. El segundo apartado principal trata acerca de todas las mejoras que se podrían realizar a este videojuego para ser más efectivo o ser usado durante más tiempo y otras opciones para, usándola misma base, poder reutilizar la idea para otros problemas de la actualidad.

5.1 Conclusiones

A continuación se muestran las conclusiones que se han sacado tras finalizar el desarrollo del proyecto. Se va a presentar primero las conclusiones sobre el sistema desarrollado, luego sobre el seguimiento del proceso software elegido, y finalmente (pero no menos importante) las conclusiones personales.

5.1.1 Sistema desarrollado

Una vez en este punto, ya se ha explicado la idea, creación y obtención de un videojuego para PC. Así, obtenido un producto para ofrecer a un usuario final, el cual tenga posibilidad de introducirse al mercado, se puede disfrutar de un videojuego con una interfaz muy simple, un uso muy sencillo y una carga educativa considerablemente alta. De este modo, la cantidad de conocimientos, costumbres, hábitos, gustos está adaptada a un público que pueda absorber con facilidad pudiendo así cumplirse el objetivo principal del proyecto.

La ventaja del uso de videojuego, es que es un contenido audiovisual por lo que se reciben datos visualmente, auditivamente y por dinámicas, así que adaptado a un público, es la herramienta ideal de transmisión de información. Este medio de comunicación es ideal para resolver el problema principal, objetivo de este proyecto.

Una vez repasado este dato, a continuación se volverán a repasar los objetivos de este proyecto, exponiendo si se han cumplido o no, comentando las causas del por qué y en qué grado se han cumplido:

- **La realización de un videojuego para PC, teniendo como objetivo un público muy joven pero que resulta muy atractivo para dicho público.**

Este objetivo se ha cumplido exitosamente, ya que, como se explicó en el capítulo3, sección1, se han seguido las principales partes del ciclo de vida de un videojuego. Empezando en la conceptualización, armando el esqueleto del producto, llegar a una versión alfa de un esqueleto con varios músculos, una

beta con todos los músculos y acabando en una versión final estable con piel y bien maquillada para ser lo más atrayente posible para el público infantil. Dada la ambientación que se ha podido otorgar por modelos 3D e interfaces 2D más las grabaciones amables enfocadas a este público, se puede decir que el grado de cumplimiento es excelente.

- **Fomentar intensamente a los jugadores como una dieta sana y equilibrada influye completamente en el crecimiento de los niños a lo largo del tiempo.**

Este otro objetivo también se cumple de una forma más indirecta y podría decirse, subliminal. Se pretende demostrar viendo a los torres de las propias duendes, un reflejo del propio niño si hubiese comido esa verdura, así que sólo por jugar, el niño inconscientemente ve a qué afecta cada alimento para el crecimiento. Junto a esto, los consejos de los mensajes que salen tras superar exitosamente cada ronda, también hacen referencia a esto mismo. El cumplimiento de este objetivo es satisfactorio en mayor parte por los consejos, que es dónde más directamente se transmite la información.

- **Facilitar a los padres una alternativa novedosa y llamativa para sus hijos, con la que puedan aprender divirtiéndose ocupando de esta forma parte de su tiempo libre.**

Para poder cumplimentar este objetivo, basta con difundir el videojuego depositándolo en algún repositorio online como los comentados del capítulo 2 sección 6 o bien entrega en mano. Difundiéndolo gratuitamente es como se lleva a cabo la realización de este objetivo, así que la parte previa a la difusión es la creación de un juego fácilmente portable con un simple ejecutable sin necesidad de instalación para poder disfrutar al instante. Debido a la ambientación, los niños quedan rápidamente engatusados con la temática del videojuego haciendo que los padres puedan respirar tranquilos mientras esta solución hace su magia. El grado de cumplimiento del objetivo depende del grado de difusión masiva que tenga el videojuego.

- **Concienciar a los niños sobre los beneficios de una dieta saludable, y los problemas que pueden surgir al no llevarla a cabo.**

Como último objetivo principal, cabe decir que también se ha llevado a cabo a la perfección. Esto se ha cumplido de dos formas. Por un lado, los consejos aleatorios que se escuchan y leen al acabar satisfactoriamente cada ronda de cada escenario, con palabras dulces y agradables informándole de qué le puede pasar si no tiene una dieta saludable. Por otro lado, al finalizar exitosamente se muestra como un niño sano se podría convertir al crecer y en caso de no acabar exitosamente, se mostrará al niño los problemas que pueden surgir en caso de no llevarla a cabo. El grado alcanzado en este objetivo podría mejorar, ya que

se ha centrado más en los otros objetivos ya que son visibles durante más tiempo y no sólo al final, pero del mismo modo se concienca a los niños.

Los objetivos personales están explicados en la parte de conclusión personal.

5.1.2 Proceso software elegido

A medida que el proyecto avanzaba, surgieron una gran cantidad de impedimentos que no permitían el correcto desarrollo del diagrama del Gantt pues producían un retraso considerable. Según estos impedimentos surgían y se solucionaban, evidentemente se fue aprendiendo de los errores y sacando conclusiones propias sobre qué se debe y qué no se debe hacer. Los problemas de esta índole más importantes son los siguientes:

- **Inicio de la mecánica TD.**

Siempre que se empieza a trabajar en un nuevo proyecto, siempre se empieza despacio hasta habituarse a la nueva idea. Si junto a esto sumas el desconocimiento de la plataforma que se está usando, el lenguaje de programación utilizado y que nadie de tu entorno puede ayudarte, a día de día, internet es tu amigo. El problema surgió debido a que, aunque el trabajo comenzó siguiendo tutoriales básicos de uso de interfaz, creación de escenarios, inserción de usuario en primera y/o tercera persona, cuando se llegó al tutorial de “cómo hacer un juego tipo TD”, vino el problema. El salto de conocimientos requeridos para poder empezar a realizar los pasos básicos sobre los que partir, tales como “cómo un elemento torre detecta en su radio de acción un elemento enemigo y debe dispararle”. Para poder hacer el “paso1” del tutorial, se necesitó al menos una semana para asimilar todos los conceptos base de elemento torre, elemento disparador, elemento disparado y elemento que recibe el disparo. Aunque antes de empezar parecía bastante simple el hecho de solo crear dos elementos que interactuasen entre sí, un paso tan sencillo como ese puede ser muy complejo. No se debe infravalorar la tarea.

- **Exportación del proyecto a plataforma Android.**

Aunque no se ha comentado hasta el momento, durante la concepción del juego también se pensó el exportar todos los datos del videojuego a un proyecto

Android para poder llegar a cuanta más gente, mejor. Como se explicó en el capítulo3 sección1, debido a la metodología empleada hubo cambios en los requisitos y se eliminó la idea del exportar el proyecto. Esto fue debido al módulo de inteligencia artificial utilizado con el algoritmo A* el cual estaba programado como una librería externa a Unity3D que no está preparada para Android. Este pequeño problema significó que completa inviabilidad de esta

idea. Ciertamente es que se buscaron posibles alternativas para hacer todo lo posible de crear un juego multiplataforma, pero el tiempo de realización de una propia inteligencia artificial para el movimiento de enemigos llevaría demasiado tiempo, lejos de la idea del proyecto que nunca ha querido centrarse en eso. Las librerías externas de un programa pueden estar muy bien, pero pueden traer problemas de compatibilidad.

- **Aspecto y uso de los duendes para transmitir la idea.**

Este problema está dentro del apartado de “Problemas de alimentación” del pensamiento principal del juego, ya que, para poder resolver el problema de la forma más eficiente posible, se debe llegar de una forma directa. Para poder llegar a esta forma, de una idea inicial se ha ido refinando la idea hasta llegar a la actual. Inicialmente todos los duendes iban a ser iguales, pero era mejor dividirlos en “buenos y malos” para que el niño asocie mejor la idea que lo “bueno” es lo que debe comer y lo “malo” es lo que no debe comer (nadie está prohibiendo nada, que no se deba no es que no se pueda). Una vez divididos por colores ambos tipos y dado un aspecto “maligno” a los “malos” faltaba información para que el niño visualmente entendiera qué era cada duende, por lo que se modificó el aspecto visual de los duendes “buenos” con iconos en el pecho. Estos iconos no eran suficientes para poder entender y diferenciar cada duende de su alimento, por lo que el siguiente punto de refinamiento era alterar el color del duende según su alimento para que, junto al icono se pudiese asociar un color a un alimento a una idea. El último punto de mejora, sucedido poco a poco con el tiempo alterando el calendario, pero con objetivo de mejorar la calidad del software, fue el de modificar físicamente la apariencia de cada duende “bueno”, para que de una forma más directa se pudiese apreciar lo que significaba el icono con su color y alimento. Los alimentos que ayudaban a crecer, están en duendes más corpulentos. Los alimentos que reflejan actitudes mentales o sensoriales, se puede ver reflejado en la cara de los propios duendes. Todas las características atribuidas son muy llamativas para que el niño rápidamente pueda asociar los conceptos y lo entienda como algo “bueno”. Todo esto finalmente se conjugó con los consejos auditivos que refuerzan toda esta información visual, que no estuvo pensada en un inicio, lo que provocó la ralentización del esquema del proyecto. Un proyecto para mejorar la calidad que ofrece, a veces debe cambiar todo su aspecto visual para para llegar mejor al consumidor.

5.1.3 Conclusiones personales

A lo largo de la realización de este proyecto, se han tenido que aprender muchísimos conocimientos nuevos, sin embargo, se ha podido llevar a cabo debido a una base asentada debido a ciertas asignaturas cursadas en esta carrera bajo el Plan de Informática 2008[31]. Debido a la naturaleza del proyecto (videojuego) sólo se han

aprovechado en torno a un 39,5% de asignaturas de la carrera, mientras que si la naturaleza hubiese sido distinta, por ejemplo de una herramienta dentro de una página web, se hubiesen utilizado muchas más. Las materias útiles y sus conocimientos más destacables usados son las siguientes ordenadas por cursos.

Primer curso:

- **Programación:** Esta asignatura podría considerarse como el centro de la carrera, ya que el aprender a programar, es el inicio del moldeo de la mente para abrirla hacia el mundo de la programación. Aunque la mayoría de los conocimientos que se muestran aquí no se consolidan hasta cursar otras asignaturas, la base de la programación es la base de la carrera.
- **Álgebra Lineal:** De cara al mundo de los videojuegos, el conocimiento y trabajo de matrices es considerado el “pan de cada día”. El manejo inicial y avanzado de estas es muy importante de cara al mundo de 3dimensiones.
- **Técnicas de búsqueda y uso de la información:** Durante el proceso de aprendizaje de cualquier tema, en esta era tecnológica, el hecho de poder acceder a internet de forma eficiente, con búsquedas precisas y atajos para localizar la información exacta, son la clave de la resolución de problemas. Aquí también se enseña a utilizar la biblioteca de la universidad y a usar y referenciar toda la información que contiene.
- **Principios de la Ingeniería Informática:** El concepto fundamental sobre el que se hizo hincapié cada día, fue el de tratar adecuadamente las tareas tipo elefante. El TFG es una tarea elefante que hay que saber desmenuzar muy bien para poder abordarla por partes de forma adecuada para que no le supere el peso de la tarea al desarrollador.

Segundo curso:

- **Técnicas de Verificación Software:** Como continuación de asignatura acerca de la Ingeniería del Software, aquí se tratan temas principalmente acerca de cómo hacer pruebas, documentar y notificar pruebas. Todo código antes de darse por correcto, ha de haber pasado unas pruebas que corroboren que sus funcionalidades se ejecutan a la perfección.
- **Sistemas Operativos:** Dejando a un lado el hecho de que se utilizó otro lenguaje de programación, se trataron temas muy importantes en el mundo de la computación como son los procesos pesados y los procesos ligeros. Aunque aquí fue el inicio, la introducción a los threads es vital pues Unity3D también puede usarlos para tareas paralelas.
- **Inteligencia Artificial:** De cara al mundo de los videojuegos, la configuración de la IA oponente o amiga siempre ha sido un punto fundamental. Aunque aquí no se trate muy a fondo ningún tema de creación y configuración de una IA, sí se

trata el término de heurística y los algoritmos de búsqueda como el A* utilizado para la mecánica principal de un TD.

- **Interfaces de Usuario:** Obviando la parte práctica de la asignatura dedicada a web, todo el tema de interfaces, colores, formas, lugares y los principios de Nielsen son información teórica extremadamente fundamental para este tipo de aplicaciones software.

Tercer curso:

- **Procesos del Desarrollo del Software:** Esta asignatura sí detalla en gran medida todo el ámbito de la Ingeniería del Software acerca de la toma requisitos y la asignación y división de funcionalidades en un proyecto. Respecto a la parte teórica de un proyecto, esta asignatura es un punto de apoyo muy importante.
- **Arquitectura de Computadores:** El tema de paralelismo de tareas con pool de threads es algo que se acaba de asentar con esta asignatura, algo muy utilizado en el motor gráfico usado. La parte de explicación del funcionamiento de componentes hardware de un PC también es útil de cara a este tema.
- **Diseño de Sistemas Operativos:** Algo transversal a la creación de videojuegos sería la orientación a la multiplataforma, lo que implica poder gestionar la misma información en distintos sistemas operativos. De cara a utilizar hardware extra para el videojuego, también es un aliciente interesante el poder configurar un driver para mejorar la funcionalidad del sistema.

Cuarto curso:

- **Dirección de Proyectos de Desarrollo de Software:** Esta asignatura es el colofón de la Ingeniería del Software pues trata de simular una empresa que lleva a cabo un proyecto con absolutamente toda la documentación necesaria para este. Desde el acta de constitución de la empresa, siguiendo por la toma de requisitos del proyecto con el cliente, especificación de funcionalidades y todo el desarrollo posterior.
- **Informática Gráfica:** Como principio al mundo del modelado 3D, este es un buen comienzo, aunque tampoco se llegaron a utilizar herramientas serias y se tuvo que aprender de otras para este proyecto. Igualmente, la parte teórica del modelado 3D con todo el tratamiento de matrices es algo importante también.
- **Sistemas de Tiempo Real:** Debido a que un videojuego se juega en tiempo real, esta asignatura cobra sentido con el proyecto. Un videojuego transcurre frame a frame sin pausa como si fuese una película, de forma similar a las prácticas de la asignatura. Desde que arranca la aplicación, no hay forma de parar la maquinaria.
- **Multimedia:** Un videojuego está dentro del mundo audiovisual, lo que equivale al uso de tecnología multimedia. Multimedia equivale al trato de sonidos e imagen estática o imagen en el tiempo (video). Aunque en principio no hace

falta conocer toda la especificación teórica del tratamiento multimedia para un videojuego, siempre es un punto importante para hacer bien las tareas conociendo lo que se maneja.

Respecto a los objetivos personales de este proyecto, siguiendo el mismo modelo que con los objetivos principales del proyecto ya redactados:

- **Comprobar que un alumno de grado es capaz de llevar a cabo un proyecto en solitario de principio a fin en un entorno de trabajo nuevo y manejando tecnologías nuevas.**

Una vez llegado a este punto de la memoria, se puede entender que este objetivo ya se ha cumplido tras un largo periodo de estudio, aprendizaje, trabajo, concluyendo con la elaboración de esta sección de esta memoria. No ha sido una tarea fácil, ya que la parte de aprendizaje de nuevo software con nuevas interfaces buscando información exclusivamente en internet ha sido la parte más difícil ya que entrar de cero en un nuevo mundo, no sólo nuevos lenguajes de programación, es muy pesado y los días pasan solos. Como apartado personal, pienso que se ha aprendido mucho de llevar a cabo un proyecto, aunque corto, entero ya que a lo largo de la carrera se han ido haciendo sólo módulos de grandes aplicaciones. Por ejemplo, haber creado un driver para un sistema operativo, hacer una web básica simulando una real o bien defender un servidor Linux idéntico a uno real, pero sin haberlo creado y configurado previamente.

- **Aportar algo que merezca la pena a la industria del videojuego.**

Aunque se puede decir que este objetivo personal se ha cumplido, personalmente no del todo, ya que para hacer un trabajo decente, hace falta un equipo de personas especializadas cada uno en su tema. Para este proyecto debería haberse contado con un diseñador gráfico y/o modelador 3D ya que yo he tenido que aprender de 0 tecnologías de este mundo y he hecho lo que he podido. También haría falta un especialista en sonidos y/o músico para la banda sonora y efectos de audio, en lugar de los sonidos obtenidos de webs con política *Creative Common* libre, aunque debido a haber trabajado durante el proceso en el departamento de audiovisuales esto no ha supuesto muchos problemas. Yo hubiera hecho la parte desarrollador, dedicándome a unificar todos los módulos en Unity3D y manipulándolos adecuadamente bajo líneas de código. A pesar de este dato, se ha realizado un juego considerable teniendo en cuenta desde donde se ha partido y hasta donde se ha llegado. Más tarde en futuros trabajos estarán redactados detalles que debido a varios factores no se han llegado a realizar y se han dejado para un futuro.

5.2 Trabajos futuros

En esta sección se detallarán tareas que se podrían llevar a cabo en un futuro para mejorar la calidad del producto.

5.2.1 Tutorial

A pesar de la escasa dificultad del juego, un nuevo escenario ultra-reducido en tamaño solo para utilizar las funciones básicas de “Comenzar nivel”, “Colocar torre”, “Evolucionar torre”, “Eliminar enemigo” sería un punto inicial importante. En el proyecto actual se explica con un audio ligeramente el funcionamiento del sistema y con el Apéndice1 del manual de usuario se puede ver la información mejor redactada. El tutorial debiera consistir en un escenario previo al escenario1, que repase el funcionamiento de las funciones recién comentadas, también explique todas las partes de la interfaz visible y de la emergente como el inicio de ronda. Para que los conceptos se puedan asimilar de una forma asequible, debe tener gran cantidad de pausas y confirmaciones del usuario sobre su entendimiento. Cada función básica y parte de interfaz debiera tener su grabación de audio que lo explique, su texto adjunto en pantalla para quien sepa leer y las confirmaciones paso a paso para cada función.

5.2.2 Escenarios

Aumentar la cantidad de escenarios, con la dificultad que implica ya no solo por la ambientación que guste y que también transmita información del problema a resolver, sino también por la de la generación de recorridos por el escenario. Esto significa hacer gran cantidad de pruebas para que ningún enemigo se quede atascado a lo largo del recorrido, colisionen entre sí por distintas velocidades o se pierdan si existen múltiples recorridos simultáneos. Este cambio ayuda a aumentar la duración del videojuego permitiendo a los niños jugar durante más tiempo antes de cansarse de esta alternativa lo que indirectamente implica que, no solo disfrute de un tiempo maravilloso con los duendes, sino que también adquiera más conocimientos útiles para poder formarse como una persona más sana sin problemas de alimentación.

5.2.3 Modo historia

Para una excelente motivación infantil, el apartado de historia/trama/hilo argumental es casi el factor más importante del propio videojuego. Actualmente existe una breve motivación que te guía y ayuda con consejos en cada escenario, sin embargo, no tiene un ente físico que el niño pueda referenciar bajo un aspecto específico. Un modo historia, junto al personaje guía específico, un ente enemigo específico y la historia que se quiera montar, de cara a montaje de videojuego consistiría en añadir distintas dinámicas al juego actual. En primer lugar, para poder marcar las pautas de la historia, todos los escenarios menos el primero (por ejemplo, el tutorial) deberían estar bloqueados y, a medida que el usuario vaya superando los escenarios disponibles en orden, irá desbloqueando los siguientes escenarios hasta llegar al final y completar la historia. Para asimilar de mejor forma la mecánica del juego, la dificultad de los escenarios debiera ser gradual, siendo el primer escenario

muy sencillo con pocas rondas y los últimos más complicado, con menos posibilidad de torres, mapas con recorridos más directos con su consecuente mayor dificultad escalada por la historia. A medida que se avanza, igualmente se podrían ir desbloqueando más torres o más niveles de evolución de cada torre, más enemigos para mostrar progresión y que el niño sienta que avanza y de verdad puede hacer más cosas debido al uso de verduras recibiendo otro refuerzo positivo más. Para poder realizar eficientemente este trabajo futuro, sería muy recomendable también realizar los otros trabajos anteriores comentados en el capítulo 6 sección 2.

5.2.4 Visualización de efectos al superar o no un nivel

Un aspecto visual muy importante pero que requiere un alto nivel de trabajo es el de crear una animación para las pantallas de victoria y derrota. Esta animación según el caso sería similar pero tendría matices diferenciadores.

En caso de victoria, la animación dependería de cómo ha jugado el usuario y se reflejaría sobre el elemento del juego “niño”. Según las torres que se hayan colocado a lo largo del escenario para combatir a los duendes enemigos, al salir victorioso, se debería visualizar una animación de cómo el elemento niño ingiere todas las verduras que tienen los duendes en los platos. La animación cambiaría según aquello que coma pues, la idea es ver al niño crecer por comer esos alimentos, por lo que si el usuario sólo ha utilizado verduras de crecimiento corporal, el elemento niño crecería proporcionalmente a las verduras que ingiera. Si ingiere verduras de crecimiento corporal y de visión, la animación sería de cómo el elemento niño crece proporcionalmente al número de verduras de crecimiento y también le aumentaría la visión proporcionalmente al número de verduras de visión. Si usa de todas, el elemento niño final crecería en todos los aspectos de los atributos que aportan esas verduras que ingiere proporcionalmente a la cantidad. Este final es la mejor forma posible para que el usuario asimile de forma directa qué le pasa a su igual, en consecuencia a él mismo si comiera esa dieta sana. En caso de, por ejemplo, no comer ninguna verdura de crecimiento pero sí del resto, se podrá visualizar cómo de mayor el elemento niño crece en todo menos tamaño. Los resultados del crecimiento de un niño no sólo dependen de su alimentación, pero la idea a transmitir es que la alimentación es un punto vital, así que este juego asume que el crecimiento SOLO depende de la alimentación.

En caso de derrota, la animación sería siempre la misma, el cómo a un niño sano se le hinchan los mofletes, se le caen los dientes y enferma mucho debido a ello, dejando a un lado que “no crece” con el paso del tiempo. En esencia sería la transición del elemento niño a la imagen de la pantalla de derrota. Esto también sirve como refuerzo negativo, ya que aun siendo malo lo que se muestra, se lo pueden tomar con buen humor y adquirir la idea.

5.2.5 Multiplataforma

Para poder conseguir la máxima difusión de este software, sería muy importante conseguir la migración de este proyecto al del navegador web, Android e iOS. Si esto no se ha hecho, ha sido debido a la incompatibilidad de Unity3D con la librería utilizada para el recorrido de la IA a través de los nodos del mapa. Para poder lograr que Unity3D permita la migración del proyecto a estas otras plataformas, primero habría que remover todo el uso de esta librería externa, realizar lo mismo con funciones nativas de Unity3D como los navmesh y ya a partir de ahí exportar y realizar todas las pruebas de funcionamiento en las nuevas plataformas.

5.2.6 Otros problemas

Puesto que toda la estructura e idea están montadas para niños pequeños, siguiendo la misma temática pero cambiando la apariencia gráfica y los modelos 3D de los elementos visibles, se podría orientar todo el videojuego a resolver otro posible problema de niños pequeños.

Apéndice I: Manual de uso para usuario genérico

Esta documentación está pensada para ser leída por los padres y transmitida a sus hijos ya que se asume en todo momento que los niños aunque tengan alguna noción de lectura como reconocer números o letras no poseen aún la suficiente comprensión lectora para asimilar un manual. La siguiente información data acerca de cómo se juega.

Una vez dentro de la aplicación, existen 3 botones para poder navegar por la interfaz:

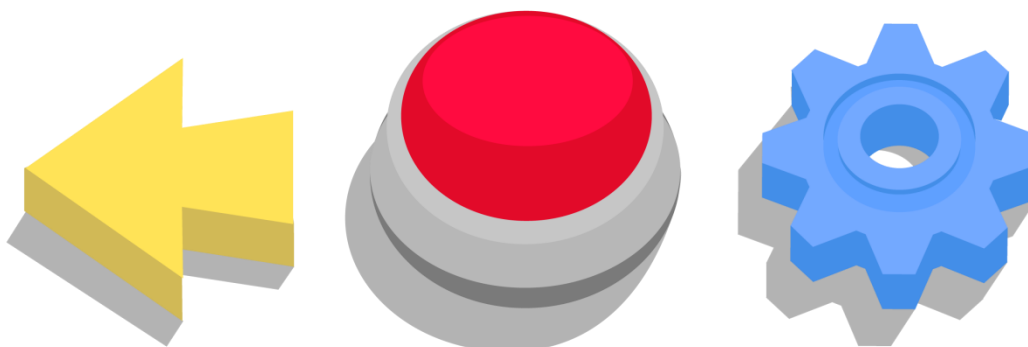


Imagen 72 Botones 1, 2 y 3 del menú principal

El primer botón con forma de flecha, el botón “Salir”, sirve para cerrar la aplicación.

El segundo botón con forma de botón rojo, el botón “comenzar”, es necesario para continuar avanzando por la interfaz principal seleccionando opciones hasta llegar a un escenario de juego. Una vez pulsado “comenzar”, se permitirá al usuario elegir en cuál de los tres escenarios desea vivir aventuras el usuario. Una vez escogido el escenario, antes de pasar a la acción se preguntará por el género de usuario para así poder hacer que el usuario se sienta más identificado aún con lo que está jugando.

El tercer botón con forma de engranaje, el botón “configuración”, conduce al jugador a un pequeño menú con la posibilidad de desactivar o activar todo tipo de efecto sonoro en la aplicación. Hecho esto se puede volver al menú principal.

Para jugar dentro de cada escenario, lo único que hay que hacer es comprar duendes con el dinero virtual del propio juego y colocarlos en las torres destacadas en la pantalla. El juego consiste en rellenar las torres con duendes y evitar que los enemigos crucen el laberinto y el jugador pierda sus vidas. La interfaz del escenario está dividida en tres zonas bien diferenciadas.

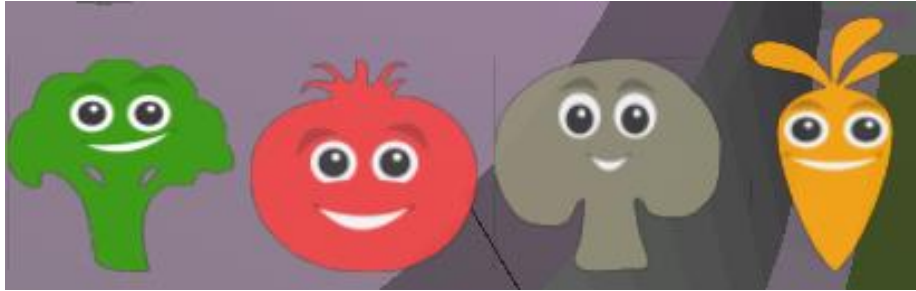


Imagen 73 Esquina inferior izquierda de la interfaz de juego

En la esquina inferior izquierda se pueden ver los cuatro duendes que se pueden escoger, que varían según cada escenario. Para usarlos se ha dicho que hay que hacer clic en el icono del alimento del duende y posteriormente hacer clic en la casilla de una torre disponible.



Imagen 74 Esquina inferior derecha de la interfaz de juego

En la esquina inferior derecha residen los datos “numéricos” del estado del juego. Esto es, la cantidad de duendes que puedes colocar en torres siendo esto el dinero virtual, la cantidad de vidas que aún tiene el usuario y el progreso, en forma de barra creciente, para acabar satisfactoriamente el escenario. A medida que se colocan los duendes con las verduras de la esquina inferior izquierda, en esta esquina se podrá visualizar como disminuye el número de duendes reflejados bajo el icono de un gorro y un número.



Imagen 75 Botones comenzar oleada y mejorar duende de la interfaz de juego

El resto de la pantalla es, al mismo tiempo, el escenario que se haya elegido de juego y la región de los mensajes emergentes. Para poder comenzar con una oleada de enemigos inicialmente se puede ver un botón que dice “¡Comenzar!” siendo el indicativo de su significado. Es muy importante haber situado a varios duendes en las torres antes de iniciar la primera oleada ya que por consiguiente los enemigos recorrerán el laberinto y llegarán al niño/a escogido en el menú principal con lo que se perderán vidas. Al acabar una oleada, se escuchará y se podrá leer un consejo sobre el propio juego pudiendo ser sólo información o consejos sobre la mecánica. La última opción que se puede hacer en este fragmento de interfaz se confiesa en un consejo, que es la posibilidad de mejorar a los duendes de las torres. Si se hace clic en el duende de una torre y se tiene suficiente dinero virtual, ese duende seleccionado se podrá mejorar obteniendo un incremento considerable en sus estadísticas y un renovado aspecto visual.

En caso de salir victorioso del escenario, al usuario se le conducirá al escenario de victoria con las opciones de salir del juego o bien volver al menú principal.

En caso de no salir victorioso del escenario, al usuario también se le conducirá a un escenario de derrota, con las mismas dos opciones que el escenario de victoria.

Consejos

Se recomienda comenzar la partida situando cuatro duendes en cuatro torres en zonas estratégicas. Las zonas estratégicas son aquellas zonas en las que los enemigos caminan durante más tiempo, por ejemplo zonas con forma de S, o de C, estando estas generalmente en el centro del mapa.

Una vez ocupadas las zonas estratégicas, en lugar de rellenar las torres de los extremos es recomendable mejorar los mismos duendes que se situaron en un inicio y continuar mejorándolos en el mismo orden que se colocaron. Se tarda menos en acabar un escenario que en reunir suficiente dinero para rellenar todas las torres o mejorarlas todas.

Bibliografía

- [1] Vallet, M. (2007). *Educar a niños y niñas de 0 a 6 años*. Madrid: Wolters Kluwer Educación.
- [2] *Documental de alimentación de Telemadrid*. (s.f.). Recuperado el 05 de 10 de 2014, de Youtube: <http://www.youtube.com/watch?v=6wMZDokA3mA>
- [3] *Programa de televisión Supernanny*. (s.f.). Recuperado el 07 de 10 de 2014, de Cuatro: <http://www.cuatro.com/supernanny>
- [4] *Unity*. (s.f.). Recuperado el 10 de 03 de 2014, de Unity3d: <http://unity3d.com/learn>
- [5] *Unity Game Engine Tutorials and Resources*. (s.f.). Recuperado el 15 de 04 de 2014, de CGCookie: <https://cgcookie.com/unity/category/tutorials>
- [6] *Extreme Programming*. (s.f.). Recuperado el 24 de 06 de 2014, de Agile Process: <http://www.extremeprogramming.org>
- [7] Nicolás Acerenza, A. C. (2009). Una Metodología para Desarrollo de Videojuegos. Recuperado el 13 de 05 de 2014, de Academia: http://www.academia.edu/453193/Una_Metodolog%C4%B1a_Para_Desarrollo_De_Videojuegos
- [8] *Ley Orgánica 15/1999, de 13 de diciembre, de Protección de Datos de Carácter Personal*. (13 de 12 de 1999). Recuperado el 21 de 10 de 2014, de Agencia Estatal Boletín Oficial del Estado: http://www.boe.es/diario_boe/txt.php?id=BOE-A-1999-23750
- [9] *Real Decreto Legislativo 1/1996, de 12 de abril de la Ley de Propiedad Intelectual*. (12 de 04 de 1996). Recuperado el 21 de 10 de 2014, de Agencia Estatal Boletín Oficial del Estado: http://www.boe.es/diario_boe/txt.php?id=BOE-A-1996-8930
- [10] *Ley 34/1988, de 11 de noviembre, General de Publicidad*. (15 de 11 de 1988). Recuperado el 21 de 10 de 2014, de Agencia Estatal Boletín Oficial del Estado: http://www.boe.es/diario_boe/txt.php?id=BOE-A-1988-26156
- [11] *Documentación para Métrica v.3*. (s.f.). Recuperado el 02 de 11 de 2014, de PAE portal administración electrónica del gobierno de España: http://administracionelectronica.gob.es/pae/Home/pae_Documentacion/pae_Metodolog/pae_Metrica_v3.html#.VODPgS5-bzI
- [12] *Audacity Software*. (s.f.). Recuperado el 27 de 11 de 2014, de Audacity: <http://audacity.sourceforge.net/about>
- [13] *Adobe Audition CC*. (s.f.). Recuperado el 27 de 11 de 2014, de Adobe Creative Cloud: <https://creative.adobe.com/es/products/audition>
- [14] *Camtasia*. (s.f.). Recuperado el 27 de 11 de 2014, de Camtasia: <http://www.techsmith.com/camtasia.html>
- [15] *Adobe Premiere CC*. (s.f.). Recuperado el 28 de 11 de 2014, de Adobe Creative Cloud: <http://www.adobe.com/es/products/premiere.html>

- [16] *Sculptris Software*. (s.f.). Recuperado el 2 de 10 de 2014, de Pixologic:
<http://pixologic.com/sculptris>
- [17] *Cinema 4D software*. (s.f.). Recuperado el 2 de 10 de 2014, de Maxon:
<http://www.maxon.net/es/products/cinema-4d-studio.html>
- [18] *Blender software*. (s.f.). Recuperado el 2 de 10 de 2014, de Blender:
<http://www.blender.org/get-involved>
- [19] *3DS MAX*. (s.f.). Recuperado el 2 de 10 de 2014, de Autodesk:
<http://www.autodesk.es/products/3ds-max/overview>
- [20] *Maya*. (s.f.). Recuperado el 2 de 10 de 2014, de Autodesk:
<http://www.autodesk.es/products/maya/overview>
- [21] *Paint*. (s.f.). Recuperado el 16 de 10 de 2014, de Windows.microsoft:
<http://windows.microsoft.com/es-es/windows7/getting-started-with-paint>
- [22] *Gimp*. (s.f.). Recuperado el 14 de 10 de 2014, de Gimp: <http://www.gimp.org.es>
- [23] *Adobe Photoshop CC*. (s.f.). Recuperado el 16 de 10 de 2014, de Adobe Creative Cloud:
<http://www.adobe.com/es/products/photoshop.html>
- [24] F, Fabri. (22 de 08 de 2013). *¿Qué es un motor gráfico 3D?* Recuperado el 08 de 11 de 2014, de Articulos.softonic: <http://articulos.softonic.com/que-es-un-motor-grafico-o-motor-3d>
- [25] *Adobe Flash Professional CC*. (s.f.). Recuperado el 11 de 03 de 2014, de Adobe Creative Cloud: <http://www.adobe.com/es/products/flash.html>
- [26] *Wimi5*. (s.f.). Recuperado el 12 de 03 de 2014, de Wimi5: <http://wimi5.com/la-plataforma>
- [27] *Unreal Engine 4*. (s.f.). Recuperado el 13 de 03 de 2014, de Unreal Engine:
<https://www.unrealengine.com/what-is-unreal-engine-4>
- [28] Muñoz, M. V. (19 de 05 de 2011). *Diseño e implementación de un juego para PC mediante Ogre 3D*. Recuperado el 14 de 11 de 2014, de e-Archivo.uc3m: <http://e-archivo.uc3m.es/handle/10016/11842>
- [29] Durán, D. G. (08 de 10 de 2013). *"Java's purgatory" : desarrollo de un juego educativo en Unity3D*. Recuperado el 14 de 11 de 2014, de e-Archivo.uc3m: <http://e-archivo.uc3m.es/handle/10016/18606>
- [30] *GanttProject*. (s.f.). Recuperado el 28 de 11 de 2014, de GanttProject:
<http://www.ganttproject.biz>
- [31] *Programa Grado en Ingeniería Informática (2008)*. (24 de 03 de 2009). Recuperado el 18 de 12 de 2014, de Universidad Carlos III de Madrid:
http://portal.uc3m.es/portal/page/portal/titulaciones_grado/infor/plan_2008