

# DOG ON THE RUN

**Carlos A. Murcia García**

Grado en Ingeniería Informática  
Trabajo Fin de Grado - Videojuegos

**Consultor: Guillermo García Romero**

**Profesor responsable de la asignatura: Joan Arnedo Moreno**

3 de Enero de 2021



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

Copyright © 2020 Carlos Murcia.

Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.3 or any later version published by the Free Software Foundation; with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts.

A copy of the license is included in the section entitled "GNU Free Documentation License".

### **C) Copyright**

© (Carlos Murcia García)

Reservados todos los derechos. Está prohibido la reproducción total o parcial de esta obra por cualquier medio o procedimiento, comprendidos la impresión, la reprografía, el microfilme, el tratamiento informático o cualquier otro sistema, así como la distribución de ejemplares mediante alquiler y préstamo, sin la autorización escrita del autor o de los límites que autorice la Ley de Propiedad Intelectual.

## FICHA DEL TRABAJO FINAL

<b>Título del trabajo:</b>	<i>Dog on the run</i>
<b>Nombre del autor:</b>	<i>Carlos A. Murcia García</i>
<b>Nombre del consultor/a:</b>	<i>Guillermo García Romero</i>
<b>Nombre del PRA:</b>	<i>Joan Arnedo Moreno</i>
<b>Fecha de entrega (mm/aaaa):</b>	01/2021
<b>Titulación:</b>	<i>Grado en Ingeniería Informática</i>
<b>Área del Trabajo Final:</b>	<i>Trabajo Fin de Grado - Videojuegos</i>
<b>Idioma del trabajo:</b>	<i>Castellano</i>
<b>Palabras clave</b>	<i>Plataformas, 2D</i>
<b>Resumen del Trabajo (máximo 250 palabras):</b> <i>Con la finalidad, contexto de aplicación, metodología, resultados i conclusiones del trabajo.</i>	
<p>Dog On The Run es un videojuego de plataformas 2D en el que los protagonistas del juego deben encontrar a su mascota perdida, pasando por números desafíos y pruebas hasta dar con ella.</p> <p>El videojuego tiene una dinámica parecida a Mario Bros o Alex Kidd cuyo objetivo es intentar pasar el nivel interactuando con los diferentes enemigos y plataformas que se encuentra en el camino.</p> <p>Para la elaboración del videojuego se ha utilizado el motor Unity y el lenguaje de programación C. El entorno de programación que hemos utilizado para realizar el videojuego ha sido Visual Studio.</p> <p>Hemos pasado por las diferentes fases del proyecto, encontrando problemas en el camino e intentándolos solucionar de la mejor forma posible. A pesar de la falta de tiempo y de la situación que estamos viviendo en estos momentos se han desarrollado gran parte de los elementos planificados y aunque algunos de estos elementos se podían haber mejorado, hemos intentando ceñirnos a la idea inicial y finalizar el proyecto con éxito.</p> <p>En Conclusión, al principio del proyecto hicimos una valoración y un diagrama de las tareas que debíamos desarrollar para realizar la idea que teníamos pensada y al final de este desarrollo podemos afirmar que tenemos un proyecto donde el usuario final podrá experimentar de un juego terminado y de que se ha cumplido con gran parte de las expectativas.</p>	

**Abstract (in English, 250 words or less):**

Dog On The Run is a 2D platform video game in which the protagonists of the game must find a lost pet, going through many challenges and tests until they find it.

The video game has a dynamic similar to Mario Bros or Alex Kidd whose objective is to try to pass the level by interacting with the different enemies and platforms that are on the way.

For the development of the video game, the Unity engine and the C programming language have been used. The programming environment that we have used to make the video game has been Visual Studio.

We have gone through the different phases of the project, finding problems along the way and trying to solve them in the best possible way. Despite the lack of time and the situation we are experiencing at the moment, a large part of the planned elements have been developed and although some of these elements could have been improved, we have tried to stick to the initial idea and finish the project successfully.

In conclusion, at the beginning of the project we made an assessment and a diagram of the tasks that we should develop to carry out the idea that we had in mind and at the end of this development we can affirm that we have a project where the end user can experience a finished game and that Much of the expectations have been met.

## Índice

1. Introducción.....	1
1.1 Contexto y justificación del trabajo .....	1
1.2 Objetivos del Trabajo.....	1
1.3 Enfoque y método seguido.....	1
1.4 Planificación del Trabajo .....	2
1.5 Breve resumen de productos obtenidos .....	4
1.6 Breve descripción de los otros capítulos de la memoria.....	4
2. Estado del arte .....	6
2.1 Género del Juego.....	6
2.2 Revisión de la Tecnología .....	6
3. Definición del juego .....	9
3.1 Idea del juego.....	9
3.1.1 Descripción del juego.....	9
3.1.2 Interacción jugador – juego.....	9
3.2.3 Plataforma de destino .....	9
3.2 Conceptualización .....	10
3.2.1 Historia y ambientación.....	10
3.2.2 Objetivos planteados al jugador.....	10
3.2.3 Definición de los personajes y elementos .....	10
4. Diseño técnico.....	13
4.1 Entorno de desarrollo .....	13
4.2 Requerimientos técnicos del entorno .....	13
4.3 Herramientas utilizadas en el desarrollo.....	14
4.4 Inventario y recursos del juego.....	15
4.5 Arquitectura y diseño del videojuego.....	17
4.6 Funcionamiento de la IA.....	18
4.7 Head-Up Display (HUD).....	18
5. Diseño de niveles .....	19
5.1 Introducción .....	19
5.2 Nivel 1 (Exterior).....	19
5.3 Nivel 2 (Interior).....	20
6. Manual de usuario.....	21
6.1 Requisitos del juego .....	21
6.2 Controles .....	21
6.3 Instrucciones de ejecución.....	21
6.4 Menús del juego .....	22
6.4.1 Menú Principal .....	22
6.4.2 Menú Options.....	22
6.4.3 Menú Sound.....	22
6.4.4 Menú Controls.....	23
6.4.5 Menú Select.....	23
6.4.6 Menú Pause.....	23
7. Conclusiones.....	24
8. Glosario .....	25
9. Bibliografía .....	26
10. Pruebas de usuario y enlaces de interés.....	27
10.1 Pruebas de usuario .....	27
10.2 Enlaces de interés .....	27

## Lista de figuras

Ilustración 1. Planificación de entregas (Diagrama de Gantt)	4
Ilustración 2: Ejemplos de videojuegos	6
Ilustración 3: Unity	7
Ilustración 4: Unreal	7
Ilustración 5: GameMaker	8
Ilustración 6: Personaje femenino	10
Ilustración 7: Personaje masculino	11
Ilustración 8: Mascota perdida	11
Ilustración 9: Zombies	11
Ilustración 10: Animación de fantasma	12
Ilustración 11: Animación de murciélago	12
Ilustración 12: Monedas	12
Ilustración 13: Spikes	12
Ilustración 14: Fuego	12
Ilustración 15: Visual Studio	14
Ilustración 16: Paint 3D	14
Ilustración 17: Audacity	14
Ilustración 18: Microsoft Office	14
Ilustración 19: Github	15
Ilustración 20: Diferentes assets	16
Ilustración 21: Sonidos	16
Ilustración 22: Fuente	16
Ilustración 23: Diagrama de flujo	17
Ilustración 24: Lógica de movimiento	18
Ilustración 25: HUB	18
Ilustración 26: Nivel 1	19
Ilustración 27: Nivel 2	20
Ilustración 28: Menú Principal	22
Ilustración 29: Menú Options	22
Ilustración 30: Menú Sound	22
Ilustración 31: Menú Controls	23
Ilustración 32: Menú Select	23
Ilustración 33: Menú Pause	23

# 1. Introducción

## 1.1 Contexto y justificación del trabajo

La industria de los videojuegos se ha convertido en una de las industrias más importantes del mundo en los últimos años. El acceso a los videojuegos y a las nuevas plataformas implantadas han sido sin duda algunas de las características más importantes para este cambio.

Hoy en día, el diseño de un videojuego tiene muchos componentes, desde el diseño del propio videojuego, hasta las artes gráficas que lo componen pasando por los profesionales del sonido y finalizándolo con la programación del mismo.

Lo que se ha querido transmitir en este trabajo es el aprendizaje de todos estos años, el saber desenvolverse en situaciones adversas en un proyecto en que se unen diferentes componentes y documentar todo el proyecto en este documento.

## 1.2 Objetivos del Trabajo

Los diferentes objetivos han sido:

- Crear un guion atractivo para el usuario final.
- Escoger las herramientas necesarias.
- Gestionar el tiempo y las herramientas disponibles.
- Implementar las ideas.
- Desarrollar un videojuego totalmente jugable.

## 1.3 Enfoque y método seguido

En una empresa convencional, normalmente existen diversos roles en la creación y desarrollo de un videojuego. En esta ocasión todos los roles necesarios para la implementación del proyecto han sido ocupados por la misma persona, cosa que ha dificultado el control y el desarrollo del mismo.

Se decidió desde un primer momento, que la idea principal del videojuego a desarrollar era una idea basada en un juego de plataformas 2D. Estando seguros de que no íbamos a crear nada novedoso en este caso, intentamos

que fuera algo diferente y entretenido, obstaculizando el camino del personaje lo máximo posible.

En lo personal tengo que decir, que creo que hemos elegido una estrategia adecuada a los tiempos que estamos viviendo en la actualidad y a los tiempos necesarios que se necesitan para implementar un videojuego. Ha sido apasionante aprender y poder demostrar los recursos aprendidos en cada una de las etapas de este proyecto.

#### 1.4 Planificación del Trabajo

La planificación de este proyecto ha sido diseñada a través de un listado de tareas iniciales que posteriormente se plasmó en un diagrama de Gantt. En dicho diagrama se ha especificado los objetivos que teníamos en cada tarea y las fechas señaladas en cada una de ellas. Se ha utilizado diferentes colores para identificar las diversas entregas que teníamos programadas.

##### PEC1

- Ideas del juego
  - o Descripción
  - o Definición del género
  - o Definición de la historia
- Conceptualización
  - o Definición de los personajes
  - o Interacción entre los personajes
  - o Objetivos planteados a los jugadores
- Desarrollo y planificación
  - o Evaluación de los diferentes engines
  - o Planificación y Diagrama de Gantt

##### PEC2

- Creación de cuenta GitHub y Unity
  - o Creación de los diferentes repositorios para las distintas versiones
- Obtención de los assets
  - o Búsqueda de los assets
  - o Personalización
- Obtención de los recursos
  - o Búsqueda de los recursos
  - o Personalización
- Creación de escenas
  - o Personalización de las diferentes escenas
  - o Definición de jugabilidad



- Creación de menús
  - o Personalización de los diferentes menús
- Revisión de jugabilidad y mejoras

### PEC3

- Desarrollo de personajes principales
  - o Definición de movimientos
  - o Definición de características
- Desarrollo de enemigos
  - o Definición de movimientos
  - o Definición de características
- Desarrollo de scripts del juego
  - o Diseño de las diferentes etapas del juego
  - o Mejora del realismo del juego
- Creación de nivel
  - o Creación del primer nivel de jugabilidad
  - o Si fuera preciso, diseño de siguientes niveles
- Composición de los sonidos
  - o Creación y diseños de los sonidos de los personajes e ítems
  - o Música de entorno
  - o Sonidos varios
- Creación de ítems
  - o Diseño de las variantes de los ítems
  - o Creación de la puntuación obtenida
- Testeo y mejoras
  - o Testeo del juego con varias repeticiones y partidas.
  - o Mejoras de los errores que se detecten

### PEC4

- Cambios y mejoras
  - o Últimos cambios y mejoras del juego para la entrega final.
- Memoria y publicación
  - o Terminar memoria del TFG
  - o Publicación de la memoria en el reposito de UOC
- Preparación de ejecutable y Showcase
- Preparación de la defensa del TFG
- Preparación de la presentación oral guiada con transparencias
  - o Creación de un breve tráiler
- Realización de un informe con un ejercicio de autoevaluación

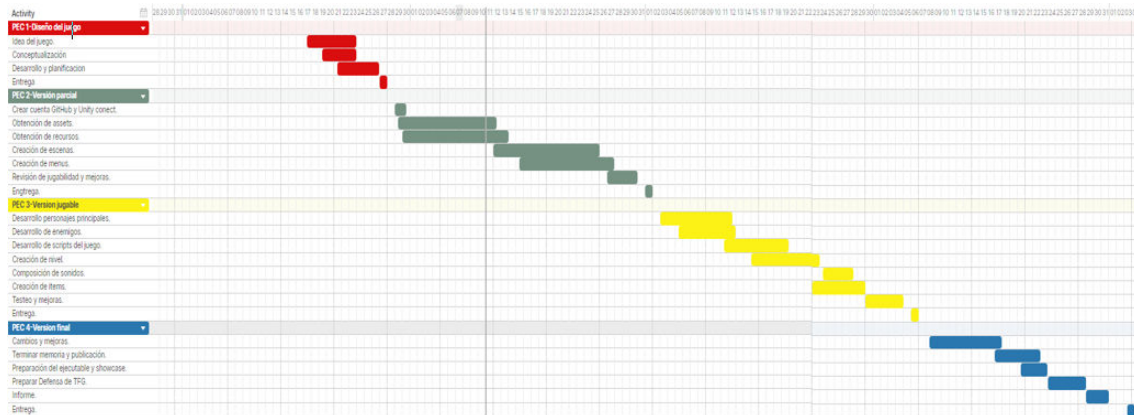


Ilustración 1. Planificación de entregas (Diagrama de Gantt)

### 1.5 Breve resumen de productos obtenidos

Se ha trabajado incansablemente para conseguir que el producto final sea un producto más que aceptable y un producto en el que el usuario final tenga una interacción importante dentro del videojuego. Los productos más importantes que se han obtenido son:

- Versión totalmente jugable.
- Memoria de final de grado.
- Informe de autoevaluación
- Diferentes videos explicativos
- Scripts del proyecto

### 1.6 Breve descripción de los otros capítulos de la memoria

- Capítulo 2: Estado del arte.
  - En este capítulo abarcaremos el género del que se trata el videojuego desarrollado, su evolución en la historia y los motores de desarrollo.
- Capítulo 3: Definición del juego.
  - En este capítulo hablaremos de la descripción y conceptualización, del videojuego, al arte conceptual de los actores y los objetivos del jugador.
- Capítulo 4: Diseño técnico.
  - Explicaremos el juego desde el enfoque técnico, definiremos el motor elegido, los requerimientos técnicos del entorno, las herramientas empleadas en la creación del mismo y los assets que hemos elegido para nuestro proyecto. También se explicarán la IA de los enemigos.
- Capítulo 5: Diseño de niveles.

- Pondremos un mapa del nivel o de los niveles que tenemos desarrollados y explicaremos el criterio de su diseño. Intentaremos que sea un guía de jugabilidad.
- Capítulo 6: Manual de usuario.
  - Se explicará los requerimientos técnicos necesarios para jugar a nuestro juego.
- Capítulo 7: Conclusiones.
  - Diferentes conclusiones generales que hemos sacado en el tiempo de desarrollo del proyecto.
- Capítulo 8: Glosario.
  - Palabras y definiciones de los términos utilizados en esta memoria.
- Capítulo 9: Bibliografía.
  - Enlaces o páginas de documentos que se han utilizado en la realización de esta memoria.
- Capítulo 10: Anexos o pruebas de usuarios.
  - Anexos a esta memoria o pruebas de usuarios reales.

## 2. Estado del arte

### 2.1 Género del Juego

Nuestro proyecto se basa en el género de plataformas 2D. Este tipo de género podemos encontrarlo sobre todo en las primeras videoconsolas que salieron al mercado y era muy popular en esa época. Se basa en diferentes mapas y una cámara frontal que seguirá al personaje principal dejándolo normalmente en el centro de la imagen, recorriendo los diferentes entornos y plataformas que nos vayamos encontrando en el camino. Este género está bastante consolidado en el mercado y es bastante común entre los desarrolladores.

Los juegos que podemos encontrar que son un ejemplo de nuestro proyecto son, Mario Bros, Alex Kidd y Spelunky, aunque también podemos pensar en algunos más modernos como Sonic o el famoso Rayman



Ilustración 2: Ejemplos de videojuegos

Todos estos ejemplos encajaban a la perfección con la idea que teníamos pensada, ya que este tipo de género se caracteriza por un recorrido horizontal, (aunque a veces también puede ser vertical) una interacción con el usuario basado en saltos y bonificaciones y una dificultad manifestada con enemigos que normalmente tendrás que evitar en el camino.

### 2.2 Revisión de la Tecnología

Hoy en día, el Engine más popular para el desarrollo de este tipo de videojuegos es Unity, ya que por su popularidad y por su reputación le hacen uno de los motores más importantes del mercado. También podemos hablar de Unreal o de GameMaker, que son otros dos motores muy potentes.

Aunque existen multitud de motores en el mercado, nosotros vamos a centrarnos en los tres que hemos mencionado anteriormente, mencionando sus características principales.

## Unity:



Ilustración 3: Unity

Es un motor comercial que fue lanzado en 2005, gratuito si no superas una facturación al mes que te indican ellos al regístrate. El motor es para desarrollo 2D como 3D y se emplea el lenguaje de programación C#.

Algunas de sus características principales es que hay una gran comunidad de desarrolladores, esto hace que sea más fácil la implementación de un proyecto ya que se puede encontrar multitud de ejemplos y de ayuda en multitud de foros. Otra de las características es que es fácil de usar y su aprendizaje es muy sencillo y te da facilidades al programar los scripts con la integración de Visual Studio.

Por último, podemos mencionar que es multiplataforma y que tiene muy poca exigencia en los recursos de ordenador, lo que facilita que se pueda instalar en cualquier ordenador personal.

## Unreal:



Este motor se lanzó mucho antes que Unity en 1998. También es gratuito, aunque algunas veces cobran por algunos conceptos que no quedan muy claros. Se puede desarrollar en 2D y en 3D y se emplea el lenguaje de programación C++.

Algunas de sus características principales es que tiene una edición en tiempo real para juegos de realidad virtual, lo que le hace muy interesante para el desarrollo de este tipo de juegos. Tiene acceso al código fuente, lo que nos permitirá adaptarlo a nuestras necesidades y modificarlo a nuestro gusto. También podemos mencionar que nos

facilita una herramienta para la programación de Scripts propia que se llama Visual Scripting.

Por último, podemos decir que este motor también es multiplataforma y que tiene un hiperrealismo de los gráficos lo que hará que elijamos este motor si queremos desarrollar un videojuego con unos gráficos y efectos superiores.

### **GameMaker Studio 2:**



GameMaker Studio 2™

Ilustración 5: GameMaker

GameMaker fue lanzado en 2011, aunque anteriormente se llamaba de otra manera. Tiene su versión gratuita, aunque si quieres hacer un videojuego más interesante debes contratar un plan profesional. Este motor está especializado en el desarrollo de juegos 2D.

Algunas de sus características principales son que tiene su propio lenguaje de script llamado GML. El editor que dispone es el más fácil de todos, ya que utiliza el arrastrar y soltar, cosa que facilita bastante la edición del videojuego. Otra característica principal que nos ofrece este motor es que podemos usar infinitos componentes prefabricados que dispone el mismo motor.

También es multiplataforma y como hemos mencionado anteriormente está especializado en la edición 2D, lo que lo hará una interesante elección si se quiere desarrollar un videojuego con esas dimensiones.

## 3. Definición del juego

### 3.1 Idea del juego

#### 3.1.1 Descripción del juego

“Dog on the run” es un juego de animación y aventuras en 2D. El personaje principal trata de encontrar a su perro perdido. En su camino encontrará difíciles obstáculos como plataformas o trampas y tendrá que avanzar entre los distintos escenarios intrigantes y terroríficos mientras sorteando los numerosos enemigos que encontrará a su paso.

Además, el protagonista tendrá que intentar conseguir los premios que vayan apareciendo durante su aventura mediante saltos, movimientos inteligentes y astucias creativas. Así y con mucha habilidad, el jugador podrá encontrarse con su mascota al final de juego y terminar feliz su aventura.

#### 3.1.2 Interacción jugador – juego

En cada partida, el jugador contará con un menú donde elegir, a través del teclado, el personaje con el que quiere jugar y/o comenzar partida. Al comienzo de cada partida, el jugador aparece en un escenario en el que tendrá que desplazarse horizontalmente mediante saltos y habilidades para superar los retos que se le proponen, ya sean escapar de enemigos o conseguir premios.

El juego pretende además de rescatar a la mascota y conseguir así el objetivo principal del juego, superar en cada partida la puntuación de la anterior, procurando que el jugador intente una y otra vez conseguir más monedas.

Los controles del teclado serán los siguientes:

- Flecha derecha o tecla D: Desplazamiento hacia la derecha.
- Flecha izquierda o tecla A: Desplazamiento hacia la izquierda.
- Tecla space: Salto o Doble salto.
- Tecla Esc: Menú pause.
- Tecla M: Mute del juego.

#### 3.2.3 Plataforma de destino

La plataforma de destino de este videojuego es la de PC. Esta decisión viene dada por la historia que tiene este tipo de videojuegos y porque es una plataforma altamente extendida dentro del mundo de los

videojuegos. Aún eligiendo esta plataforma no descartamos que en el futuro podemos integrar este videojuego a dispositivos móviles.

## 3.2 Conceptualización

### 3.2.1 Historia y ambientación

El juego se ambienta en el año 2035, en Indiana (Estados Unidos). Un apocalipsis zombie se ha apoderado del mundo. La civilización está perdida y sólo quedan los protagonistas y su perro. Los zombies y los diferentes monstruos están por toda la ciudad y matan a cualquier superviviente sólo con tocarlo. Los sonidos que emiten los zombies ahuyentan al perro del protagonista que corre hasta esconderse. Está asustado y no puede moverse por lo que los protagonistas tendrán que ir en su rescate. Para reunirse con su fiel amigo, tendrán que recoger premios y esquivar numerosos enemigos y diferentes trampas y plataformas que encontraran en su angustioso viaje.

### 3.2.2 Objetivos planteados al jugador

Superar los distintos niveles del juego y rescatar a la mascota de los personajes principales. El jugador tendrá que intentar no caer en las manos de los monstruos, no caer en las diferentes trampas que se vayan encantando en el camino y sortear las diversas plataformas para poder llegar sano y salvo y con el máximo nivel de vida y de monedas para así rescatar a Duna.

### 3.2.3 Definición de los personajes y elementos

- Personajes principales

Amber: Personaje femenino de la aventura. Es una chica enamorada de los animales y de la naturaleza. Una chica deportista y divertida.



Ilustración 6: Personaje femenino



Bobby: personaje masculino de la aventura. Es un chico apasionado de la tecnología y de la naturaleza.



Ilustración 7: Personaje masculino

Duna: es el perro que hay que rescatar en la aventura. Una fiel amiga, un poco revoltosa pero que sólo quiere estar con su compañero/a.



Ilustración 8: Mascota perdida

- Personajes secundarios (enemigos)

Zombies: Humanos que se han convertido en muertos vivientes porque han sido tocados por otros zombies.



Ilustración 9: Zombies

Fantasmas: Almas perdidas en un limbo entre la vida y la muerte.



Ilustración 10: Animación de fantasma

Murciélagos: Seres voladores que se convierten en vampiros y si te tocan te harán bastante daño.



Ilustración 11: Animación de murciélago

- Elementos del juego.

Monedas: conseguirás puntos para aumentar la puntuación. Hay de dos tipos, plateadas y doradas, según la dificultad de conseguir las.

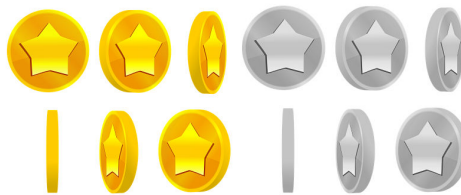


Ilustración 12: Monedas

Trampas: Trampas que se encontrarán en el camino en el que si el personaje cae en ellas su vida se verá afectada.



Ilustración 13: Spikes

Fuego: Trampas de fuego, al caer en el fuego el personaje experimentará dolor y perderá vida.



Ilustración 14: Fuego

## 4. Diseño técnico

### 4.1 Entorno de desarrollo

Para el desarrollo de este proyecto se decidió que se usara el motor Unity por la gran cantidad de información que existe sobre este motor. Al ser primerizos en el desarrollo de videojuegos nos pareció lo más sensato que cuanto más información contáramos para el desarrollo sería lo mejor para el porvenir del proyecto.

Unity tiene infinidad de información en la red, ya que dispone de foros propios y foros externos donde cualquier duda que pueda surgir, será inmediatamente solucionada. También se dispone de gran cantidad de tutoriales tanto en inglés como en español. La versión que hemos utilizado ha sido 2019.4.10f1.

Lo que nos ha sorprendido de este motor es la facilidad de implementación que tiene y el gran potencial de desarrollo del que dispone. Se puede desarrollar un videojuego en unas pocas horas y a la vez si se pretende se puede hacer un videojuego de una calidad espectacular y lo mejor de todo, es que se puede hacer para cualquier plataforma. De hecho, si tenías pensado realizar un proyecto en una plataforma, se puede cambiar de plataforma en el mismo proyecto si ningún problema.

### 4.2 Requerimientos técnicos del entorno

#### Mínimos:

- Sistemas operativos: Windows 7, MacOS Sierra, Ubuntu 16.04
- Procesador: Arquitecturas X64
- Memoria: 2Gb de RAM
- Gráficos: GPU compatibles con DX10 o superiores y OpenGL 3.2
- Almacenamiento: Mínimo 2Gb

#### Recomendados:

- Sistemas operativos: Windows 7, MacOS Sierra, Ubuntu 16.04
- Procesador: Intel Core I3
- Memoria: 4Gb de RAM
- Gráficos: Intel HD 5100 y OpenGL 3.2
- Almacenamiento: Mínimo 2Gb

#### 4.3 Herramientas utilizadas en el desarrollo

##### Visual Studio:



Visual Studio es un potente entorno de programación integrado de Microsoft. Soporta infinidad de lenguajes y se integra perfectamente con Unity, ya que dispone de herramientas especializadas para el desarrollo de un juego en Unity.

##### Paint 3D:



Paint 3D es una herramienta que viene preinstalada en dispositivos con Windows 10 instalado y es una herramienta que nos ha sido muy útil. La hemos utilizado para la creación de imágenes, cambio de tamaño de los assets y modificación de algunos elementos visuales del juego.

##### Audacity:



Audacity es una herramienta gratuita para la edición de sonidos. Hemos utilizado esta herramienta para modificar y realizar los detalles sonoros que se integran den el videojuego.

##### Microsoft Office:



Microsoft office es una herramienta muy popular dentro del mercado informático y es una herramienta muy valiosa para multitud de tareas que se hacen diariamente. Nosotros la hemos usado casi a diario, ya que los diferentes informes, la realización de la planificación y para el desarrollo de los bocetos y notas que hemos realizado para el desarrollo del proyecto.

GitHub:



Ilustración 19: Github

GitHub es un repositorio de proyectos que nosotros hemos usado para las diferentes entregas y versiones que hemos estado desarrollado. Es una herramienta fundamental para el desarrollo de software ya que siempre te permite tener la versión más reciente de tu proyecto.

#### 4.4 Inventario y recursos del juego

Los assets, sonidos, imágenes y recursos que se han utilizado en este proyecto son todos gratuitos y los hemos conseguido en páginas como Unity asset store, itch.io, sfx.freeaudiolibary y gameart2d entre otros.

Todos los recursos son reaprovechados con pequeñas modificaciones en la gran mayoría de los recursos. En los personajes se ha retocado el tamaño y la definición, en las monedas hemos retocado color y la orientación y en la música hemos recortado el tiempo o alargado según nos convenía en cada momento.

Sprites:

Los sprites los hemos descargado de la página GameArt2D y de Unity Asset Store cuyos personajes se podía reaprovechar en todo su sentido. Lo que hemos realizado en esta ocasión son la modificación de los tamaños de los assets y detallar los boxcollider de cada imagen, ya que, en el sprite del personaje femenino, las medidas venían diferentes al del masculino.

También hemos tenido que crear las diferentes animaciones de los diferentes sprites y cambiar algunos movimientos que nos convencían en su totalidad.

En definitiva, hemos realizado pequeñas modificaciones en los sprites para tenerlos a nuestro gusto, por lo que ha supuesto un tiempo importante en la realización del proyecto, aunque hay que reconocer que el desarrollo de estas operaciones empezando desde 0 hubiera sido un trabajo mucho mayor y mucho más complicado.



Ilustración 20: Diferentes assets

### Sonidos:

Lo más complicado de los sonidos lo hemos tenido en la búsqueda de la misma. Aunque nos hemos descargado varios modelos de efectos sonoros y de música de ambiente, nunca nos convencía del todo y hemos tenido que modificar la gran parte de los sonidos. También como nos pasó con los sprites son modificaciones que hemos realizado a pistas que ya estaban creadas.



Ilustración 21: Sonidos

### Fuentes:

Las fuentes que tenemos implementada en todo el videojuego es una fuente que se llama LifeisGoofy y la verdad es que estamos muy contentos de como ha quedado la implementación de esta fuente ya que buscábamos algo juvenil y con un toque a videojuego antiguo.

**Esta es la fuente elegida para el proyecto**

Ilustración 22: Fuente

## 4.5 Arquitectura y diseño del videojuego

### Diagrama del flujo:

Con el diagrama de flujo que expondremos a continuación podemos observar el flujo de una partida desde que se inicia el videojuego, mostrando todos los procesos importantes que está integrado en el.

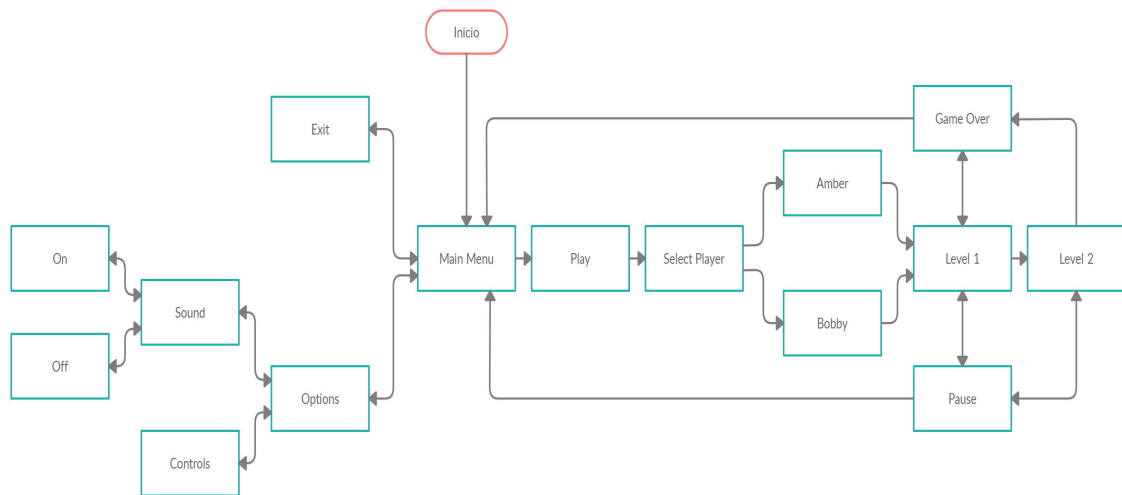


Ilustración 23: Diagrama de flujo

En resumen, lo que nos está indicando este diagrama es que nosotros empezamos desde un inicio, ese inicio es desde que ejecutamos el juego en nuestro ordenador. Después nos saldrá un menú principal donde podemos elegir entre “Jugar”, “Opciones” y “Salir”, si elegimos salir, saldremos de la aplicación, si elegimos opciones, nos llevará a otros dos supuestos más, que son “Controls”, donde se encuentran los controles de jugabilidad del juego. Si elegimos “Sound” podremos elegir entre activar o desactivar todos los sonidos del juego.

Volviendo al principio, si elegimos “Play”, nos llevará a una pantalla donde podremos elegir entre los dos personajes principales, una vez seleccionemos alguno inicializaremos el juego en cuestión, donde, si pulsamos la tecla Esc o el icono de Pause que viene en la pantalla accederemos al menú pause, donde podremos elegir entre salir al menú principal o continuar la partida. Si continuamos la partida solo nos quedaría o acceder al nivel 2, donde tendrá los mismos menús que el nivel 1 o que por desgracia nos quiten vida y nuestro personaje muera, es entonces cuando nos saldrá un menú de Game Over donde podremos elegir entre restaurar la partida, que siempre será en el nivel 1 y con la vida restaurada o tendremos también la opción de salir al menú principal.

### Lógica del personaje principal:

Para poder realizar los diversos cambios entre las diferentes animaciones que hemos implementado al personaje hace falta una lógica programable que indique en todo momento qué animación debe elegir el personaje.

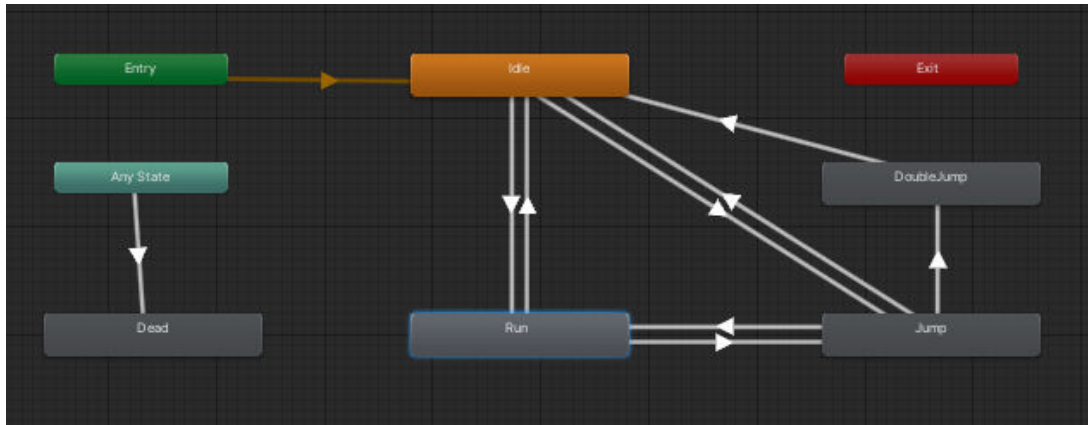


Ilustración 24: Lógica de movimiento

### 4.6 Funcionamiento de la IA

Tenemos dos casos de movimiento de los enemigos. El primero de ellos es el que hace los zombies, cuya inteligencia artificial es ir de un lado a otro sin tocar con ningún elemento. Es una inteligencia básica.

El otro caso que tenemos implementado de IA es la de los murciélagos y fantasmas. Estos enemigos tienen tres puntos de movimiento, lo que hace que la implementación en este caso sea algo diferente a lo anterior y algo más complicado, sin salirnos como hemos indicado de una IA básica.

### 4.7 Head-Up Display (HUD)

Hemos un Head Up Display muy visual y lo más descriptivo posible. Lo que hemos premiado es la facilidad y la información que se ofrece al usuario que esté jugando. Esta información es las monedas conseguidas hasta el momento, ya sean doradas o plateadas y las vidas que nos quedan por jugar.

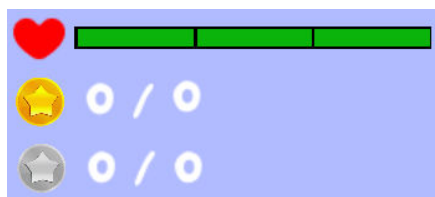


Ilustración 25: HUD



## 5. Diseño de niveles

### 5.1 Introducción

Actualmente disponemos de dos niveles, uno de exterior y otro de interior, como en una cueva. Las diferentes dificultades que hemos encontrado a lo largo del curso, la solución y la implementación de las mismas nos han obligado a implementar solo 2 niveles, aunque en un futuro, no muy lejano, tenemos la idea de incrementar tanto el nivel de dificultad como la implementación de nuevos niveles y de nuevas aventuras para nuestros protagonistas.

### 5.2 Nivel 1 (Exterior)

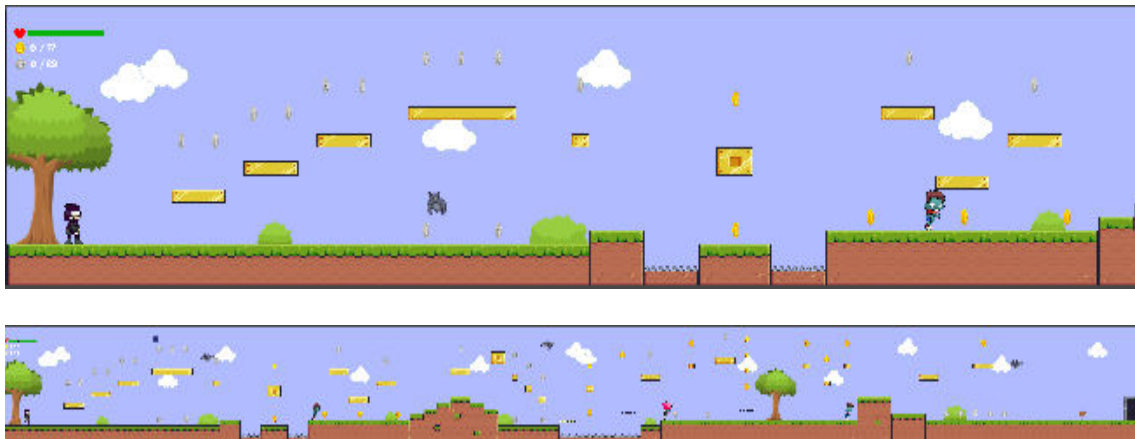


Ilustración 26: Nivel 1

Como podemos observar en las imágenes el personaje principal comienza la aventura en el inicio de la misma y delante de él hay unas plataformas y monedas que tendrá que conseguir. Existen dos tipos de monedas, plateadas y doradas. Cada una de ellas tiene un contador diferente e individual. Si avanzamos sobre el nivel nos encontramos el primer enemigo que es un murciélago y las primeras trampas que son unos pinchos insertados en el suelo. Si seguimos con la aventura nos encontraremos con una fila de pinchos que tendremos que sortear a través de unas plataformas móviles. Al finalizar este obstáculo nos encontraremos con dos zombies y una plataforma móvil para poder recolectar las monedas que tenemos en la parte superior. Después de evitar que los zombies nos toquen nos encontraremos con el ultimo enemigo, que será un murciélago, al sortear a este nos encontraremos una puerta que nos llevará directamente al nivel 2 donde nos esperaran nuevas aventuras y nuevos enemigos.

### 5.3 Nivel 2 (Interior)

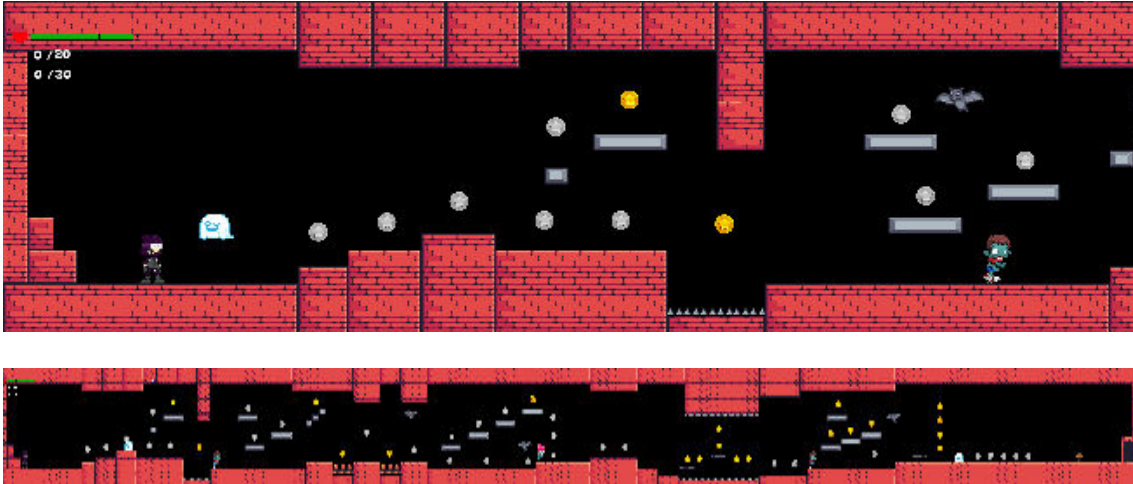


Ilustración 27: Nivel 2

En este nivel dos podemos observar que la dinámica será parecida al anterior nivel, nos encontrásemos enemigos que deberemos sortear y diferentes trampas en las que no podremos caer. Además, existirán diferentes plataformas para intentar recolectar las diferentes monedas y evitar a los enemigos. La recolección de las monedas sigue el mismo patrón que en el anterior nivel, existen monedas plateadas y monedas doradas y serán contabilizadas de la misma manera, pero es un conteo distinto. Los enemigos y las trampas en este nivel sufren un pequeño cambio y se añade a la colección fuego y fantasmas además de los que ya teníamos en el nivel anterior.

En este nivel empezaremos con una fantasma enfrente de nosotros que nos dificultará el paso y la recolección de las monedas. Poco después, encontraremos con una difícil maniobra que tiene como consecuencia unos pinchos en subsuelo. Si logramos esquivar estas pruebas nos encontraremos con dos enemigos unidos, un zombie y un murciélago, al pasar de ellos, llegará la hora del fuego, donde tendremos que sortear dos difíciles pruebas. Si todo esto nos parecía poco nos volvemos a encontrar con otros dos enemigos juntos, que vuelve a ser un murciélago y un zombie. Al pasar esta zona nos encontraremos con la temible plataforma móvil cuya finalidad tiene que nos caigamos a los afilados pinchos que tienen abajo, con nuestra habilidad y nuestra experiencia nos enfrentaremos a los últimos enemigos. Después de esto, nos encontraremos a nuestra asustada Duna y finalizaremos el juego.

## 6. Manual de usuario

### 6.1 Requisitos del juego

Para ejecutar “Dog On The Run” no hace falta que tengamos un súper equipo, aunque debemos cumplir unos mínimos deseables para que no tengamos problemas de rendimiento.

#### Mínimos:

- Sistemas operativos: Windows 7, MacOs Sierra, Ubuntu 16.04
- Procesador: Arquitecturas X64
- Memoria: 2Gb de RAM
- Gráficos: GPU compatibles con DX10 o superiores y OpenGL 3.2
- Almacenamiento: Mínimo 2Gb
- Resolución de pantalla: 800 x 600 pixeles.

### 6.2 Controles

Los controles del teclado serán los siguientes:

- Flecha derecha o tecla D: Desplazamiento hacia la derecha.
- Flecha izquierda o tecla A: Desplazamiento hacia la izquierda.
- Tecla Space: Salto o Doble salto.
- Tecla Esc: Menú pause.
- Tecla M: Mute del juego.

### 6.3 Instrucciones de ejecución

Para jugar a nuestro juego es imprescindible descargarse el archivo comprimido que se facilita con este manual, descomprimiremos la carpeta en cualquier lugar de nuestro computador y ejecutaremos en archivo con el nombre DOG ON THE RUN.exe.

Una vez ejecutado el archivo solo te quedará disfrutar de la experiencia. A continuación, te indicamos algunos menús que pueden ser importantes para el buen desarrollo de la experiencia.

## 6.4 Menús del juego

### 6.4.1 Menú Principal

Aquí podemos elegir entre iniciar una nueva partida, observar las opciones del juego o salir del juego.



Ilustración 28: Menú Principal

### 6.4.2 Menú Options

Se podrá elegir entre observar los controles de jugabilidad o activar y desactivar sonido.

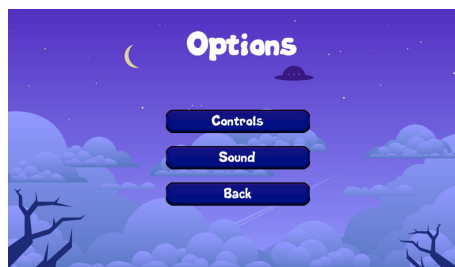


Ilustración 29: Menú Options

### 6.4.3 Menú Sound

Al seleccionar este menú, podemos desactivar o activar todos los sonidos y la música que componen el juego.

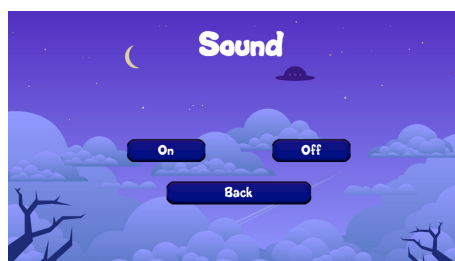


Ilustración 30: Menú Sound

#### 6.4.4 Menú Controls

En este menú podemos encontrar los diferentes controles que son necesarios para el desarrollo del juego.

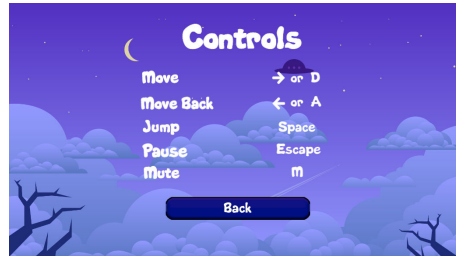


Ilustración 31: Menú Controls

#### 6.4.5 Menú Select

Este menú lo utilizaremos para seleccionar a los diferentes personajes, Bobby o Amber.

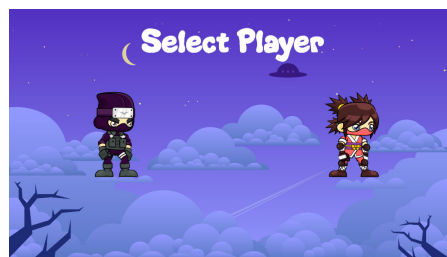


Ilustración 32: Menú Select

#### 6.4.6 Menú Pause

Por último, nos encontraremos con el menú pause, este menú se encuentra dentro del juego y es accesible con el botón pause que se encuentra en la parte superior derecha o a través de la tecla Esc.



Ilustración 33: Menú Pause

## 7. Conclusiones

Este trabajo me ha servido para darme cuenta y vivir de primera mano de lo que es realmente la gestión de un proyecto de desarrollo de un videojuego. Cuando en la primera PEC nos pedían que hiciéramos una planificación del proyecto nunca me imaginé que fuera de estas características.

Me he percatado que el diseño y desarrollo de un videojuego no solo contempla una especialidad, sino que son muchas especialidades juntas trabajando a unísono por un objetivo común. Me ha sorprendido las diferentes áreas que intervienen en la creación de un videojuego y he aprendido mucho de ello.

A pesar de lo expresado anteriormente creo que hemos logrado en gran medida y, a pesar de la multitud de dificultades que hemos encontrado en el camino, realizar la planificación que nos impusimos al principio del curso y tener un nivel de juego más que aceptable. Somos conscientes que este proyecto ha sido un diseño de un videojuego a pequeña escala, pero tenemos que decir que es más que suficiente para empezar a desarrollar y hacerse un hueco en este mercado.

En el anterior párrafo me he referido a que estábamos contentos con la finalización del proyecto, pero también tenemos que ser críticos con nosotros mismos y saber que se podía haber mejorado en muchísimos aspectos.

Algunos de los aspectos que nos hubieran gustado implementar, es guardar el Score o que nuestro personaje hubiera atacado a sus enemigos. También podríamos haber implementado la escalada del personaje o el deslizamiento por el suelo, pero como hemos indicado en capítulos anteriores, este proyecto tiene futuro y en un futuro, no muy lejano, implementaremos muchas mejoras.

Otra de las cosas que se nos quedan en el tintero, es implementar el juego a un multiplataforma. Nos hubiera gustado muchísimo implementar este juego en dispositivos móviles y probar cómo funciona en estos dispositivos.

También me hubiera gustado unas pruebas de videojuego más extendidas, ya que solo lo han probado amigos y familiares. He echado de menos una crítica constructiva técnica más extendida en muchas más personas y puntos de vista diferentes.

Para concluir, debo decir que estoy muy contento de cómo se ha desarrollado la asignatura. No me imaginaba nada igual. Creo que salgo de este proyecto siendo mejor programador, mejor diseñador y mucho más metódico. Estoy muy motivado con lo que acontece en un futuro y creo que este trabajo me ha valido para madurar profesionalmente y experimentar vivencias que nunca había tenido.

## 8. Glosario

**2D:** Es representativo de los videojuegos y significa que utilizan gráficos de dos dimensiones con lo que no tendrá profundidad.

**3D:** Es representativo de los videojuegos y significa que utilizan gráficos de tres dimensiones con lo que tendremos profundidad.

**PLATAFORMAS:** Genero de lo videojuegos muy recurrido en los inicios de los videojuegos.

**MULTIPLATAFORMA:** Se pueden usar diferentes dispositivos para el desarrollo del mismo.

**REQUISITO:** Cualidad que se requiere para funcionar.

**RESOLUCIÓN:** Numero de pixeles que contiene una imagen o pantalla.

**SCRIPTS:** Documento que contiene instrucciones escritas den códigos de programación.

**PROGRAMACIÓN:** Arte del proceso por el cual se indica a la computadora que tiene que realizar en cada momento.

**ASSET:** Recurso de un videojuego.

**SPRITE:** Imagen de mapa de bits, usado para representaciones graficas en 2D

**IA:** Inteligencia artificial.

**ENGINE:** Motor en el que se desarrolla un videojuego.

**HUD:** Información que se nos presenta sobre el estado de nuestra partida o nuestro personaje.

## 9. Bibliografía

### Unity Asset Store:

- Pixel Adventure 1:  
<https://assetstore.unity.com/packages/2d/characters/pixel-adventure-1-155360>
- Pixel Adventure 2:  
<https://assetstore.unity.com/packages/2d/characters/pixel-adventure-2-155418>
- Blue Cartoon GUI Skin:  
<https://assetstore.unity.com/packages/2d/gui/blue-cartoon-gui-skin-19535>
- Sound retro pack:  
<https://assetstore.unity.com/packages/audio/sound-fx/sound-fx-retro-pack-121743>
- Sound voices:  
<https://assetstore.unity.com/packages/audio/sound-fx/voices/attack-jump-hit-damage-human-sounds-32785>
- 8 bits music:  
<https://assetstore.unity.com/packages/audio/music/electronic/8-bit-music-free-136967N>

### Game Art 2D:

- Free Sprites:  
<https://www.gameart2d.com/freebies.html>

### Png Wings:

- Sprites monedas:  
<https://www.pngwing.com/es/free-png-zbyul>
- Sprites corazon:  
<https://www.pngegg.com/es/png-nndzj>

### Free Png:

- Imágenes nubes:  
<https://www.freepng.es/png-ycssb3/>
- Árboles y decorado:  
<https://www.freepng.es/png-aesubi/>



## 10. Pruebas de usuario y enlaces de interés

### 10.1 Pruebas de usuario

Las pruebas que hemos recibido por parte de los usuarios en general han sido buenas. La finalidad del juego y la jugabilidad son los factores que más se han valorado.

Me hubiera gustado crítica más profesional ya que las pruebas que hemos realizado han sido entre familiares y amigos y las conclusiones no han sido técnicas, por lo que me hubiera gustado una visión en la que el conjunto del juego fuera el objetivo a evaluar.

Algunas de las críticas más destacadas nos hablan de ampliación de los niveles, que la dificultad del juego es alta y mejoras del personaje como matar a los enemigos.

También hemos subido una versión a itch.io, pero no hemos tenido valoraciones destacables en este aspecto.

### 10.2 Enlaces de interés

- Github:  
<https://github.com/CarlosMurcia/TFG-Videojuegos>
- Unity Connect:  
<https://connect.unity.com/u/carlosmurcia>
- Itch.io  
<https://cmurcia.itch.io/dog-on-the-run>
- Presentación proceso de creación:  
<https://www.youtube.com/watch?v=LMQ5A7DJFMs&t=22s>
- Tráiler del videojuego:  
<https://www.youtube.com/watch?v=vIpeBRw5Q1Q>