###### 

Título del Trabajo  
Fin de Grado

Grado en Ingeniería Multimedia



Trabajo Fin de Grado

Autor:

Nombre y Apellidos

Tutor/es:

Nombre y Apellidos

Enero 2014



Justificación y Objetivos

## Objetivos

### Objetivos Generales

#### Objetivo General: Conocer los principales Frameworks de creación de videojuegos 2D para no programadores.

La principal motivación de este proyecto es analizar y conocer los Frameworks y SDK´s que permiten la creación de videojuegos sin tener conocimientos a priori de lenguaje hombre-máquina.

Dada la gran popularidad que han experimentado los videojuegos en los últimos 20 años en el mercado existen una gran cantidad de programas que permiten (o al menos lo intentan) realizar esta función, se pretende encontrar los más utilizados o los que mejor consiguen realizar esta función prestando atención a los resultados profesionales que se pueden llegar a conseguir con ellos.

Conocer accesoriamente los juegos comerciales más populares que se han desarrollado con estas herramientas así como los modelos de negocios de algunas de ellas.

#### Objetivo General: Analizar en profundidad los Frameworks y seleccionar un candidato adecuado para el estudio.

Una vez conocidos los SDK´s y Frameworks de este tipo se va a seleccionar uno de ellos en función de su importancia en el mercado, de los resultados y se va analizar todos sus aspectos y funcionalidades.

Para ellos se van a comparar los distintos programas mediante unos criterios que se determinaran para que el seleccionado se ajústate a los objetivos del presente proyecto.

### Objetivos Específicos.

#### Comparación de juegos en programación clásica y con un framework.

Para tener una visión global de lo que este tipo de programas que la presente memoria intenta analizar se necesita comprar los resultados, consumo de recursos, tiempos, curvas de aprendizaje o dificultad de uso entre otros aspectos en proyectos realizados con este tipo de software y la programación más clásica de videojuegos usando motores de programación pura como podría ser SFML o OGRE 3D.

Para conseguir esta comparación de ambas formas de crear juegos se va a realizar el mismo juego tanto en SFML como en él programa seleccionado y hacer una comparación de todos los elementos comentados anteriormente

# Introducción.

# Marco teórico.

En este apartado del proyecto se pretende describir los conceptos fundamentales con los que se va trabajar, así que será el punto de partida para todos los desarrollos teóricos y prácticos posteriores. También sentará las bases para lo que se realizará en los siguientes puntos.

El marco teórico se divide en tres apartados:

1. En la primera parte se van a presentar tres estudios relacionados con el proyecto así los conceptos que se van a manejar durante el desarrollo del trabajo.
2. La parte central del marco teórico se dedicará a definir unos criterios de selección en base a los cuales se van a puntuar los **frameworks** que se analizarán en el cuerpo del trabajo.
3. Con los criterios ya definidos, se va realizar una primera selección de **frameworks**. En esta preselección se van a reducir todo lo posible el número de programas a analizar en el cuerpo del trabajo y excluir aquellos que tengan incompatibilidades con la finalidad del proyecto o impidan la consecución de los objetivos.

## Estudios previos

### Dev Master.

Dev Master es una base de conocimiento web de motores de videojuegos, herramientas, librerías accesorias, compañías y profesionales dedicados al sector de la programación de videojuegos. En ella se encuentran más de 350 motores.

Cada sección dedicada a un motor gráfico cuenta con todos sus datos (sitio web, propietarios, licencia y fecha de lanzamiento entre otros),sus datos técnicos (por ejemplo su lenguaje de programación, motores de físicas o shaders que usa), los libros escritos sobre el mismo relacionados y un apartado para que los usuarios y profesionales puedan dejar sus comentarios y análisis sobre el programa.

La principal diferencia con el estudio del presente proyecto radica en los siguientes aspectos fundamentales:

* Dev Master trata únicamente de motores gráficos y no también de frameworks.
* No compara los motores entre si como se pretende hacer en este proyecto, sino que se limita a listar todas sus características y opiniones de los usuarios.

|  |  |
| --- | --- |
|  |  |
| Tipo | Proyecto Web |
| Año | 2017 |
| Autor/es | Colaborativo |
| Link | http://devmaster.net/ |

Tabla 1: Dev Master

### Comparison and evaluation of 3D mobile game engines.

Se trata de una tesis realizada por Akekarat Pattrasitidecha para la universidad de Gotemburgo y que trata de encontrar el mejor programa para desarrollar videojuegos en tres dimensiones para dispositivos móviles.

Al buscar y comprar entre motores 3D centra la mayoría del trabajo en los aspectos técnicos de los motores de juego como las físicas 3D, los efectos de partículas o el video en 3D dejando los aspectos como la usabilidad y la facilidad de uso en un segundo plano al contrario que en el este proyecto. Además se limita únicamente a motores de juego para dispositivos móviles y no abarca todos aquellos para escritorio y consolas como si se valora en este estudio.

|  |  |
| --- | --- |
|  |  |
| Tipo | Tesis |
| Año | 2014 |
| Autor/es | Akekarat Pattrasitidecha |
| Link | http://publications.lib.chalmers.se/records/fulltext/193979/193979.pdf |

Tabla 2: Comparison and evaluation of 3D mobile game engines

### Comparación de motores de videojuegos para la creación de juegos serios.

Proyecto de fin de grado presentado por dos estudiantes de la Universidad Tecnológica de Pereria (Colombia), que trata de buscar y definir unos criterios tanto teóricos como tecnológicos que tiene que tener un de videojuego para poder desarrollar un videojuego profesional.

También tiene un apartado dedicado a pequeño análisis y comparación de motores de juego aunque el objeto de estudio solo son tres motores: JMonkey, Unreal Engine y Unity 5.

Por limitarse a solo el estudio de tres motores y estar centrado en definir los requisitos de un motor para crear un videojuego profesional, cuando en el caso del presente proyecto se desea abarcar proyectos de todo tipo, se diferencia bastante del este estudio.

|  |  |
| --- | --- |
|  |  |
| Tipo | Tesis |
| Año | 2015 |
| Autor/es | Cristian Steven Chavarro Rico  Mauricio Valencia Arbelaez |
| Link | http://repositorio.utp.edu.co/dspace/bitstream/handle/11059/6016/00669CH512.pdf?sequence=1 |

Tabla 3: Comparación de motores de videojuegos para la creación de juegos serios

## Conceptos previos.

### Framework.

Un **framework** en esencia es un patrón o esqueleto para el desarrollo de una aplicación, juego o contenido multimedia. Una definición más académica o formal podría ser que un **framewok** es un conjunto estandarizado de conceptos, prácticas y criterios para desarrollar y crear aplicaciones.

Los **frameworks** suele incluir:

* Soporte de programas.
* Librerías.
* Lenguaje de scripting.
* Software accesorio para desarrollar partes especificas del programa.
* Un lenguaje de programación o interpretado.

Su uso implica un coste de aprendizaje aunque a largo plazo facilita el desarrollo, mantenimiento de la aplicación y permiten evitar trabajar en los detalles de bajo nivel permitiendo concretar esfuerzo en las partes esenciales de programa.

Desde el punto de vista del desarrollo de software, un **framework** es una estructura de soporte definida, en la cual otro proyecto de software puede ser organizado y desarrollado.

### SDK (Kit de desarrollo).

Un **SDK o Kit de desarrollo** es un conjunto de herramientas y programas de desarrollo de software que permite al programador crear aplicaciones para un determinado paquete y estructura de software, plataforma de hardware, sistema operativo de computadora o máquina.

Un **SDK** no es más que una **API[[1]](#footnote-1)** (interfaz de aplicaciones) que permite el uso de uno o varios (aunque esta opción es menos habitual ya que suele trabajar en exclusividad con un lenguaje) lenguajes de programación.

Los **SDK** suelen llevar incluidas herramientas que ayudan al desarrollador a crear sus aplicaciones. Entre ellas destacan las siguientes:

* Debug.
* Soporte para le detección de errores de programación.
* Códigos de ejemplo y documentación.
* Entorno de programación **IDE[[2]](#footnote-2).**

### Game Engine (Motor de videojuegos).

El **Game Engine o Motor de videojuegos** proporciona un entorno de desarrollo integrado para la creación de juegos, no es más que un framework diseñado para la creación y desarrollo de juegos.

Un motor de juego suele incluir diferentes componentes que se corresponden con los elementos de un videojuego. Los elementos que suele incluir entre otros son los siguientes:

* **Programa del juego principal.**
* **Motor de render.**
* **Motor de audio.**
* **Motor de físicas.**
* **Motor de IA.**
* **Motor de Red.**

Antes de los Game Engine se escribían como entidades singulares y no reutilizables. El término motor de juego apareció a mediados de los 90 ligado al crecimiento de los juegos 3D especialmente shooters en primera persona, ya que permitía en juegos muy complejos técnicamente no comenzar desde cero en cada nuevo desarrollo.



Figura 1 : Wof3d Pimer motor de juegos 3D

## 3.2 Criterios de selección

Uno de los objetivos fundamentales del presente proyecto es analizar y buscar el mejor framework de creación de videojuegos para todo tipo de usuarios. Para seleccionar el programa se deben definir previamente unos criterios que permitan que esta elección sea lo más correcta posible ya que de ella depende en gran medida el éxito del proyecto.

Posteriormente a la determinación de estos criterios se le otorgará a cada uno de ellos una puntuación máxima y mínima. La suma de estas puntuaciones determinará cuál el mejor framework de carácter general.

Se va usar una escala de 100 para definir la puntuación ya que da más facilidades a la hora de visualizar las mismas y comparar los distintos programas entre si. No todos los criterios tienen la misma puntuación ya que no todos tienen la misma importancia, ni los mismos elementos.

En la siguiente tabla se encuentran definidas las distintas puntuaciones que pueden tener los elementos de cada criterio y su valoración en cuanto a su importancia y denominación.

|  |  |  |
| --- | --- | --- |
| Puntuación | Importancia | Explicación |
| 0-1 | Mínima | Elemento sin mucha importancia pero que hay tener en cuenta y valorarlo. |
| 0-2 | Poca | Apartado con cierta importancia pero que se desea que no influya demasiado en el resultado final |
| 0-4 | Normal | Los criterios en este apartado tienen una importancia standard, puntuación por defecto |
| 0-6 | Alta | Elementos bastante importantes y que deben contar mucho a la hora de seleccionar un programa |
| 0-8 | Clave | Elementos clave y fundamentales. |
| >10 | Especial | Se detallará en cada caso si hay alguna puntuación de este tipo. |

Tabla 4: Clasificación de puntuaciones

### Criterio 1: Requisitos e instalación.

El primer criterio que hay tener en cuenta a la hora de analizar los diferentes motores es el requisito de funcionamiento e instalación.

En cuanto a los requisitos de instalación, dado el carácter generalista del presente proyecto, se valorará con mayor puntuación a aquellos programas En cuanto a los requisitos de instalación, dado el carácter generalista del presente proyecto, se valorará con mayor puntuación a aquellos programas que necesiten menos recursos del sistema para funcionar y que además funcionen en más sistemas operativos.

Es importante que el proceso de instalación sea tenido en cuenta, para comprobar la sencillez del mismo, la posible necesidad de instalación de otros programas o librerías accesorias y una buena guía al usuario durante todo el proceso. En este apartado será también será considera el tiempo que dura el proceso.

A continuación se muestran los apartados que compone este criterio y su importancia.

|  |  |  |
| --- | --- | --- |
| CRITERIO | IMPORTANCIA | PUNTUACIÓN MAX-MIN |
| Requisitos de instalación. | Normal | 0-4 |
| Instalación. | Poca | 0-2 |

Tabla 5: Criterio: Instalación y requisitos

#### Medición de la puntuación: Instalación.

El apartado instalación se medirá con la realización de encuestas en las que se solicitará a cada participante que mida el tiempo que tarda en todo el proceso de descarga e instalación del programa en su equipo y posteriormente se le hará las siguientes preguntas:

1. **Tiempo total de instalación.**
2. **Valoración de la complejidad del proceso:** de instalación en una escala de 10 donde 10 es sencillo y cero es complicado.
3. **Valoración del proceso de instalación:** Donde se va tratar de medir si los pasos en la instalación están bien explicados, donde 1 es que es poco o nada explicados y 10 que el programa guía al usuario de forma correcta y explica todos los pasos que va a realizar.
4. **Campo adicional: Librerías adicionales.** Donde el usuario deberá indicar si el programa le ha solicitado la instalación e alguna librería o programa adicional para funcionar.
5. **Campo adicional: Fallo de instalación.** Si el programa falla durante la instalación el usuario deberá indicarlo en este campo e indicar el error aparecido para posteriormente intentar determinar que ha podido ocurrir y si esto puede restar puntuación al candidato.

Una vez se tiene la información recopilada de las encuestas, los dos puntos correspondientes a este apartado se van a distribuir de la siguiente forma. Un punto va a corresponder al tiempo de instalación medio y se va a ponderar según la siguiente tabla.

|  |  |
| --- | --- |
| TIEMPO MEDIO DE ENCUESTAS | PUNTUACIÓN ASIGNADA |
| Tiempo menor de 5 minutos | 1 |
| Tiempo entre 5 y 7 minutos | 0.66 |
| Tiempo entre 7 y 9 minutos | 0.33 |
| Más de 9 minutos | 0 |

Tabla 6: Ponderación tiempo de instalación

El otro punto del apartado se va otorgar realizando una media de los puntos 2 y 3 de la encuesta, que corresponden a la valoración de proceso y valoración de su complejidad. En la siguiente tabla se especifica que puntuación se otorgará en función de la media obtenida.

|  |  |
| --- | --- |
| PUNTUACIÓN MEDIA | PUNTUACIÓN ASIGNADA |
| Más de 9 | 1 |
| Entre 6 y 9 | 0.66 |
| Entre 3 y 6 | 0.33 |
| Menos de 3 | 0 |

Tabla 7: Ponderación valoración del proceso de instalación

Si se encuentran librerias adicionales y fallos en la instalación, puntos 4 y 5 de la encuesta, se valorará en cada caso la penalización a aplicar.

### Criterio 2: Documentación, tutoriales y comunidad.

Como el presente proyecto pretende ser de alcance mayoritario, sin necesidad de conocimientos previos de ingeniería o programación, es fundamental que el programa elegido cuente con una buena documentación, tutoriales y una gran comunidad de usuarios.

En este apartado se incluyen toda la documentación oficial proporcionada por los propietarios del software y su soporte oficial, los tutoriales Tanto los proporcionados de forma oficial como los realizados por usuarios en plataformas de video online, como You Tube o Vimeo, en blog especializados y las comunidades o foros de desarrolladores que trabajan con el programa.

El pilar fundamental de este criterio es el análisis de los tutoriales oficiales. Estos tutoriales son la primera aproximación de los usuarios al programa y por lo tanto deben contar con todos los elementos para que se puedan llevar a cabo de manera correcta, rápida y entendible. Se necesita un programa con unos tutoriales que aporten conocimiento e inciten a seguir aprendiendo y desarrollando habilidades, ya que se pretende que el presente proyecto sea un punto de partida a un aprendizaje en profundidad del programa y por lo tanto es necesario contar con los recursos necesarios.

Se otorgará puntuación extra a aquellos programas que tengan documentación, foros y tutoriales en castellano.

Los subapartados que componente este criterio son los listados en la siguiente tabla.

|  |  |  |
| --- | --- | --- |
| CRITERIO | IMPORTANCIA | PUNTUACIÓN MAX-MIN |
| Documentación, foros y tutoriales oficiales | Normal | 0-4 |
| Análisis de los tutoriales oficiales | Clave | 0-8 |
| Comunidades y foros de desarrolladores. | Normal | 0-2 |

Tabla 6: Criterio: Documentación, tutoriales y comunidad

#### Medición de la puntuación: Documentación, tutoriales y comunidad.

Al tratarse de un apartado clave en la elección del framework , hay que tener muy claro cómo se va medir la puntuación del mismo.

Respecto al subapartado **documentación, foros y tutoriales oficiales** se va otorgar la puntuación en función de los recursos aportados por los propios desarrolladores del programa o por terceros pero que sean considerados oficiales por los propietarios del framework. Se tendrán en cuenta todos los elementos puestos a disposiciones a disposición de los usuarios, independientemente de su naturaleza o formato aunque se presentará especial atención al manual oficial, los tutoriales y si cuentan o no con un foro oficial.

En subcriterio anterior se medía la cantidad de recursos que se ponen a disposición de los usuarios de forma oficial, el subcriterio siguiente, **análisis de los tutoriales oficiales,** se trata de comprobar la calidad de estos recursos. Para medir este apartado se realizará uno o varios tutoriales oficiales y se analizará para comprobar si tienen o no errores, si su implementación es correcta, clara e entendible o si consiguen el objetivo de formar al usuario en el uso del programa e incitarla a continuar usando el programa.

Por último y para completar a los dos subapartados anteriores, nos encontramos con el subcriterio **comunidades y foros de desarrolladores** **no oficiales**. Para concretar la puntuación en este subapartado, se va comprobar la presencia de comunidades activas en Reddit[[3]](#footnote-3) , ya que es foro con mayor cantidad de usuarios a nivel mundial, y en Taringa, el equivalente en idioma español a Reddit [[4]](#footnote-4).

### Criterio 3: Funcionamiento y dificultad de uso.

El funcionamiento general del programa trata de detectar fallos técnicos tanto en la fase de instalación como posteriormente durante el uso del programa tales como errores que impiden la ejecución, el manejo o la realización de las distintas acciones y la estabilidad del programa.

Se puede decir que es una forma de medir si el programa se adecua a los estándares que un usuario puede comprender y usar de una manera rápida y efectiva consiguiendo sus propósitos funcionales.

La curva de aprendizaje es un apartado importante para decidir el programa va ser objeto de las acciones de este estudio ya que pretende ser un proyecto que en algunas partes del mismo, como puede ser las guías de aprendizaje puedan ser realizados posteriormente por usuarios de todos los niveles.

Por esta razón se busca un programa en el que la curva de aprendizaje no sea demasiado alta y pueda ser asumida por un usuario medio. Respecto a este criterio no se trata de la curva para aprender a dominar todos y cada uno de los aspectos del programa, sino de poder empezar a trabajar con el programa y crear pequeños proyectos, es decir, el foco está en el comienzo de la curva de aprendizaje.

Se busca por lo tanto un programa que sea muy sencillo en los primeros momentos y de adaptación muy rápida por parte de los usuarios sin importar tanto la dificultad de dominar todos los aspectos del mismo.

Los criterios que componentes este apartado de análisis son los mostrados en la tabla siguiente.

|  |  |  |
| --- | --- | --- |
| CRITERIO | IMPORTANCIA | PUNTUACIÓN MAX-MIN |
| Funcionamiento General | NORMAL | 0-4 |
| Primeros Pasos: Creación de un pequeño proyecto. | NORMAL | 0-4 |
| Funciones básicas | OTROS | 0-12 |
| Lógica de programación necesaria | NORMAL | 0-4 |

Tabla 7: Criterio: Funcionamiento general y curva de aprendizaje

#### Medición de la puntuación: Funcionamiento y dificultad de uso.

Dado que este criterio tiene un carácter puramente práctico se han de establecer las siguientes reglas o patrones para que la comparación entre los distintos programas sea lo más ecuánime posible, justa y adecuada para el propósito general del programa.

Detectar la curva de aprendizaje de un programa es una tarea bastante complicada de por sí, ya que entran parámetros subjetivos de la persona que está intentando identificar dicha curva de dificultad. Además en el caso que ocupa el presente trabajo presenta otra dificultad añadida, al tratarse de frameworks que tiene un funcionamiento en muchos apartados similar no se va apreciar la misma dificultad en el primer programa analizado que en los siguientes viendo ésta continuamente disminuida. Para mejorar ésta criterio que es posiblemente uno de los más importantes dentro de la comparación se van a realizar las siguientes acciones:

* Para cada programa se va realizar uno o varios tutoriales a ser posible oficial que consistan en la creación de un pequeño juego hasta completar aproximadamente 4 horas de trabajo con ellos. Documentando las dificultades que se van encontrando (en el caso que se trate de tutoriales oficiales los fallos o malas explicaciones en estos ya que afectan a otros criterios del estudio). Con lo que el apartado de “Creación de un pequeño proyecto” estará cubierto.
* Una vez completado este tiempo se estará en disposición de indicar una puntuación para los primeros pasos donde cero es la máxima dificultad y cuatro es muy sencillo.
* Con los conocimientos adquiridos en las horas dedicadas a la realización de un pequeño proyecto vía tutoriales se podrán definir como se hacen las funciones básicas de un juego y su dificultad, las funcionalidades básicas que se tendrán en cuenta son las siguientes: Movimiento de personaje por el escenario, mecánica básica de disparo, salto, colisión con otro objeto y desaparición de objeto colisionado.
* La puntuación del apartado “Creación de un pequeño proyecto” se asignará en función del nivel de complejidad conseguido el tiempo dedicado a la realización de los tutoriales oficiales, también influirá la cantidad de contenido creado o si han podido realizar diferentes tipos de juegos.
* A las puntuaciones del apartado **funciones básicas** se les va aplicar un factor corrector incremental que reducirá las puntuaciones en función de la posición en que se han realizado. **FACTOR multiplicador de 1 que se irá reduciendo en 0.05 en cada programa.** Estos frameworks pese a ser distintos tienen bastantes aspectos en común en su funcionamiento por lo que la percepción de dificultad se reducirá conforme se vayan analizando programas, por lo que es fundamental contar con el factor de corrección para equilibrar las puntuaciones y que los programas analizados en los últimos lugares no se vean beneficiados por el conocimiento adquirido en el uso este tipo de herramientas.

### Criterio 4: Editor Gráfico/Visual.

El contar con un editor visual para poder posicionar los objetos, personajes en el mundo y con el que poder crear relaciones entre ellos sin necesidad de *escribir código* y disponiendo solo de una edición gráfica es un punto esencial para seleccionar un framework u otro.

Se valorará no solo que tenga un editor gráfico sino que éste sea accesible, sencillo e intuitivo para los usuarios. Se tendrá en cuenta también el funcionamiento general, es decir, que no tenga bugs ni errores que impidan su funcionamiento o que dificulten el proceso de creación y desarrollo de los proyectos.

La usabilidad[[5]](#footnote-5) en el funcionamiento de la interfaz y de sus elementos también será un aspecto a tener en cuenta.

La tabla siguiente muestra la puntuación del presenta apartado.

|  |  |  |
| --- | --- | --- |
| CRITERIO | IMPORTANCIA | PUNTUACIÓN MAX-MIN |
| Editor/Gráfico visual | Clave | 0-8 |

Tabla 8: Criterio: Editor gráfico/visual

#### Medición de la puntuación: Editor Gráfico/Visual.

Este criterio tiene en la mayoría de sus aspectos un carácter subjetivo, por lo que la mejor forma de valorarlo es a través de realización de encuestas, una por cada programa, más una extra para comparación.

La muestra que van a realizar el estudio la compondrán personas con conocimientos en distintos campos tanto de la ingeniería informática y multimedia, como del diseño y profesionales dedicados al sector tecnológico.

La composición profesional de la muestra es la que muestra la siguiente tabla:

|  |  |
| --- | --- |
| Profesión/formación | Número de individuos. |
| Ingenieros Multimedia (Rama videojuegos) | 2 |
| Ingenieros Multimedia (Rama web) | 2 |
| Ingenieros Informáticos | 1 |
| Profesionales del sector tecnológico. | 2 |
| Arquitectos. | 1 |

Tabla 9: Participantes en la encuesta

El proceso que se va a seguir para la valoración de estos criterios por la muestra escogida será el siguiente y tendrá las siguientes características:

1. El estudio tendrá en cuenta solo las encuestas de los usuarios que completen las 5 encuestas que componen el estudio, esto se hace para mantener una coherencia y cohesión en los datos, los mismos usuarios valoran distintas interfaces.
2. El proceso tendrá una duración aproximada de un mes enviándose a los participantes una encuesta cada semana.
3. Cada encuesta tendrá un encabezado para que el usuario se identifique con un nombre, apodo o de cualquier forma que una vez completado todo el estudio se pueda recolectar los resultados tanto del conjunto como de cada encuestado individualmente, además es requisito para cumplir el punto número 1.
4. En la primera de las encuestas se integrará un apartado para que el usuario indique las características del equipo en el que va a realizar las pruebas y que va ser el mismo para todas las pruebas.
5. Los apartados que los encuestados deben valorar para determinar la puntuación de cada programa tienen una puntuación de entre 1 y 5 siendo cinco el máximo.
6. Para elapartado **editor visualintuitivo** se pedirá a los usuarios que valoren:   
   -A priori parece un programa sencillo de utilizar.  
   -Los iconos representan bien los elementos.
7. En el caso de valorar si el editor es visualmente atractivo, con un aspecto cuidado y actual los apartados a puntuar son los siguientes:  
   -La interfaz es atractiva.  
   -Tiene un aspecto cuidado y moderno.  
   -Puntúa el diseño del programa.
8. El buen funcionamiento del programa se medirá con los siguientes apartados:  
   -Rapidez al iniciarse.  
   -Consumo de recursos del programa.  
   -El programa se muestra rápido.

Existen dos apartados no obligatorios y no cuantificables, donde el usuario puede añadir información adicional que posteriormente si son completados habrá que valorar, son los siguientes apartados:

* Un apartado destinado a que el usuario indique si el programa no ha funcionado bien los errores que ha tenido. Si se produce algún caso de este tipo influirá negativamente en la valoración del funcionamiento general.
* Un campo para que el usuario valore con palabras la interfaz, que después habrá que interpretar.

### Criterio 5: Escalabilidad.

El presente proyecto en su apartado de guía de uso pretende ser una iniciación al uso de un framework de videojuegos tanto para usuarios expertos en programación como para usuarios medios.

Pero los conocimientos adquiridos durante la realización del mismo deben servir tanto para realizar pequeños proyectos como los que aquí se proponen como para futuros juegos más grandes y ambiciosos. De esta forma el programa seleccionado debe tener una gran escalabilidad que permita afrontar problemas mayores y seguir progresando en cuanto a la complejidad de los proyectos creados.

Los dos apartados que componente este criterio son los mostrados en la siguiente tabla y su puntuación es la siguiente.

|  |  |  |
| --- | --- | --- |
| CRITERIO | IMPORTANCIA | PUNTUACIÓN MAX-MIN |
| Posibilidad de crear funciones y scripts propios y/o lenguaje de programación propio. | Clave | 0-8 |
| Add-ons y plug-ins | Alta | 0-6 |

Tabla 10: Criterio: Escalabilidad

### Criterio 6: Resultados profesionales.

Los resultados que otros desarrolladores y/o creadores han conseguido con los diversos frameworks también es un criterio importante a la hora de seleccionar aquél sobre el que se va realizar las acciones del presente proyecto.

Elementos como la implantación de dichos motores en el mercado de juegos profesionales y los acabados gráficos y/o estéticos que se pueden conseguir con ellos son esenciales para que el motor sea adecuado para este estudio, ya que como se ha comentado pretende fomentar posteriormente un uso más profesional de dicha herramienta.

Un aspecto muy importante en este apartado es la diferenciación de los juegos profesionales o la no repetición de patrones visuales. Esto hace referencia a que los proyectos puedan ser tan diferentes tanto visualmente como funcionalmente de forma que no se parezcan entre ellos y se note la procedencia de los mismos. Es fundamental que el programa evite en la medida de lo posible que el usuario pueda detectar con un vistazo rápido con que herramienta ha sido desarrollado el juego. Se habla aquí de la primera sensación y de carácter usuario ya que un desarrollador experto puede detectar el motor gráfico.

En la tabla mostrada a continuación se encuentran los subapartados de este criterio, su importancia y su puntuación.

|  |  |  |
| --- | --- | --- |
| CRITERIO | IMPORTANCIA | PUNTUACIÓN MAX-MIN |
| Principales juegos desarrollados | Clave | 0-8 |
| Diferenciación | Normal | 0-4 |

Tabla 11: Criterio: Resultados profesionales

### Criterio 7: Estado actual.

A la hora de seleccionar un framework, se debe que tener en cuenta el estado actual del mismo. Se trata de encontrar un framework que permita un uso prolongado en el tiempo y para ello debe estar activo, es decir, no abandonado por sus desarrolladores. Se valorará también la frecuencia de actualizaciones ya que permite detectar que los bugs y problemas que surgen en todo programa con su uso se van solucionando, además que estas nuevas versiones permiten incluir nuevas funcionalidades.

Los elementos que componen el estado actual son los siguiente:

|  |  |  |
| --- | --- | --- |
| CRITERIO | IMPORTANCIA | PUNTUACIÓN MAX-MIN |
| Estado actual | Normal | 0-2 |
| Frecuencia de actualizaciones. | Mínima | 0-1 |
| Fecha desde la última actualización. | Mínima | 0-1 |

Tabla 12: Criterio: Estado actual

### Criterio 8: Sistemas operativos soportados y requerimientos.

El carácter mayoritario y generalista ya comentado del presente trabajo, aconseja tener en cuenta para la selección de la herramienta de desarrollo los requisitos y requerimientos del sistema para hacer funcionar los juegos. Estos deben ser adecuados para llegar al mayor número posible de usuarios teniendo en cuenta tanto el hardware como el software y sistema operativo.

En la puntuación de este criterio tendrán una gran influencia positiva aquellos programas aquellos programas que posibiliten la exportación de sus juegos a un mayor número de dispositivos posibles. En este apartado no se valora si esta exportación a los diversos dispositivos y sistemas operativos es una función propia del programa base y viene incluida en su licencia principal o necesita módulos especiales de pago o gratuitos para llevarla a cabo, ya que esto ya será tenido en cuenta en el criterio de licencia.

La siguiente tabla muestra la importancia y puntuación de este criterio.

|  |  |  |
| --- | --- | --- |
| CRITERIO | IMPORTANCIA | PUNTUACIÓN MAX-MIN |
| Posibilidades de exportación: Sistemas y plataformas | Clave | 0-8 |

Tabla 13: Criterio: Sistemas operativos soportados

#### Medición puntuación posibilidades de exportación.

Se van a evaluar las posibilidades de exportación de los juegos de cada framwork en 10 sistemas/plataformas, véase la tabla 16, por lo que se obtendrá una puntuación entre 10.

Para que un sistema o plataforma sea puntuado basta con que para uno de los subsistemas o plataformas sea posible realizar la exportación. Por ejemplo si el programa permite la exportación a Wii ,aunque no lo sea posible para Wiiu , su puntuación será completa en ese sistema general.

Posteriormente está se ponderará sobre la importancia del criterio, 8 puntos. Si bien pueden existir factores que aumenten o disminuyan está puntuación que se tratarán y justificarán en cada caso.

|  |  |
| --- | --- |
| SISTEMA/PLATAFORMA | INCLUYE |
| Windows | Sistemas operativos Widnows |
| Linux | Sistemas operativos basados en el kernel de Linux. |
| IOs | Para sistemas de escritorio y móvil. |
| Android |  |
| HTML 5 |  |
| Flash |  |
| Windows Store (Mobile y Store escritorio) | Móvil y tienda de escritorio. |
| Sistemas PlayStation (PS3, PS-VITA, PS4) | PS3, PS-Vita y PS4. |
| Sistemas Xbox (XBOX360, XBOX ONE) | Xbox 360. Xbox One. |
| Sistemas Nintendo (3DS, Wii, WiiU) | 3ds , Wii, Wiiu. |

Tabla 16: Sistemas de exportación a evaluar.

### Criterio 9: Licencia.

El precio del software necesario y el tipo de licencia de cada programa a analizar también es un punto importante a la hora de seleccionar el mismo.

Al pretender ser un proyecto abierto y generalista tendrán mejor puntuación aquellas herramientas con un precio menor o aquellas que sus versiones gratuitas cuenten con unas menores limitaciones de uso.

Pese a ello, este criterio tiene menos importancia que otros ya que no se debe desvirtuar el objeto del estudio, ni que por seleccionar un software de menor precio o licencia con menos limitaciones no se llegue a la consecución de determinados objetivos o no se completen de manera satisfactoria.

|  |  |  |
| --- | --- | --- |
| CRITERIO | IMPORTANCIA | PUNTUACIÓN MAX-MIN |
| Precio de Licencias | Normal | 0-4 |
| Modo de prueba o free | Poca | 0-2 |
| Precio módulos de exportación | Poca | 0-2 |

Tabla 14: Criterio: Licencias

### Puntuación total.

A continuación, se presenta la tabla con el resumen de las puntuaciones de todos los criterios.

|  |  |
| --- | --- |
| CRITERIO | PUNTUACIÓN MIN-MAX |
| Requisitos e instalación | 0-6 |
| Documentación, tutoriales y comunidad | 0-16 |
| Funcionamiento y dificultad de uso | 0-24 |
| Editor Gráfico/Visual | 0-8 |
| Escalabilidad | 0-14 |
| Resultados profesionales | 0-12 |
| Estado actual | 0-4 |
| Sistemas operativos soportados y requerimientos. | 0-8 |
| Licencias | 0-8 |
| TOTAL | 0-100 |

Ilustración 1: Puntuación Total

## 3.2 Primera selección de framework.

Antes de proceder a un análisis de los posibles candidatos, se ha realizado una primera lista de posibles motores, que pueden cumplir los criterios referidos en el punto anterior y ser el medio con el que se van a cumplir los objetivos del presente trabajo.

Esta primera lista parte de la búsqueda en general de motores gráficos y frameworks para desarrollar juegos 2D, sin entrar a valorarlos con los criterios anteriormente citados.

De ellos se descartaran automáticamente aquellos que posean características incompatibles con los objetivos planteados, aquellos que se vea sin necesidad de un estudio completo que no cumplen los requisitos buscados o aquellos que por otros motivos justificados no pueden ser una opción.

Los siguientes programas mostrado en la tabla son los que más se ajustan a priori al objeto de estudio del proyecto y por lo tanto se consideran a priori candidatos a ser el seleccionado para el mismo.

|  |  |
| --- | --- |
| Click Team Fusion 2.5 | Torque 2D |
| RPG-Maker | Gdevelop |
| Unreal 4 2D | Game Maker |
| Cocos2D | GODOT |
| Construct 2 | Marmalade |
| Eclipse Origins | Unity 2D |
| Adventure Game Studio | Stencyl |
| Angel 2D |  |

Tabla 15: Primera selección de motores

### Motores descartados y justificación

De la tabla mostrada anteriormente se han descartado una serie de programa a continuación se explican las razones de la exclusión de cada uno de ellos.

#### Resultado de imagen de torque 2d logoTorque 2D

Figura 2: Torque 2D

Tabla 16: Logo Torque 2D

Torque 2D o Torque Game Builder es la adaptación a 2D del motor gráfico Torque 3D creado por la compañía GarageGames para su juego Tribes 2 en 2001. Esta versión 2D del motor fue lanzada en 2013 bajo licencia MIT[[6]](#footnote-6) lo que le confiere un punto a favor al tratarse de software libre.

Pese a ser a contar con una licencia libre Torque 2D tiene la posibilidad de comprar una interfaz de desarrollo Torque Studio.

Después de una primera aproximación se ha podido comprobar que la instalación y configuración es demasiado compleja para el usuario medio, la creación de proyectos es excesivamente tediosa y es prácticamente indispensable contar con conocimientos de programación para poder crear contenido por lo que se ha descartado.

#### Unreal 4 2D

Unreal 4 es la evolución del motor desarrollado en 1998 para el juego del mismo nombre y que bajo él y sus constantes evoluciones se han desarrollado juegos tan populares como Deus Ex, Bioshock, Batman: Arkham Asylum o Mass Effect.

Tabla 17: Unreal 4

Figura 3: Unreal Engine

Como pasara en su día que evolucionó desde un motor exclusivo de juegos en primera persona o shooters a abarcar también juegos tercera persona ahora en sus últimas versiones intenta también dar soporte a juegos 2D.

Aunque su potencia es muy grande y se pueden realizar grandes proyectos con él toda su interfaz está pensada para elementos tres dimensiones con lo que en algunos aspectos resulta bastante complejo el manejo de las dos dimensiones, además es un motor bastante exigente respecto a los requerimientos del sistema tanto para trabajar con el cómo en los proyectos resultantes.

#### Cocos2D.

Cocos2D es un framework para diseñar y crear juegos 2D inicialmente basado en Phyton aunque diferentes versiones lo han adaptado a diferentes lenguajes de programación como por ejemplo la versión X o la versión Js que dan soporte a C++ y Javascript respectivamente.

Figura 4: Cocos2D

Cocos2D se descarta en esta primera fase ya que aunque posee una creación rápida de proyectos, un manejo fácil y rápido de Sprites es imprescindible programar ya sea en Phyton con la versión original o en los lenguajes propios de otras versiones.

Ilustración 2: Cocos 2D

Cuenta además con la desventaja de no contar con una interfaz visual de manejo, creación y posicionamiento de los elementos, su funcionamiento es similar a motores como SFML, en los cuales tienes que llamar desde código a las funciones propias del mismo para realizar diversas acciones, pero añadiéndole su propio entorno de programación.

#### GODOT Engine.

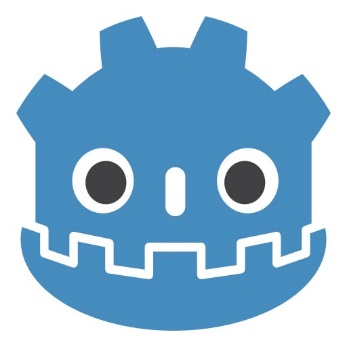
A GODOT Engine es un motor de videojuegos multiplataforma y de código abierto. Su funcionamiento basado en nodos y escenas lo diferencia en gran medida de los motores y framworks analizados en el presente proyecto.

Figura 5: GODOT Engine

Uno de los objetivos de este proyecto es que usuarios no programadores se acerquen al desarrollo de videojuegos. Puedan crear sus primeros proyectos, para posteriormente con la base adquirida poder aumentar su conocimiento, esto incluye poder utilizar cualquier otro motor o framework que no sea el seleccionado, por lo que uno que trabaje de forma tan distinta y diferencia del resto no es un buen candidato.

A parte de su funcionamiento diferente nos encontramos también en un caso como el de Cocos 2D, donde necesitamos nociones de programación para poder realizar proyectos en GODOT.

#### Marmalade.

Marmalade es un SDK y motor gráfico basado en C++. Es uno de los SDK´s de desarrollado más utilizados por los grandes estudios para crear juegos 2D y 3D , juegos como Lara Croft and the Guardian of Light, Plants vs Zombies , Cut the Rope o el modo Zombies 2D de Call of Duty: World at War. Ha sido premiado en los Games Industry Awards 2015 en la categoría Engine and Middleware, Tools and Tech.

Figura 6: Marmalade

Pese al gran potencial de Marmalade y las cotas de calidad que pueden alcanzar sus juegos, su entorno basado en el lenguaje C++ que se debe conocer y dominar para crear contenido y su alta curva de aprendizaje hacen que sea descartado.

#### Eclipse Game Engine.

Eclipse Origins es un motor gráfico y una interfaz o suite para la creación de MMORPG [[7]](#footnote-7) 2D. Su uso es sencillo y su interfaz visual hace bastante sencillas las tareas de creación de escenarios y colocación de Sprites.

Su limitación a un solo género, la escasa documentación y comunidades activas unido a que es motor que prácticamente ha quedado en desuso hacen que no sea un buen candidato.

#### Unity 2D

Unity es uno de los motores de videojuegos más utilizados hoy en día tanto por grandes equipos como por pequeños estudios independientes. Su gran escalabilidad, gran interfaz visual, carácter multiplataforma y el potente sistema de scripts lo convierten un motor apto para cualquier nivel usuarios.

Figura 7: Unity

Pero el problema principal que hace que sea descartado es que Unity está pensado principalmente para desarrollo de juegos 3D y aunque sus últimas versiones incorporan la posibilidad de desarrollar juegos 2D todavía no tiene toda la potencia que sí tiene para juegos 3D. Aunque no se duda de que en futuras versiones sigan mejorando en ese sentido.

Al igual que con Unreal 2D, al ser un motor 3D adaptado a 2D, es pesado y consume excesivos recursos.

#### Stencyl

Stencyl es un programa de creación de videojuegos 2D para computadoras y dispositivos móviles. Es un software gratuito con la opción de comprar módulos de pago.

Stencyl funciona dividiendo cada ciclo de ejecución en tres pasos: creació , dibujado y actualización de datos. Para las físicas hace uso del motor Box2D.

El principal inconveniente de Stencyl es que usa Flash para la exportación de sus juegos y las vulnerabilidades de Flash han hecho que la mayoría de navegadores hayan dejado de dar soporte a dicha tecnología. Aunque es conveniente seguir usando las aplicaciones en Flash incluyendo Flash Player en el sitio Web es conveniente tratar de usar otros programas que hagan uso de otras tecnologías.

#### Tabla resumen de programa descartados

La siguiente tabla muestra los motores descartados y las razones fundamentales para cada uno de ellos, algunos aparecen en varios motivos.

|  |  |
| --- | --- |
| MOTIVO | CANDIDATO EXCLUIDO |
| Complejidad | Torque2D, Unreal 2D, Marmalade. |
| Alto consumo de recursos | Unreal 2D, Unity2D |
| Necesidad de conocimientos en programación | Cocos2D, GODOT Engine. Marmalade |
| Diferencia excesiva con el resto | GODOT Engine. |
| Limitaciones | Eclipse Game Engine. |
| Escasa documentación | Marmalade. |
| Motivos de Seguridad | Stencyl |
| Programa en desuso | Eclipse Game Engine, Stencyl |

Tabla 18: Exclusiones de Frameworks

### Motores preseleccionados.

Después de los programas excluidos han quedado candidatos posibles para el objeto de nuestro proyecto. Es a estos frameworks a los que se les hará el estudio completo basado en los criterios expuestos.

#### Gdevelop

Gdevelop es un software para la creación de juegos 2D y 3D, si necesidad de utilizar ningún lenguaje de programación. Gdeveloper es un programa de código abierto y totalmente gratuito.

Figura 8: Gdevelop

El programa está escrito en C++ y trabaja fundamentalmente con la librería gráfica SFML , añadiéndole por encima una interfaz basada en WxWidgets[[8]](#footnote-8) y utiliza también TinyXMl para la gestión de mapas y Tiles.

Gdevelop trabaja en base a varios editores que abarcan todo el proceso de creación de un videojuego. Los editores con los que cuenta Gdevelop son los siguientes:

* **Gestor del proyecto:** Es el editor principal del programa en el cual se indican los datos básicos del juego como pueden ser nombre o autor o pantallas de cargas. A través de este editor se accede al resto de editores. Funciona como entorno principal de trabajo.
* **Editor de escenas:** Permite organizar cada escena, pantalla o nivel del juego de manera individual. En él se realiza por ejemplo el posicionamiento de objetos o enemigos. Cuenta con un depurador para comprobar el correcto funcionamiento de la escena y un *player* que posibilita probar la escena.
* **Editor de objetos en la escena:** Posibilita la creación de los distintos tipos de objetos que pueden encontrarse en una escena. Permite crear distintos tipos de objetos como podrían ser Sprites, Textos u objetos 3D. En este editor se añaden todos los efectos tanto de físicas como de partículas o comportamientos predefinidos.
* **Editor de las capas de la escena:**  Gdevelop puede trabajar con diferentes capas dentro de una escena, lo que ayuda a organizar y mejorar el trabajo pudiendo por ejemplo tener una capa para enemigos, otra para objetos de fondo, otros para objetos con los que poder colisionar, ….
* **Editor de eventos:** Gestionan los eventos de la escena y permiten realizar las funciones propias del juego. Este editor compone toda la capa lógica del juego y en la que se va definir toda la jugabilidad del mismo.
* **Banco de imágenes:** Gestiona todas las imágenes, videos y gráficos que se van a utilizar en un proyecto.

|  |  |
| --- | --- |
| GDEVELOP | |
| Fecha de lanzamiento | 2014 |
| Programado en | C++ |
| Lenguaje de programación | Javascript |
| Licencia | Libre |

Tabla 19: Gdevelop Datos

#### Clickteam Fusion.

Click Team Fusion, apartir de ahora (CTF) es un framework para la creación de videojuegos especializada en juegos 2D lanzado en el año 2013 que actualiza la herramienta de creación y programación de juegos Multimedia Fusion que fue uno de los primeros programas que permitían crear juegos de manera visual aunque de una manera limitada.

Pero con la actualización de Multimedia Fusion a Click Team Fusion se añadieron toda una serie de herramientas que hicieron del programa una potente herramienta para crear videojuegos de manera rápida y visual. Algunas de estas características fueron por ejemplo la integración del motor de físicas Box2D que le daba a las mismas un acabado profesional, un nuevo sistema de objetos que hacía más sencilla la tarea de creación de los mismos y una nueva interfaz que llevaba la programación gráfica a un nuevo nivel no visto en las versiones anteriores.

Lo más destacable de CTF es su sistema de gestión de eventos, que constituye el núcleo central de todo el programa, en la Figura 9 se observa un ejemplo de eventos de CTF.

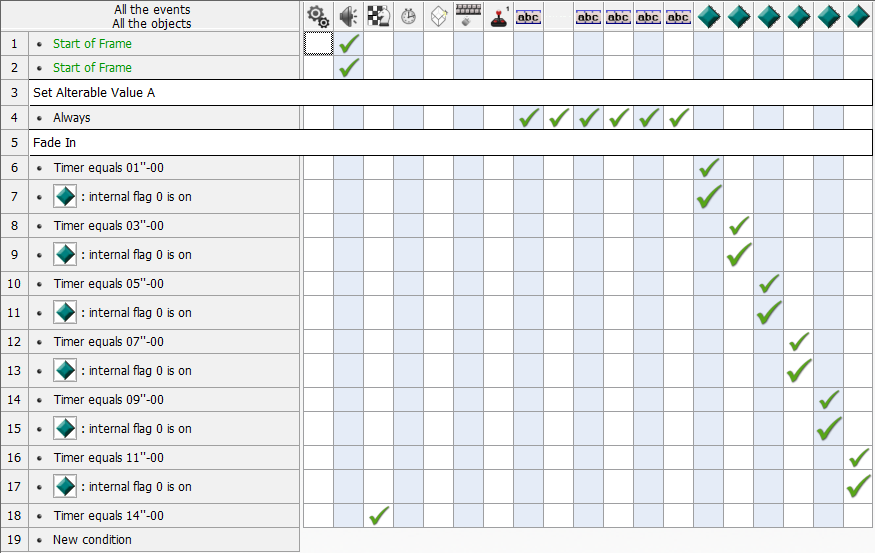


Figura 9: Sistema de eventos Click Team Fusion

|  |  |
| --- | --- |
| CLICK TEAM FUSION | |
| Fecha de lanzamiento | 2012 |
| Programado en | C, C++ |
| Lenguaje de programación | C, C++, Propio |
| Licencia | Software Propietario |

Tabla 20: Click Team Fusion Datos

#### Construct 2

La primera versión de Construct de código abierto fue creada por un grupo de estudiantes,cuya ultima build fue lanzada en 2012 dando por finalizado en ese momento el ciclo de vida del programa.

Construct 2 fue una evolución comercial del primer Construct ,creada por el mismo grupo de desarrolladores ya constituidos en la empresa Scirra, y sacada al mercado en Febrero de 2011.

Figura 10: Construct 2

Pensando como un motor de creación de videojuegos para usuarios sin conocimiento de programación, está basado basando en dos aspectos claves: el drag and drop de elementos y en la otorgación de patrones de comportamiento de manera visual.

Su funcionamiento principal es a través de las hojas de comportamiento y/o evento, en las que a partir del cumplimiento de una serie de condiciones el programa ejecuta unas sentencias previamente definidas por el usuario. Además, a estas condiciones se les pueden añadir subcondiciones, o agregaciones para hacer todo lo complejo que uno quiera el comportamiento de los objetos del juego. En la Figura 11 se puede ver un ejemplo de una hoja de comportamiento de Construct 2.

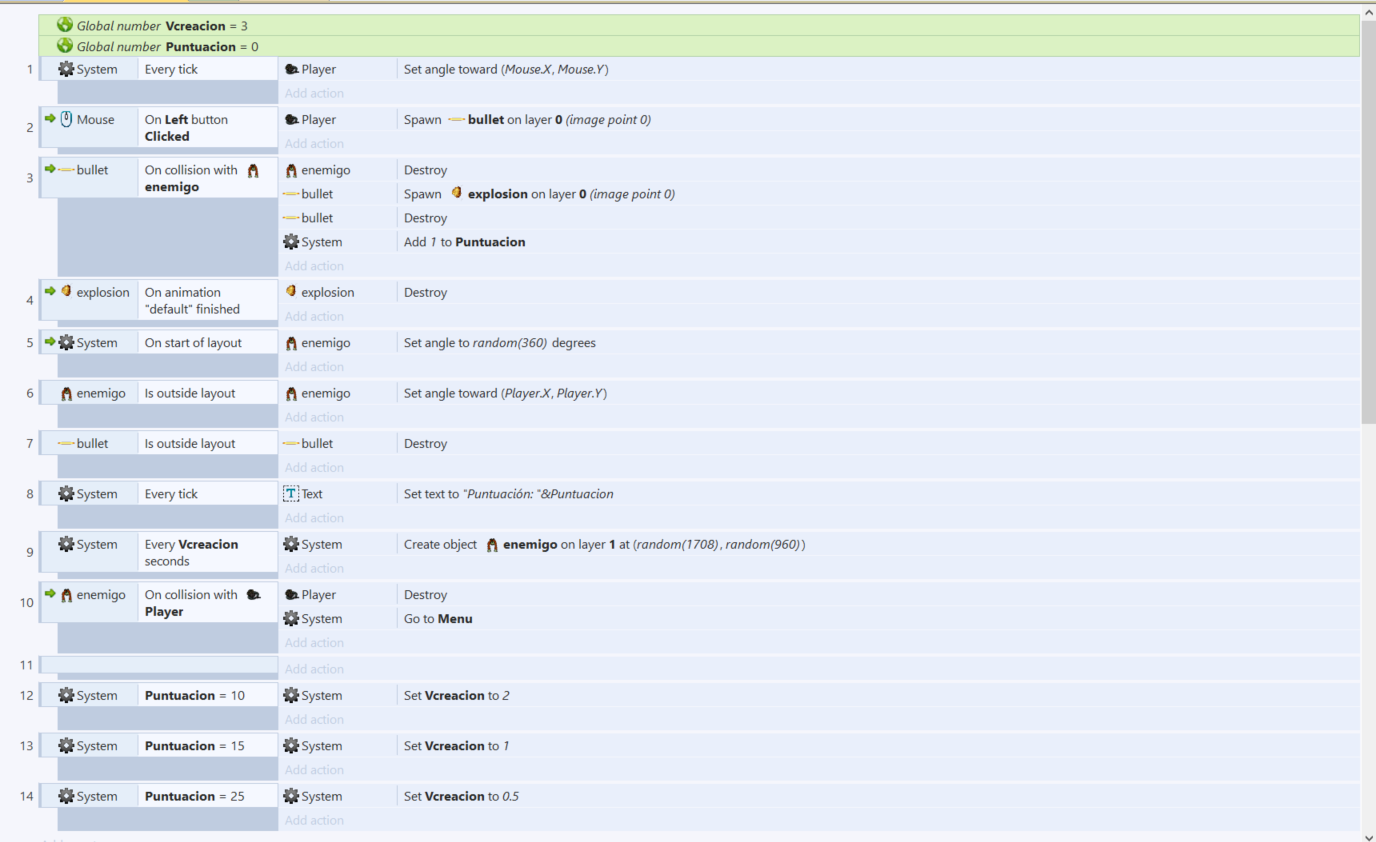


Figura 11: Hoja de Eventos Construct 2

|  |  |
| --- | --- |
| CONSTRUCT 2 | |
| Fecha de lanzamiento | 2011 |
| Programado en | C++, Javascript |
| Lenguaje de programación | Javascript |
| Licencia | Software Propietario |

Tabla 21: Construct 2 Datos

#### Game Maker

Fundamentalmente divide su funcionamiento en dos grandes apartados, por un lado en el sistema de eventos y comportamientos, que se compone de hojas de eventos en la que cada hoja tiene una serie de eventos y sus sentencias condicionales y una vez que se cumpla se realizará el comportamiento asignado. Y por otro lado en evitar la selección de instancias mediante la selección de los objetos por medio del menú contextual que especifica el creador en las hojas de comportamiento.

Game Maker Studio, (a partir de ahora GM) es un software y kit de desarrollo para la creación de videojuegos creado en **Delphi.** Nació en el ámbito académico a finales de los noventa en Países Bajos. El profesor de la universidad de Utrecht, Mark Overmars, empezó a desarrollar una herramienta para ayudar a los estudiantes en la animación de personajes tanto de juegos como de otros ámbitos del dibujo, pero el proyecto fue creciendo y se convirtió en un software para que los estudiantes pudieran crear videojuegos en los primeros momentos de su carrera con pocos conocimientos de programación. En 1999 GM dejo de ser únicamente un programa para los alumnos de Overmars y fue lanzado al mercado.

Figura 12: Game Maker

La principal característica de Game Maker es que se intenta adaptar a todo tipo de usuarios, ya que permite realizar la implementación de las funcionalidades de dos formas completamente distintas: una desde su interfaz visual haciendo uso del Drag and Drop sobre objetos y otra mediante su lenguaje de programación GML.

En 2012 Yoyo Games (compañía propietaria de GM) lanzó en paralelo a la versión estándar de Game Maker otra versión llamada Game Maker: Studio, que incorporaba cambios profundos el programa tanto a nivel visual como a bajo nivel e incluía nuevas posibilidades de exportación a plataformas y sistemas operativos. En 2015 la versión Game Maker fue retirada por Yoyo Games y la rama Game Maker Studio se convirtió en la única versión del programa.

|  |  |
| --- | --- |
| GAME MAKER | |
| Fecha de lanzamiento | Noviembre 1999 |
| Programado en | Delphi |
| Lenguaje de programación | GML (Propio) |
| Licencia | Software Propietario |

Tabla 22: Game Maker Datos

# Metodología

El presente proyecto cuenta con varios apartados bien diferenciados entre si donde algunas fases son de análisis e investigación y en otras es implementación y creación de contenidos, por lo tanto es difícil nombrar una metodología que se adapte a todo el proyecto.

Para suplir este déficit se van a implementar diferentes metodologías o formas de trabajar para cada una de las partes que componen el trabajo. Si bien se contará con una metodología común para todo el proyecto para llevar la gestión de tiempos y de las funciones propias de la administración.

## 4.1 Metodología de trabajo en general.

Para las tareas de organización y administración del trabajo se va seguir una metodología por tiempos, prioridades e importancia con una **pila de tareas** que nos va indicar que tareas debemos realizar en cada momento y cuál es la siguiente a realizar. Al comienzo del proyecto se han delimitado todas las tareas que hay que realizar para completarlo, justo a esta delimitación cada una de ellas cuenta con su tiempo de aproximado de realización, su prioridad y su escala de importancia.

**El tiempo aproximado de realización**  va medir el tiempo que debido a una análisis previo de la tarea en cuestión se ha determinado se va tardar en completar la misma. Este campo va servir para posteriormente comprobar si el análisis previo de los tiempos de las tareas se hizo correctamente, así como para realizar las mediciones de las desviaciones de tiempos de los distintos apartados del proyecto.   
Estos tiempos no son tiempos fijos ya que al ser un trabajo en el a priori mucha de la tecnología que se va usar se desconoce dar un tiempo aproximado a la realización de las tareas sería tremendamente osado y daría lugar a desviaciones muy grandes, por lo que estas estimación se van a ir realizando conforme se tengan datos más o menos fiables de los tiempos que se va tardar en realizar dicha tarea. Por lo tanto, la tabla de tiempos va ser dinámica y va estar evolucionando junto con el proyecto.

**La escala de importancia** mide la importancia de la tarea dentro del proyecto. Da una visión de la tarea en relación a las demás y al proyecto. Esta escala sirve para delimitar las partes del proyecto y/o tareas que resultan fundamentales para el mismo y sin las cuales el mismo pierde razón de ser y aquellas que tienen carácter más accesorio o secundario. Dentro de esta escala de creación ad hoc para el proyecto se ha establecido los siguientes niveles:

1. **Imprescindible:** Con este grado de la escala indicamos aquellas tareas o partes del proyecto sin los cuales el mismo pierde su razón de ser, por lo tanto prevalecen sobre todas las demás y si no se realizan el proyecto no puede continuar. Son las primeras que se deben desarrollar y sin ellas completadas el proyecto se considerará a todos los efectos inconcluso. La característica principal de este tipo de tareas y que sirve para identificarlas es que tienen valor por si mismas pueden ser entendidas sin necesidad del resto.
2. **Importantes:** Las tareas dentro de esta escala son necesarias para que el mismo se pueda entender, sin ellas el trabajo carecerá en parte de sentido y quedará incompleto. Son casi tan importantes como las Imprescindibles pero necesitan de estas para ser entendidas.
3. **Secundarias:** Aquí se trata de elementos del proyecto que ayudan y complementan a las anteriores, pero sin ellas el proyecto puede ser entendido. Incluyen elementos que dan más sentido y refuerzan los análisis y/o desarrollos realizados en las tareas Imprescindible y Muy Importantes. Un ejemplo claro de este tipo de apartados podría ser el aumentar el número de motores gráficos a analizar, el número de ejemplos de juegos analizados o el número de funciones recurrentes del manual de uso de Game Maker.
4. **Accesorias:** Esta categoría trata de aquellas tareas que no son propias del proyecto, ni complementan a las anteriores, sino que son tareas que sirven a modo de anexos o complementos para dotar al conjunto de una serie de elementos a su alrededor. Por ejemplo, la inclusión de juegos o proyectos anexos de Game Maker o la realización de una recopilación accesoria de manuales, cursos y videos para aprender a programar con el tipo de programas que son objeto de estudio son tareas de este tipo.

**CATEGORIA EXTRAORDINARIA: ADMINISTRATIVAS:** Existen tareas que no forman parte del proyecto en sí, son partes administrativas, propias de la estructura o del formato seguido para la realización del mismo. Además el carácter institucional e educativo implica ciertas partes que hay cumplimentar necesariamente. Se trata de una categoría excepcional ya que esta fuera de las propias y/o accesorias del trabajo y no se puede establecer una jerarquía de realización con el resto. En general este tipo de tareas se deben realizar en el momento que se tengan los datos necesarios para realizarlas prevaleciendo sobre el resto.

**La prioridad** va dar información de la importancia de la tarea dentro de la escala de importancia del punto anterior en el momento actual del proyecto. Una tarea con prioridad alta va ser clave para que el trabajo pueda continuar y por lo tanto tiene prevalencia sobre otras tareas con menor prioridad (siempre qué se encuentren dentro del mismo grado de la escala de importancia). Las prioridades en el proyecto se clasifican en alta, media y baja.

**Cola de tareas por escala y prioridades:** Para saber que tarea se va a realizar a continuación se utilizará una cola de tareas clasificadas por escala de importancia y prioridad en la que esta las tareas están ordenadas por orden que se van a realizar. Esta cola no es una cola inamovible y fija ya que puede ir modificándose con el paso del proyecto por los siguientes motivos:

1. Aparecen tareas con del tipo “Categoría extraordinaria: Administrativas” que como se ha comentado anteriormente se deben realizar en el momento que se cuente con los elementos necesario para hacerlo por lo que en ese momento pasaran al principio de la cola.
2. La prioridad de una tarea no fue medida correctamente porque no se conocía en profundidad el campo de conocimiento que la componen o surgen elementos nuevos que cambian la importancia de la misma.
3. Surgen tareas nuevas no previstas y que se convierten en claves para poder continuar con otra tarea.
4. Aparecen subdivisiones de tareas ya existentes.



Figura 13: Cola de tareas por prioridades

En figura anterior podemos ver la Cola de tareas por prioridades y esta cuenta con los siguientes campos:

1. **Escala de importancia.**
2. **Prioridad dentro de la escala.**
3. **Nombre de la tarea.**
4. **Fecha de inicio.**
5. **Fecha de finalización.**
6. **Porcentaje de realización:** El porcentaje de realización va a servir para poder realizar en determinados momentos una revisión general del estado de las tareas, ya que aunque sea una cola en la que las tareas en la parte superior se deben realizar antes que las que están por debajo de la misma, esto no implica que no puedan realizarse varias tareas en paralelo o que algunas de ellas se encuentren en estado bloqueado o la espera de algún elemento o recurso necesario para continuar con ella.

## Procedimiento y selección de tareas.

Como hemos comentado en el punto anterior la metodología del trabajo de la parte general y administrativa va funcionar mediante una cola de tareas basada en una escala de importancia y una prioridad. Para determinar la posición en esta cola el proceso que se debe de seguir es el siguiente:

1. **Determinar la escala de la tarea:** Mediante un estudio previo del proyecto y de las tareas se debe clasificar todas las tareas del proyecto dentro de la escala.
2. **Determinar prioridad de la tarea:** Una vez que las tareas cuente con su escala se determina para cada una de ellas su prioridad.
3. **Posicionar las tareas:** Para posicionar las tareas tenemos que tener en cuenta en primer lugar la escala de importancia, cuánto más alta sea la tarea se posicionará más arriba dentro de la cola. Cuando encontremos , tareas que se encuentran dentro del mismo grupo de la escala, atenderemos a la prioridad para determinar cuál se realizan primero.   
   Pero existe una excepción a esta regla general de posicionamiento, esta excepción se aplica a las tareas de la categoría extraordinaria administrativas, que como se ha explicado anteriormente son propias del formato y del carácter del proyecto, que en cuanto las circunstancias permitan realizarlas pasaran automáticamente a la primeras posiciones de la cola y tendrán que realizarse en primer lugar.

## Tutorización del proyecto.

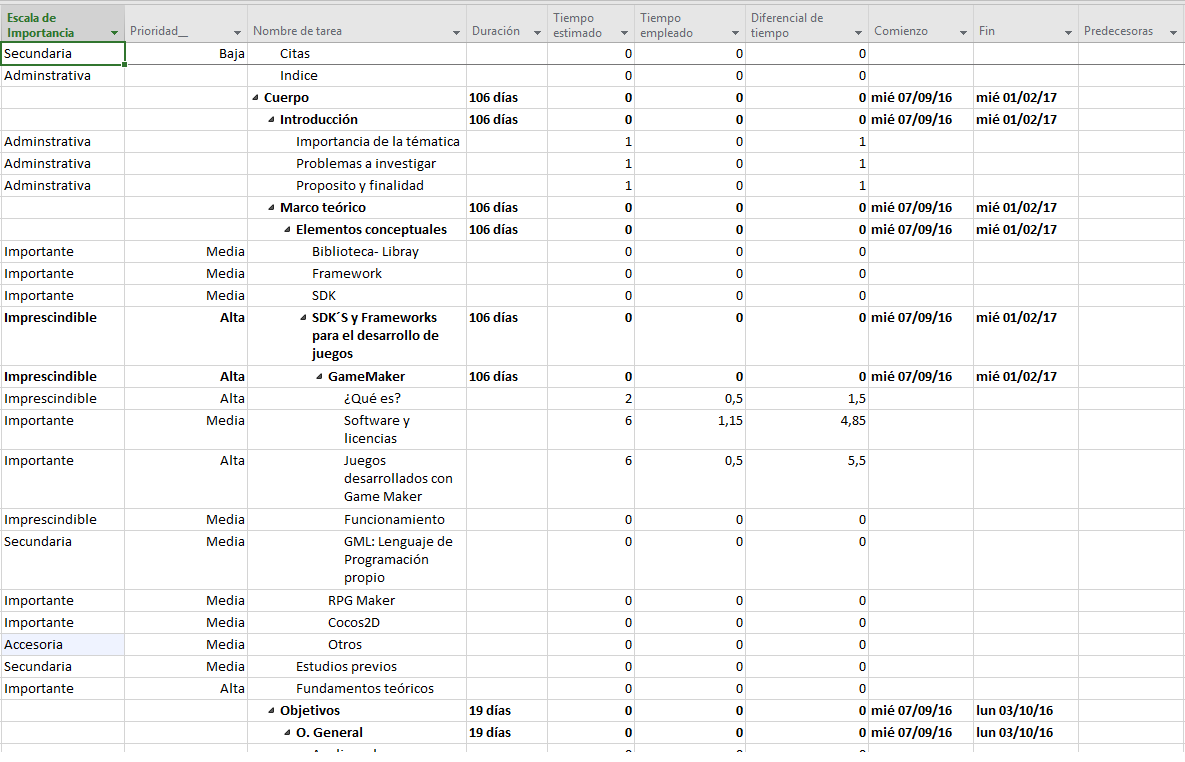
El presente proyecto va contar con una supervisión y control por parte de un tutor. La función del tutor son las de ser un apoyo al que poder acudir en caso de dudas respecto al proyecto, su organización o la redacción del presente documento.

El tutor tiene acceso en todo momento a todo a todo el material de trabajo actualizado para poder visualizar y supervisarlo si así lo desea. Además, se mantiene una comunicación constante y fluida con el mismo para que sepa en la situación que se encuentra el proyecto y si surgen problema durante el desarrollo.

Toda esta labor de tutorización vía comunicación email y con acceso a disposición de todo el material de desarrollo se complementa con reuniones presenciales constante. En general se mantendrán reuniones cada dos semanas para supervisar los avances y para detectar posibles desviaciones en la planificación del proyecto. Pero en épocas de mayor carga de trabajo o si surgen problemas puntuales se podrá aumentar la frecuencia de la misma o mantener reuniones excepcionales.

## Organización y visión general vía Project.

Para tener una visión de todo el proyecto e ir llevando un seguimiento de todos los aspectos del proyecto es necesario contar con alguna herramienta que permita visualizarlo en su conjunto de una manera rápida y clara el estado que se encuentran las tareas, cuales están terminadas, retrasadas o bloqueadas y la gestión de su tiempo. Para esta función se utiliza el software Microsoft Project.



Con Project y el sistema de campos que se ha aplicado se puede apreciar las tareas principales y las pequeñas tareas dependientes de estas. Además fruto de la metodología que con la que se va a llevar a cabo el proyecto se han añadido una serie de campos que aumentan la información y que son necesarios para tener esa visión global que con esta herramienta se persigue conseguir.

Campos de Project:

1. **Escala de Importancia.**
2. **Prioridad.**
3. **Nombre de la tarea.**
4. **Duración.**
5. **Tiempo estimado:** Conforme se disponga de la información necesaria para realizar una aproximación del tiempo que se va a necesitar para desarrollar una determinada tarea se incluirá en el documento.
6. **Tiempo empleado:** El tiempo que se ha tardado en completar la tarea. Este tiempo se va a medir gracias a la herramienta Toggl, que permite una cuantificación exacta del tiempo de realización de cada tarea.
7. **Diferencia de tiempo:** Muestra la diferencia entre el tiempo inicialmente previsto y el que posteriormente se ha tardado en desarrollar una tarea, siendo los valores los que indican que se ha tardado más de lo que inicialmente se ha previsto y los negativos a la inversa, que se ha tardado menos de lo que en la fase de estimación se pensó.  
   Este valor va permitir una vez terminado el proyecto medir lo correcta que fue la planificación y calcular diferentes índices que ayuden a entender cómo ha ido el proceso.
8. **Comienzo.**
9. **Fin.**
10. **Predecesora:** Indica si alguna tarea necesita es dependiente directa de otra, es decir, que necesita que otra este completa para poder empezar realizarse, si así fuera en este campo aparecería la tarea en cuestión.

# Cuerpo del trabajo

## Análisis de Frameworks

### Consideraciones previas.

#### Versiones de los programas.

Las pruebas tanto de instalación, realización de proyectos y prueba de los juegos se va a realizar en el mismo equipo para que la comparación entre los diferentes candidatos sea lo más justa posible. El equipo de prueba cuenta con las siguientes características:

|  |  |
| --- | --- |
| Equipo de pruebas | |
| Sistema Operativo | Windows 10. |
| Procesador | I7 4750HQ |
| Memoria | 8GB de RAM |
| Tarjeta Gráfica | GTX 950 |
| Otros | Conexión internet 100Mb. |

Tabla 1: Datos equipo de pruebas

#### Versiones de los programas.

Para realizar la implementación de las funcionalidades, la creación de pequeños proyectos y todos aquellos análisis que requieran el uso de los frameworks se va usar siempre que sea posible la versión gratuita de los mismos.

#### Precio medio de las licencias y de los módulos de exportación.

Para poder evaluar los apartados precio de licencias y de los módulos de exportación es necesario tener un baremo con el que determinar si estos precios son altos o bajos. Para ello se va usar el precio medio, tanto de las licencias como de los módulos, del conjunto de los programas. En la siguiente tabla se muestran los precios medios.

En la siguiente tabla se detalle el precio de las licencias más avanzadas de los programas y su precio medio.

|  |  |
| --- | --- |
| Precio medio licencias | |
| Gdevelop | 0 € |
| Contruct 2 | 405,99€ |
| Click Team Fusion | 211.99€ |
| Game Maker | 99€ |
| PRECIO MEDIO | 179,25€ |

Tabla 2: Precio medio de licencias

Para delimitar la puntuación de licencia se va usar el siguiente baremo.

|  |  |
| --- | --- |
| Baremo puntuación precio de licencias. | |
| Precio medio +/-20€ | 50% del total de la puntuación. |
| Precio medio hasta +50€ | 25% del total de la puntuación. |
| Precio medio hasta -50€ | 75% del total de la puntuación. |
| Precio medio + de 50€ | 0% del total de la puntuación. |
| Precio medio – de 50€ | 100% de la puntuación. |

Tabla 3: Baremo puntuación precio licencias

En el caso de los módulos de exportación para calcular la media se van utilizar el precio medio de todos los módulos disponibles para cada programa, pero solo de las versiones Android, HTML5 e Ios ya que si se incluyen precios de módulos de exportación de consolas que suelen ser los más costosos, que solo algunos de los frameworks tienen, estaríamos desvirtuando el precio medio y penalizando a aquellos programas que posibilitan este tipo de exportación.

|  |  |
| --- | --- |
| Precio medio módulos de exportación. | |
| Gdevelop | 0€ |
| Construct 2 | 0€ |
| Click Team Fusion | 76.3€ |
| Game Maker | 83.32€ |
| TOTAL PRECIO MEDIO CONJUNTOS | 39.9€ |

Tabla 4: Precio medio módulos de exportación

Al igual que para el precio de las licencias, el precio de los módulos de exportación también cuenta con un baremo que determinará la puntuación del programa en este apartado.

|  |  |
| --- | --- |
| Baremo puntuación precio de módulos. | |
| Precio medio +/-35€ | 50% del total de la puntuación. |
| Precio medio hasta +40€ | 25% del total de la puntuación. |
| Precio medio -40€ | 100% del total de la puntuación. |
| Precio medio + de 80€ | 0% del total de la puntuación. |

Tabla 5: Baremo precio módulos de exportación

### Análisis Gdevelop.

#### Criterio 1: Requisitos e instalación.

Los requisitos para poder desarrollar en Gdevelop son bastante asequibles y la mayoría de equipos pueden trabajar con el programa ya que consume pocos recursos del sistema, como se puede observar en la siguiente tabla.

|  |  |
| --- | --- |
|  | REQUISITOS GDEVELOP |
| Sistema Operativo | Mínimo Windows XP SP3, soportados Windows Vista, 7, 8 y 10  Ubuntu  Otros Linux (Versiones no oficiales)  Mac (En fase beta) |
| Procesador | 200Mhz o mayor |
| Memoria | 512 MB de RAM |
| Tarjeta Gráfica | Mínimo Tarjeta Gráfica con soporte para Direct3D 9 |
| Otros | Conexión internet necesaria para la instalación, actualización y descarga del software. |

Tabla 6: Requisitos Gdevelop

Cuenta con la ventaja de que no solo se puede desarrollar en Windows sino que también tiene disponible un entorno de desarrollo en Ubuntu y en otras distribuciones de Linux como pueden ser Fedora[[9]](#footnote-9) o Arch [[10]](#footnote-10)(aunque éstas no son versiones oficiales han sido creadas por la comunidad) y una versión para Mac que por el momento se encuentra en estado beta y con una funcionalidad bastante limitada.

Gdevelop tiene una versión app-web, llamado **Gdevapp**, véase Figura 1, que, aunque de forma más limitada que la versión de escritorio, permite trabajar con el programa desde cualquier dispositivo conectado a Internet, aunque se manejan de forma independiente. Gracias a su sistema de importación y exportación, desde la App web podemos trabajar con proyectos generados por el programa de escritorio en la aplicación web y a la inversa.

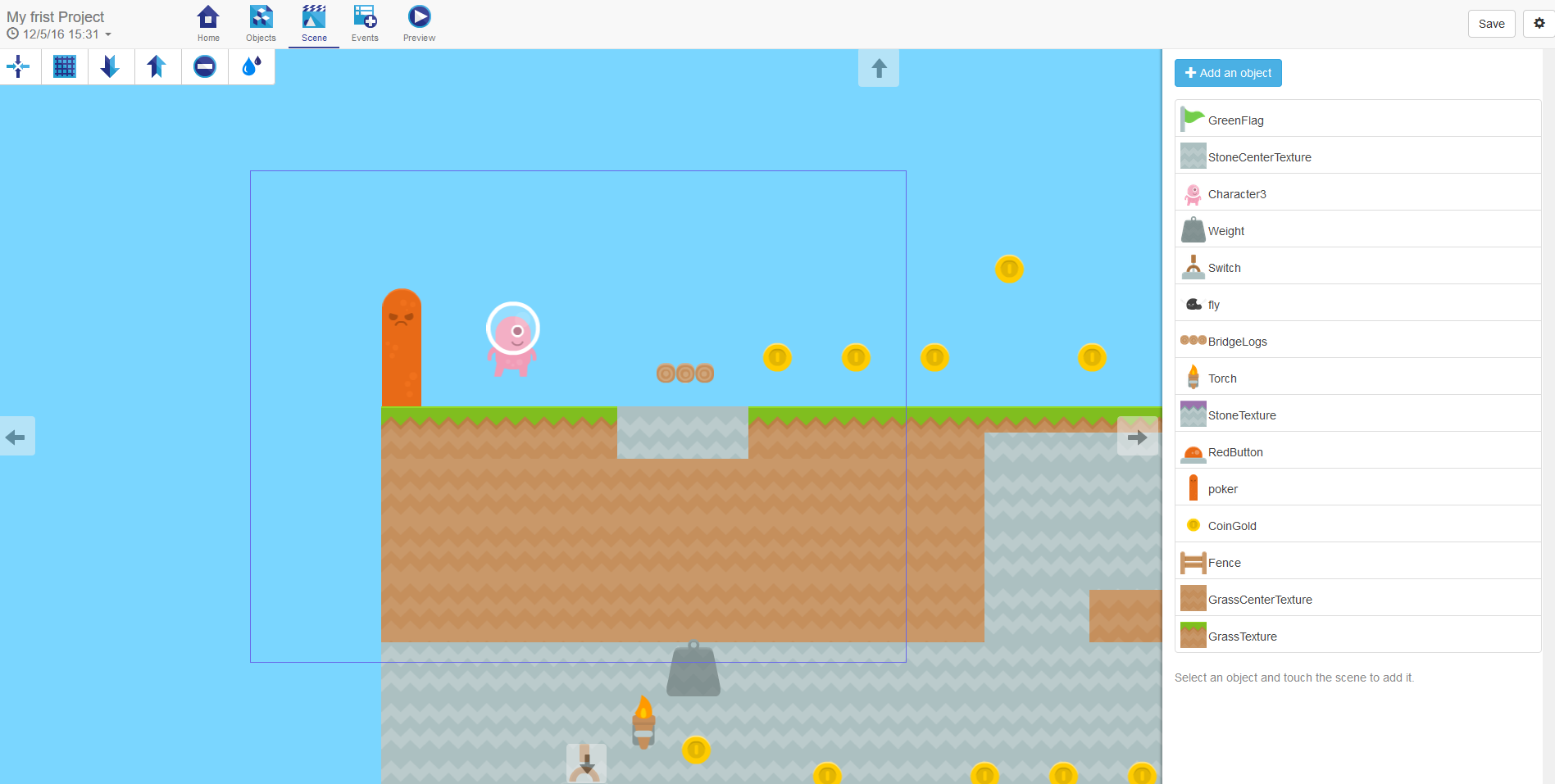


Figura 1: Gdevapp

##### Puntuación subcriterio requisitos

Como se ha comprobado los requisitos de funcionamiento de Gdevelop son asumibles por casi la totalidad de los usuarios y además no limita su uso a sistemas operativos Windows, teniendo alternativas tanto en sistema Ios como versión accesible desde cualquier sistema y dispositivo. Por estos motivos la puntuación en este apartado que muestra la tabla siguiente es la máxima posible.

----

|  |  |  |
| --- | --- | --- |
| CRITERIO | PUNTUACIÓN MAX-MIN | PUNTUACIÓN |
| Requisitos de instalación. | 0-4 | 4 |

Tabla 7: Gdevelop puntuación instalación

Una vez se ha comprobado que el equipo dónde se va a instalar GDevelop cumple los requisitos mencionados en la Tabla 1, el proceso de instalación es bastante sencillo y tiene los siguientes pasos:

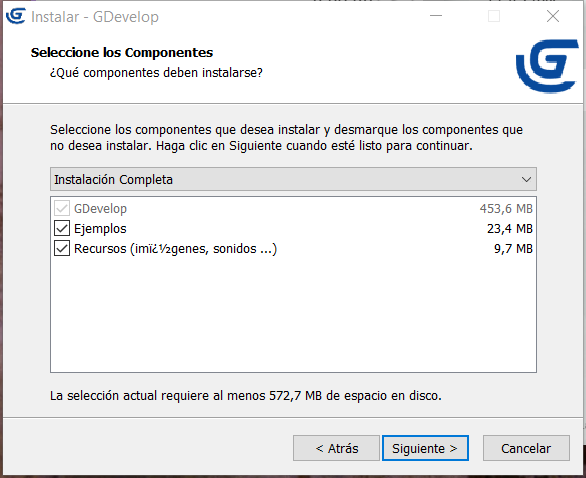
1. Descargar el paquete de instalación para el sistema en el que se vaya a desarrollar. Desde la web oficial (http://compilgames.net), la descarga tiene un peso de 109 Mb.
2. El paquete de instalación está en castellano y siguiendo las instrucciones que el programa indica se completa la instalación. Hay que indicar que en un momento del proceso se pregunta si se desea instalar no solo el programa, sino también junto a él toda una serie de ejemplos y recursos, como podrían ser imágenes y sprites, véase Figura 2.   
     
   

Figura 2: Gdevelop instalación de recursos accesorios

1. El proceso de instalación en el equipo de prueba ha tenido un tiempo total de ocho minutos y 26 segundos. La instalación del programa con los ejemplos y los recursos adicionales que se indican en el paso anterior tiene un peso total en el equipo de 572Mb.

##### Resumen proceso de instalación

La tabla siguiente muestra el resultado de la instalación y el resumen de los datos obtenidos en él equipo de pruebas.

|  |  |
| --- | --- |
| Gdevelop Instalación en el equipo de pruebas | |
| Tamaño de descarga | 107 Mb |
| Requiere programas adicionales | No |
| Instalación de módulos o recursos adicionales durante la instalación | Si |
| Tiempo total | 8 minutos 26 segundos. |
| Tamaño posterior a la instalación | 572 Mb |

Tabla 8: Resumen proceso de instalación Gdevelop

##### Datos de instalación encuesta.

En la siguiente tabla se muestra un resumen de resultados del proceso de instalación de los 8 participantes en el estudio vía encuestas realizado. Los datos completos se encuentran en el Anexo I: Resultados estudio de instalación e interfaz.

|  |  |
| --- | --- |
| Gdevelop datos estudio instalación | |
| Tiempo medio | 4 minutos 9 segundos |
| Valoración media sencillez del proceso de instalación (0 a 10) | 9.25 |
| Valoración media de la funcionalidad del proceso de instalación (0 a 10) | 9 |
| Puntuación media funcionalidad y sencillez | 9.125 |
| Fallos en instalación. | No |

Tabla 9: Gdevelop datos encuesta instalación

Como se puede comprobar en la tabla el tiempo de instalación es bastante pequeño, la valoración tanto de la funcionalidad y sencillez del proceso de instalación son muy altas y los sujetos del estudio no han tenido problemas ni fallos de instalación.

##### Puntuación subcriterio instalación.

Gdevelop ha tenido un tiempo de instalación menor de 5 minutos y su valoración media de funcionalidad y sencillez del proceso de instalación es mayor de 9, concretamente 9.125. Conforme a la ponderación definida en el marco teórico le corresponde 2 puntos en este apartado.

|  |  |  |
| --- | --- | --- |
| CRITERIO | PUNTUACIÓN MAX-MIN | PUNTUACIÓN |
| Instalación. | 0-2 | 2 |

Tabla 10: Gdevelop subcrierio instalación

##### Resumen puntuación requisitos e instalación.

En la siguiente tabla se resumen la puntuación en el criterio requisitos e instalación.

|  |  |  |
| --- | --- | --- |
| CRITERIO | PUNTUACIÓN MAX-MIN | PUNTUACIÓN |
| Requisitos de instalación. | 0-4 | 4 |
| Instalación | 0-2 | 2 |
| TOTAL | | 6 |

Tabla 11: Gdevelop criterio requisitos e instalación

#### Criterio 2: Documentación, tutoriales y comunidad.

Gdevelop cuenta con una wiki colaborativa, con la que se pueden aprender todos los aspectos del programa, de su instalación y puesta en funcionamiento así como de los distintos tipos de editores con los que cuenta y cómo funciona cada uno de ellos.

Esta wiki colaborativa esta fundamentalmente en inglés, aunque algunos artículos y apartados también se encuentran disponible en castellano y los responsables de la misma permiten que los usuarios colaboren traduciendo las partes de la misma por lo que el número de textos traducidos va en aumento.

La documentación está escrita de manera que usuarios de todos los niveles pueden entender y realizar la mayoría de las funcionalidades que en ella se explican.

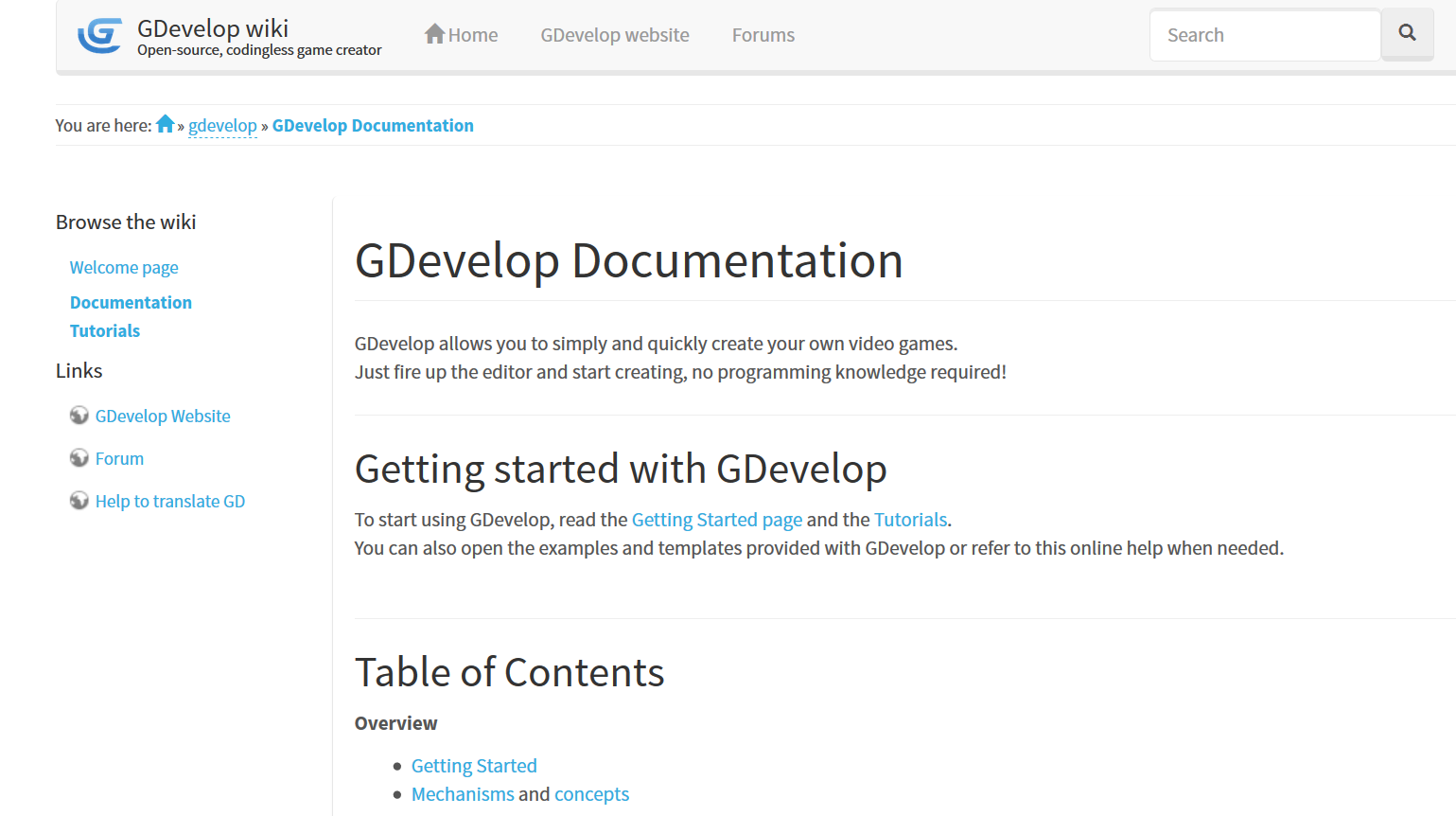


Figura 3: Portada de la wiki de Gdevelop

Gdevelop cuenta con una wiki colaborativa, con la que se pueden aprender todos los aspectos del programa, de su instalación y puesta en funcionamiento, así como de los distintos tipos de editores con los que cuenta y cómo funciona cada uno de ellos.

Esta wiki colaborativa esté desarrollada fundamentalmente en inglés, aunque algunos artículos y apartados también se encuentran disponible en castellano y los responsables de la misma permiten que los usuarios colaboren traduciendo distintas partes de la misma por lo que el número de textos traducidos va en aumento.

La documentación está escrita de manera que usuarios de todos los niveles pueden entender y realizar la mayoría de las funcionalidades que en ella se explican.

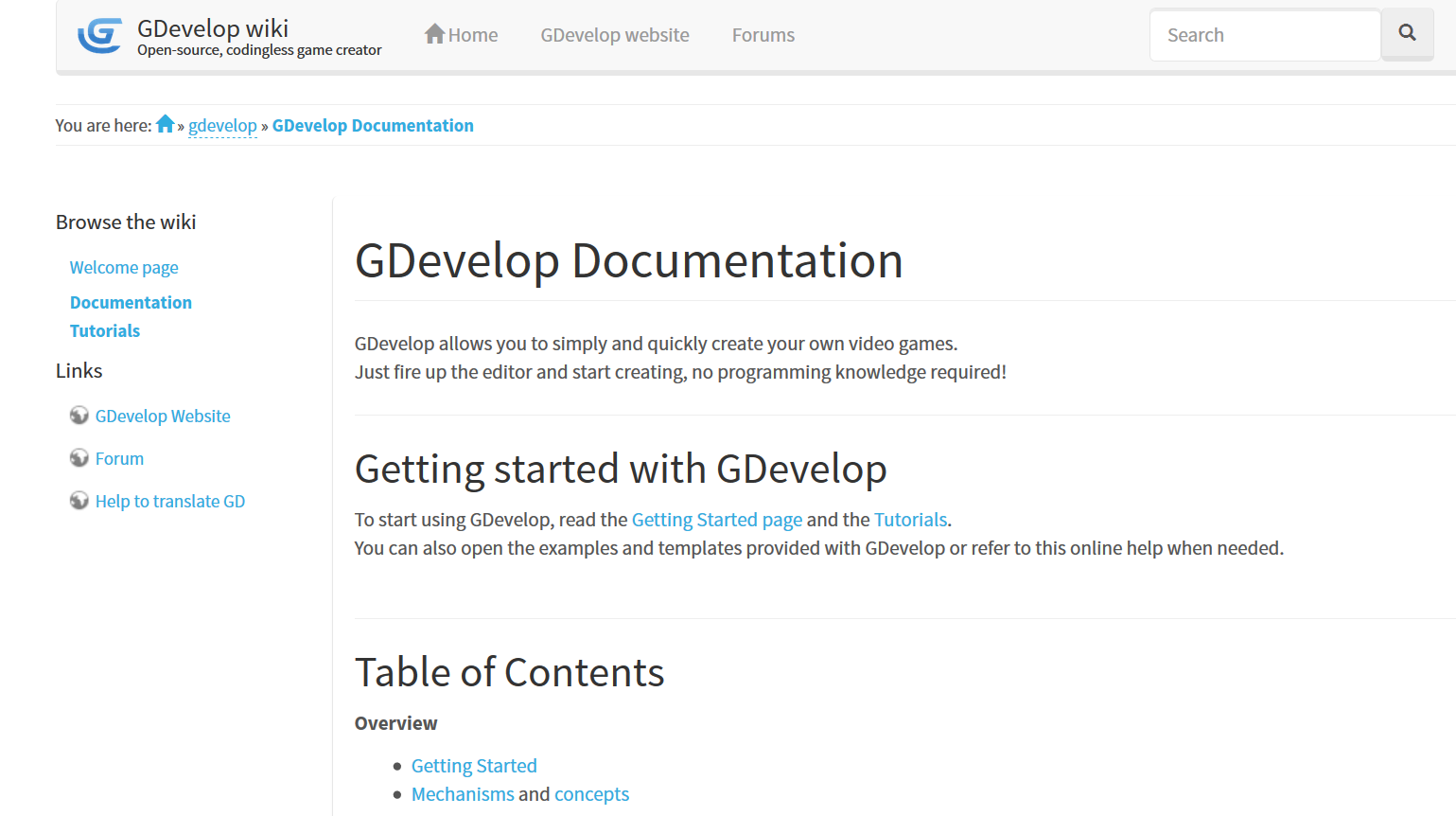


Figura 4: Portada de la wiki de Gdevelop

La wiki tiene un apartado específico dedicado a tutoriales. En el primero, recomendado para principiantes, se enseñen las funciones básicas del programa , el movimiento y control del personaje y la gestión básica del proyecto mediante la realización de un ejemplo básico de un juego de plataformas. Después de este primer tutorial que se recomienda hacer a usuarios principiantes, se encuentran una serie de tutoriales que explican determinadas funcionalidades y que están clasificados de la siguiente forma:ç

* **1. Para principiantes:** En este apartado se encuentran todos aquellos tutoriales que son necesarios para poder hacer un uso básico del programa. Entre ellos se encuentra el citado tutorial de plataformas, así como otros relativos a la exportación de proyectos y otros de elementos sencillos. Este apartado incluye 5 tutoriales.
* **2. Otros tutoriales:** Engloba funciones más avanzadas que expanden y completan a los tutoriales para principiantes. Por ejemplo, el uso de variables, la distribución de los juegos, creación de un menú de juego o dos tutoriales de juegos más avanzados para aprender más características del programa. Este apartado lo componen 10 tutoriales.
* **Consejos y pequeñas explicaciones:** Recomendaciones para trabajar correctamente y aprovechar al máximo las funciones del programa y una serie de pequeños dudas que suelen aparecer recurrentemente a los usuarios. Hay cuatro tutoriales disponibles en esta sección.
* **Tutoriales avanzados:** Enseñan el uso de las posibilidades superiores del programa. Son tutoriales para los que se necesita un gran conocimiento del programa y/o conocimiento de Javascript o C++, ya que en este aparto se explica el uso de estos lenguajes para crear eventos personalizados y nuevos comportamientos dentro de los proyectos. Encontramos 6 tutoriales en este apartado.
* **Tutoriales de la comunidad:** Los responsables de Gdvelop van recopilando los tutoriales que creen que pueden ayudar a los usuarios en el manejo del programa, los que consideran más completos o los que por razones de utilidad de lo explicado o por temática que ha realizado la comunidad en sus foros oficiales y les da visibilidad. Los tutoriales de la comunidad son 36.

La mayoría de tutoriales oficiales de Gdevelop contienen además los sprites, backgrounds y todos los recursos que van a hacer falta para seguir la explicación y completarlo, es un punto importante y que ayuda a reducir tiempo en esta fase de aprendizaje ya que no se debe dedicar tiempo a buscar estos elementos.

Los foros oficiales otro de los puntos fuertes de Gdevelop, su comunidad es bastante activa y se puede desde reportar bug detectados en el programa, realizar consultas sobre el programa, mostrar los juegos desarrollados, así como preguntar dudas respecto a la implementación de determinadas funciones. Los foros oficiales están disponibles en inglés y no existe versión en castellano por el momento.

##### Puntuación subcriterio documentación, foros y tutoriales oficiales.

Por todos los aspectos relativos a manuales, tutoriales, y foros que componen este criterio descritos anteriormente, la puntuación en este criterio es bastante alta, ya que el único punto negativo que se puede encontrar es la falta de más proyectos de ejemplo y videos explicativos.

|  |  |  |
| --- | --- | --- |
| CRITERIO | PUNTUACIÓN MAX-MIN | PUNTUACIÓN |
| Documentación, foros y tutoriales oficiales | 0-6 | 3.5 |

Tabla 12: Gdevelop puntuación Documentación, foros y tutoriales oficiales

##### Tutoriales oficiales realizados.

###### How to make a platformer game?

Es el primer tutorial que desde la página oficial se recomienda hacer para empezar a entender el funcionamiento general del programa. Incluso en su primer apartado explica el proceso de instalación de Gdevelop.

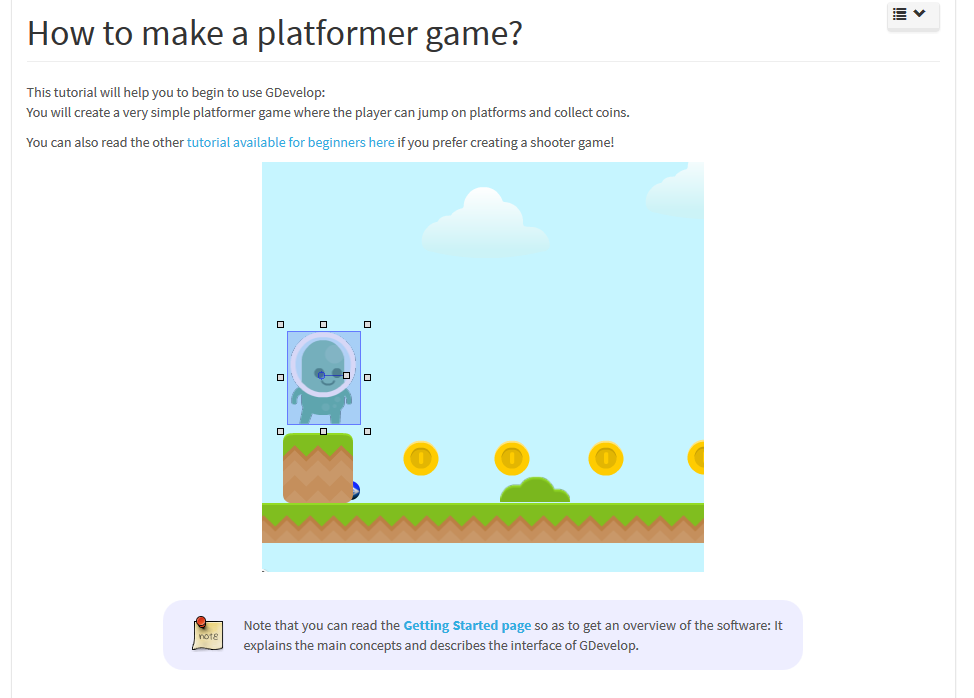
****

Figura 5:Portada del tutorial: How to make a platformer game?

En líneas generales el tutorial es muy completo, ofreciendo una guía muy detallada de todos los pasos que se deben ir realizando en el mismo. Destaca sobre todo en los siguientes aspectos:

1. Cuenta con numerosas imágenes que ayudan a visualizar todos los pasos que se deben realizar.
2. Como se ha comentado anteriormente tiene todos los recursos necesarios para completarlo (Sprites y elementos de la escena). En estas fases de aprendizaje donde lo importante que el usuario se familiarice con el entorno de desarrollo , es muy adecuado contar con los recursos, para maximizar el tiempo dedicado al aprendizaje y no a buscar recursos accesorios.
3. El apartado que incluye los eventos, está muy bien explicado, lo que permite hacerse una idea bastante buena de cómo funciona a nivel lógico.
4. Se facilita tanto el archivo fuente del proyecto por si en algún momento un usuario no sabe realizar alguna acción o quiere comparar su implementación con la del tutorial pueda acudir a este recurso como un ejemplo compilado en versión web del resultado del tutorial
5. El tutorial va escalando su dificultad y no repite conceptos. La primera vez que enseña al usuario una nueva función la explica detalladamente con diversas imágenes y textos explicativos, en las sucesivas veces que aparece esa misma acción el tutorial no la detalla en profundidad e indica que clase de acción es y en que parte del tutorial la hemos realizado. Todo esto ayuda al usuario a intentar realizar las cosas por sí mismo, progresar en su manejo del programa y hace el manual más ameno.

Pero también tiene una serie de inconvenientes de apartados que no están correctos o podría ser mejorables:

1. El tutorial está en inglés y pese a que a primera vista esto no supone ningún problema como para considerarlo un inconveniente o elemento que reduce la valoración general del tutorial, sí lo hace el hecho de que el programa se encuentre en castellano, con lo que la totalidad de las acciones que se deben llevar a cabo deben ser traducidas por el usuario o identificadas por los iconos.
2. Algunas imágenes del tutorial no son correctas. El tutorial consiste en la creación de un videojuego de plataformas, en el que en un apartado explica la creación de monedas en el escenario y su posterior colisión con el personaje, pero las imágenes que aparecen en este apartado no se corresponden con lo explicado.
3. Tiene partes desactualizadas. El tutorial parece realizado con una versión anterior del programa y hay apartados que ahora son distintas. En concreto se ha detectado este problema en los apartados que explican la colisión y como crear capas de texto.
4. El tutorial contiene la creación de un enemigo que se mueve por el escenario. Pero la creación de este movimiento está mal explicada y estructurada y es complicado entenderla correctamente.

###### Begginers Tutorial

Junto al anterior, este tutorial constituye la primera aproximación al programa y son los que se recomienda realizar en un primer momento para aprender las funciones básica y a manejarse con el mismo.

En este tutorial se explica cómo crear un juego de disparos donde el jugador tiene que destruir los tanques enemigos que se mueven hacia él apuntando y disparando con el ratón.

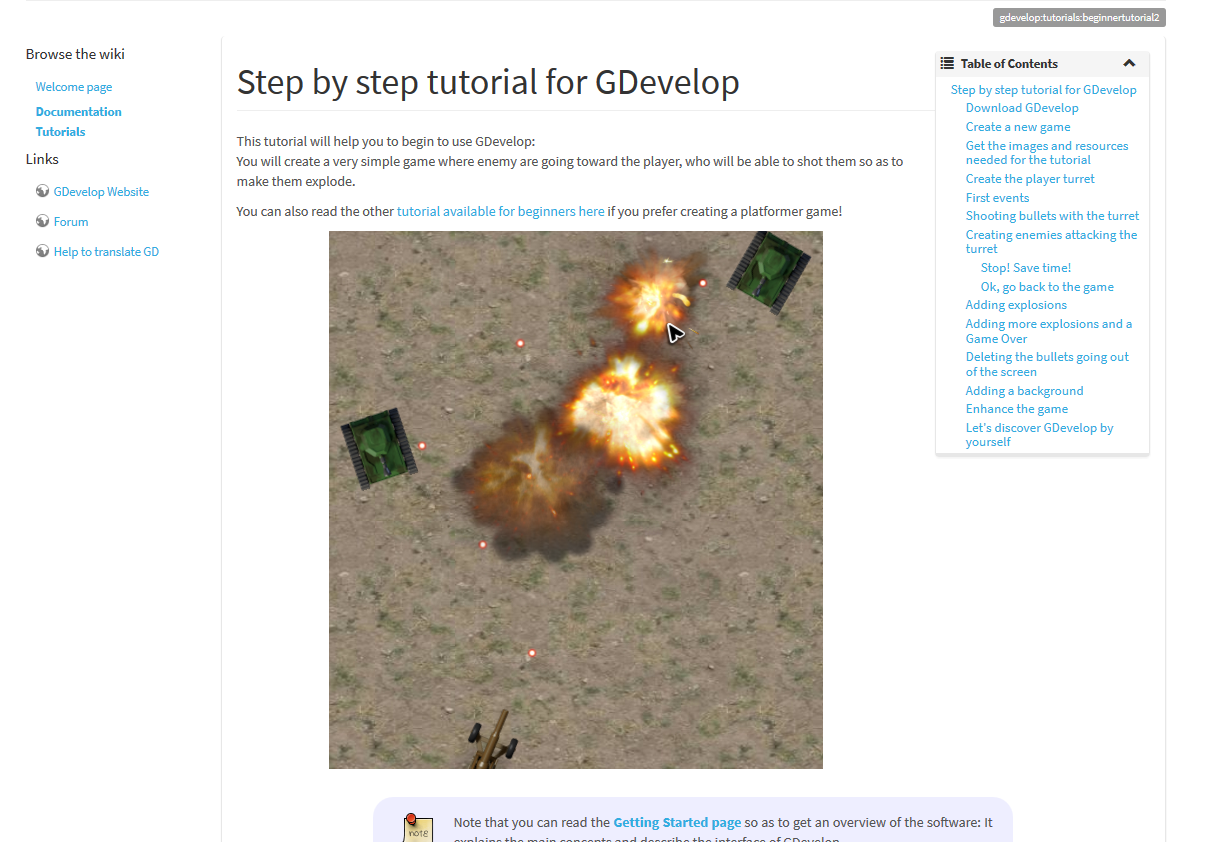


Figura 6: Tutorial Gdevelop Begginers

El tutorial comparte con anterior todos sus aspectos positivos, pero tiene los siguientes inconvenientes o errores:

1. Al igual que el anterior el tutorial esta en inglés y el programa en castellano, por lo que se ha comentado anteriormente esto produce problemas de coherencia.
2. La parte de la animación de la explosión no ésta bien explicada y si se implementa como indica el tutorial aparecerá en un lugar distinto al que debería aparecer, para corregirlo se pueden realizar dos acciones: ajustar restando a la posición de X e Y para situarla en el lugar correcto; o bien crear un nuevo punto en el objeto y que sea este el genere la colisión. En esta prueba se ha optado por la primera solución.

También se han detectado posibles mejoras y modificaciones del tutorial. Son las siguientes:

1. Toda la parte del fin de partida ha sido implementada de forma diferente a la explicada en el tutorial, ya que no era la forma correcta de realizarla en términos de gestión de recursos y un mejor flujo de juego.   
   En el manual, cuando un enemigo golpea a un jugador aparece el mensaje de Game Over sobre la pantalla y no sucede nada más y para volver a jugar hay que detener el juego y volver a ejecutarlo.

Para mejorar el proceso se ha creado otra escena a la que se cambia después del mensaje de “Game Over” y un botón que permite volver a iniciar el juego.

1. Se ha añadido una variable *puntuación* que se imprimirá en la parte superior de la pantalla y que se incrementará cada vez que un enemigo sea destruido.
2. Se ha añadido una serie de sub- eventos (véase Figura 7) que conforme la puntuación aumente también lo haga la velocidad de los enemigos, para aumentar la dificultad progresivamente.

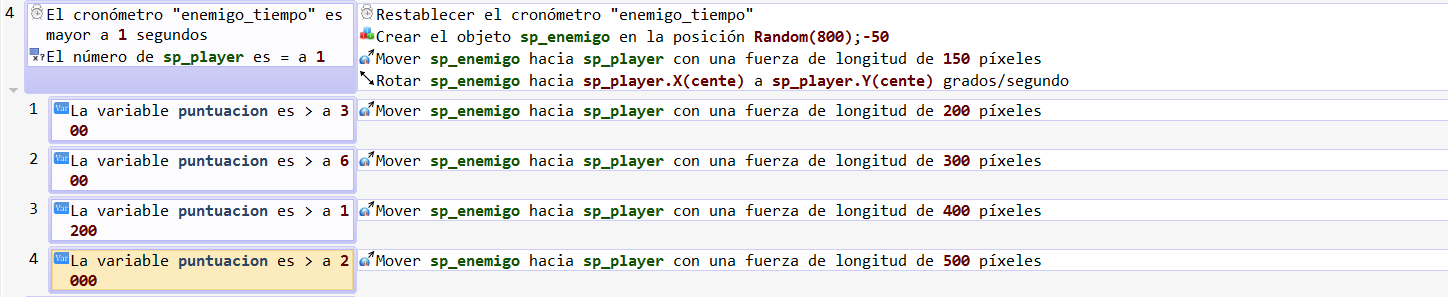


Figura 7: Eventos puntuación

1. La creación de las balas tiene una nueva condición y es que exista un jugador en la escena. Esto impide que después de destrucción del jugador (véase Figura 8) se sigan generando balas, como si ocurría con la implementación original.



Figura 8: Destrucción del jugador

##### Resumen de tutoriales realizados.

La tabla siguiente contiene el resumen de los tutoriales realizados , el tiempo invertido y las funciones aprendidas durante su realización.

|  |  |
| --- | --- |
|  |  |
| Tiempo dedicado. | 7 horas 4 minutos |
| Tutoriales realizados | 2 |
| Tutoriales completados | 2 |
| Juegos completos | 1 |
| Errores detectados en los tutoriales | Pocos |
| Modificaciones realizadas | Pocas y accesorias |
| Funcionalidades aprendidas en los tutoriales | Creación de proyectos.Añadir y gestionar Sprites.Añadir fondo a la escena.Movimiento en 2D.Salto.Movimiento en 3D.Creación de objetos y enemigos en posiciones aleatorias.Movimiento unidireccional de enemigo, con persecución y cambio de dirección.Rotar objetos con el ratón.Disparo de balas.Colisión entre objetos.Destrucción de objetos tras colisión. |

Figura 9: Gdevelop resumen tutoriales realizados

##### Puntuación subcriterio análisis de tutoriales oficiales.

Como se puede observar en la tabla del apartado anterior, el número de funcionalidades aprendidas es muy alto, los errores detectados son pocos y las modificaciones realizadas escasas y accesorias, solo algunos pequeños fallos de explicaciones , de correlación entre imágenes y explicación y coherencia impiden que tenga la máxima puntuación.

|  |  |  |
| --- | --- | --- |
| CRITERIO | PUNTUACIÓN MAX-MIN | PUNTUACIÓN |
| Análisis de tutoriales oficiales | 0-8 | 7 |

Tabla 13: Subcriterio Gdevelop Análisis de tutoriales oficiales

A continuación, se detalla la presencia de Gdevelop en Reedit y Taringa.

###### **Reddit**

La presencia de Gdevelop en Reddit es meramente anecdótica, como muestra la Figura 10, con tan solo 8 temas abiertos y el último mensaje en uno de ellos fue escrito hace más de 4 meses. Además, en todos de ellos los desarrolladores del programa remiten a los foros oficiales.

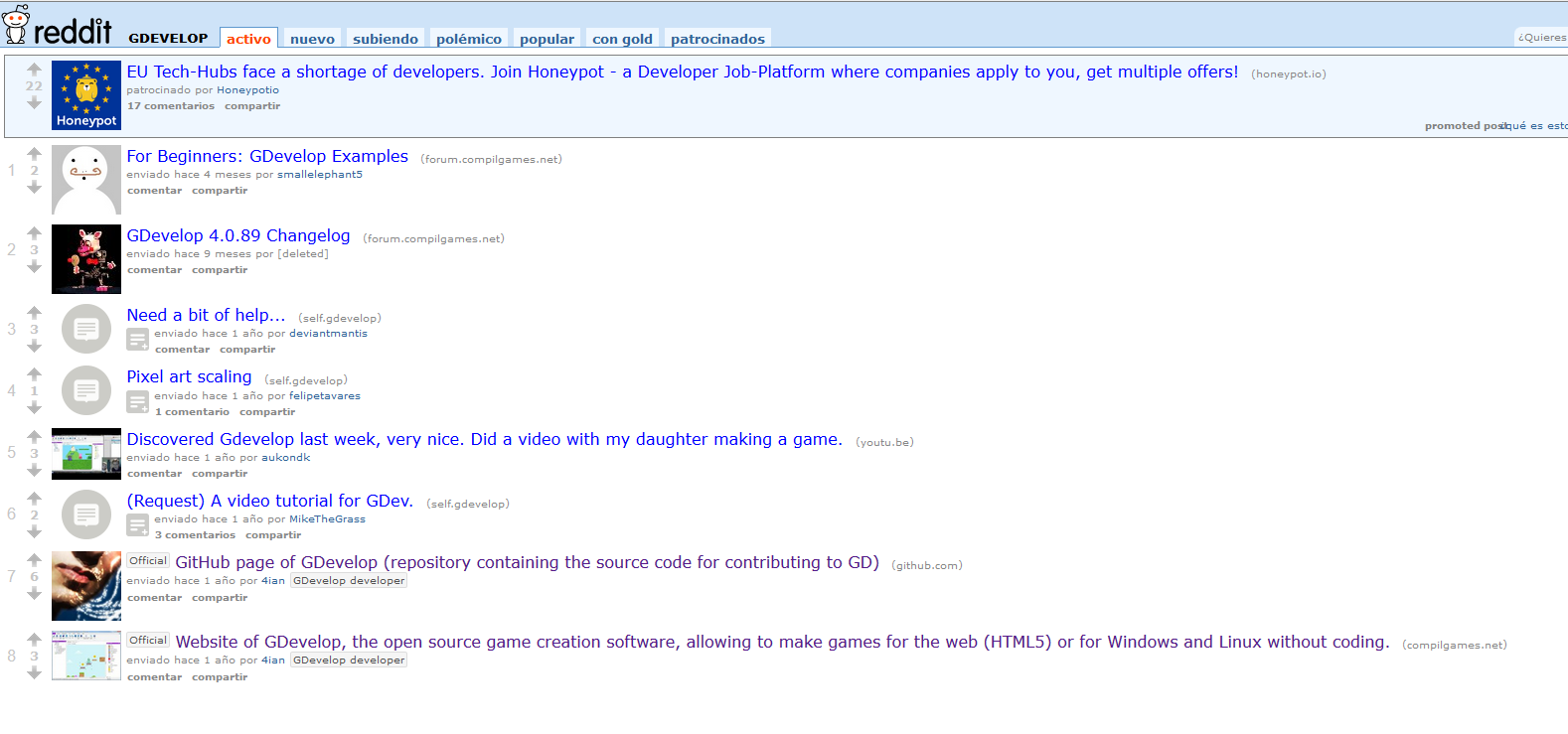


Figura 10: Gdevelop en Reedit

###### Taringa.

Existe una comunidad pero solo tiene dos miembros y un tema activo por lo que no se puede considerar como tal.

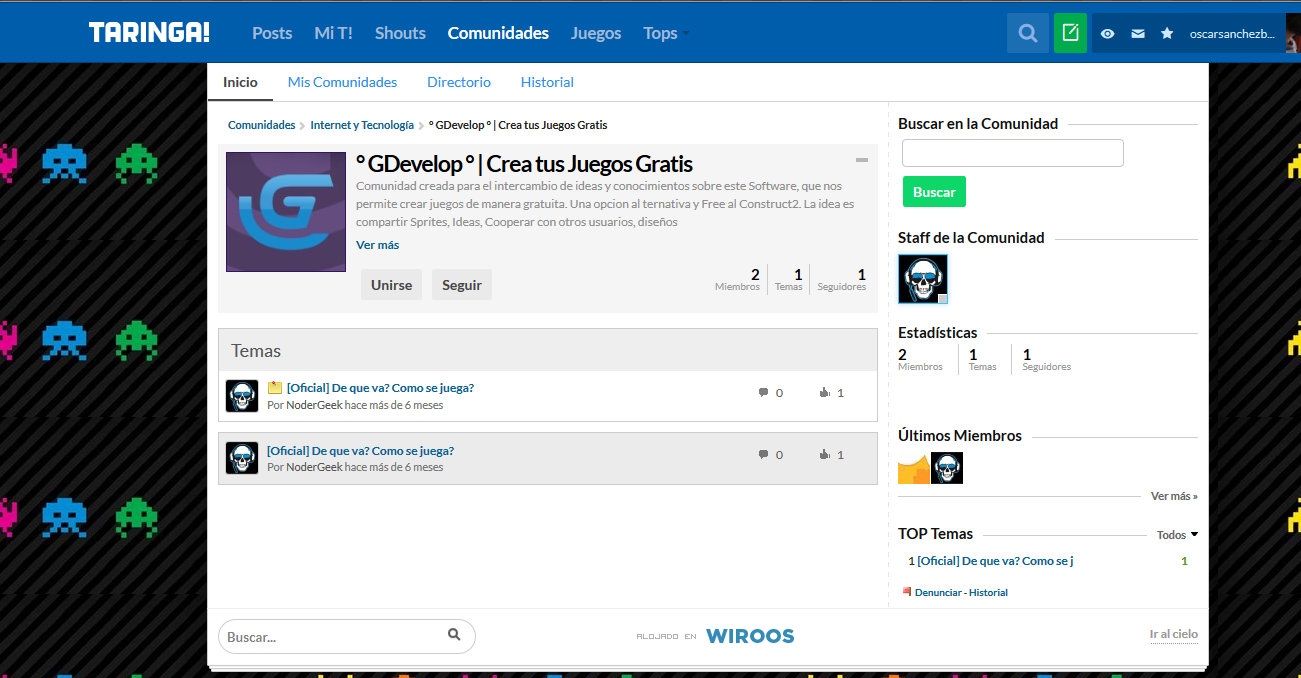


Figura 11: Gdevelop en Taringa

##### Puntuación subcriterio comunidades y foros de desarrolladores.

Como se ha podido comprar la presencia de Gdevelop en Reedit y Taringa es meramente anecdótica y además remite a los foros oficiales, por lo que la puntuación en este apartado es mínima.

|  |  |  |
| --- | --- | --- |
| CRITERIO | PUNTUACIÓN MAX-MIN | PUNTUACIÓN |
| Foros y comunidades de desarrolladores | 0-2 | 0.5 |

Tabla 14: Subcriterio foros y comunidades de desarrolladores

##### Resumen de puntuación Documentación, tutoriales y comunidad.

La tabla siguiente resume la puntuación del este criterio.

|  |  |  |
| --- | --- | --- |
| CRITERIO | PUNTUACIÓN MAX-MIN | PUNTUACIÓN |
| Documentación, foros y tutoriales oficiales | 0-4 | 3.5 |
| Análisis de tutoriales oficiales | 0-8 | 7 |
| Foros y comunidades de desarrolladores | 0-4 | 0.5 |
| TOTAL | | 11 |

Tabla 15: Gdevelop criterio documentación, tutoriales y comunidad

#### 2.3 Criterio 3: Funcionamiento y dificultad de uso.

Durante el desarrollo de los tutoriales no se han detectado fallos en el programa y ha funcionado en todo momento de forma fluida, tampoco se han reportado errores en este sentido por los participantes en las encuestas.

##### Puntuación subcriterio funcionamiento.

Por lo comentado anteriormente la puntuación en este apartado es la máxima posible, como muestra la siguiente tabla.

|  |  |  |
| --- | --- | --- |
| CRITERIO | PUNTUACIÓN MAX-MIN | PUNTUACIÓN |
| Funcionamiento General | 0-4 | 4 |

Tabla 16: Gdevelop subcriterio funcionamiento general

##### Puntuación primeros pasos: creación de un pequeño proyecto.

Gdevelop por su concepción, funcionamiento y por los distintos elementos que se comentarán a continuación hace que los primeros pasos con el programa sean bastante asequibles y sencillos para todo tipo de usuarios.

Un punto importante de Gdevelop que ayuda a conocer sus fundamentos básicos es que cuenta con una serie de plantillas o tipos de juego prestablecidos que aceleran la tarea de comprensión y aprendizaje. Estas plantillas se aplican al crear proyecto nuevo de forma que primero se pregunta la clase de juego a desarrollar. Así Gdevelop prepara todos los elementos necesarios para crear un juego de ese tipo además de crear una pequeña escena a modo de ejemplo,

Es un buen comienzo para los usuarios utilizar estas plantillas para conocer cómo funcionan los distintos elementos y eventos de Gdevelop, para una vez comprendidos poder utilizar el modo sin plantilla donde se crea todo desde cero.

Por lo visto en la fase de tutoriales en menos de 8 horas se ha podido completar la realización de un juego y se han sentado las bases para realizar otro por lo que el periodo que necesita un usuario para empezar a crear juegos es bastante corto.

Por todo lo anteriormente comentado la puntuación de en este apartado es la máxima posible.

|  |  |  |
| --- | --- | --- |
| CRITERIO | PUNTUACIÓN MAX-MIN | PUNTUACIÓN |
| Primeros pasos: Creación de un pequeño proyecto. | 0-4 | 4 |

Tabla 17: Gdevelop subcriterio funcionamiento general

##### Funciones básicas

A continuación se va a explicar cómo implementar las funciones básicas descritas en la definición de los criterios de selección y se va a puntuar en función de su dificultad.

###### Creación de un proyecto.

La creación de un proyecto nuevo es bastante sencilla, se puede observar en la Figura 11, mediante desde la interfaz con la siguiente sucesión de pasos:

|  |  |  |
| --- | --- | --- |
|  | |  |
| Ruta/comandos | Archivo->Creación de nuevo proyecto. | |
| Variables | **Plantilla del proyecto:** Para facilitar la creación de proyectos nuevos, Gdevelop permite que se use una de las plantillas disponibles que corresponden a distintos tipos de géneros o seleccionar proyecto vacío.  **Tipo de proyecto:** Existen dos tipos de proyectos, **nativo** (es decir para el sistema operativo en el que se ésta trabajando) y **HTML5** | |

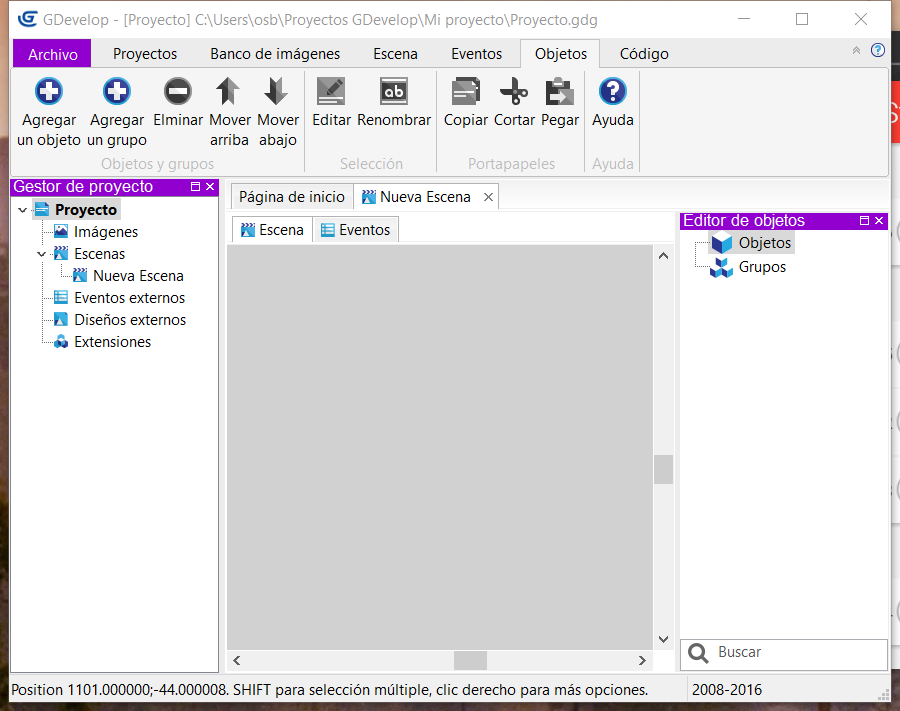


Figura 12: Gdevelop creación de un nuevo proyecto

La puntuación de este criterio es la siguiente:

|  |  |  |
| --- | --- | --- |
| Funcionalidad | Valoración | Puntuación |
| Creación proyecto. | Muy fácil | 10 |

###### Insertar fondo en la escena.

Para dotar a la escena de un fondo el proceso que hay que seguir es el siguiente:

1. Crear Objeto Sprite con la imagen de fondo.
2. Arrastrar el objeto desde la capa de editor de objetos a la escena.
3. Botón secundario sobre el objeto fondo que se acaba de arrastrar a la escena.
4. Se ajusta al tamaño a la escena, o a la parte de la misma donde se quiera que aparezca el fondo.
5. Cambiar el valor del Plano Z del objeto para que permanezca atrás del resto de los objetos.

|  |  |  |
| --- | --- | --- |
|  | |  |
| Ruta/comandos | Doble clic sobre el objeto en la escena-> Propiedades. | |
| Variables. | **Plano(Z):** Cambiar el valor Plano(Z) por el cero o un número negativo para que éste por debajo del resto de objetos**.** Este valor siempre tiene que ser menor que el Plano (Z) de todos los objetos si se quiere que se mantenga siempre como fondo. | |

**Tabla resumen funcionalidad.**

|  |  |  |
| --- | --- | --- |
| **Funcionalidad** | Valoración | Puntuación |
| Insertar fondo | Media | 5 |

Tabla 19: Gdevelop insertar fondo

###### Movimiento de personaje en dos direcciones.

Para dotar a un personaje controlable de movimiento en dos direcciones, como sería por ejemplo en un juego plataformas donde el personaje se mueve de izquierda a derecha, Gdevelop cuenta con tipos de comportamientos prestablecidos que facilitan enormemente la implementación de dicha funcionalidad.

En Gdevelop este tipo de movimiento se basa en dos elementos o patrones prestablecidos:

1. Objeto que se mueve sobre plataformas: Este es el patrón de comportamiento que debe tener el personaje que se desea controlar y realizar con el movimiento, con las variables de configuración que indica la Figura 12. Para dotar de dicho comportamiento a un objeto el proceso es el siguiente :

|  |  |  |
| --- | --- | --- |
|  | |  |
| Ruta/comandos | Doble clic sobre el objeto en la escena->Propiedades->Comportamiento->Agregar->Seleccionar “Objeto que se desplaza sobre plataformas”->Aceptar | |
| Variables | **Aceleración**: Velocidad con la que el personaje va empezar a moverse y a mantener hasta alcanzar su velocidad máxima.  **Can grab platform ledges : /investigar que es/**  **Controles por defecto:** Si se marca esta opción no se deberá especificar las teclas con las que se va controlar el personaje, de forma que el personaje se moverá con las fechas izquierda y derecha del teclado y con espacio se produce el salto.  **Desaceleración:** Velocidad de frenado del personaje.  **Grab offset on Y axis:   Grab tolerance on X asis:**  **Gravedad:** Fuerza con la que el personaje es atraído hacia abajo de la escena.  **Velocidad de caída máxima:** Tope de velocidad tras caída de plataforma que puede alcanzar.  **Velocidad de salto:** Rapidez con la que un personaje salta.  **Velocidad máx:** Velocidad que puede alcanzar el Sprite que se controla.  **Ángulo máx. de pendiente:** Ángulo a partir del cual el personaje comenzará a caer y a desplazarse. | |

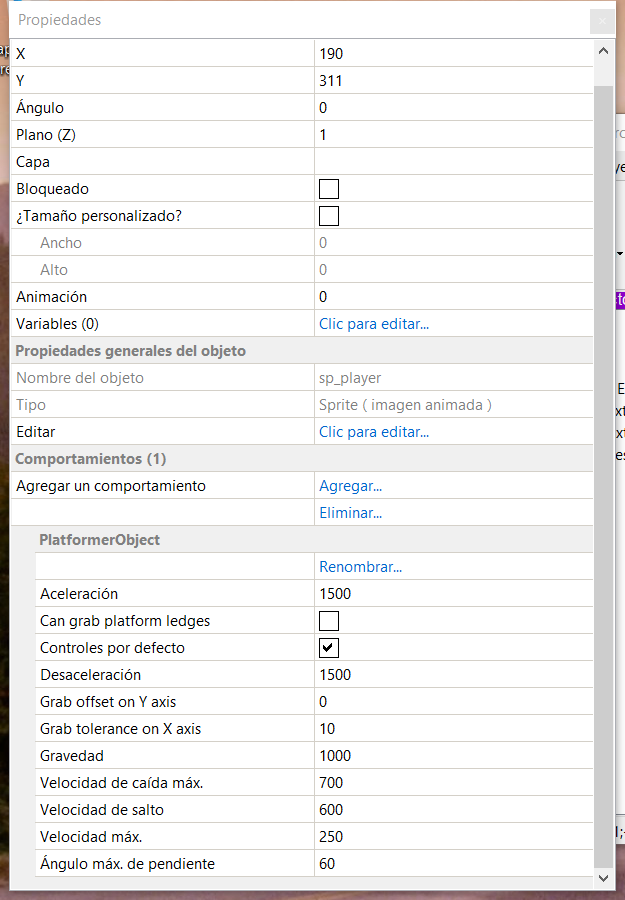
****

Figura 13: Gdevelop: Variables movimiento en 2D

1. Objeto plataforma: Pero si a un objeto se le indica que se trata de un *objeto que se desplaza sobre plataforma,* tiene que estar acompañado de otros que hagan la función de plataforma ya que si no caerían infinitamente hacia debajo de la escena.

|  |  |  |
| --- | --- | --- |
|  | |  |
| Ruta/comandos | Doble clic sobre el objeto en la escena->Propiedades->Comportamiento->Agregar ->Seleccionar “Plataforma”->Aceptar | |
| Variables | **MIRAR VARIABLES** | |

Una vez se tiene el *Objeto que se mueve sobre plataformas* y una o varias *plataformas* basta con colocar al Sprite sobre la plataforma en la escena.

**Tabla resumen funcionalidad.**

|  |  |  |
| --- | --- | --- |
| **Funcionalidad** | Valoración | Puntuación |
| Movimiento en dos dimensiones | Muy fácil | 10 |

###### Salto de personaje que se mueve en dos direcciones.

El salto del personaje en 2D mediante la pulsación de una tecla, en este caso espacio, es automática en Gdevelop si se trata de un objeto que se mueve por plataformas. Se puede variar su altura, y/o velocidad desde el editor de comportamiento de *objeto que se mueve por plataformas* e incluso desactivar o anular dicha posibilidad.

**Tabla resumen funcionalidad.**

|  |  |  |
| --- | --- | --- |
| **Funcionalidad** | Valoración | Puntuación |
| Salto. | Muy fácil | 10 |

###### Movimiento personaje en 8 direcciones.

El movimiento en 8 direcciones, es decir, izquierda, derecha, arriba, abajo y sus respectivas diagonales se hace de forma muy parecida a la implementación del movimiento en dos direcciones pero seleccionando en este caso el patrón de comportamiento “movimiento en 4-8 direcciones”.

|  |  |  |
| --- | --- | --- |
|  | |  |
| Ruta/comandos | Doble clic sobre el objeto en la escena->Propiedades->Comportamiento->Agregar->Seleccionar “Movimiento de arriba abajo (en 4-8 direcciones)->Aceptar | |
| Variables | **Aceleración**: Velocidad con la que el personaje va empezar a moverse y va mantener hasta alcanzar su velocidad máxima.  **Controles por defecto:** Si se marca esta opción no se deberá especificar las teclas con las que se va controlar el personaje, en esta opción el personaje se mueve con las fechas izquierda y derecha del teclado y con espacio se produce el salto.  **Desaceleración:** Velocidad de frenado del personaje.  **Desplazamiento angular:   Objeto a rotar:   Permitir diagonal:** Define si el personaje se va a poder mover en diagonal o solo en las 4 direcciones principales.   **Velocidad de rotación:  Velocidad máx:** Velocidad que puede alcanzar el Sprite que se controla. | |

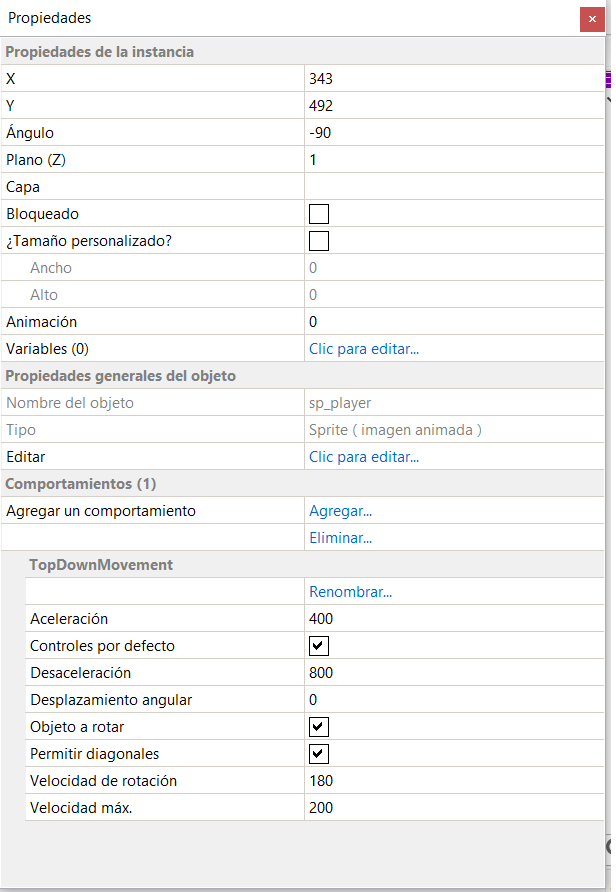
****

Figura 14: Gdevelop: variables movimiento en 8 direcciones

**Tabla resumen funcionalidad.**

|  |  |  |
| --- | --- | --- |
| **Funcionalidad** | Valoración | Puntuación |
| Movimiento 8 direcciones | Muy fácil | 10 |

Tabla 20: Gdevelop: movimiento 8 direcciones

###### Disparo de bala.

Para que el personaje controlado pueda realizar disparo de balase debe seguir el siguiente proceso:

1. En primer lugar y antes de crear toda la lógica que va permitir la creación de balas y su movimiento en una dirección, hay que crear en el Sprite un punto adicional, los que tiene por defecto (centro y origen). Así podemos precisar el lugar exacto donde se quiere que las balas se creen, por ejemplo en el caño si se trata de un tanque o en los laterales si se trata de un avión.

|  |  |  |
| --- | --- | --- |
|  | |  |
| Ruta/comandos | Ratón sobre el Sprite origen->Botón secundario->Editar ->En el menú visual se hace clic sobre la cruz "Editar los puntos de la imagen" ->Crear nuevo punto-> Hacer clic en la imagen y situar el punto en el lugar se quiere que sea origen de la bala ->renombrarlo para poder usarlo después. | |

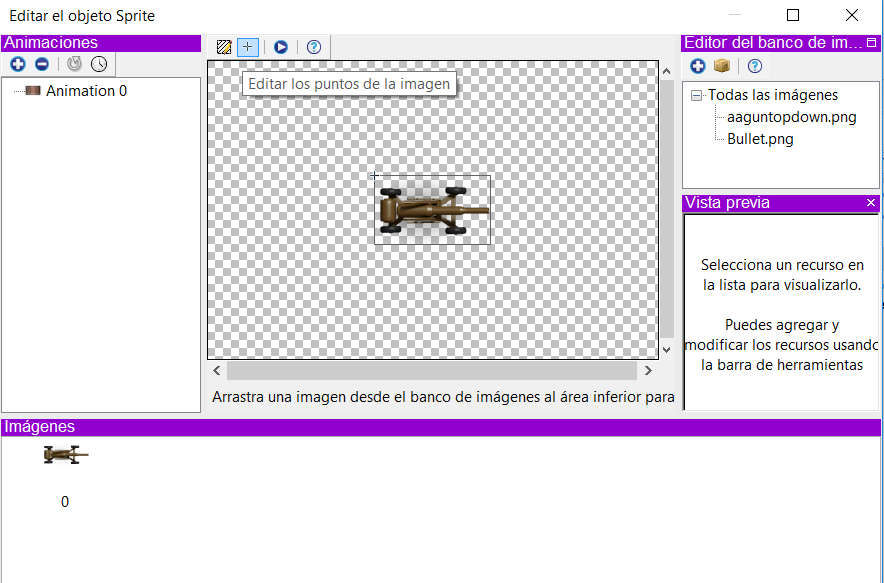


Figura 15: Gdevelop puntos de imagen

1. Creación del evento que se encargará de la creación de la bala y toda la lógica asociada. Un evento en Gdevelop consta de dos partes: la primera son **condiciones,** donde se van a especificar aquellas acciones, situaciones o estados de la escena que van a provocar que el evento comience a ejecutar, y la segunda las **acciones,** aquellos sucesos que se van a producir en la escena una vez que las condiciones especificadas suceden.

|  |  |  |
| --- | --- | --- |
|  | |  |
| Ruta/comandos | Eventos->Creación de evento. | |

1. Se agrega la condición para que al pulsar una tecla determinada él evento de creación de la bala comience.

|  |  |  |
| --- | --- | --- |
|  | |  |
| Ruta/comandos | Condición ->Ratón y Táctil ->Botón de ratón presionado o toque | |
| Variables | **Botón a evaluar:** Botón que va desencadenar el evento. | |

1. Para complementar la condición anterior, se tiene que añadir una condición que evite que se creen infinitas balas y prácticamente al mismo tiempo, introduciendo una restricción temporal para que las balas aparezcan en intervalos de tiempo. Para ello se agrega al evento la siguiente condición.

|  |  |  |
| --- | --- | --- |
|  | |  |
| Ruta/comandos | Condición-> Cronómetro y tiempo -> Valor de un cronómetro. | |
| Variables | **Tiempo en segundos:** El evento se va poder producir cuando el cronometro haya superado el tiempo introducido en este campo  **Nombre del cronómetro:** Dar un nombre al cronómetro es necesario para gestionar la creación de las balas ya que una vez que este supere el valor en la variable anterior la condición se cumplirá siempre. Pero con este nombre se pueden crear modificaciones en los pasos siguientes si se necesita reiniciarlo. | |

1. Ahora hay que crear el disparo de la bala que se ejecutará una vez se cumplan las dos condiciones creadas en los pasos anteriores. Es aconsejable que dicha opción se encuentre siempre en la última posición. Antes de realizar el disparo lo primero que habrá que hacer es restablecer el cronómetro.

|  |  |  |
| --- | --- | --- |
|  | |  |
| Ruta/comandos | Acciones ->Cronómetro y tiempo->Restablecer un cronómetro. | |
| Variables | **Nombre del cronómetro:** En este campo se introduciría el nombre dado en el paso anterior. | |

1. Ahora que ya se tienen las condiciones hay que crear la bala.

|  |  |  |
| --- | --- | --- |
|  | |  |
| Ruta/comandos | Acciones->Todos los objetos -> Objetos -> Crear un objeto | |
| Variables | **Objeto:** El objeto que se va crear, en este caso la bala.  **Posición X:** Lugar en el ancho de pantalla donde se va crear la bala. En este campo se debe introducir el objeto generador y el punto creado en la instrucción uno, de la siguiente forma: **-**nombre\_del\_objeto.X(punto\_creado\_instrucción\_1); **Posición Y:** Lo mismo pero para seleccionar a que altura aparece la bala.  nombre\_del\_objeto.Y(punto\_creado\_instrucción\_1); | |

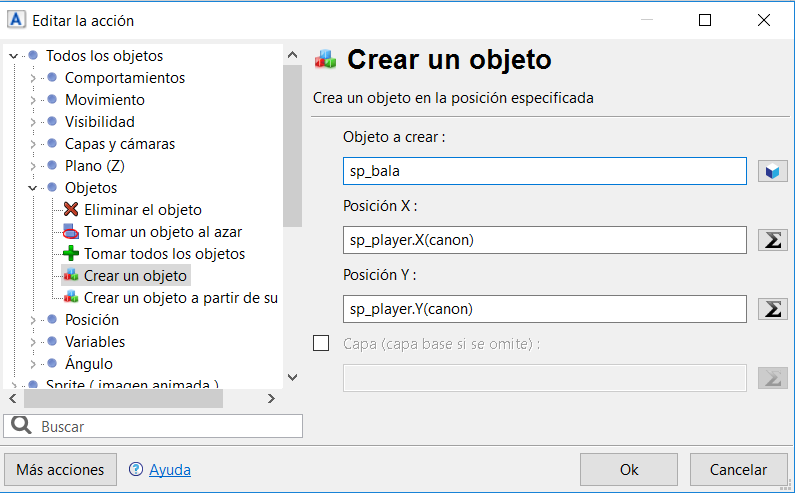


Figura 16: Gdevelop: creación de bala

1. En la instrucción anterior se ha creado la bala. En este pasose le va dar una velocidad de desplazamiento y dirección. Introduciendo la acción a continuación de la anterior.

|  |  |  |
| --- | --- | --- |
|  | |  |
| Ruta/comandos | Acciones->todos los objetos -> movimiento -> agregar una fuerza (ángulo) | |
| Variables | **Objeto:** Al que se quiere dar movimiento, en este caso la bala.  **Ángulo:** Hacia qué dirección se va a mover el objeto desde su creación.   **Velocidad**: Se indica la velocidad en pixeles del objeto.  **Amortiguamiento**: Simula la resistencia que posee el escenario a la a la bala. El valor 0 significa resistencia total, y cuanto mayor sea la cantidad menor será la resistencia que tendrá. Es un factor multiplicador de la velocidad. | |

1. Para una mejor gestión de la memoria y del juego en general se van a destruir las balas una vez que se encuentren fuera de la pantalla.

|  |  |  |
| --- | --- | --- |
|  | |  |
| Ruta/comandos | Ratón sobre el objeto bala->clic derecho del ratón ->otras propiedades->agregar comportamiento ->seleccionar destruir fuera de la pantalla | |

**Tabla resumen funcionalidad.**

|  |  |  |
| --- | --- | --- |
| **Funcionalidad** | Valoración | Puntuación |
| Disparo | Difícil | 2.5 |

###### Colisión con objeto.

|  |  |  |
| --- | --- | --- |
|  | |  |
| Ruta/comandos | Condición->Sprite -> Colisión ->Colisión(Por píxel/ Pixel Perfect). | |
| Variables | **Objeto 1:** Primer Sprite de la colisión.  **Objeto 2:** Segundo Sprite de la colisión. | |

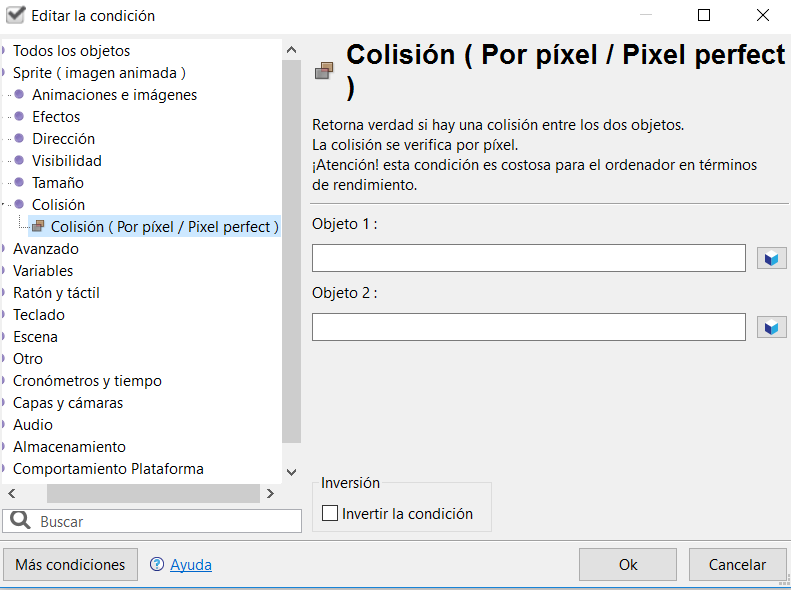


Figura 17: Gdevelop colisión por píxel

**Tabla resumen funcionalidad.**

|  |  |  |
| --- | --- | --- |
| **Funcionalidad** | Valoración | Puntuación |
| Colisión | Muy fácil | 10 |

###### Destrucción de objeto tras colisión.

1. Crear el evento con la condición colisión como se ha explicado en el punto anterior. Los objetos de la colisión serán por ejemplo Objeto 1, y Objeto 2.
2. Añadir la acción en el evento que va a destruir uno de los objetos implicados en la condición anterior.

|  |  |  |
| --- | --- | --- |
|  | |  |
| Ruta/comandos | Acción->Todos los objetos->Objetos->Eliminar el objeto | |
| Variables | **Objeto:** El objeto que se quiere eliminar. | |

|  |  |  |
| --- | --- | --- |
| Funcionalidad | Valoración | Puntuación |
| Destrucción tras colisión | Muy fácil | 10 |

###### Resumen de funciones básicas y puntuación del subcriterio.

La siguiente tabla contiene todas las funcionalidades y la dificultad y puntuación de cada una de ellas.

|  |  |  |
| --- | --- | --- |
| Función | Dificultad | Puntuación |
| Creación de un proyecto | Fácil | 10 |
| Insertar fondo a una escena. | Media. | 5 |
| Movimiento del personaje en 2 direcciones | Fácil | 10 |
| Salto de personaje en 2 direcciones | Muy fácil | 10 |
| Movimiento de personaje en 8 direcciones | Muy fácil | 10 |
| Disparo de bala. | Difícil | 2.5 |
| Colisión entre objetos. | Muy Fácil. | 10 |
| Destrucción de Objeto tras colisión. | Muy Fácil. | 10 |
| PUNTUACIÓN FINAL | | 67.5 |
| FACTOR DE CORRECIÓN | | 1 |
| PUNTUACIÓN CON FACTOR DE CORRECIÓN | | 67.5 |

Tabla 21: Gdevelop resumen de funcionalidades

##### Resumen subcriterio funcionalidades.

En la tabla siguiente se expone la puntuación escalada sobre 12 obtenida por Gdevelop en las funciones básicas.

|  |  |  |
| --- | --- | --- |
| **CRITERIO** | PUNTUACIÓN MAX-MIN | PUNTUACIÓN |
| Funciones básicas | 0-12 | 10.15 |

Tabla 22: Gdevelop puntuación funciones básicas

##### Puntuación subcriterio lógica de programación necesaria

Durante el desarrollo de los tutoriales, ejemplos y pequeños proyectos, se han detectado algunos elementos que hacen referencia directa a la programación tradicional si bien el programa cuenta con ayudas visuales para facilitar que sea el usuario quien escriba estas funciones de programación. Los elementos detectados son los siguientes:

1. Para generar números aleatorios (por ejemplo para la aparición en un lugar sin predeterminar del mapa de enemigos) se tiene que introducir la función Random (número) que da un número dentro del rango introducido.
2. Para mostrar números por pantalla, estos deben ser convertidos a cadena de texto y para ello Se debe usar la función ToString que está presente en la mayoría de lenguajes de programación.

Estas son las dos únicas referencias a los lenguajes de programación que se han encontrado. Así la puntuación en este apartado es alta ya que no son funciones muy difíciles de aprender por un usuario y ni requieren un conocimiento muy amplio de programación.

|  |  |  |
| --- | --- | --- |
| **CRITERIO** | PUNTUACIÓN MAX-MIN | PUNTUACIÓN |
| Lógica de programación necesaria | 0-4 | 3 |

Tabla 23: Gdevelop: subcriterio lógica de programación

##### Resumen puntuación funcionamiento y dificultad de uso

La siguiente tabla contiene las puntuaciones de todos los apartados que componen el criterio puntuación y dificultad de uso.

|  |  |  |
| --- | --- | --- |
| **CRITERIO** | IMPORTANCIA | PUNTUACIÓN MAX-MIN |
| Funcionamiento General | 0-4 | 4 |
| Primeros Pasos: Creación de un pequeño proyecto. | 0-4 | 3 |
| Funciones básicas | 0-12 | 10.15 |
| Lógica de programación necesaria | 0-4 | 3 |
| TOTAL | | 20.15 |

Tabla 24: Gdevelop criterio funcionamiento y facilidad de uso

#### Criterio 4: Editor Gráfico/Visual.

En la siguiente tabla se especifican las puntuaciones que ha obtenido Gdevelop en los distintos apartados de la encuesta dedicada a valorar la interfaz gráfica y visual.

|  |  |  |  |
| --- | --- | --- | --- |
| Concepto | Puntuación Máxima | Puntuación minima | Media (Sobre 5) |
| Rapidez al inciarse | 5 | 3 | 4,125 |
| El programa se muestra rápido | 5 | 3 | 4,25 |
| La interfaz es atractiva | 5 | 2 | 3,375 |
| Tiene un aspecto cuidado y moderno | 5 | 2 | 3,625 |
| Puntuación diseño del programa | 5 | 2 | 3,625 |
| Los iconos representa bien los elementos Y/o funciones | 5 | 3 | 4,25 |
| A priori parece sencillo de usar | 5 | 2 | 3,625 |
| Consumo de recursos | 5 | 3 | 3,75 |
| MEDIA TOTAL | | 3,85/5 | |
| PUNTUACIÓN PONDERADA | | 6.15/8 | |

Tabla 25: Resultado encuestas sobre interfaz

Una vez realizada la ponderación de la puntuación para adaptarla a la importancia del criterio la puntuación es la que muestra la siguiente tabla.

|  |  |  |
| --- | --- | --- |
| **CRITERIO** | IMPORTANCIA | PUNTUACIÓN MAX-MIN |
| Editor Gráfico/Visual | Clave | 6.15 |

Tabla 26: Criterio: Editor gráfico/visual

#### Criterio 5: Escalabilidad.

##### Puntuación subcriterio posibilidad de creación de funciones, scripts y/o lenguaje de programación propio.

Gdevelop no dispone de un lenguaje de programación propia y no provee, como si hacen otros frameworks, de un sistema para introducir código dentro del programa. La única posibilidad de aumentar las funcionalidades base de Gdevelop consiste en la modificación directa de los archivos de código para modificar su funcionamiento o crear copias de los plugins existentes para crear los propios. Es una tarea compleja y con ninguna documentación al respecto con lo que la mayoría de usuarios no sabrán realizar esta tarea.

Por lo expuesto anteriormente las posibilidades de creación de funciones y scripts propios son bastante limitadas y complejas por los que como indica la tabla siguiente la puntuación de este apartado es muy baja.

|  |  |  |
| --- | --- | --- |
| **CRITERIO** | PUNTUACIÓN MAX-MIN | PUNTUACIÓN |
| Posibilidad de crear funciones y scripts propios y/o lenguaje de programación propio. | 0-8 | 1 |

Tabla 27: Gdevelop: Subcriterio Posibiidad de creación de Scripts propios

La distribución de Gdevelop incluye una de serie algunas extensiones que incluyen funciones tales como un sistema de partículas, incluir Box3D en juego o un motor de red. Pero fuera de estas extensiones Gdevelop no dispone de add-ons oficiales que puedan extender las funcionalidades que el programa tiene de serie.

Al ser de código abierto y posibilitar que usuarios modifiquen, añadan o mejoren las funciones que el programa trae de serie por lo que existen algunos plug-ins no oficiales desarrollados por la comunidad, pero por la dificultad instalación para el usuario, la no garantía de que funcionen correctamente con la versión se tiene instalada del programa y las incompatibilidades entre los mismos, hacen que no se incluyan en esta sección.

La puntuación en este apartado por lo que acabamos de ver es cero ya que no se encuentran plug-ins sencillos de instalar, confiables y compatibles entre ellos.

|  |  |  |
| --- | --- | --- |
| **CRITERIO** | PUNTUACIÓN MAX-MIN | PUNTUACIÓN |
| Add-ons y plugins | 0-6 | 0 |

Tabla 28: Gdevelop: Subcriterio Add-ons y plugins

##### Resumen puntuación Escalabilidad,.

La tabla siguiente muestra el resumen de la puntuación del criterio escalabilidad.

|  |  |  |
| --- | --- | --- |
| **CRITERIO** | IMPORTANCIA | PUNTUACIÓN MAX-MIN |
| Posibilidad de crear funciones y scripts propios y/o lenguaje de programación propio. | 0-8 | 1.5 |
| Add-ons y plug-ins | 0-6 | 0 |
| TOTAL | |  |

Tabla 29: Gdevelop: puntuación criterio escalabilidad

#### Criterio 6: Resultados profesionales.

El carácter de software libre y abierto de Gdevelop hace que sea muy complicado determinar que juegos comerciales se han realizado con Gdevelop. Derivado de de este carácter de software libre no es necesaria ningún tipo de atribución. Además el tipo de licencia permite la modificación total del programa, por lo que los desarrolladores que trabajan con Gdevelop lo hacen con sus propias versiones modificadas y adaptas a sus desarrollos por lo que es prácticamente imposible determinar que juego comerciales han sido realizados con Gdevelop.

Aún si se consiguiera una lista de juegos creados con Gdevelop nos encontraríamos con una nueva dificultad todavía mayor. Ya que habría que examinar caso por caso las dimensiones de las modificaciones para determinar si se puede considerar todavía Gdevelop o estamos ante una versión tan customizada que ya estamos en presencia de un motor nuevo basado en Gdevelop .

Por lo comentado anteriormente solo se pueden tener en cuenta los juegos que la propia web de Gdevelop indica que han sido desarrollados con el programa. Respecto a estos están lejos de tener un resultado profesional y se tratan de pequeños juegos de navegador y que no se pueden catalogar de grandes proyectos, por lo tanto, se puede concluir que con el programa original sin modificaciones no existen juegos comerciales.

Visualmente los juegos se alejan mucho de ser excesivamente atractivos y no resultan potentes ni gráfica ni artísticamente, quedando en ese aspecto bastante limitados y no profesional.

##### Puntuación subcriterio principales juegos desarrollados

Como se ha comentando anteriormente es prácticamente imposible determinar que juegos comerciales han sido desarrollados con Gdevelop y los que indica en la web oficial del programa que han sido creados usando el programa están lejos de ser juegos verdaderamente profesionales y comercialmente atractivos, con lo que la puntuación en este apartado es la mínima posible.

|  |  |  |
| --- | --- | --- |
| **CRITERIO** | PUNTUACIÓN MAX-MIN | PUNTUACIÓN |
| Principales juegos desarrollados | 0-8 | 0 |

Tabla 30: Gdevelop: subcriterio Principales juegos desarrollados

##### Puntuación subcriterio diferenciación

Por el contrario, pese a que visualmente los resultados no son buenos y no son juegos comercialmente potentes, si hay cierta diferenciación entre ellos y no se detectan patrones visuales comunes.

|  |  |  |
| --- | --- | --- |
| **CRITERIO** | PUNTUACIÓN MAX-MIN | PUNTUACIÓN |
| Principales juegos desarrollados | 0-4 | 2 |

Tabla 31: Gdevelop subcriterio principales juegos desarrollados

##### Resumen puntuación resultados profesionales.

En la siguiente tabla se muestra el resumen de la puntuación el criterio resultados profesionales.

|  |  |  |
| --- | --- | --- |
| CRITERIO | IMPORTANCIA | PUNTUACIÓN MAX-MIN |
| Principales juegos desarrollados. | 0-8 | 0 |
| Diferenciación | 0-4 | 2 |

|  |  |
| --- | --- |
| TOTAL | 2 |

#### Criterio 7: Estado actual.

Gdvelop se encuentra en estado activo, en constante desarrollo y con equipo responsable trabajando en todo momento en la creación de nuevas funcionalidades.

Desde la web oficial se puede acceder al panel de trabajo interno en Trello[[11]](#footnote-11), Figura XX, donde se puede ver las funciones y mejoras en las que están trabajando actualmente, las ya implementadas, las que van a desarrollarse próximamente e ideas sobre próximas actualizaciones.

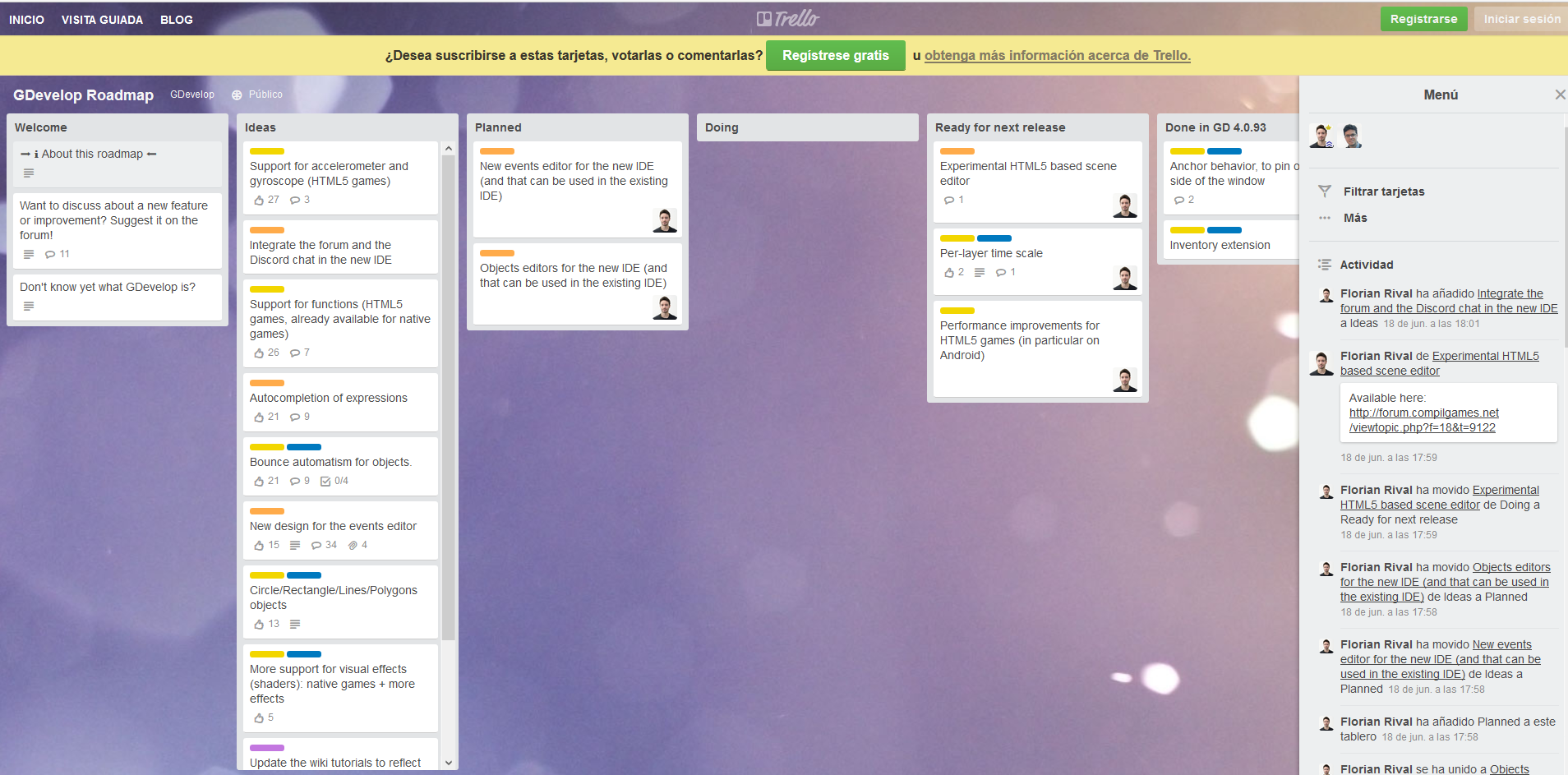


Tabla 32: Trello desarrollo Gdevelop

Con lo que no solo tenemos un flujo constante de actualizaciones, mejoras y corrección de errores, sino que también tenemos toda la información de las inminentes y futuras.

|  |  |
| --- | --- |
|  |  |
| ESTADO | ACTIVO |
| Última versión | 4.0.94 Noviembre 2016 |
| Frecuencia de actualizaciones | Constantes. |

Tabla 33: Gdevelop: estado y actualizaciones

##### Resumen puntuación criterio Estado actual

Como se ha visto, las actualizaciones son constantes y los desarrolladores están trabajando continuamente en agregar funcionalidades al programa, por lo que la puntuación en los subapartados que componen este criterio es máxima. El programa se encuentra dentro de su ciclo de vida y activo.

|  |  |  |
| --- | --- | --- |
| CRITERIO | IMPORTANCIA | PUNTUACIÓN MAX-MIN |
| Estado actual | 0-2 | 2 |
| Última versión | 0-1 | 1 |
| Fecha desde la última actualización. | 0-1 | 1 |
| TOTAL | | 4 |

Tabla 34: Gdevelop: estado actual

#### Criterio 8: Sistemas operativos soportados y requerimientos.

Las posibilidades de exportación de Gdevelop son bastante limitadas, no por el número de sistemas soportados, ya que aunque son menos que otros programas cubren los principales sistemas , sino porque se ven condicionadas por el sistema operativo en el que estemos desarrollando.

Gdevelop cuenta con las siguientes opciones de exportación:  
 1. **Exportar a Web:**  Exporta el juego para ser reproducido en la web bajos los standards de HTML5.  
 2.  **Exportar a ejecutable nativo:** Exporta el proyecto para que pueda funcionar en el sistema en el que el juego ha estado desarrollandose, es decir si se ésta trabajando con la versión de Windows el juego que con esta opción solo funciona en Windows.

Hay que añadir también que la exportación a Android e iOs de forma oficial ésta en fase de prueba y aunque se puede realizar, primero realizando una exportación a Web y posteriormente con programas externos como Intel XDK\* crear los paquetes Apk e iOs. No se trata de una creación del paquete directa, sino la conversión de un contenido web a otro adaptado para dispositivos móviles. Además el proceso resulta bastante difícil para el usuario medio y puede provocar errores e incompatibilidades que no puede controlar.

La siguiente tabla muestra las posibilidades de exportación de Gdevelop a los diferentes sistemas y plataformas.

|  |  |
| --- | --- |
| Plataforma | Exportación |
| Windows | A partir de Windows XP SP3\*1 |
| Mac | Si[[12]](#footnote-12) |
| Android | Si[[13]](#footnote-13) |
| iOs | Si5 |
| Xbox | No |
| PlayStation | No |
| Html5 | Si |
| Flash | No |
| Windows Store (Mobile y store desktop) | No |
| Linux | Si4 |

Tabla 35: Gdevelop: sistema exportación

##### Resumen puntuación sistemas operativos soportados y posibilidades de exportación.

Como se ha podido comprobar Gdevelop se puede exportar a multitud de sistemas y plataformas pero las limitaciones a la hora de realizar dicha exportación, ya que obliga a que el proyecto haya sido realizado con el sistema en el que se desea que el juego funcione, hace que la puntuación este sentido no sea excesivamente alta, como muestra la siguiente tabla.

|  |  |  |
| --- | --- | --- |
| CRITERIO | IMPORTANCIA | PUNTUACIÓN MAX-MIN |
| Posibilidades de exportación: Sistemas y plataformas | 0-8 | 4 |
| TOTAL | | 4 |

Tabla 36: Gdevelop puntuación exportación

#### Criterio 9: Licencia.

GDevelop es un software de código abierto por lo que no tienen ningún coste para el usuario y además puede ver o modificar si tiene los conocimientos necesarios el código fuente del mismo.

Al programa se le aplican varios tipos de licencia dependiendo de la parte del mismo:  
 **la IDE:** Se le aplica una licencia GPL v3, que permite usar, estudiar, compartir y modificar libremente el software siempre que estas versiones modificas conserven la misma licencia que el programa principal.  
 **Librería del código fuente, la los módulos de exportación nativa y a HTML :** Estas partes del programas están protegidas con la licencia LGPL v3\*, que permiten su libre distribución y copias pero no su modificación.  
 **Extensiones:** La licencia de las mismas es la licencia zlib/libpng, que permiten modificar y utilizar el software para cualquier fin sea comercial o no, pero el software no se debe falsificar la autoría de las partes originales y las versiones alteradas tienen que estar identificadas como tal dentro del código.

Gdevelop no tiene módulos externos para la exportación a otros sistemas como el resto de programa a la venta.

##### Resumen puntuación licencias.

Gdevelop es un software libre que usa diferentes tipos de licencia, todas libres, para cada parte del mismo. Por lo que pone a disposición de los usuarios el programa, todos los módulos de exportación de manera gratuita y sin ninguna restricción y limitación por lo que la puntuación en este apartado es la máxima posible como muestra la tabla siguiente.

|  |  |  |
| --- | --- | --- |
| CRITERIO | IMPORTANCIA | PUNTUACIÓN MAX-MIN |
| Precio de Licencias | 0-4 | 4 |
| Modo de prueba o free | 0-2 | 2 |
| Precio módulos de exportación | 0-2 | 2 |
| TOTAL | |  |

Tabla 37: Gdevelop: puntuación licencias

#### Puntuación final

Después de un análisis de Gdevelop bajo los criterios definidos el marco teórico obtiene una puntuación de 66,8 sobre 100. En la tabla siguiente se detalla la puntuación en criterios evaluables.

Estos resultados serán analizados y comparados con el resto de frameworks en el apartado conclusiones del proyecto.

|  |  |  |  |
| --- | --- | --- | --- |
| Criterio | Puntuación máxima | Puntuación obtenida | Puntuación porcentual |
| Requisitos e instalación | 6 | 6 | 100,00% |
| Documentación, tutoriales y comunidad. | 16 | 11 | 68,75% |
| Funcionamiento y dificultad de uso | 24 | 20,15 | 83,96% |
| Editor Gráfico/Visual | 8 | 6,15 | 76,88% |
| Escalabilidad | 14 | 1,5 | 10,71% |
| Resultados profesionales | 12 | 2 | 16,67% |
| Estado actual | 4 | 4 | 100,00% |
| Sistemas operativos soportados y requerimientos. | 8 | 8 | 100,00% |
| Licencias | 8 | 8 | 100,00% |
| TOTAL | 100 | 66,8 |  |

Tabla 38: Gdevelop puntuación total

1. conjunto de subrutinas, funciones y procedimientos (o métodos, en la programación orienta-da a objetos) que ofrece cierta biblioteca para ser utilizado por otro software como una capa de abstracción. [↑](#footnote-ref-1)
2. Aplicación informática que proporciona servicios integrales para facilitarle al desarrollador o programador el desarrollo de software, como pueden ser editor de código fuente, herramientas de construcción automática o compilador. [↑](#footnote-ref-2)
3. Reddit es un sitio web de marcadores sociales y agregador de noticias donde los usuarios pueden dejar enlaces a contenidos web. Otros usuarios pueden votar a favor o en contra de los enlaces, haciendo que aparezcan más o menos destacados. Se trata de un mapa de discusión, como parte de un DDS global distribuido. [↑](#footnote-ref-3)
4. Taringa! es una red social de origen argentino, de uso masivo en países de habla hispana. [↑](#footnote-ref-4)
5. La Usabilidad es la medida de la calidad de la experiencia que tiene un usuario cuando interactúa con un producto o sistema [↑](#footnote-ref-5)
6. Esta licencia es una Licencia de software libre permisiva lo que significa que impone muy pocas limitaciones en la reutilización y por tanto posee una excelente Compatibilidad de licencia [↑](#footnote-ref-6)
7. son videojuegos de rol que permiten a miles de jugadores introducirse en un mundo virtual de forma simultánea a través de internet e interactuar entre ellos. [↑](#footnote-ref-7)
8. bibliotecas multiplataforma y libres, para el desarrollo de interfaces gráficas programadas en lenguaje C++ [↑](#footnote-ref-8)
9. distribución Linux para propósitos generales basada en RPM, que se caracteriza por ser un sistema estable, la cual es mantenida gracias a una comunidad internacional de ingenieros, diseñadores gráficos y usuarios que informan de fallos y prueban nuevas tecnologías. Cuenta con el respaldo y la promoción de Red Hat. [↑](#footnote-ref-9)
10. Arch Linux es una distribución GNU/Linux independiente, de propósito general, desarrollada para x86-64. Se esfuerza en ofrecer las últimas versiones estables de la mayoría del software, siguiendo un modelo rolling-release (lanzamiento continuo). La instalación por defecto deja un sistema de base mínima, que el usuario configurará posteriormente agregando lo que necesite. [↑](#footnote-ref-10)
11. Trello es un software de administración de proyectos con interfaz web, cliente para iOS y android para organizar proyectos [↑](#footnote-ref-11)
12. Siempre que el juego se haya desarrollado en dicha plataforma [↑](#footnote-ref-12)
13. A través de programas externos [↑](#footnote-ref-13)