# Cuerpo del trabajo

## Análisis de Frameworks

### Análisis Gdevelop.

#### Criterio 1: Requisitos e instalación.

Los requisitos para poder desarrollar en Gdevelop son bastante asequibles y la mayoría 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 1: Gdevelop Requisitos de instalación

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[[1]](#footnote-1) o Arch [[2]](#footnote-2)(aunque esta no son versiones oficiales han sido creadas por la comunidad) y una versión para Mac que por el momento en estado beta y con una funcionalidad bastante limitada.

Gdevelop tiene una versión app-web, llamado **Gdevapp,** 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. Desde la App web gracias a su sistema de importación y exportación podemos trabajar con proyecto generados el programa de escritorio en la aplicación web y a la inversa.

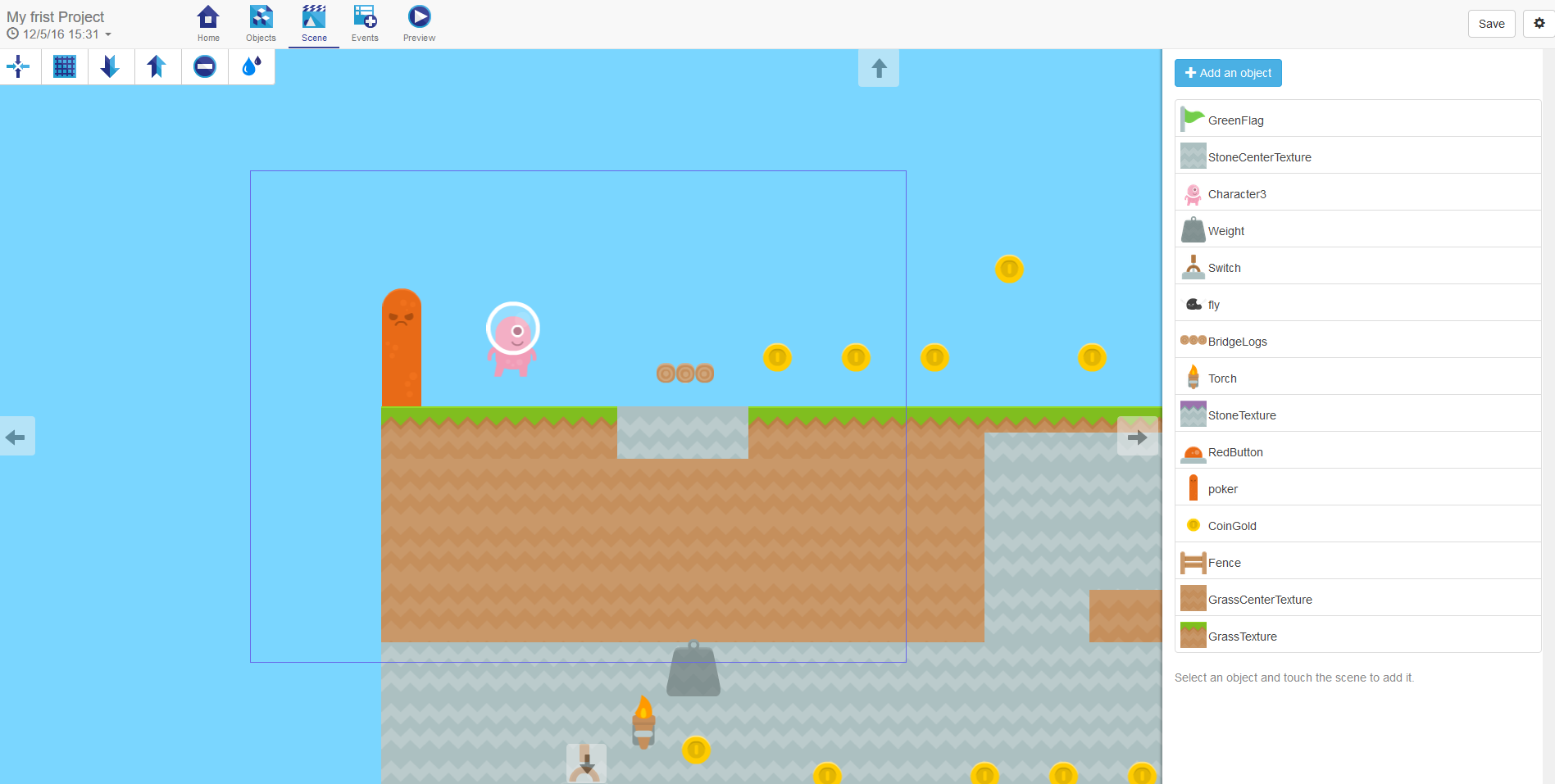


Figura 1: Gdevapp

##### Puntuación subcriterio requisitos

Como se ha comprobado los requisitos de funcionamiento de Gdevelop soy asumibles por casi la totalidad de los usuarios además no limita su uso a sistemas operativos Windows sino que aunque en el caso de Mac, todavía este en fase beta y en distribuciones Linux haya sido desarrollado por la comunidad y cuenta con aplicaciones web complementaria en la que solo se necesita un navegador web para usar el programa. 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 |

El proceso de instalación de Gdevelop una vez se ha comprobado que el equipo donde se va utilizar cumple los requisitos de instalación es bastante sencillo y tiene los siguientes pasos:

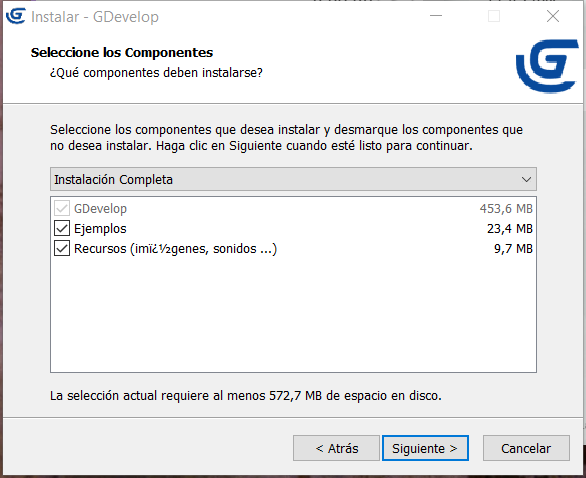
1. Descargar el paquete de instalación para él sistema en el que se vaya a desarrollar. desde la web oficial (http://compilgames.net), la descarga tiene un peso de 109 Mb.
2. El paquete de instalación ésta en castellano y siguiendo las instrucciones que el programa indica se completa la instalación. 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, 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. El programa con la instalación de 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.

|  |  |
| --- | --- |
| 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 2: Resumen proceso de instalación Gdevelop

##### Puntuación subcriterio 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.

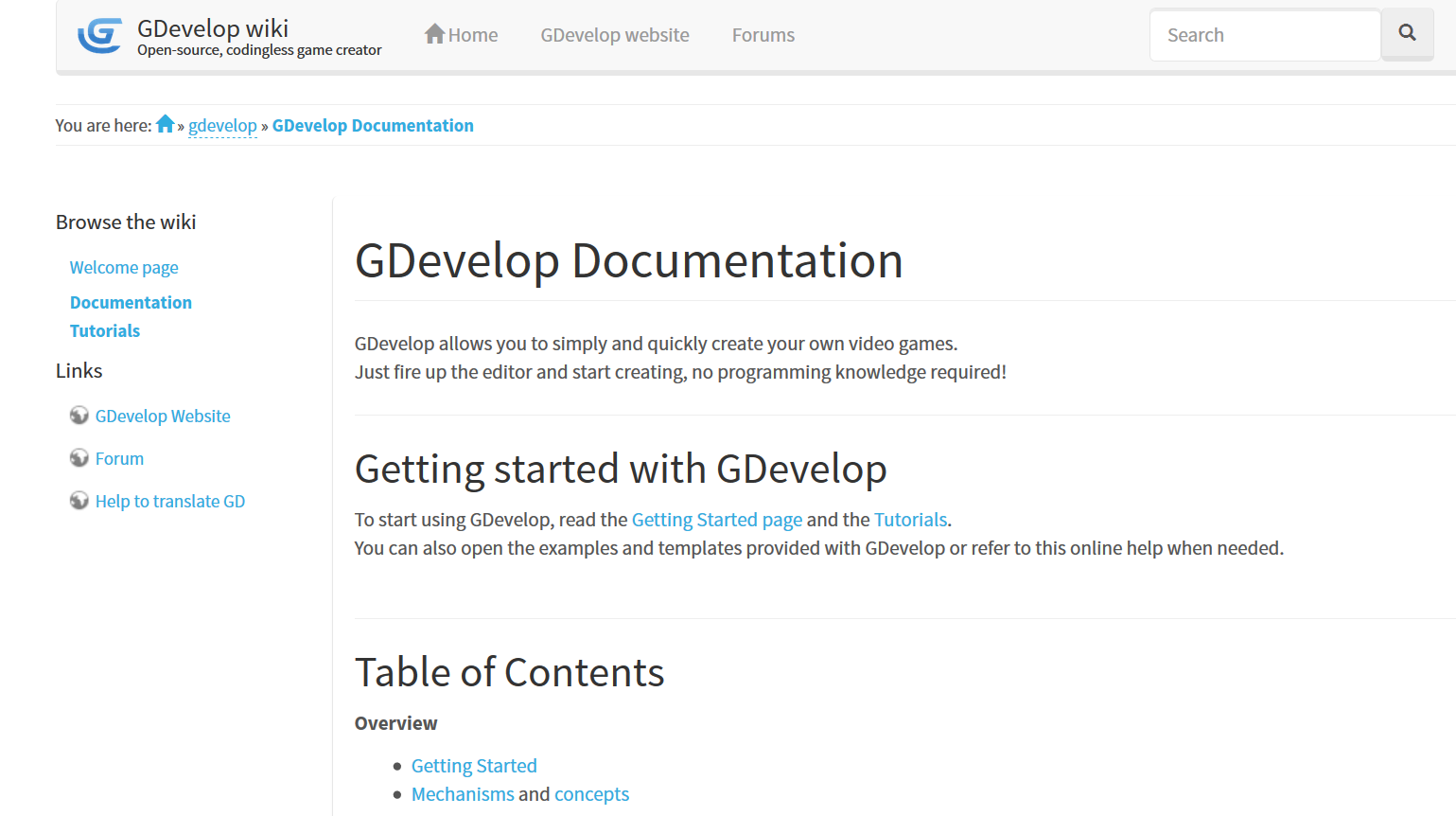


Figura 3: Portada de la wiki de Gdevelop

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

**1. Para principiantes:** En este apartado se encuentran todos aquellos tutoriales que son necesarios para poder hacer un uso básico del programa se encuentra el citado tutorial de plataformas, así como otros relativos a la exportación de proyectos y otros de elementos sencillos. *Total: 5.*

**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.   
*Total: 10.*

**3. Consejos y pequeñas explicaciones:** Recomendaciones para trabajar correctamente y aprovechar al máximo las funciones del programa y pequeños dudas que suelen aparecer recurrentemente a los usuarios.  
*Total: 4.*

**4. Tutoriales avanzados:** Enseñan el uso de las posibilidades superiores del programa. Son tutoriales para los que se necesita gran conocimiento del programa y/o conocimiento de Javascript o C++ ya que en este aparto se explica el uso de este lenguaje para crear eventos personalizados y nuevos comportamientos dentro de los proyectos.   
*Total 6.*

**5. 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.   
*Total: 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 consultar, mostrar y los juegos desarrollados, así como consultar dudas respecto a la implementación de determinadas funciones. Los foros oficiales están disponibles en inglés no existe versión por el momento en castellano.

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

Gdevelop tiene un extenso manual de uso, que además está escrito con un lenguaje entendible a todos los niveles, un foro constantemente activo y una gran cantidad de tutoriales para todos los niveles que además incluyen los sprites y recursos necesarios para centrarse únicamente en realizar el tutorial, por lo que la puntuación en este apartado casi la máxima, aunque se echa en falta algunos proyecto de ejemplo y video explicativos.

|  |  |  |
| --- | --- | --- |
| CRITERIO | PUNTUACIÓN MAX-MIN | PUNTUACIÓN |
| Documentación, foros y tutoriales oficiales | 0-4 | 3.5 |

##### Tutoriales oficiales realizados.

How to make a platformer game? **<http://wiki.compilgames.net/doku.php/gdevelop/tutorials/howtomakeeaplatformergame>**

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

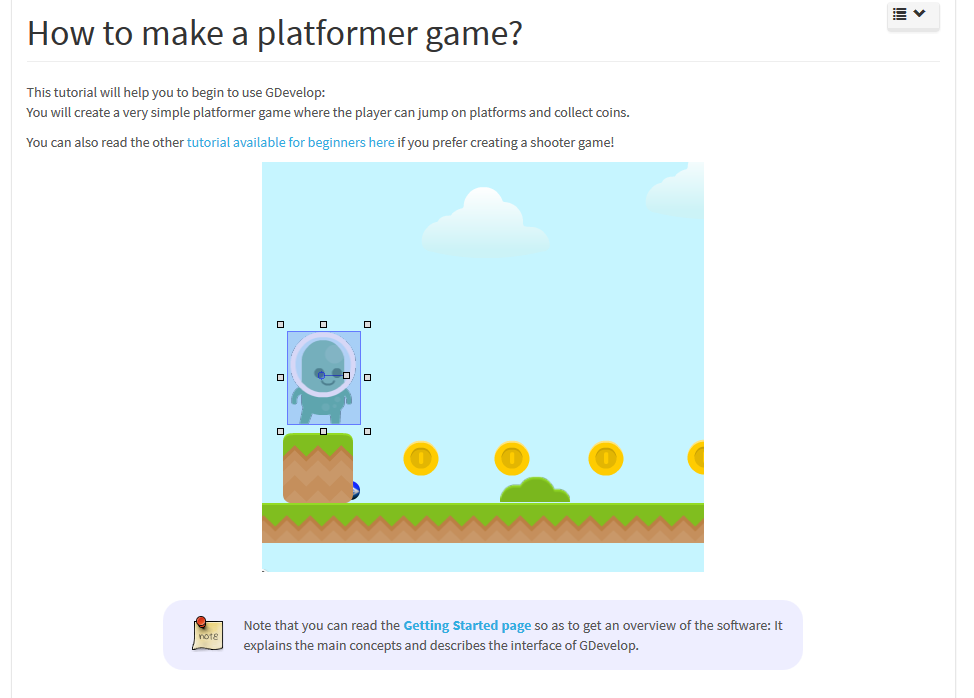
****

Figura 4:Portada del tutorial: How to make a platformer game?

El tutorial en líneas generales es un 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 apartados:

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, para maximizar el tiempo dedicado al aprendizaje y no a buscar recursos accesorios.
3. El apartado que explica los eventos, está muy bien explicado lo que permite hacerse una idea bastante buena de cómo funciona a nivel lógico.
4. Se facilita tanto un ejemplo compilado en versión web del resultado del tutorial, como el archivo fuente del proyecto por si en algún momento en usuario no sabe realizar alguna acción o quiere comparar su implementación con la del tutorial pueda acudir a este recurso.
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 indica que clase de acción es y en que parte del tutorial la hemos realizado. Lo que 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 esta en inglés y pese a que a primera vista esto no supone ningún problema como para marcarlo como inconveniente o elemento que reduce la valoración general del tutoriales, si 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 identificarlas por los iconos.
2. Algunas imágenes del tutorial no son correctas. El tutorial consiste en la creación de un videojuego de plataformas, en un apartado explica la creación de monedas en el escenario y su posterior colisión con el personaje, las imágenes que aparecen en este apartado no se corresponden con lo explicado.
3. Partes desactualizadas, en tutorial parece realizado con una versión anterior del programa y hay partes 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, la creación de este movimiento está mal explicada y estructurada es complicado entenderla correctamente.

##### Beginers Tutorial

Junto al anterior constituyen son los que dan una primera aproximación al programa y 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 disparando y mediante apuntado y disparo con el ratón.

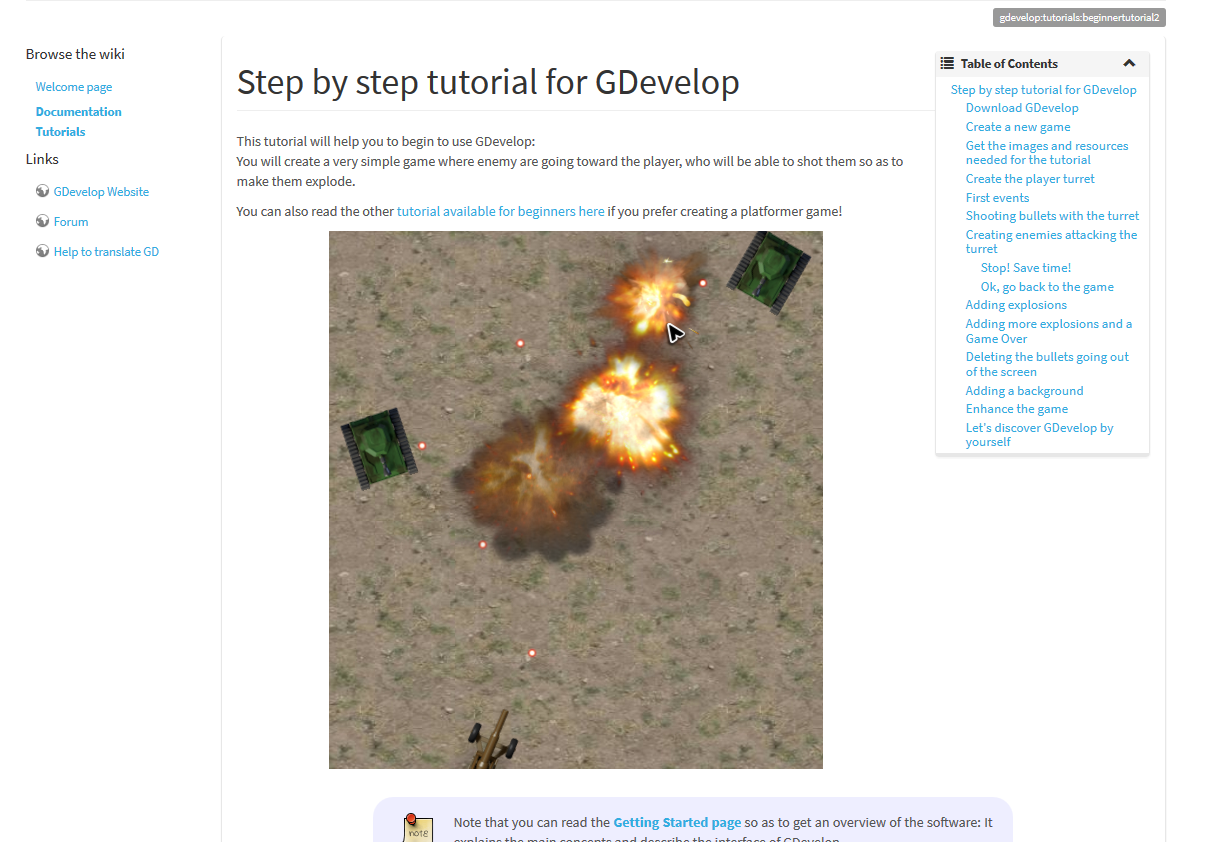


Figura 5: Beginners tutorial Gdevelop

El tutorial comparte con anterior todos los aspectos positivos que este tenía pero por su parte tiene los siguientes inconvenientes o errores:

1. Al igual que anterior el tutorial esta en inglés y el programa en castellano como se ha comentado anteriormente esto produce problemas de coherencia.
2. La parte de la animación de explosión no ésta bien explicada y si se implementa como indica el tutorial aparecerá en un lugar distinto al que se debería aparecer, para su corrección se pueden realizar dos acciones o 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:

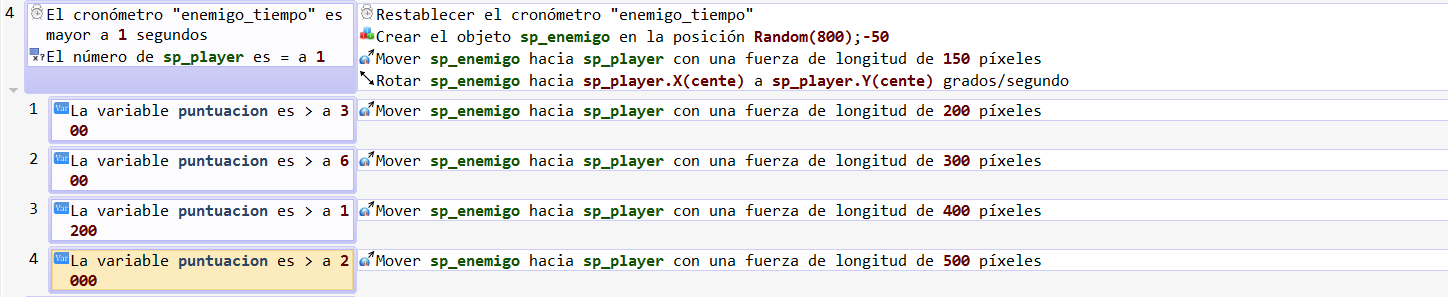
1. Toda la parte del fin de partida ha sido implementada de forma diferente a la explicada en el tutorial, ya que no era la forma correcta de realizarla en términos de gestión de recursos y un mejor flujo de juego.   
   En el manual cuando un enemigo golpea a un jugador aparece el mensaje de Game Over 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.
2. Se ha añadido una variable *puntuación* que se imprimirá en la parte superior de la pantalla y sé que incrementará cada vez que un enemigo sea destruido.
3. Se ha añadido una serie de sub- eventos, Figura que conforme la puntuación aumente también lo haga la velocidad de los enemigos, para aumentar dificultad progresivamente.   
     
   

Figura 6: Eventos puntuación

1. La creación de las balas tiene una nueva condición y es que exista un jugador en la escena, esto impide que durante el mensaje de Game Over, después de la destrucción del jugador Figura 7, sobre la escena como proponía la implementación original o en la transición entre la escena de juego y la de fin de juego de la modificación, se sigan creando balas.



Figura 7: 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 8: Resumen tutoriales Gdevelop

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

Como se puede comprar 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 e explicación y coherencia impiden que tenga la máxima puntuación.

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

Tabla 3: Subcriterio Gdevelop Análisis de tutoriales oficiales

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

RedditLa presencia de Gdevelop en Reddit es meramente anecdótica, como muestra la Figura 9, con tan solo 8 temas abierto 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.

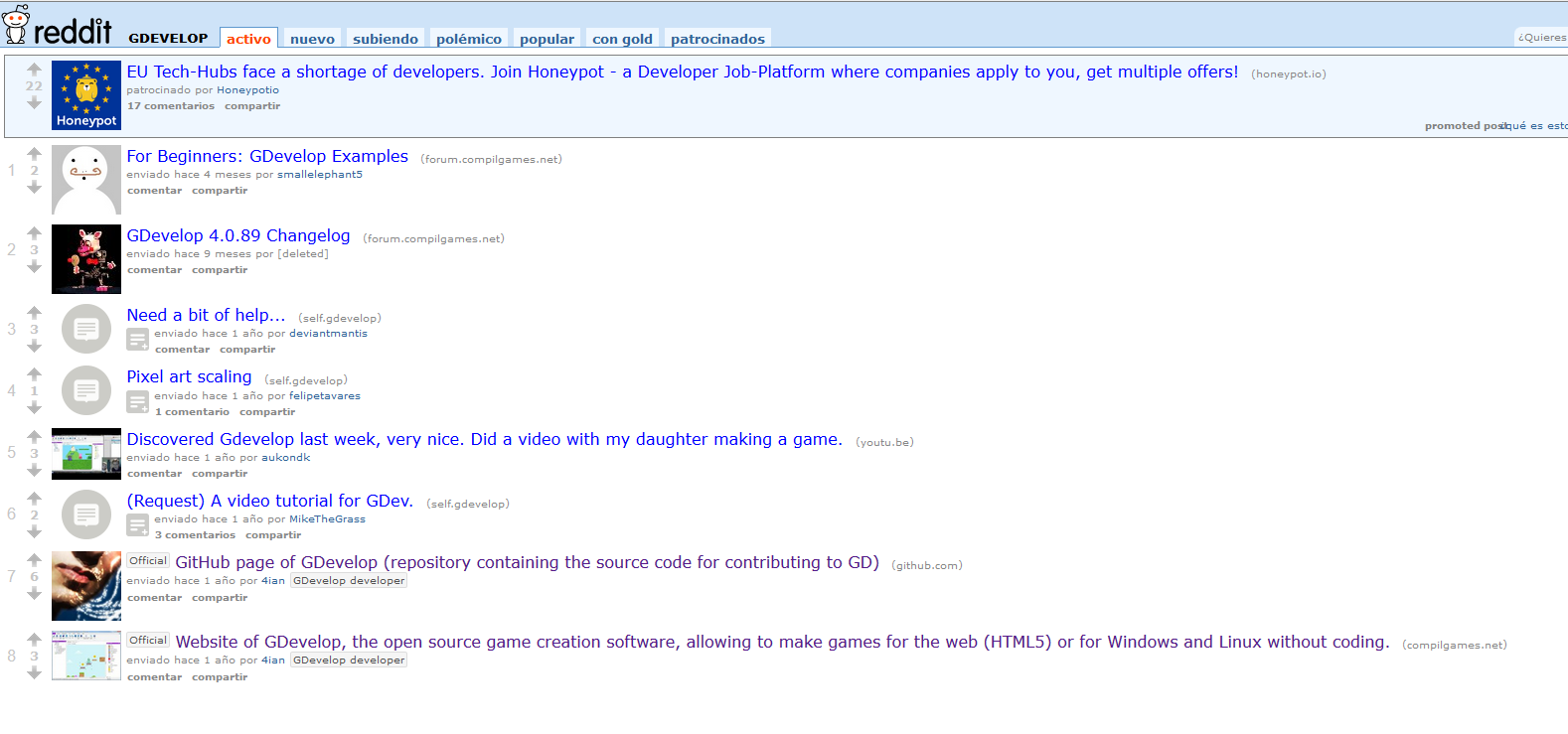


Figura 9: Gdevelop en Reedit

##### Taringa.

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

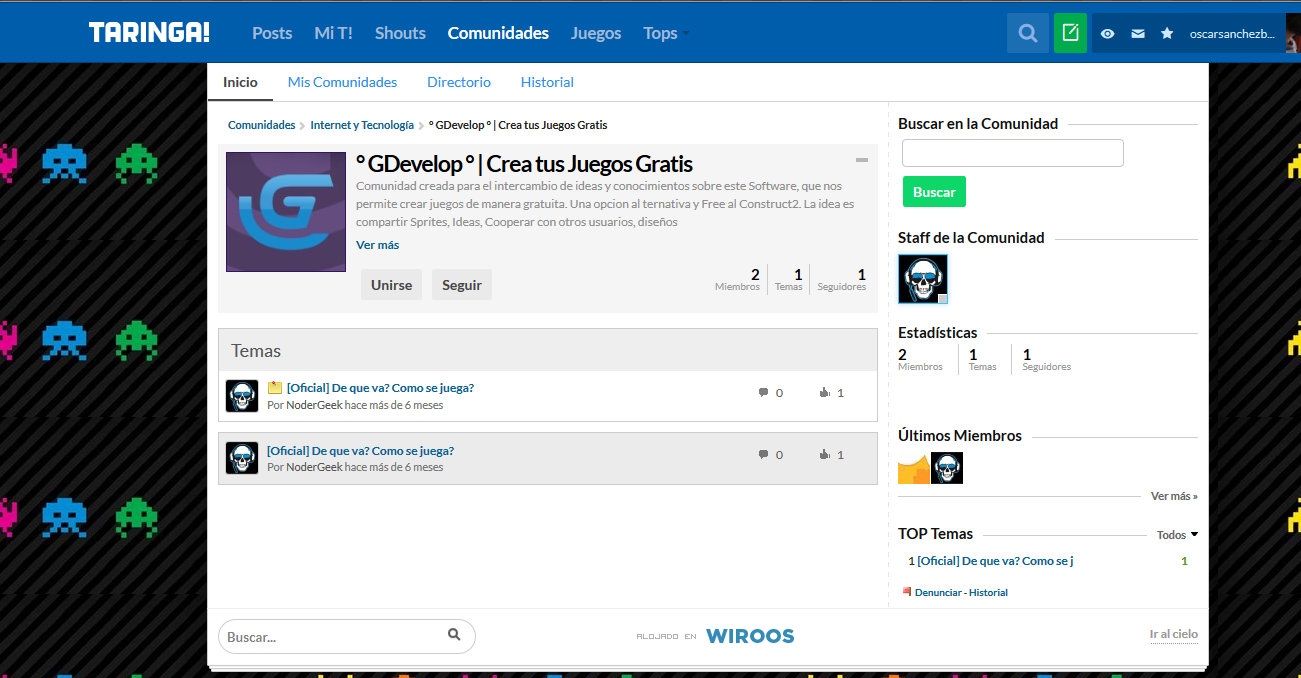


Figura 10: Gdevelop en Taringa

##### 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 | 0-4 | 0.5 | |

Tabla 7: 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 5: Gdevelop criterio documentación, tutoriales y comunidad

#### 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 fluidamente, 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 6: Gdevelop subcriterio funcionamiento general

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

Gdevelop por su concepción, funcionamiento y por los distintos elementos que se comentará 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 compresión y aprendizaje. Estas plantillas se aplican al crear un proyecto nuevo donde el programa pregunta que clase de juego, así 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 6: 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 nuevo proyecto 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:** Gdevelop permite para facilitar la creación de nuevos proyectos 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** | |

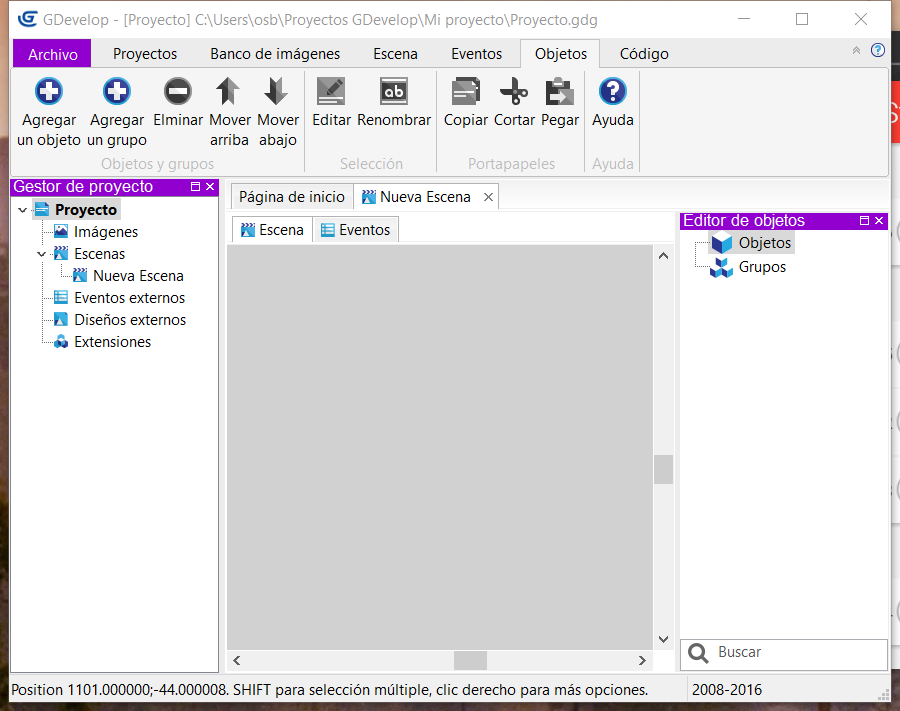


Figura 11: Gdevelop creación de un nuevo proyecto

La valoración de la creación de un proyecto es **muy fácil**, por lo que su puntuación es de 10 puntos.

###### 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 este por debajo del resto de objetos**.** Este valor siempre tiene que ser el menor que el Plano (Z) de todos los objetos si se quiere que se mantenga siempre como fondo. | |

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

Para dotar a un personaje controlable movimiento en dos direcciones, como sería 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.

Este tipo de movimiento en Gdevelop 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, tiene las variables de configuración que indica la Figura 12. Para marcar a un objeto con dicho comportamiento:

|  |  |  |
| --- | --- | --- |
|  | |  |
| 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 va 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, 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.  **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 en la que 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. | |

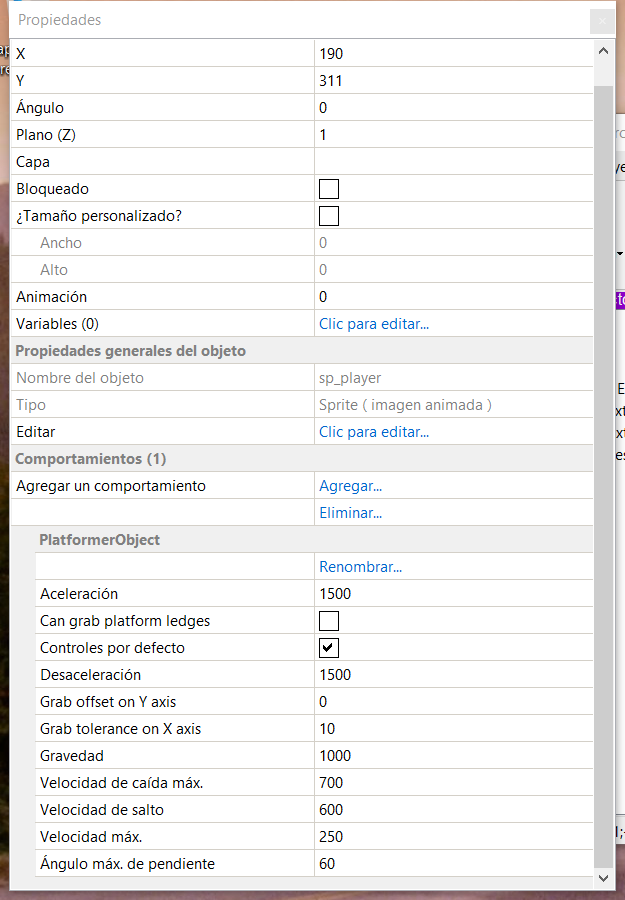
****

Figura 12: Gdevelop: Variables movimiento en 2D

1. Objeto plataforma: Pero sin 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.

Muy fácil

###### 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 negar dicha posibilidad.

Muy fácil

###### 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 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 poder mover en las diagonales o solo en las 4 direcciones principales.   **Velocidad de rotación:  Velocidad máx:** Velocidad que puede alcanzar el Sprite que se controla. | |

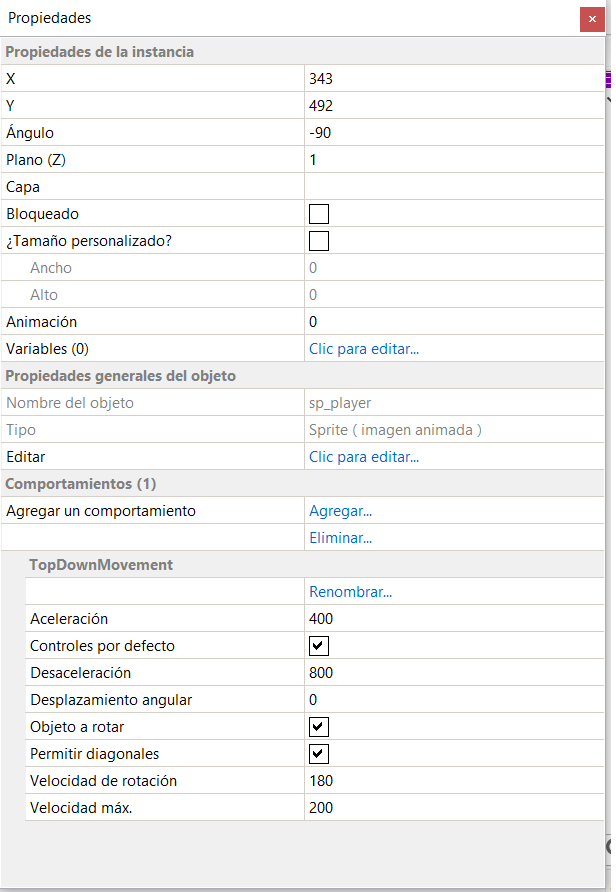
****

Figura 13: Gdevelop variables movimiento en 8 direcciones

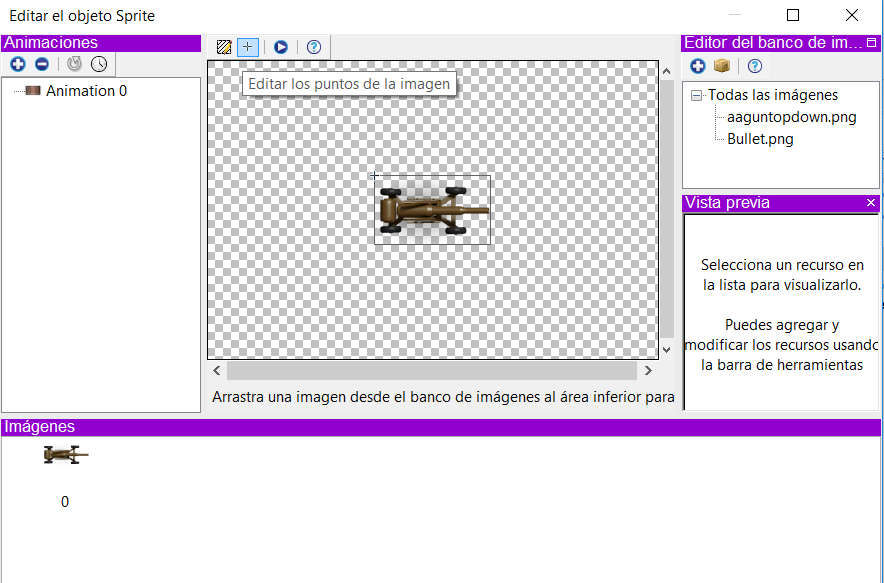
Muy fácil

###### Disparo de bala.

Para que personaje controlado por el usuario dispare balas se debe seguir el siguiente proceso:

1. En primero 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, a los que tiene por defecto (centro y origen), ya que así podemos precisar el lugar exacto donde se quiere que las balas se creen, por ejemplo en el caso de un tanque en el caño o en un avión en los laterales a modo de misiles.

|  |  |  |
| --- | --- | --- |
|  | |  |
| 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. | |



1. Creación del evento que se encargará de la creación de la bala y contener toda la lógica. 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 sucedan.

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

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

|  |  |  |
| --- | --- | --- |
|  | |  |
| Ruta/comandos | Condición ->Ratón y Táctil ->Botón de ratón presionado o toque  //PONER TAMBIÉN POR TECLA. | |
| Variables | **Botón a evaluar:** Tecla al que el programa va escuchar para en cuanto se pulse lanzar las acciones. | |

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

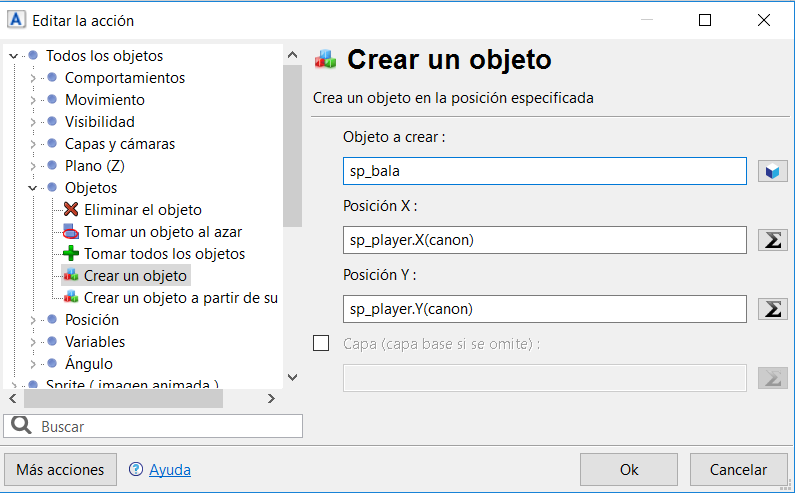
|  |  |  |
| --- | --- | --- |
|  | |  |
| Ruta/comandos | Condición-> Cronómetro y tiempo -> Valor de un cronómetro. | |
| Variables | **Tiempo en segundos:** El evento se va poder producir cuando el cronometro haya superado el tiempo aquí introducido.  **Nombre del cronómetro:** Dar un nombre al cronómetro, 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 para reiniciarlo. | |

1. Ahora hay que introducir la acción para que una vez que una bala sea creada porque se cumplen las dos condiciones se ha pulsado el botón izquierdo y el tiempo en segundo es mayor que el introducido en el valor del cronometro el cronometro se reinicie ya que sino la condición de tiempo se estaría cumpliendo todo el tiempo. Hay que indicar que para un correcto funcionamiento de la funcionalidad esta acción es conveniente que se encuentre siempre en última posición, por debajo de las acciones que se van a realizar a continuación.

|  |  |  |
| --- | --- | --- |
|  | |  |
| 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 se tienen las condiciones para que la acción de crear las balas se produzca correctamente, es turno de realizar dicha acción.

|  |  |  |
| --- | --- | --- |
|  | |  |
| 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); | |

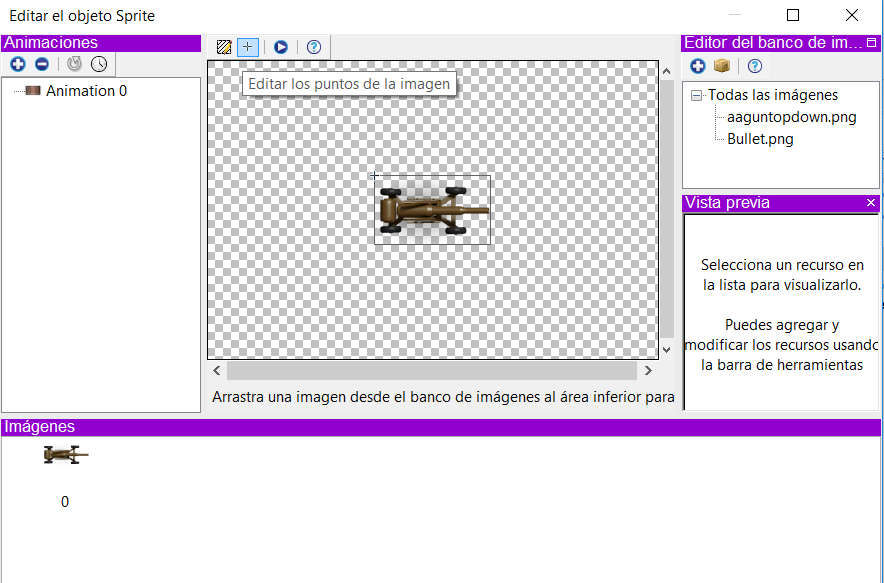


1. En la instrucción anterior se ha creado la bala, en esta se le va dar una velocidad de desplazamiento y dirección se introduce la acción debajo de la acción anterior.

|  |  |  |
| --- | --- | --- |
|  | |  |
| 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 el objeto se va mover desde su creación.  **Velocidad**: Se indica la velocidad en pixeles del objeto.  **Amortiguamiento**: Simula la resistencia que posee el escenario a la bala donde cero es total, y cuando mayor sea la cantidad menor será la resistencia tendrá. 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.

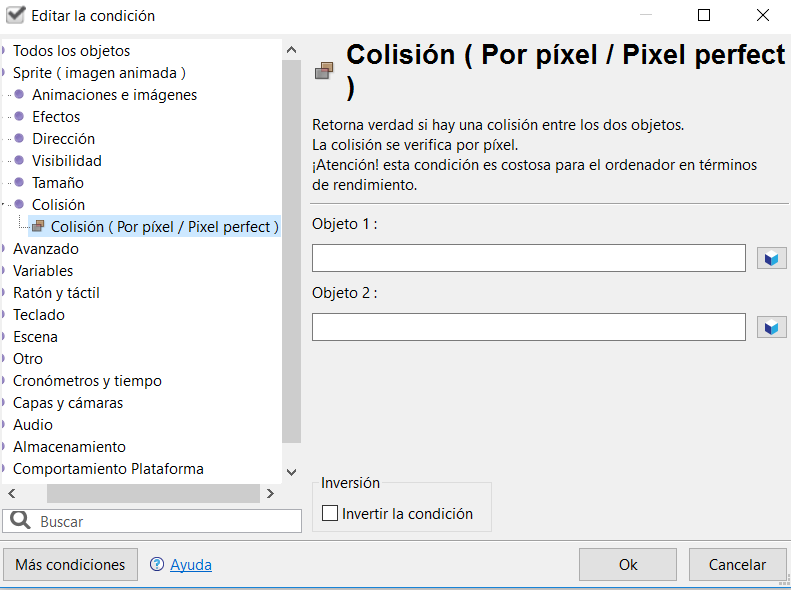
|  |  |  |
| --- | --- | --- |
|  | |  |
| Ruta/comandos | Ratón sobre el objeto bala->clic derecho del ratón ->otras propiedades->agregar comportamiento ->seleccionar destruir fuera de la pantalla | |



**DIFICIL**

###### 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. | |



**MUY FACIL.**

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

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

**MUY FÁCIL.**

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

|  |  |  |
| --- | --- | --- |
| Función | Dificultad | Puntuación |
| Creación de un proyecto | Fácil | 7.5 |
| 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 | | 65 |

**\*\*\*ESCALAR**

|  |  |  |
| --- | --- | --- |
| CRITERIO | PUNTUACIÓN MAX-MIN | PUNTUACIÓN |
| Primeros pasos: Creación de un pequeño proyecto. | 0-12 |  |

**FÁCIL**

##### Puntuación subcriterio lógica de programación necesaria

Durante el desarrollo de los tutoriales, ejemplo 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 sea el mismo 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).
2. Para mostrar elementos números por pantalla, estos deben ser convertidos a cadena de texto y para ello se debe usa una función que esta presente en la mayoría de lenguaje de programación la función ToString(Variable(Score)).

Estas son las dos únicas referencias a los lenguajes de programación que se ha encontrado, por lo que 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 |

##### Resumen puntuación funcionamiento 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 |  |
| Lógica de programación necesaria | 0-4 |  |
| TOTAL | |  |

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

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

Tabla 9: Criterio: Editor gráfico/visual

#### Criterio 5: Escalabilidad.

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

FALTA

##### 

|  |  |  |
| --- | --- | --- |
| 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 11: 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 12: 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 13: 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 14: Criterio: Sistemas operativos soportados

# 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 15: Criterio: Licencias

### Puntuación total.

|  |  |
| --- | --- |
| 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 16: 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 4: Torque 2D

Tabla 17: 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[[3]](#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 18: Unreal 4

Figura 5: 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 6: 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.

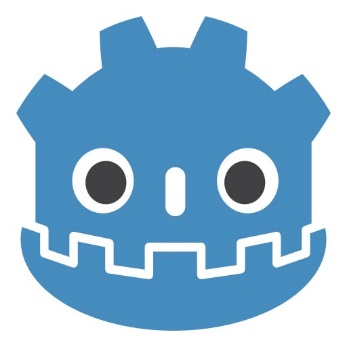
A GODOT Engine es un motor de videojuegos multiplataforma y de código abierto. Su funcionamiento basado en nodos y escenas lo diferencia en gran medida de los motores y framworks analizados en el presente proyecto.

Figura 7: 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 8: 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 [[4]](#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 9: 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 19: 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 10: Gdevelop

El programa está escrito en C++ y trabaja fundamentalmente con la librería gráfica SFML , añadiéndole por encima una interfaz basada en WxWidgets[[5]](#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 juegos. 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.

#### 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.

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

Tabla 20: Gdevelop Datos

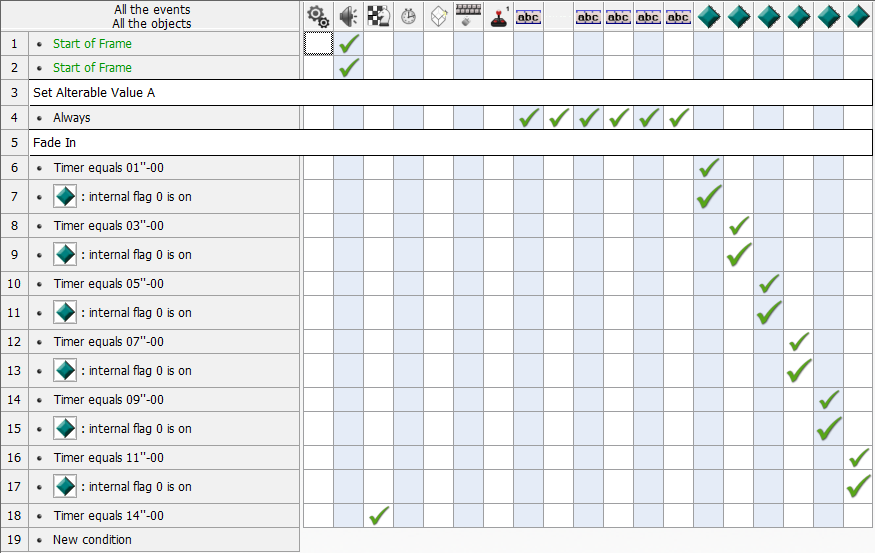


Figura 11: Sistema de eventos Click Team Fusion

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

Tabla 21: 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 12: 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.

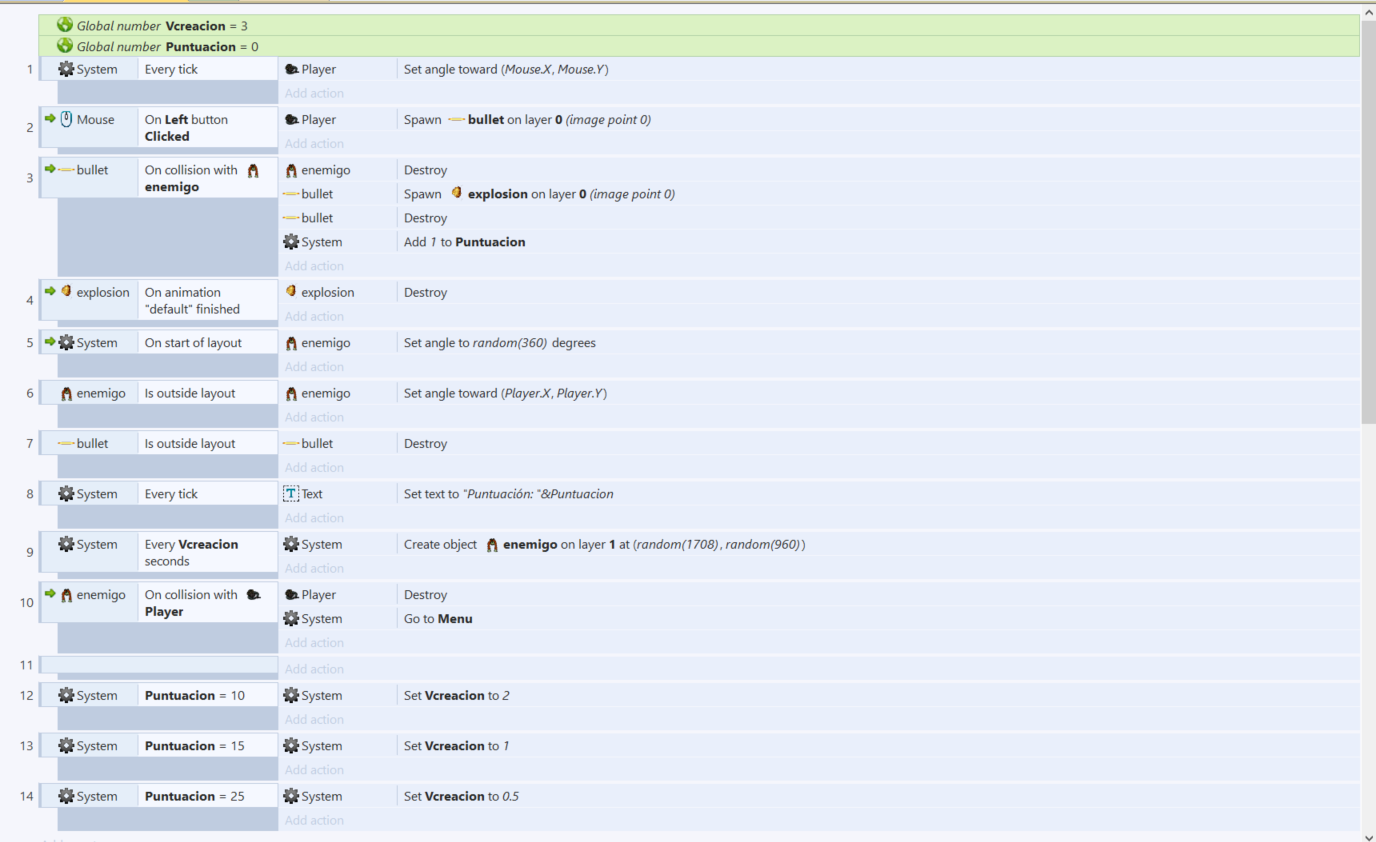
Su funcionamiento principal es a través de las hojas de comportamiento y/o evento, en las que a partir del cumplimiento de una serie de condiciones el programa ejecuta unas sentencias previamente definidas por el usuario. Además, a estas condiciones se les pueden añadir subcondiciones, o agregaciones 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 13: Hoja de Eventos Construct 2

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

Tabla 22: 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 14: 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 23: Game Maker Datos

1. 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-1)
2. 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-2)
3. 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)
4. 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)
5. bibliotecas multiplataforma y libres, para el desarrollo de interfaces gráficas programadas en lenguaje C++ [↑](#footnote-ref-8)