

Crear videojuegos sin saber programar con Construct

BIENVENIDOS!

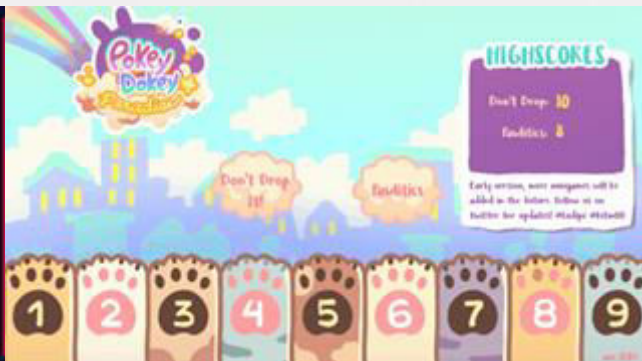
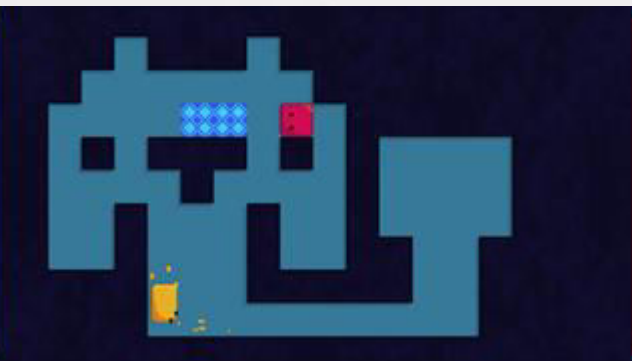
Soy Luis Díaz Peralta “Ludipe”



Me dedico a diseñar videojuegos.



ludipe.itch.io



Algunos de los juegos en los que he trabajado.

ÍNDICE (1/2)

- Introducción
 - Introducción a Construct.
 - La interfaz.
 - La lógica de Construct.
- El prototipo
 - Preparar el proyecto.
 - Planificación.
 - Creando un personaje.
 - Añadiendo disparos.
 - Añadiendo enemigos.
 - Añadiendo una IA a los enemigos.

ÍNDICE (2/2)

- Puliendo el juego
 - Introducción al game feel.
 - Aplicando efectos al juego.
 - Añadiendo assets de arte finales.
 - Añadiendo audio.
 - Creando más niveles.
 - Añadiendo progresión entre niveles.
- Terminar el juego y publicar
 - Creando un menú.
 - Compilar y publicar el juego.

Introducción

NEW Construct 3 updated 21 hours ago

CONSTRUCT 3 GAME MAKING SOFTWARE

Construct 3 is the best software to create games. Over 100,000 users monthly make & sell thousands of games globally.

[TRY IT NOW!](#)

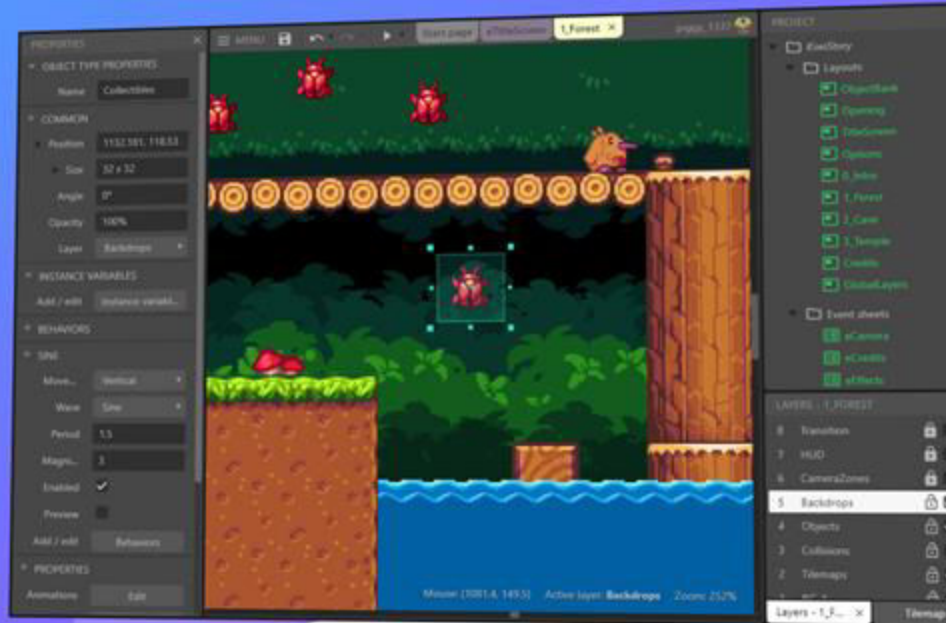
SOME OF OUR CUSTOMERS



SEGA



zyngo



¿Qué es Construct?



Un motor de juegos 2D optimizado para web (HTML5) y que permite exportar a otras plataformas.

Orientado a gente que no programa.

Puede usarse tanto para proyectos comerciales como para prototipos o juegos más pequeños.

Basado en programación visual y con una interfaz muy sencilla.

¿Por qué usar Construct?



- Para hacer juegos como hobby.
- Para empezar a hacer un portfolio y entrar en la industria del videojuego.
- Para prototipar proyectos más grandes.
- Para hacer herramientas de diseño.
- Otros (hacer una web dinámica, aplicaciones de móvil, etc)

Ventajas de Construct



POPULAR

Construct 3 is used by over **100,000 people** from all over the world each month.



PROVEN

Over **50%** of HTML5 games submitted to Kongregate are made in Construct



PRACTICAL

Runs in the browser. Translated into 6 languages.

- Hacer cualquier juego ya es un gran reto de por sí. La mayor parte de gente que está empezando tiene muchas dificultades para acabar un proyecto. Un motor accesible como Construct nos da facilidades para ir cogiendo confianza.
- Es ideal para equipos pequeños o proyectos donde hay pocos recursos.
- Permite habituarse a la programación visual (que se usa en muchos motores como Unreal o Unity).
- Optimizado para juegos web, compartir tus creaciones es muy simple.
- Gran cantidad de features y posibilidad de ampliarlo con Javascript. Tiene pocas limitaciones.

Desventajas de Construct

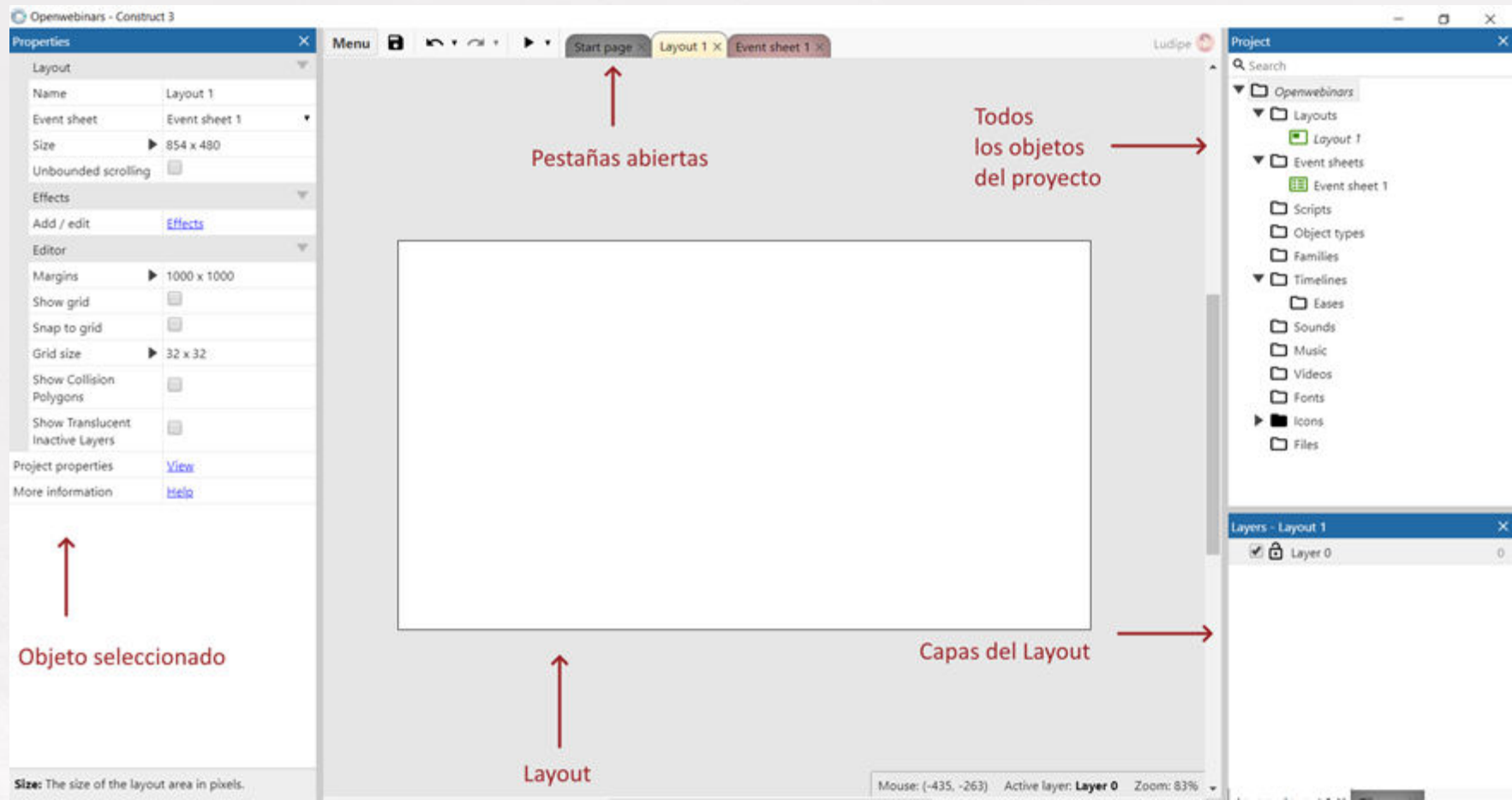
1	System	On start of layout	Add action
2	System	For each gancho	gancho Create Physics revolute joint at image point 0 to Sprite
	gancho	Is overlapping Sprite	gancho Spawn cuerda1 on layer 0 (image point 1)
	System	Pick gancho where $gancho.largocuerda = 0$	cuerda1 Create Physics revolute joint at image point 0 to gancho
			cuerda1 Set DGANCHO to <i>gancho.IID</i>

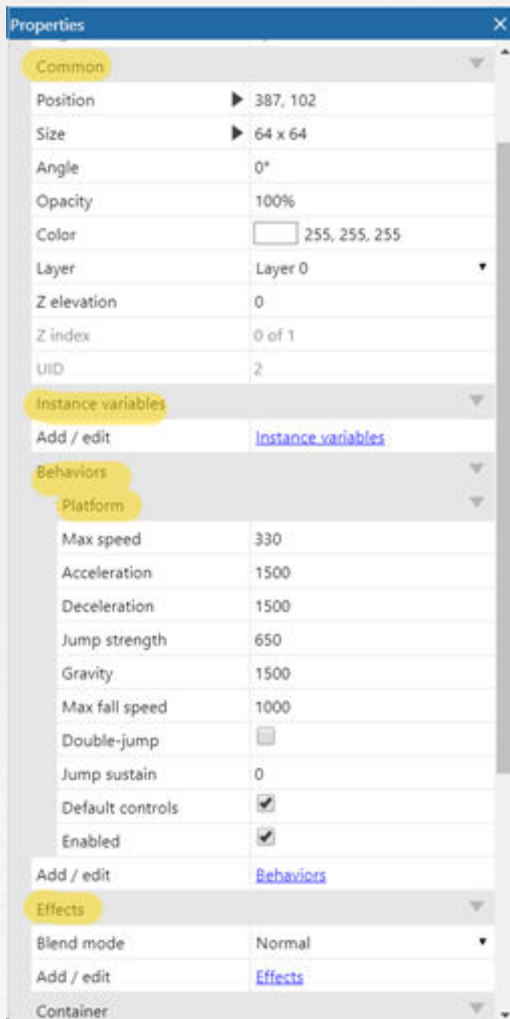
- Exportar a consolas es más complicado que con otros motores.
- Aunque hay plugins que permiten usar modelos 3D, Construct está diseñado pensando sólo en juegos 2D.
- La versión gratuita es muy limitada. Nos servirá para ver si queremos seguir usando Construct, pero es mejor comprar una licencia cuanto antes. Una licencia de Construct 2 es permanente y cuesta 160€. La licencia de Construct 3 cuesta 90€ y hay que renovarla cada año.
- Los programadores echarán en falta tener más libertad.

Construct 2 y Construct 3

- La versión gratuita de Construct 2 tiene menos limitaciones que Construct 3.
- Construct 2 es más barato a largo plazo.
- Construct 2 tiene una cantidad enorme de plugins disponible.
- Aunque se siguen corrigiendo errores, Construct 2 ya no recibe contenido nuevo.
- La interfaz de Construct 3 es casi idéntica a la de Construct 2.
- Se puede usar desde el escritorio o usando su versión web.
- Es mucho más fácil exportar a móviles.
- Permite añadir extensiones de Javascript.
- Construct 3 se actualiza con frecuencia y se añaden nuevas funciones.

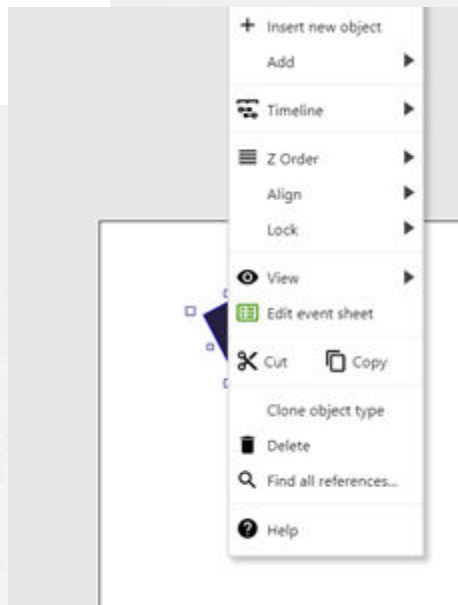
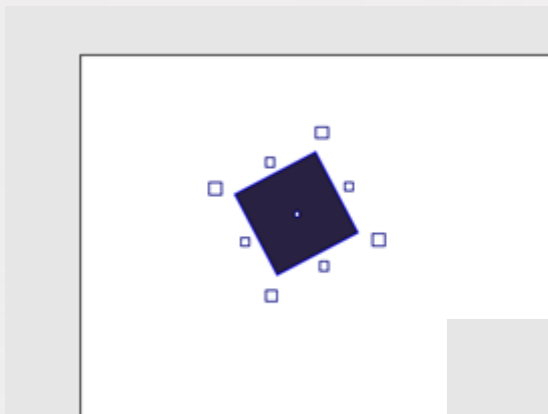
El objetivo de las siguientes diapositivas es familiarizarse con los nombres y características del motor. No es necesario memorizar todo ni entender el funcionamiento de cada elemento mostrado aquí.



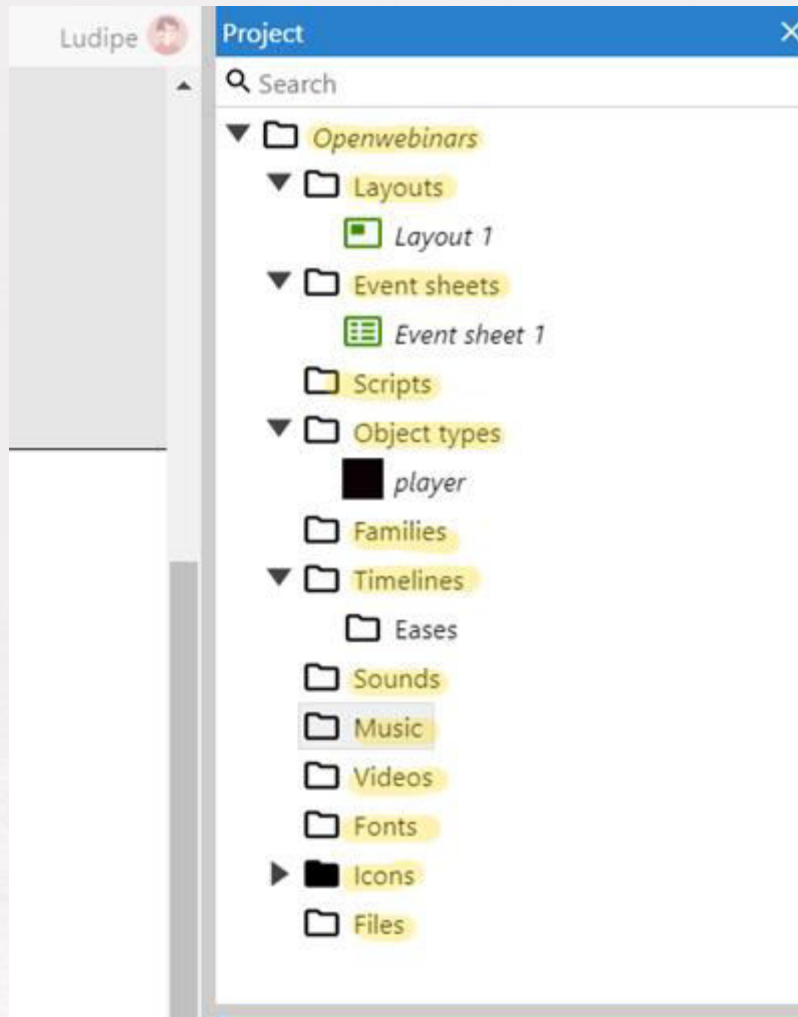


El inspector de objetos nos enseña toda la información que tiene Construct sobre el objeto seleccionado.

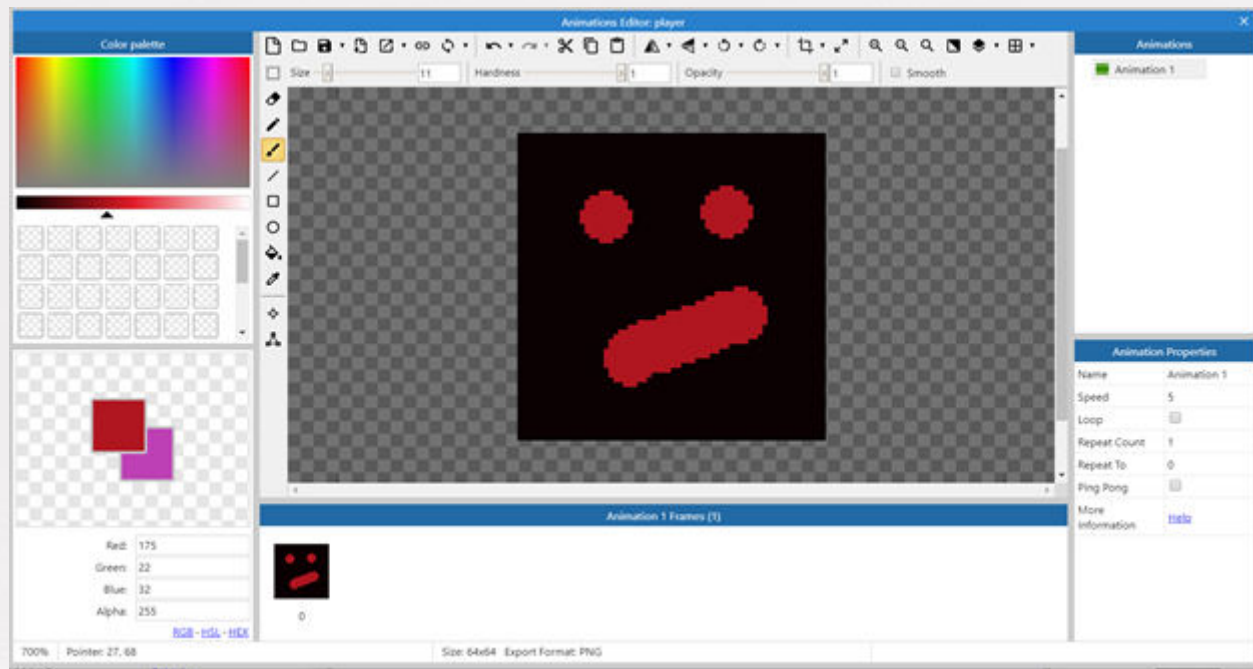
Aquí vemos desde los valores comunes hasta los campos especiales que puede tener un objeto al añadirle lógica adicional.



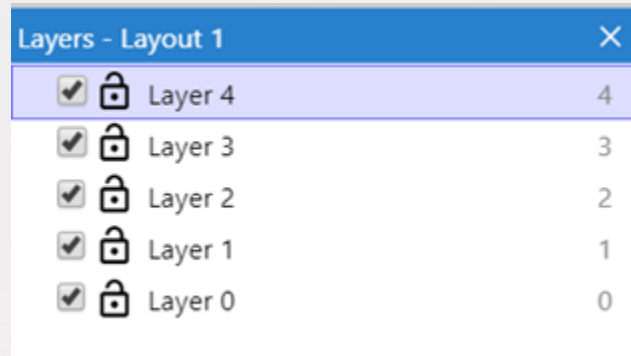
En el layout podemos arrastrar objetos para recolocarlos. Además podemos hacer click derecho para abrir un menú desplegable que nos permite cambiarlos de capa, alinearlos y usar otras funciones de Construct.



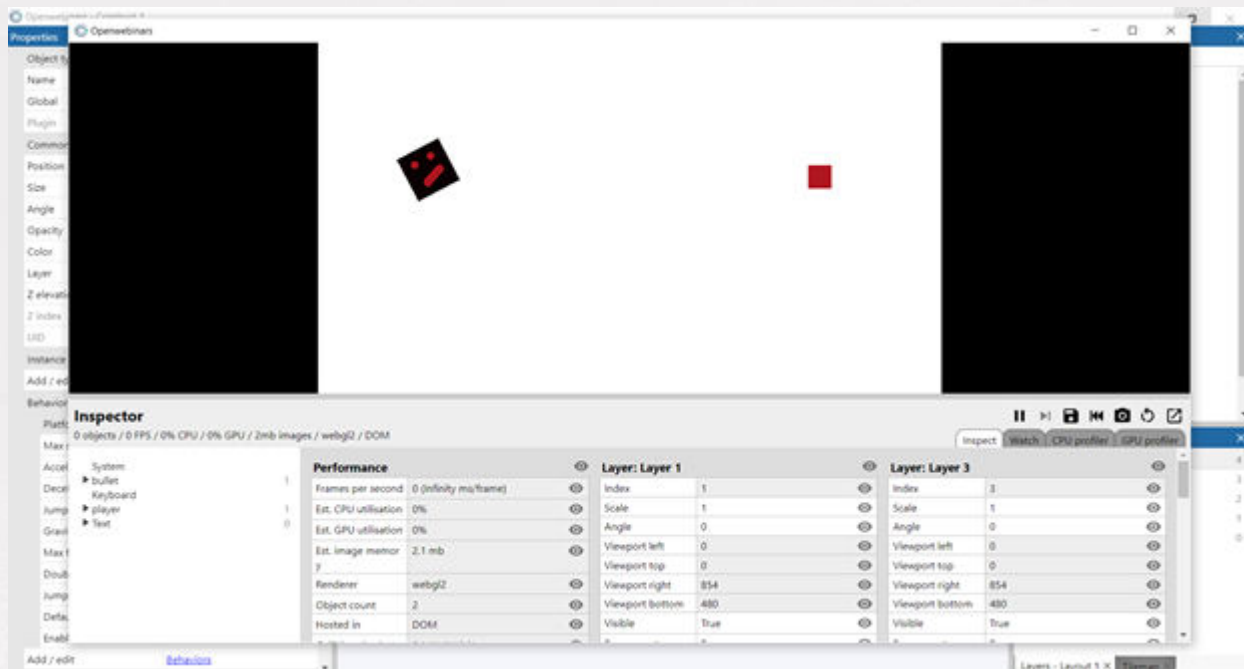
El índice del proyecto nos muestra todos los elementos de nuestro juego. Podemos hacer click derecho sobre las carpetas para añadir más o crear subcarpetas.



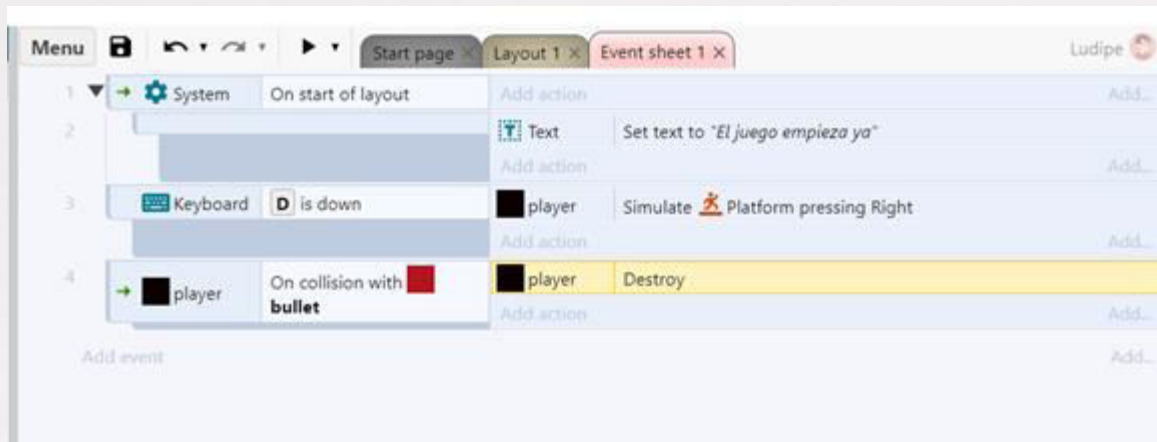
Construct tiene algunas interfaces adicionales, como el editor de “sprites” (recursos gráficos). Si hacemos doble click sobre un “sprite” se abrirá una ventana donde podemos editarlo y personalizar las animaciones.



El sistema de capas es similar al de photoshop y otros editores gráficos. Las capas situadas más arriba tendrán más prioridad y se dibujarán sobre las demás.

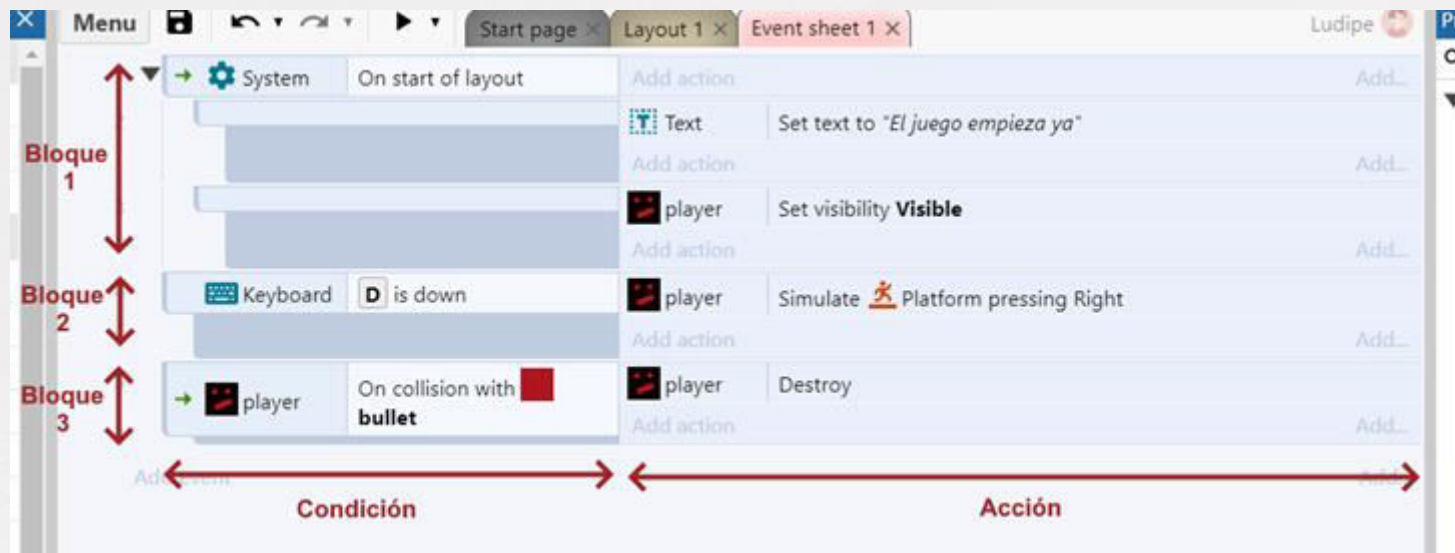


Junto al botón de “play” (que ejecuta el juego) hay un botón de “debug” que nos permite lanzar el proyecto mientras vemos los valores de todos los objetos. Esta función es muy útil para detectar y localizar errores.



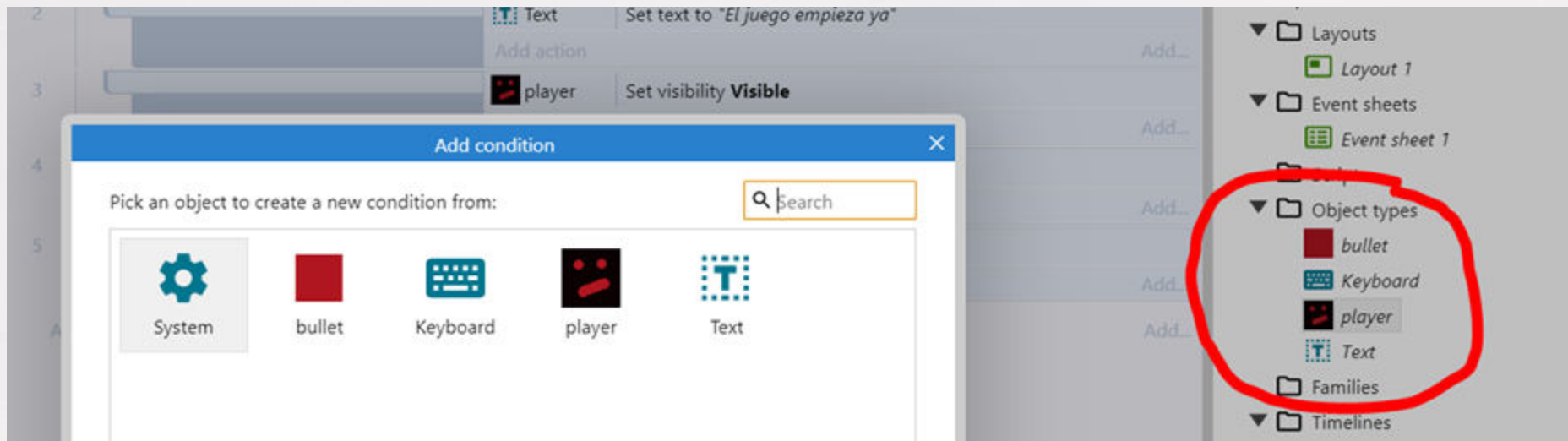
Las “event sheets” son hojas de normas que indican cómo se comportan los objetos de un “layout”.

Cada “layout” tiene asociada una única event sheet pero esa “event sheet” puede a su vez incluir otras.

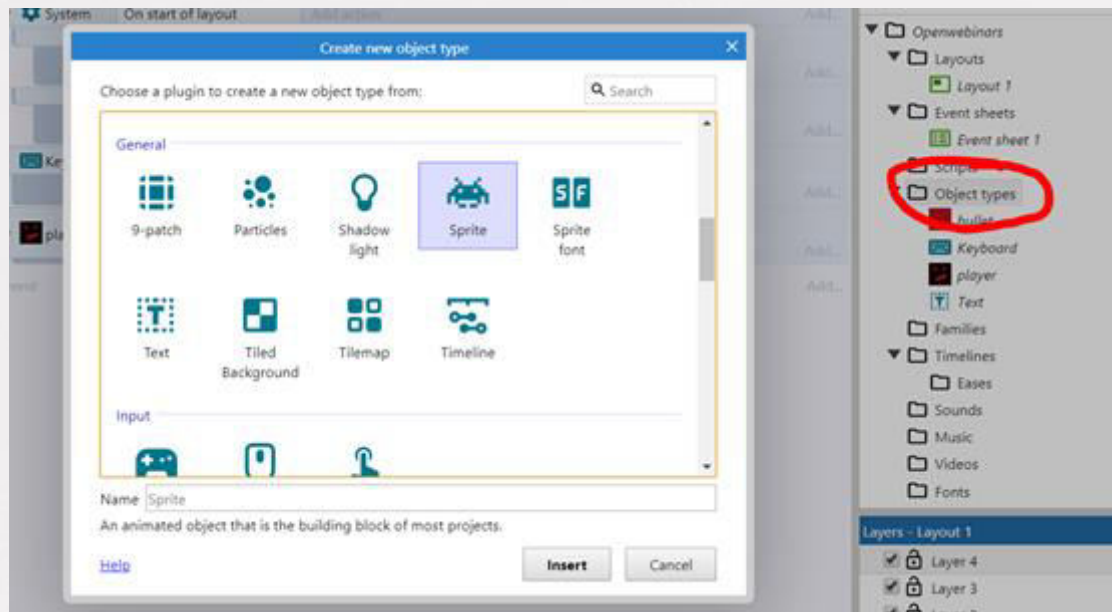


En las “event sheets” tenemos una lista de normas (o eventos) separadas por bloques. Cada norma tiene una condición (lo que necesita para activarse) y una acción asociada (qué pasa cuando se da esa condición).

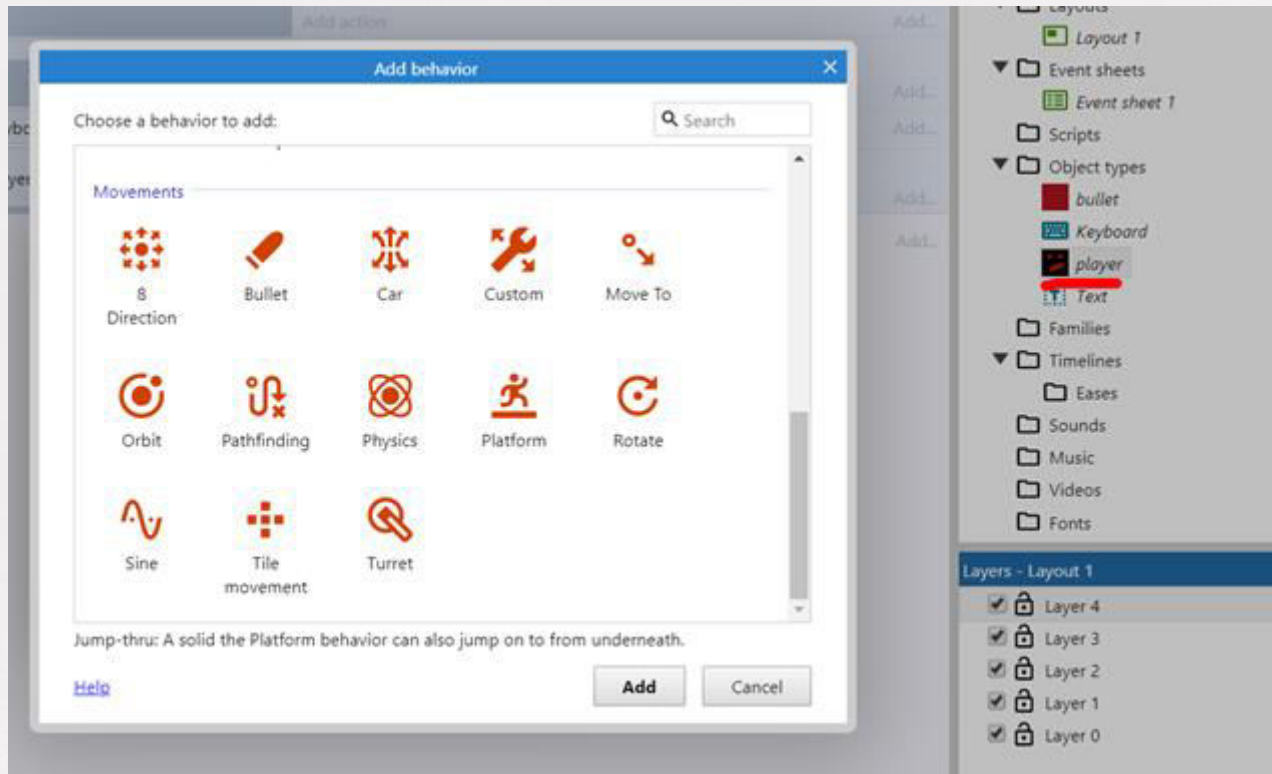
Por ejemplo, si tenemos un objeto bullet (una bala) y queremos que cuando impacte contra el jugador éste desaparezca, entonces crearemos un evento que se active con la condición “bullet hace colisión con el jugador” y cuya acción sea “destruir al jugador”.



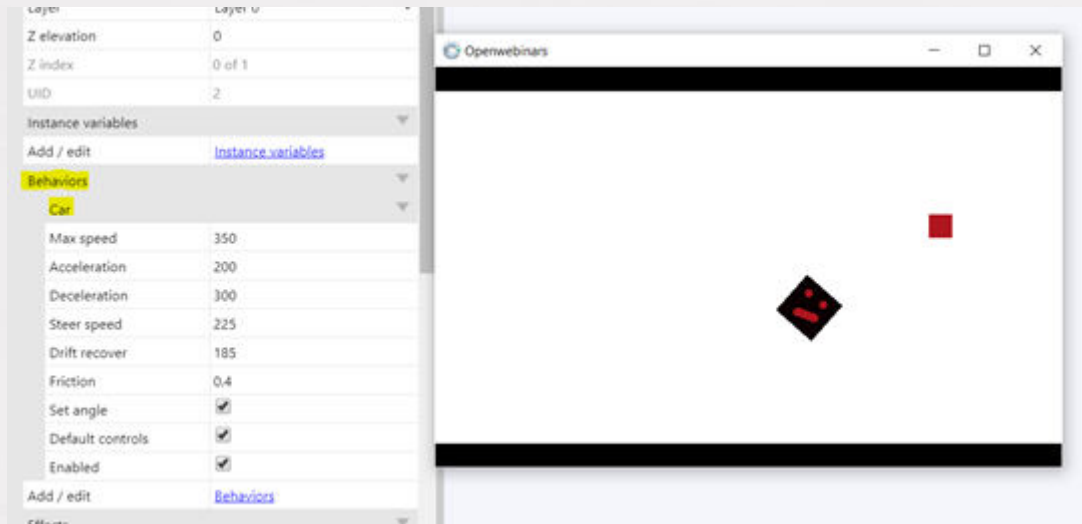
Al hacer click en crear un evento lo primero que nos pregunta es la condición que queremos establecer. Las condiciones que tenemos disponibles están limitados por los objetos que hemos añadido al proyecto. Al hacer click sobre un objeto se nos despliega una lista de posibles condiciones. Es muy recomendable leer por encima las condiciones y acciones asociadas a cada objeto así como el manual de Construct.



Al hacer click derecho en “Object types” podemos añadir un nuevo tipo de objeto al proyecto (texto, imágenes, vídeo, ratón, mando, etc). Hay una gran variedad de objetos, al seleccionarlos nos dará una breve explicación de qué es cada uno, aunque lo mejor es leer siempre el manual. El objeto “sprite” (imagen 2D) es el más común. Vamos a añadirlo a nuestro proyecto y a utilizar el editor de imágenes de Construct para pintarlo.

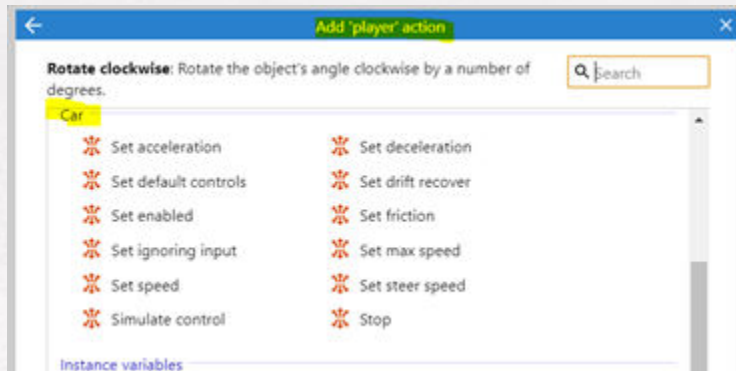


Ahora podemos arrastrar el “sprite” que hemos añadido al “layout”. Sólo es una imagen estática suspendida en la pantalla, no hay nada que le diga a Construct que esa imagen tiene que seguir algún tipo de lógica. Construct tiene un tipo adicional de componente llamados “behaviour”, son unos comportamientos que podemos asignar a un objeto para darle unas propiedades de forma rápida.



Si le damos un “behavior” de tipo coche y pulsamos “play” comprobaremos que ya podemos controlar al personaje principal como un coche pulsando las flechas del teclado.

Si hacemos click sobre el objeto también aparecerán en el inspector las propiedades de este comportamiento (“Max speed”, “acceleration”, etc) para que las podamos editar.



Finalmente, al añadir nuevos eventos veremos condiciones y acciones vinculadas al comportamiento que hemos añadido.