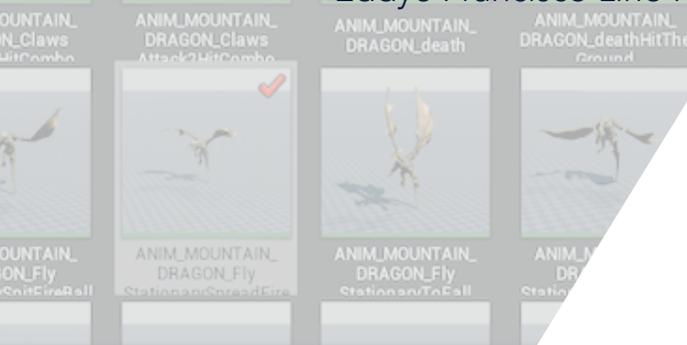


Guía paso a paso para el uso de herramientas que permitan la creación de un videojuego

Hugo Francisco Mora Marcillo
Eddy Francisco Lino Matamoras



Guía paso a paso para el uso de herramientas que permitan la creación de un videojuego

© Hugo Francisco Mora Marcillo
Eddy Francisco Lino Matamoros
Afilación: Instituto Superior Tecnológico Espíritu Santo con condición
de universitario, Guayaquil, Ecuador

Título del libro

Guía paso a paso para el uso de
herramientas que permitan la
creación de un videojuego

ISBN: 978-9942-33-627-9

Publicado 2022 por acuerdo con los autores.
© 2022, Editorial Grupo Compás
Guayaquil-Ecuador

Mora, H., Lino, E. (2022) Guía paso a paso para el uso de herramientas que
permitan la creación de un videojuego. Editorial Grupo Compás

Grupo Compás apoya la protección del copyright, cada uno de sus textos
han sido sometido a un proceso de evaluación por pares externos con base
en la normativa del editorial.

El copyright estimula la creatividad, defiende la diversidad en el ámbito de las
ideas y el conocimiento, promueve la libre expresión y favorece una cultura
viva. Quedan rigurosamente prohibidas, bajo las sanciones en las leyes, la
producción o almacenamiento total o parcial de la presente publicación,
incluyendo el diseño de la portada, así como la transmisión de la misma por
cualquiera de sus medios, tanto si es electrónico, como químico, mecánico,
óptico, de grabación o bien de fotocopia, sin la autorización de los titulares
del copyright.



@grupocompas.ec
compasacademico@icloud.com

ÍNDICE

1

CREAR UN NUEVO PROYECTO UNREAL ENGINE

13

IMPLEMENTACIÓN DE ENTORNOS

15

AGREGAR TEXTURAS

17

DESCARGAS DE ASSETS

18

IMPORTAR ASSETS

22

ILUMINACIÓN DE ENTORNOS

26

USAR ASSETS (PUERTAS)

32

USAR ASSETS (ILUMINACIÓN)

34

INTERACTUAR CON OBJETOS

36

INSPECCIONAR OBJETOS

38

IMPLEMENTACIÓN DE TRIGGERS

41

AGREGAR MATERIAL

INTRODUCCIÓN

La siguiente guía para el uso de herramientas que permitan crear un videojuego, fue desarrollada para facilitar la comprensión y orientar al desarrollador mediante el efectivo uso de herramientas que encontraremos disponibles en el motor gráfico de Unreal Engine 4 propiedad de Epic Games.

El cual está basado en una serie de Blueprints con mejoras de procesos especializadas para el fácil desarrollo de un videojuego.



_Crear un nuevo Proyecto Unreal Engine

Abrimos el programa Unreal Engine y seleccionamos un Blueprints "vacío" y damos un nombre para nuestro proyecto y damos clic en crear proyecto.



_Crear Movimientos del personaje

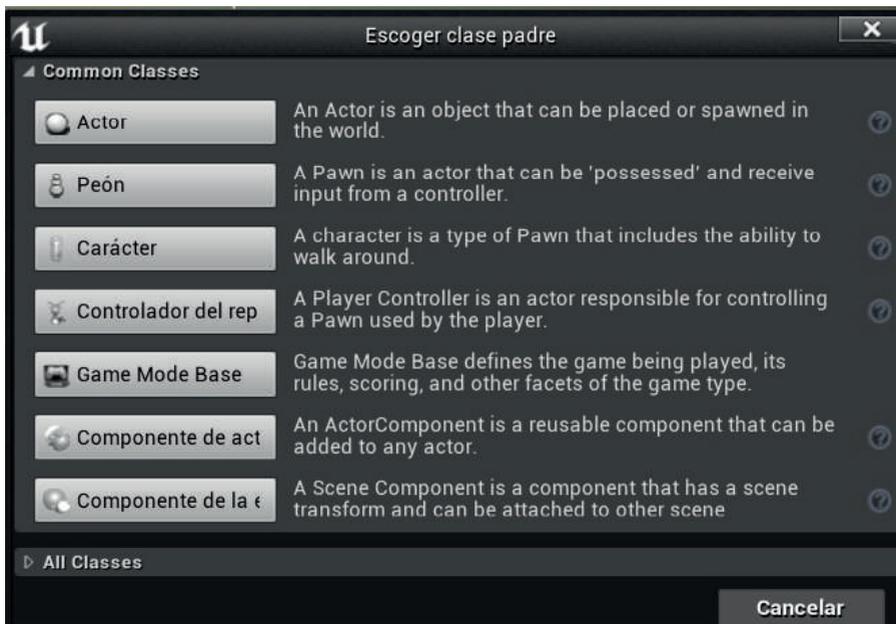
Una vez abierto el programa nos situamos en la parte inferior izquierda y damos clic derecho "carpeta nueva" para crear una carpeta contenedora para nuestro proyecto, en este caso llevará el nombre de Blueprints. Crearemos otra carpeta con el nombre "Modo juego"



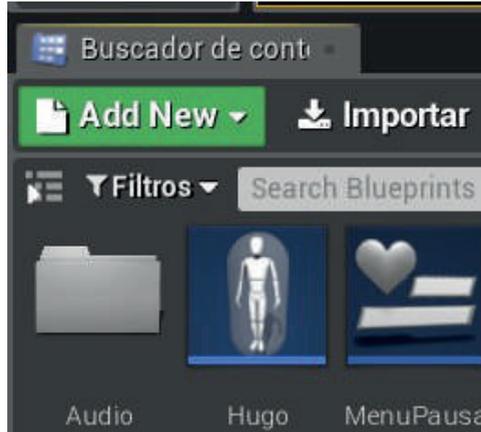
Vamos a crear un nuevo Blueprints



Seleccionamos la función "carácter" como clase padre



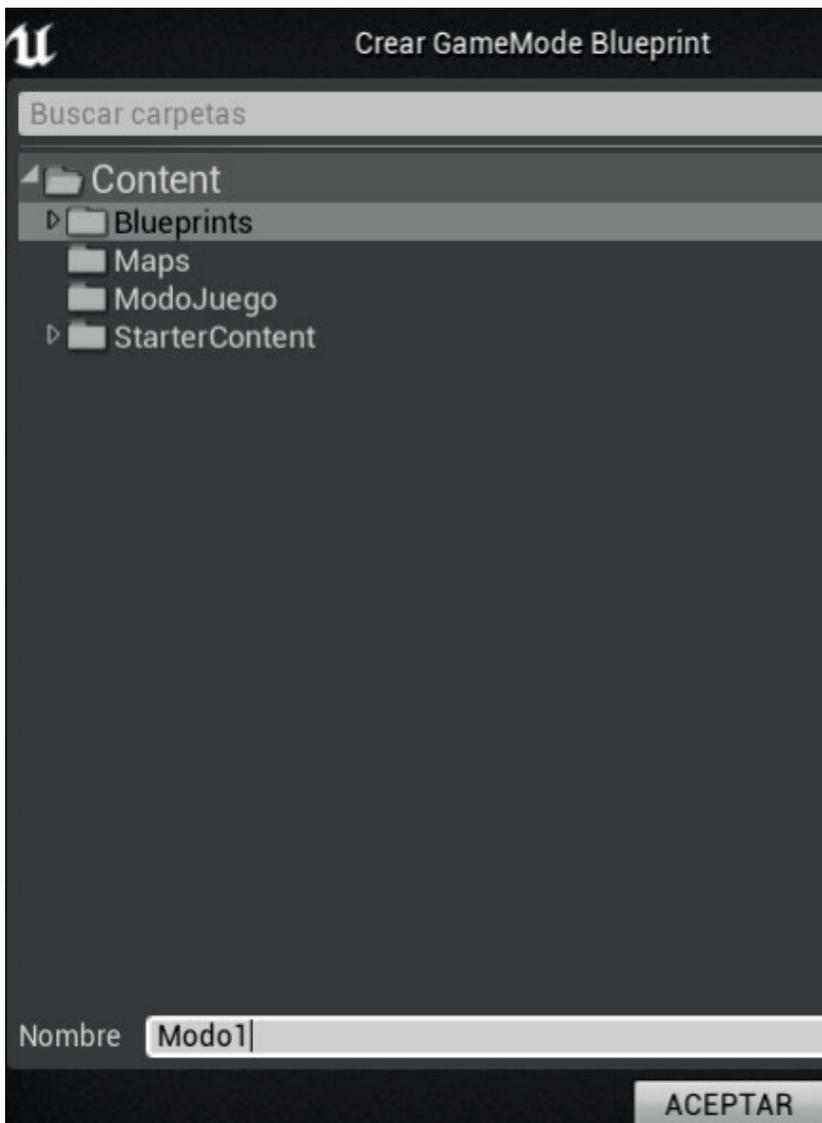
Le pondremos nombre a esta función, en esta ocasión se llamará “Hugo” que será nuestro personaje



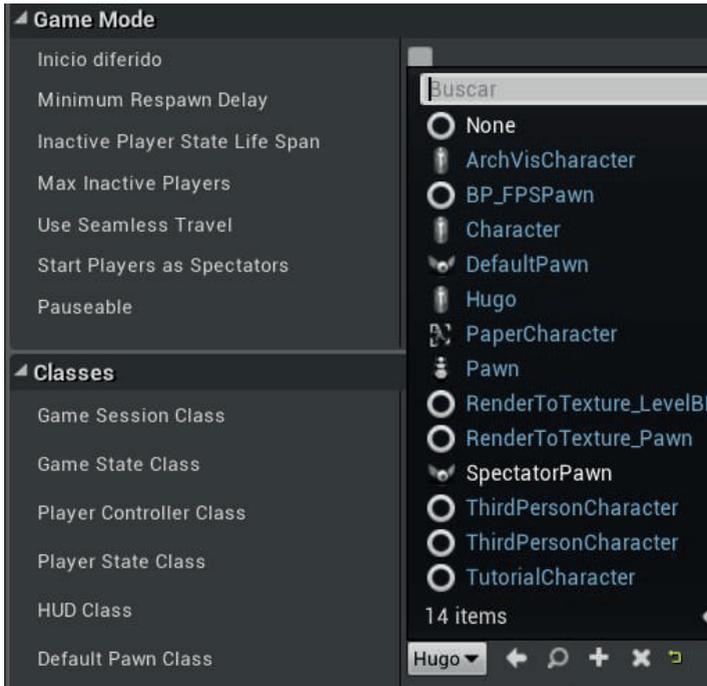
Seleccionamos en la parte superior del proyecto Blueprints/ProjectSettings/GameModeBase/Create/GameMode



Seleccionamos ModoJuego y nombraremos con Modo1 damos clic en aceptar y se habrá guardado en nuestra carpeta "Modo Juego" creada anteriormente. q



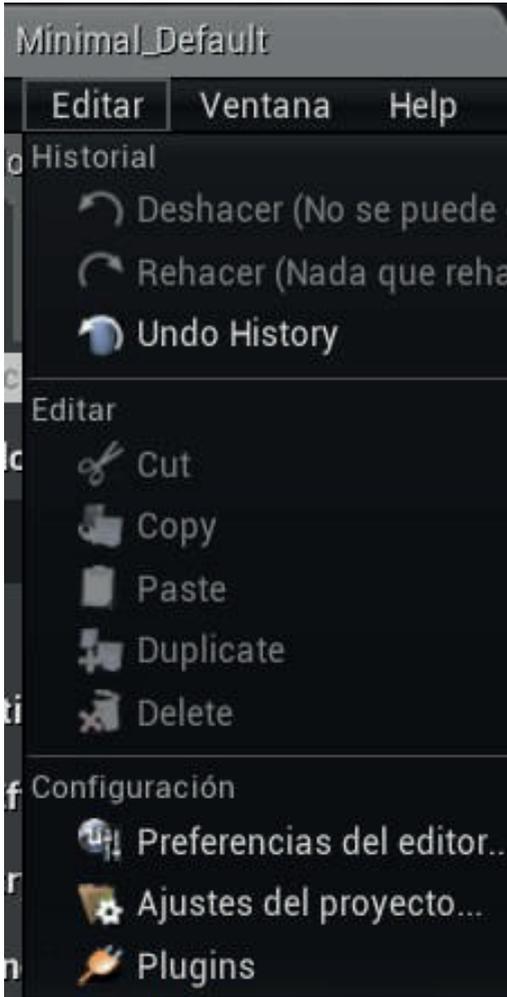
Se abrirá una ventana y nos situamos en la parte inferior derecha Default Pawn Class damos un clic y seleccionamos nuestro personaje "Hugo"



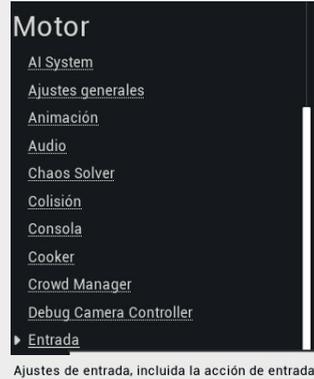
No olvidemos de compilar cada vez que realizamos cambios, cuando existan procesos sin compilar aparecerá un icono con signo de interrogación "?"



Ahora debemos configurar los movimientos del personaje, nos dirigimos a editar/ajustes del proyecto.



Nos dirigimos a la sección Motor/Entrada/Asignación del eje



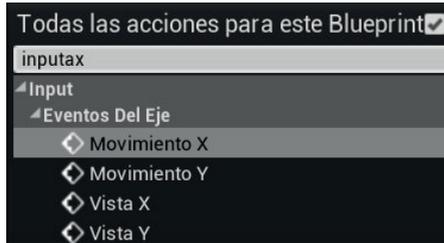
Asignamos los movimientos X, para el sentido norte usaremos la tecla "W" con escala 1.0 y para el sentido sur la tecla "S" con escala -1.0 para el movimiento Y sentido este la tecla "D" con escala 1.0 y para el sentido oeste la tecla "A" con escala -1.0.



Para el movimiento con el mouse asignamos la vista X, para el plano del cursor del mouse X con escala 1.0 y la vista Y para el cursor del mouse Y con escala -1.0. Cerramos la ventana y el proceso se guardará automáticamente.



Abrimos el personaje de "Hugo" y nos dirigimos a gráfico de evento y añadimos con clic derecho InputAxis/MovimientoX.



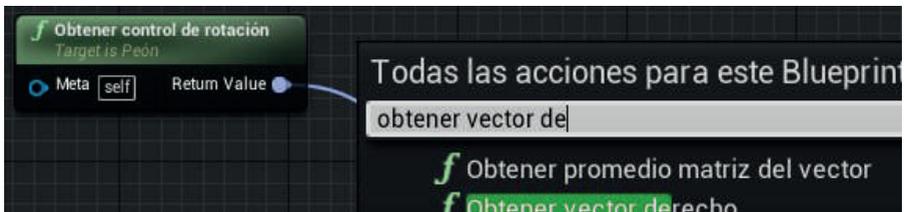
Repetimos el proceso para Movimiento Y, Vista X, Vista Y



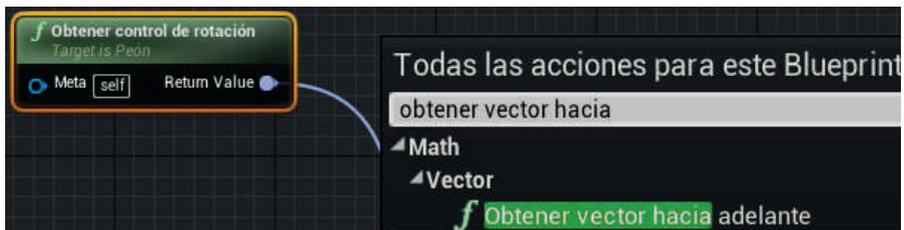
Nos dirigimos en un vacío de la pantalla y damos clic para seleccionar Obtener control de rotación.



Inmediatamente dentro del nodo "Obtener control de rotación seleccionamos Obtener vector derecho".



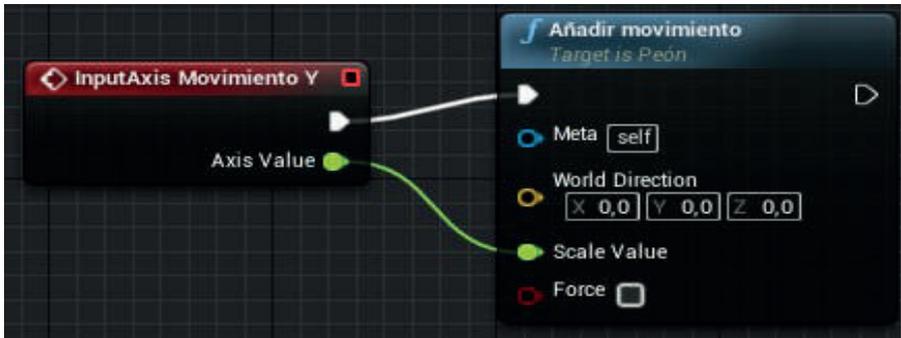
Dentro del nodo "Obtener control de rotación seleccionamos Obtener vector hacia adelante"



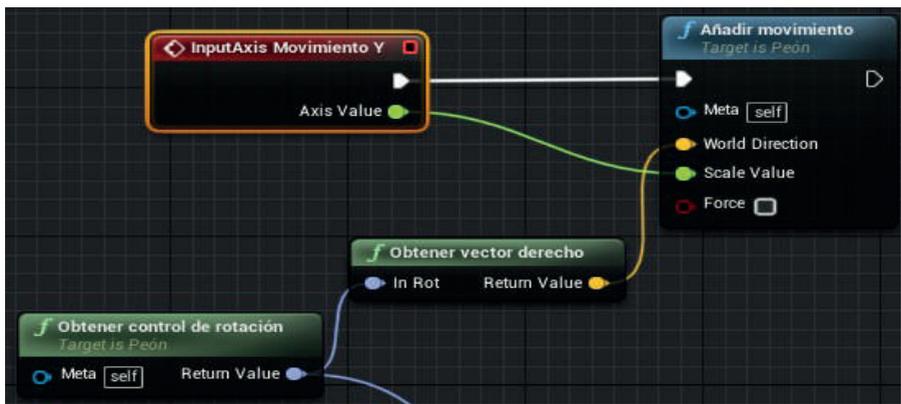
En el nodo Obtener vector derecho seleccionamos la acción "Añadir entrada de movimiento" realizamos el mismo procedimiento para el nodo Obtener vector hacia adelante.



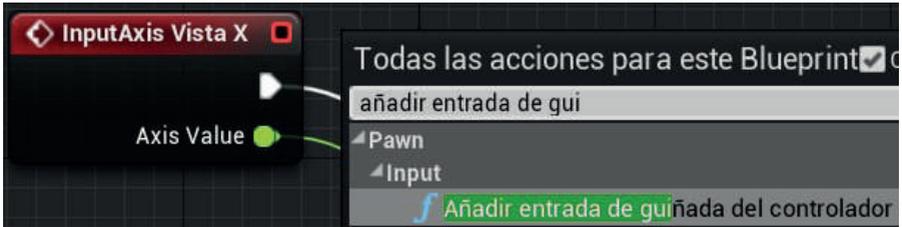
Enlazamos InputAxis Movimiento Y con el nodo Añadir entrada de movimiento.



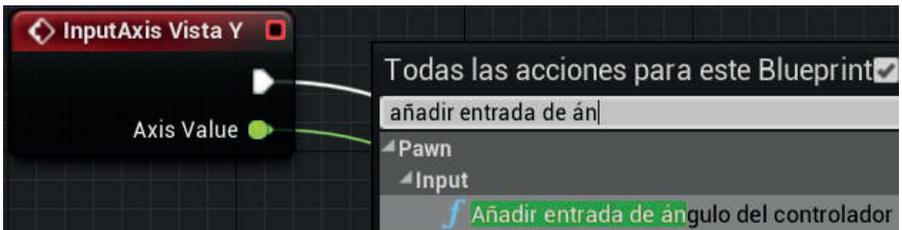
Conectamos el Axis value con el nodo Scale value



Realizamos el mismo procedimiento de conectar nodos, pero con el movimiento en X.



Para el cursor del mouse nos situamos en el nodo InputAxis Vista X a continuación seleccionamos "Añadir entrada de guiñada del controlador"



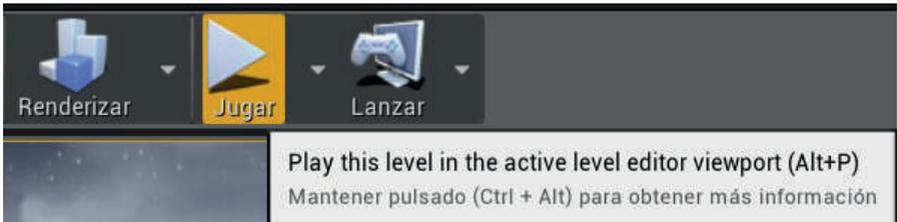
Para el cursor del mouse nos situamos en el nodo InputAxis Vista Y a continuación seleccionamos "Añadir entrada de ángulo del controlador"



En ambos nodos de movimiento del mouse, conectamos los Axis value con la acción "Val", compilamos.



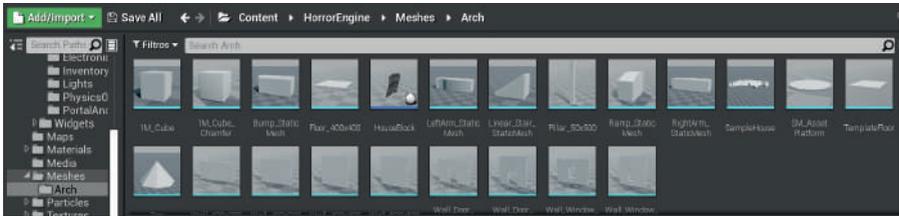
Nos dirigimos a la pantalla principal y damos clic en Reproducir para cargar el juego.



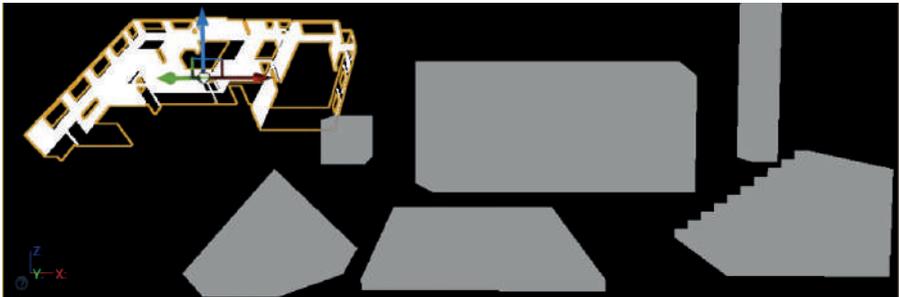
Para el cursor del mouse nos situamos en el nodo InputAxis Vista Y a continuación seleccionamos "Añadir entrada de ángulo del controlador"

_Implementación de Entornos

Dentro de nuestro proyecto encontraremos una carpeta con el nombre Meshes o Prefabs donde tendremos figuras prefabricadas que podremos implementar en la creación de nuevos entornos.



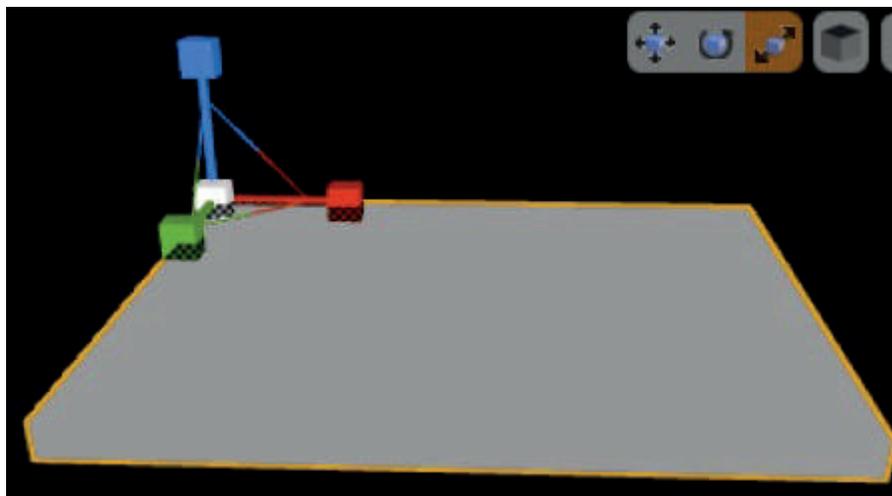
Ejemplo: pisos, paredes, entradas puertas y ventanas, escaleras y figuras geométricas.



Para crear los cimientos podremos utilizar Floor_400x400 para simular la base, recomendamos en el casillero ubicación poner las coordenadas X, Y, Z en la posición 0° para tener una medida exacta en la cual basaremos todo el desarrollo del videojuego.

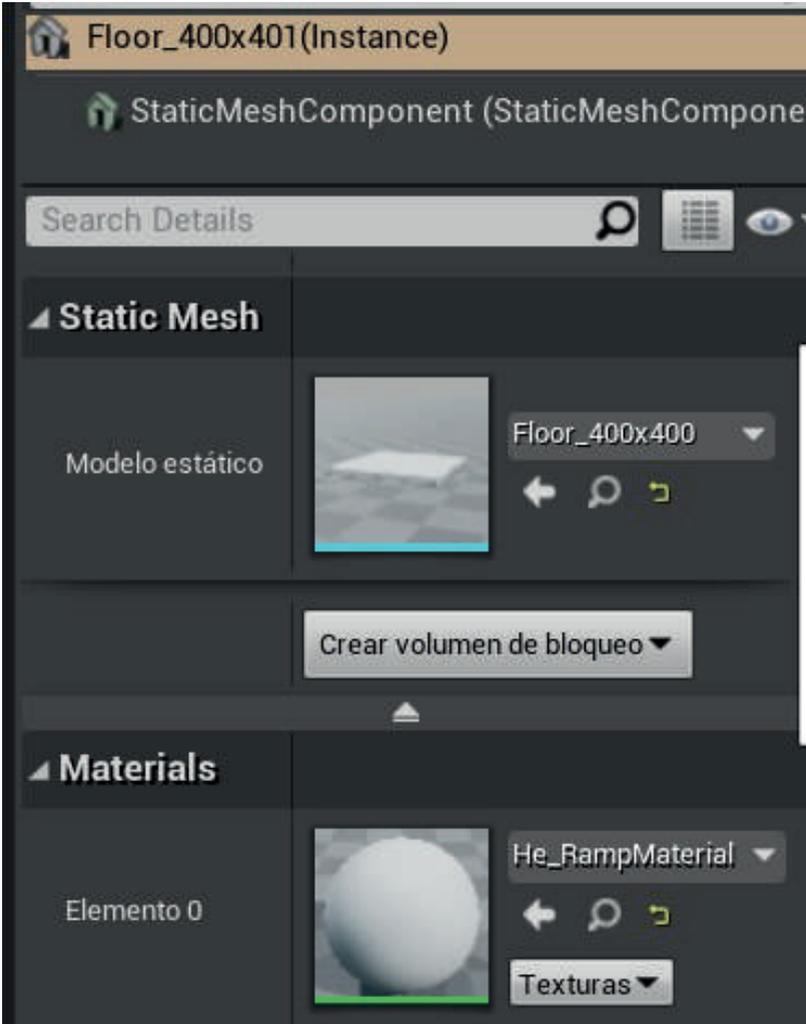


Si consideran que es una medida pequeña se dirigen a la opción Escala y cambiar las dimensiones de longitud.

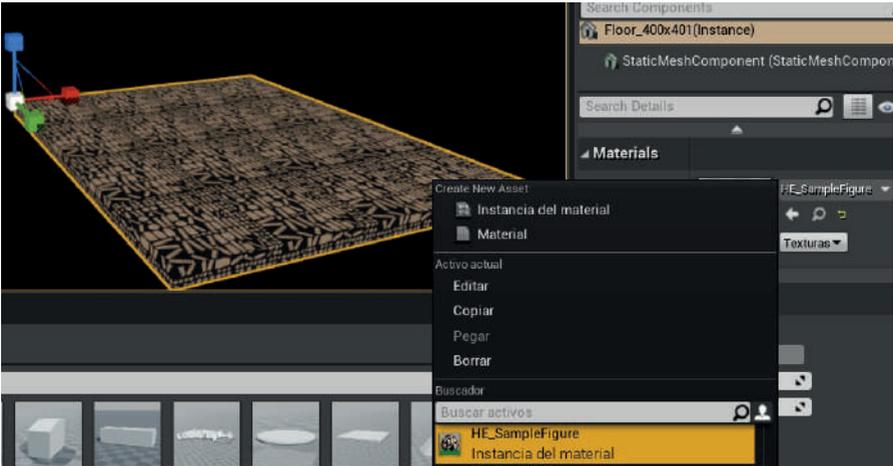


_ Agregar Texturas

Por defecto tendremos prefabs sin materiales de texturas, por suerte Unreal cuenta con una variedad de texturas que podremos usar además que en la tienda encontraremos librerías con más contenido. Nos dirigimos Ajustes del mundo y buscamos "Materials"



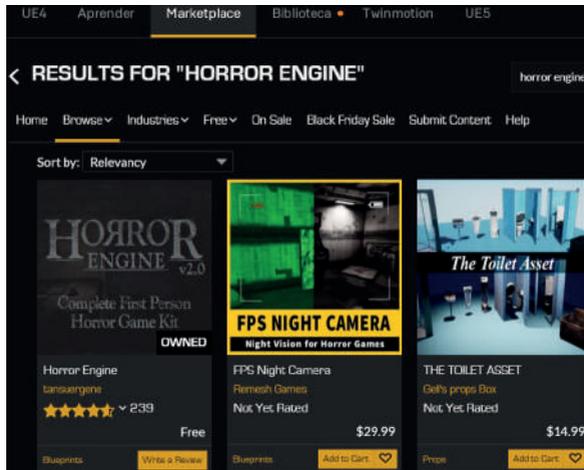
Seleccionamos en la casilla y se busca la textura que implementarán.



Nos dirigimos a la pantalla principal y damos clic en Reproducir para cargar el juego

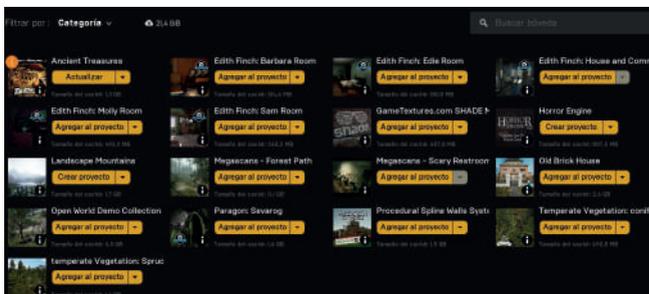
_ Descargas de Assets

Nos situamos en la aplicación de Epic Games, damos clic en Marketplace y buscamos el Assets con el nombre "HorrorEngine", posteriormente descargamos el contenido y entramos en el proyecto.



_ Crear Movimientos del personaje

Realizamos los mismos pasos para descargar diferentes Assets, dependiendo del uso que necesitemos encontraremos una gran variedad, para el desarrollo de esta guía usaremos los siguientes: Edith Finch: Molly Room, Edith Finch: Barbara Room, Edith Finch: Sam Room, Edith Finch: Edie Room, todos son contenido de descarga gratuito en la tienda de Epic Games.



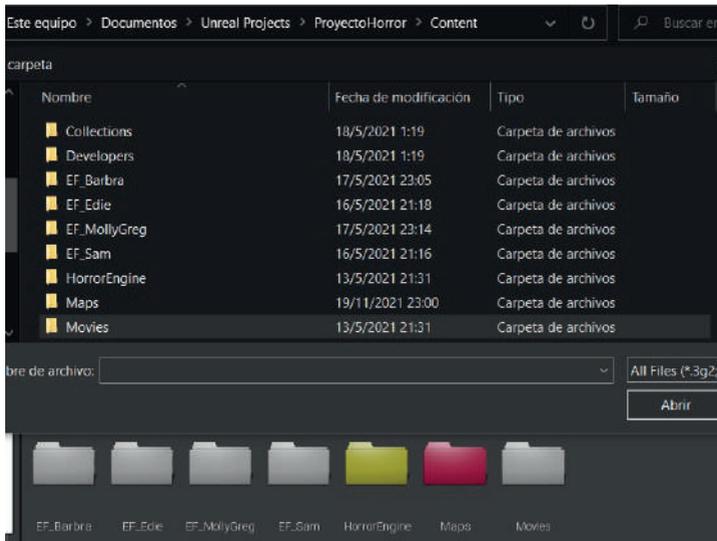
_ Importar Assets

Después de descargar debemos importarlos dentro del proyecto, nos dirigimos a la carpeta "Content" y damos clic derecho Importar

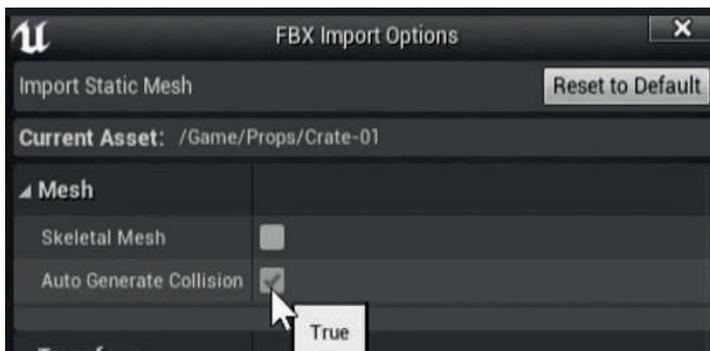




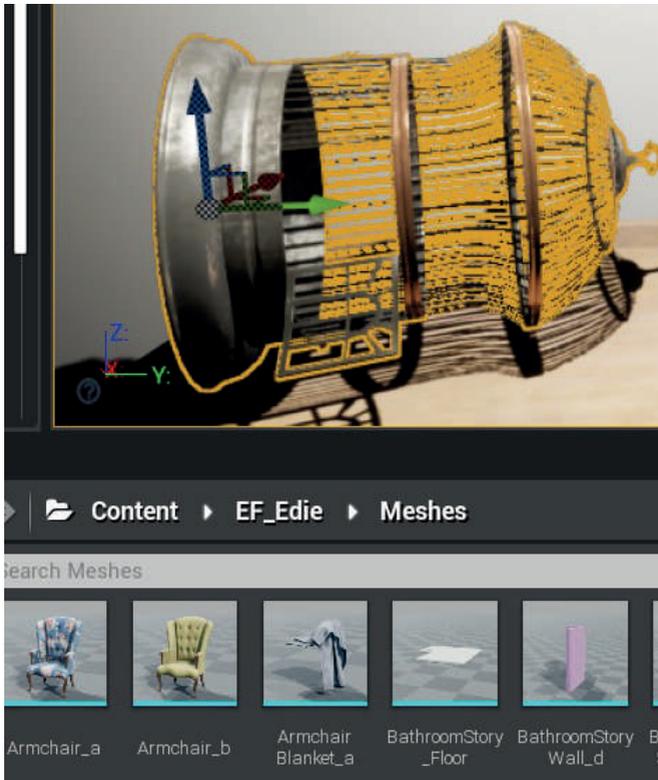
Se mostrará una ventana donde debemos buscar nuestro contenido descargado y una vez encontrado debemos seleccionarlo y dar clic en “Abrir”.



Dentro de la ventana verificamos que la casilla “Auto Generate Collision” está marcada como verdadero, esto permitirá que los objetos importados tengan una collision generada automáticamente lo que permitirá la interacción con el entorno. Daremos clic en “Input All” y el proceso empezará a realizarse automáticamente.



Buscamos la carpeta importada dentro del proyecto seleccionamos y arrastramos dentro del proyecto.

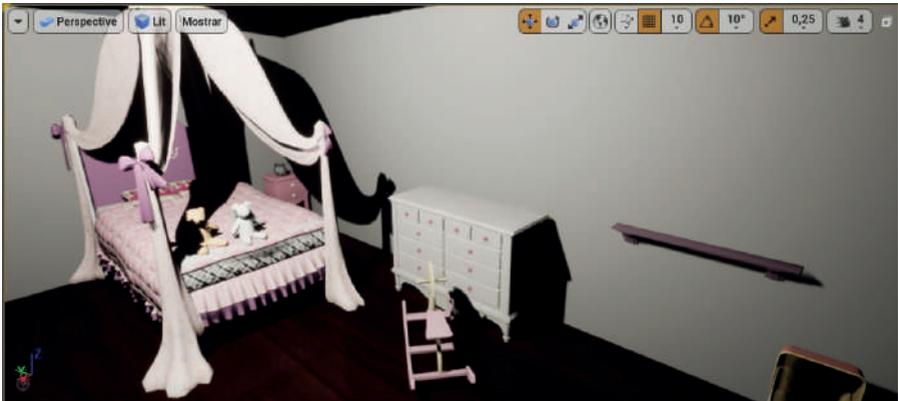


Para cada objeto importado tenemos la opción mover, rotar y escalar mediante las dimensiones tridimensionales X, Y, Z.



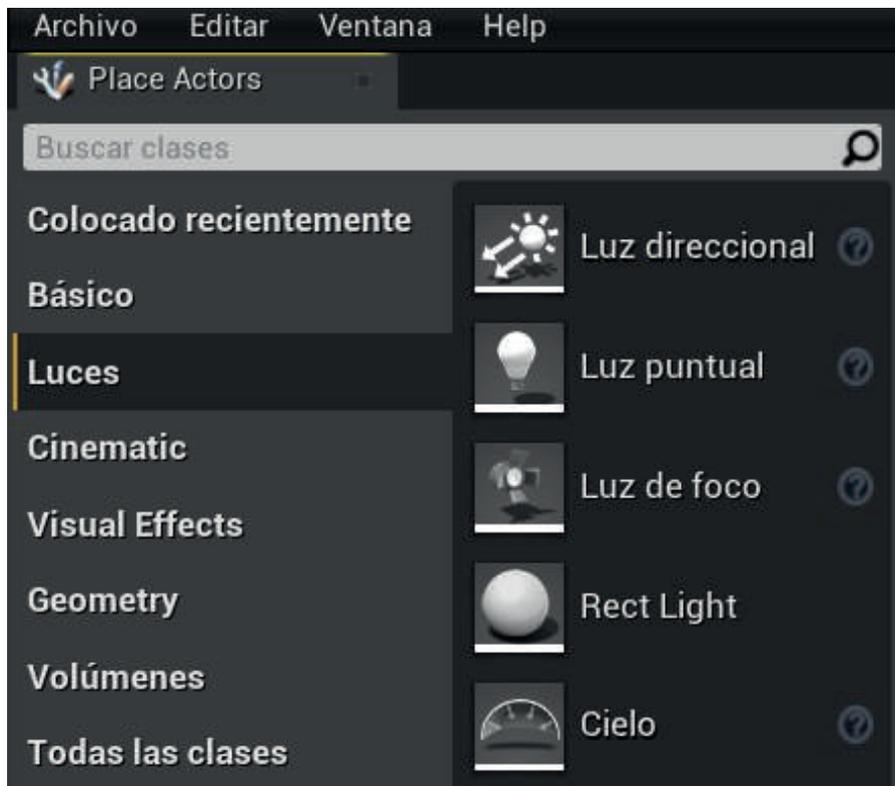


Mediante los Assets podemos crear entornos específicos para cada escenario; cocina, sala, habitaciones, baños, etc.

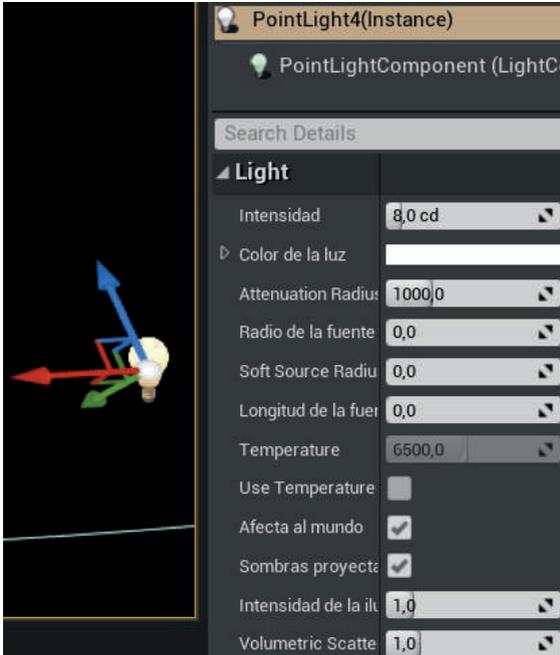


_ Iluminación de entornos

Dentro de Place Actors seleccionamos "Luces" y nos aparecerán diferentes opciones de iluminación.

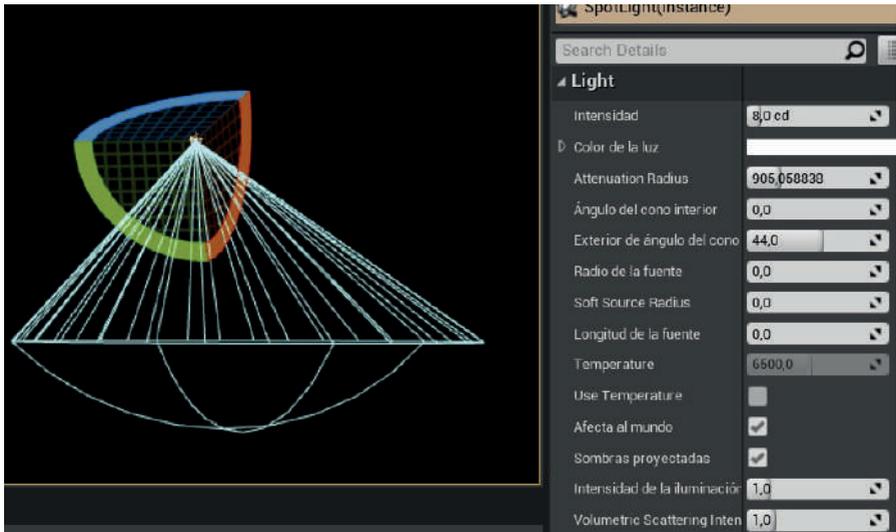


Luz direccional: nos permite tener el control de enfocar la dirección de la luz a zonas específicas permitiendo generar sombras.

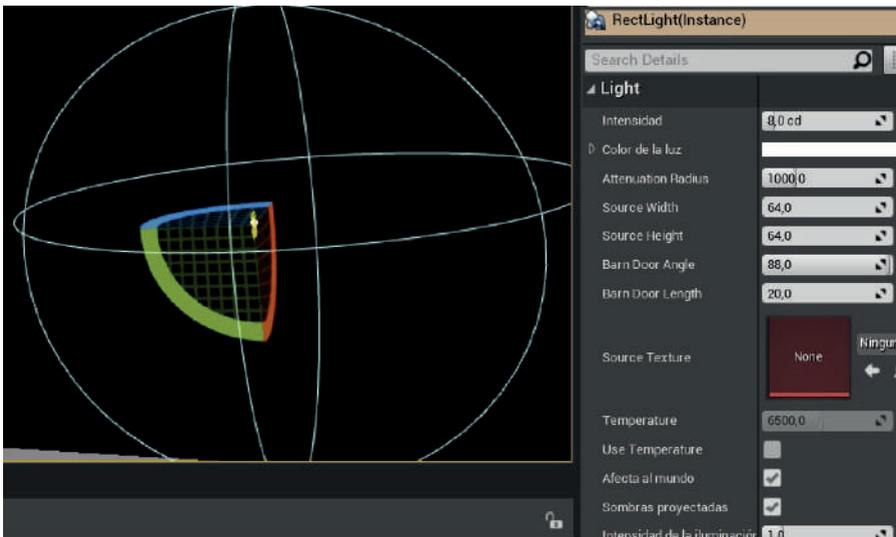


Luz puntal: nos permite proyectar luz dentro de un rango que podremos medir según las necesidades

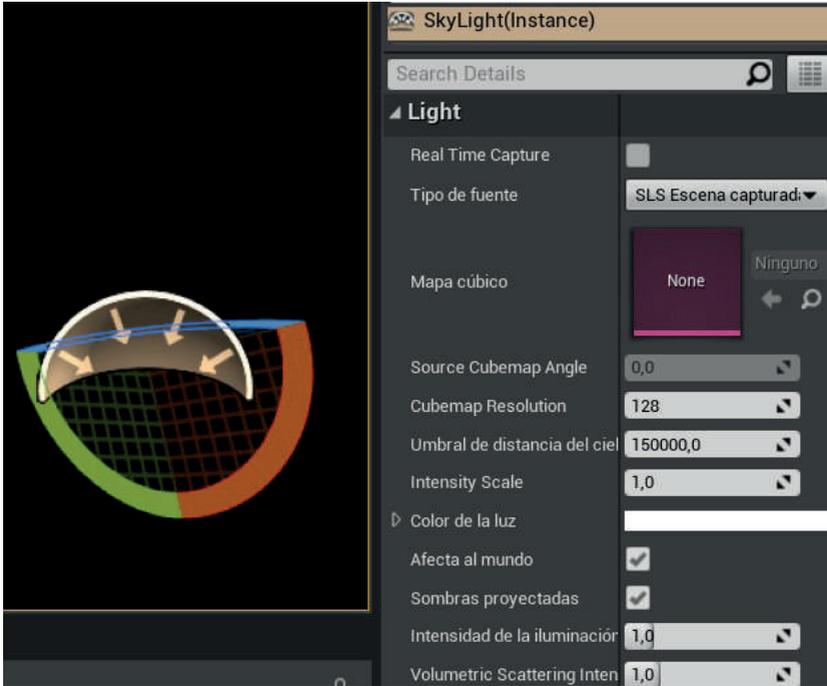
Luz de foco: Emite luz desde un solo punto



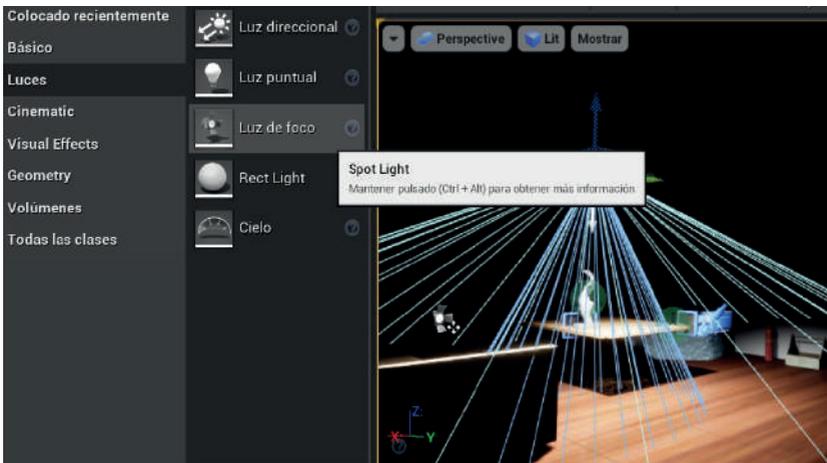
Rect light: emite una luz con altura y anchura definida, simulando iluminación de habitaciones.



Cielo: cubre con una luz todo el horizonte del proyecto en cual se desarrolla.

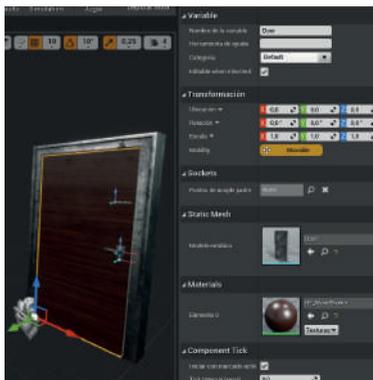


Para agregar iluminación en diferentes escenarios debemos situarnos en Place Actors escoger cuál iluminación es la idónea para nuestro escenario y arrastrar hasta el proyecto



_ Usar Assets (Puertas)

Cuando descargamos Assets la mayoría traen funciones programadas que agilizan el proceso de desarrollo de un videojuego, así mismo tendremos la opción de modificar de acuerdo con nuestras necesidades.



A continuación, demostraremos las funciones que podemos agregar a los Assets.

Es recomendable nombrar cada Assets según el escenario donde se encuentre, de esta manera resulta más fácil identificar los procesos, recordemos que podemos implementar más de una puerta en todo el proceso de desarrollo. Otra de las funciones es que podemos elegir si la puerta permanece abierta, bloqueada, abrir con llave, dimensiones de longitud, agregar texturas para cambiar la apariencia, todo esto es posible en las configuraciones que aparecen en la parte inferior derecha al momento de seleccionar la puerta.



Todas las funciones que se acaban de impartir funcionan para la categoría de puertas incluyendo los siguientes Assets.



Las puertas electrónicas funcionan con tarjetas de accesos o contraseñas, para configurar un tablero con la puerta debemos ir a la carpeta Content/HorrorEngine/Blueprints/UsableAssets/Electronics



Agregar el Asset Security Control luego lo seleccionamos y nos dirigimos a las configuraciones. Dentro de Password podemos ingresar la contraseña en dígitos numéricos, y también podemos configurar una alarma después de una cantidad de intentos fallidos además de la duración del sonido.



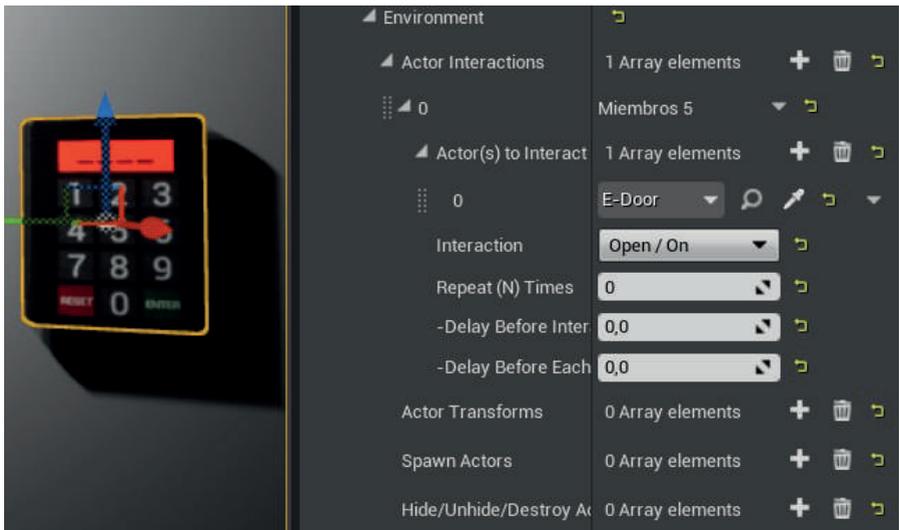
Para el personal que recién comienza a crear un videojuego recomendamos no cambiar los audios integrados para cada acción presentada en los Assets.



Para agregar el evento cuando ingresemos la contraseña la puerta pueda abrirse debemos crear un Array elements, dentro de Trigger Interaction nos aseguramos de que permanezca "Unlock" significa que cuando el personaje llegue a la puerta en primer instancia aparece bloqueada y requiere ingresar una contraseña.



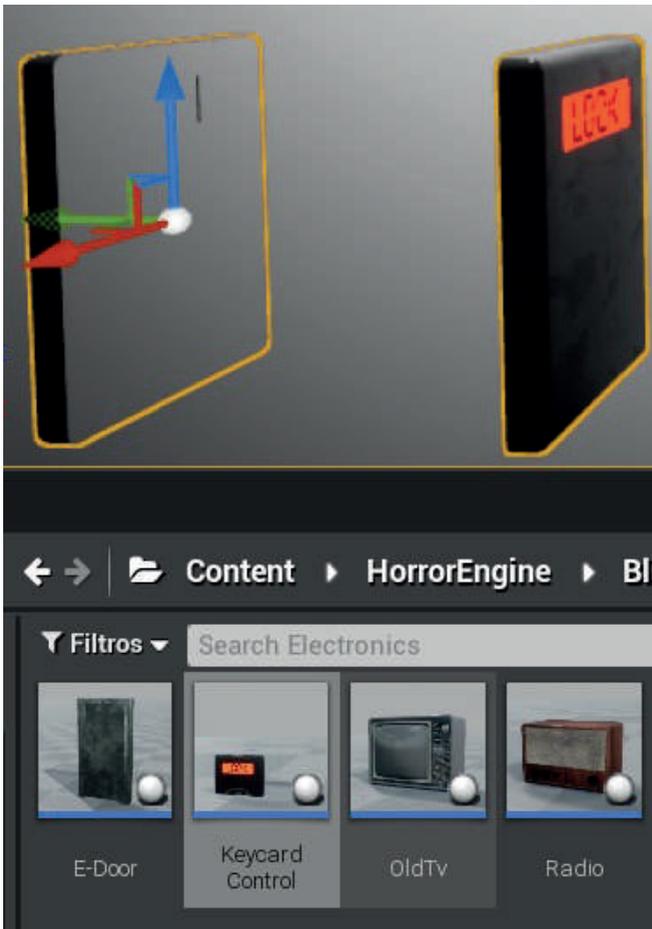
Nos dirigimos Actor Interactive y creamos un array donde ingresaremos la puerta (E-Door) que va a interactuar con el panel numérico, solo demos agregar el nombre y en la casilla de Interacción damos agregamos el evento Open que nos permitirá desbloquear la puerta.



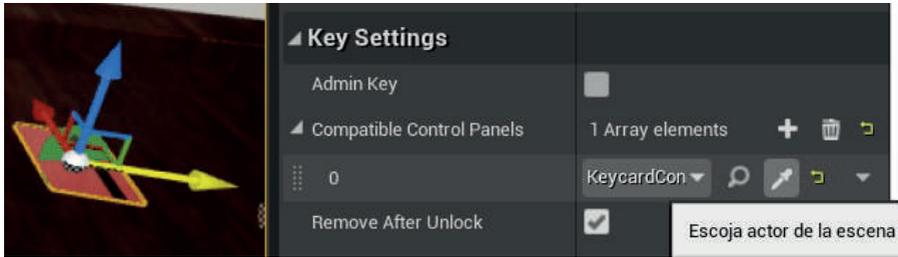
Ahora tenemos el método por tarjeta de acceso, para configurar este evento debemos situarnos en Content/ HorrorEngine/ Blueprints/ UsableAssets/ Inventory



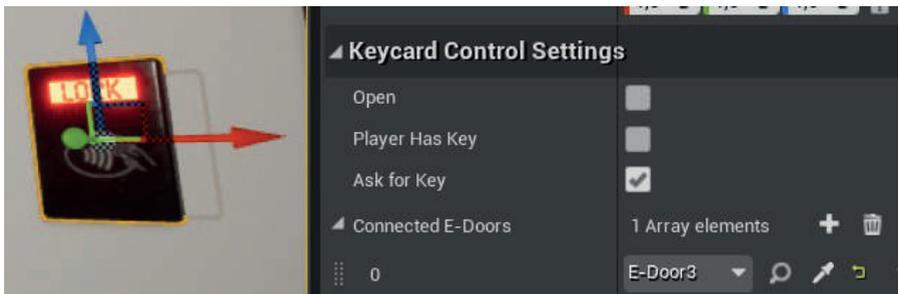
Seleccionamos Keycard Control, como podrán observar tenemos dos lados del control de seguridad que nos permite trabajar en ambos lados de la pared.



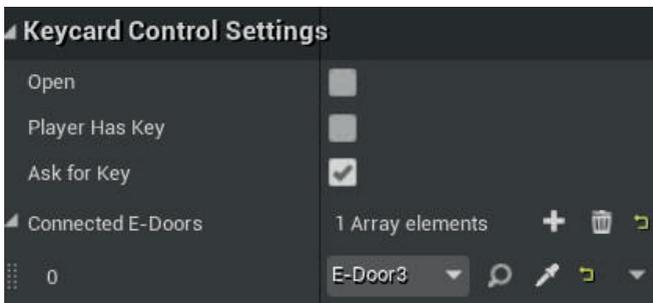
Para conectar la tarjeta de acceso con panel de seguridad debemos situarnos en la tarjeta y dar clic en el gotero.



Luego procedemos a seleccionar el control de acceso donde vamos a realizar la acción, luego volvemos a situarnos en el gotero y seleccionamos la puerta electrónica "E-Door3"

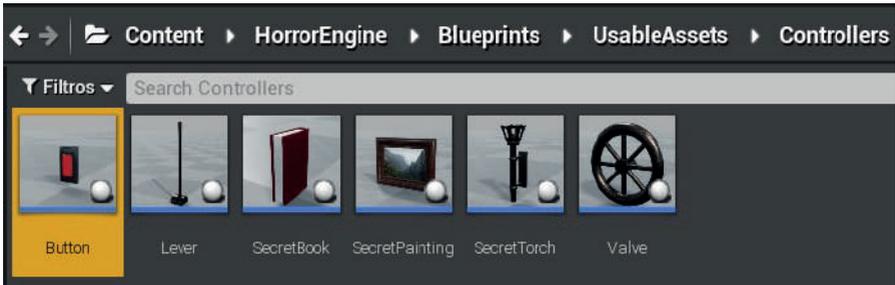


Activamos la casilla "Ask for key" de esta manera damos la acción de pedir una llave de acceso, en este caso una tarjeta. Nos situamos en Connected E-Doors creamos un nuevo array element y seleccionamos la puerta en este caso "E-Door3"



_ Usar Assets (Iluminación)

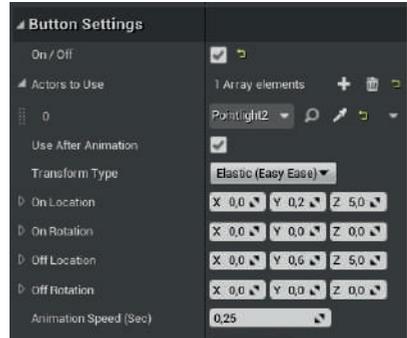
Para crear una iteración más realista tenemos botones para encender o apagar la iluminación de escenarios para ello debemos situarnos en la carpeta Content/ HorrorEngine/ Blueprints/ UsableAssets/ Controllers seleccionamos "Button" y agregamos a nuestro proyecto



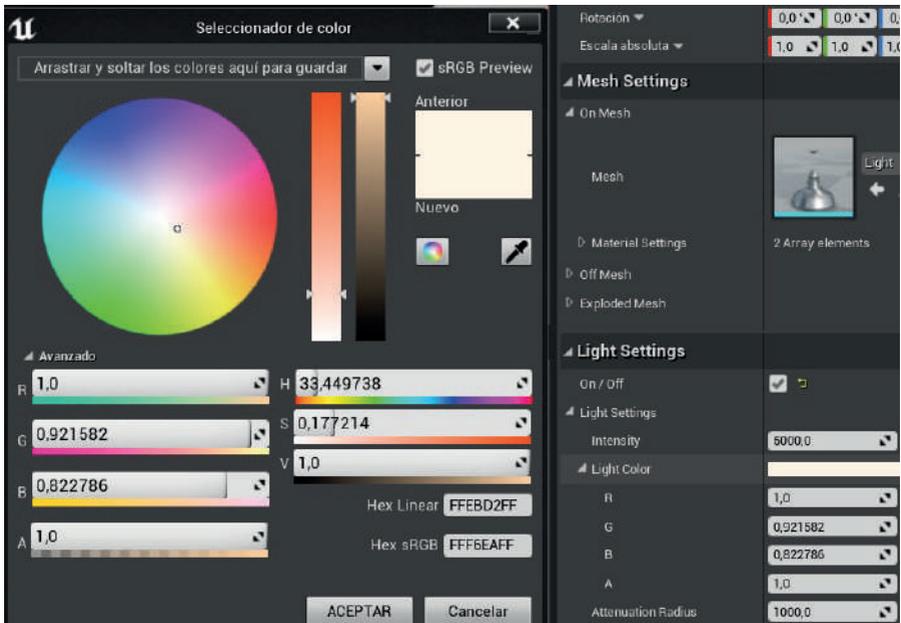
Mediante las opciones de mover, rotar y escalar podemos situar el Assest según el criterio propio.



Para este escenario en particular mantendremos la casilla activada de iluminación luego nos dirigimos a crear un nuevo array elements y seleccionaremos el foco de iluminación que en este caso sería "Pointlight2" activamos la casilla Use after animation para generar efectos animados de encendido.



Dentro de las configuraciones podemos graduar la intensidad y el color de la luz, debemos situarnos en Light settings/ Light color al seleccionar se abrirá una ventana con una gama extensa de colores, de esta forma podemos agregar diferentes entornos de iluminación.



_ Interactuar con objetos

Nos dirigimos a la carpeta Content/horror Engine/Blueprints/UsableAssets/Inventory encontraremos Assets que podemos utilizar para diversas acciones, la mayoría de Meshes poseen las mismas configuraciones, facilitando el uso de las herramientas.



Agregamos Flashlight dentro proyecto al seleccionar podremos editar las configuraciones predeterminadas como:

Takeable: nos permite llevar el objeto a nuestro inventario.

Examine: nos permite examinar el objeto.

Inspection animation: genera la animación de sostener el objeto.

Inspection animation speed: tiempo de la animación cuando interactuamos con el objeto.

Inspection shadow: permite generar sombras.

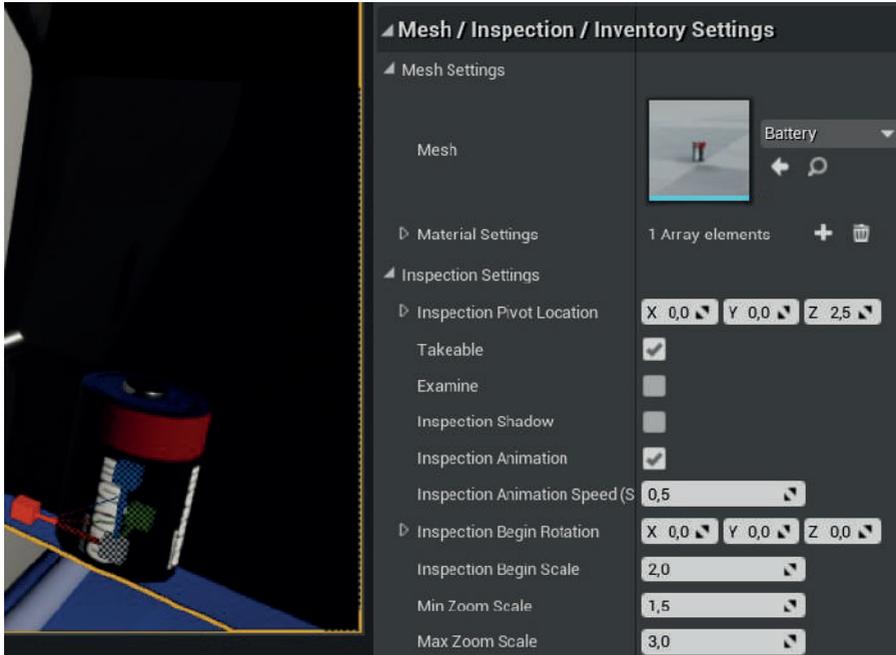
Inspection begin rotation: crear la ubicación inicial y la rotación cuando se generan por primera vez un objeto.

Inspection begin scale: define el tamaño a escala del objeto.

Min/Max zoom scale: define el tamaño del objeto.

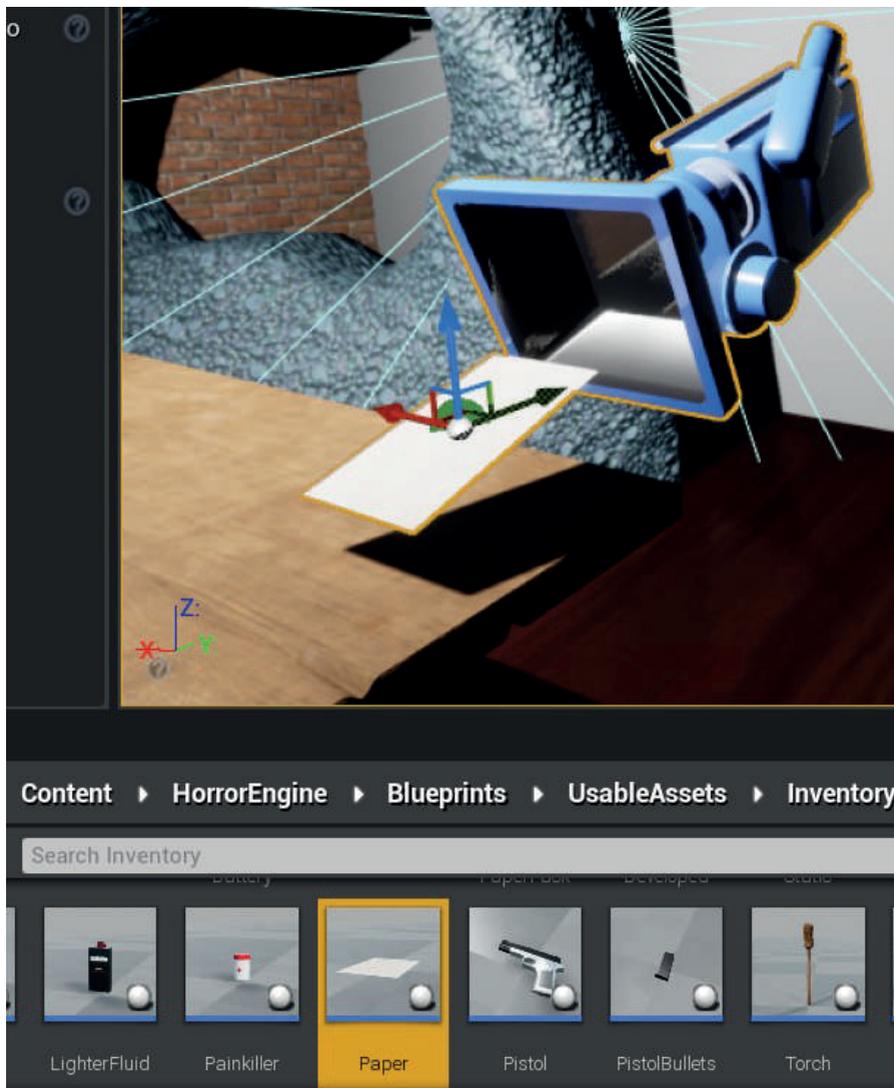


Los objetos como linterna y videograbadora tienen configurado un tiempo de función lo que permite usar Assets cómo Battery para agregar un nuevo periodo de tiempo. Lo que nos permite generar la animación utilizando las baterías para recargar los objetos. Para lograr esta animación debemos situarnos en Max slide value y agregar un valor equivalente a tiempo

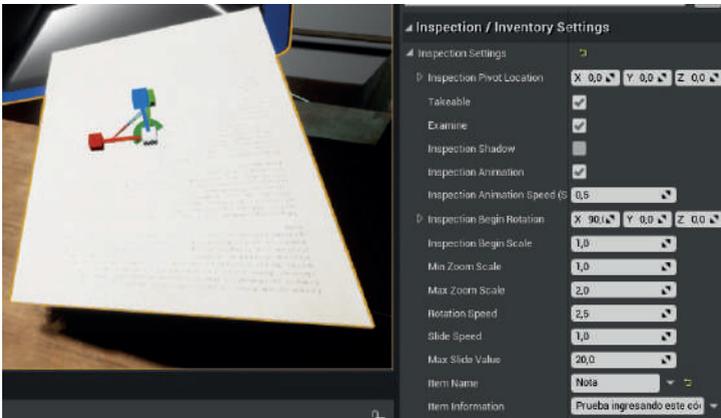


Inspeccionar Objetos

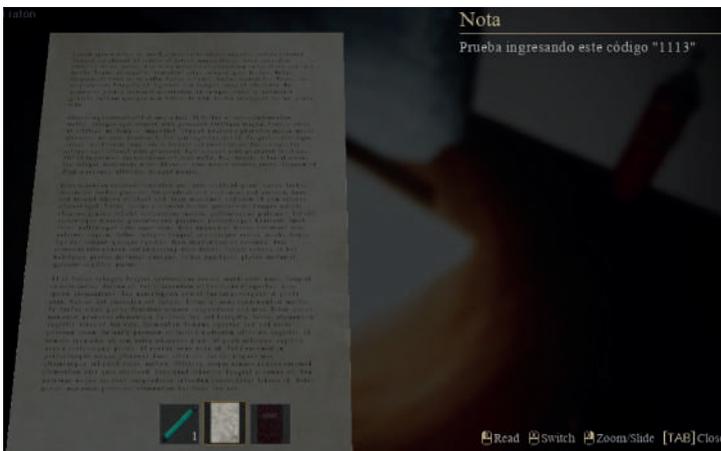
Podemos implementar esta acción para guardar mensajes, pistas, contraseñas



Seleccionamos el papel y nos situamos en las configuraciones "Inspection Settings" nos situamos en la casilla "Item Name" para escribir un título en la hoja, para ingresar una descripción escribimos el mensaje, en este caso la contraseña dentro de la casilla "Item Information"

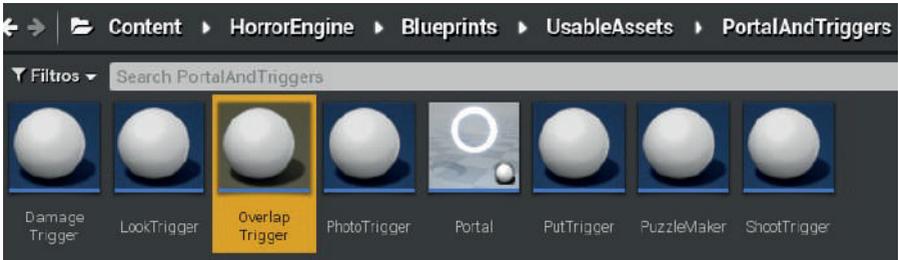


Una vez que procedemos a proyectar el juego nos situamos en la hoja dando un clic se llevará a nuestro inventario. Accedemos pulsando la tecla "Tab" y seleccionamos la hoja.

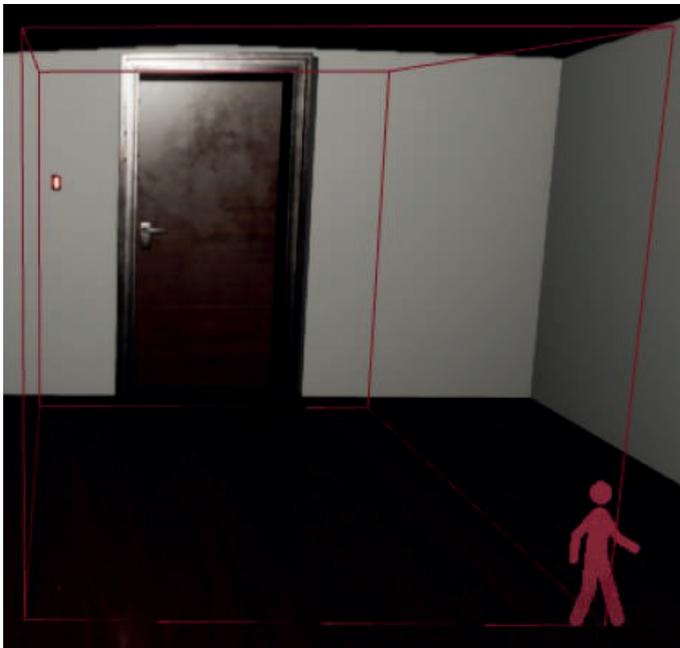


_ Implementación de triggers

Nos situamos en la carpeta Content/ HorrorEngine/ Blueprints/ UsableAssets/ PortalTriggers luego seleccionamos el Blueprints class "Overlaptrigger"



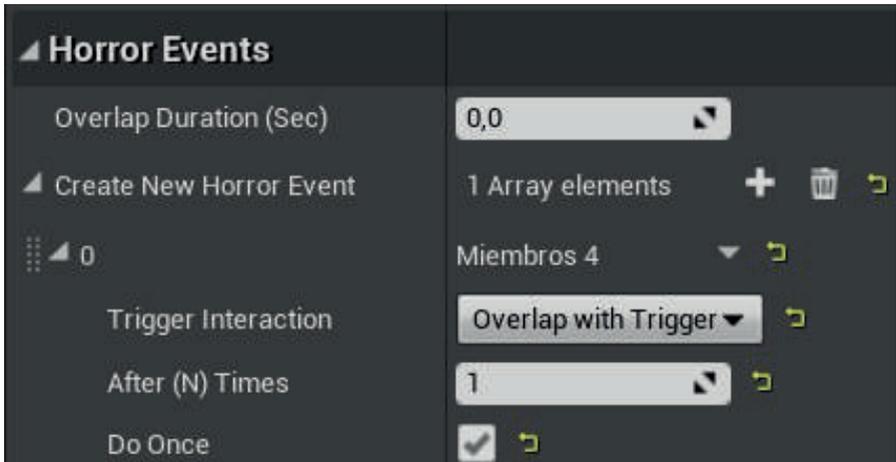
Generará un box collision que ejecutará una acción que el jugador realizará cuando entre en contacto con la caja de colisión.



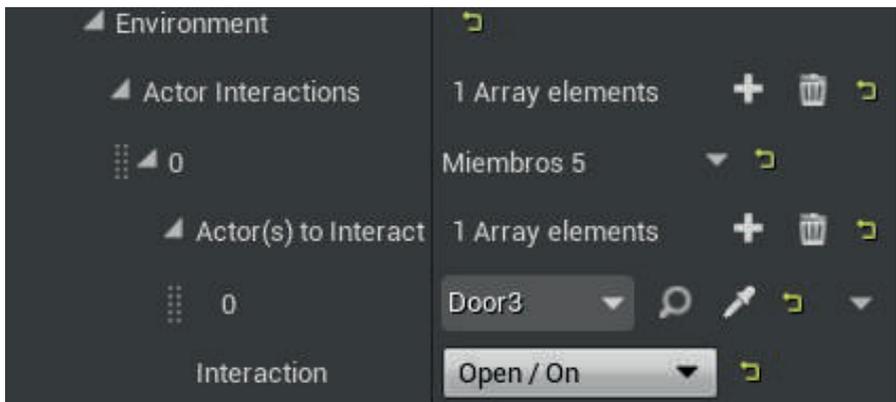
Dentro de caja de configuraciones creamos un nuevo evento con los siguientes parámetros;

Triggers Interaction: Overlap with trigger se ejecutará cuando el jugador colisione con el triggers

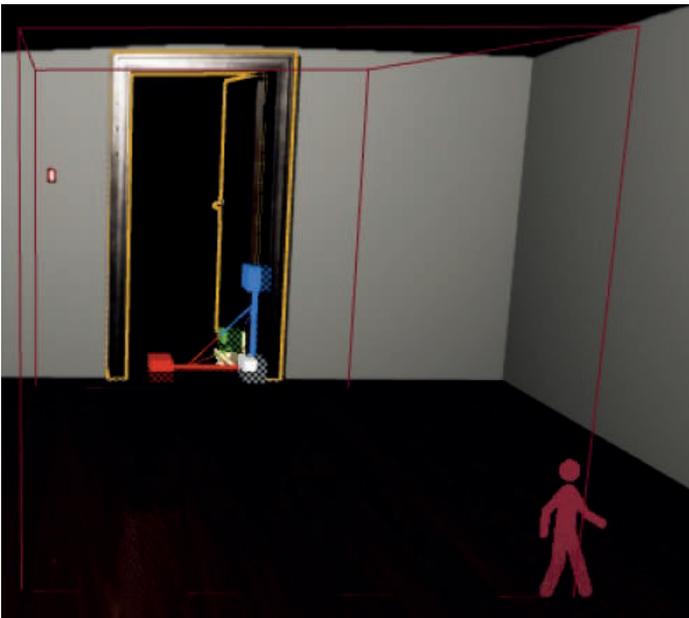
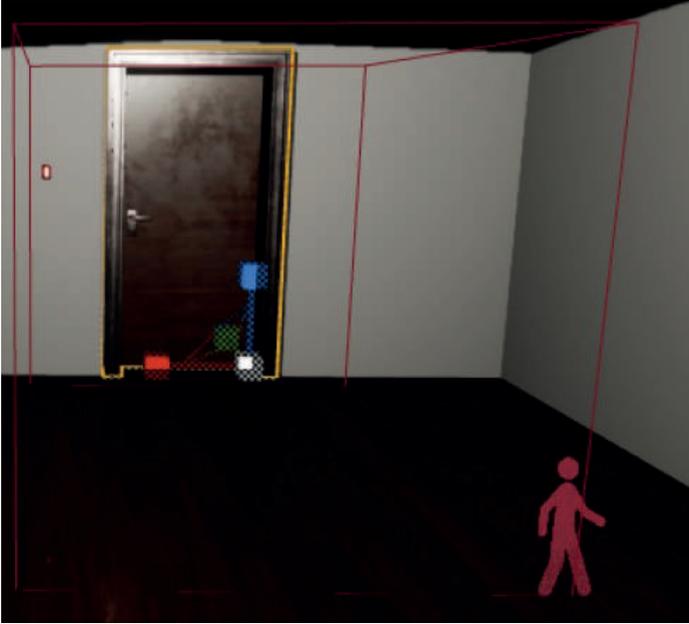
After (N) time: ejecutar la acción la cantidad de veces que necesitemos.



Creamos un array elements y dentro de actor seleccionamos la puerta con la que ejecutaremos esta acción

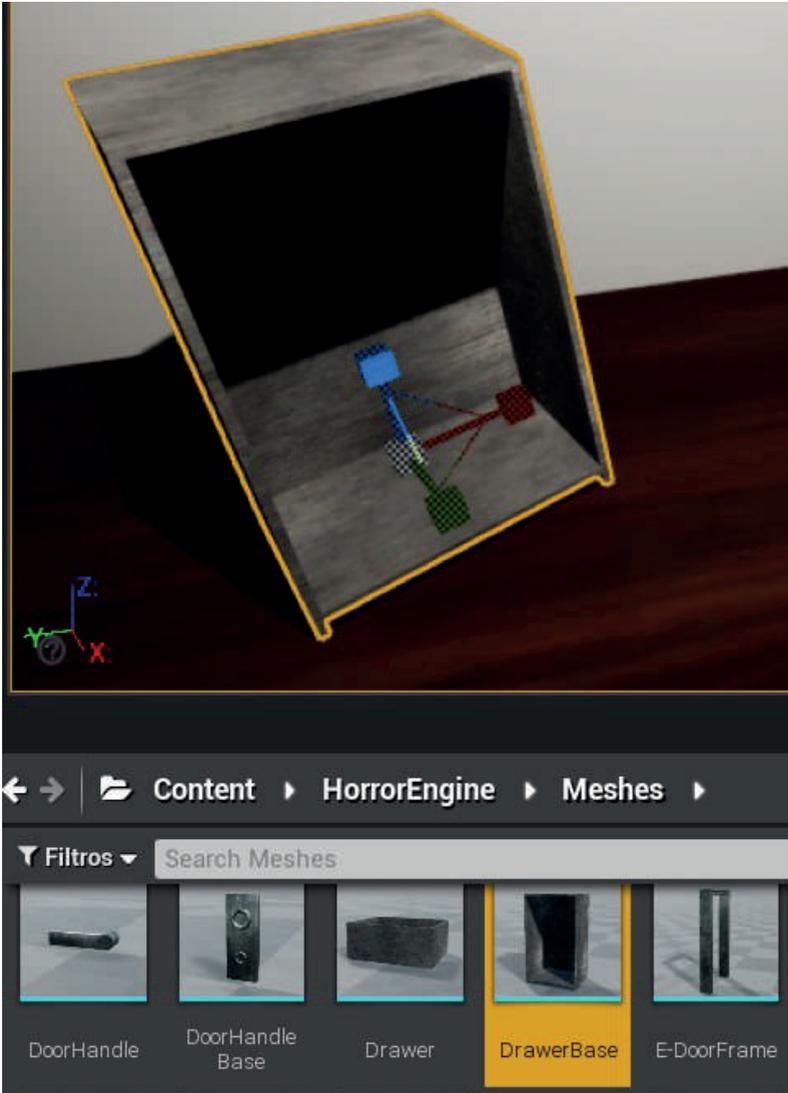


Como resultado cada vez que el jugador pise la colisión se ejecutará la acción de abrir o cerrar puertas.

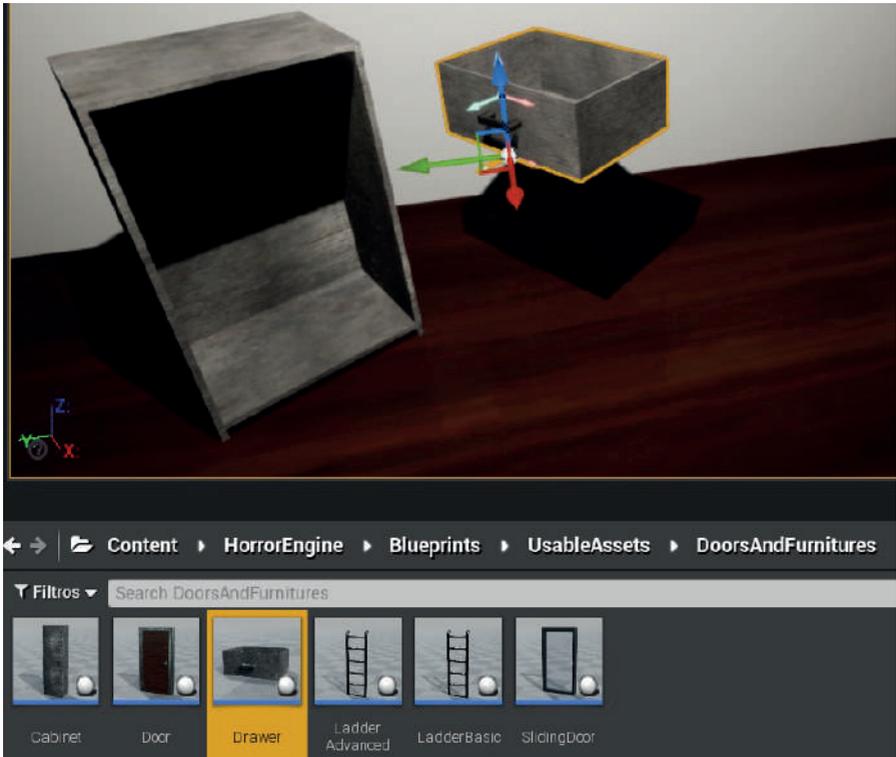


_ Agregar material

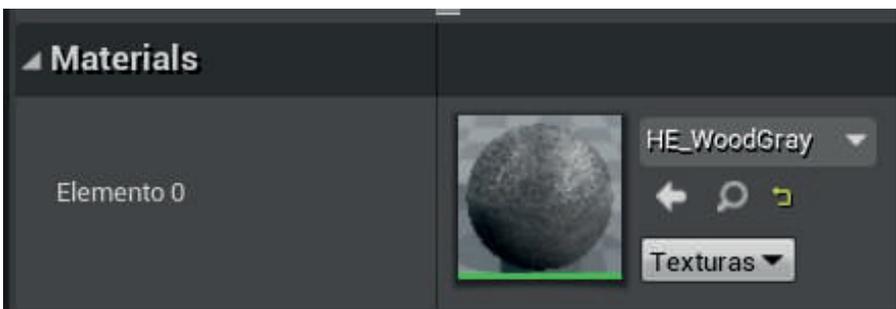
Para cambiar el material de la cajonera debemos situarnos en la carpeta Content/HorrorEngine/Meshes y seleccionamos "DrawerBase".



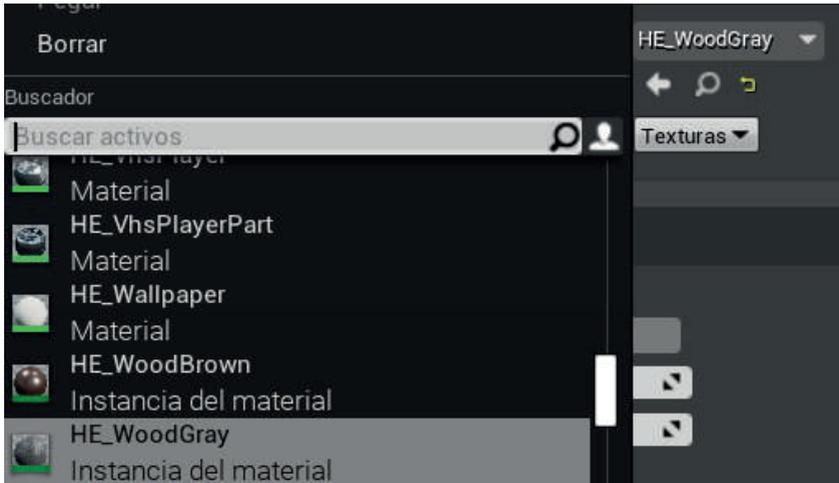
Para utilizar los cajones debemos situarnos en la carpeta Content/HorroEngine/blueprints/UsableAssets/DoorsAndFurnitures y seleccionamos "Drawer".



Una vez que tenemos la estructura en el proyecto debemos darle clic y situarnos en las configuraciones, en la sección "Materials".



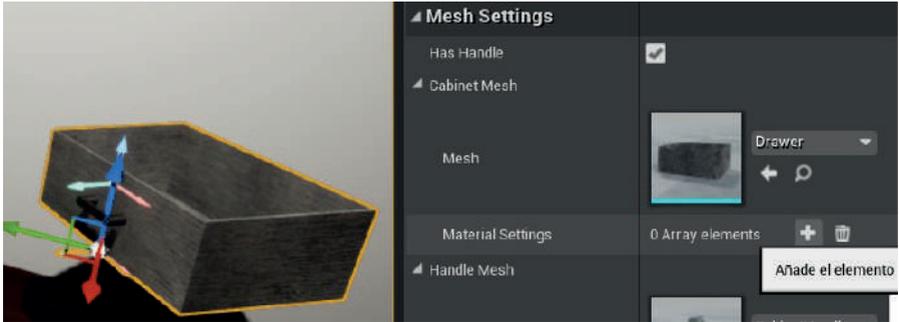
Dando clic en la casilla, nos abrirá una nueva ventana con todos los materiales disponibles de este proyecto que podemos utilizar



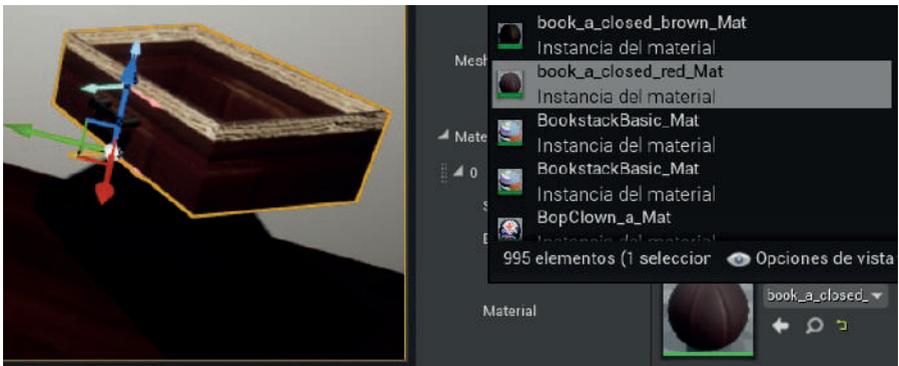
Seleccionamos el material CuttingBoard_a_Mat, automáticamente el material cambiará de apariencia.



Para Drawer debemos seleccionarla y situarnos en las configuraciones "Material Settings" agregamos un nuevo array elements.



Aparecerá una nueva ventana de configuraciones y seleccionamos el material "book_a_closed_Mat", realizando el mismo procedimiento anterior mencionado.



ISBN: 978-9942-33-627-9



9 789942 336279

compAs
Grupo de capacitación e investigación pedagógica

   @grupocompas.ec
compasacademico@icloud.com