

# Programación Unity3D



**UNIDAD 2: Programación de videojuegos 3D**

**Roberto Rodríguez Ortiz**

Versión 1.2 Noviembre 2020

Este documento se publica bajo licencia Creative Commons “Reconocimiento-CompartirIgual (by-sa)”



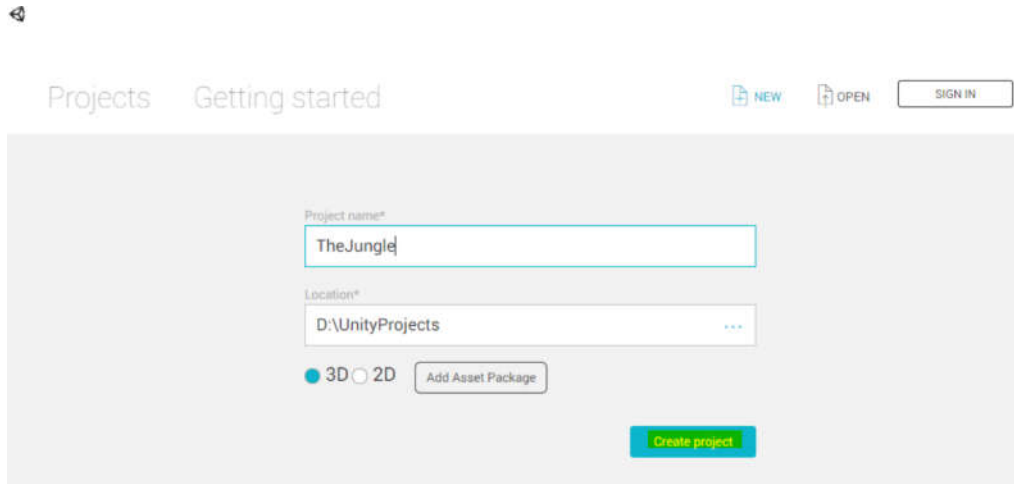
## Tabla de contenido

1.	Introducción .....	4
2.	Creación del entorno.....	5
2.1.	Añadiendo un terreno .....	5
2.2.	Configuración básica de un terreno .....	5
2.3.	Herramientas de terreno.....	7
2.4.	Definiendo una textura base.....	9
2.5.	Geometría del terreno .....	11
2.6.	Añadiendo vegetación .....	12
2.7.	Añadiendo un cielo .....	14
2.8.	Añadiendo niebla.....	16
2.9.	Color de ambiente.....	16
2.10.	Añadiendo luces .....	17
3.	Añadiendo un controlador en primera persona .....	19
3.1.	Personalizando el controlador.....	20
3.2.	Editando el Script .....	22
3.3.	Últimos detalles.....	23
4.	Incluyendo objetos en la escena .....	23
5.	Fuentes.....	24

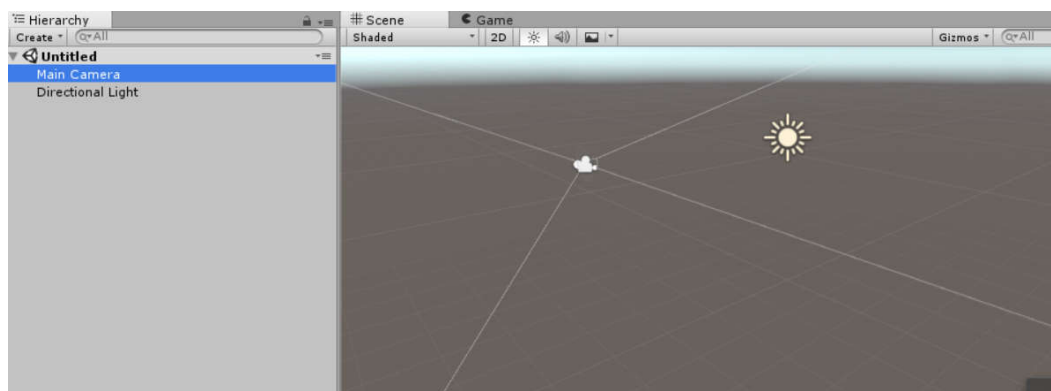
## 1. Introducción

Unity3D como su nombre indica está especialmente diseñado para trabajar en un entorno 3D, a través de esta unidad vamos a iniciarnos en el desarrollo de un escenario tridimensional en el cual podamos crear cualquier tipo de juego o experiencia interactiva.

Crearemos un proyecto vacío llamado *TheJungle*, indicando que será en 3D:



En este caso tendremos una cámara con vista en perspectiva y una luz direccional:



## 2. Creación del entorno

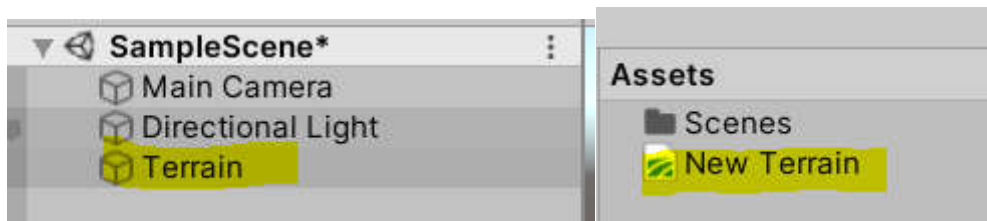
### 2.1. Añadiendo un terreno

Ya tenemos nuestro proyecto y la escena en blanco, ahora vamos a añadir un terreno para empezar a crear nuestro nivel. Mientras creamos el terreno veremos algunas características y opciones básicas de los terrenos.

Unity maneja los terrenos de la misma forma que muchos otros motores, como una [malla](#) plana que podemos texturizar y esculpir sin salir del editor.

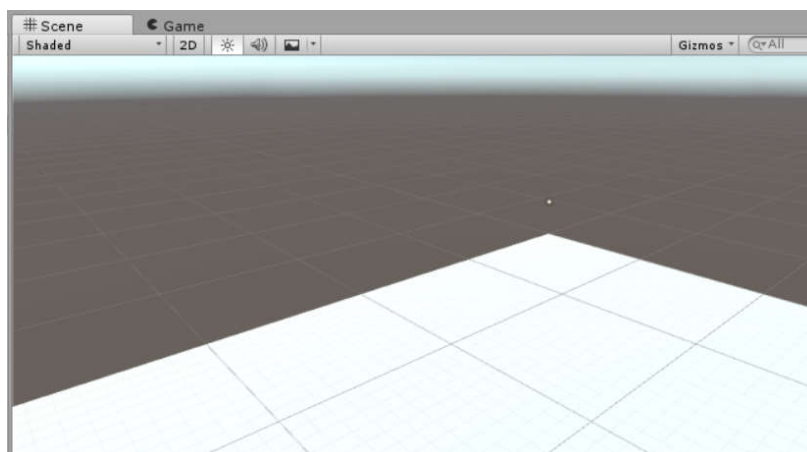
Para insertar un nuevo terreno vamos a "*GameObject->3DObject->Terrain*" desde el menú principal.

Lo que tenemos ahora no es particularmente bonito. Si no puedes ver el terreno, desactiva las luces en la vista de escena. También podrás apreciar que el terreno aparece en la jerarquía y un asset se ha añadido a la librería en la vista de proyecto.



### 2.2. Configuración básica de un terreno

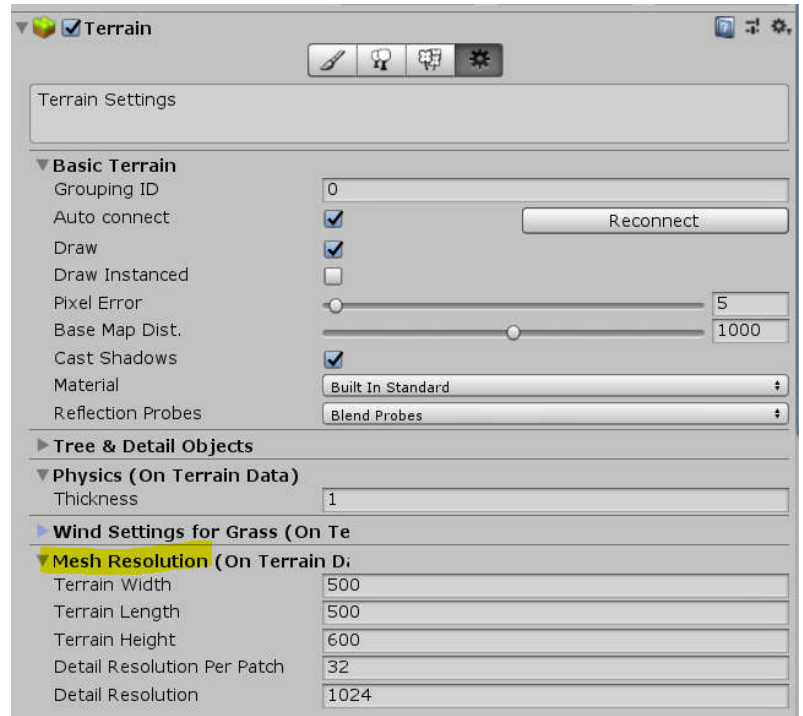
Ahora que tenemos nuestro terreno en la escena, debemos definir algunas propiedades importantes, como la longitud del terreno y algunas propiedades para controlar como de detallado debe ser el terreno.



Para modificar estas propiedades iniciales debemos seleccionar el terreno en la jerarquía de la escena y se mostrarán en el inspector.

Explicuemos que son algunas de las propiedades:

- **Width:** El ancho en **metros** de nuestro terreno.
- **Length:** La longitud en metros del terreno.
- **Height:** La máxima altura en metros del terreno.
- **Heightmap Resolution:** La resolución del heightmap. Debe tenerse en cuenta que debe ser potencia de 2 + 1. (Ejemplo: 129,513)



- **Detail Resolution:** La resolución del mapa de detalles, cuanto más resolución, más precisión a la hora de dibujar los detalles sobre el terreno y colocar objetos.
- **Control Texture Resolution:** La resolución de las texturas pintadas sobre el terreno, más resolución = más detalle, menor resolución = más rendimiento.
- **Base Texture Resolution:** Esta es la resolución base de la textura que se renderiza desde distancia (LOD).

Son bastantes cosas que recordar, especialmente si no has usado herramientas de creación de terrenos de otros motores antes. Tomarán sentido después de que trabajes un poco con terrenos, experimentar siempre es importante.

Yo usare estos valores; pero no significa que debáis tomar los mismos, podéis personalizar vuestro propio terreno.

Width: 500m

Lenght: 500m

Height: 500m

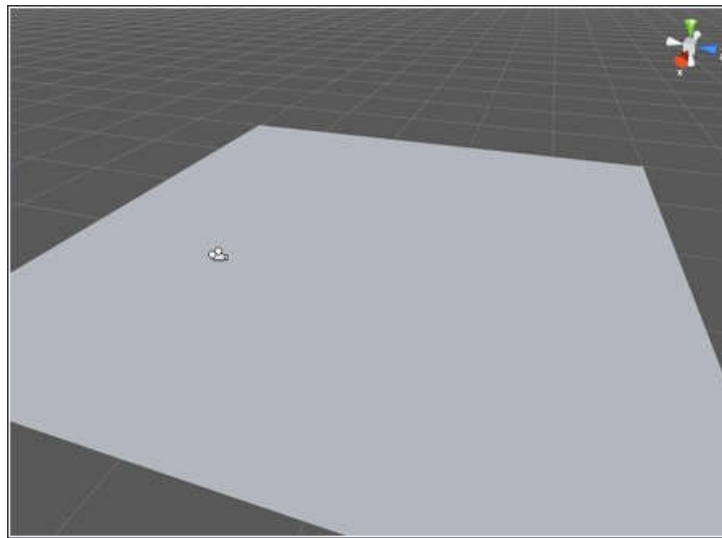
Heightmap Resolution: 513

Detail Resolution: 1024

Control Texture Resolution: 512

Base Texture Resolution: 1024

Ahora deberías tener un terreno plano y gris en tu vista de escena, como en la imagen a continuación.

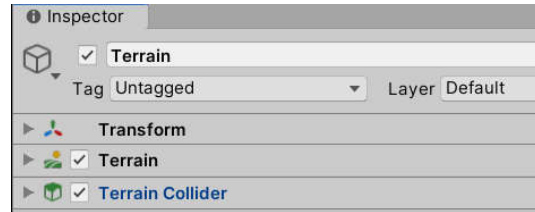


### 2.3. [Herramientas de terreno](#)

Como puedes ver nuestro terreno es fino y no muy bonito, para modificar nuestro terreno haremos uso de las **herramientas de edición de terrenos**. Para acceder a las herramientas necesitamos seleccionar el terreno en la jerarquía, por lo que haz clic sobre él. Cuando hagas esto notarás que el inspector cambia para mostrar las herramientas de edición de terrenos como en la imagen anterior.

El inspector estará dividido en 3 áreas:

- **Transform:** Permite mover y escalar el terreno sobre los ejes X,Y,Z.
- **Terrain:** Contiene varias herramientas y propiedades para el terreno, sobre las que hablaremos ahora.
- **Terrain Collider:** Contiene las propiedades de colisión para el terreno.

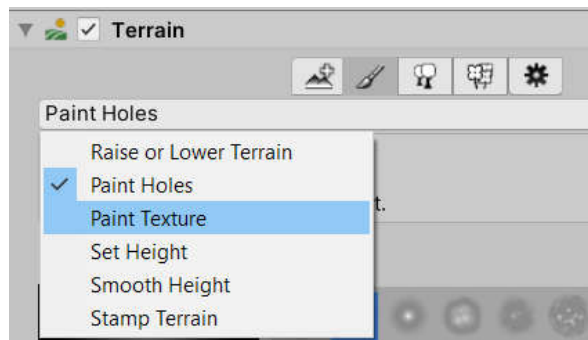


En el **panel Terrain** veremos una fila de botones; estos son los botones editores de terreno, cada uno permite realizar diferentes tareas. Aquí tenéis una descripción de lo que permite hacer cada uno de los botones, en orden de izquierda a derecha:



#### **Paint Terrain**

- *Raise / Lower:* Nos permite levantar y hundir la geometría del terreno usando un pincel.
- *Set Height:* P damos el terreno con una altura límite.
- *Smooth Height:* Nos permite suavizar un terreno para eliminar esquinas, por ejemplo.
- *Paint Texture:* Nos permite pintar texturas sobre la superficie del terreno.



**Place Trees:** Nos permite colocar árboles.



**Paint Detail:** Nos permite dibujar los detalles del terreno como hierva.



**Terrain Settings:** Accede a las propiedades del terreno donde podemos cambiar varias propiedades.



## 2.4. Definiendo una textura base

Lo primero que debemos hacer es definir una **textura base** para el terreno. La primera textura que añadas **será aplicada al terreno entero**, cualquiera que añadas después será pintada en capas superiores.

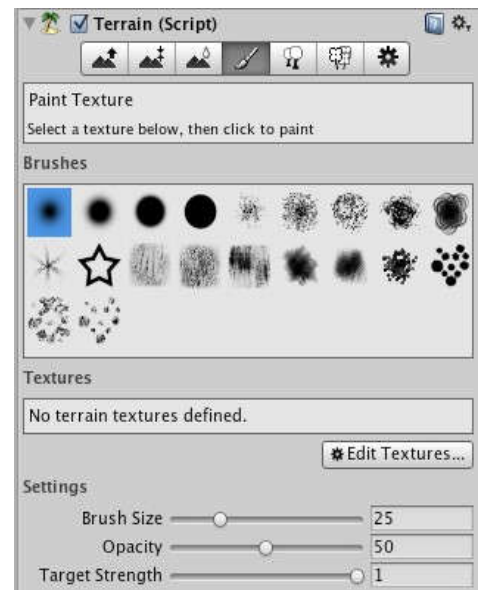
Cuando seleccionas la herramienta Paint Textures veras que aparecen algunas preferencias nuevas, como en la imagen a continuación.

La primera sección permite seleccionar el tipo de pincel, muy similar a los pinceles de Photoshop.

Debajo de los pinceles tienes el espacio de selección de texturas, donde podemos seleccionar la textura con la que queremos pintar.

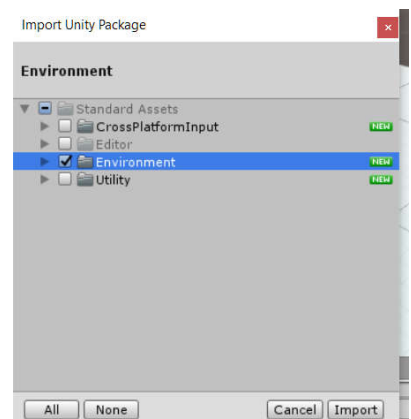
Por último encontramos las propiedades del pincel que nos permiten modificar el tamaño, la opacidad y la fuerza del pincel.

Debemos seleccionar una textura para pintar antes de hacer cualquier cosa, haz clic en "Edit Terrain Layers" y selecciona "Create Layer" , *en versiones recientes se ha sustituido la nomenclatura de Texture por Layer (capa)*.

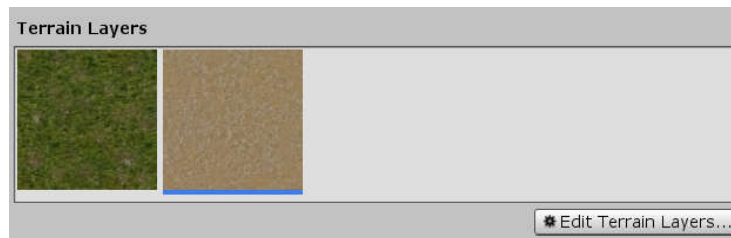


Podemos seleccionar una textura del cuadro desplegable. Selecciona "GrassHillAlbedo" para nuestro ejemplo.

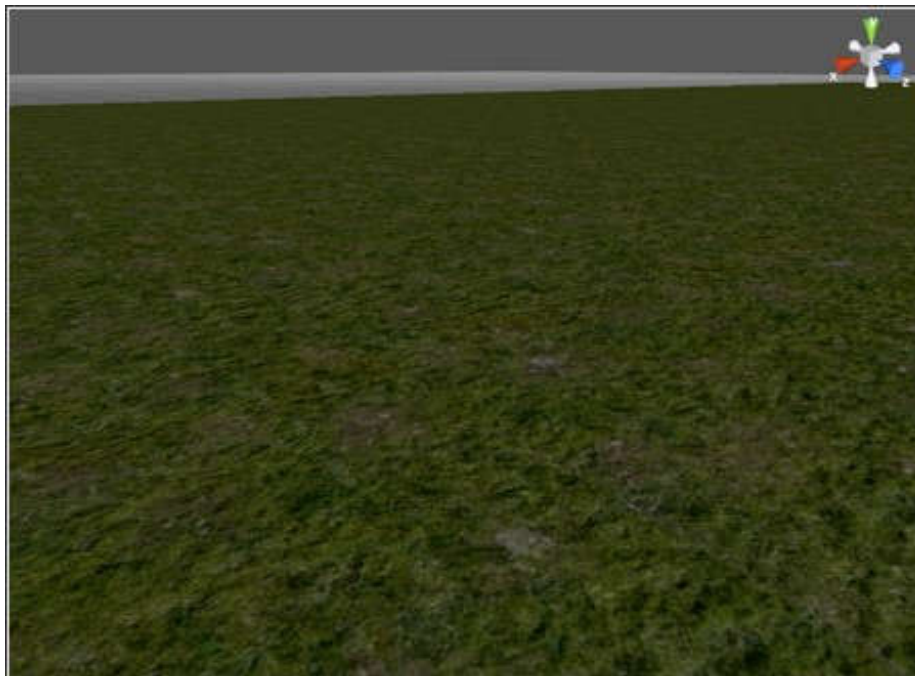
Si no te aparece ninguna textura, podemos **importar** el paquete estándar **Standard Assets** y seleccionamos el Environment.



Cuando hayas hecho clic sobre "add" veras que la textura aparece ahora en el panel sobre el botón "Edit Textures"; también notarás que la textura se ha pintado sobre el terreno.



Debería parecerse a la imagen a continuación (haz zoom sobre el terreno para conseguir una mejor vista de la textura).



Si rotas la cámara alrededor lo bastante lejos del terreno veras las "juntas" de la textura por el terreno. Si quieres añadir más texturas, estas se pintarán en el terreno sobre esta textura.

## 2.5. Geometría del terreno

Nuestro terreno sigue siendo plano y aburrido. Ya que el propósito de este tutorial es que tengas los conocimientos para hacer algo simple no vamos a tomarnos mucho tiempo en que nuestro terreno quede espectacular, veamos las herramientas básicas de geometría que serán suficientes para el cometido de este tutorial.

Si seleccionas la herramienta "*Raise / Lower*" veras que las propiedades cambian. Las opciones serán similares a las que hemos visto con la herramienta de pinceles; estilo de pincel, tamaño y **opacidad** (¡cuidado! si está a cero no modificaremos el terreno). Una opacidad baja creara la geometría despacio, una opacidad mayor hará una geometría más intensa.

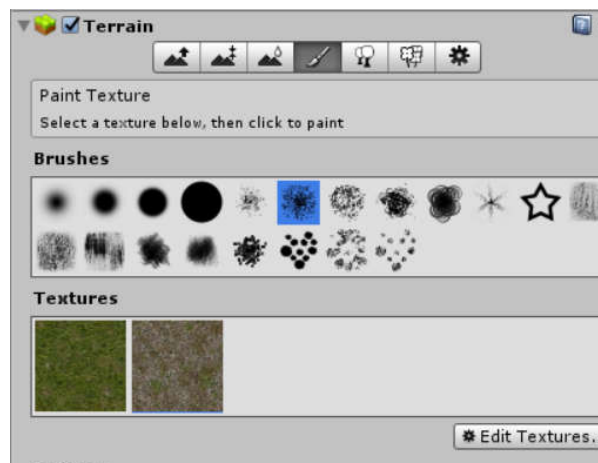
Prueba a usar la herramienta dibujando sobre el terreno. Si mantienes shift al hacer clic hundirá el terreno, pero no por debajo de 0.

### Consejo:

Es recomendable esculpir por pasos, primero las partes grandes y brutas, como montañas, ríos, caminos y luego refinamos los detalles desde los más grandes a lo más pequeños. De esta forma conseguiremos un resultado mucho mejor.

Con esto terminamos la explicación de los terrenos, como siempre lo mejor que podéis hacer es trastear por vosotros mismos esculpiendo y texturizando un terreno.

Añade otra textura, de aspecto rocoso por ejemplo:



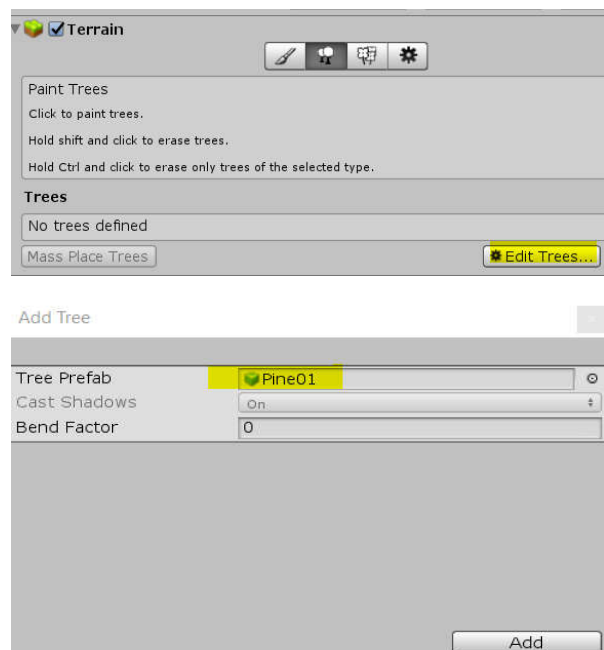
Y aplícala sobre un terreno elevado:



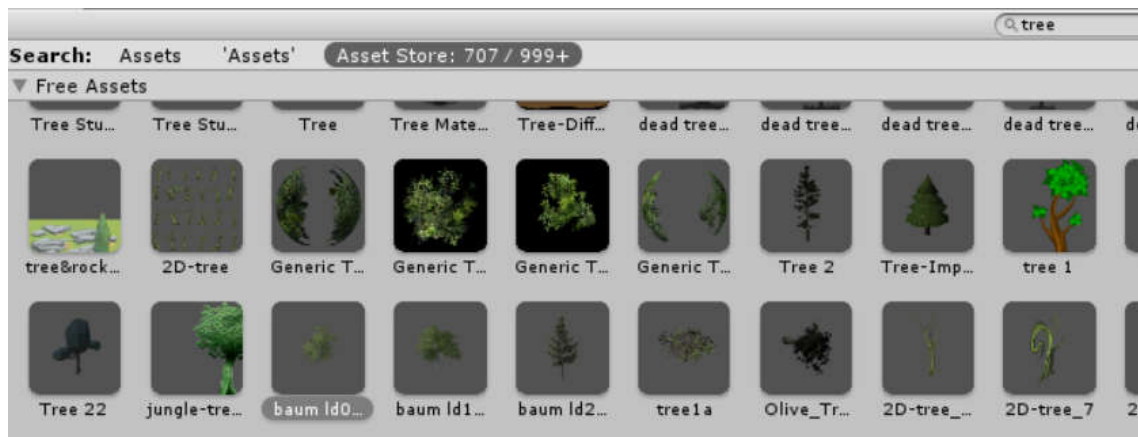
## 2.6. Añadiendo vegetación

Para dar vida a nuestra escena podemos añadir vegetación, tanto árboles como hierba.

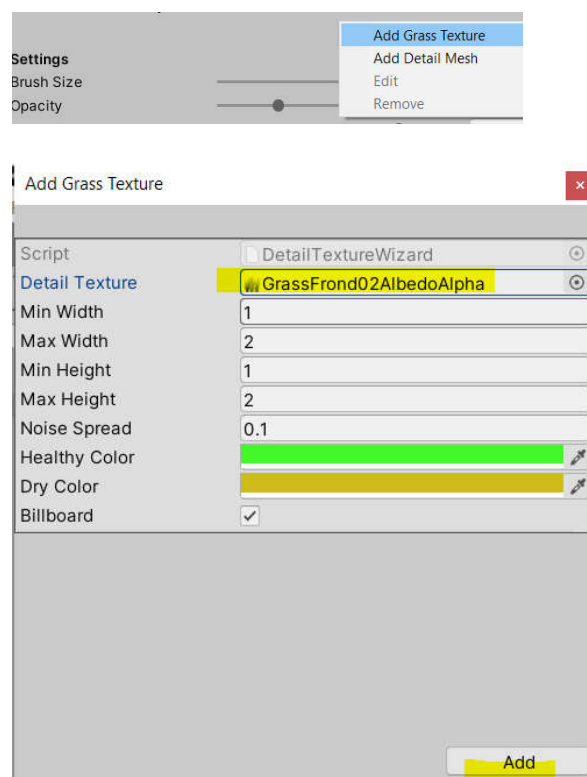
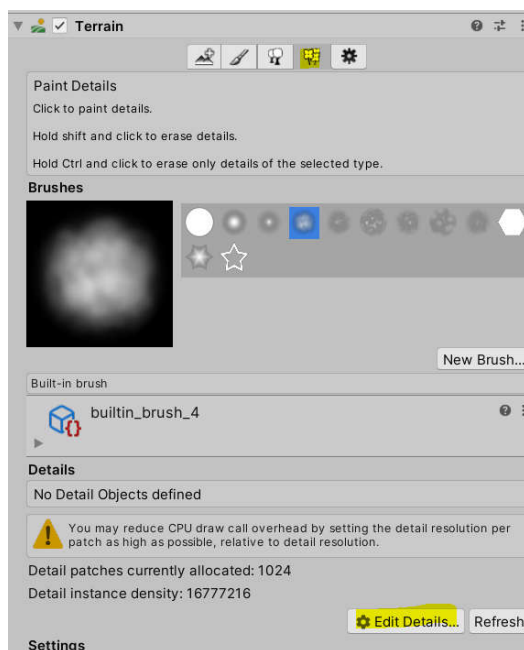
Desde el editor del terreno tenemos la opción de añadir árboles de forma similar a como texturizamos el terreno, añadimos a través de **Paint Trees** los modelos de árbol y posteriormente utilizando una brocha (Brush) podemos situar los árboles de forma rápida en el terreno.



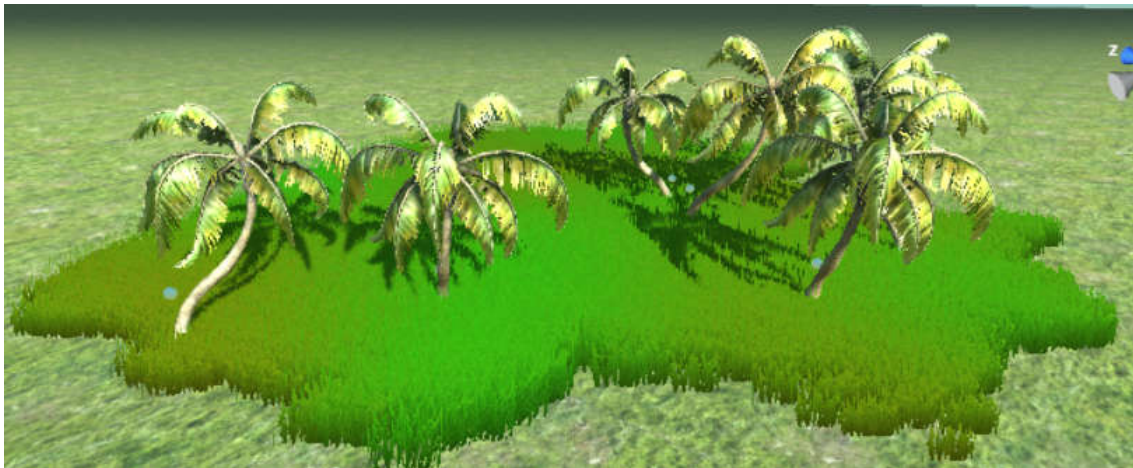
Si queremos utilizar una variedad mayor de árboles, podemos descargarlos desde el asset store, si buscamos tree en nuestros assets podemos seleccionar assets gratuitos de la store de Unity3D y descargarlos directamente en nuestro proyecto.



De igual modo procederemos con la **hierba**:



El resultado, con poco esfuerzo, puede ser espectacular:

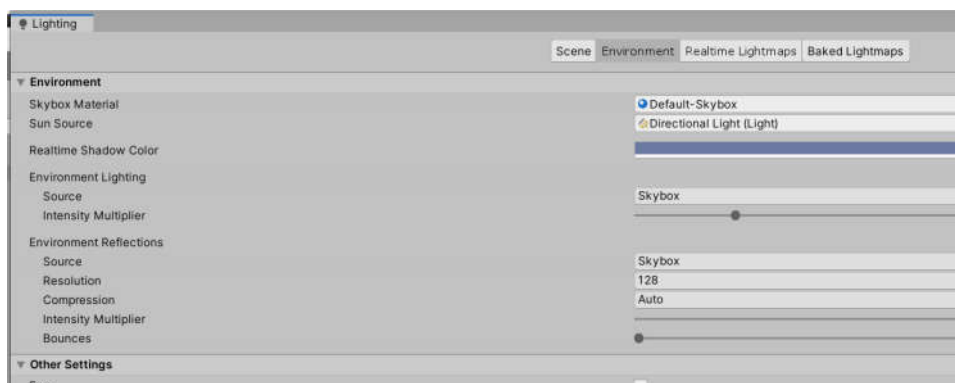
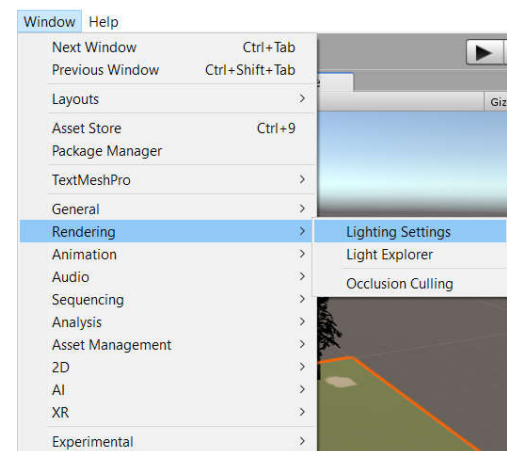


## 2.7. Añadiendo un cielo

En Unity tenemos varias opciones para crear un skybox. La primera es adjuntar un skybox a la cámara o adjuntarla a la escena directamente, haremos esto más tarde.


Vayamos a “Windows->Rendering->Lightning” para mostrar las propiedades de renderizado que se mostraran en una ventana emergente.

En las propiedades de iluminación tendremos varias pestañas, en la de entorno “Environment” veremos una entrada llamada Skybox. Si hacemos clic sobre esta entrada podremos seleccionar un **material para el cielo**.





El paquete standard de assets incluido con Unity trae un material estándar, podéis encontrar más skybox en la tienda de Unity. Por ejemplo:

**Classic Skybox**  
mgsvevo ★★★★★ 4 | 80 Reviews  
**FREE**  
[Open in Unity](#) 

License	Extension Asset
File size	4.6 MB
Latest version	1.0
Latest release date	Feb 28, 2015
Supported Unity versions	3.5.7 or higher

En este caso hemos descargado/importado el paquete **Classic Skybox** Una vez hayas seleccionado un material para el cielo lo veras aparecer en la vista de escena inmediatamente.

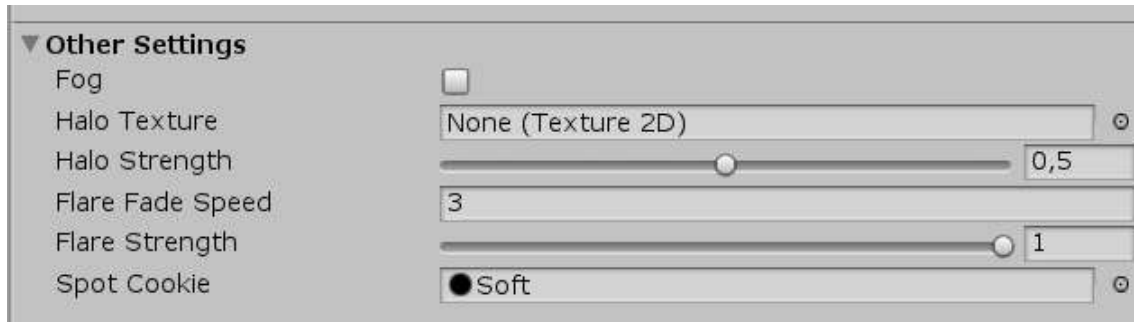
Ahora deberías de ver algo como la imagen inferior.



Con el cielo en la escena hemos avanzado bastante, ahora parece un mundo para un juego.

## 2.8. Añadiendo niebla

Unity incluye un sistema de niebla, que también se encuentra en las opciones de iluminación. Si activas la niebla en el checkbox veras que la escena se vuelve brumoso. Debajo del checkbox tenemos algunas propiedades para nuestra niebla.



La primera es el color, por defecto gris. Podemos cambiarlo según nuestras necesidades, por ejemplo si estamos trabajando en un área desértica podemos cambiar el color a amarillo o naranja, en un mundo mágico, quizás verde.

Debajo del color podemos definir la densidad de la niebla. Una densidad mayor disminuirá la visibilidad y una densidad menor la aumentara.

## 2.9. Color de ambiente

Aunque Unity no incluye un paquete AAA de iluminación global si tiene capacidades para efectos de iluminación globales.

Debajo de las propiedades de la niebla podemos ver "Ambient Light", por defecto como gris. Si queremos dar a la escena un esquema diferente de iluminación, cambiar la luz ambiental es un buen principio.

**Nota:** Recuerda que debes tener la iluminación activada en la vista de escena para que la luz ambiental se muestre. Fíjate en la captura bajo estas líneas para ver la última escena con niebla y luz ambiental.





## 2.10. Añadiendo luces

Por último debemos añadir una luz a nuestra escena, llamada luz solar.

Usando la luz de forma creativa podremos crear efectos sorprendentes, pero incluso con algo tan simple como la luz del sol podemos hacer nuestra escena mucho mejor.

Para añadir una luz vamos a "GameObject -> Create Other -> Directional Light".

Para que la luz tenga efecto sobre la escena es importante tener activado el botón en la vista de escena. Tan pronto como coloquemos la luz direccional veremos su efecto.

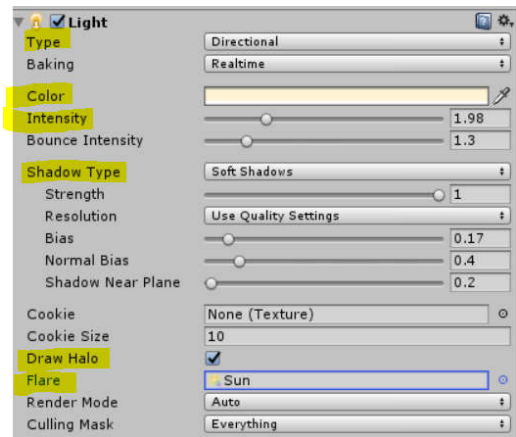
No importa donde coloquemos la luz direccional, su efecto es global a toda la escena. Lo que importa es la dirección, por lo que debes ir a la herramienta de rotación y rotar la luz para que afecte a la escena como queremos.

Podemos crear múltiples luces para crear diferentes entornos o iluminar zonas donde la fuente de luz actual no llega.

Si seleccionas la luz, veras varias propiedades en el inspector. Podemos ajustar el color de la luz para, por ejemplo, proyectar un color amarillo o naranja sobre la escena, o dar una ambientación fría con un color azul.

Cuando creemos una luz solar, queremos que esta luz se renderice como un sol. Para hacer esto, haz clic en el checkbox "Draw Halo" y selecciona "Sun" del menú desplegable Flare.

Si no tienes disponible ningún efecto en Flare puedes importar el paquete Effects desde Assets -> Import Package.



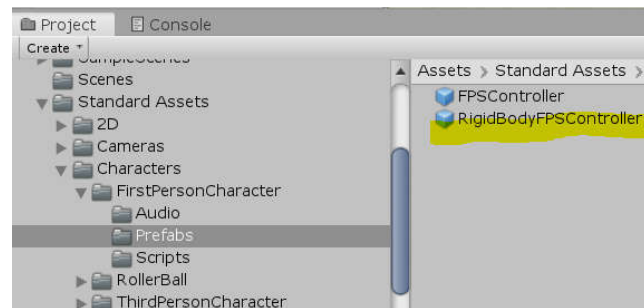
Entonces expande la sección de sombras y selecciona "Soft Shadows" del menú desplegable Type, para que esta luz realice sombras sobre el terreno.



A estas alturas ya deberías tener una escena bonita, o quizás no tanto. Te recomiendo que modifiques el terreno con bastante variación, unas colinas suaves y otras más bruscas, incluso alguna vertical ya que ahora vamos a introducir un controlador en primera persona que caminará por nuestra escena.

### 3. Añadiendo un controlador en primera persona

Unity incluye un controlador estándar en primera persona que hace que los juegos con vista en primera persona sean realmente sencillos de configurar. Para colocar este controlador en la escena, debemos ir a la vista de proyecto y seleccionar "Standard Assets -> Characters->FirstPersonCharacter->Prefabs". Dentro de esa carpeta veras un prefab llamado "**RigidBody FPS Controller**". Si no lo tienes disponible puedes importarlo desde Assets ->Import Package->Characters.



Arrastra este objeto a la vista de escena y posiciónalo de forma que el cilindro toque el terreno, pero no esté por debajo.



Cuando el controlador esté en la escena, haz clic en el botón de reproducir para jugar en tu escena con el controlador.

Los controles son:

- Flechas: Movimiento
- Ratón: Control de la cámara
- Espacio: Saltar

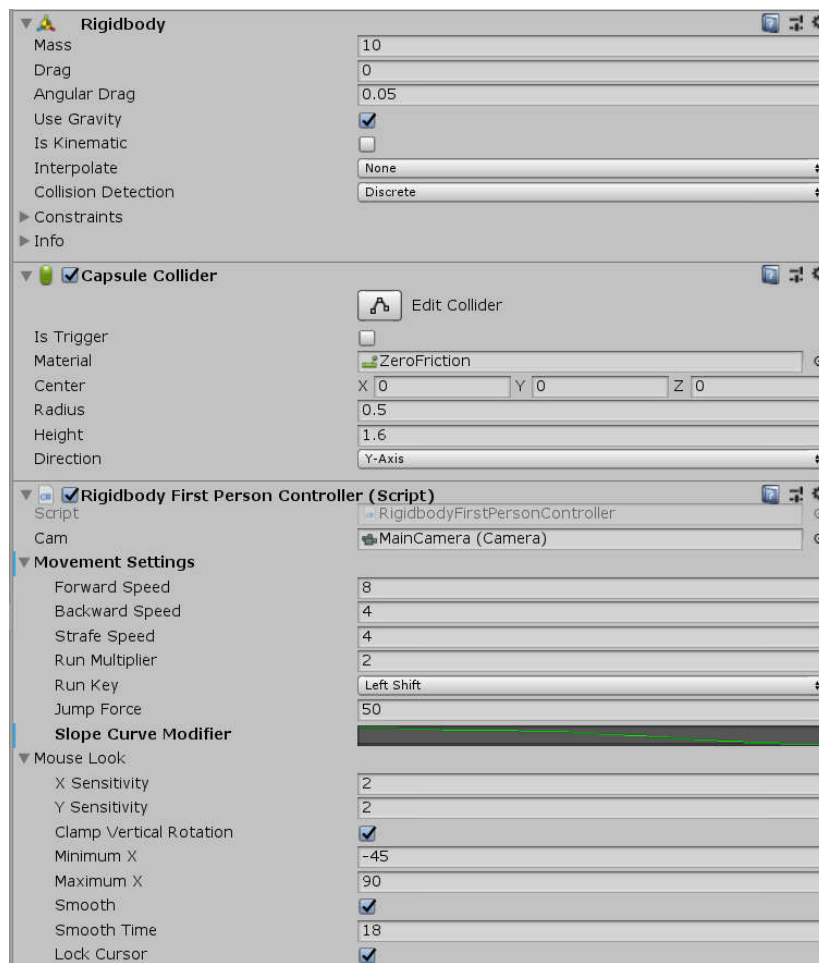
Si en este punto tu controlador cae al vacío atravesando el plano asegúrate de que no esté colocado en la escena atravesando el terreno.

Si te mueves alrededor te encontraras con que ciertas características básicas están incluidas en el controlador, como por ejemplo zonas donde el jugador no pueda pasar. La mejor forma de probar esto es probar a subir una inclinación vertical o bastante vertical. Para salir del modo de juego simplemente hacemos clic sobre el botón de reproducción de nuevo para volver a la vista de editor.

### 3.1. Personalizando el controlador

No es muy divertido ni común arrastrar los prefabricados a la escena y dejarlos como están, al menos querremos personalizarlo para que funcione como nosotros queremos que lo haga. El controlador FPS que hemos colocado en la escena puede ser personalizado a través del inspector, por lo que vamos a seleccionar el controlador en la jerarquía y echar un vistazo en el inspector, veremos que hay bastantes opciones de personalización.

Aquí tenéis una captura de pantalla:



Como se puede apreciar contiene un Rigidbody, un Collider de tipo capsula para las colisiones y un script RigidbodyFirstPersonController a través del cual podemos configurar su comportamiento.

Las propiedades más interesantes pueden ser la de movimiento donde se puede configurar las velocidades en varias direcciones y el *slope* que limita el ángulo por el que puede caminar el jugador y que se configura por medio de una curva (no puede servir para evitar que el jugador salga del escenario).

### 3.2. Editando el Script

Mientras que editar las propiedades básicas debería cumplir con los requisitos de tu juego la realidad es que no es así. En bastantes ocasiones, particularmente con elementos como los controladores de personajes, necesitamos hacer más cambios.

Estos cambios se realizan editando el script mismo.

Para editar el script del controlador, seleccionamos el controlador en la jerarquía, entonces, en el inspector buscamos la parte de RigidbodyFirstPersonController (Script). Estas propiedades son para el FPS, un script que puedes editar.

Una forma rápida de acceder al código es hacer clic en el botón cerca de RigidbodyFirstPersonController (Script). Esto nos mostrara el editor de scripts, donde podremos hacer los cambios sobre el script.

Para empezar haremos un cambio muy simple, eliminaremos la capacidad de saltar del script. Si buscas la linea donde dice:

```
if (CrossPlatformInputManager.GetButtonDown("Jump") && !m_Jump)
{
    m_Jump = true;
}
```

Este bloque de código reacciona cuando el jugador pulsa el botón predefinido como Jump, o saltar en castellano. Podemos eliminar esto comentando o simplemente borrando el bloque entero. Para comentarlo, coloca /\* antes del código y \*/ después. Ahora guarda el script y vuelve al editor.

Si ejecutas tu juego ahora veras que ya no puedes saltar. Deja el código sin comentar de nuevo, guarda y vuelve al editor. Ejecuta tu juego y la función de saltar estará ahí de nuevo.

### 3.3. Últimos detalles

Te habrás dado cuenta de que el script del controlador acepta comandos de entrada, pero, ¿de dónde?

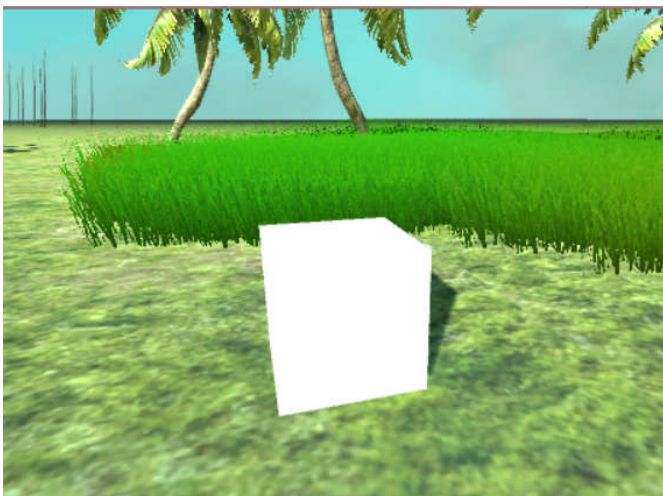
Si vas a "Edit -> Project Settings -> Input" veras el controlador de entrada para tu juego. Puedes configurar el número de entradas usando el comando size y editar cada una de ellas según tus necesidades. Cambiar el nombre cambiara el nombre de la entrada, por lo que tu juego tendrá su esquema único de entrada.

Seguramente no quieras tener tu juego solo en el editor, querrás poder jugar desde un navegador web o desde un ejecutable. Para hacer esto necesitar crear una build. Ve a "File -> Build and Run". En el cuadro de dialogo haz clic en "Add Current" para añadir la escena actual y selecciona la plataforma. Tras esto el juego se compilara en la carpeta especificada.

## 4. Incluyendo objetos en la escena

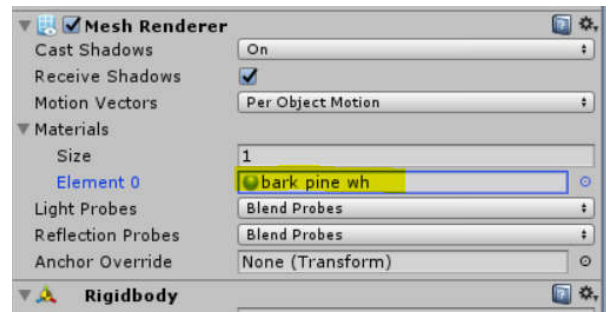
Unity3D proporciona una serie de objetos 3D con geometría muy básica, cubo, esfera..., hay que tener en cuenta que para generar objetos de forma detallada debemos utilizar editores especializados como Blender o 3DStudio.

Vamos a incluir un cubo en nuestra escena, GameObject->3D Object->Cube.





Se muestra sin ningún tipo de textura, lo que hace que parezca fuera de lugar. Vamos a aplicarle un material de forma que parezca un elemento más coherente con el entorno, por ejemplo un material de tipo madera (bark pine):



El resultado:



Para profundizar en como representa Unity3D los modelos en la escena podemos utilizar la [documentación](#).

#### **Mesh Components**

- Meshes
- Material
- Mesh Filter
- Mesh Renderer
- Skinned Mesh Renderer
- + Text Mesh

## 5. Fuentes

- Empezando en Unity3d, UnitySpain