[**victorfergar@gmail.com**](mailto:victorfergar@gmail.com)

**15 de mayo día límite**

**DefaultLayers**

Debug: Para por ejemplo dibujar los contornos de colisión

GUI: Para la interfaz

Additive: Capa que suma el color a las que haya debajo

Alpha: Usada para sprites que puedan tener transparencia

Opaque: Usada para fondos de escenario, no tienen transparencia

**Collider**

PerPixelCollider: Primero comprueba si ha habido un rectangleCollider, y si la ha habido va comprobando pixel a pixel.

Para que no consuma tantos recursos, al collider se le pasa una textura con menor resolución, y por tanto, menos pixeles que comprobar.

**SpriteAtlas**

Se usa el programa Texture Packer.

**Esqueleto**

Usar programa [www.esotericsoftware.com](http://www.esotericsoftware.com), llamado Spine

**RigidBody2D (Parameters)**

Damping: La fricción con el aire o el espacio

Restitution: Lo que va a botar

**Joint2D**

RevoluteJoint: Hace efecto de motor (unes rueda con un coche y la relovulucionas)

# Modelos 3D

Los MaterialsMap serán los que definirán por ejemplo el relieve, los brillos cuando se moja la ropa, etc.

El EnviromentMapMaterial es un mapa que no se calcula en tiempo real, es decir, si pasas entidades por delante, no saldrían reflejadas.

NormalMappingMaterial simula una textura con relieve en un objeto, pero realmente es plano. Se le pasa una textura en canal difuso y otra en normal.

# Partículas 3D

El parámetro EmitRate emite partículas cada vez que mueren las anteriores (si tenemos 100 particulas máximas, cuando mueren estas emite las 100 siguientes).

El EndDeltaScale es la escala de las partículas al morir.

# Componentes Físicos 3D

En **RigidBody3D**, el parámetro “EnableContinuousContact” hace que si lanzamos un objeto muy rápido contra una pared, este puede atravesarla sin darnos cuenta. Si activamos este parámetro, comprueba si entre el frame actual y el anterior se ha atravesado algún objeto y evita que lo haga.

El parámetro “Restitution” representa la cantidad de energía que perderá en cada choque.

**Joint3D** son las uniones entre los elementos de un modelo.

# Importante

A la hora de introducir nuevos elementos gráficos al proyecto, hay que revisar que en sus propiedades los atributos “Accion de compilación” = “Contenido” y “Copiar en el directorio” = “Copiar si es a posteriori”

**Compilar proyectos en 32 bits**

Debido a que Wave Engine solo trabaja con 32 bits (como todas las consolas y móviles del mercado), es necesario hacer un cambio.

Hay que irse al lado del botón Iniciar, que hay un cuadro desplegable en el que pone Debug, y hay que seleccionar la opción “Administrador de configuración”. Ahí debemos abrir la lista de “Plataformas de soluciones activas” > Editar y eliminar “Any CPU”

## Firmar proyecto para Windows Store

Convertimos el proyecto para Windows Store, y ahora debemos abrirlo con visual studio, abrimos Package.appxmanifest > Choose Certificate y crearlo.

## Culling

Si está activado no dibuja los objetos que esté fuera del rango de la cámara, pertenece al RenderManager