

First Person Shooter in Godot:

Esto es un documento guiado y orientado a facilitar como hacer ciertas cosas en el proyecto del videojuego en el que estoy trabajando, para así evitar con el tiempo como funcionan ciertas cosas o tener de forma escrita y clara como trabajan los diferentes códigos y comportamientos. Piensa en esto como un tutorial de qué hacer en caso de olvidar con el tiempo parte del workflow en según que cosas.

CREAR UN ARMA NUEVA LISTA PARA SER USADA POR EL JUGADOR:

-Aquí te explicaré cual es el workflow a seguir a la hora de añadir nuevas armas al proyecto.

Lo primero de todo, será crear una escena, con un nodo 3D como padre que herede de “**Handgun**”. Este deberá tener como componentes y nodos hijos:

-Shooting component (**SemiautomaticComponent/AutomaticComponent/etc**)

****Por ahora, solo un comportamiento es posible por arma, la idea es añadir modificadores**:** Esto controlará el comportamiento de disparo de un arma.

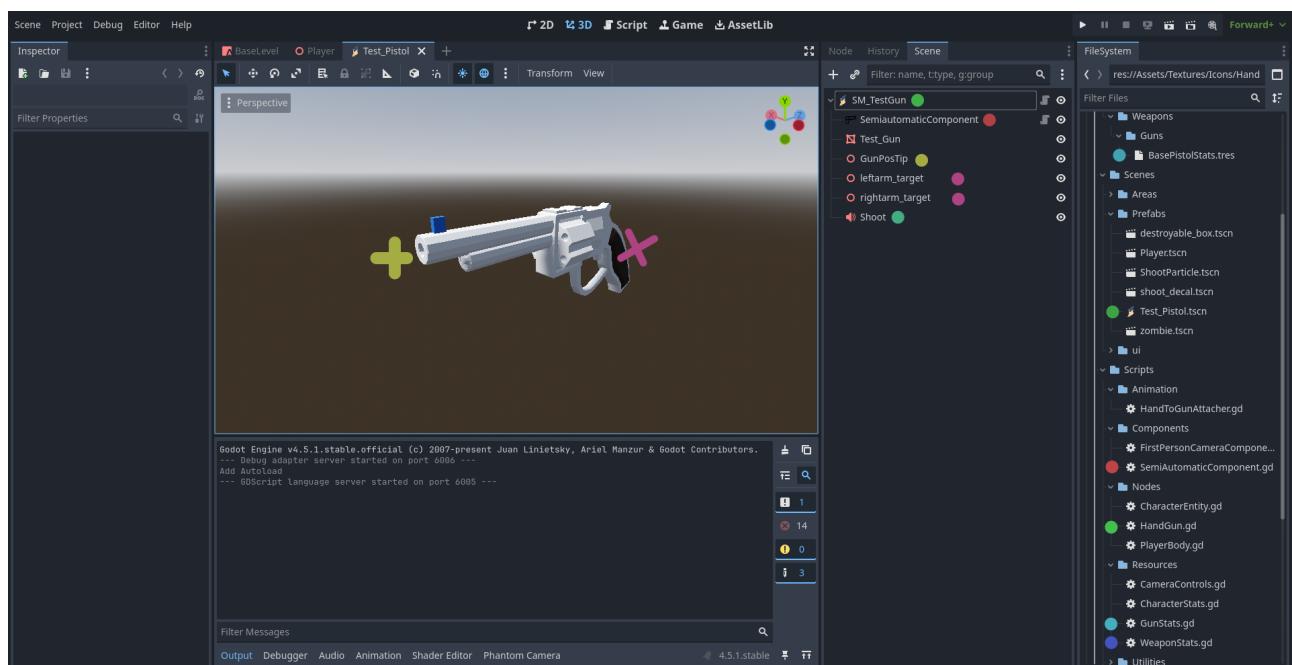
-La mesh del arma. (**MeshInstance3D**)

-Un nodo3D en la punta del cañón del arma (DEBE llamarse “**GunPosTip**”)

-Nodo 3D con la transformación que la mano del soldado tendrá:

****hay dos, uno por cada mano, deben llamarse “leftarm_target” y “rightarm_target”****

-Sonido de disparo que vaya a usar: (**AudioStreamPlayer3D**)



-Ejemplo de un arma montada-

azul oscuro: “**WeaponStats.gd**” es el código que genera un resource (**scriptable object** equivalente en Godot) que contiene las características genéricas y en común de cualquier arma que vaya a ser creada en el juego, cosas como el nombre o el cooldown entre activaciones.

cian: “**GunStats.gd**” es el código que hereda de “**WeaponStats.gd**”, añade las características en común que todas las armas que disparan tienen. (en el futuro habrá una para armas de cuerpo a cuerpo). También muestra un resource que hereda de esa clase ya creado en el editor.

Rojo: “**SemiautomaticComponent.gd**”, junto a sus futuros primos, controlan el comportamiento relacionado con los disparos y la cadencia de las armas de fuego, posiblemente se pueda reusar parte de su lógica para crear armas cuerpo a cuerpo

Verde: “**HandGun.gd**” es el código que forma parte del nodo padre del arma. Este mantiene la información necesaria para que el componente de disparo (**rojo**) funcione con normalidad. También tiene una variable de tipo **GunStats** (cian) para tener un resource que es fácilmente editable.

Amarillo: **GunPosTip** es un nodo 3D que usaremos principalmente para utilizar su posición como punto donde instanciar las partículas de disparo de la respectiva arma.

Morado: Los **targets** a los que los **IK** de las manos del modelo de los brazos se aferrarán de forma procedural, son solo nodos 3d sin más. Deben tener la rotación ya montada.

Una vez creada el arma, esta escena deberá estar siempre como hija del nodo 3D llamado “**GunPos**”, nodo hijo de la escena del player.
