

## **Introducción:**

Adjunto un documento describiendo los aspectos generales del proyecto. Como documentación adicional debería haber un acceso directo llamado **ClassDocumentation.html** dentro la carpeta raíz del proyecto con información adicional generada de forma automática con Doxygen sobre las clases del proyecto.

## **Jugador:**

Tal como se pedía es un proyecto compuesto por un Jugar con capacidad de disparar con la rotación del ratón, y que añade la funcionalidad de apuntar a los enemigos. Al apuntar hay una rotación de la cámara además del mesh del personaje. El estado de apuntar solo es cancelado si dejamos de disparar o si nos quedamos sin enemigos a los que apuntar, ya que al eliminar un enemigo automáticamente busca otro enemigo al que apuntar. Esto no incluye las dianas (representadas con un mesh en forma de pirámide), ya que no estaba seguro de si incluirlas o no. Esta inclusión podría ser hecha de forma extremadamente fácil al haber modularizado la detección de elementos atacables dentro del mapa. Toda la funcionalidad de disparar se encuentra modularizada dentro de un componente que puede ser añadido a cualquier actor.

## **Enemigos:**

Para los enemigos hice una derivación directamente de "AActor", no me quedó claro si una derivación de una de las clases hijas de "AActor" sería válida como "APawn" para utilizar las herramientas de IA dentro de Unreal. Por ello, he creado mi propia funcionalidad de IA desde cero. Toda la funcionalidad de IA reside dentro de un componente, por lo que añadir IA a un actor sencillamente requiere añadir dicho componente. La IA consta de los siguientes estados:

- Patrullar: durante 5 segundos el enemigo patrulla dentro de una zona de patrulla, teniendo en cuenta un objeto llamado "Patrol Point". Se pueden colocar estos "Patrol Point" de forma indiscriminada dentro de la zona de patrullar, y ésta los detecta automáticamente.
- Detección del Jugador: el jugador es detectado inmediatamente al entrar en la zona de patrulla. Al detectarlo, los enemigos pasan a perseguir al jugador hasta que éste sale de la zona de patrulla.

- Descanso: el enemigo descansa cada 5 segundos, en este estado se limita a estar quieto durante 3 segundos. Esto afecta a los dos estados anteriores. Al terminar esos 3 segundos vuelve a su estado anterior.

## **Input:**

Los inputs son los mismos que para la plantilla de “Third Person Template”, solo he añadido disparar al botón izquierdo del ratón y apuntar al botón derecho del ratón. Tal como indica el documento de diseño, dejar hundido el botón de disparar mantiene el disparo hasta que se suelta. De igual manera, solo es posible apuntar si se dispara a la vez, sino al cabo de unos milisegundos se cancela el estado. El estado de apuntar también se cancela si no hay más enemigos a los que apuntar, pero si hay más enemigos cambia de objetivo inmediatamente.

## **Otros:**

El proyecto también incluye la Diana con las especificaciones descritas, la plataforma de disparo que cambia la velocidad de disparo, el widget para contar los enemigos eliminados, el proyectil y el widget de vida.

Para todo he intentado modularizar los componentes y las interacciones internas entre clases para hacerlo fácilmente escalable. También hay bastante flexibilidad en el Editor como por ejemplo añadir zonas de patrulla y puntos de patrulla de forma infinita. También se pueden añadir enemigos de forma dinámica, y que la propia zona de patrulla los detecta.

Añadir también que los elementos descritos como C++ tienen sus “Scene Components” creados con C++, aunque personalmente me suele gustar más crearlos dentro del BP y linkearlos en C++ para tener más estabilidad. He tenido que corregir algunos offsets que ocurren con estos componentes durante la fase de comienzo de la partida.

Me encantaría discutir esto y más en una videollamada, ya que creo que sería más dinámico y divertido.

¡Muchas gracias!