

First person shooter (Primer humano disparador)

Descripción

Primer humano disparador es un juego de disparos en primera persona desarrollado en Unity 2022.3.4f1. El objetivo principal del juego es eliminar a todos los enemigos en cada nivel para avanzar y completar el juego.

La interfaz registra el tiempo de juego, la salud del jugador, el número de enemigos eliminados, el nivel actual y el arma en uso a través en un elemento de UI.

Desarrollo

Tomé como punto de partida el shooter en tercera persona de la primera parte del curso y me focalicé en animar a los personajes, generar mapas proceduralmente, hacer que los enemigos se muevan por un navmesh y que tomen decisiones en base a su "estado".

El repositorio está subido en <https://github.com/tjocucci/utn-shooter>

Controles

- Apuntar: posición del mouse
- Disparar: click izquierdo
- Movimiento: WASD o flechas
- Correr: Shift

Características

- Los personajes tienen modelos y animaciones sacadas de Mixamo. Usé un blend tree 2D para el personaje principal, de manera que responde a correr y a caminar y además tiene animaciones específicas para los pasos laterales y hacia atrás.
- Generación Aleatoria de Enemigos: los enemigos se generan de forma aleatoria y a tiempos espaciados dentro de cada nivel.
- Mejoré mucho la generación de mapas. Se pueden crear de manera procedural desde el editor. Me basé en el shooter de Sebastian Lague para esto pero lo cambié bastante porque en vez de obstáculos implemento paredes (tipo Doom). Se pueden cambiar varias cosas: los colores, la densidad de paredes, el tamaño etc.
- Selección de Armas: la selección de armas sigue igual a antes pero no impté los prefabs de otras armas así que hay una sola.
- Salud: tanto el jugador como los enemigos cuentan con un sistema de salud. Ambos reciben daño al recibir balazos y mueren si la salud se agota
- Mejoré el movimiento de los enemigos, cuando se crea un mapa se bakea un navmesh que los enemigos usan para moverse.
- El comportamiento de los enemigos está basado en estados:
 - Idle: solo cuando el jugador muere para que los enemigos no sigan moviéndose
 - Searching: cuando el jugador está fuera de su rango de visión, los enemigos se mueven hacia él pero sin disparar

- Chasing: cuando el jugador está dentro de su rango de visión, los enemigos se mueven hacia él y disparan
- Standing: cuando el jugador está dentro de su rango de visión pero demasiado cerca, los enemigos mantienen la distancia y disparan
- Niveles: Los niveles ahora incluyen las características de cada nivel (cantidad de enemigos, intervalos de spawn, dificultad de los enemigos, etc) pero además las características del mapa, de manera que se modifica todo en el mismo lugar.
- Movimiento y Aim: usé un CharacterController para el movimiento del jugador y ahora el aim y la cámara están diseñados para la modalidad de primera persona.
- Progresión: el objetivo es eliminar a todos los enemigos en cada nivel para avanzar al siguiente nivel. El juego se completa cuando se eliminan todos los enemigos en todos los niveles. Si morís perdés.
- Eventos: usé eventos para comunicar muerte de jugador, de enemigos y cambio de nivel.
- Hay un menú principal y una UI ingame que informa estadísticas del juego. También hay un banner que muestra instrucciones y permite volver a jugar o volver al menú principal cuando el juego termina (ganando o perdiendo). La UI tiene sonidos para ganar, perder, iniciar el juego y botones.

Pendientes e Inquietudes

No respeté las consignas a rajatabla, en parte por falta de tiempo y también hay varios aspectos a mejorar en general:

- No implementé distintos tipos de armas y proyectiles, pero creé los prefabs de manera que fácilmente se pueden crear nuevas armas y tweekear sus estadísticas (tiempo entre, disparo, daño, velocidad).
- La información que se imprime en consola no es exactamente la del enunciado.
- Solo incluí sonido para los disparos y la UI pero no para otros aspectos del gameplay
- Hay implementados 3 tipos de enemigos (fácil, intermedio y difícil) pero creé pocos mapas, solo como para probar por lo que el enemigo difícil no aparece en ningún nivel. Esto es fácil de cambiar, solo hay que agregar más mapas y se puede hacer fácilmente desde el editor.
- La navegación de agentes no permite una búsqueda demasiado inteligente. Si el enemigo está inmediatamente del otro lado de la pared, el agente no lo ve y no se mueve. Esto se podría mejorar con un sistema de waypoints o algo similar.
- Las animaciones que bajé de Mixamo tienen un tema que es que los brazos no apuntan exactamente hacia adelante. Así que le puse el arma por de una manera bien antiestética.
- No me preocupé mucho por el balance del juego y los niveles.
- No hice un buen tratamiento estético. Elegí colores y sonidos sin mucho criterio.
- En el camino saqué el devMode
- No le hice animaciones de pasos laterales o hacia atrás a los enemigos