

## Carpeta Procesos 2

Durante el segundo bloque del semestre tuvimos la oportunidad de aprender una variedad de temas para ayudar con el desarrollo del proyecto, en varios casos elaborando actividades para mostrar el conocimiento.

Comenzamos viendo los temas del multijugador donde aprendimos a utilizar dos herramientas diferentes, primero usando Unity Netcode for Objects para hacer escenas básicas con aspectos de multijugador, incluyendo luego una escena con mecánicas de disparar a otros jugadores y agarrar cubos de vida, donde estos cambios se ven reflejados en el juego del host y de los clientes.

También utilizamos Photon para aprender como hacer juegos en Unity con multijugador a través de servidores y de esta manera poder aplicarlos a nuestro juego, aunque al final estos aspectos de multijugador fueron elaborados más por mis compañeros de equipo, mientras que yo me enfoqué en otras partes del juego.

También aprendimos sobre el desarrollo de buenas interfaces de usuario, primero tomando en cuenta la plataforma en la que se va a jugar nuestro juego y los tipos de usuarios que lo van a jugar. Después de diseñar un prototipo básico, le pedimos a los otros alumnos y a unas personas más cercanas a nuestro usuario ideal que lo prueben y mejoramos nuestro prototipo basado en los resultados obtenidos. Yo tuve la oportunidad de diseñar el prototipo y después de recibir la retroalimentación del mismo, lo mejoré e implementé en Unity para nuestro juego.

Uno de los conceptos más nuevos (para mi por lo menos) fueron los shaders, herramientas para crear efectos visuales poderosos en el juego. Aprendimos cómo hacer shaders manualmente con código modificando los vértices del objeto, utilizando conceptos de tiempo y texturas de ruido para crear efectos visuales de agua y hologramas. Esto lo llevamos aún más allá cuando creamos un efecto de fuego utilizando tres texturas diferentes, cambio con tiempo, y una fórmula con muchas variables para cambiar el fuego como se desea.

En nuestro juego utilizamos mucho estos shaders manuales con código, pero yo en particular elaboré un unlit shader con movimiento horizontal para la misión del tren, con el propósito de hacer parecer que el tren se está moviendo.

Con shaders también aprendimos sobre el shader graph, una herramienta en Unity para elaborar shaders con bloques en lugar de usar código, aunque también tiene algunas ventajas distintas a esta visualización. Los shader graphs tienen nodos de ruido, tiempo, y UVs, entre otras cosas, que pueden ser muy útiles al elaborar un shader. Para el shader de nuestros menús (un efecto de scanline con estática) yo utilicé el shader graph y me ayudó a poder visualizar cada paso del shader y a evitar tener que diseñar las texturas de ruido manualmente.

De la misma manera aprendimos sobre inteligencia artificial en juegos, en particular utilizando estados de máquina para poder darle comportamientos más elaborados a los

enemigos del juego. Hicimos ejemplos cómo un pequeño juego con tanques que tienen estados de patrol, atacar, y evitar, cambiando de uno a otro basado en el ambiente. Estos conceptos los evolucionamos a distintos tipos de comportamientos avanzados, como el ejercicio de Parvadas donde tuvimos que proporcionar una detección de ambiente y cambio de movimiento para un grupo de pájaros, con el fin de que estos eviten los obstáculos en su camino.

Yo de hecho ya había utilizado una máquina de estados para nuestra clase de enemigo genérica en el juego, pero aún así elaboré dos nuevos enemigos utilizando máquinas de estados para la misión del tren, un enemigo que se mueve de punto a punto en un orden predeterminado y si encuentra al jugador lo captura, y otro que no se mueve de su lugar pero le avienta bombas al jugador basado en su posición.

Algunos otros conceptos que aprendimos durante el bloque fueron los aspectos del render pipeline, donde vimos cómo se generan los visuales de un juego desde cero. Comenzamos elaborando transformaciones de vértices con código en lugar de hacerlo con la interfaz gráfica de Unity. Este concepto lo utilizamos para diseñar un robot y hacerlo caminar como una animación.

Luego llevamos eso más allá, aprendiendo como hacer la iluminación con código tomando en cuenta la cámara, posición de la luz, y los objetos en la escena para elaborar un ejercicio donde combinamos todos los aspectos, creando esferas en posiciones aleatorias y usando el proceso de rasterización para hacerlas aparecer en una textura en la escena, luego salvando esta como una foto.

En cuanto a aspectos del juego que elaboré fuera de los temas vistos en clase, diseñé una misión del juego en un tren, con aspectos de stealth para llegar al final sin ser descubierto. Este también lo desarrollé en Unity, incluyendo el ambiente, los shaders, enemigos y boss al final.