



DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN
FACULTAD DE CIENCIAS FÍSICAS Y MATEMÁTICAS
UNIVERSIDAD DE CHILE
CC3501 MODELACIÓN Y COMPUTACIÓN GRÁFICA

TAREA 4 - PROYECTO UNITY 3D

I WANNA BE THE TOMMY

| | |
|-------------------|-------------------------------------|
| Por: | Cristóbal Jaramillo Bryan Ortiz. |
| Profesor: | Daniel Calderón |
| Fecha de Entrega: | 26 de junio de 2019 |

Solución Propuesta

Antes de empezar con este proyecto fue necesario decidir qué se iba a realizar, pues existían muchas posibilidades viables. La decisión final convergió a un videojuego, teniendo altas expectativas para el posible resultado.

Se testean diferentes motores gráficos, comparando sus entornos, los recursos de hardware que requería cada uno, y la complejidad que tenía como para empezar algo sin conocimiento previo. Finalmente se decide utilizar *Unity*, pues posee una interfaz más intuitiva y existía una mayor variedad de tutoriales.

Una vez realizado lo anterior, se intenta formular la temática principal del juego, analizando los posibles escenarios, la jugabilidad, y cómo aplicar los conocimientos aprendidos en el curso. Así, después de varias ideas de variadas complejidades, se llega a la idea de un juego estilo *shooter*, en el cual se debe sobrevivir a oleadas de zombies (y trolls), los cuales seguirán al personaje hasta terminar con su vida, este juego está fuertemente inspirado en *Box Head*, sin embargo su jugabilidad es ligeramente diferente y simplificada. Por otro lado, en el proyecto se implementan varios de los contenidos vistos en el curso, tales como transformaciones de objetos 3D, uso de cámara y perspectivas, iluminación, modelación jerárquica, entre otros.

Para la realización de este proyecto, primero se realiza una mini capacitación a base de YouTube y tutoriales incluidos en el mismo Unity, descargando *assets* de juegos prehechos y modificándolos, jugando con los parámetros y familiarizándose con el programa. Tras ello, se inicia la construcción de lo que sería el juego a desarrollar. Las tareas se dividen procurando poder trabajar en paralelo y avanzar de forma eficiente. Dentro de las principales, se destaca la creación del entorno, la inclusión, animación y programación de personajes (se usaron varios scripts prediseñados), la creación y configuración de los *mesh collider*, el ajuste de las físicas, el posicionamiento y movimiento de la cámara, las texturas, entre otros.

En la repartición de actividades a realizar, Cristóbal queda encargado de:

- 1.- Implementación de la cámara en primera persona: Para lograr esto se trabajo con una modelación jerárquica de la cámara, es decir la camara es un hijo del personaje, de esta forma se logran mover en conjunto y así lograr una mayor realidad a la hora de jugar.
- 2.- Movimiento de la camara en conjunto con el arma: Como el arma es hijo de la camara, se implemento un script que modificara los ángulos de rotación de la camara, para realizar una rotación hacia los lados y hacia arriba, como se realizo con modelación jerárquica, el arma sigue el movimiento de la camara.
- 3.- Implementación del enemigo: Para hacer esto, se busco un modelo 3D con animaciones en *Unity* y se realizaron script que buscaran las coordenadas del jugador y el enemigo fuera hacia ahí.
- 4.- Implementación de físicas del personaje principal: Para esto se realizo una hitbox al personaje, de manera que captara las colisiones con las paredes y con los disparos.

Por otro lado, Bryan se encarga de:

- 1.- Modelar el mapa físico, generando las estructuras necesarias para el diseño del laberinto a partir de cubos escalados y trasladados. Para ello es necesario generar un *mesh collider* para que los personajes detecten los objetos y no los atraviesen.
- 2.- Implementar spawners de mobs, además de los scripts y ajustes para que los enemigos puedan dañar al personaje y recibir daño de su parte.
- 3.- Implementar canvas en pantalla para mostrar información, tales como puntuación, nivel de vida y mensajes de fin del juego, para ello se utilizan algunas animaciones que dependen de los diferentes estados del juego.
- 4.- Configurar las físicas y la iluminación, ajustando las posiciones de los personajes, la gravedad, etc.

Cabe destacar que a pesar de repartirse los objetivos e intentar aplicar *divide and conquer*, en ocasiones era necesario que ambos integrantes trabajaran en una sola tarea dadas las complejidades que se iban presentando.

El resultado final consiste en un juego en primera persona, donde la cámara está ubicada exactamente en la misma posición del personaje. Este último debe sobrevivir a el ataque de zombies y trolls, cada uno con una vida y nivel de golpe predeterminado. La forma de defensa del protagonista consiste en un fusil que dispara un rayo (*raycasting*), el cual si intercepta a algún enemigo le hace daño. El juego consta con efectos de sonido hechos por los mismos integrantes, música de fondo, marcador de puntos e indicador de vida.

Dificultades Encontradas

Durante el desarrollo de esta tarea se encontraron diversas dificultades; las cuatro mas destacables fueron:

1.- Decidir: Antes de decidir hacer un juego, habían muchas ideas y cada una se veía frenada por un "¿Podemos hacer esto en un tiempo acotado?", de modo que dimensionar la complejidad fue un aspecto importante.

2.- Familiarizarse con Unity: El entorno de este motor gráfico tiene muchas opciones y configuraciones con las que en un principio es fácil perderse, hay que ser muy riguroso con el orden y recordar qué variable estaba asignada a qué. Para ello fue necesario ver varios tutoriales, e intentar y fracasar cierta cantidad de veces para notar qué cosas hacían fallar qué.

3.- Identificar errores: En muchas ocasiones se generaron errores en el editor causadas por que cierto parámetro estaba mal definido o porque cierto atributo no estaba asignado a un objeto. Descubrir estos errores costó un tiempo considerable al momento de realizar la tarea, pues a menudo hubo que buscar e intentar con las soluciones sugeridas en foros relacionados. Algunos errores eran esporádicos y desaparecían sólo eventualmente, lo cual causaba confusión de vez en cuando.

4.- Implementación de modelos 3D y Animaciones: Para este trabajo inicialmente se descargaron modelos 3D sin animaciones prefabricadas, lo cual estaba incrementando considerablemente la dificultad de la tarea. La solución a ello fue descargar modelos prefabricados, sin embargo implementarlos tampoco fue sencillo, esto a menudo trajo problemas y errores de código.

Considerando todo lo anterior, el tiempo total invertido en la tarea es cercano a las 23 horas de trabajo continuo.

Instrucciones de Ejecución

Para ejecutar el juego existe un archivo llamado Game.exe, el cual permite ejecutar el juego en una ventana, esta visualización solo permite jugar al videojuego pero no permite ver la modelación, los códigos o la escena. Para realizar esto último se debe instalar Unity (la versión utilizada es la 2019 1.5f) ejecutarlo y añadir la carpeta del proyecto, luego buscar la escena Tarea4 y ejecutarla dentro del programa.

Dentro del juego, el personaje se mueve con las teclas W, A, S, D para ir hacia adelante, izquierda, atrás y derecha respectivamente. Para disparar se utiliza el **Click izquierdo**. Cabe destacar que el juego no tiene una interfaz muy prolija, por lo que para cerrar el programa es necesario cerrarlo con **Alt + F4**.

Resultados

A continuación se muestran algunas capturas de pantalla del juego en funcionamiento:



(a) Inicio del juego.



(b) Enemigos



(c) Disparo

Figura 1: Capturas de las visualizaciones obtenidas.