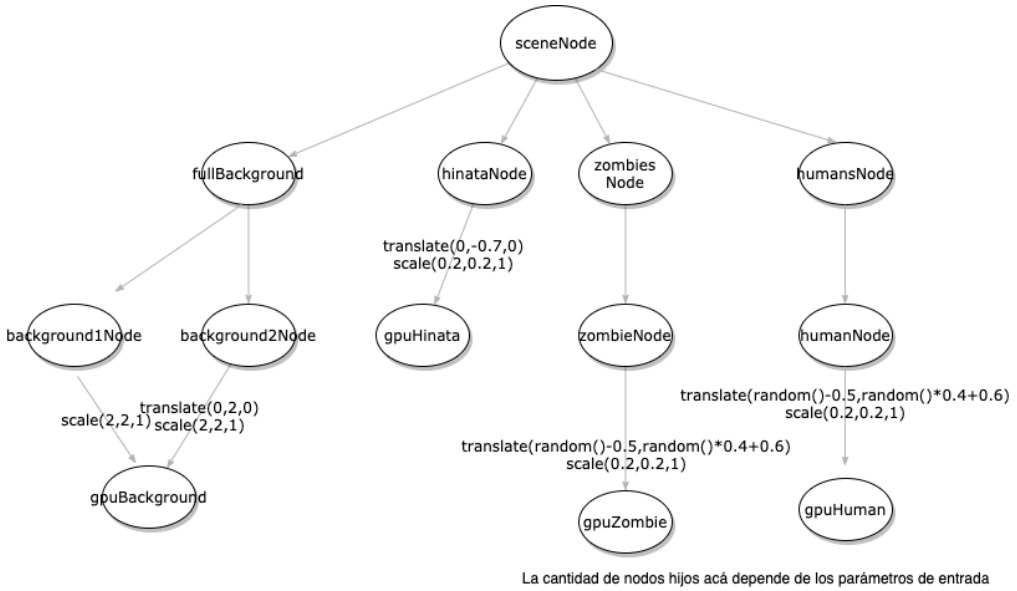
Universidad de Chile
fcfm - dcc
CC3501

Reporte Tarea N1

Valentina Garrido
10 de mayo de 2021

Solución Propuesta

La solución propuesta para este problema es una aplicación donde se dibujan distintos grafos según lo que esté pasando en el programa. Mientras el juego estñe en funcionamiento, es decir, mientras el jugador siga sano, se dibuja el nodo de escena compuesto de un fondo en movimiento, el jugador, junto a zombies y humanos que aparecen cada T segundos. Luego, si el jugador llega a un punto en el que debería transformarse en zombie (esto por chocar contra un zombie o por estar contagiado al tocar a un humano contagiado), el juego termina al dibujarse un mensaje de "GAME OVER". Por otro lado, si el tiempo total del juego, que es de 60 segundos, se termina y el jugador sigue sano, el fondo deja de moverse y se dibuja un nodo que contiene una tienda generada por formas geométricas. Después, cuando el jugador llega a la tienda, se termina el juego al dibujarse el nodo que contiene el mensaje de "YOU WIN".



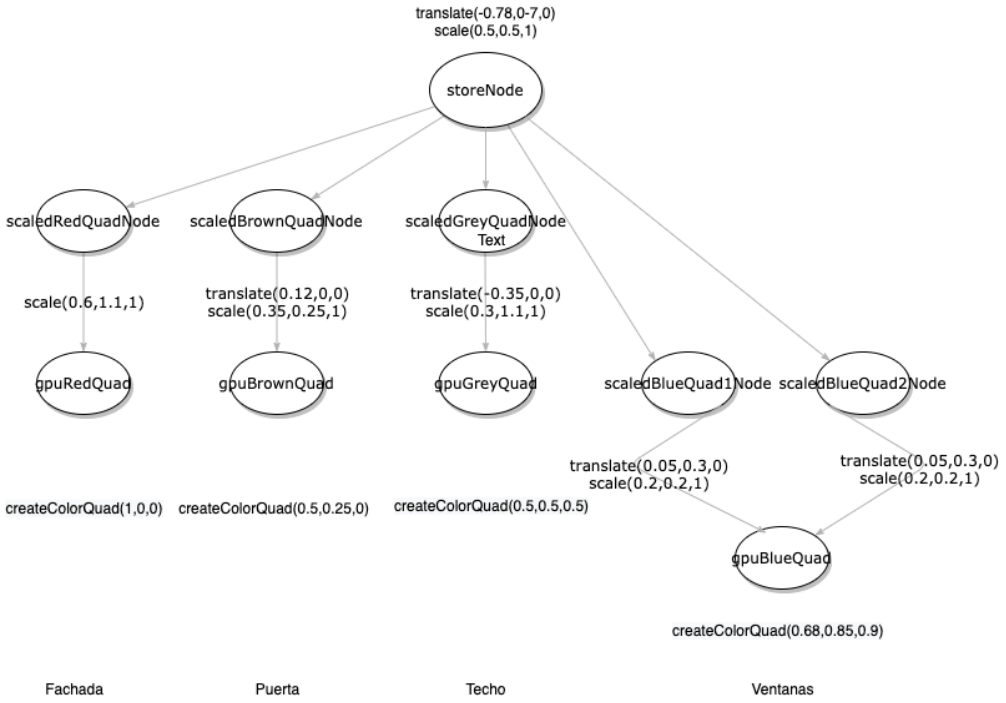
Se utilizan modelos para el jugador (Hinata), humanos, zombies, el fondo y la tienda. Esto es util para mover todas estas figuras y generar choques entre el jugador y los otros personajes y la tienda al final del juego.

Dentro de los modelos de los personajes, se crean atributos "status", que pueden ser "Healthy", "Infected" y "Dead". Al inicio del juego los status de Hinata y los zombie son "Healthy" y "Dead", respectivamente. Luego, los humanos pueden tener status "Healthy" o "Infected" al inicio, y esto se setea aleatoreamente con la librería random.

Si Hinata choca con otro personaje, este adquiere el status de este, por lo que si choca con un zombie, su status pasa a ser "Dead", por lo que el jeugo termina. Por otro lado, si choca

con un humano infectado, su comportamiento es igual al resto de estos personajes que tienen el miemo status.

Los modelos de Hinata y Humano tienen el atributo "life", que inicialmente es 1, y va disminuyendo P cada T segundos si es que el status del personaje es "Infected". Así, si life llega a 0, el status del personaje pasa a ser "Dead", por lo que termina el juego.



Luego, para hacer el juego más eficiente, se eliminan los nodos y los modelos de los jugadores (humanos y zombies) que ya no estén en pantalla. Esto se logra gracias a los atributos que de estos peronajes que detallan su posición.

Finalmente, para poder ver si los humanos están contagiados o no, se debe mantener presionada la tecla "V", con la que los humanos cuyo status es 'Ìnfected" se dibujan con un Shader que multiplica los colores de la textura por un vector que representa el verde en RGBA.

Instrucciones de Ejecución

- Librerias adicionales a instalar (solo si es necesario)
  - sys: Utilizada para poder agregar argumentos para correr el programa

- random: Genera números aleatoreos que se usan para determinar la posición inicial de zombies y humanos.

- Método de Ejecución

  Para ajacutar el programa `survival.py`, se deben agregar los atributos Z, H, T y P, donde Z y H son los números de zombies y humanos que se generan cada T segundos, y P corresponde a la probabilidad por segundo de que un humano infectado se transforme en zombie.

  En primer lugar, se crea el controlador con la clase Controller(), donde se agregan los atributos relacionados con las teclas que mueven al jugador, la tecla que hace que se visualice el status de los personajes y otros estados que son 'ènd" y "completed", que determinan si el jugador, al que se le asocia el controlador, ha perdido o ha terminado satisfactoriamente el juego, respectivamente.

  Luego, se define la función on_key que relaciona las teclas que se presionan con los cambios en los atributos del controlador.

  Posteriormente se genera la ventana con glfw, se activan las transparencias, se setea el color de fondo de la ventana y se procede a generar las figuras en GPU y algunos de los nodos más importantes. Los nodos que correspondan se asignan a modelos y luego, se inicia un ciclo while mientras la ventana esté abierta donde se van generando zombies y humanos, se actualizan posiciones y estados de los personajes, incorporando posibles choques. También, se revisa que el juego continue viendo que el status del jugador no sea "Dead".

  Luego, se incorpora la condición de que de presionarse la tecla adecuada, se dibujen los personajes infectados con un shader verde creado modificando uno de los shaders de textura del curso.

  Después, se dibujan los elementos del nodo de escena usando un shader simple de texturas de las librerías del curso. Posteriormente, si se cumplen condiciones para el fin del juego (controller.end y controller.completed), se dibujan las figuras que corresponden usando los shaders adecuados.

  Finalmente, si se cierra la ventana (con la tecla esc), se limpia la memoria eliminando los nodos de scena, de la tienda, y los que almacenan las texturas con los mensajes de "GAME OVER" y "YOU WIN".

- Modo de uso y/o teclas de control

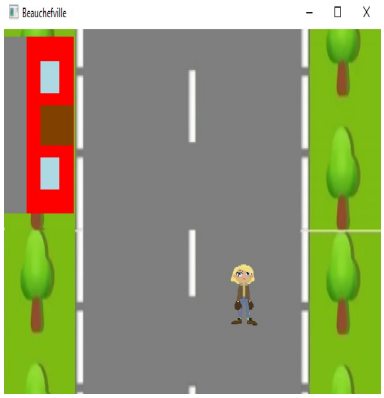
- Para mover al jugador, se deben presionar las teclas con flechas correspondientes a KEY_UP, KEY_DOWN, KEY_RIGHT y KEY_LEFT de glfw.

- Para visualizar los humanos infectados, se debe presionar la tecla "V".

Resultados

En las figuras se pueden visualizar los elementos principales de la aplicación, que son aquellos asociados a modelos. Se puede notar cómo se visualizan los humanos infectados al mantener presionada la tecla "V", además de la aparición de la tienda al final del camino cuando se acaba el tiempo, y por tanto, dejan de aparecer zombies y humanos a los que el jugador debe evadir.





Autoevaluación

Criterio-Puntaje	0	1	2	3
Evaluación General				x
OpenGL			x	
Shaders		x		
Modelos geométricos		x		x
Transformaciones			x	
Texturas		x		
Modelación jerárquica			x	
Curvas	x			
Funcionalidades mecánicas o lógica de juego				x
Entradas o Control de usuario				x
Visualización de estado del programa				x