

PROYECTO FINAL INFORMÁTICA MUSICAL

COMUNICACIÓN ENTRE SUPERCOLLIDER Y UNITY Y SONORIZACIÓN DE UN VIDEOJUEGO

Gonzalo Sanz Lastra

Jorge Rodríguez García

Objetivo

El objetivo principal del proyecto fue la posibilidad de comunicar Unity con SuperCollider mediante mensajes y utilizar este último como soporte para sonORIZAR un videojuego hecho en Unity. Esta sonorización consistiría en una melodía sencilla que iría cambiando dinámicamente en función de lo que ocurriese en el juego.

El juego elegido es “El laberinto del Sultán”, que consiste en recorrer un laberinto hasta encontrar todas las joyas escondidas en él y llevarlas a tu base, todo ello sin que te atrape el fantasma guardián o te quedes sin energía para seguir caminando. Al llevar joyas gastas más energía, por lo que a veces será buena idea soltarlas a mitad de camino para lograr llegar a tu base y reponer tu energía.

Implementación

Como el objetivo principal fue tener música dinámica que variase en función de lo que ocurriese en el juego, hemos creado un sistema de lo que hemos llamado "Modos". Cada modo tiene una serie de parámetros que le aplicará a la música: duración de las notas, amplitud, escala musical y octava, y serán los encargados de que la música cambie. Tenemos un total de 21 modos que presentan pequeñas variaciones en sus parámetros, de forma que el primer modo presenta parámetros de lo que sería una música reposada y el último de una música tensa. A medida que los modos aumentan y pretendemos ir hacia la música tensa, se reduce la duración, se aumenta la amplitud, se cambia de escala y en ocasiones se sube de octava.

Para elegir el modo a aplicar, tenemos en cuenta la distancia del jugador al fantasma, el número de joyas que quedan sin sacar del laberinto y la energía que le queda al jugador, ya que hemos considerado que eran las mecánicas que más pueden reflejar esa relación estrés-reposo que pretendemos. Para ello, teniendo X modos, calculamos en qué modo estaríamos en función de la distancia del jugador al fantasma, siendo el modo 0 de reposo si el jugador está a la máxima distancia, o el modo X de más estrés si estuviese a su lado. Calculamos de manera similar el modo en función de las demás mecánicas y posteriormente hacemos la media de todas ellas, dando lugar al modo medio de la situación global del juego.

Junto con estos modos, tenemos música ambiente escrita mediante un array de notas que hemos compuesto nosotros mismos y que sonará de fondo continuamente y en bucle, y un array de acordes que siguen la misma filosofía que los modos, una sucesión desde acordes de

reposo hasta acordes de tensión, apoyándonos en lo aprendido en los últimos temas de la asignatura.

Para que suene todo correctamente, enviamos a SuperCollider un mensaje cada segundo con la nota de música ambiente que corresponda reproducir, junto con el modo actual en el que nos encontremos, indicándole así con qué parámetros reproducirla. De la misma manera, cada vez que el usuario mueve al jugador, le enviamos a SuperCollider un mensaje con el acorde y modo en el que nos encontremos.

Por último, hemos hecho que tanto la música ambiente, los acordes y los modos sean leídos desde json. El resultado final es que el juego tiene en cuenta una serie de mecánicas para modificar dinámicamente tanto la música ambiente como los acordes que suenan al mover al jugador adaptándose a lo que ocurre en el juego, pudiendo modificar todo ello o la forma en la que varía desde un archivo externo .json.

Con todo ello, hemos conseguido implementar la idea principal que nos propusimos, añadiendo además una composición propia y conocimientos de la última parte del temario, que a nosotros en particular nos resulta más complicada que la programación, por lo que estamos bastante contentos con el resultado obtenido.

Tecnología utilizada

El juego está implementado en Unity 2019.2.1f1, utilizando Visual Studio Community 2019 para la generación de los scripts, escritos en el lenguaje C#. Para la sonorización se utiliza el programa SuperCollider y como formato de lectura de datos externos se utiliza JSON. Para la comunicación entre Unity y SuperCollider mediante mensajes se ha utilizado como referencia el repositorio <https://github.com/jorgegarcia/UnityOSC>, añadiendo pequeñas modificaciones.

El contenido del proyecto se puede encontrar en el siguiente repositorio de Github: <https://github.com/jorgerodrigar/Informatica-musical>, en la carpeta "Proyecto", ya que en el repositorio se encuentran además todos los ejercicios realizados en la asignatura. En concreto, los archivos más relevantes en la elaboración de la música dinámica se encuentran en "Assets/Scripts/DynamicMusic/" y el archivo "PlayerController.cs", desde donde se mandan los mensajes a SuperCollider. El archivo .scd de SuperCollider se encuentra en "Assets/SCFiles/", las melodías, acordes y modos se encuentran en "Assets/Music/" y los efectos de sonido (algunos hechos también por nosotros) en "Assets/Sounds/".

Posible trabajo futuro

Como posible mejora del trabajo, se podrían añadir más efectos de sonido o melodías y la posibilidad de saltar entre secciones de una canción a voluntad del diseñador, en vez de reproducir nota a nota como hasta ahora, pudiendo así producir composiciones más complejas, aleatorias y únicas.

Contribución de los integrantes

Aunque se puede hacer un seguimiento del trabajo a través del historial de commits del repositorio, éstos no implican el trabajo exclusivo del autor, ya que en la mayoría de ellos han trabajado todos los integrantes en conjunto.

La comunicación entre Unity y SuperCollider mediante mensajes, el sistema de modos y su implementación en una clase gestora de la música dinámica, los parámetros fundamentales de éstos, la lectura de json y demás cuestiones de la implementación se realizó en conjunto por ambos integrantes.

Jorge Rodríguez se centró algo más en obtener la idea del proyecto, así como de la tecnología necesaria para su implementación, y en la introducción de la sucesión de acordes presentes en el resultado final.

Gonzalo Sanz Lastra se centró en componer la melodía de fondo del juego, la variación de sus parámetros y en la inclusión de efectos de sonido como el de coger joyas, etc.

Instrucciones de uso

Si se quiere ejecutar el proyecto como tal, debe descargarse el repositorio antes citado e instalar Unity 2019.2.1f1 o superior, junto con Super Collider. Para ver el resultado, debe abrirse el archivo .scd de Super Collider, lanzar el servidor y compilar el contenido, y después dar al Play en la escena “Game” de Unity.