

**Diseño sonoro dinámico para la sonorización del videojuego Survival Shooter.**

Ana María Vásquez Avila

Universidad Nacional Abierta y a Distancia UNAD

Escuela de Ciencias Básicas, Tecnología e Ingeniería ECBTI

Tecnología en Producción de Audio

Junio de 2023

## Tabla de Contenido

Índice de figuras.....	3
Introducción .....	6
Objetivos.....	7
Objetivo General.....	7
Objetivos Específicos.....	7
Desarrollo del diseño sonoro dinámico del Videojuego Survival Shooter .....	8
Tarea 1: Análisis de un videojuego .....	8
Tarea 2: Generación de un ambiente sonoro dinámico .....	11
Tarea 3: Sonorización de un proyecto de videojuego .....	16
Tarea 4: Programación del diseño sonoro del videojuego .....	26
Tarea 5: Mezcla, pruebas y optimización de audio del videojuego .....	31
Conclusiones .....	36
Bibliografía .....	37

## Índice de figuras

Figura 1. Sayonara Wild Hearts corazones.....	9
Figura 2. Sayonara Wild Hearts personaje principal .....	9
Figura 3. Genero del videojuego.....	10
Figura 4. Categoría del sonido .....	10
Figura 5. Audios seleccionados para el diseño del ambiente.....	11
Figura 6. Audios seleccionados editados en Reaper .....	12
Figura 7. Audios de aire y pájaros en Fmod .....	13
Figura 8. Audios de lluvia suave y fuerte en Fmod .....	14
Figura 9. Audio de explosiones, gritos disparos y alarmas en Fmod.....	14
Figura 10. Multi Instrument con audios de carros, perros y pasos .....	15
Figura 11. Automatización del audio aire en Fmod .....	15
Figura 12. Automatización del audio pájaros en Fmod .....	15
Figura 13. Automatización del audio lluvia en Fmod.....	16
Figura 14. Música del videojuego Survival Shooter en Fmod.....	17
Figura 15. Multi Instrument con audios de risa y zombi en Fmod.....	17
Figura 16. Audio de reloj en Fmod .....	18
Figura 17. Audio de robot en Fmod.....	18
Figura 18. Audio de pasos en Fmod.....	18
Figura 19. Organización de los audios por personajes en Fmod .....	19
Figura 20. Bancos creados en Fmod.....	20
Figura 21. Refrescar bancos en Fmod.....	20
Figura 22. Asignación del audio música en Unity .....	21

Figura 23. Asignación del audio efectos 2 en Unity .....	22
Figura 24. Asignación del audio reloj en Unity .....	22
Figura 25. Asignación del audio robot en Unity .....	22
Figura 26. Asignación del audio de golpe al personaje principal .....	23
Figura 27. Asignación del audio de golpe a los enemigos .....	24
Figura 28. Asignación del audio de pasos a los enemigos .....	24
Figura 29. Asignación del audio de pasos al personaje principal .....	24
Figura 30. Asignación del audio ametralladora .....	25
Figura 31. Pistas editadas para la música adaptativa .....	26
Figura 32. Parámetro de música en Fmod .....	27
Figura 33. Automatizaciones en la melodía 2 y en la melodía base .....	27
Figura 34. Región de transición y marcador de tiempo .....	28
Figura 35. Condiciones de transición.....	28
Figura 36. Banco de la música adaptativa .....	29
Figura 37. Implementación de la música adaptativa a Unity .....	29
Figura 38. Box collider y música por zonas de tensión .....	30
Figura 39. Zonas de la música adaptativa .....	30
Figura 40. Ventana de mixer en Fmod .....	31
Figura 41. Grupo de efectos, enemigos, música y personaje principal.....	31
Figura 42. Zona de reverb en Unity .....	32
Figura 43. Box collider del reverb .....	32
Figura 44. Configuración del reverb en unity .....	32
Figura 45. Configuración de máxima instancia .....	33

Figura 46. Botón de live update.....	34
Figura 47. Primera prueba en el profiler de Fmod.....	34
Figura 48. Segunda prueba en el profiler de Fmod.....	34

## **Introducción**

El presente trabajo sobre audio dinámico para videojuegos permite conocer cada uno de los procedimientos necesarios para la sonorización de un videojuego (Eraso, 2022).

Estos procedimientos se ponen en práctica con el videojuego Survival Shooter, durante el proceso es necesario el manejo de programas como Fmod en el cual se realizan los efectos y la música del videojuego (Moreno Viasus, 2017) y Unity el cual contiene el videojuego y en el que encontramos procedimientos nuevos como lo son los scripts con los cuales se dan órdenes al programa para realizar distintas acciones y los triggers los cuales hacen que se realice un evento cuando es tocado (Moreno, 2021).

El desarrollo de este trabajo permite ampliar conocimientos en el área del audio para videojuegos, también brinda aprendizajes en el área del audio en 2D y 3D.

## Objetivos

### Objetivo General

Identificar los procesos para la elaboración de audio dinámico para videojuegos aplicando cada una de las herramientas y procesadores necesarios para obtener una sonorización adecuada.

### Objetivos Específicos

Analizar distintos videojuegos para así adquirir los conocimientos necesarios para la sonorización de un videojuego.

Memorizar el manejo del programa Fmod en el cual se realizará la sonorización del videojuego.

Identificar los procedimientos que se realizaran en el programa Unity para la implementación del audio diseñado para el videojuego.

Definir cada una de las necesidades del videojuego Survival Shooter en el cual se va a trabajar.

Aplicar los conocimientos adquiridos sobre los programas Fmod y Unity para realizar el audio dinámico para el videojuego Survival Shooter.

Evaluar el audio dinámico realizado para el videojuego Survival Shooter garantizando una calidad adecuada y el cumplimiento de sus necesidades.

## **Desarrollo del Diseño Sonoro Dinámico del Videojuego Survival Shooter**

El diplomado de audio dinámico para videojuegos cuenta con cinco tareas las cuales nos permiten conocer cada elemento necesario para la sonorización de un videojuego.

En cada tarea adquirimos nuevos conocimientos que fueron aplicados en el videojuego Survival Shooter.

### **Tarea 1: Análisis de un Videojuego**

En esta tarea se trabajó en un videojuego de libre elección, el objetivo de esta era identificar cada uno de los conceptos aprendidos para la tarea 1, realizando el análisis de un videojuego, en el cual se debía tener en cuenta la categoría del sonido y el tipo de audio dinámico con el que cuenta.

Para el desarrollo del trabajo se escogió el videojuego Sayonara Wild Hearts, en el cual podemos encontrar soundtracks de bandas de pop al momento de jugarlo. El objetivo del juego es recorrer niveles y conseguir puntos por medio de corazones que van apareciendo en el camino.

Se logro identificar que el videojuego cuenta con una categoría de sonidos de efectos, ambiente y música. Se puede percibir como la música se sincroniza con cada uno de los movimientos que realiza el personaje, los efectos que se perciben son los de las motos que hacen parte de algunos niveles del videojuego, otro efecto es el del contacto del personaje principal con los corazones, al momento de tener contacto con el corazón, el videojuego y la música tiene un silencio el cual le da tiempo al jugador de realizar el movimiento con el personaje.

Se considera que el videojuego cuenta con un tipo de audio dinámico de interactividad debido a que el audio cuenta con una reacción con los corazones en los distintos niveles del videojuego, también se reconoce que cuenta con un tipo de audio de variabilidad, debido a que



los beats de la música están sincronizados con los corazones por lo que la música no se reproduce hasta que el personaje principal tenga contacto con los corazones.

Figura 1. Sayonara Wild Hearts corazones.

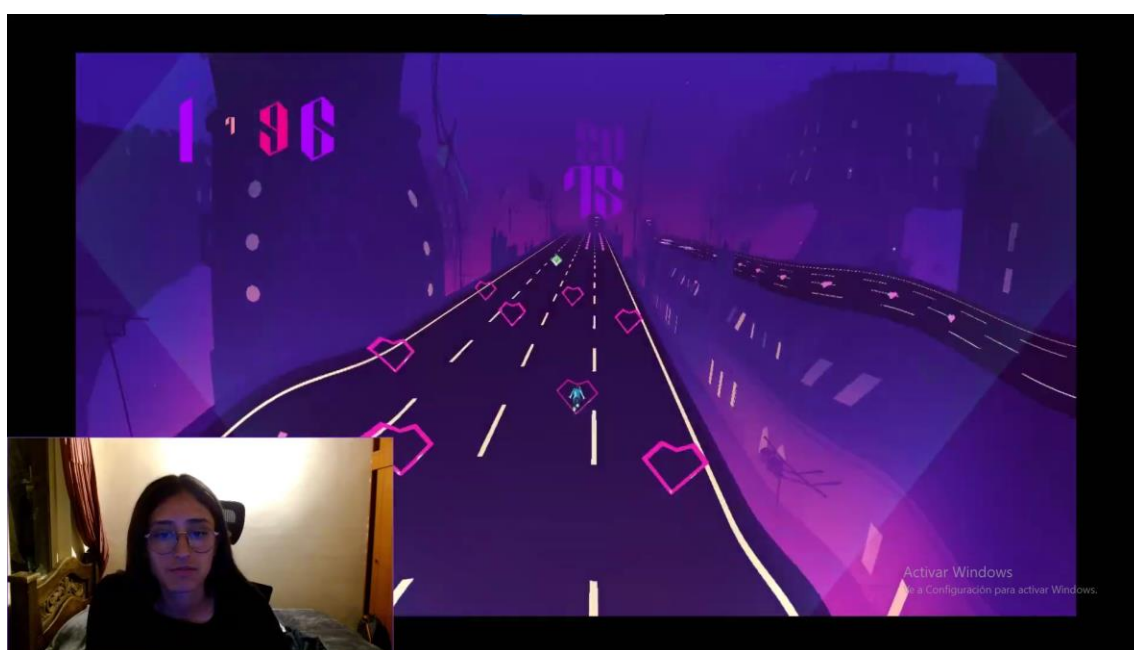


Figura 2. Sayonara Wild Hearts personaje principal

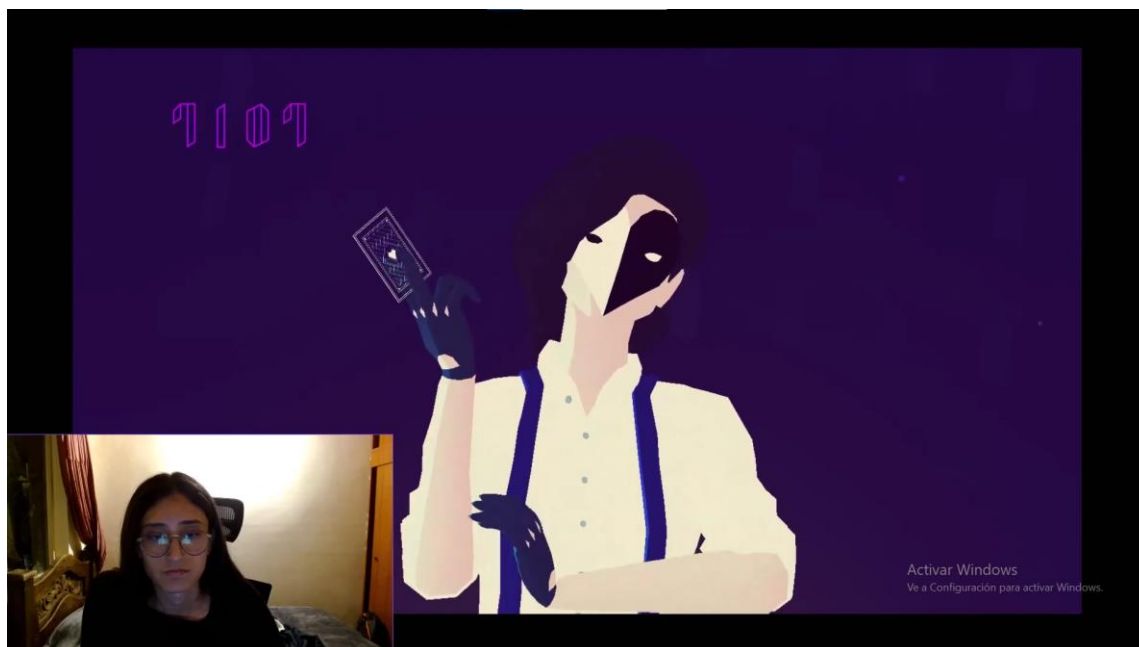


Figura 3. Genero del videojuego



## GÉNERO DEL VIDEOJUEGO

VIDEOJUEGO DE ACCIÓN  
VIDEOJUEGO DE RITMO  
VIDEOJUEGO CASUAL

Figura 4. Categoría del sonido



## **CATEGORÍA DEL SONIDO:**

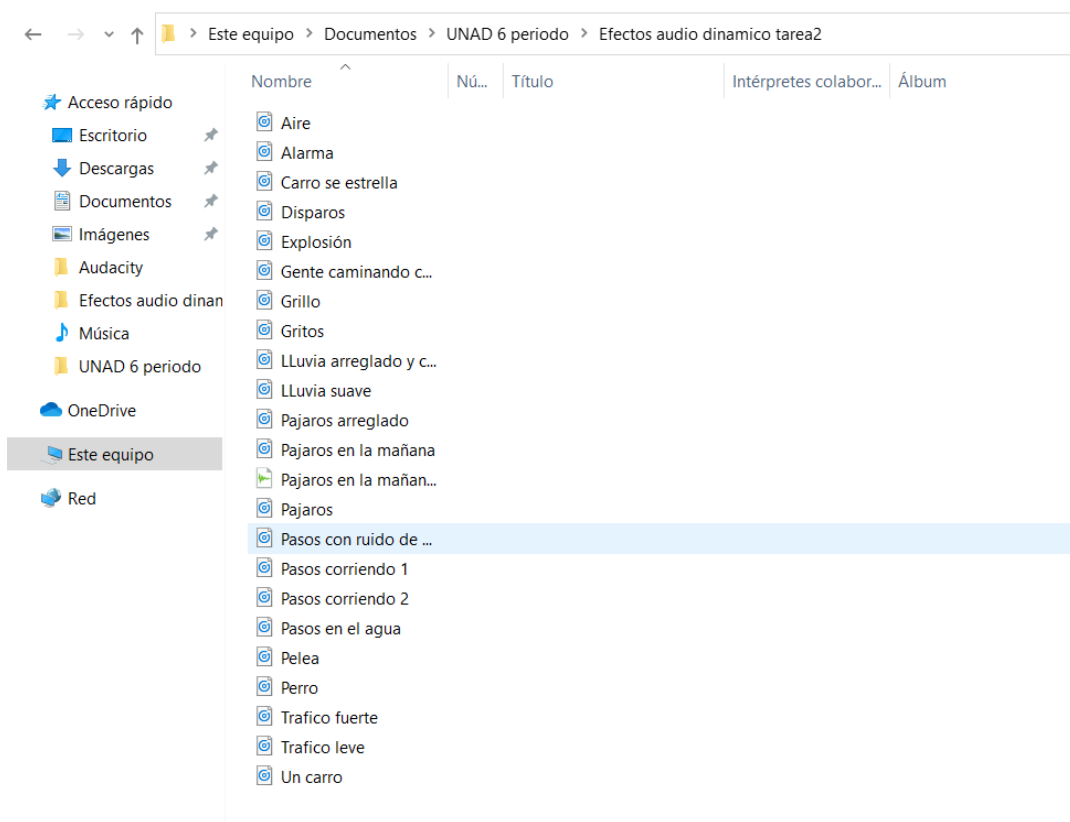
- **EFFECTOS**
- **AMBIENTE**
- **MÚSICA**

### **Tarea 2: Generación de un Ambiente Sonoro Dinámico**

En esta tarea se trabajó en la generación de un ambiente sonoro dinámico de libre elección, en este debíamos tener en cuenta los siguientes parámetros, el ambiente generado está configurado a 24 horas las cuales representan las horas del día, el ambiente generado debe contar con un ambiente de lluvia, ya sea suave o fuerte y por último se debe aplicar un parámetro de libre elección con nombre y rango que informe como el parámetro añadido modifica el evento sonoro.

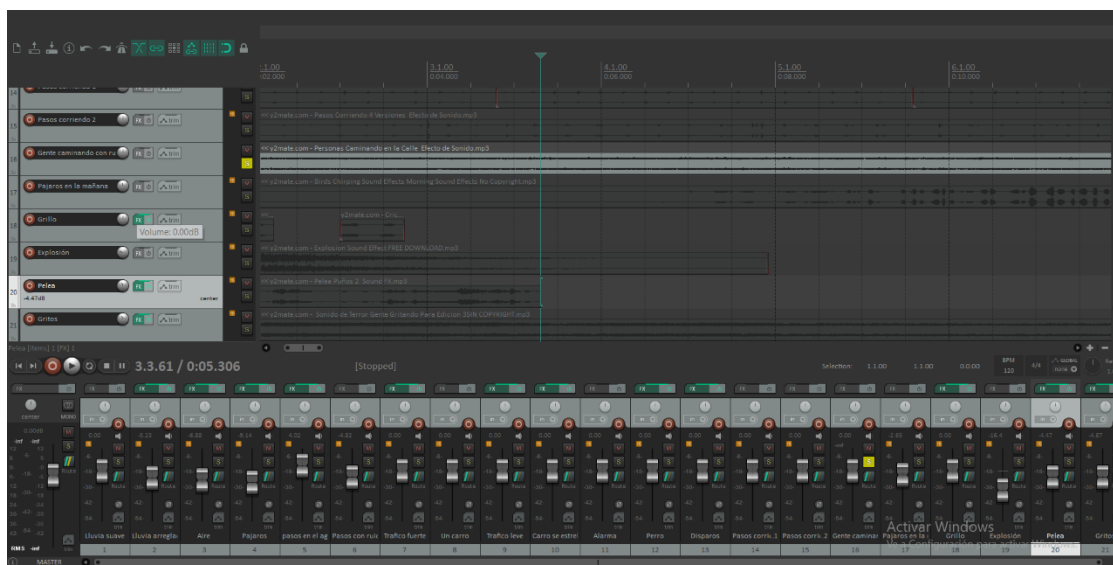
Lo primero que se desarrolló de la tarea fue la selección de algunos efectos de audio con lo que se cuentan y que serían necesarios para el diseño del ambiente.

Figura 5. Audios seleccionados para el diseño del ambiente



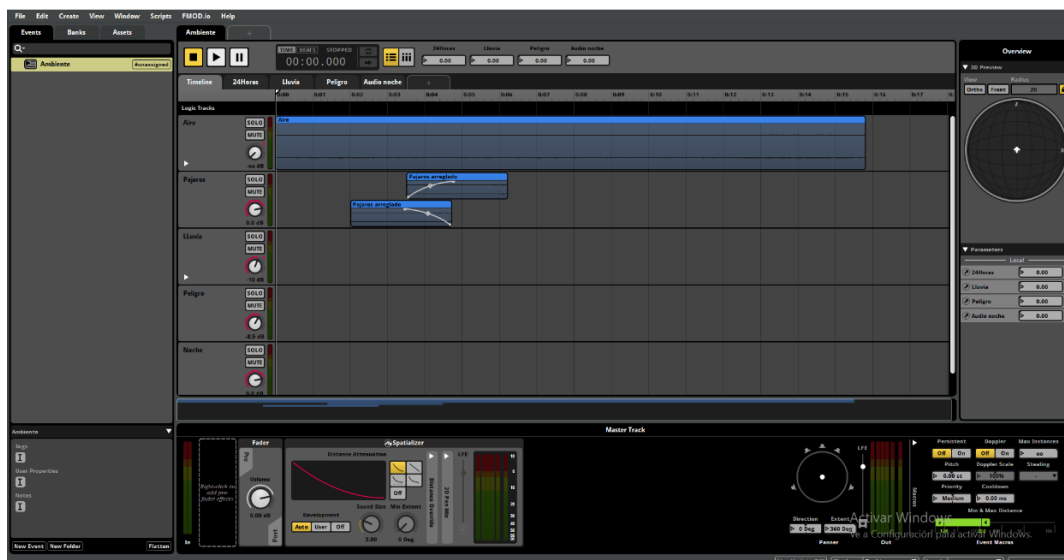
Es importante que cada uno de los audios seleccionados cuenten con la edición correcta que permita obtener un loop adecuado, por ello el segundo paso realizado fue pasar cada uno de los audios a la estación de trabajo Reaper, en la cual se desarrolló la edición de los audios obteniendo un loop adecuado para el ambiente sonoro que se estaba realizando.

Figura 6. Audios seleccionados editados en Reaper



Con los audios listos, lo siguiente fue ingresar en el programa Fmod, en el cual se creó el primer evento nombrado ambiente, en este se desarrolló el parámetro del rango 24, el cual representa las horas del día, en este se realizó la base del ambiente, por lo cual se implementaron los audios de aire y pájaros.

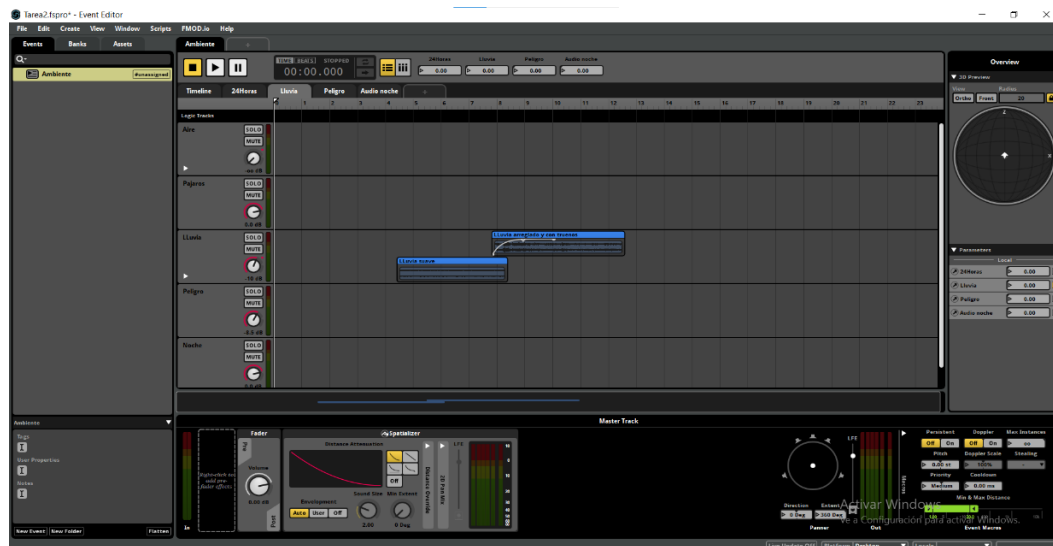
Figura 7. Audios de aire y pájaros en Fmod



Posteriormente se desarrolla el segundo parámetro nombrado como lluvia, en el cual se implementan dos tipos de lluvia, suave y fuerte. Estas son configuradas con el fin de aumentar la

intensidad de la lluvia de manera progresiva. Junto al efecto de lluvia se implementaron efectos de carros los cuales van aumentando de manera progresiva con la lluvia.

Figura 8. Audios de lluvia suave y fuerte en Fmod



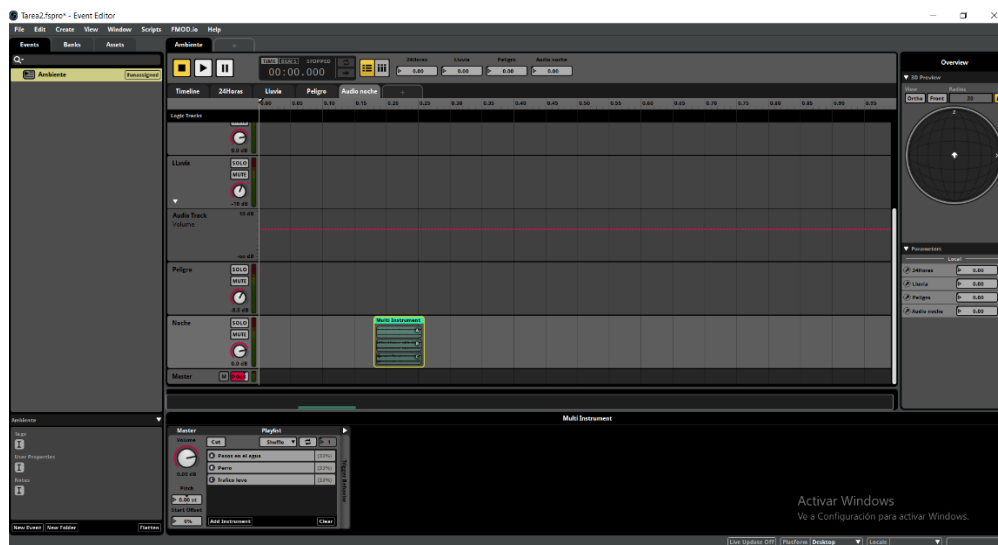
Luego de esto se desarrolló el parámetro de libre elección, el cual es nombrado peligro, debido a que en el punto más fuerte de lluvia se van a percibir efectos como explosiones, gritos, disparos y alarmas. Estos efectos se configuran para conseguir una secuencia adecuada con el fin de demostrar con claridad la intensidad de estos efectos dentro del diseño del ambiente sonoro.

Figura 9. Audio de explosiones, gritos disparos y alarmas en Fmod



Después se implementó el ultimo parámetro que es nombrado audio noche, en este se usó la herramienta Multi Instrument, en el cual se implementó un audio de carros con una intensidad menor al que fue aplicado en el parámetro de lluvia, perro ladrando y pasos en una superficie húmeda.

Figura 10. Multi Instrument con audios de carros, perros y pasos



Como paso final se automatizaron algunos audios con el fin de ajustar la intensidad de cada uno dependiendo del momento del día en el cual se encuentre.

Figura 11. Automatización del audio aire en Fmod

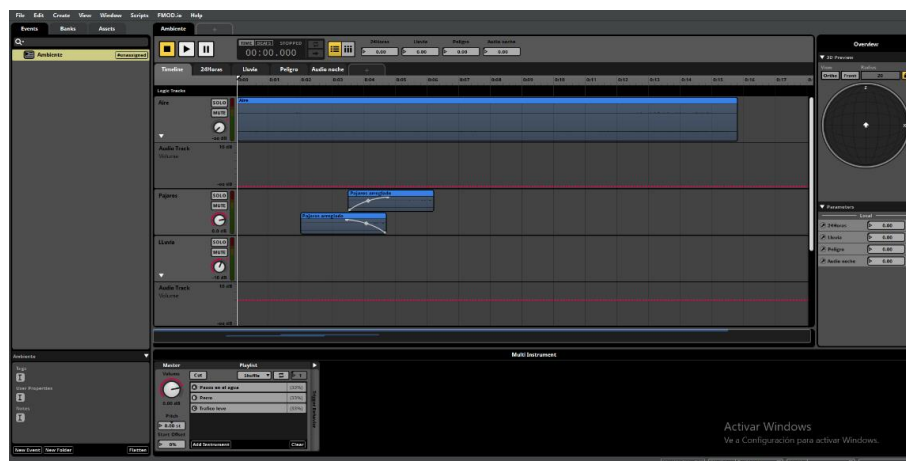


Figura 12. Automatización del audio pájaros en Fmod

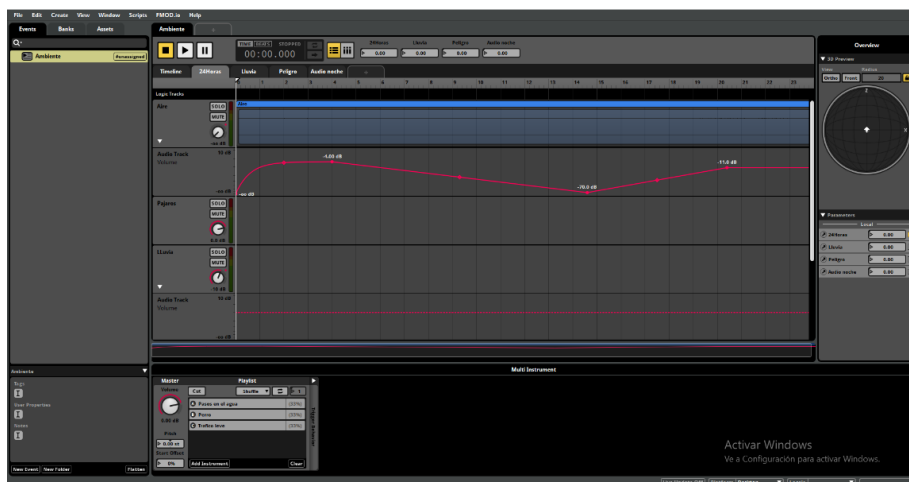
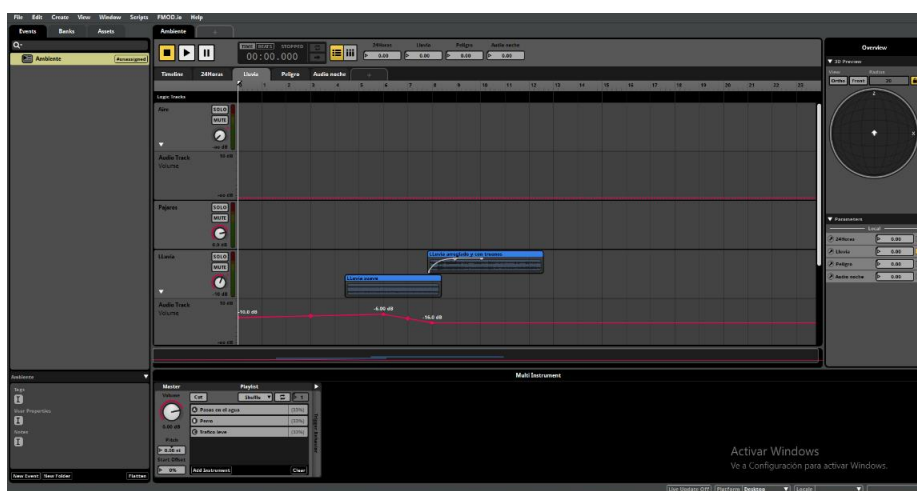


Figura 13. Automatización del audio lluvia en Fmod



### Tarea 3: Sonorización de un Proyecto de Videojuego

En esta tarea se trabajó en la sonorización del videojuego Survival Shooter, el cual debía incluir sonido ambiente, efectos sonoros y música.

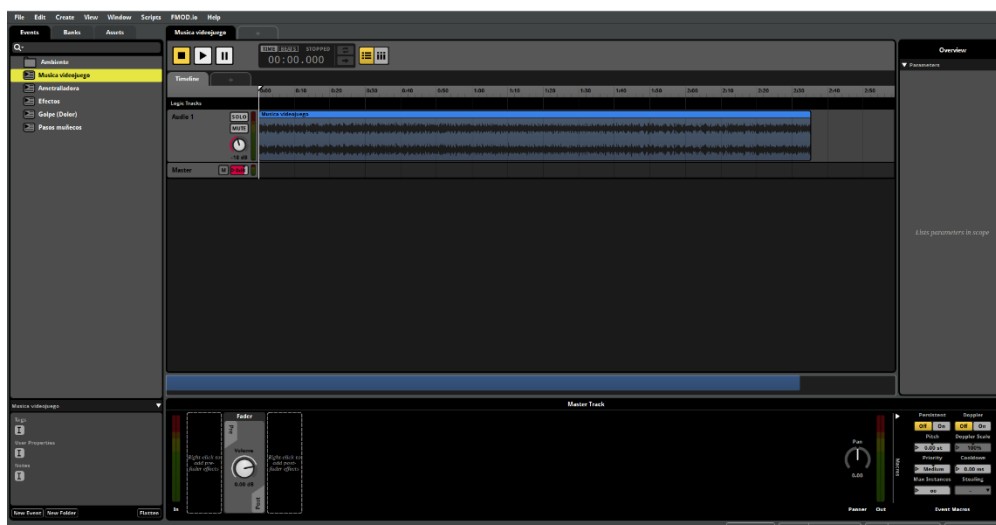
Lo primero fue la instalación del programa Unity, ya con el programa instalado se abrió el juego y se exploró para así poder identificar cada uno de los efectos que serían necesarios.

Posteriormente se pasaron los audios necesarios a la estación de trabajo Audacity en la cual se editaron cada uno de los audios seleccionados para obtener un loop adecuado. Los audios implementados fueron de ametralladora, golpe, risas y pasos.



Luego de esto se abre un nuevo proyecto en Fmod, en este se crea una carpeta nombrada ambiente, en esta se crean los eventos necesarios, el primero que se creo fue la música, la cual fue realizada por medio de la aplicación bandlab e implementada, se tuvo precaución con el volumen manejado para poder encontrar un balance con los demás efectos que serán implementados.

Figura 14. Música del videojuego Survival Shooter en Fmod



Los efectos son divididos en 2 grupos, los efectos en los cuales se agregaron los audios del reloj y del robot, y en los efectos 2 se utilizó un multi Instrument con los efectos de risa y zombi.

Figura 15. Multi Instrument con audios de risa y zombi en Fmod

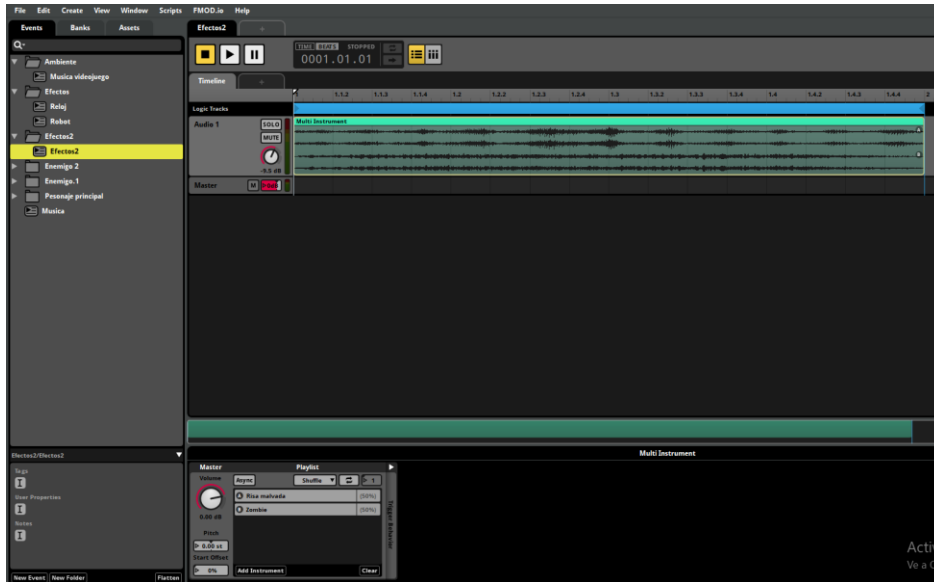


Figura 16. Audio de reloj en Fmod

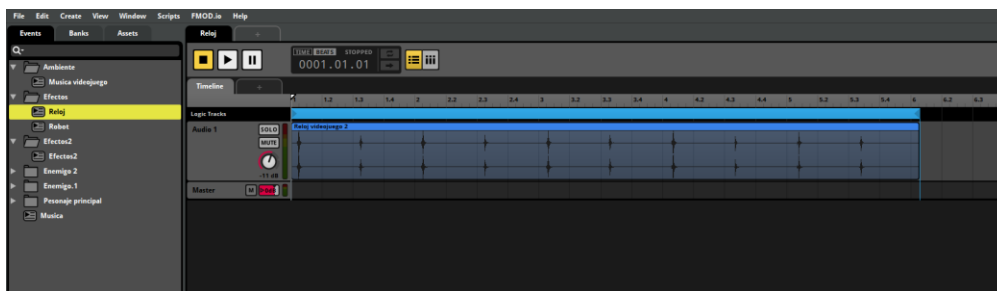
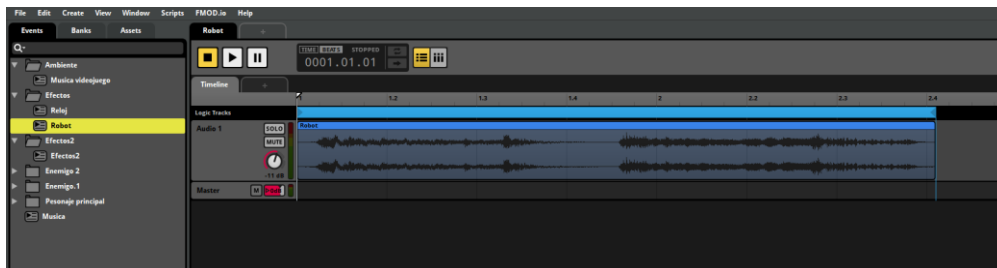
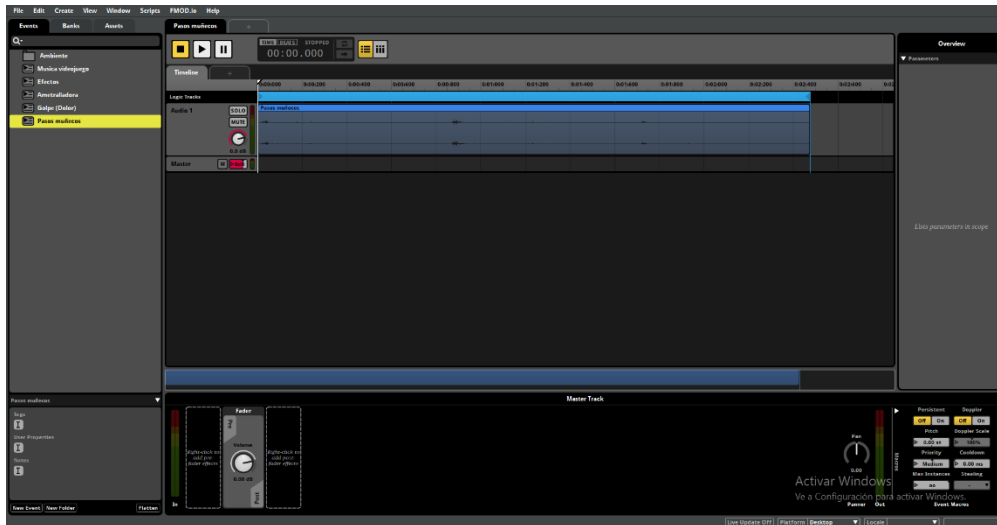


Figura 17. Audio de robot en Fmod



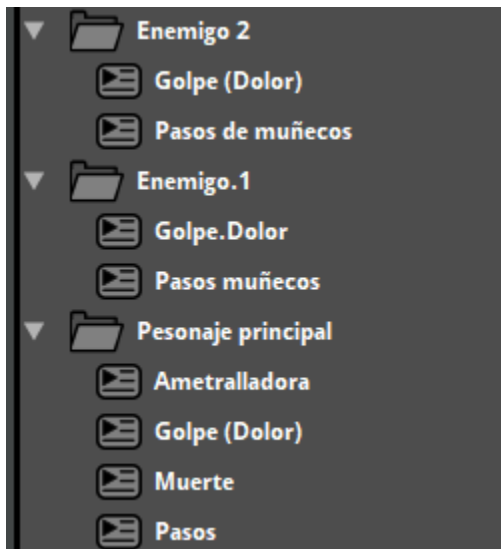
Posteriormente se añadieron los audios restantes, la ametralladora con un volumen de 0.0db, el efecto de dolor con un volumen de -15db y los pasos con un volumen de 0.0db.

Figura 18. Audio de pasos en Fmod



Se crearon carpetas por personajes para así tener organizados los audios, los enemigos cuentan con audio de golpe y pasos, el personaje principal cuenta con audio de golpe, pasos, ametralladora y muerte.

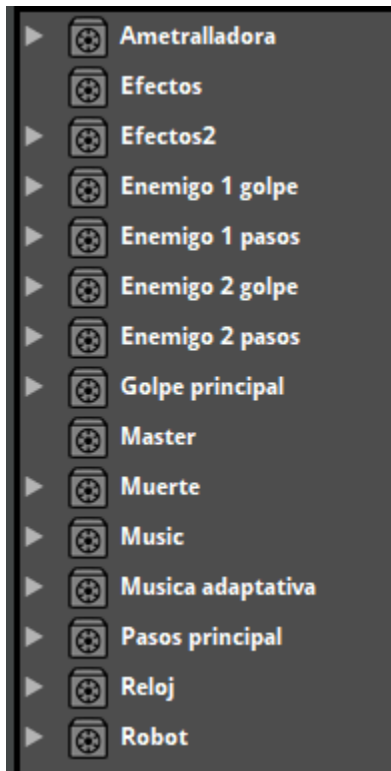
Figura 19. Organización de los audios por personajes en Fmod



Es importante tener en cuenta que cada uno de los audios está configurado con una zona de loop, la cual permite escuchar el audio de manera continua.

Luego de esto se crearon bancos para ametralladora, efectos, golpe, music y pasos divididos por personajes. En este se organizaron cada uno de los audios editados con el fin de que Unity pueda reconocerlos y reproducirlos de forma adecuada.

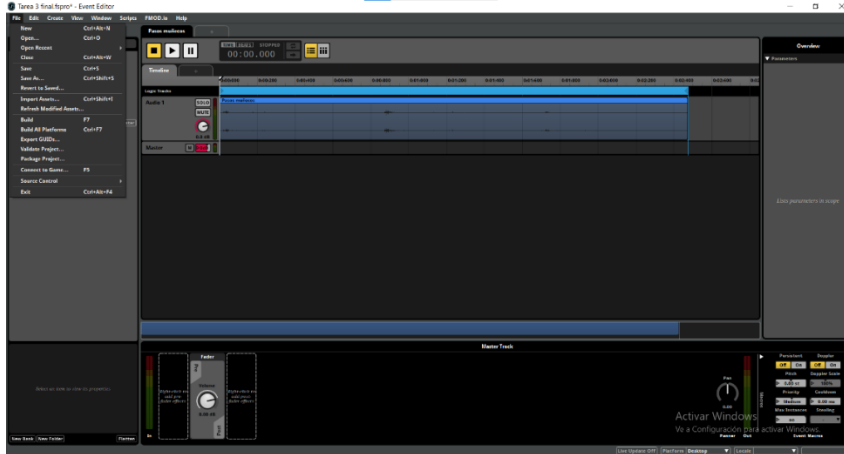
Figura 20. Bancos creados en Fmod



Después en Unity se incluye el programa Fmod, con el archivo en el cual se implementaron cada uno de los audios, la main camera del personaje principal fue modificada para que recibiera los audios que eran enviados desde el programa Fmod.

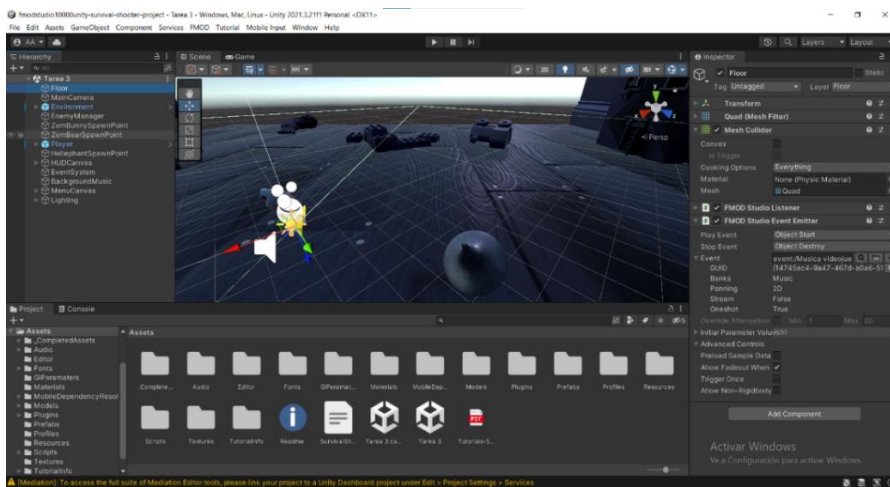
Luego de que Fmod y Unity se sincronicen de forma adecuada se busca el objeto al cual se le asignara el audio, se busca en los bancos y se selecciona, es importante recordar que para los bancos van a aparecer en Unity con los cambios realizados si son refrescados, por lo cual es necesario en Fmod ir a la pestaña file y luego seleccionar la opción build.

Figura 21. Refrescar bancos en Fmod



Para asignar el audio de la música en Unity, se selecciona la pestaña floor en la cual se añaden los componentes Fmod Studio Listener y Fmod Studio Event Emitter en el cual se agrega el audio que fue asignado para la música en Fmod, se le aplico una configuración para que la música se reproduzca cuando inicie el juego y se detenga cuando el juego termine, se decidió poner la música en la pestaña para poder percibir la música durante toda la escena del videojuego.

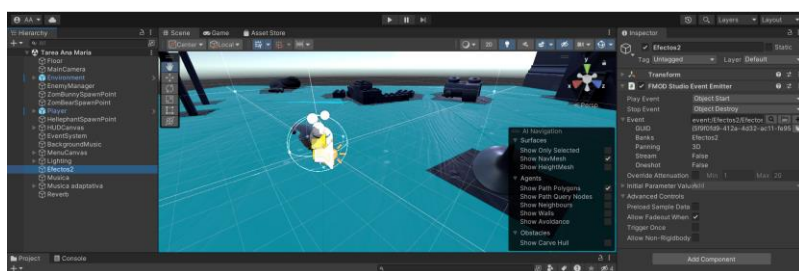
Figura 22. Asignación del audio música en Unity



Los efectos son divididos en 2 grupos, los efectos en los cuales se agregaron los audios del reloj y del robot, y en los efectos 2 se utilizó un multi Instrument con los efectos de risa y zombi.

Para la implementación de los efectos 2 se creó un empty nombrado efectos 2, el cual se ubica en la mitad de la escena, en este se aplica un Fmod Studio Event Emitter el cual está configurado para que el audio empiece cuando inicie el juego y termine cuando finalice el juego.

Figura 23. Asignación del audio efectos 2 en Unity



Y para los efectos, se añadieron a los objetos, el audio del reloj se aplicó al reloj y el robot se aplicó al robot añadiendo el componente Fmod Studio Event Emitter, el cual está configurado para que el audio empiece cuando inicie el juego y termine cuando finalice el juego, este cuenta con una configuración por distancia por lo que el audio se percibe cuando el personaje está cerca del objeto.

Figura 24. Asignación del audio reloj en Unity

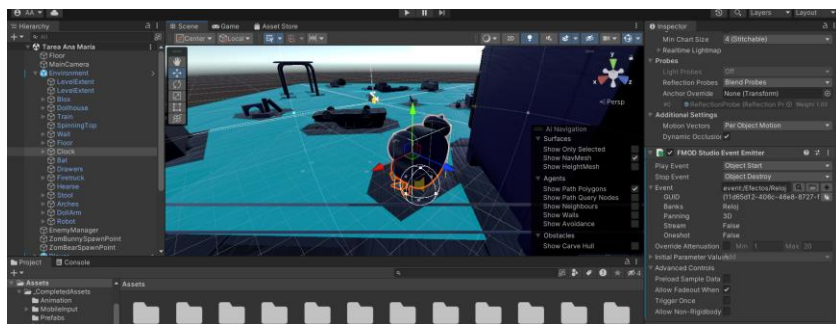
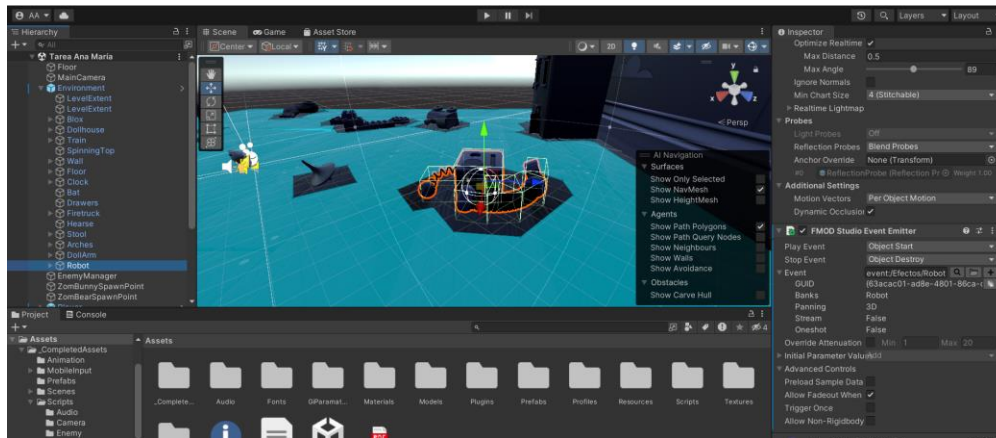
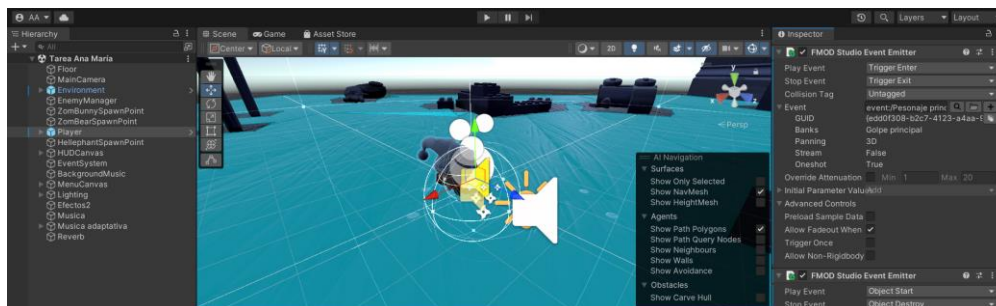


Figura 25. Asignación del audio robot en Unity



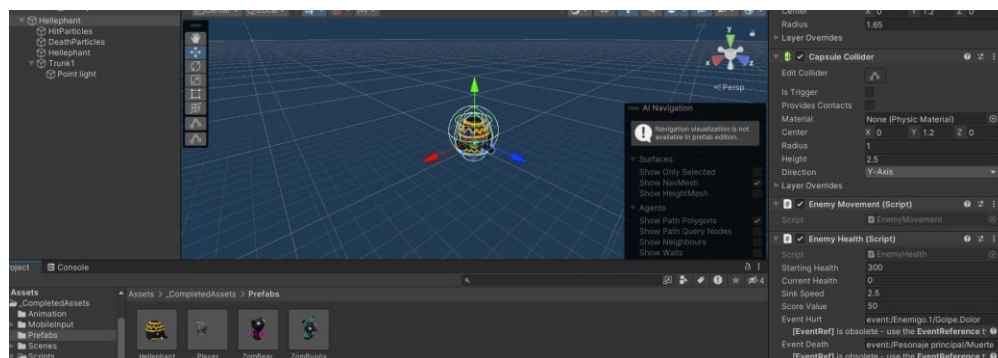
Para el audio golpe del personaje principal, se añadieron los mismos componentes en la pestaña player, pero la configuración es distinta ya que se necesita que el audio se reproduzca cuando el personaje reciba un golpe, por lo cual se configuro para que inicie en audio trigger enter y que finalice en audio trigger exit, en la pestaña collision tag se seleccionó la opción tag untagged.

Figura 26. Asignación del audio de golpe al personaje principal



Para el audio golpe de los enemigos, se buscó la pestaña Prefabs en la cual se encuentran los enemigos se elige el enemigo al cual se le aplicara el audio, para que funcione de forma adecuada es importante utilizar un script el cual le dirá al programa que cada vez que el personaje sea tocado por el enemigo reproduzca el audio que se aplicó, el script listo se agrega en la pestaña del enemigo con el nombre Enemy Health (Script) y en este se aplican los audios de golpe y muerte, el mismo procedimiento se realiza con los demás enemigos.

Figura 27. Asignación del audio de golpe a los enemigos



Para la implementación del audio de los pasos, se busca la pestaña models y en esta se selecciona la pestaña characters, en el cual se encuentran los enemigos y el personaje principal. Se selecciona el personaje al que se le va a aplicar el audio en este en el cuadro move, se agregan dos eventos los cuales son los pasos el cual fue nombrado play foot y en este se agregó el audio realizado para los pasos.

Figura 28. Asignación del audio de pasos a los enemigos

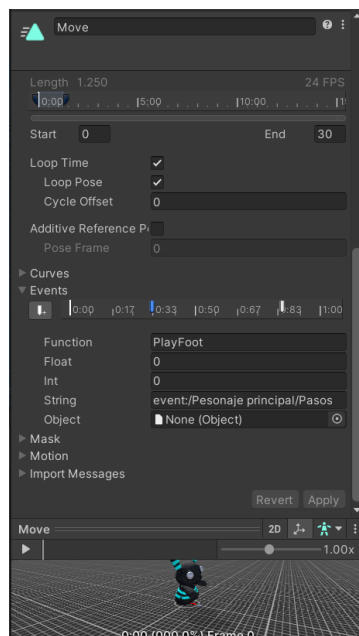
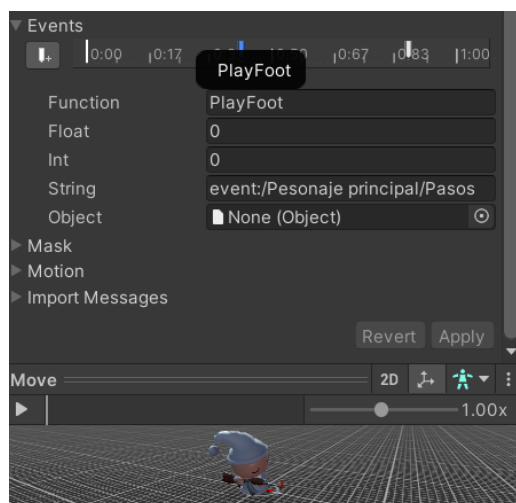


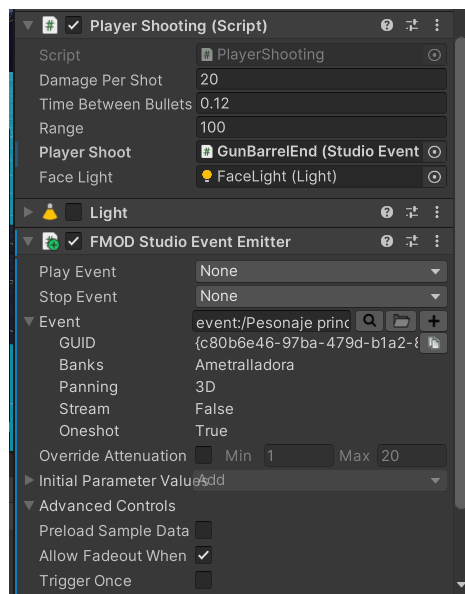
Figura 29. Asignación del audio de pasos al personaje principal





Para la implementación del audio de la ametralladora se realiza en la pestaña de player GunBarrelEnd, para que se reproduzca de forma adecuada es necesario un script el cual le dirá al programa que cada vez que se pulse el botón de la pistola suene el audio que aplico, el script listo se agrega con el nombre Player Shooting (Script) y se añade un Fmod Studio Event Emitter solo con el audio de la ametralladora sin ninguna configuración y este se agrega al script.

Figura 30. Asignación del audio ametralladora

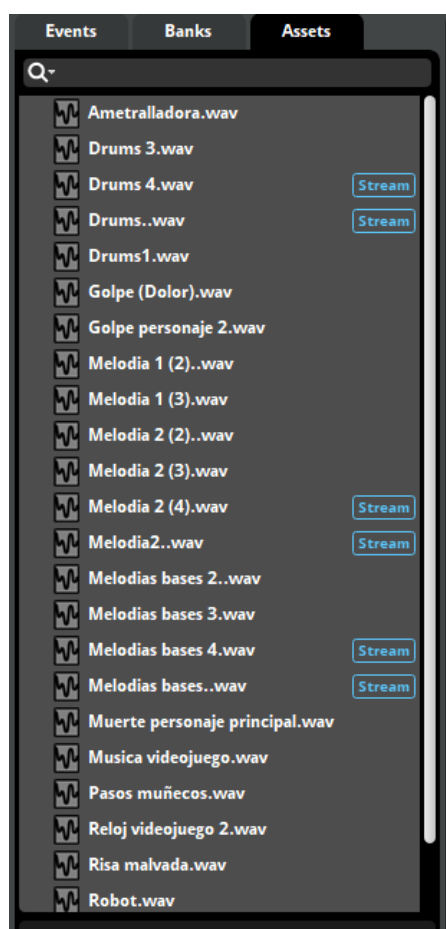


## Tarea 4: Programación del Diseño Sonoro del Videojuego

En esta tarea se trabajó la música adaptativa con cambios de tensión.

Para esto se utilizó la música implementada anteriormente y se decidió crear 4 momentos de tensión durante el juego, para esto fue necesario crear 4 pistas distintas para cada una de las zonas de tensión, cuando ya se encontraban editadas cada una de las pistas fueron añadidas en el programa Fmod.

Figura 31. Pistas editadas para la música adaptativa

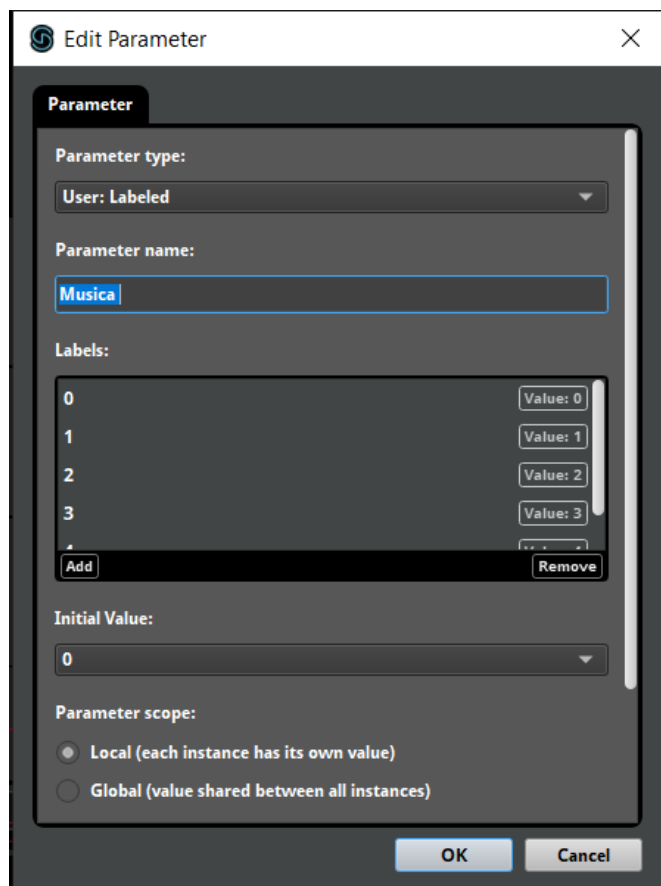


Posteriormente creamos un evento en 2d en el cual se implementó dentro de este evento una pestaña para cada audio de las 4 pistas los cuales fueron drums, melodía 1, melodía 2, melodías bases. Para tener un orden adecuado en la implementación de los audios para cada pista

se añadieron marcadores, los cuales fueron nombrados fase 1, fase 2, fase 3 y fase 4. Con esto ya ordenado se implementaron cada uno de los audios y se añadió una región de loop.

Luego de esto se añadió un parámetro nombre música el cual fue configurado tipo Labeled.

Figura 32. Parámetro de música en Fmod



En este parámetro se realizaron automatizaciones en la melodía 2 y en la melodía base, el objetivo de estas automatizaciones fue aumentar progresivamente el volumen en el cambio de una fase a otra, permitiendo a la música ser más dinámica en los cambios de intensidad.

Figura 33. Automatizaciones en la melodía 2 y en la melodía base



Por último, se añadió una región de transición la cual permite pasar de una fase a otra dependiendo de la configuración establecida, las transiciones quedaron configuradas para realizarse a 1/4 de nota, para que este cambio se realice adecuadamente es necesario añadir un marcador de tiempo el cual quedo en 140bpm 4/4.

Figura 34. Región de transición y marcador de tiempo

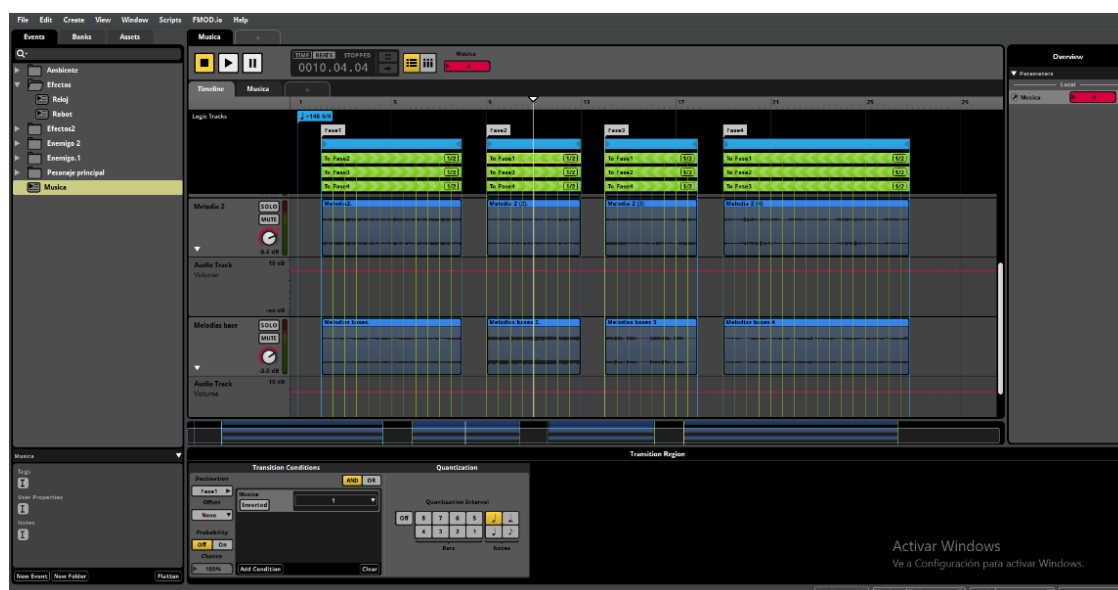
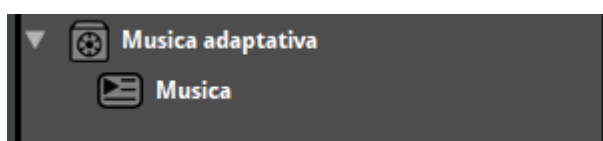


Figura 35. Condiciones de transición



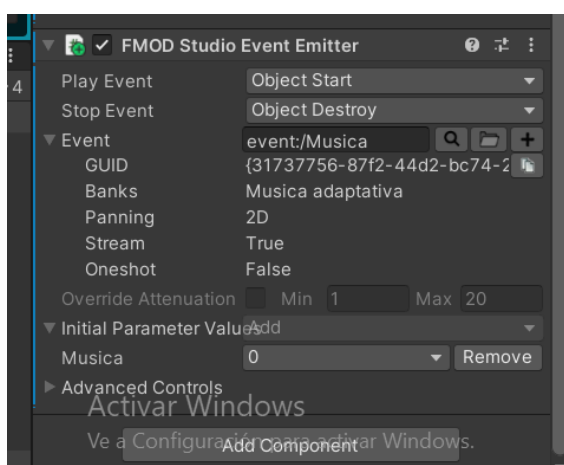
Con este procedimiento realizado, se crea un banco que es llamado música adaptativa y se refrescan los bancos, con el fin de poder visualizarlos en Unity al momento de añadirlo.

Figura 36. Banco de la música adaptativa



Después se pasó la música de la pestaña floor al player, en este se configuro para que la música inicie con el juego y finalice cuando termine el juego, otra configuración importante que se aplico es el inicio de la música para que esta inicie en el punto 0 configurado en el programa Fmod.

Figura 37. Implementación de la música adaptativa a Unity



Luego se creó un empty con la música, en este se añadieron 3 empty más que corresponden a cada una de las zonas de tensión que tendrá la música, en estos se añadieron dos

componentes, el primero un box collider el cual fue configurado como un trigger y el segundo un Fmod Studio Parameter Trigger en el cual se agregó la música desde el player, y se configuró para que el audio se reproduzca cuando el personaje entre en cada una de las zonas asignadas. Cada uno de los empty se configuró para que la música empezara a reproducirse en distintos puntos.

Figura 38. Box collider y música por zonas de tensión

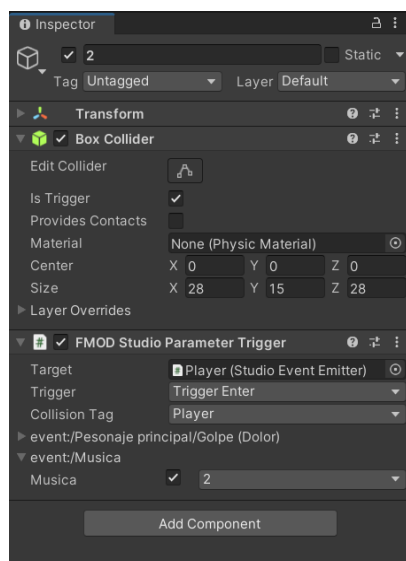
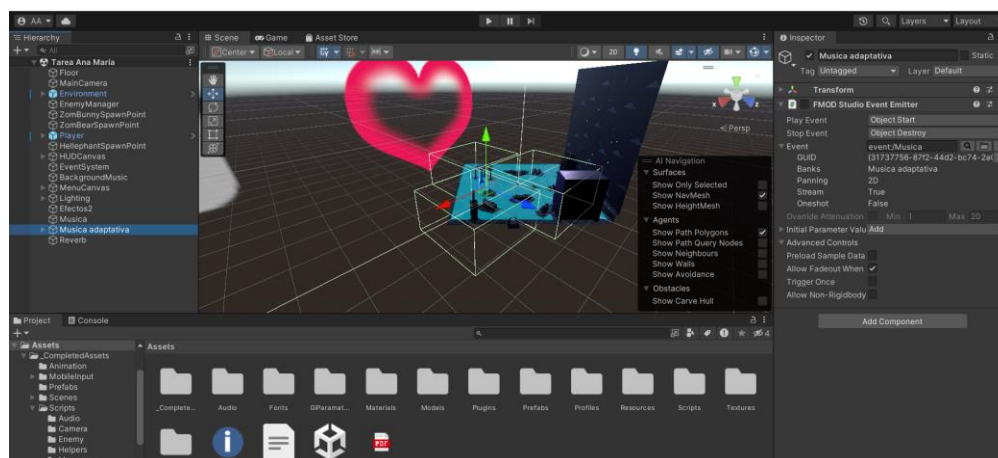


Figura 39. Zonas de la música adaptativa

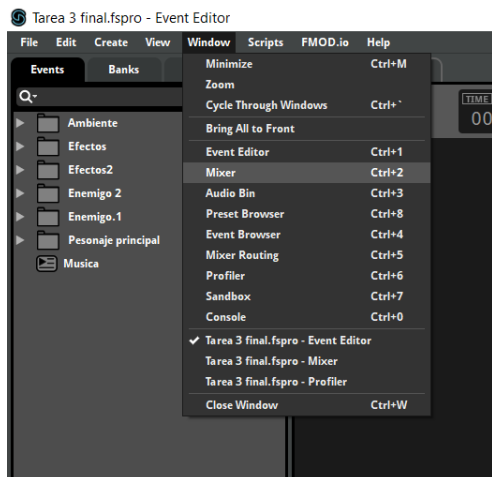


## Tarea 5: Mezcla, Pruebas y Optimización de Audio del Videojuego

En esta tarea se trabajó la mezcla y optimización del audio del videojuego que se está trabajando.

Lo primero que se realizó fue ingresar al proyecto en Fmod y abrir la ventana de mixer.

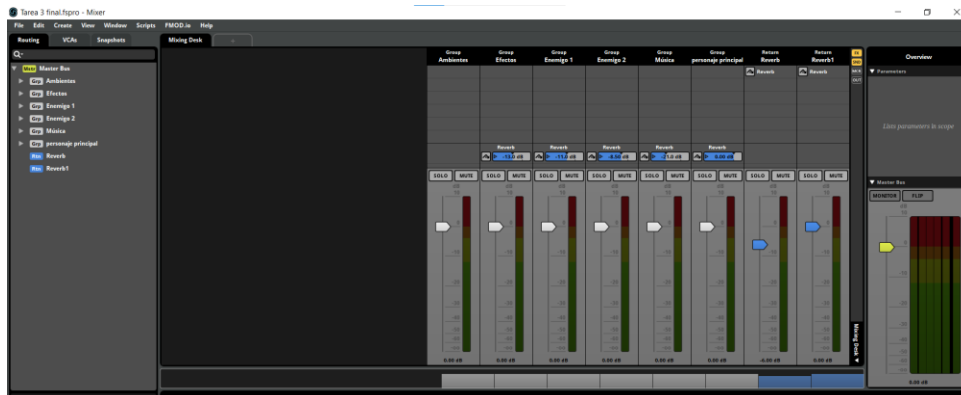
Figura 40. Ventana de mixer en Fmod



En esta se realizaron grupos con cada uno de los audios implementados en el Event editor.

Se añadió una sesión de Reverb en el cual se añadieron los efectos, los enemigos, la música y el personaje principal.

Figura 41. Grupo de efectos, enemigos, música y personaje principal



Posteriormente en Unity se añadió un box collider el cual se configuro como trigger y se agregó un Fmod Studio Event Emitter, el cual fue configurado para que inicie con un trigger enter y termine con un trigger exit, siendo el player el que activa el trigger. Con este proceso realizado se pasa a verificar que suene correctamente la reverberación aplicada.

Figura 42. Zona de reverb en Unity

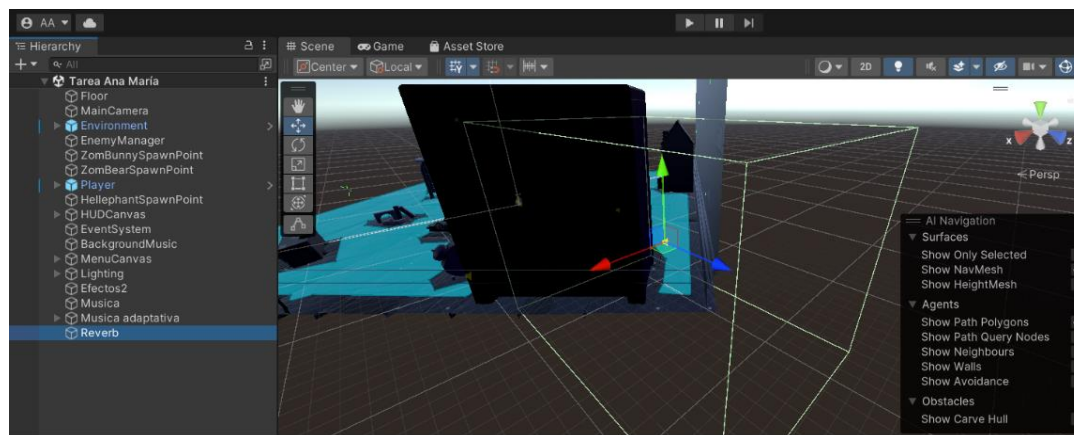


Figura 43. Box collider del reverb

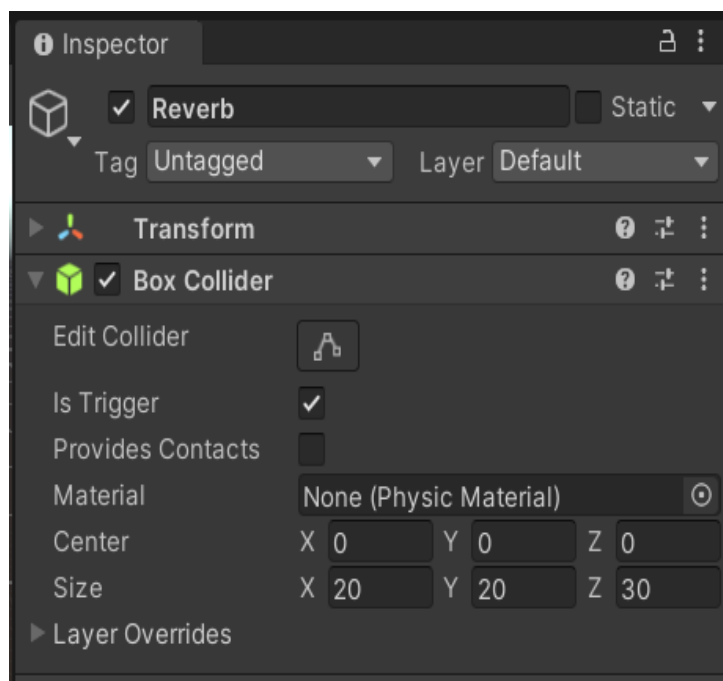
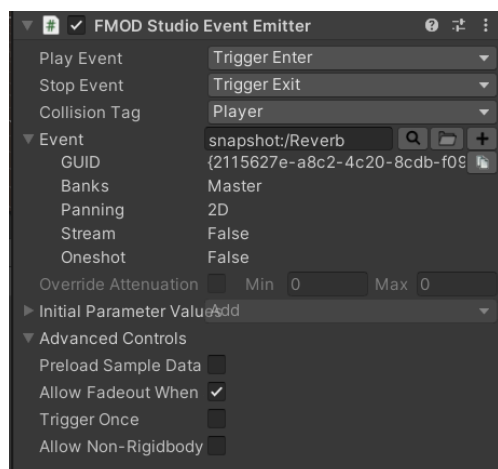


Figura 44. Configuración del reverb en unity

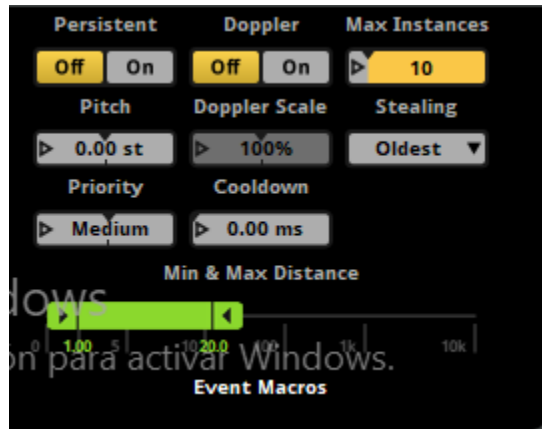




Luego de esto se realizaron unos cambios en la máxima instancia de los efectos enemigo 2, golpe y pasos quedando con un valor de 10.

Al efecto del enemigo 1, golpe y pasos se le realizo el mismo cambio y la máxima instancia quedo en 10.

Figura 45. Configuración de máxima instancia



Se refrescaron los bancos con el fin de sincronizar con Unity cada uno de los cambios realizados.

Posteriormente se realizó una prueba en la ventana de profiler, la cual permite conocer cuanta memoria se está gastando y los niveles máximos de Peak de cada uno de los audios implementados.

Para poder realizar la prueba era necesarios darle play al juego, con el juego funcionando, en Fmod se tenía que activar la opción Live update, el cual permite realizar cambios en el audio mientras el juego este activo. Con esta sincronización realizada se activa la grabación para iniciar la prueba.

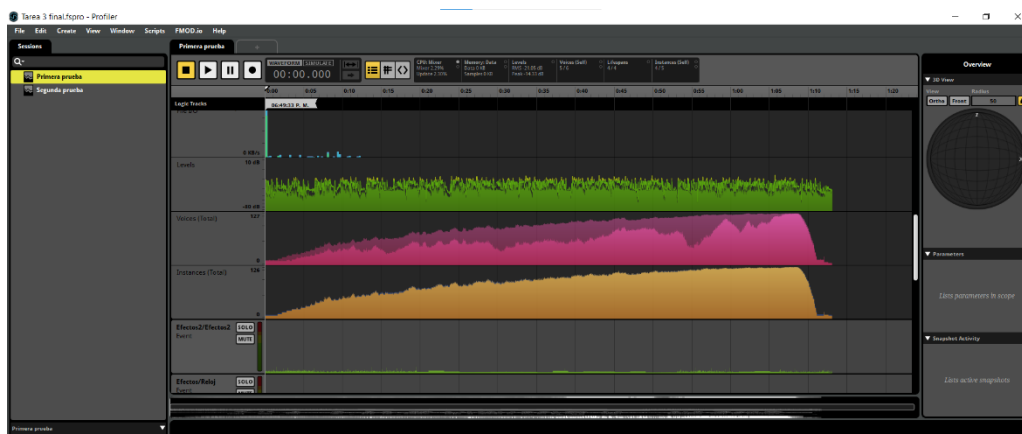
Figura 46. Botón de live update



Se realizaron dos pruebas, la primera se aplicó antes de agregar la reverberación y la segunda se aplicó después de la reverberación con el fin de conocer si contábamos con picos de saturación.

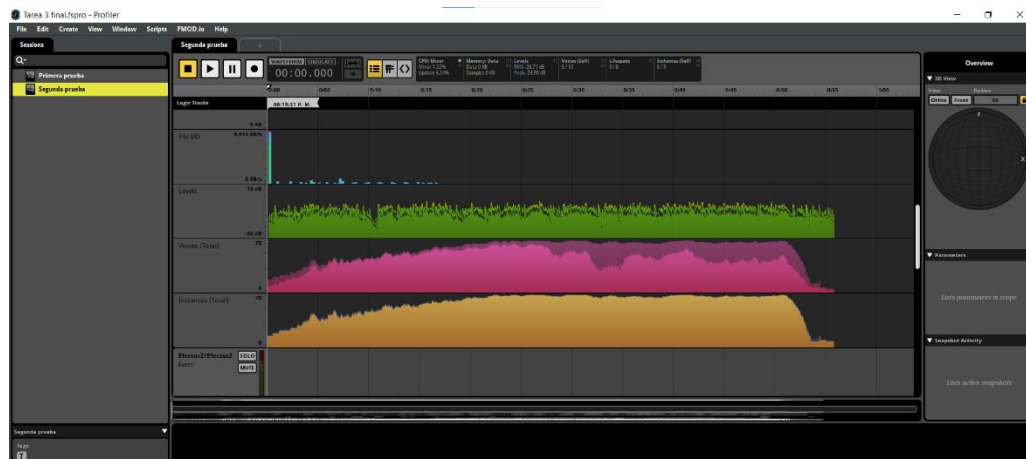
Primera prueba:

Figura 47. Primera prueba en el profiler de Fmod



Segunda prueba:

Figura 48. Segunda prueba en el profiler de Fmod



## Conclusiones

Se realizaron 5 tareas las cuales fortalecieron el conocimiento adquirido previamente y nos permitieron adquirir nuevos conocimientos, los cuales nos ayudaron durante el desarrollo de la sonorización del videojuego Survival Shooter.

Se desarrollo el diseño sonoro para el videojuego Survival Shooter, teniendo en cuenta cada uno de los procesos aprendidos, se adquirieron conocimientos sobre el manejo de programas como Fmod y Unity los cuales fueron fundamentales para el desarrollo de la sonorización del videojuego.

Se experimentaron algunas dificultades en la sincronización de los programas Fmod y unity las cuales fueron solucionadas satisfactoriamente.

Se identifican puntos importantes durante el desarrollo de la sonorización los cuales son la planeación y el orden, al tener planeado todo lo que se implementara en el juego y tener organizados los audios en Fmod es más sencillo la sincronización de los programas.

Se cumplieron los objetivos planteados obteniendo un resultado satisfactorio en el diseño del audio dinámico del juego Survival Shooter.

El diplomado de profundización en audio dinámico para videojuegos proporciona nuevos conocimientos en el manejo de los programas Unity y Fmod, profundizando en nuevas herramientas que podremos aplicar a nuevos proyectos, proporcionando unas bases sólidas las cuales garantizan un trabajo adecuado en la sonorización de videojuegos.

## Bibliografía

- Brandon, A. (2006). Game Audio Integration. Mix, 30(3), 56.  
<https://bibliotecavirtual.unad.edu.co/login?url=https://search.ebscohost.com/login.spx?direct=true&db=f6h&AN=20042238&lang=es&site=eds-live&scope=site>
- Eraso, C. (2022). Audio dinámico en los videojuegos. [Objeto\_virtual\_de\_aprendizaje\_OVA]. Repositorio Institucional UNAD. <https://repository.unad.edu.co/handle/10596/53360>
- Martínez, I. (2021). Música y sonido para videojuegos. Press Music.  
<https://pressmusic.com/musica-y-sonido-para-videojuegos/>
- Moreno Viasus, R. (15,02,2017). Audio Middlewares y Herramientas de Diseño Sonoro para Videojuegos. [Archivo de video]. <http://hdl.handle.net/10596/11604>
- Moreno Viasus, R. (10,03,2017). FMOD Studio Introducción. [Archivo de video].  
<http://hdl.handle.net/10596/11795>
- Moreno Viasus, R. (10,03,2017). FMOD Studio Single Instrument. [Archivo de video].  
<http://hdl.handle.net/10596/11789>
- Moreno Viasus, R. (10,03,2017). FMOD Studio Multi Instrument. [Archivo de video].  
<http://hdl.handle.net/10596/11788>
- Moreno Viasus, R. (10,03,2017). FMOD Studio Event Instrument. [Archivo de video].  
<http://hdl.handle.net/10596/11794>
- Moreno Viasus, R. (10,03,2017). FMOD Studio Scatterer Instrument. [Archivo de video].  
<http://hdl.handle.net/10596/11793>
- Moreno Viasus, R. (10,03,2017). FMOD Studio Programmer, command, snapshot, and Plug-in Instruments. [Archivo de video]. <http://hdl.handle.net/10596/11777>

Moreno, R. (2021). Tutorial triggers y parámetros en Fmod para una escena de Unity. Game Audio.

[https://ramorenov.github.io/gameaudio/04\\_Fmod\\_triggers\\_y\\_parametros.html](https://ramorenov.github.io/gameaudio/04_Fmod_triggers_y_parametros.html)