

V(add10) Versión 1.0

Diseño del software

Por: Alejandro De la Espriella Mazo
Dany Stevan Rios Madrid
proyectotesisvideojuego@gmail.com

Artes de la Grabación y Producción Musical
Facultad de Artes y Humanidades
ITM
2021

Contenido

1. FICHA TÉCNICA DEL PROYECTO	4
2. CONTEXTUALIZACIÓN	4
3. MOTIVACIÓN Y PROPÓSITO	7
4. CONCEPTOS DE DISEÑO	7
5. DISEÑO DE DETALLE	12
6. MÚSICA ADAPTATIVA	14
7. JUGADOR COMO INTÉRPRETE MUSICAL	16
8. DISEÑO SONORO.....	22
9. GLOSARIO.....	24
10. REFERENCIAS.....	25

Lista de tablas

Tabla 1. Problema implementación de música interactiva.....	9
Tabla 2. Elementos de evaluación	9
Tabla 3. Problema usuario final en contexto narrativo	9
Tabla 4. Elementos de evaluación de evaluación	10
Tabla 5. Elementos sonoros de evaluación.....	10
Tabla 6. Problema jugabilidad en amplio rango de edades de usuarios	10
Tabla 7. Elementos de evaluación mecánicas de juego.....	10
Tabla 8. Matriz de selección.....	11
Tabla 9. Funciones generales del software.....	12

Lista de imágenes

Imagen 1. Diagrama funcional del software	7
Imagen 2. Fases de producción.....	8
Imagen 3. Diseño de sistema general	11
Imagen 4. Flujo de trabajo y funcionamiento del software.....	13
Imagen 5. Carpeta contenedora de scrips	14
Imagen 6. Sesión de Ableton de composición de música	15
Imagen 7. Diagrama de funcionamiento del puzzle auditivo	17
Imagen 8. Scrip del puzzle auditivo 1.....	18
Imagen 9. Scrip del puzzle auditivo 2.....	18
Imagen 10. Scrip del puzzle auditivo 3.....	19
Imagen 11. Scrip del puzzle auditivo 4.....	19
Imagen 12. Sesión FMOD del puzzle auditivo.....	20
Imagen 13. Partitura puzzle auditivo	21
Imagen 14. Sesión de Studio one de diseño sonoro 1.....	23
Imagen 15. Sesión de Studio one de diseño sonoro 2.....	23
Imagen 16. Sesión de FMOD para diseño sonoro interactivo	24

1. FICHA TÉCNICA DEL PROYECTO

TÍTULO	V (add10)
TIPO DE PROYECTO	Investigación-creación
MODALIDAD	Producto derivado de actividades de desarrollo tecnológico e innovación
TIPO DE PRODUCTO	Diseño de software
PROGRAMA ACADÉMICO	Artes de la Grabación y Producción Musical
ESTUDIANTES	Alejandro De la Espriella & Dany Stevan Rios Madrid
ASESOR	Oscar Alejandro Cardoso Guzmán

2. CONTEXTUALIZACIÓN

V (add10) es el nombre de un proyecto de diseño y desarrollo de software, que corresponde a un videojuego en formato 2D (dos dimensiones) que reúne mecánicas y conceptos tradicionales del ámbito del entretenimiento interactivo. Permitió a sus desarrolladores explorar, profundizar y experimentar diferentes tipos de técnicas para la producción de música adaptativa y diseño sonoro dinámico.

El formato 2D en los videojuegos ha recobrado popularidad, hasta el punto de convertirse en una tendencia: los desarrolladores están retomándolo, aplicando nuevos conceptos de diseño, pero destacando la estética vintage en sus productos. A diferencia del formato 3D, el 2D permite abordar proyectos con presupuestos relativamente bajos, presentándose como la primera opción entre desarrolladores indie¹ de videojuegos.

A continuación, se mencionarán algunos videojuegos, destacando su contexto histórico, variables de diseño, jugabilidad, sonido, etc.:

The legend of Zelda fue una franquicia que impactó al mundo con la forma en que fusionó la narrativa con la jugabilidad, representando en su momento una forma novedosa de contar historias heroicas por medio de la interacción, a tal punto que influenció la generación de juegos posteriores, como citan los creadores del mismo en una entrevista de celebración del vigésimo quinto aniversario del juego: “Echando la vista atrás, *The Legend of Zelda: Ocarina of Time*, fue un proyecto que determinó el futuro [...]” (Iwata, Kondo, Yokota, & Osawa, 2011).

La franquicia *Final Fantasy* es un clásico mundialmente reconocido por sus innovadoras mecánicas y su profunda historia dentro del género RPG que, al igual que *The legend of Zelda*, cambiaría el rumbo de los juegos futuros (Iwata, Kondo, Yokota, & Osawa, 2011)²

¹ Indie: Es un género de videojuego creado por desarrolladores o empresas pequeñas que no cuentan con una financiación de alto presupuesto. Están fuera de la producción de empresas a gran escala y de alto presupuesto que atiende más al desarrollo de un concepto artístico.

² “The FINAL FANTASY series has always striven to push the envelope, try new things, and advance the genre and the game industry as a whole [...] FINAL FANTASY games (especially the numbered “main series” entries) have developed a reputation for cutting-edge graphics, sweeping musical scores, new and innovative battle and character

Ambas franquicias -y juegos como *Heavy Rain*, *Fahrenheit*, *God of war 4*, *Horizon: Zero dawn*- son un valioso referente para el presente proyecto de grado en su apartado narrativo, mostrando las diversas posibilidades e impactos que puede llegar a generar en la audiencia.

Del mismo modo que estos juegos se destacaron en gran medida por su componente narrativo, los hay otros que se destacan especialmente en su aspecto artístico, tanto visual como sonoro: *Hollow Knight* (2017) es un ejemplo de juego contemporáneo, que demuestra que el potencial de los videojuegos no se basa sólo en un apartado gráfico hiperrealista, sino más bien en su concepto artístico. Como ejemplo de ello, el crítico Tom Marks en una de sus reseñas para *IGN* (Marks T. , 2020):³ Explica que tal concepto, sin necesidad de ser realista, aporta mucho a la inmersión en la obra y la presente como un universo nuevo por explorar, como es el caso de *Journey*, *Gris*, *Limbo*, *Okami*, *Shadow of the colossus*, *Flower*, *Blasphemous* hasta del propio *Minecraft*. Todos ellos sirviendo como referentes artísticos del proyecto V (add10).

El hecho de que un videojuego se destaque por su aspecto artístico no implica que descuide su apartado narrativo, por el contrario, lo fusiona a su concepto. *Hollow Knight* evidencia eso en la creatividad y sorpresa que contiene, las cuales se ven directamente relacionadas con eventos que cambian constantemente el panorama inicial del juego, ampliándolo y presentándole al jugador elementos ocultos que sorprenden. Bien lo explica Tom Marks (2020) en su reseña "... Pero gran parte de lo que vale la pena encontrar, más allá del punto en el que puedes 'terminar' *Hollow Knight*, es lore"⁴

Los videojuegos son obras que tienen la capacidad de generar diversas emociones en los jugadores de maneras potenciadas: emplean la interactividad como un aspecto característico que los diferencian de otros medios audiovisuales como la televisión o el cine. Grandes juegos como *Earthbound* y *Mother* aprovecharon esto para potenciar el impacto de sus historias. Gracias a ello, han logrado establecerse como clásicos, influenciando centenas de juegos de la actualidad. Uno de ellos fue *Undertale*, como lo afirmó Toby Fox (su creador) en una entrevista para *Edge* y como lo ratifica Chris Schilling (2018) en su artículo para *pcgamer.com*: "Definitivamente puedo decir que busqué hacer algo que tuviera el mismo poder emocional, humor e incertidumbre que los juegos *Mother* sin tomar los mismos caminos para conseguirlo."⁵

Undertale utilizó con mucha eficacia la interacción en conjunto con su apartado artístico y su narrativa, para crear una de las experiencias audiovisuales interactivas más impactantes hasta la fecha. Se convierte en un referente importante para el proyecto V (add10), no sólo por los

development and – most importantly – memorable characters and deep, complex storylines [...] was a groundbreaking RPG that paved the way for many future titles." Traducción nuestra

³ "There are far more of these distinct biomes than I ever expected to discover, and the edges of each one blend together with the next in ways that help them make sense in the world" Traducción nuestra

⁴ "But so much of what's worth finding past the point where you can 'beat' *Hollow Knight* is pure, concentrated lore. To really get a picture of what you're doing (and to properly inform an important choice you have to make) you need to explore. I love that those key moments in the story tie into some of the best gameplay challenges it has to offer as well." Traducción nuestra

⁵ "I can definitely say that I wanted to make something that had as much emotional power, humour and wonder as the *Mother* games, while not necessarily taking the same paths to achieve it" Traducción nuestra

motivos expuestos anteriormente, sino también porque fue desarrollado casi en su totalidad por una sola persona, demostrando el gran potencial que pueden tener los juegos *Indie*, de bajo presupuesto y con un número reducido de participantes en su desarrollo.

Música generativa

La música generativa es un concepto que se le atribuye a Brian Eno. Consiste principalmente en hacer sonar una pieza musical de forma distinta cada vez que llega al oyente, tanto en vivo como en formatos pregrabados, por lo que puede generar diferentes emociones en el espectador.

La música generativa debe estar ligada a un medio de reproducción aleatorio seleccionado por el autor de la pieza con anterioridad. Un ejemplo podría ser un arpegiador MIDI en modo aleatorio. En este caso, dicho arpegiador generará notas en diferente orden cada vez que sea interpretado, lo que producirá la sensación de impredecibilidad de la pieza, creando lo que se conoce como un “tema generativo”.

Un tema generativo consta de una serie de notas que deben ser interpretadas en órdenes aleatorios cada vez que se presentan. Este tipo de música supone un problema con los medios de distribución actuales, ya que formatos como el mp3, wav, OGG, etc., hacen de la música, sus motivos y sus temas como algo preestablecido que llegará igual a los oyentes cada vez que sean reproducidos. Por lo anterior, la música generativa adquiere validez sólo cuando se interpreta en vivo.

Actualmente, con el avance tecnológico puede hallarse una alternativa potencial a la imposibilidad de reproducir música generativa a partir de formatos de audio convencionales: los motores de audio de los videojuegos, con scripts que generen aleatoriedad sobre los elementos a reproducir (como las notas de un tema musical), podrían convertir la música en algo generativo.

Música adaptativa

La música adaptativa es aquella que tiene la posibilidad de ajustar sus atributos y características en tiempo real en función de eventos o acontecimientos externos. Esto la relaciona estrechamente con la música generativa/procedural, con la diferencia de que aquí no se trata de una evolución progresiva y natural de la pieza, sino de cambios repentinos disparados por el entorno que rodea a la misma (Arjona, 2020).

En el libro *An Introduction to Procedural Music in Videogames* la investigadora Karen Collins explica la importancia del audio adaptativo y cómo los aspectos sonoros de un juego, debido a su no linealidad, se dividen en dos categorías:

- **Audio interactivo:** Se refiere a todo evento sonoro que es directamente activado por el jugador, afectado por el dispositivo que genera las órdenes de entrada (*joystick*, control, etc.) mientras el jugador interactúa con los elementos dinámicos del videojuego (enemigos, *colliders*, recolectables, combates, eventos de muerte, *respawn*, etc.) (Collins, 2009).
- **Audio adaptativo:** Son eventos que no son afectados por las acciones directas del jugador, aunque sean inevitablemente activadas por acciones indirectas del mismo. Los eventos de

audio adaptativo son puntos predefinidos en el videojuego que envían una señal al motor de audio, generando de esta forma la reproducción de sonidos. Cabe destacar que, rara vez, los sonidos son de repetición inmediata. (Collins, 2009).

3. MOTIVACIÓN Y PROPÓSITO

V (add10) surgió de la iniciativa por explorar y profundizar en el campo del diseño sonoro, sonorización mediante efectos y ambientes, y musicalización aplicada a los videojuegos, articulando conceptos de planeación y desarrollo hasta concebir el producto final.

Como futuros profesionales consideramos que es pertinente profundizar en temas que aportan a la inmersión del usuario en los videojuegos, mediante técnicas de diseño sonoro y producción musical, recurriendo a la generación de eventos adaptativos, conforme a las interacciones que ocurren en el juego en escenarios no lineales, diferentes a los propuestos en el cine y en los productos visuales con construcción narrativa lineal.

4. CONCEPTOS DE DISEÑO

Se identificaron conceptos importantes para el desarrollo del videojuego en coherencia con las temáticas de interés investigativo del proyecto, estos fueron:

Áreas de Interés
Música Adaptativa
Diseño sonoro interactivo
Generación de ambientes 3D adaptados al sistema biaural
Construcción del apoyo narrativo a partir del sonido
Diseño visual y elementos interactivos

Teniendo en cuenta cada uno de estos conceptos, se planteó un diagrama funcional consignando los elementos fundamentales del producto y su interacción directa con el usuario. Se presenta a continuación:



Imagen 1. Diagrama funcional del software

Se planteó un programa estructurado en fases de producción para el desarrollo del videojuego, contemplando cada una de las etapas de pre-producción, producción y post-producción del producto audiovisual V (add10).

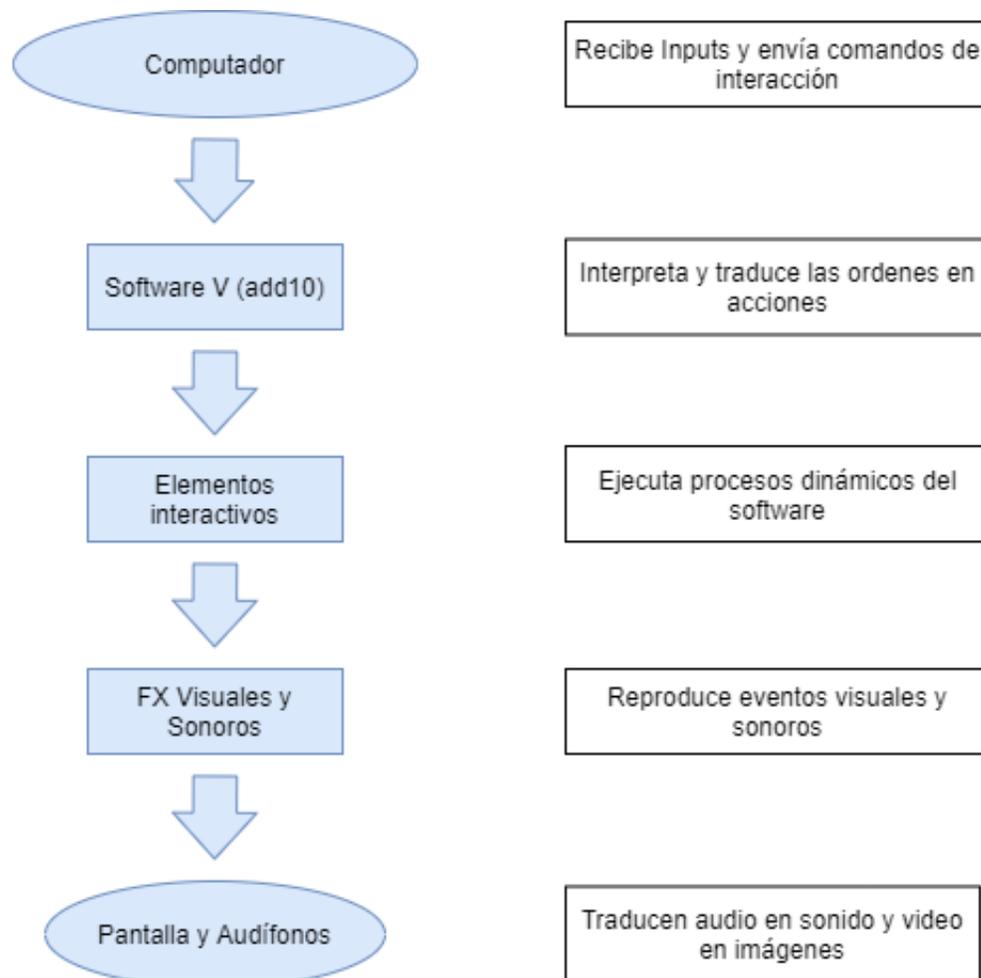


Imagen 2. Fases de producción

El proyecto presentó varias problemáticas que debían ser tratadas para su correcta ejecución:

- ¿Como implementar música interactiva en Unity?
- ¿Cómo situar al usuario final dentro del contexto narrativo a través de elementos sonoros y visuales?
- ¿Cómo interrelacionar los diferentes elementos del juego para garantizar que la jugabilidad cubra un rango amplio de edades?

Se procedió a plantear la respuesta a cada problemática mediante diferentes elementos y criterios de solución. El elemento específico en cada caso se definió a partir de un ejercicio de ponderación, donde se calificaba de la siguiente forma: valor de uno (1) al elemento que cumplía con el criterio de solución, valor de cero (0) al elemento que cumplía parcialmente con el criterio y valor de menos uno (-1) al elemento que no cumplía con dicho criterio:

Problema	Solución propuesta	Elementos	Criterios
¿Cómo implementar música interactiva en Unity?	Implementación de un motor de audio	<ul style="list-style-type: none"> • Unity-Build in • Wwise • Fmod • Fabric • ADX2 	<ul style="list-style-type: none"> • Compatibilidad con la versión 2020.1.7f1 Unity • Curva de aprendizaje • Facilidad de programación de música adaptativa • Versatilidad • Capacidad multipista • Acceso gratuito o bajo costo • Documentación de apoyo en la red

Tabla 1. Problema implementación de música interactiva

Elementos	Unity	Wwise	Fmod	Fabric	ADX2
Compatibilidad con la versión 2020.1.7f1 de Unity	1	1	1	1	1
Curva de aprendizaje	1	-1	1	-1	1
Facilidad de programación de música adaptativa	-1	0	1	0	1
Versatilidad	-1	0	1	0	1
Capacidad multipista	1	-1	1	1	1
Acceso gratuito o bajo costo	1	1	1	0	1
Documentación en la red	1	0	1	0	0
TOTAL	3	0	7	1	6

Tabla 2. Elementos de evaluación

Problema	Solución propuesta	Elementos	Criterios
¿Cómo situar al usuario final dentro del contexto narrativo planteado a través de elementos sonoros y visuales?	<ol style="list-style-type: none"> 1. Caracterización del Pacífico colombiano con elementos visuales 2. Caracterización del Pacífico colombiano con elementos sonoros 	<ol style="list-style-type: none"> 1. Visuales: <ul style="list-style-type: none"> • Assets gratuitos • Diseños propios • Diseños por encargo 2. Sonoros: <ul style="list-style-type: none"> • Instrumentos físicos • Instrumentos virtuales • Interpretes • Paisaje sonoro real • Paisaje sonoro virtual 	<ol style="list-style-type: none"> 1. Visuales: <ul style="list-style-type: none"> • Costo • Tiempo de diseño • Posibilidades narrativas • Accesibilidad 2. Sonoros: <ul style="list-style-type: none"> • Costo • Tiempo de implementación • Posibilidades narrativas • Accesibilidad

Tabla 3. Problema usuario final en contexto narrativo

Elementos Visuales	Assets gratuitos	Diseños propios	Diseños por encargo
Costo	1	1	0
Tiempo de diseño	1	-1	1
Posibilidades narrativas	-1	1	1
Accesibilidad	1	-1	0
TOTAL	2	0	2

Tabla 4. Elementos de evaluación de evaluación

Elementos Sonoros	Instrumentos físicos	Instrumentos virtuales	Intérpretes	Paisaje sonoro real	Paisaje sonoro virtual
Costo	-1	1	-1	-1	1
Tiempo de diseño	-1	1	-1	-1	1
Posibilidades narrativas	1	1	1	1	1
Accesibilidad	-1	1	-1	-1	1
TOTAL	-2	4	-2	-2	4

Tabla 5. Elementos sonoros de evaluación

Problema	Solución propuesta	Elementos	Criterios
¿Cómo interrelacionar los diferentes elementos del juego para garantizar que la jugabilidad cubra un rango amplio de edades?	Implementación de elementos interactivos intuitivos	<ul style="list-style-type: none"> Mecánicas de juego Patrones de juego Recolectables Ubicación espacial 	<ul style="list-style-type: none"> Facilidad en la curva de aprendizaje Memorización de patrones de combate de los enemigos Reconocimiento auditivo de elementos Reconocimiento visual de elementos

Tabla 6. Problema jugabilidad en amplio rango de edades de usuarios

Elementos	Mecánicas de juego	Patrones de juego	Recolectables	Ubicación espacial
Facilidad en la curva de aprendizaje	SI	SI	SI	SI
Memorización de dinámicas	SI	SI	SI	SI
Reconocimiento auditivo de elementos	SI	SI	SI	SI
Reconocimiento visual de elementos	SI	SI	SI	SI

Tabla 7. Elementos de evaluación mecánicas de juego

RESULTADO MATRICES DE SELECCIÓN			
Problema	Solución propuesta	Elemento	Elemento asociado
¿Cómo implementar música interactiva en Unity?	Utilización de motor de audio	FMOD	Computador
¿Cómo situar al usuario final dentro del contexto narrativo planteado a través de elementos sonoros y visuales?	<ol style="list-style-type: none"> Elementos característicos del pacífico colombiano con elementos visuales Elementos característicos del pacífico colombiano con elementos sonoros 	<ol style="list-style-type: none"> Assets gratuitos y Diseño por encargo Instrumentos virtuales y Paisaje sonoro virtual 	Diseñador
¿Cómo interrelacionar los diferentes elementos del juego para garantizar que la jugabilidad cubra un rango amplio de edades?	Implementación de elementos interactivos intuitivos	<ul style="list-style-type: none"> Mecánicas de juego Patrones de juego Recolectables Ubicación espacial 	<ul style="list-style-type: none"> Controles Visual Studio Menú inventario Mapa

Tabla 8. Matriz de selección

Con base en los elementos de solución y dispositivos asociados, se propuso la arquitectura del sistema, se definieron los subsistemas y se realizó el ajuste del diagrama de diseño del sistema general:

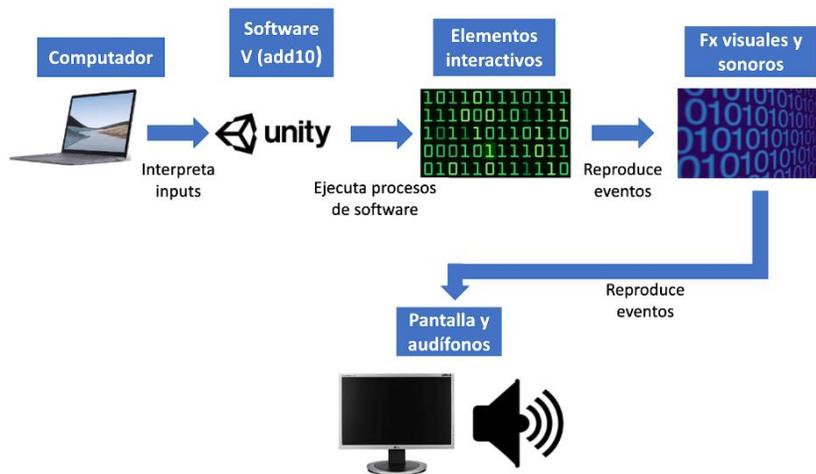


Imagen 3. Diseño de sistema general

5. DISEÑO DE DETALLE

Tomando como punto de partida el concepto de solución del sistema, se establecieron cinco (5) funciones generales para el software y las subfunciones relacionadas:

Funciones	Descripción	Sub-funciones
Detección de entrada de comandos del jugador	Reconocimiento de botones del teclado del computador asociados a funciones y acciones dentro del juego	Detección de movimientos horizontales (hacia la izquierda y hacia la derecha) por las teclas asignadas. Detección de estado de salto, ataque y lanzamiento de proyectil.
Ejecutar mecánicas de juego	Ejecución de movimientos de desplazamiento y de combate	Detección de ubicación y distancia del jugador con respecto al enemigo y ejecución de acciones de IA en consecuencia. Detección de colisión del jugador con agujeros y elementos hostiles
Activar elementos interactivos	Funciones dinámicas asociadas a objetos y eventos dentro del desarrollo del juego	Ejecución de acciones narrativas de acuerdo con el avance del juego, activación de menús e ítems interactivos, recolección de objetos de salud y poder. Ejecución de estado de combate al recibir daño o aplicar daño al enemigo
Almacenar datos de partida	Guarda el avance del jugador, el escenario donde está situado y su progreso	Almacena cantidad de vida en el punto de guardado, cantidad de proyectiles disponibles, datos del estado de los recolectables y enemigos por el escenario, posición y escena donde se encuentra el jugador
Reproducción de eventos audiovisuales	Traduce el código de programación en imágenes que aparecen en pantalla, y sonidos reproducidos por sistemas binaurales	Salida de datos de los eventos desarrollados en el juego hacia los transductores de video y audio (Pantalla y Audífonos)

Tabla 9. Funciones generales del software

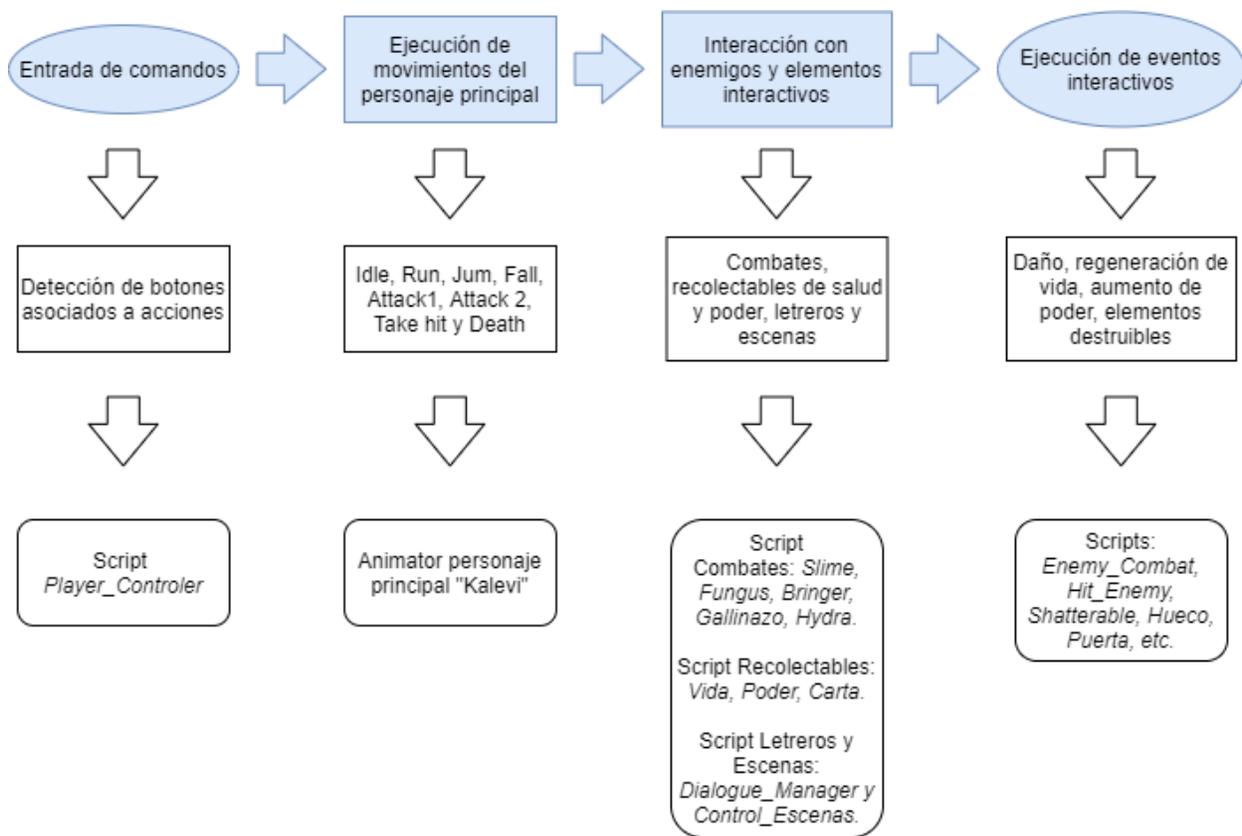


Imagen 4. Flujo de trabajo y funcionamiento del software

Los diagramas de funciones del software y la descripción de las estructuras de programación se presentan en el documento "Descripción del Programa".

A continuación, se presenta la carpeta madre que contiene los Scripts referenciados en el diagrama de la Imagen 4 y las demás subcarpetas contenedoras de Scripts necesarios para el adecuado funcionamiento del programa.

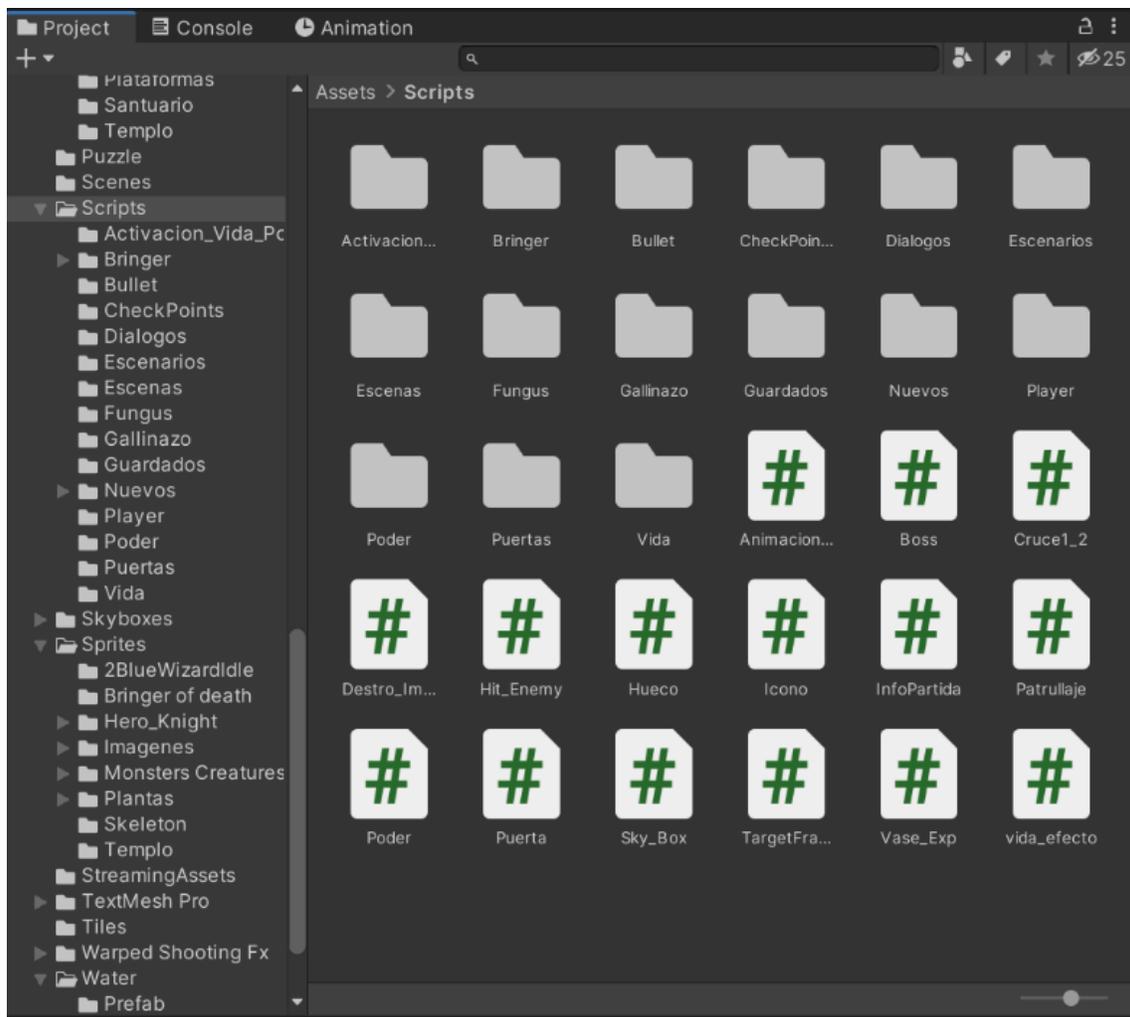


Imagen 5. Carpeta contenedora de scrips

6. MÚSICA ADAPTATIVA

La música juega un papel importante en la ambientación de V(add10), que se adapta acorde con los eventos que ocurren durante la experiencia de juego, ajustándose con el objetivo de lograr una mayor coherencia entre lo que se ve y lo que se escucha.

Las piezas musicales adaptativas que componen el juego se categorizan en: música de escenario y música de *boss*.

Música de escenario: Son las piezas musicales que se reproducen durante la mayor parte del videojuego, están presentes durante la fase de exploración del jugador a través de los escenarios de selva y el laberinto. Cada pieza posee tres (3) estados o secciones que se reproducen cuando una condición específica se cumple. Dichos estados son:

- Calma: estado que se activa mientras el jugador permanece estático o dentro del menú. Musicalmente se caracteriza por la ausencia de percusión, alto contenido de bajas frecuencias y temas o motivos melódicos.
- Exploración: es activado en el momento en que el jugador comienza a recorrer el escenario sin entrar en estado batalla. Musicalmente posee elementos que producen dinamismo en la pieza.
- Batalla: momento en el cual el jugador se enfrenta en un combate importante, como ejemplo: la “zona de combate” ubicada en la escena No. 8. Musicalmente se caracteriza por ser el momento de mayor intensidad de la pieza, con percusión dinámicas de expresión fuertes.

Música de *boss*: Está presente en las batallas contra los jefes, son temas de batalla de alta intensidad que contrastan con el resto de los temas que componen el videojuego. Estas piezas no pertenecen a la categoría “Música de escenario” razón por la cual, se asumen como estados diferentes a la “batalla”.

Las piezas de esta categoría poseen tres (3) estados:

- Fase 1: es activado desde el primer momento del encuentro.
- Fase 2: inicia en el momento en el cual el jefe llega a un 50% de vida.
- Final: La pieza llega a esta sección en el momento en el cual el jefe es derrotado por el jugador.

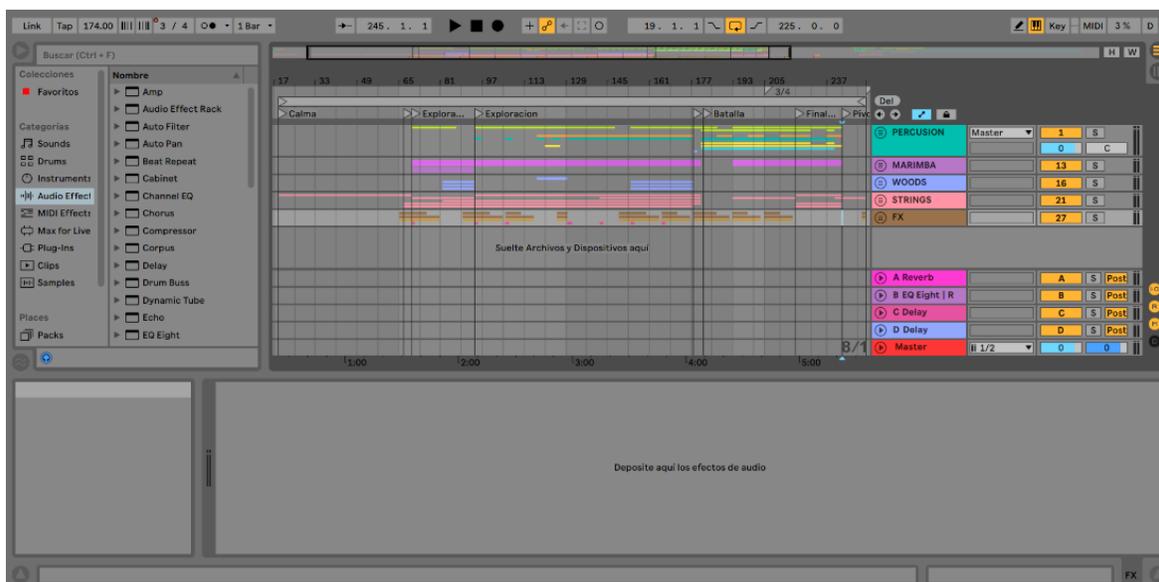


Imagen 6. Sesión de Ableton de composición de música

7. JUGADOR COMO INTÉRPRETE MUSICAL

El jugador como interprete musical

Bringer's gatekeep es una pieza musical que tiene como propósito explorar las características y posibilidades que ofrece la música Generativa/Procedural. También recoge elementos del minimalismo y música aleatoria adaptados a un contexto audiovisual interactivo a manera de *puzzle* el cual, supondrá una barrera a superar por parte del jugador. La pieza consta de una serie de tres (3) notas (Mi, Sol, Si) en conjunto con la reproducción de los efectos y ambientes sonoros de propios de la escena.

Serie de tres notas (Mi, Sol, Si)

Las notas de esta serie fueron elegidas para establecer un centro de gravedad tonal⁶ que permita al oyente identificarlo y asimilarlo. Además, en conjunto, conforman el acorde triádico de Mi menor, el cual será el eje tonal de la pieza musical a la que precede, todo con el objetivo de generar coherencia entre una pieza y otra: *Bringer's gatekeep* cumple la función de introducción para *Bringer of death*.

Todas las notas fueron generadas a partir de técnicas de síntesis aditiva y por *wavetable* con el sintetizador *Serum*, a través del siguiente procedimiento:

- Se tomó una muestra del canto de un ave típica de Colombia y se utilizó como *wavetable* para el *Serum* en uno de sus osciladores.
- Se suavizó la transición de las muestras con la función *Morph* y se delimitó aquellas que producían contenido armónico importante. De este modo suprimieron los silencios en la muestra.
- Se tomó la muestra número 196⁷ y se procesó con el módulo de síntesis aditiva de *Serum* en las etapas de saturación, chorus, reverb, delay EQ y compresión.

Efectos sonoros (Fx)

Están compuestos por todos aquellos sonidos realizados por la acción del personaje en pantalla que es controlado por el jugador. Se incluyen los pasos, saltos, aterrizajes, clothing, sonido del arma, etc. Contienen un fuerte elemento de aleatoriedad debido a lo indeterminado que pueden ser las decisiones del jugador. Resulta virtualmente imposible que dos jugadores realicen las mismas acciones al mismo tiempo y durante el mismo tiempo. Debido a esto, el *soundscape* será diferente cada vez que se ejecute la obra.

Ambientes sonoros (Ax)

Son elementos que están en el *background*, ambientando la escena como los *roomtone*, el sonido que producen las antorchas encendidas, viento, etc.

⁶ Centro de gravedad tonal: Corresponde a la base sobre la que se desarrolla una escala y que, a su vez, es el punto de máxima relajación de la sonoridad.

⁷ Es un fragmento del sonido que compone el *Wavetable*

Los ambientes sonoros, a diferencia de los efectos sonoros, serán menos lineales y menos indeterminados ya que siempre empezarán desde el mismo punto. Sin embargo, su duración dependerá enteramente del jugador y de cuánto tiempo se demore en cambiar de escena.

Bringer's gatekeep representa un *puzzle* sonoro que deberá ser resuelto por el jugador para poder avanzar a la siguiente escena. Consiste en un juego de memoria auditiva y comparación de notas, donde la respuesta correcta estará entre dos notas iguales a cada extremo de la escena.

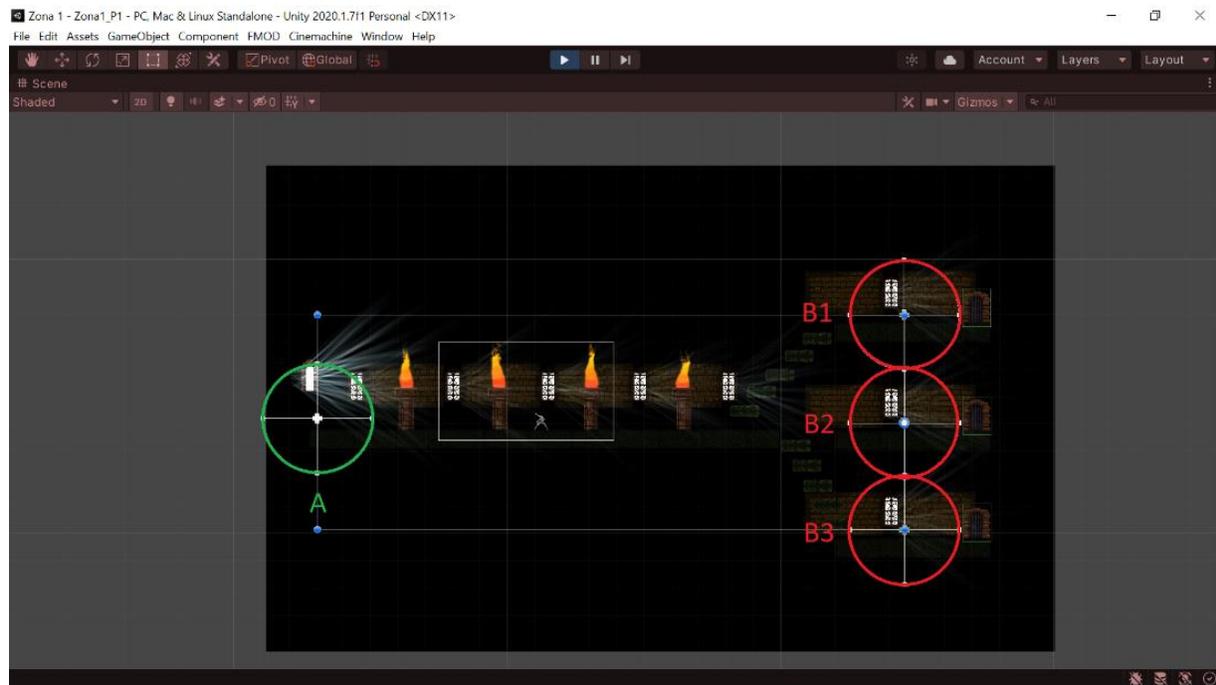


Imagen 7. Diagrama de funcionamiento del puzzle auditivo

- A: Representa el punto inicial del jugador, donde escuchará una nota de la serie que representa la nota correcta (definida aleatoriamente por el Script), que deberá ser memorizada y luego reconocida al compararla con las notas de los puntos B1, B2 y B3.
- B: Representa los puntos a los cuales se dirigirá el jugador. Habrá una nota de la serie asignada a cada punto B de manera aleatoria y sin repetir, por lo que sólo una de ellas corresponderá a la nota del punto A.

Una vez el jugador haya resuelto tres secuencias de manera exitosa, avanzará hacia un escenario tipo *Boss*. En caso de que el jugador falle en alguna de las tres secuencias del *puzzle*, será devuelto a la secuencia inicial hasta que logre descifrar la respuesta.

La aleatoriedad de la serie de tres notas es posible gracias al Script asignado en Unity, funcionando con una lógica basada en condicionales, los cuales, de llegarse a cumplir, ejecutan las acciones que tienen programadas.

The screenshot shows the Visual Studio IDE with the 'EnsayoP1.cs' script open. The script is a Unity MonoBehaviour class with the following code:

```
1 using System.Collections;
2 using System.Collections.Generic;
3 using UnityEngine;
4
5 [Unity Script (3 asset references)] 0 references
6 public class EnsayoP1 : MonoBehaviour
7 {
8     public Transform P1;
9     public Transform P2;
10    public Transform P3;
11    public Transform PuertaPuzzle;
12    GameObject Correcto;
13    GameObject CucoIncorrecto1;
14    GameObject CucoIncorrecto2;
15    GameObject Entrada;
16    public GameObject Entrada1;
17    public GameObject Entrada2;
18    public GameObject Entrada3;
19    public GameObject Correcto1;
20    public GameObject Correcto2;
21    public GameObject Correcto3;
22    public GameObject Incorrecto1;
23    public GameObject Incorrecto2;
24    public GameObject Incorrecto3;
25    int RandomNumber;
26    int RandomCorrecto;
27    int RandomIncorrecto;
28
29    // Start is called before the first frame update
30    void Start()
31    {
32        Incorrecto1.SetActive(true);
33    }
34 }
```

Imagen 8. Scrip del puzzle auditivo 1

The screenshot shows the Visual Studio IDE with the 'Ubicacion_Correcto.cs' script open. The script is a Unity MonoBehaviour class with the following code:

```
29 void Start()
30 {
31     Incorrecto1.SetActive(true);
32     Incorrecto2.SetActive(true);
33     Incorrecto3.SetActive(true);
34     RandomNumber = Random.Range(1, 4);
35     RandomCorrecto = Random.Range(1, 4);
36     RandomIncorrecto = Random.Range(1, 3);
37     Puerta_Correcta();
38     Ubicacion_Correcto();
39 }
40 public void Ubicacion_Correcto()
41 {
42     if (RandomNumber == 1)
43     {
44         Instantiate(Correcto, P1.position, P1.rotation);
45         Instantiate(Entrada, PuertaPuzzle.position, PuertaPuzzle.rotation);
46         if (RandomIncorrecto == 1)
47         {
48             Instantiate(CucoIncorrecto1, P2.position, P2.rotation);
49             Instantiate(CucoIncorrecto2, P3.position, P3.rotation);
50         }
51         else if (RandomIncorrecto == 2)
52         {
53             Instantiate(CucoIncorrecto1, P3.position, P3.rotation);
54             Instantiate(CucoIncorrecto2, P2.position, P2.rotation);
55         }
56     }
57     else if (RandomNumber == 2)
58     {
59         Instantiate(Correcto, P2.position, P2.rotation);
60     }
61 }
```

Imagen 9. Scrip del puzzle auditivo 2

```
56     }
57     else if (RandomNumber == 2)
58     {
59         Instantiate(Correcto, P2.position, P2.rotation);
60         Instantiate(Entrada, PuertaPuzzle.position, PuertaPuzzle.rotation);
61         if (RandomIncorrecto == 1)
62         {
63             Instantiate(CucoIncorrecto1, P1.position, P1.rotation);
64             Instantiate(CucoIncorrecto2, P3.position, P3.rotation);
65         }
66         else if (RandomIncorrecto == 2)
67         {
68             Instantiate(CucoIncorrecto1, P3.position, P3.rotation);
69             Instantiate(CucoIncorrecto2, P1.position, P1.rotation);
70         }
71     }
72     else if (RandomNumber == 3)
73     {
74         Instantiate(Correcto, P3.position, P3.rotation);
75         Instantiate(Entrada, PuertaPuzzle.position, PuertaPuzzle.rotation);
76         if (RandomIncorrecto == 1)
77         {
78             Instantiate(CucoIncorrecto1, P1.position, P1.rotation);
79             Instantiate(CucoIncorrecto2, P2.position, P2.rotation);
80         }
81         else if (RandomIncorrecto == 2)
82         {
83             Instantiate(CucoIncorrecto1, P2.position, P2.rotation);
84             Instantiate(CucoIncorrecto2, P1.position, P1.rotation);
85         }
86     }
87 }
```

Imagen 10. Scrip del puzzle auditivo 3

```
86     }
87     }
88     1 reference
89     public void Puerta_Correcta()
90     {
91         if (RandomCorrecto == 1)
92         {
93             Correcto = Correcto1;
94             Entrada = Entrada1;
95             CucoIncorrecto1 = Incorrecto2;
96             CucoIncorrecto2 = Incorrecto3;
97         }
98         else if (RandomCorrecto == 2)
99         {
100            Correcto = Correcto2;
101            Entrada = Entrada2;
102            CucoIncorrecto1 = Incorrecto1;
103            CucoIncorrecto2 = Incorrecto3;
104        }
105        else if (RandomCorrecto == 3)
106        {
107            Correcto = Correcto3;
108            Entrada = Entrada3;
109            CucoIncorrecto1 = Incorrecto1;
110            CucoIncorrecto2 = Incorrecto2;
111        }
112    }
113 }
```

Imagen 11. Scrip del puzzle auditivo 4



Imagen 12. Sesión FMOD del puzzle auditivo

En la Imagen 12, se visualiza la interfaz del motor de audio FMOD donde se programó el evento de sonido 3D que da vida a la obra *Bringer's gatekeep*. Este evento reacciona conforme a la distancia entre el jugador y los puntos que accionan los sonidos de la obra dentro del escenario de juego. Lo anterior permite al jugador identificar espacialmente, a través de la ecolocación, la ubicación de los elementos sonoros que componen el *puzzle*.

En el Score de la pieza se encuentra una transcripción de dos rondas aleatorias del tema generativo, que serán elegidas por el Script.

score

bringer's gatekeeper

Alejandro De la Espriella Mazo
Dany Stevan Rios Madrid

2/10/21

1 Andante ♩ = 90

Synth Pad

Ambience

8

Pad

Ax

16

Pad

2

Ax

© Alejandro De la Espriella Mazo, Dany Stevan Rios Madrid

2 de 2

bringer's gatekeeper

24

Pad

Ax

32

Pad

Ax

Imagen 13. Partitura puzzle auditivo

8. DISEÑO SONORO

V (add10) propone un diseño sonoro inmersivo que plantea eventos auditivos dinámicos y aleatorios que enriquecen la experiencia de juego, ofreciendo al jugador un ambiente exento de patrones repetitivos que puedan producir monotonía.

El diseño sonoro de V (add10) permite al jugador sumergirse en distintos ambientes de tipo selvático, de recintos sagrados y de ambientes al aire libre, que refuerzan los elementos visuales de cada escenario y dan vida a la construcción narrativa de la historia. Además, la construcción del diseño sonoro tiene un fuerte componente dinámico, el cual, se encuentra dividido en dos partes:

- **Diseño sonoro interactivo:** son todos aquellos sonidos que se activan por la participación del personaje principal al interactuar con objetos, elementos del escenario, enemigos u otros elementos dinámicos. Estos sonidos poseen una característica aleatoria que evita la reproducción consecutiva de sonidos iguales, teniendo como elementos distintivos, cambios en ganancia, afinación, timbre y modificaciones en la envolvente acústica del sonido.
- **Diseño sonoro aleatorio:** consiste en todos aquellos sonidos que interactúan de forma constante con el escenario y el personaje principal, como lo son: animales, elementos naturales (viento, fuego, agua, tierra), sonidos que evocan elementos mágicos, entre otros. Estos sonidos crean la sensación de un escenario que evoluciona acústicamente través del tiempo: debido a que el diseño sonoro se concibió bajo los principios de la música adaptativa, se activan elementos aleatorios conforme avanza el juego y de manera coherente con las acciones del personaje principal. Los eventos sonoros aleatorios se encuentran divididos en sonidos 2D y 3D.
 - Sonidos 2D: son sonidos que no dependen de una ubicación espacial específica, sino que dependen únicamente de la interacción con el personaje principal y el contexto visual planteado.
 - Sonidos 3D: son sonidos cuya función principal es permitir al usuario identificar la posición de la fuente sonora en el escenario (ubicación de la fuente que produce el sonido). Esto se logra generando diferencias de nivel y tiempo entre las señales que captará el espectador por sus oídos izquierdo y derecho.

Proceso de creación a partir del diseño sonoro

Para la creación a partir del diseño se tuvo en cuenta las siguientes categorías de sonidos: Efectos (Fx), Ambientes (Ax) y Música (Mx). En V (add10), los diálogos no se reproducen como sonidos, sino que se proyectan como texto.

Luego de la construcción del diseño sonoro, se procede a implementar los distintos eventos 2D y 3D en el motor de audio FMOD, el cual funciona como plataforma de integración con Unity, permitiendo vincular el sonido y la imagen interactiva.

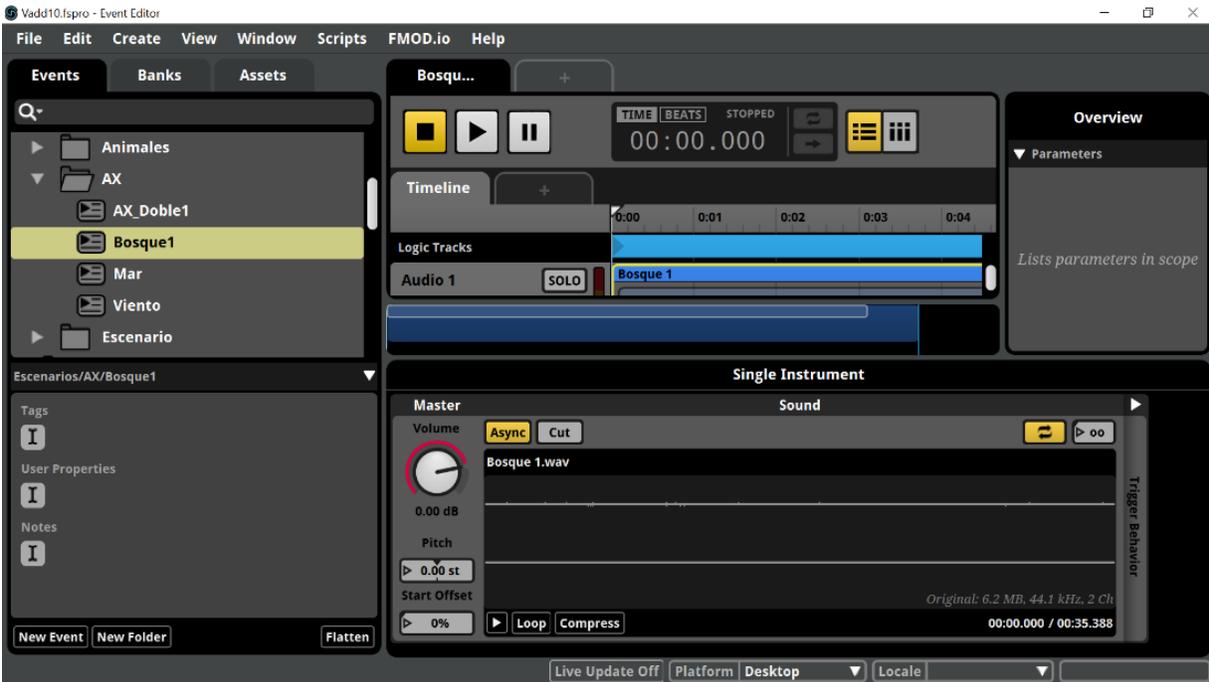


Imagen 16. Sesión de FMOD para diseño sonoro interactivo

9. GLOSARIO

- Background: Son efectos que no se encuentran explícitamente sincronizados con la imagen. Su ubicación espacial y sus planos no sobresalen sobre los sonidos principales.
- DAW (Digital Audio Workstation): Sistema integrado de hardware y software que permite al usuario grabar, editar y mezclar audio.
- FMOD: Motor de audio que permite crear eventos sonoros que se vinculan directamente con plataformas de desarrollo interactivo como Unity y Unreal.
- Lore: Hace referencia a todos los elementos que conforman el universo al que representan, esto involucra los datos, personajes e historia que dan coherencia al videojuego.
- Unity: Plataforma de desarrollo interactivo y visual para productos como videojuegos y software dinámico.
- Puzzle: Acertijo o prueba a la que se somete el jugador como parte de la propuesta interactiva del juego.

- Recolectables: Objetos utilitarios para el jugador, que pueden ser recolectados para uso inmediato o posterior.
- Roomtone: Sonido que representa la base tímbrica característica de un espacio acústico.
- Serum: Sintetizador de la compañía Xfer records. Que permite síntesis por Wavetable, Aditiva, Sustractiva y FM.
- Síntesis por Wavetable: Es una forma de síntesis basada en sampleo que usa un sonido pregrabado como la fuente sonora primaria para cada oscilador.
- Síntesis Aditiva: Es la creación de sonidos complejos por medio de la adición de componentes sencillos.
- Soundscape: Resultado final obtenido al juntar todos los elementos sonoros en el contexto de un videojuego (ambientes, efectos, diálogos y música).

10. REFERENCIAS

- Acosta, A. M. (2016). *Plan de negocio para la creación de una empresa desarrolladora de software*. Bogotá: Universidad EAN.
- Casallas, J. F., & Riveros, Ó. C. (2019). *Creación de videojuegos en la enseñanza del uso placas de desarrollo de hardware para ingenieros de sistemas*. Ibagué: Universidad Cooperativa de Colombia.
- Collins, K. (2008). *Game Sound: An Introduction to the History, Theory, and Practice of Video Game Music and Sound Design*. London: The mit press.
- D'Aoust, K. (2014). *Unity Game Development Scripting*. Birmingham: Packt publishing.
- Delmas, G., Champagnat, R., & Augeraud, M. (2020). *From Tabletop RPG to Interactive Storytelling: Definition of a Story Manager for Videogames*. Montreal.
- Egea Canales, J. (2020). *Desarrollo De Un Videojuego Para Móviles Con Unity O Cocos2d-X*. Alicante: Escuela politécnica superior.
- Gáfaró, R. E., Contreras, M. R., & Villamizar, L. A. (2017). *Gestión de alcance en proyectos de desarrollo de videojuegos*. Universidad de Pamplona.
- González, J., & Obando, O. L. (2011). *Clasificar los videojuegos como tarea dinámica*.
- González, R. M. (2015). *Doblaje y videojuegos: La incidencia de la industria del cine en un nuevo sector de ocio*. San Vicente del Raspeig: Universitat d'Alacant.
- Green, A. M. (2018). *Storytelling in Video Games: The Art of the Digital Narrative*. Matthew Wilhelm Kapell.

- Horowitz, S., & Looney, S. R. (2014). *The Essential Guide to Game Audio: The Theory and Practice of Sound for Games*. New York: Routledge.
- Ibar, E. B. (2018). *El sonido como herramienta narrativa en los videojuegos*. Cataluña: Universitat Politècnica de Catalunya.
- Iwata, S., Kondo, K., Yokota, M., & Osawa, T. (17 de 06 de 2011). Iwata pregunta: The legend of Zelda: Ocarina of Time 3D. (S. Iwata, Entrevistador)
- Lebowitz, J., & Klug, C. (2011). *Interactive storytelling for video games: A player-centered approach to creating memorable characters and stories*. Taylor & Francis.
- Marks, A. (2009). *The complete guide to game audio: for composers, musicians, sound designers, and game developers*. Elsevier.
- Marks, T. (20 de 04 de 2020). *ing.com*. Obtenido de *ing.com*:
<https://www.ign.com/articles/2018/06/22/hollow-knight-review>
- Mukherjee, S. (2015). *Video games and storytelling: Reading games and playing books*. London: Palgrave Macmillan.
- Pastor, E. M., & Pazos, A. J. (2020). *Los videojuegos como contenido de la telefonía*. Murcia: Universidad Católica San Antonio.
- Perron, B. (2009). *Horror Video Games: Essays on the Fusion of Fear and Play*. McFarland & Co.
- Planells de la Maza, A. J. (2010). *La evolución narrativa en los videojuegos de aventuras (1975-1998)*.
- Porrás Orozco, J. (2020). *Diseño sonoro para videojuegos*. Bogotá.
- Sánchez, J. L. (2010). *Jugabilidad. Caracterización de la experiencia del jugador en videojuegos*. Madrid: Universidad de Granada.
- Schilling, C. (05 de 05 de 2018). *pcgamer.com*. Obtenido de *pcgamer.com*:
<https://www.pcgamer.com/the-making-of-undertale/>
- Ulaş, E. S. (2014). *Virtual environment design and storytelling in video games*. Intellect.
- Villagrà Arnedo, C. J., Gallego Durán, F. J., Molina Carmona, R., & Llorens Largo, F. (2014). *ABPgame+: siete asignaturas, un proyecto*. Alicante: Universidad de Alicante.
- Vives, J. Á. (2018). *Creando universos: proceso para la realización de Concept Art para videojuegos del tipo plataformer en el Perú*. Lima.
- Zamora, V., & Darío, R. (2016). *Los videojuegos, nuevas narrativas del siglo XXI*. Madrid.
- Zehnder, S. M. (2006). *The Role of Music in Video Games*. París.