

TÍTULO

Diseño e implementación de un entorno computacional para la ayuda en la síntesis de sonidos musicales

RESUMEN

Con el fin de facilitar la producción de audio con enfoques al público general, de maneja semejante al impacto de herramientas computacionales, se postula la creación de modelos reproducibles y económicos en procesamiento como almacenamiento en memoria sobre un nuevo formato para el desarrollo, en un ambiente computacional integral sencillo de entender y manejar. De esta manera, se espera ofrecer una nueva alternativa a la composición musical que exija menos recursos externos y computaciones en sus formas de almacenar y replicar.

OBJETIVOS

- Construcción de una interfaz de usuario para la ayuda visual y auditiva en la composición y síntesis de sonidos.
- Implementación del los modelos generados en otros ambientes de software.

DEFINICIÓN DEL PROBLEMA

La interpretación de información en nuestros medios digitales, nos ha permitido producir y almacenar de forma eficiente y a un nivel mínimo de personal, siendo un ejemplo, solo se requiere un procesador de texto instalado en un equipo para poder construir artículos, publicaciones y gran diversidad de documentos formales, informales, con propósitos de divulgación científica ó de entretenimiento.

Todo lo anterior mencionado, gracias a los productos de software comercial disponibles, tanto en la producción de su contenido objetivo, como sus respectivos entornos de trabajo integrales para su construcción en un proceso más individual ó en un ámbito industrial, como su manutención.

Enlazando ambos conceptos, propongo aplicarlos a un ámbito artístico muy computable gracias a su amplio análisis matemático-lógico. Hablo de la música, disciplina artística de la armonía del sonido, su composición e interpretación, creada a partir de la refinación en las ondas sonoras a gusto y agrado del contexto social.

Perfectamente replanteadas en modelos trigonométricos, así como la captura de su información con el uso de archivos especializados en su fidelidad, compresión y réplica. Visto de otra manera, la información contenida en un archivo de audio, da las instrucciones y valores de comportamiento para un dispositivo de salida sonoro.

Sin importar el origen de esta información, por una captura directa, una copia de otro archivo, mezclado y renderizado por un programa de edición o incluso de la muy improbable casualidad. Solo se requiere de un método adecuado de producción.

Si el desafío reside en la fuente de origen, entonces se propone *La construcción de una interfaz de usuario para la ayuda visual y auditiva en la composición y síntesis de sonidos*. De acuerdo con el teorema de Fourier, referente a la descomposición en señales de onda para cualquier señal periódica. Será la fundamental relación de nuestra propuesta. Pues con la ayuda visual y auditiva en comparar por el oído y vista un modelo generado con respecto a una muestra recolectada, la réplica fidedigna a criterio de la escucha del público objetivo de producto final.

Esta idea tiene su génesis en el proyecto de sintetizador creado por el usuario *Javidx9/OneLoneCoder*, una plantilla hecha en lenguaje C++ que se puede encontrar en el siguiente repositorio: <https://github.com/OneLoneCoder/synth.git> En este se expresan mediante objetos virtuales contruidos sobre código duro(Hardcode) funciones trigonométricas de oscilador hechas para la réplica de notas musicales y como en variaciones de amplitud y frecuencia pueden emular instrumentos simplificados.

Así se espera, construir una forma de replicar el audio en tiempo de ejecución, pudiendo aplicar un nuevo panorama de beneficios y propuestas de solución a los ámbitos para la producción de contenido, especialmente aquellas interactivas.

MÉTODO

La siguiente imagen muestra una representación simplificada sobre la propuesta de solución al completo, donde se tiene planeadas todas sus funcionalidades y operaciones al menos de cara a un lanzamiento hacia el público interesado, más no uno comercial.

Cabe aclarar, este proyecto comprende más objetivos y características para resolver las problemáticas anteriormente planteadas. Aunque por los alcances limitados en tiempo, recursos y personal, se ha decidido centrar los esfuerzos en la parte más significativa en la solución de software, siendo el módulo *Tuner*.

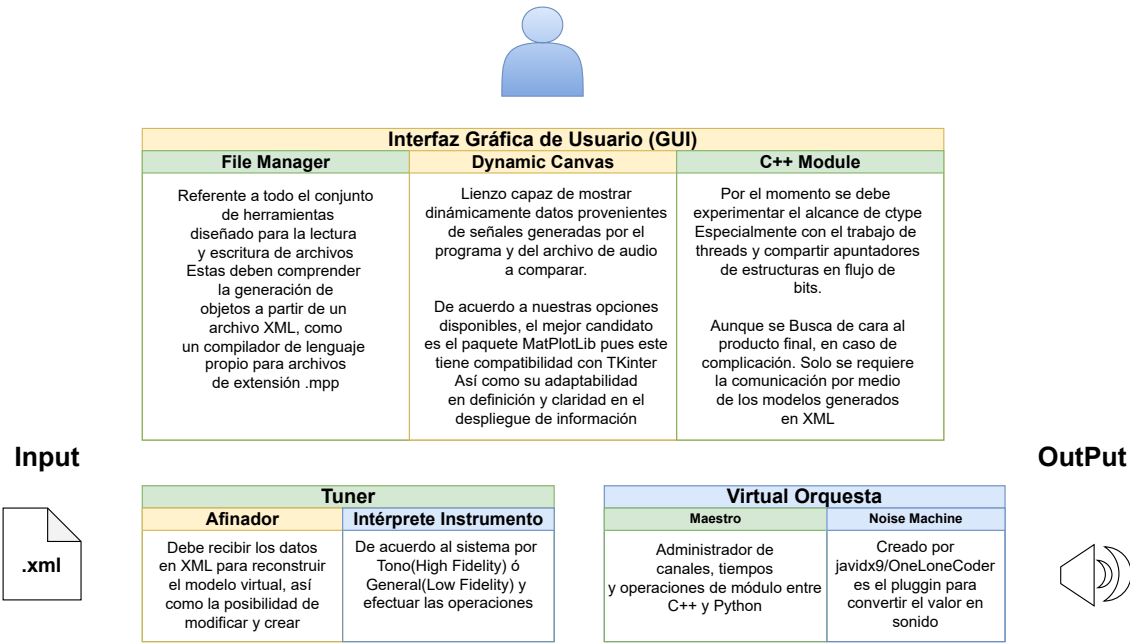


Figura 1: Diagrama General del Proyecto

Profundizando en los módulos mostrados anteriormente:

Interfaz Gráfica de Usuario (GUI)

El enfoque comercial del proyecto, exige que este contenga por lo menos una visualización cercana a los estándares en herramientas de software, si bien, el público general cambia su tendencia en cuanto conocimientos informáticos básicos, sería un error cerrarnos de cara a una presentación como la manipulación para una terminal de consola. Se ha experimentado construir una GUI con el lenguaje *C++*, sin embargo no pudo encontrarse una biblioteca adecuada, mucho menos un resultado satisfactorio para delegar todo el proyecto a un único lenguaje de programación, por lo tanto, se trabajará con Python en el *FrontEnd*, es decir, todo contacto con el usuario final.

File Manager

Ya que nuestro propósito es la generación de archivos cuya información pueda interpretarse por un estándar orientado a audio, así como la preservación del trabajo en un lenguaje de programación, es imperativo usar varios sistemas de lectura y escritura de archivos en ambos lenguajes a utilizar, siendo *Python* y *C++*, estos contienen una biblioteca en sus paquetes fundamentales, siendo la función *open* y *fstream* respectivamente. Por otra parte, se necesitará de una biblioteca especializada en convertir la información de un texto etiquetado como lo es el formato XML. No solo es un estándar que permite escalar las funciones del proyecto a futuro con otros productos de software, sino que al ser un

formato simple y en cierto punto, indicativo para ser editado manualmente.

Dynamic Canvas

Si bien, esta parte podría únicamente ser útil en lo que respecta la construcción del módulo afinador. Puede utilizarse en tiempo real la representación gráfica de la información generada, desde los armónicos que participan en la mezcla del sonido, una visión más amigable de cara al usuario de los eventos programados, entre otras cosas. Ya que esto está ligado a la GUI así como el afinador, su desarrollo sería exclusivamente en Python, optando por la opción que ofrece el paquete de *Matplotlib* por su compatibilidad con *TKinter*.

C/C++ Module

Debido a las características por implementar, es conveniente dejar a Python como aquel que tenga el ejecutable inicial, así como ser el eje de las herramientas que conlleve el proyecto. Este ofrece una opción denominada como *ctype* el cual permite convocar código en C/C++ colocando y devolviendo datos primitivos, aún no se ha experimentado del todo, pues se requiere de la certeza y técnicas en cuanto el envío de apuntadores para arreglos de datos, estructuras específicas así como su correcto funcionamiento con múltiples hilos de ejecución.

Tuner

Este módulo deberá ser construido en casi su totalidad en lenguaje *Python* pues al comparar en tiempo real la generación de señales, este deberá sumergirse por completo en el ambiente, aunque algunas partes podrían invocarse desde *C/C++* si la eficiencia es significativa. Por otra parte, debe cumplir al menos con la reproducción del modelo concurrente. Así como la lectura de información auditiva escrita en un archivo *.wav* para su comparación ante el modelo generado.

Generador de Señales

Siendo la parte más visual del proyecto, esta tiene que generar señales con los datos otorgados por el usuario, con representación gráfica en tiempo contra amplitud, frecuencia contra amplitud, frecuencia contra tiempo, en cualquiera pueda ser la información necesaria para la correcta implementación de valores en los modelos matemáticos. Así como la designación de al menos un método de aproximación en comparación a las muestras de audio.

Virtual Instrument

Siendo una convención escrita sobre las reglas del lenguaje de etiquetado XML, es una designación simple y comprensible para la re-construcción de los valores del audio. Consistiendo en 2 secciones, siendo el método, una colección de modelos trigonométricos y una plantilla de señal, siendo una representación de convertir la señal digital absoluta en una transaccional analógica.

Virtual Orquesta

Este es un módulo enfocado al procesamiento central de la información, pues estará diseñado para invocar la interpretación de las instrucciones y ajustes dados por el usuario, administración de los múltiples hilos dedicados, pausa, reinicio, incluso a la captura en tiempo real de notas generadas por el usuario mediante una entrada estándar así como la posibilidad de escalar a un instrumento especializado.

MasterChord

Siendo esta parte administrativa, deberá cumplir con una facilidad de funciones que puedan ser citadas como servicios desde la GUI así como la respuesta de información siendo instrumentos, posición de notas, tiempo real o cualquier información correspondiente a una visualización con propósitos artísticos o técnicos. Si bien las funciones gráficas son prescindibles, se deja abierta la posibilidad de escalar al producto final.

NoiseMachine

Esta es una cabecera hecha por *Javidx9/OneLoneCoder* que nos permite la comunicación con el hardware hecho para la reproducción de audio mediante la solicitud del tiempo en valor de la amplitud otorgado por una función asignada. Esta pieza limita el proyecto a plataformas con sistema operativo Windows 7 en adelante, con compatibilidad para arquitectura de x32 bits.

INVENTARIO DE ASIGNATURAS INVOLUCRADAS

Correspondientes al plan de estudios de Ingeniería en Computación(2016)

- Álgebra/Álgebra Lineal
- Fundamentos de Física/Programación
- Estructuras de datos y algoritmos I/II/Discretas
- Cálculo y Geometría Analítica/Integral
- Ecuaciones Diferenciales/ Matemáticas Avanzadas
- Señales y Sistemas/ Sistemas de Comunicaciones

ÍNDICE DESGLOSADO

1. Introducción

- 1) Objetivo
- 2) Planteamiento del problema
- 3) Estado del Arte
- 4) Fundamentos de la organización Musical

2. Análisis y Procesamiento de Audio

- 1) Definición Técnica del Audio
- 2) Captura de Información
- 3) Procesamiento de Señal
- 4) Métodos de aproximación

3. Construcción de la NoiseMachine

- 1) Estudio del proyecto olcSynth
- 2) Aporte de propuesta
- 3) Cambios y pruebas del software
- 4) Implementación y Proyección a Futuro

4. Construcción del Tunner

- 1) Estrategia y Diseño
- 2) Manejo de excepciones
- 3) Capacidades mínimas
- 4) Áreas de Innovación

5. Diseño de los Instrumentos Virtuales

- 1) Resumen de XML
- 2) Propuesta de Convención
- 3) Proyecciones con musiC++
- 4) Comparación contra formatos estandarizados

6. Guía de Usuario

- 1) Paso 1: Recolección de Muestra
- 2) Paso 2: Afinar el Instrumento Virtual
- 3) Paso 3: Prueba de Sonido
- 4) Paso 4: Exportación a otros Productos

7. Conclusiones

8. Bibliografía

9. Anexos

RESULTADOS ESPERADOS

Con el entorno de producción finalizado, se espera obtener en referencia al producto, como en propuesta de uso:

- Nuevo Formato de Almacenamiento de información orientado al arte musical
- Modelos Instrumentales digitales utilizables en otros proyectos de programación
- Facilidad en su comprensión para el usuario en su fabricación de Instrumentos virtuales
- Simulación de instrumental precisa y universal
- Diseño escalable al número de funciones y modos de interpretación del sonido producido
- Independencia de periféricos instrumentales reales

Cuando se ha mencionado el proyecto del génesis, podemos observar un área de oportunidad mucho más grande, pues si retornamos y vemos la versión *main4.cpp* integra la clase secuenciador, una muy simplificada automatización en añadir notas al proceso. Esto puede significar que una completa melodía podría implementarse de forma concreta bajo este sistema, adicionalmente a este proyecto, se podían componer canciones o replicar las existentes con los instrumentos virtuales de este proyecto, en este conjunto de soluciones propuestas.

Citando un ejemplo más específico, en la industria del entretenimiento digital como lo son los videojuegos, entornos virtuales variables sujetos a la interacción usuario computadora, donde gracias a esta propuesta se podría brindar una experiencia sonora artística mucho más apta y factible. Un alto punto de valor agregado que puede significar muchas más ganancias, prestigio y un nuevo paradigma.

CRONOGRAMA

Actividades	Semanas															
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Revisión Bibliográfica																
Recolección de Información																
Construcción de la GUI																
Implementación de lienzo y trigonometría																
Definir estándar e implementar XML																
Capturar y Graficar Audio																
Gráfica y comparación de métodos generados																
Pruebas de sonido individual																
Facilidades de producción del modelo																
Abstracción de la <code>olcNoiseMaker</code>																
Readaptación de la <code>NoiseMachine</code>																
Pruebas de sonido integral																
Reporte y redacción de avances																
Corrección final																

Figura 2: Planeación estimada del desarrollo