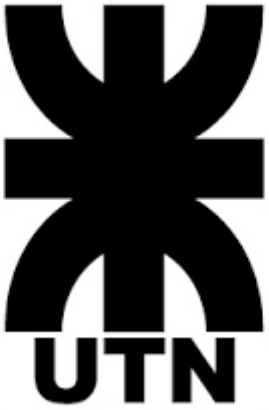


Cátedra: Soporte a la Gestión de Datos con Programación Visual
Profesor a cargo: Luque Ernesto, Castagnino Mario
Com:404
Grupo: 15

Trabajo Práctico Final

Extracción de melodía de piezas musicales polifónicas (o monofónicas) y generación de salida sintetizada

Integrantes:

- Botello, Andrés Legajo: 43697 e-mail: andresjbotello@gmail.com
- Corsetti, Ornela Legajo: 44034 e-mail: orcorsetti@gmail.com

Tabla de contenido

Narrativa – Descripción general

Dentro de la industria de la música nos encontramos con múltiples softwares de afinación de instrumentos que nos permiten reconocer la nota que está sonando en un instrumento particular, siempre y cuando esta sea monofónica (no funciona para un conjunto de notas). La idea general es imitar estos softwares. Y no solo imitar, sino que **mejorarlo** logrando la compatibilidad con sonidos polifónicos. Para ello debemos tomar cada parte de un archivo de sonido subdividido y hacer un reconocimiento general de las notas que se están tocando en ese instante en dicho archivo de audio grabado.

El programa permitirá grabar y posteriormente reproducir piezas musicales, las cuales más tarde guardará en una base de datos, y posteriormente permitirá la opción de realizar un análisis para el reconocimiento de las notas musicales presentes en cada instante de tiempo, en otras palabras, determina su **melodía**. Estos datos son presentados mediante una gráfica llamada chromagrama en la que en el eje de las abscisas se encuentra el tiempo, y en el de las ordenadas la clase tonal (do, re, mi, fa, sol, la, si o en inglés C, D, E, F, G, A, B).

El proceso de extracción de melodía se explicará en la siguiente sección.

Requerimientos

1. **Grabar audio:** Permite al usuario grabar la porción de audio que desea analizar, junto con la posibilidad de regrabar, se grabarán en formato WAV.
2. **Reproducir audio grabado:** Se debe permitir la reproducción de los audios en cuestión.
3. **Seleccionar audios grabado previamente:** Se habilita la selección de audios anteriores.
4. **Permitir mostrar gráficos:** Para los chromagramas.
5. **Subdividir archivos .WAV:** Para su posterior análisis.
6. **Consultas sobre audios grabados:** se debe permitir consultar, modificar y eliminar audios grabados.
7. **Reconocer frecuencia predominante:** Para el posterior análisis de notas.
8. **Permitir algún tipo de notación musical:** Para mostrar las notas de cada archivo (C, D, E, F, G, A, B).
9. **Archivos con frecuencias por notas:** Serán temporales, es decir, se usarán para llevar a cabo el análisis y luego se eliminarán.
10. **Melodía sintetizada:** con los resultados obtenidos en los análisis se deberá poder generar una melodía sintetizada, con el objetivo de mostrar de forma más clara los resultados.

Marco Teórico

La extracción de melodía es el proceso de estimar automáticamente la frecuencia fundamental correspondiente al tono de la pieza melódica predominante. Esta pieza puede ser polifónica o monofónica. Este proceso se ha nombrado de diversas maneras, por ejemplo:

- Extracción de melodía predominante.
- Estimación de melodía predominante.
- Estimación de la frecuencia fundamental predominante (F0).

Debemos tener en cuenta que la extracción de la melodía es distinta de la separación de fuentes de sonido, por ejemplo, la extracción de melodía no separará la primera voz de una grabación (aunque podría ayudar a lograr esto en una etapa más avanzada).

En resumen, el proceso presentado implica:

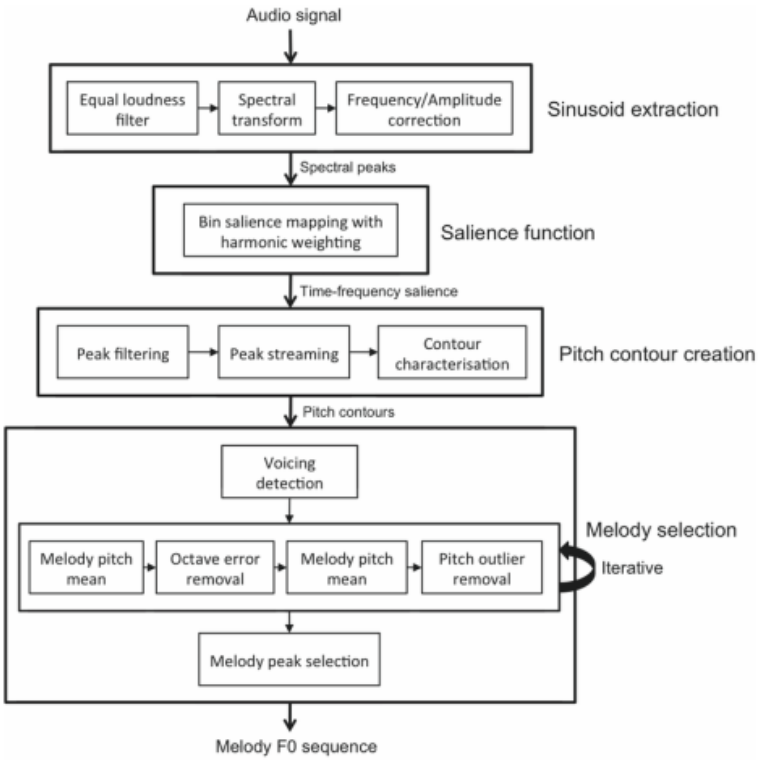
1. Estimar cuándo la melodía está presente y cuándo no lo está (también llamado detección de voz).
2. Estimar el tono o nota musical correcto/a de la melodía, cuando está presente.

Para obtener una explicación más rigurosa, véase el documento indicado en el pie de página[1].

¿Cómo se logra?

Existen, de hecho, varias maneras de lograrlo. La siguiente descripción es una manera de hacerlo y está compuesta por **cuatro** bloques:

1. Extracción de sinusoide.
2. Salience Function (función de prominencia).
3. Creación de contorno.
4. Selección de melodía.



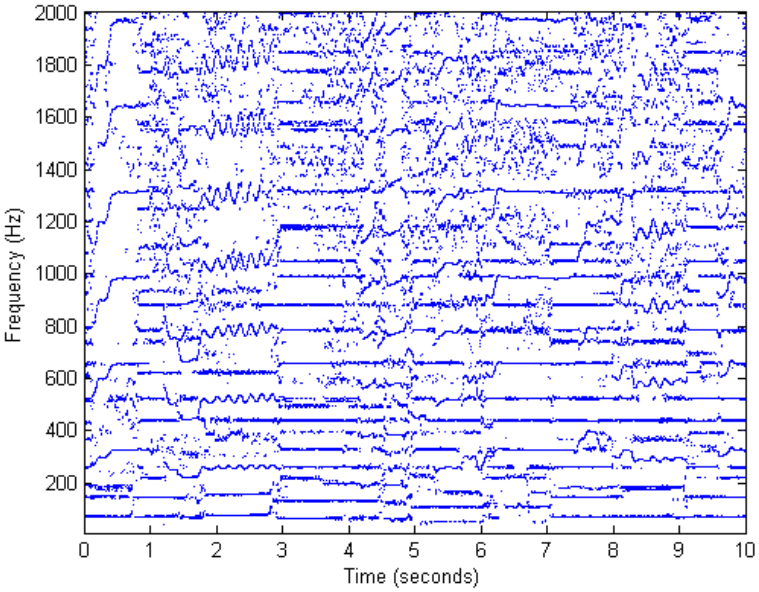
Extracción de Sinusoide

En este bloque, el objetivo es encontrar o hallar qué frecuencias están presentes en la señal de audio en cada instante de tiempo. Luego, usaremos esto para deducir cuáles tonos están presentes en cada instante de tiempo. Pero antes de que vayamos a eso, aplicamos en primera instancia un filtro de "igual volumen" a la señal de audio. Este filtro mejora las frecuencias que son perceptivamente más sensibles para nosotros, y atenúa aquellas que son menos sensibles. Aparte de lograr un "sentido perceptivo", ocurre que este filtro mejora el rango de frecuencias en el cual la melodía

- [1] J. Salamon and E. Gómez, "Melody Extraction from Polyphonic Music Signals using Pitch Contour Characteristics", IEEE Transactions on Audio, Speech and Language Processing, 20(6):1759-1770, Aug. 2012.

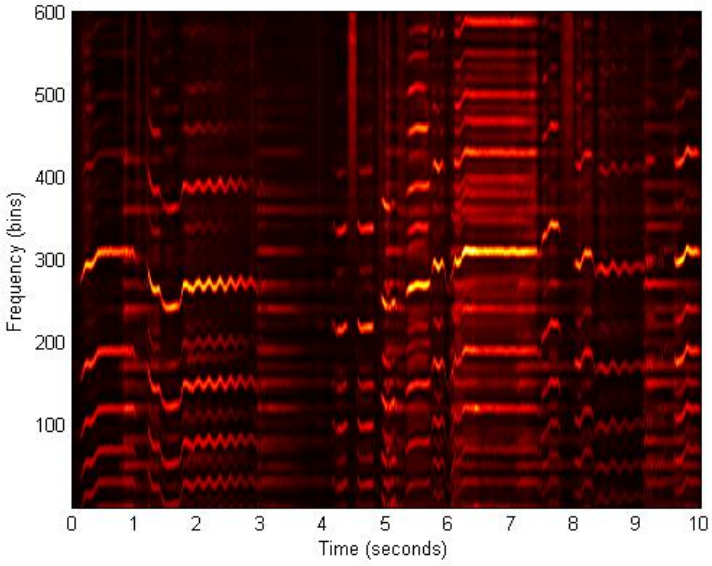
se encuentra se suele encontrar, y atenúa los rangos bajos de frecuencia donde es menos probable encontrar la melodía.

Luego, partimos la señal en pequeños bloques para seguir procesándola, cada bloque representa un momento o instante específico en el tiempo. Aplicamos la Transformada Discreta de Fourier a cada bloque, la cual nos proporciona la intensidad de cada frecuencia en el bloque de audio. Los picos espectrales son los picos de la transformada. Entonces, para cada bloque nos quedamos solo con estas frecuencias de pico y descartamos las demás. Sin embargo, la Transformada Discreta de Fourier tiene una resolución de frecuencia limitada, significando que el valor en Hertz de estas frecuencias pico puedan estar un poco erradas. Para tratar con este problema, refinamos la estimación de la frecuencia otorgada por la Transformada Discreta de Fourier computando la frecuencia instantánea (IF) de cada pico espectral usando la diferencia entre fases consecutivas espectrales. Ahora, por cada instante de tiempo (representado por un bloque de audio) tenemos un conjunto de frecuencias exactas y significativas que están presente en la señal:



Salience Function (Función predominante)

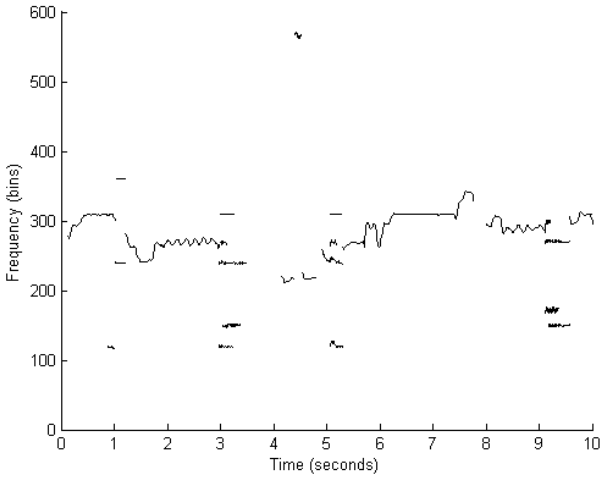
Ahora que sabemos cuáles frecuencias están presentes en la señal de audio en cada instante de tiempo, las usamos para estimar cuales tonos o notas musicales están presentes en cada instante de tiempo y cuán predominantes son. Hay que recordar que mientras que la frecuencia y el tono están relacionados, ¡no son lo mismo! Para obtener un estimado de su predominancia, usamos la suma armónica, por ejemplo, por cada tono posible (dentro de un rango razonable) buscamos una serie armónica de frecuencias que contribuiría a la percepción de este tono. La suma ponderada de la energía de estas frecuencias armónicas es considerada la "predominancia" (salience) de este tono. Repetimos esto por cada instante de tiempo (cada uno de los bloques), llevando a una representación del



tono predominante a través del tiempo, al cual nos referimos como “salience function” (función predominante):

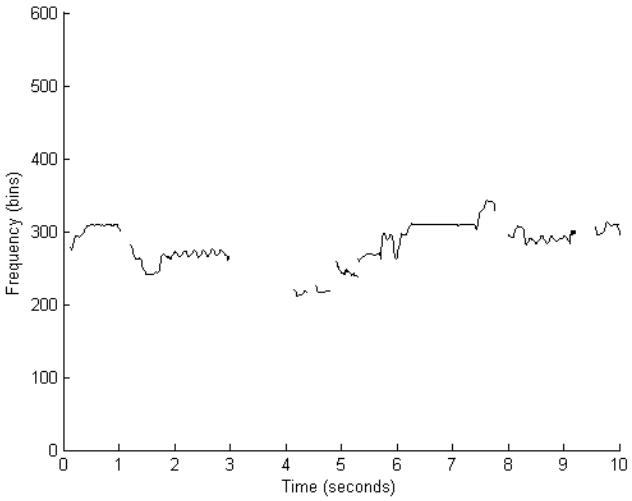
Creación de contorno

Desde la función predominante, buscamos “contornos tonales”. Un contorno tonal representa una serie de valores consecutivos de tonos los cuales son continuos en tiempo y frecuencia. La duración de un contorno tonal puede ser desde una simple nota hasta una frase corta. Para conseguir estos contornos, tomamos los picos de la función predominante en cada instante de tiempo, ya que estos representan los tonos de más predominancia. Luego usamos un conjunto de señales basadas en transmisión auditiva para agrupar estos picos y formar contornos:



Selección de melodía

Entonces, ahora que tenemos todos estos contornos tonales, la tarea restante es determinar cuáles pertenecen a la melodía y cuáles no. Acá se encuentra la parte interesante… Nuestro enfoque está basado en el cálculo de las características del contorno. Eso significa, por cada contorno, computamos un conjunto de características basadas en la evolución de su predominancia y tono. Computamos, por ejemplo, la altura promedio tonal del contorno y su predominancia, la desviación en la trayectoria tonal del contorno y además chequeamos si el contorno contiene vibrato o no. Estudiando la distribución de estas características para contornos que pertenecen a melodías y contornos que pertenecen a acompañamientos, fue posible inferir un conjunto de reglas para filtrar los contornos no melódicos. Luego de aplicar estas reglas de filtro, la melodía restante se ve como la figura siguiente:



Modelo de tres capas

El modelo que utiliza la aplicación para el manejo de los datos es el modelo de tres capas, que consiste en:

1. **Capa de Datos:** esta capa utiliza el ORM SQLAlchemy, mapeador entre objetos y transacciones relacionales, para realizar las conexiones, consultas y modificaciones a la Base de Datos que opera con el motor de base de datos SQLite en nuestro caso. En esta capa se define la estructura de la base de datos que almacena las canciones con sus respectivos análisis y melodía sintetizada.
2. **Capa de Negocio:** esta capa se encarga de:
    a. generar el análisis del audio utilizando el algoritmo “MELODIA”
    b. generar el chromagrama
    c. realizar las interacciones con la capa de datos para guardar, modificar y eliminar audios
    d. grabar audio y generar el archivo de extensión .wav
    e. conectar con la interfaz de pyAudio para permitir la reproducción del audio
3. **Capa de Vista:** esta capa es la encargada de llevar a cabo la interacción entre el usuario y las funciones de la aplicación.

Interfaz Gráfica

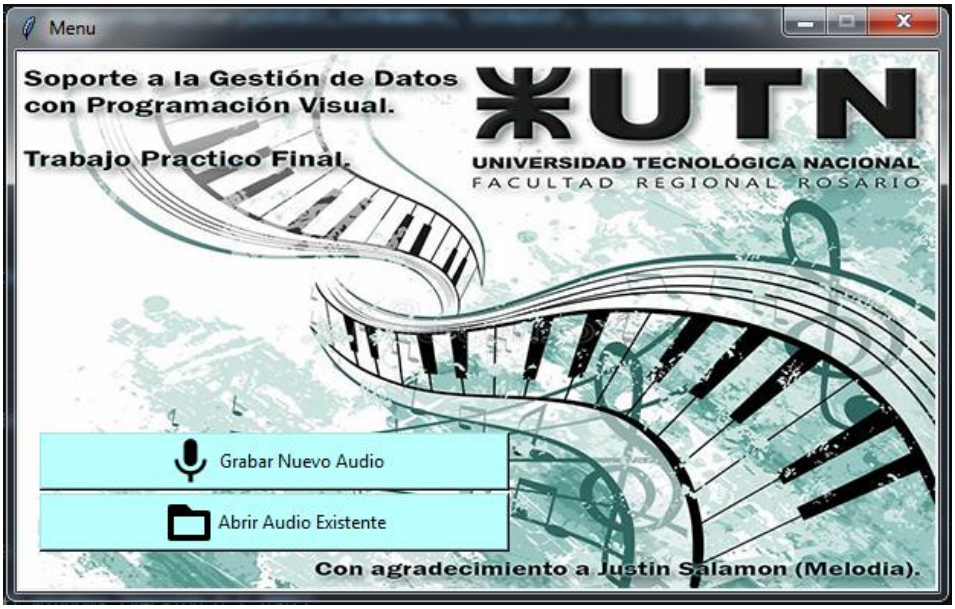
Todo lo referido a interfaz gráfica fue codificado usando la librería Tkinter de Python. Tkinter es el paquete standard de GUI (Graphical User Interface) para Python.

Nuestra aplicación cuenta con cuatro ventanas principales:

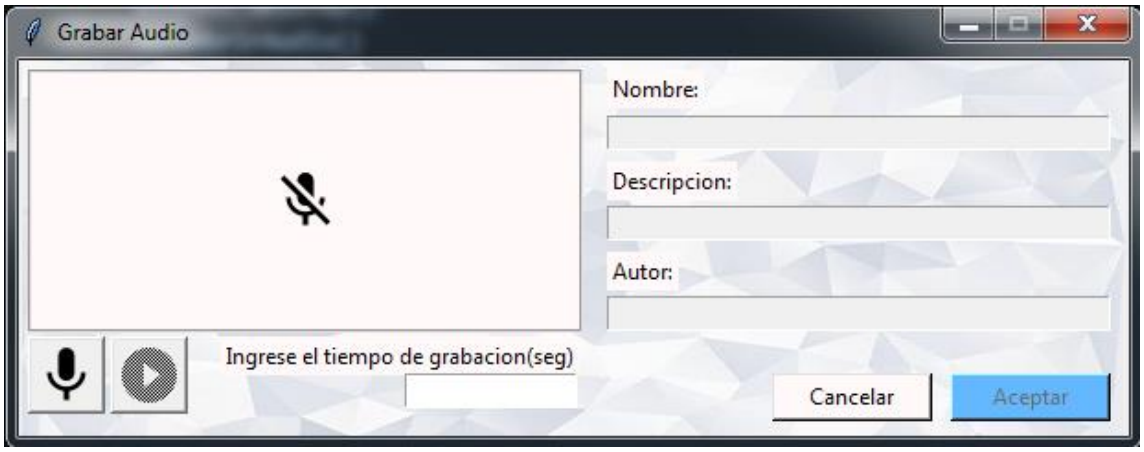
1. Ventana principal: donde es posible elegir entre grabar un nuevo audio o abrir un audio existente
2. Ventana de grabación: donde es posible grabar el audio (y regrabar de ser necesario) y guardar el mismo ingresando nombre (necesario), descripción y autor.
3. Ventana de audios existentes: aquí es posible seleccionar un audio y escuchar una vista previa del mismo, modificarlos o borrarlos. Por otro lado, podemos realizar la selección e ingresar al análisis correspondiente a ese audio llevándonos a la última ventana.
4. Ventana de análisis: aquí se puede visualizar el chromagrama y reproducir el audio original y el audio sintetizado con el objetivo de compararlos.

A continuación, se presentarán imágenes descriptivas.

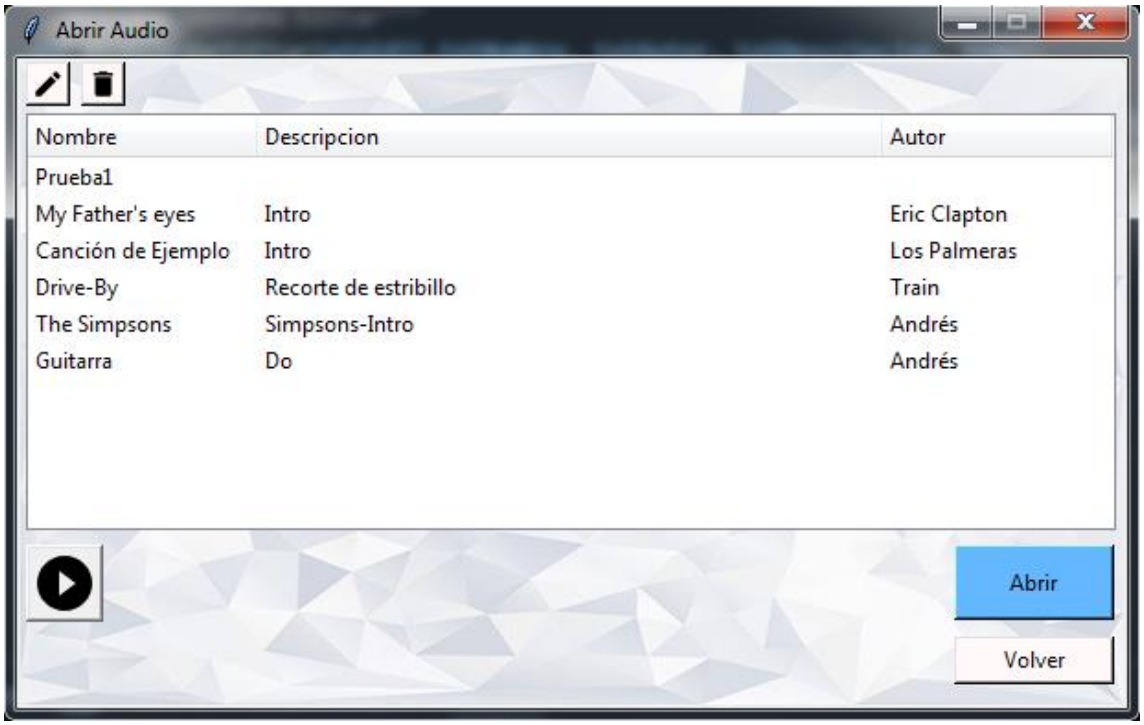
Ventana Principal:



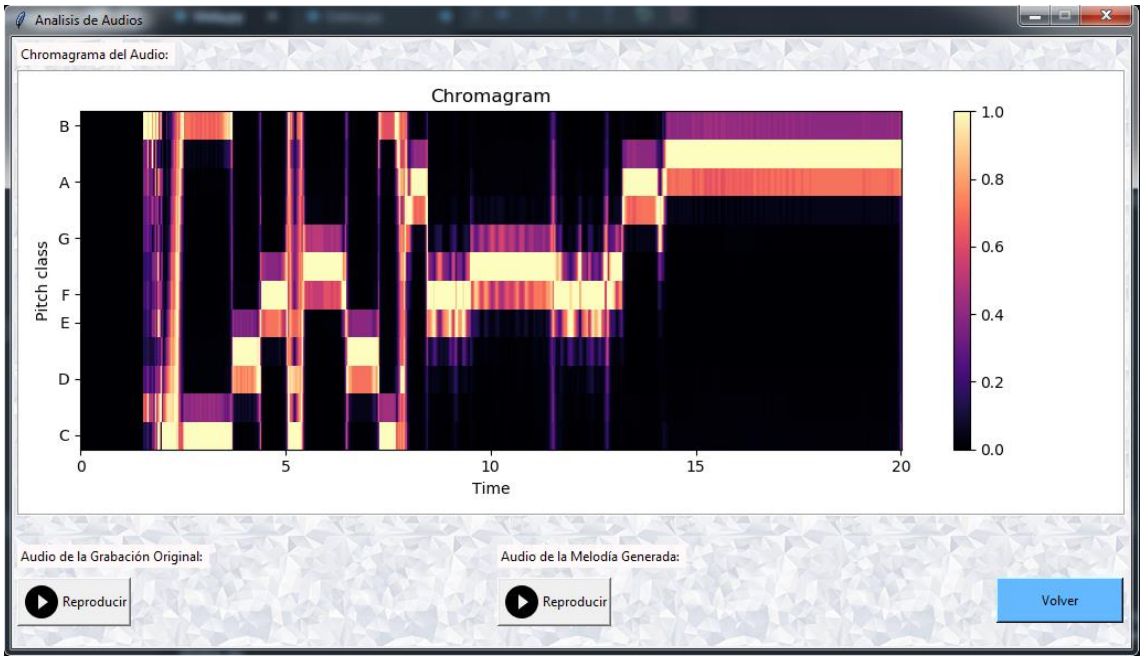
Ventana de Grabación:



Ventana de audios existentes:

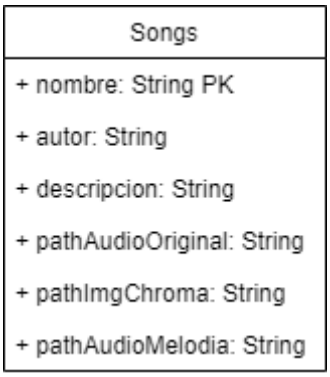


Ventana de Análisis:



Modelo de Dominio y DER

Tanto el modelo de dominio como el DER presentan una sola clase, debido a que, en nuestro caso particular, la única interacción con la base de datos se realiza para administrar las canciones grabadas.



Normalización de la tabla Songs:

Songs(nombre, autor, descripción, pathAudioOriginal, pathImgChroma, pathAudioMelodia)