

DECISIÓN DE LA ESTRUCTURA DE LA BASE DE DATOS

La elección de crear cuatro tablas en la base de datos proporciona una estructura organizada y eficiente para gestionar los datos recolectados de las APIs, asegurando la integridad, escalabilidad y rendimiento del sistema.

Se optó por dividir la información en cuatro tablas distintas para mantener la claridad y la organización. Esto ayuda a evitar la redundancia y facilita el mantenimiento de los datos.

Estructura de las Tablas:

- ***Tabla 1: artistas_spotify***

Esta tabla almacena información sobre los artistas en la plataforma Spotify. Sus columnas incluyen un identificador único (id) y el nombre del artista (nombre_artista).

- ***Tabla 2: canciones_spotify***

Esta tabla contiene detalles sobre las canciones de los artistas en Spotify. Sus columnas incluyen un identificador único (artista_id) que relaciona la canción con el artista, el nombre del artista (artista), el título de la canción (título_cancion), el nombre del álbum (nombre_album), el año de lanzamiento (año) y el género musical (genre).

- ***Tabla 3: info_artistas_last_fm***

Aquí se encuentra información adicional sobre los artistas, recopilada de Last.fm. Las columnas incluyen un identificador (id_lastfm), el nombre del artista (artista_nombre), una biografía (biografia), el número de oyentes (listeners), el conteo de reproducciones (playcount) y una lista de artistas similares (artistas_similares).

- ***Tabla 4: info_artistas_musicbrainz***

Esta tabla proporciona datos de MusicBrainz sobre los artistas, incluyendo un identificador (id_musicbrainz), el nombre del artista (artista_nombre), su tipo (por ejemplo, individual o grupo), el país de origen (pais_origen), el área de origen (area_origen), el inicio y fin de su actividad musical (inicio_actividad y fin_actividad), así como sus fechas de nacimiento y fallecimiento (fecha_nacimiento y fecha_fallecimiento).

Beneficios:

- **Escalabilidad:** Facilita la adición de nuevas características o tablas en el futuro sin complicar la estructura existente.
- **Consultas Eficientes:** Permite realizar consultas más rápidas y específicas, optimizando el rendimiento de la base de datos.

- **Facilidad de Mantenimiento:** Aislado diferentes tipos de datos, es más sencillo actualizar o modificar información sin afectar otras tablas.

Integración con APIs:

- Cada tabla está diseñada para recibir datos específicos de las APIs, lo que permite una fácil integración y actualización de la información.

Representación gráfica que muestra la estructura de la una base de datos

ER DIAGRAM

