



Universidad Experimental de Ciudad Guayana

Coordinación General de Pregrado

Proyecto de Carrera: Ingeniería Informática

Asignatura: Base de Datos II

Sistema Básico de Recomendación de Música con Análisis OLAP Simplificado en NoSQL

Profesora:

Clinia Cordero

Integrantes:

Cesar Peraza V- 30.576.958

Maria Navarro V-30498257

Octavio Malave V-29643097

Índice

Introducción.....	4
Planteamiento del Problema.....	5
Objetivo General.....	5
Objetivo Específicos.....	5
Justificación.....	7
Especificaciones del Sistema.....	7
Requerimientos Funcionales.....	7
Requerimientos no Funcionales.....	8
Casos de Uso.....	9
Modelo de Datos de las Tablas.....	9
Diseño del Sistema.....	12
Diseño de la Bases de Datos en Cassandra.....	12
Tablas de Bases de Datos.....	13
Herramienta y Lenguajes Empleados.....	15
Desarrollo en Apache Cassandra.....	16
Consultas OLAP.....	17
Primera Consulta OLAP.....	18
Segunda Consulta OLAP.....	20
Resultados de Pruebas.....	22
Inicio del Sistema de Recomendaciones de Música.....	22
Inicio de Sesión.....	23
Página Principal.....	24

Escuchado Recientemente.....	24
Canciones Más Famosas por Género y Ciudad.....	26
Canciones Más Famosas Por Mes.....	27
Conclusión.....	29

Introducción

Los sistemas de recomendación han servido de mucha ayuda para que los usuarios puedan ver sus preferencias y así poder hacer sugerencias relevantes, estos los podemos encontrar en muchas lados desde páginas de compras, plataformas interactivas, hasta en plataformas de emisión de contenido, tales como YouTube, Spotify, Facebook, entre otros.

En este proyecto, explora un enfoque simplificado basado en Análisis OLAP y bases de datos NoSQL para mejorar la recomendación de música, mientras que los sistemas que dependen exclusivamente de inteligencia artificial y aprendizaje automático, este modelo se centra en datos históricos y tendencias para identificar canciones populares según la ubicación y el tiempo.

Planteamiento del Problema

La necesidad de desarrollar un sistema de recomendación musical básico surge como una alternativa sencilla y estadística, en lugar de depender de técnicas avanzadas de inteligencia artificial. Este sistema utilizará bases de datos NoSQL, específicamente Apache Cassandra, para almacenar y analizar información de manera eficiente.

A diferencia de los enfoques tradicionales como el collaborative filtering, que utiliza las valoraciones y comportamientos de los usuarios para identificar similitudes y realizar recomendaciones, este sistema se enfocará en patrones de escucha generales. La razón principal de esta decisión es la limitación de tiempo disponible para el desarrollo del proyecto, lo que hace inviable implementar algoritmos más complejos.

Por lo tanto, el objetivo es ofrecer una solución práctica que analice tendencias musicales, permitiendo identificar qué canciones son más populares en diferentes ciudades y períodos de tiempo sin depender de perfiles individuales de usuario.

Objetivo General

Diseñar e implementar un sistema de recomendación de música con un análisis OLAP simplificado, utilizando una base de datos NoSQL(Cassandra).

Objetivo Específicos

1. Analizar problemática y requerimientos en el sistema de recomendación de música.
2. Diseña diagramas de casos de uso y un esquema NoSQL para almacenar información básica sobre usuarios, canciones y escuchas.
3. Implementar consultas OLAP optimizadas en Cassandra Diseñar y ejecutar consultas OLAP sobre una base de datos NoSQL (Apache Cassandra).

4. Desarrollo de un entorno de búsqueda interactivo web que permita a los usuarios seleccionar parámetros de consulta como ciudad, mes o género, generando resultados dinámicos sobre las canciones más reproducidas en un período específico.

Justificación

En la actualidad, los sistemas de recomendación juegan un papel fundamental en la forma en que los usuarios descubren contenido digital. Desde plataformas de música y video hasta comercio electrónico, estas herramientas facilitan la interacción al ofrecer sugerencias basadas en datos recopilados sobre los hábitos de consumo. Sin embargo, muchos sistemas avanzados de recomendación requieren modelos complejos de inteligencia artificial y aprendizaje automático, lo que implica una infraestructura robusta y tiempos de desarrollo prolongados.

Este proyecto propone una alternativa simplificada de recomendación musical, enfocándose en un análisis estadístico basado en datos históricos en lugar de modelos predictivos avanzados. Para ello, se emplea un enfoque de procesamiento analítico en línea (OLAP) que permitirá el análisis de tendencias musicales por ciudad, género y periodo de tiempo, utilizando una base de datos NoSQL (Apache Cassandra).

Especificaciones del Sistema

Requerimientos Funcionales

Gestión de Datos NoSQL

- Implementación de una base de datos NoSQL (Apache Cassandra).
- Estructura de datos simplificada para usuarios, canciones y escuchas.

Recomendación de Canciones

- Desarrollo de un algoritmo básico de recomendación de música.
- Recomendaciones basadas en género musical o ubicación del usuario.
- No incluye modelos avanzados de filtrado colaborativo o híbridos.

Análisis OLAP Simplificado

- Identificación de dimensiones clave (usuario, género, mes).
- Definición de una jerarquía de tiempo (día → mes).
- Implementación de medidas analíticas (número de escuchas).
- Diseño conceptual de un cubo OLAP simplificado.

Consultas OLAP y Visualización de Datos

- Implementación de al menos 2-3 consultas OLAP básicas.
- Ejemplo: "Número total de escuchas por género y mes"
- Generación de visualizaciones simples como tablas o gráficos.

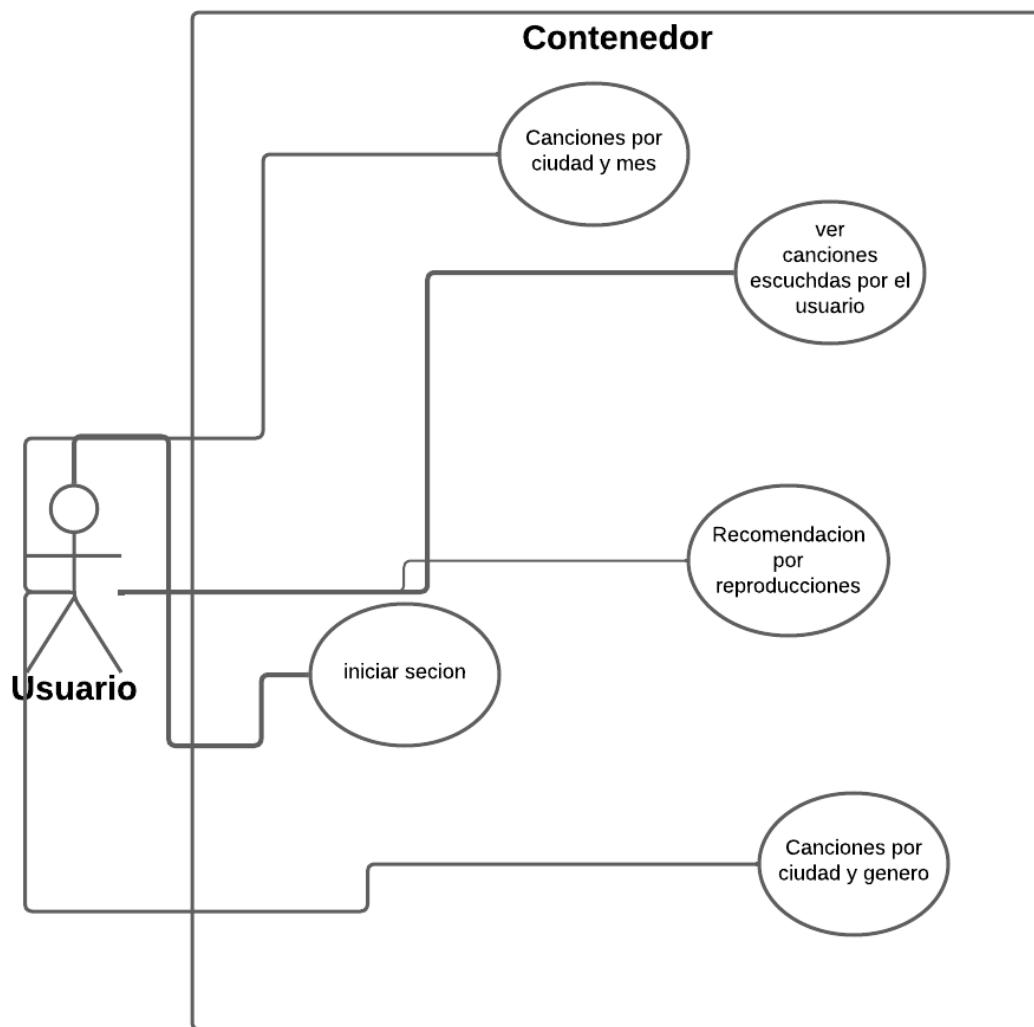
Interfaz de Usuario

- Creación de una interfaz web interactiva en React.
- Los usuarios podrán seleccionar ciudad, género y mes para buscar tendencias.
- Conexión con el backend para obtener resultados dinámicos.

Requerimientos no Funcionales

- El sistema debe ser escalable para gestionar grandes volúmenes de datos sin degradar el rendimiento.
- El sistema debe garantizar la integridad y consistencia de los datos almacenados en Apache Cassandra.

Casos de Uso



Modelo de Datos de las Tablas

- **Usuarios (usuarios):** Almacena información básica de los usuarios. Incluye id, password, nombre, ciudad y edad.

usuarios	
Campo	Tipo
id (PK)	uuid
password	string
nombre	string
ciudad	string
edad	int

- **Canciones (canciones):** Guarda detalles de cada canción. Contiene id, título, artista, álbum, género, duración y portada.

Canciones	
Campo	Tipo
id (PK)	uuid
artista	string
álbum	string
género	string
duración	int
portada	string

- **Registro de escuchas (usuario_cancion):** Historial de reproducción de canciones por usuario. Relaciona usuario_id con cancion_id, guardando fecha_reproduccion. Permite consultas OLAP sobre frecuencia de reproducción por usuario.

usuarios_cancion	
Campo	Tipo
cancion_id (PK)	uuid
fecha_reproduccion	timestamp

- **Canciones más escuchadas** (cancionesPorusuarios): Ranking de canciones según cantidad total de reproducciones. Guarda total de reproducciones por cancion_id. Útil para consultas de tendencias globales.

canciones_usuario	
Campo	Tipo
cancion_id (PK)	uuid

- **Canciones más escuchadas por género** (canciones_mas_escuchadas_por_genero): Analiza popularidad dentro de un género específico. Usa PRIMARY KEY (género, cancion_id) para consultas rápidas por género. Contiene reproducciones para análisis de tendencias en géneros.

canciones_mas_escuchadas_por_genero	
Campo	Tipo
cancion_id (PK)	uuid
género	string

- **Canciones más escuchadas por ciudad** (cancion_mas_escuchadas_ciudad): Permite encontrar las canciones más populares en cada ubicación. Usa PRIMARY KEY (ciudad, cancion_id), permitiendo consultas eficientes por ciudad.

canciones_mas_escuchadas_ciudad	
Campo	Tipo
cancion_id (PK)	uuid
ciudad	string

- **Canciones populares por mes** (canciones_populares_por_mes) → Base para análisis OLAP sobre tendencias mensuales. Usa PRIMARY KEY(ciudad, mes), cancion_id) para análisis por periodo. Permite consultas como "Top canciones de cada mes en una ciudad"

canciones_mas_escuchadas_por_mes	
Campo	Tipo
cancion_id (PK)	uuid
mes	string
ciudad	string

Diseño del Sistema

Diseño de la Bases de Datos en Cassandra

Para elaborar el sistema de recomendaciones de música, se hizo uso del sistema de gestión de bases de datos Cassandra, en ella se pueden diseñar o desarrollar base de datos NoSQL (donde los datos se organizan en un modelo columnar o un modelo clave-valor), permite realizar inserciones, modificar y eliminar datos, además esta cuenta con el diseño query-first para optimizar las consultas y en comparación con las bases de datos SQL, estas son aptas para administrar grandes volúmenes de datos. Dentro del sistema de recomendación de música, los usuarios podrán disfrutar de una increíble interacción con la interfaz y visualizar consultas que se realizaron mediante el SGBD Cassandra.

Definición del KEYSPACE de la base de datos en cassandra:

```
CREATE KEYSPACE IF NOT EXISTS musicdb WITH replication = {
    'class': 'SimpleStrategy',
    'replication_factor': 1
};
```

Tablas de Bases de Datos

- Tabla usuarios

```
async function crearTablas() {
  const queries = [
    /** LISTO */
    `CREATE TABLE IF NOT EXISTS usuarios (
      id UUID PRIMARY KEY,
      password TEXT,
      nombre TEXT,
      ciudad TEXT,
      edad INT,
    );`,
```

- Tabla canciones

```
/**listo */
`CREATE TABLE IF NOT EXISTS canciones (
  id UUID PRIMARY KEY,
  titulo TEXT,
  artista TEXT,
  album TEXT,
  genero TEXT,
  duracion TEXT,
  portada TEXT
);`,
```

- Tabla usuario_cancion

```
/**listo */
`CREATE TABLE usuario_cancion (
  usuario_id UUID,
  cancion_id UUID,
  titulo TEXT,
  artista TEXT,
  album TEXT,
  genero TEXT,
  duracion TEXT,
  portada TEXT,
  fecha_reproduccion TIMESTAMP,
  PRIMARY KEY (usuario_id, fecha_reproduccion, cancion_id)
);`,
```

- Tabla canciones_mas_escuchadas_por_genero

```
/*Recomendacion canciones mas escuchadas por genero */
`CREATE TABLE canciones_mas_escuchadas_por_genero (
    genero TEXT,
    cancion_id UUID,
    titulo TEXT,
    artista TEXT,
    album TEXT,
    duracion TEXT,
    portada TEXT,
    reproducciones INT,
    PRIMARY KEY (genero, cancion_id)
);`,
```

- Tabla canciones_mas_escuchadas_ciudad

```
/* Segunda consulta olap */
`CREATE TABLE cancion_mas_escuchadas_ciudad (
    ciudad TEXT,
    cancion_id UUID,
    titulo TEXT,
    artista TEXT,
    album TEXT,
    genero TEXT,
    duracion TEXT,
    portada TEXT,
    reproducciones INT,
    PRIMARY KEY (ciudad, reproducciones, cancion_id)
);`,
```

- Tabla canciones_mas_escuchadas_por_mes

```
/* primera consulta olap */
`CREATE TABLE canciones_populares_por_mes (
    ciudad TEXT,
    mes TEXT,
    cancion_id UUID,
    titulo TEXT,
    artista TEXT,
    album TEXT,
    genero TEXT,
    duracion TEXT,
    portada TEXT,
    reproducciones INT,
    PRIMARY KEY ((ciudad, mes), reproducciones, cancion_id)
);`
```

- Tabla cancionesPorusuarios

```
/* Recomendacion canciones mas escuchadas*/
`CREATE TABLE cancionesPorusuarios (
    cancion_id UUID,
    titulo TEXT,
    artista TEXT,
    album TEXT,
    genero TEXT,
    duracion TEXT,
    portada TEXT,
    total INT,
    PRIMARY KEY(cancion_id)
);`
```

Herramienta y Lenguajes Empleados

Para llevar a cabo el sistema de recomendaciones de canciones, se implementaron diversas herramientas , las cuales nos servirán para el almacenamiento del proyecto, la creación de la base de datos, el desarrollo de la interfaz, las cuales fueron:

- Git: Es un Sistema de control de versiones que se distribuye en diferentes ramas, este permite llevar un seguimiento de los cambios realizados en el archivo del proyecto, donde cada uno de los desarrolladores trabaja desde una copia para luego subir la misma al archivo original.
- Apache Cassandra: Como se necesita crear un sistema de recomendación de música con base de datos NoSQL, se optó por el sistema manejador de base de datos Cassandra, el cual permite manejar grandes cantidades de datos y es flexible a la hora de modelar datos.
- Nodejs: Node.js es un entorno de ejecución para JavaScript que permite desarrollar aplicaciones web o frameworks escalables y eficientes en el lado del servidor. Debido a su arquitectura basada en eventos y su modelo asincrónico, es una buena opción para construir aplicaciones web en tiempo real.
- Vite: Es una herramienta de desarrollo para el frontend, este contiene la arquitectura de desarrollo instantáneo, lo que conlleva a que los usuarios que la utilicen visualicen los cambios realizados al momento, esta herramienta se empleó junto a React, la cual es una biblioteca de JavaScript, en ella también se puede desarrollar en jsx que es una combinación de HTML y CSS. Además para facilitar el desarrollo del frontend, react hace uso de componentes, estos son archivos de códigos que se pueden reutilizar en cualquier otro archivo, permitiendo el ahorro de trabajo.

Desarrollo en Apache Cassandra

La implementación de Apache Cassandra fue la parte más fundamental de este proyecto, implementando el modelo de datos NoSQL por primera vez aplicando los conocimientos aprendidos.

Se utilizó la consola de comandos de Cassandra (cql) se realizó la creación del keyspace que se utilizará en la página web, para esto se ejecutó el comando CREATE KEYSPACE.

Para que cualquier persona que quiera probar y ejecutar la página web, se decidió desarrollar un script de migraciones para las tablas de la base de datos, estas migraciones permiten crear todas las tablas necesarias para la página web sin necesidad de hacerlo manualmente una a una, lo que sería un proceso un poco pesado, sobretodo en casos de una gran cantidad de tablas. Esto además reduce a 0 la posibilidad de errores humanos en la implementación de los nombres de dichas tablas o atributos.

Por último, cómo es necesario tener datos ingresados en la base de datos para la correcta realización de pruebas o para ver el potencial completo de la página, al igual que las migraciones, se desarrolló un script de seeders para poblar dicha base de datos. Esto permite que, al ejecutar un comando, todas las tablas sean llenadas con un dataset suministrado de más de 100 registros por tabla, evitando tener que insertar cada registro individualmente mediante la consola de Cassandra.

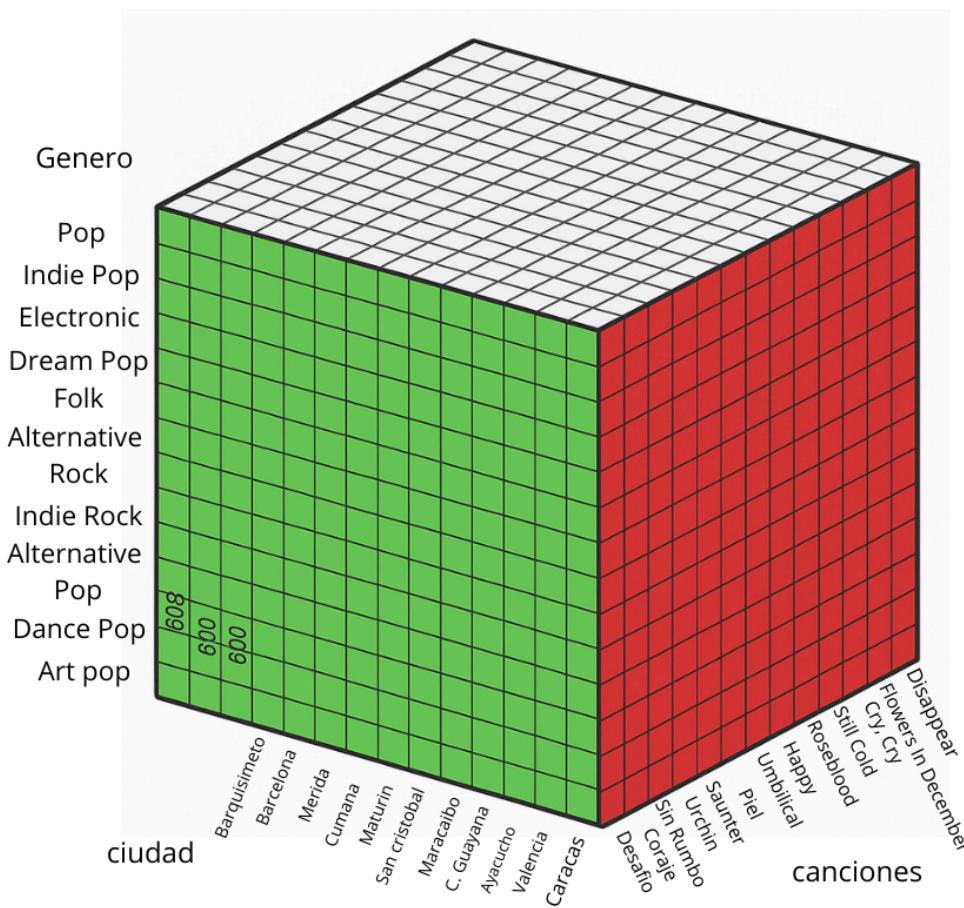
Consultas OLAP

Un cubo OLAP es una estructura de datos que pueden mostrar y sumar grandes cantidades de datos, a la vez que proporcionan a los usuarios acceso mediante búsqueda a los puntos de datos. De este modo, los datos se pueden resumir o reorganizar según sea necesario, para procesar la variedad más amplia de preguntas correspondientes al área de interés de un usuario.

Primera Consulta OLAP

La primera consulta OLAP se basa en la tabla *Canciones más escuchadas por ciudad*.

En ella, se definen tres parámetros clave para el análisis: **ciudad**, **género** y **canción**. Esta consulta permite identificar patrones de consumo musical en distintas ubicaciones, facilitando el estudio de tendencias y preferencias de los usuarios.



Las dimensiones utilizadas para el análisis OLAP fueron género, ciudad y canciones, mientras que la medida para determinar el análisis fueron las reproducciones.

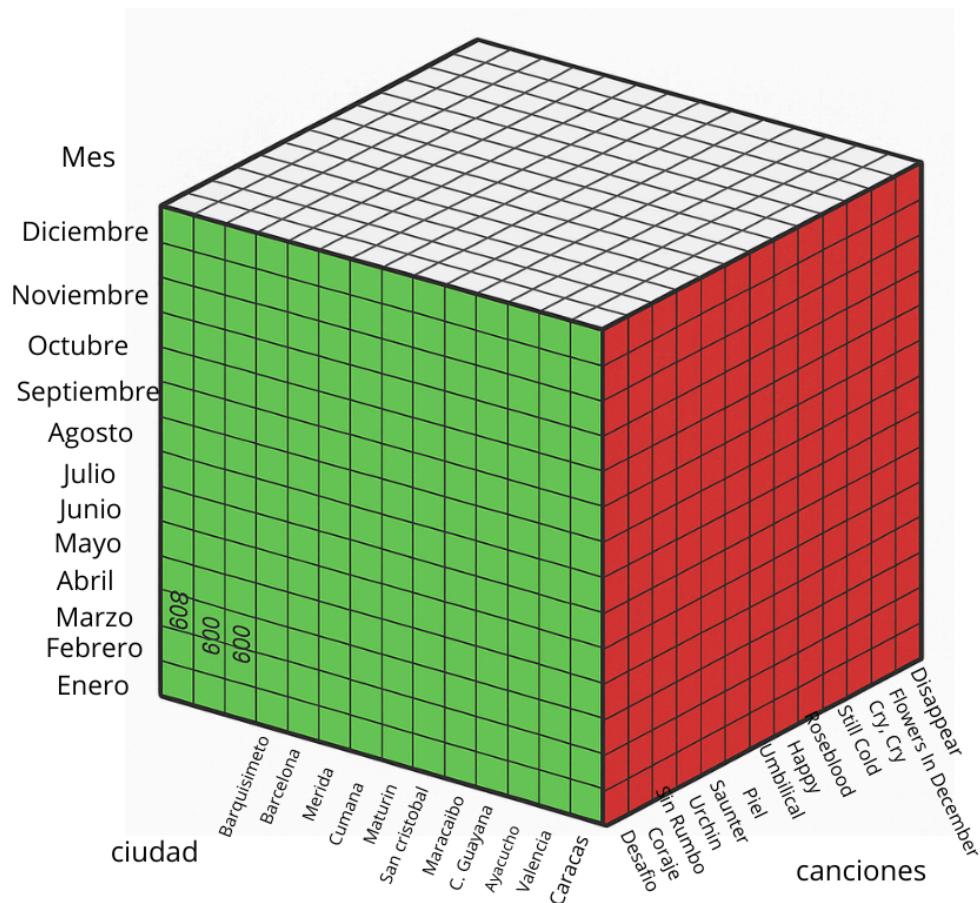
Dado que no contamos con un histórico de reproducciones fijas, la tabla se llena mediante scripts para realizar pruebas. Como dichos datos son generados aleatoriamente cada vez que se realiza una migración a la base de datos, no es posible mostrar una gráfica representativa, ya que su contenido varía constantemente. Este enfoque nos ayuda a simular

una carga de datos real de usuarios y evaluar el comportamiento del sistema bajo diferentes escenarios. Lo más cercano que se puede mostrar es una tabla de algunos ejemplos con los datos generados en la ejecución actual de la base de datos con los datos actuales extraídos de la tabla “cancion_mas_escuchadas_ciudad”.

Ciudad	Genero	Titulo	Artista	Album	Reproducciones
Barcelona	Folk	Sister	Sufjan Stevens	Seven Swans	19481
Barcelona	Folk	To Be Alone With You	Sufjan Stevens	Seven Swans	18960
Barcelona	Folk	two reverse	Adrienne Lenker	Songs	18610
Ciudad Guayana	Pop	These Walls	Dua Lipa	Radical Optimism	19973
Ciudad Guayana	Pop	Fainted Love	Conan Gray	Found Heaven	19891
Ciudad Guayana	Pop	Guilty Pleasure	Chappel Roan	The Rise and Fall of a Midwest Princess	19311
Caracas	Indie Rock	Stay Down	Boygenius	Boygenius	19765
Caracas	Indie Rock	Neighborhood #4 (7 kettles)	Arcade Fire	Funeral	19695
Caracas	Indie Rock	Haiti	Arcade Fire	Funeral	16983
Mérida	Alternative Rock	Paranoid Android	Radiohead	Ok Computer	19432
Mérida	Alternative Rock	Dream Brother	Jeff Buckley	Grace	18776
Mérida	Alternative Rock	Lover, You Should've Come Over	Jeff Buckley	Grace	18601

Segunda Consulta OLAP

La segunda consulta OLAP se basa en la tabla *Canciones populares por mes*, donde se definen tres parámetros clave: **ciudad**, **mes** y **canción**. En este análisis, se identificaron la canción más reproducida en cada ciudad durante un mes determinado, lo que permite observar tendencias de consumo musical y preferencias locales.



De igual manera que la tabla anterior, la cantidad de reproducciones se genera de manera aleatoria mediante scripts, por lo que los resultados varían por cada migración a la

base de datos y depende de quien esté ejecutando el programa y cuando lo ejecute. Sin embargo, se puede mostrar una tabla con algunos datos de ejemplo extraídos de la ejecución actual para dar una demostración más visual de cómo funciona este análisis.

Ciudad	Mes	Titulo	Artista	Album	Reproducciones
Barcelona	Enero	Giorgio by Moroder	Daft Punk	Random Access Memories	19998
Ciudad Guayana	Enero	Casual	Chappel Roan	The Rise and Fall of a Midwest Princess	19989
Caracas	Enero	The Last Time I Saw Richard	Joni Mitchell	Blue	19941
Barcelona	Febrero	AMERICA HAS A PROBLEM	Beyonce	Renaissance	19966
Ciudad Guayana	Febrero	Levitate	Twenty One Pilotsq	Trench	19917
Caracas	Febrero	360	Charli XCX	Brat	19999
Barcelona	Marzo	The Greatest	Billie Eilish	HIT ME HARD AND SOFT	19960
Ciudad Guayana	Marzo	Club Classics	Charli XCX	Brat	19976
Caracas	Marzo	Subterranean Homesick Alien	Radiohead	Ok Computer	19884
Barcelona	Abril	Forever With Me	Conan Gray	Found Heaven	19994
Ciudad Guayana	Abril	Angel on a Satellite	Magdalena Bay	Imaginal Disk	19802
Caracas	Abril	BREAK MY	Beyonce	Renaissance	19950

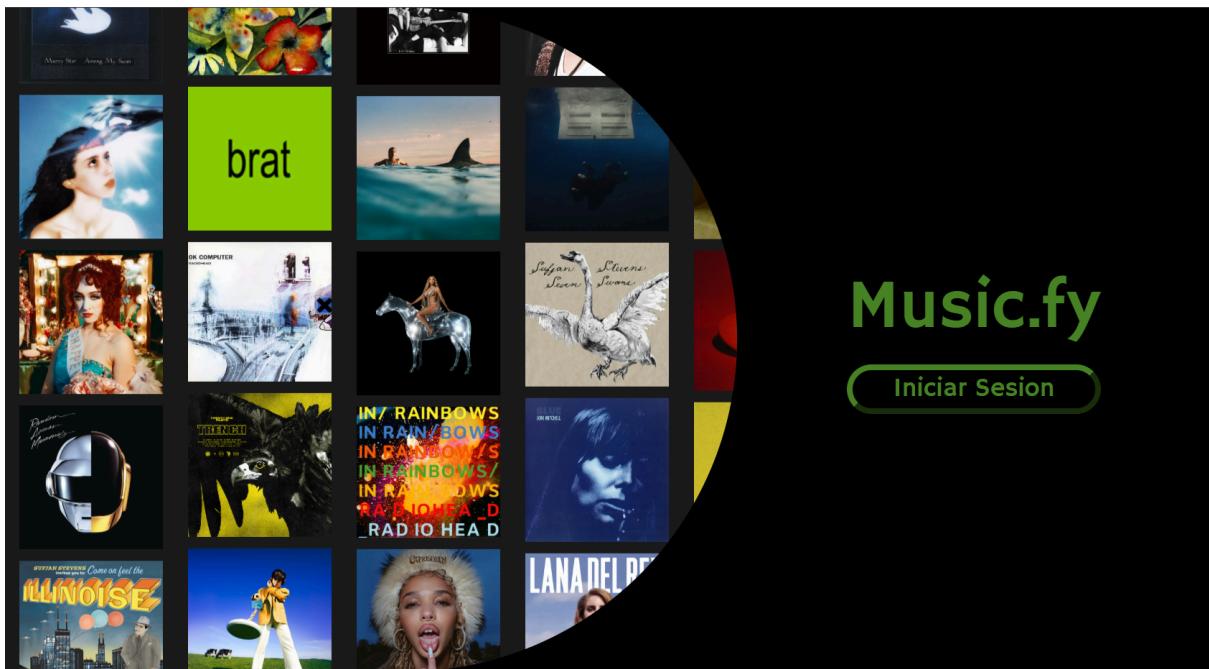
		SOUL			
Barcelona	Mayo	Hush	The Marias	CINEMA	19812
Ciudad Guayana	Mayo	After Midnight	Chappel Roan	The Rise and Fall of a Midwest Princess	19920
Caracas	Mayo	Sin Rumbo	Arca	Arca	19964
Maracaibo	Diciembre	In The Devil's Territory	Sufjan Stevens	Seven Swans	19825
Maracaibo	Junio	Roseblood	Mazzy Star	Among My Swan	19925
Maracaibo	Noviembre	Calling U Back	The marias	CINEMA	19988
Maturín	Enero	Boys & Girls	Conan Gray	Found Heaven	19985
Maturín	Febrero	Killing Time	Magdalena Bay	Imaginal Disk	19937
Maturín	Marzo	To Be Alone WIth You	Sufjan Stevens	Seven Swans	19989

Resultados de Pruebas

En este apartado se muestran las pruebas funcionales del sistema de recomendaciones de música, es decir, el resultado visual al entrar en la página. Se visualizará como es la interfaz gráfica para los usuarios y que resultados les arroja dependiendo de la acción que realicen.

Inicio del Sistema de Recomendaciones de Música

En este apartado se muestra la página de inicio del sistema en donde se puede visualizar todos los álbumes disponibles que se encuentran en la base de datos de la plataforma , incluye un botón de iniciar sesión para poder ingresar.

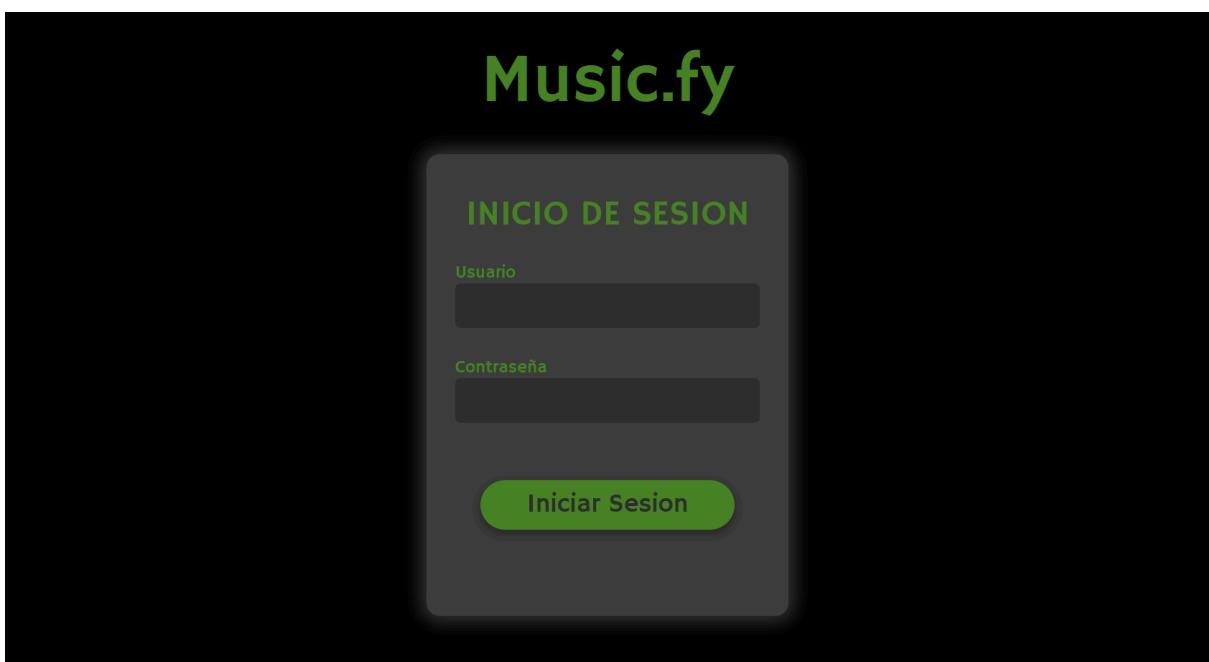


Music.fy

Iniciar Sesión

Inicio de Sesión

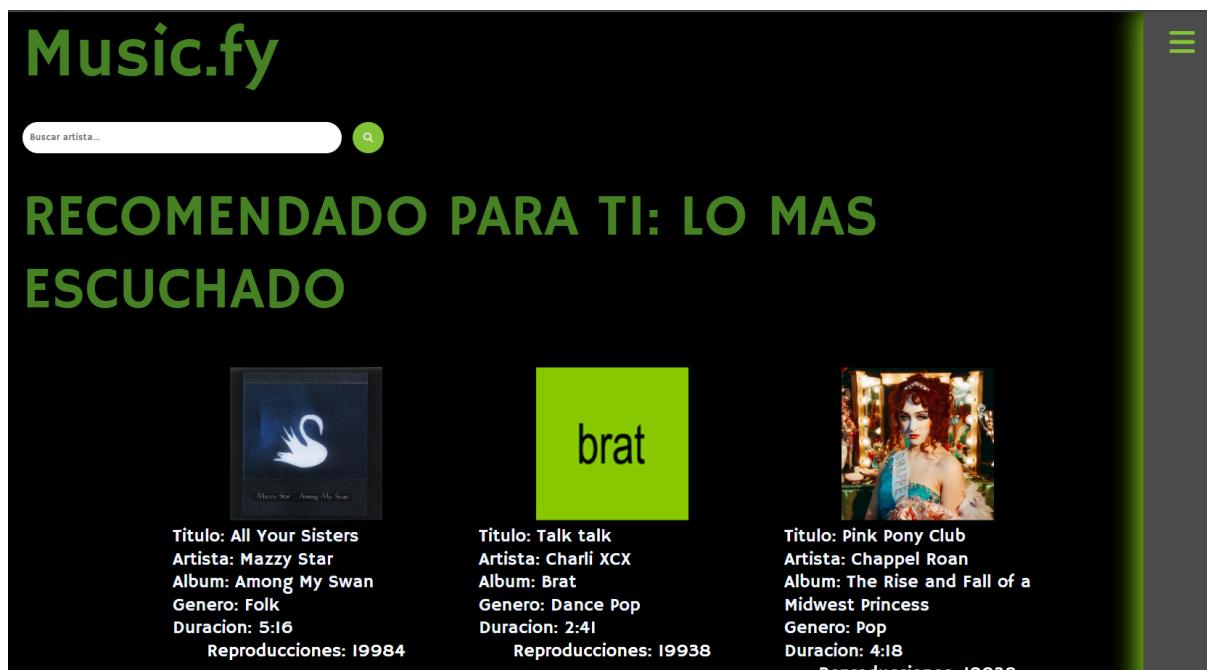
Este apartado es el resultado de apretar el botón “Iniciar Sesión” de la página de inicio, aquí el usuario podrá iniciar sesión con su usuario y contraseña para llegar a poder visualizar la página principal.



Página Principal

Dentro de la página principal lo primero que verá el usuario son las canciones recomendadas por la página, estas son las más escuchadas por todos los usuarios. Además, cuenta con una barra lateral (que también está incluida en las demás vistas), la cual tiene las siguientes opciones :

- Escuchado recientemente
- Canciones mas famosas por género y ciudad
- Canciones mas famosas por mes
- Cerrar sesión



Escuchado Recientemente

El usuario podrá ver las canciones que él o ella haya escuchado recientemente, muestra todos los datos de dichas canciones junto a su fecha de reproducción. Incluye una barra lateral con lo siguiente:

- Volver a la página principal

- Canciones mas famosas por género y ciudad
- Canciones mas famosas por mes
- Cerrar sesión

Music.fy

ESCUCHADO RECENTEMENTE

07 de diciembre de 2025



Artista: Adrienne Lenker
 Album: Songs
 Genero: Folk
 Duracion: 2:44

not a lot,
 just forever

27 de noviembre de 2025



Artista: Billie Eilish
 Album: Happier Than Ever

Music.fy

ESCUCHADO RECENTEMENTE

07 de diciembre de 2025



Artista: Adrienne Lenker
 Album: Songs
 Genero: Folk
 Duracion: 2:44

not a lot,
 just forever

27 de noviembre de 2025



Artista: Billie Eilish
 Album: Happier Than Ever



Luis

Volver

Canciones mas famosas
por genero y ciudad

Canciones mas famosas
por mes

Cerrar Sesión

Canciones Más Famosas por Género y Ciudad

En este apartado el usuario podrá filtrar por ciudad y género para ver las canciones más escuchadas en dicho género y dicha ciudad, incluye una barra lateral con lo siguiente:

- Volver a la página principal
- Escuchando recientemente
- Canciones mas famosas por mes
- Cerrar sesión



The screenshot shows the Musicfy app interface. At the top left is the logo 'Music.fy'. On the top right is a user profile icon with the name 'Luis'. Below the profile are four menu items: 'Volver', 'Escuchados recientemente', 'Canciones mas famosas por mes', and 'Cerrar Sesión'. The main content area is titled 'LO MAS ESCUCHADO'. It lists two tracks: 'Maturín' by Alternative Rock and 'Dream Brother' by Jeff Buckley. Each track has its artist, album, genre, and duration listed. Below the tracks is a section for 'Reproducciones: 19987'.

Artista	Album	Género	Duración
Jeff Buckley	Grace	Alternative Rock	5:26
Radiohead	In Rainbows	Alternative Rock	2:02

Canciones Más Famosas Por Mes

Por último, en esta página el usuario podrá filtrar por ciudad y mes para ver las canciones más escuchadas en dicho mes y dicha ciudad, incluye una barra lateral con lo siguiente:

- Volver a la página principal
- Escuchando recientemente
- Canciones mas famosas por género y ciudad
- Cerrar sesión



LO MAS ESCUCHADO EN

Barcelona

2025-09

Buscar



Artista: Arcade Fire
Album: Funeral
Genero: Indie Rock
Duracion: 5:35

Wake Up

Reproducciones: 19916



Artista: Sufjan Stevens
Album: Illinois
Genero: Pop
Duracion: 0:19



Luis

Volver

Escuchados recientemente

Canciones mas famosas
por genero y ciudad

Cerrar Sesión

LO MAS ESCUCHADO

Barcelona

2025-09

Buscar



Artista: Arcade Fire
Album: Funeral
Genero: Indie Rock
Duracion: 5:35

Wake Up

Reproducciones: 19916



Artista: Sufjan Stevens
Album: Illinois
Genero: Pop
Duracion: 0:19

Conclusión

El desarrollo de un sistema básico de recomendación de música utilizando análisis OLAP simplificado en una base de datos NoSQL empleado Apache Cassandra, representa una solución rápida, práctica y eficiente para identificar tendencias musicales, además permite analizar grandes datos históricos para hallar y concluir cuáles las canciones más populares con respecto ciudad, género y periodo de tiempo, este sistema facilita las recomendaciones de música relevantes y personalizadas a nivel estadístico. Además en este sistema el usuario puede interactuar dentro de la interfaz que nos brinda el servidor y visualizar las tendencias musicales de manera dinámica.

Este proyecto demuestra lo efectivo y adecuado que es desarrollar un sistema de recomendación mediante la implementación de una base de datos NoSQL y el análisis OLAP, sin la necesidad de utilizar tecnologías complejas para diseñar sistemas como este.