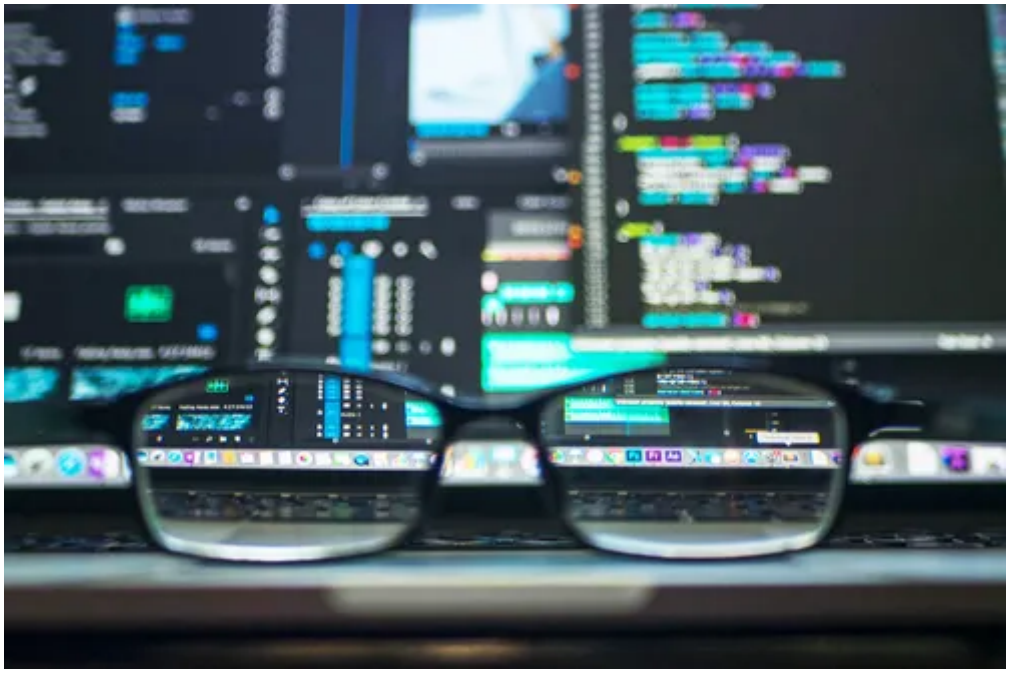
Proyecto Inteligencia Artificial y Big Data

Grupo B



Héctor Fernández Pimienta, Ignacio López García y Francisco José Puertas Teba

Fase 1: Análisis de Datos y Recomendación Musical.

El presente proyecto aborda la dificultad que enfrentan usuarios y analistas para explorar y entender las tendencias del vasto catálogo musical, así como para descubrir música de manera objetiva. Proponemos el desarrollo de una solución de Big Data e Inteligencia Artificial que integra un pipeline de ingesta y procesamiento con un Dashboard analítico (Ruta B: Spark + Dashboard).

La solución técnica se compone de dos módulos principales: (1) un data mart actualizado con métricas de canciones, artistas y géneros, visualizado a través de un Dashboard interactivo, y (2) un motor de recomendación basado en similitud de características acústicas (ej. bailabilidad, energía, valence), entrenado con Spark y versionado con MLflow.

El valor de negocio se centra en la optimización de la curación de playlists y la reducción del tiempo de análisis de tendencias. Los KPIs de éxito incluyen alcanzar una latencia de recomendación de **≤ 500ms**, una reducción del **25%** en el tiempo de análisis de tendencias y una tasa de aceptación de recomendaciones **≥ 70%**. El proyecto se desarrollará bajo prácticas MLOps y un estricto cumplimiento ético, auditando sesgos de género/origen en las recomendaciones.

1. Formulación del problema

La proliferación del contenido en plataformas de streaming musical como Spotify genera dos desafíos claves:

1.1 Saturación analítica

Los analistas y curadores de contenido tienen dificultades para obtener insights rápidos y actualizados sobre el comportamiento de las canciones (tendencias, distribución de características, anomalías) sin recurrir a herramientas complejas o dispares.

1.2 Descubrimiento subjetivo

Los sistemas de recomendación populares se basan principalmente en filtros colaborativos o historial de escucha, lo que a menudo refuerza el sesgo de popularidad (popularity bias).

Existe la necesidad de un mecanismo de descubrimiento objetivo que priorice las características intrínsecas de la música.

Problema a resolver: ¿Cómo podemos unificar la visualización de las estadísticas del catálogo musical con un motor de recomendación que utiliza las características acústicas como base objetiva para identificar canciones similares, mitigando el sesgo de popularidad?

2. Contexto Sectorial

El sector de la música en streaming es altamente competitivo, con un crecimiento exponencial en el número de canciones disponibles. Las discográficas y los curadores de contenido necesitan herramientas de análisis para tomar decisiones informadas sobre marketing y promoción. La capacidad de identificar patrones y tendencias rápidamente (Módulo Dashboard) y de ofrecer micro-recomendaciones precisas (Módulo IA) proporciona una ventaja competitiva en la retención de usuarios y en la eficiencia operativa de la curación de contenidos.

3. Usuarios y partes involucradas.

Parte interesada	Rol y Objetivo
Analista de Datos/Curadores de Música (Usuario Principal)	Utilizan el Dashboard para monitorizar tendencias, realizar benchmarking de artistas y buscar patrones. Necesitan insights rápidos.
Equipo de desarrollo interno	Utiliza la API de Recomendación y la infraestructura (MLflow, Spark) para integrarla en otros productos o realizar A/B testing.
Usuario Final (Secundario/Indirecto)	Se beneficia indirectamente de la calidad de las recomendaciones y del descubrimiento musical preciso.

4. Riesgos

Los riesgos se han analizado en función del impacto (I: Bajo 1 – Alto 5) y la probabilidad (P: Baja 1 – Alta 5).

Riesgo	Impacto	Probabilidad	Prioridad (I * P)	Plan de mitigación
R1. Disponibilidad de Datos/API	4	4	16 (Alto)	Establecer una fuente de datos alternativa (ej. Kaggle dataset de features históricos) para entrenamiento inicial, si la API de Spotify restringe las peticiones
R2. Sesgo Algorítmico (Ético)	5	3	15 (Alto)	Incluir un hito de auditoría de sesgos (género/origen) y aplicar técnicas de remuestreo/reponderación si se detecta una disparidad significativa en las recomendaciones.
R3. Escalabilidad del Feature Engineering	3	4	12 (Medio)	Priorizar el uso de Spark para el preprocesamiento de características (feature engineering) para asegurar la escalabilidad desde la Fase 1.
R4. Latencia del Modelo en Producción	4	2	8 (Medio)	Desplegar el modelo de similitud en un framework ligero y optimizado (ej. ONNX) y utilizar la versión registrada en MLflow.

5. Valor de negocio y KPIs de éxito.

5.1 Valor de Negocio

Aspecto	Descripción
Valor de Negocio Primario	Reducción del tiempo de la toma de decisiones analíticas y aumento de la eficiencia en la curación de playlists al pasar de una exploración manual a una vista consolidada y predictiva.
Valor de Negocio Secundario	Mejora en la retención de usuarios o engagement al ofrecer un sistema de descubrimiento musical más justo y diverso (mitigación del sesgo).

5.2 KPIs de éxito (SMART).

KPI	Métrica	Criterio de Éxito
Eficiencia Operacional (Dashboard)	Tiempo promedio de análisis de tendencias.	Reducción del 25% del tiempo de análisis vs. Exploración manual de API
Redimiento (API Recomendación)	Latencia de la API de similitud	Latencia media de la respuesta ≤ 500 ms.
Calidad del modelo (MLflow/Ética)	Tasa de aceptación de las 5 recomendaciones principales.	Tasa de aceptación ≥ 70% en pruebas de usuario.

6. Sección ética, legal y accesibilidad.

El diseño y desarrollo de este proyecto se regirá por un marco ético y legal estricto, priorizando la privacidad de los datos, la equidad algorítmica y la accesibilidad de la solución.

6.1 Análisis Legal y de Privacidad (PII/GDPR)

En la Fase 1, se ha determinado que el proyecto se basará exclusivamente en el **catálogo musical público** de Spotify, que incluye metadatos de canciones como título, artista, género y características acústicas. Por lo tanto, el riesgo de procesamiento de **Datos Personales Identificables (PII)** de usuarios finales es nulo o muy bajo. No se recopilarán datos sensibles ni historiales de escucha individualizados.

Como el tratamiento se limita a datos públicos, las bases legítimas de tratamiento bajo **GDPR** son simples y se centran en el **Interés Legítimo** para el desarrollo de investigación y análisis de datos en un contexto académico. No se requiere un plan riguroso de anonimización en esta fase, pero si en etapas posteriores se integrasen logs de prueba de usuarios, se aplicaría la **pseudonimización** mediante la asignación de identificadores de sesión aleatorios, asegurando la desvinculación de cualquier dato real de usuario.

6.2 Gestión de Sesgos Algorítmicos

El riesgo ético principal reside en el **sesgo inherente** al propio dataset musical, que podría llevar al motor de recomendación a reforzar el sesgo de popularidad o a mostrar una baja diversidad. Abordaremos la **gestión de sesgos (género, edad, origen)** de la siguiente manera:

☐ 6.2.1 Auditoria de sesgos.

Se incluirá un hito especifico para auditar la distribución de las características acústicas por variable potencialmente sensibles como el género musical, el origen geográfico y la época (vinculada a la edad).

☐ 6.2.2 Mitigación del modelo.

Dado que el motor de recomendación se basa en la similitud de características objetivas (bailaibilidad, energía, valence), se minimiza la dependencia directa de la popularidad.

☐ 6.2.3 Factor de diversidad.

Se propone la implementación de un mecanimo de **post-procesamiento** que aplicará un factor de diversidad a la lista final de recomendaciones para asegurar que el algoritmo no discrimine inadvertidamente a artistas de baja frecuencia o géneros minoritarios.

6.3 Accesibilidad universal (WCAG 2.1 AA)

Se garantiza que el **Dashboard analítico** cumplirá con los críterios de accesibilidad del estándar **WCAG 2.1 Nivel AA**. Esto implica tomar las siguientes medidas:

- **Perceptibilidad:** Asegurar un **contraste de color adecuado** para todo el texto y los elementos gráficos. Todos los gráficos complejos incluirán **etiquetas *alt*** o descripciones textuales para ser interpretados por lectores de pantalla.
- **Operabilidad:** La navegación completa del Dashboard será posible **mediante teclado** y sin depender de un ratón, esencial para usuarios con discapacidades motoras.

6.4 Medidas de Seguridad de Datos.

Las medidas de seguridad se diseñarán para proteger la integridad y disponibilidad de la infraestructura de datos y del modelo de IA:

- **Seguridad en Tránsito y Reposo:** Todas las comunicaciones del pipeline de ingesta (API) y la API de recomendación se realizarán utilizando **protocolos seguros (HTTPS/TLS)**. Los datos almacenados en la base de datos se protegerán con **cifrado en reposo**.

- **Control de Acceso:** Se implementará un estricto **control de acceso basado en roles y permisos** para el acceso a las herramientas clave (Spark, MLflow y la base de datos), y las claves de acceso se gestionarán de forma segura.

- **Seguridad del Código:** El repositorio de código será **privado** y se prohibirá estrictamente la subida de cualquier tipo de PII real o claves de acceso sensibles al mismo.

7. Definición de datos.

7.1 Fuentes candidatas .

Para este proyecto de análisis de los datos de una cuenta de Spotify se ha decidido la utilización de datos privados como es la propia cuenta de Spotify que requiere una autorización del usuario e inicio de sesión de este, que son extraídos de la propia API que proporciona la plataforma, para entrenar un modelo de recomendación usando ML se ha decidido usar un datasets público usando la herramienta KAGGLE.

7.2 Licencia

Se van a comentar las diferentes licencias y reglas que se van a seguir en la realización de este proyecto.

- **Spotify Developer Terms**

En estos términos de manera resumida vienen incluidos varios términos sobre la protección de datos, el no almacenamiento ni distribución de estos a través de una venta o la propia distribución de datos personales que puedan vulnerar la privacidad de las personas, siendo claros con la finalidad de los datos mostrando transparencia.

- **GDPR (Reglamento General de Protección de Datos)**

En el reglamento europeo se recoge que la protección de datos personales es un derecho fundamental.

7.3 Variables objetivo/características.

- **Variables objetivo**

Se va a detallar las variables objetivas del sistema predictivo que se tiene pensado añadir, también se resumirá de forma breve cada variable:

Tipo	**Variable objetivo**	**Descripción**	**Ejemplo**
Implícita	played_full	Se usa para saber si el usuario ha escuchado más de cierto porcentaje de la canción	1
Explícita	user_rating	Es usada para almacenar el rating de la canción por parte del usuario	4
Ponderada	relation	relación entre el tiempo de escucha y los likes	0.90
Probabilidad	ranking	La probabilidad que se usara para saber si una canción gustara al usuario o no	0.75

- **Variables características**

Para entrenar el modelo se usarán los datos del usuario, como pueden ser:

Tipo	**Variable** característica	**Descripción**
String	user_id	Primero se hará un Hash sobre el id del usuario y este servirá para identificarlo en caso de querer guardar los datasets creados.
Int	user_age	Almacena la edad del usuario
Ponderada	favorite_genres	Los géneros que más ha escuchado, dado like o guardado. Esta variable es esencial para la predicción

7.4 Volumen esperado

Al ser un proyecto para análisis de la propia cuenta del usuario no esperando recopilar mas de uno por vez, el volumen es muy bajo debido a que se esperaría que una sola persona pueda reproducir como máximo en un mes no más de 200 canciones distintas, si contamos las repetidas no se cree que pueda llegar a 800

8. Plan de Trabajo (Gantt y estimación por semanas)

Semana	Objetivo principal	Tareas clave	Responsable	Criterios de finalización (DoD)
1	Configurarel proyecto y entorno	• Crear el repositorio y estructura de carpetas. • Preparar entorno virtual y dependencias. • Cargar dataset inicial de prueba (Kaggle/CSV).	Héctor	Proyecto ejecutable en menos de 10 minutos.Repositorio organizado y limpio
2	Conectar con la API de Spotify e ingestar datos	• Configurar acceso a la API (autenticación segura). • Descargar canciones y artistas de ejemplo. • Guardar los datos en formato Parquet.	Héctor	Datos cargados correctamente y sin errores.Estructura de carpetas lista para el análisis.
3	Limpiar y preparar los datos	• Eliminar duplicados y valores nulos .• Normalizar variables (energía, bailabilidad, valence, etc.). • Generar dataset final para el modelo	Ignacio	Datos limpios y listos para entrenamiento.Cobertura de variables ≥ 98 %

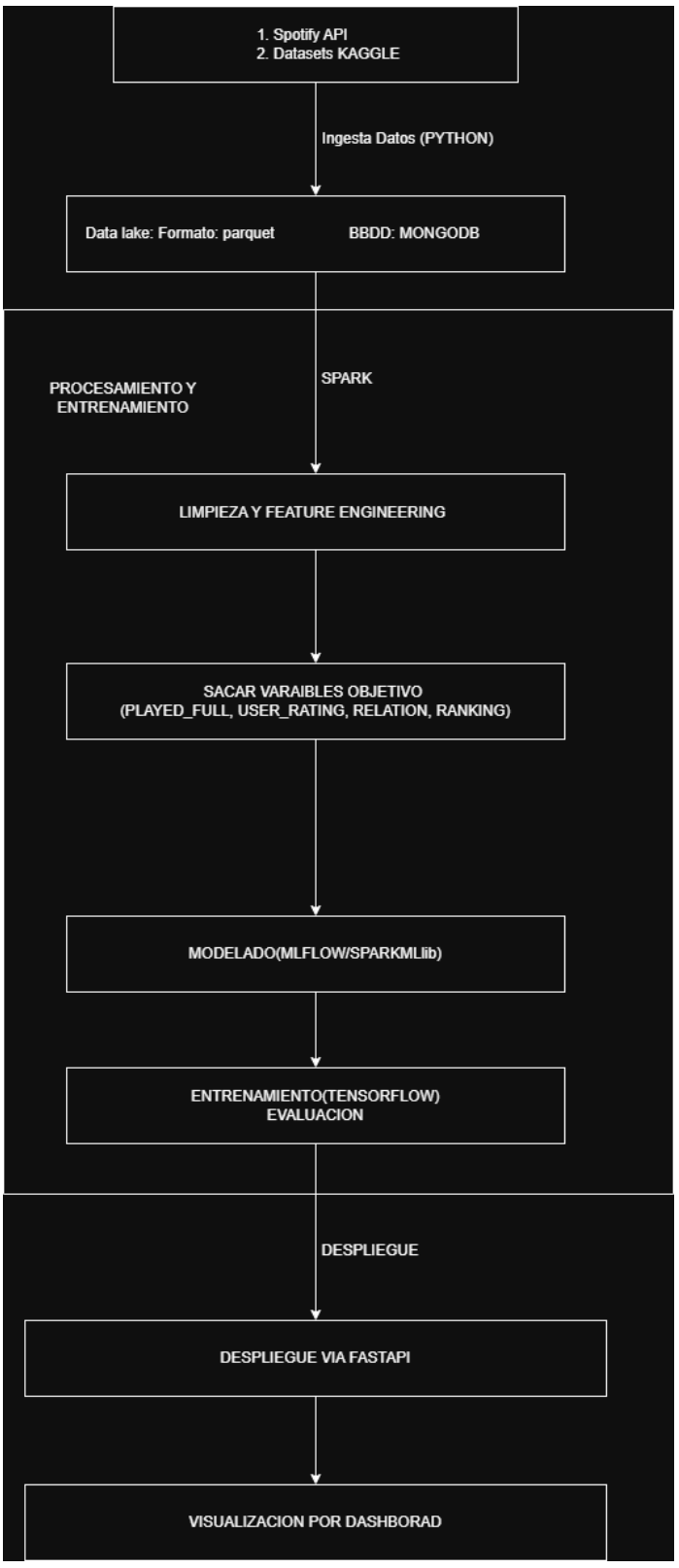
4	Entrenar el modelo de recom	• Entrenar un modelo de similitud acústica (kNN o similar). • Probar distintos	Ignacio	El modelo devuelve al menos 10 canciones similares por entrada.Precisión ≥ 25 %
5	Diseñar el dashboard analítico	• Crear panel visual (Streamlit o Power BI) .• Mostrar tendencias, métricas y recomendaciones .• Asegurar accesibilidad (WCAG 2.1 AA)	Francisco	Dashboard funcional y accesible.Tiempos de carga < 10 segundos
6	Validar resultados y documentar	• Probar el sistema completo (ingesta–modelo–dashboard • Medir aceptación de recomendaciones (≥ 70 %). • Revisar seguridad, privacidad y elaborar documentación final.	Todos	Demo funcional.Resultados alineados con los KPIs.Documentación lista para entrega.

8.1 Roles y responsabilidades del equipo

Integrantes	**Rol Principal**	**Responsabilidades clave**
Hector Fernandez Pineda	Data Engineer	• Configuración del entorno y repositorio. • Conexión e ingesta de datos desde la API de Spotify. • Procesamiento y almacenamiento en formato Parquet. • Automatización y control de calidad de los datos.
Ignacio López García	Data Scientist / Machine Learning Engineer	• Limpieza y preparación de datos para modelado. • Entrenamiento del modelo de recomendación (similitud acústica). • Evaluación del modelo y mitigación de sesgos. • Registro y seguimiento de experimentos con MLflow.
Francisco José Puertas Teba	Data Analyst / Dashboard Developer	• Diseño e implementación del dashboard interactivo (Streamlit o Power BI). • Visualización de métricas, tendencias y recomendaciones. • Aplicación de criterios de accesibilidad (WCAG 2.1 AA). • Validación final de resultados y documentación.

9. Arquitectura de alto nivel

9.1.Esquema



9.2.Descripción Componentes.

Se ha dividido el proyecto en cuatro puntos esenciales.

1. **Ingesta de datos.**
    - **Spotify web API**. Es la fuente de datos primaria que se usará para mostrar los datos del usuario que use la aplicación.
    - **DATASETS**. Es una fuente secundaria que usaremos para el entrenamiento de un modelo de recomendación.
    - **Almacenamiento**. Se usa MongoDB para el almacenamiento de los datos y cuando se necesite guardar los datasets se usará parquet como formato de los archivos.
2. **Procesamiento y modelado.**
    - **Apache Spark**. Es la herramienta que se usará para el Feature Engineering como la limpieza de los datos, cálculos necesarios como medias de rating, etc…
    - **MLFLOW.** Se usará para el versionado del modelo, rastreo de hiperparametros y ayuda en la construcción del modelo.
3. **Despliegue.**
    - **FASTAPI.** La usaremos para la comunicación del dashboard y los datos y el modelo.