

10 DE ABRIL DE 2025

Autoras: JORGE MARTÍN GARCÍA, SARA DÍAZ CASAMAYOR, CORAL CAMACHO DAPENA.

Tutor: MIKOLA SOCHINSKYI.

Ciclo: DESARROLLO DE APLICACIONES WEB.

Centro: I.E.S ENRIQUE TIERNO GALVÁN.



Índice:

Índice:	2
1. INTRODUCCIÓN	3
1.1. Descripción	3
1.2. Motivación	3
2. GESTIÓN Y ORGANIZACIÓN DEL PROYECTO	4
2.1. Metodología del desarrollo	4
2.1. Planificación	5
2.1. Ciclo de vida	5
3. OBJETIVOS	5
4. DESARROLLO DE LA APLICACIÓN	5
4.1. Análisis: requisitos	5
4.2. Diseño: arquitectura	6
4.3. Implementación: decisiones adoptadas y problemas encontrados	6
5. REALIZACIÓN DE PRUEBAS	6
5.1. Descripción de las pruebas	6
5.2. Resultados de las pruebas	7
6. CONCLUSIONES	7
7. TRABAJO FUTURO	7
8. BIBLIOGRAFÍA	7

1. INTRODUCCIÓN

1.1. Descripción

Qué se quiere hacer:

El proyecto consiste en el desarrollo de un **reproductor de música en línea** que permita a los usuarios cargar, organizar y reproducir archivos de música **de manera privada y organizada**. Se busca ofrecer una experiencia de usuario mejorada en comparación con reproductores básicos disponibles actualmente.

Objetivos:

El objetivo principal del proyecto es desarrollar una plataforma web que permita a los usuarios gestionar sus archivos de música de manera más eficiente y funcional. Los objetivos específicos incluyen:

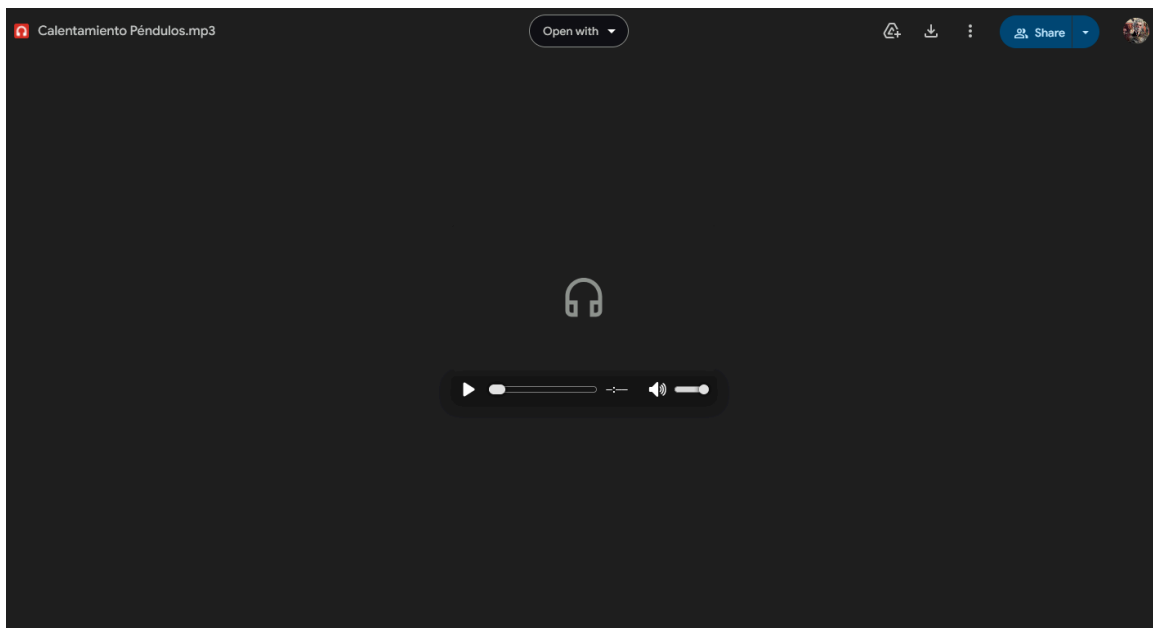
- Facilitar la carga y organización de archivos de música, permitiendo a los usuarios **crear listas de reproducción y organizar sus canciones** en carpetas personalizadas.
- **Mejorar la experiencia de escucha**, ofreciendo características como la reproducción en segundo plano, la repetición de canciones y la creación de listas de reproducción.
- **Garantizar una plataforma segura y privada** para compartir archivos de música entre usuarios, asegurando que los archivos no sean accesibles públicamente sin el consentimiento adecuado.

-
- **Crear un acceso multiplataforma**, permitiendo a los usuarios escucharla desde cualquier dispositivo a través de una aplicación web accesible desde navegadores modernos.

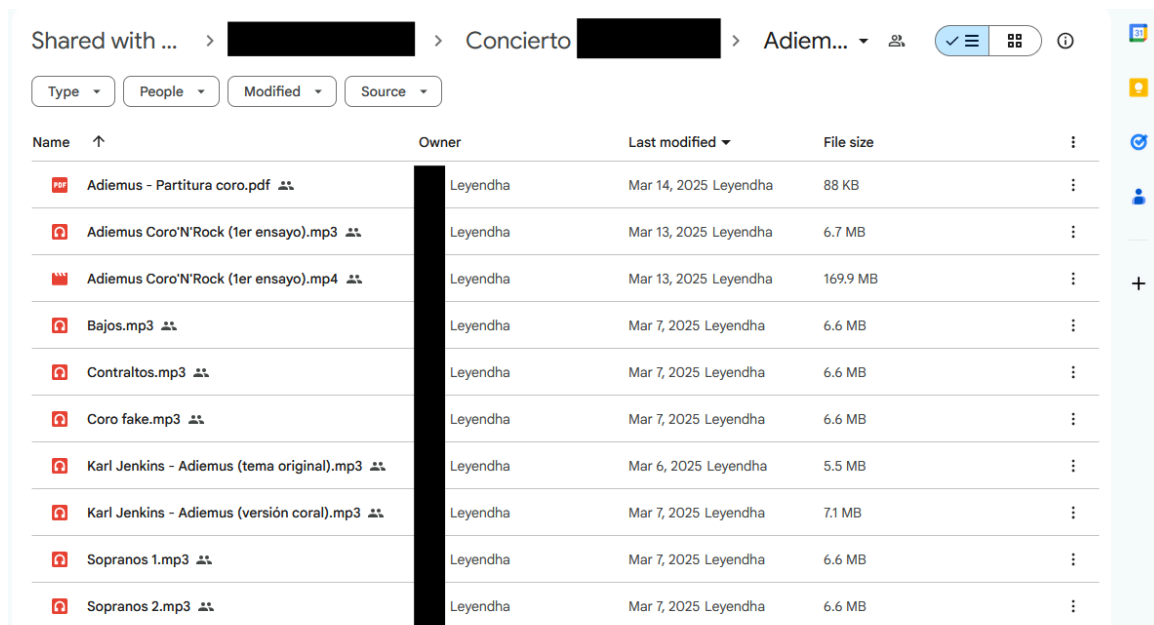
1.2. Motivación

Fundamentación:

La motivación **nace de la experiencia de usuario al utilizar Google Drive** para compartir archivos de música privados. Aunque Google Drive permite compartir archivos de manera eficaz, **su reproductor integrado presenta limitaciones importantes**, como la imposibilidad de reproducir música en segundo plano, la falta de opciones para organizar archivos en listas de reproducción y la ausencia de funcionalidades como la repetición de canciones. **Estas restricciones son especialmente problemáticas para grupos musicales** y otros usuarios que necesitan organizar sus archivos de manera específica, sin perder la estructura de carpetas que ofrece Google Drive. A través de este proyecto, se busca dar respuesta a esta problemática, ofreciendo una solución más completa y adecuada a las necesidades de estos usuarios.



Se podría pensar que una solución lógica es **bajar los temas a un teléfono móvil** o a un ordenador y utilizar otra aplicación para reproducirlos. Podría hacerse, pero en ese caso **se perdería la organización por carpetas que tiene Google Drive**. Dicha organización es muy útil para grupos musicales que tienen archivos por cuerdas (grupos de personas que utilizan un mismo archivo: tenores, contraltos, sopranos...), donde es fundamental que todo esté en su sitio.



Shared with ...		>		>	Concierto		>	Adiem...		✓	☰	ⓘ	
Type	People	Modified	Source										
Name	↑	Owner		Last modified		File size							
Adiemus - Partitura coro.pdf		Leyendha		Mar 14, 2025		88 KB							
Adiemus Coro'N'Rock (1er ensayo).mp3		Leyendha		Mar 13, 2025		6.7 MB							
Adiemus Coro'N'Rock (1er ensayo).mp4		Leyendha		Mar 13, 2025		169.9 MB							
Bajos.mp3		Leyendha		Mar 7, 2025		6.6 MB							
Contraltos.mp3		Leyendha		Mar 7, 2025		6.6 MB							
Coro fake.mp3		Leyendha		Mar 7, 2025		6.6 MB							
Karl Jenkins - Adiemus (tema original).mp3		Leyendha		Mar 6, 2025		5.5 MB							
Karl Jenkins - Adiemus (versión coral).mp3		Leyendha		Mar 7, 2025		7.1 MB							
Sopranos 1.mp3		Leyendha		Mar 7, 2025		6.6 MB							
Sopranos 2.mp3		Leyendha		Mar 7, 2025		6.6 MB							

Ejemplo de archivos por cuerda (con mismo nombre en distintas carpetas)

Destinatarios o beneficiarios:

Este proyecto está dirigido a **usuarios que necesitan compartir y organizar archivos de música privados** de manera eficiente. Los principales beneficiarios de la aplicación incluyen:

-
- **Grupos musicales y artistas** que requieren una herramienta para organizar archivos musicales por categorías (como cuerdas, voces o instrumentos) y compartirlos de forma privada con sus colaboradores.
 - **Profesionales de la música (como productores y compositores)** que buscan un reproductor más avanzado que los servicios de almacenamiento convencionales, que no ofrecen las funcionalidades específicas que se necesitan.
 - **Usuarios generales que desean gestionar su biblioteca musical privada** de forma eficiente, sin recurrir a herramientas de almacenamiento tradicionales.

2. GESTIÓN Y ORGANIZACIÓN DEL PROYECTO

2.1. Metodología del desarrollo

El desarrollo de este proyecto ha seguido una metodología iterativa e incremental, centrada en el **desarrollo ágil** de la aplicación, utilizando los principios del [manifiesto Agile](#). Este enfoque se caracteriza por construir el proyecto de manera escalonada, comenzando con las funcionalidades básicas y luego escalando características adicionales según **las iteraciones de feedback con el cliente**. Se ha elegido esta metodología por ser extensamente utilizada en el State of the Art (**SotA**) actual.

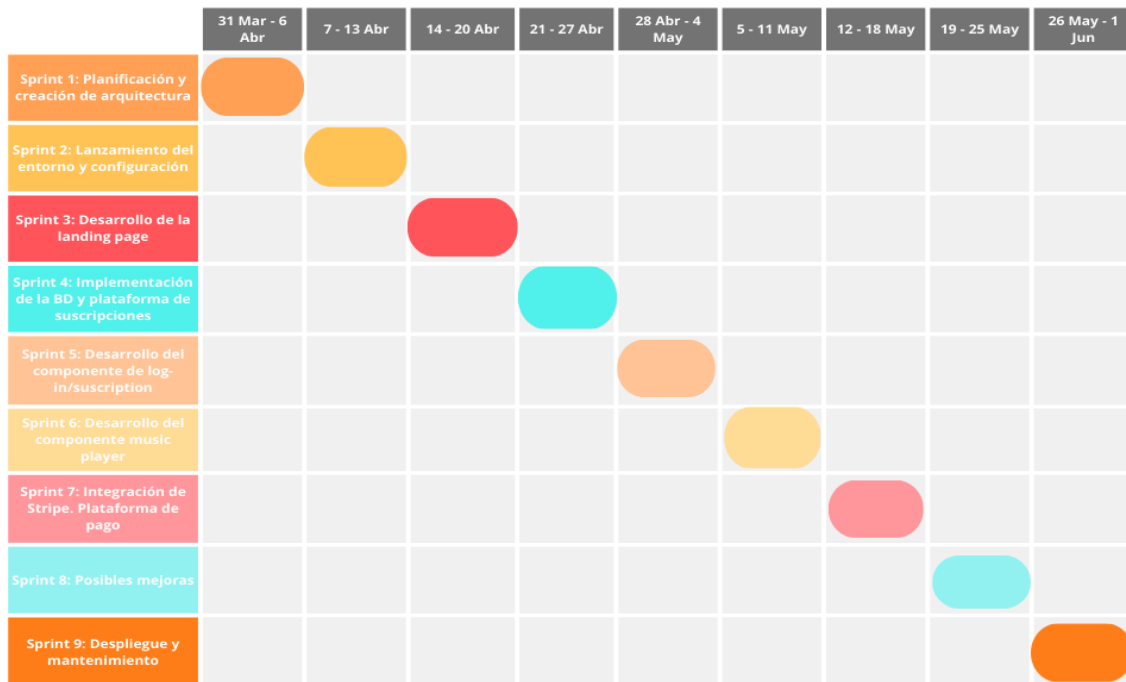
2.2. Planificación

La planificación inicial del proyecto se ha basado en dividir el desarrollo en sprints cortos, alineados con el enfoque ágil, y se estimó el tiempo para cada fase del

desarrollo de acuerdo con las funcionalidades que se querían implementar. A continuación se muestra un esquema con las especificaciones:

Los Sprints se han establecido en una semana. El sprint planning se establece el lunes y el sprint review el viernes.

- Sprint 1: Planificación y creación de arquitectura. - [31 Mar - 6 Abr]
- Sprint 2: Lanzamiento del entorno y configuración. [7 - 13 Abr]
- Sprint 3: Desarrollo de la *landing page*. [14 - 20 Abr]
- Sprint 4: Implementación de la BD y plataforma de suscripciones. [21 - 27 Abr]
- Sprint 5: Desarrollo del componente de *log-in/subscription*. [28 Abr - 4 May]
- Sprint 6: Desarrollo del componente *music player*. [5 - 11 May]
- Sprint 7: Integración de Stripe. Plataforma de pago. [12 - 18 May]
- Sprint 8: Posibles mejoras. [19 - 25 May]
- Sprint 9: Despliegue y mantenimiento. [26 May - 1 Jun]



2.3. Ciclo de vida

(...A rellenar en próximas entregas...)

3. OBJETIVOS

En la introducción del proyecto se han establecido los siguientes **objetivos generales** para el reproductor de música web que se va a desarrollar:

1. Crear un reproductor de música en línea con capacidad de reproducción en segundo plano.
2. Desarrollar una interfaz moderna y responsiva.
3. Brindar una experiencia de reproducción fluida.
4. Accesibilidad y disponibilidad.
5. Interacción con otros usuarios.
6. Calidad y optimización del audio.
7. Seguridad y privacidad.

Los **objetivos específicos** de la aplicación son:

1. Permitir la organización de canciones en listas de reproducción personalizadas.
2. Incorporar la opción de repetir canciones en bucle.
3. Implementar un sistema de registro y log-in.
4. Incorporar la reproducción en segundo plano.
5. Añadir la opción de compartir listas de reproducción.
6. Añadir la opción de añadir a pistas distintas voces.
7. Permitir la organización de listas en distintas carpetas.

4. DESARROLLO DE LA APLICACIÓN

4.1. Análisis: requisitos

El proyecto se basa en los siguientes requisitos funcionales y no funcionales:

Requisitos funcionales:

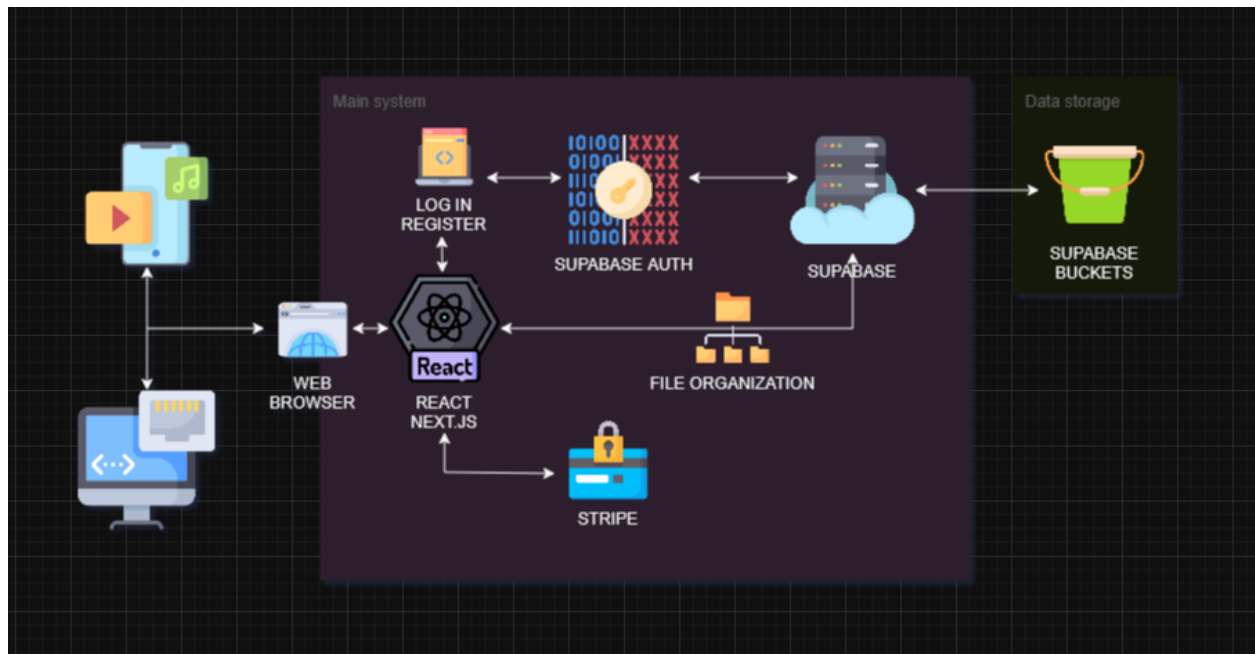
1. **Reproducción de música:** La aplicación debe permitir a los usuarios reproducir canciones en diferentes formatos, como archivos .mp3.
2. **Reproducción en segundo plano:** El reproductor debe ser capaz de seguir funcionando incluso cuando el usuario navegue a otras partes de la aplicación.
3. **Listas de reproducción personalizadas:** Los usuarios deben poder crear, editar y gestionar sus propias listas de reproducción.
4. **Repetición de canciones:** Los usuarios deben poder activar una opción para reproducir canciones en bucle.
5. **Carga de canciones:** El sistema debe permitir la carga de archivos de música y almacenarlos de manera organizada.
6. **Gestión de cuentas de usuario:** Los usuarios deben poder registrarse, iniciar sesión y gestionar su cuenta de manera segura.
7. **Interfaz intuitiva y responsiva:** La aplicación debe ser fácil de usar y adaptarse a diferentes tamaños de pantalla.

Requisitos no funcionales:

1. **Rendimiento:** La aplicación debe cargar rápidamente, incluso con una gran cantidad de canciones almacenadas.
2. **Escalabilidad:** El sistema debe poder manejar un número creciente de usuarios y archivos sin perder rendimiento.
3. **Modularidad:** El sistema debe tener facilidad a la hora de añadir nuevos módulos funcionales.

-
4. **Seguridad:** La información personal y los datos de los usuarios deben estar protegidos adecuadamente.

4.2. Diseño: arquitectura



Arquitectura principal:

1. **Frontend:** Se ha utilizado React en combinación con Next.js para desarrollar una interfaz dinámica y fluida. Next.js permite la renderización del lado del

servidor (SSR) y la creación de rutas, lo que mejora la velocidad de carga y el SEO. Tailwind CSS se ha usado para estilizar la interfaz, logrando un diseño atractivo y responsivo que se adapta a diferentes dispositivos.

2. **Backend:** El back-end Supabase maneja tanto la base de datos como la autenticación, permitiendo a los usuarios registrarse y gestionar su cuenta de manera segura. Se accede a través de peticiones HTTP (API).
3. **Base de datos:** se gestiona mediante Supabase, que proporciona una solución robusta y escalable basada en PostgreSQL.
4. **Autenticación:** Para la autenticación de usuarios, se ha implementado un sistema basado en Supabase Auth, que permite el registro, inicio de sesión y autenticación mediante correo electrónico y plataformas externas como GitHub.
5. **Plataforma de pago:** Stripe se ha integrado para gestionar las suscripciones premium de los usuarios, lo que les permite acceder a características adicionales, como la posibilidad de escuchar música sin anuncios y la opción de cargar canciones sin restricciones.

4.3. Implementación: decisiones adoptadas y problemas encontrados

(...A rellenar en próximas entregas...)

5. REALIZACIÓN DE PRUEBAS

5.1. Descripción de las pruebas

(...A rellenar en próximas entregas...)

5.2. Resultados de las pruebas

(...A rellenar en próximas entregas...)

6. CONCLUSIONES

(...A rellenar en próximas entregas...)

7. TRABAJO FUTURO

(...A rellenar en próximas entregas...)

8. BIBLIOGRAFÍA

(...A rellenar en próximas entregas...)