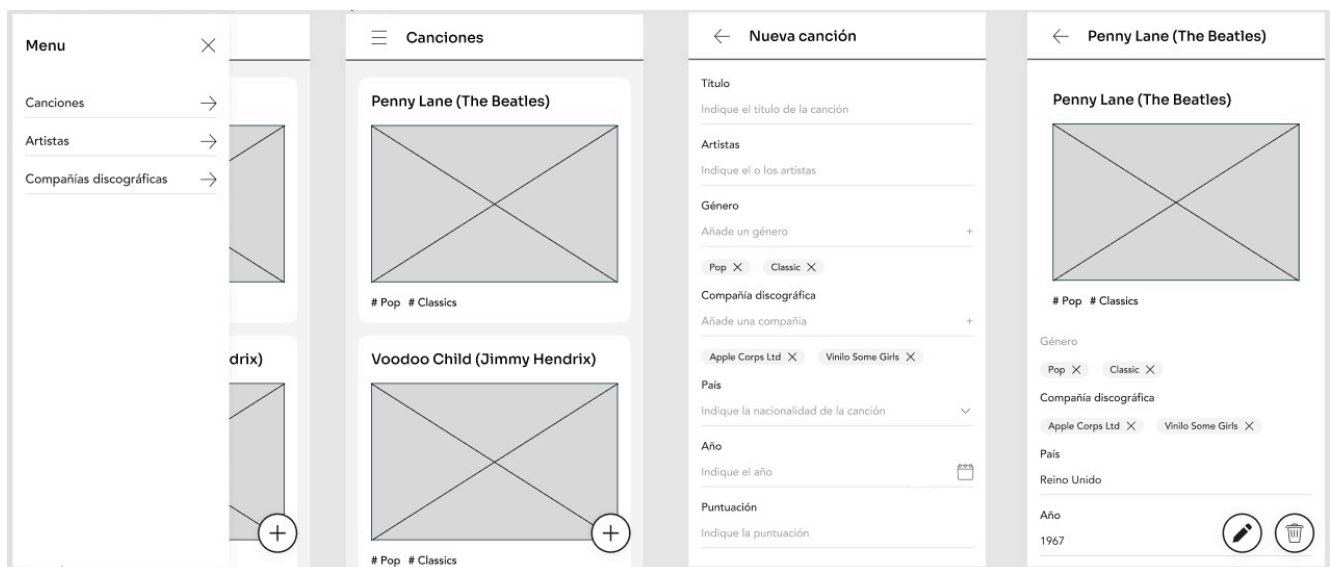


\*\* Este test incluye únicamente la parte Frontend de una aplicación básica. La parte backend se discutirá durante la entrevista y feedback del trabajo entregado.

El desafío consiste en crear una aplicación web compacta, cuyo propósito es acceder y gestionar nuestra colección de datos sobre Canciones, Artistas y Compañías Discográficas. Te encargarás de implementar la funcionalidad de “Canciones”, la cual es la más destacada entre las tres mencionadas anteriormente. El equipo de negocio nos ha facilitado un esquema inicial (wireframes) y nos han dado completa autonomía para definir el diseño.



## Instrucciones preliminares

- Emplea un repositorio GIT abierto para documentar tu progreso. Valoraremos enormemente tu metodología de trabajo (Commits, Branches, Pull Request, etc). Parte de nuestro análisis se centra en tu enfoque para resolver problemas, y la única manera de observarlo es a través de tus commits.
- Se te proporciona un archivo “mock.json” con el cual debes iniciar un servicio REST local utilizando “json-server” (<https://www.npmjs.com/package/json-server>) u otro que creas conveniente bajo el mismo fin.
- El inicio del servicio del json-server debe estar como un script del package.json.

## Requisitos obligatorios

- Estás libre de diseñar la estructura del proyecto como mejor te parezca. No nos preocupa el exceso de planificación inicial, entendiendo que el proyecto se ampliará pronto.
- Aplica las técnicas más avanzadas de desarrollo y rendimiento que conozcas.
- Se te otorga plena libertad en el diseño, y puedes elegir utilizar una biblioteca de componentes, pero asegúrate de que el diseño sea adaptable (el wireframe proporcionado muestra únicamente la vista para móviles).
- Implementa soporte para varios idiomas utilizando ngx-translate o @ngneat/transloco. **Solo en español.**
- Es necesario gestionar los estados de carga (Spinners, Skeleton Loader, Progress Bar, etc.).
- Gestiona los datos usando Manejo de Estado (State Management). Puedes optar por Redux con NgRx, BehaviorSubjects, signal store o cualquier otra herramienta con la que te sientas a gusto y más cómodo.

## Aspectos adicionales (opcionales / valorables)

- Realiza pruebas unitarias e integración con Jest o con Karma+Jasmine. Pruebas e2e con Playwright, Cypress, etc.
- Emplea una metodología CSS (BEM, Utility [TailwindCSS], etc.).