



Estudiantes: Jared Mijail Ramirez Escalan, Nicolas Rodriguez Piraban
Daniela Ariadna Rueda Hernand, Silvana Suarez Carvajal

- El lenguaje de programación y framework seleccionado realizando un comparativo de ventajas y desventajas con otras tecnologías

Se consideraron los frameworks React, Vue.js y Flutter **[Anexo 1]** para el desarrollo del proyecto. Se escogió Vue.js principalmente por la curva de aprendizaje teniendo en cuenta el tiempo con el que disponemos para finalizar el proyecto, además de su flexibilidad.

- La base de datos relacional que se utilizará (la tecnología)

La base de datos relacional se realizará en SQLite, esto dado que la aplicación no está diseñada para escalar en múltiples servidores o de forma horizontal, el tráfico que este requiere es mínimo y la capacidad de almacenamiento de SQLite excede por mucho lo que la aplicación podría llegar a usar. Además esta base de datos no tiene la complejidad de tener un cliente servidor, o sea no es necesario tener un contenedor o algún tipo de servidor para usar la base de datos, sino que todo se guarda en un único archivo.

Para el manejo de la base de datos utilizaremos un ORM, ya que este da la facilidad de no tener que escribir el programa para algún dialecto específico de sql, previene sql injections, y se obtienen demás ventajas introducidas por los ORM. Específicamente utilizaremos la librería sequelize por su facilidad de uso.

Se usará la herramienta dbdiagram.io para la realización del diagrama de la base de datos.

- Bibliotecas y herramientas complementarias planeadas para su desarrollo de proyecto

Dado que se requiere realizar una aplicación de escritorio, se hace uso de Electron, framework que permite crear aplicaciones de escritorio usando html, css y javascript. Además de esto, se usa tailwind, que es un framework utilizado para el frontend de la aplicación.

Para el administrador de paquetes usaremos bun, que es más rápido que el runtime por defecto npm, y tiene mejor resolución de conflictos, mejorando la experiencia de desarrollo.

También para los test estáticos del proyecto utilizaremos lint.

Anexos

1. Tabla generada por IA - Comparación React - Vue.js - Flutter

Característica	React	Vue.js	Flutter
Tipo de framework	Biblioteca JS para interfaces de usuario	Framework progresivo de JavaScript	SDK completo para desarrollo de interfaces
Lenguaje principal	JavaScript / TypeScript	JavaScript / TypeScript	Dart
Plataforma principal	Web	Web	Móvil (Android, iOS), Web, Escritorio
Curva de aprendizaje	Moderada	Suave	Empinada (por Dart y estructura propia)
Desempeño (performance)	Alto (nativo con React Native)	Alto (aunque algo menor que React)	Muy alto (compila a código nativo)
Ecosistema	Muy amplio (React Native, Next.js, etc.)	En crecimiento, más usado en Asia	Robusto, controlado por Google
Desarrollo móvil	Sí (React Native)	Limitado (usando wrappers como NativeScript o Capacitor)	Sí (principal foco de Flutter)
Reutilización de código	Alta entre web y móvil	Media, depende de herramientas externas	Muy alta entre plataformas
Documentación y comunidad	Excelente, muy madura	Clara y accesible, ideal para principiantes	Muy buena, creciendo rápido

Flexibilidad	Muy flexible, sin imposiciones	Muy flexible, con estructura opcional	Más estructurado (convención sobre configuración)
Empresas que lo usan	Facebook, Instagram, Airbnb	Alibaba, GitLab	Xiaomi, Google, BMW, Alibaba, eBay
Actualizaciones	Frecuentes pero estables	Frecuentes y bien gestionadas	Muy activas, dirigidas por Google