

TECNOLOGÍAS EMPLEADAS

Spring Boot se utilizó como framework de backend debido a su capacidad para desarrollar aplicaciones Java de manera rápida y sencilla, proporcionando características como la inversión de control y la inyección de dependencias. Además, Spring Boot ofrece una amplia gama de bibliotecas y módulos que facilitan la construcción de API RESTful y la integración con bases de datos.

Angular se seleccionó como framework de frontend debido a su enfoque en la creación de aplicaciones web de una sola página (Single Page Applications) y su arquitectura basada en componentes, facilitando la gestión del estado de la aplicación y la comunicación con el backend a través de peticiones HTTP.

MySQL se eligió como sistema de gestión de base de datos se basó en su estabilidad, siendo una de datos relacional ofrece un buen rendimiento y se integra bien con las tecnologías utilizadas en este proyecto.

Git se utilizó como sistema de control de versiones para facilitar la colaboración y el seguimiento de los cambios en el código fuente. Git permite mantener un historial de versiones, realizar fusiones de código de manera eficiente y gestionar ramas de desarrollo.

PATRONES DE DISEÑO

El patrón DTO se utilizó para definir objetos de transferencia de datos que encapsulan la información necesaria para las operaciones de comunicación entre las capas del sistema. Al utilizar DTOs, se logra una separación clara entre los datos de la capa de presentación y los datos de la capa de negocio, lo que facilita la serialización, la validación y el intercambio de datos entre diferentes componentes.

El patrón Mapper se empleó para establecer una transformación controlada y coherente entre los objetos DTO y los objetos de dominio del sistema. El Mapper se encarga de convertir los DTOs en objetos de dominio y viceversa, permitiendo que los diferentes componentes trabajen con los objetos adecuados según su contexto.

El patrón Builder se utilizó para la construcción flexible y escalable de objetos complejos. Al utilizar un Builder mediante el uso de Lombok, separamos la lógica de construcción del objeto de su representación final, esto facilita la creación de objetos y mejorar la legibilidad.