

The background features abstract geometric shapes in shades of green and blue. In the top left, there are green 'x' marks and a red '+' sign. In the bottom right, there are more green 'x' marks and a red circle. The main title is centered in a large, bold, dark grey font.

Sistema de vantagens

Grupo: Ana Carolina e João Pedro

Front-end



Front-end

Vue

- Framework JavaScript progressivo para construção de interfaces de usuário.
- Fácil de integrar em projetos já existente
- Reativo, componentizado e com foco em view model

Vite

- Ferramenta de construção moderna e otimizada para o desenvolvimento de aplicações web
- Build Otimizado: Agrupamento de arquivos de forma eficiente
- Suporte nativo a ES Modules: Melhor desempenho em navegadores modernos.

Back-end



Express

Back-end

Node

- Ambiente de execução JavaScript orientado a eventos e assíncrono.
- NPM (Node Package Manager): Gerenciador de pacotes integrado, que oferece uma vasta coleção de bibliotecas e ferramentas.

Express

- Framework web minimalista e flexível para Node.js, que facilita o desenvolvimento de APIs e servidores web.
- Roteamento Simplificado: Configuração e gerenciamento de rotas de forma direta, possibilitando organização da estrutura de rotas

Arquitetura do sistema

O sistema seguiu o padrão MVC porém com algumas alterações comumente utilizadas no mercado para melhorar a estrutura do código para determinadas plataformas.

O repositório foi estruturado com as seguintes camadas:

- Controller
- Entity
- Interfaces
- Repository
- Services

Camada de persistência

A camada de persistência do projeto, é composta por 4 principais entidades

- Entity: Contém os modelos que representam as tabelas do banco de dados
- Repository: Responsáveis por encapsular a lógica de acesso aos dados
- Connection: A conexão com o banco de dados remoto é realizada atrás da configuração do arquivo data source
- Migration: Contém as migrações realizadas e a serem realizadas, abstraindo a lógica de interação com o banco de dados

TypeORM

O ORM (Object-Relational Mapping) é uma técnica que permite o desenvolvedor acessar um banco de dados relacional através de uma linguagem orientada a objeto, abstraindo a comunicação entre os dois. Ele mapeia os objetos para tabelas relacionais.

Já o TypeORM é a implementação dessa estratégia para projetos em Node.js.

Estratégias de acesso

- Abstrair a comunicação com a camada de persistência, facilitando uma possível troca de banco de dados
- Interação com o banco de dados mais simples e menos sujeita a erros
- Escalável
- Migração de dados simples, garantindo maior consistência entre o código e a base de dados real