

Carlos Figueiredo

Gabriela Alvarenga

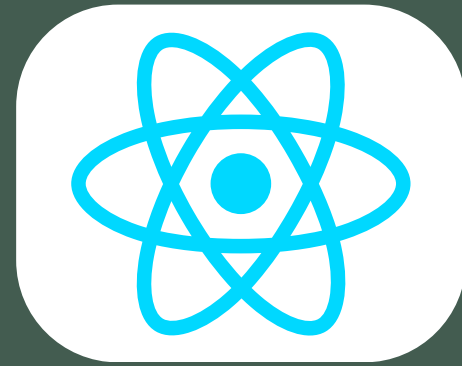
Luísa Jardim

SISTEMA DE MOEDA ESTUDANTIL



TECNOLOGIAS - FRONTEND

REACT



- Componentização: permite dividir a interface em partes reutilizáveis, facilitando manutenção e organização do código.
- Virtual DOM: melhora o desempenho das atualizações na interface.
- Grande comunidade: garante ampla documentação, suporte e diversas bibliotecas complementares.

VITE



- Build extremamente rápido: utiliza ESBuild e Rollup, otimizando tempo de compilação.
- Hot Module Replacement (HMR): atualiza a página em tempo real sem recarregar tudo.
- Configuração simples: projeto pronto para rodar em poucos comandos.
- Suporte nativo a TypeScript e JSX: integração fluida com React.
- Servidor de desenvolvimento leve e eficiente.

VANTAGENS DESSA COMBINAÇÃO

- Inicialização instantânea de projetos React.
- Ambiente moderno e otimizado para desenvolvedores.
- Menor tempo de espera entre alterações e visualização.
- Melhor desempenho no desenvolvimento e no build final.



TECNOLOGIAS – BACKEND

JAVA



- Linguagem robusta e segura.
- Portabilidade: “escreva uma vez, execute em qualquer lugar.”
- Grande comunidade e vasto ecossistema.
- Orientação a objetos bem estruturada.
- Amplo suporte corporativo.

SPRINGBOOT



- Configuração simplificada: reduz a complexidade do setup inicial.
- Servidor embutido: não precisa configurar manualmente o Tomcat.
- Injeção de dependência: facilita o desenvolvimento modular.
- Escalabilidade e performance elevadas.
- Integração fácil com bancos de dados e APIs REST.

VANTAGENS DESSA COMBINAÇÃO:

- Desenvolvimento rápido e padronizado de aplicações robustas.
- Ideal para sistemas corporativos e de grande porte.
- Alta produtividade com menos código e mais organização.



BANCO DE DADOS: MYSQL

- Gratuito e open source.
- Alta performance e confiabilidade.
- Compatível com diversas linguagens (Java, Python, PHP, etc.).
- Fácil de instalar, configurar e administrar.
- Segurança robusta com controle de acesso por usuário.
- Suporte a grandes volumes de dados.
- Alta disponibilidade com replicação e backup simples.
- Grande comunidade e ampla documentação.



ARQUITETURA MVC



MODEL

- Representa os dados e as regras de negócio do sistema.
- É responsável por acessar o banco de dados, validar informações e aplicar a lógica principal da aplicação.



VIEW

- É a interface com o usuário — aquilo que o usuário vê e interage.
- Exibe as informações do Model e envia ações do usuário para o Controller.

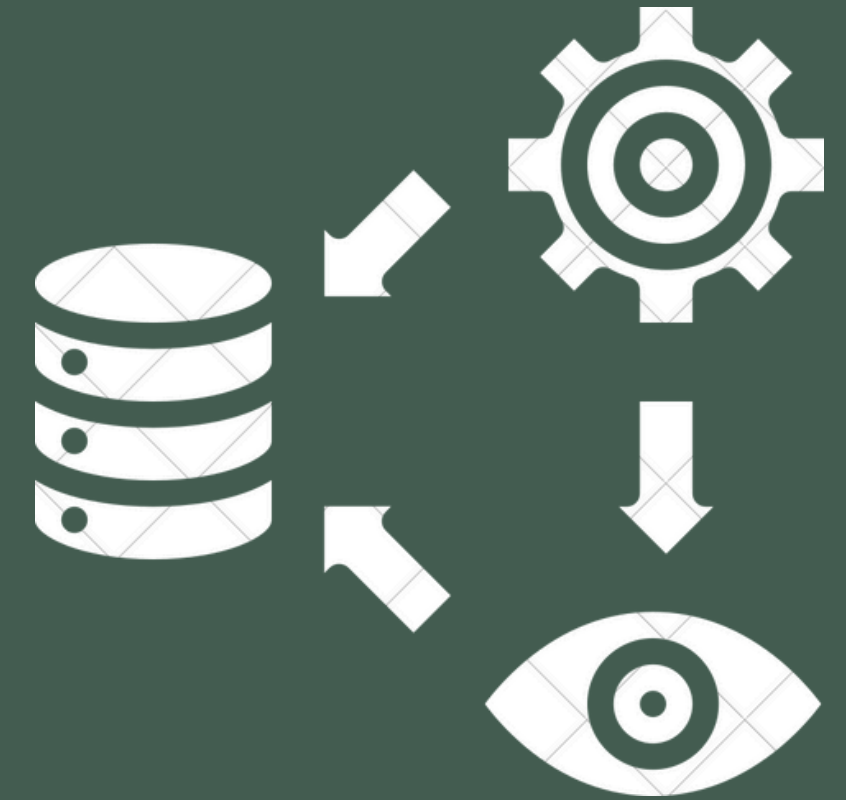


CONTROLLER

- Atua como um intermediário entre a View e o Model.
- Recebe as ações do usuário (como cliques ou envios de formulários), processa as requisições e atualiza a View conforme necessário.

VANTAGENS DA MVC

- Separação de responsabilidades: cada camada tem uma função clara (dados, interface, controle).
- Facilidade de manutenção: alterações em uma parte do sistema não afetam as outras.
- Reutilização de código: o Model e o Controller podem ser reaproveitados com diferentes Views.
- Melhor organização e escalabilidade do projeto.
- Facilita o trabalho em equipe: diferentes desenvolvedores podem atuar em camadas distintas (ex.: um no front-end e outro no back-end).
- Maior testabilidade: as camadas são isoladas, o que facilita a criação de testes unitários.



EM SUMA...

- React + Vite: garantiram uma interface moderna, rápida e interativa.
- Java + Spring Boot: proporcionaram um back-end robusto e seguro.
- Arquitetura MVC: trouxe organização, separação de responsabilidades e fácil manutenção.
- MySQL: assegurou armazenamento confiável e eficiente dos dados.
- Integração entre as tecnologias: resultou em uma aplicação completa, performática e escalável.



OBRIGADO!

