



# Arquitetura e Tecnologias do Projeto

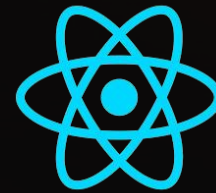
Visão geral do sistema MeritCoin

Gabriel Ramos, João Pedro e Júlia Moreira

# Frontend

## React

SPA com JavaScript.



## React Router

Navegação eficiente entre componentes.



## UI Libraries

Chakra UI e Material UI.



## Vite

Build rápido e otimizado.



# Backend

## **Spring Boot**

Framework Java robusto e escalável.

## **Arquitetura Hexagonal**

Separação clara de responsabilidades.

## **Domain-Driven Design**

Foco no domínio do negócio.



# Dependências Backend

## 1 Swagger

Documentação da API.

## 2 Lombok

Otimização de código.

## 3 Maven

Gerenciamento de dependências.







# Banco de Dados

Ambiente	Tecnologia
Produção	PostgreSQL
Testes	H2
ORM	Hibernate JPA

# Comunicação e Integração

**1**

## **APIs RESTful**

Interfaces padronizadas para comunicação entre serviços.

**2**

## **Integração Front-Back**

Comunicação eficiente via endpoints REST.

**3**

## **Serialização**

Troca de dados em formato JSON.

# Segurança do Sistema

## Spring Security

Framework robusto para autenticação e autorização.

## JWT

Tokens seguros para autenticação stateless.

## Roles

Gerenciamento granular de permissões de usuários.



# Testes e Qualidade

1

## **JUnit**

Testes unitários automatizados.

2

## **Mockito**

Simulação de comportamentos para testes isolados.

3

## **Cobertura**

Análise de cobertura de código.



# Funcionalidades Adicionais

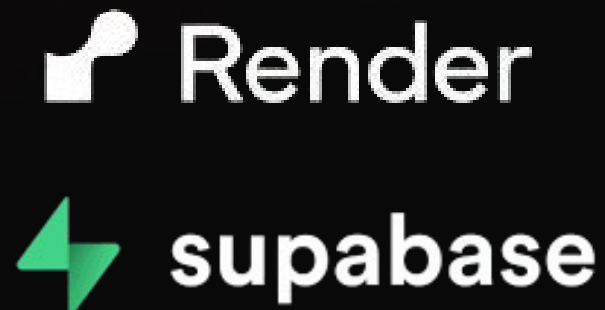
## Serviço de E-mail

Integração com JavaMail para notificações automáticas.



## Deploy

Infraestrutura em nuvem com Supabase e Render.



## Containerização

Uso de Docker para containerização do Backend.



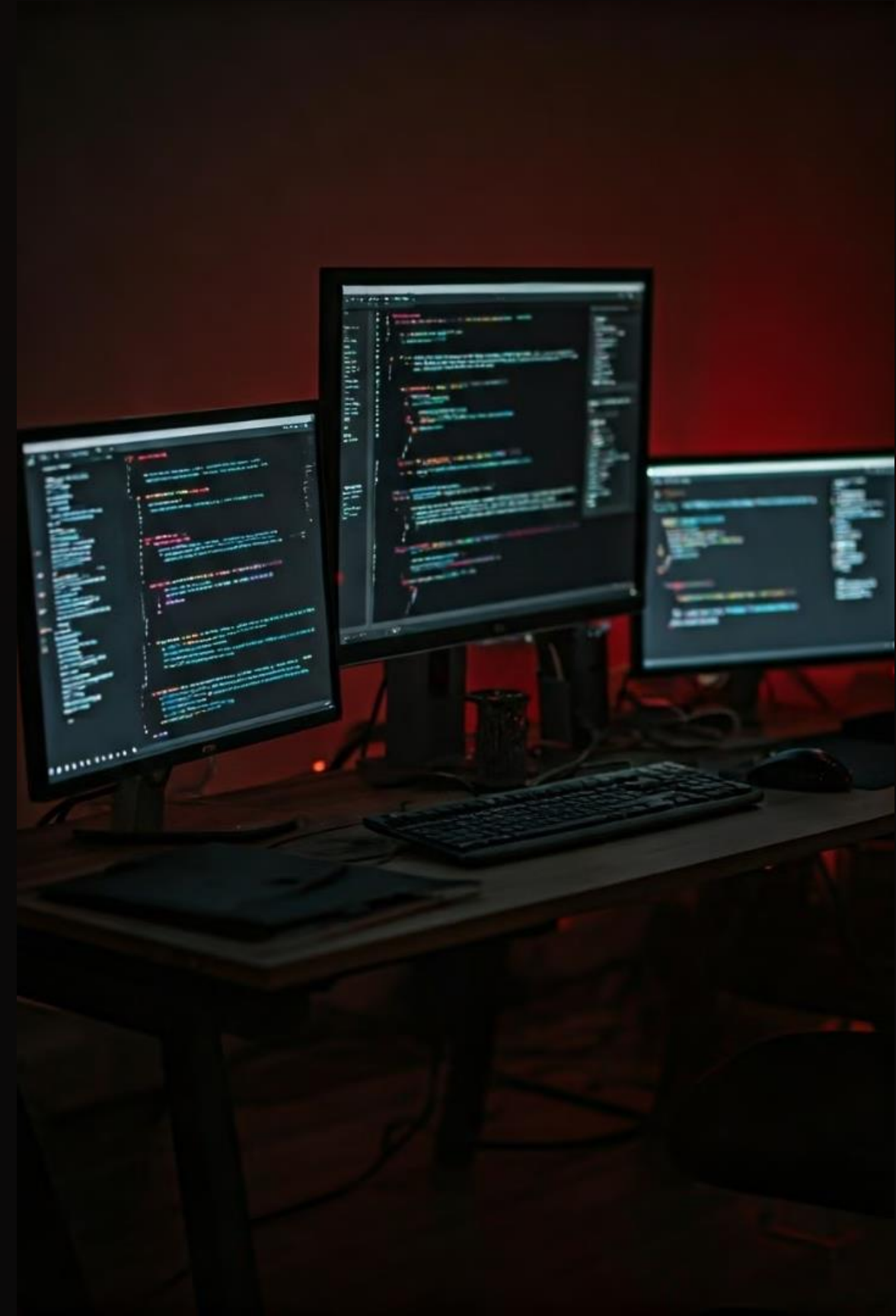
# Boas Práticas

## Clean Code

Código legível e manutenível.

## Tratamento de Erros

Spring Controller Advice para exceções centralizadas.



**Obrigado!**