

SERVIÇO PÚBLICO FEDERAL · MINISTÉRIO DA EDUCAÇÃO
UNIVERSIDADE FEDERAL DE VIÇOSA · UFV
CAMPUS FLORESTAL



CCF 323 - Arquitetura de Software

Prova de Conceito

Equipe 5º Ano - "BYTECRAFT"

Florestal - MG

2025

1. Introdução

Este documento apresenta a proposta de arquitetura para o desenvolvimento de um jogo educativo com o objetivo de ensinar a montagem de computadores e a função de seus componentes. O jogo será baseado em uma mecânica de "arrastar e soltar" (drag-and-drop), permitindo que o usuário personalize as peças utilizadas. Ao final da experiência, o jogador será submetido a um questionário para testar os conhecimentos adquiridos.

Com base na definição de que a aplicação será desenvolvida em Java com banco de dados MySQL, foram selecionadas as tecnologias, ferramentas, frameworks e bibliotecas que serão utilizadas durante o desenvolvimento. Este documento contém os detalhes do processo de tomada de decisão realizados, bem como *links* úteis que podem auxiliar no desenvolvimento do projeto.

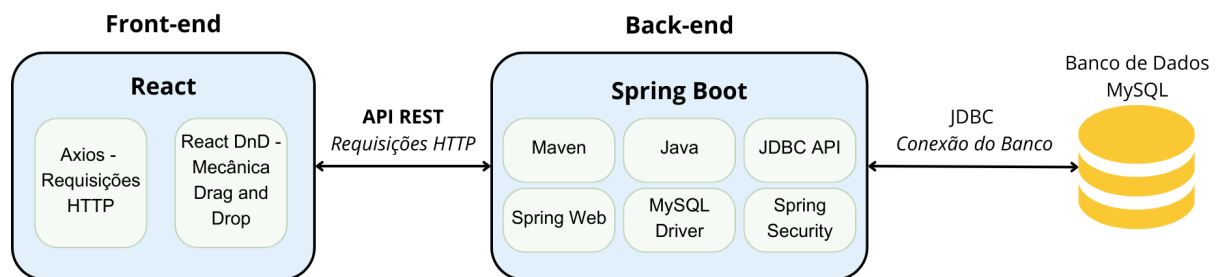


Figura 1 - Visão geral da arquitetura proposta

Na Figura 1, é apresentada a arquitetura proposta para o sistema, destacando as tecnologias utilizadas e a forma como os diferentes componentes interagem entre si. Para o front-end, optou-se pela utilização do framework React, integrado com o Vite como ferramenta de build, proporcionando uma experiência de desenvolvimento mais rápida e moderna.

A aplicação será desenvolvida em TypeScript, linguagem que oferece maior segurança e legibilidade ao código, facilitando a manutenção e escalabilidade do projeto, utilizando CSS para estilização da aplicação. O uso do React foi motivado por sua vasta comunidade, documentação abrangente e grande variedade de bibliotecas de apoio, dentre as quais serão utilizadas axios (responsável pelas requisições HTTP entre o front-end e o back-end via API REST) e React DnD (utilizada para implementar funcionalidades de arrastar e soltar).

Para o back-end foi escolhido o Spring Boot por sua robustez, facilidade de configuração e forte integração com o ecossistema Spring, especialmente na criação de APIs RESTful. A linguagem Java foi definida como base do projeto por ser amplamente utilizada nas disciplinas do curso e já conhecida pela equipe de desenvolvimento. Para o gerenciamento do ciclo de vida da aplicação e de suas dependências, utiliza-se o Maven, enquanto o Spring Web e o Spring Security são responsáveis, respectivamente, pela construção dos controladores RESTful e pela implementação dos mecanismos de autenticação e autorização. A JDBC API foi adotada para permitir a execução direta de comandos SQL, possibilitando maior aproveitamento dos scripts desenvolvidos pelos integrantes da equipe de banco de dados, em especial os desenvolvedores júnior, ao passo que o MySQL Driver estabelece a conexão com o banco de dados relacional MySQL.

A arquitetura proposta permite a separação clara entre o front-end e o back-end, comunicando-se exclusivamente por meio de API RESTful, o que favorece a escalabilidade e a independência entre as camadas.

2. Definição Técnica

Para o gerenciamento de dependências, tanto o Spring Boot quanto o Vite oferecem mecanismos automatizados para a instalação de bibliotecas que auxiliam no desenvolvimento. No entanto, é essencial avaliar criteriosamente cada nova dependência adicionada ao projeto, considerando seu impacto em termos de desempenho, consumo de recursos e compatibilidade com o ambiente de execução. No back-end, o uso da JDBC API foi adotado para permitir a execução direta de comandos SQL, o que proporciona maior aproveitamento dos scripts criados pela equipe de banco de dados. O Maven continua sendo utilizado para gerenciar as dependências e o ciclo de vida do projeto Java.

Durante o desenvolvimento da prova de conceito, diferentes IDEs foram utilizadas. Para o front-end, recomenda-se o uso do Visual Studio Code, que apresenta excelente integração com as tecnologias React, Vite e TypeScript, além de oferecer uma vasta gama de extensões que aumentam a produtividade. No back-end, a IDE IntelliJ IDEA é indicada por sua integração avançada com Java e Spring Boot, recursos de refatoração e depuração.

Com relação ao gerenciamento de estado e organização de componentes no front-end, o uso do React permite uma abordagem modular baseada em componentes reutilizáveis, facilitando a escalabilidade e manutenção do sistema. A biblioteca React DnD foi integrada ao projeto para a implementação de funcionalidades de arrastar e soltar, e a Axios é utilizada para a comunicação entre front-end e back-end, via requisições HTTP. É importante manter uma divisão clara entre componentes e estados locais e globais, adotando práticas que favoreçam a legibilidade e a organização do código.

Por fim, a estrutura dos projetos segue modelos amplamente recomendados pelas comunidades das tecnologias utilizadas. O repositório da prova de conceito apresenta uma proposta de organização de diretórios e arquivos voltada para facilitar o entendimento das diferentes partes do sistema, promovendo uma manutenção mais eficiente e reduzindo a curva de aprendizado para novos membros. Embora essa estrutura possa ser adaptada conforme a necessidade do projeto, manter uma arquitetura limpa e bem definida é essencial para o sucesso a longo prazo do desenvolvimento.

3. Ferramentas e Instalações

Nesta seção serão apresentadas as versões e o processo de instalação das ferramentas utilizadas para o desenvolvimento da prova de conceito.

1) Tecnologias do Front-end:

- React: 19.1.1
- Axios: 1.11.0
- React-DnD: 16.0.1

2) Tecnologias do Back-end:

- Java: 17
- Maven: 3.8.6
- Spring Boot: 3.5.4

- Spring Web: 3.5.4
- Spring Security: 6.5.3

3.1 Instalações para o Backend - Spring Boot

Criando o projeto:

1. Acesse o [Spring Initializr](#)

2. Configure:

- Build Tool: Maven
- Linguagem: Java
- Versão: 17

3. Adicione as dependências:

- Spring Web
- MySQL Driver
- JDBC API
- Spring Boot DevTools
- Spring Security

3.2 Instalações para o Frontend - React + Vite + React DnD + Axios

Passos para criação do projeto:

Shell

```
npm create vite@latest
```

- Escolha um nome para o projeto.
- Selecione o framework React com a variante TypeScript.

Shell

```
cd nome-do-projeto  
npm install
```

Instalação do React DnD:

Shell

```
npm install react-dnd react-dnd-html5-backend
```

4. Links úteis

Nesta seção serão disponibilizados alguns *links* que poderão auxiliar durante o desenvolvimento do projeto, sendo recomendo aos desenvolvedores a procura por outros métodos ou guias.

- React: <https://pt-br.react.dev/learn>

- Tutorial criando uma aplicação full-stack utilizando spring boot + react:
<https://www.youtube.com/watch?v=IUvureR5GqI>
- Axios: <https://axios-http.com/ptbr/docs/intro>
- React-Dnd: <https://react-dnd.github.io/react-dnd/docs/tutorial>
- JDBC API: <https://docs.oracle.com/javase/tutorial/jdbc/basics/index.html>
 - <https://www.geeksforgeeks.org/java/introduction-to-jdbc/>
- IDE Visual Studio Code: <https://code.visualstudio.com/>
- IDE IntelliJ Idea: <https://www.jetbrains.com/idea/>
- Spring Initializr: <https://start.spring.io/>