

INSTITUTO FEDERAL DE MINAS GERAIS  
CAMPUS OURO BRANCO  
TÉCNICO INTEGRADO EM INFORMÁTICA

**ANTONNY GABRIEL AZEVEDO CALDEIRA SILVA**  
**MATHEUS HENRIQUE DE MELO**

## **Documentação: Sistema de Gerenciamento Financeiro**

**Ouro Branco**  
**Abril de 2025**

## Sumário

1.	INTRODUÇÃO.....	3
2.	IMPLEMENTAÇÃO .....	4
2.1.	Estruturação do Projeto .....	4
2.2.	Desenvolvimento do back-end .....	5
2.3.	Desenvolvimento do front-end .....	5
2.4.	Integração .....	6
3.	CONCLUSÃO.....	6
4.	BIBLIOGRAFIA .....	7

# 1. INTRODUÇÃO

Foi requisitado que uma aplicação de gerenciamento financeiro fosse desenvolvida pelos discentes. A aplicação deve permitir que o usuário crie transações financeiras de diferentes tipos de entrada e saída, que serão armazenadas no banco de dados e mostradas na tela principal. Elas podem ser filtradas por meio de um intervalo de tempo, que também pode ser o mês atual que o usuário utiliza a aplicação.

Além disso, a aplicação conta com uma ferramenta de lixeira, para que, quando o usuário apagar alguma transação, ele possa recuperá-la, ou excluí-la definitivamente do banco de dados.

Cada transação deve possuir um nome, uma classificação, um valor, a data em que ocorreu e a data de cadastro na aplicação.

Para realizar o trabalho foram utilizados a linguagem de programação Java, principalmente por meio do framework Spring, o banco de dados PostgreSQL, e ferramentas de programação web (HTML, CSS e Javascript).

## 2. IMPLEMENTAÇÃO

A implementação pode ser dividida em três tópicos que a descrevem bem: **Estruturação do projeto**, **Desenvolvimento do back-end**, **Desenvolvimento do front-end** e por fim a **Integração** entre ambos.

### 2.1. Estruturação do Projeto

Inicialmente, foi necessário construir a estrutura do projeto, para que seu desenvolvimento fosse o mais rápido, direto e concreto possível. Para tanto, as funcionalidades foram separadas da seguinte forma:

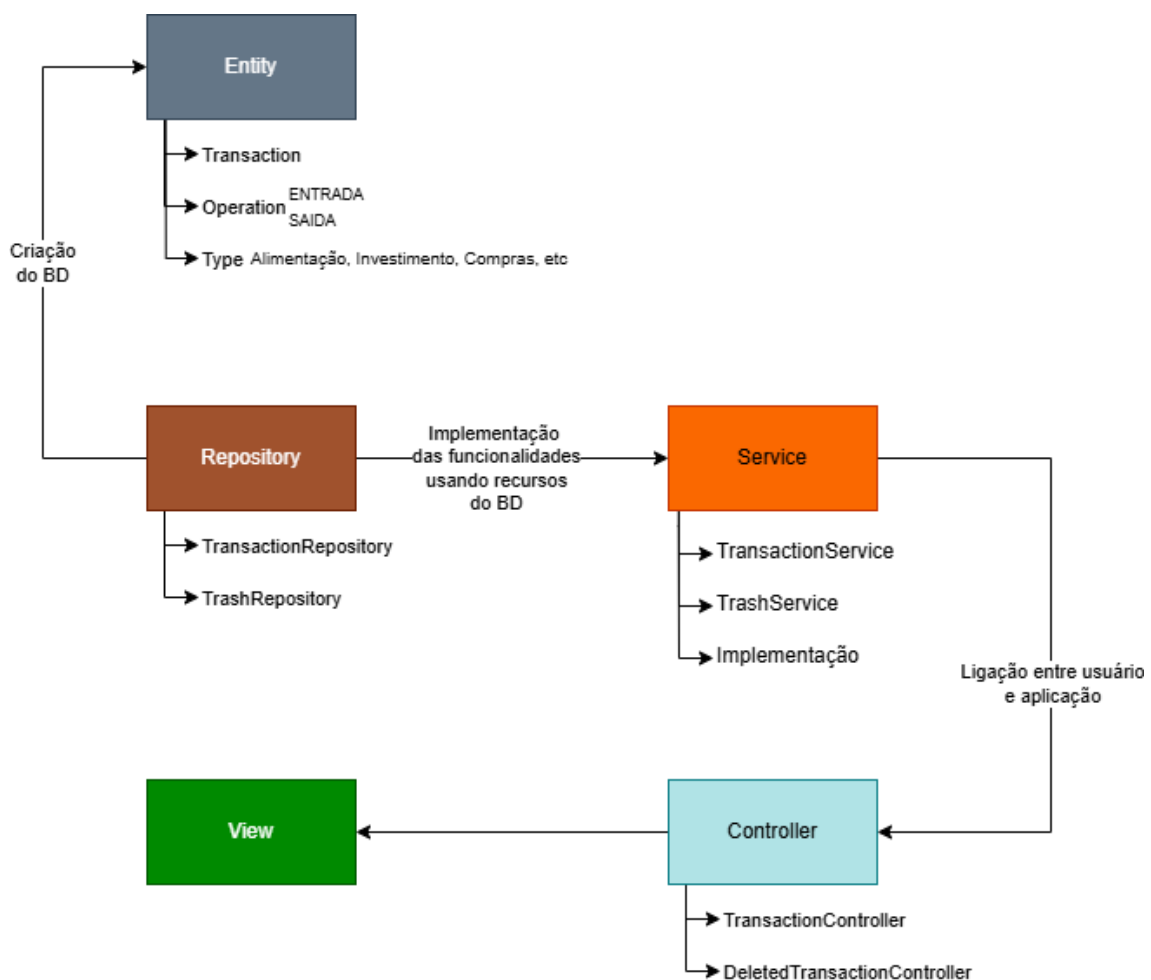


Figura 1. Diagrama da Estrutura do Projeto

Assim, a entidade será o ponto de partida, ao estabelecer a ideia mínima do projeto. Ela será aplicada ao banco de dados por meio de um Repository que a liga com o PostgreSQL, sendo implementado por um Service que vai possuir as regras de negócio e manipular as informações que serão gerenciadas pelo Controller que entregará e obterá essas informações do usuário por meio de uma interface gráfica.

## 2.2. Desenvolvimento do back-end

Para aplicar o que foi requisitado, utilizou-se Java por meio do framework Spring, com o qual grande parte do processo foi agilizado, considerando a utilização de anotações que implementam blocos de código úteis no desenvolvimento da aplicação, que tornaram as classes mais enxutas, sem o excesso de códigos repetidos. As anotações do Spring também foram úteis para gerenciar o projeto e configurar as rotas web.

O Spring facilitou também a conexão com o banco de dados PostgreSQL, permitindo sua manipulação com poucas linhas de código. Esse SGBD foi escolhido pela familiaridade dos discentes com seu sistema, além de sua robustez e praticidade.

A organização dos pacotes se baseou em um padrão muito utilizado por aplicações web, considerando que assim cada camada terá uma responsabilidade única, como também cada alteração necessária pode ser facilmente feita considerando a flexibilidade desse modelo.

Vale ressaltar o funcionamento da Lixeira que se deu por meio de um simples booleano que indicaria se a transação seria visível ou não na listagem de transações. Além disso, a utilização de Enum's também foi de tamanha importância para a legibilidade e versatilidade do código, que funcionaram como listas de tipos de transações e de operações possíveis, funcionando separadamente da entidade de transação.

## 2.3. Desenvolvimento do front-end

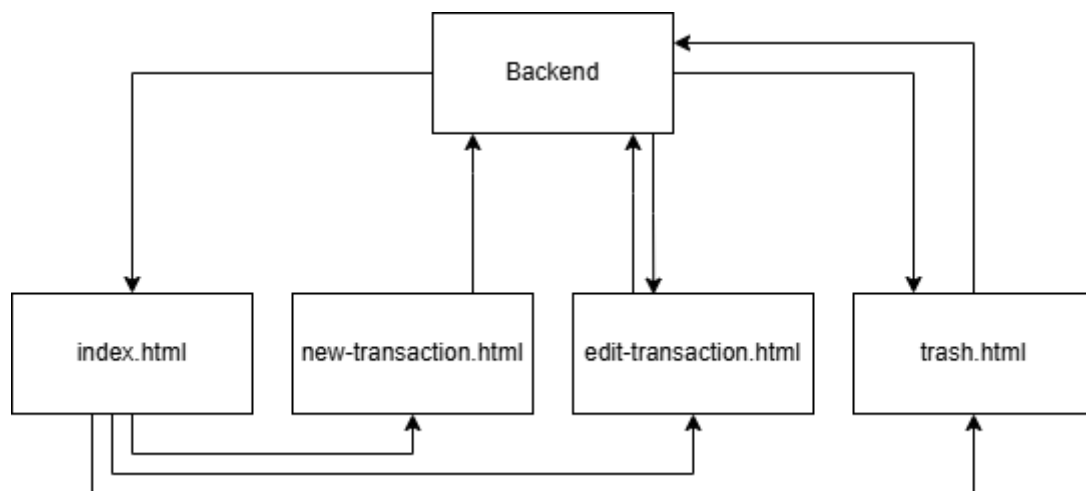
Considerando o foco do trabalho na aplicação back-end, os alunos desenvolveram a interface gráfica por meio de HTML estático. A estilização foi trabalhada com Tailwindcss, um framework CSS útil pela eficiência com que desenvolve os estilos e legibilidade. Foram criadas apenas quatro páginas:

- Tela principal
- Tela de criação de transação
- Tela de edição de transação
- Lixeira

Além disso, outro detalhe do front-end foi a criação de um script usado exclusivamente para a formatação dos valores das transações para valores monetários, tornando a interface mais amigável e profissional. O menu de navegação foi criado separado das telas para ser implementado de uma mesma maneira para todas.

## 2.4. Integração

Para integrar o back-end com o front-end, foi utilizado o motor Thymeleaf, que integra o Java ao HTML, o que tornou o fluxo de trabalho mais eficiente, considerando o quão acessível se tornou expor informações na interface como também de receber valores para inserir no banco de dados. O Thymeleaf foi importante devido a sua aplicabilidade, segurança e controle de fluxo.



## 3. CONCLUSÃO

Com o desenvolvimento deste projeto, os discentes trabalharam fortes conceitos de banco de dados, aplicações web, arquitetura de software, Programação Orientada a Objetos como também entenderam a aplicabilidade de frameworks muito utilizadas no mercado. O desafio de aplicar os conceitos aprendidos em sala em um framework novo foi de grande importância para os alunos focarem na organização do projeto, como também no trabalho em equipe, algo fundamental para aplicações de maior complexidade como a realizada.

Dessa forma, é possível concluir que o trabalho foi muito proveitoso para desenvolver novas habilidades de trabalho em equipe e de desenvolvimento de software, como também se provou útil para rever conceitos de Java e Banco de Dados.

## 4. BIBLIOGRAFIA

- <https://www.devmedia.com.br/java-spring-mvc-criando-aplicacoes-web-em-java/31521>
- <https://spring.io/guides>
- <https://www.devmedia.com.br/introducao-pratica-ao-spring-framework-com-uso-de-anotacoes/27859>
- <https://www.baeldung.com/spring-boot-react-crud>
- <https://www.devmedia.com.br/tipos-enum-no-java/25729>
- <https://www.thymeleaf.org/>
- <https://flowbite.com/>