DESCRITIVO DO DESAFIO

objetivo: Criar um projeto Spring Boot para permitir o cadastramento, consulta, exclusão, lista, pesquisa e alteração de Clientes seguindo a modelagem fornecida.

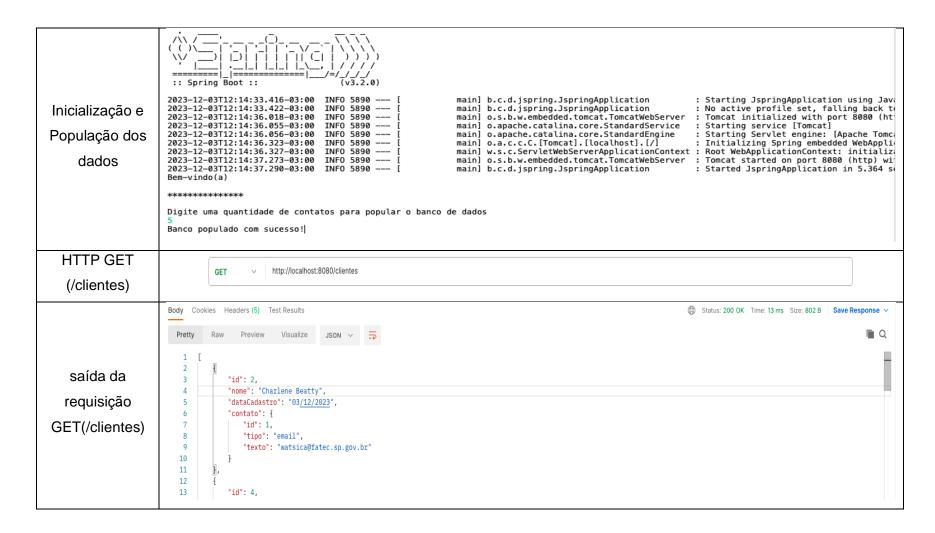
Este documento é uma versão simplificada de apresentação da resolução do desafio e, nele, você encontrará:

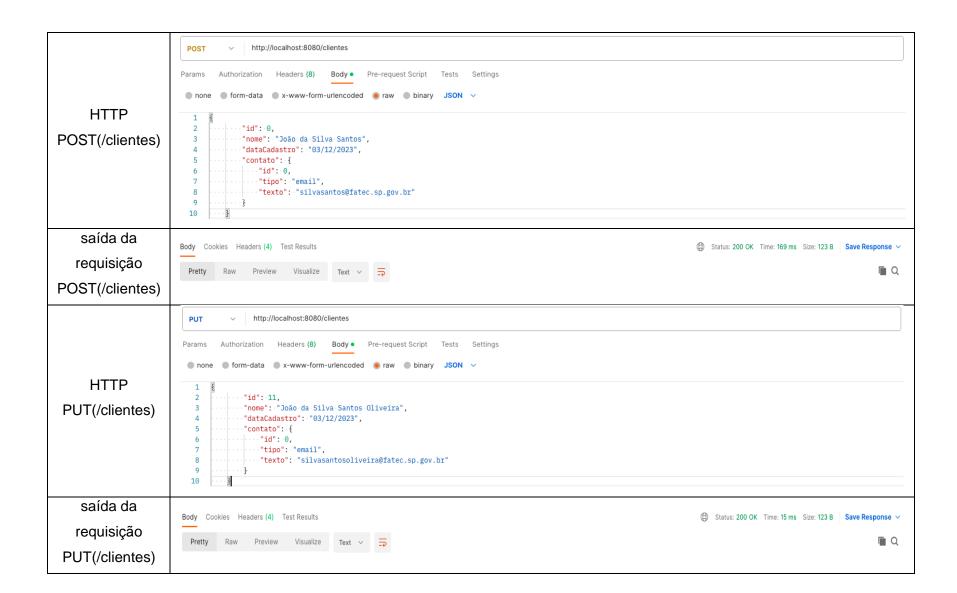
- 1. Funcionamento
- 2. Organização dos Pacotes
- 3. Observações Gerais

Você poderá acessar o código da resolução através do link https://github.com/patrick-tobias/022023_d-muralis_java-spring

FUNCIONAMENTO

Nesta seção, você encontrará capturas de telas, como a aplicação desenvolvida responde às requisições HTTP.



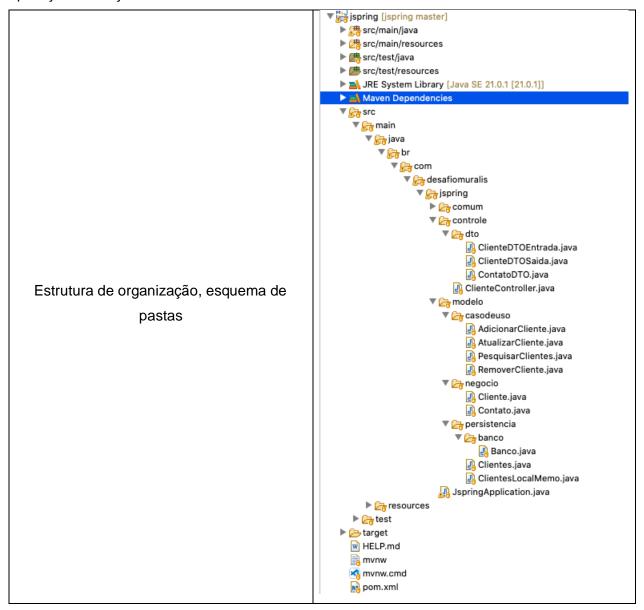




observação: Ferramenta utilizada - Postman

ORGANIZAÇÃO DOS PACOTES

Nesta seção, você encontrará a estratégia de organização das pastas e subpastas da aplicação resolução do desafio.



OBSERVAÇÕES GERAIS

Nesta seção, você encontrará o raciocínio executado para a tomada de decisão durante a realização do desafio.

Banco de Dados Simulado

A iniciativa de incluir um **banco de dados simulado** surgiu da dúvida gerada ao ler o desafio. O desafio não incluía conexão e persistência em banco de dados.

Através de Listas, foi possível simular um banco de dados e suas operações CRUD.

Testes de Funcionamento

A resolução do desafio sugeriu a construção de uma API Restful. Com a finalidade de simular requisições HTTP, utilizei a ferramenta **Postman**.

Interface Repositório

Utilizar interfaces na classe que implementam e simulam o "Banco de Dados" foi uma decisão tomada em teoria, pela facilidade de substituição do Banco de Dados e manutenção da troca.

Métodos separados em Casos de Uso

Utilizar classes que representam um caso de uso do CRUD foi uma decisão tomada com a preocupação em não sobrecarregar classes de negócio com métodos que fogem de seu controle. Assim, Cliente é responsável por gerenciar seus próprios, e somente os seus próprios, atributos.