

Itaú Unibanco - Desafio de Programação

Este é um desafio bacana tanto de desenvolvimento de software quanto de engenharia de software.

Queremos testar sua capacidade de construir um software com várias partes diferentes funcionando em conjunto!

1. Introdução

Sua missão, caso você aceite, é criar uma API REST que recebe Transações e retorna Estatísticas sob essas transações.

Para este desafio, a API deve ser criada utilizando-se de Java ou Kotlin e Spring Boot.

Dica: Não existe uma forma certa ou errada de resolver o desafio! Vamos avaliar coisas como a qualidade do seu código,

o quão fácil é de compreender o código, organização do projeto, quantidade e qualidade dos testes,

preocupação com segurança e vários outros fatores :)

2. Definição do desafio

Neste desafio você deve criar uma API REST no GitHub ou GitLab. Leia com atenção todas as instruções a seguir!

2.1. Restrições Técnicas

Seu projeto:

- DEVE estar no GitHub ou GitLab
- NÃO DEVE fazer fork de nenhum outro projeto
- DEVE ter pelo menos 1 commit por cada endpoint (mínimo de 3 commits)
- DEVE seguir exatamente os endpoints descritos a seguir
- DEVE armazenar todos os dados em memória
- DEVE aceitar e responder apenas com JSON

2.2. Endpoints da API

2.2.1. Receber Transações: POST /transacao

Este endpoint recebe transações no seguinte formato:

```
{  
  "valor": 123.45,  
  "dataHora": "2020-08-07T12:34:56.789-03:00"  
}
```

Restrições:

- valor deve ser maior ou igual a 0
- dataHora deve estar no passado

Respostas esperadas:

- 201 Created: Transação aceita
- 422 Unprocessable Entity: Transação inválida
- 400 Bad Request: JSON inválido

2.2.2. Limpar Transações: DELETE /transacao

Este endpoint remove todas as transações armazenadas.

Resposta esperada:

- 200 OK: Todas as transações foram apagadas

2.2.3. Calcular Estatísticas: GET /estatistica

Retorna estatísticas das transações dos últimos 60 segundos:

```
{  
  "count": 10,  
  "sum": 1234.56,  
  "avg": 123.456,  
  "min": 12.34,  
  "max": 123.56  
}
```

Resposta esperada:

- 200 OK: Estatísticas calculadas corretamente

3. Extras

Desafios adicionais incluem:

- Testes automatizados
- Containerização
- Logs e Observabilidade
- Tratamento de Erros
- Documentação da API e do sistema
- Configuração do tempo para cálculo de estatísticas

Boa sorte no desafio!