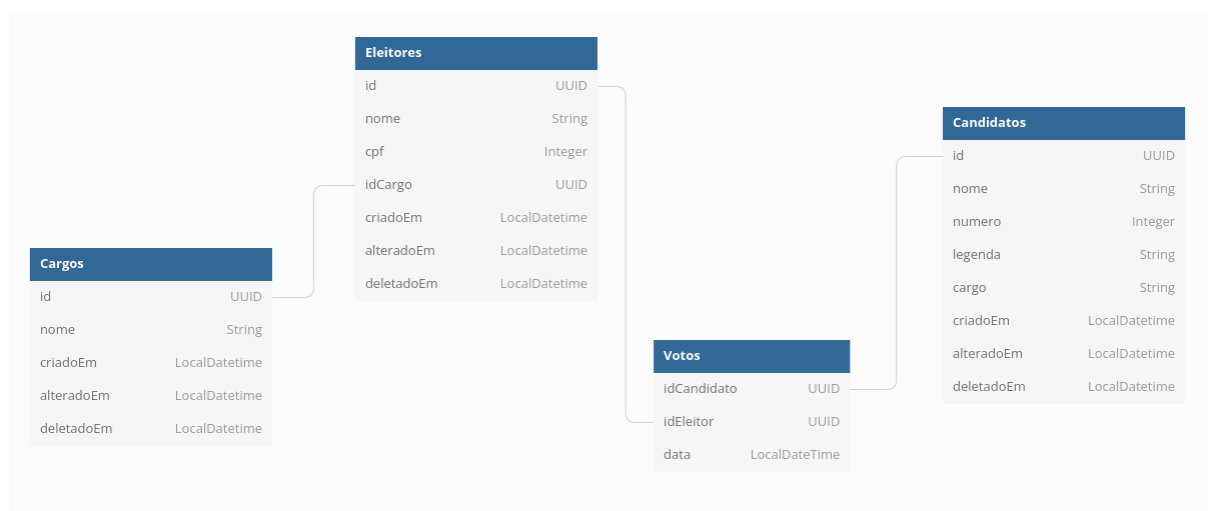


Introdução

Como parte do processo seletivo, gostaríamos que você fizesse uma pequena tarefa para que possamos lhe conhecer melhor tecnicamente. O problema proposto é uma situação fictícia e você não estará desenvolvendo uma solução para nossa empresa neste teste.

Vamos implementar uma pequena API, que permita um cadastro básico dos candidatos, eleitores, dos votos e depois um outro recurso para consultar a quantidade de votos para cada candidato. Embora pareça simples, nessa pequena API podemos treinar algumas técnicas de arquitetura e engenharia de software.

Para facilitar vamos utilizar a seguinte modelagem, onde teremos 04 tabelas, uma para o cadastro dos eleitores, outra para os cargos, votos e por fim uma para os candidatos. Abaixo segue uma imagem representando os campos em cada tabela, assim como os relacionamentos.



<https://dbdiagram.io/d/627ea17a7f945876b6161f19>

Stack sugerida

- Java 17+

- Spring Framework
- Spring Boot
- Spring Data JPA
- Spring Web
- Banco de dados relacional
- REST com OpenAPI/Swagger

Características a serem seguidas

- API em REST Full.
- Validar para permitir apenas um voto por eleitor.
- Os dados devem persistir no banco de dados.
- Validar para não permitir candidatos e eleitores duplicados.
- NÃO utilize nenhum gerador de código, tente fazer utilizando os seus conhecimentos.
- Caso algum candidato ou eleitor já possua um voto, este não poderá ser apagado, ou seja, só permitir apagar caso ele ainda não tenha nenhum voto computado.

Recursos/End-points

- CRUD para /candidatos
- CRUD para /eleitores
- CRUD para /cargos
- POST /eleitores/{id}/votar

```
{
    idCandidato: "",
    idEleitor: ""
}
```
- GET /candidatos/relatorio com a seguinte resposta:

```
[
    {
        idCargo: 1,
        nomeCargo: "PREFEITO",
```

```
        votos: 500,  
        idCandidatoVencedor: 2,  
        nomeCandidatoVencedor: "NOME DO PREFEITO"  
    },  
    {  
        idCargo: 2,  
        nomeCargo: "PRESIDENTE",  
        votos: 1000,  
        idCandidatoVencedor: 1,  
        nomeCandidatoVencedor: "NOME DO PRESIDENTE"  
    }  
]
```

O que será avaliado

- O código deverá ser versionado via git, no Github e enviado o link como resposta a este e-mail. Lembre de deixar o repositório público.
- Documente bem o que foi desenvolvido, quais os requisitos para executar o seu projeto, além das tecnologias utilizadas, criando um arquivo chamado de README.md na raiz do seu projeto. Lembrando que esses arquivos utilizam de Markdown como linguagem.
- Analisaremos todo o código escrito, os padrões utilizados e o nível do seu conhecimento com as tecnologias solicitadas. Lembrando que o objetivo é resolver o problema solicitado, não precisa de Rocket Science nem Over Engineering

O que será um diferencial

- Interface gráfica
- Utilização de cache
- Versionamento do banco de dados com Liquibase

Por fim, desejamos uma boa sorte e lembramos que o principal objetivo com isso é conhecer melhor o candidato, o seu momento e as suas habilidades técnicas adquiridas.

Boa Sorte!!