

Desafio Técnico

Objetivo

No cooperativismo, cada associado possui um voto e as decisões são tomadas em assembleias, por votação. A partir disso, você precisa criar uma solução back-end para gerenciar essas sessões de votação.

Essa solução deve ser executada na nuvem e promover as seguintes funcionalidades através de uma API REST:

- ✓ Cadastrar uma nova pauta;
- ✓ Abrir uma sessão de votação em uma pauta (a sessão de votação deve ficar aberta por um tempo determinado na chamada de abertura ou 1 minuto por default);
- ✓ Receber votos dos associados em pautas (os votos são apenas 'Sim'/'Não'. Cada associado é identificado por um id único e pode votar apenas uma vez por pauta);
- ✓ Contabilizar os votos e dar o resultado da votação na pauta.

Para fins de exercício, a segurança das interfaces pode ser abstraída e qualquer chamada para as interfaces pode ser considerada como autorizada. A escolha da linguagem, frameworks e bibliotecas é livre (desde que não infrinja direitos de uso).

É importante que as pautas e os votos sejam persistidos e que não sejam perdidos com o restart da aplicação.

Tarefas bônus

As tarefas bônus não são obrigatórias, mas nos permitem avaliar outros conhecimentos que você possa ter. A gente sempre sugere que o candidato pondere e apresente até onde consegue fazer, considerando o seu nível de conhecimento e a qualidade da entrega.

- Verifique a partir do CPF do associado, se ele pode votar
Caso o CPF seja inválido, a API retornará HTTP Status 400 (Bad Request). Você pode usar geradores de CPF para gerar CPFs válidos;
O CPF do associado deverá alternar entre (ABLE_TO_VOTE) e (UNABLE_TO_VOTE).

```
{  
  "status": "ABLE_TO_VOTE"  
}  
  
{  
  "status": "UNABLE_TO_VOTE"  
}
```

- Mensageria e filas:

A votação precisa ser informada para o restante da plataforma, isso deve ser feito através de mensageria.



Performance:

Imagine que a sua aplicação possa ser usada em cenários que existam milhares de votos. Ela deve se comportar de maneira performática nesses cenários; Testes de performance são uma boa maneira de garantir e observar como sua aplicação se comporta.



Versionamento da API:

Como você versionaria sua aplicação? Que estratégia usar? (Versionamento de URL, API etc..)



Que estratégia utilizar para o encerramento de pautas de votação? Lembrando que sua aplicação estará sendo executada na nuvem possivelmente um Kubernetes.

■ Demonstre o uso de arquitetura/design de software Hexagonal ou Clean Architecture.

-

O que será analisado

- Organização do código
- Arquitetura do projeto
- Boas práticas de programação (manutenibilidade, legibilidade etc)
- Possíveis bugs
- Tratamento de erros e exceções
- Explicação breve do porquê das escolhas tomadas durante o desenvolvimento da solução
- Uso de testes automatizados e ferramentas de qualidade
- Limpeza do código
- Documentação do código e da API
- Logs da aplicação

Observações importantes

- Não inicie o teste sem sanar todas as dúvidas
- Iremos executar a aplicação para testá-la, cuide com qualquer dependência externa e deixe claro caso haja instruções especiais para execução do mesmo
- Teste bem sua solução, evite bugs