



Especificação do teste para Tech Lead - Beleaf

Objetivo

Neste teste, o seu objetivo é montar um pequeno sistema de cardápio de marmitas online. O sistema será composto por um pequeno servidor onde será possível fazer um CRUD com autenticação e duas páginas simples que exibem os dados providos por ele.

Tecnologias

- HTML
- CSS
- Javascript
- Node.js
- Banco de dados relacional

Sinta-se livre para usar frameworks como Angular, React, templates com Pug, Sass, GraphQL, entre outros (se tiver alguma dúvida, pode perguntar também ;)). O importante é que a linguagem dos scripts seja Javascript (nada de PHP!) e seja usado Node no servidor. O banco de dados fica a seu critério, desde que seja relacional.

Requisitos

O teste é composto por duas etapas, o servidor que irá prover os dados e a página que vai exibir os resultados. Sinta-se livre para cumprir os requisitos da forma que julgar melhor, o importante é que seja possível executar todas as ações descritas neles.

Você vai perceber que os requisitos são relativamente vagos, pois queremos ver as suas escolhas e o que julga melhor para completar a tarefa.

Regras do servidor:

- Um endpoint para que seja possível fazer a autenticação. Pode ser usado qualquer usuário e senha, desde que seja especificado no README do projeto.
- Uma vez autenticado, o usuário pode fazer 3 ações:
 - Criar uma marmita. A marmita é composta pelos seguintes campos:
 - Nome (obrigatório)
 - Descrição (obrigatório)
 - Preço (obrigatório)
 - Lista de ingredientes (obrigatório)
 - Quantidade em estoque (obrigatório)
 - Url da imagem (obrigatório)
 - Porcentagem de desconto (opcional)

O endpoint deve retornar pelo menos o identificador da marmita, em caso de sucesso.

- Atualizar uma marmita, passando o identificador e os campos a serem atualizados. O retorno de sucesso pode ser o mesmo do endpoint de criação
 - Excluir uma marmita passando o seu identificador
- Sem estar autenticado será possível fazer as seguintes ações:
 - Listar marmitas, de forma que seja possível paginar e ordenar
 - Obter os detalhes de uma única marmita passando o seu identificador
- Podem haver mais endpoints ou mais recursos caso julgue que podem te ajudar.
- Os dados das marmitas precisam estar no banco de dados

Para o front:

- O front consiste de apenas duas páginas, sendo a primeira uma exibição de todas as marmitas e uma com o detalhe da marmita.
- Não se preocupe com marcas, imagens bonitas, etc. O que será avaliado é a usabilidade, a qualidade do código e a eficiência.
- Para a lista de marmitas:
 - Para cada item do endpoint de listagem do servidor, deve ser exibido o nome, a imagem e os preços (sem desconto e com desconto)

- Caso o número de marmitas em estoque seja 0, a marmitta não deve aparecer na listagem
- Ao clicar em um item, deve-se ir para a página de detalhe da marmitta
- Para o detalhe da marmitta:
 - Devem ser exibidas as mesmas informações da listagem, além da descrição e da lista de ingredientes

Entregável

- Seu código deve estar "deployado" e rodando em algum lugar.
- Repositório(s) no Github com o seu código
- O README do seu projeto deve conter:
 - Instruções para rodar o projeto localmente (lembre-se de incluir instruções sobre o banco de dados também)
 - URLs da página e da API
 - Tecnologias usadas e a razão da escolha delas
 - Instruções para fazer o login, com usuário e senha de testes.
- Seu repositório também deve conter alguma documentação sobre o uso dos endpoints da sua API

Caso não tenha cumprido todos os requisitos, não deixe de entregar, apenas escreva do README o que faltou, e como seria feito :)

Avaliação

Os critérios que serão utilizados para avaliar o seu projeto são:

- Cumprimento dos requisitos
- Qualidade do código
- Escolhas feitas em relação a tecnologias, arquitetura, padrões, schemas das entidades, etc.
- Usabilidade do sistema como um todo
- Foi além dos requisitos (adicionou mais recursos relevantes, a página é responsiva, é possível gerenciar usuários, utilizou Docker para rodar localmente, fez alguma lógica de cache, testes automatizados, etc)
- Pensou nos casos de erro (por exemplo, o que acontece se tentarmos atualizar uma marmitta que não existe?)