



Case - Desenvolvedor Front-end Angular

Neste projeto você deverá implementar uma aplicação Single Page Application para realizar o cadastro de itens.

A aplicação deverá ser composta por 2 páginas, sendo uma para cadastro e outra para listagem. Ambas as páginas devem possuir um menu no topo com links para as mesmas, além de um sistema de navegação estrutural.

Os dados devem ser persistidos no formato JSON, fazendo uso da LocalStorage ou DbJson, adicionando, removendo e editando itens do JSON em questão. Não é necessária a utilização de APIs.

1- Produto:

Deve ser criado um formulário com validação conforme os campos descritos a seguir para cadastro de itens. É necessário informar com clareza os campos que apresentem erro de validação ou obrigatoriedade.

Campo	Tipo	Obrigatoriedade	Validação
Nome do item	Texto	Sim	Tamanho máximo de 50 caracteres (somente letras)
Unidade de medida	Enumeração *	Sim	-
Quantidade	Numérico	Não	Varia conforme regra da unidade medida **
Preço	Monetário	Sim	Validações de campo monetário***
Produto perecível	Checkbox booleano	Sim	-
Data de validade	Data	Só é obrigatório caso o produto seja perecível	Data no formato pt-Br. Caso a data de validade seja inferior a data atual deve informar que o produto encontra-se vencido.
Data de fabricação	Data	Sim	Data no formato pt-Br e não pode ser superior a data de validade (caso seja um produto perecível)

- Campos com unidade de medida em litro deve permitir somente números, com até 3 casas decimais e apresentar a abreviatura "lt" ao final do campo (addon);
- Campos com unidade de medida em Quilograma deve permitir somente números, com até 3 casas decimais e apresentar a abreviatura "kg" ao final do campo (addon);
- Campos com unidade de medida em Unidade deve permitir somente números inteiros e apresentar a abreviatura "un" ao final do campo (addon);

*** Validação de campo monetário inclui exibição do tipo de moeda no início do campo e limite de casas decimais utilizando máscara (preenchimento da direita para esquerda).

O formulário deverá possuir dois botões, um para salvar e outro para cancelar. Ao selecionar o botão de salvar, caso esteja editando um item, deve salvar essas alterações, do contrário um novo será adicionado. O botão de cancelar direciona o usuário para a tela de listagem.

2- Listagem:

A listagem deverá exibir uma tabela com os itens cadastrados, bem como um botão para edição e outro para a exclusão de cada item, e um botão para adicionar um novo item. A busca dos itens para listagem deve ser realizada de forma assíncrona. Ao clicar em excluir o usuário deverá ser questionado se realmente deseja excluir o item em questão. Caso sim, o item deverá ser removido e exibida uma notificação de sucesso ou erro.

Ao clicar em editar, o usuário deverá ser redirecionado para o formulário, carregado com os dados do item em questão. O botão de adicionar apenas redirecionará para a página do formulário.

Tecnologias e conceitos que serão avaliados:

- Angular 15 +
- HTML 5;
- CSS e/ou SASS;
- TypeScript (preferencial) / JavaScript ;
- Rotas;
- Utilização de LocalStorage ou DbJson;
- Utiliza uma UI, como Bootstrap, Material ou Ant Design.

Critérios de avaliação da tela:

- Usabilidade e experiência de uso;

- Responsividade (deve ser ajustável a diferentes tamanhos de tela);
- Padrão visual;
- Cross-browser (deve ser utilizavel em ie 10 +, Edge, Chrome, FireFox)

Critério de avaliação do código:

- Qualidade;
- Clareza;
- Documentação;
- Reutilização (criação de componentes);

Não conseguiu fazer tudo?

A sua forma de priorizar a entrega também vai ser considerada. Escreva no README.md o que você conseguiu e não conseguiu implementar, descrevendo porque você preferiu priorizar desta maneira.