

Teste para Frontend Dev - Júnior

munddi.com - março/2021

1. Objetivo

Desenvolver duas páginas, uma utilizando **ReactJS** e outra **sem framework**, que contenham um mapa responsivo e que permita que um consumidor navegue por ele para obter informações a respeito de lojas na região. Os dados das lojas devem ser carregados dinamicamente pelo endpoint <https://munddi.com/dev/pdvs>.

2. Detalhes da implementação

2.1 Mapa

O mapa utilizado é de escolha livre, mas sugerimos a utilização do [google maps](#) ou do [leaflet\(openstreetmaps\)](#).

2.2 Endpoint

O endpoint munddi.com/dev/pdvs em uma requisição https GET requer 4 parâmetros:

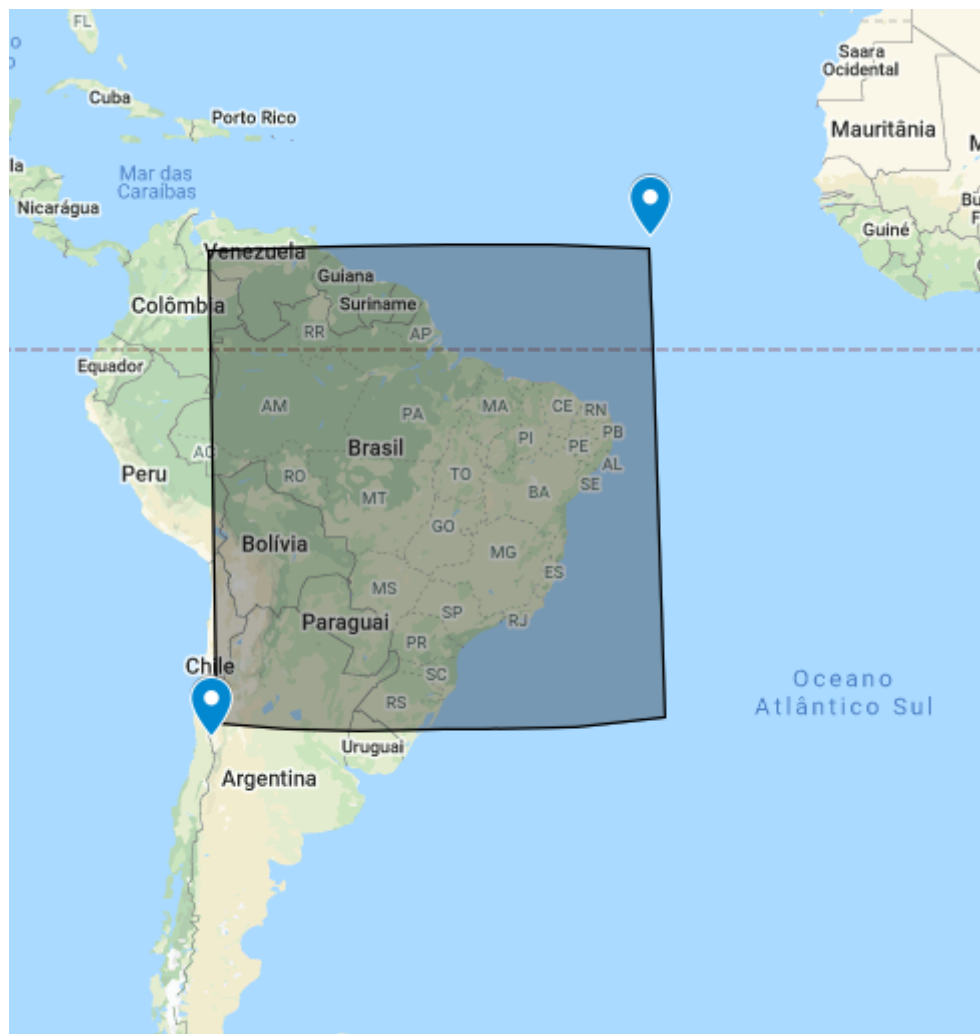
- **ne_lat** :
A latitude do ponto a nordeste do mapa.
- **ne_lng** :
A longitude do ponto a nordeste do mapa.
- **sw_lat** :
A latitude do ponto a sudoeste do mapa.
- **sw_lng** :
A longitude do ponto a sudoeste do mapa.

Ao receber dados válidos, é retornado um json contendo:

- **code** :
Um código indicando o status do request (ex: 200 = válido, 400 = bad request).
- **msg** :
Uma mensagem descritiva a respeito do código.
- **data** :
Um array contendo as informações das lojas contidas dentro dos pontos informados.

Exemplo de chamada:

https://munddi.com/dev/pdvs?ne_lat=10&ne_lng=-30&sw_lat=-30&sw_lng=-70



2.3 Aparência

Excluindo os pré requisitos de ser responsivo e permitir a navegação pelo mapa, a forma com a qual as informações serão apresentadas é de livre escolha.

2.4 Página utilizando ReactJS

Comece por esta tecnologia. Qualquer biblioteca e/ou adicionais podem ser utilizados, porém, elas devem estar devidamente documentadas no package.json e preferencialmente o resultado final deve poder ser observado com um 'npm run dev', mas também serão aceitos os arquivos finais gerados em conjunto com os arquivos source.

2.5 Página sem Framework

JQuery e Bootstrap (ou similares) podem ser utilizados preferencialmente. A ideia aqui é não utilizar Angular, Vue ou ReactJS.

3. O que será avaliado

- a) Organização em relação ao prazo: é melhor entregar algo funcional do que entregar algo com ótimas intenções, mas que não funciona;
- b) Componentização correta para reutilização do código html/css;
- c) Leitura e exibição correta dos dados vindos do endpoint;
- d) Qualidade do código (nome de variáveis, indentação, facilidade de leitura, etc).