

# Desafio Técnico de Front-End

Aqui na Geofusion valorizamos a cultura lean, temos times de desenvolvimento ágeis e com foco em fazer as coisas acontecerem. Estamos continuamente buscando novas tecnologias que melhor atendam as necessidades de nossos produtos. Uma outra busca constante, principalmente para desenvolvedores front-end, é proporcionar uma melhor experiência de uso para nossos usuários.

O que preparamos é um pedacinho do que você encontrará no dia a dia aqui na Geofusion. O desafio técnico consiste em criar uma

para acompanhamento de desempenho de lojas, conforme protótipo e dados disponibilizados. Esse tipo de problema é bastante comum no nosso universo e em várias situações precisamos apresentar as informações tanto em lista quanto em mapa.

Por favor, leia todo este documento com atenção. Os itens marcados como opcionais ficam a seu critério, os demais são obrigatórios.

De posse dos artefatos que lhe foram entregues e seguindo as instruções deste documento, você deve desenvolver:

- Sistema de build simples para a SPA.
- **Mapa:** exibir todas as lojas no mapa. As lojas com faturamento inferior ao mínimo esperado devem estar em vermelho.
- **Listagem paginada:** ao carregar a página, já trazer uma listagem com 10 lojas por página ordenadas pelo nome da loja, seguindo o protótipo. As lojas com faturamento inferior ao configurado no campo “Faturamento mínimo esperado” devem estar em vermelho.
- **Pesquisa:** Filtrar as lojas pelo nome, tanto na lista quanto no mapa. O filtro deve ser aplicado durante a digitação do texto.
- **Faturamento mínimo esperado:** As alterações neste campo devem refletir na listagem e no mapa, deixando os itens com valor igual ou inferior destacados em vermelho. O destaque deve ser aplicado durante a digitação do texto. O valor default desse campo é 15.000,00.
- Permitir reordenar a listagem pelas colunas.
- **[opcional]** Usar mecanismos de *routing* na paginação e filtro.

Artefatos entregues com este documento:

- Protótipo em PNG
- JSON com os dados da aplicação
- Fonts
- Images (ícones e markers)

## Instruções

- Você deve utilizar React JS para construir sua aplicação, gerindo o empacotamento com Webpack (não usar nenhum template/gerador de código como create-react-app ou análogos.)
- Para o mapa, utilizar uma das seguintes bibliotecas:
  - Leaflet: <https://www.npmjs.com/package/leaflet>
  - MapLibre-gl (versão open source do mapbox): <https://www.npmjs.com/package/maplibre-gl>
- Escreva seu próprio código CSS. Não utilize frameworks.
- Versione o projeto de acordo com o seu andamento, comentando os commits. Não deixe para versionar tudo no final.
- Você deve usar os dados do arquivo JSON exatamente como estão.
- Desenvolva pensando em componentização, reuso e performance.
- Deve constar a documentação necessária para executar o projeto (subir ambiente de dev, gerar o dist, instalar dependências, etc).
- A build deve gerar um diretório **./dist** com a versão de produção dos artefatos.
- Não use links externos. Todo artefato usado deve estar no dist após o build.
- O que não estiver descrito neste documento fica a seu critério como resolver.

## Pontos Extras

- Styled Components
- ES6 (transpilers)
- Linter
- Testes Unitários
- CSS responsivo

## Entrega

- Entregar o projeto em repositório **private** no GitLab (<https://gitlab.com>) com permissão de **Maintainer** para os usuários **dario.oliveros**, **jorge.delucca** e **ayumi.chiba**.
- O nome do projeto deve ser:  
**gfn\_frontend\_challenge\_<primeiro-nome>-<ultimo-nome>**
- Após a entrega do projeto, você não deve mais realizar nenhuma alteração.

Boa Sorte :)