**Desafio | Frontend Developer | Wecont**

Olá! Seja bem-vindo ao teste para a vaga de Frontend Developer.

Preparamos um teste que abordará seus conhecimentos em Vue.js. Você será encaixado na vaga de acordo com seu resultado.

A necessidade é desenvolver o projeto "Busca Produtos", onde o usuário pode listar os produtos disponíveis na loja.

**Instruções para entrega**

* Versione, com git, e hospede seu código em algum serviço de sua preferência: github, bitbucket, gitlab ou outro.
* Crie um README com instruções claras sobre como executar sua obra.
* Envie um email com o link do seu repositório para [rh.wec2019@gmail.com](mailto:rh.wec2019@gmail.com)
* Dúvidas podem ser enviadas para o mesmo email acima.

**Layout e API**

Você terá que implementar o *layout* de acordo com esse **[mock](https://xd.adobe.com/spec/aa1c5781-ecac-46c9-7032-b66139998404-dc2d/)**. O *mock* contém a estrutura do layout e todo o estilo da página.

É importante se ater aos detalhes de tamanho e espaçamento entre os elementos da página.

**Link do layout**: <https://xd.adobe.com/spec/aa1c5781-ecac-46c9-7032-b66139998404-dc2d/>

Para preencher a página com as informações dos produtos, você terá que consultar esta [**API**](https://frontend-intern-challenge-api.iurykrieger.now.sh/products?page=1) de forma paginada, sendo que cada página consultada retornará as informações de **8 produtos** e um **link para a próxima página**.

**Endpoint da API**: <https://frontend-intern-challenge-api.iurykrieger.now.sh/products?page=1>

**Requisitos**

* Implemente o HTML/CSS da tela com base no layout disponível em <https://xd.adobe.com/spec/aa1c5781-ecac-46c9-7032-b66139998404-dc2d/>
* O projeto deve ser uma *single page application (SPA)* escrita utilizando **Vue**.
* Para cada **produto** retornado pela API, um **card de produto** com as respectivas informações deve ser criado na grade de produtos.
* Ao clicar no botão **Ainda mais produtos aqui!** a próxima página da API deve ser consultada, gerando mais **8 produtos** na grade existente, abaixo dos produtos já carregados pela primeira requisição.
* O formulário de *newsletter* com o título **Compartilhe a novidade** deve ter seus campos de *input* validados de acordo com o conteúdo (ex: O campo de email deve conter um email válido);
* Ao submeter o formulário, deve-se mostrar um retorno (Sucesso, se os campos passaram na validação. Fracasso, se não) na tela;
* Não é preciso se preocupar com persistência dos dados. É esperado que as informações coletadas e submetidas sejam perdidas com o recarregamento da página (F5).
* Tornar o layout responsivo.

**Requisitos Opcionais**

* Padronização do código: seguir algum styleguide de Javascript e/ou CSS.
* Aplicação de animações.
* E qualquer outra sugestão sua

**Critérios de avaliação**

* Fidelidade ao layout solicitado;
* Fidelidade às funcionalidades solicitadas;
* Componentização e extensibilidade dos componentes Javascript;
* Clareza de nomenclatura do CSS;
* HTML estruturado de forma semântica;
* Requisitos Opcionais

**Não conseguiu fazer tudo?**

A sua forma de priorizar a entrega também vai ser considerada. Documente em um arquivo a parte o que você conseguiu e não conseguiu implementar, descrevendo por que você preferiu priorizar desta maneira.