**Challenge - Developer Frontend**

Não é esperado que todos consigam realizar esse desafio por completo, já que é destinado a todos os níveis de carreira.

Você será avaliado pela sua capacidade de escrever um código simples, de fácil manutenção, e pela quantidade de funcionalidades que você entregar.

**Instruções**

* **Nome do Projeto:** Cantina do Ovelha
* **Objetivo do Projeto:** Criar uma aplicação Web responsiva que consulte nossa API e exiba uma lista de restaurantes e o cardápio de cada um deles.
* **Tecnologia:** React.
* **User Interface:** Você pode [usar esse link](https://xd.adobe.com/spec/f6e71782-ebba-4573-6f7a-005a1a6d391f-80d6/) como referência de UI durante o desenvolvimento.
* **Entregáveis:** Crie um repositório pessoal para esse projeto e passe o acesso para o Ovelha.

**Desafio**

* Consulte a API disponibilizada para buscar as informações.
* Crie uma tela para exibir a lista de restaurantes:
  + O usuário deve ser capaz de buscar por estabelecimento.
  + Indique se cada um deles está aberto ou fechado sem ser necessário recarregar ou reabrir a página.
  + Para cada restaurante, deve ser exibido os horários de funcionamento, as promoções ativas no momento e o cardápio.
  + O restaurante deve atualizar o status de aberto/fechado, de acordo com o horário de funcionamento, sem ser necessário recarregar ou reabrir a página.
* Crie uma tela para exibir os produto do cardápio de cada um dos restaurantes:
  + O usuário deve ser capaz de buscar os produtos.
  + Para os produtos com promoção ativa, deve ser exibido o valor original e o valor promocional.
  + As promoções ativas e o valor promocional devem ser atualizados na interface, de acordo com o horário, sem a necessidade de recarregar ou reabrir a página.

**Formato de horários**

* É necessário tratar os campos que indicam horários de funcionamento.
* Os campos from e to possuem o formato HH:mm.
* Caso o campo to possua um horário anterior ao valor de from, deve-se considerar que o horário se estende até o horário contido em to do próximo dia. Por exemplo, se from for 19:00 e to for 02:00, o horário a ser considerado é das 19h do dia atual até às 02h do dia seguinte.
* O campo days guarda os dias da semana em que o horário é válido. Sendo Domingo o valor 1 e Sábado o valor 7. Os horários possuem intervalo mínimo de 15 minutos.

**O que nós vamos avaliar**

* Você será avaliado pela qualidade do código, legibilidade e pela quantidade de funcionalidades implementadas.
* Você é livre para tomar as decisões técnicas com as quais você se sente mais confortável. Apenas esteja pronto para explicar as razões que fundamentaram suas escolhas =]
* Inclua um arquivo *README* que possua:
  + desafios/problemas com os quais você se deparou durante a execução do projeto.
  + maneiras através das quais você pode melhorar a aplicação, seja em performance, estrutura ou padrões.
  + todas as intruções necessárias para que qualquer pessoa consiga rodar sua aplicação sem maiores problemas.

**Dicas**

* Documente seu projeto em arquivos markdown explicando a estrutura, processo de setup e requisitos.
* Tenha sempre um mindset de usabilidade, acessibilidade e colaboração.
* A organização das branches e os commits no repositório falam muito sobre como você organiza seu trabalho.
* Você pode utilizar bibliotecas de componentes visuais;
* O material de UI/UX que fornecemos deve servir como uma referência, você não precisa necessariamente segui-lo à risca. No entanto, quanto mais próximo, melhor =]
* Os testes unitários são opcionais porém são mais do que desejados.
* O design/estrutura do código da aplicação deve ser *production ready*.
* Use boas práticas de programação.

**FAQ**

**Posso utilizar frameworks/bibliotecas?**

Sim.

**Quanto tempo eu tenho ?**

Quanto mais tempo você demorar, mais críticos seremos na sua avaliação =]

**React, Angular ou Vue?**

Você precisa implementar em React.

**Happy coding**