

# Desafio Software Engineer

## Observações:

- O conteúdo deste desafio não pode ser copiado/divulgado.
- Você deve fazer o upload do arquivo-resposta neste formulário:
- [Formulário para enviar o desafio](#)

## DESAFIO TÉCNICO

Este documento faz parte do Processo Seletivo de Pessoas Desenvolvedoras de Software. A estrutura apresentada segue a definição de um problema, uma demanda alinhada ao problema, definição de entregáveis e questionamentos. Não é obrigatória a entrega de todos os artefatos listados, também não será considerado relevante a stack de tecnologia adotada. O ideal é que desenvolva o desafio da maneira como se sentir confortável.

### Problema

Atualmente, disponibilizamos diversos canais digitais para interação com as pessoas que compõem nosso perfil de clientes. Identificamos a necessidade de unificar a experiência destas pessoas em todos os canais, de maneira mais objetiva, precisamos possibilitar as seguintes ações:

- Iniciar interação em qualquer canal;
- Continuar interação a partir de qualquer canal, ou seja uma interação iniciada no canal A deve ser continuada de maneira fluida no canal B;
- Receber mensagem no canal aberto pelo usuário, mesmo que nenhuma interação tenha sido realizada por ele neste canal.

No entanto, entendemos que para possibilitar esse conceito de Omnichannel faz-se necessário prover uma infraestrutura escalável, que possa suportar a necessidade de todas pessoas que consomem diariamente nossos serviços.

### Desafio

1. Dê um nome para o projeto;

2. Implemente uma única feature, a de criação de contas de usuário, com os campos que julgar necessário, os campos que devem ser utilizados para a identificação de pessoas devem ser decididos por você com base no seu entendimento do problema.

A stack de tecnologia deverá ser obrigatoriamente Laravel para o Backend e Vue.js, React ou Angular para o frontEnd. A solução pode ter quantos projetos, pacotes e/ou bibliotecas você julgar necessário, deve roda em container docker;

Durante o preenchimento dos campos o usuário deve apresentar um endereço válido, para isso é necessário utilizar a API do serviço <https://viacep.com.br>.

#### Regras para a implementação:

2.1 Cadastrar usuário;

2.2 Listar usuário;

2.3 Editar usuário;

2.4 Se o Cep não for do estado do Amazonas não pode se cadastrar;

2.5 Se o usuário tiver menos de 18 anos não pode se cadastrar;

3. Obrigatoriamente o projeto de backend do desafio deverá ter um padrão de projeto baseado nos princípios do DDD utilizando camadas de apresentação, camada de aplicação, camada de infraestrutura e camada de domínio;

4. Crie um repositório no Github e configure para garantir que tenhamos acesso de leitura e clone do código fonte, deixe um passo-a-passo de como executar a solução;

5. Documente o setup para a execução do projeto no formato Markdown, com as informações que julgar relevante;

6. Devolva o caminho para o formulário: [Formulário para enviar o desafio](#)

7. Crie um diagrama utilizando a notação C4 Model (<https://c4model.com/>) para

apresentar a visão arquitetural da infraestrutura necessária para suportar este produto. Considere infraestrutura os recursos de apoio à solução, como banco de dados, loggers, cache, serviços de email, serviços de apoio à solução;

8. Crie um diagrama de Design da Solução, Utilizando a notação C4 Model, para apresentar a organização do software, as responsabilidades de cada elemento identificado e o relacionamento entre eles, por exemplo: se existem camadas de abstração, qual a responsabilidade cada camada e se existem pacotes dentro destas camadas, o que eles representam