

TESTE PARA DESENVOLVEDOR



FULLSTACK PHP / LARAVEL

Avaliação prática



Bem-vindo, futuro Dragon Maker!

Se você chegou até aqui, é porque identificamos em você o potencial para que venha a fazer parte do nosso time de desenvolvimento. Parabéns!

O Teste

Queremos que nos mostre suas habilidades de desenvolvimento através da entrega de um sistema de cadastro de contatos simples com endereço integrado ao **Via Cep** e **Google Maps**.

Requisitos

Para entregar este desafio você precisa:

- Ter conhecimento em linguagem **PHP** com framework **Laravel**.
- Saber utilizar algum **Banco de Dados Relacional**.
- Saber como funcionam **APIs REST/RESTful**.
- Conhecimento em versionamento de código com **GIT**.
- Conhecer qualquer framework para desenvolvimento do **Frontend**.

Escopo de negócio

The screenshot shows a user interface for managing contacts. On the left, a sidebar lists contacts under various categories and status filters. On the right, a map of Brazil displays the locations of these contacts as red pins.

Vamos desenvolver uma aplicação de lista de contatos.

Nesta aplicação o usuário pode:

- Se cadastrar para utilizar a plataforma.
- Recuperar a sua senha.
- Realizar login e logout.
- Gerenciar sua lista de contatos.
- Realizar pesquisa de endereços como ajuda ao cadastro de contatos.
- Excluir a sua própria conta.



Jornada do usuário

O usuário acessa a plataforma, realiza seu cadastro e em seguida faz seu login, caso não lembre a senha cadastrada pode recuperá-la através de seu e-mail.

Assim que estiver dentro do sistema, os dados dos contatos previamente cadastrados são exibidos na tela e então o usuário realiza o cadastro de um ou mais contatos utilizando um formulário contendo os campos necessários para o cadastro.

A plataforma possui um sistema de ajuda para o preenchimento do endereço do contato, onde o usuário pode informar alguns dados tais como, UF, cidade e um trecho do endereço e esse sistema de ajuda apresenta então as possibilidades de endereço baseado na pesquisa, dessa forma o usuário escolhe na lista qual o endereço lhe convém e tem os campos do formulário correspondente preenchidos automaticamente.

Quando o usuário quer localizar um contato na lista, ele utiliza um filtro de texto que traz apenas os contatos que contém o nome ou CPF equivalente ao termo pesquisado.

Sempre que o usuário clica no contato da lista, o mapa deve centralizar e marcar com um “pin” a coordenada geográfica obtida através do cadastro.



O usuário pode realizar a exclusão e a edição dos dados dos contatos a qualquer momento.

Se desejar, o usuário pode remover a sua própria conta, o que faz com que todos os dados cadastrados por ele sejam removidos da base de dados.

Escopo técnico

Design System

As interfaces do sistema devem ser implementadas respeitando o Material Design V2 ou V3

Sign In, Sign up e Autenticação

- Permitir que o usuário se cadastre no sistema;
 - É permitido um usuário por e-mail cadastrado.
 - Recuperação de senha através do e-mail.
- Autenticação padrão com login e senha

Autorização.

Os endpoints de gerenciamento de contatos, pesquisa de endereço Via Cep e gerenciamento de conta não devem ser acessíveis sem que haja autenticação.

Dados dos contatos

Os contatos devem conter pelo menos nome, CPF, telefone, endereço completo com o CEP e posição geográfica (latitude e longitude).



Cadastro de contatos

O cadastro deve apresentar mensagens de feedback ao usuário e respeitar as regras de validação.

- CPF deve ser validado segundo o algoritmo de validação oficial.
- Não é permitido CPF duplicado na base (por usuário).
- Apenas o complemento do endereço pode ser omitido no cadastro.

Lista de Contatos

A lista deve permitir pelo menos filtro por CPF ou nome do contato e deve ser ordenada por padrão em ordem alfabética crescente.

Tanto a ordenação quanto à paginação pode ser controlada pelas requisições do Frontend.

Exclusão de conta

O usuário deve informar a sua senha para excluir a conta e caso a senha seja inválida o sistema deve recusar a exclusão.

Recomendações

Gerais

A avaliação será feita com base na entrega do escopo técnico, sendo assim, recursos adicionais serão considerados apenas se todos os recursos obrigatórios estiverem disponíveis na aplicação.

Código Fonte

O que queremos ver são boas práticas de programação, código limpo, comentado e reutilizável.

Entregas

Testes

Implementar testes de unidade ou testes de integração que garantam as principais regras de negócio propostas.

Recursos do sistema

Esperamos que todas as funcionalidades sejam entregues sem “hardcode”, ou seja, a conta criada deve permitir fazer login e recuperar a senha, os dados dos contatos não podem ser uma lista estática ou em memória etc.

Integração Via CEP

Devido à característica da nossa aplicação de teste, o Frontend não pode acessar os endpoints de consulta à Via Cep diretamente, o que torna obrigatório a implementação de um endpoint no nosso Backend para intermediar essa consulta.



Integração Google Maps.

Deve ser utilizado para obter a latitude e longitude do endereço do contato quando o cadastro é realizado.

Sua postura profissional

Mesmo que você sinta dificuldades, não deixe o desespero tomar conta da situação.

Ainda que você não consiga entregar o que foi proposto, seu esforço em tentar fazer será levado muito em consideração, por isso, **capriche no código fonte e não tenha vergonha.**

Quando terminar o teste, envie o link do repositório GIT por e-mail ao recrutador e em seguida agendaremos um papo de máximo uma hora para que apresente em sua máquina o sistema rodando e também para que possamos discutir tecnicamente a implementação.

Boa sorte!