



## Descrição do Teste

O teste consiste na criação de uma aplicação básica de gerenciamento de visitas, em que o usuário deve ser capaz de consultar, concluir e criar visitas. A seguir serão descritos os recursos, observações e requisitos que devem ser cumpridos.

## Considerações

O desenvolvimento deve partir da base de código enviada, que já possui o container docker configurado. Você encontrará instruções mais detalhadas de como executar a aplicação no arquivo README.md.

Cada requisito será avaliado em partes, então caso não saiba ou tenha dúvida sobre como resolver o problema proposto, envie da maneira que achar que está correto e iremos analisar a sua proposta de resolução.

**É expressamente proibido o uso de cópia ou geração via inteligência artificial dos códigos. Caso seja identificado, o teste será imediatamente desclassificado.**

Para remeter o seu teste, publique as alterações em um repositório público de sua preferência (GitHub, BitBucket, GitLab, etc.) e nos envie o link no email [eder@mundowap.com.br](mailto:eder@mundowap.com.br) com o assunto “Teste para vaga de desenvolvedor front-end”.



## Sumário

<b>Descrição do Teste.....</b>	<b>1</b>
<b>Considerações.....</b>	<b>1</b>
<b>Sumário.....</b>	<b>2</b>
<b>Recursos para desenvolvimento.....</b>	<b>3</b>
<b>Requisitos.....</b>	<b>3</b>
Conteúdo da visita.....	3
Cabeçalho.....	3
Criação e Edição.....	4
Agrupamento de visitas.....	5
Lista de visitas.....	6
<b>Avaliação.....</b>	<b>7</b>
Critérios de avaliação.....	7
Diferenciais.....	7



## Recursos para desenvolvimento

O projeto deverá ser desenvolvido utilizando os seguintes recursos:

- Utilize o pacote React Hook Forms para fazer o gerenciamento do formulário;
- O design do app é livre, porém, toda e qualquer forma de estilização deverá ser feita com Styled Components;
- Todas as requisições do app deverão ser feitas via Axios;
- É **obrigatório** o uso de TypeScript no projeto, tipando o máximo de itens possíveis no app;

## Requisitos

### Conteúdo da visita

Cada visita da lista deve conter:

- Data da visita;
- Status (pendente ou concluída);
- Quantidade de formulários;
- Quantidade de produtos;
- CEP;
- Cidade;
- UF (MG, SP, RJ, etc.);
- Logradouro;
- Bairro;
- Número;

### Cabeçalho

O cabeçalho da aplicação, deve conter um botão que irá abrir um modal para que o usuário possa criar uma nova visita pendente.



## Criação e Edição

O modal de criação e de edição deverá conter os seguintes itens:

- Um campo de data para a data da visita;
- Um campo numérico para a quantidade de formulários;
- Um campo numérico para a quantidade de produtos;
- Um campo de texto para CEP
  - Qualquer alteração no CEP deve limpar os demais valores do formulário referentes ao endereço e desabilitar os campos de bairro e logradouro;
  - Após o usuário digitar um CEP válido, o mesmo deverá ser consultado na API [viacep.com.br](http://viacep.com.br);
  - Durante a requisição todos os campos referentes ao endereço devem estar desabilitados;
  - Após o retorno da requisição, deverão ser setados os valores de UF (uf), cidade (localidade), bairro (bairro) e logradouro (logradouro);
  - Caso o retorno da requisição não contenha os valores de bairro e logradouro, seus respectivos campos deverão ser habilitados;
- Um campo de texto para UF apenas para leitura
- Um campo de texto para cidade apenas para leitura
- Um campo de texto para bairro
- Um campo de texto para logradouro
- Um campo numérico para número
- Um botão para cancelar a criação/edição e fechar o modal;
- Um botão para concluir a criação/edição;
  - Este botão deve estar habilitado somente se todos os campos estiverem preenchidos;

A conclusão da criação/edição deve falhar se a data definida não possuir horas suficientes para a visita de acordo com as seguintes regras:

- Cada data deve ter um limite máximo de 8 horas no total;
- Cada formulário da visita deve consumir quinze minutos;
- Cada produto da visita deve consumir cinco minutos;



## Agrupamento de visitas

Deverá ser apresentada uma lista das visitas já cadastradas agrupadas pela data da visita no seguinte formato (Ter. 28/08/2023).

Cada agrupamento deverá possuir uma forma de demonstrar visualmente o percentual de minutos agendados em relação à quantidade de minutos máxima por data (8 horas).

Cada agrupamento deverá possuir uma forma de demonstrar visualmente o percentual de minutos concluídos em relação à quantidade de minutos total. Caso este percentual seja abaixo de 60%, essa demonstração deve possuir algum elemento na cor **VERMELHA**, caso este percentual seja acima de 90%, **VERDE**, em qualquer outra situação, **AZUL**.

O agrupamento deve possuir um botão que fará o fechamento desta data, ao clicar, todas as visitas pendentes desta data devem ser movidas para o início do próximo dia, caso este não comporte todas as visitas devido ao limite de horas, as últimas visitas desta data devem ser realocadas para o início da próxima data, e assim por diante.



## Lista de visitas

Dentro de cada agrupamento, deverá ser exibida a lista de visitas, onde cada item deverá seguir o seguinte padrão:

- Quantidade de formulários;
- Quantidade de produtos;
- Duração estimada da visita;
- Logradouro número e CEP no formato:
  - Logradouro, Número - CEP
- Bairro, Cidade e UF no formato:
  - Bairro, Cidade - UF
- Alguma forma visual que identifique se a visita está pendente ou concluída;
- Caso a visita esteja pendente:
  - Um botão que abrirá o modal para edição a visita;
  - Um botão para alterar o status da visita para concluída;



## Avaliação

### Critérios de avaliação

- Cumprimento dos requisitos;
- Estruturação do projeto;
- Organização e legibilidade do código;
- Performance;
- Divisão dos componentes;
- Tipagens;
- Lógica;

### Diferenciais

- Boa estilização dos componentes;
- Comentários no código;
- Utilização de recursos avançados do React (como ContextAPI e Portals, por exemplo) e das bibliotecas propostas nos tópicos descritos anteriormente;
- O app não se limita só ao exigido, caso esteja confortável com a adição de mais elementos, recursos e tratativas (como loadings, transições, animações, filtros, etc.), estes serão muito bem vindos e serão tratados como um bom diferencial na hora da avaliação.
- Implementar uma maneira de persistir a lista de visitas, de modo que ao recarregar a página os dados não sejam perdidos.

*Agora é botar a mão na massa e dar o seu melhor. Boa sorte, candidato(a)!*

*Surgiu alguma dúvida? Entre em contato pelo e-mail [eder@mundowap.com.br](mailto:eder@mundowap.com.br)*