



Desafio: Desenvolva tela funcional com consumo de API



Desafio



Arquivo do desafio:

▼ Links do desafio:

- [Link do Figma](#)
- [Link do Bootstrap](#)
- [Link para consumir o CEP](#)
- [Link para consumir previsão do tempo](#)

Projete uma interface de usuário seguindo um protótipo do Figma, desenvolvendo telas focados no design e experiência do usuário, fornecendo dados importante sobre endereço e previsão do tempo através de uma API. Este projeto visa criar uma ferramenta útil para o dia a dia dos usuários que trabalham com informações deste tipo.

Contexto

Você acaba de ser recrutado como o mais novo desenvolvedor em uma startup inovadora chamada "DncWeather". Esta startup está revolucionando a forma como as empresas gerenciam as informações internas e externas. Sua primeira missão na "DncWeather" é desenvolver um sistema interno para os funcionários, que os ajudará a acessar informações críticas sobre endereços e previsões meteorológicas de maneira eficiente.

A "DncWeather" acredita que a acessibilidade e a simplicidade são fundamentais para o sucesso de qualquer sistema interno. Portanto, a sua tarefa inicial é criar uma interface de usuário atraente e funcional que permitirá aos funcionários inserir facilmente as informações necessárias para obter dados de endereço e previsões meteorológicas. Este será o primeiro passo na construção do sistema que irá transformar a experiência dos funcionários na empresa.

Como começar?

Sua jornada começa com o desenvolvimento da tela de informações. Nesta etapa inicial, você será responsável por criar uma página da web que permitirá aos usuários inserir as informações necessárias para obter dados de endereço e previsão do tempo. Isso servirá como a base do sistema interno que você está construindo. Portanto, comece projetando e implementando a tela de entrada de informações.



Etapas de Desenvolvimento

Etapa 01) Desenvolvendo as telas

Nesta primeira etapa, você irá começar a construir o frontend da aplicação seguindo o protótipo que foi criado no Figma. O foco será na implementação da

tela de informações, onde os usuários fornecerão os dados necessários para acessar as informações desejadas.

1. **Desenvolvimento Baseado no Protótipo:** Utilize o protótipo disponível no Figma como referência precisa para a criação da interface. Isso inclui o layout, elementos gráficos, posicionamento de campos e botões, cores e estilos. Garanta que a interface seja fiel ao design apresentado no protótipo.
2. **Implementação da Tela de Informações:** Concentre-se na criação da tela de informações, onde os usuários poderão inserir os dados necessários, como nome, endereço e cidade. Certifique-se de criar todos os campos relevantes de acordo com as especificações do protótipo.
3. **Validação de Campos:** Implemente validações de entrada de dados para garantir que todos os campos obrigatórios sejam preenchidos adequadamente e que os dados inseridos estejam no formato correto. Forneça feedback claro aos usuários caso haja problemas de validação.
4. **Utilização do Bootstrap (Opcional):** Você tem a opção de utilizar o framework Bootstrap para facilitar o desenvolvimento do design da aplicação. O Bootstrap oferece componentes e estilos predefinidos que podem acelerar o processo de desenvolvimento. No entanto, seu uso não é obrigatório e fica a critério do aluno decidir se deseja utilizá-lo.



Links: [Figma](#) e [Bootstrap](#)

Etapa 02) Construindo a integração

Na segunda etapa, você irá desenvolver duas funcionalidades separadas para consumir dados: uma para obter informações de endereço e outra para obter dados de previsão do tempo. Siga os passos abaixo desenvolver essa etapa:

1. **Desenvolvimento dos Formulários:** Comece desenvolvendo os formulários em HTML para que os usuários possam inserir os dados necessários. Certifique-se de seguir o layout e os elementos gráficos conforme definido no protótipo do Figma.

2. **Funcionalidade de Consumo de Dados:** Desenvolva as funcionalidades de consumo de dados utilizando JavaScript. Você precisará criar duas funções separadas: uma para consumir os dados do CEP e outra para consumir os dados da previsão do tempo.
3. **Utilização de Práticas Assíncronas:** Utilize as práticas `async` e `await` para criar essas funções assíncronas. Isso garantirá um comportamento eficaz na tela que irá exibir os dados, permitindo que a aplicação continue funcionando de forma responsiva enquanto aguarda a resposta da API.



Dica: utilize as seguintes APIs para obter os dados:

- Captar os dados de endereço:

<https://viacep.com.br/>

- Captar os dados de previsão do tempo:

<https://open-meteo.com/en/docs/>

Etapa 03) Consumindo a API

Nesta etapa, sua missão é desenvolver as duas funcionalidades necessárias utilizando JavaScript para consumir as API's. Para conseguir realizar o desenvolvimento dessas funcionalidades siga esses passos:

1. **Declaração de Variáveis:** Inicie declarando variáveis para armazenar os dados fornecidas pelo usuário. Esses valores serão utilizados para realizar a consulta na API's de cep e previsão do tempo.
2. **Chamada da API:** Utilize o método `fetch` para fazer uma requisição na API's.
3. **Atualização da Interface:** Utilize os dados obtidos da API para atualizar dinamicamente a interface do usuário (DOM) com as informações relevantes. Por exemplo, você pode exibir a temperatura atual da região em um elemento HTML específico.



Dica: Para consumir a informação da temperatura da região, utilize o campo de **temperature_2m** com o índice 0 dentro da API.

Etapa 04) Testando as informações

Nesta etapa última etapa do desafio, você será responsável por garantir a qualidade e a confiabilidade das APIs que você desenvolveu nas etapas anteriores. É essencial que as APIs funcionem corretamente e forneçam dados precisos. Para isso, siga as seguintes diretrizes

- Verifique se todas as informações solicitadas pelos usuários estão sendo recebidas corretamente.
- Certifique-se de que a API está respondendo com um código HTTP 200 quando as solicitações são feitas.
- Certifique-se que os dados entregue das API's, estão de acordo com os dados enviados para ela.
- Garanta que os dados estejam sendo armazenados nos campos apropriados da aplicação, de acordo com a estrutura definida nas etapas anteriores.



Critérios de Avaliação:

Os critérios de avaliação são as conclusões das etapas propostas conforme os critérios abaixo.

Lembre-se de verificar os seguintes pontos em cada etapa conforme proposto abaixo.

Critérios	Atendeu às Especificações	Pontos
Design semelhante ao Figma	Certifique-se que o seu design apresente uma coerência e consistência nas cores e na tipografia utilizadas conforme a interface do Figma.	25
Desenvolvimento das funcionalidades	Desenvolva o consumo de API dividido em duas funcionalidades separadas, uma para cada consumo de API.	25
Integração entre os dados	Garanta que os dois consumos de APIs estejam trazendo os dados corretos e que estejam	25

	sendo exibidos no frontend de acordo com o design.	
Testando o consumo da API	Garanta que as APIs tenham tratamento de erro adequado, caso ocorra algum problema durante o consumo.	25



Entrega



Como entregar: Envie o link do repositório do github, e o read me deverá conter o link do Netlify que será hospedado sua landing page.

▼ Exemplo:

