



Desafio de Desenvolvimento Frontend

Você está encarregado de criar uma aplicação frontend utilizando Angular ou ReactJS que consuma uma API especificada abaixo e exiba esses dados de forma dinâmica em uma interface amigável e responsiva. Aqui estão os detalhes do desafio:

1. Configuração do Projeto:
 - a. Crie um novo projeto Angular ou ReactJS, dependendo da sua escolha e familiaridade.
 - b. Instale as dependências necessárias para fazer requisições HTTP para a API.
2. Consumo da API:
 - a. Consuma a API <https://jsonplaceholder.typicode.com/photos/> que vai retornar 5000 registros na estrutura:

```
[{  
  "albumId": 1,  
  "id": 1,  
  "title": "accusamus beatae ad facilis cum similique qui sunt",  
  "url": "https://via.placeholder.com/600/92c952",  
  "thumbnailUrl": "https://via.placeholder.com/150/92c952"  
},  
...]
```

- b. Use as ferramentas de requisição HTTP disponíveis na respectiva biblioteca/framework para fazer requisições para a API e receber os dados.
3. Exibição dos Dados:
 - a. Apresente os dados obtidos da API de forma dinâmica na interface do usuário.
 - b. Use componentes (no caso do Angular) ou componentes de função e hooks (no caso do ReactJS) para organizar e exibir os diferentes conjuntos de dados, como lista de itens, detalhes individuais etc.
 - c. Considere a utilização de funcionalidades nativas como map e filter (no caso do ReactJS) ou diretivas estruturais como *ngFor, *ngIf (no caso do Angular) para exibir os dados de forma iterativa e condicional.
4. Funcionalidades Adicionais:
 - a. Implemente ordenação dos dados (por exemplo, ordenação por nome, preço etc.).
 - b. Adicione paginação para lidar com grandes conjuntos de dados.

- c. Permita a interação do usuário, como adicionar itens ao carrinho (se aplicável), fazer comentários em posts etc.
- 5. Design e Responsividade:
 - a. Desenvolva uma interface atraente e responsiva utilizando CSS e/ou bibliotecas de componentes como Angular Material (no caso do Angular) ou Material-UI (no caso do ReactJS).
 - b. Certifique-se de que a aplicação seja amigável em diferentes dispositivos e tamanhos de tela.
- 6. Documentação:
 - a. Forneça instruções claras de como executar e testar o projeto.
 - b. Documente o código de forma adequada, explicando a estrutura, as principais funcionalidades e decisões de design.
- 7. Entrega:
 - a. URL onde a aplicação desenvolvida foi hospedada, o candidato deve hospedar a aplicação desenvolvida em algum servidor na internet e disponibilizar a URL onde a API desenvolvida pode ser chamada.
 - b. URL onde o código fonte pode ser consultado: GitHub, Bitbucket, Gitlab, ou qualquer outro repositório de código público.
 - c. Informamos que a aplicação pode ser hospedada em qualquer ambiente público na internet (lembramos que diversos provedores, inclusive de nuvem onde o candidato pode hospedar a aplicação de forma gratuita por um período. Exemplos de provedores AWS, Digital Ocean etc).

Este desafio permitirá que você demonstre suas habilidades em desenvolvimento frontend com Angular ou ReactJS, incluindo o consumo de APIs, manipulação de dados, criação de interfaces responsivas e boas práticas de codificação. Boa sorte!