

Desenvolvimento de Formulário em React com Validação e Integração de API

Por Escola Dnc

Introdução

Este ebook aborda o desenvolvimento de um formulário em React, focando na implementação de validação de campos e integração com uma API externa. Vamos explorar as melhores práticas para criar um formulário funcional, seguro e com uma boa experiência do usuário.

Configuração Inicial do Formulário

Estrutura Básica

O formulário em questão é composto por campos para nome, e-mail e mensagem. A estrutura inicial inclui:

- Um componente React para o formulário
- Estados para armazenar os valores dos campos
- Uma função de manipulação para atualizar os estados

Validação Inicial

A validação inicial é implementada para garantir que os campos obrigatórios sejam preenchidos corretamente antes do envio.

- Validação de e-mail: Verifica se o formato do e-mail é válido
- Desabilitação do botão: O botão de envio fica desabilitado até que todos os campos obrigatórios estejam preenchidos corretamente

Implementação da Função de Submit

Prevenção do Comportamento Padrão

Para evitar o comportamento padrão de recarregamento da página ao submeter o formulário HTML, implementamos:

- Uso da função `preventDefault()` no evento de submit
- Criação de uma função `handleSubmit` personalizada

Estados de Controle

Foram criados estados adicionais para melhorar o controle e a experiência do usuário:

1. `formSubmitLoading` : Controla o estado de carregamento durante o envio
2. `formSubmitted` : Indica se o formulário foi enviado com sucesso

Integração com API Externa

Configuração da Requisição

A integração com a API externa (Web3 Forms) é realizada utilizando o método `fetch`. Os principais pontos são:

- URL da API: `https://api.web3forms.com/submit`
- Método HTTP: POST
- Headers: Configuração do tipo de conteúdo como JSON
- Body: Dados do formulário em formato JSON

Tratamento de Respostas e Erros

O tratamento adequado de respostas e erros é crucial para fornecer feedback ao usuário e garantir a robustez da aplicação.

- Uso de estrutura `try/catch` para capturar erros
- Verificação do status da resposta (`response.ok`)
- Exibição de mensagens de sucesso ou erro para o usuário

Melhorias de Usabilidade

Feedback Visual

Para melhorar a experiência do usuário, implementamos:

- Desabilitação do botão de envio durante o processo de submissão
- Exibição de uma mensagem de sucesso após o envio bem-sucedido

Prevenção de Múltiplos Envios

Para evitar problemas com envios múltiplos:

- O botão de envio é desabilitado enquanto a requisição está em andamento
- Um estado de loading é utilizado para controlar a interface durante o envio

Boas Práticas e Considerações Finais

Organização do Código

- Separação de responsabilidades: validação, submissão e feedback ao usuário
- Uso de estados React para gerenciar o ciclo de vida do formulário

Segurança

- Validação tanto no cliente quanto no servidor
- Uso de HTTPS para comunicação segura com a API

Escalabilidade

- O código foi estruturado de forma a facilitar futuras adições ou modificações
- Considerações sobre a possibilidade de mover a lógica de API para um serviço separado em projetos maiores

Conclusão

A implementação de um formulário em React com validação e integração de API requer atenção a diversos aspectos, desde a estrutura básica até o tratamento de erros e feedback ao usuário. Seguindo as práticas apresentadas neste ebook, é possível criar formulários robustos, seguros e com uma excelente experiência do usuário.

