

UNIME – LAURO DE FREITAS  
BACHARELADO EM SISTEMAS DE INFORMAÇÃO

**JEAN CARLOS ROCHA NASCIMENTO**

**DESENVOLVIMENTO COM REACT: FUNDAMENTOS DO REACT  
COMPONENTES E EVENTOS**

Lauro de Freitas  
2024

**JEAN CARLOS ROCHA NASCIMENTO**

**DESENVOLVIMENTO COM REACT**  
**FUNDAMENTOS DO REACT COMPONENTES E EVENTOS**

Trabalho apresentado à disciplina  
Desenvolvimento com React do Curso  
Bacharel em Sistemas de Informação da  
UNIME - Campus Lauro de Freitas, como  
requisito parcial para a obtenção de nota.

Orientadores:

Prof. Gian Carlo Decarli  
Prof. Gian Mauricio Fritsche

Lauro de Freitas

2024

## RESUMO

Este trabalho aborda o desenvolvimento de um formulário em React, com ênfase na validação de e-mails, interação do usuário e gerenciamento eficiente do estado utilizando as bibliotecas Formik e Yup . O objetivo principal é criar uma interface que garanta a inserção de dados válidos, promovendo uma experiência fluida e responsável para os usuários.

Para atingir esse objetivo, foram adotados procedimentos que incluem a implementação de um esquema de validação com a biblioteca Yup , que garantem que os e-mails inseridos sigam um formato adequado e que o campo seja obrigatório. A interação da interface foi projetada para ser intuitiva, incorporando feedback em tempo real por meio de mensagens de erro dinâmicas e indicadores que ajudam a esclarecer problemas nas entradas. O gerenciamento do estado do formulário foi realizado com a ajuda do Formik , que automatiza a manipulação de eventos e a validação integrada, garantindo que as entradas atendam aos critérios definidos antes do envio.

Os resultados mais relevantes indicam que a validação e o feedback instantâneo melhoraram significativamente a usabilidade do formulário, reduzindo a taxa de erro e aumentando a confiança do usuário. Conclui-se que a combinação das bibliotecas Formik e Yup resulta em uma solução eficaz para a criação de formulários interativos em aplicações React, alinhadas às melhores práticas de desenvolvimento web.

Sugere-se que futuras implementações considerem a adição de mais campos de entrada e melhorias na experiência do usuário, como animações e transições mais suaves, para enriquecer ainda mais a interação.

**Palavras-chave:** Reagir. Formik. Sim. Validação de E-mail. Interação do Usuário.

## ABSTRACT

This work addresses the development of a form in React, with emphasis on email validation, user interaction and efficient state management using the Formik and Yup libraries. The main objective is to create an interface that guarantees the insertion of valid data, promoting a fluid and responsible experience for users.

To achieve this objective, procedures were adopted that include the implementation of a validation scheme with the Yup library, which ensures that the inserted emails follow an appropriate format and that the field is mandatory. The interface interaction was designed to be intuitive, incorporating real-time feedback through dynamic error messages and indicators that help clarify problems in the inputs. The form state management was carried out with the help of Formik, which automates event handling and integrated validation, ensuring that the inputs meet the defined criteria before submission.

The most relevant results indicate that validation and instant feedback significantly improved the usability of the form, reducing the error rate and increasing user confidence. It is concluded that the combination of the Formik and Yup libraries results in an effective solution for creating interactive forms in React applications, aligned with the best web development practices.

It is suggested that future implementations consider adding more input fields and improvements in the user experience, such as smoother animations and transitions, to further enrich the interaction.

**Keywords:** React. Formik. Yup. Email Validation. User Interaction.

## SUMÁRIO

<b>1- INTRODUÇÃO .....</b>	<b>6</b>
<b>2- VALIDAÇÃO DE EMAIL .....</b>	<b>7</b>
<b>3- INTERAÇÃO DO USUÁRIO COM O FORMULÁRIO. ....</b>	<b>8</b>
<b>4- GERENCIAMENTO DO ESTADO DO FORMULÁRIO COM FORMIK.....</b>	<b>9</b>
<b>5- MELHORIAS .....</b>	<b>10</b>
<b>6- CONCLUSÃO.....</b>	<b>11</b>

## 1- INTRODUÇÃO

A presente documentação tem como objetivo descrever as funcionalidades inovadoras em um formulário React, desenvolvido com foco na validação de e-mails, interação do usuário e gerenciamento eficiente do estado do formulário. A coleta de informações precisas e confiáveis é um aspecto crítico em qualquer aplicação web, e, nesse contexto, a validação de dados se torna uma necessidade primordial. O projeto visa não apenas garantir a integridade dos dados coletados, mas também fornecer uma experiência de usuário intuitiva e responsiva.

Para a realização deste projeto, foram utilizadas as bibliotecas **Formik** e **Yup**, reconhecidas por sua eficácia em lidar com formulários em aplicações React. O **Formik** simplifica o gerenciamento de estado e a manipulação de eventos, enquanto o **Yup** oferece um robusto mecanismo de validação, permitindo a definição de critérios claros para a entrada de dados. Essas ferramentas, quando integradas, permitem criar uma interface que não é apenas válida, mas também comunica de forma clara e eficaz quaisquer problemas com as entradas do usuário.

Além de abordar a validação de e-mails, o formulário foi projetado para incorporar feedback em tempo real, melhorando a interação do usuário. Por meio de mensagens de erro dinâmicas e indicadores visuais, o sistema orienta os usuários durante o preenchimento, facilitando a correção de entradas inválidas. Este documento detalhará cada uma dessas funcionalidades, destacando sua importância e a implementação técnica, assim como os benefícios que trazem para a experiência do usuário em aplicações web modernas.

## 2- VALIDAÇÃO DE EMAIL

A validação de e-mail é uma característica central do formulário, pois garante que os dados inseridos pelos usuários atendam a critérios específicos de formatação e obrigatoriedade. Essa funcionalidade é essencial para a coleta de informações precisas e seguras, garantindo que os dados sejam adequados para processamento posterior. As principais características da validação são:

- **Formato Válido:** O formulário garante que o e-mail inserido siga um padrão aceitável, como o formato típico de endereço eletrônico (ex.: nome@dominio.com). Utilizando expressões regulares, a validação se a estrutura do e-mail está correta, levando em consideração aspectos como a presença do símbolo "@" e a terminação de domínio. Em caso de não conformidade, uma mensagem de erro específica será exibida ao usuário, informando a natureza do erro e orientando-o a corrigir a entrada.
- **Campo Obrigatório:** O campo de e-mail é configurado como obrigatório. Isso significa que, ao tentar enviar o formulário sem preencher esse campo, o usuário receberá uma mensagem indicativa, alertando-o de que a entrada é necessária. Esse recurso é crucial para garantir que não sejam enviados formulários vazios, o que poderia comprometer a qualidade dos dados coletados.
- **Mensagens de erro:** As mensagens de erro são dinâmicas e exibidas em tempo real, permitindo que o usuário receba feedback imediato sobre a validade das informações inseridas. Essa abordagem interativa não só melhora a usabilidade do formulário, mas também reduz a frustração do usuário ao preencher o formulário, pois ele pode corrigir os erros à medida que surgem. Além disso, as mensagens são projetadas para serem claras e concisas, facilitando a compreensão do problema e as ações corretivas possíveis.

Essa combinação de validação de formato, obrigatoriedade de campo e feedback dinâmico contribui significativamente para a criação de um formulário robusto e amigável, que prioriza a experiência do usuário enquanto garante a integridade dos dados obtidos.

### 3- INTERAÇÃO DO USUÁRIO COM O FORMULÁRIO

A interação do usuário com o formulário foi projetada para ser intuitiva e responsiva, garantindo uma experiência fluida e agradável. Ao priorizar a usabilidade, o formulário busca minimizar frustrações e otimizar o processo de entrada de dados. Os principais aspectos dessa interação incluem:

- **Feedback em Tempo Real:** À medida que o usuário preenche o campo de e-mail, a validação é aplicada instantaneamente. Essa abordagem fornece um retorno imediato sobre a validade da entrada, permitindo que os usuários se identifiquem e corrijam problemas rapidamente. O feedback em tempo real não apenas aumenta a eficiência do preenchimento, mas também contribui para uma sensação de controle e confiança durante o uso do formulário.
- **Exibição de Mensagens de Erro:** Mensagens de erro são apresentadas logo abaixo do campo de entrada em duas situações principais: quando o campo fica vazio durante a tentativa de envio do formulário e quando o formato do e-mail não atende aos critérios estabelecidos. Essas mensagens são cuidadosamente redigidas para serem claras e informativas, ajudando o usuário a entender a natureza do erro e a agir para corrigi-lo. Esse aspecto é crucial para a manutenção da usabilidade e para garantir que o usuário saiba exatamente o que precisa ser ajustado.
- **Indicadores Visuais:** Além das mensagens de erro textuais, o campo de e-mail pode mudar de cor, como uma borda vermelha, para sinalizar visualmente que há um problema. Esses indicadores visuais são uma ferramenta poderosa para captar a atenção do usuário de forma imediata, facilitando a identificação de erros sem que o usuário precise ler mensagens de texto. Essa abordagem multissensorial melhora ainda mais a experiência do usuário, tornando o formulário mais acessível e menos sujeito a erros.
- **Transições Suaves e Animações:** Para enriquecer a interação, pequenas animações podem ser incorporadas ao feedback visual e às mensagens de erro. Por exemplo, quando o usuário corrige uma entrada



inválida, uma animação sutil pode indicar que o campo voltou a um estado válido. Essas transições suaves não apenas tornam a experiência mais agradável, mas também ajudam a comunicar mudanças de estado de maneira eficaz.

Esses elementos combinados criam um ambiente de entrada de dados que não é apenas funcional, mas também envolvente, resultando em um formulário que promove uma interação positiva e intuitiva para todos os usuários.

#### 4- GERENCIAMENTO DO ESTADO DO FORMULÁRIO COM FORMIK

- O gerenciamento do estado do formulário é realizado por meio da biblioteca Formik, que simplifica o controle de entradas e o envio de dados. Essa abordagem fornece uma maneira eficiente de lidar com a complexidade que pode surgir ao trabalhar com formulários em aplicações React. Os aspectos principais desse gerenciamento são:
- **Valores Iniciais:** O estado inicial do formulário é definido como um objeto que contém as entradas, começando com um campo de e-mail vazio. Essa configuração é realizada diretamente na função `useFormik`, permitindo que o formulário tenha um ponto de partida claro. A definição de valores iniciais também facilita a implementação de funcionalidades como redefinição e pré-preenchimento do formulário, caso necessário.
- **Manipulação de Eventos:** O Formik automatiza a gestão de eventos como `onChange`, `onBlur` e `onSubmit`, eliminando a necessidade de compromissos adicionais para essas funcionalidades. Ao vincular as entradas diretamente ao estado gerenciado pelo Formik, é necessário escrever código repetitivo para capturar e atualizar o estado é reduzido. Isso resulta em um código mais limpo e legível, onde a lógica de interação é separada da lógica de gerenciamento de estado.
- **Validação Integrada:** Durante a submissão do formulário, o Formik utiliza o esquema de validação definido pelo Yup para garantir que todas as entradas atendam aos critérios especificados. Se houver erros, o envio do formulário for bloqueado, e as mensagens de erro são apresentadas ao usuário de forma clara e imediata. Esse processo de validação integrado

garante que os dados obtidos sejam válidos antes de qualquer ação adicional, aumentando a confiabilidade das informações.

- **Função onSubmit:** Após a validação ser concluída com sucesso, a função onSubmit é acionada, permitindo a execução de ações específicas, como o envio de dados para um servidor ou a apresentação de um alerta com as informações inseridas. Essa função é um ponto crítico, pois é onde a lógica de negócio pode ser inovadora, permitindo que o desenvolvedor integre o formulário a outras partes da aplicação ou sistemas externos. Além disso, a capacidade de acessar os valores do formulário diretamente na função onSubmit simplifica a manipulação dos dados encontrados.
- **Aprimoramentos Adicionais:** O Formik também oferece a possibilidade de personalizar o comportamento do formulário com ganchos (hooks) adicionais e funções de gerenciamento de estado. Isso permite que os desenvolvedores implementem funcionalidades mais avançadas, como a adição de campos dinâmicos ou a integração com outras bibliotecas de estado, caso necessário.

Com essas características, o Formik se destaca como uma ferramenta poderosa e eficiente para gerenciar o estado de formulários em aplicações React, permitindo que os desenvolvedores se concentrem na criação de experiências de usuário envolvente e funcional, enquanto a complexidade do gerenciamento de estado é tratada de maneira eficaz.

## 5- MELHORIAS

Para melhorar o formulário, aqui estão algumas sugestões práticas:

### 1. Adição de Novos Campos Relevantes

- **Campos Dinâmicos:** Adicione campos que se ajustam com base nas respostas anteriores. Isso torna o formulário mais inteligente e menos sobrecarregado.
- **Campo de Anexo:** Permitir upload de documentos ou imagens, caso o formulário exija comprovações ou registros visuais.
- **Campo de Observações:** Um campo de texto livre ao final para o usuário fazer comentários adicionais. O usuário pode capturar dados que não foram previstos.

## 2. Melhoria na Experiência do Usuário

- **Validação em Tempo Real:** Ao usar bibliotecas como Formik e Yup, habilite validações que mostrem mensagens de erro enquanto o usuário digita. Isso reduz o atrito ao corrigir erros antes do envio final.
- **Design Responsivo:** Garanta que o formulário se adapte bem aos dispositivos móveis. Facilite a leitura e a interação em telas menores, com botões maiores e fontes inovadoras.
- **Indicador de Progresso:** Se o formulário tiver várias etapas, incluindo um indicador para mostrar em que etapa o usuário está e quanto falta para concluir.
- **Auto-salvamento e Continuação:** Permite que o formulário seja salvo automaticamente, permitindo que o usuário retome de onde parou.

## 3. Otimizações de Desempenho

- **Carregamento Condicional:** Carregue apenas os campos ou rótulos relevantes com base nas respostas dos usuários, envie o peso total do formulário.
- **Carregamento lento para componentes pesados:** Utilize blocos lentos para componentes que só serão exibidos caso o usuário avance até eles.
- **Redução de Bibliotecas Externas:** Caso existam muitas dependências, elimine as não essenciais, compatíveis com soluções nativas de JavaScript ou alternativas mais leves.

Essas melhorias podem contribuir para uma experiência mais intuitiva, rápida e envolvente para o usuário, além de reduzir o esforço de suporte técnico necessário.

## 6- CONCLUSÃO

O formulário React desenvolvido combina a simplicidade do Formik com a robustez da validação do Yup, resultando em uma experiência interativa e dinâmica para o usuário. A implementação de funcionalidades de validação rigorosas e feedback em tempo real é crucial para garantir que as informações enviadas sejam precisas e válidas, contribuindo significativamente para a confiabilidade e eficiência do aplicativo. Ao garantir que o e-mail inserido atenda aos critérios de formato e obrigatoriedade, o sistema minimiza erros e potencializa a integridade dos dados.

A interação da interface foi projetada especificamente para ser intuitiva e responsiva, incorporando indicadores visuais e mensagens de erro claras. Esses elementos não apenas facilitam a compreensão das informações a serem inseridas, mas também melhoram a experiência geral, tornando o formulário mais acessível e menos exposto a frustrações. O feedback imediato permite que os usuários façam correções instantâneas, resultando em um processo de preenchimento mais ágil e eficiente.

Além disso, o gerenciamento do estado do formulário por meio do Formik simplifica a lógica de manipulação de eventos, permitindo que os desenvolvedores se concentrem na construção de funcionalidades em vez de se perderem em detalhes técnicos. A integração entre a validação do Yup e o gerenciamento de estado do Formik exemplifica a sinergia entre diferentes bibliotecas para fornecer uma solução robusta.

Essa abordagem não apenas melhorou a experiência do usuário, mas também fundamenta as melhores práticas de desenvolvimento em aplicações web modernas, como a modularização de código, a gestão clara de estado e a importância do feedback visual. A combinação dessas práticas resulta em um aplicativo que não é apenas funcional, mas também agradável e eficaz, refletindo um compromisso com a qualidade e a usabilidade.

Em suma, o projeto declarado como a utilização adequada de ferramentas e bibliotecas pode elevar o nível de um formulário simples a uma solução poderosa, evidenciando a importância da validação, interação do usuário e gerenciamento eficiente de estados no desenvolvimento de aplicações web.

Essas lições e as práticas adotadas aprendidas podem ser aplicadas em projetos futuros, ampliando ainda mais o potencial de criação de interfaces de usuário impactantes e funcionais.