Vitor Amadeu Souza

Introdução ao

React

e

CSS

No ambiente Replit

Parte VIII

© 2024 by Cerne Tecnologia e Treinamento Ltda.

© 2024 by Vitor Amadeu Souza

Nenhuma parte desta publicação poderá ser reproduzida sem autorização prévia e escrita de **Cerne Tecnologia e Treinamento Ltda.** Este livro publica nomes comerciais e marcas registradas de produtos pertencentes a diversas companhias. O editor utiliza as marcas somente para fins editoriais e em benefício dos proprietários das marcas, sem nenhuma intenção de atingir seus direitos.

Agosto de 2024

Direitos reservados por:
Cerne Tecnologia e Treinamento Ltda
Produção: Cerne Tecnologia e Treinamento
E-mail da Empresa: cerne@cerne-tec.com.br
Home Page: www.cerne-tec.com.br.com.br
Atendimento ao Consumidor: sac@cerne-tec.com.br
Contato com o Autor: vitor@cerne-tec.com.br



FEITO NO BRASIL

“Venha também sobre mim a tua benignidade, ó Senhor, e a tua salvação, segundo a tua palavra.”

Sl 119:41

Cerne Tecnologia

A Cerne Tecnologia tem uma equipe preparada para desenvolvimento de projetos eletrônicos em diversas áreas: Médica, Entretenimento, Industrial, Robótica, Científica, Automobilística, Aeronáutica, etc. Trabalhamos com tecnologia microcontrolada usando o PIC, ARM, AVR, 8051, dsPIC, PIC24, PIC32 além do Arduino, Raspberry, Beaglebone etc. Desenvolvemos o projeto desde sua concepção até a entrega do produto final, passando pelas etapas de esquema elétrico, protótipo e desenvolvimento de circuito impresso.

Desenvolvemos aplicativos para smartphones/tablets Android, iOS, Blackberry, Windows Phone e no desenvolvimento de softwares a nível PC para plataforma Windows, usando ferramentas como o Visual Basic, C# e C++.

Atuamos na parte de montagem de placas, onde podemos fornecer ambos os serviços de desenvolvimento de projetos e produção ou apenas um destes.

Desenvolvemos esquemas elétricos e layout de PCI, tanto em tecnologia convencional como SMD.

Temos a flexibilidade de customizar um de nossos produtos, de modo a atender a uma necessidade específica do cliente, tornando o custo de desenvolvimento menor se comparado a construção de um projeto desde a sua fase inicial.

Desenvolvemos e fornecemos kits didáticos para diversos microcontroladores além de apostilas, livros e e-books.

Na hora de desenvolver um projeto ou equipar seu laboratório não hesite em nos contatar. Entre em contato conosco através do endereço cerne-tec.com.br para obter mais informações.

Cerne
Conhecimento para o Desenvolvimento

cerne-tec.com.br

Sumário

Capítulo I – Metodologia de desenvolvimento **6**

1. Introdução 6

Capítulo II – Programação em React e CSS **7**

1. Centralizar um botão 7

2. Layout de Navbar Horizontal 8

3. Layout de Navbar Horizontal com logo 12

4. Layout de Navbar Horizontal com logo 16

Capítulo I
Metodologia de desenvolvimento

1. Introdução

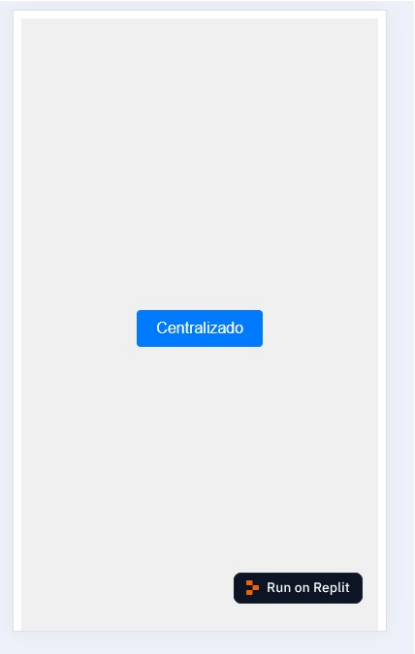
Neste livro, você encontrará uma parte da série sobre o desenvolvimento de projetos utilizando React e CSS. Ao longo desta obra, diversos experimentos serão apresentados segundo o sumário da obra, permitindo o aperfeiçoamento neste framework para a construção de interfaces responsivas juntamente com o CSS.

Capítulo II

Programação em React e CSS

1. Centralizar um botão

O próximo exemplo centraliza um item dentro de um contêiner em uma tela responsiva, fazendo uso do React e CSS, como apresentado na figura abaixo.



Acompanhe como ficam os arquivos App.css e App.jsx para este exercício.

App.css

```
.container {
  display: flex;
  justify-content: center;
  align-items: center;
  height: 100vh;
  background-color: #f0f0f0;
}

.button {
  padding: 10px 20px;
  font-size: 16px;
  color: white;
  background-color: #007bff;
  border: none;
  border-radius: 4px;
}
```

App.jsx

```
import './App.css'

export default function App() {
  return (
    <div className="container">
      <button className="button">Centralizado</button>
    </div>
  )
}
```

Este código cria uma interface simples em React, onde um botão é centralizado vertical e horizontalmente na tela. O arquivo App.css define os estilos de layout e aparência do botão e do contêiner que o envolve. No CSS, a classe .container usa display: flex para criar um layout flexível, permitindo que o conteúdo interno seja facilmente centralizado. As propriedades justify-content: center e align-items: center garantem que o botão seja posicionado exatamente no centro da tela, tanto horizontal quanto verticalmente. A altura do contêiner é definida como 100vh, o que significa que ele ocupa 100% da altura da janela de visualização. O fundo do contêiner é cinza claro, definido por background-color: #f0f0f0.

A classe .button estiliza o botão com preenchimento (padding: 10px 20px), tamanho de fonte (font-size: 16px), cor de texto branca, e fundo azul (background-color: #007bff). Além disso, o botão não tem bordas (border: none) e possui cantos arredondados (border-radius: 4px).

No arquivo App.jsx, a função App retorna um JSX que cria um div com a classe container, dentro do qual há um botão com a classe button. O texto do botão é "Centralizado". O layout flexível aplicado ao contêiner centraliza automaticamente o botão na tela.

2. Layout de Navbar Horizontal

O próximo adiciona um navbar a tela com três itens de menu, como apresentado na figura abaixo.

