Vitor Amadeu Souza

Introdução ao

React

e

CSS

No ambiente Replit

Parte X

© 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. Itens com Largura Fixa e Flexível 7
2. Itens com Flex Wrap 10
3. Gerando um artefato 14
4. Interface com Header, Sidebar, Content e Footer 15

Capítulo I
Metodologia de desenvolvimento

1. Introdução

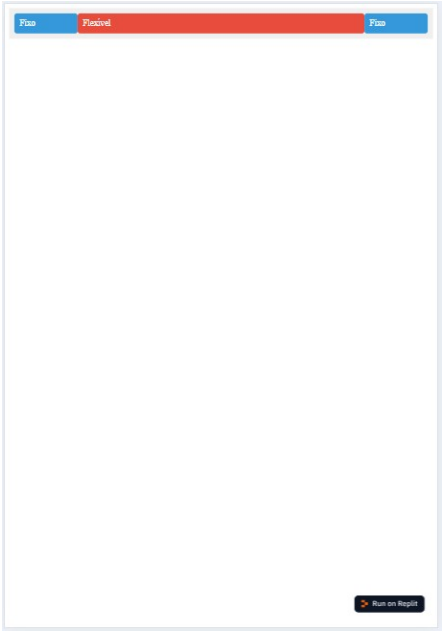
Neste livro, você encontrará uma parte da série sobre o desenvolvimento de projetos utilizando React e CSS. Ao longo desta obra, diversos experimentos serão apresentados segundo o sumário da obra, permitindo o aperfeiçoamento neste framework para a construção de interfaces responsivas juntamente com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

Capítulo II

Programação em React e CSS

1. Itens com Largura Fixa e Flexível

O próximo exemplo usa uma importante propriedade do CSS para fornecer 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: space-between;
  background-color: #f0f0f0;
  padding: 10px;
}

.fixed-item {
  background-color: #3498db;
  color: white;
  padding: 10px;
  border-radius: 5px;
  width: 100px; /* Largura fixa */
}

.flexible-item {
  background-color: #e74c3c;
  color: white;
  padding: 10px;
  border-radius: 5px;
  flex: 1; /* Largura flexível */
}
```

App.jsx

```
import React from 'react';
import './App.css';
```

```
export default function App() {
  return (
    <div className="container">
      <div className="fixed-item">Fixo</div>
      <div className="flexible-item">Flexível</div>
      <div className="fixed-item">Fixo</div>
    </div>
  );
}
```

O código apresentado cria uma interface de usuário utilizando React e CSS, onde diferentes elementos são dispostos dentro de um contêiner flexível. No arquivo App.css, a classe .container define um contêiner com a propriedade display: flex, que organiza seus elementos em uma linha. A propriedade justify-content: space-between é usada para espaçar os itens ao longo do eixo horizontal, colocando o primeiro e o último item nos extremos do contêiner, e o item flexível ocupando o espaço restante entre eles. O fundo do contêiner é cinza claro (#f0f0f0), e ele tem um padding de 10 pixels ao redor de seus elementos.

A classe .fixed-item estiliza os itens com largura fixa. Esses itens têm um fundo azul (#3498db), texto branco, padding de 10 pixels, bordas arredondadas com um raio de 5 pixels, e uma largura fixa de 100 pixels. Por outro lado, a classe .flexible-item estiliza o item com

largura flexível. Este item tem um fundo vermelho (#e74c3c), texto branco, padding de 10 pixels, bordas arredondadas de 5 pixels, e a propriedade flex: 1, que faz com que ele expanda para ocupar todo o espaço disponível dentro do contêiner.

No arquivo App.jsx, o componente principal é definido e exportado. Este componente retorna um div com a classe container, que contém três divs: dois com a classe fixed-item, que são itens de largura fixa com o texto "Fixo", e um com a classe flexible-item, que é o item de largura flexível com o texto "Flexível". O resultado final é uma interface onde dois itens de largura fixa (um à esquerda e outro à direita) são dispostos nos extremos do contêiner, enquanto o item flexível ocupa o espaço restante no meio. Isso cria um layout que equilibra elementos com tamanhos diferentes de forma clara e organizada.

2. Itens com Flex Wrap

O próximo design faz uso do flex-wrap para desenvolvimento, de maneira responsiva.