

**Vitor Amadeu Souza**

Introdução ao

# **React Native**

No ambiente Expo Snack

Parte V

© 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.

**Outubro 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](http://cerne-tec.com.br) para obter mais informações.



# Sumário

<b>Capítulo I – Metodologia de desenvolvimento.....</b>	<b>6</b>
1. Introdução.....	6
<b>Capítulo II – Programação em React Native.....</b>	<b>7</b>
1. Componente Slider.....	7
2. Componente Modal.....	11
3. Componente Picker.....	16

# Capítulo I

## Metodologia de desenvolvimento

### 1. Introdução

Neste livro, você encontrará uma parte da série dedicada ao desenvolvimento de projetos com React Native. Ao longo desta obra, diversos experimentos serão apresentados conforme o sumário, permitindo um aperfeiçoamento contínuo no uso deste framework para a construção de interfaces responsivas com foco em dispositivos móveis.

# Capítulo II

## Programação em React Native

### 1. Componente Slider

O componente Slider no React Native permite que o usuário selecione um valor dentro de uma faixa contínua movendo um controle deslizante entre um valor mínimo e máximo. Ele é amplamente utilizado em situações onde o usuário precisa ajustar configurações como volume, brilho ou intensidade de cor, entre outras. O Slider possui várias propriedades importantes para controlar seu comportamento, como `minimumValue`, que define o valor mínimo que o slider pode assumir, e `maximumValue`, que define o valor máximo. A propriedade `value` é usada para definir o valor inicial do slider, enquanto a função `onValueChange` é chamada sempre que o valor do slider muda, permitindo capturar o valor atual em tempo real. Além disso, a propriedade `step` pode ser usada para definir incrementos discretos no valor do slider, por exemplo, com um `step` de 1, o controle só permitirá valores inteiros.

No exemplo básico de uso do componente, utiliza-se o `useState` para armazenar o valor do slider, inicialmente definido como 50. O Slider é configurado para permitir que o usuário ajuste o valor entre 0 e 100, e cada vez que o valor é alterado, a função `onValueChange` atualiza o estado com o novo valor. Esse valor atual é exibido dinamicamente acima do slider usando o componente `Text`. As cores da trilha (ou `track`), que é a linha percorrida pelo controle do slider, são personalizadas com as propriedades `minimumTrackTintColor` e `maximumTrackTintColor`, que definem as cores para as partes à esquerda e à direita do ponto de controle, respectivamente. O estilo do slider é configurado com uma largura adequada para se ajustar bem à interface, utilizando a propriedade `style`.

O Slider é uma solução eficaz para várias aplicações, como controle de volume, ajustes de configurações como brilho, zoom ou seleção de intervalos de tempo. Ele pode ser customizado para se alinhar ao design da interface e às necessidades de usabilidade da aplicação, tornando-se uma ferramenta versátil em várias situações interativas.

```
import React, { useState } from 'react';
import { View, Text } from 'react-native';
import Slider from '@react-native-community/slider';

const SliderExample = () => {
```

```

const [value, setValue] = useState(50);

return (
  <View style={{ flex: 1, justifyContent: 'center', alignItems:
'center' }}>
    <Slider
      style={{ width: 200, height: 40 }}
      minimumValue={0}
      maximumValue={100}
      value={value}
      onValueChange={setValue}
      minimumTrackTintColor="#FFFFFF"
      maximumTrackTintColor="#000000"
    />
    <Text>Valor: {Math.round(value)}</Text>
  </View>
);
};

export default SliderExample;

```

O código apresentado cria um exemplo de utilização do componente Slider no React Native, permitindo ao usuário ajustar um valor entre 0 e 100. O estado inicial do slider é definido com o hook useState, começando em 50. O valor atual do slider é atualizado em tempo real através da função onValueChange, que usa a função setValue para modificar o estado conforme o usuário move o controle do slider.

O componente Slider, importado de [@react-native-community/slider](https://github.com/react-native-community/slider), permite a customização de sua aparência e comportamento. Neste exemplo, o slider tem uma largura de 200 e altura de 40. As cores da trilha são configuradas com

`minimumTrackTintColor` e `maximumTrackTintColor`, que controlam, respectivamente, as cores da parte da trilha à esquerda e à direita do ponto de controle. O valor atual do slider, arredondado para o inteiro mais próximo com `Math.round`, é exibido logo abaixo do controle utilizando o componente `Text`.

O layout da aplicação é organizado de forma que o slider e o texto fiquem centralizados na tela, utilizando as propriedades `justifyContent` e `alignItems` dentro do `View` que envolve os componentes. Essa estrutura é simples e eficaz para criar uma interface que permita ao usuário ajustar facilmente um valor e visualizar a mudança em tempo real.