

React Native

Projeto de Interfaces de Dispositivos Móveis

Projeto Albums (Parte 2)

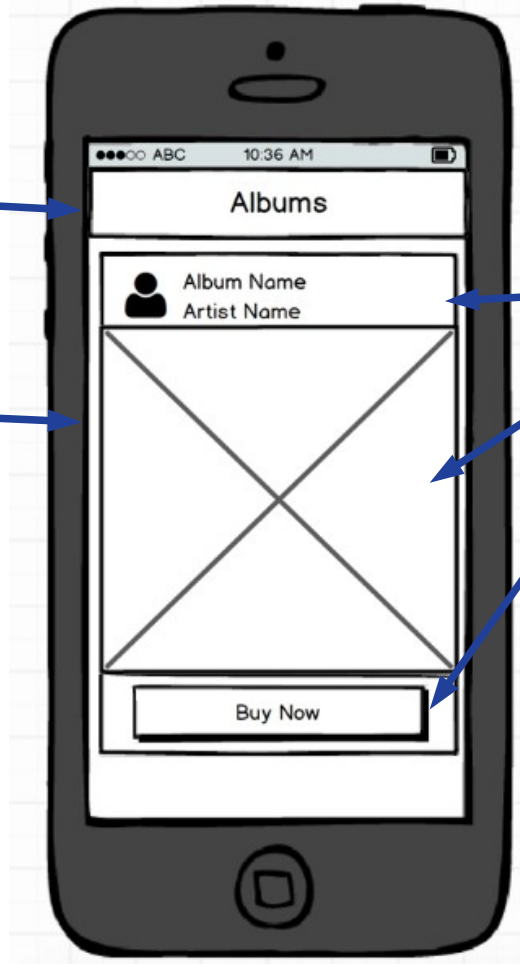
Aula 04

Introdução

- **Projeto Albums - Parte 2**
 - FlexBox
 - Imagens e botões
 - Scrolling
 - Entrada do usuário

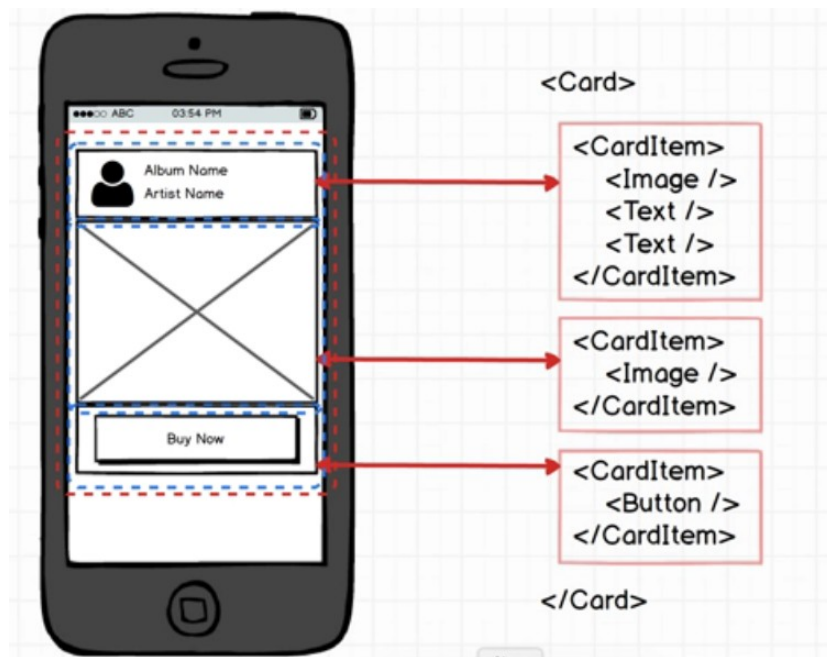
Header Component

Card Component



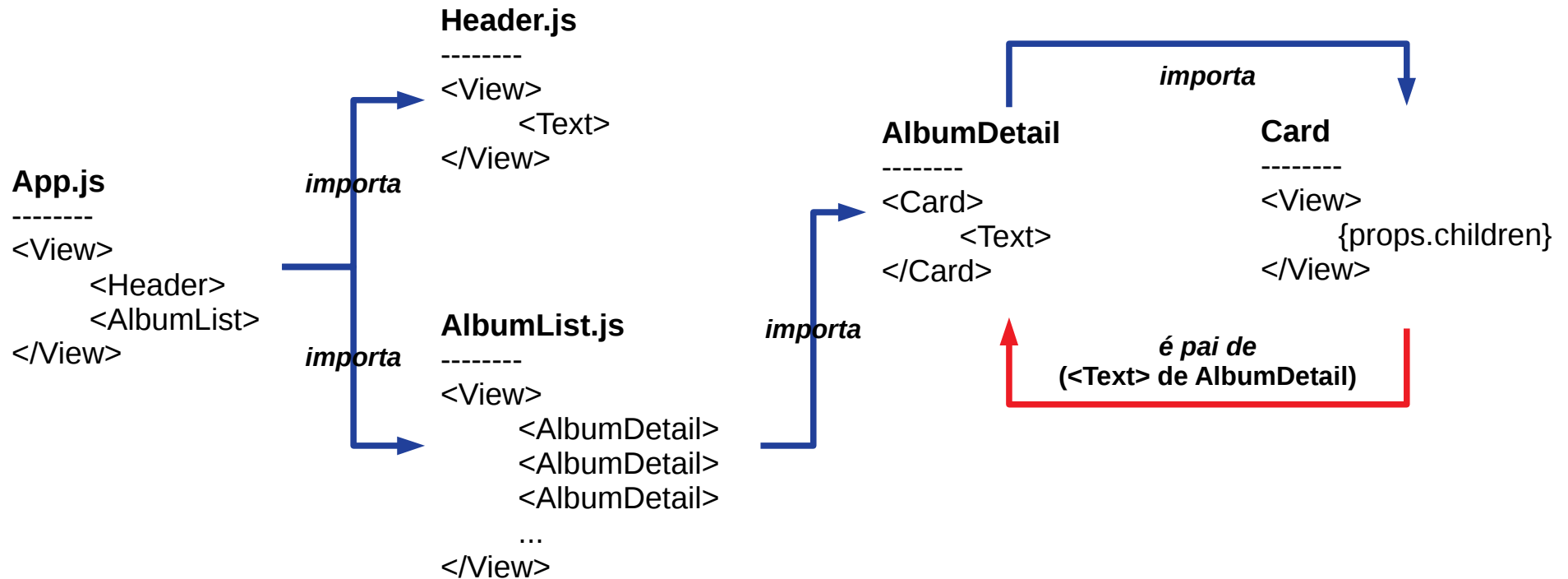
Card Section

Detalhes do Album

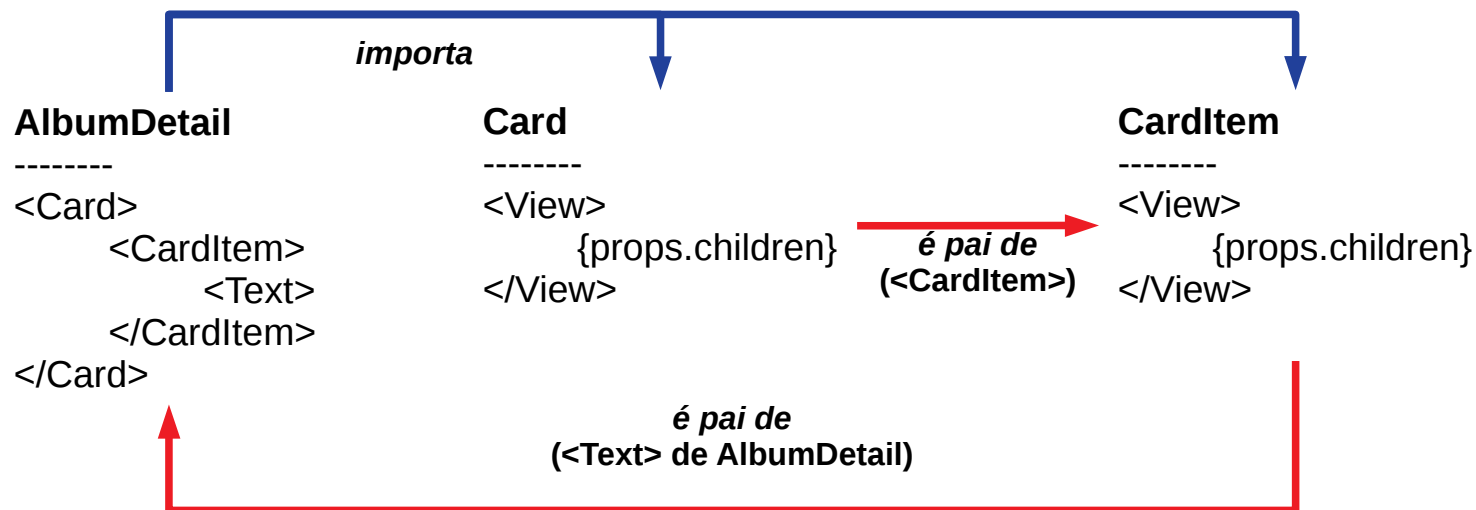


<https://www.udemy.com/the-complete-react-native-and-redux-course/>

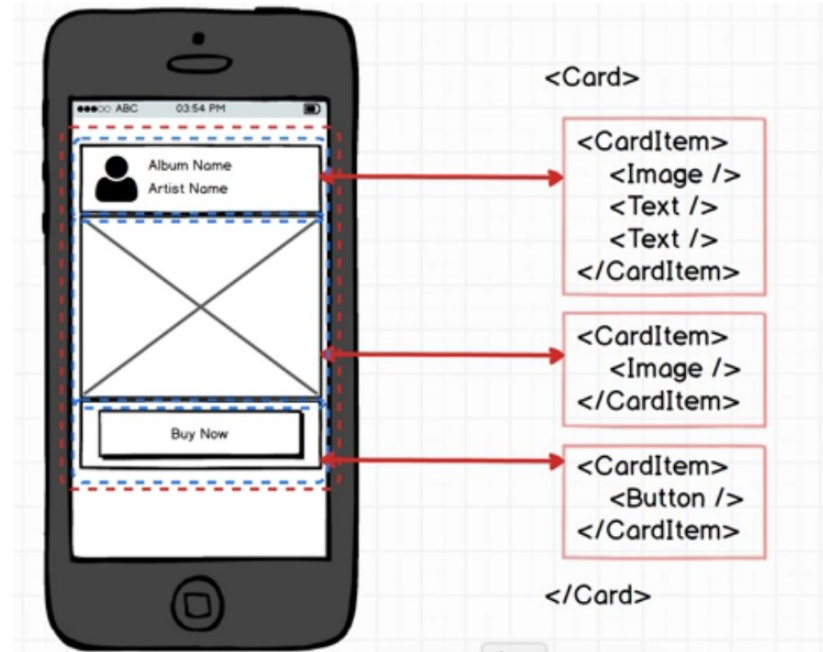
Visão geral (até agora)



Visão Geral (até agora)



Album Detail



Header do Card

Body do Card

Footnote do Card

<https://www.udemy.com/the-complete-react-native-and-redux-course/>

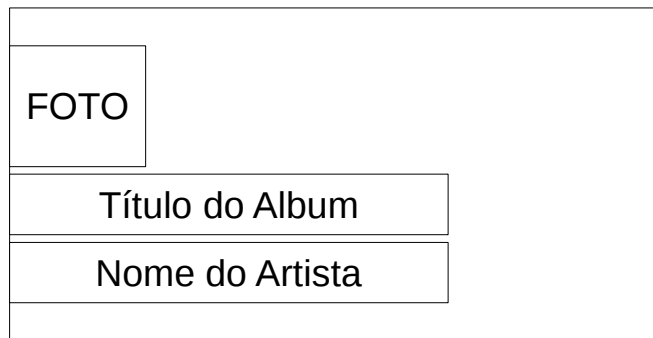
Header do Card



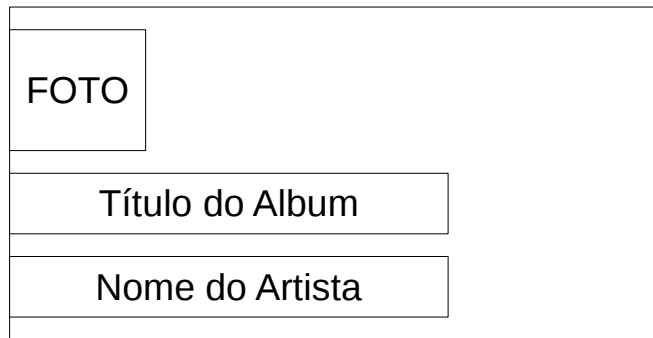
justifyContent: 'flex-end'



justifyContent: 'space-between'

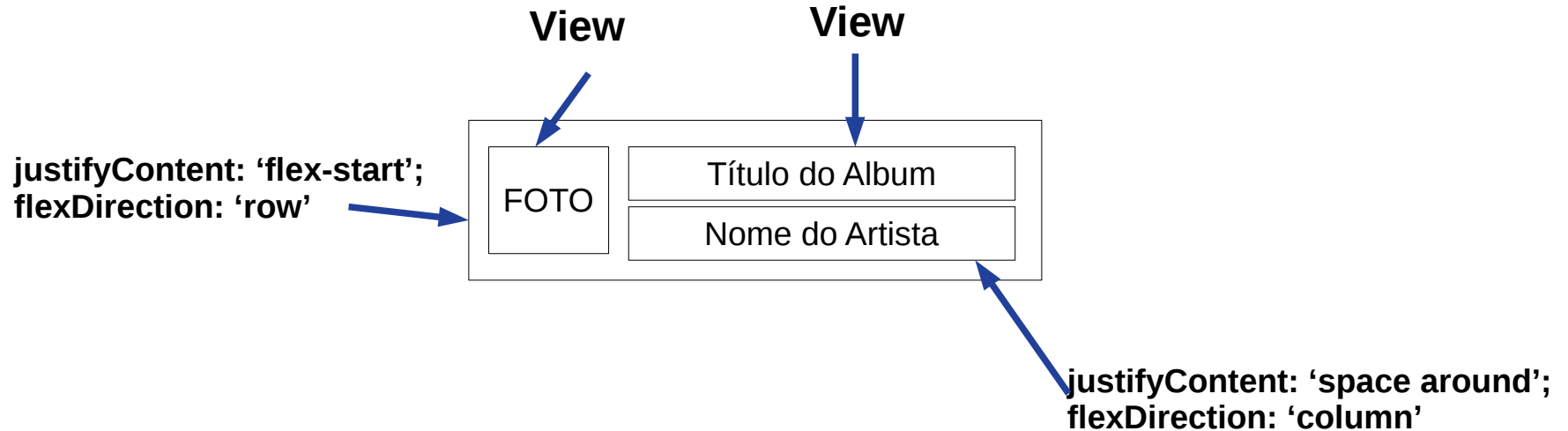


justifyContent: 'center'



justifyContent: 'space-around'

Header do Card



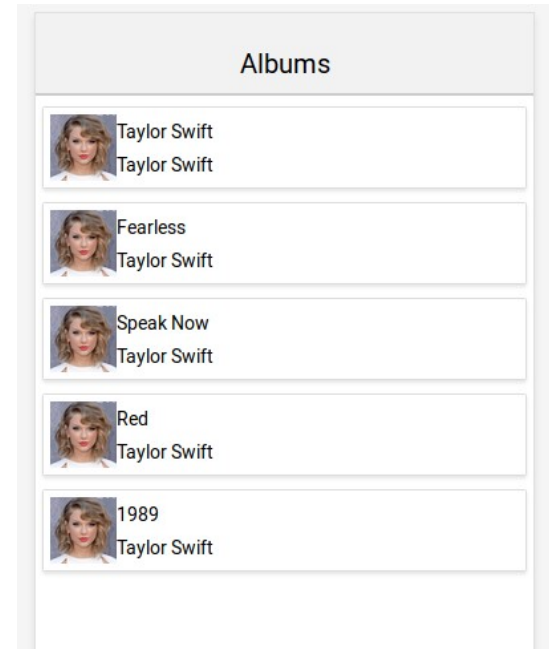
```
class AlbumDetail extends Component {  
  
  constructor(props) {  
    super(props);  
  }  
  
  render() {  
    return (  
      <Card>  
        <CardItem style={styles.cardItemStyle}>  
          <View>  
            <Text>FOTO</Text>  
          </View>  
          <View style={styles.textStyle}>  
            <Text>{this.props.album.title}</Text>  
            <Text>{this.props.album.artist}</Text>  
          </View>  
        </CardItem>  
      </Card>  
    );  
  }  
}
```

```
const styles = StyleSheet.create({  
  
  cardItemStyle: {  
    justifyContent: 'flex-start',  
    flexDirection: 'row'  
  },  
  textStyle: {  
    justifyContent: 'space-around',  
    flexDirection: 'column'  
  }  
});
```

Header do Card

- Usando a imagem

```
<Card>
  <CardItem style={styles.cardItemStyle}>
    <View>
      <Image
        style={styles.thumbStyle}
        source={{uri:this.props.album.thumbnail_image}} />
    </View>
    <View style={styles.textStyle}>
      <Text>{this.props.album.title}</Text>
      <Text>{this.props.album.artist}</Text>
    </View>
  </CardItem>
</Card>
...
thumbStyle: {
  height:50,
  width:50
}
....
```



Header do Card

- Melhorando alguns estilos...

```
<Card>
  <CardItem style={styles.cardItemStyle}>
    <View style={styles.thumbContentStyle}>
      <Image
        style={styles.thumbStyle}
        source={{uri: this.props.album.thumbnail_image}} />
    </View>
    <View style={styles.textStyle}>
      <Text style={styles.titleStyle}>{this.props.album.title}</Text>
      <Text>{this.props.album.artist}</Text>
    </View>
  </CardItem>
</Card>
```

```
....
thumbContentStyle:{
  justifyContent: 'center',
  alignItems: 'center',
  marginLeft: 10,
  marginRight:10
},
titleStyle:{
  fontSize: 18
}
...
```

Body do Card

- adicionando imagem principal, logo abaixo do `</CardItem>`

```
<Card>
  <CardItem style={styles.cardItemStyle}>
    <View style={styles.thumbContentStyle}>
      <Image
        style={styles.thumbStyle}
        source={{uri: this.props.album.thumbnail_image}} />
    </View>
    <View style={styles.textStyle}>
      <Text style={styles.titleStyle}>{this.props.album.title}</Text>
      <Text>{this.props.album.artist}</Text>
    </View>
  </CardItem>

  <CardItem>
    <Image style={styles.albumImage}
      source={{uri: this.props.album.image}} />
  </CardItem>
</Card>
```

```
...
albumImage: {
  flex: 1,
  height: 300,
  width: null
}
...
```

Body do Card

- Scrolling em AlbumList.js

```
render(){  
  return (  
    <ScrollView>  
      {this.renderAlbums()}  
    </ScrollView>  
  );  
}
```

Importante! Em App.js (componente pai de AlbumList), você deve:

```
<View style={{flex:1}}>  
  <Header title='Albums' />  
  <AlbumList />  
</View>
```

Botton do Card

- Crie o arquivo Button.js

```
import React, { Component } from 'react';  
import { AppRegistry, View, Text, TouchableOpacity } from 'react-native';
```

```
export default class Button extends Component {  
  render() {  
    return (  
      <TouchableOpacity>  
        <Text>Clique Aqui!</Text>  
      </TouchableOpacity>  
    );  
  }  
}
```

```
AppRegistry.registerComponent('albums', () => Button);
```

Botton do Card

- Criar o CardItem em AlbumDetail.js

...

```
<CardItem>  
  <Image style={styles.albumImage}  
    source={{uri:this.props.album.image}}/>  
</CardItem>
```

```
<CardItem>  
  <Button />  
</CardItem>
```

...

Botton do Card

- **Inserindo estilos**

```
<TouchableOpacity style={styles.buttonStyle}>
  <Text style={styles.textStyle}>
    Clique Aqui!?
  </Text>
</TouchableOpacity>
```

```
const styles = StyleSheet.create({
  buttonStyle : {
    flex: 1,
    alignSelf: 'stretch',
    backgroundColor: '#fff',
    borderRadius: 5,
    borderWidth: 1,
    borderColor: '#007aff',
    marginLeft: 5,
    marginRight: 5
  },
  textStyle: {
    alignSelf: 'center',
    color: '#007aff',
    fontSize: 16,
    fontWeight: '600',
    paddingTop: 10,
    paddingBottom: 10
  }
});
```

Respondendo o botão

- Exemplo simples:

```
export default class Button extends Component {  
  render() {  
    return (  
      <TouchableOpacity  
        onPress={()=>alert('Pressionado!')}  
        style={styles.buttonStyle}>  
        <Text style={styles.textStyle}>Clique Aqui!</Text>  
      </TouchableOpacity>  
    );  
  }  
}
```

Generalizando o botão

```
export default class Button extends Component {  
  constructor(props) {  
    super(props);  
  }  
  
  render() {  
    return (  
      <TouchableOpacity  
        onPress = {this.props.action}  
        style={styles.buttonStyle}>  
        <Text style={styles.textStyle}>{this.props.name}</Text>  
      </TouchableOpacity>  
    );  
  }  
}
```

Generalizando o botão

- Em AlbumDetail.js

```
<CardItem>  
  <Button  
    name={'Clique Aqui!'}  
    action={()=>alert("Clicou em "+this.props.album.title+"!")}/>  
</CardItem>
```

Link Externo

- Vamos usar a API Linking, do React Native
 - <https://facebook.github.io/react-native/docs/linking>
- O nosso objetivo é fazer com que nossa aplicação móvel se comunique com aplicações externas, como por exemplo, o browser.
- No caso da nossa aplicação, ao clicar no botão de um determinado Álbum, queremos ir para a página referente a ele.

Link Externo

```
import React, { Component } from 'react';
import { View, Text, Image, StyleSheet, Linking } from 'react-native';

import MyButton from './MyButton';
import Card from './Card';
import CardItem from './CardItem'

export default class AlbumDetail extends Component {
  //...
  <CardItem>
    <MyButton
      name="OK"
      action={()=>Linking.openURL(this.props.album.url)}
    />
  </CardItem>
```

Exercício

- Use o **this.props.children** em MyButton para poder renderizar o nome de um botão, passado como parâmetro pelas tags:
 - `<MyButton>`
 Buy Now!!!
 `</MyButton>`