Fpt University

FPT POLYTECHNIC

LẬP TRÌNH ĐA NỀN TẢNG VỚI REACT

BÀI 2: SỬ DỤNG CÁC THÀNH PHẦN TRONG REACT NATIVE

PHẦN 1: SỬ DỤNG STATE, PROPS

www.poly.edu.vn

Mục tiêu

- Giới thiệu chung về State trong React Native
- Kết hợp State và Props trong React Native
- Định dạng cho ứng dụng React bằng Styling
- Đáp ứng các kích cỡ màn hình khác nhau trong React Native

CÁC KHÁI NIỆM: STATE

- Dữ liệu bên trong các thành phần của REACT được quản lý bởi **state** và **props**.
- **State** là thành phần có thể thay đổi, còn **props** thì không thể thay đổi
- **State** có thể update trong tương lai, còn **props** thì không thể update

CÁC KHÁI NIỆM: STATE

- Sử dụng **State**:
- Chúng ta sử dụng trong **index.ios.js** hoặc **index.android.js**
- **State** được định nghĩa trong Home class bằng cách sử dụng cú pháp **state = {}**
- Cú pháp:

```
state = {
    myState: 'Day la trang thai state'
  }
```

DEMO: STATE

File **Index.android.js**

home.js | index.android.js

```
1  import React, { Component } from 'react';
2  import { AppRegistry, View } from 'react-native';
3  import Home from './src/components/home/Home.js'
4
5  class reactTutorialApp extends Component {
6     render() {
7        return (
8           <View>
9              <Home />
10          </View>
11       );
12    }
13 }
14 export default reactTutorialApp
15
16 AppRegistry.registerComponent('reactTutorialApp', () ⇒ reactTutorialApp);
```

DEMO: STATE

- File **src/components/home/Home.js**

home.js | index.android.js

```
1  import React, { Component } from 'react';
2  import { Text, View } from 'react-native';
3
4  class Home extends Component {
5      state = {
6          myState: 'Day la trang thai state'
7      }
8      render() {
9          return (
10             <View>
11                 <Text>
12                     {this.state.myState}
13                 </Text>
14             </View>
15         );
16     }
17 }
18 export default Home;
```

CÁC KHÁI NIỆM: STATE

- Update **State**:
- Chúng ta có thể update bằng cách tạo ra chức năng **deleteState** và gọi nó bằng cách sử dụng cú pháp:
- Cú pháp:

onPress = {this.deleteText}

DEMO: UPDATE STATE

- File **src/components/home/Home.js**

home.js | index.android.js

```
1  import React, { Component } from 'react'
2  import { Text, View } from 'react-native'
3
4  class Home extends Component {
5      state = {
6          myState: 'Cap nhat myState.'
7      }
8      updateState = () => this.setState({ myState: 'The state is updated' })
9      render() {
10         return (
11             <View>
12                 <Text onPress = {this.updateState}>
13                     {this.state.myState}
14                 </Text>
15             </View>
16         );
17     }
18 }
19 export default Home;
```

CÁC KHÁI NIỆM: STATE

- Ngoài ra chúng ta có thể dụng state theo cú pháp sau:

```
class Home extends Component {
        constructor(){
                super() this.updateState =
                this.updateState.bind(this)
                 }
                updateState(){ // }
                render(){ // }
}
```

CÁC KHÁI NIỆM: PROPS

- Các thành phần trình bày sẽ nhận được tất cả dữ liệu bằng cách chuyển các **props**. Chỉ các thành phần container phải có **state**.
- Cập nhật thành phần Container :
  - Thành phần này sẽ xử lý **State,** vượt qua các **props** để đến các thành phần trình bày trên ứng dụng.

CÁC KHÁI NIỆM: PROPS

- Cập nhật thành phần Container :
  - Thành phần container chỉ được sử dụng để xử lý **State**. Tất cả các chức năng liên quan đến **view** sẽ được xử lý trong các thành phần trình bày presentation.

Cú pháp sử dụng Props

- Muốn pass qua props ta dùngcú pháp sau:
- **myText = {this.state.myText}**
- Và
- **deleteText = {this.deleteText}**

THÀNH PHẦN TRÌNH BÀY (PRESENTATION COMPONENT)

- Bây giờ chúng ta sẽ tìm hiểu được thành phần trình bày là gì và nó hoạt động như thế nào?
- Các thành phần trình bày chỉ nên sử dụng để trình bày view cho người sử dụng.
- Các thành phần này không có trạng thái state.

THÀNH PHẦN TRÌNH BÀY (PRESENTATION COMPONENT)

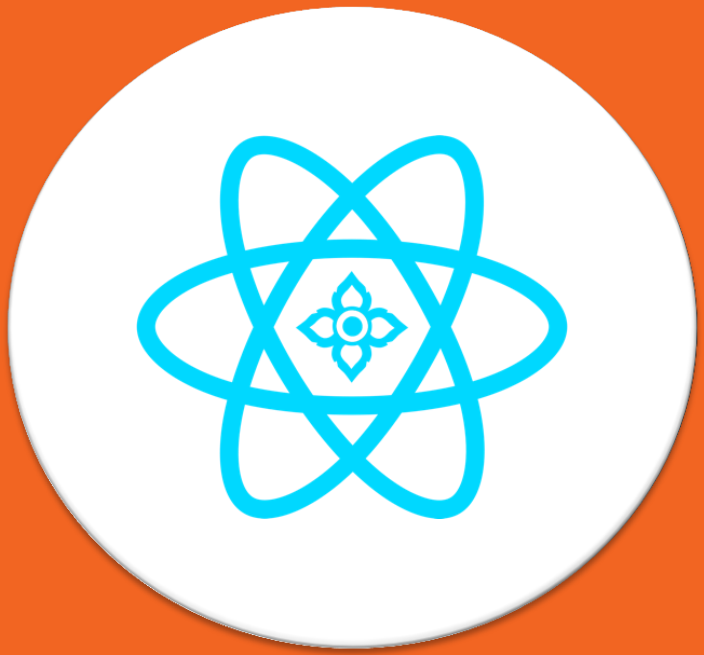
- Thành phần này nhận được tất cả các dữ liệu và chức năng như là props.
- Thực tiễn tốt nhất là sử dụng càng nhiều thành phần trình bày trong ứng dụng càng tốt.

Cú pháp sử dụng Presentational Component

- Thành phần này sẽ nhận props, trả về view, trình bày dữ liệu text
- Cú pháp trình bày dữ liệu text:
  - **{props.myText}**
- Sau đó gọi chức năng **{props.deleteText}** khi user click vào text

Fpt University

FPT POLYTECHNIC



LẬP TRÌNH ĐA NỀN TẢNG VỚI REACT

BÀI 2: SỬ DỤNG CÁC THÀNH PHẦN TRONG REACT NATIVE

PHẦN 2: SỬ DỤNG STYLING, FLEXBOX

www.poly.edu.vn

CÁC KHÁI NIỆM: STYLING

- Styling: Dùng để định dạng cho ứng dụng React Native
- Có 2 cách định dạng:
  - Inline
  - Tạo StyleSheet

CÁC KHÁI NIỆM: STYLING

- Đối với **Container Component**
- File **src/components/home/Home.js**

home.js

```
1  import React, { Component } from 'react'
2  import { View } from 'react-native'
3  import PresentationalComponent from './PresentationalComponent'
4
5  class Home extends Component {
6      state = {
7          myState: 'This is my state'
8      }
9      render() {
10         return (
11             <View>
12                 <PresentationalComponent myState = {this.state.myState}/>
13             </View>
14         )
15     }
16 }
17 export default Home
```

CÁC KHÁI NIỆM: STYLING

- Đối với **Presentational Component**
- File **src/components/home/PresentationalComponent.js**
- Ta sẽ sử dụng cách Import Stylesheet
- Ở phần bottom của file, chúng ta tạo một StyleSheet

CÁC KHÁI NIỆM: STYLING

- Đối với **Presentational Component**
- File **src/components/home/PresentationalComponent.js**
- Sau đó assign nó bằng hằng số Style
- Chú ý rằng tất cả style ở trong **camelCase**
- Chúng ta không sử dụng pixel hoặc % cho Styling

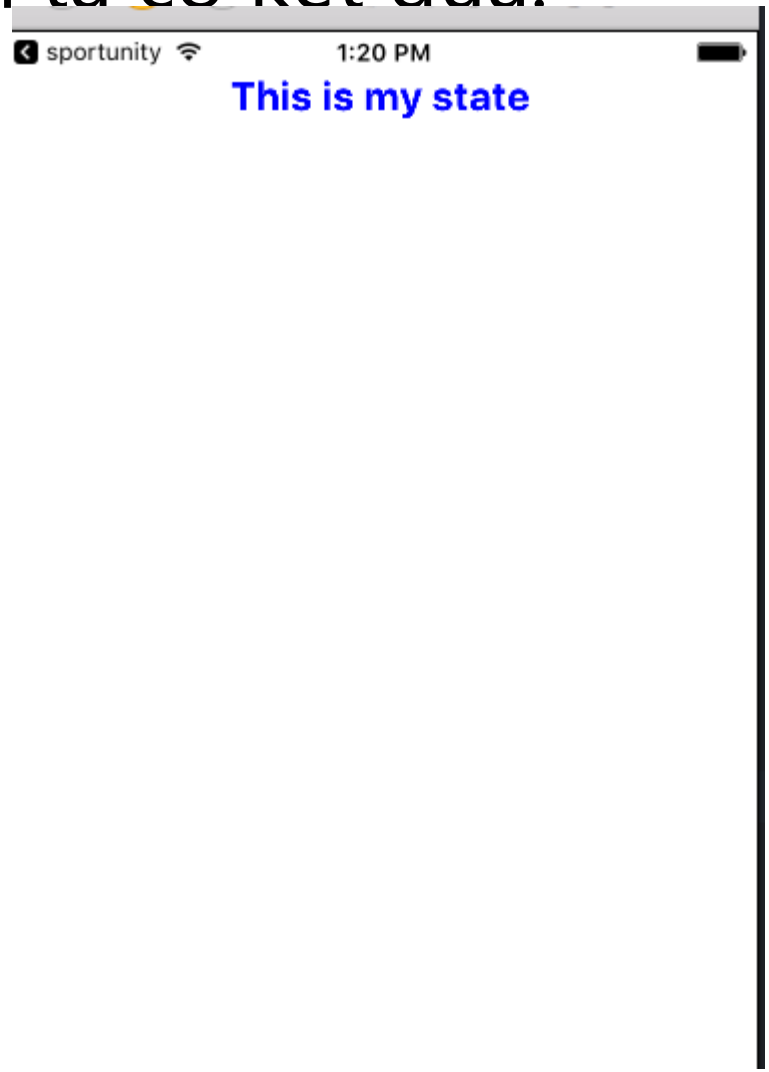
CÁC KHÁI NIỆM: STYLING

- Đối với **Presentational Component**
- File **src/components/home/PresentationalComponent.js**
- Để áp dụng Style vào text, ta dùng cú pháp sau:
  - **style = {styles.myText}**

- Đối với **Presentational Component**

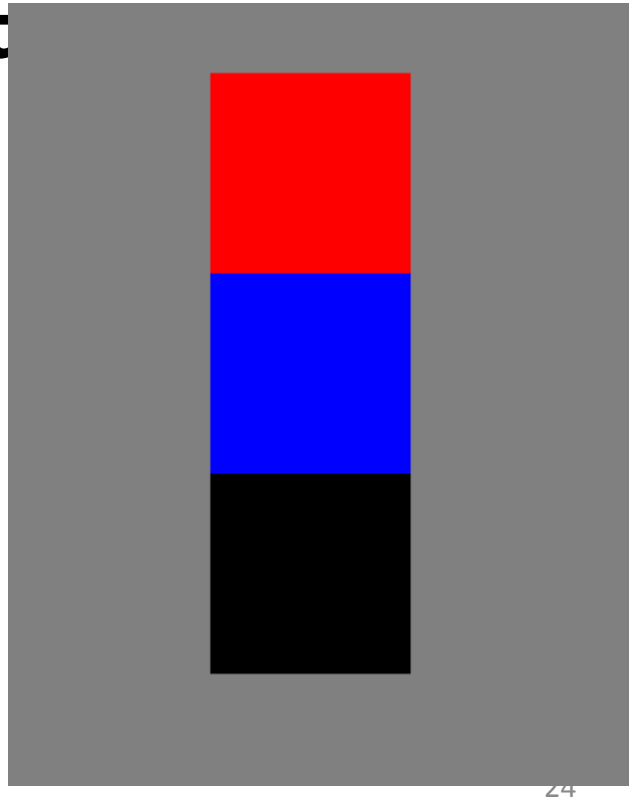
```
1  import React, { Component } from 'react'
2  import { Text, View, StyleSheet } from 'react-native'
3
4  const PresentationalComponent = (props) ⇒ {
5     return (
6        <View>
7           <Text style = {styles.myState}>
8              {props.myState}
9           </Text>
10       </View>
11    )
12 }
13 export default PresentationalComponent
14
15 const styles = StyleSheet.create ({
16    myState: {
17       marginTop: 20,
18       textAlign: 'center',
19       color: 'blue',
20       fontWeight: 'bold',
21       fontSize: 20
22    }
23 })
```

- Chạy chương trình ta có kết quả:



CÁC KHÁI NIỆM: FLEXBOX

- Flexbox: Dùng để Đáp ứng các kích cỡ màn hình khác nhau trong React Native
- Chúng ta dùng để thay đổi thành phần **PresentationalComponent**



CÁC KHÁI NIỆM: FLEXBOX

- Layout: Để đạt được bố cục mong muốn, flexbox cung cấp ba thuộc tính chính – **flexDirection, justifyContent** và **alignItems.**
- Thuộc tính **flexDirection**:
  - Giá trị: 'column', 'row'
  - Mô tả: Được sử dụng để chỉ định các phần tử sẽ được căn chỉnh theo chiều dọc hoặc chiều ngang.

CÁC KHÁI NIỆM: FLEXBOX

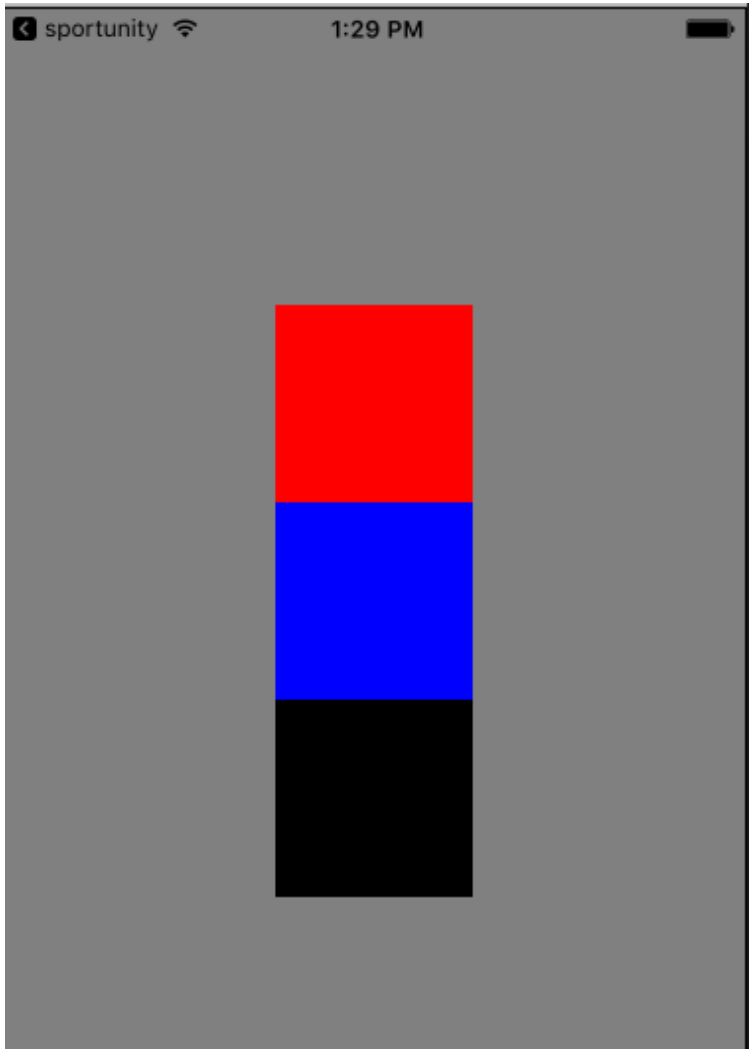
- Layout: Để đạt được bố cục mong muốn, flexbox cung cấp ba thuộc tính chính – **flexDirection, justifyContent** và **alignItems.**
- Thuộc tính **justifyContent**:
  - Giá trị: 'center', 'flex-start', 'flex-end', 'space-around', 'space-between'
  - Mô tả: Được sử dụng để xác định làm thế nào để các thành tố được phân phối bên trong container.

CÁC KHÁI NIỆM: FLEXBOX

- Layout: Để đạt được bố cục mong muốn, flexbox cung cấp ba thuộc tính chính – **flexDirection, justifyContent** và **alignItems.**
- Thuộc tính **alignItems**:
  - Giá trị: 'center', 'flex-start', 'flex-end', 'stretched'
  - Mô tả: Được sử dụng để xác định các yếu tố nên được phân phối như thế nào bên trong các container dọc theo trục thứ cấp (đối diện của flexDirection)

DEMO: FLEXBOX

- Flexbox có hình ảnh như sau



TỔNG KẾT

- Giới thiệu chung về State trong React Native
- Kết hợp State và Props trong React Native
- Định dạng cho ứng dụng React bằng Styling
- Đáp ứng các kích cỡ màn hình khác nhau trong React Native


TIN HỌC VĂN PHÒNG QUỐC TẾ THEO CHUẨN ICDL
Syllabus 5 - Office 2007
LUẬT KINH TẾ
LUẬT KINH TẾ
KỸ THUẬT SOẠN THẢO VÀ TRÌNH BÀY VĂN BẢN
TIN HỌC VĂN PHÒNG QUỐC TẾ THEO CHUẨN ICDL
Syllabus 5 - Office 2007
Cảm ơn