**BÁO CÁO**

**NGHIÊN CỨU CÁC THÀNH PHẦN CƠ BẢN TRONG REACTJS**

# MỤC LỤC

Mở đầu

Chương 1. Tổng quan về ReactJS

Chương 2. JSX (JavaScript XML)

Chương 3. Components

Chương 4. Thực hành tạo Components

Chương 5. Kết luận

Tài liệu tham khảo

# Mở đầu

ReactJS là một thư viện JavaScript mạnh mẽ, được sử dụng rộng rãi trong phát triển giao diện người dùng (UI). Các thành phần cốt lõi của React bao gồm JSX và Components. Đây chính là nền tảng giúp React trở thành một công cụ hiệu quả, linh hoạt và dễ dàng bảo trì. Báo cáo này tập trung vào việc nghiên cứu hai thành phần cơ bản đó để hiểu rõ hơn cách ReactJS hoạt động và ứng dụng trong thực tiễn.

# Chương 1. Tổng quan về ReactJS

ReactJS được phát triển bởi Facebook (nay là Meta) vào năm 2011 và được công bố mã nguồn mở vào năm 2013. React không phải là một framework mà là một thư viện, tập trung vào việc xây dựng giao diện người dùng. Điểm mạnh của React nằm ở khả năng tái sử dụng component và tốc độ cập nhật giao diện nhờ Virtual DOM.

Trong kiến trúc của React, giao diện người dùng được chia thành nhiều component nhỏ. Mỗi component là một khối độc lập, dễ bảo trì, dễ tái sử dụng và có thể lồng ghép với nhau để tạo thành ứng dụng lớn.

# Chương 2. JSX (JavaScript XML)

JSX là một cú pháp mở rộng cho JavaScript, cho phép viết mã trông giống HTML ngay trong code JavaScript. Trình duyệt không hiểu JSX, do đó cần Babel biên dịch JSX thành các lệnh gọi hàm JavaScript thuần (React.createElement).

Ví dụ minh họa JSX:

const element = <h1>Xin chào, React!</h1>;  
// Babel biên dịch thành:  
const element = React.createElement("h1", null, "Xin chào, React!");

Ưu điểm của JSX:  
- Dễ đọc, dễ hiểu, trực quan.  
- Cho phép nhúng biểu thức JavaScript trong dấu ngoặc nhọn {}.  
- Giúp phát hiện lỗi sớm trong quá trình biên dịch.

# Chương 3. Components

Components là khối xây dựng cơ bản của ứng dụng React. Mỗi component có thể coi như một hàm độc lập, có thể tái sử dụng ở nhiều nơi. Có hai loại chính là Function Component và Class Component.

## 3.1. Function Component

Function Component chỉ là một hàm JavaScript nhận props và trả về JSX. Đây là cách viết component đơn giản và hiện đại, thường kết hợp với React Hooks.

Ví dụ:

function Welcome(props) {  
 return <h1>Xin chào, {props.name}!</h1>;  
}

Ví dụ với useState:

import { useState } from 'react';  
  
function Counter() {  
 const [count, setCount] = useState(0);  
 return (  
 <div>  
 <p>Bạn đã nhấp {count} lần</p>  
 <button onClick={() => setCount(count + 1)}>Nhấp</button>  
 </div>  
 );  
}

## 3.2. Class Component

Class Component là cách khai báo component bằng class trong JavaScript, kế thừa từ React.Component. Class Component có thể quản lý state và sử dụng các phương thức vòng đời (lifecycle).

Ví dụ:

import React, { Component } from 'react';  
  
class Counter extends Component {  
 constructor(props) {  
 super(props);  
 this.state = { count: 0 };  
 }  
  
 render() {  
 return (  
 <div>  
 <p>Bạn đã nhấp {this.state.count} lần</p>  
 <button onClick={() => this.setState({ count: this.state.count + 1 })}>Nhấp</button>  
 </div>  
 );  
 }  
}

## 3.3. So sánh Function và Class Component

Function Component:  
- Ngắn gọn, dễ viết.  
- Quản lý state bằng Hooks (useState, useEffect).  
  
Class Component:  
- Cú pháp dài hơn.  
- Quản lý state bằng this.state, setState.  
- Có các phương thức vòng đời như componentDidMount, componentDidUpdate.

# Chương 4. Thực hành tạo Components

1. Tạo dự án React bằng lệnh: npx create-react-app my-app  
2. Mở thư mục dự án trong Visual Studio Code.  
3. Sửa file App.js để sử dụng các component mới.  
4. Tạo một component Greeting và một component Counter.  
5. Chạy ứng dụng bằng npm start .

# Chương 5. Kết luận

JSX và Components là hai thành phần cơ bản nhất trong ReactJS. JSX giúp việc viết giao diện dễ dàng, trực quan. Components giúp tổ chức ứng dụng thành các khối độc lập, tái sử dụng và dễ bảo trì. Trong đó, Function Component kết hợp với Hooks là xu hướng phát triển hiện nay.

# Tài liệu tham khảo

1. ReactJS Official Documentation: https://react.dev  
2. W3Schools React Tutorial: https://www.w3schools.com/react/  
3. FreeCodeCamp: https://www.freecodecamp.org  
4. Medium Blogs về ReactJS