



FPT POLYTECHNIC



Conceive Design Implement Operate



THỰC HỌC – THỰC NGHIỆP

LẬP TRÌNH FRONT-END FRAMEWORK 2

**REACT COMPONENT: FUNCTION,
CLASS, PROPS**

- ⊙ Free style
- ⊙ Component: function & class
- ⊙ React component: props
- ⊙ Component nesting (component lồng nhau)
- ⊙ Component reuse (sử dụng lại component)






PHẦN 1

REACT COMPONENT: FUNCTION, CLASS

❑ Các mẫu free style

❖ Truy xuất đường link <https://semantic-ui.com/>


UI Docs

Getting Started

New in 2.4

Introduction

Integrations

Build Tools

Recipes

Glossary

Usage

Theming

Layouts

Globals

Reset

Site

Elements

Button

Container

Divider

Flag

Header

Icon

Image

Input

Types


Comments

A basic list of comments

These examples uses `ui segment` to create content segments. This is not required.

UI Views provide structured layouts, but do not hook up site specific code. You will need to add your own Javascript behaviors to hide and show the appropriate forms. For example, having a reply button open a reply form


Comments



Matt Today at 5:42PM

How artistic!


Reply



Elliot Fu Yesterday at 12:30AM

This has been very useful for my research. Thanks as well!


Reply



Jenny Hess Just now

Elliot you are always so right :)

Reply



Joe Henderson 5 days ago

Dude, this is awesome. Thanks so much

Reply

Comment

Types ▼

Comments

Content ▼

Avatar

Metadata

Actions

Reply Form

States ⌵

Variations ⌵

❑ Cách lấy và sử dụng free style

❖ Truy xuất đường link

<https://cdnjs.com/libraries/semantic-ui>

★ 48k GitHub 30 vulnerabilities
MIT licensed <http://semantic-ui.com/>
Tags: semantic, ui

Version 2.4.1 Asset Type All

Some files are hidden, click to show all files

https://cdnjs.cloudflare.com/ajax/libs/semantic-ui/2.4.1/semantic.min.js	Link Code Shield
https://cdnjs.cloudflare.com/ajax/libs/semantic-ui/2.4.1/semantic.min.css	Link Code Shield
https://cdnjs.cloudflare.com/ajax/libs/semantic-ui/2.4.1/components/accordion.min.css	Link Code Shield
https://cdnjs.cloudflare.com/ajax/libs/semantic-ui/2.4.1/components/accordion.min.js	Link Code Shield
https://cdnjs.cloudflare.com/ajax/libs/semantic-ui/2.4.1/components/ad.min.css	Link Code Shield
https://cdnjs.cloudflare.com/ajax/libs/semantic-ui/2.4.1/components/api.min.js	Link Code Shield

JS CommentDetail.js

<> index.html ✕

JS index.js ●

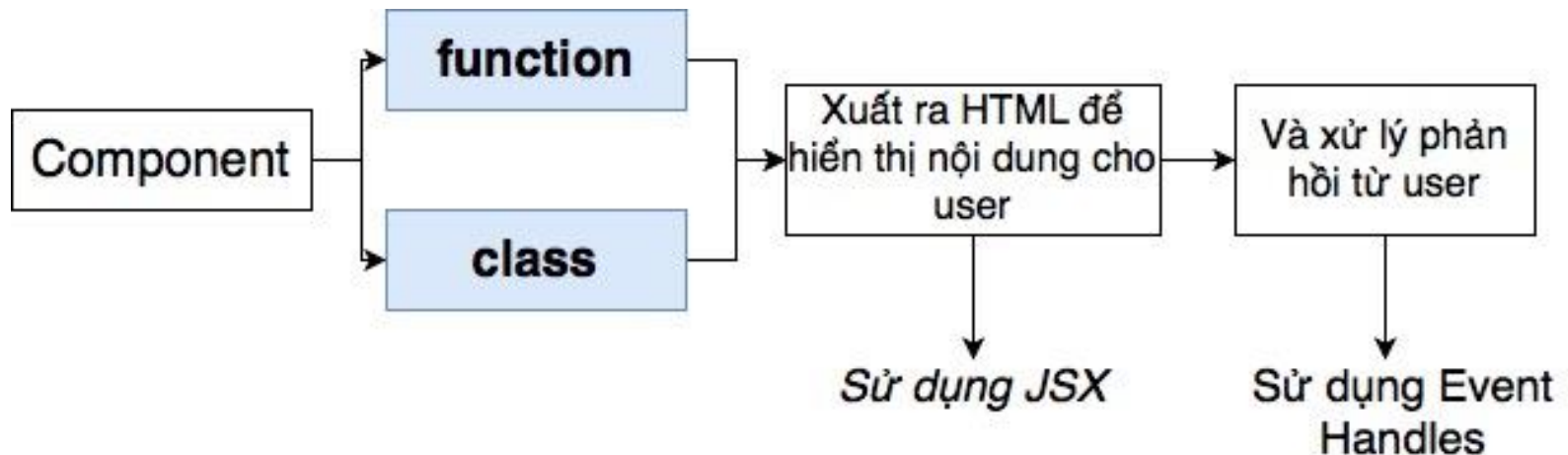
public > <> index.html > ...

```

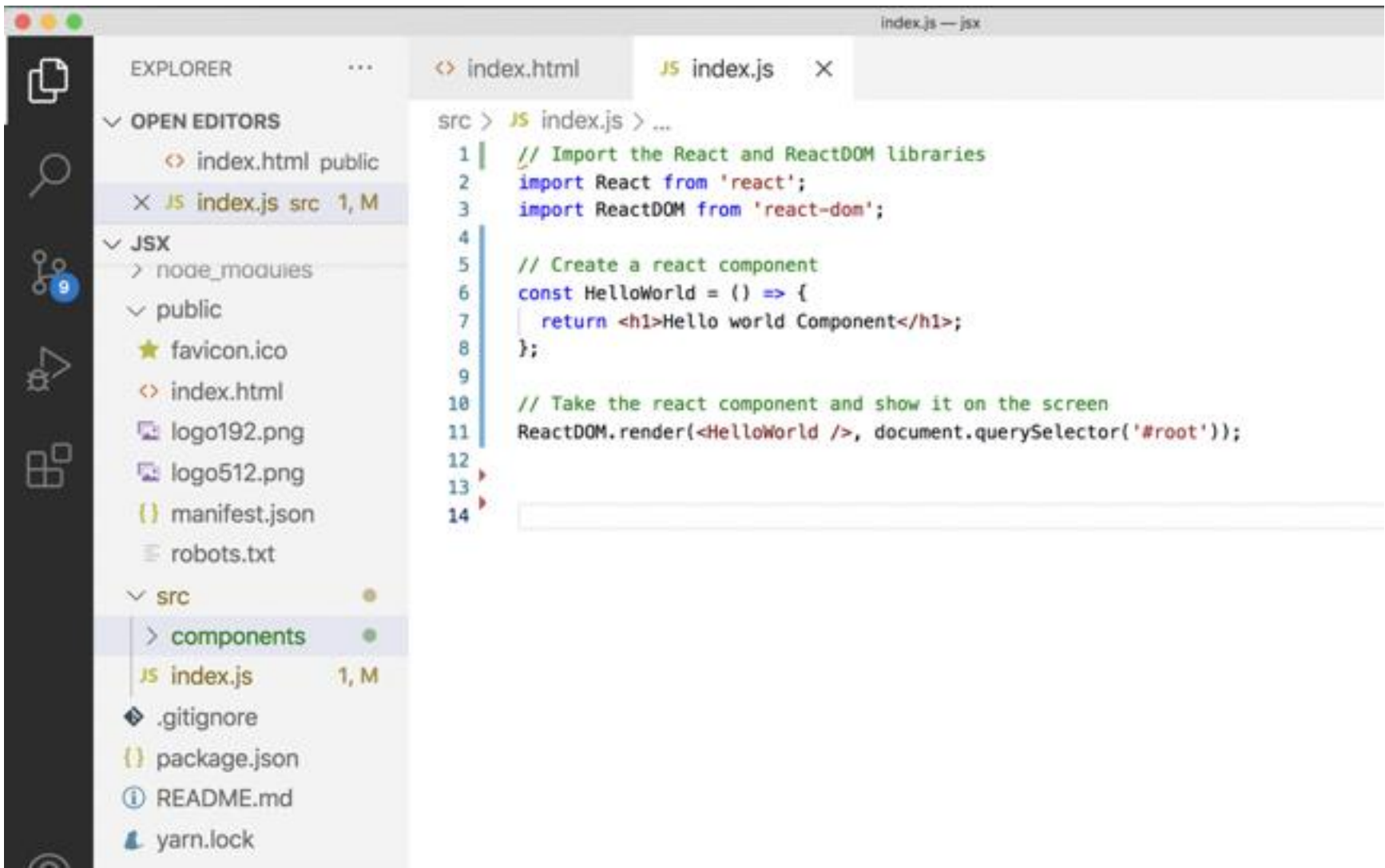
11      />
12      <link rel="apple-touch-icon" href="%PUBLIC_URL%/logo192.png" />
13      <!--
14      |   manifest.json provides metadata used when your web app is installed on a
15      |   user's mobile device or desktop. See https://developers.google.com/web/fundamentals/web-app-manifest/
16      |   -->
17      <link
18      |   rel="stylesheet"
19      |   href="https://cdnjs.cloudflare.com/ajax/libs/semantic-ui/2.4.1/semantic.min.css"
20      />
21
22      <link rel="manifest" href="%PUBLIC_URL%/manifest.json" />
23      <!--
24      |   Notice the use of %PUBLIC_URL% in the tags above.
25      |   It will be replaced with the URL of the `public` folder during the build.
26      |   Only files inside the `public` folder can be referenced from the HTML.
27

```

- ❑ Component: là một khối code độc lập, có thể tái sử dụng, chia UI thành nhiều phần nhỏ
- ❑ Tạo component: để dễ dàng quản lý và tái sử dụng
- ❑ Component trong react thường được viết theo 2 loại chính: function và class



❑ Function component đơn giản



The screenshot shows a code editor with a file explorer on the left and a code editor on the right. The file explorer shows a project structure with a 'src' directory containing a 'components' directory. The 'index.js' file is selected in the 'components' directory. The code editor shows the following code:

```
src > JS index.js > ...
1 | // Import the React and ReactDOM libraries
2 | import React from 'react';
3 | import ReactDOM from 'react-dom';
4 |
5 | // Create a react component
6 | const HelloWorld = () => {
7 |   return <h1>Hello world Component</h1>;
8 | };
9 |
10 | // Take the react component and show it on the screen
11 | ReactDOM.render(<HelloWorld />, document.querySelector('#root'));
12 |
13 |
14 |
```



```
const App = () => {
```

```
  return (
```

```
    <div className="ui container comments">
```

```
      <div className="comment">
```

```
        <a href="/" className="avatar">
```

```
          <img alt="avatar" />
```

```
        </a>
```

```
        <div className="content">
```

```
          <a href="/" className="author">
```

```
            Sam
```

```
          </a>
```

```
          <div className="metadata">
```

```
            <span className="date">Today at 6:00PM</span>
```

```
          </div>
```

```
          <div className="text">Nice blog post!</div>
```

```
        </div>
```

```
      </div>
```

```
    </div>
```


```
  );
```

```
};
```

Free style semantic-ui

Nội dung
JSX

TRÍCH JSX > NEW COMPONENT

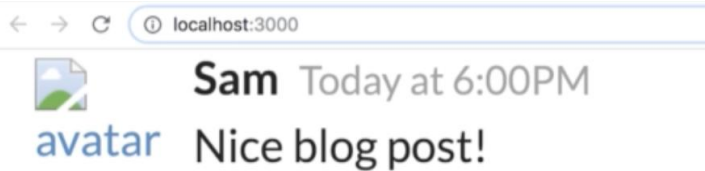


Xác định nội dung JSX được duplicate (tạo bản sao)

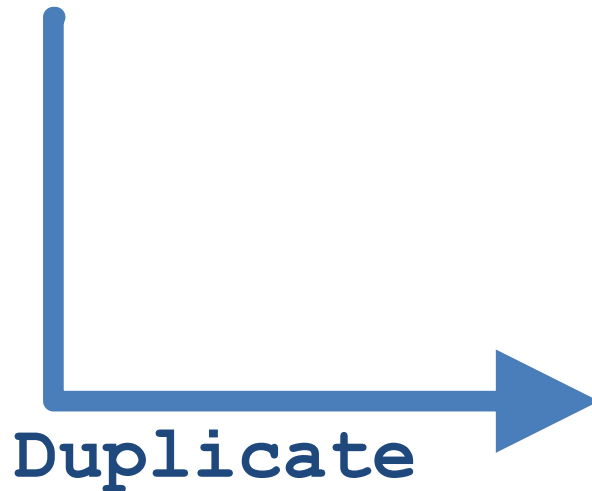
Mục đích của khối JSX đó là gì? Nghĩ ra một cái tên (mô tả chức năng,...) cho nó

Tạo 1 file mới chứa nội dung các thành phần này

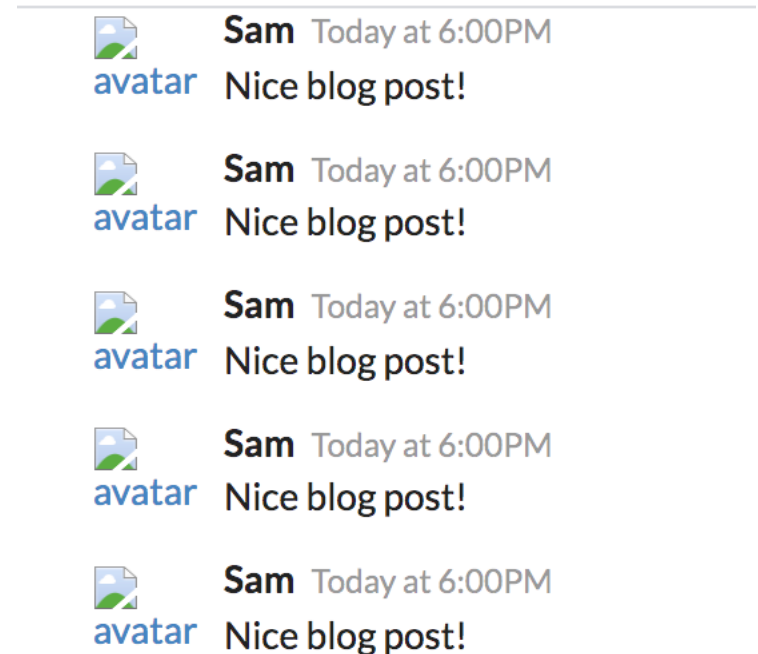
Tạo 1 new component trong file mới này, dán nội dung JSX vào



Single



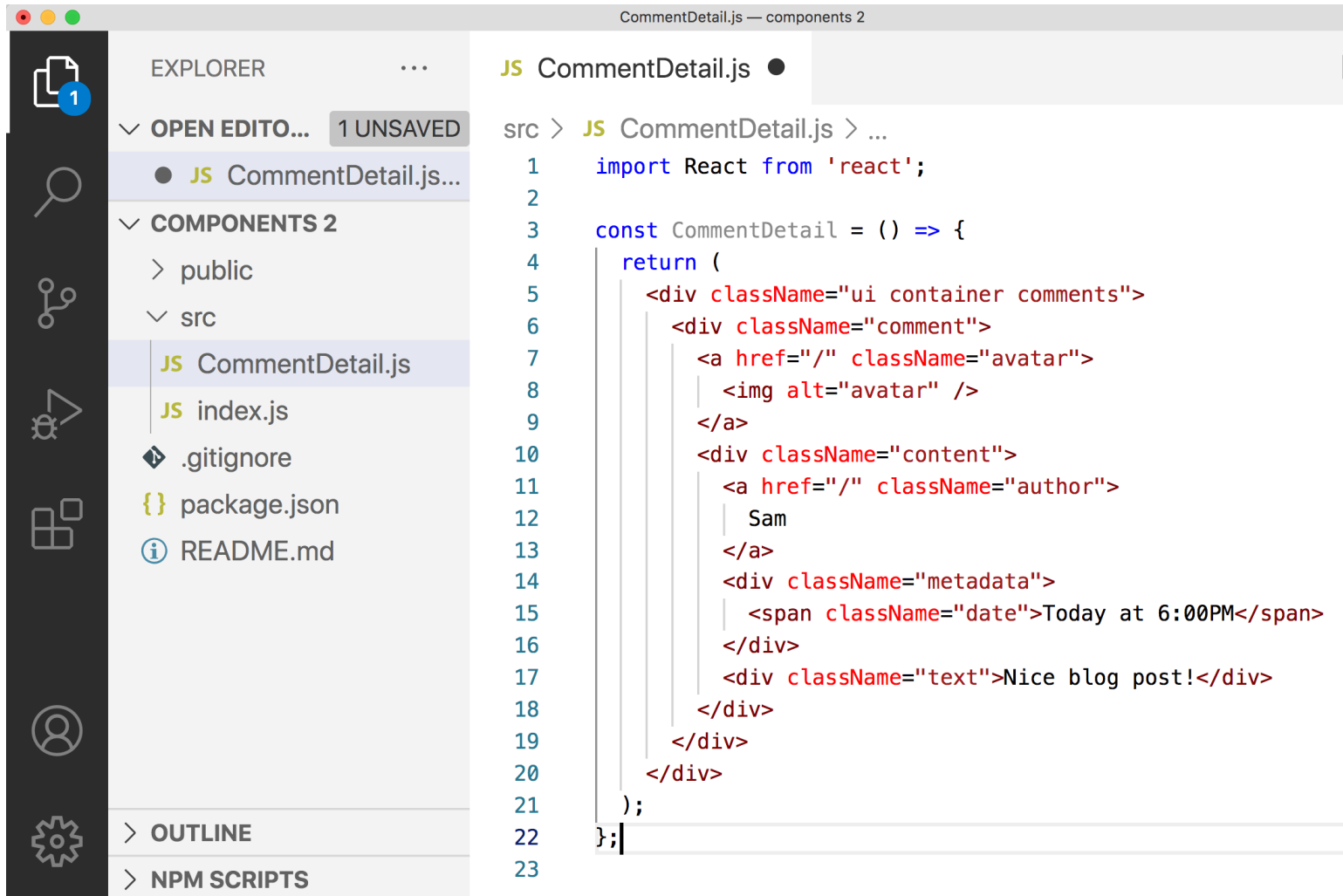
Duplicate



□ Bước 3



□ Bước 4



The screenshot shows the Visual Studio Code interface with the Explorer sidebar on the left and the Code Editor on the right. The Explorer sidebar is expanded to show the 'src' directory, which contains 'CommentDetail.js' and 'index.js'. The 'CommentDetail.js' file is selected. The Code Editor shows the content of 'CommentDetail.js', which is a React component. The component is named 'CommentDetail' and is a functional component. It uses the 'import React from 'react'' syntax. The component's return statement is a JSX element with a 'className' of 'ui container comments'. It contains a 'comment' div with an 'avatar' link, a 'content' div with an 'author' link, and a 'text' div. The component is exported as a default export.

```
src > JS CommentDetail.js > ...
1  import React from 'react';
2
3  const CommentDetail = () => {
4    return (
5      <div className="ui container comments">
6        <div className="comment">
7          <a href="/" className="avatar">
8            <img alt="avatar" />
9          </a>
10         <div className="content">
11           <a href="/" className="author">
12             Sam
13           </a>
14           <div className="metadata">
15             <span className="date">Today at 6:00PM</span>
16           </div>
17           <div className="text">Nice blog post!</div>
18         </div>
19       </div>
20     </div>
21   );
22 };
23
```

❑ Component nesting (lồng nhau)

index.js

```
import CommentDetail from  
  './CommentDetail'
```

App Component

CommentDetail.js

CommentDetail
Component

```
export default CommentDetail
```



JS CommentDetail.js ●

src > JS CommentDetail.js > ...

```

1  import React from 'react';
2
3  const CommentDetail = () => {
4    return (
5      <div className="ui container comments">
6        <div className="comment">
7          <a href="/" className="avatar">
8            <img alt="avatar" />
9          </a>
10         <div className="content">
11           <a href="/" className="author">
12             Sam
13           </a>
14           <div className="metadata">
15             <span className="date">Today at 6:00PM</span>
16           </div>
17           <div className="text">Nice blog post!</div>
18         </div>
19       </div>
20     </div>
21   );
22 };
23
24 export default CommentDetail;
25
26

```

EXPLORER

...

JS CommentDetail.js

JS index.js

●

✓ OPEN EDITO... 1 UNSAVED

JS CommentDetail.js...

● JS index.js src

✓ COMPONENTS 3

> public

✓ src

JS CommentDetail.js

JS index.js

.gitignore

{} package.json

ⓘ README.md

src > JS index.js > ...

```

1  import React from 'react';
2  import ReactDOM from 'react-dom';
3  import CommentDetail from './CommentDetail';
4
5  const App = () => {
6    return (
7      <div className="ui container comments">
8        <CommentDetail />
9        <CommentDetail />
10       <CommentDetail />
11       <CommentDetail />
12       <CommentDetail />
13     </div>
14   );
15 };
16
17 ReactDOM.render(<App />, document.querySelector('#root'));
18

```


Quy tắc tạo class component

Phải là 1 Javascript class

Phải sử dụng extend
React.Component

Phải dùng phương thức render khi
return jsx

❑ Ví dụ 01

```
import React from 'react';
import ReactDOM from 'react-dom';

class Car extends React.Component {
  render() {
    return <h2>Hi, I am a Car!</h2>;
  }
}

ReactDOM.render(<Car />, document.getElementById('root'));
```

❑ Ví dụ 02

```
class Comment extends React.Component {
  render(){
    return (
      <div className="ui container comments">
        <CommentDetail />
        <CommentDetail />
        <CommentDetail />
        <CommentDetail />
        <CommentDetail />
        </div>);
  }
}

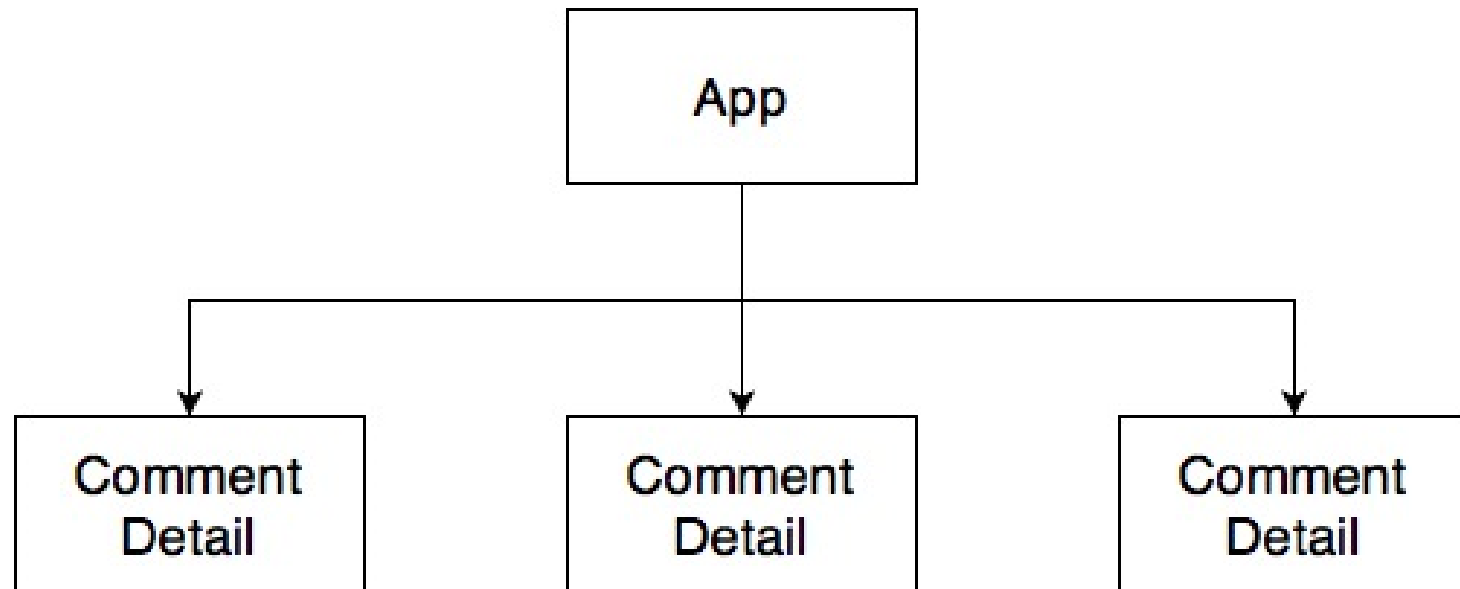
ReactDOM.render(<Comment />, document.querySelector('#root'));
```

demo



PHẦN 2

REACT COMPONENT: PROPS



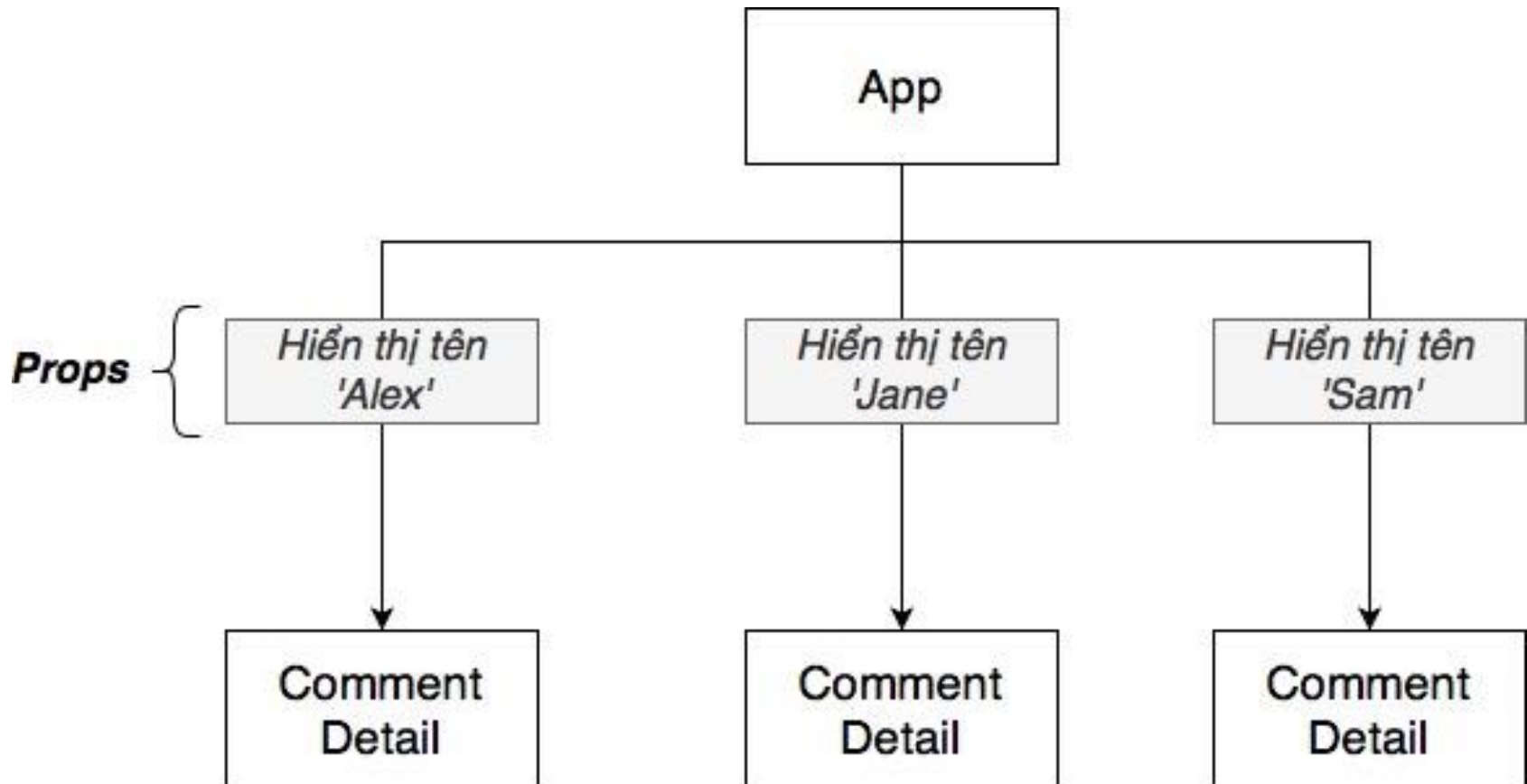
Bảng phân cấp Component

Props(properties)

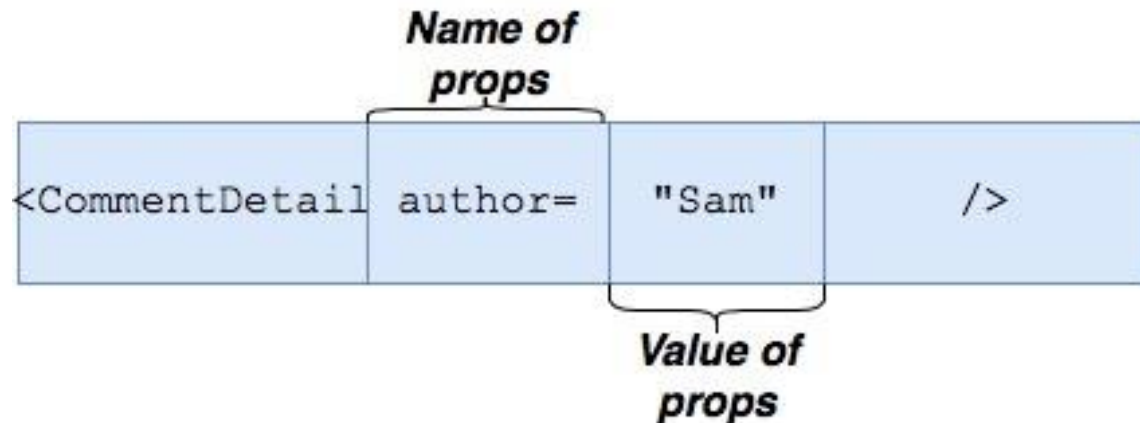
Hệ thống gửi dữ liệu từ parent component đến child component



Mục tiêu là để tùy chỉnh hoặc cấu hình child component



- ❑ Cung cấp props từ parent đến child (React props giống như đối số trong hàm javascript và thuộc tính trong HTML)
- ❑ Gửi props



- ❑ Cách pháp nhận props

functional component

<code>props.</code>	<code><name of props></code>
---------------------	------------------------------------

class component

<code>this.props.</code>	<code><name of props></code>
--------------------------	------------------------------------

□ Gửi 1 props (index.js)

```
const App = () => {  
  return (  
    <div className="ui container comments">  
      <CommentDetail author="Sam" />  
      <CommentDetail author="Alex" />  
      <CommentDetail author="Jane" />  
    </div>  
  );  
};
```

```
ReactDOM.render(<App />, document.querySelector('#root'));
```

□ Gửi nhiều props (multi props)

```
const App = () => {  
  return (  
    <div className="ui container comments">  
      <CommentDetail author="Sam" timeAgo="Today at 4:45PM" />  
      <CommentDetail author="Alex" timeAgo="Today at 2:00AM" />  
      <CommentDetail author="Jane" timeAgo="Yesterday at 5:00PM" />  
    </div>  
  );  
};
```

```
ReactDOM.render(<App />, document.querySelector('#root'));
```

□ nhận props (CommentDetail.js)

JS CommentDetail.js ×

src > JS CommentDetail.js > ...

```
1  import React from 'react';
2  import faker from 'faker';
3
4  const CommentDetail = props => {
5    return (
6      <div className="comment">
7        <a href="/" className="avatar">
8          <img alt="avatar" src={faker.image.avatar()} />
9        </a>
10       <div className="content">
11         <a href="/" className="author">
12           {props.author}
13         </a>
14         <div className="metadata">
15           <span className="date">Today at 6:00PM</span>
16         </div>
17         <div className="text">Nice blog post!</div>
18       </div>
19     </div>
20   );
21 };
22
23 export default CommentDetail;
24
```

□ nhận props (CommentDetail.js)

```
class CommentDetail extends React.Component {  
  render() {  
    return (  
      <div className="comment">  
        <a href="/" className="avatar">  
          <img alt="avatar" src={faker.image.avatar()} />  
        </a>  
        <div className="content">  
          <a href="/" className="author">  
            {this.props.author}  
          </a>  
          <div className="metadata">  
            <span className="date">Today at 6:00PM</span>  
          </div>  
          <div className="text">Nice blog post!</div>  
        </div>  
      </div>  
    );  
  }  
}
```

- Nếu trong class component có structure constructor function. Nhận props như sau

```
class CommentDetail extends React.Component {  
  constructor(props) {  
    super();  
  }  
  render() {  
    return (  
      <div className="comment">  
        <a href="/" className="avatar">  
          <img alt="avatar" src={faker.image.avatar()} />  
        </a>  
        <div className="content">  
          <a href="/" className="author">  
            {this.props.author}  
          </a>  
          <div className="metadata">  
            <span className="date">Today at 6:00PM</span>  
          </div>  
          <div className="text">Nice blog post!</div>  
        </div>  
      </div>  
    );  
  }  
}
```

❑ Cách 01: Sử dụng cú pháp

functional component

props.	<name of props>	"<default value>"
--------	-----------------	-------------------

class component

this.props.	<name of props>	"<default value>"
-------------	-----------------	-------------------

❑ Ví dụ: function component

```
const CommentDetail = props => {  
  return (  
    <div className="comment">  
      <a href="/" className="avatar">  
        <img alt="avatar" src={props.avatar} />  
      </a>  
      <div className="content">  
        <p> {props.name || "Default props"} </p>  
        <a href="/" className="author">  
          {props.author}</a>  
        <div className="metadata">  
          <span className="date">{props.timeAgo}</span>  
        </div>  
        <div className="text">{props.content}</div>  
      </div>  
    </div>  
  );  
};
```

localhost:3002



Default props

Sam Today at 6:00PM
Nice blog post!



Default props

Alex Today at 6:00PM
Nice blog post!



Default props

Jane Today at 6:00PM
Nice blog post!

❑ Ví dụ: class component

```
class CommentDetail extends React.Component {
  constructor(props) {
    super();
  }
  render() {
    return (
      <div className="comment">
        <a href="/" className="avatar">
          <img alt="avatar" src={faker.image.avatar()} />
        </a>
        <div className="content">
          <p> {this.props.name || "Default props"} </p>
          <a href="/" className="author">
            {this.props.author}
          </a>
          <div className="metadata">
            <span className="date">Today at 6:00PM</span>
          </div>
          <div className="text">Nice blog post!</div>
        </div>
      </div>
    );
  }
}
```

localhost:3002



Default props

Sam Today at 6:00PM
Nice blog post!



Default props

Alex Today at 6:00PM
Nice blog post!



Default props

Jane Today at 6:00PM
Nice blog post!

❑ Cách 02:

```
class ReactComp extends React.Component {}  
ReactComp.defaultProps = {}
```

Hoặc

```
class ReactComp extends React.Component {  
    static defaultProps = {}  
}
```

Hoặc

```
function Reactcomp(props) {}  
ReactComp.defaultProps = {}
```

□ Ví dụ:

```
const CommentDetail = props => {
  return (
    <div className="comment">
      <a href="/" className="avatar">
        <img alt="avatar" src={props.avatar} />
      </a>
      <div className="content">
        <p> {props.name} </p>
        <a href="/" className="author">
          {props.author}
        </a>
        <div className="metadata">
          <span className="date">{props.timeAgo}</span>
        </div>
        <div className="text">{props.content}</div>
      </div>
    </div>
  );
};

CommentDetail.defaultProps = {
  name: 'Default props'
}
```

□ PropTypes: chỉ định kiểu dữ liệu của props

```
import React from 'react';  
import PropTypes from 'prop-types'  
  
CommentDetail.propTypes = {  
  | name: PropTypes.string  
}
```

□ Proptypes với defaultProps

```
CommentDetail.defaultProps = {  
  | name: PropTypes.string  
}
```

□ Tái sử dụng component

Warning!

Are you sure you want to do this?

Approve

Reject



Sam Today at 4:45PM

Nice blog post

Approve

Reject



Alex Today at 2:00AM

I like the subject

Approve

Reject



Jane Yesterday at 5:00PM

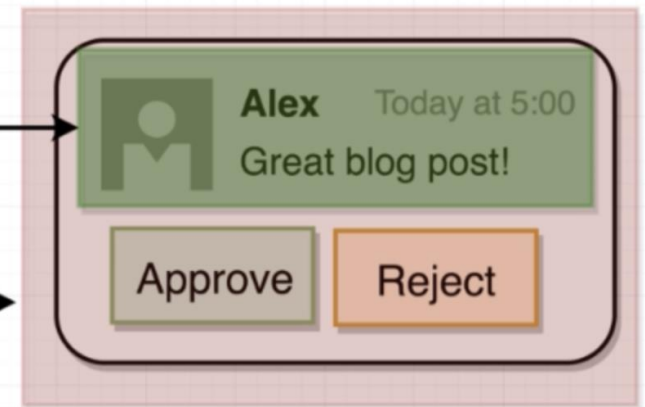
I like the writing

Approve

Reject

CommentDetail →

ApprovalCard →



COMPONENTS 2

> node_modules

> public

src

JS ApprovalCard.js — 1

JS CommentDetail.js

JS index.js

📁 .gitignore

{ } package-lock.json

{ } package.json

📖 README.md

```
const ApprovalCard = props => {
  return (
    <div className="ui card">
      <div className="content">{props.children}</div>
      <div className="extra content">
        <div className="ui two buttons">
          <div className="ui basic green button">Approve</div>
          <div className="ui basic red button">Reject</div>
        </div>
      </div>
    </div>
  );
};

export default ApprovalCard;
```

```
<ApprovalCard>
  <div>
    <h4>Warning!</h4>
    Are you sure you want to do this?
  </div>
</ApprovalCard>
```

demo

- ☑ Component: function & class
- ☑ React component: props
- ☑ Component nesting (component lồng nhau)
- ☑ Component reuse (sử dụng lại component)



thank
you!