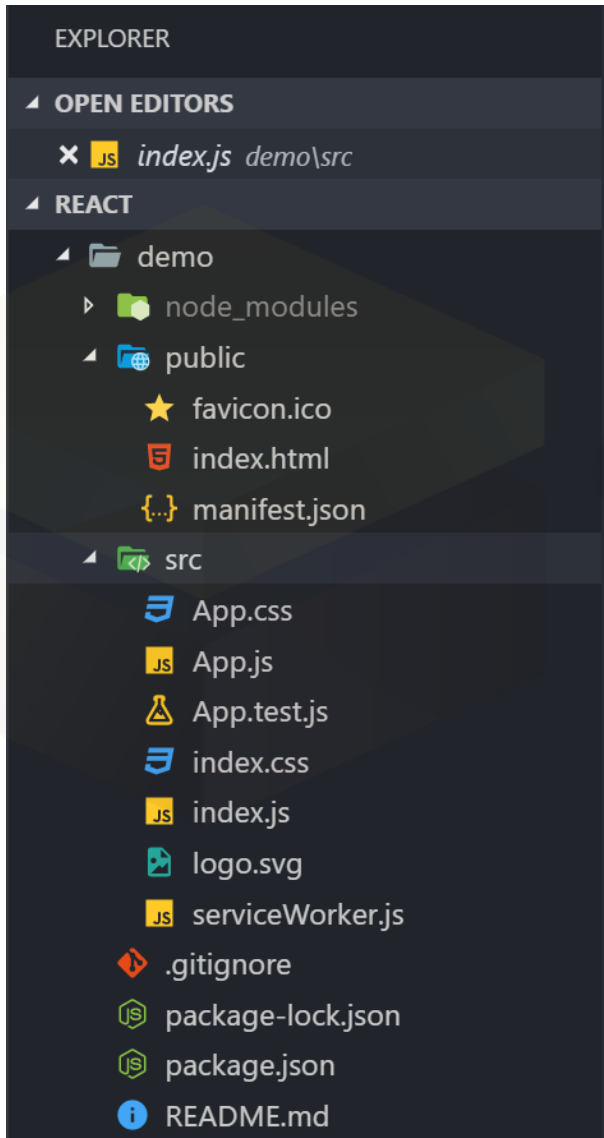


# React JS

ĐÀO TẠO CHUYÊN GIA LẬP TRÌNH

## ❑ Cấu trúc thư mục sau khi khởi tạo dự án bởi create-react-app



- ❑ ***node\_modules***: chứa các module được cài vào project
- ❑ ***Public***: chứa file index.html và hình ảnh
- ❑ ***Src***: chứa các component của ứng dụng
- ❑ ***Package.json***: lưu lại các thông tin của project và các thư viện được cài vào.

# ❑ React hoạt động như thế nào ? (Luồng đi của ứng dụng)

```
index.html
demo ▸ public ▸ index.html ▸ html ▸ body
9   manifest.json provides metadata used when your web app is added to
10  homescreen on Android. See https://developers.google.com/web/fundam
11  -->
12  <link rel="manifest" href="%PUBLIC_URL%/manifest.json">
13  <title>React App</title>
14  </head>
15  <body>
16    <noscript>
17      You need to enable JavaScript to run this app.
18    </noscript>
19    <div id="root"></div>
20  </body>
21  </html>
```

```
index.js
demo ▸ src ▸ JS index.js
1  import React from 'react';
2  import ReactDOM from 'react-dom';
3  import './index.css';
4
5  import App from './App';
6
7  import * as serviceWorker from './serviceWorker';
8
9  ReactDOM.render( <App /> , document.getElementById('root'));
10
11  // If you want your app to work offline and load faster, you can change
12  // unregister() to register() below. Note this comes with some pitfalls.
13  // Learn more about service workers: http://bit.ly/CRA-PWA
14  serviceWorker.unregister();
```

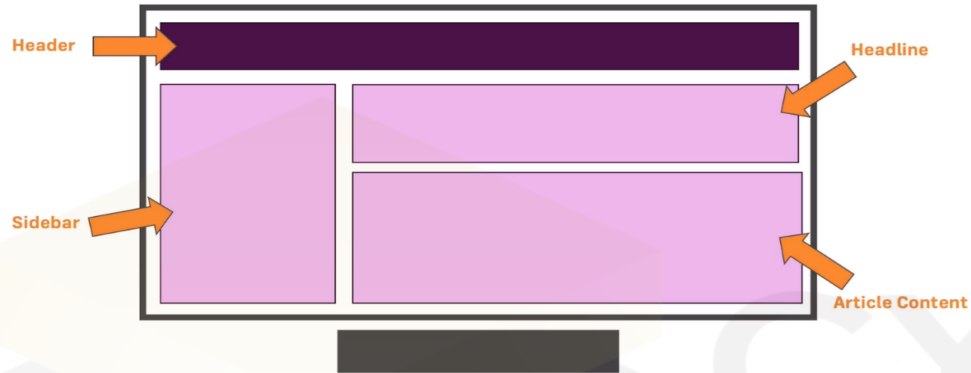
```
App.js
demo ▸ src ▸ App.js ▸ App ▸ render
4
5  class App extends Component {
6    render() {
7      return (
8
9        <div className="App">
10          <header className="App-header">
11            <img src={logo} className="App-logo" alt="logo" />
12            <p>
13              Edit <code>src/App.js</code> and save to reload.
14            </p>
15            <a
16              className="App-link"
17              href="https://reactjs.org"
18              target="_blank"
19              rel="noopener noreferrer"
20            >
21              Learn React
22            </a>
23          </header>
24        </div>
25      );
26    }
27  }
```

- ❑ Đầu tiên, browser sẽ đọc được trang index.html
- ❑ Tiếp theo sẽ đọc tới file index.js, trong file này, ReactDOM được sử dụng để render nội dung của của app component ra div root ở ngoài HTML
- ❑ App.js chính là component gốc của toàn ứng dụng

# Component là gì ?

- React xây dựng giao diện ứng dụng theo kiến trúc **component**.
- React hỗ trợ xây dựng SPAs (Single page application).

## Components?



TFL Tube Tracker

<Network />

Bakerloo Line  
Stations <Line /> Go

Central Line  
Stations <Line /> Go

Circle Line  
Stations <Line /> Go

District Line  
Stations <Line /> Go

Hammersmith & City Line  
Stations <Line /> Go

Jubilee Line  
Stations <Line /> Go

Metropolitan Line  
Stations <Line /> Go

Station Name <Predictions />

Platform 1 <DepartureBoard />

Destination	Due	Current location
White City	0:00	At Platform
Ealing Broadway	<Trains />	Holland Park
West Ruislip	4:30	Notting Hill Gate
Ealing Broadway	6:00	Queensway

Platform 2 <DepartureBoard />

Destination	Due	Current location
White City	0:00	At Platform
Ealing Broadway	<Trains />	Holland Park
West Ruislip	4:30	Notting Hill Gate
Ealing Broadway	6:00	Queensway

Standard Lightning Components on an Opportunity Page

HIGHLIGHTS PANEL

SALES PATH BAR

ACTIVITY CHATTER DETAILS

Products (0)

Notes & Attachments (0)

Contact Roles (0)

Stage History (3)

**TABS**  
(TABS is a Lightning Component containing 3 other Lightning Components inside of it: DETAILS, CHATTER and ACTIVITY)

**RELATED LISTS**

# ❑ Component

- ❑ Component biểu diễn giao diện UI (file.html).
- ❑ Nói 1 cách đơn giản 1 component là 1 thẻ do mình định nghĩa trong thẻ đó chứa các nội dung html do mình biên soạn.
- ❑ Cấu trúc 1 component bao gồm:
- ❑ Có 2 loại component:
  - ❑ Stateless component (functional component)
  - ❑ Stateful component (class component)

```
1 import React, { Component } from 'react';
2
3 import './App.css';
4
5 class App extends Component {
6   render() {
7     return (
8       // Nội dung giao diện html sẽ được viết tại đây
9     );
10  }
11 }
12
13 export default App;
```

Annotations in the image:

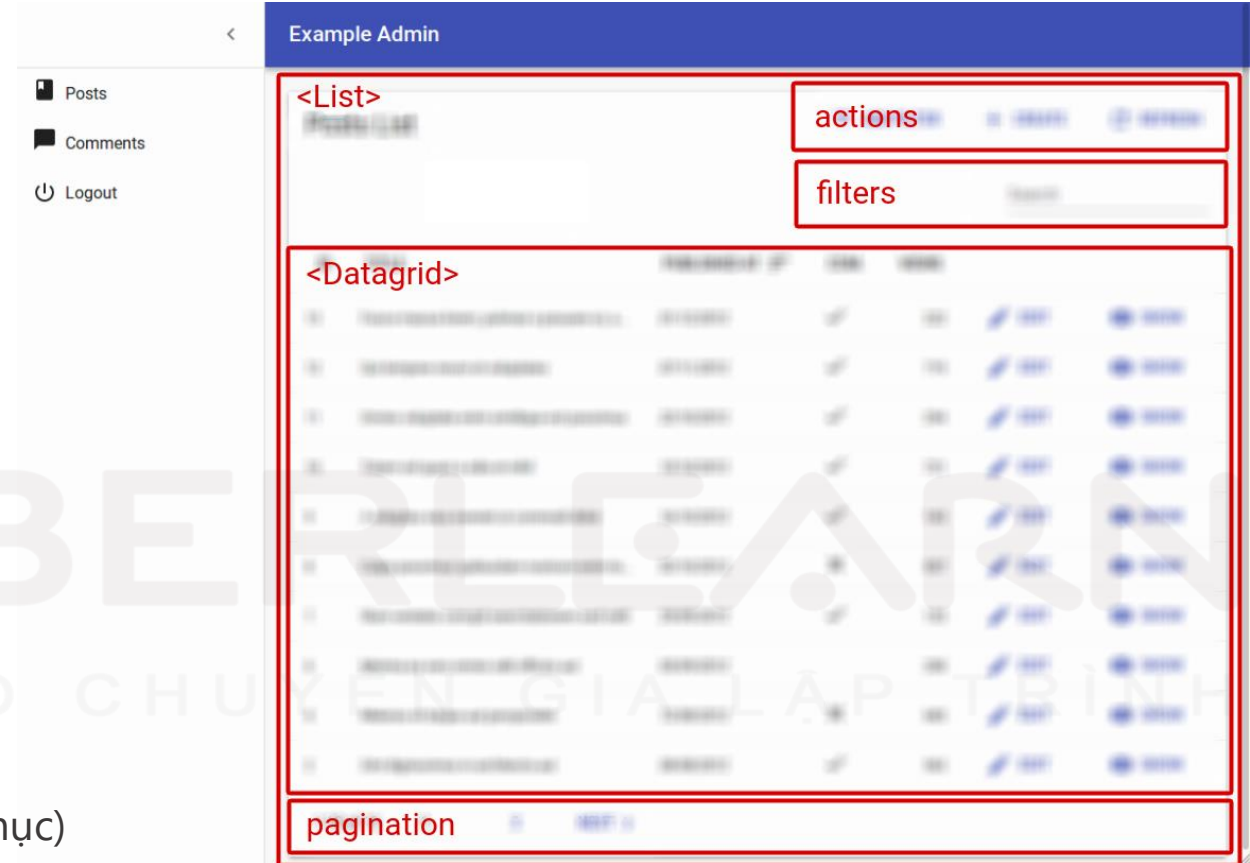
- Line 1: `import React, { Component } from 'react';` (highlighted in red)
- Line 3: `import './App.css';` (highlighted in green) with text: "import vào file css dùng để định dạng cho component"
- Line 5: `class App extends Component {` (highlighted in blue) with text: "component thực chất chỉ là một class được kế thừa từ Component của react"
- Line 8: `return (` (highlighted in blue) with text: "Nội dung giao diện html sẽ được viết tại đây"

Bottom right text: ❑ Khi component được load, hàm render sẽ chạy đầu tiên và return và một đoạn mã html. Đây chính là giao diện được hiển thị lên browser

Tại sao phải chia component ?

## ❑ Mục đích của việc chia component

- ✓ Đóng gói chức năng ứng với giao diện ( dễ tổ chức thư mục)
- ✓ Tái sử dụng (rút ngắn thời gian làm dự án)
- ✓ Dễ bảo trì nâng cấp, phát triển ( fixbug, test , thêm | bớt | thay đổi tính năng)
- ✓ Ngoài ra còn giúp cho việc tổ chức mã nguồn dễ dàng hơn (Code ngắn gọn, component nào nằm ở thư mục đó dễ tìm kiếm – thay thế)
- ✓ Tái sử dụng trên nhiều dự án khác nhau và còn nhiều lợi ích khác nữa.



## ❖ class component

```
1 import React from 'react';
2 import PropTypes from 'prop-types';
3
4 class Hello extends React.Component {
5   render() {
6     const {greeting, firstName} = this.props;
7     return (
8       <div>
9         {greeting} {firstName}
10      </div>
11    );
12  }
13 }
14
15 export default Hello;
```

## ❖ functional component

```
1 import React from 'react';
2 import PropTypes from 'prop-types';
3
4 function Hello({greeting, firstName}) {
5   return (
6     <div>
7       {greeting} {firstName}
8     </div>
9   );
10 }
11
12 export default Hello;
```

- ✓ Trước khi **react class component** và **react functional component** có nhiều khác biệt do react **class component** có nhiều thuộc tính, phương thức hỗ trợ việc render dữ liệu. Tuy nhiên thời điểm hiện tại React **functional component** cũng đã có các thư viện hooks hỗ trợ các tính năng mà react class có thể làm được.



Khóa học này chúng ta sẽ xoay quanh tìm hiểu về react classcomponent

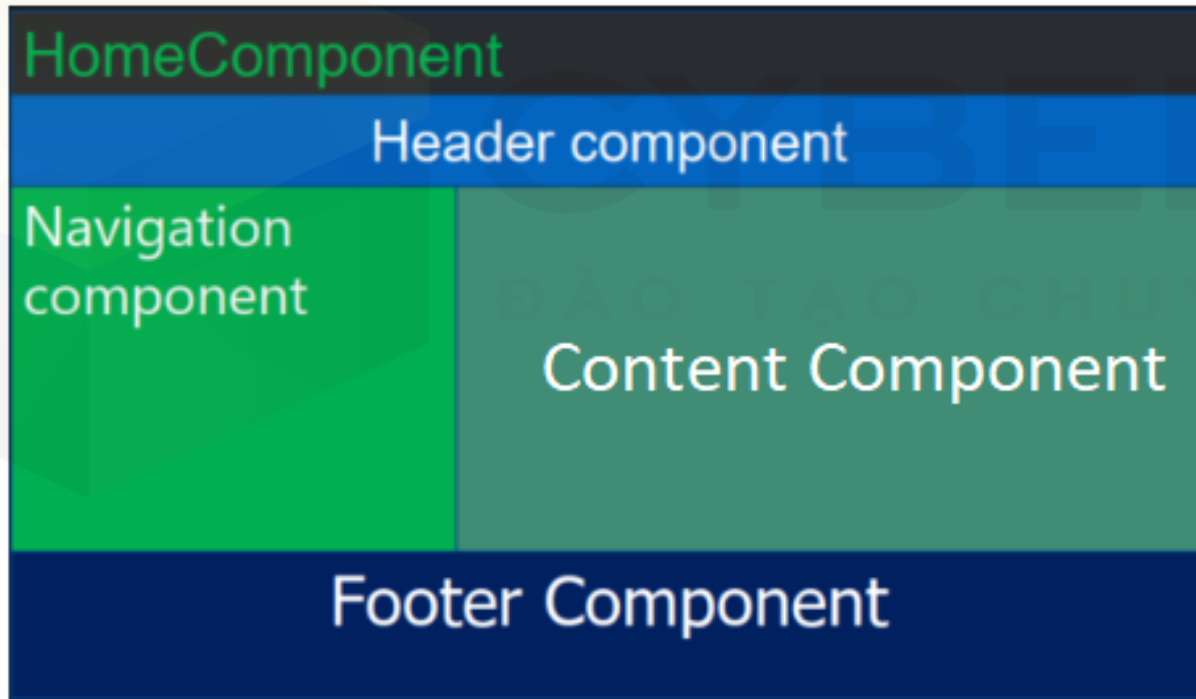
```
class ClassComponent extends React.Component {
  render() {
    return <div>{this.props.name}</div>;
  }
}
```

VS

```
const StatelessComponent = props => <div>{props.name}</div>;
```

# THỰC HÀNH

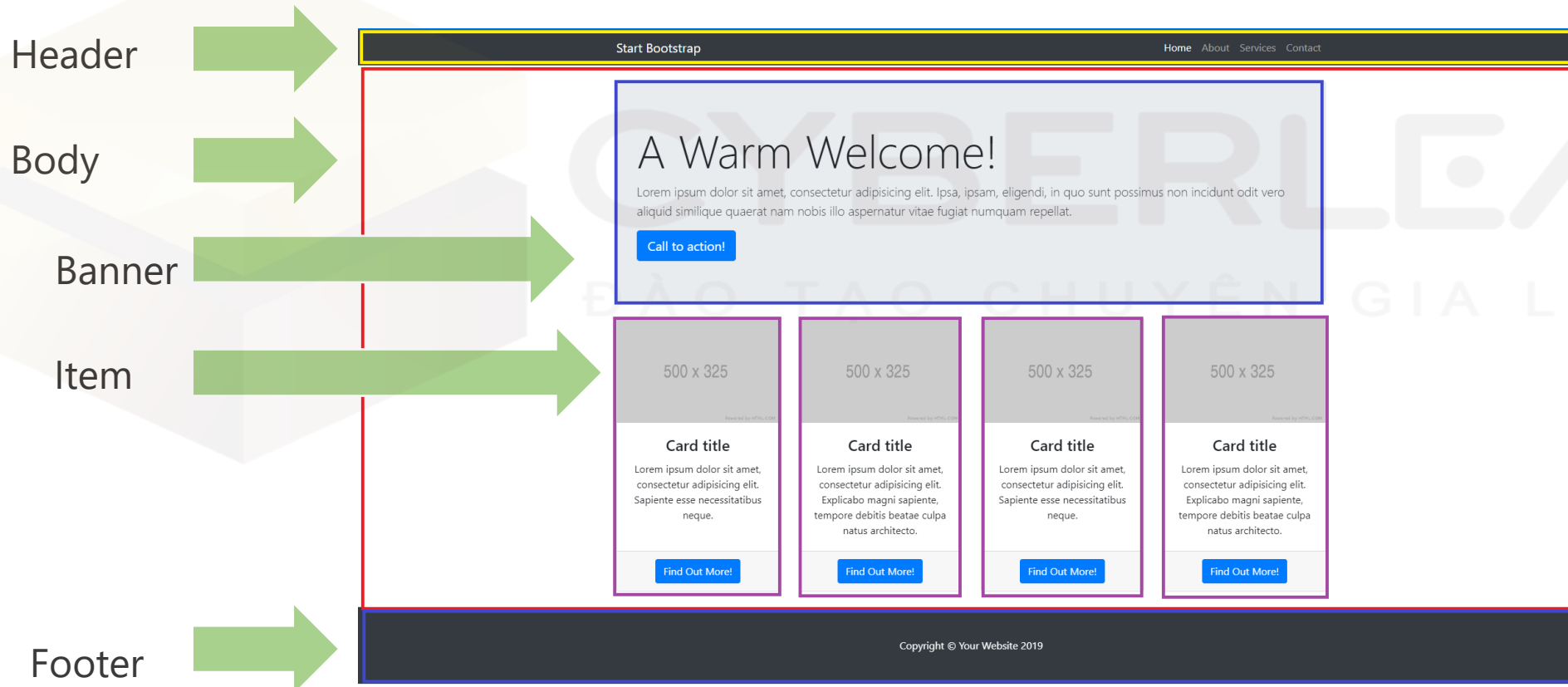
- ✓ Sử dụng cdn bootstrap
- ✓ Thực hành 1:
  - ✓ Tạo 1 component **Header** với react class component chứa tiêu đề là cyberlearn.vn
  - ✓ Tạo 1 component **Product** với react functional component nội dung chứa card bootstrap
- ✓ Thực hành 2:
  - ✓ Sử dụng bootstrap tạo các component như hình bên dưới





# BÀI TẬP

- ✓ Sử dụng cdn bootstrap
- ✓ Link layout: <https://startbootstrap.com/previews/heroic-features/>
- ✓ Dựa vào layout có sẵn các bạn hãy tạo 1 folder BaiTapLayoutComponent
- ✓ Yêu cầu tạo 1 component tên <BaiTapThucHanhLayout> Bên trong tổ chức nội dung các component như hình bên dưới





THỰC HÀNH NGAY THÔI NÀO !

CYBERLEARN  
ĐÀO TẠO CHUYÊN GIA LẬP TRÌNH