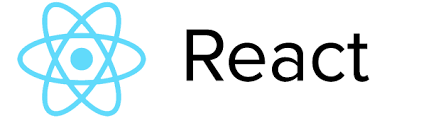
TỔNG QUAN VỀ REACTJS

1. Giới thiệu về React



1. ReactJS là gì ?

ReactJS là thư viện Javascript mã nguồn mở, để phát triển giao diện người dùng (UI) trong ứng dụng web. React được phát triển và phát hành bởi Facebook – do Jordan Walke tạo ra. React lần đầu tiên được triển khai cho ứng dụng Newsfeed của Facebook vào năm 2011, sau đó được triển khai cho Instagram.com vào năm 2012. React được mã nguồn mở tại JSConf US vào tháng 5 năm 2013.

ReactJS được các tập đoàn khổng lồ như: Netflix, Airbnb, American Express, Facebook, WhatsApp, Instagram,… dử dụng để xây dựng giao diện người dùng.

Mục đích cốt lỗi của ReactJS chính là giúp cho giao diện của các thiết bị có thể phản ứng một cách nhanh nhện hơn, hay nói các khác là gia tăng độ mượt cho website, ứng dụng.

Hiện tại, Facebook đang tiếp tục phát triển thư viện ReactJS và cải thiện nó bằng cách sửa lỗi và giới thiệu các tính năng mới.

1. Những tính năng nổi bật của ReactJS

* Người dùng hoàn toàn có thể sử dụng React để viết ứng dụng trực tiếp bằng JavaScript.
* Biến các thành phần UI phức tạp trở thành những Component độc lập.
* Chuyển các dữ liệu đã được tùy biến đến một UI Component cụ thể.
* Giúp lập trình viên thay đổi trạng thái cho nhiều Component con và không gây ảnh hưởng đến các Component gốc dù đang trong trạng thái Ststeful.
* Khả năng xác định chính xác khi nào cần render lại cũng như khi nào bỏ đi phần tử DOM.

1. Các thành phần của React

ReactJS là một thư viện Javascript hỗ trợ xây dựng tầng Views trong mô hình MVC. Với công nghệ ReactJS, người dùng có thể phát triển và xây dựng website bằng cách sử dụng Javascript để thao tác với HTML, công nghệ còn được tăng cường với Virtual-DOM.

* JavaScript

Như phần đầu giới thiệu, Javascript là ngôn ngữ được sử dụng để phát triển React. Để sử dụng ReactJS một cách hiệu quả, lập trình viên ít nhất phải hiểu về các khái niện như: objects, prototype và callback.

* Virtual-DOM

Virtual-DOM là định dạng dữ liệu của Javascritp, với khối lượng nhẹ và được dùng để thể hiện nội dung của DOM-Document Object Model – Mô hình Đối tượng Tài liệu tại một thời điểm nhất định nào đó.

Khi một ReactJS là Framework kết hợp sử dụng Virtual-DOM, mô hình này sẽ vừa thiết lập Model có thể View trong MVC. Tạo DOM, khi thay đổi View, Model cũng sẽ thay đổi theo và ngược lại. Nói một cách dễ hiểu, lập trình viên không cần tương tác trực tiếp tác động vào các phần tử DOM trong View nhưng vẫn có thể thực hiện được cở chế Data-binding. Nhờ vào cơ chế này, tốc độ ứng dụng sẽ tăng lên một cách mạnh mẽ.

* Component

ReactJS được xây dựng xoay quanh các Component, trong khi đó, các Framework khác dùng template. Để tạo ra một Component có đầy đủ những đặc tính, lập trình viên chỉ cần sử dụng phương thức createClass dùng để nhận một tham số mô tả đặc tính.

1. Tại sao lại là ReactJS ?

Đối với các doanh nghiệp, đặc biệt là những doanh nghiệp chuyên về công nghệ, việc giúp cho trải nghiệp người dùng trở nên tốt nhất sẽ mang lại lợi ích kinh tết vô cũng lớn. ReactJS chính là công nghệ có thể giúp doanh nghiệp cũng như lập trình viên một cách có hiệu quả nhất.

* **Giải quyết được vấn đề của tầng View:**

Nó giải quyết vấn đề của tầng View trong mô hình MVC (Model-View-Controller).

Giúp viết mã Javascript dễ dàng hơn với JSX

Nó sử dụng JSX (JavaScript Syntax eXtension) (Phần bổ xung cú pháp Javascript). Là một sự hòa trộn giữa Javascript và XML, vì vậy nó cũng dễ dàng hơn khi viết mã, và thân thiện hơn với các lập trình viên.

export default **class** Helloword **extends** Component{

    render(){

        return(

            <View style={styles.container}>

                <Text style={styles.welcome}>

                    welcome to ReactJS

                </Text>

                <Text style={styles.instructions}>

                    To get started

                </Text>

                <Text style={styles.instructions}>

                    Double tap R on your keyboard to reload, {'\n'}

                </Text>

            </View>

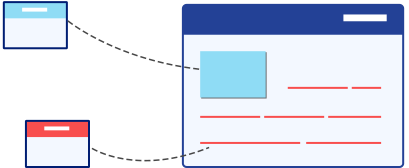
        )

    }

}

* Thành phần hóa giao diện

**React** cho phép lập trình viên tạo ra các **Component** (Thành phần) tương ứng với các phần của giao diện. Các **component** này có thể tái sử dụng, hoặc kết hợp với các **Component** khác để tạo ra một giao diện hoàn chỉnh.

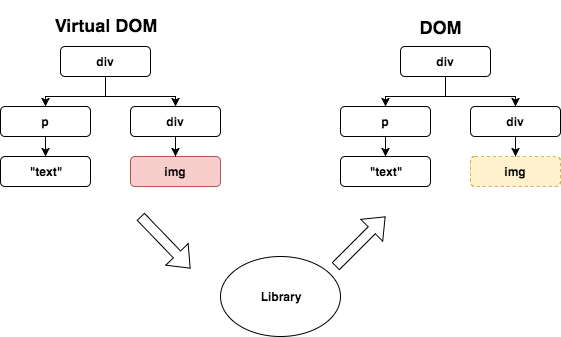


Ý tưởng về Component chính là chìa khóa giải quyết vấn đề khó khăn khi dự án ngày càng lớn. Giao diện được tạo ra từ các Component ghép lại với nhau, một Component có thể được sử dụng tại nhiều nơi trong dự án. Vì vậy thật dễ dàng khi bạn chỉ cần quản lý và sửa chữa các Component của chính bạn.

Component thực sự là tương lai của lập trình ứng dụng web, và nó cũng là tính năng quan trọng nhất mà React đem lại.

* Tăng hiệu năng với Virtual-DOM

Khi dữ liệu của Component thay đổi. React cần phải vẽ lại (rerender) giao diện. Thay vì tạo ra thay đổi trực tiếp vào mô hình DOM của trình duyệt, nó tạo ra thay đổi trên một mô hình DOM ảo (Virtual DOM). Sau đó nó tính toán sự khác biệt giữa 2 mô hình DOM, và chỉ cập nhập các khác biệt cho DOM của trình duyệt. Cách tiếp cận này mang lại hiệu năng cho ứng dụng.



Hình minh họa ở trên cho thấy mô hình ***DOM*** ảo và mô hình ***DOM*** thật có sự khác biệt duy nhất là 1 phần tử ***<img>***, ***React*** chỉ cần cập nhập phần tử này cho mô hình ***DOM*** thật.

* Thân thiện với SEO
* Một trong những vấn đề lớn của các **Javascript Framework** là nó không thân thiện với các cỗ máy tìm kiếm (**Search Engine**). Mặc dù đã có nhiều cải thiện nhưng thực sự **Search Engine** vẫn gặp rất nhiều khó khăn khi **"đọc"** các ứng dụng web chứa nhiều **Javascript**.
* **React** có thể chạy tại phía **Client**, nhưng nó cũng có thể chạy tại phía **Server**, vì vậy dữ liệu trả về cho trình duyệt là văn bản **HTML**, nó không gây ra khó khăn gì cho các **Search Engine**, vì vậy nó thân thiện với **SEO**.

1. Nhược điểm của ReactJS

* Reactjs chỉ phục vụ cho tầng View. React chỉ là View Library nó không phải là một MVC framework như những framework khác. Đây chỉ là thư viện của Facebook giúp render ra phần view. Vì thế React sẽ không có phần Model và Controller, mà phải kết hợp với các thư viện khác. React cũng sẽ không có 2-way binding hay là Ajax
* Tích hợp Reactjs vào các framework MVC truyền thống yêu cầu cần phải cấu hình lại.
* React khá nặng nếu so với các framework khác React có kích thước tương tương với Angular (Khoảng 35kb so với 39kb của Angular). Trong khi đó Angular là một framework hoàn chỉnh
* Khó tiếp cận cho người mới học Web.