

# Tìm hiểu về MERN Stack

## 1. Giới thiệu

MERN là viết tắt của MongoDB, Express, React, Node, theo tên bốn công nghệ chính tạo nên ngăn xếp.

MongoDB - cơ sở dữ liệu tài liệu

Express (.js) - Khung web Node.js

React (.js) - một khung JavaScript phía máy khách

Node (.js) - máy chủ web JavaScript hàng đầu

## 2. ReactJS (FrontEnd)

### a. Giới thiệu

Tầng trên cùng của ngăn xếp MERN là React.js, khung JavaScript khai báo để tạo các ứng dụng phía máy khách động trong HTML. React cho phép bạn xây dựng các giao diện phức tạp thông qua các Thành phần đơn giản, kết nối chúng với dữ liệu trên máy chủ phụ trợ của bạn và hiển thị chúng dưới dạng HTML.

Điểm mạnh của React là xử lý các giao diện theo hướng dữ liệu, trạng thái với mã tối thiểu và ít đau đớn nhất và nó có tất cả các chuồng và còi mà bạn mong đợi từ một khung công tác web hiện đại: hỗ trợ tuyệt vời cho các biểu mẫu, xử lý lỗi, sự kiện, danh sách và hơn thế nữa .

Những tính năng của ReactJS:

- **JSX:** viết tắt của JavaScript extension, nó là React extension , giúp cho việc thay đổi cây DOM dễ dàng hơn bằng HTML-style code đơn giản. Nó là một trong những tính năng tốt và dễ sử dụng.
- **Components:** Một trang web được xây dựng bằng ReactJS là một sự kết hợp nhiều component lại với nhau chứ không phải chung một Template như bình thường. Các component cũng như các hàm JavaScript bình thường, giúp tạo

ra các code dễ dàng bằng cách tách các logic ra thành các đoạn code độc lập có thể tái sử dụng. Chúng ta có thể sử dụng component dưới dạng function hoặc class, ngoài ra các component còn có state và props.

- Virtual DOM: ReactJS tạo một thứ gọi là Virtual DOM (DOM ảo). Đúng như tên gọi, nó là một copy của DOM thật trên trang web đó. ReactJS dùng những DOM ảo đó để tìm đúng những DOM thật cần được cập nhật khi có bất kỳ sự kiện nào làm các thành phần bên trong nó thay đổi.
- Javascript Expressions: Biểu thức JS có thể sử dụng trong file .jsx hoặc .js bằng cách sử dụng cặp dấu ngoặc nhọn “{}”.

Ưu điểm và nhược điểm của ReactJS:

- Ưu điểm:

- Vì ReactJS sử dụng DOM ảo để cache cấu trúc dữ liệu trong bộ nhớ và chỉ những thay đổi cuối cùng mới được cập nhật vào trong DOM trình duyệt. Điều này làm cho ứng dụng trở nên nhanh hơn.
- Bạn có thể tạo các component theo từng chức năng mà bạn muốn bằng cách sử dụng tính năng react component. Các component này có thể tái sử dụng theo nhu cầu của bạn, đồng thời việc tạo các component theo từng chức năng cũng giúp cho việc bảo trì sau này trở nên dễ dàng hơn.
- ReactJS là một opensource, vì vậy cũng rất dễ cho những bạn mới bắt đầu tìm hiểu nó.
- Trong những năm gần đây, ReactJS đang trở nên phổ biến hơn và được duy trì bởi Facebook và Instagram. Ngoài ra nó cũng được sử dụng bởi các công ty nổi tiếng như Apple, Netflix, ...
- Facebook vẫn đang duy trì, phát triển, và cho ra những thay đổi mới. Vì thế bạn cứ yên tâm sử dụng ReactJS cho những dự án của bạn hoặc những dự án dành do doanh nghiệp.
- ReactJS có thể được sử dụng để xây dựng giao diện người dùng cho cả các ứng dụng dành cho máy tính và các ứng dụng di động.

- Dễ dàng cho việc test và debug, vì hầu hết các code đều được thực hiện bằng JavaScript chứ không phải bằng HTML.
- Nhược điểm:
  - Vì hầu hết code được viết dưới dạng JSX, tức là HTML và CSS là một phần của JavaScript, nó không giống như những framework khác vẫn tách biệt giữa HTML và CSS nên những bạn mới làm quen với ReactJS sẽ hơi lúng túng và dễ nhầm lẫn giữa JSX và HTML. Tuy nhiên bạn sẽ nhanh chóng quen với cách kết hợp này của React mà thôi.
  - Một nhược điểm nữa của ReactJS đó là dung lượng các file của nó hơi lớn.

## **b. React hook**

Hooks là một bổ sung mới trong React 16.8.

Hooks là những hàm cho phép bạn “kết nối” React state và lifecycle vào các components sử dụng hàm.

Với Hooks bạn có thể sử dụng state và lifecycles mà không cần dùng ES6 Class.

Lợi ích của React Hook:

- Khiến các component trở nên gọn nhẹ hơn
- Giảm đáng kể số lượng code, dễ tiếp cận
- Cho phép chúng ta sử dụng state ngay trong function component

Các loại React Hook:

- useState
- useEffect
- useContext
- useReducer
- useCallback
- useMemo

- useRef ...

Rule của React Hook:

- Trong cùng một component, bạn có thể sử dụng bao nhiêu useState và useEffect tùy ý nhưng các hook này phải gọi ở trên cùng của function, không được nằm trong vòng lặp, khu vực điều kiện, hay các function con
- Nó chỉ sử dụng trong functional component
- Khi sử dụng useEffect để lấy dữ liệu, cần kiểm tra dữ liệu đã tồn tại hay chưa. Nếu không thì hàm sẽ gửi request liên tục

### **3. ExpressJS and NodeJS (Server Tier)**

Cấp độ tiếp theo là framework phía máy chủ Express.js, chạy bên trong máy chủ Node.js. Express.js tự nhận mình là một “khung web nhanh, không được tích hợp, tối giản cho Node.js” và đó thực sự là chính xác những gì nó đang có. Express.js có các mô hình mạnh mẽ để định tuyến URL (khớp URL đến với một chức năng máy chủ) và xử lý các yêu cầu và phản hồi HTTP.

Bằng cách thực hiện các Yêu cầu HTTP XML (XHR) hoặc GET hoặc POST từ giao diện người dùng React.js của bạn, bạn có thể kết nối với các hàm Express.js cung cấp năng lượng cho ứng dụng của bạn. Các chức năng đó lần lượt sử dụng trình điều khiển MongoDB’s Node.js, hoặc thông qua lệnh gọi lại để sử dụng Promises, để truy cập và cập nhật dữ liệu trong cơ sở dữ liệu MongoDB của bạn.

Những tính năng của ExpressJS:

- Phát triển máy chủ nhanh chóng: Expressjs cung cấp nhiều tính năng dưới dạng các hàm để dễ dàng sử dụng ở bất kỳ đâu trong chương trình. Điều này đã loại bỏ nhu cầu viết mã từ đó tiết kiệm được thời gian.
- Phần mềm trung gian Middleware: Đây là phần mềm trung gian có quyền truy cập vào cơ sở dữ liệu, yêu cầu của khách hàng và những phần mềm trung gian khác. Phần mềm Middleware này chịu trách nhiệm chính cho việc tổ chức có hệ thống các chức năng của Express.js.

- Định tuyến – Routing: Express js cung cấp cơ chế định tuyến giúp duy trì trạng thái của website với sự trợ giúp của URL.
- Tạo mẫu – Templating: Các công cụ tạo khuôn mẫu được Express.js cung cấp cho phép các nhà xây dựng nội dung động trên các website bằng cách tạo dựng các mẫu HTML ở phía máy chủ.
- Gỡ lỗi – Debugging: Để phát triển thành công các ứng dụng web không thể thiết đi việc gỡ lỗi. Giờ đây với Expressjs việc gỡ lỗi đã trở nên dễ dàng hơn nhờ khả năng xác định chính xác các phần ứng dụng web có lỗi.

#### **4. MongoDB (Database Tier)**

Nếu ứng dụng của bạn lưu trữ bất kỳ dữ liệu nào (hồ sơ người dùng, nội dung, nhận xét, tải lên, sự kiện, v.v.), thì bạn sẽ muốn có một cơ sở dữ liệu để làm việc như React, Express và Node.

Đó là nơi MongoDB xuất hiện: Các tài liệu JSON được tạo trong giao diện người dùng React.js của bạn có thể được gửi đến máy chủ Express.js, nơi chúng có thể được xử lý và (giả sử chúng hợp lệ) được lưu trữ trực tiếp trong MongoDB để truy xuất sau này.

Các đặc trưng của MongoDB:

- Các ad hoc query: hỗ trợ search bằng field, các phép search thông thường, regular expression searches, và range queries.
- Indexing: bất kì field nào trong BSON document cũng có thể được index.
- Replication: có ý nghĩa là “nhân bản”, là có một phiên bản giống hệt phiên bản đang tồn tại, đang sử dụng. Với cơ sở dữ liệu, nhu cầu lưu trữ lớn, đòi hỏi cơ sở dữ liệu toàn vẹn, không bị mất mát trước những sự cố ngoài dự đoán là rất cao. Vì vậy, người ta nghĩ ra khái niệm “nhân bản”, tạo một phiên bản cơ sở dữ liệu giống hệt cơ sở dữ liệu đang tồn tại, và lưu trữ ở một nơi khác, đề phòng có sự cố.
- Aggregation: Các Aggregation operation xử lý các bản ghi dữ liệu và trả về kết quả đã được tính toán. Các phép toán tập hợp nhóm các giá trị từ

nhiều Document lại với nhau, và có thể thực hiện nhiều phép toán đa dạng trên dữ liệu đã được nhóm đó để trả về một kết quả duy nhất. Trong SQL, `count(*)` và `GROUP BY` là tương đương với Aggregation trong MongoDB.

- Lưu trữ file: MongoDB được dùng như một hệ thống file tận dụng những function trên và hoạt động như một cách phân phối qua sharding.

## 5. Tài liệu tham khảo

- MongoDB, <https://www.mongodb.com/mern-stack>
- ReactJS, <https://reactjs.org/>
- Làm quen với React Hook, <https://viblo.asia/p/lam-quen-voi-react-hook-OeVKBdnQlkW>
- Tìm hiểu về ExpressJS, <https://viblo.asia/p/phan-1-tim-hieu-express-js-framework-Qbq5Qq7m5D8>