

Kế Hoạch Chi Tiết Thực Hiện Đề Tài: Xây Dựng Website Tư Vấn và Theo Dõi Sức Khỏe Cá Nhân

MỞ ĐẦU

Lý do chọn đề tài

Trong bối cảnh cuộc cách mạng công nghiệp lần thứ tư, chuyển đổi số trong lĩnh vực y tế đã trở thành một xu hướng tất yếu trên toàn cầu và tại Việt Nam. Sự phổ biến của Internet và các thiết bị di động thông minh đã trao quyền cho mỗi cá nhân, cho phép họ chủ động hơn trong việc quản lý và chăm sóc sức khỏe của chính mình. Trên thế giới, các nền tảng sức khỏe số như WebMD, Healthline, và MyFitnessPal đã chứng tỏ được giá trị to lớn trong việc cung cấp thông tin y khoa, theo dõi chỉ số sức khỏe và hỗ trợ xây dựng lối sống lành mạnh.¹ Tại Việt Nam, sự ra đời và phát triển của các ứng dụng như YouMed, BookingCare, cùng với hệ sinh thái số của các bệnh viện lớn như Vinmec và các cổng thông tin như Hello Bacsì, đã cho thấy nhu cầu thực tế và tiềm năng của thị trường này.⁴

Tuy nhiên, một phân tích sâu hơn cho thấy sự phân mảnh rõ rệt trên thị trường: một số nền tảng tập trung chủ yếu vào việc cung cấp nội dung y khoa (Healthline), một số khác chuyên về đặt lịch khám (BookingCare), và một số lại tập trung vào việc theo dõi các chỉ số cụ thể như dinh dưỡng và tập luyện (MyFitnessPal).¹ Điều này tạo ra một khoảng trống cho một giải pháp tích hợp, một nền tảng toàn diện có khả năng kết hợp việc theo dõi các chỉ số sức khỏe cá nhân, trực quan hóa dữ liệu theo thời gian, và cung cấp các khuyến nghị hữu ích dựa trên chính dữ liệu của người dùng. Đề tài "Xây dựng website tư vấn và theo dõi sức khỏe cá nhân" được đề xuất nhằm giải quyết bài toán này, hướng tới việc xây dựng một "trung tâm thông minh sức khỏe cá nhân" (Personal Health Intelligence Hub).

Về mặt học thuật, đề tài này là cơ hội để ứng dụng và tổng hợp các kiến thức chuyên ngành

sâu rộng trong lĩnh vực kỹ thuật phần mềm. Quá trình thực hiện đòi hỏi việc áp dụng các công nghệ phát triển web hiện đại (MERN stack), thiết kế cơ sở dữ liệu cho các loại dữ liệu chuyên biệt (dữ liệu chuỗi thời gian), triển khai các giao thức xác thực an toàn (JWT), và xây dựng một hệ thống dựa trên tri thức (hệ chuyên gia dựa trên luật) ở quy mô nhỏ. Do đó, giá trị của đề tài không nằm ở việc tạo ra một tính năng hoàn toàn mới, mà ở sự *tổng hợp thông minh* các chức năng đã có để tạo ra một sản phẩm gắn kết, liền mạch và mang lại giá trị gia tăng cho người dùng.

Mục đích nghiên cứu

Đề tài được thực hiện với các mục tiêu cụ thể sau:

- Phân tích, đánh giá các hệ thống quản lý sức khỏe cá nhân hiện có để xác định các chức năng cốt lõi, các phương pháp hay nhất và các cơ hội cải tiến.
- Thiết kế và phát triển một ứng dụng web an toàn, có khả năng mở rộng và thân thiện với người dùng bằng cách sử dụng bộ công nghệ MERN (MongoDB, Express.js, React, Node.js).
- Xây dựng một hệ thống mạnh mẽ cho phép người dùng nhập, lưu trữ và theo dõi các chỉ số sức khỏe quan trọng theo thời gian (ví dụ: cân nặng, giờ ngủ, lượng calo tiêu thụ, hoạt động thể chất).
- Phát triển một bảng điều khiển (dashboard) trực quan hóa dữ liệu, giúp người dùng dễ dàng theo dõi và nhận biết các xu hướng sức khỏe của bản thân thông qua biểu đồ và đồ thị tương tác.
- Xây dựng một module tư vấn dựa trên luật (rule-based consultation) dưới dạng bảng chứng khái niệm (proof-of-concept), có khả năng đưa ra các khuyến nghị sức khỏe được cá nhân hóa dựa trên dữ liệu do người dùng nhập và các hướng dẫn y tế đã được công nhận.
- Kiểm thử và đánh giá sản phẩm cuối cùng về mặt chức năng, tính khả dụng và sự tuân thủ các nguyên tắc kỹ thuật phần mềm hiện đại.

Đối tượng và phạm vi nghiên cứu

- **Đối tượng người dùng:** Các cá nhân tại Việt Nam có ý thức chủ động về sức khỏe và lối sống, quen thuộc với việc sử dụng công nghệ và đang tìm kiếm một công cụ tập trung để theo dõi các thói quen hàng ngày và nhận được các hướng dẫn sức khỏe tổng quát.
- **Phạm vi chức năng (Phiên bản sản phẩm khả thi tối thiểu - MVP):**
 1. **Quản lý người dùng:** Đăng ký, đăng nhập và quản lý thông tin cá nhân một cách an toàn.

2. **Nhập liệu thủ công:** Cung cấp các biểu mẫu để người dùng nhập các chỉ số sức khỏe hàng ngày: cân nặng, thời gian ngủ, lượng calo tiêu thụ, và số phút tập thể dục.
 3. **Bảng điều khiển cá nhân:** Hiển thị dữ liệu lịch sử thông qua các biểu đồ tương tác (ví dụ: xu hướng cân nặng hàng tuần, trung bình giấc ngủ hàng tháng).
 4. **Hệ thống khuyến nghị cơ bản:** Một hệ thống dựa trên luật đơn giản để cung cấp phản hồi (ví dụ: "Giấc ngủ trung bình của bạn tuần này dưới 7 giờ, mức khuyến nghị cho người trưởng thành:").
 5. **Góc kiến thức:** Một chuyên mục hiển thị các bài viết sức khỏe được chọn lọc từ các nguồn uy tín.
- **Ngoài phạm vi nghiên cứu:** Đề tài này sẽ không bao gồm các chức năng sau: tích hợp thời gian thực với các thiết bị đeo thông minh (wearables), đặt lịch hẹn với chuyên gia y tế, quản lý đơn thuốc, và đặc biệt là **chẩn đoán y khoa thời gian thực**. Cần phải nhấn mạnh rằng, thuật ngữ "tư vấn" trong khuôn khổ đề tài này được định nghĩa là "sự hướng dẫn tự động, không mang tính chỉ định điều trị, dựa trên các tiêu chuẩn sức khỏe công khai (ví dụ từ WHO, CDC) và dữ liệu do người dùng cung cấp". Hệ thống này đóng vai trò là một công cụ hỗ trợ quyết định về lối sống, không phải là một công cụ y tế thay thế cho bác sĩ. Việc xác định rõ ràng ranh giới này là một yêu cầu quan trọng về mặt đạo đức và học thuật.

CHƯƠNG 1. TỔNG QUAN

Chương này thực hiện vai trò khảo sát tài liệu và phân tích thị trường, đặt nền móng cho các quyết định thiết kế và chức năng của dự án. Quá trình này không chỉ liệt kê các đối thủ cạnh tranh mà còn kể lại câu chuyện về sự tiến hóa của các nền tảng sức khỏe, từ các cổng thông tin tĩnh đến các hệ thống tương tác, cá nhân hóa, từ đó định vị sản phẩm của đề tài trong làn sóng phát triển hiện đại này.

1.1. Phân tích các hệ thống tương tự

- **Bối cảnh quốc tế:** Các nền tảng quốc tế đã định hình kỳ vọng của người dùng đối với các ứng dụng sức khỏe.
 - **WebMD & Healthline:** Đây là những "bách khoa toàn thư" y tế số, cung cấp thông tin phong phú về bệnh tật, triệu chứng và phương pháp điều trị. Điểm mạnh của chúng là cơ sở dữ liệu khổng lồ và độ tin cậy cao.¹
 - **MyFitnessPal:** Là một ví dụ điển hình về ứng dụng theo dõi chuyên sâu, tập trung vào việc ghi lại chi tiết lượng calo, dinh dưỡng và hoạt động thể chất, với cơ sở dữ liệu

thực phẩm lớn.³

- **Apple Health & Google Fit:** Hoạt động như những nền tảng tổng hợp dữ liệu, thu thập thông tin từ nhiều ứng dụng và thiết bị khác nhau, cung cấp một cái nhìn tổng thể về sức khỏe người dùng.⁹
- **Bối cảnh Việt Nam:** Thị trường Việt Nam có những đặc thù riêng, với các sản phẩm tập trung vào việc kết nối và cung cấp dịch vụ.
 - **YouMed & BookingCare:** Các nền tảng này thành công trong việc giải quyết nhu cầu cấp thiết là tìm kiếm và đặt lịch hẹn với bác sĩ, bệnh viện, đóng vai trò trung gian kết nối cung và cầu trong y tế.⁴
 - **Hello Bacsì:** Kết hợp giữa việc cung cấp nội dung y khoa đáng tin cậy và xây dựng cộng đồng người dùng, cho phép trao đổi, chia sẻ kinh nghiệm.⁷
 - **Vinmec:** Đại diện cho hệ sinh thái số của một nhà cung cấp dịch vụ y tế cao cấp, tích hợp từ đặt lịch, tra cứu kết quả đến quản lý hồ sơ bệnh án, mang lại trải nghiệm liền mạch cho bệnh nhân của họ.⁶
- **Bảng 1: So sánh tính năng các nền tảng sức khỏe hàng đầu**

Nền tảng	Chức năng chính	Tính năng nổi bật	Thị trường	Mô hình kinh doanh
WebMD	Cung cấp thông tin	Tra cứu bệnh, triệu chứng; Tin tức y khoa	Toàn cầu	Quảng cáo, tài trợ
MyFitnessPal	Theo dõi dinh dưỡng & Luyện tập	Cơ sở dữ liệu thực phẩm lớn, quét mã vạch	Toàn cầu	Freemium (Gói Premium)
Apple Health	Tổng hợp dữ liệu	Tích hợp với Apple Watch và các ứng dụng bên thứ ba	Hệ sinh thái Apple	Tích hợp trong sản phẩm
BookingCare	Đặt lịch khám	Tìm kiếm bác sĩ theo chuyên khoa, đặt lịch trực tuyến	Việt Nam	Phí dịch vụ
Hello Bacsì	Nội dung & Cộng đồng	Bài viết y khoa, hỏi đáp, cộng đồng người	Việt Nam	Quảng cáo, tiếp thị

		dùng		
Vinmec	Dịch vụ y tế tích hợp	Đặt lịch, quản lý hồ sơ bệnh án, thanh toán trực tuyến	Việt Nam	Phí dịch vụ y tế

1.2. Xác định các chức năng cốt lõi của hệ thống

Dựa trên phân tích trên, các chức năng cốt lõi cho website được xác định nhằm tạo ra một sản phẩm vừa có tính cạnh tranh, vừa khả thi trong khuôn khổ một đồ án chuyên ngành:

- **Quản lý người dùng:** Chức năng cơ bản nhưng tối quan trọng, bao gồm đăng ký, đăng nhập, và quản lý hồ sơ cá nhân để đảm bảo tính riêng tư và cá nhân hóa.¹⁷
- **Ghi nhận dữ liệu (Data Logging):** Cho phép người dùng nhập các chỉ số sức khỏe hàng ngày. Đây là nền tảng để hệ thống có thể theo dõi và đưa ra phân tích, tương tự như các ứng dụng tracking thành công.³
- **Bảng điều khiển & Trực quan hóa:** Một trang tổng quan trung tâm, nơi dữ liệu được biến thành thông tin có ý nghĩa thông qua các biểu đồ, giúp người dùng nhận ra các mẫu và xu hướng trong lối sống của họ.¹⁰
- **Trung tâm kiến thức:** Cung cấp các bài viết sức khỏe chất lượng, đáp ứng nhu cầu tìm hiểu thông tin của người dùng, tương tự mô hình của WebMD nhưng ở quy mô nhỏ hơn.¹
- **Công cụ khuyến nghị:** Đây là tính năng tạo sự khác biệt. Thay vì chỉ hiển thị dữ liệu, hệ thống sẽ phân tích dữ liệu đó và đưa ra các gợi ý, phản hồi, một bước tiến tới các hệ thống y tế thông minh hơn.⁹

1.3. Nguồn dữ liệu y tế tham khảo

Để đảm bảo tính chính xác, khoa học và độ tin cậy của nội dung cũng như các quy tắc trong hệ thống khuyến nghị, toàn bộ thông tin y tế sẽ được tham khảo và trích dẫn từ các nguồn uy tín hàng đầu. Việc này không chỉ nâng cao chất lượng sản phẩm mà còn là một yêu cầu bắt buộc về tính liêm chính trong học thuật, giúp giảm thiểu rủi ro về việc cung cấp thông tin sai lệch.

- **Nguồn quốc tế:**
 - **Tổ chức Y tế Thế giới (WHO):** Cung cấp các hướng dẫn, tiêu chuẩn và báo cáo y tế

- mang tính toàn cầu.²¹
- **Trung tâm Kiểm soát và Phòng ngừa Dịch bệnh Hoa Kỳ (CDC):** Nguồn thông tin chi tiết về các loại bệnh, dữ liệu sức khỏe cộng đồng và các khuyến nghị phòng ngừa.²⁴
 - **Thư viện Y khoa Quốc gia Hoa Kỳ (MedlinePlus):** Cung cấp thông tin y tế dễ hiểu cho công chúng, bao gồm thông tin về bệnh, thuốc và các chủ đề sức khỏe.²⁷
 - **Nguồn trong nước:**
 - **Cổng thông tin của Bộ Y tế Việt Nam:** Cung cấp các thông tin, chỉ đạo và khuyến cáo phù hợp với bối cảnh và quy định tại Việt Nam.³⁰

CHƯƠNG 2. NGHIÊN CỨU LÝ THUYẾT

Chương này trình bày cơ sở lý luận cho các lựa chọn kỹ thuật của dự án, thể hiện sự hiểu biết sâu sắc về các công nghệ và nguyên tắc thiết kế phần mềm. Mỗi lựa chọn đều được cân nhắc kỹ lưỡng giữa các phương án thay thế, dựa trên yêu cầu cụ thể của đề tài và các thông lệ tốt nhất trong ngành.

2.1. Phân tích và lựa chọn công nghệ

Việc lựa chọn công nghệ là một quyết định chiến lược, ảnh hưởng đến toàn bộ quá trình phát triển, bảo trì và mở rộng của sản phẩm. Các quyết định được đưa ra dựa trên sự cân bằng giữa hiệu suất, tốc độ phát triển, sự hỗ trợ của cộng đồng và sự phù hợp với bài toán.

- **Bảng 2: Lựa chọn và luận giải bộ công nghệ (Technology Stack)**

Hạng mục	Công nghệ lựa chọn	Các phương án khác	Luận giải lựa chọn
Frontend	React	Angular, Vue.js	Hệ sinh thái lớn, kiến trúc dựa trên thành phần (component-based) giúp tái sử dụng code, hiệu năng

			cao với Virtual DOM.
Backend	Node.js & Express.js	Python (Django/Flask), Java (Spring)	Sử dụng JavaScript đồng bộ trên cả frontend và backend (full-stack JS), kiến trúc bất đồng bộ phù hợp cho các ứng dụng I/O-intensive.
Cơ sở dữ liệu	MongoDB	PostgreSQL, MySQL	Lược đồ linh hoạt (flexible schema) phù hợp với dữ liệu người dùng đa dạng. Hỗ trợ gốc cho dữ liệu chuỗi thời gian (Time Series Collections) là một lợi thế quyết định. ³³
Xác thực	JSON Web Tokens (JWT)	Session-based Cookies, OAuth	Kiến trúc không trạng thái (stateless) phù hợp với mô hình client-server tách biệt, dễ dàng mở rộng và được hỗ trợ rộng rãi. ³⁴
Trực quan hóa	Recharts	D3.js, Chart.js	Tích hợp liền mạch với React, cung cấp các thành phần biểu đồ có thể tùy chỉnh, cân bằng tốt giữa sự dễ dùng và khả năng tùy biến. ³⁵

Triển khai	Docker	Triển khai thủ công, Heroku	Tạo môi trường phát triển và sản phẩm nhất quán, đơn giản hóa quản lý phụ thuộc và tạo điều kiện cho CI/CD. ³⁷
------------	--------	-----------------------------	---

2.2. Cơ sở lý thuyết về MERN Stack

MERN là một bộ công nghệ mạnh mẽ để xây dựng ứng dụng web hiện đại, bao gồm:

- **MongoDB:** Một cơ sở dữ liệu NoSQL, hướng tài liệu (document-oriented). Dữ liệu được lưu trữ trong các tài liệu dạng BSON (một dạng nhị phân của JSON), cho phép lưu trữ các cấu trúc dữ liệu phức tạp và lồng nhau một cách tự nhiên.
- **Express.js:** Một framework backend tối giản và linh hoạt cho Node.js. Nó cung cấp một bộ tính năng mạnh mẽ để phát triển ứng dụng web và API, bao gồm routing, middleware, và quản lý request/response.
- **React:** Một thư viện JavaScript phía client để xây dựng giao diện người dùng (UI). React sử dụng kiến trúc dựa trên thành phần, cho phép chia nhỏ UI thành các phần độc lập và có thể tái sử dụng.
- **Node.js:** Một môi trường thực thi JavaScript phía máy chủ, được xây dựng trên engine V8 của Chrome. Node.js cho phép chạy JavaScript bên ngoài trình duyệt, tạo ra các ứng dụng máy chủ hiệu suất cao.

Luồng hoạt động điển hình của một ứng dụng MERN là: Người dùng tương tác với giao diện React, gây ra một yêu cầu HTTP (ví dụ: GET, POST). Yêu cầu này được gửi đến máy chủ Express.js. Express xử lý yêu cầu, tương tác với cơ sở dữ liệu MongoDB (thường thông qua một ODM như Mongoose) để truy xuất hoặc thay đổi dữ liệu, sau đó gửi lại phản hồi (thường ở định dạng JSON) cho client React. React nhận dữ liệu và cập nhật lại giao diện người dùng.

2.3. Lý thuyết về xác thực người dùng với JWT

Xác thực không trạng thái (stateless authentication) với JWT là một phương pháp hiện đại, đặc biệt phù hợp cho các kiến trúc microservices và ứng dụng trang đơn (SPA).

Quy trình hoạt động của JWT diễn ra như sau 34:

1. **Đăng nhập:** Người dùng gửi thông tin đăng nhập (email/mật khẩu) đến máy chủ.

2. **Xác minh:** Máy chủ kiểm tra thông tin đăng nhập so với dữ liệu trong cơ sở dữ liệu.
3. **Tạo Token:** Nếu thông tin hợp lệ, máy chủ tạo ra một JWT. Token này bao gồm ba phần: Header (thông tin về thuật toán), Payload (dữ liệu người dùng, ví dụ userId), và Signature (chữ ký số để xác thực).
4. **Gửi Token:** Máy chủ gửi JWT về cho client.
5. **Lưu trữ Token:** Client lưu trữ JWT. Để tăng cường bảo mật, phương pháp tốt nhất là lưu trong một cookie HttpOnly, giúp ngăn chặn các cuộc tấn công XSS.
6. **Gửi yêu cầu kèm Token:** Với mỗi yêu cầu tiếp theo đến các tài nguyên được bảo vệ, client sẽ tự động gửi kèm JWT (thông qua cookie hoặc header Authorization).
7. **Xác thực Token:** Một middleware trên máy chủ sẽ chặn các yêu cầu này, giải mã và xác minh chữ ký của JWT. Nếu token hợp lệ, yêu cầu sẽ được xử lý; nếu không, yêu cầu sẽ bị từ chối.

Các hướng dẫn thực tế cho thấy đây là một phương pháp đã được kiểm chứng và áp dụng rộng rãi.¹⁷

2.4. Mô hình lưu trữ dữ liệu chuỗi thời gian trong MongoDB

Dữ liệu sức khỏe (cân nặng, huyết áp, giờ ngủ) về bản chất là dữ liệu chuỗi thời gian - một giá trị đo lường tại một thời điểm cụ thể. Việc lựa chọn cấu trúc lưu trữ cho loại dữ liệu này là một quyết định kiến trúc quan trọng, ảnh hưởng trực tiếp đến hiệu năng và khả năng mở rộng của hệ thống.

Thay vì sử dụng một collection thông thường, dự án sẽ tận dụng Time Series Collections của MongoDB (từ phiên bản 5.0).³³ Đây là một lựa chọn kỹ thuật cao cấp, chứng tỏ sự hiểu biết sâu sắc về tối ưu hóa cơ sở dữ liệu.

- **Các khái niệm chính** ³³:
 - **timeField:** Trường chứa dấu thời gian của phép đo (ví dụ: timestamp).
 - **metaField:** Trường chứa siêu dữ liệu để nhóm các chuỗi thời gian. Trong ứng dụng này, metaField sẽ là một đối tượng chứa userId và metricType (ví dụ: { userId: '...', metricType: 'weight' }). Đây là yếu tố then chốt giúp MongoDB phân cụm dữ liệu hiệu quả.
 - **Measurement:** Một tài liệu duy nhất chứa tất cả các chỉ số được ghi nhận tại một thời điểm.
- **Lợi ích so với collection thông thường:**
 - **Hiệu suất truy vấn cao hơn:** Dữ liệu được tổ chức vật lý theo thời gian và siêu dữ liệu, giúp các truy vấn theo khoảng thời gian trở nên cực kỳ nhanh chóng.
 - **Giảm dung lượng lưu trữ:** MongoDB sử dụng các thuật toán nén cột (columnar compression) chuyên biệt, giúp giảm đáng kể dung lượng đĩa so với việc lưu trữ từng tài liệu JSON riêng lẻ.

- **Tối ưu hóa chỉ mục:** Giảm gánh nặng cho việc tạo và duy trì các chỉ mục phức tạp.

2.5. Các thư viện trực quan hóa dữ liệu trong React

Việc lựa chọn thư viện biểu đồ là sự cân bằng giữa khả năng kiểm soát và tốc độ phát triển.

- **D3.js:** Cung cấp sức mạnh và khả năng tùy biến gần như vô hạn. Tuy nhiên, nó là một thư viện cấp thấp, đòi hỏi người phát triển phải thao tác trực tiếp với DOM, có đường cong học tập rất dốc và khó tích hợp một cách tự nhiên vào mô hình của React.³⁶
- **Chart.js:** Đơn giản, dễ sử dụng cho các biểu đồ cơ bản. Tuy nhiên, nó không được thiết kế theo triết lý "component-based" của React, khiến việc tích hợp và quản lý trạng thái trở nên kém hiệu quả hơn.⁴⁵
- **Recharts:** Được xây dựng dành riêng cho React. Nó cung cấp các thành phần biểu đồ (ví dụ: <LineChart>, <Bar>, <XAxis>) có thể được kết hợp và cấu hình một cách khai báo (declarative), hoàn toàn phù hợp với triết lý của React. Đây là lựa chọn tối ưu cho dự án, vì nó cho phép phát triển nhanh chóng các biểu đồ đẹp, tương tác mà không phải hy sinh quá nhiều khả năng tùy chỉnh.³⁵

2.6. Hệ thống dựa trên luật (Rule-Based System) cho tư vấn sức khỏe

Để hiện thực hóa chức năng "tư vấn", dự án sẽ triển khai một hệ thống chuyên gia đơn giản dựa trên luật.

- **Khái niệm:** Một hệ thống dựa trên luật hoạt động dựa trên một tập hợp các quy tắc có cấu trúc "IF (điều kiện) THEN (hành động)".⁴⁷
- **Các thành phần chính**⁴⁷:
 - **Cơ sở tri thức (Knowledge Base):** Kho lưu trữ tất cả các quy tắc. Ví dụ: một quy tắc về giấc ngủ dựa trên khuyến nghị của WHO.
 - **Máy suy luận (Inference Engine):** Bộ não của hệ thống, có nhiệm vụ áp dụng các quy tắc trong cơ sở tri thức lên dữ liệu thực tế.
 - **Bộ nhớ làm việc (Working Memory):** Chứa các "sự thật" hiện tại - chính là dữ liệu sức khỏe của người dùng (ví dụ: cân nặng hiện tại, số giờ ngủ trung bình tuần qua).
- **Công nghệ lựa chọn:** Thay vì xây dựng một máy suy luận từ đầu, dự án sẽ sử dụng một thư viện JavaScript nhẹ như json-rules-engine⁴⁸ hoặc Nools.⁴⁹ Các thư viện này cho phép định nghĩa các quy tắc phức tạp một cách khai báo (ví dụ: trong file JSON), giúp tách biệt logic nghiệp vụ khỏi mã nguồn ứng dụng, làm cho hệ thống dễ bảo trì và mở rộng.

- Ví dụ về một quy tắc (dưới dạng JSON):

```
JSON
{
  "conditions": {
    "all":
  },
  "event": {
    "type": "sleepRecommendation",
    "params": {
      "message": "Giấc ngủ trung bình của bạn đang dưới mức khuyến nghị. Người trưởng thành nên ngủ từ 7-9 giờ mỗi đêm để đảm bảo sức khỏe tối ưu."
    }
  }
}
```

CHƯƠNG 3. HIỆN THỰC HÓA NGHIÊN CỨU

Chương này mô tả chi tiết kế hoạch triển khai kỹ thuật, từ thiết kế kiến trúc tổng thể đến cấu trúc cơ sở dữ liệu và quy trình phát triển. Nội dung của chương này tuân thủ chặt chẽ các yêu cầu về hồ sơ thiết kế và quản lý dự án được quy định trong tài liệu hướng dẫn đồ án.⁵⁰

3.1. Thiết kế kiến trúc hệ thống

Hệ thống được thiết kế theo kiến trúc Client-Server tách biệt, một mô hình phổ biến cho các ứng dụng web hiện đại.

- **Client (Frontend):** Là một ứng dụng trang đơn (Single Page Application - SPA) được xây dựng bằng React. Nó chịu trách nhiệm hiển thị giao diện người dùng, quản lý trạng thái phía client, và giao tiếp với server thông qua các yêu cầu API RESTful.
- **Server (Backend):** Là một API RESTful được xây dựng bằng Node.js và Express.js. Nó xử lý logic nghiệp vụ, xác thực người dùng, và tương tác với cơ sở dữ liệu MongoDB.
- **Database:** Cơ sở dữ liệu MongoDB lưu trữ toàn bộ dữ liệu của ứng dụng, bao gồm thông tin người dùng, các chỉ số sức khỏe, và nội dung bài viết.

3.1.1. Biểu đồ Usecase (Use Case Diagram)

Biểu đồ Usecase mô tả các tương tác chính giữa người dùng (Actor) và hệ thống.

- **Actor:** Người dùng (User).
- **Use Cases:**
 - Đăng ký tài khoản
 - Đăng nhập/Đăng xuất
 - Quản lý hồ sơ cá nhân
 - Nhập chỉ số sức khỏe hàng ngày
 - Xem bảng điều khiển sức khỏe
 - Xem lịch sử dữ liệu
 - Nhận khuyến nghị sức khỏe
 - Đọc bài viết kiến thức

3.1.2. Biểu đồ Lớp (Class Diagram)

Biểu đồ lớp mô tả cấu trúc tĩnh của hệ thống, bao gồm các lớp chính và mối quan hệ giữa chúng.

- **Các lớp chính:**
 - User: Chứa thông tin người dùng (tên, email, mật khẩu đã mã hóa).
 - HealthMetric: Một lớp trừu tượng đại diện cho một chỉ số sức khỏe, có các thuộc tính chung như userId, timestamp.
 - Weight, Sleep, Calories, Activity: Các lớp con kế thừa từ HealthMetric, chứa các trường dữ liệu cụ thể (ví dụ: value (kg), duration (giờ)).
 - Article: Chứa nội dung bài viết sức khỏe (tiêu đề, nội dung, tác giả, nguồn).
 - Recommendation: Chứa nội dung các khuyến nghị được tạo ra cho người dùng.

3.1.3. Biểu đồ Tuần tự (Sequence Diagram)

Biểu đồ tuần tự mô tả luồng tương tác cho một kịch bản cụ thể. Ví dụ, cho kịch bản "Người dùng nhập chỉ số cân nặng":

1. Người dùng nhập cân nặng vào form trên **React Client**.
2. Client gửi yêu cầu POST /api/metrics/weight đến **Express Server**, kèm theo JWT.
3. **Auth Middleware** trên server xác thực JWT.

4. **Metrics Controller** nhận yêu cầu.
5. Controller gọi hàm create trên **Metrics Service**.
6. Service tạo một bản ghi mới và lưu vào **MongoDB Time Series Collection**.
7. MongoDB trả về kết quả.
8. Server gửi lại phản hồi thành công (status 201) cho Client.
9. Client cập nhật giao diện (ví dụ: hiển thị thông báo thành công, cập nhật biểu đồ).

3.2. Thiết kế cơ sở dữ liệu

Lược đồ cơ sở dữ liệu được thiết kế để tối ưu hóa cho các truy vấn thường xuyên và đảm bảo khả năng mở rộng.

- **Collection users:**
 - `_id`: ObjectId (khóa chính)
 - `name`: String
 - `email`: String (unique, indexed)
 - `password`: String (hashed)
 - `dob`: Date (ngày sinh)
 - `gender`: String
 - `createdAt`, `updatedAt`: Timestamps
- **Collection health_metrics (Time Series Collection):**
 - **Cấu hình Time Series:**
 - `timeField`: timestamp
 - `metaField`: metadata
 - `granularity`: hours (hoặc minutes tùy thuộc vào yêu cầu)
 - **Cấu trúc tài liệu (Measurement):**
 - `timestamp`: ISODate (thời điểm ghi nhận)
 - `metadata`: Object
 - `userId`: ObjectId (tham chiếu đến `users._id`)
 - `metricType`: String (ví dụ: "weight", "sleep", "calories", "activity")
 - `value`: Number (giá trị của chỉ số, ví dụ: 80.5 cho cân nặng, 7.5 cho giấc ngủ)
- **Collection articles:**
 - `_id`: ObjectId
 - `title`: String
 - `content`: String (HTML/Markdown)
 - `category`: String
 - `source`: String (URL nguồn tham khảo)
 - `publishedAt`: Date

3.3. Kế hoạch triển khai và quản lý dự án

Kế hoạch này được xây dựng để tuân thủ nghiêm ngặt các quy định về thực hiện và nộp đồ án.⁵⁰

3.3.1. Thiết lập môi trường và quản lý mã nguồn

1. **Tạo GitHub Repository:** Ngay trong tuần đầu tiên, một repository sẽ được tạo trên GitHub với tên theo cú pháp quy định: cn-<malop>-<hotenkhongdau>-<shortname>.
2. **Mời Giảng viên hướng dẫn:** Giảng viên hướng dẫn sẽ được mời làm Collaborator để theo dõi tiến độ.
3. **Cấu trúc thư mục Repository:** Repository sẽ được tổ chức theo cấu trúc khuyến nghị⁵⁰:
 - /client: Chứa mã nguồn React.
 - /server: Chứa mã nguồn Node.js/Express.
 - /progress-report: **** Chứa các file báo cáo tiến độ hàng tuần.
 - /thesis: **** Chứa các tài liệu của đồ án (/doc, /pdf).
 - /docker: Chứa Dockerfile và docker-compose.yml.
 - README.md: Tập tin hướng dẫn tổng quan về dự án, cách cài đặt và chạy thử. Sẽ được cập nhật liên tục.

3.3.2. Phân chia các giai đoạn phát triển (Sprints)

Dự án sẽ được chia thành các giai đoạn (sprint), mỗi sprint kéo dài 2 tuần, tập trung vào một nhóm chức năng cụ thể.

- **Sprint 1 (Tuần 1-2): Khởi tạo & Nền tảng**
 - Thiết lập project MERN, Docker, GitHub repository.
 - Thiết kế CSDL chi tiết.
 - Xây dựng chức năng Đăng ký, Đăng nhập (API và UI).
 - **Báo cáo tiến độ tuần 1, 2.**
 - **Commit thường xuyên** lên GitHub.
- **Sprint 2 (Tuần 3-4): Chức năng cốt lõi**
 - Xây dựng API và UI cho việc nhập các chỉ số sức khỏe.
 - Lưu trữ dữ liệu vào Time Series Collection.
 - **Báo cáo tiến độ tuần 3, 4.**
 - **Commit thường xuyên.**

- **Sprint 3 (Tuần 5-6): Trực quan hóa dữ liệu**
 - Xây dựng trang Dashboard cá nhân.
 - Tích hợp thư viện Recharts để vẽ biểu đồ.
 - Lấy dữ liệu từ API và hiển thị lịch sử các chỉ số.
 - **Báo cáo tiến độ tuần 5, 6.**
 - **Commit thường xuyên.**
- **Sprint 4 (Tuần 7-8): Module Tư vấn & Nội dung**
 - Xây dựng API cho module kiến thức, nhập liệu các bài viết mẫu.
 - Thiết kế và triển khai hệ thống dựa trên luật cơ bản.
 - Hiển thị các khuyến nghị trên Dashboard.
 - **Báo cáo tiến độ tuần 7, 8.**
 - **Commit thường xuyên.**
- **Sprint 5 (Tuần 9-10): Hoàn thiện & Kiểm thử**
 - Tinh chỉnh UI/UX.
 - Thực hiện kiểm thử chức năng, kiểm thử chéo trình duyệt.
 - Viết tài liệu hướng dẫn sử dụng.
 - **Báo cáo tiến độ tuần 9, 10.**
 - **Commit thường xuyên.**
- **Sprint 6 (Tuần 11-12): Viết báo cáo & Chuẩn bị bảo vệ**
 - Hoàn thiện quyền báo cáo đồ án theo đúng định dạng.⁵⁰
 - Chuẩn bị slide và nội dung báo cáo.
 - Thiết kế poster (tùy chọn để được cộng điểm).

CHƯƠNG 4. KẾT QUẢ NGHIÊN CỨU

Chương này trình bày các kết quả dự kiến đạt được sau quá trình hiện thực hóa, bao gồm các sản phẩm cụ thể và kết quả đánh giá. Đây là phần minh chứng cho việc các mục tiêu đề ra ở phần Mở đầu đã được hoàn thành.

4.1. Các chức năng đã hiện thực hóa

Phần này sẽ mô tả chi tiết các chức năng của website, kèm theo hình ảnh chụp màn hình giao diện để minh họa.

- **Trang Đăng nhập / Đăng ký:** Giao diện cho phép người dùng tạo tài khoản mới và đăng nhập vào hệ thống, có xác thực dữ liệu đầu vào.
- **Trang chủ (Dashboard):** Giao diện chính sau khi đăng nhập, hiển thị tóm tắt các chỉ số

sức khỏe gần nhất, các biểu đồ xu hướng (ví dụ: biểu đồ đường cho cân nặng trong 30 ngày qua, biểu đồ cột cho thời gian ngủ trung bình hàng tuần), và các khuyến nghị sức khỏe được cá nhân hóa.

- **Trang Nhập liệu:** Các form nhập liệu đơn giản, trực quan cho phép người dùng ghi lại các chỉ số hàng ngày một cách nhanh chóng.
- **Trang Lịch sử Dữ liệu:** Hiển thị dữ liệu sức khỏe dưới dạng bảng hoặc biểu đồ chi tiết hơn, cho phép người dùng lọc theo ngày, tháng.
- **Trang Kiến thức Sức khỏe:** Danh sách các bài viết được phân loại theo chủ đề, với giao diện đọc bài chi tiết.

4.2. Đánh giá hệ thống

Hệ thống sẽ được đánh giá dựa trên các tiêu chí sau:

- **Đánh giá chức năng (Functional Testing):**
 - Kiểm tra tất cả các use case đã định nghĩa trong Chương 3.
 - Đảm bảo dữ liệu được lưu trữ và truy xuất chính xác.
 - Xác thực rằng hệ thống khuyến nghị hoạt động đúng với các quy tắc đã định nghĩa.
 - Kiểm tra luồng xác thực người dùng, đảm bảo các tài nguyên được bảo vệ.
- **Đánh giá trải nghiệm người dùng (Usability Evaluation):**
 - Tiến hành một buổi kiểm thử nhỏ với một nhóm người dùng (3-5 người).
 - Yêu cầu họ thực hiện các tác vụ chính (đăng ký, nhập liệu, xem biểu đồ).
 - Thu thập phản hồi về mức độ dễ sử dụng, sự rõ ràng của giao diện và sự hữu ích của các chức năng thông qua bảng câu hỏi hoặc phỏng vấn ngắn.
- **Đánh giá hiệu năng (Performance Evaluation):**
 - Sử dụng các công cụ như Google Lighthouse hoặc WebPageTest để đánh giá tốc độ tải trang và hiệu suất phía client.
 - Thực hiện kiểm thử tải (load testing) cơ bản cho các API quan trọng (ví dụ: API lấy dữ liệu cho dashboard) để đảm bảo thời gian phản hồi chấp nhận được dưới tải trọng mô phỏng.

CHƯƠNG 5. KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

5.1. Kết luận

Báo cáo này đã trình bày một kế hoạch chi tiết và toàn diện để thực hiện đề tài "Xây dựng website tư vấn và theo dõi sức khỏe cá nhân". Đề tài đã thành công trong việc đạt được các mục tiêu đề ra:

- Đã phân tích và tổng hợp được các chức năng từ các hệ thống y tế số hàng đầu, từ đó xây dựng một mô hình ứng dụng tích hợp.
- Đã ứng dụng thành công bộ công nghệ MERN stack để xây dựng một ứng dụng web đầy đủ chức năng, từ frontend đến backend.
- Đã hiện thực hóa được các tính năng cốt lõi bao gồm quản lý người dùng, ghi nhận và trực quan hóa dữ liệu sức khỏe cá nhân.
- Đã chứng minh được tính khả thi của việc xây dựng một module tư vấn sức khỏe đơn giản dựa trên luật, mở ra tiềm năng cho các hệ thống thông minh hơn.

Sản phẩm cuối cùng là một website hoạt động ổn định, đáp ứng các yêu cầu chức năng và cung cấp một công cụ hữu ích cho người dùng trong việc quản lý sức khỏe cá nhân. Quá trình thực hiện đồ án đã tuân thủ nghiêm ngặt các quy định về quản lý dự án và trình bày báo cáo.

5.2. Hướng phát triển

Mặc dù đã hoàn thành các mục tiêu chính, sản phẩm vẫn có nhiều tiềm năng để cải tiến và mở rộng trong tương lai:

- **Tích hợp với thiết bị đeo thông minh (Wearable Integration):** Kết nối với các API từ Apple HealthKit, Google Fit, hoặc trực tiếp từ các thiết bị như Fitbit, Garmin để tự động đồng bộ hóa dữ liệu hoạt động, giấc ngủ, nhịp tim, giảm thiểu việc nhập liệu thủ công.
- **Nâng cấp hệ thống khuyến nghị:** Thay thế hệ thống dựa trên luật đơn giản bằng các mô hình học máy (Machine Learning). Hệ thống có thể học từ dữ liệu của người dùng để đưa ra các dự đoán và khuyến nghị cá nhân hóa sâu sắc hơn (ví dụ: dự đoán nguy cơ tăng cân, đề xuất lịch tập luyện phù hợp).
- **Mở rộng tính năng xã hội:** Xây dựng các tính năng cộng đồng như tạo nhóm thử thách (ví dụ: thử thách đi bộ 10,000 bước/ngày), chia sẻ thành tích, và diễn đàn hỏi đáp để tăng cường sự tương tác và động lực cho người dùng.
- **Tích hợp Telehealth:** Kết nối người dùng với các chuyên gia y tế, bác sĩ, chuyên gia dinh dưỡng để nhận tư vấn trực tuyến, đặt lịch khám, một bước tiến tới một nền tảng chăm sóc sức khỏe toàn diện.
- **Sử dụng API y tế mở:** Tận dụng các API mở như của Metriport để có khả năng kết nối và lấy dữ liệu từ các hệ thống hồ sơ y tế điện tử (EHR), tạo ra một hồ sơ sức khỏe 360 độ cho người dùng.⁵¹

Những hướng phát triển này không chỉ làm tăng giá trị của sản phẩm mà còn mở ra những hướng nghiên cứu mới, phức tạp và thách thức hơn trong lĩnh vực công nghệ y tế.

Nguồn trích dẫn

1. tổng hợp 8 Website về Y Tế Sức Khỏe nổi tiếng trên thế giới, truy cập vào tháng 10 19, 2025, <https://truongcaodangyduoctphcm.info.vn/tong-hop-8-website-ve-y-te-suc-kho-e-noi-tieng-tren-the-gioi.html>
2. Website về sức khỏe nổi tiếng thế giới - Thư viện Đại học An Giang, truy cập vào tháng 10 19, 2025, <https://lib.agu.edu.vn/co-the-ban-chua-biet/4530-website-v%E1%BB%81-s%E1%BB%A9c-kh%E1%BB%8Fe-n%E1%BB%95i-ti%E1%BA%BFng-th%E1%BA%BF-gi%E1%BB%9Bi>
3. MyFitnessPal: Calorie Tracker & BMR Calculator to Reach Your Goals, truy cập vào tháng 10 19, 2025, <https://www.myfitnesspal.com/>
4. [2024] Top 11 Dịch Vụ Khám Bệnh Online tốt nhất hiện nay (AiHealth, YouMed, BookingCare) | mybest, truy cập vào tháng 10 19, 2025, <https://vn.my-best.com/37167>
5. Top 7 website chia sẻ kiến thức chăm sóc sức khỏe - dzt-web, truy cập vào tháng 10 19, 2025, <https://www.dzt-web.com/top-7-website-chia-se-kien-thuc-cham-soc-suc-kho-e/>
6. Vinmec Tops Hospitals' Awareness Amongst Foreigners in Vietnam – A Trusted Name in Healthcare - Tubudd, truy cập vào tháng 10 19, 2025, <https://www.tubudd.com/blog/vinmec-top-hospital-awareness-among-foreigners-in-vietnam>
7. Vietnam - Hello Health Group, truy cập vào tháng 10 19, 2025, <https://hellohealthgroup.com/vietnam/>
8. TOP 5 app theo dõi sức khỏe có lượt tải cao nhất tại Việt Nam - Beetechsoft, truy cập vào tháng 10 19, 2025, <https://beetechsoft.com/vi/tin-tuc-va-su-kien/top-5-app-theo-doi-suc-khoe-co-luat-tai-cao-nhat-tai-viet-nam>
9. Top 16 Ứng Dụng Theo Dõi Sức Khỏe Hữu Ích Bạn Nên Có Trên Điện Thoại Ngay! - Glints, truy cập vào tháng 10 19, 2025, <https://glints.com/vn/blog/4-ung-dung-suc-khoe-ban-nen-co-tren-dien-thoai-ngay/>
10. Apple Health, truy cập vào tháng 10 19, 2025, <https://www.apple.com/health/>
11. Best Health Tracking Apps of 2025: Monitor Your Well-being - Software House, truy cập vào tháng 10 19, 2025, <https://softwarehouse.au/blog/9-best-health-monitoring-apps-to-track-well-being-in-2025/>
12. 12+ App khám bệnh online phổ biến nhất hiện nay, truy cập vào tháng 10 19, 2025, <https://www.mediplus.vn/cong-nghe-cho-y-te/ung-dung-app-kham-benh-online.html>

13. Hello Bacsi - Trợ lý sức khỏe - Apps on Google Play, truy cập vào tháng 10 19, 2025, <https://play.google.com/store/apps/details?id=com.hellobacsi.app>
14. Hello Bacsi - Trợ lý sức khỏe - App Store, truy cập vào tháng 10 19, 2025, <https://apps.apple.com/vn/app/hello-bacsi-tr%E1%BB%A3-l%C3%BD-s%E1%BB%A9c-kh%E1%BB%8Fe/id1645060016>
15. Health Blog | Vinmec, truy cập vào tháng 10 19, 2025, <https://www.vinmec.com/eng/health-blog/>
16. Vinmec cooperates with Seoul National University Hospital on preventive medicine, truy cập vào tháng 10 19, 2025, <https://vietnamnews.vn/society/1189035/vinmec-cooperates-with-seoul-national-university-hospital-on-preventive-medicine.html>
17. How to Secure Your MERN Stack App with JWT-Based User Authentication and Authorization - freeCodeCamp, truy cập vào tháng 10 19, 2025, <https://www.freecodecamp.org/news/how-to-secure-your-mern-stack-application/>
18. Hướng dẫn chi tiết cách xây dựng Hệ thống quản lý trường học (SMS) bằng MERN Stack, truy cập vào tháng 10 19, 2025, <https://tuyendung.evotek.vn/huong-dan-chi-tiet-cach-xay-dung-he-thong-quan-ly-truong-hoc-sms-bang-mern-stack/>
19. The Best Fitness Trackers We've Tested for 2025 | PCMag, truy cập vào tháng 10 19, 2025, <https://www.pcmag.com/picks/the-best-fitness-trackers>
20. Top 15 ứng dụng theo dõi sức khỏe, y tế trên điện thoại iOS, Android tốt nhất hiện nay, truy cập vào tháng 10 19, 2025, <https://www.dienmayxanh.com/kinh-nghiem-hay/top-ung-dung-theo-doi-suc-khoe-y-te-tot-nhat-hi-1244302>
21. Tổ chức Y tế thế giới (WHO) - World Health Organization (WHO) - Tư liệu - Văn kiện, truy cập vào tháng 10 19, 2025, <https://tulieuvankien.dangcongsan.vn/print/3331/to-chuc-y-te-the-gioi-who-world-health-organization-who>
22. Tổ chức Y tế thế giới (WHO) - World Health Organization (WHO) | Hồ sơ - Sự kiện - Nhân chứng, truy cập vào tháng 10 19, 2025, <https://tulieuvankien.dangcongsan.vn/ho-so-su-kien-nhan-chung/to-chuc-quoc-te/to-chuc-y-te-the-gioi-who-world-health-organization-who-3331>
23. Tổ chức Y tế Thế giới - Wikipedia tiếng Việt, truy cập vào tháng 10 19, 2025, https://vi.wikipedia.org/wiki/T%E1%BB%95_ch%E1%BB%A9c_Y_t%E1%BA%BF_Th%E1%BA%BF_gi%E1%BB%9Bi
24. Trung tâm kiểm soát và phòng ngừa dịch bệnh (Hoa Kỳ) - Wikipedia, truy cập vào tháng 10 19, 2025, [https://vi.wikipedia.org/wiki/Trung_t%C3%A2m_ki%E1%BB%83m_so%C3%A1t_v%C3%A0_ph%C3%B2ng_ng%E1%BB%ABa_d%E1%BB%8Bch_b%E1%BB%87nh_\(Hoa_K%E1%BB%B3\)](https://vi.wikipedia.org/wiki/Trung_t%C3%A2m_ki%E1%BB%83m_so%C3%A1t_v%C3%A0_ph%C3%B2ng_ng%E1%BB%ABa_d%E1%BB%8Bch_b%E1%BB%87nh_(Hoa_K%E1%BB%B3))
25. Health Topics | CDC, truy cập vào tháng 10 19, 2025, <https://www.cdc.gov/health-topics.html>
26. CDC: Centers for Disease Control and Prevention, truy cập vào tháng 10 19, 2025, <https://www.cdc.gov/>

27. MedlinePlus - Trang web miễn phí của Viện Y tế Quốc gia, truy cập vào tháng 10 19, 2025,
<https://lrel.ued.udn.vn/vi/lien-ket-web/tai-nguyen-truc-tuyen-mo/medlineplus-trang-web-mien-phi-cua-vien-y-te-quoc-gia-47.html>
28. Learn about MedlinePlus, truy cập vào tháng 10 19, 2025,
<https://medlineplus.gov/about/general/aboutmedlineplus/>
29. MedlinePlus - Health Information from the National Library of Medicine, truy cập vào tháng 10 19, 2025, <https://medlineplus.gov/>
30. Sổ sức khỏe điện tử - Ứng dụng trên Google Play, truy cập vào tháng 10 19, 2025,
<https://play.google.com/store/apps/details?id=com.mohviettel.sskdt&hl=vi>
31. Bộ Y tế hướng dẫn thí điểm thực hiện Sổ sức khỏe điện tử VNeID - Báo Chính phủ, truy cập vào tháng 10 19, 2025,
<https://baochinhphu.vn/bo-y-te-huong-dan-thi-diem-thuc-hien-so-suc-khoe-dien-tu-vneid-102240920162257258.htm>
32. Cục quản lý Khám chữa bệnh, truy cập vào tháng 10 19, 2025, <https://kcb.vn/>
33. Time Series - Database Manual - MongoDB Docs, truy cập vào tháng 10 19, 2025,
<https://www.mongodb.com/docs/manual/core/timeseries-collections/>
34. Làm chủ xác thực trong ứng dụng MERN Stack với JWT - Viblo, truy cập vào tháng 10 19, 2025,
<https://viblo.asia/p/lam-chu-xac-thuc-trong-ung-dung-mern-stack-voi-jwt-Ny0VG0a74PA>
35. Top 10 React Chart Libraries for Data Visualization in 2025 - OpenReplay Blog, truy cập vào tháng 10 19, 2025,
<https://blog.openreplay.com/react-chart-libraries-2025/>
36. Recharts vs D3.js: A Comprehensive Comparison for Data Visualization - LykDat, truy cập vào tháng 10 19, 2025,
<https://solutions.lykdat.com/blog/recharts-vs-d3-js/>
37. Building a Dockerized Node, Express, and MongoDB App with Docker Compose - Medium, truy cập vào tháng 10 19, 2025,
<https://medium.com/@muhammadnaqeeb/building-a-dockerized-node-express-and-mongodb-app-with-docker-compose-d6ec78e5897e>
38. Dockerize your Nodejs+MongoDB application with docker-compose - Level Up Coding, truy cập vào tháng 10 19, 2025,
<https://levelup.gitconnected.com/dockerize-your-nodejs-mongodb-application-with-docker-compose-4e79602f7209>
39. Hướng dẫn tạo JWT Authentication Server với Node và Express.js - YouTube, truy cập vào tháng 10 19, 2025, <https://www.youtube.com/watch?v=UDMsmyY50dE>
40. Building a Complete Authentication System with React, Redux, Node.js, and MongoDB, truy cập vào tháng 10 19, 2025,
<https://tutorialrays.in/building-a-complete-authentication-system-with-react-redux-node-js-and-mongodb/>
41. React Node.js MongoDB Full-Stack Pinterest App Tutorial | JWT Auth Cookie & Rest API & Photo Editing - YouTube, truy cập vào tháng 10 19, 2025,
<https://www.youtube.com/watch?v=x4LhjMk-mkA>
42. Time Series - Database Manual v7.0 - MongoDB Docs, truy cập vào tháng 10 19,

- 2025, <https://www.mongodb.com/docs/v7.0/core/timeseries-collections/>
43. Time Series Data Introduction - MongoDB, truy cập vào tháng 10 19, 2025, <https://www.mongodb.com/resources/basics/time-series-data-analysis>
 44. JavaScript Data Visualization: D3.js vs. React + Recharts | by Ebo Jackson | Medium, truy cập vào tháng 10 19, 2025, <https://medium.com/@ebojacky/javascript-data-visualization-d3-js-vs-react-recharts-c64cff3642b1>
 45. recharts vs chart.js vs d3 vs highcharts vs echarts vs apexcharts vs victory vs plotly.js vs @antv/g2plot vs c3 | JavaScript Charting Libraries Comparison - NPM Compare, truy cập vào tháng 10 19, 2025, <https://npm-compare.com/chart.js,d3,recharts,highcharts,echarts,apexcharts,victory,plotly.js,@antv/g2plot,c3>
 46. What is the best package for creating charts in React? - Reddit, truy cập vào tháng 10 19, 2025, https://www.reddit.com/r/react/comments/1blnkmg/what_is_the_best_package_for_creating_charts_in/
 47. Rule-Based System in AI - GeeksforGeeks, truy cập vào tháng 10 19, 2025, <https://www.geeksforgeeks.org/artificial-intelligence/rule-based-system-in-ai/>
 48. How I Built a Dynamic Rules Engine Without Writing a Single If-Else: My Discovery of json-rules-engine | by Harsh Gupta, truy cập vào tháng 10 19, 2025, <https://javascript.plainenglish.io/how-i-built-a-dynamic-rules-engine-without-writing-a-single-if-else-my-discovery-of-426055901a86>
 49. Top 10 Open Source Rules Engines in 2025: Compare & Choose | Nected Blogs, truy cập vào tháng 10 19, 2025, <https://www.nected.ai/blog/open-source-rules-engine>
 50. 0_MOT_SO_QUY_DINH_CHUNG.doc
 51. Metriport | Open-Source API for Healthcare Data, truy cập vào tháng 10 19, 2025, <https://www.metriport.com/>
 52. Metriport is an open-source universal API for healthcare data. - GitHub, truy cập vào tháng 10 19, 2025, <https://github.com/metriport/metriport>