

TRƯỜNG ĐẠI HỌC CẦN THƠ
TRƯỜNG CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG
KHOA CÔNG NGHỆ PHẦN MỀM



LUẬN VĂN TỐT NGHIỆP ĐẠI HỌC
NGÀNH KỸ THUẬT PHẦN MỀM
ĐỀ TÀI
XÂY DỰNG WEBSITE QUẢN LÝ TRUNG TÂM ĐÀO
TẠO BẰNG LÁI XE DÙNG NEXTJS VÀ MYSQL

Sinh viên: Bùi Thị Tường Di

Mssv: B1906439

Khóa: 45

Cần Thơ, 08/2023

TRƯỜNG ĐẠI HỌC CẦN THƠ
TRƯỜNG CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG
KHOA CÔNG NGHỆ PHẦN MỀM



LUẬN VĂN TỐT NGHIỆP ĐẠI HỌC
NGÀNH KỸ THUẬT PHẦN MỀM
ĐỀ TÀI
XÂY DỰNG WEBSITE QUẢN LÝ TRUNG TÂM ĐÀO
TẠO BẰNG LÁI XE DÙNG NEXTJS VÀ MYSQL

Giảng viên hướng dẫn:

TS. Nguyễn Công Danh

Sinh viên thực hiện:

Bùi Thị Tường Di

MSSV: B1906439

Khóa 45

Cần Thơ, 08/2023

PHẦN 1: MỞ ĐẦU

1. Đặt vấn đề

Ngày nay, đời sống kinh tế, xã hội phát triển kéo theo chất lượng cuộc sống ngày càng được nâng cao và đảm bảo. Do đó, việc mua xe mô tô, ô tô và học lái xe ngày càng trở nên phổ biến. Để đáp ứng nhu cầu học lái xe, thi bằng lái xe của mọi người, các trung tâm đào tạo bằng lái xe đã xuất hiện để hỗ trợ tư vấn các dịch vụ quan trọng cần thiết.

Với việc xuất hiện ngày càng nhiều trung tâm thì việc mở rộng liên kết và nâng cao chất lượng đào tạo là rất cần thiết. Trong thời đại công nghệ số đang chiếm xu thế, việc sử dụng các giải pháp nâng cao chất lượng đào tạo lái xe trực tuyến đang trở hướng phát triển mới trong ngành giáo dục đào tạo lái xe. Với nhu cầu ngày càng tăng của người học và sự cạnh tranh khốc liệt trong thị trường đào tạo lái xe thì một trong những giải pháp nâng cao chất lượng đào tạo, sát hạch là đẩy mạnh ứng dụng công nghệ thông tin vào giảng dạy.

2. Lịch sử giải quyết vấn đề

Hiện nay đã có nhiều trung tâm đào tạo bằng lái đã ứng dụng công nghệ thông tin vào quá trình giảng dạy như mô phỏng hình ảnh, thi trực tuyến, ... Tuy nhiên muốn cạnh tranh tốt hơn trên thị trường và thu hút được nhiều học viên hơn thì các trung tâm có thể tạo ra các trang web, một khái niệm đã rất quen thuộc với mọi người khi muốn tìm hiểu thông tin. Do đó, một website với mong muốn giúp cải thiện chất lượng dịch vụ, tăng cường hiệu quả đào tạo, tăng cường sự hài lòng của học viên, tăng cường uy tín và danh tiếng của trung tâm sẽ được thực hiện.

Một số website hiện nay như:

- Thi bằng lái xe: <https://thibanglaxe.com.vn/>
- Trường dạy lái xe Đồng Tiến: <https://dongtien.edu.vn/>

3. Mục tiêu đề tài

- Mục tiêu chính của đề tài là xây dựng thành công website quản lý Trung tâm đào tạo bằng lái xe.
- Website sẽ giúp các trung tâm dễ dàng tiếp cận với nhiều học viên, tiết kiệm thời gian và chi phí của học viên cũng như của trung tâm, chủ động hơn với việc hỗ trợ học và thi trực tuyến mọi lúc mọi nơi.

4. Đối tượng và phạm vi nghiên cứu

4.1. Đối tượng nghiên cứu

- Về mặt lý thuyết:
 - Tìm hiểu quy trình phát triển ứng dụng web.

- Tìm hiểu quy trình làm việc tại trung tâm đào tạo bằng lái xe.
- Tìm hiểu, phân tích các chức năng cần có của một trang web đào tạo lái xe.
- Phân tích, thiết kế website.
- Tìm hiểu quy trình kiểm thử.
- Hoàn thiện và cài đặt website.
- Về mặt kỹ thuật:
 - Sử dụng được các ngôn ngữ và framework để phát triển ứng dụng web như HTML, CSS, JavaScript, Bootstrap, ReactJS, NextJS, Mantine, v.v.
 - Sử dụng cơ sở dữ liệu MongoDB để lưu trữ dữ liệu.
 - Hiểu được quá trình kiểm thử.

4.2. Phạm vi nghiên cứu

- Phạm vi nghiên cứu của đề tài là một trung tâm đào tạo bằng lái xe. Thực hiện một website có thể truy cập bằng ở mọi thiết bị có kết nối internet.
- Website được phát triển bởi các ngôn ngữ và framework như HTML, CSS, JavaScript, Bootstrap, ReactJS, NextJS, Mantine, v.v và được sử dụng bởi các nhóm người dùng sau:
 - Quản trị viên: quản lý thông tin website, khóa học, giáo viên, học viên
 - Giáo viên: quản lý các khóa học, lịch học, lịch thi, điểm số
 - Học viên: xem thông tin các khóa học, đăng ký khóa học, xem kết quả thi
- Thời gian thực hiện đề tài: 16 tuần.
- Tham khảo các website trực tuyến đã có như: Thibanglaxe.com.vn, dongtien.edu.vn, ...

5. Nội dung nghiên cứu

5.1. Quy trình nghiên cứu

- Lập kế hoạch, nghiên cứu tài liệu tham khảo.
- Phân tích hệ thống và thiết kế cơ sở dữ liệu phù hợp.
- Thiết kế giao diện website.
- Lập trình các chức năng của website.
- Kiểm thử website và sửa lỗi phát sinh.

5.2. Các công nghệ sử dụng

- HTML là ngôn ngữ đánh dấu siêu văn bản thường được sử dụng để tạo và cấu trúc các phần trong trang web và ứng dụng, phân chia các đoạn văn, heading, link, blockquotes, ...

- CSS là chữ viết tắt của Cascading Style Sheets, nó là một ngôn ngữ được sử dụng để tìm và định dạng lại các phần tử được tạo ra bởi các ngôn ngữ đánh dấu (HTML). Nói ngắn gọn hơn là ngôn ngữ tạo phong cách cho trang web.
- JavaScript là ngôn ngữ lập trình được nhà phát triển sử dụng để tạo trang web tương tác. Từ làm mới bảng tin trên trang mạng xã hội đến hiển thị hình ảnh động và bản đồ tương tác, các chức năng của JavaScript có thể cải thiện trải nghiệm người dùng của trang web.
- ReactJS là một thư viện JavaScript được sử dụng để xây dựng các giao diện người dùng (UI) tương tác trong các ứng dụng web. Nghiên cứu cần tập trung vào các khía cạnh lý thuyết của ReactJS, bao gồm cách hoạt động của React, các hàm lifecycle của component, cách tạo và quản lý state của component, cách xây dựng component có tính tái sử dụng và các best practice khi phát triển ứng dụng web sử dụng React.
- Next.js là một framework front-end React được phát triển dưới dạng open-source bổ sung các khả năng tối ưu hóa như render phía máy chủ (SSR) và tạo trang web static.
- MongoDB là một database hướng tài liệu (document), một dạng NoSQL database. Vì thế, MongoDB sẽ tránh cấu trúc table-based của relational database để thích ứng với các tài liệu như JSON có một schema rất linh hoạt gọi là BSON. MongoDB sử dụng lưu trữ dữ liệu dưới dạng Document JSON nên mỗi một collection sẽ các các kích cỡ và các document khác nhau. Các dữ liệu được lưu trữ trong document kiểu JSON nên truy vấn sẽ rất nhanh. MongoDB lần đầu ra đời bởi MongoDB Inc., tại thời điểm đó là thế hệ 10, vào tháng Mười năm 2007, nó là một phần của sản phẩm PaaS (Platform as a Service) tương tự như Windows Azure và Google App Engine. Sau đó nó đã được chuyển thành nguồn mở từ năm 2009.
- Các thư viện và framework khác: Ngoài các công nghệ cơ bản, các nghiên cứu cần tìm hiểu về các thư viện và framework khác phổ biến như Redux, React Router, Mantine UI, và Axios. Các nghiên cứu cần hiểu cách sử dụng các thư viện này để tăng cường tính nhất quán đối với các tính năng của ứng dụng.

5.3. Các công cụ hỗ trợ

- Công cụ lập trình: Visual Studio Code.
- Công cụ thiết kế mô hình: StarUML, Power Designer, drawio.
- Công cụ hỗ trợ viết tài liệu: Microsoft Word 2019.
- Công cụ hỗ trợ thực thi: Google Chrome.

6. Những đóng góp chính của đề tài

- Xây dựng được website quản lý Trung tâm đào tạo bằng lái xe đáp ứng được nhiều chức năng và lợi ích của người dùng như:

- Tăng khả năng tiếp cận của trung tâm.
 - Tiết kiệm thời gian và chi phí.
 - Chủ động trong việc học tập.
 - Nâng cao chất lượng và trải nghiệm người dùng.
- Xây dựng giao diện đơn giản, thân thiện với người dùng, dễ sử dụng

7. Bố cục quyền luận văn

Bố cục của luận án, ngoài danh mục ký hiệu và chữ viết tắt, danh mục bảng, danh mục hình ảnh, phụ lục trình bày các bước và cách sử dụng hệ thống, danh mục tài liệu tham khảo, nội dung của luận văn như sau:

- Phần Giới thiệu bao gồm những nội dung chính: Đặt vấn đề, tóm tắt lịch sử giải quyết vấn đề, mục tiêu của đề tài, đối tượng, phạm vi và nội dung nghiên cứu, những đóng góp chính của đề tài, bố cục quyền luận văn.
- Phần Nội dung bao gồm:
 - Đặt vấn đề: Mô tả chi tiết bài toán, phân tích đánh giá và đưa ra giải pháp tiếp cận, các yêu cầu hệ thống, sơ đồ phân rã chức năng và sơ đồ usecase.
 - Cơ sở lý thuyết: Trình bày các lý thuyết của các ngôn ngữ, công nghệ và mô hình kiến trúc được áp dụng trong luận văn.
 - Thiết kế và cài đặt giải pháp: Mô tả kiến trúc tổng quan của hệ thống, thiết kế cơ sở dữ liệu, thiết kế giao diện chức năng của hệ thống.
 - Kiểm thử và đánh giá: Mô tả mục tiêu, kế hoạch, các trường hợp kiểm thử và kết quả kiểm thử, từ đó đưa ra đánh giá đối với các chức năng của hệ thống.
- Phần Kết luận: trình bày kết quả đạt được sau khi hoàn thiện, đưa ra kết quả đạt được, các tiêu chí, hướng phát triển đánh giá mức độ hoàn thành và chưa hoàn thiện cũng như những hạn chế, những việc chưa làm được của hệ thống.

PHẦN 2: NỘI DUNG

CHƯƠNG 1. MÔ TẢ BÀI TOÁN

Chương này sẽ mô tả các yêu cầu cơ bản của website trung tâm đào tạo bằng lái xe: mô tả chi tiết bài toán, các yêu cầu giao tiếp bên ngoài, các yêu cầu phi chức năng, chức năng chính của hệ thống và sơ đồ usecase.

1.1. Mô tả chi tiết bài toán

1.1.1 Bối cảnh sản phẩm

Hiện nay, ngành công nghệ thông tin đã có những bước phát triển nhanh chóng về ứng dụng của nó trong mọi lĩnh vực của cuộc sống trên phạm vi toàn thế giới nói chung và Việt Nam nói riêng. Ứng dụng tin học được người ta quan tâm và nhắc đến nhiều hơn bao giờ hết vì nó là một phần không thể thiếu trong cuộc sống hiện đại, góp phần đẩy mạnh công cuộc công nghiệp hoá - hiện đại hoá đất nước, tiến đến nền kinh tế tri thức. Máy vi tính cùng với những phần mềm là công cụ đắc lực giúp ta quản lý, tổ chức, sắp xếp và xử lý công việc một cách nhanh chóng và chính xác.

Cuộc sống ngày càng phát triển thì nhu cầu nâng cao trải nghiệm cuộc sống của mọi người cũng tăng lên. Do đó, nhu cầu học lái xe cũng ngày càng phổ biến và dễ chạy theo xu hướng xã hội các trung tâm đào tạo bằng lái xe được ra đời. Tuy nhiên, quá trình làm việc ở các trung tâm khá phức tạp và tốn nhiều thời gian. Học viên thường phải đến trung tâm để tìm hiểu thông tin, đăng ký các khóa học nên đôi khi dẫn đến tình trạng chờ đợi lâu, quá tải xử lý thông tin và tiêu tốn nhiều thời gian của mọi người.

Hầu hết các lĩnh vực trong xã hội đều áp dụng được các ứng dụng của công nghệ thông tin nên lĩnh vực giáo dục cũng không ngoại lệ, các trung tâm đào tạo cần có một website để giới thiệu bản thân, giảng dạy và trao đổi thông tin với học viên một cách dễ dàng hơn. Nhận thức được được nhu cầu cũng như xu thế phát triển của công nghệ, website quản lý Trung tâm đào tạo bằng lái xe được xây dựng để đáp ứng yêu cầu của các trung tâm đào tạo và học viên, giúp mọi người chủ động giảng dạy, học tập.

1.1.2 Các chức năng của website

- Đăng ký: cho phép người dùng khách đăng ký tài khoản để có thể đăng ký các khóa học.
- Đăng nhập: định dạng người dùng và đăng nhập vào hệ thống để thực hiện các chức năng của người dùng hệ thống.
- Xem thông báo: người dùng có thể xem thông báo về các thay đổi liên quan.

- Sửa thông tin cá nhân: người dùng có thể chỉnh sửa các thông tin cá nhân như số điện thoại, mail....
- Quản lý bài viết: cho phép admin quản lý bài viết theo ý muốn, riêng giảng viên và học viên được thực hiện chức năng xem và báo cáo bài viết.
- Quản lý tư vấn: cho phép admin, giảng viên và học viên trao đổi với nhau thông qua các câu hỏi có sẵn hoặc trao đổi trực tuyến.
- Quản lý khoá học: cho phép admin, giảng viên thực hiện các chức năng: thêm, sửa, xoá, tìm kiếm, xem khoá học, xem kết quả khóa học. Riêng học viên chỉ thực hiện được các chức năng: đăng ký, thanh toán, tìm kiếm, xem khoá học, xem thông báo, xem kết quả khóa học.
- Quản lý người dùng: cho phép admin thêm, xoá tài khoản của giảng viên, học viên.

1.1.3 Đặc điểm người sử dụng

Nhóm người sử dụng	Đặc trưng	Các chức năng	Vai trò	Mức độ quan trọng
Admin /Người quản trị/ Quản trị viên	Là người chịu trách nhiệm quản lý toàn bộ hệ thống, có tất cả các quyền truy cập vào hệ thống	<ul style="list-style-type: none"> - Đăng nhập. - Quản lý toàn bộ người dùng. - Quản lý khoá học (thêm, sửa, xoá, tìm kiếm, xem khoá học, xem kết quả khóa học). - Quản lý bài viết. - Quản lý tư vấn. - Xem thông báo. - Sửa thông tin cá nhân. 	Admin	Rất quan trọng
Giảng viên	Là người chịu trách nhiệm tạo ra các khoá học	<ul style="list-style-type: none"> - Đăng nhập. - Quản lý khoá học (thêm, sửa, xoá, tìm 	Nhân viên	Quan trọng

	cho học viên.	kiểm, xem khoá học, xem kết quả khóa học). - Quản lý bài viết. - Quản lý tư vấn. - Sửa thông tin cá nhân. - Xem thông báo		
Học viên	Là người đăng ký các khoá học trên hệ thống.	- Đăng nhập. - Quản lý khoá học (đăng ký, thanh toán, tìm kiếm, xem khoá học, kết quả khóa học, xem thông báo.). - Quản lý bài viết. - Quản lý tư vấn. - Xem thông báo. - Sửa thông tin cá nhân.	Khách hàng	Quan trọng
Người dùng khách	Là người dùng có quan tâm đến các khoá học nhưng chưa đăng ký tài khoản.	- Đăng ký. - Xem khoá học. - Tìm kiếm khoá học. - Xem khóa học. - Xem bài viết.	Khách hàng	Quan trọng

1.1.4 Môi trường vận hành

- Hệ điều hành: Windows 10 trở lên.
- Ram: từ 4GB trở lên.
- Trình duyệt web: Microsoft Edge, Opera, Google Chrome.

- Phải kết nối Internet.

1.1.5 Các ràng buộc thực thi và thiết kế

- Lập trình bằng ngôn ngữ: HTML, CSS, JavaScript, ReactJS và NextJS.
- Sử dụng công cụ Visual Studio Code để lập trình.
- Sử dụng MongoDB để lưu trữ CSDL và khởi tạo local host.
- Công cụ Draw.io và StarUML để thiết kế các sơ đồ và mô hình.

1.1.6 Các giả định và phụ thuộc

Các yếu tố có thể làm cho quá trình xây dựng website bị ảnh hưởng:

- Website sử dụng các thư viện bên ngoài hỗ trợ để xây dựng giao diện. Nếu phiên bản sử dụng bị lỗi thời có thể ảnh hưởng đến chất lượng hiển thị.
- Không thu thập đủ các yêu cầu của người dùng dẫn đến thiếu chức năng.
- Thiết kế cài đặt không đúng với chức năng đặc tả làm cho website kém chất lượng.

1.2. Các yêu cầu giao tiếp bên ngoài

1.2.1 Giao diện người dùng

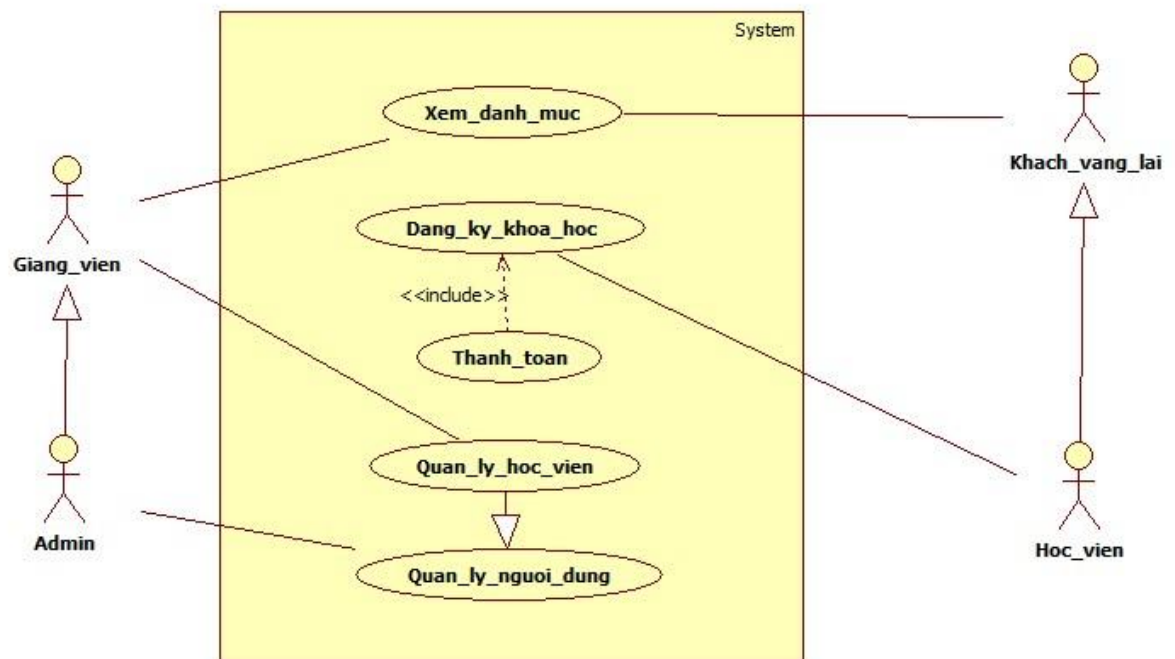
- Giao diện nền web: NextJS.
- Giao diện thân thiện với người dùng.
- Bố cục gọn gàng, dễ dùng, không rối mắt.
- Giao tiếp với người dùng qua màn hình và các thiết bị nhập xuất (bàn phím, chuột).
- Sử dụng ngôn ngữ dễ hiểu, phù hợp với người dùng.

1.2.2 Giao tiếp truyền thông tin

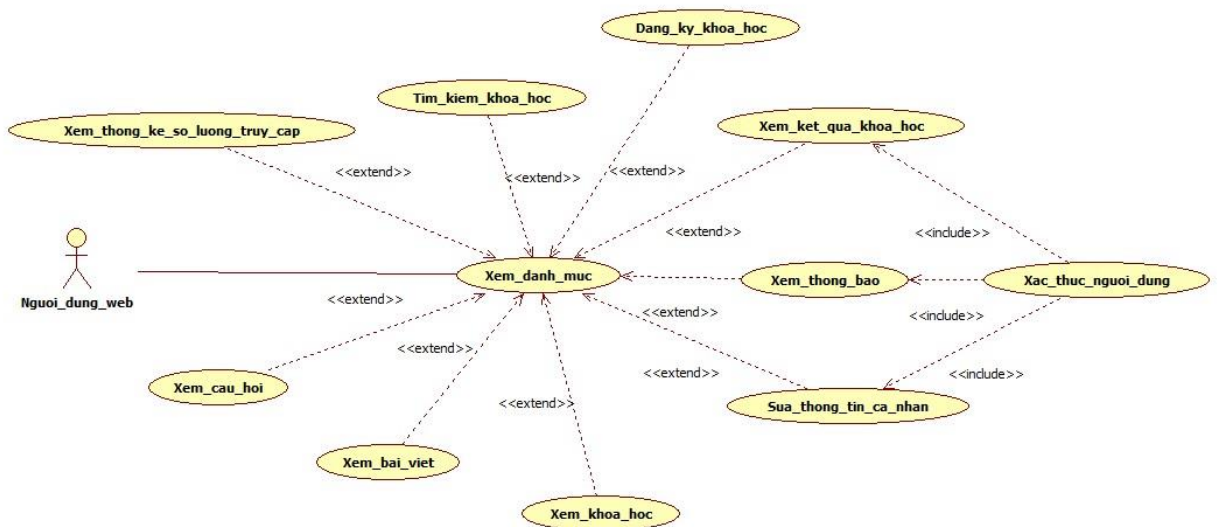
- Giao thức HTTP.

1.3. Các yêu cầu chức năng

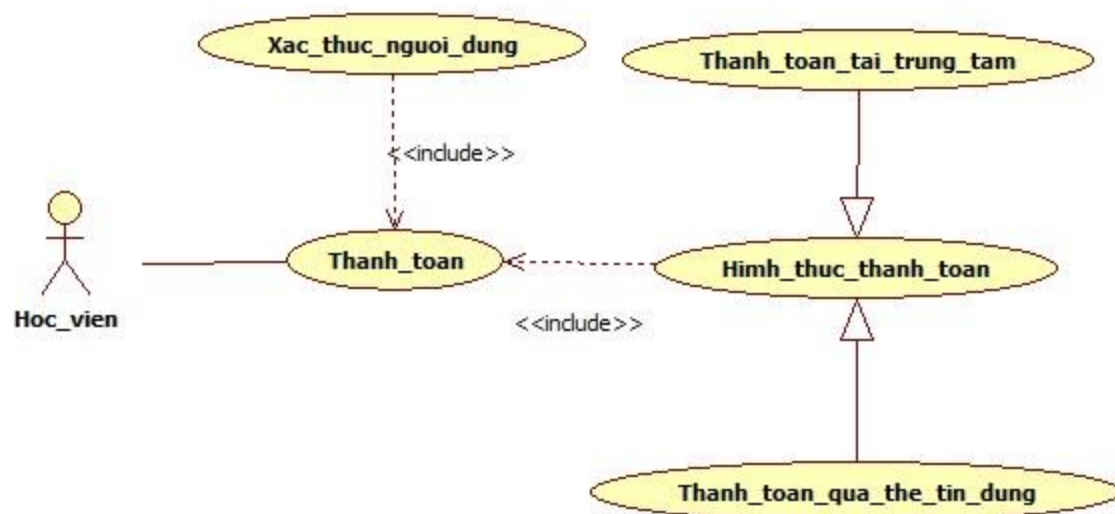
- Tổng quát:



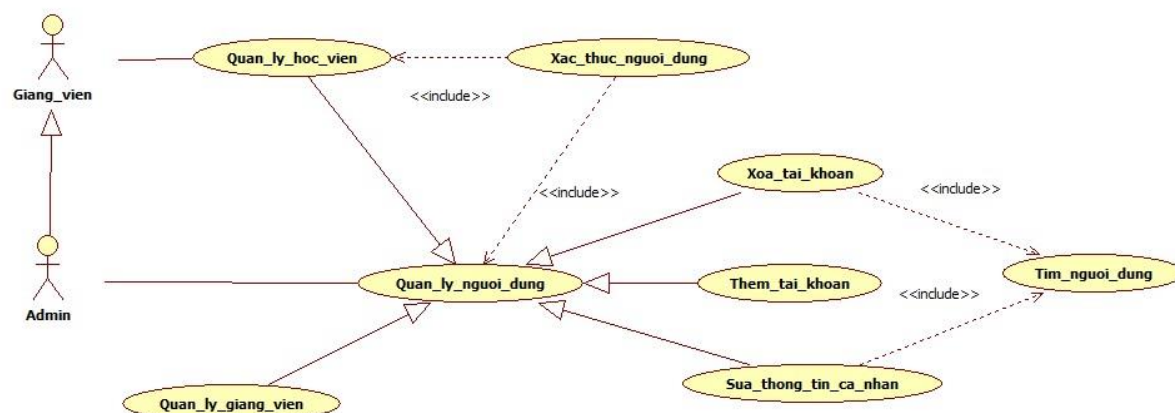
- Xem danh mục



- Thanh toán



- Quản lý người dùng



1.3.1 Chức năng đăng ký

Mã yêu cầu	UC-01
Tên yêu cầu	Đăng ký
Mức độ ưu tiên	Cao
Mô tả	Người dùng muốn đăng ký tài khoản để sử dụng hệ thống
Đối tượng	Người dùng khách
Tiền điều kiện	Không
Các thao tác xử lý	<ol style="list-style-type: none"> 1. Người dùng truy cập vào trang web. 2. Người dùng chọn đăng ký tài khoản. 3. Người dùng nhập đầy đủ thông tin cần đăng ký. 4. Hệ thống kiểm tra, xử lý thông tin. 5. Lưu vào CSDL. 6. Thông báo thành công/thất bại.

	7. Kết thúc sự kiện.
Kết quả	Người dùng đăng ký tài khoản thành công.

1.3.2 Chức năng đăng nhập

Mã yêu cầu	UC-02
Tên yêu cầu	Đăng nhập
Mức độ ưu tiên	Cao
Mô tả	Người dùng muốn đăng nhập tài khoản để sử dụng chức năng hệ thống
Đối tượng	Admin, học viên, giảng viên
Tiền điều kiện	Không
Các thao tác xử lý	<ol style="list-style-type: none"> 1. Người dùng truy cập vào trang web. 2. Người dùng chọn đăng nhập. 3. Người dùng nhập tên đăng nhập và mật khẩu. 4. Hệ thống kiểm tra thông tin đăng nhập. 5. Thông báo thành công/thất bại. 6. Kết thúc sự kiện.
Kết quả	Người dùng đăng nhập tài khoản thành công.

1.3.3 Chức năng xem thông báo

Mã yêu cầu	UC-03
Tên yêu cầu	Xem thông báo
Mức độ ưu tiên	Cao
Mô tả	Người dùng có nhu cầu xem thông báo về các khóa học, bài viết mới.
Đối tượng	Admin, học viên, giảng viên
Tiền điều kiện	Đăng nhập
Các thao tác xử lý	<ol style="list-style-type: none"> 1. Người dùng truy cập vào trang web. 2. Người dùng đăng nhập thành công. 3. Chọn xem thông báo. 4. Hệ thống hiển thị các thông báo. 5. Kết thúc sự kiện.
Kết quả	Người dùng xem được các thông báo cần thiết.

1.3.4 Chức năng quản lý người dùng

Mã yêu cầu	UC-04
Tên yêu cầu	Quản lý người dùng
Mức độ ưu tiên	Cao
Mô tả	Admin thực hiện được các chức năng thêm, tìm kiếm, thống kê và xóa tài khoản.
Đối tượng	Admin

Tiền điều kiện	Đăng nhập
Các thao tác xử lý	<ol style="list-style-type: none"> 1. Người dùng truy cập vào trang web. 2. Người dùng đăng nhập thành công. 3. Vào quản lý tài khoản. 4. Admin thực hiện tìm, thêm, xóa, thống kê tài khoản. 5. Cập nhật thông tin vào CSDL. 6. Kết thúc sự kiện.
Kết quả	Admin tìm, thêm, xóa, tài khoản thành công.

1.3.5 Chức năng quản lý khóa học

Mã yêu cầu	UC-05
Tên yêu cầu	Quản lý khóa học
Mức độ ưu tiên	Cao
Mô tả	Giảng viên hoặc admin sẽ thêm các khóa học mới vào trang web, sửa khóa học và xóa khóa học. Mọi người dùng sẽ tìm kiếm, xem các khóa học theo nhu cầu. Học viên đăng ký khóa học mong muốn.
Đối tượng	Admin, giảng viên, học viên, người dùng khách.
Tiền điều kiện	Không.
Các thao tác xử lý	<ol style="list-style-type: none"> 1. Người dùng truy cập vào trang web. 2. Người dùng vào xem khóa học. 3. Sau khi xem có thể đăng ký khóa học đối với học viên. 4. Admin, giảng viên thực hiện thêm khóa học: đăng tải các nội dung liên quan đến khóa học. Sau khi thêm khóa học có thể sửa khóa học, xóa khóa học. 5. Cập nhật thông tin vào CSDL. 6. Kết thúc sự kiện.
Kết quả	Đáp ứng nhu cầu về khóa học của người dùng.

1.3.6 Chức năng quản lý bài viết

Mã yêu cầu	UC-06
Tên yêu cầu	Quản lý bài viết
Mức độ ưu tiên	Cao
Mô tả	Học viên/giảng viên/admin sẽ tạo bài viết mới, bình luận bài viết, chia sẻ bài viết trên các nền tảng mạng xã hội, báo cáo bài viết mà mình muốn. Người dùng khách có thể xem các bài viết.
Đối tượng	Admin, giảng viên, học viên, người dùng khách.
Tiền điều kiện	Không.
Các thao tác xử lý	<ol style="list-style-type: none"> 1. Người dùng truy cập vào trang web.

	<ol style="list-style-type: none"> 2. Người dùng vào mục bài viết. 3. Thực hiện xem, thêm, xóa, báo cáo bài viết theo các phân quyền của người dùng. 4. Cập nhật thông tin vào CSDL. 5. Kết thúc sự kiện.
Kết quả	Đáp ứng nhu cầu về bài viết của người dùng.

1.3.7 Chức năng quản lý tư vấn

Mã yêu cầu	UC-07
Tên yêu cầu	Quản lý tư vấn
Mức độ ưu tiên	Cao
Mô tả	Học viên/giảng viên/admin sẽ trao đổi thắc mắc thông qua hệ thống câu hỏi có sẵn hoặc trao đổi trực tuyến. Người dùng khách có thể xem hệ thống câu hỏi.
Đối tượng	Admin, giảng viên, học viên, người dùng khách.
Tiền điều kiện	Không.
Các thao tác xử lý	<ol style="list-style-type: none"> 1. Người dùng truy cập vào trang web. 2. Người dùng vào mục tư vấn. 3. Thực hiện xem, đặt câu hỏi, trả lời câu hỏi theo các phân quyền của người dùng. 4. Cập nhật thông tin vào CSDL. 5. Kết thúc sự kiện.
Kết quả	Đáp ứng nhu cầu tư vấn thông tin của người dùng.

1.3.8 Chức năng sửa thông tin cá nhân

Mã yêu cầu	UC-08
Tên yêu cầu	Sửa thông tin cá nhân
Mức độ ưu tiên	Cao
Mô tả	Học viên/giảng viên/admin có quyền chỉnh sửa thông tin cá nhân như tên người dùng, mail, số điện thoại.
Đối tượng	Admin, giảng viên, học viên.
Tiền điều kiện	Đăng nhập
Các thao tác xử lý	<ol style="list-style-type: none"> 1. Người dùng truy cập vào trang web. 2. Người dùng đăng nhập tài khoản thành công. 3. Thực hiện chỉnh sửa thông tin cá nhân. 4. Cập nhật thông tin vào CSDL. 5. Kết thúc sự kiện.
Kết quả	Cập nhật thông tin được chỉnh sửa thành công.

1.4. Các yêu cầu phi chức năng

1.4.1 Yêu cầu thực thi

- Phần mềm chạy tốt trên cấu hình máy và hệ điều hành tối thiểu được đề ra.
- Không bị lỗi phần mềm khi chạy chương trình. Kết quả trả về sau khi thực thi phải chính xác.
- Tốc độ truy cập website nhanh và ổn định. Thời gian đáp ứng các thao tác của người dùng nhanh, dưới 5 giây.
- Hiệu suất làm việc đạt hơn 90%, hoạt động tốt, liên tục 24/24, tốn ít tài nguyên, không gây tình trạng treo máy, thao tác chậm chạp hay xuất hiện lỗi.
- Máy tính cần kết nối Internet và được cài đặt đầy đủ các phần mềm hỗ trợ chạy trang web.
- Hệ thống hoạt động tốt, tốn ít tài nguyên, thao tác ổn định và không xuất hiện lỗi khi người dùng sử dụng

1.4.2 Yêu cầu an toàn

- Người dùng chỉ được thao tác các chức năng nằm trong phạm vi cho phép được định sẵn.
- Đảm bảo về mặt khôi phục dữ liệu trong trường hợp dữ liệu bị mất mát.
- Các dữ liệu mà ứng dụng cung cấp phải đảm bảo chính xác về mặt thông tin.
- Ứng dụng hoạt động không làm ảnh hưởng hệ điều hành và các ứng dụng khác.
- Hệ thống không chứa virus, không chứa các đường link độc hại, bao gồm các đường link chứa mã độc.
- Cơ sở dữ liệu phải được sao lưu và phục hồi theo định kỳ
- Mỗi tài khoản thành viên trong website được bảo vệ một cách nghiêm túc, tránh tình trạng đánh cắp thông tin trái phép bằng bất kỳ hình thức nào.

1.4.3 Yêu cầu về hiệu suất

- Hệ thống đảm bảo khả năng chịu tải nhiều lượt truy cập cùng một lúc, thời gian phản hồi không quá 5 giây.
- Có khả năng tương thích cũng như hoạt động hiệu quả trên nhiều thiết bị và trình duyệt web khác nhau.

1.4.4 Yêu cầu về bảo mật

- Xây dựng cơ chế bảo mật bằng mật khẩu để quản lý tài khoản ứng với từng nhóm người dùng.

1.4.5 Các đặc điểm chất lượng sản phẩm

- Tính đúng đắn: Các chức năng của hệ thống phải thực hiện đúng theo yêu cầu.

- Tính khoa học: Cách xây dựng, tổ chức các chức năng của hệ thống phải khoa học. Xây dựng cơ sở dữ liệu hợp lý với hệ thống, giảm tải nguyên lưu trữ dữ liệu.
- Tính tinh cậy: Hệ thống phải đảm bảo các chức năng thực hiện tốt.
- Tính đáp ứng: Phần mềm chất lượng phải đáp ứng được nhu cầu và yêu cầu của người dùng, bao gồm tính linh hoạt và tính mở rộng để có thể cập nhật và mở rộng trong tương lai.
- Tính bảo mật: Phần mềm phải đảm bảo tính bảo mật cao, bảo vệ dữ liệu đối với lỗ hổng và tấn công từ bên ngoài.
- Hiệu suất: Phần mềm phải hoạt động hiệu quả, với tốc độ đáp ứng nhanh và không có quá trình chậm.
- Dễ sử dụng: Phần mềm phải có giao diện người dùng thân thiện, dễ sử dụng và hiệu quả, để người dùng không cần phải tìm hiểu quá nhiều để sử dụng.

1.4.6 Các yêu cầu khác

- Ngôn ngữ sử dụng trong website phải thuần Việt, từ ngữ đơn giản, dễ hiểu.
- Các nút lệnh được sắp xếp hợp lý
- Giao diện trực quan, thân thiện và dễ nhìn.
- Tốc độ xử lý nhanh chóng, chính xác.
- Dễ dàng bảo trì cũng như là nâng cấp hệ thống

CHƯƠNG 2. CƠ SỞ LÝ THUYẾT

2.1 Kiến thức tổng quan

Chương này nói về lý thuyết của một số ngôn ngữ, công nghệ, cơ sở dữ liệu và mô hình kiến trúc được áp dụng trong đề tài luận văn này, bao gồm các khái niệm, cấu trúc, nguyên lý hoạt động và hiệu quả mang lại.

2.2 Các ngôn ngữ và công nghệ được sử dụng

2.2.1 HTML

- HTML có tên đầy đủ là Hypertext Markup Language nghĩa là ngôn ngữ đánh dấu siêu văn bản. HTML thường được sử dụng để tạo và cấu trúc các phần trong trang web và ứng dụng, phân chia các đoạn văn, heading, link, blockquotes, ... HTML không phải là một ngôn ngữ lập trình mà chỉ là một ngôn ngữ đánh dấu. Điều này đồng nghĩa với việc HTML không thể thực hiện các chức năng “động”. Nói cách khác, HTML tương tự như phần mềm Microsoft Word, chỉ có tác dụng định dạng các thành phần có trong website.
- Cấu trúc của HTML: Một file code HTML được cấu thành bởi các phần tử HTML và các cặp thẻ. Song song đó, HTML gồm có nhiều dạng thẻ khác

nhau và mỗi thẻ sẽ có nhiệm vụ và ý nghĩa riêng. Ngoài ra, mỗi thẻ sẽ được bắt đầu và kết thúc bằng dấu ngoặc nhọn “<, >”. Các chữ giữa các dấu ngoặc này gọi là phần tử. Về cơ bản, cấu trúc của một trang HTML sẽ gồm 3 phần như sau:

- Phần khai báo loại file code có cấu trúc thẻ là `<!DOCTYPE html>`: Xuất hiện ở đầu hoặc trên cùng của file HTML. Qua phần này, người dùng sẽ biết được trình duyệt đang sử dụng để tạo trang là phiên bản HTML nào.
 - Phần khai báo ban đầu, khai báo về meta, little, javascript, css,... có cấu trúc bắt đầu bằng thẻ `<head>` và kết thúc với thẻ `</head>`: Chứa tiêu đề và các khai báo có thông tin nhằm phục vụ SEO. Trong đó, tiêu đề được hiển thị trên thanh điều hướng của trang web và là phần nội dung nằm giữa cặp thẻ `<title>` và `</title>`.
 - Phần chứa và hiển thị nội dung của trang web, có cấu trúc bắt đầu bằng thẻ `<body>` và kết thúc bằng thẻ `</body>`: Bao gồm tiêu đề trang web, logo, điều hướng chính và thanh tìm kiếm; nội dung chính như tiêu đề hoặc tiêu đề bài viết, nội dung bài viết, ngày đăng, tác giả,...; thanh bên hiển thị tiện ích con và điều hướng thứ cấp; chân trang cung cấp thông tin liên hệ, liên kết xã hội, bản quyền và điều hướng.
- HTML là một công cụ hữu hiệu giúp người dùng thiết kế và tạo nội dung nhanh chóng cho trang web. Nguyên nhân là bởi HTML sở hữu nhiều ưu điểm nổi bật như sau:
- Kho tài nguyên khổng lồ với cộng đồng người dùng rộng lớn.
 - Sử dụng mã nguồn mở nên người dùng có thể sử dụng miễn phí.
 - Hoạt động mượt mà trên hầu hết các trình duyệt.
 - Cách thức hoạt động đơn giản nên người học có thể dễ dàng nắm bắt được kiến thức để triển khai cho website của mình.
 - Có thể tích hợp với nhiều ngôn ngữ khác nhau như PHP, Java, NodeJs, Ruby,... giúp người dùng xây dựng được một trang web với nhiều tính năng hấp dẫn.
 - Sử dụng các markup ngắn gọn và có tính đồng nhất cao.
 - Quy định theo một tiêu chuẩn nhất định và được vận hành bởi World Wide Web Consortium (W3C).
- Nhược điểm của HTML:
- Chỉ áp dụng được cho web tĩnh, không có sự tương tác với người dùng. Nếu muốn trang bị cho web các tính năng tự động thì cần phải sử dụng dịch vụ của bên thứ 3.
 - Một số trình duyệt còn cập nhật chậm để hỗ trợ các phiên bản mới của HTML, đặc biệt là HTML5.

- Việc kiểm soát cách đọc và hiển thị file HTML của trình duyệt khó thực hiện.
- Chỉ áp dụng với cấu trúc nhất định, không có khả năng sáng tạo.

2.2.2 CSS

- CSS là viết tắt của Cascading Style Sheets, là một ngôn ngữ thiết kế được sử dụng nhằm mục đích đơn giản hóa quá trình tạo nên một website. CSS được ra mắt vào năm 1996 bởi World Wide Web Consortium (W3C).
- CSS xử lý một phần giao diện của trang web. Sử dụng CSS, bạn có thể kiểm soát màu sắc của văn bản, kiểu phông chữ, khoảng cách giữa các đoạn văn, cách các cột được đặt kích thước và bố cục, hình ảnh hoặc màu nền nào được sử dụng, thiết kế bố cục, các biến thể hiển thị cho các thiết bị và kích thước màn hình khác nhau cũng như hàng loạt các hiệu ứng khác.
- Cách CSS hoạt động đó chính là tìm kiếm dựa trên vùng chọn chẳng hạn như thẻ HTML, ID, class, v.v. Sau đó, nó sẽ áp dụng những thuộc tính buộc phải thay đổi lên các vùng đã chọn.
- CSS rất dễ học và dễ hiểu nhưng nó cung cấp khả năng kiểm soát mạnh mẽ việc trình bày tài liệu HTML. Thông thường nhất, CSS được kết hợp với các ngôn ngữ đánh dấu HTML hoặc XHTML.
- HTML và CSS có mối quan hệ gắn bó mật thiết với nhau. Nếu HTML là nền tảng của một trang web thì CSS là tất cả tính thẩm mỹ của toàn bộ trang web đó.
- Bố cục của một đoạn CSS: Bố cục của một đoạn CSS chủ yếu dựa vào hình hộp với mỗi hộp chiếm những khoảng trống trên trang web với các thuộc tính chính như:
 - Padding: Là các không gian xung quanh nội dung (ví dụ: không gian xung quanh đoạn văn bản).
 - Border: Là các đường nằm ngoài phần đệm.
 - Margin: Là khoảng cách bao quanh phía ngoài của phần tử.
- Cấu trúc của một đoạn CSS: Thông thường, một đoạn CSS sẽ bao gồm các phần: *{ vùng chọn {thuộc tính: giá trị; thuộc tính: giá trị; }*
- Đoạn CSS sẽ được khai báo bằng vùng chọn, các thuộc tính, giá trị nằm trong dấu ngoặc nhọn. Mỗi thuộc tính là một giá trị riêng ở dạng số, hoặc chính là tên của các giá trị đã có trong danh sách của CSS.
- Quy tắc khai báo đó chính là: thuộc tính và giá trị cần cách nhau bằng dấu hai chấm, mỗi dòng khai báo thuộc tính cần có dấu chấm phẩy cuối cùng. Các thuộc tính không bị giới hạn ở một vùng chọn.
- Trong đó: Bộ chọn (Selector): mẫu để chọn phần tử HTML mà bạn muốn định nghĩa phong cách. Bạn có thể áp dụng các selector cho các trường hợp sau:

- Tất cả những phần tử được định dạng theo một dạng cụ thể nào đó, ví dụ phần tử tiêu đề h2.
 - Thuộc tính id, class của phần tử.
 - Các phần tử có mối liên quan với các phần tử khác trong hệ thống cây phân cấp tài liệu.
 - Khai báo (Declaration): Khối khai báo có thể chứa một hoặc nhiều khai báo và chúng được phân tách với nhau bằng dấu chấm phẩy. Mỗi khai báo lại bao gồm tên & giá trị đặc tính CSS, được phân tách với nhau bằng dấu phẩy. Quy tắc khai báo CSS là chúng luôn phải kết thúc bằng dấu chấm phẩy, và khối khai báo phải nằm trong các dấu ngoặc móc.
 - Thuộc tính (Properties): Thuộc tính là các cách thức mà bạn có thể tạo kiểu cho một phần tử HTML. Vì vậy, với CSS, bạn chỉ cần lựa chọn thuộc tính mà bạn muốn tác động nhất trong bộ quy tắc bạn đã tạo ra.
 - Giá trị thuộc tính: Được nằm ở bên phải của thuộc tính. Việc lựa chọn một thuộc tính trong số đó phụ thuộc vào số lần xuất hiện của thuộc tính.
- Ưu điểm của CSS:
- Tăng tốc độ tải trang: CSS cho phép bạn sử dụng ít đoạn mã vì vậy tốc độ tải trang sẽ được cải thiện đáng kể. Ngoài ra, bạn còn có thể sử dụng một quy tắc CSS và áp dụng nó cho tất cả các lần xuất hiện của một thể nhất định trong tài liệu HTML.
 - Cải thiện trải nghiệm người dùng: CSS không chỉ làm cho các trang web dễ nhìn hơn, nó còn giúp các website có định dạng thân thiện với người dùng. Khi các nút và văn bản ở vị trí hợp lý và được sắp đặt tốt, trải nghiệm người dùng sẽ được cải thiện.
 - Thời gian phát triển nhanh: Với CSS, bạn có thể áp dụng các quy tắc và kiểu định dạng cụ thể cho nhiều trang bằng một chuỗi mã. Một biểu định kiểu xếp tầng có thể được sao chép trên một số trang web. Ví dụ: nếu bạn có các trang sản phẩm tất cả phải có cùng định dạng, giao diện, thì việc viết quy tắc CSS cho một trang sẽ đủ cho tất cả các trang cùng loại.
 - Thay đổi định dạng dễ dàng: Nếu bạn cần thay đổi định dạng của một nhóm trang cụ thể, bạn có thể dễ dàng thực hiện việc này với CSS mà không cần phải sửa từng trang riêng lẻ. Chỉ cần chỉnh sửa biểu định kiểu CSS tương ứng và bạn sẽ thấy các thay đổi được áp dụng cho tất cả các trang đang sử dụng biểu định kiểu đó.

- Khả năng tương thích trên các thiết bị: Thiết kế web đáp ứng là một vấn đề cần được chú trọng. Trong thời đại ngày nay, các trang web phải hiển thị đầy đủ và có thể điều hướng dễ dàng trên tất cả các thiết bị. Cho dù thiết bị di động hay máy tính bảng, máy tính để bàn hay thậm chí là TV thông minh, CSS kết hợp với HTML để tạo ra thiết kế đáp ứng.

2.2.3 JavaScript

- Javascript chính là một ngôn ngữ lập trình web rất phổ biến ngày nay. Javascript được tích hợp đồng thời nhúng vào HTML để hỗ trợ cho website trở nên sống động hơn. Chúng cũng đóng vai trò tương tự như một phần của website, cho phép Client-side Script từ người dùng tương tự máy chủ (Nodejs) để tạo ra những website động.
- Lịch sử phát triển của Javascript:
 - Brendan Eich chính là người đã phát triển Javascript tại Netscape với tiền thân là Mocha. Sau đó, Mocha được đổi thành LiveScript và cuối cùng mới đổi thành JavaScript.
 - Năm 1998, JavaScript với phiên bản mới nhất là ECMAScript 2 phát hành và đến năm 1999 thì ECMAScript 3 được ra mắt.
 - Năm 2016, ứng dụng JavaScript đã đạt kỷ lục lên tới 92% website sử dụng, đồng thời cũng được đánh giá là một công cụ cực kỳ quan trọng đối với lập trình viên.
- Cách hoạt động của javascript:
 - Thông thường, JavaScript sẽ được nhúng trực tiếp vào một website hoặc chúng được tham chiếu qua file .js hoặc .JavaScript.
 - Đây là một ngôn ngữ đến từ phía Client nên Script sẽ được download về máy client khi truy cập.
 - Tại đây, chúng sẽ được hệ thống xử lý. Vì vậy, bạn không cần phải tải về máy server rồi chờ cho chúng xử lý xong mới phản hồi được kết quả đến client.
- Mục đích của javascript:
 - Thay đổi nội dung HTML: Một trong số nhiều phương thức HTML JavaScript chính là getElementById (). Chúng được sử dụng để tìm một phần tử của HTML với id ="demo" và dùng để thay đổi nội dung của phần tử (Internal HTML) sang thành "Hello JavaScript"
 - Thay đổi giá trị thuộc tính HTML: Tổng quan về javascript còn có thể sử dụng để thay đổi các giá trị của thuộc tính. Ví dụ: thay đổi thuộc tính src (source) của tag.

- Thay đổi kiểu HTML: Đây chính là một hoạt động biến thể của việc thay đổi thuộc tính của HTML ở trên. Ví dụ:
`document.getElementById('demo').style.fontSize = '35px;`
 - Ẩn các phần tử HTML: Một hoạt động tiếp theo là Javascript có thể ẩn được các phần tử HTML. Chúng có thể được thực hiện thông qua hoạt động thay đổi kiểu hiển thị các phần tử HTML.
 - Hiển thị các phần tử HTML: Một điểm đặc biệt là JavaScript có thể hiển thị được các yếu tố HTML ẩn. Đồng thời, cũng có thể thực hiện được thông qua cách thay đổi kiểu hiển thị phần tử.
- Ưu điểm:
- Chương trình rất dễ học.
 - Những lỗi Javascript rất dễ để phát hiện, từ đó giúp bạn sửa lỗi một cách nhanh chóng hơn.
 - Những trình duyệt web có thể dịch thông qua HTML mà không cần sử dụng đến một compiler.
 - JS có thể hoạt động ở trên nhiều nền tảng và các trình duyệt web khác nhau.
 - Được các chuyên gia đánh giá là một loại ngôn ngữ lập trình nhẹ và nhanh hơn nhiều so với các ngôn ngữ lập trình khác.
 - JS còn có thể được gắn trên một số các element hoặc những events của các trang web.
 - Những website có sử dụng JS thì chúng sẽ giúp cho trang web đó có sự tương tác cũng như tăng thêm nhiều trải nghiệm mới cho người dùng.
 - Người dùng cũng có thể tận dụng JS với mục đích là để kiểm tra những input thay vì cách kiểm tra thủ công thông qua hoạt động truy xuất database.
 - Giao diện của ứng dụng phong phú với nhiều thành phần như Drag and Drop, Slider để cung cấp đến cho người dùng một Rich Interface (giao diện giàu tính năng).
 - Giúp thao tác với người dùng phía Client và tách biệt giữa các Client với nhau.
- Nhược điểm:
- JS Code Snippet khá lớn.
 - JS dễ bị các hacker và scammer khai thác hơn.
 - JS cũng không có khả năng đa luồng hoặc đa dạng xử lý.
 - Có thể được dùng để thực thi những mã độc ở trên máy tính của người sử dụng.
 - Những thiết bị khác nhau có thể sẽ thực hiện JS khác nhau, từ đó dẫn đến sự không đồng nhất.

- Vì tính bảo mật và an toàn nên các Client-Side Javascript sẽ không cho phép đọc hoặc ghi các file.
- JS không được hỗ trợ khi bạn sử dụng ở trong tình trạng thiết bị được kết nối mạng.

2.2.4 NextJs

- Next.js là một open-source React front-end framework được bổ sung các tính năng như Server-Side Rendering (SSR) và Static Site Generation (SSG). Next.js được xây dựng dựa trên thư viện React có nghĩa là Next.js lấy những lợi thế của React và bổ sung các tính năng.
- Server-Side Rendering: SSR cho phép máy chủ access tất cả required data và xử lý javascript để hiển thị lên trang web. Theo cơ chế này thì hầu hết các xử lý logic đều ở phía máy chủ. Trong đó, máy chủ thực hiện xử lý và tiến hành các thao tác với cơ sở dữ liệu để thông dịch (render) ra thành HTML, sau đó gửi phản hồi cho khách hàng. Trình duyệt của khách hàng cũng sẽ chỉ hiển thị HTML này.
- Search Engine Optimization (SEO): Sử dụng SSR cũng mang lại cho bạn lợi thế về SEO, giúp trang web của bạn hiển thị cao hơn trên các trang kết quả của công cụ tìm kiếm. SSR làm cho các trang web xếp hạng tốt hơn cho SEO vì chúng tải nhanh hơn và nhiều nội dung trang web có thể được quét bởi các trình theo dõi SEO.
- Next.js cũng cho phép bạn chỉnh sửa thẻ <head> của một trang web, điều mà bạn không thể thực hiện trong React. Thẻ <head> góp phần vào xếp hạng SEO của trang web.
- Các tính năng của NextJS
 - Server-Side rendering (SSR)
 - Static site generation (SSG)
 - Single-Page application (SPA)
 - Development of faster application
 - Optimization of pages
 - SEO websites
 - Automatic code splitting
- Ưu điểm:
 - Split Code: Nó tự động breaks code để làm cho trang tải nhanh hơn. Đó là lý do tại sao hầu hết các trang web lớn sử dụng Next.js đều có khả năng tải trang nhanh hơn mặc dù chúng được xây dựng cho một lượng lớn người xem.
 - Brings organic traffic: Các trang web được phát triển bằng Next JS không chỉ nhanh mà còn đơn giản để tìm kiếm và cung cấp trải nghiệm người dùng tuyệt vời. Ba yếu tố: tốc độ, cấu trúc và trải nghiệm người

dùng, là những yếu tố xếp hạng quan trọng sẽ tăng thứ hạng trên công cụ tìm kiếm của Google trang web của bạn.

- Trải nghiệm người dùng tốt: Việc thuận lợi nhất của việc sử dụng Next JS là trải nghiệm người dùng, chỉ đứng thứ hai sau tốc độ. Marketers yêu thích sự độc lập trong thiết kế, đặc biệt là trong ngành thương mại điện tử, nơi nhiều cửa hàng trực tuyến trông giống nhau và có một trải nghiệm người dùng tốt sẽ giúp cửa hàng nổi bật hơn so với các đối thủ cạnh tranh.
- Bảo mật: An toàn vì nó không có liên kết trực tiếp đến database, dependencies, user data hoặc thông tin bí mật khác.
- Thời gian tải nhanh hơn: Vì các trang web JS là tĩnh, chúng cực kỳ nhanh và khách truy cập sẽ rất hài lòng với kết quả. Nó cũng có thể tự động tối ưu hóa các trang bất cứ khi nào nó được yêu cầu.
- Cộng đồng hỗ trợ lớn: Next.js là React framework nên rất dễ nhận được sự hỗ trợ nhanh chóng khi bạn cần. Bạn không cần phải xây dựng nó từ đầu, vì sẽ luôn có một nhà phát triển React hỗ trợ nhanh chóng cho bạn.
- Server-side rendering: Cải thiện khả năng bảo vệ dữ liệu và tuân thủ PIPA. Server-side rendering có các ưu điểm sau:
 - Cải thiện hiệu năng
 - Hỗ trợ SEO tốt hơn
 - Social sharing
 - Có ít vấn đề hơn với khả năng tương thích của trình duyệt.

2.2.5 ReactJs

- ReactJS được hiểu nôm na là một thư viện mã nguồn mở chứa nhiều JavaScript và người tạo ra ReactJS chính là ông trùm với cái tên quen thuộc Facebook, được ra đời vào năm 2013. Mục đích của việc tạo ReactJS là tạo ra các ứng dụng web nhanh, hiệu quả và hấp dẫn với nỗ lực viết mã tối thiểu. Mục tiêu chính của ReactJS là bất kỳ trang web nào sử dụng ReactJS phải mượt mà, nhanh chóng, khả năng mở rộng cao và dễ thực hiện.
- Nhìn chung, các tính năng và điểm mạnh của ReactJS thường đến từ việc tập trung vào các phần riêng lẻ. Do đó, khi làm việc với web thay vì toàn bộ ứng dụng của một trang web bằng ReactJS, các nhà phát triển tính năng có thể tách rời và chuyển đổi giao diện người dùng từ những cách phức tạp và biến nó thành những phần đơn giản hơn. Điều này có nghĩa là kết xuất dữ liệu không chỉ ở phía máy chủ. Thực hiện với các vị trí, nhưng bạn cũng có thể thực hiện tại vị trí khách hàng khi sử dụng ReactJS.
- Những thành phần chính của ReactJS:
 - Redux:

- Đây là một phần rất quan trọng của ReactJS và không ai sử dụng Redux mà không biết. ReactJS không có mô-đun chuyên dụng để xử lý dữ liệu, vì vậy ReactJS chia chế độ xem thành các thành phần nhỏ hơn mà bạn thiết lập độc lập và làm cho chúng liên quan chặt chẽ hơn.
- Các liên kết và mối quan hệ giữa các thành phần trong ReactJS đòi hỏi sự chú ý đặc biệt, bởi vì luồng dữ liệu một chiều từ cấp độ mẹ đến cấp độ con là luồng dữ liệu duy nhất trong ReactJS. Nó áp dụng cho các dự án, nhưng bất chấp những điều cần lưu ý, nó có một mặt tốt đẹp khác. Đó là về việc thúc đẩy tất cả các tính năng và vai trò khi ReactJS sử dụng engine.
- Virtual Dom
 - Đây là phần mà hầu như tất cả các framework đều sử dụng dom ảo và ReactJS sử dụng nó khi dom ảo thay đổi. Điều đặc biệt ở đây là bạn không phải thao tác trực tiếp với dom mà có thể xem các lượt xem và thay đổi.
 - Vì Virtual Dom hoạt động như một mô hình và một khung nhìn, nên việc thay đổi một trong hai yếu tố này sẽ thay đổi yếu tố kia và ngược lại mà không cần thao tác trực tiếp. Tuy nhiên, có thể thực hiện cơ chế mù dữ liệu. Mục đích của việc này là tăng tốc đáng kể ứng dụng của bạn, đây là một trong những lợi thế lớn nhất của việc sử dụng Virtual Dom.
- Lợi ích khi sử dụng ReactJS:
 - Tạo cho chính bản thân ReactJS một dom ảo, đây là nơi các component được tồn tại trên đó. Tạo một dom như thế này sẽ cải thiện hiệu suất rất nhiều. Nếu bạn cần thay đổi hoặc cập nhật một phép tính trong dom, ReactJS sẽ tính toán nó trước và phần còn lại sẽ chạy trong dom để thực hiện công việc, vì vậy ReactJS có thể tránh các thao tác cần thiết trong dom mà không phải trả thêm phí.
 - Việc viết các đoạn mã JS sẽ trở nên dễ dàng sử dụng một cú pháp đặc biệt đó chính là cú pháp JSX, cú pháp này cho phép bạn kết hợp mã HTML và Javascript. Ngoài ra, bạn có thể thêm mã bổ sung vào hàm kết xuất mà không cần nối chuỗi. Đây được coi là một trong những tính năng thú vị của ReactJS, việc chuyển đổi các đoạn HTML sang các constructor đều được thực hiện bởi trình chuyển đổi chính là JSX.
 - Khi khởi chạy ReactJS, đừng quên cài đặt thêm ứng dụng mở rộng Chrome chuyên dành cho ReactJS. Điều này làm cho việc debug mã của bạn dễ dàng hơn. Sau khi cài đặt, bạn có thể nhìn thẳng vào mái Virtual Dom điều đó đồng nghĩa với việc bạn đang theo dõi một cây dom thông thường

- Đây là một trong những điều đặc biệt và độc đáo của ReactJS. Đây cũng là một vấn đề lớn đối với các khuôn khổ JS vì mặc dù có nhiều cải tiến nhưng hầu hết các khuôn khổ JS không thân thiện với công cụ tìm kiếm. ReactJS hỗ trợ hiển thị và trả về trình duyệt dưới dạng trang web khi bạn chạy ReactJS trên máy chủ và Dom ảo, vì vậy chúng tôi rất tự hào nói rằng chúng tôi không thuộc nhóm không thân thiện với SEO. Bởi vì điều này, React rất phù hợp với tính cách thân thiện với SEO của nó.
- JavaScript Developer:
 - ReactJS là một thư viện JavaScript dành riêng để giúp các nhà phát triển tạo giao diện người dùng và giao diện người dùng. Khi nói đến lập trình ứng dụng front-end, các lập trình viên thường phải làm việc trên hai thành phần chính: giao diện người dùng và xử lý các tương tác của người dùng. Giao diện người dùng là tập hợp các phần tử mà tất cả các ứng dụng hiển thị.
 - Trước ReactJS, các lập trình viên rất khó tạo giao diện người dùng bằng “vanilla JavaScript” (JavaScript thuần túy) và JQuery. Điều này có nghĩa là quá trình phát triển ứng dụng mất nhiều thời gian hơn và dễ xảy ra lỗi và rủi ro hơn. Năm 2011, cộng tác viên của Facebook, Jordan Walke, đã ra mắt ReactJS với mục tiêu chính là cải thiện quy trình phát triển giao diện người dùng.
 - Ngoài ra, để tăng tốc quá trình phát triển và giảm rủi ro có thể phát sinh khi viết mã, React cung cấp khả năng tạo mã có thể tái sử dụng bằng cách giới thiệu hai khái niệm quan trọng bao gồm JSX và Virtual DDO.
 - ReactJs chỉ là view Library: Là một thư viện riêng biệt của Facebook nó giúp hiển thị các chế độ xem. Ngoài ra, nó hỗ trợ tạo các thành phần giao diện người dùng tương tác, trạng thái và luôn có thể tái sử dụng. Là nơi xây dựng các thành phần, React đóng vai trò vừa là máy khách vừa là máy chủ, cho phép hai phần này kết nối với nhau. Một điểm đặc biệt nữa là React không phải là một framework MVC. Vì vậy, đừng ngạc nhiên khi React không có các mô hình và bộ điều khiển. Nếu bạn cần thao tác với ReactJS, bạn sẽ phải kết hợp nó với các thư viện khác.
 - Sử dụng ít State components: State là nơi lưu giữ linh hồn của ứng dụng của bạn. Đặc biệt là vì phản hồi trong ứng dụng của bạn ngày càng mở rộng, bạn cần giữ trạng thái đơn giản. State làm phức tạp việc kiểm tra, State phải làm với việc kết xuất để cho biết liệu nó đã được khởi tạo chưa, dữ liệu trạng thái nào đã thay đổi và liệu trình kết xuất có quay trở lại hay không. Ngoài ra, State chỉ tồn tại trong các thành phần có trao đổi dữ liệu với thế giới bên ngoài, và việc sử dụng State là không cần

thiết. Lưu ý rằng State chỉ có thể được sử dụng khi cần thiết và phản ánh đúng trạng thái của thành phần. .

- Hãy giữ Components luôn nhỏ gọn: Mọi lập trình viên cần biết cách giữ các hàm / lớp nhỏ gọn hơn để làm cho phần mềm dễ hiểu và dễ bảo trì hơn. Khi nói đến ReactJS, tôi giữ các thành phần của mình càng nhỏ càng tốt để tôi có thể sử dụng lại chúng và đạt được hiệu suất tốt nhất. Sự phân chia phụ thuộc vào cấp độ của đội.
 - Kết hợp với Redux.js: Trên thực tế, React chỉ là một khung nhìn, vì vậy React nên được kết hợp với những thứ như Redux và Flux. Một luồng dữ liệu luôn cần thiết, Redux là một trong những nguồn dữ liệu được sử dụng phổ biến nhất và tư duy của React cũng khá tốt. Bạn cũng có thể sử dụng JSX, ES6, Babel, Webpack và NPM. JSX là một trong những tính năng tốt nhất của React, nhưng mọi thứ chúng tôi đã viết đều hiển thị, và khi bạn cũng tận dụng các tính năng mới của ES6 kết hợp với biên dịch babel, NPM và Webpack là hai thứ trong một sẽ trở thành. Sử dụng thư viện đóng một vai trò hỗ trợ.
 - Các trình Dev Tools của Redux và React: Sử dụng hai phần tử này, bạn có thể nhanh chóng gỡ lỗi và tìm lỗi trong ứng dụng của mình và kiểm tra lại các thành phần React của bạn với trạng thái Props và State của mỗi thành phần. Ngoài ra, nó giúp quan sát trạng thái của hành động và lý do thay đổi trạng thái và hoàn nguyên về trạng thái trước đó.
- Ưu điểm của ReactJS:
- Thích hợp cho nhiều loại trang web. ReactJS giúp tạo trang web dễ dàng hơn vì bạn không phải viết mã nhiều như tạo trang web chỉ sử dụng JavaScript hoặc HTML. Ngoài ra, nó đã cung cấp tất cả các loại “đồ chơi” mà bạn có thể sử dụng. nó trong nhiều tình huống.
 - Tái sử dụng thành phần: Nếu bạn muốn xây dựng một thành phần đủ linh hoạt để đáp ứng ‘yêu cầu’ của nhiều dự án khác nhau, chỉ cần dành thời gian tạo và sử dụng lại hầu hết mọi thành phần trong các dự án tiếp theo là được. Bạn có thể làm điều này không chỉ với ReactJS mà còn với các framework như Flutter.
 - Có thể sử dụng cho các ứng dụng di động: Hầu hết chúng ta đều biết ReactJS được sử dụng để phát triển web, nhưng nó thực sự được sinh ra để làm hơn thế. Nếu bạn cần phát triển thêm các ứng dụng di động, hãy sử dụng React Native. Đây là một khuôn khổ khác do chính Facebook phát triển cho phép bạn dễ dàng “chia sẻ” các thành phần và sử dụng lại logic nghiệp vụ trong các ứng dụng của mình.
 - Công cụ phát triển web hot nhất hiện nay. Nếu bạn nhìn vào số liệu thống kê của Google Xu hướng cho Việt Nam trong hình ảnh bên dưới,

bạn có thể duyệt qua các trang web việc làm hàng đầu tại Việt Nam như Topdev, Itviec và hơn thế nữa. Số lượng việc làm cho các nhà phát triển React là rất cao. , một mức lương rất hấp dẫn và mức độ phổ biến hiện tại của ReactJS trên thị trường Việt Nam.

- Thân thiện với SEO: SEO là một phần thiết yếu để làm cho trang web của bạn xuất hiện cao hơn trong các tìm kiếm của Google. ReactJS về cơ bản là một thư viện JavaScript. Công cụ tìm kiếm của Google thu thập thông tin và lập chỉ mục mã JavaScript, nhưng nó cũng yêu cầu các thư viện khác hỗ trợ điều này.

2.2.6 Mantine UI

- Mantine là thư viện UI hỗ trợ React, tập trung vào việc cung cấp trải nghiệm tuyệt vời cho người dùng và nhà phát triển. Quá trình phát triển Mantine được bắt đầu vào tháng 1 năm 2021 và phiên bản 1.0 được xuất bản vào ngày 3 tháng 5 năm 2021.
- Xây dựng các ứng dụng web có thể truy cập đầy đủ chức năng nhanh hơn bao giờ hết – Mantine bao gồm hơn 100 thành phần có thể tùy chỉnh và 50 hook để hỗ trợ bạn trong mọi tình huống.
- Mỗi thành phần Mantine hỗ trợ ghi đè kiểu cho mọi phần tử bên trong bên trong bằng các lớp hoặc kiểu nội tuyến. Tính năng này cùng với các tùy chọn tùy chỉnh khác cho phép bạn thực hiện mọi sửa đổi trực quan đối với các thành phần và điều chỉnh chúng để phù hợp với hầu hết mọi yêu cầu thiết kế.

2.3 Cơ sở dữ liệu MongoDB

- MongoDB là một hệ quản trị cơ sở dữ liệu mã nguồn mở, là CSDL thuộc NoSql và được hàng triệu người sử dụng.
- MongoDB là một database hướng tài liệu (document), các dữ liệu được lưu trữ trong document kiểu JSON thay vì dạng bảng như CSDL quan hệ nên truy vấn sẽ rất nhanh.
- Với CSDL quan hệ chúng ta có khái niệm bảng, các cơ sở dữ liệu quan hệ (như MySQL hay SQL Server...) sử dụng các bảng để lưu dữ liệu thì với MongoDB chúng ta sẽ dùng khái niệm là collection thay vì bảng
- So với RDBMS thì trong MongoDB collection ứng với table, còn document sẽ ứng với row, MongoDB sẽ dùng các document thay cho row trong RDBMS.
- Các collection trong MongoDB được cấu trúc rất linh hoạt, cho phép các dữ liệu lưu trữ không cần tuân theo một cấu trúc nhất định.
- Thông tin liên quan được lưu trữ cùng nhau để truy cập truy vấn nhanh thông qua ngôn ngữ truy vấn MongoDB

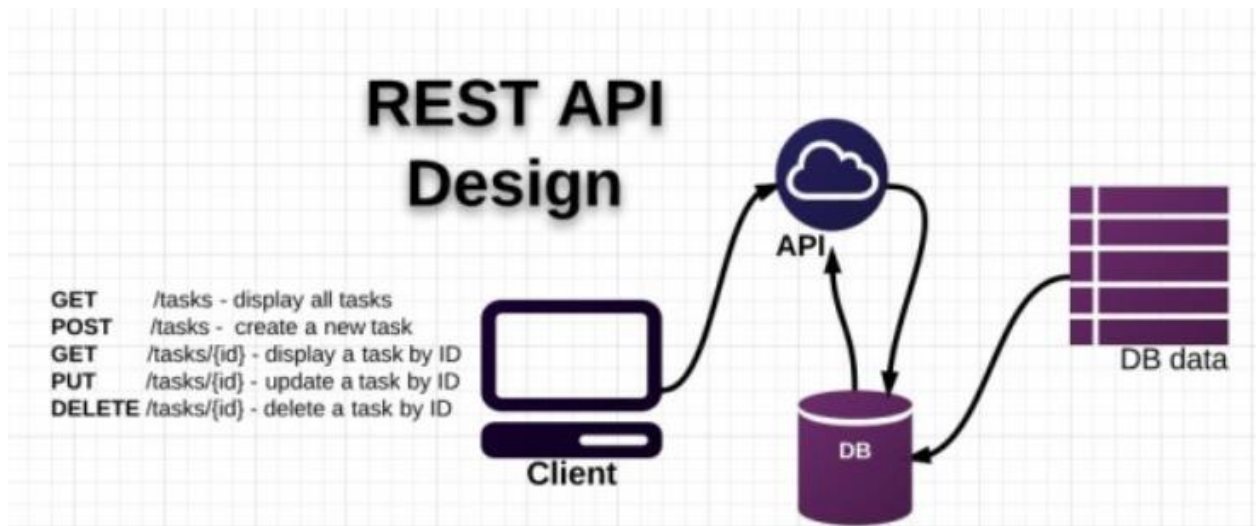
- Cấu trúc của MongoDB: Thay vì các bản ghi, trường dữ liệu như trong SQL, MongoDB sử dụng lưu trữ dữ liệu dưới dạng JSON nên từ đó mỗi Collection sẽ sở hữu những kích thước và các document riêng. Vì vậy, nó có thể sử dụng lưu trữ các dữ liệu lớn có kích thước và độ phức tạp đa dạng (Big Data).
- NoSQL hay Non-Relational SQL) là dạng cơ sở dữ liệu opensource được ra đời như mô hình cải tiến hơn về tính năng và tốc độ so với các hệ quản trị cơ sở dữ liệu RDBMS. NoSQL sử dụng dữ liệu kiểu JSON với các cặp giá trị key và value có thể xử lý nhanh và đạt hiệu suất cao cũng như có khả năng mở rộng tránh bị ràng buộc bởi các khóa như trong SQL nên nó được sử dụng khá phổ biến hiện nay nhờ sự linh hoạt mang lại.
- MongoDB có khá nhiều những tính năng hỗ trợ người dùng như:
 - Sử dụng truy vấn Ad hoc: Đây là một trong những tính năng hiệu quả nhất của chương trình. MongoDB truy vấn Ad hoc sẽ hỗ trợ các trường, truy vấn vào phạm vi và truy xuất các biểu thức để trả về những trường document cụ thể gồm những JavaScript function do người dùng xác định hoặc những truy vấn được cấu hình và trả lại những kết mẫu kết quả có kích thước xác định.
 - Khả năng nhân rộng: Khả năng nhân rộng Replication hay còn được hiểu là nhân bản, tức là sẽ có những bản sao tương tự những phiên bản mà bạn đang sử dụng. Nhu cầu lưu trữ dữ liệu lớn do cơ sở dữ liệu ngày càng đa dạng, chính vì vậy nó càng đòi hỏi dữ liệu phải được toàn vẹn và tránh bị mất mát trước những sự cố trong quá trình xử lý.
- Khả năng nhân bản của MongoDB sẽ hỗ trợ rất hiệu quả trong phòng ngừa những sự cố hư hỏng cơ sở dữ liệu xảy ra. Đặc biệt hơn MongoDB còn cho phép phân chia thứ cấp bản sao:
 - Bản sao chính được sử dụng để đọc và ghi dữ liệu đồng thời với bản gốc.
 - Bản sao thứ cấp dùng bản sao tích hợp để duy trì dữ liệu trên bản sao chính của bản gốc.
- Khả năng cân bằng tải: Thông qua Sharding, MongoDB phân chia tỷ lệ theo chiều ngang giúp người sử dụng chọn lựa chọn 1 Shard key, xác định những data được phân phối trong Collection. Về cơ bản, những dữ liệu được phân chia, phân phối đồng đều dựa trên những Shard key. Hỗ trợ chạy trên nhiều máy chủ, cân bằng tải hoặc copy dữ liệu để giúp hệ thống luôn hoạt động ngay cả khi có lỗi về phần cứng.
- Hỗ trợ lưu trữ các tệp dữ liệu: MongoDB sử dụng như hệ thống tệp nhờ khả năng lưu trữ tệp dữ liệu từ đó hỗ trợ cân bằng tải và copy dữ liệu từ nhiều máy tính. Trong đó, hệ thống tệp GridFS sẽ phân chia từng tệp ra những phần hay đoạn lưu thành những document riêng và người dùng có thể truy xuất GridFS thông qua tiện ích Mongofiles.

- Tập hợp: MongoDB cung cấp 3 phương pháp tập hợp chính bao gồm Aggregation Pipeline, Single-purpose Aggregation và Mapreduce. Trong đó, Aggregation Pipeline được đánh giá là mang lại hiệu suất cao hơn so với những hoạt động tổng hợp khác.
- Quy định kích thước giới hạn Collection: MongoDB có hỗ trợ cho những Collection giới hạn. Khi dữ liệu vượt quá mức giới hạn cho phép thì những tài liệu cũ trước đó sẽ bị xóa mà không cần người dùng sử dụng bất kỳ câu lệnh điều khiển nào.
- Khả năng giao dịch: MongoDB mang lại tính năng trợ giúp cho những giao dịch đa tài liệu ACID và bắt đầu có hiệu dụng kể từ version 4.0 ra mắt vào tháng 6 năm 2018.
- Ưu điểm:
 - Do sử dụng lưu trữ dữ liệu dưới dạng Document JSON nên mỗi Collection đều có thiết kế kích thước và thuộc những document khác nhau. Tuy nhiên chúng lại khá linh hoạt khi tiến hành lưu trữ bởi vậy nếu người dùng muốn lưu thêm dữ liệu chỉ cần insert là xong.
 - Những dữ liệu lưu trong hệ thống của MongoDB không bị ràng buộc nhau, không bị phụ thuộc bởi khóa chính hay khóa phụ như RDBMS nên khi thực hiện các thao tác thêm, sửa, xóa thì sẽ đơn giản hơn việc kiểm tra ràng buộc như trong RDBMS.
 - Khả năng mở rộng tốt của MongoDB được đánh giá cao bởi nó sử dụng cụm các node chứa những dữ liệu giao tiếp được với nhau được gọi là Cluster. Từ đó để mở rộng bạn chỉ cần thêm một node vào hệ thống.
 - Những index cho từng dữ liệu sẽ là tự động để hỗ trợ truy vấn thông tin nhanh và đạt hiệu suất cao.
 - Tốc độ truy vấn đáng kể của MongoDB là một lợi thế so với những hệ quản trị cơ sở dữ liệu khác. Thử nghiệm cho thấy ở một lượng dữ liệu chung, MongoDB có khả năng insert nhanh gấp 100 lần so với HTQ SQL.
- Nhược điểm:
 - Chính vì không bị ràng buộc các trường dữ liệu như trong RDBMS nên bạn cần phải hết sức cẩn thận khi thao tác trên những dữ liệu lưu trữ để tránh những kết quả ngoài ý muốn tác động xấu đến dữ liệu.
 - Đôi khi sẽ tồn bộ nhớ do dữ liệu lưu ở dạng key-value nên những Collection chỉ khác về value mà có phần giống nhau về key. Và vì không hỗ trợ join như RDBMS nên sẽ có tình trạng dư thừa dữ liệu.
 - Khi thực hiện các tác vụ thêm, sửa, xóa thì MongoDB chưa thể cập nhật tức khắc vào ổ cứng mà cần 60 giây sau để ghi toàn bộ những thay đổi

dữ liệu từ RAM xuống ổ cứng. Điều này khá rủi ro nếu như trong 60 giây đó có sự cố xảy ra như mất điện sẽ gây mất dữ liệu.

2.4 API và RESTful API

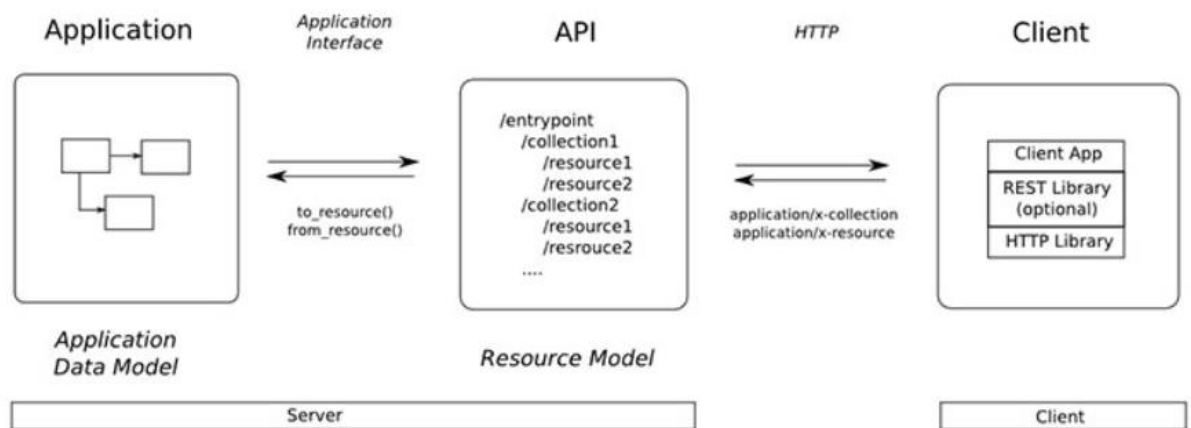
- RESTful API là một tiêu chuẩn dùng trong việc thiết kế API cho các ứng dụng web (thiết kế Web services) để tiện cho việc quản lý các resource. Nó chú trọng vào tài nguyên hệ thống (tệp văn bản, ảnh, âm thanh, video, hoặc dữ liệu động...), bao gồm các trạng thái tài nguyên được định dạng và được truyền tải qua HTTP.



- API (Application Programming Interface) là một tập các quy tắc và cơ chế mà theo đó, một ứng dụng hay một thành phần sẽ tương tác với một ứng dụng hay thành phần khác. API có thể trả về dữ liệu mà bạn cần cho ứng dụng của mình ở những kiểu dữ liệu phổ biến như JSON hay XML.
- REST (REpresentational State** T**ransfer) là một dạng chuyển đổi cấu trúc dữ liệu, một kiểu kiến trúc để viết API. Nó sử dụng phương thức HTTP đơn giản để tạo cho giao tiếp giữa các máy. Vì vậy, thay vì sử dụng một URL cho việc xử lý một số thông tin người dùng, REST gửi một yêu cầu HTTP như GET, POST, DELETE, vv đến một URL để xử lý dữ liệu.
- Chức năng quan trọng nhất của REST là quy định cách sử dụng các HTTP method (như GET, POST, PUT, DELETE...) và cách định dạng các URL cho ứng dụng web để quản các resource. RESTful không quy định logic code ứng dụng và không giới hạn bởi ngôn ngữ lập trình ứng dụng, bất kỳ ngôn ngữ hoặc framework nào cũng có thể sử dụng để thiết kế một RESTful API.
- RESTful API là một tiêu chuẩn dùng trong việc thiết kế các API cho các ứng dụng web để quản lý các resource. RESTful là một trong những kiểu

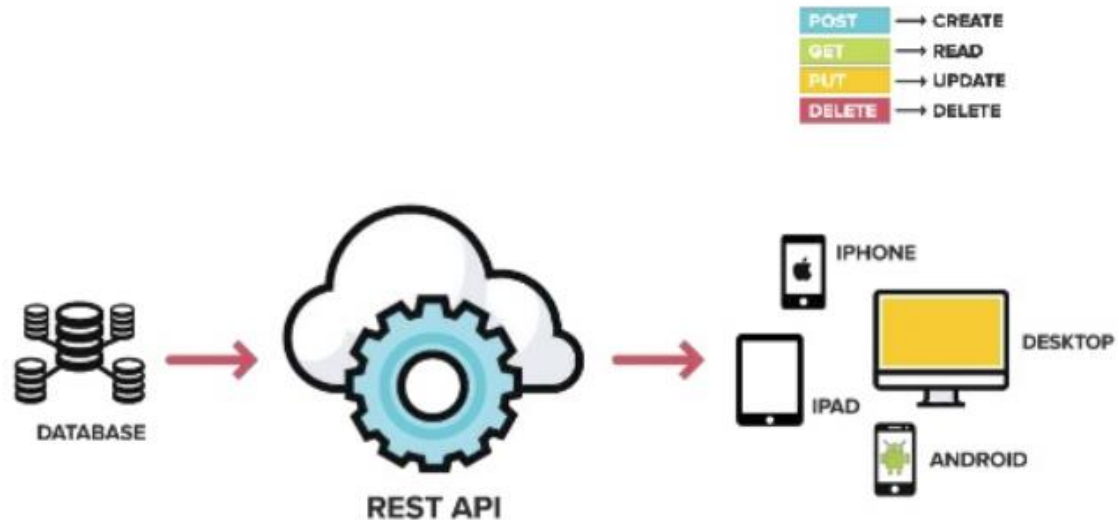
thiết kế API được sử dụng phổ biến ngày nay để cho các ứng dụng (web, mobile...) khác nhau giao tiếp với nhau.

- REST hoạt động chủ yếu dựa vào giao thức HTTP. Các hoạt động cơ bản nêu trên sẽ sử dụng những phương thức HTTP riêng.
 - GET (SELECT): Trả về một Resource hoặc một danh sách Resource.
 - POST (CREATE): Tạo mới một Resource.
 - PUT (UPDATE): Cập nhật thông tin cho Resource.
 - DELETE (DELETE): Xóa một Resource.
- Những phương thức hay hoạt động này thường được gọi là CRUD tương ứng với Create, Read, Update, Delete



- Một số ưu điểm chính khi sử dụng RESTFUL API là:

- Giúp cho ứng dụng rõ ràng hơn
- REST URL đại diện cho resource chứ không phải hành động
- Dữ liệu được trả về với nhiều định dạng khác nhau như: xml, html, json....
- Code đơn giản và ngắn gọn
- REST chú trọng vào tài nguyên của hệ thống



2.5 Mô hình MVC

- MVC là mô hình thiết kế trong ngôn ngữ lập trình Smalltalk-76 được tiến sĩ Trygve Reenskaug trình bày vào năm 1970 tại trung tâm nghiên cứu Xerox Palo Alto (PARC).
- MVC là từ viết tắt được ghép từ 3 thành phần của mô hình là Model – View – Controller. Mỗi thành phần sẽ có một hoạt động riêng biệt và khi kết hợp sẽ tạo thành mô hình thiết kế hoàn chỉnh.
- Thành phần trong mô hình MVC:
 - Model (M)
 - Model chứa một cấu trúc dữ liệu có chức năng lưu trữ toàn bộ các thông tin dữ liệu của một ứng dụng. Trong mô hình MVC thì Model đóng vai trò kết nối cho 2 thành phần View và Controller.
 - Đối với Model được thiết lập như một cơ sở dữ liệu hoặc đơn giản hóa như một file XML thông thường. Khi thiết lập thành phần model thì lập trình viên cần đảm bảo các thao tác với cơ sở dữ liệu như các hoạt động xem, truy xuất hoặc xử lý dữ liệu trong ứng dụng.
 - View (V)
 - View là thành phần liên quan đến giao diện của ứng dụng khi người dùng trải nghiệm. Thông qua dữ liệu của MVC, người dùng sẽ thực hiện các thao tác tìm kiếm, sử dụng thông tin website, ứng dụng.
 - Thành phần View được ứng dụng nhiều trong quá trình lập trình website và đây cũng là nơi mà các thành HTML tạo ra. Chức năng khác của thành phần View này chính là khả năng ghi nhận hành vi của người dùng để tương tác được với Controller.

- Mặc dù không có mối liên hệ trực tiếp với Controller nhưng View sẽ có nhiệm vụ hiển thị yêu cầu chuyển đến cho Controller xử lý thông tin.
- Dễ hình dung hơn thì bạn có thể tham khảo ví dụ khi người dùng nhấn vào nút “Back” hoặc “Trở về” là thành phần view thì người dùng đang tạo ra 1 hành động trên ứng dụng mà controller cần xử lý.
- Controller (C)
 - Controller là bộ phận sẽ xử lý các yêu cầu khi người dùng thao tác trên ứng dụng thông qua thành phần view. Lúc này, Controller sẽ thực hiện truy vấn và xuất dữ liệu phù hợp với yêu cầu của người dùng. Và để làm được điều đó controller còn có cần phải nối được với model để lấy dữ liệu.
- Luồng đi trong mô hình MVC
 - Khi người dùng thực hiện thao tác trên ứng dụng hoặc website thì từ máy Client sẽ gửi yêu cầu đến server (máy chủ). Lúc này, controller sẽ tiến hành tiếp nhận và xử lý yêu cầu. Một vài trường hợp cần truy xuất dữ liệu thì controller sẽ kết nối với Model để hỗ trợ database.
 - Sau khi Controller xử lý xong các yêu cầu thì kết quả sẽ được chuyển về View. Lúc này View sẽ tiến hành tạo các mã HTML để trả về giao diện của trình duyệt kết quả theo yêu cầu của người dùng.
- Ưu điểm
 - Kiểm tra dễ dàng: Các thành phần độc lập giúp người lập trình dễ kiểm soát và khắc phục các vấn đề, lỗi phát sinh trước khi hoàn thiện sản phẩm đến người dùng.
 - Chức năng control: Khi kết hợp với các loại ngôn ngữ lập trình thông dụng như CSS<HTML, Javascript thì mô hình MVC là sự hỗ trợ đóng vai trò tối ưu bộ control trên nền tảng ngôn ngữ lập trình.
 - View và size: MVC giúp tối ưu diện tích băng thông khi sử dụng tránh trường hợp khi nhiều yêu cầu được thực hiện cùng lúc sẽ tạo ra nhiều tệp với dung lượng lớn ảnh hưởng trực tiếp đến đường truyền mạng.
 - Chức năng Soc (Separation of Concern): Cho phép phân loại các thành Model, View, Database, ... để dễ quản lý và kiểm soát hơn.
 - Tính kết hợp: Người lập trình có thể kết hợp mô hình MVC trên nhiều nền tảng website/ ứng dụng khác nhau giúp tiện lợi hơn khi viết code và giảm tải dung lượng.
 - Kết cấu khá đơn giản: Phù hợp cho nhiều đối tượng sử dụng khi có nhu cầu lập trình website hoặc các loại ứng dụng.
- Nhược điểm

- Đối với mô hình MVC có tính phân tách cao giữa các thành phần nên phù hợp để ứng dụng trong các dự án lớn. Nếu ứng dụng MVC trong các dự án nhỏ sẽ dễ gặp tình trạng chồng chéo, tốn nguồn lực khi phát triển dự án. Đồng thời, thời gian trung chuyển dữ liệu cũng là điều cần cân nhắc khi thực hiện dự án nhỏ.
- Tại sao nên sử dụng mô hình MVC
- Mô hình MVC phân tách độc lập các thành phần nên khi xử lý dữ liệu thì phần hiển thị trên ứng dụng không bị thay đổi.
- Nếu mô hình MVC được ứng dụng và phát triển theo hướng chuyên nghiệp thì chúng sẽ giúp cho hoạt động front end, back end hạn chế xung đột trên cùng hệ thống khi thực hiện cùng lúc.
- Mô hình đơn giản, dễ nắm bắt nên người lập trình dễ dàng làm quen và triển khai nhanh chóng hơn.

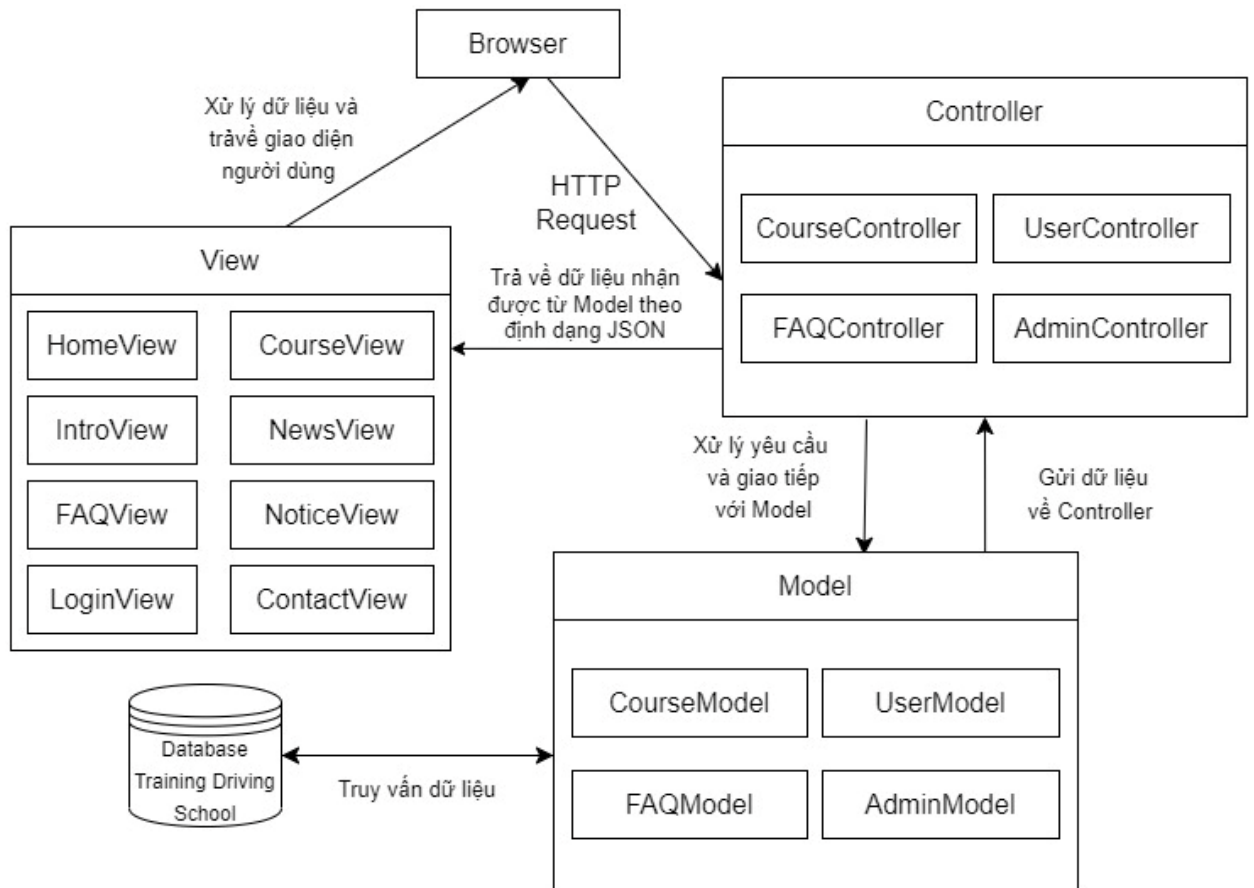
CHƯƠNG 3. THIẾT KẾ VÀ CÀI ĐẶT GIẢI PHÁP

Phần đầu chương 3 thể hiện kiến trúc của hệ thống, phần tiếp theo giải thích các lớp trong mô hình MVC được dùng trong website, các sơ đồ, biểu đồ mô tả cách thức hoạt động của trang web, cuối cùng là phần kết quả thiết kế giao diện và thiết kế chức năng của website.

3.1 Kiến trúc hệ thống

Hệ thống có 4 nhóm người dùng: Quản trị viên, giảng viên, học viên, khách vãng lai. Trong đó:

- Quản trị viên: là tài khoản có phân quyền cao nhất, có vai trò quản lý tất cả các chức năng của hệ thống.
- Giảng viên: là tài khoản có chức năng thêm, sửa, xóa khóa học và học viên, công bố kết quả của khóa học, xem bài viết, xem câu hỏi, trả lời câu hỏi cho học viên, chỉnh sửa thông tin cá nhân.
- Học viên: là tài khoản có chức năng tìm kiếm khóa học, xem khóa học, đăng ký khóa học, thanh toán khóa học, xem bài viết, xem câu hỏi, đặt câu hỏi, xem kết quả khóa học và chỉnh sửa thông tin cá nhân.
- Khách vãng lai: là người dùng chưa có tài khoản, thực hiện được các chức năng như đăng ký tài khoản, xem bài viết, xem câu hỏi, tìm kiếm khóa học, xem khóa học.

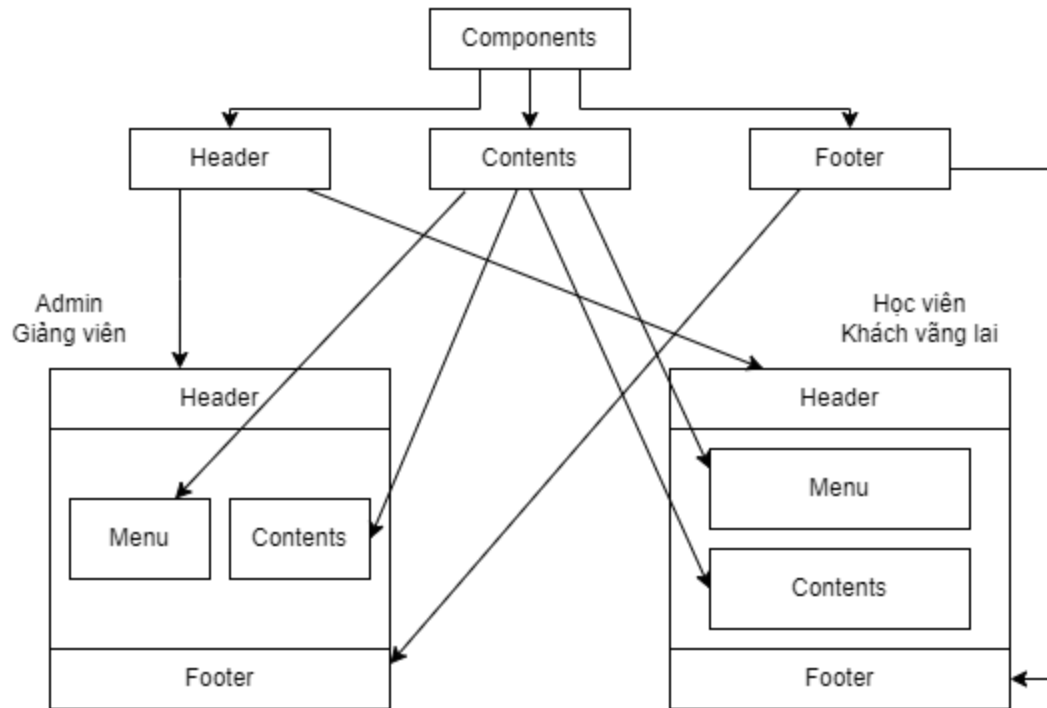


- Client gửi một yêu cầu HTTP đến Controller (ở back-end). Controller nhận yêu cầu và xử lý yêu cầu với logic tương ứng cho yêu cầu cụ thể và giao tiếp với Model (ở back-end). Model sẽ truy vấn đến cơ sở dữ liệu, sau đó nhận dữ liệu về từ cơ sở dữ liệu và gửi dữ liệu về Controller. Sau đó Controller lấy dữ liệu nhận được từ Model và trả về cho View (ở front-end) theo định dạng JSON. View sẽ xử lý dữ liệu và trả về giao diện người dùng cho phía Client.
- Model có các lớp khác nhau để lưu trữ các loại dữ liệu và đối tượng khác nhau có các quy tắc nghiệp vụ để xử lý dữ liệu. Khi nhận dữ liệu yêu cầu từ người dùng Web, dữ liệu được kiểm tra giá trị đầu có hợp lệ hay không, để thao tác xử lý phù hợp.
- Người dùng Web truy cập vào website, khi người dùng Web gửi yêu cầu (request) thì sẽ phải đi qua router, các dữ liệu (data) được truyền xuống Controller để xử lý. Nếu cần dữ liệu thì phải lấy từ Model lên hoặc cập nhật dữ liệu xuống Model, lúc này các kết quả sẽ được gửi ra View cho người dùng Web.
- Đối với website trung tâm đào tạo bằng lái xe sẽ được thiết kế như sau:

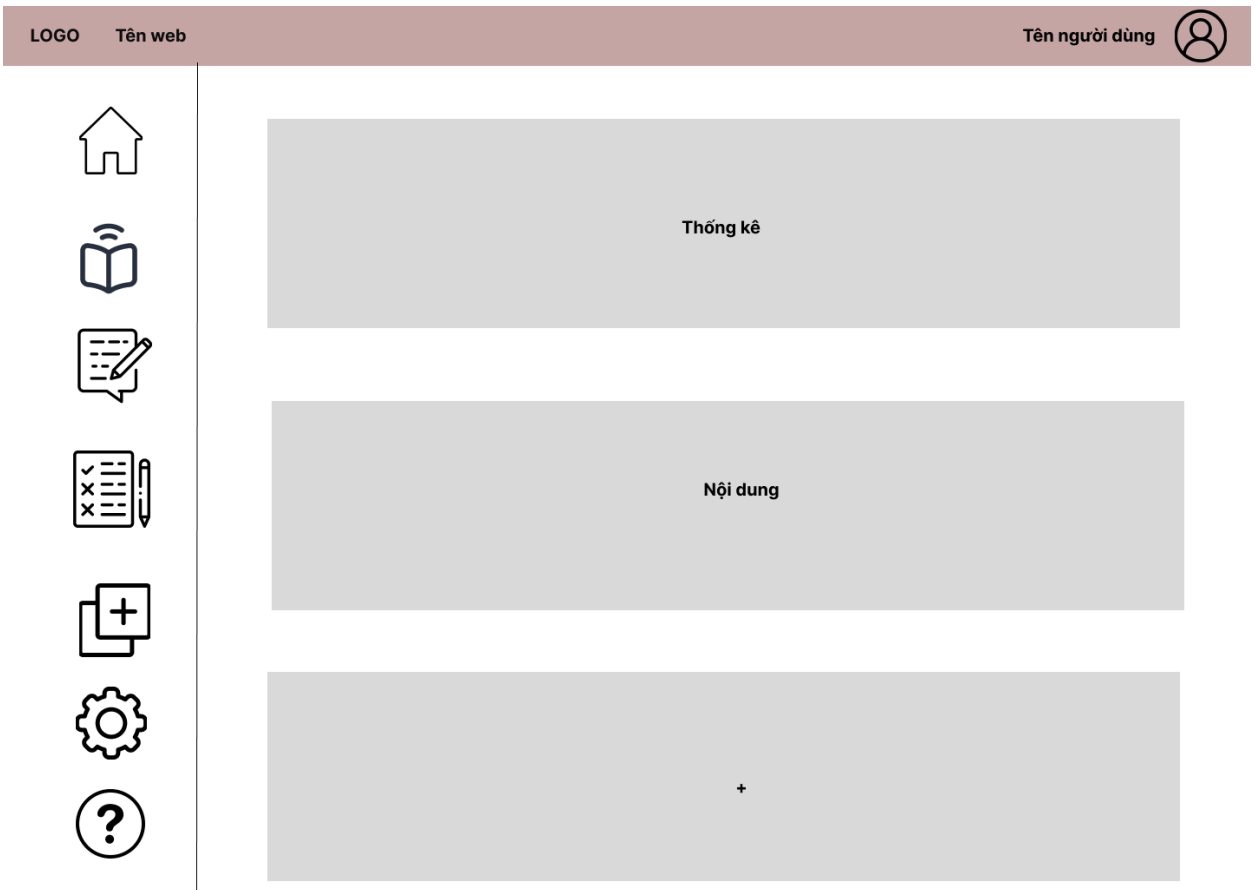
- Giao diện người dùng (UI) sẽ được xây dựng bằng Mantine UI và NextJs.
- API routes xử lý dữ liệu, xác thực ở máy chủ.
- Người dùng gửi 1 yêu cầu HTTP đến API bằng các phương thức GET, POST, PUT, DELETE. Máy chủ tiếp nhận và xử lý yêu cầu bằng cách tương tác với cơ sở dữ liệu, sau đó trả về dữ liệu theo định dạng json. Phía client sẽ xử lý dữ liệu và trả về giao diện người dùng.

Tổng quan các lớp của mô hình MVC:

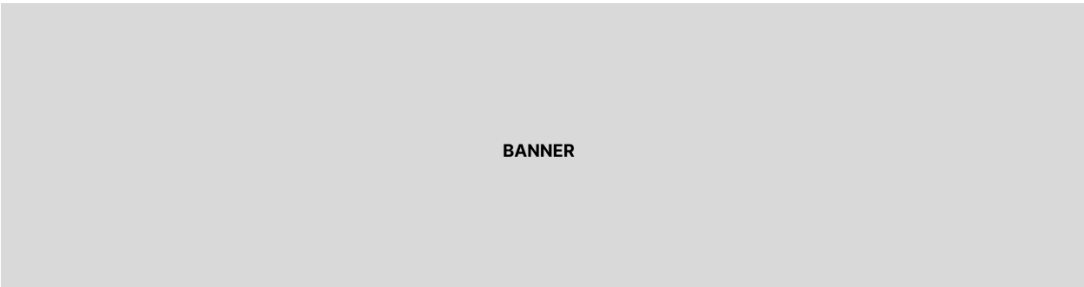
- Lớp Model:
 - Model: là lớp được xây dựng ở phía back-end để thao tác với cơ sở dữ liệu Mongo DB.
 - Đối với website trung tâm đào tạo bằng lái xe thì lớp Model là nơi chịu trách nhiệm lưu trữ, quản lý tất cả các dữ liệu, là nơi truy xuất các thực thể từ cơ sở dữ liệu MongoDB và chứa các logic được thực thi bởi ứng dụng. Nó là cầu nối giữa hai lớp còn lại là View và lớp Controller.
- Lớp View:
 - Lớp View được xây dựng ở phía front-end bằng NextJS và Mantine UI cùng với một số thư viện node modules. Lớp View bao gồm nhiều trang giao diện khác nhau, giao diện sẽ được chia nhỏ thành các component nhằm mục đích để tái sử dụng, tránh lặp lại cùng một logic ở nhiều trang.
 - Lớp View sẽ có 2 kiểu Layout: trang dành cho Học viên và Khách vãng lai, trang dành cho quản trị viên (admin) và giảng viên.
 - Hai loại layout này đều sử dụng chung các component như Header và Footer, ở giữa sẽ là một nội dung chính của trang.



- Lớp View được tách biệt hoàn toàn với lớp Model và lớp Controller và điều này tách các vai trò của lập trình viên. Lập trình viên front-end có thể tập trung vào lập trình giao diện. Bố cục của trang web có thể được thay đổi dễ dàng. Các lập trình viên của front-end và back-end có thể làm việc độc lập giúp rút ngắn thời gian thực thi.
- Giao diện Admin/Giảng viên:



- Giao diện Học viên/Khách vãng lai:



BANNER

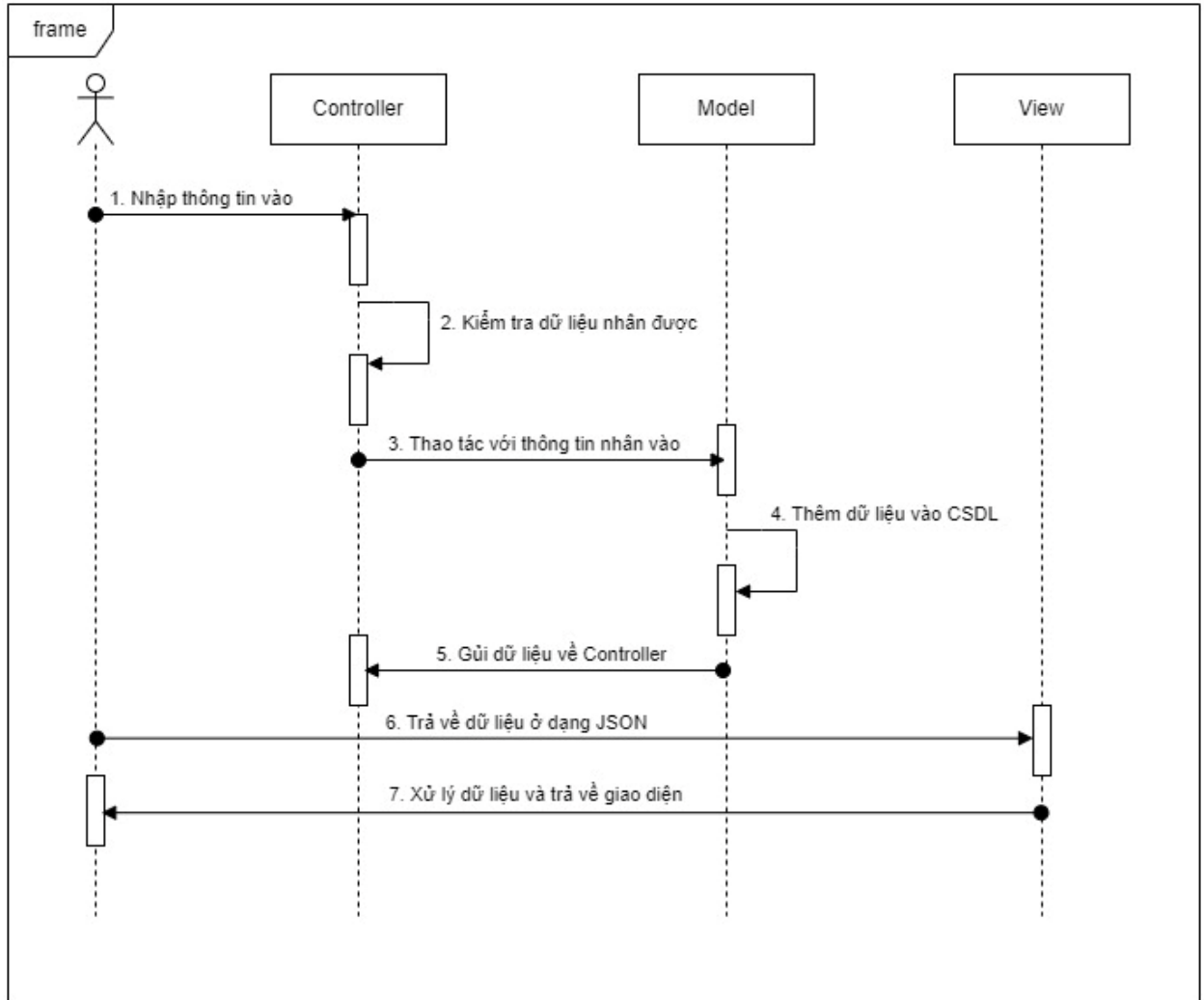


KHÓA HỌC



+

- Lớp Controller:



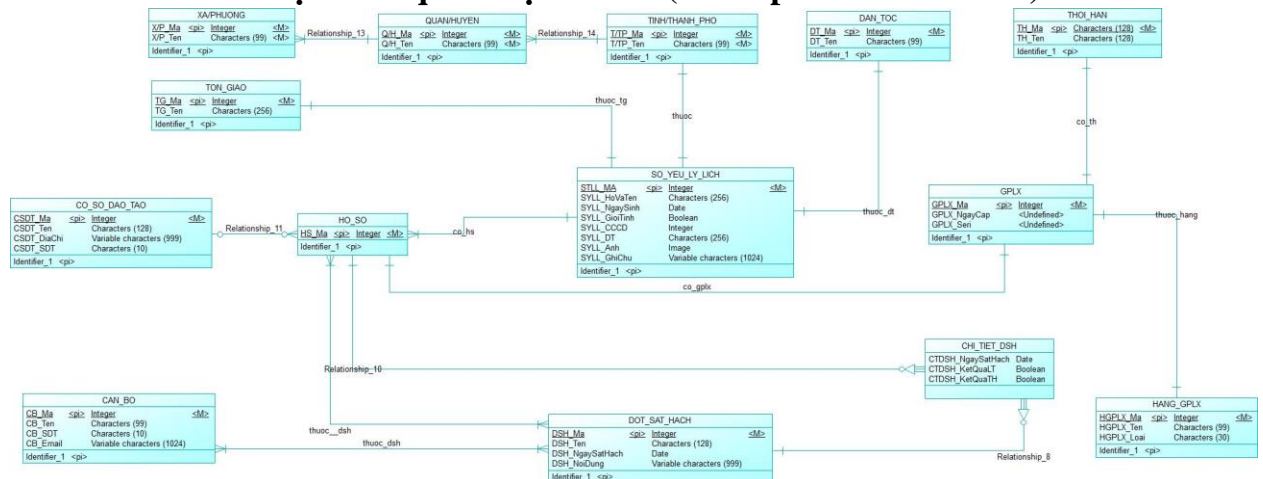
- Controller: là nơi tiếp nhận các yêu cầu xử lý từ phía người dùng, có nhiệm vụ xử lý giữa Model và View tương tác có kết nối với nhau. Controller nhận các yêu cầu (request) từ phía người dùng, sau đó gọi các phương thức để thực hiện các hoạt động được yêu cầu xử lý và gửi ra View. Lúc này View sẽ chịu trách nhiệm format lại các dữ liệu (data) từ phía Controller gửi ra và trình bày các dữ liệu đó theo một định dạng chuẩn HTML. Lớp Controller được xây dựng ở phía back-end cùng với lớp Model. Lớp này nhận các yêu cầu từ Client, xử lý yêu cầu, lấy dữ liệu từ Model và trả về dữ liệu dạng JSON cho lớp View.
- Khi người dùng gửi yêu cầu từ form nhập thông tin Controller sẽ được gọi để xử lý yêu cầu, sau đó thao tác với lớp Model. Các lớp Model này để gọi các hàm để thêm dữ liệu vào sau đó truy vấn dữ liệu gửi về Controller. Controller tiếp tục trả về dữ liệu ở dạng JSON cho lớp View để xử lý và cuối cùng là hiển thị ra giao diện thành công ở phía Client.

Cơ sở thiết kế:

- Đầu tiên, nhắc tới ưu điểm mô hình MVC thì đó là băng thông ([Bandwidth](#)) nhẹ vì không sử dụng viewstate nên khá tiết kiệm băng thông. Việc giảm băng thông giúp website hoạt động ổn định hơn.
- Kiểm tra đơn giản và dễ dàng, kiểm tra lỗi phần mềm trước khi bàn giao lại cho người dùng.
- Một lợi thế chính của MVC là nó tách biệt các phần Model, Controller và View với nhau.
- Sử dụng mô hình MVC chức năng Controller có vai trò quan trọng và tối ưu trên các nền tảng ngôn ngữ khác nhau
- Ta có thể dễ dàng duy trì ứng dụng vì chúng được tách biệt với nhau.
- Có thể chia nhiều developer làm việc cùng một lúc. Công việc của các developer sẽ không ảnh hưởng đến nhau.
- Hỗ trợ [TTD](#) (test-driven development). Chúng ta có thể tạo một ứng dụng với unit test và viết các [won test case](#).
- Phiên bản mới nhất của MVC hỗ trợ trợ thiết kế responsive website mặc định và các mẫu cho mobile. Chúng ta có thể tạo công cụ View của riêng mình với cú pháp đơn giản hơn nhiều so với công cụ truyền thống.

3.2 Thiết kế dữ liệu

3.2.1 Mô hình dữ liệu mức quan niệm CDM (Conceptual Data Model)



3.2.2 Bảng mô tả từ điển dữ liệu

- Bảng NGUOI_DUNG:

ST T	Tên thuộc tính	Kiểu	Kích thước	Khóa chính	Not null	Giá trị mặc nhiên	Chữ số thập phân	Khóa ngoại	Diễn giải
1	ND_Ma	Int		x		1			Mã người dùng
2	ND_Ten	Char	30		X				Tên người dùng
3	ND_SoDienThoai	Int			X				Số điện thoại người dùng
4	ND_NgaySinh	Date			X				Ngày sinh người dùng

- Bảng TAI_KHOAN:

STT	Tên thuộc tính	Kiểu	Kích thước	Khóa chính	Not null	Giá trị mặc nhiên	Chữ số thập phân	Khóa ngoại	Diễn giải
1	TK_Ma	Int		X		1			Mã tài khoản
2	TK_Ten	Char	30		X				Tên tài khoản
3	TK_MatKhau	Varchar	99		X				Mật khẩu tài khoản

- Bảng BAI_VIET:

STT	Tên thuộc	Kiểu	Kích	Khóa	Not	Giá	Chữ	Khóa	Diễn
-----	-----------	------	------	------	-----	-----	-----	------	------

	tính		thước	chính	null	trị mặc nhiên	số thập phân	ngoại	giải
1	BV_Ma	Int		X		1			Mã bài viết
2	BV_MoTa	Char	99						Mô tả về bài viết
3	BV_Ngay	Date & Time			X				Ngày tạo ra bài viết
4	BV_NoiDung	Varchar	999		X				Nội dung của bài viết
5	BV_HinhAnh	Image							Hình ảnh của bài viết

- Bảng KHOA_HOC:

STT	Tên thuộc tính	Kiểu	Kích thước	Khóa chính	Not null	Giá trị mặc nhiên	Chữ số thập phân	Khóa ngoại	Diễn giải
1	KH_Ma	Int		X		1			Mã khóa học
2	KH_Ten	Char	30		X				Tên khóa học

- Bảng CHI_TIET_KHOA_HOC:

ST T	Tên thuộc tính	Kiểu	Kích thước	Khóa chính	Not null	Giá trị mặc nhiên	Chữ số thập phân	Khóa ngoại	Diễn giải
---------	----------------	------	---------------	---------------	-------------	----------------------------	---------------------------	---------------	--------------

						n	phân		
1	CTKH_Ma	Int		X		1			Mã chi tiết khóa học
2	CTKH_MoTa	Char	99						Mô tả về chi tiết khóa học
3	CTKH_Gia	Money			X				Giá của khóa học
4	CTKH_NoiDung	Varchar	999		X				Nội dung chi tiết của khóa học
5	CTKH_HinhAnh/Video	Image							Hình ảnh/video về khóa học

- Bảng PHIEU_THANH_TOAN:

STT	Tên thuộc tính	Kiểu	Kích thước	Khóa chính	Not null	Giá trị mặc nhiên	Chữ số thập phân	Khóa ngoại	Diễn giải
1	PTT_STT	Int		X		1			Số thứ tự của phiếu thanh toán
2	PTT_NgayLap	Date & Time			X				Ngày lập phiếu thanh toán
3	PTT_SoTien	Money			X				Số tiền

									đã thanh toán
--	--	--	--	--	--	--	--	--	---------------------

- Bảng XA/PHUONG:

STT	Tên thuộc tính	Kiểu	Kích thước	Khóa chính	Not null	Giá trị mặc nhiên	Chữ số thập phân	Khóa ngoại	Diễn giải
1	X/P_Ma	Int		X		1			Mã xã/phường
2	X/P_Ten	Char	99		X				Tên xã/phường

- Bảng QUAN/HUYEN:

STT	Tên thuộc tính	Kiểu	Kích thước	Khóa chính	Not null	Giá trị mặc nhiên	Chữ số thập phân	Khóa ngoại	Diễn giải
1	Q/H_Ma	Int		X		1			Mã quận/huyện
2	Q/H_Ten	Char	99		X				Tên quận/huyện

- Bảng TINH/THANHPHO:

STT	Tên thuộc tính	Kiểu	Kích thước	Khóa chính	Not null	Giá trị mặc nhiên	Chữ số thập phân	Khóa ngoại	Diễn giải
1	T/TP_Ma	Int		X		1			Mã tỉnh/thành phố
2	T/TP_Ten	Char	99		X				Tên tỉnh/thành phố

3.3 Thiết kế theo chức năng