

DẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC BÁCH KHOA
KHOA KHOA HỌC VÀ KỸ THUẬT MÁY TÍNH



Bài tập lớn

LẬP TRÌNH WEB

Đề tài:

Trang bán hàng trực tuyến
của Biti's Việt Nam

GVHD: Nguyễn Hữu Hiếu

SV thực hiện:	Võ Hoàng	2113422
	Nguyễn Trường Thản	2114798
	Nguyễn Trần Quang Vũ	2115325
	Lê Minh Chiến	2112933

TP. HỒ CHÍ MINH, THÁNG 5/2024

Mục lục

1	Giới thiệu về tài	3
2	Cơ sở lý thuyết	3
2.1	ReactJS	3
2.2	HTML/CSS (Bootstrap)	5
2.3	Javascript (jQuery)	6
2.4	PHP	6
2.5	MySQL	7
2.6	API	8
2.7	SEO	8
3	Thiết kế ứng dụng	10
3.1	Cơ sở dữ liệu	10
3.1.1	Account	10
3.1.2	Cart	11
3.1.3	Comment	11
3.1.4	Description	11
3.1.5	Discount	11
3.1.6	New	12
3.1.7	Outstanding	12
3.1.8	Product	12
3.1.9	Size	12
3.1.10	Subimage	13
3.2	UseCase cho toàn bộ hệ thống	13
3.3	Các thao tác để thực hiện một số chức năng của Website	15
3.3.1	Đăng nhập	15
3.3.2	Mua hàng	16
3.3.3	Dánh giá về sản phẩm	17
3.3.4	Thêm và chỉnh sửa thông tin sản phẩm trên website của Admin	18
3.4	Cấu trúc mã nguồn	19
3.4.1	Font-end	19
3.4.2	Back-end	20
3.4.3	Mô hình Model View Controller(MCV)	21
4	Các bước nhập cơ sở dữ liệu trên phpMyAdmin.	22
5	Hiện thực	24
5.1	Người dùng khi chưa đăng nhập	24
5.2	Đăng nhập	25
5.3	User sau khi đăng nhập	25
5.4	Tìm kiếm mặt hàng của trang Web	26
5.5	Chi tiết sản phẩm	26
5.6	Giỏ hàng	27
5.7	Dánh giá của người dùng	27
5.8	Liên hệ	28
5.9	Trang quản lý của Admin	28
6	Cách thức cài đặt	28
6.1	Về Front-end	28
6.2	Về Back-end	28





1 Giới thiệu đề tài

Cùng với sự phát triển không ngừng của công nghệ và khoa học, các phương thức mua bán trực tuyến ngày càng thay thế các phương thức truyền thống. Việc này đang diễn ra trong thời đại hiện nay, khi mà chỉ cần một thiết bị kết nối Internet, chúng ta có thể sở hữu bất kỳ món đồ nào mà không cần phải đến trực tiếp đại lý bán hàng. Mua sắm trực tuyến mang lại sự tiện lợi đặc biệt với tính linh hoạt về thời gian và không gian.

Ngày nay, việc mua sắm trực tuyến trở nên càng quan trọng hơn trong cuộc sống hàng ngày của mỗi người. Sự tăng trưởng đáng kể của các ứng dụng thương mại điện tử cũng như lượng truy cập vào chúng trong thời gian qua là minh chứng rõ ràng nhất cho sức hút của lĩnh vực này. Đa dạng và phong phú là những từ có thể được dùng để mô tả các mặt hàng được bán trực tuyến, từ giày dép, quần áo cho đến các sản phẩm điện tử như laptop và điện thoại di động.Thêm vào đó, việc tìm kiếm và tham khảo dịch vụ trực tuyến cũng đang trở nên phổ biến hơn.

Nhóm chúng em đã quyết định thiết kế một trang web bán giày, dép như trang bán hàng trực tuyến Biti's Việt Nam trong khuôn khổ bài tập lớn về lập trình web. Mục tiêu của chúng tôi là tạo ra một nền tảng mua sắm trực tuyến tiện lợi và dễ dàng, giúp người tiêu dùng có thể tìm kiếm và mua được sản phẩm một cách thuận tiện nhất. Đồng thời, trang web cũng giúp doanh nghiệp quản lý kinh doanh của mình một cách hiệu quả và tiếp cận được nhiều đối tượng khách hàng hơn mà không bị giới hạn về thời gian và địa điểm. Trang web sẽ tích hợp các tính năng cần thiết của một trang web thương mại điện tử để tối ưu hóa trải nghiệm mua sắm của người dùng."

2 Cơ sở lý thuyết

Mô tả các thư viện, công nghệ

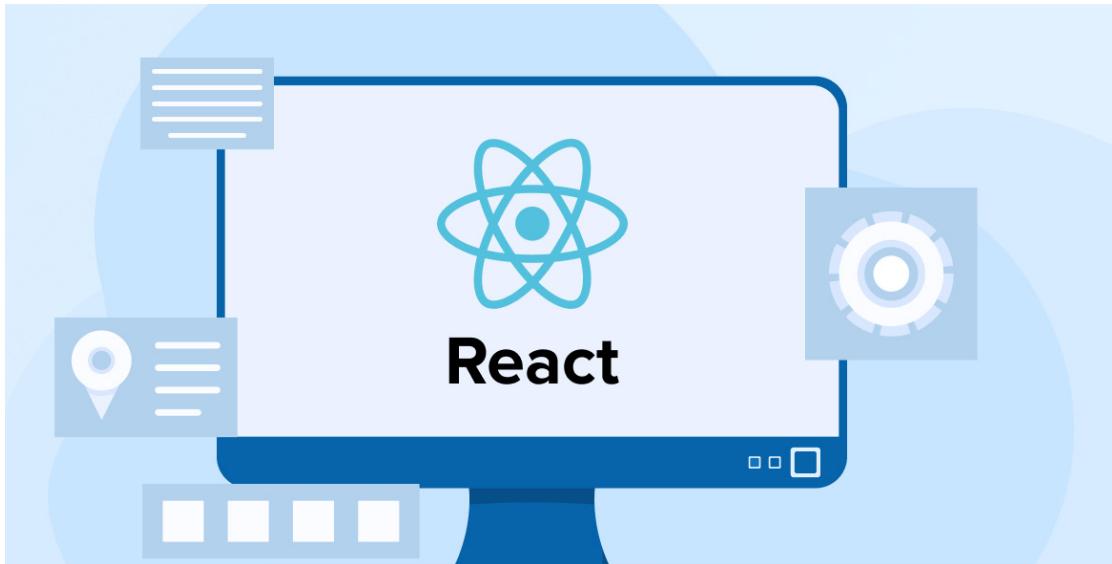
2.1 ReactJS

React, một thư viện JavaScript mạnh mẽ và linh hoạt, được thiết kế để tối ưu hóa quá trình xây dựng giao diện người dùng (UI) thông qua một mô hình lập trình khai báo. Nó cho phép các nhà phát triển tổ chức UI thành các component độc lập, từ đó tạo điều kiện cho việc tái sử dụng và mở rộng chức năng một cách dễ dàng. Mỗi component trong React có thể quản lý trạng thái (state) của mình và cập nhật UI một cách hiệu quả khi có sự thay đổi, giảm thiểu việc re-render không cần thiết và tăng cường hiệu suất ứng dụng.

React được phát triển bởi Meta (trước đây là Facebook) và được duy trì bởi một cộng đồng rộng lớn bao gồm cả các nhà phát triển và công ty cá nhân. Điều này không chỉ đảm bảo rằng thư viện luôn được cập nhật với các tính năng mới và cải tiến hiệu suất, mà còn tạo ra một hệ sinh thái mạnh mẽ với hàng loạt các thư viện và công cụ hỗ trợ.

Với khả năng tương thích cao và cộng đồng hỗ trợ mạnh mẽ, React đã trở thành một trong những lựa chọn hàng đầu cho các nhà phát triển khi xây dựng các giải pháp UI đa dạng và phức tạp. Điều này không chỉ giúp tăng tốc độ phát triển ứng dụng mà còn đảm bảo tính bảo trì và mở rộng trong tương lai.

Lý do nhóm sử dụng ReactJS để thực hiện Frontend



Hình 1: Thư viện React.

- React là Thư Viện, Không Phải Framework: Một trong những điểm mạnh của React là sự linh hoạt. React không áp đặt một cấu trúc dự án cụ thể nào, mà thay vào đó, nó tập trung vào việc cung cấp các công cụ và thư viện để xây dựng giao diện người dùng. Điều này cho phép lập trình viên tự quyết định cách sử dụng và tích hợp React vào dự án một cách linh hoạt, mà không cần phải tuân theo các quy tắc cứng nhắc của một framework.
- Cộng Đồng và Hỗ Trợ Mạnh Mẽ: React có một cộng đồng lớn và năng động, với hàng ngàn người dùng và đóng góp từ khắp nơi trên thế giới. Điều này mang lại lợi ích lớn cho việc học tập, chia sẻ kiến thức và giải quyết vấn đề trong quá trình phát triển ứng dụng. Bạn có thể tìm thấy hàng trăm nghìn tài liệu, bài viết hướng dẫn, và thảo luận trực tuyến về React, cũng như các thư viện và công cụ hỗ trợ từ cộng đồng này.
- Tích Hợp Dễ Dàng: React có khả năng tích hợp dễ dàng vào các ứng dụng web hiện có mà không cần thay đổi quá nhiều mã nguồn. Bạn có thể bắt đầu sử dụng React trong một phần của ứng dụng, và dần dần mở rộng áp dụng nó vào các phần khác mà không gặp nhiều khó khăn.
- Tái Sử Dụng Components: React được xây dựng với một triết lý quan trọng: "Hãy tạo ra mọi thứ như một component." Điều này mang lại lợi ích lớn cho việc duy trì và phát triển mã nguồn. Bằng cách tạo ra các thành phần độc lập và tái sử dụng chúng trong toàn bộ ứng dụng, lập trình viên có thể giảm thiểu lượng mã phải viết lại và đảm bảo tính nhất quán của giao diện người dùng. Việc sử dụng components cũng tạo điều kiện cho việc kiểm thử và debug hiệu quả hơn, vì mỗi thành phần có thể được kiểm tra độc lập.
- Mã Nguồn Mở và Miễn Phí: React là một thư viện mã nguồn mở và miễn phí, cho phép mọi người sử dụng, tùy chỉnh và phát triển nó một cách tự do. Điều này tạo điều kiện cho sự phát triển và sáng tạo, cũng như giúp giảm bớt chi phí phát triển cho các dự án.



Một số hạn chế của React:

- Quản Lý Trạng Thái Phức Tạp: Trong các ứng dụng lớn và phức tạp, việc quản lý trạng thái có thể trở nên phức tạp và khó khăn. Mặc dù có các thư viện như Redux để giúp quản lý trạng thái của ứng dụng, nhưng việc sử dụng chúng đôi khi có thể làm tăng độ phức tạp của mã nguồn. Đặc biệt là khi một ứng dụng phát triển đến quy mô lớn, việc theo dõi và quản lý trạng thái của các component có thể trở nên rối ren và khó khăn.
- Khó Khăn trong Việc Xử Lý Sự Kiện Ràng Buộc: Trong React, việc xử lý sự kiện ràng buộc có thể trở nên phức tạp, đặc biệt là khi xử lý các sự kiện lồng nhau hoặc khi cần phải truyền dữ liệu giữa các component. Mặc dù React cung cấp các cơ chế để xử lý sự kiện, nhưng việc quản lý sự kiện và truyền dữ liệu giữa các component có thể trở nên phức tạp, đặc biệt là trong các ứng dụng lớn và phức tạp.

2.2 HTML/CSS (Bootstrap)

HTML (HyperText Markup Language) là ngôn ngữ đánh dấu cơ bản được sử dụng để xây dựng cấu trúc và nội dung của trang web. Với HTML, chúng ta có thể định nghĩa các phần tử trên trang như đoạn văn bản, hình ảnh, liên kết, biểu mẫu và nhiều thành phần khác. HTML không chỉ giúp xác định các phần tử mà còn cung cấp khung chứa cho thông tin trên trang web. Điều này tạo nền tảng quan trọng để xây dựng trải nghiệm người dùng, cho phép họ tương tác với nội dung trang web một cách thuận tiện và hiệu quả.

CSS (Cascading Style Sheets) là ngôn ngữ được sử dụng để định dạng và trình bày các phần tử HTML. Với CSS, bạn có thể thiết lập các kiểu dáng, màu sắc, kích thước, vị trí và hiệu ứng cho các phần tử trên trang web. Điều này giúp tạo ra giao diện thẩm mỹ, tùy chỉnh và dễ đọc cho người dùng, làm cho trang web trở nên chuyên nghiệp và hấp dẫn hơn. CSS cho phép bạn tạo ra các layout phức tạp, áp dụng hiệu ứng hoạt hình và thậm chí tạo ra các trải nghiệm tương tác động đáng kinh ngạc, giúp nâng cao trải nghiệm người dùng trên web.

Bootstrap là một Framework CSS phổ biến được sử dụng rộng rãi trong phát triển web. Nó cung cấp một bộ các công cụ và giao diện giúp tạo layout và thiết kế web nhanh chóng. Bootstrap có các tính năng như grid system, responsive design, typography, form controls, buttons, navigation và nhiều tính năng khác, giúp lập trình viên dễ dàng xây dựng các trang web chuyên nghiệp và hấp dẫn.

Lợi ích của Bootstrap:

- Tăng Tốc Phát Triển: Bootstrap cung cấp một bộ công cụ và giao diện chuẩn, giúp lập trình viên tiết kiệm thời gian trong việc xây dựng layout và thiết kế web.
- Responsive Design: Bootstrap hỗ trợ responsive design, tức là các trang web được xây dựng bằng Bootstrap có thể tự động điều chỉnh và hiển thị đúng trên các thiết bị khác nhau, từ máy tính đến điện thoại di động.
- Tùy Chỉnh Dễ Dàng: Mặc dù Bootstrap cung cấp các giao diện và tính năng mặc định, nhưng bạn vẫn có thể tùy chỉnh chúng để phù hợp với nhu cầu cụ thể của dự án của mình.
- Tương Thích Tốt: Bootstrap tương thích tốt với hầu hết các trình duyệt web hiện đại, giúp đảm bảo rằng trang web của bạn sẽ hoạt động mượt mà trên mọi nền tảng.

Nhược điểm của Bootstrap:

- Khả Năng Tùy Chỉnh Hạn Chế: Mặc dù Bootstrap cho phép tùy chỉnh, nhưng có những trường hợp mà việc đạt được các thiết kế độc đáo và không phổ biến có thể gặp khó khăn.



- Phụ Thuộc vào Framework: Sử dụng quá nhiều Bootstrap có thể dẫn đến sự phụ thuộc, giới hạn khả năng tùy biến và sáng tạo của bạn. Nếu Bootstrap không được duy trì hoặc phát triển tiếp theo, việc bảo trì và cập nhật cho trang web có thể trở nên khó khăn.

2.3 Javascript (jQuery)

JavaScript là ngôn ngữ lập trình được sử dụng để thêm tính năng tương tác động cho trang web. Nó có khả năng tương tác với các phần tử HTML và CSS, điều khiển các hành vi như thao tác người dùng, thay đổi nội dung động mà không cần tải lại trang web, và giao tiếp với máy chủ bằng AJAX. JavaScript cho phép xây dựng các ứng dụng web phức tạp và có tính tương tác cao mà HTML trước đây chưa thể mang lại được.

jQuery là một thư viện JavaScript phổ biến được sử dụng trong phát triển phía frontend của các ứng dụng web. Với mục tiêu đơn giản hóa việc tương tác với DOM (Document Object Model) và xử lý sự kiện trên trang web, jQuery đã giúp rút ngắn mã nguồn và tối ưu hóa quá trình lập trình trong nhiều năm. jQuery cung cấp cú pháp ngắn gọn để truy cập và thao tác các phần tử trong DOM, cho phép thay đổi nội dung, thuộc tính và kiểu dáng của chúng một cách dễ dàng. Nó cũng tương tác với sự kiện như nhấp chuột, nhập liệu và thao tác trên trang.

Một trong những điểm mạnh của jQuery là tích hợp AJAX, cho phép gửi và nhận dữ liệu từ máy chủ mà không cần phải tải lại trang. Điều này tạo ra các ứng dụng web động và tương tác mượt mà. Tuy nhiên, do sự phát triển của các tiêu chuẩn web và việc sử dụng JavaScript nguyên thuỷ được cải thiện, vai trò của jQuery đã giảm đi một phần. Nhiều tính năng mà trước đây chỉ có thể thực hiện bằng jQuery bây giờ có thể được thực hiện bằng cách sử dụng JavaScript nguyên thuỷ hoặc các thư viện và framework khác.

2.4 PHP

PHP (Hypertext Preprocessor) là một ngôn ngữ lập trình phía server phổ biến và mạnh mẽ, thường được sử dụng để xây dựng các ứng dụng web động và tương tác với cơ sở dữ liệu.

Ưu điểm:

- Dễ học và sử dụng: PHP có cú pháp đơn giản và dễ học, khiến nó trở thành lựa chọn tốt cho người mới bắt đầu phát triển web. Nó tương tự như cú pháp kiểu C và có một cộng đồng lớn các nhà phát triển đóng góp vào tài liệu mở rộng của nó, giúp dễ dàng tìm kiếm tài nguyên và hỗ trợ.
- Cộng đồng và hệ sinh thái lớn: PHP có một cộng đồng các nhà phát triển lớn và tích cực, điều đó có nghĩa là có rất nhiều thư viện, khung và công cụ có sẵn để phát triển PHP. Các framework PHP phổ biến như Laravel, Symfony và CodeIgniter cung cấp các giải pháp mạnh mẽ và giàu tính năng để xây dựng các ứng dụng web một cách hiệu quả.
- Khả năng tương thích đa nền tảng: PHP là ngôn ngữ kịch bản đa nền tảng, có nghĩa là nó có thể chạy trên nhiều hệ điều hành khác nhau, bao gồm Windows, macOS, Linux và nhiều hệ điều hành khác. Điều này làm cho nó rất linh hoạt và cho phép các nhà phát triển triển khai các ứng dụng PHP trên các môi trường lưu trữ khác nhau.
- Tích hợp cơ sở dữ liệu: PHP có khả năng tích hợp mạnh mẽ với cơ sở dữ liệu, bao gồm MySQL, PostgreSQL, SQLite, v.v. Điều này giúp dễ dàng xây dựng các ứng dụng web động với các chức năng dựa trên cơ sở dữ liệu, chẳng hạn như xác thực người dùng, hệ thống quản lý nội dung và giải pháp thương mại điện tử.



- Chu kỳ phát triển nhanh: PHP được biết đến với chu kỳ phát triển nhanh, cho phép các nhà phát triển nhanh chóng xây dựng các ứng dụng web. PHP hỗ trợ nhiều công cụ phát triển web và IDE (Mỗi trường phát triển tích hợp), có thể nâng cao hơn nữa năng suất và tăng tốc quá trình phát triển.

Nhược điểm:

- Hỗ trợ hạn chế cho dữ liệu phi quan hệ: MySQL là một hệ thống quản lý cơ sở dữ liệu quan hệ, có nghĩa là nó không được tối ưu hóa để xử lý dữ liệu phi quan hệ, chẳng hạn như dữ liệu phân cấp hoặc dựa trên tài liệu. Mặc dù MySQL cung cấp hỗ trợ lưu trữ và truy xuất dữ liệu phi quan hệ, nhưng nó có thể không hiệu quả hoặc giàu tính năng như các cơ sở dữ liệu chuyên dụng khác được thiết kế cho dữ liệu phi quan hệ, chẳng hạn như MongoDB hoặc Cassandra.
- Thiếu các tính năng bảo mật nâng cao: Mặc dù MySQL cung cấp các tính năng bảo mật cơ bản như xác thực người dùng và kiểm soát truy cập, nhưng nó có thể thiếu một số tính năng bảo mật nâng cao có sẵn trong các cơ sở dữ liệu hiện đại khác. Ví dụ: các tính năng như mã hóa dữ liệu tự động, kiểm tra tích hợp hoặc kiểm soát truy cập chi tiết có thể yêu cầu cấu hình bổ sung hoặc plugin của bên thứ ba.
- Khó tiếp cận cho các tính năng nâng cao: Mặc dù MySQL rất dễ bắt đầu với các hoạt động cơ bản, nhưng nó có thể có đường cong học tập dốc hơn khi nói đến các tính năng nâng cao như tối ưu hóa hiệu suất, điều chỉnh cơ sở dữ liệu và tối ưu hóa truy vấn nâng cao. Kiến thức chuyên sâu về nội bộ cơ sở dữ liệu và kỹ thuật tối ưu hóa truy vấn có thể được yêu cầu để đạt được hiệu suất và khả năng mở rộng tối ưu.

2.5 MySQL

MySQL là một hệ quản trị cơ sở dữ liệu mã nguồn mở rất phổ biến và được sử dụng rộng rãi trong các ứng dụng web và phần mềm. Dưới đây là một số ưu nhược điểm của MySQL:

Ưu điểm:

- Miễn phí và mã nguồn mở: MySQL là một sản phẩm mã nguồn mở, điều này có nghĩa là bạn có thể sử dụng, tùy chỉnh và phân phối nó mà không cần phải trả bất kỳ phí nào.
- Tốc độ và hiệu suất tốt: MySQL đã được tối ưu hóa để xử lý các truy vấn cơ sở dữ liệu nhanh chóng và hiệu quả. Điều này làm cho nó trở thành một lựa chọn phù hợp cho các ứng dụng có yêu cầu về tốc độ cao.
- Hỗ trợ nhiều loại dữ liệu: MySQL hỗ trợ nhiều loại dữ liệu như số, chuỗi, ngày tháng, hình ảnh và nhiều loại dữ liệu khác. Điều này giúp phù hợp với nhiều loại ứng dụng khác nhau.
- Quản lý đa người dùng và phân quyền: MySQL cung cấp khả năng quản lý người dùng và phân quyền, cho phép bạn kiểm soát quyền truy cập vào dữ liệu và các thao tác trên cơ sở dữ liệu.

Nhược điểm:

- Khả năng mở rộng hạn chế: Mặc dù MySQL có thể xử lý tải lớn, nhưng khi ứng dụng của bạn mở rộng và phải đổi mặt với lượng dữ liệu lớn và truy vấn phức tạp, ta có thể gặp khó khăn về mặt hiệu suất so với các hệ cơ sở dữ liệu chuyên sâu hơn.



- Không hỗ trợ một số tính năng cao cấp: So với một số hệ quản trị cơ sở dữ liệu khác, MySQL không cung cấp một số tính năng cao cấp như khả năng lưu trữ dữ liệu không cấu trúc.
- Bảo mật và quản lý phiên khó khăn: Mặc dù MySQL có các tính năng bảo mật và quản lý phiên, nhưng cần phải cẩn thận khi cấu hình để đảm bảo an toàn tối đa.
- Khi sử dụng phiên bản cũ: Một số phiên bản cũ của MySQL có thể gặp vấn đề về hiệu suất, bảo mật hoặc sự ổn định, vì vậy cần cân nhắc khi chọn phiên bản sử dụng.

2.6 API

REST API (Representational State Transfer Application Programming Interface) là một phương thức chuẩn để giao tiếp giữa backend và frontend của các ứng dụng thông qua mạng. Nó cho phép các ứng dụng khác nhau trao đổi dữ liệu và thực hiện các thao tác trên các tài nguyên từ xa. REST API được xây dựng dựa trên các nguyên tắc của kiến trúc REST, trong đó việc sử dụng các phương thức HTTP để thực hiện các thao tác trên các tài nguyên là trọng tâm. Một REST API thường bao gồm các yếu tố sau:

- Tài nguyên (Resources): Đại diện cho các đối tượng hoặc dữ liệu mà bạn muốn thao tác qua API. Mỗi tài nguyên được định danh bằng một URL duy nhất.
- Phương thức HTTP (HTTP Methods): Các phương thức HTTP như GET, POST, PUT và DELETE được sử dụng để thực hiện các thao tác trên tài nguyên. Ví dụ, GET để truy vấn thông tin, POST để tạo mới, PUT để cập nhật, và DELETE để xóa.
- Dịnh dạng dữ liệu (Data Formats): Dữ liệu được trao đổi thông qua API thường sử dụng các định dạng như JSON (JavaScript Object Notation) hoặc XML (eXtensible Markup Language).
- Quản lý trạng thái (Stateless): REST API là không trạng thái, điều này có nghĩa rằng mọi yêu cầu từ phía client chứa đủ thông tin để server có thể hiểu và xử lý yêu cầu, không cần lưu trạng thái trước đó.

Ví dụ, để truy vấn thông tin về một người dùng từ một REST API, bạn có thể gửi một yêu cầu GET đến URL như sau: <https://api.example.com/users/123>, trong đó 123 là ID của người dùng. Server sẽ trả về dữ liệu người dùng dưới định dạng JSON hoặc XML. Tóm lại, REST API là một phương thức tiêu chuẩn để giao tiếp giữa các ứng dụng thông qua mạng bằng cách sử dụng các phương thức HTTP và định dạng dữ liệu như JSON hoặc XML. Điều này cho phép các ứng dụng tương tác với nhau một cách linh hoạt và hiệu quả.

2.7 SEO

Nhóm đã tìm hiểu và thực hiện các nguyên tắc về SEO để đảm bảo khi deploy trang web luôn được index cao bởi search engine của google. Một số các nguyên tắc mà nhóm đã thực hiện trong quá trình hiện thực bài tập lớn như sau:

- Dùng đúng loại ảnh và không chứa ảnh trong database để đảm tăng tốc độ load trang: Các ảnh của nhóm được lưu trên dịch vụ storage của firebase nên database MySQL của nhóm sẽ chỉ phải lưu dạng chuỗi cho các hình ảnh.
- Các content luôn được đánh heading đúng với các tag h1,h2,h3 và component typography để đảm bảo là người dùng có thể phân cấp được nội dung và chú ý vào mục họ tìm kiếm nhanh hơn.



- Các URL dùng để navigate trong page luôn được chú ý đặt để trở nên ngắn và dễ hiểu nhất có thể đối với người dùng.

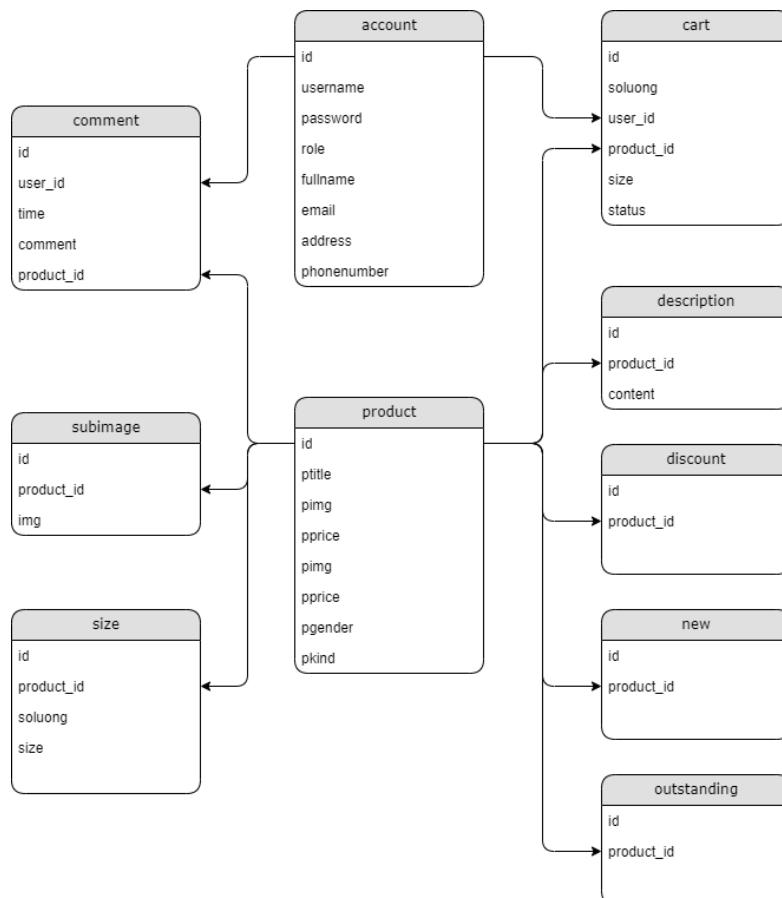
Việc tối ưu hóa trang web có thể bao gồm những công đoạn sắp xếp lại những nội dung của trang web sao cho hợp lý, giúp các công cụ tìm kiếm dễ dàng index; hoặc bao gồm việc tối ưu tốc độ tải trang web xuống mức tối thiểu, làm cho trang web tốn ít tài nguyên để index, nhờ đó tăng khả năng trang web được các công cụ tìm kiếm đưa lên kết quả.

Tuy nhiên, việc tối ưu hóa không có nghĩa là trang web của chúng ta chắc chắn sẽ được đưa lên kết quả trên các trang công cụ tìm kiếm. Ngoài ra, việc tối ưu là một công đoạn rất tốn thời gian và nguồn lực, đòi hỏi người làm web phải hiểu biết sâu về hệ thống cũng như có nội dung chất lượng ngay từ ban đầu.

3 Thiết kế ứng dụng

3.1 Cơ sở dữ liệu

Cơ sở dữ liệu của trang Web bao gồm 10 bảng dữ liệu: account, cart, comment, description, discount, new, outstanding, product, size, subimage.



Hình 2: Cơ sở dữ liệu

3.1.1 Account

Bảng dữ liệu account lưu trữ thông tin đăng ký tài khoản từ người dùng hoặc quản trị viên bao gồm các trường:

- **id:** Địa chỉ Id của tài khoản, kiểu integer và có độ dài nhỏ hơn hoặc bằng 50 kí tự.
- **username:** Tên đăng nhập, kiểu varchar và có độ dài nhỏ hơn hoặc bằng 50 kí tự.
- **password:** Mật khẩu đăng nhập, kiểu varchar và có độ dài nhỏ hơn hoặc bằng 50 kí tự.
- **role:** Mô tả vai trò của tài khoản, có thể là người dùng (user) hoặc quản trị viên (admin), được dùng để phân quyền, có độ dài nhỏ hơn hoặc bằng 50 kí tự.



3.1.2 Cart

Bảng dữ liệu cart lưu trữ thông tin giỏ hàng của sản phẩm bao gồm các trường :

- **id:** Địa chỉ id của giỏ hàng, tăng lên mỗi khi có một sản phẩm được thêm vào giỏ hàng, kiểu integer và có độ dài nhỏ hơn hoặc bằng 50 kí tự.
- **soluong:** Số lượng sản phẩm mà người dùng muốn mua đối với một sản phẩm cụ thể, kiểu integer và có độ dài nhỏ hơn hoặc bằng 50 kí tự.
- **soluong:** Số lượng sản phẩm mà người dùng muốn mua đối với một sản phẩm cụ thể, kiểu integer và có độ dài nhỏ hơn hoặc bằng 50 kí tự.
- **user_id:** id của người mua hàng, kiểu integer và có độ dài nhỏ hơn hoặc bằng 50 kí tự.
- **product_id:** id của sản phẩm được thêm vào giỏ hàng, kiểu integer và có độ dài nhỏ hơn hoặc bằng 50 kí tự.
- **size:** kích cỡ của sản phẩm, kiểu integer và có độ dài nhỏ hơn hoặc bằng 50 kí tự.
- **status:** Trạng thái của sản phẩm, kiểu varchar và có độ dài nhỏ hơn hoặc bằng 250 kí tự.

3.1.3 Comment

Bảng dữ liệu comment lưu giữ thông tin bình luận các sản phẩm bao gồm các trường:

- **id:** Khóa chính của bảng comment, kiểu interger và có độ dài nhỏ hơn hoặc bằng 50 kí tự.
- **user_id:** id của người dùng, kiểu integer và có độ dài nhỏ hơn hoặc bằng 50 kí tự.
- **time:** thời điểm mà người dùng viết bình luận, kiểu date.
- **comment:** nội dung bình luận, kiểu varchar và có độ dài nhỏ hơn hoặc bằng 250 kí tự.
- **product_id:** id của sản phẩm được bình luận, kiểu integer và có độ dài nhỏ hơn hoặc bằng 50 kí tự.

3.1.4 Description

Bảng dữ liệu description lưu trữ thông tin mô tả sản phẩm từ người bán hàng bao gồm các trường :

- **id:** Khóa chính của bảng description, kiểu interger và có độ dài nhỏ hơn hoặc bằng 11 kí tự.
- **product_id:** id của sản phẩm được mô tả, kiểu integer và có độ dài không vượt quá 11 kí tự.
- **content:** nội dung mô tả sản phẩm, kiểu longtext.

3.1.5 Discount

Bảng dữ liệu discount lưu trữ thông tin các sản phẩm đang được giảm giá, trong phạm vi bài tập lớn các sản phẩm có sẵn trong product đều được sử dụng lại, giả sử như tất cả các sản phẩm trong bảng product đều là sản phẩm giảm giá.

- **id:** Khóa chính của bảng discount, kiểu interger và có độ dài nhỏ hơn hoặc bằng 11 kí tự.
- **product_id:** id của sản phẩm được sử dụng lại, kiểu integer và có độ dài không vượt quá 11 kí tự.



3.1.6 New

Bảng dữ liệu new lưu trữ thông tin các sản phẩm mới nhất, trong phạm vi bài tập lớn các sản phẩm có sẵn trong product đều được sử dụng lại, giả sử như tất cả các sản phẩm trong bảng product đều là sản phẩm mới.

- **id:** Khóa chính của bảng new, kiểu interger và có độ dài nhỏ hơn hoặc bằng 11 kí tự.
- **product_id:** id của sản phẩm được sử dụng lại, kiểu integer và có độ dài không vượt quá 11 kí tự.

3.1.7 Outstanding

Bảng dữ liệu outstanding lưu trữ thông tin các sản phẩm nổi bật nhất, trong phạm vi bài tập lớn các sản phẩm có sẵn trong product đều được sử dụng lại, giả sử như tất cả các sản phẩm trong bảng product đều là sản phẩm nổi bật.

- **id:** Khóa chính của bảng outstanding, kiểu interger và có độ dài nhỏ hơn hoặc bằng 11 kí tự.
- **product_id:** id của sản phẩm được sử dụng lại, kiểu integer và có độ dài không vượt quá 11 kí tự.

3.1.8 Product

Bảng dữ liệu product lưu trữ thông tin sản phẩm được rao bán trên trang web :

- **id:** Địa chỉ id của sản phẩm, kiểu integer và có độ dài nhỏ hơn hoặc bằng 50 kí tự.
- **ptitle:** Tên của sản phẩm, kiểu varchar và có độ dài nhỏ hơn hoặc bằng 250 kí tự.
- **pprice:** Giá của sản phẩm, kiểu varchar và có độ dài nhỏ hơn hoặc bằng 250 kí tự.
- **pkind:** Loại sản phẩm, kiểu varchar và có độ dài nhỏ hơn hoặc bằng 100 kí tự.
- **pimg:** Link hình ảnh của các sản phẩm, kiểu varchar và có độ dài nhỏ hơn hoặc bằng 250 kí tự.
- **pgender:** Giới tính phù hợp với sản phẩm, kiểu varchar và có độ dài nhỏ hơn hoặc bằng 100 kí tự.

3.1.9 Size

Bảng dữ liệu size lưu trữ thông tin về kích cỡ của sản phẩm giày:

- **id:** Khóa chính của size, kiểu integer và có độ dài nhỏ hơn hoặc bằng 11 kí tự.
- **product_id:** id của sản phẩm, kiểu integer và có độ dài nhỏ hơn hoặc bằng 11 kí tự.
- **soluong:** số lượng sản phẩm có kích cỡ tương ứng, kiểu integer và có độ dài nhỏ hơn hoặc bằng 11 kí tự .
- **size:** Kích cỡ của sản phẩm, kiểu integer và có độ dài nhỏ hơn hoặc bằng 11 kí tự.



3.1.10 Subimage

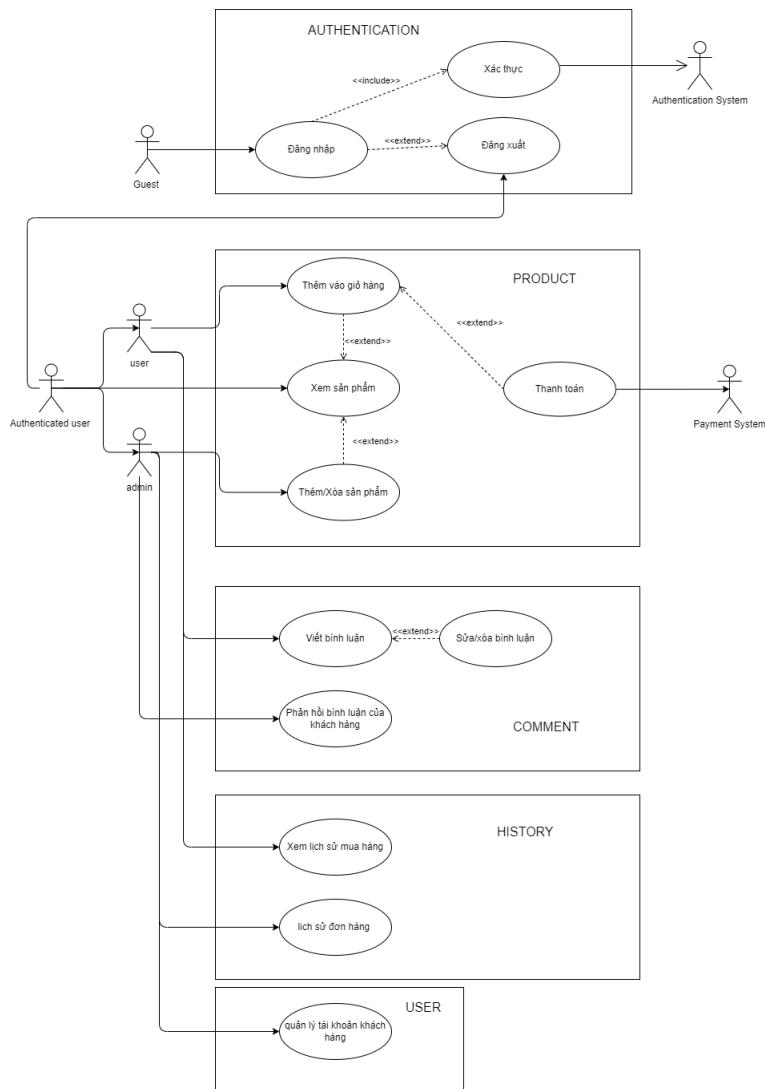
Bảng dữ liệu subimage lưu trữ thông tin về các hình ảnh có liên quan đến sản phẩm :

- **id:** Khóa chính của bảng subimage, kiểu integer và có độ dài nhỏ hơn hoặc bằng 11 kí tự.
- **product_id:** id của sản phẩm, kiểu integer và có độ dài nhỏ hơn hoặc bằng 11 kí tự.
- **img:** Lưu trữ đường link dẫn đến các hình ảnh mô tả chi tiết hơn cho sản phẩm có product_id tương ứng, kiểu varchar và có tối đa là 255 kí tự.

3.2 UseCase cho toàn bộ hệ thống

UseCase mô tả sự tương tác giữa người dùng và hệ thống. Thông thường có 2 cách phổ biến:

- Cách thức mà người dùng tương tác với hệ thống.
- Cách thức mà hệ thống tương tác với các hệ thống khác.



Hình 3: UseCase cho toàn bộ hệ thống

Mô tả ngắn gọn:

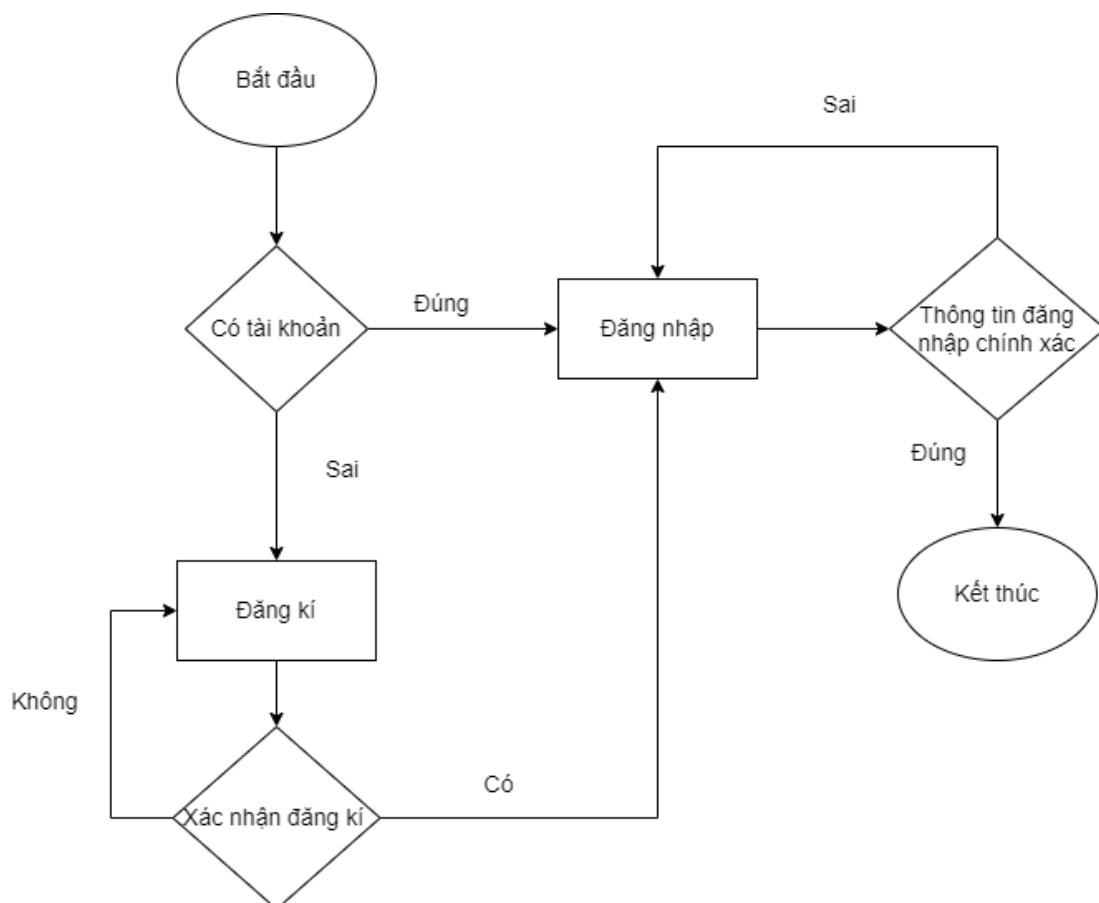
- **Guest:** Chỉ được quyền đăng nhập vào Website.
- **Admin** bao gồm những quyền sau:
 - Xem và chỉnh sửa thông tin của trang Web.
 - Xem và chỉnh sửa thông tin của sản phẩm.
 - Xem thống kê mua hàng thông qua lịch sử đơn hàng.
 - Quản lý thông tin và danh sách các khách hàng.
 - Phản hồi các bình luận của khách hàng.
 - Đăng xuất khỏi hệ thống.

- Khách hàng bao gồm những quyền sau:

- Xem và lựa chọn sản phẩm của trang Web.
- Thêm xóa sản phẩm khỏi gio hang và xem lịch sử mua hàng.
- Thanh toán đơn hàng.
- Tạo bình luận cho các sản phẩm của Shop.
- Đăng xuất khỏi hệ thống.

3.3 Các thao tác để thực hiện một số chức năng của Website

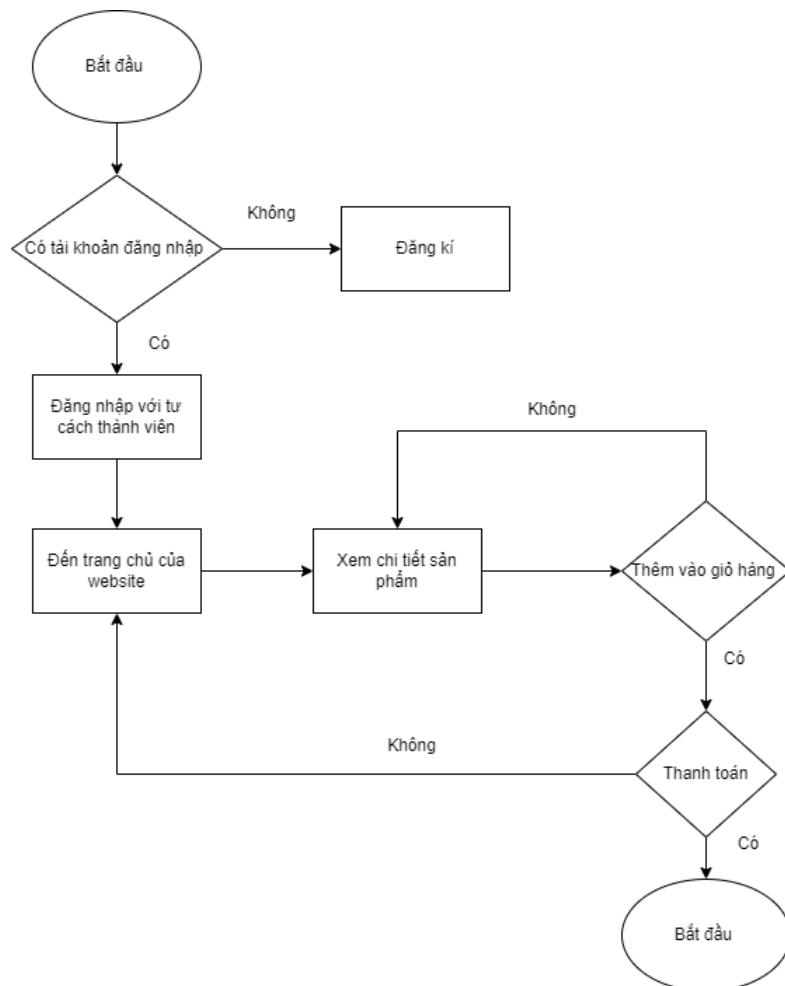
3.3.1 Đăng nhập



Hình 4: Quy trình đăng nhập tài khoản

Mô tả ngắn gọn: Khi người dùng truy cập vào Website, Để sử dụng được những dịch vụ của trang Web người dùng sẽ phải cần dùng một tài khoản đăng nhập. Nếu chưa có tài khoản người dùng sẽ phải đăng kí và được sự xác nhận từ phía admin. Trường hợp người dùng đã đăng kí thì chỉ cần tiến hành đăng nhập là sẽ hoàn tất.

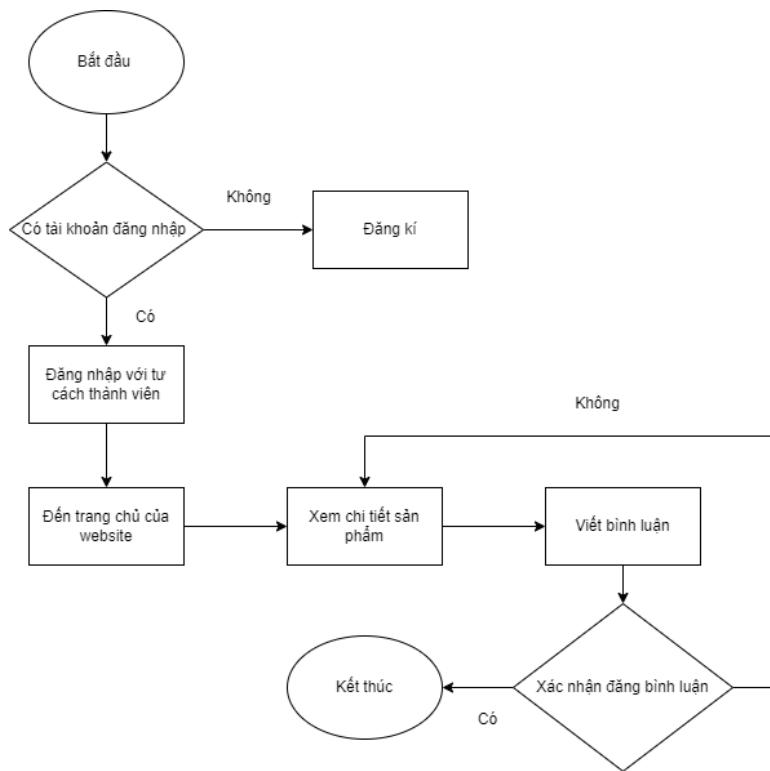
3.3.2 Mua hàng



Hình 5: Quy trình mua hàng

Mô tả ngắn gọn: Khi người dùng muốn mua mặt hàng của Shop, người dùng cần đăng nhập với vai trò là thành viên mua hàng, sau đó vào Trang chủ của sản phẩm, chọn sản phẩm cần mua rồi tiến hành thêm vào giỏ hàng. Ở trang giỏ hàng khi tiến hành thanh toán, Website sẽ dẫn người dùng đến trang thanh toán sau đó hoàn tất việc mua hàng. Nếu khách hàng chưa muốn thanh toán thì có thể quay lại Website để tìm thêm các sản phẩm dành cho người dùng.

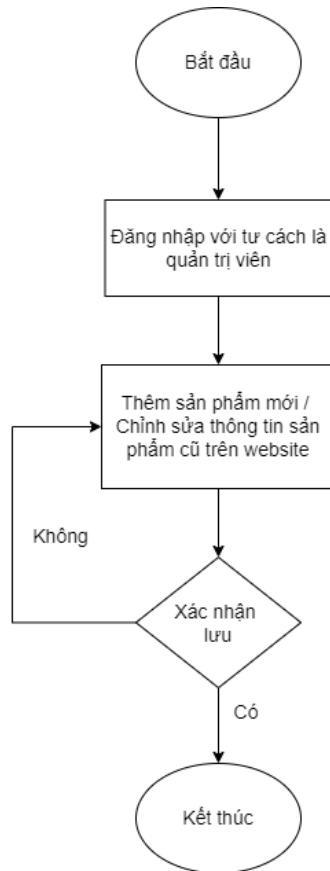
3.3.3 Đánh giá về sản phẩm



Hình 6: Quy trình thực hiện đánh giá sản phẩm

Mô tả ngắn gọn: Khi muốn để lại đánh giá về một loạt mặt hàng sản phẩm, người dùng cần đăng nhập với vai trò là một người tiêu dùng, vào Trang chủ của sản phẩm sau đó chọn chi tiết sản phẩm mà người dùng muốn để lại đánh giá. Chọn thêm bình luận sau đó nhập những thông tin mà khách hàng muốn nhận xét về sản phẩm. Chọn xác nhận để hoàn tất việc đánh giá sản phẩm.

3.3.4 Thêm và chỉnh sửa thông tin sản phẩm trên website của Admin



Hình 7: Quy trình thực hiện đăng 1 bài báo

Mô tả ngắn gọn: Khi Website cần cập nhật thêm một số sản phẩm mới hoặc chỉnh sửa thông tin của một số sản phẩm đã có trên trang Web, cần đăng nhập vào trang web với tư cách là một Admin để có thể chỉnh sửa. Cuối cùng chỉ cần xác nhận lưu thay đổi là hoàn tất việc chỉnh sửa thông tin sản phẩm.



3.4 Cấu trúc mã nguồn

3.4.1 Font-end

Source Code Front-End gồm các folder chính sau:

```
└─ front_end
    ├─ node_modules
    ├─ public
    └─ src
        ├─ assets
        ├─ components
        ├─ context
        ├─ pages
        ├─ redux
        ├─ services
        ├─ App.js
        ├─ index.css
        ├─ index.js
        └─ .gitignore
            └─ package-lock.json
            └─ package.json
            └─ README.md
            └─ bitis.sql
            └─ huong_dan_lay_product.txt
            └─ README.md

        > OUTLINE
        > TIMELINE
```

Hình 8: File code Front-End

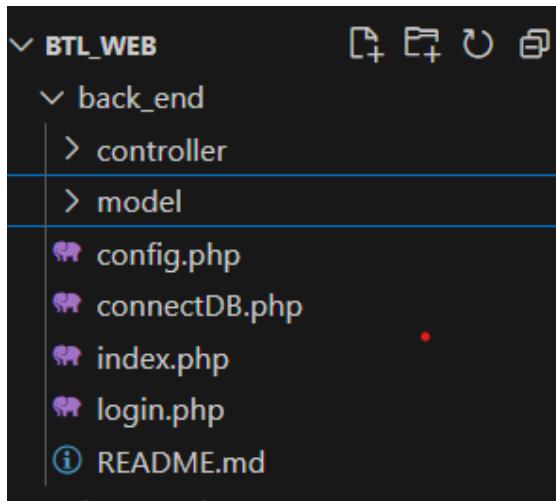
- **assets:** Chứa ảnh có liên quan và logo của website.
- **components:** Chứa các thành phần chung có thể tái sử dụng, thường là các thành phần chung giữa các trang như header, footer,...



- **context:** thư mục chứa các context được dùng để chia sẻ dữ liệu giữa các thành phần mà không cần phải truyền qua nhiều cấp độ của cây component.
- **pages:** chứa các trang của web được chia nhỏ hơn để dễ quản lý và bảo trì
- **service:** sẽ chứa các hàm trực tiếp thao tác với backend để lấy, thay đổi dữ liệu.

3.4.2 Back-end

Source Code Back-End gồm các folder chính sau:

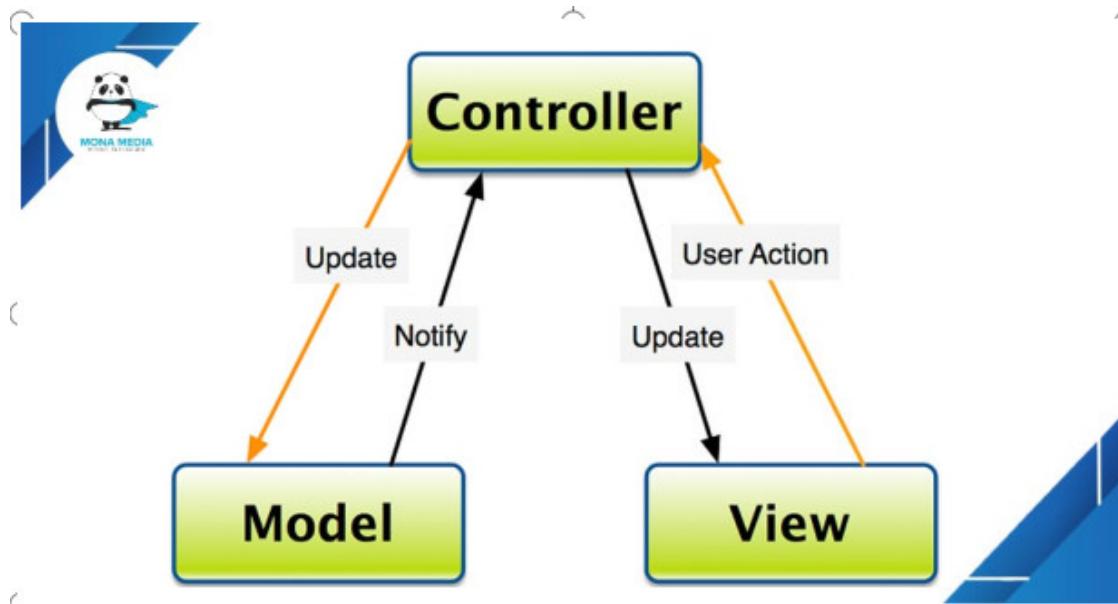


Hình 9: File code Back-End

- **controller:** là thành phần xử lý tương tác của người dùng, nhận dữ liệu từ Model, xử lý và chuyển tiếp thông tin đến View. Nó hoạt động như một trung gian giữa Model và View. Controller quản lý luồng dữ liệu trong ứng dụng, xử lý logic nghiệp vụ, và điều hướng. Nó nhận yêu cầu từ người dùng, xử lý yêu cầu đó (thường thông qua Model), và trả về phản hồi thích hợp.
- **model:** đại diện cho cấu trúc dữ liệu của ứng dụng, thường tương tác trực tiếp với cơ sở dữ liệu. Nó chứa các logic liên quan đến dữ liệu: cách truy xuất, lưu trữ và xác thực dữ liệu. Model chịu trách nhiệm cho việc lấy dữ liệu, xử lý và gửi trả dữ liệu đến Controller. Nó cũng có thể chứa logic nghiệp vụ như tính toán hoặc kiểm tra.
- **index.js:** giống như một router đóng vai trò quan trọng trong việc điều hướng và kết nối giữa Controller và View. Nó quản lý các yêu cầu đến từ client (thường là thông qua URL) và quyết định Controller nào sẽ xử lý yêu cầu đó, phân tích url và các thông tin yêu cầu khác (như phương thức HTTP) để xác định logic nào sẽ được thực thi. Nó có thể gọi một hàm trong Controller cụ thể, kích hoạt một chuỗi các sự kiện để xử lý yêu cầu.
- **file config:** định nghĩa một số biến liên quan đến tên của server, username, password cung cấp như tên của database. Khi thay đổi điều gì liên quan đến database, ta chỉ cần thay đổi giá trị của biến liên quan bên trong file này.

- **file connectDB:** dùng để kết nối tới database và được dùng chung giữa nhiều file, tránh lặp lại việc kết nối tới database nhiều lần.
- **file login:** được viết riêng để xử lý việc đăng nhập của khách hàng.

3.4.3 Mô hình Model View Controller(MCV)



Hình 10: Mô hình MVC

Về phía backend ta thực hiện theo mô hình MVC (Model – View – Controller). Đây là mô hình thiết kế sử dụng trong kỹ thuật phần mềm. Mô hình source code thành 3 phần, tương ứng mỗi từ. Mỗi từ tương ứng với một hoạt động tách biệt trong một mô hình.

Model(M): Là bộ phận có chức năng lưu trữ toàn bộ dữ liệu của ứng dụng. Bộ phận này là một cầu nối giữa 2 thành phần bên dưới là View và Controller. Model thể hiện dưới hình thức là một cơ sở dữ liệu hoặc có khi chỉ đơn giản là một file XML bình thường. Model thể hiện rõ các thao tác với cơ sở dữ liệu như cho phép xem, truy xuất, xử lý dữ liệu,...

View (V): Đây là phần giao diện (theme) dành cho người sử dụng. Nơi mà người dùng có thể lấy được thông tin dữ liệu của MVC thông qua các thao tác truy vấn như tìm kiếm hoặc sử dụng thông qua các website. Thông thường, các ứng dụng web sử dụng MVC View như một phần của hệ thống, nơi các thành phần HTML được tạo ra. Bên cạnh đó, View cũng có chức năng ghi nhận hoạt động của người dùng để tương tác với Controller. Tuy nhiên, View không có mối quan hệ trực tiếp với Controller, cũng không được lấy dữ liệu từ Controller mà chỉ hiển thị yêu cầu chuyển cho Controller mà thôi. Ví dụ: Nút “delete” được tạo bởi View khi người dùng nhấn vào nút đó sẽ có một hành động trong Controller.

Controller (C): Bộ phận có nhiệm vụ xử lý các yêu cầu người dùng đưa đến thông qua view. Từ đó, C đưa ra dữ liệu phù hợp với người dùng. Bên cạnh đó, Controller còn có chức năng



kết nối với model.

Một ví dụ về việc sử dụng mô hình MVC cho Website.
Khi hiển thị trang web, mô hình MVC sẽ hoạt động như sau :

- **View:** Ngay lúc trang web được load, view sẽ gọi controller tương ứng để lấy danh sách sản phẩm.
- **Controller:** Lúc này sẽ nhận dữ liệu từ phía view, trích xuất những thông số cần thiết và gửi những tham số này đến cho model.
- **Model:** Lúc này sẽ nhận dữ liệu từ Controller và lấy danh sách sản phẩm dựa trên tham số được truyền vào, sau đó trả về cho view thông qua controller.

Kết luận: Mỗi bộ phận thực hiện chức năng nhất định, nhưng chúng có sự thống nhất, liên kết với nhau tạo nên mô hình MVC. Mô hình này tương đối nhẹ. Nó có thể tích hợp được nhiều tính năng có trong ASP.NET hiện giờ. Ví dụ như authentication (quá trình xác thực).

4 Các bước nhập cơ sở dữ liệu trên phpMyAdmin.

- Tiến hành mở Hệ cơ sở dữ liệu XAMPP vào Admin để mở hệ cơ sở dữ liệu phpMyAdmin rồi tạo một hệ cơ sở dữ liệu mới.

The screenshot shows the 'Databases' section of the phpMyAdmin interface. At the top, there is a 'Create database' button with a plus sign icon. Below it, a text input field contains the name 'bitis'. To the right of the input field is a dropdown menu set to 'utf8mb4_general_ci'. Next to the dropdown is a 'Create' button. Below these controls are two buttons: 'Check all' and 'Drop'. The main area displays a table with three columns: 'Database', 'Collation', and 'Action'. The table lists several databases: 'bitis1', 'information_schema', 'mysql', 'performance_schema', 'phpmyadmin', 'shop', and 'test'. Each database entry includes its collation and a 'Check privileges' link under the 'Action' column. At the bottom of the table, it says 'Total: 7'.

Hình 11: Tạo hệ cơ sở dữ liệu

- Chọn file sql để tiến hành import vào hệ cơ sở dữ liệu.



Importing into the database "bitis"

File to import:

File may be compressed (gzip, bzip2, zip) or uncompressed.
A compressed file's name must end in **.[format].[compression]**. Example: **.sql.zip**

Browse your computer: (Max: 40MiB)

Choose File You may also drag and drop a file on any page.

Character set of the file:

utf-8

Partial import:

Allow the interruption of an import in case the script detects it is close to the PHP timeout limit.
This might be a good way to import large files, however it can break transactions.

Skip this number of queries (for SQL) starting from the first one:

0

Hình 12: Import file sql vào hệ cơ sở dữ liệu

- Sau khi import vào hệ cơ sở dữ liệu, Trang web của chúng tôi bao gồm 10 bảng.

Table	Action	Rows	Type	Collation	Size	Overhead
account		5	InnoDB	utf8mb4_general_ci	16.0 KiB	-
cart		3	InnoDB	utf8mb4_general_ci	48.0 KiB	-
comment		3	InnoDB	utf8mb4_general_ci	48.0 KiB	-
description		28	InnoDB	utf8mb4_general_ci	64.0 KiB	-
discount		24	InnoDB	utf8mb4_general_ci	16.0 KiB	-
new		24	InnoDB	utf8mb4_general_ci	16.0 KiB	-
orders		0	InnoDB	utf8mb4_general_ci	32.0 KiB	-
outstanding		24	InnoDB	utf8mb4_general_ci	16.0 KiB	-
product		27	InnoDB	utf8mb4_general_ci	16.0 KiB	-
size		146	InnoDB	utf8mb4_general_ci	16.0 KiB	-
subimage		201	InnoDB	utf8mb4_general_ci	64.0 KiB	-
11 tables	Sum	485	InnoDB	utf8mb4_general_ci	352.0 KiB	0 B

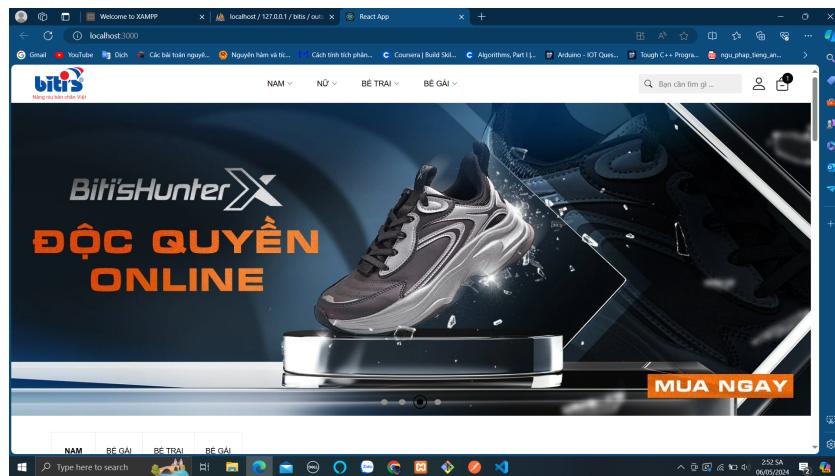
Hình 13: Các bảng trong hệ cơ sở dữ liệu



5 Hiện thực

5.1 Người dùng khi chưa đăng nhập

Là người có ít quyền nhất đối với Website. Tại đây người dùng có thể quan sát trang Web ở trang chủ và thực hiện quyền đăng ký hoặc đăng nhập tài khoản.

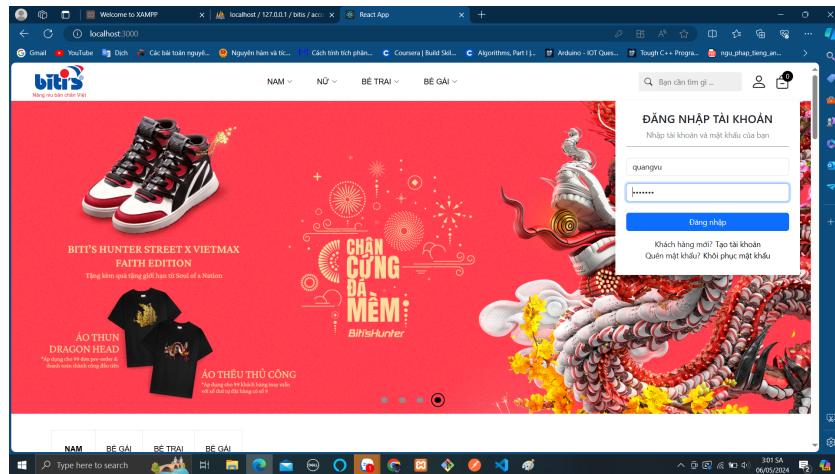


Hình 14: Trang chủ khi chưa đăng nhập



5.2 Đăng nhập

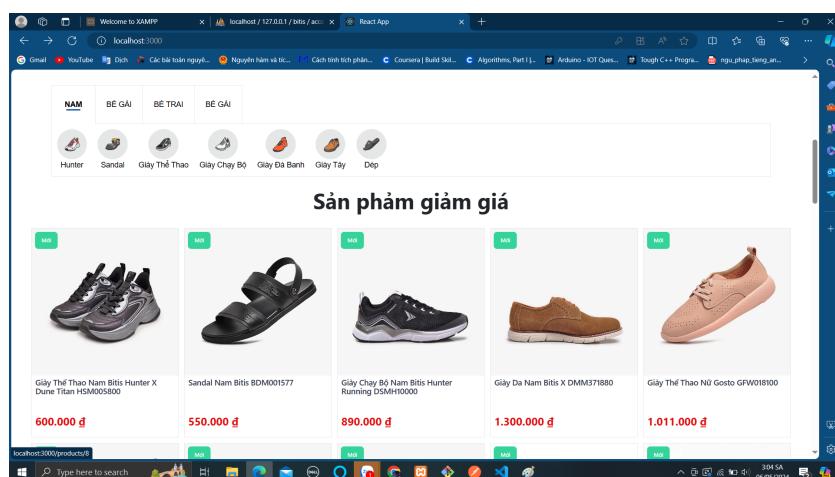
Form đăng nhập được tích hợp chung với trang web. Ở đây người dùng có thể đăng nhập với tư cách là khách hàng hoặc admin.



Hình 15: Đăng nhập tài khoản

5.3 User sau khi đăng nhập

Hình ảnh trang chủ của Website sau khi đã thực hiện đăng nhập. User có thể bắt đầu xem và mua sắm phẩm vật lúc nay.

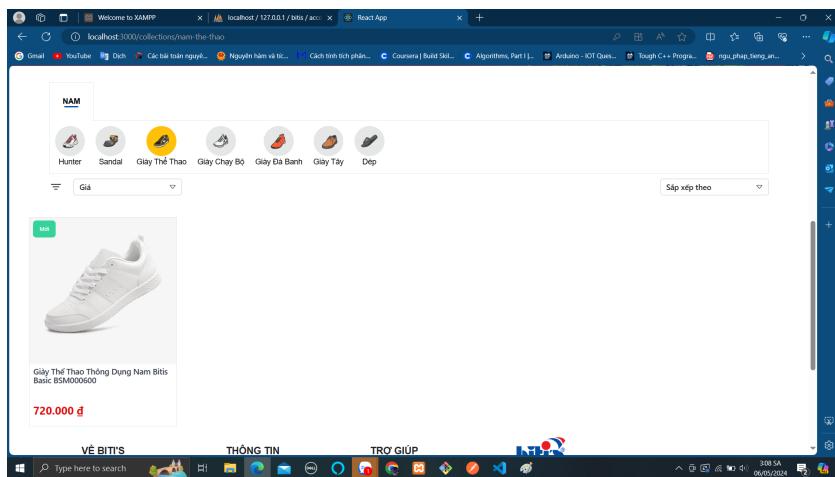


Hình 16: Xem trang chủ của website



5.4 Tìm kiếm mặt hàng của trang Web

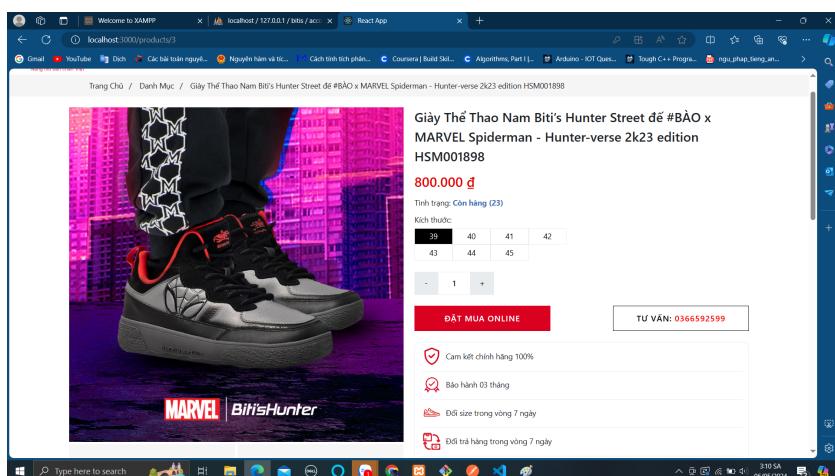
Người dùng sau khi đăng nhập có thể truy cập vào trang tìm kiếm mặt hàng theo các danh mục đã được phân loại sẵn.



Hình 17: Tìm kiếm mặt hàng theo tên

5.5 Chi tiết sản phẩm

Người dùng có thể bấm vào từng mặt hàng để xem chi tiết sau đó thêm sản phẩm vào giỏ hàng.



Hình 18: Chi tiết của sản phẩm



5.6 Giỏ hàng

Sản phẩm sau khi được thêm vào giỏ hàng. Tại đây người dùng có thể trực tiếp thanh toán thông qua nút xác nhận mua hàng.

Hình 19: Giỏ Hàng

5.7 Đánh giá của người dùng

Người dùng có thể để lại những đánh giá của mình về các sản phẩm của Shop.

Hình 20: Đánh giá



5.8 Liên hệ

Người dùng có thể liên hệ với Shop qua trang liên lạc:

VỀ BIT'S	THÔNG TIN	TRỢ GIÚP	bit's
Về Bit's	Trạng thái đơn hàng	Tuyển dụng	Nông nghiệp bền vững Việt CÔNG TY TNHH SẢN XUẤT HÀNG TIÊU DÙNG BÌNH TIỀN
Câu chuyện Bit's	Hình thức giao hàng	Hệ thống cửa hàng	Địa chỉ: 22 Lý Chiêu Hoàng, Phường 10, Quận 6, TP. Hồ Chí Minh
Bước tiến phát triển	Hình thức thanh toán	Liên hệ hợp tác	Điện thoại: (028) 38 753 443
Hoạt Động	Hướng dẫn cách chọn Size	Q&A	Email: Liên hệ các vấn đề về đơn hàng Online, kênh cửa hàng, đại lý (offline): chamsockhachhang@bit's.com.vn
Liên hệ	Chỉnh sách đổi Size		Hotline: 0966158666
	Chỉnh sách đổi trả		Thời gian tư vấn: 8h – 21h30 các ngày trong tuần (trừ ngày

Hình 21: Liên hệ Shop

5.9 Trang quản lý của Admin

Quản lý sản phẩm						
Thêm sản phẩm						
Cập nhật sản phẩm						
ID	Username	Password	Full name	Phone number	Role	Action
1	admin	21232f297a57a5a743894a0e4801fc3	admin	0366902599	admin	<button>Delete</button>
2	admin1	e00cf25ad42683b3d967b61142c6bd	admin1 admin1	0366902599	user	<button>Delete</button>
3	a0jffk123hjkjh	5e13d4470f64ab28235561ad00261e3	jhhkhkhjkjh	0366902599	user	<button>Delete</button>
6	than	7761127a460aa290ee0533098284dd1a	than than	0366902599	user	<button>Delete</button>
7	quangu	b65f30f990ec989b32c75a9fe1ec9	quangu quangu	123345454	user	<button>Delete</button>

Hình 22: Trang quản lí của Admin

6 Cách thức cài đặt

6.1 Về Front-end

Để chạy front-end, ta cần thực hiện các bước sau:

1. Cài đặt Node và NPM
2. Clone source code từ repos của nhóm.
3. Chạy npm install (hoặc npm install yarn và sau đó chạy yarn).
4. Chạy lệnh npm start (chạy yarn start nếu bước 3 đã install yarn).
5. Truy cập vào localhost trên trình duyệt.

6.2 Về Back-end

Để chạy back-end, ta cần thực hiện các bước sau:

1. Cài đặt MySQL.



2. Cài đặt PHP và Composer.
3. Clone source code từ repos của nhóm
4. Bật server XAMP.
5. Server đang được chạy trên localhost:3000



Tài liệu

1. Website chính thức của W3School. Truy cập tại: <https://www.w3schools.com/>.
2. Document ngôn ngữ PHP <https://www.php.net/manual/en/>
3. Document thư viện React [1]<https://reactjs.org/>
4. Document thư viện Tailwind <https://tailwindcss.com/>
5. Goossens, M., Mittelbach, F., Samarin, A LaTeX Companion, Addison-Wesley, Reading, MA, 1994.
6. Kopka, H., Daly P.W., A Guide to LaTeX, Addison-Wesley, Reading, MA, 1999.
7. Pan, D., "A Tutorial on MPEG/Audio Compression," IEEE Multimedia, Vol.2, pp.60-74, Summer 1998.
8. Bài bài giảng môn học Lập trình Web.