e

**NHẬN XÉT CỦA GIÁO VIÊN HƯỚNG DẪN**

*Trà Vinh, ngày 08 tháng 01 năm 2024*

**Giáo viên hướng dẫn**

*Phạm Minh Đương*

**NHẬN XÉT CỦA THÀNH VIÊN HỘI ĐỒNG**

*Trà Vinh, ngày ….. tháng …… năm ……*

**Thành viên hội đồng**

**LỜI CẢM ƠN**

Chân thành cảm ơn thầy Phạm Minh Đương đã dành thời gian, tâm huyết và kiến thức chuyên sâu để hướng dẫn và hỗ trợ tôi trong quá trình thực hiện đồ án. Những lời chỉ dẫn, phản hồi và góp ý của thầy đã giúp tôi hoàn thiện công việc một cách chất lượng và có ý nghĩa.

Cảm ơn bộ môn đã tạo cơ hội cho em cũng như toàn thể các sinh viên của bộ môn có cơ hội được thử thách và chứng minh bản thân qua các đồ án. Chân thành cảm ơn các thầy cô bộ môn Công nghệ Thông tin đã nhiệt tình và chân thành hỗ trợ tôi trong quá trình nghiên cứu đồ án.

Không chỉ thế, tôi còn muốn gửi lời cảm ơn đặc biệt đến các bạn bè và đồng đội đã chia sẻ những lời khuyên hữu ích, ý kiến chân thành và kinh nghiệm quý báu. Đó là nguồn động viên và sự hỗ trợ không ngừng, giúp tôi không chỉ vượt qua những khó khăn mà còn phát triển sâu sắc trong hành trình này.

Một lần nữa, tôi xin chân thành cảm ơn sâu sắc và bày tỏ lòng tôn trọng cao đối với công lao của thầy Phạm Minh Đương, bộ môn Công nghệ Thông tin và những người bạn đồng hành. Hy vọng rằng tinh thần hỗ trợ và khích lệ này sẽ tiếp tục đồng hành với tôi trong các hành trình sắp tới.

**MỤC LỤC**

[CHƯƠNG 1 TỔNG QUAN 12](#_Toc155041533)

[CHƯƠNG 2 NGHIÊN CỨU LÝ THUYẾT 14](#_Toc155041534)

[2.1 Cơ sở lý thuyết 14](#_Toc155041535)

[2.1.1 Tổng quan về hệ thống ứng dụng web 14](#_Toc155041536)

[2.1.2 Công nghệ 14](#_Toc155041537)

[2.1.3 Kĩ thuật 15](#_Toc155041538)

[2.1.4 Thiết kế UI/UX 15](#_Toc155041539)

[2.2 Lý luận 15](#_Toc155041540)

[2.2.1 Công nghệ 15](#_Toc155041541)

[2.2.2 Kĩ thuật CSR 16](#_Toc155041542)

[2.2.3 Thiết kế UI / UX 16](#_Toc155041543)

[2.3 Giả thuyết khoa học 17](#_Toc155041544)

[2.4 Phương pháp nghiên cứu 17](#_Toc155041545)

[CHƯƠNG 3 HIỆN THỰC HOÁ NGHIÊN CỨU 18](#_Toc155041546)

[3.1 Cài đặt phần mềm, công cụ, thiết lập môi trường trước khi viết mã. 18](#_Toc155041547)

[3.1.1 Cài đặt môi trường NodeJS 19](#_Toc155041548)

[3.1.2 Cài đặt Laragon và tạo máy chủ ảo quản lý cơ sở dữ liệu mySQL bằng PhpMyAdmin. 20](#_Toc155041549)

[3.1.3 Cài đặt phần mềm viết mã Visual Studio Code 22](#_Toc155041550)

[3.1.4 Cài đặt Postman 23](#_Toc155041551)

[3.2 Thiết kế cơ sở dữ liệu mySQL 24](#_Toc155041552)

[3.2.1 Lượt đồ cơ sở dữ liệu 24](#_Toc155041553)

[3.2.2 Mô tả một số bảng quan trọng 24](#_Toc155041554)

[3.3 Viết mã backend với ExpressJS 27](#_Toc155041555)

[3.3.1 Khởi tạo 27](#_Toc155041556)

[3.3.2 Cấu trúc thư mục 29](#_Toc155041557)

[3.3.3 Viết mã 31](#_Toc155041558)

[3.3.3.1 Mã khởi động 31](#_Toc155041559)

[3.3.3.2 Mã định tuyến 33](#_Toc155041560)

[3.3.4 Kiểm tra các tuyến của backend với Postman 34](#_Toc155041561)

[3.3.4.1 Đăng nhập 34](#_Toc155041562)

[3.3.4.2 Lấy thông tin chi tiết của một sản phẩm 35](#_Toc155041563)

[3.3.4.3 Đặt hàng 35](#_Toc155041564)

[3.4 Viết mã frontend với ReactJS 36](#_Toc155041565)

[3.4.1 Khởi tạo 36](#_Toc155041566)

[3.4.2 Cấu trúc thư mục 37](#_Toc155041567)

[3.4.3 Viết mã 38](#_Toc155041568)

[3.4.3.1 Mã khởi động 39](#_Toc155041569)

[3.4.3.2 Mã định tuyến 39](#_Toc155041570)

[3.4.4 Các giao diện của website Nowbuys 40](#_Toc155041571)

[3.4.4.1 Giao diện đăng nhập 40](#_Toc155041572)

[3.4.4.2 Giao diện trang chủ 40](#_Toc155041573)

[3.4.4.3 Trang thông tin chi tiết sản phẩm 41](#_Toc155041574)

[3.4.4.4 Trang giỏ hàng 41](#_Toc155041575)

[CHƯƠNG 4 KẾT QUẢ NGHIÊN CỨU 43](#_Toc155041576)

[4.1 Về mặt lý thuyết 43](#_Toc155041577)

[4.1.1 Framework ExpressJS 43](#_Toc155041578)

[4.1.2 Thư viện ReactJS: 43](#_Toc155041579)

[4.2 Về mặt sản phẩm 43](#_Toc155041580)

[CHƯƠNG 5 KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN 45](#_Toc155041581)

[5.1 Kết luận 45](#_Toc155041582)

[5.2 Hướng phát triển 45](#_Toc155041583)

**DANH MỤC HÌNH ẢNH**

[Hình 3.1 Lượt đồ cơ sở dữ liệu website Nowbuys 18](#_Toc155468025)

[Hình 3.2 Mở terminal tại thư mục cần tạo dự án 21](#_Toc155468026)

[Hình 3.3 Khởi tạo dự án với ExpressJS bằng terminal 21](#_Toc155468027)

[Hình 3.4 Mã khởi động ExpressJS 22](#_Toc155468028)

[Hình 3.5 Chạy dự án với ExpressJS thành công 22](#_Toc155468029)

[Hình 3.6 Cấu trúc thư mục backend 23](#_Toc155468030)

[Hình 3.7 Mã khởi động trong tệp server.js 25](#_Toc155468031)

[Hình 3.8 Mã định tuyến cho backend website Nowbuys 27](#_Toc155468032)

[Hình 3.9 Định nghĩa các tuyến đường trong tuyến xác thực. 28](#_Toc155468033)

[Hình 3.10 Định nghĩa các tuyến đường trong tuyến sản phẩm 28](#_Toc155468034)

[Hình 3.11 Chạy dự án ReactJS thành công 30](#_Toc155468035)

[Hình 3.12 Cấu trúc thư mục frontend của website Nowbuys 30](#_Toc155468036)

[Hình 3.13 Mã khởi động trong tệp App.js 32](#_Toc155468037)

[Hình 3.14 Mã định tuyến trong dự án ReactJS 32](#_Toc155468038)

[Hình 3.15 Mã nhập tên đăng nhập trong trang đăng nhập Nowbuys 33](#_Toc155468039)

[Hình 3.16 Mã xử lý nút đăng nhập trên giao diện đăng nhập Nowbuys 34](#_Toc155468040)

[Hình 3.17 Xây dựng giao diện thẻ sản phẩm website Nowbuys 35](#_Toc155468041)

[Hình 4.1 Kiểm tra tuyến đăng nhập bằng Postman 36](#_Toc155468042)

[Hình 4.2 Session được lưu sau khi đăng nhập thành công 36](#_Toc155468043)

[Hình 4.3 Lấy thông tin chi tiết của sản phẩm bằng Postman 37](#_Toc155468044)

[Hình 4.4 Sử dụng Postman kiểm tra tuyến đặt hàng 38](#_Toc155468045)

[Hình 4.5 Giao diện trang đăng ký người dùng của website Nowbuys 38](#_Toc155468046)

[Hình 4.6 Giao diện trang đăng nhập của website Nowbuys 39](#_Toc155468047)

[Hình 4.7 Trang thông tin chi tiết sản phẩm 40](#_Toc155468048)

[Hình 4.8 Trang đặt hàng website Nowbuys 40](#_Toc155468049)

[Hình 0.1 Cài đặt NodeJS 45](#_Toc155468050)

[Hình 0.2 Kiểm tra phiên bản NodeJS 45](#_Toc155468051)

[Hình 0.3 Thiết lập máy chủ ảo trên Laragon 46](#_Toc155468052)

[Hình 0.4 Giao diện chính của Laragon 47](#_Toc155468053)

[Hình 0.5 Giao diện sau khi cài đặt thành công Visual Studio Code 47](#_Toc155468054)

[Hình 0.6 Giao diện cài đặt extentions trong Visual Studio Code 48](#_Toc155468055)

[Hình 0.7 Giao diện sau khi thiết lập Postman 49](#_Toc155468056)

**DANH MỤC BẢNG BIỂU**

[Bảng 1 Bảng user trong cơ sở dữ liệu 19](#_Toc155468076)

[Bảng 2 Bảng product trong cơ sở dữ liệu 20](#_Toc155468077)

[Bảng 3 Bảng orders trong cở sở dữ liệu 20](#_Toc155468078)

[Bảng 4 Bảng otp trong cơ sở dữ liệu 21](#_Toc155468079)

**TÓM TẮT ĐỒ ÁN CƠ SỞ NGÀNH**

Trong tình hình như hiện nay, nơi mà công nghệ rất phát triển và mọi thức đều được thực hiện trực tuyến thông qua internet, cùng với đó là quyết định số 749/QĐ-TTg của Thủ tướng Chính phủ: Phê duyệt "Chương trình Chuyển đổi số quốc gia đến năm 2025, định hướng đến năm 2030" càng làm cho các hoạt động trên internet càng thêm mạnh mẽ, mua xắm trực tuyến cũng không là ngoại lệ khi có đến 81% người Việt cho biết mua sắm trực tuyến là một thói quen không thể thiếu mỗi ngày, tỉ lệ người mua sắm trực tuyến ít nhất 1 lần mỗi tuần đạt mức 59% … Điều đó càng cho thấy việc nghiên cứu và xây dưng website thương mại là điều cần thiết và cũng là yếu tố quyết định đến sự sống còn của cửa hàng ở cuộc cách mạng công nghiệp 4.0 này [1].

Vì thế nghiên cứu các công nghệ web hiện đại, phổ biến, hỗ trợ tốt các vấn đề quan trọng của một website thương mại và đảm bảo trải nghiệm người dùng tốt là điều quan trọng. Ứng dụng chúng vào xây dựng website thương mại điện tử đáp ứng được nhu cầu mua sắm như mẫu mã sản phẩm đa dạng, thông tin sản phẩm minh bạch - rõ ràng – đúng thực tế, đầy đủ thông tin đánh giá từ những người đã sử dụng, giỏ hàng để lưu lại sản phẩm chú ý, trang đặt hàng hàng trực quan dễ dàng tiếp cận, quản lý trạng thái đơn hàng 24/7 … Ngoài ra còn đảm bảo về các yêu tố tốc độ tải trang, thẫm mỹ và dễ dàng sử dụng.

Xây dựng web bán hàng Nowbuys sử dụng thư viện ReactJS và NodeJS với ExpressJS framework sẽ đáp ứng hầu hết các tiêu chí vừa nêu ở trên, đảm bảo tính hiện đại về mặt công nghệ được sử dụng để xây dựng website mà còn đáp ứng về việc trải nghiệm người dùng như trải nghiệm tiện ích, tốc độ tải trang, tốc độ phản hồi.

**MỞ ĐẦU**

1. **Lý do chọn đề tài**

Thư viện ReactJS và framework ExpressJS đều được phát triển trên ngôn ngữ lập trình Javascript, thực thi trên nền tảng NodeJS, nền tảng có hiệu suất cao. Tất cả đều dựa trên ngôn ngữ lập trình Javascript.

ReactJS sử dụng Virtual DOM và chỉ cập nhật các phần cần thiết của trang, giúp giảm độ trễ render, chuyển trang mượt mà và nhanh chóng.

ExpressJS tự do lựa chọn cấu trúc theo ý muốn, tối ưu hoá cú pháp, cung cấp nhiều phương thức/ hàm, hệ thống định tuyến mạnh mẽ, nổi bật với việc tối ưu hoá sử dụng middleware, tối ưu hoá viết code theo hướng RESTfull API.

Thiết kế web nói chung và web thương mại điện tử nói riêng đều tuân thủ các nguyên tắc thiết kế UI/UX.

Các thư viện được phát triển từ Javascript nói chung và ReactJS, ExpressJS nói riêng đều có nguồn tài liệu dồi dào, phong phú, cộng đồng hỗ trợ lớn mạnh.

Javascript 11 năm liên tiếp là ngôn ngữ lập trình phổ biến nhất thế giới vào năm 2023 chiếm 63,61% so với các ngôn ngữ khác theo thống kê của Stackoverflow [2].

React.js ở năm 2023 vẫn tiếp tục là thư viện JavaScript frontend phổ biến nhất, chiếm đến hơn 40% thị phần. Theo khảo sát của State of JavaScript vào năm 2022, 69% nhà phát triển đã sử dụng React.js và 54% cho biết họ sẽ sử dụng lại. Trong cùng một cuộc khảo sát, React.js được đánh giá là thư viện JavaScript hài lòng nhất trong số các nhà phát triển [3].

ExpressJS chiếm 19.28% độ phổ biến, xếp vị trí đầu tiên về framework backend nói chung và framework backend JavaScript nói riêng [2].

Vào giai đoạn bùng nổ công nghệ sản phẩm web không chỉ đơn giản là phục vụ nhu cầu cốt lõi của mục đích xây dựng web mà còn phải quan tâm đến vấn đề thẩm mĩ và trải nghiệm người dùng.

Các ứng dụng web sử dụng ReactJS có tốc độ chuyển trang, thao tác, phản hồi nhanh chóng có thể nói gần như là tức thời (nếu như loại bỏ đi khoảng thời gian chờ phản hồi từ server) đây là điều cực kì quan trọng trong việc xây dựng ứng dụng web nói chung và ứng dụng web thương mại điện tử nói riêng.

Xây dựng các UI đặt yêu cầu về thẫm mỹ, thu hút ánh nhìn. Tối ưu hoá UX theo slogan “đơn giản, thuận tiện, người dùng là tham chiếu”.

1. **Mục tiêu**

Nắm vững thư viện ReactJS, framework ExpressJS về các công nghệ nổi bật và tính ứng dụng của chúng.

Hiểu biết và thành thạo các quy tắc thiết kế giao diện, trải nghiệm người dùng đối với web thuộc loại thương mại điện tử.

Hiểu và nắm vững quy trình tạo ra một sản phẩm từ lập trình thực tế.

Xây dựng thành công web bán hàng có đầy đủ giao diện và backend.

Phát triển và xây dựng mới kĩ năng tự học hoàn thiện hơn từ những kinh nghiệm khi nghiên cứu.

1. **Nội dung**

Tìm hiểu thư viện ReactJS để xây dựng giao diện web bán hàng Nowbuys.

Tìm hiểu framework ExpressJS để xây dựng hệ thống backend theo hướng RESTful API phục vụ cho web bán hàng Nowbuys được viết bằng thư viện ReactJS.

Tìm hiểu các quy tắc thiết kế giao diện, trải nghiệm người dùng đối với thiết kế web đặc biệt là web thương mại điện tử.

Nghiên cứu quy trình tạo ra một sản phẩm từ lập trình thực tế.

Xây dựng website bán hàng Nowbuys để minh họa cho các thư viện/framework đã nghiên cứu.

1. **Phương pháp nghiên cứu**
   1. **Nghiên cứu lý thuyết**

Đọc và nghiên cứu tài liệu chính thức từ trang chủ của thư viện ReactJS và các tài liệu tham khảo từ các nguồn uy tín trên internet.

Đọc và nghiên cứu tài liệu chính thức từ trang chủ của framework ExpressJS và các tài liệu tham khảo từ các nguồn uy tín trên internet.

Nghiên cứu các quy tắc thiết kế giao diện, trải nghiệm người dùng, và nguyên tắc thiết kế web đặc biệt là web thương mại điện tử thông qua các web thương mại điện tử sẳn có kết hợp cùng với các video bài giảng uy tín.

* 1. **Thực nghiệm**

Lập trình giao diện web bán hàng Nowbuys bằng thư viện ReactJS.

Lập trình backend RESTful API cho web bán hàng Nowbuys bằng framework ExpressJS.

Triển khai dự án lên server tự xây dựng tại nhà, trỏ tên miền về server để công khai sản phẩm ra internet.

Ghi chú lại các vấn đề gặp phải và cách giải quyết chúng để làm kinh nghiệm cho bản thân.

1. **Đối tượng và phạm vi nghiên cứu**
   1. **Đối tượng**

Lập trình frontend ứng dụng web bằng ReactJS.

Lập trình backend ứng dụng web bằng ExpressJS.

Quy tắc thiết kế UI/UX, quy tắc thiết kế ứng dụng web thương mại điện tử.

Xây dựng website bán hàng sử dụng các công nghệ, quy tắc đã nghiên cứu.

* 1. **Phạm vi**

Về thư viện ReactJS: Nghiên cứu các khái niệm cơ bản, logic vận hành của ReactJS, nghiên cứu Virtual DOM, cách lập trình theo hướng component hoá.

Về framework ExpressJS: Nghiên cứu các khái niệm cơ bản, logic vận hành, nghiên cứu router định tuyến tối ưu, nghiên cứu middleware dùng cho bảo mật và ghi log.

Về nguyên tắc UI/UX: Nghiên cứu các khái niệm cơ bản về sử dụng màu sắc và thiết kế bố cục phù hợp với người dùng web thương mại điện tử đặc biệt là thói quen sử dụng website của người dùng Việt.

# TỔNG QUAN

Trong những năm gần đây, nhu cầu về website thương mại điện tử đã trở nên vô cùng quan trọng và ngày càng gia tăng. Điều này xuất phát từ sự chuyển đổi mạnh mẽ của người tiêu dùng sang việc sử dụng internet để mua sắm và thực hiện các giao dịch mua bán. Sự tiện lợi, đa dạng và khả năng so sánh giá dễ dàng đã làm cho mô hình thương mại điện tử trở thành lựa chọn ưa thích của đông đảo người tiêu dùng.

Đặc biệt, ảnh hưởng của sự phổ cập của thiết bị di động và tốc độ internet ngày càng nhanh cũng đóng góp quan trọng vào sự gia tăng này. Người tiêu dùng có khả năng truy cập và mua sắm từ bất kỳ đâu, bất kỳ khi nào, từ các thiết bị di động của họ, tạo ra một trải nghiệm mua sắm linh hoạt và thuận tiện.

Việc xây dựng và duy trì một trang web thương mại điện tử hiệu quả không chỉ mở ra cơ hội tiếp cận thị trường toàn cầu mà còn giúp tối ưu hóa quy trình bán hàng, tăng cường tương tác với khách hàng, và xây dựng uy tín thương hiệu.

Để trang web thương mại điện tử hoạt động hiệu quả, cần đảm bảo một loạt các yếu tố quan trọng. Trước hết, tốc độ tải trang cần được tối ưu để đảm bảo trải nghiệm người dùng không bị gián đoạn và gọn nhẹ để giảm thời gian chờ đợi. Đồng thời, việc hiển thị lỗi phải được thực hiện một cách trực quan để người dùng có thể dễ dàng hiểu và khắc phục vấn đề. Để tránh sự nhàm chán, hiệu ứng tải trang cần được thiết kế một cách sinh động và hấp dẫn, tạo ra một trải nghiệm người dùng tích cực. Giao diện thiết kế màu sắc cũng đóng vai trò quan trọng, cần phải phù hợp với xu hướng hiện đại và tạo nên một hình ảnh thương hiệu độc đáo. Cuối cùng, trải nghiệm người dùng cần được đặt lên hàng đầu, với giao diện dễ sử dụng và thuận tiện để tối ưu hóa sự hài lòng của khách hàng khi duyệt và mua sắm trên trang web.

Trong những năm trở lại đây thư viện, framework, ngôn ngữ, phần mềm trong lập trình ứng dụng web xuất hiện rất phong phú đã dạng, trong số đó NodeJS là cái tên nổi bật nhất. Được tạo bởi Ryan Dahl từ năm 2009 và phát triển dưới sự bảo trợ của Joyent. Cơ chế bất đồng bộ (non-blocking) xử lý hàng ngàn kết nối trong khoảng thời gian ngắn, single-thread xử lý nhiều kết nối chỉ với một luồng làm cho nhu cầu về RAM thấp, tối ưu hoá cho các tác vụ thời gian thực, dễ dàng mở rộng dự án,… Những điều đó đã nhanh chóng đưa NodeJs trở thành môi trường lập trình được tìm kiếm nhiều nhất trên Google, được các tập đoàn công nghệ lớn như Twitter, Netflix, PayPal, LinkedIn,… sử dụng.

NodeJS được xây dựng trên bộ máy Chrome’s V8 JavaScript vì thế khi NodeJS trở thành xu hướng thì hàng loạt các thư viện / framework thuộc họ JS cũng được ra đời.

Về frontend web, ReactJS cái tên phổ biến, được đông đảo lập trình viên sử dụng. Các thống kê nổi bật năm 2023, ReactJS là thư viện frontend được sử dụng nhiều nhất, 40,58% số lập trình viên được phỏng vấn bởi Stackoverflow có sử dụng ReactJS [2].

Về backend web, ExpressJS là một lựa tối ưu cho việc xây dựng server web, là framework backend có cú pháp đơn giản, không bị ràng buộc về cấu trúc, có thể tạo ra backend web một cách nhanh chóng. Theo Stackoverflow năm 2023, NodeJS chiếm 42,65% các công nghệ được sử dụng trong đó ExpressJS đứng thứ 4 chiếm 19.28% tổng số lập trình viên sử dụng [2].

Ứng dụng thư viện ReactJS và framework ExpressJS xây dựng website bán hàng Nowbuys trở thành website bán hàng độc đáo và hiện đại. Trang web này mang lại trải nghiệm mua sắm trực tuyến tốt nhất cho người dùng, kết hợp giữa giao diện người dùng thân thiện và tính năng đa dạng.

Website Nowbuys được thiết kế với giao diện người dùng thân thiện, dễ sử dụng và hiện đại. Giao diện này giúp người dùng dễ dàng duyệt qua các sản phẩm, tìm kiếm thông tin và thực hiện các giao dịch một cách thuận lợi.

Website Nowbuys có một hệ thống quản lý giỏ hàng linh hoạt, cho phép người dùng dễ dàng thêm, xóa và cập nhật sản phẩm trong giỏ hàng. Quá trình thanh toán được tối ưu hóa để đảm bảo tính an toàn.

Với Nowbuys người dùng có thể dễ dàng theo dõi tình trạng đơn hàng của mình thông qua giao diện cá nhân, từ khi đặt hàng đến khi nhận được sản phẩm.

Nowbuys kết nối cộng đồng mua sắm bằng cách để người dùng đánh giá và đưa ra nhận xét về sản phẩm.

# NGHIÊN CỨU LÝ THUYẾT

## Tổng quan về hệ thống ứng dụng web

Một ứng dụng web cơ bản được cấu thành từ giao diện (frontend hay client) và máy chủ xử lý (server hay còn gọi là backend). Frontend là phần giao diện người dùng, chịu trách nhiệm hiển thị thông tin và tương tác với người dùng. Backend là phần xử lý dữ liệu, chịu trách nhiệm lưu trữ, xử lý dữ liệu được gửi đến từ client và phản hồi các yêu cầu của client. Xoay quanh đó là các kĩ thuật và phương thức trao đổi thông tin giữa 2 thành phần này, chúng phụ thuộc vào kĩ thuật được chọn và tuỳ vào tính chất, yêu cầu của website [4].

### Các công nghệ website

Single page application (SPA) là công nghệ phổ biến trong thiết kế website hiện đại. Được tạo ra bằng cách sử dụng các kĩ thuật frontend để hiển thị trang web mà không cần phải tải lại trang. Tập trung xử lý ở trình duyệt của người dùng vì thế nâng cao vai trò của frontend, giảm áp lực cho backend (điều này rất quan trọng cho các hệ thống lớn có lượng người dùng truy cập cao), rất có lợi về mặt tốc độ phản hồi yêu cầu khi yêu cầu/ phản hồi đều là các chuỗi JSON (JavaScript Object Notation) khá nhỏ so với toàn bộ HTML (HyperText Markup Language) được trả về từ server [5].

ReactJS là một thư viện JavaScript mã nguồn mở được xây dựng và phát triển bởi Facebook (hiện nay l để tạo ra những ứng dụng web hấp dẫn, nhanh và hiệu quả với mã hóa tối thiểu. Mục đích cốt lõi của ReactJS không chỉ khiến cho trang web phải thật mượt mà còn phải nhanh, khả năng mở rộng cao và đơn giản. Sức mạnh của ReactJS xuất phát từ việc tập trung vào các thành phần riêng lẻ khi viết mã và Virtual DOM (Document Object Model) là sức mạnh chính cho hiệu suất kết xuất giao diện. Khi viết mã thay vì làm việc trên toàn bộ ứng dụng web, ReactJS cho phép lập trình viên có thể chia nhỏ giao diện người dùng phức tạp thành các thành phần đơn giản hơn, giúp việc quản lý dễ dàng hơn thông qua mô hình tái sử dụng thành phần được xây dựng và phát triển [6].

ExpressJS là một framework Node.js được sử dụng để xây dựng backend. Không ràng buộc về mặt cấu trúc do đó có thể lựa chọn bất kì mô hình backend nào để sử dụng, khả năng định tuyết mạnh mẽ giúp cho việc viết API trở nên đơn giản vì thế framework này rất phù hợp cho hệ thống web sử dụng Restful API, middleware là một vấn đề quan trọng trong hệ thống backend cũng được framework này hỗ trợ hoàn toàn, ExpressJS cũng cho phép can thiệp chỉnh sửa thông báo lỗi theo ý muốn từ đấy nhà phát triển có thể quản lý phản hồi của mình chủ động hơn tạo ra nhiều lợi ích hơn cho phía frontend tối ưu hoá các trải nghiệm người dùng [7].

### Kĩ thuật của website

Client-side Rendering (CSR) là một kỹ thuật tạo ra các trang web động bằng cách kết xuất các nội dung của trang trực tiếp trên phía trình duyệt web của người dùng thay vì trên phía máy chủ. Với CSR, khi có truy vấn từ trình duyệt gửi đến máy chủ, máy chủ sẽ gửi về một trang HTML trống, toàn bộ mã JavaScript và Css. Sau khi các tài nguyên được tải xong mã JavaScript sẽ được thực thi kết xuất nội dung trang vào trang HTML tuỳ thuộc vào địa chỉ (url) mà người dùng đang truy cập [6].

### Thiết kế UI/UX cho website

Giao diện người dùng, hay UI (User Interface), không chỉ là việc thiết kế mà còn bao gồm một loạt các yếu tố như màu sắc, bố cục, fonts chữ, hình ảnh, và nhiều điều khác trên toàn bộ trang web. Trong thời đại hiện nay, vai trò của UI trở nên vô cùng quan trọng không chỉ trong mọi lĩnh vực của cuộc sống mà còn đặc biệt quan trọng trong lĩnh vực thương mại điện tử.

Ngược lại, trải nghiệm người dùng, hay UX (User Experience), tập trung vào việc thiết kế sao cho người dùng có trải nghiệm tốt nhất. Điều này bao gồm việc nghiên cứu và đánh giá thói quen cũng như nhu cầu sử dụng sản phẩm của khách hàng. Từ những thông tin này, việc thiết kế UX được đặc biệt chú trọng vào tính tiện ích và hiệu quả khi hệ thống hoạt động.

## Lý do lựa chọn công nghệ, kĩ thuật xây dựng website Nowbuys

### Công nghệ web

ReactJS và ExpressJS đều là những công nghệ mạnh mẽ, nổi bật với hiệu năng và khả năng xây dựng tối ưu. Trang web được render trên trình duyệt với tốc độ gần như tức thì của ReactJS, mang lại trải nghiệm người dùng vô cùng nhanh chóng. Với ExpressJS, chúng ta được hỗ trợ đầy đủ về Restful API, middleware, và routing - những yếu tố cơ bản quan trọng trong việc xây dựng backend cho hệ thống.

Sự kết hợp giữa ReactJS và ExpressJS là một sự lựa chọn hoàn hảo, đặc biệt là trong lĩnh vực ứng dụng web thương mại điện tử. Sự kết hợp này không chỉ đảm bảo hiệu năng cao mà còn tạo ra trải nghiệm người dùng xuất sắc về mặt sản phẩm. Với ExpressJS, quá trình xây dựng backend trở nên đơn giản hóa đáng kể, khi mọi yếu tố quan trọng của một cấu trúc backend đều được hỗ trợ đầy đủ.

ReactJS, với hướng tiếp cận tái sử dụng thành phần, làm cho việc xây dựng trang web trở nên đáng giá, bởi vì có thể nhanh chóng tái sử dụng các thành phần cho các sản phẩm khác nhau. Sự kết hợp của hai công nghệ này không chỉ mang lại hiệu suất và độ tin cậy, mà còn giúp tối ưu hóa quá trình phát triển ứng dụng web.

### Kĩ thuật CSR trong thiết kế website

Client-side Rendering có những đặc điểm nổi bật đảm bảo về tốc độ phản hồi, hiệu năng của web do các trang được render trực tiếp từ trình duyệt mọi tương tác với backend đều thông qua API, tương tác giữa người dùng và web tối ưu nhất do không phải tải lại trang khi người dùng thao tác tạo cảm giác liền mạch khi sử dụng. Đặc biệt kĩ thuật này phù hợp với công nghệ được sử dụng ở trên là ReactJS [6].

### Thiết kế UI / UX cho website

Ngày nay chúng ta dễ dàng bị thu hút bởi những website được thiết kế với giao diện tốt, hợp thời và rất dễ thấy sự phàn nàn từ người dùng khi website có sự tương tác phức tạp thiếu trực quan cho các tính năng. Những điều đấy cho thấy rằng website ngày nay ngoài việc đảm bảo chuyên biệt về tính năng thì còn phải cân bằng với thiết kế giao diện và tương tác người dùng ở lại lâu hơn với website, giúp tăng lượng người tiêu dùng và tăng doanh thu.

UI là yếu tố then chốt đầu tiên quyết định đến việc tiếp tục hay rời đi của người dùng khi lần đầu họ truy cập, đảm bảo họ có thể dễ dàng nhìn thấy các tính năng cơ bản và chuyên biệt của website. Các công trình khoa học gần đây cho thấy rằng màu sắc ảnh hưởng rất lớn đến cảm xúc và các giác quan của con người, tuỳ vào loại màu sắc sẽ gợi cho người dùng một cảm xúc nhất định, làm chi phối đến các hành động của mình với website.

UX là yếu tố quan trọng chính giữ chân người dùng lâu dài, vì UX được hình thành khi có sự nghiên cứu về thói quen, hành vi… của người dùng được nghiên cứu theo từng lĩnh vực, từng khu vực địa lý thậm chí là theo văn hoá. Điều đó làm cho website trở nên gần gủi, thân thiện dễ sử dụng, thu hút người dùng hơn.

## Khái niệm bán hàng online.

Bán hàng online là quá trình thực hiện giao dịch mua bán qua Internet, trong đó người mua và người bán không cần phải gặp trực tiếp. Khái niệm này bao gồm việc quảng cáo, bày bán và mua sắm sản phẩm và dịch vụ thông qua các trang web, ứng dụng di động, và các nền tảng thương mại điện tử. Bán hàng online mang lại sự tiện lợi, đa dạng lựa chọn sản phẩm, khả năng so sánh giá dễ dàng, và tạo ra trải nghiệm mua sắm linh hoạt cho người tiêu dùng, đồng thời mở rộng phạm vi thị trường cho doanh nghiệp.

## Các công trình nghiên cứu liên quan

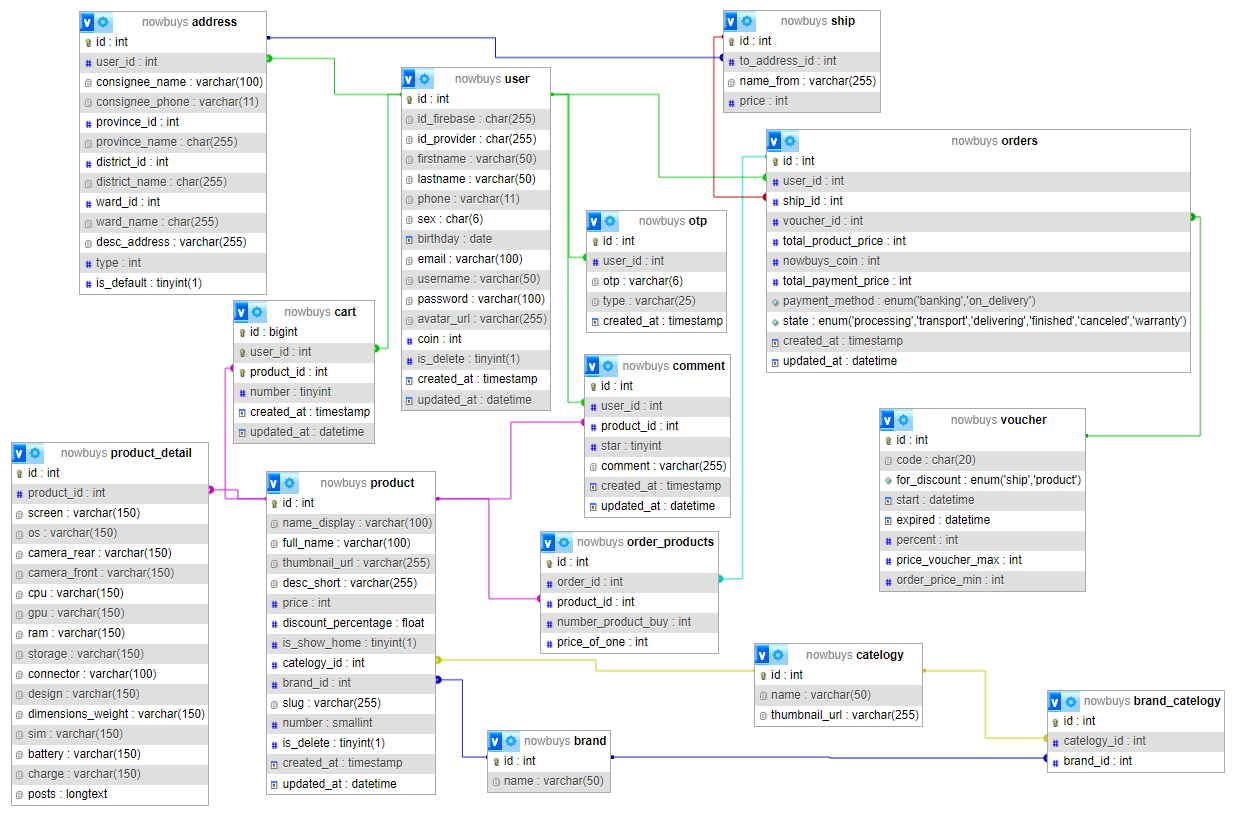
**“Lập trình web với React”** báo cáo thực tập công ty TNHH Virtual Desire Event sử dụng ReactJS xây dựng web của Nguyễn Thanh Hoàng sinh viên trường đại học Công nghệ Thông tin.

**“Xây dựng trang web review sách”** báo cáo thực tập về việc xây dựng trang web bằng công nghệ NodeJS và ExpressJs của sinh viên khoa Công nghệ Thông tin đại học Bách khoa Đà Nẵng.

# HIỆN THỰC HOÁ NGHIÊN CỨU

## Thiết kế cơ sở dữ liệu mySQL

### Lượt đồ cơ sở dữ liệu



Hình 3.1 Lượt đồ cơ sở dữ liệu website Nowbuys

**otp:** Dùng để lưu trữ mã OTP xác thực tài khoản email mỗi khi người dùng tạo tài khoản mới và mỗi hàng chỉ tồn tại trong một khoản thời gian nhất định.

**voucher:** Dùng để lưu trữ các mã giảm giá gồm mã, sử dụng cho sản phẩm hay vận chuyển, thời gian bắt đầu, thời gian hết hạn, phần trăm được giảm, giá tối đa được giảm và giá tối thiểu để áp dụng mã.

### Mô tả một số bảng quan trọng

Bảng “user”: chứa thông tin người dùng

|  |  |  |  |
| --- | --- | --- | --- |
| *Thuộc tính* | *Mô tả* | *Kiểu dữ liệu* | *Ràng buộc* |
| id | Mã người dùng | int | Primary key |
| id\_firebase | Mã người dùng khi đăng nhập bằng tài khoản bên thứ 3 do firebase cấp | varchar(255) |  |
| id\_provider | Mã người dùng khi đăng nhập bằng tài khoản bên thứ 3 | varchar(255) |  |
| firstname | Tên | varchar(50) |  |
| lastname | Tên | varchar(50) |  |
| phone | Số điện thoại | varchar(11) |  |
| sex | Giới tính | varchar(6) |  |
| birthday | Ngày sinh | date |  |
| email | Email đăng ký | varchar(100) |  |
| username | Tên đăng nhập | varchar(50) |  |
| password | Mật khẩu đăng nhập | varchar(100) |  |
| avatar\_url | Địa chỉ ảnh đại diện | varchar(100) |  |
| coin | Xu tích luỹ | int |  |
| is\_delete | Trạng thái xoá / chưa xoá tài khoản | tinyint |  |
| created\_at | Thời gian tạo | timestamp |  |
| updated\_at | Thời gian chỉnh sửa | datetime |  |

Bảng 1 Bảng user trong cơ sở dữ liệu

Bảng “product”: Chứa thông tin sản phẩm.

|  |  |  |  |
| --- | --- | --- | --- |
| **Thuộc tính** | **Mô tả** | **Kiểu dữ liệu** | **Ràng buộc** |
| id | Mã sản phẩm | int | Primary key |
| name\_display | Tên sản phẩm được hiển thị giản lượt | varchar(100) |  |
| full\_name | Tên sản phẩm đầy đủ | varchar(100) |  |
| thumbnail\_url | Địa chỉ hình nhỏ | varchar(255) |  |
| desc\_short | Mô tả ngắn | varchar(255) |  |
| price | Giá | int |  |
| discount\_percentage | Phần trăm giảm giá | float |  |
| is\_show\_home | Được / không hiển thị ở trang chủ | tinyint |  |
| catelogy\_id | Mã danh mục | int | Foreign key |
| brand\_id | Mã thương hiệu | int | Foreign key |
| slug | Mã thay thế id | varchar(255) | Unique |
| number | Số lượng sản phẩm trong kho | smallint |  |
| is\_delete | Trạng thái xoá / chưa xoá tài khoản | tinyint |  |
| created\_at | Thời gian tạo | timestamp |  |
| updated\_at | Thời gian chỉnh sửa | datetime |  |

Bảng 2 Bảng product trong cơ sở dữ liệu

Bảng “orders”: Chứa thông tin đơn đặt hàng

|  |  |  |  |
| --- | --- | --- | --- |
| **Thuộc tính** | **Mô tả** | **Kiểu dữ liệu** | **Ràng buộc** |
| id | Mã đơn đặt hàng | int | Primary key |
| user\_id | Mã người dùng | int | Foreign key |
| ship\_id | Mã vận chuyển | int | Foreign key |
| voucher\_id | Mã mã giảm giá | int | Foreign key |
| total\_product\_price | Tổng giá trị sản phẩm | int |  |
| nowbuys\_coin | Xu sử dụng | int |  |
| total\_payment\_price | Giá trị thanh toán | int |  |
| payment\_method | Phương thức thanh toán | enum |  |
| state | Trạng thái đơn đặt hàng | enum |  |
| created\_at | Thời gian tạo | timestamp |  |
| updated\_at | Thời gian chỉnh sửa | datetime |  |

Bảng 3 Bảng orders trong cở sở dữ liệu

Bảng “otp”: Chứa các mật khẩu dùng một lần.

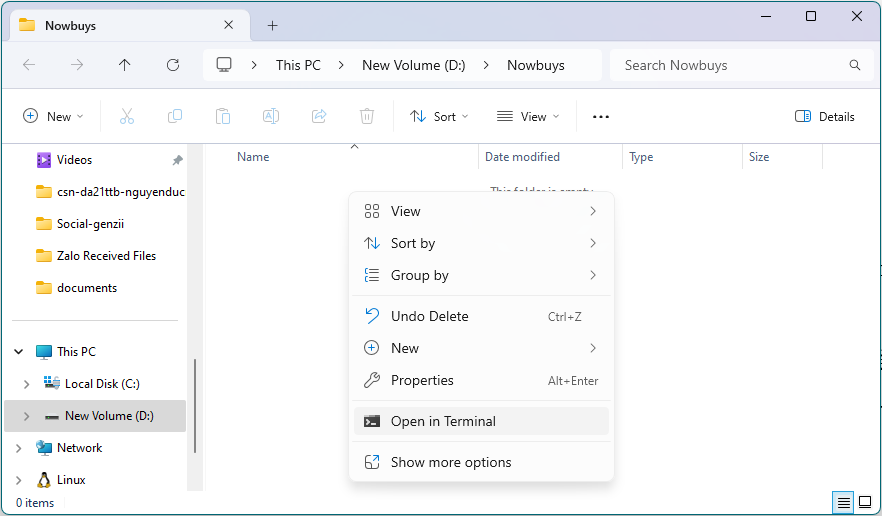
|  |  |  |  |
| --- | --- | --- | --- |
| **Thuộc tính** | **Mô tả** | **Kiểu dữ liệu** | **Ràng buộc** |
| id | Mã mã otp | int | Primary key |
| user\_id | Mã người dùng | int | Foreign key |
| otp | Mã otp | varchar(4) |  |
| type | Kiểu phục vụ | varchar(10) |  |
| created\_at | Thời gian tạo | timestamp |  |

Bảng 4 Bảng otp trong cơ sở dữ liệu

## Viết mã backend cho website Nowbuys bằng ExpressJS

### Khởi tạo dự án

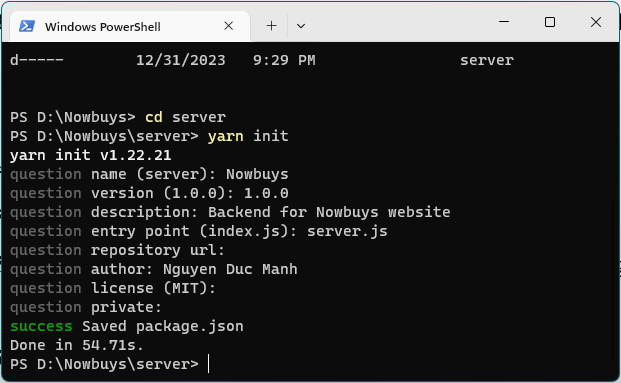
**Bước 1:** Mở terminal tại thư mục cần tạo dự án.



Hình 3.2 Mở terminal tại thư mục cần tạo dự án

**Bước 2:** Chạy lệnh “mkdir server” để tạo thư mục chứa mã backend cho website Nowbuys, sau đó di chuyển vào thư mục “server” bằng lệnh “cd server”.

**Bước 3:** Chạy lệnh “yarn init” để khởi tạo dự án, điền các thông tin theo yêu cầu hoặc có thể bỏ qua (nhấn phím enter).



Hình 3.3 Khởi tạo dự án với ExpressJS bằng terminal

**Bước 4:** Chạy lệnh “yarn add express cookie-session cors dotenv mysql2 nodemon” tải các phụ thuộc cần thiết cho dự án.

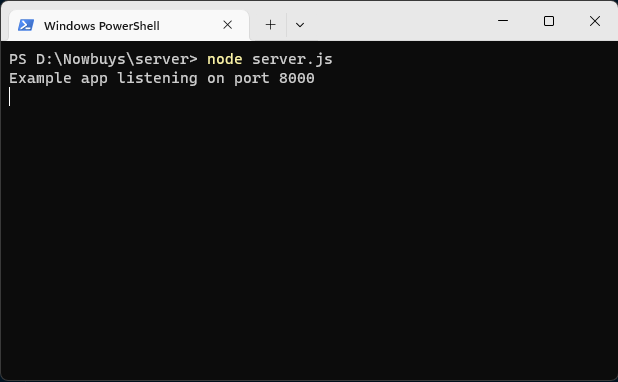
**Bước 5:** Chạy lệnh “Out-File -FilePath server.js -Force” để tạo tệp gốc (khởi động).

**Bước 6:** Thêm đoạn mã khởi động sau vào tệp server.js:



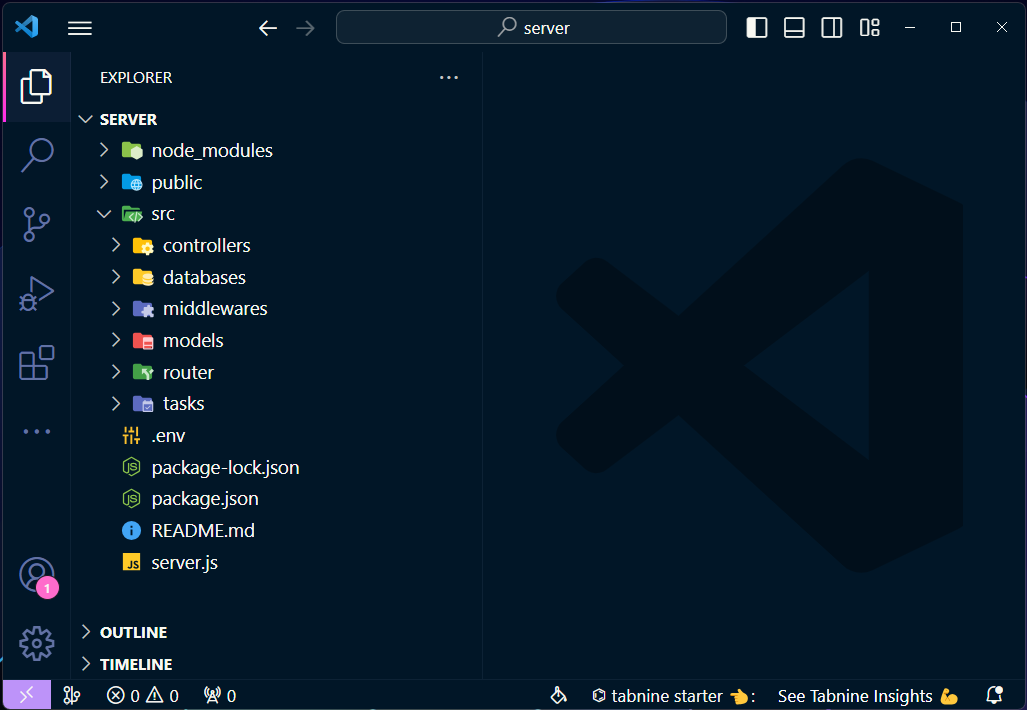
Hình 3.4 Mã khởi động ExpressJS

**Bước 7:** Chạy lệnh “node server.js” để chạy dự án.



Hình 3.5 Chạy dự án với ExpressJS thành công

### Cấu trúc thư mục



Hình 3.6 Cấu trúc thư mục backend

**node\_modules:** Chứa các thư viện, phụ thuộc cần thiết được cài đặt và mặc định của ExpressJS (Trong suốt quá trình viết mã không thay đổi bất cứ điều gì trong thư mục này).

**public:** Chứa các dữ liệu tĩnh như hình ảnh, video, âm thanh, văn bản, … Có thể thiết lập cho phép truy cập trực tiếp từ bên ngoài mà không thông qua bất kì sự kiểm duyệt nào.

**src:** Đây là thư mục làm việc chính và toàn bộ mã được viết sẽ được lưu trữ trong thư mục này.

**src / controllers:** Chứa các mã xử lý logic và dữ liệu, được dành riêng để gọi đến “src / models” tìm nạp dữ liệu hoặc trả về dữ liệu tuỳ theo yêu cầu của người dùng.

**src / models:** Chứa các mã và lệnh truy vấn trực tiếp đến cơ sở dữ liệu.

**src / databases:** Chứa các mã kết nối đến các cơ sở dữ liệu.

**src / middlewares:** Chứa các mã xử lý trung gian cho các yêu cầu gửi đến server như kiểm tra tính hợp lệ của yêu cầu, kiểm tra thông tin người dùng, …

**src / router:** Chứa mã định tuyến cho các truy vấn gọi đến, điều hướng các truy vấn đó đến các controllers phù hợp.

**src / tasks:** Chứa các mã tổng hợp cho các xử lý riêng biệt nào đó.

**.env:** Nơi chứa các biến môi trường. Thường chứa các key quan trọng liên quan nhiều đến yếu tố bảo mật, ngoài ra khi sử dụng biến ở tệp này về lâu dài khi dự án được triển khai ta cần thay đổi giá trị nào đó ta chỉ cần thay đổi ở tệp này mà không phải chạy lại toàn bộ dự án.

**server.js:** Tệp chứa mã khởi động cho các thành phần được viết trong các thư mục phía trên.

### Viết các mã quan trọng

#### Mã khởi động



Hình 3.7 Mã khởi động trong tệp server.js

Hàng 1 - 5: Thêm các thư viện cần thiết:

* express: Thêm framework Express vào tệp khởi động.
* cors (Cross-origin resource sharing): Thư viện cho phép các tài nguyên của của một trang web/ hệ thống web có thể được truy vấn từ tên miền (domain) khác với domain của trang đó.
* Cookie-session: Thư viện cho phép mã hoá và giải mã session từ yêu cầu hoặc phản hồi http.
* Body-paser: Thư viện phần mềm trung gian (middleware) để phân tích dữ liệu đến từ các yêu cầu HTTP dưới dạng mã hóa URL hoặc dưới dạng JSON và chuyển đổi chúng thành đối tượng JavaScript.
* dotenv/config: Thêm biến môi trường vào dự án mức toàn cục.
* router: Hàm định nghĩa và điều hướng các truy vấn.

Dòng 10: Chỉ định thư mục public là thư mục chứa các tài nguyên được truy cập mà không chịu sự ràng buộc nào.

Dòng 11 - 17: Thiết lập cors cho phép bất kì yêu cầu nào được gửi từ domain <http://localhost:3000> (Khi triển khai dự án thực tế sẽ thay đổi domain này thành domain của frontend).

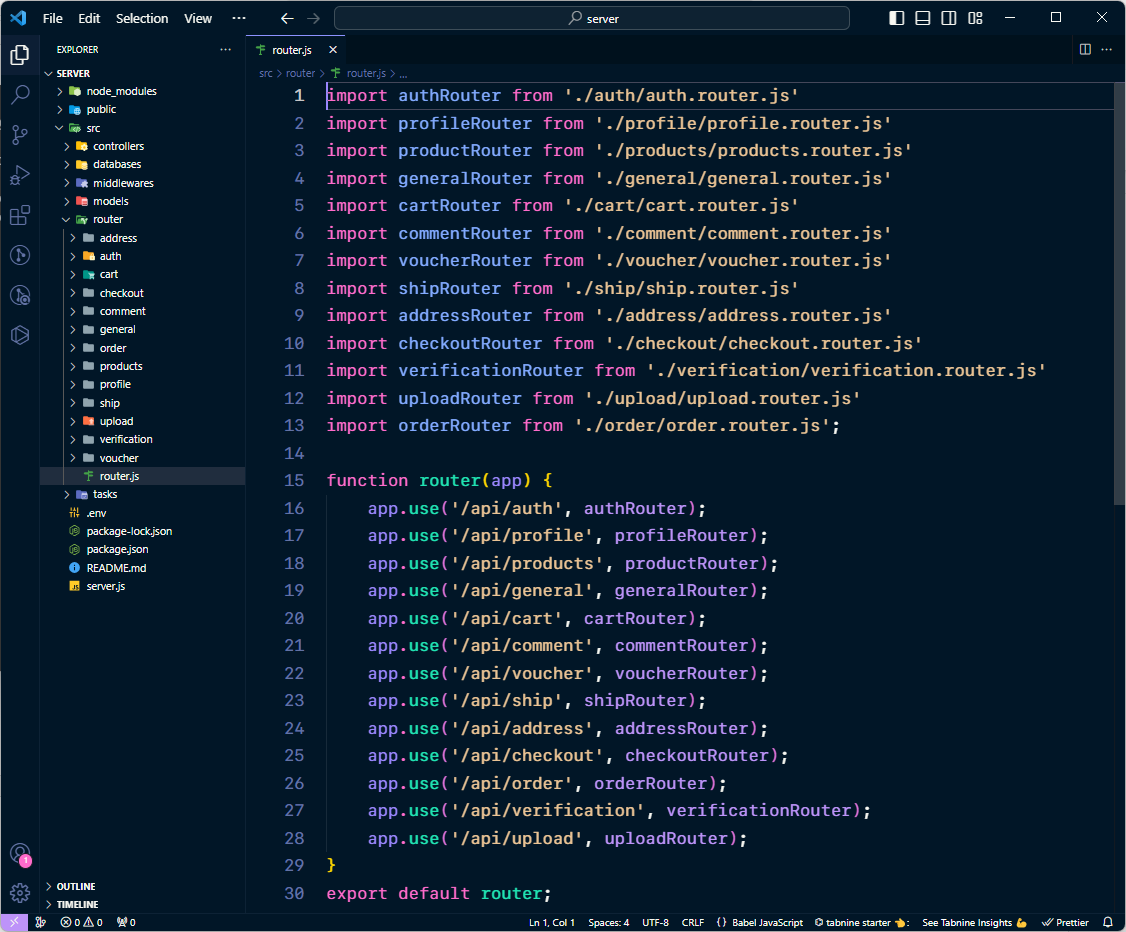
Dòng 18 - 24: Thiết lập cho cookie-session mã hoá/ giải mã session truy vấn/ phản hồi.

Dòng 26, 27: Thiết lập body-paser cho mọi truy vấn đến backend.

Dòng 29: Truyền biến “app” vào hàm “router” để xử lý, định tuyến các truy vấn.

Dòng 30, 31: Khởi tạo máy chủ với cổng 8000 và khởi chạy máy chủ.

#### Mã định tuyến



Hình 3.8 Mã định tuyến cho backend website Nowbuys

Dòng 1 - 13: Dùng để thêm các “controller” vào “router”.

Dòng 15 - 29: Xử lý các tuyến và gọi đến các controller phù hợp.

Dòng 30: Xuất hàm “router” để sử dụng ở tệp server.js.

### Viết các mã định tuyến cho các tuyến quan trọng.

#### Tuyến xác thực người dùng

A screen shot of a computer screen

Description automatically generated

Hình 3.9 Định nghĩa các tuyến đường trong tuyến xác thực.

Dòng 1 – 2: Thêm công cụ định tuyến từ thư viện express và khởi tạo biến định tuyến ở dòng 2.

Dòng 4: Thêm lớp xử lý xác thực người dùng vào (Lớp này thuộc lớp Controller trong mô hình MVC).

Dòng 6: Định nghĩa tuyến đường lấy thông tin người dùng bằng id theo.

Dòng 7: Định nghĩa tuyến đường đăng nhập bằng tài khoản Nowbuys.

Dòng 8: Định nghĩa tuyến đường đăng nhập bằng tài khoản Google.

Dòng 9: Định nghĩa tuyến đường đăng xuất khỏi website.

#### Tuyến sản phẩm

A screen shot of a computer screen

Description automatically generated

Hình 3.10 Định nghĩa các tuyến đường trong tuyến sản phẩm

Dòng 1 – 2: Thêm công cụ định tuyến từ thư viện express và khởi tạo biến định tuyến ở dòng 2.

Dòng 4: Thêm lớp xử lý sản phẩm vào (Lớp này thuộc lớp Controller trong mô hình MVC).

Dòng 6: Định nghĩa tuyến đường tìm kiếm sản phẩm qua các từ.

Dòng 7: Định nghĩa tuyến đường lấy danh sách sản phẩm được tìm nhiều nhất.

Dòng 9: Định nghĩa tuyến đường lấy danh sách sản phẩm được chỉ định hiển thị ở trang chủ.

Dòng 10: Định nghĩa tuyến đường lấy sản phẩm theo các điều kiện loại sản phẩm, thương hiệu, cách sắp xếp.

Dòng 12: Định nghĩa tuyến đường lấy thông tin chi tiết của sản phẩm qua slug của sản phẩm đó.

## Viết mã frontend cho website Nowbuys bằng ReactJS

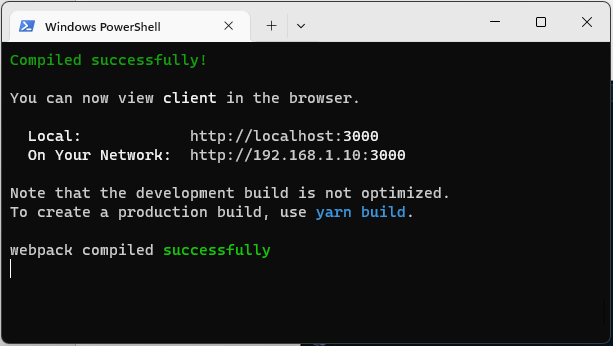
### Khởi tạo dự án

**Bước 1:** Mở terminal tại thư mục chứa mã frontend.

**Bước 2:** Chạy lệnh “yarn create react-app client” để tạo dự án với tên “client”.

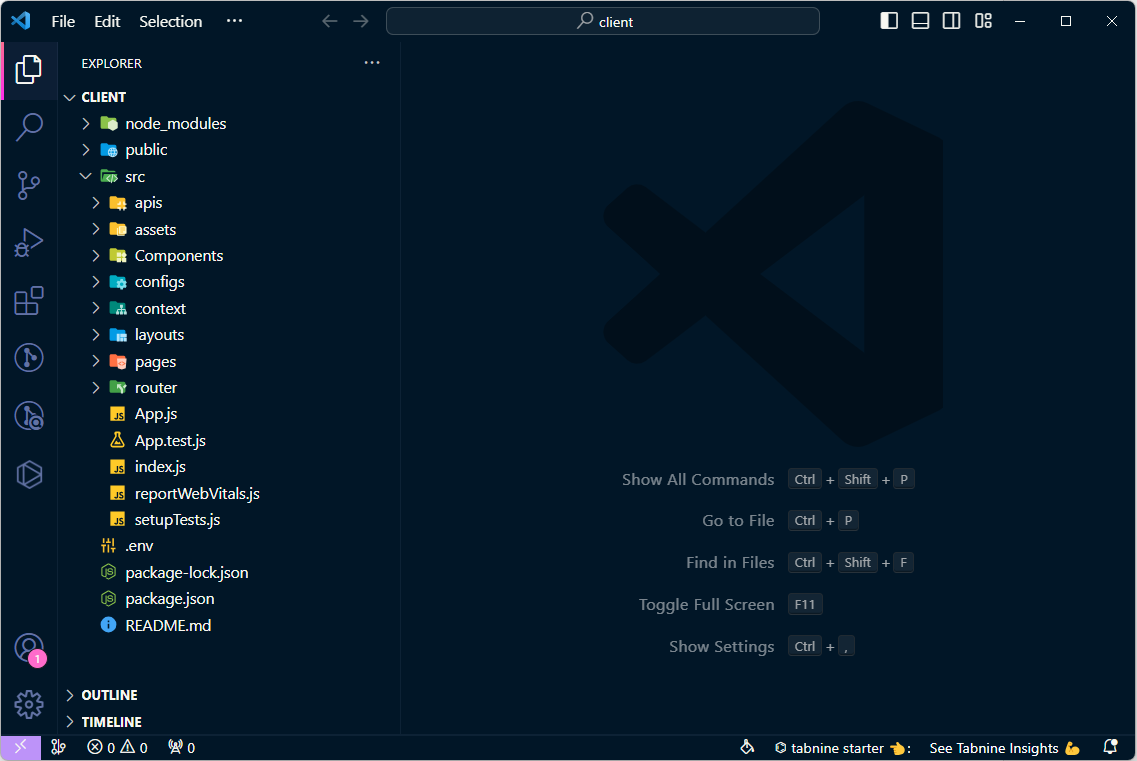
**Bước 3:** Sau khi khởi tạo xong, di chuyển vào thư mục của dự án bằng cách chạy lệnh “cd client”.

**Bước 4:** “yarn start” để chạy dự án, mở trình duyệt web truy cập url sau để kiểm tra <http://localhost:3000/> hoặc <http://127.1.1.1:3000>



Hình 3.11 Chạy dự án ReactJS thành công

### Cấu trúc thư mục



Hình 3.12 Cấu trúc thư mục frontend của website Nowbuys

**node\_modules:** Chứa các thư viện, phụ thuộc cần thiết được cài đặt và mặc định của ReactJS.

**public:** Thư mục chứa tập tin index.html duy nhất làm neo cho các mã Javascript thực thi và kết xuất giao diện vào đấy.

**src:** Đây là thư mục làm việc chính và toàn bộ mã được viết ra sẽ được lưu trữ trong thư mục này.

**src / apis:** Thư mục chứa các định tuyến đến server, thực hiện viện cấu hình yêu cầu và định dạng dữ liệu phản hồi cho frontend.

**src / assets:** Chứa các dữ liệu tĩnh như hình ảnh, icon cố định hoặc ít phải thay đổi.

**src / components:** Chứa các thành phần của website được sử dụng ở nhiều nơi được viết ở đây để tái sử dụng.

**src / configs:** Cấu hình các thư viện đăng nhập như Axios, Firebase, …

**src / contexts:** Đây là nơi chứa các context (các thành phần nắm dữ liệu hoặc trung gian truyền dữ liệu cho toàn bộ website) như thông tin người dùng, giỏ hàng, các context điều khiển khác, …

**src / layouts:** Chứa các loại bố cục màn hình khác nhau như màn hình có header - footer, màn hình đơn giản như màn hình đăng nhập, …

**src / pages:** Chứa các component chịu trách nhiệm kết suất giao diện cho một trang nào đó cụ thể như trang chủ, trang sản phẩm, trang chi tiết sản phẩm, trang giỏ hàng, trang đặt hàng, …

**src / router:** Chứa file định tuyến cho frontend nhận biết và kết xuất giao diện tương ứng.

**App.js:** Tệp mã gốc nơi bắt đầu cho toàn bộ mã được viết phía trên còn được gọi là tệp khởi động.

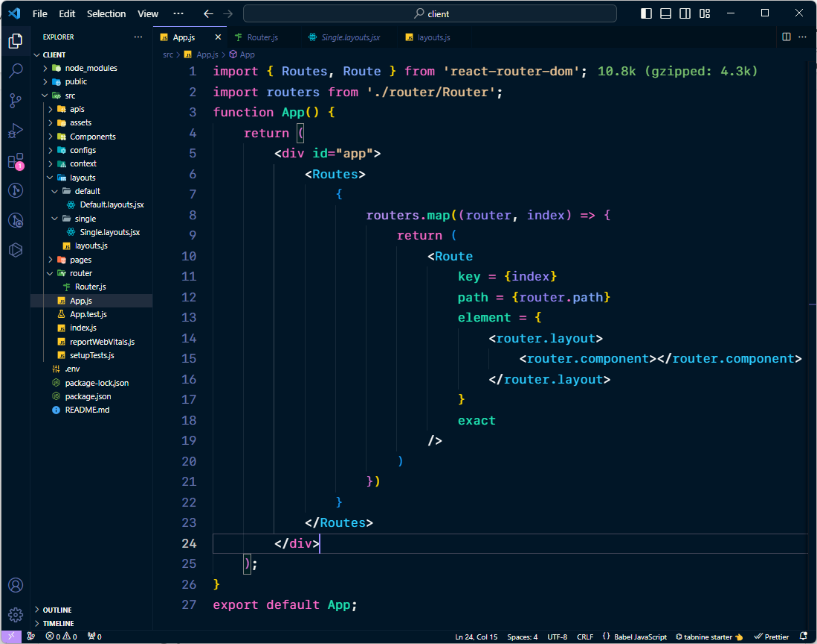
**index.js:** Là tệp cầu nối giữa App.js với trình duyệt web.

**.env:** Nơi chứa các biến môi trường. Thường chứa các key quan trọng liên quan nhiều đến yếu tố bảo mật, ngoài ra khi sử dụng biến ở tệp này về lâu dài khi dự án được triển khai ta cần thay đổi giá trị nào đó ta chỉ cần thay đổi ở tệp này mà không phải chạy lại toàn bộ dự án.

### Viết các mã quan trọng

Một số mã có vai trò quan trọng trong cấu trúc dự án với ReactJS

#### Mã khởi động



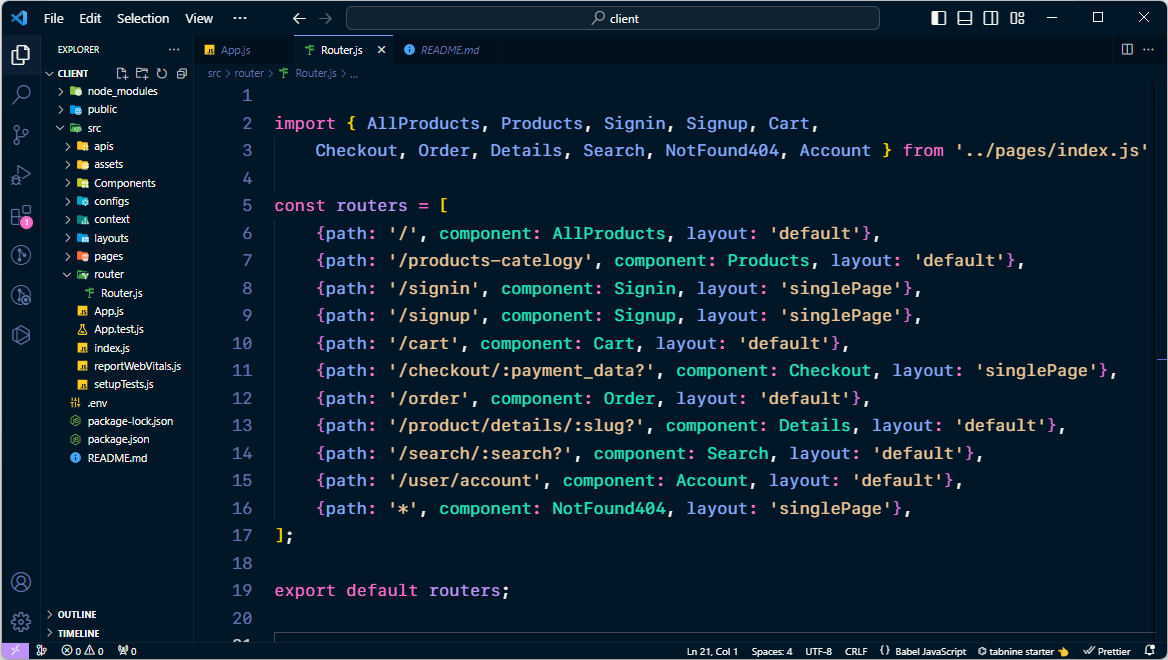
Hình 3.13 Mã khởi động trong tệp App.js

Dòng 1: Thêm thư viện “react-router-dom” để quản lý tuyến cho website.

Dòng 2: Thêm biến routers chứa các định tuyến đã được thiết lập trong tệp Router.js.

Dòng 3 – 26: Tạo hàm App để kết xuất các định tuyến ra giao diện.

#### Mã định tuyến



Hình 3.14 Mã định tuyến trong dự án ReactJS

Dòng 2 – 3: Thêm các trang được viết dưới dạng “function component”.

Dòng 5 – 17: Định nghĩa các tuyến và chỉ định các trang sẽ được kết xuất tương ứng.

Dòng 19: Xuất biến routers để sử dụng ở tệp khởi động App.js.

### Xây dựng các giao diện đặc thù của website Nowbuys

#### Xây dựng thành phần giao diện đăng nhập

A screen shot of a computer

Description automatically generated

Hình 3.15 Mã nhập tên đăng nhập trong trang đăng nhập Nowbuys

Dòng 2 – 6: Xử lý trải nghiệm người dùng, khi người dùng nhập tên đăng nhập vào ô tên đăng nhập thì lập tức hiển thị đoạn văn bản ngắn ở phía trên cho người dùng luôn biết họ đang làm gì với ô nhập liệu này.

Dòng 7 – 19: Thiết lập xử lý với thuộc tính của thẻ input trong ReactJS điển hình là thuộc tính “onChange” mỗi khi dữ liệu được nhập vào thì thuộc tính này sẽ được gọi và nhận giá trị của thẻ input, xử lý cho nó trở thành nơi nhận tên đăng nhập khi người dùng nhập vào và lưu trữ vào biến signinForm qua state setSigninForm. Thuộc tính “onKeyDown” mỗi khi có thao tác phím bên trong thẻ input thì thuộc tính này sẽ được gọi và nhận giá trị trong đó có mã của phím vừa nhấn, xử lý cho nó bắt được phím enter của người dùng khi họ nhập tên đăng nhập xong.

Dòng 20 – 36: Xử lý hiển thị lỗi khi người dùng nhập dữ liệu vào chưa hợp lệ.

A computer screen shot of a program code

Description automatically generated

Hình 3.16 Mã xử lý nút đăng nhập trên giao diện đăng nhập Nowbuys

Dòng 2: Biến lưu trạng thái đang xử lý / xử lý hoàn tất của xử lý đăng nhập để hiển thị lên giao diện tương ưng với việc hiển thị hiệu ứng xử lý / nút đăng nhập.

Dòng 4 – 6: Thẻ hiển thị hiệu ứng đang xử lý lên giao diện.

Dòng 8 – 12: Nút đăng nhập và xử lý khi người dùng nhấn đăng nhập qua thuộc tính của thẻ button ở dòng 11.

#### Xây dựng giao diện thẻ sản phẩm

A screen shot of a computer program

Description automatically generated

Hình 3.17 Xây dựng giao diện thẻ sản phẩm website Nowbuys

Dòng 1: Thẻ hiển thị tên sản phẩm.

Dòng 2 – 4: Hiển thị ảnh sản phẩm.

Dòng 8: Thẻ hiển thị giá sản phẩm sau khi giảm giá.

Dòng 11: Thẻ hiển thị giá trị gốc của sản phẩm.

Dòng 14 – 16: Hiển thị mô tả ngắn về sản phẩm.

Dòng 18 – 54: Xử lý số lượt sao dánh giá từ 1 đến 5 của người dùng về.

# KẾT QUẢ NGHIÊN CỨU

## Server website Nowbuys viết bằng ExpressJS framework

### Tuyến đăng nhập

A screenshot of a computer

Description automatically generated

Hình 4.1 Kiểm tra tuyến đăng nhập bằng Postman

Gửi yêu cầu đến tuyến đăng nhập theo địa chỉ: <http://localhost:8000/api/auth/log-in/local> với dữ liệu gửi kèm là tên đăng nhập (username) và mật khẩu (password) được đính kèm vào phần body của truy vấn http.

Dữ liệu phản hồi 200 cho thấy việc truy vấn đã thành công. Thuộc tính “is\_login: true” cho thấy thông tin đăng nhập đã chính xác.

Kiểm tra cookie cho thấy có một session mã hoá đã được lưu trữ.

A screenshot of a computer

Description automatically generated

Hình 4.2 Session được lưu sau khi đăng nhập thành công

### Tuyến lấy thông tin chi tiết của một sản phẩm

A screenshot of a computer

Description automatically generated

Hình 4.3 Lấy thông tin chi tiết của sản phẩm bằng Postman

Gửi yêu cầu đến tuyến đăng nhập theo địa chỉ: [http://localhost:8000/api /products/details/get?slug=iphone-14-pro-max-nobw00-phone](http://localhost:8000/api%20/products/details/get?slug=iphone-14-pro-max-nobw00-phone) với yêu cầu này dữ liệu không được đính kèm vào phần thân của yêu cầu mà sẽ được đính vào địa chỉ của yêu cầu theo kiểu thông số (params).

### Tuyến đặt hàng bằng

A screenshot of a computer

Description automatically generated

Hình 4.4 Sử dụng Postman kiểm tra tuyến đặt hàng

Cũng giống như việc yêu cầu tuyến đăng nhập, dữ liệu cũng được truyền qua phần thân của yêu cầu để backend kiểm tra tính chính xác, hợp lệ của dữ liệu và tiến hành tính toán các thông tin như tổng giá trị đơn hàng, tổng giá trị thanh toán.

## Website Nowbuys viết bằng thư viện ReactJS

### Trang đăng ký

A screenshot of a computer

Description automatically generated

Hình 4.5 Giao diện trang đăng ký người dùng của website Nowbuys

Ở trang đăng ký người dùng yêu cầu các thông tin họ và tên, email, tên đăng nhập, mật khẩu đăng nhập.

Ngoài ra còn cho phép đăng ký người dùng bằng tài khoản của bên thứ ba có uy tín và quy mô lớn gồm: Google, Facebook, Microsoft.

Khi dữ liệu được nhập vào không hợp lệ sẽ có thông báo cụ thể đến vị trí đó và hiển thị đỏ cảnh báo người dùng, giúp cho họ dễ dàng hơn trong việc nhập liệu.

### Trang đăng nhập

A screenshot of a computer

Description automatically generated

Hình 4.6 Giao diện trang đăng nhập của website Nowbuys

Cho phép đăng nhập bằng tài khoản Nowbuys, cho phép đăng nhập bằng tài khoản của bên thứ ba tương tự như trang đăng ký.

### Trang chi tiết sản phẩm

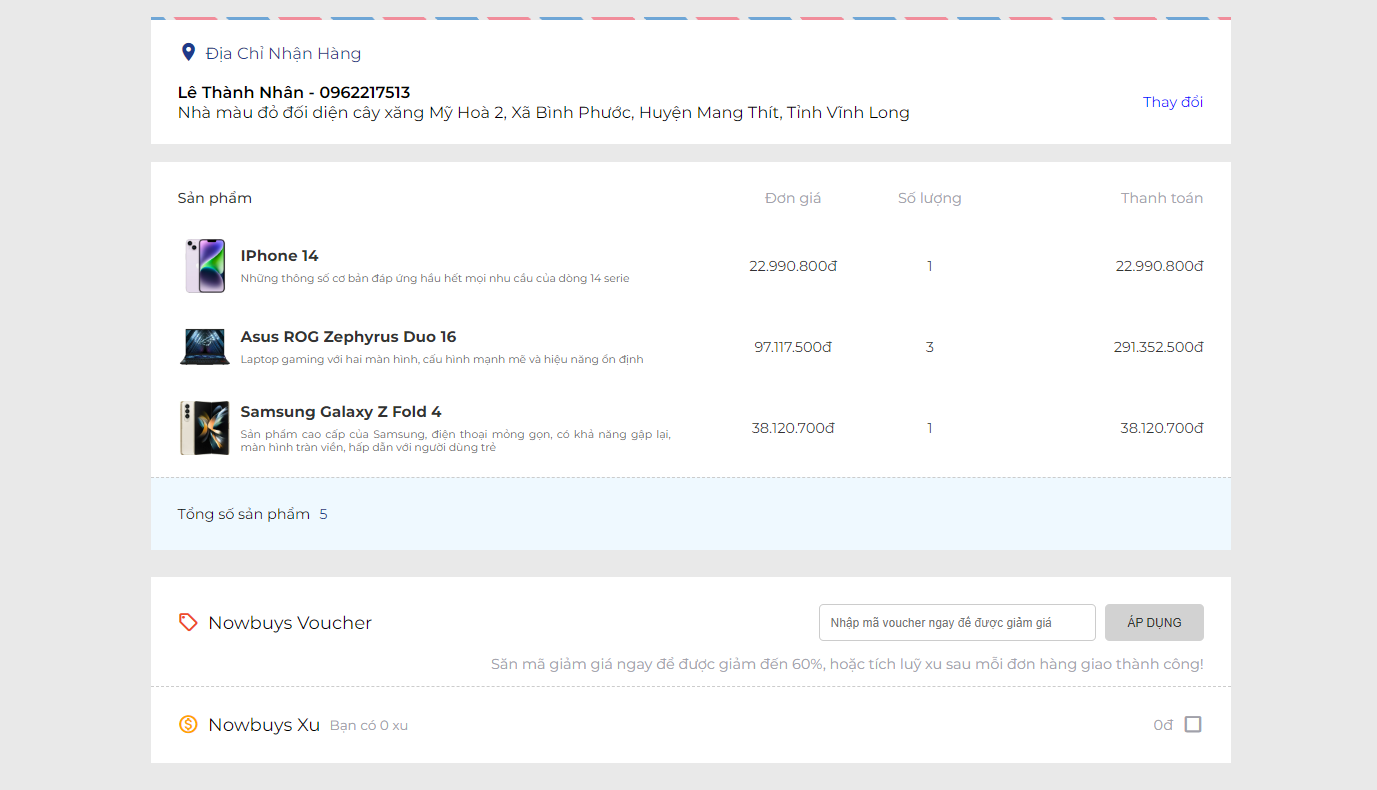
A screenshot of a computer

Description automatically generated

Hình 4.7 Trang thông tin chi tiết sản phẩm

Trang thông tin chi tiết sản phẩm có đầy đủ các thông tin về thông số kĩ thuật của sản phẩm, bài viết mô tả chi tiết, thống kê lượt đánh giá của sản phẩm, xem được các lượt đánh giá cụ thể của sản phẩm. Tại trang này cũng cho phép thêm sản phẩm vào giỏ hàng hoặc nhấn mua sản phẩm đó một cách nhanh chóng.

### Trang đặt hàng



Hình 4.8 Trang đặt hàng website Nowbuys

Trang đặt hàng cho phép chọn địa chỉ, đổi địa nhận hàng hoặc thêm địa chỉ, sửa địa chỉ tuỳ vào nhu cầu người đặt hàng ngay thời điểm đó. Xem được thông tin sản phẩm đặt mua rõ ràng cho người dùng dễ nắm bắt. Có tính năng áp dụng mã giảm giá, hoặc sử dụng xu tích luỹ. Có 3 lựa chọn thanh toán cho phép chọn.

# KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

## Kết luận

Lựa chọn nghiên cứu, kết hợp thành công thư viện ReactJS với framework ExpressJS. Sự kết hợp này cho website thương mại là một sự kết hợp cực kì tốt khi nó giải quyết được hầu hết các vấn đề quan trọng của một website thương mại điện tử yêu cầu và đảm bảo tính mới trong việc xây dựng website ở hiện tại.

### Framework ExpressJS

Hiểu được khái niệm giao thức HTTP, HTTPS và các phương thức của chúng: GET, POST, PUT, PATCH, DELETE, …

Hiểu được khái niệm về mô hình xử lý MVC (Model-View-Controller) và áp dụng thành công vào dự án.

Hiểu được middleware và vận dụng chúng vào các tuyến quan trọng như tuyến đăng nhập, tuyến đặt hàng, …

Đã nghiên cứu lựa chọn cấu trúc thư mục tối ưu cho mô hình MVC kết hợp với framework ExpressJS.

### Thư viện ReactJS:

Hiểu được các khái niệm về hook, router, component, Vitual DOM, …

Hiểu được logic xử lý của ReactJS, hiểu được cách các hook làm việc, …

Đã tìm hiểu và áp dụng tốt hook useContext để quản lý trạng thái lựa chọn chế độ sáng tối của người dùng, điều khiển đóng mở popup từ bất kì đâu đối với một component popup duy nhất.

Hiểu và kết hợp thành công, hiệu quả React với các thư viện khác nhằm phục vụ giải quyết vấn đề nào đó cho dự án. Điển hình như việc gửi yêu cầu và nhận phản hồi từ server bằng cách kết hợp các tính năng của React với thư viện Axios, không chỉ tái sử dụng component mà còn tái sử dụng được các thành phần giao tiếp với server.

Đã nghiên cứu lựa chọn cấu trúc thư mục tối ưu với kiến trúc và logic kết xuất của React. Các thành phần như component, layout, apis, config, router được viết riêng vào từng thư mục đảm bảo việc thành phần hoá chúng phù hợp với tiêu chí của một dự án React.

## Về mặt sản phẩm

Đã ứng dụng các công nghệ web hiện đại xây dựng thành công website bán hàng có đầy đủ các tính năng cơ bản của một website thương mại:

* Cho phép truy cập không cần đăng nhập.
* Đăng nhập bằng nhiều phương thức.
* Xem thông tin chi tiết các sản phẩm và số lượng / đánh giá của người dùng về sản phẩm đó.
* Giỏ hàng, trang đặt hàng, trang theo dõi trạng thái đơn hàng, …

Thiết kế website với màu sắc hài hoà thu hút, font chữ phù hợp / hiện đại, hình ảnh rõ ràng thực tế, thao tác dễ dàng dễ hiểu phù hợp với người Việt Nam khi có đến 90% người được khảo sát cho rằng thiết kế UI / UX của sản phẩm rất tốt.

## Hướng phát triển

Việc nghiên cứu công nghệ và xây dựng thành công website chỉ là một bước thành công trong chuỗi các bước cần thực hiện để đưa website vào sử dụng thực tế. Trong đó việc triển khai dự án lên internet là bước quan trọng tiếp theo. Lựa chọn máy chủ để triển khai hoặc có thể tự xây dựng riêng cho mình một máy chủ web, tìm hiểu hệ điều hành Ubuntu server, tìm hiểu máy chủ web Nginx, thuê tên miền để truy cập, … Đó là những việc cần làm trong bước tiếp theo cũng là hướng phát triển tiếp theo của nghiên cứu.

# DANH MỤC TÀI LIỆU THAM KHẢO

|  |  |
| --- | --- |
| [1] | Tạp chí công thương, “Xu hướng hành vi mua hàng trực tuyến của người tiêu dùng Việt Nam,” Bộ Công Thương, 12 4 2023. [Trực tuyến]. Available: https://tapchicongthuong.vn/bai-viet/xu-huong-hanh-vi-mua-hang-truc-tuyen-cua-nguoi-tieu-dung-viet-nam-104014.htm. [Đã truy cập 1 12 2023]. |
| [2] | survey, “Thống kê các công nghệ phổ biến năm 2023,” stackoverflow, 5 2023. [Trực tuyến]. Available: https://survey.stackoverflow.co/2023/#most-popular-technologies-language. [Đã truy cập 1 12 2023]. |
| [3] | Linkedin, “React.js năm 2023: Xu hướng và số liệu thống kê dành cho nhà phát triển Frontend,” Linkedin, 29 3 2023. [Trực tuyến]. Available: https://www.linkedin.com/pulse/reactjs-2023-trends-statistics-frontend-developers-abekus. [Đã truy cập 1 12 2023]. |
| [4] | geeksforgeeks, “Cách thức hoạt động của Web – Kiến trúc ứng dụng web cho người mới bắt đầu,” geeksforgeeks, 30 3 2023. [Trực tuyến]. Available: https://www.geeksforgeeks.org/how-web-works-web-application-architecture-for-beginners/. [Đã truy cập 1 12 2023]. |
| [5] | developer, “SPA (Single-page application),” mozilla, 8 4 2022. [Trực tuyến]. Available: https://developer.mozilla.org/en-US/docs/Glossary/SPA. [Đã truy cập 1 12 2023]. |
| [6] | React, “ReactJS,” Mã nguồn mở Meta, [Trực tuyến]. Available: https://react.dev/reference/react. [Đã truy cập 1 12 2023]. |
| [7] | StrongLoop, IBM, and other expressjs.com contribut, “ExpressJS,” ExpressJS, [Trực tuyến]. Available: https://expressjs.com/en/guide/routing.html. [Đã truy cập 1 12 2023]. |
| [8] | Geeksforgeeks, “Server Side Rendering và Client Side Rendering và Server Side Generation,” Geeksforgeeks, [Trực tuyến]. Available: https://www.geeksforgeeks.org/server-side-rendering-vs-client-side-rendering-vs-server-side-generation/. [Đã truy cập 1 12 2023]. |
| [9] | “WordPress.org,” [Trực tuyến]. Available: https://wordpress.org/documentation/. [Đã truy cập 28 11 2023]. |

# PHỤ LỤC

### Cài đặt môi trường NodeJS

Là một nền tảng mã nguồn mở được xây dựng trên Chrome's V8 JavaScript runtime để thực thi các ứng dụng viết bằng ngôn ngữ Javascript nói chung và ReactJS, ExpressJS nói riêng.

**Bước 1:** Truy cập <https://nodejs.org/en/download> để lựa chọn phiên bản phù hợp với thiết bị đang sử dụng, tải về và cài đặt.

A screenshot of a computer

Description automatically generated

Hình 0.1 Cài đặt NodeJS

**Bước 2:** Kiểm tra NodeJS đã cài đặt thành công trên máy và phiên bản bằng cách vào terminal chạy lệnh “node –version” hoặc “node -v”, nếu thấy dòng thông báo như sau hoặc tương tự thì NodeJS đã được cài đặt thành công “v20.9.0”.

A screenshot of a computer

Description automatically generated

Hình 0.2 Kiểm tra phiên bản NodeJS

### Cài đặt Laragon và tạo máy chủ ảo quản lý cơ sở dữ liệu mySQL bằng PhpMyAdmin.

Là một môi trường phát triển web (WAMP - Windows, Apache, MySQL, PHP) dành cho hệ điều hành Windows. Phần mềm này cung cấp công cụ thực hiện quản lý cơ sở dữ liệu mySQL thông qua giao diện PhpMyAdmin trên nền tảng Web.

**Bước 1:** Tải và cài đặt từ <https://laragon.org/download/>

**Bước 2:** Thiết lập máy chủ ảo, cơ sở dữ liệu và cổng như sau:

A screenshot of a computer

Description automatically generated

Hình 0.3 Thiết lập máy chủ ảo trên Laragon

**Bước 3:** Thêm PhpMyAdmin vào Laragon:

Tải PhpMyAdmin từ địa chỉ: <https://www.phpmyadmin.net/downloads>

Giải nén vào thư mục: C:\laragon\etc\apps

**Bước 4:** Nhấp chuột trái vào nút “Start All” để chạy máy chủ ảo sau đó nhấp chuột trái vào nút Database để khởi chạy PhpMyAdmin. Vào trình duyệt web truy cập địa chỉ sau để sử dụng MySQL: <http://localhost:8080/phpmyadmin/>

*A screenshot of a computer

Description automatically generated*

Hình 0.4 Giao diện chính của Laragon

### Cài đặt phần mềm viết mã Visual Studio Code

Là phần mềm soạn thảo mã, được Microsoft phát triển. Có dung lượng nhẹ, hỗ trợ tích hợp mạnh mẽ với một loạt các tiện ích mở rộng phục vụ nhiều tính năng tiện ích như kiểm tra mã nguồn, tự động hoàn tất mã, và tích hợp với hệ thống kiểm soát phiên bản.

**Bước 1:** Truy cập vào link sau để tải: <https://code.visualstudio.com/download>

**Bước 2:** Cài đặt phần mềm:

A screenshot of a computer program

Description automatically generated

Hình 0.5 Giao diện sau khi cài đặt thành công Visual Studio Code

**Bước 3:** Cài đặt, thiết đặt các công cụ mở rộng (extentions) (Tuy là phần mềm phổ biến và thân thiện với hầu hết các lập trình viên hiện tại, tuy nhiên đây chỉ là trình soạn thảo mã (code editor) vì thế có thể dùng để viết mã cho hầu hết các ngôn ngữ lập trình phổ biến nhưng với mỗi ngôn ngữ bạn phải tự mình thiết lập những cài đặt riêng để thực thi mã, đồng thời với mỗi ngôn ngữ sẽ có định dạng và gợi ý mã khác nhau nên cần phải cài đặt thêm các extendsion phù hợp để việc viết mã trở nên tiện lợi và nhanh chóng) chọn vào biểu tượng 4 vuông xếp thứ năm từ trên xuống ở thanh công cụ bên cạnh phải màn hình, nhập tên extentions vào ô tìm kiếm và chọn install để cài đặt.

A screenshot of a computer

Description automatically generated

Hình 0.6 Giao diện cài đặt extentions trong Visual Studio Code

### Cài đặt công cụ Postman kiểm tra API

Là công cụ phần mềm phổ biến nhất dùng để thử nghiệm các API. Có thể gọi Rest API và nhận phản hồi (response) mà không cần viết mã lập trình. Hỗ trợ tất cả các phương thức HTTP (GET, POST, PUT, PATCH, DELETE, …). Bên cạnh đó, nó còn cho phép lưu lại lịch sử các lần yêu cầu (request), rất tiện cho việc sử dụng lại khi cần.

**Bước 1:** Tải và cài đặt Postman từ <https://www.postman.com/downloads/>

**Bước 2:** Tạo tài khoản để sử dụng Postman.

**Bước 3:** Thiết lập Postman làm việc phân cấp từ workspace > collection > request. Workspace tương đương với nhóm làm việc, mỗi collection sẽ phục vụ cho một backend, mỗi request tương ứng với mỗi yêu cầu đến server của frontend.

A screenshot of a computer

Description automatically generated

Hình 0.7 Giao diện sau khi thiết lập Postman