**TRƯỜNG ĐẠI HỌC VĂN LANG**

**KHOA CÔNG NGHỆ THÔNG TIN**

***A red and white logo

AI-generated content may be incorrect.***

**ĐỒ ÁN CUỐI KỲ**

**MÔN HỌC LẬP TRÌNH WEB NÂNG CAO**

***Chủ đề:***

**XÂY DỰNG GIAO DIỆN WEB QUẢN TRỊ CỬA HÀNG THỜI TRANG – ATOM SHOP**

**SVTH: Trần Minh Phúc – 227480201694**

**Châu Gia Kiệt – 2274802010449**

**Võ Đình Ngọc Bình – 2274802010066**

**Nguyễn Hoàng Khanh - 227480201393**

**LHP: 242\_71ITSE41403\_0401**

**GVHD: Trần Vũ Đại**

**TP. Hồ Chí Minh – 4/2025**

MỤC LỤC

[LỜI CẢM ƠN 1](#_Toc195486442)

[PHẦN I: MỞ ĐẦU 2](#_Toc195486443)

[1. Tổng Quan 2](#_Toc195486444)

[1.1 Bối cảnh hiện tại 2](#_Toc195486445)

[1.2 Lý do chọn đề tài 2](#_Toc195486446)

[1.3 Mục tiêu 2](#_Toc195486447)

[2. Giới Thiệu 3](#_Toc195486448)

[PHẦN II: NỘI DUNG 5](#_Toc195486449)

[1. Trang Quản Trị (Admin) 5](#_Toc195486450)

[1.1 Đăng nhập (Login) 6](#_Toc195486451)

[1.2 Thêm sản phẩm (Add Item) 7](#_Toc195486452)

[1.3 Danh sách (List) 8](#_Toc195486453)

[1.4 Danh sách đặt hàng (Order) 9](#_Toc195486454)

[2. Trang Người Dùng (User) 10](#_Toc195486455)

[2.1 Trang chủ (Home) 11](#_Toc195486456)

[2.2 Bộ sưu tập (Collection) 13](#_Toc195486457)

[2.3 Tin tức (Blog) 14](#_Toc195486458)

[2.4 Liên hệ (Contact) 15](#_Toc195486459)

[2.5 Giỏ hàng (Cart) 15](#_Toc195486460)

[2.6 Đăng nhập (Login) 16](#_Toc195486461)

[2.7 Đặt hàng (Order) 17](#_Toc195486462)

[2.8 Xác nhận đơn hàng (PlaceOrder) 19](#_Toc195486463)

[2.9 Xác thực (Verify) 20](#_Toc195486464)

[2.10 Chi tiết sản phẩm (Product) 21](#_Toc195486465)

[2.11 Các thành phần (Component) 22](#_Toc195486466)

[3. Hệ Thống Xử Lý Nghiệp Vụ (BackEnd) 27](#_Toc195486467)

[3.1 Cấu trúc thư mục 27](#_Toc195486468)

[3.2 Kết nối cơ sở dữ liệu bằng Docker 27](#_Toc195486469)

[3.3 Quản lý dữ liệu bằng TablePlus 28](#_Toc195486470)

[3.4 Tích hợp Cloudinary 28](#_Toc195486471)

[PHẦN III: KẾT LUẬN 30](#_Toc195486472)

[1. Kết quả đạt được 30](#_Toc195486473)

[2. Kiến thức và kỹ năng đạt được 30](#_Toc195486474)

[3. Định hướng phát triển 30](#_Toc195486475)

[1. Chức năng mở rộng: 30](#_Toc195486476)

[2. Nâng cấp kỹ thuật: 30](#_Toc195486477)

[PHẦN IV: PHỤ LỤC 30](#_Toc195486478)

[1. Phụ Lục 30](#_Toc195486479)

[2. Phân Công 30](#_Toc195486480)

[TÀI LIỆU THAM KHẢO 30](#_Toc195486481)

**DANH MỤC HÌNH ẢNH**

[Hình 1. Script chạy trang Admin 5](#_Toc195486482)

[Hình 2. Giao diện ban đầu 5](#_Toc195486483)

[Hình 3. Giao diện đăng nhập 6](#_Toc195486484)

[Hình 4. Giao diện sau khi đăng nhập quản trị 7](#_Toc195486485)

[Hình 5. Giao diện thêm sản phẩm 8](#_Toc195486486)

[Hình 6. Giao diện danh sách sản phẩm 9](#_Toc195486487)

[Hình 7. Giao diện quản lý đơn hàng 10](#_Toc195486488)

[Hình 8. Giao diện trang chủ 11](#_Toc195486489)

[Hình 9. Giao diện trang chủ 12](#_Toc195486490)

[Hình 10. Giao diện trang chủ 13](#_Toc195486491)

[Hình 11. Giao diện trang bộ sưu tập 14](#_Toc195486492)

[Hình 12. Giao diện trang tin tức 15](#_Toc195486493)

[Hình 13. Giao diện trang giỏ hàng 16](#_Toc195486494)

[Hình 14. Giao diện trang đăng nhập 17](#_Toc195486495)

[Hình 15. Trang xác nhận đặt hàng 19](#_Toc195486496)

[Hình 16. Trang điền thông tin thẻ 20](#_Toc195486497)

[Hình 17. Giao diện trang xác thực 21](#_Toc195486498)

[Hình 18. Giao diện trang chi tiết sản phẩm 22](#_Toc195486499)

[Hình 19. Giao diện container MongoDB đang chạy trong Docker Desktop 28](#_Toc195486500)

[Hình 20. API Key trên Cloudinary được sử dụng để cấu hình upload ảnh 29](#_Toc195486501)

# LỜI CẢM ƠN

Trước hết, nhóm xin bày tỏ lòng biết ơn sâu sắc đến Thầy **Trần Vũ Đại** – người đã tận tình giảng dạy, hướng dẫn và đồng hành cùng chúng em trong suốt quá trình học tập cũng như thực hiện bài báo cáo này. Sự quan tâm, hỗ trợ và những chỉ dẫn quý báu của Thầy chính là động lực quan trọng giúp chúng em hoàn thành tốt nhiệm vụ được giao.

Nhóm cũng xin gửi lời cảm ơn chân thành đến quý Thầy, Cô trong Khoa đã xây dựng một chương trình học thiết thực, cập nhật và phù hợp với thực tiễn. Nhờ đó, chúng em đã có cơ hội tiếp cận và vận dụng những công nghệ lập trình web hiện đại, cũng như tích lũy được nhiều kiến thức nền tảng và chuyên sâu trong lĩnh vực phát triển ứng dụng web – hành trang quý báu cho chặng đường nghề nghiệp sắp tới.

Mặc dù bài báo cáo đã được thực hiện với tinh thần trách nhiệm và sự nỗ lực cao nhất, nhưng do hạn chế về thời gian và kinh nghiệm thực tế, nhóm không tránh khỏi những thiếu sót. Chúng em kính mong nhận được sự góp ý, nhận xét từ Thầy để có thể tiếp tục hoàn thiện và nâng cao kỹ năng chuyên môn trong các dự án tương lai.

Xin chân thành cảm ơn!

# PHẦN I: MỞ ĐẦU

## 1. Tổng Quan

### 1.1 Bối cảnh hiện tại

Trong thời đại công nghệ số phát triển mạnh mẽ, xu hướng mua sắm trực tuyến các thiết bị điện tử như tai nghe, điện thoại, đồng hồ thông minh, phụ kiện công nghệ đang ngày càng phổ biến. Tuy nhiên, nhiều cửa hàng kinh doanh nhỏ lẻ vẫn gặp khó khăn trong việc quản lý sản phẩm, đơn hàng và khách hàng một cách hiệu quả – thường chỉ ghi chép thủ công hoặc qua các công cụ rời rạc.

### 1.2 Lý do chọn đề tài

Lý do nhóm chọn đề tài “ATOM SHOP – Xây dựng hệ thống web bán hàng và quản trị sản phẩm công nghệ” xuất phát từ mong muốn áp dụng những kiến thức đã học vào một dự án thực tiễn, có tính ứng dụng cao. Thông qua việc xây dựng hệ thống này, nhóm không chỉ rèn luyện kỹ năng lập trình, thiết kế giao diện và quản lý cơ sở dữ liệu, mà còn tiếp cận với quy trình phát triển phần mềm hoàn chỉnh, từ phân tích yêu cầu đến triển khai sản phẩm. Bên cạnh đó, lĩnh vực kinh doanh sản phẩm công nghệ là một thị trường tiềm năng, có nhu cầu cao về sự chuyên nghiệp và tự động hóa trong vận hành, nên nhóm kỳ vọng hệ thống sẽ mang lại trải nghiệm tốt cho người dùng và hỗ trợ hiệu quả cho quản trị viên.

### 1.3 Mục tiêu

Mục tiêu của đề tài là xây dựng một hệ thống website bán hàng thiết bị công nghệ có đầy đủ chức năng thực tế, đáp ứng nhu cầu sử dụng của cả khách hàng và quản trị viên thông qua một nền tảng hiện đại, dễ sử dụng. Cụ thể, dự án hướng đến các mục tiêu chính sau:

* **Phát triển giao diện người dùng (frontend)** dành cho khách hàng, với các chức năng:
  + Duyệt danh sách sản phẩm và lọc theo danh mục.
  + Xem chi tiết sản phẩm, lựa chọn màu sắc, thêm vào giỏ hàng.
  + Thực hiện đặt hàng và thanh toán trực tuyến (hỗ trợ COD và Stripe).
  + Đăng nhập và theo dõi lịch sử mua hàng, trạng thái vận chuyển.
* **Xây dựng hệ thống quản trị (admin)** dành cho chủ cửa hàng, bao gồm:
  + Đăng nhập với tài khoản quản trị viên.
  + Thêm mới sản phẩm, upload hình ảnh và cập nhật thông tin chi tiết.
  + Quản lý danh sách sản phẩm, xóa những sản phẩm không còn kinh doanh.
  + Theo dõi và cập nhật trạng thái xử lý đơn hàng (Packing, Shipped, Delivered,...).
* **Đảm bảo kiến trúc hệ thống rõ ràng**, tổ chức mã nguồn theo hướng tách biệt giữa frontend, admin và backend, giúp dễ dàng trong việc quản lý, phát triển và bảo trì hệ thống.
* **Ứng dụng công nghệ hiện đại** trong toàn bộ dự án:
  + **Frontend & Admin**: ReactJS, Tailwind CSS, Axios, React Router DOM
  + **Backend**: NodeJS, ExpressJS, MongoDB, JWT, Stripe, Cloudinary
* **Rèn luyện kỹ năng làm việc nhóm**: phân chia chức năng hợp lý, sử dụng GitHub để quản lý mã nguồn và phối hợp hiệu quả trong quá trình phát triển.
* **Mô phỏng sát với thực tế** quy trình vận hành của một cửa hàng điện tử online: từ việc quản lý sản phẩm đến đặt hàng, thanh toán và theo dõi trạng thái đơn hàng.

## 2. Giới Thiệu

ATOM SHOP là hệ thống website thương mại điện tử chuyên kinh doanh thiết bị công nghệ, được thiết kế với đầy đủ hai phần chức năng chính: giao diện người dùng (frontend) dành cho khách hàng và giao diện quản trị (admin) dành cho chủ cửa hàng.

Trang người dùng cho phép khách hàng duyệt các sản phẩm công nghệ như tai nghe, đồng hồ thông minh, điện thoại, chuột, loa,... với các tính năng như tìm kiếm, lọc theo danh mục, thêm vào giỏ hàng, đặt hàng và thanh toán trực tuyến qua COD hoặc Stripe. Người dùng có thể đăng nhập để theo dõi lịch sử đặt hàng và trạng thái đơn hàng.

Trang quản trị dành cho admin hỗ trợ đăng nhập hệ thống, thêm mới và quản lý sản phẩm, kiểm soát danh sách đơn hàng, cập nhật trạng thái giao hàng theo thời gian thực. Tất cả được thực hiện trên giao diện hiện đại, responsive và thân thiện với người dùng.

Hệ thống được phát triển với kiến trúc tách biệt ba phần:

* **Frontend**: Xây dựng bằng **ReactJS**, sử dụng thư viện **React Router DOM** cho điều hướng, **Axios** để gọi API, và **Context API** để quản lý trạng thái ứng dụng.
* **Admin**: Cũng sử dụng **ReactJS**, với thiết kế tối ưu cho thao tác quản trị. Thư viện **react-icons**, **react-toastify** và các thành phần UI hiện đại được tích hợp nhằm tăng trải nghiệm người dùng.
* **Backend**: Phát triển bằng **NodeJS** và **ExpressJS**, kết nối với **MongoDB** để lưu trữ dữ liệu. Hệ thống hỗ trợ các chức năng xác thực (JWT), phân quyền, xử lý thanh toán (Stripe), và cung cấp RESTful API cho frontend và admin.

Trong quá trình thực hiện, nhóm đã trải qua các giai đoạn: phân tích hệ thống, xây dựng giao diện, lập trình backend, kết nối API và kiểm thử. Dự án không chỉ giúp nhóm rèn luyện kỹ năng lập trình và quản lý dự án, mà còn mô phỏng tương đối đầy đủ một quy trình triển khai website bán hàng thực tế, hướng đến ứng dụng thực tiễn.

# PHẦN II: NỘI DUNG

## 1. Trang Quản Trị (Admin)

Link code: <https://github.com/TranMinhPhuc04/ATOM_SHOP/tree/main/admin>

- Trong phần giao diện quản trị của hệ thống ATOM SHOP, nhóm đã xây dựng đầy đủ các chức năng quản lý sản phẩm và đơn hàng, đảm bảo luồng thao tác tiện lợi cho admin. Các chức năng đều được xây dựng bằng ReactJS, giao diện thiết kế với Tailwind CSS, và kết nối với server thông qua axios. Ngoài ra, thư viện react-toastify được sử dụng để hiển thị thông báo tương tác rõ ràng, tăng trải nghiệm người dùng.

- Script để chạy source trang quản trị: ***npm run dev*** (local: <http://localhost:5174/>)

**A screen shot of a computer

AI-generated content may be incorrect.**

Hình 1. Script chạy trang Admin

- Kết quả: hiển thị giao diện ban đầu

**A screenshot of a login page

AI-generated content may be incorrect.**

Hình 2. Giao diện ban đầu

### 1.1 Đăng nhập (Login)

Link code: <https://github.com/TranMinhPhuc04/ATOM_SHOP/blob/main/admin/src/pages/Login.jsx>

- **Mô tả chức năng:**

Hiển thị form đăng nhập dành cho quản trị viên. Sau khi nhập email và mật khẩu, hệ thống sẽ gửi yêu cầu đến server để xác thực. Nếu thông tin hợp lệ, người dùng được cấp quyền truy cập vào trang quản trị và lưu token để duy trì phiên đăng nhập.

**- Kỹ thuật áp dụng:**

* useState: lưu trạng thái form.
* axios: gửi yêu cầu xác thực đến backend.
* toast: hiển thị thông báo.
* Tailwind CSS: thiết kế form đẹp, responsive.
* setToken: lưu token sau khi đăng nhập thành công.

A screenshot of a login box

AI-generated content may be incorrect.

Hình 3. Giao diện đăng nhập

- Kết quả:

A screenshot of a computer

AI-generated content may be incorrect.

Hình 4. Giao diện sau khi đăng nhập quản trị

### 1.2 Thêm sản phẩm (Add Item)

Link code: <https://github.com/TranMinhPhuc04/ATOM_SHOP/blob/main/admin/src/pages/Add.jsx>

- **Mô tả chức năng:**

Cho phép admin nhập thông tin sản phẩm mới như:

* Tên (Product Name)
* Mô Tả (Product Description)
* Danh Mục (Category)
* Giá (Product Price)
* Màu Sắc (Product Colors)
* Hình Ảnh (Image)
* Độ Phổ Biến (Add to Popular)

Sau đó gửi lên server để lưu trữ. Giao diện hỗ trợ tải lên tối đa 4 ảnh.

**- Kỹ thuật áp dụng:**

* useState: lưu trạng thái form và ảnh.
* FormData: gửi dữ liệu text + ảnh lên server.
* axios: gửi POST request kèm token xác thực.
* toast: thông báo kết quả thao tác.
* Tailwind CSS: dựng layout form hiện đại.

A screenshot of a computer

AI-generated content may be incorrect.

Hình 5. Giao diện thêm sản phẩm

### 1.3 Danh sách (List)

Link code: <https://github.com/TranMinhPhuc04/ATOM_SHOP/blob/main/admin/src/pages/List.jsx>

- **Mô tả chức năng:**

Hiển thị toàn bộ sản phẩm trong hệ thống để admin dễ dàng theo dõi. Cho phép xoá sản phẩm trực tiếp thông qua biểu tượng thùng rác.

**- Kỹ thuật áp dụng:**

* useEffect: gọi API khi trang được tải.
* useState: lưu danh sách sản phẩm.
* axios: lấy/xoá sản phẩm.
* toast: thông báo kết quả thao tác.
* react-icons: dùng icon TbTrash.
* Tailwind CSS: bố cục bảng sản phẩm.

A screenshot of a computer

AI-generated content may be incorrect.

Hình 6. Giao diện danh sách sản phẩm

### 1.4 Danh sách đặt hàng (Order)

Link code: <https://github.com/TranMinhPhuc04/ATOM_SHOP/blob/main/admin/src/pages/Orders.jsx>

- **Mô tả chức năng:**

Hiển thị danh sách các đơn hàng đã được đặt bởi khách hàng. Admin có thể xem chi tiết từng đơn, bao gồm sản phẩm, thông tin người nhận và trạng thái đơn hàng. Có thể cập nhật trạng thái đơn theo tiến độ xử lý.

**- Kỹ thuật áp dụng:**

* useEffect, useState: quản lý danh sách đơn hàng.
* axios: gửi yêu cầu lấy và cập nhật đơn hàng.
* toast: thông báo khi cập nhật trạng thái.
* react-icons: dùng icon TfiPackage.
* Tailwind CSS: bố cục thông tin đơn hàng.

A screenshot of a computer

AI-generated content may be incorrect.

Hình 7. Giao diện quản lý đơn hàng

## 2. Trang Người Dùng (User)

Link code: <https://github.com/TranMinhPhuc04/ATOM_SHOP/tree/main/frontend>

- Trong phần giao diện người dùng của hệ thống ATOM SHOP, nhóm đã phát triển đầy đủ các tính năng cần thiết cho trải nghiệm mua sắm trực tuyến như xem danh sách sản phẩm, chi tiết sản phẩm, giỏ hàng, thanh toán, theo dõi đơn hàng và đăng nhập người dùng. Giao diện bao gồm các trang chính như **Home** (trang chủ), **Collection** (bộ sưu tập sản phẩm), **Blog** (tin tức), **Contact** (liên hệ), **Cart** (giỏ hàng) và **Login** (đăng nhập), giúp người dùng dễ dàng tìm kiếm, quản lý đơn hàng và tương tác với hệ thống. Toàn bộ frontend được xây dựng bằng ReactJS kết hợp với Vite nhằm tối ưu tốc độ tải trang, sử dụng Tailwind CSS để thiết kế giao diện hiện đại và tương thích đa thiết bị. Dữ liệu được lấy và gửi qua server thông qua axios, đảm bảo luồng tương tác mượt mà. Hệ thống định tuyến được quản lý bằng React Router, cùng với context API để xử lý trạng thái toàn cục như token người dùng, thông tin giỏ hàng và URL backend.

A person wearing headphones and smiling

AI-generated content may be incorrect.

Hình 8. Giao diện trang chủ

### 2.1 Trang chủ (Home)

Link code: <https://github.com/TranMinhPhuc04/ATOM_SHOP/blob/main/frontend/src/pages/Home.jsx>

- **Mô tả chức năng:**

Trang chủ giới thiệu tổng thể về cửa hàng với các thành phần chính như banner, mô tả thương hiệu, danh sách sản phẩm mới và phổ biến. Đây là điểm chạm đầu tiên với người dùng nên được thiết kế hiện đại, thu hút và dễ điều hướng.

**- Kỹ thuật sử dụng:**

* ReactJS kết hợp useEffect để gọi dữ liệu sản phẩm từ API.
* axios: fetch danh sách sản phẩm nổi bật từ backend.
* Tailwind CSS: dựng layout responsive.
* react-router-dom: điều hướng khi người dùng bấm vào sản phẩm.

**- Component sử dụng:**

* Header.jsx: thanh menu điều hướng toàn trang.
* Hero.jsx: ảnh nền đầu trang, giới thiệu thương hiệu.
* Features.jsx: mô tả các ưu điểm như giao hàng nhanh, đổi trả dễ, thanh toán an toàn.
* NewArrivals.jsx: hiển thị các sản phẩm mới nhất từ server.
* PopularProducts.jsx: hiển thị các sản phẩm được đánh giá cao hoặc mua nhiều.
* Newsletter.jsx: khối nhập email nhận tin tức mới.
* Footer.jsx: chứa thông tin liên hệ, liên kết chính sách, mạng xã hội.

A person wearing headphones

AI-generated content may be incorrect.

Hình 9. Giao diện trang chủ

A screenshot of a website

AI-generated content may be incorrect.

Hình 10. Giao diện trang chủ

### 2.2 Bộ sưu tập (Collection)

Link code: <https://github.com/TranMinhPhuc04/ATOM_SHOP/blob/main/frontend/src/pages/Collection.jsx>

- **Mô tả chức năng:**

Trang hiển thị tất cả sản phẩm của cửa hàng. Cho phép người dùng duyệt qua các sản phẩm, chọn nhanh, và điều hướng tới chi tiết sản phẩm.

**- Kỹ thuật sử dụng:**

* useState để lưu danh sách sản phẩm.
* useEffect để gọi API sản phẩm (/api/product/list).
* axios để lấy dữ liệu từ backend.
* map() để hiển thị từng sản phẩm dưới dạng card.

**- Component sử dụng:**

* Header.jsx, Footer.jsx
* ProductItem.jsx: card sản phẩm gồm ảnh, tên, giá, màu sắc.
* BreadCrumb.jsx: đường dẫn breadcrumb giúp định hướng trang.
* Pagination.jsx: nếu có số lượng sản phẩm lớn (có thể mở rộng).

A screenshot of a computer device

AI-generated content may be incorrect.

Hình 11. Giao diện trang bộ sưu tập

### 2.3 Tin tức (Blog)

Link code: <https://github.com/TranMinhPhuc04/ATOM_SHOP/blob/main/frontend/src/pages/Blog.jsx>

- **Mô tả chức năng:**

Hiển thị các bài viết tin tức hoặc blog về sản phẩm, xu hướng mua sắm, hoặc hướng dẫn. Giúp tăng giá trị thương hiệu và SEO cho website.

**- Kỹ thuật sử dụng:**

* Sử dụng mảng bài viết tĩnh hoặc gọi API nếu backend có hỗ trợ blog.
* Dựng giao diện bằng Tailwind, chia lưới bài viết.
* Có thể mở rộng để click vào bài viết chi tiết.

**- Component sử dụng:**

* Header.jsx, Footer.jsx
* BlogItem.jsx: hiển thị từng bài viết gồm ảnh, tiêu đề và đoạn giới thiệu.
* Newsletter.jsx: đăng ký nhận tin tức mới.

A screenshot of a computer shop

AI-generated content may be incorrect.

Hình 12. Giao diện trang tin tức

### 2.4 Liên hệ (Contact)

Link code: <https://github.com/TranMinhPhuc04/ATOM_SHOP/blob/main/frontend/src/components/Navbar.jsx>

- **Mô tả chức năng:**

Cung cấp thông tin liên hệ và form để người dùng gửi ý kiến hoặc phản hồi đến cửa hàng.

### 2.5 Giỏ hàng (Cart)

Link code: <https://github.com/TranMinhPhuc04/ATOM_SHOP/blob/main/frontend/src/pages/Cart.jsx>

- **Mô tả chức năng:**

Hiển thị các sản phẩm người dùng đã thêm vào giỏ. Cho phép điều chỉnh số lượng, xoá sản phẩm và chuyển sang bước đặt hàng.

**- Kỹ thuật sử dụng:**

* useContext: lấy danh sách giỏ hàng từ StoreContext.
* useState: tính tổng giá đơn hàng và cập nhật số lượng.
* Tính toán tổng tiền tự động khi thay đổi giỏ hàng.

**- Component sử dụng:**

* Header.jsx, Footer.jsx
* CartItem.jsx: hiển thị từng sản phẩm đã chọn gồm ảnh, tên, màu, số lượng, giá và nút xoá.
* CartSummary.jsx: bảng tổng kết đơn hàng và nút “Tiến hành đặt hàng”.

A screenshot of a phone

AI-generated content may be incorrect.

Hình 13. Giao diện trang giỏ hàng

### 2.6 Đăng nhập (Login)

Link code: <https://github.com/TranMinhPhuc04/ATOM_SHOP/blob/main/frontend/src/pages/Login.jsx>

**- Mô tả chức năng:**

Cho phép người dùng đăng nhập bằng email và mật khẩu. Sau khi xác thực thành công, lưu token để thực hiện mua hàng, xem lịch sử đơn hàng.

**- Kỹ thuật sử dụng:**

* useState: lưu email và mật khẩu.
* axios: gửi POST đến API /api/user/login.
* react-toastify: hiển thị thông báo kết quả.
* navigate: điều hướng sang trang chính khi đăng nhập thành công.

**- Component sử dụng:**

* Header.jsx, Footer.jsx
* LoginForm.jsx: gồm input email, password và nút đăng nhập.
* ToastContainer: thông báo phản hồi đăng nhập.

A close up of a headphones

AI-generated content may be incorrect.

Hình 14. Giao diện trang đăng nhập

### 2.7 Đặt hàng (Order)

Link code: <https://github.com/TranMinhPhuc04/ATOM_SHOP/blob/main/frontend/src/pages/Orders.jsx>

**- Mô tả chức năng:**

Trang này hiển thị danh sách các đơn hàng đặt của người dùng. Mỗi đơn hàng được hiển thị theo từng sản phẩm đã mua, kèm theo thông tin chi tiết như tên sản phẩm, hình ảnh, giá, số lượng, màu sắc, phương thức thanh toán, trạng thái và ngày đặt hàng.

**- Kỹ thuật sử dụng:**

* useContext: lấy token, backendUrl, và đơn vị tiền tệ từ ShopContext.
* useEffect: gọi dữ liệu đơn hàng mỗi khi token thay đổi.
* axios: gửi POST request tới /api/order/userorders để lấy danh sách đơn hàng.
* Dữ liệu đơn hàng được xử lý để hiển thị chi tiết từng sản phẩm thuộc từng đơn.

**- Component sử dụng:**

* Title.jsx: tiêu đề “Order List” hiển thị ở đầu trang với định dạng rõ ràng.
* Footer.jsx: phần chân trang thống nhất toàn hệ thống.
* Khối UI hiển thị đơn hàng (trực tiếp trong file, chưa tách thành component riêng): mỗi đơn được hiển thị với:
  + Ảnh sản phẩm (item.image[0])
  + Tên sản phẩm (item.name)
  + Giá tiền (item.price)
  + Số lượng (item.quantity)
  + Màu sắc (item.color)
  + Trạng thái đơn hàng (item.status)
  + Phương thức thanh toán (item.paymentMethod)
  + Ngày đặt hàng (item.date)

- Điền đầy đủ thông tin để shop có thể vận chuyển đến khách hàng.

A screenshot of a computer

AI-generated content may be incorrect.

Hình 15. Trang xác nhận đặt hàng

### 2.8 Xác nhận đơn hàng (PlaceOrder)

Link code: <https://github.com/TranMinhPhuc04/ATOM_SHOP/blob/main/frontend/src/pages/PlaceOrder.jsx>

- **Mô tả chức năng:**

Sau khi xác nhận giỏ hàng, người dùng nhập thông tin giao hàng để tạo đơn. Dữ liệu sẽ được gửi lên server và chuyển sang trang xác minh.

**- Kỹ thuật sử dụng:**

* useContext: lấy dữ liệu giỏ hàng từ global state.
* useState: lưu thông tin người nhận hàng.
* axios: gửi đơn hàng lên backend.
* navigate: chuyển đến Verify.jsx.

**- Component sử dụng:**

* Header.jsx, Footer.jsx
* ShippingForm.jsx: nhập thông tin người nhận.
* OrderSummary.jsx: tổng tiền và danh sách sản phẩm.

- Nếu chọn “Stripe” bên trên sẽ dẫn đến trang điền thông tin thẻ.

A screenshot of a computer

AI-generated content may be incorrect.

Hình 16. Trang điền thông tin thẻ

- Chỉ có thể điền thông tin thẻ ở Hoa Kì, tham khảo thêm các trang thông tin thẻ tại: <https://paymentcloudinc.com/blog/stripe-test-cards/>

### 2.9 Xác thực (Verify)

Link code: <https://github.com/TranMinhPhuc04/ATOM_SHOP/blob/main/frontend/src/pages/Verify.jsx>

- **Mô tả chức năng:**

Trang xử lý sau thanh toán. Kiểm tra xem đơn hàng đã được xác nhận hay chưa và chuyển hướng phù hợp.

**- Kỹ thuật sử dụng:**

* useSearchParams: lấy orderId và success từ URL.
* axios: gửi request xác minh đơn hàng.
* useEffect: thực thi kiểm tra khi tải trang.
* toast: hiển thị lỗi nếu có.

A screenshot of a computer

AI-generated content may be incorrect.

Hình 17. Giao diện trang xác thực

### 2.10 Chi tiết sản phẩm (Product)

Link code: <https://github.com/TranMinhPhuc04/ATOM_SHOP/blob/main/frontend/src/pages/Product.jsx>

- **Mô tả chức năng:**

Trang này hiển thị chi tiết một sản phẩm cụ thể mà người dùng chọn từ trang danh sách. Người dùng có thể xem nhiều ảnh sản phẩm, tên, giá, đánh giá, mô tả, các tính năng, chọn màu sắc, và thêm sản phẩm vào giỏ hàng. Ngoài ra, trang còn đề xuất các sản phẩm liên quan để tăng khả năng mua sắm thêm.

**- Kỹ thuật sử dụng:**

* useParams: lấy productId từ URL.
* useContext: lấy danh sách sản phẩm từ ShopContext để tìm đúng sản phẩm cần hiển thị.
* useState: lưu ảnh đang hiển thị, màu sắc được chọn.
* useEffect: cập nhật thông tin sản phẩm mỗi khi productId thay đổi.
* Tương tác "add to cart" sử dụng hàm addToCart(product, color) từ context.

**- Component sử dụng:**

* ProductDescription.jsx: hiển thị mô tả chi tiết sản phẩm.
* ProductFeatures.jsx: liệt kê các đặc điểm kỹ thuật hoặc điểm nổi bật.
* RelatedProducts.jsx: hiển thị sản phẩm liên quan dựa theo danh mục hoặc thuộc tính.
* Footer.jsx: chân trang website.
* Các icon từ react-icons:
  + FaStar, FaStarHalfStroke: đánh giá sản phẩm bằng sao.
  + FaCheck, FaTruckFast: biểu tượng cho đặc điểm nổi bật (giao hàng nhanh, hàng chính hãng,...).
  + FaHeart: biểu tượng yêu thích.
  + TbShoppingBagPlus: biểu tượng thêm vào giỏ hàng.

A screenshot of a phone

AI-generated content may be incorrect.

Hình 18. Giao diện trang chi tiết sản phẩm

### 2.11 Các thành phần (Component)

Link code: <https://github.com/TranMinhPhuc04/ATOM_SHOP/tree/main/frontend/src/components>

#### 1. About.jsx

Link code: <https://github.com/TranMinhPhuc04/ATOM_SHOP/blob/main/frontend/src/components/About.jsx>

* Chức năng: Hiển thị thông tin giới thiệu về thương hiệu hoặc cửa hàng.
* Trang sử dụng: Home.jsx hoặc Contact.jsx (nếu có mở rộng).

#### 2. Banner.jsx

Link code: <https://github.com/TranMinhPhuc04/ATOM_SHOP/blob/main/frontend/src/components/Banner.jsx>

* Chức năng: Banner lớn đầu trang, dùng để giới thiệu sản phẩm nổi bật hoặc chương trình ưu đãi.
* Trang sử dụng: Home.jsx.

#### 3. Blog.jsx

Link code: <https://github.com/TranMinhPhuc04/ATOM_SHOP/blob/main/frontend/src/components/Blog.jsx>

* Chức năng: Hiển thị một bài viết ngắn trong danh sách blog gồm ảnh, tiêu đề và đoạn mô tả.
* Trang sử dụng: Blog.jsx.

#### 4. CartTotal.jsx

Link code: <https://github.com/TranMinhPhuc04/ATOM_SHOP/blob/main/frontend/src/components/CartTotal.jsx>

* Chức năng: Tổng hợp thông tin đơn hàng trong trang giỏ hàng như tổng tiền, phí ship, nút “Đặt hàng”.
* Trang sử dụng: Cart.jsx, PlaceOrder.jsx.

#### 5. Features.jsx

Link code: <https://github.com/TranMinhPhuc04/ATOM_SHOP/blob/main/frontend/src/components/Features.jsx>

* Chức năng: Liệt kê các tính năng nổi bật của shop (VD: giao nhanh, đổi trả dễ,...).
* Trang sử dụng: Home.jsx.

#### 6. Footer.jsx

Link code: <https://github.com/TranMinhPhuc04/ATOM_SHOP/blob/main/frontend/src/components/Footer.jsx>

* Chức năng: Phần chân trang dùng chung toàn bộ hệ thống, chứa thông tin liên hệ, điều hướng nhanh.
* Trang sử dụng: Tất cả các trang.

#### 7. Header.jsx

Link code: <https://github.com/TranMinhPhuc04/ATOM_SHOP/blob/main/frontend/src/components/Header.jsx>

* Chức năng: Thanh điều hướng chính của website gồm logo, menu, icon đăng nhập, giỏ hàng.
* Trang sử dụng: Tất cả các trang.

#### 8. Hero.jsx

Link code: <https://github.com/TranMinhPhuc04/ATOM_SHOP/blob/main/frontend/src/components/Hero.jsx>

* Chức năng: Vùng nổi bật đầu trang với hình ảnh lớn và thông điệp thu hút.
* Trang sử dụng: Home.jsx.

#### 9. Item.jsx

Link code: <https://github.com/TranMinhPhuc04/ATOM_SHOP/blob/main/frontend/src/components/Item.jsx>

* Chức năng: Card sản phẩm đơn, gồm hình ảnh, tên, giá, nút thêm vào giỏ.
* Trang sử dụng: NewArrivals.jsx, PopularProducts.jsx, Collection.jsx.

#### 10. NewArrivals.jsx

Link code: <https://github.com/TranMinhPhuc04/ATOM_SHOP/blob/main/frontend/src/components/NewArrivals.jsx>

* Chức năng: Hiển thị danh sách các sản phẩm mới.
* Trang sử dụng: Home.jsx.

#### 11. NewsLetter.jsx

Link code: <https://github.com/TranMinhPhuc04/ATOM_SHOP/blob/main/frontend/src/components/NewsLetter.jsx>

* Chức năng: Ô nhập email để đăng ký nhận thông báo/tin tức từ cửa hàng.
* Trang sử dụng: Home.jsx, Blog.jsx.

#### 12. PopularProducts.jsx

Link code: <https://github.com/TranMinhPhuc04/ATOM_SHOP/blob/main/frontend/src/components/PopularProducts.jsx>

* Chức năng: Hiển thị danh sách sản phẩm phổ biến, được ưa chuộng.
* Trang sử dụng: Home.jsx.

#### 13. ProductDescription.jsx

Link code: <https://github.com/TranMinhPhuc04/ATOM_SHOP/blob/main/frontend/src/components/ProductDescription.jsx>

* Chức năng: Hiển thị phần mô tả chi tiết về sản phẩm đang xem.
* Trang sử dụng: Product.jsx.

#### 14. ProductFeatures.jsx

Link code: <https://github.com/TranMinhPhuc04/ATOM_SHOP/blob/main/frontend/src/components/ProductFeatures.jsx>

* Chức năng: Hiển thị các đặc điểm kỹ thuật, công dụng hoặc ưu điểm nổi bật của sản phẩm.
* Trang sử dụng: Product.jsx.

#### 15. RelatedProducts.jsx

Link code: <https://github.com/TranMinhPhuc04/ATOM_SHOP/blob/main/frontend/src/components/RelatedProducts.jsx>

* Chức năng: Đề xuất các sản phẩm cùng danh mục hoặc tương tự sản phẩm đang xem.
* Trang sử dụng: Product.jsx.

#### 16. Search.jsx

Link code: <https://github.com/TranMinhPhuc04/ATOM_SHOP/blob/main/frontend/src/components/Search.jsx>

* Chức năng: Thanh tìm kiếm sản phẩm theo tên hoặc từ khoá.
* Trang sử dụng: Header.jsx hoặc Collection.jsx.

#### 17. Title.jsx

Link code: <https://github.com/TranMinhPhuc04/ATOM_SHOP/blob/main/frontend/src/components/Title.jsx>

* Chức năng: Component hiển thị tiêu đề chính với kiểu chữ nổi bật và định dạng đẹp.
* Trang sử dụng: Orders.jsx, Product.jsx, hoặc bất kỳ trang nào cần tiêu đề rõ ràng.

## 3. Hệ Thống Xử Lý Nghiệp Vụ (BackEnd)

- Đóng vai trò là trung tâm xử lý dữ liệu và logic nghiệp vụ cho toàn bộ website. Backend được xây dựng bằng **Node.js** kết hợp với **Express.js**, với cơ sở dữ liệu **MongoDB** được kết nối thông qua thư viện **Mongoose**. Toàn bộ hệ thống backend hoạt động dưới dạng **RESTful API**, cho phép frontend gọi đến để thực hiện các chức năng như: quản lý sản phẩm, xác thực người dùng, xử lý đơn hàng và thanh toán.

### 3.1 Cấu trúc thư mục

|  |  |
| --- | --- |
| **Thư mục / Tệp tin** | **Chức năng** |
| config/ | |  | | --- | |  |  |  | | --- | | Cấu hình kết nối MongoDB (db.js) và các biến môi trường. | |
| controllers/ | Xử lý logic cho các API như sản phẩm, đơn hàng, người dùng. |
| middleware/ | Chứa middleware kiểm tra token (verifyToken.js) và xử lý lỗi. |
| models/ | |  | | --- | |  |  |  | | --- | | Định nghĩa các schema Mongoose cho các đối tượng: Product, Order, User. | |
| routes/ | |  | | --- | |  |  |  | | --- | | Tổ chức các endpoint theo nhóm: /product, /order, /user,... | |
| .env | |  | | --- | |  |  |  | | --- | | Chứa biến môi trường (DB\_URL, PORT, v.v.). | |
| server.js | |  | | --- | |  |  |  | | --- | | File khởi động server chính, cấu hình middleware và mount route. | |
| vercel.json | |  | | --- | |  |  |  | | --- | | File cấu hình deploy server lên Vercel. | |

### 3.2 Kết nối cơ sở dữ liệu bằng Docker

- Để khởi động MongoDB local bằng Docker, nhóm sử dụng lệnh:

***docker run --name mongo-container -d -p 27017:27017 mongo***

A screenshot of a computer

AI-generated content may be incorrect.

Hình 19. Giao diện container MongoDB đang chạy trong Docker Desktop

### 3.3 Quản lý dữ liệu bằng TablePlus

- Để quản lý dữ liệu MongoDB dễ dàng và trực quan hơn, nhóm sử dụng phần mềm TablePlus để:

* Kết nối tới MongoDB đang chạy qua Docker.
* Xem dữ liệu các collection: products, orders, users.
* Hỗ trợ trực quan hóa các tài liệu JSON dạng bảng.

### 3.4 Tích hợp Cloudinary

- Đối với chức năng upload ảnh sản phẩm, backend sử dụng Cloudinary – dịch vụ lưu trữ đám mây – để:

* Nhận ảnh từ frontend gửi lên qua FormData.
* Tự động upload lên Cloudinary thông qua API Key riêng.
* Lưu đường dẫn ảnh trả về để frontend hiển thị.

A screenshot of a computer

AI-generated content may be incorrect.

Hình 20. API Key trên Cloudinary được sử dụng để cấu hình upload ảnh

# PHẦN III: KẾT LUẬN

Sau quá trình thiết kế và phát triển, nhóm đã hoàn thiện hệ thống website thương mại điện tử ATOM SHOP, bao gồm ba thành phần chính: giao diện khách hàng, giao diện quản trị và hệ thống backend xử lý dữ liệu. Các phần được tổ chức tách biệt nhưng liên kết chặt chẽ, đảm bảo luồng hoạt động xuyên suốt từ quản lý sản phẩm đến đặt hàng và xử lý đơn hàng. Hệ thống mô phỏng tương đối đầy đủ quy trình vận hành của một cửa hàng bán thiết bị công nghệ trên nền tảng trực tuyến.

## 1. Kết quả đạt được

* Hệ thống đã hoạt động ổn định, đầy đủ các luồng chức năng cơ bản và nâng cao.
* Giao diện trực quan, dễ sử dụng, thân thiện với cả người dùng và quản trị viên.
* Mã nguồn được tổ chức rõ ràng, API backend thiết kế hợp lý, dễ mở rộng và tái sử dụng.
* Dự án có thể triển khai thực tế cho các mô hình kinh doanh thương mại điện tử quy mô nhỏ đến vừa.

## 2. Kiến thức và kỹ năng đạt được

Thông qua quá trình thực hiện đồ án, các thành viên trong nhóm đã đạt được nhiều kiến thức và kỹ năng thực tiễn, cụ thể:

* Hiểu rõ quy trình phát triển một hệ thống web hoàn chỉnh từ frontend đến backend.
* Thành thạo các công nghệ hiện đại như **ReactJS, Tailwind CSS, NodeJS, MongoDB, Stripe, Cloudinary**,...
* Rèn luyện kỹ năng xử lý lỗi, quản lý state, tối ưu giao diện và xử lý bất đồng bộ.
* Nâng cao kỹ năng sử dụng công cụ quản lý mã nguồn **Git & GitHub**.
* Phối hợp làm việc nhóm hiệu quả, chia nhỏ tính năng, kiểm thử và triển khai theo từng giai đoạn.
* Rèn luyện kỹ năng trình bày dự án mạch lạc, đúng cấu trúc, dễ tiếp cận với người đọc và người đánh giá.

## 3. Định hướng phát triển

Trong tương lai, hệ thống ATOM SHOP có thể được mở rộng và nâng cấp với các tính năng và cải tiến sau:

### 1. Chức năng mở rộng:

* Quản lý người dùng nâng cao: phân quyền, trạng thái hoạt động, lịch sử đặt hàng chi tiết.
* Cải tiến trải nghiệm người dùng trên thiết bị di động, tối ưu UI/UX.
* Tăng cường khả năng tìm kiếm, lọc và sắp xếp sản phẩm/đơn hàng theo nhiều tiêu chí.
* Tích hợp API bên thứ ba: dịch vụ vận chuyển, thông báo đơn hàng qua email hoặc Zalo.
* Bổ sung hệ thống đánh giá, bình luận sản phẩm từ khách hàng sau khi nhận hàng.

### 2. Nâng cấp kỹ thuật:

* Tái cấu trúc mã nguồn theo mô hình MVC hoặc sử dụng Redux Toolkit, Zustand để quản lý trạng thái hiệu quả hơn.
* Tích hợp hệ thống CMS đơn giản để quản lý nội dung blog, chương trình khuyến mãi.
* Tăng cường bảo mật: chống XSS, CSRF, triển khai xác thực đa yếu tố (2FA).

# PHẦN IV: PHỤ LỤC

## 1. Phụ Lục

Link GitHub: <https://github.com/TranMinhPhuc04/ATOM_SHOP>

Link Vercel:

* Admin: <https://atom-shop-jdbr.vercel.app/>
* User: <https://atom-shop-dka7.vercel.app/>

## 2. Phân Công

|  |  |  |
| --- | --- | --- |
| **Họ và tên** | **Công việc cụ thể** | **Đánh giá** |
| **Trần Minh Phúc** | - Lập trình chính toàn bộ hệ thống giao diện ReactJS. - Thiết lập cấu trúc thư mục dự án. - Tạo layout tổng thể, routing và điều hướng trang. - Viết logic xử lý các chức năng chính: thêm sản phẩm, đơn hàng, đăng nhập. - Tích hợp react-icons, toast thông báo, xử lý lỗi import. | **100%** |
| **Võ Đình Ngọc Bình** | - Hỗ trợ lập trình giao diện form thêm sản phẩm. - Xử lý truyền props giữa các component. - Kiểm tra dữ liệu đầu vào, hỗ trợ xử lý hiển thị. - Viết phần mô tả chức năn g cho báo cáo. | **100%** |
| **Châu Gia Kiệt** | - Hỗ trợ kiểm thử giao diện toàn hệ thống. - Tìm và chuẩn bị tài nguyên hình ảnh, icon cho phần giao diện. - Sửa lỗi import ảnh và lỗi chạy khi thiếu thư viện. - Đóng góp ý kiến về UI tổng thể. | **100%** |
| **Nguyễn Hoàng Khanh** | - **Phụ trách toàn bộ phần viết báo cáo Word cho dự án.** - Tổng hợp thông tin, trình bày nội dung mạch lạc, đầy đủ. - Bố cục hợp lý, viết lời cảm ơn, giới thiệu, mô tả, kết luận, phân công. - Thực hiện **thiết kế slide PowerPoint** để thuyết trình. | **100%** |

# TÀI LIỆU THAM KHẢO

Fyke, B. (2025, 01 15). *Stripe Test Credit Card Numbers: Comprehensive List for 2025*. Retrieved 04 01, 2025, from Payment Cloud: https://paymentcloudinc.com/blog/stripe-test-cards/

**--- HẾT---**