ĐẠI HỌC BÁCH KHOA HÀ NỘI

TRƯỜNG CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG

──────── \* ────────

A picture containing text, poster, font, graphics

Description automatically generated

**BÁO CÁO**

HỌC PHẦN: NGHIÊN CỨU TỐT NGHIỆP 2

*(Mã học phần: IT5022)*

*Đề tài:*  **NGHIÊN CỨU VÀ ÁP DỤNG CÔNG NGHỆ NODEJS VÀO TRONG WEBSITE BÁN HÀNG**

|  |  |  |
| --- | --- | --- |
| **GV hướng dẫn:** | Bùi Quốc Trung | |
| **Sinh viên thực hiện:** | Bùi Trung Hùng | - **MSSV**: 20200255 |
|  |  |  |
|  |  |  |
|  |  |  |

*Hà Nội, tháng 1 năm 2025*

Contents

[LỜI MỞ ĐẦU 3](#_Toc187863126)

[LỜI CẢM ƠN 4](#_Toc187863127)

[**I. Tổng Quan** 5](#_Toc187863128)

[1.1 Mục Đích 5](#_Toc187863129)

[1.2 Lý Do Phát Triển 5](#_Toc187863130)

[1.3 Ý Nghĩa 5](#_Toc187863131)

[**II. Mô tả đề tài** 6](#_Toc187863132)

[2.1 Xác định yêu cầu 6](#_Toc187863133)

[2.2 Xác định chức năng của Website 6](#_Toc187863134)

[2.3 Các kịch bản sử dụng chính 6](#_Toc187863135)

[2.4 Các đặc điểm quan trọng 7](#_Toc187863136)

[**III. Usecase** 8](#_Toc187863137)

[3.1 Sơ đồ Usecase 8](#_Toc187863138)

[3.1.1 Sơ đồ Usecase tổng quan 8](#_Toc187863139)

[3.1.2 Sơ đồ Usecase phân rã 9](#_Toc187863140)

[3.2 Đặc tả chức năng 11](#_Toc187863141)

[3.2.1 Chức năng chung 11](#_Toc187863142)

[**IV. Mô tả chi tiết hệ thống** 14](#_Toc187863143)

[4.1 Thiết kế hệ thống 14](#_Toc187863144)

[4.2 Biểu đồ hoạt động 17](#_Toc187863145)

[4.3 Các màn hình chính 28](#_Toc187863146)

[**V. Công nghệ sử dụng** 33](#_Toc187863147)

[5.1 HTML 33](#_Toc187863148)

[5.2 CSS 34](#_Toc187863149)

[5.3 JavaScript 34](#_Toc187863150)

[5.4 ReactJS 35](#_Toc187863151)

[5.5 NodeJS 36](#_Toc187863152)

[**VI. Tổng kết** 37](#_Toc187863153)

[5.1 Khó khăn và bài học 37](#_Toc187863154)

[5.2 Hướng phát triển sau này 37](#_Toc187863155)

[5.3 Kết luận 38](#_Toc187863156)

# LỜI MỞ ĐẦU

Trong quá trình học tập và khám phá các công nghệ lập trình mới, việc thực hành xây dựng các ứng dụng thực tế đóng vai trò quan trọng trong việc củng cố kiến thức và nâng cao kỹ năng. Với mục tiêu làm quen và tìm hiểu sâu hơn về nền tảng Node.js, tôi đã chọn thực hiện một dự án xây dựng website bán hàng.

Node.js là một công nghệ mạnh mẽ, phổ biến trong việc phát triển các ứng dụng web hiện đại nhờ khả năng xử lý đồng thời hiệu quả, hiệu suất cao và cộng đồng phát triển lớn mạnh. Dự án này không chỉ giúp tôi nắm bắt các kiến thức cơ bản về Node.js, mà còn mở ra cơ hội học hỏi cách xây dựng một hệ thống web hoàn chỉnh, từ cấu trúc cơ bản đến tích hợp chức năng cần thiết.

Báo cáo này ghi lại hành trình học tập và thực hành của tôi, từ quá trình tìm hiểu lý thuyết, thiết kế đến triển khai các tính năng cơ bản của website. Dù sản phẩm còn mang tính thử nghiệm và chưa hoàn thiện, nhưng tôi tin rằng đây là một bước đi quan trọng trong việc tiếp cận công nghệ mới và chuẩn bị cho những dự án phức tạp hơn trong tương lai.

# LỜI CẢM ƠN

Trong quá trình làm nghiên cứu và học tập công nghệ NodeJS để làm sản phẩm Website bán hàng này, em xin gửi lời cảm ơn chân thành đến: Giáo viên – TS. Bùi Quốc Trung đã hết lòng hỗ trợ, giúp đỡ, tạo điều kiện cũng như hướng dẫn tận tình cùng với những nhận xét, góp ý đúng đắn để giúp em có thể hoàn thành tốt nhất.

# **I. Tổng Quan**

## 1.1 Mục Đích

Mục đích của dự án này là xây dựng một website bán hàng cơ bản sử dụng nền tảng Node.js nhằm:

* Làm quen và thực hành các kiến thức về lập trình backend với Node.js.
* Tìm hiểu cách thiết kế và triển khai một ứng dụng web hoàn chỉnh, từ backend đến giao diện người dùng.
* Hiểu rõ quy trình phát triển phần mềm thông qua việc phân tích yêu cầu, thiết kế hệ thống và triển khai chức năng.
* Đáp ứng nhu cầu học tập và nâng cao kỹ năng lập trình trong môi trường thực tiễn.

## 1.2 Lý Do Phát Triển

Sự phát triển không ngừng của công nghệ đã thúc đẩy nhu cầu xây dựng các ứng dụng web nhanh chóng, mạnh mẽ và dễ mở rộng. Node.js là một trong những công nghệ nổi bật, được nhiều công ty và cá nhân sử dụng để phát triển các ứng dụng hiện đại.

Dự án được thực hiện dựa trên những lý do chính sau:

* Node.js cho phép lập trình viên sử dụng cùng một ngôn ngữ (JavaScript) cho cả frontend và backend, giúp giảm thiểu độ phức tạp.
* Nhu cầu học tập và thực hành thực tế về phát triển web để chuẩn bị cho các dự án lớn hơn.
* Thử nghiệm khả năng tích hợp các công nghệ mới với Node.js, như Express.js, MongoDB và các công cụ frontend.

## 1.3 Ý Nghĩa

Dự án không chỉ mang lại ý nghĩa về mặt học tập mà còn có giá trị thực tiễn:

* Về học tập: Củng cố và mở rộng kiến thức lập trình web, giúp hiểu rõ hơn về cách xây dựng một hệ thống từ đầu đến cuối.
* Về kỹ năng: Phát triển kỹ năng lập trình, tư duy giải quyết vấn đề và quản lý thời gian khi thực hiện dự án.
* Về thực tế: Dự án có thể làm nền tảng cho việc phát triển các hệ thống thương mại điện tử hoặc các ứng dụng khác trong tương lai.

Thông qua dự án này, tôi mong muốn tạo ra một sản phẩm học tập vừa hữu ích vừa thực tiễn, đồng thời rèn luyện khả năng tự học và tiếp cận công nghệ mới.

# **II. Mô tả đề tài**

## 2.1 Xác định yêu cầu

Dự án hướng tới việc học công nghệ NodeJS sau đó áp dụng vào việc xây dựng một trang web bán hàng cơ bản với các yêu cầu sau:

Giao diện người dùng:

* Giao diện thân thiện, dễ sử dụng
* Hiển thị danh sách sản phẩm với thông tin chi tiết
* Website phân quyền giữa người dùng thông thường và quản trị viên

Chức năng cơ bản:

* Người dùng có thể xem, tìm kiếm sản phẩm, thêm sản phầm vào giỏ hàng, đặt hàng, có thể thay đổi thông tin cá nhân để phù hợp với bản thân
* Quản trị viên có thể quản lý sản phẩm, đồng thời có thể nắm bắt được thông tin của khách hàng và những tài khoản quản trị viên khác

Yêu cầu:

* Backend sử dụng NodeJS và framework Express.js
* Cơ sở dữ liệu: MongoDB
* Xác thực người dùng với hệ thống đăng nhập/đăng ký

## 2.2 Xác định chức năng của Website

Đối với khách hàng:

* Xem danh sách sản phẩm
* Xem danh sách sản phẩm theo danh mục
* Tìm kiếm sản phẩm
* Thêm sản phầm vào giỏ hàng
* Đăng ký tài khoản, đăng nhập để có thể mua, theo dõi đơn hàng

Đối với quản trị viên:

* Quản lý danh mục sản phẩm
* Quản lý sản phẩm: Thêm, sửa, xóa
* Quản lý tài khoản: Xem, sửa, xóa tài khoản

## 2.3 Các kịch bản sử dụng chính

Khách hàng duyệt sản phẩm và đặt hàng:

* Truy cập trang web, xem danh sách sản phẩm
* Chọn một sản phẩm để xem chi tiết
* Thêm sản phầm vào giỏ hàng
* Thực hiện đặt hàng

Quản trị viên quản lý sản phẩm

* Đăng nhập vào với tài khoản quản trị viên
* Thêm sản phẩm mới với thông tin chi tiết
* Sửa thông tin sản phẩm hoặc xóa sản phầm không còn kinh doanh

Khách hàng đăng ký tài khoản

* Truy cập trang đăng ký
* Điền thông tin và tạo tài khoản
* Đăng nhập để mua, theo dõi đơn hàng

## 2.4 Các đặc điểm quan trọng

Hiệu suất:

* Sử dụng NodeJS để xử lý yêu cầu đồng thời đảm bảo tốc độ tài trang nhanh và xử lý hiệu quả

Bảo mật:

* Mã hóa mật khẩu người dùng ( bcrypt )
* Xác thực và phân quyền người dùng( JWT )

Thân thiện với người dùng:

* Giao diện đơn giản, dễ sử dụng.

Khả năng mở rộng:

* Hệ thống được thiết kế module hóa để dễ dang mở rộng và thêm chức năng trong tương lai.

# **III. Usecase**

## 3.1 Sơ đồ Usecase

### 3.1.1 Sơ đồ Usecase tổng quan

A diagram of a person's relationship

Description automatically generated

### 3.1.2 Sơ đồ Usecase phân rã

UsecaseGuest:

A diagram of a diagram

Description automatically generated with medium confidence

UsecaseUser:

A diagram of a person

Description automatically generated

UsecaseAdmin:

A diagram of a person with text

Description automatically generated with medium confidence

## 3.2 Đặc tả chức năng

### 3.2.1 Chức năng chung

3.2.1.1 Xem toàn bộ sản phẩm

Mô tả: Người dùng và admin đều có thể xem danh sách toàn bộ sản phẩm, bao gồm các thông tin như: tên sản phẩm, giá, mô tả, tình trạng kho, hình ảnh.

Mục đích: Hiển thị danh sách sản phẩm để khách hàng dễ dàng tiếp cận thông tin.

Chi tiết:

* + Giao diện chính hiển thị danh sách sản phẩm.
  + Chức năng phân trang nếu danh sách sản phẩm dài.
  + API kết nối cơ sở dữ liệu MongoDB để truy vấn sản phẩm.

3.2.1.2 Tìm kiếm sản phẩm

Mô tả: Người dùng có thể tìm kiếm sản phẩm dựa trên từ khóa hoặc lọc theo tên sản phẩm.

Mục đích: Tăng khả năng tìm kiếm và tối ưu trải nghiệm người dùng.

Chi tiết:

* + Hộp tìm kiếm ở vị trí cố định trên giao diện.
  + API trả về danh sách sản phẩm phù hợp với từ khóa.

3.2.1.3 Xem chi tiết sản phẩm

Mô tả: Người dùng có thể xem thông tin chi tiết về sản phẩm như thông số kỹ thuật, đánh giá, mô tả chi tiết.

Mục đích: Cung cấp đủ thông tin để người dùng đưa ra quyết định mua hàng.

Chi tiết:

* + Trang chi tiết sản phẩm hiển thị đầy đủ thông tin.
  + Hỗ trợ gợi ý sản phẩm tương tự.

3.2.1.4 Đăng ký, đăng nhập và đăng xuất

Mô tả: Người dùng có thể tạo tài khoản, đăng nhập để sử dụng các chức năng cá nhân hóa hoặc đăng xuất khi không sử dụng.

Mục đích: Quản lý tài khoản và bảo mật thông tin.

Chi tiết:

* + Form đăng ký tài khoản yêu cầu thông tin cơ bản.
  + Token-based authentication (JWT) để bảo mật phiên người dùng.

3.1.2 Chức năng dành cho người dùng

3.1.2.1 Thay đổi thông tin cá nhân

Mô tả: Người dùng có thể cập nhật thông tin cá nhân như tên, địa chỉ, avatar, email, và số điện thoại.

Mục đích: Đảm bảo thông tin chính xác để hỗ trợ mua sắm và giao hàng.

Chi tiết:

* + Form cập nhật thông tin.
  + API ghi nhận và lưu trữ thông tin cập nhật vào cơ sở dữ liệu.

3.1.2.2 Quản lý giỏ hàng

Mô tả: Người dùng có thể thêm, sửa hoặc xóa sản phẩm trong giỏ hàng.

Mục đích: Quản lý các sản phẩm đã chọn trước khi tiến hành thanh toán.

Chi tiết:

* + Nút "Chọn mua" trên trang chi tiết sản phẩm.
  + Trang giỏ hàng hiển thị danh sách sản phẩm đã chọn có thể tăng giảm số lượng hoặc xóa sản phẩm khỏi giỏ hàng.
  + API điều chỉnh số lượng hoặc xóa sản phẩm.

3.1.2.3 Thanh toán ( đang hoàn thiện)

Mô tả: Người dùng hoàn tất đơn hàng thông qua các bước thanh toán, bao gồm nhập thông tin giao hàng và chọn phương thức thanh toán.

Mục đích: Ghi nhận đơn hàng và thực hiện giao dịch.

Chi tiết:

* + Form nhập thông tin giao hàng (họ tên, địa chỉ, số điện thoại nếu thiếu thì điền thêm).
  + API cập nhật và lưu thông tin cá nhân.
  + Tích hợp cổng thanh toán online.
  + API xử lý đơn hàng và lưu vào cơ sở dữ liệu.

3.1.3 Chức năng dành cho Admin

3.1.3.1 Quản lý người dùng

Mô tả: Admin có thể xem danh sách người dùng, sửa thông tin, hoặc xóa tài khoản người dùng, đồng thười có thể xuất excel toàn bộ thông tin người dùng.

Mục đích: Đảm bảo thông tin người dùng được cập nhật và quản lý tốt.

Chi tiết:

* + Bảng danh sách người dùng.
  + API hỗ trợ sửa thông tin hoặc xóa tài khoản.

3.1.3.2 Quản lý sản phẩm

Mô tả: Admin có thể thêm mới, sửa đổi thông tin hoặc xóa sản phẩm, đồng thời có thể xuất excel toàn bộ chi tiết sản phẩm

Mục đích: Quản lý danh mục sản phẩm và đảm bảo tính nhất quán dữ liệu.

Chi tiết:

* + Form thêm sản phẩm với các trường thông tin chi tiết.
  + API hỗ trợ thêm mới, sửa hoặc xóa sản phẩm khỏi cơ sở dữ liệu.

# **IV. Mô tả chi tiết hệ thống**

## 4.1 Thiết kế hệ thống

4.1.1 Thiết kế kiến trúc

Phần mềm phát triển dựa trên mô hình Client – Server là mô hình mạng máy tính gồm có 2 thành phần chính đó là máy khách (client) và máy chủ (server). Server là nơi lưu trữ tài nguyên, cài đặt dịch vụ. Còn client là nơi tiếp nhận các yêu cầu của người dùng, thực hiện gửi các yêu cầu này đến sever sau đó sẽ tổ chức trả lại kết quả đó cho người dùng.

Từ mô hình kiến trúc Client – Server, em đã xây dựng và phát triển phần mềm dựa trên 2 thành phần bao gồm máy khách (user) bằng thư viện ReactJS và máy chủ (server) bằng nền tảng MongoDB.

A black arrow pointing to a point

Description automatically generated

4.1.2 Thiết kế dữ liệu

Bảng User:

|  |  |
| --- | --- |
| Tên trường | Kiểu dữ liệu |
| name | String |
| email | String |
| password | String |
| isAdmin | Boolean |
| phone | Number |
| address | String |
| avatar | String |

Bảng Product:

|  |  |
| --- | --- |
| Tên trường | Kiểu dữ liệu |
| name | String |
| image | String |
| type | String |
| price | Number |
| countInStock | Number |
| rating | Number |
| description | String |
| discount | Number |
| sold | Number |

Bảng Order:

|  |  |
| --- | --- |
| Tên trường | Kiểu dữ liệu |
| orderItems | [] |
| shippingAddress | [] |
| paymentMethod | String |
| itemsPrice | Boolean |
| shippingPrice | Number |
| taxPrice | String |
| totalPrice | String |
| user | id |
| isPaid | Boolean |
| paidAt | Date |
| isDelivered | Boolean |
| deliveredAt | Date |

Bảng OrderItems:

|  |  |
| --- | --- |
| Tên trường | Kiểu dữ liệu |
| name | String |
| amount | Number |
| image | String |
| price | Number |
| product | id |

Bảng ShippingAddress:

|  |  |
| --- | --- |
| Tên trường | Kiểu dữ liệu |
| name | String |
| address | String |
| city | String |
| country | String |
| phone | Number |

## 4.2 Biểu đồ hoạt động

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A diagram with text and words

Description automatically generated with medium confidence

A screenshot of a computer screen

Description automatically generated

A diagram with text and words

Description automatically generated with medium confidence

A screenshot of a computer

Description automatically generated

A screenshot of a diagram

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A diagram with many lines and dots

Description automatically generated with medium confidence

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

## 4.3 Các màn hình chính

4.3.1 Màn hình chính

A screenshot of a computer

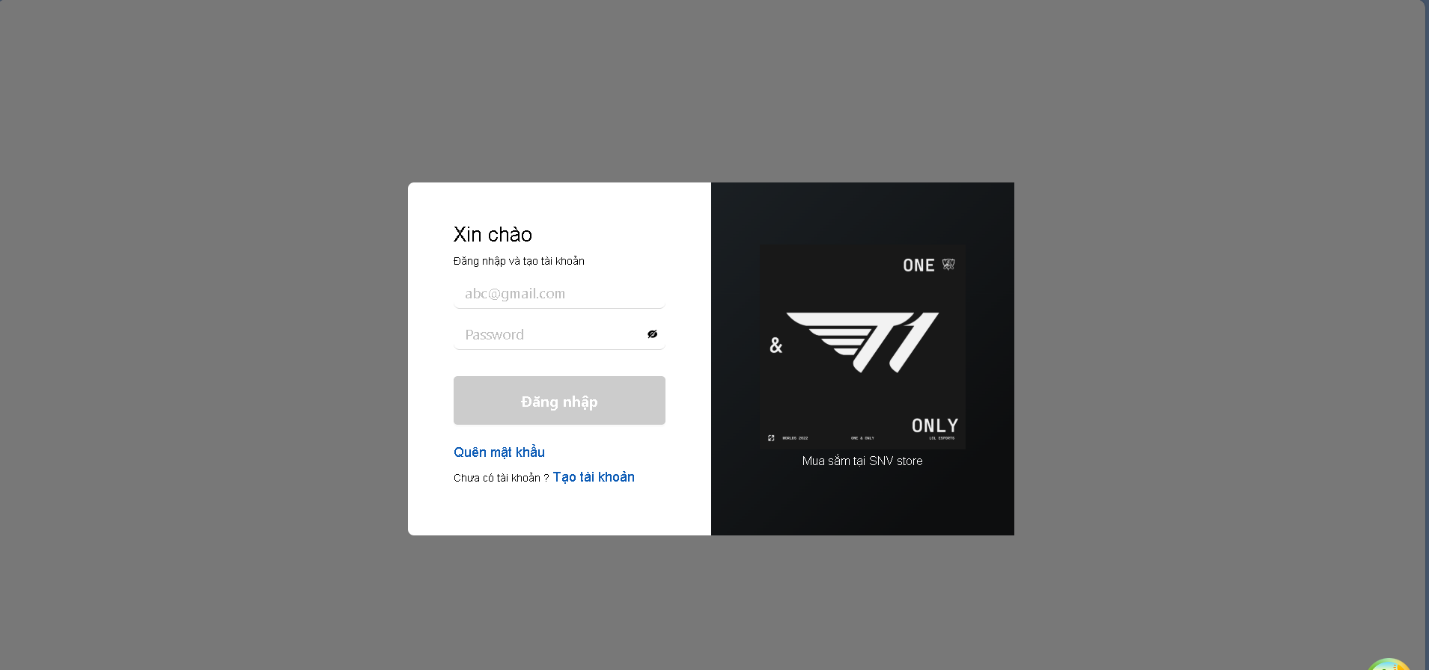
Description automatically generated

4.3.2 Đăng ký

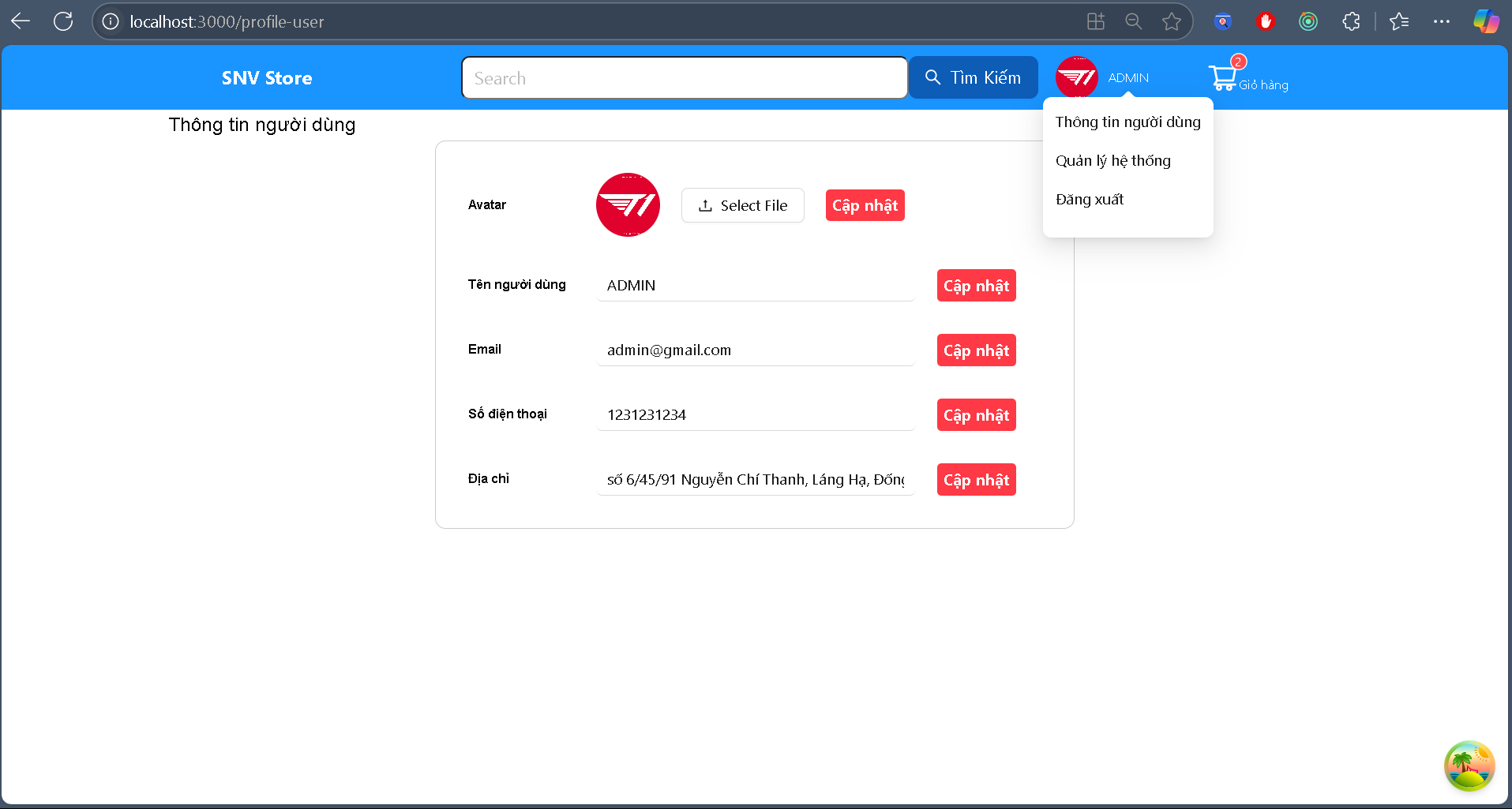
A screenshot of a login page

Description automatically generated

4.3.3 Đăng nhập



4.3.4 Quản lý thông tin cá nhân



4.3.5 Chi tiết sản phẩm

A screenshot of a computer

Description automatically generated

4.3.6 Giỏ hàng

A screenshot of a computer

Description automatically generated

4.3.7 Quản lý người dùng

A screenshot of a computer

Description automatically generatedA screenshot of a computer

Description automatically generated

4.3.8 Quản lý sản phẩm

A screenshot of a computer

Description automatically generatedA screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

# **V. Công nghệ sử dụng**

## 5.1 HTML

A logo of a html website

Description automatically generated

● Để tạo khuôn cho trang web trao đổi lập trình, em đã lựa chọn ngôn ngữ HTML, là chữ viết tắt của Hypertext Markup Language, nhờ đó có thể cấu trúc các thành phần trong trang web, phân chia các đoạn văn, heading, links, blockquotes, … và cũng là đáp ứng yêu cầu của môn học.

● HTML không dùng để tạo những thành phần có tính “động”, chỉ dùng để tạo bố cục, do đó em đưa ra cách trình bày bố cục sao cho thật sự dễ nhìn, sau đó mới kết hợp với các công cụ trình bày khác để giúp trang web đẹp hơn và “động” hơn.

## 5.2 CSS

A blue and white logo

Description automatically generated

● Để trang web được sinh động hơn với bố cục đã xây dựng bằng HTML cũng như đảm bảo yêu cầu môn học, em chọn ngôn ngữ CSS là một công cụ thiết kế giao diện.

● CSS là chữ viết tắt của Cascading Style Sheets, nó là một ngôn ngữ được sử dụng để tìm và định dạng lại các phần tử được tạo ra bởi các ngôn ngữ đánh dấu như HTML. Đây đích thực là ngôn ngữ tạo phong cách cho trang web, có thể thay đổi bố cục, màu sắc trang, đổi màu chữ, font chữ, thay đổi cấu trúc, … Phương thức hoạt động của CSS là nó sẽ tìm các vùng ở trong trang HTML dựa tên một thẻ, tên một ID, class hay nhiều kiểu khác. Sau đó là nó sẽ áp dụng các thuộc tính cần thay đổi lên vùng chọn đó.

## 5.3 JavaScript

A yellow background with white text

Description automatically generated

● JavaScript (JS) được sử dụng trong cả frontend và backend của dự án, đóng vai trò cốt lõi trong việc xây dựng các chức năng chính của hệ thống.

● JavaScript là một ngôn ngữ lập trình linh hoạt, phổ biến, được thiết kế để làm cho các trang web trở nên động và tương tác hơn. Với khả năng hoạt động trên cả trình duyệt và server (với NodeJS), JavaScript là sự lựa chọn hoàn hảo cho các dự án web.

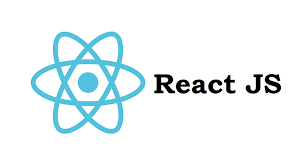
● Trong dự án, em sử dụng JavaScript để xử lý các nhiệm vụ:

- Frontend: Tạo các giao diện động thông qua ReactJS. JavaScript giúp xây dựng các component tương tác, quản lý sự kiện (event handling), và giao tiếp với backend thông qua API RESTful bằng các thư viện như axios hoặc fetch.

- Backend: Kết hợp với NodeJS, JavaScript được sử dụng để viết logic xử lý nghiệp vụ, kết nối cơ sở dữ liệu MongoDB, và xử lý bất đồng bộ thông qua async/await.

● Việc sử dụng JavaScript giúp em tận dụng tối đa sự nhất quán trong ngôn ngữ lập trình cho cả frontend và backend, giảm thiểu độ phức tạp khi phối hợp giữa các thành phần hệ thống.

## 5.4 ReactJS



● ReactJS là một thư viện JavaScript mã nguồn mở được sử dụng để xây dựng giao diện người dùng (User Interface - UI), đặc biệt là cho các ứng dụng web đơn trang (Single Page Applications - SPA). ReactJS được phát triển và duy trì bởi Facebook, cung cấp cách tiếp cận khai báo giúp việc xây dựng các thành phần UI trở nên dễ dàng và trực quan.

● ReactJS cho phép chia giao diện thành các component nhỏ, có thể tái sử dụng, giúp tăng hiệu quả và sự linh hoạt khi phát triển. Các component này được viết bằng JavaScript kết hợp với JSX (JavaScript XML) để định nghĩa giao diện. JSX cho phép dễ dàng kết hợp logic JavaScript với HTML.

● Một tính năng nổi bật của ReactJS là Virtual DOM – cơ chế so sánh và cập nhật hiệu quả các thay đổi trên giao diện mà không cần tải lại toàn bộ trang, cải thiện hiệu suất đáng kể.

● Trong dự án, em đã chọn ReactJS để xây dựng giao diện vì:

- Tái sử dụng code: Các component được viết một lần và sử dụng ở nhiều nơi, giúp tiết kiệm thời gian.

- Hiệu năng cao: Virtual DOM giúp trang phản hồi nhanh ngay cả khi có nhiều thay đổi.

- Tích hợp dễ dàng: ReactJS tương thích với nhiều thư viện và công nghệ khác.

## 5.5 NodeJS

A logo of a company

Description automatically generated

● Để xây dựng phần backend của trang web, nhóm đã chọn NodeJS làm công cụ chính, đáp ứng tốt các yêu cầu về xử lý dữ liệu và giao tiếp giữa client và server.

● NodeJS là một môi trường chạy JavaScript phía server, được xây dựng trên V8 Engine của Google Chrome. NodeJS không chỉ giúp thực thi JavaScript bên ngoài trình duyệt mà còn hỗ trợ xử lý tác vụ bất đồng bộ thông qua cơ chế Event Loop, mang lại hiệu suất cao cho các ứng dụng web.

● Trong quá trình phát triển, NodeJS cho phép em dễ dàng xử lý các yêu cầu từ phía client thông qua giao thức HTTP và thực hiện các nghiệp vụ phức tạp như quản lý cơ sở dữ liệu, xác thực người dùng, và bảo mật. Đặc biệt, em sử dụng NPM (Node Package Manager) để tích hợp các thư viện như Express, Mongoose, jsonwebtoken, giúp tăng tốc độ và tính chính xác khi phát triển backend.

● Nhờ vào sự đơn giản và linh hoạt của NodeJS, em đã xây dựng backend mạnh mẽ, hỗ trợ tốt các tính năng như quản lý sản phẩm, giỏ hàng, thanh toán và quản lý người dùng. Ngoài ra, NodeJS còn giúp em đảm bảo hiệu năng cao khi xử lý đồng thời nhiều kết nối từ phía client.

# **VI. Tổng kết**

## 5.1 Khó khăn và bài học

Khó khăn:

* Thiết kế hệ thống: Việc xây dựng kiến trúc hệ thống từ đầu yêu cầu sự phối hợp giữa frontend, backend, và cơ sở dữ liệu, dẫn đến thời gian phát triển kéo dài.
* Bảo mật: Quản lý thông tin người dùng và thanh toán đòi hỏi cơ chế bảo mật nghiêm ngặt để tránh các lỗ hổng.
* Hiệu năng: Tối ưu hóa tốc độ truy vấn cơ sở dữ liệu và tải trang khi danh sách sản phẩm lớn là một thách thức lớn.
* Trải nghiệm người dùng: Đảm bảo giao diện thân thiện và dễ sử dụng cần nhiều thử nghiệm và điều chỉnh dựa trên phản hồi.

Bài học:

* Quản lý dự án: Phân chia công việc rõ ràng và lập kế hoạch hợp lý giúp tối ưu hóa thời gian và công sức.
* Kiểm thử sớm: Phát hiện lỗi sớm thông qua kiểm thử liên tục giúp giảm thiểu rủi ro trong giai đoạn triển khai.

## 5.2 Hướng phát triển sau này

Tích hợp AI:

* Gợi ý sản phẩm thông minh dựa trên hành vi mua sắm của người dùng.
* Chatbot hỗ trợ khách hàng tự động.
* Hỗ trợ đa nền tảng:

Tăng cường bảo mật:

* Sử dụng các giao thức bảo mật nâng cao như OAuth2.0, HTTPS mọi nơi.
* Tích hợp xác thực hai lớp (2FA) để bảo vệ tài khoản người dùng.
* Phát triển hệ thống báo cáo nâng cao:

Mở rộng chức năng:

* Hỗ trợ khách hàng đăng ký thành viên VIP với các quyền lợi riêng.
* Tích hợp hệ thống đánh giá và nhận xét sản phẩm.

## 5.3 Kết luận

Dự án xây dựng website quản lý sản phẩm đã hoàn thành với các chức năng cơ bản phục vụ cho hai nhóm đối tượng chính: người dùng và admin. Hệ thống được thiết kế linh hoạt, hiện đại, sử dụng các công nghệ mới nhất như ReactJS, NodeJS, và MongoDB.

Mặc dù gặp phải một số khó khăn trong quá trình phát triển, dự án đã mang lại nhiều bài học quý giá, đặc biệt trong việc tối ưu hóa hệ thống và nâng cao trải nghiệm người dùng.

Trong tương lai, việc mở rộng và nâng cấp hệ thống sẽ giúp dự án tiếp cận được nhiều khách hàng hơn, nâng cao chất lượng dịch vụ, và hỗ trợ các mục tiêu kinh doanh dài hạn.