

BÁO CÁO TIẾN ĐỘ THÁNG 4

Sinh viên: Phạm Văn Á

MSSV: cm21v7x306

- Đặc tả/ mô tả đề tài

Studio Như đã thành lập trên 10 năm tại Phường 2 - TP.Cà Mau. Tiệm chụp ảnh có nhu cầu quản lý nhân sự, quản lý sản phẩm, quảng bá sản phẩm và đặc biệt là tìm giải pháp kinh doanh online cho các sản phẩm về ảnh, khung ảnh và chỉnh sửa ảnh.

Quản lý được tất cả sản phẩm của tiệm ảnh, quản lý nhân sự, quảng bá sản phẩm của tiệm ảnh

- Các nghiên cứu có liên quan

Dựa vào kiến thức nền tảng khi thực hiện Niên luận cơ sở và Niên luận ngành sinh viên hoàn thiện Đề tài Website Studio Như.

Phát triển ứng dụng web với Nodejs, MongoDB

- Phương pháp thực hiện giải quyết bài toán

1. Thực hiện từng bước tìm hiểu công nghệ phát triển web.
2. Xây dựng ứng dụng và lưu code vào Github
<https://github.com/phamvana/TieuluanTN/>
3. Phân tích nhu cầu và xây dựng các use case cho ứng dụng.
4. Thực hiện báo cáo đề tài:

1. Phạm Văn Á_cm21v7x306_CT271_Tiểu luận TN ngành CNTT

MÔ TẢ BÀI TOÁN

I. MÔ TẢ YÊU CẦU

Sau khi thực hiện khảo sát, có thể xác định được các tác nhân sử dụng hệ thống như sau:

- Quản lý cửa hàng: Người quản lý hệ thống, có toàn quyền sử dụng và quản lý trên dữ liệu hệ thống.
- Thành viên: Là người đã có tài khoản và có thể mua sản phẩm.
- Người dùng khách: Là khách hàng chưa có tài khoản thành viên trong hệ thống.

Dựa vào tác nhân trên ứng dụng có thể chia làm 2 module chính: một module dùng cho quản trị viên, một module dành cho khách hàng.

Thực hiện: Phạm Văn Á

Thời gian: 31/3/2024 đến 7/4/2024

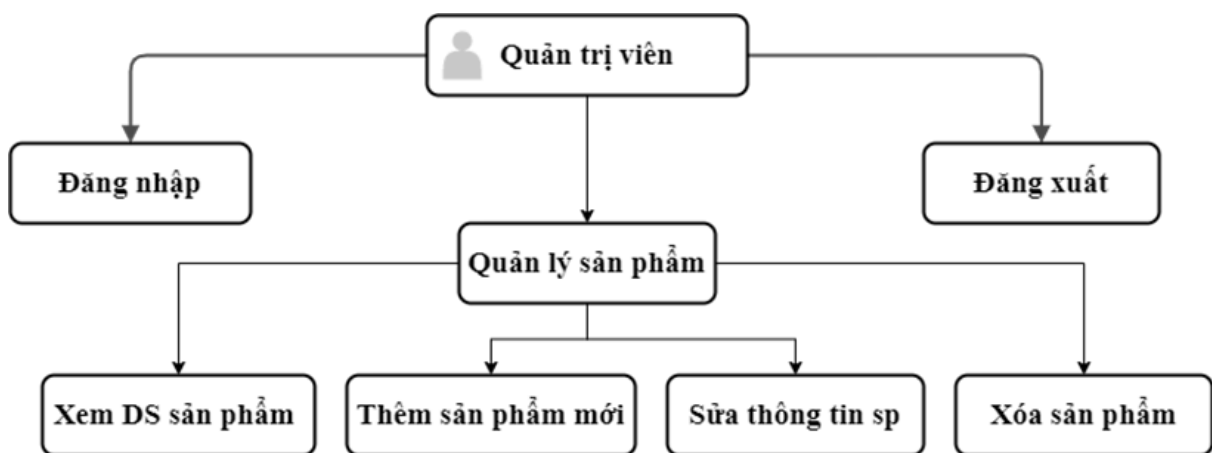
Version: 1.0.0

Tham khảo: Tài liệu phân tích và thiết kế hệ thống thông tin

II. MÔ TẢ CHỨC NĂNG

1. Module dành cho quản trị viên

- Mô tả chức năng:

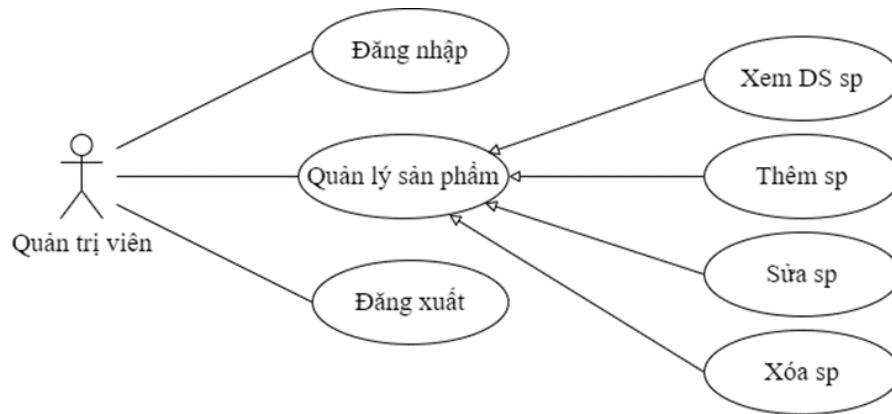


Hình : Sơ đồ phân rã chức năng của module quản trị viên

- **Sơ đồ usecase của tác nhân quản trị viên:**

Quản trị viên có thể:

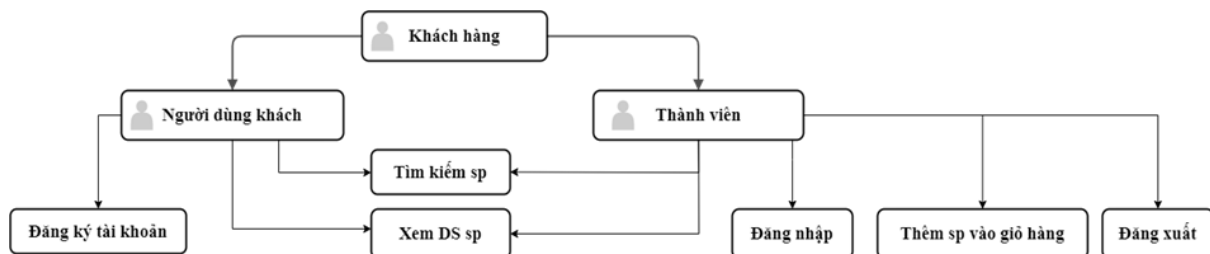
- Đăng nhập, đăng xuất.
- Quản lý sản phẩm: xem danh sách sản phẩm, thêm sản phẩm mới, sửa xóa sản phẩm đang có.



Hình : Sơ đồ trường hợp sử dụng của quản trị viên

2. Module dành cho khách hàng

- Mô tả chức năng



Hình : Sơ đồ phân rã chức năng của module khách hàng

- Sơ đồ use case của tác nhân khách hàng (bao gồm người dùng khách và thành viên)

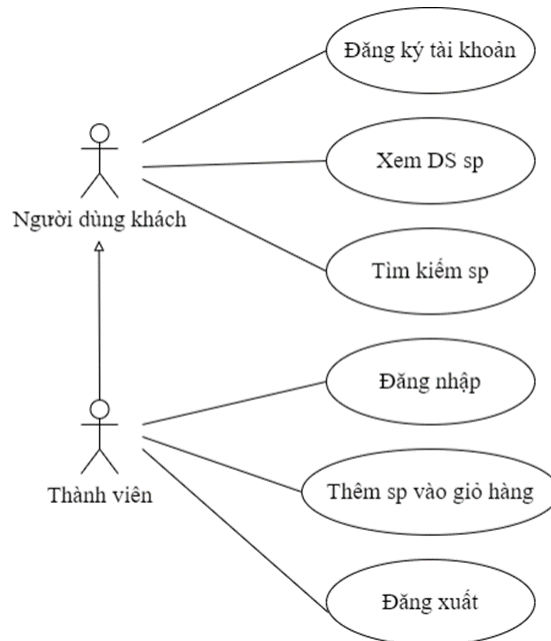
Người dùng khách có thể:

- Xem danh sách sản phẩm: Khách hàng có thể xem toàn bộ danh sách sản phẩm có trên hệ thống.
- Xem chi tiết sản phẩm: Khách hàng có thể xem thông tin chi tiết về một sản phẩm có trên hệ thống.
- Tìm kiếm sản phẩm: Khách hàng có thể tìm kiếm sản phẩm trên hệ thống theo một tiêu chí hoặc từ khóa nào đó.
- Đăng ký tài khoản: Người dùng có thể đăng kí tài khoản khi chưa là thành viên. Đăng ký thành công sẽ có thể đăng nhập và thêm sản phẩm vào giỏ hàng.

Thành viên có thể:

- Đăng nhập: Cho phép thành viên đăng nhập vào hệ thống, sau khi đăng nhập thành công người dùng có thể thêm sản phẩm vào giỏ hàng.
- Xem danh sách sản phẩm.
- Xem chi tiết sản phẩm.

- Tìm kiếm sản phẩm.
- Thêm sản phẩm cần mua vào giỏ hàng.
- Đăng xuất khỏi hệ thống.

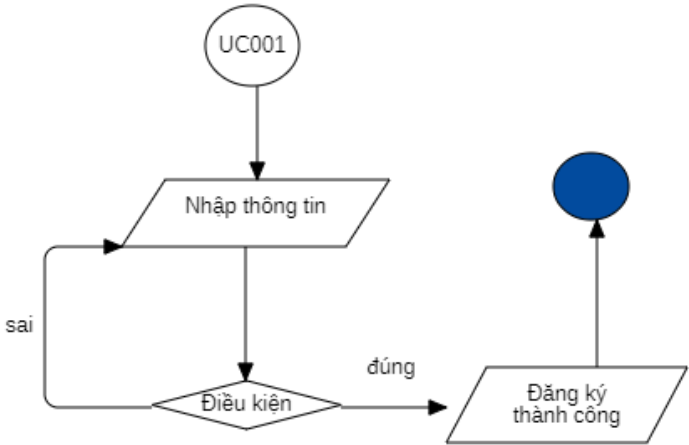


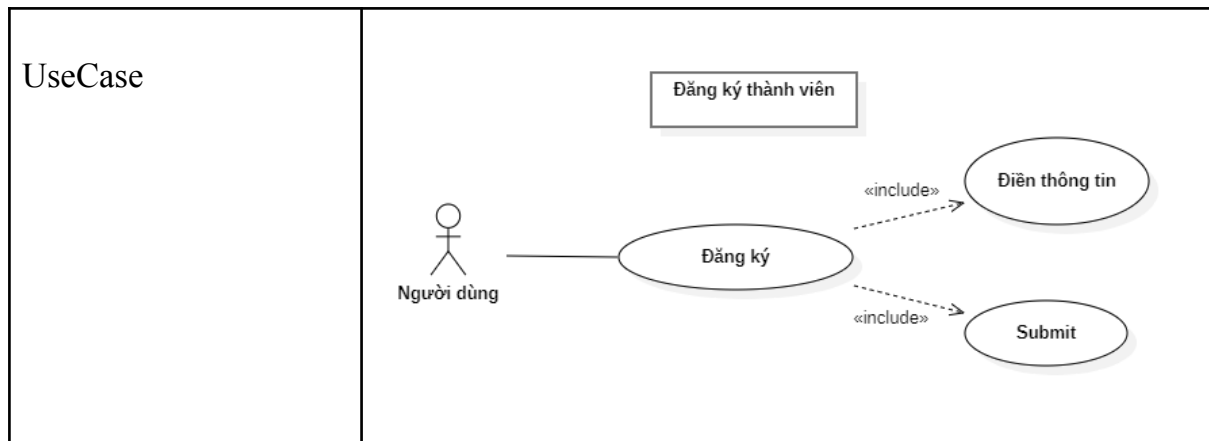
Hình : Sơ đồ trường hợp sử dụng của khách hàng và thành viên

3. Mô tả các usecase

Bảng 1: Mô tả usecase đăng ký tài khoản

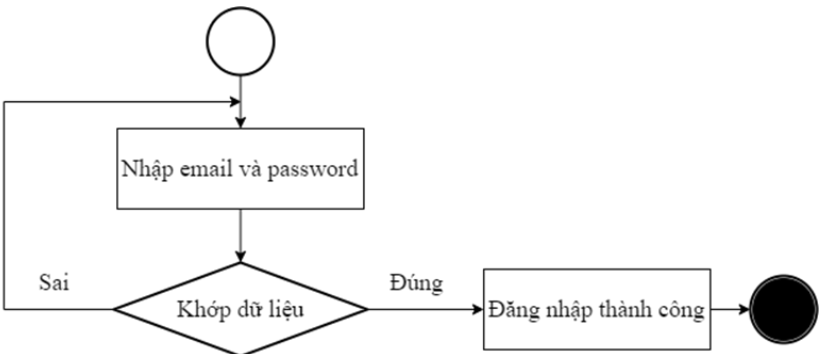
Mã yêu cầu	UC001
Tên yêu cầu	Đăng ký tài khoản
Mức độ ưu tiên	Cao
Nội dung	Cho phép người dùng đăng ký tài khoản với hệ thống nhằm lưu giữ các thông tin như họ tên, số điện thoại, ...
Đối tượng sử dụng	Người dùng
Tiền điều kiện	Email chưa được đăng ký với hệ thống

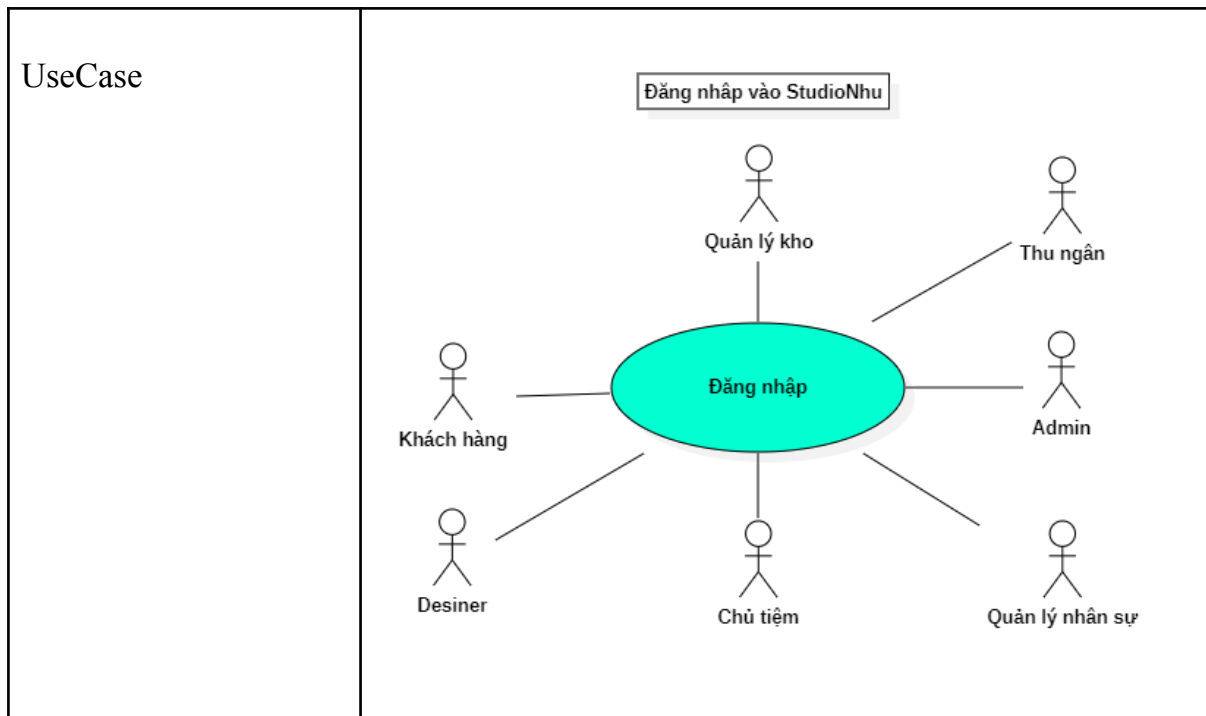
<p>Xử lý</p>	<ol style="list-style-type: none"> 1. Người dùng bấm chọn vào nút đăng ký và điền đầy đủ thông tin form yêu cầu. 2. Hệ thống sẽ kiểm tra các thông tin người dùng nhập vào, nếu bất kỳ thông tin nào của người dùng bị sai hay các thông tin bắt buộc bị bỏ trống, hệ thống sẽ tiến hành hiển thị ra các lỗi tương ứng và người dùng sẽ nhập lại thông tin đó. 3. Sau khi thông tin của người dùng được kiểm tra và hợp lệ, các thông tin đó sẽ được gửi lên server sau đó được lưu vào cơ sở dữ liệu của hệ thống. 4. Sau khi tài khoản của người dùng được tạo ra, một thông báo rằng tài khoản đã được tạo thành công hiện ra. Lúc này, người dùng có thể đăng nhập vào hệ thống.
<p>Kết quả</p>	<p>Người dùng tạo thành công tài khoản trên hệ thống.</p>
<p>Luồng xử lý</p>	 <pre> graph TD UC001((UC001)) --> Input[/Nhập thông tin/] Input --> Decision{Điều kiện} Decision -- sai --> Input Decision -- đúng --> Output[/Đăng ký thành công/] Output --> End(()) style End fill:#000080,color:#fff </pre>



Bảng 2: Mô tả usecase đăng nhập

Mã yêu cầu	UC002
Tên yêu cầu	Đăng nhập
Mức độ ưu tiên	Cao
Nội dung	Cho phép người dùng đăng nhập vào hệ thống
Đối tượng sử dụng	Quản trị viên, thành viên đã có tài khoản
Tiền điều kiện	Thành viên đã đăng ký tài khoản

Xử lý	<ol style="list-style-type: none">1. Người dùng truy cập vào trang web.2. Chọn chức năng đăng nhập.3. Người dùng kiểm tra lại thông tin đăng nhập trước khi bấm chọn đăng nhập.4. Người dùng nhấn chọn đăng nhập.5. Nếu thất bại thì nhận được thông báo rằng tài khoản hoặc mật khẩu không hợp lệ và nhập lại thông tin.6. Khi tên tài khoản và mật khẩu đã hợp lệ, người dùng đăng nhập thành công vào hệ thống.
Kết quả	Quản trị viên, thành viên đăng nhập thành công vào hệ thống
Luồng xử lý	 <pre>graph TD; Start(()) --> Input[Nhập email và password]; Input --> Decision{Khớp dữ liệu}; Decision -- Sai --> Input; Decision -- Đúng --> Success[Đăng nhập thành công]; Success --> End((()));</pre>



Bảng 3: Mô tả usecase thêm sản phẩm vào giỏ hàng

Mã yêu cầu	UC003
Tên yêu cầu	Thêm sản phẩm vào giỏ hàng
Mức độ ưu tiên	Cao
Nội dung	Cho phép người dùng thêm sản phẩm vào giỏ hàng
Đối tượng sử dụng	Thành viên
Tiền điều kiện	Thành viên đã đăng nhập vào hệ thống
Xử lý	1. Người dùng truy cập vào trang web. 2. Tại trang chủ có hiển thị danh sách sản phẩm, người dùng chọn “Mua” tại sản phẩm cần thêm vào giỏ hàng.

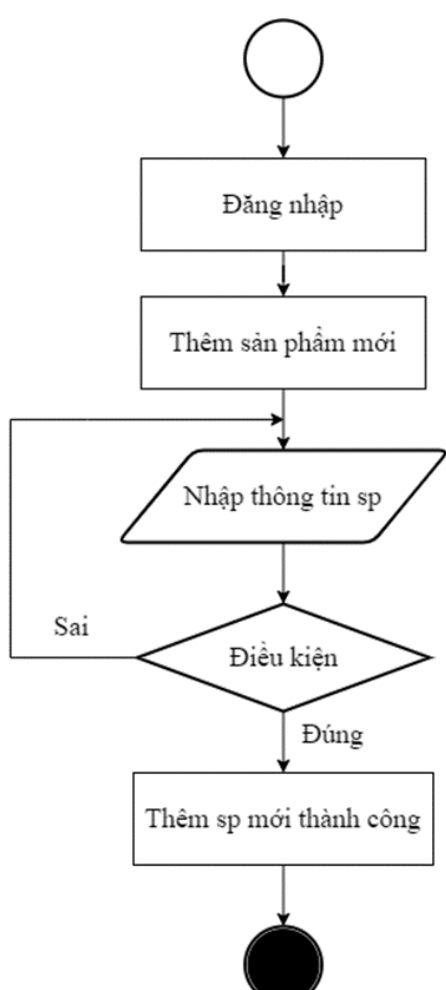
	3. Hệ thống xử lý và đưa ra thông báo thêm thành công. Nếu thất bại sẽ báo lí do và người dùng thực hiện lại thao tác.
Kết quả	Thành viên mua sản phẩm thành công.
Luồng xử lý	
UseCase	

Bảng 4: Mô tả usecase tìm kiếm sản phẩm

Mã yêu cầu	UC004
Tên yêu cầu	Tìm kiếm sản phẩm
Mức độ ưu tiên	Cao
Nội dung	Cho phép người dùng tìm kiếm sản phẩm theo ý muốn
Đối tượng sử dụng	Thành viên
Tiền điều kiện	
Xử lý	<p>1. Người dùng truy cập vào trang web.</p> <p>2. Tại trang chủ, người dùng nhập thông tin cần tìm kiếm vào ô tìm kiếm và nhấn “Tìm”</p> <p>3. Trang web hiển thị kết quả tìm kiếm của người dùng.</p>
Kết quả	Thành viên tìm kiếm sản phẩm thành công
Ghi chú	

Bảng 5: Mô tả usecase quản lý sản phẩm

Mã yêu cầu	UC005
Tên yêu cầu	Quản lý sản phẩm
Mức độ ưu tiên	Cao
Nội dung	Cho phép người quản trị xem danh sách sản phẩm cũng như thêm, sửa, xóa các sản phẩm đó
Đối tượng sử dụng	Người quản trị
Tiền điều kiện	Người quản trị đã đăng nhập
Xử lý	<p>1. Sau khi đăng nhập, hệ thống sẽ dẫn đến trang “Quản lý sản phẩm”.</p> <p>2. Để thêm sản phẩm mới, người quản trị chọn mục “Thêm sản phẩm mới” để hiển thị chức năng thêm sản phẩm mới. Người quản trị nhập tất cả các thông tin cần thiết và nhấn nút “Thêm”. Nếu tất cả các thông tin đã được điền đầy đủ và hợp lệ, một thông báo xuất hiện thông báo thêm sản phẩm mới thành công.</p> <p>3. Trang “Quản lý sản phẩm” đã hiển thị sẵn danh sách tất cả sản phẩm có trên hệ thống theo dạng danh sách theo hàng ngang.</p> <p>3.1. Nếu muốn sửa thông tin của sản phẩm, người quản trị chọn “Sửa” tại hàng của sản phẩm đó. Modal chứa thông tin của sản phẩm đó xuất hiện, người quản trị sửa các thông tin mình muốn và chọn “Lưu” để lưu thay đổi hoặc chọn “Đóng” để hủy thao tác. Nếu chọn “Lưu”, một bảng thông báo xuất hiện xác nhận việc lưu các thay đổi thành công.</p>

	<p>3.2. Nếu muốn xóa sản phẩm, người quản trị chọn “Xóa” tại hàng của sản phẩm đó. Một thông báo xác nhận xóa sản phẩm xuất hiện nhằm đảm bảo người quản trị chắc chắn muốn xóa. Người quản trị chọn “Xóa” để xác nhận xóa hoặc chọn “Không” để hủy thao tác. Nếu chọn “Xóa”, một bảng thông báo xuất hiện xác nhận việc xóa thành công.</p>
Kết quả	Người dùng thêm, sửa thông tin của sản phẩm hoặc xóa sản phẩm thành công.
Luồng xử lý	 <pre> graph TD Start(()) --> A[Đăng nhập] A --> B[Thêm sản phẩm mới] B --> C[/Nhập thông tin sp/] C --> D{Điều kiện} D -- Sai --> C D -- Đúng --> E[Thêm sp mới thành công] E --> End((())) </pre>

TÀI LIỆU API

Tài liệu cập nhật các trường hợp API của ứng dụng. Tài liệu dùng để kiểm tra hoạt động của backend có đúng yêu cầu người dùng.

<Thời gian thực hiện: 27/3/2024 đến 31/3/2024>

<Người thực hiện: Phạm Văn Á>

Version: 1.0.0

A. Kiểm tra sự hoạt động của backend

1. Kiểm tra hoạt động của API

Kiểm tra hoạt động của backend. Khởi chạy server.js để kiểm tra.

URL	http://localhost:3300/
Methods	GET
Header	<none>
Data params	<none>
Success response	Status: 200 OK Content: Trang web đã được thực thi ...
Error response	Không thể truy cập trang web này localhost đã từ chối kết nối.
Simple call	http://localhost:3300/abc http://localhost:3300/12345
Note	PORT có thể thay đổi tùy vào biến môi trường cài đặt (.env) và PORT của server js do phát triển code quy định.

B. Các API của Products (sản phẩm)

1. API trả về toàn bộ sản phẩm

Tất cả thông tin sản phẩm trong dữ liệu được thể hiện tại đây. Tất cả thành viên, người dùng, khách không phải là thành viên đều có thể truy cập xem sản phẩm.

URL	http://localhost:3300/api/products/
Methods	GET

Header	<none>
Data params	<none>
Success response	<p>Status: 200 OK</p> <p>Content:</p> <pre>{ "products": [{ ... }, { "count": ..., "page": ..., "pages": ... }] }</pre>
Error response	Không tìm thấy sản phẩm
Simple call	
Note	<p>Không yêu cầu đăng nhập.</p> <p>Chia trang, số lượng sản phẩm trên 1 trang là 8.</p> <p>Đếm số lượng sản phẩm có trong cơ sở dữ liệu.</p>

2. API trả về một sản phẩm

Trả về thông tin 1 sản phẩm. Tất cả người dùng đều có thể xem được thông tin chi tiết sản phẩm.

URL	<a href="http://localhost:3300/api/products/<id>">http://localhost:3300/api/products/<id>
Methods	GET
Header	<none>
Data params	<p><id></p> <p>ví dụ: 65fa4abebfd8f991ec0f5746</p>
Success response	<p>Status: 200 OK</p> <p>Content:</p> <pre>{ "_id": "...", "user": "...", "name": "...", "image": "...", ... }</pre>

	<pre> "category": "...", "description": "...", "rating": ..., "numReviews": ..., "price": .., "countInStock": .., "reviews": [], "__v": 0, "createdAt": "..", "updatedAt": "... </pre>
Error response	Status: 404 Content: Product not found Không tìm thấy sản phẩm
Simple call	
Note	Thông tin chi tiết 1 sản phẩm.

3. API trả về sản phẩm có lượt đánh giá cao nhất

Trả về kết quả 5 sản phẩm có đánh giá cao nhất.

URL	http://localhost:3300/api/products/top
Methods	GET
Header	<none>
Data params	/top
Success response	Status: 200 OK
Error response	Không tìm thấy sản phẩm
Simple call	
Note	Không yêu đăng nhập

4. API thêm sản phẩm

Thêm một sản phẩm bao gồm: tên, giá, thông tin người tạo, hình ảnh sản phẩm, ...

URL	http://localhost:3300/api/products/
Methods	POST

Header	Content-Type: application/json Authorization: Bearer
Data params	<thông tin các trường của sản phẩm>
Success response	Status: 201
Error response	Status: 404
Simple call	
Note	Cần đăng nhập là thành viên quản trị mới có quyền tạo sản phẩm mới.

5. API cập nhật sản phẩm

URL	http://localhost:3300/api/products/<id>
Methods	PUT
Header	Content-Type: application/json Authorization: Bearer
Data params	<thông tin cập nhật sản phẩm>
Success response	Status: 201
Error response	Status: 404
Simple call	
Note	Cần đăng nhập là thành viên quản trị mới có quyền

6. API xóa sản phẩm

URL	http://localhost:3300/api/products/<id>
Methods	DELETE
Header	Content-Type: application/json Authorization: Bearer
Data params	<id>
Success	Status: 200 OK

response	
Error response	Status: 404
Simple call	
Note	Cần đăng nhập là thành viên quản trị mới có quyền

C. Các API của User (thành viên)

1. API đăng ký thành viên mới

Người dùng bình thường đăng ký vào hệ thống để trở thành thành viên.

URL	http://localhost:3300/api/users
Methods	POST
Header	'Content-Type: application/json'
Params	<none>
data-raw	{ "name":"tên người dùng", "email":"định dạng email", "password":"mật khẩu" }
Success response	Status: 201 Created Content { "_id": "...", "name": "...", "email": "...", "token": "...", "isAdmin":... }
Error response	Test Case 1: userExists Status: 400 Content: "User already exists" Test Case 2: Invalid User ID: 400 Content: "User already exists"
Simple call	
Note	Kiểm tra email đã tồn tại trong cơ sở dữ liệu hay chưa.

2. API đăng nhập

URL	localhost:3300/api/users/login
Methods	POST
Header	<none>
Data params	email, password
Success response	Status: 200 OK Content <thông tin thành viên>
Error response	Status: 401
Simple call	
Note	Cần đăng nhập quản trị để xem thông tin thành viên.

3. API xem thông tin thành viên

URL	http://localhost:3300/api/users
Methods	GET
Header	Content-Type: application/json Authorization: Bearer
Data params	<none>
Success response	Status: 200 OK Content
Error response	Status:
Simple call	
Note	Cần đăng nhập quản trị để xem thông tin thành viên.

4. API cập nhật thành viên

URL	http://localhost:3300/api/users/<id>
Methods	PUT
Header	Content-Type: application/json Authorization: Bearer

Data params	<id>
Success response	Status: 200 OK
Error response	Status: 404
Simple call	<none>
Note	Quản trị có quyền. Lưu thông tin người cập nhật, ngày cập nhật.

5. API xoá thành viên

URL	http://localhost:3300/api/users/<id>
Methods	DELETE
Header	Content-Type: application/json Authorization: Bearer
Data params	<id>
Success response	Status: 200 OK
Error response	Status: 404
Simple call	<none>
Note	Quản trị mới có quyền xóa thành viên. Chuyển thành viên thành trạng thái không hoạt động.

D. Các API của Order (đơn hàng)

1. API thêm đơn hàng

URL	http://localhost:3300/api/orders
Methods	POST
Header	Content-Type: application/json Authorization: Bearer
Data params	
data-type	Thêm các thông tin đơn hàng
Success	Status: 201 Created

response	
Error response	Status: 400 " No order items"
Simple call	
Note	Đăng nhập thành viên để có quyền đặt hàng.

2. API thông tin tất cả đơn hàng

Lấy thông tin đơn hàng.

URL	http://localhost:3300/api/orders
Methods	GET
Header	Content-Type: application/json Authorization: Bearer
Data params	<none>
Success response	Status: 200 OK
Error response	Status: 401
Simple call	
Note	Cần có quyền quản lý

3. API thông tin đơn hàng của thành viên

URL	http://localhost:3300/api/orders/myorders
Methods	GET
Header	Content-Type: application/json Authorization: Bearer
Data params	
Success response	Status: 200 OK
Error response	Status: 200
Simple call	
Note	

4. API cập nhật đơn hàng

URL	http://localhost:3300/api/orders/<id>/pay
Methods	GET
Header	Content-Type: application/json Authorization: Bearer
Data params	<id>
Success response	
Error response	
Simple call	
Note	

Tham khảo:

<https://www.youtube.com/watch?v=cR-MgEyvY-M>

<https://stoplight.io/api-documentation-guide>

<https://viblo.asia/p/viet-api-document-cuc-don-gian-GrLZDJ4e5k0>