**TRƯỜNG ĐẠI HỌC VINH**

**VIỆN KỸ THUẬT VÀ CÔNG NGHỆ**

****

**ĐỒ ÁN HỌC PHẦN**

**CÁC HỆ CƠ SỞ DỮ LIỆU**

**TIÊN TIẾN**

**XÂY DỰNG WEBSITE**

**CÔNG THỨC NẤU ĂN**

**VỚI NODE.JS VÀ MONGODB**

**NHÓM: 2**

|  |  |
| --- | --- |
| GVHD: | TS. Phan Anh Phong |
| SVTH: | Trương Trung Dũng, 215748020110496 - NT |
|  | Phạm Quang Hưng, 215748020110030 |
|  | Bùi Văn Oai, 215748020110360 |

**Nghệ An, 06/202****5**

# LỜI CẢM ƠN

Chúng em xin gửi lời cảm ơn chân thành và sâu sắc nhất đến thầy Phan Anh Phong, giảng viên Trường Đại học Vinh, người đã tận tình hướng dẫn và hỗ trợ chúng em trong suốt quá trình thực hiện đề tài "Xây dựng website công thức nấu ăn". Sự giúp đỡ tận tâm của thầy đã đóng góp một phần quan trọng trong việc định hướng, cải thiện và hoàn thiện đề tài, đồng thời tạo điều kiện để chúng em rèn luyện, nâng cao kiến thức và kỹ năng thực tế của bản thân.

Trong suốt thời gian thực hiện đề tài, thầy không chỉ truyền đạt cho chúng em những kiến thức chuyên môn cần thiết, mà còn chia sẻ nhiều kinh nghiệm thực tiễn quý báu, giúp chúng em giải quyết các vấn đề phát sinh và vượt qua những khó khăn trong quá trình nghiên cứu và triển khai hệ thống.

Không chỉ dừng lại ở việc hỗ trợ về mặt học thuật, thầy còn luôn động viên và khích lệ tinh thần chúng em trong những giai đoạn thử thách, khi dự án gặp phải những trở ngại tưởng chừng khó vượt qua. Chính sự tận tâm, kiên nhẫn và chu đáo của thầy đã trở thành nguồn động lực to lớn, giúp chúng em nỗ lực không ngừng để hoàn thành đồ án một cách tốt nhất. Thầy đã tạo điều kiện để chúng em được tự do sáng tạo, thử nghiệm các giải pháp mới, đồng thời luôn sẵn sàng lắng nghe và hướng dẫn mỗi khi chúng em cần sự hỗ trợ.

Một lần nữa, chúng em xin bày tỏ lòng biết ơn sâu sắc đến thầy Phan Anh Phong. Nhờ sự chỉ dẫn tận tình của thầy, chúng em không chỉ hoàn thành tốt đề tài, mà còn tích lũy được nhiều kinh nghiệm quý giá cho hành trình học tập và phát triển sau này. Chúng em hy vọng trong tương lai sẽ tiếp tục nhận được sự chỉ bảo, định hướng và đồng hành của thầy để không ngừng hoàn thiện bản thân.

Kính chúc thầy luôn dồi dào sức khỏe, giữ vững nhiệt huyết với nghề và gặt hái nhiều thành công trong sự nghiệp giảng dạy cũng như nghiên cứu khoa học.

Một lần nữa, chúng em xin chân thành cảm ơn thầy!

**MỤC LỤC**

[LỜI CẢM ƠN 1](#_Toc199939142)

[DANH MỤC CÁC HÌNH 4](#_Toc199939143)

[DANH MỤC CÁC BẢNG 6](#_Toc199939144)

[CHƯƠNG 1. GIỚI THIỆU ĐỀ TÀI 7](#_Toc199939145)

[1.1. Lý do chọn đề tài 7](#_Toc199939146)

[1.2. Mục đích nghiên cứu 7](#_Toc199939147)

[1.3. Phạm vi nghiên cứu 8](#_Toc199939148)

[1.4. Phương pháp thực hiện 9](#_Toc199939149)

[CHƯƠNG 2. CƠ SỞ LÝ THUYẾT 10](#_Toc199939150)

[2.1. Mô tả tổng quan website 10](#_Toc199939151)

[2.2. Giới thiệu MongoDB 12](#_Toc199939152)

[2.3. Các công nghệ hỗ trợ khác 17](#_Toc199939153)

[2.4. Một số ứng dụng liên quan đến đề tài 20](#_Toc199939154)

[CHƯƠNG 3. PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG 23](#_Toc199939155)

[3.1. Phân tích yêu cầu và xác định các tác nhân 23](#_Toc199939156)

[3.2. Sprint 1 Quản lí nguyên liệu 25](#_Toc199939157)

[3.3. Sprint 2: Quản lí công thức món ăn 30](#_Toc199939158)

[3.4. Sprint 3: Quản lí người dùng 36](#_Toc199939159)

[CHƯƠNG 4. XÂY DỰNG HỆ THỐNG 43](#_Toc199939160)

[4.1. Môi trường phát triển 43](#_Toc199939161)

[4.2. Quản lý dự án trên Asana và Github 45](#_Toc199939162)

[4.3. Xây dựng chức năng quản lý nguyên liệu 47](#_Toc199939163)

[4.4. Xây dựng chức năng quản lý công thức 49](#_Toc199939164)

[4.5. Xây dựng chức năng quản lý người dùng 51](#_Toc199939165)

[KẾT LUẬN 54](#_Toc199939166)

[1. Kết quả đạt được 54](#_Toc199939167)

[2. Hạn chế 54](#_Toc199939168)

[3. Hướng phát triển 55](#_Toc199939169)

[BẢNG PHÂN CHIA CÔNG VIỆC 56](#_Toc199939170)

[TÀI LIỆU THAM KHẢO 57](#_Toc199939171)

# DANH MỤC CÁC HÌNH

[Hình 2.1. So sánh tổng quan giữa mongoDB và MySql 12](#_Toc199939180)

[Hình 2.2. Cấu trúc dữ liệu mongoDB 14](#_Toc199939181)

[Hình 2.3. Nodejs, express 17](#_Toc199939182)

[Hình 2.4. Cấu trúc liên kết của mongoose với nodejs 18](#_Toc199939183)

[Hình 2.5. Ứng dụng công thức nấu ăn Cookpad 20](#_Toc199939184)

[Hình 2.6. Ứng dụng chia sẻ công thức nấu ăn Allrecipes 20](#_Toc199939185)

[Hình 2.7. Ứng dụng chia sẻ công thức nấu ăn Tasty 21](#_Toc199939186)

[Hình 3.1. Biểu đồ ca sử dụng quản lý nguyên liệu 24](#_Toc199939187)

[Hình 3.2. Biểu đồ tuần tự quản lý nguyên liệu 28](#_Toc199939188)

[Hình 3.3. Giao diện mẫu danh sách nguyên liệu 29](#_Toc199939189)

[Hình 3.4. Biểu đồ ca sử dụng quản lý công thức món ăn 30](#_Toc199939190)

[Hình 3.5. Biểu đồ tuần tự quản lý công thức món ăn 35](#_Toc199939191)

[Hình 3.6. Giao diện mẫu danh sách công thức món ăn 35](#_Toc199939192)

[Hình 3.7. Biểu đồ ca sử dụng quản lý người dùng 36](#_Toc199939193)

[Hình 3.8. Biểu đồ tuần tự quản lý người dùng 40](#_Toc199939194)

[Hình 3.9. Giao diện mẫu danh sách người dùng 41](#_Toc199939195)

[Hình 4.1. Giao diện mẫu dùng cho khách hàng 43](#_Toc199939196)

[Hình 4.2. Giao diện trang chủ admin 44](#_Toc199939197)

[Hình 4.3. Công cụ quản lý dự án Asana 45](#_Toc199939198)

[Hình 4.4. Công cụ quản lý mã nguồn Github 45](#_Toc199939199)

[Hình 4.5. Giao diện quản lý nguyên liệu 46](#_Toc199939200)

[Hình 4.6. Giao diện quản lý công thức 49](#_Toc199939201)

# DANH MỤC CÁC BẢNG

[Bảng 3.1. Bảng lưu trữ thông tin nguyên liệu 25](#_Toc199939328)

[Bảng 3.2. Bảng lưu trữ thông tin công thức món ăn 31](#_Toc199939329)

[Bảng 3.3. Bảng lưu trữ thông tin người dùng 37](#_Toc199939330)

[Bảng 4.1. Kiểm thử các chức năng quản lý nguyên liệu 47](#_Toc199939331)

[Bảng 4.2. Kiểm thử các chức năng quản lý công thức 50](#_Toc199939332)

[Bảng 4.3. Kiểm thử các chức năng quản lý người dùng 52](#_Toc199939333)

# CHƯƠNG 1. GIỚI THIỆU ĐỀ TÀI

## 1.1. Lý do chọn đề tài

Việc chọn đề tài "Xây dựng website công thức nấu ăn" xuất phát từ nhu cầu thực tế của thị trường và xu hướng phát triển công nghệ hiện nay như:

- Nhu cầu thực tế: Ẩm thực là một lĩnh vực luôn nhận được sự quan tâm lớn từ cộng đồng. Trong cuộc sống hiện đại, nhu cầu tìm kiếm, chia sẻ và lưu trữ các công thức nấu ăn qua các website ngày càng trở nên thiết yếu, giúp mỗi cá nhân dễ dàng tiếp cận tri thức về ẩm thực, làm phong phú thực đơn gia đình cũng như học hỏi những bí quyết nấu ăn mới. Đây chính là nhu cầu thực tế xuất phát từ chính cộng đồng người dùng.

- Tiềm năng phát triển: Xu hướng chuyển đổi số và sự phát triển mạnh mẽ của công nghệ thông tin đã mở ra tiềm năng lớn cho việc phát triển các nền tảng số trong lĩnh vực ẩm thực. Các website công thức nấu ăn không chỉ đơn thuần là nơi lưu trữ thông tin mà còn có thể trở thành diễn đàn chia sẻ kinh nghiệm, kết nối cộng đồng, hoặc tích hợp nhiều tiện ích thông minh trong tương lai như cá nhân hóa đề xuất món ăn, lập kế hoạch bữa ăn, mua sắm nguyên liệu trực tuyến… Nhận thấy nhu cầu thực tế và tiềm năng phát triển như vậy, nhóm chúng tôi quyết định lựa chọn đề tài “Xây dựng website công thức nấu ăn” cho dự án học phần này.

## 1.2. Mục đích nghiên cứu

- Xây dựng một website cho phép người dùng dễ dàng tra cứu, thêm mới, chỉnh sửa, xóa và chia sẻ các công thức nấu ăn một cách thuận tiện và linh hoạt.

- Thiết kế website hỗ trợ người dùng đăng ký, đăng nhập và quản lý tài khoản cá nhân, đảm bảo mỗi người dùng có thể lưu trữ, chỉnh sửa và bảo mật thông tin cá nhân cũng như các công thức của riêng mình.

- Nghiên cứu, tìm hiểu và áp dụng các công nghệ web hiện đại như Node.js, Express.js, MongoDB, Mongoose, EJS, SCSS và JavaScript vào quá trình phát triển website.

- Kết hợp các công nghệ để sản phẩm có hiệu năng tốt, khả năng mở rộng cao, đồng thời nâng cao hiểu biết thực tiễn về lập trình backend, frontend và quản trị cơ sở dữ liệu NoSQL cho các thành viên trong nhóm.

- Tạo ra một sản phẩm hoàn chỉnh, có thể vận hành thực tế, giúp các thành viên rèn luyện kỹ năng làm việc nhóm, phân tích và giải quyết vấn đề, tiếp cận với quy trình phát triển phần mềm hiện đại.

- Góp phần cung cấp một công cụ hữu ích phục vụ cộng đồng yêu thích ẩm thực, đồng thời tích lũy thêm kinh nghiệm, chuẩn bị hành trang cho những dự án lớn hơn trong tương lai.

## 1.3. Phạm vi nghiên cứu

- Đề tài tập trung vào việc xây dựng các chức năng cơ bản cho một website công thức nấu ăn. Cụ thể, hệ thống sẽ cho phép người dùng đăng ký và đăng nhập tài khoản, đảm bảo mỗi người dùng có không gian cá nhân để quản lý các công thức do mình tạo ra. Bên cạnh đó, website hỗ trợ chức năng quản lý công thức nấu ăn với các thao tác cơ bản như thêm mới, chỉnh sửa, xóa và xem chi tiết từng công thức. Như vậy, người dùng có thể dễ dàng tạo ra bộ sưu tập công thức của riêng mình cũng như cập nhật, chỉnh sửa thông tin món ăn một cách thuận tiện.

- Phạm vi của dự án chỉ tập trung vào các chức năng quản lý cơ bản, chưa triển khai các tính năng nâng cao như tìm kiếm thông minh theo nguyên liệu, gợi ý món ăn, đánh giá hoặc bình luận công thức, cũng như chưa tích hợp các dịch vụ bên ngoài như mạng xã hội hay thanh toán trực tuyến. Điều này giúp nhóm có điều kiện tập trung hoàn thiện và tối ưu các chức năng cốt lõi, đảm bảo hệ thống hoạt động ổn định và dễ bảo trì.

- Về mặt công nghệ, website được phát triển dựa trên các công nghệ web phổ biến hiện nay như Node.js, Express.js cho phần backend, MongoDB và Mongoose cho quản trị và truy vấn cơ sở dữ liệu, EJS và SCSS hỗ trợ xây dựng giao diện người dùng. Việc lựa chọn các công nghệ này nhằm mục đích phù hợp với mục tiêu ứng dụng thực tiễn cũng như đáp ứng khả năng học hỏi, nghiên cứu của các thành viên trong nhóm.

- Dự án triển khai và kiểm thử chủ yếu trong môi trường máy chủ nội bộ hoặc local, chưa tiến hành triển khai chính thức lên môi trường Internet hoặc các dịch vụ hosting công cộng. Điều này giúp nhóm kiểm soát tốt hơn quá trình phát triển, dễ dàng sửa lỗi và hoàn thiện sản phẩm trong phạm vi cho phép của đề tài học phần.

- Tổng kết lại, phạm vi nghiên cứu của đề tài vừa đủ để nhóm tập trung nghiên cứu, phát triển các chức năng thiết yếu, đảm bảo hoàn thành mục tiêu học tập đồng thời tạo ra một sản phẩm có tính thực tiễn, có thể mở rộng và phát triển thêm trong tương lai nếu có điều kiện

## 1.4. Phương pháp thực hiện

- Đầu tiên, nhóm tiến hành nghiên cứu tổng quan về các công nghệ sẽ sử dụng trong dự án, bao gồm Node.js, Express.js, MongoDB, Mongoose, EJS, SCSS và JavaScript. Việc tìm hiểu này giúp các thành viên nắm vững kiến thức nền tảng, hiểu rõ vai trò cũng như cách tích hợp các công nghệ vào quá trình xây dựng website công thức nấu ăn.

- Tiếp theo, nhóm phân tích yêu cầu của đề tài, xác định rõ các chức năng chính cần thực hiện như quản lý tài khoản người dùng, quản lý công thức nấu ăn, và giao diện người dùng. Trên cơ sở đó, nhóm xây dựng sơ đồ chức năng, phác thảo giao diện, đồng thời thiết kế cấu trúc cơ sở dữ liệu MongoDB phù hợp với đặc điểm dữ liệu của các công thức nấu ăn.

- Sau khi đã hoàn thiện khâu thiết kế, nhóm bắt đầu lập trình xây dựng website. Quá trình phát triển theo mô hình MVC (Model - View - Controller), trong đó backend sử dụng Node.js và Express.js để xử lý logic, frontend sử dụng EJS và SCSS để xây dựng giao diện, còn MongoDB kết hợp Mongoose đảm nhận vai trò lưu trữ và truy xuất dữ liệu. Các chức năng được xây dựng từng bước, vừa lập trình vừa kiểm thử để phát hiện và sửa lỗi kịp thời.

- Khi các chức năng cơ bản đã hoàn thành, nhóm tiến hành kiểm thử toàn bộ hệ thống. Việc kiểm thử được thực hiện với nhiều kịch bản khác nhau nhằm đảm bảo các tính năng hoạt động ổn định, đáp ứng đúng yêu cầu đề ra. Những lỗi hoặc điểm chưa hợp lý sẽ được ghi nhận và sửa chữa ngay trong quá trình này.

# CHƯƠNG 2. CƠ SỞ LÝ THUYẾT

Hiện nay, việc chia sẻ và quản lý công thức nấu ăn tại nhiều gia đình cũng như cộng đồng yêu thích ẩm thực vẫn chủ yếu được thực hiện theo phương pháp truyền thống như ghi chép bằng tay, lưu trữ trong sổ tay cá nhân hoặc đăng tải rải rác trên các mạng xã hội, blog, hoặc file Word, Excel. Cách làm này dẫn đến nhiều bất cập trong việc tìm kiếm, hệ thống hóa và chia sẻ kiến thức ẩm thực. Các công thức dễ bị thất lạc, khó tổng hợp, và việc cập nhật, chỉnh sửa hoặc trao đổi thông tin giữa những người cùng quan tâm cũng không được thuận tiện. Ngoài ra, việc phân loại, đánh giá chất lượng công thức cũng gặp nhiều khó khăn, khiến người dùng khó tiếp cận được những công thức phù hợp với nhu cầu hoặc khẩu vị riêng của mình.

Bên cạnh đó, nhu cầu tìm kiếm, lưu trữ và chia sẻ công thức nấu ăn ngày càng trở nên phổ biến trong xã hội hiện đại, nhất là khi xu hướng tự nấu ăn tại nhà, sáng tạo món mới và giao lưu văn hóa ẩm thực phát triển mạnh mẽ. Tuy nhiên, sự thiếu hụt các nền tảng trực tuyến chuyên biệt, tiện lợi cho việc quản lý, tìm kiếm và tương tác với kho công thức nấu ăn đã phần nào hạn chế khả năng tiếp cận, học hỏi và giao lưu của cộng đồng yêu ẩm thực.

Trong chương này, chúng ta sẽ phân tích những vấn đề tồn tại của phương pháp quản lý công thức nấu ăn truyền thống, đồng thời làm rõ nhu cầu cấp thiết của việc xây dựng một hệ thống website công thức nấu ăn hiện đại. Từ đó, bài toán số hóa và tối ưu hóa việc quản lý, chia sẻ công thức nấu ăn sẽ được xác định rõ ràng, tạo tiền đề cho việc phát triển hệ thống website hoàn chỉnh ở các chương tiếp theo.

## 2.1. Mô tả tổng quan website

Website công thức nấu ăn được phát triển dựa trên kiến trúc Client – Server hiện đại, một mô hình phổ biến mang lại hiệu quả cao trong việc tổ chức và vận hành các hệ thống web. Trong mô hình này, toàn bộ hệ thống được chia thành hai phần rõ rệt: phía máy khách (Client) và phía máy chủ (Server). Máy khách là nơi người dùng trực tiếp thao tác với website, thông qua trình duyệt web trên máy tính hoặc thiết bị di động. Khi người dùng muốn đăng ký tài khoản, đăng nhập, tìm kiếm công thức, thêm công thức mới hoặc chỉnh sửa thông tin món ăn, tất cả các thao tác này sẽ được thực hiện trên giao diện client và tạo thành các yêu cầu (request) gửi đến server. Phía máy chủ (server) tiếp nhận các yêu cầu đó, xử lý logic nghiệp vụ như xác thực thông tin người dùng, kiểm tra dữ liệu đầu vào, truy xuất hoặc cập nhật dữ liệu trong hệ quản trị cơ sở dữ liệu, sau đó phản hồi (response) lại kết quả cho client để hiển thị cho người dùng. Việc phân tách này giúp hệ thống vận hành ổn định, bảo mật và hỗ trợ hàng trăm, hàng nghìn người dùng truy cập đồng thời mà không xảy ra xung đột thông tin. Ngoài ra, kiến trúc Client – Server còn cho phép dễ dàng mở rộng, tích hợp thêm các tính năng mới hoặc nâng cấp hệ thống trong tương lai mà không gây ảnh hưởng đến tổng thể.

Phần frontend (giao diện người dùng) là bộ mặt của website, nơi mọi trải nghiệm của người dùng được thể hiện. Trong dự án này, frontend được thiết kế bằng EJS (Embedded JavaScript), SCSS và JavaScript. Việc sử dụng EJS giúp các trang web có thể hiển thị nội dung động, cá nhân hóa theo từng tài khoản hoặc từng công thức cụ thể, đồng thời vẫn đảm bảo tốc độ tải trang nhanh và khả năng mở rộng. SCSS là một phần mở rộng của CSS truyền thống, cho phép tổ chức mã nguồn giao diện một cách khoa học, dễ bảo trì, dễ chỉnh sửa khi cần thay đổi thiết kế. JavaScript đóng vai trò quan trọng trong việc xử lý các tương tác động như xác thực dữ liệu đầu vào trên trình duyệt, cập nhật nội dung trang mà không cần tải lại (AJAX), hiển thị thông báo, điều hướng linh hoạt giữa các trang. Người dùng có thể dễ dàng tìm kiếm, lọc công thức theo từng tiêu chí, xem chi tiết món ăn, thêm mới, chỉnh sửa hoặc xóa công thức, quản lý thông tin cá nhân và chia sẻ công thức với cộng đồng. Tất cả các thao tác này đều được truyền tải qua giao diện trực quan, thân thiện và tối ưu trải nghiệm người dùng.

Phần backend (xử lý nghiệp vụ và quản lý dữ liệu) là "bộ não" của hệ thống, chịu trách nhiệm tiếp nhận và xử lý mọi yêu cầu từ frontend. Backend của website được xây dựng dựa trên Node.js và Express.js, hai công nghệ mạnh mẽ giúp phát triển web server có khả năng đáp ứng đồng thời nhiều yêu cầu với hiệu năng cao. Express.js cung cấp một framework linh hoạt, cho phép xây dựng các route (đường dẫn API), xử lý các lớp middleware, phân quyền truy cập và tổ chức logic nghiệp vụ một cách có hệ thống. Mỗi khi nhận được yêu cầu từ client, backend sẽ kiểm tra quyền truy cập, xác thực thông tin, xử lý logic như thêm mới, chỉnh sửa, xóa hoặc tìm kiếm công thức nấu ăn, đồng thời thực hiện các thao tác bảo mật cần thiết như mã hóa mật khẩu, xác thực phiên đăng nhập. Tất cả các dữ liệu như tài khoản người dùng, công thức nấu ăn, nguyên liệu, hình ảnh minh họa… đều được lưu trữ tập trung trong cơ sở dữ liệu MongoDB. Thư viện Mongoose đóng vai trò trung gian, giúp định nghĩa cấu trúc dữ liệu (schema), kiểm soát tính hợp lệ của dữ liệu và thực hiện các thao tác truy vấn, cập nhật một cách hiệu quả và an toàn. Backend còn đảm nhiệm việc bảo vệ dữ liệu cá nhân, kiểm soát phân quyền và đảm bảo an toàn cho mọi thông tin của người dùng.

Quá trình trao đổi dữ liệu giữa frontend và backend diễn ra liên tục, bảo mật và đồng bộ. Khi người dùng thực hiện một thao tác (chẳng hạn như tìm kiếm công thức mới, đăng nhập, hoặc chia sẻ công thức), frontend sẽ gửi dữ liệu lên backend dưới dạng các HTTP request (GET, POST, PUT, DELETE). Backend tiếp nhận yêu cầu, kiểm tra tính hợp lệ, truy xuất hoặc cập nhật dữ liệu trong cơ sở dữ liệu, sau đó trả về phản hồi (thường là dữ liệu JSON hoặc trang HTML động) cho frontend. Frontend tiếp nhận phản hồi này, cập nhật giao diện, hiển thị thông báo và nội dung phù hợp cho người dùng. Quá trình này giúp dữ liệu luôn được đồng bộ, các thay đổi được cập nhật tức thời và đảm bảo trải nghiệm người dùng liên tục, không bị gián đoạn. Nhờ mô hình này, mọi thao tác của người dùng đều được xử lý nhanh chóng, chính xác và an toàn.

Việc phát triển website công thức nấu ăn theo mô hình Client – Server với sự phân tách rõ ràng giữa frontend và backend mang lại nhiều ưu điểm vượt trội. Hệ thống dễ dàng bảo trì, nâng cấp và mở rộng, bởi mỗi thành phần có thể được phát triển độc lập. Tính bảo mật của hệ thống được nâng cao nhờ kiểm soát tốt việc truy cập và xử lý dữ liệu nhạy cảm. Sử dụng các công nghệ hiện đại như Node.js, Express.js, MongoDB, Mongoose, EJS, SCSS và JavaScript giúp hệ thống đạt hiệu năng cao, giao diện linh hoạt, thân thiện với người dùng. Ngoài ra, kiến trúc này còn tạo điều kiện thuận lợi để tích hợp thêm các chức năng nâng cao như upload hình ảnh, đánh giá, bình luận, tìm kiếm thông minh hoặc mở rộng sang phát triển ứng dụng di động trong tương lai. Nhờ đó, website không chỉ đáp ứng tốt nhu cầu quản lý, chia sẻ công thức nấu ăn hiện tại mà còn có tiềm năng phát triển mạnh mẽ, phục vụ cộng đồng người yêu ẩm thực ngày càng đông đảo

## 2.2. Giới thiệu MongoDB

MongoDB là một hệ quản trị cơ sở dữ liệu NoSQL phổ biến, được thiết kế để lưu trữ dữ liệu dưới dạng tài liệu linh hoạt theo cấu trúc tương tự JSON. Khác với các hệ quản trị cơ sở dữ liệu quan hệ truyền thống, MongoDB không sử dụng bảng mà tổ chức dữ liệu thành các bộ sưu tập (collection) và tài liệu (document), giúp lưu trữ và truy xuất dữ liệu nhanh chóng, dễ dàng mở rộng. Nhờ khả năng quản lý dữ liệu phi cấu trúc và tích hợp tốt với các công nghệ web hiện đại, MongoDB ngày càng được ưa chuộng trong phát triển các ứng dụng web, đặc biệt là những hệ thống cần lưu trữ dữ liệu lớn, đa dạng và thường xuyên thay đổi như website công thức nấu ăn.

### 2.2.1. MongoDB là gì? So sánh với MySQL

MongoDB là một hệ quản trị cơ sở dữ liệu NoSQL mã nguồn mở, được phát triển để lưu trữ và quản lý dữ liệu dưới dạng tài liệu (document) thay vì các bảng (table) như trong hệ quản trị cơ sở dữ liệu quan hệ (SQL). Dữ liệu trong MongoDB được lưu dưới dạng BSON (Binary JSON), có cấu trúc linh hoạt và tương tự JSON, giúp dễ dàng mở rộng, cập nhật và phù hợp với các ứng dụng hiện đại yêu cầu xử lý khối lượng dữ liệu lớn, dữ liệu phi cấu trúc hoặc thường xuyên thay đổi. MongoDB cho phép lưu trữ dữ liệu theo mô hình collection (bộ sưu tập) và document (tài liệu), giúp tối ưu hóa tốc độ đọc/ghi và hiệu quả mở rộng hệ thống.

**Hình 2.1**. So sánh tổng quan giữa mongoDB và MySql

|  |  |  |
| --- | --- | --- |
| Tiêu chí | MongoDB | Mysql |
| Loại cơ sở dữ liệu | NoSQL | Quan hệ |
| Cách lưu trữ dữ liệu | Collection, Document | Table, Row, Column |
| Cấu trúc dữ liệu | Linh hoạt, không cần schema cố định | Cần định nghĩa schema rõ ràng trước khi lưu dữ liệu |
| Tính mở rộng | Dễ dàng mở rộng theo chiều ngang | Chủ yếu mở rộng theo chiều dọc |
| Hỗ trợ quan hệ hoặc liên kết | Yếu, không hỗ trợ JOIN phức tạp, chủ yếu nhúng hoặc liên kết | Mạnh, hỗ trợ JOIN phức tạp, ràng buộc khóa ngoại |
| Truy vấn dữ liệu | MongoDB Query Language (MQL), cú pháp giống JavaScript | SQL |
| Tốc độ xử lý | Nhanh với dữ liệu lớn, phi cấu trúc, truy vấn phân tán tốt | Ổn định với dữ liệu có cấu trúc, truy vấn quan hệ mạnh |
| Tính linh hoạt | Cao, có thể thay đổi cấu trúc document bất cứ lúc nào | Thấp, thay đổi schema phức tạp và rủi ro |
| Hỗ trợ ngôn ngữ lập trình | JavaScript/Node.js, Python, Java, Go, … | PHP, Java, Python, C#, Ruby, ... |

### 2.2.2. Cấu trúc dữ liệu (collection, document)

MongoDB lưu trữ dữ liệu theo mô hình tài liệu (document) thay vì bảng (table) như các hệ quản trị cơ sở dữ liệu quan hệ truyền thống. Dữ liệu trong MongoDB được tổ chức thành các bộ sưu tập (collection), mỗi bộ sưu tập chứa nhiều tài liệu (document). Mỗi document là một đối tượng dữ liệu dưới dạng BSON (Binary JSON), có cấu trúc linh hoạt, cho phép lưu trữ các trường (field) với kiểu dữ liệu khác nhau, thậm chí các tài liệu trong cùng một bộ sưu tập cũng không bắt buộc phải có cấu trúc giống nhau.

A screenshot of a computer

AI-generated content may be incorrect.

**Hình 2.2**. Cấu trúc dữ liệu mongoDB

Database (Cơ sở dữ liệu): MongoDB có thể chứa nhiều database khác nhau. Mỗi database giống như một nhóm dữ liệu riêng biệt.

Collection (Bộ sưu tập): Bên trong mỗi database là các collection. Collection có thể hiểu tương tự như bảng (table) trong các hệ quản trị cơ sở dữ liệu quan hệ, nhưng khác biệt mấu chốt là không cần định nghĩa cấu trúc dữ liệu trước.

Document (Tài liệu): Đây là đơn vị nhỏ nhất của dữ liệu trong MongoDB, có cấu trúc dưới dạng BSON. Mỗi document có thể chứa các trường (key-value pairs), mảng, hoặc thậm chí các tài liệu lồng nhau (embedded document).

### 2.2.3. Ưu điểm của MongoDB trong xây dựng web

MongoDB được đánh giá là một trong những hệ quản trị cơ sở dữ liệu NoSQL phổ biến nhất hiện nay, đặc biệt phù hợp với các ứng dụng web hiện đại. Một trong những ưu điểm nổi bật nhất của MongoDB chính là cấu trúc dữ liệu linh hoạt. Khác với các hệ quản trị cơ sở dữ liệu quan hệ như MySQL, MongoDB không yêu cầu phải định nghĩa trước một schema cố định cho dữ liệu. Điều này cho phép các nhà phát triển dễ dàng thay đổi, bổ sung hoặc loại bỏ các trường thông tin trong quá trình phát triển mà không ảnh hưởng đến toàn bộ hệ thống. Nhờ vậy, MongoDB rất phù hợp với những website thường xuyên cập nhật chức năng, mở rộng tính năng hoặc có dữ liệu đầu vào đa dạng, điển hình như các website công thức nấu ăn, mạng xã hội, blog cá nhân, thương mại điện tử,…

Một ưu điểm khác của MongoDB là khả năng xử lý dữ liệu lớn và tốc độ truy xuất cao. MongoDB tối ưu hóa các thao tác đọc/ghi nhờ lưu trữ dữ liệu dưới dạng tài liệu (document) và hỗ trợ truy vấn đồng thời hiệu quả. Điều này giúp các website có lượng truy cập lớn, thường xuyên phải xử lý nhiều truy vấn cùng lúc vẫn có thể duy trì hiệu năng ổn định và đáp ứng nhanh yêu cầu của người dùng. Ngoài ra, nhờ khả năng mở rộng theo chiều ngang (sharding), MongoDB cho phép chia nhỏ và phân phối dữ liệu trên nhiều máy chủ khác nhau. Khi lượng dữ liệu ngày càng tăng hoặc khi website phát triển, việc mở rộng hạ tầng trở nên dễ dàng và tiết kiệm chi phí hơn so với các giải pháp truyền thống vốn chỉ mở rộng theo chiều dọc (scale-up).

Sự linh hoạt trong việc lưu trữ dữ liệu phi cấu trúc và bán cấu trúc cũng là một điểm mạnh của MongoDB. Các website hiện nay thường lưu trữ nhiều loại dữ liệu khác nhau như văn bản, hình ảnh, video, bình luận, đánh giá… MongoDB cho phép lưu trữ tất cả những dữ liệu này mà không cần lo lắng về việc phải thiết kế và cập nhật lại cấu trúc bảng dữ liệu mỗi khi phát sinh trường thông tin mới. Điều này giúp các dự án web dễ dàng thích nghi với nhu cầu thực tế, đáp ứng tốt các thay đổi hoặc mở rộng đột xuất.

Bên cạnh đó, MongoDB hỗ trợ tích hợp tốt với rất nhiều ngôn ngữ lập trình và framework hiện đại như Node.js, Python, Java, PHP, Ruby, Go,... Nhà phát triển có thể dễ dàng kết nối, thao tác và xây dựng các tính năng cho website mà không gặp rào cản về công nghệ. Việc sử dụng MongoDB cùng với các framework phổ biến giúp rút ngắn thời gian phát triển dự án, tối ưu hóa nguồn lực và chi phí.

Khả năng truy vấn dữ liệu trong MongoDB cũng rất phong phú và mạnh mẽ. MongoDB cung cấp ngôn ngữ truy vấn riêng (MQL) cho phép thực hiện các thao tác tìm kiếm, lọc, sắp xếp, phân trang, xử lý dữ liệu dạng mảng hoặc tài liệu lồng nhau một cách trực tiếp và hiệu quả. Điều này giúp các website có thể thực hiện các truy vấn phức tạp, phục vụ các chức năng như tìm kiếm công thức, lọc sản phẩm, phân loại bài viết… mà không tốn nhiều công sức xây dựng logic ở tầng ứng dụng.

Ngoài ra, MongoDB còn nổi bật ở khả năng đảm bảo an toàn và sẵn sàng của dữ liệu. Tính năng replica set (cụm bản sao) giúp tự động sao lưu, đồng bộ hóa dữ liệu giữa các máy chủ, đảm bảo dữ liệu luôn sẵn sàng kể cả khi xảy ra sự cố phần cứng hoặc mất kết nối. Điều này giúp website duy trì hoạt động ổn định, tăng độ tin cậy và giảm nguy cơ mất dữ liệu.

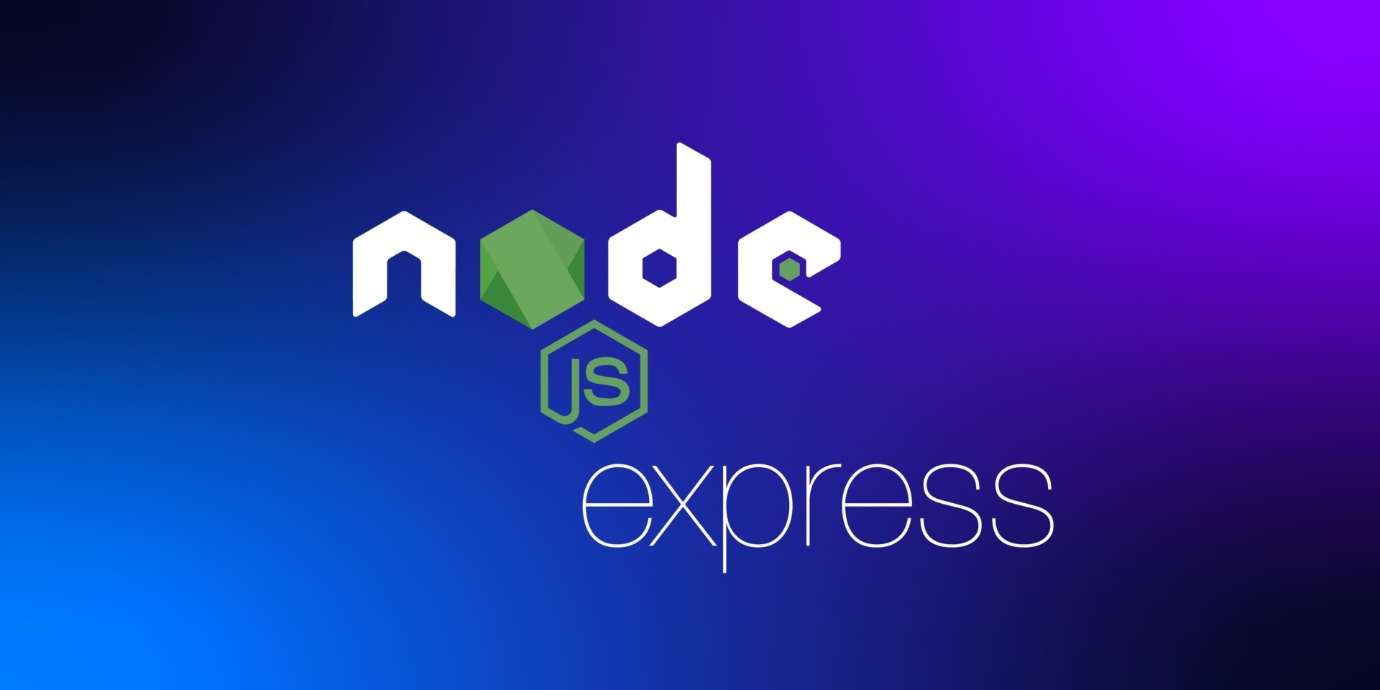
## 2.3. Các công nghệ hỗ trợ khác

Trong quá trình xây dựng website công thức nấu ăn, bên cạnh các công nghệ chính như ngôn ngữ lập trình, cơ sở dữ liệu và framework phát triển web, việc lựa chọn và tích hợp các công nghệ hỗ trợ khác đóng vai trò rất quan trọng nhằm tối ưu hiệu suất, tăng tính tương tác và nâng cao trải nghiệm người dùng. Những công nghệ này có thể bao gồm các thư viện giao diện, công cụ quản lý phiên làm việc, dịch vụ lưu trữ hình ảnh, tối ưu SEO, phân tích hành vi người dùng cũng như các công cụ hỗ trợ bảo mật và triển khai hệ thống. Việc áp dụng linh hoạt và phù hợp các công nghệ hỗ trợ sẽ giúp website vận hành ổn định, dễ mở rộng và đáp ứng tốt nhu cầu người dùng trong lĩnh vực ẩm thực ngày càng đa dạng.

### 2.3.1. Node.js, Express.js

Node.js: Là nền tảng chạy JavaScript phía máy chủ, Node.js cho phép xây dựng các ứng dụng web hiệu năng cao nhờ cơ chế xử lý bất đồng bộ (asynchronous) và mô hình hướng sự kiện (event-driven). Trong hệ thống website công thức nấu ăn, Node.js đóng vai trò là bộ khung chính của backend, chịu trách nhiệm tiếp nhận và xử lý các yêu cầu từ người dùng, giao tiếp với cơ sở dữ liệu, xử lý logic nghiệp vụ như quản lý công thức, người dùng, bình luận, đánh giá, lưu trữ hình ảnh minh họa,... Nhờ khả năng mở rộng tốt và sử dụng tài nguyên hiệu quả, Node.js giúp website đáp ứng lượng lớn truy cập đồng thời, đảm bảo tốc độ tải trang nhanh và trải nghiệm mượt mà.

NodeJS là một mã nguồn mở được sử dụng rộng bởi hàng ngàn lập trình viên trên toàn thế giới. NodeJS có thể chạy trên nhiều nền tảng hệ điều hành khác nhau từ WIndow cho tới Linux, OS X nên đó cũng là một lợi thế. NodeJS cung cấp các thư viện phong phú ở dạng Javascript Module khác nhau giúp đơn giản hóa việc lập trình và giảm thời gian ở mức thấp nhất.



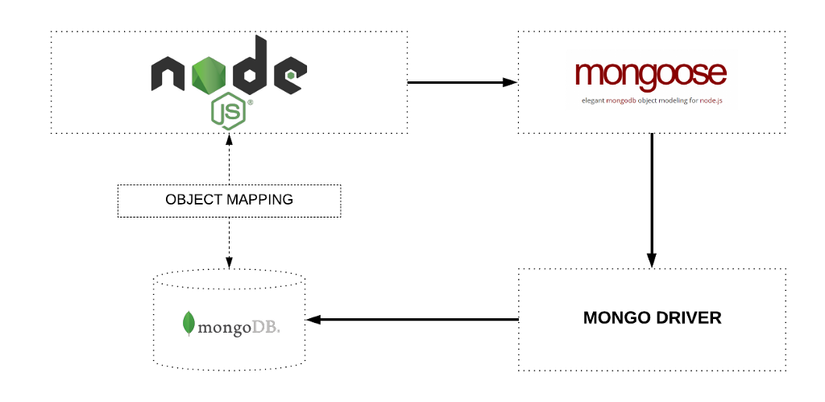
**Hình 2.3**. Nodejs, express

Express.js: Đây là framework web phổ biến nhất cho Node.js, cung cấp bộ công cụ mạnh mẽ để phát triển các API RESTful, quản lý định tuyến (routing), xử lý middleware, xác thực người dùng, kiểm soát lỗi và tích hợp dễ dàng với các module khác. Trong dự án website công thức nấu ăn, Express.js giúp tổ chức mã nguồn backend một cách rõ ràng, dễ bảo trì cũng như dễ dàng mở rộng thêm các chức năng mới như quản lý nguyên liệu, danh sách mua sắm, hoặc phân tích thống kê. Express.js còn hỗ trợ tích hợp các template engine như EJS để render trang HTML động, phục vụ tốt cho các tính năng hiển thị danh sách công thức, chi tiết món ăn, hoặc giao diện quản trị.

Việc kết hợp Node.js và Express.js giúp hệ thống website công thức nấu ăn đạt được các tiêu chuẩn về hiệu năng, khả năng mở rộng, dễ phát triển, bảo trì và tối ưu hóa trải nghiệm người dùng trên cả phía người dùng cuối lẫn quản trị viên.

### 2.3.2. Mongoose (ORM cho MongoDB)

Mongoose là một Object Data Modeling (ODM) library dành cho Node.js và MongoDB, giúp quản lý dữ liệu trong cơ sở dữ liệu NoSQL một cách có cấu trúc, rõ ràng và an toàn hơn. Trong hệ thống website công thức nấu ăn, Mongoose đóng vai trò như một “cầu nối” giữa ứng dụng Node.js/Express.js với cơ sở dữ liệu MongoDB. Thay vì thao tác trực tiếp với dữ liệu dạng JSON hoặc BSON không kiểm soát, Mongoose cho phép định nghĩa các schema (lược đồ dữ liệu) rõ ràng cho từng đối tượng như công thức nấu ăn, nguyên liệu, người dùng, bình luận, v.v.



**Hình 2.4.** Cấu trúc liên kết của mongoose với nodejs

Việc sử dụng Mongoose giúp đảm bảo tính nhất quán và toàn vẹn dữ liệu, hỗ trợ kiểm tra kiểu dữ liệu, ràng buộc các thuộc tính (bắt buộc, duy nhất, giá trị mặc định, validate...), đồng thời giúp thao tác truy vấn (tìm kiếm, lọc, phân trang...) trở nên đơn giản và tối ưu hơn. Lập trình viên có thể dễ dàng thêm mới, chỉnh sửa, xóa hoặc truy xuất dữ liệu từ MongoDB thông qua các phương thức do Mongoose cung cấp, thay vì phải viết lệnh truy vấn phức tạp.

Ngoài ra, Mongoose còn hỗ trợ các tính năng nâng cao như middleware (xử lý logic trước/sau khi lưu dữ liệu), population (liên kết nhiều bảng dữ liệu), virtuals (trường ảo), hooks (bắt sự kiện thao tác trên dữ liệu) và các plugin mở rộng khác. Điều này giúp cho hệ thống website công thức nấu ăn dễ dàng mở rộng, bảo trì, đồng thời tối ưu hiệu năng hoạt động khi số lượng dữ liệu và người dùng tăng lên.

Mongoose là giải pháp ORM/ODM lý tưởng để xây dựng các ứng dụng Node.js sử dụng cơ sở dữ liệu MongoDB, đặc biệt phù hợp với các hệ thống hiện đại như website công thức nấu ăn cần quản lý nhiều loại dữ liệu động và phức tạp.

### 2.3.3. Ejs, Dotenv

EJS là một template engine phổ biến cho Node.js và Express.js, cho phép nhúng trực tiếp JavaScript vào trong mã HTML để tạo ra các trang web động và linh hoạt. Trong dự án website công thức nấu ăn, EJS đóng vai trò là cầu nối giữa dữ liệu từ backend và giao diện người dùng, giúp trình bày danh sách công thức, hiển thị chi tiết món ăn, giao diện chức năng như đăng ký, đăng nhập, bình luận, đánh giá, v.v... với dữ liệu được truyền từ server.

EJS hỗ trợ các thao tác như lặp qua danh sách (loop), điều kiện (if-else), nhúng biến, định dạng dữ liệu, và cả việc chia nhỏ giao diện thành các thành phần tái sử dụng (partials) như header, footer, navigation,... Điều này giúp giao diện luôn được cập nhật theo thời gian thực, thân thiện với người dùng, đồng thời dễ dàng thay đổi, bảo trì và mở rộng khi có thêm yêu cầu mới.

Sử dụng EJS còn giúp tách biệt rõ ràng giữa phần xử lý logic (controller) và phần hiển thị (view), giúp mã nguồn trở nên sạch sẽ, dễ đọc và quản lý hơn. Ngoài ra, EJS còn tương thích tốt với các CSS framework và kỹ thuật responsive, góp phần tạo nên giao diện đẹp, hiện đại và tối ưu trên mọi thiết bị.

Dotenv là một thư viện không thể thiếu đối với các ứng dụng Node.js hiện đại, đặc biệt là những hệ thống có nhiều cấu hình nhạy cảm hoặc cần triển khai ở nhiều môi trường khác nhau (local, staging, production,...).

Dotenv cho phép lưu trữ toàn bộ biến môi trường (environment variables) trong một file .env riêng biệt, thay vì khai báo trực tiếp trong mã nguồn. Các biến này có thể bao gồm thông tin cấu hình kết nối cơ sở dữ liệu, secret key, port, đường dẫn lưu trữ ảnh, thông tin tích hợp dịch vụ bên ngoài,...

Khi khởi động ứng dụng, dotenv sẽ tự động nạp tất cả biến môi trường từ file .env vào biến môi trường toàn cục process.env của Node.js, giúp các thành phần khác của hệ thống dễ dàng truy cập và sử dụng mà không cần thay đổi mã nguồn.

Việc sử dụng dotenv không chỉ tăng cường bảo mật (giảm rủi ro lộ thông tin nhạy cảm khi public mã nguồn), mà còn giúp việc chuyển đổi cấu hình giữa các môi trường trở nên linh hoạt, thuận tiện khi triển khai và bảo trì.

## 2.4. Một số ứng dụng liên quan đến đề tài

Đề tài xây dựng website quản lý và chia sẻ công thức nấu ăn có nhiều ứng dụng thực tiễn và liên quan chặt chẽ tới nhu cầu của xã hội hiện đại. Một số ứng dụng bao gồm:

A screenshot of a website

AI-generated content may be incorrect.

**Hình 2.5.** Ứng dụng công thức nấu ăn Cookpad

Cookpad là một trong những cộng đồng chia sẻ công thức nấu ăn lớn nhất toàn cầu. Người dùng có thể đăng tải, tìm kiếm, lưu trữ công thức, bình luận, đánh giá món ăn và tương tác với các đầu bếp khác. Cookpad hỗ trợ đa ngôn ngữ, trong đó có tiếng Việt, và có hàng triệu công tA screenshot of a website

AI-generated content may be incorrect.hức được cập nhật liên tục.

**Hình 2.6**. Ứng dụng chia sẻ công thức nấu ăn Allrecipes

Allrecipes là nền tảng nổi tiếng của Mỹ, cho phép người dùng chia sẻ, lưu trữ công thức, đánh giá, bình luận, xây dựng bộ sưu tập cá nhân và gợi ý thực đơn phù hợp với từng dịp lễ, sự kiện. Hệ thống còn tích hợp tính năng lên danh sách mua sắm nguyên liệu và video hướng dẫn chi tiết.

A screenshot of a website

AI-generated content may be incorrect.

**Hình 2.7.** Ứng dụng chia sẻ công thức nấu ăn Tasty

Tasty là ứng dụng nổi bật của BuzzFeed, chuyên về các video hướng dẫn nấu ăn hấp dẫn, trực quan. Người dùng có thể tìm kiếm công thức theo nguyên liệu, chủ đề bữa ăn, mức độ khó; lưu công thức yêu thích và tương tác với cộng đồng.

# CHƯƠNG 3. PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG

Trong môi trường số hóa hiện đại, việc quản lý và chia sẻ công thức nấu ăn thông qua nền tảng web đóng vai trò quan trọng trong việc nâng cao trải nghiệm người dùng và lan tỏa giá trị ẩm thực. Việc tối ưu hóa quy trình tìm kiếm, lưu trữ và chia sẻ công thức không chỉ giúp người dùng dễ dàng tiếp cận với những món ăn đa dạng mà còn góp phần xây dựng một cộng đồng ẩm thực năng động, chuyên nghiệp và đáng tin cậy.

Chương này sẽ đi vào phân tích và thiết kế hệ thống quản lý web công thức nấu ăn, tập trung vào việc xây dựng một mô hình hoạt động hiệu quả, giao diện thân thiện với người dùng và có khả năng đáp ứng tốt nhu cầu của các đầu bếp tại gia, người yêu ẩm thực cũng như quản trị viên của hệ thống. Hệ thống sẽ bao gồm các chức năng như quản lý danh mục món ăn, đăng tải và chỉnh sửa công thức, đánh giá và bình luận của người dùng, tìm kiếm theo nguyên liệu hoặc loại món, lưu trữ công thức yêu thích, và gợi ý thực đơn theo khẩu vị hoặc dịp đặc biệt.

## 3.1. Phân tích yêu cầu và xác định các tác nhân

### 3.1.1. Phân tích yêu cầu hệ thống

Chúng em sử dụng mô hình SCRUM để phát triển hệ thống web công thức nấu ăn. Đầu tiên, nhóm phân tích yêu cầu và xây dựng cơ sở dữ liệu ban đầu. Sau đó, phát triển các chức năng quản lý công thức và danh mục món ăn. Tiếp theo, hoàn thiện các tính năng đăng công thức, đánh giá, tìm kiếm và lưu công thức yêu thích. Qua từng Sprint, nhóm liên tục cải tiến để hệ thống đáp ứng tốt nhu cầu người dùng.

|  |  |  |
| --- | --- | --- |
| **Sprint** | **Nội dung** | **Mục đích** |
| 1 | Quản lí nguyên liệu | Sprint này tập trung vào việc thu thập yêu cầu hệ thống, phân tích nghiệp vụ và thiết kế cơ sở dữ liệu liên quan đến nguyên liệu. Đồng thời, nhóm triển khai chức năng quản lý nguyên liệu, bao gồm thêm, sửa, xóa, tìm kiếm và phân loại nguyên liệu theo loại hoặc tình trạng (còn đủ, sắp hết, hết). |
| 2 | Quản lí công thức món ăn | Xây dựng chức năng thêm, sửa, xóa, xem chi tiết các loại công thức như món khai vị, món chính, món tráng miệng,… nhằm phục vụ việc phân loại và tổ chức danh mục công thức. Từ đó, người quản trị có thể dễ dàng quản lý kho công thức và hỗ trợ người dùng trong quá trình tìm kiếm, lựa chọn món ăn phù hợp. |
| 3 | Quản lí người dùng | Phát triển chức năng cho phép người dùng đăng ký, đăng nhập, cập nhật thông tin cá nhân và xem lịch sử công thức đã lưu hoặc đã đánh giá. Hệ thống kiểm tra tính hợp lệ của thông tin người dùng và lưu trữ dữ liệu để quản trị viên dễ dàng quản lý và hỗ trợ. |

### 3.1.2. Các tác nhân của hệ thống

Hệ thống quản lý công thức nấu ăn được thiết kế phục vụ cho 2 tác nhân chính: **Quản trị viên** và **Người dùng**.

**Người dùng** là những người truy cập và sử dụng nền tảng web để tìm kiếm, lưu trữ và chia sẻ các công thức nấu ăn. Họ có thể là cá nhân hoặc nhóm yêu thích ẩm thực, tìm kiếm công thức phù hợp với sở thích, nguyên liệu sẵn có hoặc dịp sử dụng. Người dùng sẽ đăng ký tài khoản, tìm kiếm công thức theo nhiều tiêu chí, lưu lại công thức yêu thích, đánh giá và bình luận các món ăn. Ngoài ra, họ có thể theo dõi lịch sử tương tác và nhận thông báo về các công thức mới, mẹo nấu ăn hoặc chương trình ưu đãi trên nền tảng.

**Quản trị viên** chịu trách nhiệm quản lý và vận hành hệ thống web công thức nấu ăn. Họ có quyền truy cập và quản lý toàn bộ dữ liệu trên hệ thống, bao gồm thông tin người dùng, công thức nấu ăn, nguyên liệu, đánh giá và các chương trình khuyến mãi. Quản trị viên cũng điều chỉnh các danh mục, duy trì chất lượng nội dung, xử lý phản hồi và giám sát hoạt động nhằm đảm bảo hệ thống vận hành an toàn, chính xác và mang lại trải nghiệm tốt nhất cho cộng đồng người dùng.

## 3.2. Sprint 1 Quản lí nguyên liệu

### 3.2.1. Giới thiệu Sprint 1

Sprint 1 tập trung vào chức năng **quản lý nguyên liệu** trong hệ thống web công thức nấu ăn. Mục tiêu là xây dựng và hoàn thiện các chức năng cơ bản như: thêm nguyên liệu, sửa thông tin nguyên liệu, xóa nguyên liệu, xem danh sách nguyên liệu, phân loại nguyên liệu theo loại, trạng thái tồn kho và nguồn gốc. Sprint này đặt nền móng cho việc quản lý nguyên liệu một cách hiệu quả và thuận tiện cho người dùng khi tạo và chỉnh sửa công thức.

A diagram of a person's life cycle

AI-generated content may be incorrect.

Hình 3.1. Biểu đồ ca sử dụng quản lý nguyên liệu

### 3.2.2. Thiết kế dữ liệu

Để thực hiện việc lưu trữ thông tin nguyên liệu như đã mô tả trong Chương 1, phần này sẽ trình bày chi tiết về việc thiết kế cơ sở dữ liệu nhằm đáp ứng các yêu cầu lưu trữ và quản lý nguyên liệu một cách hiệu quả. Bảng dữ liệu được xây dựng dựa trên việc phân tích các thông tin liên quan đến nguyên liệu sử dụng trong các công thức nấu ăn, đồng thời bổ sung thêm một số thuộc tính cần thiết nhằm phục vụ cho việc mở rộng phần mềm trong tương lai.

Các thông tin cụ thể về cấu trúc và nội dung của bảng dữ liệu lưu trữ nguyên liệu được mô tả chi tiết trong Bảng 3.1.

Mục đích: lưu trữ thông tin nguyên liệu

Tên bảng: Ingredients

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Diễn giải** | **Tên trường** | **Kiểu** | **Ràng buộc** |
| 1 | Mã nguyên liệu | \_id | ObjectId | PK |
| 2 | Tên nguyên liệu | name | String | Not null |
| 3 | Đơn vị | unit | String | Not null |
| 4 | Danh mục | category | String | Not null |

**Bảng 3.1.** Bảng lưu trữ thông tin nguyên liệu

### 3.2.3. Quy trình quản lý nguyên liệu

Quản lý công thức nấu ăn gồm các thao tác chính như thêm mới, chỉnh sửa, tìm kiếm, xóa và hiển thị thông tin công thức. Những chức năng này nhằm đảm bảo việc tổ chức và cập nhật dữ liệu về các món ăn một cách hiệu quả, hỗ trợ tốt cho quá trình tra cứu, học tập và chia sẻ công thức nấu ăn. Chi tiết các thao tác chức năng như sau:

* *Thêm nguyên liệu mới:*

|  |  |
| --- | --- |
| **Mục đích** | Thêm công thức nấu ăn mới vào hệ thống nhằm mở rộng và cập nhật cơ sở dữ liệu về các món ăn, phục vụ nhu cầu tra cứu và chia sẻ công thức một cách hiệu quả. |
| **Các bước thực hiện** | 1. Truy cập giao diện quản lý công thức: Đăng nhập vào hệ thống, truy cập trang quản lý và chọn "Thêm công thức mới". 2. Nhập thông tin công thức: Nhập các trường như tên món, nguyên liệu, cách chế biến, hình ảnh minh họa và mô tả chi tiết. 3. Lưu thông tin công thức: Kiểm tra lại thông tin, sau đó nhấn "Lưu" để hoàn tất. Hệ thống sẽ xác nhận và cập nhật danh sách công thức. |
| **Các bước bổ sung** | Ở bước 3, nếu các ô nhập liệu bắt buộc (ví dụ: tên món, nguyên liệu hoặc bước thực hiện) bị để trống, hệ thống sẽ hiển thị thông báo lỗi và yêu cầu người dùng nhập đầy đủ thông tin trước khi tiếp tục. |
| **Các ngoại lệ** | - Nếu công thức đã tồn tại trong hệ thống (trùng tên và nguyên liệu), dữ liệu sẽ không được lưu và hệ thống sẽ hiển thị thông báo cảnh báo.  - Người dùng có thể chọn hủy thao tác trước khi nhấn "Lưu", để quay lại hoặc nhập công thức khác. |

*- Chỉnh sửa thông tin nguyên liệu:*

|  |  |
| --- | --- |
| **Mục đích** | Cập nhật thông tin công thức nhằm đảm bảo dữ liệu món ăn luôn chính xác và phù hợp, nhất là khi có sự thay đổi về nguyên liệu, cách chế biến hoặc bổ sung mô tả chi tiết. |
| **Các bước thực hiện** | 1. Truy cập giao diện quản lý công thức: Đăng nhập vào hệ thống, chọn công thức cần chỉnh sửa từ danh sách hiện có.  2. Chỉnh sửa thông tin công thức: Cập nhật các thông tin cần thiết như tên món, nguyên liệu, bước thực hiện, hình ảnh, mô tả...  3. Lưu thông tin cập nhật: Kiểm tra lại và nhấn "Lưu" để xác nhận thay đổi. Hệ thống sẽ cập nhật nội dung công thức tương ứng. |
| **Các bước bổ sung** | - Nếu danh sách công thức có số lượng lớn, người quản trị có thể sử dụng thanh tìm kiếm theo tên món hoặc nguyên liệu để nhanh chóng tìm công thức cần chỉnh sửa.  - Ở bước 3, nếu bất kỳ trường bắt buộc nào bị để trống (ví dụ: tên món, nguyên liệu, cách nấu), hệ thống sẽ hiển thị thông báo lỗi và yêu cầu điền đầy đủ trước khi lưu. |
| **Các ngoại lệ** | Không |

*- Xóa nguyên liệu:*

|  |  |
| --- | --- |
| **Mục đích** | Loại bỏ những công thức không còn sử dụng hoặc bị trùng lặp nhằm đảm bảo cơ sở dữ liệu công thức luôn gọn gàng, chính xác và dễ quản lý. |
| **Các bước thực hiện** | 1. Truy cập giao diện quản lý công thức: Đăng nhập vào hệ thống, chọn công thức cần xóa từ danh sách hiển thị.  2. Xác nhận xóa công thức: Nhấn nút "Xóa" và xác nhận yêu cầu để đảm bảo công thức được loại bỏ khỏi hệ thống.  3. Hoàn tất quá trình xóa: Hệ thống sẽ xác nhận việc xóa và cập nhật lại danh sách công thức hiện có. |
| **Các bước bổ sung** | Không |
| **Các ngoại lệ** | - Nếu danh sách công thức quá nhiều, quản trị viên có thể sử dụng ô tìm kiếm theo tên món hoặc nguyên liệu để nhanh chóng tìm công thức cần xóa.  - Người dùng có thể hủy thao tác xóa trước khi xác nhận, để tránh xóa nhầm công thức quan trọng. |

### 3.2.4. Thiết kế quy trình nghiệp vụ

Quy trình quản lý công thức nấu ăn bắt đầu bằng việc hệ thống truy vấn cơ sở dữ liệu để hiển thị danh sách các công thức hiện có, kèm theo các thông tin chi tiết như tên món ăn, nguyên liệu chính, thời gian chế biến, hình ảnh minh họa và mô tả cách thực hiện. Quản trị viên có thể thực hiện các chức năng như thêm mới, chỉnh sửa hoặc xóa công thức thông qua giao diện quản lý.

Khi thực hiện chức năng thêm mới, hệ thống sẽ yêu cầu người dùng nhập các thông tin cơ bản của công thức như tên món, danh sách nguyên liệu, các bước chế biến, mô tả chi tiết và hình ảnh minh họa. Sau khi lưu dữ liệu vào cơ sở dữ liệu, hệ thống sẽ cập nhật và hiển thị danh sách công thức mới nhất trên giao diện quản lý.

Đối với chức năng chỉnh sửa, quản trị viên có thể thay đổi thông tin của các công thức đã có, ví dụ như cập nhật nguyên liệu, bổ sung bước chế biến, chỉnh sửa mô tả hoặc thay đổi hình ảnh. Các thay đổi sẽ được ghi nhận vào cơ sở dữ liệu, đảm bảo tính chính xác và thống nhất thông tin.

Khi quản trị viên chọn xóa một công thức, hệ thống sẽ hiển thị thông báo xác nhận để tránh thao tác nhầm. Sau khi xác nhận, hệ thống sẽ loại bỏ công thức khỏi cơ sở dữ liệu và cập nhật lại danh sách hiển thị trên hệ thống.

Toàn bộ quy trình này đảm bảo rằng các công thức nấu ăn trong hệ thống luôn được cập nhật đầy đủ, chính xác và dễ quản lý, từ đó hỗ trợ hiệu quả cho việc chia sẻ, tìm kiếm và tra cứu công thức trong cộng đồng người dùng.

A diagram of a diagram

AI-generated content may be incorrect.

**Hình 3.2.** Biểu đồ tuần tự quản lý nguyên liệu

### 3.2.5. Thiết kế giao diện quản lý nguyên liệu



**Hình 3.3.** Giao diện mẫu danh sách nguyên liệu

## 3.3. Sprint 2: Quản lí công thức món ăn

### 3.3.1. Giới thiệu Sprint 2

Trong Sprint 2, nhóm phát triển tập trung vào chức năng quản lý công thức món ăn, một thành phần quan trọng trong hệ thống web công thức nấu ăn. Việc quản lý và phân loại công thức giúp người dùng dễ dàng tìm kiếm và lựa chọn món ăn phù hợp với nhu cầu, khẩu vị và sở thích cá nhân.

Các loại công thức có thể bao gồm các nhóm như: món khai vị, món chính, món tráng miệng, món chay, món châu Á, món Âu,... Mỗi loại sẽ có những đặc điểm riêng về thành phần nguyên liệu, cách chế biến và thời gian nấu nướng. Việc quản lý loại công thức cho phép quản trị viên thêm mới, chỉnh sửa hoặc xóa thông tin các loại món ăn hiện có trong hệ thống, từ đó tạo điều kiện thuận lợi cho người dùng trong việc lọc và lựa chọn công thức.

Sprint này nhằm mục tiêu xây dựng giao diện trực quan và quy trình xử lý dữ liệu hiệu quả cho chức năng quản lý loại công thức, đồng thời hỗ trợ tối ưu cho các chức năng liên quan như tìm kiếm, lọc công thức và đề xuất món ăn phù hợp với từng đối tượng người dùng.

A diagram of a person with text

AI-generated content may be incorrect.

**Hình 3.4.** Biểu đồ ca sử dụng quản lý công thức món ăn

### 3.3.2. Thiết kế dữ liệu

Để phục vụ cho chức năng quản lý công thức món ăn, hệ thống cần thiết kế một cấu trúc dữ liệu hợp lý, đảm bảo lưu trữ đầy đủ các thông tin cần thiết cho mỗi loại công thức và hỗ trợ hiệu quả trong việc tra cứu, chỉnh sửa hoặc cập nhật dữ liệu. Việc thiết kế cơ sở dữ liệu loại công thức giúp chuẩn hóa thông tin, giảm thiểu trùng lặp và tăng hiệu quả quản lý.

Mục đích: Cho phép lưu trữ thông tin công thức món ăn

Tên bảng: Recipes

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Diễn giải** | **Tên trường** | **Kiểu** | **Ràng buộc** |
| 1 | Mã công thức | ingredient\_id | ObjectId | PK |
| 2 | Hướng dẫn | instructions | String | Not null |
| 3 | Thời gian nấu | cookingTimeMinutes | Number | Not null |
| 4 | Số lượng người ăn | servings | Number | Not null |
| 5 | Ảnh | imageUrls | Array | Null |
| 6 | Mã danh mục | category\_Id | String | Null |
| 7 | Thẻ | tags | Array | Not null |
| 8 | Lượt xem | views | Number | Null |
| 9 | Mã người dùng | userId | ObjectId | Not null |
| 10 | Ngày tạo | createdAt | Date | Not null |
| 11 | Ngày cập nhật | updatedAt | Date | Not null |

**Bảng 3.2.** Bảng lưu trữ thông tin công thức món ăn

### 3.3.3. Quy trình quản lý công thức nấu ăn

Quy trình quản lý công thức bắt đầu từ việc thêm mới loại công thức với các thông tin như tên loại, mô tả đặc trưng, thời gian chế biến trung bình và trạng thái hoạt động (ví dụ: đang sử dụng hoặc không còn áp dụng). Quản trị viên có thể chỉnh sửa thông tin khi có thay đổi hoặc xóa loại công thức khi không còn phù hợp hoặc không sử dụng nữa.

Hệ thống hỗ trợ tìm kiếm, lọc công thức theo các tiêu chí như tên loại, thời gian chế biến, hoặc đặc điểm món ăn (món chay, món ngọt, món mặn...), giúp quản lý dễ dàng và chính xác hơn.

Quy trình này đảm bảo dữ liệu công thức luôn được cập nhật đầy đủ, chính xác và phản ánh đúng thực tế hoạt động của hệ thống, từ đó góp phần nâng cao hiệu quả trong việc phân loại và đề xuất món ăn phù hợp cho người dùng.

* *Thêm công thức món ăn:*

|  |  |
| --- | --- |
| **Mục đích** | Người quản lý có thể dễ dàng thêm các công thức món ăn mới vào hệ thống để phù hợp với nhu cầu phân loại và quản lý công thức, giúp người dùng thuận tiện hơn trong việc tìm kiếm và lựa chọn món ăn. |
| **Các bước thực hiện** | 1. Truy cập vào hệ thống quản lý loại công thức và chọn tùy chọn "Thêm công thức mới" hoặc tương tự.  2. Nhập thông tin cho công thức mới như tên loại (ví dụ: món chính, món tráng miệng, món chay,...), mô tả đặc trưng, thời gian chế biến, trạng thái hoạt động và các thông tin liên quan khác.  3. Lưu thông tin và xác nhận việc thêm loại công thức mới vào hệ thống.  4. Cập nhật thông tin công thức mới trên hệ thống để hiển thị cho người dùng và quản trị viên khi cần thiết. |
| **Các bước bổ sung** | Ở bước 3, nếu dữ liệu trong các ô bắt buộc bị trống, hệ thống sẽ hiển thị thông báo lỗi và nhắc người dùng nhập đầy đủ thông tin. |
| **Các ngoại lệ** | - Nếu công thức đã tồn tại trong cơ sở dữ liệu, dữ liệu mới sẽ không được lưu và hệ thống sẽ hiển thị thông báo cảnh báo cho quản trị viên.  - Quản trị viên có thể hủy thao tác trước khi lưu để tránh nhập nhầm hoặc thay đổi ý định. |

* *Cập nhật công thức món ăn:*

|  |  |
| --- | --- |
| **Mục đích** | Người quản lý cập nhật thông tin của các công thức món ăn hiện có trong hệ thống để đảm bảo dữ liệu luôn được cập nhật đầy đủ và chính xác, giúp việc phân loại và tìm kiếm món ăn thuận tiện hơn. |
| **Các bước thực hiện** | 1. Truy cập vào hệ thống quản lý công thức và chọn công thức cần cập nhật từ danh sách các công thức hiện có.  2. Chỉnh sửa các thông tin cần thay đổi như tên, mô tả đặc trưng, thời gian chế biến, trạng thái hoạt động và các thông tin liên quan khác.  3. Lưu các thay đổi đã thực hiện.  4. Xác nhận việc cập nhật thông tin công thức trên hệ thống để đảm bảo tính chính xác trong quản lý và hiển thị cho người dùng. |
| **Các bước bổ sung** | Ở bước 3, nếu dữ liệu trong các trường bắt buộc bị bỏ trống, hệ thống sẽ hiển thị thông báo lỗi và yêu cầu nhập đầy đủ thông tin. |
| **Các ngoại lệ** | - Nếu công thức đã tồn tại với thông tin trùng lặp trong cơ sở dữ liệu, dữ liệu mới sẽ không được lưu và hệ thống sẽ thông báo cho quản trị viên.  - Quản trị viên có thể hủy thao tác trước khi lưu dữ liệu để tránh sai sót hoặc thay đổi ý định. |

* *Xoá công thức món ăn:*

|  |  |
| --- | --- |
| **Mục đích** | Người quản lý loại bỏ các công thức món ăn không cần thiết hoặc không còn áp dụng trong hệ thống, giúp giữ cho dữ liệu luôn gọn gàng và chính xác. |
| **Các bước thực hiện** | 1. Truy cập vào hệ thống quản lý công thức và chọn công thức cần xóa từ danh sách các loại hiện có.  2. Xác nhận rằng loại công thức đã chọn là cần xóa và kiểm tra xem có công thức nào đang sử dụng hay không. Nếu không có công thức nào liên quan, tiến hành xác nhận việc xóa.  3. Nếu có công thức đang sử dụng công thức này, cần cân nhắc chuyển các công thức đó sang loại khác trước khi tiến hành xóa.  4. Xác nhận việc xóa công thức và lưu lại các thay đổi trên hệ thống. |
| **Các bước bổ sung** | Không |
| **Các ngoại lệ** | Không |

### 3.3.4. Thiết kế quy trình nghiệp vụ

Quy trình nghiệp vụ quản lý công thức món ăn bắt đầu bằng việc hệ thống truy xuất và hiển thị danh sách các công thức hiện có, kèm theo các thông tin chi tiết như tên công thức, mô tả đặc trưng, thời gian chế biến, và trạng thái hoạt động. Quản trị viên có thể thực hiện các thao tác như thêm mới, chỉnh sửa hoặc xóa loại công thức.

Khi thêm mới, quản trị viên nhập các thông tin cần thiết vào biểu mẫu. Hệ thống sẽ kiểm tra tính hợp lệ của dữ liệu trước khi lưu vào cơ sở dữ liệu và cập nhật danh sách loại công thức. Với thao tác chỉnh sửa, quản trị viên có thể cập nhật bất kỳ thông tin nào và hệ thống sẽ ghi nhận thay đổi. Khi xóa, hệ thống yêu cầu xác nhận để đảm bảo an toàn, tránh xóa nhầm dữ liệu.

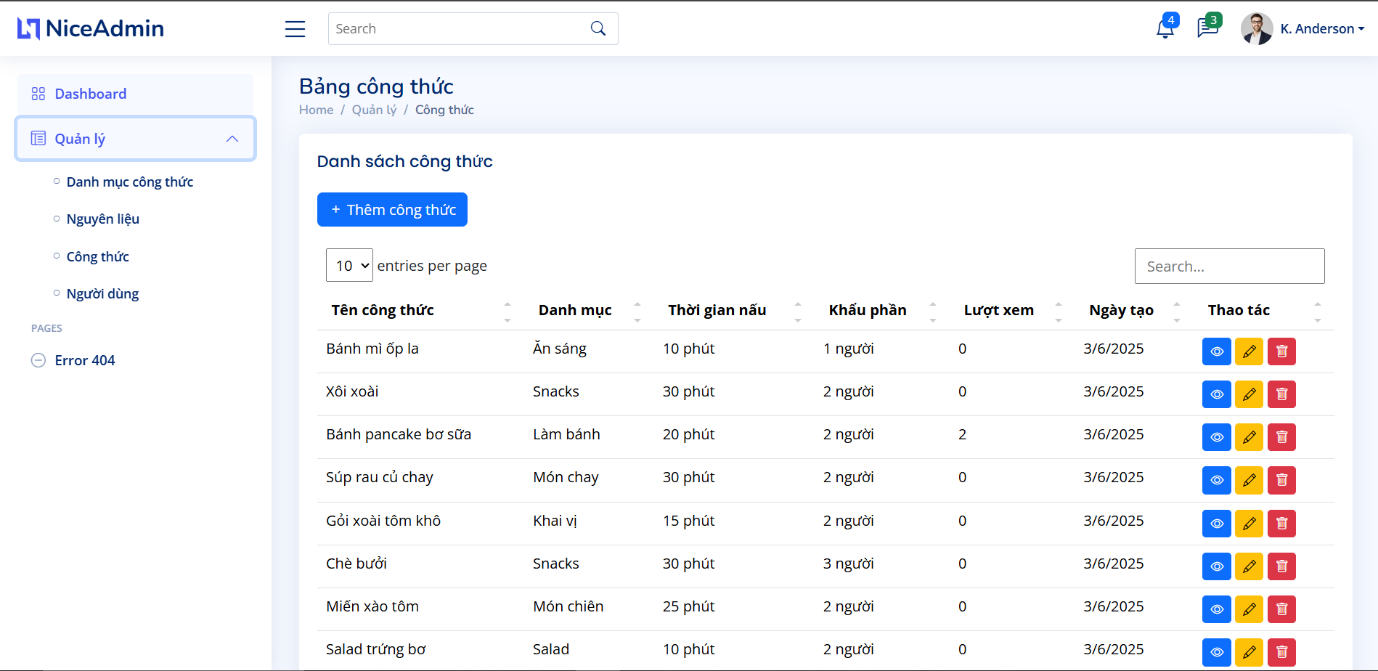
Toàn bộ quy trình đảm bảo rằng thông tin công thức món ăn luôn được cập nhật đầy đủ, chính xác và hỗ trợ hiệu quả cho các chức năng khác như tìm kiếm công thức, lọc món ăn và đề xuất món ăn phù hợp cho người dùng.

A diagram of a system

AI-generated content may be incorrect.

**Hình 3.5.** Biểu đồ tuần tự quản lý công thức món ăn

### 3.3.5. Thiết kế giao diện quản lý công thức món ăn



**Hình 3.6.** Giao diện mẫu danh sách công thức món ăn

## 3.4. Sprint 3: Quản lí người dùng

### 3.4.1. Giới thiệu Sprint 3

Sprint 3 tập trung phát triển chức năng quản lý người dùng – một trong những thành phần quan trọng của hệ thống web công thức nấu ăn. Mục tiêu chính của Sprint này là xây dựng và triển khai các tính năng giúp người dùng dễ dàng đăng ký, đăng nhập, quản lý tài khoản cá nhân, đồng thời đảm bảo tính bảo mật và trải nghiệm mượt mà.

Trong Sprint này, hệ thống sẽ hỗ trợ các thao tác như: tạo tài khoản mới, chỉnh sửa hoặc xóa thông tin cá nhân, quản lý quyền truy cập người dùng, cũng như theo dõi lịch sử hoạt động và tương tác với công thức nấu ăn. Các nghiệp vụ được thiết kế nhằm giúp người dùng quản lý tài khoản hiệu quả, đồng thời tạo điều kiện thuận lợi cho quản trị viên trong việc giám sát và quản lý cộng đồng người dùng trên nền tảng.

A diagram of a diagram

AI-generated content may be incorrect.

**Hình 3.7.** Biểu đồ ca sử dụng quản lý người dùng

### 3.4.2. Thiết kế dữ liệu

Để phục vụ cho chức năng quản lý người dùng, hệ thống cần một cấu trúc dữ liệu rõ ràng và chặt chẽ để lưu trữ thông tin liên quan đến tài khoản người dùng. Dữ liệu người dùng sẽ bao gồm các thông tin cơ bản như: mã người dùng, tên đăng nhập, mật khẩu, họ tên, email, số điện thoại, vai trò (quản trị viên, người dùng thường), trạng thái tài khoản, và các thông tin cá nhân khác.

Việc thiết kế bảng dữ liệu người dùng nhằm đảm bảo việc truy xuất, cập nhật và kiểm soát thông tin người dùng một cách hiệu quả. Bảng dữ liệu được xây dựng để đảm bảo tính bảo mật và toàn vẹn thông tin, hỗ trợ tốt cho các chức năng xác thực và phân quyền trong hệ thống.

Cấu trúc dữ liệu sẽ hỗ trợ các thao tác như: tạo mới tài khoản, chỉnh sửa thông tin cá nhân, thay đổi mật khẩu, xóa tài khoản, và theo dõi lịch sử hoạt động người dùng. Điều này giúp hệ thống duy trì quản lý người dùng hiệu quả, đồng thời đảm bảo trải nghiệm và bảo mật cho người dùng.

Mục đích: Lưu trữ các thông tin tài khoản người dùng

Tên bảng: Users

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Diễn giải** | **Tên trường** | **Kiểu** | **Ràng buộc** |
| 1 | Mã người dùng | id | ObjectId | Pk |
| 2 | Tên | username | String | Not null |
| 3 | Email | email | String | Not null |
| 4 | Mật khẩu | password | String | Null |
| 5 | Ảnh | avatarUrl | Array | Null |
| 6 | Thông tin | bio | String | Null |
| 7 | Ngày tham gia | joinedAt | Date | Not null |

**Bảng 3.3.** Bảng lưu trữ thông tin người dùng

### 3.4.3. Quy trình quản lí người dùng

Quy trình quản lý người dùng trong hệ thống web công thức nấu ăn bắt đầu bằng việc người dùng đăng ký tài khoản mới. Người dùng nhập các thông tin như tên đăng nhập, mật khẩu, họ tên, email và số điện thoại. Hệ thống kiểm tra tính hợp lệ và trùng lặp trước khi lưu thông tin vào cơ sở dữ liệu.

Sau khi đăng ký, người dùng có thể đăng nhập để sử dụng các chức năng của hệ thống. Quản trị viên có quyền xem danh sách người dùng, chỉnh sửa thông tin cá nhân hoặc phân quyền truy cập (quản trị viên, người dùng thường). Nếu cần thiết, quản trị viên cũng có thể xóa tài khoản không còn hoạt động hoặc vi phạm quy định.

Ngoài ra, hệ thống hỗ trợ tính năng tìm kiếm, lọc người dùng theo tên, email hoặc vai trò để quản lý dễ dàng và nhanh chóng. Quy trình này đảm bảo thông tin người dùng được cập nhật chính xác, bảo mật và phục vụ hiệu quả cho cả trải nghiệm người dùng lẫn hoạt động quản trị hệ thống.

* *Tạo tài khoản người dùng mới:*

|  |  |
| --- | --- |
| **Mục đích** | Người quản lý thực hiện tạo tài khoản người dùng mới. |
| **Các bước thực hiện** | 1. Truy cập vào hệ thống chọn chức năng quản lý người dùng.  2. Chọn chức năng tạo tài khoản.  3. Thực hiện điền thông tin: Email. Mật khẩu, Họ và tên, Số điện thoại, Địa chỉ, Giới tính, Ảnh đại diện, Vai trò. |
| **Các bước bổ sung** | Ở bước 3, nếu dữ liệu trong các ô văn bản trống, hệ thống sẽ hiển thị thông báo lỗi và nhắc nhập dữ liệu vào các ô văn bản trống. |
| **Các ngoại lệ** | - Nếu tài khoản đã tồn tại trong cơ sở dữ liệu, dữ liệu đã nhập sẽ không được lưu vào cơ sở dữ liệu và sẽ hiển thị thông báo cho quản trị viên.  - Quản trị viên có thể hủy thao tác trước khi lưu dữ liệu. |

- *Chỉnh sửa thông tin tài khoản:*

|  |  |
| --- | --- |
| **Mục đích** | Người quản lý thực hiện sửa thông tin người dùng. |
| **Các bước thực hiện** | 1. Truy cập vào hệ thống chọn chức năng quản lý người dùng.  2. Chọn chức năng chỉnh sửa thông tin.  3. Thực hiện chỉnh sửa thông tin: Email. Mật khẩu, Họ và tên, Số điện thoại, Địa chỉ, Giới tính, Ảnh đại diện, Vai trò. |
| **Các bước bổ sung** | - Nếu có quá nhiều tài khoản trong danh sách, quản trị viên có thể tìm kiếm người dùng theo tên người dùng, số điện thoại hoặc email.  - Ở bước 3, nếu dữ liệu tại các ô trống, hệ thống sẽ hiển thị thông báo lỗi và nhắc nhập dữ liệu vào các ô trống. |
| **Các ngoại lệ** | - Nếu người dùng đã tồn tại trong cơ sở dữ liệu, dữ liệu đã chỉnh sửa sẽ không được lưu vào cơ sở dữ liệu và sẽ hiển thị thông báo cho quản trị viên.  - Quản trị viên có thể hủy thao tác trước khi lưu dữ liệu. |

*- Xóa tài khoản:*

|  |  |
| --- | --- |
| **Mục đích** | Người quản lý thực hiện tạo tài khoản người dùng mới. |
| **Các bước thực hiện** | 1. Truy cập vào hệ thống chọn chức năng quản lý người dùng.  2. Chọn chức năng xoá tài khoản. |
| **Các bước bổ sung** | Không |
| **Các ngoại lệ** | Không |

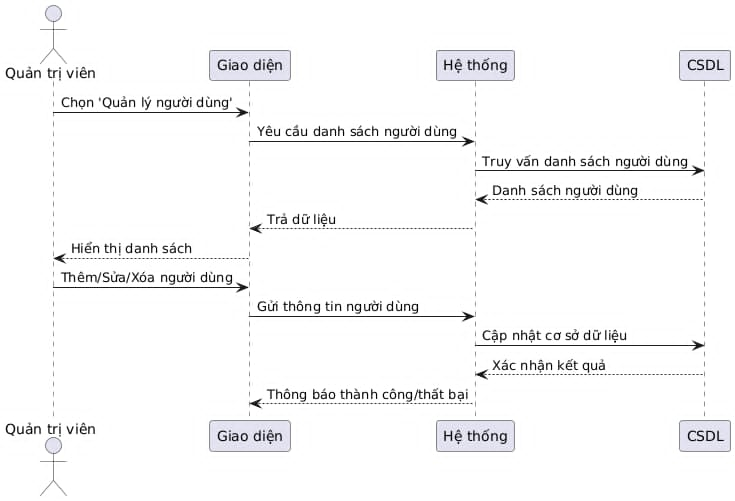
### 3.4.4. Thiết kế quy trình nghiệp vụ

Quy trình nghiệp vụ quản lý người dùng bắt đầu khi quản trị viên truy cập vào chức năng quản lý người dùng trên hệ thống web công thức nấu ăn. Hệ thống sẽ truy vấn cơ sở dữ liệu và hiển thị danh sách người dùng hiện có kèm theo các thông tin như: họ tên, email, số điện thoại, vai trò (người dùng, quản trị viên) và trạng thái hoạt động của tài khoản.

Từ giao diện quản lý, quản trị viên có thể thực hiện các thao tác như: thêm mới, chỉnh sửa, khóa/mở khóa, hoặc xóa tài khoản người dùng. Khi thêm mới, hệ thống yêu cầu nhập đầy đủ thông tin: họ tên, email, mật khẩu, số điện thoại, địa chỉ, giới tính, ảnh đại diện và vai trò. Sau khi kiểm tra tính hợp lệ và trùng lặp tài khoản, thông tin sẽ được lưu vào hệ thống và cập nhật vào danh sách người dùng.

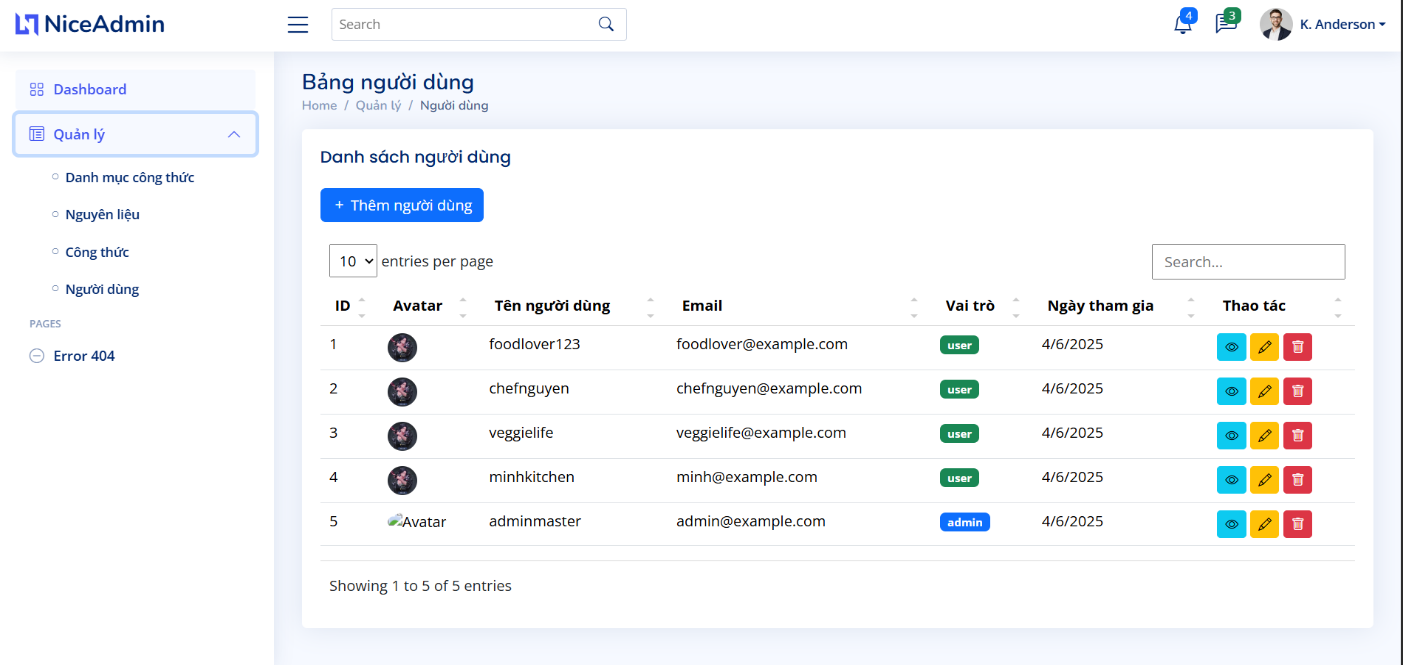
Với chức năng chỉnh sửa, quản trị viên có thể cập nhật lại thông tin cá nhân, ảnh đại diện hoặc vai trò của người dùng. Trong trường hợp tài khoản có hành vi vi phạm, quản trị viên có thể thực hiện khóa tài khoản tạm thời để hạn chế truy cập vào hệ thống. Nếu cần thiết, quản trị viên có thể xóa hoàn toàn tài khoản khỏi hệ thống để đảm bảo an toàn cho nền tảng.

Toàn bộ quy trình được thiết kế nhằm đảm bảo tính linh hoạt, bảo mật và kiểm soát chặt chẽ hoạt động của người dùng, góp phần xây dựng một cộng đồng nấu ăn lành mạnh và hiệu quả trên nền tảng.



**Hình 3.8.** Biểu đồ tuần tự quản lý người dùng

### 2.4.5. Thiết kế giao diện quản lý người dùng



**Hình 3.9.** Giao diện mẫu danh sách người dùng

# CHƯƠNG 4. XÂY DỰNG HỆ THỐNG

Dựa trên cơ sở phân tích và thiết kế hệ thống đã được trình bày ở Chương 3, chương này sẽ tập trung vào quá trình hiện thực hóa hệ thống thông qua việc xây dựng website công thức nấu ăn. Nội dung bao gồm việc cài đặt môi trường phát triển, lựa chọn các công nghệ phù hợp và triển khai các chức năng chính của hệ thống. Các chức năng như quản lý tài khoản người dùng, đăng tải và chỉnh sửa công thức, phân loại món ăn, tìm kiếm theo nguyên liệu hoặc tên món, đánh giá và bình luận, cùng với các tiện ích hỗ trợ người dùng khác sẽ được xây dựng và kiểm thử một cách chi tiết. Mục tiêu của chương là đảm bảo hệ thống được phát triển với đầy đủ các tính năng đã phân tích, có tính ổn định, bảo mật và thân thiện với người dùng.

## 4.1. Môi trường phát triển

### 4.1.1. Công nghệ và công cụ

Để xây dựng và triển khai hệ thống website công thức nấu ăn, nhóm đã sử dụng các công nghệ và công cụ hiện đại, đảm bảo hiệu quả phát triển, khả năng mở rộng, bảo trì dễ dàng và thân thiện với lập trình viên. Cụ thể như sau:

* **Ngôn ngữ lập trình JavaScript (Node.js):** Node.js là môi trường chạy JavaScript phía server, cho phép xây dựng các ứng dụng web hiệu năng cao và dễ mở rộng. Việc sử dụng Node.js giúp đồng bộ ngôn ngữ lập trình giữa client và server, đồng thời tận dụng khả năng xử lý bất đồng bộ mạnh mẽ.
* **Express.js:** Express là framework nhẹ và linh hoạt của Node.js, giúp đơn giản hóa quá trình xây dựng server, định tuyến các yêu cầu HTTP và quản lý middleware. Express được sử dụng để triển khai các chức năng như đăng ký tài khoản, đăng tải công thức, tìm kiếm món ăn,...
* **MongoDB:** MongoDB là hệ quản trị cơ sở dữ liệu NoSQL phổ biến, cho phép lưu trữ dữ liệu dạng tài liệu (document) với cấu trúc linh hoạt. Hệ thống sử dụng MongoDB để lưu trữ thông tin người dùng, công thức nấu ăn, bình luận, đánh giá và các nội dung liên quan một cách hiệu quả và dễ mở rộng.
* **Visual Studio Code (VS Code):** VS Code là môi trường phát triển tích hợp nhẹ, hỗ trợ tốt cho JavaScript/Node.js. Công cụ này cung cấp nhiều tiện ích như tự động hoàn thành, kiểm tra lỗi, tích hợp Git và các tiện ích mở rộng phục vụ cho lập trình web hiện đại.
* **Asana**: Asana là công cụ quản lý dự án trực tuyến, hỗ trợ theo dõi tiến độ và phân công nhiệm vụ trong nhóm. Nhóm đã sử dụng Asana để tổ chức công việc theo phương pháp Agile, phân chia thành các Sprint rõ ràng, giúp quá trình phát triển hệ thống trở nên minh bạch, hiệu quả và có khả năng điều phối nhóm tốt hơn.

### 4.1.2. Lựa chọn giao diện cho bài toán

Ở giao diện dành cho người dùng, cụ thể là những khách truy cập hoặc người dùng có phân quyền “người dùng thông thường”, chúng tôi sử dụng mẫu giao diện miễn phí dành cho các website chia sẻ công thức nấu ăn, được cung cấp bởi Themewagon.com, có tên là **“Delicious – Free Bootstrap 4 HTML5 Recipe Website Template”**. Giao diện này có thiết kế hiện đại, bố cục rõ ràng, dễ sử dụng và hỗ trợ tốt trên các thiết bị di động. Việc sử dụng mẫu giao diện này giúp nhóm tiết kiệm thời gian thiết kế giao diện từ đầu, đồng thời mang lại trải nghiệm thân thiện và trực quan cho người dùng khi thực hiện các thao tác như xem công thức, tìm kiếm món ăn, lọc theo danh mục, hoặc tương tác với nội dung qua đánh giá và bình luận.

A close up of a burger

AI-generated content may be incorrect.

**Hình 4.1.** Giao diện mẫu dùng cho khách hàng

Đối với giao diện dành cho quản trị viên – những tài khoản có phân quyền “Admin”, chúng tôi sử dụng mẫu giao diện quản trị **“NiceAdmin – Bootstrap Admin HTML Template”** được cung cấp bởi **BootstrapMade.com**. Giao diện này hỗ trợ tốt cho việc quản lý thông tin hệ thống như tài khoản người dùng, công thức, nguyên liệu,... với bố cục chuyên nghiệp, có khả năng hiển thị biểu đồ, bảng dữ liệu và điều hướng rõ ràng. Việc sử dụng mẫu giao diện này giúp quản trị viên thao tác hiệu quả, đồng thời mang lại sự linh hoạt trong quản lý nhờ các tính năng sẵn có như quản lý biểu mẫu, thông báo, và hỗ trợ responsive trên các thiết bị khác nhau.

A screenshot of a computer

AI-generated content may be incorrect.

**Hình 4.2.** Giao diện trang chủ admin

## 4.2. Quản lý dự án trên Asana và Github

Trong quá trình phát triển hệ thống website công thức nấu ăn, nhóm chúng tôi sử dụng **Asana** để quản lý dự án và **GitHub** để quản lý mã nguồn, phối hợp làm việc nhóm một cách hiệu quả và có tổ chức.

**Asana** là công cụ quản lý công việc trực quan, hỗ trợ lập kế hoạch và theo dõi tiến độ dự án thông qua các bảng phân công công việc, dòng thời gian, danh sách công việc và lịch trình. Mỗi Sprint được tạo thành một dự án riêng trong Asana, trong đó các nhiệm vụ như phân tích yêu cầu, thiết kế cơ sở dữ liệu, lập trình chức năng, kiểm thử, và viết tài liệu được phân chia rõ ràng và gán cho từng thành viên phụ trách. Nhờ đó, cả nhóm có thể theo dõi trạng thái từng đầu việc, đảm bảo tiến độ và hiệu suất làm việc.

Bên cạnh đó, nhóm sử dụng **GitHub** để quản lý mã nguồn và phối hợp làm việc theo mô hình phân nhánh (branch). Mỗi thành viên thực hiện lập trình trên nhánh riêng, sau đó gửi pull request để hợp nhất mã nguồn vào nhánh chính (main/master) sau khi đã được xem xét và kiểm thử. Việc sử dụng GitHub không chỉ giúp kiểm soát phiên bản mã nguồn mà còn hỗ trợ cộng tác, phát hiện lỗi sớm và đảm bảo tính ổn định của hệ thống trong quá trình phát triển.

Sự kết hợp giữa Asana và GitHub giúp nhóm đảm bảo quy trình phát triển phần mềm diễn ra tuần tự, có tổ chức, dễ kiểm soát và đạt hiệu quả cao.

A screenshot of a computer

AI-generated content may be incorrect.

**Hình 4.3.** Công cụ quản lý dự án Asana

A screenshot of a computer

AI-generated content may be incorrect.

**Hình 4.4.** Công cụ quản lý mã nguồn Github

## 4.3. Xây dựng chức năng quản lý nguyên liệu

### 4.3.1. Xây dựng sản phẩm

Phần này trình bày quá trình hiện thực hóa chức năng quản lý nguyên liệu bao gồm việc xây dựng giao diện, xử lý logic bằng NodeJs Express, tương tác với cơ sở dữ liệu MongoDB và đảm bảo hoạt động ổn định theo đúng yêu cầu đã phân tích ở Chương 2.

const Ingredient = require('../models/Ingredient');

exports.getAllIngredients = async (req, res) => {

  try {

    const ingredients = await Ingredient.find().sort({ \_id: 1 });

    res.render('admin/ingredient-table', {

      ingredients,

      title: 'Quản lý nguyên liệu'

    });

  } catch (error) {

    console.error('Error fetching ingredients:', error);

    res.status(500).send('Server Error');

  }

};

### 4.3.2. Giao diện quản lý nguyên liệu

Giao diện chức năng quản lý phòng được thiết kế dựa trên template đã chọn, đảm bảo tính trực quan, dễ sử dụng và thân thiện với người dùng. Người dùng có thể thực hiện các thao tác cơ bản như thêm, sửa, xóa và tìm kiếm thông tin.

A screenshot of a computer

AI-generated content may be incorrect.

**Hình 4.5.** Giao diện quản lý nguyên liệu

### 4.3.3. Kiểm thử chức năng quản lý nguyên liệu

Bảng dưới đây mô tả quá trình kiểm thử chức năng quản lý nguyên liệu trong hệ thống, bao gồm các thao tác thêm, sửa, xóa và tìm kiếm nguyên liệu. Quá trình kiểm thử được thực hiện dựa trên giao diện quản trị của website.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên Test Case** | **Phương thức** | **Dữ liệu kiểm thử** | **Kết quả mong đợi** |
| 1 | Thêm nguyên liệu hợp lệ | POST | Tên: Trứng gà, Đơn vị: Quả, Số lượng tồn: 20 | |  | | --- | | Nguyên liệu được thêm thành công |  |  | | --- | |  | |
| 2 | |  | | --- | | Thêm nguyên liệu thiếu tên |  |  | | --- | |  | | POST | |  | | --- | | Tên: (rỗng), Đơn vị: Gram, Số lượng tồn: 50 |  |  | | --- | |  | | |  | | --- | | Hiển thị lỗi yêu cầu nhập tên nguyên liệu |  |  | | --- | |  | |
| 3 | |  | | --- | | Sửa thông tin nguyên liệu | | PUT | Chỉnh sửa đơn vị từ “Quả” → “Chiếc” cho nguyên liệu Trứng gà | Cập nhật thông tin nguyên liệu thành công |
| 4 | |  | | --- | | Xóa nguyên liệu |  |  | | --- | |  | | DELETE | |  | | --- | | Xóa nguyên liệu: Muối trắng |  |  | | --- | |  | | |  | | --- | | Nguyên liệu bị xóa khỏi hệ thống |  |  | | --- | |  | |
| 5 | |  | | --- | | Tìm kiếm nguyên liệu theo tên | | GET | |  | | --- | | Từ khóa tìm kiếm: “Đường” |  |  | | --- | |  | | Hiển thị danh sách nguyên liệu có chứa từ “Đường” |

**Bảng 4.1.** Kiểm thử các chức năng quản lý nguyên liệu

## 4.4. Xây dựng chức năng quản lý công thức

### 4.4.1. Xây dựng sản phẩm

Phần này trình bày quá trình hiện thực hóa chức năng quản lý công thức bao gồm việc xây dựng giao diện, xử lý logic bằng NodeJs Express, tương tác với cơ sở dữ liệu MongoDB và đảm bảo hoạt động ổn định theo đúng yêu cầu đã phân tích ở Chương 2.

Dưới đây là một đoạn mã liên quan đến chức năng quản lý công thức mới từ hệ thống

const Recipe = require('../models/Recipe');

const Category = require('../models/Category');

// Get all recipes

exports.getAllRecipes = async (req, res) => {

  try {

    const recipes = await Recipe.find().sort({ createdAt: -1 });

    const categories = await Category.find();

    const categoryMap = {};

    categories.forEach(cat => {

      categoryMap[cat.id] = cat.name;

    });

    res.render('admin/recipe-table', {

      recipes,

      categoryMap,

      categories, // Pass categories for select dropdown

      title: 'Quản lý công thức'

    });

  } catch (error) {

    console.error('Error fetching recipes:', error);

    res.status(500).send('Server Error');

  }

};

### 4.4.2. Giao diện quản lý công thức

Giao diện chức năng quản lý công thức được thiết kế dựa trên template đã chọn, đảm bảo tính trực quan, dễ sử dụng và thân thiện với người dùng. Người dùng có thể thực hiện các thao tác cơ bản như thêm, sửa, xóa và tìm kiếm thông tin.

Hệ thống đảm bảo tính toàn vẹn dữ liệu bằng cách kiểm tra thông tin trước khi lưu vào cơ sở dữ liệu và kiểm tra quyền hạn của người dùng đăng nhập để đảm bảo chỉ quản trị viên được phép mới có thể truy cập và thực hiện các thao tác quản lý.

A screenshot of a computer

AI-generated content may be incorrect.

**Hình 4.6.** Giao diện quản lý công thức

### 4.4.3. Kiểm thử chức năng quản lý công thức

Bảng dưới đây mô tả quá trình kiểm thử chức năng quản lý công thức nấu ăn trong hệ thống. Bao gồm các thao tác như thêm công thức, kiểm tra trùng tên, chỉnh sửa nội dung, xóa công thức đang dùng và tìm kiếm..

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên Test Case** | **Phương thức** | **Dữ liệu kiểm thử** | **Kết quả mong đợi** |
| 1 | |  |  |  | | --- | --- | --- | | |  | | --- | | Thêm công thức hợp lệ |  |  | | --- | |  | | | POST | |  | | --- | | Tên: "Bún bò Huế", Nguyên liệu: [Bò, Sả, Ớt], Hướng dẫn: "Luộc bò, nêm nếm, hoàn thành" |  |  | | --- | |  | | |  |  |  | | --- | --- | --- | | |  | | --- | | Công thức được thêm thành công |  |  | | --- | |  | | |
| 2 | |  |  |  | | --- | --- | --- | | |  | | --- | | Thêm công thức trùng tên |  |  | | --- | |  | |  |  | | --- | |  | | POST | |  |  |  | | --- | --- | --- | | |  | | --- | | Tên: "Bún bò Huế" (đã tồn tại), Nguyên liệu: ..., Hướng dẫn: ... |  |  | | --- | |  | |  |  | | --- | |  | | |  |  |  | | --- | --- | --- | | |  | | --- | | Hiển thị thông báo lỗi: "Tên công thức đã tồn tại" |  |  | | --- | |  | |  |  | | --- | |  | |
| 3 | |  | | --- | | Thêm công thức thiếu hướng dẫn |  |  | | --- | |  | | |  | | --- | | POST |  |  | | --- | |  | | Tên: "Canh chua", Nguyên liệu: [Cá, Cà chua], Hướng dẫn: (rỗng) | Hiển thị lỗi yêu cầu nhập hướng dẫn nấu |
| 4 | |  | | --- | | Sửa công thức |  |  | | --- | |  | | PUT | Cập nhật tên công thức "Canh chua" → "Canh chua cá lóc" | |  | | --- | | Thông tin công thức được cập nhật |  |  | | --- | |  | |
| 5 | Xóa công thức đang được đánh dấu yêu thích | |  | | --- | | DELETE |  |  | | --- | |  | | |  | | --- | | Xóa công thức "Cơm chiên trứng" đang nằm trong danh sách yêu thích của người dùng | | |  | | --- | | Hiển thị thông báo: "Không thể xóa công thức đang được sử dụng" |  |  | | --- | |  | |

**Bảng 4.2.** Kiểm thử các chức năng quản lý công thức

## 4.5. Xây dựng chức năng quản lý người dùng

### 4.5.1. Xây dựng sản phẩm

Phần này trình bày quá trình hiện thực hóa chức năng quản lý người dùng bao gồm việc xây dựng giao diện, xử lý logic bằng NodeJs Express, tương tác với cơ sở dữ liệu MongoDB và đảm bảo hoạt động ổn định theo đúng yêu cầu đã phân tích ở Chương 2..

const User = require('../models/User');

exports.getAllUsers = async (req, res) => {

  try {

    const users = await User.find().sort({ id: 1 });

    res.render('admin/user-table', {

      users,

      title: 'Quản lý người dùng'

    });

  } catch (error) {

    console.error('Error fetching users:', error);

    res.status(500).send('Server Error');

  }

};

exports.getUserById = async (req, res) => {

  try {

    const user = await User.findOne({ id: req.params.id });

    if (!user) {

      return res.status(404).json({ message: 'Không tìm thấy người dùng' });

    }

    res.json(user);

  } catch (error) {

    res.status(500).json({ message: 'Lỗi server' });

  }

};

### 4.5.2. Giao diện quản lý người dùng

Giao diện chức năng quản lý người dùng được thiết kế dựa trên template đã chọn, đảm bảo tính trực quan, dễ sử dụng và thân thiện với người dùng. Người dùng có thể thực hiện các thao tác cơ bản như thêm, sửa, xóa và tìm kiếm thông tin.

A screenshot of a computer

AI-generated content may be incorrect.

Hình 4.7. Giao diện quản lý người dùng

### 4.5.3. Kiểm thử chức năng quản lý người dùng

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên Test Case** | **Phương thức** | **Dữ liệu kiểm thử** | **Kết quả mong đợi** |
| 1 | Thêm người dùng mới | POST | Email: newuser@example.com, Mật khẩu: 123456 | Người dùng được tạo thành công |
| 2 | Thêm người dùng trùng email | POST | Email: newuser@example.com (đã tồn tại) | Hiển thị lỗi: email đã tồn tại trong hệ thống |
| 3 | Chỉnh sửa người dùng | PUT | Cập nhật số điện thoại thành 0123456789 | Thông tin người dùng được cập nhật |
| 4 | Xóa người dùng | DELETE | Chọn người dùng có email: olduser@example.com | Người dùng bị xóa khỏi hệ thống |
| 5 | Tìm kiếm người dùng | GET | Tìm theo từ khóa “admin” | Hiển thị danh sách người dùng phù hợp với tìm kiếm |

**Bảng 4.3.** Kiểm thử các chức năng quản lý người dùng

# KẾT LUẬN

## 1. Kết quả đạt được

Sau một thời gian tập trung triển khai đồ án, em đã hoàn thành được Website chia sẻ công thức nấu ăn – một nền tảng giúp người dùng dễ dàng tìm kiếm, thêm mới, chỉnh sửa và lưu trữ các công thức nấu ăn một cách khoa học và tiện lợi. Dù trong quá trình thực hiện còn gặp nhiều khó khăn và hạn chế về kiến thức cũng như kinh nghiệm thực tế, nhưng nhờ sự cố gắng tìm hiểu tài liệu trên internet, học hỏi thêm từ các nguồn khác nhau, em đã xây dựng được một website có giao diện thân thiện, dễ sử dụng và phù hợp với mục đích của người dùng.

Qua việc thực hiện đề tài, em đã tích lũy thêm nhiều kỹ năng thực tiễn trong lập trình web, đặc biệt là phát triển ứng dụng sử dụng **Node.js**, **Express**, **MongoDB** cùng với các công nghệ frontend như **HTML**, **CSS**, **JavaScript**. Bên cạnh đó, em cũng hiểu sâu hơn về quy trình xây dựng hệ thống web hoàn chỉnh từ thiết kế cơ sở dữ liệu, tạo API đến xây dựng giao diện người dùng và kiểm thử chức năng.

Nhờ đó, kỹ năng lập trình và tư duy phát triển phần mềm của em đã được cải thiện đáng kể. Đây sẽ là nền tảng hữu ích giúp em phát triển hơn trong các dự án thực tế cũng như sẵn sàng làm việc tại các doanh nghiệp công nghệ trong tương lai.

## 2. Hạn chế

Mặc dù đề tài đã đạt được những kết quả nhất định, nhưng trong quá trình thực hiện, vẫn còn tồn tại một số hạn chế cần được cải thiện để hoàn thiện sản phẩm hơn:

- Giao diện còn đơn giản: Giao diện hiện tại còn khá cơ bản, chưa mang tính thẩm mỹ cao do hạn chế về kỹ năng thiết kế UI/UX.

- Chưa có phân quyền người dùng: Hệ thống hiện tại chưa hỗ trợ phân quyền giữa người quản trị và người dùng thường.

- Thiếu tính năng tương tác cộng đồng: Website chưa có chức năng đánh giá, bình luận hoặc chia sẻ công thức giữa người dùng với nhau.

- Thiếu hỗ trợ đa nền tảng: Giao diện chưa được tối ưu tốt cho thiết bị di động và máy tính bảng.

- Khả năng mở rộng còn hạn chế: Website chưa được tối ưu để xử lý dữ liệu với số lượng người dùng và công thức lớn.

- Tính năng bảo mật cơ bản: Các biện pháp bảo mật đang ở mức cơ bản, chưa được kiểm thử chuyên sâu để phòng chống các cuộc tấn công phổ biến như XSS, CSRF.

- Thiếutối ưu SEO**:** Website hiện chưa được tối ưu cho các công cụ tìm kiếm, ảnh hưởng đến khả năng tiếp cận người dùng qua Google hoặc các nền tảng tìm kiếm khác.

## 3. Hướng phát triển

Sau khi hoàn thành đề tài, em mong muốn tiếp tục phát triển hệ thống theo các hướng sau:

- Cải thiện giao diện người dùng: Thiết kế giao diện đẹp mắt, hiện đại hơn bằng công cụ chuyên nghiệp như Figma, tích hợp các thành phần UI/UX nhất quán và thân thiện.

- Tăng tính tương tác: Bổ sung tính năng đánh giá công thức, bình luận, chia sẻ qua mạng xã hội để tăng mức độ tương tác người dùng.

- Phân quyền người dùng: Xây dựng hệ thống tài khoản với vai trò như người dùng, người tạo công thức và quản trị viên.

- Hệ thống tìm kiếm nâng cao: Cải tiến tính năng tìm kiếm với bộ lọc theo nguyên liệu, thời gian nấu, mức độ khó,...

- Đề xuất thông minh: Áp dụng thuật toán gợi ý công thức theo sở thích hoặc lịch sử truy cập của người dùng.

- Tối ưu bảo mật: Tăng cường các biện pháp bảo mật như mã hóa thông tin nhạy cảm, xác thực người dùng đa lớp, chống tấn công XSS/SQLi.

- Triển khai thực tế: Đưa website lên nền tảng đám mây (như Vercel, Heroku hoặc VPS riêng) để thử nghiệm với người dùng thật.

Với những định hướng này, em kỳ vọng dự án sẽ trở thành một hệ thống hoàn chỉnh hơn, hữu ích hơn cho cộng đồng những người yêu thích nấu ăn, đồng thời góp phần nâng cao kỹ năng và kinh nghiệm của bản thân trong ngành công nghệ phần mềm.

# BẢNG PHÂN CHIA CÔNG VIỆC

A screenshot of a computer

AI-generated content may be incorrect.

Dũng:

- Ý tưởng: 3 tuần

- Thiết kế CSDL: 6 tuần

- Lập trình: 8 tuần

- Kiểm tra & sửa lỗi: 4 tuần

- Tổng: ~ 40%

Hưng:

- Ý tưởng: 3 tuần

- Thiết kế CSDL: 6 tuần

- Nghiên cứu: 4 tuần

- Lập trình: 8 tuần

-Tổng: ~ 30%

Oai:

- Ý tưởng: 3 tuần

- Nghiên cứu: 4 tuần

- Lập trình: 8 tuần

- Hoàn thành: 2 tuần

-Tổng: ~ 30%

# TÀI LIỆU THAM KHẢO

[1] Lê Văn Phùng, Kỹ nghệ phần mềm, NXB Thông tin và Truyền thông, 2014.

[2] Lê Văn Phùng, Lê Hương Giang, Kỹ nghệ phần mềm nâng cao, NXB Thông tin và

truyền thông, 2015.

[3] Đoàn Văn Ban, Nguyễn Thị Tĩnh, Giáo trình phân tích thiết kế hệ thống hướng đối

tượng bằng UML, NXB Đại học sư phạm, 2011.

[4] Ian Sommerville, Software Engineering, Ninth Edition, Addison-Wesley, 2011.

[5] Scott Tilley, Harry J. Rosenblatt, Systems Analys and Design, Shelly Cashman

Series, 11th Edition, 2016.

[6] Matt Stauffer, Laravel: From Apprentice to Artisan, 2nd Edition, O'Reilly Media,

2021.