

TRƯỜNG KỸ THUẬT VÀ CÔNG NGHỆ
KHOA CÔNG NGHỆ THÔNG TIN



THỰC TẬP ĐỒ ÁN CHUYÊN NGÀNH
HỌC KỲ I, NĂM HỌC 2025-2026

Xây dựng website chia sẻ công thức nấu ăn

Giảng viên hướng dẫn:
ThS. Hà Thị Thúy Vi

Sinh viên thực hiện:
Họ tên: Nguyễn Thanh Duy
MSSV: 110122062
Lớp: DA22TTD

Vĩnh Long, tháng 12 năm 2025

TRƯỜNG KỸ THUẬT VÀ CÔNG NGHỆ
KHOA CÔNG NGHỆ THÔNG TIN



THỰC TẬP ĐỒ ÁN CHUYÊN NGÀNH
HỌC KỲ I, NĂM HỌC 2025-2026

Xây dựng website chia sẻ công thức nấu ăn

Giảng viên hướng dẫn:
ThS. Hà Thị Thúy Vi

Sinh viên thực hiện:
Họ tên: Nguyễn Thanh Duy
MSSV: 110122062
Lớp: DA22TTD

Vĩnh Long, tháng 12 năm 2025

NHẬN XÉT CỦA GIẢNG VIÊN HƯỚNG DẪN

[illegible]

Vĩnh Long, ngày tháng năm

Giảng viên hướng dẫn

(Ký tên và ghi rõ họ tên)

NHẬN XÉT CỦA THÀNH VIÊN HỘI ĐỒNG

[illegible]

Vĩnh Long, ngày tháng năm

Thành viên hội đồng
(Ký tên và ghi rõ họ tên)

LỜI CẢM ƠN

Trước khi đi sâu vào dự án, em xin gửi lời cảm ơn chân thành đến trường Đại học Trà Vinh đã tạo điều kiện cho em thực hiện dự án này, những cá nhân đã hỗ trợ và giúp đỡ em một cách tận tình. Cũng như là sự hỗ trợ và giúp đỡ của Cô Hà Thị Thúy Vi, người đã đóng vai trò quan trọng trong việc phát triển và hoàn thành dự án này.

Em rất cảm kích vì sự giúp đỡ của những người quanh em, những người đã dành thời gian, công sức và kiến thức của họ để giúp đỡ em. Các ý kiến đóng góp và sự hợp tác của mọi người là nguồn động lực giúp em phát triển bản thân.

Em rất quý trọng những người đã hỗ trợ và giúp đỡ em trong suốt thời gian qua, đặc biệt là sự giúp đỡ của Cô Hà Thị Thúy Vi, nhờ có sự giúp đỡ của Cô mà em mới có thể thực hiện và hoàn thành dự án. Một lần nữa, xin chân thành cảm ơn trường Đại học Trà Vinh và Cô Hà Thị Thúy Vi đã giúp đỡ và em mong rằng sẽ nhận được sự ủng hộ của mọi người trong những dự án sắp tới.

Trân trọng!

MỤC LỤC

CHƯƠNG 1: TỔNG QUAN	1
1.1 Bối cảnh	1
1.2 Vấn đề đặt ra	1
1.3 Mục tiêu của hệ thống.....	1
1.4 Phạm vi nghiên cứu	1
1.5 Phương pháp và hướng tiếp cận.....	2
1.6 Kết quả mong đợi.....	2
CHƯƠNG 2: NGHIÊN CỨU LÝ THUYẾT.....	3
2.1 Kiến trúc và nguyên lý web hiện đại	3
2.2 Bảo mật và xác thực.....	3
2.2.1 JSON Web Token (JWT).....	3
2.2.2 Bcrypt và bảo vệ mật khẩu	4
2.2.3 CORS (Cross-Origin Resource Sharing)	4
2.3 Công nghệ và kỹ thuật triển khai	4
2.3.1 Frontend: React, React Router, Axios, CSS Responsive.....	4
2.3.2 Backend: Node.js (Non-blocking I/O), Express Middleware.....	5
2.3.3 Dữ liệu và lưu trữ.....	5
2.4 Cấu trúc dữ liệu và thuật toán cho nested comments.....	5
2.4.1 Nested Comments (Bình luận lồng nhau).....	5
2.4.2 Like System (Hệ thống thích).....	5
2.4.3 Follow System (Hệ thống theo dõi).....	6
2.4.4 Report Quota System (Hệ thống quota báo cáo)	6
2.4.5 Violation Auto-Block System (Hệ thống tự động khóa khi vi phạm).....	6
2.4.6 Broadcast Notification System (Hệ thống thông báo hàng loạt)	6
CHƯƠNG 3: HIỆN THỰC HÓA NGHIÊN CỨU.....	7
3.1 Đặc tả nhu cầu.....	7
3.2 Thiết kế hệ thống	7
3.2.1 Kiến trúc tổng thể	7
3.2.2 Mô hình dữ liệu.....	8
3.2.3 Thiết kế API (REST)	14
3.2.4 Thiết kế giao diện (UI/UX).....	14
3.3 Hiện thực hóa (Implementation).....	15
3.3.1 Frontend (React)	15
3.3.2 Backend (Node.js + Express)	16

3.3.3 Database (MySQL)	17
3.4 Bảo mật và quản lý phiên	18
3.5 Thiết kế chi tiết nested comments và like system.....	19
3.5.1 Nested Comments	19
3.5.2 Like System	19
3.5.3 Follow System	19
3.5.4 Report System.....	19
3.5.5 Notification System	20
3.5.6 Theme System.....	20
3.6 Kiểm thử	20
CHƯƠNG 4: KẾT QUẢ NGHIÊN CỨU.....	21
4.1 Kết quả chức năng	21
4.2 Trải nghiệm người dùng (UX/UI).....	21
4.3 Hiệu năng và kỹ thuật	22
4.4 Bảo mật	22
4.5 Giao diện minh họa (mô tả nhanh)	22
4.6 Đánh giá nhanh	48
4.7 Hướng phát triển	48
CHƯƠNG 5: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN	49
5.1 Kết luận.....	49
5.2 Hạn chế	49
5.3 Hướng phát triển	49
DANH MỤC TÀI LIỆU THAM KHẢO.....	50
PHỤ LỤC	51

DANH MỤC HÌNH ẢNH

Hình 3.1 Mô hình dữ liệu ERD.....	8
Hình 4.1: Giao diện Đăng nhập (Login)	22
Hình 4.2: Giao diện Đăng ký (Register)	23
Hình 4.3: Giao diện Quên mật khẩu nhập Email(Forgot Password)	23
Hình 4.3.1: Giao diện Quên mật khẩu nhập OTP (Forgot Password)	24
Hình 4.3.2: Giao diện Quên mật khẩu nhập mật khẩu mới (Forgot Password).....	24
Hình 4.4.1: Thanh điều hướng khi chưa đăng nhập (Navbar)	25
Hình 4.4.2: Thanh điều hướng Khi đã đăng nhập của User (Navbar)	25
Hình 4.4.3: Thanh điều hướng Khi đã đăng nhập của admin/moderator (Navbar)	25
Hình 4.4.4: Thanh điều hướng khi đăng nhập trên mobile (Navbar).....	25
Hình 4.5: Giao diện Trang chủ (Home)	26
Hình 4.5.1: Giao diện Trang chủ Phân trang (Home).....	26
Hình 4.5.2: Giao diện Trang chủ trên mobile (Home).....	26
Hình 4.5.3: Giao diện Trang chủ phần danh sách công thức trên mobile (Home)	27
Hình 4.6: Giao diện Tìm kiếm (Search)	28
Hình 4.6.1: Giao diện Tìm kiếm gợi ý công thức(Search)	28
Hình 4.6.2: Giao diện Tìm kiếm khi không có công thức phù hợp (Search).....	28
Hình 4.7: Giao diện Tạo công thức (Create Recipe)	29
Hình 4.7.1: Giao diện Tạo công thức Khu vực nguyên liệu(Create Recipe).....	29
Hình 4.7.2: Giao diện Tạo công thức Khu vực các bước nấu(Create Recipe)	29
Hình 4.8: Giao diện Chỉnh sửa công thức (Edit Recipe)	30
Hình 4.8.1: Giao diện Chỉnh sửa công thức: Nút (Edit Recipe)	30
Hình 4.9: Giao diện Chi tiết công thức (Recipe Detail)	31
Hình 4.9.1: Giao diện Chi tiết công thức khi là tác giả (Recipe Detail).....	31
Hình 4.9.2: Giao diện Chi tiết công thức khi không là tác giả (Recipe Detail).....	32
Hình 4.9.3: Giao diện Chi tiết công thức Các bước nấu (Recipe Detail)	32
Hình 4.9.4: Giao diện Chi tiết công thức khu vực bình luận (Recipe Detail)	32
Hình 4.10: Giao diện Công thức của tôi (My Recipes)	33
Hình 4.11: Giao diện Công thức yêu thích (Favorite Recipes).....	34
Hình 4.12: Giao diện Hồ sơ cá nhân (Profile)	35
Hình 4.13: Giao diện Hồ sơ người dùng khác (User Profile)	35
Hình 4.14: Giao diện Danh sách Followers	36
Hình 4.14.1: Giao diện Danh sách Following	36
Hình 4.15: Giao diện Cài đặt tài khoản (Account Settings)	37

Hình 4.15.1: Giao diện Cài đặt tài khoản Section thông tin cá nhân (Account Settings)....	37
Hình 4.16: Giao diện Thông báo của User (Notifications).....	38
Hình 4.17: Giao diện Tùy chỉnh Theme (Theme Customization)	39
Hình 4.18: Giao diện Theme Marketplace.....	39
Hình 4.19: Modal Báo cáo bài viết.....	40
Hình 4.20: Modal Báo cáo bình luận.....	41
Hình 4.21: Modal Báo cáo người dùng	41
Hình 4.22: Giao diện Quản trịQuản Lý Công Thức (Admin Dashboard)	42
Hình 4.22.1: Giao diện Quản trịQuản Lý Người Dùng(Admin Dashboard)	42
Hình 4.23: Giao diện Quản lý báo cáo (Admin Reports)	43
Hình 4.24: Giao diện Gửi thông báo cá nhân (Send Notification)	44
Hình 4.24.1: Giao diện Gửi thông báo hàng loạt (Send Notification).....	44
Hình 4.25: Giao diện Ẩn/Hiện bài viết	45
Hình 4.25.1: Giao diện Ẩn/Hiện bài viết.....	45
Hình 4.26: Giao diện Quy định cộng đồng (Rules)	46
Hình 4.27: Footer.....	46
Hình 4.28: Giao diện Responsive trên Mobile	47

DANH MỤC BẢNG BIỂU

Bảng 2.1: Mã trạng thái HTTP	3
Bảng 2.2: So sánh phương pháp xác thực.....	4
Bảng 2.3: Công nghệ sử dụng.....	4
Bảng 3.1: Schema cơ sở dữ liệu	8
Bảng 3.2: Phân quyền user vs admin.....	18
Bảng PLA.1: Môi trường phát triển.....	51
Bảng PLC.1: Bảng nguoi_dung.....	54
Bảng PLC.2: Bảng cong_thuc	54
Bảng PLG.1: Test cases chính	60

TÓM TẮT ĐỒ ÁN CHUYÊN NGÀNH

VẤN ĐỀ NGHIÊN CỨU

Trong thời đại số hóa, nhu cầu chia sẻ và trao đổi kiến thức về nấu ăn ngày càng tăng cao. Tuy nhiên, các nền tảng hiện tại thường thiếu tính tích hợp hoặc khó sử dụng. Đồ án này nhằm xây dựng một website chia sẻ công thức nấu ăn (CookShare) với đầy đủ chức năng:

- Cho phép người dùng tạo, chỉnh sửa, xóa công thức riêng của mình
- Hỗ trợ tương tác xã hội (bình luận, đánh giá, yêu thích, theo dõi người dùng)
- Cung cấp tìm kiếm và phân loại công thức hiệu quả
- Triển khai hệ thống quản lý admin/moderator để kiểm duyệt nội dung
- Hệ thống báo cáo vi phạm đa dạng (bài viết, bình luận, người dùng)
- Tùy chỉnh giao diện cá nhân và chia sẻ theme
- Tối ưu hóa giao diện đáp ứng (Responsive Design) trên mọi thiết bị

CÁC HƯỚNG TIẾP CẬN

1. Kiến trúc 3 tầng (Three-Layer Architecture):

- Tầng Giao diện (Presentation Layer): React 19.x với Component-based architecture
- Tầng Ứng dụng (Application Layer): Node.js + Express 4.x cung cấp REST API
- Tầng Dữ liệu (Data Layer): MySQL 8.0+ lưu trữ dữ liệu quan hệ

2. Mô hình MVC (Model-View-Controller):

- Model: Cơ sở dữ liệu MySQL (nguai_dung, cong_thuc, binh_luan, danh_gia, favorite, follow, bao_cao, notifications, ...)
- View: React components (Home, CreateRecipe, RecipeDetail, AdminDashboard, ThemeCustomization, v.v.)
- Controller: Express routes (auth.js, recipe.js, rating.js, favorite.js, admin.js, follow.js, report.js, notification.js, theme.js)

3. Bảo mật & Xác thực:

- Dùng Bcrypt (với Salt & Adaptive hashing) để mã hóa mật khẩu
- Dùng JSON Web Token (JWT) để quản lý phiên người dùng (token hết hạn sau 7 ngày)
- Cấu hình CORS để kiểm soát truy cập giữa frontend và backend
- Middleware xác thực để kiểm tra quyền (user/moderator/admin)

4. Tối ưu hóa & Hiệu suất:

- Non-blocking I/O: Node.js xử lý nhiều request đồng thời mà không chặn
- Cloudinary: Lưu trữ ảnh trên đám mây, giảm tải server
- CSS Responsive: Giao diện tự động thích ứng với mọi kích thước màn hình
- Scheduled Tasks: Tác vụ tự động chạy định kỳ (reset quota, xóa bài vi phạm)

KẾT QUẢ ĐẠT ĐƯỢC

1. Chức năng Xác thực & Quản lý Người dùng:

- Đăng ký tài khoản với xác thực email, mật khẩu
- Đăng nhập với JWT token (hết hạn 7 ngày)
- Hồ sơ người dùng với avatar, bio
- Quên mật khẩu (gửi OTP qua email)
- Phân biệt vai trò: user / moderator / admin

2. Chức năng Quản lý Công Thức:

- Tạo công thức (tiêu đề, nguyên liệu, bước nấu, ảnh, khẩu phần, thời gian nấu)
- Ảnh cho từng bước nấu (step images)
- Danh sách công thức với phân trang
- Xem chi tiết công thức với đếm lượt xem
- Chỉnh sửa công thức (chỉ chủ sở hữu)
- Xóa công thức (chủ sở hữu hoặc admin/moderator)
- Tìm kiếm công thức theo từ khóa

3. Chức năng Tương tác & Bình luận:

- Bình luận trên công thức (tạo, xem, xóa)
- Bình luận lồng nhau (nested comments) trả lời bình luận
- Thích bình luận (like system)
- Đánh giá sao (1-5 sao, mỗi user chỉ được đánh giá 1 lần)
- Yêu thích công thức (lưu vào danh sách yêu thích)

4. Chức năng Theo dõi (Follow System):

- Theo dõi/hủy theo dõi người dùng khác
- Xem danh sách followers/following
- Đếm số lượng followers/following
- Xem profile người dùng khác

5. Chức năng Báo cáo Vi phạm:

- Báo cáo bài viết vi phạm
- Báo cáo bình luận vi phạm
- Báo cáo người dùng vi phạm
- Hệ thống quota báo cáo (giới hạn số lần báo cáo)
- Upload ảnh bằng chứng khi báo cáo
- Tự động khóa tính năng khi vi phạm nhiều lần
- Gửi email thông báo khi xử lý báo cáo

6. Chức năng Thông báo:

- Thông báo cá nhân từ admin/moderator
- Thông báo hàng loạt (broadcast) đến tất cả người dùng
- Trả lời thông báo
- Đánh dấu đã đọc
- Đếm thông báo chưa đọc

7. Chức năng Tùy chỉnh Giao diện:

- Thay đổi màu chủ đạo
- Thay đổi hình nền
- Lưu và export theme
- Chia sẻ theme lên marketplace(cửa hàng chủ đề)
- Tải theme từ marketplace

8. Chức năng Quản lý Admin/Moderator:

- Xem danh sách tất cả người dùng
- Thay đổi vai trò người dùng (user ↔ moderator ↔ admin)
- Xóa người dùng hoặc công thức vi phạm
- Ẩn bài viết thủ công
- Xử lý báo cáo (chấp nhận/bác bỏ)
- Gửi thông báo đến người dùng
- Gửi thông báo hàng loạt

ĐÁNH GIÁ & HẠN CHẾ

Điểm mạnh:

- Kiến trúc module hóa, dễ mở rộng
- Bảo mật tốt (Bcrypt, JWT, CORS, Middleware)
- Giao diện responsive, thân thiện với người dùng
- Hệ thống báo cáo và kiểm duyệt hoàn chỉnh
- Tính năng xã hội phong phú (follow, like, comment)
- Tùy chỉnh giao diện cá nhân

Hạn chế & hướng phát triển trong tương lai:

- Chưa triển khai Redis để cache dữ liệu
- Chưa có hệ thống notification real-time (WebSocket)
- Chưa triển khai unit test toàn diện
- Chưa có CI/CD pipeline
- Có thể thêm Machine Learning để gợi ý công thức

MỞ ĐẦU

1. Lý do chọn đề tài

Xu hướng số hóa trong lĩnh vực ẩm thực ngày càng mạnh, nhu cầu chia sẻ và tìm kiếm công thức nấu ăn trực tuyến rất lớn. Các nền tảng hiện có thường phân tán, thiếu quản lý chất lượng nội dung hoặc chưa tối ưu trải nghiệm người dùng (UX/UI) trên đa thiết bị. Mong muốn xây dựng một hệ thống tập trung, thân thiện, cho phép người dùng dễ dàng đăng tải, tìm kiếm, tương tác và quản trị nội dung công thức nấu ăn.

2. Mục đích nghiên cứu

- Xây dựng website CookShare đáp ứng các chức năng cốt lõi: đăng ký/đăng nhập, tạo-chỉnh sửa-xóa công thức, tìm kiếm, đánh giá, bình luận, yêu thích, theo dõi người dùng
- Đảm bảo bảo mật dữ liệu người dùng (mã hóa mật khẩu, xác thực bằng JWT, kiểm soát quyền truy cập)
- Thiết kế giao diện đáp ứng (responsive) cho cả desktop và mobile, tối ưu tốc độ và trải nghiệm
- Xây dựng hệ thống báo cáo vi phạm và kiểm duyệt nội dung hoàn chỉnh
- Đề xuất kiến trúc mở rộng, dễ bảo trì

3. Đối tượng nghiên cứu

- **Người dùng cuối:** Cá nhân đam mê nấu ăn, muốn chia sẻ hoặc tìm kiếm công thức
- **Kiểm duyệt viên (moderator):** Hỗ trợ admin kiểm duyệt nội dung, xử lý báo cáo
- **Quản trị viên (admin):** Quản trị hệ thống, quản lý người dùng, phân quyền
- **Công nghệ và giải pháp:** React (UI), React Router, Axios, CSS responsive; Node.js/Express (API), JWT, Bcrypt; MySQL; Cloudinary; Nodemailer

4. Phạm vi nghiên cứu

Chức năng người dùng:

- Đăng ký, đăng nhập, quản lý profile (avatar, bio)
- Quản lý công thức (tạo/sửa/xóa) với ảnh từng bước
- Bình luận nested, đánh giá, yêu thích
- Theo dõi người dùng khác
- Báo cáo vi phạm (bài viết/bình luận/người dùng)
- Tùy chỉnh giao diện cá nhân
- Nhận thông báo

Chức năng kiểm duyệt viên (moderator):

- Xử lý báo cáo vi phạm
- Ẩn bài viết/xóa bình luận vi phạm
- Gửi thông báo đến người dùng

Chức năng quản trị viên (admin):

- Toàn bộ quyền của moderator
- Quản lý người dùng, đổi vai trò
- Xóa tài khoản

Hạ tầng:

- Frontend: React (localhost:3000)
- Backend: Node/Express (localhost:3001)
- Database: MySQL
- Lưu trữ ảnh: Cloudinary
- Gửi email: Nodemailer

Ngoài phạm vi:

- Realtime notifications (WebSocket)
- Gợi ý cá nhân hóa bằng ML
- Cache/Redis
- CI/CD tự động

CHƯƠNG 1: TỔNG QUAN

1.1 Bối cảnh

- Xu hướng tìm kiếm và chia sẻ công thức nấu ăn trực tuyến tăng mạnh cùng với sự phổ biến của mạng xã hội và thiết bị di động
- Người dùng cần một nền tảng tập trung, dễ dùng, tin cậy để:
 - + Đăng tải, lưu trữ, chỉnh sửa công thức cá nhân
 - + Tìm kiếm, học hỏi từ cộng đồng
 - + Tương tác (bình luận, đánh giá, yêu thích, theo dõi)
 - + Được kiểm duyệt nội dung và báo cáo vi phạm

1.2 Vấn đề đặt ra

- Nội dung ẩm thực hiện có phân tán, chất lượng không đồng đều, thiếu kiểm duyệt
- Trải nghiệm người dùng chưa tối ưu trên nhiều thiết bị
- Bảo mật và quyền riêng tư chưa luôn được đảm bảo
- Thiếu cơ chế quản trị để xử lý nội dung vi phạm
- Thiếu tính năng tương tác xã hội (theo dõi, thích bình luận)
- Thiếu khả năng tùy chỉnh giao diện cá nhân

1.3 Mục tiêu của hệ thống

Xây dựng website CookShare với các chức năng:

- **Xác thực & Phân quyền:** Đăng ký/đăng nhập, JWT, mã hóa mật khẩu, phân quyền user/moderator/admin
- **Quản lý công thức:** Tạo, chỉnh sửa, xóa, xem chi tiết, tìm kiếm, ảnh từng bước
- **Tương tác:** Bình luận (nested), đánh giá sao, yêu thích, thích bình luận
- **Theo dõi:** Follow/unfollow người dùng, xem profile
- **Báo cáo:** Báo cáo bài viết/bình luận/người dùng, hệ thống quota
- **Thông báo:** Thông báo cá nhân, broadcast, trả lời thông báo
- **Tùy chỉnh giao diện:** Theme cá nhân, marketplace chia sẻ theme
- **Quản trị:** Kiểm duyệt nội dung, xử lý báo cáo, quản lý người dùng

1.4 Phạm vi nghiên cứu

Trong phạm vi:

- **Chức năng người dùng:** quản lý tài khoản, công thức, tương tác, theo dõi
- **Chức năng báo cáo:** báo cáo bài viết/bình luận/người dùng

- **Chức năng thông báo:** cá nhân và hàng loạt
- **Chức năng tùy chỉnh giao diện**
- **Chức năng quản trị:** duyệt/xóa nội dung, thay đổi vai trò
- **Hạ tầng:** frontend React, backend Node/Express, MySQL, Cloudinary

Ngoài phạm vi:

- Realtime notification (WebSocket)
- Gợi ý cá nhân hóa bằng ML
- Cache/Redis
- CI/CD tự động

1.5 Phương pháp và hướng tiếp cận

- **Kiến trúc 3 tầng:** Giao diện (React) – Ứng dụng (Express REST API) – Dữ liệu (MySQL)
- **Mô hình MVC trên backend:** Model (MySQL), Controller (routes Express), View (JSON responses)
- **Bảo mật:** Bcrypt, JWT (7 ngày), CORS
- **Hiệu năng:** Non-blocking I/O, Cloudinary, Scheduled Tasks

1.6 Kết quả mong đợi

Một nền tảng web hoàn chỉnh cho chia sẻ công thức nấu ăn:

- Người dùng đăng nhập, tạo và quản lý công thức, tương tác an toàn
- Theo dõi người dùng khác, xem profile
- Báo cáo nội dung vi phạm
- Tùy chỉnh giao diện cá nhân
- Quản trị viên có công cụ kiểm duyệt, quản lý người dùng/nội dung
- Giao diện thân thiện, phản hồi nhanh, chạy tốt trên nhiều thiết bị

CHƯƠNG 2: NGHIÊN CỨU LÝ THUYẾT

2.1 Kiến trúc và nguyên lý web hiện đại

Kiến trúc 3 tầng (Three-Layer Architecture)

- **Presentation:** React hiển thị UI, quản lý trạng thái client, gọi API
- **Application:** Node.js/Express xử lý logic nghiệp vụ, định tuyến REST
- **Data:** MySQL lưu trữ quan hệ

Mô hình RESTful API

- Tài nguyên được ánh xạ vào URL; thao tác CRUD qua HTTP methods
- Chuẩn hóa mã trạng thái HTTP (200/201/400/401/403/404/500)
- Phản hồi dạng JSON; stateless giữa các request

Mô hình MVC trên backend Express

- **Model:** Tầng truy cập dữ liệu MySQL
- **View:** JSON responses
- **Controller:** Router/handler tổ chức logic và gọi Model

Bảng 2.1: Mã trạng thái HTTP

Mã	Tên	Khi sử dụng	Ví dụ
200	OK	Request thành công	GET /recipe/list → trả danh sách
201	Created	Tạo tài nguyên mới	POST /auth/register → tạo user
400	Bad Request	Lỗi validation từ client	Thiếu trường email khi đăng ký
401	Unauthorized	Chưa xác thực/token sai	Không gửi JWT hoặc token hết hạn
403	Forbidden	Không đủ quyền	User thường gọi API admin
404	Not Found	Tài nguyên không tồn tại	GET /recipe/9999 (ID không có)
409	Conflict	Xung đột dữ liệu	Báo cáo đã tồn tại
500	Internal Server Error	Lỗi server	Exception, lỗi database

2.2 Bảo mật và xác thực

2.2.1 JSON Web Token (JWT)

- Cấu trúc 3 phần: Header (thuật toán, kiểu token), Payload (claims: id, email, role, exp), Signature (chữ ký bảo toàn toàn vẹn).
- Quy trình: đăng nhập → sign token (exp 7 ngày) → client gửi **Authorization: Bearer <token>** → middleware verify.
- Ưu điểm: stateless, dễ mở rộng microservice.

Bảng 2.2: So sánh phương pháp xác thực

Tiêu chí	Session-based	JWT (Token-based)
Lưu trữ server	Có (session store)	Không (stateless)
Scalability	Khó (cần shared session)	Dễ (mỗi server verify độc lập)
Băng thông	Nhẹ (chỉ session ID)	Nặng hơn (toàn bộ token)
Bảo mật	Phụ thuộc session store	Token có thể bị đánh cắp
Hết hạn	Server kiểm soát	Client giữ đến khi exp
Phù hợp	Web truyền thống	API, microservices, mobile

2.2.2 Bcrypt và bảo vệ mật khẩu

- **Adaptive hashing:** điều chỉnh cost theo thời gian.
- **Salt:** chuỗi ngẫu nhiên gắn vào mật khẩu trước khi hash, chống rainbow table.
- **One-way:** không thể giải ngược hash; so sánh bằng bcrypt.compare.

2.2.3 CORS (Cross-Origin Resource Sharing)

- Kiểm soát truy cập giữa các origin
- Preflight (OPTIONS) kiểm tra Access-Control-Allow-Origin/ Methods/ Headers/Credentials.
- Cho phép gửi JWT qua header khi được cấu hình.

2.3 Công nghệ và kỹ thuật triển khai

2.3.1 Frontend: React, React Router, Axios, CSS Responsive

- React: Virtual DOM, component-based, state/props để quản lý UI.
- React Router: điều hướng client-side, bảo vệ tuyến qua ProtectedRoute.
- Axios: HTTP client với interceptor cho JWT.
- CSS Responsive: media queries, grid/flex để thích ứng đa màn hình.

Bảng 2.3: Công nghệ sử dụng

Thành phần	Công nghệ	Phiên bản	Vai trò
Frontend	React	19.x	Xây dựng UI component-based, Virtual DOM
	React Router	6.x	Điều hướng client-side, ProtectedRoute
	Axios	1.x	HTTP client, gọi API backend
	CSS3	-	Responsive design, layout
Backend	Node.js	16.x+	JavaScript runtime, non-blocking I/O
	Express	4.x	Web framework, REST API, middleware

Thành phần	Công nghệ	Phiên bản	Vai trò
	jsonwebtoken	9.x	Tạo và verify JWT token
	bcryptjs	2.x	Hash mật khẩu với Salt
	Multer	1.x	Upload file
	nodemailer	6.x	Gửi email
Database	MySQL	8.0+	Cơ sở dữ liệu quan hệ, ACID
Cloud Storage	Cloudinary	-	Lưu trữ và CDN hóa ảnh
Tools	npm	-	Quản lý package dependencies

2.3.2 Backend: Node.js (Non-blocking I/O), Express Middleware

- Non-blocking I/O: event loop xử lý nhiều request đồng thời.
- Middleware: xử lý tuần tự (logging, CORS, body parsing, auth) trước handler.
- Validation: kiểm tra đầu vào, trả mã lỗi phù hợp.

2.3.3 Dữ liệu và lưu trữ

- MySQL: quan hệ, ACID; khóa ngoại giữa các bảng.
- Cloudinary: lưu trữ ảnh trên cloud, trả URL tối ưu để hiển thị trên frontend.

2.4 Cấu trúc dữ liệu và thuật toán cho nested comments

2.4.1 Nested Comments (Bình luận lồng nhau)

- Cách triển khai: Sử dụng cột `parent_comment_id` trong bảng `binh_luan`
- Lợi ích: Người dùng có thể trả lời trực tiếp bình luận của người khác.
- Truy vấn recursion: Backend truy vấn theo `parent_comment_id = NULL` (top-level), sau đó lấy nested với `parent_comment_id = comment_id`.
- Frontend: Sử dụng recursive component (CommentItem gọi lại chính nó) để render tree structure.

2.4.2 Like System (Hệ thống thích)

- Bảng riêng `comment_likes` lưu trữ: `comment_id, user_id, created_at`.
- Ràng buộc UNIQUE: Mỗi user chỉ thích 1 lần mỗi comment (toggle like/unlike).
- Toggle endpoint: POST `/comment/:id/like` trả về `{liked: true/false}`.
- Query optimization: Include `like_count, user_liked` khi fetch comments.

2.4.3 Follow System (Hệ thống theo dõi)

- Mô hình quan hệ nhiều-nhiều: Một user có thể follow nhiều user khác và được follow bởi nhiều user
- Bảng `follow` với 2 khóa ngoại: follower_id (người theo dõi) và following_id (người được theo dõi)
- Ràng buộc UNIQUE (follower_id, following_id) đảm bảo không follow trùng
- Kiểm tra không cho phép follow chính mình
- Query đếm followers/following sử dụng COUNT với điều kiện phù hợp

2.4.4 Report Quota System (Hệ thống quota báo cáo)

- Mục đích: Chống spam báo cáo, đảm bảo người dùng báo cáo có trách nhiệm
- Mỗi user có 3 lượt báo cáo cho mỗi loại (recipe/comment/user)
- Quota được hoàn lại khi báo cáo được xử lý (accept hoặc reject)
- Nếu báo cáo bị reject quá nhiều (3 lần/tuần), user bị khóa tính năng báo cáo 30 ngày
- Bảng `user_report_quota` theo dõi số lượt còn lại theo từng loại

2.4.5 Violation Auto-Block System (Hệ thống tự động khóa khi vi phạm)

- Theo dõi số vi phạm trong khoảng thời gian (7 ngày cho bài viết, 1 tháng cho bình luận)
- Ngưỡng khóa:
 - + 3 bài viết bị ẩn trong tháng → khóa đăng bài 30 ngày
 - + 3 bình luận vi phạm trong tháng → khóa bình luận 30 ngày
 - + 3 báo cáo bị bác bỏ trong tuần → khóa báo cáo 30 ngày
- Tự động mở khóa khi hết hạn (kiểm tra tại middleware hoặc scheduled task)
- Gửi email thông báo khi bị khóa

2.4.6 Broadcast Notification System (Hệ thống thông báo hàng loạt)

- Đã triển khai ở mức demo/mô phỏng
- Khác với thông báo cá nhân (1-1), broadcast gửi đến tất cả user
- Bảng riêng `broadcast_notifications` lưu nội dung broadcast
- Bảng `user_broadcast_read` theo dõi user nào đã đọc broadcast nào
- Khi fetch thông báo: kết hợp personal notifications và broadcast chưa đọc
- Tối ưu: không tạo N bản ghi cho N user, chỉ tạo 1 bản ghi broadcast

CHƯƠNG 3: HIỆN THỰC HÓA NGHIÊN CỨU

3.1 Đặc tả nhu cầu

Tác nhân:

- **Người dùng (user):** Đăng ký/đăng nhập, tạo-sửa-xóa công thức của mình, xem/tìm kiếm, bình luận, đánh giá, yêu thích, theo dõi người dùng khác, báo cáo vi phạm, tùy chỉnh giao diện
- **Kiểm duyệt viên (moderator):** Toàn bộ quyền của user + xử lý báo cáo, ẩn bài viết vi phạm, gửi thông báo
- **Quản trị viên (admin):** Toàn bộ quyền của moderator + đổi vai trò người dùng, xóa tài khoản, quản lý toàn hệ thống

Chức năng chính:

- **Xác thực & phân quyền:** đăng ký, đăng nhập, JWT 7 ngày, phân biệt user/moderator/admin
- **Quản lý công thức:** CRUD, tải ảnh lên Cloudinary, tìm kiếm, ảnh từng bước
- **Tương tác:** bình luận (nested), đánh giá sao, yêu thích, thích bình luận
- **Theo dõi:** follow/unfollow, xem profile người dùng
- **Báo cáo:** báo cáo bài viết/bình luận/người dùng, quota system
- **Thông báo:** cá nhân, broadcast, reply
- **Tùy chỉnh giao diện:** theme cá nhân, marketplace
- **Quản trị:** quản lý người dùng, xử lý báo cáo, ẩn bài viết

Phi chức năng:

- Bảo mật: Bcrypt, JWT, CORS, middleware phân quyền
- Hiệu năng: Non-blocking I/O, tách lưu trữ ảnh sang cloud
- UX/UI: Responsive, điều hướng mượt, thông báo rõ ràng

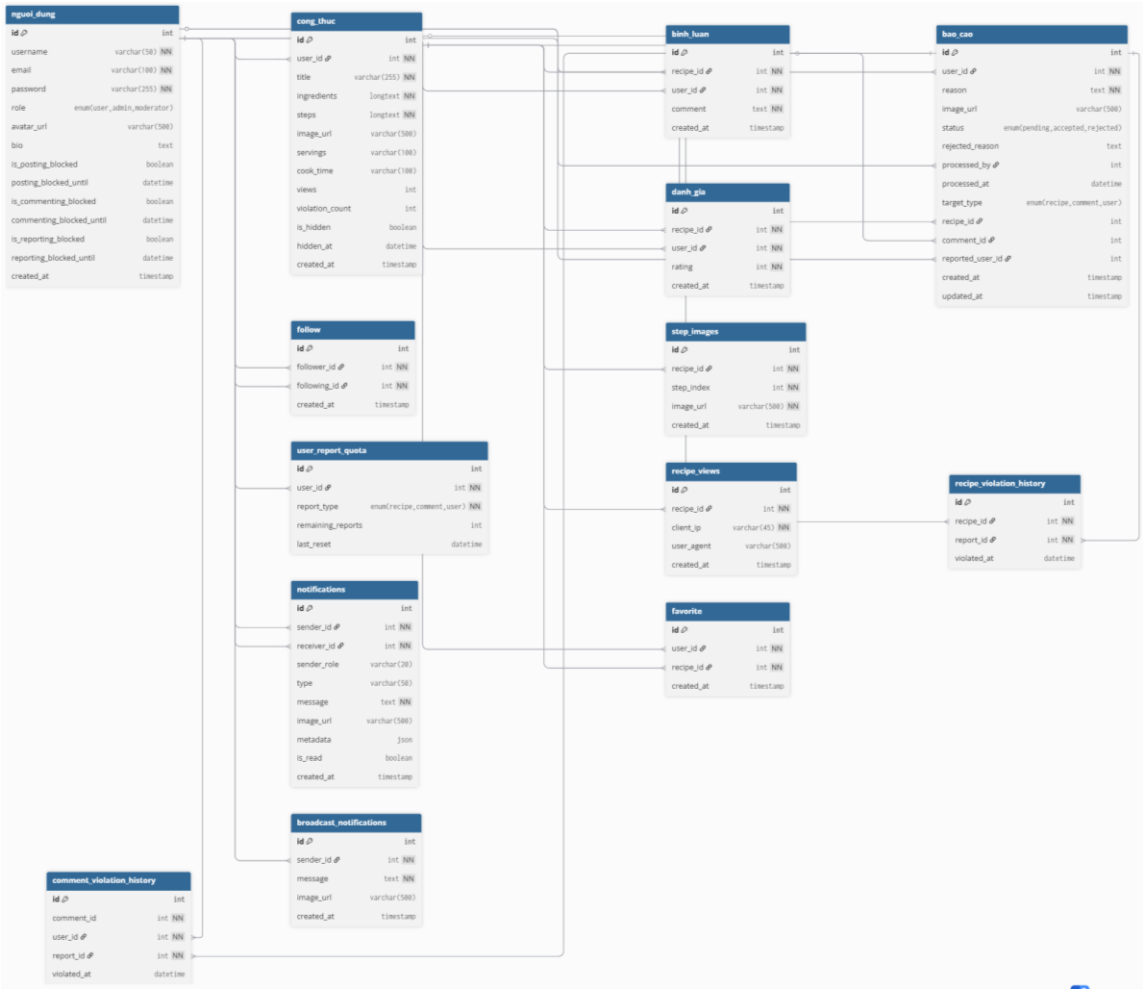
3.2 Thiết kế hệ thống

3.2.1 Kiến trúc tổng thể

- **3 tầng:** Frontend (React) ↔ Backend (Express REST) ↔ Database (MySQL); ảnh trên Cloudinary.
- **MVC (backend):** Model (MySQL queries), Controller (Express routes), View (JSON response).
- **Giao tiếp:** HTTP/HTTPS, JSON; JWT gửi qua header Authorization.

3.2.2 Mô hình dữ liệu

Mô hình dữ liệu ERD:



Hình 3.1 Mô hình dữ liệu ERD

Bảng 3.1: Schema cơ sở dữ liệu

Bảng	Trường	Kiểu dữ liệu	Ràng buộc
nguoi_dung	id	INT	PRIMARY KEY, AUTO_INCREMENT
	username	VARCHAR(50)	UNIQUE, NOT NULL
	email	VARCHAR(100)	UNIQUE, NOT NULL
	password	VARCHAR(255)	NOT NULL (Bcrypt hash)

Bảng	Trường	Kiểu dữ liệu	Ràng buộc
	role	ENUM	'user', 'moderator', 'admin'
	avatar_url	VARCHAR(500)	NULL
	bio	TEXT	NULL
	is_posting_blocked	BOOLEAN	DEFAULT FALSE
	is_commenting_blocked	BOOLEAN	DEFAULT FALSE
	is_reporting_blocked	BOOLEAN	DEFAULT FALSE
cong_thuc	id	INT	PRIMARY KEY, AUTO_INCRE MENT
	user_id	INT	FOREIGN KEY → nguoi_dung(id)
	title	VARCHAR(255)	NOT NULL
	ingredients	LONGTEXT	NOT NULL
	steps	LONGTEXT	NOT NULL
	image_url	VARCHAR(500)	NULL
	servings	VARCHAR(100)	NULL
	cook_time	VARCHAR(100)	NULL
	views	INT	DEFAULT 0
	violation_count	INT	DEFAULT 0
	is_hidden	BOOLEAN	DEFAULT FALSE
binh_luan	id	INT	PRIMARY KEY, AUTO_INCRE MENT
	recipe_id	INT	FOREIGN

Bảng	Trường	Kiểu dữ liệu	Ràng buộc
			KEY → cong_thuc(id)
	user_id	INT	FOREIGN KEY → nguoi_dung(id)
	comment	TEXT	NOT NULL
	parent_comment_id	INT	FOREIGN KEY → binh_luan(id), NULL
comment_likes	id	INT	PRIMARY KEY, AUTO_INCRE MENT
	comment_id	INT	FOREIGN KEY → binh_luan(id)
	user_id	INT	FOREIGN KEY → nguoi_dung(id)
			UNIQUE (comment_id, user_id)
danh_gia	id	INT	PRIMARY KEY, AUTO_INCRE MENT
	recipe_id	INT	FOREIGN KEY → cong_thuc(id)
	user_id	INT	FOREIGN KEY → nguoi_dung(id)

Bảng	Trường	Kiểu dữ liệu	Ràng buộc
	rating	INT	CHECK (1–5), UNIQUE (recipe_id, user_id)
favorite	id	INT	PRIMARY KEY, AUTO_INCREMENT
	recipe_id	INT	FOREIGN KEY → cong_thuc(id)
	user_id	INT	FOREIGN KEY → nguoi_dung(id)
			UNIQUE (recipe_id, user_id)
follow	id	INT	PRIMARY KEY, AUTO_INCREMENT
	follower_id	INT	FOREIGN KEY → nguoi_dung(id)
	following_id	INT	FOREIGN KEY → nguoi_dung(id)
			UNIQUE (follower_id, following_id)
bao_cao	id	INT	PRIMARY KEY, AUTO_INCREMENT

Bảng	Trường	Kiểu dữ liệu	Ràng buộc
			MENT
	user_id	INT	FOREIGN KEY → nguoi_dung(id)
	target_type	ENUM	'recipe', 'comment', 'user'
	status	ENUM	'pending', 'accepted', 'rejected'
	recipe_id	INT	FOREIGN KEY, NULL
	comment_id	INT	FOREIGN KEY, NULL
	reported_user_id	INT	FOREIGN KEY, NULL
	image_url	VARCHAR(500)	NULL (ảnh bằng chứng)
	processed_by	INT	FOREIGN KEY → nguoi_dung(id), NULL
user_report_quota	user_id	INT	FOREIGN KEY → nguoi_dung(id)
	report_type	ENUM	'recipe', 'comment', 'user'
	remaining_reports	INT	DEFAULT 3 UNIQUE (user_id, report_type)
notifications	id	INT	PRIMARY KEY, AUTO_INCRE

Bảng	Trường	Kiểu dữ liệu	Ràng buộc
			MENT
	sender_id	INT	FOREIGN KEY → nguoi_dung(id)
	receiver_id	INT	FOREIGN KEY → nguoi_dung(id)
	type	VARCHAR(50)	DEFAULT 'manual'
	message	TEXT	NOT NULL
	is_read	BOOLEAN	DEFAULT FALSE
broadcast_notifications	id	INT	PRIMARY KEY, AUTO_INCRE MENT
	sender_id	INT	FOREIGN KEY → nguoi_dung(id)
	message	TEXT	NOT NULL
user_theme_preferences	user_id	INT	UNIQUE, FOREIGN KEY → nguoi_dung(id)
	primary_color	VARCHAR(7)	DEFAULT 'ff7f50'
	is_shared	BOOLEAN	DEFAULT FALSE
step_images	id	INT	PRIMARY KEY, AUTO_INCRE MENT
	recipe_id	INT	FOREIGN

Bảng	Trường	Kiểu dữ liệu	Ràng buộc
			KEY → cong_thuc(id)
	step_index	INT	NOT NULL
	image_url	VARCHAR(500)	NOT NULL
recipe_views	id	INT	PRIMARY KEY, AUTO_INCREMENT
	recipe_id	INT	INT
	client_ip	VARCHAR(45)	NOT NULL
	user_agent	VARCHAR(500)	NULL

3.2.3 Thiết kế API (REST)

- Auth: POST /auth/register, POST /auth/login
- Recipe: GET /recipe/list, GET /recipe/search, POST /recipe/create, [PUT /recipe/update/:id](#), [DELETE /recipe/delete/:id](#)
- Interaction: POST /rating/:recipeId, POST /favorite/:recipeId, POST /comment/:recipeId
- Admin: GET /admin/users, [PUT /admin/users/:id/role](#), [DELETE /admin/recipes/:id](#)
- Mã trạng thái: 200/201/400/401/403/404/500; thông điệp lỗi rõ ràng.

3.2.4 Thiết kế giao diện (UI/UX)

- Trang chính: danh sách công thức, thanh tìm kiếm, lưới ảnh.
- Navbar: logo, đăng nhập/đăng ký, tên người dùng + logout, link admin nếu role=admin.
- Trang chi tiết công thức: nguyên liệu, bước, ảnh, bình luận, đánh giá.
- Trang quản lý cá nhân: My Recipes, Favorite Recipes.
- Trang quản trị: danh sách người dùng, đổi role, xóa công thức.

3.3 Hiện thực hóa (Implementation)

3.3.1 Frontend (React)

Công nghệ: React 19.x, React Router 6.x, Axios 1.x, CSS responsive.

Cấu trúc thư mục:

```

```
src/cookshare/src/
├── components/
│ ├── Navbar.jsx Thanh điều hướng
│ ├── ProtectedRoute.jsx Bảo vệ route cần đăng nhập
│ ├── FollowButton.jsx Nút theo dõi
│ ├── FollowersList.jsx Danh sách followers/following
│ ├── ReportButton.jsx Nút báo cáo
│ ├── ProfileHeader.jsx Header trang profile
│ ├── ImageLightbox.jsx Xem ảnh phóng to
│ ├── BroadcastNotification.jsx Thông báo hàng loạt
│ ├── AvatarCropper.jsx Crop avatar
│ ├── Footer.jsx Footer
│ └── RulesModal.jsx Modal quy định
├── pages/
│ ├── Home.jsx Trang chủ
│ ├── Login.jsx Đăng nhập
│ ├── Register.jsx Đăng ký
│ ├── ForgotPassword.jsx Quên mật khẩu
│ ├── CreateRecipe.jsx Tạo công thức
│ ├── EditRecipe.jsx Sửa công thức
│ ├── RecipeDetail.jsx Chi tiết công thức
│ ├── MyRecipes.jsx Công thức của tôi
│ ├── FavoriteRecipes.jsx Công thức yêu thích
│ ├── Search.jsx Tìm kiếm
│ ├── Profile.jsx Profile cá nhân
│ ├── UserProfile.jsx Profile người khác
│ ├── AccountSettings.jsx Cài đặt tài khoản
│ ├── Notifications.jsx Thông báo
│ ├── ThemeCustomization.jsx Tùy chỉnh giao diện
│ └── ThemeMarketplace.jsx Marketplace theme
```

```
| |— AdminDashboard.jsx Quản trị người dùng
| |— AdminReports.jsx Quản lý báo cáo
| |— Rules.jsx Quy định cộng đồng
|— styles/ CSS files
|— hooks/Custom hooks
|— utils/Utility functions
...

```

### 3.3.2 Backend (Node.js + Express)

**Công nghệ:** Node.js 16+, Express 4.x, jsonwebtoken, bcryptjs, multer/cloudinary SDK, dotenv.

**Cấu trúc:**

```
...
src/backend/
|— config/
| |— db.js Kết nối MySQL
| |— cloudinary.js Cấu hình Cloudinary
| |— mailer.js Cấu hình Nodemailer
|— middleware/
| |— auth.js Middleware xác thực JWT, phân quyền
|— routes/
| |— auth.js Xác thực (register, login, profile)
| |— recipe.js Quản lý công thức, bình luận
| |— rating.js Đánh giá
| |— favorite.js Yêu thích
| |— follow.js Theo dõi người dùng
| |— report.js Báo cáo vi phạm
| |— notification.js Thông báo
| |— theme.js Tùy chỉnh giao diện
| |— admin.js Quản trị hệ thống
|— scripts/
| |— scheduled_tasks.js Tác vụ tự động định kỳ
| |— [migration scripts] Scripts tạo bảng, migrate
|— uploads/Thư mục lưu file tạm
|— server.js Entry point
...

```



### Luồng chính:

- **Auth:** Hash mật khẩu với Bcrypt, tạo JWT (exp 7 ngày), middleware verify token và kiểm tra role
- **Recipe:** CRUD công thức + upload ảnh lên Cloudinary, kiểm tra quyền owner/admin/moderator
- **Follow:** Follow/unfollow người dùng, kiểm tra không cho follow chính mình, đếm followers/following
- **Report:** Báo cáo bài viết/bình luận/người dùng, hệ thống quota, tự động khóa tính năng khi vi phạm
- **Notification:** Gửi thông báo cá nhân, broadcast, trả lời thông báo
- **Theme:** Lưu/lấy theme preferences, chia sẻ theme lên marketplace

### 3.3.3 Database (MySQL)

#### Bảng chính:

- `nguoi\_dung` Người dùng với các trường khóa tính năng
- `cong\_thuc` Công thức với views, violation\_count, is\_hidden
- `step\_images` Ảnh từng bước nấu
- `binh\_luan` Bình luận với parent\_comment\_id cho nested
- `comment\_likes` Like bình luận
- `danh\_gia` Đánh giá sao
- `favorite` Yêu thích
- `follow` Theo dõi người dùng
- `bao\_cao` Báo cáo vi phạm (đa loại)
- `user\_report\_quota` Quota báo cáo
- `recipe\_violation\_history` Lịch sử vi phạm
- `notifications` Thông báo cá nhân
- `broadcast\_notifications` Thông báo hàng loạt
- `user\_broadcast\_read` Đánh dấu đã đọc broadcast
- `user\_theme\_preferences` Tùy chỉnh giao diện
- `admin\_hidden\_recipes` Ẩn bài viết thủ công
- `recipe\_views` Theo dõi lượt xem

### 3.4 Bảo mật và quản lý phiên

- **Bcrypt:** Hash mật khẩu với Salt, cost 10
- **JWT:** Lưu trên client; gửi Bearer token; middleware verify; exp 7 ngày
- **CORS:** Cho phép credentials; hạn chế header/method cần thiết
- **Phân quyền:** Middleware kiểm tra role cho route admin/moderator; kiểm tra owner trước khi sửa/xóa
- **Khóa tính năng:** Tự động khóa đăng bài/bình luận/báo cáo khi vi phạm nhiều lần

*Bảng 3.2: Phân quyền user vs admin*

| Hành động                           | User | Moderator | Admin |
|-------------------------------------|------|-----------|-------|
| Xem danh sách công thức             | ✓    | ✓         | ✓     |
| Tìm kiếm công thức                  | ✓    | ✓         | ✓     |
| Tạo công thức mới                   | ✓    | ✓         | ✓     |
| Sửa công thức của mình              | ✓    | ✓         | ✓     |
| Xóa công thức của mình              | ✓    | ✓         | ✓     |
| Xóa công thức của người khác        | X    | ✓         | ✓     |
| Bình luận, đánh giá                 | ✓    | ✓         | ✓     |
| Reply bình luận (nested)            | ✓    | ✓         | ✓     |
| Like / Unlike bình luận             | ✓    | ✓         | ✓     |
| Xóa bình luận của người khác        | X    | ✓         | ✓     |
| Theo dõi người dùng                 | ✓    | ✓         | ✓     |
| Báo cáo vi phạm                     | ✓    | ✓         | ✓     |
| Xem danh sách báo cáo               | X    | ✓         | ✓     |
| Xử lý báo cáo                       | X    | ✓         | ✓     |
| Ấn bài viết thủ công                | X    | ✓         | ✓     |
| Gửi thông báo đến người dùng        | X    | ✓         | ✓     |
| Gửi thông báo hàng loạt (broadcast) | X    | ✓         | ✓     |
| Xem danh sách người dùng            | X    | X         | ✓     |
| Đổi vai trò người dùng              | X    | X         | ✓     |
| Xóa tài khoản người dùng            | X    | X         | ✓     |
| Tùy chỉnh giao diện                 | ✓    | ✓         | ✓     |

| Hành động                             | User | Moderator | Admin |
|---------------------------------------|------|-----------|-------|
| Chia sẻ theme                         | ✓    | ✓         | ✓     |
| Xóa theme của người khác              | ✗    | ✗         | ✓     |
| Nhận thông báo                        | ✓    | ✓         | ✓     |
| Lưu công thức yêu thích               | ✓    | ✓         | ✓     |
| Xem profile người khác                | ✓    | ✓         | ✓     |
| Chỉnh sửa profile của mình            | ✓    | ✓         | ✓     |
| Hủy báo cáo đang ở trạng thái pending | ✓    | ✓         | ✓     |
| Bị khóa tính năng khi vi phạm         | ✓    | ✓         | ✓     |

### 3.5 Thiết kế chi tiết nested comments và like system

#### 3.5.1 Nested Comments

- Khi user trả lời bình luận, gửi POST với `parent\_comment\_id`
- Backend lưu vào bảng `binh\_luan` với trường `parent\_comment\_id` không NULL
- Frontend: Component recursive, render reply form inline

#### 3.5.2 Like System

- Bảng riêng `comment\_likes` (comment\_id, user\_id, UNIQUE)
- Toggle endpoint: kiểm tra UNIQUE constraint; nếu tồn tại → delete, không tồn tại → insert

#### 3.5.3 Follow System

- Bảng `follow` với `follower\_id` và `following\_id`
- Không cho phép follow chính mình
- Endpoints: follow, unfollow, check is-following, counts, list followers/following

#### 3.5.4 Report System

- 3 loại báo cáo: recipe, comment, user
- Quota system: Mỗi user có 3 lượt báo cáo mỗi loại
- Upload ảnh bằng chứng khi báo cáo
- Tự động khóa tính năng khi vi phạm nhiều lần

### **3.5.5 Notification System**

- Thông báo cá nhân: Admin/moderator gửi đến user cụ thể
- Thông báo hàng loạt (broadcast): Gửi đến tất cả user
- Hỗ trợ đính kèm ảnh, trả lời thông báo

### **3.5.6 Theme System**

- Lưu preferences: primary\_color, background\_image, theme\_name
- Chia sẻ theme lên marketplace (is\_shared = true)
- Marketplace: Danh sách theme công khai, có thể áp dụng

## **3.6 Kiểm thử**

- Chức năng: Đăng ký/đăng nhập, CRUD công thức, tìm kiếm, bình luận nested, đánh giá, yêu thích, theo dõi, báo cáo, thông báo, tùy chỉnh giao diện
- Bảo mật: Từ chối request thiếu/invalid JWT, kiểm tra owner, kiểm tra quota
- Hiệu năng: Đo thời gian phản hồi API, kiểm tra responsive

## CHƯƠNG 4: KẾT QUẢ NGHIÊN CỨU

### 4.1 Kết quả chức năng

- Hoàn thiện luồng xác thực: đăng ký, đăng nhập, JWT (hết hạn 7 ngày), phân quyền user/moderator/admin
- Quản lý công thức: tạo, sửa, xóa, xem chi tiết với ảnh từng bước; tìm kiếm; đếm lượt xem; upload ảnh lên Cloudinary
- Tương tác người dùng: bình luận nested, like bình luận, đánh giá sao, yêu thích
- Theo dõi người dùng: follow/unfollow, xem profile, đếm followers/following
- Báo cáo vi phạm: 3 loại (bài viết/bình luận/người dùng), quota system, upload bằng chứng, tự động khóa tính năng
- Thông báo: cá nhân, broadcast, reply, đếm chưa đọc
- Tùy chỉnh giao diện: theme cá nhân, marketplace chia sẻ
- Quản trị: quản lý người dùng, xử lý báo cáo, ẩn bài viết, gửi thông báo
- API REST chuẩn: mã trạng thái 200/201/400/401/403/404/409/500
- Phân trang (pagination) cho danh sách công thức
- Lọc công thức theo category/cuisine/tags (nếu đã implement)
- Scroll to top button
- Chia sẻ công thức (share button)

### 4.2 Trải nghiệm người dùng (UX/UI)

- **Giao diện responsive:** hiển thị tốt trên desktop, tablet, mobile
- **Navbar động:** hiển thị nút đăng nhập/dăng ký khi chưa login; tên người dùng + logout + badge thông báo khi đã login; link admin cho admin/moderator
- **Trang chủ:** lưới công thức, thanh tìm kiếm, nút tạo công thức khi đã đăng nhập
- **Trang chi tiết:** ảnh, nguyên liệu, bước nấu với ảnh từng bước; bình luận nested và đánh giá; nút yêu thích, báo cáo
- **Trang profile:** avatar, bio, số followers/following, danh sách công thức, nút follow
- **Trang cá nhân:** My Recipes, Favorite Recipes, Account Settings
- **Trang thông báo:** danh sách thông báo với nút trả lời, đánh dấu đã đọc
- **Trang tùy chỉnh giao diện:** chọn màu, upload hình nền, preview
- **Theme Marketplace:** danh sách theme, nút áp dụng
- **Trang quản trị:** bảng người dùng, đổi role; danh sách báo cáo, xử lý; gửi thông báo
- Lightbox xem ảnh phóng to
- Modal quy định cộng đồng

### 4.3 Hiệu năng và kỹ thuật

- Non-blocking I/O (Node.js + Express) xử lý đồng thời nhiều request
- Upload ảnh chuyển sang Cloudinary, giảm tải server
- CORS cấu hình an toàn; JWT qua header Authorization
- MySQL đáp ứng truy vấn; ràng buộc khóa ngoại đảm bảo toàn vẹn
- Scheduled tasks tự động xử lý định kỳ

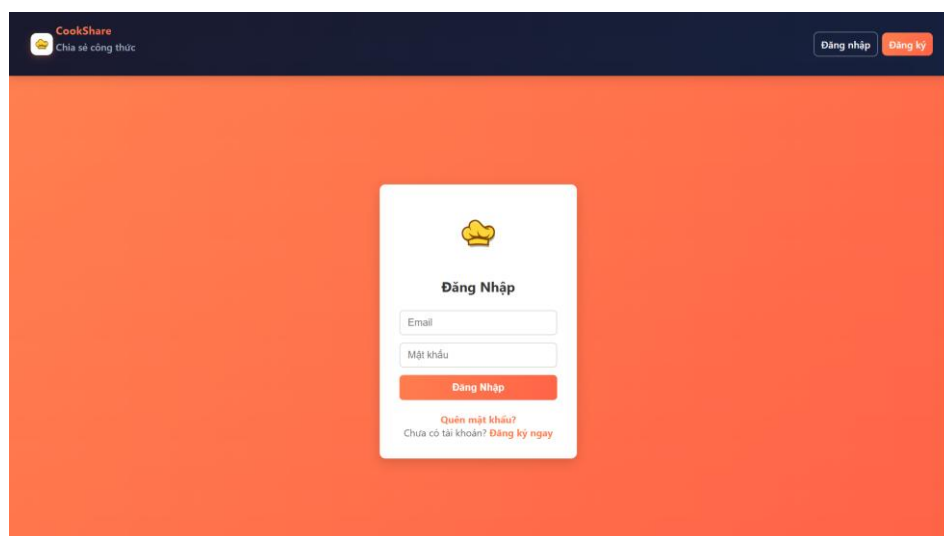
### 4.4 Bảo mật

- Mật khẩu được hash với Bcrypt (Salt + adaptive hashing)
- JWT có hạn 7 ngày, kiểm tra ở middleware
- Phân quyền rõ ràng: user / moderator / admin
- Không lưu mật khẩu gốc
- Từ chối truy cập khi thiếu/invalid token
- Hạn chế origin qua CORS
- Kiểm tra owner trước khi sửa/xóa
- Quota system chống spam báo cáo
- Tự động khóa tính năng khi vi phạm

### 4.5 Giao diện minh họa (mô tả nhanh)

Phần này trình bày chi tiết các giao diện chức năng của hệ thống CookShare, bao gồm mô tả các thành phần và tương tác người dùng.

#### 4.5.1 Giao diện xác thực người dùng



Hình 4.1: Giao diện Đăng nhập (Login)

Mô tả: Giao diện đăng nhập cho phép người dùng truy cập vào hệ thống bằng tài khoản đã đăng ký. Các thành phần chính bao gồm:

- Trường nhập email: Yêu cầu định dạng email hợp lệ
- Trường nhập mật khẩu: Hiện thị dạng ẩn với nút toggle hiện/ẩn
- Nút "Đăng nhập": Gửi thông tin xác thực đến server
- Link "Quên mật khẩu": Chuyển đến trang khôi phục mật khẩu
- Link "Đăng ký": Chuyển đến trang tạo tài khoản mới
- Thông báo lỗi hiển thị khi thông tin không hợp lệ

CookShare  
Chia sẻ công thức

Đăng nhập Đăng ký

**Đăng Ký**

Tên đăng nhập

Email

Mật khẩu

Xác nhận mật khẩu

☐ Tôi đồng ý với [Quy tắc cộng đồng](#)

Đăng Ký

Đã có tài khoản? [Đăng nhập tại đây](#)

Hình 4.2: Giao diện Đăng ký (Register)

Mô tả: Giao diện đăng ký cho phép người dùng mới tạo tài khoản. bao gồm:

- Trường nhập tên người dùng (username): Tối thiểu 3 ký tự, không trùng lặp
- Trường nhập email: Định dạng email hợp lệ, không trùng lặp
- Trường nhập mật khẩu: Tối thiểu 6 ký tự
- Trường xác nhận mật khẩu: Phải khớp với mật khẩu đã nhập
- Nút "Đăng ký": Tạo tài khoản mới
- Link "Đã có tài khoản": Chuyển về trang đăng nhập
- Validation realtime hiển thị lỗi ngay khi nhập

CookShare  
Chia sẻ công thức

Đăng nhập Đăng ký

**Quên Mật Khẩu**

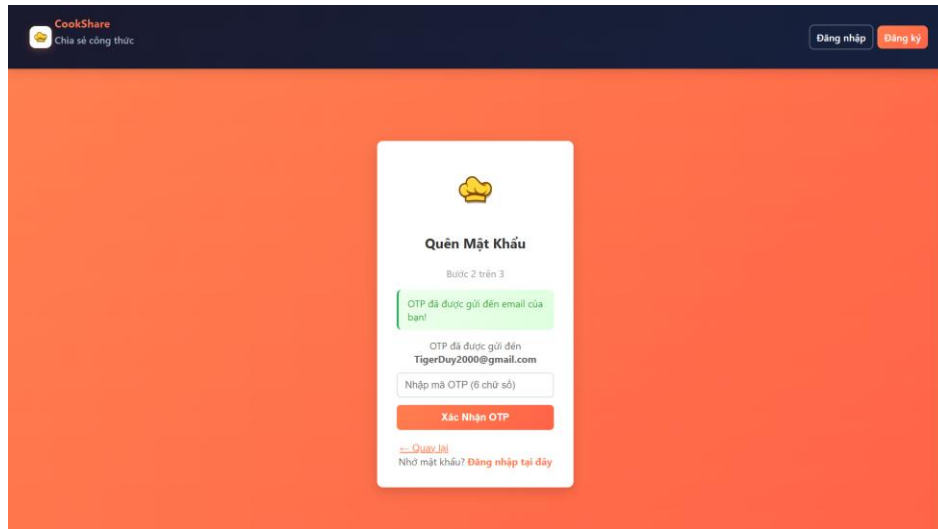
Bước 1 trên 3

Nhập email của bạn

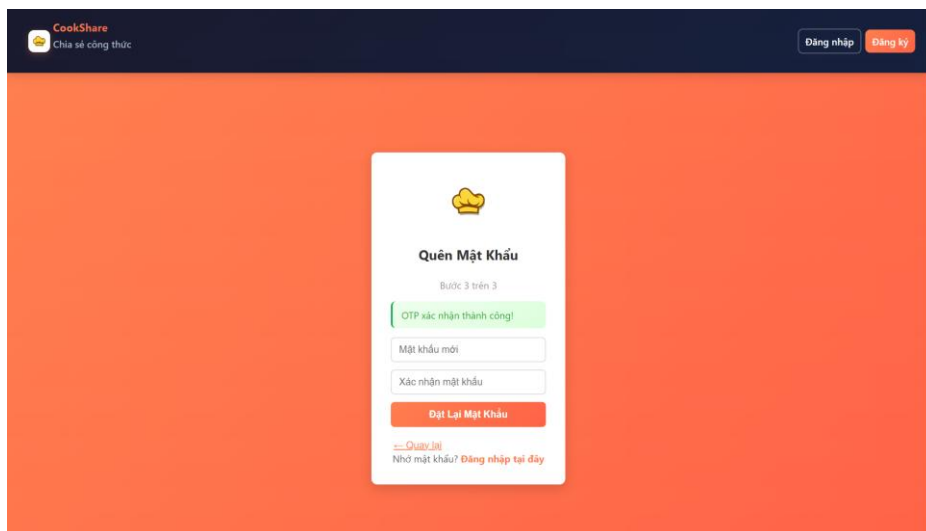
Gửi OTP

Nhớ mật khẩu? [Đăng nhập tại đây](#)

Hình 4.3: Giao diện Quên mật khẩu nhập Email(Forgot Password)



Hình 4.3.1: Giao diện Quên mật khẩu nhập OTP (Forgot Password)



Hình 4.3.2: Giao diện Quên mật khẩu nhập mật khẩu mới (Forgot Password)

Mô tả: Giao diện khôi phục mật khẩu qua email. Các thành phần chính bao gồm:

- Trường nhập email đã đăng ký
- Nút "Gửi mã OTP": Gửi mã xác thực đến email
- Trường nhập mã OTP (hiển thị sau khi gửi)
- Trường nhập mật khẩu mới
- Trường xác nhận mật khẩu mới
- Nút "Đặt lại mật khẩu": Cập nhật mật khẩu mới
- Thông báo trạng thái gửi email và kết quả



## 4.5.2 Giao diện trang chủ và điều hướng



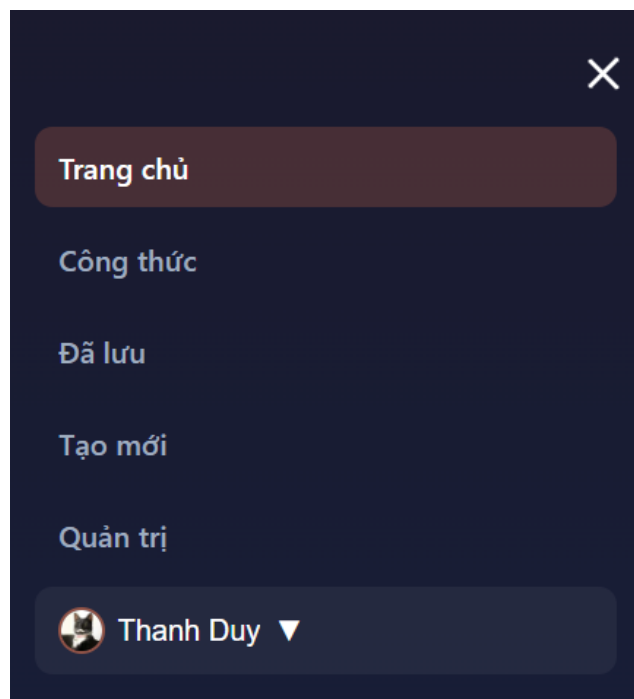
Hình 4.4.1: Thanh điều hướng khi chưa đăng nhập (Navbar)



Hình 4.4.2: Thanh điều hướng Khi đã đăng nhập của User (Navbar)



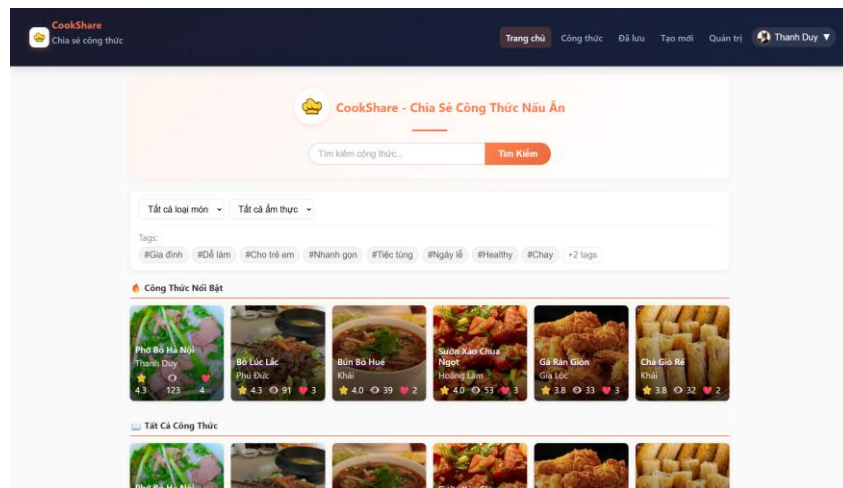
Hình 4.4.3: Thanh điều hướng Khi đã đăng nhập của admin/moderator (Navbar)



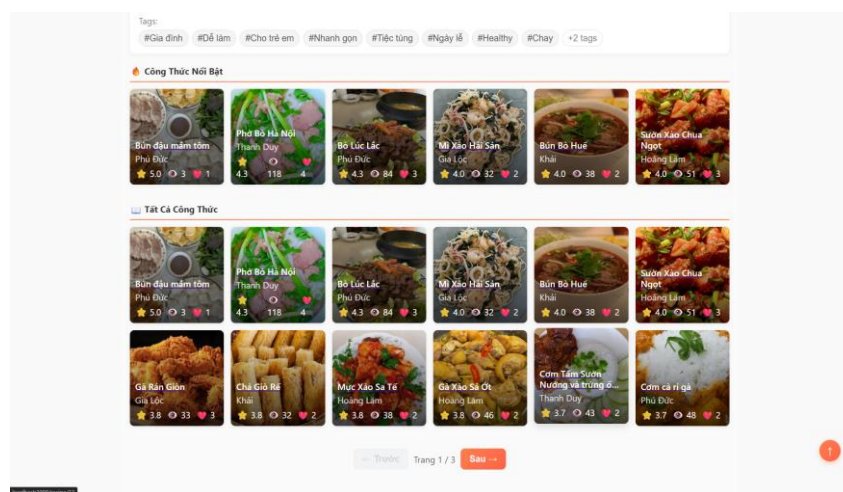
Hình 4.4.4: Thanh điều hướng khi đăng nhập trên mobile (Navbar)

Mô tả: Thanh điều hướng cố định ở đầu trang, thay đổi theo trạng thái đăng nhập. Các thành phần chính bao gồm:

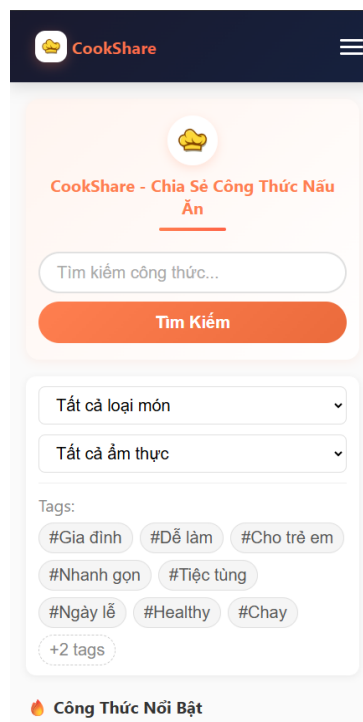
- Logo và tên ứng dụng "CookShare": Click để về trang chủ
- Khi chưa đăng nhập: Nút "Đăng nhập" và "Đăng ký"
- Khi đã đăng nhập:
  - + Tên người dùng với dropdown menu (Profile, My Recipes, Favorites, Settings, Logout)
  - + Icon thông báo với badge đếm số chưa đọc
  - + Nút "Tạo công thức" (icon +)
- Khi là admin/moderator: Thêm link "Quản trị"
- Responsive: Thu gọn thành hamburger menu trên mobile



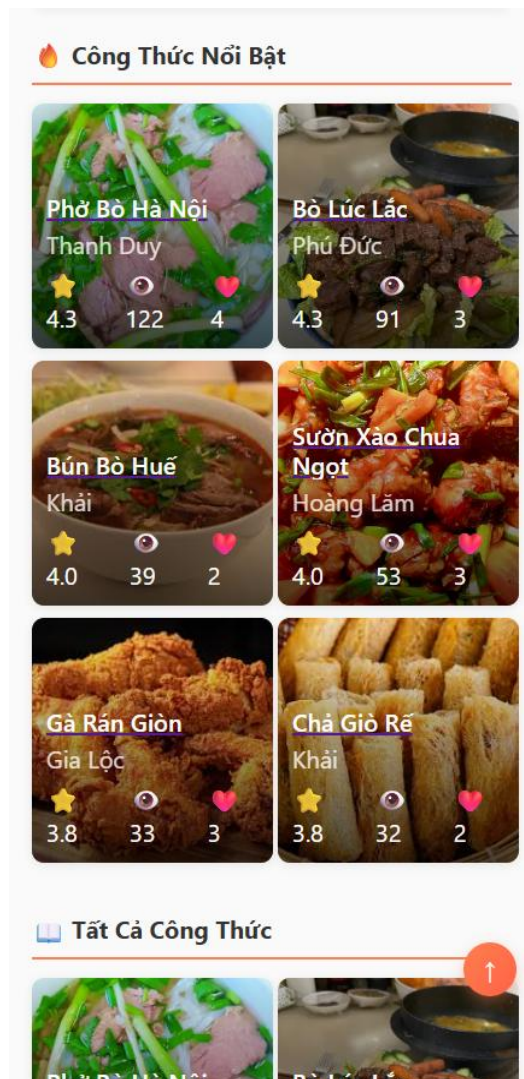
Hình 4.5: Giao diện Trang chủ (Home)



Hình 4.5.1: Giao diện Trang chủ Phân trang (Home)



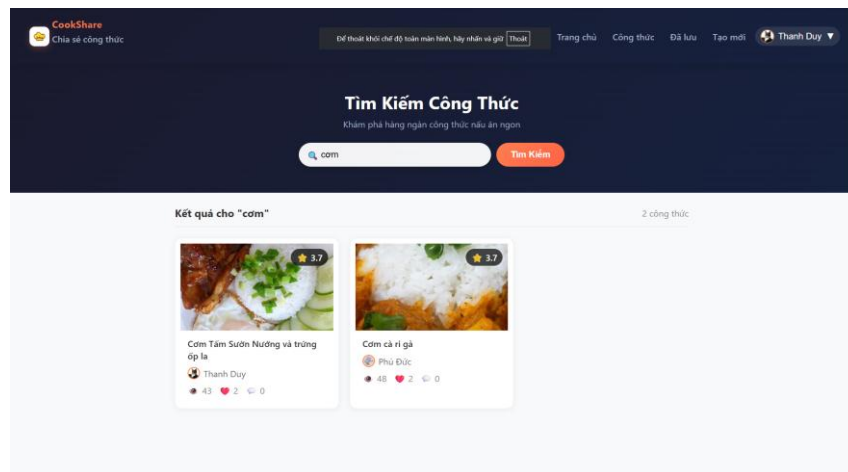
Hình 4.5.2: Giao diện Trang chủ trên mobile (Home)



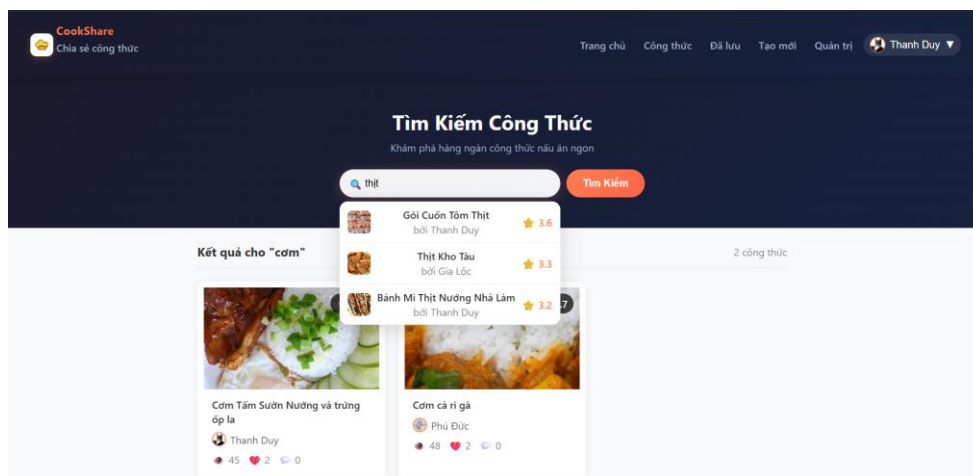
Hình 4.5.3: Giao diện Trang chủ phần danh sách công thức trên mobile (Home)

Mô tả: Trang chủ hiển thị danh sách công thức nấu ăn dạng lưới. bao gồm:

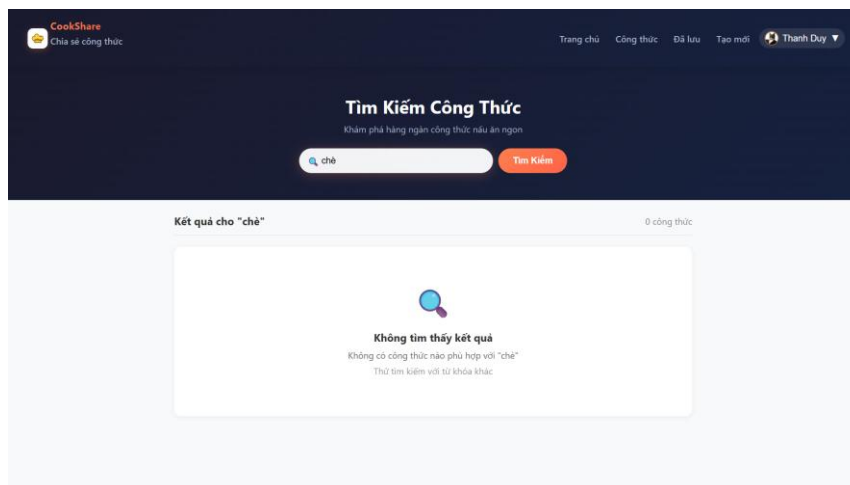
- Thanh tìm kiếm: Nhập từ khóa để tìm công thức
- Bộ lọc: Lọc theo danh mục, thời gian nấu, độ khó
- Lưới công thức (Recipe Cards): Mỗi card hiển thị:
  - + Ảnh đại diện công thức
  - + Tiêu đề công thức
  - + Tên tác giả (click để xem profile)
  - + Đánh giá trung bình (sao)
  - + Thời gian nấu và khẩu phần
  - + Số lượt xem
- Phân trang: Nút Previous/Next hoặc số trang
- Nút "Tạo công thức mới" (khi đã đăng nhập)



Hình 4.6: Giao diện Tìm kiếm (Search)



Hình 4.6.1: Giao diện Tìm kiếm gợi ý công thức (Search)



Hình 4.6.2: Giao diện Tìm kiếm khi không có công thức phù hợp (Search)

Mô tả: Trang kết quả tìm kiếm công thức. Các thành phần chính bao gồm:

- Thanh tìm kiếm với từ khóa đã nhập
- Số lượng kết quả tìm được
- Danh sách công thức phù hợp (dạng lưới hoặc danh sách)
- Thông báo "Không tìm thấy kết quả" khi không có công thức phù hợp
- Gợi ý tìm kiếm khi không có kết quả

4.5.3 Giao diện quản lý công thức

CookShare

Chia sẻ công thức

Trang chủ

Công thức

Đã lưu

Tạo mới

Thanh Duy

Viết món mới

Bạn đã đăng hình món mình nấu ở đây chưa?  
Chia sẻ với mọi người thành phẩm của bạn nào!

Tên món:

Món canh bí ngon nhất nhà mình

Phân loại

Loại món

Ẩm thực

Khác

Khác

Tags

#Gia đình

#Dễ làm

#Cho trẻ em

#Nhanh gọn

#Tiệc tùng

#Ngày lễ

#Healthy

#Chay

#ít calo

#Không gluten

Hình 4.7: Giao diện Tạo công thức (Create Recipe)

#Tiệc tùng

#Ngày lễ

#Healthy

#Chay

#ít calo

#Không gluten

Thêm tag mới...

+ Thêm

Nguyên Liệu

Khẩu phần

2

người

250g bột

...

250g bột

...

250g bột

...

250g bột

...

250g bột

...

+ Nguyên liệu

Các bước

Thời gian nấu

1 tiếng 30 phút

1

Trộn bột và nước đến khi đặc lại

...

Hình 4.7.1: Giao diện Tạo công thức Khu vực nguyên liệu(Create Recipe)

Các bước

Thời gian nấu

1 tiếng 30 phút

1

Trộn bột và nước đến khi đặc lại

...

2

Trộn bột và nước đến khi đặc lại

...

3

Trộn bột và nước đến khi đặc lại

...

+ Bước làm

Hủy

Đăng bài

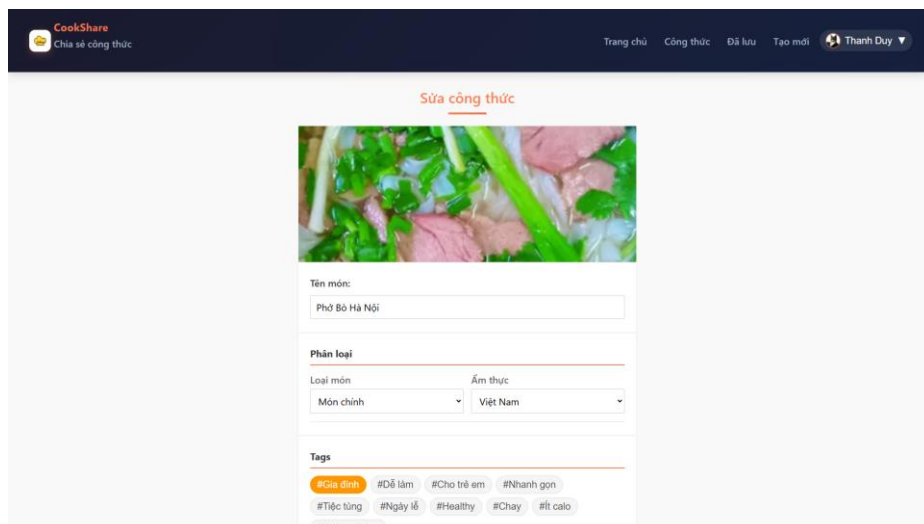
Hình 4.7.2: Giao diện Tạo công thức Khu vực các bước nấu(Create Recipe)

Nguyễn Thanh Duy

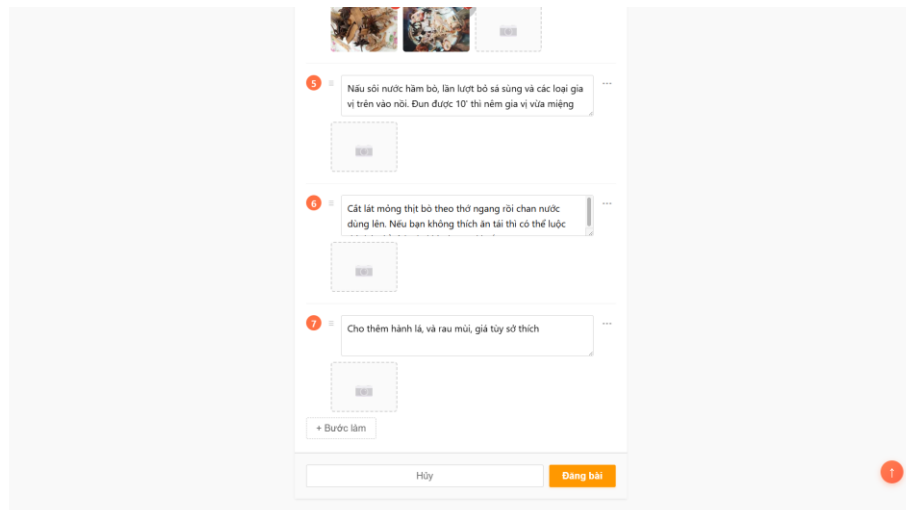
29

Mô tả: Form tạo công thức mới với đầy đủ thông tin. Các thành phần chính bao gồm:

- Trường tiêu đề công thức: Bắt buộc, tối đa 255 ký tự
- Upload ảnh đại diện: Kéo thả hoặc click để chọn file, preview ảnh
- Trường khẩu phần ăn: Số người ăn
- Trường thời gian nấu: Phút hoặc giờ
- Khu vực nguyên liệu:
  - + Danh sách nguyên liệu động (thêm/xóa dòng)
  - + Mỗi dòng: tên nguyên liệu, số lượng, đơn vị
- Khu vực các bước nấu:
  - + Danh sách bước động (thêm/xóa/sắp xếp)
  - + Mỗi bước: số thứ tự, mô tả chi tiết, upload ảnh minh họa
- Nút "Đăng công thức": Lưu và publish
- Nút "Hủy": Quay lại trang trước
- Validation hiển thị lỗi cho từng trường



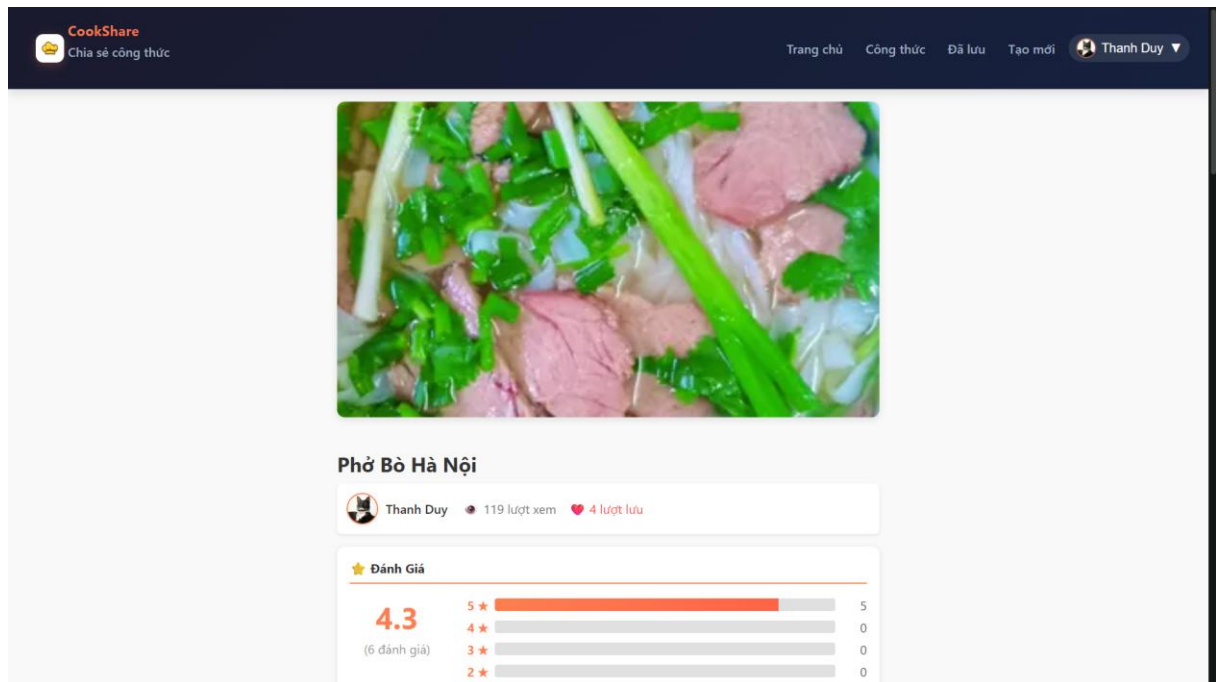
Hình 4.8: Giao diện Chỉnh sửa công thức (Edit Recipe)



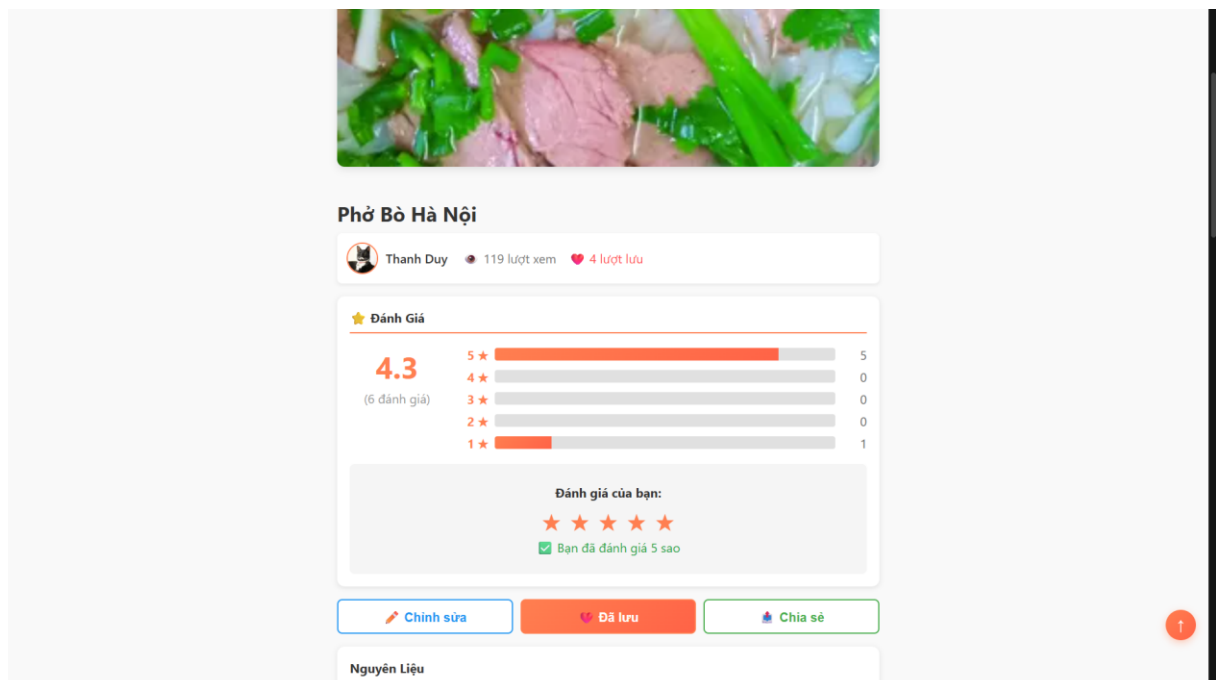
Hình 4.8.1: Giao diện Chỉnh sửa công thức: Nút (Edit Recipe)

Mô tả: Form chỉnh sửa công thức đã tạo, tương tự Create Recipe nhưng có dữ liệu sẵn.  
Các thành phần chính bao gồm:

- Tất cả trường như Create Recipe với dữ liệu hiện tại
- Preview ảnh hiện tại với option thay đổi
- Nút "Đăng bài": Lưu thay đổi
- Nút "Hủy": Hủy chỉnh sửa
- Chỉ hiển thị cho chủ sở hữu công thức

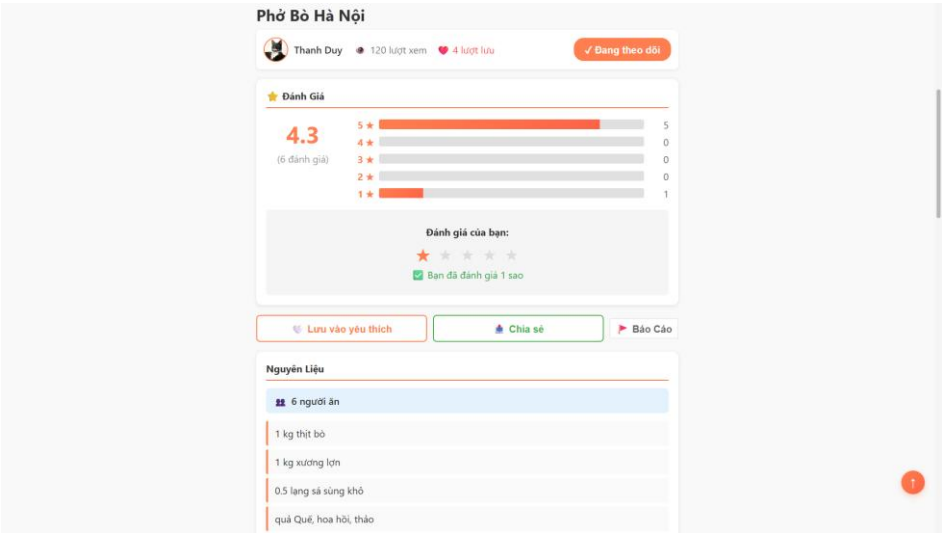


Hình 4.9: Giao diện Chi tiết công thức (Recipe Detail)

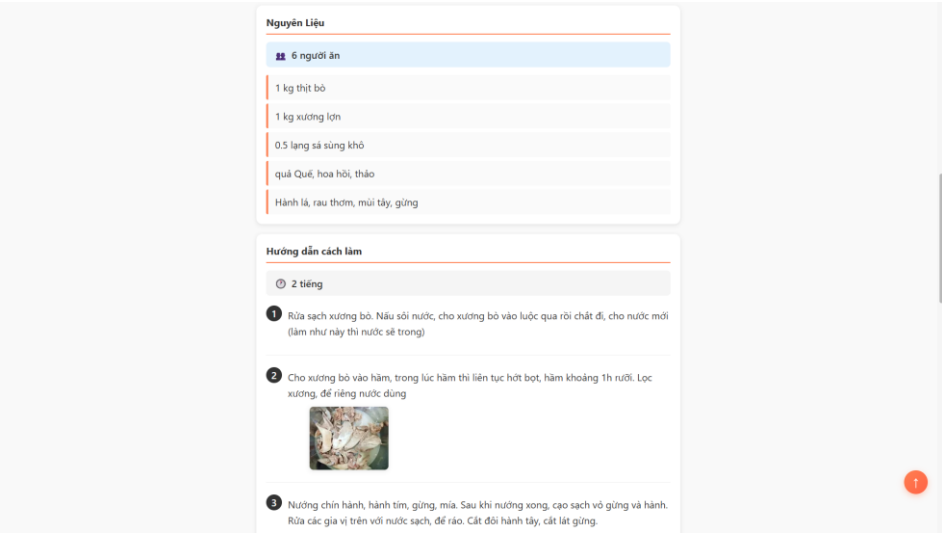


Hình 4.9.1: Giao diện Chi tiết công thức khi là tác giả (Recipe Detail)

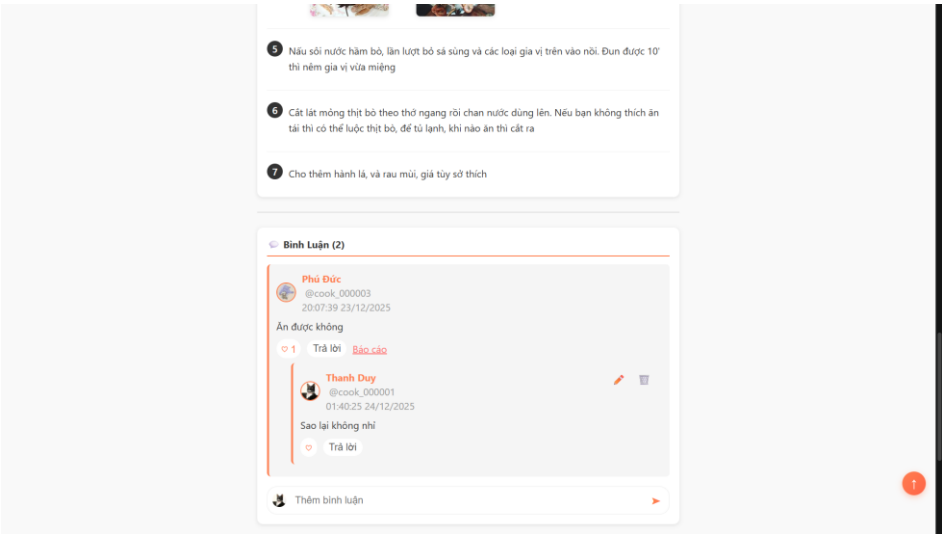




Hình 4.9.2: Giao diện Chi tiết công thức khi không là tác giả (Recipe Detail)



Hình 4.9.3: Giao diện Chi tiết công thức Các bước nấu (Recipe Detail)

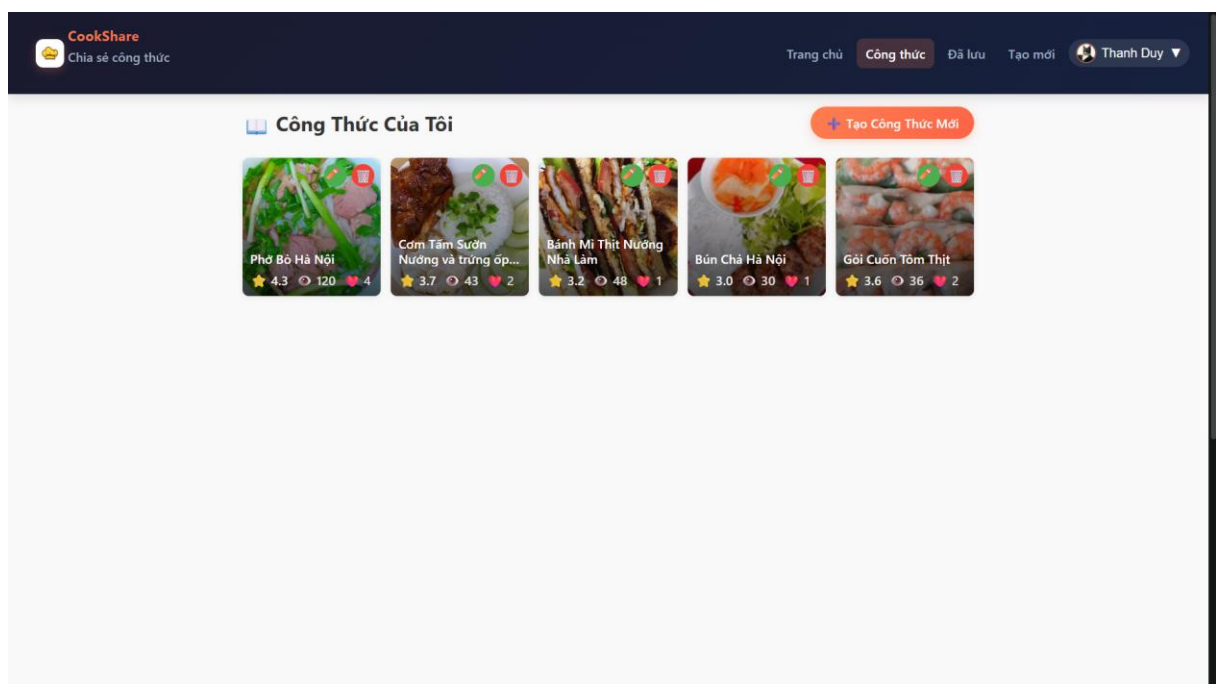


Hình 4.9.4: Giao diện Chi tiết công thức khu vực bình luận (Recipe Detail)



Mô tả: Trang hiển thị đầy đủ thông tin một công thức. Các thành phần chính bao gồm:

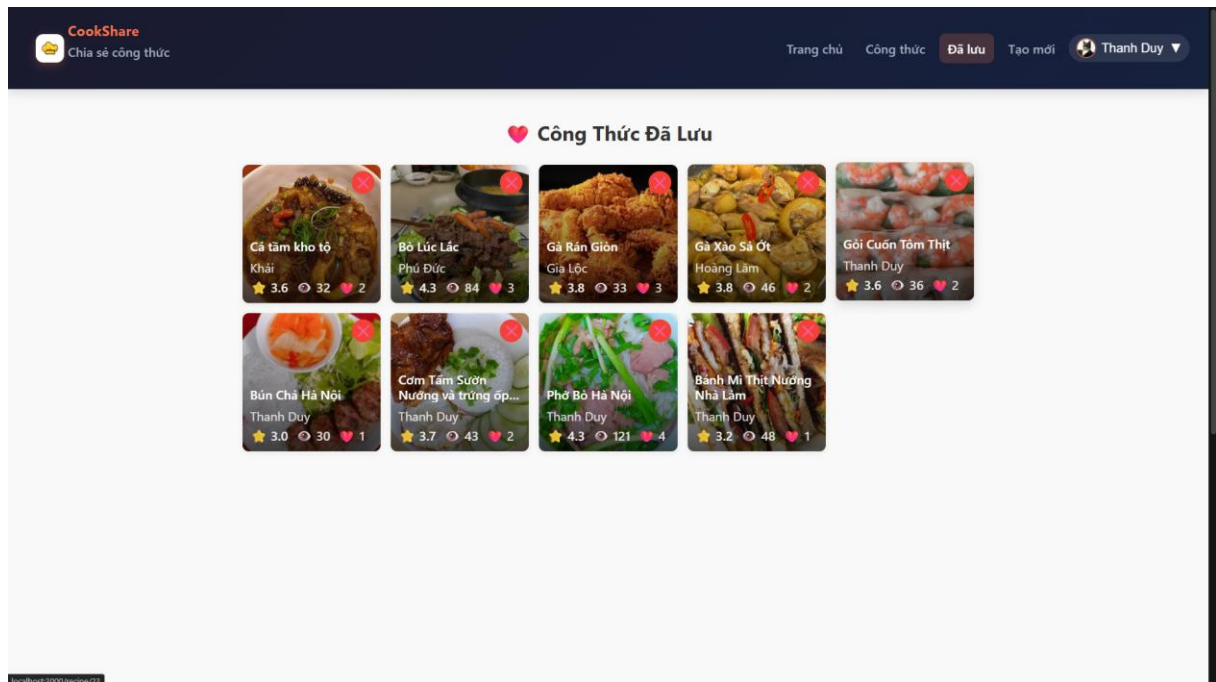
- Header: Tiêu đề, tác giả (link profile), ngày đăng, lượt xem
- Ảnh đại diện lớn (click để xem lightbox)
- Thông tin: Khẩu phần, thời gian nấu
- Đánh giá trung bình và số lượt đánh giá
- Nút tương tác:
  - + Nút yêu thích (heart icon, toggle)
  - + Nút báo cáo (flag icon)
  - + Nút chia sẻ
  - + Nút chỉnh sửa/xóa (nếu là chủ sở hữu)
- Các bước nấu: Số thứ tự, mô tả, ảnh minh họa từng bước
- Khu vực đánh giá: Chọn số sao (1-5), chỉ được đánh giá 1 lần
- Khu vực bình luận:
  - + Form nhập bình luận mới
  - + Danh sách bình luận với avatar, tên, thời gian
  - + Nút reply để trả lời bình luận (nested comments)
- Nút like bình luận với số lượt like
- Nút xóa (cho chủ bình luận hoặc admin/mod)
- Nút báo cáo bình luận



Hình 4.10: Giao diện Công thức của tôi (My Recipes)

Mô tả: Trang quản lý các công thức do người dùng tạo. Các thành phần chính bao gồm:

- Tiêu đề "Công thức của tôi"
- Thống kê: Tổng số công thức, tổng lượt xem, tổng đánh giá
- Danh sách công thức dạng lưới hoặc bảng
- Mỗi công thức hiển thị: Ảnh, tiêu đề, ngày tạo, lượt xem, rating
- Nút "Chỉnh sửa" và "Xóa" cho mỗi công thức
- Nút "Tạo công thức mới"
- Thông báo khi chưa có công thức nào

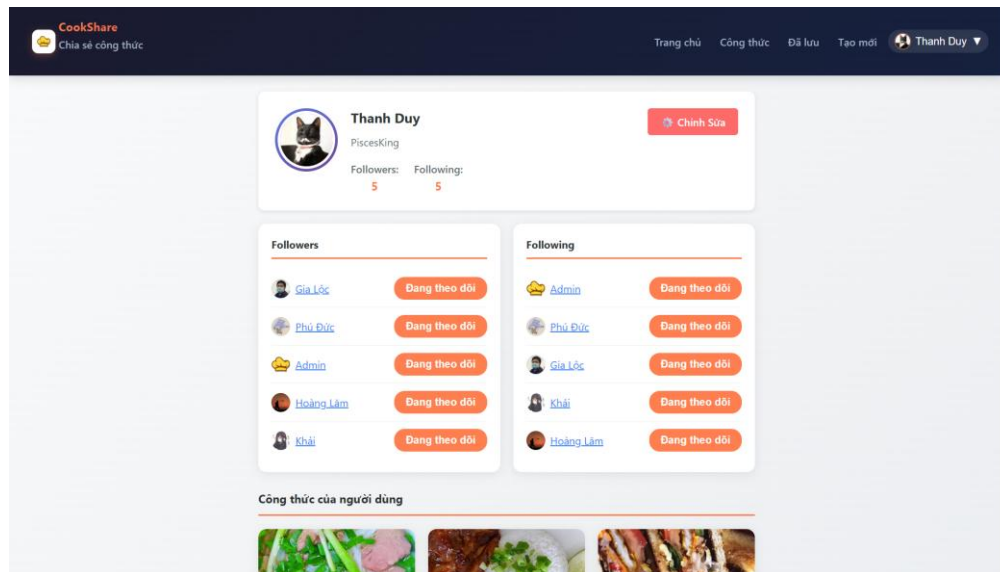


Hình 4.11: Giao diện Công thức yêu thích (Favorite Recipes)

Mô tả: Trang hiển thị danh sách công thức đã lưu yêu thích. bao gồm:

- Tiêu đề "Công thức yêu thích"
- Số lượng công thức đã lưu
- Danh sách công thức dạng lưới
- Mỗi card: Ảnh, tiêu đề, tác giả, rating
- Nút bỏ yêu thích (heart filled → outline)
- Thông báo khi chưa có công thức yêu thích

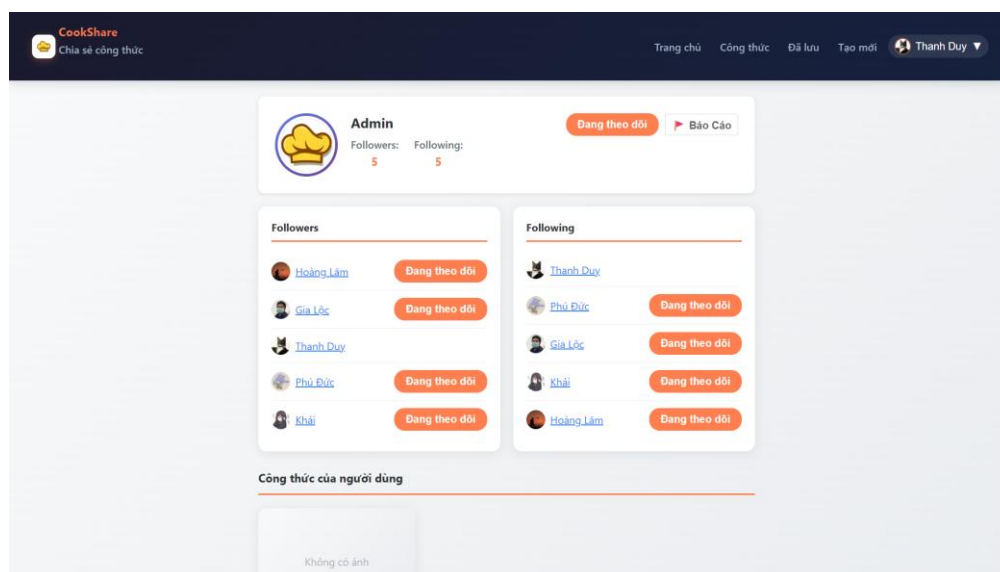
## 4.5.4 Giao diện hồ sơ người dùng



Hình 4.12: Giao diện Hồ sơ cá nhân (Profile)

Mô tả: Trang hồ sơ cá nhân của người dùng đang đăng nhập. bao gồm:

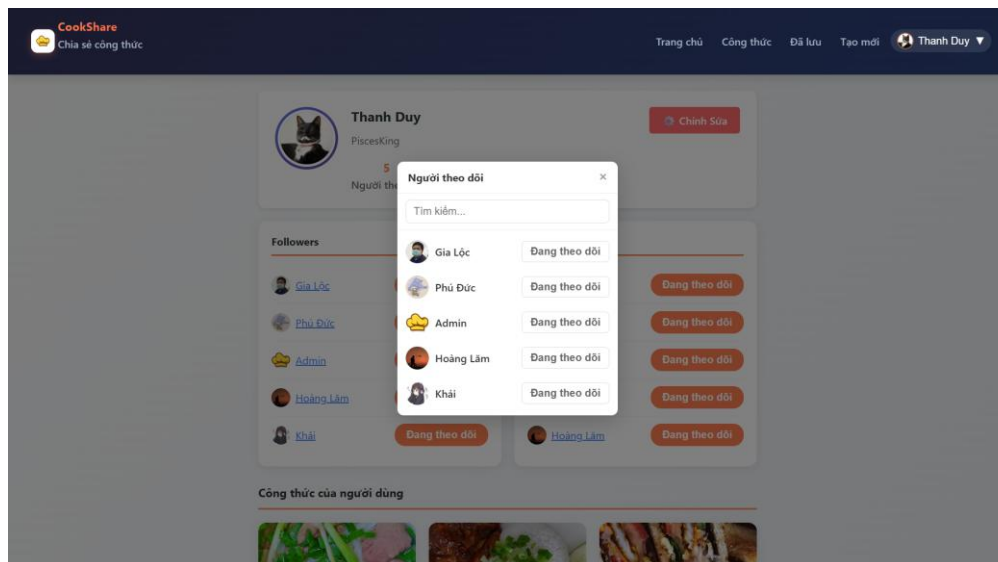
- Header profile:
  - + Avatar (click để thay đổi)
  - + Tên người dùng
  - + Bio/giới thiệu ngắn
  - + Số followers và following (click để xem danh sách)
  - + Ngày tham gia
- Danh sách công thức đã đăng
- Nút "Chỉnh sửa hồ sơ"



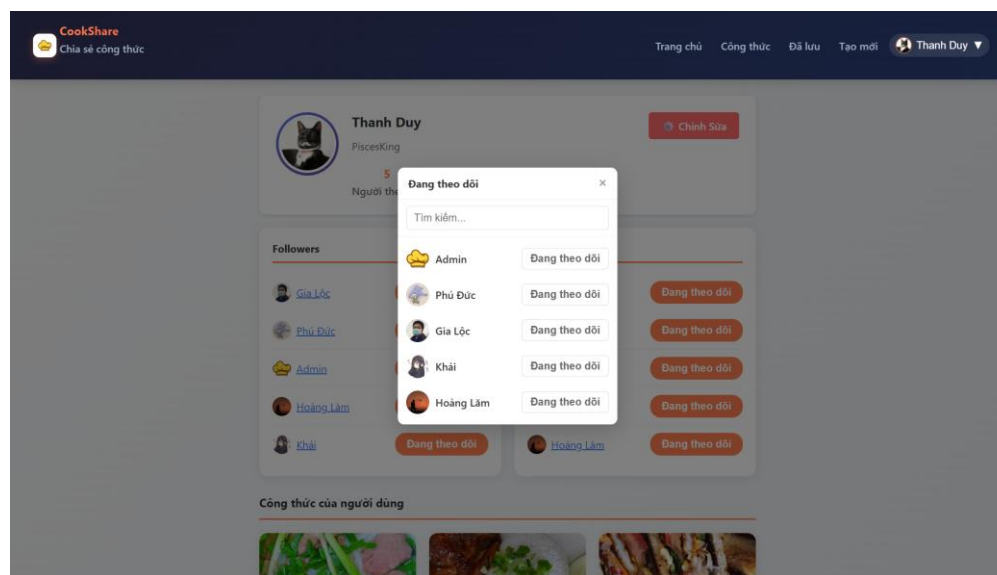
Hình 4.13: Giao diện Hồ sơ người dùng khác (User Profile)

Mô tả: Trang xem hồ sơ của người dùng khác. Các thành phần chính bao gồm:

- Header profile tương tự Profile nhưng không có nút chỉnh sửa
- Nút "Theo dõi" / "Đang theo dõi" (toggle)
- Nút "Báo cáo người dùng"
- Danh sách công thức công khai của người dùng
- Không hiển thị thông tin nhạy cảm (email, settings)



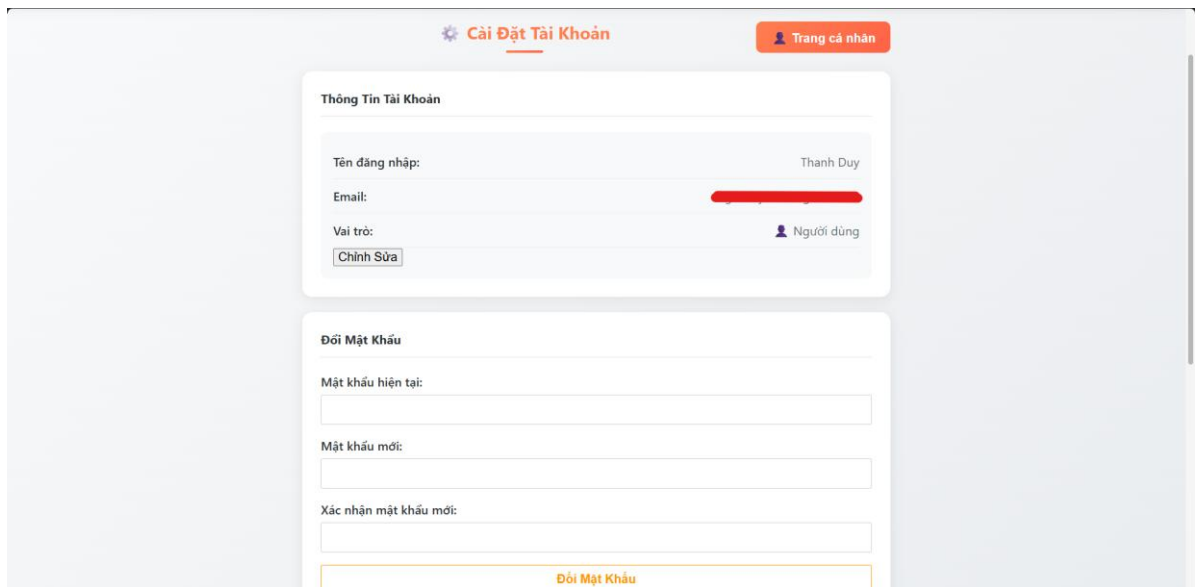
Hình 4.14: Giao diện Danh sách Followers



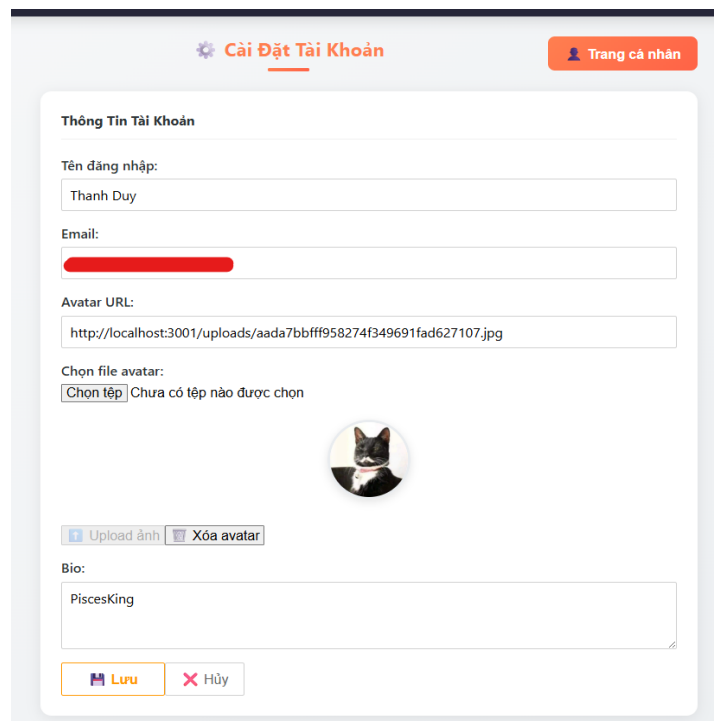
Hình 4.14.1: Giao diện Danh sách Following

Mô tả: Modal hoặc trang hiển thị danh sách người theo dõi/đang theo dõi. Các thành phần chính bao gồm:

- Tab chuyển đổi: Followers / Following
- Danh sách người dùng: Avatar, username, bio ngắn
- Nút "Theo dõi" / "Đang theo dõi" cho mỗi người
- Phân trang khi danh sách dài
- Thanh tìm kiếm trong danh sách



Hình 4.15: Giao diện Cài đặt tài khoản (Account Settings)

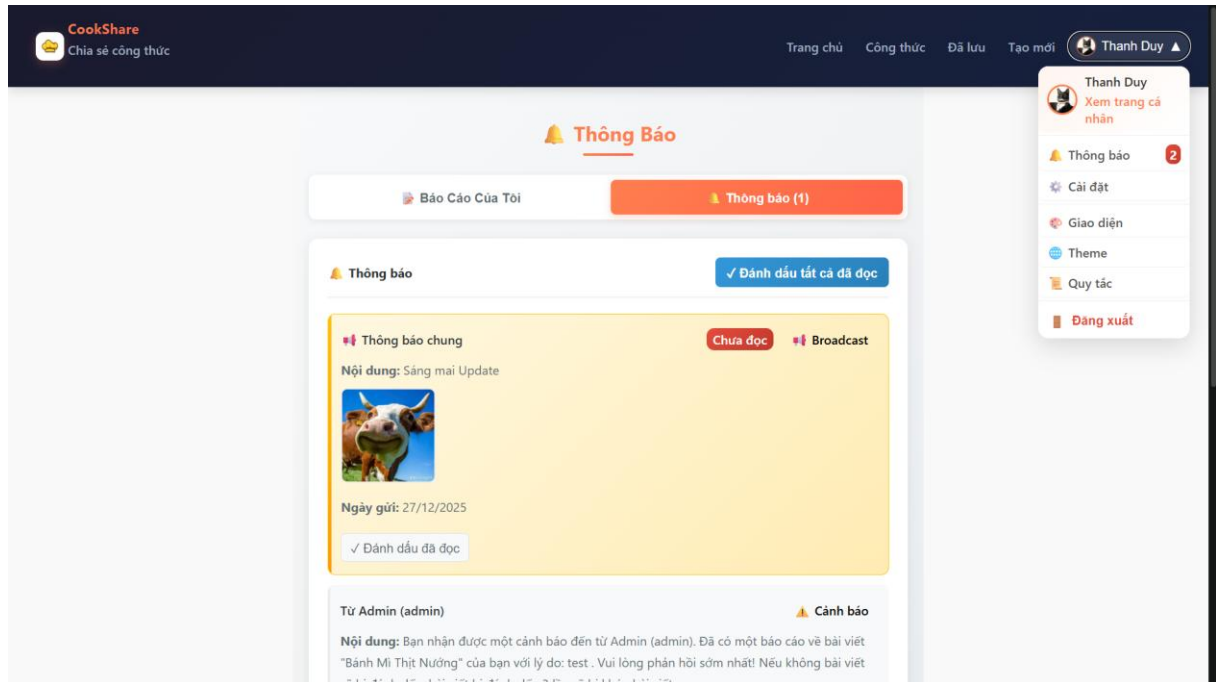


Hình 4.15.1: Giao diện Cài đặt tài khoản Section thông tin cá nhân (Account Settings)

Mô tả: Trang cài đặt và chỉnh sửa thông tin tài khoản. Các thành phần chính bao gồm:

- Section thông tin cá nhân:
  - + Thay đổi avatar (crop ảnh)
  - + Chỉnh sửa username
  - + Chỉnh sửa bio
- Section bảo mật:
  - + Đổi mật khẩu (nhập mật khẩu cũ + mới)
  - + Đổi email

## 4.5.5 Giao diện thông báo

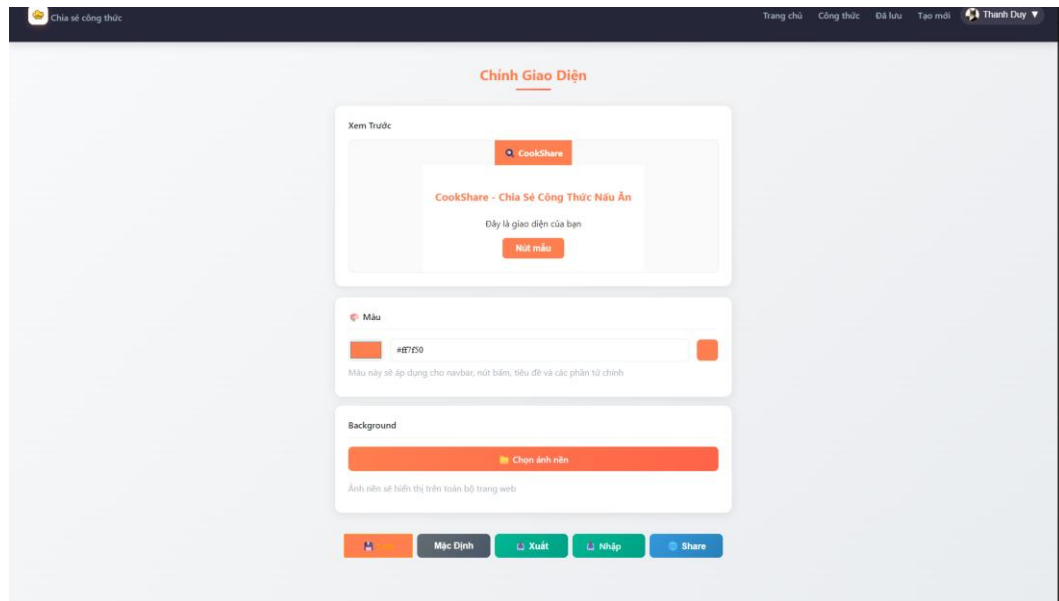


Hình 4.16: Giao diện Thông báo của User (Notifications)

Mô tả: Trang quản lý tất cả thông báo của người dùng. Các thành phần chính bao gồm:

- Tab phân loại: Tất cả, Chưa đọc, Cá nhân, Hệ thống (Broadcast)
- Danh sách thông báo:
  - + Icon loại thông báo
  - + Nội dung thông báo
  - + Người gửi (admin/moderator)
  - + Thời gian nhận
  - + Trạng thái đọc/chưa đọc (highlight)
  - + Ảnh đính kèm (nếu có)
- Nút "Đánh dấu đã đọc" cho từng thông báo
- Nút "Đánh dấu tất cả đã đọc"
- Nút "Trả lời" cho thông báo cá nhân
- Form trả lời thông báo (textarea + nút gửi)
- Badge đếm số thông báo chưa đọc

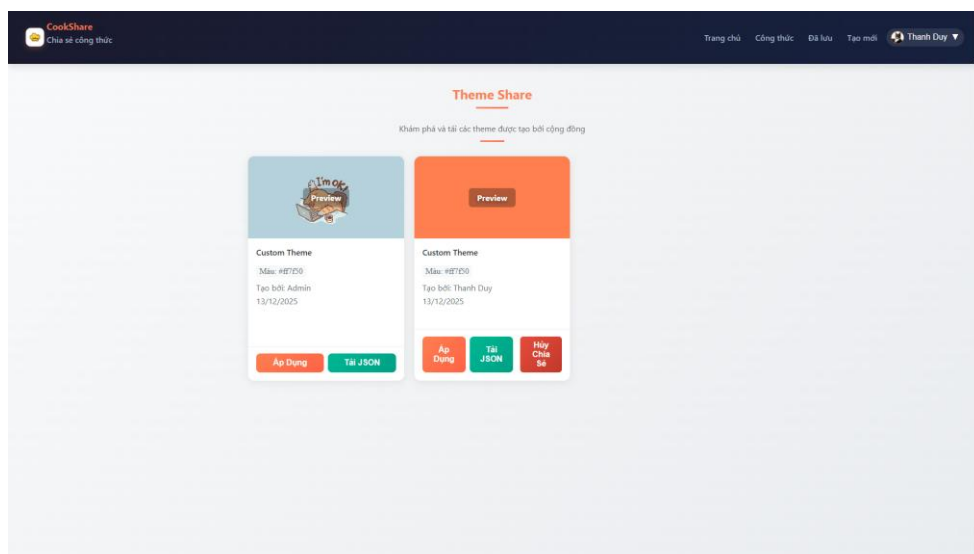
## 4.5.6 Giao diện tùy chỉnh giao diện



Hình 4.17: Giao diện Tùy chỉnh Theme (Theme Customization)

Mô tả: Trang cho phép người dùng tùy chỉnh giao diện cá nhân. bao gồm:

- Color Picker: Chọn màu chủ đạo (primary color)
- Upload hình nền: Chọn ảnh làm background
- Preview realtime: Xem trước thay đổi ngay lập tức
- Nút "Reset về mặc định"
- Nút "Luu theme"
- Section chia sẻ theme:
  - + Trường nhập tên theme
  - + Nút "Chia sẻ lên Marketplace"
- Danh sách theme đã chia sẻ của mình
- Nút xóa theme đã chia sẻ

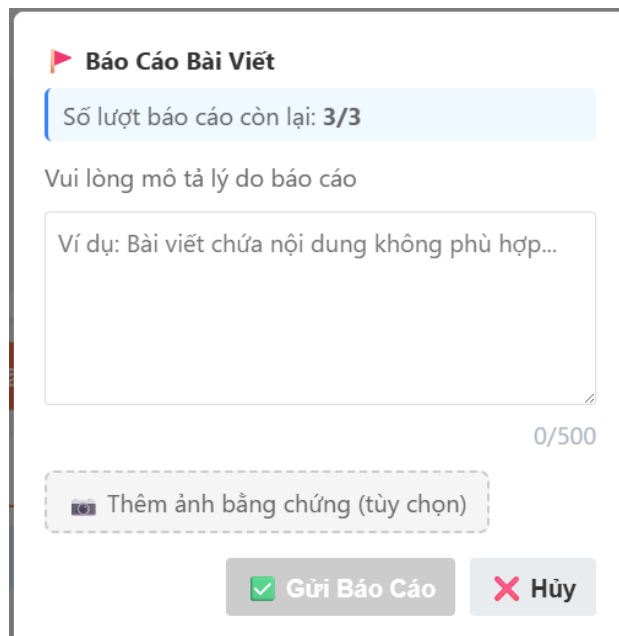


Hình 4.18: Giao diện Theme Marketplace

Mô tả: Trang duyệt và tải theme từ cộng đồng. Các thành phần chính bao gồm:

- Tiêu đề "Theme Marketplace"
- Lưới các theme công khai:
  - + Preview màu sắc và hình nền
  - + Tên theme
  - + Tên tác giả (link profile)
  - + Ngày chia sẻ
- Nút "Áp dụng theme" cho mỗi theme
- Confirm dialog trước khi áp dụng
- Thông báo áp dụng thành công

### 4.5.7 Giao diện báo cáo vi phạm



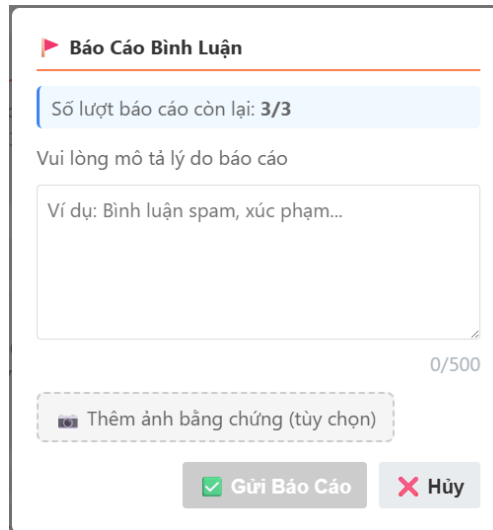
Hình 4.19: Modal Báo cáo bài viết

Mô tả: Modal popup khi người dùng báo cáo công thức vi phạm. bao gồm:

- Tiêu đề "Báo cáo bài viết"
- Thông tin bài viết đang báo cáo
- Dropdown chọn lý do báo cáo:
  - + Nội dung không phù hợp
  - + Spam/Quảng cáo
  - + Vi phạm bản quyền
  - + Thông tin sai lệch
  - + Khác



- Textarea mô tả chi tiết
- Upload ảnh bằng chứng (tùy chọn)
- Hiện thị quota còn lại (VD: "Còn 2/3 lượt báo cáo")
- Nút "Gửi báo cáo"
- Nút "Hủy"

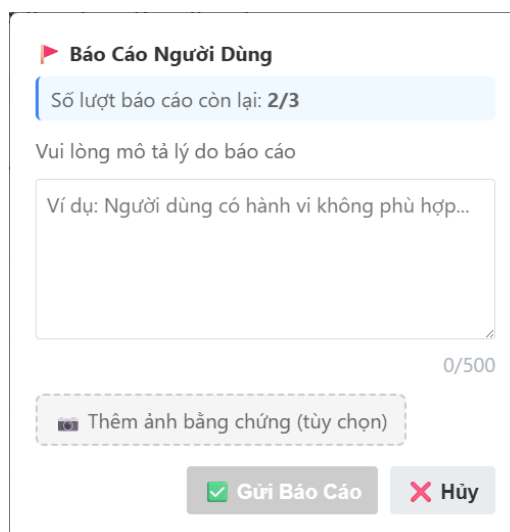


The modal titled "Báo Cáo Bình Luận" (Report Comment) features a red flag icon. It displays a quota status of "Số lượt báo cáo còn lại: 3/3" (Remaining reports: 3/3). Below this, it prompts the user to "Vui lòng mô tả lý do báo cáo" (Please describe the reason for reporting) with a text area containing the example "Ví dụ: Bình luận spam, xúc phạm..." (Example: Spam comment, insult...). A character count "0/500" is shown at the bottom right of the text area. There is a dashed box for "Thêm ảnh bằng chứng (tùy chọn)" (Add evidence photo (optional)). At the bottom, there are two buttons: "Gửi Báo Cáo" (Send Report) with a green checkmark icon and "Hủy" (Cancel) with a red X icon.

Hình 4.20: Modal Báo cáo bình luận

Mô tả: Modal popup khi báo cáo bình luận vi phạm. Các thành phần tương tự báo cáo bài viết, bao gồm:

- Nội dung bình luận đang báo cáo
- Lý do và mô tả chi tiết
- Upload bằng chứng
- Quota còn lại



The modal titled "Báo Cáo Người Dùng" (Report User) features a red flag icon. It displays a quota status of "Số lượt báo cáo còn lại: 2/3" (Remaining reports: 2/3). Below this, it prompts the user to "Vui lòng mô tả lý do báo cáo" (Please describe the reason for reporting) with a text area containing the example "Ví dụ: Người dùng có hành vi không phù hợp..." (Example: User has inappropriate behavior...). A character count "0/500" is shown at the bottom right of the text area. There is a dashed box for "Thêm ảnh bằng chứng (tùy chọn)" (Add evidence photo (optional)). At the bottom, there are two buttons: "Gửi Báo Cáo" (Send Report) with a green checkmark icon and "Hủy" (Cancel) with a red X icon.

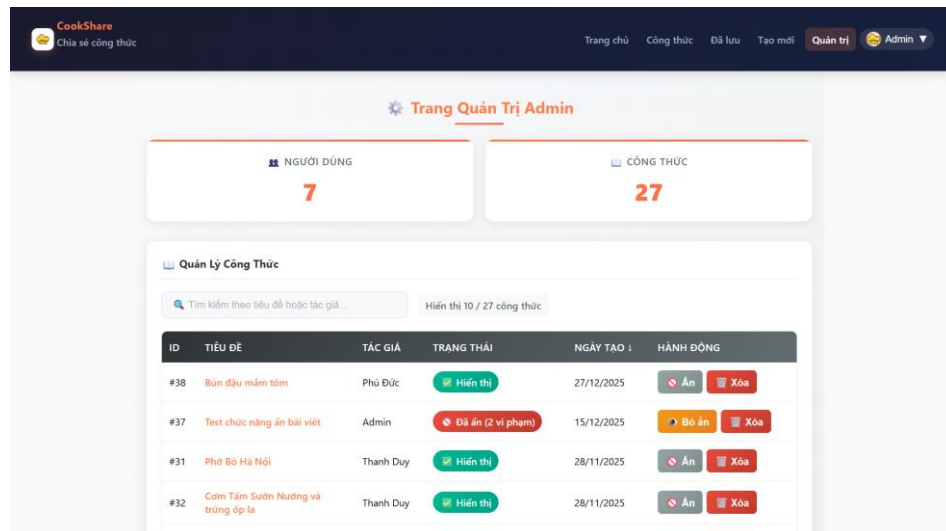
Hình 4.21: Modal Báo cáo người dùng

Mô tả: Modal popup khi báo cáo người dùng vi phạm. Các thành phần chính bao gồm:

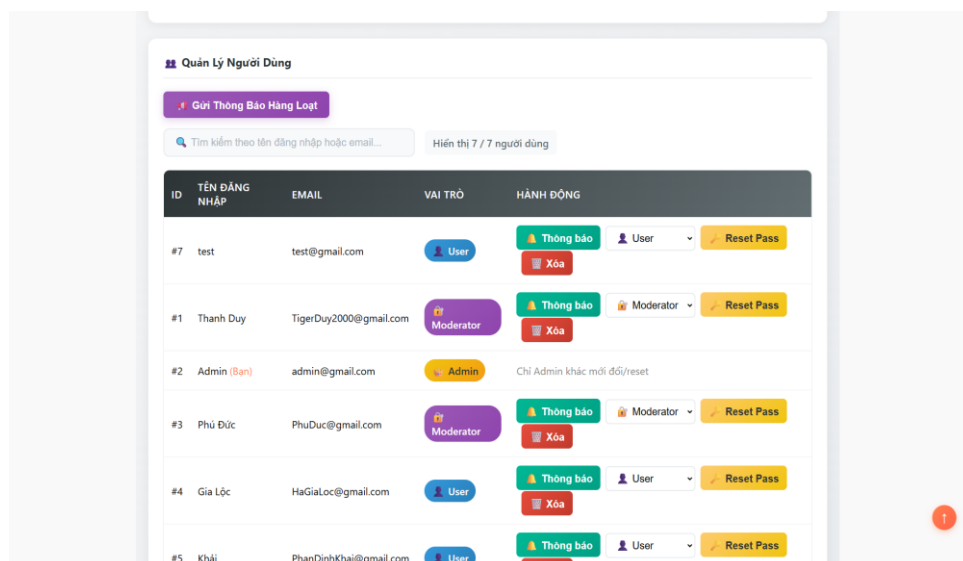
Thông tin người dùng đang báo cáo (avatar, username)

- Lý do báo cáo:
  - + Hành vi quấy rối
  - + Spam
  - + Mạo danh
  - + Nội dung không phù hợp
  - + Khác
- Mô tả chi tiết
- Upload bằng chứng
- Quota còn lại
- Không cho phép báo cáo admin/moderator

### 4.5.8 Giao diện quản trị (Admin/Moderator)



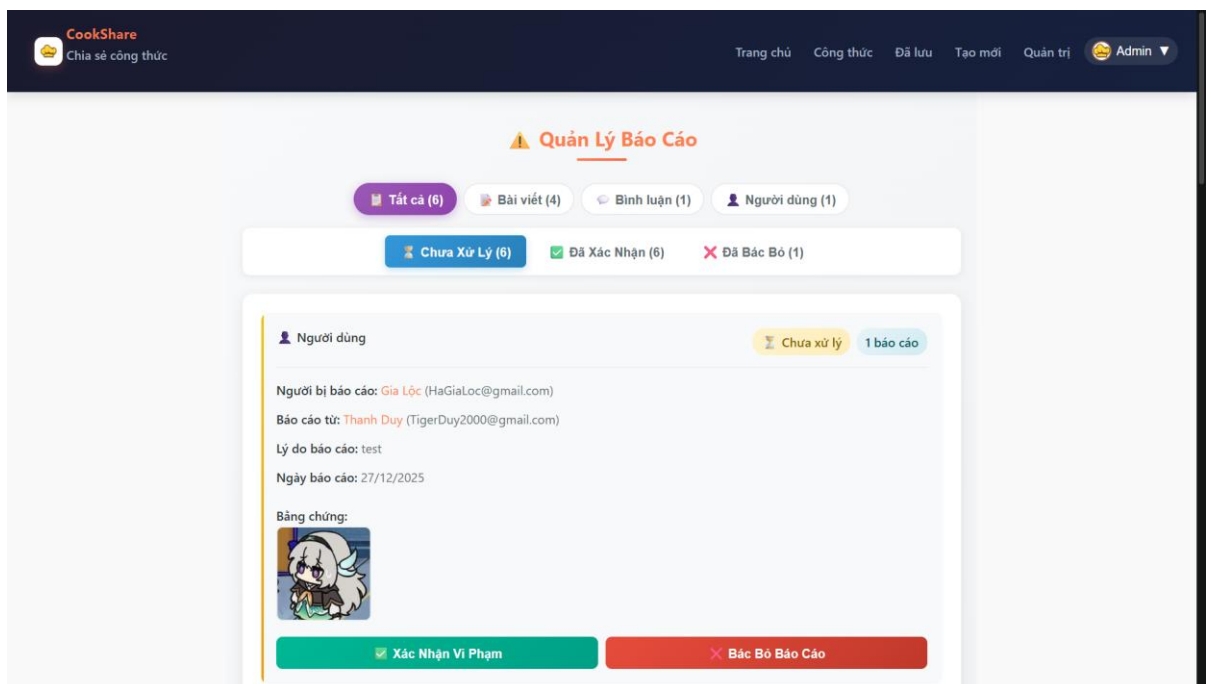
Hình 4.22: Giao diện Quản trị Quản Lý Công Thức (Admin Dashboard)



Hình 4.22.1: Giao diện Quản trị Quản Lý Người Dùng (Admin Dashboard)

Mô tả: Trang quản lý người dùng dành cho admin. Các thành phần chính bao gồm:

- Thống kê tổng quan: Tổng user, moderator, admin, user bị khóa
- Thanh tìm kiếm người dùng
- Bảng danh sách người dùng:
  - + ID
  - + Avatar
  - + Username
  - + Email
  - + Vai trò (user/moderator/admin)
  - + Ngày tạo
  - + Trạng thái (active/blocked)
- Dropdown đổi vai trò (chỉ admin)
- Nút "Xóa tài khoản" (chỉ admin, có confirm)
- Nút "Gửi thông báo" cho từng user
- Phân trang

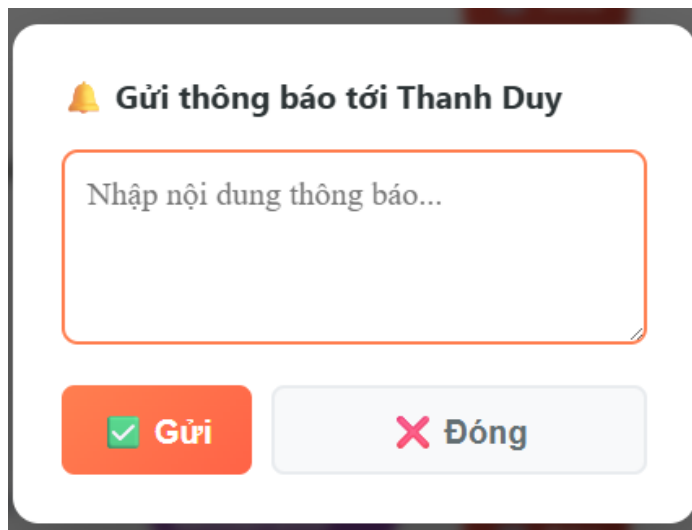


Hình 4.23: Giao diện Quản lý báo cáo (Admin Reports)

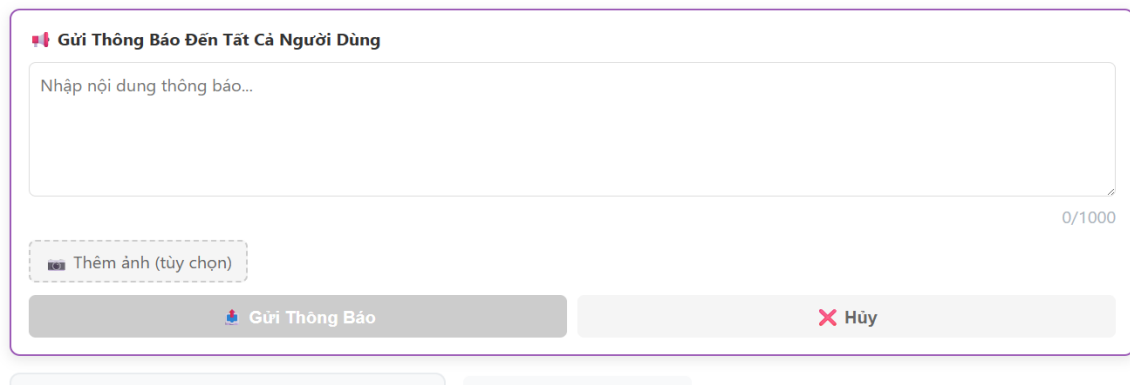
Mô tả: Trang xử lý các báo cáo vi phạm. Các thành phần chính bao gồm:

- Tab lọc theo trạng thái: Tất cả, Chờ xử lý, Đã chấp nhận, Đã từ chối
- Tab lọc theo loại: Bài viết, Bình luận, Người dùng
- Thống kê: Số báo cáo chờ xử lý
- Danh sách báo cáo:
  - + ID báo cáo

- + Loại (recipe/comment/user)
- + Người báo cáo
- + Đối tượng bị báo cáo (link xem chi tiết)
- + Lý do
- + Ảnh bằng chứng (click xem lightbox)
- + Ngày báo cáo
- + Trạng thái
- Chi tiết báo cáo khi click:
  - + Nội dung đầy đủ
  - + Link đến bài viết/bình luận/user bị báo cáo
- Nút "Chấp nhận": Ẩn bài viết/xóa bình luận/khóa user
- Nút "Từ chối": Nhập lý do từ chối
- Gửi email thông báo kết quả cho người báo cáo



Hình 4.24: Giao diện Gửi thông báo cá nhân (Send Notification)



Hình 4.24.1: Giao diện Gửi thông báo hàng loạt (Send Notification)

Mô tả: Form gửi thông báo đến người dùng. Các thành phần chính bao gồm:

- Tab chọn loại: Cá nhân / Hàng loạt (Broadcast)
- Gửi cá nhân:
  - + Dropdown/Search chọn người nhận
  - + Textarea nội dung thông báo
  - + Upload ảnh đính kèm (tùy chọn)
- Gửi hàng loạt:
  - + Textarea nội dung (gửi đến tất cả user)
  - + Upload ảnh đính kèm
  - + Cảnh báo: "Thông báo sẽ gửi đến tất cả người dùng"
- Nút "Gửi thông báo"
- Lịch sử thông báo đã gửi



Hình 4.25: Giao diện Ẩn/Hiện bài viết

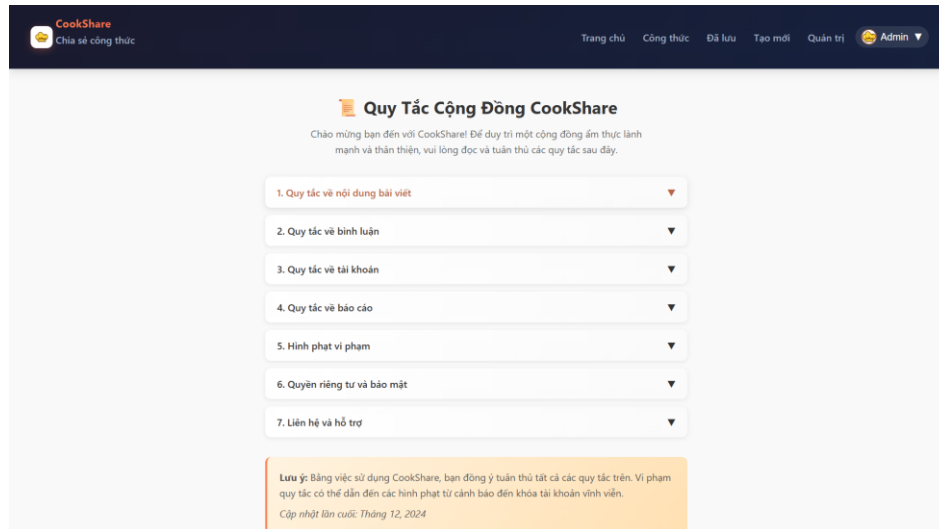
| ID  | TIÊU ĐỀ                    | TÁC GIẢ   | TRẠNG THÁI        | NGÀY TẠO   | HÀNH ĐỘNG |
|-----|----------------------------|-----------|-------------------|------------|-----------|
| #37 | Test chức năng ẩn bài viết | Admin     | Đã ẩn (2 vi phạm) | 15/12/2025 | Bỏ ẩn Xóa |
| #38 | Bún đậu mắm tôm            | Phú Đức   | Hiện thị          | 27/12/2025 | Ẩn Xóa    |
| #31 | Phở Bò Hà Nội              | Thanh Duy | Hiện thị          | 28/11/2025 | Ẩn Xóa    |

Hình 4.25.1: Giao diện Ẩn/Hiện bài viết

Mô tả: Chức năng ẩn bài viết vi phạm thủ công. Các thành phần chính bao gồm:

- Trong trang Recipe Detail (với quyền admin/mod):
- Nút "Ẩn bài viết" (nếu đang hiện)
- Nút "Bỏ ẩn bài viết" (nếu đang ẩn)
- Confirm dialog với lý do ẩn
- Thông báo trạng thái bài viết
- Bài viết bị ẩn hiển thị banner cảnh báo

## 4.5.9 Giao diện khác



Hình 4.26: Giao diện Quy định cộng đồng (Rules)

Mô tả: Trang hiển thị quy định và hướng dẫn sử dụng. Các thành phần chính bao gồm:

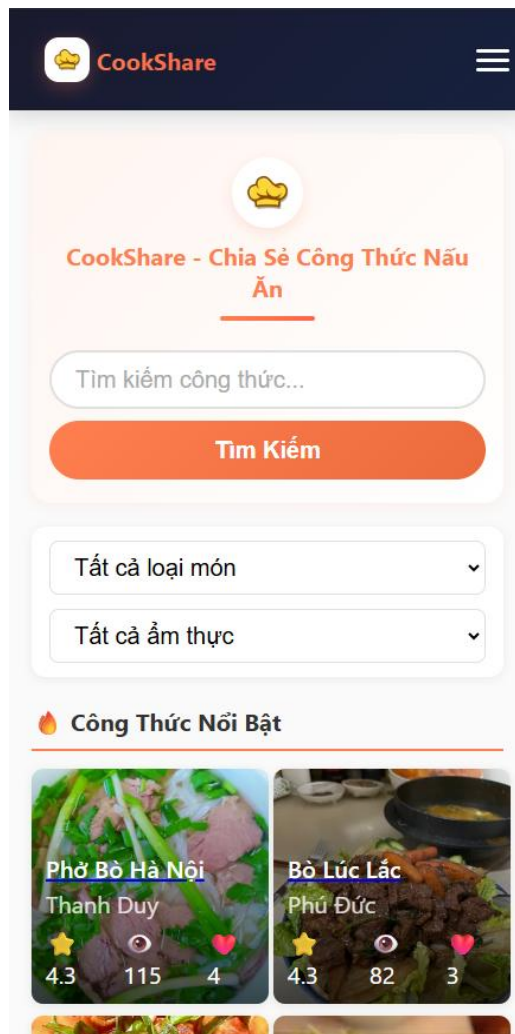
- Tiêu đề "Quy định cộng đồng CookShare"
- Các section quy định:
  - Quy định về nội dung
  - Quy định về bình luận
  - Quy định về báo cáo
  - Hình thức xử lý vi phạm
- Accordion/Collapse cho từng section
- Link liên hệ hỗ trợ



Hình 4.27: Footer

Mô tả: Phần chân trang cố định. Các thành phần chính bao gồm:

- Logo và slogan CookShare
- Link nhanh: Trang chủ, Quy định, Liên hệ
- Thông tin bản quyền
- Link mạng xã hội (nếu có)
- Responsive: Stack vertical trên mobile



Hình 4.28: Giao diện Responsive trên Mobile

Mô tả: Minh họa giao diện trên thiết bị di động. Các điểm nổi bật:

- Navbar thu gọn thành hamburger menu
- Recipe cards hiển thị 1 cột
- Form input full-width
- Nút và touch target đủ lớn (44px)
- Font size và spacing phù hợp
- Scroll smooth và native feel
- Nút "Scroll to Top" Hiện ra khi cuộn xuống quá 300px, nhấn vào sẽ cuộn mượt lên đầu trang
- Auto scroll to top khi chuyển trang Mỗi khi người dùng chuyển sang route khác (ví dụ từ Home sang Profile), trang sẽ tự động cuộn lên đầu

## **4.6 Đánh giá nhanh**

### **Đạt được:**

- Đầy đủ chức năng cốt lõi và nâng cao
- Giao diện thân thiện và responsive
- Bảo mật tốt (hash mật khẩu, JWT, phân quyền 3 cấp)
- Hệ thống báo cáo và kiểm duyệt hoàn chỉnh
- Tính năng xã hội phong phú (follow, like, nested comments)
- Tùy chỉnh giao diện cá nhân với marketplace
- Hiệu năng phù hợp quy mô đề tài

### **Hạn chế:**

- Chưa có cache (Redis)
- Chưa có realtime notification (WebSocket)
- Chưa triển khai CI/CD và bộ test tự động
- Chưa có gợi ý cá nhân hóa (ML)

## **4.7 Hướng phát triển**

- Thêm realtime notifications (WebSocket) cho bình luận/like/follow mới
- Bổ sung cache (Redis) cho danh sách và tìm kiếm
- Viết test tự động (unit/integration) và thiết lập CI/CD
- Pagination/virtualization cho bình luận nhiều cấp
- Gợi ý công thức cá nhân hóa (recommendation/ML)
- Tối ưu SEO và accessibility



## **CHƯƠNG 5: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN**

### **5.1 Kết luận**

- Hoàn thành website CookShare với đầy đủ chức năng cốt lõi và nâng cao: đăng ký/đăng nhập (JWT), quản lý công thức (tạo/sửa/xóa/tìm kiếm với ảnh từng bước), bình luận nested, đánh giá, yêu thích, theo dõi người dùng, và phân quyền user/moderator/admin
- Xây dựng hệ thống báo cáo vi phạm hoàn chỉnh với 3 loại báo cáo (bài viết/bình luận/người dùng), quota system, upload bằng chứng, tự động khóa tính năng khi vi phạm nhiều lần, gửi email thông báo kết quả
- Triển khai hệ thống thông báo đa dạng: thông báo cá nhân, thông báo hàng loạt (broadcast), trả lời thông báo, đếm số chưa đọc
- Tích hợp tính năng tùy chỉnh giao diện cá nhân với marketplace chia sẻ theme
- Áp dụng kiến trúc 3 tầng, RESTful API, bảo mật Bcrypt + JWT, CORS, và lưu trữ ảnh qua Cloudinary; giao diện React responsive vận hành ổn định
- Đảm bảo cơ sở dữ liệu quan hệ MySQL với ràng buộc toàn vẹn; trải nghiệm người dùng mạch lạc, quy trình nghiệp vụ rõ ràng
- Đóng góp: một giải pháp chia sẻ công thức nấu ăn có kiểm duyệt đa cấp (user/moderator/admin), dễ mở rộng, bảo mật tốt, tổ chức mã theo chuẩn (MVC, middleware, phân tầng)

### **5.2 Hạn chế**

- Chưa có pagination cho threads bình luận sâu; dữ liệu nhiều cấp sẽ chậm khi tải
- Notifications chưa realtime (chưa dùng WebSocket, mới ở mức polling/thủ công)
- Chưa dùng cache (Redis) cho danh sách/tìm kiếm; dễ bị chậm khi tải lớn
- Chưa có test tự động (unit/integration) và chưa thiết lập CI/CD
- Chưa có hệ thống gợi ý cá nhân hóa (recommendation/ML) và tối ưu SEO/accessibility

### **5.3 Hướng phát triển**

- Thêm realtime notifications (WebSocket) cho bình luận/like/follow/reply mới
- Bổ sung cache (Redis) cho danh sách và tìm kiếm để giảm độ trễ
- Viết test tự động (unit/integration) và thiết lập CI/CD để tăng độ tin cậy
- Pagination/virtualization cho bình luận nhiều cấp để tối ưu hiệu năng
- Gợi ý công thức cá nhân hóa (recommendation/ML) dựa trên hành vi người dùng
- Tối ưu SEO và accessibility
- Mở rộng marketplace theme với tính năng rating, download count
- Thêm tính năng chat/message giữa người dùng

## **DANH MỤC TÀI LIỆU THAM KHẢO**

- [1] paessler, "REST," [Online]. Available: <https://www.paessler.com/it-explained/rest>. [Accessed 08 12 2025].
- [2] "JWT Debugger," Introduction to JSON Web Tokens, [Online]. Available: <https://www.jwt.io/introductionwhat-is-json-web-token>. [Accessed 10 12 2025].
- [3] dominhhai, "Hai's Blog," [Node.js] Mã hoá mật khẩu với Bcrypt, 02 02 2016. [Online]. Available: <https://dominhhai.github.io/vi/2016/02/nodejs-ecrypt-password-with-bcrypt/>. [Accessed 14 12 2025].
- [4] "M mdn\_," Cross-Origin Resource Sharing (CORS), [Online]. Available: <https://developer.mozilla.org/en-US/docs/Web/HTTP/Guides/CORS>. [Accessed 09 12 2025].
- [5] M. O. Source, "React," React The library for web and native user interfaces, [Online]. Available: <https://react.dev/>. [Accessed 25 11 2025].
- [6] "Express5.2.1," Fast, unopinionated, minimalist web framework for Node.js, [Online]. Available: <https://expressjs.com>. [Accessed 08 12 2025].
- [7] O. Foundation, "nodejs.org," [Online]. Available: <https://nodejs.org/en/>. [Accessed 08 12 2025].
- [8] MySQL, "MySQL HeatWave," [Online]. Available: <https://dev.mysql.com/doc>. [Accessed 13 12 2025].
- [9] cloudinary, "Image and Video API Platform," [Online]. Available: <https://cloudinary.com/documentation>. [Accessed 22 12 2025].
- [10] OWASP, "OWASP Cheat Sheet Series," OWASP: OWASP Cheat Sheet Series (Authentication, Password Storage, REST Security), [Online]. Available: <https://cheatsheetseries.owasp.org/>. [Accessed 24 12 2025].

## PHỤ LỤC

### PHỤ LỤC A: Cấu hình môi trường

#### File .env (Backend)

*Bảng PLA.1: Môi trường phát triển*

| Thành phần | Phiên bản/Cấu hình         | Ghi chú                        |
|------------|----------------------------|--------------------------------|
| OS         | Windows 11 / macOS / Linux | Đa nền tảng                    |
| Node.js    | 16.20.0+                   | LTS                            |
| npm        | 8.19.0+                    | Package manager                |
| MySQL      | 8.0.33                     | Database server                |
| IDE        | VS Code 1.85+              | Với extension ESLint, Prettier |
| Browser    | Chrome 120+                | DevTools support               |
| Postman    | 10.0+                      | API testing                    |

#### File .env (Backend)

```
'''

PORT=3001
DB_HOST=localhost
DB_USER=root
DB_PASSWORD=yourpassword
DB_NAME=cookingdb
SECRET_KEY=your-jwt-secret-key-here
JWT_SECRET=your-jwt-secret-key-here
CLOUDINARY_CLOUD_NAME=your-cloud-name
CLOUDINARY_API_KEY=your-api-key
CLOUDINARY_API_SECRET=your-api-secret
EMAIL_USER=your-email@gmail.com
EMAIL_PASSWORD=your-app-password
'''
```

#### Yêu cầu hệ thống:

- Node.js: 16.x trở lên
- MySQL: 8.0 trở lên
- npm: 8.x trở lên
- Trình duyệt: Chrome 90+, Firefox 88+, Safari 14+

## **PHỤ LỤC B: Hướng dẫn cài đặt**

### **Bước 1: Clone repository**

```
```bash
git clone <repository-url>
cd DoAnChuyenNganh
```
```

### **Bước 2: Cài đặt Backend**

```
```bash
cd src/backend
npm install
Tạo file .env theo mẫu ở Phụ lục A
Tạo database MySQL
mysql -u root -p < ../database/database.sql
npm start
```
```

### **Bước 3: Cài đặt Frontend**

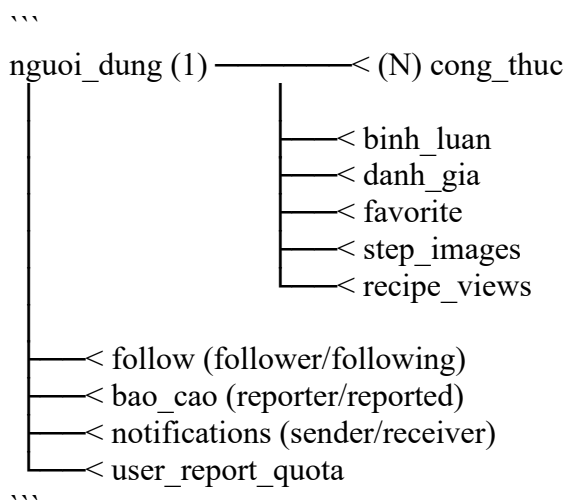
```
```bash
cd src/cookshare
npm install
npm start
```
```

### **Bước 4: Truy cập ứng dụng**

- Frontend: <http://localhost:3000>
- Backend API: <http://localhost:3001>

**PHỤ LỤC C: Schema cơ sở dữ liệu chi tiết****C.1. Tổng quan**

Hệ thống CookShare sử dụng MySQL 8.0 với 14 bảng chính, được thiết kế theo mô hình quan hệ với các ràng buộc toàn vẹn dữ liệu.

**C.2. Sơ đồ quan hệ các bảng****C.3. Chi tiết các bảng****C.3.1. Bảng nguoi\_dung (Người dùng)**

```

'''sql
CREATE TABLE nguoi_dung (
 id INT PRIMARY KEY AUTO_INCREMENT,
 username VARCHAR(50) UNIQUE NOT NULL,
 email VARCHAR(100) UNIQUE NOT NULL,
 password VARCHAR(255) NOT NULL,
 role ENUM('user', 'admin', 'moderator') DEFAULT 'user',
 avatar_url VARCHAR(500),
 bio TEXT,
 -- Các trường khóa tính năng khi vi phạm
 is_posting_blocked BOOLEAN DEFAULT FALSE,
 posting_blocked_until DATETIME DEFAULT NULL,
 is_commenting_blocked BOOLEAN DEFAULT FALSE,
 commenting_blocked_until DATETIME DEFAULT NULL,
 is_reporting_blocked BOOLEAN DEFAULT FALSE,
 reporting_blocked_until DATETIME DEFAULT NULL,
 created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP
);'''

```

*Bảng PLC.1: Bảng người dùng*

| Cột        | Kiểu dữ liệu | Mô tả                                        |
|------------|--------------|----------------------------------------------|
| id         | INT          | Khóa chính (PRIMARY KEY), tự tăng            |
| username   | VARCHAR(50)  | Tên đăng nhập, duy nhất (UNIQUE)             |
| email      | VARCHAR(100) | Email, duy nhất (UNIQUE)                     |
| password   | VARCHAR(255) | Mật khẩu đã mã hóa bằng Bcrypt               |
| role       | ENUM         | Vai trò người dùng: user / moderator / admin |
| avatar_url | VARCHAR(500) | Đường dẫn ảnh đại diện                       |
| bio        | TEXT         | Tiểu sử cá nhân                              |

**C.3.2. Bảng công\_thuc (Công thức nấu ăn)**

```
```sql
```

```
CREATE TABLE cong_thuc (  
    id INT PRIMARY KEY AUTO_INCREMENT,  
    user_id INT NOT NULL,  
    title VARCHAR(255) NOT NULL,  
    ingredients LONGTEXT NOT NULL,  
    steps LONGTEXT NOT NULL,  
    image_url VARCHAR(500),  
    servings VARCHAR(100),  
    cook_time VARCHAR(100),  
    views INT DEFAULT 0,  
    violation_count INT DEFAULT 0,  
    is_hidden BOOLEAN DEFAULT FALSE,  
    hidden_at DATETIME DEFAULT NULL,  
    created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP,  
    FOREIGN KEY (user_id) REFERENCES nguoi_dung(id) ON DELETE CASCADE  
);``
```

Bảng PLC.2: Bảng công_thuc

Cột	Kiểu dữ liệu	Mô tả
id	INT	Khóa chính (PRIMARY KEY), tự tăng
user_id	INT	Khóa ngoại (FK) → nguoi_dung(id)
title	VARCHAR(255)	Tiêu đề công thức
ingredients	LONGTEXT	Danh sách nguyên liệu (lưu dạng JSON)

Cột	Kiểu dữ liệu	Mô tả
steps	LONGTEXT	Các bước thực hiện (lưu dạng JSON)
image_url	VARCHAR(500)	Đường dẫn ảnh đại diện công thức
servings	VARCHAR(100)	Khẩu phần ăn
cook_time	VARCHAR(100)	Thời gian nấu
views	INT	Số lượt xem

C.3.3. Bảng binh_luan (Bình luận)

```
```sql
```

```
CREATE TABLE binh_luan (
 id INT PRIMARY KEY AUTO_INCREMENT,
 recipe_id INT NOT NULL,
 user_id INT NOT NULL,
 comment TEXT NOT NULL,
 created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
 FOREIGN KEY (recipe_id) REFERENCES cong_thuc(id) ON DELETE CASCADE,
 FOREIGN KEY (user_id) REFERENCES nguoi_dung(id) ON DELETE CASCADE
);``
```

### C.3.4. Bảng danh\_gia (Đánh giá)

```
```sql
```

```
CREATE TABLE danh_gia (  
    id INT PRIMARY KEY AUTO_INCREMENT,  
    recipe_id INT NOT NULL,  
    user_id INT NOT NULL,  
    rating INT CHECK (rating >= 1 AND rating <= 5),  
    created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP,  
    UNIQUE KEY unique_rating (recipe_id, user_id),  
    FOREIGN KEY (recipe_id) REFERENCES cong_thuc(id) ON DELETE CASCADE,  
    FOREIGN KEY (user_id) REFERENCES nguoi_dung(id) ON DELETE CASCADE  
);``
```

C.3.5. Bảng favorite (Yêu thích)

```
```sql
CREATE TABLE favorite (
 id INT PRIMARY KEY AUTO_INCREMENT,
 user_id INT NOT NULL,
 recipe_id INT NOT NULL,
 created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
 UNIQUE KEY unique_favorite (user_id, recipe_id),
 FOREIGN KEY (user_id) REFERENCES nguoi_dung(id) ON DELETE CASCADE,
 FOREIGN KEY (recipe_id) REFERENCES cong_thuc(id) ON DELETE CASCADE
);```
```

### **C.3.6. Bảng follow (Theo dõi)**

```
```sql
CREATE TABLE follow (
    id INT PRIMARY KEY AUTO_INCREMENT,
    follower_id INT NOT NULL,
    following_id INT NOT NULL,
    created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
    UNIQUE KEY unique_follow (follower_id, following_id),
    FOREIGN KEY (follower_id) REFERENCES nguoi_dung(id) ON DELETE
CASCADE,
    FOREIGN KEY (following_id) REFERENCES nguoi_dung(id) ON DELETE
CASCADE
);```
```

C.3.7. Bảng step_images (Ảnh các bước)

```
```sql
CREATE TABLE step_images (
 id INT PRIMARY KEY AUTO_INCREMENT,
 recipe_id INT NOT NULL,
 step_index INT NOT NULL,
 image_url VARCHAR(500) NOT NULL,
 created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
 FOREIGN KEY (recipe_id) REFERENCES cong_thuc(id) ON DELETE CASCADE
);```
```



### **C.3.8. Bảng bao\_cao (Báo cáo vi phạm)**

```
```sql
CREATE TABLE bao_cao (
    id INT PRIMARY KEY AUTO_INCREMENT,
    user_id INT NOT NULL,
    reason TEXT NOT NULL,
    image_url VARCHAR(500) DEFAULT NULL,
    status ENUM('pending', 'accepted', 'rejected') DEFAULT 'pending',
    rejected_reason TEXT DEFAULT NULL,
    processed_by INT DEFAULT NULL,
    processed_at DATETIME DEFAULT NULL,
    target_type ENUM('recipe', 'comment', 'user') DEFAULT 'recipe',
    recipe_id INT DEFAULT NULL,
    comment_id INT DEFAULT NULL,
    reported_user_id INT DEFAULT NULL,
    created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
    updated_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP ON UPDATE
    CURRENT_TIMESTAMP,
    FOREIGN KEY (user_id) REFERENCES nguoi_dung(id) ON DELETE CASCADE,
    FOREIGN KEY (recipe_id) REFERENCES cong_thuc(id) ON DELETE CASCADE,
    FOREIGN KEY (comment_id) REFERENCES binh_luan(id) ON DELETE CASCADE,
    FOREIGN KEY (reported_user_id) REFERENCES nguoi_dung(id) ON DELETE
    CASCADE,
    FOREIGN KEY (processed_by) REFERENCES nguoi_dung(id) ON DELETE SET
    NULL
);```
```

C.3.9. Bảng notifications (Thông báo)

```
```sql
CREATE TABLE notifications (
 id INT PRIMARY KEY AUTO_INCREMENT,
 sender_id INT NOT NULL,
 receiver_id INT NOT NULL,
 sender_role VARCHAR(20) DEFAULT NULL,
 type VARCHAR(50) DEFAULT 'manual',
```

```
message TEXT NOT NULL,
image_url VARCHAR(500) DEFAULT NULL,
metadata JSON DEFAULT NULL,
is_read BOOLEAN DEFAULT FALSE,
created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
FOREIGN KEY (sender_id) REFERENCES nguoi_dung(id) ON DELETE CASCADE,
FOREIGN KEY (receiver_id) REFERENCES nguoi_dung(id) ON DELETE CASCADE
);``
```

### **C.3.10. Bảng user\_report\_quota (Quota báo cáo)**

```
``sql
CREATE TABLE user_report_quota (
 id INT PRIMARY KEY AUTO_INCREMENT,
 user_id INT NOT NULL,
 report_type ENUM('recipe', 'comment', 'user') NOT NULL,
 remaining_reports INT DEFAULT 3,
 last_reset DATETIME DEFAULT CURRENT_TIMESTAMP,
 UNIQUE KEY unique_user_report_type (user_id, report_type),
 FOREIGN KEY (user_id) REFERENCES nguoi_dung(id) ON DELETE CASCADE
);``
```

### **C.3.11. Bảng recipe\_views (Theo dõi lượt xem)**

```
``sql
CREATE TABLE recipe_views (
 id INT PRIMARY KEY AUTO_INCREMENT,
 recipe_id INT NOT NULL,
 client_ip VARCHAR(45) NOT NULL,
 user_agent VARCHAR(500),
 created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
 FOREIGN KEY (recipe_id) REFERENCES cong_thuc(id) ON DELETE CASCADE,
 INDEX idx_recipe_views_lookup (recipe_id, client_ip, user_agent, created_at)
);``
```

#### **C.4. Indexes tối ưu hiệu năng**

```
```sql
-- Indexes cho truy vấn thường xuyên
CREATE INDEX idx_user_id ON cong_thuc(user_id);
CREATE INDEX idx_recipe_id ON binh_luan(recipe_id);
CREATE INDEX idx_recipe_id_favorite ON favorite(recipe_id);
CREATE INDEX idx_recipe_id_rating ON danh_gia(recipe_id);
CREATE INDEX idx_bao_cao_status ON bao_cao(status);
CREATE INDEX idx_bao_cao_target_type ON bao_cao(target_type);
CREATE INDEX idx_notifications_receiver ON notifications(receiver_id);
CREATE INDEX idx_notifications_type ON notifications(type);
```
```

#### **C.5. Ràng buộc toàn vẹn dữ liệu**

1. Khóa chính (Primary Key): Mỗi bảng có cột `id` tự tăng làm khóa chính
2. Khóa ngoại (Foreign Key): Đảm bảo tính toàn vẹn tham chiếu với `ON DELETE CASCADE`
3. Unique Constraint: Ngăn chặn dữ liệu trùng lặp (email, username, đánh giá, yêu thích)
4. Check Constraint: Giới hạn giá trị rating từ 1-5
5. Default Values: Giá trị mặc định cho các trường như role, status, created\_at

### **PHỤ LỤC D: API Documentation chi tiết**

#### **D.1 Authentication Endpoints**

POST /auth/register

- Body: **{ username, email, password, confirmPassword }**
- Response: **{ message: "Đăng ký thành công!" }**
- Errors: 400 (validation), 500 (server)

POST /auth/login

- Body: **{ email, password }**
- Response: **{ message, token, username, role, userId }**
- Errors: 400 (invalid credentials), 500 (server)

## D.2 Recipe Endpoints

GET /recipe/list

- Response: [ { id, title, ingredients, steps, image url, username, ... } ]

GET /recipe/search?q=keyword

- Query: q (search keyword)
- Response: [ { matching recipes } ]

POST /recipe/create

- Headers: Authorization: Bearer <token>
- Body: FormData with title, ingredients, steps, image
- Response: { message, recipeId }

PUT /recipe/update/:id

- Headers: Authorization: Bearer <token>
- Body: FormData
- Response: { message }
- Errors: 403 (not owner), 404 (not found)

DELETE /recipe/delete/:id

- Headers: Authorization: Bearer <token>
- Response: { message }
- Errors: 403 (not owner/admin), 404 (not found)

## PHỤ LỤC E: Kiểm thử

### E.1 Test cases chính

*Bảng PLG.1: Test cases chính*

| ID   | Chức năng     | Input                 | Expected Output      | Kết quả |
|------|---------------|-----------------------|----------------------|---------|
| TC01 | Đăng ký       | Valid email, password | 201, message success | ✓ Pass  |
| TC02 | Đăng ký       | Duplicate email       | 400, email exists    | ✓ Pass  |
| TC03 | Đăng nhập     | Valid credentials     | 200, JWT token       | ✓ Pass  |
| TC04 | Đăng nhập     | Invalid password      | 400, wrong password  | ✓ Pass  |
| TC05 | Tạo công thức | Valid data + token    | 201, recipe created  | ✓ Pass  |
| TC06 | Tạo công thức | No token              | 401, unauthorized    | ✓ Pass  |
| TC07 | Xóa công thức | Owner                 | 200, deleted         | ✓ Pass  |
| TC08 | Xóa công thức | Not owner             | 403, forbidden       | ✓ Pass  |
| TC09 | Đánh giá      | Valid rating 1-5      | 201, rated           | ✓ Pass  |

| <b>ID</b> | <b>Chức năng</b>      | <b>Input</b>         | <b>Expected Output</b> | <b>Kết quả</b> |
|-----------|-----------------------|----------------------|------------------------|----------------|
| TC10      | Đánh giá lần 2        | Duplicate rating     | 400, already rated     | ✓ Pass         |
| TC11      | Admin đổi role        | Admin token          | 200, role changed      | ✓ Pass         |
| TC12      | User đổi role         | User token           | 403, not admin         | ✓ Pass         |
| TC13      | Nested comment        | Reply với parent_id  | 201, reply created     | ✓ Pass         |
| TC14      | Nested invalid        | Parent không tồn tại | 400, invalid parent    | ✓ Pass         |
| TC15      | Like comment lần đầu  | Valid comment_id     | 200, {liked: true}     | ✓ Pass         |
| TC16      | Unlike comment        | Like lại             | 200, {liked: false}    | ✓ Pass         |
| TC17      | Xóa reply owner       | Owner xóa reply      | 200, deleted           | ✓ Pass         |
| TC18      | Xóa reply admin       | Admin xóa reply      | 200, deleted           | ✓ Pass         |
| TC19      | Lấy nested comments   | GET /recipe/comment  | 200, [nested array]    | ✓ Pass         |
| TC20      | Notification on reply | User A reply → B     | 201, noti created      | ✓ Pass         |