|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| TRƯỜNG ĐẠI HỌC GIAO THÔNG VẬN TẢI  **KHOA CÔNG NGHỆ THÔNG TIN**  A blue and yellow logo  AI-generated content may be incorrect.  **PROJECT 1**  ĐỀ TÀI  **ỨNG DỤNG CHIA SẺ CÁC CÔNG THỨC NẤU ĂN**   |  |  | | --- | --- | | Giảng viên hướng dẫn | : TS. Nguyễn Quốc Tuấn | | Nhóm | : 13 | | Lớp | : CNTT1 – K63 | | Danh sách thành viên | : Nguyễn Quang Lương – 221230911  Bạch Phương Anh – 221230729  Nguyễn Duy Chính – 221230754  Lê Văn Chung – 221230756  Nguyễn Tài Nhất - 221230948 |   **Hà Nội – 2025** |

# LỜI CẢM ƠN

Trước hết, nhóm em xin gửi lời cảm ơn chân thành và sâu sắc nhất đến quý thầy cô Khoa Công nghệ Thông tin đã tận tình giảng dạy, truyền đạt cho em những kiến thức quý báu trong suốt quá trình học tập và rèn luyện tại trường. Những kiến thức đó chính là nền tảng vững chắc giúp em có thể thực hiện và hoàn thiện đề tài này.

Đặc biệt, nhóm em xin bày tỏ lòng biết ơn sâu sắc đến thầy Nguyễn Quốc Tuấn, người đã trực tiếp hướng dẫn, chỉ bảo tận tình, định hướng và giúp đỡ nhóm em trong suốt quá trình nghiên cứu, xây dựng và hoàn thiện đề tài “**Ứng dụng Chia sẻ các Công thức Nấu ăn**”. Những góp ý và nhận xét quý báu của thầy đã giúp em hiểu rõ hơn về cách tư duy, phân tích vấn đề và phát triển giải pháp hiệu quả.

Mặc dù đã cố gắng hết sức, song do thời gian và kiến thức còn hạn chế, bài báo cáo không thể tránh khỏi những thiếu sót. Nhóm em rất mong nhận được sự đóng góp ý kiến từ quý thầy cô và các bạn để đề tài được hoàn thiện hơn.

# MỤC LỤC

[LỜI CẢM ƠN 2](#_Toc216702734)

[MỤC LỤC 3](#_Toc216702735)

[DANH MỤC CÁC TỪ VIẾT TẮT 7](#_Toc216702736)

[DANH MỤC BẢNG BIỂU 8](#_Toc216702737)

[DANH MỤC HÌNH ẢNH 9](#_Toc216702738)

[MỞ ĐẦU 10](#_Toc216702739)

[CHƯƠNG 1: PHÂN TÍCH YÊU CẦU 12](#_Toc216702740)

[1.1. Giới thiệu chung bài toán 12](#_Toc216702741)

[1.2. Thu nhập yêu cầu 12](#_Toc216702742)

[1.2.1. Phỏng vấn 12](#_Toc216702743)

[1.2.2. Phiếu hỏi 18](#_Toc216702744)

[1.2.3. Tổng hợp chức năng sau khi thực hiện khảo sát 25](#_Toc216702745)

[1.2.4. Khảo sát các phần mềm đã có 26](#_Toc216702746)

[1.3. So sánh các ứng dụng đã có trên thị trường 27](#_Toc216702747)

[1.4. Mô tả chức năng 28](#_Toc216702748)

[CHƯƠNG 2: Phân tích thiết kế hệ thống 30](#_Toc216702749)

[2.1. Biểu đồ ca sử dụng 30](#_Toc216702750)

[2.1.1. Biểu đồ Quản lý người dùng 30](#_Toc216702751)

[2.1.2. Biểu đồ Quản lí công thức nấu ăn 30](#_Toc216702752)

[2.1.3. Biểu đồ Tìm kiếm công thức 31](#_Toc216702753)

[2.1.4. Biểu đồ Gợi ý công thức 32](#_Toc216702754)

[2.1.5. Biểu đồ Tương tác người dùng 33](#_Toc216702755)

[2.2. Biểu đồ hoạt động 34](#_Toc216702756)

[2.2.1. Biểu đồ hoạt động Đăng nhập 34](#_Toc216702757)

[2.2.2. Biểu đồ hoạt động Tạo công thức 35](#_Toc216702758)

[2.2.3. Biểu đồ hoạt động Tìm kiếm công thức 36](#_Toc216702759)

[2.2.4. Biểu đồ hoạt động Gợi ý công thức theo nguyên liệu 37](#_Toc216702760)

[*2.2.5.* Biểu đồ hoạt động Theo dõi người dùng 38](#_Toc216702761)

[2.3. Biểu đồ tuần tự 39](#_Toc216702762)

[2.3.1. Biểu đồ tuần tự Tạo công thức 39](#_Toc216702763)

[2.3.2. Biểu đồ tuần tự Đăng ký tài khoản 39](#_Toc216702764)

[2.3.3. Biểu đồ tuần tự tìm kiếm công thức 40](#_Toc216702765)

[2.3.4. Biểu đồ tuần tự Gợi ý công thức theo nguyên liệu 41](#_Toc216702766)

[2.3.5. Biểu đồ tuần tự Lưu công thức 41](#_Toc216702767)

[2.4. Thiết kế cơ sở dữ liệu 42](#_Toc216702768)

[2.4.2. Danh sách các bảng trong cơ sở dữ liệu 43](#_Toc216702769)

[2.4.2.1. Bảng Users (Người dùng) 43](#_Toc216702770)

[2.4.2.2. Bảng Recipes (Công thức nấu ăn) 44](#_Toc216702771)

[2.4.2.3. Bảng Recipe\_Steps (Các bước thực hiện) 45](#_Toc216702772)

[2.4.2.4. Bảng Ingredients (Nguyên liệu) 45](#_Toc216702773)

[2.4.2.6. Bảng Recipe\_Ingredients (Nguyên liệu của công thức) 46](#_Toc216702774)

[2.4.2.7. Bảng Categories (Danh mục) 46](#_Toc216702775)

[2.4.2.8. Bảng Recipe\_Categories (Danh mục của công thức) 46](#_Toc216702776)

[2.4.2.9. Bảng Tags (Thẻ gắn) 47](#_Toc216702777)

[2.4.2.10. Bảng Recipe\_Tags (Thẻ của công thức) 47](#_Toc216702778)

[2.4.2.11. Bảng Recipe\_Ratings (Đánh giá công thức) 47](#_Toc216702779)

[2.4.2.12. Bảng Recipe\_Likes (Lượt thích công thức) 48](#_Toc216702780)

[2.4.2.13. Bảng Collections (Bộ sưu tập công thức) 48](#_Toc216702781)

[2.4.2.14. Bảng Collection\_Recipes (Công thức trong bộ sưu tập) 49](#_Toc216702782)

[2.4.2.15. Bảng Comments (Bình luận) 49](#_Toc216702783)

[2.4.2.16. Bảng Follows (Theo dõi) 49](#_Toc216702784)

[2.4.2.17. Bảng Notifications (Thông báo) 50](#_Toc216702785)

[2.4.2.18. Bảng Activity\_Logs (Hoạt động) 50](#_Toc216702786)

[2.4.2.19. Bảng Search\_History (Lịch sử tìm kiếm) 51](#_Toc216702787)

[2.4.2.20. Bảng Reports (Báo cáo vi phạm) 51](#_Toc216702788)

[2.4.2.21. Bảng Forgot\_Passwords (Quên mật khẩu) 52](#_Toc216702789)

[2.5. Thiết kế giao diện 53](#_Toc216702790)

[2.5.1. Giao diện người dùng 53](#_Toc216702791)

[2.5.2. Giao diện người dùng quản trị 57](#_Toc216702792)

[CHƯƠNG 3: TRIỂN KHAI DỰ ÁN 58](#_Toc216702793)

[3.1. Công nghệ backend 58](#_Toc216702794)

[3.1.1. Các công nghệ sử dụng phổ biến cho backend 58](#_Toc216702795)

[3.1.2. Bảng so sánh công nghệ 59](#_Toc216702796)

[3.1.3. Kết luận, lựa chọn công nghệ 60](#_Toc216702797)

[3.2. Công nghệ frontend 60](#_Toc216702798)

[3.2.1. Các công nghệ sử dụng phổ biến cho frontend 60](#_Toc216702799)

[3.2.2. Bảng so sánh công nghệ 62](#_Toc216702800)

[3.2.3. Kết luận, lựa chọn công nghệ 63](#_Toc216702801)

[3.3. Cơ sở dữ liệu 63](#_Toc216702802)

[3.3.1. Các cơ sở dữ liệu phổ biến 63](#_Toc216702803)

[3.3.2. Bảng so sánh cơ sở dữ liệu 64](#_Toc216702804)

[3.3.3. Kết luận, lựa chọn cơ sở dữ liệu 65](#_Toc216702805)

[3.4. Lưu trữ ảnh cho hệ thống 65](#_Toc216702806)

[3.4.1. Các công nghệ phổ biến để lưu trữ hình ảnh 65](#_Toc216702807)

[3.4.2. Kết luận lựa chọn công nghệ 68](#_Toc216702808)

[CHƯƠNG 4: KIỂM THỬ ỨNG DỤNG 69](#_Toc216702809)

[4.1. Công cụ và môi trường kiểm thử 69](#_Toc216702810)

[4.1.1. Môi trường kiểm thử 69](#_Toc216702811)

[4.1.2 Công cụ kiểm thử 69](#_Toc216702812)

[4.2. Xây dựng các kịch bản kiểm thử 70](#_Toc216702813)

[4.2.1. Tìm kiếm 70](#_Toc216702814)

[4.2.2. Công thức 71](#_Toc216702815)

[4.2.3. Chức năng theo dõi 72](#_Toc216702816)

[4.2.4. Chức năng bình luận 73](#_Toc216702817)

[4.2.5. Chức năng thông báo 73](#_Toc216702818)

[4.2.6. Chức năng hiển thị gợi ý 74](#_Toc216702819)

[4.2.7. Chức năng thích, đánh giá 75](#_Toc216702820)

[4.2.8. Đăng nhập/Đăng xuất 75](#_Toc216702821)

[4.2.9. Quản lý tài khoản người dùng 76](#_Toc216702822)

[4.3 Đánh giá kết quả kiểm thử 77](#_Toc216702823)

[4.3.1. Tổng quan 77](#_Toc216702824)

[4.3.2. Phân tích kết quả kiểm thử JUNIT 77](#_Toc216702825)

[4.3.2.1. Tổng quan 78](#_Toc216702826)

[4.3.2.2. Phân tích chi tiết theo module 78](#_Toc216702827)

[4.3.2.3. Kết luận 79](#_Toc216702828)

[KẾT LUẬN VÀ KIẾN NGHỊ 80](#_Toc216702829)

[DANH MỤC TÀI LIỆU THAM KHẢO 82](#_Toc216702830)

# DANH MỤC CÁC TỪ VIẾT TẮT

|  |  |  |
| --- | --- | --- |
| STT | Từ viết tắt | Ý nghĩa |
| 1 | ACID | Atomicity, Consistency, Isolation, Durability |
| 2 | API | Application Programming Interface – Giao diện lập trình ứng dụng |
| 3 | CSDL | Cơ sở dữ liệu |
| 4 | HSTORE | |  | | --- | |  |   Hash Store - Kiểu dữ liệu trong PostgreSQL cho phép lưu trữ cặp *key-value* |
| 5 | IOS | iPhone Operating System - |
| 6 | JPA | Java Persistence API - Chuẩn Java dùng để ánh xạ (ORM) giữa các đối tượng Java và bảng trong cơ sở dữ liệu quan hệ. |
| 7 | JS | JavaScript |
| 8 | JSONB | JavaScript Object Notation (Binary) |
| 9 | JWT | JSON Web Token |
| 10 | LDAP | Lightweight Directory Access Protocol |
| 11 | OAuth2 | Open Authorization 2.0 |
| 12 | SQL | Structured Query Language |
| 13 | SSO | Single Sign-On |
| 14 | XML | eXtensible Markup Language |

# DANH MỤC BẢNG BIỂU

[**Bảng 2. 1.** Bảng so sánh các ứng dụng đã có trên thị trường 27](#_Toc216702596)

[**Bảng 2. 2.** Bảng Users 43](#_Toc216702597)

[**Bảng 2. 3.** Bảng Recipes 44](#_Toc216702598)

[**Bảng 2. 4.** Bảng Recipe\_Steps 45](#_Toc216702599)

[**Bảng 2. 5.** Bảng Ingredients 45](#_Toc216702600)

[**Bảng 2. 6.** Bảng Recipe\_Ingredients 46](#_Toc216702601)

[**Bảng 2. 7.** Bảng Categories 46](#_Toc216702602)

[**Bảng 2. 8.** Bảng Recipe\_Categories 46](#_Toc216702603)

[**Bảng 2. 9.** Bảng Tags 47](#_Toc216702604)

[**Bảng 2. 10.** Bảng Recipe\_Tags 47](#_Toc216702605)

[**Bảng 2. 11.** Bảng Recipe\_Ratings 47](#_Toc216702606)

[**Bảng 2. 12.** Bảng Recipe\_Likes 48](#_Toc216702607)

[**Bảng 2. 13.** Bảng Collections 48](#_Toc216702608)

[**Bảng 2. 14.** Bảng Collection\_Recipes 49](#_Toc216702609)

[**Bảng 2. 15.** Bảng Comments 49](#_Toc216702610)

[**Bảng 2. 16.** Bảng Follows 49](#_Toc216702611)

[**Bảng 2. 17.** Bảng Notifications 50](#_Toc216702612)

[**Bảng 2. 18**. Bảng Activity\_Logs 50](#_Toc216702613)

[**Bảng 2. 19**. Bảng Search\_History 51](#_Toc216702614)

[**Bảng 2. 20.** Bảng Reports 51](#_Toc216702615)

[**Bảng 2. 21.** Bảng Forgot\_Passwords 52](#_Toc216702616)

[**Bảng 3. 1.** Bảng so sánh công nghệ Backend 59](#_Toc216702617)

[**Bảng 3. 2.** Bảng so sánh công nghệ Fontend 62](#_Toc216702618)

[**Bảng 3. 3.** Bảng so sánh cơ sở dữ liệu 64](#_Toc216702619)

# DANH MỤC HÌNH ẢNH

[**Hình 2. 1.** Use case Quản lý người dùng 30](#_Toc216702575)

[**Hình 2. 2.** Use case Quản lý công thức nấu ăn 31](#_Toc216702576)

[**Hình 2. 3.** Use case Tìm kiếm công thức 31](#_Toc216702577)

[**Hình 2. 4.** Use case Gợi ý/ đề xuất công thức 32](#_Toc216702578)

[**Hình 2. 5.** Use case Tương tác người dùng 33](#_Toc216702579)

[**Hình 2. 6.** Activity chức năng đăng nhập 34](#_Toc216702580)

[**Hình 2. 7.** Activity chức năng tạo công thức 35](#_Toc216702581)

[**Hình 2. 8.** Activity chức năng tìm kiếm công thức 36](#_Toc216702582)

[**Hình 2. 9.** Activity chức năng gợi ý công thức theo nguyên liệu 37](#_Toc216702583)

[**Hình 2. 10.** Activity chức năng theo dõi người dùng 38](#_Toc216702584)

[**Hình 2. 11.** Sequence chức năng tạo công thức 39](#_Toc216702585)

[**Hình 2. 12.** Sequence chức năng đăng ký tài khoản 39](#_Toc216702586)

[**Hình 2. 13.** Sequence chức năng tìm kiếm 40](#_Toc216702587)

[**Hình 2. 14.** Sequence chức năng gợi ý theo nguyên liệu 41](#_Toc216702588)

[**Hình 2. 15.** Sequence chức năng lưu công thức 41](#_Toc216702589)

[**Hình 2. 18.** CSDL tổng quan 42](#_Toc216702590)

[**Hình 2. 19.** Giao diện trang chủ, trang tìm kiếm, kết quả tìm kiếm 53](#_Toc216702591)

[**Hình 2. 20.** Giao diện trang đăng nhập, thống báo, cài đặt 54](#_Toc216702592)

[**Hình 2. 21.** Giao diện trang tạo công thức, chi tiết công thức 55](#_Toc216702593)

[**Hình 2. 22.** Giao diện trang cá nhân, người dùng, công thức đã lưu 56](#_Toc216702594)

[**Hình 2. 23.** Giao diện của người dùng quản trị 57](#_Toc216702595)

# MỞ ĐẦU

Trong thời đại công nghệ thông tin phát triển mạnh mẽ, chuyển đổi số đã và đang trở thành xu thế tất yếu trên toàn cầu. Công nghệ thông tin không chỉ đóng vai trò quan trọng trong lĩnh vực kinh tế, giáo dục, y tế mà còn góp phần thay đổi sâu sắc cách con người tiếp cận, học hỏi và chia sẻ tri thức trong đời sống hằng ngày. Đặc biệt, khi bước vào cuộc cách mạng công nghiệp 4.0, việc ứng dụng công nghệ trong các hoạt động thường nhật, như nấu ăn và chia sẻ kinh nghiệm ẩm thực, đang ngày càng phổ biến và cần thiết.

Ẩm thực là một phần không thể thiếu trong văn hóa của mỗi quốc gia. Tại Việt Nam, với nền ẩm thực phong phú, đa dạng và mang đậm bản sắc dân tộc, việc chia sẻ các công thức nấu ăn, bí quyết chế biến và cách trình bày món ăn không chỉ giúp bảo tồn giá trị truyền thống mà còn tạo cơ hội giao lưu, học hỏi giữa các thế hệ và vùng miền. Tuy nhiên, trong thực tế, nhiều người yêu thích nấu ăn gặp khó khăn trong việc tìm kiếm nguồn công thức chất lượng, có hình ảnh minh họa rõ ràng, hướng dẫn cụ thể hoặc công cụ tương tác với cộng đồng nấu ăn khác. Các trang web hoặc ứng dụng hiện có đôi khi thiếu sự tiện lợi, giao diện thân thiện hoặc chưa hỗ trợ đầy đủ các tính năng như bình luận, đánh giá, theo dõi người dùng hay tạo bộ sưu tập công thức riêng.

Chính vì vậy, nhóm chúng em lựa chọn đề tài **“Xây dựng ứng dụng chia sẻ các công thức nấu ăn”** với mục tiêu phát triển một hệ thống giúp người dùng có thể dễ dàng đăng tải, lưu trữ, tìm kiếm, chia sẻ và đánh giá các công thức nấu ăn. Ứng dụng hướng đến việc tạo ra một cộng đồng ẩm thực trực tuyến, nơi mọi người có thể tương tác, học hỏi và truyền cảm hứng cho nhau trong hành trình nấu nướng.

Báo cáo gồm hai chương chính:

**Chương 1: Giới thiệu đề tài và các công nghệ sử dụng.**

Chương này trình bày tổng quan về đề tài, mục tiêu xây dựng ứng dụng, cùng các công nghệ được lựa chọn để phát triển như Java, PostgreSQL, Spring Boot và React Native.

**Chương 2: Phân tích và thiết kế hệ thống.**

Chương này tập trung vào việc thu thập, phân tích yêu cầu, mô tả chức năng, xây dựng các biểu đồ UML (use case, hoạt động, tuần tự), thiết kế cơ sở dữ liệu

**Chương 3: Triển khai dự án**

Chương này trình bày về môi trường cài đặt và các giao diện mà nhóm đã xây dựng được để phục vụ các chức năng của người dùng

**Chương 4: Kiểm thử**

Chương này tập trung vào việc mô tả và thực hiện các unit test cùng với các ca kiểm thử để kiểm thử các chức năng của dự án.

Thông qua việc thực hiện đề tài này, nhóm mong muốn củng cố kiến thức đã học về quy trình phát triển phần mềm, đồng thời tạo ra một sản phẩm có tính ứng dụng cao, góp phần lan tỏa niềm yêu thích nấu ăn và kết nối cộng đồng ẩm thực Việt Nam.  
Nhóm xin chân thành cảm ơn thầy cô đã hướng dẫn, hỗ trợ và đóng góp ý kiến quý báu để chúng em hoàn thiện đề tài này!

# CHƯƠNG 1: PHÂN TÍCH YÊU CẦU

## 1.1. Giới thiệu chung bài toán

Trong thời đại số, nhu cầu tìm kiếm, học hỏi và chia sẻ kinh nghiệm nấu ăn ngày càng tăng cao. Mọi người không chỉ coi nấu ăn là một công việc hàng ngày mà còn là một sở thích, một phương pháp thư giãn và kết nối với gia đình, bạn bè hay những người có cùng sở thích với nhau. Tuy nhiên, các công thức nấu ăn hiện đang bị phân mảnh trên nhiều nền tảng khác nhau như mạng xã hội, blog cá nhân, website, YouTube, TikTok khiến người dùng gặp khó khăn trong việc tìm kiếm, lưu trữ và quản lý một cách có hệ thống.

Việc thiếu một không gian chuyên biệt dành cho cộng đồng cũng làm giảm khả năng tương tác, học hỏi và trao đổi kinh nghiệm giữa những người có cùng đam mê. Người dùng khó có thể theo dõi những đầu bếp mình yêu thích, thảo luận về một công thức cụ thể, hay nhận được những gợi ý món ăn phù hợp với điều kiện cá nhân (nguyên liệu có sẵn, thời tiết).

Từ thực tế đó, bài toán xây dựng một **"Ứng dụng Chia sẻ Công thức Nấu ăn"** được đặt ra. Đây không chỉ là một kho lưu trữ công thức mà còn là một mạng xã hội thu nhỏ, nơi người dùng có thể tạo, chia sẻ, khám phá và tương tác với nhau, giải quyết triệt để những vấn đề nêu trên.

Bài toán sẽ hướng tới việc xây dựng một nền tảng tập trung, nơi người dùng có thể dễ dàng tìm kiếm, chia sẻ, quản lý các công thức của bản thân hoặc theo dõi người khác một cách tiện lợi, giúp mọi người có thể chia sẻ và học hỏi lẫn nhau. Cùng với các tính năng thông minh để giúp người dùng có thể trải nghiệm một cách tiện lợi và tốt nhất để những người có chung niềm đam mê, góp phần làm phong phú thêm văn hóa ẩm thực.

## 1.2. Thu nhập yêu cầu

### 1.2.1. Phỏng vấn

|  |  |
| --- | --- |
| Kế hoạch phỏng vấn | |
| Người được hỏi: Sinh viên | Người phỏng vấn: Nguyễn Tài Nhất, Bạch Phương Anh |
| Địa chỉ: Trường GTVT | Thời gian: 08/09/2025 |
| Giới thiệu chủ đề sẽ đề cập, xin phép ghi âm  Chủ đề 1: Các câu hỏi về nhu cầu tìm công thức nấu ăn  Chủ đề 2: Các câu hỏi về những chức năng cần thiết cho hệ thống   Tổng hợp và kết thúc | 2 phút  7 phút  9 phút  2 phút |
| Tổng cộng: 20 phút | |

|  |  |
| --- | --- |
| Phiếu phỏng vấn | |
| Người được hỏi: Sinh viên – Nguyễn Thị Thu Trang – K64 | Ngày 08/09 Người hỏi: Nguyễn Tài Nhất |
| C1: Bạn thường sử dụng nền tảng nào để tìm công thức nấu ăn? Vì sao? | Mình thường dùng YouTube vì có video hướng dẫn trực quan, dễ làm theo. |
| C2: Điều gì bạn thích và không thích ở nền tảng đó? | Mình thích vì có nhiều video phong phú, nhưng không thích vì nhiều video dài dòng, mất thời gian. |
| C3: Tính năng nào là quan trọng nhất với bạn khi dùng ứng dụng nấu ăn? | Có video hướng dẫn ngắn gọn và hình ảnh minh họa rõ ràng. |
| C4: Bạn có gặp khó khăn nào khi tìm công thức nấu ăn không? Nếu có, đó là gì? | Khó khăn lớn nhất là đôi khi công thức không có định lượng rõ ràng, khó làm chuẩn. |
| C5: Bạn mong đợi gì ở một ứng dụng chia sẻ công thức nấu ăn mới? | Mình muốn có tính năng lọc công thức theo nguyên liệu sẵn có ở nhà. |
| C6: Bạn có thường chia sẻ lại công thức nấu ăn của mình không? Nếu có, bạn muốn chia sẻ bằng cách nào? | Mình ít khi tự đăng công thức, nhưng nếu ứng dụng có sẵn mẫu ghi chép công thức đơn giản (chỉ cần điền nguyên liệu + bước nấu) thì mình sẽ dễ dàng chia sẻ hơn. |
| C7: Bạn có muốn tương tác với những người cùng sở thích nấu ăn không? Nếu có, bạn mong muốn những hình thức tương tác nào? | Có, mình muốn có phần “hỏi đáp nhanh” để mình có thể đặt câu hỏi khi bị vướng trong lúc nấu và nhận được phản hồi từ người khác. |

|  |  |
| --- | --- |
| Phiếu phỏng vấn | |
| Người được hỏi: Sinh viên – Nguyễn Thị Thu Trang – K64 | Ngày 08/09 Người hỏi: Nguyễn Tài Nhất |
| C1: Bạn thường sử dụng nền tảng nào để tìm công thức nấu ăn? Vì sao? | Mình thường dùng YouTube vì có video hướng dẫn trực quan, dễ làm theo. |
| C2: Điều gì bạn thích và không thích ở nền tảng đó? | Mình thích vì có nhiều video phong phú, nhưng không thích vì nhiều video dài dòng, mất thời gian. |
| C3: Tính năng nào là quan trọng nhất với bạn khi dùng ứng dụng nấu ăn? | Có video hướng dẫn ngắn gọn và hình ảnh minh họa rõ ràng. |
| C4: Bạn có gặp khó khăn nào khi tìm công thức nấu ăn không? Nếu có, đó là gì? | Khó khăn lớn nhất là đôi khi công thức không có định lượng rõ ràng, khó làm chuẩn. |
| C5: Bạn mong đợi gì ở một ứng dụng chia sẻ công thức nấu ăn mới? | Mình muốn có tính năng lọc công thức theo nguyên liệu sẵn có ở nhà. |
| C6: Bạn có thường chia sẻ lại công thức nấu ăn của mình không? Nếu có, bạn muốn chia sẻ bằng cách nào? | Mình ít khi tự đăng công thức, nhưng nếu ứng dụng có sẵn mẫu ghi chép công thức đơn giản (chỉ cần điền nguyên liệu + bước nấu) thì mình sẽ dễ dàng chia sẻ hơn. |
| C7: Bạn có muốn tương tác với những người cùng sở thích nấu ăn không? Nếu có, bạn mong muốn những hình thức tương tác nào? | Có, mình muốn có phần “hỏi đáp nhanh” để mình có thể đặt câu hỏi khi bị vướng trong lúc nấu và nhận được phản hồi từ người khác. |

|  |  |
| --- | --- |
| Phiếu phỏng vấn | |
| Người được hỏi: Sinh viên – Trần Văn Huy – K66 | Ngày 09/09 Người hỏi: Nguyễn Tài Nhất |
| C1: Bạn thường sử dụng nền tảng nào để tìm công thức nấu ăn? Vì sao? | Mình hay dùng Cookpad vì có nhiều công thức đa dạng và có bình luận từ người dùng khác. |
| C2: Điều gì bạn thích và không thích ở nền tảng đó? | Mình thích phần bình luận vì có kinh nghiệm thực tế. Tuy nhiên, nhiều công thức bị lặp lại, khó chọn. |
| C3: Tính năng nào là quan trọng nhất với bạn khi dùng ứng dụng nấu ăn? | Phần đánh giá/xếp hạng công thức để chọn công thức đáng tin cậy. |
| C4: Bạn có gặp khó khăn nào khi tìm công thức nấu ăn không? Nếu có, đó là gì? | Khó tìm công thức phù hợp khẩu vị vì phần tìm kiếm chưa thông minh. |
| C5: Bạn mong đợi gì ở một ứng dụng chia sẻ công thức nấu ăn mới? | Mình muốn có gợi ý công thức dựa trên nguyên liệu trong tủ lạnh. |
| C6: Bạn có thường chia sẻ lại công thức nấu ăn của mình không? Nếu có, bạn muốn chia sẻ bằng cách nào? | Mình ít chia sẻ nhưng nếu ứng dụng có giao diện dễ đăng kèm hình ảnh và video ngắn thì mình cũng muốn thử. |
| C7: Bạn có muốn tương tác với những người cùng sở thích nấu ăn không? Nếu có, bạn mong muốn những hình thức tương tác nào? | Có, mình muốn có chỗ bình luận, trao đổi kinh nghiệm và có thể tham gia nhóm cộng đồng theo từng chủ đề món ăn. |

|  |  |
| --- | --- |
| Phiếu phỏng vấn | |
| Người được hỏi: Sinh viên – Trần Thu Hà – K65 | Ngày 09/09 Người hỏi: Bạch Phương Anh |
| C1: Bạn thường sử dụng nền tảng nào để tìm công thức nấu ăn? Vì sao? | Mình thường tìm trên Google vì nhanh và có nhiều nguồn để tham khảo. |
| C2: Điều gì bạn thích và không thích ở nền tảng đó? | Thích: Kết quả đa dạng, dễ chọn lọc. Không thích: Công thức lẫn lộn, nhiều khi không biết công thức nào chính xác và uy tín. |
| C3: Tính năng nào là quan trọng nhất với bạn khi dùng ứng dụng nấu ăn? | Tìm kiếm thông minh theo món ăn hoặc nguyên liệu để không mất thời gian lọc kết quả. |
| C4: Bạn có gặp khó khăn nào khi tìm công thức nấu ăn không? Nếu có, đó là gì? | Có, nhiều công thức không ghi rõ thời gian nấu, nên khó sắp xếp khi mình chỉ có ít thời gian rảnh. |
| C5: Bạn mong đợi gì ở một ứng dụng chia sẻ công thức nấu ăn mới? | Mình muốn ứng dụng có chức năng hẹn giờ nấu ăn kèm thông báo (như khi luộc trứng hay hầm canh) để tiện theo dõi. |
| C6: Bạn có thường chia sẻ lại công thức nấu ăn của mình không? Nếu có, bạn muốn chia sẻ bằng cách nào? | Mình thỉnh thoảng có chia sẻ, nhưng chủ yếu qua Facebook hoặc gửi trực tiếp cho bạn bè. Nếu ứng dụng có chỗ đăng kèm hình ảnh và video ngắn thì mình sẽ thích hơn. |
| C7: Bạn có muốn tương tác với những người cùng sở thích nấu ăn không? Nếu có, bạn mong muốn những hình thức tương tác nào? | Có, mình muốn có phần bình luận, đánh giá và nhóm cộng đồng nhỏ để cùng trao đổi kinh nghiệm, chứ không chỉ xem công thức một chiều. |

**Tổng hợp kết quả sau khi phỏng vấn:**

Người thực hiện: Nguyễn Tài Nhất, Bạch Phương Anh

Đối tượng phỏng vấn: Sinh viên Trường đại học GTVT

Thời gian phỏng vấn: 08/09/2025 – 09/09/2025

**Mục tiêu:** Khảo sát thói quen, kênh tìm công thức, trải nghiệm hiện tại và kỳ vọng tính năng của sinh viên đối với ứng dụng chia sẻ công thức nấu ăn.

**Tổng hợp kết quả phỏng vấn:**

* + Ứng dụng phổ biến: YouTube, Cookpad, Google (cũng có dùng Facebook group). Sinh viên dùng đa kênh: video (YouTube) + nền tảng tiếng Việt (Cookpad) + tìm nhanh qua Google.
  + Điểm yêu thích:
    - Video/hình ảnh minh họa trực quan (dễ làm theo).
    - Nội dung đa dạng, nhiều công thức phù hợp khẩu vị.
    - Bình luận/kinh nghiệm từ người dùng giúp chọn công thức.
  + Điểm không thích / hạn chế:
    - Quảng cáo nhiều và video quá dài (YouTube).
    - Công thức bị lặp, khó chọn lọc (Cookpad).
    - Độ tin cậy/khó xác định công thức chuẩn (Google).
    - Khó tìm lại công thức đã xem; thiếu định lượng / thời gian nấu rõ ràng.
  + Tính năng mong muốn:
    - Gợi ý công thức theo nguyên liệu sẵn có.
    - Tìm kiếm thông minh (theo nguyên liệu, thời gian, độ khó).
    - Lưu/Quản lý công thức (sổ tay cá nhân).
    - Gợi ý theo thời tiết hàng ngày
    - Đăng công thức đơn giản (form + upload ảnh/video).
  + Khó khăn khi sử dụng hiện nay:
    - Công thức thiếu định lượng/chuẩn khiến khó làm chuẩn.
    - Khó tìm lại công thức đã xem (bài bị trôi).
    - Tìm công thức phù hợp khẩu vị khó do bộ lọc chưa thông minh.

### 1.2.2. Phiếu hỏi

Phiếu hỏi này được xây dựng nhằm khảo sát nhu cầu, thói quen và mong đợi của người dùng đối với một ứng dụng chia sẻ công thức nấu ăn. Thông tin thu thập được sẽ giúp nhóm nghiên cứu:

* Xác định nền tảng, công cụ mà người dùng hiện đang sử dụng để tìm kiếm công thức nấu ăn.
* Tìm hiểu những khó khăn, hạn chế khi người dùng tiếp cận các công thức hiện nay.
* Đánh giá các tính năng quan trọng và cần thiết cho một ứng dụng chia sẻ công thức nấu ăn.
* Ghi nhận các đề xuất, kỳ vọng của người dùng để thiết kế một ứng dụng phù hợp, tiện lợi và mang lại trải nghiệm tốt nhất.

1. Bạn là? ☐ Học sinh/Sinh viên ☐ Nhân viên văn phòng ☐ Nội trợ ☐ Khác: ..............

2. Bạn thường nấu ăn mấy lần/tuần?

☐ Hàng ngày ☐ 3–4 lần/tuầng ☐ Hiếm khi ☐ Hầu như không

3. Khi nấu ăn, bạn quan tâm nhất đến điều gì? (Có thể chọn nhiều)

☐ Thời gian nấu nhanh

☐ Nguyên liệu dễ kiếm

☐ Dinh dưỡng và sức khỏe

☐ Trình bày đẹp mắt

☐ Khác:..........

4. Bạn thường tìm công thức nấu ăn trên nền tảng nào? (Có thể chọn nhiều)

☐ YouTube ☐ Facebook Group ☐ Google ☐ Cookpad ☐ Tasty ☐ TikTok

☐ Khác: ............

5. Bạn thường gặp khó khăn gì khi tìm công thức nấu ăn? (Có thể chọn nhiều)

☐ Công thức không rõ định lượng

☐ Video quá dài hoặc thiếu chi tiết

☐ Khó tìm lại công thức đã xem

☐ Thiếu tính năng lọc theo nguyên liệu/khẩu vị/thời gian

☐ Khác: ………

6. Tính năng nào bạn thấy quan trọng nhất trong một ứng dụng chia sẻ công thức nấu ăn? (Có thể chọn nhiều)

☐ Tìm kiếm thông minh (theo nguyên liệu, tên món, thời gian)

☐ Video hướng dẫn ngắn gọn, dễ hiểu

☐ Bình luận, đánh giá từ cộng đồng

☐ Lưu & quản lý sổ tay công thức riêng

☐ Khác: ………

7. Bạn mong muốn ứng dụng có thêm những chức năng gì? (Có thể chọn nhiều)

☐ Gợi ý món ăn theo nguyên liệu sẵn có

☐ Gợi ý theo thời tiết/khẩu vị

☐ Đơn vị đo lường linh hoạt (gram, ml, cup…)

☐ Hẹn giờ nấu ăn và thông báo

☐ Gợi ý thay thế nguyên liệu khó tìm

8. Bạn muốn xem công thức dưới dạng nào?

☐ Văn bản + hình ảnh minh họa

☐ Video ngắn (2-3 phút)

☐ Video chi tiết (10–15 phút)

☐ Kết hợp hình ảnh và video

9. Nếu có một “cộng đồng mini” trong ứng dụng (nơi thảo luận, hỏi đáp), bạn có tham gia không?

☐ Có, rất muốn

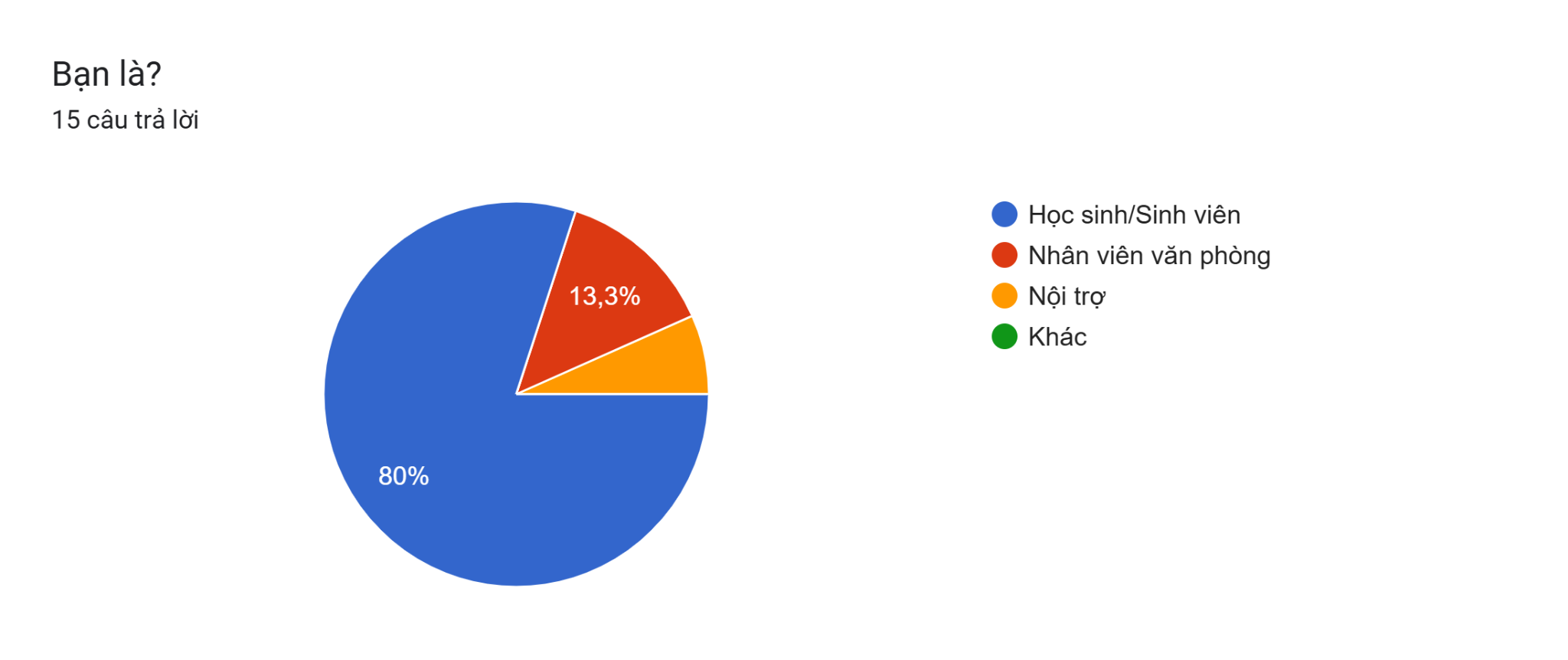
☐ Có thể, nếu tiện lợi

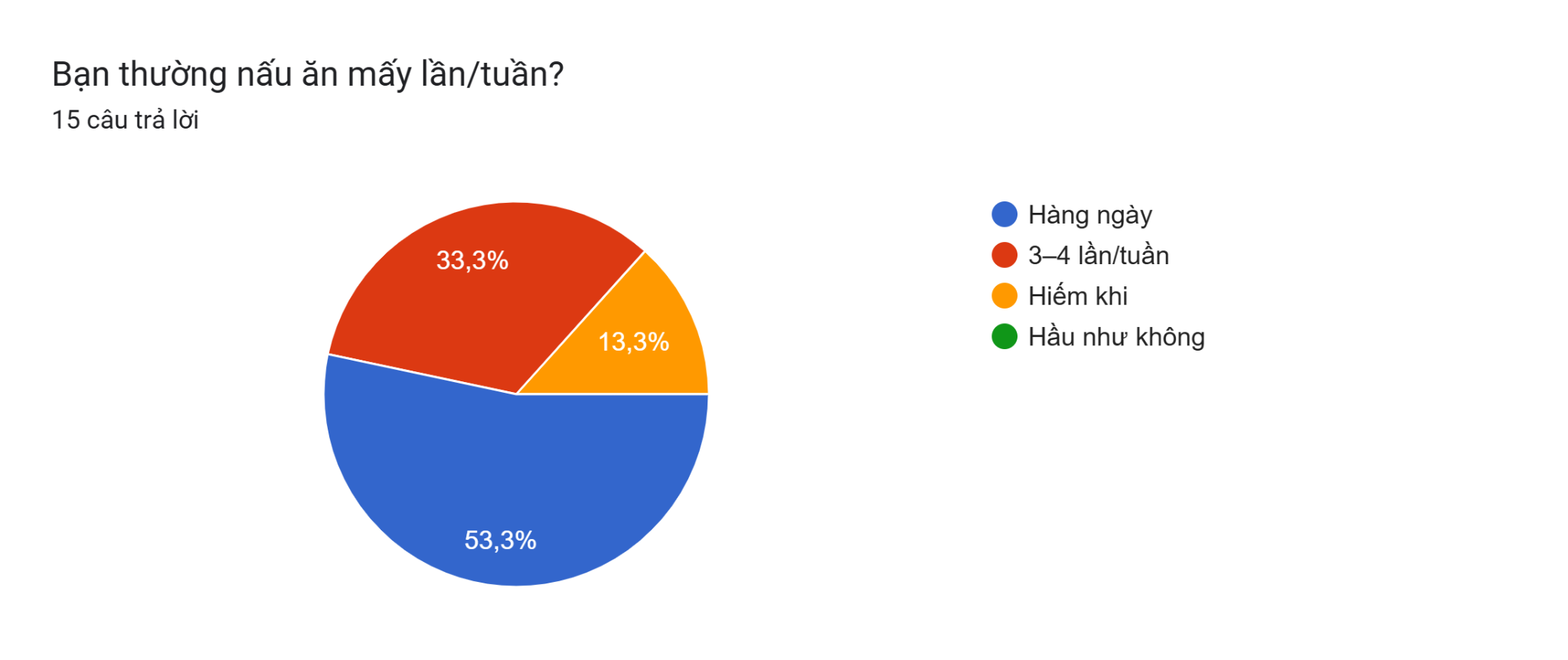
☐ Không

Nếu ứng dụng này thành công, bạn muốn trở thành người dùng bình thường hay người sáng tạo công thức (creator)? Vì sao?

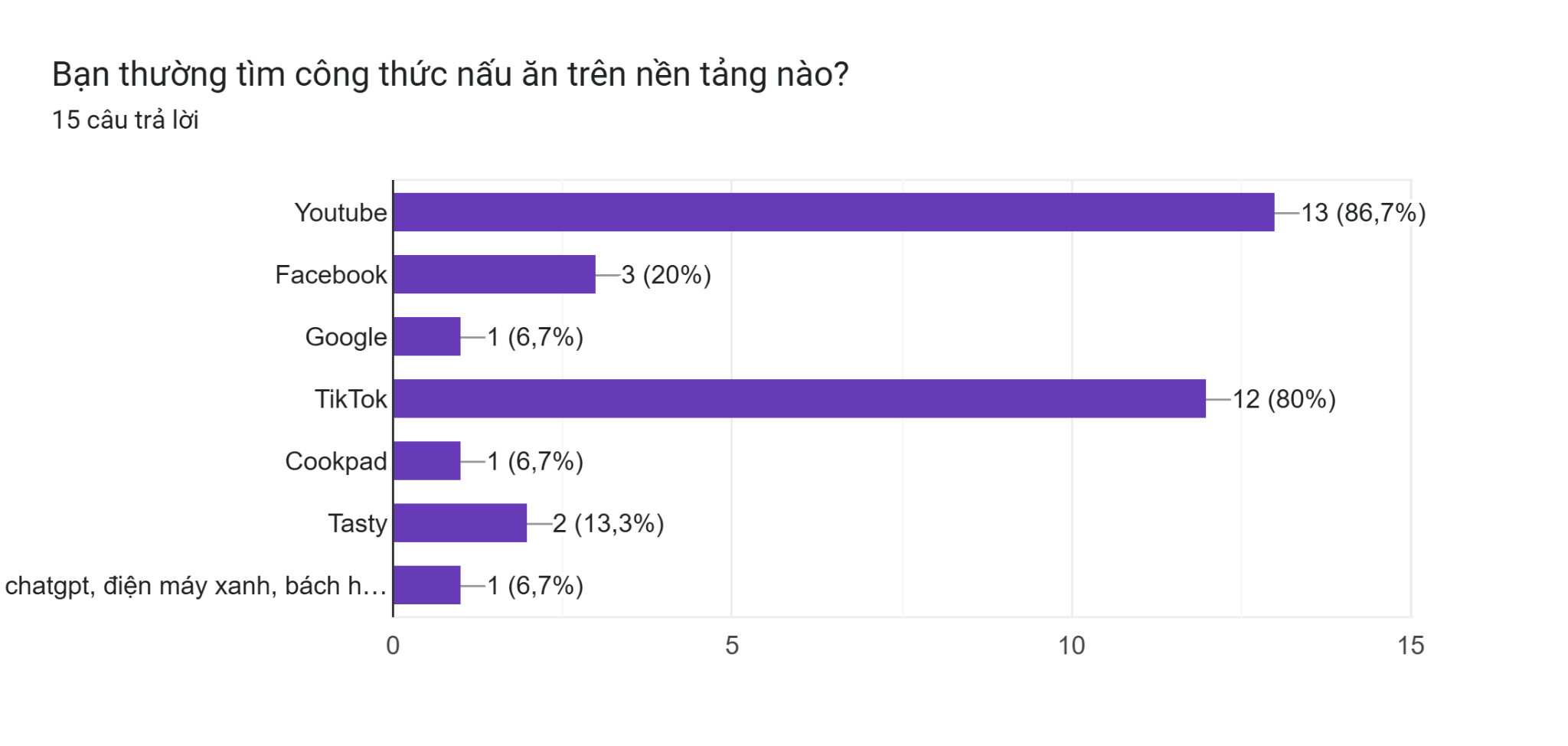
…………………………………………………..

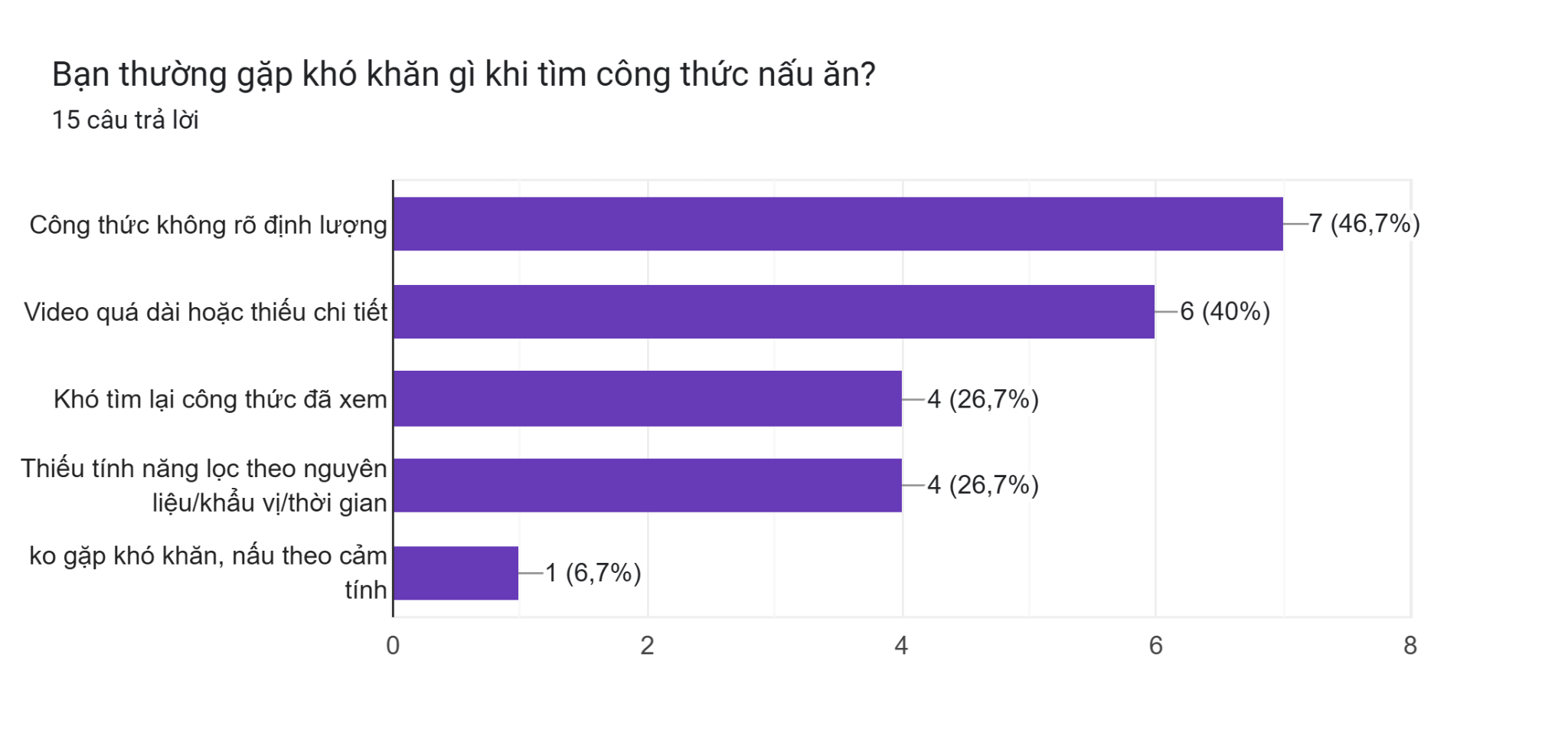
**Kết quả sau khi thực hiện phiếu hỏi:**

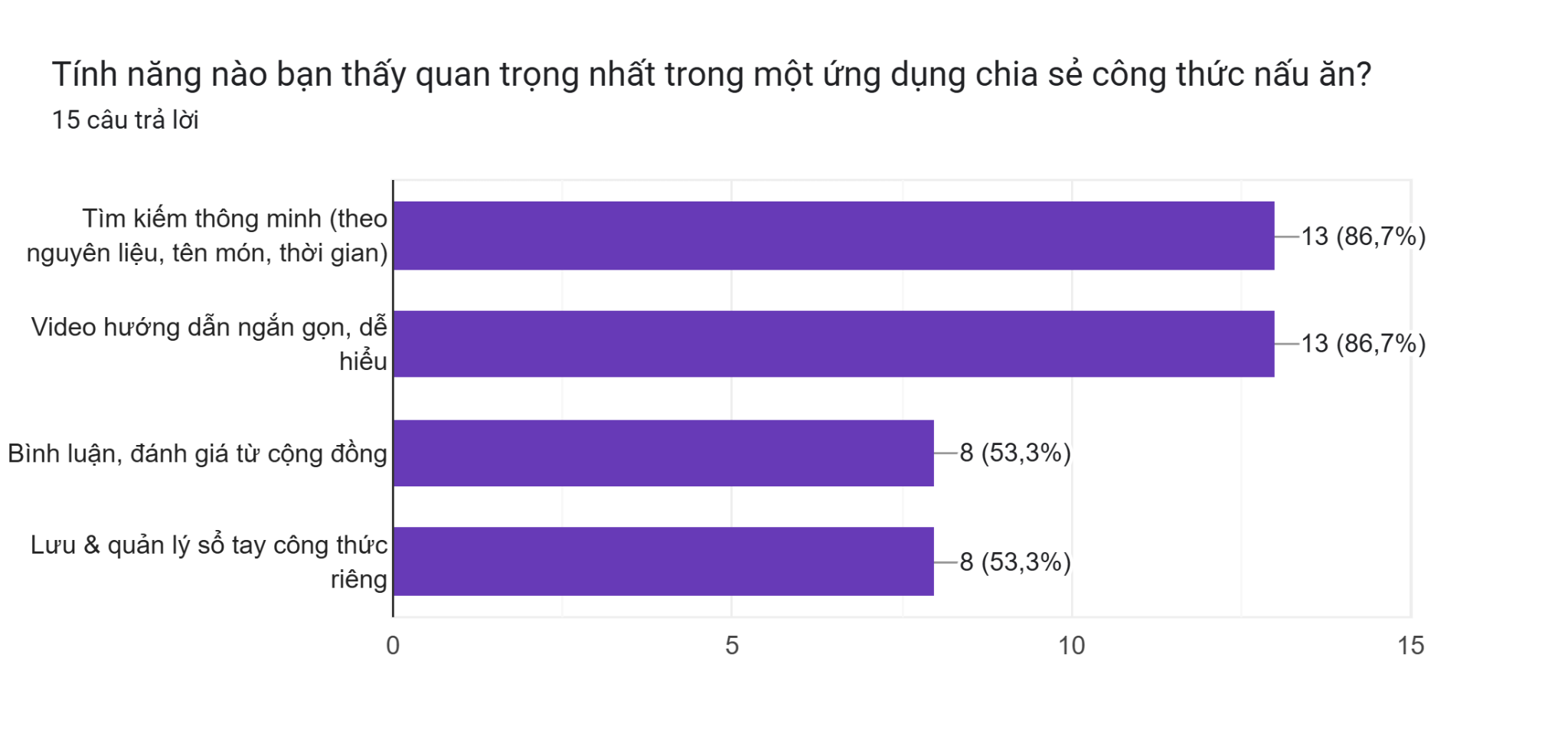


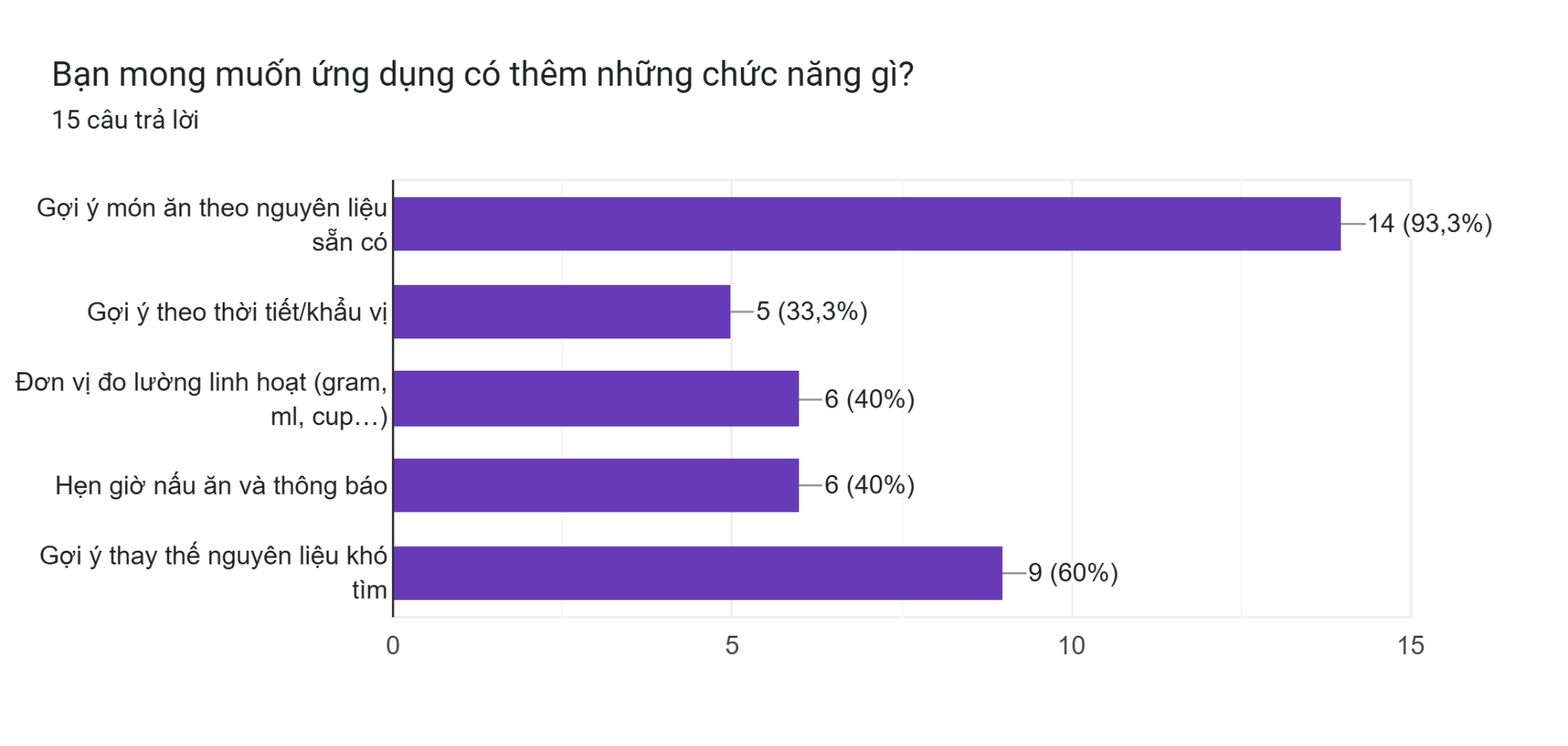


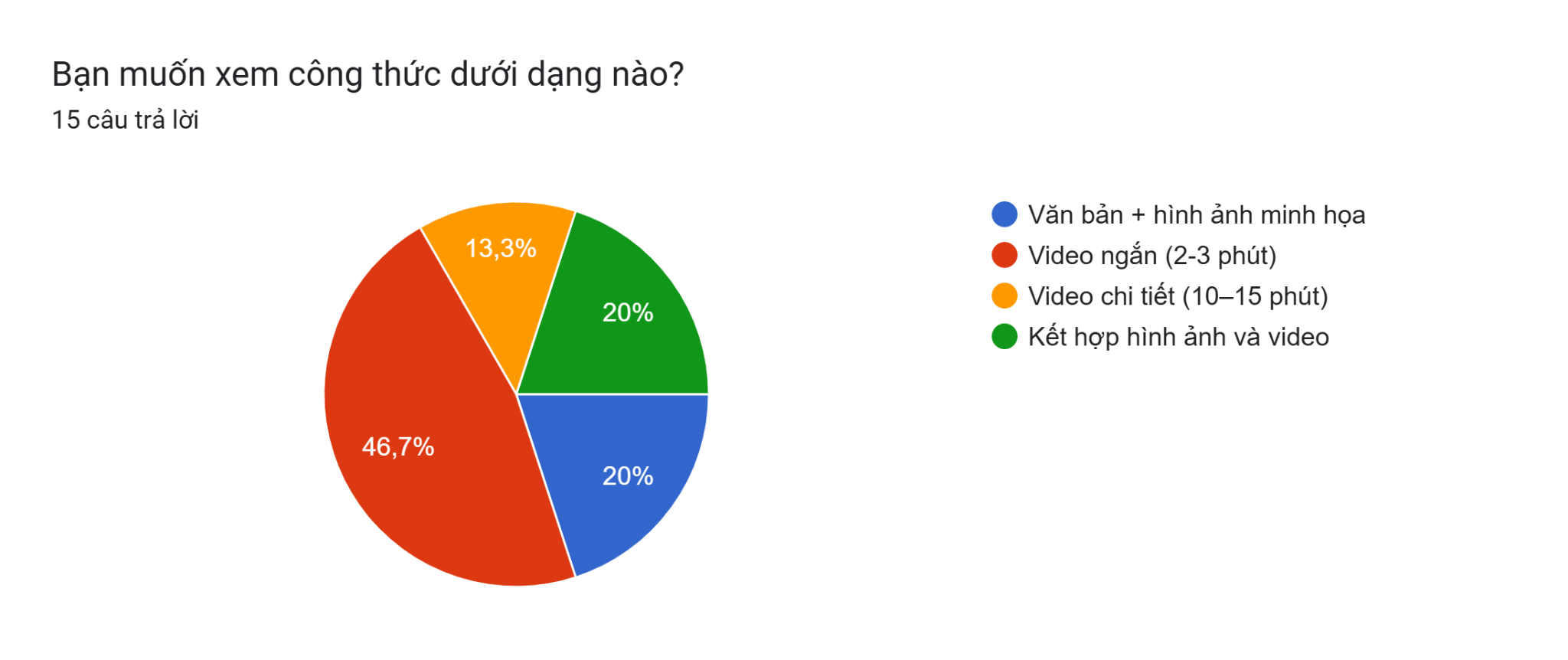
Biểu đồ câu trả lời của biểu mẫu. Tên câu hỏi: Khi nấu ăn, bạn quan tâm nhất đến điều gì?
. Số lượng câu trả lời: 15 câu trả lời.















A screenshot of a phone

AI-generated content may be incorrect.

Khảo sát cho thấy đa số người tham gia là học sinh, sinh viên với thói quen nấu ăn thường xuyên (hàng ngày hoặc ít nhất 3–4 lần/tuần). Khi nấu ăn, họ quan tâm nhiều nhất đến yếu tố dinh dưỡng, thời gian chế biến nhanh và nguyên liệu dễ tìm. Các nền tảng phổ biến để tìm công thức là Youtube, TikTok và Facebook, tuy nhiên người dùng thường gặp khó khăn do công thức thiếu chi tiết, không rõ định lượng hoặc video quá dài. Vì vậy, tính năng được mong muốn nhất ở một ứng dụng chia sẻ công thức là khả năng tìm kiếm thông minh theo nguyên liệu, tên món hay chế độ ăn. Bên cạnh đó, người dùng cũng kỳ vọng ứng dụng có thêm gợi ý món ăn từ nguyên liệu sẵn có, gợi ý theo khẩu vị, hoặc theo thời tiết đồng thời hỗ trợ trình bày công thức bằng cả video ngắn, video chi tiết và hình ảnh minh họa. Đa số đồng ý tham gia cộng đồng mini trong ứng dụng nếu tiện lợi, và phần lớn mong muốn sử dụng ứng dụng như người dùng bình thường, một số ít sẵn sàng trở thành người sáng tạo công thức.

### 1.2.3. Tổng hợp chức năng sau khi thực hiện khảo sát

Các chức năng mà người dùng mong muốn:

* Gợi ý công thức theo nguyên liệu sẵn có, thời tiết, theo bảng xếp hạng người dùng
* Tìm kiếm thông minh (theo nguyên liệu, thời gian, độ khó).
* Lưu/Quản lý công thức (sổ tay cá nhân).
* Gợi ý theo thời tiết hàng ngày
* Đăng công thức đơn giản (form + upload ảnh/video).
* Định lượng nguyên liệu theo công thức
* Hình ảnh hướng dẫn kèm video kết hợp mô tả từng bước
* Bình luận, like, đánh giá từ cộng đồng
* Lập kế hoạch công thức theo bữa ăn, tuần
* Tích hợp danh sách mua nguyên liệu
* Các chế độ ăn dinh dưỡng để người dùng có thể các nhân hóa

Các chức năng dự kiến nhóm sẽ thực hiện khi triển khai ứng dụng:

* Đăng nhập, đăng ký
* Tìm kiếm công thức theo nguyên liệu
* Tạo và đăng công thức nấu ăn
* Hiển thị chi tiết công thức kết hợp giữa mô tả văn bản và hình ảnh
* Quản lý trang cá nhân và lưu công thức yêu thích theo bộ sưu tập
* Gợi ý công thức theo độ phổ biến, theo nguyên liệu
* Theo dõi người dùng, like công thức nấu ăn
* Bảng xếp hạng món ăn (theo độ yêu thích mà người dùng bình chọn)

### 1.2.4. Khảo sát các phần mềm đã có

1. **App CookPad**

**Tính năng:**

* Đăng nhập, đăng ký
* Tìm kiếm, phân loại theo nguyên liệu
* Hiển thị công thức cách làm thông qua hình ảnh và mô tả
* Tạo và Chia sẻ công thức giữa mọi người
* Thực hiện các tương tác theo dõi người dùng, comment
* Quản lý theo tài khoản cá nhân

**Ưu điểm:**

* Giao diện đơn giản dễ sử dụng
* Phù hợp với người muốn tìm kiếm nhanh không yêu cầu nhiều chức năng

**Nhược điểm:**

Phiên bản miễn phí bị giới hạn chức năng, thường xuyên có quảng cáo

Chỉ mô tả các công thức một cách đơn giản

Không thể truy cập một số công thức theo bảng xếp hạng

1. **App Kitchen Stories**

**Tính năng:**

* Xem hơn 10.000 công thức nấu ăn miễn phí.
* Video HD và hình ảnh minh họa cho từng công thức.
* Cooking Mode – hướng dẫn từng bước khi nấu.
* Chuyển đổi lượng nguyên liệu theo số người.
* Hẹn giờ (Timer) cho từng bước nấu.
* Lưu công thức yêu thích vào “Cookbook” cá nhân.
* Tìm kiếm nâng cao: theo tên món, chế độ ăn, độ khó, thời gian, khẩu vị.  
  Cộng đồng chia sẻ: bình luận, đăng ảnh món đã nấu.

**Ưu điểm:**

* Nhiều công thức miễn phí, hướng dẫn chi tiết
* Mô tả hình ảnh kết hợp video minh họa
* Định lượng nguyên liệu theo số người
* Tìm kiếm đa dạng theo độ khó, thời gian, khẩu vị

**Nhược điểm:**

* Một số chức năng bị giới hạn trong bản miễn phí
* Các công thức upload cần thời gian kiểm duyệt không được tự ý đăng tải
* Không có bảng xếp hạng các công thức được yêu thích

## 1.3. So sánh các ứng dụng đã có trên thị trường

**Bảng 2. 1.** Bảng so sánh các ứng dụng đã có trên thị trường

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | App Cookpad | App Allrecipes | App Tasty | App CookBook | App Kitchen Stories | App dự tính |
| Đăng ký, đăng nhập | Có | Có | Có | Không có | Có | Có |
| Tìm kiếm, phân loại | Có | Có | Có | Có | Có | Có |
| Tạo và chia sẻ công thức | Có | Có | Không | Không | Có | Có |
| Hiển thị chi tiết công thức | Có | Có | Có | Có | Có | Có |
| Theo dõi, like | Có | Có | Có | Có | Có | Có |
| Comment | Không có | Có | Có | Có | Có | Không có |
| Quản lý người dùng | Có | Có | Có | Không có | Có | Có |
| Quản lý trang cá nhân | Có | Có | Có | Không có | Có | Có |
| Lưu công thức yêu thích | Có | Có | Có | Có | Có | Có |
| Gợi ý trên nguyên liệu có sẵn | Có | Có | Có | Có | Có | Có |
| Video hướng dẫn | Không có | Không có | Có | Không có | Có | Không có |
| BXH món ăn | Có | Có | Không có | Không có | Không có | Có |

## 1.4. Mô tả chức năng

1. **Tài khoản người dùng**

* **Đăng ký / Đăng nhập:** Cho phép người dùng tạo tài khoản mới bằng email/số điện thoại hoặc đăng nhập vào hệ thống. Để tiện lợi cũng có thể cho người dùng mới, tích hợp thêm các cách đăng nhập qua Google, Facebook.
* **Phân quyền người dùng:** Hệ thống sẽ có các nhóm quyền hạn khác nhau để đảm bảo vận hành và kiểm duyệt hiệu quả. User**:** Quyền cơ bản (đăng công thức, like, comment, follow). Admin**:** Quyền cao nhất (quản lý người dùng, quản lý nội dung, xem báo cáo).
* **Quản lý hồ sơ cá nhân (Profile):** Mỗi người dùng có một trang cá nhân riêng, nơi có thể cập nhật profile như ảnh đại diện , giới thiệu bản thân, xem lại các công thức đã đăng, và truy cập nhanh danh sách các món ăn đã lưu vào mục "Yêu thích".

1. **Công thức nấu ăn**

* **Tạo công thức:** Cung cấp công cụ đầy đủ để người dùng có thể tạo các công thức của riêng mình. Giao diện tạo/sửa công thức cần trực quan, dễ sử dụng.
* **Hiển thị & Sắp xếp:** Danh sách các công thức có thể được duyệt và sắp xếp theo nhiều tiêu chí khác nhau như: công thức mới nhất, công thức phổ biến nhất (nhiều lượt xem/like), hoặc theo các chuyên mục ẩm thực.
* **Chi tiết công thức:** Chi tiết của một công thức là nơi hiển thị đầy đủ thông tin nhất, bao gồm: tên món ăn, mô tả, danh sách nguyên liệu , các bước thực hiện chi tiết (có thể kèm ảnh minh họa cho từng bước), thời gian chuẩn bị, thời gian nấu, độ khó, và media (hình ảnh/video).
* **Phân loại :** Mỗi công thức sẽ được gắn vào các danh mục (category) được định nghĩa trước để người dùng dễ dàng khám phá. Ví dụ: Món khai vị, Món chính, Món tráng miệng, Món chay, Đồ uống…
* **Tìm kiếm (Search):** Chức năng tìm kiếm mạnh mẽ cho phép người dùng tìm công thức dựa trên: tên món ăn, một hoặc nhiều nguyên liệu chính.

1. **Gợi ý, đề xuất**

* **Gợi ý công thức phổ biến:** Dựa trên các dữ liệu về lượt xem, lượt thích, và lượt lưu, hệ thống sẽ đề xuất các công thức đang là "hot trend" cho người dùng.
* **Gợi ý dựa trên nguyên liệu:** Tính năng cho phép người dùng nhập vào các nguyên liệu họ đang có sẵn, và ứng dụng sẽ gợi ý các công thức có thể chế biến từ những nguyên liệu đó.

1. **Bảng xếp hạng & Tương tác**

* **Bảng xếp hạng:** Hiển thị danh sách các món ăn được yêu thích nhất theo tuần/tháng dựa trên lượt thích, lượt lưu hoặc lượt xem.
* **Theo dõi (Follow):** Tính năng cho phép người dùng theo dõi các người dùng khác mà họ yêu thích. Bảng tin (news feed) của người dùng có thể sẽ ưu tiên hiển thị công thức từ những người họ theo dõi.
* **Lưu công thức:** Cho phép người dùng lưu lại các công thức họ muốn thử sau này vào một bộ sưu tập cá nhân, dễ dàng truy cập lại khi cần.

# CHƯƠNG 2: Phân tích thiết kế hệ thống

## 2.1. Biểu đồ ca sử dụng

### 2.1.1. Biểu đồ Quản lý người dùng

A screenshot of a chat

AI-generated content may be incorrect.

**Hình 2. 1.** Use case Quản lý người dùng

### 2.1.2. Biểu đồ Quản lí công thức nấu ăn

A black background with white ovals

AI-generated content may be incorrect.

**Hình 2. 2.** Use case Quản lý công thức nấu ăn

### 2.1.3. Biểu đồ Tìm kiếm công thức

A black background with white ovals

AI-generated content may be incorrect.

**Hình 2. 3.** Use case Tìm kiếm công thức

### 2.1.4. Biểu đồ Gợi ý công thức

A black and white image of two ovals

AI-generated content may be incorrect.

**Hình 2. 4.** Use case Gợi ý/ đề xuất công thức

### 2.1.5. Biểu đồ Tương tác người dùng

A group of white ovals with black text

AI-generated content may be incorrect.

**Hình 2. 5.** Use case Tương tác người dùng

## 2.2. Biểu đồ hoạt động

### 2.2.1. Biểu đồ hoạt động Đăng nhập

A black background with white squares

AI-generated content may be incorrect.

**Hình 2. 6.** Activity chức năng đăng nhập

### 2.2.2. Biểu đồ hoạt động Tạo công thức

A screenshot of a computer

AI-generated content may be incorrect.

**Hình 2. 7.** Activity chức năng tạo công thức

### 2.2.3. Biểu đồ hoạt động Tìm kiếm công thức

A group of white rectangular objects

AI-generated content may be incorrect.

**Hình 2. 8.** Activity chức năng tìm kiếm công thức

### 2.2.4. Biểu đồ hoạt động Gợi ý công thức theo nguyên liệu

A black screen with white text

AI-generated content may be incorrect.

**Hình 2. 9.** Activity chức năng gợi ý công thức theo nguyên liệu

### *2.2.5.* Biểu đồ hoạt động Theo dõi người dùng

A screenshot of a cell phone

AI-generated content may be incorrect.

**Hình 2. 10.** Activity chức năng theo dõi người dùng

## 2.3. Biểu đồ tuần tự

### 2.3.1. Biểu đồ tuần tự Tạo công thức

A black screen with white text

AI-generated content may be incorrect.

**Hình 2. 11.** Sequence chức năng tạo công thức

### 2.3.2. Biểu đồ tuần tự Đăng ký tài khoản

A black background with white lines

AI-generated content may be incorrect.

**Hình 2. 12.** Sequence chức năng đăng ký tài khoản

### 2.3.3. Biểu đồ tuần tự tìm kiếm công thức

A screenshot of a computer

AI-generated content may be incorrect.

**Hình 2. 13.** Sequence chức năng tìm kiếm

### 2.3.4. Biểu đồ tuần tự Gợi ý công thức theo nguyên liệu

A screenshot of a computer

AI-generated content may be incorrect.

**Hình 2. 14.** Sequence chức năng gợi ý theo nguyên liệu

### 2.3.5. Biểu đồ tuần tự Lưu công thức

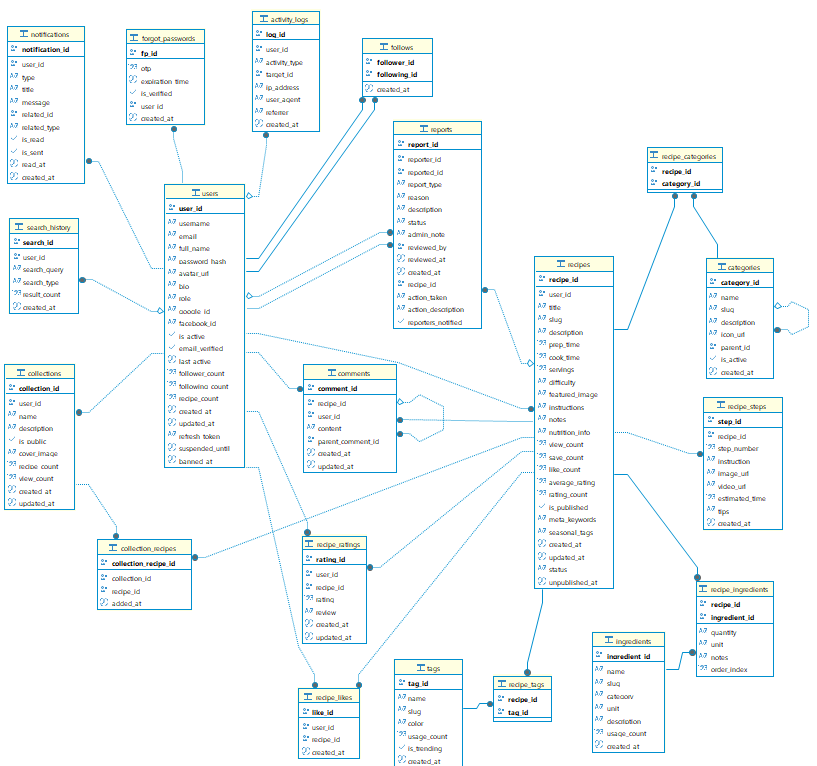
A black screen with white text

AI-generated content may be incorrect.

**Hình 2. 15.** Sequence chức năng lưu công thức

## 2.4. Thiết kế cơ sở dữ liệu

**2.4.1. Sơ đồ cơ sở dữ liệu tổng quan**

****

**Hình 2. 18.** CSDL tổng quan

### 2.4.2. Danh sách các bảng trong cơ sở dữ liệu

#### 2.4.2.1. Bảng Users (Người dùng)

**Bảng 2. 2.** Bảng Users

|  |  |  |
| --- | --- | --- |
| **Tên cột** | **Kiểu dữ liệu** | **Ghi chú** |
| user\_id | uuid | Mã người dùng (khóa chính) |
| username | varchar(100) | Tên đăng nhập (duy nhất) |
| email | varchar(255) | Địa chỉ email (duy nhất) |
| full\_name | varchar(255) | Họ và tên đầy đủ |
| password\_hash | varchar(255) | Mật khẩu đã mã hóa |
| avatar\_url | varchar(1000) | Đường dẫn ảnh đại diện |
| bio | text | Tiểu sử, giới thiệu bản thân |
| role | varchar(50) | Vai trò (USER hoặc ADMIN) |
| google\_id | varchar(100) | ID tài khoản Google (đăng nhập Google) |
| facebook\_id | varchar(100) | ID tài khoản Facebook (đăng nhập Facebook) |
| is\_active | boolean | Trạng thái hoạt động (mặc định: true) |
| email\_verified | boolean | Email đã xác thực chưa (mặc định: false) |
| last\_active | timestamp | Thời gian hoạt động gần nhất |
| follower\_count | integer | Số người theo dõi |
| following\_count | integer | Số người đang theo dõi |
| recipe\_count | integer | Số công thức đã tạo |
| refresh\_token | varchar(500) | Token làm mới phiên đăng nhập |
| created\_at | timestamp | Thời gian tạo tài khoản |
| updated\_at | timestamp | Thời gian cập nhật cuối |
| suspended\_until | timestamp | Hạn tạm khoá |
| banned\_at | timestamp | Thời gian bị cấm |

#### 2.4.2.2. Bảng Recipes (Công thức nấu ăn)

**Bảng 2. 3.** Bảng Recipes

|  |  |  |
| --- | --- | --- |
| **Tên cột** | **Kiểu dữ liệu** | **Ghi chú** |
| recipe\_id | uuid | Mã công thức (khóa chính) |
| user\_id | uuid | Mã người tạo (khóa ngoại → users) |
| title | varchar(255) | Tiêu đề công thức |
| slug | varchar(255) | Đường dẫn URL (duy nhất) |
| description | text | Mô tả chi tiết công thức |
| prep\_time | integer | Thời gian chuẩn bị (phút) |
| cook\_time | integer | Thời gian nấu (phút) |
| servings | integer | Số người ăn (1-100) |
| difficulty | varchar(50) | Độ khó (EASY, MEDIUM, HARD, EXPERT) |
| featured\_image | varchar(1000) | Ảnh đại diện công thức |
| instructions | text | Hướng dẫn thực hiện tổng quát |
| notes | text | Ghi chú thêm |
| nutrition\_info | text | Thông tin dinh dưỡng |
| view\_count | integer | Số lượt xem |
| save\_count | integer | Số lượt lưu |
| like\_count | integer | Số lượt thích |
| average\_rating | numeric(3,2) | Điểm đánh giá trung bình (0-5) |
| rating\_count | integer | Số lượt đánh giá |
| is\_published | boolean | Đã xuất bản chưa (mặc định: false) |
| is\_featured | boolean | Công thức nổi bật (mặc định: false) |
| meta\_keywords | varchar(255) | Từ khóa SEO |
| seasonal\_tags | varchar(255) | Thẻ theo mùa |
| created\_at | timestamp | Thời gian tạo |
| updated\_at | timestamp | Thời gian cập nhật cuối |
| status | varchar | Trạng thái |
| unpublished\_at | timestamp | Thời gian gỡ bài |

#### 2.4.2.3. Bảng Recipe\_Steps (Các bước thực hiện)

**Bảng 2. 4.** Bảng Recipe\_Steps

|  |  |  |
| --- | --- | --- |
| **Tên cột** | **Kiểu dữ liệu** | **Ghi chú** |
| step\_id | uuid | Mã bước thực hiện (khóa chính) |
| recipe\_id | uuid | Mã công thức (khóa ngoại → recipes) |
| step\_number | integer | Số thứ tự bước (≥1) |
| instruction | text | Nội dung hướng dẫn bước |
| image\_url | varchar(255) | Ảnh minh họa bước |
| video\_url | varchar(255) | Video hướng dẫn bước |
| estimated\_time | integer | Thời gian ước tính cho bước (phút) |
| tips | text | Mẹo vặt cho bước này |
| created\_at | timestamp | Thời gian tạo |

#### 2.4.2.4. Bảng Ingredients (Nguyên liệu)

**Bảng 2. 5.** Bảng Ingredients

|  |  |  |
| --- | --- | --- |
| **Tên cột** | **Kiểu dữ liệu** | **Ghi chú** |
| ingredient\_id | uuid | Mã nguyên liệu (khóa chính) |
| name | varchar(255) | Tên nguyên liệu |
| slug | varchar(255) | Đường dẫn URL (duy nhất) |
| category | varchar(100) | Danh mục nguyên liệu |
| unit | varchar(50) | Đơn vị đo lường mặc định |
| description | text | Mô tả nguyên liệu |
| usage\_count | integer | Số lần được sử dụng |
| created\_at | timestamp | Thời gian tạo |

#### 2.4.2.6. Bảng Recipe\_Ingredients (Nguyên liệu của công thức)

**Bảng 2. 6.** Bảng Recipe\_Ingredients

|  |  |  |
| --- | --- | --- |
| **Tên cột** | **Kiểu dữ liệu** | **Ghi chú** |
| recipe\_id | uuid | Mã công thức (khóa chính, khóa ngoại → recipes) |
| ingredient\_id | uuid | Mã nguyên liệu (khóa chính, khóa ngoại → ingredients) |
| quantity | varchar(50) | Khối lượng/số lượng |
| unit | varchar(50) | Đơn vị đo (gram, ml, thìa, ...) |
| notes | text | Ghi chú cho nguyên liệu này |
| order\_index | integer | Thứ tự hiển thị |

#### 2.4.2.7. Bảng Categories (Danh mục)

**Bảng 2. 7.** Bảng Categories

|  |  |  |
| --- | --- | --- |
| **Tên cột** | **Kiểu dữ liệu** | **Ghi chú** |
| category\_id | uuid | Mã danh mục (khóa chính) |
| name | varchar(255) | Tên danh mục |
| slug | varchar(255) | Đường dẫn URL (duy nhất) |
| description | text | Mô tả danh mục |
| icon\_url | varchar(255) | Đường dẫn icon danh mục |
| parent\_id | uuid | Mã danh mục cha (khóa ngoại → categories) |
| is\_active | boolean | Trạng thái kích hoạt (mặc định: true) |
| created\_at | timestamp | Thời gian tạo |

#### 2.4.2.8. Bảng Recipe\_Categories (Danh mục của công thức)

**Bảng 2. 8.** Bảng Recipe\_Categories

|  |  |  |
| --- | --- | --- |
| **Tên cột** | **Kiểu dữ liệu** | **Ghi chú** |
| recipe\_id | uuid | Mã công thức (khóa chính, khóa ngoại → recipes) |
| category\_id | uuid | Mã danh mục (khóa chính, khóa ngoại → categories) |

#### 2.4.2.9. Bảng Tags (Thẻ gắn)

**Bảng 2. 9.** Bảng Tags

|  |  |  |
| --- | --- | --- |
| **Tên cột** | **Kiểu dữ liệu** | **Ghi chú** |
| tag\_id | uuid | Mã thẻ (khóa chính) |
| name | varchar(100) | Tên thẻ |
| slug | varchar(100) | Đường dẫn URL (duy nhất) |
| color | varchar(20) | Mã màu hiển thị |
| usage\_count | integer | Số lần được sử dụng |
| is\_trending | boolean | Thẻ đang thịnh hành (mặc định: false) |
| created\_at | timestamp | Thời gian tạo |

#### 2.4.2.10. Bảng Recipe\_Tags (Thẻ của công thức)

**Bảng 2. 10.** Bảng Recipe\_Tags

|  |  |  |
| --- | --- | --- |
| **Tên cột** | **Kiểu dữ liệu** | **Ghi chú** |
| recipe\_id | uuid | Mã công thức (khóa chính, khóa ngoại → recipes) |
| tag\_id | uuid | Mã thẻ (khóa chính, khóa ngoại → tags) |

#### 2.4.2.11. Bảng Recipe\_Ratings (Đánh giá công thức)

**Bảng 2. 11.** Bảng Recipe\_Ratings

|  |  |  |
| --- | --- | --- |
| **Tên cột** | **Kiểu dữ liệu** | **Ghi chú** |
| rating\_id | uuid | Mã đánh giá (khóa chính) |
| user\_id | uuid | Mã người đánh giá (khóa ngoại → users) |
| recipe\_id | uuid | Mã công thức (khóa ngoại → recipes) |
| rating | integer | Điểm đánh giá (1-5 sao) |
| review | text | Nội dung bình luận đánh giá |
| created\_at | timestamp | Thời gian tạo |
| updated\_at | timestamp | Thời gian cập nhật |

#### 2.4.2.12. Bảng Recipe\_Likes (Lượt thích công thức)

**Bảng 2. 12.** Bảng Recipe\_Likes

|  |  |  |
| --- | --- | --- |
| **Tên cột** | **Kiểu dữ liệu** | **Ghi chú** |
| user\_id | uuid | Mã người dùng (khóa chính, khóa ngoại → users) |
| recipe\_id | uuid | Mã công thức (khóa chính, khóa ngoại → recipes) |
| created\_at | timestamp | Thời gian thích |
| like\_id | uuid | Mã lượt thích |

#### 2.4.2.13. Bảng Collections (Bộ sưu tập công thức)

**Bảng 2. 13.** Bảng Collections

|  |  |  |
| --- | --- | --- |
| **Tên cột** | **Kiểu dữ liệu** | **Ghi chú** |
| collection\_id | uuid | Mã bộ sưu tập (khóa chính) |
| user\_id | uuid | Mã người tạo (khóa ngoại → users) |
| name | varchar(255) | Tên bộ sưu tập |
| description | text | Mô tả bộ sưu tập |
| is\_public | boolean | Công khai hay riêng tư (mặc định: true) |
| cover\_image | varchar(255) | Ảnh bìa bộ sưu tập |
| recipe\_count | integer | Số công thức trong bộ sưu tập |
| view\_count | integer | Số lượt xem |
| created\_at | timestamp | Thời gian tạo |
| updated\_at | timestamp | Thời gian cập nhật |

#### 2.4.2.14. Bảng Collection\_Recipes (Công thức trong bộ sưu tập)

**Bảng 2. 14.** Bảng Collection\_Recipes

|  |  |  |
| --- | --- | --- |
| **Tên cột** | **Kiểu dữ liệu** | **Ghi chú** |
| collection\_id | uuid | Mã bộ sưu tập (khóa chính, khóa ngoại → collections) |
| recipe\_id | uuid | Mã công thức (khóa chính, khóa ngoại → recipes) |
| added\_at | timestamp | Thời gian thêm vào bộ sưu tập |
| collection\_recipe\_id | uuid | Mã chi tiết |

#### 2.4.2.15. Bảng Comments (Bình luận)

**Bảng 2. 15.** Bảng Comments

|  |  |  |
| --- | --- | --- |
| **Tên cột** | **Kiểu dữ liệu** | **Ghi chú** |
| comment\_id | uuid | Mã bình luận (khóa chính) |
| recipe\_id | uuid | Mã công thức (khóa ngoại → recipes) |
| user\_id | uuid | Mã người bình luận (khóa ngoại → users) |
| content | text | Nội dung bình luận |
| parent\_comment\_id | uuid | Mã bình luận cha (trả lời bình luận, khóa ngoại → comments) |
| created\_at | timestamp | Thời gian tạo |
| updated\_at | timestamp | Thời gian cập nhật |

#### 2.4.2.16. Bảng Follows (Theo dõi)

**Bảng 2. 16.** Bảng Follows

|  |  |  |
| --- | --- | --- |
| **Tên cột** | **Kiểu dữ liệu** | **Ghi chú** |
| follower\_id | uuid | Mã người theo dõi (khóa chính, khóa ngoại → users) |
| following\_id | uuid | Mã người được theo dõi (khóa chính, khóa ngoại → users) |
| created\_at | timestamp | Thời gian bắt đầu theo dõi |

#### 2.4.2.17. Bảng Notifications (Thông báo)

**Bảng 2. 17.** Bảng Notifications

|  |  |  |
| --- | --- | --- |
| **Tên cột** | **Kiểu dữ liệu** | **Ghi chú** |
| notification\_id | uuid | Mã thông báo (khóa chính) |
| user\_id | uuid | Mã người nhận (khóa ngoại → users) |
| type | varchar(50) | Loại thông báo (FOLLOW, LIKE, RATING, ...) |
| title | varchar(255) | Tiêu đề thông báo |
| message | text | Nội dung thông báo |
| related\_id | uuid | Mã đối tượng liên quan |
| related\_type | varchar(50) | Loại đối tượng (recipe, user, comment, collection) |
| is\_read | boolean | Đã đọc chưa (mặc định: false) |
| is\_sent | boolean | Đã gửi chưa (mặc định: false) |
| read\_at | timestamp | Thời gian đọc |
| created\_at | timestamp | Thời gian tạo |

#### 2.4.2.18. Bảng Activity\_Logs (Hoạt động)

**Bảng 2. 18**. Bảng Activity\_Logs

|  |  |  |
| --- | --- | --- |
| **Tên cột** | **Kiểu dữ liệu** | **Ghi chú** |
| log\_id | uuid | Mã nhật ký (khóa chính) |
| user\_id | uuid | Mã người dùng (khóa ngoại → users) |
| activity\_type | varchar(50) | Loại hoạt động (VIEW, CREATE, UPDATE, DELETE, LOGIN, ...) |
| target\_id | uuid | Mã đối tượng được thao tác |
| ip\_address | varchar | Địa chỉ IP |
| user\_agent | varchar(255) | Thông tin trình duyệt |
| referrer | varchar(255) | Nguồn truy cập |
| created\_at | timestamp | Thời gian thực hiện |

#### 2.4.2.19. Bảng Search\_History (Lịch sử tìm kiếm)

**Bảng 2. 19**. Bảng Search\_History

|  |  |  |
| --- | --- | --- |
| **Tên cột** | **Kiểu dữ liệu** | **Ghi chú** |
| search\_id | uuid | Mã tìm kiếm (khóa chính) |
| user\_id | uuid | Mã người dùng (khóa ngoại → users) |
| search\_query | text | Từ khóa tìm kiếm |
| search\_type | varchar(50) | Loại tìm kiếm |
| result\_count | integer | Số kết quả tìm được |
| created\_at | timestamp | Thời gian tìm kiếm |

#### 2.4.2.20. Bảng Reports (Báo cáo vi phạm)

**Bảng 2. 20.** Bảng Reports

|  |  |  |
| --- | --- | --- |
| **Tên cột** | **Kiểu dữ liệu** | **Ghi chú** |
| report\_id | uuid | Mã báo cáo (khóa chính) |
| reporter\_id | uuid | Mã người báo cáo (khóa ngoại → users) |
| reported\_id | uuid | Mã người bị báo cáo (khóa ngoại → users) |
| recipe\_id | uuid | Mã công thức bị báo cáo (khóa ngoại → recipes) |
| report\_type | varchar(50) | Loại vi phạm (SPAM, INAPPROPRIATE, COPYRIGHT, ...) |
| reason | varchar(255) | Lý do báo cáo |
| description | text | Mô tả chi tiết |
| status | varchar(50) | Trạng thái xử lý (PENDING, REVIEWING, RESOLVED, ...) |
| admin\_note | text | Ghi chú của quản trị viên |
| reviewed\_by | uuid | Mã người xem xét (khóa ngoại → users) |
| reviewed\_at | timestamp | Thời gian xem xét |
| created\_at | timestamp | Thời gian tạo báo cáo |

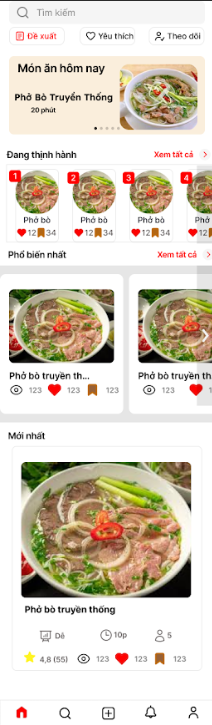
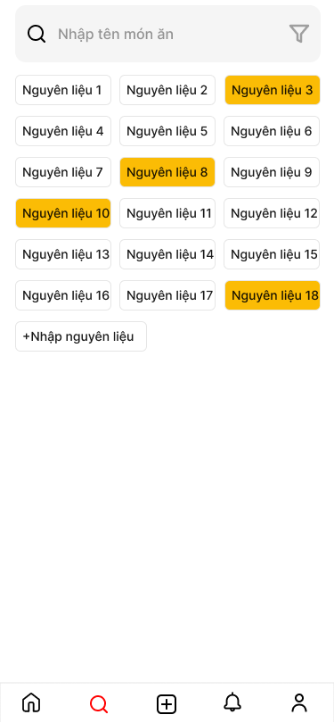
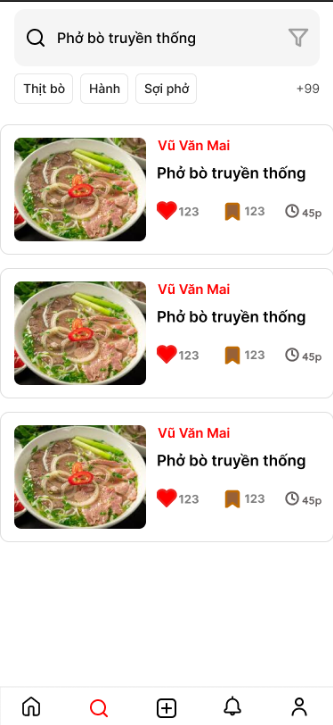
#### 2.4.2.21. Bảng Forgot\_Passwords (Quên mật khẩu)

**Bảng 2. 21.** Bảng Forgot\_Passwords

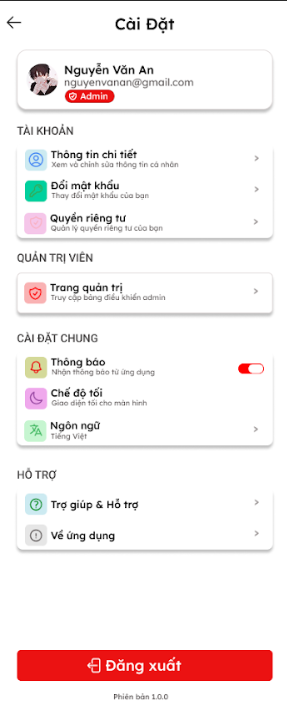
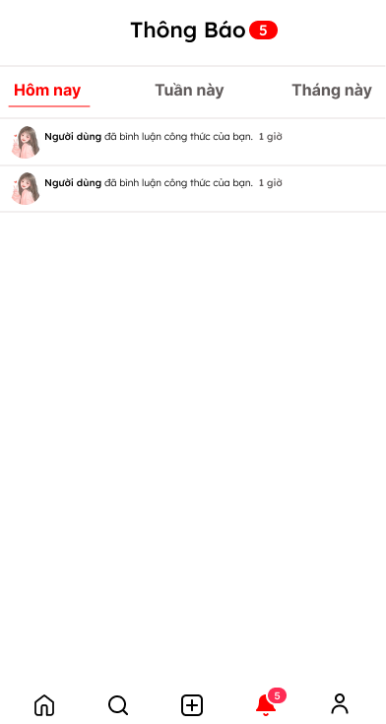
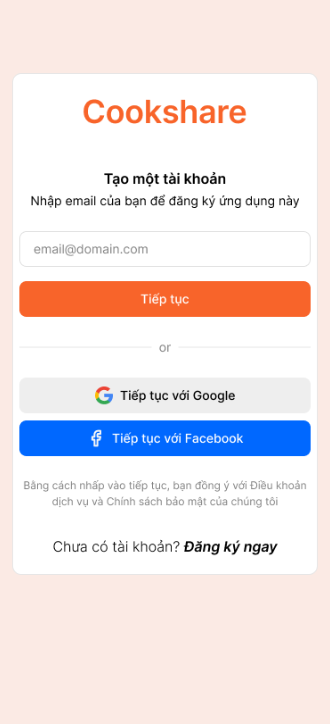
|  |  |  |
| --- | --- | --- |
| **Tên cột** | **Kiểu dữ liệu** | **Ghi chú** |
| fp\_id | uuid | Mã yêu cầu (khóa chính) |
| user\_id | uuid | Mã người dùng (khóa ngoại → users) |
| otp | integer | Mã OTP xác thực |
| expiration\_time | timestamp | Thời gian hết hạn OTP |
| is\_verified | boolean | Đã xác thực chưa (mặc định: false) |
| created\_at | timestamp | Thời gian tạo yêu cầu |

## 2.5. Thiết kế giao diện

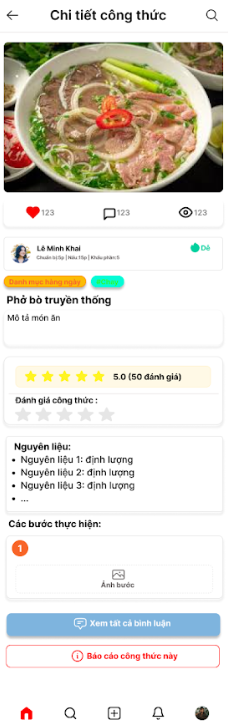
### 2.5.1. Giao diện người dùng

**** **** ****

**Hình 2. 19.** Giao diện trang chủ, trang tìm kiếm, kết quả tìm kiếm

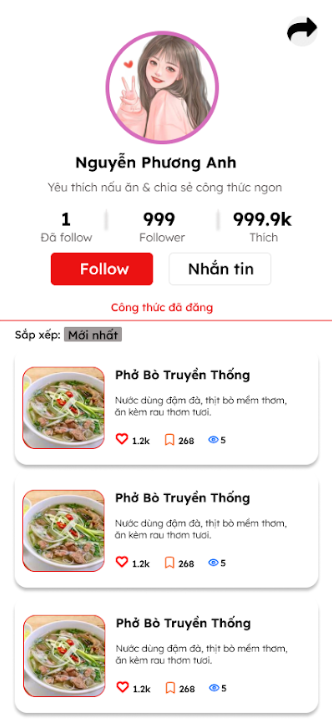
****

**Hình 2. 20.** Giao diện trang đăng nhập, thống báo, cài đặt

**Hình 2. 21.** Giao diện trang tạo công thức, chi tiết công thức

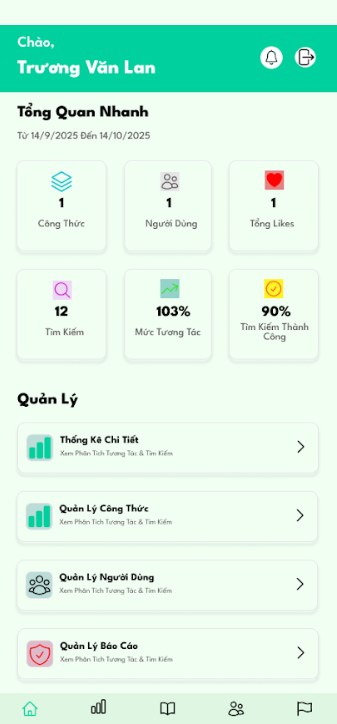
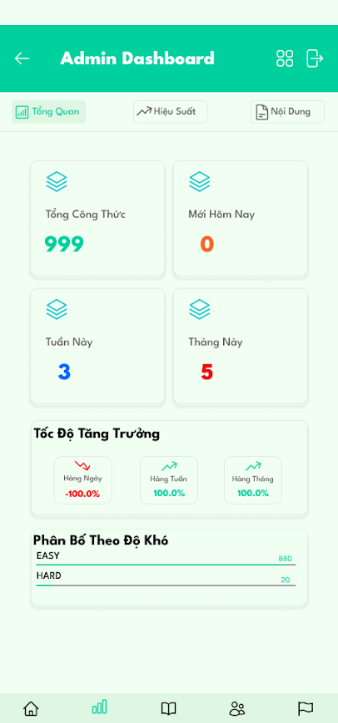
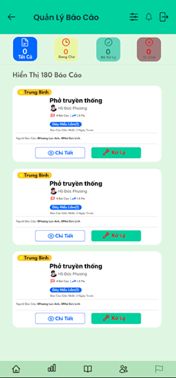
A screenshot of a phone

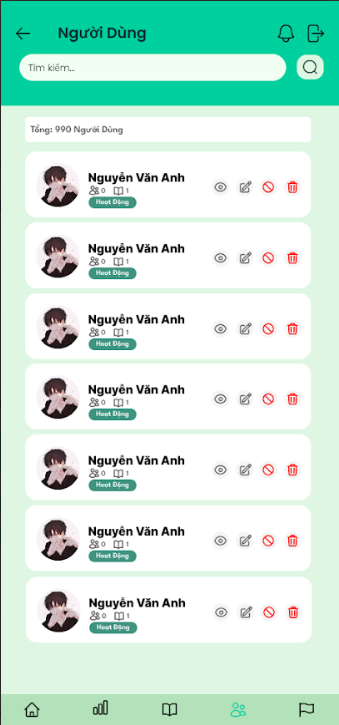
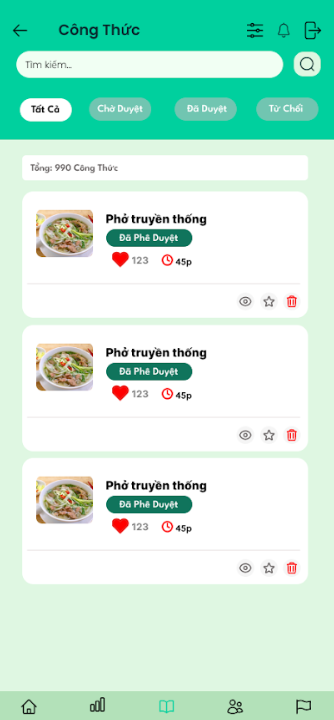
AI-generated content may be incorrect. **** A screenshot of a phone

AI-generated content may be incorrect.

**Hình 2. 22.** Giao diện trang cá nhân, người dùng, công thức đã lưu

### 2.5.2. Giao diện người dùng quản trị

**** **** 

**** ****

**Hình 2. 23.** Giao diện của người dùng quản trị

# CHƯƠNG 3: TRIỂN KHAI DỰ ÁN

## 3.1. Công nghệ backend

### 3.1.1. Các công nghệ sử dụng phổ biến cho backend

1. **Node.js**

Node.js là một môi trường chạy JavaScript phía server, dựa trên engine V8 của Google Chrome. Node.js áp dụng mô hình I/O bất đồng bộ, non-blocking event-driven, rất phù hợp cho các ứng dụng web real-time và microservices. Với npm (Node Package Manager), Node.js sở hữu hệ sinh thái thư viện lớn nhất hiện nay, hỗ trợ đa dạng nhu cầu phát triển.

**Ưu điểm:**

* Sử dụng JavaScript cho cả frontend và backend → thuận lợi cho full-stack.
* Mô hình non-blocking I/O → rất mạnh trong xử lý real-time (chat, streaming, game online).
* Cộng đồng đông đảo, thư viện npm khổng lồ.
* Dễ dàng triển khai theo mô hình microservices.
* Học nhanh, cú pháp đơn giản, phù hợp cho startup cần ra sản phẩm nhanh.

**Nhược điểm:**

* Xử lý CPU-bound kém (các tác vụ tính toán nặng có thể làm nghẽn event loop).
* Chất lượng package npm không đồng đều, dễ gặp vấn đề bảo mật.
* Debug và quản lý ứng dụng lớn có thể phức tạp nếu không có kiến trúc rõ ràng.

1. **Django**

Django là framework web mã nguồn mở viết bằng Python, nổi tiếng với triết lý **“batteries included”** – cung cấp đầy đủ công cụ cần thiết từ ORM, Auth, Form cho đến Admin Panel. Django hướng tới phát triển nhanh, bảo mật cao và dễ bảo trì, rất phù hợp cho các ứng dụng CRUD, SaaS và tích hợp AI/ML.

**Ưu điểm:**

* Phát triển cực nhanh nhờ nhiều tính năng tích hợp sẵn (ORM, Admin, Auth, Security).
* Cấu trúc rõ ràng, tuân thủ nguyên tắc DRY (Don’t Repeat Yourself).
* Bảo mật tốt: chống SQL Injection, XSS, CSRF… được tích hợp sẵn.
* Cộng đồng lớn trong cả web lẫn AI/ML → dễ tích hợp với các thư viện Machine Learning, Data Science.
* Dễ học, dễ đọc code, phù hợp cho team nhỏ, startup, sinh viên.

**Nhược điểm:**

* Python có hiệu năng thấp hơn JS/Java → không phù hợp cho ứng dụng real-time hoặc hệ thống yêu cầu hiệu năng cực cao.
* Scale ngang được, nhưng không mạnh bằng Node.js hoặc Spring Boot.
* Một số giới hạn khi xây dựng microservices hoặc hệ thống phân tán lớn.

1. **Spring Boot**

Spring Boot là framework Java giúp phát triển ứng dụng nhanh hơn trên nền tảng Spring. Nó cung cấp các **starter** và cấu hình mặc định sẵn, giảm bớt boilerplate code, nhưng vẫn giữ được sức mạnh và sự linh hoạt của hệ sinh thái Spring. Spring Boot đặc biệt phù hợp với các ứng dụng enterprise, tài chính, và hệ thống lớn cần hiệu năng và bảo mật cao.

**Ưu điểm:**

* Hiệu năng cao, quản lý multi-thread tốt, thích hợp cho ứng dụng lớn, chịu tải nặng.
* Hệ sinh thái Spring Cloud mạnh mẽ → hỗ trợ microservices, distributed system.  
  Spring Security tích hợp sẵn → bảo mật cực mạnh (OAuth2, JWT, LDAP, SSO…).
* Khả năng mở rộng tuyệt vời, phù hợp cho enterprise-scale.
* Tích hợp dễ dàng với các công nghệ Java khác (Hibernate, Kafka, JPA).

**Nhược điểm:**

* Học khó hơn so với Node.js/Django → yêu cầu kiến thức Java + Spring ecosystem.
* Code verbose hơn, thời gian phát triển lâu hơn.
* Quá nặng cho các ứng dụng nhỏ hoặc prototype nhanh.

### 3.1.2. Bảng so sánh công nghệ

**Bảng 3. 1.** Bảng so sánh công nghệ Backend

|  |  |  |  |
| --- | --- | --- | --- |
| **Tiêu chí** | **Node.js** | **Django** | **Spring Boot** |
| Ngôn ngữ | JavaScript/  TypeScript | Python | Java/Kotlin |
| Hiệu năng | Rất mạnh real-time, yếu CPU-heavy | Trung bình | Rất mạnh, tối ưu CPU & multi-thread |
| Bảo mật | Trung bình, phụ thuộc libary | Tốt, build-in | Rất mạnh, Spring Security |
| Hệ sinh thái | npm, Express, NestJS | Django REST, AI/ML | Spring Cloud, Hibernate |
| Khả năng tiếp cận | Dễ, phổ biến cho full-stack | Dễ, code ngắn gọn | Khó hơn, yêu cầu Java + Spring |
| Dự án phù hợp | Startup, chat app, realtime app | Hệ thống cần AI/ML (recommendation, chatbot), Website thương mại điện tử vừa & nhỏ | Ngân hàng, fintech, Thương mại điện tử quy mô lớn |

### 3.1.3. Kết luận, lựa chọn công nghệ

Sau khi đánh giá và so sánh với Node.js và Django, nhóm quyết định lựa chọn Spring Boot làm nền tảng phát triển Backend chính nhờ các ưu điểm phù hợp với yêu cầu dự án:

* Nền tảng vững chắc: Kế thừa sự ổn định, hiệu năng mạnh mẽ và tính bảo mật cao từ hệ sinh thái Java.
* Khả năng mở rộng và tích hợp: Kiến trúc linh hoạt, hỗ trợ kết nối liền mạch với các công nghệ hiện đại (WebSocket, Kafka, Redis) để giải quyết bài toán tương tác thời gian thực và xử lý đồng thời lượng lớn người dùng.
* Hỗ trợ phát triển toàn diện: Hưởng lợi từ hệ thống tài liệu phong phú và cộng đồng lớn, đảm bảo khả năng bảo trì và mở rộng tính năng lâu dài.
* Tối ưu hóa nguồn lực: Tận dụng kinh nghiệm sẵn có của đội ngũ phát triển với Spring Boot giúp rút ngắn thời gian triển khai, đồng thời đảm bảo chất lượng và hiệu suất mã nguồn.

## 3.2. Công nghệ frontend

### 3.2.1. Các công nghệ sử dụng phổ biến cho frontend

1. **React Native**

React Native, được phát triển bởi Facebook, đã nhanh chóng trở thành một trong những framework front-end được săn đón nhất để xây dựng các ứng dụng di động đa nền tảng. Các nhà phát triển sử dụng JavaScript và React để xây dựng các ứng dụng tương tự native cho nền tảng iOS và Android, sử dụng kiến ​​trúc dựa trên thành phần, giúp tái sử dụng mã nguồn và phát triển nhanh chóng biến React Native trở thành một giải pháp hấp dẫn đối với cả nhà phát triển và doanh nghiệp. Với hệ sinh thái thư viện và cộng đồng hỗ trợ ngày càng mở rộng, React Native tiếp tục phát triển, cung cấp cho các nhà phát triển những công cụ và tính năng mới mạnh mẽ để tạo ra các ứng dụng di động mạnh mẽ.

**Ưu điểm:**

* Cơ sở mã duy nhất cho IOS và Android
* Cộng đồng lớn, nhiều plugin mature (media, auth, push).
* Tốc độ phát triển nhanh với tool hỗ trợ như **Expo** (hot reload, preview trên thiết bị).
* Dễ tái sử dụng logic giữa web (React) và mobile.

**Nhược điểm:**

* Một số tác vụ nặng (animation phức tạp, xử lý video/encode) gặp overhead do bridge JS ↔ native.
* Có thể cần viết module native (Android/iOS) khi plugin không đáp ứng yêu cầu.

1. **Flutter**

Flutter, một bộ công cụ giao diện người dùng nguồn mở của Google, đã nhanh chóng tạo nên làn sóng trong cộng đồng phát triển ứng dụng di động. Sử dụng ngôn ngữ lập trình Dart và có kiến ​​trúc phân lớp với các tiện ích tùy chỉnh để lặp lại nhanh hơn, Flutter cho phép các nhà phát triển nhanh chóng tạo ra các ứng dụng hiệu suất cao, đẹp mắt cho nhiều nền tảng từ một cơ sở mã. Hơn nữa, tính năng tải lại nóng (hot reload) cho phép xem các thay đổi theo thời gian thực, đồng thời tính năng tải lại nóng (hot reload) giúp quá trình phát triển nhanh hơn và hiệu quả hơn bao giờ hết.

**Ưu điểm:**

* Hiệu năng rất tốt, giao diện mượt, kiểm soát UI chi tiết.
* Hot reload mạnh mẽ, phát triển UI nhanh.
* Đồng nhất giữa Android & iOS, dễ tạo trải nghiệm độc đáo.
* Hệ thống widget phong phú; phù hợp cho demo/ứng dụng cần UI bắt mắt.

**Nhược điểm:**

* Phải học Dart nếu nhóm chưa biết.
* Kích thước binary có thể lớn hơn.
* Một vài plugin chuyên biệt đôi khi ít mature hơn RN, nhưng hệ sinh thái đang phát triển nhanh.

1. **Ionic**

Ionic là một framework front-end từng đoạt giải thưởng, được sử dụng để phát triển các ứng dụng di động lai sử dụng các công nghệ web như HTML, CSS và JavaScript. Framework Ionic sở hữu một bộ thành phần giao diện người dùng toàn diện với hiệu suất tương đương native và tích hợp liền mạch với các framework phổ biến như Angular và React. Hơn nữa, phương pháp tiếp cận độc lập nền tảng của nó cho phép các nhà phát triển nhắm mục tiêu đến nhiều nền tảng chỉ với một cơ sở mã, đồng thời tiết kiệm thời gian và công sức trong quá trình phát triển.

**Ưu điểm:**

* Cơ sở mã duy nhất cho IOS, Android và Web
* Plugin Capacitor hỗ trợ camera, filesystem, push, SQLite.

**Nhược điểm:**

* Chạy trong WebView → trải nghiệm & hiệu năng kém hơn native/Flutter, đặc biệt với animation phức tạp hoặc danh sách nhiều ảnh/video.
* Không phù hợp khi cần xử lý media nặng hoặc hiệu ứng mượt.

### 3.2.2. Bảng so sánh công nghệ

**Bảng 3. 2.** Bảng so sánh công nghệ Fontend

|  |  |  |  |
| --- | --- | --- | --- |
| **Tiêu chí** | **React Native** | **Flutter** | **Ionic** |
| Ngôn ngữ | JavaScript / TypeScript (React) | Dart | HTML, CSS, JavaScript (Angular / React / Vue) |
| Hiệu năng | Tốt, gần native, nhưng có overhead JS ↔ native bridge | Rất mạnh, native-like (engine Skia) | Trung bình, chạy trong WebView nên kém hơn |
| Bảo mật | Trung bình, phụ thuộc vào lib/plugin bảo mật | Tốt, kiểm soát code toàn bộ bằng Dart, ít phụ thuộc bridge | Trung bình, dựa vào bảo mật của WebView & plugin |
| Hệ sinh thái | Lớn (npm, Expo, nhiều plugin mature: media, auth, push) | Đang phát triển nhanh, hệ thống widget phong phú, hỗ trợ đa nền tảng | Rộng, hỗ trợ PWA, Capacitor plugin (camera, filesystem, push, SQLite) |
| Khả năng tiếp cận | |  | | --- | | Dễ, phổ biến nhờ cộng đồng JS/React | | Cần học Dart, dễ nếu quen OOP | Dễ nếu đã biết web dev (HTML/CSS/JS, Angular/React) |
| Dự án phù hợp | Startup, ứng dụng di động phổ biến, cần ra nhanh | Ứng dụng cần UI đẹp, hiệu năng cao, trải nghiệm mượt | Ứng dụng hybrid, PWA, app đơn giản đến trung bình |

### 3.2.3. Kết luận, lựa chọn công nghệ

Sau khi cân nhắc giữa Flutter và Ionic, nhóm quyết định sử dụng React Native làm nền tảng phát triển ứng dụng di động nhờ những ưu điểm phù hợp với nguồn lực và mục tiêu dự án:

* Tận dụng kỹ năng sẵn có: Kế thừa kinh nghiệm về JavaScript và React của đội ngũ, giúp rút ngắn thời gian học tập và đảm bảo chất lượng sản phẩm.
* Hiệu suất phát triển cao: Sự hỗ trợ từ Expo và tính năng Hot Reload giúp tăng tốc độ xây dựng và kiểm thử ứng dụng.
* Tối ưu đa nền tảng: Khả năng tái sử dụng phần lớn mã nguồn cho cả iOS và Android (cũng như Web) giúp tiết kiệm đáng kể công sức phát triển và bảo trì.
* Hệ sinh thái phong phú: Cộng đồng lớn và kho thư viện đa dạng hỗ trợ giải quyết nhanh chóng các yêu cầu kỹ thuật phức tạp.

## 3.3. Cơ sở dữ liệu

### 3.3.1. Các cơ sở dữ liệu phổ biến

1. **MySQL**

MySQL là hệ quản trị cơ sở dữ liệu quan hệ (RDBMS) phổ biến nhất thế giới, mã nguồn mở, được Oracle duy trì. MySQL được dùng nhiều trong các ứng dụng web, startup và hệ thống thương mại điện tử.

**Ưu điểm:**

* Miễn phí, mã nguồn mở, cộng đồng lớn.
* Hỗ trợ tốt cho các ứng dụng web quy mô vừa và nhỏ.
* Tài liệu phong phú, dễ học, dễ cài đặt.
* Tích hợp tốt với hầu hết các ngôn ngữ backend (Java, PHP, Python, Node.js).
* Hiệu năng đọc nhanh, tối ưu cho ứng dụng CRUD.

**Nhược điểm:**

* Khả năng scale ngang còn hạn chế so với NoSQL (MongoDB).
* Không mạnh bằng PostgreSQL về xử lý truy vấn phức tạp.
* Các tính năng cao cấp (ví dụ JSONB, full-text search nâng cao) còn hạn chế.

1. **PostgreSQL**

PostgreSQL là hệ quản trị CSDL quan hệ mã nguồn mở tiên tiến, được xem là đối thủ trực tiếp của MySQL. Nó nổi tiếng với việc tuân thủ chuẩn SQL, hỗ trợ mở rộng, và tính năng mạnh mẽ.

**Ưu điểm:**

* Hỗ trợ đầy đủ chuẩn SQL và ACID → dữ liệu an toàn, đáng tin cậy.
* Hỗ trợ nhiều kiểu dữ liệu hiện đại (JSONB, XML, HSTORE…).
* Truy vấn phức tạp, xử lý dữ liệu lớn và phân tích tốt hơn MySQL.
* Hệ thống mở rộng mạnh: có thể viết hàm bằng nhiều ngôn ngữ (Python, JS…).
* Rất phù hợp cho ứng dụng cần phân tích dữ liệu, AI/ML tích hợp.

**Nhược điểm:**

* Cài đặt và quản trị phức tạp hơn MySQL.
* Hiệu năng insert/update trong một số trường hợp kém MySQL.
* Ít hosting miễn phí so với MySQL.

1. **MongoDB**

MongoDB là hệ quản trị CSDL NoSQL phổ biến nhất hiện nay, dựa trên mô hình document (dữ liệu dạng JSON). Nó rất phù hợp cho ứng dụng real-time, dữ liệu phi cấu trúc, và hệ thống cần mở rộng ngang dễ dàng.

**Ưu điểm:**

* Lưu trữ dữ liệu dạng document → linh hoạt, không cần schema cứng.
* Hiệu năng cao cho ứng dụng real-time, dữ liệu lớn.
* Scale ngang tốt, dễ triển khai hệ thống phân tán.
* Phù hợp cho ứng dụng social media (comment, follow, like).
* Dữ liệu JSON dễ tích hợp với frontend (React, Angular, Vue).

**Nhược điểm**

* Không đảm bảo tính ACID mạnh mẽ như RDBMS (chỉ có ở bản mới với transaction).
* Thiếu khả năng join mạnh như SQL → khó cho báo cáo phức tạp.
* Quản trị phức tạp hơn MySQL/PostgreSQL.

### 3.3.2. Bảng so sánh cơ sở dữ liệu

**Bảng 3. 3.** Bảng so sánh cơ sở dữ liệu

|  |  |  |  |
| --- | --- | --- | --- |
| **Tiêu chí** | **MySQL** | **PostgreSQL** | **MongoDB** |
| Mô hình dữ liệu | Quan hệ (RDBMS) | Quan hệ (RDBMS) nâng cao | Document (NoSQL) |
| Chuẩn SQL | Hỗ trợ tốt, phổ biến | Chuẩn SQL mạnh, mở rộng | Không (dùng query JSON) |
| Kiểu dữ liệu | Cơ bản | Đa dạng (JSONB, GIS, XML…) | JSON, BSON |
| Hiệu năng CRUD | Nhanh, nhẹ | Ổn định, tốt hơn cho truy vấn phức tạp | Rất nhanh với dữ liệu lớn |
| Scale ngang | Hạn chế | Có nhưng phức tạp | Rất mạnh |
| Bảo mật & ACID | Tốt | Rất tốt | Trung bình (trừ bản enterprise) |
| Ứng dụng phù hợp | Web nhỏ và vừa, thương mại điện tử | Ứng dụng phân tích dữ liệu, hệ thống lớn | Mạng xã hội, realtime app, dữ liệu phi cấu trúc |

### 3.3.3. Kết luận, lựa chọn cơ sở dữ liệu

Sau khi phân tích, nhóm quyết định sử dụng PostgreSQL làm nền tảng lưu trữ dữ liệu chính nhờ sự cân bằng giữa tính ổn định và sự linh hoạt:

* Toàn vẹn dữ liệu: Cơ chế ACID mạnh mẽ đảm bảo độ chính xác tuyệt đối cho các mối quan hệ phức tạp (Người dùng - Công thức - Tương tác).
* Hỗ trợ đa dạng: Khả năng xử lý JSONB ưu việt cho phép quản lý dữ liệu phi cấu trúc (metadata, bình luận) linh hoạt như NoSQL ngay trong môi trường SQL.
* Khả năng mở rộng: Hệ sinh thái phong phú và hiệu suất đã được kiểm chứng tạo nền tảng vững chắc cho việc phát triển các tính năng phân tích và gợi ý thông minh trong tương lai.

## 3.4. Lưu trữ ảnh cho hệ thống

### 3.4.1. Các công nghệ phổ biến để lưu trữ hình ảnh

1. **AWS**

Amazon Web Services (AWS) là một nền tảng điện toán đám mây toàn diện và được sử dụng rộng rãi nhất thế giới, cung cấp hơn 200 dịch vụ đầy đủ tính năng từ các trung tâm dữ liệu trên toàn cầu. Về cơ bản, AWS cho phép bạn thuê và sử dụng tài nguyên máy tính, lưu trữ, cơ sở dữ liệu và các dịch vụ công nghệ thông tin khác qua Internet, thay vì phải tự xây dựng và bảo trì cơ sở hạ tầng của riêng mình.

**Ưu điểm:**

* **Độ bền và tính sẵn sàng cao:** Amazon S3 được thiết kế để cung cấp độ bền cao, đảm bảo dữ liệu gần như không bao giờ bị mất.
* **Khả năng mở rộng không giới hạn:** Chúng ta có thể lưu trữ bao nhiêu dữ liệu tùy thích và AWS sẽ tự động mở rộng dung lượng.
* **Tốc độ phân phối nhanh chóng:** Khi kết hợp S3 với dịch vụ CDN Amazon CloudFront, hình ảnh và video sẽ được lưu trữ tại các máy chủ biên trên toàn cầu.
* **Bảo mật mạnh mẽ:** AWS cung cấp nhiều lớp bảo mật, bao gồm mã hóa dữ liệu khi lưu trữ và truyền tải, quản lý quyền truy cập chi tiết (IAM), giúp kiểm soát hoàn toàn ai có thể truy cập vào dữ liệu .

**Nhược điểm:**

* **Chi phí có thể khó dự đoán:** Mặc dù mô hình trả theo dung lượng sử dụng rất linh hoạt, nhưng chi phí truyền dữ liệu ra ngoài có thể trở nên tốn kém nếu ứng dụng có lưu lượng truy cập cao.
* **Độ phức tạp ban đầu:** Đối với người mới bắt đầu, hệ sinh thái AWS với hàng trăm dịch vụ và các tùy chọn cấu hình có thể gây choáng ngợp. Việc thiết lập một hệ thống lưu trữ và phân phối media tối ưu đòi hỏi kiến thức kỹ thuật nhất định.
* **Quản lý chi phí cần sự chủ động:** Nếu không theo dõi và cấu hình các quy tắc quản lý vòng đời dữ liệu một cách hợp lý, chi phí lưu trữ có thể tăng lên không cần thiết khi dữ liệu ngày càng nhiều.

1. **Firebase Storage**

Firebase Storage là dịch vụ lưu trữ đối tượng (hình ảnh, video, file) của nền tảng Firebase của Google. Firebase Storage được xây dựng trên nền tảng của Google Cloud Storage (GCS), có nghĩa là nó thừa hưởng sức mạnh về khả năng mở rộng và bảo mật của Google Cloud, nhưng được "gói" lại trong một giao diện lập trình (SDK) cực kỳ đơn giản và tích hợp sâu với các dịch vụ Firebase khác.

**Ưu điểm:**

* **Cực kỳ dễ sử dụng và tích hợp:** Đây là điểm mạnh lớn nhất của Firebase Storage. Các SDK dành cho Web, iOS, và Android rất trực quan. Việc tải lên một tệp tin chỉ mất vài dòng mã, giúp giảm đáng kể thời gian phát triển.
* **Tích hợp sâu với hệ sinh thái Firebase:** Firebase Authentication: Chúng ta có thể viết các quy tắc bảo mật cực kỳ mạnh mẽ và đơn giản. Việc này trên AWS phức tạp hơn nhiều. Cloud Functions for Firebase: Có thể tự động kích hoạt một hàm xử lý ngay khi có một tệp mới được tải lên.
* **Hoạt động tốt trong điều kiện mạng kém:** SDK của Firebase được thiết kế đặc biệt cho ứng dụng di động, có khả năng tự động tạm dừng và tiếp tục quá trình tải lên/tải xuống khi kết nối mạng bị gián đoạn, đảm bảo trải nghiệm người dùng tốt hơn.
* **Gói miễn phí:** Firebase cung cấp gói miễn phí với 5 GB dung lượng lưu trữ, 1 GB tải xuống mỗi ngày và 50,000 lượt tải lên mỗi tháng, rất phù hợp cho các dự án nhỏ.

**Nhược điểm:**

* Firebase Storage che giấu đi nhiều sự phức tạp của Google Cloud Storage. Điều này giúp nó dễ sử dụng nhưng cũng đồng nghĩa với việc có ít quyền kiểm soát các tính năng nâng cao hơn so với AWS S3
* Mặc dù dễ sử dụng, nhưng nếu ứng dụng có lưu lượng băng thông cực lớn, chi phí có thể sẽ cao hơn một chút so với việc sử dụng trực tiếp AWS S3 hoặc Google Cloud Storage và tự tối ưu mọi thứ.
* Để đạt được hiệu suất CDN thực sự cho người dùng toàn cầu, sẽ cần cấu hình thêm Google Cloud CDN, điều này làm tăng thêm một chút độ phức tạp.
* Việc sử dụng Firebase Storage thường kéo theo việc sẽ muốn sử dụng các dịch vụ khác của Firebase/Google Cloud để tận dụng tối đa sức mạnh của nó. Điều này có thể không lý tưởng nếu muốn xây dựng một hệ thống đa nền tảng (multi-cloud).

1. **Cloudinary**

**Cloudinary** không chỉ đơn thuần là một dịch vụ lưu trữ file như Google Drive hay Dropbox. Nó là một nền tảng quản lý media toàn diện trên nền tảng đám mây, được thiết kế đặc biệt cho các nhà phát triển web và ứng dụng di động. Nó cung cấp một giải pháp trọn gói từ khâu tải lên, lưu trữ, xử lý, tối ưu hóa cho đến phân phối nội dung đến người dùng một cách nhanh chóng và hiệu quả.

**Ưu điểm**

* Cloudinary cung cấp trọn gói từ lưu trữ, xử lý, tối ưu hóa, đến phân phối nội dung qua CDN tốc độ cao, không cần kết hợp nhiều dịch vụ khác nhau.
* Khả năng biến đổi hình ảnh và video qua URL giúp tiết kiệm cực lớn thời gian và công sức.
* Tính năng format auto và quality auto là "vũ khí bí mật", giúp website tải nhanh hơn đáng kể và tiết kiệm băng thông mà không cần can thiệp thủ công.
* Tích hợp CDN sẵn có
* Giao diện quản lý thân thiện (Media Library)

**Nhược điểm**

* Chi phí cao hơn: Chi phí được tính dựa trên một hệ thống "credit" phức tạp, bao gồm dung lượng lưu trữ, băng thông và số lần biến đổi.
* Không phải nơi lưu trữ đa năng: Cloudinary được thiết kế chuyên biệt cho media.
* Vì logic xử lý ảnh/video của nằm hết trong cấu trúc URL của Cloudinary, việc di chuyển sang một nền tảng khác sau này sẽ rất khó khăn và tốn kém.

### 3.4.2. Kết luận lựa chọn công nghệ

Sau khi so sánh với AWS S3 và Cloudinary, nhóm quyết định sử dụng Firebase Storage nhờ các ưu điểm vượt trội phù hợp với đặc thù dự án:

* Tối ưu cho di động: SDK hỗ trợ xử lý mạnh mẽ trong điều kiện mạng yếu, bao gồm tính năng tự động khôi phục (resume) khi quá trình tải lên/xuống bị gián đoạn.
* Triển khai nhanh chóng: Tài liệu rõ ràng và bộ SDK đơn giản giúp rút ngắn đáng kể thời gian phát triển.
* Hệ sinh thái đồng bộ: Tích hợp liền mạch với Firebase Authentication để quản lý phân quyền và Cloud Functions để tự động hóa quy trình xử lý media (nén ảnh, tạo thumbnail) ngay sau khi tải lên.
* Hiệu quả chi phí: Gói miễn phí (5GB lưu trữ, 1GB tải xuống/ngày) đáp ứng tốt nhu cầu vận hành trong giai đoạn MVP và kiểm thử.

# CHƯƠNG 4: KIỂM THỬ ỨNG DỤNG

## 4.1. Công cụ và môi trường kiểm thử

### 4.1.1. Môi trường kiểm thử

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Thành phần** | **Chi tiết** | **Mục đích** |
| 1 | Ứng dụng di động | Thiết bị thật: Điện thoại hệ điều hành Android, Ios | Kiểm tra hiển thị giao diện, và thao tác người dùng (UX/UI) trên các hiệu điều hành khác nhau. |
| 2 | Hệ thống Backend | Môi trường Test Server (Staging): Máy chủ độc lập với dữ liệu giả lập. | Đảm bảo các API hoạt động đúng logic nghiệp vụ |
| 3 | Điều kiện mạng | Mạng Tốt và Mạng Kém/Mất kết nối. Online/Offline | Kiểm tra tính ổn định của ứng dụng khi chuyển đổi giữa các trạng thái mạng, đảm bảo người dùng có thể truy cập nội dung đã load khi Offline và không gặp lỗi gián đoạn |

### 4.1.2 Công cụ kiểm thử

|  |  |  |
| --- | --- | --- |
| **Tên công cụ** | **Vai trò và Ứng dụng** | **Lý do lựa chọn** |
| Postman | Kiểm thử API: Dùng để gửi các request HTTP (GET, POST, PUT, DELETE) tới Server nhằm kiểm tra đầu ra (JSON Response) và mã trạng thái (Status Code) của từng API một cách độc lập trước khi tích hợp vào Frontend. | Đảm bảo tính đúng đắn của dữ liệu đầu vào (input) và đầu ra (output) của các dịch vụ quan trọng. |
| JUnit 5 & Mockito | Kiểm thử đơn vị (Unit Testing): Sử dụng trong mã nguồn Backend (Spring Boot) để kiểm tra logic nghiệp vụ của từng hàm, từng class (đặc biệt là các lớp Service và Controller). | Là framework kiểm thử tiêu chuẩn của hệ sinh thái Java/Spring Boot. Giúp phát hiện lỗi logic ngay trong quá trình viết code (kiểm thử hộ trắng) và đảm bảo độ phủ mã nguồn (Code Coverage) cao. |

## 4.2. Xây dựng các kịch bản kiểm thử

### 4.2.1. Tìm kiếm

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **STT** | **Chức năng** | **Kịch bản kiểm thử** | **Kết quả mong muốn** | **Kết quả thực tế** | **Trạng thái** |
| 1 | Tìm kiếm theo tên món | Người dùng nhập từ khóa tìm kiếm hợp lệ (ví dụ: "phở") vào ô tìm kiếm. | Hệ thống trả về danh sách các công thức có tên chứa từ khóa, đồng thời lưu lại lịch sử tìm kiếm. | Như mong đợi | Đạt |
| 2 | Tìm kiếm theo tên món | Người dùng để trống từ khóa hoặc nhập toàn khoảng trắng. | Hệ thống ném ra ngoại lệ báo lỗi từ khóa rỗng | Như mong đợi | Đạt |
| 3 | Tìm kiếm theo tên món | Người dùng nhập từ khóa quá ngắn (dưới 2 ký tự) hoặc quá dài (> 80 ký tự). | Hệ thống báo lỗi độ dài từ khóa không hợp lệ | Như mong đợi | Đạt |
| 4 | Tìm kiếm theo tên món | Người dùng nhập từ khóa chứa ký tự đặc biệt không hợp lệ | Hệ thống từ chối và báo lỗi ký tự không hợp lệ | Như mong đợi | Đạt |
| 5 | Tìm kiếm theo nguyên liệu | Người dùng tìm kiếm theo tên món kết hợp với danh sách nguyên liệu có sẵn. | Hệ thống trả về danh sách công thức phù hợp với cả tên và nguyên liệu, lưu lịch sử tìm kiếm. | Như mong đợi | Đạt |
| 6 | Tìm kiếm theo nguyên liệu | Người dùng tìm kiếm nhưng không có công thức nào phù hợp. | Hệ thống trả về danh sách rỗng và không lưu lại lịch sử tìm kiếm rác. | Như mong đợi | Đạt |
| 7 | Tìm kiếm tổng hợp (Tên/Tác giả) | Người dùng nhập từ khóa không trùng với tên món ăn nào, nhưng trùng tên người dùng (User). | Hệ thống tự động chuyển sang tìm kiếm và trả về danh sách Người dùng (Tác giả) phù hợp. | Như mong đợi | Đạt |
| 8 | Gợi ý tìm kiếm | Người dùng nhập một vài ký tự đầu (ví dụ: "nguyễn"). | Hệ thống trả về danh sách gợi ý tên món ăn hoặc tên người dùng (tối đa 5 kết quả) khớp với ký tự nhập. | Như mong đợi | Đạt |
| 9 | Lịch sử tìm kiếm | kiếmNgười dùng đã đăng nhập truy cập vào lịch sử tìm kiếm. | Hệ thống hiển thị danh sách 5 từ khóa tìm kiếm gần nhất của người dùng đó. | Như mong đợi | Đạt |

### 4.2.2. Công thức

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **STT** | **Chức năng** | **Kịch bản kiểm thử** | **Kết quả mong muốn** | **Kết quả thực tế** | **Trạng thái** |
| 1 | Tạo công thức mới | Người dùng tạo công thức với thông tin cơ bản hợp lệ. | Công thức được lưu vào CSDL, activity log được ghi lại | Như mong đợi | Đạt |
| 2 | Tạo công thức mới | Tạo công thức kèm theo Danh mục (Category), Thẻ (Tag), Nguyên liệu mới chưa có trong hệ thống. | Hệ thống tự động tạo mới Danh mục/Thẻ/Nguyên liệu đó rồi liên kết vào công thức. | Như mong đợi | Đạt |
| 3 | Tạo công thức mới | Tạo công thức chọn Danh mục/Thẻ/Nguyên liệu đã tồn tại. | Hệ thống tái sử dụng ID cũ để liên kết, không tạo trùng lặp. | Như mong đợi | Đạt |
| 4 | Tạo công thức mới | Người dùng upload ảnh đại diện (Featured Image) và ảnh các bước làm (Step Images). | Ảnh được upload lên Firebase, đường dẫn (URL) được lưu chính xác vào công thức và các bước. | Như mong đợi | Đạt |
| 5 | Tạo công thức mới | Người dùng upload file ảnh bị lỗi hoặc tên file không hợp lệ. | Hệ thống bỏ qua file lỗi, không upload và không lưu URL rác. | Như mong đợi | Đạt |
| 6 | Cập nhật công thức | Người dùng chỉnh sửa thông tin công thức (Tiêu đề, Mô tả). | Thông tin mới được lưu, dữ liệu cũ (Steps, Ingredients cũ) được dọn dẹp và thay thế. | Như mong đợi | Đạt |
| 7 | Cập nhật công thức | Người dùng thay đổi ảnh đại diện của món ăn. | Ảnh cũ trên Firebase bị xóa, ảnh mới được upload và cập nhật URL | Như mong đợi | Đạt |
| 8 | Cập nhật công thức | Người dùng cập nhật công thức nhưng KHÔNG gửi danh sách Nguyên liệu/Thẻ/Danh mục (null). | Hệ thống giữ nguyên dữ liệu cũ của các trường này (không bị xóa mất). | Như mong đợi | Đạt |
| 9 | Xem chi tiết công thức | Người dùng xem chi tiết một công thức theo ID. | Trả về đầy đủ thông tin (Các bước, Nguyên liệu, Tác giả...). Nếu ID sai, báo lỗi | Như mong đợi | Đạt |
| 10 | Xóa công thức | Người dùng xóa một công thức. | Xóa công thức trong CSDL, xóa toàn bộ dữ liệu liên quan (Steps, Ingredients, Tags), xóa ảnh trên Firebase và xóa các thông báo liên quan. | Như mong đợi | Đạt |
| 11 | Lấy danh sách công thức | Lấy danh sách tất cả công thức có phân trang (Pagination). | Trả về danh sách công thức đúng theo trang và số lượng yêu cầu. | Như mong đợi | Như mong đợi |

### 4.2.3. Chức năng theo dõi

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **STT** | **Chức năng** | **Kịch bản kiểm thử** | **Kết quả mong muốn** | **Kết quả thực tế** | **Trạng thái** |
| 1 | Theo dõi người dùng | Người dùng gửi yêu cầu theo dõi (Follow) một tài khoản khác hợp lệ. | Hệ thống xử lý logic follow, thông tin xác nhận (FollowerID, FollowingID). | Như mong đợi | Đạt |
| 2 | Hủy theo dõi | Người dùng gửi yêu cầu hủy theo dõi (Unfollow) một tài khoản đang theo dõi. | Hệ thống xóa liên kết theo dõi, thông báo "Unfollow thành công". | Như mong đợi | Đạt |
| 3 | Danh sách người theo dõi | Người dùng xem danh sách những người đang theo dõi mình. | Danh sách người dùng | Như mong đợi | Đạt |
| 4 | Danh sách đang theo dõi | Người dùng xem danh sách những người mình đang theo dõi | Danh sách người dùng | Như mong đợi | Đạt |
| 5 | Bảng tin | Lấy danh sách công thức mới nhất từ những người dùng mà mình đang theo dõi. | danh sách công thức từ những người được follow. | Như mong đợi | Đạt |

### 4.2.4. Chức năng bình luận

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **STT** | **Chức năng** | **Kịch bản kiểm thử** | **Kết quả mong muốn** | **Kết quả thực tế** | **Trạng thái** |
| 1 | Xem danh sách bình luận | Người dùng xem danh sách các bình luận của một công thức cụ thể | Hệ thống trả về danh sách bình luận theo trang yêu cầu. | Như mong đợi | Đạt |
| 2 | Xem phản hồi | Người dùng xem chi tiết các câu trả lời nằm trong một bình luận gốc. | Hệ thống trả về danh sách các câu trả lời con | Như mong đợi | Đạt |
| 3 | Đăng bình luận mới | Người dùng đã đăng nhập gửi nội dung để tạo một bình luận mới cho công thức. | Hệ thống trả về thông tin chi tiết của bình luận vừa gửi. | Như mong đợi | Đạt |
| 4 | Chỉnh sửa bình luận | Người dùng đã đăng nhập gửi nội dung mới để chỉnh sửa bình luận do mình tạo ra. | Hệ thống cập nhật nội dung, thông tin bình luận sau khi sửa. | Như mong đợi | Đạt |
| 5 | Bảng tin | Lấy danh sách công thức mới nhất từ những người dùng mà mình đang theo dõi. | danh sách công thức từ những người được follow. | Như mong đợi | Đạt |
| 6 | Xóa bình luận | Người dùng đã đăng nhập yêu cầu xóa một bình luận do mình tạo ra. | Hệ thống xóa bình luận thành công | Như mong đợi | Đạt |

### 4.2.5. Chức năng thông báo

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **STT** | **Chức năng** | **Kịch bản kiểm thử** | **Kết quả mong muốn** | **Kết quả thực tế** | **Trạng thái** |
| 1 | Xem danh sách thông báo | Người dùng xem danh sách thông báo cá nhân. | Hệ thống trả về danh sách bình luận theo trang yêu cầu. | Như mong đợi | Đạt |
| 2 | Đánh dấu đã đọc (Một tin) | Người dùng nhấn vào một thông báo cụ thể để xem hoặc đánh dấu là đã đọc. | Hệ thống cập nhật trạng thái cho thông báo đó, | Như mong đợi | Đạt |
| 3 | Đánh dấu đã đọc (Tất cả) | Người dùng chọn chức năng "Đánh dấu tất cả là đã đọc". | Hệ thống cập nhật toàn bộ thông báo của người dùng sang trạng thái đã đọc. | Như mong đợi | Đạt |
| 4 | Xóa thông báo | Người dùng xóa một thông báo cụ thể khỏi danh sách. | Hệ thống xóa thông báo, xác nhận thành công. | Như mong đợi | Đạt |

### 4.2.6. Chức năng hiển thị gợi ý

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **STT** | **Chức năng** | **Kịch bản kiểm thử** | **Kết quả mong muốn** | **Kết quả thực tế** | **Trạng thái** |
| 1 | Tổng hợp gợi ý trang chủ | Người dùng truy cập trang chủ, hệ thống tải đồng thời các mục: Nổi bật, Phổ biến, Mới nhất, Đánh giá cao, Xu hướng và Gợi ý hôm nay. | Trả về đầy đủ 6 danh sách trong một phản hồi duy nhất. Các tác vụ chạy song song để đảm bảo tốc độ phản hồi nhanh. | Như mong đợi | Đạt |
| 2 | Tổng hợp gợi ý trang chủ | Người dùng truy cập trang chủ nhưng Token không hợp lệ hoặc User không tồn tại trong hệ thống | Hệ thống từ chối trả về dữ liệu cá nhân hóa. | Như mong đợi | Đạt |
| 3 | Gợi ý hôm nay (Daily) | Người dùng refresh (tải lại) danh sách "Gợi ý hôm nay" nhiều lần trong cùng một ngày. | Danh sách các món ăn giữ nguyên thứ tự, không bị thay đổi ngẫu nhiên. | Như mong đợi | Đạt |
| 4 | Lấy danh sách | API yêu cầu lấy danh sách (Nổi bật/Phổ biến...) với tham số giới hạn (limit) không hợp lệ (<= 0 hoặc > 50). | Hệ thống báo lỗi dữ liệu đầu vào (yêu cầu limit từ 1 đến 50). | Như mong đợi | Đạt |
| 5 | Hiển thị thông tin | Danh sách trả về chứa các công thức từ nhiều người dùng khác nhau | Thông tin người tạo (Tên, Avatar) từng công thức tương ứng. | Như mong đợi | Đạt |

### 4.2.7. Chức năng thích, đánh giá

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **STT** | **Chức năng** | **Kịch bản kiểm thử** | **Kết quả mong muốn** | **Kết quả thực tế** | **Trạng thái** |
| 1 | Thích công thức | Người dùng nhấn "Thích" một công thức chưa từng thích trước đó. | Lưu lượt thích vào CSDL, tăng số lượng like của món ăn, tạo thông báo cho chủ bài viết. | Như mong đợi | Đạt |
| 2 | Bỏ thích công thức | Người dùng nhấn "Bỏ thích" một công thức mà họ đang thích. | Xóa lượt thích khỏi CSDL, cập nhật lại (giảm) số lượng like của món ăn. | Như mong đợi | Đạt |
| 3 | Xem danh sách đã thích | Người dùng xem danh sách các món ăn mình đã thích | Trả về danh sách các công thức đã thích | Như mong đợi | Đạt |
| 4 | Đánh giá mới | Người dùng thực hiện đánh giá một công thức lần đầu tiên (chưa từng đánh giá trước đó). | Hệ thống lưu điểm đánh giá mới, tính lại điểm trung bình (Average Rating) và tăng tổng số lượt đánh giá của công thức lên 1. | Như mong đợi | Đạt |
| 5 | Cập nhật đánh giá | Người dùng thay đổi điểm đánh giá cho một công thức đã đánh giá trước đó (ví dụ: từ 3 sao lên 4 sao). | Hệ thống cập nhật điểm số mới, tính lại điểm trung bình, nhưng giữ nguyên tổng số lượt đánh giá (không tăng thêm). | Như mong đợi | Đạt |

### 4.2.8. Đăng nhập/Đăng xuất

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **STT** | **Chức năng** | **Kịch bản kiểm thử** | **Kết quả mong muốn** | **Kết quả thực tế** | **Trạng thái** |
| 1 | Đăng nhập | Đăng nhập thành công với thông tin hợp lệ. | Hệ thống cấp phiên, chuyển hướng đến trang chủ. | Như mong đợi | Đạt |
| 2 | Đăng nhập | Đăng nhập Tên Tài khoản, Mật khẩu sai. | Hệ thống hiển thị thông báo lỗi: "Thông tin đăng nhập không hợp lệ". | Như mong đợi | Đạt |
| 3 | Đăng nhập | Đăng nhập khi để trống cả Tài Khoản và Mật khẩu. | Hệ thống hiển thị thông báo lỗi yêu cầu nhập đủ thông tin. | Như mong đợi | Đạt |
| 4 | Đăng nhập | Đăng nhập với tài khoản đang bị khóa/chưa kích hoạt (nếu có). | Hệ thống hiển thị thông báo lỗi tương ứng. | Như mong đợi | Đạt |
| 5 | Đăng xuất | Click vào nút "Đăng xuất". | Phiên làm việc kết thúc, chuyển hướng về trang Đăng nhập. | Như mong đợi | Đạt |

### 4.2.9. Quản lý tài khoản người dùng

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **STT** | **Chức năng** | **Kịch bản kiểm thử** | **Kết quả mong muốn** | **Kết quả thực tế** | **Trạng thái** |
| 1 | Tạo tài khoản | Đăng ký thành công với tất cả thông tin hợp lệ (họ tên, email, mật khẩu). | Tài khoản được tạo, thông báo thành công, tài khoản mặc định có vai trò "Người học". | Như mong đợi | Đạt |
| 2 | Tạo tài khoản | Đăng ký thất bại khi sử dụng email đã tồn tại. | Hệ thống từ chối đăng ký và hiển thị thông báo lỗi rõ ràng. | Như mong đợi | Đạt |
| 3 | Tạo tài khoản | Đăng ký thất bại khi mật khẩu không đáp ứng quy tắc (ví dụ: quá ngắn, thiếu ký tự đặc biệt). | Hệ thống từ chối và hiển thị thông báo chỉ rõ yêu cầu của mật khẩu. | Như mong đợi | Đạt |
| 4 | Truy xuất thông tin | Quản trị viên (Admin) tìm kiếm người dùng theo Email chính xác. | Hiển thị chính xác thông tin chi tiết của người dùng được tìm kiếm. | Như mong đợi | Đạt |
| 5 | Chỉnh sửa | Admin chỉnh sửa Họ tên, Mật khẩu của một tài khoản "Người dùng". | Thông tin được cập nhật thành công, người dùng có thể đăng nhập bằng mật khẩu mới. | Như mong đợi | Đạt |
| 6 | Xóa tài khoản | Admin xóa một tài khoản "Người dùng" hiện có. | Tài khoản bị xóa khỏi hệ thống và không thể đăng nhập lại | Như mong đợi | Đạt |

## 4.3 Đánh giá kết quả kiểm thử

### 4.3.1. Tổng quan

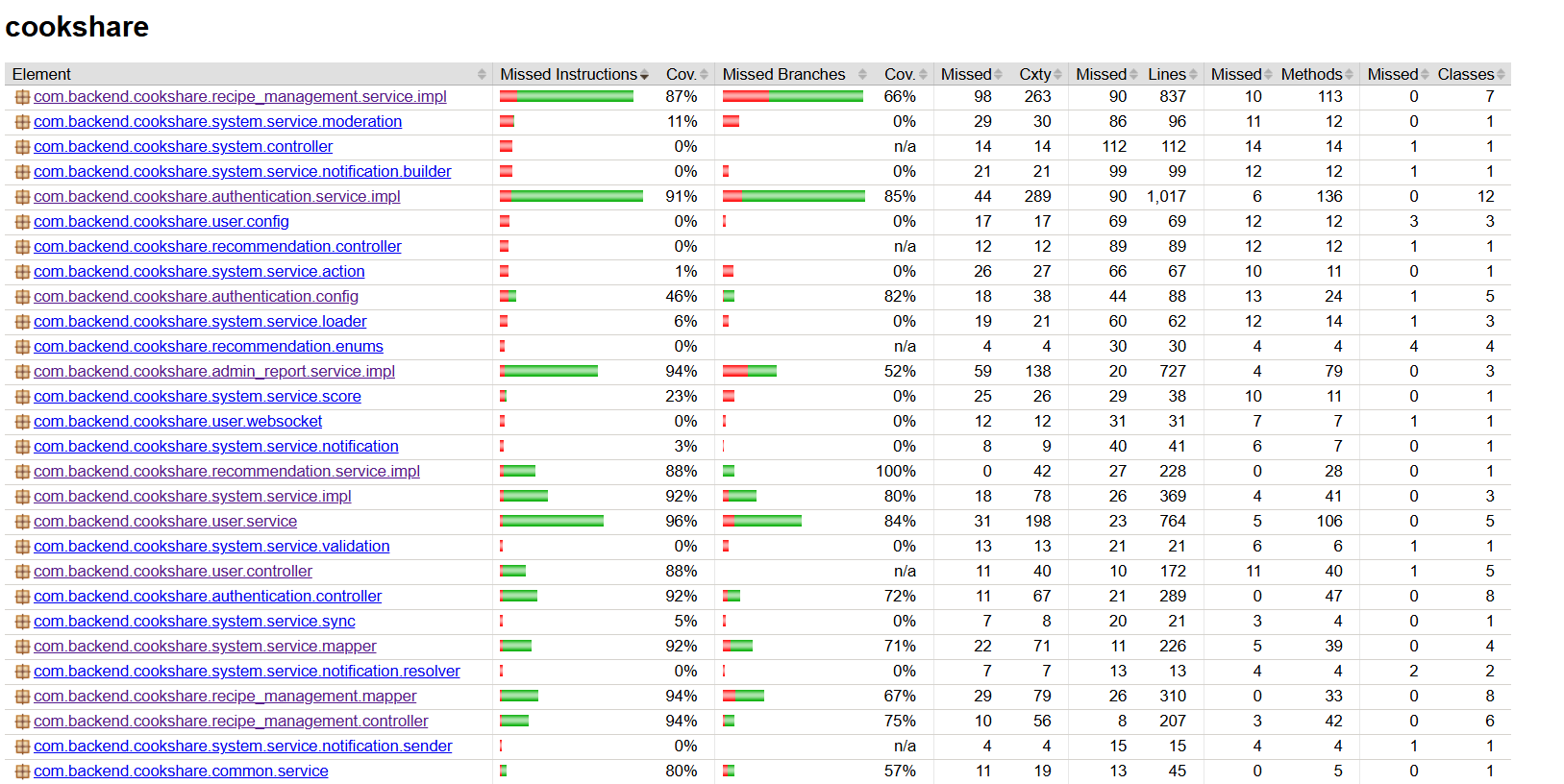
Tổng số Kịch bản Kiểm thử: 46 được thực hiện, bao trọn các chức năng nghiệp vụ cốt lõi của hệ thống.

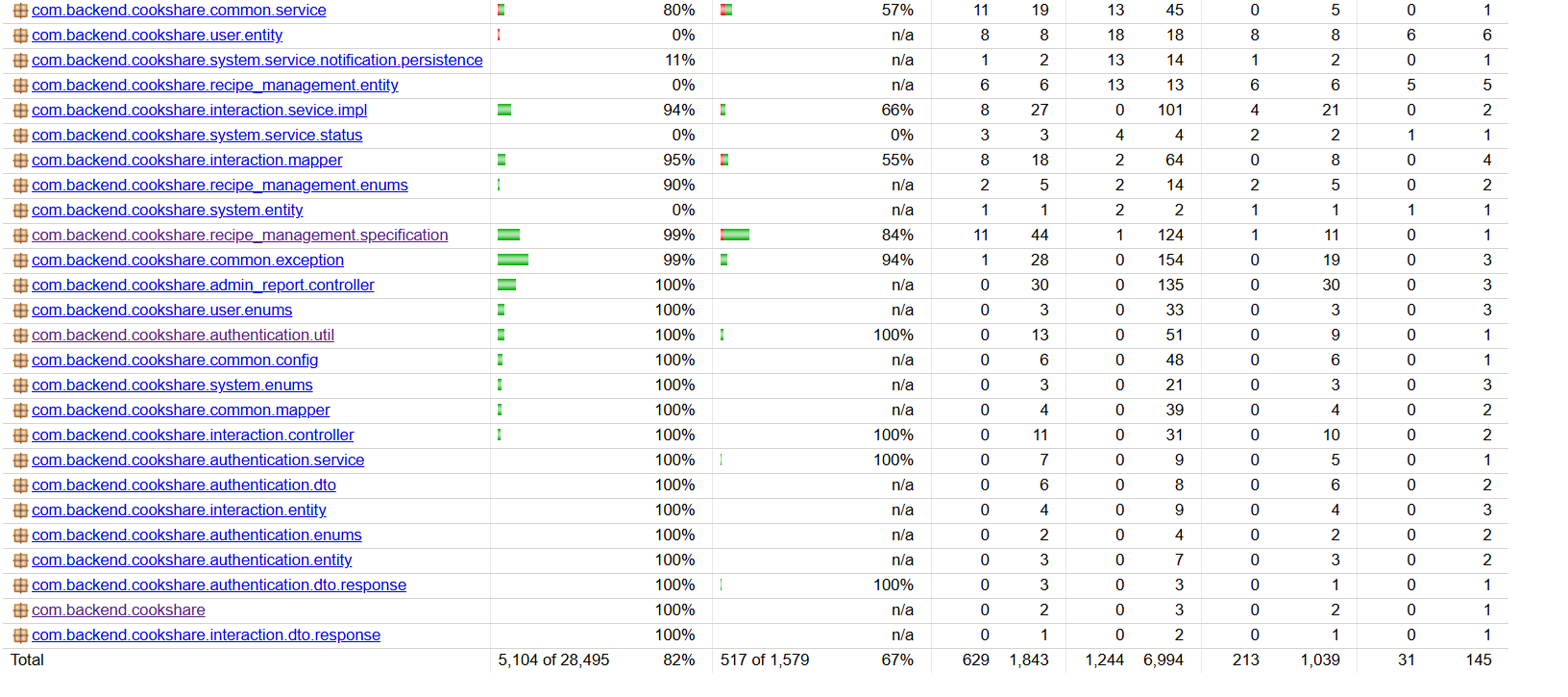
Tỷ lệ Đạt: Đạt 100% (với 46 kịch bản đạt yêu cầu).

Lỗi Phát hiện: 8 được phát hiện, chủ yếu là lỗi cấp độ nhẹ (Minor) liên quan đến giao diện người dùng và xử lý đầu vào.

Tình trạng Khắc phục: 100% số lỗi đã được khắc phục và kiểm thử lại thành công.

### 4.3.2. Phân tích kết quả kiểm thử JUNIT





**Hình 4. 1.** Kết quả thực hiển unit test

#### 4.3.2.1. Tổng quan

Dự án đã thực hiện kiểm thử đơn vị (Unit Test) sử dụng JUnit với kết quả tổng thể như sau:

* **Độ bao phủ mã nguồn (Code Coverage):** 82% (5,104/28,495 instructions)
* **Độ bao phủ nhánh (Branch Coverage):** 67% (517/1,579 branches)
* **Tổng số phương thức:** 2,472 methods (1,843 covered, 629 missed)
* **Tổng số lớp:** 176 classes (31 missed)

Kết quả này cho thấy dự án đã có một nền tảng kiểm thử khá tốt với hơn 4/5 mã nguồn được kiểm tra.

#### 4.3.2.2. Phân tích chi tiết theo module

1. **Module Authentication (Xác thực)**

Module authentication có độ bao phủ rất tốt với hầu hết các components đạt **90-100% coverage**. Các chức năng xác thực, phân quyền, và quản lý phiên đăng nhập đều đã được kiểm thử đầy đủ. Đây là điểm mạnh của dự án vì authentication là một module quan trọng liên quan đến bảo mật hệ thống.

1. **Module System (Hệ thống)**

Module system có độ bao phủ khá tốt với **coverage trung bình 85-95%**. Tuy nhiên, một số components cần được chú ý:

* **System services:** Đạt coverage cao (88-96%)
* **System validation:** Chỉ đạt 0% - chưa có test cases
* **System notification:** Chỉ đạt 3% - hầu như chưa kiểm thử
* **System loader:** Chỉ đạt 6% - cần bổ sung tests

Mặc dù các service chính hoạt động tốt, nhưng các utility components như validation và notification cần được ưu tiên kiểm thử.

1. **Module User (Người dùng)**

Module user có **coverage 88%** cho controller, tuy nhiên **branch coverage chỉ 0%**. Điều này cho thấy:

* Các luồng xử lý cơ bản đã được test
* Các trường hợp ngoại lệ và điều kiện rẽ nhánh chưa được kiểm tra
* Cần bổ sung test cases cho các edge cases

1. **Module Recipe Management (Quản lý công thức)**

Module này đạt **coverage 87-94%** cho các components chính. Độ bao phủ khá tốt, tuy nhiên branch coverage dao động từ 52-67%, cho thấy vẫn còn nhiều điều kiện logic chưa được kiểm tra đầy đủ.

1. **Module Admin & Report (Quản trị & Báo cáo)**

Module admin đạt **coverage 94%** với branch coverage 52%. Các chức năng quản trị và báo cáo đã được kiểm thử tương đối đầy đủ, tuy nhiên vẫn cần bổ sung tests cho các nhánh điều kiện.

#### 4.3.2.3. Kết luận

Kết quả kiểm thử cho thấy dự án đã có nền tảng test khá tốt với 82% code coverage. Các module core như Authentication, System, và Recipe Management đều đạt độ bao phủ tốt.

* Tuy nhiên, vẫn tồn tại một số điểm yếu cần khắc phục khẩn cấp:
* Branch coverage còn thấp (67%)
* Một số utility services chưa được kiểm thử đầy đủ

Nhóm phát triển cần tập trung bổ sung tests trong các sprint tiếp theo, đặc biệt ưu tiên module Recommendation, để nâng cao chất lượng sản phẩm và giảm thiểu rủi ro phát sinh lỗi trong môi trường production.

# KẾT LUẬN VÀ KIẾN NGHỊ

Qua quá trình nghiên cứu và thực hiện đề tài “Ứng dụng chia sẻ các công thức nấu ăn”, nhóm phát triển đã hoàn thành xuất sắc mục tiêu đề ra là xây dựng một hệ sinh thái phần mềm hoàn chỉnh, đáp ứng đầy đủ các yêu cầu chức năng nghiệp vụ cũng như các tiêu chuẩn phi chức năng về hiệu năng và trải nghiệm người dùng. Hệ thống được thiết kế và phát triển thành công dựa trên kiến trúc mô hình client-server hiện đại, đảm bảo sự tách biệt rõ ràng giữa các tầng. Trong đó, phía Frontend sử dụng framework React-Native mang lại trải nghiệm mượt mà đa nền tảng, còn phía Backend được xây dựng vững chắc trên nền tảng Spring Boot, kết hợp với hệ quản trị cơ sở dữ liệu PostgreSQL để đảm bảo tính ổn định và khả năng mở rộng dữ liệu.

Về mặt chức năng, ứng dụng đã hoàn thiện và vận hành trơn tru các nhóm tính năng cốt lõi:

* Quản lý người dùng: Quy trình xác thực bảo mật (đăng ký, đăng nhập) và quản lý hồ sơ cá nhân được tối ưu hóa.
* Quản lý công thức: Người dùng có thể dễ dàng đóng góp nội dung thông qua các thao tác tạo mới, chỉnh sửa, xóa và chia sẻ công thức nấu ăn với cộng đồng.
* Hệ thống tìm kiếm nâng cao: Cung cấp bộ lọc đa chiều, cho phép người dùng tra cứu linh hoạt theo danh sách nguyên liệu, thể loại món ăn, tên món hoặc tìm kiếm theo tác giả.
* Công cụ quản trị (Admin): Cung cấp cái nhìn tổng quan qua các dashboard thống kê, đồng thời hỗ trợ kiểm duyệt nội dung và quản lý danh sách người dùng hiệu quả.
* Tương tác xã hội & Thời gian thực: Hệ thống thúc đẩy tính cộng đồng thông qua các hành động thích, bình luận, theo dõi người dùng và lưu trữ công thức yêu thích. Điểm nhấn công nghệ là việc tích hợp thành công WebSocket, cho phép thông báo và phản hồi diễn ra trong thời gian thực (real-time), giúp tăng cường sự tương tác sống động giữa các người dùng.

Trong suốt quá trình triển khai đề tài, nhóm đã không chỉ áp dụng mà còn củng cố sâu sắc nhiều khối kiến thức và kỹ năng quan trọng:

* Làm chủ quy trình phát triển ứng dụng Full-stack, kết hợp nhuần nhuyễn giữa các framework mã nguồn mở hàng đầu hiện nay là Spring Boot và React-Native.
* Nâng cao tư duy thiết kế hệ thống thông qua việc xây dựng và tối ưu hóa cơ sở dữ liệu quan hệ (RDBMS) với PostgreSQL, đảm bảo tuân thủ các dạng chuẩn hóa dữ liệu, tính toàn vẹn tham chiếu và tối ưu hiệu suất truy vấn.
* Thành thạo quy trình xây dựng RESTful API chuẩn mực, kỹ thuật xử lý dữ liệu JSON và áp dụng triệt để mô hình kiến trúc MVC (Model-View-Controller) trong phát triển Backend, giúp mã nguồn dễ bảo trì và mở rộng.
* Tích lũy kinh nghiệm thực tế quý báu về **DevOps** thông qua việc triển khai thử nghiệm ứng dụng lên nền tảng điện toán đám mây **Google Cloud Platform**, qua đó hiểu rõ hơn về quy trình tích hợp và triển khai liên tục (CI/CD) cũng như quản lý môi trường **production**.
* Sử dụng thành thạo và chuyên nghiệp bộ công cụ hỗ trợ phát triển phần mềm hiện đại như Postman, Git, Visual Studio Code và IntelliJ IDEA.

Nhìn chung, hệ thống sau khi hoàn thiện đã hoạt động ổn định trên môi trường thử nghiệm, giao diện được thiết kế trực quan, thân thiện, giải quyết tốt bài toán hỗ trợ người dùng chia sẻ và khám phá ẩm thực. Mặc dù vẫn còn tồn tại một số hạn chế nhỏ về mặt tính năng nâng cao, nhưng ứng dụng hiện tại đã là một nền tảng kỹ thuật vững chắc, sẵn sàng cho các bước nâng cấp tiếp theo.

Để tiếp tục hoàn thiện sản phẩm và nâng cao giá trị trải nghiệm cho người dùng, nhóm xin đề xuất lộ trình phát triển trong tương lai với các hạng mục cụ thể:

* Nghiên cứu và tích hợp các thuật toán gợi ý hoặc AI để đề xuất công thức nấu ăn thông minh dựa trên danh sách nguyên liệu "có gì nấu nấy" của người dùng. Đồng thời, mở rộng tính năng mạng xã hội như tạo nhóm nấu ăn, nhắn tin trực tiếp và kết bạn để tăng tính gắn kết cộng đồng.
* Cần lựa chọn nâng cấp các phần cứng để có hiệu năng tốt hơn hiện tại khi chỉ triển khai trên môi trường server
* Tiếp tục tinh chỉnh mã nguồn để giảm độ trễ API, áp dụng các kỹ thuật Caching để tăng tốc độ tải dữ liệu trên thiết bị di động. Song song đó, tiến hành thiết kế lại giao diện (UI Redesign) theo các xu hướng UX hiện đại nhất, tập trung vào sự tối giản và tiện dụng.
* Triển khai ứng dụng đến một nhóm người dùng thử nghiệm (Beta testers) rộng lớn hơn nhằm thu thập phản hồi thực tế. Các dữ liệu này sẽ là cơ sở quan trọng để nhóm phân tích, xác định các lỗi tiềm ẩn và định hướng chính xác cho việc cập nhật tính năng trong các phiên bản phát hành chính thức.

# DANH MỤC TÀI LIỆU THAM KHẢO

1. **Nguyễn Quốc Tuấn**. 2025 .Slide bài giảng học phần Project 1, Trường Đại học Giao thông Vận tải.

2. **VMware (2024).** Spring Boot Documentation, <https://spring.io/projects/spring-boot>.

3. **Meta Platforms (2024).** React Documentation, <https://react.dev/>.

4. **The PostgreSQL Global Development Group (2024).** PostgreSQL Documentation, <https://www.postgresql.org/docs/>.