

TRƯỜNG KỸ THUẬT VÀ CÔNG NGHỆ
KHOA CÔNG NGHỆ THÔNG TIN



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

HỆ THỐNG TƯ VẤN VÀ BÁN THUỐC BẢO VỆ THỰC VẬT THÔNG MINH

Giảng viên hướng dẫn:

TS. Nguyễn Bảo Ân

Sinh viên thực hiện:

Họ tên: **Huỳnh Hữu Lộc**

MSSV: **110122104**

Lớp: **DA22TTA**

Vĩnh Long, tháng 12 năm 2025

NHẬN XÉT CỦA GIẢNG VIÊN HƯỚNG DẪN

Vĩnh Long, ngày tháng năm

Giảng viên hướng dẫn

(Ký tên và ghi rõ họ tên)

NHẬN XÉT CỦA THÀNH VIÊN HỘI ĐỒNG

Vĩnh Long, ngày tháng năm

Thành viên hội đồng

(Ký tên và ghi rõ họ tên)

LỜI CẢM ƠN

Em xin bày tỏ lòng biết ơn sâu sắc đến giảng viên hướng dẫn thầy Nguyễn Bảo Ân, người đã tận tình hướng dẫn, góp ý và hỗ trợ em trong suốt quá trình thực hiện đồ án chuyên ngành. Với sự tận tâm và kiến thức chuyên môn sâu rộng, thầy đã giúp em định hướng đúng đắn, tháo gỡ nhiều khó khăn trong quá trình nghiên cứu và xây dựng hệ thống.

Em xin chân thành cảm ơn quý thầy cô Khoa Công nghệ Thông tin, Trường Kỹ thuật và Công nghệ đã truyền đạt những kiến thức nền tảng quý báu trong suốt thời gian học tập, tạo điều kiện để em có thể vận dụng vào thực tế trong quá trình thực hiện đồ án.

Em xin chân thành cảm ơn !

MỤC LỤC

CHƯƠNG 1: TỔNG QUAN	12
1.1 Thực trạng sử dụng thuốc bảo vệ thực vật tại Việt Nam.....	12
1.2 Tổng quan các hệ thống và nền tảng liên quan hiện nay.....	12
1.3 Khoảng trống và hướng giải quyết	13
1.4 Mục tiêu cụ thể của đề án.....	13
CHƯƠNG 2: NGHIÊN CỨU LÝ THUYẾT	14
2.1 Tổng quan về hệ thống Website tư vấn và bán thuốc bảo vệ thực vật.....	14
2.1.1 Giới thiệu hệ thống.....	14
2.1.2 Các phân hệ chức năng.....	14
2.2 Kiến trúc MEAN Stack.....	15
2.2.1 Tổng quan kiến trúc MERN Stack	15
2.2.2 Mô hình MVC trong kiến trúc MERN Stack	15
2.2.3 Luồng xử lý dữ liệu trong hệ thống MERN	15
2.2.4 Đánh giá kiến trúc MERN Stack.....	16
2.3 Công nghệ và framework front-end	16
2.3.1 React.js	16
2.3.2 Công nghệ TailwindCSS.....	17
2.4 Công nghệ và framework back-end.....	17
2.4.1 Tổng quan Back-end của hệ thống.....	17
2.4.2 Công nghệ Node.js	17
2.4.3 Công nghệ Express.js.....	18
2.5 Chatbot tư vấn thông minh RAG tích hợp Google Gemini.....	18
2.5.1 Tổng quan.....	18
2.5.2 Kho tri thức và vai trò của Google Gemini	18
2.5.3 Kho tri thức và vai trò của Google Gemini.....	19
2.6 Hệ quản trị cơ sở dữ liệu	19
2.6.1 Giới thiệu hệ quản trị cơ sở dữ liệu MongoDB.....	19
2.6.2 Giới thiệu thư viện Mongoose	19
2.6.3 Ứng dụng trong đề án	19
2.7 Công cụ hỗ trợ phát triển và triển khai	20
2.7.1 Công cụ phát triển mã nguồn Visual Studio Code (VS Code).....	20

2.7.2 Công cụ quản lý mã nguồn GitHub.....	20
CHƯƠNG 3: HIỆN THỰC HÓA NGHIÊN CỨU	21
3.1 Mô tả bài toán và mục tiêu hệ thống	21
3.2 Quy trình nghiệp vụ tổng quát.....	21
3.3 Đối tượng sử dụng	22
3.3.1 Người dùng thông thường (Khách hàng)	23
3.3.2 Quản trị viên hệ thống (Administrator).....	23
3.3.3 Khách truy cập (Visitor).....	23
3.4 Phân tích và thiết kế hệ thống.....	23
3.4.1 Yêu cầu chức năng	23
3.4.2 Yêu cầu phi chức năng	25
3.5 Thiết kế cơ sở dữ liệu	25
3.5.1 Bảng người dùng (User).....	27
3.5.2 Bảng sản phẩm (Product)	28
3.5.3 Bảng đơn hàng (Order)	29
3.5.4 Bảng phiên tư vấn (ChatSession)	30
3.5.5 Bảng kho tri thức (KnowledgeBase).....	30
3.6 Phân tích và thiết kế hệ thống.....	31
3.6.1 Phân tích và thiết kế Backend(Giao diện người dùng)	31
3.6.2 Phân tích và thiết kế Frontend (Giao diện người dùng).....	34
3.6.3 Phân tích và thiết kế Frontend (Giao diện người dùng).....	36
CHƯƠNG 4: KẾT QUẢ NGHIÊN CỨU	38
4.1 Tổng quan kết quả đạt được	38
4.2 Kết quả triển khai các chức năng.....	38
4.2.1 Kết quả triển khai chức năng bán hàng	38
4.2.2 Kết quả triển khai chức năng quản trị	43
4.2.3 Chức năng tư vấn bệnh cây trồng bằng Chatbot thông minh.....	48
4.2.4 Kho tri thức và khả năng mở rộng	52
CHƯƠNG 5: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN.....	53
5.1 Kết luận.....	53
5.1.1 Kết quả đạt được	53
5.1.2 Đóng góp mới.....	53

5.1.3 Hạn chế.....	54
5.2 Hướng phát triển.....	54
DANH MỤC TÀI LIỆU THAM KHẢO	56

DANH MỤC HÌNH ẢNH

Hình 2.1 Sơ đồ hoạt động của hệ thống	22
Hình 2.2 Mô hình ERD của cơ sở dữ liệu	26
Hình 2.3 Sơ đồ truy vấn API của Backend	32
Hình 3.1 Trang chủ hệ thống	39
Hình 3.2 Trang danh sách sản phẩm	40
Hình 3.3 Trang chi tiết sản	40
Hình 3.4 Trang giỏ hàng	
Hình 3.5 Xác nhận đặt hàng	42
Hình 3.6 Trang lịch sử mua hàng	42
Hình 3.7 Trang hồ sơ cá nhân	43
Hình 3.8 Thống kê Tổng quan về cửa hàng và doanh thu	44
Hình 3.9 Thống kê Sản phẩm bán chạy	44
Hình 3.10 Thống kê khách hàng thân thiết	44
Hình 3.11 Form thêm sản phẩm mới	45
Hình 3.12 Sửa thông tin sản phẩm	46
Hình 3.13 Quản lý danh sách sản phẩm	46
Hình 3.14 Quản lý đơn hàng	47
Hình 3.15 Quản lý người dùng	48
Hình 3.16 Khung chat tư vấn	48
Hình 3.17 Nhập triệu chứng và chờ kết quả tư vấn	49
Hình 3.18 Kết quả tư vấn từ hệ thống	49
Hình 3.19 Hệ thống gợi ý thuốc điều trị có trong cửa hàng	50
Hình 3.20 Lịch sử tư vấn	50
Hình 3.21 Giao diện Chat Component	51
Hình 3.22 Dữ liệu kho tri thức	52

DANH MỤC BẢNG BIỂU

Bảng 3.1 Các trường trong bảng user	27
Bảng 3.2 Các trường trong bảng Product	28
Bảng 3.3 Các trường trong bảng Order	29
Bảng 3.4 Các trường trong bảng ChatSession	30
Bảng 3.5 Các trường trong bảng ChatSession	31

TÓM TẮT ĐỒ ÁN CHUYÊN NGÀNH

Đồ án tập trung nghiên cứu và xây dựng Hệ thống tư vấn và bán thuốc bảo vệ thực vật thông minh nhằm hỗ trợ người dân và hộ kinh doanh trong việc tra cứu, mua bán và tư vấn các sản phẩm nông nghiệp một cách thuận tiện và hiệu quả. Trong bối cảnh chuyển đổi số đang diễn ra mạnh mẽ, việc ứng dụng công nghệ thông tin vào lĩnh vực nông nghiệp đóng vai trò quan trọng trong việc nâng cao năng suất, giảm chi phí và tăng khả năng tiếp cận thông tin chính xác cho người nông dân.

Hệ thống được phát triển theo mô hình **MERN Stack (MongoDB, Express.js, React.js, Node.js)**, đáp ứng đầy đủ các chức năng như: quản lý sản phẩm thuốc bảo vệ thực vật, phân bón, thức ăn chăn nuôi; tìm kiếm và lọc sản phẩm theo hoạt chất, công dụng; quản lý giỏ hàng, đơn hàng; và tích hợp chatbot tư vấn hỗ trợ người dùng. Giao diện được thiết kế thân thiện, trực quan, phù hợp với người dùng phổ thông trong lĩnh vực nông nghiệp.

Kết quả đạt được là một website hoạt động ổn định, có khả năng triển khai thực tế cho các cửa hàng vật tư nông nghiệp, góp phần hỗ trợ công tác kinh doanh và nâng cao hiệu quả tư vấn kỹ thuật cho bà con nông dân.

MỞ ĐẦU

1. Lý do chọn đề tài

Nông nghiệp là một trong những ngành kinh tế trọng điểm của Việt Nam, đóng vai trò quan trọng trong đảm bảo lương thực quốc gia.

Tuy nhiên, thực tế cho thấy việc tiếp cận thông tin về thuốc bảo vệ thực vật, phân bón và thức ăn chăn nuôi của người dân vẫn còn nhiều hạn chế. Phần lớn các cửa hàng vật tư nông nghiệp hiện nay vẫn hoạt động theo mô hình truyền thống, thiếu hệ thống quản lý tập trung và chưa khai thác hiệu quả các nền tảng công nghệ số.

Bên cạnh đó, tình trạng sử dụng thuốc và phân bón không đúng kỹ thuật do thiếu tư vấn chính xác vẫn còn phổ biến, gây ảnh hưởng đến năng suất cây trồng, vật nuôi và môi trường. Vì vậy, việc xây dựng một website vừa phục vụ kinh doanh, vừa hỗ trợ tư vấn và cung cấp thông tin kỹ thuật là hết sức cần thiết.

Xuất phát từ những vấn đề trên, em lựa chọn đề tài “Hệ thống tư vấn và bán thuốc bảo vệ thực vật thông minh” nhằm ứng dụng kiến thức chuyên ngành Công nghệ thông tin vào giải quyết một bài toán thực tiễn trong lĩnh vực nông nghiệp.

2. Mục đích nghiên cứu

Mục đích của đề tài là nghiên cứu và xây dựng một hệ thống website hoàn chỉnh phục vụ cho việc kinh doanh thuốc bảo vệ thực vật, đồng thời tích hợp chức năng tư vấn bệnh hại trên cây trồng. Hệ thống hướng đến việc giúp người dùng dễ dàng tra cứu, lựa chọn và mua các sản phẩm phù hợp, cũng như nhận được tư vấn kỹ thuật chính xác trong quá trình canh tác.

Về mặt xã hội, đề tài góp phần nâng cao nhận thức và kiến thức của người nông dân trong việc sử dụng thuốc bảo vệ thực vật đúng cách, đúng bệnh và đúng liều lượng. Thông qua hệ thống tư vấn thông minh, người dùng có thể hạn chế việc lạm dụng thuốc, từ đó giảm thiểu rủi ro cho sức khỏe con người và môi trường, đồng thời nâng cao hiệu quả sản xuất nông nghiệp.

Về mặt kỹ thuật, đề tài nhằm ứng dụng các công nghệ phát triển web hiện đại theo mô hình MERN Stack (MongoDB, Express.js, React.js, Node.js) để xây dựng một ứng dụng web có hiệu năng cao, giao diện thân thiện và hoạt động ổn định. Đặc biệt, hệ thống tích hợp Chatbot RAG sử dụng mô hình ngôn ngữ lớn Google Gemini, cho phép chẩn đoán bệnh cây trồng dựa trên triệu chứng văn bản hoặc hình ảnh và truy xuất thông tin điều trị từ cơ sở tri thức, mang lại khả năng tư vấn nhanh chóng và chính xác.

3. Đối tượng và phạm vi nghiên cứu

- Đối tượng nghiên cứu
 - + Các công nghệ lập trình web Full-stack, bao gồm:
 - React.js dùng để xây dựng giao diện người dùng (Frontend).
 - Node.js và Express.js dùng để xây dựng hệ thống máy chủ và API (Backend).
 - MongoDB dùng để lưu trữ dữ liệu sản phẩm, người dùng, đơn hàng và kho tri thức.
 - + Các mô hình và kỹ thuật tích hợp trí tuệ nhân tạo, cụ thể là:
 - Mô hình ngôn ngữ lớn (LLM) Google Gemini.
 - Kỹ thuật Retrieval-Augmented Generation (RAG) trong tư vấn bệnh cây trồng.
 - + Nhu cầu tra cứu thông tin, mua sắm và tư vấn kỹ thuật của người dùng trong lĩnh vực nông nghiệp, đặc biệt là nông dân và các hộ kinh doanh vật tư nông nghiệp.
 - + Dữ liệu bệnh hại cây trồng, triệu chứng và biện pháp xử lý được xây dựng trong cơ sở tri thức (Knowledge Base).
- Phạm vi nghiên cứu
 - + Về nội dung:
 - Tập trung vào lĩnh vực thuốc bảo vệ thực vật và tư vấn bệnh trên cây trồng.
 - Cung cấp thông tin về triệu chứng bệnh và hướng xử lý phù hợp.
 - + Về chức năng hệ thống:
 - Xem, tìm kiếm, lọc và sắp xếp sản phẩm thuốc bảo vệ thực vật.
 - Quản lý giỏ hàng, đặt hàng và theo dõi lịch sử đơn hàng.
 - Tư vấn bệnh cây trồng thông qua chatbot sử dụng văn bản và hình ảnh.
 - + Về công nghệ:
 - Giới hạn trong phạm vi phát triển ứng dụng web (Web Application).
 - Sử dụng JavaScript làm ngôn ngữ lập trình chính.
 - Áp dụng MERN Stack và Tailwind CSS cho giao diện.

+ Về đối tượng người dùng:

- Hướng đến nông dân, người kinh doanh vật tư nông nghiệp và người quan tâm đến kỹ thuật canh tác.

+ Về không gian và thời gian:

- Đề tài được thực hiện trong khuôn khổ đề án chuyên ngành.
- Thời gian triển khai từ ngày 03/11/2025 đến ngày 28/12/2025.

4. Ý nghĩa khoa học và thực tiễn

Về ý nghĩa khoa học, đề tài là cơ hội để sinh viên hệ thống hóa và vận dụng tổng hợp các kiến thức đã được học trong chương trình đào tạo như lập trình web, cơ sở dữ liệu, kiến trúc phần mềm và bảo mật hệ thống. Việc xây dựng hệ thống theo mô hình Client–Server và kiến trúc MVC giúp sinh viên nâng cao khả năng tổ chức mã nguồn, thiết kế API chuẩn và quản lý dữ liệu một cách khoa học.

Về ý nghĩa thực tiễn, sản phẩm của đề tài có khả năng triển khai và sử dụng trong thực tế tại các cửa hàng vật tư nông nghiệp. Website không chỉ hỗ trợ hoạt động kinh doanh mà còn đóng vai trò như một công cụ tư vấn kỹ thuật hữu ích cho nông dân. Trong tương lai, hệ thống có thể tiếp tục được mở rộng, bổ sung thêm nhiều dữ liệu bệnh hại và sản phẩm mới, góp phần nâng cao hiệu quả sản xuất nông nghiệp và thúc đẩy quá trình chuyển đổi số trong lĩnh vực nông nghiệp tại Việt Nam.

CHƯƠNG 1: TỔNG QUAN

1.1 Thực trạng sử dụng thuốc bảo vệ thực vật tại Việt Nam

Nông nghiệp là một trong những ngành kinh tế trọng điểm của Việt Nam, đóng vai trò quan trọng trong đảm bảo an ninh lương thực và sinh kế của hàng triệu hộ dân. Tuy nhiên, song song với sự phát triển của sản xuất nông nghiệp là những vấn đề tồn tại trong việc sử dụng thuốc bảo vệ thực vật. Trên thực tế, nhiều nông dân vẫn còn gặp khó khăn trong việc tiếp cận thông tin chính xác về bệnh hại cây trồng, cách nhận biết triệu chứng cũng như lựa chọn loại thuốc phù hợp để xử lý.

Việc sử dụng thuốc bảo vệ thực vật hiện nay chủ yếu dựa trên kinh nghiệm truyền miệng hoặc tư vấn trực tiếp tại các cửa hàng vật tư nông nghiệp truyền thống. Điều này dẫn đến tình trạng dùng sai thuốc hoặc phun không đúng thời điểm, gây ảnh hưởng tiêu cực đến năng suất cây trồng và tiềm ẩn nguy cơ ô nhiễm môi trường. Bên cạnh đó, việc thiếu một hệ thống quản lý tập trung cũng khiến người kinh doanh khó kiểm soát tồn kho, đơn hàng và thông tin khách hàng một cách hiệu quả.

Trong bối cảnh chuyển đổi số đang diễn ra mạnh mẽ, việc ứng dụng công nghệ thông tin vào lĩnh vực tư vấn và kinh doanh thuốc bảo vệ thực vật trở nên cần thiết nhằm nâng cao hiệu quả sản xuất nông nghiệp và hỗ trợ người nông dân tiếp cận tri thức một cách nhanh chóng, chính xác hơn.

1.2 Tổng quan các hệ thống và nền tảng liên quan hiện nay

Hiện nay, trên thị trường đã tồn tại một số website và nền tảng cung cấp thông tin về nông nghiệp, thuốc bảo vệ thực vật và kỹ thuật canh tác. Một số trang tập trung vào việc giới thiệu sản phẩm, trong khi các trang khác chủ yếu cung cấp kiến thức kỹ thuật dưới dạng bài viết hoặc tài liệu hướng dẫn. Tuy nhiên, phần lớn các hệ thống này hoạt động rời rạc, chưa có sự kết hợp giữa chức năng tư vấn kỹ thuật và thương mại điện tử.

Ngoài ra, các kênh mạng xã hội và diễn đàn trực tuyến cũng được người nông dân sử dụng để hỏi đáp về bệnh hại cây trồng. Tuy nhiên, thông tin trên các nền tảng này thường thiếu kiểm chứng, không có cơ chế phân loại rõ ràng và phụ thuộc nhiều vào kinh nghiệm cá nhân. Điều này gây khó khăn cho người dùng trong việc xác định đâu là giải pháp phù hợp và an toàn.

Một số ứng dụng và hệ thống tư vấn tự động đã xuất hiện, nhưng đa phần chỉ dừng lại ở mức tra cứu thông tin đơn giản, chưa khai thác hiệu quả các công nghệ trí tuệ

nhân tạo hiện đại để hỗ trợ chẩn đoán bệnh cây trồng dựa trên triệu chứng cụ thể hoặc hình ảnh thực tế.

1.3 Khoảng trống và hướng giải quyết

Từ thực trạng trên có thể nhận thấy rằng vẫn còn tồn tại một khoảng trống lớn trong việc xây dựng một hệ thống tích hợp đầy đủ giữa tư vấn kỹ thuật nông nghiệp và hoạt động kinh doanh thuốc bảo vệ thực vật. Cụ thể, thiếu một nền tảng cho phép người dùng vừa có thể tra cứu, chẩn đoán bệnh cây trồng một cách khoa học, vừa có thể lựa chọn và mua sản phẩm phù hợp ngay trên cùng một hệ thống.

Bên cạnh đó, việc ứng dụng trí tuệ nhân tạo trong lĩnh vực này tại Việt Nam vẫn còn hạn chế, đặc biệt là các hệ thống có khả năng kết hợp giữa mô hình ngôn ngữ lớn và cơ sở tri thức chuyên ngành để đưa ra tư vấn chính xác, dễ hiểu cho người nông dân.

Xuất phát từ khoảng trống đó, đề án đề xuất xây dựng một hệ thống website tư vấn và bán thuốc bảo vệ thực vật thông minh, ứng dụng công nghệ MERN Stack kết hợp Chatbot RAG sử dụng mô hình Google Gemini. Hệ thống hướng đến việc hỗ trợ người dùng chẩn đoán bệnh cây trồng dựa trên triệu chứng văn bản hoặc hình ảnh, truy xuất thông tin điều trị từ kho tri thức và gợi ý các sản phẩm thuốc phù hợp, từ đó nâng cao hiệu quả sản xuất và giảm rủi ro trong quá trình sử dụng thuốc.

1.4 Mục tiêu cụ thể của đề án

Mục tiêu của đề án là xây dựng một hệ thống website hoạt động ổn định, có giao diện thân thiện và dễ sử dụng, đáp ứng tốt nhu cầu tra cứu và mua sắm của người dùng trong lĩnh vực nông nghiệp. Hệ thống cho phép người dùng xem, tìm kiếm, lọc và sắp xếp các sản phẩm thuốc bảo vệ thực vật, đồng thời hỗ trợ quản lý giỏ hàng, đặt hàng và theo dõi lịch sử mua hàng.

Bên cạnh đó, đề án tập trung phát triển Chatbot RAG thông minh có khả năng tư vấn bệnh cây trồng dựa trên dữ liệu văn bản và hình ảnh, giúp người dùng tiếp cận thông tin kỹ thuật nhanh chóng và chính xác. Đối với phía quản trị, hệ thống cung cấp các chức năng quản lý sản phẩm, đơn hàng, khách hàng và thống kê doanh thu, góp phần nâng cao hiệu quả vận hành cho cửa hàng vật tư nông nghiệp.

CHƯƠNG 2: NGHIÊN CỨU LÝ THUYẾT

2.1 Tổng quan về hệ thống Website tư vấn và bán thuốc bảo vệ thực vật

2.1.1 Giới thiệu hệ thống

Hệ thống được xây dựng là một nền tảng web thương mại điện tử kết hợp tư vấn kỹ thuật nông nghiệp, tập trung vào lĩnh vực thuốc bảo vệ thực vật. Mục tiêu của hệ thống là hỗ trợ người dùng, đặc biệt là nông dân và các hộ sản xuất nông nghiệp, trong việc nhận biết bệnh hại cây trồng, lựa chọn thuốc phù hợp và thực hiện mua sắm trực tuyến một cách thuận tiện.

Điểm nổi bật của hệ thống là tích hợp Chatbot thông minh theo mô hình RAG (Retrieval-Augmented Generation) sử dụng mô hình ngôn ngữ lớn Google Gemini. Chatbot có khả năng tiếp nhận mô tả triệu chứng hoặc hình ảnh cây trồng bị bệnh, sau đó truy xuất thông tin từ kho tri thức (Knowledge Base) để đưa ra tư vấn điều trị chính xác và dễ hiểu. Thông qua đó, hệ thống không chỉ đóng vai trò là một website bán hàng mà còn là công cụ hỗ trợ kỹ thuật, góp phần nâng cao hiệu quả sản xuất nông nghiệp và hạn chế việc sử dụng thuốc không đúng cách.

2.1.2 Các phân hệ chức năng

Hệ thống được chia thành hai phân hệ chính dựa trên vai trò và quyền hạn của người dùng, được kiểm soát thông qua cơ chế phân quyền (Role-based Authorization):

- Phân hệ Người dùng (Client-side):

Phân hệ này phục vụ khách hàng và người sử dụng cuối, cho phép người dùng truy cập các chức năng chính của hệ thống. Người dùng có thể xem danh sách sản phẩm thuốc bảo vệ thực vật, thực hiện tìm kiếm, lọc và sắp xếp sản phẩm theo tên, giá, hoạt chất hoặc loại bệnh điều trị. Bên cạnh đó, hệ thống hỗ trợ chức năng giỏ hàng, đặt hàng, quản lý sổ địa chỉ và theo dõi lịch sử đơn hàng.

Ngoài chức năng mua sắm, người dùng có thể tương tác với Chatbot tư vấn để mô tả triệu chứng bệnh cây trồng bằng văn bản hoặc hình ảnh. Chatbot sẽ phân tích thông tin, truy xuất dữ liệu từ kho tri thức và đưa ra gợi ý điều trị kèm theo các sản phẩm phù hợp có sẵn trong hệ thống.

- Phân hệ Quản trị (Admin-side):

Phân hệ quản trị dành cho người quản lý hệ thống, cho phép thực hiện các chức

năng quản lý toàn diện. Quản trị viên có thể thêm, sửa, xóa sản phẩm, quản lý thông tin thuốc bao gồm giá bán, tồn kho, hoạt chất và bệnh đặc trị. Hệ thống cũng hỗ trợ quản lý đơn hàng, cập nhật trạng thái xử lý và theo dõi lịch sử giao dịch.

Ngoài ra, phân hệ quản trị cung cấp các chức năng quản lý người dùng và thống kê, hiển thị tổng quan hoạt động kinh doanh thông qua các biểu đồ doanh thu, số lượng đơn hàng và sản phẩm bán chạy, hỗ trợ quá trình ra quyết định của người quản lý.

2.2 Kiến trúc MEAN Stack

2.2.1 Tổng quan kiến trúc MERN Stack

MERN Stack là một kiến trúc phát triển ứng dụng web full-stack hiện đại, sử dụng hoàn toàn ngôn ngữ JavaScript. MERN là chữ viết tắt của bốn công nghệ chính gồm MongoDB, Express.js, React.js và Node.js. Kiến trúc này cho phép xây dựng các ứng dụng web theo mô hình Client–Server, hỗ trợ phát triển các ứng dụng Single Page Application (SPA) với hiệu năng cao và khả năng mở rộng linh hoạt.

Việc lựa chọn MERN Stack giúp thống nhất ngôn ngữ lập trình xuyên suốt từ frontend đến backend, rút ngắn thời gian phát triển và thuận tiện việc bảo trì hệ thống.

2.2.2 Mô hình MVC trong kiến trúc MERN Stack

Trong hệ thống, kiến trúc MERN Stack được triển khai theo mô hình MVC phân tán. React.js đảm nhiệm vai trò View, chịu trách nhiệm hiển thị giao diện và tiếp nhận tương tác từ người dùng. Node.js kết hợp với Express.js đóng vai trò Controller, xử lý logic nghiệp vụ, xác thực người dùng và điều phối các luồng dữ liệu. MongoDB cùng với Mongoose Schema đảm nhiệm vai trò Model, lưu trữ và quản lý dữ liệu dưới dạng các document JSON.

Việc áp dụng mô hình MVC giúp hệ thống có cấu trúc rõ ràng, tách biệt giữa giao diện, xử lý nghiệp vụ và dữ liệu, từ đó nâng cao khả năng bảo trì và mở rộng trong tương lai.

2.2.3 Luồng xử lý dữ liệu trong hệ thống MERN

Khi người dùng thực hiện thao tác trên giao diện React, các yêu cầu được gửi đến server thông qua các RESTful API. Express.js tiếp nhận yêu cầu, thực hiện kiểm tra xác thực (JWT) và xử lý các logic nghiệp vụ tương ứng. Sau đó, server tương tác với cơ sở dữ liệu MongoDB để truy xuất hoặc cập nhật dữ liệu cần thiết.

Kết quả xử lý được trả về phía client dưới dạng JSON và React cập nhật lại giao diện người dùng. Riêng đối với chức năng tư vấn, hệ thống sẽ phân tích nội dung người dùng cung cấp, truy vấn kho tri thức (Knowledge Base) để lấy thông tin bệnh và hướng xử lý phù hợp, sau đó gửi dữ liệu này cho Gemini API nhằm tổng hợp và diễn đạt câu trả lời bằng ngôn ngữ tự nhiên trước khi phản hồi lại cho người dùng.

2.2.4 Đánh giá kiến trúc MERN Stack

Kiến trúc MERN Stack được đánh giá là phù hợp với các hệ thống web hiện đại có mức độ tương tác cao và yêu cầu xử lý dữ liệu linh hoạt. Việc sử dụng JavaScript làm ngôn ngữ thống nhất cho cả frontend và backend giúp giảm độ phức tạp trong phát triển, đồng thời tăng khả năng đồng bộ dữ liệu giữa các thành phần của hệ thống. MERN Stack cũng có ưu thế về tốc độ phát triển, khả năng mở rộng và cộng đồng hỗ trợ lớn.

Tuy nhiên, kiến trúc này cũng đòi hỏi người phát triển phải nắm vững kiến thức về lập trình bất đồng bộ, quản lý trạng thái phía client và tổ chức kiến trúc hệ thống hợp lý để tránh mã nguồn rời rạc. Với đặc thù là một website thương mại điện tử kết hợp tư vấn thông minh cho nông nghiệp, MERN Stack đáp ứng tốt yêu cầu về hiệu năng, tính linh hoạt và khả năng mở rộng lâu dài của đồ án.

2.3 Công nghệ và framework front-end

2.3.1 React.js

React.js là thư viện JavaScript mã nguồn mở do Facebook (Meta) phát triển, chuyên dùng để xây dựng giao diện người dùng cho các ứng dụng web hiện đại. React được thiết kế theo hướng component-based, cho phép chia nhỏ giao diện thành các thành phần độc lập, dễ tái sử dụng và dễ bảo trì. Nhờ đặc điểm này, React đặc biệt phù hợp với các ứng dụng Single Page Application (SPA), nơi giao diện cần cập nhật liên tục mà không làm tải lại toàn bộ trang.

React hoạt động dựa trên cơ chế Virtual DOM, giúp tối ưu hiệu năng bằng cách chỉ cập nhật những phần giao diện thực sự thay đổi. Trong đồ án, React.js được sử dụng để xây dựng toàn bộ giao diện phía người dùng, bao gồm trang danh sách sản phẩm, chi tiết sản phẩm, giỏ hàng, lịch sử đơn hàng và giao diện tương tác với chatbot tư vấn. React cũng đảm nhiệm việc xử lý các tương tác như đăng nhập, tìm kiếm, lọc dữ liệu và gửi yêu cầu đến backend thông qua API.

Về ưu điểm, React mang lại hiệu năng cao, khả năng tái sử dụng component tốt và hệ sinh thái thư viện phong phú. Tuy nhiên, do chỉ tập trung vào tầng giao diện, React yêu cầu kết hợp thêm các thư viện khác để xử lý định tuyến, quản lý trạng thái và tối ưu SEO.

2.3.2 Công nghệ TailwindCSS

TailwindCSS là framework CSS theo hướng utility-first, cho phép xây dựng giao diện bằng cách sử dụng trực tiếp các lớp tiện ích có sẵn. Thay vì viết CSS thủ công, lập trình viên có thể kết hợp các class để tạo giao diện linh hoạt và nhất quán.

Trong đồ án, TailwindCSS được sử dụng để thiết kế toàn bộ giao diện frontend, đảm bảo website có bố cục rõ ràng, hiện đại và hiển thị tốt trên nhiều kích thước màn hình. TailwindCSS giúp rút ngắn thời gian thiết kế giao diện, đồng thời tạo điều kiện dễ dàng chỉnh sửa và mở rộng về sau.

2.4 Công nghệ và framework back-end

2.4.1 Tổng quan Back-end của hệ thống

Phần Back-end của hệ thống đóng vai trò trung tâm trong việc xử lý nghiệp vụ, quản lý dữ liệu và kết nối giữa giao diện người dùng với các dịch vụ bên ngoài. Back-end được xây dựng theo mô hình Client – Server, trong đó server chịu trách nhiệm tiếp nhận yêu cầu từ frontend, xử lý logic nghiệp vụ, truy xuất cơ sở dữ liệu và trả kết quả về cho client dưới dạng JSON.

Hệ thống Back-end của đồ án sử dụng Node.js kết hợp Express.js để xây dựng các RESTful API và MongoDB làm hệ quản trị cơ sở dữ liệu NoSQL. Kiến trúc này phù hợp với các ứng dụng web hiện đại có lượng truy cập lớn, yêu cầu xử lý bất đồng bộ và khả năng mở rộng cao.

2.4.2 Công nghệ Node.js

Node.js là môi trường thực thi JavaScript phía máy chủ, được xây dựng trên nền tảng V8 JavaScript Engine của Google Chrome. Node.js cho phép lập trình viên sử dụng cùng một ngôn ngữ JavaScript cho cả frontend và backend, giúp giảm độ phức tạp trong quá trình phát triển hệ thống.

Node.js hoạt động theo mô hình Event-driven và cơ chế Non-blocking I/O, cho phép xử lý nhiều yêu cầu đồng thời mà không cần tạo luồng mới cho mỗi request. Trong

đề án, Node.js đảm nhiệm vai trò chạy server, tiếp nhận các yêu cầu HTTP từ frontend và phối hợp với Express.js để xử lý nghiệp vụ như đăng nhập, quản lý sản phẩm, đơn hàng và chatbot tư vấn.

2.4.3 Công nghệ Công nghệ Express.js

Express.js là framework web tối giản và linh hoạt dành cho Node.js, đóng vai trò là lớp trung gian giúp đơn giản hóa việc xây dựng server và API. Express cung cấp các cơ chế quan trọng như Routing, Middleware và Error Handling, giúp tổ chức luồng xử lý request–response một cách rõ ràng.

Trong đề án, Express.js được sử dụng để xây dựng hệ thống RESTful API phục vụ các chức năng như xác thực người dùng (JWT), quản lý sản phẩm, giỏ hàng, đơn hàng, quản trị hệ thống và giao tiếp với chatbot RAG. Các API được tổ chức theo mô hình MVC với các thư mục routes, controllers và middleware, giúp mã nguồn dễ bảo trì và mở rộng.

2.5 Chatbot tư vấn thông minh RAG tích hợp Google Gemini

2.5.1 Tổng quan

Chatbot tư vấn thông minh là chức năng nổi bật của hệ thống, được xây dựng nhằm hỗ trợ người dùng chẩn đoán bệnh hại cây trồng và đề xuất hướng xử lý phù hợp. Chatbot cho phép người dùng nhập mô tả triệu chứng bằng văn bản hoặc gửi hình ảnh cây trồng để nhận tư vấn trực tiếp trên website.

Hệ thống chatbot được triển khai theo mô hình RAG (Retrieval-Augmented Generation), kết hợp giữa việc truy xuất dữ liệu từ kho tri thức nội bộ và khả năng tổng hợp ngôn ngữ tự nhiên của Google Gemini. Thay vì để AI tự suy đoán toàn bộ nội dung, hệ thống ưu tiên sử dụng các thông tin đã được kiểm chứng trong Knowledge Base để đảm bảo độ chính xác và phù hợp với thực tế sản xuất nông nghiệp tại Việt Nam.

2.5.2 Kho tri thức và vai trò của Google Gemini

Kho tri thức (Knowledge Base) được xây dựng và lưu trữ trong cơ sở dữ liệu MongoDB, bao gồm các thông tin về bệnh hại cây trồng như triệu chứng, nguyên nhân, hướng xử lý và các hoạt chất thuốc bảo vệ thực vật khuyến nghị. Đây là nguồn dữ liệu chính để hệ thống truy xuất khi người dùng yêu cầu tư vấn.

2.5.3 Kho tri thức và vai trò của Google Gemini

Khi người dùng gửi triệu chứng, dữ liệu được chuyển từ giao diện React đến server thông qua RESTful API. Server xử lý yêu cầu, xác định bệnh hại liên quan và truy xuất thông tin điều trị từ MongoDB. Các dữ liệu này sau đó được gửi đến Google Gemini để tổng hợp thành câu trả lời hoàn chỉnh và trả về giao diện người dùng.

Việc áp dụng Chatbot RAG giúp hệ thống cung cấp thông tin tư vấn chính xác, nhất quán và dễ tiếp cận, góp phần hỗ trợ người nông dân sử dụng thuốc bảo vệ thực vật đúng cách, hạn chế rủi ro trong sản xuất và giảm tác động tiêu cực đến môi trường. Đồng thời, chatbot nâng cao trải nghiệm người dùng và tạo nên điểm khác biệt công nghệ cho đề án.

2.6 Hệ quản trị cơ sở dữ liệu

2.6.1 Giới thiệu hệ quản trị cơ sở dữ liệu MongoDB

MongoDB là hệ quản trị cơ sở dữ liệu NoSQL, lưu trữ dữ liệu dưới dạng document theo cấu trúc JSON. Ưu điểm nổi bật của MongoDB là khả năng mở rộng cao, truy vấn linh hoạt và không yêu cầu lược đồ cứng nhắc. Điều này rất phù hợp với hệ thống có dữ liệu đa dạng như thông tin sản phẩm, lịch sử đơn hàng và dữ liệu tư vấn bệnh cây trồng.

Trong đề án, MongoDB được sử dụng để lưu trữ các collection chính như: User, Product, Order, KnowledgeBase và ChatSession. Cách tổ chức này giúp hệ thống dễ dàng mở rộng khi bổ sung thêm tính năng mới.

2.6.2 Giới thiệu thư viện Mongoose

Mongoose là thư viện ODM (Object Data Modeling) cho MongoDB trong môi trường Node.js. Mongoose cho phép định nghĩa Schema, ràng buộc kiểu dữ liệu và kiểm tra tính hợp lệ của dữ liệu trước khi lưu vào cơ sở dữ liệu. Nhờ đó, dữ liệu trong hệ thống luôn đảm bảo tính nhất quán và hạn chế lỗi phát sinh trong quá trình xử lý.

Mongoose cũng hỗ trợ các middleware và phương thức truy vấn nâng cao, giúp việc thao tác dữ liệu trong backend trở nên rõ ràng, dễ bảo trì và thuận tiện cho việc phát triển lâu dài.

2.6.3 Ứng dụng trong đề án

Hệ quản trị cơ sở dữ liệu chịu trách nhiệm lưu trữ và cung cấp dữ liệu cho toàn

bộ hệ thống. Đối với chức năng bán hàng, cơ sở dữ liệu hỗ trợ quản lý sản phẩm, tồn kho, đơn hàng và khách hàng. Đối với chức năng tư vấn, MongoDB đóng vai trò là kho tri thức, lưu trữ thông tin bệnh hại, triệu chứng, hoạt chất và hướng xử lý để Chatbot RAG truy xuất và tổng hợp câu trả lời chính xác cho người dùng.

2.7 Công cụ hỗ trợ phát triển và triển khai

2.7.1 Công cụ phát triển mã nguồn Visual Studio Code (VS Code)

Trong quá trình xây dựng hệ thống, Visual Studio Code (VS Code) được sử dụng làm môi trường phát triển chính cho cả Frontend và Backend. Đây là công cụ lập trình nhẹ, mạnh mẽ, hỗ trợ tốt cho các công nghệ sử dụng trong đề án như React.js, Node.js và Express.js. VS Code cung cấp nhiều tiện ích mở rộng giúp kiểm tra cú pháp, định dạng mã nguồn và phát hiện lỗi sớm, nâng cao chất lượng code và hiệu quả làm việc.

2.7.2 Công cụ quản lý mã nguồn GitHub

GitHub là nền tảng quản lý mã nguồn được sử dụng để lưu trữ, theo dõi và quản lý các phiên bản của hệ thống. Việc sử dụng GitHub giúp nhóm phát triển dễ dàng quản lý lịch sử thay đổi mã nguồn, phối hợp làm việc và khôi phục phiên bản khi cần thiết.

Bên cạnh đó, GitHub Actions được sử dụng để tự động hóa quy trình tích hợp và triển khai liên tục (CI/CD). Thông qua các workflow được cấu hình sẵn, hệ thống có thể tự động thực hiện các bước như kiểm tra mã nguồn, build ứng dụng và triển khai lên môi trường server mỗi khi có thay đổi được đẩy lên repository. Điều này giúp giảm thiểu sai sót do thao tác thủ công và nâng cao độ tin cậy của quá trình triển khai.

CHƯƠNG 3: HIỆN THỰC HÓA NGHIÊN CỨU

3.1 Mô tả bài toán và mục tiêu hệ thống

Trong bối cảnh chuyển đổi số trong lĩnh vực nông nghiệp còn nhiều hạn chế, việc tiếp cận thông tin chính xác về thuốc bảo vệ thực vật và cách phòng trừ bệnh hại cây trồng của người dân vẫn chủ yếu dựa vào kinh nghiệm cá nhân hoặc tư vấn truyền thống tại cửa hàng. Điều này dễ dẫn đến việc sử dụng thuốc không đúng kỹ thuật, gây ảnh hưởng đến năng suất cây trồng, môi trường và sức khỏe con người.

Từ thực tế đó, hệ thống được xây dựng nhằm cung cấp một nền tảng trực tuyến tích hợp giữa bán thuốc bảo vệ thực vật và tư vấn bệnh hại cây trồng thông minh. Website cho phép người dùng tra cứu, lọc và mua sản phẩm, đồng thời nhận tư vấn kỹ thuật thông qua chatbot tích hợp trí tuệ nhân tạo.

Mục tiêu chính của hệ thống là xây dựng một website thương mại điện tử chuyên ngành nông nghiệp, hỗ trợ người dùng tiếp cận thông tin chính xác, nhanh chóng và đáng tin cậy về thuốc bảo vệ thực vật. Bên cạnh đó, hệ thống hướng đến việc hỗ trợ nông dân trong việc chẩn đoán bệnh cây trồng và đề xuất hướng xử lý phù hợp, góp phần nâng cao hiệu quả sản xuất nông nghiệp.

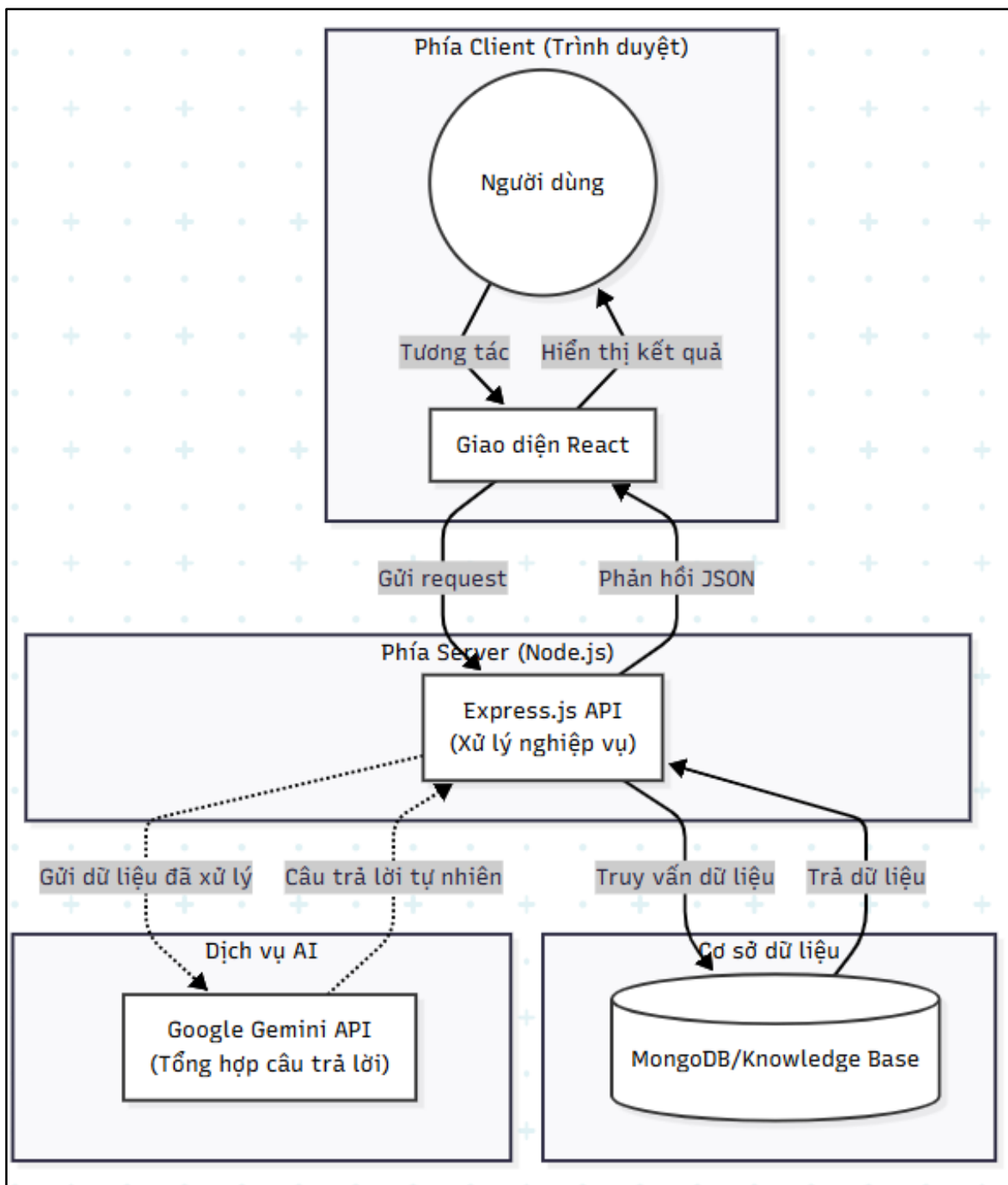
3.2 Quy trình nghiệp vụ tổng quát

Hệ thống vận hành theo mô hình Client–Server với các quy trình nghiệp vụ chính như sau:

Đối với người dùng, hệ thống cho phép đăng ký và đăng nhập tài khoản để sử dụng đầy đủ các chức năng. Sau khi xác thực thành công, người dùng có thể duyệt danh sách sản phẩm, lọc theo danh mục hoặc hoạt chất, thêm sản phẩm vào giỏ hàng và thực hiện đặt hàng trực tuyến. Toàn bộ thông tin đơn hàng được lưu trữ và quản lý tập trung trên hệ thống.

Song song với chức năng bán hàng, hệ thống cung cấp quy trình tư vấn bệnh cây trồng. Người dùng nhập mô tả triệu chứng hoặc tải hình ảnh cây trồng bị bệnh. Dữ liệu này được gửi đến server để xử lý, kết hợp với kho tri thức (Knowledge Base) và dịch vụ AI Google Gemini nhằm đưa ra kết luận về bệnh và đề xuất thuốc điều trị phù hợp.

Đối với quản trị viên, hệ thống hỗ trợ quản lý sản phẩm, quản lý đơn hàng, theo dõi khách hàng và thống kê hoạt động kinh doanh. Các thao tác quản trị được kiểm soát chặt chẽ thông qua cơ chế phân quyền.



Hình 2.1 Sơ đồ hoạt động của hệ thống

3.3 Đối tượng sử dụng

Hệ thống “Tư vấn và bán thuốc bảo vệ thực vật thông minh” được thiết kế nhằm phục vụ nhiều nhóm người dùng khác nhau, mỗi nhóm có vai trò và quyền hạn riêng biệt. Việc phân chia đối tượng sử dụng giúp hệ thống vận hành hiệu quả, đảm bảo đúng chức năng và tăng tính bảo mật trong quá trình khai thác.

3.3.1 Người dùng thông thường (Khách hàng)

Người dùng thông thường là các cá nhân có nhu cầu tìm kiếm, mua sắm thuốc bảo vệ thực vật và nhận tư vấn về bệnh hại trên cây trồng. Đối tượng này bao gồm nông dân, hộ sản xuất nông nghiệp và những người quan tâm đến kỹ thuật canh tác.

Sau khi đăng ký và đăng nhập tài khoản, người dùng có thể sử dụng chức năng mua sản phẩm. Ngoài ra, người dùng còn có thể sử dụng chức năng chatbot để nhập mô tả triệu chứng hoặc hình ảnh cây trồng nhằm nhận tư vấn bệnh và đề xuất thuốc điều trị phù hợp.

3.3.2 Quản trị viên hệ thống (Administrator)

Khách truy cập là những người chưa đăng ký tài khoản nhưng có nhu cầu tìm hiểu thông tin chung về hệ thống. Đối tượng này chỉ được phép xem các nội dung công khai như danh sách sản phẩm và thông tin giới thiệu, nhưng không thể thực hiện các thao tác như đặt hàng hay sử dụng chatbot tư vấn.

Việc cho phép khách truy cập tiếp cận nội dung cơ bản giúp tăng khả năng tiếp cận người dùng mới và khuyến khích họ đăng ký tài khoản để sử dụng đầy đủ các chức năng của hệ thống.

3.3.3 Khách truy cập (Visitor)

Khách truy cập là những người chưa đăng ký tài khoản nhưng có nhu cầu tìm hiểu thông tin chung về hệ thống. Đối tượng này chỉ được phép xem các nội dung công khai như danh sách sản phẩm và thông tin giới thiệu, nhưng không thể thực hiện các thao tác như đặt hàng hay sử dụng chatbot tư vấn.

Việc cho phép khách truy cập tiếp cận nội dung cơ bản giúp tăng khả năng tiếp cận người dùng mới và khuyến khích họ đăng ký tài khoản để sử dụng đầy đủ các chức năng của hệ thống.

3.4 Phân tích và thiết kế hệ thống

3.4.1 Yêu cầu chức năng

3.4.1.1 Chức năng dành cho người dùng

Hệ thống cho phép người dùng đăng ký tài khoản mới và đăng nhập để sử dụng đầy đủ các chức năng. Sau khi xác thực thành công, người dùng có thể truy cập và sử dụng các chức năng sau:

- Xem danh sách sản phẩm thuốc bảo vệ thực vật được cung cấp trên hệ thống.
- Tìm kiếm và lọc sản phẩm theo danh mục, tên sản phẩm hoặc hoạt chất.
- Xem thông tin sản phẩm, bao gồm mô tả, giá bán, hoạt chất và mô tả.
- Thêm sản phẩm vào giỏ hàng, cập nhật hoặc xóa sản phẩm khỏi giỏ hàng.
- Thực hiện đặt hàng trực tuyến và lưu trữ thông tin đơn hàng trên hệ thống.
- Xem lịch sử đơn hàng và theo dõi trạng thái xử lý đơn hàng.
- Sử dụng chức năng tư vấn bệnh cây trồng bằng cách nhập mô tả triệu chứng hoặc tải hình ảnh cây trồng bị bệnh.
- Nhận kết quả tư vấn về bệnh và đề xuất thuốc điều trị thông qua Chatbot RAG tích hợp Google Gemini.

3.4.1.2 Chức năng dành cho quản trị viên

Đối với quản trị viên, hệ thống cung cấp các chức năng quản lý và giám sát toàn bộ hoạt động của website, bao gồm:

- Quản lý sản phẩm: thêm mới, chỉnh sửa, xóa sản phẩm, cập nhật hình ảnh và cập nhật số lượng tồn kho.
- Quản lý đơn hàng: xem danh sách đơn hàng, cập nhật trạng thái xử lý và theo dõi quá trình giao dịch.
- Quản lý người dùng: quản lý danh sách và thông tin người dùng.
- Quản lý kho tri thức: phục vụ cho chức năng tư vấn bệnh cây trồng.
- Theo dõi và thống kê hoạt động kinh doanh, bao gồm số lượng đơn hàng và doanh thu theo thời gian.

3.4.1.3 Chức năng dành cho khách truy cập

Đối với khách truy cập chưa đăng ký tài khoản, hệ thống cho phép Xem thông tin giới thiệu về hệ thống và các sản phẩm cơ bản.

Các chức năng nâng cao như đặt hàng và tư vấn bệnh chỉ được kích hoạt khi người dùng đăng nhập

3.4.2 Yêu cầu phi chức năng

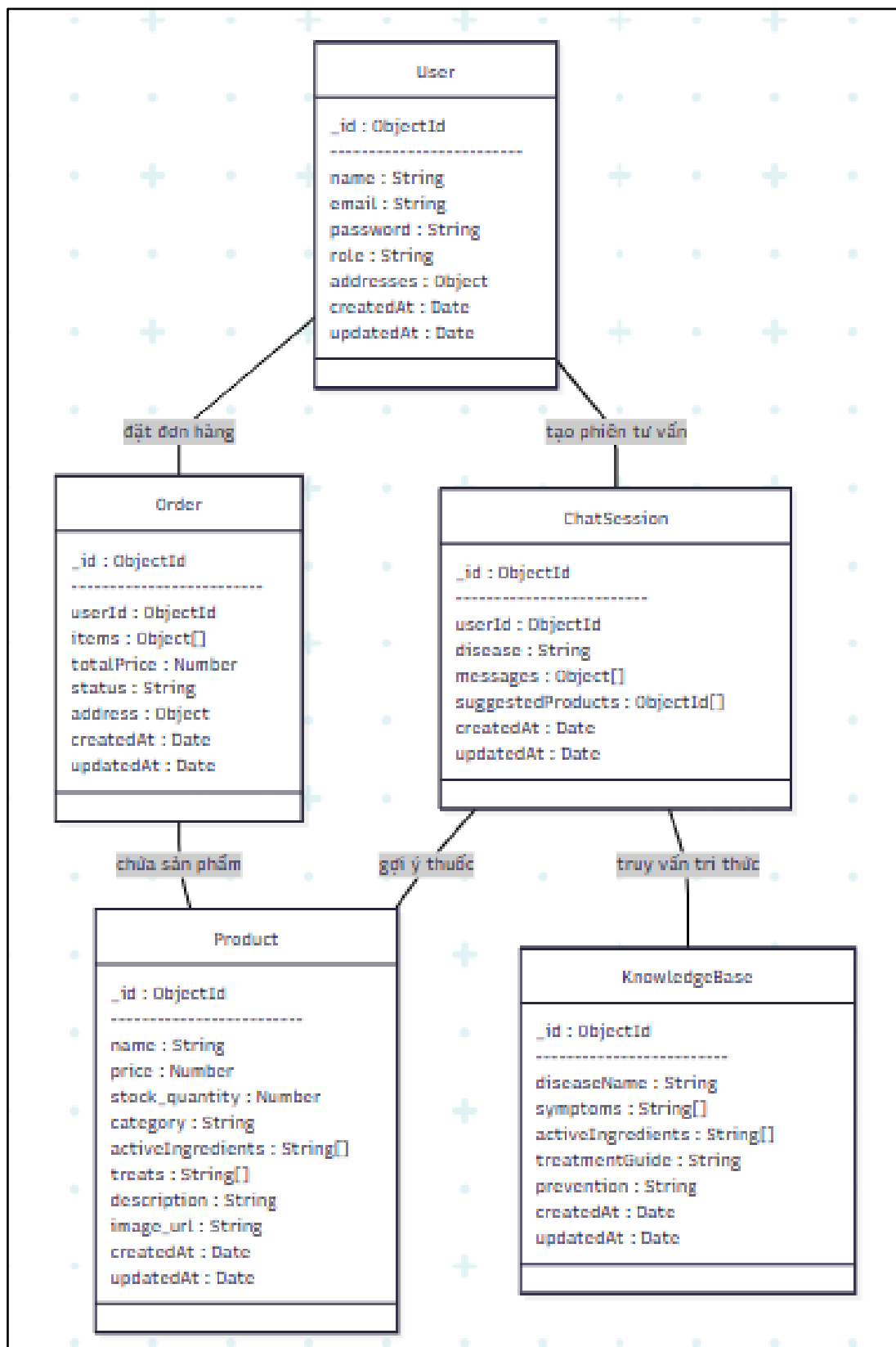
Hệ thống phải đảm bảo hiệu năng tốt với thời gian phản hồi nhanh và khả năng phục vụ nhiều người dùng đồng thời. Về bảo mật, hệ thống sử dụng cơ chế xác thực bằng JSON Web Token (JWT) và mã hóa mật khẩu để bảo vệ thông tin người dùng.

Giao diện được thiết kế thân thiện, dễ sử dụng và tương thích trên nhiều thiết bị khác nhau. Ngoài ra, hệ thống cần hoạt động ổn định, có khả năng mở rộng và đảm bảo an toàn dữ liệu trong suốt quá trình sử dụng.

3.5 Thiết kế cơ sở dữ liệu

Cơ sở dữ liệu của hệ thống được thiết kế theo mô hình NoSQL, sử dụng MongoDB làm hệ quản trị cơ sở dữ liệu chính. Việc lựa chọn MongoDB phù hợp với đặc thù của hệ thống vì dữ liệu có cấu trúc linh hoạt, thường xuyên thay đổi và cần khả năng mở rộng cao, đặc biệt đối với các chức năng thương mại điện tử và chatbot tư vấn bệnh cây trồng.

Dữ liệu trong hệ thống được tổ chức dưới dạng các collection, mỗi collection tương ứng với một đối tượng nghiệp vụ cụ thể. Các collection liên kết với nhau thông qua các trường khóa tham chiếu (ObjectId), đảm bảo vừa linh hoạt trong lưu trữ, vừa thuận tiện trong truy vấn và mở rộng về sau.



Hình 2.2 Mô hình ERD của cơ sở dữ liệu

3.5.1 Bảng người dùng (User)

Collection User dùng để lưu trữ thông tin tài khoản của người sử dụng hệ thống, bao gồm cả người dùng thông thường và quản trị viên. Mỗi người dùng được định danh duy nhất bằng trường `_id`. Ngoài các thông tin cơ bản như họ tên, email và mật khẩu đã được mã hóa, bảng User còn lưu vai trò người dùng (role) để phục vụ cho cơ chế phân quyền. Thông tin địa chỉ được lưu dưới dạng đối tượng, giúp hỗ trợ chức năng đặt hàng và số địa chỉ. Các trường `createdAt` và `updatedAt` giúp theo dõi thời gian tạo và cập nhật tài khoản.

Tên trường	Kiểu dữ liệu	Mô tả
<code>_id</code>	ObjectId	Khóa chính, định danh người dùng
<code>name</code>	String	Họ tên người dùng
<code>email</code>	String	Email đăng nhập (duy nhất)
<code>password</code>	String	Mật khẩu đã mã hóa
<code>role</code>	String	Vai trò (user, admin)
<code>addresses</code>	Object	Thông tin địa chỉ giao hàng
<code>createdAt</code>	Date	Ngày tạo tài khoản
<code>updatedAt</code>	Date	Ngày cập nhật gần nhất

Bảng 3.1 Các trường trong bảng user

3.5.2 Bảng sản phẩm (Product)

Collection Product lưu trữ toàn bộ thông tin về các sản phẩm thuốc bảo vệ thực vật được kinh doanh trên hệ thống. Mỗi sản phẩm bao gồm tên, giá bán, số lượng tồn kho, danh mục và mô tả chi tiết. Đặc biệt, các trường `activeIngredients` và `treats` được thiết kế dưới dạng mảng, cho phép một sản phẩm có thể chứa nhiều hoạt chất và điều trị được nhiều loại bệnh khác nhau. Giúp hệ thống dễ dàng thực hiện chức năng lọc sản phẩm theo hoạt chất hoặc theo bệnh, đồng thời hỗ trợ chatbot gợi ý thuốc phù hợp.

Tên trường	Kiểu dữ liệu	Mô tả
<code>_id</code>	ObjectId	Khóa chính, định danh người dùng
<code>name</code>	String	Họ tên người dùng
<code>email</code>	String	Email đăng nhập (duy nhất)
<code>password</code>	String	Mật khẩu đã mã hóa
<code>role</code>	String	Vai trò (user, admin)
<code>addresses</code>	Object	Thông tin địa chỉ giao hàng
<code>createdAt</code>	Date	Ngày tạo tài khoản
<code>updatedAt</code>	Date	Ngày cập nhật gần nhất

Bảng 3.2 Các trường trong bảng Product

3.5.3 Bảng đơn hàng (Order)

Collection **Order** dùng để quản lý toàn bộ đơn hàng phát sinh trong hệ thống. Mỗi đơn hàng liên kết với một người dùng thông qua trường `userId`. Danh sách sản phẩm trong đơn hàng được lưu trong trường `items` dưới dạng mảng đối tượng, bao gồm thông tin sản phẩm, số lượng và giá tại thời điểm đặt hàng. Ngoài ra, bảng `Order` còn lưu tổng giá trị đơn hàng, trạng thái xử lý (chờ xử lý, đã xác nhận, đã giao, hủy đơn) và thông tin địa chỉ giao hàng. Thiết kế này giúp hệ thống dễ dàng theo dõi vòng đời của đơn hàng và phục vụ công tác quản lý của quản trị viên.

Tên trường	Kiểu dữ liệu	Mô tả
<code>_id</code>	<code>ObjectId</code>	Khóa chính
<code>userId</code>	<code>ObjectId</code>	Tham chiếu đến User
<code>items</code>	<code>Object[]</code>	Danh sách sản phẩm trong đơn
<code>totalPrice</code>	<code>Number</code>	Tổng giá trị đơn hàng
<code>status</code>	<code>String</code>	Trạng thái đơn hàng
<code>address</code>	<code>Object</code>	Địa chỉ giao hàng
<code>createdAt</code>	<code>Date</code>	Ngày tạo đơn
<code>updatedAt</code>	<code>Date</code>	Ngày cập nhật

Bảng 3.3 Các trường trong bảng Order

3.5.4 Bảng phiên tư vấn (ChatSession)

Collection **ChatSession** đại diện cho mỗi phiên tư vấn bệnh cây trồng giữa người dùng và hệ thống chatbot. Mỗi phiên tư vấn gắn với một người dùng thông qua `userId` và có thể lưu thông tin bệnh được xác định trong quá trình tư vấn. Trường `messages` lưu lại toàn bộ nội dung hội thoại trong phiên, trong khi `suggestedProducts` dùng để lưu danh sách các sản phẩm được chatbot gợi ý. Thiết kế này cho phép hệ thống vừa phục vụ tư vấn tức thời, vừa có thể phân tích lại lịch sử tư vấn khi cần.

Tên trường	Kiểu dữ liệu	Mô tả
<code>_id</code>	<code>ObjectId</code>	Khóa chính
<code>userId</code>	<code>ObjectId</code>	Người dùng bắt đầu tư vấn
<code>disease</code>	<code>String</code>	Tên bệnh được xác định
<code>messages</code>	<code>Object[]</code>	Nội dung hội thoại
<code>suggestedProducts</code>	<code>ObjectId[]</code>	Sản phẩm được gợi ý
<code>createdAt</code>	<code>Date</code>	Thời điểm bắt đầu
<code>updatedAt</code>	<code>Date</code>	Thời điểm cập nhật

Bảng 3.4 Các trường trong bảng ChatSession

3.5.5 Bảng kho tri thức (KnowledgeBase)

Collection **KnowledgeBase** đóng vai trò là kho tri thức trung tâm của hệ thống tư vấn thông minh. Bảng này lưu trữ thông tin về các loại bệnh trên cây trồng, bao gồm tên bệnh, danh sách triệu chứng, các hoạt chất phù hợp để điều trị, hướng dẫn điều trị và biện pháp phòng ngừa.

Dữ liệu trong **KnowledgeBase** được chatbot truy vấn trong quá trình tư vấn theo

mô hình RAG (Retrieval-Augmented Generation), kết hợp với Google Gemini để tổng hợp và trả lời người dùng một cách chính xác và dễ hiểu.

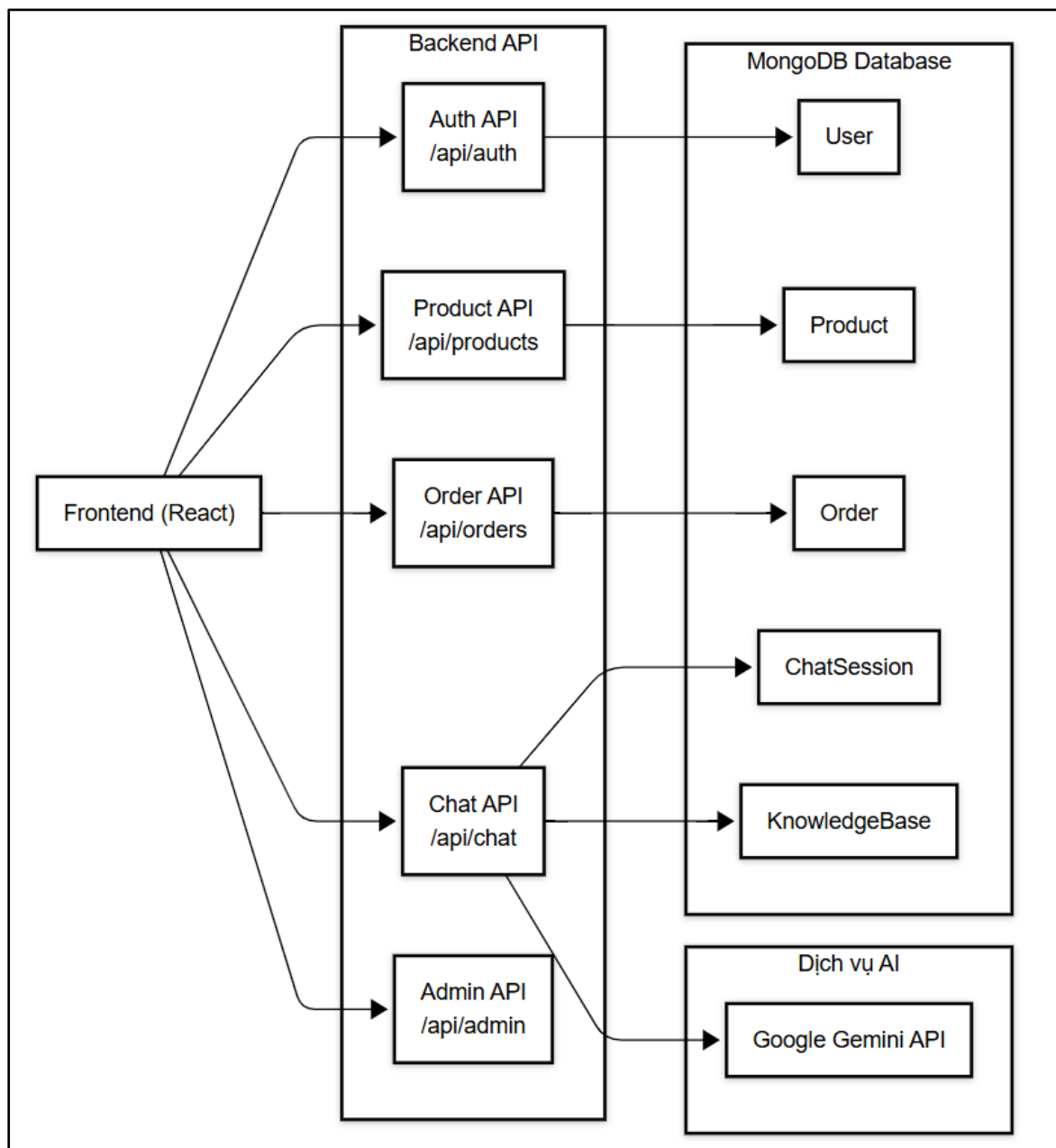
Tên trường	Kiểu dữ liệu	Mô tả
_id	ObjectId	Khóa chính
diseaseName	String	Tên bệnh
symptoms	String[]	Danh sách triệu chứng
activeIngredients	String[]	Hoạt chất điều trị
treatmentGuide	String	Hướng dẫn điều trị
prevention	String	Biện pháp phòng ngừa
createdAt	Date	Ngày tạo
updatedAt	Date	Ngày cập nhật

Bảng 3.5 Các trường trong bảng ChatSession

3.6 Phân tích và thiết kế hệ thống

3.6.1 Phân tích và thiết kế Backend(Giao diện người dùng)

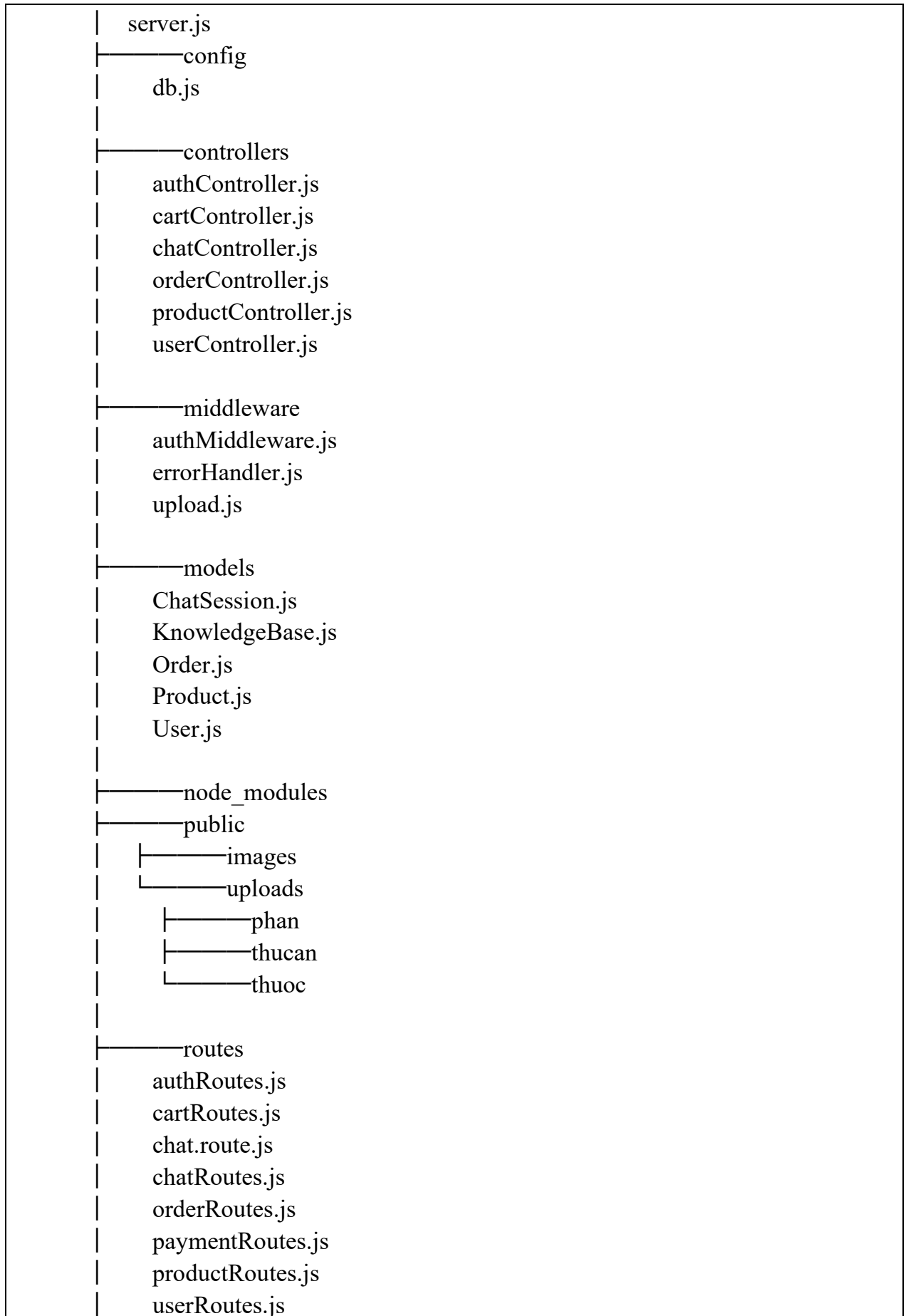
Backend của hệ thống được xây dựng trên nền tảng Node.js kết hợp với Express.js, đóng vai trò xử lý toàn bộ logic nghiệp vụ và quản lý dữ liệu. Backend cung cấp các API phục vụ cho các chức năng chính như quản lý người dùng, sản phẩm, đơn hàng và tư vấn bệnh cây trồng.



Hình 2.3 Sơ đồ truy vấn API của Backend

Hệ thống backend được tổ chức theo mô hình phân tầng, bao gồm các thành phần như routes, controllers và models, giúp mã nguồn rõ ràng và dễ bảo trì. Mỗi yêu cầu từ frontend được backend tiếp nhận, kiểm tra xác thực người dùng (JWT), xử lý nghiệp vụ tương ứng và tương tác với cơ sở dữ liệu MongoDB để truy xuất hoặc cập nhật dữ liệu.

Cấu trúc thư mục Backend:



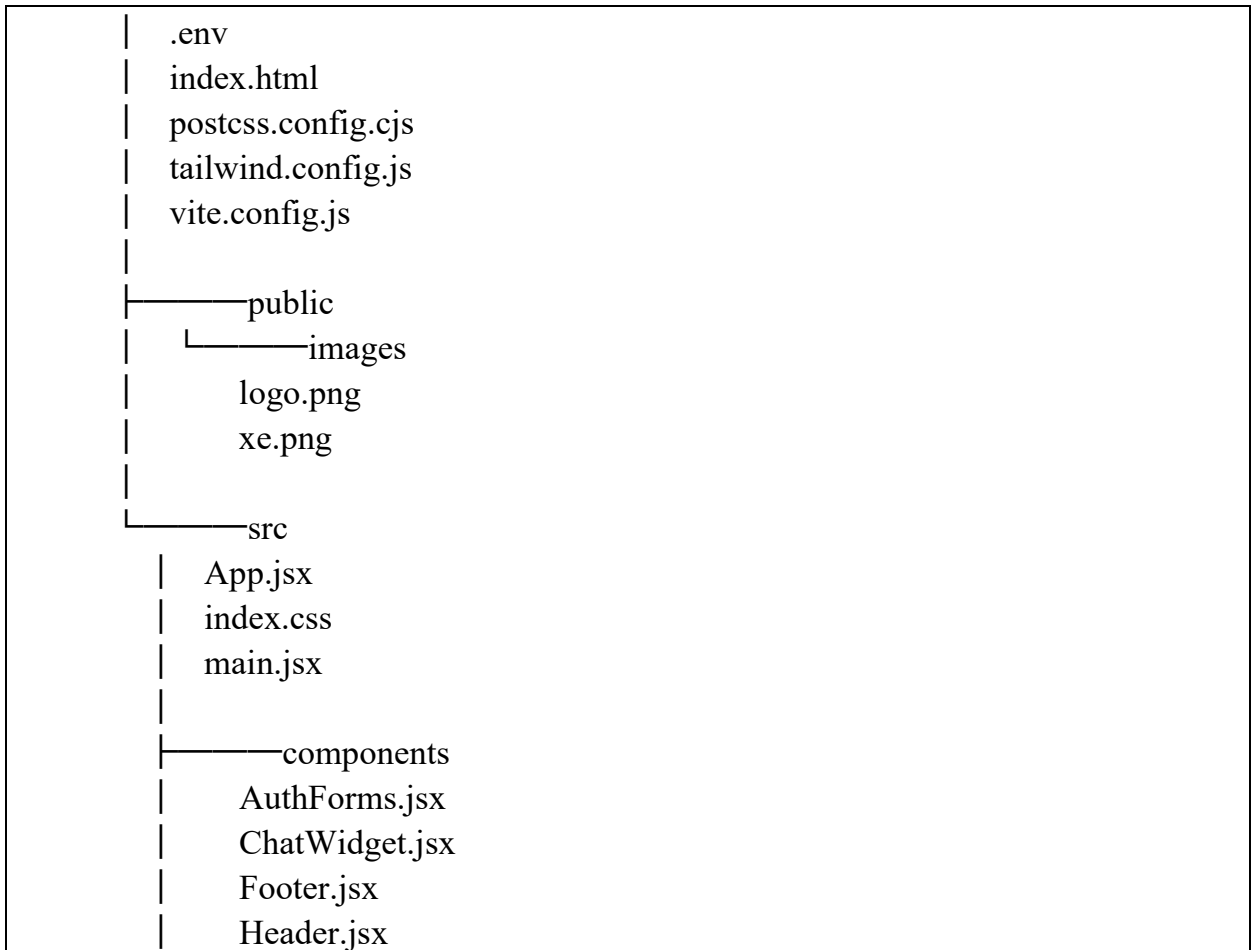
3.6.2 Phân tích và thiết kế Frontend (Giao diện người dùng)

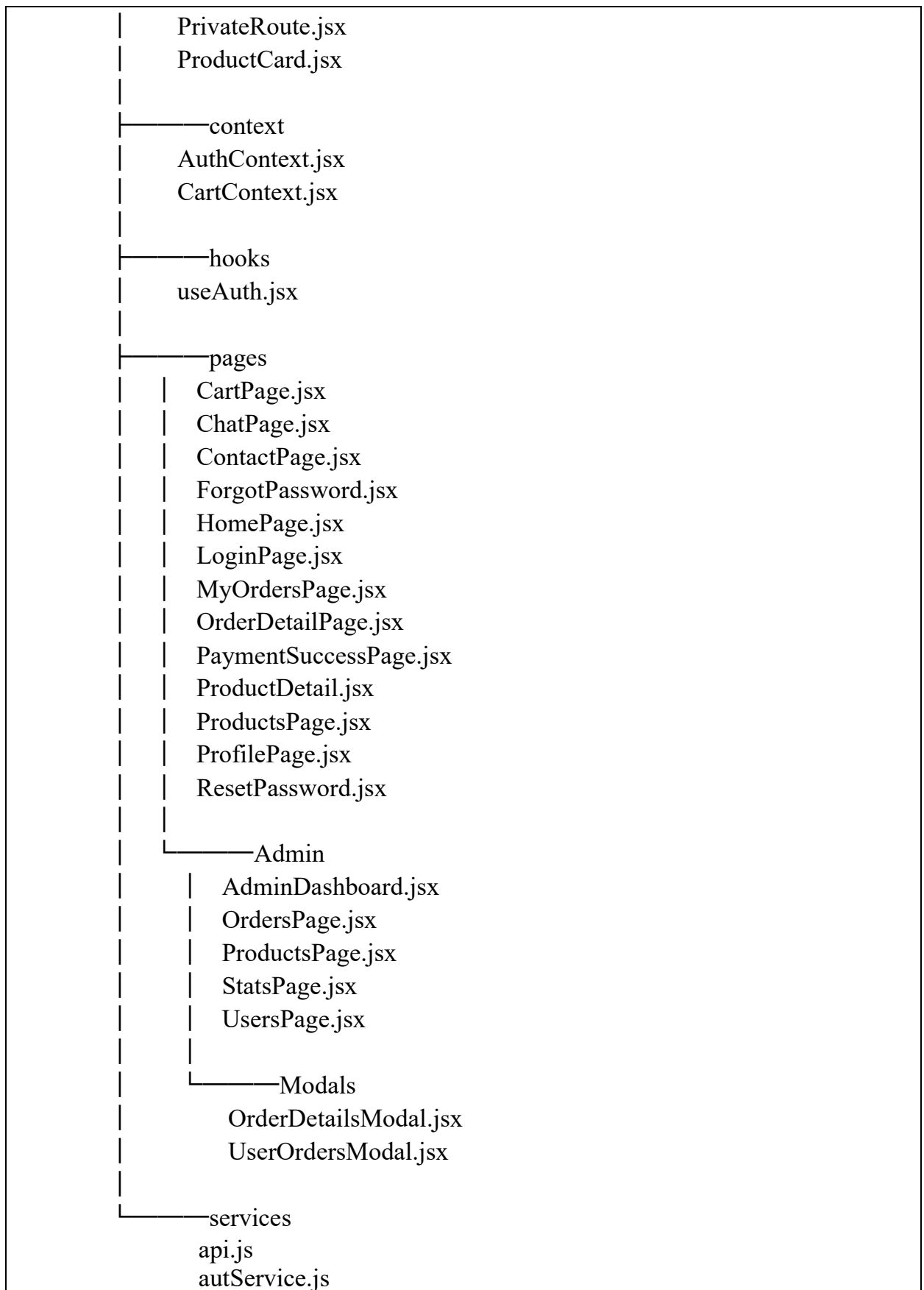
Frontend của hệ thống được xây dựng bằng React.js, đóng vai trò là lớp giao tiếp trực tiếp giữa người dùng và hệ thống. Giao diện được thiết kế theo hướng thân thiện, dễ sử dụng, phù hợp với người dùng trong lĩnh vực nông nghiệp, kể cả những người không am hiểu nhiều về công nghệ.

Hệ thống frontend đảm nhiệm các chức năng chính như hiển thị danh sách sản phẩm, cho phép tìm kiếm và lọc thuốc theo danh mục hoặc hoạt chất, quản lý giỏ hàng và thực hiện đặt hàng trực tuyến. Ngoài ra, giao diện chatbot được tích hợp trực tiếp trong website, giúp người dùng dễ dàng gửi câu hỏi và nhận tư vấn mà không cần chuyển sang nền tảng khác.

Dữ liệu từ frontend được gửi đến backend thông qua các RESTful API dưới định dạng JSON. Nhờ mô hình Single Page Application (SPA), các thao tác như tìm kiếm sản phẩm, gửi câu hỏi chatbot hay thêm sản phẩm vào giỏ hàng đều được xử lý mà không cần tải lại toàn bộ trang, góp phần nâng cao trải nghiệm người dùng.

Cấu trúc thư mục Frontend:





3.6.3 Phân tích và thiết kế Frontend (Giao diện người dùng)

3.6.3.1 Vai trò và mục tiêu của Chatbot trong hệ thống

Chatbot tư vấn bệnh cây trồng là thành phần trung tâm, tạo nên tính “thông minh” và giá trị thực tiễn cho hệ thống. Mục tiêu chính của chatbot không chỉ dừng lại ở việc trả lời câu hỏi thông thường mà hướng đến hỗ trợ người nông dân nhận diện bệnh cây trồng và đề xuất phương án xử lý phù hợp một cách nhanh chóng và dễ hiểu.

Trong hệ thống này, chatbot đóng vai trò như một trợ lý tư vấn nông nghiệp ảo, giúp người dùng:

- Mô tả triệu chứng bệnh bằng ngôn ngữ tự nhiên.
- Nhận được kết luận về bệnh dựa trên dữ liệu chuyên môn đã được lưu trữ.
- Được gợi ý thuốc bảo vệ thực vật phù hợp với từng loại bệnh.

Việc tích hợp chatbot giúp giảm sự phụ thuộc vào tư vấn trực tiếp tại cửa hàng, đồng thời hạn chế tình trạng sử dụng thuốc không đúng bệnh, không đúng hoạt chất.

3.6.3.2 Lý do lựa chọn mô hình RAG (Retrieval-Augmented Generation)

Trong lĩnh vực nông nghiệp, đặc biệt là tư vấn bệnh cây trồng, độ chính xác của thông tin là yếu tố quan trọng hàng đầu. Các mô hình ngôn ngữ lớn (LLM) như Google Gemini có khả năng diễn đạt rất tốt, tuy nhiên nếu chỉ sử dụng AI thuần túy sẽ tiềm ẩn nguy cơ trả lời sai lệch hoặc không phù hợp với điều kiện thực tế.

Vì vậy, hệ thống áp dụng mô hình RAG kết hợp giữa:

- Retrieval (Truy xuất dữ liệu): Tìm kiếm thông tin bệnh và hướng điều trị từ kho tri thức nội bộ (Knowledge Base).
- Generation (Sinh ngôn ngữ): Sử dụng Google Gemini để tổng hợp và diễn đạt câu trả lời một cách tự nhiên, dễ hiểu.

Cách tiếp cận này giúp đảm bảo rằng nội dung tư vấn luôn dựa trên dữ liệu đã được kiểm soát, còn AI chỉ đóng vai trò hỗ trợ diễn đạt, không tự ý “sáng tạo” thông tin.

3.6.3.3 Quy trình nhận diện bệnh dựa trên triệu chứng

Khi người dùng gửi câu hỏi hoặc mô tả triệu chứng, hệ thống không gửi trực tiếp dữ liệu này cho AI mà thực hiện xử lý nội bộ trước. Nội dung đầu vào được chuẩn hóa bằng cách loại bỏ dấu tiếng Việt và phân tách từ khóa.

Tiếp theo, hệ thống tiến hành so khớp thông tin người dùng cung cấp với dữ liệu trong kho tri thức. Việc so khớp được thực hiện thông qua hai cơ chế chính:

- So khớp tên bệnh (nếu người dùng nhắc trực tiếp tên bệnh).
- So khớp các từ khóa trong mô tả triệu chứng với danh sách triệu chứng của từng bệnh.

Mỗi lần trùng khớp sẽ được chấm điểm, bệnh có tổng điểm cao nhất sẽ được xem là kết quả phù hợp nhất. Cách tiếp cận này giúp chatbot đưa ra kết luận dựa trên dữ liệu thực tế, thay vì suy đoán.

3.6.3.4 Tích hợp Google Gemini trong giai đoạn sinh câu trả lời

Sau khi hoàn tất các bước truy xuất và xử lý dữ liệu, hệ thống mới gửi thông tin đã được tổng hợp cho Google Gemini API. Gemini không tham gia vào quá trình nhận diện bệnh hay lựa chọn thuốc mà chỉ thực hiện nhiệm vụ:

- Viết lại nội dung tư vấn bằng ngôn ngữ tự nhiên.
- Trình bày thông tin rõ ràng, dễ hiểu, phù hợp với người dùng nông nghiệp.
- Giữ giọng điệu thân thiện, gần gũi.

Nhờ đó, câu trả lời cuối cùng vừa đảm bảo chính xác về mặt chuyên môn, vừa có tính giao tiếp cao.

3.6.3.5 Truy xuất thuốc và đề xuất phương án xử lý

Sau khi xác định được bệnh, hệ thống tiếp tục truy vấn danh sách thuốc bảo vệ thực vật phù hợp. Việc đề xuất thuốc được ưu tiên dựa trên:

- Danh sách sản phẩm đã được liên kết sẵn với bệnh trong kho tri thức.
- Trường hợp chưa có liên kết trực tiếp, hệ thống sẽ tìm các sản phẩm có hoạt chất trùng khớp với hoạt chất điều trị của bệnh.

Cơ chế này đảm bảo các sản phẩm được đề xuất có cơ sở chuyên môn rõ ràng, tránh việc gợi ý thuốc không phù hợp hoặc không liên quan.

CHƯƠNG 4: KẾT QUẢ NGHIÊN CỨU

4.1 Tổng quan kết quả đạt được

Sau quá trình phân tích yêu cầu, thiết kế kiến trúc và triển khai hệ thống, đề tài “Xây dựng Hệ thống tư vấn và bán thuốc bảo vệ thực vật thông minh” đã được hoàn thành với đầy đủ các chức năng theo mục tiêu đề ra. Hệ thống được xây dựng dựa trên kiến trúc MERN Stack (MongoDB, Express.js, React.js, Node.js), kết hợp với mô hình Chatbot RAG tích hợp Google Gemini, nhằm hỗ trợ tư vấn bệnh cây trồng và cung cấp giải pháp điều trị phù hợp cho người dùng.

Kết quả thực hiện cho thấy hệ thống đã triển khai thành công các nhóm chức năng chính gồm: quản lý người dùng, quản lý sản phẩm thuốc bảo vệ thực vật, quản lý đơn hàng, tư vấn bệnh cây trồng bằng chatbot thông minh và quản trị hệ thống. Các chức năng này được tích hợp đồng bộ, giúp người dùng vừa có thể mua thuốc trực tuyến vừa nhận được tư vấn kỹ thuật chính xác, kịp thời.

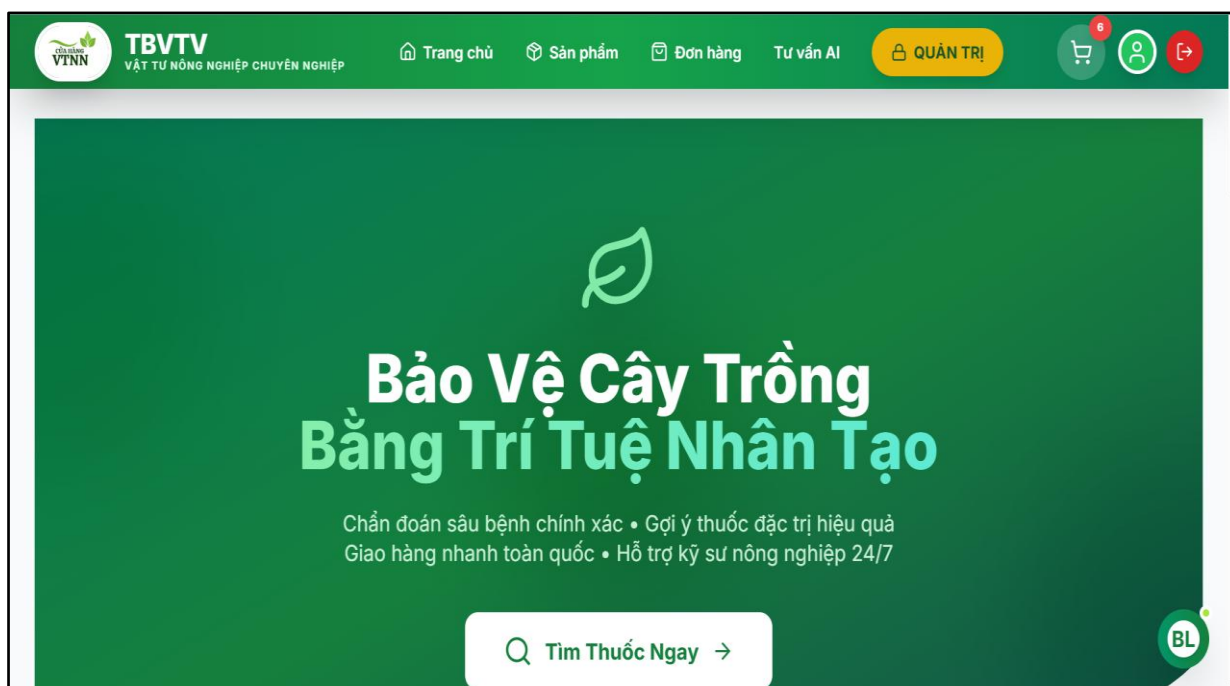
4.2 Kết quả triển khai các chức năng

4.2.1 Kết quả triển khai chức năng bán hàng

4.2.1.1 Trang chủ hệ thống

Trang chủ là điểm truy cập đầu tiên của người dùng khi vào hệ thống, đóng vai trò giới thiệu tổng quan và định hướng sử dụng các chức năng chính. Giao diện trang chủ được thiết kế đơn giản, dễ nhìn, phù hợp với đối tượng người dùng là nông dân và các hộ kinh doanh vật tư nông nghiệp.

Tại trang chủ, hệ thống hiển thị các khu vực nổi bật như danh sách sản phẩm thuốc bảo vệ thực vật mới nhất, các nhóm thuốc phổ biến theo cây trồng hoặc công dụng, cùng nút truy cập nhanh đến chức năng tư vấn bệnh cây trồng. Cách bố trí này giúp người dùng nhanh chóng tiếp cận được nội dung mình quan tâm mà không cần nhiều thao tác phức tạp.



Hình 3.1 Trang chủ hệ thống

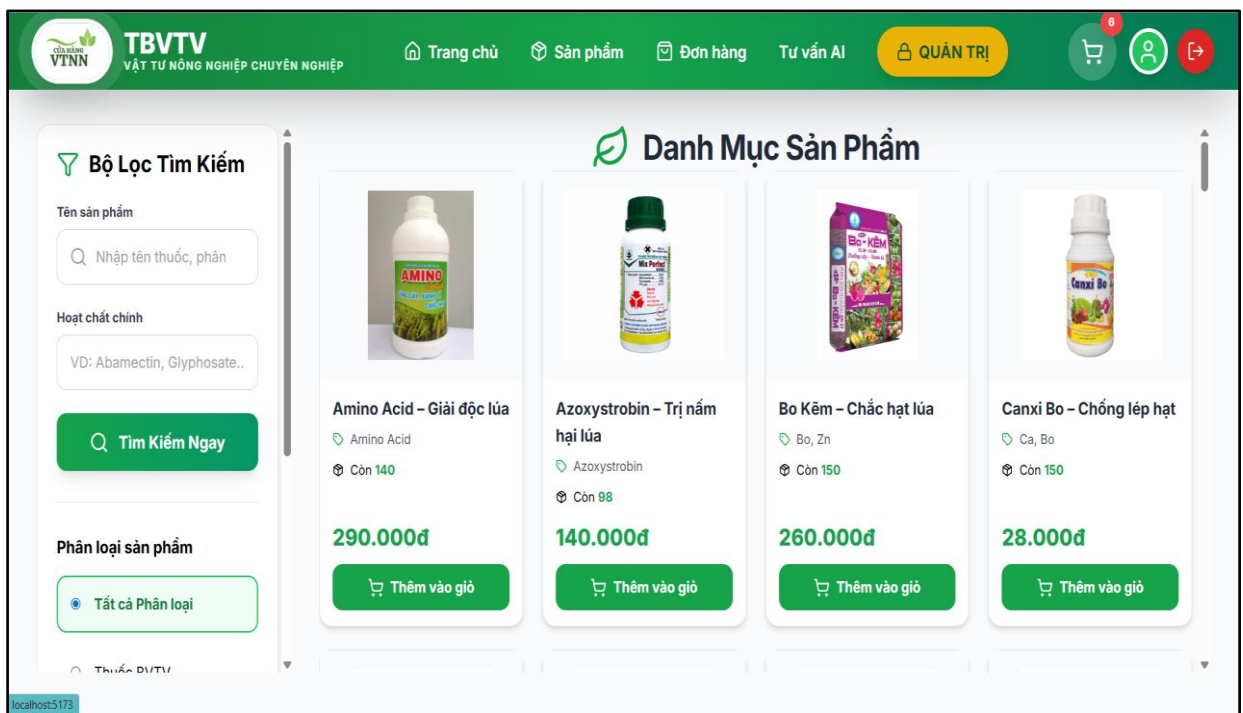
Bố cục trang chủ được xây dựng theo hướng đơn giản, dễ quan sát, phù hợp với người dùng không rành về công nghệ. Nhờ đó, người dùng có thể nhanh chóng nắm bắt được mục đích của hệ thống và cách sử dụng các chức năng chính.

4.2.1.2 Trang danh sách sản phẩm

Trang danh sách sản phẩm được xây dựng nhằm hỗ trợ người dùng duyệt, tìm kiếm và lựa chọn thuốc bảo vệ thực vật một cách thuận tiện. Tại đây, các sản phẩm được hiển thị rõ ràng với các thông tin cơ bản như tên thuốc, giá bán, hình ảnh minh họa và công dụng chính.

Hệ thống hỗ trợ chức năng lọc sản phẩm theo danh mục và theo hoạt chất, giúp người dùng nhanh chóng thu hẹp phạm vi tìm kiếm. Điều này đặc biệt hữu ích khi người dùng đã được chatbot tư vấn về hoạt chất phù hợp và muốn tìm sản phẩm tương ứng để mua ngay.

Trang danh sách sản phẩm giúp người dùng tiết kiệm thời gian tìm kiếm, giảm nhầm lẫn khi lựa chọn thuốc, đồng thời nâng cao hiệu quả mua sắm trong thực tế sản xuất nông nghiệp.



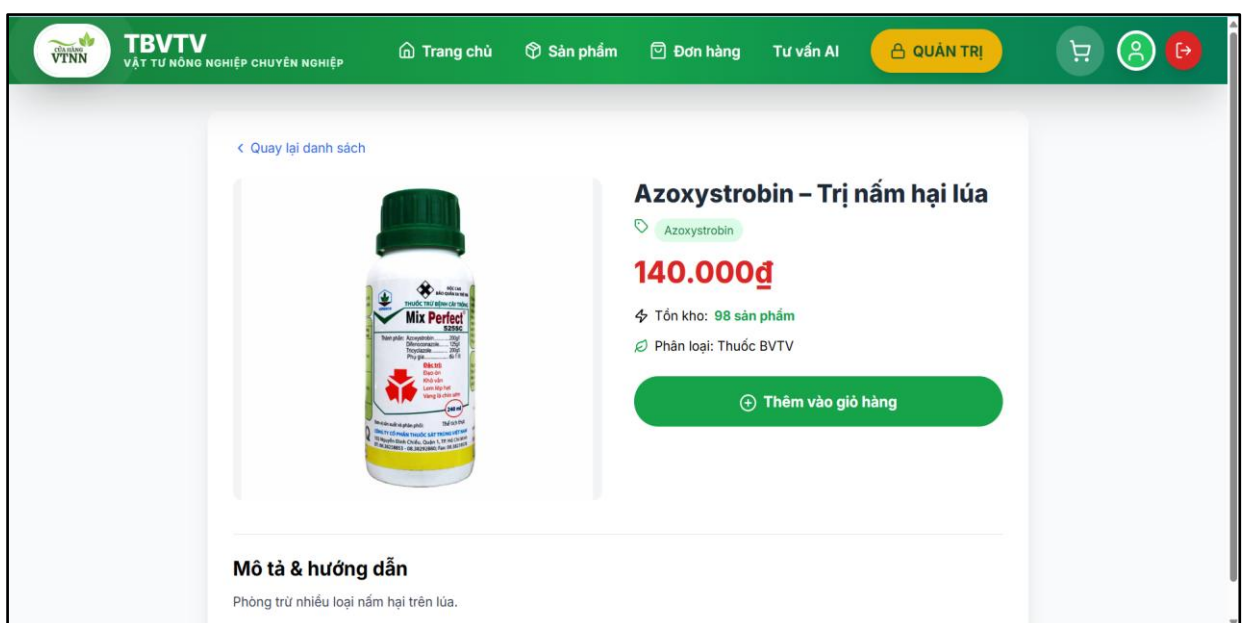
Hình 3.2 Trang danh sách sản phẩm

4.2.1.3 Trang chi tiết sản phẩm

Trang chi tiết sản phẩm cung cấp thông tin đầy đủ cho từng loại thuốc, bao gồm tên thuốc, giá bán, hoạt chất, công dụng và mô tả chi tiết.

Trang này giúp người dùng hiểu rõ hơn về đặc tính và cách sử dụng thuốc trước khi quyết định mua, góp phần hạn chế việc sử dụng thuốc sai mục đích hoặc sai kỹ thuật.

Ngoài ra, việc hiển thị rõ ràng tình trạng tồn kho cũng giúp người dùng chủ động hơn trong quá trình đặt hàng.



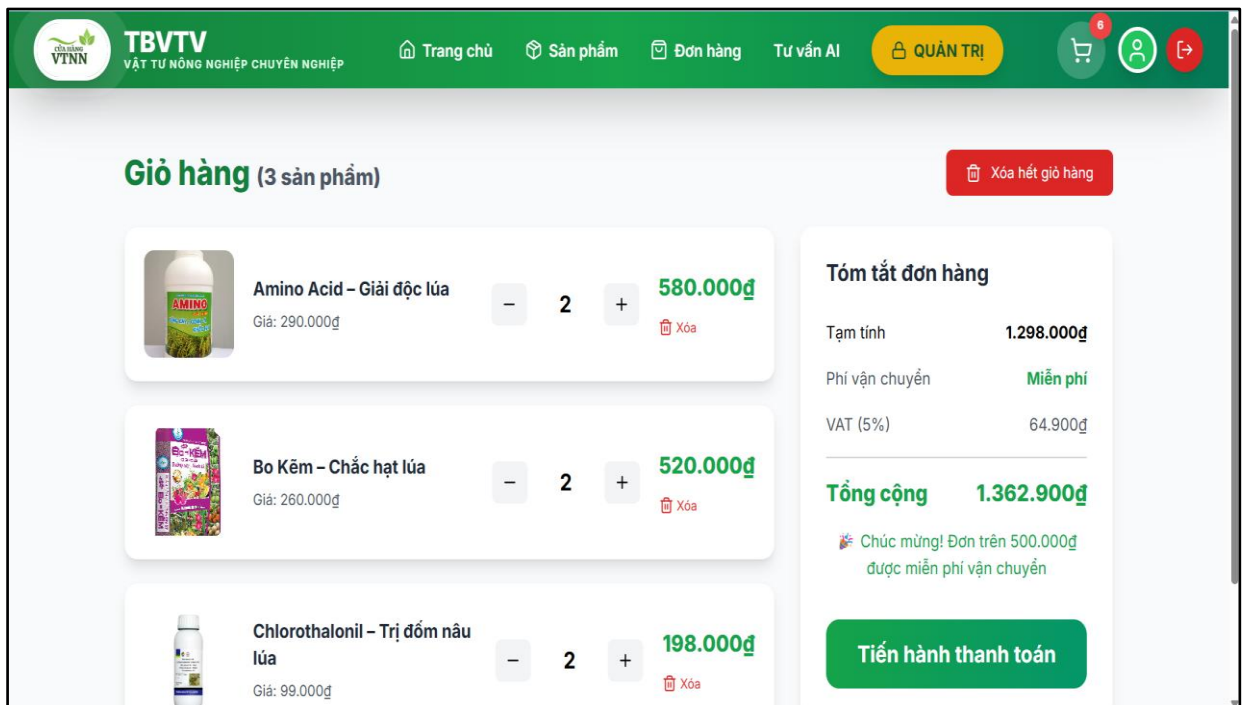
Hình 3.3 Trang chi tiết sản phẩm

4.2.1.4 Trang giỏ hàng

Trang giỏ hàng cho phép người dùng quản lý các sản phẩm đã chọn trước khi tiến hành đặt hàng. Tại đây, người dùng có thể xem lại danh sách thuốc đã thêm vào giỏ, điều chỉnh số lượng hoặc xóa sản phẩm không còn nhu cầu sử dụng.

Việc hiển thị rõ ràng giá từng sản phẩm và tổng giá trị đơn hàng giúp người dùng chủ động kiểm soát chi phí, tránh phát sinh sai sót khi đặt mua. Giao diện được thiết kế đơn giản, dễ thao tác, phù hợp với cả những người dùng ít tiếp xúc với công nghệ.

Trang giỏ hàng đóng vai trò là bước trung gian quan trọng, giúp người dùng kiểm tra lại toàn bộ lựa chọn trước khi xác nhận đơn hàng, góp phần nâng cao độ chính xác và trải nghiệm mua sắm.



Hình 3.4 Trang giỏ hàng

4.2.1.5 Trang xác nhận đặt hàng

Trang xác nhận đặt hàng là bước cuối cùng trong quy trình mua thuốc trực tuyến. Tại đây, người dùng nhập hoặc kiểm tra thông tin giao hàng, bao gồm địa chỉ nhận hàng và thông tin liên hệ.

Hệ thống hiển thị lại toàn bộ thông tin đơn hàng trước khi người dùng xác nhận, giúp hạn chế tối đa các lỗi sai trong quá trình đặt mua. Sau khi hoàn tất, đơn hàng được lưu trữ trên hệ thống và sẵn sàng cho các bước xử lý tiếp theo.

Trang xác nhận đặt hàng giúp quá trình mua bán diễn ra minh bạch, rõ ràng và

phù hợp với nhu cầu sử dụng thực tế của người dùng.

Hình 3.5 Xác nhận đặt hàng

4.2.1.6 Trang lịch sử mua hàng

Trang lịch sử mua hàng được xây dựng nhằm hỗ trợ người dùng theo dõi toàn bộ các giao dịch đã phát sinh trên hệ thống một cách minh bạch và thuận tiện. Giúp người dùng kiểm soát quá trình mua sắm và sử dụng thuốc trong thời gian dài.

Tại trang này, người dùng có thể xem danh sách các đơn hàng đã đặt theo thứ tự thời gian, bao gồm các thông tin cơ bản như: mã đơn hàng, ngày đặt, tổng giá trị đơn hàng và trạng thái xử lý. Việc hiển thị trạng thái đơn hàng giúp người dùng chủ động theo dõi tiến trình giao dịch, tránh nhầm lẫn hoặc thất lạc thông tin.

Mã đơn	Ngày đặt	Tổng tiền	Trạng thái	Chi tiết	Hành động
15F63E5B	20:54 Th 7, 27/12/2025	458.400 đ	Đã giao hàng	Xem	—
E938065B	18:51 Th 7, 27/12/2025	1.643.850 đ	Đã giao hàng	Xem	—

Hình 3.6 Trang lịch sử mua hàng

4.2.1.7 Trang hồ sơ cá nhân

Trang hồ sơ cá nhân được xây dựng nhằm giúp người dùng quản lý thông tin tài khoản và bảo mật cá nhân một cách chủ động. Đây là khu vực trung tâm để người dùng cập nhật thông tin liên quan đến quá trình sử dụng hệ thống, đảm bảo tính chính xác và an toàn trong các giao dịch mua bán.

Tại trang hồ sơ cá nhân, người dùng có thể xem và chỉnh sửa các thông tin cơ bản như họ tên, email, số điện thoại, địa chỉ và ngày sinh. Việc cho phép cập nhật thông tin giúp hệ thống luôn lưu trữ dữ liệu đúng với thực tế, đặc biệt quan trọng trong các chức năng liên quan đến giao hàng và chăm sóc khách hàng.

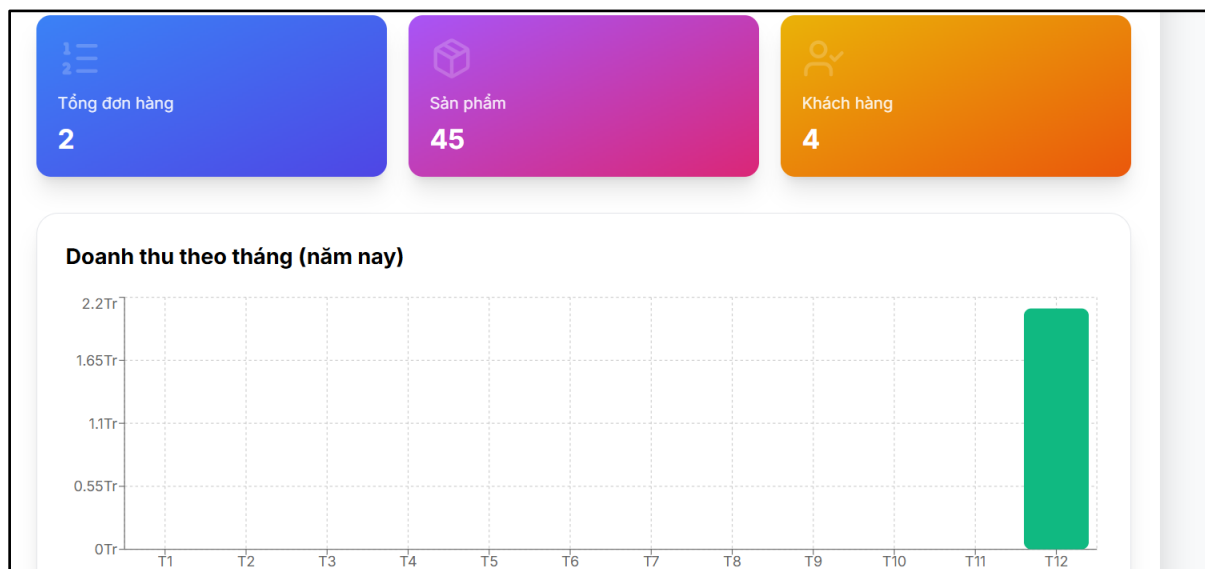
Hình 3.7 Trang hồ sơ cá nhân

4.2.2 Kết quả triển khai chức năng quản trị

4.2.2.1 Trang thống kê và tổng quan hệ thống

Trang thống kê đóng vai trò là bảng điều khiển trung tâm (Dashboard) của quản trị viên. Tại đây, hệ thống hiển thị các thông tin tổng quan như số lượng người dùng, số sản phẩm, tổng số đơn hàng:

- Thống kê số tổng đơn hàng đã hoàn thành, tổng sản phẩm trong cửa hàng, số khách hàng và doanh thu theo tháng trong năm.



Hình 3.8 Thống kê Tổng quan về cửa hàng và doanh thu

- Thống kê các sản phẩm bán chạy, có thể sắp xếp theo số lượng bán và doanh thu.

		Theo số lượng Theo số lượng Theo doanh thu	Top 10 Lấy
	Chlorothalonil – Phòng cháy lá lúa Mã: 694fe503		4 sp 408.000đ
	Amino Acid – Giải độc lúa Mã: 6951c526		2 sp 580.000đ
	Bo Kẽm – Chắc hạt lúa Mã: 6951c526		2 sp 520.000đ
	Chlorothalonil – Trị đốm nâu lúa Mã: 6951c526		2 sp 198.000đ
	Cám cá tra tăng trưởng Mã: 6951f506		1 sp 350.000đ

Hình 3.9 Thống kê Sản phẩm bán chạy

- Thống kê các khách hàng thân thiết (có thể lọc theo số tiền mua tối thiểu).

		0 Lọc
1	lộc huynhhuuloc080204@gmail.com	3.363.900đ 3 đơn hàng

Hình 3.10 Thống kê khách hàng thân thiết

Các số liệu được trình bày trực quan giúp admin nhanh chóng nắm bắt tình hình hoạt động của hệ thống mà không cần truy cập sâu vào từng phân hệ. Trang này hỗ trợ việc giám sát và ra quyết định kịp thời trong quá trình vận hành.

4.2.2.2 Trang quản lý sản phẩm

Mỗi sản phẩm được quản lý với đầy đủ thông tin như tên thuốc, giá bán, hoạt chất, công dụng, hình ảnh minh họa và tồn kho. Trang quản lý sản phẩm cho phép admin thực hiện các chức năng quản lý sản phẩm như:

- Thêm sản phẩm mới.

Quản lý sản phẩm

Phân loại: Thuốc BVTV

Tên sản phẩm:

Hoạt chất:

☒ Hình ảnh: Chọn tệp Không có tệp nào được chọn

Bệnh đặc trị: đạo ôn, thán thư, cháy lá


Giá: Tồn kho:

Mô tả chi tiết:

+ Thêm sản phẩm

Hình 3.11 Form thêm sản phẩm mới

- Sửa thông tin sản phẩm.


Chỉnh sửa sản phẩm

Phân loại

Phân Bón

Tên sản phẩm

Amino Acid – Giải độc lúa

Hoạt chất

☒ **Hình ảnh**

Chọn tệp


Không có tệp nào được chọn

Bệnh đặc trị

giải độc

290000

137



Giải độc thuốc, giúp lúa phục hồi nhanh.

















+

Lưu thay đổi

Hủy

Hình 3.12 Sửa thông tin sản phẩm

- Quản lý danh sách sản phẩm hiện có.

Danh sách sản phẩm (45)					
Tên	Loại	Bệnh trị	Giá	Sửa	Xóa
Amino Acid – Giải độc lúa	Phân Bón	giải độc	290.000đ		
Azoxystrobin – Trị nấm hại lúa	Thuốc BVTV	bệnh đạo ôn, bệnh cháy lá	140.000đ		
Bo Kẽm – Chắc hạt lúa	Phân Bón	chắc hạt	260.000đ		
Canxi Bo – Chống lép hạt	Phân Bón	chống lép hạt	28.000đ		
Carbendazim – Trị thối thân lúa	Thuốc BVTV	bệnh thối thân	92.000đ		
Chlorothalonil – Phòng cháy lá lúa	Thuốc BVTV	bệnh cháy lá	102.000đ		
Chlorothalonil – Trị đốm nâu lúa	Thuốc BVTV	bệnh đốm nâu	99.000đ		
Copper Hydroxide – Trị cháy bìa lá	Thuốc BVTV	bệnh cháy bìa lá	95.000đ		

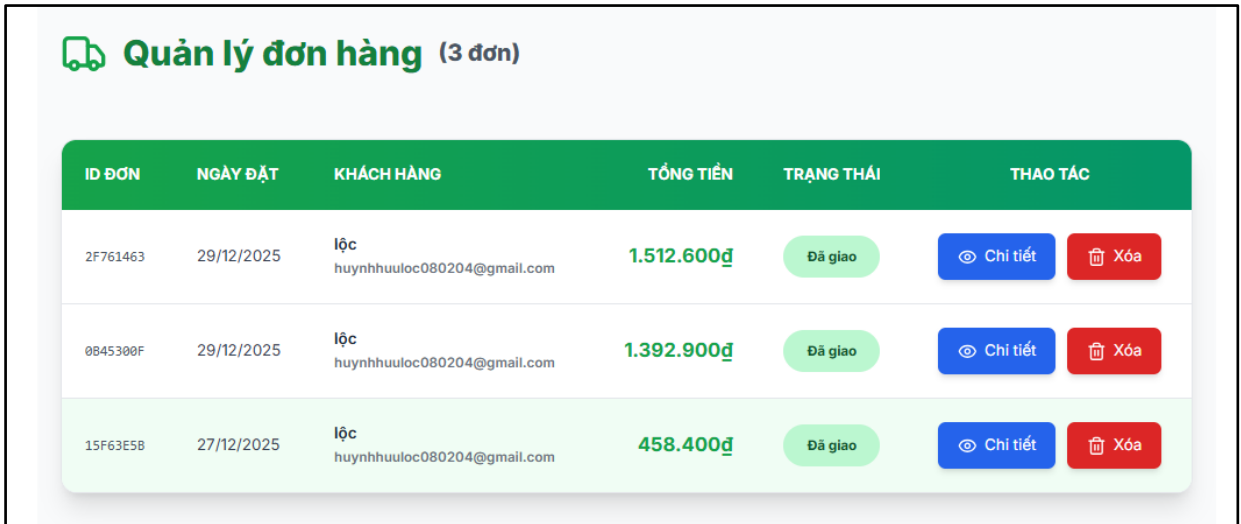
Hình 3.13 Quản lý danh sách sản phẩm

Việc quản lý sản phẩm tập trung giúp admin dễ dàng cập nhật danh mục thuốc theo thực tế thị trường và đảm bảo thông tin hiển thị cho người dùng luôn chính xác. Đây là phân hệ quan trọng trong chức năng bán hàng của hệ thống

4.2.2.3 Trang quản lý đơn hàng

Trang quản lý đơn hàng cho phép quản trị viên theo dõi toàn bộ quá trình mua hàng của người dùng. Admin có thể xem danh sách đơn hàng, kiểm tra thông tin người mua, danh sách sản phẩm trong đơn và trạng thái xử lý.

Chức năng này giúp admin kiểm soát tốt quy trình bán hàng, hỗ trợ xử lý đơn kịp thời và đảm bảo quyền lợi cho cả người mua và hệ thống. Ngoài ra, việc lưu trữ lịch sử đơn hàng còn giúp phục vụ công tác thống kê và đánh giá hoạt động kinh doanh.




ID ĐƠN	NGÀY ĐẶT	KHÁCH HÀNG	TỔNG TIỀN	TRẠNG THÁI	THAO TÁC
2F761463	29/12/2025	lộc huynhhuuloc080204@gmail.com	1.512.600đ	Đã giao	Chi tiết Xóa
0845308F	29/12/2025	lộc huynhhuuloc080204@gmail.com	1.392.900đ	Đã giao	Chi tiết Xóa
15F63E5B	27/12/2025	lộc huynhhuuloc080204@gmail.com	458.400đ	Đã giao	Chi tiết Xóa

Hình 3.14 Quản lý đơn hàng

4.2.2.4 Trang quản lý người dùng

Trang quản lý người dùng cho phép admin theo dõi danh sách tài khoản đã đăng ký trên hệ thống. Tại đây, admin có thể xem thông tin cơ bản của người dùng, phân loại theo vai trò (người dùng thường, quản trị viên) và thực hiện các thao tác quản lý.

Chức năng này giúp đảm bảo an toàn hệ thống, ngăn chặn các hành vi sử dụng sai mục đích và duy trì môi trường hoạt động ổn định cho toàn bộ người dùng.

 Quản lý người dùng (5 tài khoản)				
ID	Họ và tên	Email	Vai trò	Ngày đăng ký
fd82b2ef	lộc	huynhhuuloc080204@gmail.com	QUẢN TRỊ VIÊN	11/10/2025
99deaa46	Trà Mi	tmi@gmail.com	KHÁCH HÀNG	31/10/2025
73cab7ba	Quốc Đạm	qdam@gmail.com	KHÁCH HÀNG	1/12/2025
c61cffe9	linhthach	linh@gmail.com	KHÁCH HÀNG	23/12/2025
b4fb3ec2	Hân	hhan@gmail.com	KHÁCH HÀNG	23/12/2025

Hình 3.15 Quản lý người dùng

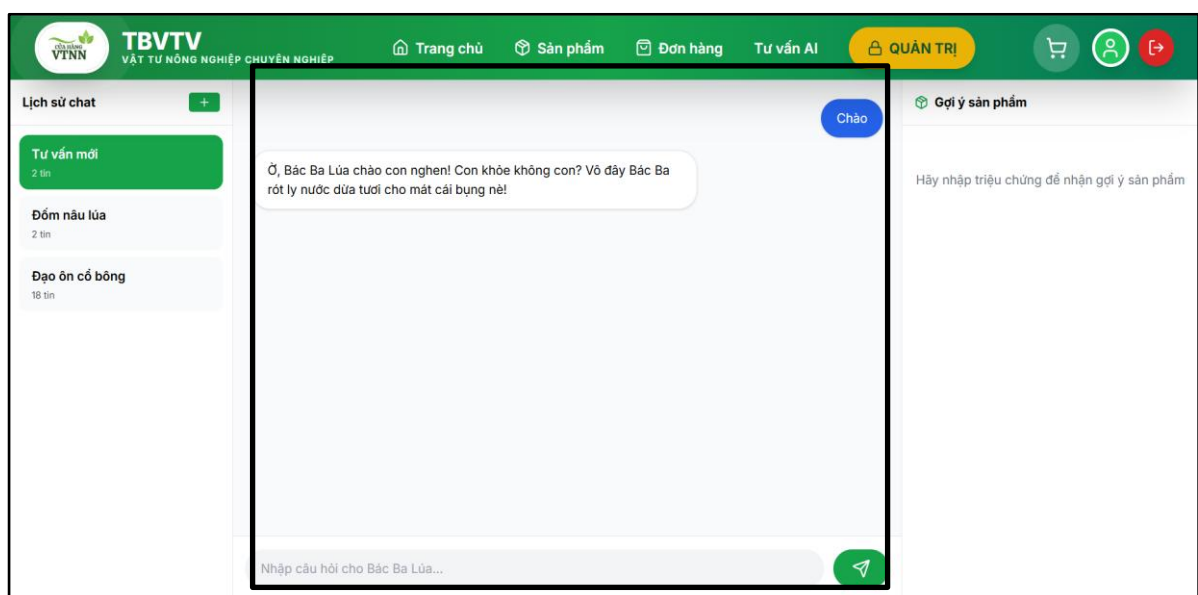
4.2.3 Chức năng tư vấn bệnh cây trồng bằng Chatbot thông minh

4.2.3.1 Trang tư vấn bệnh cây trồng chuyên sâu

Trang tư vấn bệnh cây trồng là một trang chức năng độc lập, được thiết kế dành riêng cho nhu cầu tư vấn kỹ thuật nông nghiệp. Đây là nơi người dùng có thể tương tác đầy đủ với chatbot để nhận được kết quả chẩn đoán rõ ràng và chi tiết.

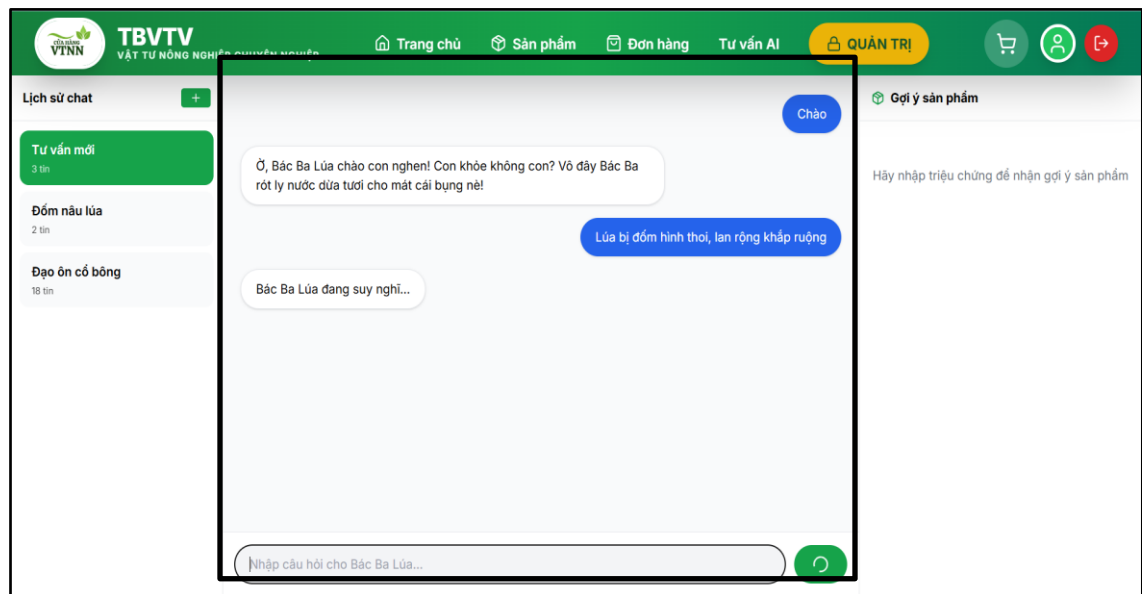
Tại trang này, người dùng có thể:

- Theo dõi toàn bộ nội dung hội thoại giữa người dùng và chatbot



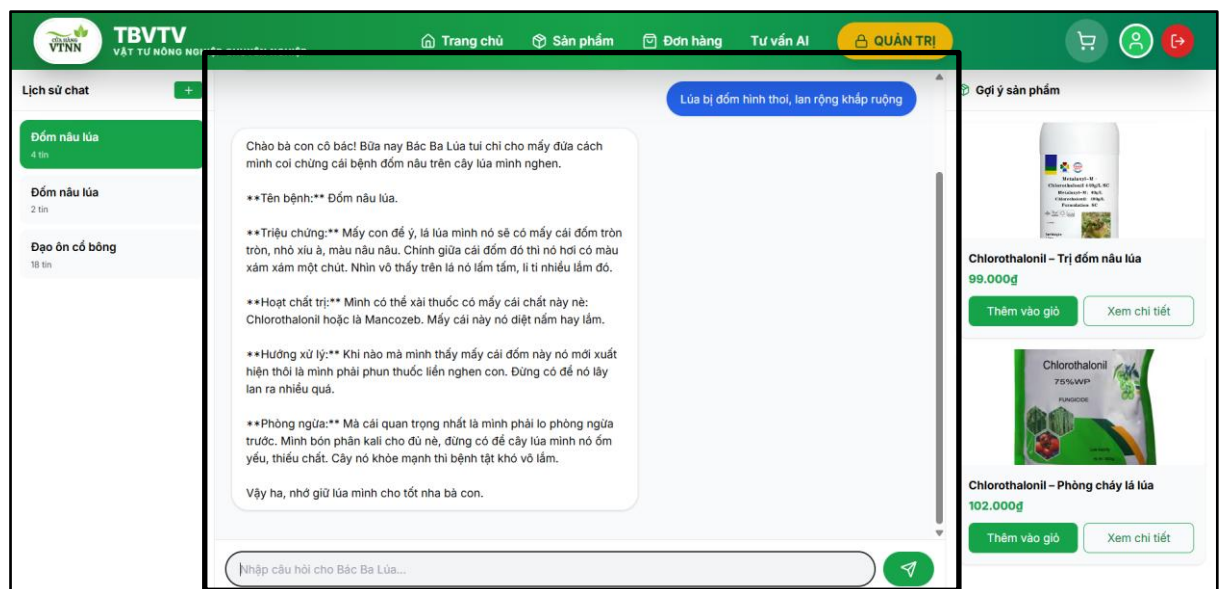
Hình 3.16 Khung chat tư vấn

- Nhập mô tả chi tiết về triệu chứng bệnh cây trồng (ví dụ: màu sắc lá, vị trí bị hại, tốc độ lây lan)



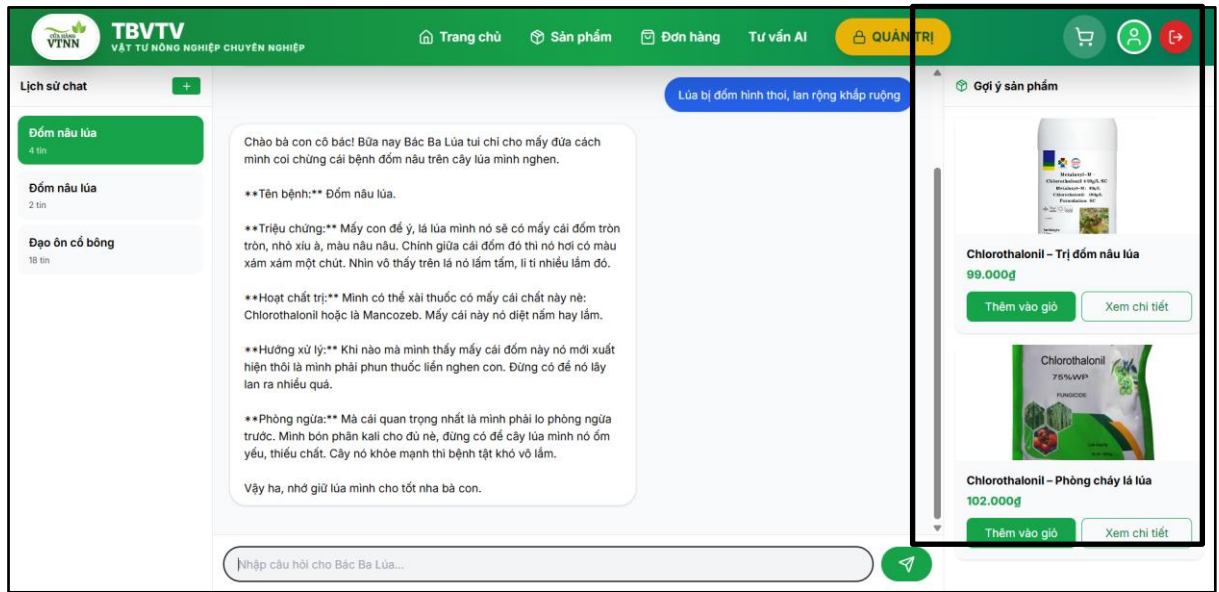
Hình 3.17 Nhập triệu chứng và chờ kết quả tư vấn

- Nhận kết luận về tên bệnh cây trồng dựa trên dữ liệu trong kho tri thức
- Xem hướng xử lý, biện pháp phòng ngừa và các lưu ý khi sử dụng thuốc



Hình 3.18 Kết quả tư vấn từ hệ thống

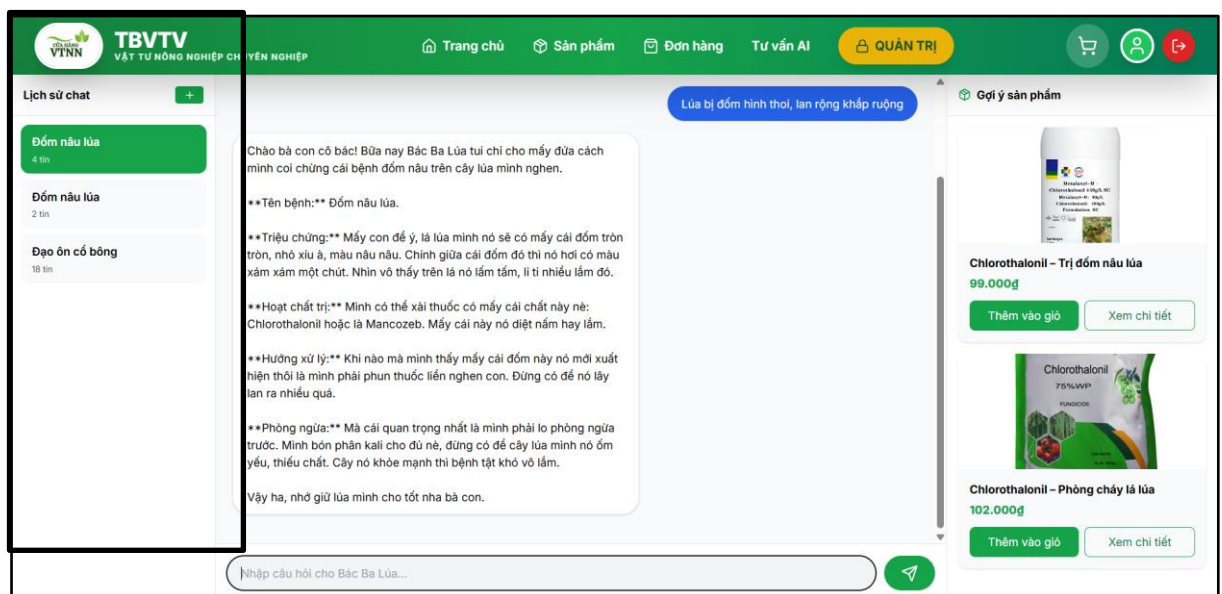
- Nhận danh sách sản phẩm thuốc bảo vệ thực vật được gợi ý phù hợp với bệnh đã xác định



Hình 3.19 Hệ thống gợi ý thuốc điều trị có trong cửa hàng

Trang tư vấn được thiết kế với bố cục rõ ràng, tập trung vào nội dung tư vấn, giúp người dùng dễ dàng đọc hiểu và áp dụng trong thực tế. Việc tách riêng trang tư vấn giúp người dùng có không gian làm việc chuyên sâu, đặc biệt phù hợp trong các trường hợp cần tư vấn kỹ thuật trước khi quyết định mua thuốc.

Ngoài ra, lịch sử tư vấn của người dùng cũng được lưu trữ, cho phép người dùng xem lại các lần tư vấn trước đó, đồng thời giúp hệ thống phân tích và cải thiện chất lượng tư vấn trong tương lai.



Hình 3.20 Lịch sử tư vấn

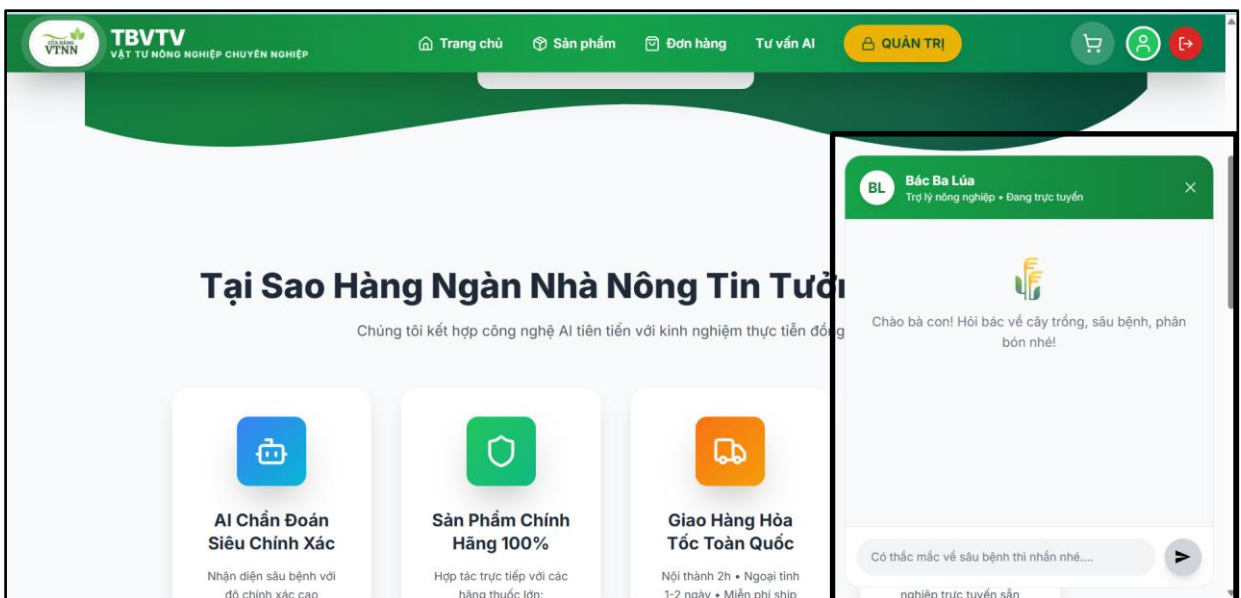
4.2.3.2 Component tư vấn nhanh (Chatbot Component)

Bên cạnh trang tư vấn chuyên sâu, hệ thống còn cung cấp component tư vấn nhanh, được tích hợp trực tiếp trên các trang khác của website như trang chủ, trang sản phẩm hoặc trang giỏ hàng. Component này cho phép người dùng tiếp cận chức năng tư vấn mà không cần rời khỏi trang đang sử dụng.

Component tư vấn nhanh hỗ trợ người dùng:

- Đặt các câu hỏi ngắn về triệu chứng hoặc công dụng thuốc
- Nhận phản hồi nhanh từ chatbot trong thời gian thực
- Xác nhận sơ bộ tình trạng bệnh hoặc hướng xử lý ban đầu
- Hình thức tư vấn này đặc biệt phù hợp khi người dùng:
 - Đang duyệt sản phẩm và cần hỏi nhanh về công dụng
 - Muốn kiểm tra nhanh triệu chứng trước khi vào trang tư vấn chi tiết
 - Không có nhu cầu tư vấn chuyên sâu nhưng vẫn cần thông tin hỗ trợ

Về mặt kỹ thuật, component tư vấn nhanh sử dụng chung hệ thống xử lý chatbot ở backend với trang tư vấn chuyên sâu, đảm bảo nội dung tư vấn nhất quán và chính xác. Tuy nhiên, giao diện và phạm vi hiển thị được tối giản để không làm ảnh hưởng đến trải nghiệm mua sắm của người dùng.



Hình 3.21 Giao diện Chat Component]

4.2.4 Kho tri thức và khả năng mở rộng

Kho tri thức (Knowledge Base) được xây dựng độc lập, cho phép quản trị viên cập nhật, bổ sung hoặc chỉnh sửa thông tin bệnh cây trồng mà không ảnh hưởng đến logic xử lý của chatbot. Đây là ưu điểm lớn giúp hệ thống dễ dàng mở rộng trong tương lai khi có thêm các loại cây trồng hoặc bệnh mới.

<pre> _id: ObjectId('6951c94edceccb3d5089d5f5') diseaseName: "Đạo ôn lá lúa" description: "Bệnh do nấm gây hại phổ biến trên lúa, phát triển mạnh khi thời tiết ẩ..." ▼ symptoms: Array (5) 0: "vết bệnh hình thoi" 1: "tâm vết bệnh màu xám trắng" 2: "viền vết bệnh màu nâu đậm" 3: "vết bệnh lan dọc phiến lá" 4: "lá khô từng mảng" ▼ activeIngredients: Array (2) 0: "Tricyclazole" 1: "Isoprothiolane" treatmentGuide: "Phun thuốc đặc trị đạo ôn khi bệnh mới xuất hiện, giữ ruộng thông thoáng..." prevention: "Bón phân cân đối, không bón thừa đạm, gieo sạ mật độ hợp lý." ► recommendedProducts: Array (empty) </pre>
<pre> _id: ObjectId('6951c94edceccb3d5089d5f6') diseaseName: "Đạo ôn cổ bông" description: "Bệnh gây hại trên cổ bông lúa, làm bông lúa bị gãy hoặc lép hạt." ▼ symptoms: Array (5) 0: "cổ bông thắt lại" 1: "cổ bông chuyển màu nâu đen" 2: "bông lúa bị gãy rũ" 3: "hạt lúa không vào gạo" 4: "bông lúa bạc trắng" ▼ activeIngredients: Array (2) 0: "Isoprothiolane" 1: "Tricyclazole" </pre>

Hình 3.22 Dữ liệu kho tri thức

CHƯƠNG 5: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

5.1 Kết luận

5.1.1 Kết quả đạt được

Sau quá trình nghiên cứu, phân tích yêu cầu, thiết kế và triển khai, đồ án “Xây dựng Hệ thống tư vấn và bán thuốc bảo vệ thực vật thông minh” đã được hoàn thành với các mục tiêu đề ra. Hệ thống được xây dựng trên nền tảng MERN Stack (MongoDB, Express.js, React.js, Node.js), tích hợp Chatbot tư vấn thông minh theo mô hình RAG kết hợp Google Gemini, đáp ứng đồng thời nhu cầu kinh doanh thuốc bảo vệ thực vật và tư vấn kỹ thuật nông nghiệp cho người dùng.

Về mặt chức năng, hệ thống đã triển khai đầy đủ các nghiệp vụ chính gồm:

- Quản lý người dùng và xác thực đăng nhập.
- Quản lý danh mục và sản phẩm thuốc bảo vệ thực vật.
- Hỗ trợ mua hàng trực tuyến với giỏ hàng, đặt hàng và theo dõi lịch sử mua hàng.
- Hệ thống quản trị dành cho quản trị viên để quản lý sản phẩm, đơn hàng và người dùng.
- Chatbot tư vấn bệnh cây trồng, hỗ trợ chẩn đoán và đề xuất hướng xử lý.

Về mặt kỹ thuật, Backend được tổ chức theo mô hình API RESTful, đảm bảo phân tách rõ ràng giữa các lớp xử lý nghiệp vụ và dữ liệu. Cơ sở dữ liệu MongoDB được thiết kế theo hướng NoSQL với các collection như User, Product, Order, KnowledgeBase, ChatSession, ChatHistory, phù hợp với đặc thù dữ liệu nông nghiệp và hội thoại. Frontend sử dụng React kết hợp TailwindCSS giúp giao diện thân thiện, dễ sử dụng, phù hợp với đối tượng người dùng là nông dân và hộ kinh doanh vật tư nông nghiệp.

Đặc biệt, chatbot tư vấn bệnh cây trồng hoạt động ổn định, có khả năng tiếp nhận mô tả triệu chứng từ người dùng, truy xuất thông tin từ kho tri thức và tổng hợp câu trả lời rõ ràng, dễ hiểu, góp phần hỗ trợ người dùng ra quyết định nhanh chóng và chính xác hơn.

5.1.2 Đóng góp mới

Đồ án đã mang lại một số đóng góp đáng kể cả về mặt học thuật và thực tiễn.

Thứ nhất, đề tài đã xây dựng thành công một hệ thống kết hợp giữa thương mại điện tử và tư vấn kỹ thuật nông nghiệp, thay vì chỉ tập trung vào bán hàng đơn thuần. Cách tiếp cận này giúp người dùng không chỉ mua sản phẩm mà còn hiểu rõ nguyên nhân, triệu chứng bệnh và cách sử dụng thuốc đúng kỹ thuật.

Thứ hai, việc áp dụng mô hình Retrieval-Augmented Generation (RAG) trong chatbot tư vấn đã giúp khắc phục hạn chế của AI sinh văn bản thuần túy. Chatbot không tự suy đoán mà dựa trên dữ liệu tri thức đã được kiểm chứng trong kho Knowledge Base, từ đó nâng cao độ tin cậy của kết quả tư vấn.

Thứ ba, hệ thống kho tri thức được thiết kế độc lập, cho phép mở rộng linh hoạt. Quản trị viên có thể bổ sung bệnh mới, triệu chứng mới hoặc cập nhật hướng xử lý mà không cần chỉnh sửa logic chatbot, giúp hệ thống có khả năng phát triển lâu dài.

Ngoài ra, đồ án còn là minh chứng cho việc vận dụng hiệu quả các kiến thức đã học như lập trình web, thiết kế cơ sở dữ liệu, xây dựng API, xác thực người dùng và tích hợp dịch vụ AI vào một bài toán thực tế trong lĩnh vực nông nghiệp.

5.1.3 Hạn chế

Bên cạnh các kết quả đạt được, hệ thống vẫn còn một số hạn chế nhất định. Kho tri thức bệnh cây trồng hiện tại mới tập trung vào một số nhóm cây trồng và bệnh phổ biến, chưa bao phủ đầy đủ các loại cây công nghiệp, cây ăn trái và cây đặc thù vùng miền. Chức năng tư vấn hiện mới dựa trên mô tả văn bản, chưa tích hợp hoàn chỉnh việc nhận diện bệnh trực tiếp từ hình ảnh.

Ngoài ra, hệ thống chưa hỗ trợ thanh toán trực tuyến mà mới dừng ở mức đặt hàng và quản lý đơn hàng. Một số tính năng nâng cao như thống kê hành vi người dùng, gợi ý thuốc theo lịch sử mua hàng hoặc cá nhân hóa tư vấn vẫn chưa được triển khai.

5.2 Hướng phát triển

Trong thời gian tới, hệ thống tư vấn và bán thuốc bảo vệ thực vật thông minh có thể tiếp tục được mở rộng và hoàn thiện theo nhiều hướng nhằm nâng cao giá trị sử dụng thực tế. Trước hết, kho tri thức bệnh cây trồng cần được bổ sung thêm dữ liệu cho nhiều nhóm cây khác nhau như cây ăn trái, cây công nghiệp và rau màu theo từng vùng sinh thái. Việc mở rộng này giúp chatbot có khả năng tư vấn chính xác hơn, phù hợp với điều kiện canh tác đa dạng của người dùng.

Bên cạnh đó, chức năng tư vấn hiện tại chủ yếu dựa trên mô tả triệu chứng bằng văn bản, trong tương lai có thể tích hợp thêm khả năng nhận diện bệnh từ hình ảnh cây trồng thông qua các mô hình AI thị giác của Google Gemini. Điều này sẽ giúp người dùng, đặc biệt là nông dân ít kinh nghiệm, có thể chẩn đoán bệnh nhanh chóng và trực quan hơn, giảm thiểu sai sót trong quá trình sử dụng thuốc.

Về mặt thương mại điện tử, hệ thống có thể được nâng cấp bằng cách tích hợp các

cổng thanh toán trực tuyến, quản lý vận chuyển và theo dõi trạng thái giao hàng. Ngoài ra, việc phân tích lịch sử mua hàng và lịch sử tư vấn chatbot có thể được khai thác để gợi ý sản phẩm phù hợp, cá nhân hóa trải nghiệm người dùng và nâng cao hiệu quả kinh doanh cho cửa hàng vật tư nông nghiệp.

Đối với phía quản trị, hệ thống có thể phát triển thêm các chức năng thống kê nâng cao như phân tích doanh thu theo thời gian, theo nhóm sản phẩm hoặc theo mùa vụ, từ đó hỗ trợ người quản lý đưa ra quyết định nhập hàng và phân phối hợp lý. Dữ liệu tư vấn bệnh cây trồng cũng có thể được sử dụng để dự báo xu hướng dịch hại, góp phần nâng cao vai trò của hệ thống trong hỗ trợ sản xuất nông nghiệp.

Cuối cùng, hệ thống có tiềm năng triển khai thực tế tại các cửa hàng vật tư nông nghiệp, hợp tác xã hoặc mô hình kinh doanh trực tuyến. Khi được áp dụng rộng rãi, hệ thống không chỉ đóng vai trò là một website bán hàng mà còn trở thành một công cụ hỗ trợ kỹ thuật số trong nông nghiệp, góp phần nâng cao hiệu quả sản xuất và sử dụng thuốc bảo vệ thực vật một cách an toàn, hợp lý.

DANH MỤC TÀI LIỆU THAM KHẢO

- [1] World Bank. (2023). *Vietnam – Climate Change and Green Growth DPF*. The World Bank Group.
<https://documents.worldbank.org/en/publication/documents-reports/documentdetail/099062023102015813>
- [2] React.js Team. (2025). *React Documentation: Learn React*. React.dev.
<https://react.dev/learn>
- [3] Node.js Foundation. (2025). *Node.js Documentation*. Node.js.
<https://nodejs.org/en/docs>
- [4] Tailwind Labs. (2025). *Tailwind CSS Documentation: Installation and Usage*. Tailwind CSS.
<https://tailwindcss.com/docs>
- [5] MongoDB, Inc. (2025). *MongoDB Manual: Introduction and Core Concepts*. MongoDB Documentation.
<https://www.mongodb.com/docs/manual/>
- [6] GeeksforGeeks. (2025). *MVC Design Pattern*. GeeksforGeeks.
<https://www.geeksforgeeks.org/system-design/mvc-design-pattern/>
- [7] Fielding, R. T. (2000). *Architectural Styles and the Design of Network-based Software Architectures* (Chapter 5: Representational State Transfer – REST). University of California, Irvine.
- Tài liệu cập nhật: REST API Tutorial. (2025). *What is REST?*
<https://restfulapi.net/>
- [8] Provos, N., & Mazières, D. (1999). *A Future-Adaptable Password Scheme*. Proceedings of the USENIX Annual Technical Conference.
- Tài liệu tham khảo bổ sung: Wikipedia. (2025). *Bcrypt*.
<https://en.wikipedia.org/wiki/Bcrypt>
- [9] Auth0. (2025). *JSON Web Token Introduction*. JWT.io.
<https://jwt.io/introduction>
- [10] U.S. Environmental Protection Agency (EPA). (2025). *Air Quality Index (AQI) Basics*. AirNow.gov.
<https://www.airnow.gov/aqi/aqi-basics>