

**TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT TP.HCM
KHOA CÔNG NGHỆ THÔNG TIN
BỘ MÔN CÔNG NGHỆ PHẦN MỀM**



**VÕ VĂN ĐỨC : 20110635
ÔN GIA PHÚ : 20110697**

Đề Tài :

**XÂY DỰNG HỆ THỐNG HỖ TRỢ NGƯỜI DÙNG
MUA BÁN CAFE**

KHÓA LUẬN TỐT NGHIỆP KỸ SƯ CNTT

**GIÁO VIÊN HƯỚNG DẪN
ThS. LÊ THỊ MINH CHÂU**

KHÓA 2020 – 2024

**TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT TP.HCM
KHOA CÔNG NGHỆ THÔNG TIN
BỘ MÔN CÔNG NGHỆ PHẦN MỀM**



**VÕ VĂN ĐỨC : 20110635
ÔN GIA PHÚ : 20110697**

Đề Tài :

**XÂY DỰNG HỆ THỐNG HỖ TRỢ NGƯỜI DÙNG
MUA BÁN CAFE**

KHÓA LUẬN TỐT NGHIỆP KỸ SƯ CNTT

**GIÁO VIÊN HƯỚNG DẪN
ThS. LÊ THỊ MINH CHÂU**

KHÓA 2020 - 2024

PHIẾU NHẬN XÉT CỦA GIÁO VIÊN HƯỚNG DẪN

Họ và tên Sinh viên 1 : Võ Văn Đức MSSV 1: 20110635

Ngành: Công nghệ Thông tin

Tên đề tài: Xây dựng hệ thống hỗ trợ người dùng mua bán cafe.

Họ và tên Giáo viên hướng dẫn: ThS. Lê Thị Minh Châu

NHẬN XÉT

- #### 1. Về nội dung đề tài & khối lượng thực hiện:

.....

- ## 2. Ưu điểm:

.....

.....

- ### 3. Khuyết điểm:

.....

4. Đề nghị cho bảo vệ hay không?

5. Đánh giá loại :

6. Điểm :

Tp. Hồ Chí Minh, ngày tháng năm

Giáo viên hướng dẫn

(Ký & ghi rõ họ tên)

PHIẾU NHẬN XÉT CỦA GIÁO VIÊN PHẢN BIỆN

Họ và tên Sinh viên 1 : Võ Văn Đức MSSV 1: 20110635

Ngành: Công nghệ Thông tin

Tên đề tài: Xây dựng hệ thống hỗ trợ người dùng mua bán cafe.

Họ và tên Giáo viên phản biện: TS. Lê Vĩnh Thịnh

NHẬN XÉT

- #### 1. Về nội dung đề tài & khối lượng thực hiện:

.....

- ## 2. Ưu điểm:

.....

- ### 3. Khuyết điểm:

.....

4. Đề nghị cho bảo vệ hay không?

5. Đánh giá loại :

6. Điểm :

Tp. Hồ Chí Minh, ngày tháng năm

Giáo viên phản biện

(Ký & ghi rõ họ tên)

LỜI CẢM ƠN

Lời đầu tiên nhóm em xin phép được gửi lời cảm ơn chân thành đến với Khoa Công Nghệ Thông Tin – Trường Đại học Sư phạm Kỹ thuật Thành phố Hồ Chí Minh đã tạo điều kiện cho nhóm em được học tập và phát triển nền tảng kiến thức để thực hiện đề tài này.

Và bên cạnh đó nhóm chúng em xin gửi đến cô Lê Thị Minh Châu lời cảm ơn thiết nhất. Trải qua một quá trình học tập và thực hiện đề tài trong thời gian qua, cô đã tận tâm chỉ bảo nhóm em trong suốt quá trình từ lúc bắt đầu đến lúc kết thúc đề tài này.

Dưới sự giảng dạy nhiệt tình và tận tâm của cô, chúng em đã học và hiểu những kiến thức cơ bản về Website thương mại điện tử cùng với những công nghệ được sử dụng trong các Website tương tự. Từ đó góp phần lớn vào xây dựng lượng kiến thức dồi dào để chúng em lấy đó làm hành trang kiến thức trong công việc sau này.

Tuy nhiên, lượng kiến thức là vô hạn và khả năng của chúng em có phần hạn chế, chúng em đã rất nỗ lực để thực hiện nó một cách tốt nhất có thể. Vì vậy, những sai sót là không thể tránh khỏi. Chúng em rất mong nhận được những ý kiến đóng góp nhiệt tình của quý thầy(cô) để có thể rút kinh nghiệm và trau dồi, hoàn thiện và nâng cấp sản phẩm của mình một cách tốt nhất có thể. Chúng em xin chân thành cảm ơn!

Trường ĐH Sư Phạm Kỹ Thuật TP.HCM
Khoa : CNTT

ĐỀ CƯƠNG LUẬN VĂN TỐT NGHIỆP

Họ và Tên SV thực hiện 1 : Võ Văn Đức

Mã Số SV : 20110635

Họ và Tên SV thực hiện 2 : Ôn Gia Phú

Mã Số SV : 20110697

Thời gian làm luận văn : từ : 08/01/2024 Đến : 01/07/2024

Chuyên ngành : Công nghệ phần mềm

Tên luận văn : Xây dựng hệ thống hỗ trợ người dùng mua bán cafe

GV hướng dẫn : ThS. Lê Thị Minh Châu

Nhiệm Vụ Của Luận Văn :

1. Tích hợp hệ thống đề xuất sản phẩm.
2. Tích hợp hệ thống Chatbot AI.
3. Triển khai dự án lên môi trường thực tế.

Đề cương viết luận văn :

MỤC LỤC

1. Phần MỞ ĐẦU

- 1.1. Tính cấp thiết của đề tài
- 1.2. Mục đích của đề tài
- 1.3. Cách tiếp cận và phương pháp nghiên cứu
- 1.4. Phân tích những công trình có liên quan
- 1.5. Kết quả dự kiến đạt được

2. Phần NỘI DUNG

1. Chương 1: Cơ sở lý thuyết
 - 1.1. Springboot
 - 1.2. Angular
 - 1.3. Structured query language - SQL
 - 1.4. Thuật toán K-Nearest neighbor KNN – Recommendation system

- 1.5. Chatbot AI và Model LLM của Open AI
- 2. Chương 2: Yêu cầu người dùng
 - 2.1. Yêu cầu nghiệp vụ
 - 2.2. Các phương pháp tiếp cận
 - 2.3. Sản phẩm bàn giao cuối
- 3. Chương 3: Phân tích hệ thống
 - 3.1. Xác định vai trò Actor trong lược đồ Use case
 - 3.2. Mô tả vai trò người dùng
 - 3.3. Lược đồ Use case
 - 3.4. Đặc tả Use case
 - 3.5. Lược đồ activity
 - 3.6. Lược đồ sequence
 - 3.7. Recommendation system – K – nearest neighbor
 - 3.8. Chatbot AI – Model LLM của OpenAI
- 4. Chương 4: Thiết kế hệ thống
 - 4.1. Thiết kế Database
 - 4.2. Giao diện và thiết kế quy trình
- Chương 5: Cài đặt và kiểm thử
 - 5.1. Chuẩn bị
 - 5.2. CI/CD
 - 5.3. Unit test

KẾT LUẬN

PHỤ LỤC

DANH MỤC TÀI LIỆU THAM KHẢO

KẾ HOẠCH THỰC HIỆN

Ngày tháng năm 202

Người viết đề cương

Ý kiến của giáo viên hướng dẫn

.....

.....

.....

.....

.....

.....

.....

(ký và ghi rõ họ tên)

MỤC LỤC

DANH SÁCH BẢNG	4
DANH SÁCH HÌNH ẢNH	5
MỞ ĐẦU	8
NỘI DUNG.....	17
Chương 1: CƠ SỞ LÝ THUYẾT.....	17
1.1. SPRING BOOT	17
1.2. ANGULAR.....	18
1.3. STRUCTURED QUERY LANGUAGE – SQL	19
1.4. THUẬT TOÁN K-NEAREST NEIGHBORS KNN – RECOMMENDATION SYSTEM.....	20
1.4.1. Khái niệm	20
1.4.2. Độ phù hợp của thuật toán đối với dự án	23
1.4.3. Thư viện sử dụng.....	24
1.5. CHATBOT AI VÀ MODEL LLM CỦA OPEN AI.....	28
1.5.1. Khái niệm	28
1.5.2. Độ phù hợp model LLM của OPEN AI đối với dự án.....	28
1.5.3. Thư viện sử dụng.....	29
Chương 2: YÊU CẦU NGƯỜI DÙNG.....	38
2.1. YÊU CẦU NGHIỆP VỤ	38
2.2. CÁC PHƯƠNG PHÁP TIẾP CẬN	39
2.3. SẢN PHẨM BÀN GIAO CUỐI	39
Chương 3: PHÂN TÍCH HỆ THỐNG	40
3.1. XÁC ĐỊNH VAI TRÒ ACTOR TRONG LUỢC ĐỒ USE CASE	40
3.2. MÔ TẢ VAI TRÒ NGƯỜI DÙNG.....	40
3.2.1. Chi tiết vai trò actor.....	40
3.2.2. Chi tiết chức năng	42
3.3. LUỢC ĐỒ USE CASE	43
3.4. ĐẶC TẢ USE CASE	46
3.4.1 Xem giỏ hàng	46
3.4.2 Thêm sản phẩm vào giỏ hàng	46
3.4.3 Xóa sản phẩm khỏi giỏ hàng.....	47

3.4.4 Đặt hàng	48
3.4.5 Thanh toán bằng ví điện tử Momo	49
3.4.6 Thanh toán bằng cổng thanh toán Vnpay	50
3.4.7 Trò chuyện cùng Chatbot AI.....	52
3.4.8 Xem danh sách sản phẩm được đề xuất	52
3.5. LUẬT ĐỎ ACTIVITY	54
3.5.1 Xem giỏ hàng	54
3.5.2 Thêm sản phẩm vào giỏ hàng	54
3.5.3 Xóa sản phẩm khỏi giỏ hàng.....	55
3.5.4 Đặt hàng	55
3.5.5 Thanh toán bằng ví điện tử Momo	56
3.5.6 Thanh toán bằng cổng thanh toán Vnpay	57
3.5.7 Trò chuyện cùng Chatbot AI.....	58
3.5.8 Xem danh sách sản phẩm được đề xuất	58
3.6. LUẬT ĐỎ SEQUENCE.....	59
3.6.1 Xem giỏ hàng	59
3.6.2 Thêm sản phẩm vào giỏ hàng	60
3.6.3 Xóa sản phẩm khỏi giỏ hàng.....	60
3.6.4 Đặt hàng	61
3.6.5 Thanh toán bằng ví điện tử Momo	62
3.6.6 Thanh toán bằng cổng thanh toán Vnpay	63
3.6.7 Trò chuyện cùng Chatbot AI.....	64
3.6.8 Xem danh sách sản phẩm đề xuất	65
3.7. RECOMMENDATION SYSTEM – K-NEAREST NEIGHBOR .	66
3.7.1 Tiền xử lý dữ liệu	66
3.7.2 Kiểm tra độ tin cậy của model	68
3.7.3 Chọn tham số k.....	70
3.7.4 Xây dựng hàm	74
3.8. CHATBOT AI – MODEL LLM CỦA OPENAI.....	75
3.8.1 Sơ hô hoạt động.....	75
3.8.2 Tích hợp LLM của OpenAI vào hệ thống chatbot.....	75
Chương 4: THIẾT KẾ HỆ THỐNG	79

4.1. THIẾT KẾ DATABASE.....	79
4.1.1. Lược đồ ERD	79
4.1.2. Chi tiết bảng trong database.....	85
4.1.3. Class diagram	86
4.2. GIAO DIỆN VÀ THIẾT KẾ QUY TRÌNH.....	88
Chương 5: CÀI ĐẶT VÀ KIỂM THỦ.....	96
 5.1. CHUẨN BỊ.....	96
5.1.1. Công cụ	96
5.1.2. Công nghệ	96
 5.2. CI/CD.....	96
5.2.1. Khái niệm	96
5.2.2. Thực thi	97
 5.3. UNIT TEST.....	101
5.3.1. Khái niệm	101
5.3.2. Thực thi	101
KẾT LUẬN	104
KẾ HOẠCH THỰC HIỆN.....	106
DANH MỤC TÀI LIỆU THAM KHẢO.....	108
PHỤ LỤC	110

DANH SÁCH BẢNG

Bảng 1 Bảng so sánh các trang web đã khảo sát	16
Bảng 2 So sánh các thuật toán thường được sử dụng trong hệ thống đề xuất	23
Bảng 3 So sánh KNN, SVD, NMF.....	26
Bảng 4 So sánh KNNBasic, KNNWithMeans, KNNWithZScore	26
Bảng 5 So sánh model LLM giữa Open AI và Hugging Face	28
Bảng 6 So sánh Word Embeddings và Sentence/ Document Embeddings	36
Bảng 7 Vai trò actors trong lược đồ usecase	40
Bảng 8 Bảng chi tiết vai trò actors	40
Bảng 9 Bảng chi tiết chức năng.....	42
Bảng 10 Đặc tả usecase xem giỏ hàng	46
Bảng 11 Đặc tả usecase thêm sản phẩm vào giỏ hàng	46
Bảng 12 Đặc tả usecase xóa sản phẩm khỏi giỏ hàng	47
Bảng 13 Đặc tả usecase đặt hàng	48
Bảng 14 Đặc tả usecase thanh toán bằng ví điện tử Momo	49
Bảng 15 Đặc tả usecase bằng công thanh toán Vnpay	50
Bảng 16 Đặc tả usecase trò chuyện cùng Chatbot AI	52
Bảng 17 Đặc tả usecase xem danh sách sản phẩm được đề xuất	52
Bảng 18 Bảng dữ liệu mẫu cho thuật toán KNN	66
Bảng 19 Tập dữ liệu mẫu cho thuật toán KNN	67
Bảng 20 Dataset cho thuật toán KNN	67
Bảng 21 Kết quả của GridSearchCV	70
Bảng 22 30 lần đo giá trị RMSE	72
Bảng 23 Ưu điểm và nhược điểm của CI/CD	97
Bảng 24 Bảng phân công công việc	106
Bảng 25 Danh sách những từ tiếng anh được sử dụng.....	110

DANH SÁCH HÌNH ẢNH

Hình 1 Website siêu thị cà phê – từ khóa tìm kiếm.....	11
Hình 2 Website siêu thị cà phê – banner và header [1].....	12
Hình 3 Website siêu thị cà phê – danh sách hình ảnh sản phẩm [1]	12
Hình 4 Trung Nguyên Coffee – trang chủ [2]	13
Hình 5 Trung Nguyên Coffee – giỏ hàng [2]	13
Hình 6 Cà phê nguyên chất – tra cứu sản phẩm [3]	14
Hình 7 Cà phê nguyên chất – chi tiết sản phẩm [3]	15
Hình 8 Cà phê nguyên chất – giỏ hàng [3].....	15
Hình 9 Công thức KNN basic [17]	27
Hình 10 Các thành phần của Agent [18]	32
Hình 11 Sơ đồ hoạt động của Chatbot AI [19].....	33
Hình 12 SQL Agent [20]	37
Hình 13 Usecase Diagram – Mối liên quan giữa các tác nhân (Actors)	43
Hình 14 Quyền khách vãng lai - Guest Rights	44
Hình 15 Quyền khách hàng - User rights	44
Hình 16 Quyền nhân viên – Employee rights	45
Hình 17 Quyền Admin – Admin rights	45
Hình 18 Activity diagram – xem giỏ hàng	54
Hình 19 Activity diagram – thêm sản phẩm vào giỏ hàng	54
Hình 20 Activity diagram – xóa sản phẩm khỏi giỏ hàng	55
Hình 21 Activity diagram – đặt hàng	55
Hình 22 Activity diagram – thanh toán bằng ví điện tử Momo	56
Hình 23 Activity diagram – thanh toán bằng cổng thanh toán Vnpay	57
Hình 24 Activity diagram – trò chuyện cùng Chatbot AI	58
Hình 25 Activity diagram – xem danh sách sản phẩm được đề xuất	58
Hình 26 Sequence diagram – xem giỏ hàng	59
Hình 27 Sequence diagram – thêm sản phẩm vào giỏ hàng.....	60
Hình 28 Sequence diagram – xóa sản phẩm khỏi giỏ hàng.....	60
Hình 29 Sequence diagram – đặt hàng	61
Hình 30 Sequence diagram – thanh toán bằng ví điện tử Momo	62

Hình 31 Sequence diagram – thanh toán bằng cổng thanh toán Vnpay.....	63
Hình 32 Sequence diagram – Trò chuyện cùng Chatbot AI.....	64
Hình 33 Sequence diagram – Xem danh sách sản phẩm được đề xuất.....	65
Hình 34 Biểu đồ RMSE so với Cosine, Pearson và MSD lần 1.....	71
Hình 35 Biểu đồ RMSE so với Cosine, Pearson và MSD lần 2.....	71
Hình 36 Biểu đồ RMSE so với Cosine, Pearson và MSD lần 3.....	72
Hình 37 Công thức Cosine [15].....	73
Hình 38 Sơ đồ hoạt động của ChatBot OpenAI.....	75
Hình 39 Entity relationship diagram (ERD).....	79
Hình 40 Dữ liệu bảng cart, order_detail, product, user, sale.....	80
Hình 41 Dữ liệu bảng invoice, shipping-method, review, product_origin, flavor	81
Hình 42 Dữ liệu của bảng brand, wishlist_item, vendor, product_image, wishlist	82
Hình 43 Dữ liệu bảng product_detail, cart_item, order, category, employee, wishlist.	83
Hình 44 ERD - Mối liên hệ giữa các entity.....	84
Hình 45 Class Diagram	88
Hình 46 Giao diện trang chủ	89
Hình 47 Giao diện trang danh sách sản phẩm	89
Hình 48 Giao diện trang chi tiết sản phẩm	90
Hình 49 Giao diện trang giỏ hàng	90
Hình 50 Giao diện trang đặt hàng.....	91
Hình 51 Giao diện trang hồ sơ tài khoản khách hàng	91
Hình 52 Giao diện trang đăng nhập quản trị	92
Hình 53 Giao diện trang doanh thu	92
Hình 54 Giao diện trang quản trị sản phẩm.....	93
Hình 55 Giao diện chọn địa điểm giao hàng	93
Hình 56 Giao diện chọn phí vận chuyển	94
Hình 57 Giao diện phần gọi ý sản phẩm	94
Hình 58 Giao diện chatbot AI	95
Hình 59 CI/CD GitHub Action.....	98
Hình 60 CI/CD – các giai đoạn của việc build và deploy Spring Boot.....	99
Hình 61 Unit test – brand	102
Hình 62 Unit test – category.....	102

Hình 63 Unit test – flavor	103
Hình 64 Unit test – product detail	103

MỞ ĐẦU

1.1. TÍNH CẤP THIẾT CỦA ĐỀ TÀI

Đáp ứng nhu cầu thị trường: trong thời điểm hiện nay, cà phê dần khẳng định được sự phổ biến của mình trên khắp thế giới, những mặt hàng của sản phẩm này xuất hiện mọi nơi, từ đường phố, nhà hàng, khách sạn, hay trên những trang quảng cáo, mạng internet,... Nhưng vì đợt dịch Covid vừa qua, nhu cầu mua bột cà phê đóng hộp ngày càng tăng. Điều này chứng minh rằng việc mua sắm trực tuyến cà phê ngày càng được phổ biến rộng rãi.

Tiện ích cho người tiêu dùng: trang web bán cà phê giúp người dùng dễ dàng tra cứu thông tin của sản phẩm, dễ dàng đặt hàng, thanh toán qua ví điện tử hoặc cổng thanh toán.

Quản lý hiệu quả: giúp đội ngũ nhân viên cửa hàng thuận tiện trong việc quản lý thông tin sản phẩm, doanh thu. Với việc sử dụng Spring Boot và SQL để quản lý và tổ chức thông tin, trang web này sẽ khiến việc quản lý trở nên mượt mà và chính xác.

Phản hồi và tương tác: trang web được thiết kế để phù với nhu cầu phản hồi, tương tác giữa các bên liên quan, từ đó có thể dễ dàng tổng hợp và nâng cao chất lượng sản phẩm, vận hành của doanh nghiệp

1.2. MỤC ĐÍCH CỦA ĐỀ TÀI

Xây dựng một dự án website thương mại điện tử kinh doanh cà phê có đặc điểm và chức năng sau:

- Sử dụng công nghệ Angular (frontend), Spring Boot (Backend), Sql (Database).
- Cung cấp đủ chức năng của một website thương mại điện tử, bao gồm quản lý thông tin liên quan tới sản phẩm, người dùng, đơn hàng, nhân viên.
- Sử dụng API từ Giao Hàng Nhanh Express.
- Tích hợp thanh toán trực tuyến thông qua ví điện tử Momo và cổng thanh toán Vnpay.

- Tích hợp hệ thống để xuất sản phẩm cho người dùng.
- Tích hợp hệ thống Chatbot AI, hỗ trợ người dùng tra cứu thông tin nhanh chóng.
- Triển khai CI/CD cho dự án, tăng tốc độ xử lý của website và dễ dàng bảo trì.

1.3. CÁCH TIẾP CẬN VÀ PHƯƠNG PHÁP NGHIÊN CỨU

Đối tượng nghiên cứu

Khách hàng tiêu thụ cà phê: tìm hiểu, nghiên cứu về người dùng tiêu thụ cà phê ở khía cạnh mặt hàng, hương vị, nguồn gốc, thương hiệu, hình thức thanh toán để từ đó có thể thiết kế mô hình kinh doanh phù hợp.

Quản trị doanh nghiệp cà phê: tìm hiểu cách vận hành của một doanh nghiệp kinh doanh trực tuyến cà phê, tham khảo từ những trang web bán hàng trên internet, từ đó đúc kết được những ưu và nhược điểm và phát triển cho trang web của doanh nghiệp

Loại hình giao dịch: nghiên cứu cách vận hành của các loại hình thanh toán trực tuyến hiện đang tồn tại, và tích hợp chúng vào trang web của doanh nghiệp.

Phạm vi nghiên cứu

Dự án xây dựng một website kinh doanh các mặt hàng về cà phê, cho phép khách hàng tiếp cận những loại cà phê mới, thông qua website có thể tra cứu, đặt mua, thanh toán trực tiếp. Doanh nghiệp dựa vào giao diện trên trang quản trị tối ưu hóa quy trình quản lý doanh nghiệp.

Phạm vi dữ liệu:

- Dữ liệu các cửa hàng trực tuyến trên Internet.
- Các loại sản phẩm cà phê.

Công nghệ thực hiện:

- Front-end:
 - HTML5, SCSS

- Bootstrap
- TypeScript
- Angular
- Backend:
 - Spring Boot
- Database:
 - SQL
- AI:
 - Surprise – để xuất sản phẩm
 - LangChain và Model LLM của OpenAI – Chatbot AI

Yêu cầu hệ thống:

- Giao diện đẹp mắt, dễ dàng sử dụng đối với khách hàng.
- Dễ dàng quản lý và điều hành.
- Dễ dàng nâng cấp và bảo trì hệ thống.

Yêu cầu người dùng:

- Chức năng cơ bản của trang Website thương mại điện tử: tra cứu, giỏ hàng, đặt hàng, thanh toán, phản hồi.
- Quản lý thông tin: danh mục, sản phẩm, hình ảnh, giá, nguồn gốc, nhà cung cấp, thương hiệu, đơn hàng, doanh thu.
- Xử lý đơn hàng: nghiên cứu chuyên sâu về quy trình đặt hàng, bao gồm đặt hàng, duyệt đơn, giao hàng, hủy đơn hàng, đổi trả.
- Phản hồi và tương tác khách hàng: từ đánh giá của khách hàng để đúc kết lại chất lượng của sản phẩm và trang website, từ đó để có những hiệu chỉnh thích hợp.
- Tích hợp thanh toán an toàn: tích hợp ví điện tử, cổng thanh toán cho quy trình thanh toán, từ đó khách hàng có đa dạng cách xử lý đơn đặt hàng.
- Tích hợp hệ thống để xuất sản phẩm và Chatbot AI nhằm tăng tốc độ tra cứu thông tin của khách hàng.

- Tự động hóa quy trình kinh doanh: tự động quản lý thông tin website, đơn hàng, doanh thu trong trang quản trị.
- An toàn dữ liệu và bảo mật: tìm hiểu, nghiên cứu các lỗ hổng liên quan tới website, đặc biệt là website thương mại điện tử để đưa ra các giải pháp bảo mật dữ liệu của khách hàng và doanh nghiệp.

Đơn vị thực hiện: Nhóm 60 – CNPM _Đại Trà

Ước lượng thời gian hoàn thành: Khoảng 6 tháng

- Ngày bắt đầu: 08/01/2024.
- Ngày kết thúc: 01/07/2024.

Tổng kinh phí cho dự án: 1500000 VNĐ (một triệu năm trăm ngàn đồng)

1.4. PHÂN TÍCH NHỮNG CÔNG TRÌNH CÓ LIÊN QUAN

1.4.1. Khảo sát website <https://sieuthicafe.vn/>

Từ khóa tìm kiếm: mua cà phê online.



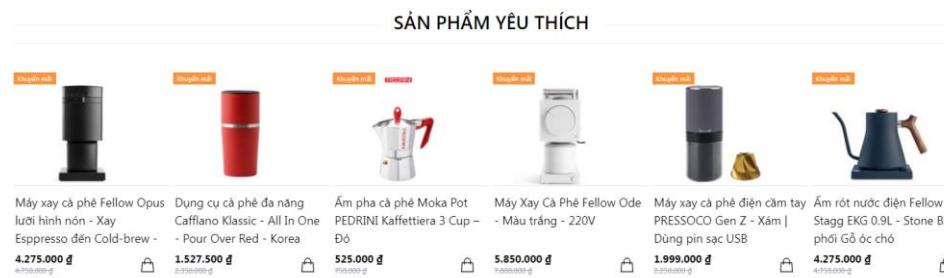
Hình 1 Website siêu thị cà phê – từ khóa tìm kiếm

Khi vào trang web có thể thấy giao diện khá là bắt mắt với màu trắng chủ thể và một banner cuốn hút. Có phần tìm kiếm sản phẩm. Thanh công cụ sử dụng nền đen chữ trắng khá hài hòa và nổi bật.



Hình 2 Website siêu thị cà phê – banner và header [1]

Kéo xuống dưới có các danh mục đề xuất như sản phẩm yêu thích, dụng cụ pha chế thế giới. Khi nhấp vào sản phẩm bất kỳ sẽ điều hướng đến trang chi tiết sản phẩm.



DỤNG CỤ PHA CHẾ THẾ GIỚI

Nơi tập hợp và sưu tầm đa dạng phương pháp pha và yêu thích trên toàn thế giới



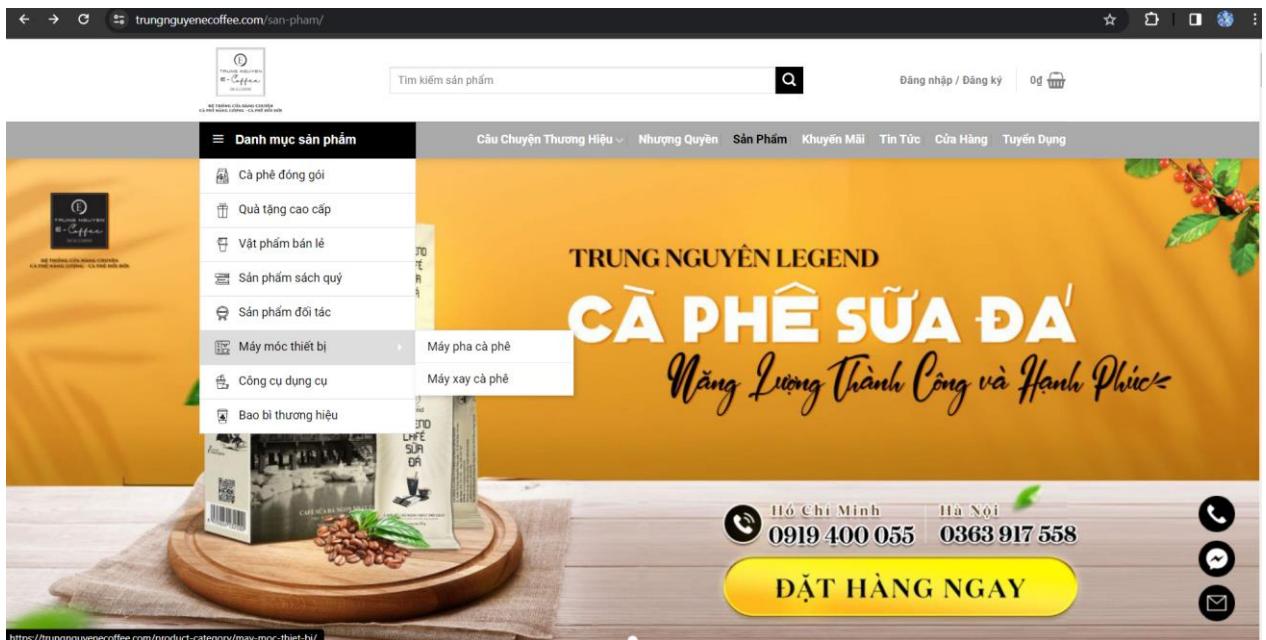
Hình 3 Website siêu thị cà phê – danh sách hình ảnh sản phẩm [1]

Bên cạnh banner, header, trang chủ hiển thị danh sách sản phẩm, còn có trang chi tiết sản phẩm, giỏ hàng, trang đặt hàng, footer.

Nhận xét: Trang web này được thiết kế khá hợp lý, đầy đủ các chức năng. Tuy nhiên ở phần trang chủ, nó được chèn khá nhiều hình ảnh khiến cho trang web bị dài ra.

1.4.2. Khảo sát website <https://trungnguyencoffee.com/>

Banner tông nền màu vàng, kích thước chiếm đa số khi khách hàng vừa mới truy cập vào website, thanh navigation bar với tông màu đen, xám.



Hình 4 Trung Nguyên Coffee – trang chủ [2]

Giỏ hàng bố cục trực quan, tông màu trắng, đen, xám.

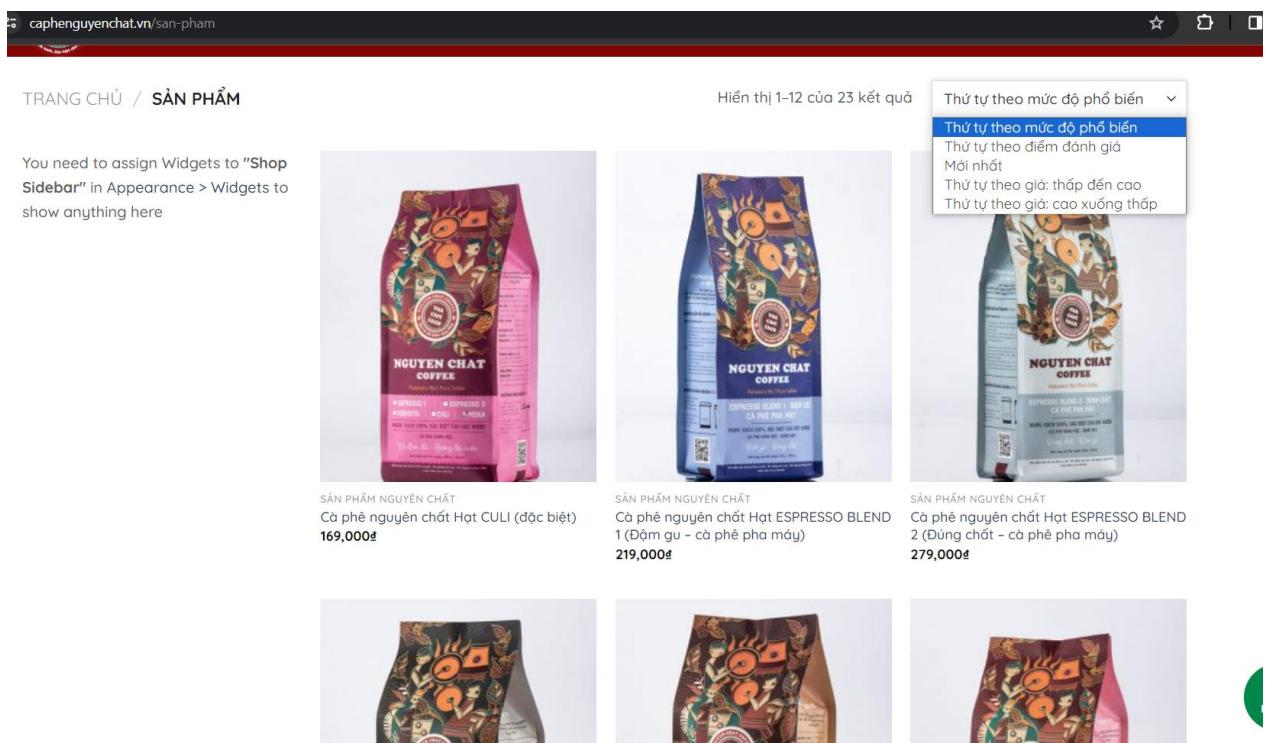
A screenshot of the Trung Nguyen Coffee website's shopping cart page. The header includes a search bar and navigation links. The main content shows a table with a single item: 'G7 2in1 hộp 15 sachets' at '50.000đ' with a quantity of '1'. To the right, a 'CỘNG GIỎ HÀNG' section shows the total as '50.000đ'. Below it, shipping information is listed: 'Phi vận chuyển: 15.000đ', 'Vận chuyển đến Hồ Chí Minh.', and 'Đổi địa chỉ'. At the bottom is a large black button labeled 'Tiến hành thanh toán'.

Hình 5 Trung Nguyên Coffee – giỏ hàng [2]

Nhận xét: Website Trung Nguyên Coffee có bố cục trực quan, đơn giản. Bên cạnh banner, giỏ hàng, trang hiển thị danh sách sản phẩm trực quan, dễ dàng thực hiện các thao tác khác. Tuy nhiên phần tra cứu sản phẩm vẫn chưa hỗ trợ tìm kiếm theo nhu cầu người dùng như mức giá mong muốn.

1.4.3. Khảo sát website <https://caphenguyenchat.vn/>

Danh sách sản phẩm bao gồm hình ảnh, giá cả, tên. Hỗ trợ tra cứu với mức độ phổ biến, điểm đánh giá, mới nhất và giá cả.



Hình 6 Cà phê nguyên chất – tra cứu sản phẩm [3]

Trong trang chi tiết sản phẩm, hình ảnh sản phẩm có kích thước lớn, bên cạnh là những thông tin cần biết về sản phẩm.

Cà phê nguyên chất Hạt ESPRESSO BLEND 1 (Đậm gu - cà phê pha máy)

219,000đ

Giá Chiết Khấu Đại Lý Gia Đình Cafe Sạch Chỉ Còn
82.000 VND

- Tên tiếng Anh: Espresso Blend
- Tên tiếng Việt: Cafe Espresso Blend Nguyên Chất
- Mùi vị: Cafe Nguyên Chất Hạt Espresso Blend sạch 100% đặc biệt có hương vị tinh túy tự nhiên nhất được giữ nguyên vẹn trong từng giọt cà phê nguyên chất theo phong cách Ý. Mùi thơm nồng nàn gây ấn tượng và chinh phục. Hậu vị kéo dài, truyền cảm hứng.
- Phong cách: Gu Tây và người Việt, phù hợp với Nam/Nữ
- Pha chế: Pha Espresso, uống với đường hay sữa (Nóng hay đá) tùy thích.



Hình 7 Cà phê nguyên chất – chi tiết sản phẩm [3]

Giỏ hàng bô cục đơn giản, rõ ràng, dễ sử dụng.

SẢN PHẨM	GIÁ	SỐ LƯỢNG	TẠM TÍNH
 Cà phê nguyên chất Hạt ESPRESSO BLEND 1 (Đậm gu - cà phê pha máy)	219,000đ	<input type="button" value="-"/> <input checked="" type="button" value="1"/> <input type="button" value="+"/>	219,000đ

[← TIẾP TỤC XEM SẢN PHẨM](#)
[CẬP NHẬT GIỎ HÀNG](#)

CỘNG GIỎ HÀNG	
Tạm tính	219,000đ
Tổng	219,000đ

[TIẾN HÀNH THANH TOÁN](#)

Hình 8 Cà phê nguyên chất – giỏ hàng [3]

Nhận xét: đã đáp ứng những chức năng cơ bản của website thương mại điện tử, nhưng bô cục hình ảnh sản phẩm cần được hiệu chỉnh để phù hợp với bô cục trang web.

1.4.4. So sánh các trang web đã khảo sát

Bảng 1 Bảng so sánh các trang web đã khảo sát

Tiêu chí Trang web	Giao diện bắt mắt, bô cục rõ ràng	Tốc độ tải trang web	Độ thân thiện, dễ sử dụng	Xếp hạng
sieuthicafe.vn	7.5	8.5	7	2
trungnghuyencoffee.com	8	7	8.5	1
caphenguyenchat.vn	7.5	8	7	3

Sau khi nghiên cứu 3 website ở trên, mỗi website tồn tại những ưu và nhược điểm riêng. Từ đó tổng hợp và đề ra một hướng đi thiết kế để đáp được nhu cầu của người dùng, doanh nghiệp.

1.5. KẾT QUẢ DỰ KIẾN ĐẠT ĐƯỢC

Dự án website thương mại điện tử kinh doanh cà phê có đặc điểm và chức năng sau:

- Sử dụng công nghệ Angular (frontend), Spring Boot (Backend), Sql (Database).
- Cung cấp đủ chức năng của một website thương mại điện tử, bao gồm quản lý thông tin liên quan tới sản phẩm, người dùng, đơn hàng, nhân viên.
- Sử dụng API từ Giao Hàng Nhanh Express.
- Tích hợp thanh toán trực tuyến thông qua ví điện tử Momo và cổng thanh toán Vnpay.
- Tích hợp hệ thống đề xuất sản phẩm.
- Tích hợp hệ thống Chatbot AI.
- Triển khai CI/CD.

NỘI DUNG

Chương 1: CƠ SỞ LÝ THUYẾT

1.1. SPRING BOOT

Java Spring Boot là một framework phổ biến, mã nguồn mở, enterprise – level (tập hợp các quy trình, công cụ, khuôn mẫu thường được sử dụng để xây dựng kế hoạch, xây dựng hệ thống kiến trúc ứng dụng mức độ doanh nghiệp) trong việc tạo ra một ứng dụng độc lập hoặc production – grade (phần mềm chất lượng, làm việc mượt mà và xử lý được nhiều người dùng). Spring Boot là một công cụ hỗ trợ khiết cho quá trình phát triển ứng dụng web và các microsoftware trở nên nhanh hơn, dễ dàng hơn thông qua các điểm nổi bật sau đây:

- Tự động cấu hình: Spring Boot cung cấp cơ chế tự động cấu hình các thành phần của ứng dụng dựa trên các thư viện dependencies. Điều này giúp người lập trình tiết kiệm thời gian, loại bỏ hầu hết cấu hình thủ công.
- Hỗ trợ những Server nhúng: Spring Boot hỗ trợ các servers nhúng như Tomcat, Jetty, Undertow. Có nghĩa là ứng dụng có thể được gói vào file jar hoặc war với server nhúng, và từ đó đơn giản giai đoạn triển khai ra thực tế, giảm sự phụ thuộc vào bên ngoài.
- Kiến trúc microservices: Spring Boot rất phù hợp với việc xây dựng kiến trúc microservices.
- Spring Boot Starter: là các bản mẫu đã được cấu hình trước đó và được sử dụng trong rất nhiều trường hợp, giúp đơn hóa việc cấu hình các dependencies, cho phép lập trình viên thêm trực tiếp những hàm chức năng mà không cần phải viết thủ công.
- Spring Boot CLI: cho phép lập trình viên xây dựng ứng dụng và chạy ứng dụng Spring Boot trực tiếp trên Command Line, hỗ trợ Groovy Script.

Với những đặc tính và tiện ích mà Spring Boot mang lại, nó trở thành một lựa chọn mạnh mẽ cho việc phát triển ứng dụng Java hiện đại và dễ bảo trì.

1.2. ANGULAR

Angular là một framework front-end phổ biến, được sử dụng để xây dựng các ứng dụng web dynamic. Angular cung cấp một tập các tính năng toàn diện, những tính năng này đã làm tăng độ phổ biến của Angular và khiến nó nổi trội hơn so với những framework front-end khác. Sau đây là một số điểm nổi bật:

- Two-Way Data Binding: cho phép tự động đồng bộ hóa dữ liệu giữa các model và view. Những sự thay đổi trong dữ liệu model sẽ được phản ánh trực tiếp trên view và ngược lại. Điều này đơn giản hóa việc quản lý và thiết kế giao diện người dùng và giảm đi việc thao tác với DOM.
- Xem từng component là module: Angular sử dụng kiến trúc component-based. Các phần khác nhau trên giao diện người dùng sẽ được đóng gói trong các phần riêng biệt và được xử lý riêng biệt, có thể tái sử dụng các thành phần này.
- Dependency Injection: Angular có hệ thống dependency injection giúp có thể quản lý và sử dụng thư viện dễ dàng hơn.
- Typescript language: Angular được phát triển dựa trên ngôn ngữ typescript, là một phân nhánh của javascript. Ngôn ngữ này giúp bắt lỗi trong quá trình phát triển một cách tốt hơn, cung cấp những công cụ hỗ trợ mạnh mẽ, cải thiện việc bảo trì.
- Angular CLI: Angular cung cấp tính năng làm việc trực tiếp với command line, đơn giản hóa việc tạo, xây dựng, triển khai, kiểm thử ứng dụng Angular.
- RxJS và Observables: Angular tích hợp các thư viện này giúp xử lý các yêu cầu, sự kiện một cách bắt đồng bộ, từ đó tăng khả năng trải nghiệm của người dùng.
- Angular Universal: cho phép người dùng có thể tự render bên phía server.
- Community mạnh mẽ: Cộng đồng lớn và tích cực của Angular đảm bảo rằng có nhiều tài nguyên học tập và hỗ trợ từ cộng đồng.

Angular là một lựa chọn mạnh mẽ cho việc phát triển các ứng dụng web phức tạp và hiện đại, đặc biệt là khi cần một framework linh hoạt, dễ bảo trì và có khả năng mở rộng.

1.3. STRUCTURED QUERY LANGUAGE – SQL

Structured Query Language (SQL) là một ngôn ngữ lập trình chuyên dụng được sử dụng để quản lý và tương tác với các hệ thống quản lý cơ sở dữ liệu quan hệ (Relational Database Management System - RDBMS). SQL cung cấp một phương tiện mạnh mẽ để thực hiện các thao tác như truy vấn dữ liệu, cập nhật dữ liệu, chèn dữ liệu mới, xóa dữ liệu, và quản lý cấu trúc của cơ sở dữ liệu.

Dưới đây là một số điểm quan trọng về SQL:

- Chia ra nhiều loại lệnh: Data Query Language (DQL) được sử dụng để truy vấn dữ liệu từ cơ sở dữ liệu, thường sử dụng lệnh SELECT; Data Definition Language (DDL) được sử dụng để định nghĩa cấu trúc của cơ sở dữ liệu, thường sử dụng lệnh CREATE, ALTER, DROP; Data Manipulation Language (DML) được sử dụng để thay đổi dữ liệu trong cơ sở dữ liệu, thường sử dụng lệnh INSERT, UPDATE, DELETE.
- Có khả năng mở rộng: SQL không chỉ được sử dụng cho hệ thống quản lý cơ sở dữ liệu quan hệ truyền thống mà còn có thể được áp dụng cho nhiều loại cơ sở dữ liệu khác nhau như cơ sở dữ liệu không quan hệ, cơ sở dữ liệu đối tượng.
- Quản lý dữ liệu quan hệ: SQL chủ yếu được thiết kế để làm việc với dữ liệu quan hệ, trong đó dữ liệu được tổ chức thành các bảng có mối quan hệ với nhau.
- Hỗ trợ truy vấn phức tạp: SQL hỗ trợ truy vấn dữ liệu phức tạp thông qua các câu lệnh JOIN, GROUP BY, HAVING, ORDER BY, giúp người phát triển thực hiện các tác vụ phức tạp trên dữ liệu.
- An toàn và bảo mật: SQL cung cấp cơ chế an toàn và bảo mật để quản lý quyền truy cập dữ liệu, giúp bảo vệ thông tin quan trọng khỏi truy cập trái phép.

SQL đóng vai trò quan trọng trong quản lý và tương tác với cơ sở dữ liệu trong nhiều ứng dụng và hệ thống thông tin, là công cụ không thể thiếu cho các nhà phát triển và quản trị cơ sở dữ liệu.

1.4. THUẬT TOÁN K-NEAREST NEIGHBORS KNN – RECOMMENDATION SYSTEM

1.4.1. Khái niệm

Trong các bài toán hồi quy (regression) và bài toán phân loại (classification), Thuật toán K-Nearest Neighbor (KNN) được sử dụng rỗng rãi vì sự đơn giản và hiệu quả của ngôn ngữ học máy này. Với thông tin k điểm dữ liệu gần nhất, thuật toán này sẽ xác định được nhãn (đối với bài toán classification) hoặc xác định được giá trị (đối với bài toán hồi quy) của một điểm dữ liệu đầu vào.

Vì thuật toán dựa trên thông tin từ các thực thể trong không gian để xác định hoặc đưa ra các quyết định đối với thực thể mới, thuật toán K-Nearest Neighbor (KNN) là phương pháp học có giám sát. Như đã nói, cốt lõi của thuật toán này là dựa vào những thông tin có được từ k thực thể tồn tại trong không gian và từ những thông tin đó đưa ra dự đoán về giá trị hoặc phân loại cho dữ liệu mới. Vì tập dữ liệu không cần phải giả định cụ thể về cấu trúc dữ liệu khiến cho thuật toán KNN trở lên linh hoạt và áp dụng trong nhiều tình huống.

Ví dụ về việc học có giám sát và không giám sát:

Học có giám sát: Nhãn của một thực thể mới sẽ được quyết định dựa trên tập dữ liệu đã được cung cấp trước. Ví dụ phân loại email có spam hay không dựa trên tập dữ liệu đã được cung cấp trước đó (từ khóa, tần suất lặp từ, địa chỉ người gửi)

Học không giám sát: Làm việc với dữ liệu không được gắn nhãn và cố gắng tìm ra cấu trúc hoặc mẫu từ tập dữ liệu. Sau đó dựa vào mẫu mới tạo ra để xác định nhãn cho thực thể mới. Ví dụ như bài toán phân loại khách hàng, với tập dữ liệu khách hàng chưa được phân loại, thuật toán sẽ tiến hành học và phân cụm thành từng nhóm. Sau đó gán nhãn cho từng loại khách hàng.

Ví dụ về tập dữ liệu giải định cụ thể về cấu trúc dữ liệu và không giả định cụ thể về cấu trúc dữ liệu:

Tập dữ liệu không giả định cụ thể về cấu trúc dữ liệu: Xác định chủ đề của một bài báo ("Thể thao", "Chính trị", "Kinh tế") dựa trên tập bài báo đã được cung cấp. Mỗi bài báo được biểu diễn bằng một vector dựa trên tần suất xuất hiện của các từ khóa. Ở đây,

ta không hề thấy bất kỳ ràng buộc gì giữa những từ khóa có trong tờ báo hoặc bất kỳ mối quan hệ tuyến tính nào giữa từ khóa (đặc trưng) và chủ đề tờ báo (nhãn). Chủ đề của một tờ báo không bị quyết định một cách rõ ràng bởi việc xuất hiện nhiều hay ít của một từ khóa nhất định.

Tập dữ liệu giả định cụ thể về cấu trúc dữ liệu: dự đoán giá nhà dựa trên diện tích, số căn phòng, cửa sổ, tiện ích, số tầng. Ở đây, ta thấy giữa các đặc trưng và nhãn (giá nhà) có mối quan hệ tuyến tính. Nếu xét theo thực tế thì nếu diện tích càng tăng thì giá nhà càng tăng, hoặc tiện ích càng nhiều thì giá nhà càng tăng.

Trong lĩnh vực hệ thống gợi ý (Recommendation System), thuật toán này dựa trên sở thích, hành vi của người dùng để đề xuất những sản phẩm, dịch vụ tương tự. Dựa vào một số tiêu chí có giá trị tương đồng như khoảng cách Euclidean hoặc khoảng cách cosine, KNN sẽ chọn ra k người dùng hoặc sản phẩm tương tự với người dùng hoặc sản phẩm cụ thể. Từ đó sẽ đề xuất những sản phẩm hoặc dịch vụ này tới người dùng. Trong dự án này, KNN sẽ đánh giá sự tương đồng giữa các người dùng dựa trên việc đánh giá sản phẩm của họ. Qua đó, các sản phẩm mà một người dùng cụ thể có khả năng quan tâm được lấy từ sở thích của những người dùng gần giống họ sẽ được đề xuất.

Có hai loại hệ thống gợi ý dựa trên KNN: dựa trên người dùng (user-based) và dựa trên sản phẩm (item-based). Với hệ thống dựa trên người dùng, KNN sẽ tìm kiếm k người dùng gần tương đồng với người dùng cần xét và đề xuất những sản phẩm dựa trên sở thích của k người dùng đó. Với hệ thống dựa trên sản phẩm, KNN sẽ xác định K sản phẩm tương tự với những sản phẩm mà người dùng đó thích trước đó.

Nhược điểm:

Mặc dù đơn giản và hiệu quả nhưng KNN gặp vấn đề khi tiến hành mở rộng và tình trạng ma trận thưa.

Về khả năng mở rộng, theo đánh giá khách quan thì thuật toán KNN là thuật toán lười biếng vì nó không tạo ra được mô hình dự đoán từ dữ liệu đã học trước đó mà lại tiến hành tìm k điểm dữ liệu tương đồng trong toàn bộ dữ liệu mỗi khi đưa ra dự đoán.

Điều này cực kỳ tốn kém khi tính toán, dữ liệu càng lớn thì hiệu suất càng giảm và chi phí càng tăng.

Về vấn đề ma trận thưa: trong hệ thống gợi ý, ma trận được sử dụng để biểu diễn các điểm dữ liệu (tương tác giữa người dùng và sản phẩm), ma trận này là dữ liệu đầu vào cho hệ thống gợi ý. Trong đồ án này, giá trị của các điểm trong ma trận là mức độ đánh giá của người dùng đối với sản phẩm. Tuy nhiên, không phải một người dùng đều mua toàn bộ sản phẩm, nên khi tạo ma trận dựa trên tập dữ liệu trong database, phần lớn các giá trị trong ma trận là số không xác định hoặc bằng 0. Điều này trực tiếp gây nhiều việc tính toán tìm k hàng xóm, làm giảm độ chính xác của hệ thống đề xuất.

Tuy nhiên, với sự phát triển của các kỹ thuật xử lý dữ liệu và tối ưu hóa, các tập dữ liệu đều được tiền xử lý bằng nhiều cách như giảm kích thước dữ liệu, thay đổi cách tính toán để đảm bảo KNN vẫn là một thuật toán mạnh trong hệ thống đề xuất.

1.4.2. Độ phù hợp của thuật toán đối với dự án

So sánh các loại thuật toán dùng trong hệ thống đề xuất

Bảng 2 So sánh các thuật toán thường được sử dụng trong hệ thống đề xuất

	Content-Based Filtering	Collaborative Filtering	Hybrid Filtering
Đặc điểm tập dữ liệu	Dựa vào thông tin của sản phẩm	Dựa vào hành vi tương tác của người dùng đối với sản phẩm	Kết hợp thông tin sản phẩm và hành vi người dùng
Thời gian thực thi	<ul style="list-style-type: none"> Tương đối nhanh Đòi hỏi thời gian để xây dựng và cập nhật hồ sơ người dùng trên thời gian thực (lịch sử tìm kiếm, mua hàng, giỏ hàng, đánh giá) 	Tương đối chậm vì phải quét hết toàn bộ dữ liệu trong tập dữ liệu nếu dữ liệu lớn Cần thời gian cập nhật hành vi người dùng	Thời gian thực thi tăng do tập dữ liệu phức tạp và lớn
Tài nguyên	Không quá lớn	Tiêu tốn tài nguyên hơn Content-Based Filtering do phải xử lý dữ liệu bằng ma trận, gấp ván đề ma trận thura	Cần công nghệ và kỹ thuật tiên tiến

Với tập dữ liệu của dự án, nhóm quyết định sử dụng thuật toán Collaborative Filtering bởi vì:

- Hệ thống không phát triển quản lý lịch sử tìm kiếm. Sản phẩm cà phê chỉ bao gồm những thuộc tính giá, khối lượng, loại, hương vị, nguồn gốc, nhà sản xuất. Trong đó thì các thuộc tính khối lượng, loại, hương vị, nguồn gốc, nhà sản xuất

không đa dạng trong miền giá trị nên không sử dụng thuật toán Content-Based Filtering.

- Ngân sách dự án không đủ để đáp ứng được yêu cầu tài nguyên của thuật toán Hybrid Filtering.
- Tập dữ liệu tương đối nhỏ, chỉ dựa vào việc đánh giá của người dùng đối với sản phẩm. Sử dụng thư viện Surprise của python để xử lý dữ liệu, loại bỏ vấn đề ma trận thừa.

1.4.3. Thư viện sử dụng

Thư viện: Surprise

Ngôn ngữ: Python

Trong lĩnh vực đề xuất, cần có hiểu biết nhất định về các thuật toán phức tạp để có thể triển khai chúng lên môi trường phần mềm. Trong python, thư viện Surprise là một công cụ mạnh, được thiết kế với mã nguồn mở, tích hợp nhiều thuật toán đề xuất như KNNBasic, KNNWithMeans, KNNWithZScore, SVD, SVD++, NMF,... nên thư viện này được sử dụng phổ biến mỗi khi nhắc tới hệ thống đề xuất.

Bởi vì số lượng thuật toán mà thư viện Surprise cung cấp khá nhiều, nên phải xem xét tính chất của ma trận đầu vào sau khi tiền xử lý dữ liệu như kích thước và độ thưa của ma trận, từ đó chọn được thuật toán phù hợp nhất, tránh lãng phí tài nguyên mà hiệu suất lại không cao.

Mục tiêu của việc tạo ra thư viện Surprise là tối ưu hóa những việc làm từ xử lý dữ liệu để chuẩn bị cho mô hình đầu vào cho đến việc triển khai các thuật toán mà nó cung cấp, tính toán để đưa ra kết quả mong muốn mà vẫn đảm bảo được tính chính xác và hiệu suất ổn định. Với thuật toán đa dạng, lập trình viên có thể đánh giá một cách khách quan xem đâu là thuật toán phù hợp nhất với tập dữ liệu của dự án, từ đó tiến hành tinh chỉnh tập dữ liệu hoặc chọn thuật toán khác để tối ưu hóa hệ thống đề xuất.

Thư viện Surprise được phát triển với khả năng tinh chỉnh tham số và cấu hình liên quan, từ đó tạo ra sự đa dạng để thích ứng với nhiều dạng mô hình với những bộ dữ liệu khác nhau. Từ việc cung cấp nhiều thuật toán hỗ trợ cho hệ thống đề xuất; việc điều chỉnh các tham số đầu vào(k trong KNN; n_factors (số lượng yếu tố tiềm ẩn), n_epochs (số lượng vòng lặp), lr_all (tốc độ học), và reg_all (hệ số điều tiết) trong SVD); sử dụng cross-validation để đánh giá hiệu suất của mô hình với nhiều tham số khác nhau; sử dụng GridSearchCV để tìm kiếm nhằm tìm ra bộ tham số phù hợp với mô hình;... Những ví dụ nêu trên chỉ là một phần nhỏ về lợi ích của thư viện này mang lại. Với Surprise, việc tinh chỉnh mô hình được cải thiện đáng kể, tăng độ chính xác của các dự đoán và từ đó có cái nhìn toàn diện về cách các mô hình hoạt động với bộ dữ liệu của dự án.

Thư viện Surprise phát triển tính năng kiểm định chéo Cross-Validation để đánh giá hiệu suất của mô hình với phương thức cross_validate. Trong cross-validation, dữ liệu thường được chia làm nhiều phần và kiểm định chéo cho nhau. Mô hình được chia làm n phần sẽ được huấn luyện n lần. Mỗi lần sẽ sử dụng một 1 dữ liệu để kiểm tra, và n-1 phần còn lại sẽ làm tập huấn luyện. Từ đó tính ra được độ chính xác RMSE (Root Mean Square Error), MAE (Mean Absolute Error) để từ đó biết được mô hình của dự án có đáng tin cậy hay không, sau đó tiến hành tinh chỉnh và tiền xử lý dữ liệu.

Thư viện Surprise cũng tích hợp GridSearchCV. Về cách thức hoạt động, đầu tiên cần xác định trước một tập các giá trị tham số và thực hiện kiểm định chéo để tìm ra bộ tham số tối ưu. Trong ví dụ nêu trên với mô hình SVD, với số lượng yếu tố tiềm ẩn, xác định một tập tham số {50, 80, 120}; với tốc độ học {0.005, 0.007, 0.009}; hệ số điều tiết {0.02, 0.04}. Sau đó, kỹ thuật này sẽ tiến hành kiểm định chéo bằng cách tổ hợp mỗi giá trị của một thuộc tính và từ đó tính ra độ chính xác RMSE hoặc MAE.

Đánh giá mô hình được Surprise hỗ trợ rất hiệu quả. Thư viện cung cấp các phương pháp đánh giá như cross-validation, A/B testing giúp đa dạng hóa trong việc đánh giá mô hình. Việc đánh giá theo nhiều cách khác nhau làm tăng độ tin cậy của mô hình, từ đó tăng sự chính xác của việc dự đoán. Việc tối ưu hóa tham số đối với các phương pháp

là khác nhau dựa trên các mô hình khác nhau. Tuy nhiên cũng nên kiểm tra theo nhiều phương pháp để tự đó đạt được kết quả khách quan hơn.

Với việc kết hợp giữa sự linh hoạt và nhu cầu trong chủ đề đề xuất, Surprise đã trở thành công cụ gần như khó thay thế trong thị trường hiện nay. Những nhà nghiên cứu và các kỹ sư phần mềm vẫn đang tận dụng thư viện này trong quá trình nghiên cứu và xây dựng ứng dụng. Việc ứng dụng thư viện này giúp đơn giản hóa quá trình phát triển và tăng khả năng mở rộng với đa dạng công cụ mà nó cung cấp. Hiện nay, thư viện này vẫn đang được nhà cung cấp phát triển và đưa ra phiên bản mới hơn.

So sánh thuật toán KNN (K-Nearest Neighbors), SVD (Singular Value Decomposition), NMF (Non-negative matrix factorization)

Bảng 3 So sánh KNN, SVD, NMF

	KNN	SVD	NMF
Đặc điểm tập dữ liệu	Tập dữ liệu nhỏ đến trung bình Dữ liệu có ít chiều	Tập dữ liệu lớn và thưa	Dữ liệu thường là văn bản, xử lý ảnh, biểu đồ
Thời gian thực thi	Chậm nếu tập dữ liệu càng lớn Không cần tạo ra mô hình từ dữ liệu	Nhanh	Phụ thuộc vào kích thước tập dữ liệu, chậm hơn SVD
Tài nguyên tiêu thụ	Tăng nếu tập dữ liệu càng lớn	Nhiều do tính toán phức tạp	Nhiều vì có quá trình phân rã ma trận

Vì dữ liệu của dự án chỉ ở mức nhỏ (dưới 400 dòng), tài nguyên có hạn, tập dữ liệu không có tính chất thưa nên sử dụng thuật toán KNN vào hệ thống đề xuất

So sánh thuật toán KNNBasic, KNNWithMeans, KNNWithZScore.

Bảng 4 So sánh KNNBasic, KNNWithMeans, KNNWithZScore

	KNNBasic	KNNWithMeans	KNNWithZScore

Dữ liệu	Người dùng trong tập dữ liệu có xu hướng đánh giá tương tự nhau	Có sự chênh lệch trong xu hướng đánh giá: giá trị đánh giá là một khoảng	Có sự chênh lệch trong xu hướng đánh giá: giá trị đánh giá là một giá trị cụ thể
---------	---	--	--

Ví dụ chung: Tập dữ liệu nhỏ về đánh giá cà phê

Sử dụng KNNBasic: Đa số người dùng đánh giá cà phê từ 1 đến 5 sao. Phần lớn người dùng có xu hướng đánh giá giống nhau. Nếu thích sản phẩm, người dùng đánh giá 4-5 sao, không thích sản phẩm thì đánh giá 1-2 sao.

Sử dụng KNNWithMeans: Một phần người dùng thường xuyên đánh giá 4-5 sao cho dù sản phẩm có tệ hay không. Một phần người dùng lại khắt khe hơn khi chỉ đánh giá sản phẩm từ 1-3 sao.

Sử dụng KNNWithZScore: Một phần người dùng chỉ đánh giá 1 sao hoặc 5 sao, trong khi người dùng khác lại đánh giá từ 1 đến 5 sao.

Trong tập dữ liệu của dự án, khách hàng có xu hướng tương đồng khi đánh giá sản phẩm, hầu hết họ đều đánh giá phim mà họ thích 4 hoặc 5 sao và phim họ không thích 1 hoặc 2 sao nên lựa chọn thuật toán KNNBasic.

Thuật toán sử dụng: KNN Basic

Công thức tính giá trị dự đoán:

$$\hat{r}_{ui} = \frac{\sum_{v \in N_i^k(u)} \text{sim}(u, v) \cdot r_{vi}}{\sum_{v \in N_i^k(u)} \text{sim}(u, v)}$$

Hình 9 Công thức KNN basic [17]

Giá trị dự đoán sẽ được tính bằng cách lấy tổng các tích khoảng cách và giá trị của neighbor chia cho tổng k các khoảng cách giữa điểm cần xét và neighbor

1.5. CHATBOT AI VÀ MODEL LLM CỦA OPEN AI

1.5.1. Khái niệm

Chatbot AI: Một chatbot AI (Trí tuệ nhân tạo) là một chương trình máy tính được thiết kế để mô phỏng cuộc trò chuyện với người dùng con người, thông qua văn bản hoặc giọng nói. Chatbot AI thường được sử dụng trong dịch vụ hỗ trợ khách hàng, hệ thống tương tác qua tin nhắn tự động, và nhiều ứng dụng khác yêu cầu giao tiếp tự nhiên.

Model LLM: LLM là viết tắt của "Large Language Model," tức là mô hình ngôn ngữ lớn. Đây là các mô hình trí tuệ nhân tạo chuyên sâu, được đào tạo trên lượng lớn dữ liệu văn bản để hiểu và tạo ra ngôn ngữ tự nhiên. Các LLM có khả năng thực hiện nhiều nhiệm vụ ngôn ngữ khác nhau như dịch thuật, tóm tắt văn bản, trả lời câu hỏi và thậm chí sáng tạo văn bản.

OpenAI: OpenAI là một tổ chức nghiên cứu trí tuệ nhân tạo với mục tiêu phát triển và thúc đẩy AI theo hướng có lợi cho cả nhân loại. Họ phát triển nhiều công nghệ AI tiên tiến, bao gồm cả các mô hình ngôn ngữ lớn. OpenAI nổi tiếng với việc tạo ra GPT (Generative Pretrained Transformer) series, mà GPT-3 là một trong những mô hình ngôn ngữ lớn được biết đến rộng rãi.

Khi kết hợp các khái niệm này, chatbot AI sử dụng model LLM của Open AI mô tả một chatbot sử dụng một mô hình ngôn ngữ lớn phát triển bởi OpenAI để xử lý và tương tác với người dùng một cách tự nhiên và thông minh. Các chatbot như vậy có thể hiểu rõ các yêu cầu của người dùng và cung cấp các phản hồi phù hợp, đôi khi không thể phân biệt được so với một người thật.

1.5.2. Độ phù hợp model LLM của OPEN AI đối với dự án

Nhóm đã tiến hành thử nghiệm việc sử dụng model LLM của Open AI và Hugging Face. Ở đây, nhóm so sánh giữa model LLM của gpt-3.5-turbo-0125 và model LLM ggml-vistral-7B-chat-q4_0.gguf của Hugging face. Đây là kết quả:

Bảng 5 So sánh model LLM giữa Open AI và Hugging Face

	Model LLM Hugging Face	Model LLM Open AI
--	------------------------	-------------------

Tốc độ chạy	Chậm	Nhanh, nhưng đôi khi thời gian phản hồi chậm nếu máy chủ gặp tải cao
Độ chính xác	<ul style="list-style-type: none"> Với 7 tỷ tham số, độ chính xác mà model này mang lại chưa cao. Sử dụng sai câu lệnh sql để truy vấn dữ liệu. 	Với 175 tỷ tham số, độ chính xác cao.
Tài nguyên	<ul style="list-style-type: none"> Kích thước Model lớn, tốn tài nguyên (GPU) nên chỉ dùng cho mục đích thử nghiệm Tốn tài nguyên khi deploy 	<ul style="list-style-type: none"> Vì sử dụng API của Open AI nên không tốn tài nguyên cho việc tải model. Tốn phí hàng tháng: 20\$/tháng

Vì những ưu điểm mà API của Open AI mang lại, nhóm quyết định sử dụng model LLM của Open AI.

1.5.3. Thư viện sử dụng

Sử dụng Model LLM của Open AI bằng cách sử dụng API của OpenAI

Thư viện python được sử dụng: LangChain

Langchain

LangChain là một thư viện python được phát triển với mã nguồn mở, được sử dụng khi phát triển các ứng dụng có liên quan tới mô hình ngôn ngữ lớn LLM. Đây là mô hình học sâu với dữ liệu đầu vào được lấy từ nhiều nguồn khác nhau. Hiện tại có nhiều tổ chức cung cấp mô hình LLM này như OpenAI, HuggingFace. Mô hình có khối lượng dữ liệu càng lớn thì độ chính xác của câu trả lời sẽ càng cao. Hơn nữa, thư viện LangChain còn cung cấp nhiều yếu tố khác để điều chỉnh độ chính xác, những tính chất của câu trả lời. Ngoài ra, thư viện này cũng được phát triển cho phép LLM có thể tiếp xúc và học những tập dữ liệu mới mà không cần phải đào lại toàn bộ dữ liệu.

Mô hình ngôn ngữ lớn LLM phản hồi các câu hỏi trong bối cảnh chung khá chính xác, phù thuộc vào tập dữ liệu mà đã được đào tạo. Tuy nhiên, trong một lĩnh vực cụ thể, nếu mô hình chưa được đào tạo thì hầu như không thể đưa ra đáp án chính xác. Ví dụ như với giá cà phê được bán trên thế giới, dữ liệu đã được công khai trên internet thì LLM đã được đào tạo với tập dữ liệu này, tuy nhiên khi ứng dụng mô hình LLM vào hệ thống của bạn, nó không thể hiểu tập dữ liệu của hệ thống nếu chưa được đào tạo. Vì thế những câu hỏi của người dùng như hỏi giá cà phê trong hệ thống của bạn, LLM sẽ đưa ra những phản hồi không chính xác. Prompts được sử dụng trong trường hợp này, nó đóng vai trò như là câu truy vấn để LLM dựa vào đó đưa ra câu trả lời chính xác nhất.

Để có thể thực hiện việc này, nhà phát triển ứng dụng phải tích hợp mô hình LLM với dữ liệu nội bộ của hệ thống. Việc tinh chỉnh dữ liệu đầu vào là một bước bắt buộc để LLM có thể hiểu và phản hồi chính xác câu trả lời đến người dùng.

Khi sử dụng LangChain, ta có thể tái sử dụng các Model từ nhiều nguồn khác nhau mà không cần phải đào tạo hay tinh chỉnh bất cứ gì. Với những thông tin mà nhà phát triển cung cấp, ta có thể sử dụng nó như những API, phát triển các hệ thống khác nhau trên hệ thống như chatbot, gợi ý tìm kiếm, tạo nội dung. Việc chia sẻ dữ liệu nội bộ và sử dụng các model do bên thứ 3 cung cấp cũng sẽ được bảo mật. LangChain cung cấp các phương thức khác nhau để tạo ra các câu truy vấn, từ đó lấy được nội dung câu trả lời và gửi chúng cho mô hình LLM.

Trong đề tài này, thay vì phát triển một AI, tiêu tốn thời gian training cho mỗi lần cập nhật tập dữ liệu mới, việc tích hợp nguồn dữ liệu với bên thứ 3 (model LLM của OpenAI) và điều chỉnh những prompt sẽ giảm độ phức tạp và tăng tốc độ phát triển của dự án. Prompts càng chi tiết thì độ chính xác phản hồi càng cao. Nhà phát triển ứng dụng chỉ cần sửa những mẫu do LangChain cung cấp, cấu hình tích hợp với tập dữ liệu của hệ thống thì Chatbot AI sẽ được hoàn thành.

LangChain đóng vai trò như cầu nối giữa nhà phát triển ứng dụng và nhà cung cấp các dịch vụ AI trong việc kết nối tập dữ liệu nội bộ và mô hình LLM của họ. Vì được phát triển theo hướng mã nguồn mở, LangChain trở thành thư viện phổ biến trong lĩnh vực AI, người dùng dễ dàng tiếp cận đến thư viện này.

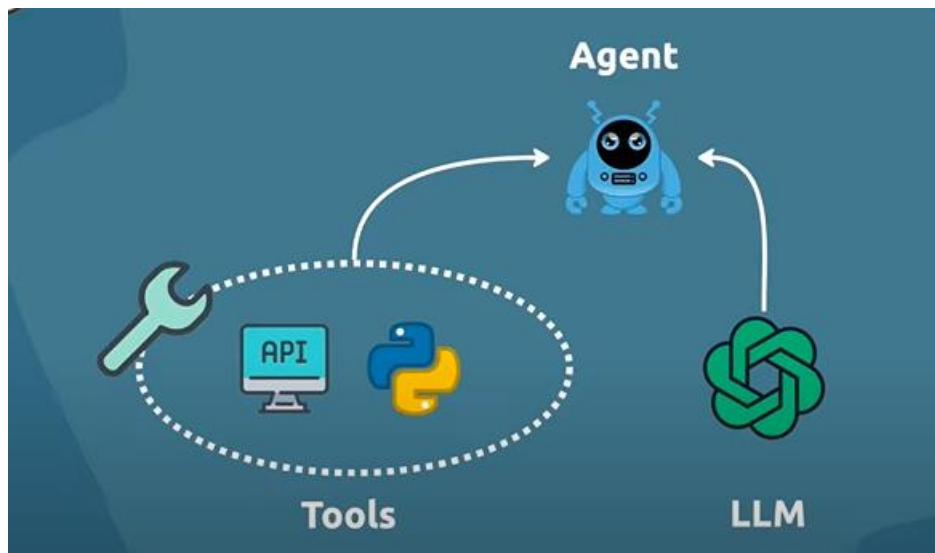
Cách hoạt động của LangChain:

Sử dụng LangChain giúp các nhà phát triển ứng dụng có thể điều chỉnh mô hình LLM phù hợp với bối cảnh của dự án. Chain là tập hợp các thành phần AI để có thể nhận biết ngữ cảnh hiện tại (câu hỏi của người dùng) và từ đó đưa ra câu trả lời phù hợp. Cũng có thể hiểu Chain là các hoạt động đã được phát triển một cách tự động, bắt đầu từ việc hiểu ngữ cảnh, tạo ra các câu truy vấn và sử dụng kết quả để tạo ra câu trả lời thích hợp. Tiến trình này được các nhà phát triển ứng dụng rỗng rã như dùng để kết nối với các nguồn dữ liệu khác, giúp tạo nội dung cho các bài viết từ một yêu cầu đơn giản, hoặc hay hơn là hiểu câu hỏi theo nhiều ngôn ngữ khác nhau, và quan trọng nhất là hiểu được yêu cầu của người dùng. Một tập hợp Chain xảy ra theo trình tự sẽ tạo ra một Links. Như đã nói ở trên, Link sẽ bắt đầu từ lúc người dùng nhập câu hỏi, thông qua các quá trình hiểu và xử lý, đưa ra câu trả lời theo mô hình LLM.

Các thành phần cốt lõi của LangChain:

LangChain cung cấp các API để nhà phát triển kết nối và truy vấn mô hình LLM. Họ có thể giao tiếp với các nhà cung cấp dịch vụ LLM như GPT, Bard, PaLM thông qua thư viện LangChain bằng các lệnh gọi API đơn giản, thay vì phải cấu hình phức tạp.

Một Prompt template là một mẫu đã được tạo sẵn, tích hợp bên trong LangChain. Nó có tác dụng định dạng câu hỏi truy vấn, giúp đồng bộ dữ liệu, từ đó hệ thống AI đưa ra các câu trả lời có tính nhất quán và chính xác. Template này được sử dụng nhiều trong trường hợp làm chatbot, hệ thống lúc này có ít dữ liệu, nên việc viết những hướng dẫn cụ thể cũng không mất nhiều thời gian và làm tăng độ tin cậy của câu trả lời. Trong nhiều trường hợp, prompt template này có thể được tái sử dụng với nhiều loại model khác và ứng dụng khác nhau.



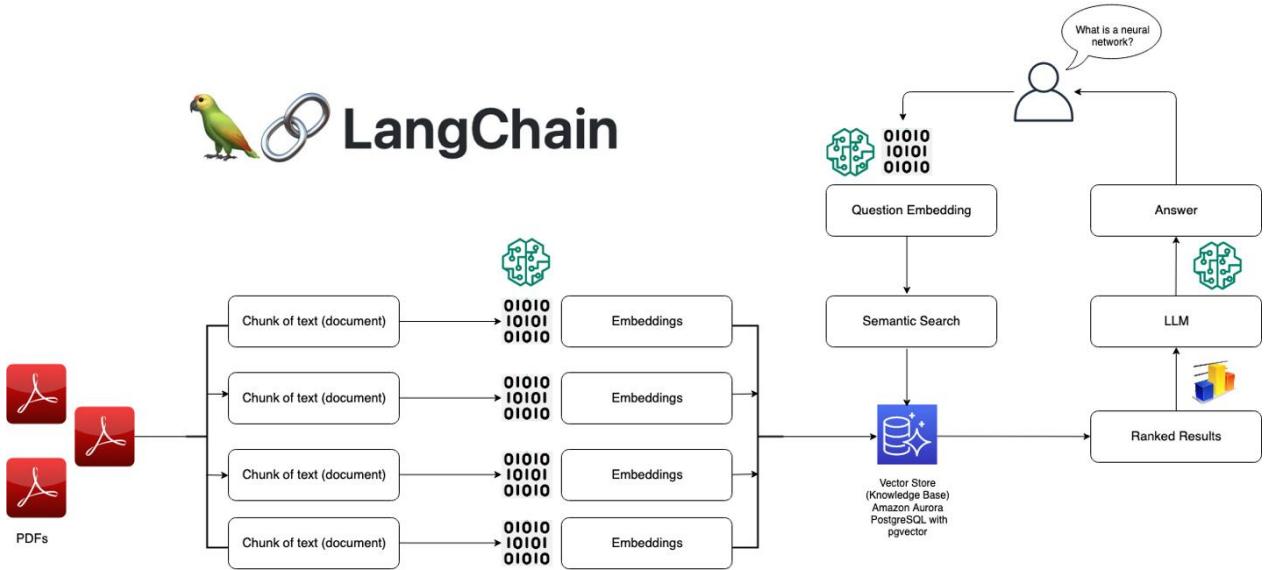
Hình 10 Các thành phần của Agent [18]

Agent được tạo ra từ mô hình LLM và các tool trong LangChain, bao gồm API kết nối với mô hình được cung cấp từ bên thứ 3, một prompt template do nhà phát triển tự định nghĩa, và các tool khác để đọc hoặc truy xuất dữ liệu để xử lý những yêu cầu khác nhau.

Một số ứng dụng thường xuyên yêu cầu việc tạo câu trả lời từ những tương tác (câu hỏi - trả lời) trước kia của người dùng và hệ thống. LangChain cung cấp một bộ nhớ để có thể lưu những tương tác đó theo thời gian thực. Thư viện này sẽ sử dụng những cấu trúc phức tạp để lưu những cuộc trò chuyện này và phân tích chúng để đảm bảo tính nhất quán giữa các câu trả lời.

Trong một số trường hợp câu trả lời không được tạo ra hoặc theo một cấu trúc không mong muốn, cơ chế agent sẽ giúp lặp lại Links thêm một vài lần (3 – 4) để tạo ra những câu trả lời tiếp theo, tránh trường hợp người dùng nhận câu trả lời khó hiểu.

Nguyên tắc hoạt động:



Hình 11 Sơ đồ hoạt động của Chatbot AI [19]

Cách tạo ra mô hình LLM:

Bước 1: Thu tập dữ liệu

Quá trình phát triển bắt đầu bằng việc thu thập dữ liệu văn bản từ mọi nguồn, từ sách, báo, tạp chí, internet, mạng xã hội,... để làm đa dạng hơn cho tập dữ liệu. Từ đó, ngôn ngữ tự nhiên được hiểu rõ nghĩa hơn trong nhiều ngữ cảnh khác nhau.

Bước 2: Tiền xử lý dữ liệu

Ở bước này, dữ liệu được xử lý qua các bước sau:

- Làm sạch: dữ liệu sẽ được loại bỏ lỗi, có thể là những từ sai chính tả, hoặc không nhất quán chữ viết với ngôn ngữ liên quan, hoặc là những ký tự lạ, sau đó sẽ loại bỏ những dấu câu, những từ không mang ý nghĩa.
- Chuẩn hóa: giúp dữ liệu phải nhất quán về câu từ. Mọi dữ liệu trong tập phải được đưa về dạng chuẩn nhất định, thông thường sẽ được đưa về dạng thường (chữ cái latin). Có 2 kỹ thuật trong việc biến đổi từ: stemming và lemmatization. 2 kỹ thuật này là kỹ thuật tiêu biểu được sử dụng khi thao tác với ngôn ngữ tự

nhiên NPL, giúp giảm đa dạng từ mà vẫn giữ được ý nghĩa cốt lõi của từ, từ đó nâng cao hiệu suất đọc và hiểu tài liệu của mô hình

Stemming: kỹ thuật này sẽ cắt tiền tố, hậu tố của từ input và đưa nó về dạng gốc của từ. Ví dụ: play - plays - playing sẽ được chuyển về chữ play, better sẽ chuyển thành chữ better. Tuy nhiên, phương pháp này sẽ không hiểu được hoàn toàn ý nghĩa của một câu vì những tiền tố hay hậu tố của một từ sẽ mang một ý nghĩa khác nhau khi đứng tổ hợp với những từ khác.

Lemmatization: kỹ thuật này sẽ chuyển một từ thành từ cơ bản nhất theo ngữ pháp và từ điển nhưng vẫn hiểu được ý nghĩa của từ và ngữ cảnh xung quanh nó. Ví dụ: play - plays - playing sẽ được chuyển về chữ play, better sẽ chuyển thành chữ good. Trong trường hợp này, kỹ thuật lemmatization phải tìm hiểu rõ mối liên hệ giữa từ better và từ good, xem từ good có nghĩa trong câu đang xét hay không. Việc này tiêu tốn nhiều tài nguyên và thời gian hơn, nhưng độ chính xác của việc hiểu ngữ cảnh sẽ tăng.

- Tokenization sau quá trình chuyển đổi từ về dạng chuẩn, câu văn sẽ được token hóa, nghĩa là tách ra thành các thành phần nhỏ, mỗi thành phần có thể là một từ hoặc một cụm từ hoặc là một câu hoặc là tài liệu đang đọc. Việc này giúp giảm kích thước của từ điển và giảm tỉ lệ xuất hiện những từ mà mô hình không biết.

Ví dụ cho bước tiền xử lý dữ liệu:

Với dữ liệu đầu vào là một câu văn “Loại trái cây nào chứa vitamin A?”

Câu văn này sẽ được xử lý thông qua các bước sau:

- Loại bỏ dấu câu: " Loại trái cây nào chứa vitamin A"
- Chuyển đổi thành chữ thường: "loại trái cây nào chứa vitamin a"
- Tokenization: ["loại", "trái", "cây", "nào", "chứa", "vitamin", "a"]

Bước 3: Embeddings

Embeddings là quá trình chuyển dữ liệu từ dạng văn bản thành dạng vector, là kiểu dữ liệu đầu vào cho mô hình LLM. Có 2 phương pháp trong quá trình embeddings: word embeddings - Sentence/Document Embeddings.

Word embedding là một phương pháp mà đầu ra của nó là vector đa chiều dùng để biểu diễn từ trong một câu, dữ liệu sẽ được lấy từ bước tokenization đã làm ở trên. Với mỗi từ là một vector đa chiều, vì thế ý nghĩa và ngữ pháp của từng từ sẽ được biểu diễn rõ qua chiều của các vector. Kỹ thuật này sẽ xem xét những từ gần nghĩa với từ đang xét, từ đó rút ra được mối quan hệ giữa chúng. Có 3 kỹ thuật liên quan tới phương pháp này: Word2Vec, GloVe, và FastText

- Work2Vec tập trung vào việc xét ý nghĩa những từ mà khi kết hợp với từ đang xét sẽ tạo được 1 câu có ý nghĩa hoàn chỉnh, đúng theo ngữ cảnh thực tế. Ví dụ, với từ apple, mô hình sẽ dự đoán các từ xung quanh là từ "I", "eat", "an", từ 4 từ này, vector tương ứng với từ apple sẽ là "apple": [0.13, -0.24, 0.58, 0.69, -0.45, ...]
- Glove tập trung vào việc sử dụng thông tin trong ma trận tập dữ liệu đã được chuẩn bị trước đó. Dựa vào ma trận này, mô hình sẽ có cái nhìn toàn cục về ý nghĩa của từ so với các từ trong từ điển. Ví dụ, khi xét từ apple, dựa vào ma trận tập dữ liệu, mô hình sẽ tính ra mức độ liên quan từ các từ trong từ điển. Từ thường xuyên hay xuất hiện với từ apple là từ "fruit", trong khi đó, từ "car" lại ít khi hoặc không xuất hiện với từ apple. Từ những dữ liệu thu được, vector sẽ được chuyển thành "apple": [0.81, -0.61, 0.12, -0.33, 0.67, ...]
- FastText phức tạp hơn Work2Vec và Glove. Với một từ, Fasttext sẽ tiến hành tạo ra các tổ hợp ký tự từ trái sang phải, sau đó tính vector biểu diễn cho từng tổ hợp. Sau đó tính ra vector trung bình. Với từ apple, ta có tổ hợp ký tự là "ap", "ppl", "ple". Sau đó sẽ tính vector cho các tổ hợp này. Với 3 vector đa chiều mà ta có được, tiến hành tính vector apple có giá trị là vector trung bình của 3 vector "apple": [0.23, -0.17, 0.88, 0.45, -0.76, ...].

Sentence/Document Embeddings cũng là một phương pháp với đầu ra cũng là một vector đa chiều. Nhưng thay vì là mỗi từ là một vector, thì kỹ thuật này sẽ đưa 1

câu hoặc là toàn bộ tài liệu thành một vector đa chiều. Với kỹ thuật này, mô hình sẽ hiểu rõ hơn ý nghĩa tổng thể của câu chữ hay tài liệu vì phải xử lý rất nhiều thông tin trong câu/ tài liệu, về cấu trúc, ngữ pháp, ý nghĩa ngữ cảnh. Có 2 kỹ thuật ứng với phương pháp này: BERT, GPT

- BERT: hiểu nghĩa của từng từ trong câu bằng cách dự đoán từ bị ẩn đi dựa trên ngữ cảnh của các từ xung quanh. Sau khi tạo ra câu hoàn chỉnh sẽ tiến hành tính vector cho câu này.
- GPT: hiểu nghĩa và ngữ cảnh của từng từ và từ đó tạo ra văn bản từ ngữ cảnh đó. Sau khi tạo ra câu hoàn chỉnh sẽ tiến hành tính vector cho câu này.

Sự khác biệt giữa Word embeddings và Sentence/Document Embeddings:

Bảng 6 So sánh Word Embeddings và Sentence/ Document Embeddings

	Word emdding	Sentence/Document Embeddings
Mức độ	Từ	Câu/ Tài liệu
Ngữ cảnh	Ý nghĩa, ngữ pháp của 1 từ	Ý nghĩa, ngữ pháp của cả câu/ tài liệu
Trường hợp sử dụng	Phân loại từ, dịch từ tự động, đánh giá cảm xúc của từ	Phân loại văn bản, tóm tắt, tạo tự động văn bản

Bước 4: Xây dựng mô hình và học sâu:

Đầu tiên, cần xây dựng một mô hình sơ khai, ví dụ là một mạng no-ron như Transformer. No-ron có thể hiểu là một hàm toán học, nó có tác dụng biến đổi một vector đa chiều thành một vector đa chiều khác. Một mạng no-ron bao gồm nhiều lớp no-ron, một lớp no-ron bao nhiêu nhiều no-ron. Giữa các lớp no-ron có các trọng số. Trọng số này có thể là một ma trận, hoặc là một số. Ví dụ một lớp có 3 no-ron và lớp tiếp theo có 2 no-ron, ma trận trọng số giữa hai lớp này sẽ có kích thước 3×2 .

Tiếp theo, mô hình sẽ huấn luyện trên tập dữ liệu lớn. Mỗi lần huấn luyện sẽ là điều chỉnh trọng số giữa các lớp no-ron để từ dữ liệu đầu vào tạo ra được vector đầu ra mong muốn. Cấu trúc của một no-ron như đã nói ở trên bao gồm 3 phần:

- Đầu vào input (có thể là một vector)

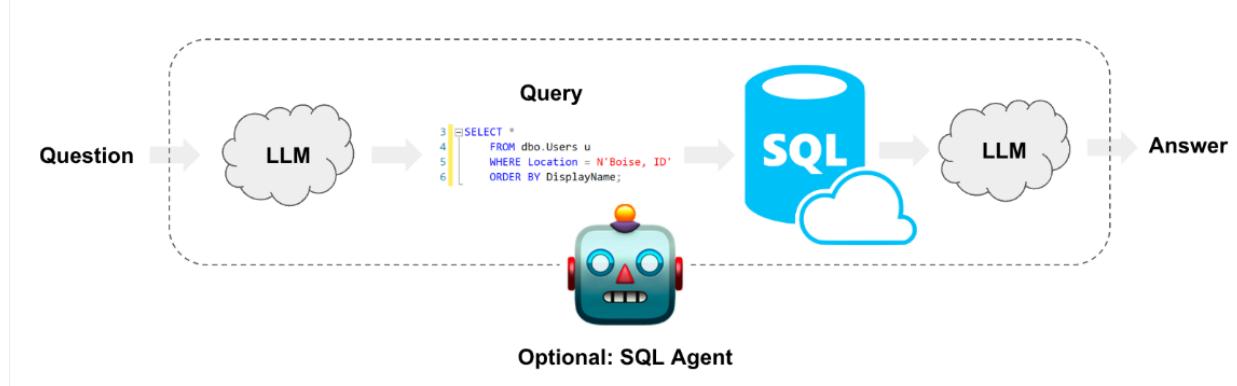
- Tổng trọng số Weighted Sum: nhân vector này với trọng số
- Hàm kích hoạt (Activation Function): quyết định xem có nên gửi tín hiệu cho một nơ-ron khác hay không. Trong trường hợp không gửi tín hiệu thì kết quả sẽ có sai sót, từ đó điều chỉnh lại trọng số để không xảy ra tình trạng nghẽn tính hiệu tại các nơ-ron.

Ví dụ: với vector "apple": [0.13, -0.24, 0.58, 0.69, -0.45, ...]. Khi qua mỗi lớp nơ-ron, vector sẽ được biến đổi thành 1 vector khác, đầu ra của một hình là vector biểu diễn từ tiếp theo trong câu. Sau đó từ vector này sẽ tạo ra một token và ánh xạ token này sẽ tạo ra 1 từ ngữ tự nhiên. Ví dụ với tập kiểm thử, sau từ apple là từ pie, cake,... nhưng nếu vector output (hoặc là token) biểu diễn từ car, bike thì phải điều chỉnh trọng số giữa các lớp nơ-ron.

Tiếp theo sẽ tiến hành fine-tuning mô hình. Sau khi tiến hành huấn luyện trên một tập dữ liệu, tiến hành kiểm thử trên một tập dữ liệu khác với các nhiệm vụ khác nhau như tạo văn bản, phân loại từ. Với những kiểu tập dữ liệu khác nhau, ngữ cảnh mà mô hình có thể hiểu được cũng ngày càng đa dạng. Với mỗi tập dữ liệu, nhà phát triển mô hình sẽ tiếp tục cẩn chỉnh trọng số để dần hoàn thiện mô hình.

Tiếp theo sẽ tiến hành đánh giá và tối ưu hóa lại mô hình để cải thiện độ chính xác trong việc dự đoán thông qua điều chỉnh các kiến trúc, hoặc tăng mật độ huấn luyện, mở rộng với những tập dữ liệu đầu vào khác nhau.

Cách tạo ra phản hồi khi sử dụng SQL Agent Chatbot AI:



Hình 12 SQL Agent [20]

- Câu hỏi sau khi được nhập bởi người dùng sẽ được gửi đến Model LLM của Open AI thông qua API của thư viện LangChain.
- Câu hỏi sẽ được tiền xử lý, thông qua hệ thống NLP sẽ biến đổi câu hỏi thành tập hợp các vector đa chiều.
- Vector đa chiều sẽ được biến đổi thông qua Model LLM của Open AI để tạo ra câu truy vấn SQL.
- Tiếp theo, câu truy vấn sẽ được trả về thông qua API, LangChain sử dụng thư viện để thực thi câu truy vấn này dưới database để lấy kết quả.
- Sau khi có kết quả thực thi câu lệnh SQL, kết quả sẽ được gửi lại Model LLM qua API mà LangChain cung cấp để thực hiện chuyển đổi sang ngôn ngữ tự nhiên.
- Sau khi có phản hồi, câu trả lời sẽ được gửi về hệ thống.

Kỹ thuật Embeddings sử dụng: OpenAiEmbedding

Nơi lưu trữ vector sau khi Embedding: Chroma

Thư viện truy vấn cơ sở dữ liệu: SQLDatabaseToolkit.

Chương 2: YÊU CẦU NGƯỜI DÙNG

2.1. YÊU CẦU NGHIỆP VỤ

Dự án phần mềm ở đây là website kinh doanh cà phê.

Người sử dụng phần mềm: quản lý website, nhân viên , khách hàng, khách vãng lai

Mục đích của dự án xây dựng website cung cấp các chức năng của website thương mại điện tử, thiết kế trang web dễ sử dụng có nhiều tính năng linh hoạt như:

Về phía khách hàng:

- Dễ dàng tìm kiếm thông tin, tham khảo các sản phẩm
- Đặt hàng online, dễ dàng thanh toán chi phí.

Về phía nhân viên:

- Quản lý các đơn hàng của khách
- Báo cáo lợi nhuận cho quản lý.

Về phía quản trị viên:

- Quản lý tài khoản của hệ thống
- Tất cả các tính năng nhanh chóng và chính xác

2.2. CÁC PHƯƠNG PHÁP TIẾP CẬN

Tìm hiểu, khảo sát quy trình quản lý sản phẩm trong cửa hàng tương tự trên internet.

Tìm hiểu những nhu cầu của khách hàng dựa vào đó để xây dựng phần mềm.

Thu nhập thông tin từ các dịch vụ du lịch tương tự và liên quan.

Đánh giá, kiểm tra, cho ra mắt (quảng cáo) hướng dẫn người dùng sử dụng dịch vụ.

2.3. SẢN PHẨM BÀN GIAO CUỐI

Website bán cà phê có các chức năng chính sau:

- Phía khách hàng:
 - Đăng ký, đăng nhập vào hệ thống
 - Tìm kiếm thông tin cà phê theo nhu cầu.
 - Xem thông tin chi tiết của cà phê được hiển thị trên trang web.
 - Thực hiện việc đặt hàng, thanh toán trực tuyến.
 - Quản lý thông tin, tài khoản cá nhân.
 - Xem được danh sách sản phẩm được đề xuất.
 - Trò chuyện cùng Chatbot AI.
- Phía nhân viên:
 - Bao gồm các chức năng của khách hàng
 - Quản lý các đơn đặt hàng, doanh thu.
 - Quản lý các thông tin liên quan tới cà phê.

Chương 3: PHÂN TÍCH HỆ THỐNG

3.1. XÁC ĐỊNH VAI TRÒ ACTOR TRONG LUẬC ĐỒ USE CASE

Bảng 7 Vai trò actors trong lược đồ usecase

Actor	Mô tả
Guess	Khách hàng chưa có tài khoản, khách hàng vãng lai
User	Khách hàng đã có tài khoản
Employee	Nhân viên làm việc tại cửa hàng
Admin	Quản lý cửa hàng

3.2. MÔ TẢ VAI TRÒ NGƯỜI DÙNG

3.2.1. Chi tiết vai trò actor

Bảng 8 Bảng chi tiết vai trò actors

Actor	Quyền
Guess	<ul style="list-style-type: none">Đăng ký tài khoản khách hàngTìm kiếm sản phẩmXem thông tin chi tiết sản phẩmXem danh sách sản phẩm, danh mục tư cách khách hàngQuản lý danh sách sản phẩm yêu thíchQuản lý giỏ hàngTrò chuyện cùng Chatbot AI
User	Kế thừa quyền Guest <ul style="list-style-type: none">Thay đổi mật khẩu khi chưa đăng nhậpXem thông tin chi tiết tài khoản với tư cách khách hàngThay đổi hồ sơ người dùng tư cách khách hàngQuản lý bình luận về sản phẩmThay đổi mật khẩu sau khi đăng nhập tư cách khách hàngĐặt hàngThanh toán bằng cổng thanh toán VnpayThanh toán bằng ví điện tử MomoXem lịch sử đặt hàng

	<ul style="list-style-type: none"> • Hủy đơn hàng • Xem chi tiết đơn hàng • Xem danh sách sản phẩm được đề xuất
Employee	<p>Kế thừa quyền User</p> <ul style="list-style-type: none"> • Quản lý sản phẩm • Quản lý danh mục sản phẩm • Quản lý nhà cung cấp • Quản lý nguồn gốc • Quản lý khách hàng • Quản lý chi tiết sản phẩm • Quản lý hình ảnh sản phẩm • Quản lý hương vị • Quản lý thương hiệu • Quản lý thông tin tài khoản • Xem doanh thu • Thay đổi mật khẩu tài khoản trên trang quản trị
Admin	<p>Kế thừa quyền Employee</p> <ul style="list-style-type: none"> • Quản lý nhân viên • Đăng ký tài khoản nhân viên

3.2.2. Chi tiết chức năng

Bảng 9 Bảng chi tiết chức năng

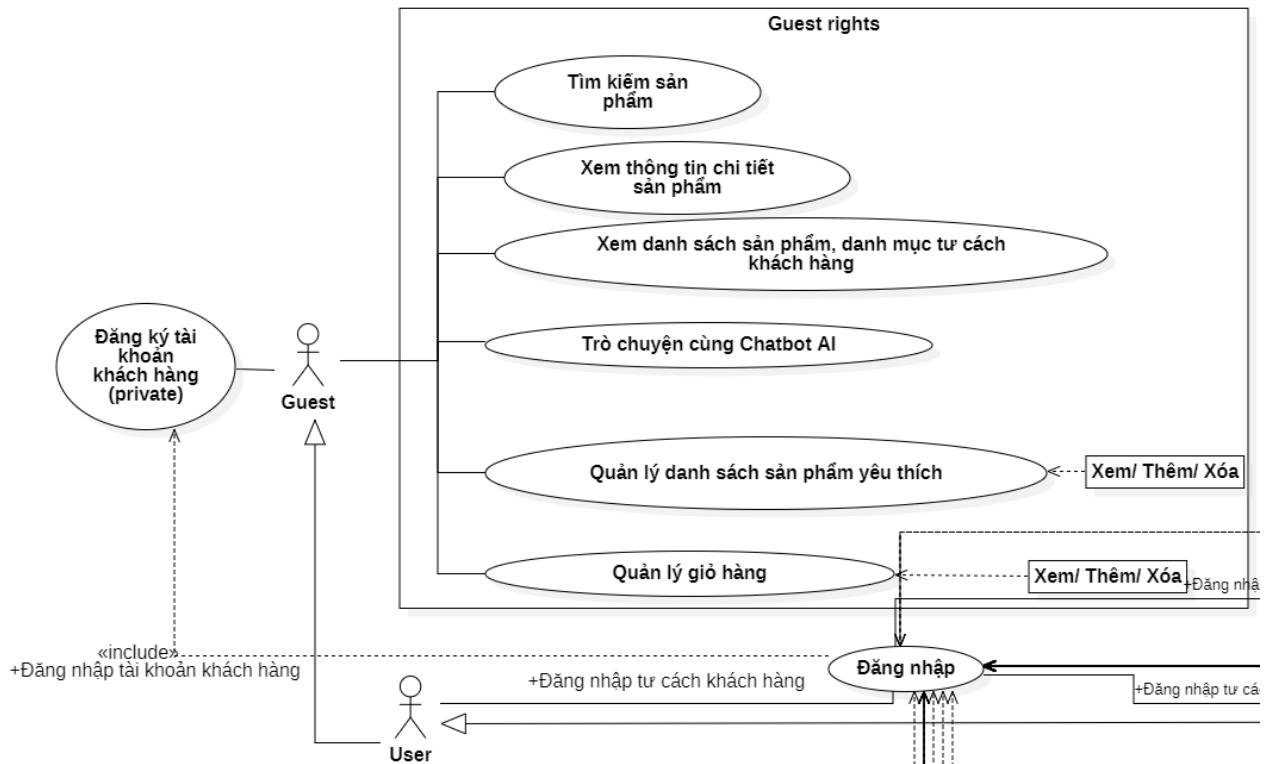
STT	Chức năng	Mô tả
1	Đăng ký tài khoản khách hàng	
2	Tìm kiếm sản phẩm	
3	Xem thông tin chi tiết sản phẩm	
4	Xem danh sách sản phẩm, danh mục tư cách khách hàng	
5	Quản lý danh sách sản phẩm yêu thích	Bao gồm xem, thêm, xóa
6	Quản lý giỏ hàng	Bao gồm xem, thêm, xóa
7	Thay đổi mật khẩu khi chưa đăng nhập	
8	Xem thông tin chi tiết tài khoản với tư cách khách hàng	
9	Thay đổi hồ sơ người dùng tư cách khách hàng	
10	Quản lý bình luận về sản phẩm	Bao gồm bình luận, phản hồi bình luận, chỉnh sửa, xóa
11	Thay đổi mật khẩu sau khi đăng nhập tư cách khách hàng	
12	Quản lý sản phẩm	Bao gồm xem, thêm, cập nhật, xem chi tiết
13	Quản lý danh mục sản phẩm	Bao gồm xem, thêm, cập nhật, xem chi tiết
14	Quản lý nhà cung cấp	Bao gồm xem, thêm, cập nhật, xem chi tiết
15	Quản lý nguồn gốc	Bao gồm xem, thêm, cập nhật, xem chi tiết
16	Quản lý khách hàng	Bao gồm xem, thêm, cập nhật, xem chi tiết
17	Quản lý chi tiết sản phẩm	Bao gồm xem, thêm, cập nhật, xem chi tiết
18	Quản lý hình ảnh sản phẩm	Bao gồm xem, thêm, cập nhật, xem chi tiết
19	Quản lý hương vị	Bao gồm xem, thêm, cập nhật, xem chi tiết
20	Quản lý thương hiệu	Bao gồm xem, thêm, cập nhật, xem chi tiết
21	Quản lý thông tin tài khoản	Bao gồm xem, cập nhật
22	Xem doanh thu	

23	Thay đổi mật khẩu tài khoản trên trang quản trị	
24	Đăng ký tài khoản nhân viên	
25	Quản lý nhân viên	Xem/ Cập nhật/ Xem chi tiết
26	Trò chuyện cùng Chatbot AI	Sử dụng Model LLM của ChatBot AI
27	Xem danh sách sản phẩm được đề xuất	Sử dụng thuật toán KNN

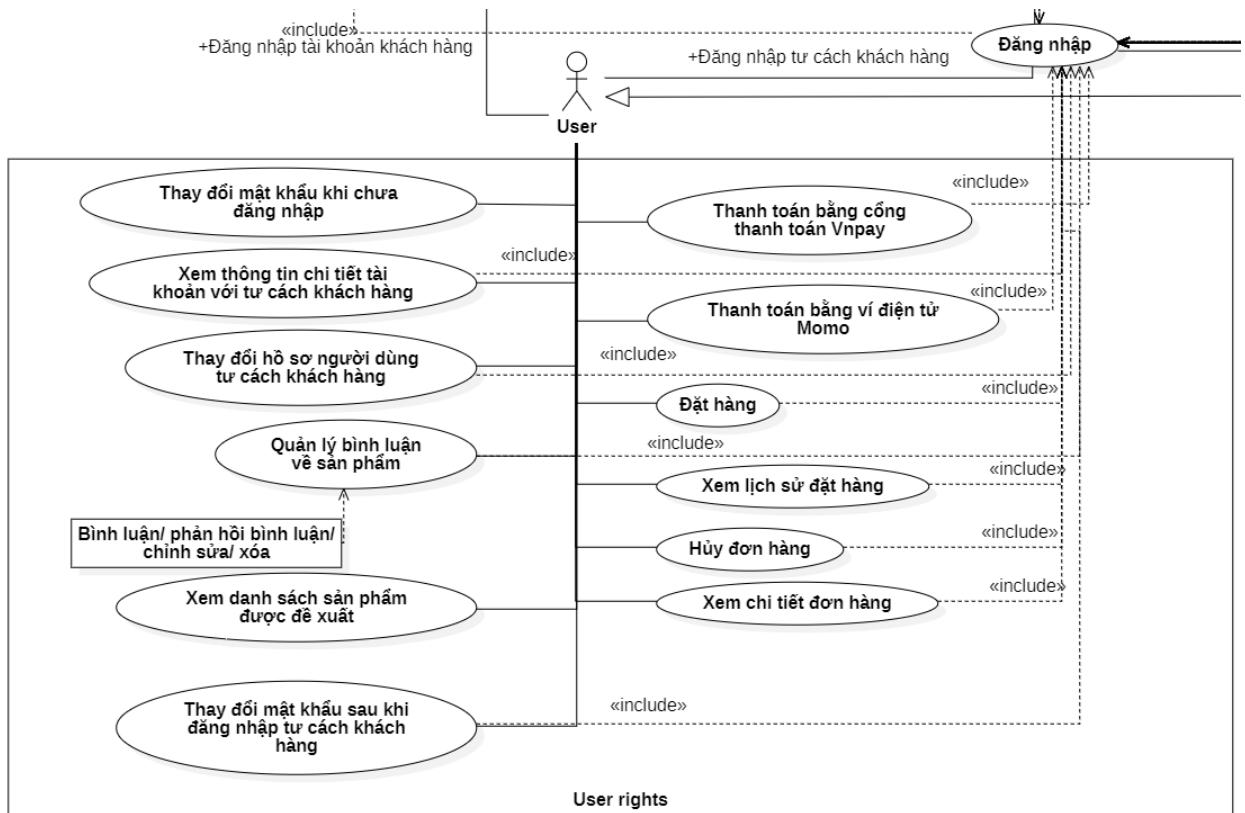
3.3. LUẬC ĐỒ USE CASE



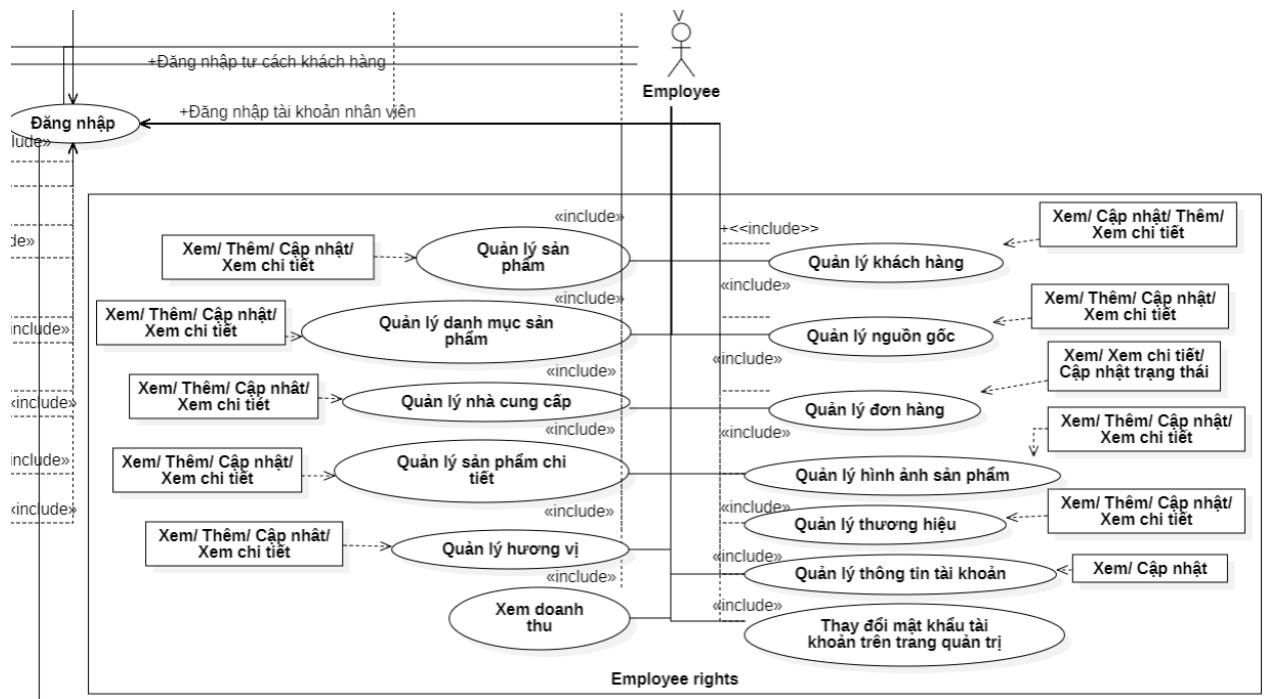
Hình 13 Usecase Diagram – Mối liên quan giữa các tác nhân (Actors)



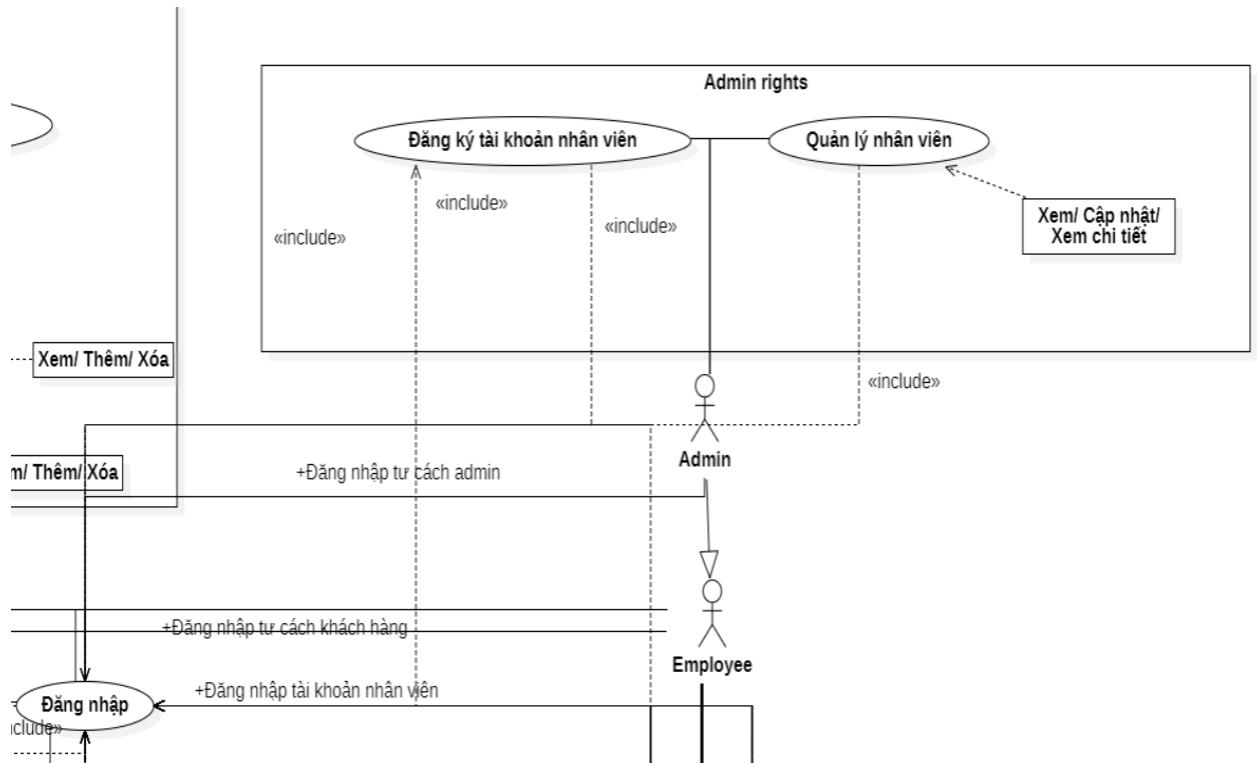
Hình 14 Quyền khách vãng lai - Guest Rights



Hình 15 Quyền khách hàng - User rights



Hình 16 Quyền nhân viên – Employee rights



Hình 17 Quyền Admin – Admin rights

3.4. ĐẶC TẢ USE CASE

3.4.1 Xem giỏ hàng

Bảng 10 Đặc tả usecase xem giỏ hàng

Name	Xem giỏ hàng
Brief Description	Cho phép người dùng xem giỏ hàng.
Actor(s)	User, Employee, Admin, Guest
Pre-Conditions	
Title	Description
Post-Conditions	
Title	Description
Thành công	Người dùng xem giỏ hàng thành công.
Main Flow	
1. Nhấn vào biểu tượng giỏ hàng.	
2. Trả về thông tin của giỏ hàng.	
Alternate Flows	
Title	Description
Exception Flows	
Title	Description

3.4.2 Thêm sản phẩm vào giỏ hàng

Bảng 11 Đặc tả usecase thêm sản phẩm vào giỏ hàng

Name	Thêm sản phẩm vào giỏ hàng
Brief Description	Cho phép người dùng thêm sản phẩm vào giỏ hàng.
Actor(s)	User, Employee, Admin, Guest
Pre-Conditions	
Title	Description
Post-Conditions	
Title	Description

Thành công	Người dùng thêm sản phẩm vào giỏ hàng thành công.
Main Flow	
1. Nhấn "Thêm vào giỏ"	
2. Trả về số lượng sản phẩm trong giỏ hàng	
Alternate Flows	
Title	Description
Exception Flows	
Title	Description

3.4.3 Xóa sản phẩm khỏi giỏ hàng

Bảng 12 Đặc tả usecase xóa sản phẩm khỏi giỏ hàng

Name	Xóa sản phẩm vào giỏ hàng
Brief Description	Cho phép người dùng xóa sản phẩm vào giỏ hàng.
Actor(s)	User, Employee, Admin, Guest
Pre-Conditions	
Title	Description
	1. Người dùng đang truy cập trang giỏ hàng (usecase xem giỏ hàng)
Post-Conditions	
Title	Description
Thành công	Người dùng xóa sản phẩm vào giỏ hàng thành công.
Main Flow	
1. Nhấn vào biểu tượng "X" của sản phẩm cần xóa.	
2. Trả về thông tin của giỏ hàng. [2.a]	
Alternate Flows	
Title	Description
Exception Flows	
Title	Description
[2.a]	Trả về thông báo "Không tìm thấy item trong gio hang" + id

3.4.4 Đặt hàng

Bảng 13 Đặc tả usecase đặt hàng

Name	Đặt hàng
Brief Description	Cho phép người dùng có thể đặt hàng
Actor(s)	User, Admin, Employee
Pre-Conditions	
Title	Description
	<ol style="list-style-type: none"> 1. Người dùng đã đăng nhập vào trang dành cho khách hàng. 2. Người dùng đang truy cập trang giỏ hàng.
Post-Conditions	
Title	Description
Thành công	<ol style="list-style-type: none"> 1. Đơn hàng được tạo thành công. Nếu người dùng chọn phương thức thanh toán là thanh toán qua ví điện tử Momo hoặc thanh toán qua cổng thanh toán Vnpay thì phải tiếp tục thực hiện use case thanh toán qua ví điện tử Momo hoặc thanh toán qua cổng thanh toán Vnpay, nếu không đơn hàng sẽ bị hủy.
Main Flow	
	<ol style="list-style-type: none"> 1. Nhấn nút “Tiến hành thanh toán”. 2. Nhập họ tên, email, số điện thoại, địa chỉ giao hàng, thành phố/ tỉnh, quận/ huyện, xã/ phường, ghi chú. 3. Nhấn nút “Tiếp theo”. 4. Chọn phương thức vận chuyển, phương thức thanh toán. 5. Nhấn nút “Tiếp tục”. [5.1] 6. Nhấn nút “Thanh toán”. Đặt hàng thành công. Kết thúc Use case
Alternate Flows	
Title	Description
[5.1]	Nhấn nút “Quay lại” để kiểm tra họ tên, email, số điện thoại, địa chỉ giao hàng, thành phố/ tỉnh, quận/ huyện, xã/ phường, ghi chú.
Exception Flows	
Title	Description

3.4.5 Thanh toán bằng ví điện tử Momo

Bảng 14 Đặc tả usecase thanh toán bằng ví điện tử Momo

Name	Thanh toán bằng ví điện tử Momo
Brief Description	Cho phép người dùng có thể thanh toán đơn đặt hàng bằng ví điện tử Momo
Actor(s)	User, Admin, Employee
Pre-Conditions	
Title	Description
	<ol style="list-style-type: none"> 1. Người dùng đã đăng nhập trên trang web dành cho khách hàng. 2. Người dùng đã đăng nhập tài khoản trên App Momo và đang ở màn hình chính. 3. Người dùng đang truy cập trang đặt hàng và hoàn thành xong bước thứ 5 trong Use case “Đặt hàng” 4. Người dùng chọn “Thanh toán qua ví điện tử Momo” khi chọn phương thức thanh toán (bước thứ 4 Use case “Đặt hàng”) 5. Bước 1 trong Use case “Thanh toán bằng ví điện tử Momo” tương đương bước tiếp theo sau Use case “Đặt hàng”.
Post-Conditions	
Title	Description
Thành công	Người dùng thanh toán đơn đặt hàng thành công thông qua ví điện tử Momo.
Main Flow	
	<ol style="list-style-type: none"> 1. Nhấn nút “Thanh toán” trên trang đặt hàng. 2. Nhấn nút “Quét mã” trên App Momo.[2.1][2.2] 3. Quét mã QR code hiện thị trên web Momo. 4. Chọn tài khoản/ thẻ, chọn thẻ quà tặng (nếu có). 5. Nhấn nút “Xác nhận”. 6. Nhập mật khẩu tài khoản “Momo”. 7. Nhấn nút “Xác nhận”. 8. Hiển thị thông báo thanh toán thành công. Kết thúc Use case.

Alternate Flows	
Title	Description
[2.1]	<p>2.1.1 Nhấn nút “Quay về” trên web Momo.</p> <p>2.1.2 Nhấn nút “Hủy giao dịch”.</p> <p>2.1.3 Hiển thị thông báo “Thanh toán không thành công”. Kết thúc Use case</p>
Tile	Description
[2.2]	<p>2.2.1 Hết thời gian 15 phút thanh toán.</p> <p>2.2.2 Hiển thị thông báo “Thanh toán không thành công”. Kết thúc Use case.</p>
Exception Flows	
Title	Description

3.4.6 Thanh toán bằng cổng thanh toán Vnpay

Bảng 15 Đặc tả usecase bằng cổng thanh toán Vnpay

Name	Thanh toán bằng cổng thanh toán Vnpay
Brief Description	Cho phép người dùng có thẻ thanh toán đơn đặt hàng bằng cổng thanh toán. Use case này hướng dẫn thanh toán bằng VISA
Actor(s)	User, Admin, Employee
Pre-Conditions	
Title	Description
	<ol style="list-style-type: none"> Người dùng đã đăng nhập trên trang web dành cho khách hàng. Người dùng đang truy cập trang đặt hàng và hoàn thành xong bước thứ 5 trong Use case “Đặt hàng” Người dùng chọn “Thanh toán qua cổng điện tử VNPay” khi chọn phương thức thanh toán (bước thứ 4 Use case “Đặt hàng”) Bước 1 trong Use case “Thanh toán bằng cổng thanh toán VNPay” tương đương bước 6 trong Use case “Đặt hàng”.
Post-Conditions	

Title	Description
Thành công	Người dùng thanh toán đơn đặt hàng thành công thông qua cổng thanh toán VNPay
Main Flow	
<ol style="list-style-type: none"> 1. Nhấn nút “Thanh toán” trên trang đặt hàng. Sau đó được chuyển tiếp sang trang thanh toán Vnpay 2. Chọn phương thức thanh toán VISA trong mục "Thẻ thanh toán quốc tế". 3. Nhập số thẻ, ngày hết hạn, CVC/CVV, họ và tên khách hàng, email, quốc gia, tỉnh/thành phố/ địa chỉ. [3.1] 4. Chọn hoặc nhập mã khuyến mại (nếu có). 5. Nhấn nút “Tiếp tục”. 6. Nhập code OTP. 7. Nhấn nút “Submit”. [7.1], [7.2] 8. Hiển thị thông báo thanh toán thành công. Kết thúc Use case. 	
Alternate Flows	
Title	Description
Title	Description
[7.1] [OTP không hợp lệ]	7.1.1 Nhấn nút “RESEND CODE”. 7.1.2 Trả về code OTP mới
Title	Description
Exception Flows	
Title	Description
[3.1] [Hủy thanh toán]	2.1.1 Nhấn nút “Hủy thanh toán” trên cổng VNPay. 2.1.2 Nhấn nút “Xác nhận hủy”. 2.1.3 Hiển thị thông báo “Thanh toán không thành công”. Kết thúc Use case
[7.2] [Hủy thanh toán]	7.2.1 Nhấn nút “CANCEL” 7.2.2. Trả về thông báo "Có lỗi xảy ra trong quá trình xử lý.". Kết thúc Use case

3.4.7 Trò chuyện cùng Chatbot AI

Bảng 16 Đặc tả usecase trò chuyện cùng Chatbot AI

Name	Trò chuyện cùng Chatbot AI
Brief Description	Cho phép người dùng tra cứu thông tin sản phẩm một cách nhanh chóng thông qua hệ thống Chatbot AI của hệ thống
Actor(s)	Guess, User, Admin, Employee
Pre-Conditions	
Title	Description
Post-Conditions	
Title	Description
Thành công	Người dùng tra cứu thông tin thành công.
Main Flow	
1.	Nhấn nút có biểu tượng Chatbot phía dưới bên phải màn hình.
2.	Hiển thị khung trò chuyện cùng với Chatbot.
3.	Nhập câu hỏi và nhấn nút có biểu tượng gửi.
4.	Trả về câu trả lời cho người dùng. [4.a]
Alternate Flows	
Title	Description
[4.a]	Trả về phản hồi không tìm thấy thông tin cho người dùng: “I'm sorry, but I don't have information about” + tên chủ đề câu hỏi.
Exception Flows	
Title	Description

3.4.8 Xem danh sách sản phẩm được đề xuất

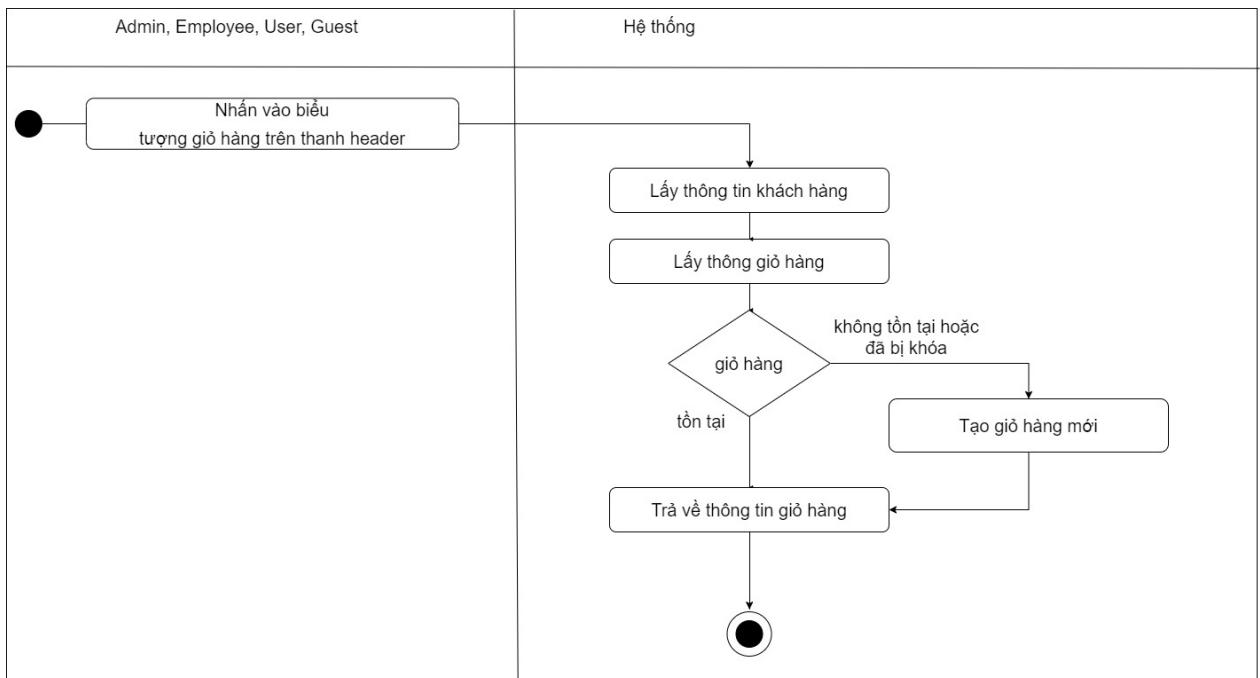
Bảng 17 Đặc tả usecase xem danh sách sản phẩm được đề xuất

Name	Xem danh sách sản phẩm được đề xuất
Brief Description	Cho phép người xem danh sách sản phẩm được đề xuất
Actor(s)	User, Admin, Employee
Pre-Conditions	
Title	Description

	Người dùng đã đăng nhập, đã từng mua hàng và đơn hàng đã được xác nhận.
Post-Conditions	
Title	Description
Thành công	Người dùng tra cứu thông tin thành công.
Main Flow	
1. Truy cập vào trang chủ của hệ thống tại đường dẫn https://filtroncoffee.com/ 2. Hiển thị danh sách sản phẩm đề xuất (sản phẩm được đánh giá) [2.a]	
Alternate Flows	
Title	Description
[2.a]	Hiển thị danh sách 10 sản phẩm có lượt đánh giá trung bình cao nhất. (đơn hàng được xác nhận và chưa đánh giá sản phẩm nào)
Exception Flows	
Title	Description

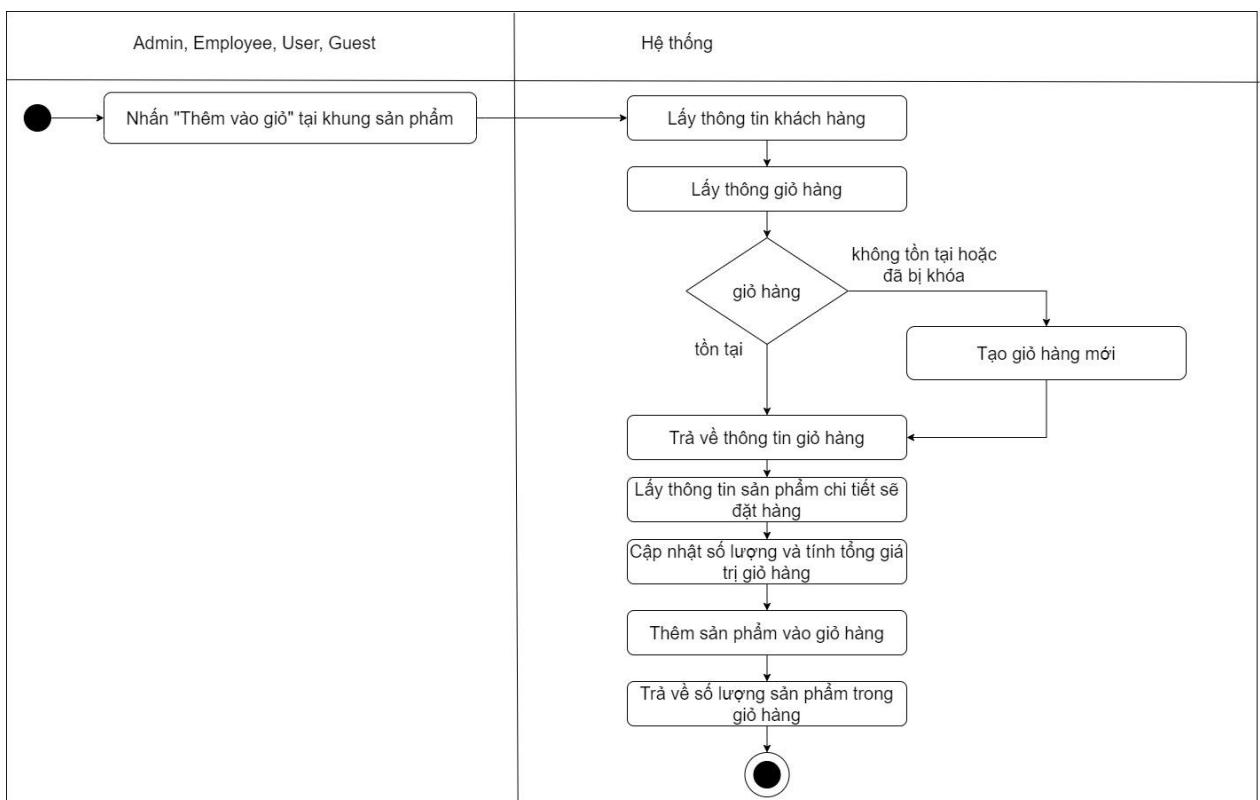
3.5. LUẬC ĐỒ ACTIVITY

3.5.1 Xem giỏ hàng



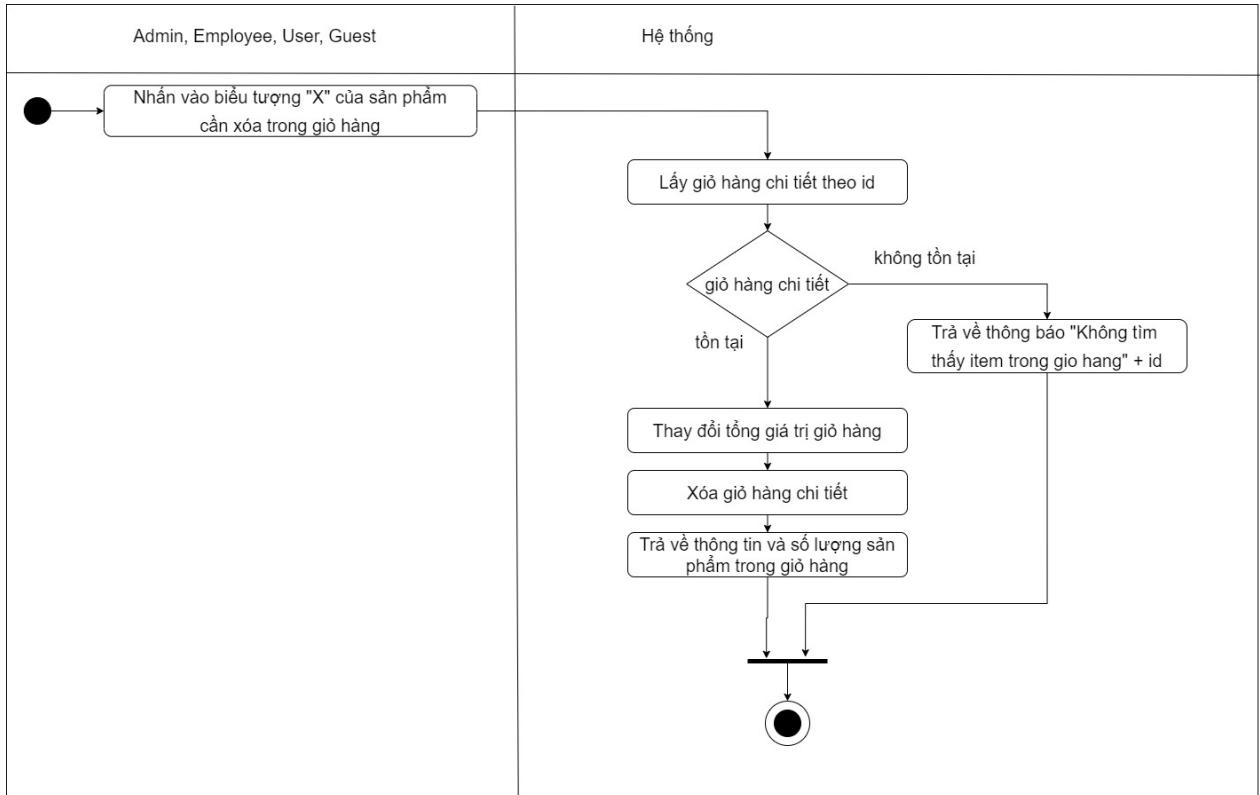
Hình 18 Activity diagram – xem giỏ hàng

3.5.2 Thêm sản phẩm vào giỏ hàng



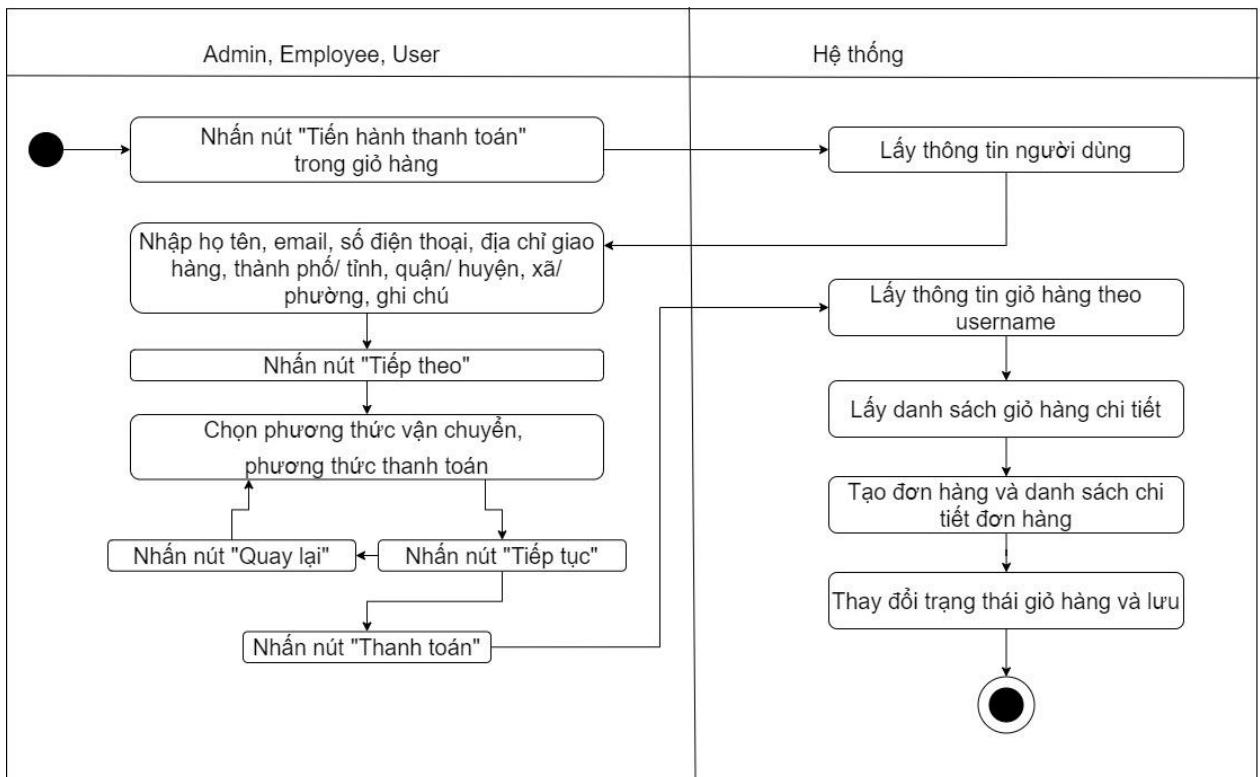
Hình 19 Activity diagram – thêm sản phẩm vào giỏ hàng

3.5.3 Xóa sản phẩm khỏi giỏ hàng



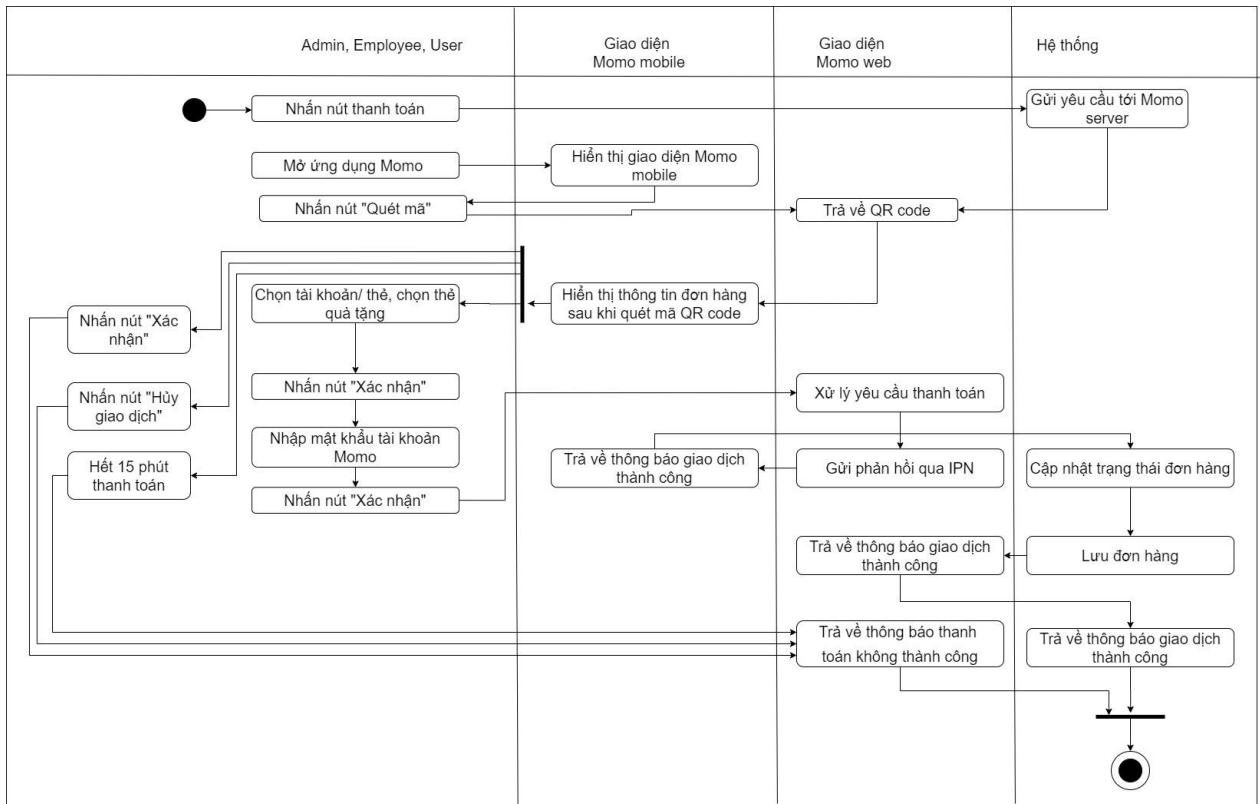
Hình 20 Activity diagram – xóa sản phẩm khỏi giỏ hàng

3.5.4 Đặt hàng



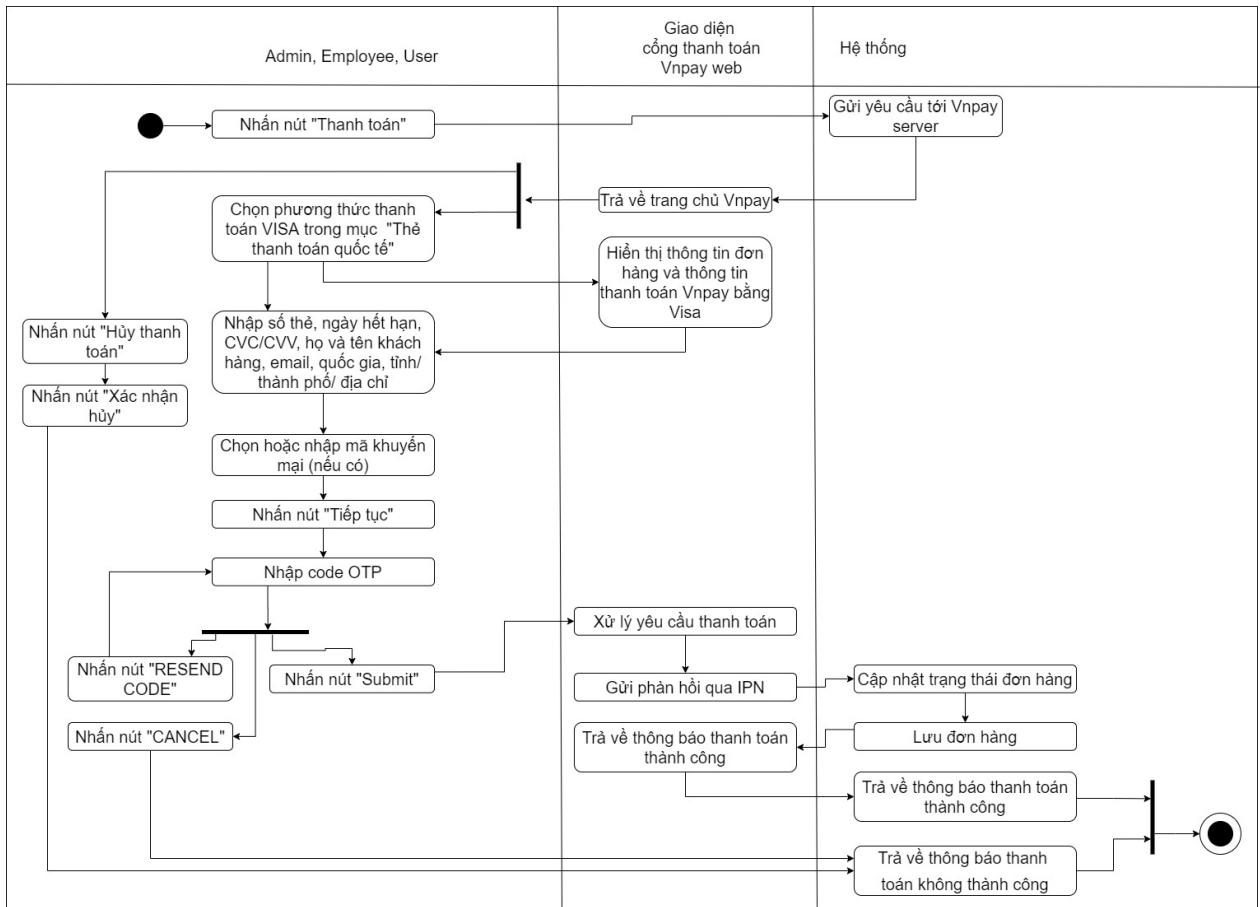
Hình 21 Activity diagram – đặt hàng

3.5.5 Thanh toán bằng ví điện tử Momo



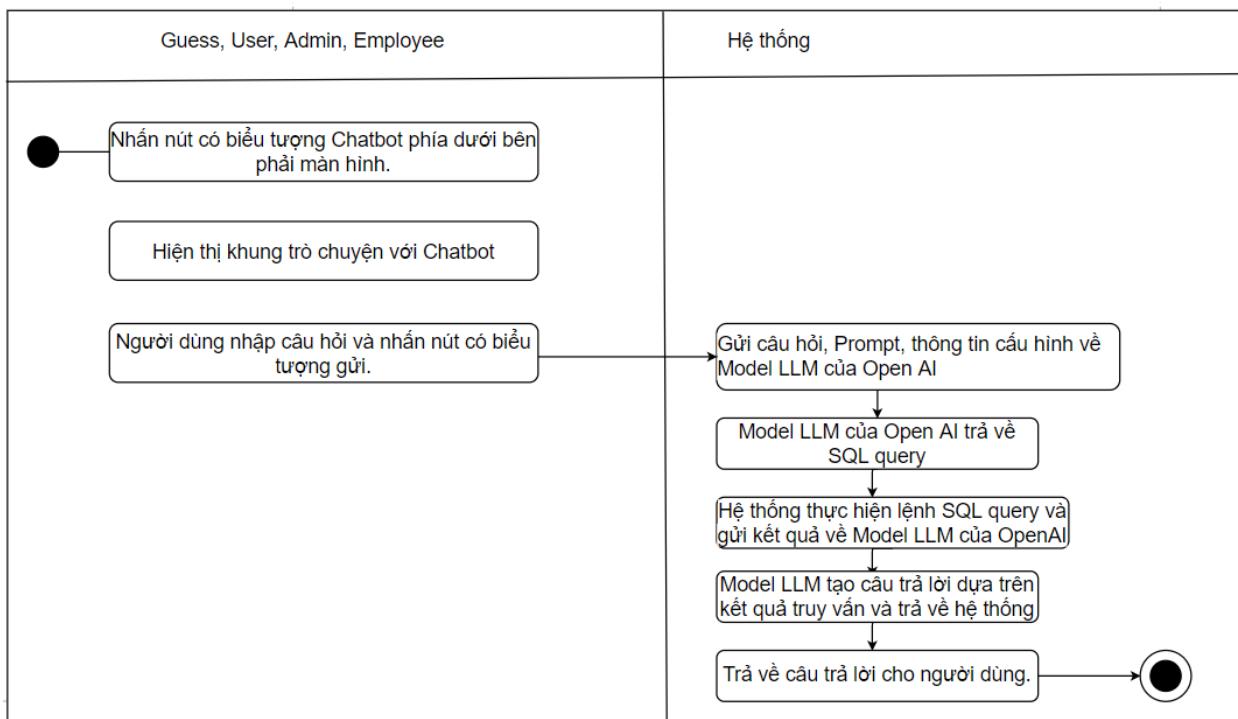
Hình 22 Activity diagram – thanh toán bằng ví điện tử Momo

3.5.6 Thanh toán bằng cổng thanh toán Vnpay



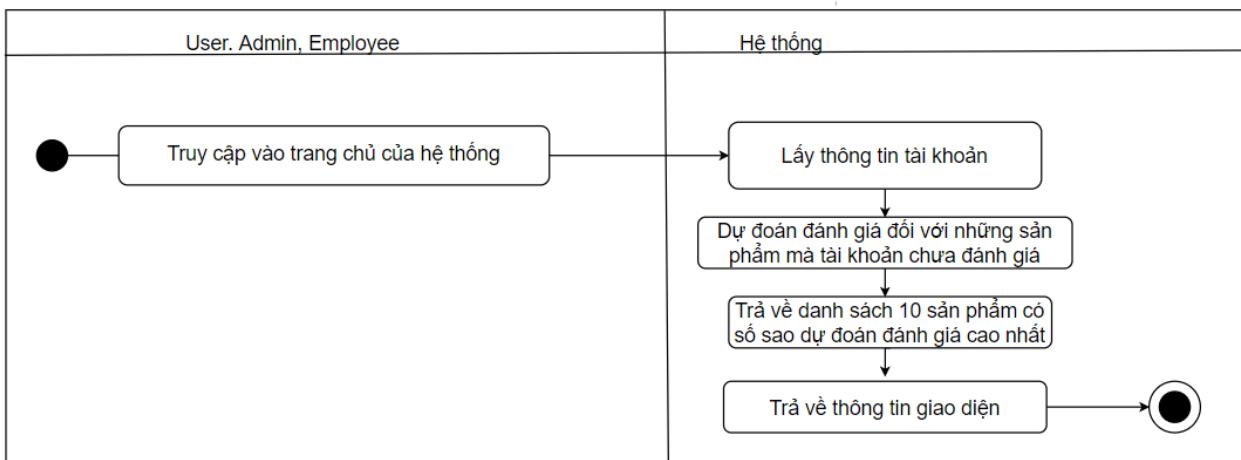
Hình 23 Activity diagram – thanh toán bằng cổng thanh toán Vnpay

3.5.7 Trò chuyện cùng Chatbot AI



Hình 24 Activity diagram – trò chuyện cùng Chatbot AI

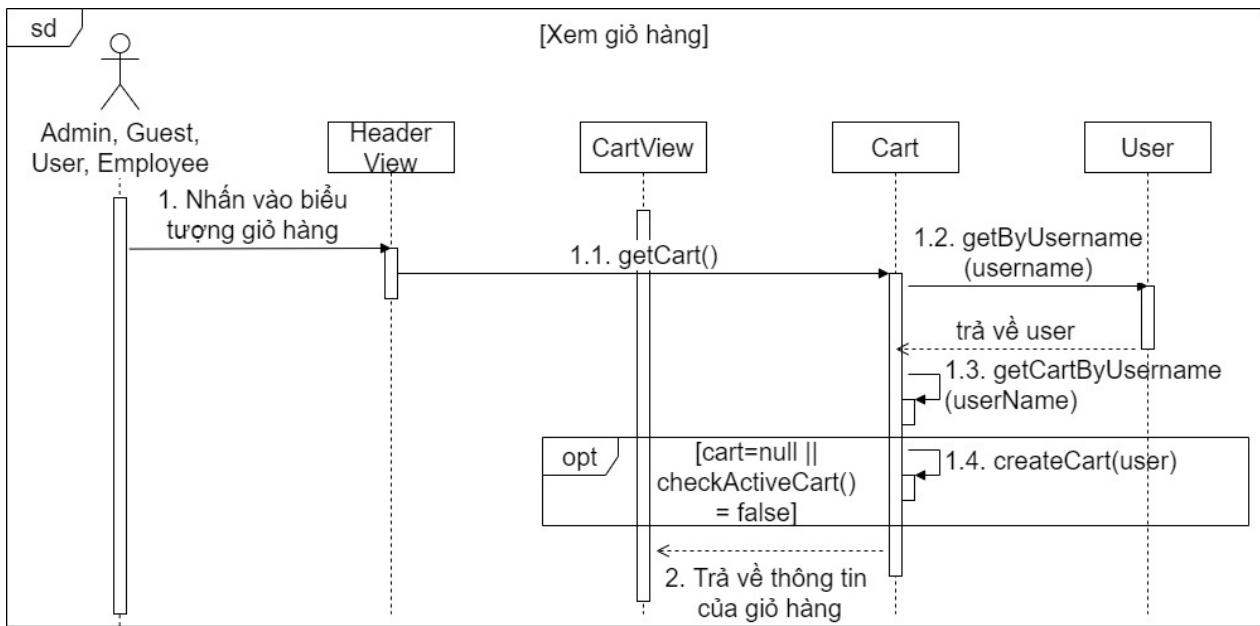
3.5.8 Xem danh sách sản phẩm được đề xuất



Hình 25 Activity diagram – xem danh sách sản phẩm được đề xuất

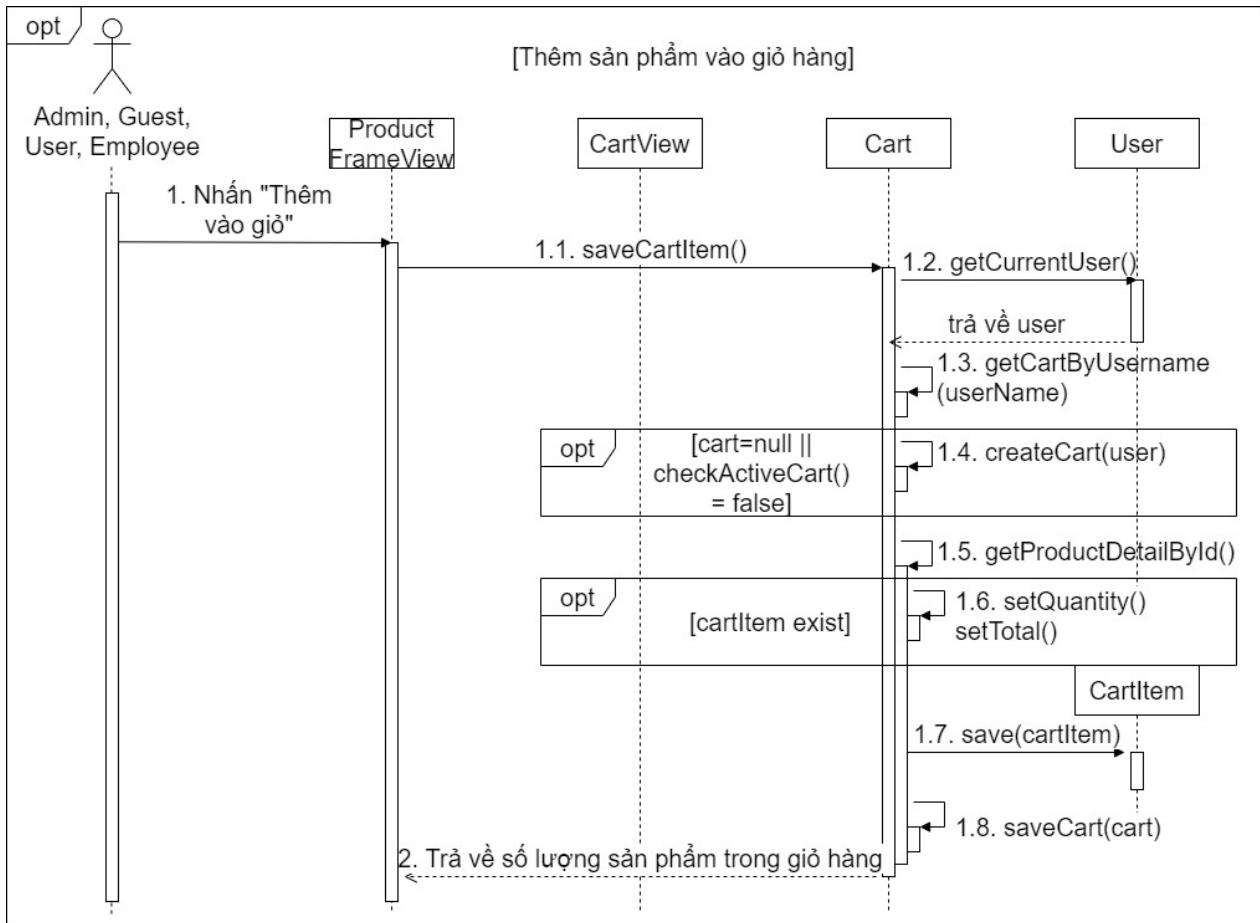
3.6. LUẬC ĐỒ SEQUENCE

3.6.1 Xem giỏ hàng



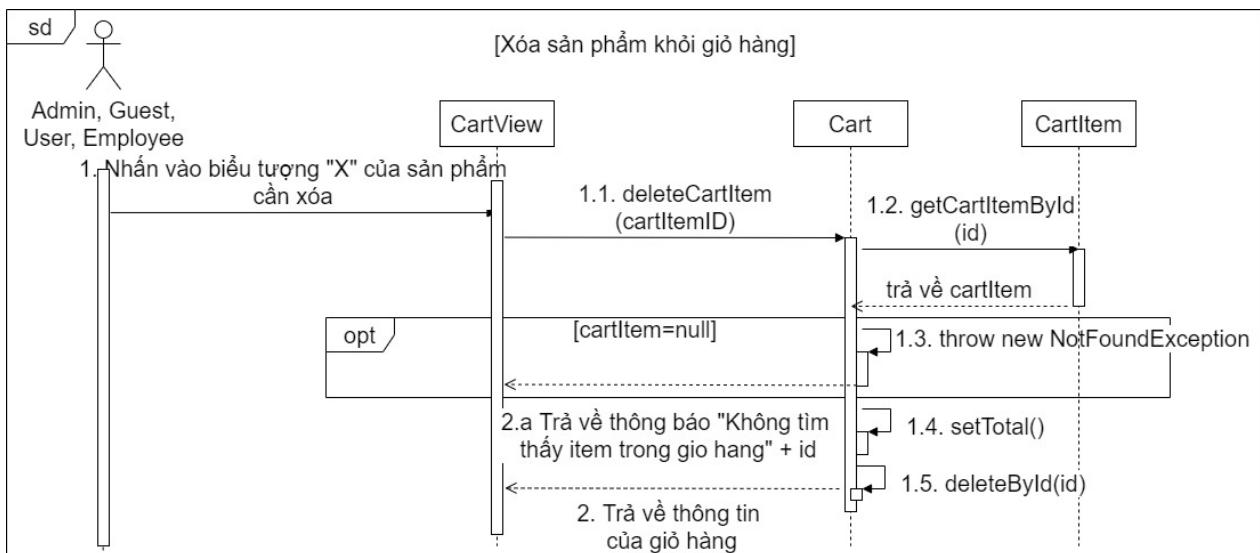
Hình 26 Sequence diagram – xem giỏ hàng

3.6.2 Thêm sản phẩm vào giỏ hàng



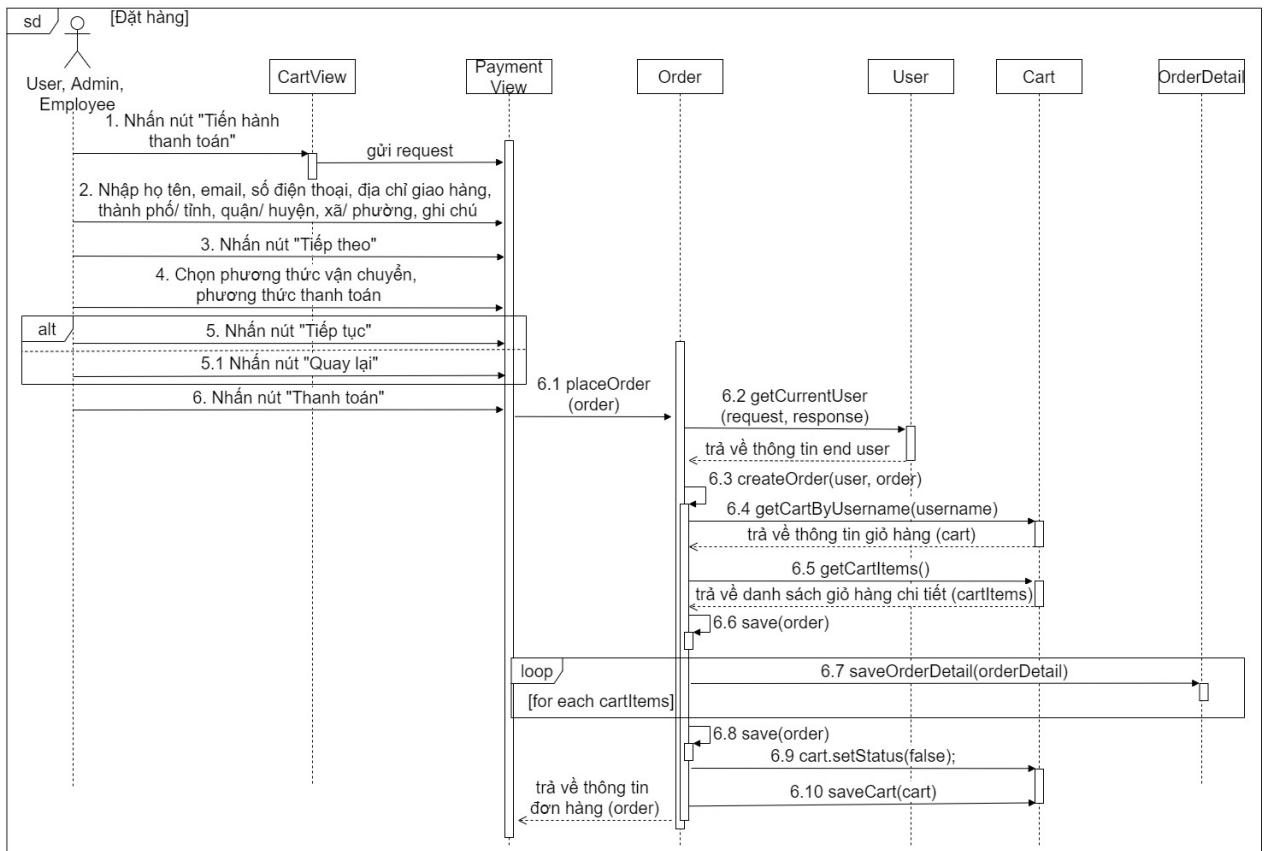
Hình 27 Sequence diagram – thêm sản phẩm vào giỏ hàng

3.6.3 Xóa sản phẩm khỏi giỏ hàng



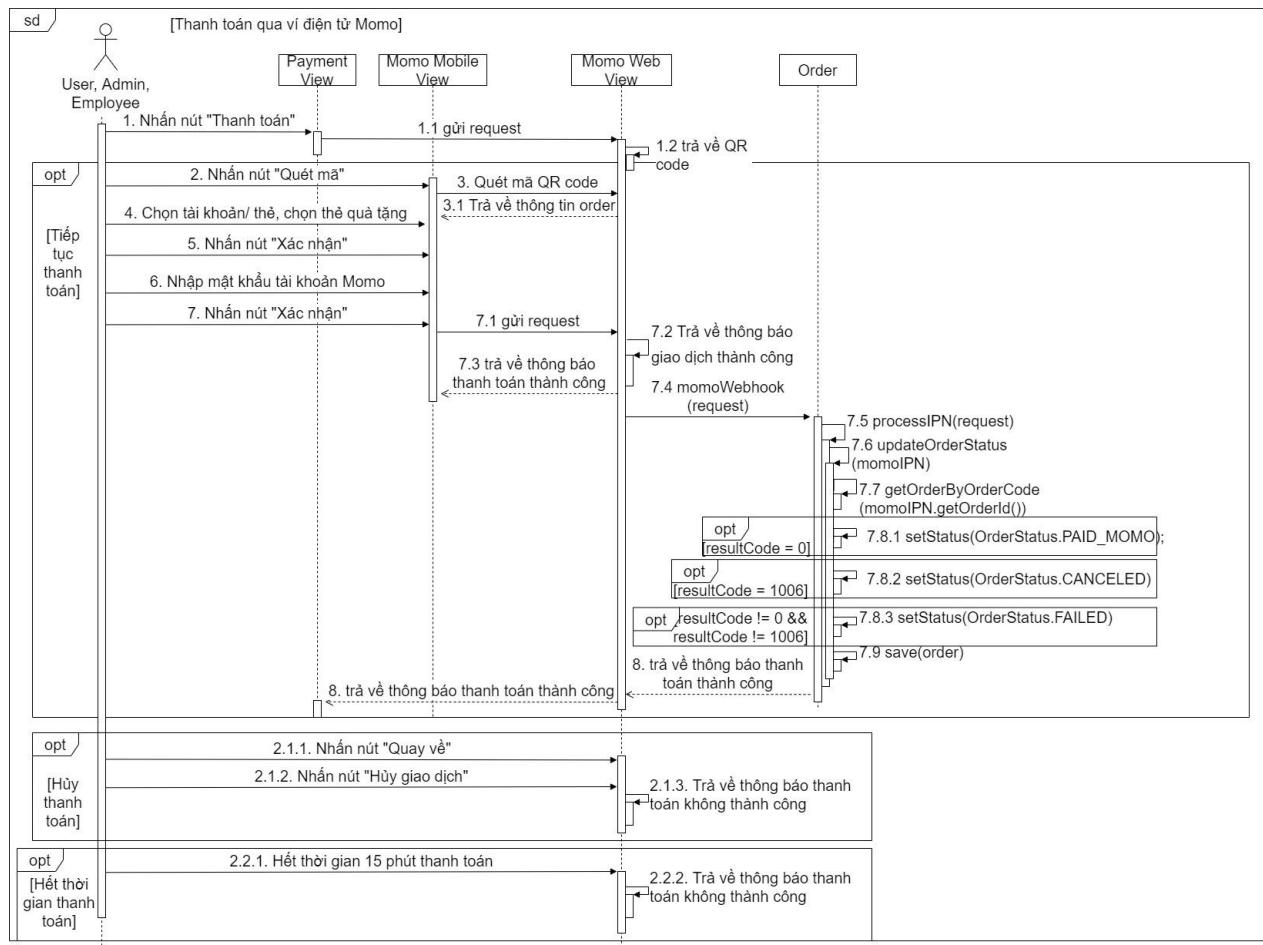
Hình 28 Sequence diagram – xóa sản phẩm khỏi giỏ hàng

3.6.4 Đặt hàng



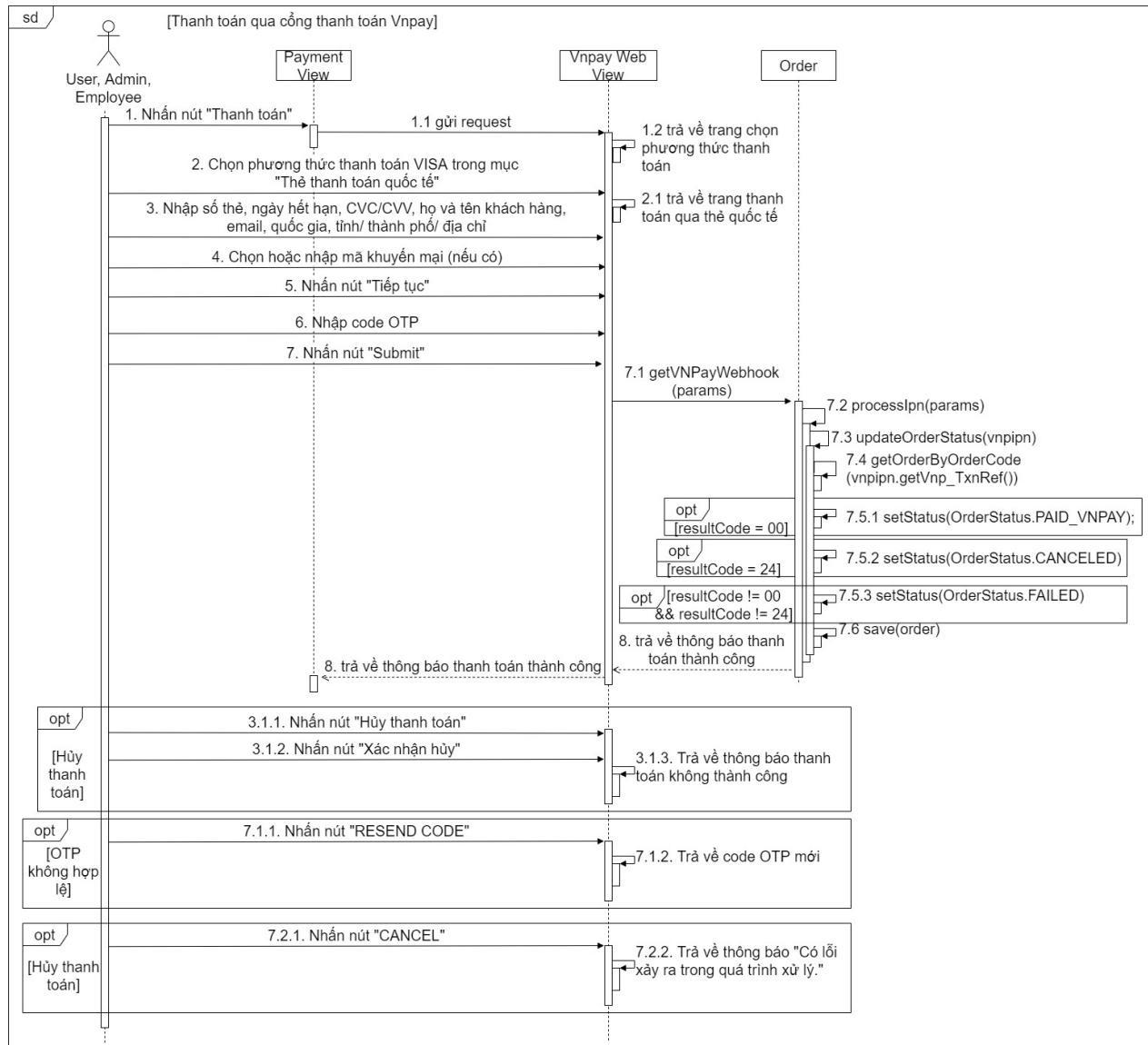
Hình 29 Sequence diagram – đặt hàng

3.6.5 Thanh toán bằng ví điện tử Momo



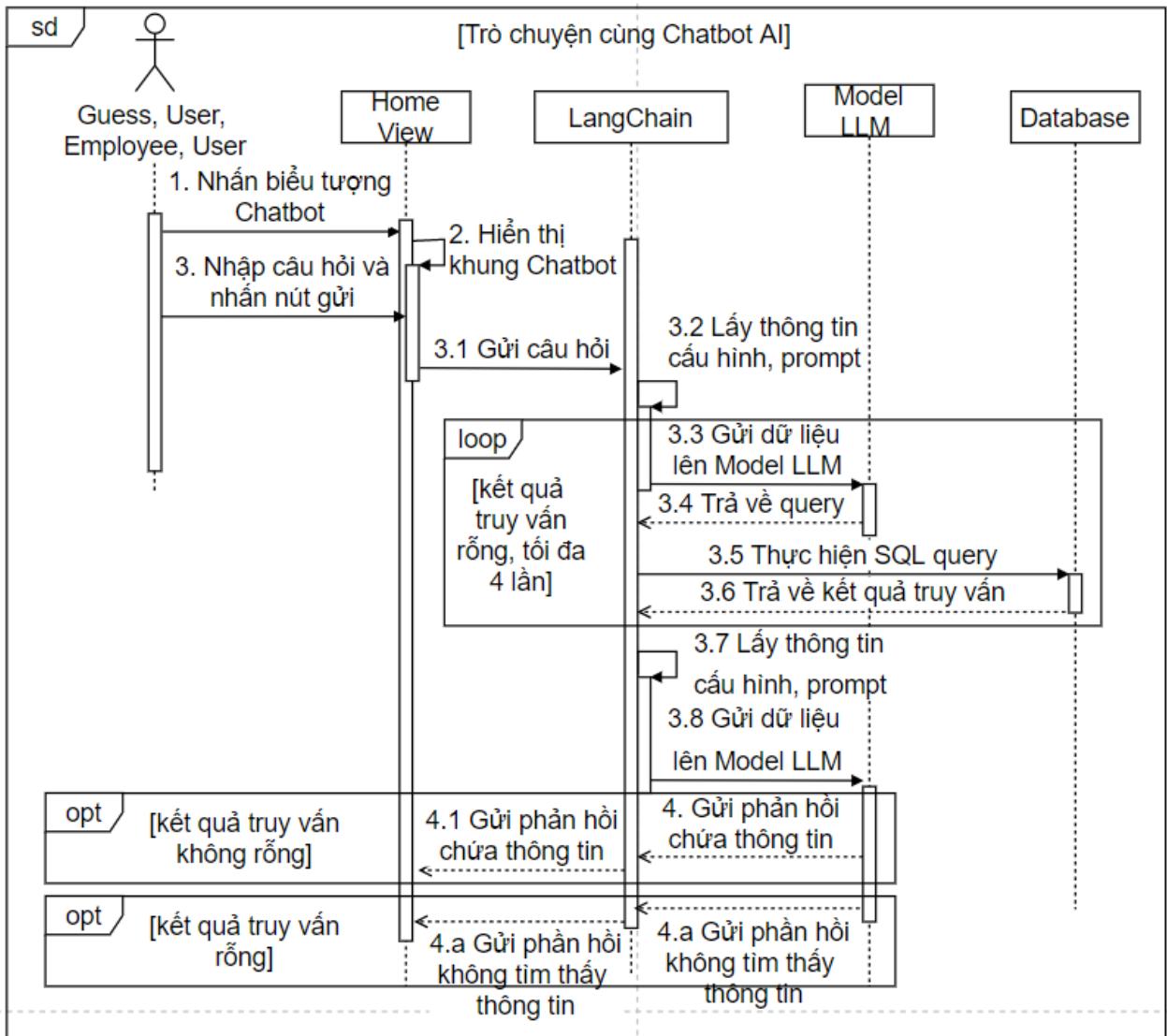
Hình 30 Sequence diagram – thanh toán bằng ví điện tử Momo

3.6.6 Thanh toán bằng cổng thanh toán Vnpay



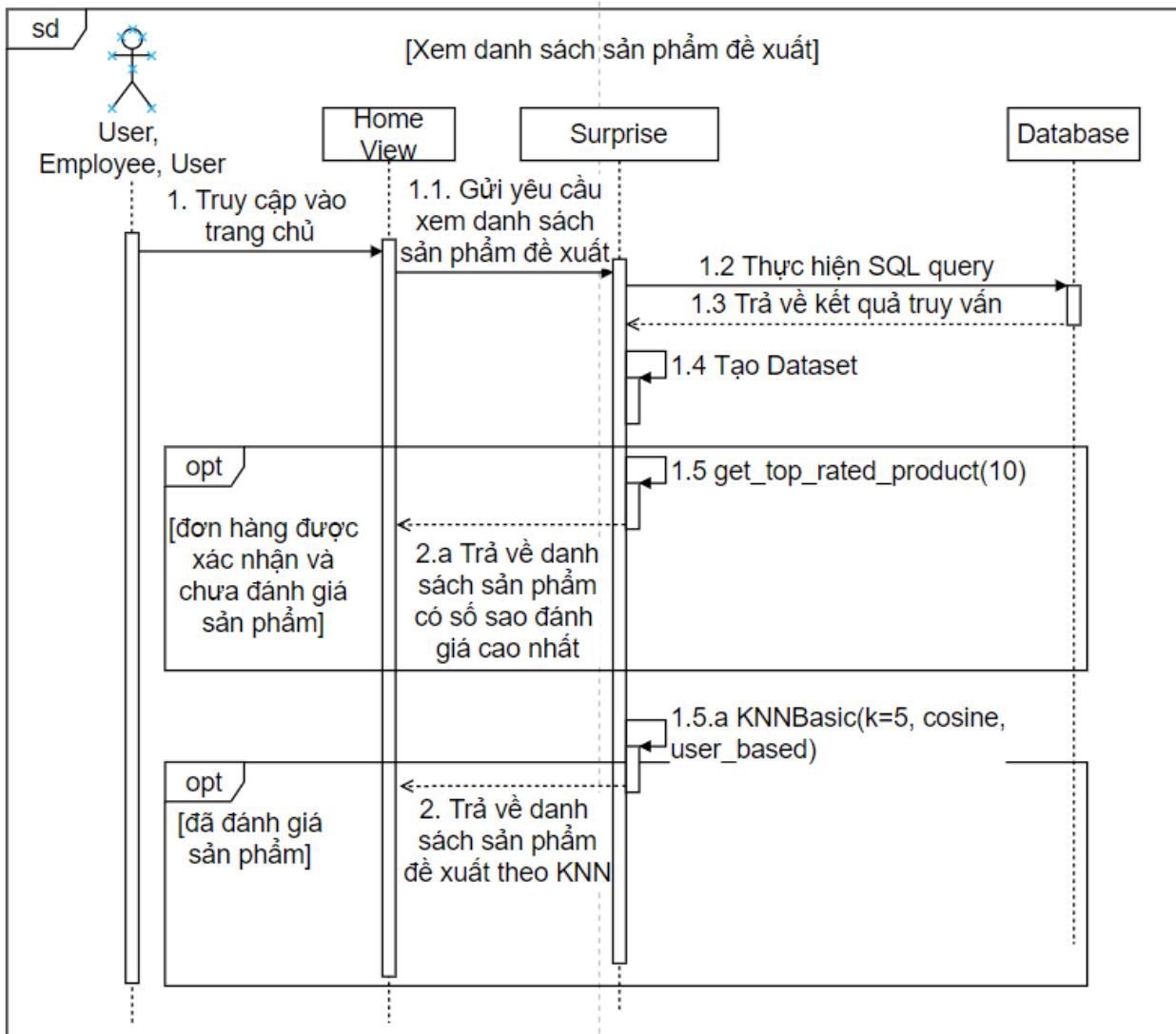
Hình 31 Sequence diagram – thanh toán bằng cổng thanh toán Vnpay

3.6.7 Trò chuyện cùng Chatbot AI



Hình 32 Sequence diagram – Trò chuyện cùng Chatbot AI

3.6.8 Xem danh sách sản phẩm đề xuất



Hình 33 Sequence diagram – Xem danh sách sản phẩm được đề xuất

3.7. RECOMMENDATION SYSTEM – K-NEAREST NEIGHBOR

3.7.1 Tiết xử lý dữ liệu

```
import pandas as pd

review_query = "SELECT * FROM REVIEW"

product_query = "SELECT * FROM PRODUCT"

reviews = pd.read_sql(review_query, self.db)

products = pd.read_sql(product_query, self.db)
```

Sử dụng thư viện pandas để thực hiện truy vấn các câu SQL query. Ở đây, model sẽ lấy dữ liệu từ 2 bảng Review và Product.

Đây là cấu tạo của 2 bảng:

Product: {id, name, slogan, brand_id, sold, rating, flavor_id, description, created_at, updated_at, is_special, is_limited, status, sale_id, category_id, product_origin_id, vendor_id}

Review: {id, user_id, product_id, rating, comment, created_at, parent_id}

Đối với table Product, tiến hành đổi tên để các trường không bị trùng với table Review khi tiến hành gộp.

- "id": "product_id"
- "rating": "avg_rating"
- "created_at": "cre_at"

Sử dụng thư viện pandas để gộp 2 table Review và Product theo cách left join thông qua khóa product_id để giữ lại toàn bộ dữ liệu trên bảng Review. Sau đó loại bỏ những cột không cần thiết, ta được bảng dữ liệu như sau:

Bảng 18 Bảng dữ liệu mẫu cho thuật toán KNN

id	user_id	product_id	rating	comment	created_at	parent_id	name
----	---------	------------	--------	---------	------------	-----------	------

55	19	7	1	Ngon		NaN	
59	44	32	5	Tốt		NaN	
66	19	31	5	Ngon		NaN	
68	19	18	5	Ôn		NaN	

Các đánh giá bao gồm 2 loại:

- Đánh giá kèm số sao
- Phản hồi đánh giá của người dùng khác

Tập dữ liệu sẽ loại bỏ những đánh giá không kèm số sao và loại bỏ các cột {“id”, “comment”, “created_at”, “parent_id”}. Lúc này, tập dữ liệu sẽ theo mẫu:

Bảng 19 Tập dữ liệu mẫu cho thuật toán KNN

user_id	product_id	rating	Name
19	7	1	Colombia Granja
44	32	5	Fine Robusta

Sau đó, tập dữ liệu sẽ được tiếp tục xử lý bằng cách sử dụng dữ liệu của 3 cột {user_id, product_id, rating} làm dữ liệu cho Dataset. Sau bước này, tiếp tục tạo nên một tập TrainSet.

Kiểu dữ liệu mẫu của Dataset:

Bảng 20 Dataset cho thuật toán KNN

user_id	product_id	rating
19	7	1
44	32	5

Kiểu dữ liệu mẫu của Trainset:

[(44, 2, 3), (44, 3, 3), (44, 6, 3), (44, 7, 3), ...]

3.7.2 Kiểm tra độ tin cậy của model

Bước 1: Tạo train data và test data

Vì tập dữ liệu nhỏ nên:

Tập train data sẽ được xây dựng từ toàn bộ tập dữ liệu data.

Tập test data là tập kiểm thử được tạo ra từ toàn bộ Train data.

=> Tập train data và tập test data sẽ giống nhau, chỉ khác nhau về định dạng kiểu dữ liệu. (Train data: dạng bảng, test data: dạng tập hợp,
ví dụ: [(19, 7, 1.0), (19, 31, 5.0)])

Bước 2: Kiểm tra độ sai số RMSE và MAE

```
sim_options = {"name": "cosine", "user_based": True}  
k=1  
knn1 = KNNBasic(k, sim_options=sim_options)  
knn1.fit(train_df)  
prediction1 = knn1.test(test_df)  
accuracy.rmse(prediction1)
```

Từ bộ trainset và testset đã được tạo từ trước, kiểm tra giá trị của RMSE. Sử dụng phương pháp cosine để tính toán độ tương đồng, thử nghiệm trên chế độ dựa trên người dùng user_based. Với giá trị K bằng 1, sử dụng thuật toán KNNBasic để huấn luyện mô hình KNNBasic với tập trainset và sử dụng tập test data để tiến hành kiểm thử, từ đó tính ra giá trị RMSE. Sau khi huấn luyện và kiểm thử, giá trị RMSE là 1.0136266691392073.

Trong hệ thống đánh giá sản phẩm từ 1 – 5 sao, giá trị sai số RMSE như trên chấp nhận được. Tiến hành sử dụng Cross-validation để tiến hành tính 2 chỉ số RMSE và MAE với các giá trị K từ 1 – 50.

```
param_grid = {"k":range(1, 50, 1), "sim_options": {"name": ["cosine", "pearson",  
"msd"], "user_based": [True]}}  
  
gs = GridSearchCV(KNNBasic, param_grid, measures=["rmse", "mae"], cv=10)  
  
gs.fit(data)
```

Đối với kiểm thử độ tin cậy của model, nhóm sử dụng 3 phương pháp cosine, pearson, msd để tính độ tương đồng cho cả tập dữ liệu. Sử dụng GridSearchCV để biểu diễn mọi kết quả tính toán sau mỗi vòng lặp cross-validation. Tiến hành kiểm tra với 50 giá trị k, từ 1 đến 50, bước nhảy 1. Với mỗi giá trị k (mặc định sẽ được kiểm tra 3 lần), tập dữ liệu sẽ được chia làm 10 phần để tiến hành kiểm định chéo cross-validation, tập dữ liệu sẽ được huấn luyện với 9 phần, kiểm thử 1 phần. Dựa vào sự chênh lệch của giá trị thực và giá trị kiểm thử để tính các giá trị của RMSE và MAE:

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2}$$

Trong đó y_i là giá trị thực tế; \hat{y}_i là giá trị dự đoán; n là số lượng mẫu

$$MAE = \frac{1}{n} \sum_{i=1}^n |y_i - \hat{y}_i|$$

Trong đó y_i là giá trị thực tế; \hat{y}_i là giá trị dự đoán; n là số lượng mẫu

Đây là kết quả sau khi sử dụng GridSearchCV:

Bảng 21 Kết quả của GridSearchCV

K	rmse0	...	rmse9	mean_rmse	mae0	...	mae9	mean_mae
1	1.660595	...	1.767767	1.749356	1.303030	...	1.437500	1.315862
1	1.636655	...	1.540833	1.524956	1.275193	...	1.188028	1.171132
1	1.325736	...	1.879162	1.674889	1.030303	...	1.468750	1.251752
2	1.687028	...	1.389634	1.550360	1.363165	...	1.203052	1.201864
2	1.627534	...	1.597738	1.518777	1.271181	...	1.244062	1.176940
...
50	1.448340	...	1.493197	1.467406	1.173168	...	1.238048	1.160242

Miền giá trị của RMSE từ 1.4 đến 1.7, miền giá trị của MAE từ 1.1 đến 1.3.

Giá trị trung bình RMSE bằng 1.479049.

Giá trị trung bình MAE bằng 1.163579.

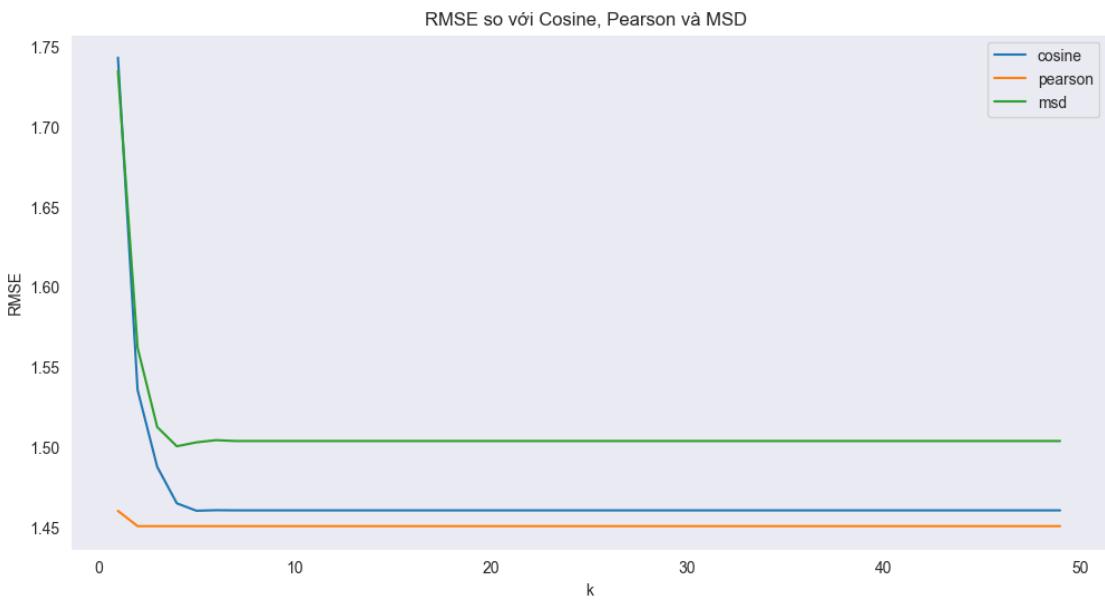
Vì giá trị số sao bình luận có giá trị từ 1 cho đến 5, với 2 giá trị trung bình trên, kết luận Model {user_id, product_id, rating} tin cậy khi được huấn luyện với KNNBasic.

3.7.3 Chọn tham số k

K: số lượng neighbor để xem xét, tính giá trị rating dự đoán của một user đối với 1 sản phẩm cụ thể. Sau khi xác định độ tin cậy của model, tiến hành tìm K. Với kết quả dạng lưới được trả về ở bước trước, tiến hành biểu diễn thành dạng biểu đồ.

Để kết quả chính xác và khách quan hơn, tiến hành huấn luyện nhiều lần (trên 30 lần) để tìm giá trị của K. Vì tập dữ liệu không có nhiều ngoại lệ (dữ liệu có giá trị quá cao hoặc quá thấp so với tập dữ liệu, tần suất xuất hiện quá ít) nên sử dụng chỉ số RMSE để tìm K.

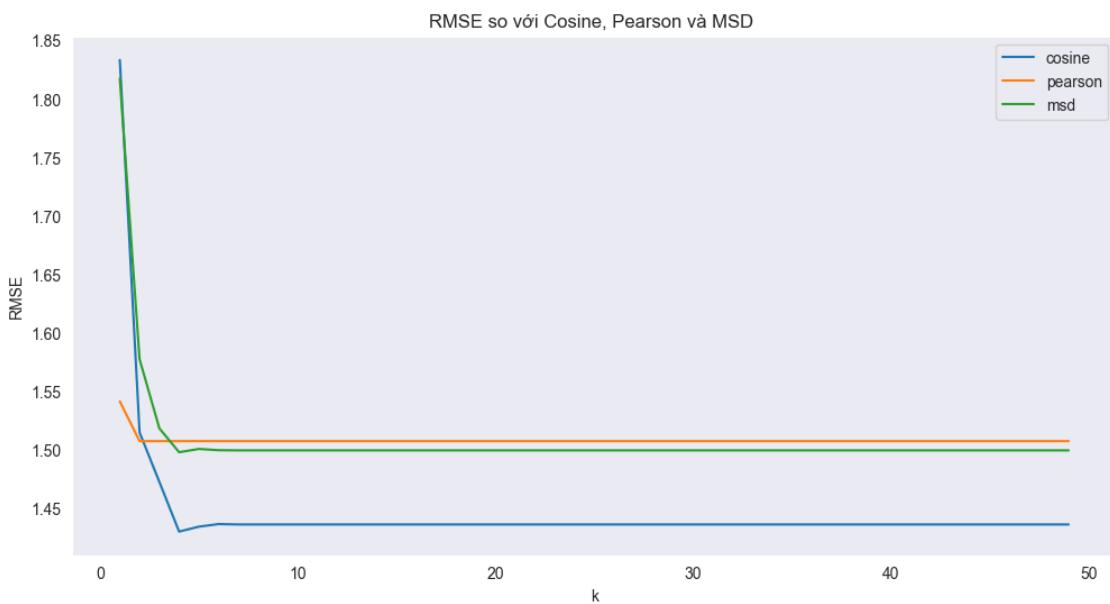
Lần 1:



Hình 34 Biểu đồ RMSE so với Cosine, Pearson và MSD lần 1

Với lần training đầu tiên, ta thấy được với $k = 5$ thì giá trị RMSE là nhỏ nhất 1.450687.

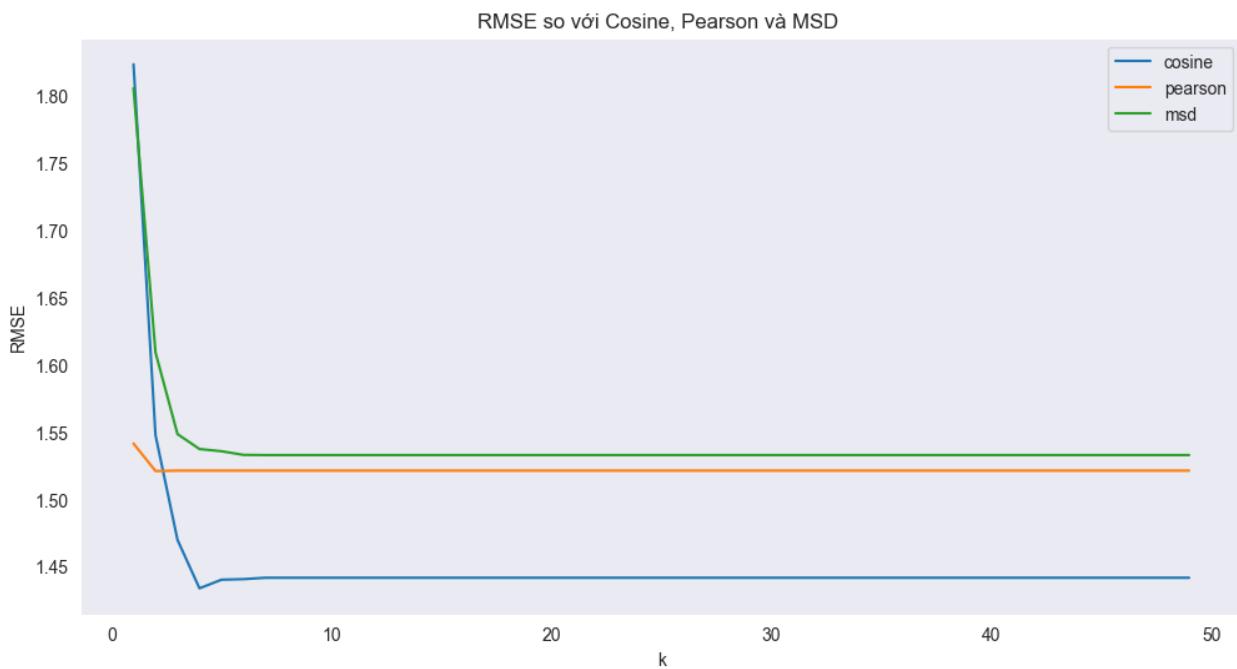
Lần 2:



Hình 35 Biểu đồ RMSE so với Cosine, Pearson và MSD lần 2

Với lần training thứ 2, ta thấy được với $k = 5$ thì giá trị RMSE nhỏ nhất 1.430459.

Lần 3:



Hình 36 Biểu đồ RMSE so với Cosine, Pearson và MSD lần 3

Với lần training thứ 3, ta thấy được với $k = 4$ thì giá trị RMSE nhỏ nhất 1.433869.

Đây là kết quả sau 30 lần đo:

Bảng 22 30 lần đo giá trị RMSE

Lần	K	RMSE
1	5	1.450687
2	5	1.430459
3	4	1.433869
4	5	1.475219
...
29	5	1.4654
30	5	1.464608

⇒ Chọn giá trị K cho thuật toán KNN: 5

Trong các biểu đồ biểu diễn mối liên hệ giữa K, RMSE khi sử dụng các phương pháp cosine, pearson, msd, ta thấy được với phương pháp Cosine, sự biến đổi RMSE là rõ ràng nhất. Nên trong bước xây dựng hàm để gọi ý sản phẩm, phương pháp cosine sẽ được sử dụng để tính khoảng cách giữa các vector – độ tương đồng của các điểm trong tập dữ liệu.

- Pearson : được sử dụng khi muốn đo lường mối quan hệ tuyến tính giữa 2 biến. Việc đánh giá sản phẩm của mỗi người dùng là độc lập, dựa trên cảm tính và yêu thích của mỗi người, nên việc sử dụng MSD là bất hợp lý. Ví dụ khi xét 2 người dùng, tiến hành biểu diễn số sao đánh giá và id sản phẩm lên trên hệ thống tọa độ, 2 tập dữ liệu được xem là tuyến tính nếu các điểm dữ liệu tập trung thành 1 đường thẳng. Việc này rất khó xảy ra khi xét với tập dữ liệu hiện tại, cho dù có mối quan hệ tuyến tính giữa 2 tập dữ liệu của 2 người dùng thì tần xuất xảy ra cũng rất ít.
- MSD (Mean Squared Difference) và cosine: được sử dụng để so sánh sự tương đồng của văn bản, được ứng dụng trong hệ thống gợi ý sản phẩm dựa trên nội dung đánh giá hoặc được sử dụng để tìm ra các sản phẩm tương đồng dựa trên các đặc điểm của sản phẩm hoặc mức độ ưa thích của người dùng đối với sản phẩm đó.

Công thức của cosine:

$$\cos(\theta) = \frac{\mathbf{A} \cdot \mathbf{B}}{\|\mathbf{A}\| \|\mathbf{B}\|} = \frac{\sum_{i=1}^n A_i B_i}{\sqrt{\sum_{i=1}^n A_i^2} \cdot \sqrt{\sum_{i=1}^n B_i^2}},$$

Hình 37 Công thức Cosine [15]

Trong đó, A_i và B_i là các thành phần của vector \mathbf{A} , \mathbf{B} .

3.7.4 Xây dựng hàm

Hệ thống gợi ý sản phẩm.

Input: user_id

Output: danh sách sản phẩm

Bước 1: Tiền xử lý dữ liệu – tạo dataset ở bước 3.7.1

Bước 2: Huấn luyện mô hình KNN với tập datatrain tạo từ bước 1

Bước 3: Kiểm tra nếu dữ liệu đầu vào user_id không tồn tại thì sẽ đề xuất danh sách 10 sản phẩm được đánh giá cao nhất (số sao nhiều nhất)

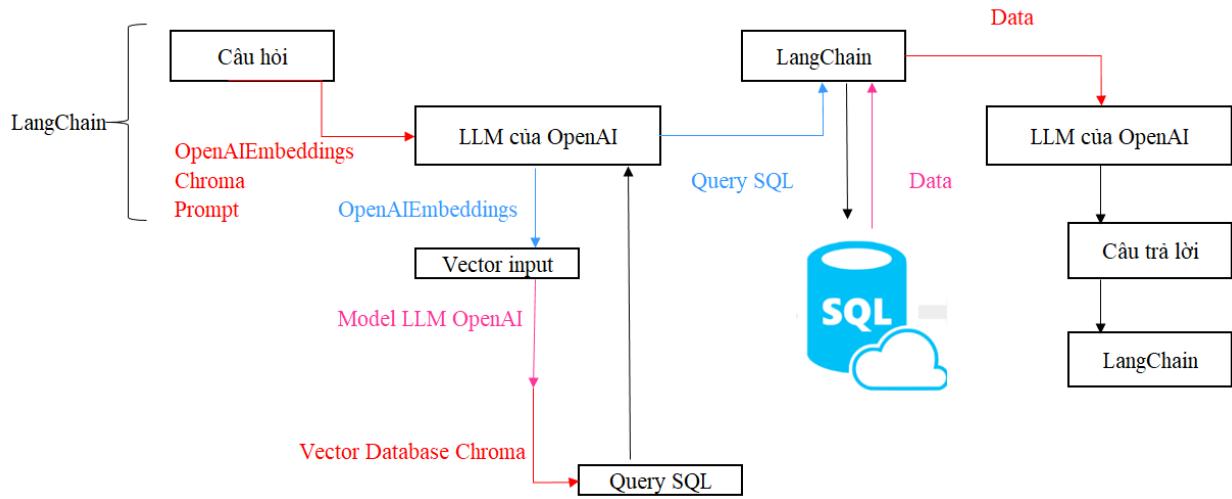
Bước 4: Kết quả của hệ thống đề xuất sản phẩm là danh sách các sản phẩm được sắp xếp dựa trên số sao dự đoán. Tập data test là danh sách sản phẩm mà người dùng (đang xét để đưa ra danh sách sản phẩm đề xuất) chưa đánh giá.

Sau khi lấy được danh sách sản phẩm mà người dùng chưa đánh giá, tiến hành tạo ra một dataset để làm tập data test. Mẫu data test có mẫu {user_id, product_id, avg_rating}. Vì số sao có giá trị từ 1 – 5 sao nên chọn giá trị avg_rating bằng 3 làm giá trị tạm thời để biểu diễn các vector này vào hệ tọa độ, từ đó sử dụng thuật toán KNN để tìm ra danh sách sản phẩm đề xuất.

Sau khi biểu diễn các vector vào hệ trực tọa độ, tiến hành tính khoảng cách hoặc độ tương đồng để tìm ra K vector gần nhất. Từ K vector này, đối với mỗi vector đầu vào, hệ thống sử dụng công thức KNNBasic được nêu ở phần 1.4 để tính ra số sao dự đoán của mỗi sản phẩm – tìm giá trị avg-rating trong mỗi vector của data test. Sau khi tìm ra được tất cả số sao dự đoán của các sản phẩm mà người dùng chưa đánh giá, lấy ra 10 sản phẩm được đánh giá cao nhất và gửi về cho người dùng.

3.8. CHATBOT AI – MODEL LLM CỦA OPENAI

3.8.1 Sơ đồ hoạt động



Hình 38 Sơ đồ hoạt động của ChatBot OpenAI

3.8.2 Tích hợp LLM của OpenAI vào hệ thống chatbot

Bước 1: Cấu hình thông tin cho chatbot.

```
model = "gpt-3.5-turbo-0125"

llm = ChatOpenAI(model=self.model, temperature=0)

example_selector = SemanticSimilarityExampleSelector.from_examples(
    self.examples,
    OpenAIEmbeddings(),
    Chroma(),
    k=5,
    input_keys=["input"]
)

toolkit = SQLDatabaseToolkit(db=self.db, llm=llm)
```

Sử dụng model gpt-3.5-turbo-0125 hoặc mô hình gpt-4-turbo mà OpenAI cung cấp. Cấu hình temperature bằng 0, có nghĩ độ sáng tạo hoặc ngẫu nhiên của câu trả lời sẽ bằng 0, câu trả lời sẽ bám sát dữ liệu của hệ thống mà nhà phát triển cung cấp, không xuất hiện những cụm từ hoặc ký tự khó hiểu hoặc không có nghĩa. Sử dụng

OpenAIEmbeddings làm bộ Embedding cho dữ liệu đầu vào để chuyển câu hỏi thành vector. Sử dụng similarity_search để tìm k vector tương đồng từ vector câu hỏi trong vector database Chroma, từ đó sinh ra token của câu trả lời. Sử dụng bộ công cụ SQLDatabaseToolkit để truy vấn SQL query được trả từ model LLM của OpenAI.

Bước 2: Viết Prompt cho hệ thống Chatbot

Prompt là một “lời nhắc” mà mô hình LLM sẽ dựa vào để hiểu và phân tích câu hỏi. Đây là cấu trúc của một Prompt

```
Prompt = [
    {
        "input": "...",
        "query": "..."
    },
    {
        "input": "...",
        "query": "..."
    }
]
```

Mô hình LLM sẽ dựa các câu input và query mà nhà phát triển cung cấp để hiểu được ngữ cảnh của dữ liệu. Từ đó tăng độ chính xác của câu trả lời mà model phản hồi. Ví dụ, sau khi phân tích prompt, model LLM hiểu khi gặp từ “sản phẩm” trong câu hỏi, thì từ tương ứng trong query sẽ là “product”, “loại sản phẩm” là “category”. Từ đây tránh được sự nhầm lẫn và độ chính xác sẽ được tăng lên.

Ngoài Prompt, nhà phát triển có thể phát triển thêm System_prefix. Đây là tập hợp những hướng dẫn và quy tắc mà nhà lập trình muốn Chatbot thực hiện. Trong dự án này, System_prefix là những hướng dẫn yêu cầu Chatbot AI làm theo như:

- Trả lời một cách thân thiện với vai trò là một đại lý bán hàng;
- Nếu người dùng không chỉ rõ số lượng sản phẩm mà họ muốn, luôn hiển thị tối đa k sản phẩm;

- Không bao giờ truy vấn toàn bộ cột của một table cụ thể, chỉ truy cập một vài cột liên quan tới câu hỏi;
- Có quyền truy cập tới công cụ tương tác với database (được sử dụng SQLDatabaseToolkit);
- Chỉ được sử dụng những công cụ đã được cung cấp;
- Bắt buộc kiểm tra 2 lần câu query trước khi thực thi nó (không được vi phạm các quy tắc được đề ra dưới đây);
- Nếu gặp lỗi khi truy vấn dữ liệu dựa trên câu query, hãy viết lại query và thực thi nó.
- Ngoài ra, hệ thống được cung cấp thêm các “rule” luật lệ khác, những luật lệ này nhắm vào chi tiết thực hiện bên trong câu query, như không được thay đổi dữ liệu, không thực hiện bất kỳ câu lệnh DML nào (insert, update, delete, drop,...), truy cập vào danh sách các bảng trong database, truy cập vào thông tin cá nhân của người khác.
- Những câu hỏi không liên quan tới database, hãy trả lời đơn giản là bạn không biết.

Bước 3: Thông qua API của LangChain, gửi những thông tin đã cấu hình kèm câu hỏi tới Model LLM của OpenAI.

Bước 4: Theo thông tin đã được cấu hình, Model LLM sẽ xử lý dữ liệu từ câu hỏi và sử dụng OpenAIEmbedding để chuyển thành vector input. Sau đó sẽ trải qua các lớp nơron trong Model LLM để tính toán ra một vector có tính liên quan tới câu trả lời. Sau đó, sử dụng những thông tin có trong Prompt và System_prefix và thuật toán similarity_search để tìm k vector tương đồng từ vector câu hỏi trong vector database Chroma (có thể Prompt và System_prefix đã được chuyển thành vector thông qua kỹ thuật OpenAIEmbedding và được lưu trong vector database Chroma). Sau khi tìm ra K vector tương đồng, model LLM tiến hành tạo ra chuỗi token, từ đó hình thành câu query SQL. Sau đó, câu query SQL này sẽ được gửi về LangChain.

Bước 5: Sử dụng bộ công cụ SQLDatabaseToolkit để thực hiện câu query. Sau khi có kết quả, LangChain sẽ gửi kết quả này về Model LLM thông qua API của OpenAI.

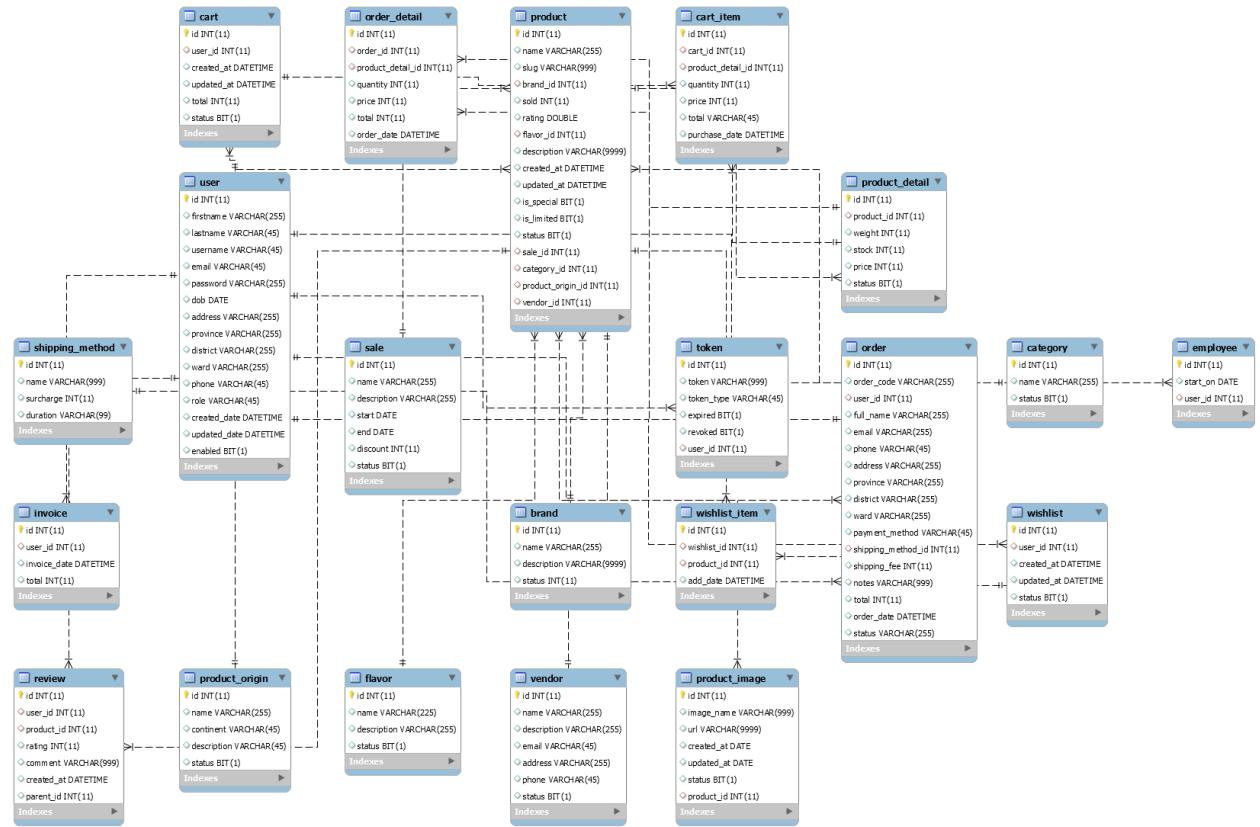
Bước 6: Model LLM sẽ xử lý tương tự như *bước 3*

Bước 7: Sau khi ánh xạ chuỗi token sang câu trả lời, model LLM sẽ gửi câu trả lời về LangChain.

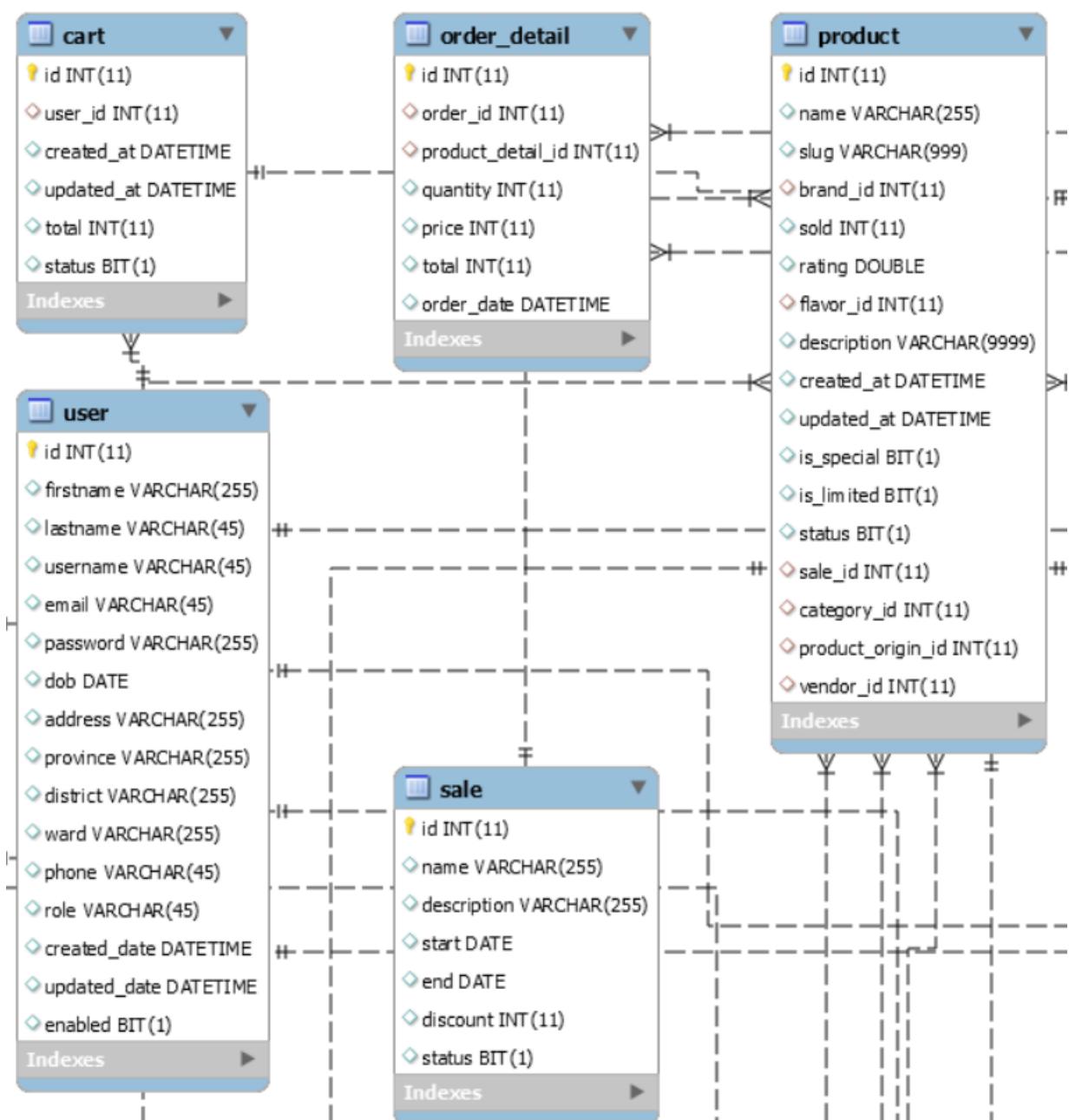
Chương 4: THIẾT KẾ HỆ THÔNG

4.1. THIẾT KẾ DATABASE

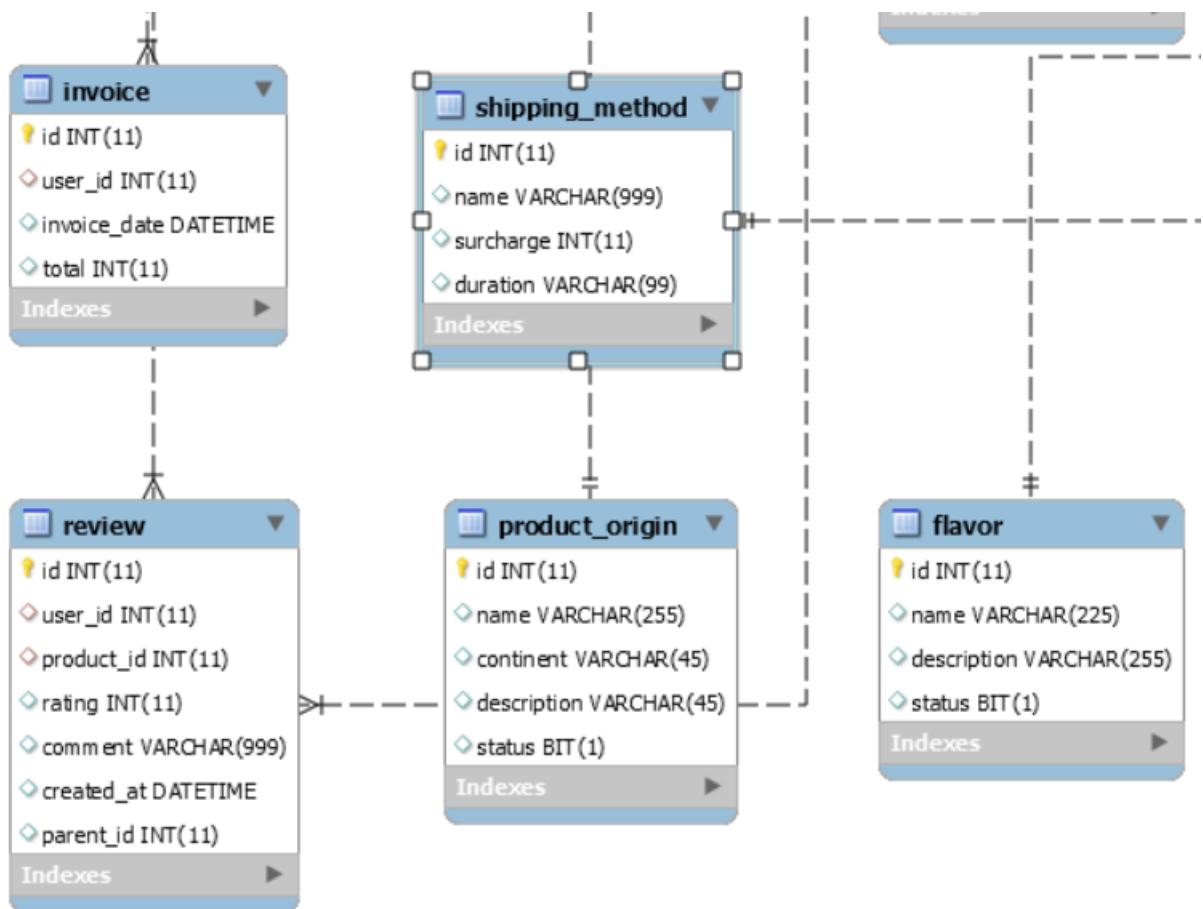
4.1.1. Lược đồ ERD



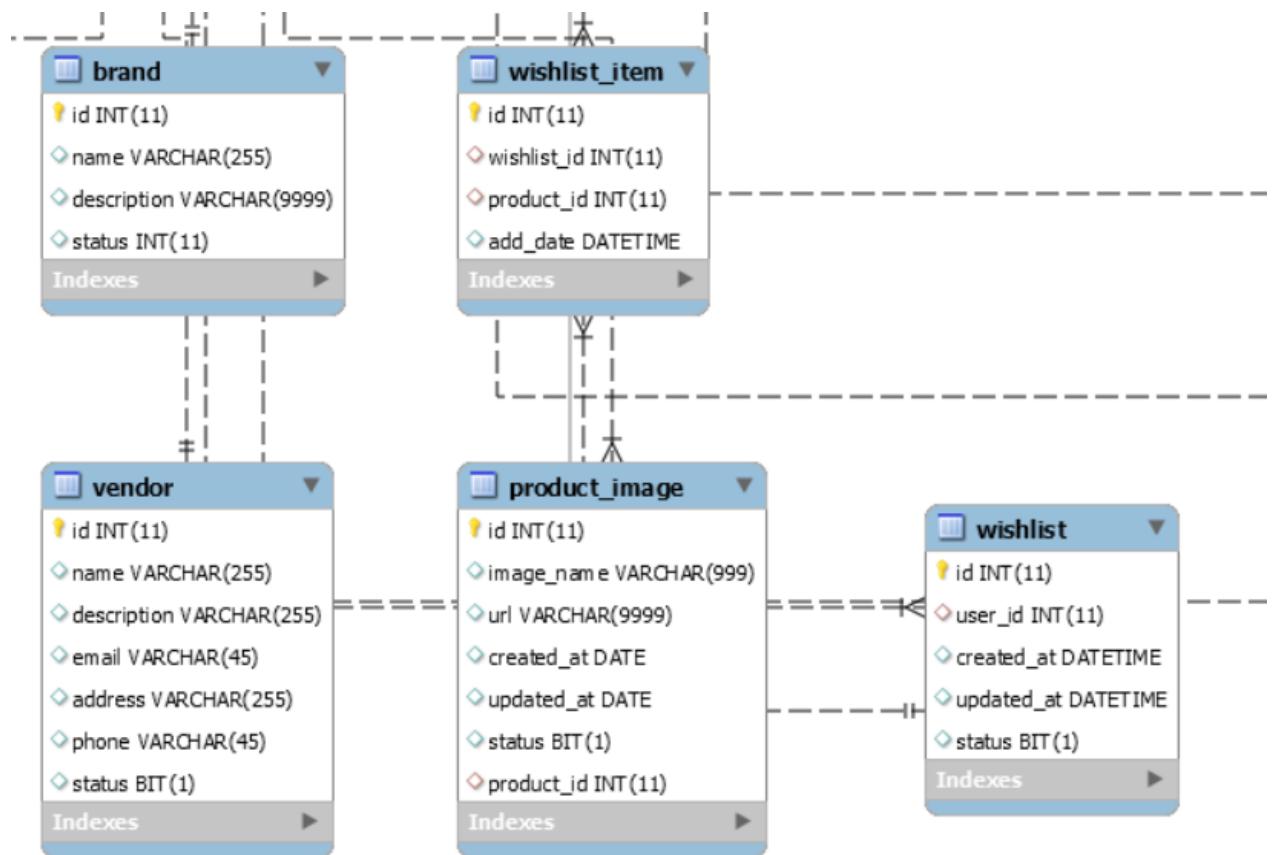
Hình 39 Entity relationship diagram (ERD)



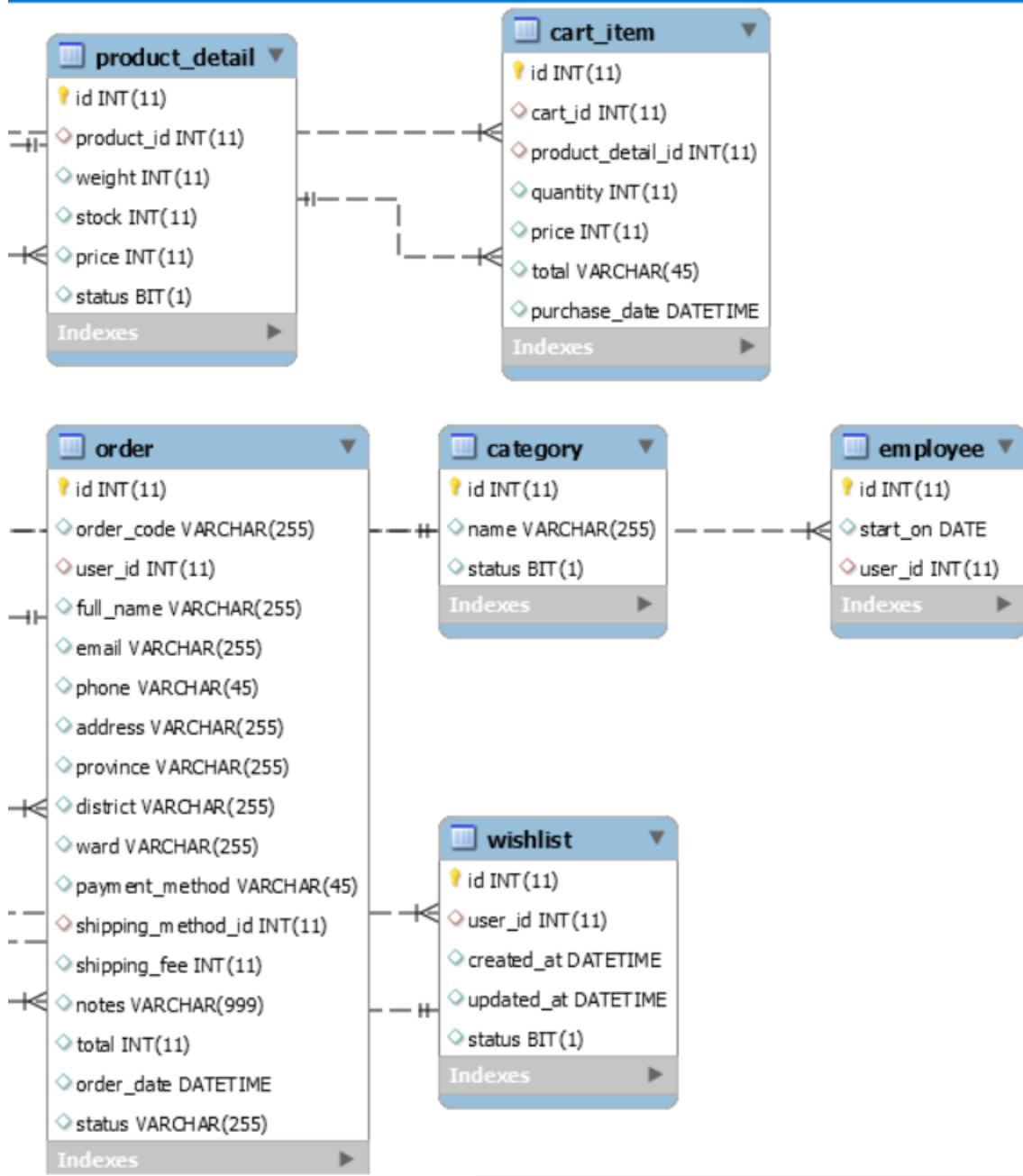
Hình 40 Dữ liệu bảng cart, order_detail, product, user, sale



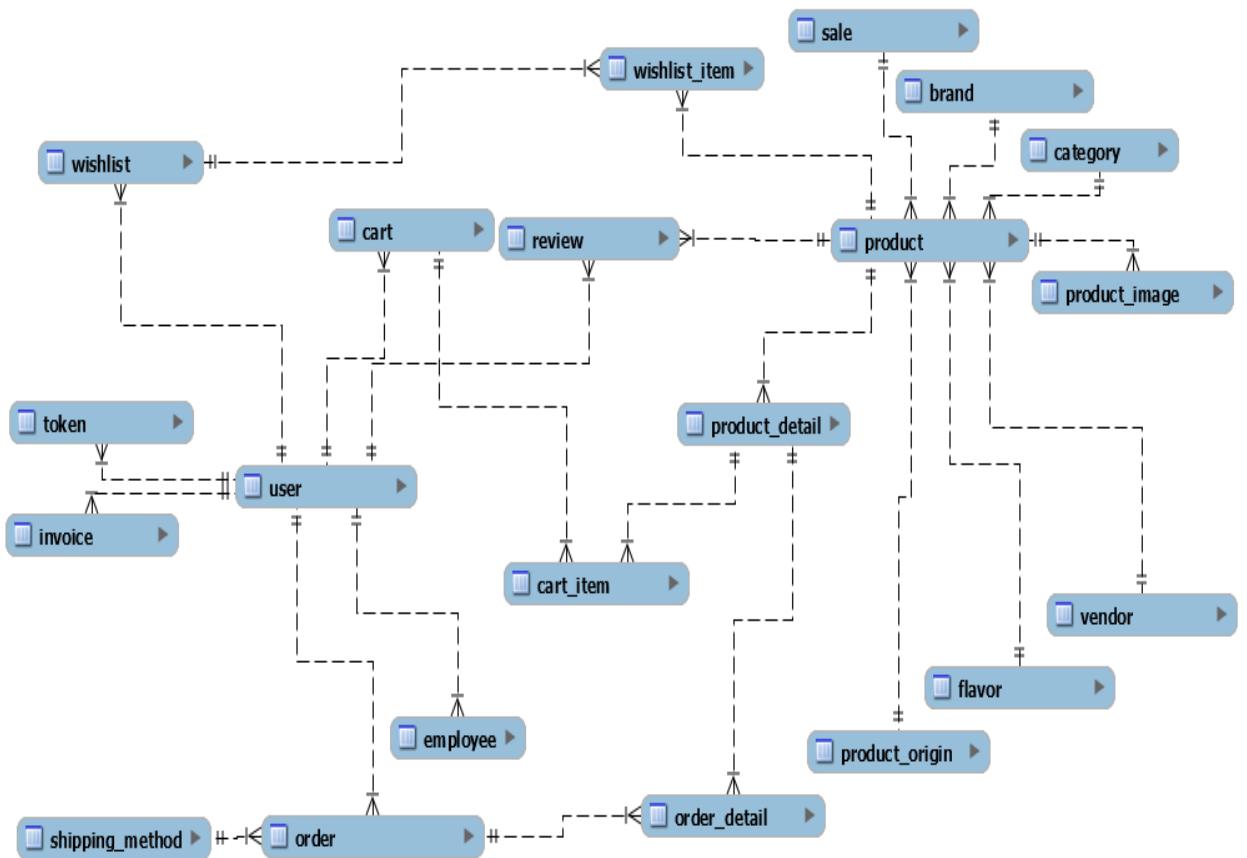
Hình 41 Dữ liệu bảng invoice, shipping-method, review, product_origin, flavor



Hình 42 Dữ liệu của bảng brand, wishlist_item, vendor, product_image, wishlist



Hình 43 Dữ liệu bảng product_detail, cart_item, order, category, employee, wishlist

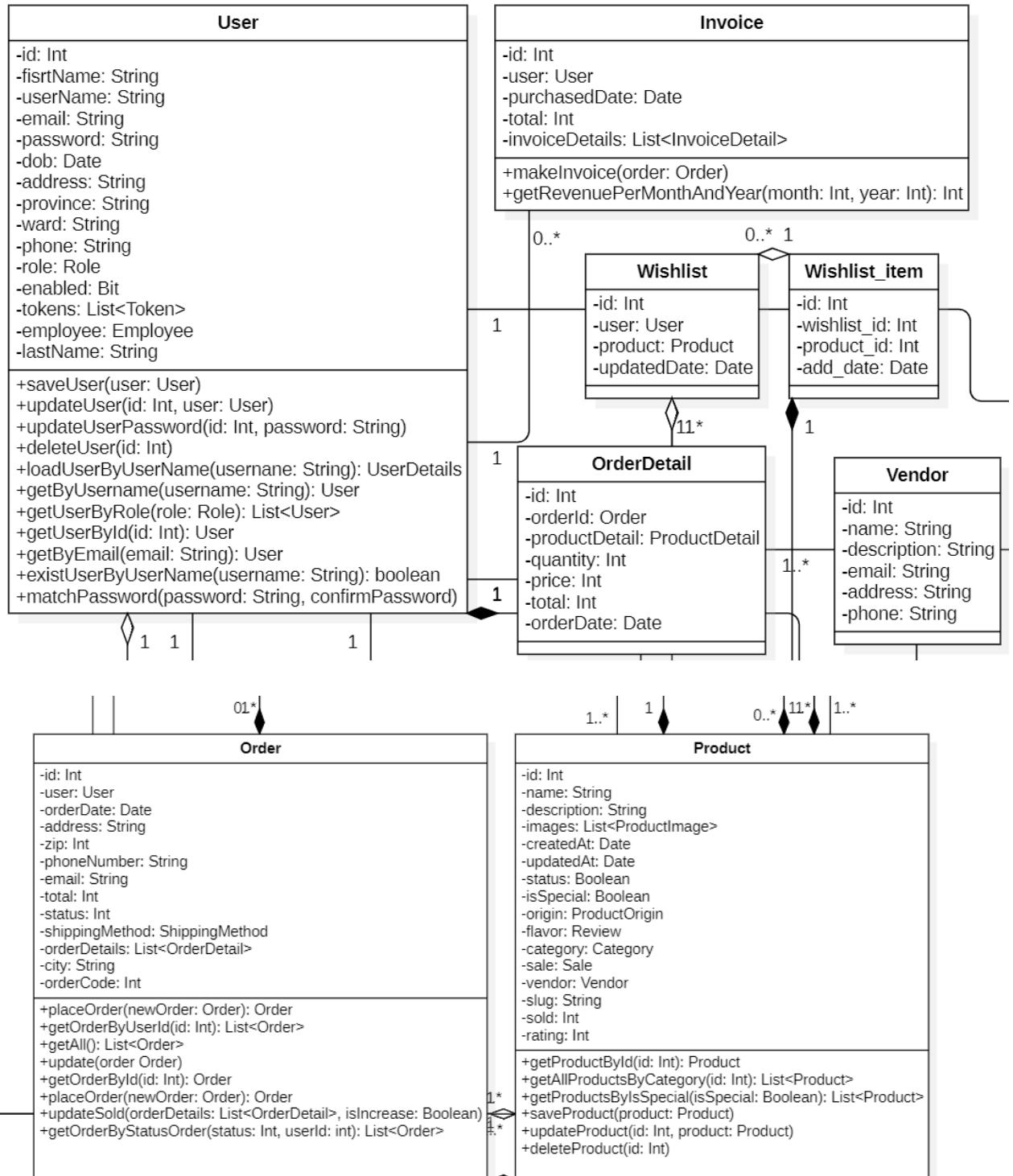


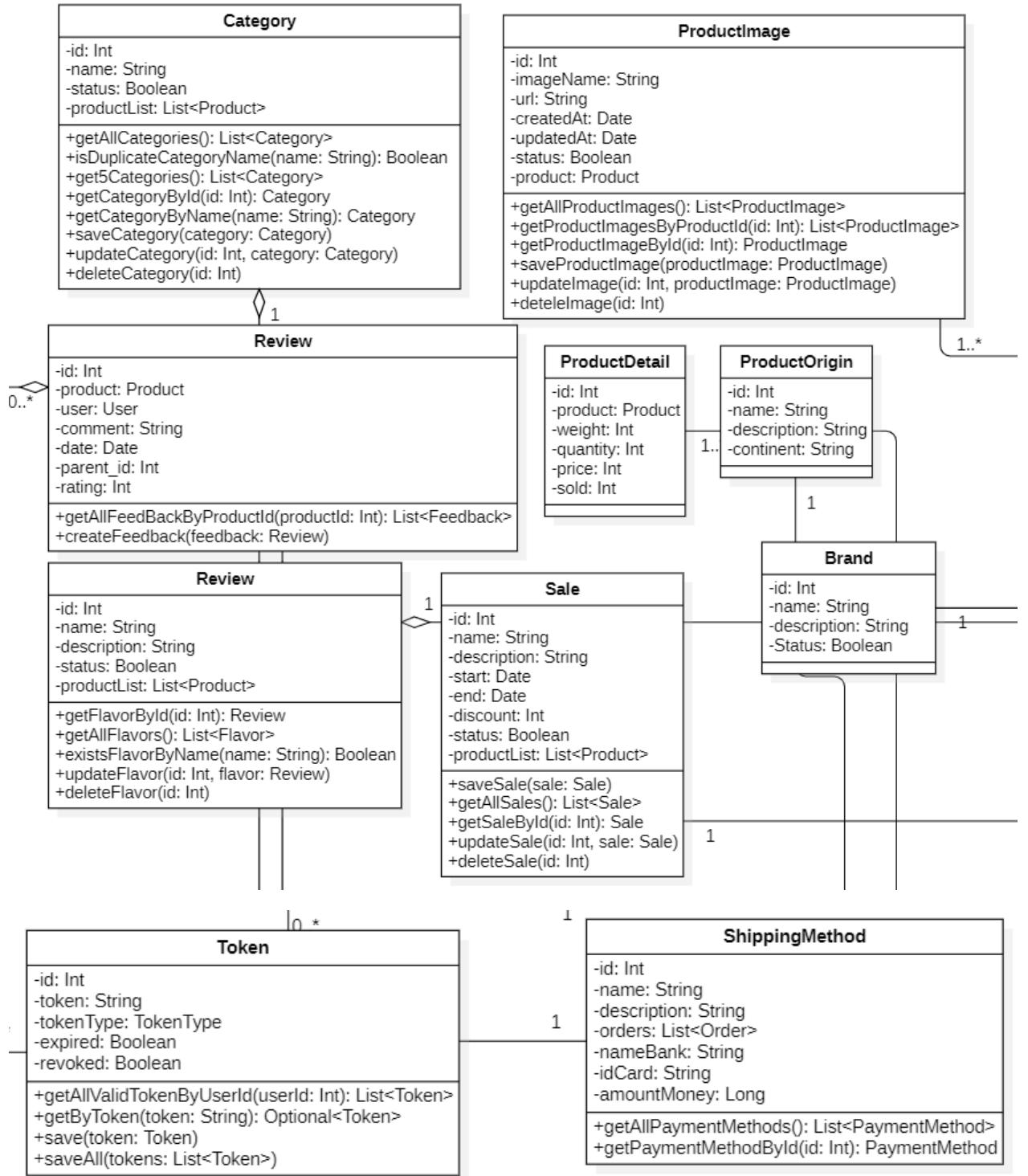
Hình 44 ERD - Mối liên hệ giữa các entity

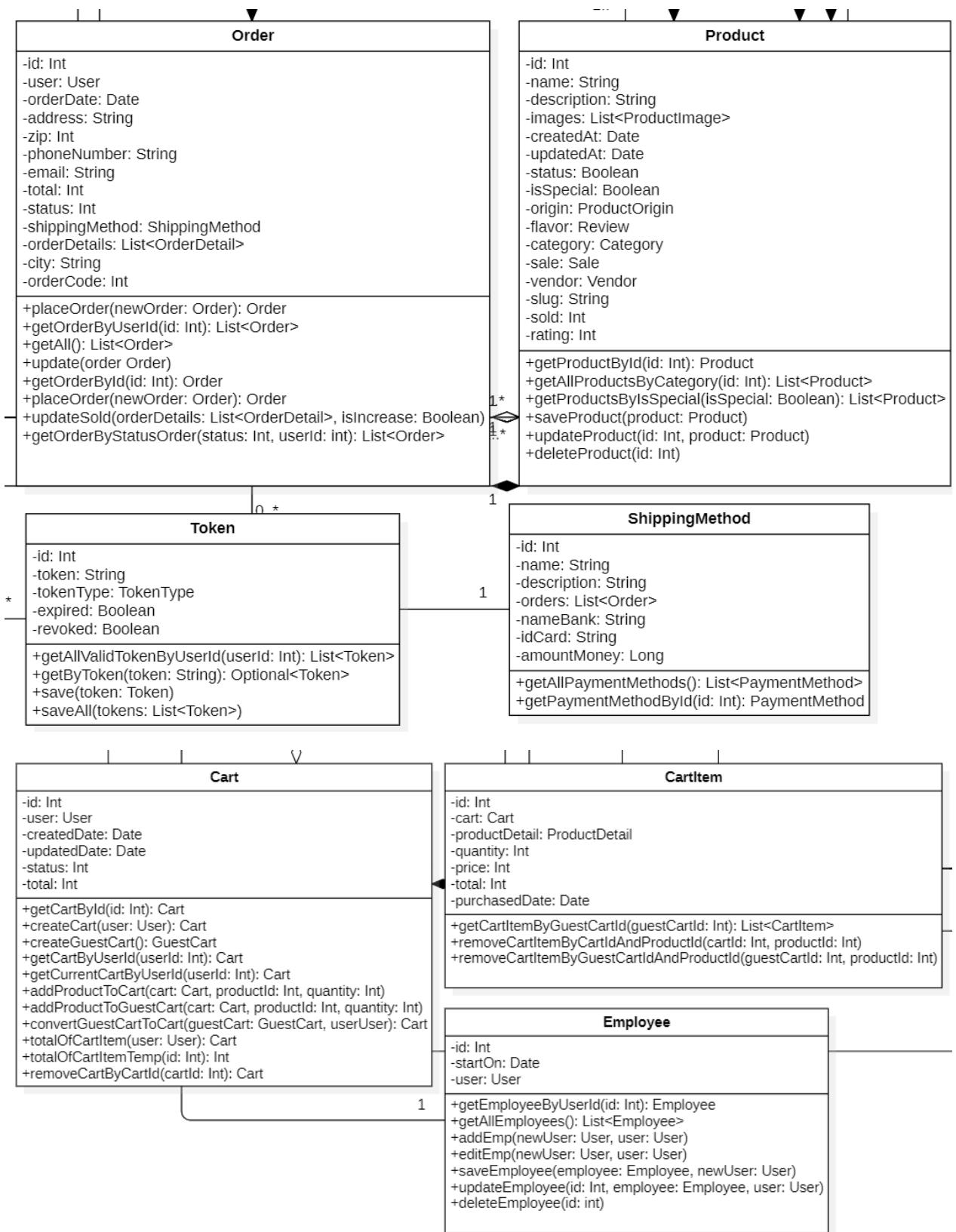
4.1.2. Chi tiết bảng trong database

wishlist(id, user_id, created_at, updated_at, status)
token(id, token, token_type, expired, revoked, user_id)
invoice(id, user_id, invoice_date, total)
shipping_method(id, name, surcharge, duration)
cart(id, user_id, created_at, updated_at, total, status)
user(id, firstname, lastname, username, email, password, dob, address, province, district, ward, phone, role, created_date, updated_date, enabled)
employee(id, start_on, user_id)
order(id, order_code, user_id, full_name, email, phone, address, province, district, ward, payment_method, shipping_method_id, shipping_fee, notes, total, order_date, status)
review(id, user_id, product_id, rating, comment, created_at, parent_id)
cartItem(id, cart_id, product_detail_id, quantity, price, total, purchase_date)
wishlistItem(id, wishlist_id, product_id, add_date)
product_detail(id, product_id, weight, stock, price, status)
order_detail(id, order_id, product_detail_id, quantity, price, total, order_date)
sale(id, name, description, start, end, discount, status)
brand(id, name, description, status)
category(id, name, status)
product_image(id, image_name, url, created_at, updated_at, status, product_id)
product(id, name, slug, brand_id, sold, rating, flavor_id, description, created_at, updated_at, is_special, is_limited, status, sale_id, category_id, product_origin_id, vendor_id)
vendor(id, name, description, email, address, phone, status)
flavor(id, name, description, status)
product_origin(id, name, continent, description, status)

4.1.3. Class diagram



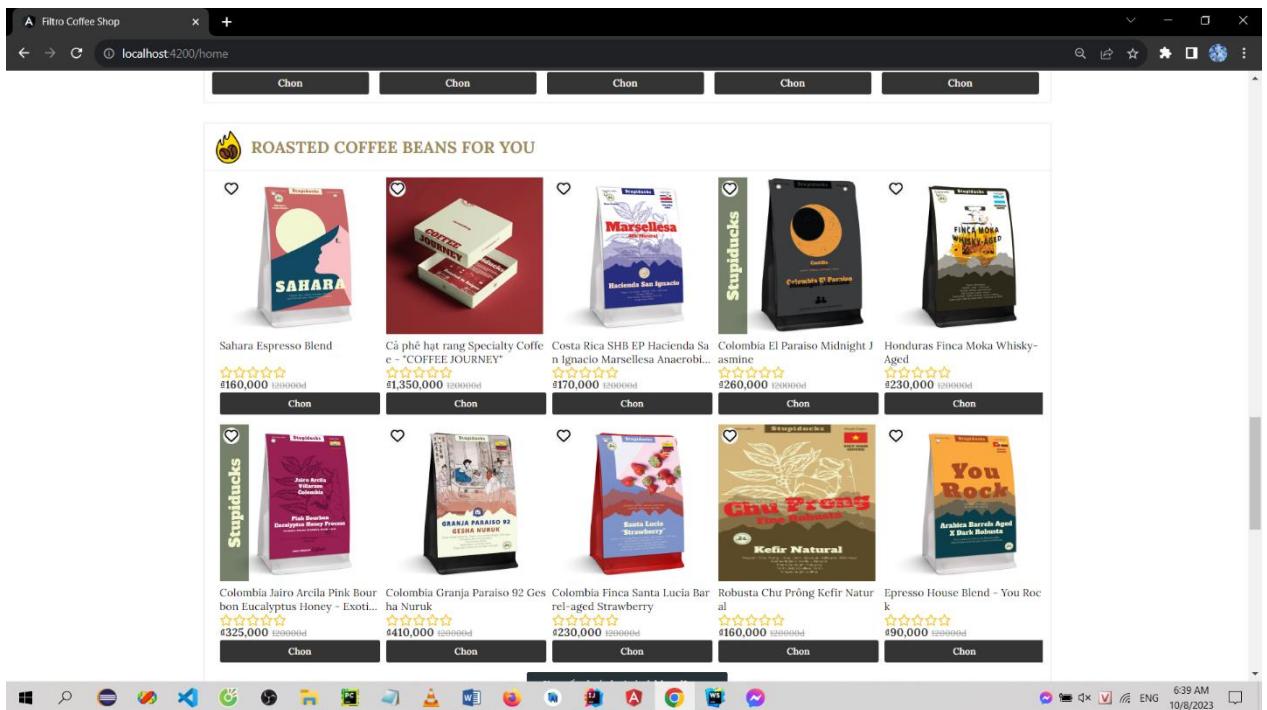




Hình 45 Class Diagram

4.2. GIAO DIỆN VÀ THIẾT KẾ QUY TRÌNH

4.2.5. Trang chủ



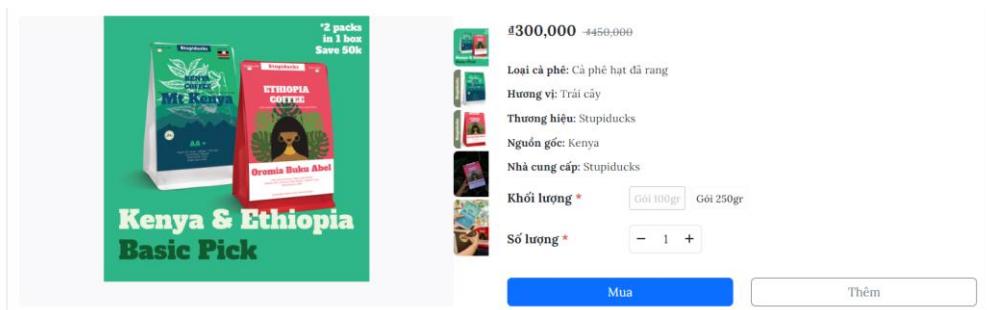
Hình 46 Giao diện trang chủ

4.2.6. Trang danh sách sản phẩm

This page lists products under the category 'HỘP QUÀ TẶNG (SPECIAL BOX)'. It features a sidebar with filters for 'Loại', 'Nguồn gốc', 'Nhà cung cấp', and 'SẢN PHẨM BÁN CHẠY'. The main area shows a large image of coffee beans and a grid of product cards for 'SAHARA Espresso Blend', 'Cà phê hạt rang Specialty Coffee ~ COFFEE JOURNEY~', 'Costa Rica SHB EP Hacienda San Ignacio Marelles Anaerobic...', 'Colombia El Paraiso Midnight Jasmin', and 'Honduras Finca Moka Whisky-Aged'.

Hình 47 Giao diện trang danh sách sản phẩm

4.2.7. Trang chi tiết sản phẩm



Thông tin sản phẩm	Chính sách đổi trả	Hướng dẫn bảo quản	Đánh giá sản phẩm
Mô tả: Cà phê chất lượng cao, ổn định phù hợp với người sử dụng cà phê hàng ngày với yêu cầu tiêu chuẩn cao về cà phê. Kenya Washed & Ethiopia Natural đang trên bàn thử nếm, nếu bạn cần một bình cà phê ngon ổn định hàng ngày thì không thể bỏ qua 2 mẫu cà phê đến từ 2 vùng cà phê nổi tiếng Kenya & Ethiopia. Mẫu Kenya nổi bật về độ chua sáng đi kèm với vị ngọt mong nước còn Ethiopia mang đến sự ngọt ngào hơn với phương pháp sơ chế Natural cùng với các nốt hương hoa phong phú. Cùng thử 2 mẫu này với các dụng cụ pha chế tại nhà có sẵn tại nhà để tìm ra nhiều sự mới mẻ hơn vì mỗi bình cà phê là khác biệt. Trivia: - Cà phê mới rang mỗi ngày với dịch vụ xay miễn phí Bạn có thể chọn mức độ xay theo nhu cầu của bạn. - Đặt hàng ngay, đóng gói ngay ... đặt hàng trước trưa giao hàng trong ngày đặt hàng sau buổi trưa giao hàng ngày hôm sau. - Có dịch vụ viết thiệp để thương. Theo thông điệp mà khách hàng muốn miễn phí! - Cung cấp lời khuyên về các công thức pha chế khác nhau trực tiếp từ nhân viên pha cà phê. - Nếu bạn muốn có đầy đủ hóa đơn thuế Bạn có thể liên hệ với quản trị viên qua trò chuyện (Zalo/ Facebook).			

Hình 48 Giao diện trang chi tiết sản phẩm

4.2.8. Trang Giỏ hàng

Sản phẩm	Khối lượng	Giá tiền	Số lượng	Tổng cộng	Hành động
Colombia Jairo Arcila Pink Bourbon Eucalyptus Honey - Exotics Cofinet	Gói 100 gr	325.000 ₫	- 1 +	325.000 ₫	X
Colombia Granja Paraiso 92 Gesha Nuruk	Gói 100 gr	410.000 ₫	- 1 +	410.000 ₫	X

Coupon Code	Nhập mã coupon
Tổng quan giỏ hàng	
Tạm tính	735.000 ₫
Giảm giá (%)	-----
Tổng cộng	735.000 ₫
Tiến hành thanh toán	

Hình 49 Giao diện trang giỏ hàng

4.2.9. Trang đặt hàng

THANH TOÁN ĐƠN HÀNG

1 Thông tin mua hàng
2 Phương thức vận chuyển và thanh toán
3 Xác nhận thông tin

Phương thức vận chuyển

- Giao hàng chuẩn (nhận từ 5 - 7 ngày) 35.000 ₫
- Giao hàng nhanh (nhận từ 1 - 2 ngày) 50.000 ₫

Phương thức thanh toán

- Thanh toán khi nhận hàng (COD)
- Thanh toán qua ví điện tử Momo

Quay lại
Tiếp tục

Nhập mã coupon

Áp dụng

Tổng quan đơn hàng

	Colombia Jairo Arcila Pi...	325.000 ₫
Gói 100 gr	1 sản phẩm	-
	Colombia Granja Parais...	410.000 ₫
Gói 100 gr	1 sản phẩm	-
Tạm tính		735.000 ₫
Chi phí vận chuyển		-
Giảm giá (%)		-
Voucher		-
Tổng cộng		735.000 ₫

Hình 50 Giao diện trang đặt hàng

duco100

Cà phê filtro xin chào, chúc bạn buổi trưa vui vẻ

o(*^-^*)ゞ (^~^•ゞ {x^*ゞ

*** *** ***

Thông tin cá nhân
Thay đổi mật khẩu

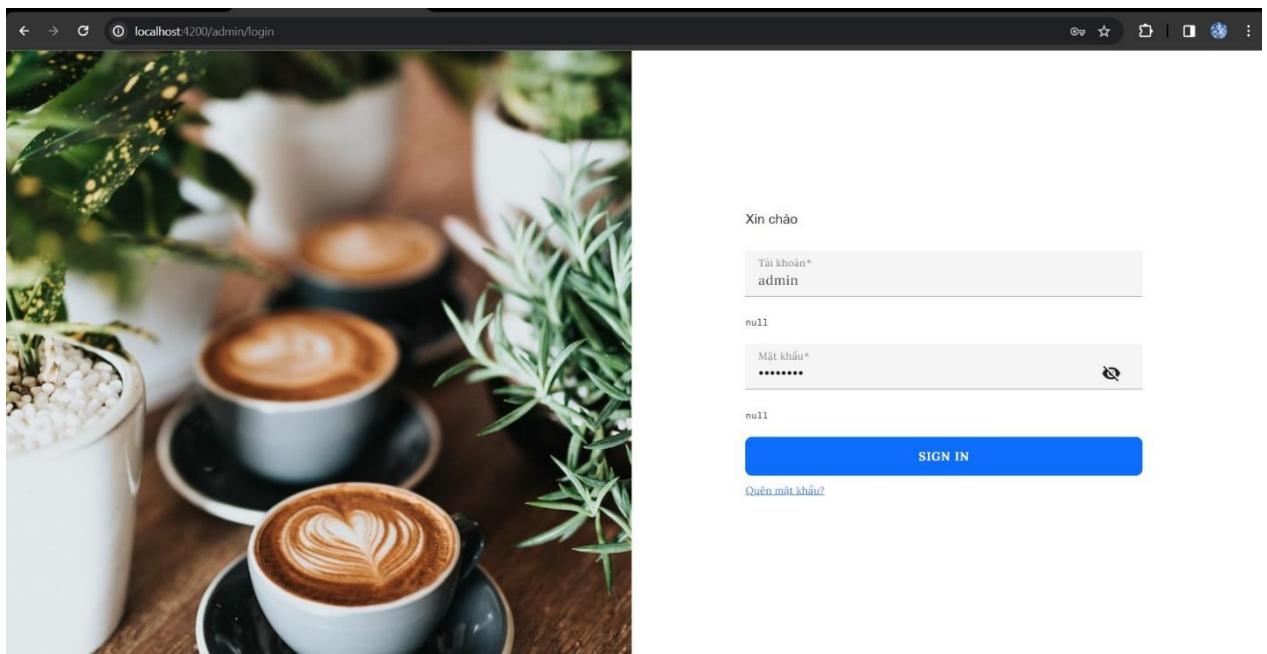
Thay đổi thông tin của bạn

Username duco100	Email voduco100@gmail.com	
Họ Võ ^&U*IO	Tên Vân Đức %^&*	
Ngày sinh	Phone	
DD/MM/YYYY		
Địa chỉ		
Thành phố/Tỉnh	Quận/Huyện	Xã/Phường

Cập nhật hồ sơ

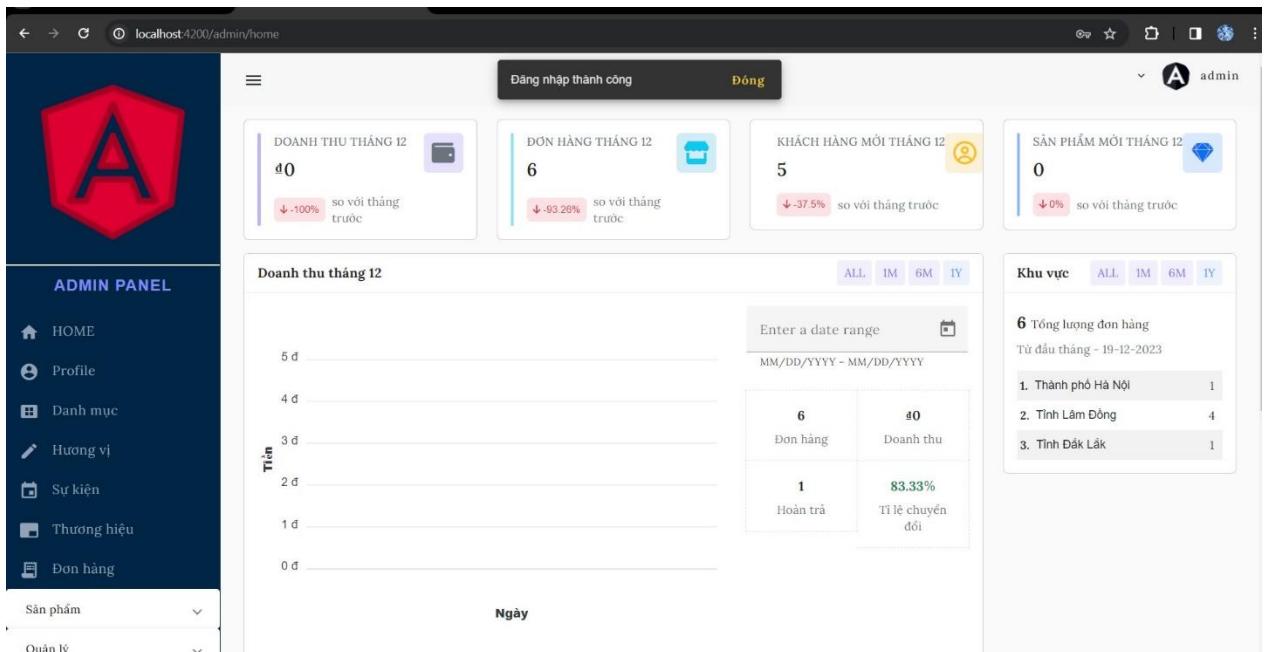
Hình 51 Giao diện trang hồ sơ tài khoản khách hàng

4.2.11. Trang đang nhập quản trị



Hình 52 Giao diện trang đăng nhập quản trị

4.2.12. Trang doanh thu



Hình 53 Giao diện trang doanh thu

4.2.13. Trang quản trị sản phẩm

DANH SÁCH SẢN PHẨM

+ Thêm sản phẩm

Tìm kiếm

ID	Sản phẩm	Thương hiệu	Mô tả	Đã bán	Hương vị	Danh mục	Sự kiện	Ngày thêm	Ngày sửa	Đặc biệt	Giới hạn	Nguồn gốc	Nhà cung cấp	Trạng thái	Hành động
#1	Sahara Espresso Blend	Stupiducks	Cà phê hạt rang Sahara Espresso Blend được được rang phù hợp với phương pháp pha chế Espresso, Aeropress, Moka pot, Protarfilter, Nampresso, Cold brew, Cold drip, Vietnam Phin...	18	Trái cây	Cà phê hạt đã rang	Không	2023-09-16 13:24:14	2023-10-05 15:54:06	Không	Không	Brazil	Stupiducks	Mới	
#2	Cà phê hạt rang Specialty Coffee - 'COFFEE JOURNEY'	Stupiducks	Bao gồm 6 hoặc 12 gói cà phê 100g được sơ chế đặc biệt mang lại hương vị bùn nổ từ các nông trại nổi tiếng.	69	Trung tính	Cà phê hạt đã rang	Không	2023-09-10 20:07:54	2023-10-06 20:39:25	Đáng	Không	Brazil	Stupiducks	Mới	

Hình 54 Giao diện trang quản trị sản phẩm

4.2.23. API giao hàng nhanh khi tạo đơn hàng

Chọn thành phố/Tỉnh

- Lào Cai
- Hưng Yên
- Hòa Bình
- Sơn La
- Điện Biên

Thành phố/Tỉnh*

Hưng Yên

Quận/Huyện*

Xã/Phường*

Hình 55 Giao diện chọn địa điểm giao hàng

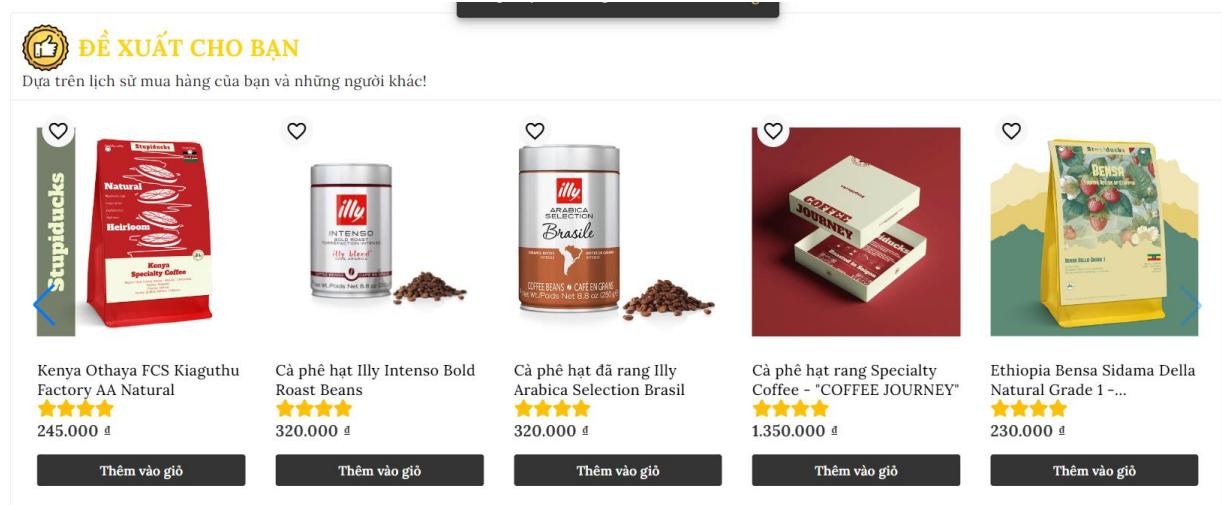
Phương thức vận chuyển

Phí vận chuyển: 57.201 ₫



Hình 56 Giao diện chọn phí vận chuyển

4.2.24. Hệ thống gợi ý sản phẩm



Hình 57 Giao diện phần gợi ý sản phẩm

4.2.25. Chatbot AI



Hình 58 Giao diện chatbot AI

Chương 5: CÀI ĐẶT VÀ KIỂM THỦ

5.1. CHUẨN BỊ

5.1.1. Công cụ

- Intelij IDEA 2022.3.2 cho code backend – Spring boot
- Webstorm 2023.2.1 cho code frontend – Angular
- Mysql 8.0 cho database – SQL
- Github – Dockerhub: để quản lý source code, image
- Docker desktop
- 1 server VPS: Vnahost
- Tên miền: Vnahost
- Chứng chỉ SSL : Let's Encrypt

5.1.2. Công nghệ

- Frontend: Angular
- Backend: Spring Boot
- Database: Sql

5.2. CI/CD

5.2.1. Khái niệm

Continuous Integration/Continuous Deployment, hay là CI/CD là một quá trình quan trọng trong việc phát triển một ứng dụng. Trong quá trình này, mã nguồn sẽ được tích hợp liên tục và được triển khai liên tục, không bị gián đoạn mỗi khi có cập nhật mới, không ảnh hưởng tới trải nghiệm người dùng mà vẫn tối ưu được quy trình phát triển và triển khai ứng dụng.

Như tên gọi, CI là việc tích hợp liên tục mã nguồn vào một kho lưu trữ. Ở đó, ta có thể theo dõi mọi sự thay đổi của mã nguồn, luôn cập nhật được phiên bản mới nhất của mã nguồn, đảm bảo phiên bản của ứng dụng luôn cập nhật ở trạng thái mới nhất. Hơn nữa, việc cập nhật này diễn ra tự động, và dễ dàng hơn trong việc giám sát lỗi hoặc xung đột giữa mã nguồn của các thành viên.

CD chính là triển khai mã nguồn một cách liên tục lên môi trường sản phẩm hoặc môi trường kiểm thử. Quá trình CD luôn đảm bảo được sản phẩm đang được kiểm tra hoặc sử dụng là phiên bản mới nhất. Mọi công việc trong quá trình triển khai đều được tự động thông qua những file cấu hình đã tạo trước đó. Nếu không có CD, nhà phát triển phải triển khai thủ công mã nguồn lên môi trường thực tế.

Ưu điểm và nhược điểm của CI/CD:

Bảng 23 Ưu điểm và nhược điểm của CI/CD

Ưu điểm	Nhược điểm
Phiên bản của mã nguồn được cập nhật liên tục	Đầu tư: mua server VPS, tên miền
Dễ giám sát, phát hiện lỗi, xung đột	Các thành viên trong nhóm phát triển có kiến thức về CI/CD
Triển khai lên môi trường thực tế nhanh chóng	Bảo trì liên tục để phiên bản mới nhất không gây gián đoạn người dùng.
Nhất quán về phiên bản mã nguồn trong quá trình phát triển	

5.2.2. Thực thi

Nhóm sử dụng:

- Sử dụng Github để quản lý phiên bản của mã nguồn.
- Sử dụng Dockerhub để quản lý Image.
- Sử dụng VPS làm server.
- Sử dụng SSL từ Let's Encrypt.

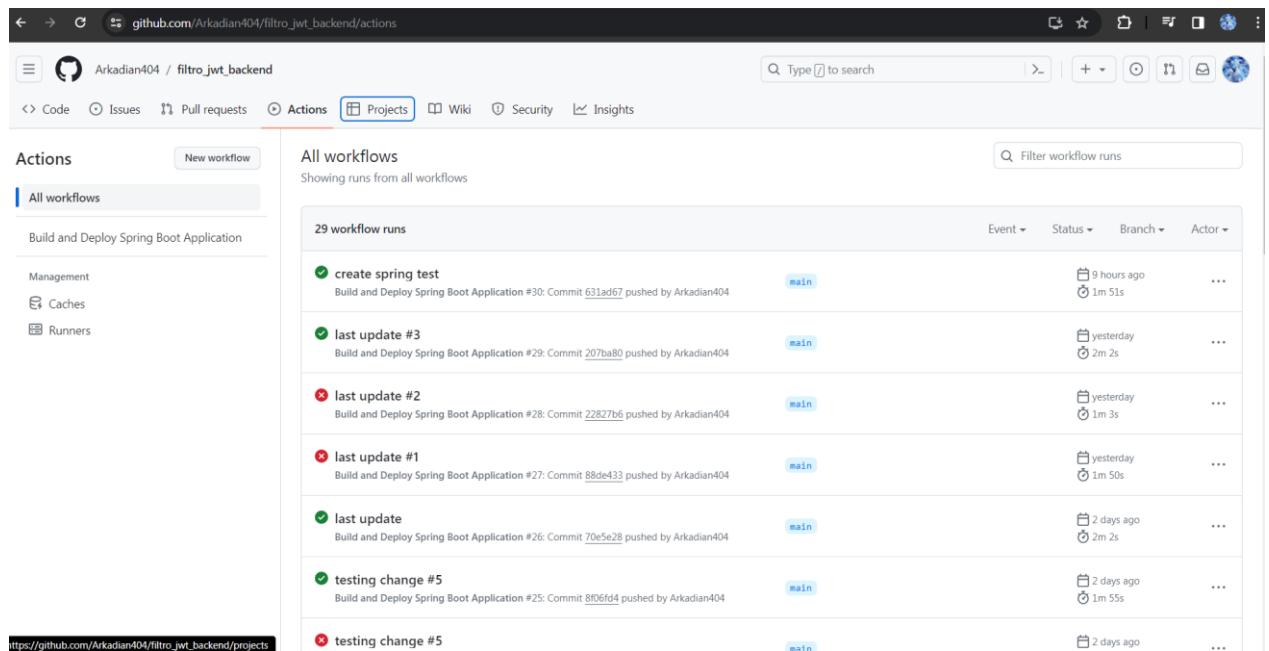
Quá trình CI bao gồm những công đoạn sau:

- Đẩy mã nguồn lên Github
- Kết nối Dockerhub tạo Image từ mã nguồn trong Github
- Tiến hành Unitest và cập nhật mã nguồn

Quá trình CD bao gồm những công đoạn sau:

- Kết nối chứng chỉ SSL (Let's Inscript) với tên miền (Vnahost) thông qua server VPS.
- Sử dụng Nginx để kết nối tên miền và Image
- Sử dụng Docker trong server VPS để tạo container, triển khai ứng dụng lên server.

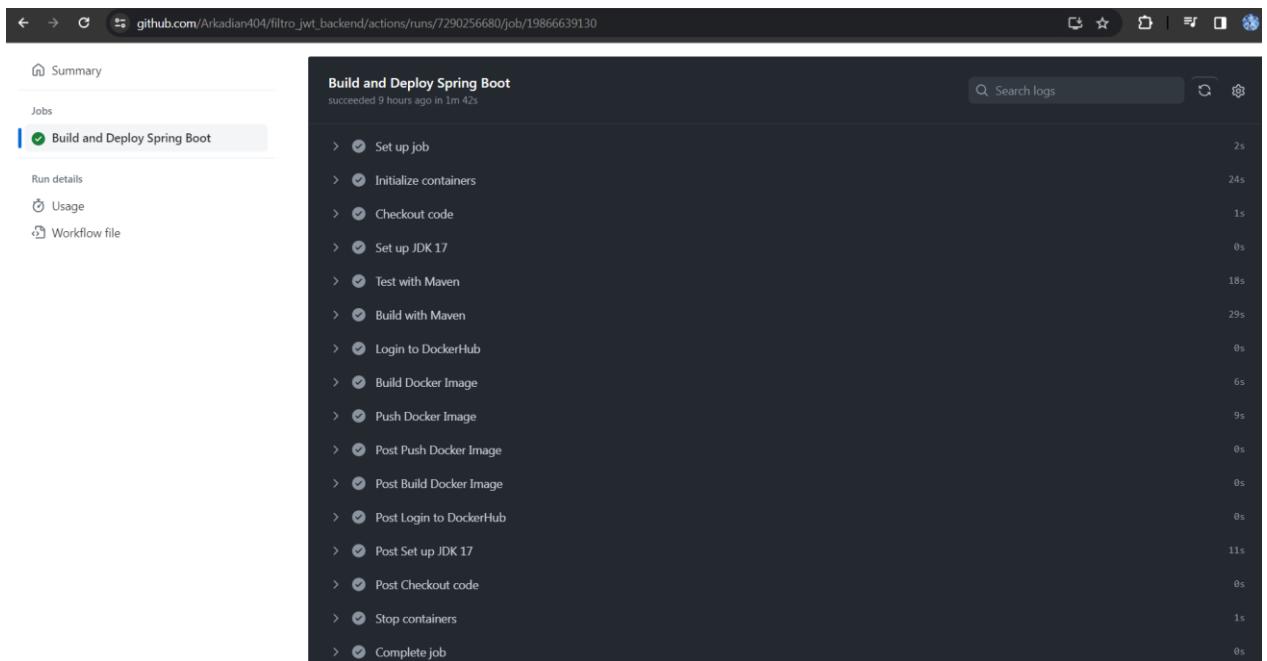
Khi triển khai CI thành công trên Github Action, mỗi khi chúng ta đẩy mã nguồn từ dưới local lên Repository, GitHub Action sẽ tự động thực hiện các giai đoạn của quy trình CI/CD.



The screenshot shows the GitHub Actions interface for the repository "Arkadian404 / filtro_jwt_backend". The "Actions" tab is selected. On the left, there is a sidebar with sections for "Actions", "All workflows", "Build and Deploy Spring Boot Application", "Management", "Caches", and "Runners". The main area displays "All workflows" and "29 workflow runs". Each run is listed with a status icon (green checkmark for successful, red X for failed), the name of the workflow, the commit it was triggered by, the branch it ran on (main), and the time it was completed. For example, the first run is "create spring test" from commit 631ad67, completed 9 hours ago. The last run is "testing change #5" from commit 8ff06fd4, completed 2 days ago.

Hình 59 CI/CD GitHub Action

Trong phần workflows create spring test, ta có thể thấy được GitHub Action đã tự động thực hiện các giai đoạn trong CI:



Hình 60 CI/CD – các giai đoạn của việc build và deploy Spring Boot

Các giai đoạn của CI bao gồm:

Set up job -> Initialize container -> checkout code -> Set up JDK 17 -> Test with Maven -> Build with Maven -> Login to DockerHub -> Build Docker image -> Push Docker image -> Post push Docker image -> Post build Docker Image -> Post Login to DockerHub -> Post Set up JDK 17 -> Post Checkout code -> Stop containers -> Complete job

Các giai đoạn của CD bao gồm:

- Kết nối chứng chỉ SSL và tên miền thông qua server VPS

```
listen      443 ssl http2;
listen      [::]:443 ssl http2;
server_name filtrocoffee.com www.filtrocoffee.com;
ssl_certificate /etc/letsencrypt/live/filtrocoffee.com/fullchain.pem;
ssl_certificate_key /etc/letsencrypt/live/filtrocoffee.com/privkey.pem;
ssl_protocols TLSv1.2 TLSv1.3;
ssl_prefer_server_ciphers on;
```

- Sử dụng Nginx để kết nối tên miền và Image

```
front-end:
```

```
image: zafuog/angular-filtro:latest
```

```
volumes:
```

```
- ./config/default.conf:/etc/nginx/conf.d/default.conf
```

```
location ^~ /springboot/ {  
    proxy_pass http://back-end:8080/;  
    proxy_http_version 1.1;  
}  
  
location ^~ /fastapi/ {  
    proxy_pass http://fast-api:8000/;  
    proxy_http_version 1.1;
```

- Sử dụng docker compose để tạo container từ các image, chạy trên server VPS

```
services:
```

```
docker-mysql:
```

```
image: mysql:8.0
```

```
fast-api:
```

```
image: zafuog/fastapi_filtro
```

```
back-end:
```

```
image: zafuog/springboot_filtro:latest
```

```
front-end:
```

```
image: zafuog/angular-filtro:latest
```

5.3. UNIT TEST

5.3.1. Khái niệm

Quá trình unit test phải được hoạt động độc lập và tự động hóa, hạn chế can thiệp thủ công. Điều này đảm bảo rằng các đơn vị mã nguồn không bị thay đổi mục đích khi mã nguồn bị thay đổi, từ đó đảm bảo tính nhất quán cho toàn bộ chương trình.

5.3.2. Thực thi

Dự án này, backend được viết bằng SpringBoot và frontend được viết bằng Angular.

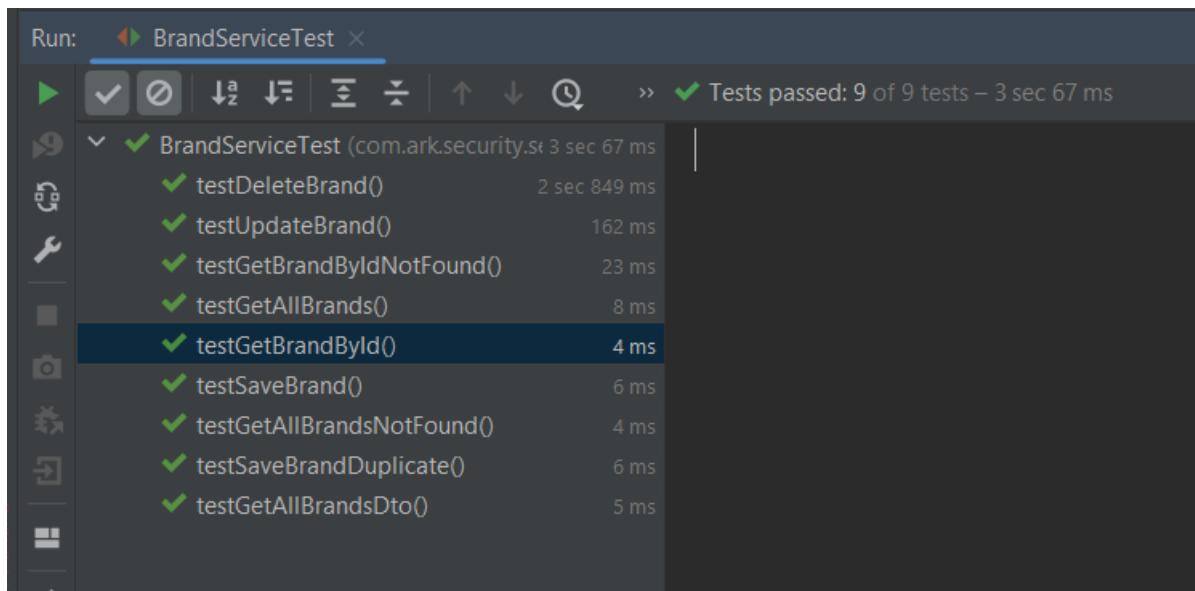
Nhóm sẽ tiến hành unit test một vài đơn vị trong backend bằng thư viện Mockito phiên bản 5.8.0 và thư viện JUnit phiên bản 5.10.0 của Spring Boot. Mockito có tác dụng tạo ra các đối tượng giả mạo và JUnit sử dụng các đối tượng giả này cho quá trình unit test.

Quy trình thực hiện viết các hàm unit test:

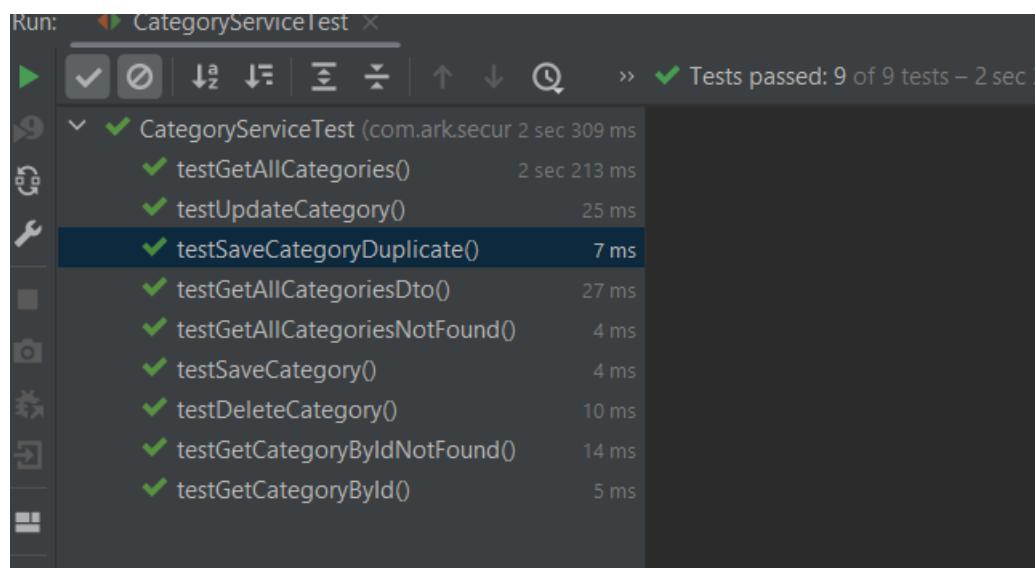
- Tạo ra một phương thức đánh dấu annotation là một testcase
- Tạo ra một đối tượng giả mạo bằng Mockito để kiểm soát hành vi trong quá trình kiểm thử
- Tạo ra dữ liệu giả (ở đây là các class giả) đã được chứa dữ liệu giả cho Mockito tạo ra
- Sử dụng phương thức when để thực thi phương thức cần được kiểm thử
- Sử dụng phương thức verify để kiểm tra kết quả kiểm thử có đáp ứng các yêu cầu cần thiết hay không.
- Sử dụng phương thức assertEquals để so sánh kết quả testcase và kết quả thực tế.

Kết quả thực thi unit test:

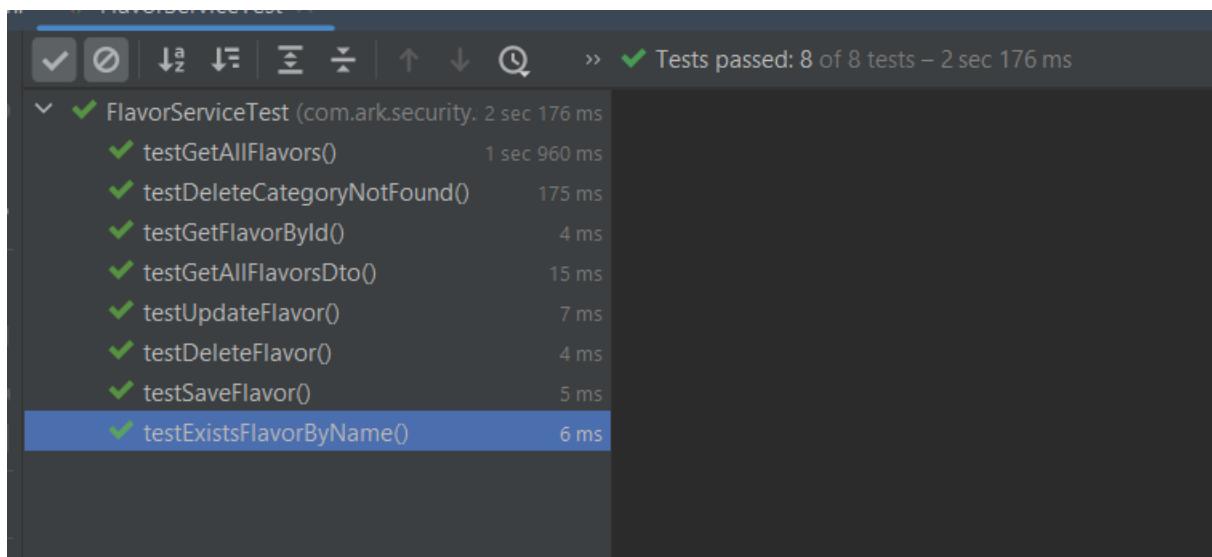
Test service product_brand (thương hiệu):



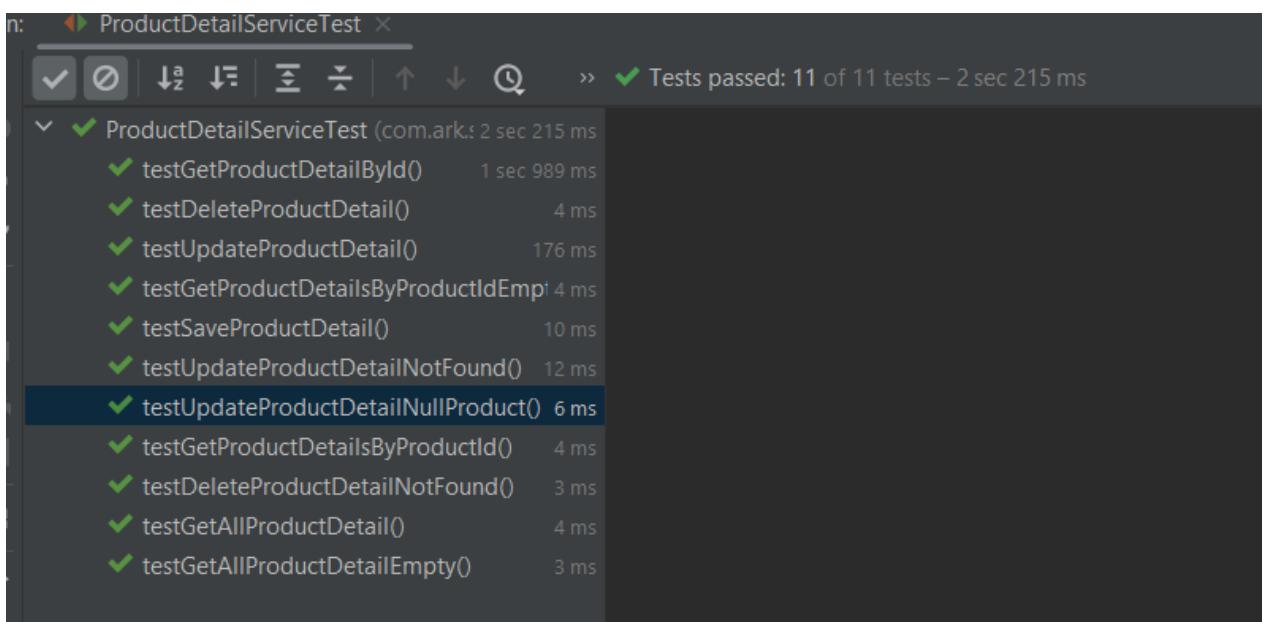
Hình 61 Unit test – brand



Hình 62 Unit test – category



Hình 63 Unit test – flavor



Hình 64 Unit test – product detail

Ngoài ra, unit test được kiểm thử trên các model dữ liệu khác, đã được liệt kê trong chương 4, phần thiết kế database.

KẾT LUẬN

Kết quả đạt được

Dự án website thương mại điện tử kinh doanh cà phê có đặc điểm và chức năng sau:

- Sử dụng công nghệ Angular (Frontend), Spring Boot (Backend), Sql (Database).
- Cung cấp đủ chức năng của một website thương mại điện tử, bao gồm quản lý thông tin liên quan tới sản phẩm, người dùng, đơn hàng, nhân viên.
- Sử dụng API từ Giao Hàng Nhanh Express.
- Tích hợp thanh toán trực tuyến thông qua ví điện tử Momo và cổng thanh toán Vnpay.
- Tích hợp hệ thống đề xuất sản phẩm bằng thuật toán KNN – K Nearest Neighbors thông qua thư viện Surprise.
- Tích hợp hệ thống Chatbot AI với Model LLM của OpenAI thông qua thư viện LangChain, hỗ trợ người dùng tra cứu thông tin nhanh chóng.
- Triển khai CI/CD cho dự án, tăng tốc độ xử lý của website và dễ dàng bảo trì.

Ưu điểm

- Đa dạng phương thức thanh toán (COD, Momo, Vnpay).
- Sản phẩm được đề xuất phù hợp với sở thích của người dùng.
- Thông tin phản hồi từ ChatbotAI chính xác với dữ liệu của hệ thống.
- Dễ dàng bảo trì và phát triển tính năng của hệ thống theo mô hình CI/CD.

Nhược điểm

- Hệ thống Chatbot AI phản hồi chậm.
- Hệ thống CI/CD chưa hoàn chỉnh, cần tải Image thủ công trong máy chủ.
- Website chưa được kiểm thử về hiệu suất khi số lượng yêu cầu cao (request).

Hướng phát triển

- Mở rộng ngân hàng câu hỏi nhằm tăng độ chính xác của phản hồi và tốc độ phản hồi từ Chatbot AI.
- Cấu hình tải Image tự động từ Dockerhub trong máy chủ VPS mỗi khi cập nhật.
- Sử dụng Google PageSpeed Insights hoặc Gtmetrix để kiểm tra tốc độ Website.
- Thử nghiệm hệ thống để xuất sản phẩm với các thuật toán trong phương pháp Hybrid Filtering như Weighted Hybrid, Switching Hybrid, Feature Augmentation.

KẾ HOẠCH THỰC HIỆN

Bảng 24 Bảng phân công công việc

STT	Thời gian	Công việc	Ghi chú
1	20/08/2023 – 01/09/2023	Cấu hình Spring Security	Phú
2	03/09/2023 – 28/09/2023	Viết mã nguồn cho trang quản trị khách hàng, sản phẩm chi tiết, hình ảnh sản phẩm, hương vị, thương hiệu, đăng nhập trang admin.	Phú
3	03/09/2023 – 28/09/2023	Viết mã nguồn cho trang quản trị sản phẩm, danh mục, nhà cung cấp, nguồn gốc, nhân viên.	Đức
4	30/09/2023 – 25/10/2023	Viết mã nguồn cho trang chủ, tra cứu sản phẩm, đặt hàng, bình luận sản phẩm, lấy lại mật khẩu.	Phú
5	30/09/2023 – 25/10/2023	Viết mã nguồn cho trang chi tiết sản phẩm, giỏ hàng, sản phẩm yêu thích, đăng nhập, đăng ký.	Đức
6	28/10/2023 – 28/11/2023	Tích hợp thanh toán qua ví điện tử Momo	Phú
7	28/10/2023 – 28/11/2023	Tích hợp thanh toán qua cổng thanh toán Vnpay.	Đức
8	01/12/2023 – 15/12/2023	Unit test	Phú
9	01/12/2023 – 15/12/2023	CI	Đức
10	15/01/2024 – 30/01/2024	Viết mã bổ sung các sự kiện giảm giá, voucher.	Phú
11	01/02/2024 – 15/02/2024	Bổ sung phần giao hàng nhanh, lấy API từ GHN – Giao hàng nhanh toàn quốc.	Đức

12	17/02/2024 – 23/03/2024	Hệ thống gợi ý sản phẩm dựa trên hành vi người dùng – thuật toán K-Nearest Neighbor KNN.	<ul style="list-style-type: none"> • Đức <ul style="list-style-type: none"> ◦ Tiền xử lý dữ liệu ◦ Xác thực mô hình ◦ Chọn K ◦ Xây dựng hàm • Phú <ul style="list-style-type: none"> ◦ Xây dựng hàm (chính) ◦ Xây dựng UI/UX
13	26/03/2024 – 29/04/2024	Chatbot AI với Model LLM của Open AI.	<ul style="list-style-type: none"> • Đức <ul style="list-style-type: none"> ◦ Kiểm tra với Model LLM của Hugging Face ◦ Viết Prompt cho Chatbot ◦ Xây dựng hàm • Phú <ul style="list-style-type: none"> ◦ Xây dựng hàm (chính) ◦ Xây dựng UI/UX
14	05/05/2024 – 26/06/2024	CD	Phú
15	15/06/2024 – 01/07/2024	Viết báo cáo	Đức

DANH MỤC TÀI LIỆU THAM KHẢO

- [1] Siêu thị cà phê. "Trang chủ siêu thị cà phê", 2023. [Trực tuyến]. Địa chỉ: <https://sieuthicafe.vn/> [Truy cập 15/9/2023].
- [2] Trung Nguyên. "Trung Nguyên E-Coffee - Hệ thống cửa hàng chuyên Cà phê Năng lượng", 2020. [Trực tuyến]. Địa chỉ: <https://trungnguyenecoffee.com/> [Truy cập 15/9/2023].
- [3] Cà phê nguyên chất. "Trang chủ cà phê nguyên chất", 2023. [Trực tuyến]. Địa chỉ: <https://caphenguyenchat.vn/> [Truy cập ngày 15/9/2023].
- [4] iViettech. "2. Bản vẽ Use Case (Use Case Diagram)", 29/06/2014. [Trực tuyến]. Địa chỉ: <https://iviettech.vn/blog/543-ban-ve-use-case-use-case-diagram.html> [Truy cập 10/12/2023].
- [5] iViettech. "9. Thực hành xây dựng Sequence Diagram", 22/06/2014. [Trực tuyến]. Địa chỉ: <https://iviettech.vn/blog/1358-thuc-hanh-xay-dung-sequence-diagram.html> [Truy cập 14/10/2023].
- [6] Google LLC. "Angular Material", 2023. [Trực tuyến]. Địa chỉ: <https://material.angular.io/> [Truy cập 25/9/2023].
- [7] Tutorialspoint. "Spring Boot Tutorial", 2023. [Trực tuyến]. Địa chỉ: https://www.tutorialspoint.com/spring_boot/index.htm [Truy cập 28/9/2023].
- [8] Duck Mart. "Trang chủ Coffee Store", 2023. [Trực tuyến]. Địa chỉ: <https://www.coffeestore.vn/> [Truy cập 15/9/2023].
- [9] LangChain, Inc. "LangChain Tutorials", 2024. [Trực tuyến]. Địa chỉ: https://python.langchain.com/v0.1/docs/additional_resources/tutorials/ [Truy cập 26/03/2024].
- [10] Nicolas Hug. "Surprise - Getting Started", 2015. [Trực tuyến]. Địa chỉ: https://surprise.readthedocs.io/en/stable/getting_started.html [Truy cập 17/02/2024].
- [11] GeeksforGeeks. "Develop an LLM Application using Openai", 06/01/2024. [Trực tuyến]. Địa chỉ: <https://www.geeksforgeeks.org/develop-an-llm-application-using-openai/> [Truy cập 09/01/2024].
- [12] GeeksforGeeks. "Recommender Systems using KNN", 11/01/2024. [Trực tuyến]. Địa chỉ: <https://www.geeksforgeeks.org/recommender-systems-using-knn/> [Truy cập 25/01/2024].
- [13] Ayush Garg. "Comparing different types of top-N recommendation engines on a single numerical scale", 08/01/2021. [Trực tuyến]. Địa chỉ: <https://ayushgarg-64289.medium.com/comparing-different-types-of-top-n-recommendation-engines-on-a-single-numerical-scale-a0c883ae1f0a> [Truy cập 05/02/2024].
- [14] Peterroelants. "ReAct REPL Agent", 2023. [Trực tuyến]. Địa chỉ: <https://peterroelants.github.io/posts/react-repl-agent/> [Truy cập 15/04/2024].

- [15] Wikipedia. "Cosine similarity", 2024. [Trực tuyến]. Địa chỉ: https://en.wikipedia.org/wiki/Cosine_similarity#Definition [Truy cập 08/04/2024].
- [16] Datacamp. "How to Build a Chatbot Using the OpenAI API & Pinecone", 01/2024. [Trực tuyến]. Địa chỉ: How to Build a Chatbot Using the OpenAI API & Pinecone [Truy cập 27/03/2024].
- [17] Buomsoo Kim. "Recommender systems with Python - (9) Memory-based collaborative filtering - 6 (k-NN with Surprise)", 06/09/2020. [Trực tuyến]. Địa chỉ: <https://buomsoo-kim.github.io/recommender%20systems/2020/09/07/Recommender-systems-collab-filtering-9.md> [Truy cập 27/02/2024].
- [18] Rabbitmetrics. "LangChain Agents: Simply Explained!", 26/05/2023. [Trực tuyến]. Địa chỉ: <https://www.youtube.com/watch?v=Xi9Ui-9qcPw&t=40s> [Truy cập 01/04/2024].
- [19] AWS. "What is LangChain?", 2023. [Trực tuyến]. Địa chỉ: <https://aws.amazon.com/vi/what-is/langchain/> [Truy cập 01/04/2024]
- [20] LangChain, Inc. "Quickstart", 2024. [Trực tuyến]. Địa chỉ: https://python.langchain.com/v0.1/docs/use_cases/sql/quickstart/ [Truy cập 03/04/2024].

PHỤ LỤC

Bảng 25 Danh sách những từ tiếng anh được sử dụng

Tiếng anh	Tiếng việt
Actor	Tác nhân
Guest	Khách vãng lai
User	Khách hàng
Employee	Nhân viên
Admin	Chủ kinh doanh
Cart	Giỏ hàng
Order_detail	Chi tiết đơn hàng
Product	Sản phẩm
Cart_item	Chi tiết giỏ hàng
Product_detail	Chi tiết sản phẩm
Shipping_method	Phương thức vận chuyển
Sale	Sự kiện
Token	Chuỗi
Order	Đơn hàng
Category	Danh mục
Invoice	Hóa đơn
Brand	Thương hiệu
Wishlist_item	Sản phẩm yêu thích
Wishlist	Danh sách sản phẩm yêu thích
End User	Người dùng cuối