

HỌ VÀ TÊN SINH VIÊN: NGUYỄN VĂN HÙNG

LỚP: D19CNPM07

BỘ THÔNG TIN VÀ TRUYỀN THÔNG
HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG



ĐỒ ÁN
TỐT NGHIỆP ĐẠI HỌC

Đề tài: “XÂY DỰNG TRANG WEB TRAO ĐỔI MẶT HÀNG THỜI TRANG”

Người hướng dẫn: TS. NGUYỄN QUÝ SỸ
Sinh viên thực hiện: NGUYỄN VĂN HÙNG
Mã sinh viên: B19DCCN301
Lớp: D19CNPM07
Khóa: 2019-2024
Hệ: Đại học chính quy

HÀ NỘI - 2023

BỘ THÔNG TIN VÀ TRUYỀN THÔNG
HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG



ĐỒ ÁN
TỐT NGHIỆP ĐẠI HỌC

Đề tài: “XÂY DỰNG TRANG WEB TRAO ĐỔI MẶT HÀNG THỜI TRANG”

Người hướng dẫn:

TS. NGUYỄN QUÝ SỸ

Sinh viên thực hiện:

NGUYỄN VĂN HÙNG

Mã sinh viên:

B19DCCN301

Lớp:

D19CNPM07

Khóa:

2019-2024

Hệ:

Đại học chính quy

MỤC LỤC

MỤC LỤC.....	i
LỜI CẢM ƠN.....	iv
DANH MỤC BẢNG BIỂU.....	v
DANH MỤC HÌNH VẼ.....	vi
DANH MỤC CÁC TỪ VÀ THUẬT NGỮ VIẾT TẮT.....	viii
MỞ ĐẦU.....	1
CHƯƠNG 1: GIỚI THIỆU CHUNG.....	2
1.1. Mục tiêu và ý nghĩa của đồ án.....	2
1.1.1. Hiện trạng.....	2
1.1.2. Mục tiêu và ý nghĩa của đồ án.....	2
1.2. Khảo sát các hệ thống tương tự.....	3
1.2.1. Bunz trade zone.....	3
1.2.2. Các hội nhóm trên Facebook.....	4
1.3. Xác định yêu cầu của hệ thống.....	6
1.4. Kết luận chương.....	7
CHƯƠNG 2: CƠ SỞ LÝ THUYẾT.....	8
2.1. Giới thiệu React.....	8
2.2. Giới thiệu Spring Boot.....	10
2.3. Giới thiệu cơ sở dữ liệu MySQL.....	12
2.4. Giới thiệu về YOLO.....	13
2.4.1. Bài toán nhận diện đối tượng.....	13
2.4.2. Các mô hình có thể áp dụng cho bài toán.....	13
2.4.3. Giới thiệu về YOLO.....	15
2.4.4. Các công cụ cần dùng.....	20
2.5. Kết luận chương.....	20
CHƯƠNG 3: PHÂN TÍCH VÀ THIẾT KẾ.....	21
3.1. Xác định chức năng.....	21
3.1.1. Use case tổng quan.....	21

3.1.2. Use case Đăng nhập.....	24
3.1.3. Use case Xem chi tiết bài đăng.....	24
3.1.4. Use case Quản lý bài đăng.....	25
3.1.5. Use case Chat.....	26
3.1.6. Use case Yêu cầu trao đổi.....	27
3.1.7. Use case Xử lý yêu cầu trao đổi.....	28
3.1.8. Use case Xem lịch sử trao đổi.....	28
3.1.9. Use case Xem lịch hẹn.....	29
3.1.10. Use case Kiểm duyệt.....	29
3.2. Xây dựng kịch bản cho các chức năng.....	30
3.2.1. Kịch bản chính cho Use case Đăng nhập.....	30
3.2.2. Kịch bản chính cho Use case Xem chi tiết bài đăng.....	31
3.2.3. Kịch bản chính cho Use case Quản lý bài đăng - Tạo bài đăng.....	32
3.2.4. Kịch bản chính cho Use case Chat.....	34
3.2.5. Kịch bản chính cho Use case Yêu cầu trao đổi.....	35
3.2.6. Kịch bản chính cho Use case Xử lý yêu cầu trao đổi - Xác nhận.....	36
3.2.7. Kịch bản chính cho Use case Xem lịch sử trao đổi.....	37
3.2.8. Kịch bản chính cho Use case Xem lịch hẹn.....	38
3.2.9. Kịch bản chính cho Use case Kiểm duyệt bài đăng thủ công.....	39
3.2.10. Kịch bản chính cho Use case Kiểm duyệt bài đăng tự động.....	41
3.3. Biểu đồ lớp phân tích.....	42
3.4. Thiết kế biểu đồ tuần tự.....	43
3.4.1. Đăng nhập.....	43
3.4.2. Xem chi tiết bài đăng.....	44
3.4.3. Quản lý bài đăng - Tạo bài đăng.....	45
3.4.4. Chat.....	46
3.4.5. Tạo yêu cầu trao đổi.....	47
3.4.6. Xử lý yêu cầu trao đổi - Xác nhận.....	48
3.4.7. Xem lịch sử trao đổi.....	48
3.4.8. Xem lịch hẹn.....	49
3.4.9. Kiểm duyệt bài thủ công.....	50

3.4.10. Kiểm duyệt bài tự động.....	51
3.5. Thiết kế cơ sở dữ liệu.....	51
3.5.1. Các thông tin cần quản lý.....	51
3.5.2. Đề xuất các lớp thực thể thành các bảng trong cơ sở dữ liệu.....	52
3.5.3. Quan hệ giữa các bảng.....	53
3.6. Chức năng kiểm duyệt tự động.....	54
3.6.1. Dữ liệu.....	54
3.6.2. Huấn luyện mô hình.....	56
3.6.3. Đánh giá mô hình.....	58
3.7. Kết luận chương.....	61
CHƯƠNG 4: TRIỂN KHAI VÀ THỬ NGHIỆM.....	62
4.1. Cài đặt ứng dụng Spring Boot.....	62
4.2. Cài đặt ứng dụng ReactJS.....	63
4.3. Cài đặt ứng dụng Flask.....	64
4.4. Các giao diện.....	64
KẾT LUẬN.....	71
TÀI LIỆU THAM KHẢO.....	72

LỜI CẢM ƠN

Em xin gửi lời cảm ơn chân thành đến TS. Nguyễn Quý Sỹ đã tận tình hướng dẫn, chỉ bảo em trong suốt quá trình hoàn thành đồ án tốt nghiệp này.

Em cũng xin cảm ơn các thầy cô của khoa Công nghệ thông tin trong thời gian qua đã tận tình truyền đạt những kiến thức quan trọng về Công nghệ thông tin để từ đó xây dựng nền móng cho một thế hệ kỹ sư Công nghệ thông tin mới. Công sức của thầy cô đem lại ý nghĩa rất lớn đối với sinh viên Công nghệ thông tin của Học viện Công nghệ Bưu chính Viễn thông nói chung và đối với em nói riêng.

Cuối cùng, em cũng muốn gửi lời cảm ơn đến gia đình, bạn bè. Đó là những người đã luôn quan tâm, động viên em trong suốt quá trình học tập.

Do kiến thức và kinh nghiệm còn hạn hẹp nên Đồ án của em vẫn không tránh khỏi những thiếu sót. Em rất mong nhận được sự góp ý của các thầy cô để sản phẩm có thể hoàn thiện hơn.

Em xin chân thành cảm ơn!

Hà nội, tháng 12 năm 2023
Sinh viên thực hiện

Nguyễn Văn Hùng

DANH MỤC BẢNG BIỂU

Bảng 2.1: So sánh các biến thể của YOLOv8 [3].....	18
Bảng 3.1: Kịch bản cho Use case Đăng nhập.....	31
Bảng 3.2: Kịch bản cho Use Case Xem chi tiết bài đăng.....	32
Bảng 3.3: Kịch bản Use Case Tạo bài đăng.....	34
Bảng 3.4: Kịch bản chính cho Use case Chat.....	34
Bảng 3.5: Kịch bản Use Case Yêu cầu trao đổi.....	36
Bảng 3.6: Kịch bản Use Case Xác nhận trao đổi.....	37
Bảng 3.7: Kịch bản Use Case Xem lịch sử trao đổi.....	38
Bảng 3.8: Kịch bản Use Case Xem lịch hẹn.....	39
Bảng 3.9: Kịch bản Use Case Kiểm duyệt thủ công.....	40
Bảng 3.10: Kịch bản cho Use case kiểm duyệt tự động.....	41

DANH MỤC HÌNH VẼ

Hình 1.1: Trang web Bunz Trade Zone.....	3
Hình 1.2: Một hội nhóm đổi đồ trên Facebook.....	5
Hình 2.1: Logo của ReacJS.....	8
Hình 2.2: Logo của Spring Boot.....	10
Hình 2.3: Logo của MySQL.....	12
Hình 2.4: Kiến trúc Faster RCNN [1].....	14
Hình 2.5: Logo YOLO.....	15
Hình 2.6: Nhận diện một chiếc áo phông.....	17
Hình 2.7: So sánh YOLOv8 với các phiên bản tiền nhiệm [3].....	17
Hình 2.8: Kiến trúc của YOLOv8 theo người dùng github RangeKing [4].....	19
Hình 3.1: Biểu đồ Use Case tổng quát.....	23
Hình 3.2: Biểu đồ Use case Đăng nhập.....	24
Hình 3.3: Biểu đồ Use case Xem chi tiết bài đăng.....	24
Hình 3.4: Biểu đồ Use case quản lý bài đăng.....	25
Hình 3.5: Biểu đồ Use Case Chat.....	26
Hình 3.6: Biểu đồ Use case yêu cầu trao đổi.....	27
Hình 3.7: Biểu đồ Use case Xử lý yêu cầu trao đổi.....	28
Hình 3.8: Biểu đồ Use case Xem lịch sử trao đổi.....	28
Hình 3.9: Biểu đồ Use case Xem lịch hẹn.....	29
Hình 3.10: Biểu đồ Use case kiểm duyệt.....	30
Hình 3.11: Biểu đồ lớp phân tích.....	42
Hình 3.12: Biểu đồ tuần tự chức năng Đăng nhập.....	43
Hình 3.13: Biểu đồ tuần tự chức năng Xem chi tiết bài đăng.....	44
Hình 3.14: Biểu đồ tuần tự chức năng tạo bài đăng.....	45
Hình 3.15: Biểu đồ tuần tự chức năng Chat.....	46
Hình 3.16: Biểu đồ tuần tự chức năng Yêu cầu trao đổi.....	47
Hình 3.17: Biểu đồ tuần tự chức năng Xử lý yêu cầu trao đổi.....	48
Hình 3.18: Biểu đồ tuần tự Xem lịch sử trao đổi.....	48
Hình 3.19: Biểu đồ tuần tự Xem lịch hẹn.....	49
Hình 3.20: Biểu đồ tuần tự chức năng kiểm duyệt bài thủ công.....	50
Hình 3.21: Biểu đồ tuần tự kiểm duyệt bài đăng tự động.....	51
Hình 3.22: Mô hình quan hệ thực thể.....	53
Hình 3.23: Cơ sở dữ liệu cho trang web.....	54
Hình 3.24: Nhãn của một hình ảnh.....	55
Hình 3.25: Hình ảnh dùng làm tập dữ liệu.....	56
Hình 3.26: Quá trình train cho 5 epoch đầu.....	57
Hình 3.27: Quá trình train cho các epoch cuối.....	57
Hình 3.28: Đường cong Precision-Recall.....	58

Hình 3.29: Biểu đồ biến động của các hàm loss.....	59
Hình 3.30: Thử nghiệm model.....	59
Hình 3.31: Kết quả thử nghiệm model.....	60
Hình 3.32: Mô hình dự đoán nhãn “pants”	60
Hình 3.33: Mô hình dự đoán nhãn “sweater”	61
Hình 3.34: Mô hình không có dấu hiệu bắt nhầm vật thể.....	61
Hình 4.1: Cấu trúc ứng dụng Spring Boot.....	62
Hình 4.2: Khởi chạy ứng dụng Spring Boot.....	63
Hình 4.3: Cấu trúc ứng dụng React.....	63
Hình 4.4: Khởi chạy ứng dụng React.....	64
Hình 4.5: Trạng thái ứng dụng khi khởi chạy.....	64
Hình 4.6: Giao diện Đăng nhập.....	64
Hình 4.7: Giao diện Trang chủ.....	65
Hình 4.8: Giao diện Thêm bài đăng.....	65
Hình 4.9: Giao diện Chi tiết bài đăng.....	66
Hình 4.10: Giao diện tạo yêu cầu trao đổi.....	66
Hình 4.11: Quản lý lịch sử trao đổi.....	67
Hình 4.12: Xác nhận trao đổi.....	67
Hình 4.13: Giao diện Chat.....	68
Hình 4.14: Giao diện quản lý bài đăng của Admin.....	68
Hình 4.15: Giao diện quản lý người dùng của Admin.....	69
Hình 4.16: Giao diện kiểm duyệt.....	69
Hình 4.17: Giao diện thống kê.....	70

DANH MỤC CÁC TỪ VÀ THUẬT NGỮ VIẾT TẮT

Từ	Từ gốc	Ý nghĩa
DOM	Document Object Model	Mô hình đối tượng tài liệu – được định nghĩa bởi W3C
HTML	HyperText Markup Language	Là một ngôn ngữ đánh dấu được thiết kế để tạo nên các trang web
XML	Extensible Markup Language	Là một ngôn ngữ được tạo ra bởi W3C
CSDL	Cơ sở dữ liệu	Hệ thống lưu trữ dữ liệu
SQL	Structure Query Language	Ngôn ngữ truy vấn dữ liệu
YOLO	You only look once	Một mô hình nhận diện đối tượng
SOTA	State of the art	Chỉ mô hình có chất lượng đi đầu trong một lĩnh vực
mAP	Mean Average Precision	Chỉ số thường được dùng để đánh giá mô hình nhận diện đối tượng trong ảnh

MỞ ĐẦU

Trong xã hội hiện đại, việc mua bán, tiêu thụ, sản xuất ngày càng tăng cao, nhu cầu mua sắm, thói quen tiêu thụ ngày càng phát triển, mức tiêu thụ quá thừa thãi dẫn đến vấn đề đồ đạc cũ, không có cách xử lý. Vấn đề về đồ cũ, đặc biệt là trang phục, đang trở thành một thách thức đáng kể. Hàng năm, triệu tấn quần áo bị loại bỏ, tạo ra một tình trạng lãng phí tài nguyên và gây tác động không tốt đến môi trường.

Với trang phục cũ, cũng vẫn luôn có cách giải quyết. Chúng ta có thể tái sử dụng cho người thân hoặc gia đình, sử dụng lại những món đồ của người thân, hoặc mang tặng những món đồ không dùng nữa. Chúng ta cũng có thể đem tặng những món đồ đó cho các tổ chức từ thiện để những món đồ cũ đó đến được với những người thực sự cần. Ngoài ra có một phương pháp khác là chúng ta có thể trao đổi những món đồ cũ với nhau. Cách thức xử lý đồ cũ này mang lại nhiều lợi ích tốt như giảm ô nhiễm môi trường, tăng sự tương tác của mọi người với nhau thông qua quá trình trao đổi, giảm thiểu lãng phí tài nguyên, ...

Trên cơ sở lợi ích mà việc trao đổi trang phục cũ đem lại, hướng tới một thói quen xử lý trang phục cũ khác, việc xây dựng lên một nền tảng web trao đổi đồ thời trang hỗ trợ cho việc đó là rất cần thiết. Việc lựa chọn web là nền tảng chính giúp tăng khả năng tiếp cận từ người dùng, tăng sự linh hoạt và tiện ích của nền tảng.

Nội dung chính của đồ án bao gồm các phần sau:

Chương I: Giới thiệu chung - Chương này sẽ trình bày mục tiêu, ý nghĩa thực tiễn của đồ án. Khảo sát các hệ thống tương tự. Xác định vấn đề và các yêu cầu của trang web

Chương II: Cơ sở lý thuyết - Chương này sẽ trình bày các kiến thức cơ bản liên quan đến đề tài. Các phương pháp để xử lý các yêu cầu đặt ra ở Chương I.

Chương III: Phân tích và Thiết kế hệ thống - Chương này sẽ trình bày phương pháp thiết kế để đáp ứng các yêu cầu của bài toán. Bao gồm các sơ đồ kiến trúc cho các chức năng và cơ sở dữ liệu.

Chương 4: Triển khai và thử nghiệm hệ thống - Chương này sẽ trình bày việc xây dựng và triển khai hệ thống, bao gồm cả quá trình thử nghiệm sử dụng.

CHƯƠNG 1: GIỚI THIỆU CHUNG

Chương 1 của đồ án sẽ trình bày về mục tiêu và ý nghĩa đồ án hướng tới, xác định sản phẩm hướng tới, khảo sát một số hệ thống với mục đích tương tự và từ đó đưa ra các yêu cầu của bài toán.

1.1. Mục tiêu và ý nghĩa của đồ án

1.1.1. Hiện trạng

Hiện tại, vấn đề về trang phục cũ đang trở thành một trong những thách thức lớn đối mặt với xã hội hiện đại. Việc tiêu thụ và loại bỏ quần áo cũ ngày càng tăng lên, dẫn đến những tác động tiêu cực đáng kể đến môi trường và xã hội.

Một trong những tác động tiêu cực rõ ràng nhất của vấn đề này là gây ô nhiễm môi trường. Bình quân mỗi người dân Mỹ thải ra trung bình 37kg rác thải quần áo mỗi năm, với 85% trong số đó được đưa đến các bãi chôn lấp hoặc lò đốt mặc dù 95% có thể được tái sử dụng. Việc sản xuất và xử lý quần áo tiêu tốn nhiều nguồn tài nguyên, góp phần gia tăng lượng khí thải và chất thải nhựa trong quá trình sản xuất, vận chuyển và xử lý. Hơn nữa, việc nạo vét các nguồn tài nguyên tự nhiên để sản xuất vải cũng gây ra nhiều vấn đề về khai thác không bền vững.

Có nhiều cách khác nhau để giải quyết đồ cũ một cách có ích và thân thiện với môi trường. Một trong những phương pháp đem lại nhiều hiệu quả là việc đổi đồ. Thay vì loại bỏ các món đồ không còn sử dụng, việc đổi đồ tạo ra một cơ hội để chia sẻ và tái sử dụng những sản phẩm đó với người khác. Ngoài ra, việc trao đổi cũng tạo cho chúng ta cơ hội gặp gỡ, làm quen với những người khác, phần nào cũng giúp tăng sự gắn kết trong xã hội.

1.1.2. Mục tiêu và ý nghĩa của đồ án

Dựa trên việc trao đổi đồ cũ, đồ án này đặt ra mục tiêu quan trọng là xây dựng một nền tảng web trao đổi trang phục cũ, tạo ra một cộng đồng sôi động và tương tác giữa những người quan tâm đến việc tái sử dụng và chia sẻ mặt hàng thuộc nhóm đồ trang phục đã qua sử dụng. Nền tảng sẽ cung cấp một môi trường thuận lợi và an toàn để người dùng có thể đăng tải, tìm kiếm và thực hiện các giao dịch trao đổi. Trang web cũng bao gồm các chức năng quản lý cơ bản giúp người quản lý trang web theo dõi được tình hình hoạt động của người dùng.

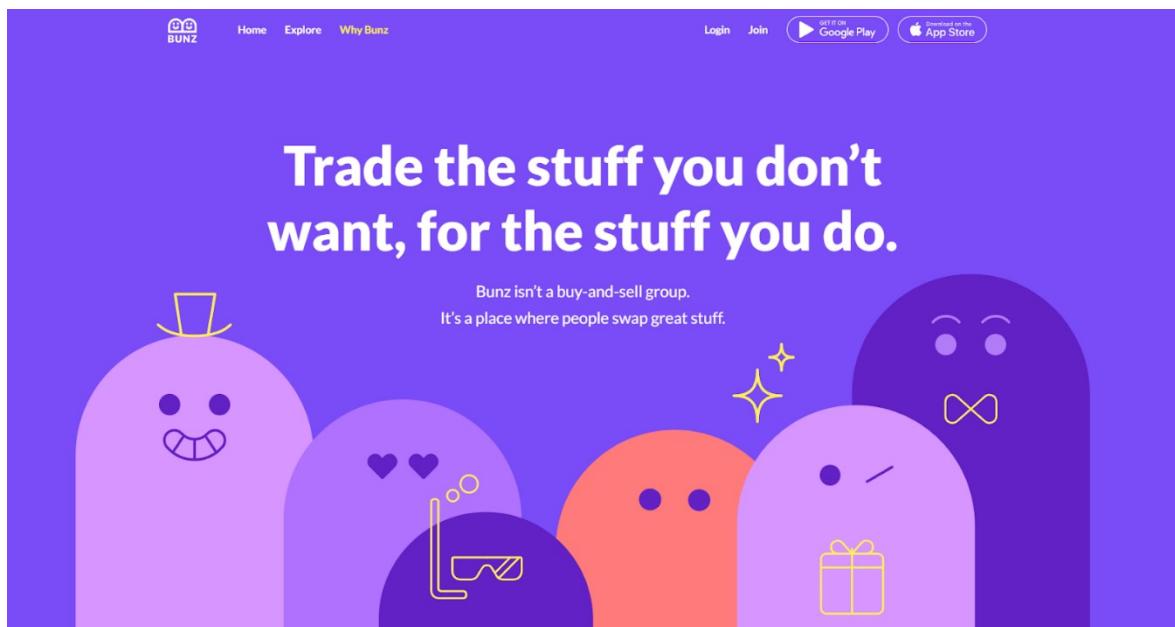
Bằng cách này, sản phẩm đồ án hướng tới không chỉ mang lại lợi ích kinh tế và môi trường mà còn tạo ra một cộng đồng sáng tạo và tiến bộ, đóng góp vào việc xây dựng một tương lai bền vững hơn cho môi trường và xã hội.

1.2. Khảo sát các hệ thống tương tự

1.2.1. Bunz trade zone

a. Giới thiệu

Bunz Trade Zone là một nền tảng trao đổi mặt hàng sử dụng trong cộng đồng, nổi tiếng với sự tiện lợi và tính bền vững. Được tạo ra tại Toronto, Canada, Bunz đã nhanh chóng trở thành một điểm đến phổ biến cho những người muốn đổi đồ cũ, mua bán và chia sẻ các mặt hàng và dịch vụ. Bunz có đa dạng loại mặt hàng và cộng đồng của Bunz có tính đa văn hóa, mang lại trải nghiệm trao đổi độc đáo và đồng thời thúc đẩy ý thức về việc sử dụng tài nguyên và bảo vệ môi trường.



Hình 1.1: Trang web Bunz Trade Zone

b. Đối tượng sử dụng

Bunz là một nền tảng trao đổi mặt hàng phù hợp cho mọi đối tượng. Dành cho những người yêu thích sự tiện lợi và tính tiết kiệm, Bunz mang đến cơ hội đổi đồ cũ một cách thông minh và bền vững. Do được xây dựng hướng tới nhiều thể loại sản phẩm trao đổi, các thành viên của cộng đồng Bunz cũng rất đa dạng, từ những người đam mê thời trang đến những người quan tâm đến môi trường. Đây là một nơi hội tụ các cá nhân mang trong mình ý thức về việc sử dụng tài nguyên một cách thông minh, đồng thời tạo nên một không gian giao lưu xã hội đầy sôi động. Ban đầu cộng đồng người sử dụng chỉ tập trung ở Toronto, Canada (Nơi ở của người thành lập Bunz - Emily Bitze). Mặc dù có vẻ Bunz có muôn mở rộng phạm vi của cộng đồng, tuy nhiên vì tính chất hướng đến việc trao đổi trực tiếp nên Bunz chưa hoàn toàn mở rộng được.

Trang chủ của trang web: bunz.com

c. Chức năng chính

Bunz Trade Zone cung cấp nhiều chức năng chính giúp người dùng thực hiện các giao dịch trao đổi mặt hàng cũ.

- Trao đổi mặt hàng: Bunz cho phép người dùng đăng các mặt hàng cũ của mình và tìm kiếm các mặt hàng mà người khác muốn đổi. Người dùng có thể gửi yêu cầu đổi mặt hàng và thương lượng điều kiện giao dịch qua chat.
- Đăng tin gian hàng cá nhân: Người dùng có thể tạo gian hàng cá nhân để hiển thị các mặt hàng mà họ có sẵn để đổi hoặc các dòng mặt hàng mà họ có nhu cầu tìm kiếm. Ngoài ra còn có thông tin về những lần trao đổi thành công cũng như những tương tác cá nhân như lượt follow và review.
- Các danh mục đa dạng: Bunz cung cấp nhiều danh mục khác nhau cho các mặt hàng, bao gồm thời trang, đồ điện tử, đồ gia dụng, ... Đặc biệt hơn, bunz hỗ trợ dòng sản phẩm phi hiện vật.
- Tích điểm và thưởng: Bunz có hệ thống tích điểm và thưởng để khuyến khích sự giao lưu và tham gia của cộng đồng.
- Cộng đồng và diễn đàn: Bunz cung cấp diễn đàn và cộng đồng để người dùng có thể thảo luận, chia sẻ kinh nghiệm và tìm kiếm gợi ý từ cộng đồng.
- Hẹn lịch gặp mặt giữa 2 người: Khi 2 người đồng ý trao đổi với nhau, họ có thể hẹn gặp nha.
- Hỗ trợ đa nền tảng: Bunz hỗ trợ cả nền tảng web và app.

d. Ưu và nhược điểm

Ưu điểm

- Một trong những nền tảng đi đầu theo mô hình trao đổi đồ cũ, có một cộng đồng lớn và sôi nổi
- Giao diện thân thiện và tiện lợi cho người sử dụng
- Đa dạng các dòng đồ từ đồ thời trang, đồ gia dụng, các dòng đồ phi hiện vật, ...
- Bunz hỗ trợ cả nền tảng web và mobile, tăng tính tiện lợi của hệ thống

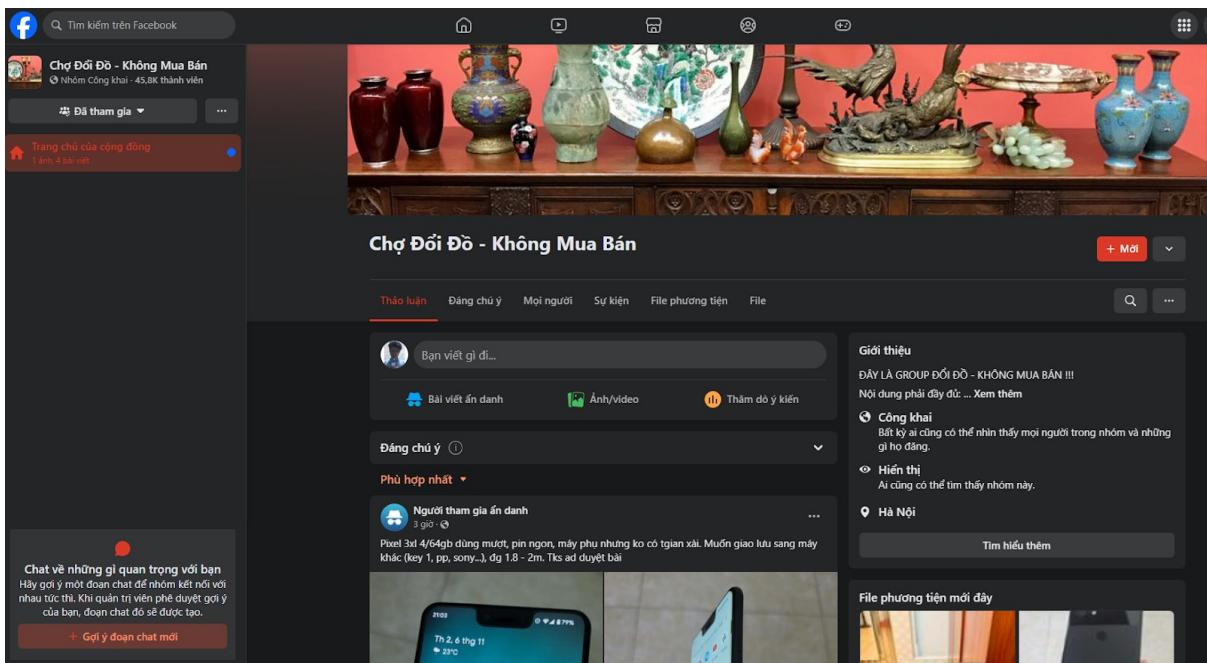
Nhược điểm

- Chưa có các biện pháp kiểm duyệt bài đăng và kiểm soát độ trung thực của người sử dụng
- Chất lượng của các dòng sản phẩm chưa có hình thức đánh giá
- Hạn chế về vị trí vì Bunz chỉ hỗ trợ một vùng địa lý.
- Bunz đã bị giảm nhiều sức hút vì kết hợp cơ chế coin

1.2.2. Các hội nhóm trên Facebook

a. Giới thiệu

Các hội nhóm đổi đồ trên Facebook là những cộng đồng trực tuyến nơi các thành viên có thể trao đổi, mua bán hoặc tặng những món đồ mà họ không còn cần đến những người khác trong nhóm. Thông qua việc chia sẻ ảnh và thông tin về sản phẩm, thành viên có thể tìm kiếm những món đồ mà họ quan tâm.



Hình 1.2: Một hội nhóm đổi đồ trên Facebook

b. Đối tượng sử dụng

Bất kỳ người dùng Facebook nào cũng có thể tham gia các hội nhóm Facebook miễn là đảm bảo các quy tắc nhóm đưa ra. Các hội nhóm hướng tới các đối tượng có nhu cầu về một dòng sản phẩm cụ thể.

Người tham gia các hội nhóm đổi đồ thường hướng tới các nhu cầu sau:

- Tiết kiệm và tái sử dụng
- Sưu tầm, tìm kiếm các món đồ độc đáo
- Quan tâm đến bảo vệ môi trường
- Muốn kết nối và chia sẻ cộng đồng
- Muốn có cách xử lý cho những món đồ cũ mà không phải là bỏ đi

c. Các tính năng chính

Chủ yếu sử dụng các tính năng cơ bản của chính Facebook như đăng bài, tương tác, bình luận, chat, ...

Cơ bản luồng công việc khi muốn có một trao đổi sẽ như sau:

- Người có đồ muốn đổi sẽ đăng bài trên hội nhóm về sản phẩm và thông tin liên lạc
- Những người có nhu cầu sẽ bình luận, chat hoặc liên lạc trực tiếp với chủ bài đăng bằng chức năng của Facebook
- Chủ bài đăng sẽ thương thảo với những người chủ bài đăng thấy tiềm năng
- Sau đó họ có thể hẹn nhau về cách đổi 2 món đồ, có thể là gặp nhau hoặc gửi qua các dịch vụ chuyển phát

d. Ưu và nhược điểm

Ưu điểm:

- Tính năng của Facebook đã rất hoàn thiện và dễ sử dụng
- Lượng người sử dụng Facebook là rất lớn nên chỉ cần chọn đúng nhóm mặt hàng, hoàn toàn có thể có một lượng lớn người dùng tham gia
- Việc kiểm duyệt cũng khá ổn định nhờ hệ thống kiểm duyệt của Facebook

Nhược điểm:

- Hệ thống bài đăng có phần thiếu nghiệp vụ cụ thể vì Facebook được xây dựng hướng mạng xã hội hơn là vì mục đích trao đổi đồ
- Vấn đề kiểm soát lừa đảo vẫn tồn tại
- Các hội nhóm không có tính tập trung, quá nhiều hội nhóm với quá nhiều dòng hàng, thậm chí là một dòng hàng lại có nhiều hội nhóm
- Về cơ bản vẫn sử dụng hệ sinh thái của Facebook nên vẫn bao gồm nhiều chức năng không cần thiết

1.3. Xác định yêu cầu của hệ thống

Từ mục đích của đồ án và những khảo sát từ các hệ thống tương tự ta có thể đưa ra phạm vi và các yêu cầu trong hệ thống như sau:

Phạm vi:

Hệ thống trang web hướng đến tất cả những người sử dụng internet có nhu cầu trao đổi những món đồ trang phục cũ như quần áo, giày dép, phụ kiện, ...

Yêu cầu:

Hệ thống trang web trao đổi mặt hàng thời trang cũ là trang web nơi người dùng có thể đăng tải, quản lý những món đồ cũ của mình thuộc dòng hàng là đồ thời trang cho những người khác xem, trao đổi. Trên nền tảng này, người dùng có thể xem thông tin về những mặt hàng cũ, những mặt hàng người dùng khác đang tìm kiếm. Để phục vụ cho việc đổi đồ, người dùng có thể quản lý những yêu cầu đổi đồ từ người dùng khác, nếu chấp nhận một cuộc trao đổi, người dùng có thể đặt lịch hẹn cho cả hai. Người dùng cũng có thể báo cáo những bài đăng, tài khoản không sử dụng trang web đúng mục đích. Trang web cũng giúp người dùng có thể chat để trao đổi với nhau về món đồ của mình. Quản lý trang web có thể quản lý người dùng, thể loại sản phẩm, người dùng, thống kê một số thông tin, kiểm duyệt các bài đăng thủ công hoặc tự động bằng cách nhận diện đối tượng trang phục trong ảnh.

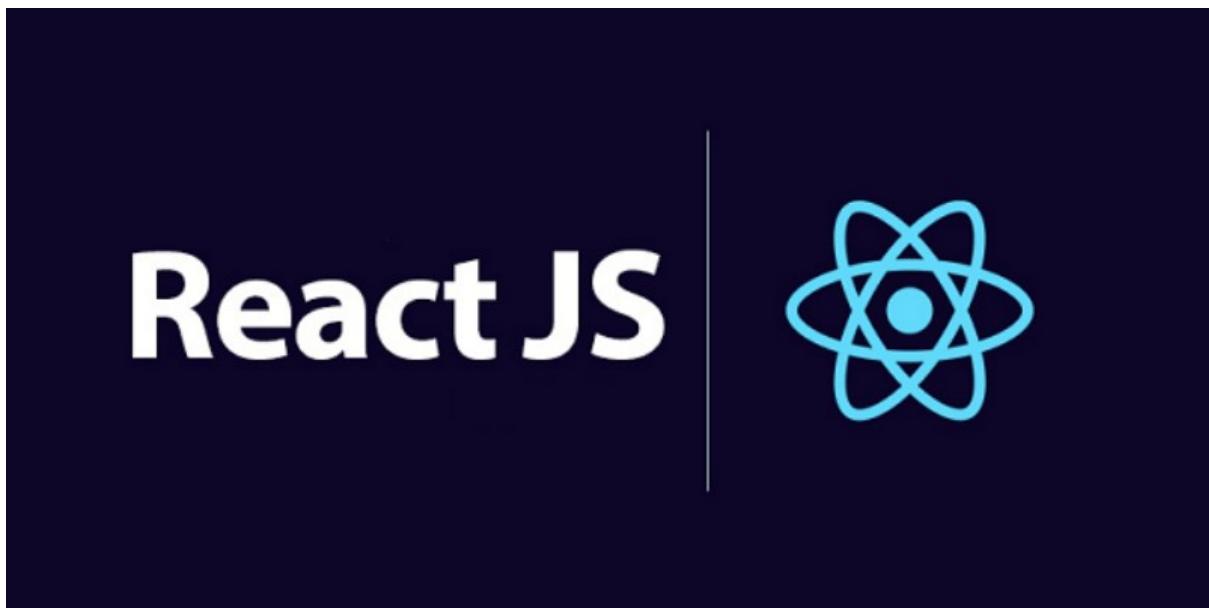
1.4. Kết luận chương

Qua chương I chúng ta đã xác định được mục tiêu của đồ án là xây dựng một trang web giúp người dùng có thể trao đổi các mặt hàng thời trang cũ. Ngoài ra, chúng ta đã khảo sát qua một số hệ thống có ý nghĩa tương tự và từ đó xác định được các yêu cầu cần thiết của trang web để đạt được mục tiêu đề ra. Ở chương tiếp theo, đồ án sẽ trình bày về các phương pháp để hoàn thiện các yêu cầu được đưa ra ở chương này.

CHƯƠNG 2: CƠ SỞ LÝ THUYẾT

Chương 2 của đồ án sẽ trình bày các công cụ và phương pháp nhằm giải quyết các yêu cầu của trang web về các thành phần như: giao diện, server xử lý, model nhận diện ảnh, ... được rút ra từ Chương 1.

2.1. Giới thiệu React



Hình 2.3: Logo của ReacJS

React (còn được gọi là ReactJS) là một Thư viện javascript được tạo ra bởi sự cộng tác giữa Facebook và Instagram. Nó cho phép những nhà phát triển web tạo ra giao diện người dùng nhanh chóng. Phần Views của Reactjs thường được hiển thị bằng việc chủ yếu dùng các component mà chứa các component cũ thẻ hoặc các thẻ HTML. Một trong những đặc trưng duy nhất của Reactjs là việc render dữ liệu không những có thể thực hiện ở tầng server mà còn ở tầng client.

Nó cũng sử dụng khái niệm là Virtual DOM (DOM ảo). Virtual DOM tạo ra bản cache cấu trúc dữ liệu của ứng dụng trên bộ nhớ. Sau đó, ở mỗi vòng lặp, nó liệt kê những thay đổi và sau đó là cập nhật lại sự thay đổi trên DOM của trình duyệt một cách hiệu quả. Điều này cho phép ta viết các đoạn code như thẻ toàn bộ trang được render lại dù thực tế là Reactjs chỉ render những component hay subcomponent nào thực sự thay đổi.

Ưu điểm của ReactJS:

- Reactjs có hiệu quả tốt: Reactjs tạo ra cho chính nó Virtual DOM – nơi mà các component thực sự tồn tại trên đó. Điều này sẽ giúp cải thiện hiệu suất rất nhiều. Reactjs cũng tính toán những thay đổi nào cần cập nhật lên DOM và chỉ thực hiện chúng. Điều này giúp Reactjs tránh những thao tác cần trên DOM mà nhiều chi phí.
- Reactjs giúp việc viết các đoạn code JS dễ dàng hơn: Nó dùng cú pháp đặc biệt là JSX (Javascript mở rộng) cho phép ta trộn giữa code HTML và Javascript. Ta có thể thêm vào các đoạn HTML vào trong hàm render mà không cần phải nối chuỗi. Đây là đặc tính thú vị của Reactjs. Nó sẽ chuyển đổi các đoạn HTML thành các hàm khởi tạo đối tượng HTML bằng bộ biến đổi JSX.
- Hiệu năng cao đối với các ứng dụng có dữ liệu thay đổi liên tục, dễ dàng cho bảo trì và sửa lỗi.

2.2. Giới thiệu Spring Boot



Hình 2.4: Logo của Spring Boot

Spring Boot là một trong số các module của spring framework chuyên cung cấp các tính năng RAD(Rapid Application Development) cho phép tạo ra và phát triển các ứng dụng độc lập dựa trên spring một cách nhanh chóng.

Spring Boot ra đời với mục đích loại bỏ những cấu hình phức tạp của Spring, nó không yêu cầu cấu hình XML và nâng cao năng suất cho các nhà phát triển. Với sự góp mặt của Spring Boot, hệ sinh thái Spring đã trở nên mạnh mẽ, phổ biến và hiệu quả hơn bao giờ hết.

Ưu điểm của Spring Boot:

Spring Boot được thiết kế để giúp các kỹ sư phần mềm đẩy nhanh quá trình phát triển, cho phép họ loại bỏ việc thiết lập và cấu hình ban đầu tốn thời gian của môi trường triển khai. Các lợi ích chính của Spring Boot bao gồm:

- Phát triển các ứng dụng dựa trên Spring một cách tiết kiệm thời gian và dễ dàng.
- Tự động cấu hình tất cả các components cho một ứng dụng Spring cấp sản xuất.

- Các máy chủ nhúng được tạo sẵn (Tomcat, Jetty và Undertow), dẫn đến việc triển khai ứng dụng được tăng tốc và hiệu quả hơn.
- Điểm cuối HTTP, cho phép nhập các tính năng bên trong ứng dụng như chỉ số, tình trạng sức khỏe, v.v.
- Nhiều lựa chọn bổ sung, hỗ trợ nhà phát triển làm việc với cơ sở dữ liệu được nhúng và trong bộ nhớ.
- Dễ dàng truy cập cơ sở dữ liệu và các dịch vụ hàng đợi như MySQL, Oracle, MongoDB, Redis, ActiveMQ và các dịch vụ khác.
- Tích hợp trơn tru với hệ sinh thái Spring.
- Cộng đồng lớn và rất nhiều hướng dẫn, tạo điều kiện cho giai đoạn làm quen.

2.3. Giới thiệu cơ sở dữ liệu MySQL



Hình 2.5: Logo của MySQL

MySQL là hệ quản trị cơ sở dữ liệu tự do nguồn mở phổ biến nhất thế giới và được các nhà phát triển rất ưa chuộng trong quá trình phát triển ứng dụng. Vì MySQL là hệ quản trị cơ sở dữ liệu tốc độ cao, ổn định và dễ sử dụng, có tính khả chuyen, hoạt động trên nhiều hệ điều hành cung cấp một hệ thống lớn các hàm tiện ích rất mạnh. Với tốc độ và tính bảo mật cao, MySQL rất thích hợp cho các ứng dụng có truy cập CSDL trên internet. Người dùng có thể tải về MySQL miễn phí từ trang chủ. MySQL có nhiều phiên bản cho các hệ điều hành khác nhau: phiên bản Win32 cho các hệ điều hành dòng Windows, Linux, Mac OS X, Unix, FreeBSD, NetBSD, Novell NetWare, SGI Irix, Solaris, SunOS,... .

Ưu điểm của MySQL:

- MySQL là một cơ sở dữ liệu đa nền tảng tốc độ cao, ổn định, dễ sử dụng, cung cấp một tập hợp lớn các chức năng tiện ích rất mạnh mẽ.
- Do MySQL có nhiều tính năng bảo mật, tất cả đều mức cao nên nó rất lý tưởng cho các ứng dụng truy cập cơ sở dữ liệu trên Internet.
- Bất kỳ ai học MySQL cơ bản đều biết MySQL hỗ trợ nhiều hàm SQL được mong đợi của một hệ quản trị cơ sở dữ liệu quan hệ, cả trực tiếp và gián tiếp.

- MySQL có thể xử lý một lượng lớn dữ liệu và nếu cần và nó có thể mở rộng nhanh chóng.
- Việc áp dụng một số tiêu chuẩn cho phép MySQL hoạt động rất hiệu quả và tiết kiệm chi phí, tăng tốc độ thực thi.

2.4. Giới thiệu về YOLO

2.4.1. Bài toán nhận diện đối tượng

Theo yêu cầu đặt ra cho hệ thống, chúng ta cần có một module giúp bài đăng đăng lên được kiểm duyệt một cách tự động. Do giới hạn về thời gian và tài nguyên, trong đồ án này sẽ tập trung vào việc kiểm duyệt thông qua các ảnh của bài đăng với ý tưởng như sau: Khi bài đăng được tạo bởi người dùng, danh sách ảnh của bài đăng sẽ được gửi cho một hệ thống dự đoán, hệ thống này trả về kết quả cho hệ thống chính và từ đó hệ thống chính quyết định sẽ kiểm duyệt, khóa hay chuyển cho Quản lý trang web kiểm duyệt thủ công.

Cụ thể chúng ta sẽ kiểm duyệt qua ảnh như thế nào? Chúng ta cần nhận biết bức ảnh đó có “thỏa mãn” yêu cầu của chúng ta không. Yêu cầu được đặt ra ở đây là “Bức ảnh đó có liên quan đến quần áo không?”. Từ đó ta cần nhận diện được liệu trong bức ảnh đó có chụp đối tượng là quần áo không. Với luồng tư duy đó, chúng ta cần có kiến thức về bài toán Phát hiện đối tượng (Object detection) trong Thị giác máy tính.

Phát hiện đối tượng là một bài toán phổ biến trong thị giác máy tính. Nó liên quan đến việc khoanh một vùng quan tâm trong ảnh và phân loại vùng này tương tự như phân loại hình ảnh. Tuy nhiên, một hình ảnh có thể bao gồm một số vùng quan tâm trỏ đến các đối tượng khác nhau. Điều này làm cho việc phát hiện đối tượng trở thành một vấn đề nâng cao hơn của phân loại hình ảnh.

2.4.2. Các mô hình có thể áp dụng cho bài toán

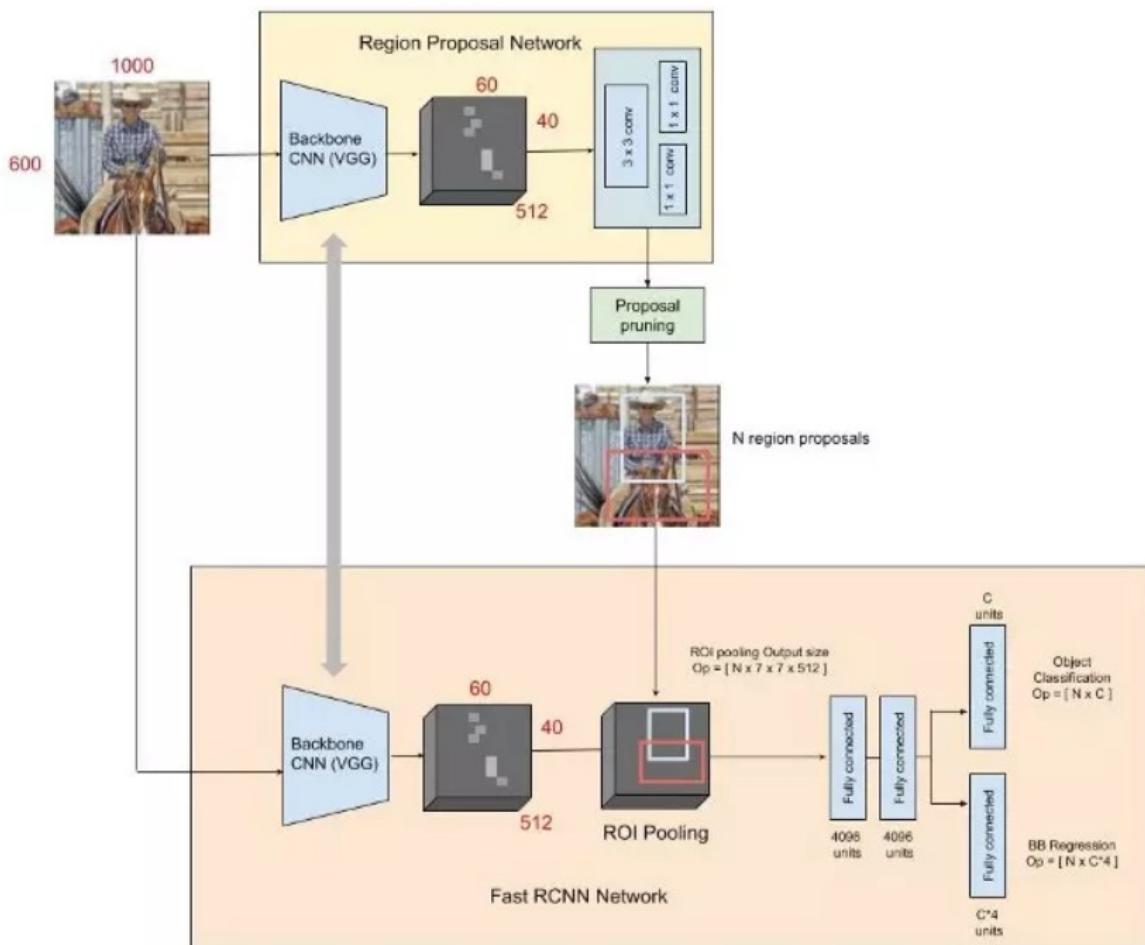
a. Faster RCNN

Giới thiệu

Faster RCNN là một trong các phương pháp phát hiện đối tượng sử dụng mạng Deep learning đạt độ chính xác cao trên các tập dữ liệu chuẩn như COCO. Faster RCNN được cải tiến dựa trên 2 phương pháp trước đó là RCNN và Fast RCNN. Trong Faster RCNN, tác giả đề xuất sử dụng mạng các vùng đề xuất RPN (Region Proposal Network) để tạo ra các vùng đề xuất. Sau khi có được các đặc trưng từ các lớp tích chập (convolutional) đầu tiên, mạng RPN sử dụng cửa sổ trượt trên bản đồ đặc trưng (feature map) để rút trích đặc trưng cho mỗi vùng đề xuất. RPN được xem như là một

mạng liên kết đầy đủ cùng lúc thực hiện 2 nhiệm vụ đó là dự đoán tọa độ cho các đối tượng và độ tin cậy cho đối tượng đó (objectness score). So với các phương pháp trước đó Faster RCNN đạt kết quả cao hơn và có thời gian xử lý nhanh hơn, tuy nhiên tốc độ vẫn vẫn chưa thể đáp ứng xử lý theo thời gian thực.

Mô hình Faster RCNN có thể được mô phỏng như hình sau:



Hình 2.6: Kiến trúc Faster RCNN [1]

b. SSD (Single Shot Detector)

Giới thiệu

Trong SSD, tại mỗi vị trí trên feature map, SSD đặt các default bounding boxes với kích thước, tỷ lệ khác nhau. Trong quá trình xử lý, SSD sẽ đánh giá và tìm đối tượng trên các bounding box này nhằm tìm ra box phù hợp nhất với đối tượng cần tìm kiếm.Thêm vào đó, bằng việc tìm kiếm trên các feature map khác nhau, SSD có thể tìm kiếm các đối tượng với kích thước khác nhau mà không cần thay đổi kích thước của các bounding box. Thực nghiệm cho thấy, SSD đạt 74.3% mAP trên tập test của VOC2007, đạt 59 FPS khi sử dụng Nvidia Titan X với kích thước ảnh đầu vào 512 x 512, nhanh hơn Faster R-CNN.

Hướng tiếp cận

Ý tưởng chính của SSD đến từ việc sử dụng các bounding box, bằng việc khởi tạo sẵn các box tại mỗi vị trí trên ảnh, SSD sẽ tính toán và đánh giá thông tin tại mỗi vị trí xem vị trí đó có vật thể hay không, nếu có thì là vật thể nào, và dựa trên kết quả của các vị trí gần nhau, SSD sẽ tính toán được một box phù hợp nhất bao trọn vật thể.

Ngoài ra, bằng việc tính toán bounding box trên các feature map khác nhau, tại mỗi tầng feature map, một box sẽ ôm trọn một phần hình ảnh với các kích thước khác nhau. Như trên ví dụ trên, con mèo và con chó có thể được phát hiện ở 2 tầng feature map khác nhau, 2 kích thước và tỉ lệ khác nhau. Thay vì sử dụng 1 box và thay đổi kích thước box cho phù hợp với vật thể, thì SSD sử dụng nhiều box trên nhiều tầng, từ đó tổng hợp ra vị trí và kích thước box kết quả. Bằng việc loại trừ các region proposal, SSD có thể đạt được tốc độ xử lý cao hơn Faster R-CNN.

c. YOLO (*You look only once*)

YOLO được xem là phương pháp đầu tiên xử lý dữ liệu theo thời gian thực và vẫn đạt được độ chính xác cao. Ý tưởng cốt lõi của YOLO là thay vì sử dụng các vùng đề xuất để rút trích đặc trưng thì YOLO sử dụng các thông tin cục bộ từ dữ liệu huấn luyện để học các đặc trưng cần quan tâm bằng cách chia ảnh dữ liệu đầu vào thành lưới (grid view) để khai thác đặc trưng trên lưới. Nếu trọng tâm của đối tượng rơi vào ô nào trong lưới thì ô đó chịu trách nhiệm phát hiện đối tượng. Kích thước lưới như thế nào phụ thuộc vào phiên bản của YOLO, hiện tại YOLO đã có phiên bản thứ 8.



Hình 2.7: Logo YOLO

Phiên bản mới nhất của YOLO là YOLOv8 với nhiều cải tiến về tốc độ và chất lượng dự đoán so với các phiên bản tiền nhiệm. Bên cạnh đó, YOLO cũng được xem như SOTA (state of the art) trong lĩnh vực nhận diện đối tượng với cộng đồng người sử dụng đông đảo, vì vậy rất phù hợp với yêu cầu được đề ra với chức năng kiểm duyệt. Trong đồ án này sẽ tập trung tìm hiểu và ứng dụng YOLOv8 với chức năng kiểm duyệt tự động đã đề ra.

2.4.3. Giới thiệu về YOLO

YOLO (You Only Look Once) là mô hình phát hiện đối tượng phổ biến được biết đến với tốc độ nhanh và độ chính xác cao. Mô hình này lần đầu tiên được giới thiệu bởi Joseph Redmon và cộng sự vào năm 2016. Kể từ đó đến nay, đã có nhiều phiên bản của YOLO, một trong những phiên bản gần đây nhất là YOLO v8.

Điểm khác biệt chính mà YOLO mang lại là cách đặt vấn đề. Tác giả đã đặt lại vấn đề phát hiện đối tượng dưới dạng một bài toán hồi quy (dự đoán tọa độ hộp giới hạn) thay vì phân loại [2].

Các mô hình YOLO được huấn luyện trước trên các bộ dữ liệu lớn như COCO và ImageNet. Điều này cho phép chúng vừa có khả năng cung cấp dự đoán cực kỳ chính xác với các lớp đã được huấn luyện, vừa có thể học các lớp mới một cách tương đối dễ dàng.

Cách làm việc của YOLO có thể tóm tắt như sau:

1. Chia bức ảnh thành các Cells. Ví dụ: 19x19, 13x13, ...
2. Mỗi Cell chịu trách nhiệm dự đoán ra b Bounding Box. $b = 3, 5, 7, \dots$
3. Output của việc dự đoán bao gồm: (p, x, y, w, h, C)

Trong đó:

p : xác suất có đối tượng trong box hay không

x, y : tọa độ tâm của bounding box

w, h : chiều rộng và chiều cao của bounding box

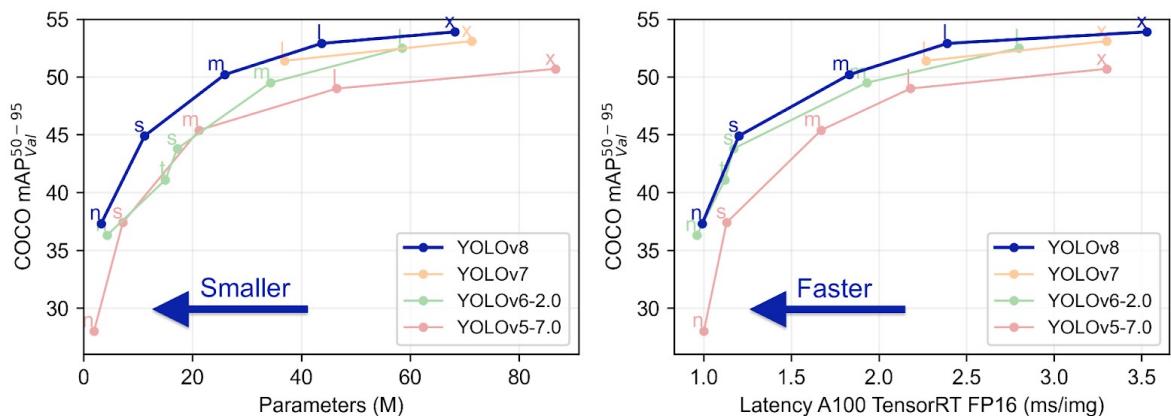
C : xác suất của các class



Hình 2.8: Nhận diện một chiếc áo phông

YOLO V8

YOLOv8 là phiên bản mới nhất trong các mô hình phát hiện đối tượng thời gian thực YOLO, mang lại hiệu suất vượt trội cả về độ chính xác và tốc độ nhận diện so với các phiên bản tiền nhiệm nói riêng và các mô hình tương tự nói chung. Dựa trên những tiến bộ của các phiên bản YOLO trước đó, YOLOv8 giới thiệu các tính năng mới khiến nó trở thành lựa chọn lý tưởng cho các tác vụ phát hiện đối tượng khác nhau trong nhiều ứng dụng. Độ vượt trội của YOLOv8 được tác giả mô phỏng như trong hình sau:



Hình 2.9: So sánh YOLOv8 với các phiên bản tiền nhiệm [3]

YOLOv8 có hỗ trợ 3 nhiệm vụ là: Object Detection, Instance Segmentation và Image Classification, với 5 biến thể tương ứng với kích thước model từ nhỏ nhất đến lớn nhất theo size là: n, s, m, l, x. Sau đây là các chỉ số khi thực hiện detection trên tập dữ liệu COCO.

Model	Kích thước ảnh đầu vào (pixels)	mAP 50-95	Speed CPU ONNX (ms)	Speed A100 TensorRT (ms)	Số lượng param	FLOPs (B)
YOLOv8n	640	37.3	80.4	0.99	3.2	8.7
YOLOv8s	640	44.9	128.4	1.20	11.2	28.6
YOLOv8m	640	50.2	234.7	1.83	25.9	78.9
YOLOv8l	640	52.9	375.2	2.39	43.7	165.2
YOLOv8x	640	53.9	479.1	3.53	68.2	257.8

Bảng 2.1: So sánh các biến thể của YOLOv8 [3]

Trong đó:

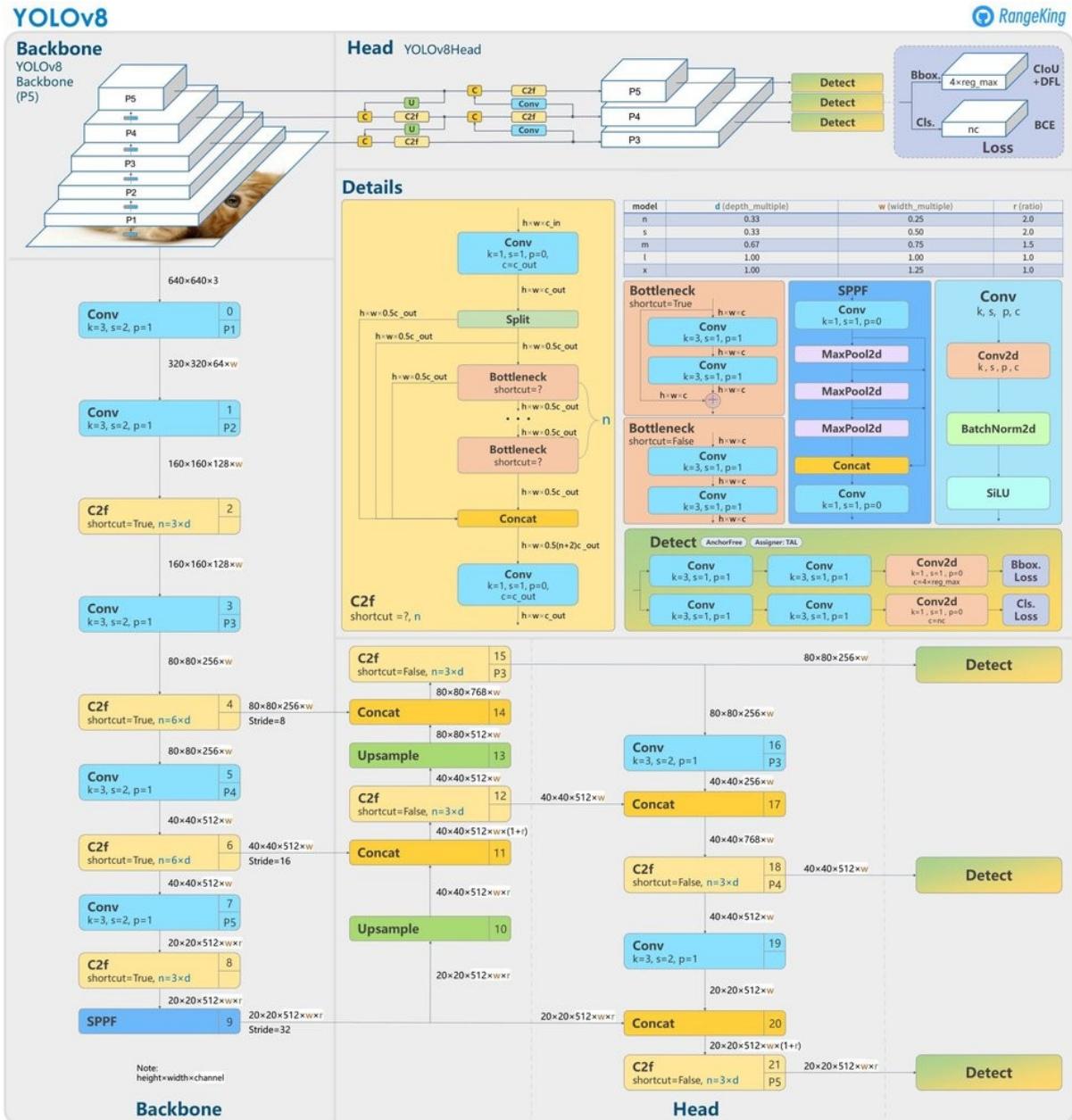
- mAP 50-95: Giá trị mAP (Mean Average Precision) đo lường độ chính xác của mô hình. Giá trị này có thể được tính dựa trên các ngưỡng xác suất từ 50% đến 95%
- Speed CPU ONNX (ms): Thời gian xử lý trung bình của mô hình trên CPU khi sử dụng định dạng ONNX. Đây là thời gian tính bằng mili giây (ms) để mô hình dự đoán trên một ảnh.
- Speed A100 TensorRT (ms): Thời gian xử lý trung bình của mô hình trên GPU A100 khi sử dụng TensorRT. Đây là thời gian tính bằng mili giây (ms) để mô hình dự đoán trên một ảnh.
- FLOPs (B): Số lượng phép toán dấu chấm động (Floating Point Operations) tính toán bởi mô hình, đơn vị tính là tỷ (B). Đây là một đánh giá về độ phức tạp tính toán của mô hình.

YOLOv8 cũng cung cấp các pretrain-model từ phiên bản nano với ít tham số nhất đến phiên bản extra với nhiều tham số nhất. Các mô hình pretrain cho việc detection được đào tạo trên bộ dữ liệu COCO detection với độ phân giải hình ảnh là 640 pixels. Các mô hình pretrain cho việc segmentation được đào tạo trên tập dữ liệu COCO segmentation với độ phân giải hình ảnh là 640. Các mô hình pretrain cho việc classification được huấn luyện trước trên bộ dữ liệu ImageNet với độ phân giải hình ảnh là 224.

Sau khi đánh giá các mô hình pretrain, để tiết kiệm tài nguyên mà vẫn giữ được độ chính xác tương đối, trong đồ án này sẽ sử dụng phiên bản nano cho nhiệm vụ detection.

Kiến trúc của YOLOv8

Trong đồ án này tuy tập trung vào việc ứng dụng yolo, tuy nhiên vẫn sẽ tìm hiểu tổng quan về kiến trúc của YOLOv8. Với YOLOv8, theo người dùng GitHub RangeKing, ta có tệp yolov8-p2.yaml, với hình minh họa sau:



Hình 2.10: Kiến trúc của YOLOv8 theo người dùng github RangeKing [4]

Có thể thấy, YOLOv8 vẫn có 1 khối backbone để trích xuất đặc trưng. Kết quả của khối backbone được đi qua khối Head để có 3 feature map phục vụ cho detect 3 kích thước đối tượng là: nhỏ, vừa và lớn với khối (80, 80, 256) cho dự đoán đối tượng có kích thước nhỏ, khối (40, 40, 512) cho dự đoán đối tượng có kích thước vừa và (20, 20, 512) cho dự đoán đối tượng có kích thước lớn.

2.4.4. Các công cụ cần dùng

Để ứng dụng YOLOv8 chúng ta cần một số công cụ sau:

- *Google Colab*: Sử dụng cho việc train model với GPU miễn phí do Google cung cấp.
- *Ngôn ngữ Python*
- *Công cụ LabelImg*: Sử dụng cho việc gán nhãn dữ liệu
- *Thư viện Flask*: Sử dụng cho việc tạo api để sử dụng model
- *Thư viện Ultralytics*: Sử dụng cho việc đọc model pretrain của YOLOv8
- *Pretrain model phiên bản nano của YOLOv8*

2.5. Kết luận chương

Sau chương này, chúng ta đã xác định được giải pháp và các công cụ để xây dựng nền hệ thống đồ án hướng tới. Trong chương tiếp theo chúng ta sẽ phân tích các yêu cầu để đưa ra thiết kế đề xuất cho hệ thống.

CHƯƠNG 3: PHÂN TÍCH VÀ THIẾT KẾ

Chương 3 của đồ án sẽ xác định các chức năng cho trang web từ đó thiết kế hệ thống theo hướng đối tượng. Ngoài ra chương này cũng trình bày quá trình train mô hình dự đoán đối tượng trong ảnh.

3.1. Xác định chức năng

3.1.1. Use case tổng quan

a. Các tác nhân và chức năng tương ứng

- *Thành viên hệ thống (Member)*: Đăng nhập, đăng ký, quản lý thông tin cá nhân, xem thông tin của người khác, chat, xem danh sách các bài đăng, nhận thông báo.
- *Người dùng (User)*: có đầy đủ chức năng của Member, ngoài ra còn có thể: Xem lịch sử trao đổi, Quản lý các bài đăng của mình (thêm - xóa), Gửi yêu cầu trao đổi đến một bài đăng khác, Quản lý yêu cầu trao đổi, Tạo lịch hẹn, Tìm kiếm bài đăng, Báo cáo, Đánh giá
- *Quản lý trang web (Admin)*: có đầy đủ chức năng của Member, ngoài ra còn có thể: Quản lý thông tin trang web, Quản lý thẻ loại sản phẩm được đăng (thêm, sửa, xóa), Quản lý người dùng (User) (tìm kiếm, khóa tài khoản), Quản lý các bài đăng (tìm kiếm, khóa bài đăng), Kiểm duyệt các bài đăng (tự động, thủ công), Xử lý báo cáo vi phạm.

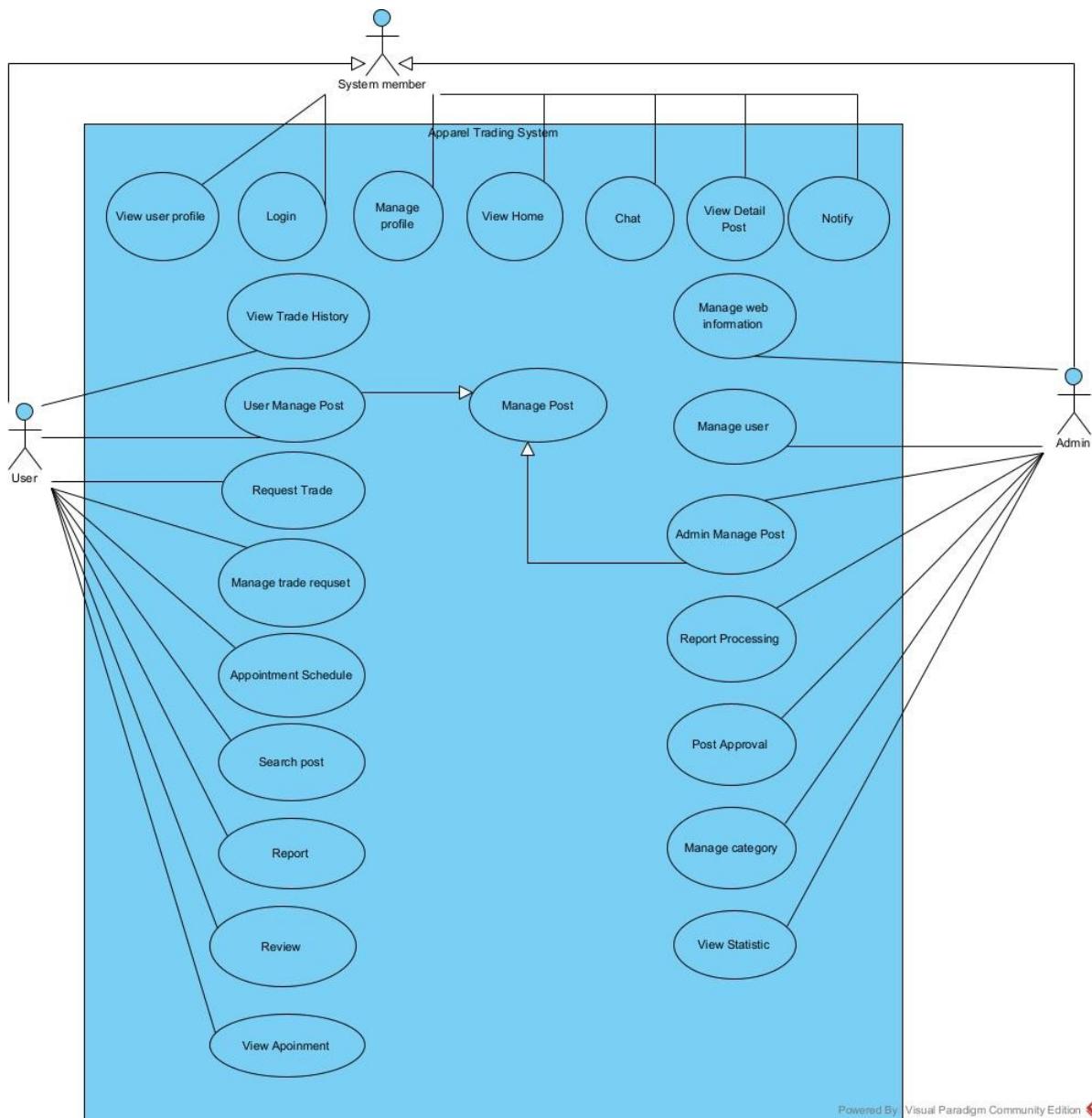
b. Các ca sử dụng

- *Đăng nhập*: Thành viên hệ thống có thể đăng nhập vào hệ thống và sử dụng hệ thống với thông tin của mình.
- *Đăng ký*: Người dùng có thể tạo tài khoản để trở thành thành viên hệ thống
- *Quản lý thông tin cá nhân*: Người dùng, Người quản lý trang web có thể quản lý thông tin của mình, quản lý bao gồm việc: xem, thêm, sửa thông tin.
- *Xem thông tin của người dùng khác*: Người dùng có thể xem thông tin của người dùng khác bao gồm các thông tin cơ bản như tên, ảnh đại diện, ... và có

thể xem danh sách các bài đăng của người đó. Với người quản lý thì có thể xem các thông tin chi tiết hơn như số điện thoại, email, ...

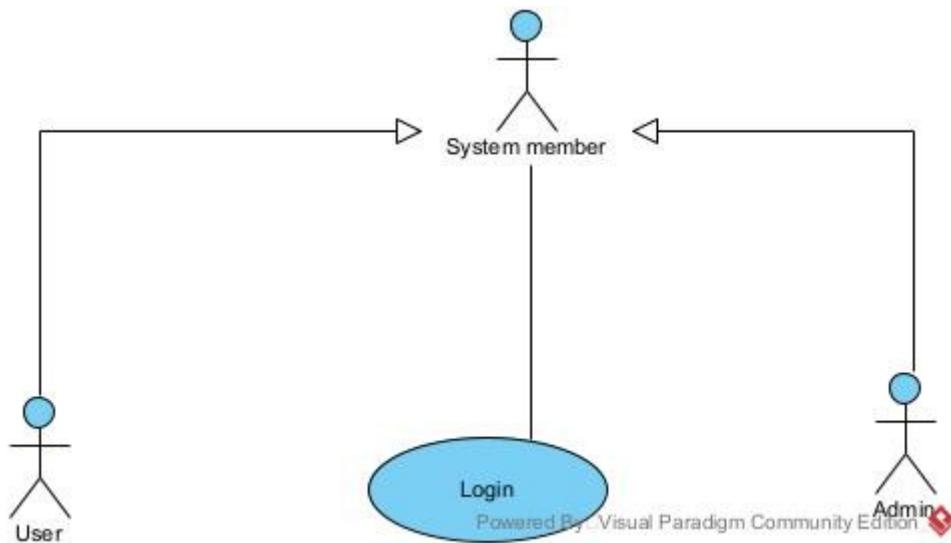
- *Chat*: Người dùng có thể nhắn tin cho nhau.
- *Xem danh sách bài đăng*: Người dùng và Người quản lý đều có thể xem danh sách các bài đăng. Người dùng chỉ xem được các bài đăng không bị khóa, chưa trao đổi. Người quản lý có thể xem đầy đủ mọi loại bài.
- *Nhận thông báo*: Người dùng có thể nhận thông báo từ hệ thống.
- *Quản lý bài đăng của người dùng*: Người dùng có thể quản lý bài đăng của mình, bao gồm: thêm, sửa, xóa bài đăng.
- *Xem lịch sử trao đổi*: Người dùng có thể xem lịch sử các lần trao đổi của mình.
- *Gửi yêu cầu trao đổi*: Người dùng có thể tạo yêu cầu trao đổi sản phẩm với một người dùng khác.
- *Tạo lịch hẹn*: Nếu chấp nhận một lượt trao đổi, người dùng có thể tạo lịch hẹn với người còn lại
- *Tìm kiếm bài đăng*: Người dùng và Quản lý đều có thể tìm kiếm bài đăng theo: tên, thể loại sản phẩm, thể loại bài đăng, khoảng thời gian.
- *Báo cáo vi phạm*: Người dùng có thể báo các bài đăng không phù hợp.
- *Quản lý các thể loại sản phẩm được đăng*: Quản lý có thể quản lý các hạng mục sản phẩm của hệ thống, bao gồm: thêm, sửa, xóa.
- *Quản lý người dùng của Admin*: Người quản lý có thể quản lý các tài khoản của người dùng, bao gồm: xem thông tin, khóa tài khoản.
- *Quản lý bài đăng của Admin*: Quản lý trang web có thể quản lý các bài đăng của hệ thống, bao gồm: xem, khóa bài đăng.
- *Xử lý các báo cáo vi phạm từ người dùng*: Người quản lý có thể xem các báo cáo của người dùng, từ đó có biện pháp xử lý yêu cầu báo cáo đó.
- *Kiểm duyệt bài đăng*: Người quản lý sẽ kiểm tra các bài đăng trước khi cho bài đăng đó hiện lên bảng tin.
- *Thống kê*: Người quản lý có thể xem một số thống kê của hệ thống.

Từ đó ta có biểu đồ use case như hình sau. Phân tiếp theo sẽ trình bày biểu đồ Use Case cho các chức năng chính như sau:



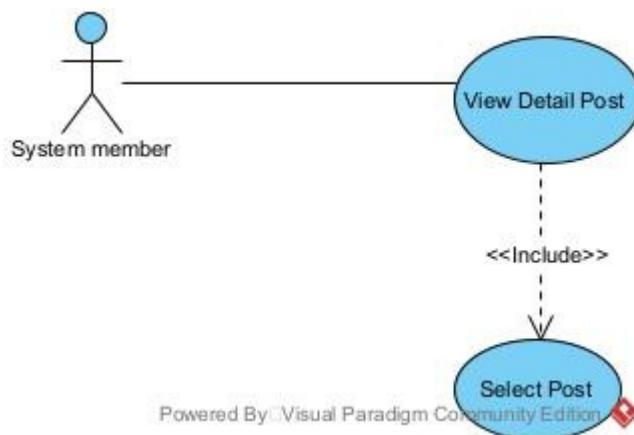
Hình 3.11: Biểu đồ Use Case tổng quát

3.1.2. Use case Đăng nhập



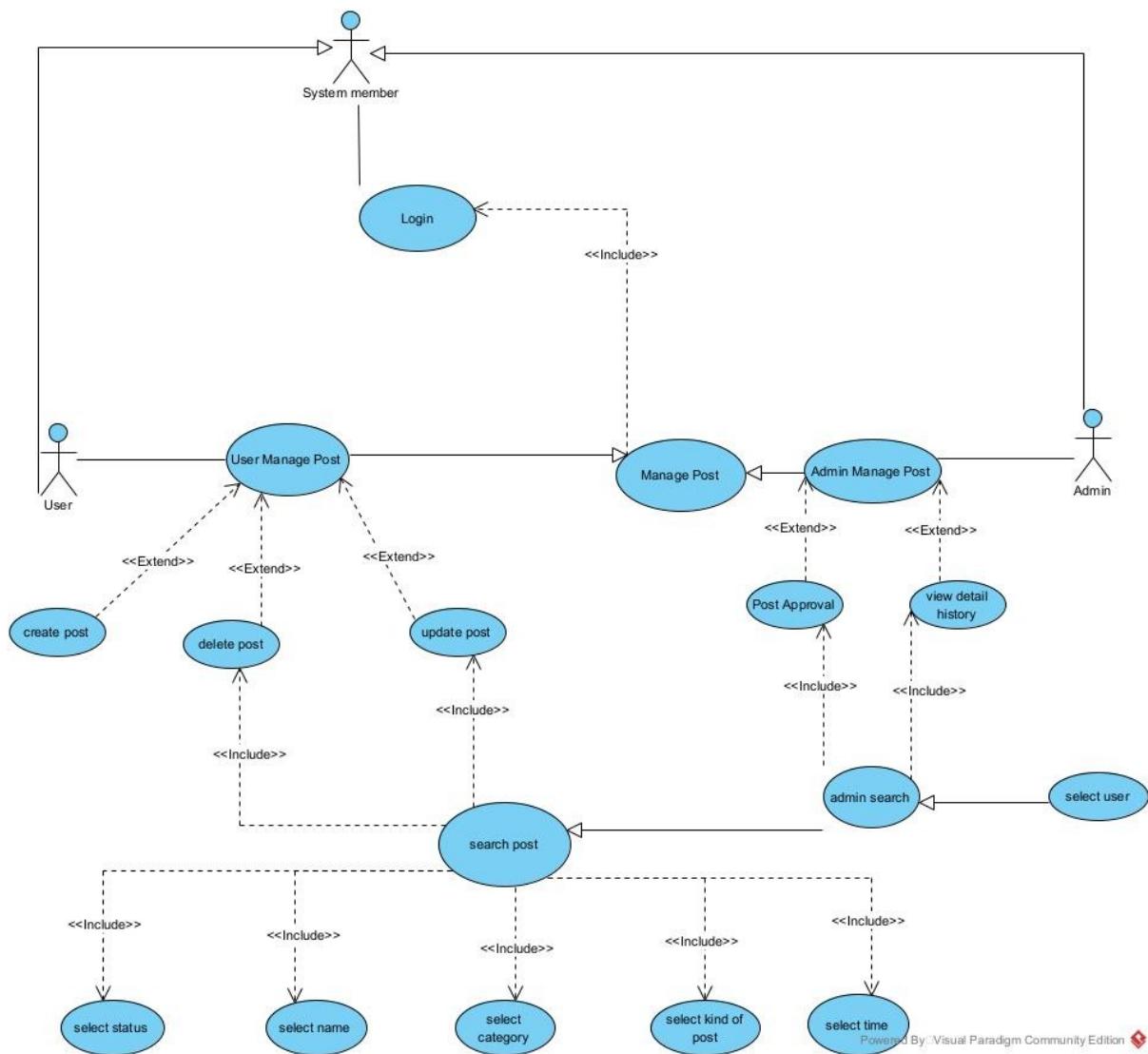
Hình 3.12: Biểu đồ Use case Đăng nhập

3.1.3. Use case Xem chi tiết bài đăng



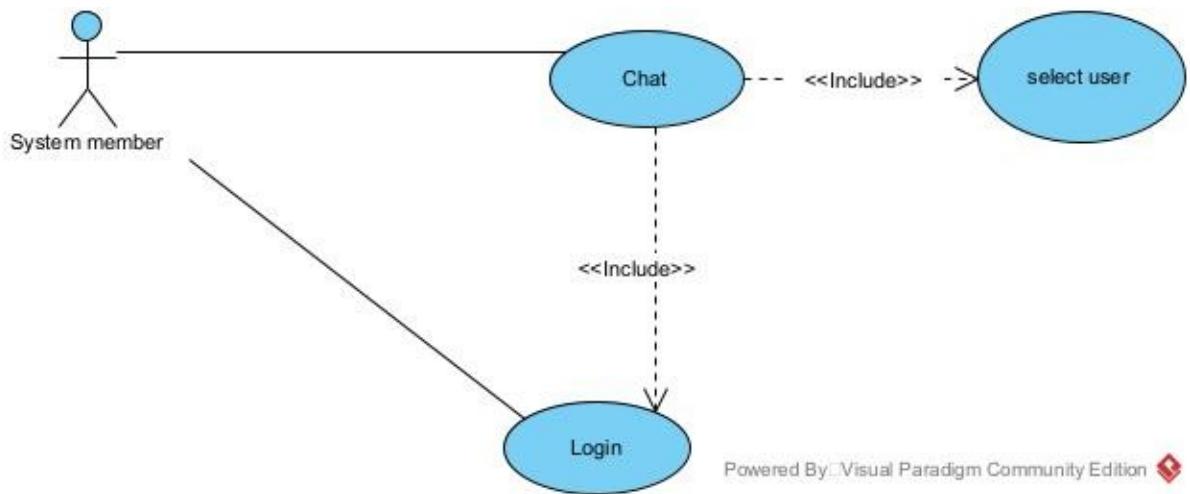
Hình 3.13: Biểu đồ Use case Xem chi tiết bài đăng

3.1.4. Use case Quản lý bài đăng



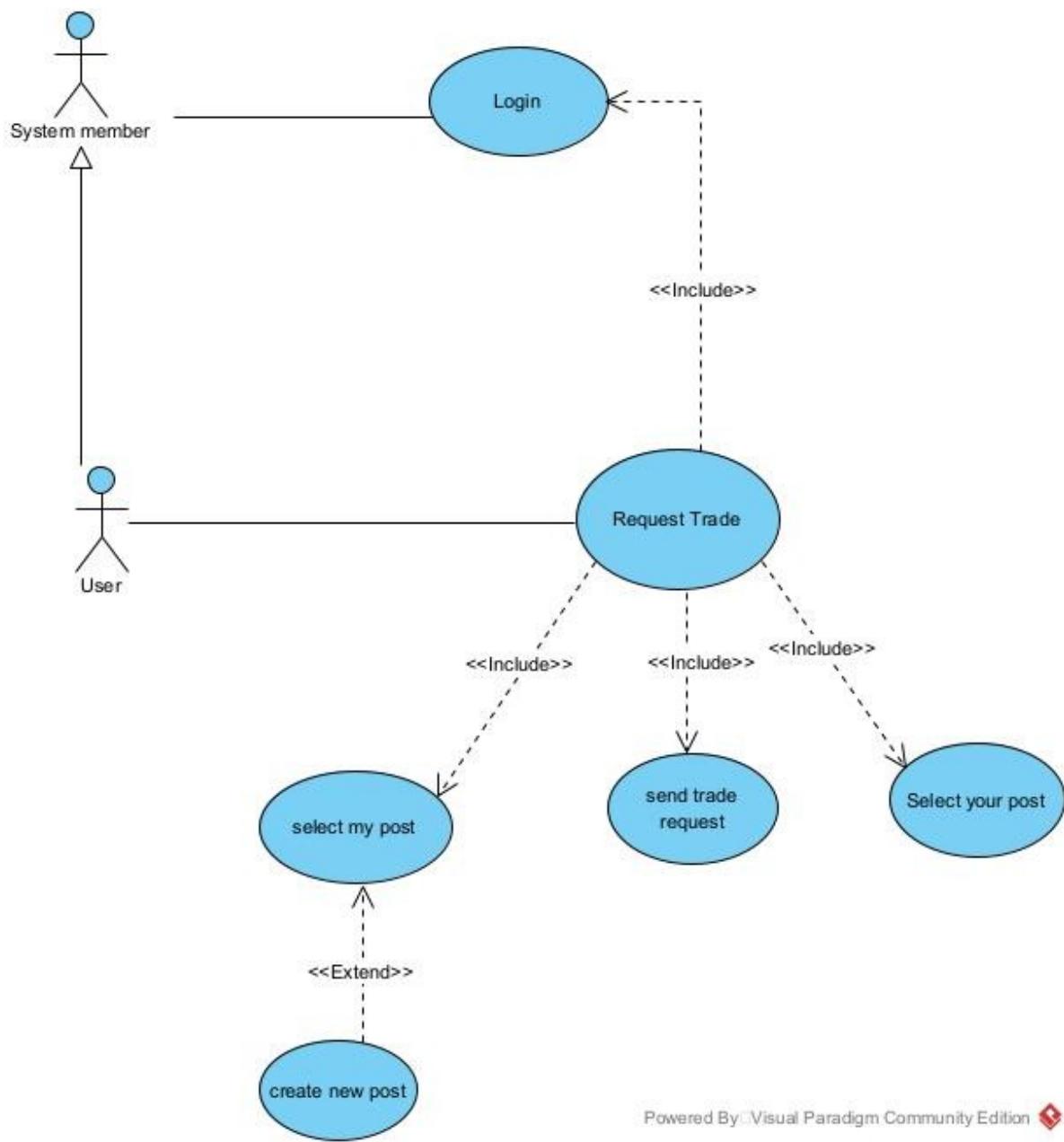
Hình 3.14: Biểu đồ Use case quản lý bài đăng

3.1.5. Use case Chat



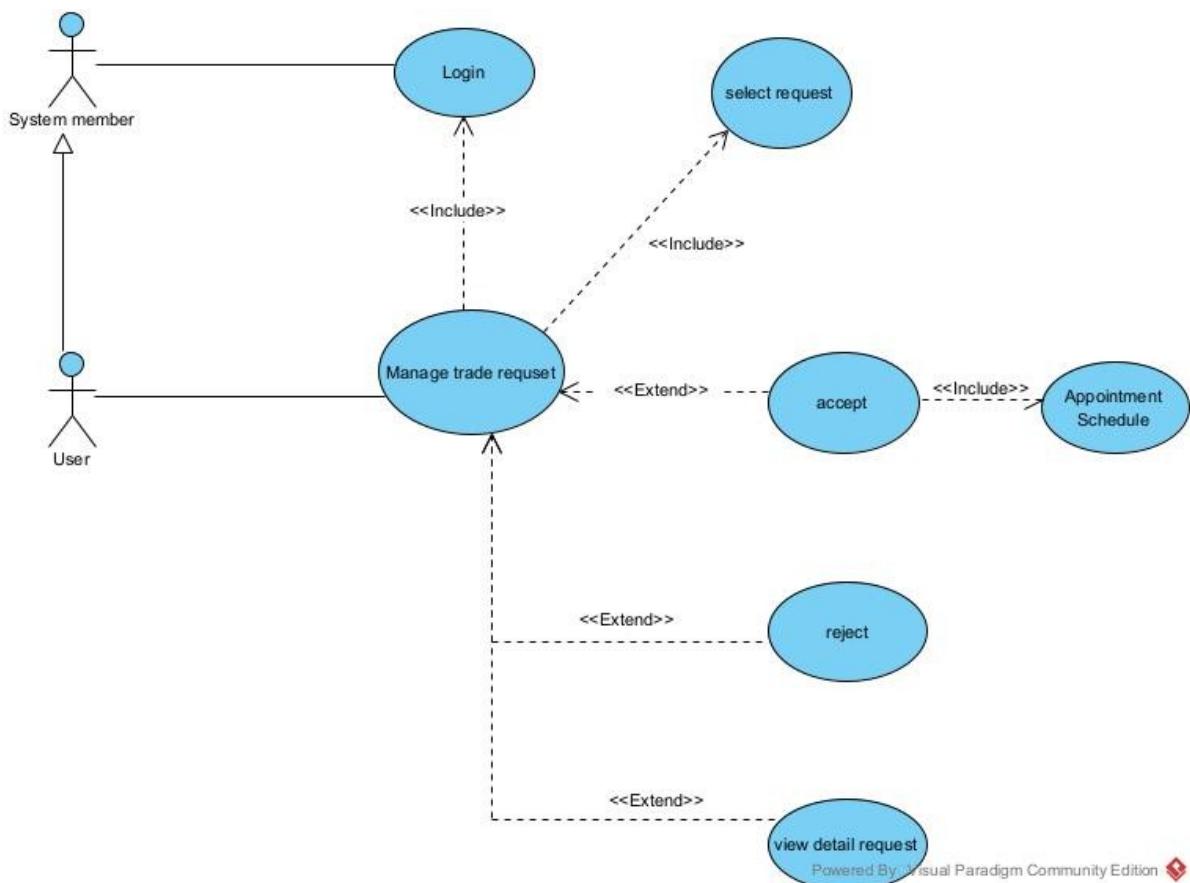
Hình 3.15: Biểu đồ Use Case Chat

3.1.6. Use case Yêu cầu trao đổi



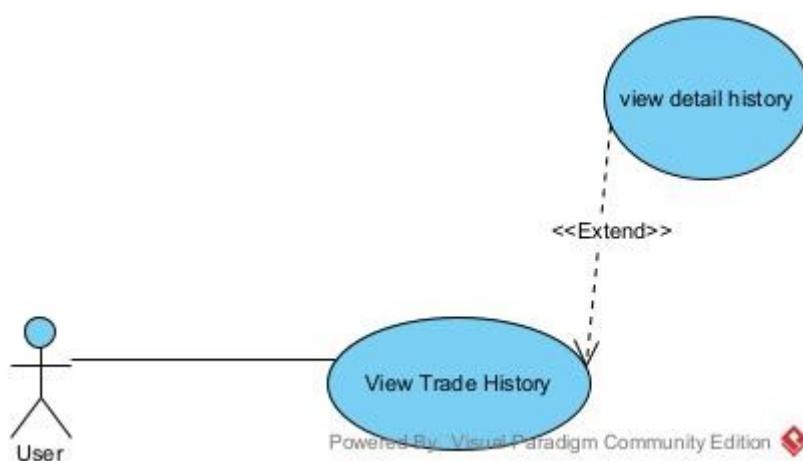
Hình 3.16: Biểu đồ Use case yêu cầu trao đổi

3.1.7. Use case Xử lý yêu cầu trao đổi



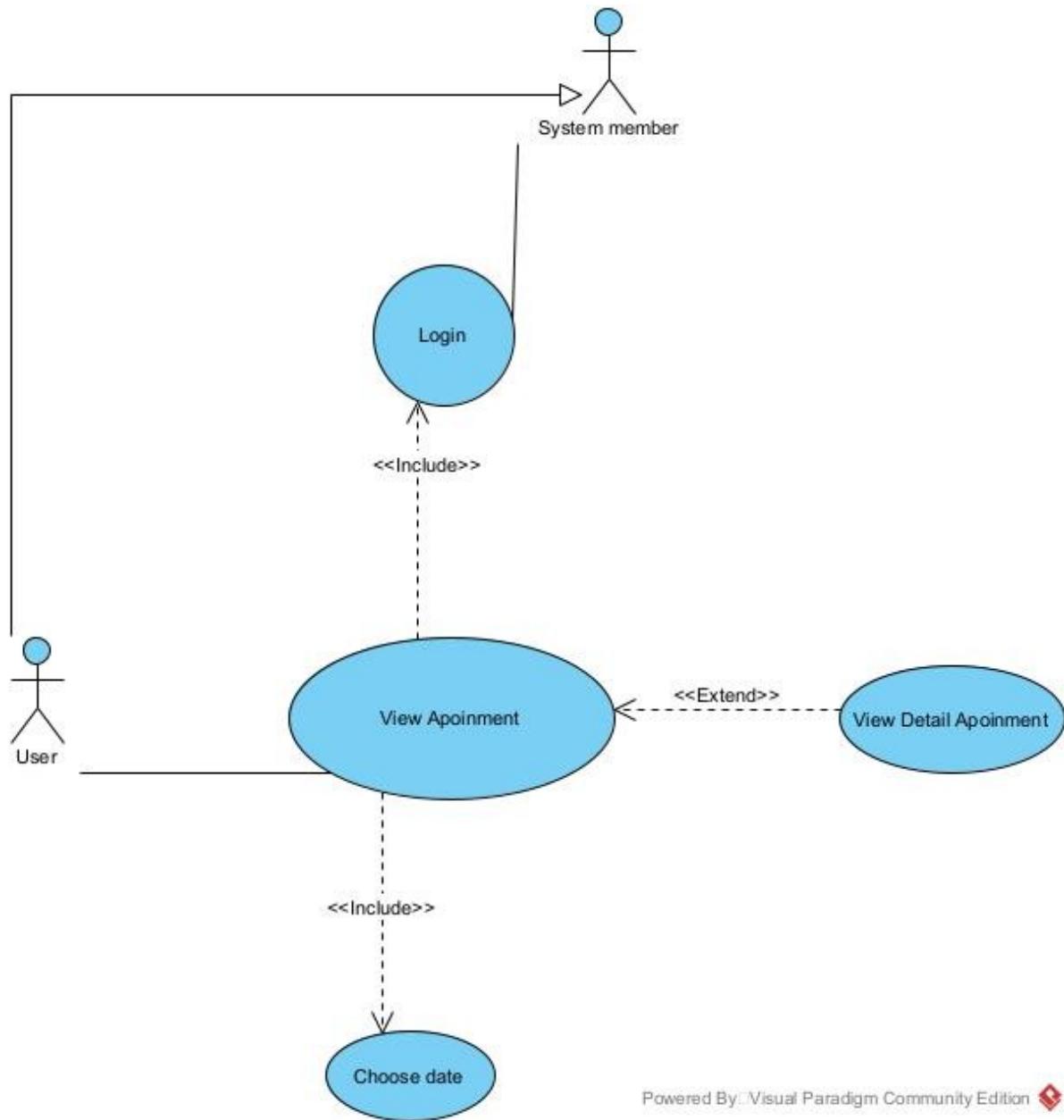
Hình 3.17: Biểu đồ Use case Xử lý yêu cầu trao đổi

3.1.8. Use case Xem lịch sử trao đổi



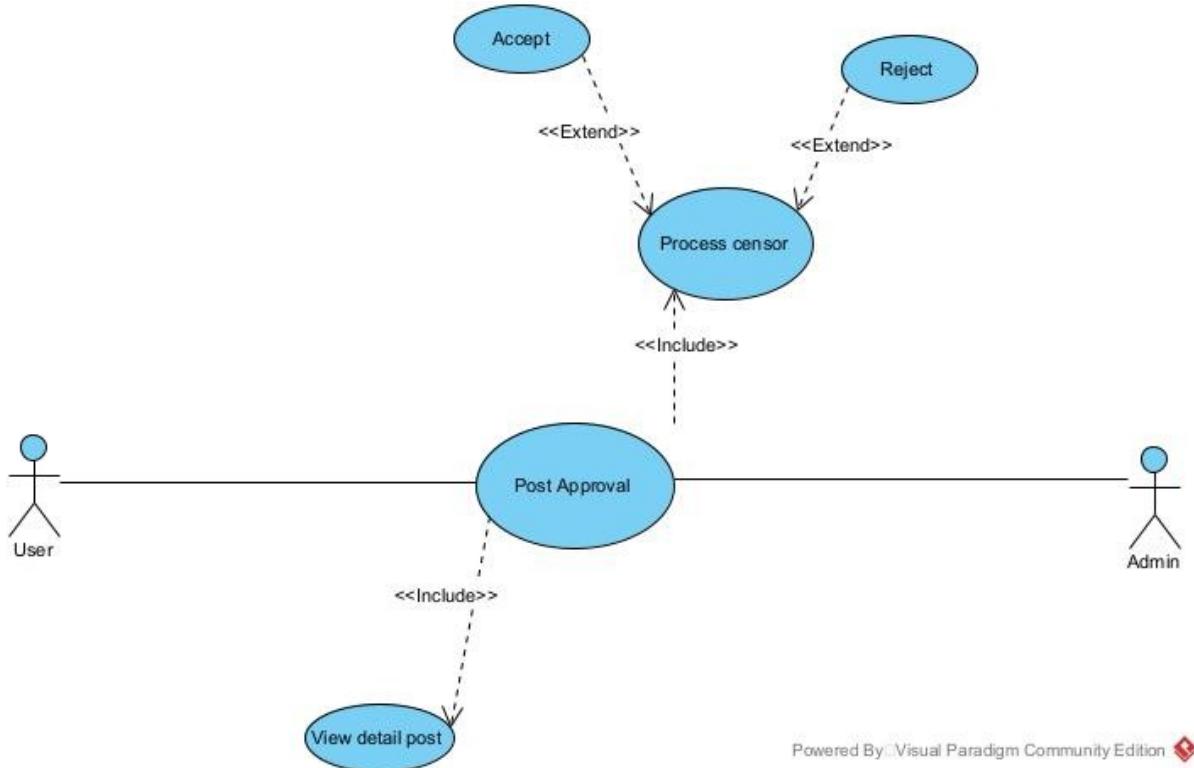
Hình 3.18: Biểu đồ Use case Xem lịch sử trao đổi

3.1.9. Use case Xem lịch hẹn



Hình 3.19: Biểu đồ Use case Xem lịch hẹn

3.1.10. Use case Kiểm duyệt



Hình 3.20: Biểu đồ Use case kiểm duyệt

3.2. Xây dựng kịch bản cho các chức năng

3.2.1. Kịch bản chính cho Use case Đăng nhập

Use case	Đăng nhập
Tác nhân	Thành viên hệ thống (Member)
Tiền điều kiện	Member đã có tài khoản
Kết quả	Đăng nhập thành công vào hệ thống
Kịch bản chính	<ol style="list-style-type: none"> Trên giao diện trang chủ, người dùng chọn đăng nhập Hệ thống hiển thị giao diện đăng nhập có ô nhập tài khoản, mật khẩu và nút đăng nhập Người dùng nhập thông tin: <ul style="list-style-type: none"> Tài khoản: abc123

	<ul style="list-style-type: none"> - Mật khẩu: 123456 - Sau đó ấn nút đăng nhập <p>4. Hệ thống hiển thị giao diện trang chủ với thông tin người dùng được hiển thị.</p>
Ngoại lệ	4. Nếu tài khoản bị khóa, hệ thống hiển thị thông báo tài khoản bị khóa.

Bảng 3.2: Kịch bản cho Use case Đăng nhập

3.2.2. Kịch bản chính cho Use case Xem chi tiết bài đăng

Use case	Xem chi tiết bài đăng															
Tác nhân	Thành viên hệ thống (Member)															
Tiền điều kiện	Member đã đăng nhập vào hệ thống															
Kết quả	Xem được thông tin chi tiết của bài đăng															
Kịch bản chính	<p>1. Trên giao diện trang chủ của hệ thống, người dùng nhập:</p> <ul style="list-style-type: none"> - Tên sản phẩm: “Quần bò” - Thể loại: “Quần” - Ngày đăng: từ 30-10-2023 đến 5-11-2023 - Sắp xếp theo: lượt xem từ cao xuống thấp <p>2. Hệ thống hiển thị danh sách bài đăng có thông tin như đã nhập</p> <table border="1"> <thead> <tr> <th>STT</th> <th>Tên</th> <th>Thể loại</th> <th>Ngày đăng</th> <th>Số lượt xem</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>Quần bò Boo</td> <td>Quần</td> <td>1-11-2023</td> <td>100</td> </tr> <tr> <td>2</td> <td>Quần bò mua ngoài chợ</td> <td>Quần</td> <td>4-11-2023</td> <td>10</td> </tr> </tbody> </table> <p>3. Người dùng chọn sản phẩm “Quần bò Boo”</p>	STT	Tên	Thể loại	Ngày đăng	Số lượt xem	1	Quần bò Boo	Quần	1-11-2023	100	2	Quần bò mua ngoài chợ	Quần	4-11-2023	10
STT	Tên	Thể loại	Ngày đăng	Số lượt xem												
1	Quần bò Boo	Quần	1-11-2023	100												
2	Quần bò mua ngoài chợ	Quần	4-11-2023	10												

	<p>4. Hệ thống hiển thị thông tin chi tiết sản phẩm bao gồm:</p> <ul style="list-style-type: none"> - Tên: Quần bò Boo - Trạng thái: Chưa trao đổi - Người đăng: linh123 - Địa chỉ: Tôi có thể đổi trong địa phận hà nội - Mô tả: Quần mới mua 1 tháng mặc đúng 2 lần, cần đổi vì màu không hợp - Yêu cầu: Nếu được thì tôi muốn đổi lấy một chiếc jogger tương tự - Danh mục sản phẩm: Quần - Danh mục bài viết: Sản phẩm muốn trao đổi - Số lượt xem: 100 - Ngày đăng bài: 1/11/2023 <p>Và có nút trao đổi</p>
Ngoại lệ	2. Không có bài đăng tương ứng

Bảng 3.3: Kịch bản cho Use Case Xem chi tiết bài đăng

3.2.3. Kịch bản chính cho Use case Quản lý bài đăng - Tạo bài đăng

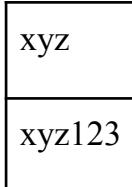
Use case	Tạo bài đăng của người dùng
Tác nhân	Người dùng (User)
Tiền điều kiện	Người dùng đã đăng nhập vào hệ thống
Kết quả	Bài viết được tạo
Kịch bản chính	<ol style="list-style-type: none"> 1. Trên giao diện hệ thống, người dùng chọn “Tạo bài đăng” 2. Hệ thống hiển thị giao diện tạo bài đăng gồm có các thông tin <ul style="list-style-type: none"> - Tên bài viết - Danh mục sản phẩm (Loại sản phẩm) - Danh mục bài đăng (Đò muôn đổi/Đò tìm kiếm) - Địa chỉ (Nơi bạn có thể hẹn trao đổi) - Mô tả sản phẩm - Yêu cầu về món hàng bạn sẽ trao đổi

	<ul style="list-style-type: none"> - Ảnh đại diện - Danh sách ảnh - Nút thêm bài viết <p>3. Người dùng nhập</p> <ul style="list-style-type: none"> - Tên: Áo phao cũ - Danh mục sản phẩm: Áo - Danh mục bài viết: Sản phẩm muốn trao đổi - Địa chỉ: Tôi có thể trao đổi trong khu vực quận Long Biên Hà Nội - Mô tả: Áo hàng ABC mới mua 2 ngày nhưng mặc chật - Yêu cầu: muốn đổi cái áo tương tự nhưng size XL, nếu không thì đổi sang một cái Vasity cũng được - Chọn ảnh đại diện và danh sách ảnh <p>Người dùng ấn Thêm bài viết</p> <p>4. Hệ thống hiển thị thêm bài viết thành công</p> <p>5. Người dùng ấn vào Quản lý bài đăng</p> <p>6. Hệ thống hiển thị danh sách bài đăng</p>
Ngoại lệ	

Bảng 3.4: Kịch bản Use Case Tạo bài đăng

3.2.4. Kịch bản chính cho Use case Chat

Use case	Chat
Tác nhân	Thành viên hệ thống (Member)

Tiền điều kiện	Member đã đăng nhập vào hệ thống
Kết quả	Gửi tin nhắn thành công
Kịch bản chính	<p>1. Trên giao diện trang chủ của hệ thống, người dùng chọn chat</p> <p>2. Hệ thống hiển thị giao diện chat gồm có ô tìm kiếm người dùng, danh sách người dùng tìm được, hộp chat người dùng được chọn.</p> <p>3. Người dùng nhập tên vào ô tìm kiếm “xyz”</p> <p>4. Hệ thống hiển thị danh sách người dùng có “xyz” trong tên</p>  <p>5. Người dùng chọn xyz</p> <p>6. Hệ thống hiện hộp chat với người dùng xyz, gồm các tin nhắn đã nhắn</p> <p>7. Người dùng gõ “Xin chào” vào ô nhập tin nhắn và ấn gửi</p> <p>8. Hệ thống gửi tin nhắn “Xin chào” cho người dùng “xyz”</p>
Ngoại lệ	4. Không tìm được người dùng

Bảng 3.5: Kịch bản chính cho Use case Chat

3.2.5. Kịch bản chính cho Use case Yêu cầu trao đổi

Use case	Yêu cầu trao đổi
Tác nhân	Người dùng (User)
Tiền điều kiện	Người dùng đã đăng nhập vào hệ thống
Kết quả	Gửi yêu cầu trao đổi cho đối phương thành công
Kịch bản chính	<p>1. Trên giao diện trang chủ có ô tìm kiếm bài đăng, người dùng nhập tên “Quân”</p> <p>2. Hệ thống hiển thị danh sách bài đăng có tên “Quân”</p>

	STT	Tên	Ngày đăng	Người đăng				
1	Quần jogger	30/10/2023	abc123					
2	Quần bò	25/10/2023	xyz123					
<p>3. Người dùng ấn vào Quần jogger</p> <p>4. Hệ thống hiển thị chi tiết bài đăng, có nút “Trao đổi”</p> <p>5. Người dùng chọn Trao đổi</p> <p>6. Hệ thống hiển thị giao diện tạo yêu cầu trao đổi bao gồm:</p> <ul style="list-style-type: none"> - Nội dung trao đổi - Danh sách bài viết của bạn 								
<table border="1"> <thead> <tr> <th>STT</th> <th>Tên</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>Áo sơ mi</td> </tr> </tbody> </table> <ul style="list-style-type: none"> - Nút thêm sản phẩm 					STT	Tên	1	Áo sơ mi
STT	Tên							
1	Áo sơ mi							
<p>7. Người dùng nhập:</p> <ul style="list-style-type: none"> - Nội dung trao đổi: tôi đã xem profile của bạn, có vẻ tôi có cái áo bạn đang tìm kiếm <p>và chọn Áo sơ mi rồi ấn trao đổi</p>								
<p>8. Hệ thống hiển thị tạo yêu cầu thành công</p> <p>9. Người dùng ấn vào trang cá nhân chọn lịch sử trao đổi</p> <p>10. Hệ thống hiển thị danh sách yêu cầu trao đổi của bạn</p>								
	STT	Sản phẩm của bạn	Sản phẩm bạn muốn đổi	Ngày gửi yêu cầu	Trạng thái			
	1	Áo sơ mi	Quần Jogger	30/10/2023	Chờ xác nhận			
Ngoại lệ	<p>6. Hệ thống hiện danh sách sản phẩm của bạn trong</p> <p>6.1. Người dùng chọn “Thêm sản phẩm”</p> <p>6.2. Hệ thống hiển thị giao diện Thêm sản phẩm</p> <p>6.3. Từ bước này tương tự use case Thêm bài đăng</p>							

Bảng 3.6: Kịch bản Use Case Yêu cầu trao đổi

3.2.6. Kịch bản chính cho Use case Xử lý yêu cầu trao đổi - Xác nhận

Use case	Xử lý yêu cầu trao đổi - Xác nhận										
Tác nhân	Người dùng (User)										
Tiền điều kiện	Người dùng đã đăng nhập vào hệ thống										
Kết quả	Yêu cầu trao đổi được xử lý (Chấp nhận/Từ chối)										
Kịch bản chính	<p>1. Sau khi đăng nhập, người dùng chọn Xử lý yêu cầu trao đổi</p> <p>2. Hệ thống hiển thị danh sách yêu cầu trao đổi gửi tới bạn</p> <table border="1"> <thead> <tr> <th>STT</th> <th>Người gửi</th> <th>Sản phẩm của họ</th> <th>Sản phẩm của bạn</th> <th>Xử lý</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>linh123</td> <td>Quần jogger</td> <td>Áo sơ mi</td> <td>Xác nhận Từ chối Chi tiết</td> </tr> </tbody> </table> <p>3. Người dùng chọn xác nhận</p> <p>4. Hệ thống hiển thị giao diện đặt lịch bao gồm: Ngày trao đổi, địa chỉ</p> <p>5. Người dùng nhập</p> <ul style="list-style-type: none"> - Ngày trao đổi: 1/11/2023 - Địa chỉ: Quán cf abc123 trên đường Nguyễn Văn Cừ, Long Biên - Ân xá nhận <p>6. Hệ thống thông báo đặt lịch thành công, yêu cầu đổi được chuyển sang trạng thái “Chờ trao đổi”</p>	STT	Người gửi	Sản phẩm của họ	Sản phẩm của bạn	Xử lý	1	linh123	Quần jogger	Áo sơ mi	Xác nhận Từ chối Chi tiết
STT	Người gửi	Sản phẩm của họ	Sản phẩm của bạn	Xử lý							
1	linh123	Quần jogger	Áo sơ mi	Xác nhận Từ chối Chi tiết							
Ngoại lệ	2. Hệ thống hiển thị danh sách trống										

Bảng 3.7: Kịch bản Use Case Xác nhận trao đổi

3.2.7. Kịch bản chính cho Use case Xem lịch sử trao đổi

Use case	Xem lịch sử trao đổi																		
Tác nhân	Người dùng																		
Tiền điều kiện	Người dùng đã đăng nhập vào hệ thống																		
Kết quả	Xem chi tiết một lần trao đổi																		
Kịch bản chính	<ol style="list-style-type: none"> 1. Trên giao diện trang chủ, người dùng chọn xem lịch sử trao đổi 2. Hệ thống hiển thị lịch sử trao đổi có bộ lọc ngày bắt đầu, ngày kết thúc 3. Người dùng chọn từ ngày 25/10/2023 - 30/10/2023 rồi ấn lọc 4. Hệ thống hiển thị danh sách các lần trao đổi <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th>STT</th> <th>Loại</th> <th>Sản phẩm của bạn</th> <th>Sản phẩm của đối phương</th> <th>Ngày yêu cầu</th> <th>Trạng thái</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>Bạn nhận</td> <td>Quần bò</td> <td>Quần đùi</td> <td>26/10/2023</td> <td>Từ chối</td> </tr> <tr> <td>2</td> <td>Bạn gửi</td> <td>Áo len</td> <td>Áo phao</td> <td>30/10/2023</td> <td>Chờ hẹn</td> </tr> </tbody> </table> <ol style="list-style-type: none"> 5. Người dùng chọn dòng 1 6. Hệ thống hiển thị thông tin chi tiết về yêu cầu trao đổi đó 	STT	Loại	Sản phẩm của bạn	Sản phẩm của đối phương	Ngày yêu cầu	Trạng thái	1	Bạn nhận	Quần bò	Quần đùi	26/10/2023	Từ chối	2	Bạn gửi	Áo len	Áo phao	30/10/2023	Chờ hẹn
STT	Loại	Sản phẩm của bạn	Sản phẩm của đối phương	Ngày yêu cầu	Trạng thái														
1	Bạn nhận	Quần bò	Quần đùi	26/10/2023	Từ chối														
2	Bạn gửi	Áo len	Áo phao	30/10/2023	Chờ hẹn														
Ngoại lệ	4. Hệ thống hiển thị danh sách rỗng																		

Bảng 3.8: Kịch bản Use Case Xem lịch sử trao đổi

3.2.8. Kịch bản chính cho Use case Xem lịch hẹn

Use case	Xem lịch hẹn
----------	--------------

Tác nhân	Người dùng (User)									
Tiền điều kiện	Người dùng đã đăng nhập vào hệ thống									
Kết quả	Xem được chi tiết một buổi hẹn									
Kịch bản chính	<p>1. Trên giao diện hệ thống, người dùng ấn vào “Lịch”</p> <p>2. Hệ thống hiển thị lịch tháng này và những ngày có hẹn</p> <p>3. Người dùng ấn vào một ngày có hẹn, ví dụ 30/10/2023</p> <p>4. Hệ thống hiển thị danh sách các cuộc hẹn trong ngày 30/10/2023</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th>STT</th><th>Sản phẩm của đối phương</th><th>Tài khoản đối phương</th></tr> </thead> <tbody> <tr> <td>1</td><td>Áo len</td><td>linh123</td></tr> <tr> <td>2</td><td>Giày da</td><td>huy456</td></tr> </tbody> </table> <p>5. Người dùng chọn dòng 1</p> <p>6. Hệ thống hiển thị thông tin chi tiết của cuộc hẹn</p> <ul style="list-style-type: none"> - Sản phẩm bạn cần mang: Quần bò - Sản phẩm của đối phương: Áo len - Tài khoản đối phương: linh123 - Địa điểm: Quán P Coffee trường PTIT - Hà Nội - Thời gian: 9 giờ sáng - 10 giờ sáng - Ghi chú: Bạn đến trước cứ gọi nước giúp tôi nhé, tôi uống trà chanh 	STT	Sản phẩm của đối phương	Tài khoản đối phương	1	Áo len	linh123	2	Giày da	huy456
STT	Sản phẩm của đối phương	Tài khoản đối phương								
1	Áo len	linh123								
2	Giày da	huy456								
Ngoại lệ	<p>3. Người dùng ấn vào một ngày không có hẹn</p> <p>3.1. Hệ thống hiển thị danh sách rỗng</p>									

Bảng 3.9: Kịch bản Use Case Xem lịch hẹn

3.2.9. Kịch bản chính cho Use case Kiểm duyệt bài đăng thủ công

Use case	Kiểm duyệt bài đăng
Tác nhân	Quản lý trang web (Admin)

Tiền điều kiện	Admin đã đăng nhập vào hệ thống												
Kết quả	Bài đăng đã được xử lý kiểm duyệt												
Kịch bản chính	<p>1. Admin vào trang quản lý chọn “Kiểm duyệt bài đăng”</p> <p>2. Hệ thống hiển thị danh sách bài đăng gồm các thông tin:</p> <ul style="list-style-type: none"> - Tên bài đăng - Ngày đăng (từ ngày - đến ngày) - Người đăng - Hoạt động (Duyệt, Khóa, Chi tiết) - Bộ lọc (Tên, Khoảng ngày, Danh mục bài, Danh mục sản phẩm) <p>3. Admin nhập thông tin để tìm kiếm</p> <ul style="list-style-type: none"> - Ngày đăng: 30/10/2023 - 5/11/2023 - Danh mục sản phẩm: Quần - Danh mục bài đăng: Sản phẩm trao đổi - Án tìm <p>4. Hệ thống hiển thị danh sách bài đăng</p> <table border="1"> <thead> <tr> <th>STT</th><th>Tên</th><th>Ngày đăng</th><th>Loại bài</th><th>Loại sản phẩm</th><th>Hoạt động</th></tr> </thead> <tbody> <tr> <td>1</td><td>Quần jogger</td><td>30/10/2023</td><td>Sản phẩm trao đổi</td><td>Quần</td><td>Duyệt Khóa Chi tiết</td></tr> </tbody> </table> <p>5. Admin chọn xem chi tiết để xem chi tiết bài đăng Quần jogger</p> <p>6. Hệ thống hiển thị chi tiết bài đăng</p>	STT	Tên	Ngày đăng	Loại bài	Loại sản phẩm	Hoạt động	1	Quần jogger	30/10/2023	Sản phẩm trao đổi	Quần	Duyệt Khóa Chi tiết
STT	Tên	Ngày đăng	Loại bài	Loại sản phẩm	Hoạt động								
1	Quần jogger	30/10/2023	Sản phẩm trao đổi	Quần	Duyệt Khóa Chi tiết								

	<p>7. Admin kiểm tra thông tin rồi ấn quay lại</p> <p>8. Hệ thống quay về danh sách bài đăng</p> <p>9. Admin chọn Khóa</p> <p>10. Hệ thống hiển thị giao diện xác nhận khóa có ô nhập lý do khóa</p> <p>11. Admin nhập lý do: Bài đăng không có ảnh của sản phẩm, ẩn xác nhận</p> <p>12. Hệ thống thông báo khóa bài đăng thành công và thông báo cho người dùng</p>
Ngoại lệ	4. Hệ thống hiển thị danh sách trống

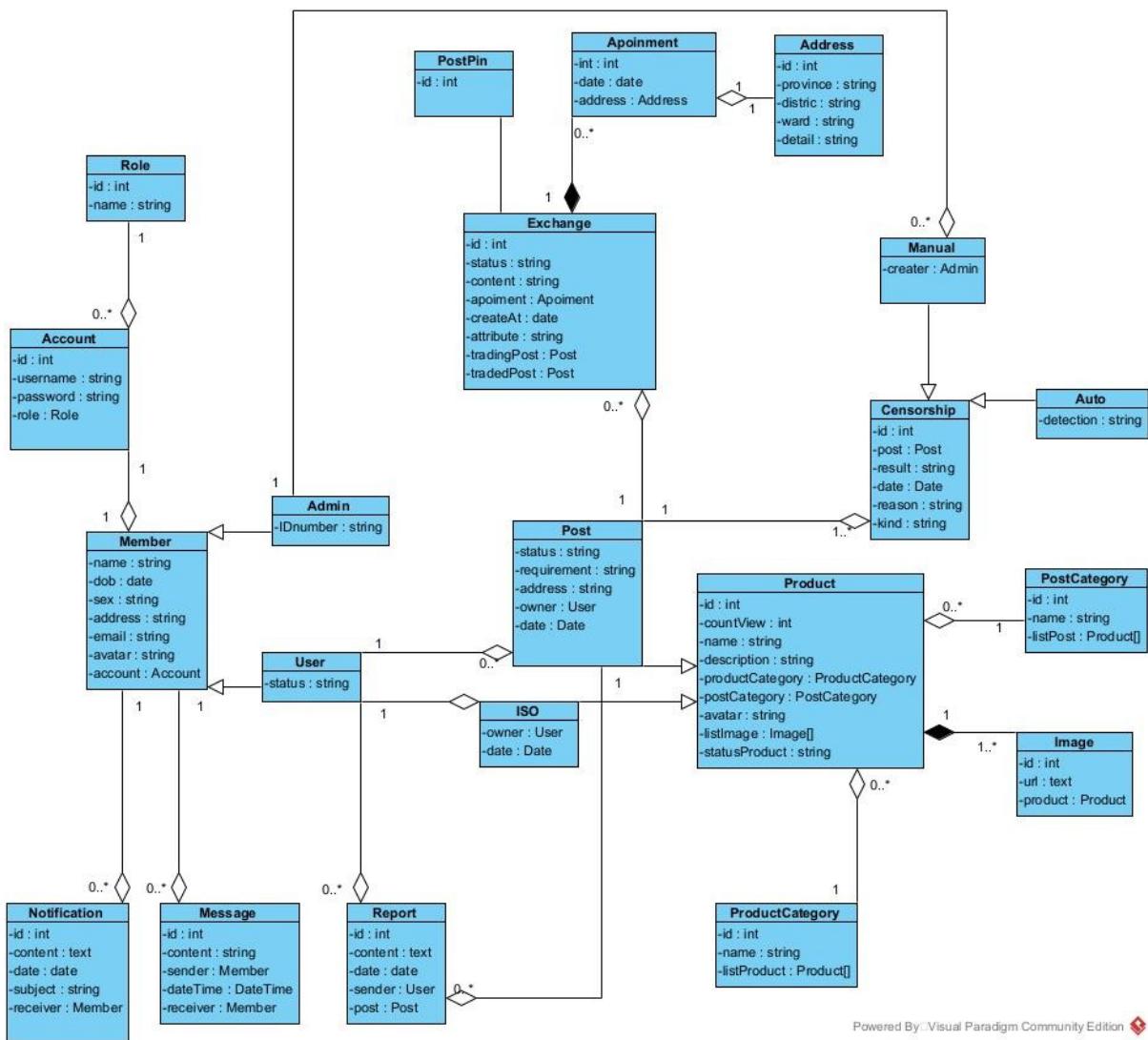
Bảng 3.10: Kịch bản Use Case Kiểm duyệt thủ công

3.2.10. Kịch bản chính cho Use case Kiểm duyệt bài đăng tự động

Use case	Kiểm duyệt bài đăng tự động
Tác nhân	Người dùng (User)
Tiền điều kiện	Người dùng đã đăng nhập vào hệ thống
Kết quả	Bài đăng được kết luận: Duyệt, Khóa hoặc chuyển cho Quản lý trang web duyệt
Kịch bản chính	<p>1. Người dùng tạo bài đăng</p> <p>2. Hệ thống lưu bài đăng vào cơ sở dữ liệu, đồng thời gửi danh sách ảnh của bài đăng cho hệ thống dự đoán</p> <p>3. Hệ thống dự đoán trả kết quả cho hệ thống chính</p> <p>4. Hệ thống chính kết luận xét duyệt và lưu lần xét duyệt vào cơ sở dữ liệu</p> <p>5. Hệ thống sửa đổi trạng thái bài đăng theo kết quả xét duyệt</p>
Ngoại lệ	

Bảng 3.11: Kịch bản cho Use case kiểm duyệt tự động

3.3. Biểu đồ lớp phân tích

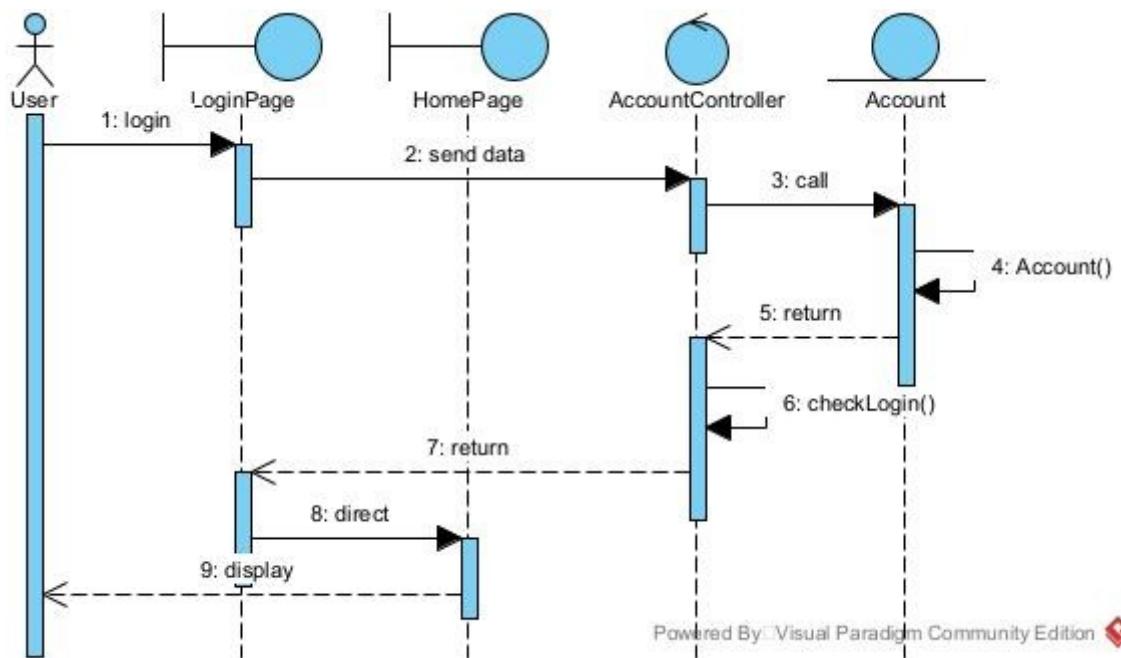


Powered By: Visual Paradigm Community Edition

Hình 3.21: Biểu đồ lớp phân tích

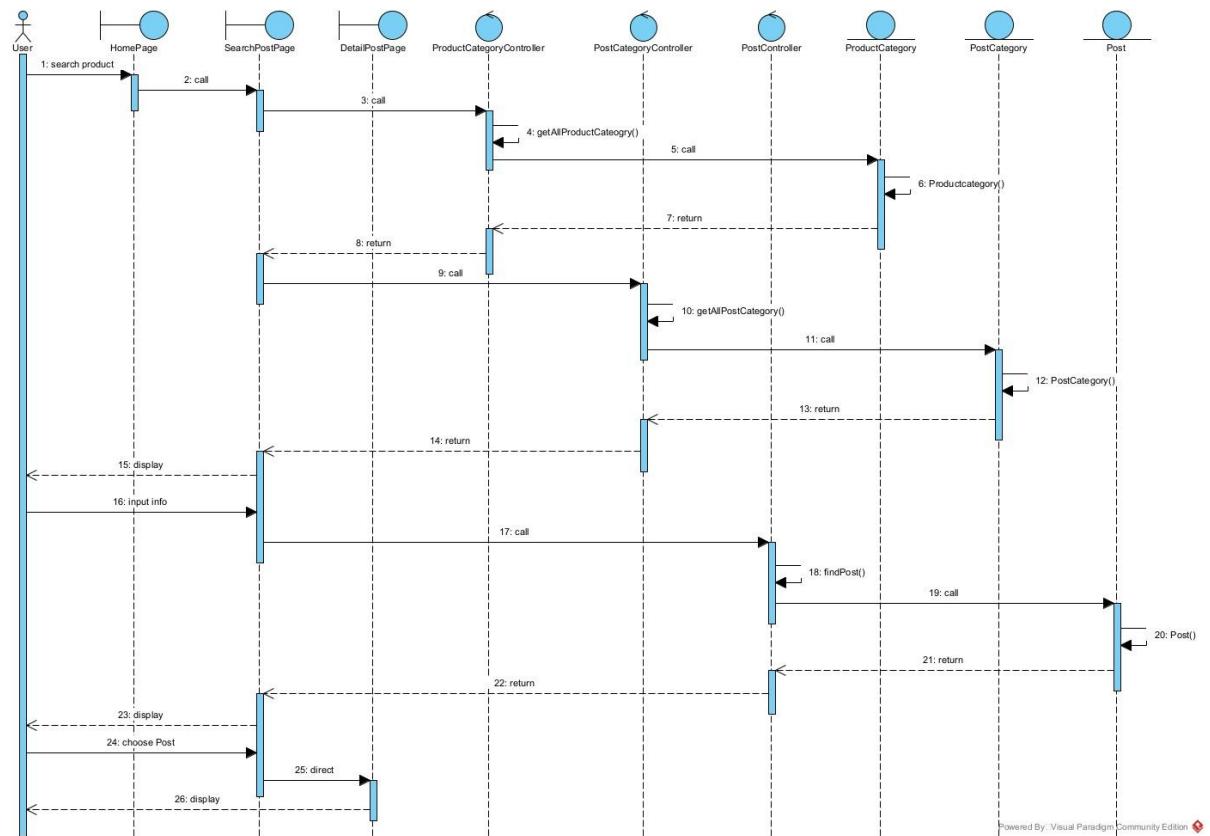
3.4. Thiết kế biểu đồ tuần tự

3.4.1. Đăng nhập



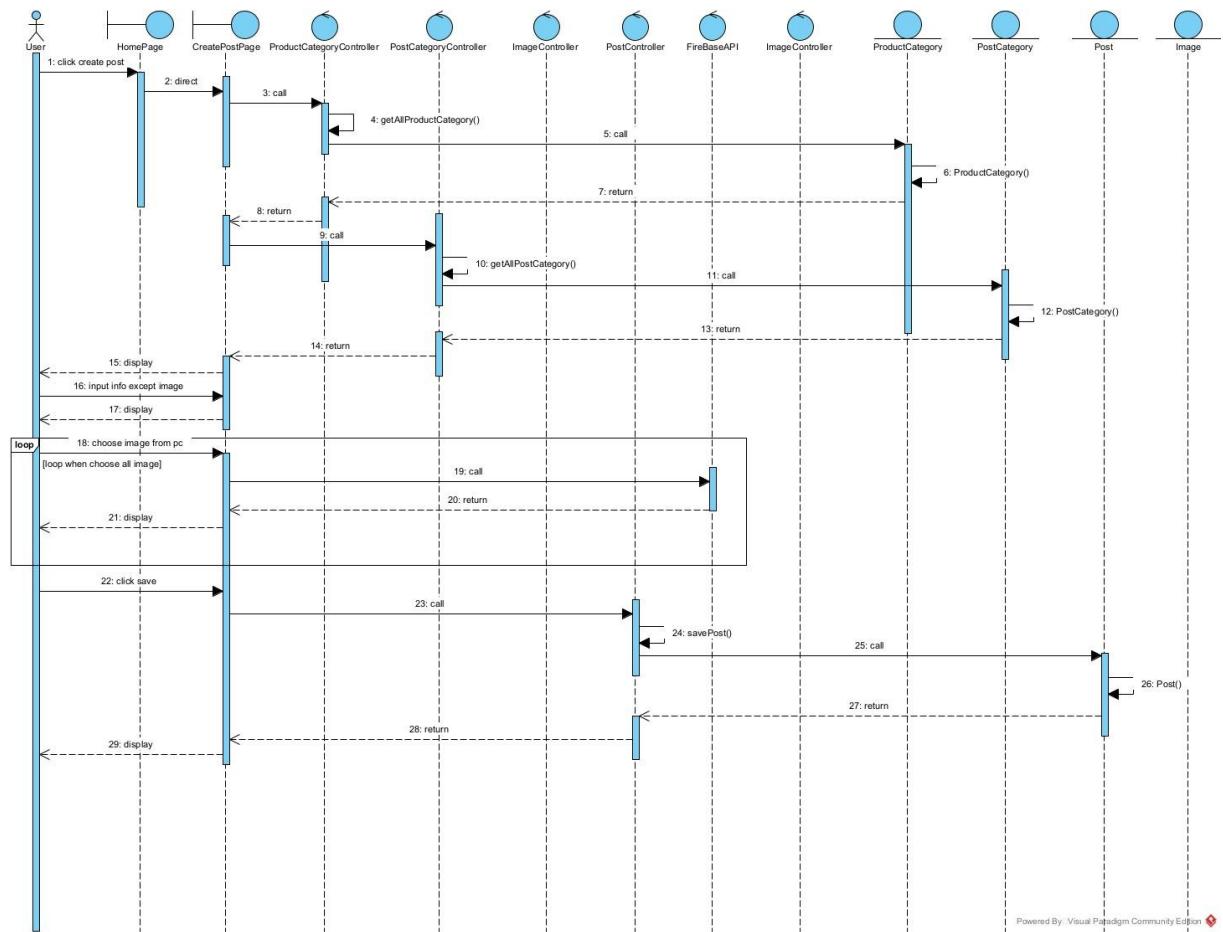
Hình 3.22: Biểu đồ tuần tự chức năng Đăng nhập

3.4.2. Xem chi tiết bài đăng



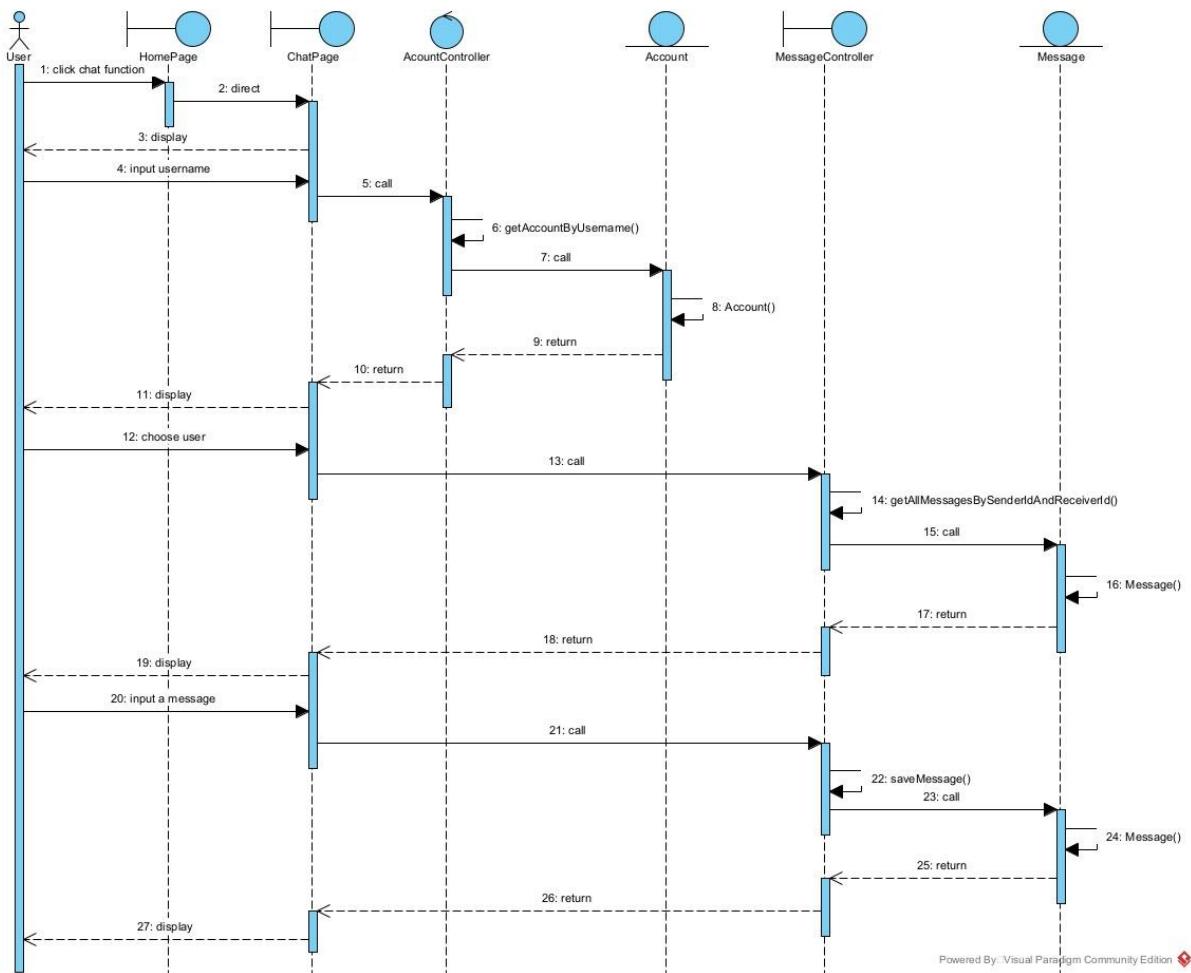
Hình 3.23: Biểu đồ tuần tự chức năng Xem chi tiết bài đăng

3.4.3. Quản lý bài đăng - Tạo bài đăng



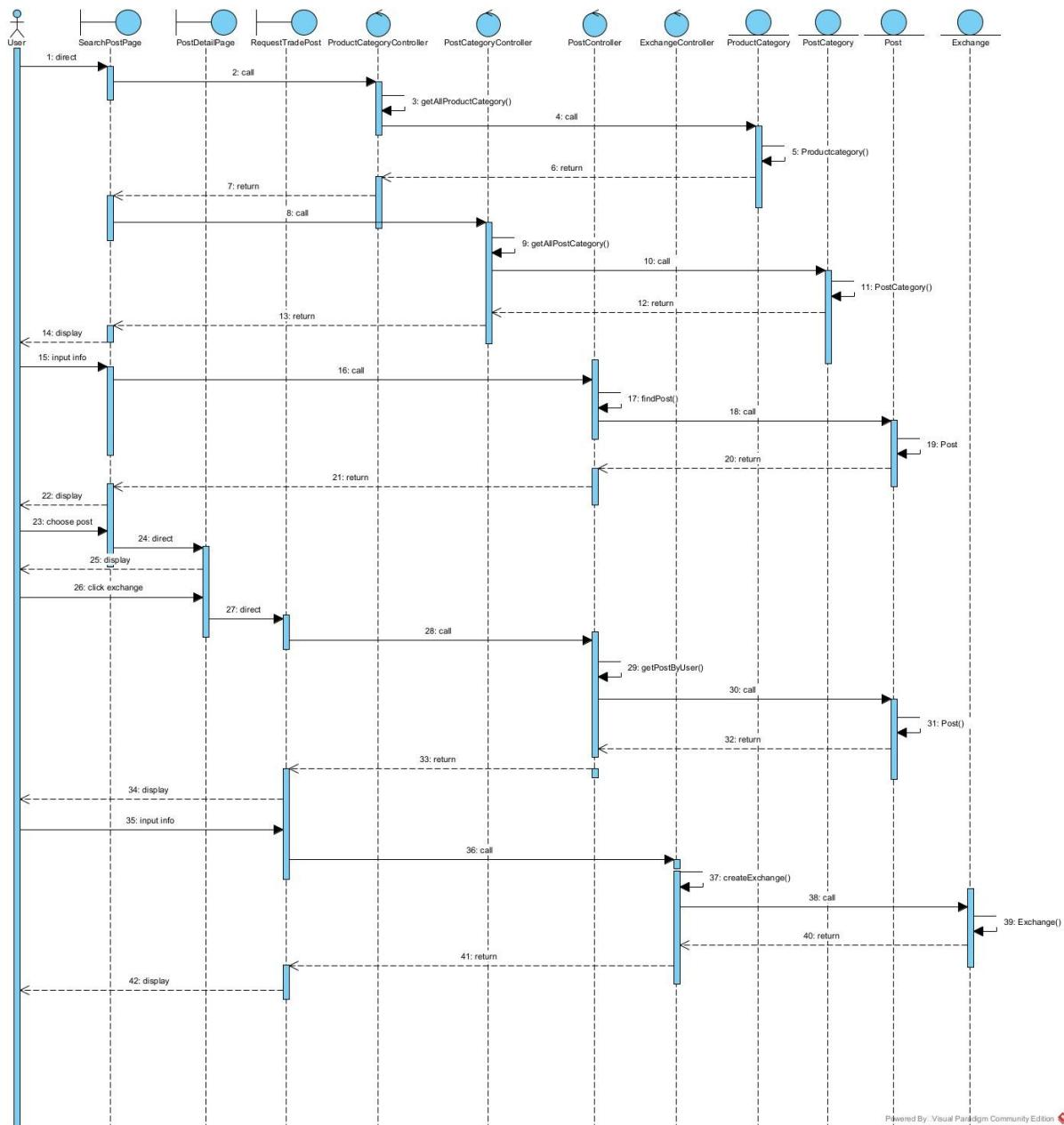
Hình 3.24: Biểu đồ tuần tự chức năng tạo bài đăng

3.4.4. Chat



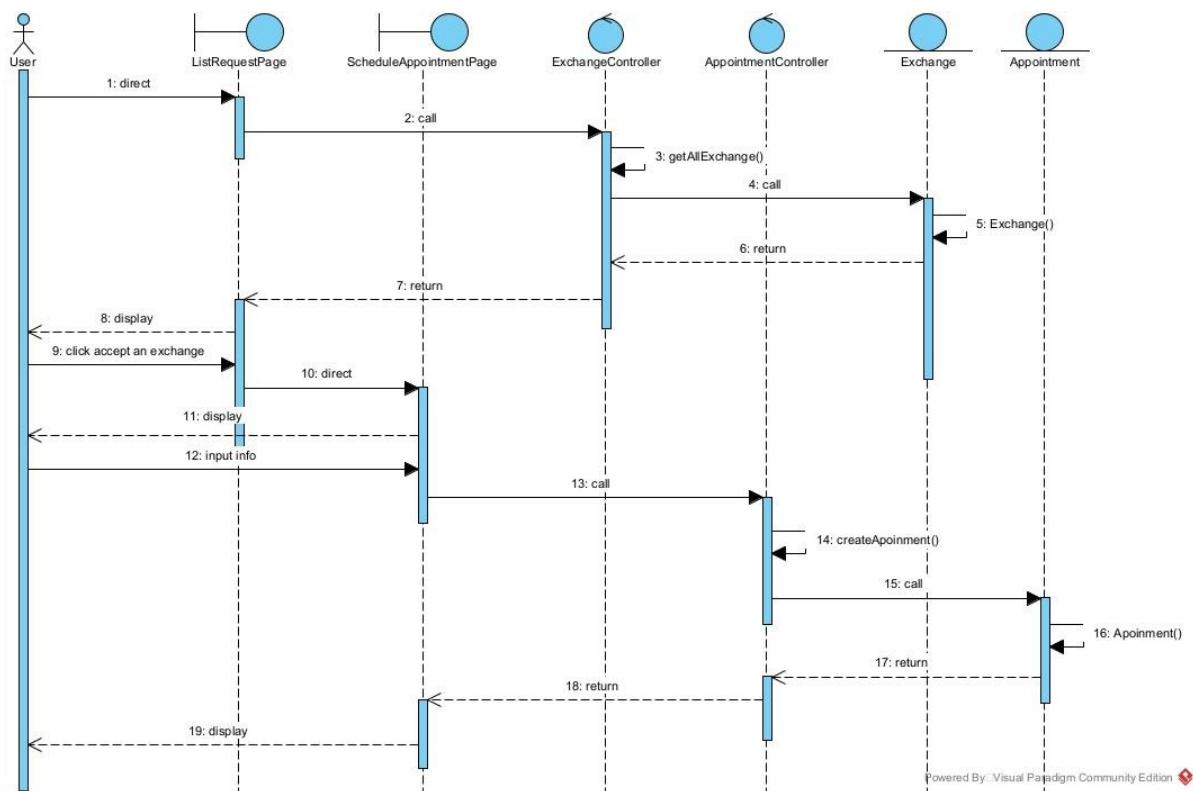
Hình 3.25: Biểu đồ tuần tự chức năng Chat

3.4.5. Tạo yêu cầu trao đổi



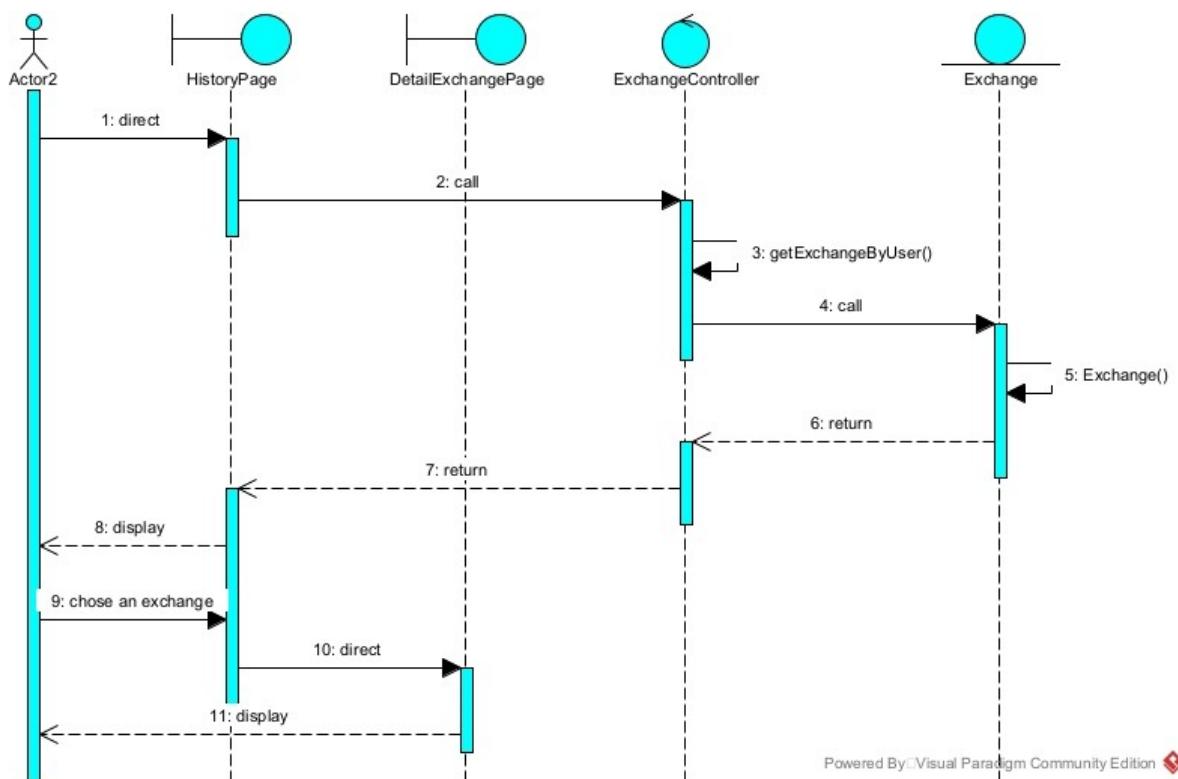
Hình 3.26: Biểu đồ tuần tự chức năng Yêu cầu trao đổi

3.4.6. Xử lý yêu cầu trao đổi - Xác nhận



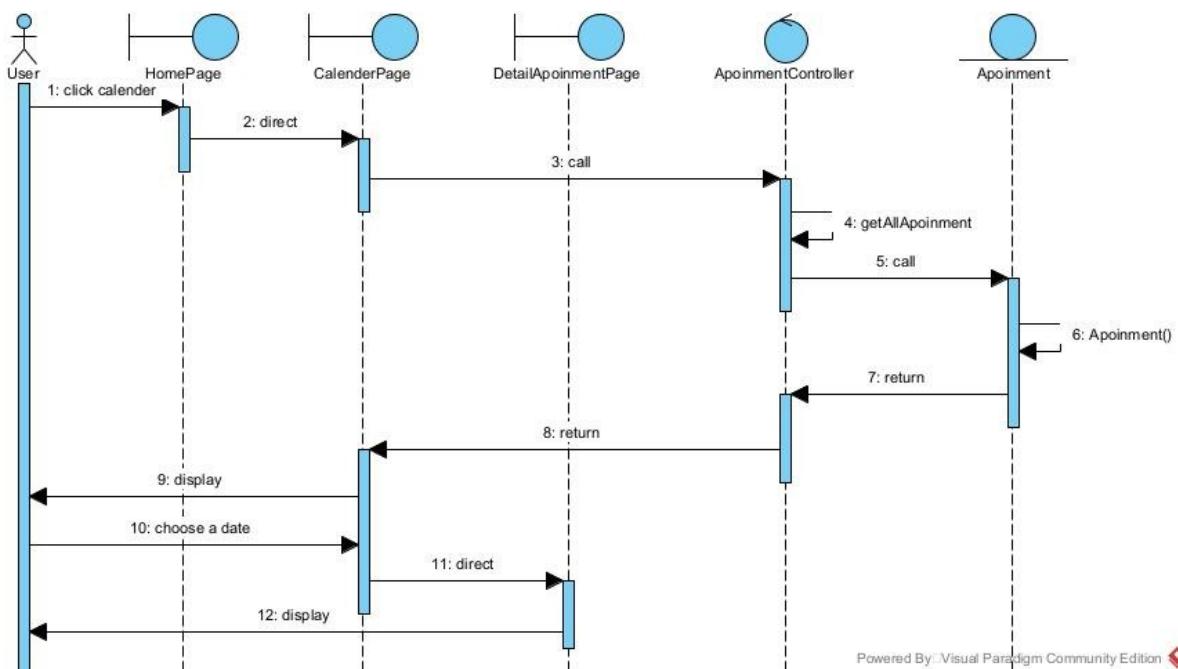
Hình 3.27: Biểu đồ tuần tự chức năng Xử lý yêu cầu trao đổi

3.4.7. Xem lịch sử trao đổi



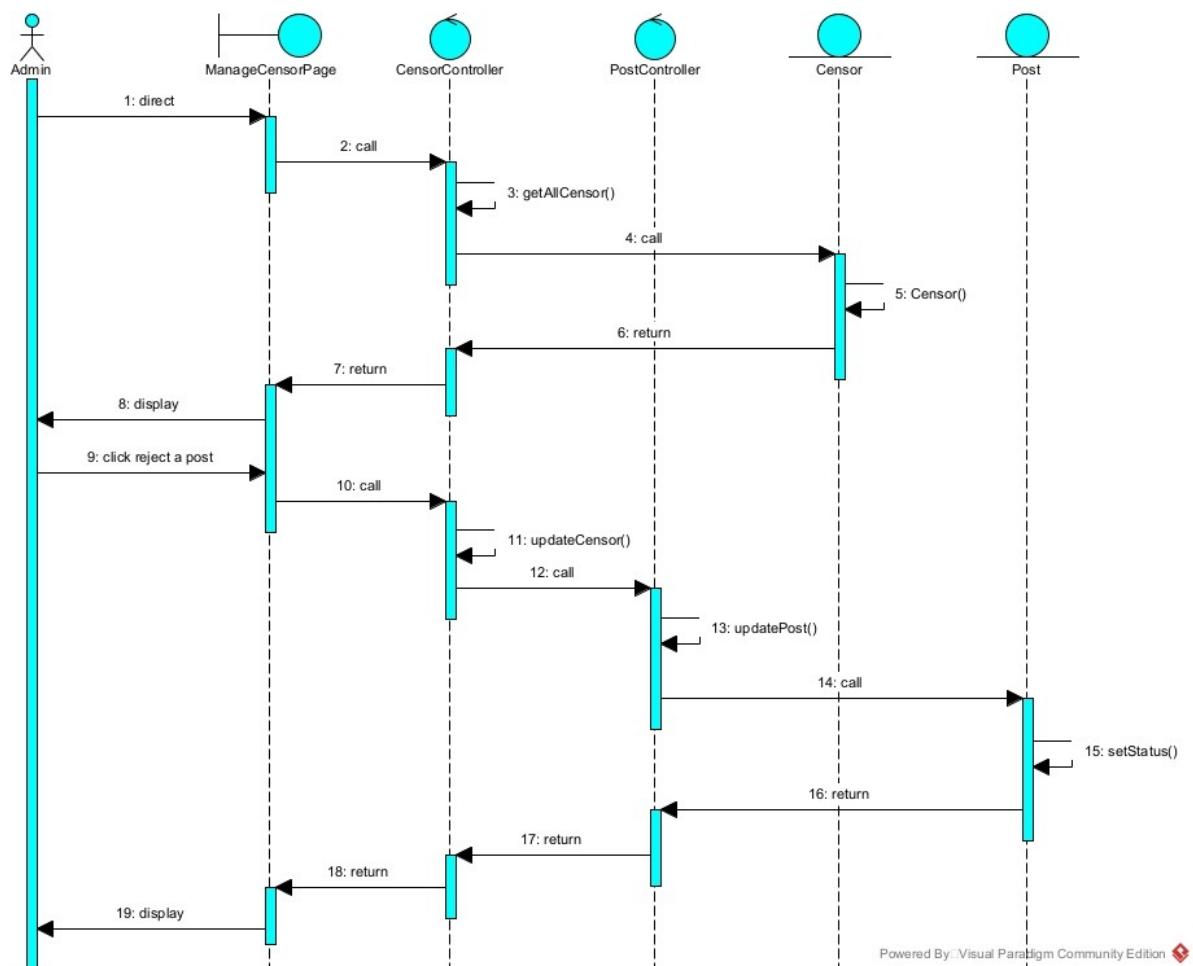
Hình 3.28: Biểu đồ tuần tự Xem lịch sử trao đổi

3.4.8. Xem lịch hẹn



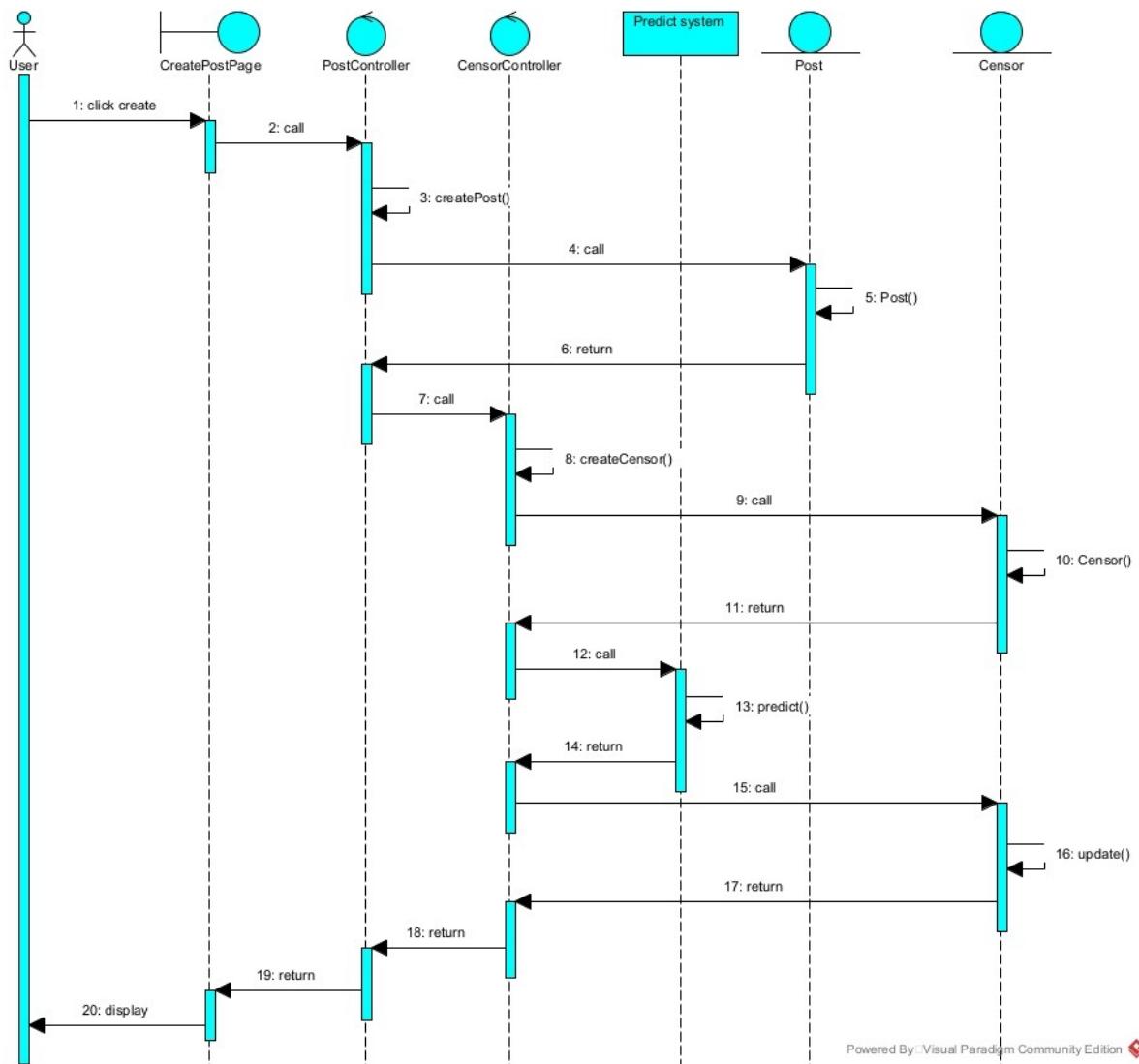
Hình 3.29: Biểu đồ tuần tự Xem lịch hẹn

3.4.9. Kiểm duyệt bài thủ công



Hình 3.30: Biểu đồ tuần tự chức năng kiểm duyệt bài thủ công

3.4.10. Kiểm duyệt bài tự động



Hình 3.31: Biểu đồ tuần tự kiểm duyệt bài đăng tự động

3.5. Thiết kế cơ sở dữ liệu

3.5.1. Các thông tin cần quản lý

Dựa vào biểu đồ lớp thực thể mục 3.3 ta cần quan tâm đến các thông tin sau:

- *Thông tin về người dùng*: tài khoản người dùng, họ và tên, số điện thoại, email, địa chỉ, avatar.
- *Thông tin về bài đăng*: tiêu đề bài đăng, avatar, danh sách ảnh, thẻ loại sản phẩm, thẻ loại bài đăng, mô tả, yêu cầu, địa chỉ có thể trao đổi.
- *Thông tin về thẻ loại sản phẩm của bài đăng và thẻ loại của bài đăng*: id, tên.

- *Thông tin về loại tài khoản:* id, tên.
- *Thông tin về tin nhắn:* người gửi, người nhận, ngày tạo, nội dung, trạng thái
- *Thông tin về báo cáo:* người gửi, nội dung, ngày tạo
- *Thông tin về ảnh:* thuộc bài đăng nào, url là gì
- *Thông tin về lượt trao đổi:* bài đăng đổi, bài đăng đổi phương, nội dung, ngày tạo, trạng thái
- *Thông tin về lượt kiểm duyệt:* thuộc bài đăng nào, do ai đánh giá, trạng thái là gì, nội dung kiểm duyệt, ngày tạo, ngày sửa, kiểu kiểm duyệt.
- *Thông tin về cuộc hẹn:* người tạo, người được mời, ngày hẹn, địa điểm, thời gian.

3.5.2. Đề xuất các lớp thực thể thành các bảng trong cơ sở dữ liệu

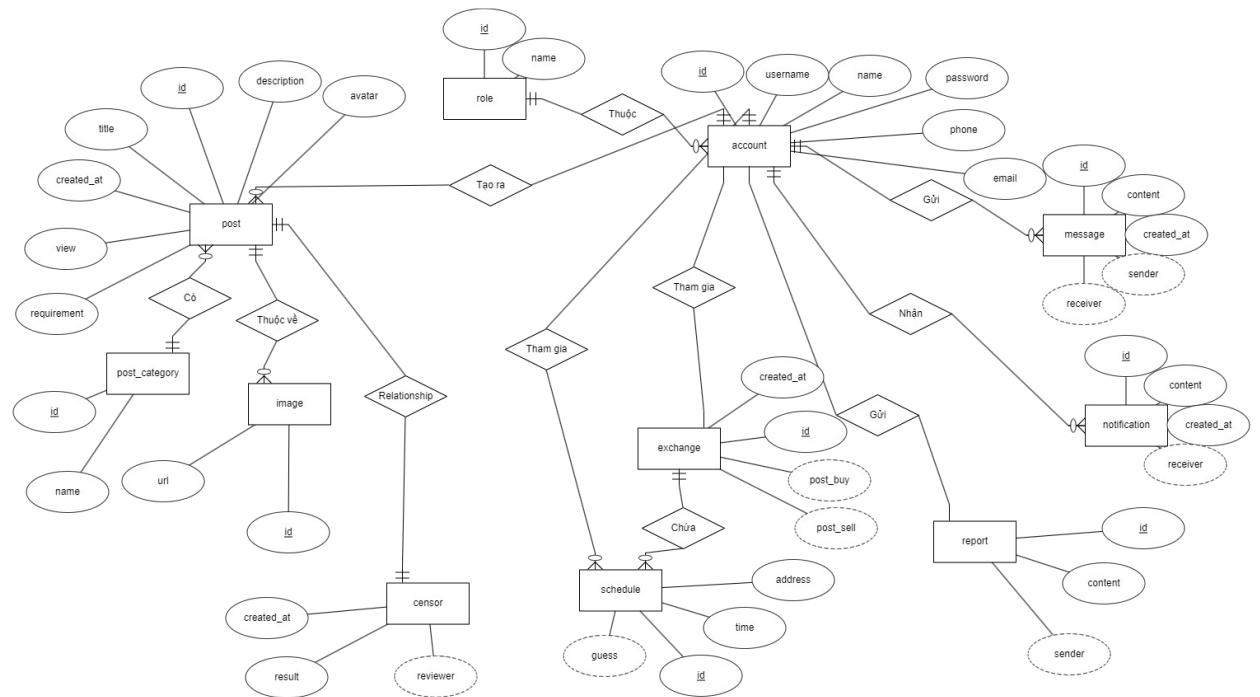
- Lớp Account: Đề xuất bảng account
- Lớp Role: Đề xuất bảng role
- Lớp Member: Nhận thấy Account là thông tin cho lớp Member -> Đề xuất gộp thông tin của Member vào Account
- Lớp Admin và User: Đề xuất phân loại theo bảng role với 2 role là “ROLE_ADMIN” và “ROLE_USER”
- Lớp Notification: Đề xuất bảng notification
- Lớp Message: Đề xuất bảng message
- Lớp Report: Đề xuất bảng report
- Lớp ProductCategory: Đề xuất bảng category_product
- Lớp PostCategory: Đề xuất bảng category_post
- Lớp Post và IOS: Về cơ bản hai lớp này là hai loại bài viết và có các thuộc tính tương tự -> Đề xuất 1 bảng chung cho cả hai là bảng post
- Lớp Product: Vì Post chứa thông tin của Product -> Đề xuất Product trở thành thông tin trong bảng post
- Lớp Image: Đề xuất bảng image
- Lớp Exchange: Đề xuất bảng exchange
- Lớp PostPin: Đề xuất bảng post_pin

- Lớp Appointment: Đề xuất bảng appointment
- Lớp Address: Đề xuất trở thành thuộc tính của bảng appointment_schedule
- Lớp Censor: đề xuất bảng censor
- Lớp Manual và Auto: Đề xuất trở thành thuộc tính của bảng censor

3.5.3. Quan hệ giữa các bảng

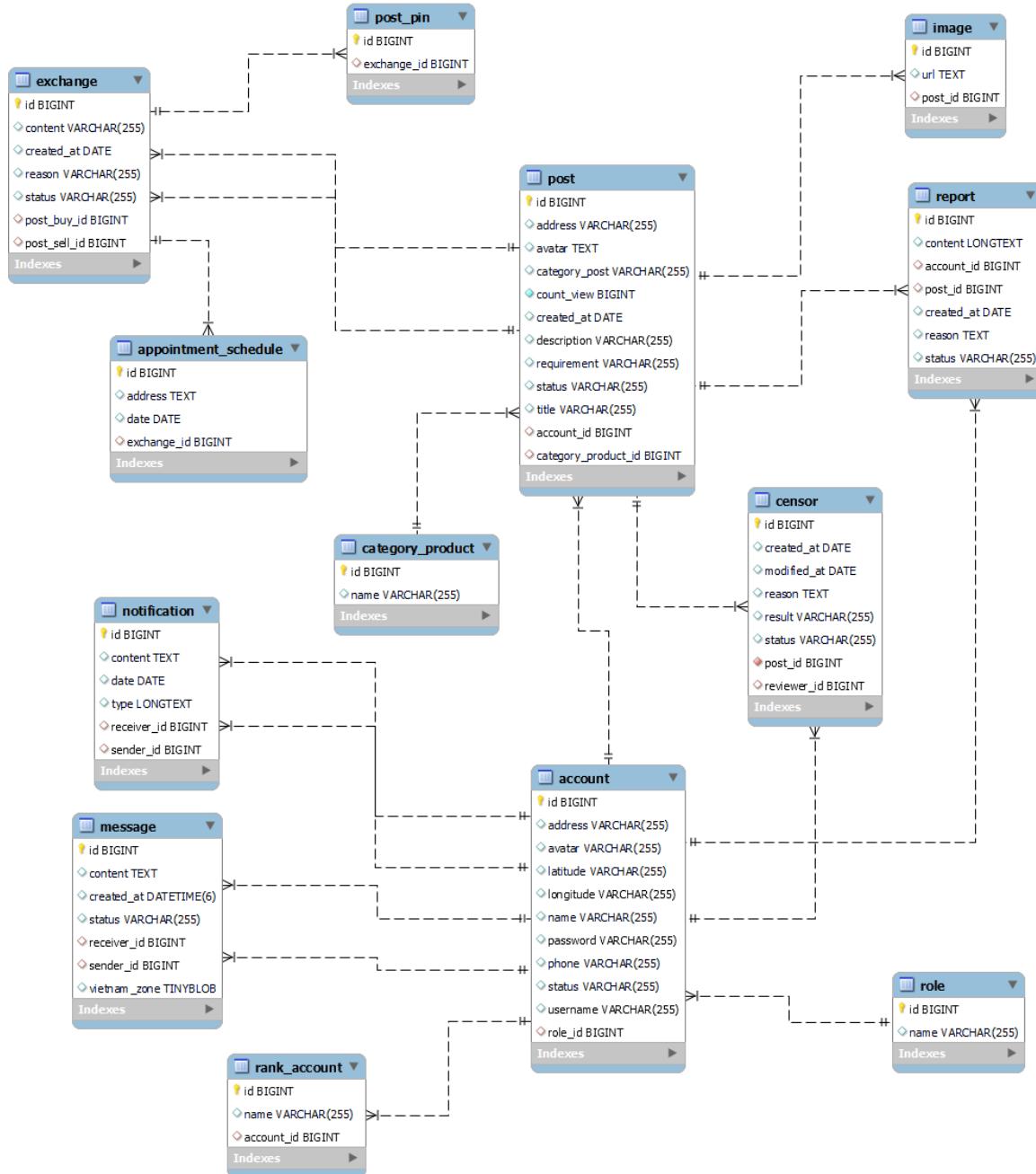
- account - message -> 1 - n
- account - notification -> 1 - n
- account - report -> 1 - n
- account - post -> 1 - n
- post - report -> 1 - n
- post - image -> 1 - n
- post - exchange -> 1 - n
- category_product - post -> 1 - n
- exchange - appointment_schedule -> 1 - n
- exchange - post_pin -> 1 - n

Ta có Mô hình quan hệ thực thể như sau:



Hình 3.32: Mô hình quan hệ thực thể

Vậy từ những phân tích ở trên ta có cơ sở dữ liệu như ảnh sau:



Hình 3.33: Cơ sở dữ liệu cho trang web

3.6. Chức năng kiểm duyệt tự động

3.6.1. Dữ liệu

a. Định dạng dữ liệu

Trong bài toán nhận diện quần áo với yolo, tập dữ liệu sẽ là các ảnh với nhãn là file .txt cùng tên với ảnh, có định dạng [c, x, y, w, h]. Trong đó:

- c: nhãn class ở dạng số (1, 2, 3, 4, ...)
- x, y: tọa độ tâm vật thể, được chuẩn hóa về từ 0-1
- w, h: độ rộng và độ cao của vật thể được chuẩn hóa về từ 0-1

Để dễ hình dung, ảnh và nhãn tương ứng được biểu diễn trong hình sau với nhãn tương ứng là (4 0.51 0.55 0.84 0.90) - 4 là class “shirt”



Hình 3.34: Nhãn của một hình ảnh

b. Gán nhãn dữ liệu

Để có thể thuận tiện gán nhãn cho ảnh ta cần công cụ LabelImg. Công cụ này hỗ trợ gán nhãn theo nhiều định dạng của các mô hình khác nhau.

Tổng cộng tập dữ liệu đầy đủ sau cùng bao gồm:

- 8 class bao gồm: 'Tshirt', 'dress', 'jacket', 'pants', 'shirt', 'short', 'skirt', 'sweater'.

- Tập dữ liệu cho training: 1404 ảnh
- Tập dữ liệu cho validation: 254 ảnh
- Tập dữ liệu cho test: 143 ảnh

Tất cả các thông tin được mô tả trong file data.yaml như ảnh sau:

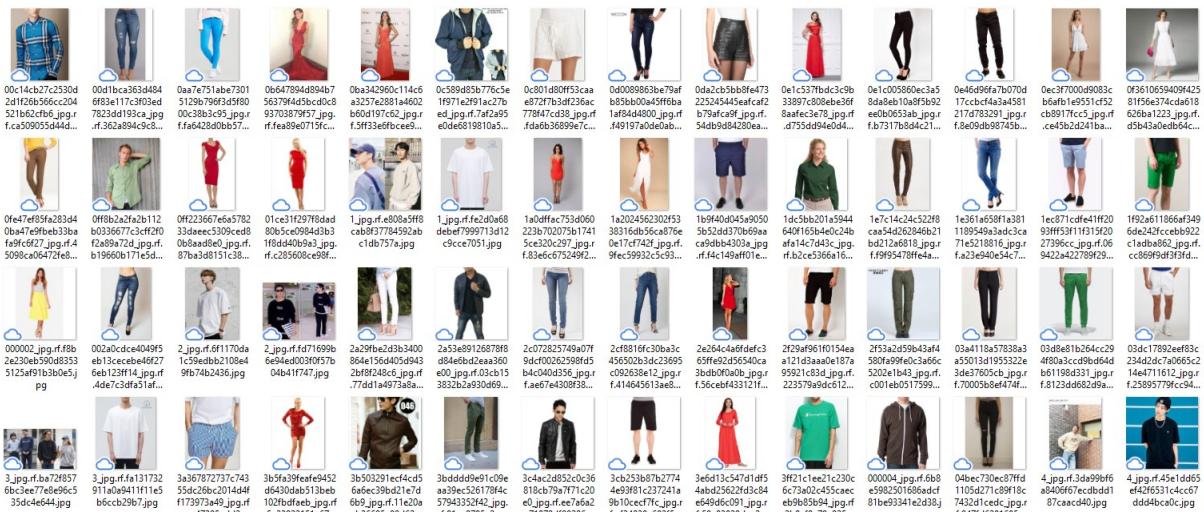
```

1 train: ../train
2 val: ../valid
3 test: ../test
4
5 nc: 8
6 names: ['Tshirt', 'dress', 'jacket', 'pants', 'shirt', 'short', 'skirt', 'sweater']

```

Trong đó trong mỗi folder của từng tập dữ liệu, sẽ chia làm folder chứa ảnh và folder chứa nhãn.

Sau đây là một số hình ảnh trong tập dữ liệu của chúng ta



Hình 3.35: Hình ảnh dùng làm tập dữ liệu

3.6.2. Huấn luyện mô hình

Tất cả các thao tác sau sẽ thực hiện trên google colab với thời gian chạy là T4

Trước tiên chúng ta cần cài đặt thư viện ultralytics để có thể sử dụng được mô hình, sau đó chúng ta sẽ tải pretrain model và được file yolov8n.pt.

```
!pip install ultralytics
```

```
!wget https://github.com/ultralytics/assets/releases/download/v0.0.0/yolov8n.pt
```

Khi đã có pretrain model, chúng ta sẽ tiến hành train model trên tập dữ liệu chúng ta đã chuẩn bị với task (nhiệm vụ) là detect, mode là train, model là model yolov8n.pt như vừa tải ở trên, định nghĩa data theo file data.yaml, với 50 epochs, size ảnh là 640 pixels và batch là 32.

```
# Train with cli
```

```
!yolo task=detect mode=train model=yolov8n.pt data="/content/drive/MyDrive/Data/clothing/data.yaml"
epochs=50 imgsz=640 batch=32
```

Có thể thấy các chỉ số được YOLOv8 dùng để đánh giá (theo hình bên dưới) lần lượt là: box_loss (hàm mất mát cho nhiệm vụ dự đoán tọa độ bounding box), cls_loss (hàm mất mát cho nhiệm vụ dự đoán class), P (Precision), R (Recall), mAP50 và mAP50-95. YOLOv8 sẽ lưu lại 2 lựa chọn với model của lần train sau epoch cuối cùng và lần tốt nhất (tốt nhất tính theo mAP50). Trong đó mAP là thể hiện đơn giản nhất cho việc model có học tốt hay không, ở epoch đầu tiên có mAP50 là 0.256 và giá trị đó là 0.842 ở epoch cuối cùng, điều ấy thể hiện model đã thực hiện tốt việc đào tạo.

Hình biểu diễn quá trình train cho 5 epoch đầu:

Epoch	GPU_mem	box_loss	cls_loss	dfl_loss	Instances	Size
1/50	4.65G	1.007	3.153	1.407	99	640: 100% 44/44 [00:47<00:00, 1.09s/it]
	Class	Images	Instances	Box(P	R	mAP50 mAP50-95): 100% 4/4 [00:06<00:00, 1.53s/it]
	all	254	447	0.569	0.178	0.256 0.186
Epoch	GPU_mem	box_loss	cls_loss	dfl_loss	Instances	Size
2/50	4.31G	0.9361	2.107	1.318	96	640: 100% 44/44 [00:42<00:00, 1.03s/it]
	Class	Images	Instances	Box(P	R	mAP50 mAP50-95): 100% 4/4 [00:06<00:00, 1.75s/it]
	all	254	447	0.542	0.347	0.412 0.289
Epoch	GPU_mem	box_loss	cls_loss	dfl_loss	Instances	Size
3/50	4.29G	0.9393	1.84	1.301	84	640: 100% 44/44 [00:42<00:00, 1.04s/it]
	Class	Images	Instances	Box(P	R	mAP50 mAP50-95): 100% 4/4 [00:03<00:00, 1.04s/it]
	all	254	447	0.499	0.513	0.425 0.292
Epoch	GPU_mem	box_loss	cls_loss	dfl_loss	Instances	Size
4/50	4.32G	0.9412	1.692	1.297	78	640: 100% 44/44 [00:43<00:00, 1.02s/it]
	Class	Images	Instances	Box(P	R	mAP50 mAP50-95): 100% 4/4 [00:04<00:00, 1.01s/it]
	all	254	447	0.534	0.548	0.523 0.369
Epoch	GPU_mem	box_loss	cls_loss	dfl_loss	Instances	Size
5/50	4.32G	0.9249	1.525	1.276	100	640: 100% 44/44 [00:43<00:00, 1.00s/it]
	Class	Images	Instances	Box(P	R	mAP50 mAP50-95): 100% 4/4 [00:05<00:00, 1.27s/it]
	all	254	447	0.552	0.535	0.509 0.357

Hình 3.36: Quá trình train cho 5 epoch đầu

10 epoch cuối sẽ dùng áp dụng mosaic

Epoch	GPU_mem	box_loss	cls_loss	dfl_loss	Instances	Size
41/50	4.29G	0.5278	0.4467	1.019	45	640: 100% 44/44 [00:47<00:00, 1.08s/it]
	Class	Images	Instances	Box(P	R	mAP50 mAP50-95): 100% 4/4 [00:04<00:00, 1.09s/it]
	all	254	447	0.748	0.762	0.814 0.628
Epoch	GPU_mem	box_loss	cls_loss	dfl_loss	Instances	Size
42/50	4.29G	0.5229	0.4124	1.025	44	640: 100% 44/44 [00:40<00:00, 1.10s/it]
	Class	Images	Instances	Box(P	R	mAP50 mAP50-95): 100% 4/4 [00:04<00:00, 1.03s/it]
	all	254	447	0.728	0.816	0.832 0.645
Epoch	GPU_mem	box_loss	cls_loss	dfl_loss	Instances	Size
43/50	4.28G	0.5133	0.3919	1.019	51	640: 100% 44/44 [00:42<00:00, 1.03s/it]
	Class	Images	Instances	Box(P	R	mAP50 mAP50-95): 100% 4/4 [00:03<00:00, 1.14s/it]
	all	254	447	0.798	0.76	0.811 0.637
Epoch	GPU_mem	box_loss	cls_loss	dfl_loss	Instances	Size
44/50	4.29G	0.5089	0.3778	1.012	41	640: 100% 44/44 [00:40<00:00, 1.09s/it]
	Class	Images	Instances	Box(P	R	mAP50 mAP50-95): 100% 4/4 [00:04<00:00, 1.17s/it]
	all	254	447	0.757	0.801	0.819 0.644

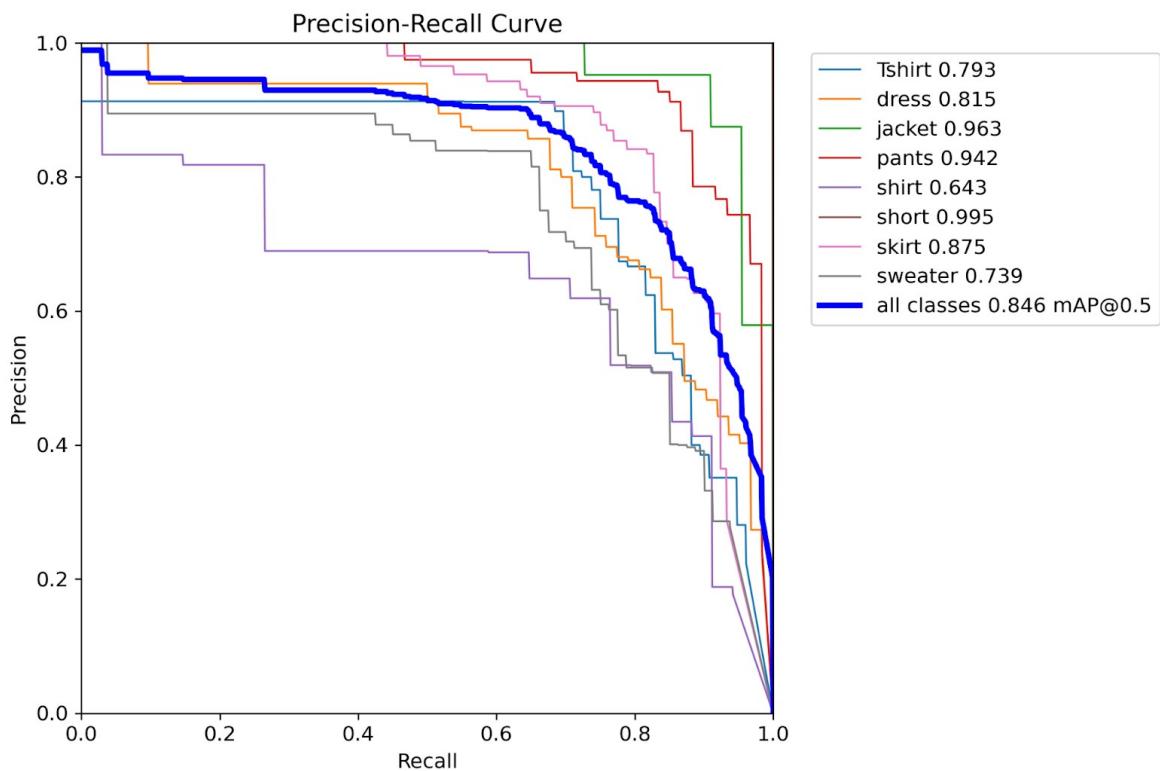
Hình 3.37: Quá trình train cho các epoch cuối

Tổng thời gian train xấp xỉ 41 phút, không tính thời gian load dữ liệu.

3.6.3. Đánh giá mô hình

Một số biểu đồ mà YOLOv8 đã rút ra trong quá trình đào tạo giúp chúng ta có thể “nhìn” được quá trình học của model

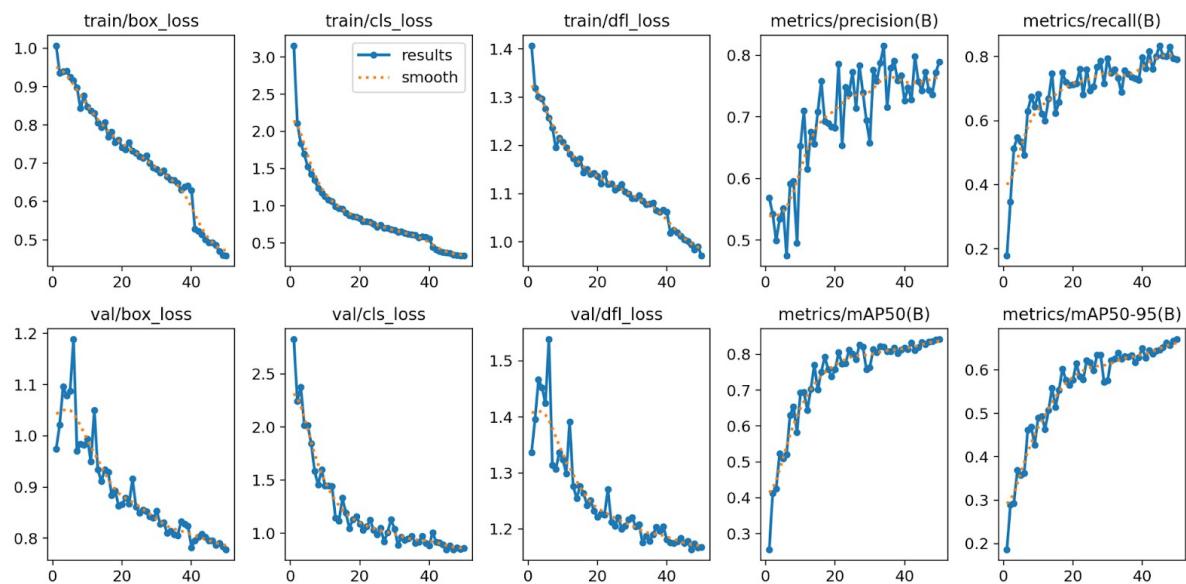
Precision-Recall



Hình 3.38: Đường cong Precision-Recall

Nhận thấy có vẻ việc dự đoán “shirt” của model yếu hơn so với các class khác, có thể là do dữ liệu về class đó ít hơn các class còn lại. Tuy nhiên về tổng thể thì mAP vẫn ở một mức chấp nhận được.

Các hàm loss



Hình 3.39: Biểu đồ biến động của các hàm loss

Nhìn vào các biểu đồ của hàm mất mát, ta thấy model đào tạo khá nhanh trong 10-20 epoch đầu, từ epoch 20-40 có chậm hơn một chút nhưng vẫn là một tỷ lệ học tốt và đều, ở 10 epoch cuối (từ 40-50) có lẽ việc dừng mosaic khiến các hàm loss giảm đột ngột. Quá trình train trên tập train có vẻ khá thuận lợi, tuy nhiên trên tập validation lại có khá nhiều biến động ở việc dự đoán bounding box.

Chúng ta sẽ thử dự đoán một vài tấm ảnh và xem kết quả của model

```

from ultralytics import YOLO      Khai báo YOLO từ ultralytics để đọc model

# Load model
modelSrc = "clothing-detect.pt"  Load model mà chúng
model = YOLO(modelSrc)          ta đã train được

# list images
listImage = ['https://static-images.vnncdn.net/files/publish/2022/9/3/bien-vo-cuc-thai-binh-346.jpeg',
             'https://image.bnews.vn/MediaUpload/Org/2021/04/16/94881840-3512380665444994-5140652141903347712-n.jpg']  Danh sách ảnh (có thể
                                                                                                     là url hoặc đường dẫn
                                                                                                     trực tiếp trong máy)

detection_outputs = model.predict(source=listImage, conf=0.25, save=True)  Thực hiện dự đoán

```

Ảnh đầu vào	Độ tự tin chấp nhận	Option có lưu ảnh kết quả không
	phát hiện	

Hình 3.40: Thủ nghiệm model

Kết quả khi chạy như sau:

```
[Running] python -u "d:\Code\Python\yolov8-safe-envir\code\yolo_clothing.py"

0: 640x640 1 dress, 1: 640x640 1 Tshirt, 253.0ms
Speed: 4.5ms preprocess, 126.5ms inference, 1.0ms postprocess per image at shape (1, 3, 640, 640)
Results saved to esc[1mruns\detect\predict11esc[0m

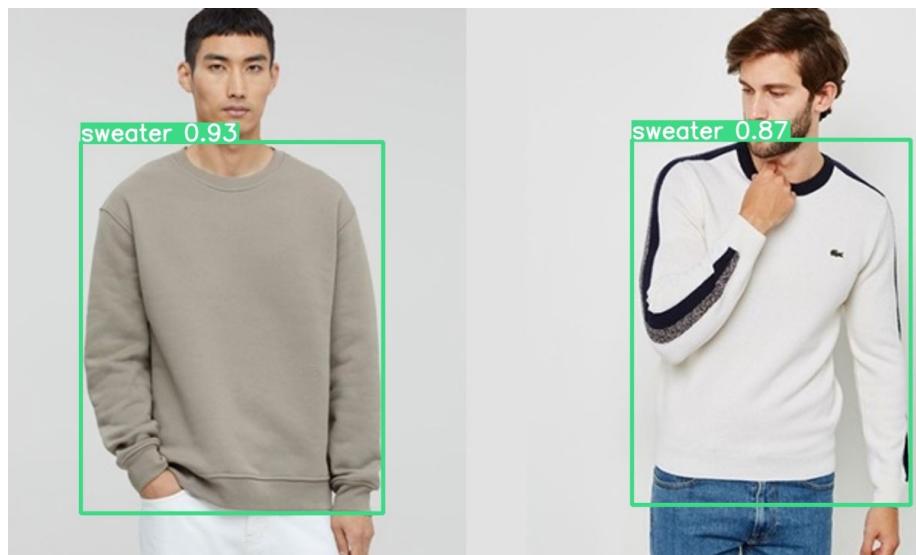
[Done] exited with code=0 in 6.377 seconds
```

Hình 3.41: Kết quả thử nghiệm model

Chúng ta hết khoảng 6 giây cho cả việc tải ảnh (vì là url ảnh) và việc dự đoán. Các dự đoán khi được biểu thị trên ảnh sẽ như các hình sau. Có thể thấy model đang thực hiện khá tốt việc dự đoán với các lớp có mAP cao.



Hình 3.42: Mô hình dự đoán nhãn “pants”



Hình 3.43: Mô hình dự đoán nhãn “sweater”

Chúng ta cũng sẽ thử với hình ảnh không liên quan đến quần áo. Ví dụ như trong ảnh dưới, model có vẻ không nhận nhầm.



Hình 3.44: Mô hình không có dấu hiệu bắt nhầm vật thể

3.7. Kết luận chương

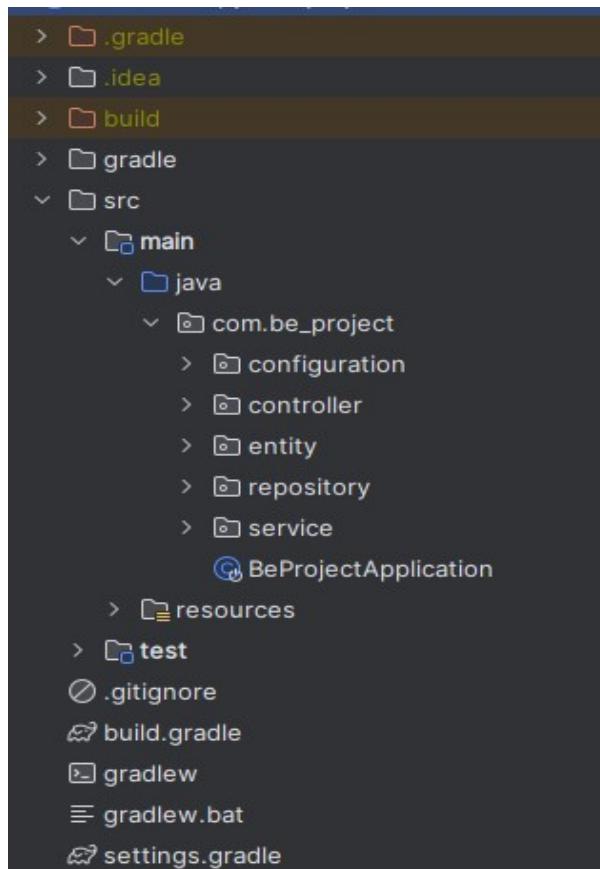
Trong chương này chúng ta đã hoàn thành thiết kế hệ thống của trang web và đã hoàn thành việc tạo ra model dự đoán bằng YOLO. Chương tiếp theo của đồ án sẽ trình bày quá trình cài đặt và sử dụng trang web.

CHƯƠNG 4: TRIỂN KHAI VÀ THỬ NGHIỆM

Chương 4 của đồ án sẽ thực hiện việc triển khai và thử nghiệm sử dụng trang web.

4.1. Cài đặt ứng dụng Spring Boot

Cấu trúc thư mục:



Hình 4.45: Cấu trúc ứng dụng Spring Boot

Khởi chạy ứng dụng Spring Boot

```

Task :bootRun

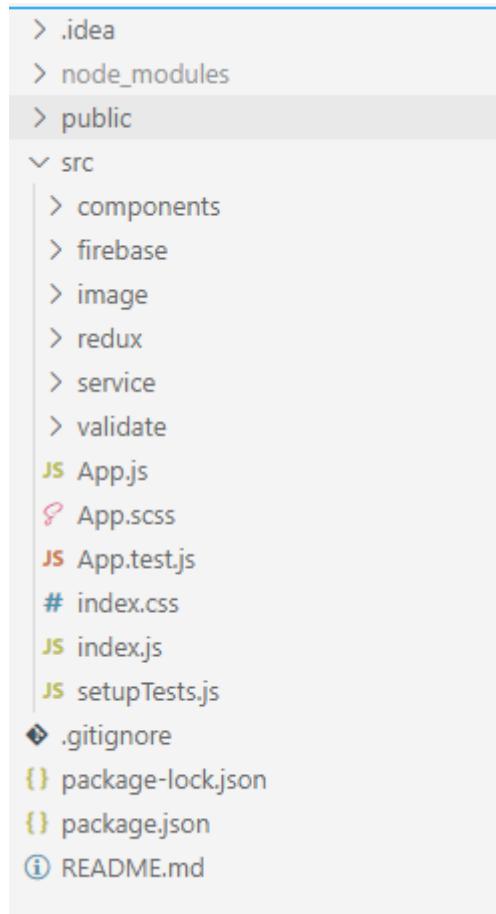
[...]
:: Spring Boot ::      (v2.7.17)

2023-12-23 02:05:33.604  INFO 22640 --- [  main] com.be_project.BeProjectApplication      : Starting BeProjectApplication using Java 17.0.4.1 on DESKTOP-4025V18 with PID 22640 (D:\DoAn\DoAn-BE-Copy\build\classes\java\main)
n started by HungNguyen in D:\DoAn\DoAn-BE-Copy
2023-12-23 02:05:33.607  INFO 22640 --- [  main] com.be_project.BeProjectApplication      : No active profile set, falling back to 1 default profile: "default"
2023-12-23 02:05:34.377  INFO 22640 --- [  main] .s.d.r.c.RepositoryConfigurationDelegate : Bootstrapping Spring Data JPA repositories in DEFAULT mode.
2023-12-23 02:05:37.890  INFO 22640 --- [  main] o.s.w.DefaultSecurityFilterChain        : Will secure any request with [org.springframework.security.web.session.DisableEncodeUrlFilter@6bba9e1, org.springframework.security.web.context.request.async.WebAsyncManagerIntegrationFilter@4c51fc9a, org.springframework.security.web.context.SecurityContextPersistenceFilter@1d8ac75, org.springframework.security.web.header.HeaderWriterFilter@5de1422, g.springframework.web.filter.CorsFilter@2b370cc, org.springframework.security.web.authentication.logout.LogoutFilter@139843a, com.be_project.configuration.filter.JwtAuthenticationFilter@366d312, org.springframework.security.web.savedrequest.RequestCacheAwareFilter@194ef9, org.springframework.security.web.context.request.ServletWebRequestFilter@355a049, org.springframework.security.web.authentication.UsernamePasswordAuthenticationFilter@287050, o.org.springframework.security.web.session.SessionManagementFilter@8ead73b, org.springframework.security.web.access.ExceptionTranslationFilter@701da511, org.springframework.security.web.access.intercept.FilterSecurityInterceptor@694450e]
2023-12-23 02:05:38.270  INFO 22640 --- [  main] o.s.b.w.embedded.tomcat.TomcatWebServer : Tomcat started on port(s): 8080 (http) with context path ''
2023-12-23 02:05:38.272  INFO 22640 --- [  main] o.s.m.s.b.SimpleBrokerMessageHandler   : Starting ...
2023-12-23 02:05:38.272  INFO 22640 --- [  main] o.s.m.s.b.SimpleBrokerMessageHandler   : BrokerAvailabilityEvent[available=true, SimpleBrokerMessageHandler [org.springframework.messaging.simp.broker.DefaultSubscriptionRegistry@3045a3e0]]
2023-12-23 02:05:38.278  INFO 22640 --- [  main] o.s.m.s.b.SimpleBrokerMessageHandler   : Started.
2023-12-23 02:05:38.278  INFO 22640 --- [  main] com.be_project.BeProjectApplication     : Started BeProjectApplication in 5.047 seconds (JVM running for 5.428)
[...]
:bootRun
-----> 80% EXECUTING (22s)
```

Hình 4.46: Khởi chạy ứng dụng Spring Boot

4.2. Cài đặt ứng dụng ReactJS

Cấu trúc thư mục:



Hình 4.47: Cấu trúc ứng dụng React

Khởi chạy ứng dụng:

Compiled successfully!

You can now view case-fe in the browser.

Local: http://localhost:3000
On Your Network: http://172.17.240.1:3000

Note that the development build is not optimized.
To create a production build, use `npm run build`.

webpack compiled **successfully**

Hình 4.48: Khởi chạy ứng dụng React

4.3. Cài đặt ứng dụng Flask

Khởi chạy ứng dụng

```
[Running] python -u "d:\Code\Python\yolov8-safe-envir\code\my_app\app.py"
* Serving Flask app 'app'
* Debug mode: on
WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead.
* Running on http://127.0.0.1:5000
Press CTRL+C to quit
* Restarting with stat
* Debugger is active!
* Debugger PIN: 296-549-956
```

Hình 4.49: Trạng thái ứng dụng khi khởi chạy

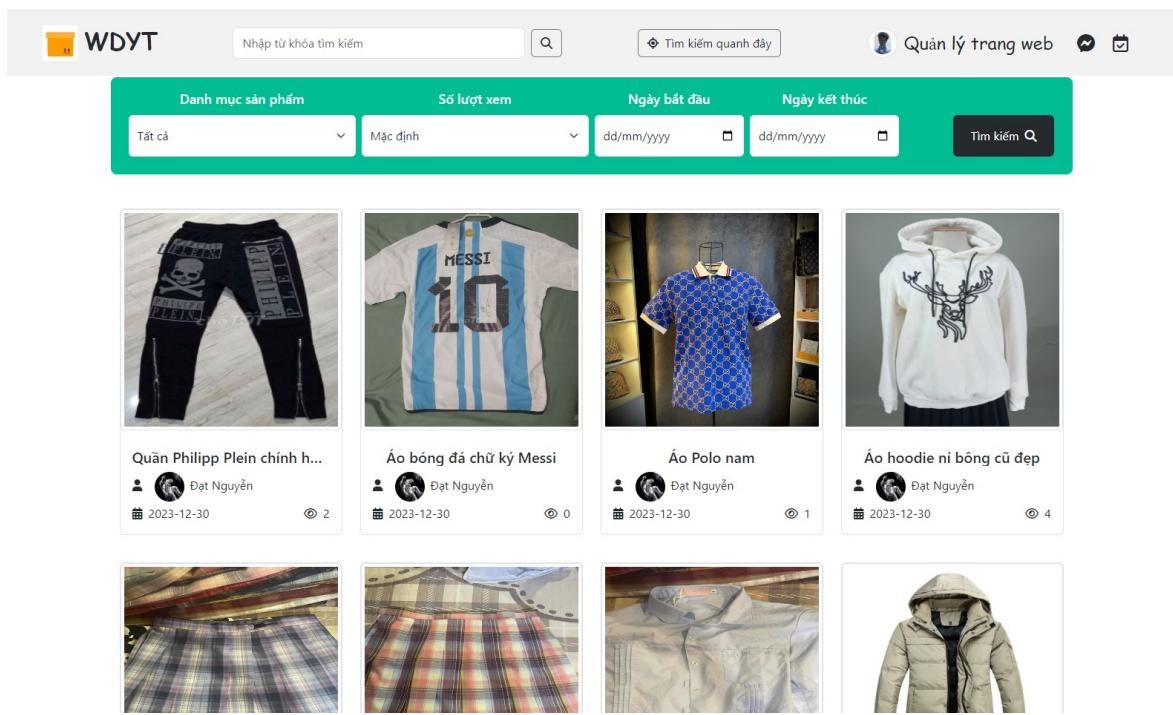
4.4. Các giao diện

a. Phía người dùng

Giao diện đăng nhập:

Hình 4.50: Giao diện Đăng nhập

Giao diện trang chủ:



Hình 4.51: Giao diện Trang chủ

Giao diện thêm bài đăng:

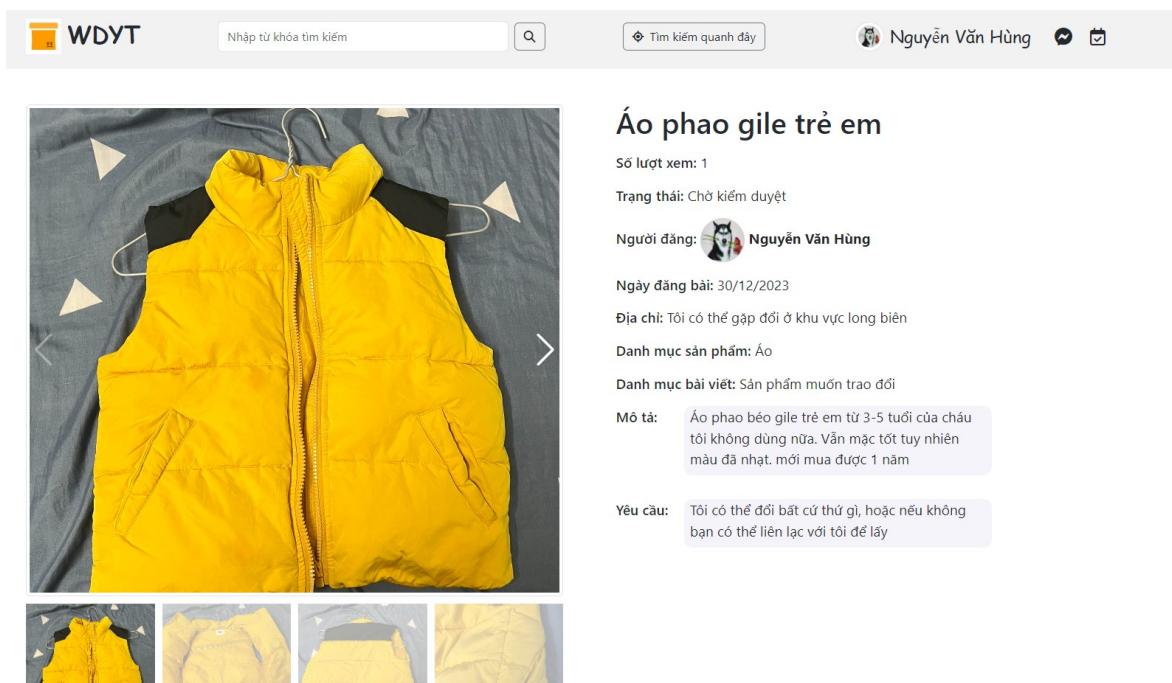
The screenshot shows the "Thêm bài viết mới" (Add new post) page. At the top, there is a search bar with placeholder text "Nhập từ khóa tìm kiếm" and a magnifying glass icon. To its right is a user profile icon and the name "Nguyễn Văn Hùng" with a gear and checkmark icon. Below the header is a form with several fields:

- Tên bài viết ***: A text input field containing "Áo phao gile trẻ em".
- Danh mục sản phẩm ***: A dropdown menu showing "Áo".
- Danh mục bài viết ***: A dropdown menu showing "Sản phẩm muốn trao đổi".
- Địa chỉ ***: A text input field containing "Tôi có thể gặp đối ở khu vực long biên".
- Mô tả sản phẩm ***: A text input field containing "Áo phao béo gile trẻ em từ 3-5 tuổi của cháu tôi không dùng nữa. Vẫn mặc tốt tuy nhiên màu đã nhạt, mới mua được 1 năm".
- Yêu cầu ***: A text input field containing "Tôi có thể đổi bất cứ thứ gì, hoặc nếu không bạn có thể liên lạc với tôi để lấy".
- Ảnh đại diện ***: A file upload field with the placeholder "Chọn tệp" and "Không có tệp nào được chọn".
- Ảnh giới thiệu chi tiết ***: A file upload field with the placeholder "Chọn tệp" and "Không có tệp nào được chọn".

Below the form are four thumbnail images of a yellow puffer vest. At the bottom are two buttons: "Thêm bài viết" (Add post) in blue and "Hủy" (Cancel) in grey.

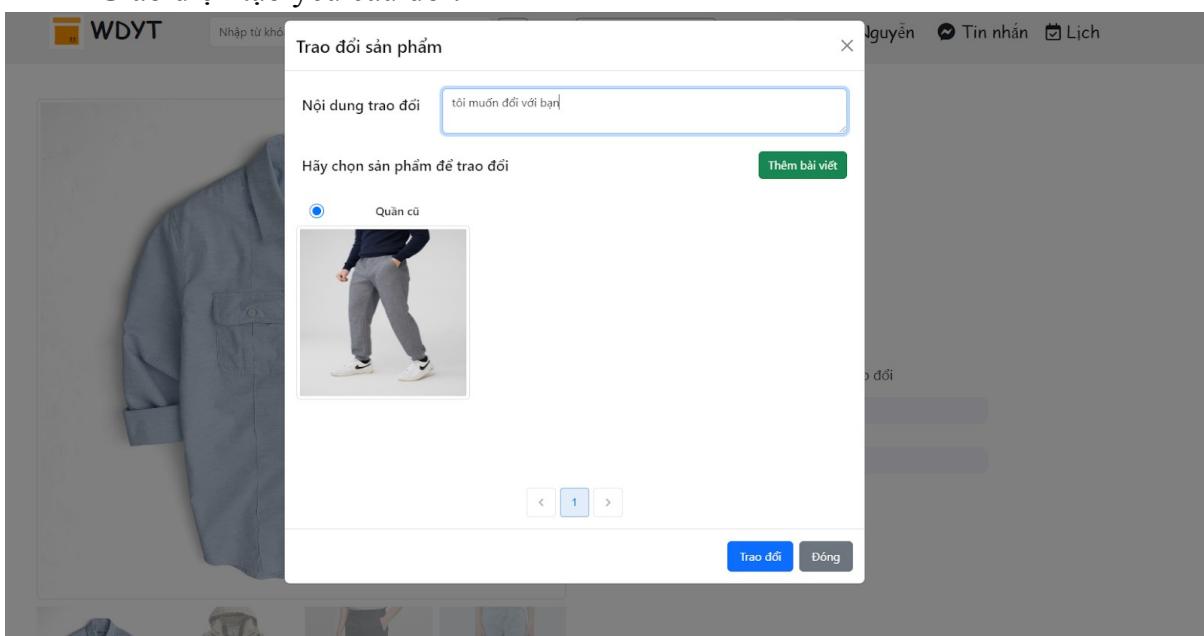
Hình 4.52: Giao diện Thêm bài đăng

Giao diện chi tiết bài đăng:



Hình 4.53: Giao diện Chi tiết bài đăng

Giao diện tạo yêu cầu đổi:



Hình 4.54: Giao diện tạo yêu cầu trao đổi

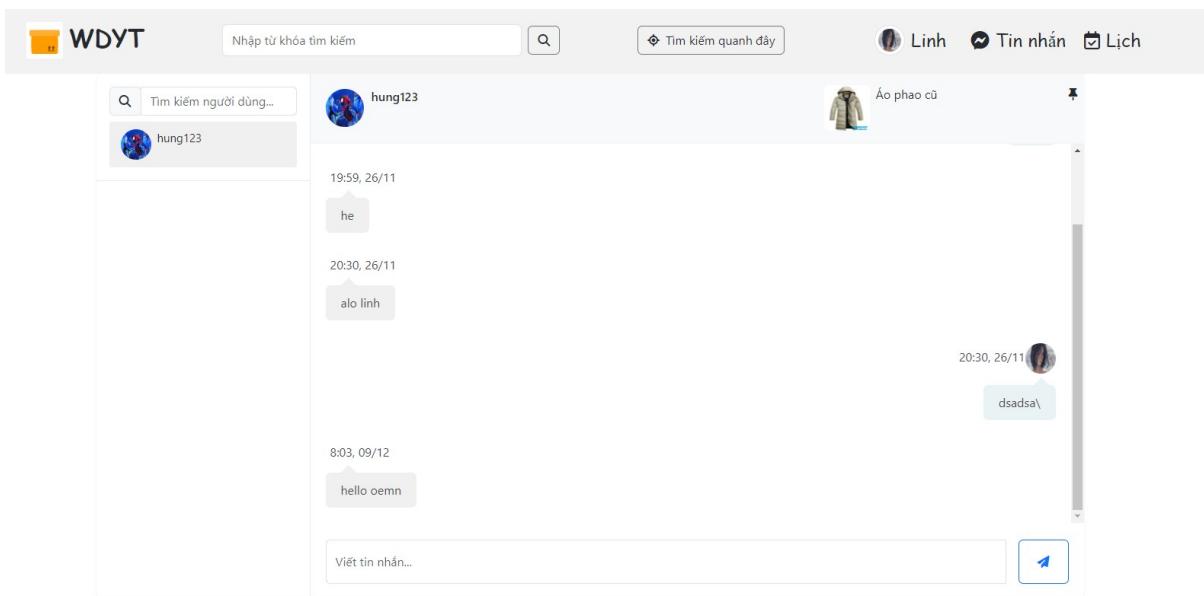
Quản lý lịch sử trao đổi:

Hình 4.55: Quản lý lịch sử trao đổi

Khi xác nhận muốn đổi:

Hình 4.56: Xác nhận trao đổi

Giao diện tin nhắn:



Hình 4.57: Giao diện Chat

b. Phía Quản lý trang web

Giao diện quản lý các bài đăng:

DANH SÁCH BÀI ĐĂNG					
Danh mục bài viết	Danh mục sản phẩm	Trạng thái	Tim kiếm theo người đăng		
Tất cả	Tất cả	Tất cả	Nhập từ khóa tìm kiếm		
STT	Tên bài đăng	Người đăng	Ngày đăng	Trạng thái	Hành động
1	Quần ống xuông clownz	hungacmaa	30/12/2023	Chưa trao đổi	<button>Chi tiết</button> <button>Khóa</button>
2	Quần thể dục	hungacmaa	30/12/2023	Chờ kiểm duyệt	<button>Chi tiết</button> <button>Khóa</button>
3	Áo phao gile trẻ em	hungacmaa	30/12/2023	Chờ kiểm duyệt	<button>Chi tiết</button> <button>Khóa</button>
4	Quần Philipp Plein chính hãng	dat77bg	30/12/2023	Chưa trao đổi	<button>Chi tiết</button> <button>Khóa</button>
5	Áo bóng đá chữ ký Messi	dat77bg	30/12/2023	Chưa trao đổi	<button>Chi tiết</button> <button>Khóa</button>
6	Áo Polo nam	dat77bg	30/12/2023	Chưa trao đổi	<button>Chi tiết</button> <button>Khóa</button>
7	Áo nam Gucci	dat77bg	30/12/2023	Chưa trao đổi	<button>Chi tiết</button> <button>Khóa</button>
8	quần	user1	30/12/2023	Khóa	<button>Chi tiết</button> <button>Khóa</button>
9	Áo hoodie ni bông cũ đẹp	dat77bg	30/12/2023	Chưa trao đổi	<button>Chi tiết</button> <button>Khóa</button>
10	Váy kè ca rô màu tím	trangg	30/12/2023	Chưa trao đổi	<button>Chi tiết</button> <button>Khóa</button>

Hình 4.58: Giao diện quản lý bài đăng của Admin

Giao diện quản lý người dùng:

DANH SÁCH NGƯỜI DÙNG

STT	Tên	Vai trò	Trạng thái	Hành động
1	Quản lý trang web	Admin	Đang hoạt động	<button>Chi tiết</button> <button>Khóa</button>
2	Phương Tuấn	Người dùng	Đang hoạt động	<button>Chi tiết</button> <button>Khóa</button>
3	Predict System	Admin	Đang hoạt động	<button>Chi tiết</button> <button>Khóa</button>
4	Nguyễn Văn Hùng	Người dùng	Đang hoạt động	<button>Chi tiết</button> <button>Khóa</button>
5	Ngô Khắc Phương	Người dùng	Đang hoạt động	<button>Chi tiết</button> <button>Khóa</button>
6	lê hàn trang	Người dùng	Đang hoạt động	<button>Chi tiết</button> <button>Khóa</button>
7	Nguyễn Hữu Huy	Người dùng	Đang hoạt động	<button>Chi tiết</button> <button>Khóa</button>
8	Ngô Khắc Phương	Người dùng	Đang hoạt động	<button>Chi tiết</button> <button>Khóa</button>
9	Đạt Nguyễn	Người dùng	Đang hoạt động	<button>Chi tiết</button> <button>Khóa</button>
10	Huy	Người dùng	Đang hoạt động	<button>Chi tiết</button> <button>Khóa</button>

Hình 4.59: Giao diện quản lý người dùng của Admin

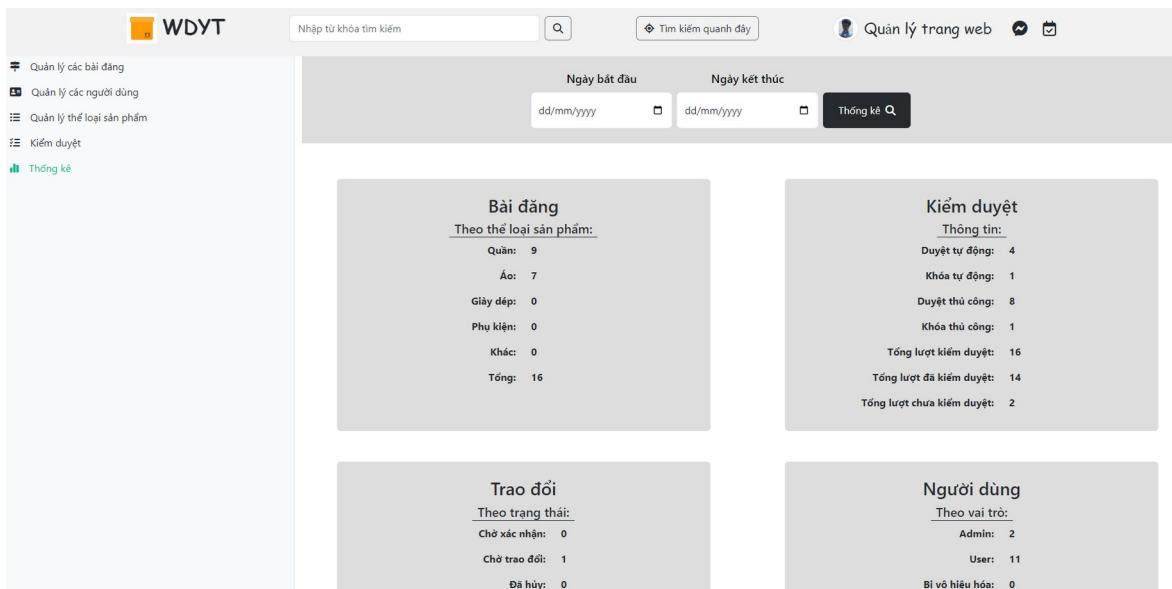
Giao diện kiểm duyệt:

DANH SÁCH KIỂM DUYỆT

STT	Tên bài đăng	Ngày tạo	Ngày kiểm duyệt	Người kiểm duyệt	Trạng thái	Hành động
1	Quần ống xuông clownz	2023-12-30	2023-12-30	Predict System	Đã kiểm duyệt	<button>Duyệt</button> <button>Khóa</button> <button>Chi tiết</button>
2	Quần thể dục	2023-12-30	Chưa duyệt	Chưa duyệt	Chờ kiểm duyệt	<button>Duyệt</button> <button>Khóa</button> <button>Chi tiết</button>
3	Áo phao gile trẻ em	2023-12-30	Chưa duyệt	Chưa duyệt	Chờ kiểm duyệt	<button>Duyệt</button> <button>Khóa</button> <button>Chi tiết</button>
4	Quần Philipp Plein chính hãng	2023-12-30	2023-12-30	Quản lý trang web	Đã kiểm duyệt	<button>Duyệt</button> <button>Khóa</button> <button>Chi tiết</button>
5	Áo bóng đá chữ ký Messi	2023-12-30	2023-12-30	Quản lý trang web	Đã kiểm duyệt	<button>Duyệt</button> <button>Khóa</button> <button>Chi tiết</button>
6	Áo Polo nam	2023-12-30	2023-12-30	Quản lý trang web	Đã kiểm duyệt	<button>Duyệt</button> <button>Khóa</button> <button>Chi tiết</button>
7	Áo nam Gucci	2023-12-30	2023-12-30	Quản lý trang web	Đã kiểm duyệt	<button>Duyệt</button> <button>Khóa</button> <button>Chi tiết</button>
8	quần	2023-12-30	2023-12-30	Quản lý trang web	Đã kiểm duyệt	<button>Duyệt</button> <button>Khóa</button> <button>Chi tiết</button>
9	Áo hoodie ni bóng cũ đẹp	2023-12-30	2023-12-30	Quản lý trang web	Đã kiểm duyệt	<button>Duyệt</button> <button>Khóa</button> <button>Chi tiết</button>
10	Váy kè ca rô màu tím	2023-12-30	2023-12-30	Quản lý trang web	Đã kiểm duyệt	<button>Duyệt</button> <button>Khóa</button> <button>Chi tiết</button>

Hình 4.60: Giao diện kiểm duyệt

Thống kê một số thông tin:



Hình 4.61: Giao diện thống kê

KẾT LUẬN

Dánh giá đồ án

Sau quá trình tìm hiểu, nghiên cứu và xây dựng đề tài “Xây dựng trang web trao đổi mặt hàng thời trang”, đồ án đã đạt được một số kết quả như sau:

➤ Về mặt lý thuyết, đồ án đã đạt được các mục tiêu cơ bản:

- Quy trình xây dựng một trang web dạng đăng tin
- Năm được nền tảng của Spring Boot, React và MySQL
- Năm được nền tảng về bài toán dự đoán đối tượng trong ảnh và cách sử dụng cơ bản của YOLOv8 để xây dựng mô hình dự đoán vật thể trong ảnh

➤ Về mặt thực tiễn

- Đã xây dựng thành công trang web theo yêu cầu đề ra với các chức năng chính
- Xây dựng được quy trình kiểm duyệt bài đăng tự động và thủ công
- Xây dựng được quy trình trao đổi các món đồ
- Đưa trang web vào sử dụng thực tế trong một thời gian ngắn

➤ Hạn chế

- Giao diện trang web chưa dễ sử dụng và chưa có giao diện cho điện thoại.
- Chưa có một hệ thống bảo mật và tối ưu đủ.
- Lượng mặt hàng bị giới hạn trong một số thể loại nhất định
- Hệ thống kiểm duyệt tự động còn chưa nhắm vào các yếu tố phức tạp hơn

Định hướng phát triển

Hệ thống hiện tại vẫn chưa thực sự hoàn thiện và có thể hướng tới việc phát triển như:

- Cải thiện hệ thống tối ưu cho trang web cả về kiến trúc và hạ tầng

- Cải thiện tính bảo mật
- Hướng tới phát triển cả nền tảng di động
- Mở rộng thể loại mặt hàng trao đổi

TÀI LIỆU THAM KHẢO

- [1] N. V. Hoai, "Viblo," Viblo, 17 5 2020. [Online]. Available:
<https://viblo.asia/p/faster-r-cnn-cho-bai-toan-phat-hien-doi-tuong-gDVK2Jz2KLj>.
- [2] VBD, "VINBIGDATA," 16 8 2023. [Online]. Available:
<https://vinbigdata.com/cong-nghe-hinh-anh/yolov8-co-gi-nang-cap-so-voicac-phien-ban-truoc.html>.
- [3] Ultralytics, "Ultralytics," 12 11 2023. [Online]. Available:
<https://docs.ultralytics.com/models/yolov8/#citations-and-acknowledgements>.
- [4] RangeKing, "Github," 10 1 2023. [Online]. Available:
<https://github.com/ultralytics/ultralytics/issues/189>.
- [5] S. R. Schach, Object-Oriented and Classical Software Eighth Edition, Vanderbilt University.
- [6] T. Đ. Quế, Giáo trình Phân tích và Thiết kế Hệ thống Thông tin, Học viện công nghệ Bưu chính Viễn thông, 2013.
- [7] Trần Đình Quế, Nguyễn Mạnh Hùng, Nhập môn công nghệ phần mềm, Học viện công nghệ Bưu chính Viễn thông, 2020.
- [8] Hector Garcia-Molina,, Database Systems: The Complete Book Second edition, Department of Computer Science Stanford University.