

TRƯỜNG ĐẠI HỌC PHENIKAA
KHOA CÔNG NGHỆ THÔNG TIN



BÁO CÁO HỌC PHẦN
PHÂN TÍCH VÀ THIẾT KẾ PHẦN MỀM
NHÓM 8 - XÂY DỰNG ỨNG DỤNG WEB BÁN SÁCH

Tên sinh viên:	Đỗ Đăng Hoàn - 21010666 Nguyễn Đức Minh – 21010666 Đinh Bá Việt Anh - 21013232
Giảng viên hướng dẫn:	Vũ Quang Dũng
Khoa:	Công nghệ thông tin

HÀ NỘI, 02/2025

	MỤC LỤC	
1. GIỚI THIỆU		9
1.1. Lý do chọn đề tài		9
1.2. Mục tiêu nghiên cứu		10
1.3. Phạm vi và đối tượng nghiên cứu		10
1.4. Phương pháp nghiên cứu		11
1.5. Cấu trúc báo cáo		12
2. KHẢO SÁT VÀ PHÂN TÍCH HỆ THỐNG		14
2.1. Giới thiệu về hệ thống web bán sách		14
2.2. Khảo sát các hệ thống tương tự		14
2.2.1. Tiki		15
2.2.2. Fahasa		15
2.2.3. Bài học rút ra		15
2.3. Xác định yêu cầu hệ thống		16
2.3.1. Yêu cầu chức năng		16
2.3.2. Yêu cầu phi chức năng		17
2.4. Mô hình nghiệp vụ (Business Model)		18
2.5. Phân tích người dùng (User Persona)		19
2.5.1. Persona 1: Khách hàng		19
2.5.2. Persona 2: Quản trị viên		19
3. PHÂN TÍCH HỆ THỐNG		20
3.1. Mô hình Use Case		20
3.1.1. User case khách hàng		20
3.1.2. Usecase admin		22
3.2. Biểu đồ hoạt động (Activity Diagram)		24
3.3. Biểu đồ trình tự (Sequence Diagram)		25

3.3.1. Người dùng	25
3.3.2. Người quản lý	29
3.4. Biểu đồ lớp (Class Diagram)	32
3.5. Biểu đồ thực thể - quan hệ (ERD)	33
4. THIẾT KẾ HỆ THỐNG	34
4.1. Kiến trúc tổng quan	34
4.2. Thiết kế giao diện người dùng	36
4.2.1. Trang chủ	36
4.2.2. Trang chi tiết sản phẩm	37
4.2.3. Giỏ hàng.....	38
4.2.4. Đăng nhập.....	39
4.2.6. Quên mật khẩu	41
4.2.7. Trang người dùng.....	41
4.2.8. Trang admin.....	42
4.2.9. Tìm kiếm	45
4.2.10. Bộ lọc.....	46
4.3. Thiết kế cơ sở dữ liệu	47
4.4. Thiết kế API backend	48
5. CÀI ĐẶT VÀ TRIỂN KHAI	50
5.1. Công nghệ sử dụng	50
5.2. Triển khai backend với Node.js	50
5.3. Triển khai frontend với Vue.js	51
5.4. Quản lý cơ sở dữ liệu với MySQL	52
5.5. Kết nối API giữa frontend và backend	53
6. KIỂM THỬ VÀ ĐÁNH GIÁ	54

6.1. Phương pháp kiểm thử.....	54
6.2. Kết quả kiểm thử	54
6.3. Đánh giá hiệu suất hệ thống.....	55
6.4. Hạn chế và hướng phát triển	55

DANH MỤC HÌNH ẢNH

<i>Hình 1: Mô hình nghiệp vụ</i>	<i>18</i>
<i>Hình 2: Use Case khách hàng</i>	<i>20</i>
<i>Hình 3: Use Case Admin</i>	<i>22</i>
<i>Hình 4: Biểu đồ hoạt động thanh toán đơn hàng.....</i>	<i>24</i>
<i>Hình 5: Biểu đồ trình tự đăng ký.....</i>	<i>25</i>
<i>Hình 6: Biểu đồ trình tự đăng nhập.....</i>	<i>26</i>
<i>Hình 7: Biểu đồ trình tự xem chi tiết sách</i>	<i>26</i>
<i>Hình 8: Biểu đồ trình tự thêm vào giỏ hàng</i>	<i>27</i>
<i>Hình 9: Biểu đồ trình tự cập nhật giỏ hàng.....</i>	<i>27</i>
<i>Hình 10: Biểu đồ trình tự đặt hàng.....</i>	<i>28</i>
<i>Hình 11: Biểu đồ trình tự xem lịch sử đơn hàng.....</i>	<i>28</i>
<i>Hình 12: Biểu đồ trình tự quản lý sách</i>	<i>29</i>
<i>Hình 13: Biểu đồ trình tự quản lý đơn hàng.....</i>	<i>30</i>
<i>Hình 14: Biểu đồ trình tự quản lý thể loại</i>	<i>31</i>
<i>Hình 15: Biểu đồ lớp.....</i>	<i>32</i>
<i>Hình 16: Biểu đồ thực thể - quan hệ 1.....</i>	<i>33</i>
<i>Hình 17: Biểu đồ thực thể - quan hệ 2.....</i>	<i>34</i>
<i>Hình 18: Trang chủ.....</i>	<i>37</i>
<i>Hình 19: Trang chi tiết sản phẩm</i>	<i>38</i>
<i>Hình 20: Trang giỏ hàng.....</i>	<i>38</i>
<i>Hình 21: Popup đăng nhập.....</i>	<i>39</i>
<i>Hình 22: Popup đăng ký.....</i>	<i>40</i>
<i>Hình 23: Popup quên mật khẩu.....</i>	<i>41</i>
<i>Hình 24: Trang thông tin người dùng</i>	<i>41</i>
<i>Hình 25: Địa chỉ người dùng.....</i>	<i>42</i>
<i>Hình 26: Trang sản phẩm yêu thích của người dùng.....</i>	<i>42</i>

<i>Hình 27: Trang Admin Dashboard</i>	42
<i>Hình 28: Trang quản lý sách</i>	43
<i>Hình 29: Trang quản lý người dùng</i>	43
<i>Hình 30: Trang quản lý đơn hàng</i>	43
<i>Hình 31: Trang quản lý Banner</i>	44
<i>Hình 32: Trang quản lý Tag Book</i>	44
<i>Hình 33: Trang quản lý Khuyến mại</i>	44
<i>Hình 34: Khung tìm kiếm sản phẩm</i>	45
<i>Hình 35: Kết quả sau tìm kiếm</i>	45
<i>Hình 36: Bộ lọc sản phẩm</i>	46

BẢNG PHÂN CÔNG CÔNG VIỆC

Người đảm nhiệm	Vai trò	Nhiệm vụ
Đỗ Đăng Hoàn	Nhóm trưởng	<ul style="list-style-type: none"> - Thiết kế và phát triển giao diện người dùng (UI) bằng Vue.js - Kết nối API từ backend bằng Axios. - Thiết kế các trang: Trang chủ, Chi tiết sản phẩm, Giỏ hàng, Đăng nhập/Đăng ký, Trang người dùng. - Kiểm thử giao diện (User Acceptance Test). - Tham gia viết báo cáo, trang thuyết trình.
Nguyễn Đức Minh	Thành viên	<ul style="list-style-type: none"> - Thiết kế và quản lý cơ sở dữ liệu bằng MySQL. - Xây dựng các bảng (users, books, orders, carts, v.v.) và mối quan hệ (ERD). - Tối ưu hóa truy vấn bằng index. - Quản lý dữ liệu mẫu để kiểm thử. - Đảm bảo tính toàn vẹn dữ liệu (chuẩn hóa, ràng buộc). - Hỗ trợ kết nối backend với cơ sở dữ liệu (mysql2). - Tham gia viết báo cáo, trang thuyết trình.
Đinh Bá Việt Anh	Thành viên	<ul style="list-style-type: none"> - Phát triển backend bằng Node.js và Express.js.

		<ul style="list-style-type: none"> - Xây dựng RESTful API cho các chức năng (quản lý người dùng, sách, đơn hàng, thanh toán, v.v.). - Xác thực và phân quyền. - Xử lý logic nghiệp vụ (đăng ký, đăng nhập, đặt hàng, quản lý kho). - Bảo mật API (HTTPS, giới hạn request, mã hóa mật khẩu). - Kiểm thử tích hợp (Integration Test). - Tham gia viết báo cáo, trang thuyết trình.
--	--	---

MỞ ĐẦU

Trong bối cảnh công nghệ thông tin phát triển mạnh mẽ, thương mại điện tử đang dần thay thế phương thức mua sắm truyền thống. Ngành kinh doanh sách cũng không nằm ngoài xu hướng này khi các nền tảng bán sách trực tuyến ngày càng được ưa chuộng. Việc xây dựng một ứng dụng web bán sách giúp kết nối người bán và người mua một cách dễ dàng, thuận tiện hơn, đồng thời mở rộng thị trường và nâng cao hiệu quả kinh doanh.

Ứng dụng web bán sách không chỉ cung cấp nền tảng để khách hàng tìm kiếm, lựa chọn và đặt mua sách trực tuyến mà còn giúp quản lý kho hàng, đơn hàng và hỗ trợ thanh toán một cách hiệu quả. Để xây dựng hệ thống này, cần áp dụng các công nghệ web hiện đại như Vue.js cho giao diện người dùng, Node.js cho backend và MySQL để quản lý cơ sở dữ liệu.

Tuy nhiên, để phát triển một hệ thống chính xác, hiệu quả và có khả năng mở rộng, cần có một quy trình phân tích và thiết kế bài bản. Việc sử dụng các kỹ thuật phân tích như mô hình Use Case, ERD, Class Diagram, Sequence Diagram sẽ giúp làm rõ yêu cầu, xác định kiến trúc hệ thống và đảm bảo tính logic trong quá trình phát triển.

Báo cáo này trình bày quy trình phân tích và thiết kế hệ thống web bán sách, tập trung vào các giai đoạn chính bao gồm xác định yêu cầu, mô hình hóa dữ liệu, thiết kế hệ thống và đánh giá giải pháp. Thông qua nghiên cứu này, báo cáo cung cấp cái nhìn tổng quan về cách ứng dụng các phương pháp phân tích và thiết kế phần mềm vào thực tế, góp phần xây dựng một hệ thống tối ưu và hiệu quả.

1. GIỚI THIỆU

1.1. Lý do chọn đề tài

Trong bối cảnh công nghệ thông tin phát triển mạnh mẽ, thương mại điện tử đã trở thành một xu hướng tất yếu, đóng vai trò quan trọng trong đời sống kinh tế và xã hội hiện đại. Ngành kinh doanh sách – một lĩnh vực mang tính văn hóa với nhu cầu ổn định và đa dạng – cũng không nằm ngoài sự chuyển dịch từ mô hình truyền thống sang các nền tảng trực tuyến. Việc mua sắm sách trực tuyến mang lại nhiều lợi ích thiết thực: sự tiện lợi cho người tiêu dùng, khả năng tiếp cận đa dạng sản phẩm, đồng thời giúp các nhà xuất bản và cửa hàng sách mở rộng thị trường, tối ưu hóa hiệu quả kinh doanh. Tuy nhiên, tại Việt Nam, nhiều cửa hàng sách truyền thống vẫn chưa tận dụng triệt để tiềm năng của thương mại điện tử. Các ứng dụng web bán sách chuyên nghiệp, thân thiện với người dùng vẫn còn hạn chế về số lượng và chất lượng.

Đề tài "**Xây dựng ứng dụng web bán sách**" được lựa chọn nhằm mục đích nghiên cứu và phát triển một hệ thống thương mại điện tử hoàn chỉnh, đáp ứng nhu cầu tìm kiếm, lựa chọn và mua sắm sách trực tuyến của người dùng. Đồng thời, đây cũng là cơ hội để áp dụng các kiến thức về phân tích và thiết kế phần mềm vào một dự án thực tế, giúp nâng cao kỹ năng và hiểu biết về quy trình phát triển phần mềm chuyên nghiệp.

Lý do cụ thể dẫn đến việc chọn đề tài này bao gồm:

- Nhu cầu thực tế: Sự gia tăng của thói quen mua sắm trực tuyến, đặc biệt trong lĩnh vực sách, đòi hỏi những giải pháp công nghệ phù hợp để đáp ứng nhu cầu ngày càng cao của người dùng.
- Tính ứng dụng cao: Ứng dụng web bán sách không chỉ phục vụ mục đích học tập mà còn có tiềm năng triển khai thực tế, mang lại giá trị kinh tế và xã hội.
- Phù hợp với chương trình học: Đề tài cho phép vận dụng các kỹ thuật phân tích và thiết kế phần mềm như mô hình Use Case, ERD, Class Diagram, Sequence Diagram, qua đó củng cố lý thuyết thông qua thực hành.

Việc thực hiện đề tài không chỉ mang ý nghĩa học thuật mà còn góp phần giải quyết một vấn đề thực tiễn trong bối cảnh số hóa hiện nay.

1.2. Mục tiêu nghiên cứu

Mục tiêu chính của nghiên cứu là xây dựng một ứng dụng web bán sách hoàn chỉnh, đáp ứng đầy đủ các yêu cầu chức năng và phi chức năng từ phía người dùng lẫn quản trị viên. Các mục tiêu cụ thể được xác định như sau:

- Xác định và phân tích yêu cầu hệ thống: Tiến hành khảo sát nhu cầu của người dùng và nghiên cứu các hệ thống bán sách trực tuyến hiện có để làm rõ các yêu cầu chức năng (ví dụ: tìm kiếm sách, đặt hàng) và phi chức năng (ví dụ: tốc độ, bảo mật).
- Thiết kế và phát triển hệ thống: Áp dụng các kỹ thuật thiết kế phần mềm để xây dựng một kiến trúc hệ thống rõ ràng, dễ bảo trì và có khả năng mở rộng trong tương lai.
- Triển khai và kiểm thử: Sử dụng các công nghệ hiện đại như Vue.js (frontend), Node.js (backend) và MySQL (database) để phát triển ứng dụng, sau đó kiểm thử toàn diện để đảm bảo hệ thống hoạt động ổn định và chính xác.
- Đánh giá và đề xuất cải tiến: Phân tích hiệu suất hoạt động của hệ thống, nhận diện các hạn chế và đưa ra các giải pháp cải tiến hoặc hướng phát triển nâng cao trong tương lai.

Thông qua việc hoàn thành các mục tiêu trên, nghiên cứu không chỉ tạo ra một sản phẩm cụ thể – ứng dụng web bán sách – mà còn nâng cao hiểu biết về quy trình phát triển phần mềm từ giai đoạn phân tích, thiết kế đến triển khai thực tế. Kết quả của đề tài có thể được sử dụng như một tài liệu tham khảo hoặc nền tảng để phát triển các hệ thống thương mại điện tử tương tự.

1.3. Phạm vi và đối tượng nghiên cứu

Phạm vi nghiên cứu.

Hệ thống web bán sách được đặt tên là "PlayBook", tập trung vào việc cung cấp các chức năng cơ bản và thiết yếu, bao gồm:

- **Đối với người dùng (khách hàng):**
 - + Đăng ký/đăng nhập tài khoản.
 - + Tìm kiếm sách theo từ khóa, thể loại hoặc tác giả.
 - + Xem thông tin chi tiết của sách (tên, giá, mô tả, hình ảnh).

- + Thêm sách vào giỏ hàng, đặt hàng và theo dõi trạng thái đơn hàng.
- **Đối với quản trị viên:**
 - + Quản lý danh mục sách (thêm, sửa, xóa thông tin sách).
 - + Quản lý kho hàng (cập nhật số lượng tồn kho).
 - + Xử lý đơn hàng (xác nhận, hủy, cập nhật trạng thái).
 - + Xem báo cáo doanh thu cơ bản.

Giới hạn phạm vi: Nghiên cứu không bao gồm các tính năng nâng cao như tích hợp cổng thanh toán trực tuyến, hệ thống quản lý khuyến mãi phức tạp, hoặc chức năng đề xuất sách thông minh dựa trên sở thích người dùng.

Đối tượng nghiên cứu.

Người dùng (khách hàng): Bao gồm các nhóm đối tượng có nhu cầu mua sách trực tuyến như học sinh, sinh viên, người đi làm, hoặc những người yêu thích đọc sách nói chung. Đây là nhóm chính sử dụng các chức năng mua sắm của hệ thống.

Quản trị viên: Là những người quản lý hệ thống, có thể là nhân viên cửa hàng sách hoặc quản lý kho hàng. Nhiệm vụ của họ bao gồm cập nhật thông tin sách, xử lý đơn hàng và theo dõi doanh thu.

Phạm vi và đối tượng nghiên cứu được xác định rõ ràng nhằm đảm bảo tính khả thi của dự án trong khuôn khổ học phần, đồng thời tập trung vào các chức năng cốt lõi để đáp ứng nhu cầu cơ bản của người dùng.

1.4. Phương pháp nghiên cứu.

Để thực hiện đề tài, nhóm nghiên cứu áp dụng các phương pháp sau một cách có hệ thống:

Khảo sát và phân tích: Nghiên cứu các hệ thống bán sách trực tuyến phổ biến tại Việt Nam như Tiki, Fahasa để học hỏi về giao diện, chức năng và trải nghiệm người dùng. Từ đó, xác định các yêu cầu cần thiết cho hệ thống "PlayBook".

Mô hình hóa hệ thống: Sử dụng các công cụ và kỹ thuật phân tích, thiết kế phần mềm:

- Use Case Diagram: Mô tả các chức năng chính và tương tác giữa người dùng (khách hàng, quản trị viên) với hệ thống.

- Activity Diagram: Mô tả luồng hoạt động của các quy trình như tìm kiếm sách, đặt hàng.
- Sequence Diagram: Chi tiết hóa trình tự thực hiện các chức năng quan trọng như đặt hàng, xử lý đơn hàng.
- Class Diagram: Thiết kế cấu trúc lớp của hệ thống, bao gồm các đối tượng như User, Book, Order.
- ERD (Entity-Relationship Diagram): Mô hình hóa cơ sở dữ liệu, xác định các thực thể (User, Book, Order) và mối quan hệ giữa chúng.

Phát triển và triển khai: Sử dụng các công nghệ web hiện đại:

- Frontend: Vue.js để xây dựng giao diện người dùng động, thân thiện và responsive.
- Backend: Node.js với framework Express để xử lý logic nghiệp vụ và cung cấp API.
- Database: MySQL để lưu trữ và quản lý dữ liệu như thông tin sách, đơn hàng, tài khoản người dùng.

Kiểm thử: Áp dụng các phương pháp kiểm thử:

- Unit Test: Kiểm tra từng module riêng lẻ.
- Integration Test: Đảm bảo các module hoạt động tốt khi tích hợp.
- User Acceptance Test: Kiểm tra với người dùng thực tế để đánh giá tính khả dụng và hiệu quả.

Phương pháp nghiên cứu được xây dựng dựa trên quy trình phát triển phần mềm truyền thống (Waterfall) kết hợp với cách tiếp cận thực hành, đảm bảo tính khoa học và thực tiễn trong quá trình thực hiện đề tài.

1.5. Cấu trúc báo cáo

Báo cáo được tổ chức thành các chương chính, phản ánh đầy đủ quá trình nghiên cứu và phát triển hệ thống:

Chương 1: Mở đầu

Giới thiệu lý do chọn đề tài, mục tiêu, phạm vi, phương pháp nghiên cứu và cấu trúc báo cáo.

Chương 2: Khảo sát và phân tích hệ thống

Trình bày quá trình khảo sát các hệ thống tương tự, xác định yêu cầu chức năng và phi chức năng, phân tích người dùng và mô hình nghiệp vụ.

Chương 3: Phân tích hệ thống

Mô tả chi tiết các biểu đồ như Use Case, Activity Diagram, Sequence Diagram, Class Diagram và ERD.

Chương 4: Thiết kế hệ thống

Thiết kế kiến trúc tổng quan, giao diện người dùng, cơ sở dữ liệu và API backend.

Chương 5: Cài đặt và triển khai

Giới thiệu công nghệ sử dụng, quy trình triển khai frontend, backend và kết nối API.

Chương 6: Kiểm thử và đánh giá

Trình bày phương pháp kiểm thử, kết quả, đánh giá hiệu suất và đề xuất cải tiến.

Kết luận

Tóm tắt quá trình nghiên cứu, đánh giá kết quả đạt được và đề xuất hướng phát triển trong tương lai.

Tài liệu tham khảo

Liệt kê các nguồn tài liệu, sách, bài báo, website được tham khảo trong quá trình thực hiện.

Cấu trúc báo cáo được xây dựng logic, chặt chẽ, nhằm đảm bảo nội dung được trình bày rõ ràng, dễ theo dõi và phản ánh đầy đủ các bước thực hiện đề tài.

2. KHẢO SÁT VÀ PHÂN TÍCH HỆ THỐNG

2.1. Giới thiệu về hệ thống web bán sách

Hệ thống web bán sách "PlayBook" là một nền tảng thương mại điện tử được thiết kế nhằm cung cấp dịch vụ mua sắm sách trực tuyến tiện lợi cho người dùng. Hệ thống cho phép khách hàng dễ dàng tìm kiếm, xem thông tin chi tiết và đặt mua các loại sách thuộc nhiều thể loại khác nhau. Đồng thời, "PlayBook" cũng hỗ trợ quản trị viên quản lý danh mục sách, kho hàng và đơn hàng, giúp tối ưu hóa quy trình kinh doanh.

Mục tiêu của hệ thống:

- Cung cấp một giao diện thân thiện, dễ sử dụng để người dùng tìm kiếm và mua sắm sách.
- Hỗ trợ quản lý hiệu quả các hoạt động bán hàng, kho hàng và đơn hàng.
- Đảm bảo tính bảo mật, độ tin cậy và khả năng mở rộng của hệ thống trong tương lai.
- Các thành phần chính của hệ thống
- Frontend: Giao diện người dùng được phát triển bằng Vue.js, mang lại trải nghiệm mượt mà và tương thích trên nhiều thiết bị (máy tính, điện thoại, máy tính bảng).
- Backend: Được xây dựng bằng Node.js và Express, chịu trách nhiệm xử lý logic nghiệp vụ và cung cấp các API cho frontend.
- Database: Sử dụng MySQL để lưu trữ thông tin về sách, người dùng, đơn hàng và các dữ liệu liên quan khác.

Hệ thống "PlayBook" không chỉ đơn thuần là một website bán sách mà còn là một ví dụ thực tiễn về việc áp dụng các nguyên tắc phân tích và thiết kế phần mềm để xây dựng một nền tảng thương mại điện tử hoàn chỉnh, đáp ứng nhu cầu của cả khách hàng và quản trị viên.

2.2. Khảo sát các hệ thống tương tự

Để phát triển hệ thống "PlayBook", nhóm nghiên cứu đã tiến hành khảo sát và phân tích một số hệ thống web bán sách hiện có trên thị trường. Mục tiêu là nhận diện các điểm mạnh, điểm yếu của các hệ thống này, từ đó rút ra bài học và cải tiến cho "PlayBook".

2.2.1. Tiki

Điểm mạnh:

- Giao diện người dùng trực quan, dễ điều hướng.
- Tính năng tìm kiếm nhanh chóng, chính xác nhờ sử dụng thuật toán tối ưu.
- Hệ thống gợi ý sản phẩm thông minh dựa trên lịch sử mua sắm của khách hàng.
- Hỗ trợ nhiều phương thức thanh toán (thẻ tín dụng, ví điện tử, COD) và giao hàng linh hoạt.

Điểm yếu:

- Thời gian tải trang đôi khi chậm khi hiển thị danh sách sản phẩm lớn.
- Chưa hỗ trợ lọc sản phẩm theo nhiều tiêu chí kết hợp (ví dụ: giá + thể loại + đánh giá).

2.2.2. Fahasa

Điểm mạnh:

- Phân loại sách rõ ràng theo thể loại, tác giả và nhà xuất bản, giúp khách hàng dễ dàng tìm kiếm.
- Chương trình khuyến mãi và giảm giá thường xuyên, thu hút người dùng.
- Quản lý kho hàng hiệu quả, cập nhật số lượng tồn kho chính xác.

Điểm yếu:

- Giao diện chưa tối ưu cho thiết bị di động, gây khó khăn khi sử dụng trên điện thoại.
- Thiếu tính năng đánh giá và nhận xét từ khách hàng, làm giảm tương tác.

2.2.3. Bài học rút ra

Từ kết quả khảo sát, nhóm nghiên cứu rút ra các điểm cần lưu ý để thiết kế hệ thống "PlayBook":

- Giao diện cần trực quan, thân thiện và tương thích trên mọi thiết bị.
- Tốc độ tải trang phải nhanh, đặc biệt khi hiển thị danh sách sản phẩm lớn.
- Tích hợp tính năng tìm kiếm và lọc sản phẩm linh hoạt theo nhiều tiêu chí.
- Quản lý kho hàng và đơn hàng cần chính xác, cập nhật theo thời gian thực.

Dựa trên những bài học này, "PlayBook" sẽ được phát triển để khắc phục các hạn chế và phát huy ưu điểm của các hệ thống tương tự, tạo ra một nền tảng vượt trội hơn.

2.3. Xác định yêu cầu hệ thống

2.3.1. Yêu cầu chức năng

Yêu cầu chức năng xác định các tính năng mà hệ thống "PlayBook" cần cung cấp để đáp ứng nhu cầu của khách hàng và quản trị viên.

- Đối với khách hàng
 - + Đăng ký tài khoản: Người dùng tạo tài khoản bằng cách cung cấp thông tin cơ bản (tên, email, mật khẩu).
 - + Đăng nhập: Đăng nhập vào hệ thống bằng email và mật khẩu đã đăng ký.
 - + Tìm kiếm sách: Tìm kiếm sách theo từ khóa (tên sách, tác giả, thể loại) hoặc bộ lọc (giá, đánh giá, thể loại).
 - + Xem chi tiết sách: Hiển thị đầy đủ thông tin sách, bao gồm tên, tác giả, giá, mô tả, hình ảnh và số lượng tồn kho.
 - + Thêm vào giỏ hàng: Cho phép thêm sách vào giỏ hàng để chuẩn bị đặt hàng.
 - + Đặt hàng: Xem giỏ hàng, chỉnh sửa số lượng, nhập thông tin giao hàng và xác nhận đặt hàng.
 - + Theo dõi đơn hàng: Kiểm tra trạng thái đơn hàng (đang xử lý, đã giao, hoàn tất, v.v.).
 - + Đánh giá sách: Để lại nhận xét và đánh giá cho các sách đã mua.
- Đối với quản trị viên
 - + Quản lý danh mục sách: Thêm, sửa, xóa thông tin sách (tên, tác giả, giá, thể loại, hình ảnh).
 - + Quản lý kho hàng: Cập nhật số lượng tồn kho của từng sách theo thời gian thực.
 - + Quản lý đơn hàng: Xem danh sách đơn hàng, cập nhật trạng thái (xác nhận, hủy, giao hàng).
 - + Quản lý người dùng: Xem danh sách người dùng, khóa/mở khóa tài khoản nếu cần thiết.

- + Xem báo cáo doanh thu: Thống kê doanh thu theo ngày, tháng hoặc sản phẩm cụ thể.

2.3.2. Yêu cầu phi chức năng

Yêu cầu phi chức năng xác định các tiêu chuẩn về chất lượng, hiệu suất và bảo mật mà hệ thống cần đáp ứng.

Hiệu suất:

- Thời gian phản hồi cho mỗi yêu cầu không vượt quá 2 giây.
- Hệ thống xử lý được ít nhất 1000 người dùng truy cập đồng thời mà không bị quá tải.

Bảo mật:

- Mật khẩu người dùng được mã hóa trước khi lưu trữ (ví dụ: sử dụng bcrypt).
- Sử dụng giao thức HTTPS để bảo vệ dữ liệu truyền tải.
- Phân quyền rõ ràng giữa khách hàng và quản trị viên.

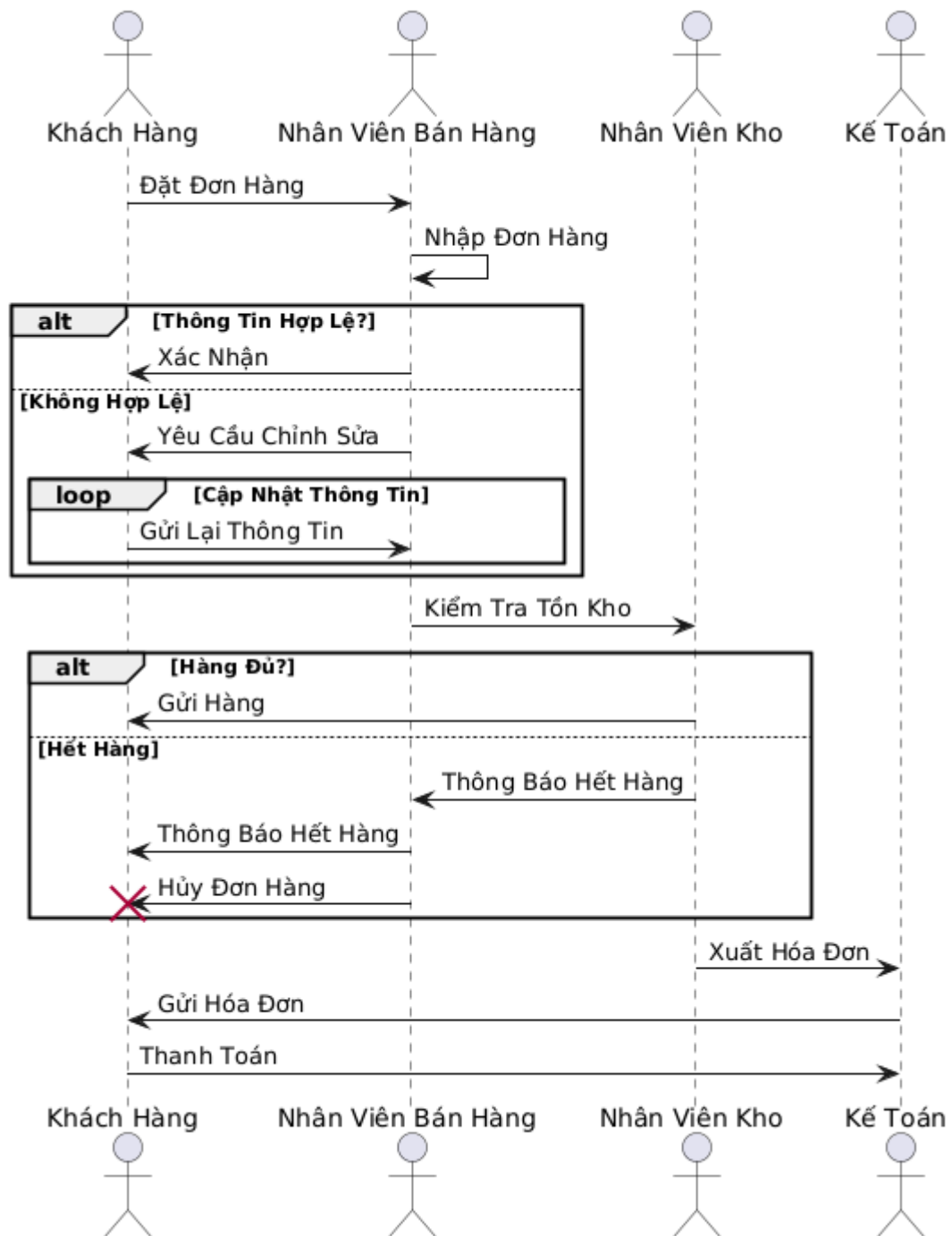
Khả năng mở rộng:

- Thiết kế theo kiến trúc module, dễ dàng tích hợp thêm tính năng mới trong tương lai.

Tính khả dụng:

- Hệ thống hoạt động liên tục 24/7, với thời gian downtime không quá 1% mỗi tháng.
- Tính thân thiện với người dùng:
- Giao diện đơn giản, trực quan, hỗ trợ responsive trên các thiết bị di động và máy tính bảng.

2.4. Mô hình nghiệp vụ (Business Model)



Hình 1: Mô hình nghiệp vụ

Biểu đồ tuần tự mô tả quy trình xử lý đơn hàng với sự tham gia của Khách Hàng, Nhân Viên Bán Hàng, Nhân Viên Kho và Kế Toán. Quá trình bắt đầu khi Khách Hàng gửi yêu cầu "Đặt Đơn Hàng" đến Nhân Viên Bán Hàng. Tiếp theo, Nhân Viên Bán Hàng nhận và kiểm tra thông tin đơn hàng. Nếu thông tin hợp lệ, đơn hàng được xác nhận với

Khách Hàng; nếu không, Khách Hàng sẽ được yêu cầu chỉnh sửa thông tin và gửi lại cho đến khi hợp lệ.

Sau đó, Nhân Viên Bán Hàng chuyển yêu cầu "Kiểm Tra Tồn Kho" đến Nhân Viên Kho. Nếu hàng đủ, Nhân Viên Kho sẽ "Gửi Hàng" trực tiếp đến Khách Hàng. Nếu hết hàng, Nhân Viên Kho thông báo cho Nhân Viên Bán Hàng, người sẽ thông báo lại cho Khách Hàng và hủy đơn hàng, kết thúc quy trình. Khi hàng được gửi thành công, Nhân Viên Kho yêu cầu Kế Toán "Xuất Hóa Đơn", sau đó Kế Toán gửi hóa đơn đến Khách Hàng. Cuối cùng, Khách Hàng thực hiện "Thanh Toán" để hoàn tất quy trình.

2.5. Phân tích người dùng (User Persona)

2.5.1. Persona 1: Khách hàng

Tên: Nguyễn Văn A

Tuổi: 25

Nghề nghiệp: Sinh viên

Sở thích: Đọc sách kỹ năng sống, tiểu thuyết

Nhu cầu:

- Tìm kiếm sách nhanh chóng theo từ khóa hoặc thể loại.
- Xem đánh giá của người dùng khác để quyết định mua.
- Giao diện đơn giản, dễ sử dụng trên điện thoại.

Kỳ vọng: Giao hàng nhanh, chính xác, có thể theo dõi trạng thái đơn hàng.

2.5.2. Persona 2: Quản trị viên

Tên: Trần Thị B

Tuổi: 30

Nghề nghiệp: Nhân viên quản lý cửa hàng sách

Nhu cầu:

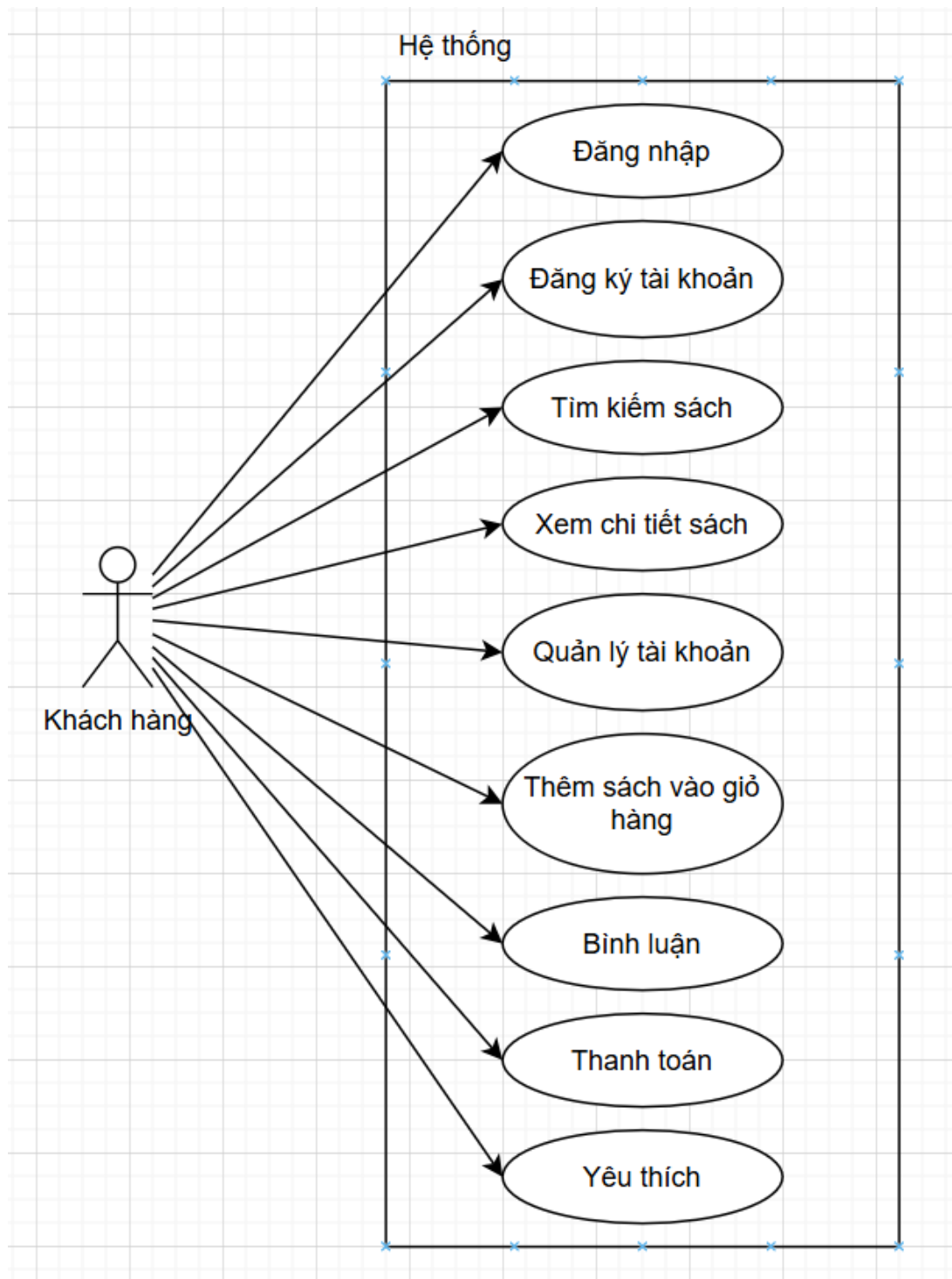
- Công cụ quản lý sách và kho hàng dễ sử dụng (thêm, sửa, xóa sách; cập nhật tồn kho).
- Xem báo cáo doanh thu và đơn hàng chi tiết.
- Quản lý đơn hàng nhanh chóng, cập nhật trạng thái hiệu quả.

Kỳ vọng: Hệ thống ổn định, dễ truy cập từ xa, giao diện thân thiện.

3. PHÂN TÍCH HỆ THỐNG

3.1. Mô hình Use Case

3.1.1. User case khách hàng

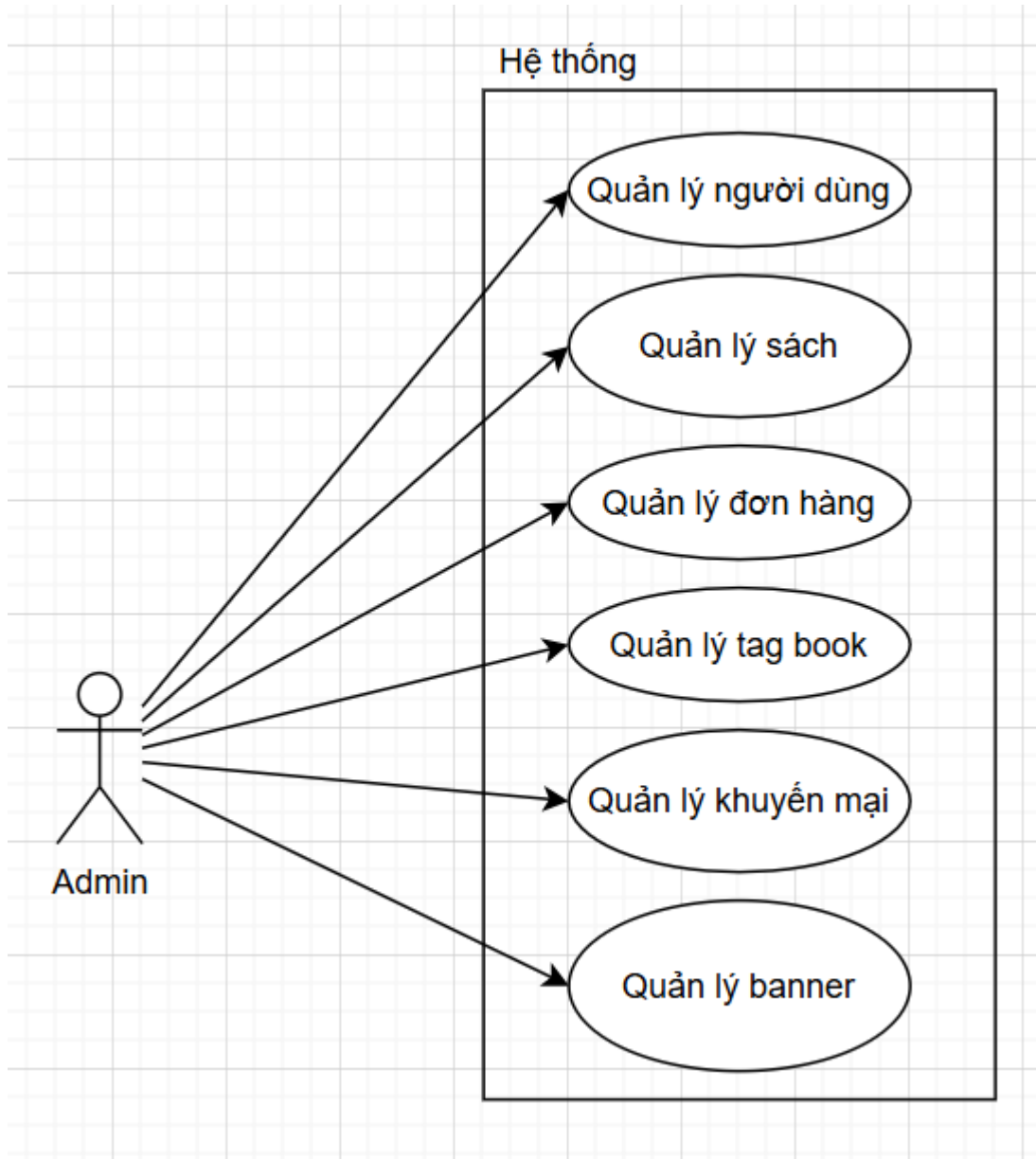


Hình 2: Usecase khách hàng

Hệ thống bán sách trực tuyến với tác nhân chính là "Khách hàng". Hệ thống này được thiết kế nhằm đáp ứng nhu cầu của người dùng khi mua sách trên nền tảng thương mại điện tử. Khách hàng có thể tương tác với hệ thống thông qua chín chức năng chính bao gồm: đăng nhập, đăng ký tài khoản, tìm kiếm sách, xem chi tiết sách, quản lý tài khoản, thêm sách vào giỏ hàng, bình luận, thanh toán và yêu thích. Mỗi chức năng được thể hiện bằng một hình ellipse kết nối với tác nhân khách hàng thông qua các đường liên kết. Chức năng đăng nhập và đăng ký tài khoản cho phép người dùng tạo và truy cập vào tài khoản cá nhân. Các chức năng tìm kiếm và xem chi tiết sách giúp khách hàng dễ dàng tìm thấy thông tin về sách họ quan tâm.

Hệ thống cũng hỗ trợ khách hàng quản lý thông tin cá nhân thông qua chức năng quản lý tài khoản. Chức năng thêm sách vào giỏ hàng và thanh toán tạo nên quy trình mua hàng cơ bản. Ngoài ra, khách hàng còn có thể tương tác với cộng đồng thông qua việc bình luận và đánh dấu sách yêu thích để theo dõi sau này. Mặc dù biểu đồ đã bao quát các chức năng cơ bản, nhưng vẫn có thể bổ sung thêm một số chức năng như quản lý đơn hàng, theo dõi tình trạng giao hàng hay xem lịch sử mua hàng để nâng cao trải nghiệm người dùng. Nhìn chung, biểu đồ use case này cung cấp một nền tảng vững chắc để phát triển các mô hình chi tiết hơn và thiết kế giao diện người dùng cho hệ thống bán sách trực tuyến.

3.1.2. Usecase admin



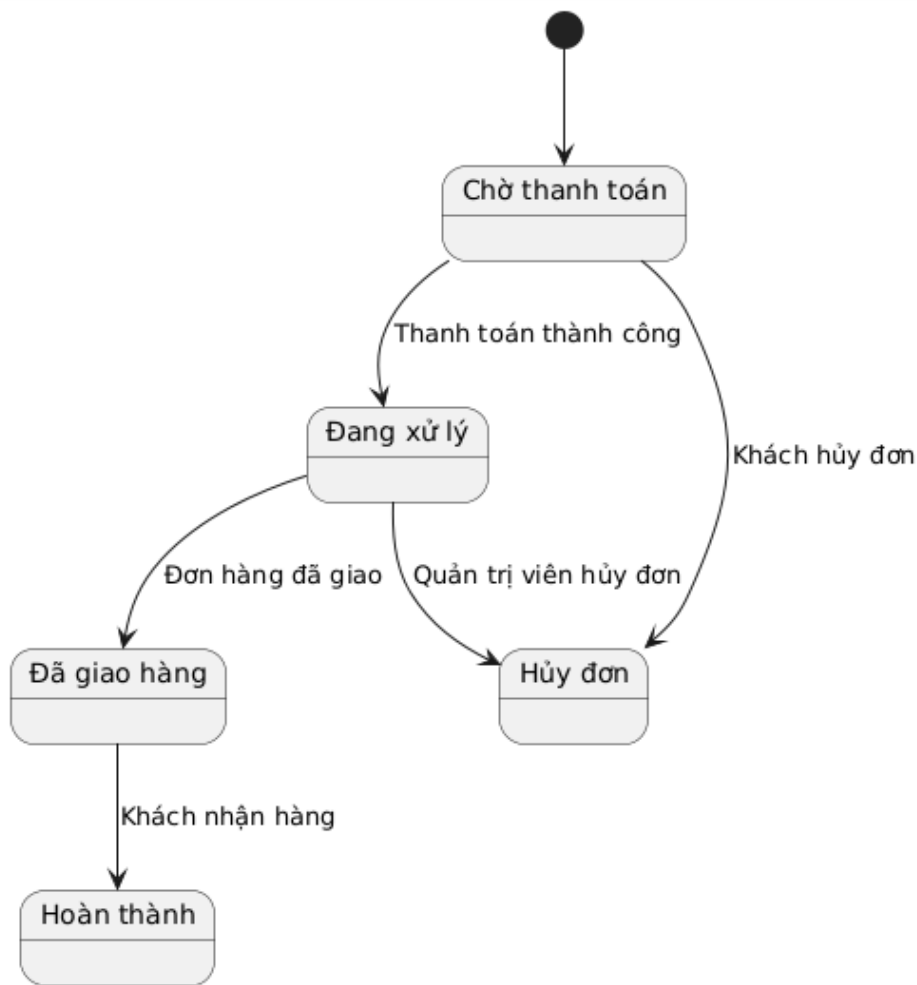
Hình 3: Usecase Admin

Use case này thể hiện phần hệ thống dành cho tác nhân "Admin" trong cùng hệ thống bán sách trực tuyến. Khác với vai trò của khách hàng, Admin đóng vai trò quản trị viên với các chức năng quản lý toàn diện hệ thống. Biểu đồ chỉ ra sáu chức năng chính mà Admin có thể thực hiện, bao gồm: quản lý người dùng, quản lý sách, quản lý đơn hàng, quản lý tag book, quản lý khuyến mãi và quản lý banner. Mỗi chức năng được biểu diễn bằng hình ellipse và được kết nối với tác nhân Admin thông qua các đường liên kết. Chức năng quản lý người dùng cho phép Admin kiểm soát thông tin và quyền

truy cập của khách hàng trong hệ thống, trong khi chức năng quản lý sách giúp Admin thêm, sửa, xóa thông tin sách trong cơ sở dữ liệu.

Chức năng quản lý đơn hàng cho phép Admin theo dõi và xử lý các đơn đặt hàng từ khách hàng, còn chức năng quản lý tag book giúp phân loại sách theo các thể loại hoặc chủ đề để thuận tiện cho việc tìm kiếm. Ngoài ra, Admin còn có thể thiết lập các chương trình khuyến mãi thông qua chức năng quản lý khuyến mãi và điều chỉnh giao diện trang web qua chức năng quản lý banner. Các chức năng này tạo thành một hệ thống quản trị toàn diện, cho phép Admin kiểm soát mọi khía cạnh của hệ thống bán sách trực tuyến, từ sản phẩm, người dùng đến giao diện và chiến lược marketing. Biểu đồ use case này kết hợp với biểu đồ của khách hàng tạo nên bức tranh tổng thể về các chức năng của hệ thống bán sách trực tuyến, phục vụ cả người dùng cuối và đội ngũ quản trị.

3.2. Biểu đồ hoạt động (Activity Diagram)



Hình 4: Biểu đồ hoạt động thanh toán đơn hàng

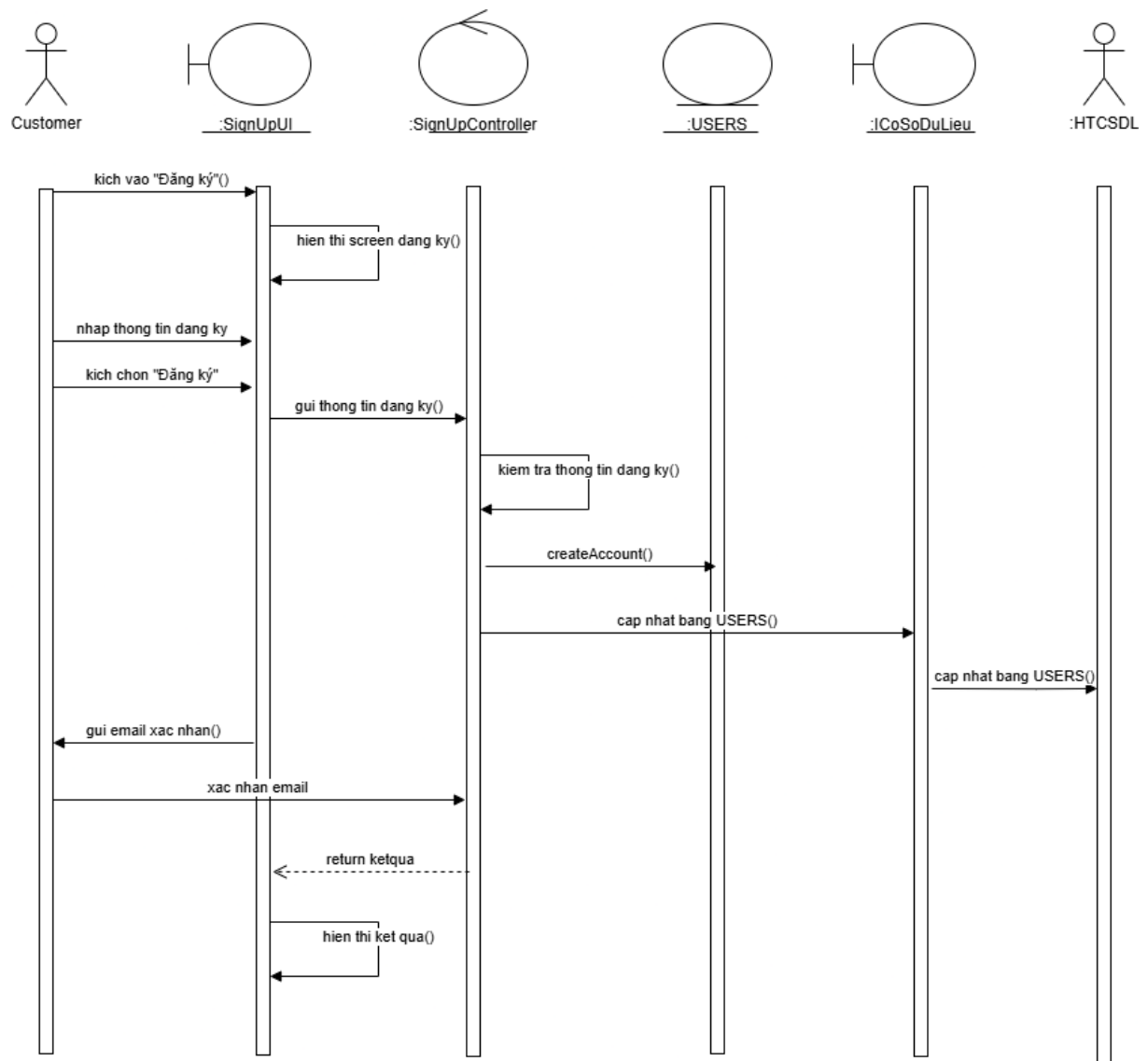
Quy trình xử lý đơn hàng bắt đầu từ khi đơn hàng được tạo và ở trạng thái "Chờ thanh toán". Ở bước này, khách hàng cần thanh toán để đơn hàng được xử lý tiếp. Nếu thanh toán thành công, đơn hàng chuyển sang trạng thái "Đang xử lý", nơi hệ thống sẽ chuẩn bị và đóng gói hàng để giao.

Trong giai đoạn "Đang xử lý", có thể xảy ra hai trường hợp. Nếu khách hàng hoặc quản trị viên quyết định hủy đơn, đơn hàng sẽ chuyển sang trạng thái "Hủy đơn" và dừng lại. Còn nếu không bị hủy, đơn hàng sẽ được giao đến khách hàng, chuyển sang trạng thái "Đã giao hàng".

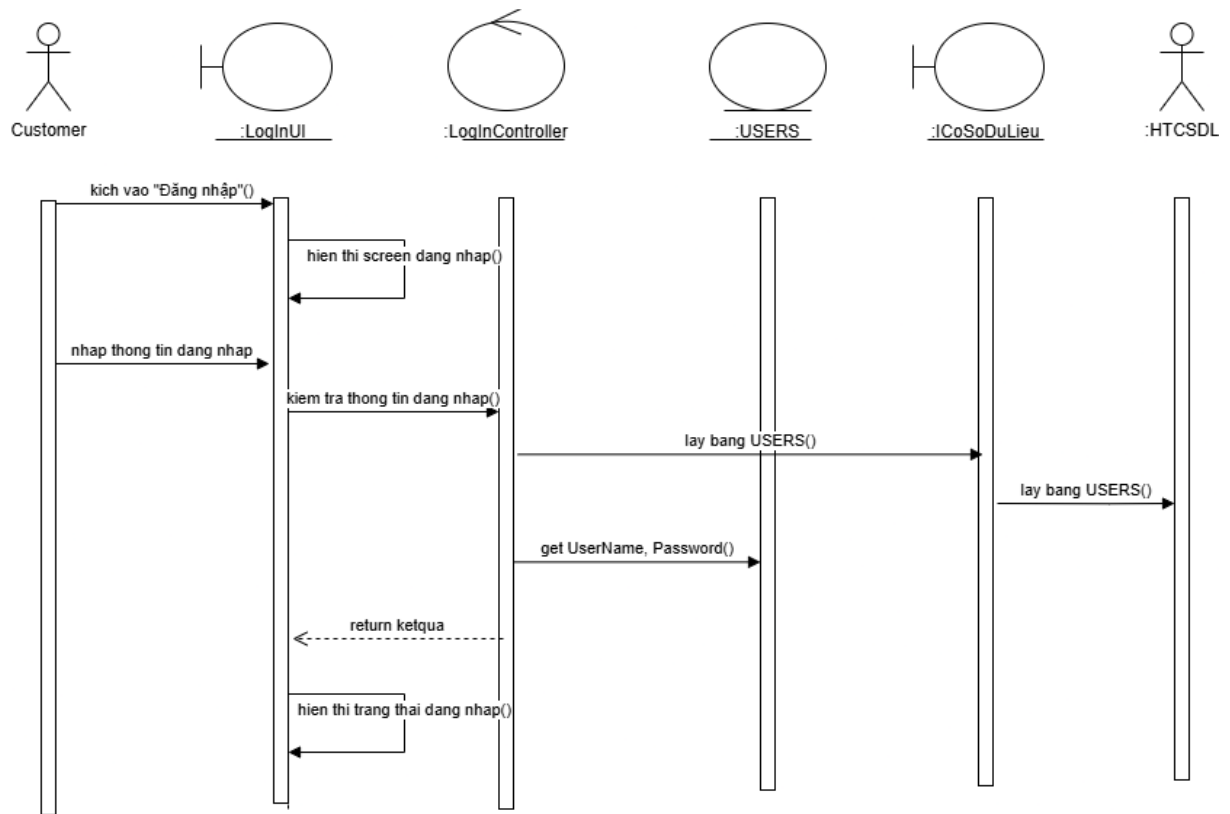
Khi hàng đã được giao, nếu khách hàng nhận hàng thành công và không có vấn đề gì, đơn hàng sẽ chuyển sang trạng thái "Hoàn thành", đánh dấu quy trình kết thúc. Tuy nhiên, quy trình này chưa đề cập đến các trường hợp như thanh toán thất bại, giao hàng không thành công, hay khách từ chối nhận hàng, nên cần bổ sung để đầy đủ hơn.

3.3. Biểu đồ trình tự (Sequence Diagram)

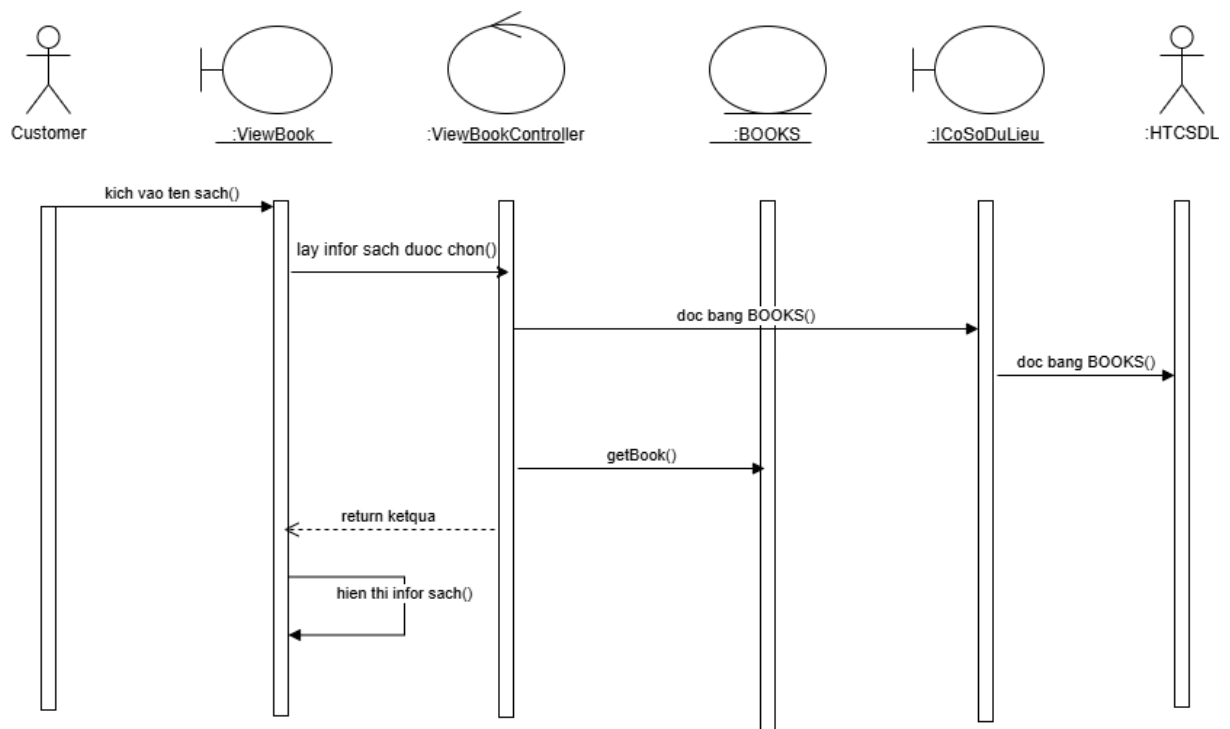
3.3.1. Người dùng



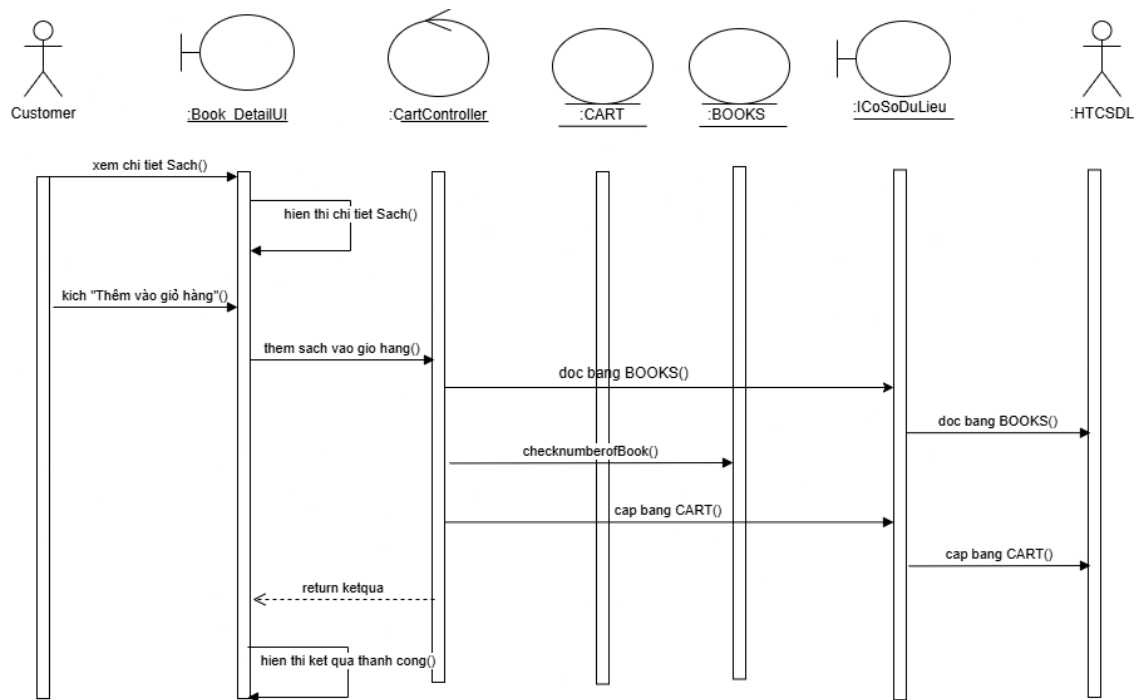
Hình 5: Biểu đồ trình tự đăng ký



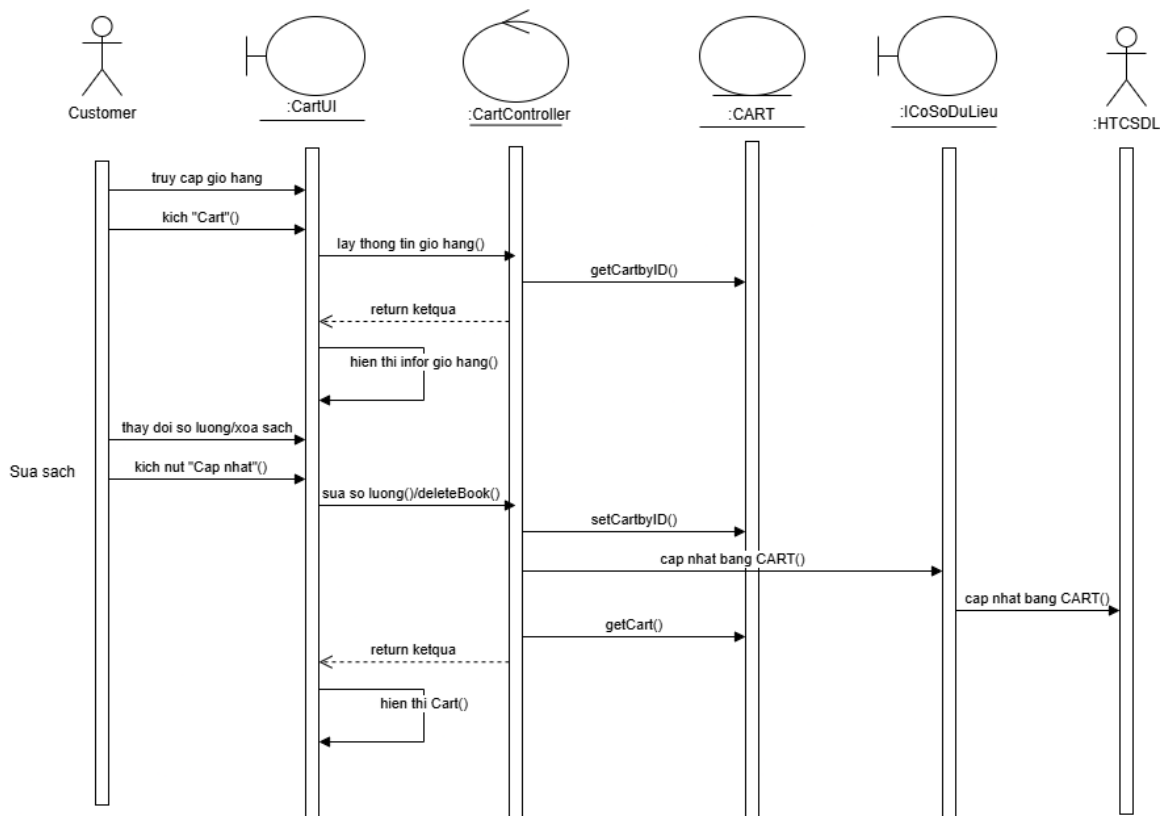
Hình 6: Biểu đồ trình tự đăng nhập



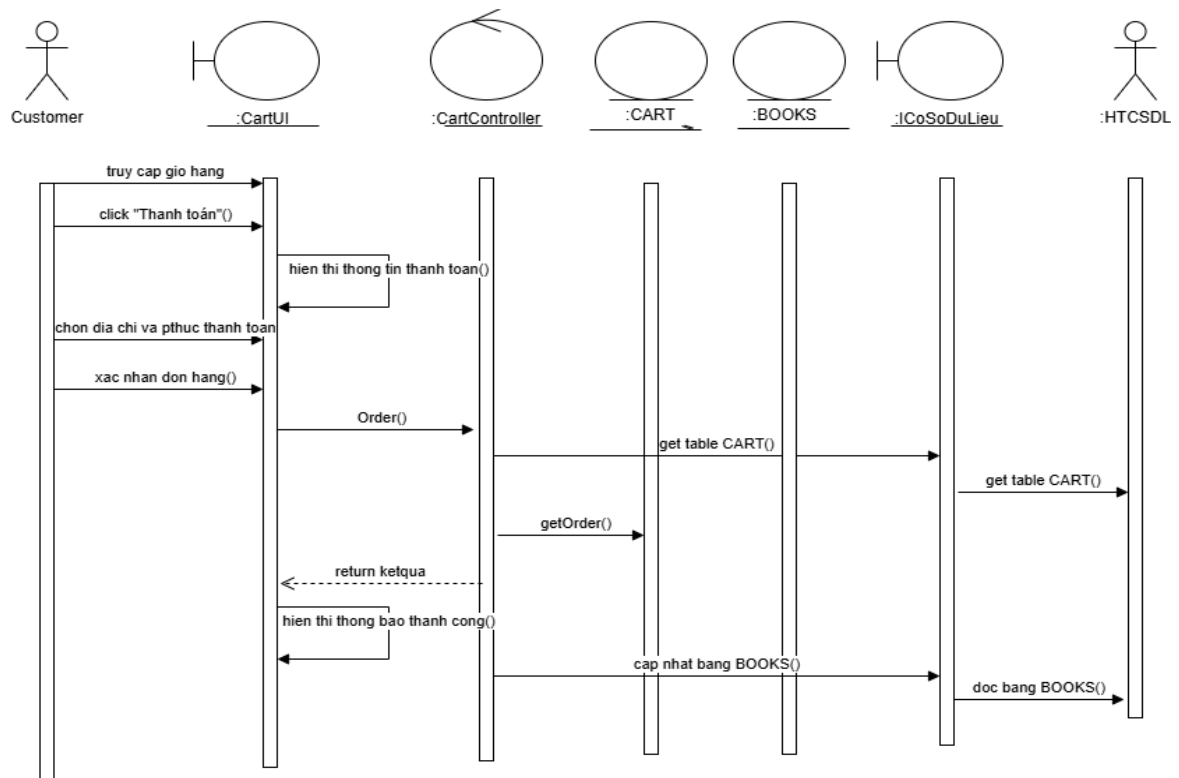
Hình 7: Biểu đồ trình tự xem chi tiết sách



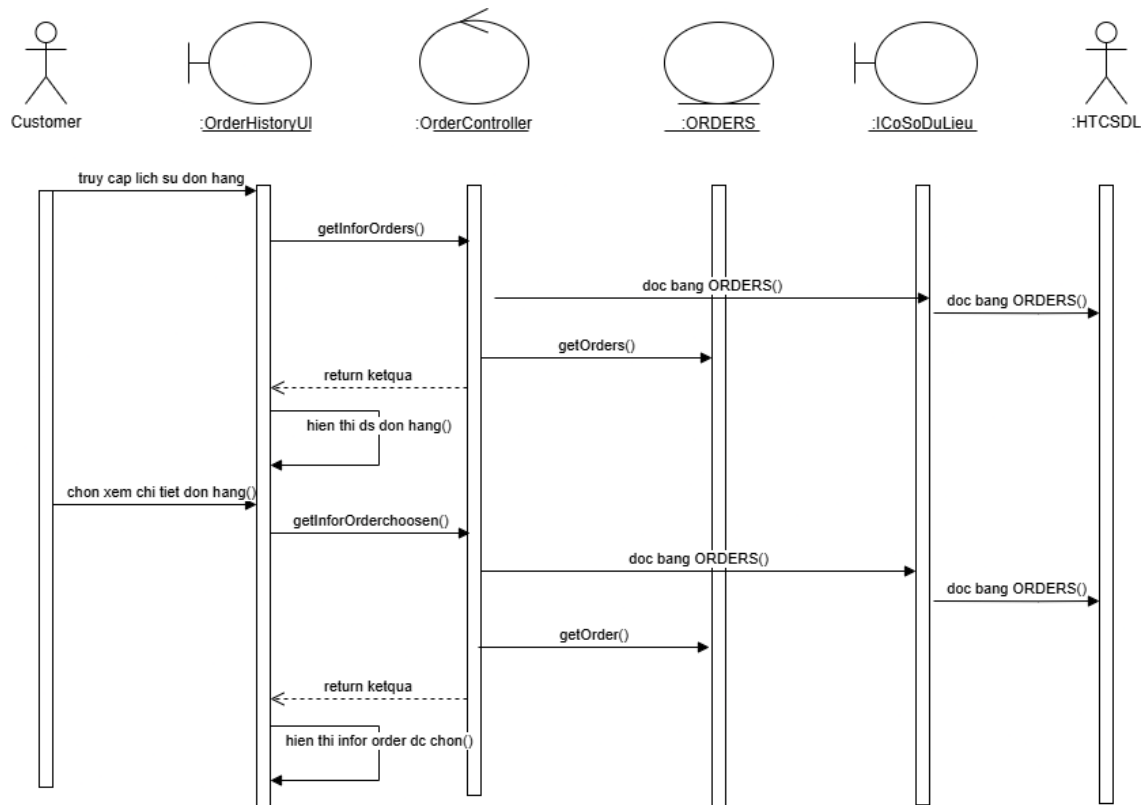
Hình 8: Biểu đồ trình tự thêm vào giỏ hàng



Hình 9: Biểu đồ trình tự cập nhật giỏ hàng

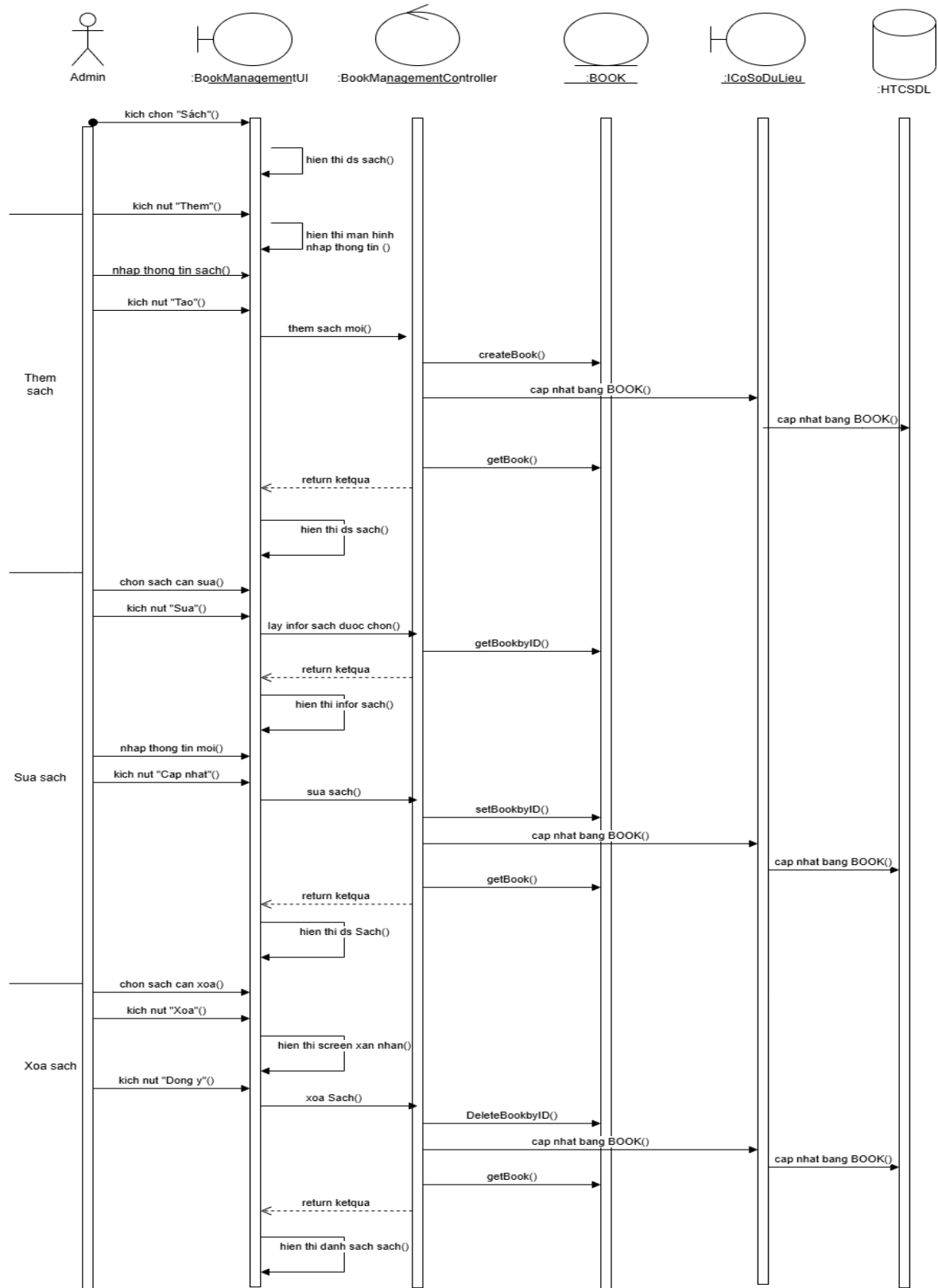


Hình 10: Biểu đồ trình tự đặt hàng

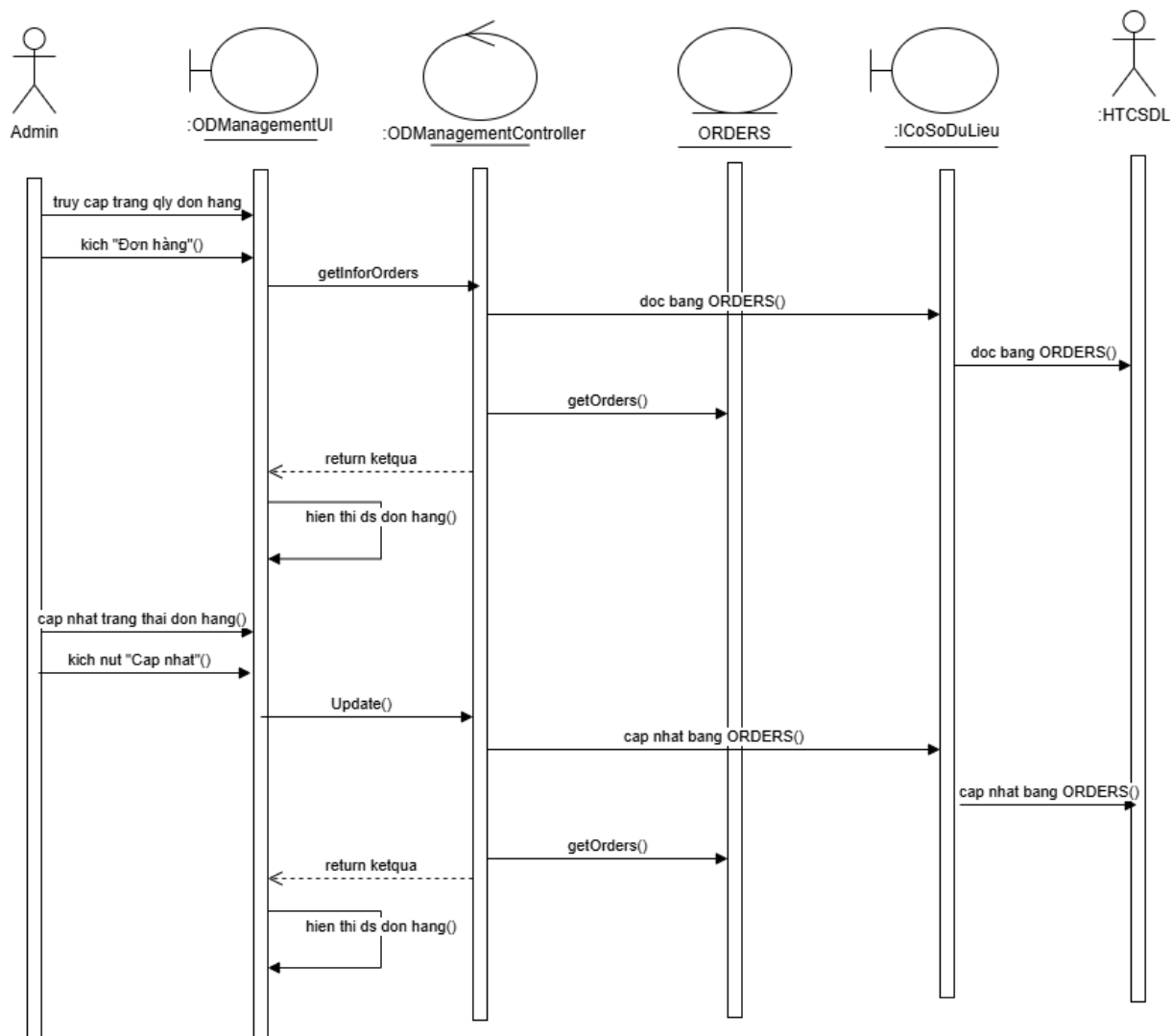


Hình 11: Biểu đồ trình tự xem lịch sử đơn hàng

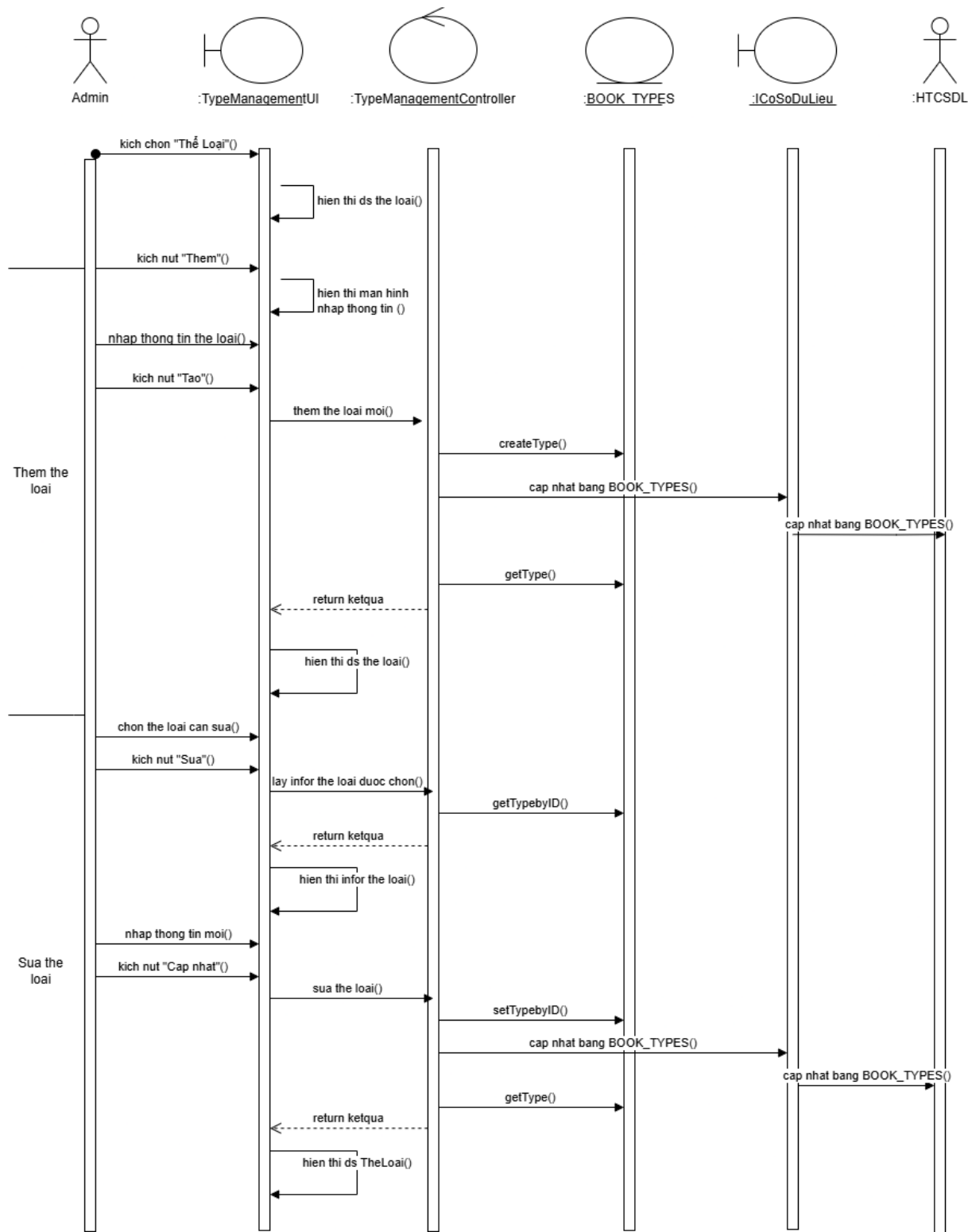
3.3.2. Người quản lý



Hình 12: Biểu đồ trình tự quản lý sách

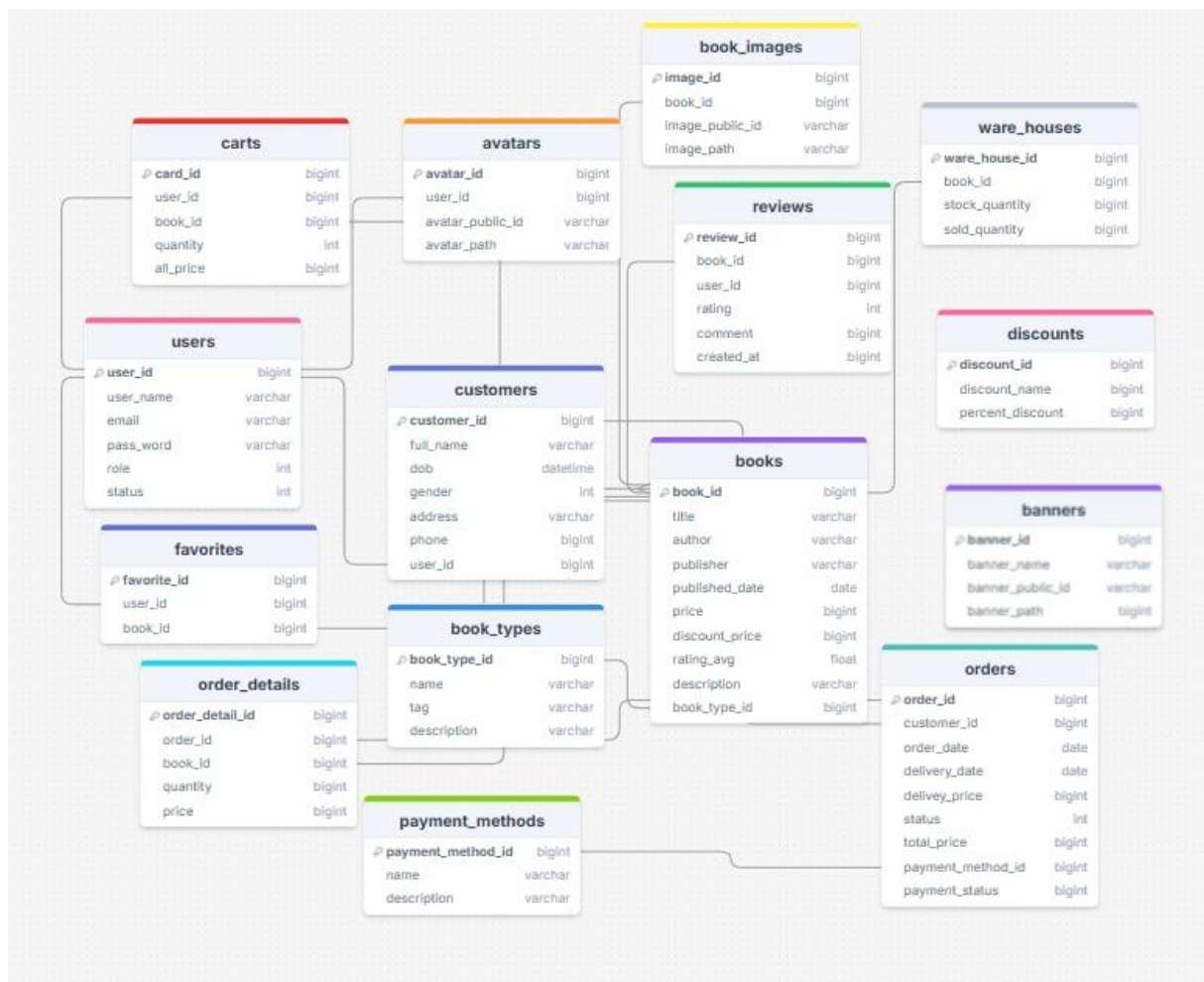


Hình 13: Biểu đồ trình tự quản lý đơn hàng



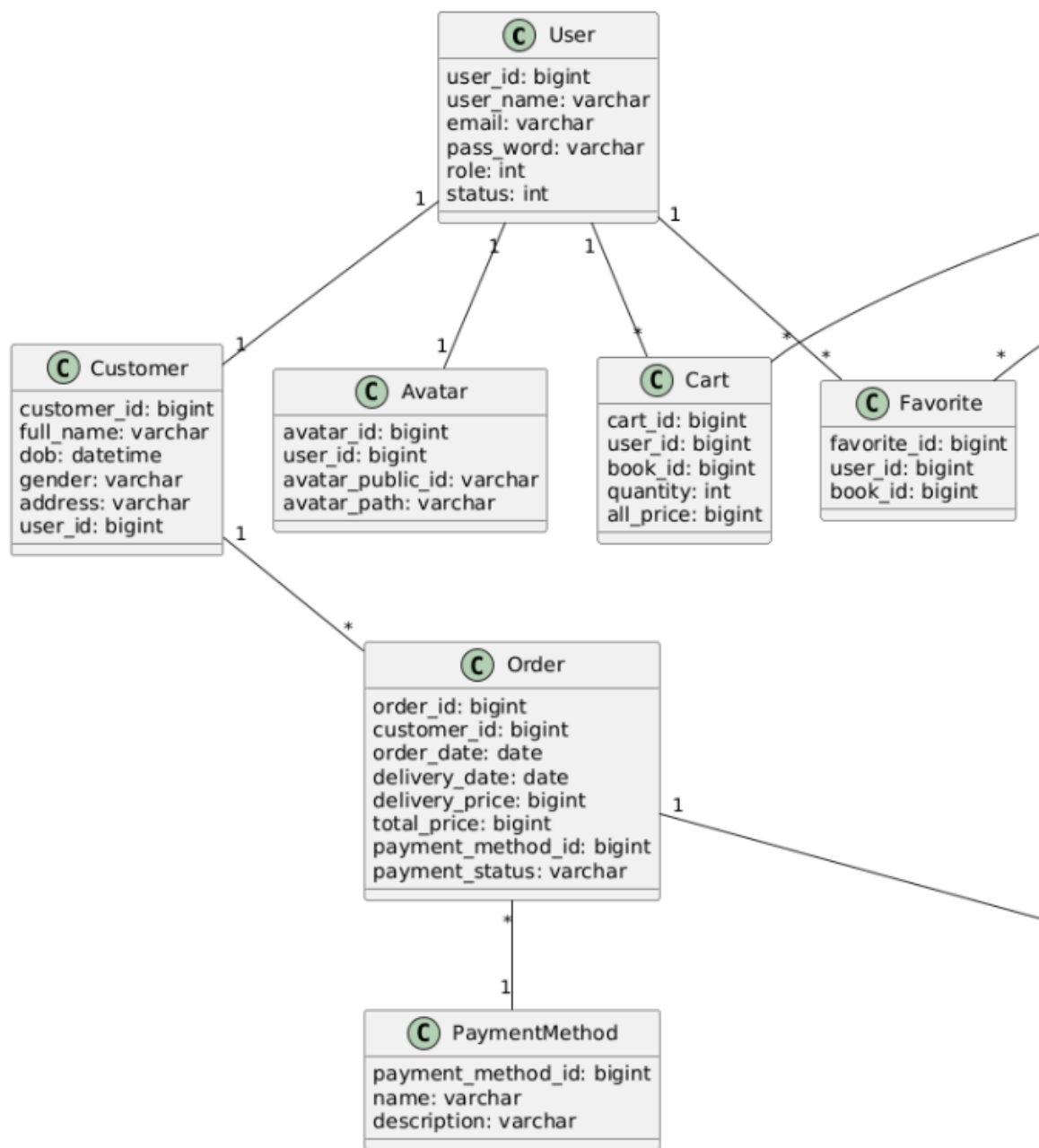
Hình 14: Biểu đồ trình tự quản lý thể loại

3.4. Biểu đồ lớp (Class Diagram)

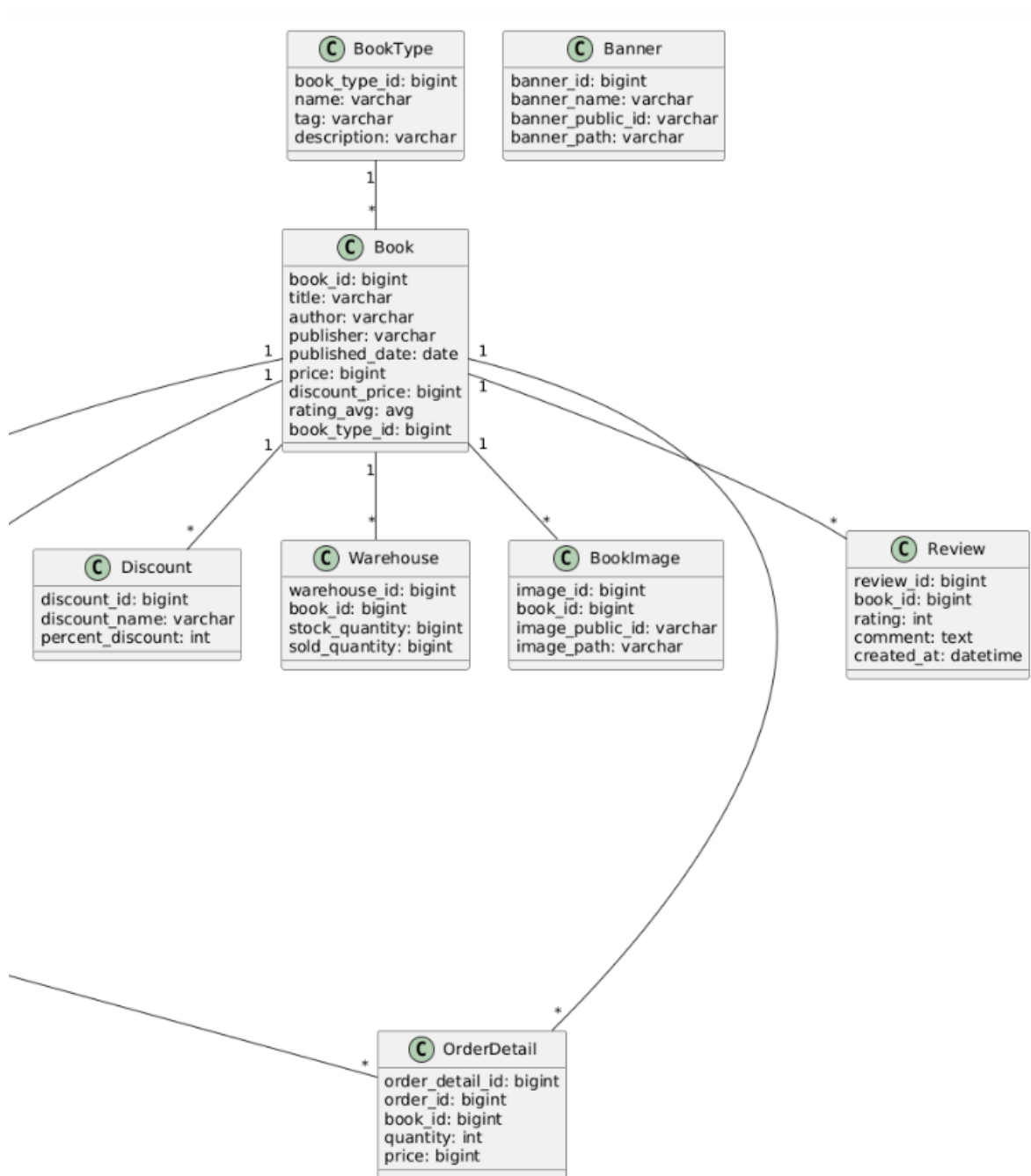


Hình 15: Biểu đồ lớp

3.5. Biểu đồ thực thể - quan hệ (ERD)



Hình 16: Biểu đồ thực thể - quan hệ 1



Hình 17: Biểu đồ thực thể - quan hệ 2

4. THIẾT KẾ HỆ THỐNG

4.1. Kiến trúc tổng quan

Hệ thống được thiết kế theo mô hình client-server, trong đó frontend (Vue.js) đảm nhận giao diện người dùng và backend (Node.js + Express) xử lý dữ liệu và logic

ng nghiệp vụ. Dữ liệu được lưu trữ và quản lý bởi MySQL, đảm bảo tính toàn vẹn và hiệu suất truy vấn cao.

Thành phần chính của hệ thống:

- **Frontend (Vue.js):**

- Giao diện người dùng
- Gửi và nhận dữ liệu từ backend thông qua API
- Xử lý logic hiển thị và xác thực người dùng

- **Backend (Node.js + Express):**

- Cung cấp API để giao tiếp với frontend
- Xử lý logic nghiệp vụ (đăng ký, đăng nhập, đặt hàng, quản lý sách, v.v.)
- Xác thực và phân quyền người dùng bằng JWT
- Kết nối với cơ sở dữ liệu MySQL

- **Cơ sở dữ liệu (MySQL):**

- Lưu trữ dữ liệu người dùng, sách, đơn hàng, giỏ hàng
- Được tối ưu với index cho các truy vấn thường xuyên

- **Kiến trúc API:**

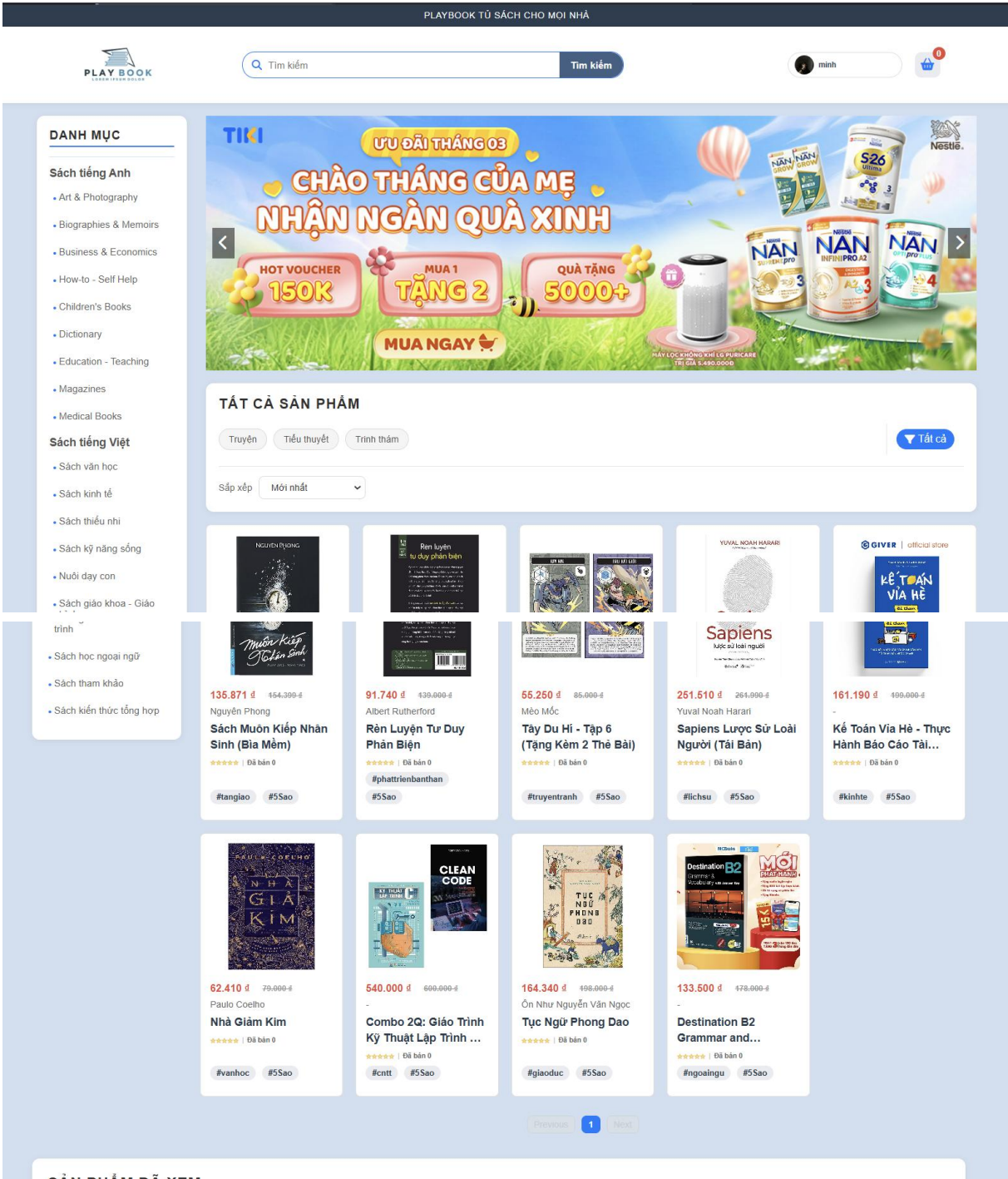
- Giao tiếp giữa frontend và backend qua RESTful API
- API trả về dữ liệu theo định dạng JSON
- Sử dụng Swagger để định nghĩa và quản lý API

- **Bảo mật và hiệu suất:**

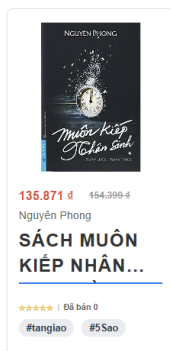
- Sử dụng HTTPS để bảo vệ dữ liệu truyền tải
- Mã hóa mật khẩu trước khi lưu vào database
- Áp dụng caching để giảm tải truy vấn cơ sở dữ liệu
- Giới hạn số lượng request từ client để ngăn chặn tấn công DDoS

4.2. Thiết kế giao diện người dùng

4.2.1. Trang chủ



SẢN PHẨM ĐÃ XEM



Hỗ trợ khách hàng

Hotline: 0987 654 321
Email: dohoan@gmail.com
Gửi yêu cầu hỗ trợ
Hướng dẫn đặt hàng
Chính sách kiểm hàng
Phương thức vận chuyển
Chính sách vận chuyển
Bảo lỗi

Về PlayBook

Giới thiệu về PlayBook
PlayBook Blog
Tuyển dụng
Chính sách bảo mật thanh toán
Chính sách giải quyết khiếu nại
Chính sách bảo mật thông tin các nhân
Điều khoản sử dụng
Tiếp thị liên kết PlayBook
Bán hàng doanh nghiệp

Hợp tác và liên kết

Bán hàng cùng PlayBook

Phương thức thanh toán



Phương thức vận chuyển



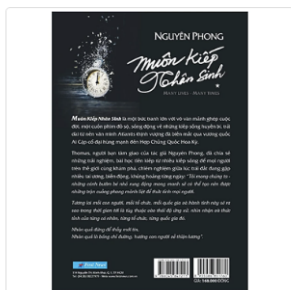
Kết nối với chúng tôi



Hình 18: Trang chủ

4.2.2. Trang chi tiết sản phẩm

Trang chủ > Sách tiếng Việt > Sách Muôn Kiếp Nhân Sinh (Bìa Mềm)



Sách Muôn Kiếp Nhân Sinh (Bìa Mềm)

★★★★★ (0 đánh giá)

135.871 VNĐ -154.399 VNĐ

Số lượng

Mua ngay

Thêm vào giỏ



📍 Hoàn Kiếm, Hà Đông, Hà Nội

🚚 Giao phí chỉ từ 10k cho đơn từ 45k

Thông tin sản phẩm

Phiên bản	Tái bản
Ngày xuất bản	25/3/2025
Tác giả	Nguyễn Phong
Kích thước	16 x 24 cm
Số trang	100000
Công ty phát hành	Nhà Xuất Bản Tổng hợp TP.HCM

Mô tả sản phẩm

"Muôn kiếp nhân sinh"

[Xem thêm](#)

Đặc điểm nổi bật

- Tác phẩm kinh điển của Nguyễn Phong
- Ngôn ngữ sắc sảo, phản ánh xã hội sâu sắc
- Ấn bản mới từ Nhà Xuất Bản Tổng hợp TP.HCM

Khách hàng đánh giá

Mô tả sản phẩm
"Muốn kiếp nhân sinh"
[Xem thêm](#)

Khách hàng đánh giá

Previous

Next

Thêm đánh giá của bạn
Đánh giá (1-5 sao):

5 sao

▼

Bình luận:

Viết đánh giá của bạn...

Gửi đánh giá

Hình 19: Trang chi tiết sản phẩm

4.2.3. Giỏ hàng

Giỏ hàng

✓	STT	Hình ảnh	Tên sách	Giá tiền	Số lượng	Hành động
✓	1		Rèn Luyện Tư Duy Phản Biện	91.740 đ	- 1 +	
✓	2		Sách Muốn Kiếp Nhân Sinh (Bìa Mềm)	308.798 đ	- 2 +	

Giá tổng
Không có thông tin. Vui lòng cập nhật !!
Khuyến mãi
Có thể chọn 1

Giảm 10% (Giảm 10%)

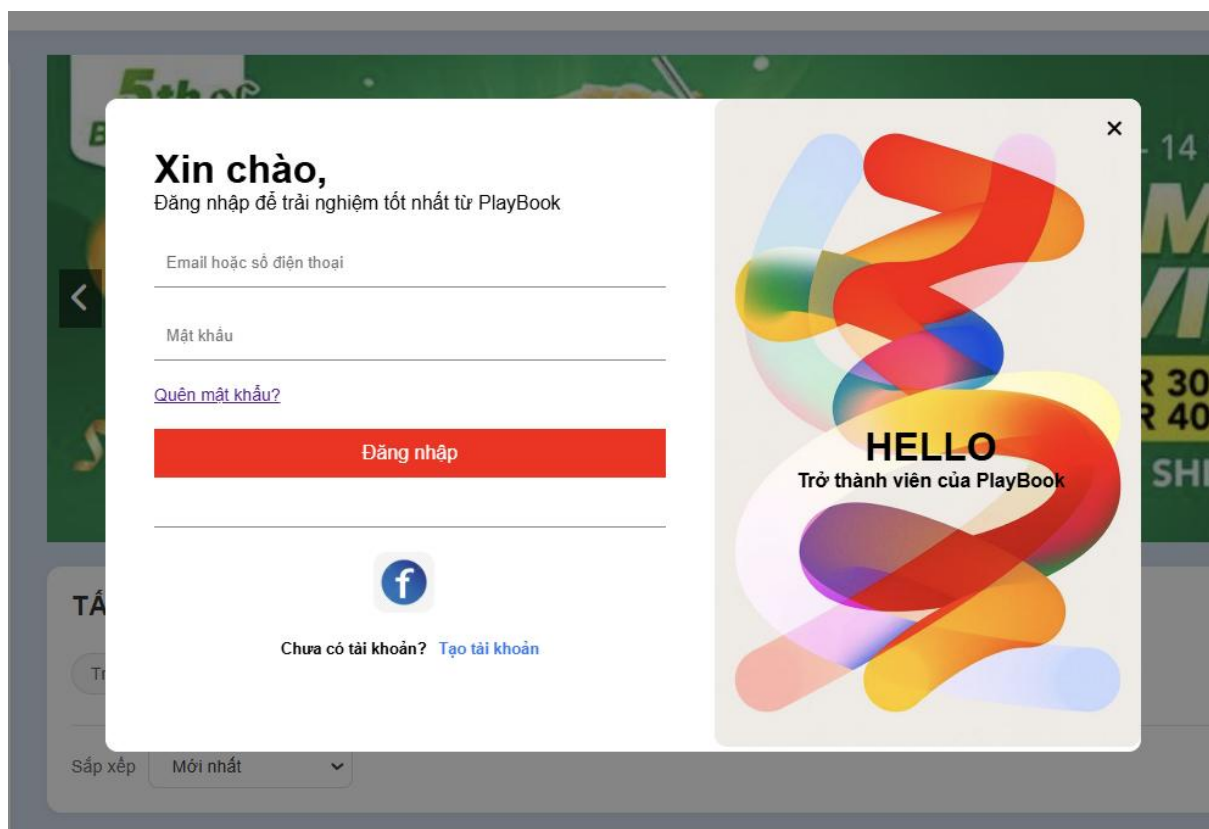
Phương thức thanh toán
☐ MoMo
☒ Thẻ ngân hàng
☐ Thanh toán sau khi giao hàng

Tạm tính: **400.538 đ**
Phí vận chuyển: **50.000 đ**
Giảm giá: **40.054 đ**
Tổng tiền thanh toán: 410.484 đ

THANH TOÁN

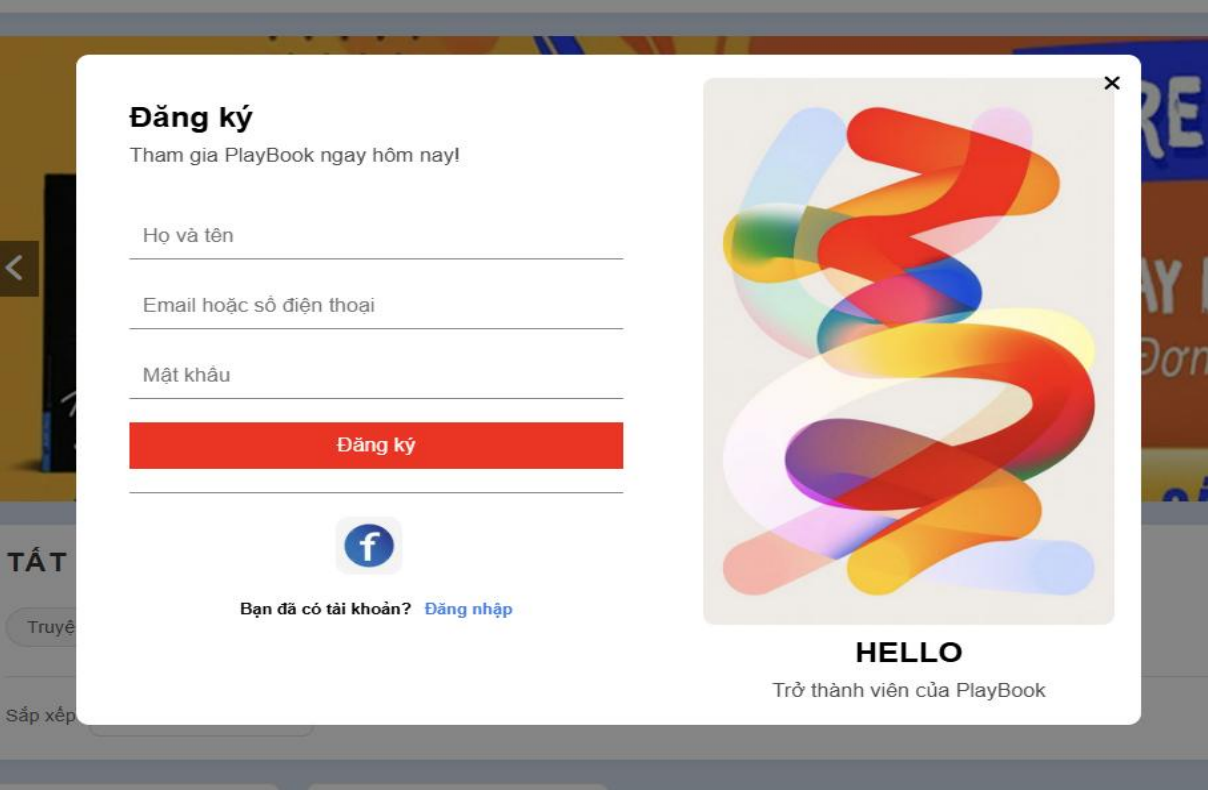
Hình 20: Trang giỏ hàng

4.2.4. Đăng nhập



Hình 21: Popup đăng nhập

4.2.5. Đăng ký




Đăng ký
Tham gia PlayBook ngay hôm nay!

Họ và tên

Email hoặc số điện thoại

Mật khẩu

Đăng ký

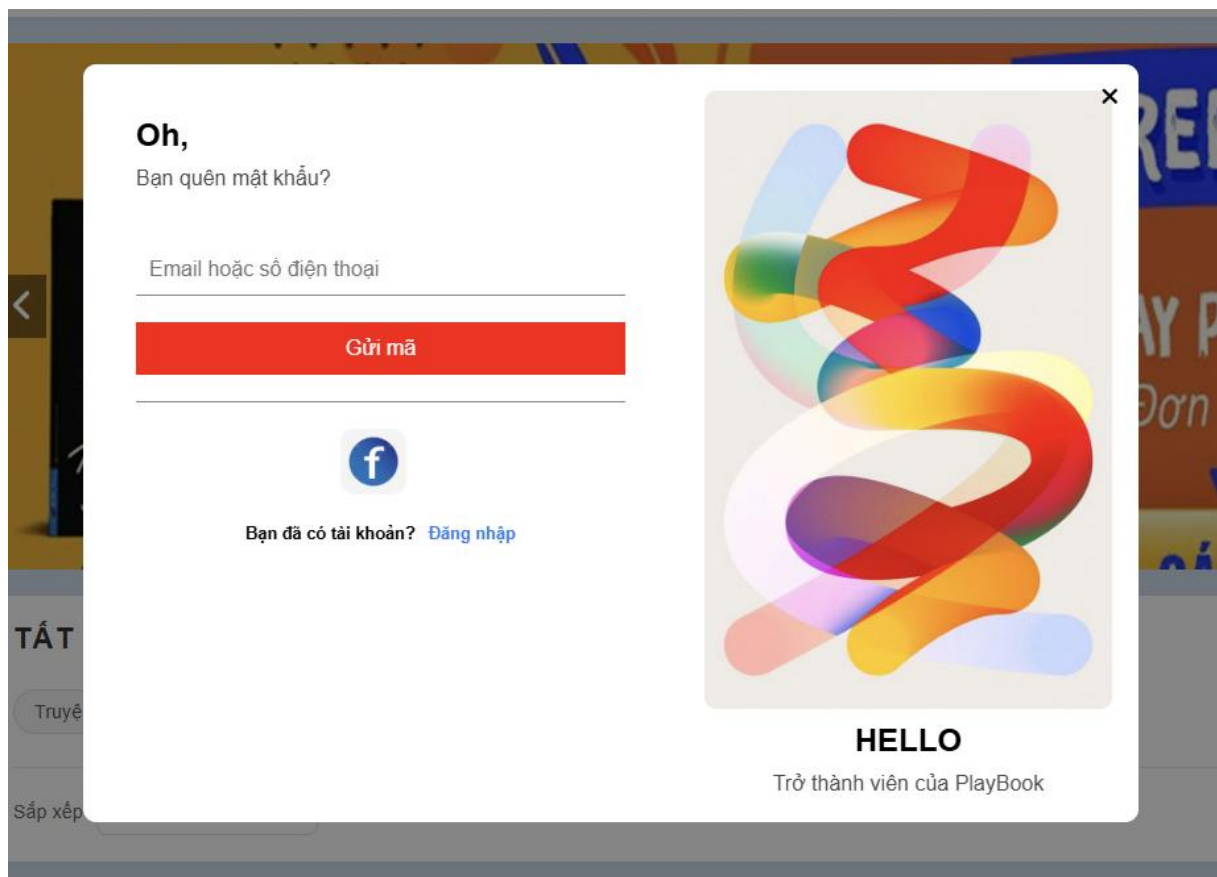


Bạn đã có tài khoản? [Đăng nhập](#)

HELLO
Trở thành viên của PlayBook

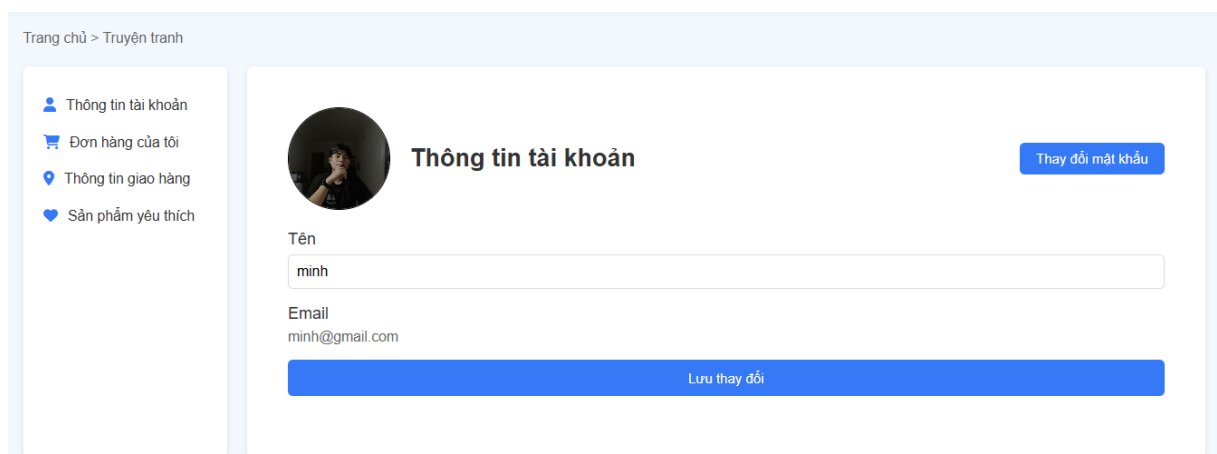
Hình 22: Popup đăng ký

4.2.6. Quên mật khẩu



Hình 23: Popup quên mật khẩu

4.2.7. Trang người dùng



Hình 24: Trang thông tin người dùng

Trang chủ > Truyền tranh

- Thông tin tài khoản
- Đơn hàng của tôi
- Thông tin giao hàng
- Sản phẩm yêu thích

Thông tin giao hàng của bạn

Thêm thông tin giao hàng

Thông tin giao hàng hiện tại

Họ và tên: hoan
 Ngày sinh: 13/03/2025
 Giới tính: Nam
 Thông tin: Ba Đình, Hà Nội
 Số điện thoại: 90219230

Sửa Xóa

Hình 25: Địa chỉ người dùng

Trang chủ > Truyền tranh

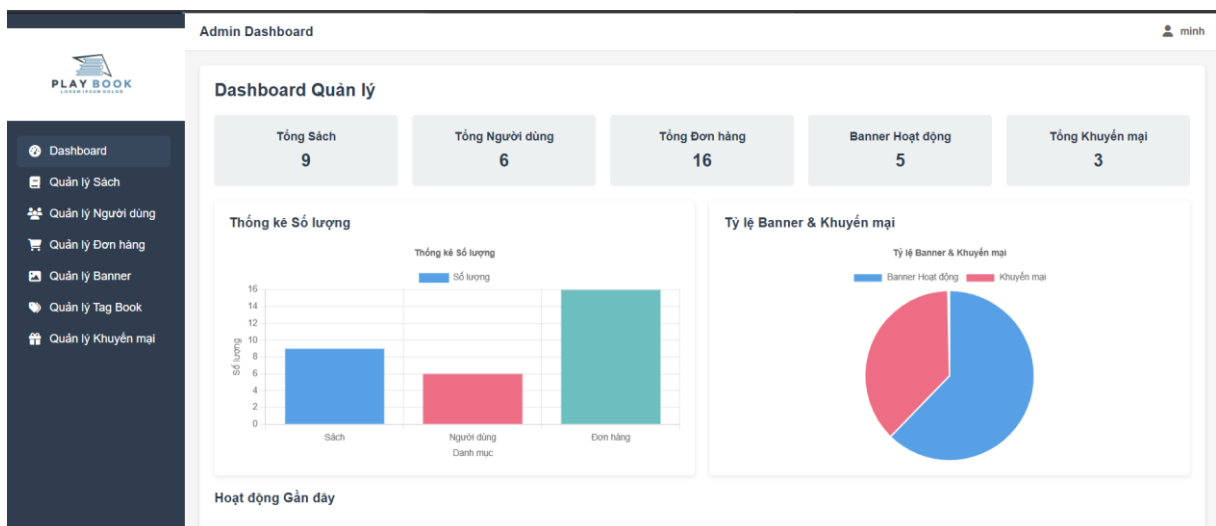
- Thông tin tài khoản
- Đơn hàng của tôi
- Thông tin giao hàng
- Sản phẩm yêu thích

Sản phẩm yêu thích

STT	Hình ảnh	Tên sản phẩm	Giá tiền	Hành động
1		Rèn Luyện Tư Duy Phản Biện	139.000 VNĐ	Xóa
2		Sách Muốn Kiếp Nhân Sinh (Bia Mềm)	154.399 VNĐ	Xóa
3		Combo 2Q: Giáo Trình Kỹ Thuật Lập Trình C Căn Bản Và Nâng Cao + Clean Code - Mã Sạch Và Con Đường Trở Thành Lập Trình Viên Giỏi (Tặng Mốc Khóa)	600.000 VNĐ	Xóa

Hình 26: Trang sản phẩm yêu thích của người dùng

4.2.8. Trang admin

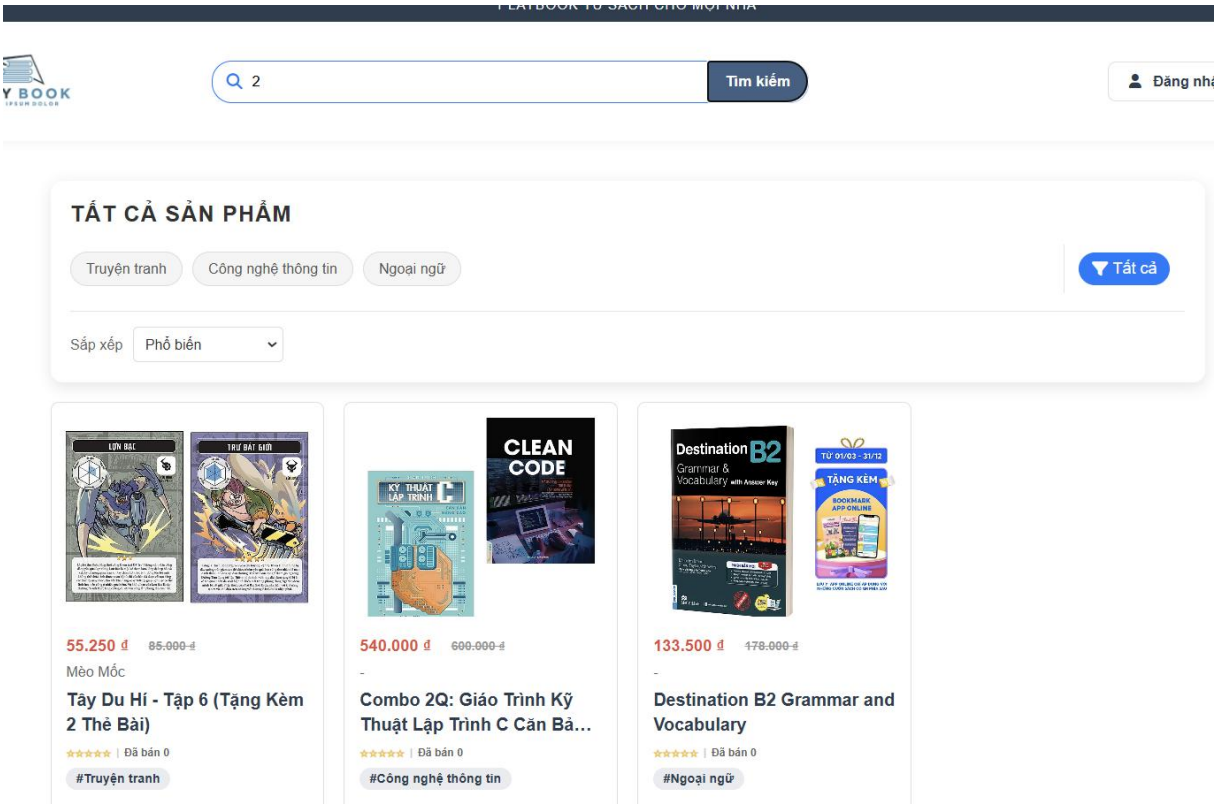


Hình 27: Trang Admin Dashboard

4.2.9. Tìm kiếm



Hình 34: Khung tìm kiếm gợi ý



Hình 35: Kết quả sau tìm kiếm

4.2.10. Bộ lọc

Tất cả bộ lọc

X

Dịch vụ

☐ Giao hàng miễn phí

Đánh giá

☐ ★ 1 Sao

☐ ★ ★ 2 Sao

☐ ★ ★ ★ 3 Sao

☐ ★ ★ ★ ★ 4 Sao

☐ ★ ★ ★ ★ ★ 5 Sao

Giá

☐ Dưới 100.000đ

☐ 100.000đ - 200.000đ

☐ 200.000đ - 500.000đ

☐ 500.000đ - 1.000.000đ

☐ 1.000.000đ - 2.000.000đ

☐ Trên 2.000.000đ

Tự nhập khoảng giá

Từ

Đến

Xóa

Xóa tất cả

Xem kết quả

Hình 36: Bộ lọc sản phẩm

4.3. Thiết kế cơ sở dữ liệu

Hệ thống sử dụng cơ sở dữ liệu quan hệ MySQL để lưu trữ và quản lý dữ liệu. Cơ sở dữ liệu được thiết kế theo mô hình chuẩn hóa, đảm bảo tính toàn vẹn dữ liệu và tối ưu hiệu suất truy vấn.

Các bảng chính trong cơ sở dữ liệu:

- **Bảng users:** Lưu trữ thông tin người dùng, bao gồm tên đăng nhập, email, mật khẩu đã mã hóa và vai trò (admin hoặc khách hàng).
- **Bảng books:** Lưu trữ thông tin sách như tiêu đề, tác giả, giá, số lượng tồn kho, hình ảnh và mô tả.
- **Bảng orders:** Lưu trữ đơn hàng của người dùng, bao gồm thông tin khách hàng, tổng giá trị đơn hàng, trạng thái đơn hàng và phương thức thanh toán.
- **Bảng order_details:** Liên kết giữa đơn hàng và sách, giúp theo dõi số lượng sách trong từng đơn hàng.
- **Bảng favorites:** Lưu trữ danh sách sách yêu thích của từng người dùng.
- **Bảng reviews:** Lưu trữ đánh giá của khách hàng về sách, bao gồm điểm số và bình luận.
- **Bảng carts:** Lưu trữ giỏ hàng của người dùng, giúp theo dõi những sản phẩm họ đang có ý định mua.
- **Bảng warehouses:** Quản lý số lượng sách trong kho, bao gồm số lượng tồn và số lượng đã bán.
- **Bảng discounts:** Lưu thông tin các chương trình giảm giá, giúp điều chỉnh giá bán sách.
- **Bảng payment_methods:** Lưu thông tin về các phương thức thanh toán như thẻ tín dụng, ví điện tử hoặc tiền mặt.

Ràng buộc và quan hệ giữa các bảng:

- Một người dùng có thể có nhiều đơn hàng → (Quan hệ 1-N giữa users và orders).
- Một đơn hàng có thể chứa nhiều sách, và một sách có thể thuộc nhiều đơn hàng → (Quan hệ N-N giữa orders và books, sử dụng bảng order_details).
- Một sách có thể có nhiều đánh giá từ nhiều người dùng → (Quan hệ 1-N giữa books và reviews).
- Một người dùng có thể thêm nhiều sách yêu thích → (Quan hệ 1-N giữa users và favorites).
- Một sách có thể có nhiều hình ảnh minh họa → (Quan hệ 1-N giữa books và book_images).
- Một sách có thể thuộc nhiều kho hàng khác nhau → (Quan hệ 1-N giữa books và warehouses).

4.4. Thiết kế API backend

Hệ thống API backend được xây dựng theo kiến trúc RESTful, đảm bảo tính rõ ràng, dễ mở rộng và bảo trì. Các API này cung cấp chức năng quản lý tài khoản, sách, giỏ hàng, đơn hàng, thanh toán và nhiều tính năng khác cho hệ thống bán sách trực tuyến.

Tuân thủ nguyên tắc thiết kế RESTful bằng cách sử dụng các phương thức HTTP phù hợp như GET, POST, PUT, DELETE. API được thiết kế để đảm bảo tính bảo mật thông qua xác thực bằng JWT (JSON Web Token) và cơ chế phân quyền để kiểm soát truy cập.

Danh sách API chính:

- **API Quản lý người dùng (Users):**
 - POST /api/auth/register → Đăng ký người dùng
 - POST /api/auth/login → Đăng nhập và nhận JWT token
 - GET /api/users/profile → Lấy thông tin cá nhân (yêu cầu xác thực)
 - PUT /api/users/profile → Cập nhật thông tin cá nhân

- GET /api/users/:id → Lấy thông tin người dùng theo ID (chỉ admin)
- **API Quản lý sách (Books):**
 - GET /api/books → Lấy danh sách tất cả sách
 - GET /api/books/:id → Lấy thông tin sách theo ID
 - POST /api/books → Thêm sách mới (chỉ admin)
 - PUT /api/books/:id → Cập nhật thông tin sách (chỉ admin)
 - DELETE /api/books/:id → Xóa sách (chỉ admin)
- **API Quản lý loại sách (Book Categories)**
 - GET /api/booktypes → Lấy danh sách tất cả loại sách
 - POST /api/booktypes → Thêm loại sách mới (chỉ admin)
 - PUT /api/booktypes/:id → Cập nhật loại sách (chỉ admin)
 - DELETE /api/booktypes/:id → Xóa loại sách (chỉ admin)
- **API Quản lý giỏ hàng (Cart):**
 - GET /api/cart → Lấy giỏ hàng của người dùng (yêu cầu xác thực)
 - POST /api/cart → Thêm sách vào giỏ hàng
 - PUT /api/cart/:id → Cập nhật số lượng sách trong giỏ hàng
 - DELETE /api/cart/:id → Xóa sách khỏi giỏ hàng
- **API Quản lý đơn hàng (Orders):**
 - GET /api/orders → Lấy danh sách đơn hàng (chỉ admin)
 - GET /api/orders/user → Lấy danh sách đơn hàng của người dùng (yêu cầu xác thực)
 - POST /api/orders → Tạo đơn hàng mới
 - PUT /api/orders/:id → Cập nhật trạng thái đơn hàng (chỉ admin)
 - DELETE /api/orders/:id → Hủy đơn hàng
- **API Quản lý thanh toán (Payments):**
 - POST /api/payment → Xử lý thanh toán đơn hàng
- **API Quản lý đánh giá sách (Reviews):**
 - POST /api/reviews → Thêm đánh giá cho sách
 - DELETE /api/reviews/:id → Xóa đánh giá (chỉ người dùng tạo đánh giá hoặc admin)

- **API Quản lý danh sách yêu thích (Favorites):**
 - GET /api/favorites → Lấy danh sách sách yêu thích của người dùng
 - POST /api/favorites → Thêm sách vào danh sách yêu thích
 - DELETE /api/favorites/:id → Xóa sách khỏi danh sách yêu thích
- **API Quản lý banner (Banners):**
 - GET /api/banners → Lấy danh sách banner
 - POST /api/banners → Thêm banner mới (chỉ admin)
 - DELETE /api/banners/:id → Xóa banner (chỉ admin)
- **API Quản lý mã giảm giá (Discounts):**
 - GET /api/discounts → Lấy danh sách mã giảm giá
 - POST /api/discounts → Thêm mã giảm giá (chỉ admin)
 - PUT /api/discounts/:id → Cập nhật mã giảm giá (chỉ admin)
 - DELETE /api/discounts/:id → Xóa mã giảm giá (chỉ admin)

Tất cả API yêu cầu dữ liệu gửi và nhận dưới dạng **JSON**, đồng thời các API quan trọng cần **JWT token** để xác thực và kiểm soát quyền truy cập.

5. CÀI ĐẶT VÀ TRIỂN KHAI

5.1. Công nghệ sử dụng

Hệ thống "PlayBook" được phát triển dựa trên các công nghệ hiện đại và phổ biến, đảm bảo tính ổn định và khả năng mở rộng:

Frontend: Vue.js kết hợp Vuex để quản lý state và Vuetify để thiết kế giao diện.

Backend: Node.js với framework Express.js, hỗ trợ RESTful API.

Database: MySQL cho dữ liệu có cấu trúc (sách, người dùng, đơn hàng).

Công cụ hỗ trợ: Postman để kiểm thử API, Git để quản lý phiên bản.

5.2. Triển khai backend với Node.js

Quy trình triển khai:

1. Khởi tạo dự án: Sử dụng `npm init` để tạo cấu trúc thư mục.
2. Cài đặt dependencies:

```
npm install express mysql2 bcryptjs cors dotenv
```

3. Xây dựng API:

- Tạo các endpoint cho quản lý sách, người dùng, đơn hàng (ví dụ: /api/books, /api/users).
- Kết nối MySQL bằng thư viện mysql2, sử dụng connection pool để tối ưu hiệu suất.
- Mã hóa mật khẩu người dùng bằng bcryptjs.

4. Bảo mật:

- Sử dụng JWT (JSON Web Token) để xác thực người dùng.
- Triển khai middleware kiểm tra quyền truy cập cho các route admin.

5. Triển khai lên server:

- Build ứng dụng với `npm run build`.

5.3. Triển khai frontend với Vue.js

Quy trình triển khai:

1. Khởi tạo dự án:

```
vue create playbook-frontend
```

2. Cài đặt thư viện:

```
npm install vuex axios vuetify vue-router
```

3. Xây dựng component:

- Thiết kế trang chủ, trang sản phẩm, giỏ hàng bằng Vuetify.
- Tích hợp Vue Router để điều hướng giữa các trang.

4. Kết nối API:

- Sử dụng Axios để gọi API từ backend.
- Quản lý state toàn cục với Vuex (lưu trữ thông tin người dùng, giỏ hàng).

5. Tối ưu hóa:

- Áp dụng lazy loading cho hình ảnh.
- Triển khai SSR (Server-Side Rendering) bằng Nuxt.js để cải thiện SEO.

6. Deploy:

- Build ứng dụng với `npm run serve`.

5.4. Quản lý cơ sở dữ liệu với MySQL

Hệ thống sử dụng MySQL để quản lý dữ liệu với các bảng chính:

- **users:** Lưu thông tin người dùng, bao gồm user_id, user_name, email, pass_word, role, và status.
- **customers:** Mở rộng từ bảng users, lưu thông tin khách hàng như full_name, dob, gender, address, và user_id.
- **books:** Chứa thông tin về sách, gồm book_id, title, author, publisher, published_date, price, discount_price, rating_avg, và book_type_id.
- **book_types:** Xác định danh mục sách với các trường book_type_id, name, tag, và description.
- **orders:** Lưu trữ thông tin đơn hàng, bao gồm order_id, customer_id, order_date, delivery_date, delivery_price, total_price, payment_method_id, và payment_status.
- **order_details:** Liên kết giữa đơn hàng và sách với các trường order_detail_id, order_id, book_id, quantity, và price.
- **carts:** Lưu giỏ hàng của người dùng, gồm cart_id, user_id, book_id, quantity, và all_price.
- **favorites:** Quản lý danh sách yêu thích với favorite_id, user_id, và book_id.
- **reviews:** Chứa đánh giá của khách hàng về sách, gồm review_id, book_id, rating, comment, và created_at.
- **discounts:** Lưu thông tin giảm giá với discount_id, discount_name, và percent_discount.
- **warehouses:** Quản lý kho sách với warehouse_id, book_id, stock_quantity, và sold_quantity.
- **payment_methods:** Lưu thông tin phương thức thanh toán với payment_method_id, name, và description.
- **banners:** Quản lý quảng cáo với banner_id, banner_name, banner_public_id, và banner_path.

- **book_images:** Lưu trữ hình ảnh sách với `image_id`, `book_id`, `image_public_id`, và `image_path`.
- **avatars:** Quản lý ảnh đại diện người dùng với `avatar_id`, `user_id`, `avatar_public_id`, và `avatar_path`.

Sử dụng **MySQL Workbench** để thiết kế sơ đồ cơ sở dữ liệu, đảm bảo mối quan hệ giữa các bảng và tạo **chỉ mục (indexes)** trên các trường tìm kiếm quan trọng như `title`, `author` trong bảng `books`.

5.5. Kết nối API giữa frontend và backend

Trước tiên, cần xác định các API cần thiết để giao tiếp giữa frontend (Vue.js) và backend (Node.js). Sử dụng OpenAPI (Swagger) để mô tả chi tiết các endpoint, bao gồm URL, phương thức HTTP (GET, POST, PUT, DELETE), tham số đầu vào và cấu trúc dữ liệu trả về.

CORS (Cross-Origin Resource Sharing) cần được cấu hình để cho phép frontend truy cập backend. Trong Node.js, middleware CORS được sử dụng để cấp quyền truy cập từ domain frontend.

Xác thực và phân quyền là một phần quan trọng trong kết nối API. Khi người dùng đăng nhập, frontend gửi yêu cầu và nhận JWT Token từ backend. Mỗi khi gửi request, frontend đính kèm token vào header `Authorization`. Backend sẽ kiểm tra và xác thực token trong mỗi request để đảm bảo quyền truy cập hợp lệ.

Gửi và nhận dữ liệu giữa frontend và backend phải tuân theo các phương thức HTTP phù hợp. GET được dùng để lấy dữ liệu, POST để gửi dữ liệu mới, PUT để cập nhật dữ liệu và DELETE để xóa. Backend xử lý dữ liệu, truy vấn vào database và phản hồi kết quả về frontend theo định dạng JSON thống nhất, ví dụ:

Xử lý lỗi API cần rõ ràng để đảm bảo trải nghiệm người dùng. Backend phải trả về status code phù hợp như 200 OK (thành công), 400 Bad Request (yêu cầu không hợp lệ), 401 Unauthorized (không có quyền truy cập), 404 Not Found (không tìm thấy dữ liệu) và 500 Internal Server Error (lỗi hệ thống). Frontend cần hiển thị thông báo lỗi phù hợp để người dùng hiểu nguyên nhân sự cố.

Bảo mật API cần được đảm bảo bằng cách mã hóa mật khẩu trước khi lưu vào database, sử dụng HTTPS để bảo vệ dữ liệu khi truyền qua mạng và giới hạn số lần gọi API để tránh bị tấn công DDoS.

Tối ưu hiệu suất API là yếu tố quan trọng giúp cải thiện tốc độ phản hồi. Có thể sử dụng caching (Redis, Memory Cache) để giảm tải cho server, tạo index trong database để tăng tốc truy vấn và giảm số lượng request bằng cách gộp API hoặc tải dữ liệu theo trang (pagination).

Việc kết nối API giữa Vue.js và Node.js cần được thực hiện chặt chẽ để đảm bảo tính bảo mật, hiệu suất và trải nghiệm người dùng tốt nhất.

6. KIỂM THỬ VÀ ĐÁNH GIÁ

6.1. Phương pháp kiểm thử

Kiểm thử đơn vị (Unit Test): Sử dụng Jest để kiểm tra từng hàm nghiệp vụ (ví dụ: tính tổng giá đơn hàng).

Kiểm thử tích hợp (Integration Test): Kiểm tra luồng đặt hàng từ frontend đến backend.

Kiểm thử hiệu năng: Sử dụng JMeter để mô phỏng 1000 người dùng truy cập đồng thời.

Kiểm thử bảo mật: Quét lỗ hổng bằng OWASP ZAP.

6.2. Kết quả kiểm thử

Loại kiểm thử	Kết quả
Unit Test	Đạt 85% coverage. Lỗi phát hiện ở hàm tính giảm giá. Đã sửa.
Integration Test	Đơn hàng được xử lý thành công, nhưng delay 3s khi thanh toán.
Hiệu năng	Thời gian phản hồi trung bình: 1.2s. Hệ thống chịu tải 800 user đồng thời.

Bảo mật	Phát hiện lỗi XSS nhỏ ở form bình luận. Đã khắc phục bằng sanitize input
----------------	--

6.3. Đánh giá hiệu suất hệ thống

Ưu điểm:

- Giao diện tải nhanh (<2s) nhờ tối ưu hình ảnh và CDN.
- API xử lý ổn định dưới áp lực 500 request/giây.

Nhược điểm:

- Thời gian xử lý thanh toán còn chậm do chưa tối ưu transaction database.
- Chưa hỗ trợ cache cho các truy vấn thường xuyên (ví dụ: danh sách sách bán chạy).

6.4. Hạn chế và hướng phát triển

Hạn chế:

- Chưa tích hợp cổng thanh toán trực tuyến.
- Hệ thống đề xuất sách còn đơn giản, chưa áp dụng AI.

Hướng phát triển:

- Xây dựng recommendation engine dựa trên lịch sử mua hàng.
- Triển khai Redis để cache dữ liệu và giảm tải database.
- Mở rộng sang mobile app với React Native.

KẾT LUẬN

Báo cáo này đã trình bày quá trình phân tích và thiết kế hệ thống web bán sách, từ việc xác định yêu cầu, mô hình hóa dữ liệu đến thiết kế hệ thống. Thông qua các phương pháp như Use Case Diagram, ERD, Class Diagram, Sequence Diagram, hệ thống đã được xây dựng với một kiến trúc rõ ràng, đảm bảo tính logic và khả năng mở rộng trong tương lai.

Việc ứng dụng các công nghệ Vue.js, Node.js và MySQL giúp hệ thống hoạt động hiệu quả, hỗ trợ tốt quá trình quản lý và mua bán sách trực tuyến. Ngoài ra, các tính năng như quản lý người dùng, giỏ hàng, đơn hàng và thanh toán được thiết kế một cách tối ưu, đáp ứng nhu cầu của cả người mua và người bán.

Tuy nhiên, trong quá trình phát triển, hệ thống có thể gặp phải một số thách thức như tối ưu hiệu suất, bảo mật dữ liệu và khả năng mở rộng. Do đó, việc tiếp tục nghiên cứu, cải tiến và bổ sung các tính năng mới sẽ giúp hệ thống ngày càng hoàn thiện hơn.

Báo cáo này không chỉ cung cấp một giải pháp cụ thể cho hệ thống web bán sách mà còn là một ví dụ thực tế về việc ứng dụng các nguyên tắc phân tích và thiết kế phần mềm vào phát triển hệ thống. Những phương pháp và mô hình được sử dụng trong báo cáo có thể áp dụng cho nhiều dự án phần mềm khác, góp phần nâng cao chất lượng và hiệu quả trong lĩnh vực phát triển ứng dụng web.

TÀI LIỆU THAM KHẢO

- [1]. Paul Halliday, Full-Stack Web Development with Vue.js and Node.js, Nhà xuất bản Packt Publishing.
- [2]. Mario Casciaro, Node.js Design Patterns, Nhà xuất bản Packt Publishing
- [3]. Fabrizio Romano, Building Progressive Web Applications with Vue.js, Nhà xuất bản Packt Publishing.
- [4]. Evan Hahn, Express in Action: Writing, building, and testing Node.js applications, Nhà xuất bản Manning Publications.
- [5]. Nguyễn Đức Việt, Học Node.js & MySQL, Toidicode.com.