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, 02/2024** | | |

**MỤC LỤC**

[**1. MỞ ĐẦU** 4](#_Toc190406555)

[**1.1. Lý do chọn đề tài** 4](#_Toc190406556)

[**1.2. Mục tiêu nghiên cứu** 4](#_Toc190406557)

[**1.3. Phạm vi và đối tượng nghiên cứu** 4](#_Toc190406558)

[**1.4. Phương pháp nghiên cứu** 4](#_Toc190406559)

[**1.5. Cấu trúc báo cáo** 4](#_Toc190406560)

[**2. KHẢO SÁT VÀ PHÂN TÍCH HỆ THỐNG** 4](#_Toc190406561)

[**2.1. Giới thiệu về hệ thống web bán sách** 4](#_Toc190406562)

[**2.2. Khảo sát các hệ thống tương tự** 4](#_Toc190406563)

[**2.3. Xác định yêu cầu hệ thống** 4](#_Toc190406564)

[**2.3.1. Yêu cầu chức năng** 4](#_Toc190406565)

[**2.3.2. Yêu cầu phi chức năng** 4](#_Toc190406566)

[**2.4. Mô hình nghiệp vụ (Business Model)** 4](#_Toc190406567)

[**2.5. Phân tích người dùng (User Persona)** 4](#_Toc190406568)

[**3. PHÂN TÍCH HỆ THỐNG** 4](#_Toc190406569)

[**3.1. Mô hình Use Case** 4](#_Toc190406570)

[**3.2. Biểu đồ hoạt động (Activity Diagram)** 4](#_Toc190406571)

[**3.3. Biểu đồ trình tự (Sequence Diagram)** 4](#_Toc190406572)

[**3.4. Biểu đồ lớp (Class Diagram)** 4](#_Toc190406573)

[**3.5. Biểu đồ thực thể - quan hệ (ERD)** 4](#_Toc190406574)

[**4. THIẾT KẾ HỆ THỐNG** 4](#_Toc190406575)

[**4.1. Kiến trúc tổng quan** 4](#_Toc190406576)

[**4.2. Thiết kế giao diện người dùng** 4](#_Toc190406577)

[**4.2.1. Trang chủ** 5](#_Toc190406578)

[**4.2.2. Trang sản phẩm** 5](#_Toc190406579)

[**4.2.3. Giỏ hàng** 5](#_Toc190406580)

[**4.2.4. Đăng nhập, đăng ký** 5](#_Toc190406581)

[**4.3. Thiết kế cơ sở dữ liệu** 5](#_Toc190406582)

[**4.4. Thiết kế API backend** 5](#_Toc190406583)

[**5. CÀI ĐẶT VÀ TRIỂN KHAI** 5](#_Toc190406584)

[**5.1. Công nghệ sử dụng** 5](#_Toc190406585)

[**5.2. Triển khai backend với Node.js** 5](#_Toc190406586)

[**5.3. Triển khai frontend với Vue.js** 5](#_Toc190406587)

[**5.4. Quản lý cơ sở dữ liệu với MySQL** 5](#_Toc190406588)

[**5.5. Kết nối API giữa frontend và backend** 5](#_Toc190406589)

[**6. KIỂM THỬ VÀ ĐÁNH GIÁ** 5](#_Toc190406590)

[**6.1. Phương pháp kiểm thử** 5](#_Toc190406591)

[**6.2. Kết quả kiểm thử** 5](#_Toc190406592)

[**6.3. Đánh giá hiệu suất hệ thống** 5](#_Toc190406593)

[**6.4. Hạn chế và hướng phát triển** 5](#_Toc190406594)

**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. MỞ ĐẦU**

## **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)**

A screenshot of a diagram

AI-generated content may be incorrect.

*Hình : 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)**

Phân tích người dùng (User Persona) giúp xác định đối tượng sử dụng hệ thống, từ đó thiết kế giao diện và tính năng phù hợp với nhu cầu thực tế.

**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.

Phân tích personas giúp định hướng thiết kế hệ thống **"PlayBook"**, tập trung vào các tính năng quan trọng và tối ưu trải nghiệm người dùng.

# **3. PHÂN TÍCH HỆ THỐNG**

## **3.1. Mô hình Use Case**

3.1.1. User case khách hàng

A diagram of a person with lines and dots

AI-generated content may be incorrect.

*Hình : Usercase 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. Usercase admin

A diagram of a person with text and circles

AI-generated content may be incorrect.

Hình : Usercase 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)**

## **3.3. Biểu đồ trình tự (Sequence Diagram)**

A diagram with black text

AI-generated content may be incorrect.

Hình : Biểu đồ trình tự

## **3.4. Biểu đồ lớp (Class Diagram)**

## **3.5. Biểu đồ thực thể - quan hệ (ERD)**

# **4. THIẾT KẾ HỆ THỐNG**

## **4.1. Kiến trúc tổng quan**

## **4.2. Thiết kế giao diện người dùng**

### **4.2.1. Trang chủ**

### **4.2.2. Trang sản phẩm**

### **4.2.3. Giỏ hàng**

### **4.2.4. Đăng nhập, đăng ký, quên mật khẩu**

**4.2.5. Trang người dùng**

**4.2.6. Trang admin**

## **4.3. Thiết kế cơ sở dữ liệu**

## **4.4. Thiết kế API backend**

# **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).

DevOps: Docker để container hóa ứng dụng, Jenkins cho CI/CD, AWS EC2 để triển khai server.

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

1. 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.
2. 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.
3. Triển khai lên server:
   * Build ứng dụng với npm run build.
   * Sử dụng PM2 để quản lý process trên AWS EC2.

## **5.3. Triển khai frontend với Vue.js**

**Quy trình triển khai**:

1. Khởi tạo dự án:

vue create playbook-frontend

1. Cài đặt thư viện:

npm install vuex axios vuetify vue-router

1. Xây dựng component:
   * Thiết kế trang chủ, trang sản phẩm, giỏ hàng bằng Vuetify.
   * Tích hợp Vue Router để điều hướng giữa các trang.
2. 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).
3. Tối ưu hóa:
   * Áp dụng lazy loading cho hình ảnh.
   * Triển khai SSR (Server-Side Rendering) bằng Nuxt.js để cải thiện SEO.
4. Deploy:
   * Build ứng dụng với npm run build.
   * Host trên AWS S3 kết hợp CloudFront để phân phối CDN.

## **5.4. Quản lý cơ sở dữ liệu với MySQL**

**Thiết kế database**:

* Bảng *users*: Lưu thông tin người dùng (id, email, password\_hash, role).
* Bảng *books*: Lưu thông tin sách (id, title, author, price, quantity).
* Bảng *orders*: Lưu đơn hàng (id, user\_id, total\_price, status).
* Bảng *order\_items*: Liên kết đơn hàng với sách (order\_id, book\_id, quantity).

**Triển khai**:

* Sử dụng MySQL Workbench để thiết kế schema và tạo indexes cho các trường tìm kiếm thường xuyên (ví dụ: title, author).
* Backup dữ liệu định kỳ bằng cron job.
* Tối ưu query bằng EXPLAIN để tránh full-table scan.

## **5.5. Kết nối API giữa frontend và backend**

**Quy trình tích hợp**:

1. Định nghĩa API contract: Sử dụng OpenAPI (Swagger) để mô tả các endpoint.
2. Xử lý CORS: Cấu hình middleware Express.js để cho phép truy cập từ domain frontend.
3. Xác thực token:
   * Frontend gửi token JWT qua header Authorization sau khi đăng nhập.
   * Backend giải mã token và kiểm tra quyền truy cập.
4. Xử lý lỗi:
   * Trả về HTTP status code phù hợp (ví dụ: 404 cho "Không tìm thấy sách").
   * Thống nhất format response (ví dụ: { success: boolean, data: any, message: string }).

# **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:

* Tích hợp VNPay và Momo để hỗ trợ thanh toán đa dạng.
* 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.