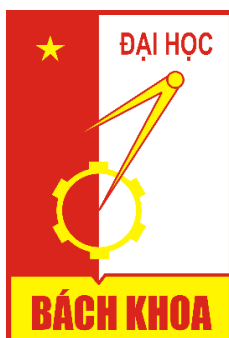


TRƯỜNG ĐẠI HỌC BÁCH KHOA HÀ NỘI
VIỆN CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG

_____ * _____



ĐỒ ÁN
TỐT NGHIỆP ĐẠI HỌC
NGÀNH CÔNG NGHỆ THÔNG TIN

**HỆ THỐNG ĐÁNH GIÁ SÁCH
LOVEBOOK**

Sinh viên thực hiện: **Nguyễn Huy Long**
Lớp CN CNTT 1 – K58

Giáo viên hướng dẫn: **ThS. Nguyễn Hồng Phương**

HÀ NỘI 5 – 2017

PHIẾU GIAO NHIỆM VỤ ĐỒ ÁN TỐT NGHIỆP

1. Thông tin về sinh viên

Họ và tên sinh viên: Nguyễn Huy Long

Điện thoại liên lạc: 01699659849

Email: nguyenhuylong5695@gmail.com

Lớp: CN CNTT 1 – K58

Hệ đào tạo: Đại học chính quy

Đồ án tốt nghiệp được thực hiện tại: Bộ môn Hệ Thống Thông Tin, Viện Công Nghệ Thông Tin và Truyền Thông, Đại học Bách Khoa Hà Nội

Thời gian làm ĐATN: Từ ngày 13/02/2017 đến 19/05/2017

2. Mục đích nội dung của ĐATN

Nghiên cứu ngôn ngữ lập trình Ruby và Framework Rails. Ứng dụng xây dựng hệ thống đánh giá sách LoveBook.

3. Các nhiệm vụ cụ thể của ĐATN

- Nghiên cứu ngôn ngữ lập trình Ruby và Framework Rails.
- Tìm hiểu, khảo sát bài toán thực tế, phân tích các yêu cầu bài toán.
- Xây dựng hệ thống đánh giá sách LoveBook và triển khai hệ thống lên server Heroku.

4. Lời cam đoan của sinh viên:

Tôi – *Nguyễn Huy Long* – cam kết ĐATN là công trình nghiên cứu của bản thân tôi dưới sự hướng dẫn của *ThS. Nguyễn Hồng Phương*.

Các kết quả nêu trong ĐATN là trung thực, không phải là sao chép toàn văn của bất kỳ công trình nào khác.

Hà Nội, ngày tháng năm 2017

Tác giả ĐATN

Nguyễn Huy Long

5. Xác nhận của giáo viên hướng dẫn về mức độ hoàn thành của ĐATN và cho phép bảo vệ:

Hà Nội, ngày tháng năm

Giáo viên hướng dẫn

ThS. Nguyễn Hồng Phương

TÓM TẮT NỘI DUNG ĐỒ ÁN TỐT NGHIỆP

Đề tài nghiên cứu việc xây dựng một website trên nền tảng ngôn ngữ lập trình Ruby, Framework Rails và triển khai trên server Heroku. Cụ thể là một website đánh giá sách.

Mục đích của website này là nơi để những người ham mê với việc đọc sách có thể viết bài đánh giá về những cuốn sách hay mà họ yêu thích cũng như những cuốn sách mà họ không thích. Những người muốn tìm hiểu về một cuốn sách cũng có thể lấy đây làm một trong những địa chỉ để tham khảo trước khi mua hay đọc. Từ đó xây dựng nên một cộng đồng những người thích đọc và bàn luận về sách.

Nội dung đồ án được trình bày theo các thành phần chính sau:

Phần 1: Khảo sát hệ thống

Phần này sẽ nêu ra những vấn đề gặp phải trong thực tế khi tìm hiểu, phân tích đề tài và đề xuất giải pháp.

Giới thiệu về các công nghệ sẽ áp dụng để giải quyết bài toán.

Phần 2: Phân tích hệ thống

Phần này sẽ phân tích các yêu cầu chức năng và phi chức năng của hệ thống.

Phần 3: Thiết kế hệ thống

Phần này bao gồm:

- Kiến trúc hệ thống theo Framework Ruby on Rails
- Thiết kế cơ sở dữ liệu
- Thiết kế giao diện

Phần 4: Cài đặt hệ thống

Kết quả cài đặt chương trình.

Phần 5: Kết luận

Kết luận chung về đồ án và định hướng phát triển trong tương lai.

LỜI CẢM ƠN

Lời đầu tiên, em xin gửi lời cảm ơn tới các thầy cô trong trường Đại học Bách Khoa Hà Nội, đặc biệt là các thầy cô thuộc Viện Công nghệ Thông tin và Truyền thông đã truyền đạt kiến thức và kinh nghiệm cho em trong những năm học đại học.

Em xin được gửi lời cảm ơn và tri ân sâu sắc đến **Thạc sĩ Nguyễn Hồng Phương**, người đã tận tình hướng dẫn và giúp đỡ em trong suốt quá trình thực hiện đồ án tốt nghiệp.

Cuối cùng em xin gửi lời cảm ơn tới gia đình, bạn bè đã quan tâm, động viên và giúp đỡ em về mặt tinh thần cũng như trong những lúc gặp khó khăn để em có thể hoàn thành đồ án tốt nghiệp.

Với điều kiện thời gian có hạn cũng như khả năng, kinh nghiệm còn hạn chế nên đồ án của em sẽ không thể tránh khỏi những thiếu sót. Em rất mong nhận được sự chỉ bảo, đóng góp ý kiến của thầy cô cùng toàn thể các bạn để em có điều kiện bổ sung, nâng cao kiến thức của mình và để đồ án của em được hoàn thiện hơn.

Em xin trân trọng cảm ơn!

MỤC LỤC

PHIẾU GIAO NHIỆM VỤ ĐỒ ÁN TỐT NGHIỆP	2
TÓM TẮT NỘI DUNG ĐỒ ÁN TỐT NGHIỆP	3
LỜI CẢM ƠN	4
DANH MỤC HÌNH ẢNH	8
DANH MỤC BẢNG	10
PHẦN 1 – KHẢO SÁT	11
1.1. Đặt vấn đề	11
1.1.1. Bối cảnh.....	11
1.1.2. Đề xuất đề tài.....	11
1.2. Phân tích đề tài	11
1.2.1 Phạm vi của đề tài.....	11
1.2.2. Đối tượng của đề tài	12
1.2.3. Vai trò và ý nghĩa của đề tài.....	12
1.3. Mô tả bài toán.....	12
1.3.1. Giới thiệu bài toán	12
1.3.2. Những thuận lợi và khó khăn của đề tài.....	13
1.3.3. Tính khả thi của đề tài	13
1.4. Các công nghệ sử dụng	13
1.4.1. Ngôn ngữ lập trình Ruby	14
1.4.2. Framework Ruby on Rails (Rails).....	14
1.4.3. SQLite.....	17
1.4.4. GitHub	18
1.4.5. Heroku	18
PHẦN 2 – PHÂN TÍCH HỆ THỐNG	19
2.1. Phân tích yêu cầu tổng quát	19
2.1.1. Các tác nhân của hệ thống.....	19
2.1.2. Biểu đồ ca sử dụng tổng quát	20

2.2. Phân tích yêu cầu chi tiết chức năng	21
2.2.1. Đăng kí	21
2.2.2. Đăng nhập.....	22
2.2.3. Đăng xuất.....	22
2.2.4. Xem danh sách sách	23
2.2.5. Tìm kiếm sách	23
2.2.6. Xem chi tiết một cuốn sách	24
2.2.7. Viết bài đánh giá sách	25
2.2.8. Quản lý bài đánh giá sách.....	26
2.2.9. Quản lý bình luận	28
2.2.10. Báo cáo bài đánh giá sách	31
2.2.11. Gửi yêu cầu sách	32
2.2.12. Quản lý yêu cầu sách.....	34
2.2.13. Đánh dấu sách yêu thích.....	35
2.2.14. Xem danh sách thành viên.....	37
2.2.15. Xem thông tin thành viên	37
2.2.16. Theo dõi thành viên.....	38
2.2.17. Quản lý thông tin cá nhân.....	39
2.2.18. Quản lý thành viên.....	40
2.2.19. Quản lý thể loại sách	41
2.2.20. Quản lý sách	43
2.2.21. Quản lý báo cáo đánh giá sách	46
2.2.22. Quản lý yêu cầu sách của thành viên	46
2.3. Yêu cầu phi chức năng	48
PHẦN 3 – THIẾT KẾ HỆ THỐNG.....	49
3.1. Kiến trúc Ruby on Rails	49
3.2. Thiết kế cơ sở dữ liệu	53
3.2.1. Cơ sở dữ liệu của hệ thống	53
3.2.2. Bảng User	54
3.2.3. Bảng Book	54
3.2.4. Bảng Category	55
3.2.5. Bảng Favourite Book.....	55
3.2.6. Bảng Review	55
3.2.7. Bảng Comment.....	56
3.2.8. Bảng Request.....	56
3.2.9. Bảng Report.....	56
3.2.10. Bảng Activity.....	57

3.2.11. Bảng Relationship	57
3.3. Thiết kế giao diện.....	58
3.3.1. Giao diện trang chủ	58
3.3.2. Giao diện đăng nhập và đăng ký	59
3.3.3. Giao diện trang xem sách và tìm kiếm sách.....	60
3.3.4. Giao diện bài đánh giá.....	63
3.3.5. Giao diện viết bài đánh giá.....	64
3.3.6. Giao diện quản lý yêu cầu của thành viên.....	65
3.3.7. Giao diện trang cá nhân của thành viên	66
3.3.8. Giao diện danh sách thành viên.....	67
3.3.9. Giao diện quản lý thể loại sách của quản trị viên.....	68
3.3.10. Giao diện thêm mới sách	69
3.3.11. Giao diện quản lý yêu cầu của quản trị viên	69
3.3.12. Giao diện quản lý báo cáo của quản trị viên	70
PHẦN 4 – CÀI ĐẶT HỆ THỐNG	71
4.1. Kết quả cài đặt.....	71
4.1.1. Giao diện trang chủ	71
4.1.2. Giao diện trang đăng kí, đăng nhập.....	72
4.1.3. Giao diện trang xem sách và tìm kiếm sách.....	72
4.1.4. Giao diện đối với thành viên	73
4.1.5. Giao diện đối với quản trị viên.....	76
4.2. Đánh giá hệ thống	78
4.2.1. Ưu điểm.....	78
4.2.2. Nhược điểm	78
PHẦN 5 – KẾT LUẬN	79
5.1. Kết luận	79
5.2. Định hướng phát triển	79
TÀI LIỆU THAM KHẢO.....	80

DANH MỤC HÌNH ẢNH

Hình 1: Logo ngôn ngữ Ruby	14
Hình 2: Logo Framework Ruby on Rails (Rails)	15
Hình 3: Mô hình MVC trong Framework Rails	16
Hình 4: Biểu tượng của SQLite	17
Hình 5: Biểu tượng của GitHub	18
Hình 6: Biểu tượng của Heroku	18
Hình 7: Mối quan hệ của các tác nhân trong hệ thống	19
Hình 8: Biểu đồ ca sử dụng tổng quát của hệ thống	20
Hình 9: Biểu đồ hoạt động chức năng “Tìm kiếm sách”	24
Hình 10: Biểu đồ hoạt động chức năng “Viết bài đánh giá sách”	26
Hình 11: Biểu đồ phân rã usecase “Quản lý bài đánh giá sách”	26
Hình 12: Biểu đồ phân rã usecase “Quản lý bình luận”	28
Hình 13: Biểu đồ hoạt động chức năng “Quản lý bình luận”	30
Hình 14: Biểu đồ hoạt động chức năng “Báo cáo bài đánh giá”	32
Hình 15: Biểu đồ hoạt động chức năng “Gửi yêu cầu sách”	33
Hình 16: Biểu đồ phân rã usecase “Quản lý yêu cầu sách”	34
Hình 17: Biểu đồ hoạt động chức năng “Đánh dấu sách yêu thích”	36
Hình 18: Biểu đồ hoạt động chức năng “Theo dõi thành viên”	39
Hình 19: Biểu đồ phân rã usecase “Quản lý thể loại sách”	41
Hình 20: Biểu đồ hoạt động chức năng “Quản lý thể loại sách”	43
Hình 21: Biểu đồ phân rã usecase “Quản lý sách”	43
Hình 22: Biểu đồ hoạt động chức năng “Quản lý sách”	45
Hình 23: Biểu đồ phân rã usecase “Quản lý yêu cầu sách của thành viên”	46
Hình 24: Biểu đồ hoạt động chức năng “Quản lý yêu cầu sách của thành viên”	47
Hình 25: Kiến trúc tổng quan của Ruby on Rails	49
Hình 26: Cơ sở dữ liệu của hệ thống	53
Hình 27: Mockup giao diện Trang chủ trước đăng nhập	58
Hình 28: Mockup giao diện Trang chủ sau khi đã đăng nhập	59
Hình 29: Mockup trang đăng nhập	59
Hình 30: Mockup trang đăng ký	60
Hình 31: Mockup trang hiển thị và tìm kiếm sách của khách và thành viên	61
Hình 32: Mockup trang hiển thị và tìm kiếm sách của quản trị viên	61
Hình 33: Mockup trang xem chi tiết sách	62
Hình 34: Mockup trang xem chi tiết một bài đánh giá sách của thành viên	63
Hình 35: Mockup trang xem chi tiết một bài đánh giá sách của quản trị viên	64
Hình 36: Mockup trang viết bài đánh giá	65
Hình 37: Mockup trang quản lý yêu cầu của thành viên	65
Hình 38: Mockup biểu mẫu gửi yêu cầu	66

Hình 39: Mockup trang cá nhân của thành viên	67
Hình 40: Mockup trang hiển thị danh sách thành viên của thành viên.....	67
Hình 41: Mockup hiển thị danh sách thành viên của quản trị viên.....	68
Hình 42: Mockup trang quản lý thể loại của quản trị viên	68
Hình 43: Mockup biểu mẫu thêm sách mới dành cho quản trị viên	69
Hình 44: Mockup trang quản lý yêu cầu của quản trị viên	70
Hình 45: Mockup trang quản lý báo cáo của quản trị viên	70
Hình 46: Giao diện trang chủ trước đăng nhập.....	71
Hình 47: Giao diện trang chủ sau khi đăng nhập.....	71
Hình 48: Giao diện trang đăng nhập	72
Hình 49: Giao diện trang đăng ký	72
Hình 50: Giao diện trang hiển thị danh sách và tìm kiếm sách	72
Hình 51: Giao diện trang hiển thị thông tin một cuốn sách cụ thể	73
Hình 52: Giao diện trang hiển thị danh sách thành viên.....	73
Hình 53: Giao diện trang cá nhân (1).....	74
Hình 54: Giao diện trang cá nhân (2).....	74
Hình 55: Giao diện trang viết bài đánh giá	74
Hình 56: Giao diện xem chi tiết một bài đánh giá của người dùng khác.....	75
Hình 57: Giao diện xem chi tiết một bài đánh giá của tài khoản đang đăng nhập ...	75
Hình 58: Giao diện trang quản lý yêu cầu của thành viên	76
Hình 59: Giao diện trang quản lý thể loại sách của quản trị viên	76
Hình 60: Giao diện trang thêm sách mới của quản trị viên	77
Hình 61: Giao diện trang quản lý thành viên của quản trị viên	77
Hình 62: Giao diện trang quản lý yêu cầu của quản trị viên.....	78
Hình 63: Giao diện trang quản lý báo cáo của quản trị viên.....	78

DANH MỤC BẢNG

Bảng 1: Bảng thông tin chi tiết User.....	54
Bảng 2: Bảng thông tin chi tiết Book.....	54
Bảng 3: Bảng thông tin chi tiết Category.....	55
Bảng 4: Bảng thông tin chi tiết Favourite Book	55
Bảng 5: Bảng thông tin chi tiết Review	55
Bảng 6: Bảng thông tin chi tiết Comment.....	56
Bảng 7: Bảng thông tin chi tiết Request	56
Bảng 8: Bảng thông tin chi tiết Report	56
Bảng 9: Bảng thông tin chi tiết Activity	57
Bảng 10: Bảng thông tin chi tiết Relationship	57

PHẦN 1 – KHẢO SÁT

1.1. Đặt vấn đề

1.1.1. Bối cảnh

Nhu cầu đọc sách ở nước ta ngày càng cao. Tuy nhiên, nên đọc sách nào để không mất thời gian là một vấn đề khó. Với sách tiếng Anh, trước khi đọc, nhiều người trong chúng ta thường sử dụng Amazon để xem các đánh giá của người dùng và chuyên gia đối với cuốn sách mình quan tâm.

Mới đây, một nghiên cứu do giáo sư Micheal Luca thuộc Trường Kinh Doanh Harvard đã phát hiện ra rằng điểm số đánh giá sách trung bình của các chuyên gia phê bình xấp xỉ với điểm số do độc giả bình chọn trên trang Amazon. Do vậy, khi tìm kiếm sách trên trang web này, bạn hoàn toàn có thể tin tưởng vào điểm số đánh giá chất lượng của các độc giả để quyết định có nên đọc nó hay không, bởi vì, nó cũng đáng tin như của các chuyên gia vậy.

Bên cạnh sự tương đồng lớn giữa nhà phê bình và người đọc, giáo sư Luca cũng bắt gặp một hiện tượng khá thú vị, đó là các chuyên gia có xu hướng cho điểm sách của các tác giả phổ biến hoặc đạt giải thưởng điểm số cao còn người dùng lại thường nghiêng về phía các tác giả mới viết sách lần đầu tiên. Có vẻ như các phê bình thường bảo thủ và chậm đọc các sách mới của các tác giả chưa nổi tiếng hơn một chút so với người đọc bình thường.[1]

1.1.2. Đề xuất đề tài

Qua nghiên cứu trên, chúng ta thấy rõ sự cần thiết có một công cụ nhanh chóng và đơn giản để giúp ta lựa chọn một cuốn sách hay giữa hàng ngàn cuốn sách được xuất bản mỗi năm. Tất cả những gì cần làm là vào website đó, gõ tên sách mà ta quan tâm, và xem đánh giá của những độc giả khác trước khi quyết định đọc cuốn sách đó hay không.

Chính vì vậy, website LoveBook đã ra đời để giải quyết nhu cầu đó bằng cách kết nối những người yêu sách trên khắp mọi nơi thông qua mạng Internet, tạo ra một cộng đồng những người yêu sách và chia sẻ với nhau niềm vui đọc sách.

1.2. Phân tích đề tài

1.2.1 Phạm vi của đề tài

Trong giới hạn của một đồ án tốt nghiệp cùng với thời gian nghiên cứu, phân tích, thực hiện và triển khai, đề tài sẽ tập trung nghiên cứu ngôn ngữ lập trình Ruby, Framework Rails cùng các công nghệ để xây dựng một trang web đánh giá sách thân thiện và phù hợp với người dùng ở Việt Nam.

1.2.2. Đối tượng của đề tài

Về mặt công nghệ, đối tượng nghiên cứu của đề tài là ngôn ngữ Ruby và Framework Rails cùng các công nghệ liên quan để có thể xây dựng nên một website.

Về mặt tác nhân, đề tài nghiên cứu về nhu cầu của những người yêu sách, ham mê đọc sách. Nhu cầu chủ yếu của họ là tìm hiểu về những cuốn sách, chia sẻ suy nghĩ sau khi đọc một cuốn sách hoặc là giao lưu với những người có cùng chung sở thích....

1.2.3. Vai trò và ý nghĩa của đề tài

Đối với bản thân: Cơ hội để nghiên cứu và tìm hiểu các công nghệ mới, cụ thể là ngôn ngữ Ruby và Framework Rails

Đối với thực tế: Xây dựng một trang web phục vụ cho những nhu cầu thiết thực của cộng đồng những người yêu sách.

1.3. Mô tả bài toán

1.3.1. Giới thiệu bài toán

LoveBook là một trang web đánh giá sách, được xây dựng bằng ngôn ngữ Ruby và Framework Rails. Đây là địa chỉ tin cậy để độc giả tìm kiếm cho mình một cuốn sách hay và phù hợp với bản thân, đồng thời đây cũng là nơi mọi người chia sẻ với nhau về những cuốn sách mình đã đọc.

Các chức năng chính dành cho người dùng mà website cần có:

- Đăng kí tài khoản trên website để có thể đăng nhập và đăng xuất.
- Có thể xem toàn bộ sách có trên website và tìm kiếm theo các tiêu chí như tên sách, tên tác giả, thể loại...
- Sau khi đăng nhập, người dùng có thể xem cụ thể một cuốn sách và viết bài đánh giá cho cuốn sách đó.
- Người dùng có thể viết bình luận cho bài đánh giá của mình cũng như của những người khác.
- Khi thấy những bài đánh giá có nội dung không phù hợp, người dùng có thể báo cáo tới quản trị viên để quản trị viên xử lý bài đánh giá đó.
- Người dùng có thể đánh dấu một cuốn sách trở thành cuốn sách yêu thích của mình.
- Để website trở thành một cộng đồng thu nhỏ của những người yêu sách, người dùng có thể theo dõi người dùng khác để có thể xem được hoạt động của họ.
- Ngoài ra, nếu thấy trên website chưa có quyển sách mà mình quan tâm, người dùng có thể gửi yêu cầu tới quản trị viên để đưa cuốn sách đó vào hệ thống.

Đối với tác nhân là Quản trị viên, website cần đáp ứng được các chức năng:

- Quản lý sách: Tạo mới hoặc sửa, xóa một cuốn sách.
- Quản lý bài đánh giá: Xóa những bài đánh giá có nội dung không phù hợp.

- Quản lý người dùng: Xóa những người dùng thường xuyên vi phạm nội quy của website
- Quản lý yêu cầu sách: Đồng ý hoặc từ chối yêu cầu cập nhật sách do người dùng gửi đến.

1.3.2. Những thuận lợi và khó khăn của đề tài

- *Thuận lợi:*
 - Cú pháp của Ruby khá đơn giản và dễ hiểu.
 - Framework Rails cung cấp cho người dùng nhiều thư viện mã nguồn mở có sẵn, tạo sự tiện lợi, rút ngắn thời gian code cho lập trình viên.
 - Cộng đồng Ruby on Rails ở nước ngoài lớn, cùng với cộng đồng tại Việt Nam cũng đang phát triển không ngừng, tạo điều kiện cho việc tiếp cận tốt hơn với công nghệ mới này.
- *Khó khăn:*
 - Ruby là một ngôn ngữ lập trình tương đối mới, nguồn tài nguyên học liệu để tiếp cận nghiên cứu và tìm hiểu còn hạn chế.
 - Dù có nhu cầu lớn trong việc tìm một cuốn sách hay nhưng người đọc ở Việt Nam chưa quen lắm với một trang web đánh giá sách, nên bước đầu website sẽ khó thu hút được nhiều người quan tâm.

1.3.3. Tính khả thi của đề tài

Mặc dù tỉ lệ đọc sách ở nước ta còn thấp, nhưng nhu cầu đọc sách ở nước ta đã tăng cao trong những năm gần đây. Số lượng sách xuất bản hằng năm từ đó cũng tăng theo, nảy sinh nhu cầu tìm kiếm những cuốn sách hay và phù hợp với bản thân trong hàng trăm nghìn cuốn sách có trên thị trường. Một trang web như LoveBook sẽ có khả năng giải quyết yêu cầu bài toán được đặt ra.

Một số trang web tương tự nổi tiếng trên Thế giới có thể kể đến như: Goodreads (goodreads.com), BookBrowse (bookbrowse.com), LibraryThing (librarything.com).

Ở Việt Nam cũng có trang web đánh giá sách đơn giản và chưa thu hút được nhiều người quan tâm: Sách Thần Kỳ (sachthanky.com).

1.4. Các công nghệ sử dụng

- Back-End: Sử dụng ngôn ngữ Ruby với Framework Ruby on Rails (Rails).
- Front-End: HTML, Javascript, CSS.
- Database: SQLite3.
- GitHub: Quản lý mã nguồn đề tài.
- Heroku: Nền tảng giúp triển khai ứng dụng thực tế.

1.4.1. Ngôn ngữ lập trình Ruby



Hình 1: Logo ngôn ngữ Ruby

Ruby là một ngôn ngữ lập trình động, mã nguồn mở, chú trọng vào đơn giản và hiệu suất cao. Nó có một cú pháp rõ ràng, gần gũi và dễ viết.

Ruby được tạo ra bởi Yukihiro "Matz" Matsumoto vào ngày 24/2/1993 và đưa ra bản chính thức vào năm 1995. Phiên bản mới nhất hiện nay của Ruby là 2.3.4 ra mắt vào ngày 30/3/2017.

Quan điểm chính trong việc thiết kế của Matz là nhằm giảm thiểu các công việc nhàm chán mà các lập trình viên buộc phải làm, tiếp đến là nguyên tắc thiết kế giao diện người dùng (user interface) hiệu quả. Ông nhấn mạnh rằng việc thiết kế hệ thống cần phải tập trung vào con người hơn là vào máy tính.

Yukihiro “Matz” Matsumoto đã kết hợp những đặc điểm hay từ các ngôn ngữ như Perl, Smalltalk, Eiffel, Ada và Lisp để tạo ra một ngôn ngữ cân bằng giữa lập trình hướng chức năng (functional programming) và lập trình hướng thủ tục (imperative programming). Ruby hỗ trợ hầu hết các mô hình lập trình truyền thống, bao gồm lập trình hướng đối tượng, lập trình phản xạ, lập trình mệnh lệnh.... Ruby được lấy cảm hứng từ Perl, Smalltalk, Eiffel và Lisp. Ông nói: “Tôi muốn tìm một ngôn ngữ scripting lợi hại hơn Perl, và hướng đối tượng hơn Python.”

Sau khi được phát hành ra công chúng năm 1995, Ruby đã thu hút nhiều lập trình viên. Vào năm 2006, Ruby đã được thừa nhận ở mức phổ biến, điều này được chứng minh qua số lượng lớn những nhóm người dùng hoạt động tích cực ở các thành phố lớn trên thế giới và tại các hội nghị liên quan đến Ruby luôn bán hết vé.[2]

1.4.2. Framework Ruby on Rails (Rails)

Ruby On Rails (gọi tắt là Rails) là một Framework cho phép phát triển ứng dụng Web gồm 2 phần cơ bản:

- Phần ngôn ngữ Ruby.
- Phần Framework Rails bao gồm nhiều thư viện liên kết.



Hình 2: Logo Framework Ruby on Rails (Rails)

Rails ra mắt công chúng lần đầu tiên vào năm 2004, Rails thoát đầu được dùng như là nền tảng cho một công cụ quản lý dự án được đặt tên là Basecamp và được tạo ra bởi nhà phát triển web David Heinemeier Hansson, một nhân viên của công ty phát triển website 37signals (Mỹ).

Ban đầu họ xây dựng Rails không phải với mục đích là xây dựng ra một Framework riêng, chủ tâm ban đầu là dùng nó để xây dựng các ứng dụng khác của 37signals. Sau đó Heinemeier Hansson thấy tiềm năng của nó giúp cho anh ấy làm các công việc dễ dàng hơn bằng cách rút ra các tính năng phổ biến như trừu tượng cơ sở dữ liệu và khuôn mẫu (template) bên trong, và sau đó nó trở thành phiên bản đầu tiên được tung ra của Ruby on Rails.[3]

Phiên bản mới nhất hiện nay của Rails là Rails 5.1.0, ra mắt ngày 27/04/2017.

Mô hình MVC trong Ruby on Rails

Mô hình MVC (viết tắt của 3 từ Model-View-Controller) là một mô hình kiến trúc phần mềm mà chia ứng dụng thành ba phần khác nhau là Model, View và Controller. Ba thành phần này có nhiệm vụ riêng biệt và độc lập với các thành phần khác, nhưng lại tương tác với nhau để xây dựng nên từng chức năng của một ứng dụng. Dưới đây là mô tả của ba thành phần này trong Rails:

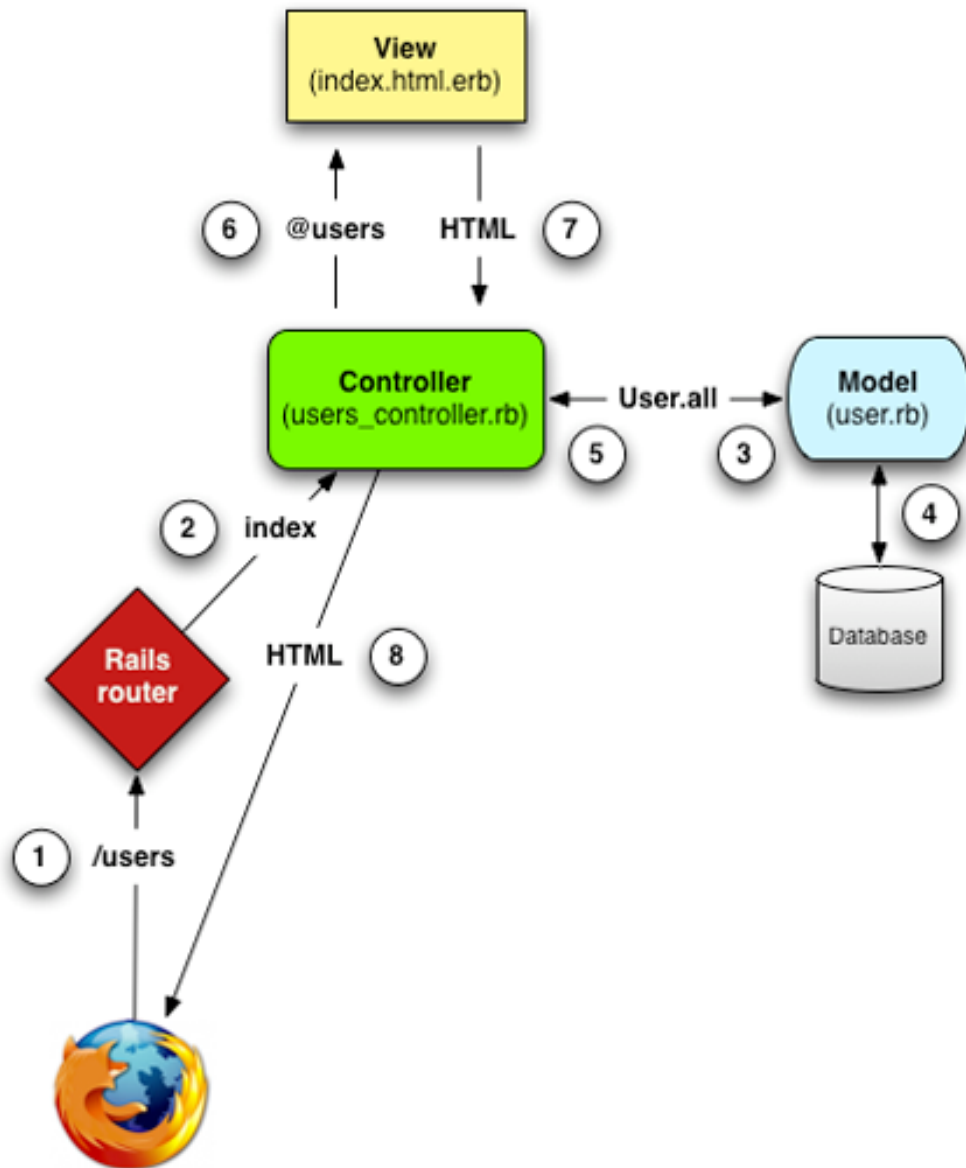
- **Model:** Thành phần trực tiếp thao tác với dữ liệu (database) và xử lý các nghiệp vụ logic trên dữ liệu đó. Trong Rails, Model được triển khai thông qua thư viện ActiveRecord. Với ActiveRecord, chúng ta có thể thực hiện các thao tác CRUD (create, read, update, delete) với dữ liệu trong database thông qua các method của Ruby Object tương ứng mà không cần viết các truy vấn SQL thông thường. Ngoài ra, Model còn thực hiện các xử lý như là validate dữ liệu, thiết lập các quan hệ, thực thi các transaction...

- **View:** Thành phần chịu trách nhiệm hiển thị dữ liệu và giao diện tương tác với người dùng. Trong Rails, View là các file html được nhúng code ruby (file .erb hoặc .haml tùy config) và được triển khai thông qua ActionView.

- **Controller:** Thành phần trung gian giữa Model và View. Cụ thể, thông thường khi 1 request được xử lý bởi Controller, nó sẽ thực hiện việc lấy dữ liệu từ database

thông qua Model, xử lý cơ sở dữ liệu đó và trả dữ liệu cho View để hiển thị thông tin. Trong Rails, Controller được triển khai thông qua ActionController.[4]

Dưới đây là ví dụ về cách mà Rails tương tác giữa các thành phần MVC để xử lý một request của người dùng.



Hình 3: Mô hình MVC trong Framework Rails

Trong mô hình trên, các bước thực hiện được tóm tắt như sau:

1. Trình duyệt web thực hiện một request đến URL `/users` để lấy danh sách các user.
2. Rails dẫn request đến action `index` trong `Users Controller` để xử lý.
3. Controller request đến Model `User` để lấy dữ liệu.

4. Model truy vấn đến cơ sở dữ liệu để lấy dữ liệu ra.
5. Model trả về dữ liệu danh sách các user cho Controller.
6. Controller lưu dữ liệu này vào biến `@users` và truyền xuống View.
7. View sử dụng code ruby nhúng trong file `index.erb` để hiển thị dữ liệu thành HTML.
8. Controller trả lại HTML cho trình duyệt.

1.4.3. SQLite



Hình 4: Biểu tượng của SQLite

SQLite là hệ thống cơ sở dữ liệu quan hệ nhỏ gọn, hoàn chỉnh, có thể cài đặt bên trong các trình ứng dụng khác. SQLite được Richard Hipp viết dưới dạng thư viện bằng ngôn ngữ lập trình C.

SQLite có các ưu điểm sau:

- Tin cậy: các hoạt động chuyển giao nội trong cơ sở dữ liệu được thực hiện trọn vẹn, không gây lỗi khi xảy ra sự cố phần cứng.
- Tuân theo chuẩn SQL92 (chỉ có một vài đặc điểm không hỗ trợ).
- Không cần cài đặt cấu hình.
- Kích thước chương trình gọn nhẹ, với cấu hình đầy đủ chỉ không đầy 300 kB.
- Thực hiện các thao tác đơn giản nhanh hơn các hệ thống cơ sở dữ liệu khách/chủ khác.
- Không cần phần mềm phụ trợ.
- Phần mềm tự do với mã nguồn mở, được chú thích rõ ràng.[5]

1.4.4. GitHub



Hình 5: Biểu tượng của GitHub

GitHub là một dịch vụ cung cấp kho lưu trữ mã nguồn Git dựa trên nền web cho các dự án phát triển phần mềm. GitHub cung cấp cả phiên bản trả tiền lẫn miễn phí cho các tài khoản. Các dự án mã nguồn mở sẽ được cung cấp kho lưu trữ miễn phí. Tính đến tháng 4 năm 2016, GitHub có hơn 14 triệu người sử dụng với hơn 35 triệu kho mã nguồn, làm cho nó trở thành máy chủ chứa mã nguồn lớn trên thế giới.

Github đã trở thành một yếu tố có sức ảnh hưởng trong cộng đồng phát triển mã nguồn mở. Thậm chí nhiều nhà phát triển đã bắt đầu xem nó là một sự thay thế cho sơ yếu lý lịch và một số nhà tuyển dụng yêu cầu các ứng viên cung cấp một liên kết đến tài khoản Github để đánh giá ứng viên.

Linh vật của Github là chú mèo Octocat.[6]

1.4.5. Heroku



Hình 6: Biểu tượng của Heroku

Heroku là một nền tảng đám mây giúp xây dựng và triển khai các ứng dụng web. Nó được mô tả là “cách nhanh nhất để đi từ ý tưởng đến URL, bỏ qua mọi rắc rối về vấn đề cơ sở hạ tầng. Bạn chỉ việc tập trung vào ứng dụng của bạn”. [7]

PHẦN 2 – PHÂN TÍCH HỆ THỐNG

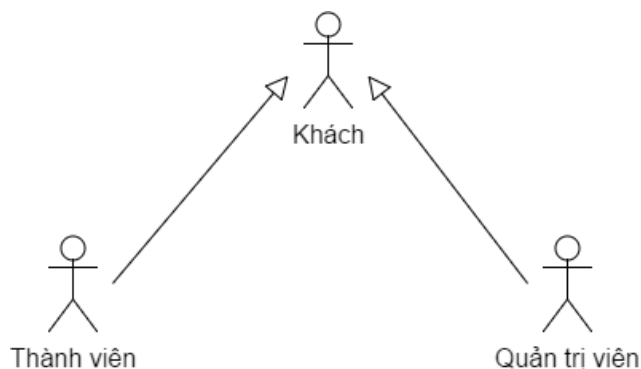
2.1. Phân tích yêu cầu tổng quát

2.1.1. Các tác nhân của hệ thống

Website đánh giá sách LoveBook có 3 tác nhân:

- Khách:
 - Là người chưa có tài khoản đăng nhập và có thể đăng kí tài khoản
 - Có nhu cầu tham khảo nội dung trang web
 - Muốn xem toàn bộ sách, tìm kiếm sách và xem chi tiết một cuốn sách cùng những bài đánh giá cuốn sách đó
- Thành viên:
 - Là người đã có tài khoản trên website
 - Ngoài những chức năng giống của khách, thành viên còn có những chức năng chính như: viết bài đánh giá sách, đánh dấu sách yêu thích, theo dõi những người dùng khác, báo cáo bài đánh giá có nội dung không phù hợp, gửi yêu cầu sách...
 - Ngoài ra thành viên cũng có thể có nhu cầu thay đổi và cập nhật thông tin cá nhân.
- Quản trị viên:
 - Quản lý thể loại sách: Thêm/sửa/xóa thể loại sách
 - Quản lý sách: Thêm/sửa/xóa sách
 - Quản lý thành viên: Xem danh sách thành viên, xóa thành viên
 - Quản lý bài đánh giá: Xem xét những bài đánh giá bị người dùng báo cáo và có thể xóa bài đánh giá nếu thấy cần thiết
 - Quản lý yêu cầu sách: Có thể đồng ý hoặc từ chối những yêu cầu sách của người dùng

Ba tác nhân trên của website có quan hệ kế thừa nhau về mặt chức năng. Thành viên và Quản trị viên kế thừa các chức năng của Khách.



Hình 7: Mối quan hệ của các tác nhân trong hệ thống

2.1.2. Biểu đồ ca sử dụng tổng quát



Hình 8: Biểu đồ ca sử dụng tổng quát của hệ thống

2.2. Phân tích yêu cầu chi tiết chức năng

2.2.1. Đăng kí

UC01: Đăng kí		
Mô tả: Khách hàng đăng kí tài khoản để trở thành thành viên.		
Tác nhân: Khách		
Điều kiện sử dụng: Không có		
Luồng sự kiện chính	Người dùng	Hệ thống
	1. Khách hàng yêu cầu đăng kí tài khoản	2. Hệ thống hiện thị form đăng kí, bao gồm: Họ và tên, địa chỉ email, mật khẩu và xác nhận mật khẩu
	3. Khách hàng điền thông tin.	4..Hệ thống kiểm tra: 4.1. Nếu khách hàng điền đầy đủ các thông tin, hệ thống thông báo đăng kí thành công. 4.2. Nếu khách hàng điền thiếu thông tin cần thiết hoặc phát hiện thông tin khách hàng điền đã tồn tại thì hệ thống thông báo lỗi.
	5. Khi hệ thống thông báo lỗi khách hàng lựa chọn: 5.1. Tiếp tục đăng kí. 5.2. Hủy quá trình đăng kí	6. Hệ thống thực hiện yêu cầu: 6.1. Quay lại bước 2. 6.2 Hệ thống hủy quá trình đăng kí.
Mức độ sử dụng:	Thường xuyên	

2.2.2. Đăng nhập

UC02: Đăng nhập		
Mô tả: Thực hiện chức năng đăng nhập vào hệ thống.		
Tác nhân: Thành viên, Quản trị viên.		
Điều kiện sử dụng: Đã có tài khoản trên hệ thống.		
Luồng sự kiện chính	Người dùng	Hệ thống
	1. Người dùng yêu cầu đăng nhập.	2. Hệ thống hiện thị form đăng nhập.
	3. Người dùng nhập thông tin: Email và Mật khẩu.	4. Hệ thống xác thực thông tin người dùng và cấp quyền. 4.1.Nếu thông tin là chính xác, hệ thống báo đăng nhập thành công. 4.2 Thông tin của người dùng không chính xác, hệ thống thông báo lỗi.
	5. Khi hệ thống thông báo lỗi, người dùng chọn: 5.1. Tiếp tục đăng nhập lại. 5.2. Bỏ đăng nhập.	6. Hệ thống thực hiện yêu cầu: 6.1. Quay lại bước 2. 6.2. Hệ thống hủy quá trình đăng nhập.
Mức độ sử dụng:	Thường xuyên	

2.2.3. Đăng xuất

UC03: Đăng xuất.
Mô tả: Người dùng thực hiện đăng xuất khỏi hệ thống.
Tác nhân: Thành viên, Quản trị viên
Điều kiện sử dụng: Người dùng đang trong trạng thái đăng nhập.

	Người dùng	Hệ thống
Luồng sự kiện chính	1. Người dùng lựa chọn đăng xuất.	2. Hệ thống thực hiện yêu cầu đăng xuất.
Mức độ sử dụng:	Thường xuyên	

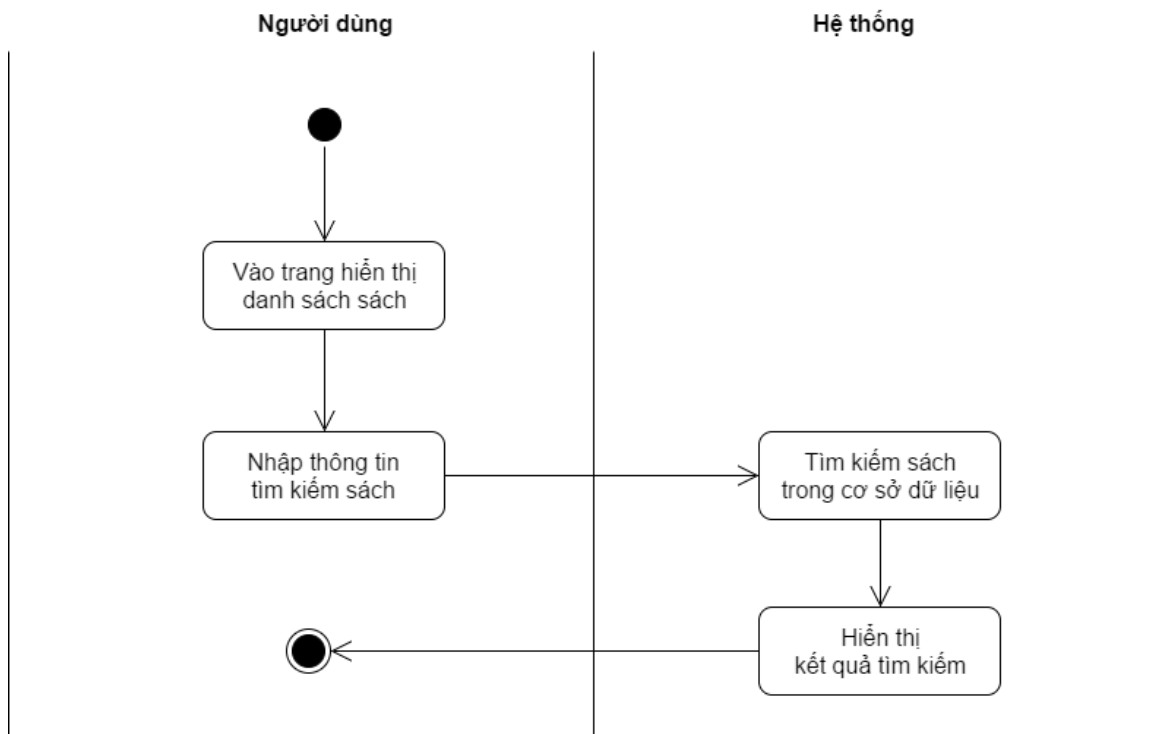
2.2.4. Xem danh sách sách

UC04: Xem danh sách sách		
Mô tả: Người dùng có thể xem toàn bộ sách trên hệ thống		
Tác nhân: Quản trị viên, Thành viên, Khách.		
Điều kiện sử dụng: Không có.		
	Người dùng	Hệ thống
Luồng sự kiện chính	1. Người dùng chọn xem toàn bộ sách.	2. Hệ thống hiển thị danh sách toàn bộ sách có trong cơ sở dữ liệu.
Mức độ sử dụng:	Thường xuyên.	

2.2.5. Tìm kiếm sách

UC05: Tìm kiếm sách		
Mô tả: Người dùng có thể tìm kiếm sách theo tên sách, tên tác giả hoặc theo thể loại.		
Tác nhân: Quản trị viên, Thành viên, Khách.		
Điều kiện sử dụng: Không có.		
	Người dùng	Hệ thống
Luồng sự kiện chính	1. Người dùng nhập thông tin tìm kiếm sách.	2. Hệ thống kiểm tra thông tin người dùng đã điền.

	Hiển thị các kết quả phù hợp với thông tin tìm kiếm của người dùng.
Mức độ sử dụng:	Thường xuyên.



Hình 9: Biểu đồ hoạt động chức năng “Tìm kiếm sách”

2.2.6. Xem chi tiết một cuốn sách

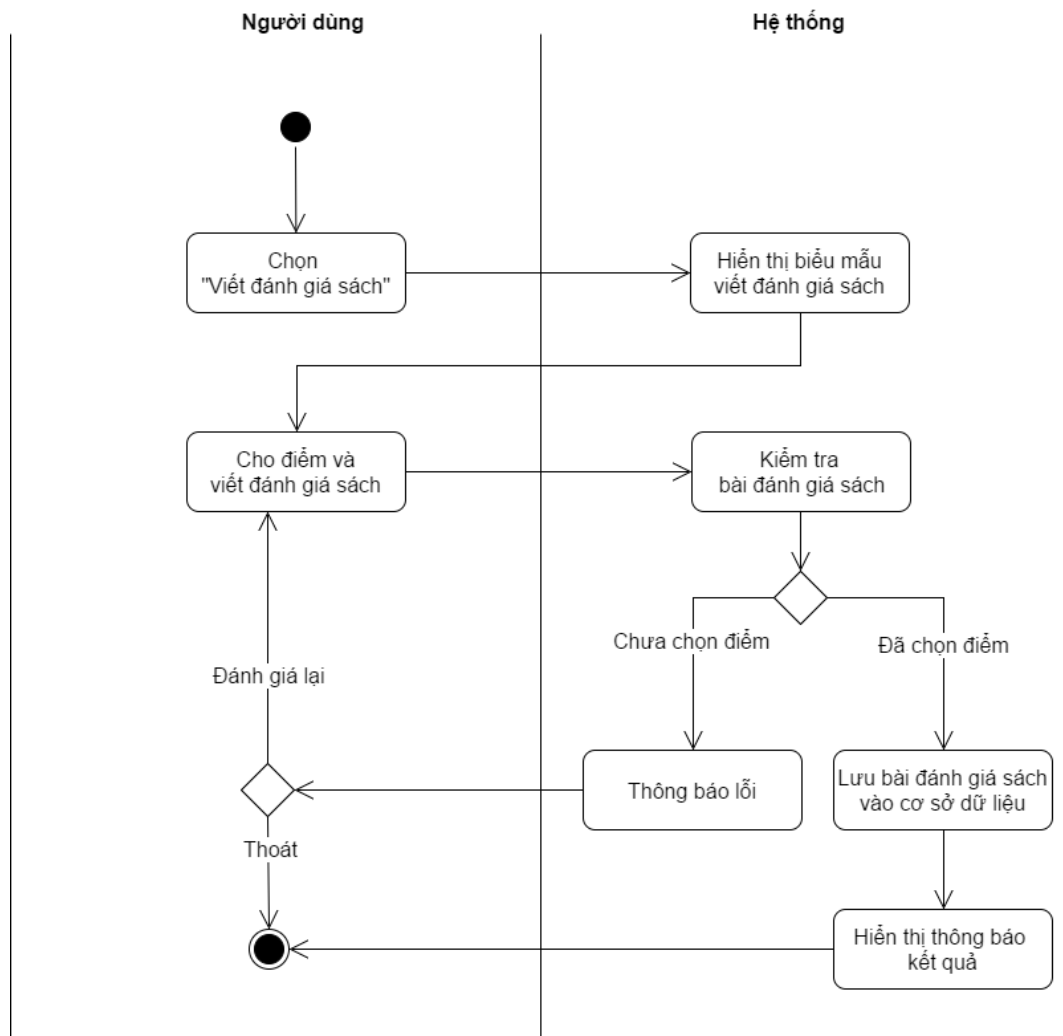
UC06: Xem chi tiết một cuốn sách		
Mô tả: Người dùng có thể xem chi tiết thông tin một cuốn sách cùng các bài đánh giá, bình luận.		
Tác nhân: Quản trị viên, Thành viên, Khách.		
Điều kiện sử dụng: Không có.		
Luồng sự kiện chính	Người dùng	Hệ thống

	1. Người dùng chọn một cuốn sách muốn xem chi tiết.	2. Hệ thống hiển thị thông tin chi tiết về cuốn sách cùng những bài đánh giá về cuốn sách đó.
Mức độ sử dụng:	Thường xuyên.	

2.2.7. Viết bài đánh giá sách

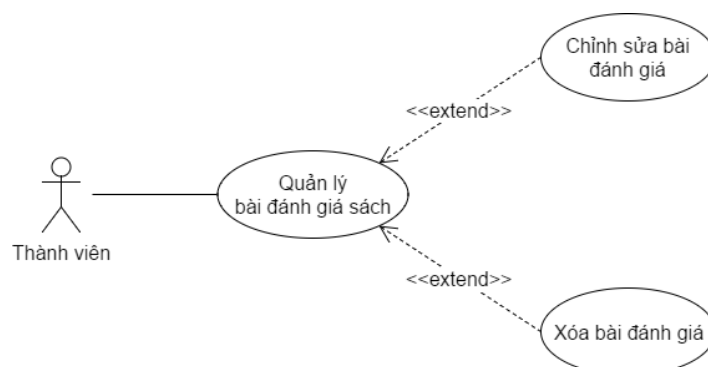
UC07: Viết bài đánh giá sách		
Mô tả: Người dùng có thể viết bài đánh giá một cuốn sách sau khi xem thông tin chi tiết cuốn sách đó.		
Tác nhân: Quản trị viên, Thành viên.		
Điều kiện sử dụng: Đăng nhập vào hệ thống.		
Luồng sự kiện chính	Người dùng	Hệ thống
	1. Người dùng yêu cầu viết bài đánh giá sách.	2. Hệ thống hiển thị form viết bài đánh giá sách gồm: điểm và nội dung đánh giá.
	3. Người dùng chấm điểm sách và viết bài đánh giá	4. Hệ thống kiểm tra nội dung người dùng đã nhập 4.1. Nếu nội dung có điểm, hệ thống lưu bài đánh giá, cập nhật điểm cho cuốn sách và quay lại trang hiển thị sách. 4.2. Nếu nội dung không có điểm, hệ thống hiển thị thông báo lỗi cho người dùng
	5. Khi có thông báo lỗi, người dùng lựa chọn 5.1. Tiếp tục viết lại bài đánh giá.	6. Hệ thống xử lý yêu cầu người dùng: 6.1. Quay lại bước 2

	5.2. Hủy quá trình viết bài đánh giá	6.2. Hủy quá trình viết bài đánh giá
Mức độ sử dụng:	Thường xuyên.	



Hình 10: Biểu đồ hoạt động chức năng “Viết bài đánh giá sách”

2.2.8. Quản lý bài đánh giá sách

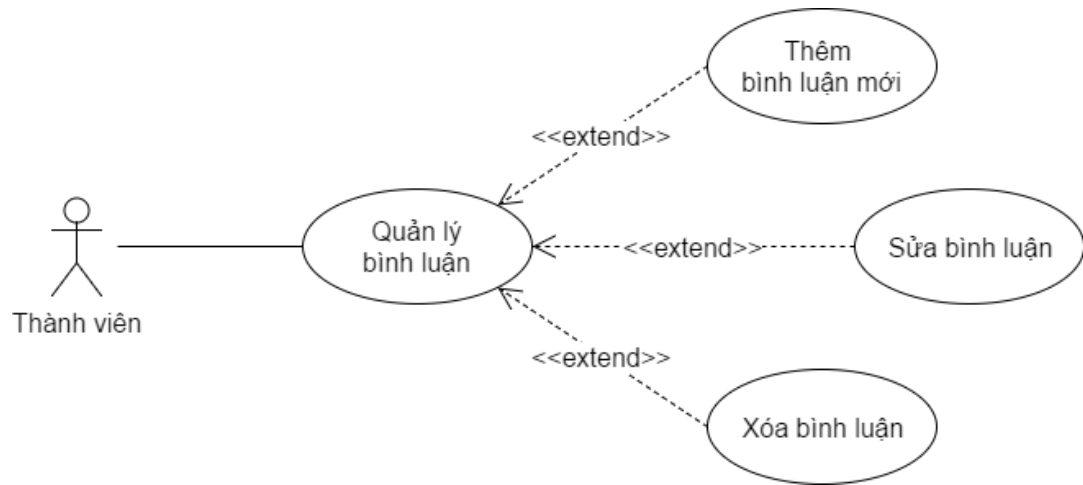


Hình 11: Biểu đồ phân rã usecase “Quản lý bài đánh giá sách”

UC08: Quản lý bài đánh giá sách		
Mô tả: Người dùng có thể chỉnh sửa/xóa các bài đánh giá sách của mình		
Tác nhân: Quản trị viên, Thành viên.		
Điều kiện sử dụng: Đăng nhập vào hệ thống.		
Luồng sự kiện chính	Người dùng	Hệ thống
	Chỉnh sửa bài đánh giá	
	1. Người dùng yêu cầu chỉnh sửa đánh giá sách.	2. Hệ thống hiển thị form chỉnh sửa đánh giá sách
	3. Người dùng chấm điểm sách và viết bài đánh giá	4. Hệ thống kiểm tra nội dung người dùng đã nhập 4.1. Nếu nội dung hợp lệ, hệ thống cập nhật bài đánh giá, cập nhật điểm cho cuốn sách và quay lại trang hiển thị sách. 4.2. Nếu nội dung không hợp lệ, hệ thống hiển thị thông báo lỗi cho người dùng
	5. Khi có thông báo lỗi, người dùng lựa chọn 5.1. Viết lại bài đánh giá. 5.2. Hủy quá trình viết bài đánh giá	6. Hệ thống xử lý yêu cầu người dùng: 6.1. Quay lại bước 2 6.2. Hủy quá trình viết bài đánh giá
	Xóa bài đánh giá	
	1. Người dùng yêu cầu xóa bài đánh giá	2. Hiển thị thông báo xác nhận xóa bài đánh giá
	3. Người dùng lựa chọn: 3.1. Xác nhận xóa 3.2. Hủy xóa	4. Hệ thống thực hiện yêu cầu của người dùng: 4.1. Xóa bài đánh giá 4.2. Hủy xóa

Mức độ sử dụng:	Thường xuyên.
-----------------	---------------

2.2.9. Quản lý bình luận



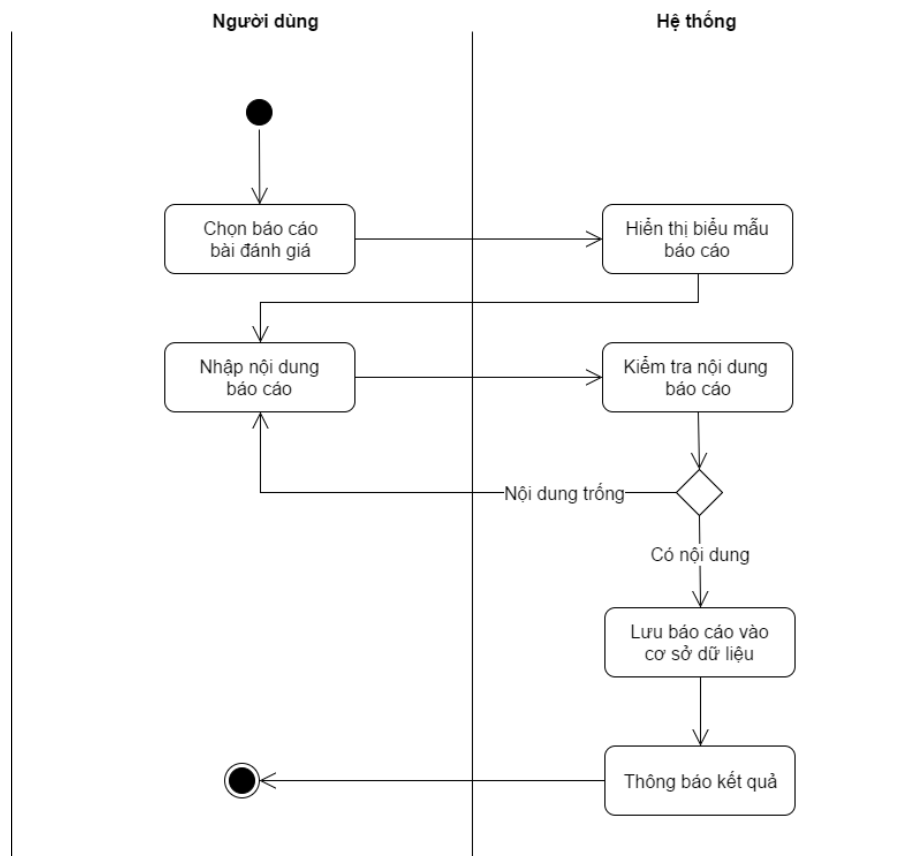
Hình 12: Biểu đồ phân rã usecase “Quản lý bình luận”

UC09: Quản lý bình luận		
Mô tả: Người dùng có thể viết bình luận cho một bài đánh giá sách.		
Tác nhân: Quản trị viên, Thành viên.		
Điều kiện sử dụng: Đăng nhập vào hệ thống. Xem chi tiết một bài đánh giá		
Luồng sự kiện chính	Người dùng	Hệ thống
	Viết bình luận mới	
	1. Người dùng viết bình luận dưới bài đánh giá.	2. Hệ thống kiểm tra bình luận 2.1. Bình luận hợp lệ, hệ thống lưu bình luận vào cơ sở dữ liệu và hiển thị bình luận mới 2.2. Nếu bình luận không hợp lệ, hệ thống cho phép người dùng nhập lại.
	Chỉnh sửa bình luận	
	1. Người dùng yêu cầu chỉnh sửa bình luận	2. Hiển thị form cập nhật bình luận

	3. Người dùng cập nhật bình luận	4. Hệ thống kiểm tra bình luận 4.1. Bình luận hợp lệ, hệ thống cập nhật bình luận vào cơ sở dữ liệu và hiển thị lại bình luận 4.2. Nếu bình luận không hợp lệ, hệ thống cho phép người dùng nhập lại.
	Xóa bình luận	
	1. Người dùng yêu cầu xóa bình luận của mình. 3. Người dùng lựa chọn 3.1. Xác nhận xóa 3.2. Hủy xóa bình luận	2. Hệ thống hiển thị thông báo xác nhận xóa bình luận 4. Hệ thống xử lý yêu cầu: 4.1. Xóa bình luận 4.2. Hủy xóa bình luận
Mức độ sử dụng:	Thường xuyên.	

2.2.10. Báo cáo bài đánh giá sách

UC10: Báo cáo bài đánh giá sách		
Mô tả: Người dùng có thể báo cáo cho Quản trị viên những bài đánh giá có nội dung không phù hợp		
Tác nhân: Thành viên.		
Điều kiện sử dụng: Đăng nhập vào hệ thống và xem chi tiết một bài đánh giá		
Luồng sự kiện chính	Người dùng	Hệ thống
	1. Người dùng báo cáo bài đánh giá. 3. Người dùng viết lý do báo cáo bài đánh giá vi phạm và gửi lên hệ thống	2. Hệ thống hiển thị form cho người dùng viết lý do báo cáo bài đánh giá 4. Hệ thống kiểm tra nội dung: 4.1. Nếu nội dung đầy đủ, hệ thống hiển thị thông báo thành công 4.2. Nếu nội dung không đầy đủ, hệ thống cho phép người dùng nhập lại
Mức độ sử dụng:	Thỉnh thoảng.	

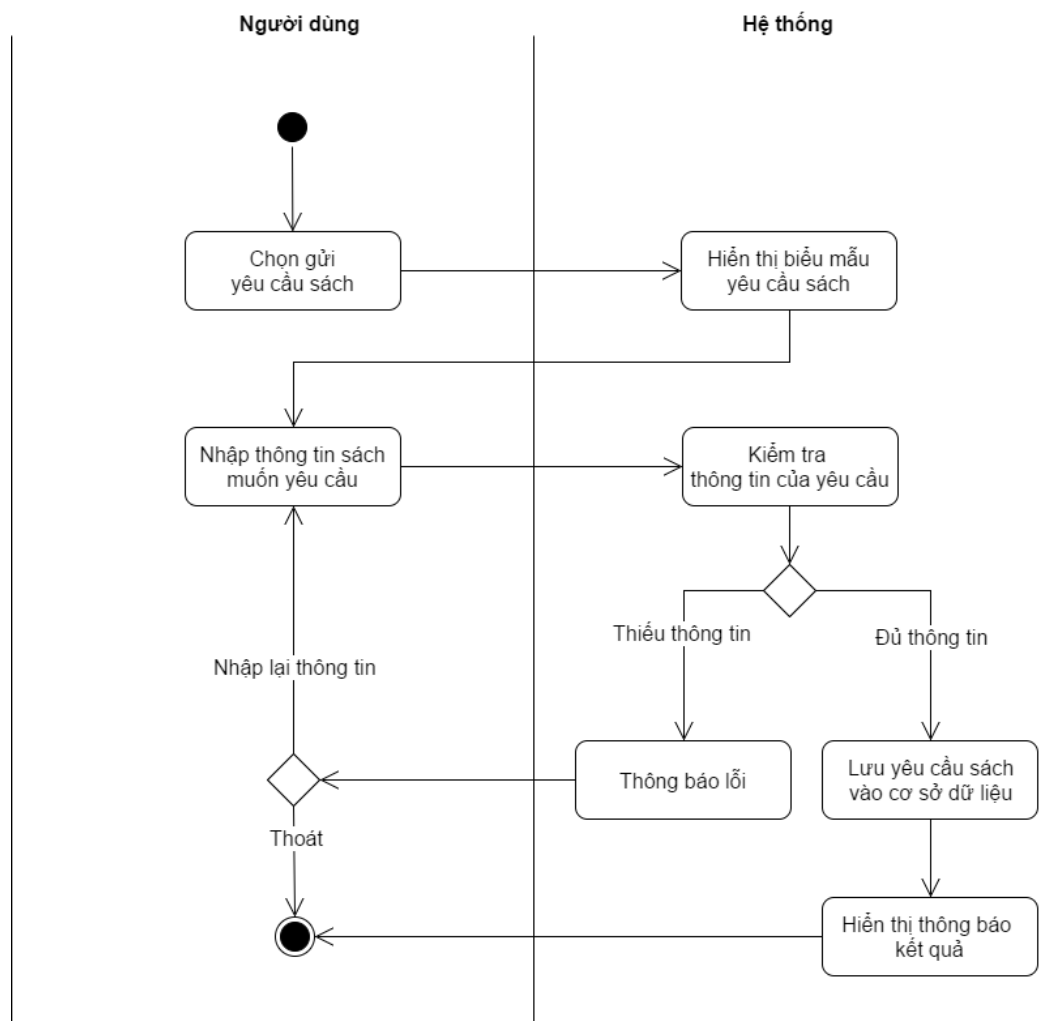


Hình 14: Biểu đồ hoạt động chức năng “Báo cáo bài đánh giá”

2.2.11. Gửi yêu cầu sách

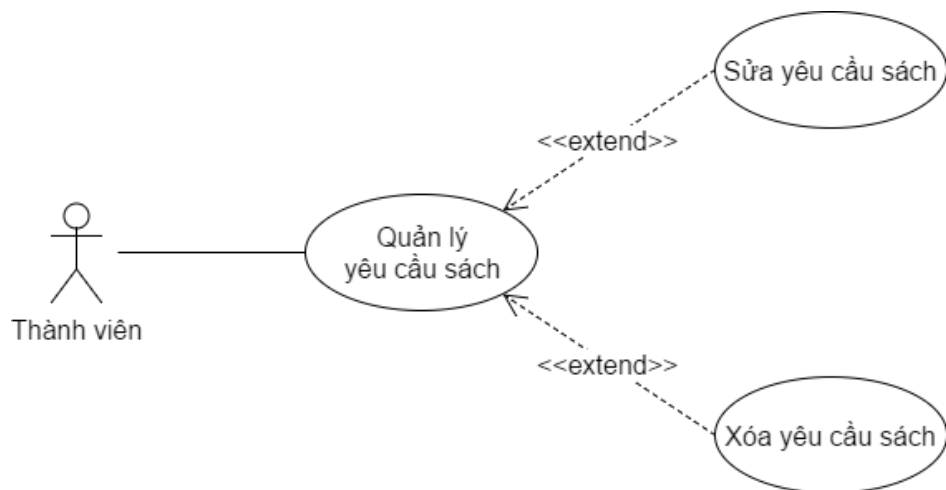
UC11: Gửi yêu cầu sách		
Mô tả: Người dùng có thể gửi cho Quản trị viên yêu cầu cập nhật những quyển sách chưa có trong hệ thống		
Tác nhân: Thành viên.		
Điều kiện sử dụng: Đăng nhập vào hệ thống		
Luồng sự kiện chính	Người dùng	Hệ thống
	1. Người dùng lựa chọn yêu cầu sách	2. Hệ thống hiển thị form cho người dùng viết thông tin sách muốn yêu cầu: Tên sách và tên tác giả.

	<p>3. Người dùng điền thông tin yêu cầu sách và gửi lên hệ thống</p> <p>5. Nếu bị lỗi, người dùng có thể lựa chọn</p> <p>5.1. Viết lại yêu cầu</p> <p>5.2. Hủy yêu cầu</p>	<p>4. Hệ thống kiểm tra nội dung người dùng đã điền</p> <p>4.1. Nếu nội dung đầy đủ, hệ thống hiển thị thông báo thành công</p> <p>4.2. Nếu nội dung không đầy đủ, hệ thống hiển thị thông báo lỗi</p> <p>6. Hệ thống xử lý yêu cầu người dùng</p> <p>6.1. Quay lại bước 2</p> <p>6.2. Hủy yêu cầu</p>
Mức độ sử dụng:	Thỉnh thoảng.	



Hình 15: Biểu đồ hoạt động chức năng “Gửi yêu cầu sách”

2.2.12. Quản lý yêu cầu sách



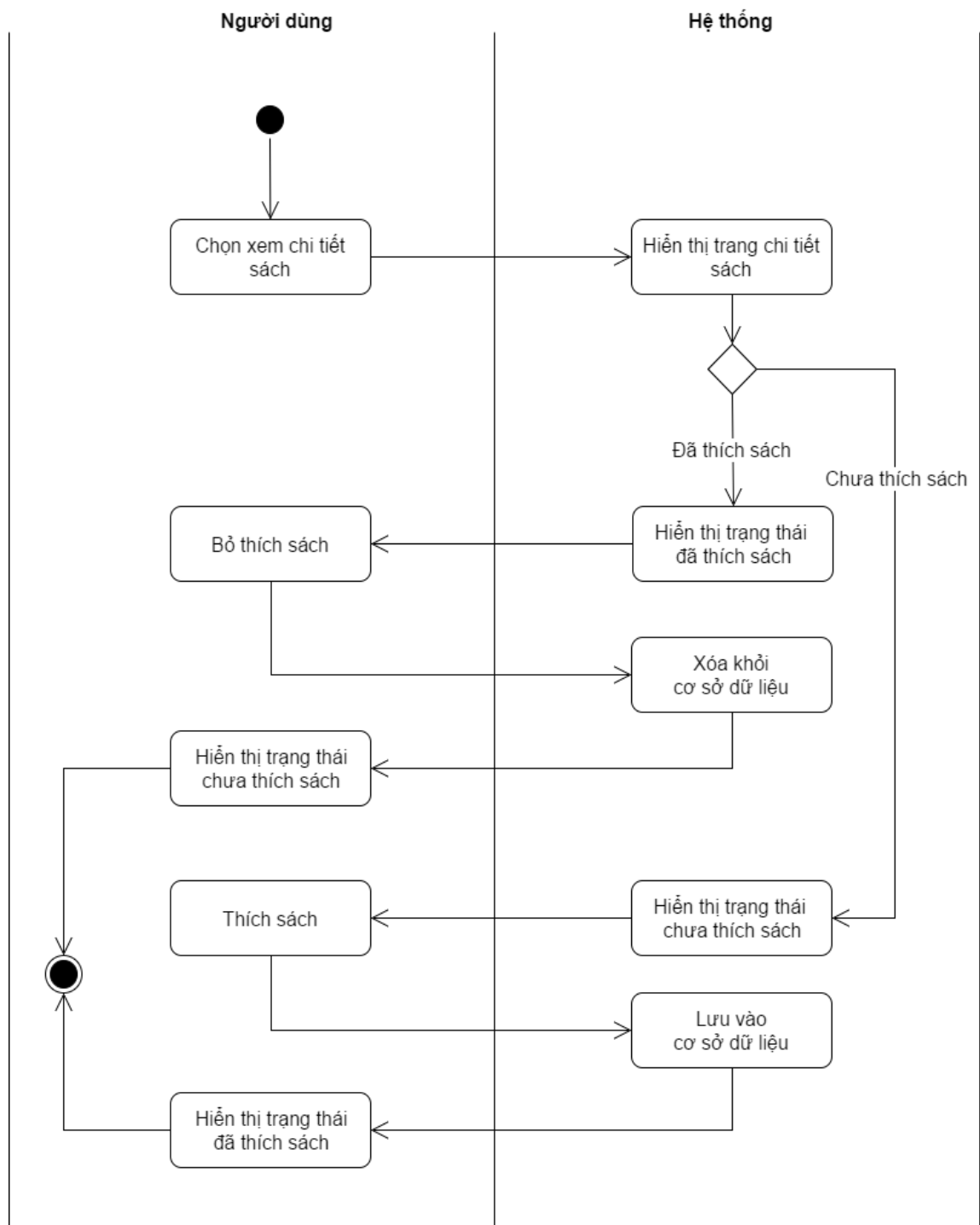
Hình 16: Biểu đồ phân rã usecase “Quản lý yêu cầu sách”

UC12: Quản lý yêu cầu sách		
Mô tả: Người dùng có thể xem lại những yêu cầu sách của mình để có thể chỉnh sửa/xóa những yêu cầu có trạng thái là đang chờ		
Tác nhân: Thành viên.		
Điều kiện sử dụng: Đăng nhập vào hệ thống và xem chi tiết một bài đánh giá		
Luồng sự kiện chính	Người dùng	Hệ thống
	Chỉnh sửa yêu cầu	
	1. Chọn chỉnh sửa yêu cầu	2. Hiện thị form chỉnh sửa yêu cầu
	3. Người dùng điền thông tin muốn thay đổi	4. Hệ thống xử lý thông tin người dùng gửi lên 4.1. Thông tin đầy đủ, hệ thống cập nhật yêu cầu và hiển thị thông báo thành công 4.2. Thông tin không đầy đủ, hệ thống hiển thị thông báo lỗi và cho phép người dùng nhập lại hoặc hủy chỉnh sửa
Xóa yêu cầu		

	1. Người dùng yêu cầu xóa yêu cầu của mình. 3. Người dùng lựa chọn 3.1. Xác nhận xóa 3.2. Hủy xóa	2. Hệ thống hiển thị thông báo xác nhận xóa yêu cầu 4. Hệ thống xử lý yêu cầu: 4.1. Xóa yêu cầu 4.2. Hủy xóa
Mức độ sử dụng:	Thỉnh thoảng.	

2.2.13. Đánh dấu sách yêu thích

UC13: Đánh dấu sách yêu thích		
Mô tả: Người dùng có thể đánh giá một quyển sách làm sách yêu thích		
Tác nhân: Thành viên.		
Điều kiện sử dụng: Đăng nhập vào hệ thống và xem chi tiết một cuốn sách		
Luồng sự kiện chính	Người dùng	Hệ thống
	Đánh dấu yêu thích sách	
	1. Người dùng bấm vào nút yêu thích	2. Hệ thống hiển thị nút bấm chuyển sang màu đỏ và số lượng người thích cuốn sách đó tăng lên 1
	Bỏ yêu thích sách	
	1. Nếu không thích cuốn sách nào đó nữa, người dùng có thể bỏ thích cuốn sách	2. Hệ thống hiển thị nút về trạng thái ban đầu, số lượng người thích cuốn sách giảm đi 1
Mức độ sử dụng:	Thường xuyên.	



Hình 17: Biểu đồ hoạt động chức năng “Đánh dấu sách yêu thích”

2.2.14. Xem danh sách thành viên

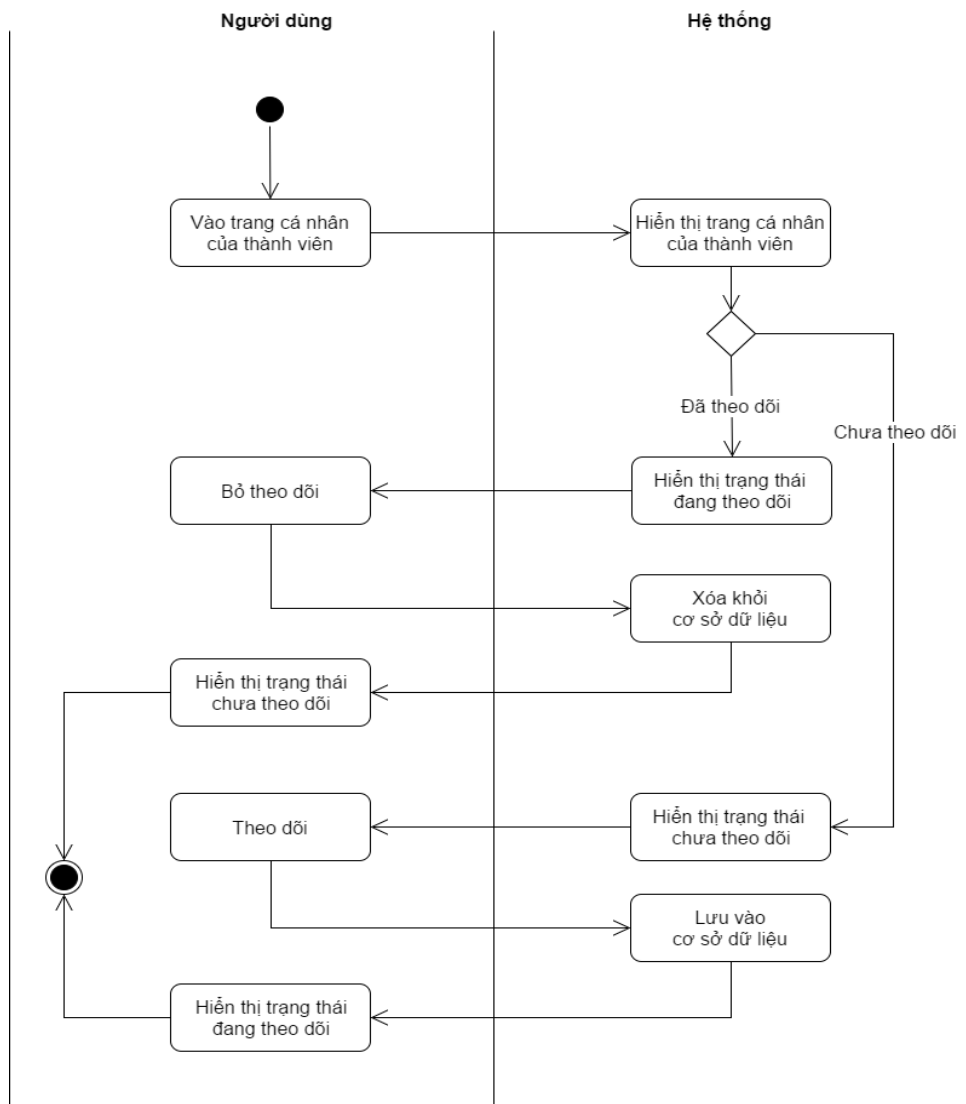
UC14: Xem danh sách thành viên		
Mô tả: Người dùng có thể xem danh sách các thành viên có trong hệ thống		
Tác nhân: Thành viên.		
Điều kiện sử dụng: Đăng nhập vào hệ thống		
Luồng sự kiện chính	Người dùng	Hệ thống
	1. Người dùng chọn xem các thành viên. 3. Người dùng lựa chọn các trang của danh sách	2. Hệ thống hiển thị danh sách các thành viên 4. Hệ thống hiển thị các thành viên theo trang người dùng lựa chọn
Mức độ sử dụng:	Thỉnh thoảng.	

2.2.15. Xem thông tin thành viên

UC15: Xem thông tin thành viên		
Mô tả: Người dùng có thể lựa chọn một thành viên để xem thông tin chi tiết		
Tác nhân: Thành viên.		
Điều kiện sử dụng: Đăng nhập vào hệ thống		
Luồng sự kiện chính	Người dùng	Hệ thống
	1. Người dùng chọn một thành viên để xem thông tin chi tiết	2. Hệ thống hiển thị thông tin thành viên được lựa chọn, bao gồm những cuốn sách yêu thích, những bài đánh giá họ đã viết, số lượng người theo dõi và đang theo dõi.
Mức độ sử dụng:	Thỉnh thoảng.	

2.2.16. Theo dõi thành viên

UC16: Theo dõi thành viên		
Mô tả: Người dùng có thể theo dõi người dùng khác để cập nhật những bài đánh giá của người đấy viết, hoặc cũng có thể bỏ theo dõi một người dùng đang theo dõi		
Tác nhân: Thành viên.		
Điều kiện sử dụng: Đăng nhập vào hệ thống, vào trang cá nhân của một thành viên		
Luồng sự kiện chính	Người dùng	Hệ thống
	Theo dõi thành viên	
	1. Bấm chọn “Theo dõi”	2. Hiện thị nút “Bỏ theo dõi”, số lượng người theo dõi tăng lên 1
	Bỏ theo dõi thành viên	
	1. Bấm chọn “Bỏ theo dõi”	2. Hiện thị nút “Theo dõi”, số lượng người theo dõi giảm đi 1
Mức độ sử dụng:	Thỉnh thoảng.	



Hình 18: Biểu đồ hoạt động chức năng “Theo dõi thành viên”

2.2.17. Quản lý thông tin cá nhân

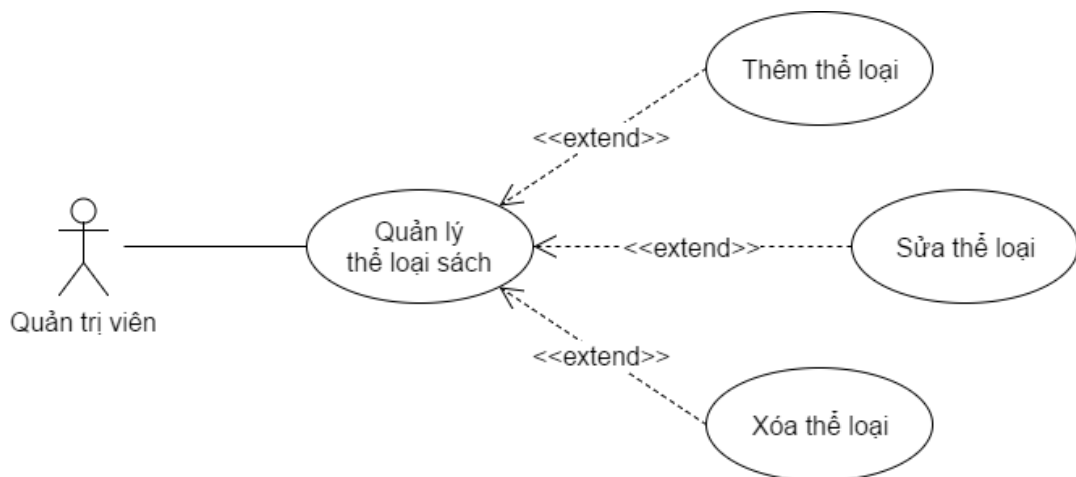
UC17: Quản lý thông tin cá nhân		
Mô tả: Thành viên thay đổi thông tin tài khoản của mình		
Tác nhân: Thành viên.		
Điều kiện sử dụng: Đăng nhập với quyền là thành viên.		
Luồng sự kiện chính	Người dùng	Hệ thống
	1. Người dùng yêu cầu thay đổi thông tin của mình	2. Hệ thống hiển thị form thông tin tài khoản

	<p>3. Người dùng thay đổi các trường muốn thay đổi và lưu lại.</p> <p>5. Khi hệ thống báo lỗi, người dùng lựa chọn:</p> <p>5.1. Hủy quá trình thay đổi.</p> <p>5.2. Tiếp tục thay đổi thông tin tài khoản.</p>	<p>4..Hệ thống kiểm tra thông tin người dùng đã nhập vào.</p> <p>4.1. Nếu thông tin nhập vào đúng định dạng, hệ thống thông báo thay đổi thành công.</p> <p>4.2.Nếu thông tin nhập vào là sai, hệ thống thông báo lỗi.</p> <p>6. Hệ thống thực hiện yêu cầu:</p> <p>6.1. Hủy quá trình thay đổi</p> <p>6.2. Quay lại bước 2.</p>
Mức độ sử dụng:	Hiếm khi	

2.2.18. Quản lý thành viên

UC18: Quản lý thành viên		
Mô tả: Quản trị viên có thể xóa tài khoản những thành viên không phù hợp		
Tác nhân: Quản trị viên		
Điều kiện sử dụng: Đăng nhập với quyền Quản trị viên		
Luồng sự kiện chính	Người dùng	Hệ thống
	<p>1. Chọn tài khoản cần xóa.</p> <p>3. Quản trị viên lựa chọn yêu cầu thực hiện:</p> <p>3.1. Xác nhận xóa.</p> <p>3.2. Lựa chọn hủy xóa thành viên</p>	<p>2. Hệ thống yêu cầu xác nhận xóa tài khoản.</p> <p>4. Hệ thống thực hiện yêu cầu của Quản trị viên:</p> <p>4.1. Hệ thống xóa tài khoản khỏi cơ sở dữ liệu.</p> <p>4.2. Hệ thống hủy yêu cầu xóa.</p>
Mức độ sử dụng	Hiếm khi	

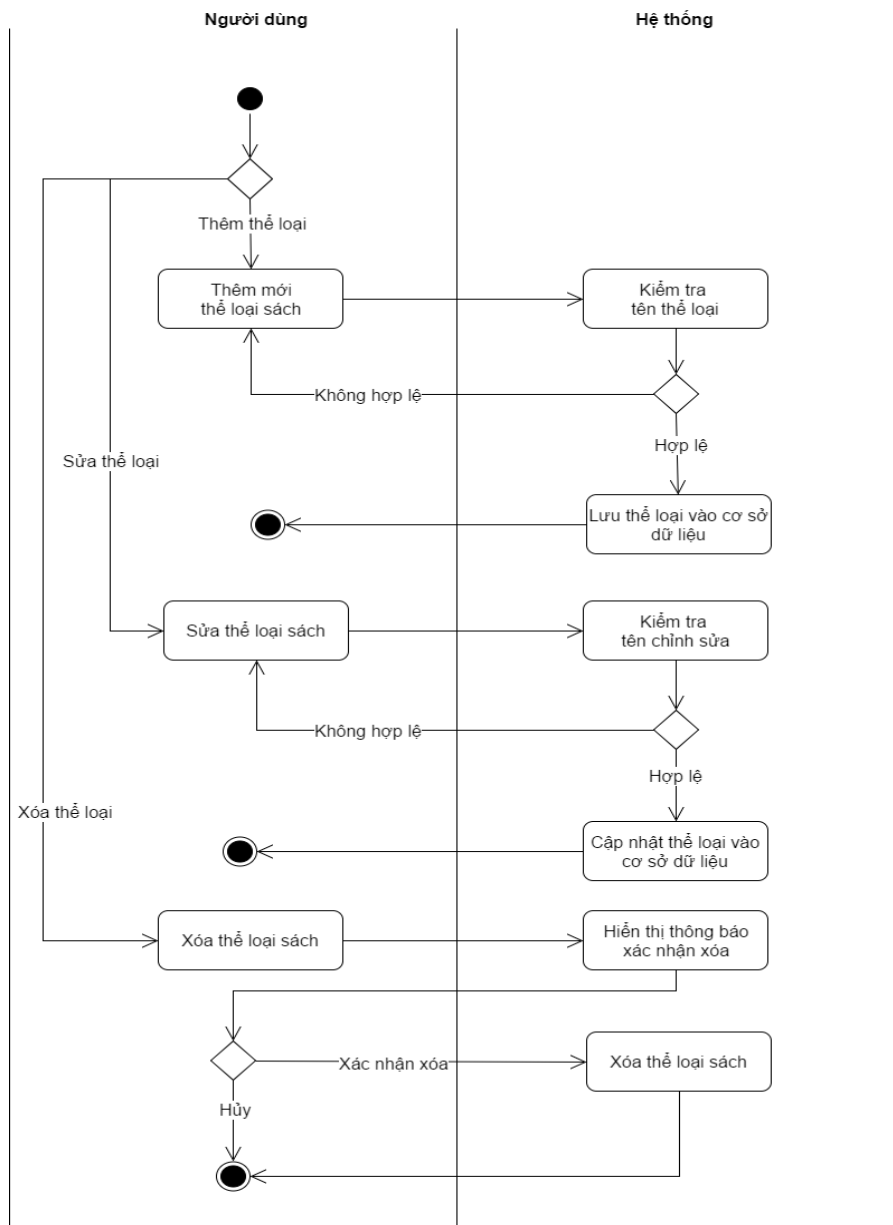
2.2.19. Quản lý thẻ loại sách



Hình 19: Biểu đồ phân rã usecase “Quản lý thẻ loại sách”

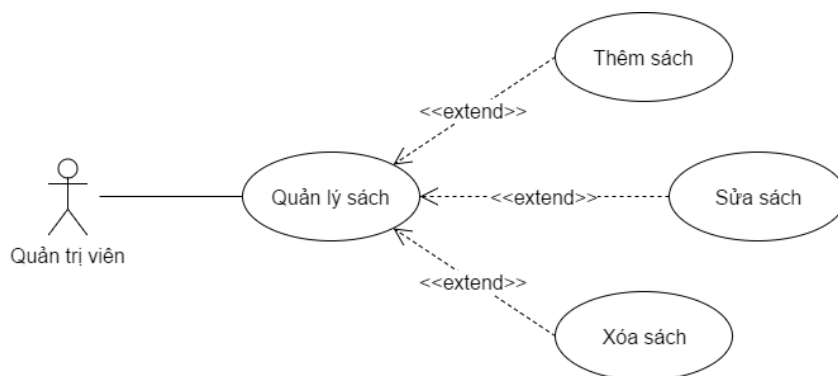
UC19: Quản lý thẻ loại sách		
Mô tả: Quản trị viên có thể thêm/sửa/xóa các thẻ loại sách		
Tác nhân: Quản trị viên.		
Điều kiện sử dụng: Đăng nhập vào hệ thống dưới quyền quản trị viên		
Luồng sự kiện chính	Người dùng	Hệ thống
	Thêm thẻ loại mới	
	1. Quản trị viên nhập tên thẻ loại sách mới	2. Hệ thống kiểm tra thông tin quản trị viên đã nhập 2.1. Nếu thông tin nhập hợp lệ, lưu thẻ loại mới vào cơ sở dữ liệu 2.2. Nếu thông tin không hợp lệ, cho phép nhập lại
	Chỉnh sửa thẻ loại	
	1. Quản trị viên có thể chọn chỉnh sửa một thẻ loại sách 3. Nhập thông tin để chỉnh sửa	2. Hệ thống hiển thị form cho quản trị viên chỉnh sửa 4. Hệ thống kiểm tra và hiển thị thông báo

		<p>4.1. Thông tin thay đổi hợp lệ, cập nhật tên thẻ loại vào cơ sở dữ liệu</p> <p>4.2. Thông tin không hợp lệ, cho phép nhập lại</p>
	Xóa thẻ loại	
	<p>1. Quản trị viên chọn xóa một thẻ loại</p> <p>3. Quản trị viên lựa chọn</p> <p>3.1. Xác nhận xóa</p> <p>3.2. Hủy xóa</p>	<p>2. Hệ thống hiển thị thông báo xác nhận xóa</p> <p>4. Xử lý yêu cầu</p> <p>4.1. Xóa thẻ loại</p> <p>4.2. Hủy xóa</p>
Mức độ sử dụng:	Thỉnh thoảng.	



Hình 20: Biểu đồ hoạt động chức năng “Quản lý thẻ loại sách”

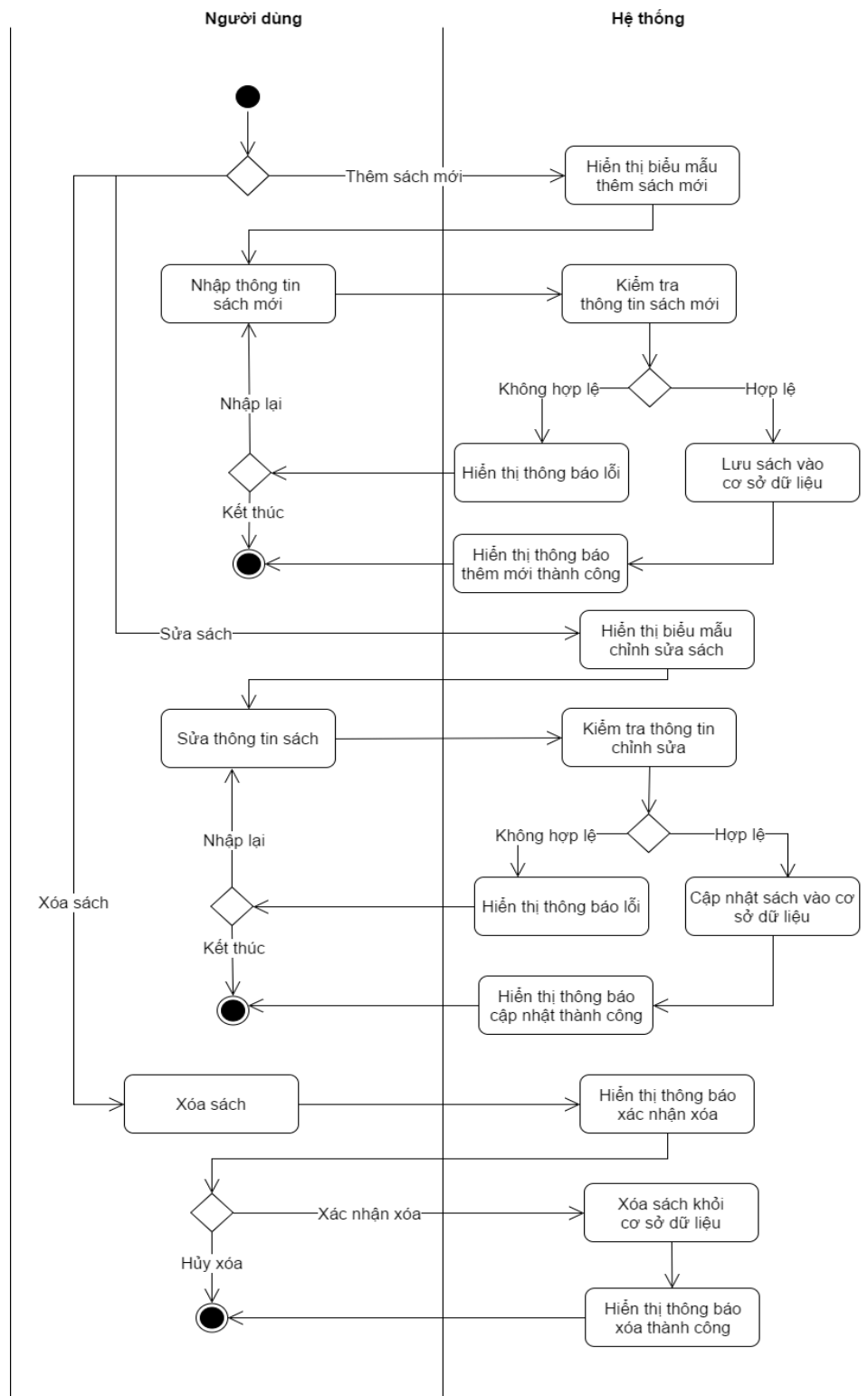
2.2.20. Quản lý sách



Hình 21: Biểu đồ phân rã usecase “Quản lý sách”

UC20: Quản lý sách		
Mô tả: Quản trị viên có thể thêm/sửa/xóa sách		
Tác nhân: Quản trị viên.		
Điều kiện sử dụng: Đăng nhập vào hệ thống dưới quyền quản trị viên		
Luồng sự kiện chính	Người dùng	Hệ thống
	Thêm sách mới	
	1. Quản trị viên lựa chọn thêm sách mới	2. Hệ thống hiển thị form nhập thông tin sách mới
	3. Nhập thông tin sách mới	4. Hệ thống kiểm tra thông tin quản trị viên đã nhập 2.1. Nếu thông tin nhập hợp lệ, lưu sách mới vào cơ sở dữ liệu 2.2. Nếu thông tin không hợp lệ, hiển thị thông báo lỗi và cho phép nhập lại
	Chỉnh sửa sách	
	1. Quản trị viên có thể chọn chỉnh sửa một cuốn sách	2. Hệ thống hiển thị form cho quản trị viên chỉnh sửa
	3. Nhập thông tin để chỉnh sửa	4. Hệ thống kiểm tra và hiển thị thông báo 4.1. Thông tin thay đổi hợp lệ, cập nhật thông tin sách vào cơ sở dữ liệu 4.2. Thông tin không hợp lệ, hiển thị thông báo lỗi và cho phép nhập lại
	Xóa sách	
	1. Quản trị viên chọn xóa một cuốn sách	2. Hệ thống hiển thị thông báo xác nhận xóa

	3. Quản trị viên lựa chọn 3.1. Xác nhận xóa 3.2. Hủy xóa	4. Xử lý yêu cầu 4.1. Xóa thể loại 4.2. Hủy xóa
Mức độ sử dụng:	Thỉnh thoảng.	

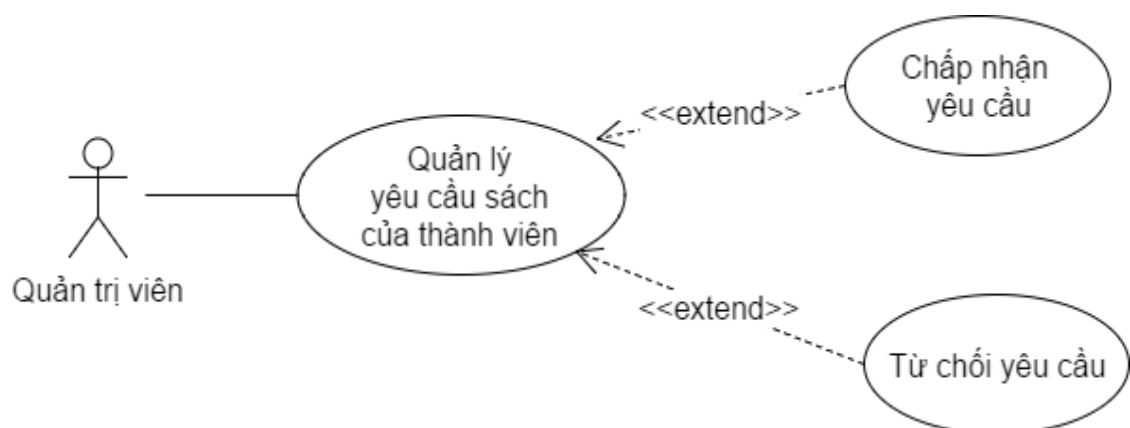


Hình 22: Biểu đồ hoạt động chức năng “Quản lý sách”

2.2.21. Quản lý báo cáo đánh giá sách

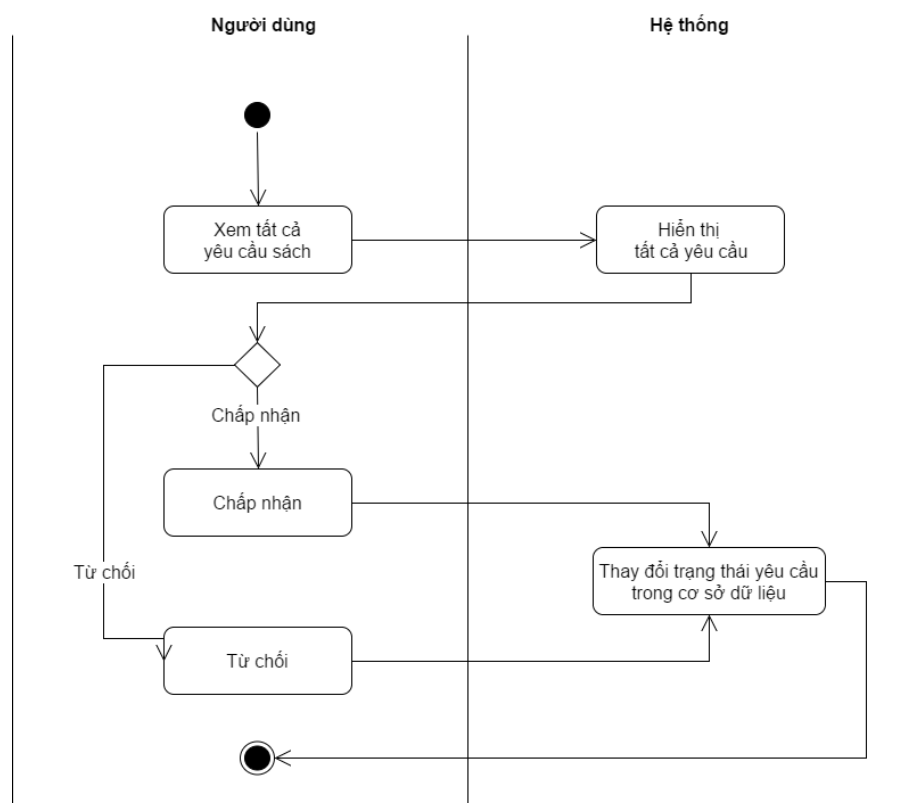
UC21: Quản lý báo cáo đánh giá sách		
Mô tả: Quản trị viên có thể xem/xóa các bài đánh giá		
Tác nhân: Quản trị viên.		
Điều kiện sử dụng: Đăng nhập vào hệ thống với quyền quản trị viên		
	Người dùng	Hệ thống
Luồng sự kiện chính	1. Quản trị viên yêu cầu xem các báo cáo vi phạm	2. Hệ thống hiển thị danh sách các báo cáo vi phạm
	3. Quản trị viên có thể chọn xem chi tiết một bài đánh giá bị báo cáo	4. Hệ thống hiển thị bài đánh giá bị báo cáo
	5. Sau khi xem, quản trị viên có thể xóa bài đánh giá 5.1. Xác nhận xóa 5.2. Hủy xóa	6. Hệ thống hiển thị thông báo xác nhận xóa 6.1. Xóa thể loại 6.2. Hủy xóa
Mức độ sử dụng:	Thỉnh thoảng.	

2.2.22. Quản lý yêu cầu sách của thành viên



Hình 23: Biểu đồ phân rã usecase “Quản lý yêu cầu sách của thành viên”

UC22: Quản lý yêu cầu sách của thành viên		
Mô tả: Quản trị viên có xem/chấp nhận/từ chối các yêu cầu sách của thành viên		
Tác nhân: Quản trị viên.		
Điều kiện sử dụng: Đăng nhập vào hệ thống với quyền Quản trị viên		
Luồng sự kiện chính	Người dùng	Hệ thống
	1. Quản trị viên yêu cầu xem các yêu cầu của thành viên 3. Sau khi xem, Quản trị viên có thể chấp nhận/từ chối yêu cầu 3.1. Chấp nhận yêu cầu 3.2. Từ chối yêu cầu	2. Hệ thống hiển thị danh sách các yêu cầu 4. Hệ thống thực hiện yêu cầu của Quản trị viên 4.1. Trạng thái yêu cầu chuyển thành “Chấp nhận” 4.2. Trạng thái yêu cầu chuyển thành “Từ chối”
Mức độ sử dụng:	Bình thường.	



Hình 24: Biểu đồ hoạt động chức năng “Quản lý yêu cầu sách của thành viên”

2.3. Yêu cầu phi chức năng

- Giao diện thuận tiện cho người dùng.
- Hiệu năng tốt, thời gian xử lý yêu cầu nhanh
- Hỗ trợ tốt trên các hệ điều hành khác nhau (Windows, MacOS, Ubuntu) và các trình duyệt khác nhau (Google Chrome, Cốc Cốc, Firefox...)

PHẦN 3 – THIẾT KẾ HỆ THỐNG

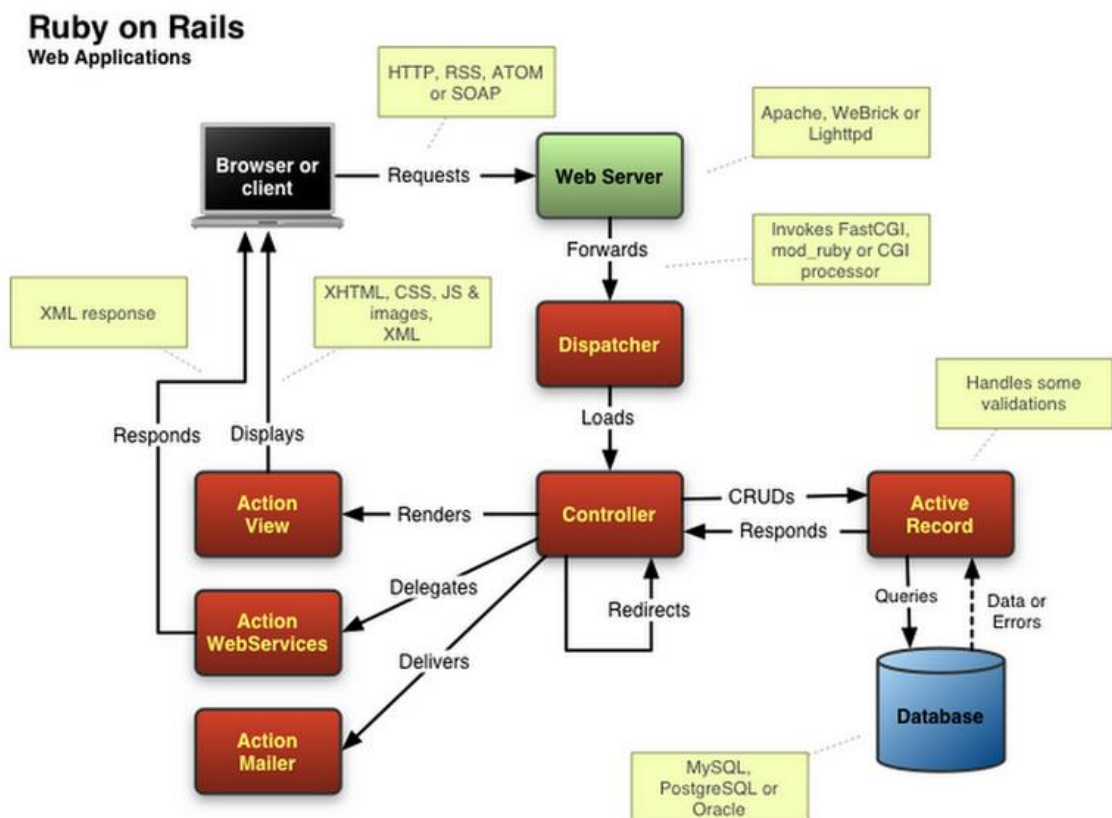
3.1. Kiến trúc Ruby on Rails

Kiến trúc của Ruby on Rails gồm những yếu tố sau:

- Kiến trúc MVC
- Representational State Transfer (REST)
- Hỗ trợ nhiều hệ quản trị cơ sở dữ liệu lớn như MySQL, Oracle, MS SQL, PostgreSQL...
- Ngôn ngữ script Ruby được viết bên phía Server
- Có các bộ sinh script để tự động thực hiện các task

Tương ứng với các đặc điểm trên Ruby on Rails gồm các thành phần sau:

- Action Mailer
- Action Pack
- Active Model
- Active Record
- Active Resource
- RESTful Architecture
- Active Support
- Railties



Hình 25: Kiến trúc tổng quan của Ruby on Rails

- ***Action Mailer (theo tài liệu [8])***

Mô-đun này chịu trách nhiệm cung cấp các dịch vụ email. Nó xử lý các email đến và cho phép tạo mới các email gửi đi. Mô-đun này có thể xử lý từ những email text đơn giản đến những email có các định dạng phức tạp. Action Mailer được kế thừa từ Action Controller, nó cung cấp các phương thức để tạo email với các template cũng giống như Action View sử dụng nó để hiển thị các trang lên trình duyệt.

- ***Action Pack (theo tài liệu [8])***

Mô-đun Action Pack cung cấp các lớp ở tầng Controller và View trong mô hình MVC. Những mô-đun này tiếp nhận request từ client và sau đó ánh xạ chúng đến các action tương ứng, những action này được định nghĩa trong tầng Controller và sau đó các action này sẽ hiển thị view hiển thị lên trình duyệt. Action Pack được chia thành 3 mô-đun con, đó là:

Action Dispatch: Xử lý điều hướng các request từ phía client, nó phân tích cú pháp các request và xử lý một vài quá trình liên quan đến giao thức HTTP như xử lý cookies, session...

Action Controller: Sau khi Action Dispatch xử lý request nó sẽ điều hướng các request đến các controller tương ứng, Mô-đun này cung cấp các controller cơ bản để các controller khác có thể kế thừa từ nó. Action Controller chứa các action để điều khiển model và view, thêm vào đó nó quản lý các session người dùng, luồng chảy ứng dụng và mô-đun helper. Các action của Action Controller bao gồm: index, new, create, show, edit, update, destroy.

Action View: Được gọi bởi Action Controller. Nó hiển thị các view khi có yêu cầu, Action View cung cấp các layouts, templates và các view helpers, các thành phần này hỗ trợ việc sinh tự động ra phần khung cho các trang HTML hay các định dạng khác. Có 3 template trong Rails là: rhtml, rxml và rjs. Định dạng rhtml sinh ra các view HTML cho người dùng có nhúng thêm các đoạn code Ruby (ERB), rxml được sử dụng để xây dựng các tài liệu XML, rjs cho phép tạo ra các đoạn mã động JavaScript để thực thi các phương thức AJAX.

- ***Active Model (theo tài liệu [8])***

Định nghĩa interface giữa mô-đun Action Pack và Active Record. Giao diện Action Record có thể được dùng bên ngoài Rails, cung cấp các chức năng Object – relational mapping (ORM). ORM là một kỹ thuật lập trình giúp chuyển đổi dữ liệu giữa các hệ thống không tương thích được viết bởi các ngôn ngữ hướng đối tượng

- ***Active Record (theo tài liệu [8])***

Được sử dụng để quản lý dữ liệu trong các cơ sở dữ liệu quan hệ thông qua các đối tượng. Trong Ruby on Rails, mô-đun Active Record cung cấp Object – relational mapping đến các class. Mô-đun này xây dựng nên tầng model, giúp kết nối

các bảng database với các lớp tương ứng trong các lớp Ruby. Rails cung cấp các công cụ để thực thi chức năng CRUD (Create – Read – Update – Destroy) mà không phải cấu hình. CRUD cho phép tạo mới, đọc, cập nhật và xóa các bản ghi trong cơ sở dữ liệu thông qua các đối tượng Ruby. Thêm vào đó, nó cũng cung cấp các khả năng tìm kiếm thông minh và khả năng tạo ra các mối quan hệ hay các liên kết giữa các model. Active Records có những quy ước rất chặt chẽ bắt các developer phải tuân thủ theo như: phải đặt tên lớp, các bảng, các khóa chính và khóa ngoài như thế nào.

Mô-đun này được sử dụng để tạo ra các lớp trong tầng Model, chúng chứa đựng phần logic nghiệp vụ, xử lý validate và các mối quan hệ, ánh xạ đến các bảng và hỗ trợ kết nối đến các hệ quản trị cơ sở dữ liệu khác nhau.

- ***Active Resource (theo tài liệu [8])***

Quản lý các kết nối giữa RESTful và các đối tượng nghiệp vụ, làm giảm số lượng code cần viết để ánh xạ đến các tài nguyên. Active Resource ánh xạ model class đến các tài nguyên REST ở xa cũng giống như cách mà Active Record ánh xạ từ model class đến các bảng cơ sở dữ liệu. Nó cũng cung cấp các cơ chế proxy giữa Active Resource (client) và một RESTful service. Khi một có một request cần truy cập đến các tài nguyên ở xa, một REST XML được sinh ra và được truyền đi, sau đó kết quả sẽ được phân tích sang các đối tượng Ruby.

- ***RESTful Architecture (theo tài liệu [9])***

RESTful web service là các web service được viết dựa trên kiến trúc REST. REST đã được chọn sửa dụng rộng rãi thay thế cho các web service dựa trên SOAP và WSDL. RESTful web service nhẹ, có khả năng dễ mở rộng và bảo trì. Trong bài viết này mình muốn làm rõ các quy tắc, quy ước giúp chúng ta có thể viết được các web service đúng chuẩn REST. Bài viết hướng tới đối tượng là những người chưa biết hoặc chưa rõ về RESTful web service.

REST là viết tắt của REpresentational State Transfer là một kiểu kiến trúc lập trình, nó định nghĩa các quy tắc để thiết kế các web service chú trọng vào tài nguyên hệ thống. Trong kiến trúc REST mọi thứ đều được coi là tài nguyên, chúng có thể là: tệp văn bản, ảnh, trang html, video, hoặc dữ liệu động... REST server cung cấp quyền truy cập vào các tài nguyên, REST client truy cập và thay đổi các tài nguyên đó. Ở đây các tài nguyên được định danh dựa vào URI, REST sử dụng một vài đại diện để biểu diễn các tài nguyên như văn bản, JSON, XML.

Điểm chú ý đầu tiên khi cài đặt RESTful web service là sử dụng một cách rõ ràng các phương thức HTTP theo cách mà chúng được định nghĩa bởi RFC 2616. Ví dụ phương thức GET được định nghĩa với mục đích để client sử dụng vào việc lấy dữ liệu từ server hoặc thực hiện một truy vấn để server tìm và trả về một tập các tài nguyên phù hợp.

REST yêu cầu các nhà phát triển sử dụng một cách rõ ràng các phương thức HTTP và nhất quán với cách mà chúng được định nghĩa. Quy tắc này của REST giúp thiết lập ánh xạ một một giữa các hành động tạo, đọc, cập nhật và xóa với các phương thức HTTP. Dưới đây trình bày bốn quy tắc cơ bản để cài đặt một RESTful web service:

- Để tạo một tài nguyên trên server ta dùng phương thức POST, tương ứng với action: create
- Để lấy (đọc) tài nguyên trên server ta dùng phương thức GET, tương ứng với action: index, new, edit, show
- Để update tài nguyên trên server ta dùng phương thức PUT, tương ứng với action: update
- Để xóa tài nguyên trên server ta dùng phương thức DELETE, tương ứng với action: destroy.

- ***Active Support (theo tài liệu [8])***

Là một bộ tiện ích và các thư viện chuẩn của Ruby, nó hữu ích cho quá trình phát triển ứng dụng vì nó bao gồm rất nhiều sự hỗ trợ cho các multi-bytes string, internationalization, time zones và testing.

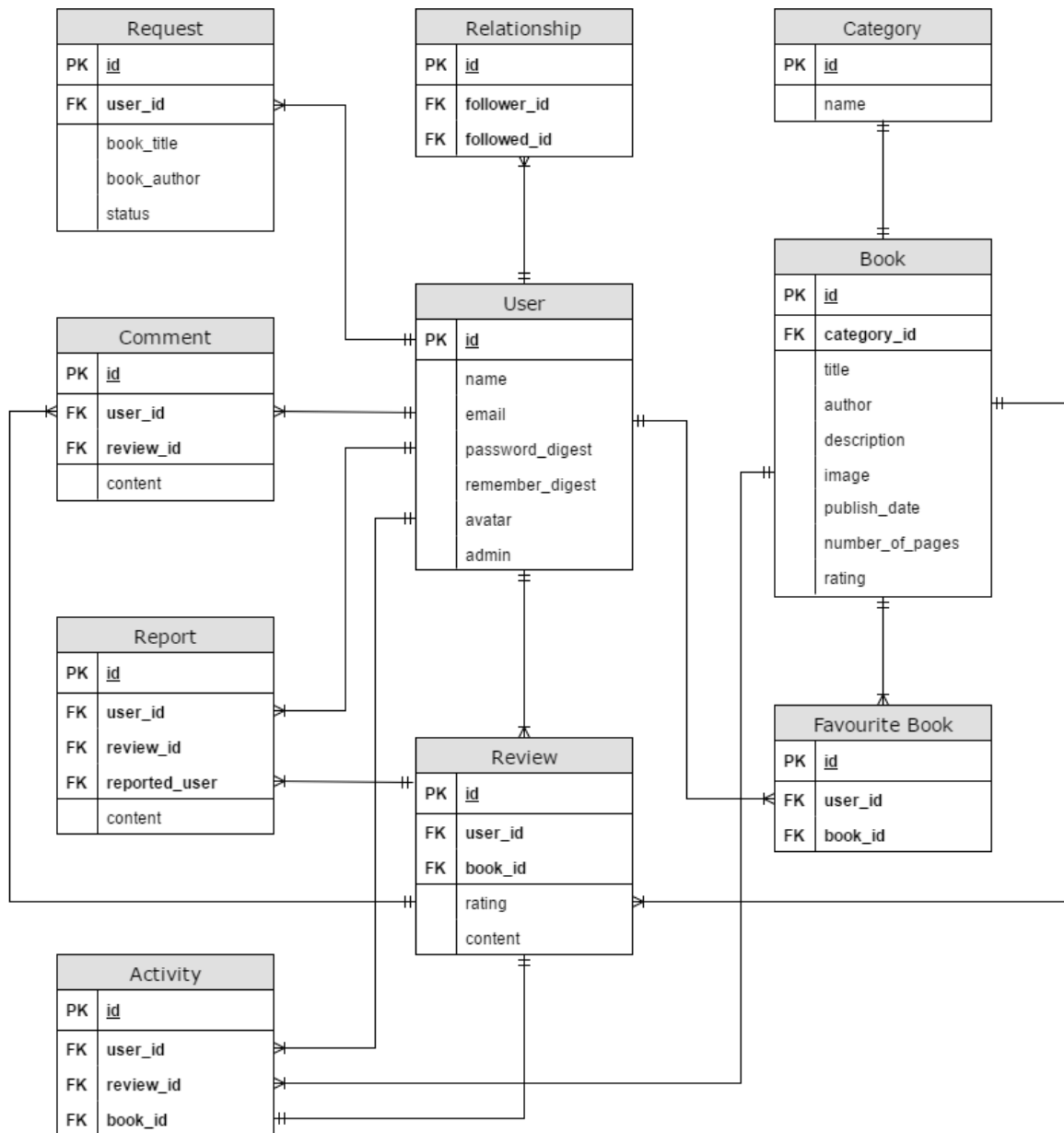
Active Support là một thành phần của Ruby on Rails được cung cấp nhằm mục đích mở rộng ngôn ngữ Ruby. Có nghĩa Active Support là tập hợp các hàm tiện ích được Rails nhúng thẳng vào lớp của thư viện chuẩn của Ruby (như Array, String). Nhờ tính năng reopen của Ruby, việc nhúng này rất dễ dàng, mang lại cảm giác các hàm này có sẵn trong bản thân ngôn ngữ. Các tiện ích trong ActiveSupport quá hay, đến nỗi nhiều người cứ tưởng các chúng là tính năng của Ruby, góp phần tạo nên triệu chứng biết Rails mà không biết Ruby!

- ***Railties (theo tài liệu [8])***

Đảm nhiệm kết nối tất các các mô-đun ở trên với nhau, thêm vào đó nó xử lý quá trình môi cho ứng dụng, giao diện dòng lệnh, và cung cấp bộ sinh code của Rails. Rake là một trong những lệnh được sử dụng để thực hiện các task database, triển khai, tài liệu, testing.

3.2. Thiết kế cơ sở dữ liệu

3.2.1. Cơ sở dữ liệu của hệ thống



Hình 26: Cơ sở dữ liệu của hệ thống

Giải thích các bảng trong cơ sở dữ liệu

- User: Danh sách tài khoản người dùng của hệ thống
- Book: Danh sách các cuốn sách trong hệ thống
- Category: Danh sách thể loại sách
- Favourite Book: Danh sách những cuốn sách yêu thích của người dùng
- Review: Danh sách các bài đánh giá trong hệ thống
- Comment: Danh sách các bình luận của những bài đánh giá
- Request: Danh sách các yêu cầu sách của người dùng

- Report: Danh sách các báo cáo vi phạm
- Activity: Danh sách hoạt động viết bài đánh giá của người dùng
- Relationship: Danh sách quan hệ của người dùng với nhau

3.2.2. Bảng User

Bảng 1: Bảng thông tin chi tiết User

Tên trường	Kiểu	Ý nghĩa
id	Integer	Khóa chính của bảng, tăng tự động
name	String	Tên của người dùng
email	String	Email của người dùng
password_digest	String	Password của người dùng sau khi mã hóa
remember_digest	String	Mã lưu tài khoản đăng nhập
avatar	String	Ảnh đại diện của người dùng
admin	Boolean, default: false	Xác định quyền admin của người dùng, có giá trị mặc định là false

3.2.3. Bảng Book

Bảng 2: Bảng thông tin chi tiết Book

Tên trường	Kiểu	Ý nghĩa
id	Integer	Khóa chính của bảng, tăng tự động
category_id	Integer	Thẻ loại của sách Khóa ngoại tham chiếu tới “id” của bảng Category
title	String	Tên sách
author	String	Tên tác giả
description	String	Giới thiệu sách
image	String	Ảnh bìa cuốn sách
publish_date	Datetime	Ngày phát hành
number_of_pages	Integer	Số trang sách
rating	Float, default: 0.00	Điểm số trung bình của sách

3.2.4. Bảng Category

Bảng 3: Bảng thông tin chi tiết Category

Tên trường	Kiểu	Ý nghĩa
id	Integer	Khóa chính của bảng, tăng tự động
name	String	Tên của thể loại sách

3.2.5. Bảng Favourite Book

Bảng 4: Bảng thông tin chi tiết Favourite Book

Tên trường	Kiểu	Ý nghĩa
id	Integer	Khóa chính của bảng, tăng tự động
user_id	Integer	Id của người dùng Khóa ngoại tham chiếu tới “id” của bảng User
book_id	Integer	Id của sách được thích Khóa ngoại tham chiếu tới “id” của bảng Book

3.2.6. Bảng Review

Bảng 5: Bảng thông tin chi tiết Review

Tên trường	Kiểu	Ý nghĩa
id	Integer	Khóa chính của bảng, tăng tự động
user_id	Integer	Id của người dùng viết bài đánh giá Khóa ngoại tham chiếu tới “id” của bảng User
book_id	Integer	Id của sách được viết bài đánh giá Khóa ngoại tham chiếu tới “id” của bảng Book
rating	Integer	Điểm của bài đánh giá
content	String	Nội dung của bài đánh giá

3.2.7. Bảng Comment

Bảng 6: Bảng thông tin chi tiết Comment

Tên trường	Kiểu	Ý nghĩa
id	Integer	Khóa chính của bảng, tăng tự động
user_id	Integer	Id của người dùng viết bình luận Khóa ngoại tham chiếu tới “id” của bảng User
review_id	Integer	Id của bài đánh giá được bình luận Khóa ngoại tham chiếu tới “id” của bảng Review
content	String	Nội dung bình luận

3.2.8. Bảng Request

Bảng 7: Bảng thông tin chi tiết Request

Tên trường	Kiểu	Ý nghĩa
id	Integer	Khóa chính của bảng, tăng tự động
user_id	Integer	Id của người dùng gửi yêu cầu sách Khóa ngoại tham chiếu tới “id” của bảng User
book_title	String	Tên sách yêu cầu
book_author	String	Tên tác giả
status	Integer, default: 0	Trạng thái của yêu cầu 0: Đang chờ (mặc định) 1: Chấp nhận 2: Từ chối

3.2.9. Bảng Report

Bảng 8: Bảng thông tin chi tiết Report

Tên trường	Kiểu	Ý nghĩa
id	Integer	Khóa chính của bảng, tăng tự động
user_id	Integer	Id của người dùng gửi báo cáo Khóa ngoại tham chiếu tới “id” của bảng User
review_id	Integer	Id của bài đánh giá bị báo cáo Khóa ngoại tham chiếu tới “id” của bảng Book

reported_user	Integer	Id của người viết bài đánh giá bị báo cáo Khóa ngoại tham chiếu tới “id” của bảng User
content	String	Nội dung của bài đánh giá

3.2.10. Bảng Activity

Bảng 9: Bảng thông tin chi tiết Activity

Tên trường	Kiểu	Ý nghĩa
id	Integer	Khóa chính của bảng, tăng tự động
user_id	Integer	Id của người dùng viết bài đánh giá Khóa ngoại tham chiếu tới “id” của bảng User
review_id	Integer	Id của bài đánh giá Khóa ngoại tham chiếu tới “id” của bảng Book
book_id	Integer	Id của sách được viết đánh giá Khóa ngoại tham chiếu tới “id” của bảng User

3.2.11. Bảng Relationship

Bảng 10: Bảng thông tin chi tiết Relationship

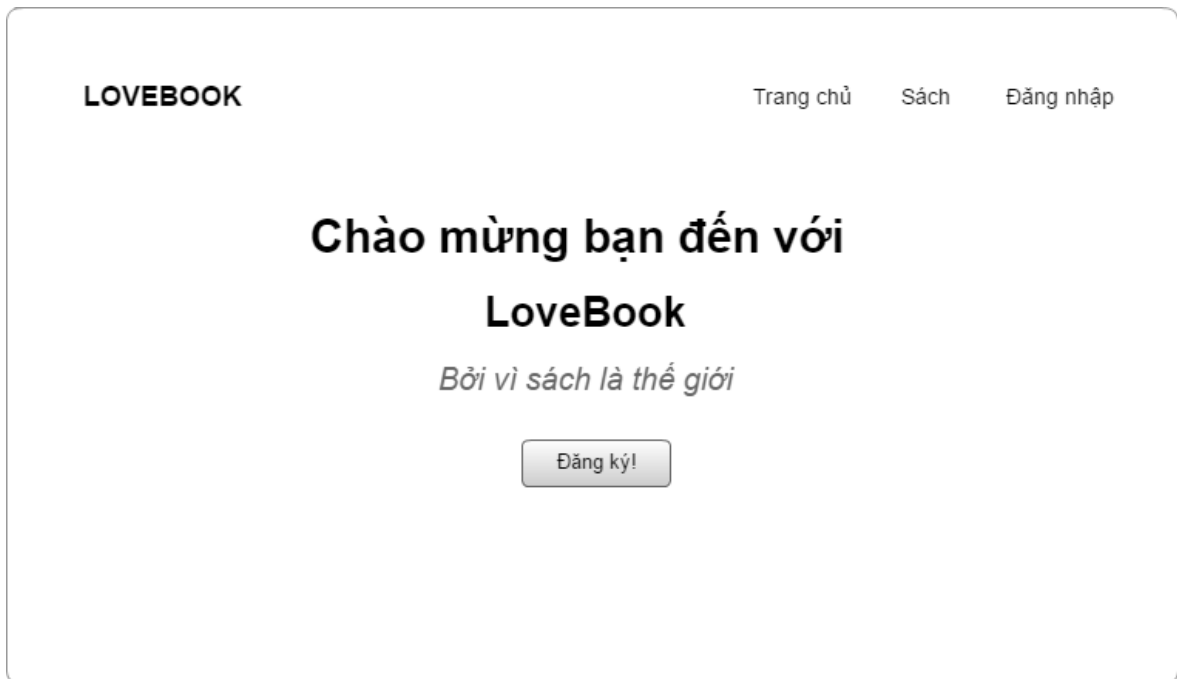
Tên trường	Kiểu	Ý nghĩa
id	Integer	Khóa chính của bảng, tăng tự động
follower_id	Integer	Id của người theo dõi Khóa ngoại tham chiếu tới “id” của bảng User
followed_id	Integer	Id của người được theo dõi Khóa ngoại tham chiếu tới “id” của bảng Book

3.3. Thiết kế giao diện

3.3.1. Giao diện trang chủ

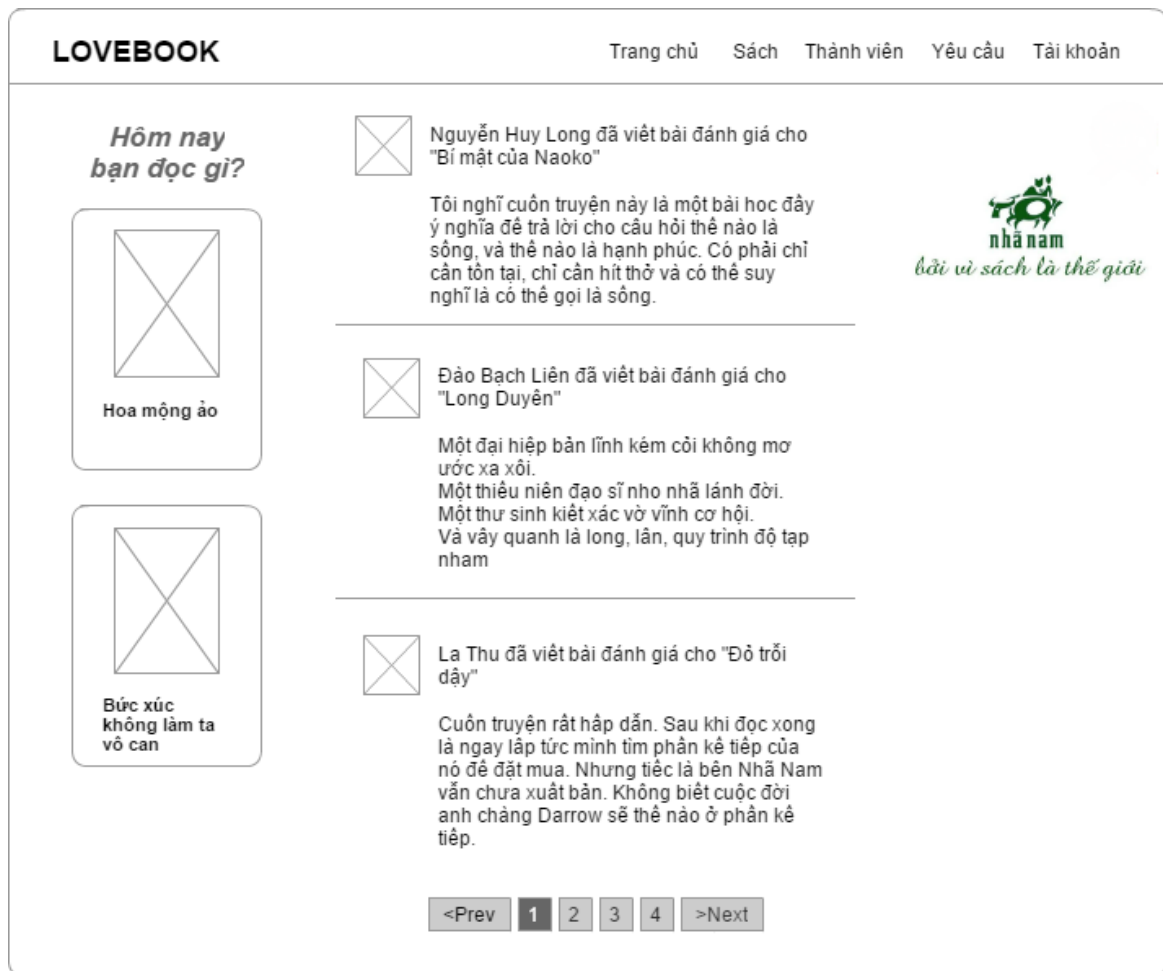
- Giao diện trang chủ ban đầu khi vào website được thiết kế như hình bên dưới, bao gồm 2 lựa chọn trên thanh công cụ: Sách, Đăng nhập. Ngoài ra thanh công cụ còn có tên của website: LoveBook

Chính giữa trang chủ sẽ là slogan của website cùng nút “Đăng ký!”



Hình 27: Mockup giao diện Trang chủ trước đăng nhập

- Sau khi đăng nhập, giao diện trang chủ sẽ được chia là 3 phần:
 - Ở chính giữa sẽ hiển thị thông tin về các hoạt động đánh giá sách của chính người dùng và những thành viên được người dùng theo dõi. Các thông tin này sẽ được hệ thống phân trang cụ thể.
 - Bên trái sẽ hiển thị ngẫu nhiên 5 cuốn sách bất kì mà hệ thống lựa chọn để giới thiệu cho người dùng
 - Bên phải, trong tương lai, khi website phát triển sẽ được cho thuê để trở thành mục quảng cáo
- Trên thanh công cụ lúc này sẽ có thêm 2 lựa chọn là: Yêu cầu và Thành viên



Hình 28: Mockup giao diện Trang chủ sau khi đã đăng nhập

3.3.2. Giao diện đăng nhập và đăng ký

- Người dùng đã có đăng ký tài khoản, có quyền đăng nhập vào hệ thống. Giao diện trang đăng nhập được thiết kế như sau:

Đăng nhập

Email

Mật khẩu

☒ Nhớ mật khẩu

Người dùng mới? [Đăng ký!](#)

Hình 29: Mockup trang đăng nhập

Người dùng muốn đăng nhập vào hệ thống được yêu cầu cung cấp email và mật khẩu để xác thực. Ngoài ra nếu muốn duy trì trạng thái đăng nhập, người dùng có thể chọn vào ô “Nhớ mật khẩu”.

Nếu người dùng là khách vãng lai chưa có tài khoản đăng nhập, trên giao diện đăng nhập sẽ có một đường link “Đăng ký!” dẫn trực tiếp người dùng đến trang đăng ký.

- Giao diện trang đăng ký yêu cầu người dùng điền các thông tin: Tên, email, mật khẩu và nhập lại mật khẩu để xác nhận.

Sau khi người dùng bấm nút “Tạo mới tài khoản”, hệ thống hiển thị thông báo đăng kí tài khoản thành công và chuyển sang trang đăng nhập để người dùng đăng nhập vào tài khoản mới tạo.



The image shows a registration form titled "Đăng ký" (Register). It contains four input fields: "Tên" (Name), "Email", "Mật khẩu" (Password), and "Nhập lại mật khẩu" (Repeat password). Below the fields is a button labeled "Tạo mới tài khoản" (Create new account).

Hình 30: Mockup trang đăng ký

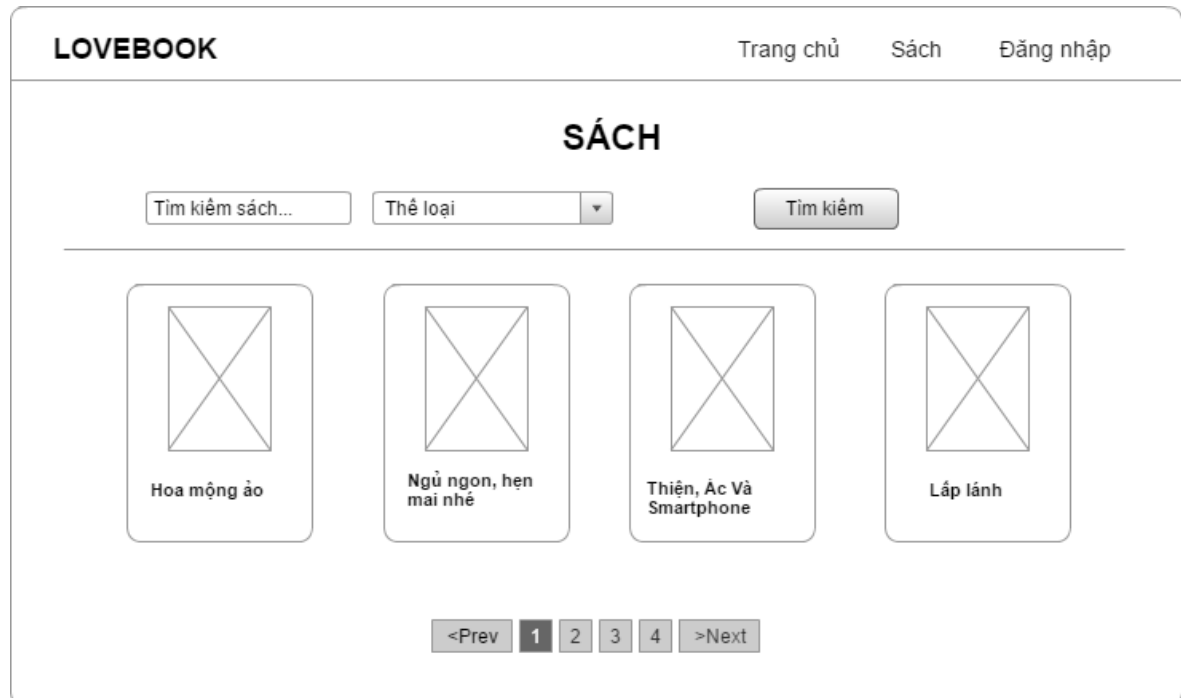
3.3.3. Giao diện trang xem sách và tìm kiếm sách

- Khi chọn vào mục “Sách” trên thanh công cụ của trang web, người dùng sẽ được chuyển đến trang xem toàn bộ sách và tìm kiếm sách.

Hệ thống tự động hiển thị toàn bộ sách có trong cơ sở dữ liệu và có phân trang để người dùng tiện theo dõi. Mỗi cuốn sách sẽ hiển thị các thông tin cơ bản là hình ảnh bìa sách, tên sách, tên tác giả.

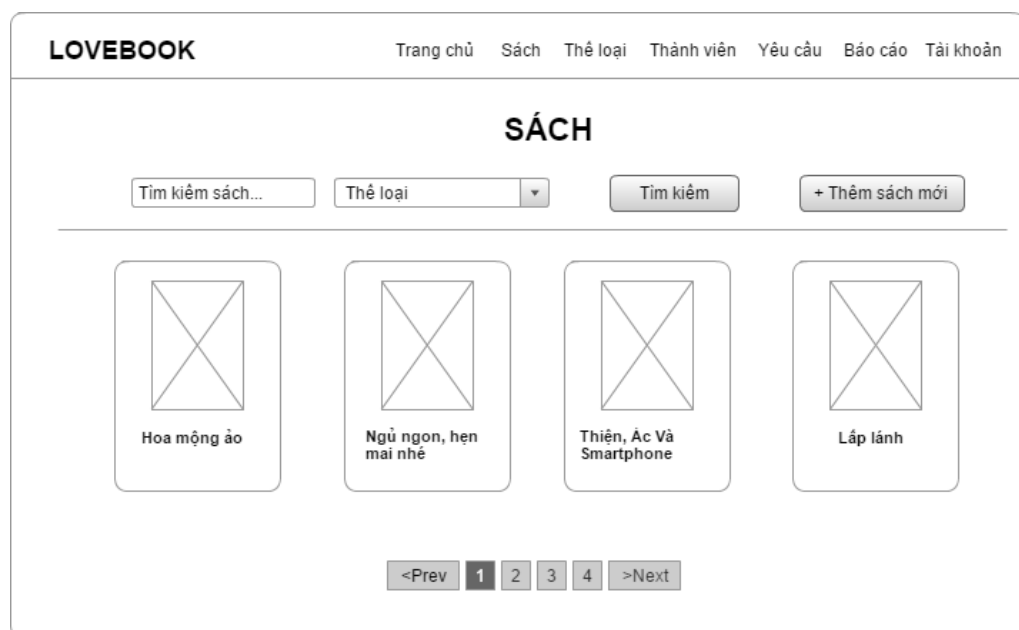
Trên trang này, người dùng có thể tìm kiếm cuốn sách mình quan tâm bằng cách nhập thông tin vào ô tìm kiếm và chọn thể loại sách. Người dùng có thể lựa chọn

chỉ nhập thông tin tìm kiếm, hoặc chỉ chọn thể loại sách hoặc vừa nhập thông tin vừa chọn thể loại sách. Hệ thống sẽ tìm kiếm sách theo tên sách hoặc tên tác giả dựa và thông tin người dùng nhập vào ô tìm kiếm và thể loại sách đã chọn. Sau đó hiển thị kết quả tìm kiếm cho người dùng.



Hình 31: Mockup trang hiển thị và tìm kiếm sách của khách và thành viên

- Trên giao diện xem sách và tìm kiếm sách của quản trị viên sẽ có thêm nút “+ Thêm sách mới” dẫn quản trị viên đến trang tạo mới sách.



Hình 32: Mockup trang hiển thị và tìm kiếm sách của quản trị viên

- Sau khi xem toàn bộ sách, người dùng có thể lựa chọn một cuốn sách để xem cụ thể. Trên giao diện của một cuốn sách cụ thể sẽ được hiển thị các thông tin cơ bản của cuốn sách đó như tên sách, tên tác giả, được số trung bình, số lượng người yêu thích, số trang, ngày phát hành, thể loại và giới thiệu sách. Ngoài ra giao diện còn hiển thị những bài đánh giá của cuốn sách đó.

Trên trang hiển thị sách của quản trị viên sẽ có thêm 2 nút “Chỉnh sửa” và “Xóa” cho phép quản trị viên chỉnh sửa hoặc xóa cuốn sách đang xem.


Mỗi một thành viên sẽ chỉ được quyền viết một bài đánh giá cho một cuốn sách. Với những cuốn sách mà thành viên chưa viết đánh giá, trên giao diện sẽ có đường link “Viết bài đánh giá của bạn”, dẫn thành viên có nhu cầu đến trang viết bài đánh giá. Đường link này sẽ không hiển thị với khách và thành viên đã có bài đánh giá về cuốn sách đang xem.

Những bài đánh giá của cuốn sách sẽ được xếp theo thứ tự thời gian từ gần đến xa nhất. Thành viên có thể lựa chọn chỉnh sửa hoặc xóa bài đánh giá của mình trên chính giao diện này.

Ngoài ra, người dùng còn có thể lựa chọn “Xem thêm” để được dẫn đến trang xem cụ thể một bài đánh giá.

LOVEBOOK

Trang chủSáchThành viênYêu cầuTài khoản



Nắng trong vườn

Thạch Lam

♡ 23

☆☆☆☆☆ 4.5 (13 bài đánh giá)

Thạch Lam có sở trường viết truyện ngắn. Văn phong của ông giản dị, trong sáng, nhẹ nhàng nhưng sâu sắc, thâm trầm. Thạch Lam đôi khi còn đặt nhân vật của mình vào vùng ranh giới tranh chấp giữa cái thiện và cái ác, để rồi tự bản thân con người bằng việc thức tỉnh của lương tri, của phẩm giá, xác lập chỗ đứng tốt đẹp cho mình trong cuộc sống đầy bộn nhoe của xã hội cũ.

Đánh giá của cộng đồng ([Viết bài đánh giá của bạn](#))

Heep đánh giá 4/5

[Sửa](#) [Xóa](#)

Thạch Lam xứng đáng là bậc thầy về truyện ngắn lãng mạn. Những truyện ngắn trong tập Nắng trong vườn đều là những viên ngọc- trong trẻo, ngọt ngào, mơ màng. Nhã Nam chăm chút, đầu tư về mặt hình thức cho bộ sách này, mình rất hài lòng về hình thức của tập Nắng trong vườn nói riêng, và bộ Việt Nam danh tác nói chung.

[Xem thêm](#)

<Prev1234>Next

Hình 33: Mockup trang xem chi tiết sách

3.3.4. Giao diện bài đánh giá

- Trên giao diện hiển thị bài đánh giá sẽ có các thông tin: Bài đánh giá của ai, viết cho cuốn sách nào, đánh giá bao nhiêu điểm và nội dung bài đánh giá.

Nếu là bài đánh giá của chính tài khoản đang đăng nhập hệ thống thì sẽ có 2 nút “Sửa” và “Xóa”, cho phép người dùng chuyển sang trang chỉnh sửa hoặc xóa bài đánh giá của mình.

Nếu là bài đánh giá của thành viên khác thì sẽ hiển thị lựa chọn “Báo cáo” để người dùng báo cáo nếu thấy bài đánh giá có nội dung không phù hợp.


Ngoài ra trên giao diện này còn cho phép người dùng trực tiếp bình luận bài đánh giá. Sau khi bình luận xong có thể chỉnh sửa bình luận hoặc xóa bình luận. Mỗi bình luận sẽ hiển thị ảnh đại diện và tên người bình luận cùng nội dung bình luận.

LOVEBOOK

Trang chủSáchThành viênYêu cầuTài khoản

Bài đánh giá của Heep > Năng trong vườn

SửaXóa



☆☆☆☆☆ 4/5

Thạch Lam xứng đáng là bậc thầy về truyện ngắn lãng mạn. Những truyện ngắn trong tập Năng trong vườn đều là những viên ngọc- trong trẻo, ngọt ngào, mơ màng. Nhà Nam chăm chút, đầu tư về mặt hình thức cho bộ sách này, mình rất hài lòng về hình thức của tập Năng trong vườn nói riêng, và bộ Việt Nam danh tác nói chung.

Bình luận

Bình luận

Sửa Xóa

Sửa Xóa

Sửa Xóa

Hình 34: Mockup trang xem chi tiết một bài đánh giá sách của thành viên

- Trên giao diện xem chi tiết một bài đánh giá của quản trị viên sẽ chỉ có nút “Xóa”, cho phép quản trị viên xóa những bài đánh giá có nội dung không phù hợp.

LOVEBOOK

Trang chủ
Sách
Thành viên
Yêu cầu
Tài khoản

Bài đánh giá của Heep > Nắng trong vườn

Xóa

☆☆☆☆☆ 4/5

Thạch Lam xứng đáng là bậc thầy về truyện ngắn lãng mạn. Những truyện ngắn trong tập Nắng trong vườn đều là những viên ngọc- trong trẻo, ngọt ngào, mơ màng. Nhà Nam chăm chút, đầu tư về mặt hình thức cho bộ sách này, mình rất hài lòng về hình thức của tập Nắng trong vườn nói riêng, và bộ Việt Nam danh tác nói chung.

Bình luận

Bình luận

Sửa Xóa

Sửa Xóa

Sửa Xóa

Hình 35: Mockup trang xem chi tiết một bài đánh giá sách của quản trị viên

3.3.5. Giao diện viết bài đánh giá

Giao diện viết bài đánh giá sẽ hiển thị tên cuốn sách đang được viết đánh giá và yêu cầu người dùng chấm điểm cũng như nêu những suy nghĩ, nhận xét của cá nhân về cuốn sách.

Người dùng có thể chỉ chấm điểm chứ không nhất thiết cần viết đánh giá cụ thể cho cuốn sách. Điểm sẽ được đánh theo thang điểm từ 1 đến 5, không có điểm lẻ.

Thiện, Ác Và Smartphone > Bài đánh giá

Điểm

Điểm

Bạn nhận xét gì về cuốn sách

Lưu

Hình 36: Mockup trang viết bài đánh giá

3.3.6. Giao diện quản lý yêu cầu của thành viên

Khi người dùng lựa chọn “Yêu cầu” trên thanh công cụ, hệ thống sẽ chuyển đến trang quản lý yêu cầu của người dùng.

Yêu cầu

+ Gửi yêu cầu mới

Tên sách	Tác giả	Trạng thái	
		Đang chờ	Sửa Xóa
		Chấp nhận	
		Từ chối	

Hình 37: Mockup trang quản lý yêu cầu của thành viên

Tại đây sẽ hiển thị toàn bộ yêu cầu sách của người dùng cùng với trạng thái của yêu cầu. Mỗi yêu cầu sẽ có 1 trong 3 trạng thái: Đang chờ, Chấp nhận, Từ chối.

Mỗi yêu cầu được gửi đi ban đầu mặc định sẽ ở trạng thái “đang chờ”, chỉ có ở trạng thái này, người dùng mới có quyền chỉnh sửa hoặc xóa yêu cầu của mình.

Sau khi được quản trị viên xem xét, quản trị viên sẽ lựa chọn chấp nhận hoặc từ chối yêu cầu, lúc này trạng thái của yêu cầu cũng được thay đổi và người dùng không còn quyền sửa hoặc xóa yêu cầu của mình.

Khi muốn gửi mới một yêu cầu, người dùng sẽ bấm vào nút “+ Gửi yêu cầu mới” và được chuyển đến giao diện viết yêu cầu. Ở đây, người dùng sẽ được yêu cầu nhập tên và tác giả cuốn sách mà mình muốn có trong hệ thống.

Gửi yêu cầu sách

Tên sách

Tác giả

Gửi yêu cầu

Hình 38: Mockup biểu mẫu gửi yêu cầu

3.3.7. Giao diện trang cá nhân của thành viên

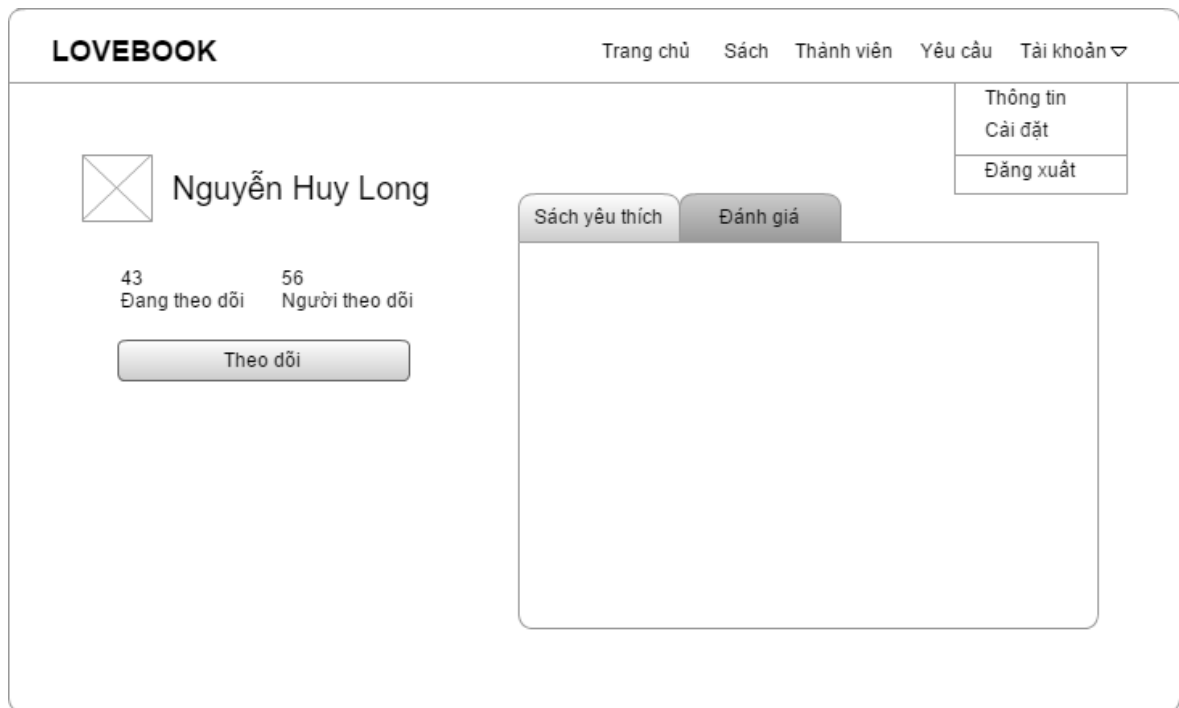
Trên trang cá nhân của mỗi thành viên sẽ được chia làm 2 phần.

Bên trái hiển thị ảnh đại diện và tên thành viên. Ngoài ra còn hiển thị số lượng người đang theo dõi và được theo dõi bởi thành viên đó.

Bên phải sẽ có 2 tab: “Sách yêu thích” hiển thị những cuốn sách được người dùng yêu thích và “Đánh giá” hiển thị những bài đánh giá người dùng đã viết

Sau khi đăng nhập thành công vào hệ thống, thay thế cho phần “Đăng nhập” trên thanh công cụ sẽ là “Tài khoản” bao gồm 3 lựa chọn:

- “Thông tin” dẫn đến trang trang cá nhân của tài khoản đang đăng nhập
- “Cài đặt” dẫn đến trang chỉnh sửa thông tin cá nhân
- “Đăng xuất”

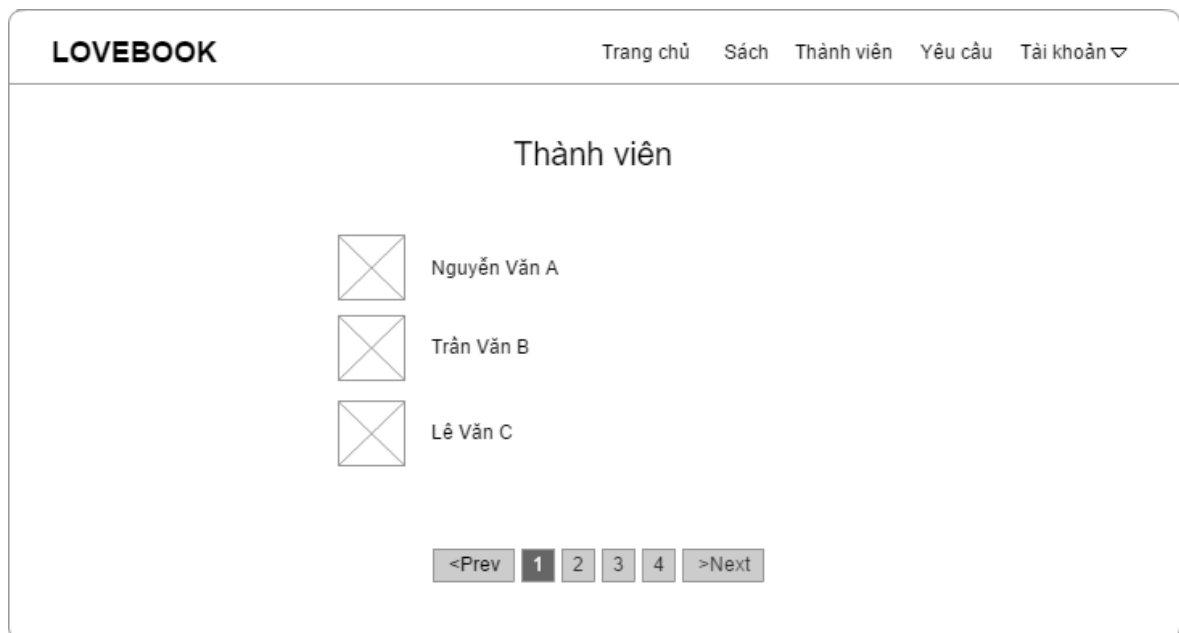


Hình 39: Mockup trang cá nhân của thành viên

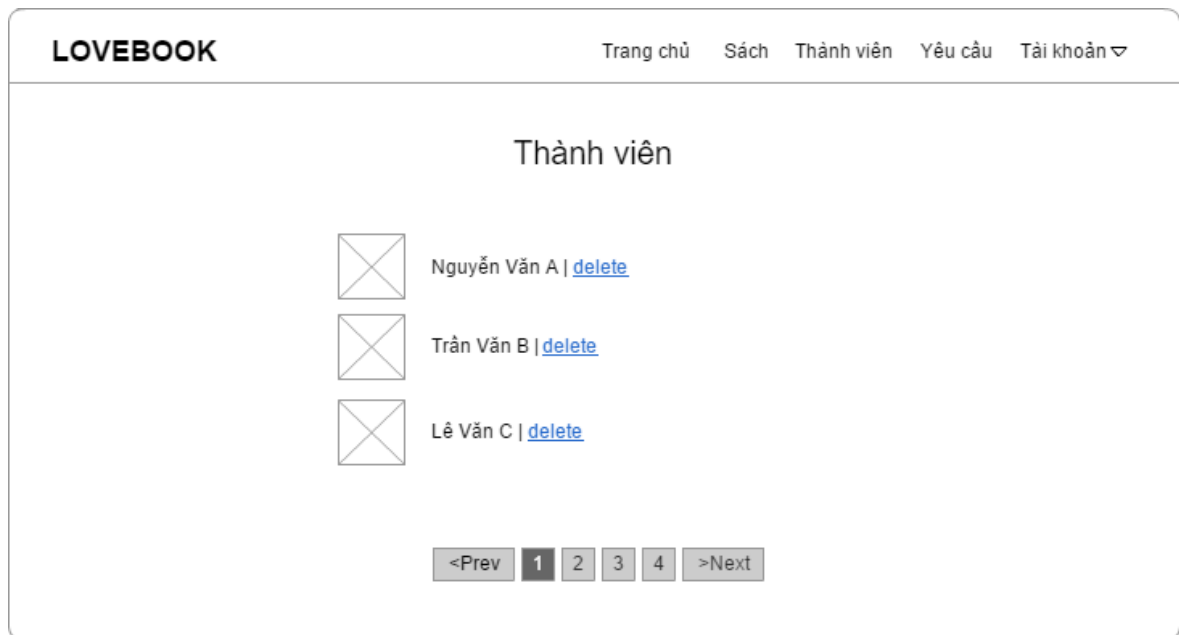
3.3.8. Giao diện danh sách thành viên

Giao diện danh sách thành viên sẽ hiển thị ảnh đại diện cùng với tên các thành viên. Danh sách này sẽ được phân trang, mỗi trang gồm 15 thành viên.

Nếu tài khoản đang đăng nhập là của quản trị viên, bên cạnh tên mỗi thành viên sẽ có nút xóa, cho phép quản trị viên xóa người dùng bất kì.



Hình 40: Mockup trang hiển thị danh sách thành viên của thành viên



Hình 41: Mockup hiển thị danh sách thành viên của quản trị viên

3.3.9. Giao diện quản lý thể loại sách của quản trị viên

Trang quản lý thể loại sách dành riêng cho quản trị viên. Trên trang này sẽ chia làm 2 phần:

- Bên trái là nơi tạo thêm thể loại sách, gồm textbox để nhập tên thể loại cùng nút bấm “+ Thêm thể loại mới”.
- Bên phải là danh sách các thể loại sách có phân trang. Hơn nữa, quản trị viên có thể lựa chọn nút “Sửa” hoặc “Xóa” để chỉnh sửa hoặc xóa trực tiếp các thể loại sách ngay trên chính giao diện này.

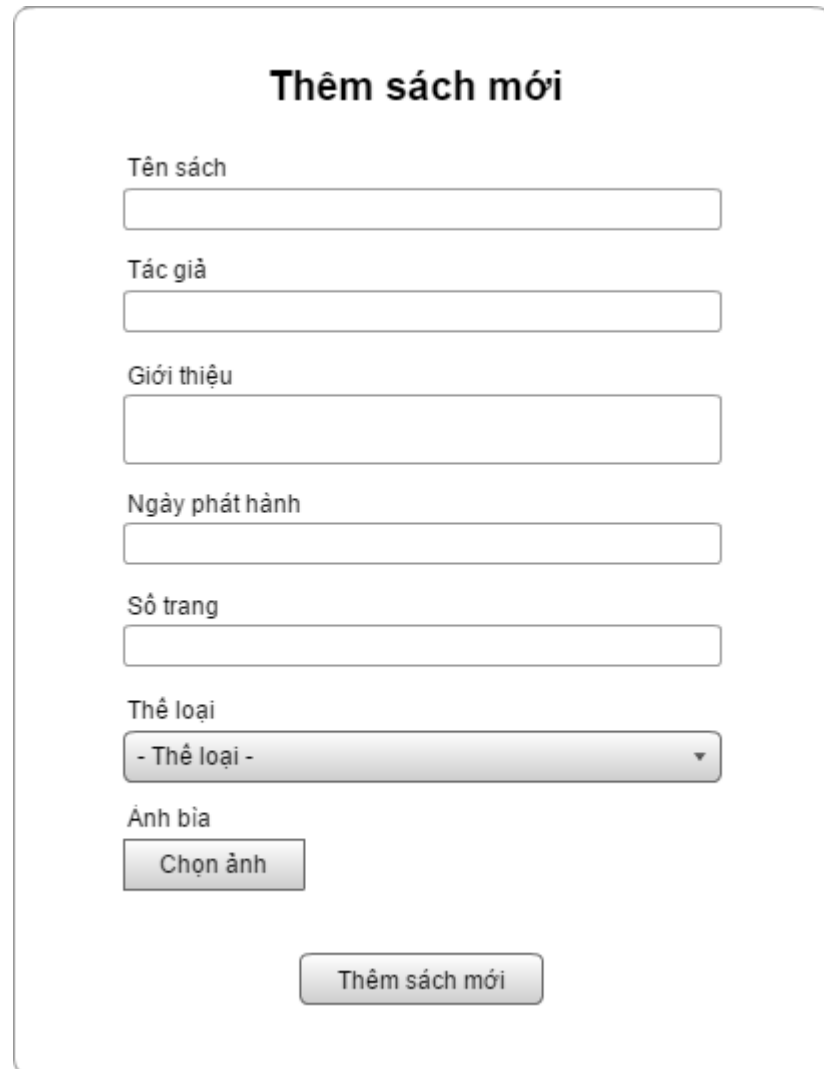


Hình 42: Mockup trang quản lý thể loại của quản trị viên

3.3.10. Giao diện thêm mới sách

Giao diện này dành riêng cho quản trị viên khi bấm nút “+ Thêm sách mới” trên trang xem sách và tìm kiếm sách dành cho quản trị viên.

Trang thêm sách mới yêu cầu quản trị viên điền các thông tin: Tên sách, tác giả, giới thiệu sách, ngày phát hành, số trang, thể loại và ảnh bìa cuốn sách.



Thêm sách mới

Tên sách

Tác giả

Giới thiệu

Ngày phát hành

Số trang

Thể loại
- Thể loại - ▼

Ảnh bìa

Hình 43: Mockup biểu mẫu thêm sách mới dành cho quản trị viên

3.3.11. Giao diện quản lý yêu cầu của quản trị viên

Trang quản lý yêu cầu sẽ hiển thị chi tiết những yêu cầu của thành viên gửi đến, gồm tên sách, tác giả và người yêu cầu.

Quản trị viên sẽ lựa chọn “Chấp nhận” hoặc “Từ chối” yêu cầu.

Yêu cầu			
Tên sách	Tác giả	Người yêu cầu	
Bộ Bộ Kinh Tâm	Đồng Hoa	Phạm Văn A	Chấp nhận Từ chối
Nhật ký trong tù	Hồ Chí Minh	Nguyễn Thị B	Chấp nhận
Truyện Kiều	Nguyễn Du	Trần Quốc C	Từ chối

Hình 44: Mockup trang quản lý yêu cầu của quản trị viên

3.3.12. Giao diện quản lý báo cáo của quản trị viên

Trang quản lý báo cáo sẽ hiển thị thông tin báo cáo vi phạm mà thành viên gửi đến bao gồm: người báo cáo, bài viết bị báo cáo, người viết bài viết đó và thời gian báo cáo.

Tại đây, quản trị viên có thể chọn link để dẫn đến trang hiển thị bài viết bị báo cáo để đánh giá nội dung.

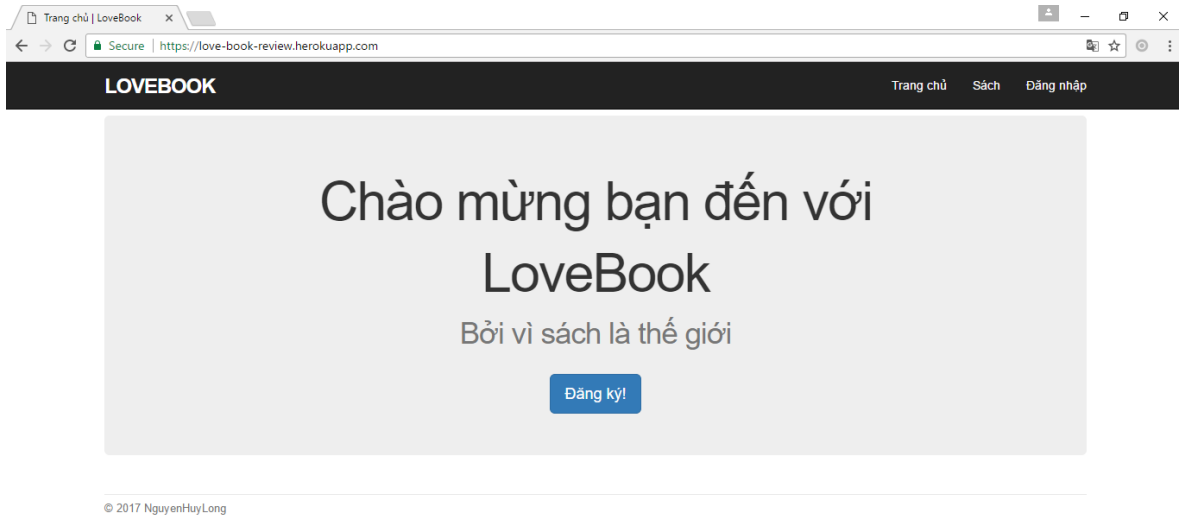
Báo cáo			
Người báo cáo	Bài viết bị báo cáo	Người bị báo cáo	Thời gian báo cáo
Alex	Bài đánh giá về "Trăm năm cô đơn"	Daniel	9:00 09/06/2017
Blank	Bài đánh giá về "Sổ đỏ"	Elisa	10:00 18/5/2017
Christan	Bài đánh giá về "Tiger & Wolf"	Frank	05:06 04/03/2017

Hình 45: Mockup trang quản lý báo cáo của quản trị viên

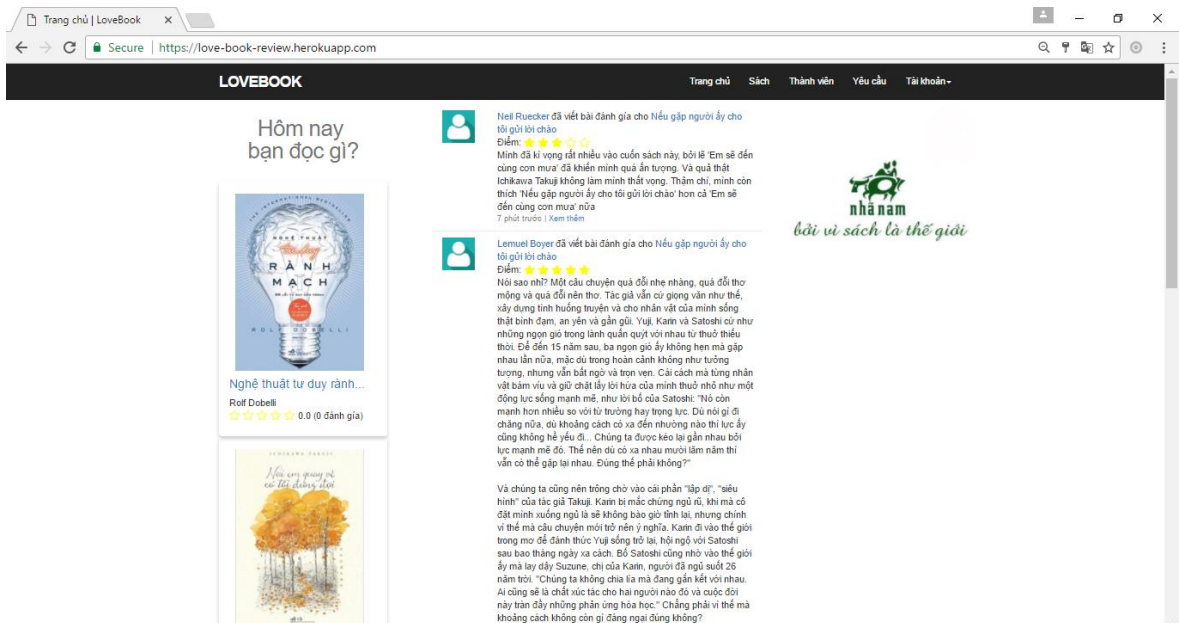
PHẦN 4 – CÀI ĐẶT HỆ THỐNG

4.1. Kết quả cài đặt

4.1.1. Giao diện trang chủ

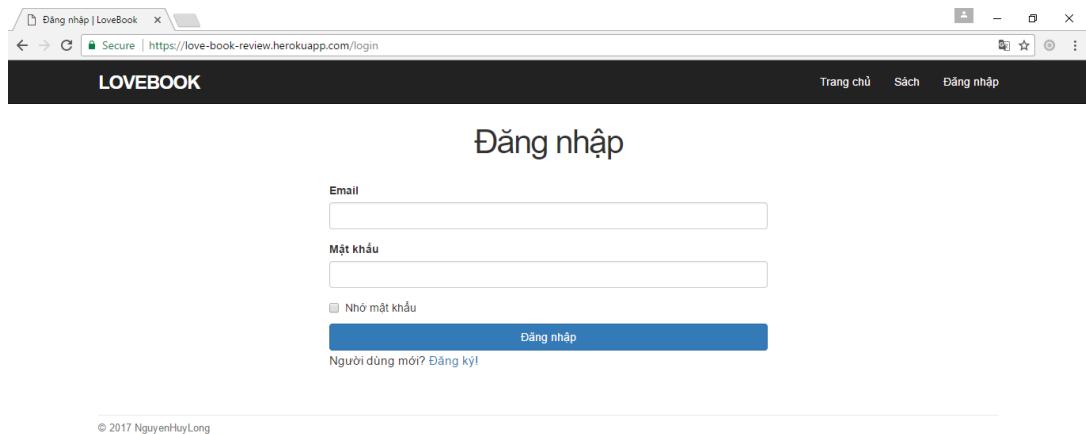


Hình 46: Giao diện trang chủ trước đăng nhập

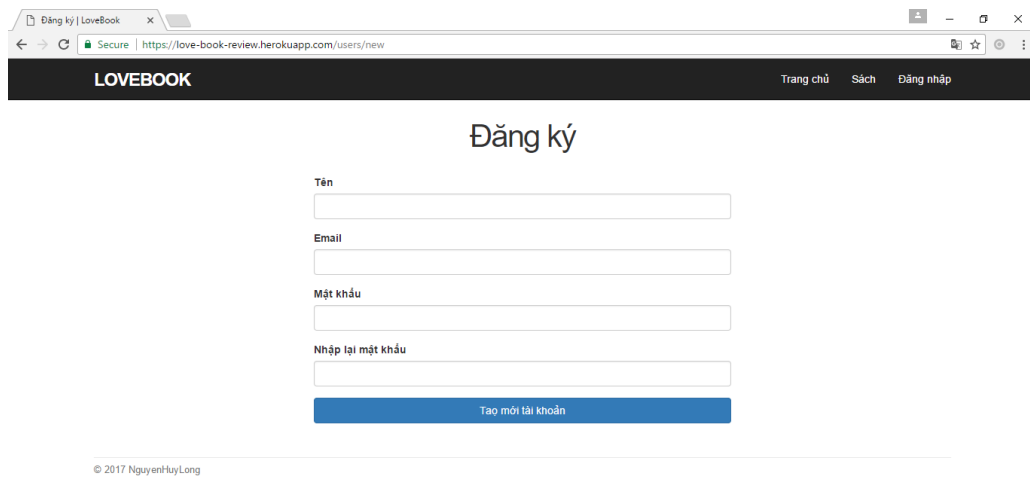


Hình 47: Giao diện trang chủ sau khi đăng nhập

4.1.2. Giao diện trang đăng kí, đăng nhập

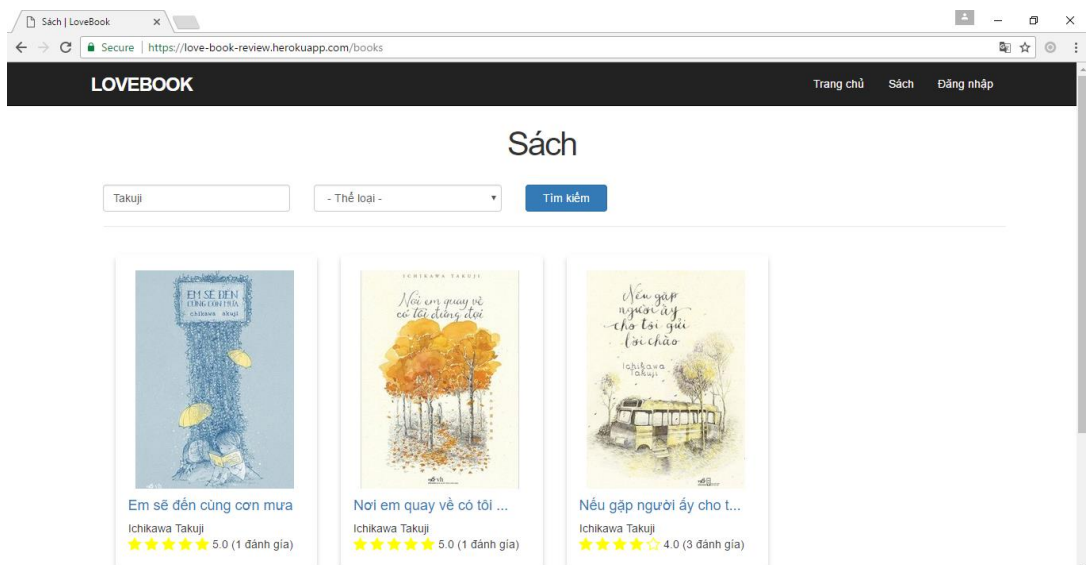


Hình 48: Giao diện trang đăng nhập

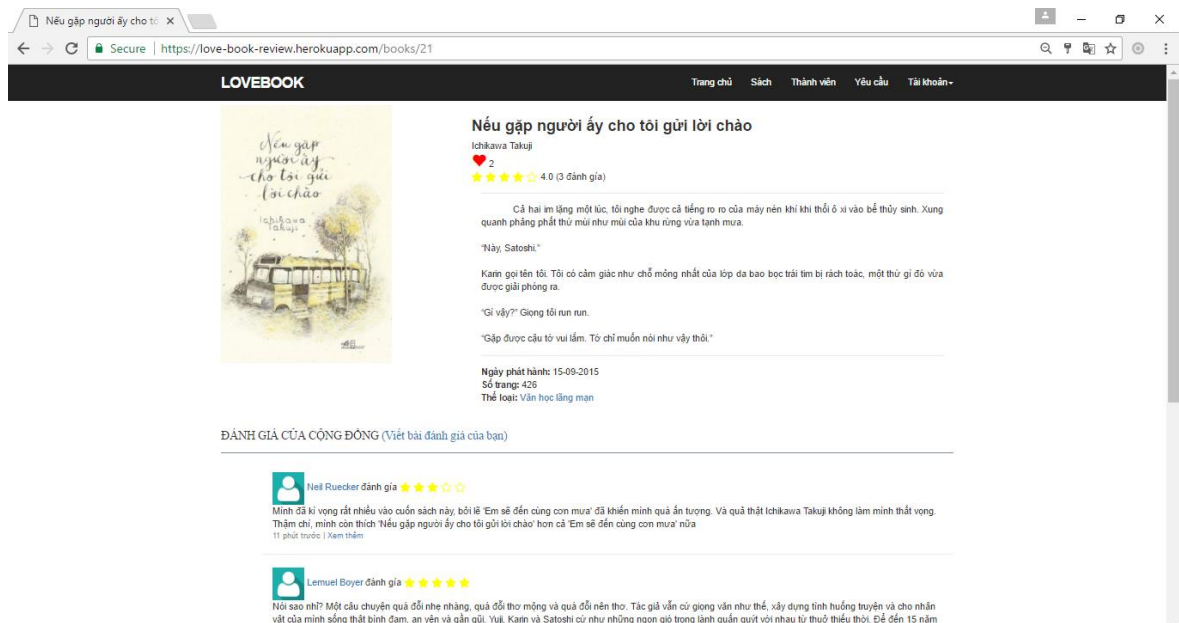


Hình 49: Giao diện trang đăng ký

4.1.3. Giao diện trang xem sách và tìm kiếm sách

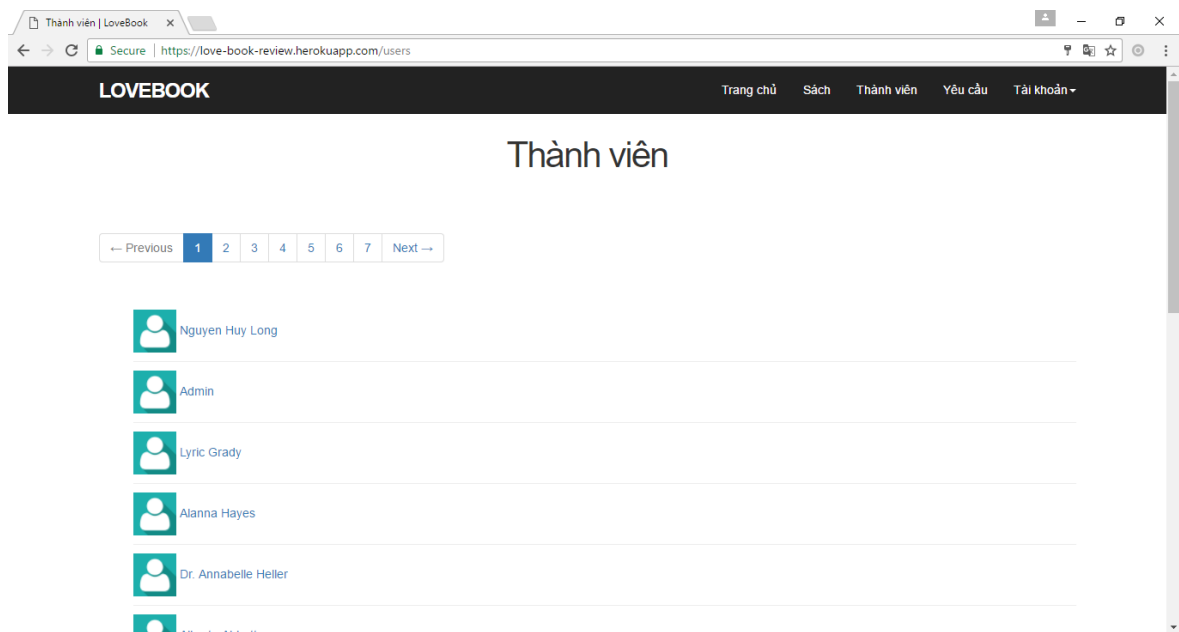


Hình 50: Giao diện trang hiển thị danh sách và tìm kiếm sách

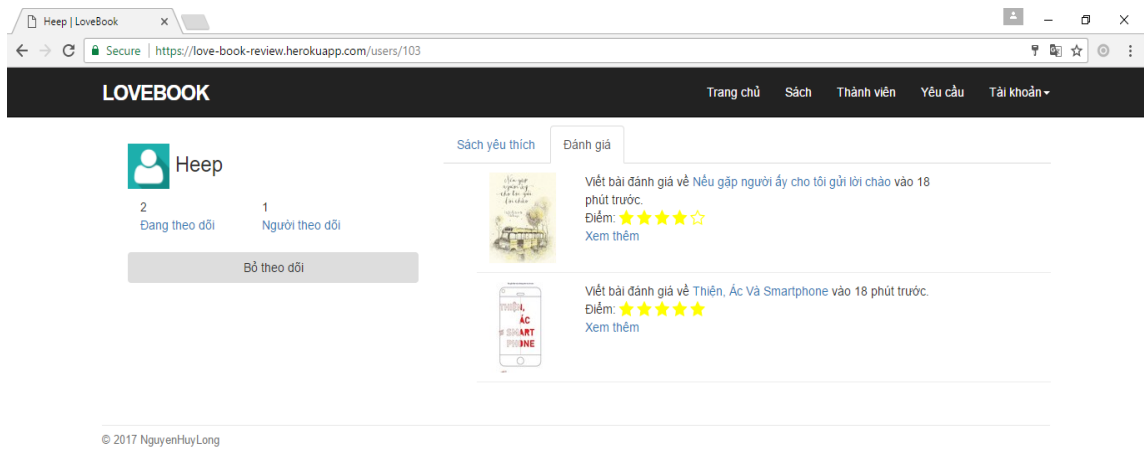


Hình 51: Giao diện trang hiển thị thông tin một cuốn sách cụ thể

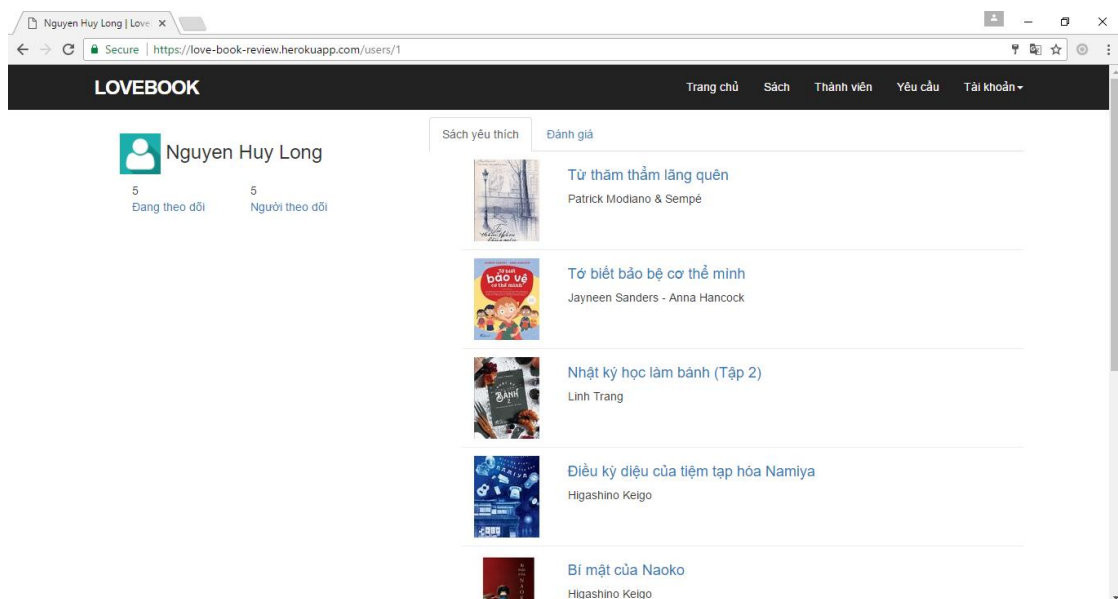
4.1.4. Giao diện đối với thành viên



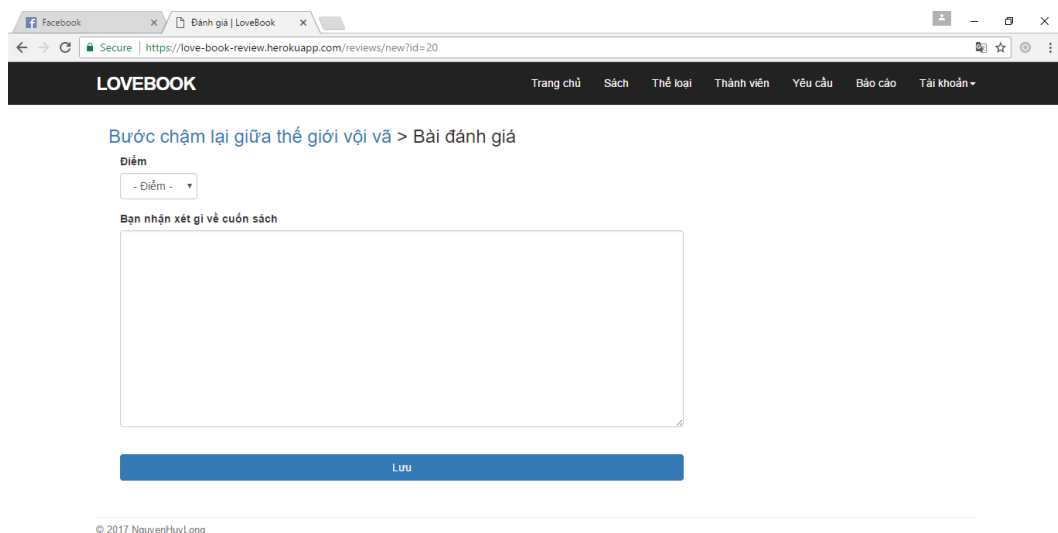
Hình 52: Giao diện trang hiển thị danh sách thành viên



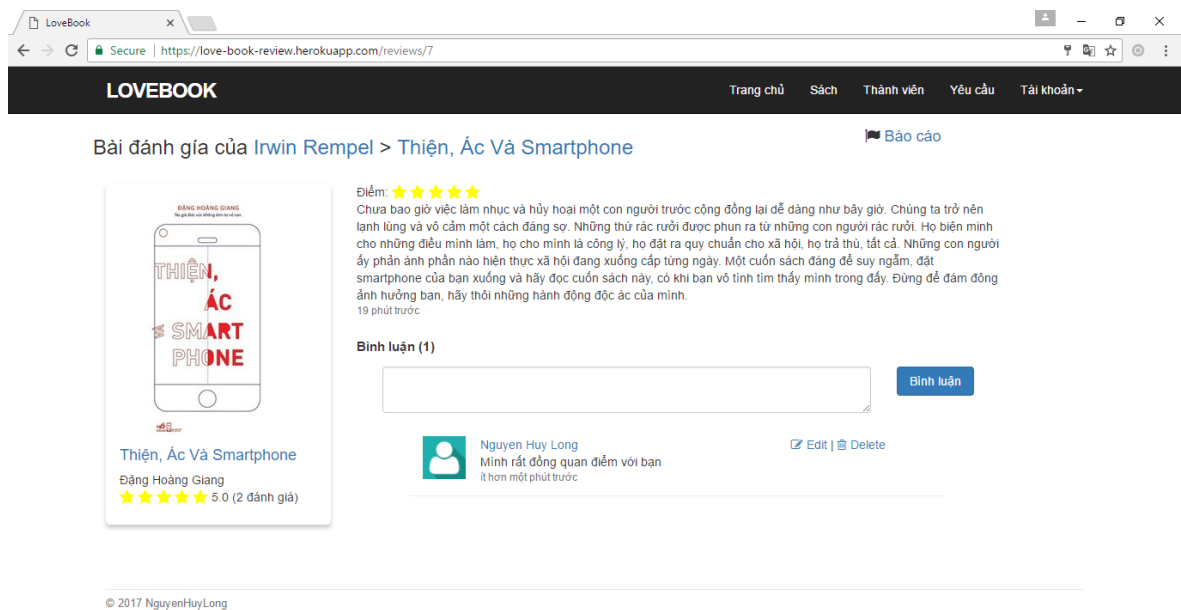
Hình 53: Giao diện trang cá nhân (1)



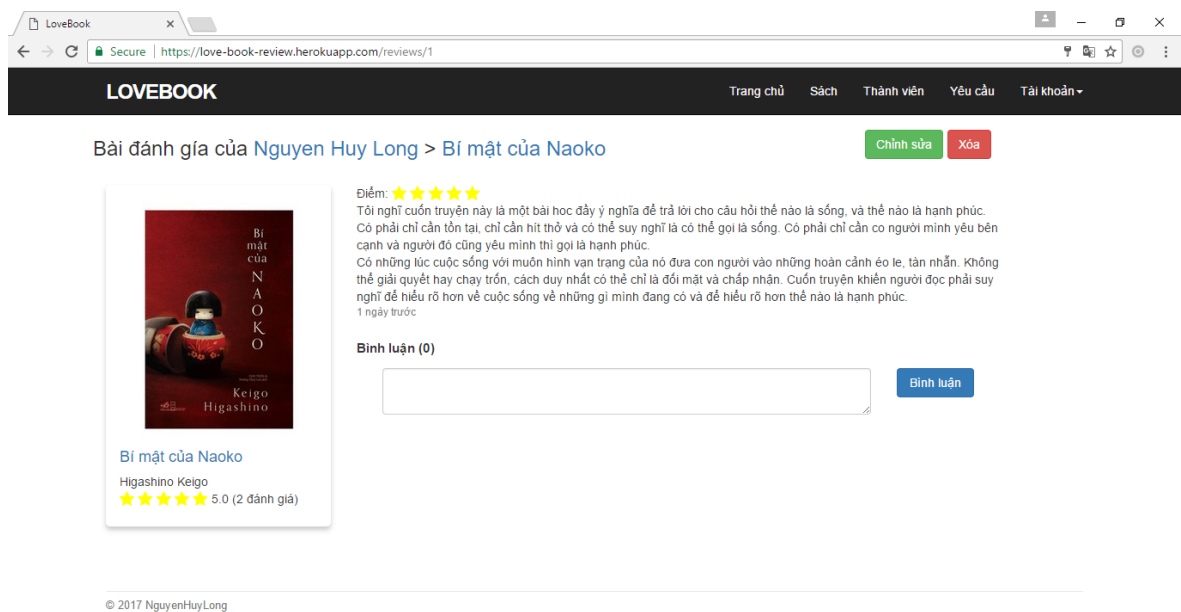
Hình 54: Giao diện trang cá nhân (2)



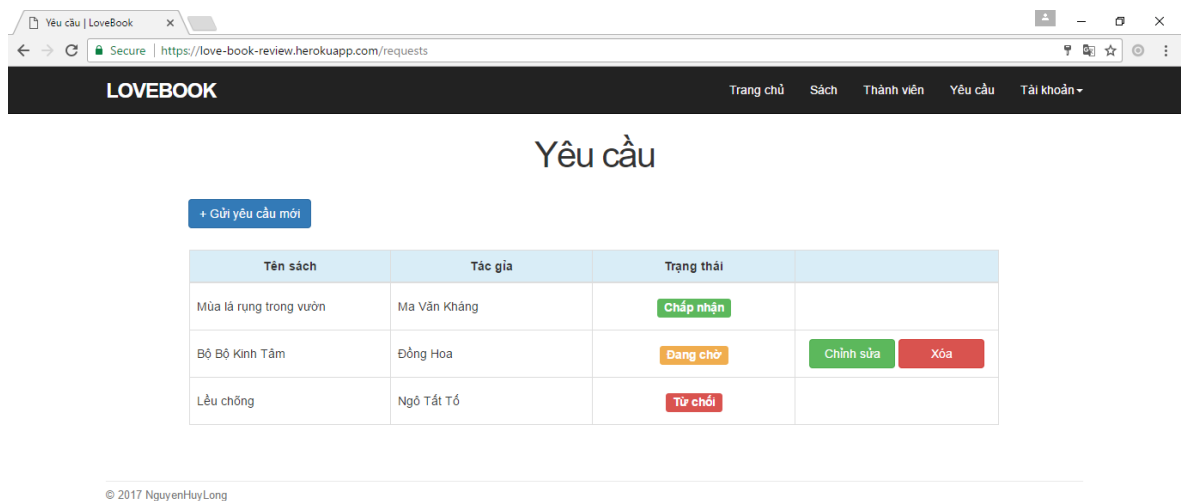
Hình 55: Giao diện trang viết bài đánh giá



Hình 56: Giao diện xem chi tiết một bài đánh giá của người dùng khác

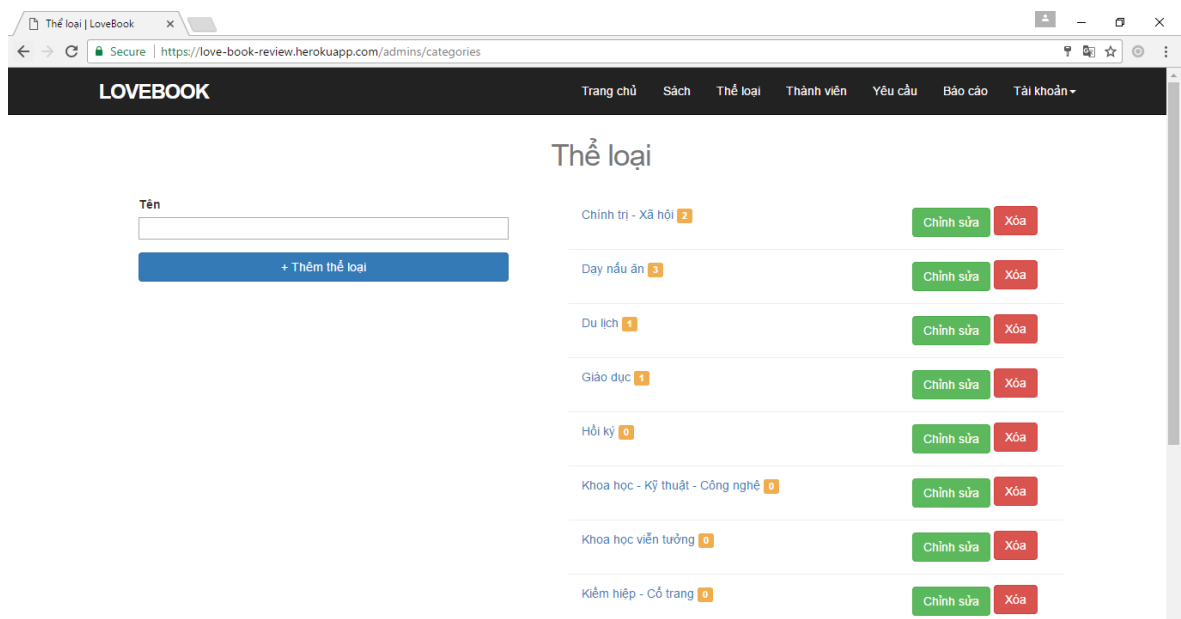


Hình 57: Giao diện xem chi tiết một bài đánh giá của tài khoản đang đăng nhập



Hình 58: Giao diện trang quản lý yêu cầu của thành viên

4.1.5. Giao diện đối với quản trị viên



Hình 59: Giao diện trang quản lý thể loại sách của quản trị viên

Facebook x Sách | LoveBook x

Secure | https://love-book-review.herokuapp.com/admins/books/new

LOVEBOOK Trang chủ Sách Thể loại Thành viên Yêu cầu Báo cáo Tài khoản

Thêm sách mới

Tên sách

Tác giả

Giới thiệu

Ngày phát hành

Số trang

Thể loại

Ảnh bìa No file chosen

© 2017 NguyenHuyLong

Hình 60: Giao diện trang thêm sách mới của quản trị viên

Thành viên | LoveBook x

Secure | https://love-book-review.herokuapp.com/users

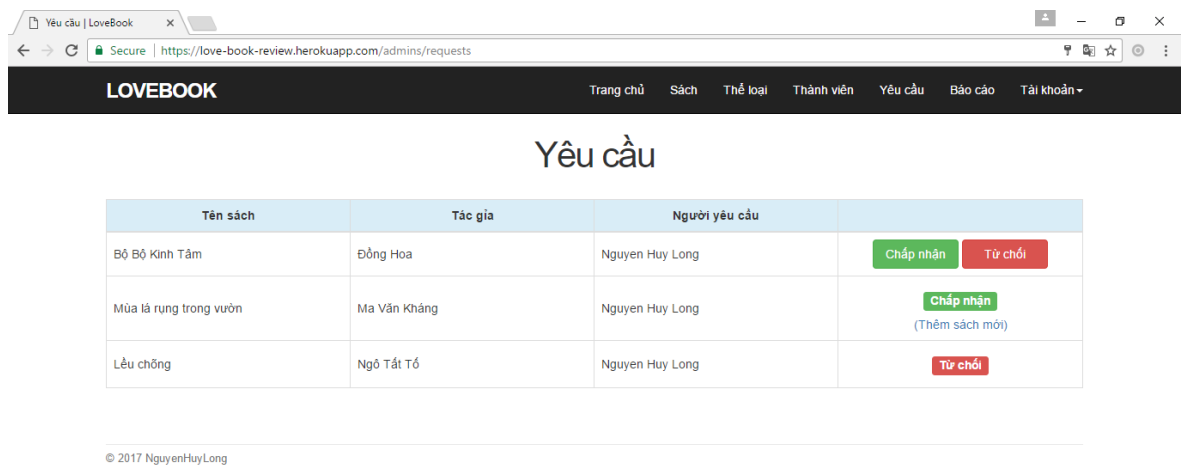
LOVEBOOK Trang chủ Sách Thể loại Thành viên Yêu cầu Báo cáo Tài khoản

Thành viên

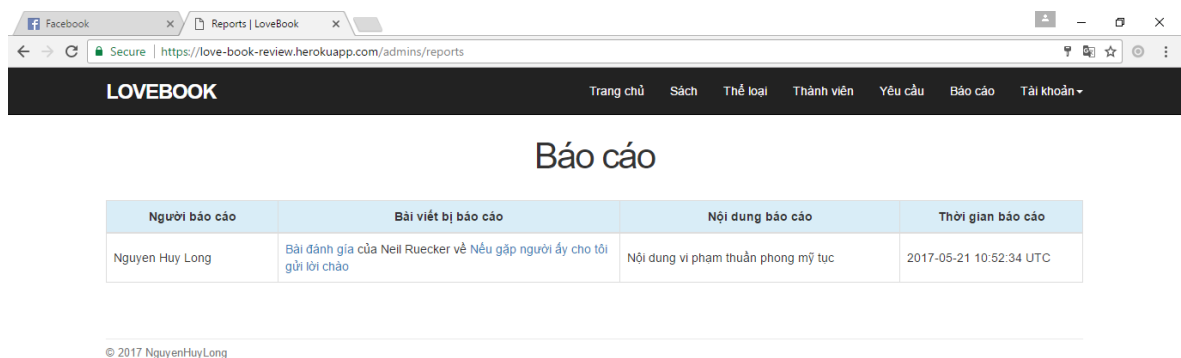
← Previous 1 2 3 4 5 6 7 Next →

	Admin	
	Lyric Grady Xóa	
	Alanna Hayes Xóa	
	Dr. Annabelle Heller Xóa	
	Alberto Abbott Xóa	

Hình 61: Giao diện trang quản lý thành viên của quản trị viên



Hình 62: Giao diện trang quản lý yêu cầu của quản trị viên



Hình 63: Giao diện trang quản lý báo cáo của quản trị viên

4.2. Đánh giá hệ thống

4.2.1. Ưu điểm

- Hệ thống hoạt động tốt trên nền tảng hệ điều hành Windows, Ubuntu và MacOS.
- Hệ thống hoạt động tốt trên các trình duyệt khác nhau Chrome, Cốc Cốc, Firefox...
- Các chức năng cơ bản đáp ứng đầy đủ các nhu cầu của các đối tượng người dùng khác nhau.

4.2.2. Nhược điểm

- Giao diện người dùng còn đơn giản, kém hấp dẫn người dùng

PHẦN 5 – KẾT LUẬN

5.1. Kết luận

Việc thực hiện đồ án tốt nghiệp đã tạo cơ hội tốt cho em áp dụng những kiến thức đã được học để phân tích, thiết kế và cài đặt một website hoàn chỉnh, phù hợp với thực tế. Sau một thời gian nghiên cứu đề tài, em đã thu được những kết quả sau:

- Tìm hiểu ngôn ngữ Ruby, Framework Rails cùng những công nghệ liên quan để xây dựng được một website đánh giá sách.
- Hoàn thành việc phân tích yêu cầu, thiết kế, xây dựng hệ thống, cài đặt và triển khai lên server thực tế.

5.2. Định hướng phát triển

- Về phía thành viên:
 - Mở rộng việc tạo tài khoản qua Facebook, Google, Twitter...
 - Cho phép chia sẻ những bài đánh giá sách hay qua Facebook, Google, Twitter...
- Về phía hệ thống:
 - Nghiên cứu tích hợp việc hiển thị quảng cáo cho các công ty sách và các website bán sách online.

TÀI LIỆU THAM KHẢO

- [1]. “Nghiên cứu của Harvard cho thấy đánh giá sách của độc giả Amazon đáng tin như của chuyên gia”,
<https://tinhte.vn/threads/nguyen-cuu-cua-harvard-cho-thay-danh-gia-sach-cua-doc-gia-amazon-dang-tin-nhu-cua-chuyen-gia.1265669/>,
truy cập lần cuối tháng 5 2017.
- [2]. “Về Ruby”, <https://www.ruby-lang.org/vi/about/>, truy cập lần cuối tháng 5 2017.
- [3]. Techmaster Team, “Ruby on Rails là gì?”,
<https://techmaster.vn/posts/7944/hoc-lap-trinh-web-bang-ruby-on-rails>,
truy cập lần cuối tháng 5 2017.11
- [4]. “Mô hình MVC và cấu trúc thư mục trong Rails app”,
<http://trenduongray.com/nhap-mon-rails-phan-2-mo-hinh-mvc-va-cau-truc-thu-muc-trong-rails-app/>, truy cập lần cuối tháng 5 2017.
- [5]. “SQLite”, <https://vi.wikipedia.org/wiki/SQLite>, truy cập lần cuối tháng 5 2017.
- [6]. “GitHub”, <https://vi.wikipedia.org/wiki/GitHub>, truy cập lần cuối tháng 5 2017.
- [7]. https://datvv.gitbooks.io/bao-cao-railsapp/content/chapter_4.html, truy cập lần cuối tháng 5 2017.
- [8]. Techmaster Team, “Kiến trúc của Ruby on Rails”,
<https://techmaster.vn/posts/7951/hoc-lap-trinh-web-bang-ruby-on-rails-2>,
truy cập lần cuối tháng 5 2017.
- [9]. Phung The Tai, “Tìm hiểu về RESTful web service”,
<https://viblo.asia/phung.the.tai/posts/OEqGj5JNM9bL>, truy cập lần cuối tháng 5 2017.