

ĐẠI HỌC QUỐC GIA HÀ NỘI
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ



BÁO CÁO BÀI TẬP LỚN CUỐI KỲ
WEBSITE QUẢN LÝ SÁCH

Bộ môn: Công nghệ phần mềm (INT2208 3)

Mã sinh viên

21020294
21020276
21020011
19020005

Họ và tên

Nguyễn Khánh Duy
Nguyễn Tuấn Anh
Nguyễn Trần Đạt
Hoàng Gia Anh Đức

Mục Lục

Mục Lục	iii
Danh sách hình ảnh	1
Danh sách bảng	3
1 Đặt vấn đề	4
2 Mô tả kiến thức nền tảng	6
2.1 Tổng quan về quản lý thư viện	6
2.1.1 Khái niệm về quản lý thư viện	6
2.1.2 Các mô hình đang được áp dụng trong quản lý thư viện	7
2.1.3 Lợi ích của mô hình thư viện số	7
2.2 Trình bày tổng quan về Framework, thư viện và DBMS	8
2.2.1 Giới thiệu	8
2.2.2 NodeJS	9
2.2.3 ExpressJS	9
2.2.4 Axios	10
2.2.5 ReactJS	10
2.2.6 MySQL	11
2.2.7 MongoDB	11
2.2.8 Cassandra	13
3 Phân tích đặc tả yêu cầu	16
3.1 Phân tích yêu cầu	16
3.1.1 Phân tích yêu cầu chung của hệ thống	16
3.2 Thu thập yêu cầu	17
3.2.1 Yêu cầu chức năng	17
3.2.2 Yêu cầu phi chức năng	18
3.3 Phân tích và đặc tả ca sử dụng	18

3.3.1	Sơ đồ use case	18
3.3.2	Ca sử dụng Đăng ký tài khoản	19
3.3.3	Ca sử dụng Đăng nhập tài khoản	21
3.3.4	Ca sử dụng Tra cứu sách	23
3.3.5	Ca sử dụng tra cứu hồ sơ, thông tin người dùng	25
3.3.6	Ca sử dụng Mượn/trả sách	26
3.3.7	Hiện thị thống kê những cuốn sách mượn nhiều nhất	29
4	Phân tích thiết kế và xây dựng hệ thống	32
4.1	Thiết kế kiến trúc chung của hệ thống	32
4.1.1	Phân tích kiến trúc	32
4.1.2	Mô hình MICROSERVICE	32
4.2	Biểu đồ tuần tự cho ca sử dụng chính	33
4.3	Thiết kế API	36
4.4	Thiết kế cơ sở dữ liệu	36
4.5	Thiết kế giao diện người dùng	38
5	Cài đặt và kiểm thử	40
5.1	Kiểm thử hệ thống	40
6	Kết luận và định hướng phát triển	47
6.1	Kết luận	47
6.2	Định hướng phát triển	47
	Tài liệu tham khảo	49

Danh sách hình ảnh

Hình ảnh trang web

1. Hình ảnh trang chủ của website (Hình 1.1).
2. Hình ảnh giao diện đăng ký tài khoản (Hình 3.3).
3. Hình ảnh giao diện đăng nhập (Hình 3.5).
4. Hình ảnh giao diện tìm kiếm sách (Hình 3.7).
5. Hình ảnh giao diện thông tin người dùng (Hình 3.9).
6. Hình ảnh giao diện của thông tin những cuốn sách đang mượn (Hình 3.11).
7. Hình ảnh giao diện của thông tin những cuốn sách đã trả (Hình 3.12).
8. Giao diện thống kê (Hình 3.14).

Hình ảnh các biểu đồ ca sử dụng

1. Hình ảnh sơ đồ Use-case (Hình 3.1).
2. Hình ảnh biểu đồ hoạt động ca sử dụng đăng kí (Hình 3.2).
3. Hình ảnh biểu đồ hoạt động ca sử dụng đăng đăng nhập (Hình 3.4).
4. Hình ảnh biểu đồ hoạt động ca sử dụng tìm kiếm sách (Hình 3.6).
5. Hình ảnh biểu đồ hoạt động ca sử dụng tra cứu hồ sơ, thông tin người dùng (Hình 3.8).
6. Hình ảnh biểu đồ hoạt động ca sử dụng mượn/trả sách (Hình 3.10).
7. Biểu đồ hoạt động ca sử dụng thống kê (Hình 3.13).

Hình ảnh biểu đồ tuần tự các ca hoạt động

1. Biểu đồ tuần tự cho ca sử dụng đăng nhập (Hình 4.1).
2. Biểu đồ tuần tự cho ca sử dụng đăng ký (Hình 4.2).
3. Biểu đồ tuần tự cho ca sử dụng mượn sách (Hình 4.3).
4. Biểu đồ tuần tự cho ca sử dụng trả sách (Hình 4.4).
5. Biểu đồ tuần tự cho ca sử dụng tra cứu sách (Hình 4.5).
6. Biểu đồ tuần tự cho ca sử dụng thống kê (Hình 4.6).

Hình ảnh về database

1. Hình ảnh lưu trữ tài khoản người dùng trên MySQL (Hình 2.1).
2. Hình ảnh lưu trữ sách trên MongoDB (Hình 2.2).
3. Hình ảnh lưu trữ lịch sử mượn trả trên Cassandra (Hình 2.3).
4. Hình ảnh cơ sở dữ liệu (Hình 4.7).

Danh sách bảng

Bảng mô tả các biểu đồ ca sử dụng

1. Bảng mô tả biểu đồ hoạt động ca sử dụng đăng kí (3.1).
2. Bảng mô tả biểu đồ hoạt động ca sử dụng đăng nhập (3.2).
3. Bảng mô tả biểu đồ hoạt động ca sử dụng tìm kiếm sách (3.3).
4. Bảng mô tả biểu đồ hoạt động ca sử dụng tra cứu hồ sơ, thông tin người dùng (3.4).
5. Bảng mô tả biểu đồ hoạt động ca sử dụng mượn/trả sách (3.5).
6. Bảng mô tả biểu đồ hoạt động ca sử dụng thống kê (3.6).

Bảng mô tả kiểm thử các chức năng

1. Kiểm thử đăng ký (5.1).
2. Kiểm thử đăng nhập (5.2).
3. Kiểm thử mượn sách (5.3).
4. Kiểm thử trả sách (5.4).
5. Kiểm thử tra cứu sách (5.5).
6. Kiểm thử thanh Navbar (5.6).
7. Kiểm thử thay đổi mật khẩu (5.7).
8. Kiểm thử thống kê mượn sách (5.3).
9. Kiểm thử thống kê (5.8).

Chương 1

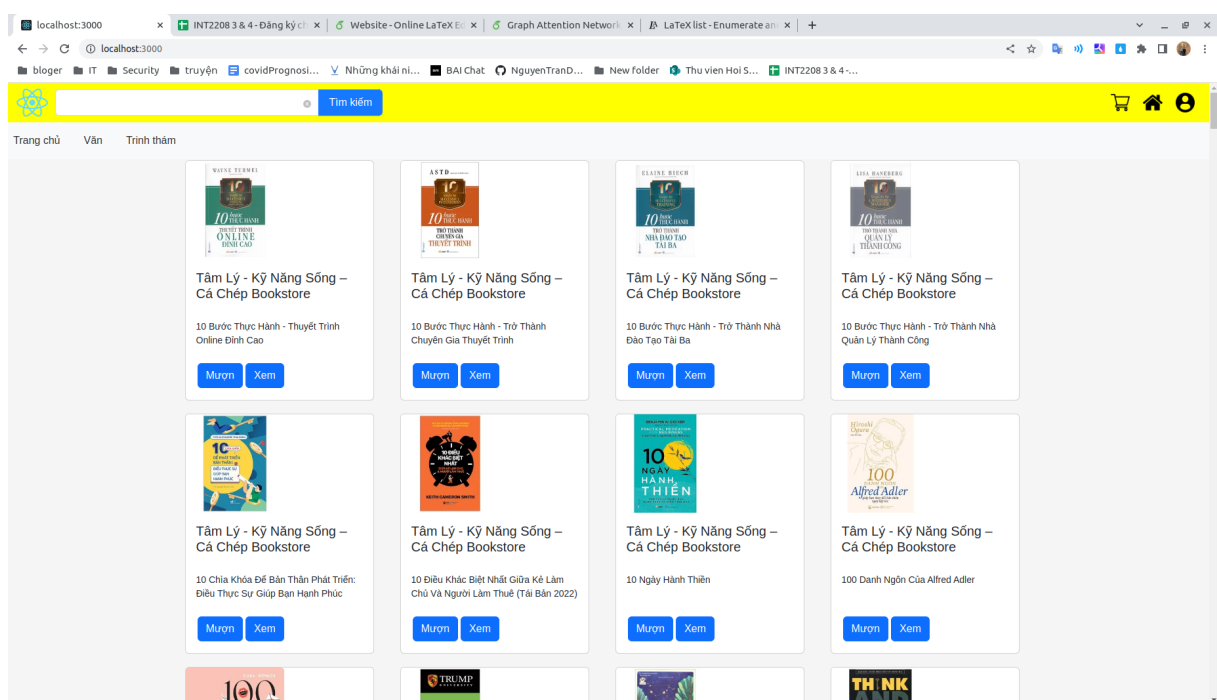
Đặt vấn đề

Trong xã hội ngày nay, nhu cầu đọc sách và tìm kiếm thông tin là điều không thể thiếu đối với mọi người. Tuy nhiên, việc tìm kiếm và quản lý các tài liệu đôi khi gặp nhiều khó khăn, đặc biệt là đối với các tổ chức như trường học, thư viện hay các công ty. Vì vậy, việc sử dụng một trang web quản lý thư viện trở nên cần thiết hơn bao giờ hết.

Một trang web quản lý thư viện sẽ giúp cho việc quản lý các tài liệu dễ dàng và hiệu quả hơn, từ việc lưu trữ, tìm kiếm cho đến việc theo dõi các mượn/trả sách. Bên cạnh đó, người dùng có thể tìm kiếm, đặt mượn và đặt trước các tài liệu một cách thuận tiện và nhanh chóng, giúp tiết kiệm thời gian và công sức.

Với sự phát triển không ngừng của công nghệ thông tin, việc xây dựng một trang web quản lý thư viện trở nên dễ dàng hơn bao giờ hết. Nếu được sử dụng đúng cách và phát triển thường xuyên, trang web quản lý thư viện sẽ giúp cho việc quản lý các tài liệu trở nên dễ dàng và thuận tiện hơn, từ đó giúp nâng cao chất lượng dịch vụ của các tổ chức và mang lại sự hài lòng cho người dùng.

Chính vì những lý do đó, nhóm chúng em đã thiết kế một website "Quản lý thư viện" với các chức năng: Hiển thị sách, mượn sách, xem được các sách đã mượn, xem được lịch sử mượn sách, tạo được tài khoản, xem thông tin cá nhân, thay đổi được mật khẩu. Và có thể đọc sách online (đọc sách online mới chỉ ở bản demo). Đối với nhân viên thư viện, website còn giúp thông kê 10 cuốn sách được mượn nhiều nhất.



Hình 1.1: Hình ảnh trang chủ của website.

Chương 2

Mô tả kiến thức nền tảng

2.1 Tổng quan về quản lý thư viện

Quản lý thư viện là quá trình tổ chức, bảo quản, phục vụ và cho mượn tài liệu trong thư viện. Nó cũng bao gồm việc tạo ra các chính sách, quy trình và tiêu chuẩn để đảm bảo việc quản lý tài liệu được thực hiện hiệu quả và đáp ứng nhu cầu của người dùng.

Trong quản lý thư viện, các tài liệu được phân loại và đánh số để có thể tìm kiếm và truy cập một cách dễ dàng. Các tài liệu này bao gồm sách, báo, tạp chí, tài liệu điện tử, phim và âm nhạc.

Các thư viện có thể hoạt động trên nhiều cấp độ, bao gồm thư viện cá nhân, thư viện học đường, thư viện công cộng, thư viện đại học và thư viện quốc gia. Mỗi cấp độ thư viện có quy mô và chức năng khác nhau, với các yêu cầu và thách thức riêng.

Với sự phát triển của công nghệ, quản lý thư viện đã có sự thay đổi đáng kể trong thời gian gần đây. Việc sử dụng các phần mềm quản lý thư viện và các hệ thống tự động hóa giúp quản lý thư viện trở nên dễ dàng hơn và hiệu quả hơn.

2.1.1 Khái niệm về quản lý thư viện

Quản lý thư viện là quá trình tổ chức, quản lý và duy trì một bộ sưu tập tài liệu, sách và tài nguyên thông tin khác trong một thư viện. Nó bao gồm các hoạt động như mua sắm, phân loại, lưu trữ, phục vụ và bảo quản tài liệu trong thư viện. Quản lý thư viện cũng đảm bảo việc cung cấp dịch vụ thư viện hiệu quả, bao gồm tìm kiếm tài liệu, hướng dẫn sử dụng, giáo dục người dùng và tiếp thị. Quản lý thư viện cũng có thể bao gồm việc quản lý các hoạt động liên quan đến thư viện, bao gồm quản lý tài chính, nhân sự và hợp

đồng.

2.1.2 Các mô hình đang được áp dụng trong quản lý thư viện

Có nhiều mô hình quản lý thư viện được áp dụng hiện nay, tùy thuộc vào quy mô và mục đích của thư viện đó. Sau đây là một số mô hình quản lý thư viện phổ biến:

1. Mô hình truyền thống: đây là mô hình quản lý thư viện phổ biến nhất và đã được áp dụng trong nhiều năm. Theo mô hình này, thư viện có trách nhiệm thu thập, bảo quản và cung cấp tài liệu cho người đọc. Các tài liệu được phân loại và sắp xếp theo thể loại và đưa vào kho sách, thư viện cung cấp thẻ thư viện cho người đọc để mượn sách và giám sát việc mượn trả sách.
2. Mô hình thư viện số: mô hình này sử dụng công nghệ số để quản lý và cung cấp tài liệu. Thay vì mượn sách trực tiếp, người đọc sử dụng các thiết bị điện tử để truy cập tài liệu và đọc trực tuyến. Thư viện số có thể cung cấp các tài liệu mà không bị giới hạn bởi không gian vật lý.
3. Mô hình thư viện hỗn hợp: mô hình này kết hợp cả thư viện truyền thống và thư viện số. Người đọc có thể mượn sách trực tiếp từ thư viện và cũng có thể truy cập tài liệu trực tuyến.
4. Mô hình thư viện tự phục vụ: mô hình này cho phép người đọc tự truy cập và mượn sách mà không cần sự giám sát của thủ thư. Người đọc sử dụng các thiết bị tự phục vụ để tìm kiếm, mượn và trả sách.
5. Mô hình thư viện cộng đồng: mô hình này đặt trọng tâm vào các hoạt động văn hóa và giáo dục của cộng đồng, thay vì chỉ tập trung vào cung cấp tài liệu. Thư viện cộng đồng có thể tổ chức các sự kiện văn hóa, hội thảo, đọc truyện cho trẻ em, cung cấp các dịch vụ giáo dục cho người lớn tuổi, v.v.
6. Mô hình thư viện di động: mô hình này cho phép thư viện đến gần với người dân, đặc biệt là những khu vực xa xôi hoặc khó tiếp cận.

2.1.3 Lợi ích của mô hình thư viện số

Mô hình thư viện số (digital library) mang lại nhiều lợi ích cho cả người quản lý thư viện và người sử dụng. Sau đây là một số lợi ích của mô hình thư viện số:

1. Tiết kiệm chi phí: Mô hình thư viện số cho phép tiết kiệm chi phí về không gian lưu trữ, vận chuyển tài liệu, và tiết kiệm chi phí cho các hoạt động liên quan đến việc bảo quản sách, tạp chí, tài liệu.
2. Tiện lợi: Người dùng có thể truy cập thư viện số 24/7 bất kể nơi đâu, không cần phải đến trực tiếp thư viện để mượn sách.
3. Khả năng tìm kiếm tài liệu nhanh chóng: Mô hình thư viện số cho phép người dùng tìm kiếm tài liệu nhanh chóng và chính xác hơn bằng cách sử dụng các công cụ tìm kiếm mạnh mẽ, đồng thời cung cấp thêm các thông tin phụ trợ như nhận xét, đánh giá từ người dùng khác, giúp người dùng có được sự lựa chọn phù hợp với nhu cầu của mình.
4. Dễ dàng chia sẻ và tái sử dụng tài liệu: Tài liệu trong thư viện số có thể được dễ dàng chia sẻ với những người dùng khác, đồng thời tài liệu có thể được tái sử dụng nhiều lần mà không cần phải sản xuất lại.
5. Tính bảo mật cao: Mô hình thư viện số có thể cung cấp tính bảo mật cao đối với các tài liệu nhạy cảm và thông tin cá nhân của người dùng.
6. Tính khả dụng cao: Mô hình thư viện số cho phép thư viện mở rộng phạm vi phục vụ người dùng, bao gồm cả những người dùng ở xa, bị khuyết tật hoặc không có thời gian để đến thư viện.

2.2 Trình bày tổng quan về Framework, thư viện và DBMS

2.2.1 Giới thiệu

Framework là một cấu trúc phần mềm cung cấp cho lập trình viên một cách tiếp cận có hệ thống để xây dựng ứng dụng. Nó định nghĩa cách tổ chức, triển khai và chạy ứng dụng, và cung cấp một loạt các thư viện và công cụ để giúp lập trình viên xử lý các tác vụ phổ biến. Ví dụ : ExpressJS.

Thư viện là một bộ sưu tập các mã nguồn đã được viết trước và có sẵn để sử dụng trong việc phát triển ứng dụng web. Các thư viện cung cấp các chức năng và tính năng đã được xây dựng sẵn, giúp giảm bớt công sức và thời gian phát triển bằng cách cung cấp các phương thức và lớp đã được xây dựng sẵn để xử lý các tác vụ phổ biến. Ví dụ : ReactJS, Axios, Vue.js,...

DBMS là viết tắt của "Database Management System", tức là Hệ thống Quản lý Cơ sở dữ liệu trong tiếng Việt. Nó là một phần mềm hoặc hệ thống phần cứng được sử dụng để quản lý, tổ chức, lưu trữ và truy xuất dữ liệu trong cơ sở dữ liệu. DBMS cung cấp các công cụ và chức năng cho phép người dùng tạo, cập nhật và truy vấn cơ sở dữ liệu một cách dễ dàng và hiệu quả. Nó giúp đảm bảo tính nhất quán và an toàn của dữ liệu, và cung cấp các tính năng như kiểm soát truy cập, sao lưu và phục hồi dữ liệu. Ví dụ : MySQL, PostgreSQL, MongoDB, Cassandra,...

2.2.2 NodeJS

Node.js là một môi trường chạy mã JavaScript ở phía máy chủ (server-side runtime environment). Nó được xây dựng dựa trên Chrome V8 JavaScript engine và cho phép chạy mã JavaScript bên ngoài trình duyệt.

Node.js cho phép lập trình viên sử dụng JavaScript để phát triển ứng dụng máy chủ, không chỉ giới hạn trong việc xây dựng ứng dụng web mà còn cả các ứng dụng mạng và ứng dụng dòng lệnh. Với Node.js, JavaScript có thể được sử dụng để xây dựng các API, xử lý yêu cầu HTTP, quản lý các kết nối đến cơ sở dữ liệu, xử lý tác vụ hàng đợi và nhiều tác vụ khác liên quan đến phía máy chủ. Node.js có một hệ sinh thái phong phú của các module và thư viện mở rộng, cho phép lập trình viên sử dụng các module có sẵn hoặc tạo các module riêng để mở rộng khả năng của Node.js. Nó đã trở thành một công cụ phổ biến cho phát triển ứng dụng web hiệu năng cao, ứng dụng realtime, ứng dụng IoT và nhiều lĩnh vực phát triển phần mềm khác.

2.2.3 ExpressJS

Express.js là một framework phát triển ứng dụng web bằng ngôn ngữ JavaScript và chạy trên môi trường Node.js. Nó được xây dựng trên cơ sở HTTP module của Node.js và cung cấp một cách tiếp cận đơn giản và mạnh mẽ để xây dựng các ứng dụng web và API.

Express.js giúp đơn giản hóa việc xây dựng và quản lý các tác vụ phổ biến trong phát triển web, bao gồm định tuyến (routing), xử lý yêu cầu và phản hồi (request/response), quản lý middleware, xử lý lỗi, và nhiều chức năng khác. Nó giúp tổ chức mã nguồn dễ dàng và linh hoạt, giúp lập trình viên xây dựng ứng dụng web nhanh chóng và hiệu quả.

Express.js cũng hỗ trợ việc tích hợp các module và thư viện khác để bổ sung các tính năng cho ứng dụng, chẳng hạn như middleware PassportJS để xác thực người dùng, Sequelize để làm việc với cơ sở dữ liệu, Socket.IO để xây dựng ứng dụng realtime, và nhiều module khác.

Với cú pháp đơn giản và cộng đồng phát triển sôi nổi, Express.js đã trở thành một trong những framework phát triển ứng dụng web phổ biến nhất trong hệ sinh thái Node.js.

2.2.4 Axios

Axios là một thư viện JavaScript mã nguồn mở được sử dụng để gửi các yêu cầu HTTP từ phía client tới một máy chủ và xử lý các phản hồi từ máy chủ. Nó cung cấp một cách dễ dàng và linh hoạt để tương tác với các API từ phía client trong các ứng dụng web và mobile.

Axios hỗ trợ các phương thức HTTP như GET, POST, PUT, DELETE và PATCH và cung cấp các tính năng như xử lý các yêu cầu và phản hồi bất đồng bộ, quản lý các tiêu đề yêu cầu và phản hồi, xử lý các lỗi mạng, và các chức năng liên quan đến việc tương tác với API.

Axios là một thư viện JavaScript phổ biến để gửi các yêu cầu HTTP và xử lý phản hồi từ máy chủ, giúp tương tác với các API một cách dễ dàng và hiệu quả trong các ứng dụng web và mobile.

2.2.5 ReactJS

ReactJS là một thư viện JavaScript mã nguồn mở được sử dụng để xây dựng giao diện người dùng (UI) trong các ứng dụng web. Nó được phát triển bởi Facebook và cộng đồng React và đã trở thành một trong những công cụ phát triển phổ biến nhất cho việc xây dựng giao diện người dùng động và tương tác.

ReactJS tập trung vào việc xây dựng các thành phần UI tái sử dụng (reusable UI components), giúp tách giao diện thành các phần nhỏ, độc lập và dễ quản lý. Nó sử dụng cú pháp JSX để kết hợp HTML và JavaScript để mô tả giao diện.

ReactJS là một thư viện JavaScript mạnh mẽ cho việc xây dựng giao diện người dùng linh hoạt, dễ quản lý và hiệu suất cao trong các ứng dụng web.

Ngoài ra, để giúp thao tác nhanh hơn, nhóm chúng em còn sử dụng lazy loading cho trang web, hiển thị theo thời gian loading của trang web.

2.2.6 MySQL

MySQL sử dụng ngôn ngữ truy vấn cấu trúc (Structured Query Language - SQL) để tương tác với cơ sở dữ liệu. Nó cung cấp một hệ thống quản lý cơ sở dữ liệu mạnh mẽ và linh hoạt, hỗ trợ các tính năng như tạo, xóa, cập nhật và truy vấn dữ liệu. Nó hỗ trợ đa người dùng, đa luồng, cung cấp tính bảo mật và ổn định.

MySQL có thể được triển khai trên nhiều hệ điều hành khác nhau, bao gồm Windows, macOS và các phiên bản của Linux. Ngoài ra, nó cũng hỗ trợ các ngôn ngữ lập trình phổ biến như PHP, Python, Java và nhiều ngôn ngữ khác để tương tác với cơ sở dữ liệu.

MySQL cũng được đánh giá cao về khả năng mở rộng, với khả năng hỗ trợ hàng triệu lượt truy cập một ngày và có thể mở rộng lên đến hàng trăm máy chủ. Điều này làm cho MySQL trở thành một giải pháp lý tưởng cho các ứng dụng web có lưu lượng truy cập lớn.

Ngoài ra, MySQL còn có tính năng bảo mật tốt, bao gồm các cơ chế xác thực và phân quyền cấp độ người dùng, giúp bảo vệ dữ liệu khỏi các cuộc tấn công độc hại.

Từ những ưu điểm trên, nhóm em quyết định sử dụng MySQL làm một database lưu thông tin tài khoản người dùng.

Trong MySQL chúng em sử dụng một số kiểu dữ liệu như INT, VARCHAR. Và sử dụng tìm kiếm trên id để hỗ trợ tìm kiếm nhanh hơn, mạnh hơn, tối ưu hơn.

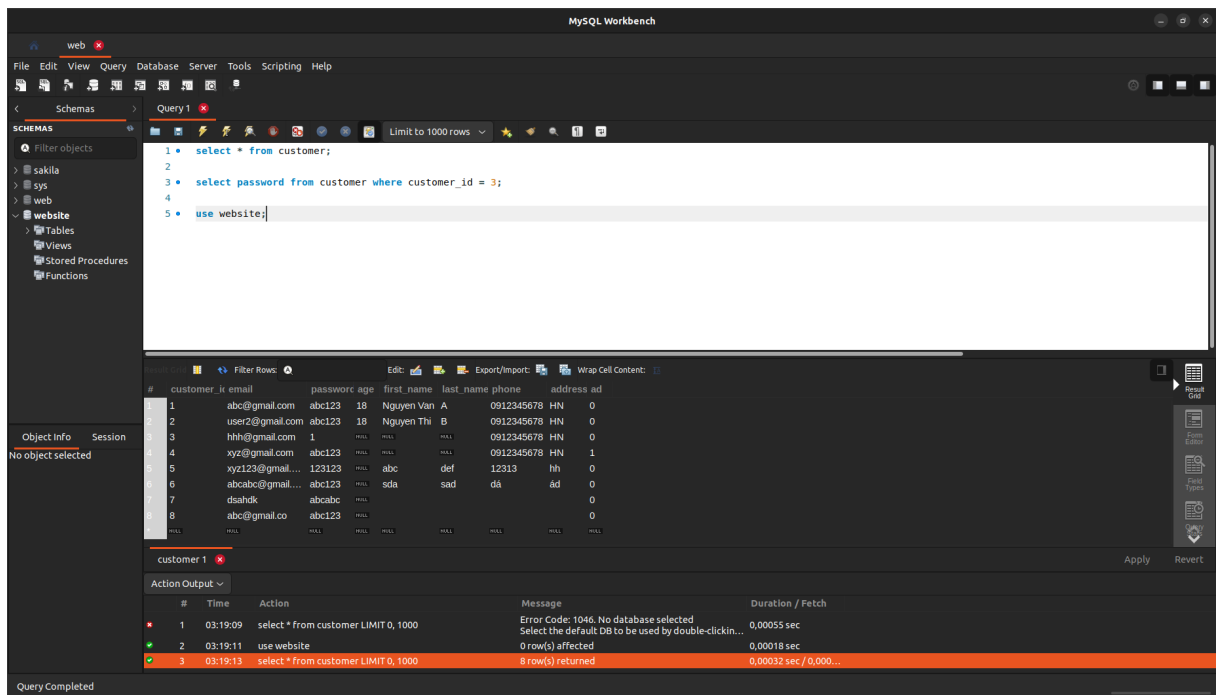
2.2.7 MongoDB

MongoDB là một hệ quản trị cơ sở dữ liệu phi quan hệ (non-relational database management system - NoSQL DBMS) phổ biến và mã nguồn mở. Nó được phát triển bởi MongoDB Inc. và là một phần của phong cách cơ sở dữ liệu không cần lược đồ (schema-less).

MongoDB lưu trữ dữ liệu dưới dạng tài liệu (document), sử dụng định dạng JSON-like gọi là BSON (Binary JSON). Mỗi tài liệu trong MongoDB được lưu trữ trong một collection, tương tự như bảng trong cơ sở dữ liệu quan hệ. Tuy nhiên, trong MongoDB, các tài liệu trong cùng một collection không cần phải có cấu trúc dữ liệu giống nhau.

Một số đặc điểm và lợi ích của MongoDB bao gồm:

- Linh hoạt: MongoDB cho phép thay đổi cấu trúc dữ liệu một cách dễ dàng, không



Hình 2.1: Hình ảnh lưu trữ tài khoản người dùng trên MySQL

cần phải thay đổi lược đồ trước.

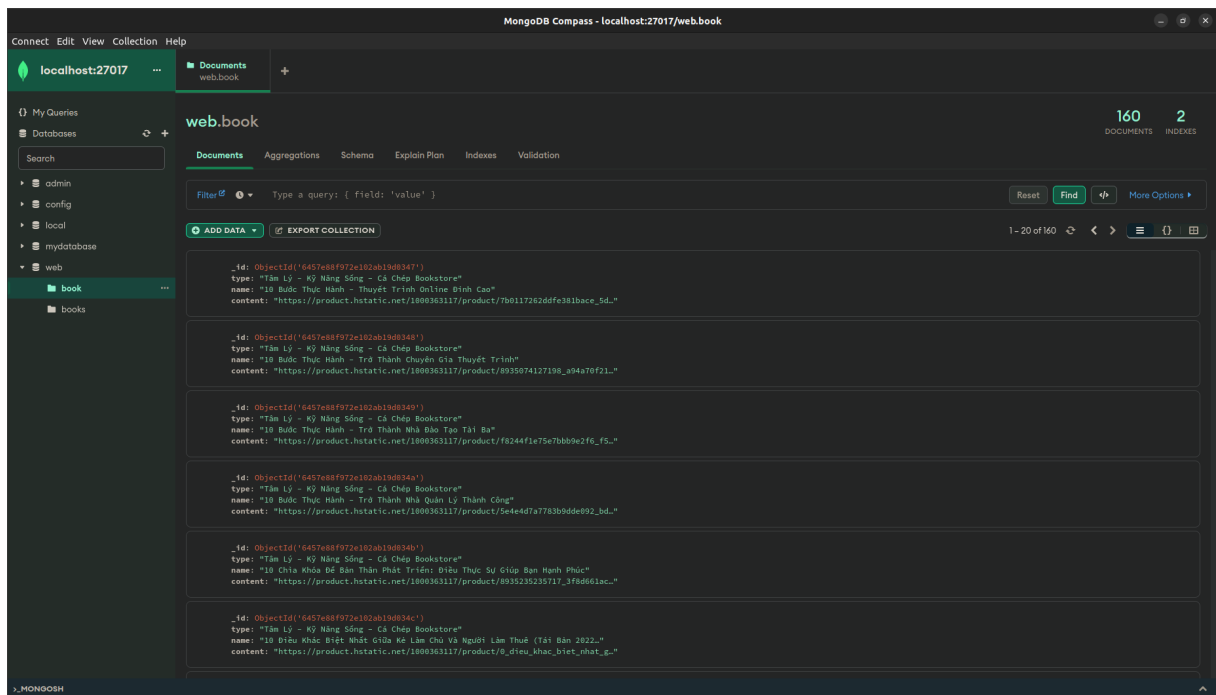
- Mở rộng: Nó hỗ trợ mô hình mở rộng ngang (horizontal scaling), cho phép mở rộng hệ thống dựa trên khả năng xử lý của nhiều máy chủ.
- Tính dự phòng và phục hồi: MongoDB cung cấp các tính năng dự phòng và phục hồi dữ liệu, như replication và sharding, để đảm bảo tính sẵn sàng và bảo mật dữ liệu.
- Truy vấn linh hoạt: Nó hỗ trợ các truy vấn phong phú và mạnh mẽ, bao gồm các truy vấn tìm kiếm, so sánh, phân tích và kết hợp dữ liệu.

MongoDB được sử dụng rộng rãi trong các ứng dụng web, mobile và Internet of Things (IoT), nơi linh hoạt và khả năng mở rộng của nó là điểm mạnh.

Từ những ưu điểm trên, nhóm em đã quyết định sử dụng MongoDB để lưu thông tin về sách.

Trong MongoDB, nhóm em sử dụng một số kiểu dữ liệu như: ObjectId, text. Nhóm em cũng hỗ trợ tìm kiếm sách dựa trên tên và thể loại của sách. Ngoài ra, để hỗ trợ tìm kiếm, nhóm em cũng sử dụng tìm kiếm trên id để giúp tìm kiếm sách một cách nhanh chóng và hiệu quả. Không những thế, nhóm cũng đang phát triển tìm kiếm dựa trên từ

gợi ý để hỗ trợ người dùng truy xuất dựa trên những từ đã nhập vào nhưng bị thiếu (nhóm đang phát triển đối với việc người dùng nhập sai).



Hình 2.2: Hình ảnh lưu trữ sách trên MongoDB

2.2.8 Cassandra

Cassandra là một hệ quản trị cơ sở dữ liệu phi quan hệ (non-relational database management system - NoSQL DBMS) phân tán và mã nguồn mở. Nó được thiết kế để xử lý dữ liệu lớn và đảm bảo tính sẵn sàng và khả năng mở rộng cao.

Cassandra được phát triển bởi Apache Software Foundation và được xây dựng dựa trên các nguyên tắc của mô hình cột (column-oriented model) và phân tán trên nhiều máy chủ (distributed architecture). Nó sử dụng mô hình dữ liệu cấu trúc dạng cột để lưu trữ và truy vấn dữ liệu.

Một số đặc điểm và lợi ích của Cassandra bao gồm:

- **Tính mở rộng ngang:** Cassandra được thiết kế để mở rộng theo chiều ngang với khả năng thêm máy chủ một cách dễ dàng để xử lý tải lớn và dữ liệu phân tán.
- **Tính sẵn sàng và bền vững:** Nó cung cấp tính sẵn sàng cao thông qua việc sao chép dữ liệu trên nhiều máy chủ và có khả năng tự phục hồi khi xảy ra sự cố.
- **Hỗ trợ truy vấn linh hoạt:** Cassandra cung cấp ngôn ngữ truy vấn CQL (Cassandra Query Language) để truy vấn dữ liệu, hỗ trợ các truy vấn phức tạp và linh hoạt.


```
trandat@trandat-Nitro-ANS15-S2: ~
bash: export: '=': not a valid identifier
bash: export: ':/home/hadoop/cassandra_jars/*': not a valid identifier
(base) trandat@trandat-Nitro-ANS15-S2: ~$ cqlsh
Connected to test cluster at 127.0.0.1:9042
[cqlsh 6.1.0 | Cassandra 4.1.1 | CQL spec 3.4.6 | Native protocol v5]
Use HELP for help.
cqlsh> select * from muon;
InvalidRequest: Error from server: code=2200 [Invalid query] message="No keyspace has been specified. USE a keyspace, or explicitly specify keyspace.tablename"
cqlsh> use web;
cqlsh:web> select * from muon;

muon_id | book_id | customer_id | rental_date | rental_id
-----|-----|-----|-----|-----
1964b7f7-92a6-4278-b805-62e405de1ef2 | 6457e88f972e102ab19d037c | 4 | 2023-05-14 20:05:26.493000+0000 | null
32410901-38db-407c-84f0-b002e435500e | 6457e88f972e102ab19d0347 | 4 | 2023-05-14 19:11:28.189000+0000 | null
f3da0389-ddd3-43df-9aea-5900340fb1c7 | 6457e88f972e102ab19d0349 | 4 | 2023-05-14 19:11:30.561000+0000 | null
bcdca65-9aac-4dff-b76c-4f4febae52e9 | 6457e88f972e102ab19d0364 | 4 | 2023-05-14 20:02:13.368000+0000 | null
(4 rows)
cqlsh:web> select * from tra;

tra_id | book_id | customer_id | return_date
-----|-----|-----|-----
1f138e47-e524-4b68-b906-52fc23b7764a | 6457e88f972e102ab19d0356 | 1 | 2023-05-14 20:03:04.581000+0000
ab4fc934-11f0-42d3-913a-1efd763bcc03 | 6457e88f972e102ab19d034c | 4 | 2023-05-14 20:02:21.597000+0000
a9591e9f-edcd-4c18-9c50-91970a51a4ec | 6457e88f972e102ab19d0349 | 4 | 2023-05-14 18:45:22.478000+0000
5fdeff5d-0f3d-4dfe-bc1f-4f7d22a614e | 6457e88f972e102ab19d0349 | 4 | 2023-05-14 18:44:28.971000+0000
5f3f6194-5d1f-4023-affa-1c54837bf434 | 6457e88f972e102ab19d0347 | 1 | 2023-05-14 20:02:58.524000+0000
4d92add0-f24d-4f59-b809-2f3ecbb2a405 | 6457e88f972e102ab19d0353 | 1 | 2023-05-14 20:03:00.577000+0000
82fff415-5e04-474d-8080-a0539227c3b4 | 6457e88f972e102ab19d0347 | 4 | 2023-05-14 18:45:19.062000+0000
73a6e6d6-4152-4b03-b1c1-ab38e27fb1bb | 6457e88f972e102ab19d0354 | 1 | 2023-05-14 20:02:56.479000+0000
00c3d050-e6b1-47bc-ae28-e64169e582ee | 6457e88f972e102ab19d0353 | 1 | 2023-05-14 20:03:00.577000+0000
bf7eabf2-650c-4641-8647-5e541db07921 | 6457e88f972e102ab19d0347 | 4 | 2023-05-14 18:44:21.481000+0000
474511b0-a1f6-4c0a-9780-c473433b3b1 | 6457e88f972e102ab19d0347 | 4 | 2023-05-14 18:45:19.063000+0000
4125b910-2c0d-44a6-917b-45daec3113c1 | 6457e88f972e102ab19d034c | 4 | 2023-05-14 20:02:21.597000+0000
a9c5f899-e75b-43ca-9a82-781f77b485b1 | 6457e88f972e102ab19d0347 | 4 | 2023-05-14 18:45:19.064000+0000
c43df093-0652-4250-845d-511ea0ff807c | 6457e88f972e102ab19d0347 | 1 | 2023-05-14 20:02:58.524000+0000
b3b45261-51b1-49b6-a4e0-ea16e92157b2 | 6457e88f972e102ab19d034e | 4 | 2023-05-14 18:44:26.523000+0000
cfe99ecd-bdfe-4ff3-95fd-81d52c4fdfcc | 6457e88f972e102ab19d0348 | 4 | 2023-05-14 18:44:23.647000+0000
87fc29e2-bd14-4b9b-9c43-b5b6f2eb1400 | 6457e88f972e102ab19d034c | 4 | 2023-05-14 20:02:21.597000+0000
(17 rows)
cqlsh:web> 
```

Hình 2.3: Hình ảnh lưu trữ lịch sử mượn trả trên Cassandra

- Tích hợp dễ dàng với ứng dụng: Cassandra cung cấp các giao diện và API cho phép lập trình viên dễ dàng tích hợp và tương tác với cơ sở dữ liệu.

Cassandra được sử dụng trong nhiều lĩnh vực, bao gồm các ứng dụng web với dữ liệu lớn, hệ thống IoT, cơ sở hạ tầng mạng xã hội, và các hệ thống phân tán khác đòi hỏi tính mở rộng và khả năng xử lý dữ liệu phân tán.

Từ những ưu điểm trên, nhóm em đã quyết định sử dụng Cassandra để lưu thông tin lịch sử mượn/trả sách.

Trong Cassandra, nhóm em sử dụng một số kiểu dữ liệu như: uuid, text, int, timestamp để lưu trữ. Nhóm em đang cố gắng phát triển để thông kê số lần mượn của một cuốn sách và tự động tính tiền khi khách hàng mượn quá số ngày quy định.

Việc sử dụng nhiều database giúp cho thông tin khách hàng và sản phẩm phân tán. Điều đó giúp ứng dụng của chúng em bảo mật hơn. Đồng thời, việc sử dụng nhiều database cũng hỗ trợ được nhược điểm của việc sử dụng một database gây nên, và cũng hỗ trợ cải thiện, nâng cấp khi có thay đổi. Không những thế chúng ta còn tận dụng được ưu điểm của các database khi lưu trữ. Tuy nhiên, việc sử dụng nhiều database cũng gây ra hiện tượng dữ liệu không đồng nhất. Đây cũng là vấn đề nhóm đang hướng tới trong tương lai.

Tổng kết chương: Với sự kết hợp của các framework, thư viện và hệ quản trị cơ

sở dữ liệu, website quản lí thư viện được xây dựng để giúp người quản lí có thể quản lí được hàng trăm đầu sách, những ca mượn, trả sách một cách có hệ thống và đảm bảo được bảo mật, tránh tình trạng mất sách.

Chương 3

Phân tích đặc tả yêu cầu

3.1 Phân tích yêu cầu

3.1.1 Phân tích yêu cầu chung của hệ thống

Công việc quản lý thư viện là một công việc phức tạp, đòi hỏi người quản lý phải linh hoạt, nhạy bén. Một số ứng dụng được tạo nên để hỗ trợ người quản lý một phần công việc. Website quản lý thư viện cũng là một ứng dụng như thế. Một số yêu cầu về Website quản lý thư viện bao gồm:

1. Đăng ký thành viên: Hệ thống cần cung cấp đăng ký thành viên để cho phép người dùng đăng nhập và sử dụng các chức năng của thư viện như mượn sách, đặt sách, tìm kiếm sách, và quản lý thông tin cá nhân (Đã làm được).
2. Quản lý thông tin sách: Hệ thống cần có chức năng quản lý thông tin về sách, bao gồm tên sách, tác giả, mã ISBN, năm xuất bản, nhà xuất bản, thể loại và số lượng sách trong thư viện (Đã làm được).
3. Quản lý mượn sách: Hệ thống cần cho phép người dùng mượn sách và lưu trữ thông tin về sách đã mượn, thời gian mượn (Đã làm được).
4. Tìm kiếm sách: Hệ thống cần có công cụ tìm kiếm sách để cho phép người dùng tìm kiếm sách theo tiêu chí khác nhau, bao gồm tên sách, thể loại (Đã làm được).
5. Quản lý thông tin cá nhân: Hệ thống cần cho phép người dùng quản lý thông tin cá nhân, bao gồm đổi mật khẩu, cập nhật thông tin liên lạc và xem lịch sử mượn sách (Đã làm được).

6. Thống kê: Hệ thống cần thống kê các quyển sách có số lượng người dùng nhiều nhất để hệ thống có thể xác định những cuốn sách có số lượng người dùng nhiều nhất để có thể đưa ra các quyết định chính xác về việc quản lý sách. Bên cạnh đó, thông kê sách cũng giúp cho quản lý thư viện nắm bắt được xu hướng đọc của độc giả và đưa ra các kế hoạch phù hợp để đáp ứng nhu cầu của độc giả.
- Và một số những yêu cầu khác như đa ngôn ngữ, quản lý nhân viên, quản lý hoạt động, v.v. Nhóm mà nhóm cũng đang hướng tới trong tương lai.

3.2 Thu thập yêu cầu

3.2.1 Yêu cầu chức năng

Yêu cầu người đọc:

- Người dùng có thể mượn sách
- Người dùng có thể xem sách
- Người dùng có thể tìm kiếm sách qua thanh tìm kiếm, có thể tìm kiếm qua các từ gợi ý.
- Người dùng có thể quản lý thông tin đăng nhập tên, mật khẩu
- Người dùng có thể xem được sách đang mượn
- Người dùng có thể trả sách

Yêu cầu người quản lý:

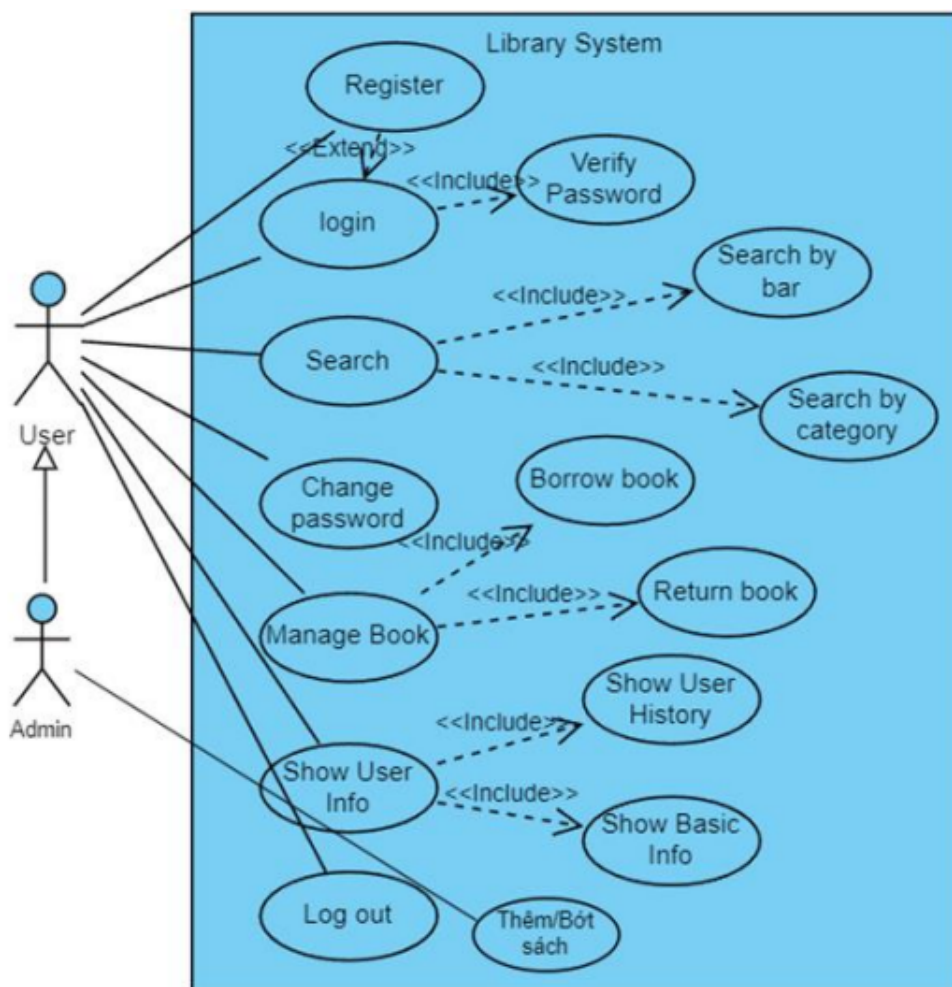
- Người quản lý có thể thêm sách
- Người quản lý có thể thêm sách
- Người quản lý có thể quản lý người đọc bao gồm : thêm, xóa, sửa
- Người quản lý có các chức năng còn lại như người đọc.
- Người quản lý có thể xem được thống kê sách nào mượn nhiều nhất.

3.2.2 Yêu cầu phi chức năng

- Hệ thống hoạt động 24/7
- Hệ thống đưa ra phản hồi nhanh , dưới 1s cho thao tác xem sách và dưới 2s với các thao tác khác.
- Bảo mật thông tin khách hàng tốt theo tiêu chuẩn OWASP
- Giao diện dễ dùng thân thiện đảm bảo có thể để người dùng tự khám phá hết chức năng trong 5 phút.

3.3 Phân tích và đặc tả ca sử dụng

3.3.1 Sơ đồ use case



Hình 3.1: Sơ đồ Use case

3.3.2 Ca sử dụng Đăng ký tài khoản

Mô tả ca sử dụng: Đăng ký tài khoản



Hình 3.2: Biểu đồ hoạt động ca sử dụng đăng kí

Bảng 3.1: Bảng mô tả biểu đồ hoạt động ca sử dụng đăng kí

Tên Use Case	Đăng kí
Mô tả	Use Case này cho phép người dùng đăng ký tài khoản mới
Các tác nhân	Người dùng (cả người đọc và người quản lí)
Điều kiện kích hoạt	Người dùng chọn chức năng đăng ký.
Tiền điều kiện	<ul style="list-style-type: none"> - Người dùng chưa đăng nhập vào ứng dụng. - Người dùng chưa có tài khoản. - Thiết bị kết nối Internet
Hậu điều kiện	<ul style="list-style-type: none"> - Phần mềm sẽ chuyển sang giao diện đăng nhập - Thông tin được lưu vào cơ sở dữ liệu
Luồng cơ bản	<ol style="list-style-type: none"> 1. Người dùng truy cập vào web. 2. Người dùng chọn đăng kí 3. Nhập thông tin người dùng 4. Hệ thống check user đã tồn tại chưa 5. Thành công
Luồng thay thế	Không có.
Yêu cầu	<ul style="list-style-type: none"> - User không được để trống. - Password không được để trống. - Confirm password phải giống password. - Email phải nhập đúng định dạng.

Đăng ký

Họ

Nguyễn Văn

Tên

A

SDT

0987654321

Địa chỉ

HN, TP.HCM,...

Email

abc@gmail.com

Mật khẩu

.....

Xác nhận mật khẩu

Nhập lại mật khẩu

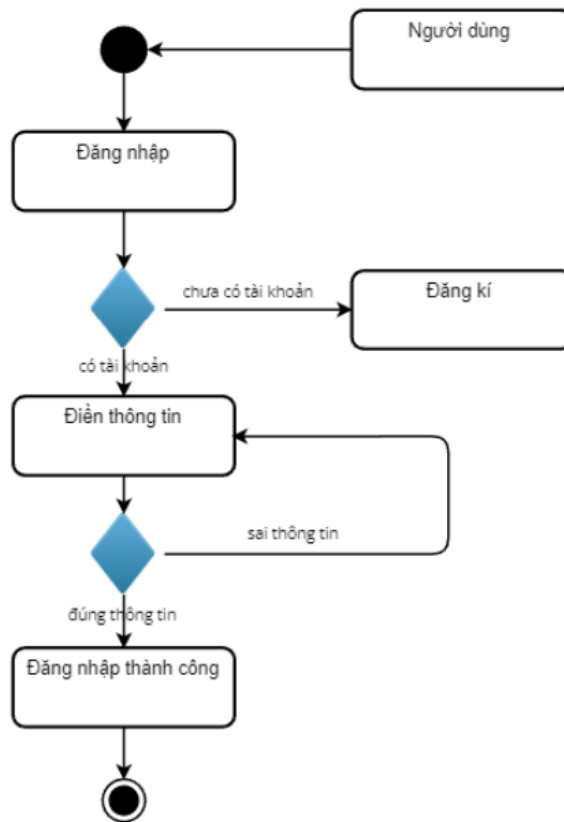
Đăng ký

Bạn đã có tài khoản? [Đăng nhập.](#)

Hình 3.3: Giao diện đăng ký tài khoản

3.3.3 Ca sử dụng Đăng nhập tài khoản

Mô tả ca sử dụng: Đăng nhập tài khoản



Hình 3.4: Biểu đồ hoạt động ca sử dụng đăng đăng nhập

Bảng 3.2: Bảng mô tả biểu đồ hoạt động ca sử dụng đăng nhập

Tên Use Case	Đăng nhập
Mô tả	Use Case này cho phép người dùng đăng nhập tài khoản mới
Các tác nhân	Người dùng (cả người đọc và người quản lí)
Điều kiện kích hoạt	Người dùng vào hệ thống
Tiền điều kiện	<ul style="list-style-type: none"> - Người dùng chưa đăng nhập hệ thống - Có kết nối với server
Hậu điều kiện	Vào giao diện người dùng
Luồng cơ bản	<ol style="list-style-type: none"> 1. Người dùng truy cập vào web. 2. Người dùng chọn đăng nhập 3. Nhập thông tin người dùng 4. Hệ thống check user có hợp lệ hay không 5. Chuyển sang giao diện người dùng
Luồng thay thế	Người dùng chọn đăng kí rồi quay lại đăng nhập sau
Yêu cầu	<ul style="list-style-type: none"> - User không được để trống. - Password không được để trống.

Email:

abc@gmail.com

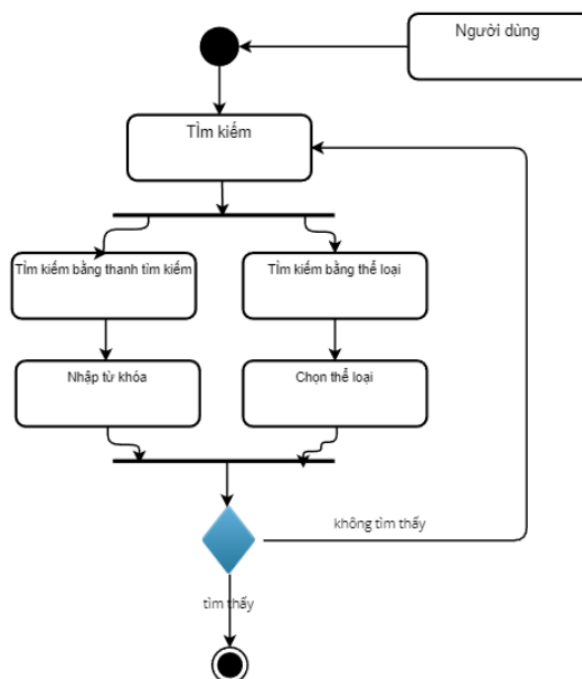
Mật khẩu:

Đăng nhập

Bạn chưa có tài khoản? [Đăng ký.](#)

Hình 3.5: Giao diện đăng nhập tài khoản

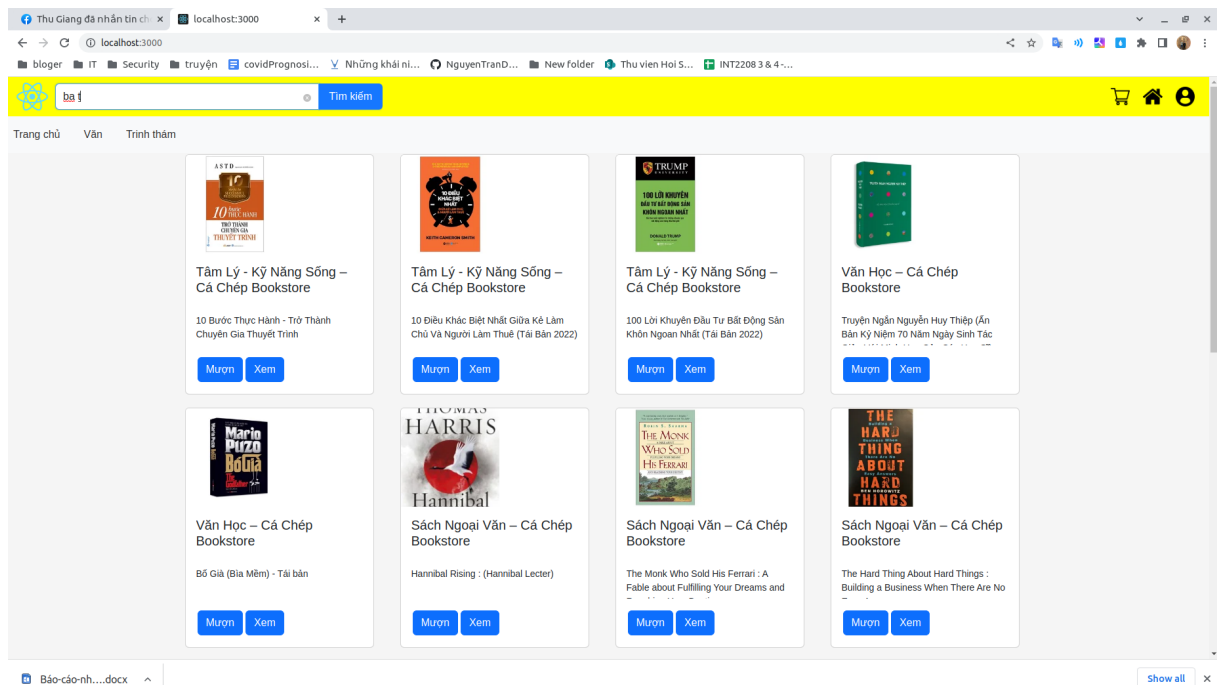
3.3.4 Ca sử dụng Tra cứu sách



Hình 3.6: Biểu đồ hoạt động ca sử dụng tìm kiếm sách

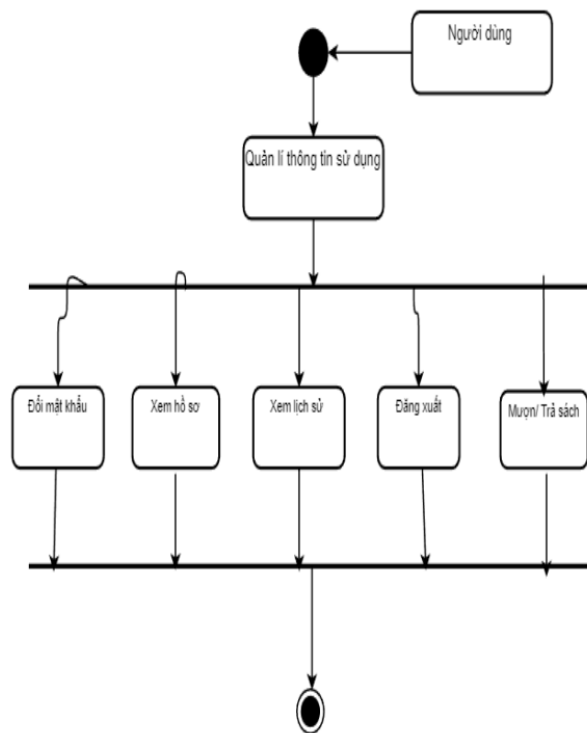
Bảng 3.3: Bảng mô tả biểu đồ hoạt động ca sử dụng tìm kiếm sách

Tên Use Case	Tìm kiếm sách
Mô tả	Use case cho phép người dùng tìm kiếm sách
Các tác nhân	Người dùng
Điều kiện kích hoạt	Người dùng sử dụng thanh tìm kiếm hoặc phân loại sách
Tiền điều kiện	<ul style="list-style-type: none"> - Đã đăng nhập hệ thống - Có kết nối với server
Hậu điều kiện	Hiện ra các sách muốn tìm
Luồng cơ bản	<p>Luồng 1:</p> <ol style="list-style-type: none"> 1. Người dùng gõ từ khóa tìm kiếm vào thanh tìm kiếm 2. Người dùng nhận dữ liệu ra phù hợp <p>Luồng 2:</p> <ol style="list-style-type: none"> 1. Người dùng chọn thể loại văn học hoặc trình thám 2. Người dùng nhận được thông tin phù hợp yêu cầu
Luồng thay thế	Người dùng quay lại tìm kiếm tới khi thỏa mãn nhu cầu
Yêu cầu	Từ khóa không có kí tự đặc biệt



Hình 3.7: Giao diện tìm kiếm sách

3.3.5 Ca sử dụng tra cứu hồ sơ, thông tin người dùng



Hình 3.8: Biểu đồ hoạt động ca sử dụng tra cứu hồ sơ, thông tin người dùng

Bảng 3.4: Bảng mô tả biểu đồ hoạt động ca sử dụng tra cứu hồ sơ, thông tin người dùng

Tên Use Case	Tra cứu hồ sơ, thông tin người dùng
Mô tả	Use case cho phép người dùng tra cứu quản lý thông tin người dùng và thông tin sử dụng phần mềm
Các tác nhân	Người dùng
Điều kiện kích hoạt	Người dùng sử dụng chức năng quản lý thông tin
Tiền điều kiện	- Đã đăng nhập hệ thống - Có kết nối với server
Hậu điều kiện	Hiện ra thông tin người dùng
Luồng cơ bản	Người dùng ấn vào trang quản lý thông tin Người dùng chọn 1 trong các mục: đổi mật khẩu, xem hồ sơ, đăng xuất, sách đang mượn, sách đã trả, xem lịch sử Người dùng sử dụng chức năng rồi kết thúc
Luồng thay thế	Không có
Yêu cầu	Người dùng đã đăng nhập thành công. Thỏa mãn yêu cầu tùy theo chức năng lựa chọn

Quản lý thư viện

Hồ sơ của tôi

Thông tin cá nhân

Đổi mật khẩu

Đăng xuất

Lịch sử

Sách đang mượn

Sách đã trả

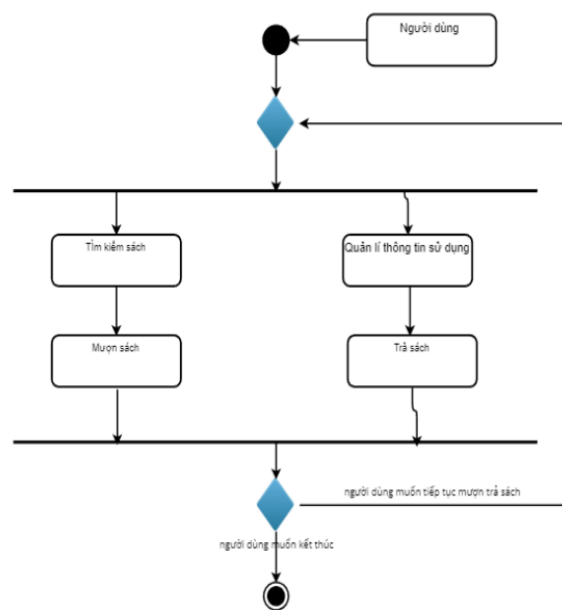
Nguyen VanA
18

Số điện thoại:
0912345678

Địa chỉ:
HN

Hình 3.9: Giao diện tra cứu hồ sơ, thông tin người dùng

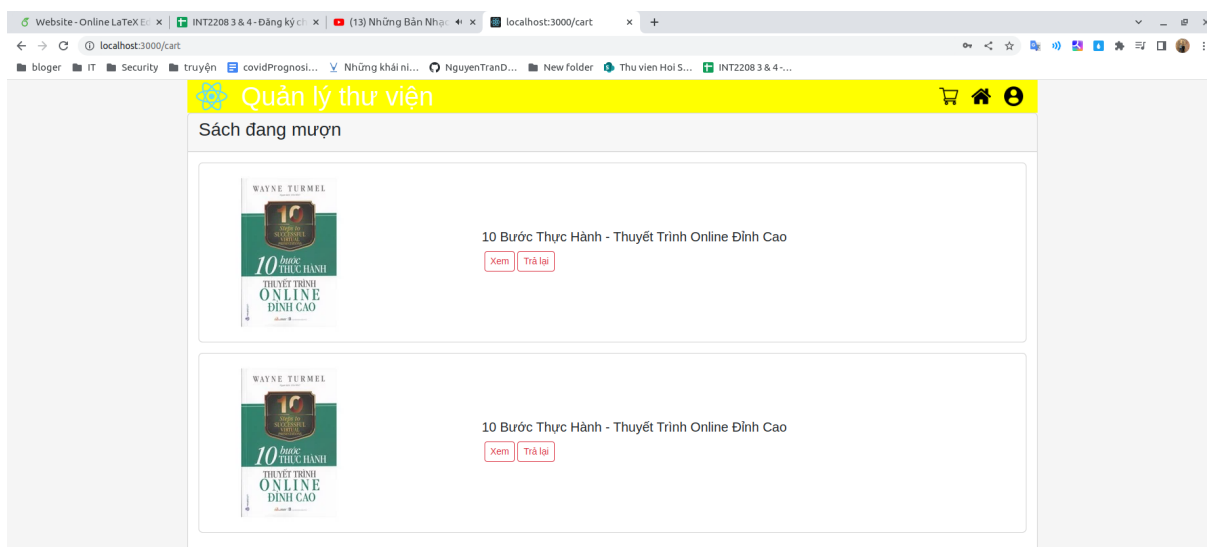
3.3.6 Ca sử dụng Mượn/trả sách



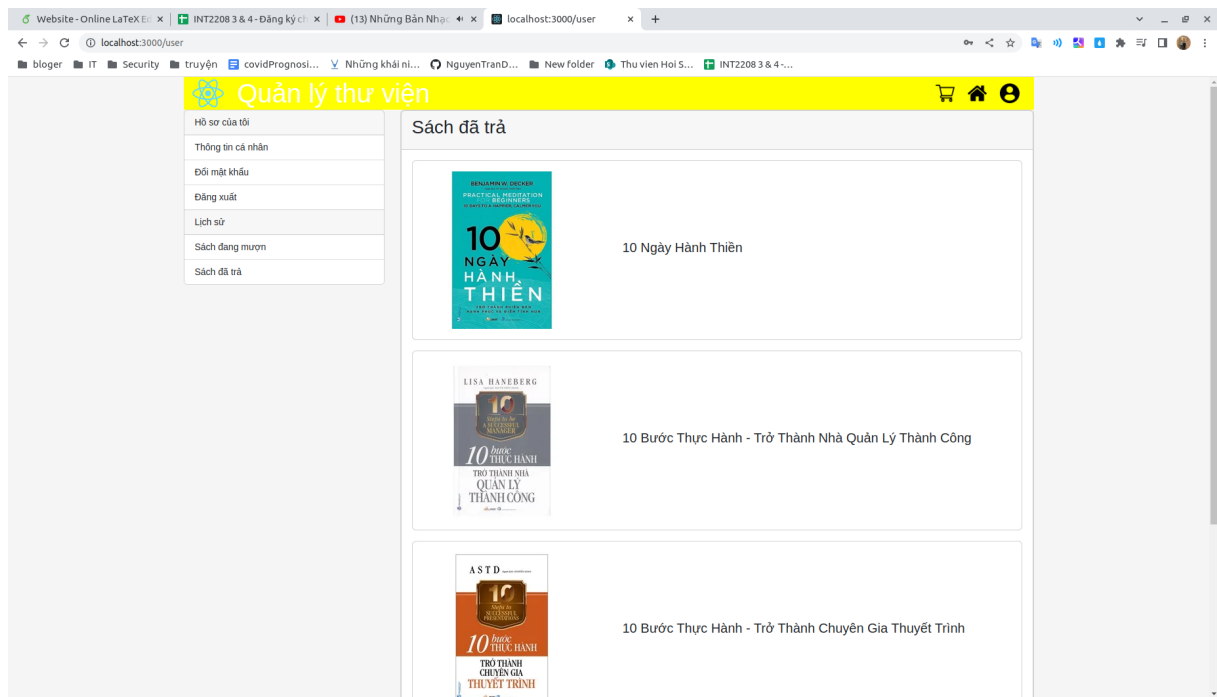
Hình 3.10: Biểu đồ hoạt động ca sử dụng mượn/trả sách

Bảng 3.5: Bảng mô tả biểu đồ hoạt động ca sử dụng mượn/trả sách

Tên Use Case	Mượn/Trả sách
Mô tả	Use case cho phép người dùng mượn và trả sách
Các tác nhân	Người dùng
Điều kiện kích hoạt	Người dùng sử dụng chức năng quản lý thông tin(để trả sách) và sử dụng chức năng tìm kiếm sách(để mượn sách)
Tiền điều kiện	- Đã đăng nhập hệ thống - Có kết nối với server
Hậu điều kiện	Thêm/Bớt sách vào danh sách mượn, cập nhật lịch sử và thông tin sử dụng
Luồng cơ bản	<p>Luồng 1: Mượn sách</p> <ol style="list-style-type: none"> 1. Người dùng tìm kiếm sách 2. Người dùng nhấn mượn sách <p>Luồng 2: Trả sách</p> <ol style="list-style-type: none"> 1. Người dùng vào quản lý thông tin sử dụng 2. Người dùng nhấn trả từng cuốn hoặc trả hết sách
Luồng thay thế	Người dùng quay lại mượn trả sách tới khi thỏa mãn nhu cầu
Yêu cầu	Không có

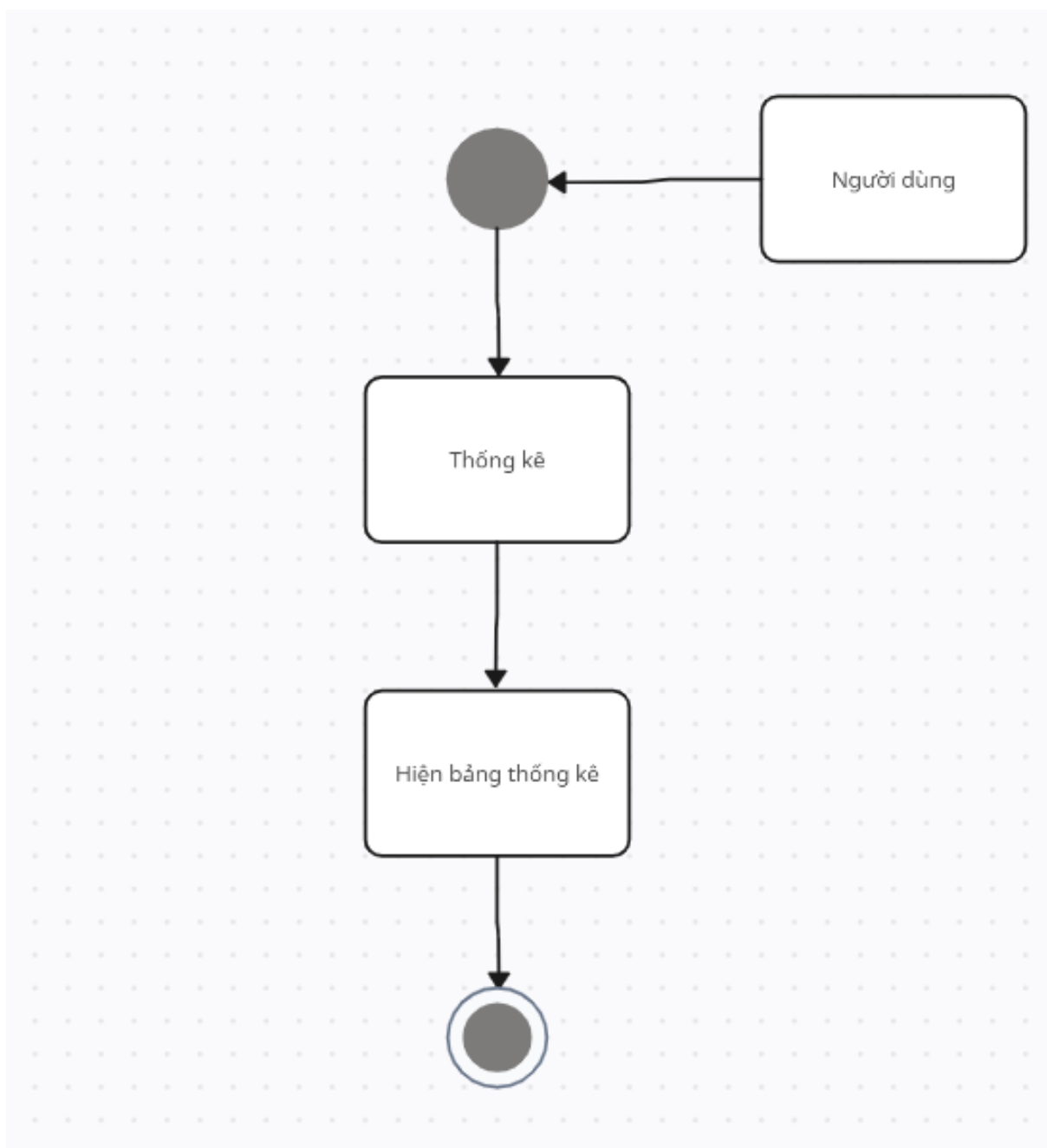


Hình 3.11: Giao diện thông tin những cuốn sách đã mượn.



Hình 3.12: Giao diện thông tin những cuốn sách đã trả.

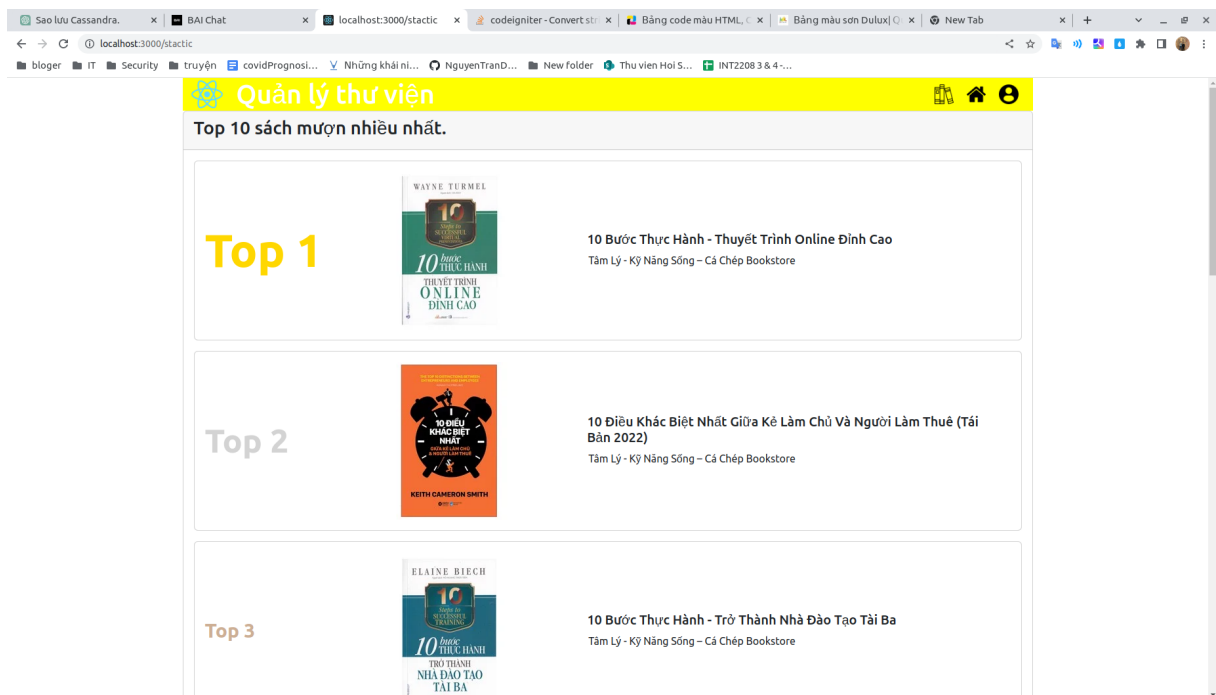
3.3.7 Hiển thị thống kê những cuốn sách mượn nhiều nhất



Hình 3.13: Biểu đồ hoạt động ca sử dụng thống kê.

Bảng 3.6: Bảng mô tả biểu đồ hoạt động ca sử dụng thống kê.

Tên Use Case	Thống kê những cuốn sách mượn nhiều nhất
Mô tả	Use case cho phép quản trị viên xem thống kê
Các tác nhân	Quản trị viên
Điều kiện kích hoạt	Đăng nhập vào hệ thống với quyền quản trị viên
Tiền điều kiện	Có kết nối với server Đã đăng nhập vào hệ thống
Hậu điều kiện	Hiển thị top 10 cuốn sách mượn nhiều nhất
Luồng cơ bản	Quản trị viên đăng nhập vào hệ thống Quản trị viên nhấn vào thống kê trên thanh navbar
Luồng thay thế	Không có
Yêu cầu	Phải đăng nhập bằng quyền quản trị viên



Hình 3.14: Giao diện thống kê.

Tổng kết chương: Biểu đồ ca hoạt động được sử dụng để mô phỏng các luồng hoạt động và tương tác giữa các đối tượng trong hệ thống. Biểu đồ ca hoạt động giúp hiểu rõ quy trình làm việc, các hoạt động, quyết định, và các luồng điều khiển trong một quy trình hoặc một chức năng cụ thể. Sử dụng biểu đồ ca hoạt động có những lợi ích sau:

- Mô tả quy trình làm việc: Biểu đồ ca hoạt động giúp mô tả các bước và hoạt động trong quy trình làm việc hoặc chức năng. Nó giúp xác định các công việc cần được

thực hiện và cách chúng tương tác với nhau.

- Hiểu quyền kiểm soát và luồng điều khiển: Biểu đồ ca hoạt động cho thấy cách các hoạt động được thực hiện và kiểm soát như thế nào trong hệ thống. Nó có thể hiển thị luồng điều khiển tuần tự, rẽ nhánh, lặp lại và các quyết định trong quy trình.
- Xác định tương tác đối tượng: Biểu đồ ca hoạt động có thể hiển thị các tương tác giữa các đối tượng trong hệ thống. Nó cho phép xác định cách các đối tượng giao tiếp, truyền thông tin và thực hiện các hoạt động chung.
- Phân rõ trách nhiệm và vai trò: Biểu đồ ca hoạt động có thể phân rõ trách nhiệm và vai trò của các đối tượng hoặc các phần tử trong quy trình. Nó giúp xác định người chịu trách.
- Phân tích và thiết kế hệ thống: Biểu đồ ca hoạt động cung cấp một cách để phân tích và thiết kế hệ thống. Nó giúp làm rõ yêu cầu và chức năng của hệ thống, đồng thời xác định các hoạt động và luồng điều khiển cần thiết.

Chương 4

Phân tích thiết kế và xây dựng hệ thống

4.1 Thiết kế kiến trúc chung của hệ thống

4.1.1 Phân tích kiến trúc

Định nghĩa các thành phần trừ tượng hóa của hệ thống:

- Customer: là tài khoản người dùng đăng ký qua hệ thống nhằm cung cấp các chức năng đặc thù như tìm kiếm, mượn sách, tra cứu,

- Book: là một bản ghi các cuốn sách.

- Rental: là một bản ghi lưu thông tin người mượn sách, cuốn sách đã mượn, cuốn sách đã trả, ngày mượn, ngày trả.

4.1.2 Mô hình MICROSERVICE

Mô hình microservice là một kiến trúc phần mềm được sử dụng để phát triển các ứng dụng lớn, phức tạp và dễ mở rộng. Trong mô hình này, ứng dụng được phân chia thành nhiều dịch vụ nhỏ hơn, được gọi là microservices, mỗi dịch vụ chịu trách nhiệm thực hiện một chức năng cụ thể trong hệ thống. Mỗi microservice có thể được phát triển, triển khai và quản lý độc lập với nhau, và có thể được viết bằng các ngôn ngữ và công nghệ khác nhau.

Mô hình microservice giúp phân tách ứng dụng thành các thành phần nhỏ hơn, dễ

quản lý và mở rộng hơn. Bằng cách phân tách ứng dụng thành các dịch vụ độc lập, các nhà phát triển có thể tập trung vào việc phát triển một phần của hệ thống mà không ảnh hưởng đến các phần khác. Mô hình này cũng giúp giảm thiểu rủi ro khi có lỗi xảy ra trong một dịch vụ, vì các dịch vụ khác vẫn có thể tiếp tục hoạt động bình thường.

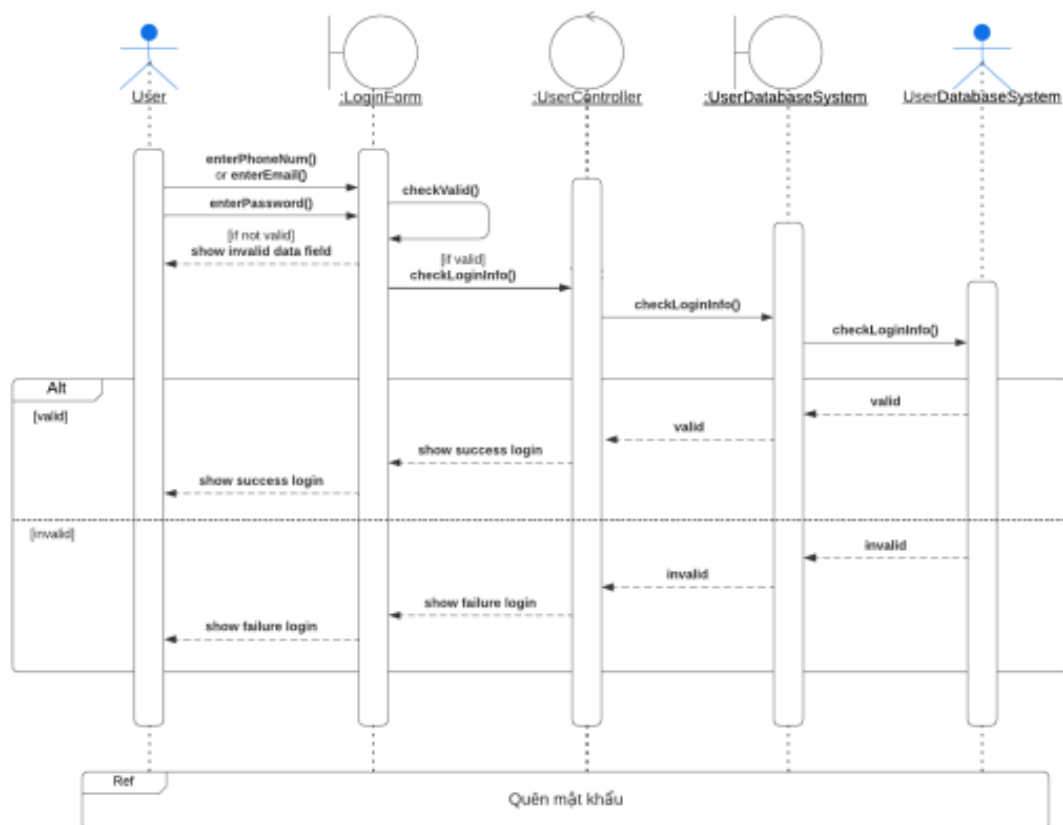
Từ những ưu điểm đó, nhóm chọn sử dụng mô hình microservice để hỗ trợ phát triển và bảo trì sản phẩm một cách tốt nhất có thể. Không những thế, mô hình còn giúp phân chia nhiệm vụ một cách dễ dàng hơn khi mỗi một thành viên sẽ phụ trách một cơ sở dữ liệu nhất định.

- API: cung cấp giao thức để hỗ trợ truy xuất cơ sở dữ liệu dựa trên hoạt động của khách hàng.

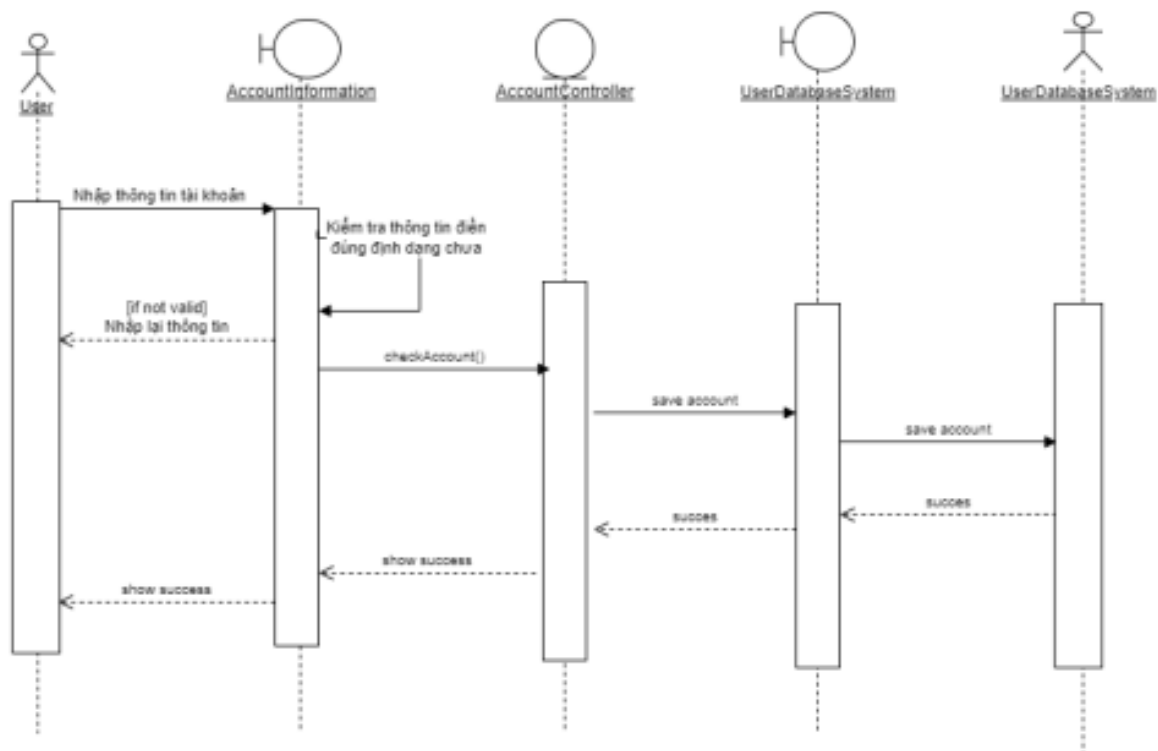
- View: hiển thị thông tin, tương tác trực tiếp với người dùng, chuyển tiếp các yêu cầu tới hệ thống để hiển thị ra màn hình.

- Controller: có nhiệm vụ nhận yêu cầu của người dùng và gọi các phương thức phù hợp để xử lý và thao tác với cơ sở dữ liệu.

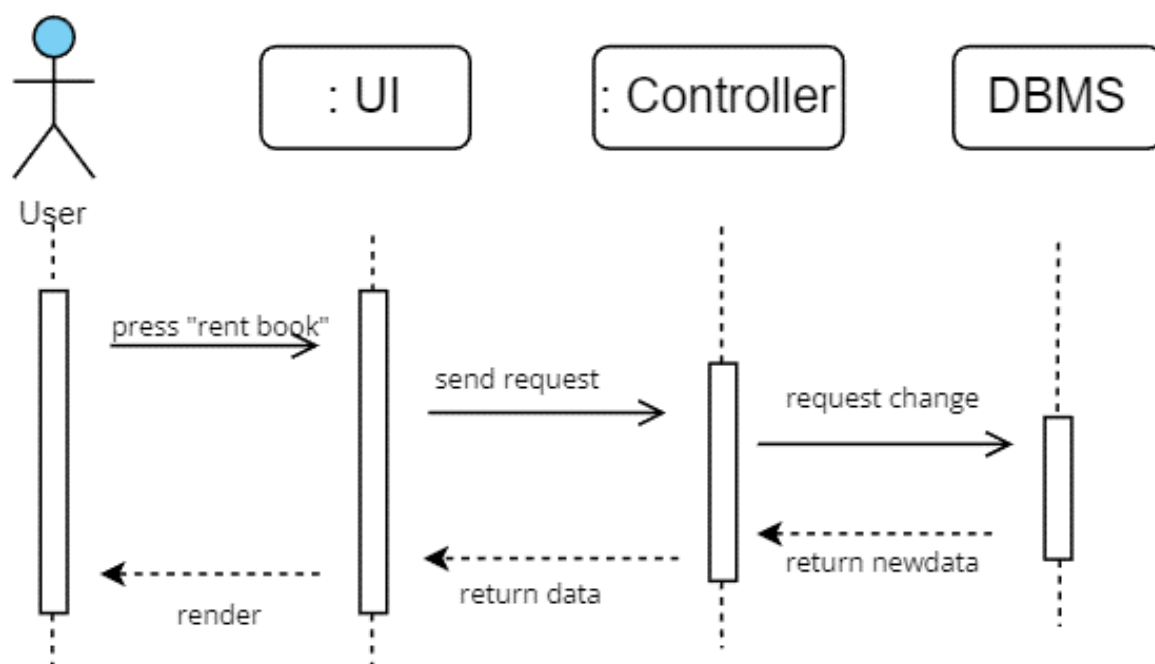
4.2 Biểu đồ tuần tự cho ca sử dụng chính



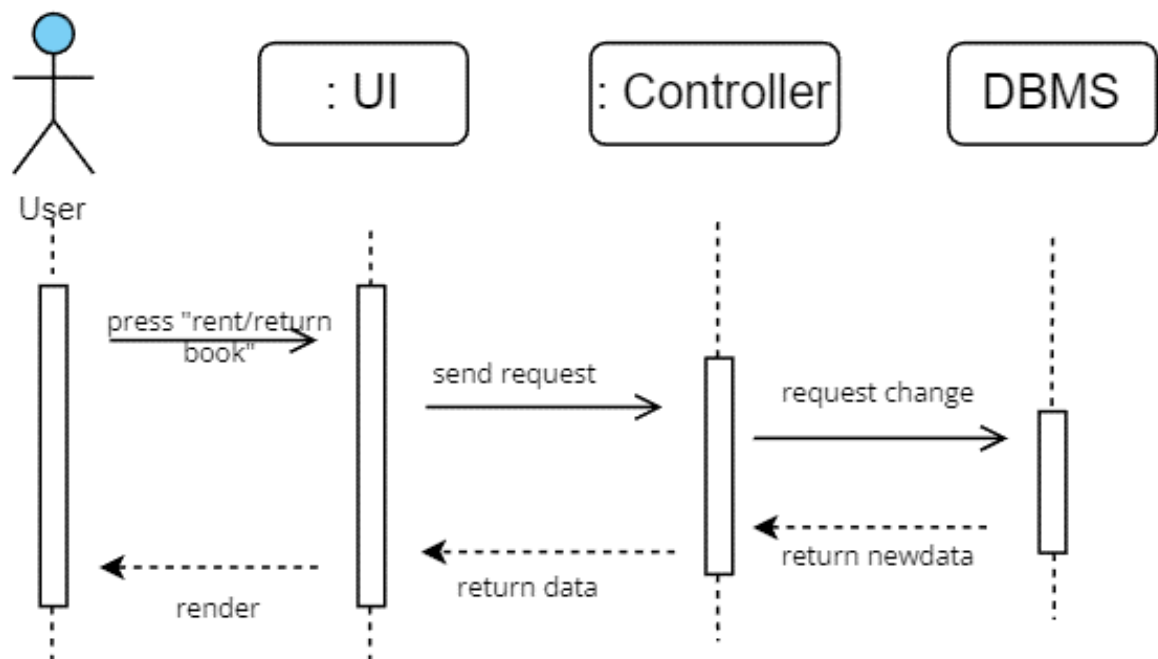
Hình 4.1: Biểu đồ tuần tự cho ca sử dụng đăng nhập



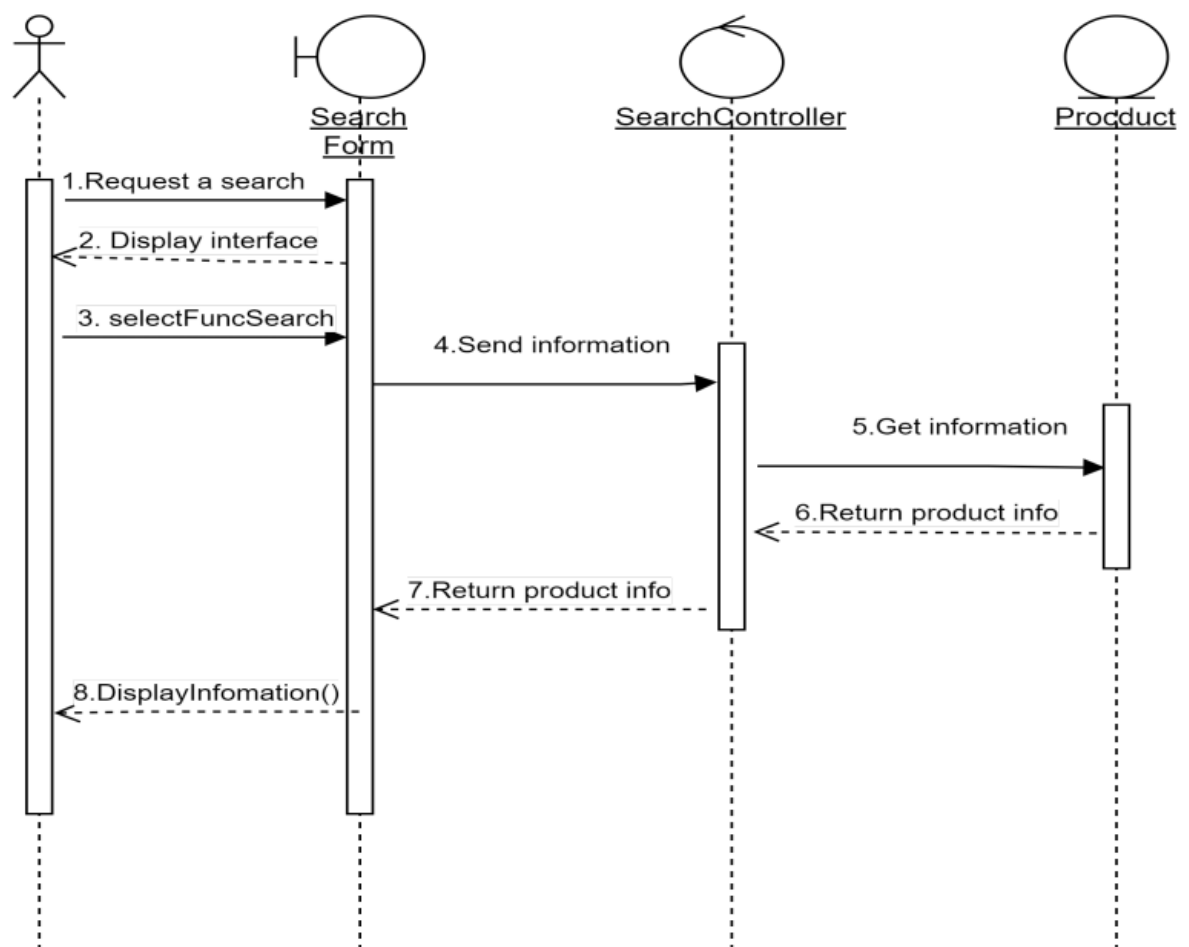
Hình 4.2: Biểu đồ tuần tự cho ca sử dụng đăng ký.



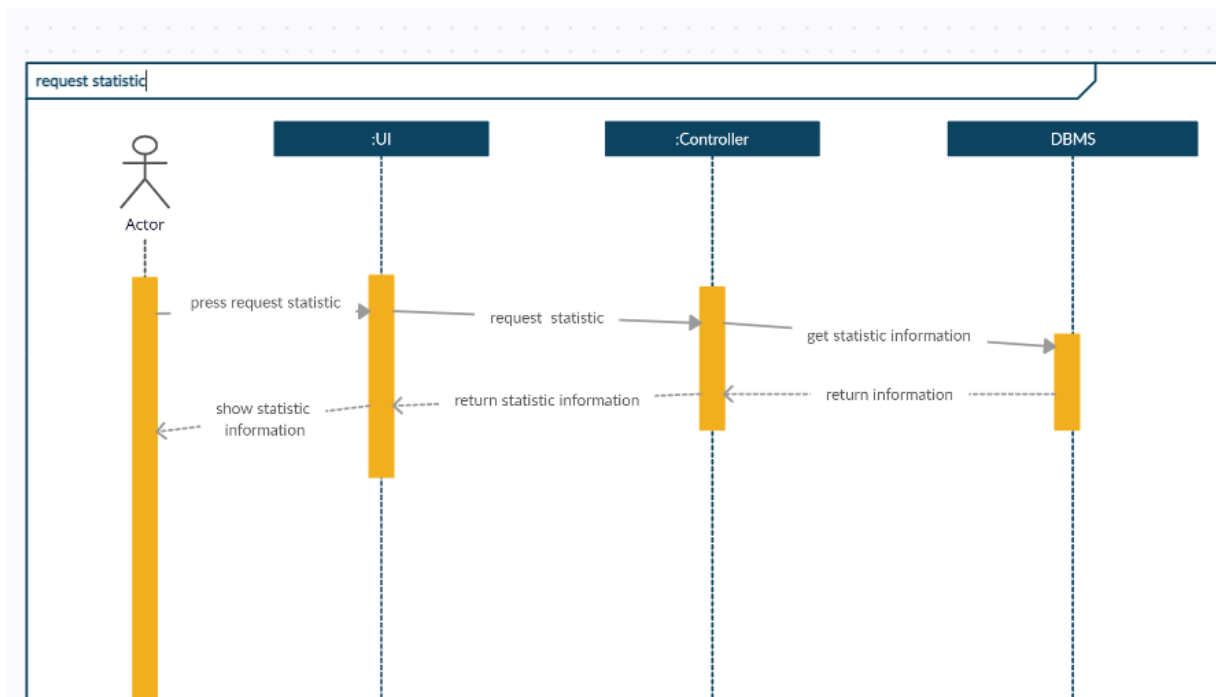
Hình 4.3: Biểu đồ tuần tự cho ca sử dụng mượn sách.



Hình 4.4: Biểu đồ tuần tự cho ca sử dụng trả sách.



Hình 4.5: Biểu đồ tuần tự cho ca sử dụng tra cứu sách.



Hình 4.6: Biểu đồ tuần tự cho ca sử dụng thống kê.

4.3 Thiết kế API

Người dùng sẽ yêu cầu hệ thống hiển thị ra màn hình thông qua các thao tác trên website và sẽ chỉ gửi được yêu cầu về. Hệ thống sẽ thông qua APT để nhận những yêu cầu của người dùng và thực hiện truy vấn trên cơ sở dữ liệu và hiển thị cho người dùng.

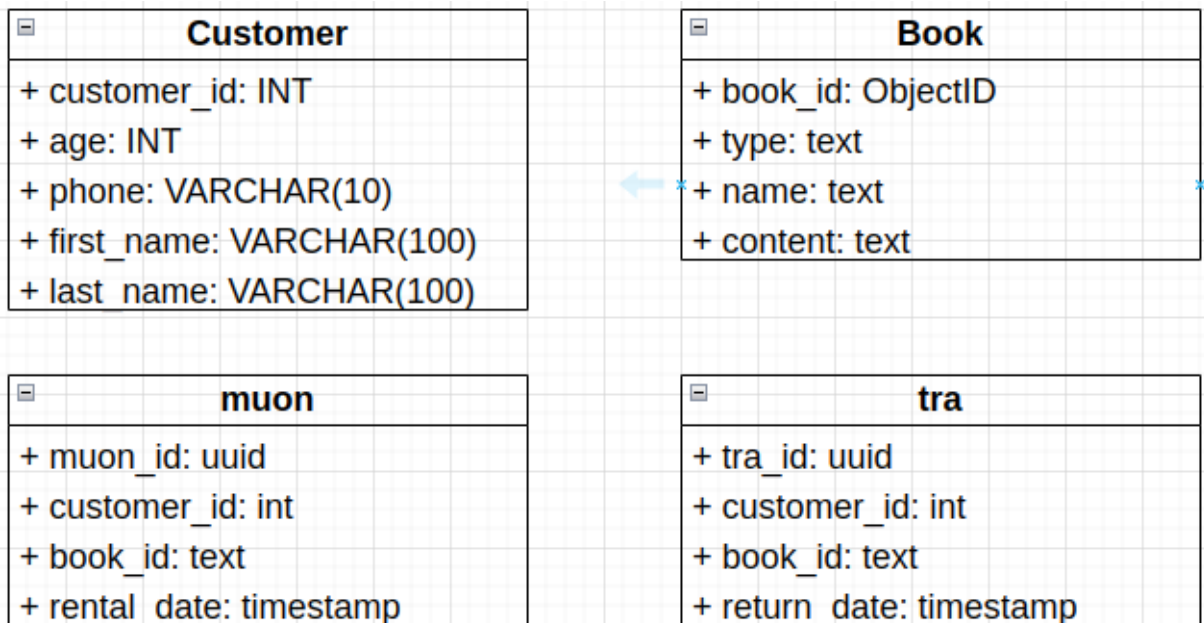
Yêu cầu: Không được hiển thị các thông tin quan trọng như mật khẩu, ... cho người ngoài. Người dùng chỉ có thể nhập mật khẩu dưới dạng * và gửi mật khẩu đến hệ thống. Chức năng thêm sản phẩm chỉ có thủ thư hoặc quản lý thư viện mới được thêm.

Mục đích: Đảm bảo bảo mật cho hệ thống, tránh truy cập vào cơ sở dữ liệu. Giúp tránh mất mát thông tin, và không cho phép truy cập trái phép.

4.4 Thiết kế cơ sở dữ liệu

Nhóm em chia nhỏ các cơ sở dữ liệu và dựa vào ưu điểm của các hệ quản trị cơ sở dữ liệu để lưu trữ và truy xuất.

- MySQL: có tính bảo mật tốt, bao gồm cơ chế xác thực và phân quyền cấp độ người dùng, giúp bảo vệ dữ liệu khỏi các cuộc tấn công độc hại. MySQL sẽ lưu trữ thông tin của các user (người đọc, admin) bao gồm : email, mật khẩu, tên tuổi, số điện thoại, địa chỉ, phân quyền. (Hình 2.1)



Hình 4.7: Hình ảnh cơ sở dữ liệu

- MongoDB: cho phép lưu trữ và truy xuất dữ liệu dễ dàng hơn so với các hệ quản trị cơ sở dữ liệu quan hệ truyền thống và có khả năng mở rộng dễ dàng. Từ những ưu điểm trên, nhóm em đã quyết định sử dụng MongoDB để lưu thông tin về sách. Trong MongoDB, nhóm em sử dụng một số kiểu dữ liệu như: ObjectID, text. Nhóm em cũng hỗ trợ tìm kiếm sách dựa trên tên và thể loại của sách. Ngoài ra, để hỗ trợ tìm kiếm, nhóm em cũng sử dụng tìm kiếm trên id để giúp tìm kiếm sách một cách nhanh chóng và hiệu quả. Không những thế, nhóm cũng đang phát triển tìm kiếm dựa trên từ gợi ý để hỗ trợ truy xuất được nhanh chóng hơn. (Hình 2.2)
- Cassandra: cho phép mở rộng một cách linh hoạt và dễ dàng để xử lý các tải trọng dữ liệu lớn. Vì Cassandra hỗ trợ nhiều tính năng hữu ích như khả năng sao lưu và khôi phục dữ liệu tự động, khả năng sao chép dữ liệu qua các máy chủ và khả năng đồng bộ dữ liệu giữa các trung tâm dữ liệu khác nhau nên rất phù hợp để lưu thông tin lịch sử mượn/trả sách (đây là thông tin cần đc truy xuất nhiều). Trong Cassandra, nhóm em sử dụng một số kiểu dữ liệu như: uuid, text, int, timestamp để lưu trữ. Nhóm em đang cố gắng phát triển để thông kê số lần mượn của một cuốn sách và tự động tính tiền khi khách hàng mượn quá số ngày quy định. (Hình 2.3)

4.5 Thiết kế giao diện người dùng

Để thiết kế giao diện website quản lý thư viện, nhóm đã tập trung vào trải nghiệm người dùng và tính thẩm mỹ của giao diện. Giao diện của nhóm em hướng đến sự đơn giản, dễ sử dụng và trực quan để giúp người dùng tìm kiếm, mượn và quản lý sách một cách thuận tiện nhất.

Trong giao diện chính, nhóm em sẽ chia thành các phần chính như: thanh header, thanh điều hướng, phần tìm kiếm sách, phần hiển thị thông tin sách, phần thông tin người dùng, phần hiển thị những quyển sách đã mượn và thông tin của thư viện.

Thanh header giúp người dùng dễ dàng truy cập thông qua hình ảnh trực quan. Trong thanh header có thanh tìm kiếm giúp người dùng tìm kiếm sách theo yêu cầu. Và bên phải thanh header là những hình ảnh trực quan về một số nút để người dùng thao tác.

Thanh điều hướng sẽ giúp người dùng dễ dàng truy cập đến các chức năng chính như tìm kiếm sách. Phần tìm kiếm sách sẽ cho phép người dùng tìm kiếm sách theo tên và thể loại. Phần hiển thị thông tin sách sẽ hiển thị thông tin chi tiết về sách như tên sách, thể loại, hình ảnh của sách. Phần thông tin người dùng sẽ cho phép người dùng xem thông tin cá nhân, đổi mật khẩu và xem được những sách đã mượn, đang mượn.

Để tạo sự thuận tiện và tối ưu cho trải nghiệm người dùng, ta cần đảm bảo giao diện thân thiện, có độ phản hồi nhanh và sử dụng hình ảnh, biểu tượng và màu sắc hợp lý để tăng tính thẩm mỹ.

Về cách bố trí, ta sẽ chia cột trái và cột phải để hiển thị thông tin sách và thông tin người dùng. Phần tìm kiếm sách sẽ được đặt ở vị trí trung tâm, nổi bật và dễ dàng để sử dụng.

Tổng thể, thiết kế giao diện website quản lý thư viện cần phải đáp ứng được các yêu cầu về tính năng, trải nghiệm người dùng và tính thẩm mỹ để mang lại sự thuận tiện và thoải mái cho người dùng.

Tổng kết chương: Biểu đồ tuần tự là một loại biểu đồ trong kỹ thuật phần mềm, dùng để mô tả các tương tác giữa các đối tượng trong một hệ thống. Biểu đồ ca tuần tự giúp chúng em:

- Mô tả tương tác giữa các đối tượng: Biểu đồ tuần tự cho phép mô tả chi tiết các thông điệp và tương tác xảy ra giữa các đối tượng trong hệ thống. Nó cho thấy sự truyền thông tin, gọi hàm và tương tác giữa các đối tượng trong quá trình thực hiện

một hoạt động cụ thể.

- **Hiểu luồng thời gian và sự tương tác:** Biểu đồ tuần tự cho thấy trình tự thực hiện và tương tác giữa các đối tượng theo thứ tự thời gian. Nó giúp hiểu rõ luồng thực hiện của các hoạt động và sự tương tác giữa các đối tượng trong quá trình chạy.
- **Phân rõ trách nhiệm và vai trò của các đối tượng:** Biểu đồ tuần tự cho phép xác định rõ trách nhiệm và vai trò của từng đối tượng trong quá trình tương tác. Nó giúp hiểu rõ những đối tượng nào tham gia và đóng góp vào quá trình thực hiện một hoạt động.
- **Giao tiếp và truyền đạt thông tin:** Biểu đồ tuần tự là một công cụ trực quan để giao tiếp và truyền đạt thông tin về các tương tác và luồng thực hiện giữa các đối tượng trong hệ thống. Nó giúp các thành viên trong nhóm phát triển và các bên liên quan hiểu rõ quy trình và tương tác trong hệ thống.

Chương 5

Cài đặt và kiểm thử

5.1 Kiểm thử hệ thống

Tổng kết chương: Kiểm thử là một phần quan trọng trong quy trình phát triển phần mềm. Nó giúp đảm bảo chất lượng và độ tin cậy của phần mềm thông qua việc tìm lỗi, xác nhận tính đúng đắn và kiểm tra các yêu cầu và chức năng của hệ thống. Qua kiểm thử chúng em có thể xác định được:

- **Phát hiện lỗi và khuyết điểm:** Kiểm thử giúp xác định các lỗi, khuyết điểm hoặc hành vi không mong muốn trong phần mềm. Việc phát hiện lỗi sớm giúp giảm thiểu rủi ro và chi phí sửa chữa sau này.
- **Đảm bảo chức năng và yêu cầu:** Kiểm thử đảm bảo rằng phần mềm hoạt động đúng theo yêu cầu và chức năng đã được xác định. Nó giúp đảm bảo rằng hệ thống phục vụ được mục đích của nó và cung cấp kết quả chính xác.
- **Đánh giá hiệu suất và tải:** Kiểm thử hiệu năng và tải giúp đảm bảo rằng phần mềm hoạt động ổn định và có khả năng xử lý đúng lượng công việc được đề ra. Nó đảm bảo rằng hệ thống có thể xử lý tải cao và đáp ứng các yêu cầu về hiệu suất.

Bảng 5.1: Kiểm thử đăng ký

STT	Yêu cầu	Kiểm tra	Kết quả mong đợi	Kết quả thực tế	Ngày kiểm thử	Người kiểm thử
1	Người dùng có thể nhập thông tin đăng ký bao gồm mật khẩu, email, số điện thoại	Nhập đầy đủ thông tin đăng kí và ấn đăng kí	Quay về màn hình đăng nhập	Thành công	10/05/2023	Tuấn Anh
2	Người dùng không thể đăng ký nếu chưa nhập đủ thông tin (email và mật khẩu)	Không nhập đầy đủ thông tin và bấm nút đăng ký	Không thể đăng ký tài khoản	Thành công	10/05/2023	Tuấn Anh
3	Mật khẩu phải được ẩn khi người dùng nhập	Nhập mật khẩu	Mật khẩu được ẩn bằng dấu chấm	Thành công	10/05/2023	Tuấn Anh
4	Người dùng phải xác nhận lại mật khẩu để đăng ký	Nhập mật khẩu và nhập lại mật khẩu xác nhận	Nếu nhập sai mật khẩu xác nhận sẽ không đăng ký thành công	Thành công	10/05/2023	Tuấn Anh
5	Email phải đúng định dạng	Nhập email	Nếu nhập sai định dạng email sẽ báo lỗi	Thành công	14/05/2023	Tuấn Anh
	Người dùng không được nhập email đã tồn tại trong hệ thống	Nhập email	Nếu email đó đã tồn tại phải có thông báo.	Thành công	14/05/2023	Tuấn Anh

Bảng 5.2: Kiểm thử đăng nhập

STT	Yêu cầu	Kiểm tra	Kết quả mong đợi	Kết quả thực tế	Ngày kiểm thử	Người kiểm thử
1	Người dùng phải nhập tài khoản và mật khẩu	Nhập đúng tài khoản và mật khẩu đã đăng kí	Vào trang chủ	Thành công	10/05/2023	Tuấn Anh
2	Người dùng phải nhập tài khoản và mật khẩu	Nhập sai mật khẩu	Hiện thông báo “Sai tên đăng nhập hoặc mật khẩu”	Thành công	10/05/2023	Tuấn Anh
3	Người dùng phải nhập tài khoản và mật khẩu	Nhập tên đăng nhập chưa được đăng ký	Hiện thông báo “Tài khoản này chưa được đăng kí”	Thất bại (Hiện thông báo “Sai tên đăng nhập hoặc mật khẩu”)	10/05/2023	Tuấn Anh
4	Người dùng không nhập mật khẩu	Người dùng chỉ nhập tài khoản và ấn đăng nhập	Hiện thông báo “Vui long nhập mật khẩu”	Thất bại (Hiện thông báo “Sai tên đăng nhập hoặc mật khẩu”)	10/05/2023	Tuấn Anh
5	Người dùng không nhập gì	Người dùng không nhập gì và ấn đăng kí	Hiện thông báo “Vui long nhập tài khoản và mật khẩu”	Thất bại (Hiện thông báo “Sai tên đăng nhập hoặc mật khẩu”)	10/05/2023	Tuấn Anh
6	Người dùng không nhập mật khẩu	Người dùng chỉ nhập tài khoản và ấn đăng nhập	Hiện thông báo “Vui long nhập mật khẩu”	Thành công	14/05/2023	Tuấn Anh
7	Người dùng không nhập gì	Người dùng không nhập gì và ấn đăng kí	Hiện thông báo “Vui long nhập tài khoản và mật khẩu”	Thành công	14/05/2023	Tuấn Anh

Bảng 5.3: Kiểm thử mượn sách

STT	Yêu cầu	Kiểm tra	Kết quả mong đợi	Kết quả thực tế	Ngày kiểm thử	Người kiểm thử
1	Mượn sách	Ấn nút “mượn” sách và ấn nút “OK” để xác nhận	Sách được thêm vào danh sách mượn	Thành công	10/05/2023	Tuấn Anh
2	Mượn sách	Ấn nút “mượn” sách nhưng không ấn nút “OK” mà ấn “Cancel để hủy	Sách không thêm vào danh sách mượn	Thành công	10/05/2023	Tuấn Anh
3	Mượn sách	Mượn sách nếu sách vẫn còn ở thư viện	Nếu không còn sách thì sẽ không thể mượn	Thất bại (Khi ấn nút “mượn” sách sẽ không có thông báo sách có đang còn không)	10/05/2023	Tuấn Anh
4	Mượn sách	Người dùng chỉ nhập tài khoản và ấn đăng nhập	Chỉ được thêm 1 sách mỗi loại vào danh sách mượn	Thất bại (Vẫn mượn được nhiều sách cùng 1 loại)	10/05/2023	Tuấn Anh
5	Xem qua thông tin sách	Ấn vào nút “xem” để xem qua tên sách và tác giả	Hiện ra thông tin	Thành công	10/05/2023	Tuấn Anh

Bảng 5.4: Kiểm thử trả sách

STT	Yêu cầu	Kiểm tra	Kết quả mong đợi	Kết quả thực tế	Ngày kiểm thử	Người kiểm thử
1	Trả sách	Ấn nút “trả sách”, sau đó xác thực nhấn nút “OK” và reload lại trang sẽ trả lại sách	Sách không còn trong danh sách mượn	Thành công	10/05/2023	Tuấn Anh
2	Hủy trả sách	Ấn nút “trả sách”, sau đó nhấn nút “Cancel” và reload lại trang sẽ trả lại sách	Sách vẫn còn trong danh sách mượn	Thành công	10/05/2023	Tuấn Anh
3	Xem sách	Ấn vào nút "Xem" để đọc sách	Hiện ra trang sách và đọc	Thành công (Nhưng đây mới chỉ là tính năng demo)	10/05/2023	Tuấn Anh
4	Quay trở về	Ấn biểu tượng "Trở lại" để quay lại danh sách mượn	Quay lại trang danh sách mượn	Thất bại (Chưa có button quay trở lại)	10/05/2023	Tuấn Anh
5	Trả sách toàn bộ	Ấn nút "Trả tất cả" để trả lại toàn bộ sách	Trả lại toàn bộ sách	Thất bại (Tính năng chưa có)	10/05/2023	Tuấn Anh

Bảng 5.5: Kiểm thử tra cứu sách

STT	Yêu cầu	Kiểm tra	Kết quả mong đợi	Kết quả thực tế	Ngày kiểm thử	Người kiểm thử
1	Tra cứu đúng tên	Nhập đúng 100% tên cuốn sách	Hiện ra những đầu sách khớp với tên	Thành công	10/05/2023	Tuấn Anh
2	Tra cứu đúng 1 hoặc nhiều từ đứng cạnh nhau có trong tên sách (trừ trường hợp đúng hết)	Nhập các từ gợi ý sẽ hiện ra sách	Hiện ra những đầu sách khớp với các từ gợi ý	Thành công	10/05/2023	Tuấn Anh
3	Tra cứu đúng 1 hoặc nhiều từ không đứng cạnh nhau có trong tên sách	Nhập các từ gợi ý sẽ hiện ra sách	Hiện ra những đầu sách khớp với các từ gợi ý	Thất bại (Không hiển thị đầu sách cần tìm)	10/05/2023	Tuấn Anh
4	Ấn biểu tượng xóa để xóa các từ trong thanh tìm kiếm	Làm trống thanh tìm kiếm	Thanh tìm kiếm không còn kí tự nào	Thành công	10/05/2023	Tuấn Anh
5	Tra cứu đúng tiêu đề	Nhập đúng 100% tiêu đề cuốn sách	Hiện ra những đầu sách khớp tiêu đề	Thành công	10/05/2023	Tuấn Anh
6	Tra cứu đúng 1 hoặc nhiều từ đứng cạnh nhau có trong tiêu đề sách (trừ trường hợp đúng hết)	Nhập các từ gợi ý sẽ hiện ra sách	Hiện ra những đầu sách khớp với các từ gợi ý	Thành công	10/05/2023	Tuấn Anh
7	Tra cứu đúng 1 hoặc nhiều từ không đứng cạnh nhau có trong tên sách	Nhập các từ gợi ý sẽ hiện ra sách	Hiện ra những đầu sách khớp với các từ gợi ý	Thành công	14/05/2023	Tuấn Anh

Bảng 5.6: Kiểm thử thanh Navbar

STT	Yêu cầu	Kiểm tra	Kết quả mong đợi	Kết quả thực tế	Ngày kiểm thử	Người kiểm thử
1	Lọc loại sách theo các nút trên thanh lọc	Ấn vào “Văn” sẽ chỉ hiện ra những sách thể loại văn học	Chỉ hiện ra những sách thể loại văn học	Thành công	10/05/2023	Tuấn Anh
2	Lọc loại sách theo các nút trên thanh lọc	Ấn vào “Trình thám” sẽ chỉ hiện ra những sách thể loại văn học	Chỉ hiện ra những sách thể loại trình thám	Thành công	10/05/2023	Tuấn Anh
3	Trở về trang chủ	Nhập các từ gợi ý sẽ hiện ra sách	Hiện ra trang chủ	Thành công	14/05/2023	Tuấn Anh

Bảng 5.7: Kiểm thử thay đổi mật khẩu

STT	Yêu cầu	Kiểm tra	Kết quả mong đợi	Kết quả thực tế	Ngày kiểm thử	Người kiểm thử
1	Người dùng có thể thay đổi mật khẩu hiện tại	Nhập mật khẩu hiện tại và mật khẩu mới	Hiện thị thông báo đổi mật khẩu thành công và đăng nhập lại	Thành công	14/05/2023	Tuấn Anh
2	Người dùng không thể thay đổi mật khẩu nếu chưa nhập đủ thông tin	Không nhập đầy đủ thông tin và bấm nút đổi mật khẩu	Hiện thị thông báo lỗi yêu cầu nhập đầy đủ thông tin	Thành công	14/05/2023	Tuấn Anh
3	Người dùng không thể thay đổi mật khẩu nếu mật khẩu hiện tại không đúng	Nhập mật khẩu hiện tại không đúng và mật khẩu mới	Hiện thị thông báo lỗi yêu cầu nhập đúng mật khẩu hiện tại	Thành công	14/05/2023	Tuấn Anh
4	Mật khẩu phải được ẩn khi người dùng nhập	Nhập mật khẩu	Mật khẩu được ẩn bằng dấu chấm hoặc ký tự khác	Thành công	14/05/2023	Tuấn Anh
5	Người dùng phải xác nhận lại mật khẩu mới để thay đổi mật khẩu	Nhập mật khẩu mới và nhập lại mật khẩu xác nhận	Hiện thị thông báo lỗi nếu mật khẩu xác nhận không khớp	Thành công	14/05/2023	Tuấn Anh

Bảng 5.8: Kiểm thử thống kê

STT	Yêu cầu	Kiểm tra	Kết quả mong đợi	Kết quả thực tế	Ngày kiểm thử	Người kiểm thử
1	Nhân viên có thể xem được thống kê	Nhập tài khoản quản trị viên	Thấy được mục thống kê	Thành công	14/05/2023	Tuấn Anh
2	Khi người dùng mượn thì sẽ cập nhật được bảng thống kê	Người dùng mượn một cuốn sách	Bảng thống kê thay đổi	Thành công	14/05/2023	Tuấn Anh
3	Người dùng trả một cuốn sách thì bảng thống kê vẫn không đổi	Bảng thống kê không đổi	Bảng thống kê không đổi	Thành công	14/05/2023	Tuấn Anh

Chương 6

Kết luận và định hướng phát triển

6.1 Kết luận

Nhóm đã xây dựng được một số tính năng cơ bản một website quản lý sách cần có như: Qua việc thử nghiệm, nhóm em thấy website đã hoạt động tốt trên các tính năng đó và chưa gặp sự cố nào. Tuy nhiên, website của chúng em còn nhiều hạn chế mà chúng em sẽ khắc phục trong tương lai.

6.2 Định hướng phát triển

Nhóm em dự định phát triển thêm một số tính năng như: để giúp website hoàn thiện hơn. Đồng thời cũng sẽ thêm một số chức năng bảo mật và sẽ tạo ra một số những ưu đãi khi người dùng mượn sách hoặc có đóng góp cho thư viện.

Phân chia công việc

STT	Mã Sinh viên	Họ và tên	Công việc	Ghi chú
1	21020294	Nguyễn Khánh Duy	Viết báo cáo	
2	21020276	Nguyễn Tuấn Anh	Viết báo cáo kiểm thử Tạo database(book)	Nhóm trưởng
3	21020011	Nguyễn Trần Đạt	Code	Làm nhiều nhất
4	19020005	Hoàng Gia Anh Đức		Làm ít nhất(Không làm)

Tài liệu tham khảo

- [1] <https://www.w3schools.com/>.
- [2] <https://mongoosejs.com/docs/>.
- [3] <https://viblo.asia/p/su-dung-mysql-voi-node-js-express-6J3Zg2mWKmB>.
- [4] <https://www.npmjs.com/package/mysql>.
- [5] https://hocwebchuan.com/tutorial/reactjs/reactjs_mysql.php.
- [6] <https://viblo.asia/p/su-dung-mongodb-voi-nodejs-ZDEeLXRoeJb>.
- [7] <https://getbootstrap.com/docs/3.4/components/>.
- [8] <https://react.dev/learn>.
- [9] <https://cassandra.apache.org/doc/latest/cassandra/cql/index.html>.
- [10] <https://www.instaclustr.com/support/documentation/cassandra/using-cassandra/connect-to-cassandra-with-node-js/>.
- [11] <https://www.instaclustr.com/support/documentation/cassandra/using-cassandra/connect-to-cassandra-with-node-js/>.
- [12] <https://www.npmjs.com/package/cassandra-driver>.
- [13] <https://www.npmjs.com/package/mongoose>.