Contents

[**Lời mở đầu** 2](#_heading=h.1qrvasqrnc0h)

[**I Giới thiệu và lập kế hoạch** 3](#_heading=h.rnoicog26qey)

[A. Giới thiệu đề tài: Hệ thống quản lý thư viện 3](#_heading=h.i7txt5cui4tp)

[1. Lý do chọn đề tài 3](#_heading=h.ui9swhxyt89o)

[2. Mục tiêu của đề tài 3](#_heading=h.7j7o13w1geit)

[3. Ý nghĩa của đề tài 3](#_heading=h.ub2y4v1k7ft5)

[4. Phạm vi dự án 3](#_heading=h.n4r4isr18ylt)

[5. Giới hạn các chức năng 4](#_heading=h.1lgucdi24ro1)

[6. Hạn chế 4](#_heading=h.aec0w1ex5m3e)

[B. Mô hình quy trình: Agile 5](#_heading=h.64547br33m3g)

[1. Tính linh hoạt cao 5](#_heading=h.8zzc52blq7r3)

[2. Phù hợp với quy mô và phạm vi đề tài 5](#_heading=h.sf01pe3o2t7o)

[3. Tăng cường kiểm thử và phản hồi 5](#_heading=h.55ct4tny0cyg)

[4. Nâng cao hiệu quả làm việc nhóm 5](#_heading=h.tfxf55c7mwle)

[5. Phù hợp với mục tiêu học tập 5](#_heading=h.s7mela4igp87)

[C. Lập kế hoạch 6](#_heading=h.ru34ytcoz27a)

[Bảng phân chia nhiệm vụ các thành viên 6](#_heading=h.rkzpwrw005lx)

[Bảng phân chia nhiệm vụ tuần 6](#_heading=h.lucufh94bel6)

[**II. Phân tích yêu cầu** 7](#_heading=h.csl24x23kq0x)

[A.Yêu cầu chức năng 7](#_heading=h.7pefcvj04b8i)

[1. Quản lý người dùng 7](#_heading=h.5s97b7nqui54)

[2. Quản lý sách 7](#_heading=h.sl751lqo6lt8)

[3. Quản lý mượn – trả 7](#_heading=h.tfnoug38pzzy)

[4.](#_heading=h.4g2qgllq1f3) **Báo cáo – Thống kê** 7

[B. Yêu cầu phi chức năng 7](#_heading=h.b0912qpdq52k)

[2. Bảo mật: 8](#_heading=h.k1w0482gg372)

[3. Khả năng sử dụng 8](#_heading=h.8v3s7zttwdkf)

[4. Khả năng mở rộng 8](#_heading=h.e1iyv198sekw)

[C. Mô hình hóa 8](#_heading=h.zf4ypeilbt4s)

# 

# **Lời mở đầu**

Trong thời đại công nghệ thông tin phát triển mạnh mẽ, việc ứng dụng các hệ thống phần mềm vào quản lý, vận hành đã trở thành xu hướng tất yếu trong nhiều lĩnh vực của đời sống xã hội. Công nghệ không chỉ góp phần tiết kiệm thời gian, giảm thiểu sai sót mà còn nâng cao năng suất và hiệu quả công việc. Đặc biệt, trong lĩnh vực giáo dục và đào tạo, sự hỗ trợ của các hệ thống phần mềm quản lý ngày càng chứng minh được vai trò quan trọng và cần thiết.

Thư viện là nơi lưu trữ, bảo quản và cung cấp tri thức cho người học, giảng viên cũng như cộng đồng nghiên cứu. Tuy nhiên, các phương pháp quản lý truyền thống bằng sổ sách hoặc công cụ thủ công thường gặp phải nhiều khó khăn: khối lượng sách ngày càng lớn, quá trình ghi chép dễ sai sót, việc tra cứu mất nhiều thời gian và khó đáp ứng kịp thời nhu cầu đa dạng của bạn đọc. Điều này không chỉ làm giảm hiệu quả quản lý mà còn ảnh hưởng đến trải nghiệm của người sử dụng thư viện.

Trước thực trạng đó, nhu cầu tin học hóa trong công tác quản lý thư viện trở thành một đòi hỏi cấp thiết. Một hệ thống phần mềm quản lý thư viện có khả năng tự động hóa các nghiệp vụ như quản lý thông tin sách, thông tin độc giả, mượn – trả sách, tìm kiếm và thống kê dữ liệu sẽ giúp khắc phục những hạn chế của cách quản lý truyền thống. Đồng thời, hệ thống cũng giúp tiết kiệm nguồn lực, nâng cao chất lượng phục vụ và góp phần thúc đẩy quá trình số hóa trong hoạt động học tập, nghiên cứu.

Xuất phát từ những yêu cầu trên, nhóm chúng em đã tiến hành nghiên cứu và xây dựng **Hệ thống quản lý thư viện**. Phần mềm này được thiết kế với giao diện thân thiện, dễ sử dụng, tích hợp các chức năng cơ bản như quản lý sách, quản lý người dùng, mượn – trả sách, cùng khả năng mở rộng thêm nhiều tính năng trong tương lai. Thông qua dự án, chúng em mong muốn không chỉ tạo ra một công cụ hỗ trợ hữu ích cho công tác quản lý thư viện mà còn rèn luyện và nâng cao kỹ năng lập trình, phân tích, thiết kế và triển khai phần mềm của bản thân.

Với những nỗ lực và kiến thức tích lũy được trong quá trình học tập, chúng em hy vọng rằng đề tài sẽ mang lại giá trị thiết thực cả về mặt học thuật lẫn ứng dụng thực tiễn, đồng thời là bước khởi đầu quan trọng để tiếp tục phát triển những hệ thống phần mềm phục vụ nhu cầu quản lý trong nhiều lĩnh vực khác.

# **I Giới thiệu và lập kế hoạch**

## Giới thiệu đề tài: Hệ thống quản lý thư viện

### Lý do chọn đề tài

Trong thực tế, việc quản lý thư viện theo phương pháp thủ công còn nhiều bất cập như: khó kiểm soát số lượng sách lớn, dễ sai sót khi ghi chép, mất nhiều thời gian tra cứu, khó đáp ứng nhu cầu đa dạng của bạn đọc.  
Để khắc phục những hạn chế đó, nhóm lựa chọn đề tài “Hệ thống quản lý thư viện” với mục tiêu ứng dụng công nghệ thông tin nhằm tự động hóa các nghiệp vụ cơ bản của thư viện.

### Mục tiêu của đề tài

* Xây dựng hệ thống quản lý thư viện có khả năng quản lý thông tin sách, độc giả, mượn – trả sách một cách hiệu quả.
* Thiết kế giao diện trực quan, dễ sử dụng cho cả cán bộ quản lý và người dùng.
* Ứng dụng kiến thức lập trình, cơ sở dữ liệu và quy trình phát triển phần mềm vào thực tế.
* Tạo tiền đề cho việc mở rộng và nâng cấp hệ thống trong tương lai (thêm tính năng đặt sách online, quản lý tài liệu điện tử,…).

### Ý nghĩa của đề tài

* **Về mặt thực tiễn:** Đề tài góp phần giải quyết những hạn chế của phương pháp quản lý thư viện thủ công, giảm tải công việc cho cán bộ thư viện, nâng cao hiệu quả quản lý và chất lượng phục vụ bạn đọc.
* **Về mặt học thuật:** Đề tài giúp người thực hiện củng cố và vận dụng các kiến thức đã học về lập trình, cơ sở dữ liệu, phân tích và thiết kế hệ thống, đồng thời rèn luyện kỹ năng làm việc nhóm và quản lý dự án.
* **Về mặt xã hội:** Hỗ trợ thúc đẩy quá trình số hóa trong lĩnh vực giáo dục và đào tạo, đáp ứng nhu cầu học tập và nghiên cứu ngày càng cao của sinh viên, giảng viên và độc giả.

### Phạm vi dự án

* Hệ thống tập trung vào các chức năng cơ bản: quản lý thông tin sách, quản lý người dùng, mượn – trả sách và tra cứu sách.
* Đối tượng sử dụng: cán bộ thư viện và độc giả.
* Công nghệ sử dụng: Django kết hợp với hệ quản trị cơ sở dữ liệu để đảm bảo khả năng lưu trữ và xử lý dữ liệu.
* p chỉ dừng lại ở mức độ thử nghiệm và phục vụ mục đích học tập, chưa triển khai quy mô lớn trong thực tế.

### Giới hạn các chức năng

Trong khuôn khổ đề tài **Hệ thống quản lý thư viện**, dự án tập trung triển khai các chức năng cơ bản và cần thiết nhất, cụ thể như sau:

1. **Quản lý sách**
   * Thêm, sửa, xóa thông tin sách.
   * Lưu trữ các thuộc tính cơ bản: tên sách, tác giả, thể loại, năm xuất bản, số lượng.
   * Tra cứu, tìm kiếm sách theo tên, tác giả hoặc thể loại.
2. **Quản lý độc giả/người dùng**
   * Đăng ký tài khoản người dùng.
   * Lưu trữ thông tin cơ bản: họ tên, mã độc giả, tài khoản đăng nhập.
   * Quản lý danh sách người dùng (thêm, sửa, xóa).
3. **Quản lý mượn – trả sách**
   * Ghi nhận thông tin mượn sách: độc giả, tên sách, ngày mượn, ngày trả dự kiến.
   * Cập nhật trạng thái trả sách, kiểm soát số lượng sách còn lại trong kho.
   * Thống kê cơ bản số lượt mượn – trả.
4. **Đăng nhập và phân quyền**
   * Người dùng đăng nhập để sử dụng hệ thống.
   * Phân biệt quyền quản trị viên (quản lý toàn bộ hệ thống) và người dùng (mượn/trả, tra cứu sách).

### Hạn chế

* Hệ thống chỉ hỗ trợ các nghiệp vụ cơ bản, chưa triển khai các tính năng nâng cao như: đặt sách online, quản lý tài liệu số, quét mã QR, tính phí phạt trễ hạn, hay tích hợp hệ thống email thông báo.
* Dữ liệu chỉ mang tính thử nghiệm trong phạm vi nhỏ, chưa triển khai trên quy mô thư viện lớn.

## Mô hình quy trình: Agile

Trong quá trình phát triển phần mềm, việc lựa chọn mô hình phát triển phù hợp đóng vai trò quan trọng, ảnh hưởng trực tiếp đến tiến độ, chất lượng và khả năng hoàn thiện sản phẩm. Với đề tài **Hệ thống quản lý thư viện**, nhóm chúng em quyết định áp dụng **mô hình Agile** bởi những lý do sau:

### Tính linh hoạt cao

* + Agile cho phép chia nhỏ quá trình phát triển thành nhiều vòng lặp (iteration/sprint). Ở mỗi vòng lặp, nhóm có thể xây dựng và hoàn thiện một phần chức năng của hệ thống.
  + Nếu có thay đổi trong yêu cầu hoặc phát hiện sai sót, nhóm có thể dễ dàng điều chỉnh ở vòng tiếp theo, tránh lãng phí thời gian như mô hình Waterfall (tuần tự, khó thay đổi).

### Phù hợp với quy mô và phạm vi đề tài

* + Đề tài chỉ tập trung vào một số chức năng cơ bản (quản lý sách, quản lý người dùng, mượn – trả sách, đăng nhập và phân quyền).
  + Với Agile, nhóm có thể ưu tiên phát triển các chức năng cốt lõi trước, sau đó bổ sung dần các chức năng phụ trợ, đảm bảo hệ thống luôn có phiên bản chạy được để kiểm thử và đánh giá.

### Tăng cường kiểm thử và phản hồi

* + Sau mỗi sprint, nhóm đều có thể chạy thử hệ thống, kiểm tra lỗi và nhận phản hồi từ giảng viên hoặc chính thành viên trong nhóm.
  + Điều này giúp sản phẩm được cải tiến liên tục, chất lượng tốt hơn so với việc kiểm thử ở cuối dự án.

### Nâng cao hiệu quả làm việc nhóm

* + Agile khuyến khích sự tương tác, trao đổi thường xuyên giữa các thành viên. Nhóm có thể họp nhanh (daily meeting) để cập nhật tiến độ, phân công công việc và giải quyết khó khăn kịp thời.
  + Nhờ đó, tiến độ được kiểm soát chặt chẽ, tránh tình trạng dồn việc hoặc bỏ sót chức năng.

### Phù hợp với mục tiêu học tập

* + Việc áp dụng Agile giúp nhóm rèn luyện kỹ năng quản lý dự án, làm việc nhóm và phản ứng nhanh với thay đổi – đây đều là những kỹ năng quan trọng trong thực tế phát triển phần mềm.

**Kết luận:** Với những ưu điểm về tính linh hoạt, khả năng thích ứng với thay đổi, tăng cường kiểm thử và tính cộng tác cao, mô hình Agile là lựa chọn phù hợp để phát triển hệ thống quản lý thư viện trong khuôn khổ thời gian và phạm vi của đề tài này.

## Lập kế hoạch

### Bảng phân chia nhiệm vụ các thành viên

|  |  |  |
| --- | --- | --- |
| STT | Tên thành viên | Nhiệm vụ đảm nhận |
| 1 | Nguyễn Trường Sơn |  |
| 2 | Nguyễn Đăng Hiếu |  |

### Bảng phân chia nhiệm vụ tuần



# **II. Phân tích yêu cầu**

## A.Yêu cầu chức năng

Hệ thống quản lý thư viện cần đáp ứng các chức năng chính sau:

### Quản lý người dùng

* + Đăng ký tài khoản người dùng.
  + Đăng nhập/đăng xuất hệ thống.
  + Quản lý thông tin cá nhân.
  + Phân quyền: Quản trị viên (Admin) và Người dùng (User).

### Quản lý sách

* + Thêm mới sách.
  + Sửa thông tin sách.
  + Xóa sách.
  + Tìm kiếm/tra cứu sách theo tên, tác giả, thể loại.

### Quản lý mượn – trả

* + Đăng ký mượn sách.
  + Cập nhật thông tin trả sách.
  + Quản lý trạng thái sách (còn/trả/hết).
  + Thống kê lượt mượn – trả.

### Báo cáo – Thống kê

* + Thống kê số lượng sách theo thể loại.
  + Thống kê số lượt mượn của từng người dùng.
  + Báo cáo số lượng sách còn lại trong kho.

## B. Yêu cầu phi chức năng

Ngoài các chức năng chính, hệ thống cần đáp ứng một số yêu cầu phi chức năng như sau:

**1.** Hiệu năng:

* + Thời gian tải trang/dữ liệu trung bình < 2 giây.
  + Xử lý đồng thời tối thiểu 20 – 30 người dùng truy cập.

### 2. Bảo mật:

* + Phân quyền rõ ràng: Admin có quyền quản lý toàn bộ dữ liệu; User chỉ có quyền mượn/trả, tra cứu sách.
  + Dữ liệu nhạy cảm (mật khẩu) được mã hóa.
  + Hệ thống có cơ chế đăng nhập an toàn

### 3. Khả năng sử dụng

* + Giao diện thân thiện, dễ sử dụng, hỗ trợ tìm kiếm nhanh.
  + Các chức năng quan trọng (mượn, trả, tìm kiếm sách) dễ truy cập trong 1–2 thao tác.
  + Có thông báo lỗi/nhắc nhở khi nhập sai dữ liệu hoặc thao tác không hợp lệ.

### 4. Khả năng mở rộng

* + Hệ thống có thể bổ sung thêm chức năng mới (ví dụ: đặt sách online, email nhắc nhở).
  + Cơ sở dữ liệu được thiết kế linh hoạt, dễ mở rộng quy mô.

## C. Mô hình hóa

Các Actor chính

- Người dùng (User / Độc giả)

- Thủ thư(Librarian / Nhân viên quản lý thư viện)

- Quản trị viên (Admin)

- Hệ thống (tác nhân phụ trợ: xử lý, gửi thông báo)

Use Case

**Người dùng (User)**

- Đăng ký / Đăng nhập

- Tìm kiếm sách

- Xem danh sách và chi tiết sách

- Mượn sách

- Trả sách

- Gia hạn sách

- Viết đánh giá / nhận xét

**Thủ thư (Librarian)**

- Quản lý sách (thêm, cập nhật, xóa)

- Quản lý người dùng (tạo, khóa, mở tài khoản)

- Quản lý phiếu mượn – trả

- Kiểm tra tình trạng sách

**Quản trị viên (Admin)**

- Xem báo cáo thống kê (sách mượn nhiều, tồn kho, người dùng vi phạm)

- Quản lý thủ thư (tạo/sửa/xoá tài khoản thủ thư)

- Quản lý hệ thống

**Yêu cầu chức năng (Functional Requirements)**

- Hệ thống cho phép người dùng đăng ký/đăng nhập tài khoản.

- Người dùng tìm kiếm sách theo tên, tác giả, thể loại.

- Người dùng mượn/trả/gia hạn sách trong thời gian quy định.

- Người dùng có thể viết nhận xét, đánh giá sách.

- Quản trị viên quản lý dữ liệu sách và người dùng.

- Quản trị viên theo dõi phiếu mượn – trả và xuất báo cáo.

**Yêu cầu phi chức năng**

- Hiệu năng: Thời gian phản hồi ≤ 2 giây khi tra cứu.

- Bảo mật: Mật khẩu mã hóa, phân quyền User/Admin.

- Khả dụng: Hoạt động 24/7, downtime < 1%.

- Thân thiện: Giao diện dễ sử dụng, hỗ trợ tìm kiếm nhanh.

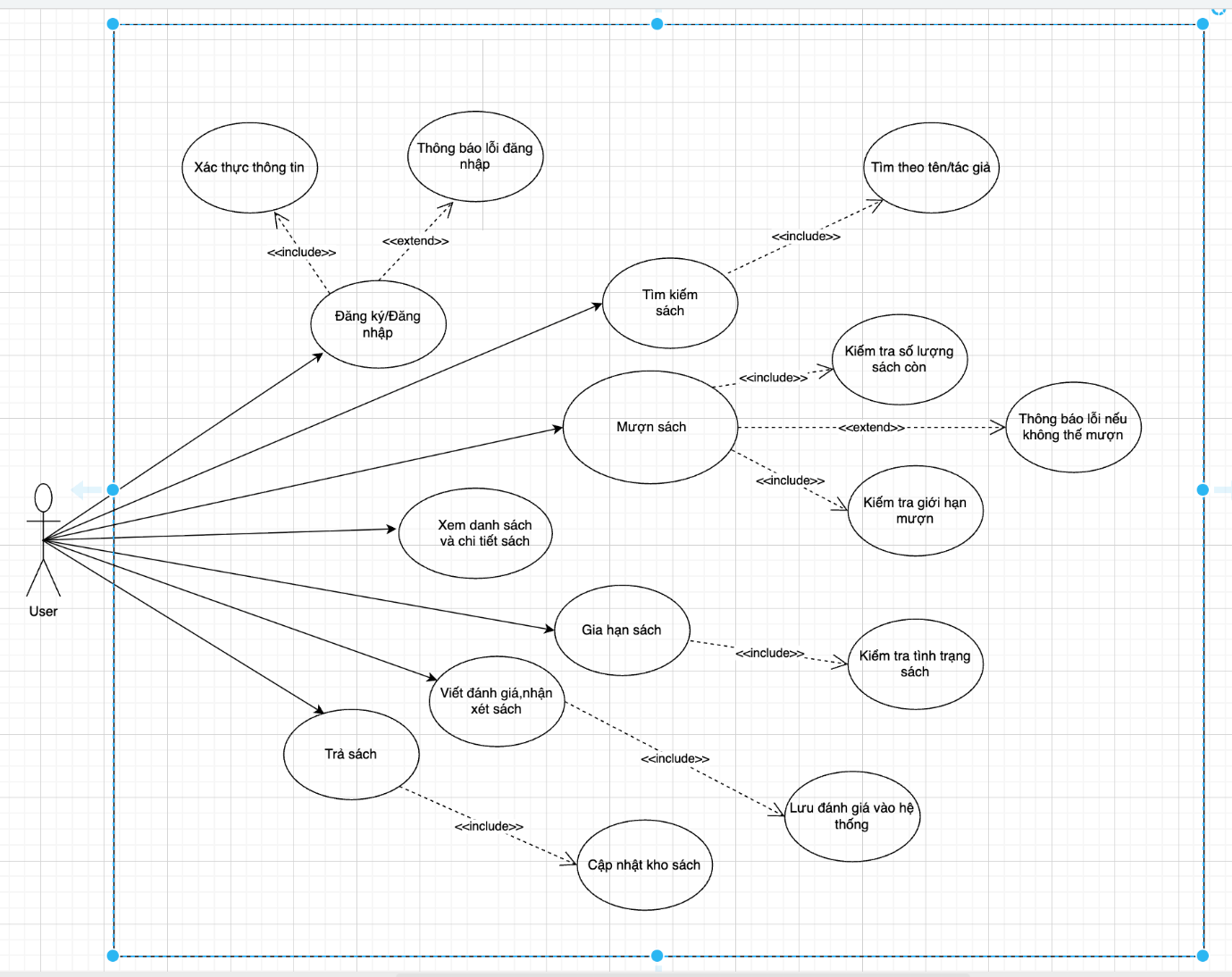
- Mở rộng: Hỗ trợ số lượng lớn sách và người dùng.

Mối quan hệ Actor – Use Case

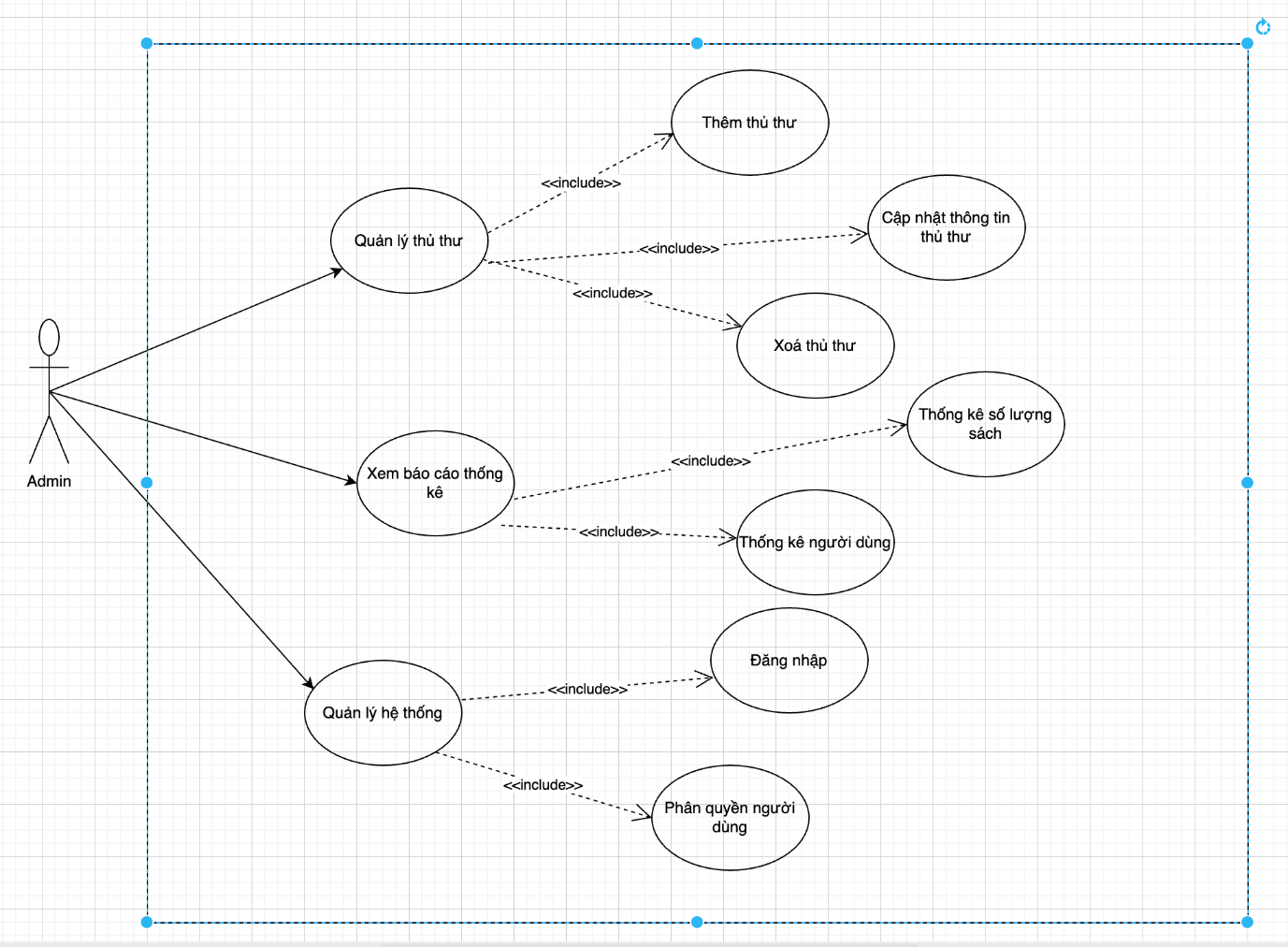
|  |  |
| --- | --- |
| Actor | Use case |
| Người dùng | Đăng ký/Đăng nhập, tìm kiếm sách, xem chi tiết, mượn sách, trả sách, gia hạn sách, đánh giá |
| Thủ thư | Quản lý sách, quản lý người dùng, quản lý mượn-trả |
| Quản trị viên | Quản lý thủ thư, Quản lý hệ thống |

Biểu đồ Use Case

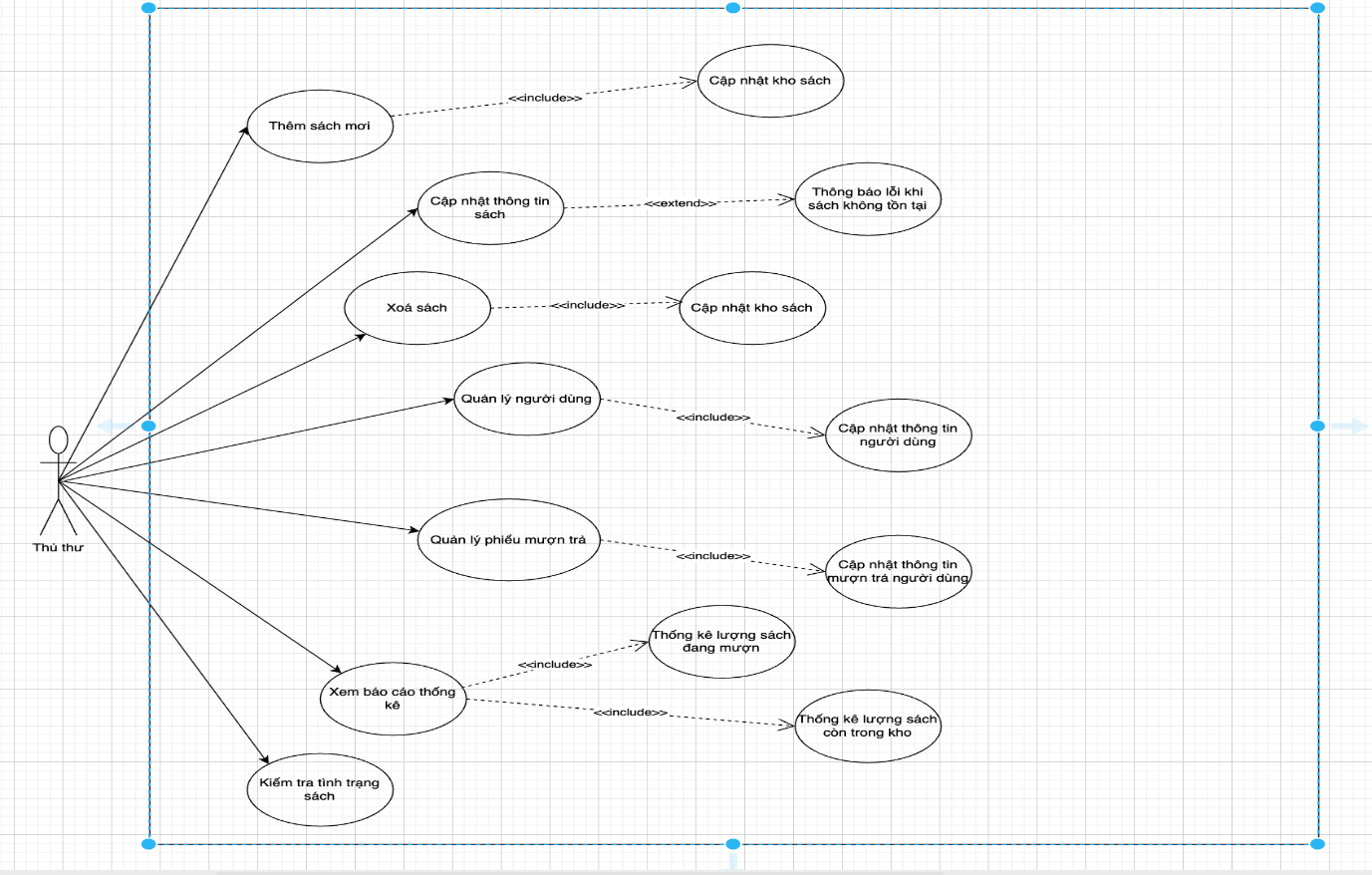
- Người dùng ↔ (Đăng ký/Đăng nhập, Tìm kiếm sách, Xem chi tiết, Mượn, Trả, Gia hạn, Đánh giá)



- Admin ↔ (Xem báo cáo thống kê,quản lý thủ thư,quản lý hệ thống)



- Thủ thư ↔ (Quản lý sách, Quản lý người dùng, Quản lý mượn – trả,Kiểm tra tình trạng sách)



Use Case - Quản lý thư viện

Use Case 1: Mượn sách

Mục tiêu

Độc giả mượn thành công sách từ thư viện.

Tác nhân chính

- Độc giả (hoặc thủ thư nếu hệ thống cho phép mượn qua quầy).

Điều kiện tiên quyết

- Độc giả đã đăng nhập vào hệ thống.

- Sách cần mượn còn trong kho (chưa bị mượn hết).

Kịch bản chính (Primary Scenario)

1. Độc giả chọn chức năng "Mượn sách"

2. Hệ thống hiển thị danh sách sách hiện có (tên, mã sách, tác giả, số lượng còn lại).

3. Độc giả nhập hoặc chọn mã sách muốn mượn.

4. Hệ thống kiểm tra:

- Độc giả có còn hạn thẻ thư viện không.

- Độc giả có vi phạm quy định (nợ sách, quá hạn) không.

- Sách còn số lượng không.

5. Nếu hợp lệ, hệ thống ghi nhận giao dịch mượn sách (ngày mượn, ngày phải trả).

6. Hệ thống trừ đi số lượng sách còn trong kho.

7. Hệ thống thông báo “Mượn sách thành công”.

Use Case 2: Trả sách

Mục tiêu

Độc giả trả sách đã mượn.

Tác nhân chính

- Độc giả (hoặc thủ thư).

Điều kiện tiên quyết

- Độc giả đã mượn sách trước đó.

Kịch bản chính (Primary Scenario)

1. Độc giả chọn chức năng "Trả sách"

2. Hệ thống hiển thị danh sách sách mà độc giả đang mượn.

3. Độc giả chọn sách cần trả.

4. Hệ thống cập nhật trạng thái sách là “đã trả”.

5. Hệ thống tăng số lượng sách còn lại trong kho.

6. Nếu trả đúng hạn → Thông báo “Trả sách thành công”.

7. Nếu trả trễ hạn → Hệ thống tính phí phạt và thông báo cho độc giả.

Use Case 3: Quản lý sách (Thêm sách mới)

Mục tiêu

Thủ thư/Quản trị viên thêm sách mới vào hệ thống.

Tác nhân chín

- Thủ thư / Quản trị viên.

Điều kiện tiên quyết

- Người dùng có quyền quản trị.

Kịch bản chính (Primary Scenario)

1. Thủ thư chọn chức năng "Thêm sách mới".

2. Hệ thống hiển thị form nhập thông tin sách (mã sách, tên sách, tác giả, thể loại, số lượng, năm xuất bản).

3. Thủ thư nhập đầy đủ thông tin và xác nhận.

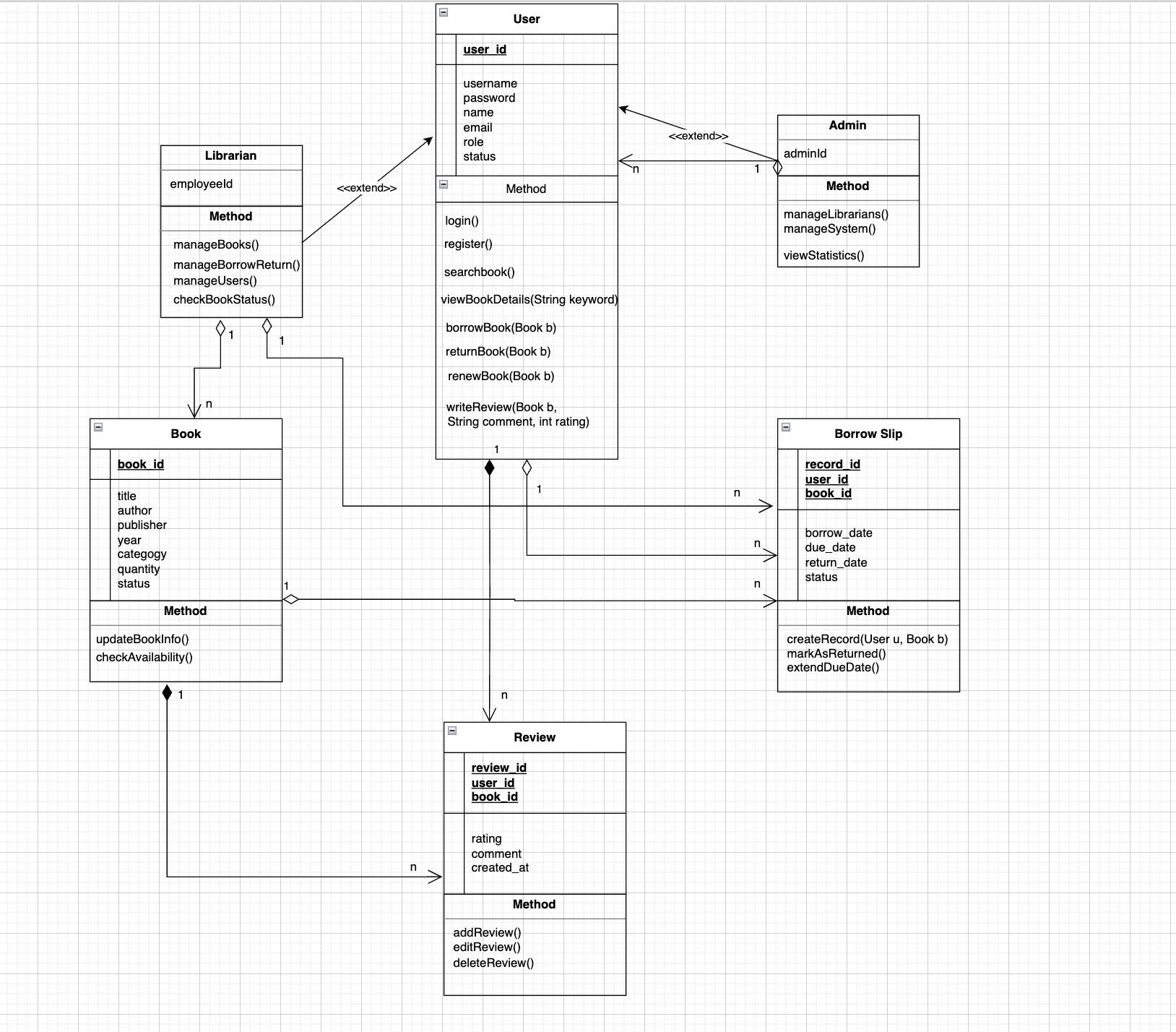
4. Hệ thống kiểm tra dữ liệu (mã sách có trùng không, số lượng hợp lệ).

5. Nếu hợp lệ, hệ thống lưu thông tin sách vào cơ sở dữ liệu.

6. Hệ thống thông báo “Thêm sách thành công”.

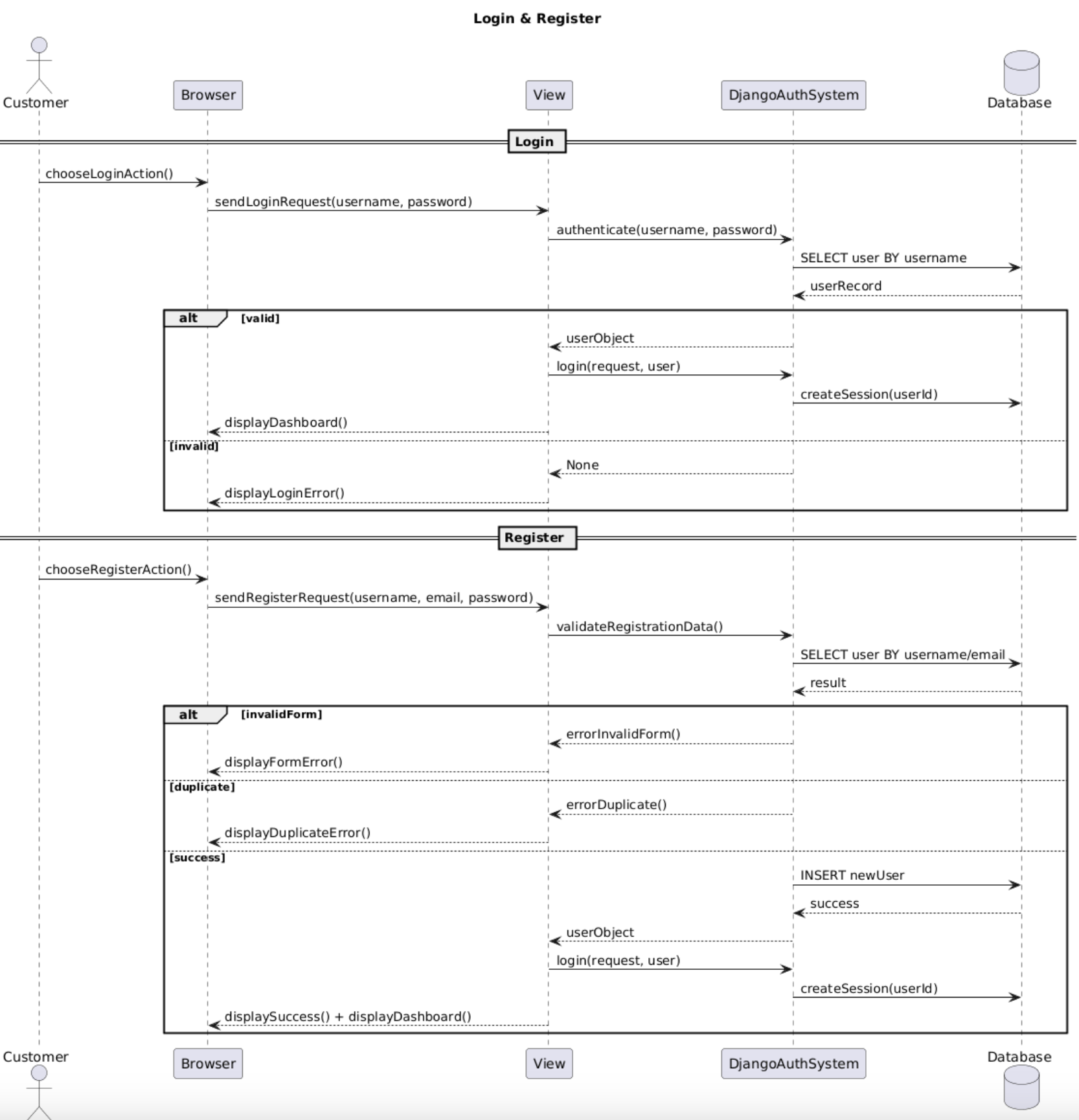
7. Sách mới xuất hiện trong danh mục để độc giả có thể tra cứu/mượn.

- Thiết kế sơ đồ lớp

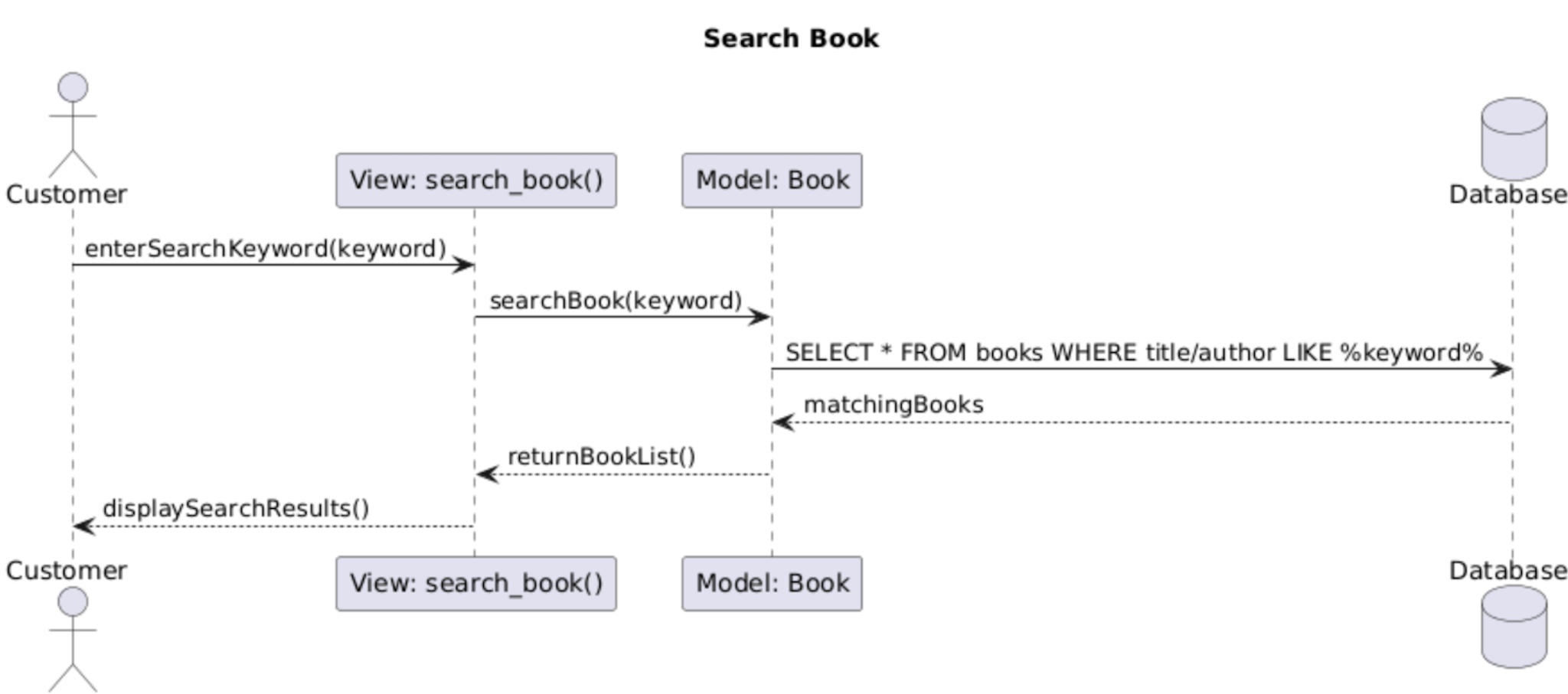


Sequence Diagram

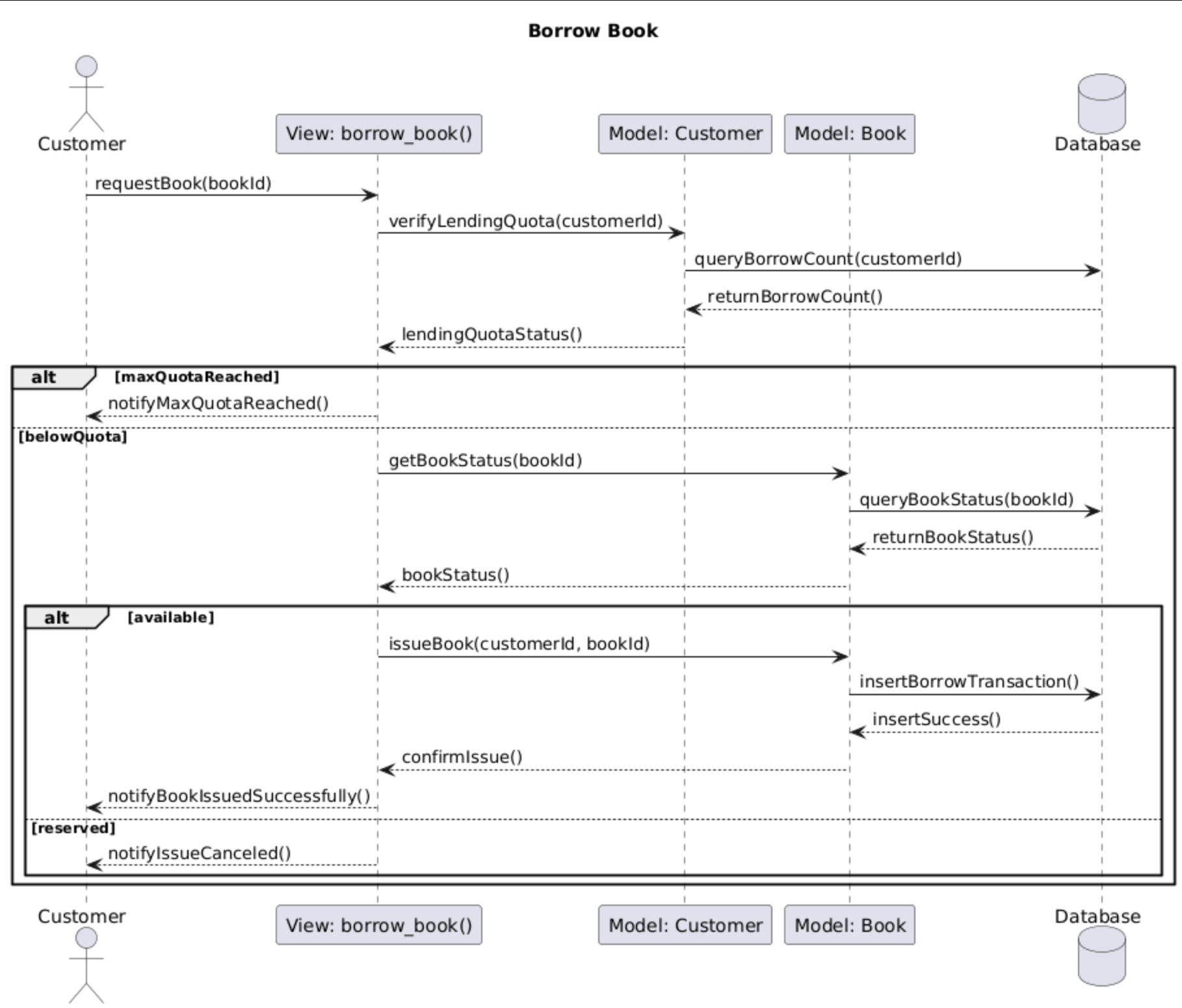
Sơ đồ đăng ký và đăng nhập



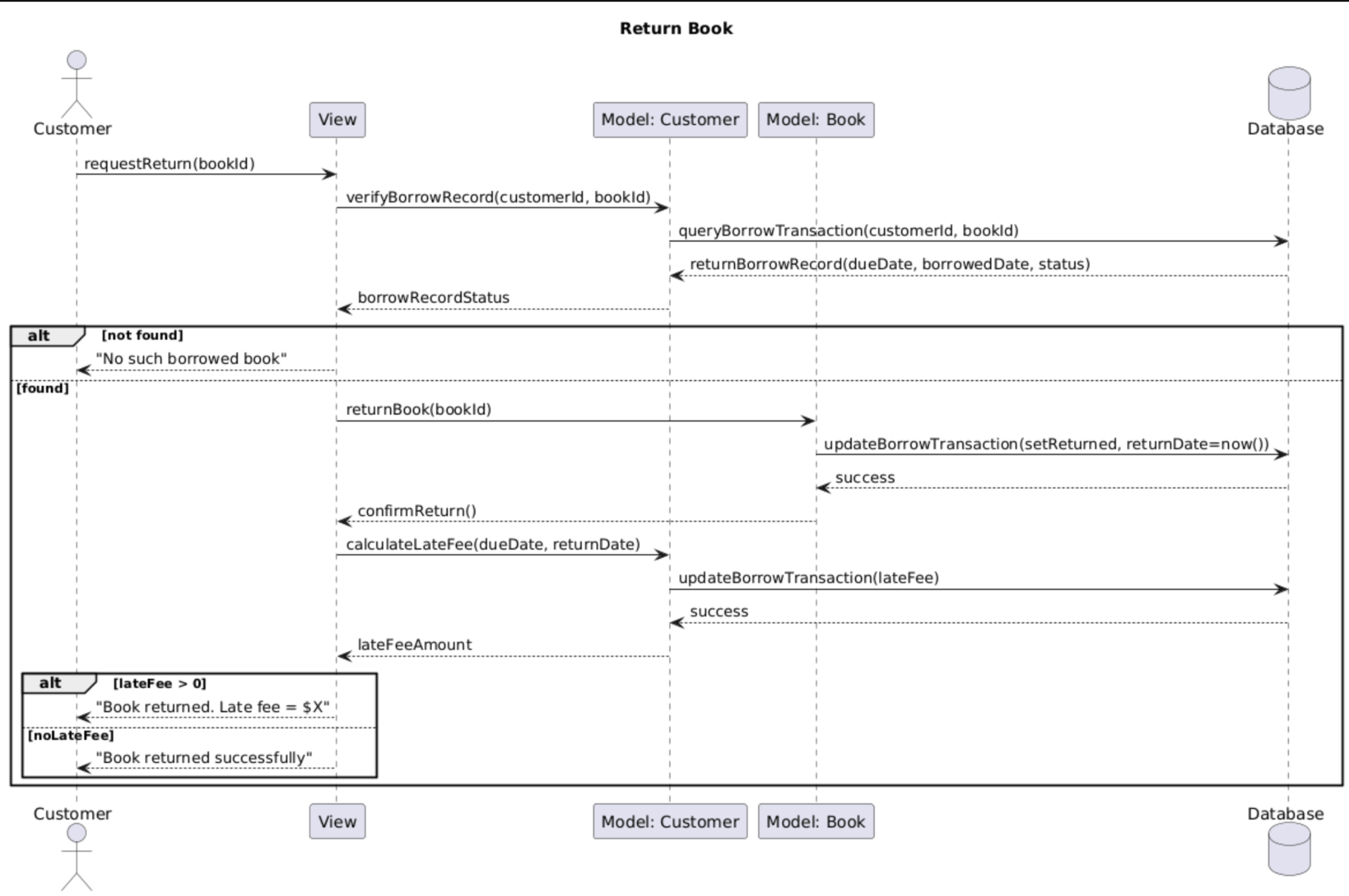
Sơ đồ tìm kiếm sách



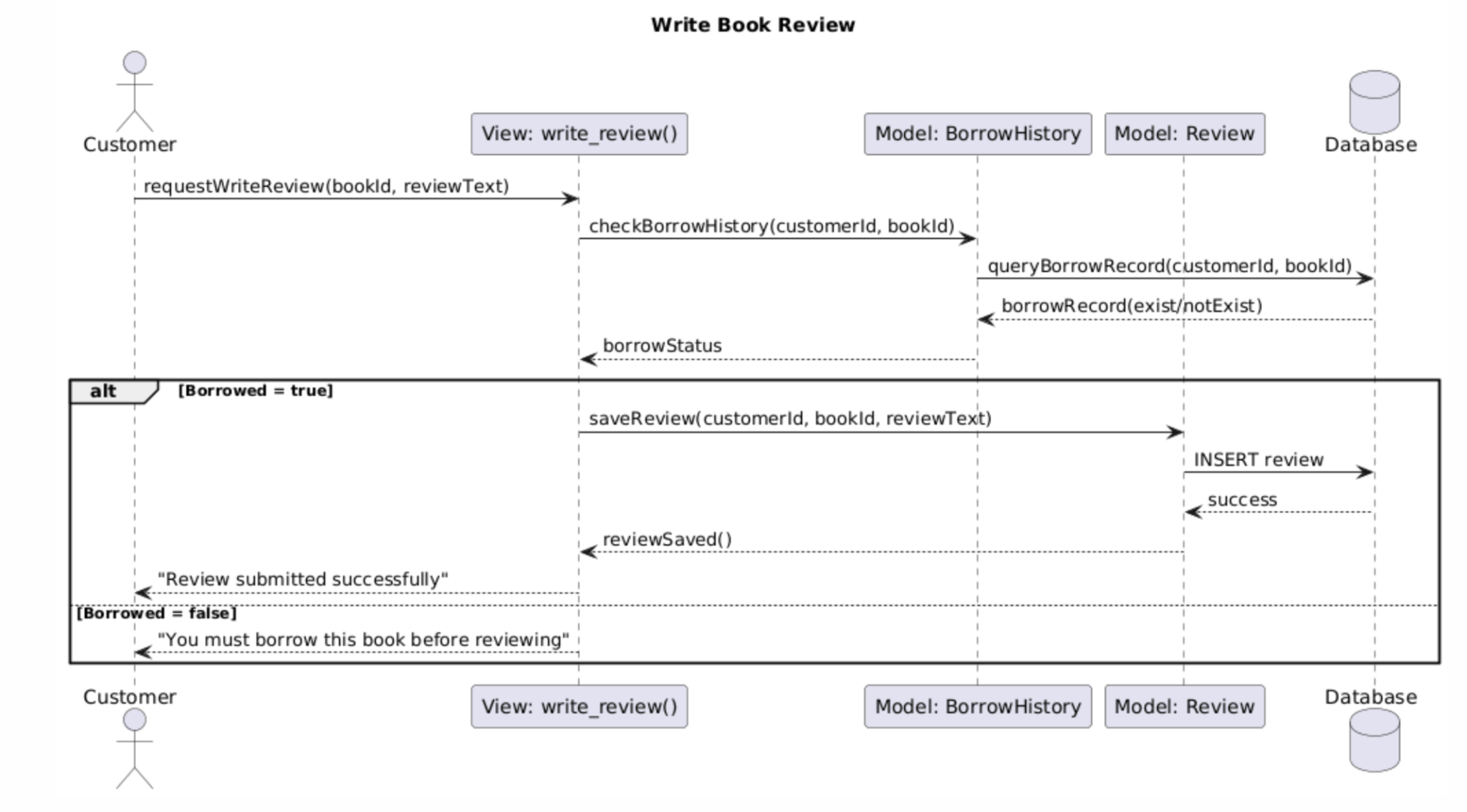
Sơ đồ mượn sách



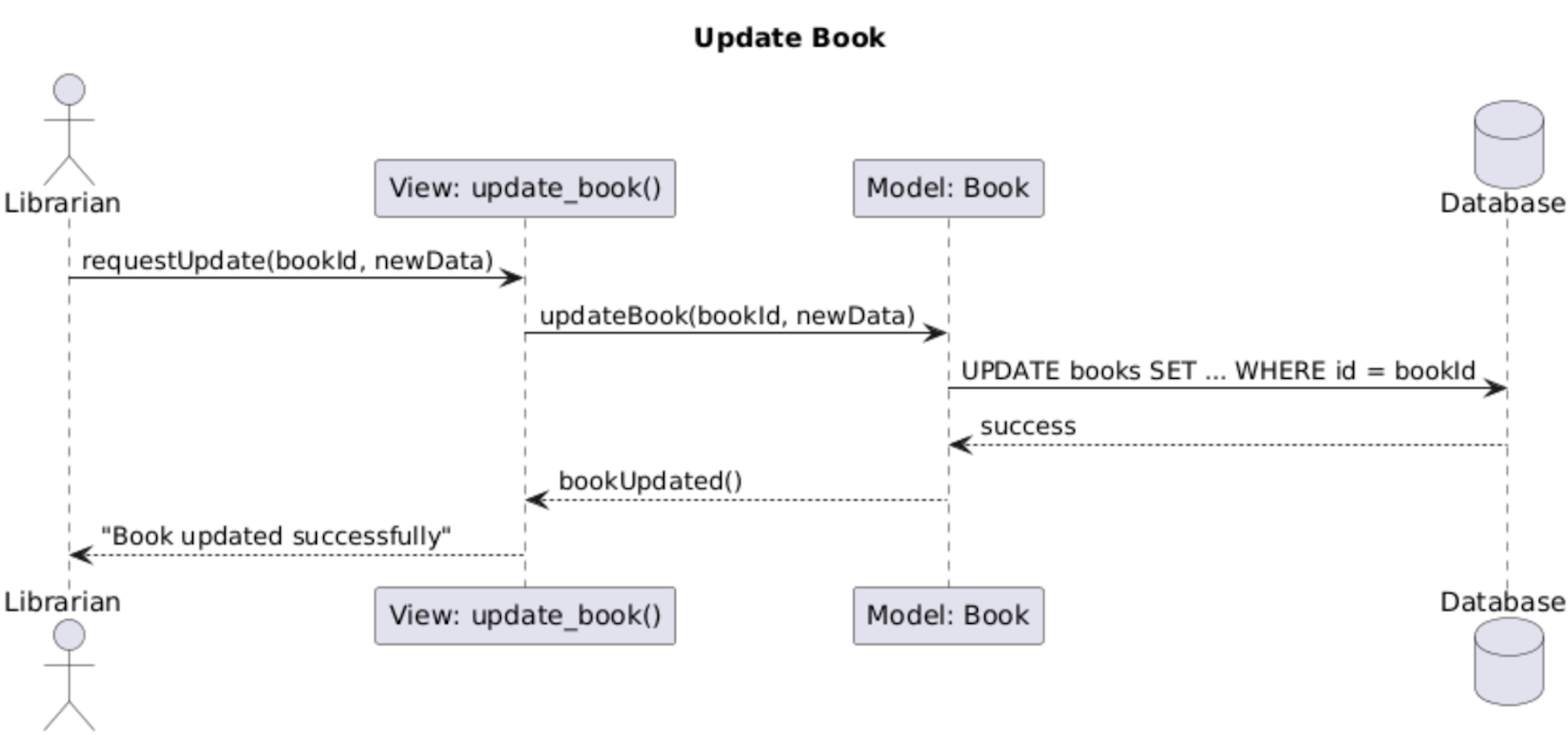
Sơ đồ trả sách



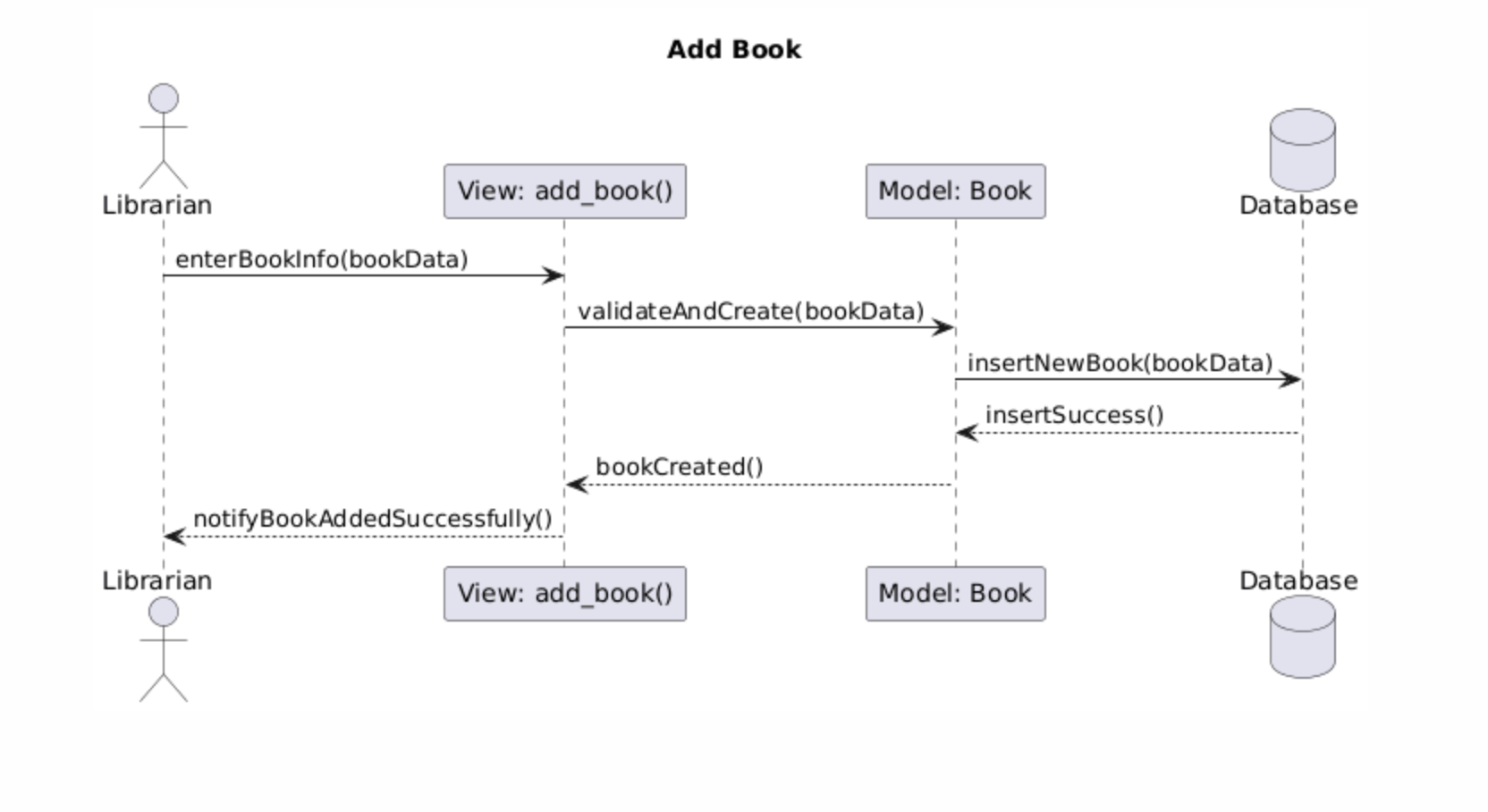
Sơ đồ viết đánh giá sách



Sơ đồ cập nhật sách



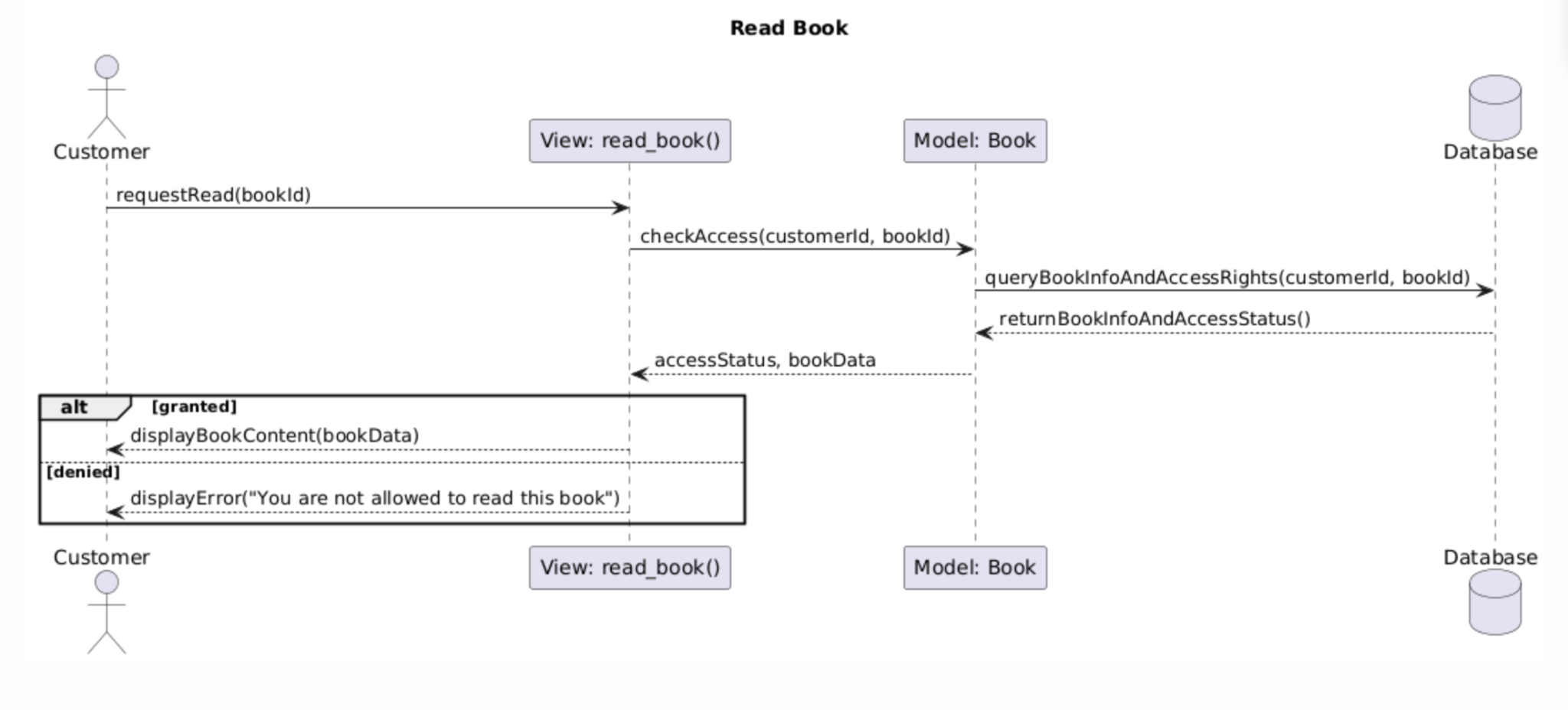
Sơ đồ thêm sách



Sơ đồ xóa sách

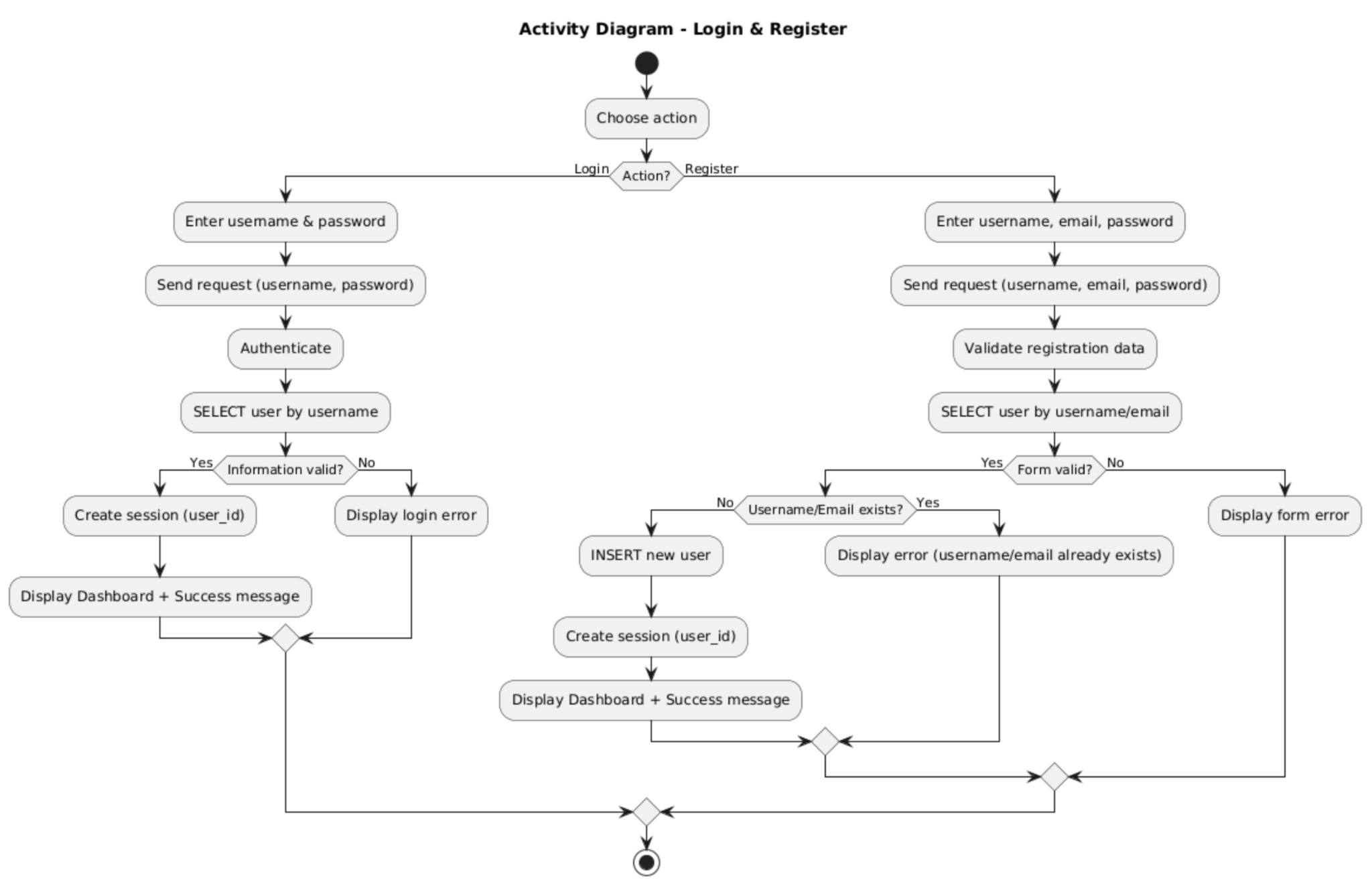


Sơ đồ đọc sách

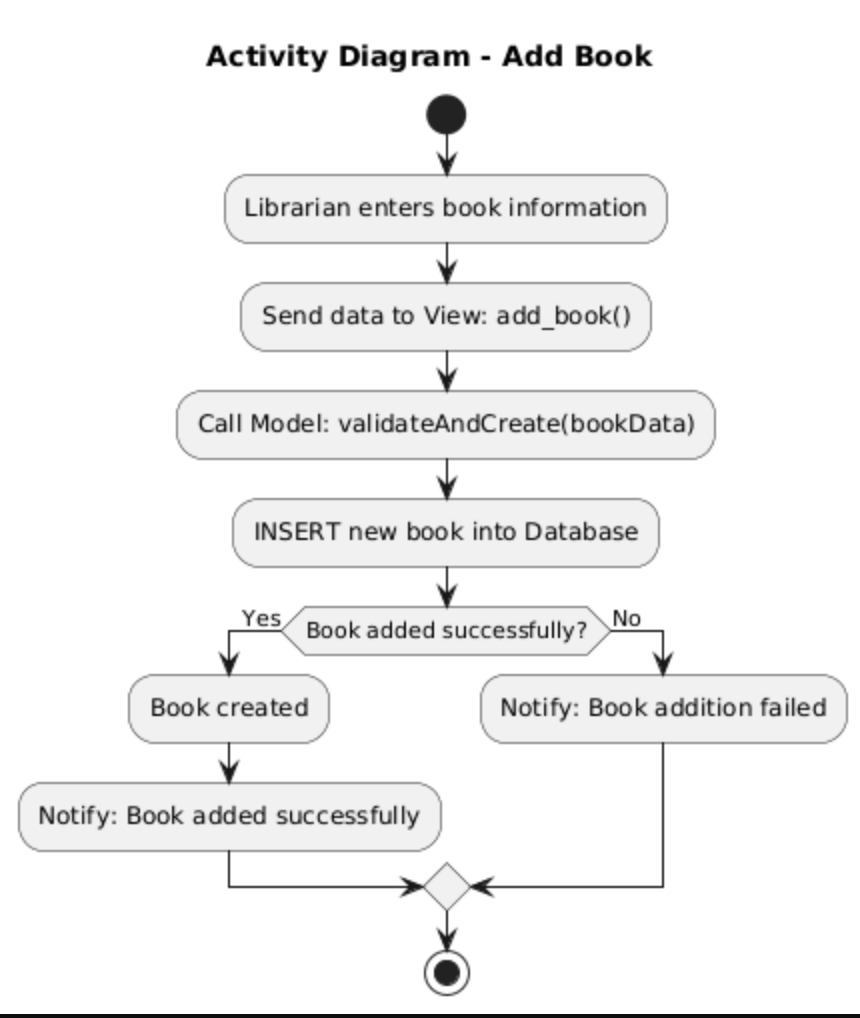


Biểu đồ hoạt động

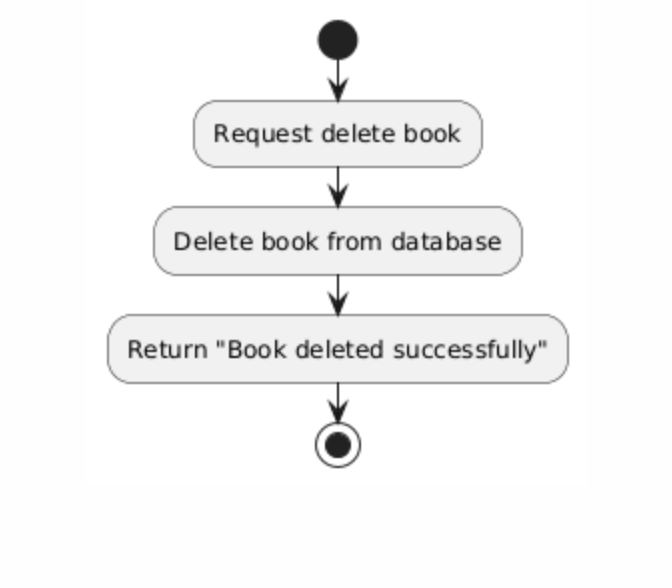
Đăng nhập và đăng ký



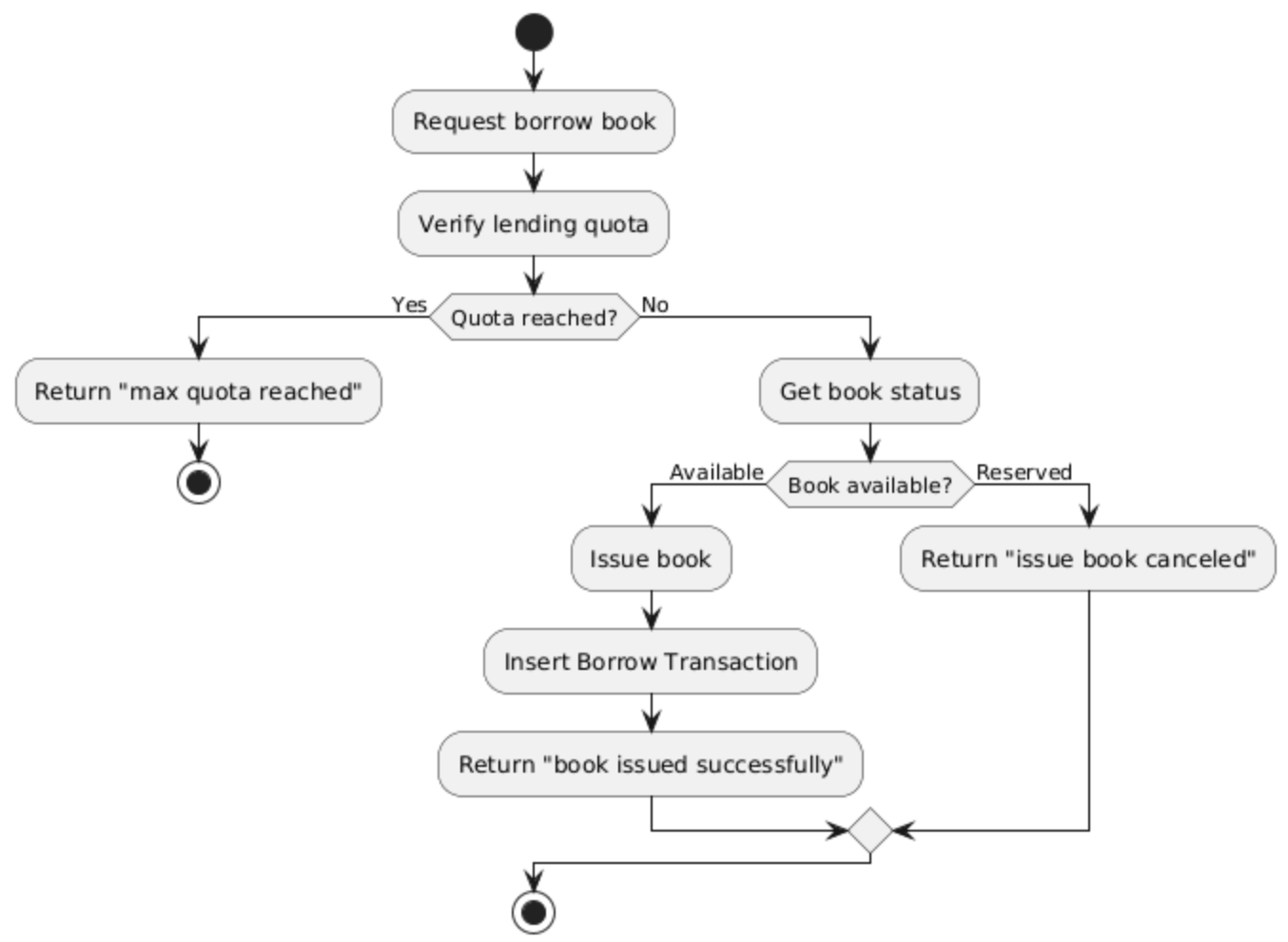
Thêm sách



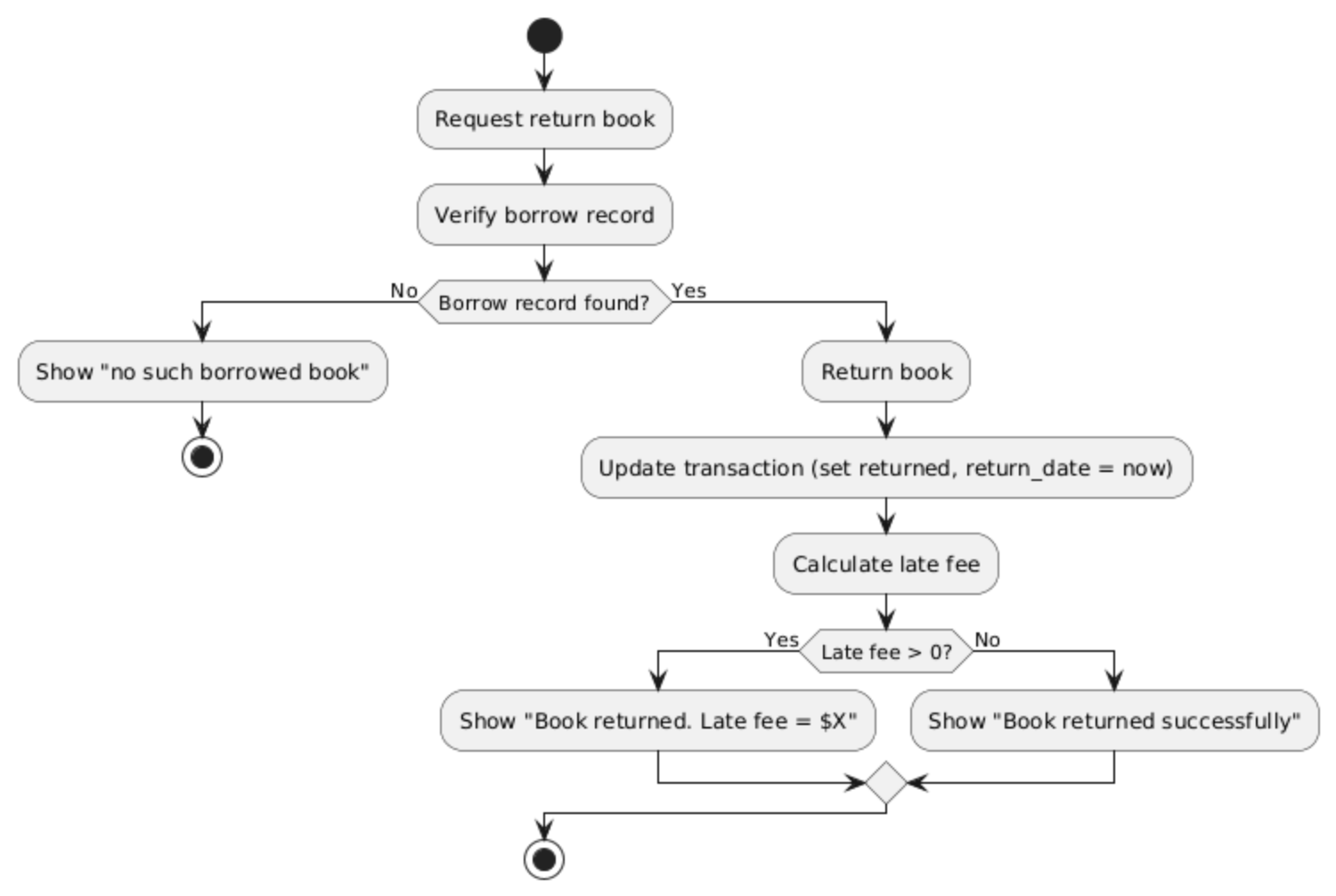
Xoá sách



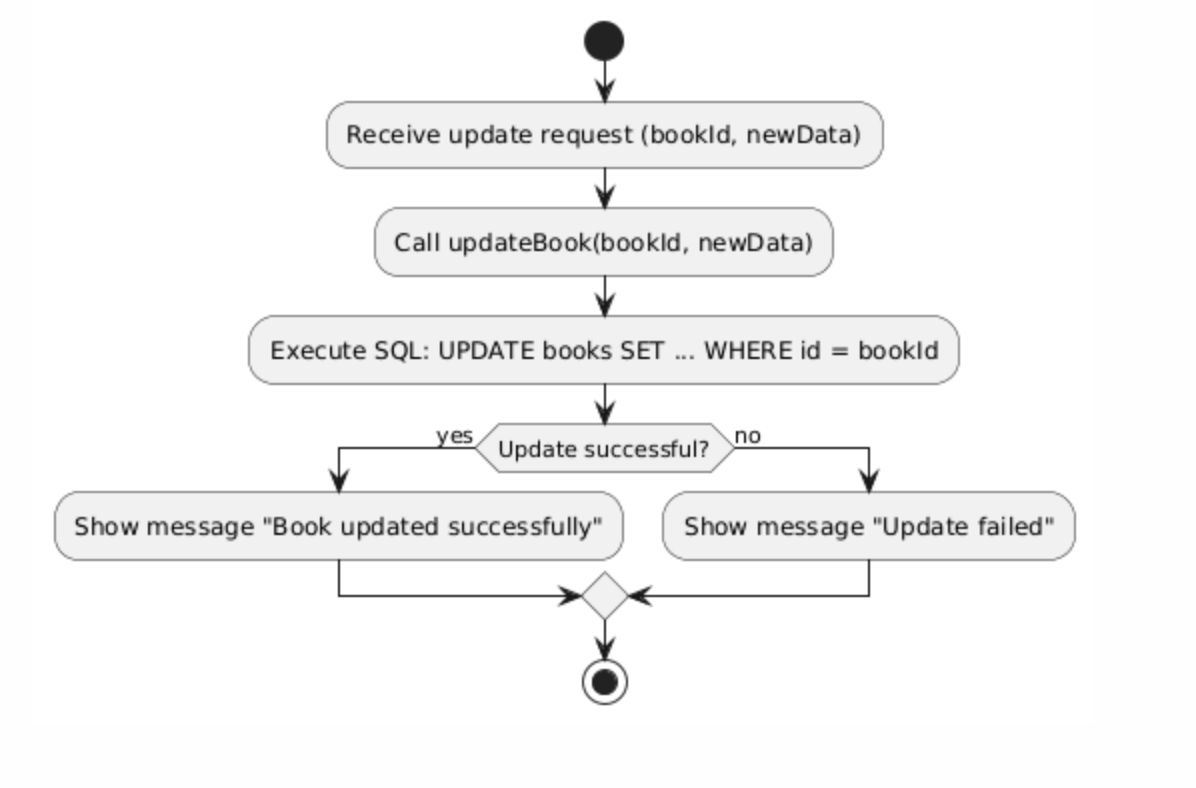
Mượn sách



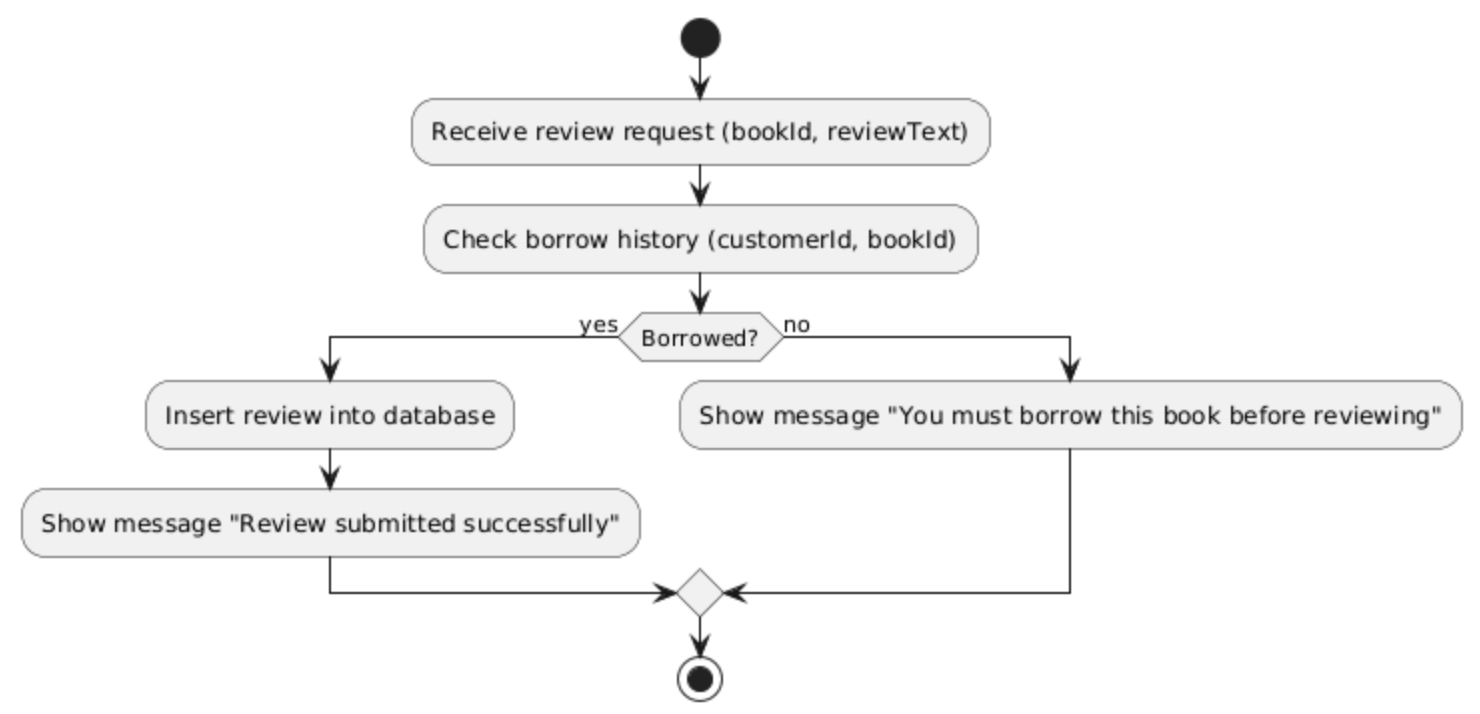
Trả sách



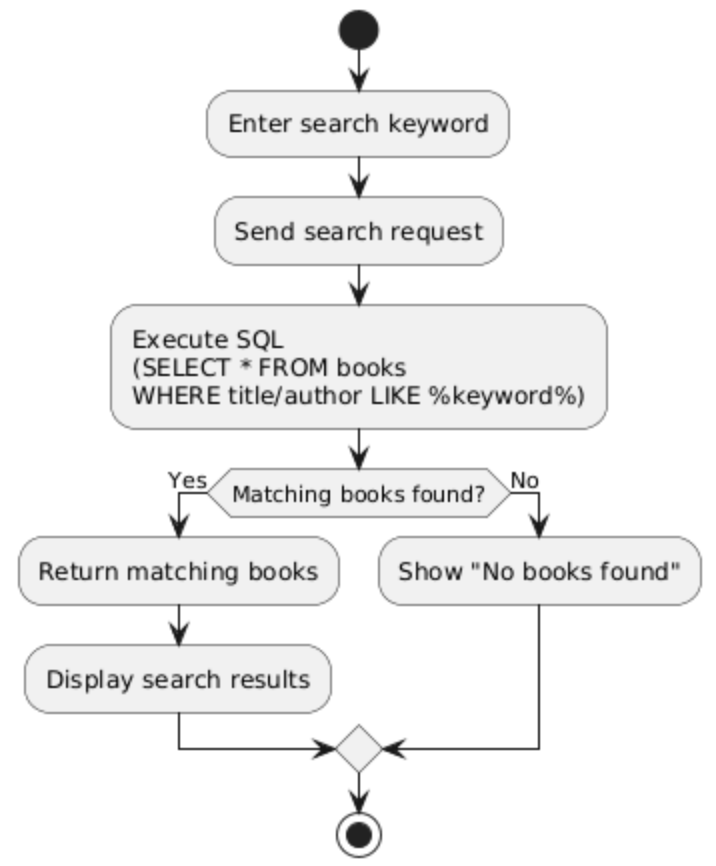
Cập nhật sách



Viết đánh giá

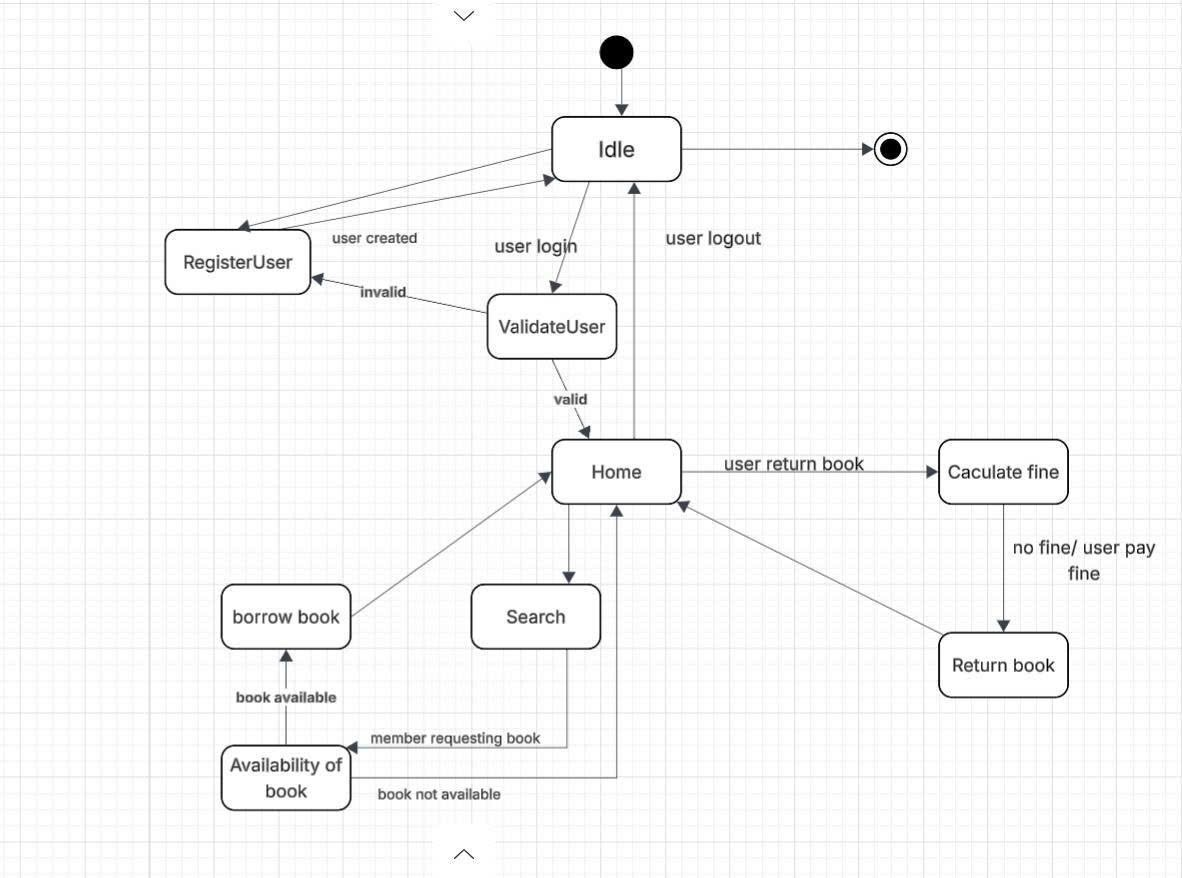


Tìm kiếm sách

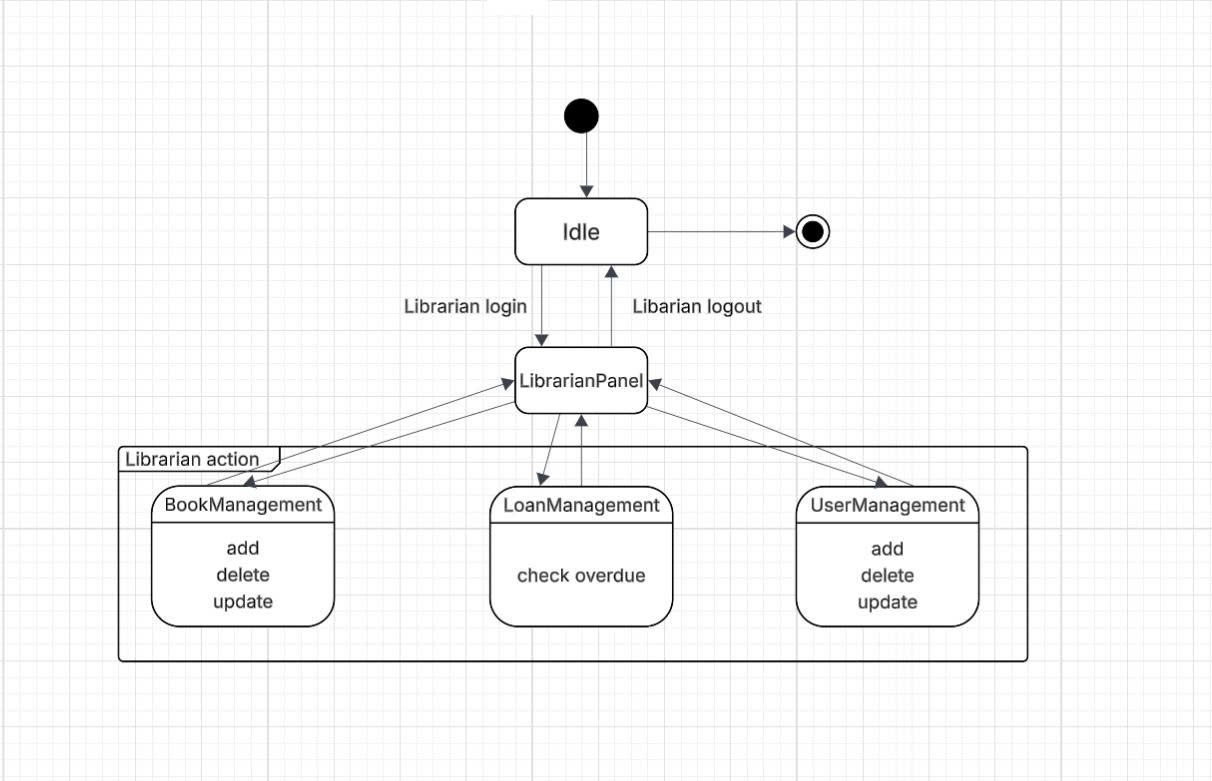


Sơ đồ trạng thái

User



Librarian



Giao diện figma của nhóm

<https://www.figma.com/design/s7yvlSoelriKwyylcqdRp1/Libary-management-system?node-id=0-1&t=UDabU9SuIbZjMNbW-1>