Chương 1: Giới thiệu đề tài

* 1. Tổng quan.

Dự án của chúng em là một trang web đọc sách trực tuyến với tên gọi là Roy Reading Website, nơi người dùng có thể truy cập và đọc sách từ một thư viện trực tuyến đa dạng. Mục tiêu của dự án là cung cấp một nền tảng thuận tiện cho người dùng để mọi người có thể trải nghiệm cũng như lĩnh hội tri thức từ sách bằng nhiều cách dễ dàng hơn, bất kể nơi nào và bất kể thời gian nào.

Sự cần thiết của dự án này trong ngữ cảnh hiện nay là không thể phủ nhận. Với cuộc sống ngày càng bận rộn và nhanh chóng, việc có một nền tảng trực truyến cho phép mọi người truy cập vào nguồn tài liệu đa dạng là rất quý báu. Dự án của chúng em đáp ứng như cầu này bằng cách cung cấp một giao diện trực tuyến dễ sử dụng và thư việc sách với nhiều thể loại, từ tiểu thuyết cho đến sách học thuật.

* 1. Đặt vấn đề.

Mục đích chính của dự án không chỉ dừng lại ở việc cung cấp dịch vụ đọc sách trực tuyến, mà còn cho phép người dùng tiếp cận sách bằng nhiều cách, qua việc kết hợp giữa dịch vụ đọc sách trực tuyến và audiobook (sách nói), hứa hẹn sẽ mang đến những trải nghiệm phong phú cho người dùng, giúp họ tiếp cận kiến thức và giải trí một cách thuận tiện và linh hoạt nhất có thể. Bên cạnh đó bằng việc khai thác vào khía cạnh sáng tạo và nhu cầu chia sẻ của người dùng, chúng em cũng đã cung cấp thêm các dịch vụ cho phép người dùng tự viết cuốn sách của mình và chia sẻ với mọi người thông qua nền tảng, tất nhiên vẫn sẽ dưới sự quản lý bản quyền cũng như phê duyệt của các quản trị viên.

* 1. Nêu ý nghĩa.

Dự án của chúng em có ý nghĩa quan trọng trong việc thúc đẩy việc đọc sách và lắng nghe audio. Đặc biệt, trong thời đại số hóa hiện nay, chúng em cung cấp một giải pháp hiện đại, linh hoạt và tiện lợi cho những người yêu thích sách và âm thanh. Dự án cũng hướng đến việc khám phá và khai thác tiềm năng của kho tàng tri thức trong mỗi tác phẩm.

Dự án Roy Reading Website, được chúng em gọi tắt là Roy Reading, không chỉ đơn thuần là một trang web đọc sách trực tuyến. Nó đại diện cho nỗ lực mang tri thức đến với mọi người một cách thuận tiện và linh hoạt. Dưới đây là một số khía cạnh quan trọng mà Roy Reading đem lại:

Trải nghiệm đọc sách đa dạng: Roy Reading không giới hạn một thể loại cụ thể. Thư viện của website chúng em đa dạng với các thể loại sách tiểu thuyết, sách giáo dục, lịch sử, khoa học, và nhiều thể loại khác. Người dùng có thể tìm thấy sách phù hợp với sở thích và nhu cầu cá nhân của họ.

Khả năng tiếp cận mọi lúc, mọi nơi: Cuộc sống hiện đại đặt ra nhiều thách thức về thời gian và vị trí. Roy Reading giải quyết vấn đề này bằng cách cho phép người dùng truy cập vào thư viện sách trực tuyến từ bất kỳ thiết bị nào có kết nối internet. Mọi người có thể đọc sách yêu thích của mình khi đang chờ trong hàng xếp hàng, trên xe buýt, hoặc thậm chí tại nhà.

Khám phá tri thức và giải trí: Roy Reading không chỉ dừng lại ở việc cung cấp sách để đọc. Chúng em kết hợp cả dịch vụ audiobook để mang đến trải nghiệm đọc sách đa dạng hơn. Mọi người có thể lắng nghe các cuốn sách khi họ không thể hoặc không muốn đọc trên giấy. Điều này mở ra một loạt các cách mới để tiếp cận tri thức và giải trí.

Không giới hạn sáng tạo: Chúng em luôn khuyến khích người dùng tham gia sáng tạo bằng cách viết cuốn sách của riêng họ. Mọi người đều có thể tự sáng tác và chia sẻ kiến thức, truyền đạt câu chuyện của mình đến với cộng đồng. Tất nhiên, các tác phẩm này sẽ được quản lý bản quyền và phê duyệt để đảm bảo tính chất lượng và đạo đức.

Với sự kết hợp của những khía cạnh này, Roy Reading không chỉ là một trang web đọc sách, mà còn là một cộng đồng kết nối tri thức và giải trí đa dạng, nhằm mang lại lợi ích và ý nghĩa sâu sắc cho mọi người trong cuộc sống hàng ngày.

* 1. Khảo sát các công trình liên quan.

Trước khi bắt đầu triển khai dự án, nhóm chúng em đã quan sát và phân tích về một vài trang web đọc sách có liên quan. Tiêu biểu nhất trong số đó là hai trang web lớn khá phổ biến là Wattpad – Trang web đọc sách kết hợp cùng ứng dụng mobile và Truyện FULL – Trang web có nguồn sách phong phú ở cả hai loại truyện chữ và truyện tranh.

Dưới đây là một số phân tích về chức năng và giao diện của cả hai trang:

Wattpad:

Chức năng: Wattpad nổi tiếng với khả năng cho phép người dùng tự viết và đăng tải truyện của họ. Nền tảng này có một cộng đồng lớn các tác giả và độc giả, cho phép tương tác qua việc bình luận, đánh giá truyện và chia sẻ. Ngoài ra, Wattpad còn cung cấp dịch vụ audiobook (cho phép chèn video Youtube vào ngay trong chương truyện, sách) và có hỗ trợ trên cả ứng dụng di động giúp việc đọc sách dễ dàng hơn.

A screenshot of a computer

Description automatically generated

Hình 1. Phần giao diện chức năng chèn các mục đa phương tiện vào phần đăng truyện của Wattpad.

Giao diện: Trang web có giao diện trực quan và thân thiện với người dùng. Người dùng có thể dễ dàng tìm kiếm truyện theo thể loại, tác giả, hoặc từ khóa. Trang cá nhân của người dùng hiển thị thông tin cá nhân và truyện họ đã viết. Giao diện ứng dụng di động Wattpad cũng tương tự và cung cấp trải nghiệm đọc sách trên di động một cách tiện lợi.

Truyện FULL:

Chức năng: Truyện FULL là một trang web chuyên về việc cung cấp nguồn sách phong phú, bao gồm cả truyện chữ và truyện tranh (được chuyển hướng đến trang web khác là NetTruyenFull). Người dùng có thể tìm kiếm sách theo các thể loại, tác giả, hoặc từ khóa. Trang web này không yêu cầu người dùng đăng ký để đọc, khiến cho việc trải nghiệm của người dùng tuy dễ dàng hơn vì không cần đăng ký tài khoản nhưng sẽ có bất tiện trong việc lưu trữ, quản lý danh sách đọc cũng như không thể đăng tải chia sẻ tác phẩm của bản thân.

Giao diện: Giao diện của Truyện FULL tương đối đơn giản và dễ sử dụng. Trang web chú trọng vào việc hiển thị danh sách các cuốn sách và cho phép người dùng chọn để đọc trực tiếp. Không có yêu cầu đăng nhập để truy cập sách, điều này tạo thuận lợi cho người đọc.

Cả hai trang web đều có các yếu điểm và điểm mạnh riêng, nhưng điểm chung của họ là cung cấp nền tảng để người đọc truy cập và tương tác với sách trực tuyến một cách thuận tiện. Dự án Roy Reading của nhóm chúng em đang hướng đến việc kết hợp những ưu điểm này và mong muốn có thể mang đến cho người dùng những trải nghiệm đọc sách đa dạng và sáng tạo.

Chương 2: Lý thuyết cơ sở

* 1. Giới thiệu về Spring Boot
     1. Khái niệm

Spring Boot là một framework phát triển ứng dụng Java dựa trên Spring Framework, nhằm tạo ra các ứng dụng Java một cách nhanh chóng, dễ dàng và hiệu quả. Điểm đặc biệt của Spring Boot là khả năng tự cấu hình mà nó mang lại, giúp giảm thiểu sự phức tạp trong việc cấu hình và triển khai ứng dụng.

* + 1. Lý do lựa chọn

Spring Boot cung cấp cho người dùng một môi trường phát triển đáng tin cậy cho việc xây dựng các ứng dụng web Java dựa trên nguyên tắc “Convention over configuration” [1]– Qui ước về cấu hình. Quy ước này khiến việc cấu hình trở nên đơn giản hơn vì chỉ cần chỉ định các ngoại lệ, từ đó tăng năng suất phát triển, giúp người dùng tập trung hơn vào việc xây dựng các tính năng chính của dự án.

Ngoài ra, Spring Boot được chọn vì tích hợp tốt với hệ thống quản lý cơ sở dữ liệu MySQL, cho phép người dùng tạo ra một hệ thống mạnh mẽ để quản lý và truy xuất dữ liệu dùng một cách hiệu quả. Việc sử dụng Spring Boot cũng giúp chúng em xây dựng các APIs RESTful dễ dàng, đáp ứng yêu cầu của ứng dụng React frontend và các ứng dụng di động trong tương lai.

Trong quá trình phát triển dự án, việc sử dụng Spring Boot để xây dựng RESTful APIs đã mang lại nhiều lợi ích quan trọng. Spring Boot là một framework mạnh mẽ cho phát triển ứng dụng web, đặc biệt là trong việc tạo và quản lý các API endpoints.

Sử dụng Spring Boot, chúng em dễ dàng xác định các API endpoints thông qua các annotation đơn giản như *@RestController* và *@RequestMapping*. Điều này giúp chúng em nhanh chóng tạo ra các dịch vụ cho phép truy cập và quản lý dữ liệu.

Ngoài ra, tích hợp với Spring Data JPA, chúng em có thể thực hiện các thao tác CRUD với cơ sở dữ liệu một cách hiệu quả. Spring Boot cũng cung cấp tích hợp với Spring Security để bảo vệ các API khỏi các cuộc tấn công bên ngoài.

* + 1. Kiến trúc và thành phần

Để bắt đầu dự án Roy Reading Website, nhóm chúng em đã sử dụng Spring Initializr, một công cụ hữu ích để tạo ra dự án Spring Boot với các cấu hình cơ bản. Chúng em đã chọn các dependencies cần thiết như Spring Web, Spring Data JPA, Spring Security, sau đó tạo một dự án Maven dựa trên cấu trúc dự án tương tự như sau:

A screenshot of a computer program

Description automatically generated

Hình 2. a Cấu trúc thư mục dự án

A screenshot of a computer program

Description automatically generated

Hình 2. 1 Cấu trúc thư mục dự án

Thư mục *src/main/java/com.example.apiroy:* Đây là nơi chúng em đặt mã nguồn Java chính của dự án. Thư mục này được chia thành các phần chính sau:

Thư mục Configs: Chứa các tệp cấu hình, trong đó có SecurityConfig để thiết lập bảo mật cho ứng dụng.

Thư mục Pojo: Đây là nơi chúng tôi định nghĩa các lớp POJO (Plain Old Java Objects) như Book, User, AudioFile, Chapter, ViewHistory, Genre, AuthRequest và Comment. Các lớp này biểu diễn các đối tượng dữ liệu trong ứng dụng.

Thư mục Controller: Chứa các lớp controller như BookController, UserController... để xử lý các yêu cầu từ phía người dùng và gửi lại phản hồi.

Thư mục Service: Chứa các giao diện service như BookService, UserService... để quản lý logic của ứng dụng.

Thư mục Impl: Chứa các implementation của các giao diện service như BookServiceImpl, UserServiceImpl... Đây này là nơi chúng em triển khai logic cụ thể.

Thư mục Repository: Chứa các giao diện repository như BookRepository…, cho phép tương tác với cơ sở dữ liệu.

Ngoài các thư mục chính, cấu trúc thư mục của dự án này còn có một vài thư mục con chứ các thành phần không kém phần quan trọng như Filters – dùng để chứa thành phần trong việc xử lý xác thực, hoặc folder Enum chứa các lớp enum quản lý các giá trị hằng số của các đối tượng, giúp cho mã nguồn dễ đọc và dễ hiểu hơn.

* 1. Giới thiệu về ReactJS
     1. Khái niệm

React là một thư viện JavaScript phát triển bởi Facebook, được sử dụng rộng rãi trong phần giao diện người dùng. React giúp xây dựng giao diện người dùng động và tương tác một cách hiệu quả.

* + 1. Lý do lựa chọn

Nhóm chúng em đã chọn React vì tính linh hoạt và khả năng tạo giao diện người dùng tương tác dễ dàng. React cho phép chúng em chia thành phần giao diện thành các phần riêng lẻ, giúp quản lý mã nguồn dễ dàng và tái sử dụng chúng trên nhiều trang và tính năng của trang web. Khả năng tương tác với các APIs từ phía máy chủ cũng giúp chúng em tải dữ liệu một cách linh hoạt và hiệu quả hơn.

* 1. Giới thiệu về MySQL
     1. Khái niệm

MySQL là một hệ quản trị cơ sở dữ liệu quan hệ mã nguồn mở phổ biến, được sử dụng để lưu trữ và quản lý dữ liệu. Được biết đến như một hệ quản trị cơ sở dữ liệu (Relational Database Management System: *RDBMS*) mã nguồn mở phát triển và duy trì bởi Oracle Corporation. Nó là một phần quan trọng của ngăn xếp công nghệ LAMP (Linux, Apache, MySQL, PHP/Python/Perl) và là một giải pháp cơ sở dữ liệu phổ biến trong cộng đồng phát triển phần mềm mã nguồn mở.

MySQL hoạt động dựa trên mô hình Client-Server, điều này mang lại một số lợi ích quan trọng:

*Phân tách và Độc lập:* Mô hình Client-Server tách riêng phần giải quyết dữ liệu (Server) và phần tương tác với người dùng (Client). Điều này giúp giữ cho mã nguồn và dữ liệu ở phía Server được quản lý một cách riêng lẻ và độc lập với phía Client. Điều này giúp quản lý, bảo trì, và cải thiện hiệu suất dễ dàng hơn.

*Đa Người Dùng:* Mô hình Client-Server cho phép nhiều người dùng cùng truy cập và làm việc với cơ sở dữ liệu MySQL cùng một lúc. Server MySQL có khả năng xử lý nhiều kết nối từ các Client khác nhau, cho phép nhiều người dùng thực hiện các tác vụ cơ sở dữ liệu song song mà không gây xung đột.

Bảo Mật: MySQL cung cấp kiến thức bảo mật tích hợp, cho phép quản lý quyền truy cập và xác thực dựa trên người dùng và máy chủ. Mô hình Client-Server cho phép kiểm soát quyền truy cập và đảm bảo rằng chỉ những người dùng được phép có thể tương tác với dữ liệu.

Khả Năng Mở Rộng: Mô hình Client-Server dựa trên kiến trúc mạng, giúp dễ dàng mở rộng hệ thống. Bạn có thể thêm nhiều máy chủ MySQL vào mô hình Server nếu cần thiết, để xử lý khối lượng dữ liệu lớn hơn hoặc tăng cường khả năng chịu tải.

Hiệu Suất: Mô hình Client-Server cho phép máy chủ MySQL chuyên biệt tập trung vào xử lý và quản lý dữ liệu, cung cấp hiệu suất cao hơn. Các Client có thể làm việc trên nhiều nền tảng khác nhau và thậm chí có thể cài đặt các ứng dụng máy khách để thực hiện các thao tác cơ sở dữ liệu một cách hiệu quả.

* + 1. Lý do lựa chọn

Việc chúng em lựa chọn MySQL hoàn toàn dựa trên những ưu điểm đáng kể của nó:

Dễ sử dụng: MySQL được lựa chọn vì tính dễ sử dụng của nó. Giao diện quản trị thân thiện và khả năng tương thích trên nhiều hệ điều hành giúp đơn giản hóa việc quản lý dữ liệu.

Độ bảo mật cao: Bảo vệ dữ liệu là một ưu tiên hàng đầu trong dự án của chúng em. MySQL cung cấp một loạt các tính năng bảo mật mạnh mẽ, bao gồm quản lý quyền truy cập và mã hóa dữ liệu, giúp bảo đảm an toàn cho thông tin quan trọng của người dùng.

Đa tính năng: MySQL là một hệ quản trị cơ sở dữ liệu quan hệ đa tính năng. Điều này cho phép chúng em làm việc với dữ liệu một cách linh hoạt và hiệu quả. Chúng em có khả năng sử dụng các truy vấn SQL phức tạp để truy xuất và quản lý dữ liệu dễ dàng.

Khả năng mở rộng và mạnh mẽ: Dự án Roy Reading Website của chúng em không ngừng phát triển. MySQL cho phép chúng em mở rộng cơ sở dữ liệu một cách linh hoạt, đảm bảo rằng chúng tôi có thể xử lý lượng dữ liệu ngày càng tăng mà không gặp vấn đề.

Hiệu suất tốt: MySQL tuân theo các tiêu chuẩn hiệu suất nghiêm ngặt, giúp ứng dụng của chúng em hoạt động nhanh chóng và đáng tin cậy. Điều này đặc biệt quan trọng khi người dùng truy cập vào dịch vụ đọc sách và audiobook của chúng em.

Sự kết hợp của những ưu điểm này làm cho MySQL trở thành một phần quan trọng của cơ sở hạ tầng dữ liệu của dự án Roy Reading Website, giúp đảm bảo rằng người dùng có những trải nghiệm thú vị và bảo mật khi sử dụng dịch vụ của nhóm chúng em.

Chương 3: Phân tích, thiết kế, thực hiện hệ thống.

* 1. Phân tích hệ thống.
     1. Giới thiệu về hệ thống

Hệ thống website đọc sách của chúng em với tên gọi "Roy Reading", là một nền tảng đa dạng và tiện lợi cho việc đọc sách trực tuyến. Với mục tiêu tạo ra một môi trường giúp mọi người trải nghiệm tri thức và giải trí một cách linh hoạt, dễ dàng và đa dạng, Roy Reading cung cấp một thư viện sách đa thể loại và một loạt các tính năng hấp dẫn.

Người dùng sẽ tìm thấy trên trang chủ những cuốn sách vừa mới cập nhật, cùng với danh sách sách yêu thích của họ. Tính năng tìm kiếm cho phép người dùng dễ dàng tìm sách theo tên sách, thể loại, tác giả, hoặc người đăng. Ngoài ra, Roy Reading còn cho phép người dùng viết và đọc sách của riêng họ, cùng với việc thực hiện bình luận ở mỗi chương hoặc file audio của sách.

Hệ thống cũng đặc biệt hữu ích cho những người muốn tạo và chia sẻ tác phẩm của riêng họ. Hệ thống cung cấp một giao diện dễ sử dụng cho việc đăng sách, và sách sẽ được duyệt bởi quản trị viên trước khi xuất bản, đảm bảo chất lượng và tính bản quyền của các tác phẩm. Roy Reading đánh dấu một bước tiến quan trọng trong việc thúc đẩy đọc sách và chia sẻ tri thức trong thời đại số hóa ngày nay.

* + 1. Phân tích yêu cầu dự án
       1. A diagram of a network

          Description automatically generatedLược đồ use case

Hình 3. Tổng quan lược đồ use case

* + - 1. Đặc tả use case

⬩ Use Case 1: Tìm kiếm sách

Mô tả: Người dùng muốn tìm kiếm sách dựa trên thể loại của chúng.

Actor chính: Người dùng.

Luồng sự kiện chính:

Người dùng truy cập tính năng tìm kiếm.

Người dùng chọn thể loại cụ thể hoặc nhập từ khóa tìm kiếm.

Hệ thống hiển thị danh sách sách dựa trên thể loại hoặc từ khóa tìm kiếm.

⬩ Use Case 2: Đọc sách online

Mô tả: Người dùng muốn đọc một cuốn sách trực tuyến.

Actor chính: Người dùng.

Luồng sự kiện chính:

Người dùng tìm kiếm hoặc chọn một cuốn sách.

Người dùng chọn chương hoặc trang để đọc.

Hệ thống hiển thị nội dung của sách hoặc chương tương ứng.

⬩ Use Case 3: Viết sách mới

Mô tả: Người dùng muốn viết và đăng một cuốn sách mới.

Actor chính: Người dùng.

Luồng sự kiện chính:

Người dùng đăng nhập vào tài khoản.

Người dùng truy cập tính năng sáng tác sách.

Người dùng viết nội dung của cuốn sách.

Người dùng đăng cuốn sách và chờ quản trị viên duyệt.

⬩ Use Case 4: Duyệt sách

Mô tả: Quản trị viên muốn duyệt và xác nhận cuốn sách mà người dùng đã đăng lên trang web.

Actor chính: Quản trị viên.

Luồng sự kiện chính:

Quản trị viên đăng nhập vào tài khoản quản trị.

Quản trị viên truy cập tính năng "Duyệt sách."

Hệ thống hiển thị danh sách các cuốn sách mà người dùng đã đăng lên chờ duyệt.

Quản trị viên chọn một cuốn sách để duyệt.

Quản trị viên xem nội dung của cuốn sách và kiểm tra tính phù hợp với quy định của trang web.

Quản trị viên đưa ra quyết định duyệt hoặc từ chối cuốn sách.

Hệ thống cập nhật trạng thái của cuốn sách trong cơ sở dữ liệu thành "Đã được duyệt" hoặc "Bị từ chối" tùy thuộc vào quyết định của quản trị viên.

Luồng sự kiện thay thế: Nếu quản trị viên từ chối cuốn sách, hệ thống gửi thông báo cho người dùng về quyết định.

⬩ Use Case 5: Bình luận

Mô tả: Người dùng muốn bình luận về một chương trong cuốn sách.

Actor chính: Người dùng.

Luồng sự kiện chính:

Người dùng đăng nhập vào tài khoản của họ.

Người dùng truy cập cuốn sách và chọn một chương.

Người dùng thêm bình luận và gửi.

Hệ thống hiển thị bình luận trên chương tương ứng.

⬩ Use Case 6: Yêu thích sách

Mô tả: Người dùng muốn thêm một cuốn sách vào “Danh sách yêu thích" của họ để dễ dàng theo dõi và quay lại sau này.

Actor chính: Người dùng.

Luồng sự kiện chính:

Người dùng đăng nhập vào tài khoản của họ.

Người dùng truy cập trang thông tin chi tiết của một cuốn sách.

Hệ thống hiển thị một tùy chọn "Yêu Thích" hoặc biểu tượng trái tim mà người dùng có thể nhấp vào.

Người dùng nhấp vào "Yêu Thích" hoặc biểu tượng trái tim để thêm cuốn sách vào “Danh sách yêu thích" của họ.

Hệ thống cập nhật danh sách "Yêu Thích" trong tài khoản của người dùng.

⬩ Use Case 7: Xem Lịch Sử Đọc

Mô tả: Người dùng muốn xem danh sách các cuốn sách mà họ đã đọc trước đó.

Actor chính: Người dùng.

Điều kiện tiên quyết: Người dùng đã đăng nhập vào tài khoản.

Luồng sự kiện chính:

Người dùng đăng nhập vào tài khoản của họ.

Người dùng truy cập tính năng "Danh sách đã đọc."

Hệ thống hiển thị danh sách các cuốn sách mà người dùng đã đọc trước đó, sắp xếp theo thứ tự thời gian đọc gần nhất.

Người dùng có thể nhấp vào một cuốn sách để xem nội dung hoặc thông tin chi tiết.

⬩ Use Case 8: Thống Kê

Mô tả: Quản trị viên muốn xem thống kê dữ liệu liên quan đến hoạt động của người dùng và cuốn sách trên trang web để hiểu sự phát triển và sử dụng của nền tảng.

Actor chính: Quản trị viên

Điều kiện tiên quyết: Quản trị viên đã đăng nhập vào tài khoản quản trị viên.

Luồng sự kiện chính:

Quản trị viên đăng nhập vào tài khoản quản trị viên.

Quản trị viên truy cập tính năng "Thống Kê."

Hệ thống hiển thị các tùy chọn thống kê, bao gồm thống kê người dùng, thống kê cuốn sách, thống kê hoạt động trên trang web, và thống kê báo cáo từ người dùng.

Quản trị viên chọn một trong các tùy chọn thống kê.

Hệ thống hiển thị dữ liệu thống kê chi tiết dưới dạng biểu đồ, báo cáo số liệu hoặc danh sách.

Quản trị viên có thể lọc và sắp xếp dữ liệu thống kê theo nhiều tiêu chí khác nhau.

Quản trị viên có thể xuất báo cáo thống kê hoặc lưu chúng cho việc tham khảo sau này.

⬩ Use Case 9: Đăng nhập

Mô tả: Người dùng đã đăng ký nhập thông tin đăng nhập để truy cập vào tài khoản cá nhân.

Actor chính: Người dùng.

Điều kiện tiên quyết: Người dùng đã đăng ký tài khoản.

Luồng sự kiện chính:

Người dùng truy cập trang đăng nhập.

Người dùng nhập tên đăng nhập và mật khẩu.

Hệ thống kiểm tra thông tin đăng nhập, xác thực người dùng.

Nếu thông tin đúng, hệ thống cho phép người dùng truy cập vào tài khoản.

Nếu thông tin không đúng hoặc tài khoản bị khoá, hệ thống thông báo lỗi.

⬩ Use Case 10: Đăng ký

Mô tả: Người dùng chưa có tài khoản tạo mới tài khoản trên hệ thống.

Actor chính: Người dùng.

Luồng sự kiện chính:

Người dùng truy cập trang đăng ký.

Người dùng điền thông tin cá nhân: tên, email, mật khẩu, v.v.

Hệ thống kiểm tra tính hợp lệ của thông tin và lưu tài khoản mới.

Tài khoản được tạo thành công.

* 1. Thiết kế hệ thống.
     1. Sơ đồ Activity Diagram của các chức năng

Sơ đồ Activity Diagram Đăng ký

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generatedSơ đồ Activity Diagram Đăng nhập

Sơ đồ Activity Diagram Tìm kiếm sách

Sơ đồ Activity Diagram Đọc sách online

Sơ đồ Activity Diagram Viết sách mới

Sơ đồ Activity Diagram Duyệt sách

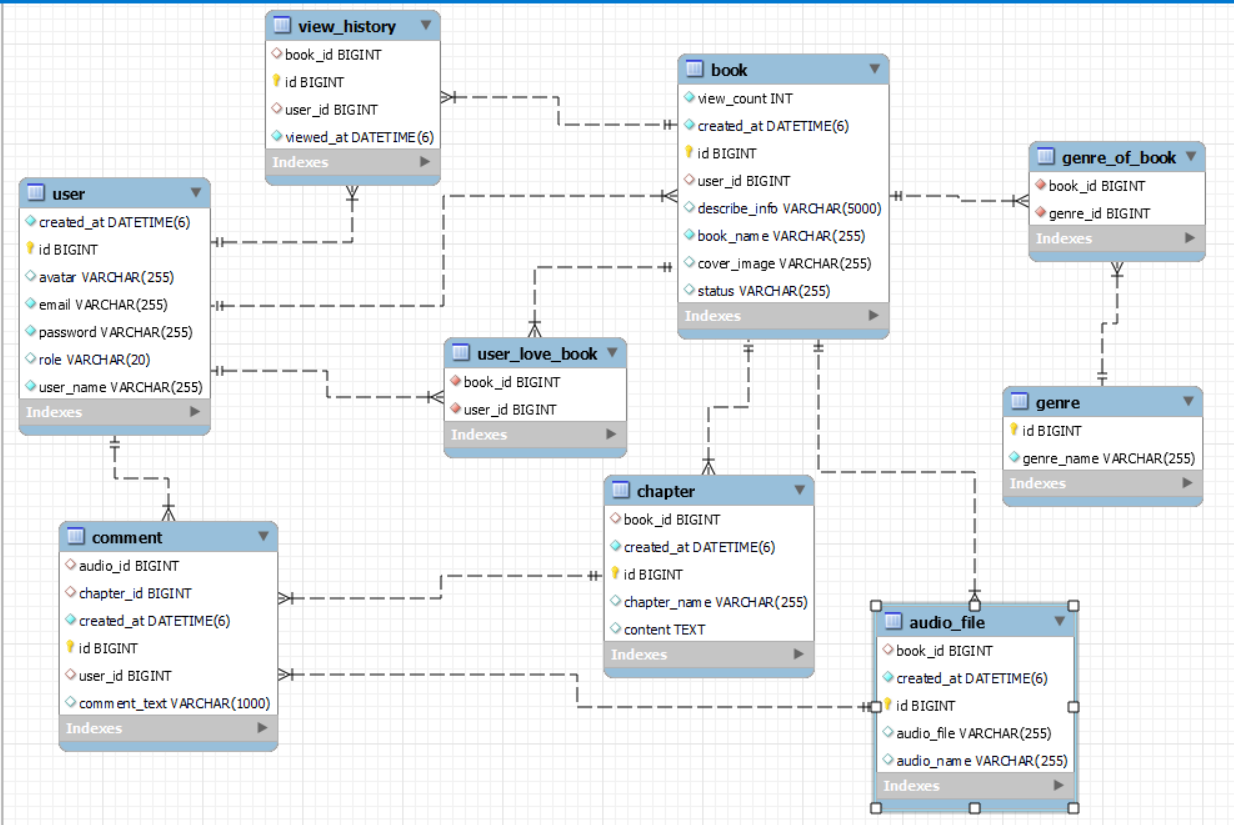
Sơ đồ Activity Diagram Bình luận

Sơ đồ Activity Diagram Yêu thích sách

Sơ đồ Activity Diagram Xem lịch sử đọc

Sơ đồ Activity Diagram Thống kê

* + 1. Lược đồ cơ sở dữ liệu quan hệ



* + 1. A computer screen shot of a computer

       Description automatically generatedSơ đồ lớp

Phân tích mối quan hệ giữa các lớp:

⬩ Genre và Book:

Lớp Genre và Book có mối quan hệ association trọng số nhiều - nhiều (Many-to-Many). Mỗi thể loại sách (Genre) có thể thuộc nhiều cuốn sách, và mỗi cuốn sách cũng có thể thuộc nhiều thể loại sách.

⬩ AudioFile và Book:

Lớp AudioFile và Book có mối quan hệ composition trọng số nhiều – một (Many-to-One). Mỗi tập tin âm thanh (AudioFile) được liên kết với một cuốn sách cụ thể.

⬩ Chapter và Book:

Lớp Chapter và Book cũng có mối quan hệ composition trọng số nhiều – một (Many-to-One). Mỗi chương của một cuốn sách được liên kết với cuốn sách cụ thể đó.

⬩ Comment và Chapter, AudioFile:

Lớp Comment có mối quan hệ nhiều – một (Many-to-One) với cả Chapter và AudioFile. Mỗi bình luận được liên kết với một chương hoặc một tập tin âm thanh cụ thể. Điều này cho phép người dùng bình luận trên các phần khác nhau của sách hoặc tập tin âm thanh.

⬩ User và Book:

Lớp User và Book có mối quan hệ association trọng số nhiều – nhiều (Many – to – Many). Mỗi người dùng có thể xem/yêu thích nhiều cuốn sách, và mỗi cuốn sách có thể được xem/yêu thích bởi nhiều người dùng. Ngoài ra trong chức năng đăng sách, User và Book lại có mối quan hệ aggregation trọng số một – nhiều (One – to – Many) bởi một người dùng có thể đăng nhiều cuốn sách, nhưng mỗi cuốn sách chỉ được đăng bởi một người dùng.

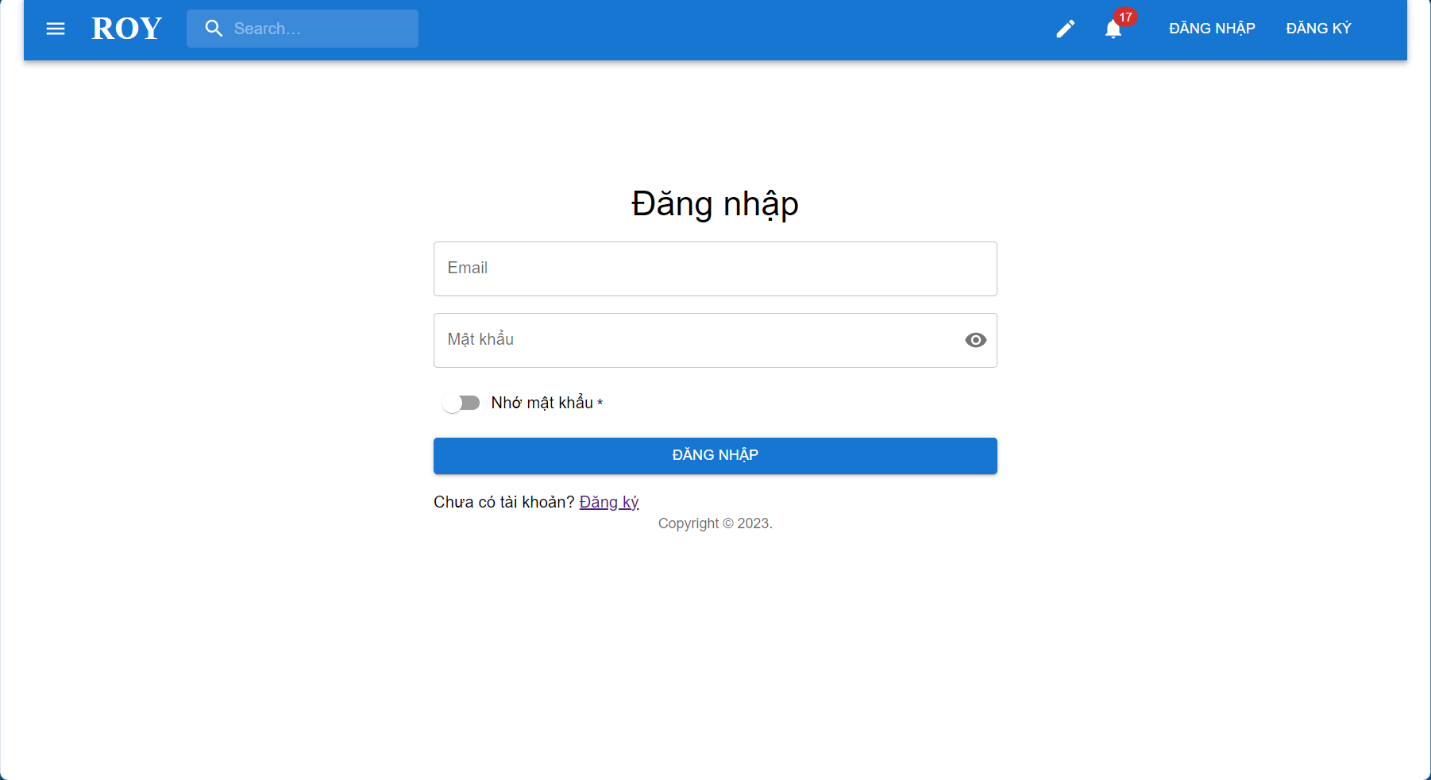
⬩ User và ViewHistory:

Lớp User và ViewHistory có mối quan hệ một – nhiều (One – to – Many). Mỗi người dùng có thể có nhiều lịch sử xem sách (ViewHistory), một lịch sử xem chỉ thuộc một người dùng.

* + 1. Giao diện và thiết kế xử lý

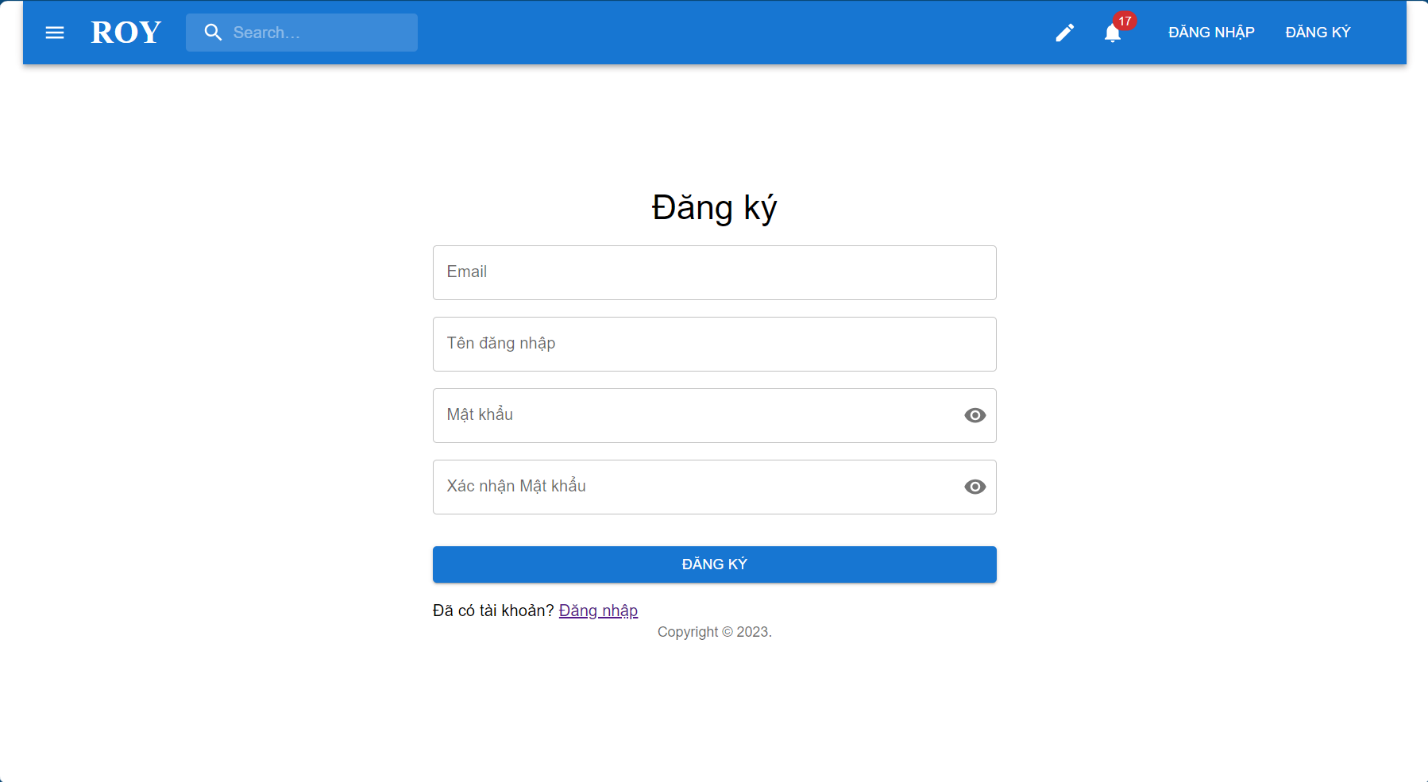
Giao diện đăng nhập

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Tên sự kiện | Điều kiện gọi | Ý nghĩa |
| 1 | BtnLogIn\_Click | Khi người dùng nhấp chuột vào nút đăng nhập. | Thực hiện chức năng xác thực tài khoản và đăng nhập. |
| 2 | BtnRegister\_Click | Khi người dùng nhấp chuột vào nút đăng ký. | Thực hiện chức năng đăng ký tài khoản cho người dùng. |

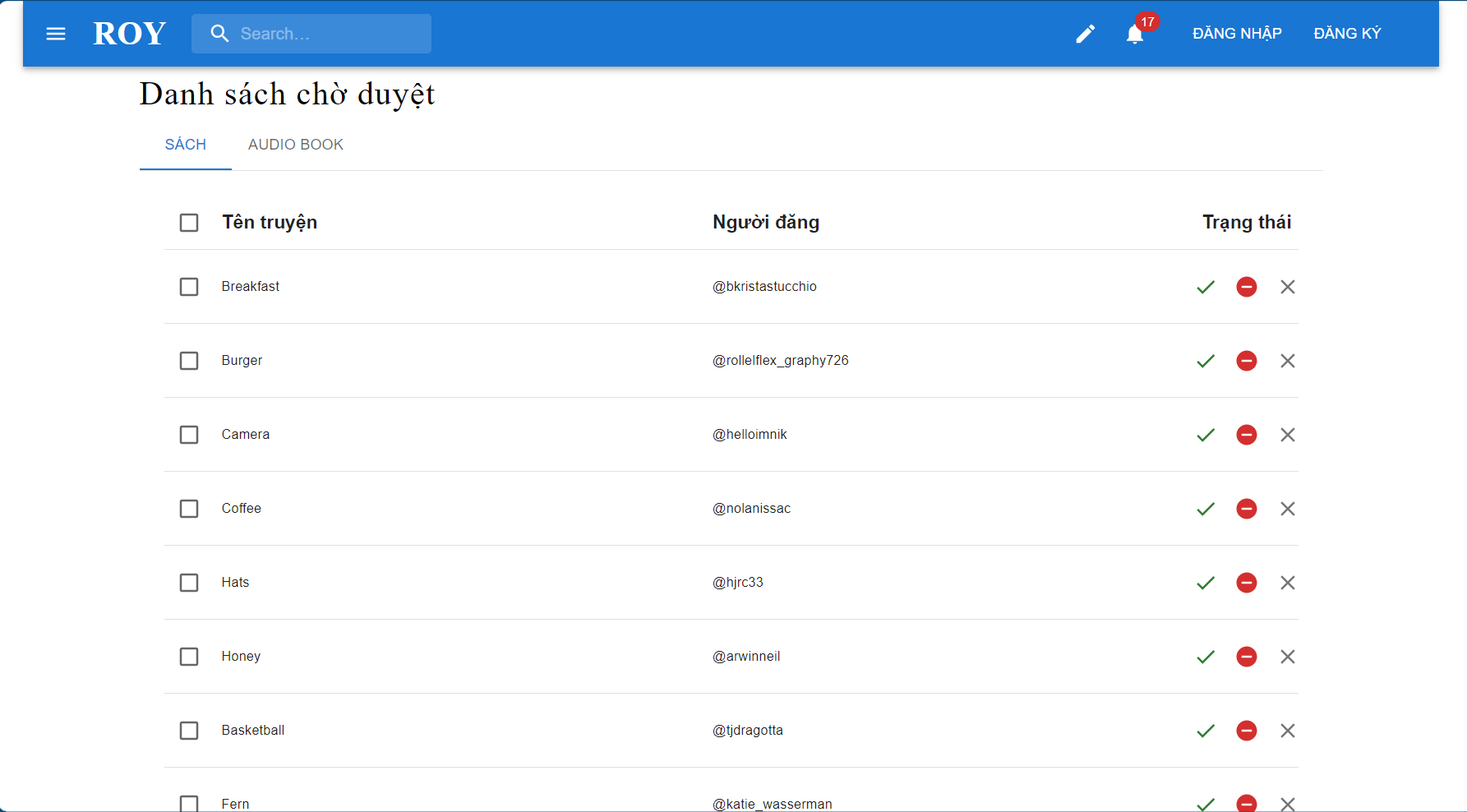


Giao diện đăng ký

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Tên sự kiện | Điều kiện gọi | Ý nghĩa |
| 1 | BtnLogIn\_Click | Khi người dùng nhấp chuột vào nút đăng nhập. | Thực hiện chức năng xác thực tài khoản và đăng nhập. |
| 2 | BtnRegister\_Click | Khi người dùng nhấp chuột vào nút đăng ký. | Thực hiện chức năng đăng ký tài khoản cho người dùng. |

Giao diện duyệt sách của Quản trị viên

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Tên sự kiện | Điều kiện gọi | Ý nghĩa |
| 1 | BtnReject\_Click | Khi người dùng nhấp chuột vào nút reject | Thực hiện chức năng từ chối duyệt đăng sách. |
| 2 | BtnApprove\_Click | Khi người dùng nhấp chuột vào nút dấu check ✓ | Thực hiện chức năng đồng ý duyệt đăng sách |
| 3 | BtnRemove\_Click | Khi người dùng nhấp chuột vào nút dấu x | Thực hiện chức năng loại bỏ yêu cầu duyệt. |



Chương 4: Thử nghiệm, đánh giá kết quả và phân tích.

* 1. Thử nghiệm
  2. Đánh giá và phân tích kết quả.

Chương 5: Kết luận và hướng phát triển.

* 1. Kết luận.
  2. Hướng phát triển.

Tài liệu tham khảo.