**TRƯỜNG ĐẠI HỌC PHENIKAA  
KHOA CÔNG NGHỆ THÔNG TIN**

**A logo with blue and orange circle

Description automatically generated**------**🙠🕮🙢**-----

**BÀI TẬP LỚN**

**KĨ THUẬT PHẦN MỀM**

**Tên đề tài: Quản lý sách trong một thư viện**

**Giáo viên hướng dẫn :** **Vũ Quang Dũng**

**Lớp : CSE702025-1-3-24 ( nhóm 5 )**

**Sinh viên thực hiện:**

|  |  |  |
| --- | --- | --- |
| Mã sinh viên | Họ và tên | Điểm |
| 2301044 | Trần Văn Dương |  |
| 23010220 | Trần Phương Nam |  |
| 23010211 | Trần Đình Danh |  |

**Hà Nội – 2025**

**LỜI MỞ ĐẦU**

Trên thế giới cũng như ở Việt Nam, công nghệ thông tin đã trở thành một

công nghệ mũi nhọn, nó là ngành khoa học kỹ thuật không thể thiếu trong việc

áp dụng vào các hoạt động xã hội như: quản lý hệ thống thư viện sách, kinh tế,

thông tin, kinh doanh, và mọi hoạt động trong lĩnh vực nghiên cứu...

Ở nước ta hiện nay, việc áp dụng vi tính hóa trong quản lý tại các cơ quan,

xí nghiệp, tổ chức đang rất phổ biến và trở nên cấp thiết, bởi ngành nghề nào

cũng đòi hỏi con người phải xử lý khối lượng công việc khổng lồ, và những kiến

thức, những suy nghĩ, những đào tạo chuyên sâu. Một vấn đề cấp thiết đặt ra

trong quản lý là làm thế nào để chuẩn hóa cách xử lý dữ liệu ở trường học.

Mặc dù đã rất cố gắng để hoàn thành công việc, nhưng do thời gian có hạn

và thiếu kinh nghiệm cũng như kỹ năng chưa cao nên việc phân tích và thiết kế

hệ thống quản lý còn nhiều thiếu sót, kính mong quý thầy cô góp ý,

bổ sung để chúng em hoàn thiện cho bài tập tốt hơn nữa.

Chúng em xin chân thành cảm ơn!

Sinh viên thực hiện !

**Mục lục**

LỜI MỞ ĐẦU

Chương 1. Phân tích thiết kế thư viện sách

1.1 Giới thiệu đề tài

1.2. Mục tiêu

1.3 Phạm vi sử dụng

Chương 2: Cài đặt

2.1 Giới thiệu công cụ

2.2 tài liệu đặc tả yêu cầu người dùng

2.3 Phân tích hệ thống

2.4 Thiết kế hệ thống

Chương 3: Thiết kế giao diện

3.1 Thiết kế hệ thống

Chương 4: Triển khai phần mềm

Chương 5 : Kết luận

5.1 Kết luận

5.2 Đánh giá kết quả đạt được

5.3 Hướng phát triển trong tương lai

Tài liệu tham khảo

**Chương 1: Giới thiệu về đề tài quản lý thư viện sách**

**1. Giới thiệu đề tài**

**1.1 Tên đề tài:**

Library Management System (LMS) – Phần mềm quản lý thư viện sách

Lý do chọn đề tài:

Trong thời đại công nghệ số hiện nay, nhu cầu quản lý thư viện sách theo cách truyền thống (sổ ghi tay, thẻ mượn trả) không còn đáp ứng được sự phát triển nhanh chóng của các thư viện hiện đại. Với lượng đầu sách ngày càng lớn, người dùng ngày càng đa dạng (sinh viên, giảng viên, nhân viên...), thư viện cần một hệ thống giúp quản lý thông tin sách, người mượn và quá trình mượn trả một cách hiệu quả, chính xác và tiết kiệm thời gian.

Bên cạnh đó, nhiều thư viện ở các trường học, trung tâm giáo dục hoặc khu dân cư vẫn chưa được trang bị hệ thống phần mềm chuyên dụng. Điều này dẫn đến tình trạng thất lạc sách, thống kê không đầy đủ, hoặc việc tra cứu mất nhiều thời gian và công sức.

Chính vì những lý do trên, nhóm quyết định chọn đề tài “Library Management System (LMS)” nhằm xây dựng một phần mềm có giao diện thân thiện, dễ sử dụng và phục vụ tốt nhu cầu quản lý thư viện sách cho các đơn vị giáo dục và cộng đồng.

**1.2. Mục tiêu**

Mục tiêu của hệ thống LMS là tạo ra một công cụ phần mềm hiệu quả, hỗ trợ công tác quản lý thư viện sách một cách khoa học và hiện đại. Cụ thể:

Quản lý thông tin sách trong thư viện: tên sách, tác giả, nhà xuất bản, thể loại, năm xuất bản, vị trí lưu trữ...

Hỗ trợ thủ thư và người dùng trong việc tra cứu nhanh sách theo từ khóa hoặc danh mục.

Quản lý quá trình mượn – trả sách: theo dõi người mượn, ngày mượn, ngày trả, tình trạng quá hạn.

Hạn chế tối đa tình trạng mất sách hoặc trả chậm nhờ hệ thống nhắc nhở tự động.

Cung cấp báo cáo thống kê định kỳ như: số sách đang có, sách đang được mượn, sách mượn nhiều nhất...

Tăng tính chuyên nghiệp và hiện đại trong công tác thư viện.

**1.3. Phạm vi sử dụng**

Phần mềm LMS được thiết kế phù hợp với quy mô thư viện vừa và nhỏ, áp dụng chủ yếu cho các trường học, trung tâm đào tạo, hoặc thư viện cộng đồng. Các chức năng chính bao gồm:

Quản lý sách: Thêm mới, chỉnh sửa, xóa sách; tìm kiếm sách theo tên, tác giả, thể loại...

Quản lý người dùng (thủ thư và người mượn): Lưu thông tin người mượn, tạo tài khoản người dùng.

Quản lý mượn – trả: Cập nhật lịch sử mượn/trả sách, trạng thái sách (đang mượn / có sẵn).

Hệ thống nhắc nhở: Gửi thông báo khi đến hạn trả sách hoặc sách quá hạn.

Thống kê – báo cáo: Tổng hợp các báo cáo như sách mượn nhiều nhất, sách chưa trả, lịch sử mượn...

Đăng nhập bảo mật: Người dùng và thủ thư đăng nhập vào hệ thống bằng tài khoản được cấp.

Hệ thống hiện hỗ trợ một quản trị viên thư viện và nhiều người mượn, chưa hỗ trợ phân quyền nâng cao.

**1.4. Công nghệ hỗ trợ**

+ Apache NetBeans

Apache NetBeans là một môi trường phát triển tích hợp (IDE) mã nguồn mở, chủ yếu dùng để lập trình Java, đồng thời hỗ trợ các ngôn ngữ khác như PHP, C/C++, HTML, JavaScript. Được phát triển bởi Apache Software Foundation, NetBeans có giao diện thân thiện, dễ sử dụng, phù hợp cho cả sinh viên và lập trình viên chuyên nghiệp.

NetBeans tích hợp nhiều công cụ tiện ích như trình biên dịch, trình gỡ lỗi, thiết kế giao diện đồ họa (GUI Builder), hỗ trợ Maven, Gradle và Git. IDE này cũng hỗ trợ phát triển ứng dụng web và desktop với Java EE, JavaFX. Với hệ thống plugin mở rộng và cộng đồng hỗ trợ mạnh mẽ, Apache NetBeans là lựa chọn phổ biến trong giáo dục và phát triển phần mềm Java.

+ XAMPP

XAMPP là một phần mềm mã nguồn mở dùng để xây dựng môi trường máy chủ web cục bộ. Tên gọi XAMPP là viết tắt của X (cross-platform), Apache, MySQL, PHP, và Perl. Phần mềm này giúp người dùng dễ dàng cài đặt và chạy máy chủ web trên các hệ điều hành như Windows, Linux, và macOS.

XAMPP được sử dụng phổ biến trong phát triển web, đặc biệt là với các ứng dụng sử dụng PHP và MySQL. Ưu điểm của XAMPP là cài đặt đơn giản, giao diện thân thiện, dễ sử dụng cho cả người mới học và lập trình viên chuyên nghiệp. Ngoài ra, XAMPP còn tích hợp sẵn phpMyAdmin để quản lý cơ sở dữ liệu MySQL, giúp thao tác với dữ liệu trở nên thuận tiện hơn. Đây là công cụ hữu ích để kiểm thử ứng dụng web trước khi triển khai chính thức lên máy chủ thật.

+Java I/O: Đọc và Ghi Tập Tin Văn Bản

Trong phát triển phần mềm, khả năng xử lý dữ liệu từ tập tin là một yếu tố quan trọng, đặc biệt là trong các ứng dụng yêu cầu lưu trữ và truy xuất thông tin. Trong ngôn ngữ lập trình Java, việc làm việc với các tập tin văn bản có thể được thực hiện thông qua hai gói chính: java.io và java.nio.+ MySQL

MySQL là một hệ quản trị cơ sở dữ liệu quan hệ (RDBMS) mã nguồn mở, được sử dụng rộng rãi để lưu trữ và quản lý dữ liệu trong các ứng dụng web và phần mềm. MySQL sử dụng ngôn ngữ truy vấn SQL (Structured Query Language) để thao tác dữ liệu.

MySQL nổi bật với khả năng xử lý nhanh, đáng tin cậy và dễ tích hợp với các ngôn ngữ lập trình phổ biến như PHP, Java, Python. Đây là một thành phần quan trọng trong nhiều hệ thống như XAMPP, LAMP, được dùng để xây dựng các ứng dụng như hệ thống quản lý thư viện, website bán hàng, quản lý sinh viên,...

+ GUI Builder (Matisse)

GUI Builder, còn được gọi là Matisse, là công cụ thiết kế giao diện đồ họa được tích hợp sẵn trong NetBeans IDE. Công cụ này cho phép người dùng tạo giao diện Java Swing một cách trực quan bằng cách kéo và thả các thành phần như nút bấm (JButton), nhãn (JLabel), bảng (JTable), ô nhập liệu (JTextField)...

Với GUI Builder, lập trình viên có thể dễ dàng thiết kế giao diện người dùng mà không cần viết mã thủ công, giúp tiết kiệm thời gian và giảm lỗi khi xây dựng các ứng dụng desktop. Giao diện được tạo sẽ tự động sinh mã Java tương ứng ở phần thiết kế

**Chương 2: Phân tích thiết kế thư viện sách**

**2.1 Giới thiệu**

Quản lý sách trong thư viện bằng Java là một ứng dụng giúp tổ chức và theo dõi các hoạt động liên quan đến sách, chẳng hạn như thêm sách mới, cập nhật thông tin, tìm kiếm và quản lý trạng thái mượn trả. Bằng cách sử dụng các lớp và cấu trúc dữ liệu trong Java, ứng dụng này hỗ trợ quản lý hiệu quả danh mục sách trong thư viện, giúp người dùng dễ dàng truy cập thông tin và duy trì hệ thống sách một cách khoa học.

* 1. **Tài liệu đặc tả yêu cầu người dùng**

-Người sử dụng phần mềm gồm: Quản lý, nhân viên.

-Hoạt động của hệ thống quản lý thư viện sách:

* + Quản lý có thể thêm, xóa và sửa thông tin nhân viên và các chức năng của nhân viên
  + Quản lý có thể nhập sách từ các nhà cung cấp khác nhau
  + Nhân viên có thể thống kê sách xem mỗi đầu sách còn bao nhiêu quyển, và những sinh viên quá hạn mà chưa trả sách
  + Thủ thư có thể xem thông tin sinh viên và sách đang mượn.
  + Thủ thư có thể quản lý xem sinh viên mượn sách gì, thời gian mượn.
  + Thủ thư có thể xem, sửa thông tin sách. Lọc theo tác giả, ngày nhập kho, mã số sách, tên sách, loại sách, tình trạng và trạngthái mượn
  + Sinh viên có thể mượn tối đa 7 ngày, nếu mượn quá sẽ cảnh báo không được mượn nữa
  + Không thể mượn sách nếu không đủ số lượng trong thư viện, và những sách không có quyền mượn về
  + Xử lý các trường hợp quá hạn trả sách: Hệ thống sẽ tự động nhận diện các trường hợp mượn sách quá hạn, hiển thị thông báo cho người quản trị để xử lý, và có thể ghi nhận các khoản phạt nếu cần.

-Yêu cầu chức năng:

+ Người quản lý, nhân viên:

Quản lý truy cập. sách

Nhập sách mới, thống kê số lượng sách

Quản lý mượn trả sách, sinh viên mượn sách

* Yêu cầu phi chức năng:

Giao diện đồ họa, Tương thích với hệ điều hành Windows

Hoạt động hệ thống trơn tru, hiệu năng của hệ thống

**2.3 Phân tích hệ thống**

* **Mô hình hóa chức năng hệ thống USECASE vẽ dưới dạng sơ đồ UML**

A diagram of a computer

Description automatically generated

* **Mô tả vắn tắt các UseCase:**

**A white sheet with black text

Description automatically generated**

**Ghi chú:**

Admin: Quản trị viên có đầy đủ quyền quản lý hệ thống

Người dùng: Tài khoản chung và đã được cấp thì mới có có thể xem thông tin cơ bản

Các use case con (4-7, 9-13) được bao gồm trong use case cha tương ứng

**2.4Thiết kế hệ thống**

**2.4.1 Giới thiệu :**

Hệ thống phần mềm quản lý thư viện sách được xây dựng với cấu trúc rõ ràng và phân chia theo từng chức năng cụ thể. Mỗi lớp đại diện cho một thành phần trong hệ thống, đảm nhiệm vai trò riêng biệt nhưng có sự liên kết chặt chẽ với nhau thông qua cơ sở dữ liệu và các thao tác người dùng. Dưới đây là mô tả chi tiết các thành phần chính:

**2.4.2: Các lớp**

- Lớp kết nối cơ sở dữ liệu – DBConnection

Lớp DBConnection đóng vai trò trung tâm trong việc kết nối giữa phần mềm và cơ sở dữ liệu. Lớp này cung cấp phương thức getConnection(): Connection để thiết lập và trả về đối tượng kết nối, giúp các lớp khác có thể thực hiện truy vấn dữ liệu. Việc tách riêng lớp kết nối giúp dễ bảo trì và mở rộng hệ thống sau này.

- Nhóm chức năng quản lý người dùng

LoginPage – Trang đăng nhập

Lớp LoginPage cung cấp giao diện và xử lý quá trình xác thực người dùng. Bao gồm:

+ Các trường nhập bt\_username, bt\_password.

+ Các phương thức chính:

\* validateLogin(): boolean – kiểm tra thông tin hợp lệ.

\* login(): void – thực hiện đăng nhập nếu thông tin hợp lệ.

SignupPage – Trang đăng ký tài khoản

Lớp SignupPage cho phép người dùng đăng ký tài khoản mới. Bao gồm:

+ Các trường nhập như bt\_username, bt\_password, bt\_email, bt\_contact.

+ Các phương thức chính:

\* insertSignupDetails(): void – lưu thông tin người dùng mới.

\* checkDuplicateUser(): boolean – kiểm tra trùng lặp tài khoản.

- Trang chủ hệ thống – HomePage

Lớp HomePage là giao diện trung tâm sau khi đăng nhập thành công. Tại đây, người dùng có thể truy cập các chức năng chính của hệ thống như quản lý sách, sinh viên, mượn – trả sách. Các thành phần chính:

+ model: DefaultTableModel – quản lý dữ liệu bảng.

+ Các phương thức:

\* setStudentToTable() – hiển thị danh sách sinh viên.

\* setBookDetailsToTable() – hiển thị thông tin sách.

\* showPieChart() – hiển thị biểu đồ thống kê mượn – trả.

- Quản lý thông tin sinh viên – ManageStudents

Lớp ManageStudents dùng để thêm, sửa, xóa, hiển thị thông tin sinh viên trong hệ thống thư viện. Các thành phần:

+ Các thuộc tính: studentName, studentId, course, branch.

+ Phương thức chính:

\* setStudentDetailsToTable() – đưa dữ liệu sinh viên vào bảng.

\* clearTable() – xóa dữ liệu bảng trước khi cập nhật mới.

- Quản lý sách thư viện – ManageBooks

Lớp ManageBooks cho phép người dùng thêm mới, chỉnh sửa hoặc xóa thông tin sách. Các thuộc tính chính:

+ bookId, bookName, author, quantity.

+ Phương thức:

\* setBookDetailsToTable() – hiển thị danh sách sách.

\* clearTable() – xóa dữ liệu cũ để cập nhật mới.

- Mượn và trả sách

IssueBook – Quản lý mượn sách

Lớp IssueBook xử lý việc mượn sách cho sinh viên. Các thành phần:

+ Trường nhập: bt\_bookId, bt\_studentId.

+ Nhãn hiển thị tên sách và tên sinh viên.

+ Phương thức:

\* getIssueBookDetails() – lấy thông tin sách được mượn.

\* issueBook() – thực hiện thao tác mượn.

\* isAlreadyIssued() – kiểm tra xem sách đã được mượn chưa.

ReturnBook – Quản lý trả sách

Lớp ReturnBook thực hiện quá trình trả sách và cập nhật số lượng sách hiện có. Các phương thức chính:

+ returnBook() – xử lý việc trả sách.

+ updateBookCount() – cập nhật lại số lượng sách trong kho.

- Xem và quản lý lịch sử mượn trả – ViewAllRecord

Lớp ViewAllRecord cung cấp chức năng tra cứu lịch sử mượn và trả sách. Người dùng có thể lọc dữ liệu theo thời gian hoặc mã số sinh viên. Các thành phần chính:

+ Các trường chọn thời gian: date\_fromDate, date\_toDate.

+ Bảng hiển thị dữ liệu mượn/trả.

+ Phương thức:

\* search() – lọc và tìm kiếm thông tin bản ghi.

- Hiển thị chi tiết sách đã mượn – IssueBookDetails

Lớp này chịu trách nhiệm thống kê và hiển thị tất cả sách đang được mượn.

+ Thuộc tính: model: DefaultTableModel.

+ Phương thức:

\* setIssueBookDetailsToTable() – đổ dữ liệu vào bảng.

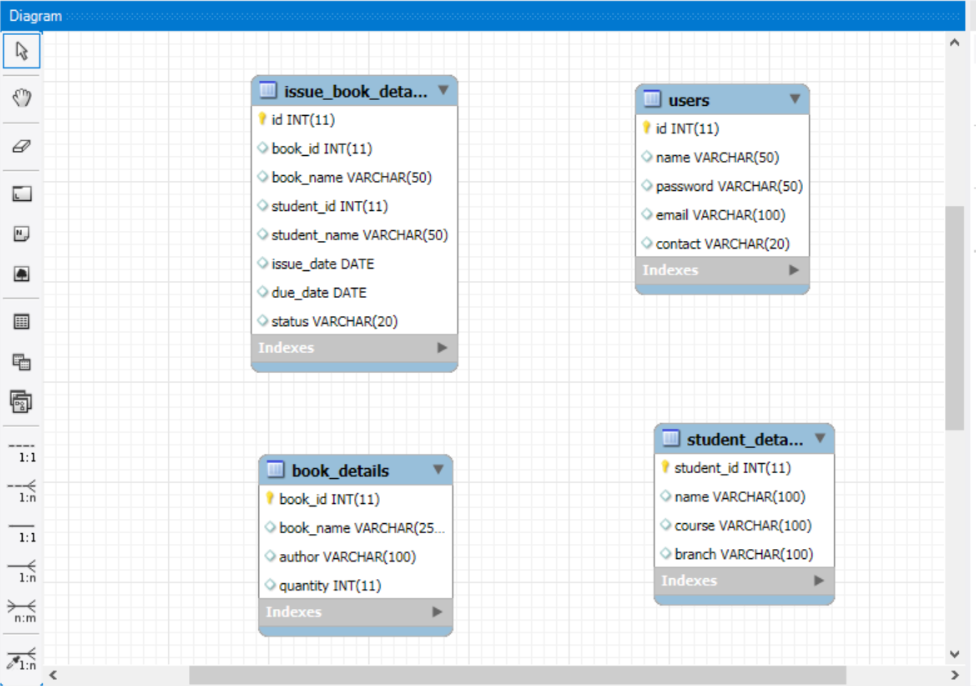
\* clearTable() – xóa bảng trước khi cập nhật.

* **Sơ đồ lớp:**

**A screenshot of a computer

Description automatically generated**

**2.4.3 CSDL**

- Mô hình EER:

**Chương 3: Thiết kế giao diện**

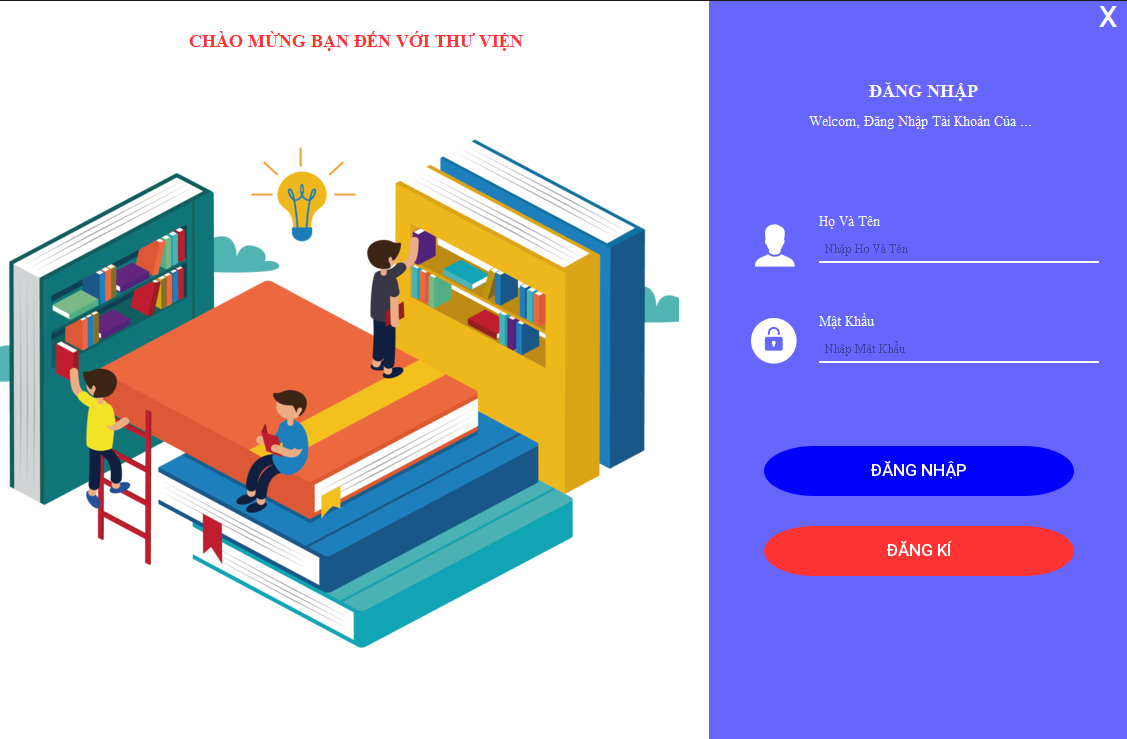
**3.1 Thiết kế hệ thống**

**- Thiết kế giao diện :**

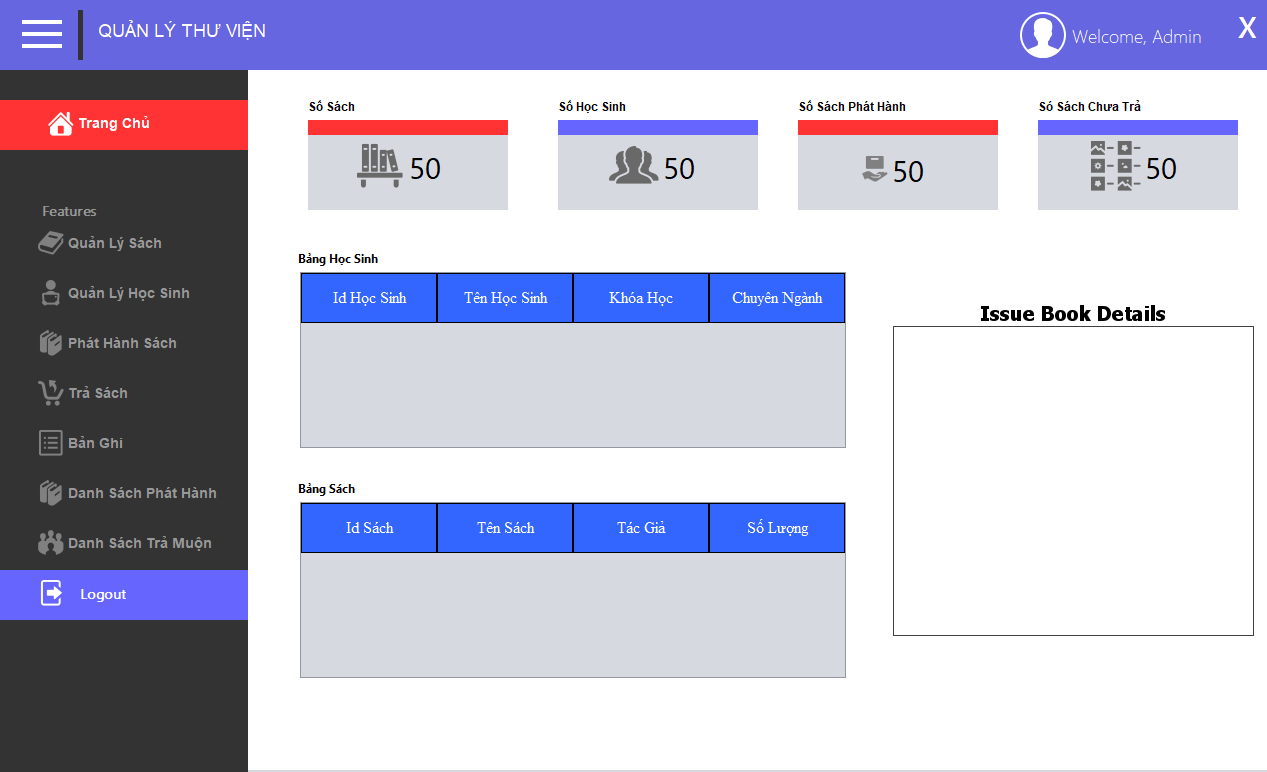
**A group of people sitting on books

Description automatically generated+ Phần đăng ký:**

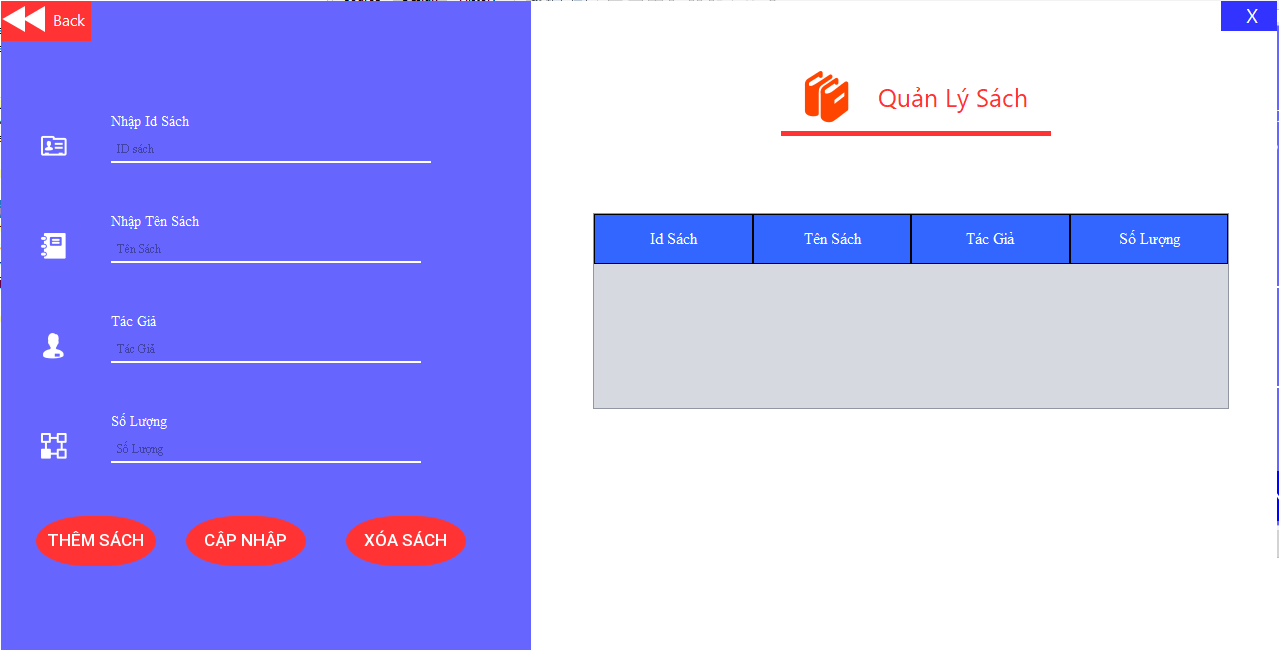
**+ Phần đăng nhập**

****

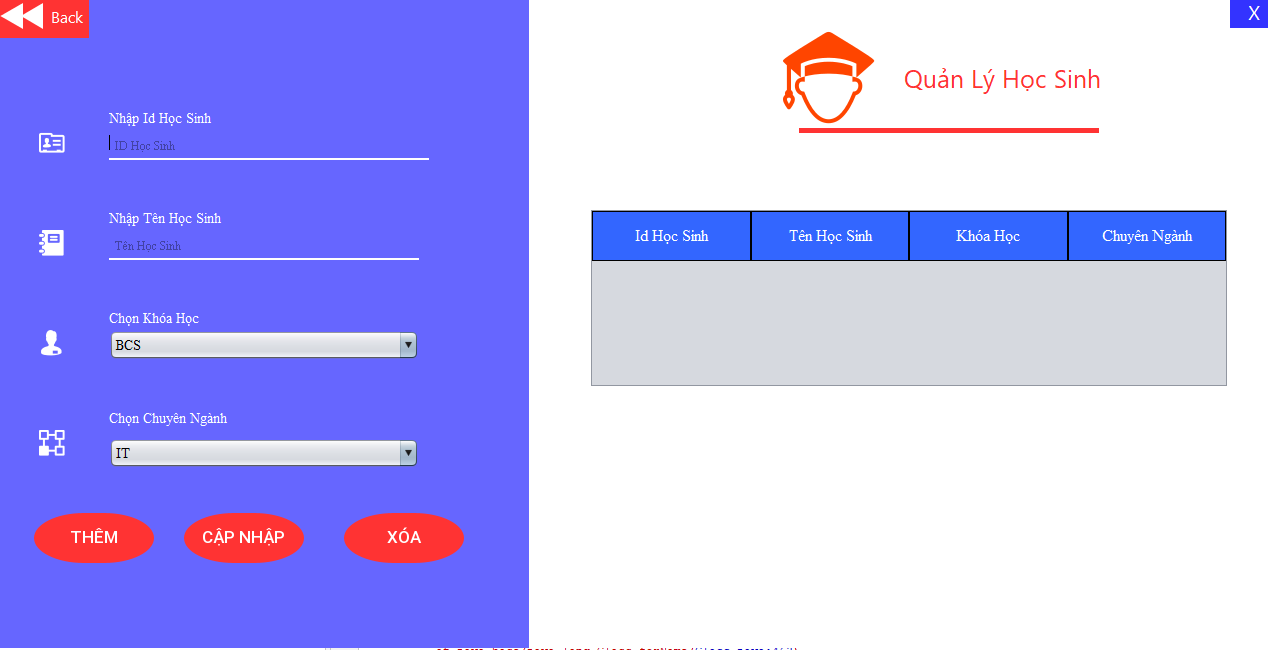
**+ Trang chủ của phần mềm:**

****

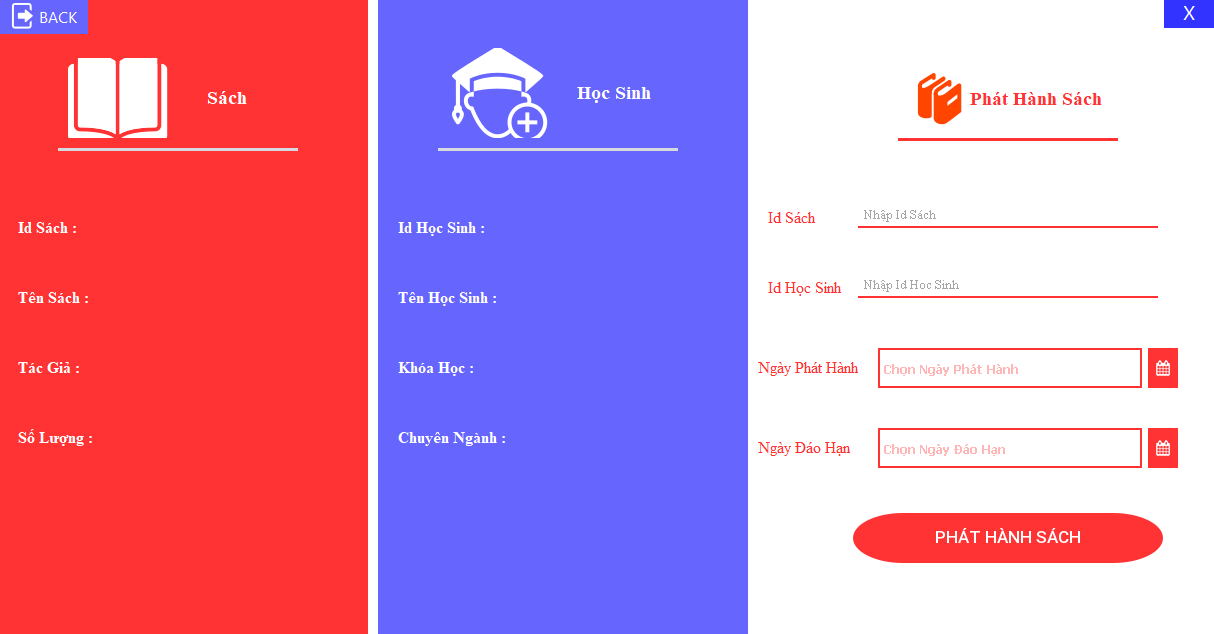
**+ Phần quản lý sách:**



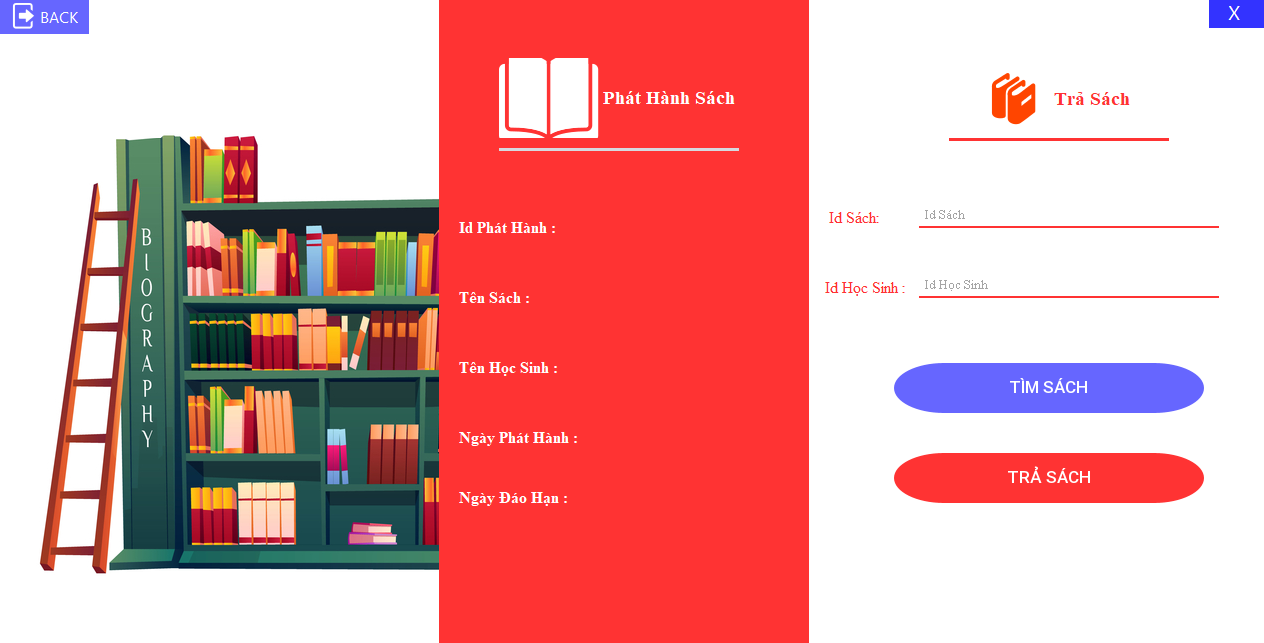
**+ Phần quản lý độc giả:**

****

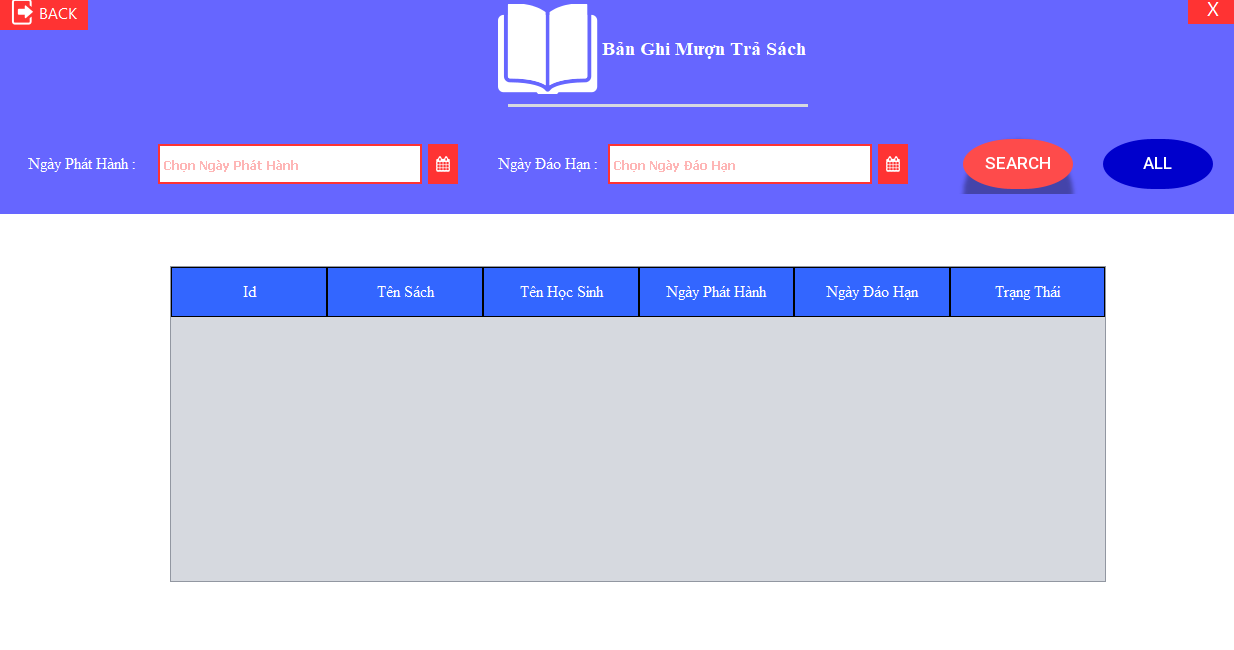
**+ Phần phát hành sách :**



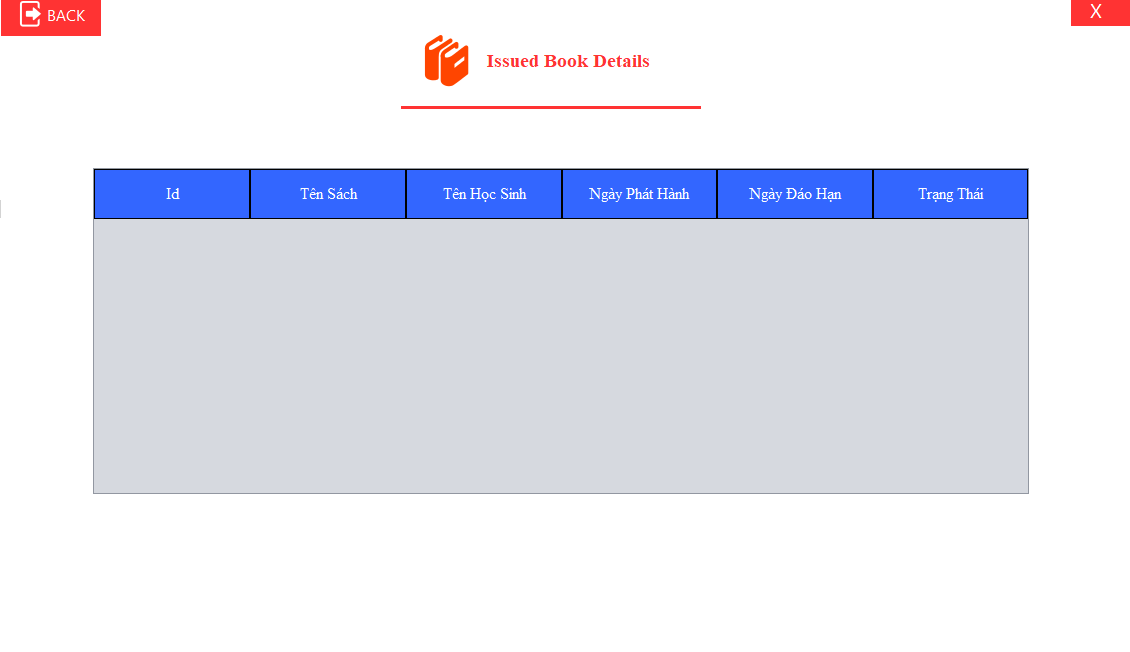
**+ Phần Quản lý mượn trả :**



**+ Phần tìm kiếm:**

****

**+ Phần hiển thị chi tiết sách đã mượn :**

****

**+ Phần danh sách trả muộn :**

**A screenshot of a computer

Description automatically generated**

**Chương 4: Triển khai phần mềm**

Link code : <https://github.com/CSE702025-Nhom-5/projectNhom5>

Cách chạy ứng dụng:

1 Mở XAMPP → Start MySQL hoặc truy cập : http://localhost/phpmyadmin/index.php?route=/sql&pos=0&db=library\_ms&table=book\_details

2 Tạo database và các bảng cần thiết trong phpMyAdmin.

3 Mở NetBeans → Import hoặc mở project Java của bạn.

4 Thêm thư viện MySQL JDBC (mysql-connector-java.jar) vào project.

5 Kiểm tra file DBConnection.java để đảm bảo đúng thông tin:

6 URL kết nối DB

7 Tên người dùng/password, hoặc đăng kí nếu chưa có tài khoản

8 Run file chính LoginPage.java

**Chương 5: Kết luận**

**5.1. Kết luận**

Phần mềm Quản lý thư viện sách đã được xây dựng với các chức năng cơ bản. Giao diện được thiết kế trực quan bằng NetBeans GUI Builder, kết hợp với hệ quản trị cơ sở dữ liệu MySQL, giúp người dùng dễ thao tác và quản lý dữ liệu một cách hiệu quả.

Thông qua quá trình xây dựng, hệ thống đã đáp ứng được các yêu cầu cơ bản của một thư viện nhỏ hoặc trung bình, hỗ trợ người quản lý tiết kiệm thời gian, giảm thiểu sai sót và dễ dàng truy xuất thông tin khi cần.

**5.2. Đánh giá kết quả đạt được**

- Giao diện thân thiện, dễ sử dụng với các tính năng được phân chia rõ ràng.

- Hệ thống kết nối cơ sở dữ liệu ổn định, thực hiện được các thao tác CRUD (Thêm – Xóa – Sửa – Tìm kiếm) hiệu quả.

- Chức năng quản lý mượn – trả sách giúp theo dõi được tình trạng sách và sinh viên mượn.

- Có hỗ trợ thống kê thông tin và biểu đồ giúp quản lý dễ dàng nắm bắt tình hình.

**5.3. Hướng phát triển trong tương lai**

Để phần mềm hoàn thiện hơn và phù hợp với thực tế sử dụng rộng rãi, hệ thống có thể được nâng cấp với các hướng phát triển sau:

Phát triển phiên bản Web/App: Sử dụng Spring Boot hoặc PHP cho web; hoặc Flutter cho app mobile, giúp truy cập mọi lúc mọi nơi.

Thêm chức năng phân quyền người dùng: Phân biệt Admin – Nhân viên – Người dùng tra cứu.

Tích hợp thông báo qua email/SMS khi sách sắp đến hạn trả hoặc có sách mới.

Tăng cường bảo mật: Mã hóa mật khẩu người dùng, kiểm soát truy cập dữ liệu.

Mở rộng báo cáo nâng cao: Biểu đồ mượn sách theo tháng, thống kê sách mượn nhiều nhất,...

Kết nối với nền tảng đám mây: Lưu trữ dữ liệu trực tuyến để dễ quản lý và sao lưu.

**TÀI LIỆU THAM KHẢO**

Sommerville, I. (2016). Software Engineering (10th Edition). Pearson Education.

Pressman, R. S., & Maxim, B. R. (2014). Software Engineering: A Practitioner’s Approach (8th Edition). McGraw-Hill Education.

Freeman, E., Robson, E., Bates, B., & Sierra, K. (2008). Head First Design Patterns. O'Reilly Media.

Bell, I. (2020). Engineering Software Products – An Introduction to Modern Software Engineering. Pearson Education.

Cũng như được chắt lọc và tổng hợp từ nhiều kênh youtube khác nhau