**TRƯỜNG ĐẠI HỌC KINH TẾ - KỸ THUẬT CÔNG NGHIỆP**

**KHOA CÔNG NGHỆ THÔNG TIN**

A blue and white logo

AI-generated content may be incorrect.

**BÀI TẬP LỚN**

**ĐỀ TÀI: Chương trình quản lý thư viện**

Giảng viên hướng dẫn: **Th.S. Trần Thị Huệ**

Nhóm sinh viên thực hiện:

**Nguyễn Tuấn Minh**

**Lương Minh Sơn**

**Trần Nam Khánh**

**Lê Gia Khánh**

Lớp: **DHTI16A1HN**

Khoa: **Công nghệ thông tin**

**Hà Nội - 2025**

**TRƯỜNG ĐẠI HỌC KINH TẾ - KỸ THUẬT CÔNG NGHIỆP**

**KHOA CÔNG NGHỆ THÔNG TIN**

A blue and white logo

AI-generated content may be incorrect.

**BÀI TẬP LỚN**

**ĐỀ TÀI: Chương trình quản lý thư viện**

Giảng viên hướng dẫn: **Th.S. Trần Thị Huệ**

Nhóm sinh viên thực hiện:

**Nguyễn Tuấn Minh**

**Lương Minh Sơn**

**Trần Nam Khánh**

**Lê Gia Khánh**

Lớp: **DHTI16A1HN**

Khoa: **Công nghệ thông tin**

**Hà Nội - 2025**

LỜI CẢM ƠN

Chúng em xin gửi lời cảm ơn chân thành đến cô ThS. Trần Thị Huệ đã luôn sẵn lòng giúp đỡ và hỗ trợ trong quá trình nghiên cứu và thực hiện bài tập lớn.

Cô đã là người đồng hành đáng tin cậy trong suốt quá trình này. Sự tận tâm và kiến thức chuyên môn sâu sắc của cô đã giúp chúng em vượt qua những khó khăn, hiểu rõ hơn về quy trình phát triển phần mềm để áp dụng kiến thức vào thực tế. Không những vậy, cô còn dành thời gian để giải đáp những câu hỏi, hướng dẫn trong việc xây dựng phát triển bài tập lớn môn học. Nhờ những chỉ dẫn và góp ý này, chúng em đã nắm bắt được những khía cạnh quan trọng của quy trình phát triển phần mềm, cách thức xây dựng cơ sở dữ liệu và triển khai ứng dụng.

Chúng em cảm kích sự tận tâm và sự chia sẻ của cô. Những kinh nghiệm và kiến thức nhận được không chỉ giúp chúng em hoàn thành bài tập lớn mà còn định hình một nền tảng vững chắc trong lĩnh vực này.

Một lần nữa, xin gửi lời cảm ơn sâu sắc đến ThS. Trần Thị Huệ đã dành thời gian và công sức để hướng dẫn trong quá trình thực hiện bài tập lớn. Chúng em tự hào khi được học tập và làm việc dưới sự hướng dẫn của cô và tự hứa sẽ nỗ lực không ngừng, áp dụng những kiến thức, kỹ năng đã học được để đóng góp tích cực vào lĩnh vực công nghệ thông tin.

Cuối cùng chúng em xin kính chúc các thầy, cô giáo cùng toàn thể Ban lãnh đạo cùng các phòng ban Trường đại học Kinh tế Kỹ thuật Công nghiệp dồi dào sức khỏe, công tác tốt và gặt hái được nhiều thành công trong sự nghiệp.

Chúng em xin chân thành cảm ơn!

MỤC LỤC

[LỜI CẢM ƠN ii](#_Toc199337344)

[MỤC LỤC iii](#_Toc199337345)

[DANH MỤC BẢNG BIỂU vi](#_Toc199337346)

[DANH MỤC HÌNH ẢNH vii](#_Toc199337347)

[LỜI MỞ ĐẦU 1](#_Toc199337348)

[CHƯƠNG 1: CƠ SỞ LÝ THUYẾT 3](#_Toc199337349)

[1.1 NGÔN NGỮ LẬP TRÌNH JAVA 3](#_Toc199337350)

[1.1.1 Giới thiệu về Java 3](#_Toc199337351)

[1.1.2 Các tính năng của Java 3](#_Toc199337352)

[1.1.3 Ưu & nhược điểm của Java 5](#_Toc199337353)

[1.2 CÔNG CỤ NETBEANS 5](#_Toc199337354)

[1.2.1 Giới thiệu công cụ NetBeans 5](#_Toc199337355)

[1.2.2 Một số tính năng nổi bật của NetBeans: 6](#_Toc199337356)

[1.2.3 Lợi ích khi sử dụng NetBeans: 6](#_Toc199337357)

[1.2.4 Cài đặt chương trình 6](#_Toc199337358)

[1.3 XAMPP 10](#_Toc199337359)

[1.3.1 Giới thiệu về XAMPP 10](#_Toc199337360)

[1.3.2 Ưu & nhược điểm của XAMPP 10](#_Toc199337361)

[1.4 MYSQL 11](#_Toc199337362)

[1.4.1 Tổng quan về MySQL 11](#_Toc199337363)

[1.4.2 Lịch sử hình thành và phát triển của MySQL 12](#_Toc199337364)

[1.5 THƯ VIỆN JDBC 12](#_Toc199337365)

[1.5.1 Tổng quan về thư viên JDBC 12](#_Toc199337366)

[1.5.2 Các thành phần của JDBC 13](#_Toc199337367)

[CHƯƠNG 2: PHÂN TÍCH HỆ THỐNG VÀ THIẾT KẾ CƠ SỞ DỮ LIỆU 14](#_Toc199337368)

[2.1 MÔ TẢ HỆ THỐNG 14](#_Toc199337369)

[2.1.1 Quy trình đăng nhập 14](#_Toc199337370)

[2.1.2 Quản lý tài khoản 14](#_Toc199337371)

[2.1.3 Quản lý sinh viên 14](#_Toc199337372)

[2.1.4 Quản lý sách 14](#_Toc199337373)

[2.1.5 Quản lý Mượn trả sách 14](#_Toc199337374)

[2.1.6 Thống kê 15](#_Toc199337375)

[2.2 PHÂN TÍCH THIẾT KẾ HỆ THỐNG 15](#_Toc199337376)

[2.2.1 Sơ đồ phân rã chức năng 15](#_Toc199337377)

[2.2.2 Biểu đồ Use Case 15](#_Toc199337378)

[2.2.2.1 Biểu đồ use case chức năng Đăng nhập 16](#_Toc199337379)

[2.2.2.2 Biểu đồ use case chức năng Quản lý người dùng 17](#_Toc199337380)

[2.2.2.3 Biểu đồ use case Quản lý sách 18](#_Toc199337381)

[2.2.2.4 Biểu đồ use case Quản lý sinh viên 19](#_Toc199337382)

[2.2.2.5 Biểu đồ use case Quản lý mượn trả 21](#_Toc199337383)

[2.2.2.6 Biểu đồ use case chức năng đăng ký mượn sách 22](#_Toc199337384)

[2.2.2.7 Biểu đồ use case chức năng Lịch sử mượn trả sách 23](#_Toc199337385)

[2.2.2.8 Biểu đồ use case chức năng Thống kê 25](#_Toc199337386)

[2.2.3 Biểu đồ trình tự 26](#_Toc199337387)

[2.2.3.1 Biểu đồ trình tự quản lý tài khoản 26](#_Toc199337388)

[2.2.3.2 Biểu đồ trình từ Đăng nhập 27](#_Toc199337389)

[2.2.3.3 Biểu đồ trình từ Quản lý sách 28](#_Toc199337390)

[2.2.3.4 Biểu đồ trình từ Quản lý sinh viên 29](#_Toc199337391)

[2.3 BẢNG CƠ SỞ DỮ LIỆU 29](#_Toc199337392)

[2.3.1 Các bảng dữ liệu quan hệ 29](#_Toc199337393)

[2.3.2 Các bảng dữ liệu quan hệ 29](#_Toc199337394)

[2.3.3 Mô hình quan hệ (Database Diagrams) 32](#_Toc199337395)

[CHƯƠNG 3: CÀI ĐẶT VÀ THỬ NGHIỆM 33](#_Toc199337396)

[3.1: GIAO DIỆN ĐĂNG NHẬP VÀ ĐĂNG KÝ 33](#_Toc199337397)

[3.1.1 Đăng nhập 33](#_Toc199337398)

[3.1.2 Đăng ký 33](#_Toc199337399)

[3.2 GIAO DIỆN QUẢN TRỊ (ADMIN) 34](#_Toc199337400)

[3.2.1 Giao diện trang chủ 34](#_Toc199337401)

[3.2.2 Giao diện quản lý 35](#_Toc199337402)

[3.3 GIAO DIỆN NGƯỜI DÙNG (USER) 36](#_Toc199337403)

[3.3.1 Giao diện quản lý 36](#_Toc199337404)

[3.3.2 Giao diện quản lý sách 36](#_Toc199337405)

[3.3.3 Giao diện quản lý sinh viên 37](#_Toc199337406)

[3.3.4: Giao diện quản lý mượn trả 37](#_Toc199337407)

[3.3.4.1: Giao diện đăng ký mượn sách 37](#_Toc199337408)

[3.3.4.2: Giao diện đăng ký mượn sách 38](#_Toc199337409)

[3.3.5: Giao diện thống kê 38](#_Toc199337410)

[3.4 KIỂM THỬ HỆ THỐNG 39](#_Toc199337411)

[KẾT LUẬN VÀ ĐỀ NGHỊ 42](#_Toc199337412)

[TÀI LIỆU THAM KHẢO 44](#_Toc199337413)

DANH MỤC BẢNG BIỂU

[Bảng 2.1: Use case Đăng nhập 16](#_Toc199189161)

[Bảng 2.2: Use case Quản lý người dùng 18](#_Toc199189162)

[Bảng 2.3: Use case quản lý sách 19](#_Toc199189163)

[Bảng 2.4: Use case quản lý sinh viên 21](#_Toc199189164)

[Bảng 2.5 Use case quản lý Đăng ký mượn sách 23](#_Toc199189165)

[Bảng 2.6 Use case Lịch sử mượn trả sách 24](#_Toc199189166)

[Bảng 2.7 Use case Thống kê 25](#_Toc199189167)

[Bảng 2.8: Bảng nguoi\_dung 30](#_Toc199189168)

[Bảng 2.9: Bảng sach 30](#_Toc199189169)

[Bảng 2.10: Bảng sinh\_vien 31](#_Toc199189170)

[Bảng 2.11: Bảng nha\_xuat\_ban 31](#_Toc199189171)

[Bảng 2.12: Bảng phieu\_muon 31](#_Toc199189172)

[Bảng 2.13: Bảng phieu\_phat 32](#_Toc199189173)

[Bảng 3.1 Bảng kịch bản kiểm thử (Test Case) cho hệ thống 41](#_Toc199189174)

DANH MỤC HÌNH ẢNH

[Hình 1.1: Giao diện khi khởi động NetBeans IDE 22 7](#_Toc199189175)

[Hình 1.2: Giao diện tạo Project mới 8](#_Toc199189176)

[Hình 1.3: Giao diện đặt tên Project 8](#_Toc199189177)

[Hình 1.5: Giao diện đặt tên Jframe Form 9](#_Toc199189178)

[Hình 1.6: Giao diện code Project 10](#_Toc199189179)

[Hình 2.1: Sơ đồ phân rã chức năng 15](#_Toc199189180)

[Hình 2.2: Biểu đồ use case Đăng nhập 16](#_Toc199189181)

[Hình 2.3: Biểu đồ use case Quản lý người dùng 17](#_Toc199189182)

[Hình 2.4: Biểu đồ use case Quản lý sách 18](#_Toc199189183)

[Hình 2.5: Biểu đồ use case Quản lý sinh viên 20](#_Toc199189184)

[Hình 2.6: Biểu đồ use case Quản lý mượn trả 21](#_Toc199189185)

[Hình 2.7 Biểu đồ use case Đăng ký mượn sách 22](#_Toc199189186)

[Hình 2.8: Biểu đồ use case Lịch sử mượn trả sách 23](#_Toc199189187)

[Hình 2.9 Biểu đồ use case Thống kê 25](#_Toc199189188)

[Hình 2.10: Biểu đồ trình tự đăng ký 26](#_Toc199189189)

[Hình 2.11: Biểu đồ trình tự Đăng nhập 27](#_Toc199189190)

[Hình 2.12: Biểu đồ trình tự Quản lý sách 28](#_Toc199189191)

[Hình 2.13: Biểu đồ trình tự Quản lý sinh viên 29](#_Toc199189192)

[Hình 2.14: Mô hình quan hệ 32](#_Toc199189193)

[Hình 3.1: Giao diện đăng nhập 33](#_Toc199189194)

[Hình 3.2 Giao diện đăng ký 34](#_Toc199189195)

[Hình 3.3 Giao diện trang chủ 34](#_Toc199189196)

[Hình 3.4: Giao diện quản lý (admin) 35](#_Toc199189197)

[Hình 3.5: Giao diện quản lý người dùng 35](#_Toc199189198)

[Hình 3.6: Giao diện quản lý (user) 36](#_Toc199189199)

[Hình 3.7: Giao diện quản lý sách 36](#_Toc199189200)

[Hình 3.8: Giao diện quản lý sinh viên 37](#_Toc199189201)

[Hình 3.9: Giao diện quản lý mượn trả sách 37](#_Toc199189202)

[Hình 3.10: Giao diện đăng ký mượn sách 38](#_Toc199189203)

[Hình 3.11: Giao diện quản lý lịch sử 38](#_Toc199189204)

[Hình 3.12: Giao diện thống kê 39](#_Toc199189205)

LỜI MỞ ĐẦU

**1. LÝ DO CHỌN ĐỀ TÀI**

Trong thời đại công nghệ thông tin phát triển mạnh mẽ, việc số hóa và tự động hóa các quy trình quản lý trở nên cần thiết ở hầu hết các lĩnh vực, đặc biệt trong môi trường giáo dục. Quản lý thư viện – nơi lưu trữ và cung cấp tri thức – nếu được ứng dụng công nghệ phù hợp sẽ góp phần nâng cao hiệu quả tra cứu, mượn trả sách và quản lý dữ liệu, từ đó cải thiện trải nghiệm của người học và cán bộ quản lý.

Trong thực tế, nhiều thư viện vẫn còn áp dụng phương pháp quản lý thủ công, gây khó khăn trong việc tìm kiếm tài liệu, cập nhật thông tin và theo dõi lịch sử mượn sách. Chính vì vậy, nhóm chúng em lựa chọn đề tài “Chương trình quản lý thư viện” với mục tiêu xây dựng một hệ thống phần mềm có tính ứng dụng cao, hỗ trợ việc quản lý thư viện hiệu quả hơn, dễ sử dụng và phù hợp với thực tiễn triển khai trong nhà trường hoặc các tổ chức giáo dục.

**2. MỤC TIÊU NGHIÊN CỨU**

Xây dựng một hệ thống phần mềm hỗ trợ quản lý thư viện gồm các chức năng: quản lý sách, người dùng, sinh viên và hoạt động mượn trả sách.

Hỗ trợ cán bộ thư viện trong việc kiểm soát số lượng sách, tình trạng mượn trả, thống kê và tra cứu thông tin nhanh chóng, chính xác.

Nâng cao trải nghiệm người dùng qua giao diện thân thiện, dễ thao tác.

Ứng dụng công nghệ Java, JDBC và MySQL để triển khai hệ thống đảm bảo tính ổn định, bảo mật và dễ bảo trì.

**3. ĐỐI TƯỢNG VÀ PHẠM VI NGHIÊN CỨU**

***3.1 Đối tượng nghiên cứu***:

Các hoạt động nghiệp vụ trong quản lý thư viện truyền thống và các yêu cầu chức năng cần thiết cho một hệ thống quản lý hiện đại.

***3.2 Phạm vi nghiên cứu:***

Hệ thống được xây dựng phục vụ nhu cầu quản lý thư viện nội bộ trong trường học, hỗ trợ nhân viên thư viện và sinh viên trong việc sử dụng tài nguyên hiệu quả. Hệ thống bao gồm các chức năng cơ bản: đăng nhập, quản lý người dùng, quản lý sách, sinh viên, mượn/trả sách và thống kê dữ liệu.

**4. PHƯƠNG PHÁP NGHIÊN CỨU**

Khảo sát thực trạng quản lý thư viện hiện tại tại một số trường học để hiểu rõ nhu cầu.

Phân tích yêu cầu và thiết kế hệ thống theo hướng trực quan, dễ sử dụng.

Lựa chọn công nghệ phù hợp: sử dụng Java làm ngôn ngữ lập trình chính, MySQL để lưu trữ dữ liệu và thư viện JDBC để kết nối.

Kiểm thử hệ thống trên dữ liệu mẫu và thu thập phản hồi để cải tiến.

CHƯƠNG 1: CƠ SỞ LÝ THUYẾT

1.1 NGÔN NGỮ LẬP TRÌNH JAVA

1.1.1 Giới thiệu về Java

Java là ngôn ngữ lập trình đa nền tảng (cross-platform), được phát triển bởi James Gosling tại Sun Microsystems (nay là Oracle Corporation). Ngôn ngữ lập trình này ra đời vào năm 1995 và được thiết kế để có thể chạy trên các nền tảng khác nhau, từ máy tính cá nhân đến thiết bị di động, các máy chủ và thiết bị nhúng.

Java sử dụng cấu trúc lập trình hướng đối tượng (object-oriented programming - OOP) và được xây dựng trên cơ sở của ngôn ngữ lập trình C++. Nó cung cấp một môi trường chạy ảo (virtual machine) gọi là Java Virtual Machine (JVM), giúp các chương trình Java có thể chạy trên nhiều nền tảng khác nhau mà không cần phải biên dịch lại.

Hiện tại, Java là một trong những ngôn ngữ lập trình phổ biến nhất hiện nay, được sử dụng rộng rãi trong các ứng dụng di động, phát triển web, lập trình trên các thiết bị nhúng, máy tính cá nhân, máy chủ, game và nhiều lĩnh vực khác.

1.1.2 Các tính năng của Java

**Đa nền tảng**

Java được thiết kế để có thể chạy trên nhiều nền tảng khác nhau, vì vậy nó rất phù hợp cho việc phát triển các ứng dụng đa nền tảng. Java sử dụng một máy ảo (JVM - Java Virtual Machine) để chạy mã nguồn, vì vậy mã nguồn được viết một lần và có thể chạy trên nhiều hệ điều hành khác nhau mà không cần thay đổi.

**Quản lý bộ nhớ tự động**

Java có tính năng tự động quản lý bộ nhớ, tức là nó tự động thu dọn các vùng nhớ không sử dụng nữa để giảm thiểu các lỗi bộ nhớ. Điều này giúp cho các ứng dụng được viết bằng Java có thể chạy ổn định và tránh các lỗi liên quan đến bộ nhớ.

**Hỗ trợ đa luồng**

Java có thể xử lý đa luồng, cho phép chương trình thực hiện nhiều tác vụ cùng một lúc. Điều này giúp cho các ứng dụng có thể chạy nhanh và hiệu quả hơn, đặc biệt là khi phải xử lý nhiều tác vụ cùng một lúc.

**Tính bảo mật cao**

Java có các tính năng bảo mật như kiểm tra kiểu tĩnh và kiểm tra lỗi trên đường dẫn. Java được thiết kế để giảm thiểu các lỗ hổng bảo mật và các vấn đề liên quan đến an ninh.

**Hỗ trợ các thư viện và công cụ phát triển mạnh mẽ**

Java có các thư viện tiêu chuẩn và các công cụ phát triển như IDE (Integrated Development Environment) để giúp các nhà phát triển tạo ra các ứng dụng dễ dàng hơn. Các thư viện và công cụ này cung cấp các tính năng như: tạo giao diện người dùng, xử lý dữ liệu, kết nối cơ sở dữ liệu và nhiều hơn nữa ,...

**Tính di động**

Java được sử dụng rộng rãi trong lĩnh vực di động bao gồm: các thư viện hỗ trợ việc phát triển ứng dụng di động, đóng gói ứng dụng thành các file .jar hoặc .apk, cung cấp các tính năng như xử lý đa nhiệm, kết nối mạng và tích hợp với các thiết bị phần cứng như máy ảnh.

**Tính độc lập với nền tảng**

Java có thể chạy trên nhiều nền tảng khác nhau và không phụ thuộc vào bất kỳ nền tảng cụ thể nào. Điều này giúp cho các ứng dụng Java có thể được triển khai trên nhiều hệ thống khác nhau mà không cần sửa đổi mã nguồn.

**Tính kế thừa và đa hình**

Java là ngôn ngữ lập trình hướng đối tượng (OOP), vì vậy nó có các tính năng như kế thừa, đa hình và đóng gói. Các tính năng này giúp cho mã nguồn được tái sử dụng và giảm thiểu sự trùng lặp trong mã nguồn.

**Tính mở rộng**

Java có tính năng mở rộng, cho phép các nhà phát triển thêm các tính năng mới vào ngôn ngữ bằng cách tạo các thư viện và API riêng. Điều này giúp cho Java có thể được sử dụng trong nhiều lĩnh vực khác nhau và được phát triển theo các hướng khác nhau.

1.1.3 Ưu & nhược điểm của Java

**Ưu điểm của Java**

* Độ tin cậy cao
* Tính đa nền tảng
* Quản lý bộ nhớ tự động
* Công cụ phát triển phong phú
* Hỗ trợ đa luồng

**Nhược điểm của Java**

* Tốc độ chậm hơn so với các ngôn ngữ lập trình gần sát với phần cứng, chẳng hạn như C hoặc C++.
* Java có thể chạy trên nhiều nền tảng khác nhau, nhưng ứng dụng này có thể cần đến một trình biên dịch hoặc máy ảo Java riêng biệt để có thể chạy trên các thiết bị di động.
* Sử dụng bộ nhớ lớn hơn so với một số ngôn ngữ lập trình khác.
* Cú pháp phức tạp hơn so với một số ngôn ngữ lập trình khác.

1.2 CÔNG CỤ NETBEANS

1.2.1 Giới thiệu công cụ NetBeans

Apache NetBeans là một môi trường phát triển tích hợp (IDE) mã nguồn mở được sử dụng phổ biến để lập trình các ứng dụng Java, và cũng hỗ trợ nhiều ngôn ngữ khác như PHP, C/C++, HTML5, JavaScript và Python thông qua các plugin mở rộng.

Ban đầu được phát triển bởi Sun Microsystems và sau này được Oracle duy trì, NetBeans hiện nay là một dự án thuộc Apache Software Foundation, với phiên bản mới nhất là NetBeans 22 (phát hành năm 2024–2025).

1.2.2 Một số tính năng nổi bật của NetBeans:

Hỗ trợ Java toàn diện: bao gồm Java SE, Java EE, JavaFX, Maven, Gradle.

* Giao diện trực quan, dễ sử dụng: phù hợp với cả người mới học lập trình và lập trình viên chuyên nghiệp.
* Trình biên dịch và gỡ lỗi tích hợp: giúp lập trình viên kiểm tra và sửa lỗi nhanh chóng.
* Hỗ trợ kéo – thả giao diện (GUI Builder): giúp tạo ứng dụng giao diện người dùng nhanh chóng.
* Tích hợp quản lý dự án với Maven và Gradle.
* Mã nguồn mở và miễn phí 100%: phù hợp sử dụng trong môi trường học tập và nghiên cứu.

1.2.3 Lợi ích khi sử dụng NetBeans:

* Cài đặt đơn giản, không cần cấu hình nhiều.
* Được cộng đồng hỗ trợ mạnh mẽ.
* Tài liệu phong phú, dễ tìm kiếm hướng dẫn học tập.
* **Kết luận:** Apache NetBeans là một IDE mạnh mẽ và linh hoạt, rất phù hợp để học lập trình Java, làm bài tập lớn, xây dựng ứng dụng desktop và web.

1.2.4 Cài đặt chương trình

Để tạo một ứng dụng Java sử dụng Apache NetBeans 22 cần đáp ứng các yêu cầu sau:

* Hệ điều hành: Windows (hoặc macOS, Linux).
* Đã cài đặt Java Development Kit (JDK) 17 trở lên (JDK 21 được khuyến nghị vì là phiên bản LTS).
* Apache NetBeans 22 IDE (hỗ trợ đầy đủ cho Java SE, Java EE, JavaFX...).

**1.2.4.1 Các bước cài đặt:**

**Cài đặt JDK**

* Truy cập trang tải JDK chính thức:  
  <https://www.oracle.com/java/technologies/javase/jdk21-archive-downloads.html>
* Tải JDK phù hợp với hệ điều hành và cài đặt như phần mềm thông thường.
* Thiết lập biến môi trường JAVA\_HOME

**Cài đặt Apache NetBeans 22:**

* Truy cập trang chủ NetBeans:  
  <https://netbeans.apache.org/front/main/download/index.html>
* Tải về bản cài đặt NetBeans 22 cho hệ điều hành của bạn.
* Cài đặt như phần mềm thông thường (chạy file .exe trên Windows).

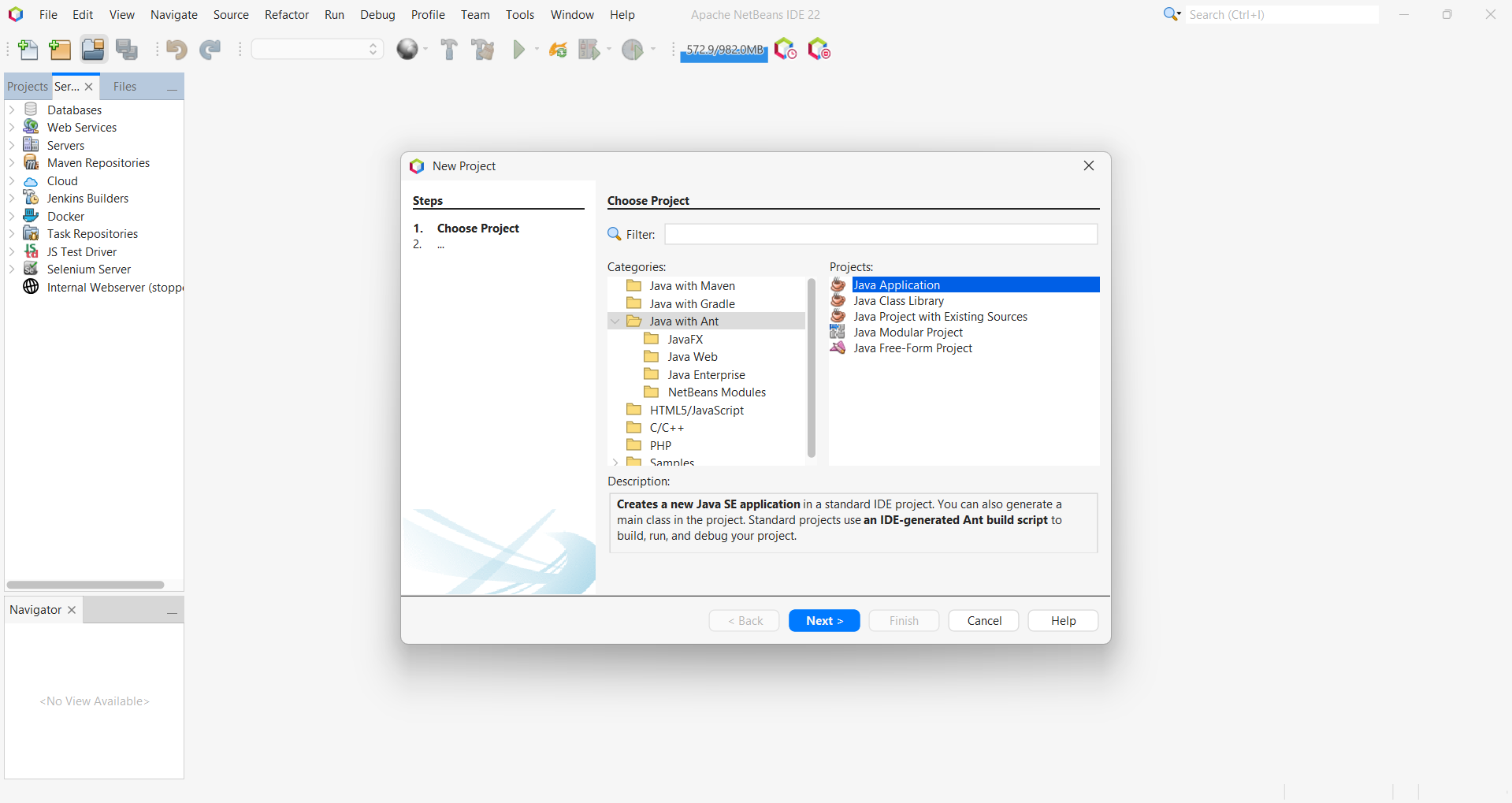
**1.2.4.2 Khởi động NetBeans và tạo Project**

**Bước 1:** Mở NetBean 22 và hiện giao diện như sau:

A screenshot of a computer

AI-generated content may be incorrect.Hình 1.1: Giao diện khi khởi động NetBeans IDE 22

Bước 2: Chọn File => New Project => Java with Ant => Java Application => Next



Hình 1.2: Giao diện tạo Project mới

Bước 3: Đặt tên dự án ở Project name => Finish

A screenshot of a computer

AI-generated content may be incorrect.

Hình 1.3: Giao diện đặt tên Project

Bước 4: Click chuột vào Project vừa tạo => Package => New => Jframe FormA screenshot of a computer

AI-generated content may be incorrect.  
 Hình 1.4: Giao diện tạo Jframe Form

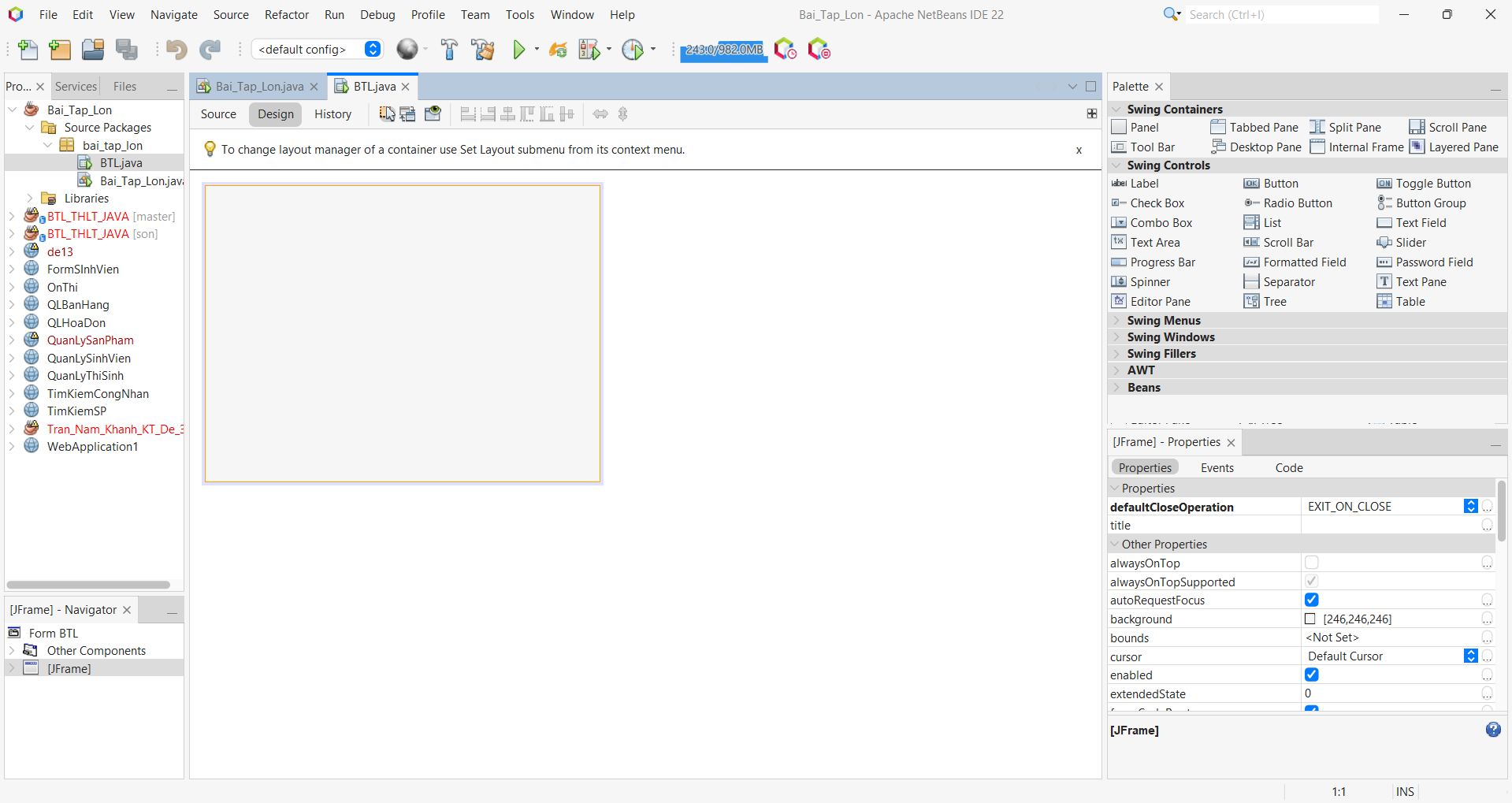
Bước 5: Đặt tên Jframe Form tại mục Class name => Finish

A screenshot of a computer

AI-generated content may be incorrect.

Hình 1.5: Giao diện đặt tên Jframe Form

Bước 6: Hoàn thành tạo Project mới



Hình 1.6: Giao diện code Project

1.3 XAMPP

1.3.1 Giới thiệu về XAMPP

XAMPP là một phần mềm cho phép giả lập môi trường server hosting ngay trên máy tính của bạn, cho phép bạn chạy demo website mà không cần phải mua hosting hay [VPS](https://www.thegioididong.com/hoi-dap/vps-la-gi-dung-de-lam-gi-co-uu-nhuoc-diem-nhu-the-nao-1337214). Chính vì vậy, XAMPP hay được phục vụ cho hoạt động học tập giảng dạy thực hành và phát triển web.

XAMPP được viết tắt của X + Apache + MySQL + PHP + Perl vì nó được tích hợp sẵn , [Apache](https://vi.wikipedia.org/wiki/Apache_(HTTP)), [MySQL](https://www.thegioididong.com/game-app/huong-dan-cach-tai-cai-dat-mysql-ban-moi-nhat-chi-tiet-tung-1299084), [PHP](https://vi.wikipedia.org/wiki/PHP) FTP server, Mail Server. Còn X thể hiện cho sự đa nền tảng của XAMPP vì nó có thể dùng được cho 4 hệ điều hành khác nhau: [Windows](https://www.thegioididong.com/phan-mem/windows-10-home-32-bit-64-bit-all-languages-kw9-0), [MacOS](http://www.thegioididong.com/hoi-dap/he-dieu-hanh-macos-la-gi-956642), [Linux](https://www.thegioididong.com/hoi-dap/he-dieu-hanh-linux-la-gi-uu-nhuoc-diem-cua-he-dieu-hanh-1312530) và [Solaris](https://vi.wikipedia.org/wiki/Solaris_(h%E1%BB%87_%C4%91i%E1%BB%81u_h%C3%A0nh)).

1.3.2 Ưu & nhược điểm của XAMPP

**Ưu điểm**

* Có thể chạy được linh hoạt được trên mọi hệ điều hành: [Cross-platform](https://vi.wikipedia.org/wiki/%C4%90a_n%E1%BB%81n_t%E1%BA%A3ng), Windows, MacOS, Linux.
* Có cấu hình đơn giản nhưng mang đến nhiều chức năng hữu ích như: Lập [Server](https://www.thegioididong.com/hoi-dap/may-chu-server-la-gi-co-may-loai-co-vai-tro-nhu-the-nao-1335540) giả định, lập Mail Server giả định và hỗ trợ SSL trên [localhost](https://vi.wikipedia.org/wiki/Localhost).
* Tích hợp được nhiều tính năng với các thành phần quan trọng như: Apache, PHP, MySql. Vì thế, người dùng không cần cài đặt từng phần trên riêng lẻ mà chỉ cần cài XAMPP là có 1 web server hoàn chỉnh.
* Tạo mã nguồn mở: Giao diện quản lý dễ dàng và tiện lợi giúp người dùng luôn chủ động được trong chế độ khởi động lại hay bật/ tắt đối với máy chủ theo thời gian phù hợp nhất.

**Nhược điểm**

* Do cấu hình khá đơn giản cho nên Xampp hoàn toàn không nhận được sự hỗ trợ về cấu hình [Module](https://www.thegioididong.com/game-app/module-la-gi-dinh-nghia-cua-module-trong-cac-linh-vuc-1336435), đồng thời cũng không có cả MySQL.
* Dung lượng tương đối nặng (141Mb).
* Không có nhiều phiên bản cho từng thành phần của server như PHP, Apache mà phải cài đặt riêng.

1.4 MYSQL

1.4.1 Tổng quan về MySQL

MySQL là một hệ quản trị cơ sở dữ liệu quan hệ (Relational Database Management System – RDBMS) mã nguồn mở, được sử dụng rộng rãi để lưu trữ, quản lý và truy xuất dữ liệu. MySQL hoàn toàn miễn phí nằm trong nhóm LAMP (Linux – Apache – MySQL – PHP).

Vì hệ quản trị cơ sở dữ liệu MySQL ổn định và dễ sử dụng, có tính khả chuyển, hoạt động trên nhiều hệ điều hành cung cấp một hệ thống lớn các hàm tiện ích rất mạnh và Mysql cũng có cùng một cách truy xuất và mã lệnh tương tự với ngôn ngữ SQL chính vì thế nên MySQL được sử dụng và hỗ trợ của những lập trình viên yêu thích mã nguồn mở.

Nhưng MySQL không bao quát toàn bộ những câu truy vấn cao cấp như SQL Server. Vì vậy MySQL chỉ đáp ứng việc truy xuất đơn giản trong quá trình vận hành của website, thích hợp cho các ứng dụng có truy cập [CSDL](https://topdev.vn/blog/phan-1-gioi-thieu-co-so-du-lieu-csdl-he-quan-tri-csdl/) trên internet và có thể giải quyết hầu hết các bài toán trong PHP, Perl.

MySQL có nhiều phiên bản cho các hệ điều hành khác nhau: phiên bản Win32 cho các hệ điều hành dòng Windows,Linux, Mac OSX, Unix, FreeBSD, NetBSD, Novell NetWare, SGI Irix, Solaris, SunOS,…

1.4.2 Lịch sử hình thành và phát triển của MySQL

MySQL được phát triển vào năm 1994 bởi công ty Thụy Điển MySQL AB. Sau khi trải qua nhiều giai đoạn phát triển và chuyển giao, với những cột mốc quan trọng như việc trở thành mã nguồn mở vào năm 2000, sau đó Sun Microsystems đã mua lại MySQL vào năm 2008. Nhưng sau đó chỉ hai năm gã khổng lồ Oracle đã giữ quyền sở hữu MySQL vào năm 2010

Sự ra đời của MariaDB như một “fork” của MySQL đã tạo ra một cuộc cạnh tranh lành mạnh, thúc đẩy cả hai hệ thống không ngừng cải tiến. Nhờ tính năng mã nguồn mở, hiệu suất cao, dễ sử dụng và cộng đồng lớn, MySQL đã trở thành một trong những hệ quản trị cơ sở dữ liệu phổ biến nhất thế giới, được sử dụng rộng rãi trong các ứng dụng web và các hệ thống lưu trữ dữ liệu.

1.5 THƯ VIỆN JDBC

1.5.1 Tổng quan về thư viên JDBC

JDBC, là viết tắt của Java Database Connectivity, là một Java API chuẩn để kết nối giữa ngôn ngữ lập trình Java và các cơ sở dữ liệu đa dạng. Sử dụng JDBC, bạn có thể thực hiện nhiều tác vụ đa dạng khi làm việc với cơ sở dữ liệu như tạo, xóa cơ sở dữ liệu; tạo và thực thi các lệnh SQL hoặc MySQL; tạo, xóa các bản ghi; …

Trước khi có JDBC, ODBC API được sử dụng là API để kết nối và thực thi truy vấn với cơ sở dữ liệu. Nhưng API này có một hạn chế rõ ràng là nó sử dụng ODBC driver được viết bằng ngôn ngữ C (là phụ thuộc nền tảng và không an toàn).

Do đó, Java đã định nghĩa riêng API (là JDBC API) để thực hiện các thao tác khi làm việc với cơ sở dữ liệu, và tất nhiên các JDBC driver này được viết bằng ngôn ngữ Java để cung cấp các lợi thế rõ rệt của ngôn ngữ này.

API là viết tắt của Application Programming Interface, là một tài liệu gồm sự mô tả về tất cả các đặc điểm của một sản phẩm hoặc phần mềm. Nó biểu diễn các Class và Interface mà các chương trình phần mềm có thể theo sau để giao tiếp với nhau. Một API có thể được tạo cho các ứng dụng, các thư viện, các hệ điều hành, …

1.5.2 Các thành phần của JDBC

JDBC API cung cấp các Class và Interface sau:

* DriverManager: Lớp này quản lý các Database Driver. Ánh xạ các yêu cầu kết nối từ ứng dụng Java với Data driver thích hợp bởi sử dụng giao thức kết nối phụ.
* Driver: Interface này xử lý các kết nối với Database Server. Hiếm khi, bạn tương tác trực tiếp với các đối tượng Driver này. Thay vào đó, bạn sử dụng các đối tượng DriverManager để quản lý các đối tượng kiểu này.
* Connection: Đối tượng Connection biểu diễn ngữ cảnh giao tiếp. Interface này chứa nhiều phương thức đa dạng để tạo kết nối với một Database.
* Statement: Bạn sử dụng các đối tượng được tạo từ Interface này để đệ trình các lệnh SQL tới Database. Ngoài ra, một số Interface kết thừa từ nó cung chấp nhận thêm các tham số để thực thi các thủ tục đã được lưu trữ.
* ResultSet: Các đối tượng này giữ dữ liệu được thu nhận từ một Database sau khi bạn thực thi một truy vấn SQL. Nó nóng vai trò như một Iterator để cho phép bạn vọc qua dữ liệu của nó.
* SQLException: Lớp này xử lý bất cứ lỗi nào xuất hiện trong khi làm việc với Database.

CHƯƠNG 2: PHÂN TÍCH HỆ THỐNG VÀ THIẾT KẾ CƠ SỞ DỮ LIỆU

2.1 MÔ TẢ HỆ THỐNG

2.1.1 Quy trình đăng nhập

Khi nhân viên muốn sử dụng hệ thống, nhân viên sẽ phải truy cập vào trang web của hệ thống và tiến hành đăng nhập. Nhân viên đăng nhập vào hệ thống bằng tài khoản cùng mật khẩu đã đăng ký bởi admin. Hệ thống sẽ hiển thị giao diện.

2.1.2 Quản lý tài khoản

Chức năng Quản lý cho phép Admin tạo, chỉnh sửa, xóa tài khoản người dùng. Khi tạo tài khoản mới, Admin nhập thông tin, hệ thống kiểm tra kích hoạt. Admin có thể chỉnh sửa thông tin của người dùng. Nu cần, Admin có thể xóa tài khoản khi cần thiết. Hệ thống đảm bảo mọi thay đổi được cập nhật trong cơ sở dữ liệu và thông báo kết quả cho Admin.

2.1.3 Quản lý sinh viên

Sau khi đăng nhập, nhân viên sẽ được chuyển đến giao diện quản lý thư viện. Tại đây nhân viên có thể quản lý sinh viên, thông tin sinh viên được hiện ra trên bảng . Nhân viên có thể thêm, sửa, xóa , tìm kiếm thông tin sinh viên

2.1.4 Quản lý sách

Tại đây nhân viên có thể thống kê được số lượng sách có trong thư viên. Nhân viên có thể thêm, sửa, xóa , tìm kiếm thông tin của sách.

2.1.5 Quản lý Mượn trả sách

Sau khi đăng nhập, nhân viên sẽ được chuyển đến giao diện quản lý thư viện. Trong quản lý mượn trả sách có đăng ký mượn sách và quản lý lịch sử.

Trong đăng ký mượn sách, nhân viên nhập mã sinh viên trên hệ thông hiển thị thông tin sinh viên, nhập mã sách để hiển thị thông tin sách, nhập ngày trả dự kiến và phí mượn sách. Cuối cùng hiện thị thông báo thêm sinh viên vào lịch sử mượn sách.

Trong quản lý lịch sử mượn, nhân viên nhập mã sinh viên để hiển thị thông tin sinh viên đã mượn sách . Nhân viên nhập ngày sinh viên trả sách, rồi bấm xác nhận trả. Bảng hiển thị tình trạng là đã trả sách.

2.1.6 Thống kê

Chức năng thống kê cho phép nhân viên và admin thống kê được danh sách lịch sử sinh viên mượn sách. Tại đây có thể thống kê được danh sách các sinh viên đã trả sách, mượn sách, hoặc là đang mượn sách.

2.2 PHÂN TÍCH THIẾT KẾ HỆ THỐNG

2.2.1 Sơ đồ phân rã chức năng

A diagram of a company

AI-generated content may be incorrect.

Hình 2.1: Sơ đồ phân rã chức năng

2.2.2 Biểu đồ Use Case

2.2.2.1 Biểu đồ use case chức năng Đăng nhập

A diagram of a flowchart

AI-generated content may be incorrect.

Hình 2.2: Biểu đồ use case Đăng nhập

**Đặc tả use case:**

|  |  |
| --- | --- |
| **Hoạt động của tác nhân** | **Phản ứng của hệ thống** |
| 1: Chọn đăng nhập |  |
|  | 2: Hiển thị giao diện đăng nhập |
| 3: Nhập thông tin tài khoản, mật khẩu |  |
| 4: Gửi thông tin đến hệ thống |  |
|  | 5: Kiểm tra thông tin và xác nhận |
|  | 6: Đúng thì đưa người dùng về giao diện được phân quyền |
|  | 7: Sai thì yêu cầu nhập lại thông tin |

Bảng 2.1: Use case Đăng nhập

**Tên use case:** Đăng Nhập

**Tác nhân:** Người dùng

**Chức năng của use case:** Đăng nhập để sử dụng các chức năng được hệ thống phân quyền

**Dòng sự kiện chính:**

* Người dùng nhập thông tin đăng nhập để đăng nhập vào hệ thống
* Sau khi đăng nhập thành công có thể sử dụng các chức năng được hệ thống phân quyền

**Dòng sự kiện thay thế:**

* Nếu người dùng nhập sai hoặc thiếu thông tin thì sẽ được yêu cầu nhập lại.

2.2.2.2 Biểu đồ use case chức năng Quản lý người dùng

A diagram of a diagram

AI-generated content may be incorrect.

Hình 2.3: Biểu đồ use case Quản lý người dùng

**Đặc tả use case:**

|  |  |
| --- | --- |
| **Hoạt động của tác nhân** | **Phản ứng của hệ thống** |
| 1: Đăng nhập vào tài khoản admin |  |
|  | 2: Hiển thị giao diện quản lý tài khoản |
| 3: Thực hiện các chức năng của admin |  |
|  | 4: Hiển thị giao diện tương ứng với chức năng admin chọn |

Bảng 2.2: Use case Quản lý người dùng

**Tên use case:** Quản lý người dùng

**Tác nhân:** Admin

**Chức năng của use case:** Admin có thể quản lý, thêm, sửa, xóa, phân quyền cho các tài khoản.

**Dòng sự kiện chính:**

* Người dùng đăng nhập vào hệ thống với phân quyền là admin
* Admin có thể thực hiện các thêm, sửa, xóa tài khoản của người dùng khác.

2.2.2.3 Biểu đồ use case Quản lý sách

A diagram of a diagram

AI-generated content may be incorrect.

Hình 2.4: Biểu đồ use case Quản lý sách

Quản lý sách

|  |  |
| --- | --- |
| **Hoạt động của tác nhân** | **Phản ứng của hệ thống** |
| 1: Đăng nhập vào hệ thống với tài khoản được phân quyền là nhân viên |  |
|  | 2: Hiển thị màn hình chính tương ứng |
| 3: Thực hiện thao tác thêm, sửa, xóa tìm kiếm sách |  |
|  | 4: Hiển thị giao diện để thêm, sửa, xóa tìm kiếm sách |
| 5: Nhập các thông tin tương ứng |  |
| 6: Xác nhận thêm, sửa, xóa, tìm kiếm sách |  |
|  | 7: Cập nhật lại CSDL và hiển thị ra màn hình |

Bảng 2.3: Use case quản lý sách

**Tên use case:** Quản lý sách

**Tác nhân:** Nhân viên

**Chức năng của use case:** Giúp nhân viên có thể thêm sửa xóa tìm kiếm sách để hỗ trợ việc quản lý thư viện 1 cách dễ dàng.

**Dòng sự kiện chính:**

* Người dùng đăng nhập vào hệ thống với tư cách là nhân viên
* Chọn thao tác thêm, sửa, xóa, tìm kiếm sách mà nhân viên muốn thực hiện
* Thực hiện nhập thông tin hệ thống yêu cầu và xác nhận chức năng muốn thực hiện
* Hệ thống dựa theo yêu cầu của nhân viên để thêm, sửa, xóa , tìm kiếm sách trong CSDL

**Dòng sự kiện thay thế:**

* Khi nhập thiếu thông tin hệ thống sẽ yêu cầu nhập đủ

Khi nhân viên muốn hủy thực hiện chức năng thì hệ thống sẽ hủy yêu cầu cập nhật CSDL

2.2.2.4 Biểu đồ use case Quản lý sinh viên

A diagram of a diagram

AI-generated content may be incorrect.

Hình 2.5: Biểu đồ use case Quản lý sinh viên

**Đặc tả use case:**

Quản lý sách

|  |  |
| --- | --- |
| **Hoạt động của tác nhân** | **Phản ứng của hệ thống** |
| 1: Đăng nhập vào hệ thống với tài khoản được phân quyền là nhân viên |  |
|  | 2: Hiển thị màn hình chính tương ứng |
| 3: Thực hiện thao tác thêm, sửa, xóa tìm kiếm sinh viên |  |
|  | 4: Hiển thị giao diện để thêm, sửa, xóa tìm kiếm sinh viên |
| 5: Nhập các thông tin tương ứng |  |
| 6: Xác nhận thêm, sửa, xóa , tìm kiếm sinh viên |  |
|  | 7: Cập nhật lại CSDL và hiển thị ra màn hình |

Bảng 2.4: Use case quản lý sinh viên

**Tên use case:** Quản lý sách

**Tác nhân:** Nhân viên

**Chức năng của use case:** Giúp nhân viên có thể thêm sửa xóa tìm kiếm sách để hỗ trợ việc quản lý sinh viên 1 cách dễ dàng.

**Dòng sự kiện chính:**

* Người dùng đăng nhập vào hệ thống với tư cách là nhân viên
* Chọn thao tác thêm, sửa, xóa, tìm kiếm sách nhân viên muốn thực hiện
* Thực hiện nhập thông tin hệ thống yêu cầu và xác nhận chức năng muốn thực hiện ..
* Hệ thống dựa theo yêu cầu của nhân viên để thêm, sửa, xóa , tìm kiếm sinh viên trong CSDL.

**Dòng sự kiện thay thế:**

* Khi nhập thiếu thông tin hệ thống sẽ yêu cầu nhập đủ.
* Khi nhân viên muốn hủy thực hiện chức năng thì hệ thống sẽ hủy yêu cầu cập nhật CSDL.

2.2.2.5 Biểu đồ use case Quản lý mượn trả

A diagram of a diagram

AI-generated content may be incorrect.

Hình 2.6: Biểu đồ use case Quản lý mượn trả

**Tên user care:** Quản lý mượn trả sách

**Tác nhân:** Nhân viên

**Chức năng của use case:** Thực hiện các chức năng giúp sinh viên có thể đăng ký mượn sách, trả sách, nhân viên có thể thống kê danh sách mượn trả sách

**Dòng sự kiện chính:**

* Người dùng đăng nhập vào hệ thống với tư cách là nhân viên
* Sử dụng các chức năng hỗ trợ việc mượn sách của sinh viên
* Thống kê tình trạng mượn trả

2.2.2.6 Biểu đồ use case chức năng đăng ký mượn sách

A diagram of a flowchart

AI-generated content may be incorrect.

Hình 2.7 Biểu đồ use case Đăng ký mượn sách

**Đặc tả use case:**

|  |  |
| --- | --- |
| **Hoạt động của tác nhân** | **Phản ứng của hệ thống** |
| 1: Đăng nhập vào hệ thống với tài khoản được phân quyền là nhân viên |  |
|  | 2: Hiển thị màn hình chính tương ứng |
| 3: Tìm kiếm tên sinh viên |  |
|  | 4: Hiển thị tên sinh viên trùng với mã đã nhập |
| 5: Tìm kiếm mã sách |  |
|  | 6: Hiển thị thông tin sách trùng với mã đã nhập |
|  | 7: Thêm sinh viên vào lịch sử mượn sách |

Bảng 2.5 Use case quản lý Đăng ký mượn sách

**Tên user care:** Quản lý đăng ký mượn sách

**Tác nhân:** Nhân viên

**Chức năng của use case:** Nhân viên giúp sinh viên tìm sách muốn mượn

**Dòng sự kiện chính:**

* Người dùng đăng nhập vào hệ thống với tư cách là nhân viên
* Nhập mã mã sinh viên để hiển thị thông tin sinh viên
* Nhập mã sách để tìm sách sinh viên muốn mượn
* Thêm sinh viên vào lịch sử mượn sách

2.2.2.7 Biểu đồ use case chức năng Lịch sử mượn trả sách

A diagram of a diagram

AI-generated content may be incorrect.

Hình 2.8: Biểu đồ use case Lịch sử mượn trả sách

**Đặc tả use case:**

|  |  |
| --- | --- |
| **Hoạt động của tác nhân** | **Phản ứng của hệ thống** |
| 1: Đăng nhập vào hệ thống với tài khoản được phân quyền là nhân viên |  |
|  | 2: Hiển thị màn hình chính tương ứng |
| 3: Tìm kiếm mã sinh viên |  |
|  | 4: Hiển thị thông tin sinh viên, sách , tình trạng sách trùng với mã đã nhập |
| 5: Xác nhận trả sách |  |
|  | 6: Hiển thị thị tình trạng sách đã được trả |

Bảng 2.6 Use case Lịch sử mượn trả sách

**Tên user care:** Quản lý đăng ký mượn sách

**Tác nhân:** Nhân viên

**Chức năng của use case:** Nhân viên giúp sinh viên trả sách đã mượn

**Dòng sự kiện chính:**

* Người dùng đăng nhập vào hệ thống với tư cách là nhân viên
* Nhập mã mã sinh viên để hiển thị thông tin sinh viên
* Nhập mã sách để hiển thị sách sinh viên đã mượn
* Hiển thị tình trạng sách đã được trả
* Nếu trả sách quá hạn thì in phiếu phạt

2.2.2.8 Biểu đồ use case chức năng Thống kê

A diagram of a diagram

AI-generated content may be incorrect.

Hình 2.9 Biểu đồ use case Thống kê

**Đặc tả use case:**

|  |  |
| --- | --- |
| **Hoạt động của tác nhân** | **Phản ứng của hệ thống** |
| 1: Đăng nhập vào hệ thống với tài khoản được phân quyền là nhân viên |  |
|  | 2: Hiển thị màn hình chính tương ứng |
| 3: Thống kê lịch sử mượn, trả |  |
|  | 4: Hiển thị thông tin sinh viên mượn trả |
| 5: Phân loại theo tình trạng sách |  |
|  | 6: Hiển thị thông tin sinh viên mượn trả theo từng tình trạng |

Bảng 2.7 Use case Thống kê

**Tên user care:** Thống kê

**Tác nhân:** Nhân viên

**Chức năng của use case:** Nhân viên thống kê lịch sử mượn, trả.

**Dòng sự kiện chính:**

* Người dùng đăng nhập vào hệ thống với tư cách là nhân viên
* Ấn thống kê để hiển thị lịch sử mượn, trả
* Tích vào các ô tình trạng sách: “Đang mượn”, “Quá hạn”, “Đã trả”
* Hiển thị thông tin chi tiết tình trạng mượn sách

2.2.3 Biểu đồ trình tự

2.2.3.1 Biểu đồ trình tự quản lý tài khoản

**A diagram of a company

AI-generated content may be incorrect.**

Hình 2.10: Biểu đồ trình tự đăng ký

**Mô tả tóm tắt:** Người dùng đăng nhập vào hệ thống với tư cách là admin

***Chức năng Thông tin tài khoản:***

* Admin gửi yêu cầu xem thông tin của tài khoản.
* Hệ thống sẽ nhận được yêu cầu và tìm kiếm thông tin trong CSDL.
* Sau khi tìm được thông tin tài khoản hệ thống sẽ hiển thị thông tin cho admin xem.

***Chức năng Tạo tài khoản:***

* Admin nhập thông tin, phân quyền cho tài khoản cần tạo và gửi yêu cầu tạo tài khoản đến hệ thông.
* Hệ thống nhận thông tin và xác thực thông tin
* Nếu thông tin tài khoản không bị trùng thì thông báo tạo thành công, ngược lại nếu tài khoản đã tồn tại sẽ thông báo cho admin.

***Chức năng Cập nhật tài khoản:***

* Admin gửi yêu cầu cập nhật cho hệ thống
* Hệ thống kiểm tra thông tin và phản hồi lại admin

***Chức năng Xóa tài khoản:***

* Admin gửi yêu cầu xóa tài khoản đến hệ thông
* Hệ thống kiểm tra thông tin và phản hồi lại admin

2.2.3.2 Biểu đồ trình từ Đăng nhập

A diagram of a diagram

AI-generated content may be incorrect.

Hình 2.11: Biểu đồ trình tự Đăng nhập

**Mô tả tóm tắt:**

* Người dùng nhập thông tin mà hệ thống yêu cầu để đăng nhập vào hệ thống.
* Hệ thống nhận thông tin và kiểm tra xác thực thông tin xem tài khoản có trong CSDL không.
* Nếu các thông tin đăng nhập của tài khoản đúng thì người dùng sẽ được đưa đến giao diện màn hình chính và sử dụng các chức năng được phân quyền.
* Ngược lại, nếu người dùng nhập sai hoặc thiếu thông tin thì hệ thống sẽ báo lỗi và đăng nhập thất bại.

2.2.3.3 Biểu đồ trình từ Quản lý sách

**A diagram of a diagram

AI-generated content may be incorrect.**

Hình 2.12: Biểu đồ trình tự Quản lý sách

**Mô tả tóm tắt:** Người dùng đăng nhập vào hệ thống với tư cách là nhân viên

***Chức năng Thông tin sách:***

* Nhân viên gửi yêu cầu xem thông tin sách
* Hệ thống sẽ nhận được yêu cầu và tìm kiếm thông tin trong CSDL.
* Sau khi tìm được thông tin tài khoản hệ thống sẽ hiển thị thông tin cho admin xem.

***Chức năng Thêm sách:***

* Nhân viên nhập thông tin sách
* Hệ thống nhận thông tin và xác thực thông tin
* Nếu thông tin sách không bị trùng thì thông báo thêm sách thành công, ngược lại nếu sách đã tồn tại sẽ thông báo sách đã tồn tại.

***Chức năng Cập nhật tài khoản:***

* Nhân viên gửi yêu cầu cập nhật cho hệ thống
* Hệ thống kiểm tra thông tin và phản hồi lại nhân viên

***Chức năng Xóa tài khoản:***

* Admin gửi yêu cầu xóa sách đến hệ thông
* Hệ thống kiểm tra thông tin và phản hồi lại nhân viên

2.2.3.4 Biểu đồ trình từ Quản lý sinh viên

A diagram of a workflow

AI-generated content may be incorrect.

Hình 2.13: Biểu đồ trình tự Quản lý sinh viên

2.3 BẢNG CƠ SỞ DỮ LIỆU

2.3.1 Các bảng dữ liệu quan hệ

* **Nguoi\_dung** : Lưu trữ thông tin tài khoản, mật khẩu, vai trò và quyền.
* **Nha\_xuat\_ban** : Lữu trữ thông tin về nhà xuất bản.
* **Sach** : Lưu trữ thông tin sách trong thư viện.
* **Sinh\_vien** : Lưu trữ thông tin sinh viên.
* **Phieu\_muon** : Lưu trữ thông tin sinh viên mượn sách.
* **Phieu\_phat** : Lưu trữ thông tin sinh viên mượn sách quá hạn.

2.3.2 Các bảng dữ liệu quan hệ

* **Nguoi\_dung** (id, username, password, quyen\_han)
* **Nha\_xuat\_ban** (ma\_nxb, ten\_nxb, dia\_chi, sdt)
* **Sach** (ma\_sach, tua\_de, tac\_gia, nam\_xb, gia, so\_luong, ngon\_ngu, tinh\_trang, ma\_nxb)
* **Sinh\_vien** ( ma\_sv, ho\_ten, gioi\_tinh, tuoi, dia\_chi, sdt, email)
* **Phieu\_muon** (id, ma\_sv, ma\_sach, ngay\_muon, ngay\_tra\_du\_kien, ngay\_tra\_thuc\_te, phi\_muon, tinh\_trang\_muon)
* **Phieu\_phat** ( id, phieu\_muon\_id, ngay\_lap, so\_ngay\_qua\_han, tien\_phat)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Khóa | Tên trường | Kiểu dữ liệu | Diễn giải |
| 1 | PK | id | Int(11) | Id người dùng |
| 2 |  | username | Varchar(50) | Tài khoản đăng nhập |
| 3 |  | password | Varchar(255) | Mật khẩu đăng nhập |
| 4 |  | quyen\_han | Enum(“Admin”, “User”) | Phân quyền người vào thư viện |

Bảng 2.8: Bảng nguoi\_dung

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Khóa | Tên trường | Kiểu dữ liệu | Diễn giải |
| 1 | PK | ma\_sach | Varchar(10) | Id người dùng |
| 2 |  | tua\_de | Varchar(255) | Tên tác phẩm |
| 3 |  | tac\_gia | Varchar(255) | Tên tác giả |
| 4 |  | nam\_xb | Year(4) | Năm xuất bản |
| 5 |  | gia | Decimal(10,2) | Giá sách |
| 6 |  | so\_luong | Int(11) | Số lượng |
| 7 |  | ngon\_ngu | Varchar(50) | Ngôn ngữ |
| 8 |  | tinh\_trang | Enum(“Mới”, “Cũ”) | Tình trạng sách |
| 9 | FK | ma\_nxb | Varchar(10) | Mã nhà xuất bản |

Bảng 2.9: Bảng sach

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Khóa | Tên trường | Kiểu dữ liệu | Diễn giải |
| 1 | PK | ma\_sv | Varchar(10) | Mã sinh viên |
| 2 |  | ho\_ten | Varchar(100) | Họ và tên |
| 3 |  | gioi\_tinh | Emun(“Nam”, “Nữ”) | Giới tính |
| 4 |  | tuoi | Int(11) | Tuổi |
| 5 |  | dia\_chi | Varchar(255) | Địa chỉ |
| 6 |  | sdt | Varchar(15) | Số điện thoại |
| 7 |  | email | Varchar(100) | Email |

Bảng 2.10: Bảng sinh\_vien

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Khóa | Tên trường | Kiểu dữ liệu | Diễn giải |
| 1 | PK | ma\_nxb | Varchar(10) | Mã nhà xuất bản |
| 2 |  | Ten\_nxb | Varchar(255) | Tên nhà xuất bản |
| 3 |  | Dia\_chi | Varchar(255) | Địa chỉ |
| 4 |  | sdt | Varchar(20) | Số điện thoại |

Bảng 2.11: Bảng nha\_xuat\_ban

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Khóa | Tên trường | Kiểu dữ liệu | Diễn giải |
| 1 | PK | id | Int(11) | ID |
| 2 | FK | Ma\_sv | Varchar(10) | Mã sinh viên |
| 3 | FK | Ma\_sach | Varchar(10) | Mã sách |
| 4 |  | Ngay\_muon | Date | Ngày mượn sách |
| 5 |  | Ngay\_tra\_du\_kien | Date | Ngày trả sách dự kiến |
| 6 |  | Ngay\_tra\_thuc\_te | Date | Ngày trả sách thực tế |
| 7 |  | Phi\_muon | Decimal(10,2) | Phí mượn sách |
| 8 |  | Tinh\_trang\_muon | Enum(“”) | Tình trạng mượn sách |

Bảng 2.12: Bảng phieu\_muon

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Khóa | Tên trường | Kiểu dữ liệu | Diễn giải |
| 1 | PK | id | Int(11) | ID |
| 2 | FK | Phieu\_muon\_id | Int(11) | Phiếu mượn |
| 3 |  | Ngay\_lap | Date | Ngày lập phiếu phạt |
| 4 |  | So\_ngay\_qua\_han | Int(11) | Số ngày quá hạn |
| 5 |  | Tien\_phat | Decimal(10,2) | Tiền phạt |

Bảng 2.13: Bảng phieu\_phat

2.3.3 Mô hình quan hệ (Database Diagrams)

A screenshot of a computer

AI-generated content may be incorrect.

Hình 2.14: Mô hình quan hệ

CHƯƠNG 3: CÀI ĐẶT VÀ THỬ NGHIỆM

3.1: GIAO DIỆN ĐĂNG NHẬP VÀ ĐĂNG KÝ

3.1.1 Đăng nhập

A screenshot of a computer

AI-generated content may be incorrect.

Hình 3.1: Giao diện đăng nhập

3.1.2 Đăng ký

**A screen shot of a computer

AI-generated content may be incorrect.**

Hình 3.2 Giao diện đăng ký

3.2 GIAO DIỆN QUẢN TRỊ (ADMIN)

3.2.1 Giao diện trang chủ

A screenshot of a phone

AI-generated content may be incorrect.

Hình 3.3 Giao diện trang chủ

3.2.2 Giao diện quản lý

**A screenshot of a computer

AI-generated content may be incorrect.**

Hình 3.4: Giao diện quản lý (admin)

A screenshot of a computer

AI-generated content may be incorrect.

Hình 3.5: Giao diện quản lý người dùng

3.3 GIAO DIỆN NGƯỜI DÙNG (USER)

3.3.1 Giao diện quản lý

A screenshot of a computer

AI-generated content may be incorrect.

Hình 3.6: Giao diện quản lý (user)

3.3.2 Giao diện quản lý sách

**A screenshot of a computer

AI-generated content may be incorrect.**

Hình 3.7: Giao diện quản lý sách

3.3.3 Giao diện quản lý sinh viên

A screenshot of a computer

AI-generated content may be incorrect.

Hình 3.8: Giao diện quản lý sinh viên

3.3.4: Giao diện quản lý mượn trả

A screenshot of a computer

AI-generated content may be incorrect.

Hình 3.9: Giao diện quản lý mượn trả sách

3.3.4.1: Giao diện đăng ký mượn sách

*A screenshot of a computer

AI-generated content may be incorrect.*

Hình 3.10: Giao diện đăng ký mượn sách

3.3.4.2: Giao diện đăng ký mượn sách

A screenshot of a computer

AI-generated content may be incorrect.

Hình 3.11: Giao diện quản lý lịch sử

3.3.5: Giao diện thống kê

A screenshot of a computer

AI-generated content may be incorrect.

Hình 3.12: Giao diện thống kê

3.4 KIỂM THỬ HỆ THỐNG

Kiểm thử hệ thống nhằm đảm bảo rằng phần mềm hoạt động đúng theo yêu cầu đặt ra, phát hiện và sửa chữa lỗi trước khi triển khai chính thức.

**Môi trường kiểm thử:**

* Hệ điều hành: Windows 11
* Trình duyệt: Google Chorme

**Quy trình kiểm thử:**

* Viết kịch bản kiểm thử (Test Case)
* Tiến hành chạy thử chức năng theo kịch bản kiểm thử
* Ghi nhận kết quả thực tế và so sánh với kết quả mong đợi
* Đánh giá Pass/Fail

**Kịch bản kiểm thử (Test Case) cho hệ thống:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ID | Mô tả | Bước thực hiện | Dữ liệu test | Kết quả mong đợi | Kết quả |
| **Đăng nhập** | | | | | |
| 1 | Kiểm tra giao diện màn hình đăng nhập | 1.Truy cập vào hệ thống  2.Ấn “Đăng nhập” |  | -Hiển thị logo và tên hệ thống “Quản lý thư viện”.  -Dòng hiển thị thống kê rõ ràng.  -Ô nhập tài khoản mật khẩu hiển thị phù hợp.  -Button “Đăng nhập”, “Đăng ký”. | Pass |
| 2 | Kiểm tra chức năng đăng nhập | 1.Nhập tài khoản  2.Nhập mật khẩu  3.Ấn “Đăng nhập” | Tài Khoản: TranNamKhanh  Mật Khẩu: 123456 | -Đăng nhập thành công  -Chuyển đến giao diện chính | Pass |
| 3 | Kiểm tra nhập sai tài khoản | 1.Nhập sai tài khoản  2.Nhập mật khẩu  3.Ấn “Đăng nhập” | Tài khoản: NamKhanh  Mật khẩu: 123456 | -Hiển thị “Thông tin tài khoản hoặc mật khẩu không đúng”  -Đăng nhập thất bại | Pass |
| 4 | Kiểm tra nhập sai mật khẩu | 1.Nhập tài khoản  2.Nhập sai mật khẩu  3.Ấn “Đăng nhập” | Tài khoản: TranNamKhanh  Mật khẩu: 123 | -Hiển thị “Thông tin tài khoản hoặc mật khẩu không đúng”  -Đăng nhập thất bại | Pass |
| 5 | Kiểm tra nhập sai tài khoản mật khẩu | 1.Nhập sai tài khoản  2.Nhập sai mật khẩu  3.Ấn “Đăng nhập” | Tài Khoản: TranKhanh  Mật Khẩu: 123456789 | -Hiển thị “Thông tin tài khoản hoặc mật khẩu không đúng”  -Đăng nhập thất bại | Pass |
| **Đăng ký** | | | | | |
| 6 | Kiểm tra đăng ký | 1.Nhập tài khoản  2.Nhập mật khẩu  3. Ấn “Đăng nhập” | Tài Khoản: TranNamKhanh012  Mật khẩu: 123456 | -Hiển thị “Đăng ký thành công”  -Chuyển về giao diện đăng nhập. | Pass |
| 7 | Kiểm tra trùng tên tài khoản đăng ký | 1.Nhập tài khoản  2.Nhập mật khẩu  3. Ấn “Đăng ký” | Tài khoản: TranNamKhanh  Mật Khẩu 123456 | -Hiển thị thông báo: “Tên tài khoản đã tồn tại”.  -Đăng ký thất bại | Pass |

Bảng 3.1 Bảng kịch bản kiểm thử (Test Case) cho hệ thống

KẾT LUẬN VÀ ĐỀ NGHỊ

**1.Kết luận**

Trong bối cảnh công nghệ thông tin ngày càng đóng vai trò trọng yếu trong việc hiện đại hóa và tối ưu hóa các hoạt động quản lý, việc xây dựng một hệ thống quản lý thư viện hiệu quả không chỉ là yêu cầu tất yếu của các cơ sở giáo dục mà còn là một bước tiến quan trọng trong công cuộc chuyển đổi số trong lĩnh vực giáo dục – đào tạo.

Với đề tài “Chương trình quản lý thư viện”, nhóm đã triển khai xây dựng một hệ thống phần mềm với các chức năng cốt lõi như quản lý sách, sinh viên, người dùng và hoạt động mượn trả, từ đó đáp ứng được nhu cầu quản lý thư viện cơ bản tại các trường học hoặc tổ chức giáo dục có quy mô vừa và nhỏ.

Quá trình thực hiện đề tài không chỉ giúp nhóm vận dụng hiệu quả các kiến thức chuyên môn về lập trình hướng đối tượng, xử lý cơ sở dữ liệu với MySQL, kết nối thông qua thư viện JDBC mà còn rèn luyện tư duy hệ thống và kỹ năng làm việc nhóm – những năng lực quan trọng đối với một kỹ sư phần mềm trong tương lai.

Sản phẩm của đề tài đã đáp ứng được các yêu cầu kỹ thuật đề ra, vận hành ổn định, giao diện đơn giản, dễ sử dụng, hỗ trợ người quản lý thư viện thực hiện các nghiệp vụ thường xuyên một cách hiệu quả và chính xác. Tuy nhiên, nhóm cũng nhận thức rõ rằng hệ thống vẫn còn một số hạn chế nhất định như chưa tích hợp tính năng thông báo tự động, chưa hỗ trợ thống kê trực quan, và bảo mật hệ thống còn ở mức cơ bản.

**2.Đề nghị**

Trên cơ sở đó, nhóm xin đưa ra một số đề xuất và định hướng phát triển hệ thống trong tương lai như sau:

* Tích hợp hệ thống thông báo qua email giúp nhắc nhở hạn trả sách và cảnh báo quá hạn.
* Phát triển giao diện người dùng hiện đại hơn, thân thiện với thiết bị di động.
* Bổ sung tính năng thống kê trực quan bằng biểu đồ, phục vụ công tác quản trị và đánh giá hoạt động mượn – trả.
* Nâng cao mức độ bảo mật dữ liệu, phân quyền chi tiết và mã hóa thông tin người dùng.
* Xây dựng hệ thống dạng web-based hoặc tích hợp cloud để phục vụ nhu cầu sử dụng từ xa.

TÀI LIỆU THAM KHẢO

[1]. Database Star, 2022, Database Design for a Library Management System, Channel Database Star Youtube.  
[2]. GeeksforGeeks (thường xuyên cập nhật), Library Management System Project | Software Development, GeeksforGeeks, Trực tuyến.  
[3]. YouTube (thường xuyên cập nhật), Java Swing - Modern Application UI Design, Playlist video, Trực tuyến.  
[4]. OpenAI (thường xuyên cập nhật), ChatGPT Documentation, OpenAI, Trực tuyến.  
[5]. GitHub (thường xuyên cập nhật), GitHub Copilot Documentation, GitHub Docs, Trực tuyến.  
[6]. Google (thường xuyên cập nhật), Gemini AI Documentation, Google, Trực tuyến.

[7]. Edureka (thường xuyên cập nhật), Library Management System Project in Java, Edureka, Trực tuyến.  
[8]. Slideshare (2024), Nghiên cứu quản lý thư viện trường Đại học Sao, Slideshare, Trực tuyến.