

Phát triển ứng dụng Java quản lý thư viện

Ghi chú: Trong quá trình thực hành, chương trình được phép bổ sung thêm lớp và hàm cần thiết, dữ liệu đầu vào và đầu ra của mỗi hàm. Sinh viên được phép cài đặt cấu trúc tùy ý, không nhất thiết tuân theo cấu trúc chương trình trong hướng dẫn. Chương trình có thể sử dụng thêm các thư viện Java bên ngoài nếu cần.

Các phiên bản cần cài đặt

1. Phiên bản dòng lệnh cơ bản:
 - Chức năng quản lý tài liệu (Thêm, Xóa, Sửa, Tìm kiếm).
 - Chức năng quản lý người dùng thư viện (mượn/trả tài liệu, thông tin thành viên).
 - Xử lý các trường hợp lỗi (kiểm tra điều kiện khi mượn tài liệu).
 - Giao diện dòng lệnh đơn giản cho người dùng.
2. Phiên bản giao diện đồ họa:
 - Xây dựng giao diện người dùng bằng Swing hoặc JavaFX.
 - Tích hợp API tra cứu thông tin tài liệu.
 - Sử dụng đa luồng để cải thiện trải nghiệm người dùng.
3. Cải tiến:
 - Chức năng tự sáng tạo (mã QR cho tài liệu, hệ thống gợi ý tài liệu, đánh giá nhận xét tài liệu, v.v.).
 - Cải thiện giao diện người dùng thân thiện và đẹp mắt hơn.

Sinh viên có thể cài đặt một trong hai phiên bản của ứng dụng.

Phiên bản dòng lệnh cơ bản

Phát triển ứng dụng quản lý thư viện phiên bản dòng lệnh với các chức năng đơn giản.

Cài đặt phiên bản dòng lệnh

- Xây dựng giao diện menu tương tác với người dùng, ví dụ như sau:

Welcome to My Application!

[0] Exit

[1] Add Document

[2] Remove Document

[3] Update Document

[4] Find Document

[5] Display Document

[6] Add User

- [7] Borrow Document
- [8] Return Document
- [9] Display User Info

- Người dùng có thể điền các con số tương ứng với STT của thao tác.
- Nếu người dùng nhập một con số không thuộc [0-9] hoặc ký tự không hợp lệ thì in ra màn hình “Action is not supported”.

Các yêu cầu chi tiết về phiên bản dòng lệnh được mô tả như sau:

Xác định các lớp chính và triển khai kế thừa giữa các lớp

- Xác định các lớp cần thiết như Document, User, Library, v.v.
- Định nghĩa các thuộc tính và phương thức chính cho từng lớp, ví dụ: title, author cho lớp Document.
- Xác định mối quan hệ kế thừa giữa các lớp.
- Đảm bảo các lớp con kế thừa đúng thuộc tính và phương thức từ lớp cha.

Sử dụng đúng các nguyên tắc OOP

Áp dụng các nguyên tắc đóng gói, kế thừa, đa hình và trừu tượng hóa vào thiết kế ứng dụng.

- Đóng gói: Sử dụng đúng mức truy cập của các thuộc tính và phương thức của một đối tượng.
- Kế thừa: Cho phép các lớp con kế thừa các thuộc tính và phương thức từ lớp cha.
- Đa hình: Phương thức có thể hoạt động khác nhau tùy theo lớp con. Ví dụ, *printInfo* có thể hiển thị thông tin khác nhau cho Book và Thesis.
- Trừu tượng hóa: Đơn giản hóa ứng dụng bằng cách ẩn đi các chi tiết không cần thiết.

Quản lý tài liệu

- Thêm tài liệu: Thêm tài liệu mới vào danh sách thư viện.
- Xóa tài liệu: Xóa tài liệu khỏi danh sách khi không còn cần thiết.
- Sửa tài liệu: Chỉnh sửa thông tin tài liệu, như số lượng hoặc tác giả.
- Tìm kiếm tài liệu: Tìm kiếm tài liệu theo tên, tác giả, hoặc thể loại.

Quản lý người dùng thư viện

- Mượn tài liệu: Kiểm tra điều kiện trước khi cho mượn, như số lượng tài liệu có sẵn.
- Trả tài liệu: Cập nhật trạng thái sau khi trả tài liệu.
- Thông tin thành viên: Quản lý thông tin cá nhân và trạng thái mượn tài liệu.

Xử lý các trường hợp lỗi, như kiểm tra điều kiện khi mượn tài liệu

- Kiểm tra điều kiện khi mượn tài liệu: Ví dụ, không cho phép mượn tài liệu nếu không có sẵn.
- Thông báo lỗi: Cung cấp thông báo rõ ràng khi xảy ra lỗi, như nhập sai thông tin.

Phiên bản giao diện đồ họa

Tiếp tục phát triển ứng dụng với giao diện đồ họa để nâng cao trải nghiệm người dùng.

Xây dựng giao diện người dùng

- Sử dụng Swing hoặc JavaFX để tạo giao diện.
- Giao diện bao gồm:
 - Danh sách tài liệu: Hiển thị tất cả tài liệu trong thư viện.
 - Thông tin tài liệu: Hiển thị chi tiết khi chọn một cuốn tài liệu.
 - Chức năng quản lý: Nút hoặc menu để thêm, xóa, sửa tài liệu và thành viên.
 - Mượn/Trả tài liệu: Giao diện cho phép thành viên mượn hoặc trả tài liệu.

Tích hợp API tra cứu thông tin tài liệu

- Sử dụng API như [Google Books API](#) để tự động lấy thông tin tài liệu dựa trên ISBN hoặc tiêu đề.
- Khi thêm tài liệu mới, cho phép nhập ISBN và tự động điền các thông tin còn lại.

Sử dụng đa luồng

- Thực hiện các tác vụ nặng như gọi API hoặc truy vấn cơ sở dữ liệu trong luồng riêng.
- Đảm bảo giao diện người dùng không bị treo trong quá trình xử lý.

Cải tiến

Các yêu cầu nâng cao, khuyến khích sự sáng tạo và ứng dụng kiến thức.

Chức năng tự sáng tạo

- Mã QR cho tài liệu:
 - Tạo mã QR chứa thông tin tài liệu để quản lý và kiểm tra nhanh.
 - Cho phép quét mã QR để hiển thị thông tin tài liệu hoặc thực hiện mượn/trả.
- Hệ thống đề xuất gợi ý tài liệu:
 - Gợi ý tài liệu cho thành viên dựa trên lịch sử mượn hoặc thể loại yêu thích.

- Sử dụng thuật toán đơn giản hoặc tích hợp hệ thống gợi ý phức tạp hơn.
- Đánh giá và nhận xét tài liệu:
 - Cho phép thành viên đánh giá tài liệu theo thang điểm.
 - Thành viên có thể viết nhận xét và xem nhận xét của người khác.

Cải thiện giao diện người dùng

- Tham khảo [Design System](#)
- Thiết kế giao diện thân thiện và dễ sử dụng.
- Sử dụng màu sắc, biểu tượng và bố cục hợp lý để tăng trải nghiệm.

Đặt tên biến, phương thức, và lớp đúng và dễ hiểu

- Tuân thủ các quy tắc đặt tên trong lập trình Java.
- Sử dụng tiếng Anh cho tên biến, phương thức và lớp.

Sử dụng các nguyên tắc coding convention

- Tham khảo [Google Java Style Guide](#)
- Tuân thủ chuẩn code về thụt lề, khoảng trắng và cấu trúc mã.
- Thêm chú thích (comments) cho các đoạn code phức tạp.

Trình bày và hỏi đáp

- Trình bày các tính năng chính, kiến trúc hệ thống và các kỹ thuật đã sử dụng.
- Sẵn sàng trả lời các câu hỏi về thiết kế, mã nguồn.

Chấm điểm

Yêu cầu các nhóm cung cấp mã nguồn trên git và file chạy.

Thực hiện tất cả các yêu cầu bắt buộc để được đánh giá bài tập lớn.

Yêu cầu	Điểm	Mức
Thiết kế lớp và cây kế thừa		
Xác định các lớp chính và triển khai kế thừa giữa các lớp	2.5	Bắt buộc
Sử dụng đúng các nguyên tắc OOP	0.5	Bắt buộc
Chức năng chính của ứng dụng		
Quản lý tài liệu (Thêm, Xóa, Sửa, Tìm kiếm)	1	Bắt buộc
Quản lý người dùng thư viện (mượn/trả tài liệu, thông tin thành viên)	1	Bắt buộc
Xử lý các trường hợp lỗi, như kiểm tra điều kiện khi mượn tài liệu	0.5	Bắt buộc
Giao diện người dùng	0.5	Tự chọn
Tích hợp API tra cứu thông tin tài liệu	0.5	Tự chọn
Sử dụng đa luồng để cải thiện trải nghiệm người dùng	0.5	Tự chọn
Chức năng tự sáng tạo (mã QR cho tài liệu, Hệ thống đề xuất gợi ý tài liệu, đánh giá nhận xét tài liệu, v.v)	1	Tự chọn
Chất lượng code và quy tắc lập trình		
Đặt tên biến, phương thức, và lớp đúng và dễ hiểu	0.5	Tự chọn
Sử dụng các nguyên tắc coding convention	0.5	Tự chọn
Cài đặt Unit Test bằng JUnit	0.5	Tự chọn
Sử dụng Design Pattern vào ứng dụng	0.5	Tự chọn
Trình bày + hỏi đáp	1	Bắt buộc
Tổng điểm	10 + 1	