**BỘ CÔNG THƯƠNG**

**TRƯỜNG ĐẠI HỌC CÔNG THƯƠNG TP. HỒ CHÍ MINH**

**KHOA CÔNG NGHỆ THÔNG TIN**

-----□[□-----

A blue and white logo with a book and a graduation cap

AI-generated content may be incorrect.

**ĐỀ TÀI: ỨNG DỤNG QUẢN LÍ SÁCH THƯ VIỆN**

**NHÓM : 16**

**GVHD: Đinh Nguyễn Trọng Nghĩa**

1. Nguyễn Thị Ngọc Hiểu - MSSV: 2033230081

2. Lâm Văn Tiến - MSSV: 2001230808

**Thành phố Hồ Chí Minh, tháng 6 năm 2025**

**Mục Lục**

1. Giới thiệu đề tài
2. Mục tiêu
3. Phạm vi và đối tượng sử dụng
4. Công nghệ và công cụ
5. Phân tích yêu cầu
6. Thiết kế hệ thống
7. Cài đặt và triển khai
8. Kiểm thử và đánh giá
9. Kết luận
10. Tài liệu tham khảo

**1.Giới thiệu đề tài**

* **Tổng quan về đề tài:**
* **Mục đích:**

- **Tự động hóa quy trình quản lý thư viện**: Giúp thay thế các thao tác thủ công như ghi chép, tra cứu, thống kê bằng phần mềm, nâng cao hiệu suất làm việc.

- **Quản lý thông tin sách và người dùng hiệu quả**: Lưu trữ, tra cứu và cập nhật thông tin sách, người mượn/trả sách một cách nhanh chóng, chính xác.

- **Hỗ trợ nghiệp vụ thư viện**: Như mượn - trả sách, theo dõi tình trạng sách, hạn trả, v.v.

- **Phân quyền người dùng**: Cho phép người quản trị và người dùng thường sử dụng hệ thống theo đúng chức năng được cấp.

* **Ý nghĩa:**

- **Tiết kiệm thời gian và công sức** cho thủ thư và người mượn sách.

- **Giảm thiểu sai sót** trong việc ghi nhận, thống kê và tra cứu thông tin.

- **Tăng cường khả năng truy xuất dữ liệu**: Cho phép tìm kiếm sách hoặc thông tin người dùng một cách nhanh chóng và chính xác.

- **Hỗ trợ chuyển đổi số** trong lĩnh vực giáo dục và thư viện.

- **Tạo môi trường hiện đại và chuyên nghiệp** trong công tác quản lý, thúc đẩy việc đọc và mượn sách hiệu quả hơn.

**2. Mục tiêu của ứng dụng quản lý thư viện:**

1. **Quản lý sách hiệu quả**

- Cho phép thêm, sửa, xoá, và tra cứu sách.

- Theo dõi tình trạng sách: còn/mượn/hết.

1. **Quản lý người dùng thư viện**

- Quản lý thông tin người dùng (họ tên, mã người dùng, loại tài khoản…).

- Phân quyền người dùng: quản trị và người dùng thường.

1. **Quản lý quá trình mượn – trả sách**

- Ghi nhận thông tin mượn và trả sách.

- Tính toán và cảnh báo sách mượn quá hạn.

- Ghi nhận tiền phạt (nếu có).

1. **Giao diện thân thiện, dễ sử dụng**

- Giao diện trực quan, đơn giản, phù hợp với người dùng không chuyên.

- Có thông báo rõ ràng và phản hồi khi thực hiện thao tác.

1. **Lưu trữ dữ liệu an toàn và ổn định**

- Dữ liệu người dùng, sách, lịch sử mượn – trả được lưu trữ lâu dài (ví dụ: trong file JSON hoặc CSDL).

- Đảm bảo tính toàn vẹn của dữ liệu khi thao tác.

1. **Tìm kiếm và thống kê**

- Hỗ trợ tìm kiếm sách theo tên, tác giả, thể loại…

- Thống kê số lượng sách, số lần mượn, sách đang được mượn nhiều nhất, v.v.

**3. Phạm vi và đối tượng sử dụng:**

1. **Người quản trị thư viện (Admin):**

- Có toàn quyền sử dụng hệ thống.

- Thực hiện thêm/sửa/xoá sách, quản lý người dùng, xử lý mượn – trả, thống kê các sách yêu thích.

1. **Người dùng thường (User):**

- Có thể tra cứu sách, thực hiện thao tác mượn – trả theo quyền hạn được cấp, xem được thống kế các sách yêu thích.

1. **Giáo viên / thủ thư / cán bộ thư viện:**

- Sử dụng hệ thống để hỗ trợ công việc quản lý sách trong trường học hoặc cơ quan.

1. **Học sinh / sinh viên / nhân viên:**

- Sử dụng hệ thống để tìm kiếm và mượn sách một cách nhanh chóng, tiện lợi.

**4. Công nghệ và công cụ:**

- Ngôn ngữ lập trình: Python phiên bản 3.13.3

- Thư viện sử dụng: Thư viện chuẩn (Standard Libraries)

- Các công cụ hỗ trợ: IDE (Visual Studio Code, PyCharm)

- Các API sử dụng: Không sử dụng

1. **Phân tích yêu cầu:**

* **Chức năng của hệ thống:**

#### **Create (Tạo mới) :**

**-** Thêm sách mới vào thư viện:

+ Nhập thông tin: mã sách, tên sách, tác giả, thể loại, năm xuất bản...

- Thêm người dùng mới:

+ Tạo tài khoản cho người dùng: họ tên, tên đăng nhập, mật khẩu, phân quyền (admin/user).

* **Read (Xem / Tra cứu):**

**-** Xem danh sách sách trong thư viện:

+ Hiển thị các sách hiện có, tình trạng (còn/mượn/hết).

- **Tra cứu thông tin người dùng:**

**+** Xem danh sách tài khoản, vai trò, lịch sử mượn trả.

- Tìm kiếm nâng cao:

+ Tìm theo tên sách, tác giả, thể loại, mã sách...

* **Update (Cập nhật):**

- Sửa thông tin sách:

+ Cập nhật tên, tác giả, số lượng, tình trạng…

- Cập nhật thông tin người dùng:

+ Đổi mật khẩu, tên, quyền hạn (nếu là admin).

- Cập nhật trạng thái mượn – trả:

+ Khi người dùng trả sách, hệ thống cập nhật ngày trả

* **Delete (Xoá):**

- Xoá sách khỏi hệ thống:

+ Xoá khi sách không còn sử dụng hoặc sai thông tin (yêu cầu quyền admin).

- Xoá người dùng:

+ Khi người dùng không còn hoạt động

* **Yêu cầu kỹ thuật:**

**1. Yêu cầu về giao diện người dùng (User Interface - UI)**

**- Thân thiện, dễ sử dụng:**

+ Giao diện đơn giản, dễ hiểu cho người dùng không chuyên, hỗ trợ tiếng Việt.

**- Thiết kế rõ ràng, trực quan:**

+ Các chức năng được bố trí hợp lý: Quản lý sách, người dùng, mượn – trả sách.

+ Có menu, nút điều hướng, và thông báo thao tác rõ ràng.

**- Phản hồi người dùng:**

+ Hiển thị thông báo khi thao tác thành công/thất bại (VD: “Thêm sách thành công”, “Không tìm thấy người dùng”...).

**- Đăng nhập và phân quyền:**

+ Giao diện đăng nhập với hai loại tài khoản: **Quản trị viên** và **Người dùng thường**.

+ Mỗi loại tài khoản chỉ truy cập được các chức năng tương ứng.

**- Công nghệ sử dụng:**

+ Ngôn ngữ: **Python**

**+** Giao diện: **Tkinter** (thư viện GUI chuẩn của Python)

**2. Yêu cầu về quản lý dữ liệu**

**- Lưu trữ dữ liệu ổn định và có thể mở rộng:**

+ Dữ liệu được lưu dưới dạng **file JSON** (hoặc có thể nâng cấp lên cơ sở dữ liệu như SQLite).

**- Cấu trúc dữ liệu rõ ràng:**

+ Các thông tin sách, người dùng, và lịch sử mượn – trả được lưu riêng biệt theo cấu trúc phân cấp.

**- Tương tác dữ liệu an toàn:**

+ Kiểm tra đầu vào, tránh trùng lặp mã sách/người dùng.

- Đảm bảo các thao tác CRUD (Tạo mới – Đọc – Cập nhật – Xoá) không gây lỗi hệ thống.

**- Phân quyền dữ liệu:**

+ Người dùng thường không được phép sửa, xoá, thêm dữ liệu hệ thống.

+ Chỉ quản trị viên được cấp quyền truy cập đầy đủ.

1. **Thiết kế hệ thống:**

* **Kiến trúc hệ thống:**

**Mô hình kiến trúc:**

- Ứng dụng được xây dựng theo mô hình 3 lớp (Three-layer Architecture) đơn giản:

**1. Giao diện người dùng (UI Layer)**

* **Công nghệ:** Tkinter (Python GUI)

**- Chức năng:**

+ Giao diện nhập liệu: thêm sách, mượn – trả, đăng nhập…

+ Hiển thị thông tin sách, người dùng, lịch sử mượn – trả

+ Điều hướng người dùng tới các chức năng chính

**2. Lớp xử lý nghiệp vụ (Business Logic Layer)**

**- Chức năng:**

+ Kiểm tra dữ liệu nhập vào (VD: trùng mã sách, trống thông tin…)

+ Thực hiện các chức năng: thêm/sửa/xoá/tìm kiếm sách, người dùng

+ Tính toán logic: kiểm tra sách còn/mượn

**3. Lớp dữ liệu (Data Layer)**

**- Công nghệ:** File JSON

**- Chức năng:**

+ Lưu trữ thông tin sách, người dùng, lịch sử mượn – trả

+ Đọc/ghi dữ liệu từ các file .json

+ Dữ liệu được phân loại:

* books.json – danh sách sách
* users.json – tài khoản người dùng
* borrow\_history.json – lịch sử mượn – trả
* Thiết kế giao diện:

**Giao diện đăng nhập :**

A screenshot of a computer

Description automatically generated

**Giao diện admin sau khi đăng nhập:**

A screenshot of a computer

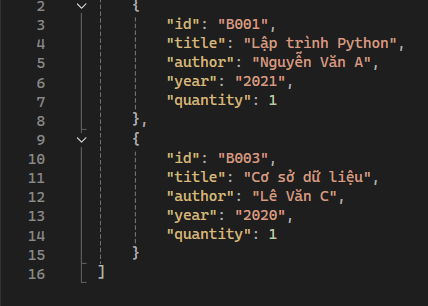
Description automatically generated

**Giao diện users sau khi đăng nhập:**

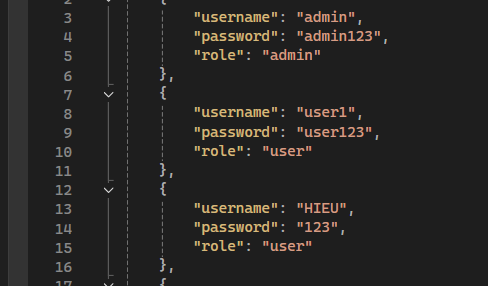
A screenshot of a computer

Description automatically generated

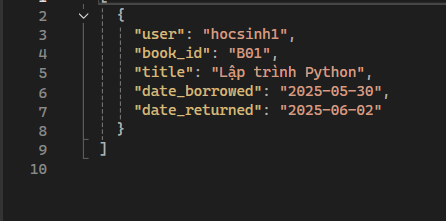
* Cấu trúc dữ liệu – Lưu trữ bằng JSON
* Ứng dụng lưu trữ dữ liệu dưới dạng **file JSON** để đảm bảo dễ đọc, dễ xử lý bằng Python, và không cần dùng đến hệ quản trị cơ sở dữ liệu (SQL).
* **Cấu trúc lưu trữ sách – books.json**



* **Cấu trúc lưu trữ người dùng – users.json**



* **Cấu trúc lưu trữ lịch sử mượn - Trả – borrow\_history.json**



1. **Cài đặt và triển khai**

* **Quá trình phát triển ứng dụng**
* **Các bước triển khai:**

|  |  |  |
| --- | --- | --- |
| **STT** | **Bước thực hiện** | **Mô tả chi tiết** |
| 1 | **Phân tích yêu cầu** | Xác định chức năng cần thiết: quản lý sách, người dùng, mượn – trả, phân quyền |
| 2 | **Thiết kế giao diện (UI)** | Sử dụng Tkinter để tạo các cửa sổ nhập liệu, nút bấm, bảng danh sách... |
| 3 | **Thiết kế cấu trúc dữ liệu** | Xác định cấu trúc file JSON: sách, người dùng, mượn trả |
| 4 | **Xây dựng chức năng nghiệp vụ** | Viết hàm xử lý thêm/xoá/sửa/tìm kiếm dữ liệu, đăng nhập, phân quyền |
| 5 | **Liên kết UI với logic xử lý** | Kết nối các nút trên giao diện với chức năng tương ứng |
| 6 | **Kiểm thử và sửa lỗi (debug)** | Thử nghiệm các thao tác CRUD, kiểm tra dữ liệu lưu đúng vào JSON |
| 7 | **Hoàn thiện và đóng gói báo cáo** | Tổng hợp báo cáo, hướng dẫn sử dụng, chuẩn bị trình bày |

* **Mô tả mã nguồn**

|  |  |
| --- | --- |
| **Main.py** | **Khởi chạy ứng dụng**  **Login, phân Quyền** |
| **Thuvien.py** | **CRUD cho sách** |
| **Users.json** | **Dữ liệu** |
| **Books.json** | **Dữ liệu** |
| **Borrow\_history.json** | **Dữ liệu** |

* **Hướng dẫn sử dụng ứng dụng**
* **Các bước sử dụng cơ bản:**

**- Chạy ứng dụng:**

+ Mở terminal hoặc IDE và chạy main.py

+ Giao diện đăng nhập xuất hiện

**- Đăng nhập (ADMIN):**

+ Tài khoản mẫu:

+ Tên: admin, Mật khẩu: admin123

+ Hoặc dùng tài khoản người dùng thường

**- Chức năng sau khi đăng nhập (ADMIN)**

**+ Quản lý sách**

+ Thêm, sửa, xoá sách

**+ Quản lý người dùng** (admin)

+ Tạo tài khoản, sửa đổi quyền

**- Mượn – trả sách**

+ Ghi nhận mượn sách, cập nhật trạng thái khi trả

**- Tìm kiếm và thống kê:**

+ Tìm sách theo tên, mã sách

+ Xem danh sách sách đang mượn, quá hạn, đã trả...

1. **Kiểm thử và đánh giá**

* **Quá trình kiểm thử:**
* **Các bước kiểm thử chính:**

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Chức năng được kiểm thử** | **Nội dung kiểm thử** | **Kết quả mong đợi** |
| 1 | Đăng nhập | Nhập đúng/sai tài khoản và mật khẩu | Đăng nhập thành công hoặc báo lỗi |
| 2 | Thêm sách mới | Nhập đầy đủ thông tin, bấm nút “Thêm” | Sách được thêm vào file books.json |
| 3 | Tìm kiếm sách | Tìm theo mã sách, tên sách | Hiển thị thông tin tương ứng |
| 4 | Sửa thông tin sách | Chỉnh sửa tên sách, số lượng, tác giả... | Dữ liệu được cập nhật đúng |
| 5 | Xoá sách | Chọn sách và xoá | Sách bị loại khỏi danh sách và file JSON |
| 6 | Mượn sách | Chọn sách, bấm “Mượn” | Ghi nhận mượn, giảm số lượng sách khả dụng |
| 7 | Trả sách | Chọn sách đang mượn, bấm “Trả” | Cập nhật trạng thái và tăng số lượng lại |
| 8 | Tạo tài khoản người dùng | Chỉ cho phép admin tạo tài khoản mới | Tài khoản được thêm vào users.json |
| 9 | Phân quyền truy cập | Kiểm tra người dùng thường bị giới hạn chức năng | Người dùng thường không truy cập chức năng admin |
| 10 | Lưu dữ liệu vào file JSON | Sau mỗi thao tác CRUD | File JSON được cập nhật chính xác |

* **Kết quả kiểm thử :**

|  |  |  |
| --- | --- | --- |
| **Chức năng kiểm thử** | **Trạng thái** | **Ghi chú** |
| Đăng nhập | Đạt | Đúng quyền, phát hiện sai mật khẩu |
| Thêm sách mới | Đạt | Dữ liệu ghi đúng định dạng JSON |
| Tìm kiếm sách | Đạt | Tìm được chính xác theo từ khóa |
| Cập nhật sách | Đạt | Kiểm tra lại trong JSON đúng dữ liệu |
| Xoá sách | Đạt | Sách bị xoá hoàn toàn khỏi danh sách |
| Mượn sách | Đạt | Kiểm tra trạng thái mượn và giảm số lượng |
| Trả sách | Đạt | Trạng thái chuyển về “returned”, số lượng tăng |
| Tạo người dùng | Đạt | Chỉ tài khoản admin có quyền |
| Phân quyền | Đạt | Người dùng thường bị giới hạn đúng chức năng |

1. **Kết luận**

* **Tóm tắt kết quả đạt được**
* **Sau quá trình thiết kế và triển khai, ứng dụng Quản lý thư viện đã được xây dựng thành công với các đặc điểm sau:**
* Giao diện người dùng trực quan bằng **Tkinter**, dễ thao tác với người dùng phổ thông.
* Quản lý **sách, người dùng, mượn – trả** hiệu quả thông qua các chức năng **CRUD (Tạo, Đọc sách, Thêm sách, Xóa sách)**.
* Dữ liệu được lưu trữ bằng **file JSON**, đảm bảo tính đơn giản, nhẹ và dễ chỉnh sửa.
* Hệ thống **phân quyền người dùng** giúp phân biệt giữa **quản trị viên** và **người dùng thường**.
* Hoàn thành quá trình **kiểm thử thủ công**, tất cả chức năng hoạt động đúng như thiết kế, không phát hiện lỗi nghiêm trọng.
* **Ứng dụng đáp ứng đúng yêu cầu ban đầu và có thể sử dụng cho các mô hình thư viện quy mô nhỏ như: thư viện lớp học, câu lạc bộ, trung tâm học tập…**
* **Hướng phát triển trong tương lai**
* **Để nâng cao tính ứng dụng thực tế và mở rộng quy mô, có thể phát triển thêm các chức năng sau:**
* Gửi thông báo : Gửi email tự động khi sách gần đến hạn trả
* Thống kê : Thống kê sách mượn nhiều, sách chưa trả, người dùng mượn nhiều nhất
* **Đa nền tảng :** Chuyển sang web app (Flask/Django) hoặc mobile app (Kivy, React Native)

1. **Tài liệu tham khảo**

Giao diện người dung GUI với tkinter: <https://docs.python.org/3/library/tkinter.html>

Xử lí dữ liệu lưu trữ:

<https://docs.python.org/3/library/json.html>

1. **Phụ lục:**

Mã nguồn (Link GitHub): <https://github.com/ngochieu17/thuvien.guthub.git>