**MỤC LỤC**

[I. MỞ ĐẦU 1](#_Toc185182210)

[1.1. Đặt vấn đề 1](#_Toc185182211)

[1.2. Mục tiêu 1](#_Toc185182212)

[1.3. Các chức năng dự kiến 1](#_Toc185182213)

[1.4. Đối tượng hướng đến 2](#_Toc185182214)

[1.5. Phương pháp thực hiện 2](#_Toc185182217)

[II. PHÂN TÍCH THIẾT KẾ 3](#_Toc185182218)

[2.1. Màn hình chính 3](#_Toc185182220)

[2.2. Quản lý độc giả 4](#_Toc185182221)

[2.2.1 Quản lý độc giả 4](#_Toc185182222)

[2.2.2. Quản lý loại độc giả 6](#_Toc185182223)

[2.3. Quản lý sách 7](#_Toc185182224)

[2.3.1. Quản lý sách 7](#_Toc185182225)

[2.3.2. Quản lý thể loại 8](#_Toc185182226)

[2.3.3. Quản lý tác giả 10](#_Toc185182227)

[2.4. Quản lý tài khoản 11](#_Toc185182228)

[2.5. Quản lý mượn, trả sách 13](#_Toc185182229)

[2.5.1. Quản lý mượn sách 13](#_Toc185182230)

[2.5.2. Quản lý trả sách 14](#_Toc185182231)

[III. QUÁ TRÌNH THỰC HIỆN VÀ MÃ NGUỒN 16](#_Toc185182232)

[3.1. Xây dựng dữ liệu 16](#_Toc185182233)

[3.2. Cài đặt và xây dựng giao diện bằng Winform C# 17](#_Toc185182234)

[3.2.1. Cài đặt Winform 17](#_Toc185182235)

[3.2.2. Xây dựng giao diện mã nguồn 17](#_Toc185182236)

[3.3. Xây dựng xử lý sự kiện tiêu biểu 17](#_Toc185182237)

[3.3.1. Hiển thị các màn hình quản lý khi nhấn nút 17](#_Toc185182238)

[3.3.2. Kết nối database và sự kiện thêm, xóa, sửa, mặc định, tìm kiếm 17](#_Toc185182239)

[3.3.3. Chức năng mượn trả sách 17](#_Toc185182240)

[IV. KIỂM THỬ VÀ KẾT QUẢ 18](#_Toc185182241)

[4.1. Kết quả đạt được 18](#_Toc185182242)

[4.2. Hạn chế 19](#_Toc185182243)

[V. KẾT LUẬN 20](#_Toc185182244)

[VI. TÀI LIỆU THAM KHẢO 21](#_Toc185182245)

[6.1. Sách 21](#_Toc185182246)

[6.2. Trang web 21](#_Toc185182247)

[VII. LỜI CẢM ƠN 22](#_Toc185182248)

# I. MỞ ĐẦU

## 1.1. Đặt vấn đề

Trong thời đại số hóa hiện nay, việc sử dụng phần mềm quản lý là một phần không thể thiếu trong hoạt động của các thư viện. Phần mềm quản lý thư viện giúp tối ưu hóa quy trình quản lý sách, mượn trả, lưu trữ thông tin độc giả và kiểm soát hoạt động của thư viện một cách hiệu quả. Đề tài này tập trung vào việc xây dựng một ứng dụng quản lý thư viện sử dụng ngôn ngữ lập trình C#. Thông qua việc phát triển ứng dụng này, người lập trình có cơ hội nắm vững quy trình thiết kế và triển khai một phần mềm quản lý trên nền tảng desktop, đồng thời đảm bảo tính khả thi và hữu ích trong thực tế.

## 1.2. Mục tiêu

Mục tiêu của đề tài là xây dựng một ứng dụng quản lý thư viện sách đơn giản, thân thiện và dễ sử dụng. Ứng dụng sẽ hỗ trợ các tính năng quản lý thông tin sách, độc giả, và quá trình mượn trả sách. Cụ thể, ứng dụng được thiết kế với giao diện đồ họa trực quan bằng ngôn ngữ lập trình C#, giúp người quản lý thư viện và nhân viên vận hành thuận tiện. Các tính năng chính bao gồm: lưu trữ thông tin sách và độc giả, quản lý quá trình mượn trả sách, theo dõi trạng thái sách, thống kê hoạt động, và hỗ trợ xuất báo cáo khi cần thiết.

## 1.3. Các chức năng dự kiến

* Quản lý thông tin sách: Cho phép thêm, sửa, xóa thông tin về sách, bao gồm tên sách, tác giả, thể loại, nhà xuất bản, năm xuất bản, và số lượng.
* Quản lý thông tin độc giả: Lưu trữ thông tin cá nhân của độc giả như họ tên, địa chỉ, số điện thoại, giới tính, ngày lập thẻ và ngày hết hạn.
* Quản lý tài khoản: Hỗ trợ tạo, sửa, và xóa tài khoản cho quản trị viên hoặc nhân viên. Bao gồm tính năng phân quyền sử dụng, giúp đảm bảo an toàn và bảo mật thông tin trong hệ thống.
* Quản lý mượn trả sách: Hỗ trợ ghi nhận thông tin mượn sách, bao gồm tên sách, tên độc giả, ngày mượn và ngày dự kiến trả. Đồng thời, cập nhật trạng thái sách sau khi trả.
* Tìm kiếm và tra cứu: Cho phép tìm kiếm thông tin sách hoặc độc giả dựa trên tiêu chí như mã sách, tên sách, hoặc số điện thoại độc giả.

## 1.4. Đối tượng hướng đến

## Thư viện nhỏ và vừa: Các thư viện có quy mô nhỏ hoặc trung bình cần một giải pháp quản lý hiệu quả, dễ sử dụng và chi phí thấp để quản lý sách và độc giả.

## Người học lập trình C#: Những cá nhân hoặc nhóm muốn học cách phát triển ứng dụng quản lý trên nền tảng desktop bằng ngôn ngữ lập trình C#, đồng thời nâng cao kỹ năng lập trình và thiết kế giao diện đồ họa.

## 1.5. Phương pháp thực hiện

* Xây dựng giao diện bằng C# và Windows Forms/WPF: Thiết kế giao diện trực quan với các cửa sổ, ô nhập dữ liệu, nút lệnh, danh sách hiển thị thông tin, và các bảng điều khiển để người dùng dễ dàng thao tác.
* Lập trình xử lý sự kiện: Xây dựng logic cho các thao tác trên giao diện như lưu, xóa, cập nhật, tìm kiếm, và các sự kiện liên quan đến tương tác giữa các thành phần giao diện và cơ sở dữ liệu.
* Kết nối cơ sở dữ liệu: Sử dụng cơ sở dữ liệu (ví dụ: SQL Server hoặc SQLite) để lưu trữ thông tin về sách, độc giả, tài khoản người dùng, và lịch sử mượn trả sách. Xây dựng các truy vấn SQL để xử lý thêm, sửa, xóa, và tìm kiếm dữ liệu.
* Kiểm thử và tối ưu hóa: Kiểm tra tính chính xác và hiệu quả của ứng dụng qua các tình huống sử dụng thực tế, đồng thời cải thiện hiệu suất và trải nghiệm người dùng.

# II. PHÂN TÍCH THIẾT KẾ

## Giao diện người dùng của ứng dụng quản lý thư viện được thiết kế để đảm bảo tính thân thiện, dễ sử dụng và trực quan. Giao diện kết hợp giữa tính thẩm mỹ và hiệu suất nhằm hỗ trợ người dùng thao tác nhanh chóng và hiệu quả trong việc quản lý thông tin về sách, độc giả, tài khoản và quá trình mượn trả sách.

## 2.1. Màn hình chính

Màn hình chính là trung tâm điều hành của ứng dụng, là nơi có thể truy cập các tính năng quan trọng và xem thông tin tổng quan về ứng dụng quản lý khách sạn.

1. Menu bên trái:

Các mục được sắp xếp theo nhóm chức năng với biểu tượng minh họa, giúp người dùng dễ dàng nhận diện và truy cập:

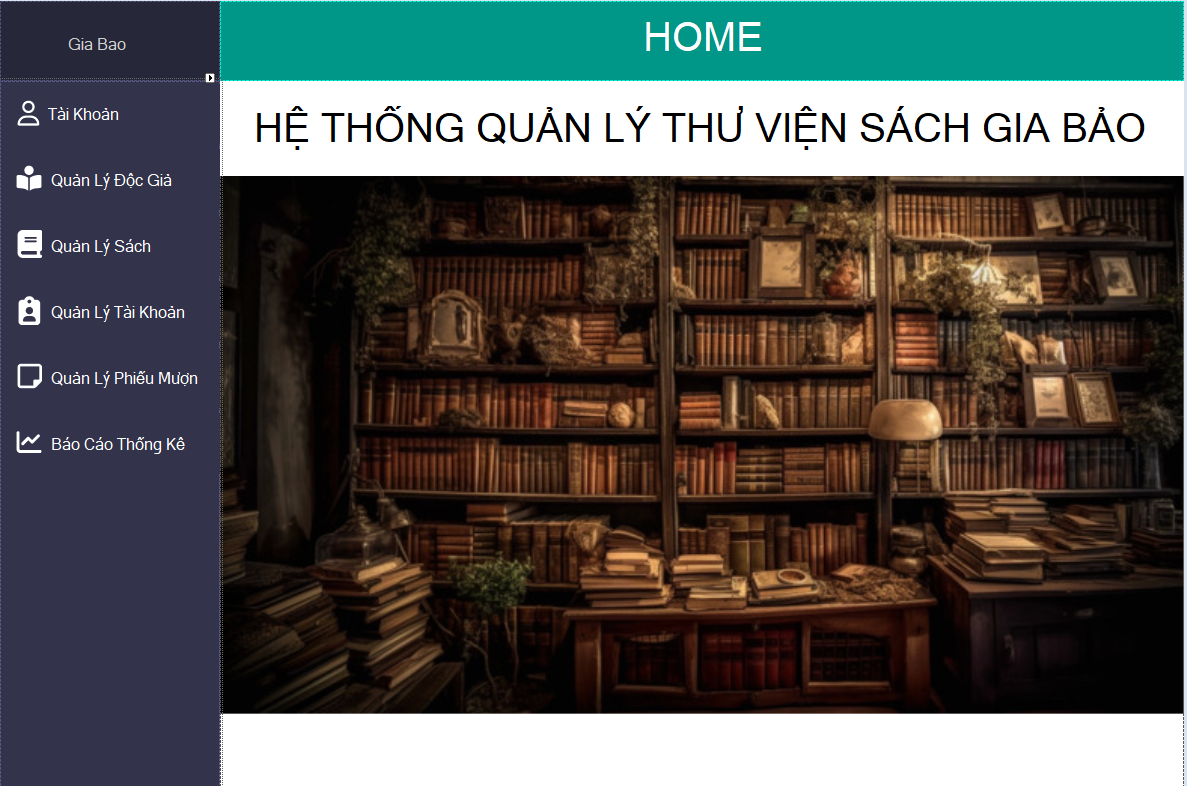
* Tài Khoản: Quản lý thông tin người dùng.
* Quản Lý Độc Giả: Thao tác với thông tin độc giả.
* Quản Lý Sách: Thực hiện quản lý sách trong thư viện.
* Quản Lý Tài Khoản: Cấu hình, thêm hoặc xóa tài khoản nhân viên quản trị.
* Quản Lý Phiếu Mượn: Xử lý thông tin mượn trả sách.
* Báo Cáo Thống Kê: Xem báo cáo về hoạt động của thư viện.

1. Phần tiêu đề:

Hiển thị logo hoặc tiêu đề lớn “HỆ THỐNG QUẢN LÝ THƯ VIỆN SÁCH GIA BẢO”, tạo điểm nhấn cho giao diện và truyền tải tên hệ thống.

1. Hình ảnh minh họa:

Khu vực trung tâm màn hình hiển thị hình ảnh giá sách, tạo cảm giác gần gũi, phù hợp với chủ đề quản lý thư viện.



*Hình 2.1. Màn hình chính*

## 2.2. Quản lý độc giả

### 2.2.1 Quản lý độc giả

1. Thanh tìm kiếm:

* Người dùng có thể dễ dàng tìm kiếm độc giả thông qua ô tìm kiếm ở phía trên cùng, giúp truy xuất nhanh các thông tin liên quan đến mã độc giả, tên độc giả hoặc các thông tin khác.
* Nút làm mới giúp hiển thị lại toàn bộ danh sách tài khoản khi cần.

1. Danh sách độc giả:

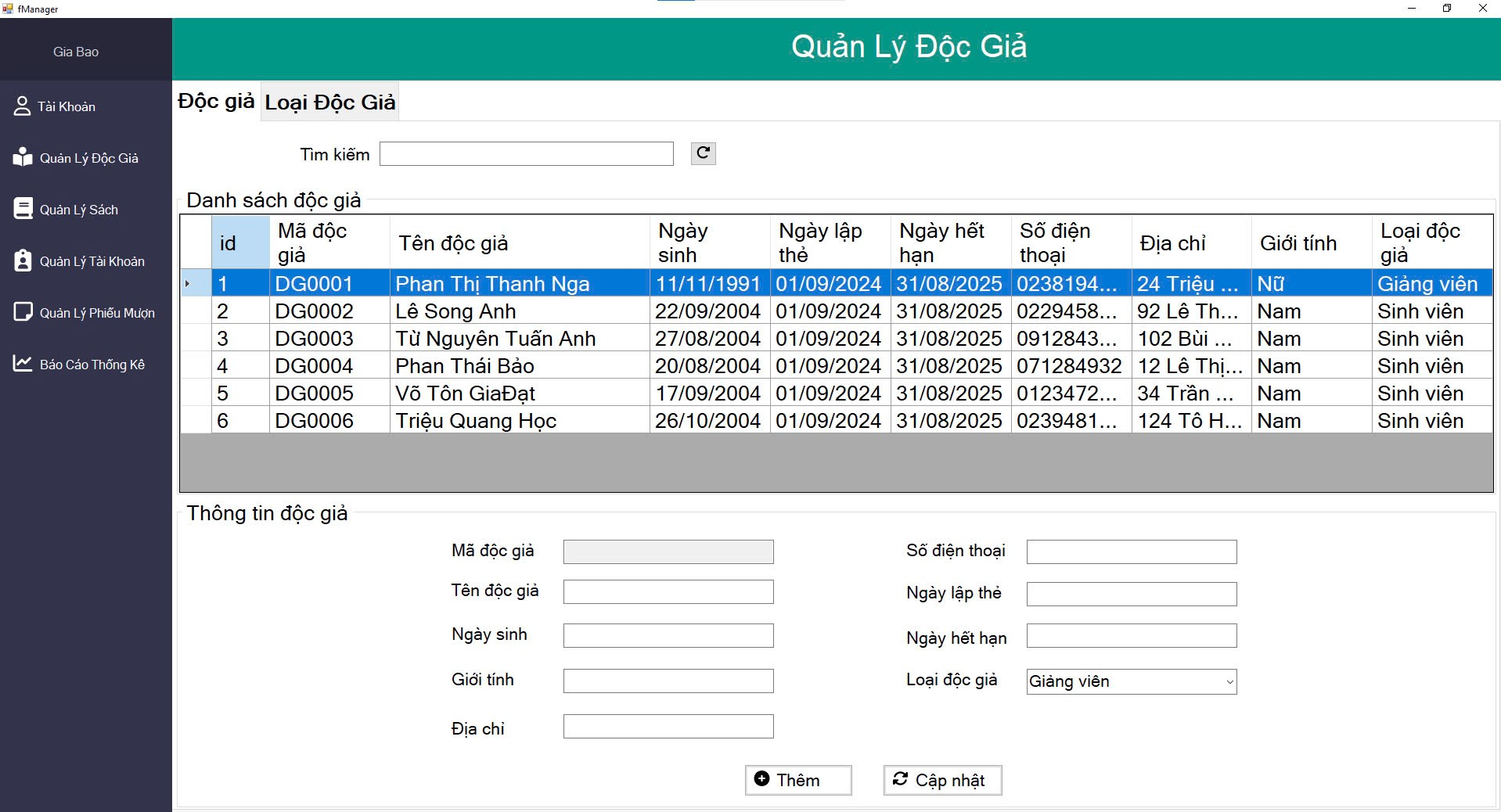
* Các thông tin cơ bản của từng độc giả được hiển thị trong bảng gồm các cột như: mã độc giả, tên độc giả, ngày sinh, ngày lập thẻ, ngày hết hạn, số điện thoại, địa chỉ, giới tính và loại độc giả.
* Bảng này cho phép người dùng xem và thao tác với các độc giả hiện có trong hệ thống, dễ dàng quản lý thông tin.

1. Thông tin chi tiết độc giả:

* Khi người dùng chọn một độc giả trong danh sách, các thông tin chi tiết của độc giả đó sẽ được hiển thị ở phía dưới, gồm các trường như: mã độc giả, tên, ngày sinh, giới tính, địa chỉ, số điện thoại, ngày lập thẻ, ngày hết hạn và loại độc giả.
* Các ô nhập liệu này sẽ cho phép người dùng thêm mới hoặc cập nhật thông tin của độc giả.

1. Nút thao tác:

* Thêm: Cho phép người dùng thêm mới thông tin độc giả vào hệ thống.
* Cập nhật: Người dùng có thể sửa đổi các thông tin của độc giả đã có trong hệ thống.



*Hình 2.2.1. Màn hình quản lý độc giả*

### 2.2.2. Quản lý loại độc giả

1. Thanh tìm kiếm:

* Người dùng có thể dễ dàng tìm kiếm loại độc giả thông qua ô tìm kiếm ở phía trên cùng, giúp truy xuất nhanh các thông tin liên quan đến mã loại độc giả, tên loại độc giả hoặc các thông tin khác.
* Nút làm mới giúp hiển thị lại toàn bộ danh sách tài khoản khi cần.

1. Danh sách loại độc giả:

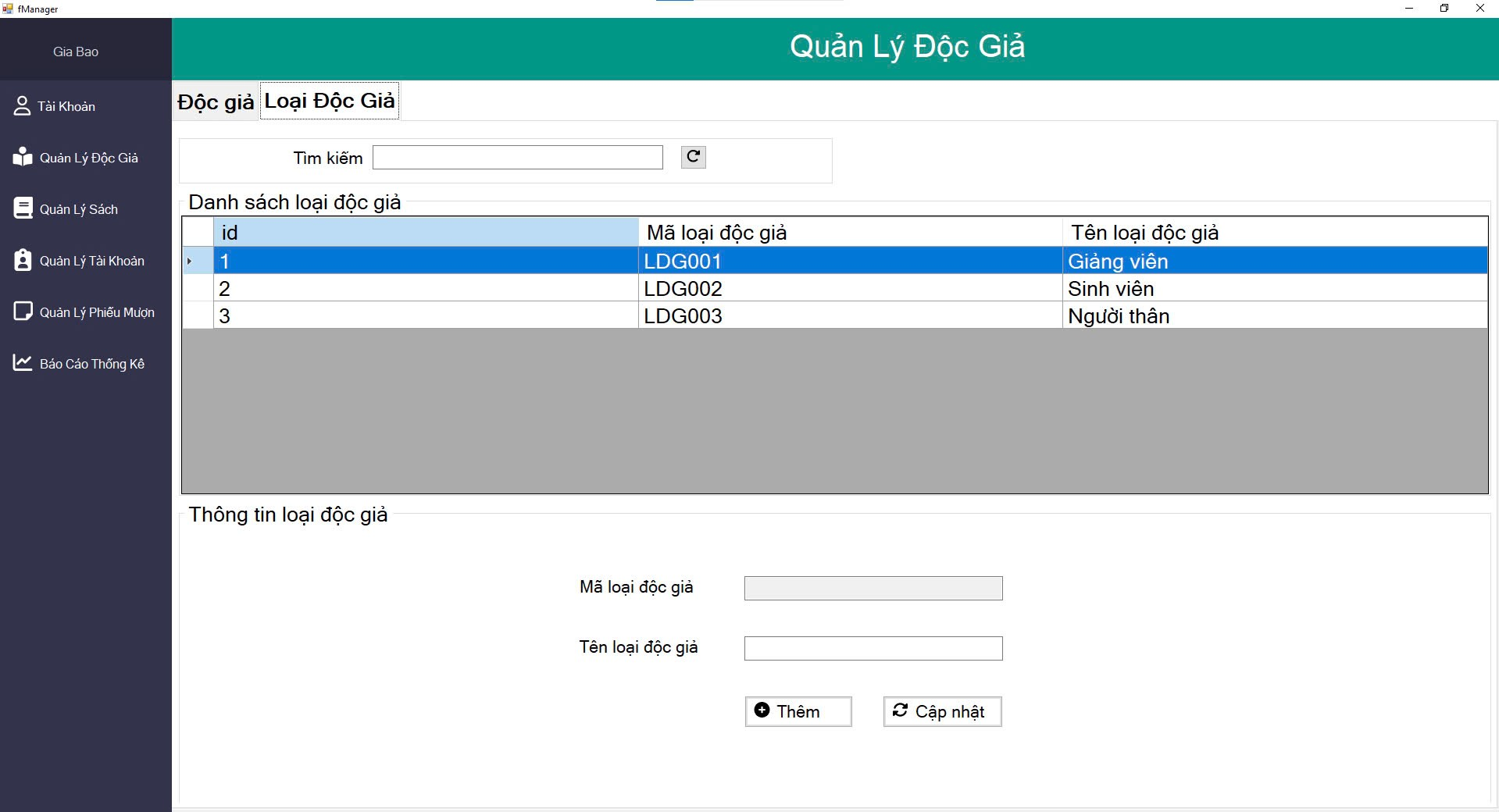
* Các thông tin cơ bản của từng loại độc giả được hiển thị trong bảng gồm các cột như: mã loại độc giả, tên loại độc giả.
* Bảng này cho phép người dùng xem và thao tác với các loại độc giả hiện có trong hệ thống, dễ dàng quản lý thông tin.

1. Thông tin chi tiết loại độc giả:

* Khi người dùng chọn một loại độc giả trong danh sách, các thông tin chi tiết của loại độc giả đó sẽ được hiển thị ở phía dưới, gồm các trường như: mã loại độc giả, tên loại độc giả.
* Các ô nhập liệu này sẽ cho phép người dùng thêm mới hoặc cập nhật thông tin của loại độc giả.

1. Nút thao tác:

* Thêm: Cho phép người dùng thêm mới thông tin loại độc giả vào hệ thống.
* Cập nhật: Người dùng có thể sửa đổi các thông tin của loại độc giả đã có trong hệ thống.



*Hình 2.2.2. Màn hình quản lý loại độc giả*

## 2.3. Quản lý sách

### 2.3.1. Quản lý sách

1. Thanh tìm kiếm:

* Người dùng có thể nhập mã sách, tên sách, tác giả, thể loại, hoặc nhà xuất bản để tìm kiếm thông tin sách nhanh chóng.
* Thanh tìm kiếm nằm phía trên, giúp truy cập nhanh danh sách sách.
* Nút làm mới giúp hiển thị lại toàn bộ danh sách tài khoản khi cần.

1. Danh sách sách:

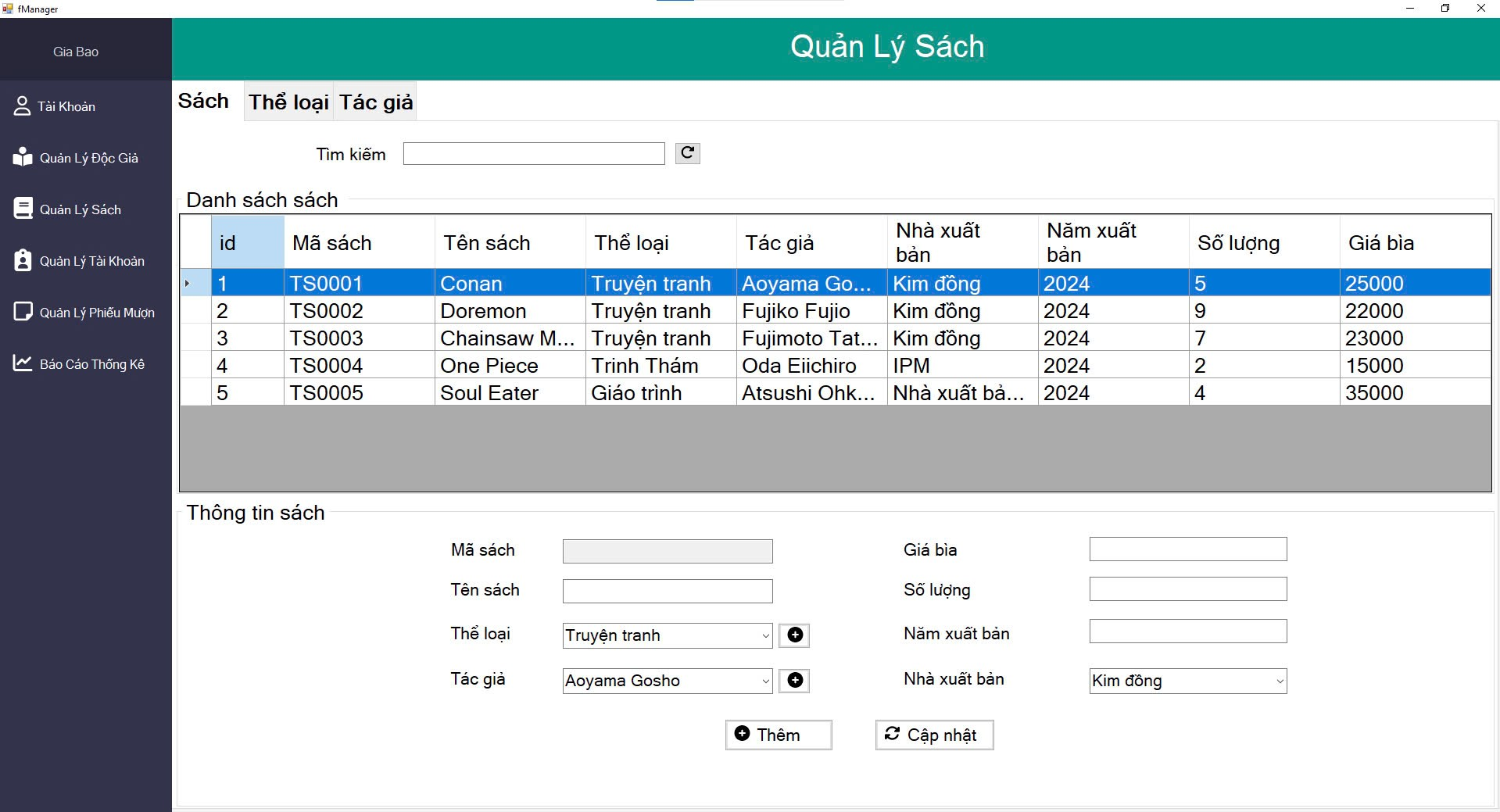
* Hiển thị thông tin các sách hiện có trong thư viện dưới dạng bảng.
* Các cột trong bảng bao gồm:
  + ID: Mã định danh tự động tăng.
  + Mã sách: Mã duy nhất của sách (ví dụ: TS0001), được sinh tự động.
  + Tên sách: Tên sách.
  + Thể loại: Loại sách (ví dụ: Truyện tranh, Giáo trình...).
  + Tác giả: Tên tác giả.
  + Nhà xuất bản: Nhà xuất bản của sách.
  + Năm xuất bản: Năm phát hành sách.
  + Số lượng: Số lượng sách có sẵn trong thư viện.
  + Giá bìa: Giá niêm yết của sách.

1. Thông tin chi tiết sách:

* Khi chọn một hàng trong danh sách sách, các thông tin chi tiết sẽ được hiển thị ở bảng nhập liệu bên dưới.
* Các trường thông tin bao gồm: mã sách, tên sách, thể loại, tác giả, nhà xuất bản, năm xuất bản, số lượng, giá bìa
* Các trường nhập liệu này cho phép người dùng thêm mới hoặc cập nhật thông tin sách.

1. Nút chức năng:

* Thêm: Người dùng nhập các thông tin chi tiết của một quyển sách mới và nhấn nút để thêm vào cơ sở dữ liệu.
* Cập nhật: Khi muốn chỉnh sửa thông tin một sách hiện có, người dùng có thể chỉnh sửa trong ô nhập liệu và nhấn nút này để lưu thay đổi.



*Hình 2.3.1. Màn hình quản lý sách*

### 2.3.2. Quản lý thể loại

1. Thanh tìm kiếm:

* Người dùng có thể nhập tên thể loại để tìm kiếm thông tin thể loại nhanh chóng.
* Thanh tìm kiếm nằm phía trên, giúp truy cập nhanh danh sách thể loại.
* Nút làm mới giúp hiển thị lại toàn bộ danh sách tài khoản khi cần.

1. Danh sách thể loại:

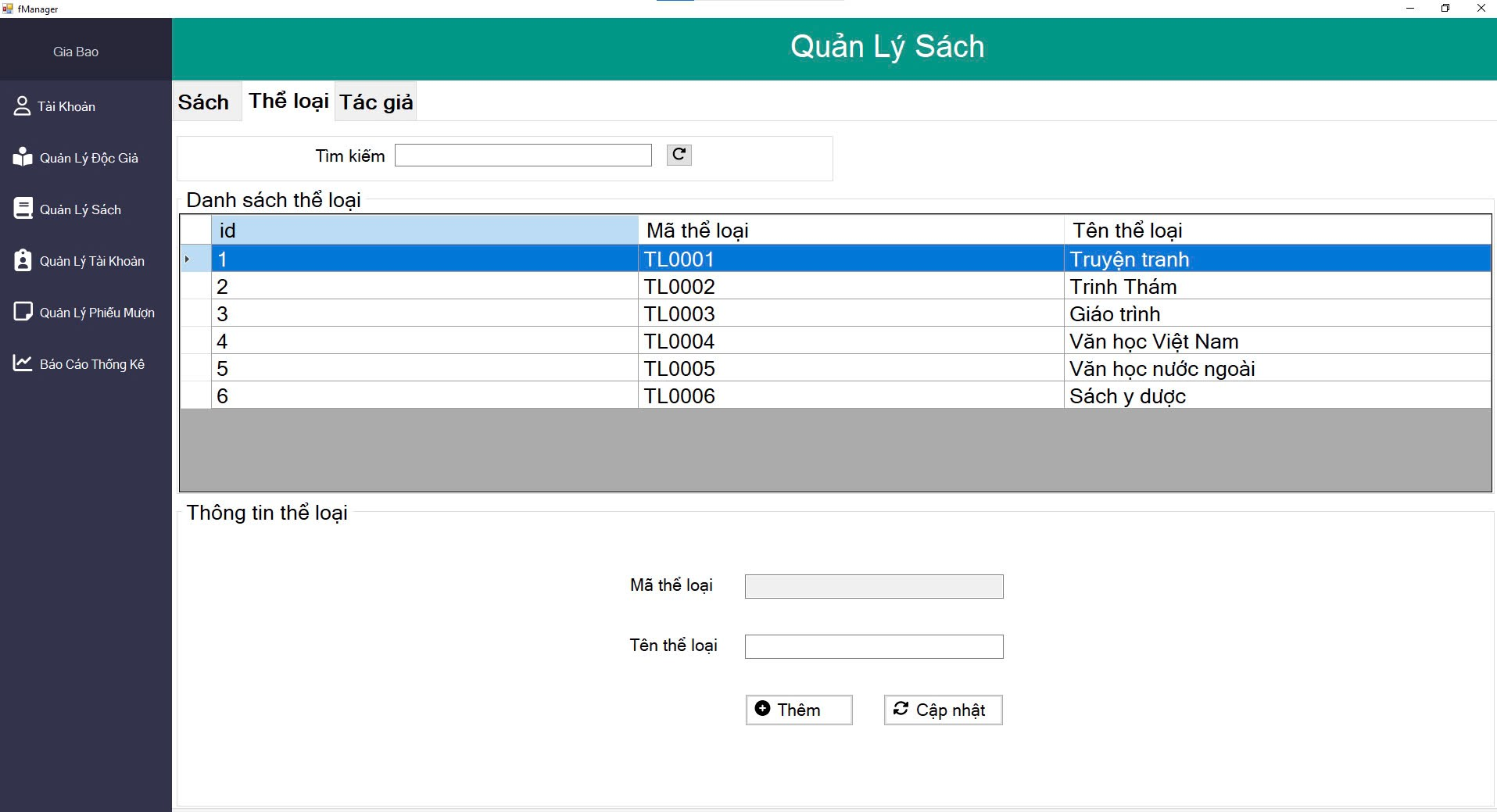
* Hiển thị thông tin các thể loại hiện có trong thư viện dưới dạng bảng.
* Các cột trong bảng bao gồm:
  + ID: Mã định danh tự động tăng.
  + Mã thể loại: Mã duy nhất của thể loại (ví dụ: TL0001), được sinh tự động.
  + Tên thể loại: Tên thể loại.

1. Thông tin chi tiết thể loại:

* Khi chọn một hàng trong danh sách thể loại, các thông tin chi tiết sẽ được hiển thị ở bảng nhập liệu bên dưới.
* Các trường thông tin bao gồm: mã thể loại, tên thể loại.
* Các trường nhập liệu này cho phép người dùng thêm mới hoặc cập nhật thông tin thể loại.

1. Nút chức năng:

* Thêm: Người dùng nhập các thông tin chi tiết của một thể loại mới và nhấn nút để thêm vào cơ sở dữ liệu.
* Cập nhật: Khi muốn chỉnh sửa thông tin một thể loại hiện có, người dùng có thể chỉnh sửa trong ô nhập liệu và nhấn nút này để lưu thay đổi.



*Hình 2.3.2. Màn hình quản lý thể loại*

### 2.3.3. Quản lý tác giả

1. Thanh tìm kiếm:

* Người dùng có thể nhập tên tác giả để tìm kiếm thông tin thể loại nhanh chóng.
* Thanh tìm kiếm nằm phía trên, giúp truy cập nhanh danh sách tác giả.
* Nút làm mới giúp hiển thị lại toàn bộ danh sách tài khoản khi cần.

1. Danh sách tác giả:

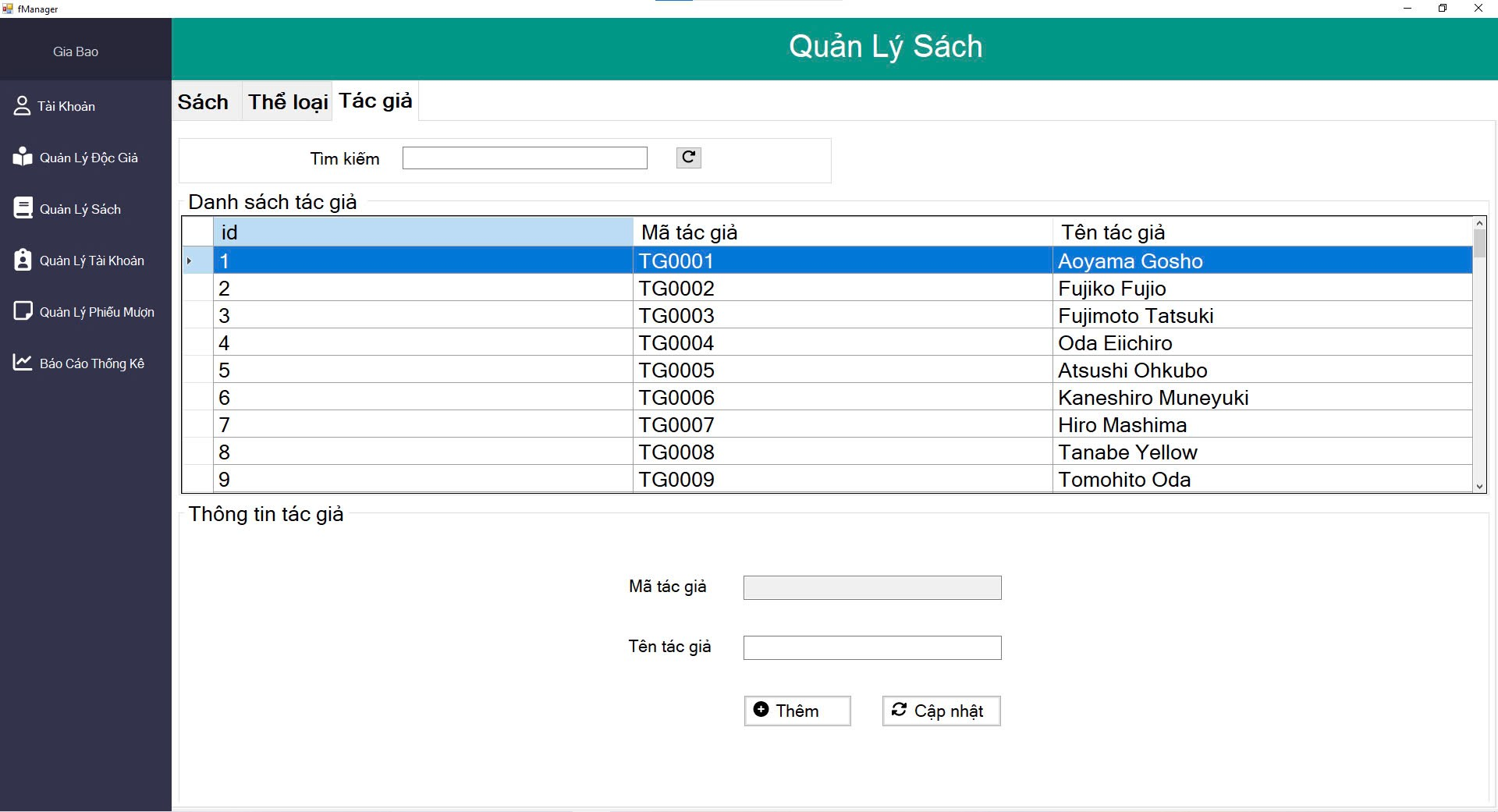
* Hiển thị thông tin các tác giả hiện có trong thư viện dưới dạng bảng.
* Các cột trong bảng bao gồm:
  + ID: Mã định danh tự động tăng.
  + Mã tác giả: Mã duy nhất của tác giả (ví dụ: TG0001), được sinh tự động.
  + Tên tác giả: Tên tác giả.

1. Thông tin chi tiết tác giả:

* Khi chọn một hàng trong danh sách tác giả, các thông tin chi tiết sẽ được hiển thị ở bảng nhập liệu bên dưới.
* Các trường thông tin bao gồm: mã tác giả, tên tác giả.
* Các trường nhập liệu này cho phép người dùng thêm mới hoặc cập nhật thông tin tác giả.

1. Nút chức năng:

* Thêm: Người dùng nhập các thông tin chi tiết của một tác giả mới và nhấn nút để thêm vào cơ sở dữ liệu.
* Cập nhật: Khi muốn chỉnh sửa thông tin một tác giả hiện có, người dùng có thể chnh sửa trong ô nhập liệu và nhấn nút này để lưu thay đổi.



*Hình 2.3.3. Màn hình quản lý tác giả*

## 2.4. Quản lý tài khoản

1. Thanh tìm kiếm:

* Người dùng có thể nhập thông tin như mã tài khoản, tên tài khoản, hoặc tên đăng nhập để nhanh chóng tìm kiếm tài khoản trong hệ thống.
* Nút làm mới giúp hiển thị lại toàn bộ danh sách tài khoản khi cần.

1. Danh sách tài khoản:

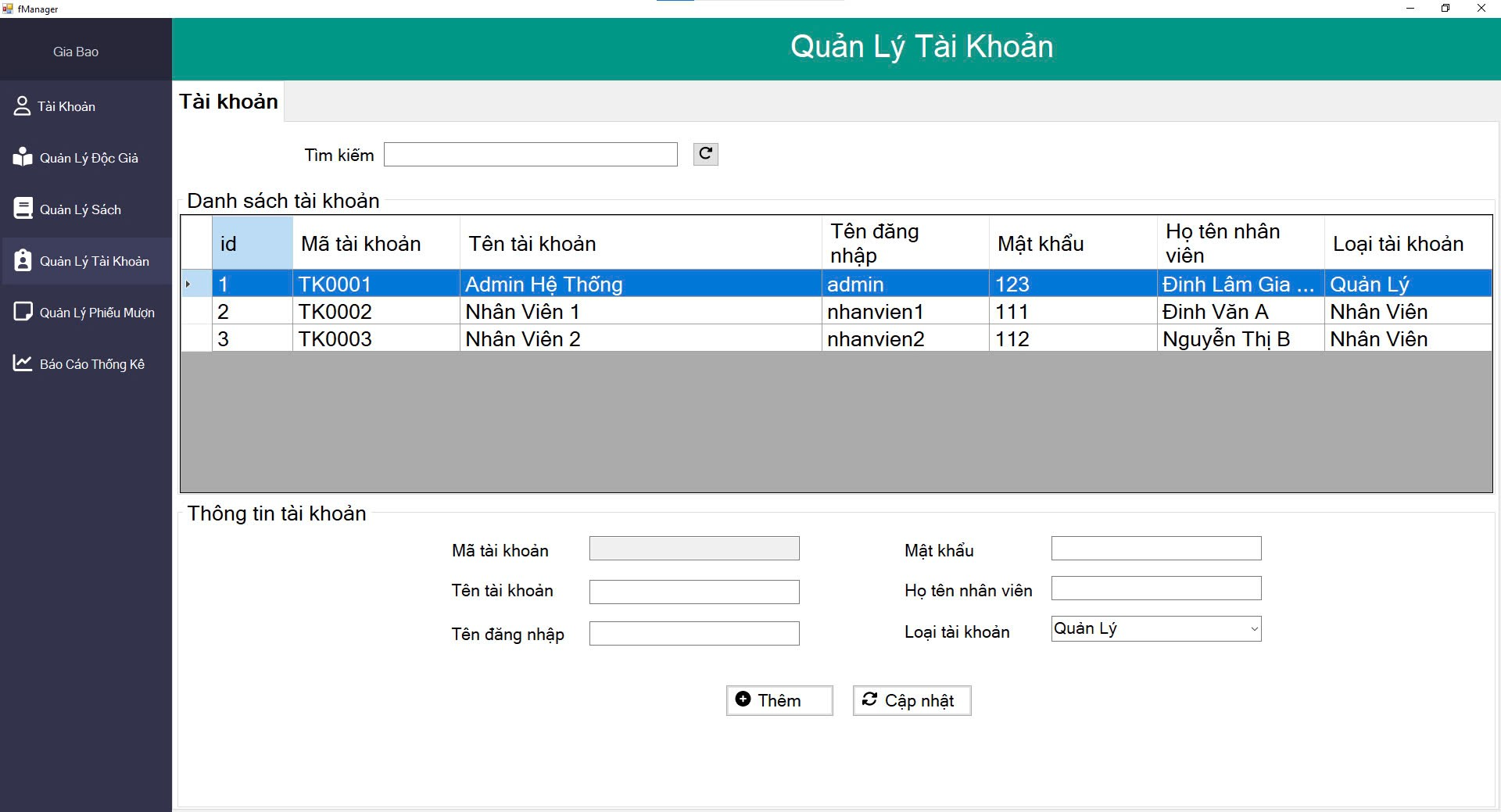
* Hiển thị thông tin các tài khoản đang có trong hệ thống dưới dạng bảng.
* Các cột trong bảng bao gồm:
  + ID: Mã định danh tự động tăng.
  + Mã tài khoản: Mã duy nhất của tài khoản (ví dụ: TK0001).
  + Tên tài khoản: Tên gọi của tài khoản (ví dụ: Admin Hệ Thống, Nhân Viên...).
  + Tên đăng nhập: Tên người dùng sử dụng để đăng nhập hệ thống.
  + Mật khẩu: Mật khẩu của tài khoản.
  + Họ tên nhân viên: Họ và tên đầy đủ của nhân viên sở hữu tài khoản.
  + Loại tài khoản: Phân loại tài khoản theo vai trò (ví dụ: Quản lý, Nhân viên...).

1. Thông tin chi tiết tài khoản:

* Khi chọn một hàng trong danh sách, thông tin chi tiết của tài khoản sẽ hiển thị tại bảng nhập liệu bên dưới.
* Các trường nhập liệu bao gồm:
  + Mã tài khoản
  + Tên tài khoản
  + Tên đăng nhập
  + Mật khẩu
  + Họ tên nhân viên
  + Loại tài khoản (có danh sách thả xuống để chọn).

1. Nút chức năng:

* Thêm: Cho phép nhập thông tin chi tiết của một tài khoản mới và thêm vào hệ thống.
* Cập nhật: Khi muốn chỉnh sửa thông tin của tài khoản hiện có, người dùng có thể thay đổi các trường nhập liệu và nhấn nút này để lưu lại.

**

*Hình 2.4. Màn hình quản lý tài khoản*

## 2.5. Quản lý mượn, trả sách

### 2.5.1. Quản lý mượn sách

1. Tìm kiếm sách:

* Cho phép người dùng nhập từ khóa để tìm kiếm sách trong hệ thống dựa trên mã sách hoặc tên sách.
* Giúp rút ngắn thời gian tìm kiếm thông tin sách cần mượn.

1. Danh sách sách:

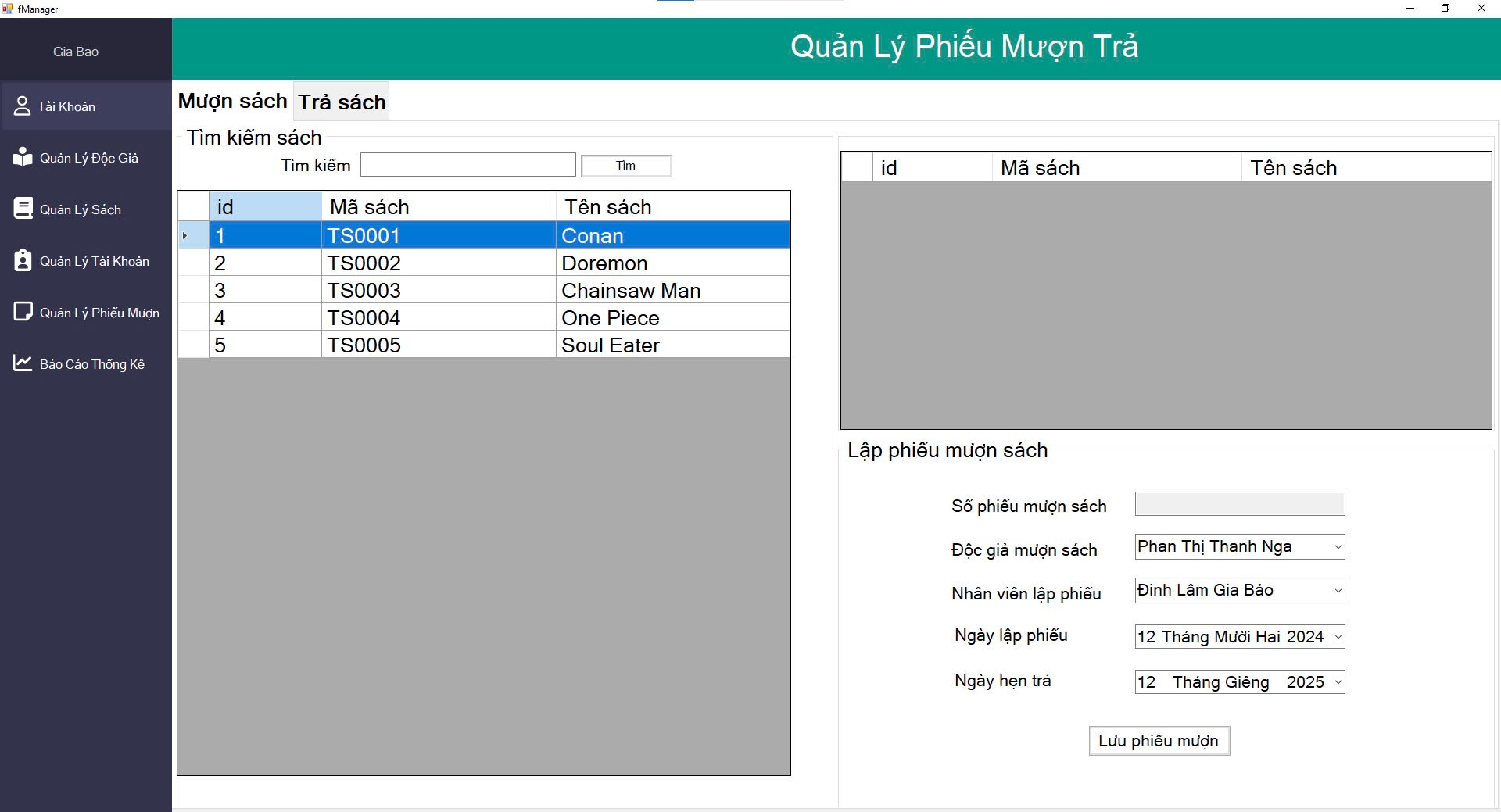
* Hiển thị danh sách sách có sẵn trong thư viện (hoặc những sách còn trong kho mượn).
* Bao gồm các cột thông tin:
  + ID
  + Mã sách
  + Tên sách

1. Lập phiếu mượn sách:

* Cho phép tạo mới một phiếu mượn sách với các thông tin:
* Số phiếu mượn sách: Là mã định danh duy nhất cho mỗi giao dịch.
* Độc giả mượn sách: Danh sách thả xuống để chọn tên độc giả.
* Nhân viên lập phiếu: Danh sách thả xuống để chỉ định nhân viên thực hiện thao tác mượn sách.
* Ngày lập phiếu: Tự động lấy ngày hiện tại hoặc cho phép chỉnh sửa.
* Ngày hẹn trả: Cho phép người dùng nhập ngày dự kiến độc giả trả sách.

1. Lưu phiếu mượn:

* Khi nhấn nút "Lưu phiếu mượn", dữ liệu sẽ được lưu vào hệ thống, bao gồm toàn bộ thông tin về phiếu mượn.



*Hình 2.5.1 Màn hình quản lý mượn sách*

### 2.5.2. Quản lý trả sách

1. Tìm kiếm thông tin trả sách:

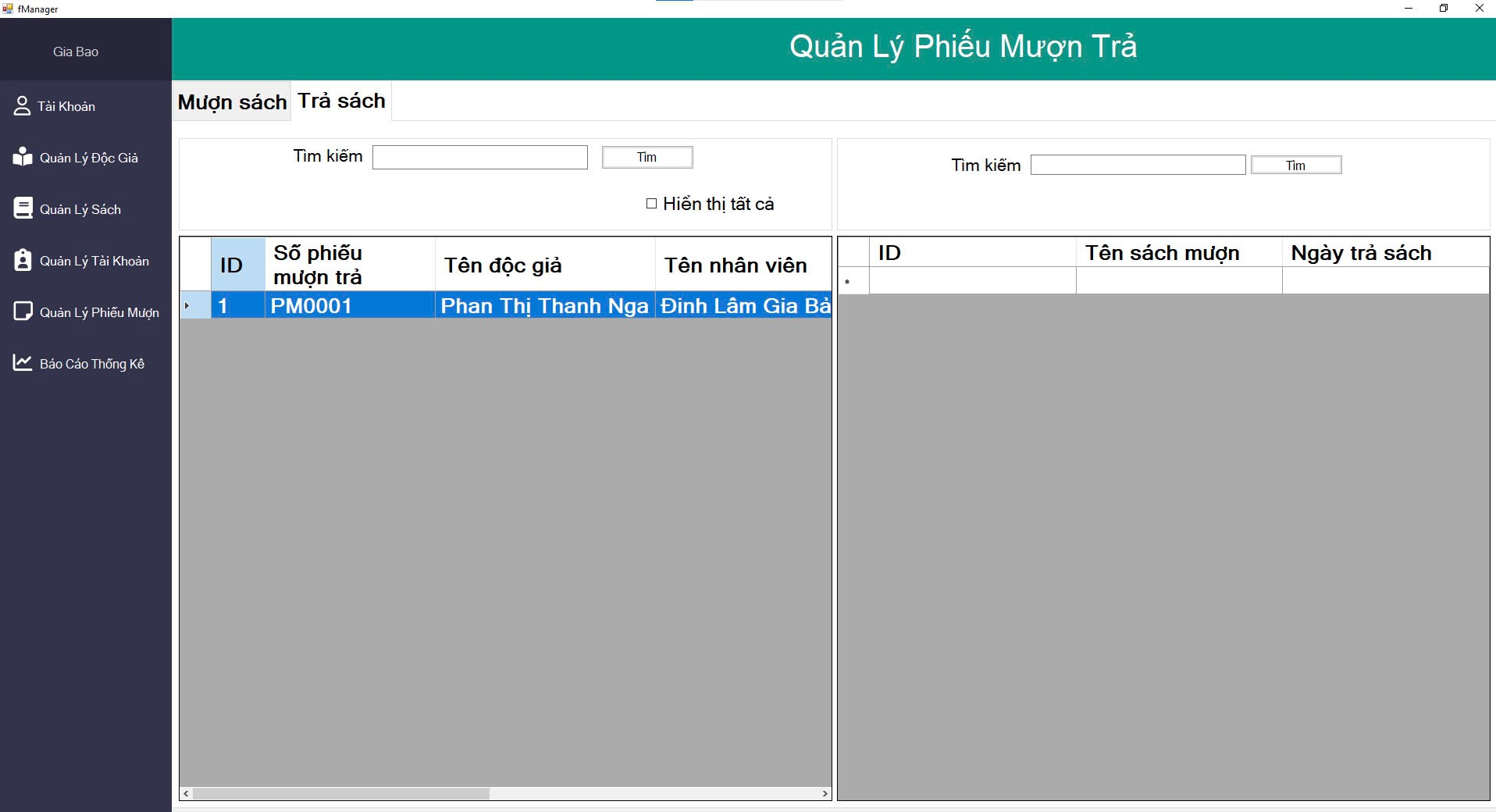
* Ô tìm kiếm cho phép nhập từ khóa (theo số phiếu, tên độc giả, hoặc tên sách).
* Nút "Tìm" thực hiện lọc danh sách phiếu trả dựa trên từ khóa.

1. Bảng danh sách phiếu mượn trả:

* Hiển thị thông tin cơ bản:
  + ID: Số thứ tự của phiếu mượn trả.
  + Số phiếu mượn trả: Mã định danh duy nhất của phiếu.
  + Tên độc giả: Độc giả đã mượn sách.
  + Tên nhân viên: Nhân viên chịu trách nhiệm lập phiếu trả.
  + Ngày mượn: Hiển thị ngày lập phiếu mượn sách.
  + Hạn trả: Hiển thị hạn cuối cùng mà độc giả phải trả sách,
  + Đã trả sách: false nếu độc giả có 1 hoặc nhiều cuốn sách chưa trả, true nếu độc giả đã trả hết toàn bộ sách.
  + Tiền phạt: phạt tiền khi trả sách quá hạn trả
* Bảng này cho phép người dùng trả một lần hết toàn bộ sách, cột dữ liệu Đã trả sách chuyển từ false thành true

1. Thông tin chi tiết sách mượn:

* Khi chọn một phiếu mượn ở bảng bên trái, bảng bên phải sẽ hiển thị danh sách sách tương ứng với phiếu mượn đó.
* Các thông tin hiển thị bao gồm:
  + ID: Mã định danh sách mượn trong phiếu.
  + Tên sách mượn: Danh sách các tựa sách được mượn.
  + Ngày trả sách: Ngày độc giả thực hiện trả sách.

****

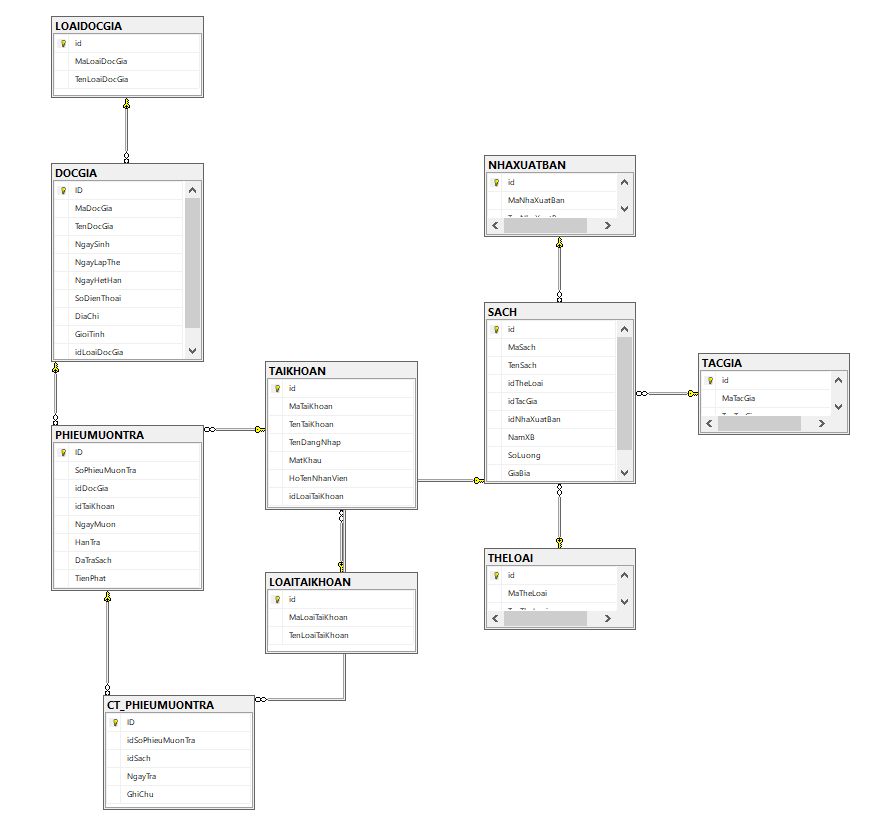
*Hình 2.5.2. Màn hình quản lý trả sách*

# III. QUÁ TRÌNH THỰC HIỆN VÀ MÃ NGUỒN

## 3.1. Xây dựng dữ liệu

Cơ sở dữ liệu gồm các bảng:

* Bảng LOAIDOCGIA: Quản lý thông tin loại độc giả.
* Bảng DOCGIA: Quản lý thông tin độc giả.
* Bảng PHIEUMUONTRA: Quản lý thông tin phiếu mượn và trả sách.
* Bảng CT\_PHIEUMUONTRA: Quản lý chi tiết phiếu mượn sách.
* Bảng SACH: Quản lý thông tin sách.
* Bảng NHAXUATBAN: Quản lý thông tin nhà xuất bản.
* Bảng TACGIA: Quản lý thông tin tác giả.
* Bảng THELOAI: Quản lý thông tin thể loại sách.
* Bảng TAIKHOAN: Quản lý thông tin tài khoản người dùng.
* Bảng LOAITAIKHOAN: Quản lý loại tài khoản.



*Hình 3.1. Lược đồ minh họa dữ liệu*

## 3.2. Cài đặt và xây dựng giao diện bằng Winform C#

### 3.2.1. Cài đặt Winform

* Sử dụng C# và Windows Forms để xây dựng giao diện ứng dụng quản lý thư viện sách.
* Cài đặt Visual Studio làm môi trường phát triển tích hợp (IDE) để thiết kế giao diện và viết mã nguồn.
* Kết nối SQL Server để quản lý dữ liệu của hệ thống.

### 3.2.2. Xây dựng giao diện mã nguồn

* Thiết kế các giao diện chính bao gồm:
  + Giao diện quản lý độc giả.
  + Giao diện quản lý sách.
  + Giao diện quản lý phiếu mượn/trả sách.
  + Giao diện quản lý tài khoản người dùng.
* Bố cục giao diện: Sử dụng các điều khiển như Button, DataGridView, TextBox, ComboBox để thiết lập các chức năng quản lý như Thêm, Xóa, Sửa, Tìm kiếm.
* Phân tách giao diện: Các màn hình được thiết kế riêng và hiển thị khi nhấn các nút chức năng.

## 3.3. Xây dựng xử lý sự kiện tiêu biểu

### 3.3.1. Hiển thị các màn hình quản lý khi nhấn nút

### 3.3.2. Kết nối database và sự kiện thêm, xóa, sửa, mặc định, tìm kiếm

### 3.3.3. Chức năng mượn trả sách

# 

# IV. KẾT LUẬN

## 4.1. Kết quả đạt được

Trong quá trình phát triển ứng dụng quản lý thư viện sách sử dụng ngôn ngữ C# và kết nối cơ sở dữ liệu, dự án đã đạt được một số kết quả quan trọng như sau:

1. Giao diện người dùng thân thiện và trực quan:

* Giao diện được thiết kế đơn giản, dễ sử dụng và đáp ứng tốt nhu cầu quản lý thư viện. Người dùng có thể dễ dàng thêm, cập nhật, xóa và tìm kiếm thông tin về độc giả, sách, phiếu mượn và tài khoản thông qua các bảng biểu và ô nhập liệu.

1. Quản lý thông tin sách và độc giả hiệu quả:

* Ứng dụng cho phép quản lý đầy đủ thông tin về độc giả như mã độc giả, họ tên, ngày sinh, số điện thoại, địa chỉ, loại độc giả và thời hạn sử dụng thẻ.
* Quản lý thông tin sách bao gồm mã sách, tên sách, tác giả, thể loại, năm xuất bản, nhà xuất bản, số lượng và giá bìa.
* Các chức năng như thêm mới, cập nhật, xóa, hiển thị và tìm kiếm giúp việc quản lý trở nên nhanh chóng và hiệu quả.

1. Quản lý phiếu mượn trả linh hoạt:

* Ứng dụng hỗ trợ quản lý việc mượn và trả sách, bao gồm theo dõi thông tin chi tiết về sách mượn, ngày mượn, ngày trả và tình trạng của sách.

1. Quản lý tài khoản người dùng:

* Hệ thống cho phép tạo, chỉnh sửa và xóa tài khoản người dùng. Việc phân quyền (quản trị viên, nhân viên) đảm bảo bảo mật thông tin và phù hợp với từng vai trò sử dụng.

1. Kết nối cơ sở dữ liệu đáng tin cậy:

* Ứng dụng kết nối với cơ sở dữ liệu SQL Server để lưu trữ và truy xuất thông tin một cách nhanh chóng và an toàn. Cơ sở dữ liệu đảm bảo tính toàn vẹn và bảo mật của dữ liệu, giúp duy trì và bảo vệ thông tin trong thư viện.

1. Thử nghiệm và kiểm tra chất lượng:

* Ứng dụng đã được thử nghiệm và kiểm tra đầy đủ để đảm bảo hoạt động ổn định, hiệu quả và không phát sinh lỗi trong quá trình sử dụng. Việc xác minh các tính năng như thêm, xóa, sửa, tìm kiếm và báo cáo giúp đảm bảo đáp ứng đúng yêu cầu đặt ra.

## 4.2. Hạn chế

Mặc dù đã đạt được nhiều kết quả tích cực, ứng dụng quản lý thư viện sách sử dụng WinForm C# cũng còn tồn tại một số hạn chế như sau:

1. Giao diện hạn chế:

* WinForm tuy dễ sử dụng và phổ biến nhưng còn hạn chế trong việc thiết kế các giao diện hiện đại và phức tạp.
* Ví dụ: Chưa có các tiện ích thân thiện như datetime picker nâng cao để chọn ngày mượn và trả sách hoặc các thiết kế bảng biểu sinh động.

1. Khả năng mở rộng:

* Ứng dụng hiện tại chỉ tập trung vào quản lý sách và độc giả, phiếu mượn trả. Để hoàn thiện hơn, cần mở rộng thêm các tính năng như:
  + Xuất hóa đơn mượn/trả sách trên cửa sổ riêng.
  + Quản lý các dịch vụ bổ sung như gia hạn sách, thông báo trễ hạn qua email.
  + Tăng cường mối liên hệ giữa các chức năng quản lý độc giả và quản lý sách để đảm bảo thông tin luôn đồng bộ và liên kết chặt chẽ.

1. Hiệu suất:

* Khi số lượng độc giả và sách trong hệ thống tăng lớn, ứng dụng có thể gặp vấn đề về hiệu suất và thời gian phản hồi chậm do chưa được tối ưu hóa.

1. Hạn chế về tích hợp:

* Ứng dụng hiện tại chưa hỗ trợ tích hợp với các dịch vụ hoặc công nghệ ngoài như:
  + Hệ thống thanh toán trực tuyến cho các khoản phí trễ hạn.
  + Gửi thông báo qua email/SMS để nhắc nhở trả sách.
  + Kết nối với các cơ sở dữ liệu ngoài hoặc hệ thống thư viện khác.

1. Hạn chế về đa nhiệm:

* WinForm vẫn còn hạn chế khi xử lý các tác vụ đa nhiệm. Ứng dụng có thể gặp khó khăn khi thực hiện nhiều sự kiện đồng thời, ví dụ như xử lý dữ liệu trong nền trong khi người dùng thao tác trên giao diện.

## . Hướng phát triển

1. Tích hợp chức năng gửi thông báo tự động

* Tích hợp tính năng gửi email hoặc tin nhắn SMS tự động để thông báo cho độc giả về:
  + Ngày đến hạn trả sách.
  + Sách mượn quá hạn và tiền phạt (nếu có).
  + Xác nhận phiếu mượn/trả khi độc giả hoàn thành giao dịch

1. **Xuất báo cáo và thống kê dữ liệu**

* **Phát triển chức năng xuất báo cáo về:**
  + **Thống kê số lượng sách còn trong kho và sách đã cho mượn.**
  + **Thống kê danh sách độc giả mượn sách nhiều nhất.**
  + **Báo cáo tình trạng sách quá hạn và tiền phạt tương ứng.**
* **Hỗ trợ xuất báo cáo ra các định dạng file như PDF, Excel, hoặc in trực tiếp.**

1. **Cải tiến giao diện người dùng**

* **Tối ưu hóa và làm mới giao diện bằng cách sử dụng các thư viện UI hiện đại như Bunifu, MetroFramework để giao diện trực quan, đẹp mắt và thân thiện hơn.**
* **Xây dựng giao diện responsive để phù hợp với nhiều kích thước màn hình khác nhau.**

1. **Phát triển hệ thống quản lý thư viện sách nâng cao**

* **Bổ sung chức năng quản lý tình trạng sách, bao gồm:**
  + **Đánh dấu sách hư hỏng, mất mát hoặc cần bổ sung.**
  + **Cập nhật số lượng sách trong kho khi có sự thay đổi.**
* **Xây dựng tính năng nhập và quản lý sách mới từ file Excel để tiết kiệm thời gian thao tác.**

1. **Tối ưu hóa hiệu suất hệ thống**

* **Tối ưu hóa truy vấn SQL để đảm bảo hệ thống hoạt động mượt mà và nhanh chóng khi dữ liệu lớn.**
* **Cải thiện khả năng xử lý đa nhiệm, cho phép người dùng thao tác nhiều chức năng cùng lúc.**

# V. TÀI LIỆU THAM KHẢO

## 6.1. Sách

* Sách hướng dẫn thực hành phát triển ứng dụng desktop

## 6.2. Trang web

* Giới thiệu mô hình 3 lớp của Winform C#:

<https://viblo.asia/p/gioi-thieu-mo-hinh-3-lop-trong-c-gDVK2Q9w5Lj>

* Tạo ứng dụng Window Forms bằng C#:

[https://learn.microsoft.com/en-us/visualstudio/ide/create-csharp-winform-visual- studio?view=vs-2022](https://learn.microsoft.com/en-us/visualstudio/ide/create-csharp-winform-visual-%20%20studio?view=vs-2022)

* A Step by Step Guide to In-Memory Caching in ASP.NET Core:

<https://www.ezzylearning.net/tutorial/a-step-by-step-guide-to-in-memory-caching-in-asp-net-core#google_vignette>

# VI. LỜI CẢM ƠN

Trong quá trình hoàn thành dự án này, em xin bày tỏ lòng biết ơn sâu sắc đến cô Phan Thị Thanh Nga – giáo viên hướng dẫn của em trong môn lập trình C# WinForm. Cô đã tận tình hướng dẫn, cung cấp kiến thức và hỗ trợ em xuyên suốt quá trình thực hiện dự án. Những góp ý, nhận xét và khuyến nghị của cô đã đóng vai trò quan trọng trong việc định hướng và nâng cao chất lượng của sản phẩm này.

Em cũng xin gửi lời cảm ơn chân thành đến các bạn sinh viên trong lớp đã nhiệt tình hỗ trợ và đóng góp ý kiến quý báu. Những cuộc thảo luận, phản hồi và sự chia sẻ kiến thức từ các bạn đã giúp em hoàn thiện hơn dự án của mình.

Cuối cùng, em xin chân thành cảm ơn tất cả những ai đã trực tiếp hoặc gián tiếp đóng góp, hỗ trợ và tạo điều kiện thuận lợi để em có thể hoàn thành dự án này. Sự giúp đỡ và động viên của mọi người chính là động lực để em cố gắng và hoàn thành tốt công việc được giao.

Trân trọng,

Đinh Lâm Gia Bảo