**MÔ TẢ HỆ THỐNG QUẢN LÝ THƯ VIỆN (SỬ DỤNG JAVA + MYSQL + JAVAFX)**

1. Mục tiêu hệ thống

Xây dựng ứng dụng desktop hỗ trợ quản lý thư viện, bao gồm quản lý sách, người mượn, quá trình mượn/trả sách, và thống kê số liệu sử dụng. Hệ thống có giao diện thân thiện với người dùng, xử lý dữ liệu hiệu quả và lưu trữ an toàn trong MySQL.

1. Công nghệ sử dụng

|  |  |
| --- | --- |
| Thành phần | Công nghệ |
| Ngôn ngữ lập trình | Java |
| Giao diện | JavaFX (có thể dùng FXML + SceneBuilder) |
| Cơ sở dữ liệu | MySQL |
| Kiến trúc | MVC (Model – View – Controller) |
| Control JavaFX | TableView, TextField, Button, ComboBox, DatePicker |
| Layout JavaFX | VBox, HBox, BorderPane, GridPane |

1. Chức năng hệ thống
2. Quản lý sách

* Thêm sách mới: tên sách, tác giả, tình trạng mượn.
* Sửa thông tin sách.
* Xóa sách khỏi hệ thống.
* Hiển thị danh sách sách dưới dạng bảng.
* Tìm kiếm sách theo tên hoặc tác giả.

1. Quản lý người mượn

* Thêm người mượn mới: tên, email, thông tin liên lạc.
* Sửa thông tin người mượn.
* Xoá người mượn.
* Tìm kiếm người mượn theo tên hoặc email.
* Hiển thị danh sách người mượn.

1. Quản lý Mượn/Trả sách

* Chọn người mượn và chọn sách để thực hiện mượn sách.
* Hiển thị danh sách sách đang được mượn (chưa trả).
* Cập nhật trạng thái khi người dùng trả sách.
* Tự động cập nhật trạng thái sách (còn/đã mượn).
* Không cho mượn nếu sách đã được mượn.

1. Thống kê

* Thống kê tổng số sách, số sách hiện đang được mượn.
* Thống kê số người mượn đang hoạt động.
* Top 5 người mượn nhiều sách nhất.
* Thống kê theo thời gian: số lượt mượn theo tháng/năm.
* Giao diện biểu đồ đơn giản (dùng PieChart, BarChart trong JavaFX).

1. Lưu trữ dữ liệu

Toàn bộ dữ liệu được lưu trong MySQL.

Sử dụng JDBC để kết nối và thực hiện thao tác CRUD.

Ràng buộc khoá ngoại giữa bảng books, users, và loans.

1. Điều hướng giao diện

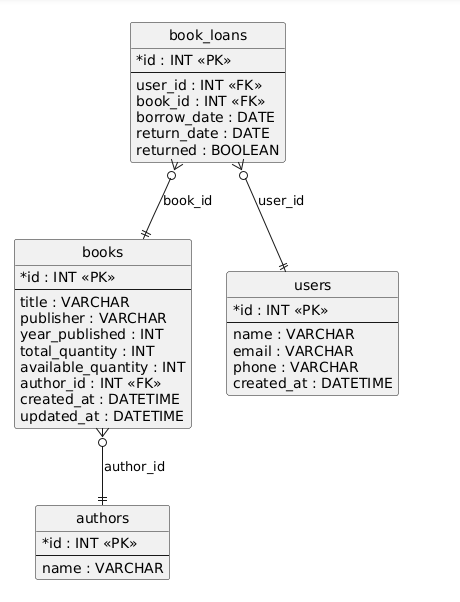
Giao diện sử dụng menu trái hoặc tab điều hướng chính:

* [🏠 Trang chủ]
* [📚 Sách]
* [👤 Người mượn]
* [🔁 Mượn / Trả]
* [📊 Thống kê]

Mỗi chức năng tương ứng với 1 FXML và 1 controller riêng biệt.

Nội dung hiển thị trong StackPane hoặc Center của BorderPane.

1. Thiết kế database



1. Bảng books – Quản lý sách

|  |  |  |
| --- | --- | --- |
| **Tên cột** | **Kiểu dữ liệu** | **Mô tả** |
| id | INT (PK, AUTO\_INCREMENT) | Mã định danh sách |
| title | VARCHAR(255) | Tên sách |
| author\_id | INT (FK → authors.id) | Tác giả |
| publisher | VARCHAR(255) | Nhà xuất bản |
| year\_published | INT | Năm xuất bản |
| total\_quantity | INT | Tổng số lượng bản in |
| available\_quantity | INT | Số bản hiện đang có sẵn để mượn |
| created\_at | DATETIME | Ngày thêm sách |
| updated\_at | DATETIME | Ngày cập nhật gần nhất |

1. Bảng users – Quản lý người mượn

|  |  |  |
| --- | --- | --- |
| **Tên cột** | **Kiểu dữ liệu** | **Mô tả** |
| id | INT (PK) | Mã người dùng |
| name | VARCHAR(255) | Họ và tên |
| email | VARCHAR(255) | Email liên hệ |
| phone | VARCHAR(50) | Số điện thoại |
| created\_at | DATETIME | Ngày đăng ký |

1. Bảng authors – Quản lý tác giả

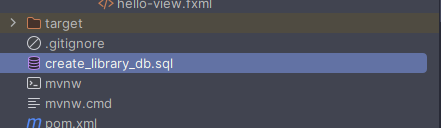
|  |  |  |
| --- | --- | --- |
| **Tên cột** | **Kiểu dữ liệu** | **Mô tả** |
| id | INT (PK) | Mã tác giả |
| name | VARCHAR(255) | Tên tác giả |

1. Bảng book\_loans – Quản lý mượn, trả sách

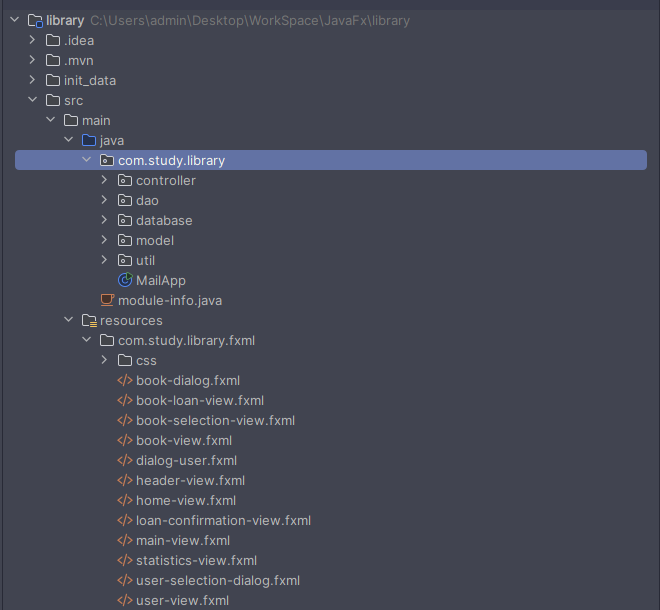
|  |  |  |
| --- | --- | --- |
| **Tên cột** | **Kiểu dữ liệu** | **Mô tả** |
| id | INT (PK) | Mã phiếu mượn |
| user\_id | INT (FK → users.id) | Người mượn |
| book\_id | INT (FK → books.id) | Sách được mượn |
| borrow\_date | DATE | Ngày mượn |
| return\_date | DATE | Ngày trả (nếu có) |
| returned | BOOLEAN | Trạng thái đã trả hay chưa |

Bảng có thể thay đổi để phù hợp vs quy trình, nghiệp vụ

* Script create table lưu trong create\_library\_db.sql



1. Cấu trúc dự án



1. Controller

* Controller trong JavaFX là nơi xử lý logic và sự kiện cho UI.
* Kết nối với file .fxml qua fx:controller và @FXML.
* Là trung tâm xử lý các tương tác người dùng và điều khiển UI theo logic chương trình.

1. Dao

* Kết nối tới database
* Thực hiện các truy vấn SQL (SELECT, INSERT, UPDATE, DELETE)
* Trả kết quả về dưới dạng Model (entity) cho tầng trên sử dụng.

1. Database

* Quản lý thông tin cấu hình database (URL, username, password).
* Tạo và cung cấp đối tượng Connection từ JDBC.

1. Model

* Model là lớp đại diện cho dữ liệu và logic nghiệp vụ của ứng dụng. Nó phản ánh các thực thể trong cơ sở dữ liệu và thường được dùng để trao đổi dữ liệu giữa các tầng (DAO ↔ Controller).

1. Util



Đang lưu Enum

1. Fxml

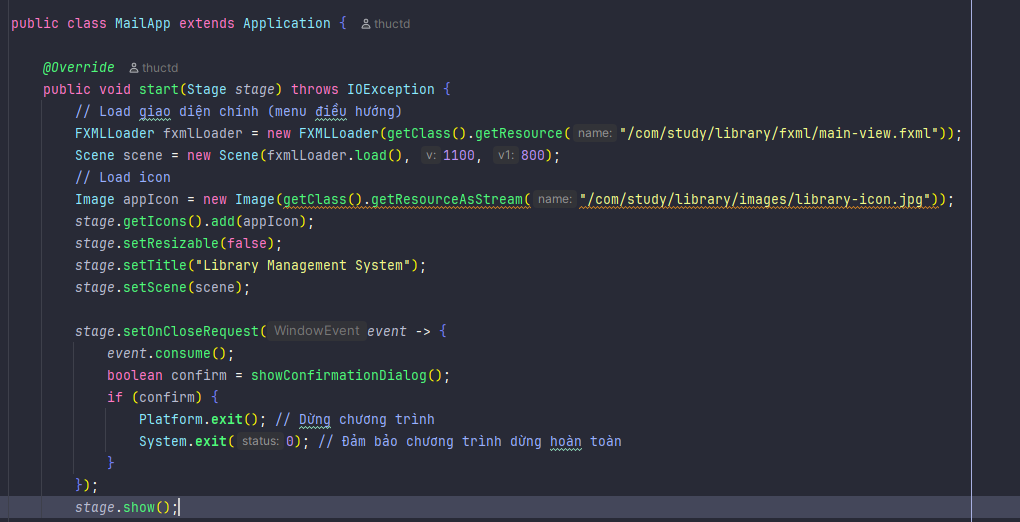
* FXML là XML mô tả giao diện JavaFX.
* Dùng để thiết kế layout một cách rõ ràng, sạch sẽ.
* Gắn với controller thông qua fx:controller và fx:id.

1. Fxml/css

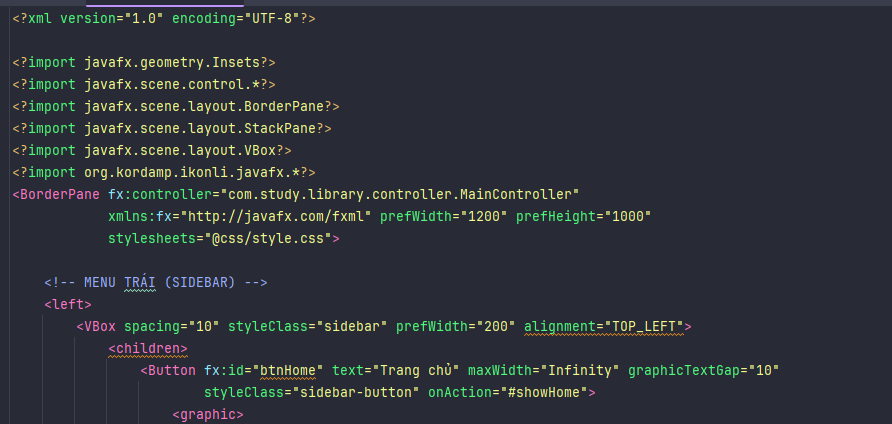
|  |  |
| --- | --- |
| **Thành phần** | **Vai trò** |
| .css file | Định nghĩa style cho giao diện JavaFX |
| styleClass / id | Gắn class hoặc ID để target trong CSS |
| -fx-\* | Prefix các thuộc tính JavaFX CSS |
| Gắn vào FXML hoặc Java | Có thể attach CSS qua XML hoặc code |

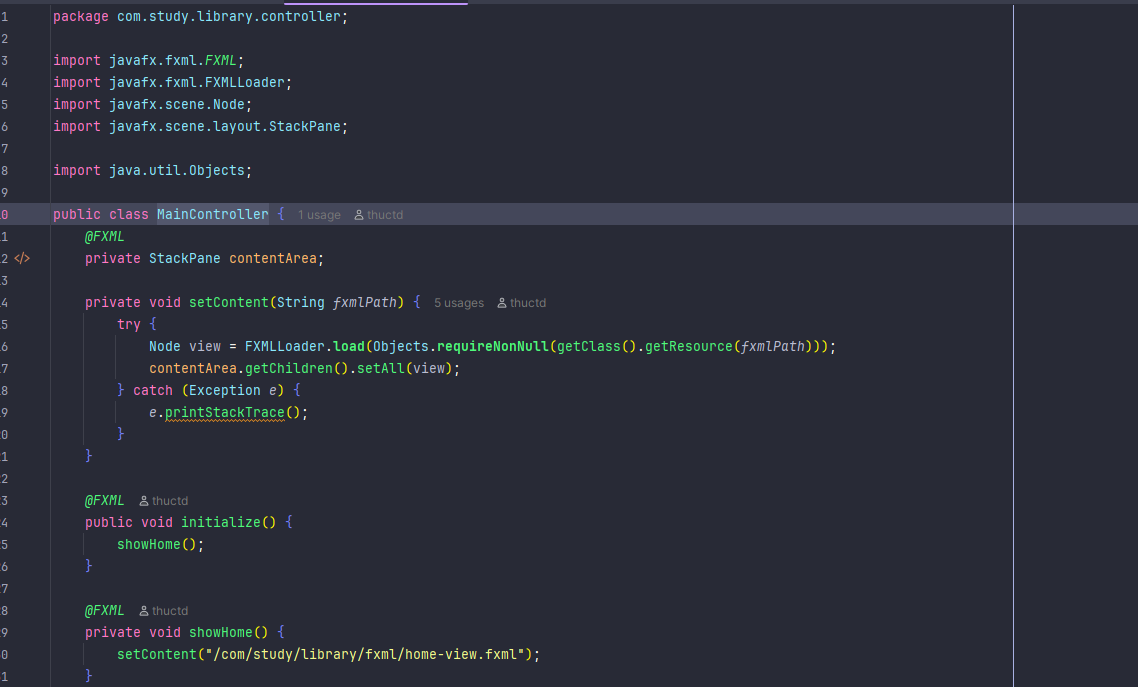
1. Phân tích Luồng

AppMain được chạy

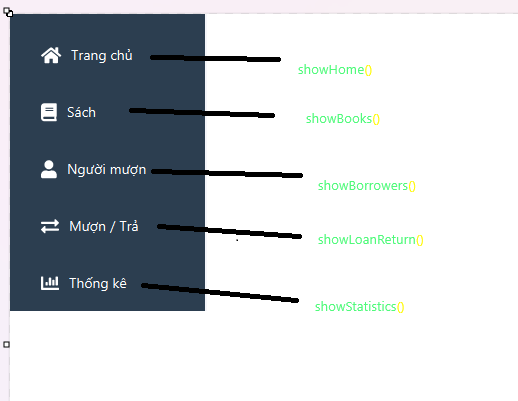


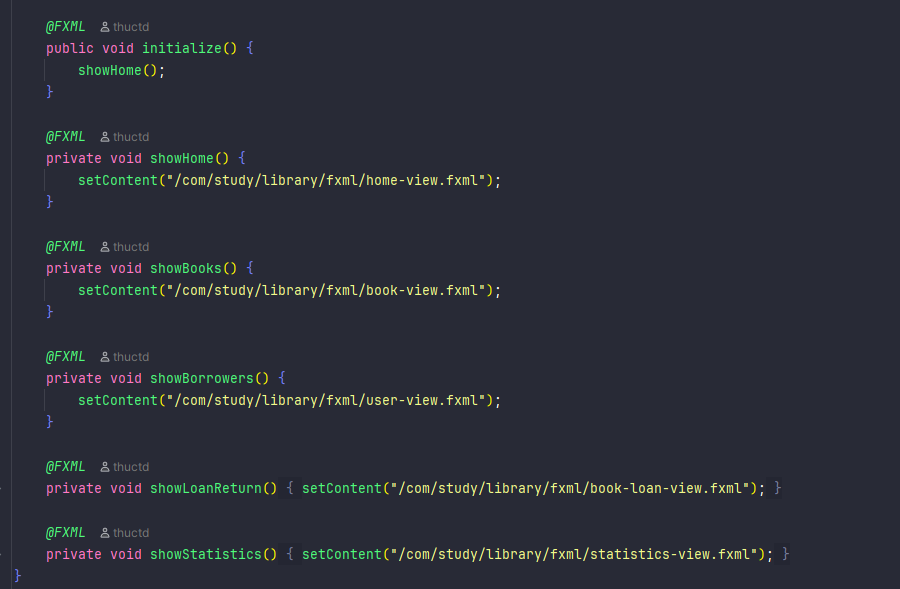
Main-view được load 🡪 controller được khởi tạo



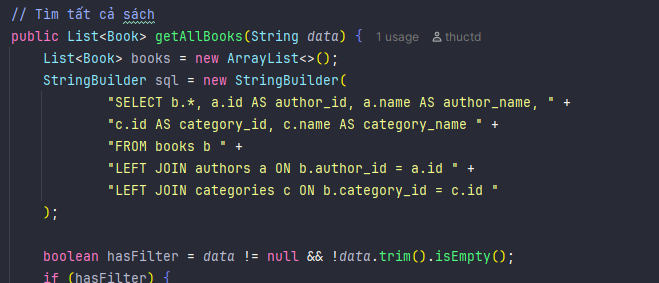


Xử lý tương tác với người dùng trong controller

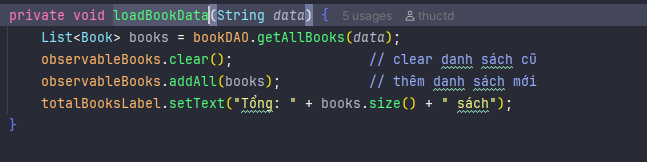




DAO truy vấn dữ liệu



Controller nhận dữ liệu và cập nhật View



CSS được áp dụng cùng lúc vs quy trình trên

[MainApp.java]

↓ load FXML

[FXML file]

↓ gọi controller

[Controller.java]

↓ gọi DAO

[DAO.java]

↓ lấy dữ liệu từ DB

[Model.java]

↑ trả dữ liệu

[Controller.java]

↑ gán vào TableView/ListView...

[FXML View]

→ giao diện cập nhật

Trong lúc đó:

[CSS file] → định dạng UI