|  |
| --- |
| FPT POLYTECHNIC |
| **DỰ ÁN 1** |
| NGÀNH CÔNG NGHỆ THÔNG TIN (ỨNG DỤNG PHẦN MỀM) |

c

|  |
| --- |
| CẦN THƠ 2021 |

|  |
| --- |
| GIẢNG VIÊN HƯỚNG DẪN: ĐINH THÀNH NHÂN |
| SINH VIÊN THỰC HIỆN |
| 1. Nguyễn Chí Hiện (Trưởng nhóm) 2. Trần Thanh Ngoan |

**Lời Nói Đầu**

Trên thế giới cũng như ở Việt Nam, công nghệ thông tin đã trở thành một công nghệ mũi nhọn, nó là ngành khoa học kỹ thuật không thể thiếu trong việc áp dụng vào các hoạt động xã hội như: quản lý hệ thống thư viện sách, kinh tế, thông tin, kinh doanh, và mọi hoạt động trong lĩnh vực nghiên cứu…

Ở nước ta hiện nay, việc áp dụng vi tính hóa trong quản lý tại các cơ quan, xí nghiệp, tổ chức đang rất phổ biến và trở nên cấp thiết, bởi ngành nghề nào cũng đòi hỏi con người phải xử lý khối lượng công việc khổng lồ, và những kiến thức, những suy nghĩ, những đào tạo chuyên sâu. Một vấn đề cấp thiết đặt ra trong quản lý là làm thế nào để chuẩn hóa cách xử lý dữ liệu ở trường học, chính vì thế chúng em chọn đề tài “Phân tích thiết kế hệ thống quản lý thư viện sách”.

Mặc dù đã rất cố gắng để hoàn thành công việc, nhưng do thời gian có hạn và thiếu kinh nghiệm cũng như kỹ năng chưa cao nên việc phân tích và thiết kế còn nhiều thiếu sót, kính mong quý thầy cô và các bạn góp ý, bổ sung để chúng em hoàn thiện cho bài tập tốt hơn nữa. Chúng em xin chân thành cảm ơn!

Cần Thơ, 01/12/2021

# Phân tích

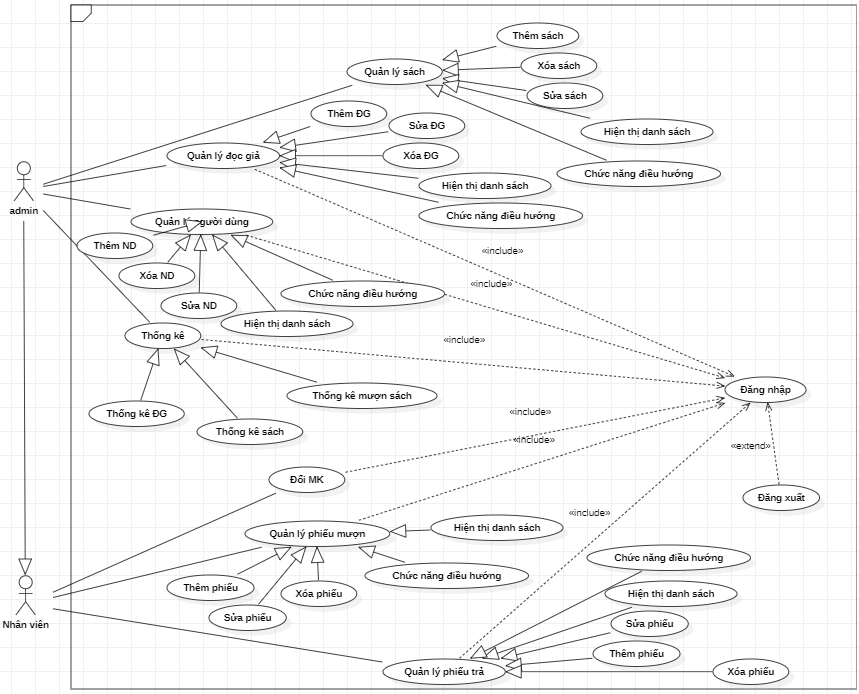
## Hiện trạng

Ở nước ta hiện nay, một số thư viện sách vẫn còn quản lý bằng cách thủ công như quản lý bằng sổ sách hay các phần mềm văn phòng như exel,… khiến cho việc thống kê trở nên khó khăn . Vì vậy chúng em đã chọn đề tài quản lý thư viện sách bằng phần mềm thiết kế bằng công nghệ Java swing và mọi dữ liệu được lưu bằng hệ quản trị dữ liệu database. Điều đó giúp cho việc quản lý thư viện trở nên dễ dàng hơn và có thể tăng hiệu suất công việc cao hơn.

## Yêu cầu hệ thống

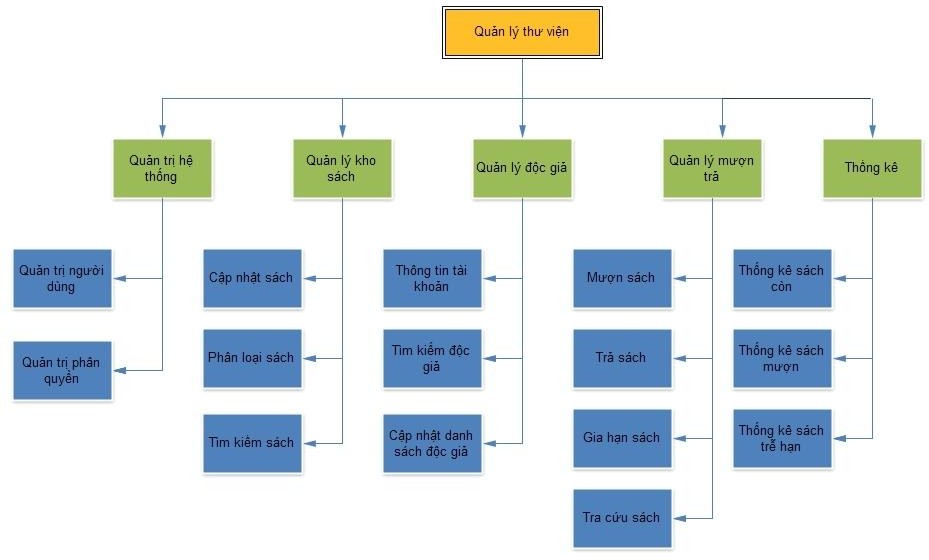
1. Hệ thống thư viện sách được xây dựng nhằm mục đích giải quyết các yêu cầu chức năng sau:
   * Giúp độc giả tra cứu sách theo loại sách, theo tên sách, theo tác giả, theo ngôn ngữ, … trên các máy tính trạm.
   * Cung cấp cho thủ thư thông tin về các đầu sách một độc giả đang mượn và hạn phải trả, và các cuốn sách còn đang được mượn.
   * Thống kê hàng tháng số sách cho mượn theo các chủ đề, tác giả,… Thống kê các đầu sách không có người mượn trên 1 năm, 2 năm, 3 năm.
   * Hỗ trợ thủ thư cập nhật thông tin sách, xác nhận cho mượn sách và nhận lại sách khi độc giả trả sách.
   * Hỗ trợ quản lý các thông tin về độc giả dựa trên thẻ độc giả, thông tin phiếu mượn.
   * Hỗ trợ chức năng quản trị chung hệ thống, trong đó người quản trị chung có thể thay đổi thông tin hoặc thêm bớt các thủ thư.
2. Yêu cầu phi chức năng:
   * Độc giả có thể tra cứu thông tin sách trên môi trường mạng nội bộ của thư viện. Tuy nhiên, việc mượn và trả sách phải thực hiện trực tiếp trên thư viện. Thủ thư sử dụng hệ thống để cập nhật và quản lý quá trình mượn trả sách.
   * Thông tin thống kê phải đảm bảo tính chính xác, khách quan. Các hình thức phạt đối với độc giả quá hạn sẽ được lưu lại và thông báo cho độc giả biết.

## Use case

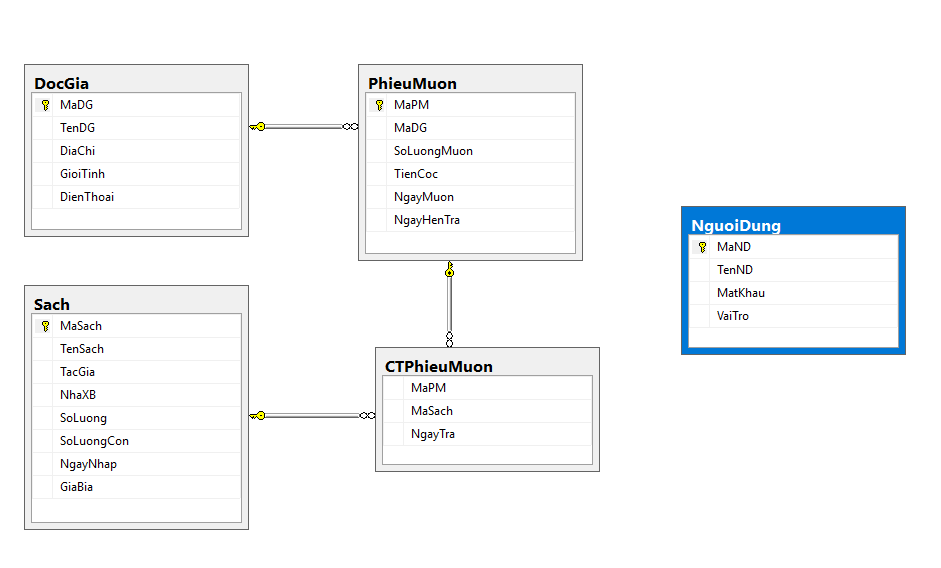


# Thiết kế

## Mô hình triển khai



## Thiết kế CSDL



### 2.2.1 Sơ đồ quan hệ thực thể

Phiếu mượn

Mượn

Sách

Nhập

Điền

Ghi

Người Dùng

Độc giả

### 2.2.2Thiết kế chi tiết các thực thể

##### Mô tả ca sử dụng quản lý độc giả:

* + Mục đích: sử dụng để quản lý thông tin độc giả, thêm, xóa, sửa thông tin độc giả, báo thẻ độc giả hết hạn.
  + Tác nhân kích hoạt: thủ thư.
  + Các bước tiến hành:
* Chọn chức năng quản lý độc giả, hệ thống sẽ hiển thị màn hình quản lý độc giả: danh sách các độc giả, các chức năng con thêm, xóa, sửa độc giả.
* Thủ thư thêm, xóa, sửa độc giả qua các form thêm độc giả, xóa, sửa thông tin độc giả.
* Ca sử dụng kết thúc.
  + Các trường hợp khác:
* Người dùng thoát khỏi hệ thống.
* Nếu có sẵn dữ liệu về độc giả thì cho phép sửa, xóa thông tin.
* Báo lỗi khi nhập trùng mã độc giả.
  + Điều kiện trước: người dùng phải đăng nhập trước khi sử dụng chức năng quản lý độc giả.
  + Điều kiện sau: hiển thị danh sách độc giả sau khi cập nhật thông tin.

##### Mô tả ca sử dụng quản lý mượn trả:

* Mục đích: quản lý quy trình mượn trả sách.
* Tác nhân kích hoạt: thủ thư.
* Các bước tiến hành:
  + Độc giả yêu cầu mượn( trả )sách, thủ thư xác nhận thông tin độc giả bằng chức năng tìm kiếm độc giả.
  + Xác nhận thông tin độc giả thành công, thủ thư kiểm tra thông tin sách độc giả yêu cầu mượn.
  + Thành công, thủ thư tạo phiếu mượn, yêu cầu độc giả thanh toán tiền cọc.
  + Với trường hợp trả sách, thủ thu kiểm tra thông tin phiếu mượn và thông tin sách mượn.
  + Thủ thư hoàn trả lại tiền cọc cho độc giả, cập nhật lại thông tin cho phiếu mượn.
* Các trường hợp khác:
  + Thẻ độc giả không đúng, báo vi phạm, độc giả không được phép mượn sách.
  + Sách độc giả muốn mượn đã hết.
  + Độc giả vi phạm nội quy quá số lần quy định, không được mượn sách.
  + Sách trả trễ hạn, hỏng, độc giả phải bổi thường theo quy định.
* Điều kiện trước: thủ thư phải đăng nhập hệ thống.
* Điều kiện sau: hiển thị thông tin phiếu mượn mới.

##### Mô tả ca sử dụng thống kê:

* Mục đích: giúp thủ thư nắm rõ tình trạng mượn trả sách, thống kê độc giả thân thiết và sách được mượn nhiều của thư viện.
* Tác nhân kích hoạt: thủ thư.
* Các bước tiến hành:
  + Thủ thư chọn chức năng thống kê, màn hình thống kê hiện ra.
  + Thủ thư chọn thống kê theo các tiêu chí khác nhau: thống kê sách, độc giả .... theo tháng, năm...
  + Màn hình hiển thị kết quả.
  + Ca sử dụng kết thúc.
* Các trường hợp khác:
  + Người dùng có thể thoát tại bất cứ thời điểm nào.
  + Các tiêu chí thống kê không đúng.
* Điều kiện trước: thủ thư phải đăng nhập hệ thống.
* Điều kiện sau: hiển thị danh sách thống kê theo yêu cầu.

##### Mô tả ca sử dụng quản trị hệ thống:

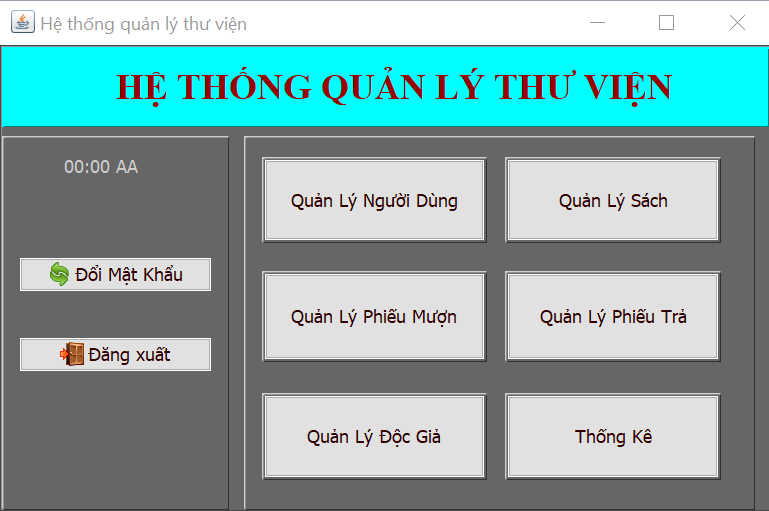
* Mục đích: quản lý vai trò, quyền hạn của người sử dụng hệ thống.
* Tác nhân kích hoạt: admin
* Các bước tiến hành:
  + Admin đăng nhập bằng tài khoản admin và chọn quản trị hệ thống.
  + Admin lựa chọn các chức năng thêm, xóa, sửa thông tin người dùng và phân quyền cho người dùng.
  + Ca sử dụng kết thúc.
* Các trường hợp khác:
  + Người dùng không phải admin chọn chức năng quản trị hệ thống, hệ thống sẽ báo lỗi, người dùng không được phép sử dụng chức năng này.
  + Admin sửa thông tin người dùng trong khi người dùng đang đăng nhập hệ thống, hệ thống sẽ báo lỗi và yêu cầu người dùng thoát khỏi hệ thống.
* Điều kiện trước: admin phải đăng nhập hệ thống với quyền admin.
* Điều kiện sau: hiển thị thông tin người dùng sau khi sửa đổi.

##### Mô tả ca sử dụng tìm kiếm:

* Mục đích: tìm kiếm thông tin sách, độc giả, tác giả...
* Tác nhân kích hoạt: thủ thư, độc giả.
* Các bước tiến hành:
  + Người dùng chọn chức năng tìm kiếm.
  + Màn hình tìm kiếm hiển thị, người dùng tiếp tục chọn tìm kiếm theo các tiêu chí có sẵn hoặc nhập từ khóa.
  + Màn hình kết quả tìm kiếm hiển thị.
  + Kết thúc ca sử dụng.
* Các trường hợp khác:
  + Người dùng có thể thoát tại bất cứ thời điểm nào.
  + Lỗi kết nối cơ sở dữ liệu, không tìm kiếm được thông tin người dùng cần.
* Điều kiện trước: có chức năng tìm kiếm
* Điều kiện sau: hiển thị danh sách kết quả tìm kiếm.

## Thiết kế giao diện

### Giao diện chính của phần mềm



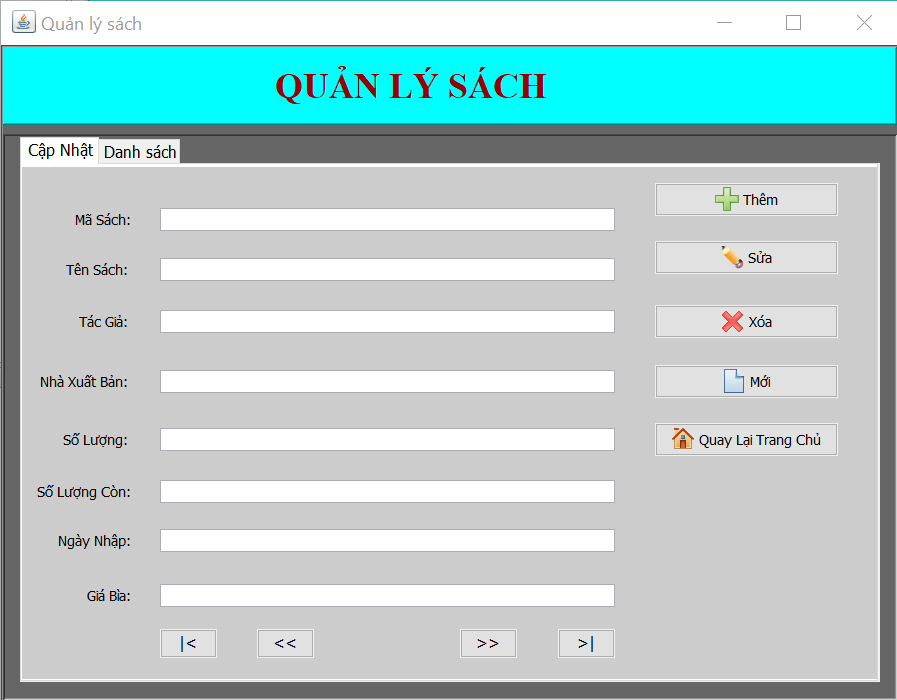
### Giao diện form Login



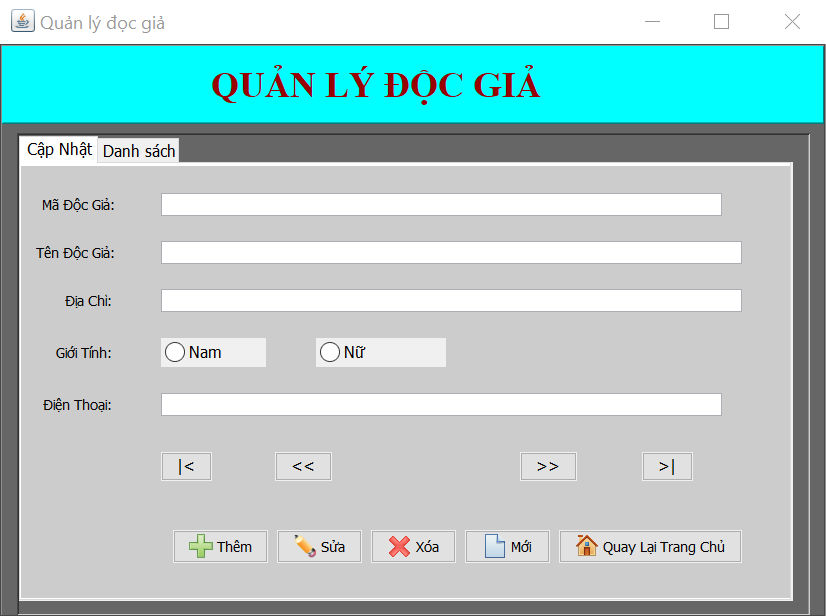
### Giao diện form Quản lý người dung

****

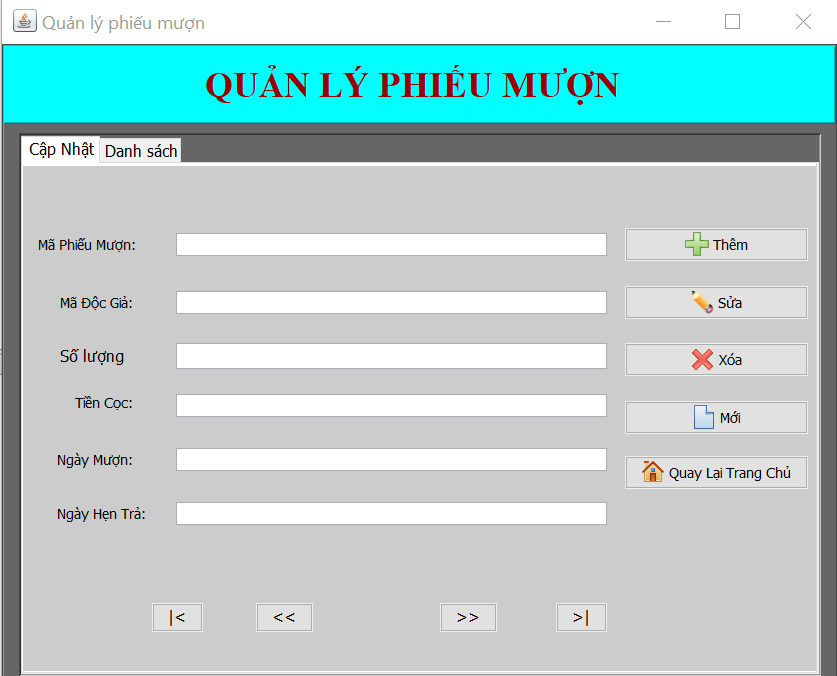
### Giao diện form Quản lý Sách



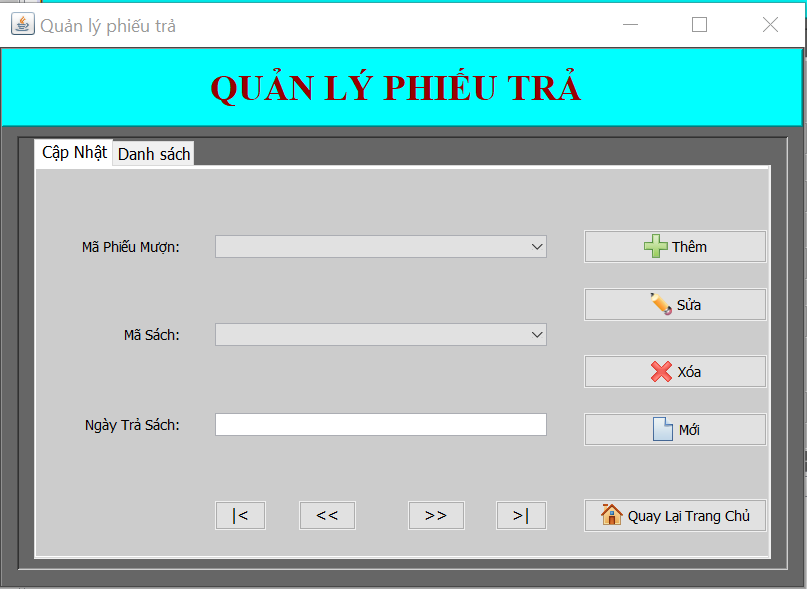
### Giao diện form Quản lý Độc giả



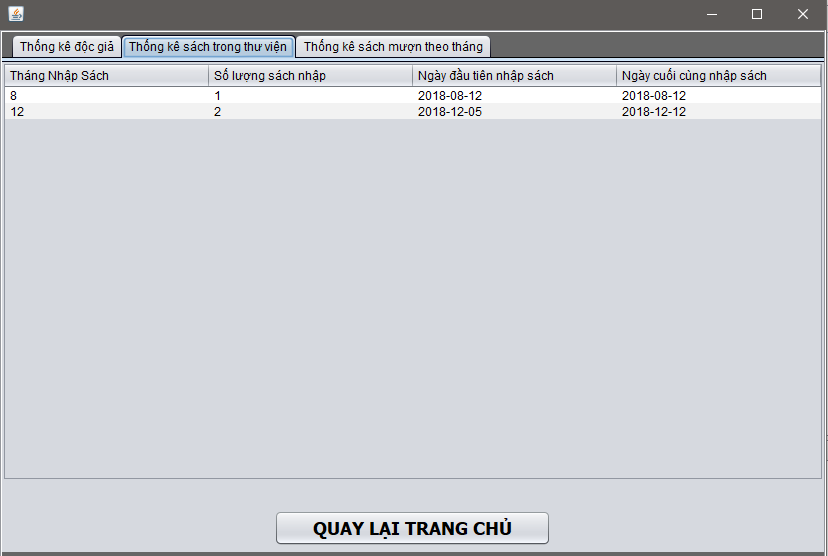
### Giao diện form Quản lý Phiếu mượn

****

### Giao diện form Quản lý Phiếu trả



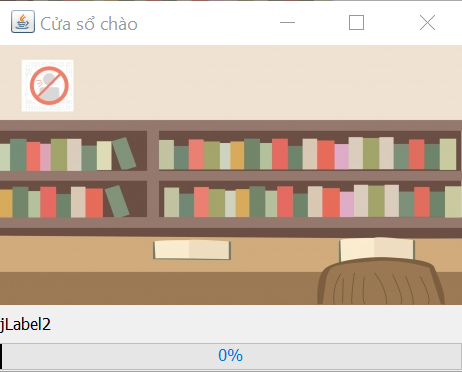
### Giao diện form Quản lý Thống Kê



### 2.3.9 Giao diện form Đổi mật khẩu



**2.4.0 Giao diện form Chao**

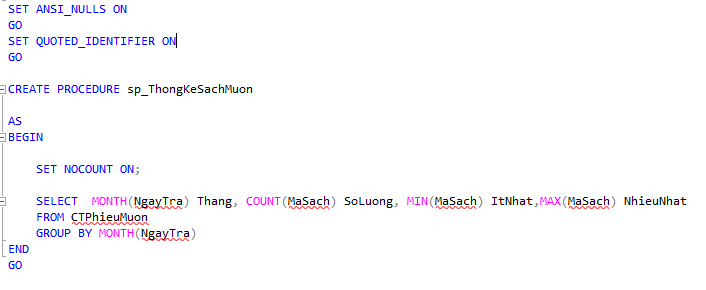
****

# Thực hiện viết mã

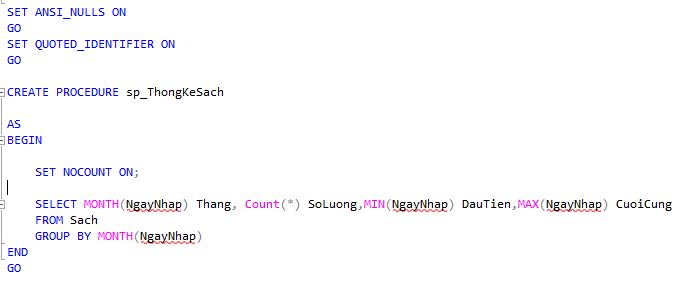
## Viết mã tạo CSDL

### Các thủ tục lưu tổng hợp thống kê

#### Procedure 1 – Thống kê phiếu mượn



#### Procedure 2 – Thống kê sách



## Lập trình JDBC

### Lớp hỗ trợ

#### Lớp tiện ích DateHelper

Chứa các phương thức chuyển đổi thời gian và chuỗi giúp lập trình sau này khi xử lý thời gian dễ dàng hơn.

package com.poly.helper;

import java.text.ParseException;

import java.text.SimpleDateFormat;

import java.util.Date;

public class DateHelper {

static final SimpleDateFormat DATE\_FORMATER = new SimpleDateFormat("MM/dd/yyyy");

public static Date toDate(String date, String... pattern) {

try {

if (pattern.length > 0) {

DATE\_FORMATER.applyPattern(pattern[0]);

}

if (date == null) {

return DateHelper.now();

}

return DATE\_FORMATER.parse(date);

} catch (Exception e) {

throw new RuntimeException(e);

}

}

public static String toString(Date date, String... pattern) {

if (pattern.length > 0) {

DATE\_FORMATER.applyPattern(pattern[0]);

}

if (date == null) {

date = DateHelper.now();

}

return DATE\_FORMATER.format(date);

}

public static Date now() {

return new Date();

}

public static Date addDay(Date date, int days) {

date.setTime(date.getTime() + days \* 24 \* 60 \* 60 \* 1000);

return date;

}

public static Date add(int days) {

Date now = DateHelper.now();

now.setTime(now.getTime() + days \* 24 \* 60 \* 60 \* 1000);

return now;

}

}

#### Lớp tiện ích JDBCHelper

package com.poly.helper;

import java.sql.Connection;

import java.sql.DriverManager;

import java.sql.PreparedStatement;

import java.sql.ResultSet;

import java.sql.SQLException;

public class JdbcHelper {

private static String Driver = "com.microsoft.sqlserver.jdbc.SQLServerDriver";

private static String dburl = "jdbc:sqlserver://localhost:1433;databaseName=SampleProject1";

private static String username = "sa";

private static String password = "123";

static {

try {

Class.forName(Driver);

} catch (Exception e) {

throw new RuntimeException(e);

}

}

public static PreparedStatement prepareStatement(String sql, Object... args) throws SQLException {

Connection con = DriverManager.getConnection(dburl, username, password);

PreparedStatement pst = null;

if (sql.trim().startsWith("{")) {

pst = con.prepareCall(sql);

} else {

pst = con.prepareStatement(sql);

}

for (int i = 0; i < args.length; i++) {

pst.setObject(i + 1, args[i]);

}

return pst;

}

public static void executeUpdate(String sql, Object... args) {

try {

PreparedStatement stm = prepareStatement(sql, args);

try {

stm.executeUpdate();

} finally {

stm.getConnection().close();

}

} catch (Exception e) {

e.printStackTrace();

throw new RuntimeException();

}

}

public static ResultSet executeQuery(String sql, Object... args) {

try {

PreparedStatement stm = prepareStatement(sql, args);

return stm.executeQuery();

} catch (SQLException e) {

throw new RuntimeException(e);

}

}

}

#### Lớp tiện ích DialogHelper

package com.poly.helper;

import java.awt.Component;

public class DialogHelper {

public static void alert(Component parent, String message) {

JOptionPane.showMessageDialog(parent, message, "Hệ thống quản lý thư viện", JOptionPane.INFORMATION\_MESSAGE);

}

public static boolean confirm(Component parent, String message) {

int result = JOptionPane.showConfirmDialog(parent, message,

"Hệ thống quản lý thư viện",

JOptionPane.YES\_NO\_OPTION, JOptionPane.QUESTION\_MESSAGE);

return result == JOptionPane.YES\_OPTION;

}

public static String prompt(Component parent, String message) {

return JOptionPane.showInputDialog(parent, message,

"Hệ thống quản lý thư viện", JOptionPane.INFORMATION\_MESSAGE);

}

}

### Model class - Các lớp mô tả dữ liệu

#### Sách

#### Người Dùng

#### Độc Giả

#### Phiếu Mượn

#### CT Phiếu Mượn

### DAO Class - Các lớp truy xuất dữ liệu

#### Lớp DocGiaDAO làm việc với bảng Quản Lý Độc Giả

#### Lớp NguoiDungDAO làm việc vs bảng Quản lý Người Dùng

#### Lớp PhieuMuonDAO làm việc với bảng Quản lý Phiếu Mượn

#### Lớp PhieuTraDAO làm việc với bảng Quản lý Phiếu Trả

#### Lớp SachDAO làm việc với bảng Quản lý Sach

#### Lớp CTPhieuMuonDAO làm việc với bảng Quản lý Quản Lý Chi Tiết Phiếu Mượn

***3.2.3.7 Lớp ThongKeDAO làm việc với bảng Thống Kê***

# Kiểm thử

## Kiểm thử form Login

Có tất cả 2 ô text box, khi người dung click nút button sẽ tiến hành validate thông tin người vừa nhập có hợp lệ hay không :

Username/Password: Không được để trống, Đã có tài khoản trên hệ thống

## Kiểm thử form Người dùng

Khi người dung click nút button sẽ tiến hành validate thông tin người vừa nhập có hợp lệ hay không :

Mã người dùng: Không được để trống, Độ dài có 8 kí tự trở lên, Không đc trùng với mã ND có sẵn

Họ Tên: chỉ được chứa alphabet và kí tự trắng.

Mật khẩu: phải từ 6 đến 32 ký tự.

Vai trò : Nếu chưa tích chọn vai trò thì phần mềm sẽ đưa ra thông báo để người dùng chọn.

## Các form quản lý (sách + phiếu mượn + phiếu trả + CT phiếu mượn)

Các ô ID thì không được nhập quá 8 kí tự và không được trùng mã.

Ô Name thì chỉ được chứa alphabet và kí tự trắng.

Email thì phải nhập đúng định dạng email

Số điện thoại thì phải là số.

Ngày nhập ngày trả thì phải đúng định dạng ngày.

* Các ô ko được để trống dữ liệu.

5 ĐÓNG GÓI VÀ TRIỂN KHAI5.1 HƯỚNG DẪN CHUYỂN ĐỔI JAR THÀNH EXE✓ Sử dụng phần mềm EXE4J để chuyển đổi jar sang exe  
✓ Sử dụng phần mềm InnoSetup để đóng gói  
5.2 HƯỚNG DẪN CÀI ĐẶT TRIỂN KHAI✓ Cài SQL Server 2008 trở lên  
✓ Tạo CSDL SampleProject1 bằng cách chạy file QuanLyThuVien.sql  
✓ Cài đặt JDK 1.8 trở lên  
✓ Chạy file setup  
5.3 HƯỚNG DẪN SỬ DỤNG PHẦN MỀM✓ Đăng nhập với tài khoản username: admin và password 123456

✓ Cấp các tài khoản có vai trò là Nhân viên , Admin  
✓ Xem hướng dẫn sử dụng trong menu trợ giúp của phần mềm

# 6 Tổng Kết

**6.1 Ưu điểm**

- Phần mềm đơn giản, dễ hiểu và dễ sử dụng.

- Bảo mật tối đa được dữ liệu.

## 6.2 Nhược điểm

- Số lượng rủi ro và sự không chắc chắn cao.

- Thiếu sự chú trọng vào thiết kế và tài liệu cần thiết.

**6.3 Hướng Phát Triển**

- Cải tiến lại giao diện đẹp hơn

- Phát triển phần mềm sang hướng Mobile và Web.