**TRƯỜNG ĐẠI HỌC KINH TẾ - KỸ THUẬT CÔNG NGHIỆP**

**KHOA CÔNG NGHỆ THÔNG TIN**

**ĐỒ ÁN 1**

QUẢN LÝ HỆ THỐNG THƯ VIỆN

**Giảng viên hướng dẫn: Vũ Anh Tuấn**

**Sinh viên thực hiện:**

**Vũ Hoài Ngọc**

**Nguyễn Minh Trí**

**Nguyễn Đình Phúc**

**Bùi Nhật Minh**

**Đinh Trường Phước**

**Hà Nội - 2023**

**Nội dung**

[**PHẦN I: MÔ TẢ DỰ ÁN** 2](#_Toc134795704)

[**PHẦN II: MÔ TẢ CHI TIẾT CHỨC NĂNG NGHIỆP VỤ CỦA HỆ THỐNG.** 3](#_Toc134795705)

[1. Chức năng quản lí bạn đọc 3](#_Toc134795706)

[2. Chức năng quản lí tài liệu 3](#_Toc134795707)

[3. Chức năng đặt mua, bổ sung 3](#_Toc134795708)

[4. Chức năng quản lí mượn trả 4](#_Toc134795709)

[5. Chức năng báo cáo và thống kê 4](#_Toc134795710)

[**PHẦN III: SƠ ĐỒ LUỒNG DỮ LIỆU SƠ ĐỒ LIÊN KẾT THỰC THỂ VÀ SƠ ĐỒ DIAGRAM MÔ TẢ CÁC CHỨC NĂNG CỦA HỆ THỐNG** 5](#_Toc134795711)

[**PHẦN IV: DEMO BÀI TOÁN QUẢN LÍ THƯ VIỆN TRÊN WINDOWNS FORM** 17](#_Toc134795712)

[1. Form Design 17](#_Toc134795713)

[2. Form bạn đọc 19](#_Toc134795714)

[3. Form quản lí nhân viên: 25](#_Toc134795715)

[4. Form quản lí nhà cung cấp: 26](#_Toc134795716)

[5. Form quản lí sách: 27](#_Toc134795717)

[6. Form quản lí phiếu nhập: 28](#_Toc134795718)

[7. Form quản lí phiếu nhập sách: 29](#_Toc134795719)

[8. Form quản lí phiếu mượn trả: 30](#_Toc134795720)

[9. Form quản lí phiếu mượn trả sách: 31](#_Toc134795721)

[10. Code chi tiết cho tất các form nêu trên: 31](#_Toc134795722)

[ **Form đăng nhập:** 31](#_Toc134795723)

[ **Form Menu** 33](#_Toc134795724)

[ **Form quản lí bạn đọc:** 35](#_Toc134795725)

[ **Form nhân viên:** 40](#_Toc134795726)

[ **Form nhà cung cấp:** 45](#_Toc134795727)

[ **Form sách:** 50](#_Toc134795728)

[ **Form phiếu mượn trả:** 54](#_Toc134795729)

[ **Form phiếu mượn trả sách:** 59](#_Toc134795730)

[ **Form phiếu nhập:** 64](#_Toc134795731)

[ **Form phiếu nhập sách:** 70](#_Toc134795732)

[**PHẦN V: KẾT LUẬN** 75](#_Toc134795733)

# **PHẦN I: MÔ TẢ DỰ ÁN**

Trong xã hội ngày nay công nghệ thông tin ngày càng phát triển, con người càng ngày càng thuộc về các thiết bị, phần mềm, sản phẩm và công nghệ. Một người bước chân ra ngoài đường có thể sử dụng một số ứng dụng nhỏ trên điện thoại để đặt xe đón, mua đồ ăn, tìm đường, thậm chí là đặt vé du lịch và giao dịch tiền tệ… Công nghệ thông tin có những hạn chế nhất định khi trả lại tác hại là ảnh hưởng đến sức khỏe con người, làm con người phải thuộc về nó rất nhiều. Nhưng nhìn nhận ở mặt tích cực hoạt động nghệ thuật thông tin đã đóng góp rất nhiều vào các lĩnh vực khoa học, văn hóa và xã hội. Đặc biệt trong các vấn đề thuộc phạm vi lĩnh vực văn hóa - xã hội nghệ thuật thông tin đã giúp ích rất nhiều. Ở đây, trong bài viết này chúng em muốn nói đến là ứng dụng công nghệ thông tin để góp phần xây dựng, bảo tồn và phát triển văn bản: hệ thống quản lí thư viện.

Công cụ Công nghệ thông tin nâng cao năng suất hoạt động của thư viện nhờ vào tính năng hỗ trợ tích cực cho công việc quản lý; đa dạng sản phẩm, thư viện dịch vụ - thông tin, chia sẻ tài nguyên thông tin và nâng cao vai trò, vị trí của thư viện, những thay đổi hay cải tiến trong thư viện đều phải tính đến nhu cầu và sự thuận lợi cho bạn đọc. Đẩy mạnh đồng thời, hỗ trợ bạn đọc trong tiếp cận, sử dụng thông tin và tăng khả năng trả lời những yêu cầu rất khác nhau của từng nhóm đối tượng trong cộng đồng phục vụ.

Với mục tiêu được xây dựng một hệ thống quản lý chất lượng, khoa học và hệ thống thân thiện phải đảm bảo những yêu cầu sau:

* Thực hiện các chức năng tốt.
* Hệ thống phải dễ sử dụng, nhanh chóng truy cập dữ liệu và chính xác, đơn giản thao tác.
* Giao diện thân thiện, khoa học.
* Search, nhanh chóng thống kê, theo yêu cầu chính xác.
* Có đầy đủ các tính năng của một nhà quản lý hệ thống.
* Tiếp nhận phản hồi và đánh giá của bạn đọc, qua đó sẽ có những điều chỉnh thích hợp về chế độ làm việc của nhân viên thư viện một cách hợp lý hơn.
* Bạn đọc có thể đặt mượn sách online trước khi đến thư viện để có thể rút ngắn thời gian mượn sách và tiện lợi hơn.

Các nhiệm vụ cơ bản trong hệ thống quản lí thư viện:

* Quản lí bạn đọc
* Quản lí tài liệu
* Quản lí mượn trả
* Đặt mua, bổ sung
* Báo cáo và thống kê

# **PHẦN II: MÔ TẢ CHI TIẾT CHỨC NĂNG NGHIỆP VỤ CỦA HỆ THỐNG.**

## Chức năng quản lí bạn đọc

* 1. **Làm thẻ bạn đọc**

Một bạn đọc muốn vào thư viên mượn sách thì nhân viên quản lí cần xác minh danh tính của người mượn và để cho nhân viên thư viện dễ quản lí thông tin người mượn sách thì bạn đọc cần phải làm thẻ thư viện( thẻ bạn đọc) và mỗi lần tới mượn sách thi bạn đọc cần xuất trình thẻ thư viện thì mới được mượn sách ở trung tâm thư viện.

* 1. **Cập nhật thẻ bạn đọc**

Sau khi đăng kí làm thẻ thư viện nhân viên thư viện cập nhật thẻ bạn đọc lên hệ thống để tiện cho người đọc để có thể đăng kí mượn sách cho những lần mượn tiếp theo.

* 1. **Tìm kiếm thông tin bạn đọc**

Khi bạn đọc mượn sách nhân viên thư viện có nhiệm vụ tìm kiếm thông tin bạn đọc theo thẻ bạn đọc đã được cấp và tìm kiếm và kiểm tra tình trạng hiện tại của đầu sách được mượn rồi mới cho mượn sách.

* 1. **Xóa thẻ bạn đọc**

Khi bạn đọc không còn nhu cầu mượn sách hoặc thẻ bạn đọc đã hết hạn sử dụng thì nhân viên thư viện sẽ thực hiện công việc xóa thẻ bạn đọc.

## Chức năng quản lí tài liệu

* 1. **Tìm kiếm thông tin tài liệu**

Khi có bạn đọc đến mượn tài liệu thì bộ phận quản lí sẽ có nhiệm vụ tìm kiếm trên hệ thống thông tin tài liệu mà bạn đọc muốn mượn.

* 1. **Cập nhật thông tin tài liệu**

Cập nhật thông tin tài liệu( những tài liệu nào được tái bản mới thì sẽ được cập nhật lại để đem lại những nội dung gần gũi và chân thực nhất đối với bạn đọc)

* 1. **Xóa bỏ tài liệu quá hạn hư hỏng**

Bên cạnh đó, bộ phận quản lí cũng là bộ phận trực tiếp làm nhiệm vụ quản lý kho sách của thư viện để có thể nắm bắt được rất rõ tình hình hiện trạng các đầu sách của thư viện. Với những đầu sách mà số lượng sách hư hỏng hay nhu cầu cần phải đáp ứng quá nhiều.

Xử lí những tài liệu đã quá hạn hư hỏng.

## Chức năng đặt mua, bổ sung

* 1. **Thêm yêu cầu bổ sung tài liệu**

Nhân viên thư viện có nhiệm vụ quản lí và bổ sung các loại tài liệu để có thể đem lại nguồn sách phong phú cho bạn đọc thỏa sức lựa chọn. thư viện có kế hoạch mua, bổ sung thêm sách, bộ phận quản lí đặt mua, bổ sung sẽ thống kê và duyệt đối với các yêu cầu bổ sung sách, xem xét nhu cầu, cân đối kinh phí và lên một danh sách các đầu sách sẽ được đặt mua

* 1. **Lên danh sách tài liệu đặt mua**

Sau khi thông kê các mặt sách cần bổ sung thì nhân viên quản lí lên danh sách các tài liệu đặt mua và gửi qua cho nhà cung cấp.

* 1. **Tạo yêu cầu đặt/ in mua tài liệu**

Tạo phiếu khảo sát lấy ý kiến của bạn đọc về các mặt sách đang thịnh hành để bộ phận quản lí thư viện có thể bổ sung kịp thời một cách đầy đủ để phục vụ cho bạn đọc có một kho tài liệu hứng thủ và bổ ích.

## Chức năng quản lí mượn trả

* 1. **Ghi nhận thông tin tài liệu mượn**

Nếu có bạn đọc đến mượn được sách thì nhân viên quản lí thư viện sẽ yêu cầu bạn đọc đưa ra thẻ bạn đọc và mã thẻ để có thể nhập lên hệ thống và xác thực danh tín của bạn đọc rồi mới cho mượn sách. Thủ thư của thư viện cần ghi nhận mã số thẻ và ghi nhận thông tin các tài liệu mà bạn đọc đã mượn vào cơ sở dữ liệu .

* 1. **Ghi nhận thông tin tài liệu được trả**

Khi bạn đọc đến trả sách thì nhân viên quản lí thư viện sẽ kiểm tra sách xem sách có bị rách nát hay không còn nguyên vẹn như lúc cho mượn không. Nếu mọi sách bình thường và không có dấu hiệu hư hỏng thì nhân viên quản lí sẽ in phiếu đã trả sách cho bạn đọc và sửa lại nhưng thay đổi đã lưu trên hệ thống về việc mượn sách.

* 1. **In phiếu mượn sách và yêu cầu đền sách**

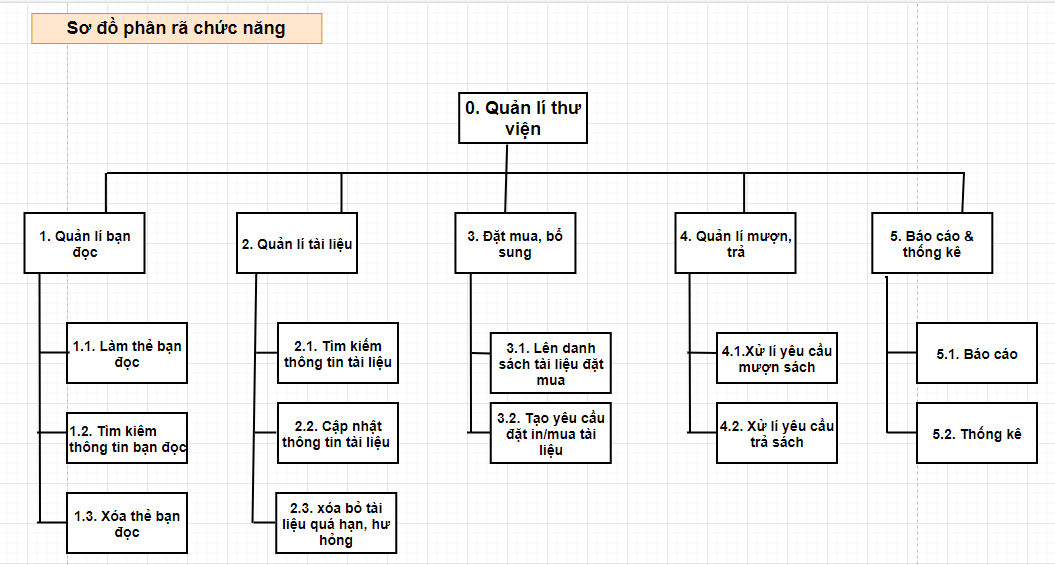
Nếu khi đến trả sách bạn đọc làm mất tài liệu hoặc cố tình không trả tài liệu hoặc đã làm cho tài liệu có dấu hiệu hư hỏng thì nhân viên quản lí sẽ in ra hóa đơn yêu cầu đền tài liệu trong trường hợp làm mất tài liệu hoặc phiếu yêu cầu đền bù trong trường hợp làm tài liệu hư hỏng.

## Chức năng báo cáo và thống kê

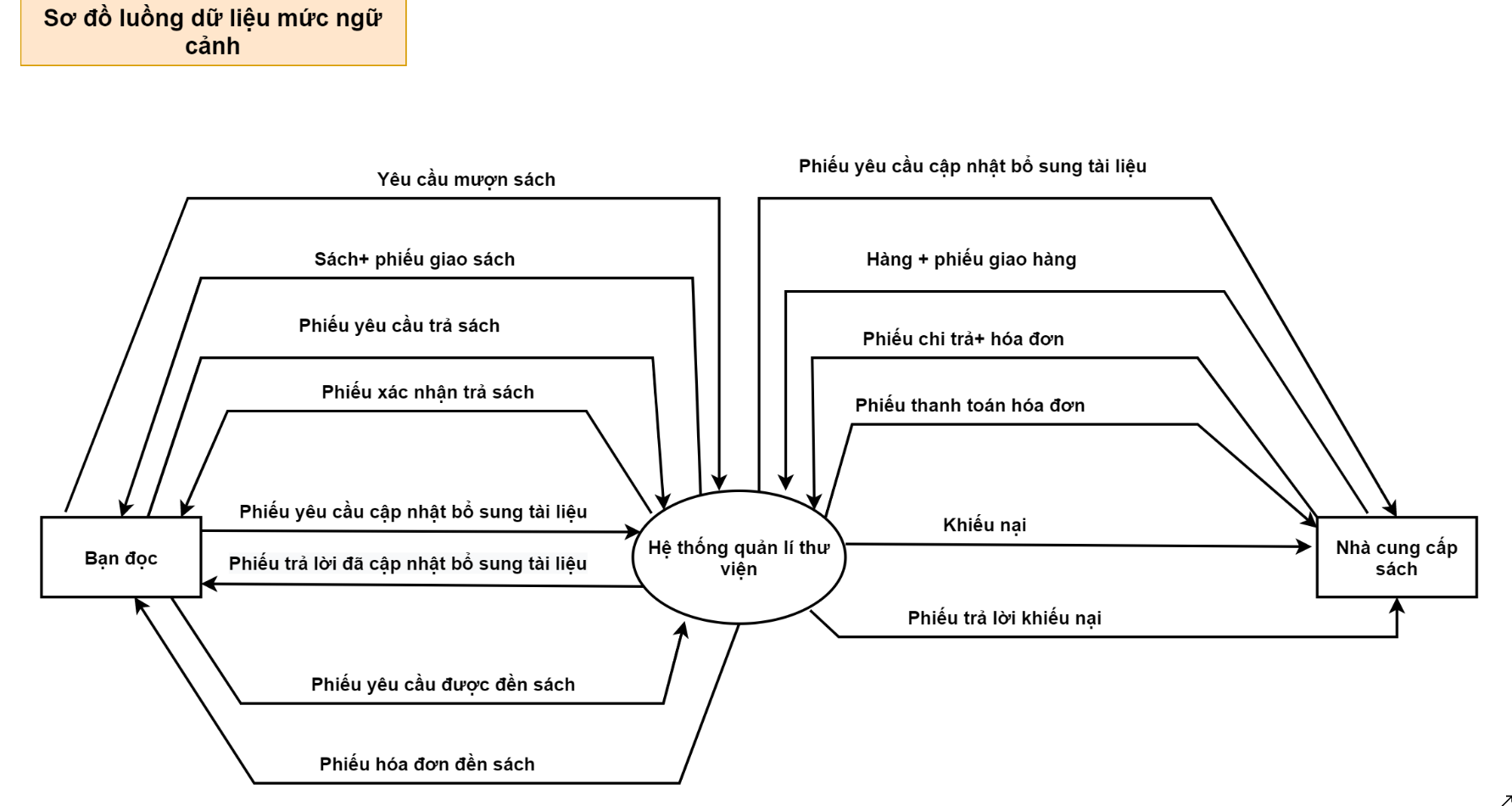
Nhân viên thư viện có nhiệm vụ thống kê và báo cáo lại với nhà quản lí (giám đốc thư viện) về những tình tiết đã diễn ra trong thư viện: số sách được mượn trong ngày, trong tháng, hóa đơn đền sách, tài liệu, danh sách tài liệu cập nhật và bổ sung.

# **PHẦN III: SƠ ĐỒ LUỒNG DỮ LIỆU SƠ ĐỒ LIÊN KẾT THỰC THỂ VÀ SƠ ĐỒ DIAGRAM MÔ TẢ CÁC CHỨC NĂNG CỦA HỆ THỐNG**

1. **Sơ đồ luồng dữ liệu**



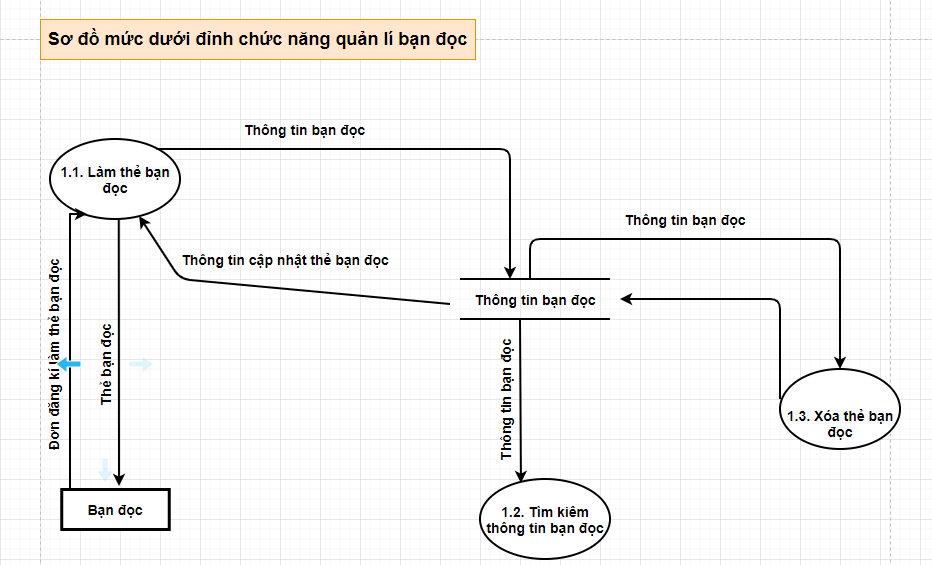
Hình 1: Sơ đồ phân cấp chức năng



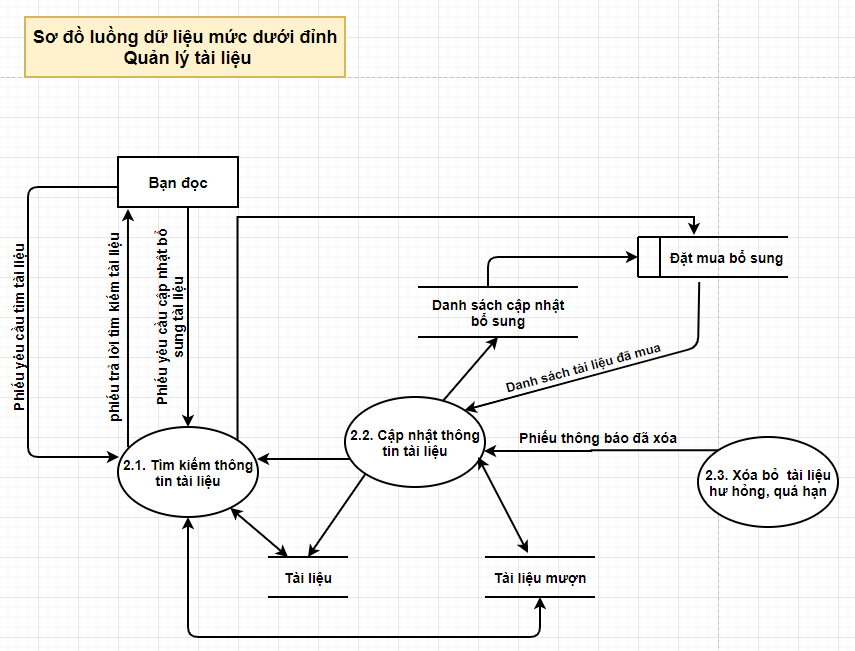
Hình 2: Sơ đồ mức dữ liệu mức ngữ cảnh



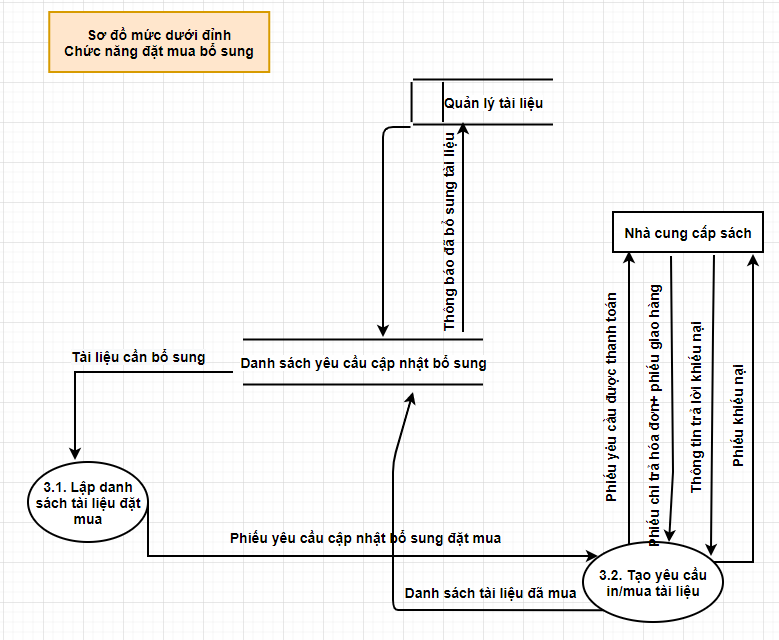
Hình 3: Sơ đồ mức dữ liệu mức đỉnh



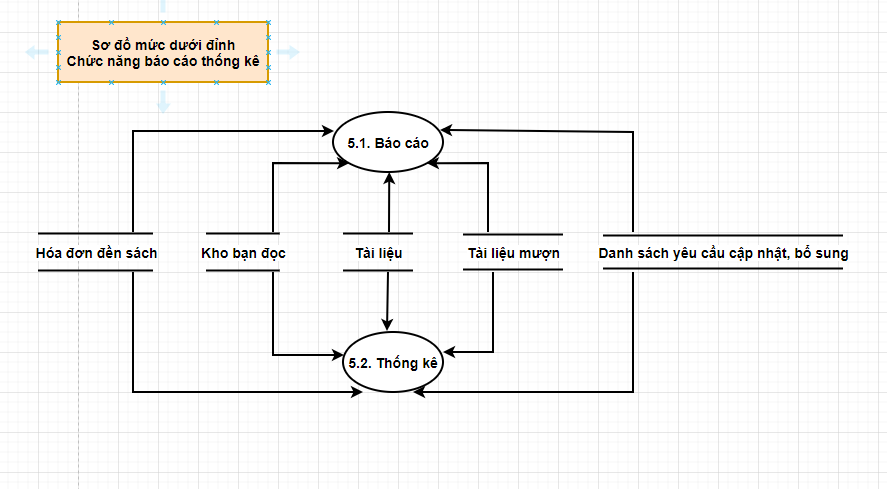
Hình 4: Sơ đồ mức dưới đỉnh chức năng quản lí bạn đọc



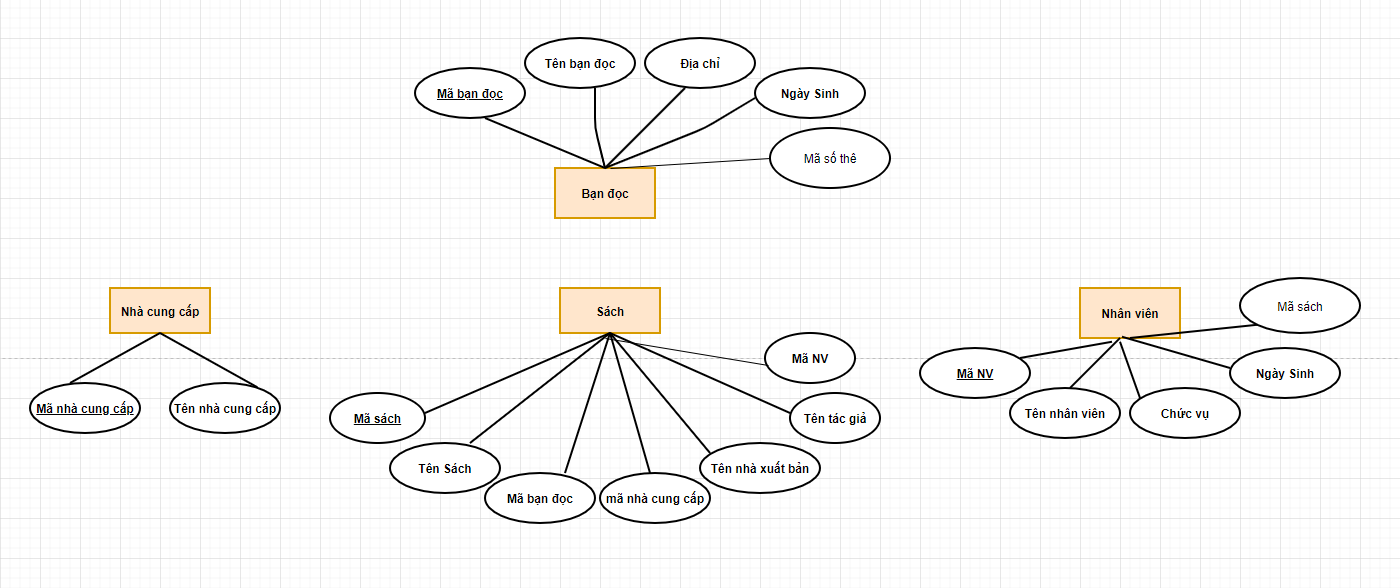
Hình 5: Sơ đồ mức dưới đỉnh chức năng quản lí tài liệu

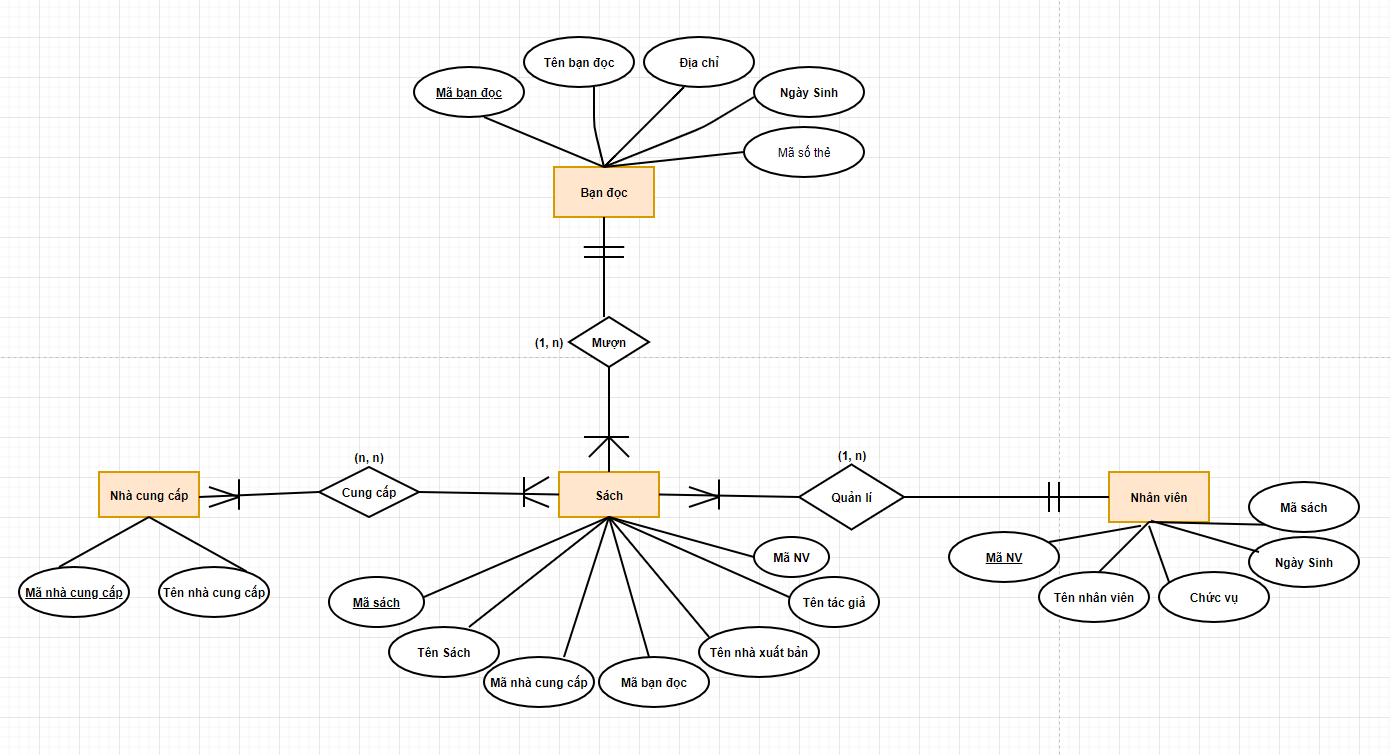
Hình 6: Sơ đồ mức dưới đỉnh chức năng đặt mua bổ sung

Hình 7: Sơ đồ mức dưới đỉnh chức năng quản lí mượn trả

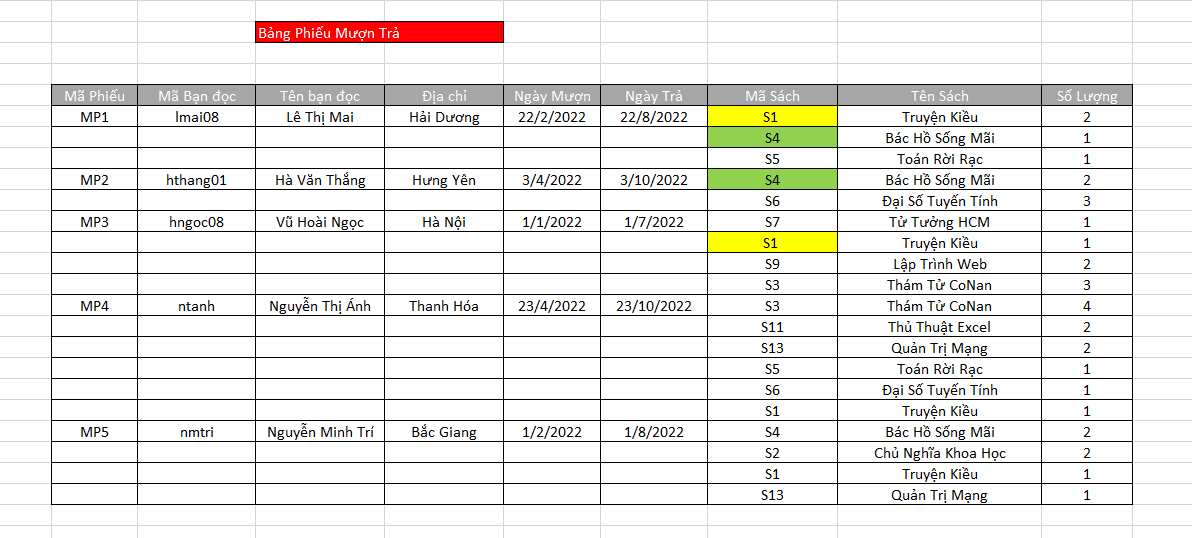
****Hình 8: Sơ đồ mức dưới đỉnh chức năng báo cáo và thông kê

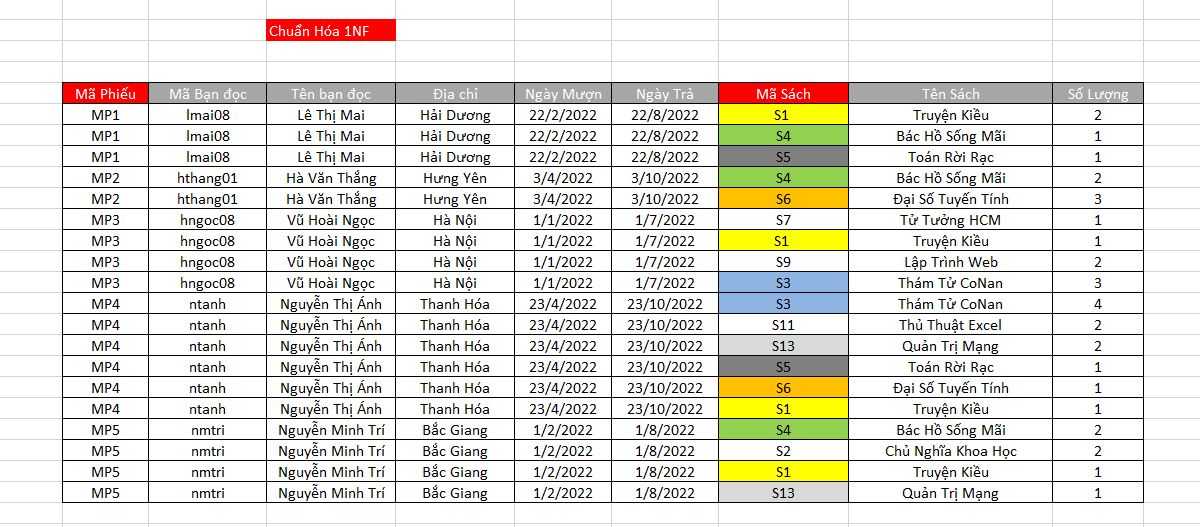
1. **Sơ đồ liên kết thực thể**

Hình 9: Sơ đồ thực thể

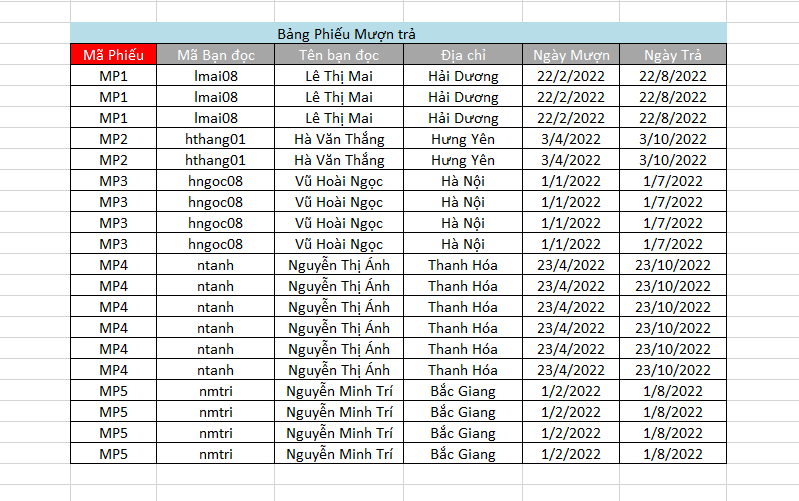
Hình 10: Sơ đồ liên kết thực thể

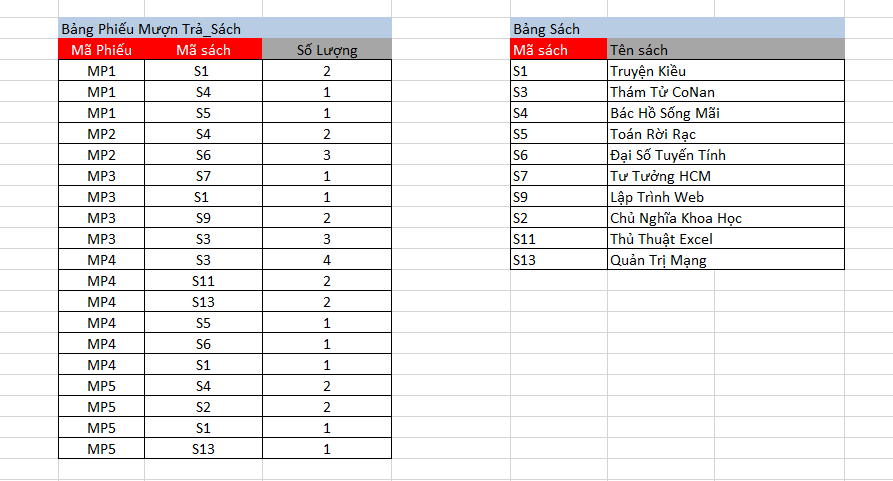
1. **Sơ đồ diagram**
   * Quá trình chuẩn hóa dữ liệu phiếu mượn trả sách về dạng chuẩn 3NF

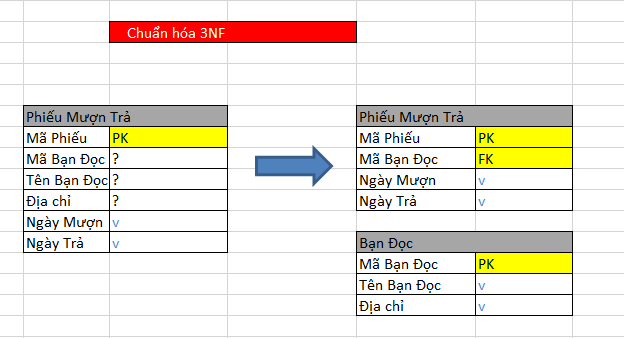
****

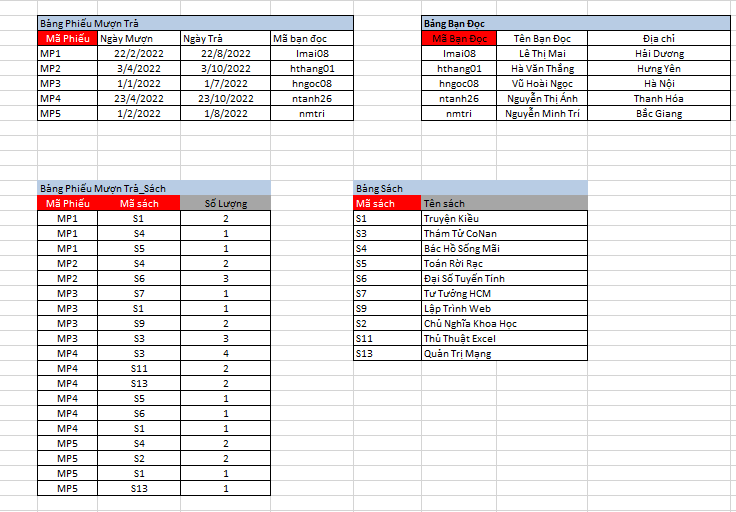
****

****

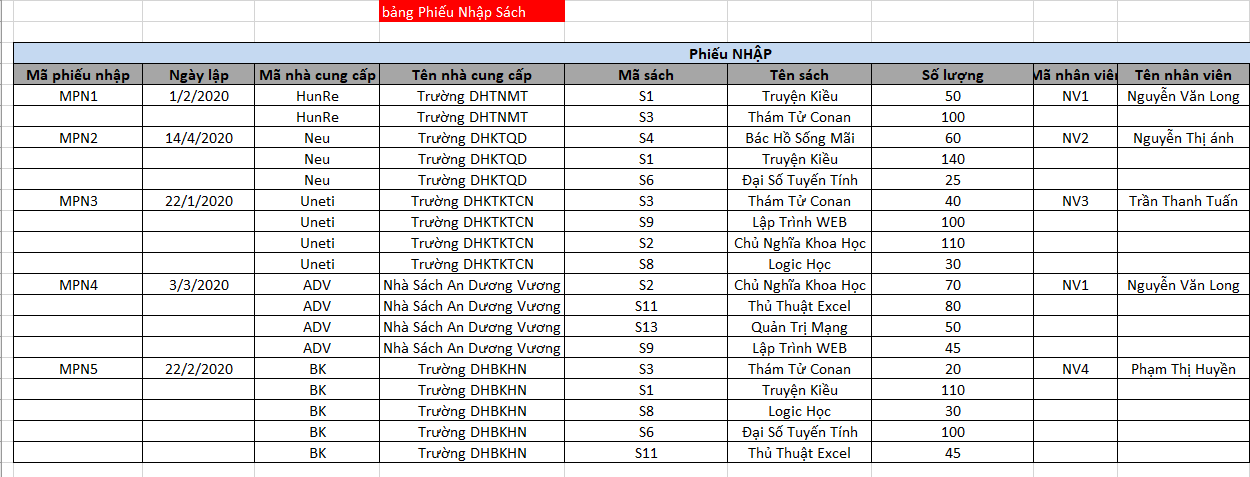
****

****

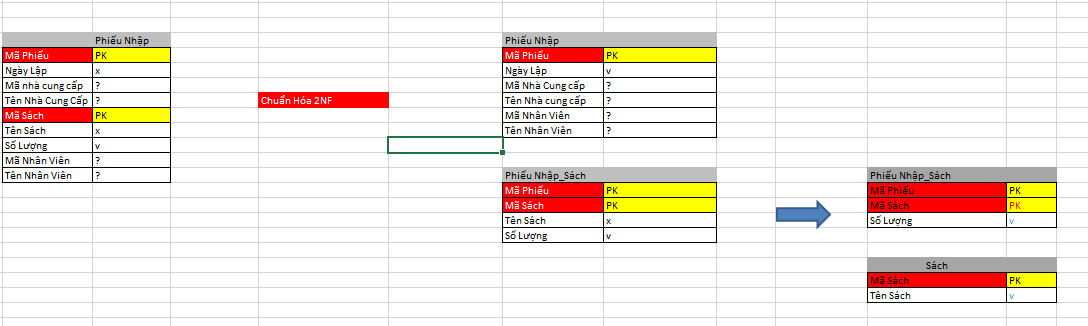
****

****

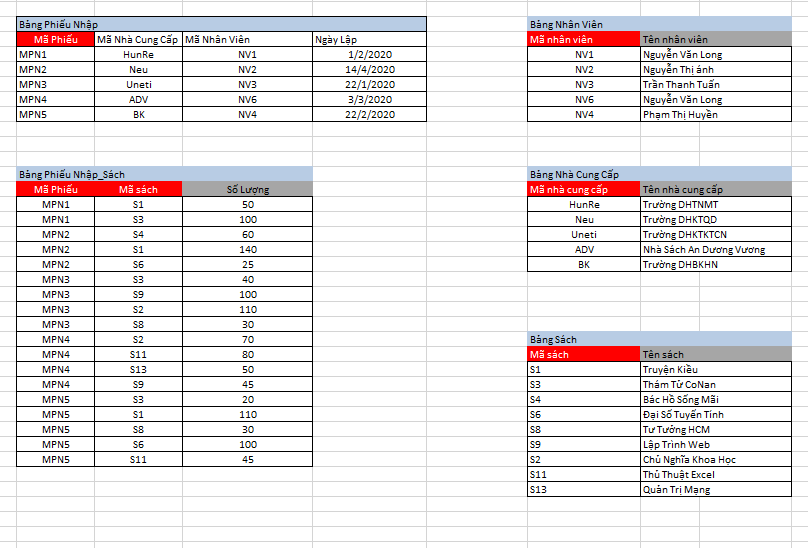
* + Chuẩn hóa phiếu nhập sách

****

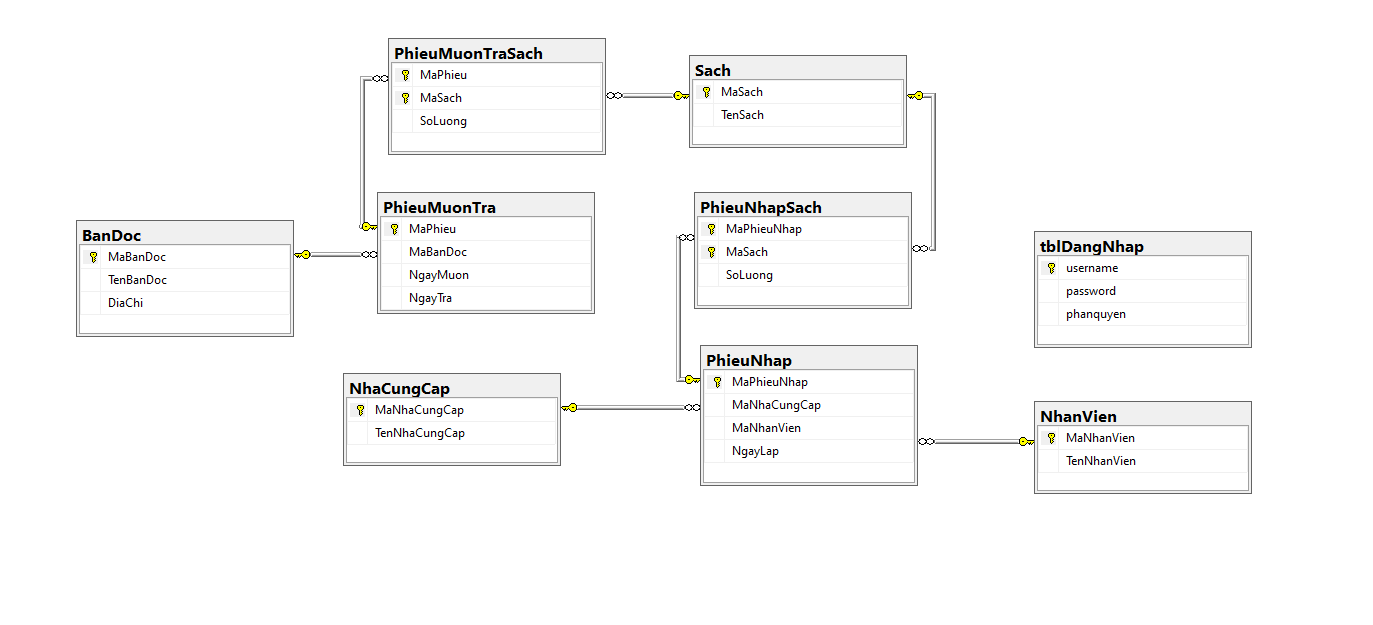
****

****

****

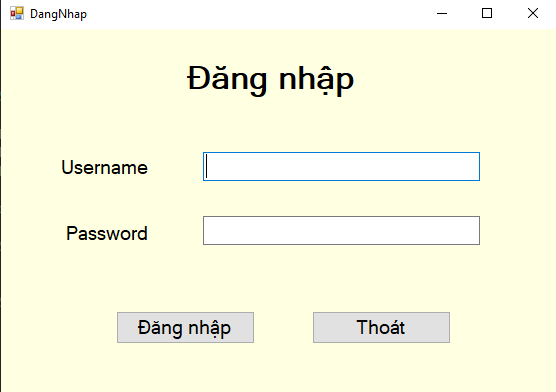
****

* + Sau khi chuẩn hóa dữ liệu về dạng 3NF thì ta sẽ thu được database cơ sở dữ liệu có mô hình diagram như sau**:**

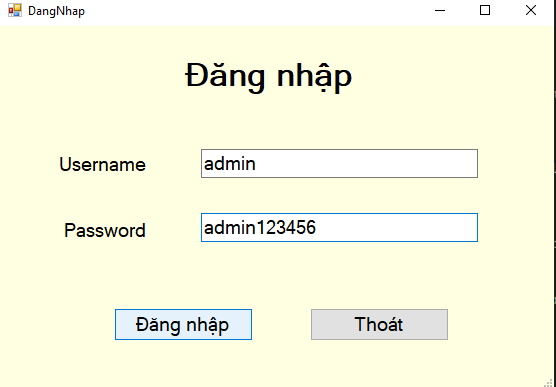
****Hình 11: Sơ đồ diagram

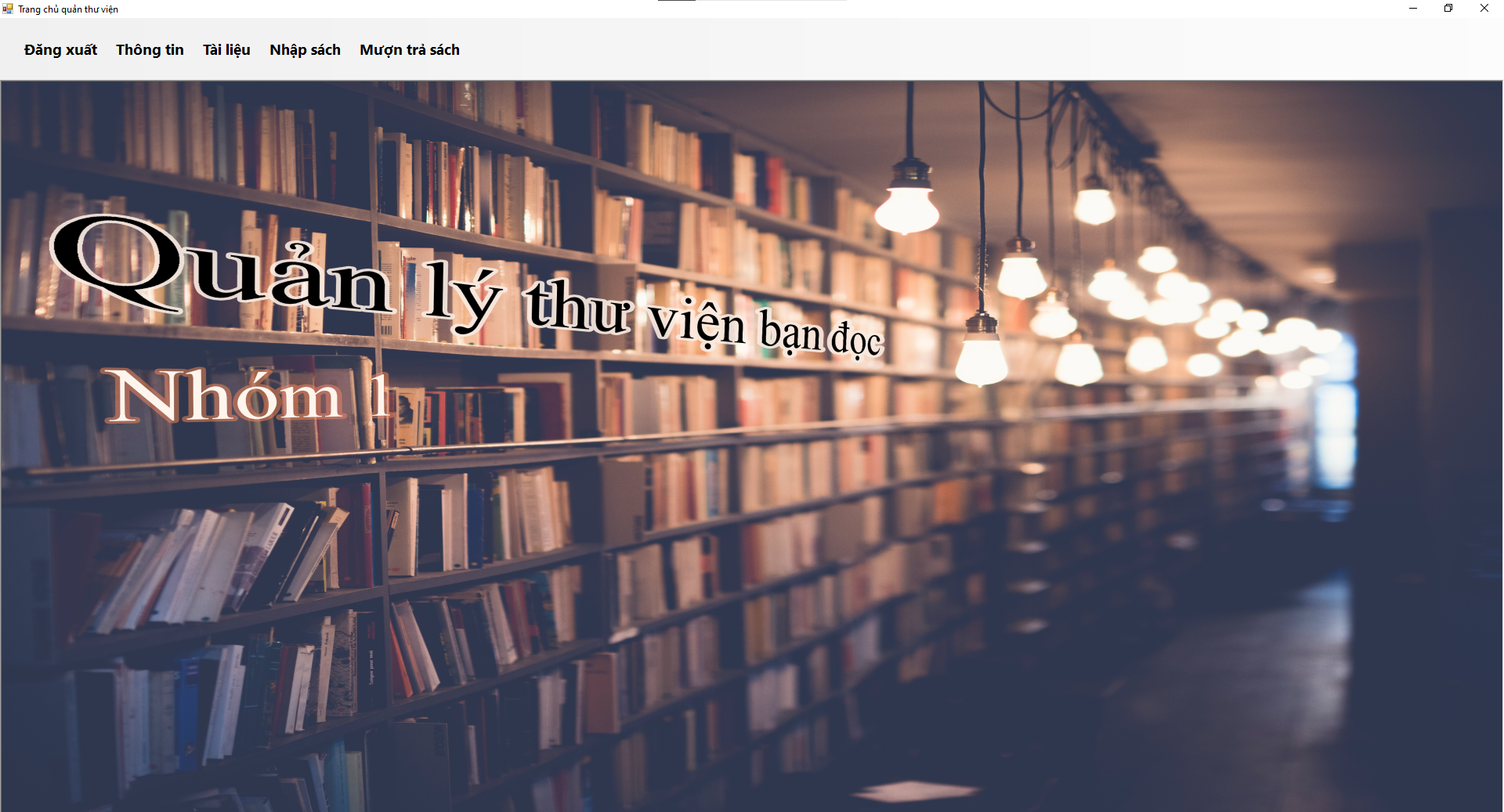
# **PHẦN IV: DEMO BÀI TOÁN QUẢN LÍ THƯ VIỆN TRÊN WINDOWNS FORM**

## Form Design

****

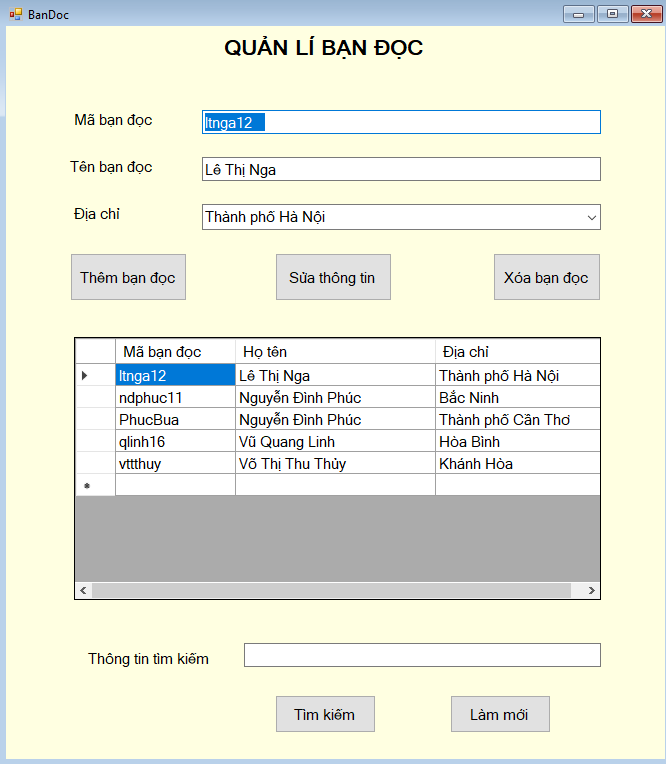
* + Khi ta nhập tài khoản của Admin quản lí thư viện vào thì sẽ đi tới trang chủ của thư viện để thực hiện các chức năng trên hệ thống



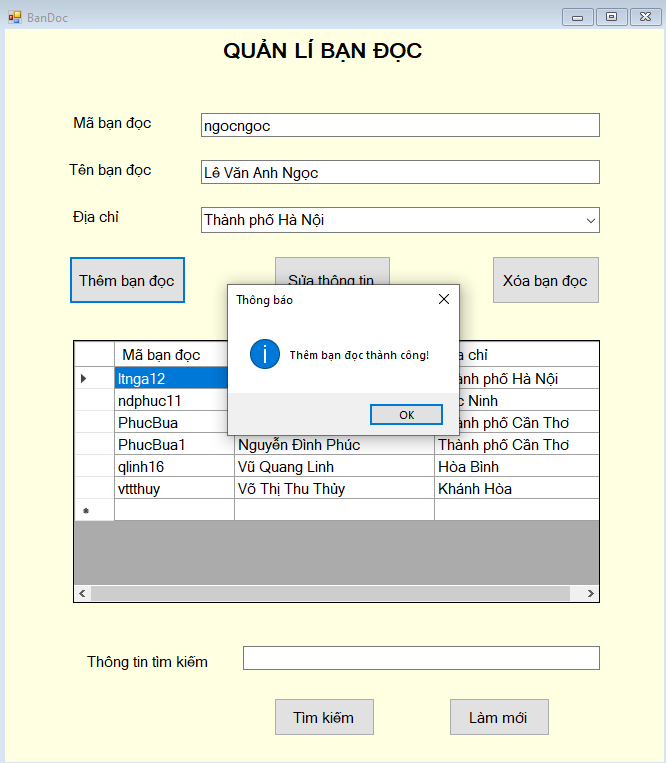


* + Trong đây ta sẽ có các chức năng như quản lí bạn đọc, quản lí nhân viên, quản lí nhà cung cấp, quản lí mượn trả sách và nhập sách. Trong các chức năng đó thì sẽ có các chức năng nhỏ hơn ví dụ như là thêm thông tin, thay đổi thông tin, xóa thông tin hay là tìm kiếm thông tin do người dùng đảm nhiệm. Sau đây ta sẽ đi vào lần lượt các form thể hiện các chức năng đó

## Form bạn đọc

****

Hình 12: Form quản lí bạn đọc

****

Hình 13: Thêm bạn đọc

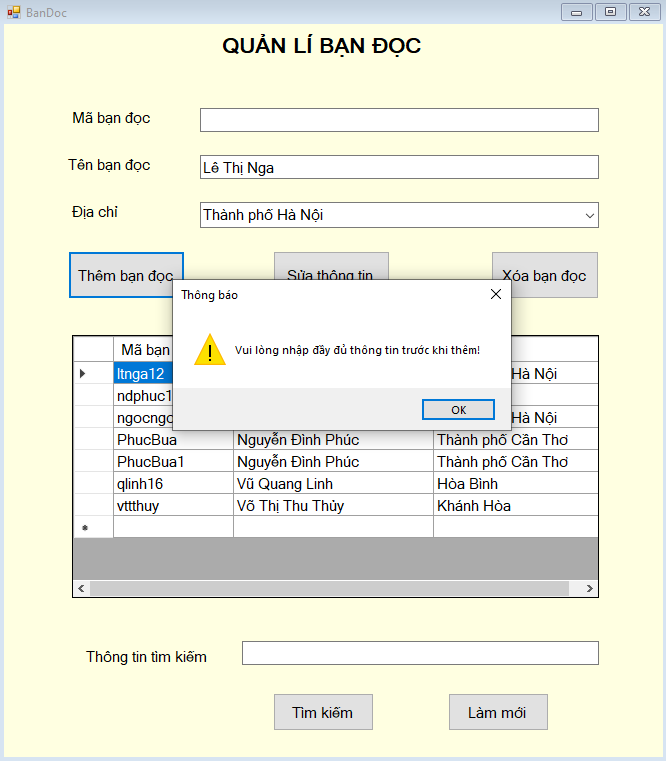
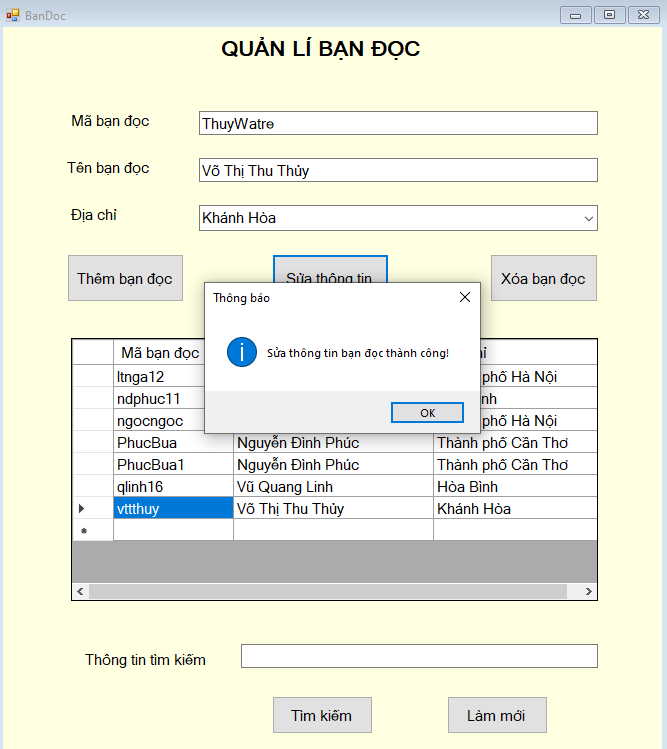
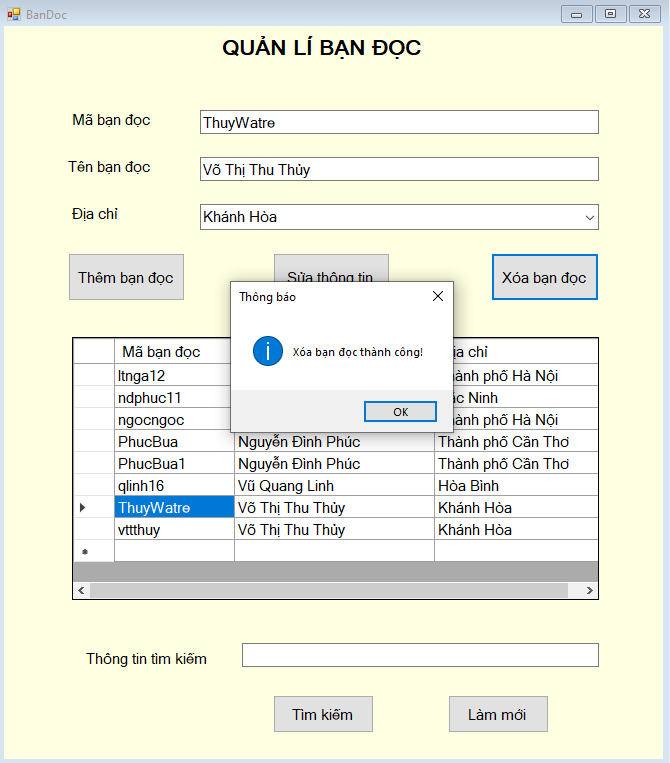


Table 1: Trường hợp người dùng nhập chưa đầy đủ dữ liệu đã bấm nút thêm

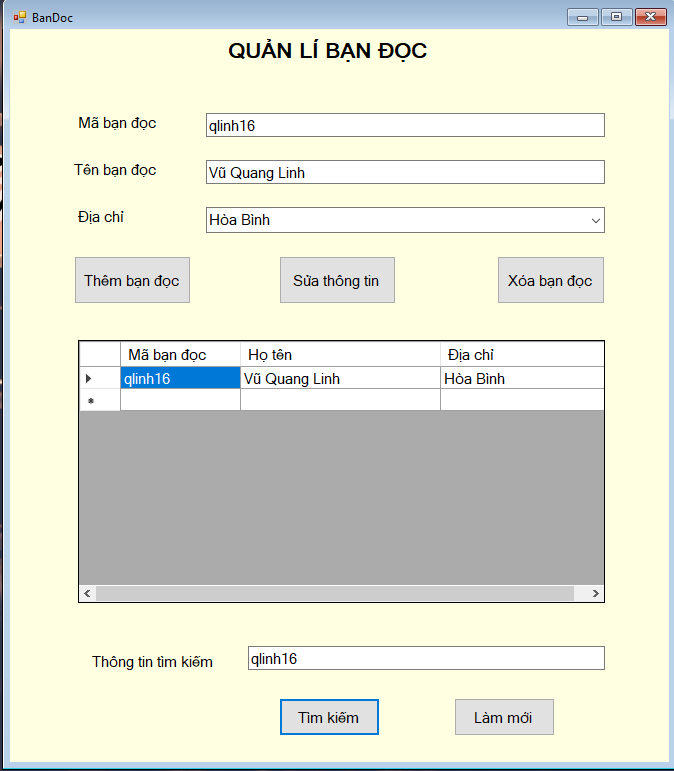
****

Hình 14: Thay đổi thông tin bạn đọc

* + Chỉ sửa được thông tin bạn đọc trong trường hợp bạn đọc đang trong trường hợp bạn đọc đó không cầm sách của thư viện



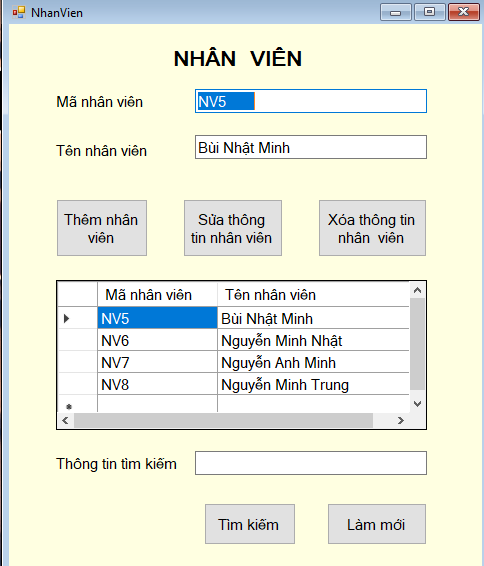
Hình 15: Xóa thông tin bạn đọc



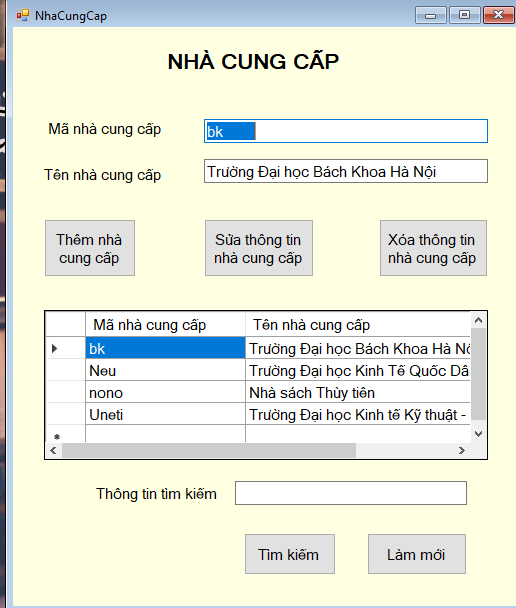
Hình 16: Tìm kiếm thông tin bạn đọc

* + Dưới đây là các form có các chức năng và nhiệm vụ như form Bạn đọc:

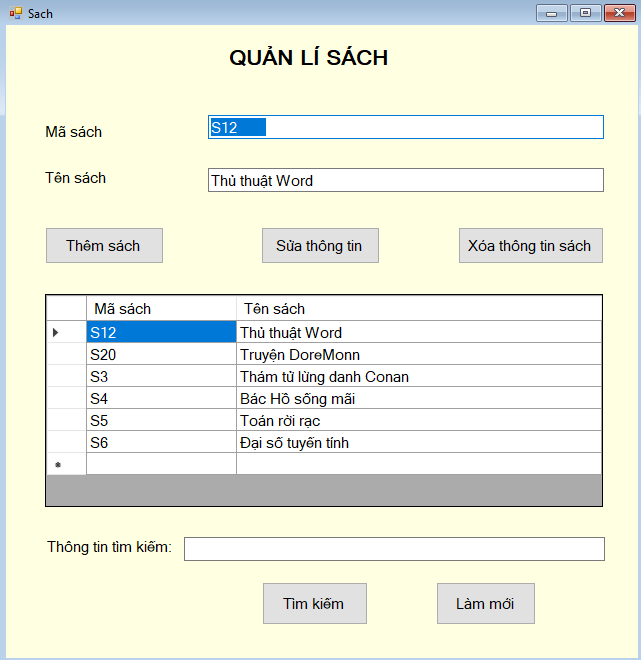
## Form quản lí nhân viên:



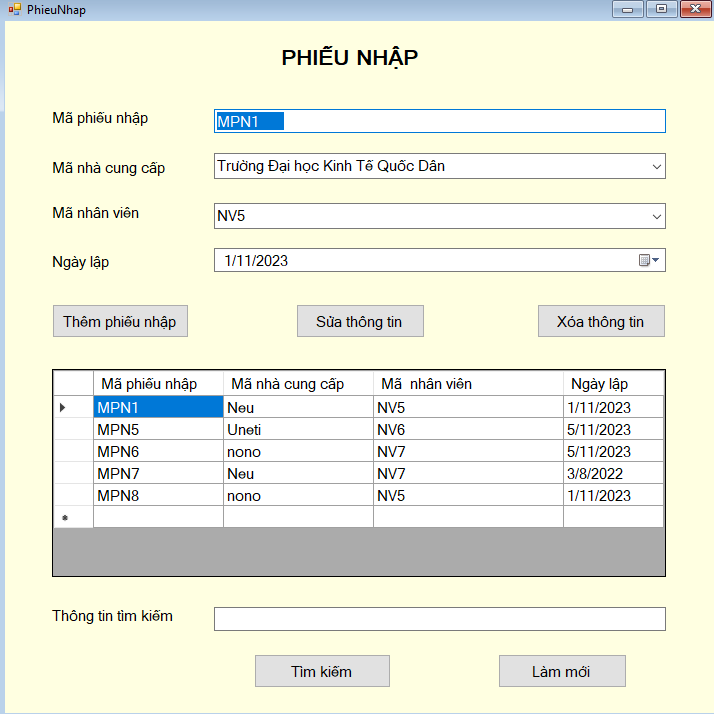
## **Form quản lí nhà cung cấp**:



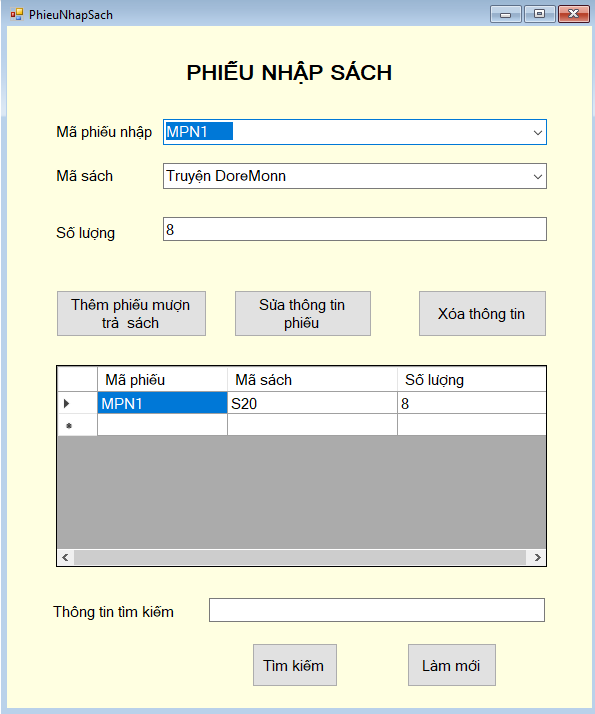
## Form quản lí sách:



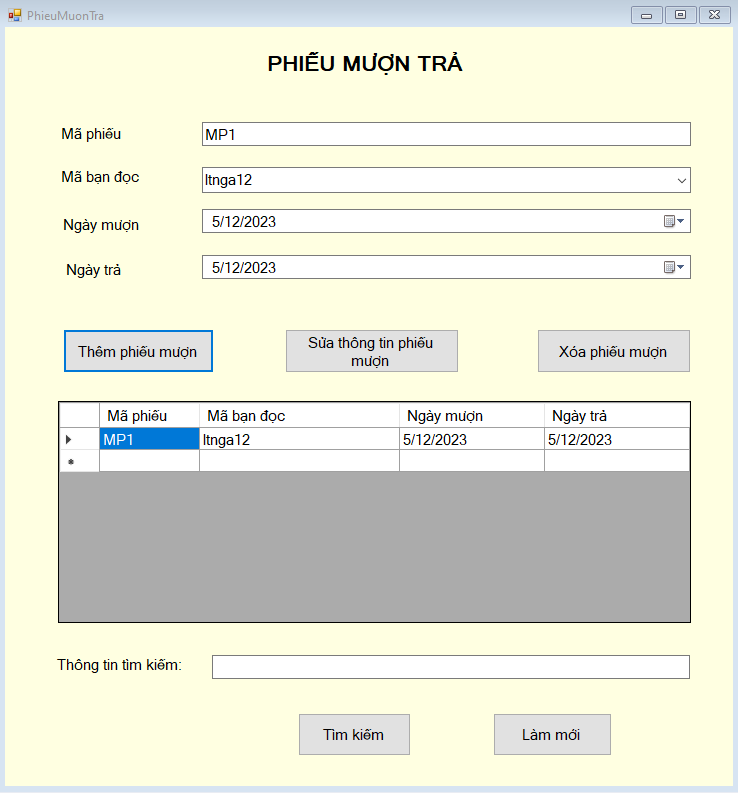
## Form quản lí phiếu nhập:



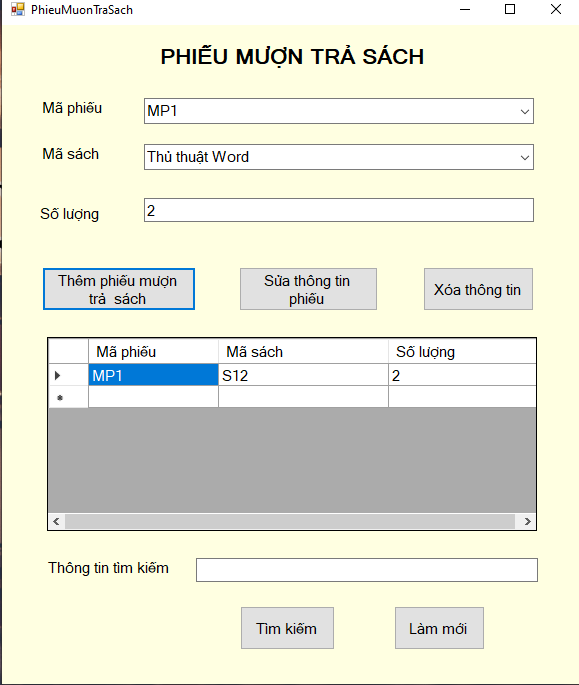
## Form quản lí phiếu nhập sách:



## Form quản lí phiếu mượn trả:



## Form quản lí phiếu mượn trả sách:



## Code chi tiết cho tất các form nêu trên:

### **Form đăng nhập:**

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

using System.Data.SqlClient;

namespace Menu

{

public partial class DangNhap : Form

{

SqlConnection con;

public DangNhap()

{

InitializeComponent();

con = new SqlConnection(@"Data Source=VUHOAINGOC\SQLEXPRESS;Initial Catalog=QuanLiThuVien;Integrated Security=True");

}

private void DangNhap\_Load(object sender, EventArgs e)

{

if(con.State == ConnectionState.Closed)

{

con.Open();

}

}

private void btnDangNhap\_Click(object sender, EventArgs e)

{

string username = txtUsername.Text;

string password = txtPassword.Text;

SqlCommand cmd = new SqlCommand("select phanquyen from tblDangNhap where username = @username and password = @password", con);

cmd.Parameters.AddWithValue("@username", username);

cmd.Parameters.AddWithValue("@password", password);

object result = cmd.ExecuteScalar();

if(result != null &&(bool)result)

{

MenuAdmin menuadmin = new MenuAdmin();

menuadmin.Tag = username;

menuadmin.Show();

this.Hide();

}

else

{

MessageBox.Show("Tài khoản hoặc mật khẩu không đúng! Vui lòng kiểm tra lại!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Error);

}

}

private void btnThoat\_Click(object sender, EventArgs e)

{

DialogResult exit = MessageBox.Show("Bạn có chắc chắn muốn thoát không?", "Thông báo", MessageBoxButtons.YesNoCancel, MessageBoxIcon.Question);

if (exit == DialogResult.Yes)

{

Application.Exit();

}

}

}

}

* **Form Menu:**

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

namespace Menu

{

public partial class MenuAdmin : Form

{

public MenuAdmin()

{

InitializeComponent();

}

private void nhânViênToolStripMenuItem\_Click(object sender, EventArgs e)

{

NhanVien nv = new NhanVien();

nv.MdiParent = this;

nv.Show();

}

private void nhàCungCấpToolStripMenuItem\_Click(object sender, EventArgs e)

{

NhaCungCap ncc = new NhaCungCap();

ncc.MdiParent = this;

ncc.Show();

}

private void sáchToolStripMenuItem\_Click(object sender, EventArgs e)

{

Sach s = new Sach();

s.MdiParent = this;

s.Show();

}

private void quảnLíPhiếuNhậpToolStripMenuItem\_Click(object sender, EventArgs e)

{

PhieuNhap pn = new PhieuNhap();

pn.MdiParent = this;

pn.Show();

}

private void quảnLíPhiếuNhậpSáchToolStripMenuItem1\_Click(object sender, EventArgs e)

{

PhieuNhapSach pns = new PhieuNhapSach();

pns.MdiParent = this;

pns.Show();

}

private void đăngXuấtToolStripMenuItem\_Click(object sender, EventArgs e)

{

DialogResult kq = MessageBox.Show("Bạn có chắc chắn muốn đăg xuất không!", "Thông báo", MessageBoxButtons.YesNoCancel, MessageBoxIcon.Question);

if (DialogResult.Yes == kq)

{

this.Hide();

DangNhap dangnhap = new DangNhap();

dangnhap.ShowDialog();

this.Close();

}

}

private void quảnLíPhiếuMượnTrảToolStripMenuItem\_Click(object sender, EventArgs e)

{

PhieuMuonTra pmt = new PhieuMuonTra();

pmt.MdiParent = this;

pmt.Show();

}

private void quảnLíPhiếuMượnTrảSáchToolStripMenuItem\_Click(object sender, EventArgs e)

{

PhieuMuonTraSach pmts = new PhieuMuonTraSach();

pmts.MdiParent = this;

pmts.Show();

}

private void bạnĐọcToolStripMenuItem\_Click(object sender, EventArgs e)

{

BanDoc bd = new BanDoc();

bd.MdiParent = this;

bd.Show();

}

}

}

### **Form quản lí bạn đọc:**

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

using System.Data.SqlClient;

namespace Menu

{

public partial class BanDoc : Form

{

SqlConnection con;

public BanDoc()

{

InitializeComponent();

con = new SqlConnection(@"Data Source=VUHOAINGOC\SQLEXPRESS;Initial Catalog=QuanLiThuVien;Integrated Security=True");

}

public void hienThiThongTin()

{

SqlCommand cmd = new SqlCommand("select \* from BanDoc", con);

SqlDataAdapter da = new SqlDataAdapter(cmd);

DataTable dt = new DataTable();

da.Fill(dt);

showInfromationdataGridView.DataSource = dt;

}

private void BanDoc\_Load(object sender, EventArgs e)

{

if (con.State == ConnectionState.Closed)

{

con.Open();

}

hienThiThongTin();

}

private void BanDoc\_FormClosed(object sender, FormClosedEventArgs e)

{

if (con.State == ConnectionState.Open)

{

con.Close();

}

}

private void btnThem\_Click(object sender, EventArgs e)

{

if(txtMaBanDoc.Text.Trim() == ""|| txtHoTen.Text.Trim() =="")

{

MessageBox.Show("Vui lòng nhập đầy đủ thông tin trước khi thêm!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Warning);

}

else

{

try

{

SqlCommand cmd = new SqlCommand("insert into BanDoc(MaBanDoc, TenBanDoc, DiaChi) values(@MaBanDoc, @TenBanDoc, @DiaChi)", con);

cmd.Parameters.AddWithValue("@MaBanDoc", txtMaBanDoc.Text);

cmd.Parameters.AddWithValue("@TenBanDoc", txtHoTen.Text);

cmd.Parameters.AddWithValue("@DiaChi", cmbDiaChi.SelectedItem);

if (cmd.ExecuteNonQuery() > 0)

{

MessageBox.Show("Thêm bạn đọc thành công!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Information);

}

else

MessageBox.Show("Thêm bạn đọc thất bại!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Information);

hienThiThongTin();

}

catch

{

MessageBox.Show("Mã bạn đọc đã tồn tại, vui lòng nhập mã sách khác!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Warning);

}

}

}

private void btnSua\_Click(object sender, EventArgs e)

{

int dongchon = showInfromationdataGridView.CurrentRow.Index;

if (txtMaBanDoc.Text.Trim() == "" || txtHoTen.Text.Trim() == "")

{

MessageBox.Show("Vui lòng nhập đầy đủ thông tin trước khi sửa!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Warning);

}

else

{

try

{

SqlCommand cmd = new SqlCommand("update BanDoc set MaBanDoc = @MaBanDoc, TenBanDoc = @TenBanDoc, DiaChi = @DiaChi where MaBanDoc = @MaBanDocCu", con);

cmd.Parameters.AddWithValue("@MaBanDoc", txtMaBanDoc.Text);

cmd.Parameters.AddWithValue("@TenBanDoc", txtHoTen.Text);

cmd.Parameters.AddWithValue("@DiaChi", cmbDiaChi.SelectedItem);

cmd.Parameters.AddWithValue("@MaBanDocCu", showInfromationdataGridView.Rows[dongchon].Cells["MaBanDoc"].Value.ToString());

if (cmd.ExecuteNonQuery() > 0)

{

MessageBox.Show("Sửa thông tin bạn đọc thành công!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Information);

}

else

MessageBox.Show("Sửa thông tin bạn đọc thất bại!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Information);

hienThiThongTin();

}

catch

{

MessageBox.Show("Bạn đọc có mã này đang mượn sách của thư viện nên không thể thay đổi thông tin!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Warning);

}

}

}

private void showInfromationdataGridView\_SelectionChanged(object sender, EventArgs e)

{

int dongchon = showInfromationdataGridView.CurrentRow.Index;

txtMaBanDoc.Text = showInfromationdataGridView.Rows[dongchon].Cells["MaBanDoc"].Value.ToString();

txtHoTen.Text = showInfromationdataGridView.Rows[dongchon].Cells["TenBanDoc"].Value.ToString();

cmbDiaChi.SelectedItem = showInfromationdataGridView.Rows[dongchon].Cells["DiaChi"].Value.ToString();

}

private void btnXoa\_Click(object sender, EventArgs e)

{

MessageBox.Show("Bạn có chắc chắn muốn xóa không? Nếu đồng ý bấm OK", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Question);

try

{

int dongchon = showInfromationdataGridView.CurrentRow.Index;

SqlCommand cmd = new SqlCommand("delete from BanDoc where MaBanDoc = @MaBanDoc", con);

cmd.Parameters.AddWithValue("@MaBanDoc", showInfromationdataGridView.Rows[dongchon].Cells["MaBanDoc"].Value.ToString());

if (cmd.ExecuteNonQuery() > 0)

{

MessageBox.Show("Xóa bạn đọc thành công!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Information);

}

else

MessageBox.Show("Xóa bạn đọc thất bại!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Information);

hienThiThongTin();

}

catch

{

MessageBox.Show("Bạn đọc có mã này đang trong thời gian mượn sách nên không thể xóa khỏi hệ thống!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Error);

}

}

private void btnTimKiem\_Click(object sender, EventArgs e)

{

SqlCommand cmd = new SqlCommand("select \* from BanDoc where MaBanDoc like N'%" + txtTimKiem.Text + "%'" +

"or TenBanDoc like N'%" + txtTimKiem.Text + "%'" +

"or DiaChi like N'%" + txtTimKiem.Text + "'", con);

SqlDataAdapter da = new SqlDataAdapter(cmd);

DataTable dt = new DataTable();

da.Fill(dt);

showInfromationdataGridView.DataSource = dt;

}

private void btnLamMoi\_Click(object sender, EventArgs e)

{

BanDoc\_Load(sender, e);

}

}

}

### **Form nhân viên:**

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

using System.Data.SqlClient;

using System.Diagnostics.Eventing.Reader;

namespace Menu

{

public partial class NhanVien : Form

{

SqlConnection con;

public NhanVien()

{

InitializeComponent();

con = new SqlConnection(@"Data Source=VUHOAINGOC\SQLEXPRESS;Initial Catalog=QuanLiThuVien;Integrated Security=True");

}

public void HienThiThongTin()

{

SqlCommand cmd = new SqlCommand("select \* from NhanVien", con);

SqlDataAdapter da = new SqlDataAdapter(cmd);

DataTable dt = new DataTable();

da.Fill(dt);

dgvThongTinNhanVien.DataSource = dt;

}

private void NhanVien\_Load(object sender, EventArgs e)

{

if (con.State == ConnectionState.Closed)

{

con.Open();

}

HienThiThongTin();

}

private void NhanVien\_FormClosed(object sender, FormClosedEventArgs e)

{

if (con.State == ConnectionState.Open)

{

con.Close();

}

}

private void btnThem\_Click(object sender, EventArgs e)

{

if (txtMaNhanVien.Text.Trim() == "" || txtTenNhanVien.Text.Trim() == "")

{

MessageBox.Show("Vui lòng nhập đầu đủ thông tin trước khi thêm!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Warning);

}

else

{

try

{

SqlCommand cmd = new SqlCommand("insert into NhanVien(MaNhanVien, TenNhanVien) values(@MaNhanVien, @TenNhanVien)", con);

cmd.Parameters.AddWithValue("@MaNhanVien", txtMaNhanVien.Text);

cmd.Parameters.AddWithValue("@TenNhanVien", txtTenNhanVien.Text);

if (cmd.ExecuteNonQuery() > 0)

{

MessageBox.Show("Thêm nhân viên thành công!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Information);

}

else

MessageBox.Show("Thêm nhân viên thất bại!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Information);

HienThiThongTin();

}

catch

{

MessageBox.Show("Mã nhân viên đã tồn tại, vui lòng nhập mã sách khác!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Warning);

}

}

}

private void btnSua\_Click(object sender, EventArgs e)

{

int dongchon = dgvThongTinNhanVien.CurrentRow.Index;

if (txtMaNhanVien.Text.Trim() == "" || txtTenNhanVien.Text.Trim() == "")

{

MessageBox.Show("Vui lòng nhập đầu đủ thông tin trước khi sửa!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Warning);

}

else

{

try

{

SqlCommand cmd = new SqlCommand("update NhanVien set MaNhanVien = @MaNhanVien, TenNhanVien = @TenNhanVien where MaNhanVien = @MaNhanViencu", con);

cmd.Parameters.AddWithValue("@MaNhanVien", txtMaNhanVien.Text);

cmd.Parameters.AddWithValue("@TenNhanVien", txtTenNhanVien.Text);

cmd.Parameters.AddWithValue("@MaNhanVienCu", dgvThongTinNhanVien.Rows[dongchon].Cells["MaNhanVien"].Value);

if (cmd.ExecuteNonQuery() > 0)

{

MessageBox.Show("Sửa thông tin nhân viên thành công!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Information);

}

else

MessageBox.Show("Sửa thông tin nhân viên thất bại!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Information);

HienThiThongTin();

}

catch

{

MessageBox.Show("Nhân viên có mã này đang quản lí nhập sách nên không thể thay đổi thông tin!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Warning);

}

}

}

private void dgvThongTinNhanVien\_SelectionChanged(object sender, EventArgs e)

{

int dongchon = dgvThongTinNhanVien.CurrentRow.Index;

txtMaNhanVien.Text = dgvThongTinNhanVien.Rows[dongchon].Cells["MaNhanVien"].Value.ToString();

txtTenNhanVien.Text = dgvThongTinNhanVien.Rows[dongchon].Cells["TenNhanVien"].Value.ToString();

}

private void btnXoa\_Click(object sender, EventArgs e)

{

MessageBox.Show("Bạn có chắc chắn muốn xóa không? Nếu đồng ý bấm OK", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Question);

try

{

int dongchon = dgvThongTinNhanVien.CurrentRow.Index;

SqlCommand cmd = new SqlCommand("delete from NhanVien where MaNhanVien = @MaNhanVien", con);

cmd.Parameters.AddWithValue("@MaNhanVien", dgvThongTinNhanVien.Rows[dongchon].Cells["MaNhanVien"].Value);

if (cmd.ExecuteNonQuery() > 0)

{

MessageBox.Show("Xóa thông tin nhân viên thành công!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Information);

}

else

MessageBox.Show("Xóa thông tin nhân viên thất bại!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Information);

HienThiThongTin();

}

catch

{

MessageBox.Show("Nhân viên này đang trong quá trình quản lí việc nhập sách nên không thể xóa khỏi hệ thống!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Error);

}

}

private void btnTimKiem\_Click(object sender, EventArgs e)

{

SqlCommand cmd = new SqlCommand("select \* from NhanVien where MaNhanVien like N'%" + txtTimKiem.Text + "%'" +

"or TenNhanVien like N'%" + txtTimKiem.Text + "%'", con);

SqlDataAdapter da = new SqlDataAdapter(cmd);

DataTable dt = new DataTable();

da.Fill(dt);

dgvThongTinNhanVien.DataSource = dt;

}

private void btnLamMoi\_Click(object sender, EventArgs e)

{

NhanVien\_Load(sender, e);

}

}

}

### **Form nhà cung cấp:**

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

using System.Data.SqlClient;

namespace Menu

{

public partial class NhaCungCap : Form

{

SqlConnection con;

public NhaCungCap()

{

InitializeComponent();

con = new SqlConnection(@"Data Source=VUHOAINGOC\SQLEXPRESS;Initial Catalog=QuanLiThuVien;Integrated Security=True");

}

public void HienThiThongTin()

{

SqlCommand cmd = new SqlCommand("select \* from NhaCungCap", con);

SqlDataAdapter da = new SqlDataAdapter(cmd);

DataTable dt = new DataTable();

da.Fill(dt);

dgvThongTinNhaCungCap.DataSource = dt;

}

private void NhaCungCap\_Load(object sender, EventArgs e)

{

if (con.State == ConnectionState.Closed)

{

con.Open();

}

HienThiThongTin();

}

private void NhaCungCap\_FormClosed(object sender, FormClosedEventArgs e)

{

if (con.State == ConnectionState.Open)

{

con.Close();

}

}

private void btnThem\_Click(object sender, EventArgs e)

{

if(txtMaNhaCungCap.Text.Trim() == ""|| txtTenNhaCungCap.Text.Trim()=="")

{

MessageBox.Show("Vui lòng nhập đầy đủ thông tin trước khi thêm!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Warning);

}

else

{

try

{

SqlCommand cmd = new SqlCommand("insert into NhaCungCap(MaNhaCungCap, TenNhaCungCap) values( @MaNhaCungCap, @TenNhaCungCap)", con);

cmd.Parameters.AddWithValue("@MaNhaCungCap", txtMaNhaCungCap.Text);

cmd.Parameters.AddWithValue("@TenNhaCungCap", txtTenNhaCungCap.Text);

if (cmd.ExecuteNonQuery() > 0)

{

MessageBox.Show("Thêm nhân viên thành công!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Information);

}

else

MessageBox.Show("Thêm nhân viên thất bại!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Information);

HienThiThongTin();

}

catch

{

MessageBox.Show("Mã nhà cung cấp đã tồn tại, vui lòng nhập mã sách khác!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Warning);

}

}

}

private void btnSua\_Click(object sender, EventArgs e)

{

int dongchon = dgvThongTinNhaCungCap.CurrentRow.Index;

if (txtMaNhaCungCap.Text.Trim() == "" || txtTenNhaCungCap.Text.Trim() == "")

{

MessageBox.Show("Vui lòng nhập đầy đủ thông tin trước khi sửa!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Warning);

}

else

{

try

{

SqlCommand cmd = new SqlCommand("update NhaCungCap set MaNhaCungCap = @MaNhaCungCap, TenNhaCungCap = @TenNhaCungCap where MaNhaCungCap = @MaNhaCungCapCu", con);

cmd.Parameters.AddWithValue("@MaNhaCungCap", txtMaNhaCungCap.Text);

cmd.Parameters.AddWithValue("@TenNhaCungCap", txtTenNhaCungCap.Text);

cmd.Parameters.AddWithValue("@MaNhaCungCapCu", dgvThongTinNhaCungCap.Rows[dongchon].Cells["MaNhaCungCap"].Value);

if (cmd.ExecuteNonQuery() > 0)

{

MessageBox.Show("Sửa thông tin nhà cung cấp thành công!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Information);

}

else

MessageBox.Show("Sửa thông tin nhà cung cấp thất bại!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Information);

HienThiThongTin();

}

catch

{

MessageBox.Show("Nhà cung cấp có mã này chưa hoàn thành việc giao sách cho trung tâm như đơn đã đặt nên chưa thể xóa ra khỏi hệ thống!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Error);

}

}

}

private void dgvThongTinNhaCungCap\_SelectionChanged(object sender, EventArgs e)

{

int dongchon = dgvThongTinNhaCungCap.CurrentRow.Index;

txtMaNhaCungCap.Text = dgvThongTinNhaCungCap.Rows[dongchon].Cells["MaNhaCungCap"].Value.ToString();

txtTenNhaCungCap.Text = dgvThongTinNhaCungCap.Rows[dongchon].Cells["tenNhaCungCap"].Value.ToString();

}

private void btnXoa\_Click(object sender, EventArgs e)

{

MessageBox.Show("Bạn có chắc chắn muốn xóa không? Nếu đồng ý bấm OK", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Question);

try

{

int dongchon = dgvThongTinNhaCungCap.CurrentRow.Index;

SqlCommand cmd = new SqlCommand("delete from NhaCungCap where MaNhaCungCap = @MaNhaCungCap", con);

cmd.Parameters.AddWithValue("@MaNhaCungCap", dgvThongTinNhaCungCap.Rows[dongchon].Cells["MaNhaCungCap"].Value);

if (cmd.ExecuteNonQuery() > 0)

{

MessageBox.Show("Xóa thông tin nhà cung cấp thành công!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Information);

}

else

MessageBox.Show("Xóa thông tin nhà cung cấp thất bại!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Information);

HienThiThongTin();

}

catch

{

MessageBox.Show("Nhà cung cấp có mã này chưa hoàn thành việc giao sách cho trung tâm như đơn đã đặt nên chưa thể xóa ra khỏi hệ thống!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Error);

}

}

private void btnTimKiem\_Click(object sender, EventArgs e)

{

SqlCommand cmd = new SqlCommand("select \* from NhaCungCap where MaNhaCungCap like N'%" + txtTimKiem.Text + "%'" +

"or TenNhaCungCap like N'%" + txtTimKiem.Text + "%'", con);

SqlDataAdapter da = new SqlDataAdapter(cmd);

DataTable dt = new DataTable();

da.Fill(dt);

dgvThongTinNhaCungCap.DataSource = dt;

}

private void btnLamMoi\_Click(object sender, EventArgs e)

{

NhaCungCap\_Load(sender, e);

}

}

}

### **Form sách:**

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

using System.Data.SqlClient;

namespace Menu

{

public partial class Sach : Form

{

SqlConnection con;

public Sach()

{

InitializeComponent();

con = new SqlConnection(@"Data Source=VUHOAINGOC\SQLEXPRESS;Initial Catalog=QuanLiThuVien;Integrated Security=True");

}

public void HienThiThongTinSach()

{

SqlCommand cmd = new SqlCommand("select \* from Sach", con);

SqlDataAdapter da = new SqlDataAdapter(cmd);

DataTable dt = new DataTable();

da.Fill(dt);

showThongTinSach.DataSource = dt;

}

private void Sach\_Load(object sender, EventArgs e)

{

if (con.State == ConnectionState.Closed)

{

con.Open();

}

HienThiThongTinSach();

}

private void Sach\_FormClosed(object sender, FormClosedEventArgs e)

{

if (con.State == ConnectionState.Open)

{

con.Close();

}

}

private void btnThem\_Click\_1(object sender, EventArgs e)

{

if(txtMaSach.Text.Trim() == "" || txtTenSach.Text.Trim()=="")

{

MessageBox.Show("Vui lòng nhập đầu đủ thông tin trước khi thêm!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Warning);

}

else

{

try

{

SqlCommand cmd = new SqlCommand("insert into Sach(MaSach, TenSach) values(@MaSach, @TenSach)", con);

cmd.Parameters.AddWithValue("@MaSach", txtMaSach.Text);

cmd.Parameters.AddWithValue("@TenSach", txtTenSach.Text);

if (cmd.ExecuteNonQuery() > 0)

{

MessageBox.Show("Thêm sách thành công!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Information);

}

else

MessageBox.Show("Thêm sách thất bại!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Information);

HienThiThongTinSach();

}

catch

{

MessageBox.Show("Mã sách đã tồn tại, vui lòng nhập mã sách khác!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Warning);

}

}

}

private void showThongTinSach\_SelectionChanged(object sender, EventArgs e)

{

int dongchon = showThongTinSach.CurrentRow.Index;

txtMaSach.Text = showThongTinSach.Rows[dongchon].Cells["MaSach"].Value.ToString();

txtTenSach.Text = showThongTinSach.Rows[dongchon].Cells["TenSach"].Value.ToString();

}

private void btnSua\_Click(object sender, EventArgs e)

{

int dongchon = showThongTinSach.CurrentRow.Index;

if (txtMaSach.Text.Trim() == "" || txtTenSach.Text.Trim() == "")

{

MessageBox.Show("Vui lòng nhập đầu đủ thông tin trước khi sửa!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Warning);

}

else

{

try

{

SqlCommand cmd = new SqlCommand("update Sach set MaSach = @MaSach, TenSach = @TenSach where MaSach = @MaSachCu", con);

cmd.Parameters.AddWithValue("@MaSach", txtMaSach.Text);

cmd.Parameters.AddWithValue("@TenSach", txtTenSach.Text);

cmd.Parameters.AddWithValue("@MaSachCu", showThongTinSach.Rows[dongchon].Cells["MaSach"].Value.ToString());

if (cmd.ExecuteNonQuery() > 0)

{

MessageBox.Show("Sửa thông tin sách thành công!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Information);

}

else

MessageBox.Show("Sửa thông tin sách thất bại!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Information);

HienThiThongTinSach();

}

catch

{

MessageBox.Show("Sách có mã này đang được cho mượn hoặc đang được nhập về nên không thể thay đổi thông tin được!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Warning);

}

}

}

private void btnXoa\_Click(object sender, EventArgs e)

{

try

{

int dongchon = showThongTinSach.CurrentRow.Index;

SqlCommand cmd = new SqlCommand("delete from Sach where MaSach = @MaSach", con);

cmd.Parameters.AddWithValue("@MaSach", showThongTinSach.Rows[dongchon].Cells["MaSach"].Value.ToString());

if (cmd.ExecuteNonQuery() > 0)

{

MessageBox.Show("Xóa thành công!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Information);

}

else

MessageBox.Show("Xóa thất bại!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Information);

HienThiThongTinSach();

}

catch

{

MessageBox.Show("Sách có mã này đang được cho mượn hoặc đang được nhập về nên không thể thay đổi thông tin được!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Warning);

}

}

private void btnTimKiem\_Click(object sender, EventArgs e)

{

SqlCommand cmd = new SqlCommand("select \* from Sach where MaSach like N'%" + txtTimKiem.Text + "%'" +

" or TenSach like N'%" + txtTimKiem.Text + "%'", con);

SqlDataAdapter da = new SqlDataAdapter(cmd);

DataTable dt = new DataTable();

da.Fill(dt);

showThongTinSach.DataSource = dt;

}

private void btnLamMoi\_Click(object sender, EventArgs e)

{

Sach\_Load(sender, e);

}

}

}

### **Form phiếu mượn trả:**

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

using System.Data.SqlClient;

namespace Menu

{

public partial class PhieuMuonTra : Form

{

SqlConnection con;

public PhieuMuonTra()

{

InitializeComponent();

con = new SqlConnection(@"Data Source=VUHOAINGOC\SQLEXPRESS;Initial Catalog=QuanLiThuVien;Integrated Security=True");

}

public void hienThiThngTin()

{

SqlCommand cmd = new SqlCommand("select \* from PhieuMuonTra", con);

SqlDataAdapter da = new SqlDataAdapter(cmd);

DataTable dt = new DataTable();

da.Fill(dt);

dgvThongTinPhieuMuon.DataSource = dt;

}

private void PhieuMuonTra\_Load(object sender, EventArgs e)

{

if (con.State == ConnectionState.Closed)

{

con.Open();

}

hienThiThngTin();

SqlCommand cmd = new SqlCommand("select \* from BanDoc", con);

SqlDataAdapter da = new SqlDataAdapter(cmd);

DataTable dt = new DataTable();

da.Fill(dt);

cmbMaBanDoc.DataSource = dt;

cmbMaBanDoc.DisplayMember = "MaBanDoc";

cmbMaBanDoc.ValueMember = "MaBanDoc";

}

private void PhieuMuonTra\_FormClosed(object sender, FormClosedEventArgs e)

{

if (con.State == ConnectionState.Open)

{

con.Close();

}

}

private void btnThem\_Click(object sender, EventArgs e)

{

if(txtMaPhieu.Text.Trim() == "" || cmbMaBanDoc.SelectedIndex == -1)

{

MessageBox.Show("Vui lòng điền đầy đủ thông tin trước khi thêm!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Warning);

}

else

{

try

{

SqlCommand cmd = new SqlCommand("insert into PhieuMuonTra(MaPhieu, MaBanDoc, NgayMuon, NgayTra)" +

"values(@MaPhieu, @MaBanDoc, @NgayMuon, @NgayTra)", con);

cmd.Parameters.AddWithValue("@MaPhieu", txtMaPhieu.Text);

cmd.Parameters.AddWithValue("@MaBanDoc", cmbMaBanDoc.SelectedValue);

cmd.Parameters.AddWithValue("@NgayMuon", dtpNgayMuon.Value);

cmd.Parameters.AddWithValue("@NgayTra", dtpNgayTra.Value);

if (cmd.ExecuteNonQuery() > 0)

{

MessageBox.Show("Đã hoàn thành phiếu mượn sách!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Information);

MessageBox.Show("Chuyển qua phiếu mượn trả sách để hoàn thành phiếu mượn trả!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Information);

PhieuMuonTraSach pmts = new PhieuMuonTraSach();

pmts.Show();

}

else

{

MessageBox.Show("Việc mượn sách đã xảy ra gián đoạn! Vui lòng kiểm tra lại!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Information);

}

hienThiThngTin();

}

catch

{

MessageBox.Show("Mã phiếu đã tồn tại! Vui lòng nhập mã khác!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Warning);

}

}

}

private void dgvThongTinPhieuMuon\_SelectionChanged(object sender, EventArgs e)

{

int dongchon = dgvThongTinPhieuMuon.CurrentRow.Index;

txtMaPhieu.Text = dgvThongTinPhieuMuon.Rows[dongchon].Cells["MaPhieu"].Value.ToString();

cmbMaBanDoc.SelectedValue = dgvThongTinPhieuMuon.Rows[dongchon].Cells["MaBanDoc"].Value.ToString();

dtpNgayMuon.Value = DateTime.Parse(dgvThongTinPhieuMuon.Rows[dongchon].Cells["NgayMuon"].Value.ToString());

dtpNgayTra.Value = DateTime.Parse(dgvThongTinPhieuMuon.Rows[dongchon].Cells["NgayTra"].Value.ToString());

}

private void btnSua\_Click(object sender, EventArgs e)

{

int dongchon = dgvThongTinPhieuMuon.CurrentRow.Index;

if (txtMaPhieu.Text.Trim() == "" || cmbMaBanDoc.SelectedIndex == -1)

{

MessageBox.Show("Vui lòng điền đầy đủ thông tin trước khi sửa!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Warning);

}

else

{

try

{

SqlCommand cmd = new SqlCommand("update PhieuMuonTra set MaPhieu = @MaPhieu, MaBanDoc = @MaBanDoc," +

" NgayMuon = @NgayMuon, NgayTra = @NgayTra where MaPhieu = @MaPhieuCu", con);

cmd.Parameters.AddWithValue("@MaPhieu", txtMaPhieu.Text);

cmd.Parameters.AddWithValue("@MaBanDoc", cmbMaBanDoc.SelectedValue);

cmd.Parameters.AddWithValue("@NgayMuon", dtpNgayMuon.Value);

cmd.Parameters.AddWithValue("@NgayTra", dtpNgayTra.Value);

cmd.Parameters.AddWithValue("@MaPhieuCu", dgvThongTinPhieuMuon.Rows[dongchon].Cells["MaPhieu"].Value.ToString());

if (cmd.ExecuteNonQuery() > 0)

{

MessageBox.Show("Sửa thông tin phiếu thành công!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Information);

}

else

MessageBox.Show("Sửa thông tin phiếu thất bại!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Information);

hienThiThngTin();

}

catch

{

MessageBox.Show("Bạn đọc này đang cầm sách có mã phiếu này của thư viện nên không thể thay đổi thông tin được!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Information);

}

}

}

private void btnXoa\_Click(object sender, EventArgs e)

{

MessageBox.Show("Bạn có chắc chắn muốn xóa không? Nếu đồng ý bấm OK", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Question);

try

{

int dongchon = dgvThongTinPhieuMuon.CurrentRow.Index;

SqlCommand cmd = new SqlCommand("delete from PhieuMuonTra where MaPhieu = @MaPhieu", con);

cmd.Parameters.AddWithValue("@MaPhieu", dgvThongTinPhieuMuon.Rows[dongchon].Cells["MaPhieu"].Value.ToString());

if (cmd.ExecuteNonQuery() > 0)

{

MessageBox.Show("Xóa thông tin phiếu thành công!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Information);

}

else

MessageBox.Show("Xóa thông tin phiếu thất bại!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Information);

hienThiThngTin();

}

catch

{

MessageBox.Show("Bạn đọc có mã phiếu mượn này chưa hoàn thành việc trả sách nên không thể xóa khỏi hệ thống!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Error);

}

}

private void btnTimKiem\_Click(object sender, EventArgs e)

{

SqlCommand cmd = new SqlCommand("select \* from PhieuMuonTra where MaPhieu like N'%" + txtTimKiem.Text + "%'" +

"or MaBanDoc like N'%" + txtTimKiem.Text + "%'", con);

SqlDataAdapter da = new SqlDataAdapter(cmd);

DataTable dt = new DataTable();

da.Fill(dt);

dgvThongTinPhieuMuon.DataSource = dt;

}

private void btnLamMoi\_Click(object sender, EventArgs e)

{

PhieuMuonTra\_Load(sender, e);

}

}

}

### **Form phiếu mượn trả sách:**

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

using System.Data.SqlClient;

namespace Menu

{

public partial class PhieuMuonTraSach : Form

{

SqlConnection con;

public PhieuMuonTraSach()

{

InitializeComponent();

con = new SqlConnection(@"Data Source=VUHOAINGOC\SQLEXPRESS;Initial Catalog=QuanLiThuVien;Integrated Security=True");

}

public void HienThiThongTin()

{

SqlCommand cmd = new SqlCommand("select \* from PhieuMuonTraSach", con);

SqlDataAdapter da = new SqlDataAdapter(cmd);

DataTable dt = new DataTable();

da.Fill(dt);

dgvShowThongTin.DataSource = dt;

}

private void PhieuMuonTraSach\_Load(object sender, EventArgs e)

{

if (con.State == ConnectionState.Closed)

{

con.Open();

}

HienThiThongTin();

SqlCommand cmd = new SqlCommand("select \* from PhieuMuonTra", con);

SqlDataAdapter da = new SqlDataAdapter(cmd);

DataTable dt = new DataTable();

da.Fill(dt);

cmbMaPhieu.DataSource = dt;

cmbMaPhieu.DisplayMember = "MaPhieu";

cmbMaPhieu.ValueMember = "MaPhieu";

SqlCommand cmd1 = new SqlCommand("select \* from Sach", con);

SqlDataAdapter da1 = new SqlDataAdapter(cmd1);

DataTable dt1 = new DataTable();

da1.Fill(dt1);

cmbMaSach.DataSource = dt1;

cmbMaSach.DisplayMember = "TenSach";

cmbMaSach.ValueMember = "MaSach";

}

private void PhieuMuonTraSach\_FormClosed(object sender, FormClosedEventArgs e)

{

if (con.State == ConnectionState.Open)

{

con.Close();

}

}

private void btnThem\_Click(object sender, EventArgs e)

{

if(cmbMaPhieu.SelectedIndex == -1 || cmbMaSach.SelectedIndex == -1 || txtSoLuong.Text.Trim() == "")

{

MessageBox.Show("Vui lòng nhập đầy đủ thông tin trước khi thêm!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Warning);

}

else

{

try

{

SqlCommand cmd = new SqlCommand("insert into PhieuMuonTraSach(MaPhieu, MaSach, SoLuong) values(@MaPhieu, @MaSach, @SoLuong) ", con);

cmd.Parameters.AddWithValue("@MaPhieu", cmbMaPhieu.SelectedValue);

cmd.Parameters.AddWithValue("@MaSach", cmbMaSach.SelectedValue);

cmd.Parameters.AddWithValue("@SoLuong", txtSoLuong.Text);

if (cmd.ExecuteNonQuery() > 0)

{

MessageBox.Show("Thêm phiếu mượn trả sách thành công!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Information);

}

else

MessageBox.Show("Thêm phiếu mượn trả sách thất bại!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Information);

HienThiThongTin();

}

catch

{

MessageBox.Show("Bạn đọc đã mượn cuốn sách này, vui lòng mượn cuốn khác!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Warning);

}

}

}

private void btnSua\_Click(object sender, EventArgs e)

{

int dongchon = dgvShowThongTin.CurrentRow.Index;

if (cmbMaPhieu.SelectedIndex == -1 || cmbMaSach.SelectedIndex == -1 || txtSoLuong.Text.Trim() == "")

{

MessageBox.Show("Vui lòng nhập đầy đủ thông tin trước khi thêm!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Warning);

}

else

{

try

{

SqlCommand cmd = new SqlCommand("update PhieuMuonTraSach set MaPhieu = @MaPhieu," +

" MaSach = @MaSach, SoLuong = @SoLuong where MaPhieu = @MaPhieuCu", con);

cmd.Parameters.AddWithValue("@MaPhieu", cmbMaPhieu.SelectedValue);

cmd.Parameters.AddWithValue("@MaSach", cmbMaSach.SelectedValue);

cmd.Parameters.AddWithValue("@SoLuong", txtSoLuong.Text);

cmd.Parameters.AddWithValue("@MaPhieuCu", dgvShowThongTin.Rows[dongchon].Cells["MaPhieu"].Value.ToString());

if (cmd.ExecuteNonQuery() > 0)

{

MessageBox.Show("Sửa thông tin phiếu mượn trả sách thành công!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Information);

}

else

MessageBox.Show("Sửa thông tin phiếu mượn trả sách thất bại!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Information);

HienThiThongTin();

}

catch

{

MessageBox.Show("Tồn tại nhiều mã phiếu giống nhau nên không sửa thông tin của một trong các mã phiếu được!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Warning);

}

}

}

private void dgvShowThongTin\_SelectionChanged(object sender, EventArgs e)

{

int dongchon = dgvShowThongTin.CurrentRow.Index;

cmbMaPhieu.SelectedValue = dgvShowThongTin.Rows[dongchon].Cells["MaPhieu"].Value;

cmbMaSach.SelectedValue = dgvShowThongTin.Rows[dongchon].Cells["MaSach"].Value;

txtSoLuong.Text = dgvShowThongTin.Rows[dongchon].Cells["SoLuong"].Value.ToString();

}

private void btnXoa\_Click(object sender, EventArgs e)

{

int dongchon = dgvShowThongTin.CurrentRow.Index;

SqlCommand cmd = new SqlCommand("delete from PhieuMuonTraSach where MaPhieu = @MaPhieu ", con);

cmd.Parameters.AddWithValue("@MaPhieu", dgvShowThongTin.Rows[dongchon].Cells["MaPhieu"].Value.ToString());

if (cmd.ExecuteNonQuery() > 0)

{

MessageBox.Show("Xóa phiếu mượn trả sách thành công!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Information);

}

else

MessageBox.Show("Xóa phiếu mượn trả sách thất bại!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Information);

HienThiThongTin();

}

private void btnTimKiem\_Click(object sender, EventArgs e)

{

SqlCommand cmd = new SqlCommand("select \* from PhieuMuonTraSach where MaPhieu like'%" + txtTimKiem.Text + "%'" +

"or MaSach like'%" + txtTimKiem.Text + "%'", con);

SqlDataAdapter da = new SqlDataAdapter(cmd);

DataTable dt = new DataTable();

da.Fill(dt);

dgvShowThongTin.DataSource = dt;

}

private void btnLamMoi\_Click(object sender, EventArgs e)

{

PhieuMuonTraSach\_Load(sender, e);

}

}

}

### **Form phiếu nhập:**

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

using System.Data.SqlClient;

namespace Menu

{

public partial class PhieuNhap : Form

{

SqlConnection con;

public PhieuNhap()

{

InitializeComponent();

con = new SqlConnection(@"Data Source=VUHOAINGOC\SQLEXPRESS;Initial Catalog=QuanLiThuVien;Integrated Security=True");

}

public void HienThiThongTin()

{

SqlCommand cmd = new SqlCommand("select \* from PhieuNhap", con);

SqlDataAdapter da = new SqlDataAdapter(cmd);

DataTable dt = new DataTable();

da.Fill(dt);

dgvHienThiThongTinPhieuNhap.DataSource = dt;

}

private void PhieuNhap\_Load(object sender, EventArgs e)

{

if (con.State == ConnectionState.Closed)

{

con.Open();

}

HienThiThongTin();

SqlCommand cmd = new SqlCommand("select \* from NhaCungCap", con);

SqlDataAdapter da = new SqlDataAdapter(cmd);

DataTable dt = new DataTable();

da.Fill(dt);

cmbMaNhaCungCap.DataSource = dt;

cmbMaNhaCungCap.DisplayMember = "TenNhaCungCap";

cmbMaNhaCungCap.ValueMember = "MaNhaCungCap";

SqlCommand cmd1 = new SqlCommand("select \* from NhanVien", con);

SqlDataAdapter da1 = new SqlDataAdapter(cmd1);

DataTable dt1 = new DataTable();

da1.Fill(dt1);

cmbHoTenNhanVien.DataSource = dt1;

cmbHoTenNhanVien.DisplayMember = "MaNhanVien";

cmbHoTenNhanVien.ValueMember = "MaNhanVien";

}

private void PhieuNhap\_FormClosed(object sender, FormClosedEventArgs e)

{

if (con.State == ConnectionState.Open)

{

con.Close();

}

}

private void btnThem\_Click(object sender, EventArgs e)

{

if (txtMaPhieuNhap.Text.Trim() == ""|| cmbMaNhaCungCap.SelectedIndex == -1 || cmbHoTenNhanVien.SelectedIndex == -1 || dtpNgayLap.Text == "")

{

MessageBox.Show("Vui lòng nhập đầy đủ thông tin trước khi thêm!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Warning);

}

else

{

try

{

SqlCommand cmd = new SqlCommand("insert into PhieuNhap(MaPhieuNhap, MaNhaCungCap, MaNhanVien, NgayLap) values(@MaPhieuNhap, @MaNhaCungCap, @MaNhanVien, @NgayLap) ", con);

cmd.Parameters.AddWithValue("@MaPhieuNhap", txtMaPhieuNhap.Text);

cmd.Parameters.AddWithValue("@MaNhaCungCap", cmbMaNhaCungCap.SelectedValue);

cmd.Parameters.AddWithValue("@MaNhanVien", cmbHoTenNhanVien.SelectedValue);

cmd.Parameters.AddWithValue("@NgayLap", dtpNgayLap.Value);

if (cmd.ExecuteNonQuery() > 0)

{

MessageBox.Show("Thêm phiếu nhập sách thành công!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Information);

MessageBox.Show("Hãy chuyển qua phiếu nhập sách để hoàn thành việc nhập sách!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Information);

PhieuNhapSach pns = new PhieuNhapSach();

pns.Show();

}

else

MessageBox.Show("Thêm phiếu nhập sách thất bại!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Information);

HienThiThongTin();

}

catch

{

MessageBox.Show("Mã phiếu nhập đã tồn tại! Vui lòng nhập mã khác!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Warning);

}

}

}

private void dgvHienThiThongTinPhieuNhap\_SelectionChanged(object sender, EventArgs e)

{

int dongchon = dgvHienThiThongTinPhieuNhap.CurrentRow.Index;

txtMaPhieuNhap.Text = dgvHienThiThongTinPhieuNhap.Rows[dongchon].Cells["MaPhieuNhap"].Value.ToString();

cmbMaNhaCungCap.SelectedValue = dgvHienThiThongTinPhieuNhap.Rows[dongchon].Cells["MaNhaCungCap"].Value;

cmbHoTenNhanVien.SelectedValue = dgvHienThiThongTinPhieuNhap.Rows[dongchon].Cells["MaNhanVien"].Value;

dtpNgayLap.Value = DateTime.Parse(dgvHienThiThongTinPhieuNhap.Rows[dongchon].Cells["NgayLap"].Value.ToString());

}

private void btnSua\_Click(object sender, EventArgs e)

{

if (txtMaPhieuNhap.Text.Trim() == "" || cmbMaNhaCungCap.SelectedIndex == -1 || cmbHoTenNhanVien.SelectedIndex == -1 || dtpNgayLap.Text == "")

{

MessageBox.Show("Vui lòng nhập đầy đủ thông tin trước khi sửa!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Warning);

}

else

{

try

{

int dongchon = dgvHienThiThongTinPhieuNhap.CurrentRow.Index;

SqlCommand cmd = new SqlCommand("update PhieuNhap set MaPhieuNhap = @MaPhieuNhap, MaNhaCungCap = @MaNhaCungCap, MaNhanVien = @MaNhanVien, NgayLap = @NgayLap where MaPhieuNhap = @MaPhieuNhapCu", con);

cmd.Parameters.AddWithValue("@MaPhieuNhap", txtMaPhieuNhap.Text);

cmd.Parameters.AddWithValue("@MaNhaCungCap", cmbMaNhaCungCap.SelectedValue);

cmd.Parameters.AddWithValue("@MaNhanVien", cmbHoTenNhanVien.SelectedValue);

cmd.Parameters.AddWithValue("@NgayLap", dtpNgayLap.Value);

cmd.Parameters.AddWithValue("@MaPhieuNhapCu", dgvHienThiThongTinPhieuNhap.Rows[dongchon].Cells["MaPhieuNhap"].Value.ToString());

if (cmd.ExecuteNonQuery() > 0)

{

MessageBox.Show("Sửa thông tin phiếu nhập sách thành công!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Information);

}

else

MessageBox.Show("Sửa thông tin phiếu nhập sách thất bại!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Information);

HienThiThongTin();

}

catch

{

MessageBox.Show("Mã phiếu nhập đã tồn tại! Vui lòng nhập mã khác!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Warning);

}

}

}

private void btnXoa\_Click(object sender, EventArgs e)

{

MessageBox.Show("Bạn có chắc chắn muốn xóa không? Nếu đồng ý bấm OK", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Question);

try

{

int dongchon = dgvHienThiThongTinPhieuNhap.CurrentRow.Index;

SqlCommand cmd = new SqlCommand("delete from PhieuNhap where MaPhieuNhap = @MaPhieuNhap", con);

cmd.Parameters.AddWithValue("@MaPhieuNhap", dgvHienThiThongTinPhieuNhap.Rows[dongchon].Cells["MaPhieuNhap"].Value.ToString());

if (cmd.ExecuteNonQuery() > 0)

{

MessageBox.Show("Xóa thông tin phiếu nhập sách thành công!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Information);

}

else

MessageBox.Show("Xóa thông tin phiếu nhập sách thất bại!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Information);

HienThiThongTin();

}

catch

{

MessageBox.Show("Việc nhập sách chưa hoàn thành nên chưa thể xóa phiếu nhập này ra khỏi hệ thống!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Error);

}

}

private void btnTimKiem\_Click(object sender, EventArgs e)

{

SqlCommand cmd = new SqlCommand("select \* from PhieuNhap where MaPhieuNhap like'%" + txtTimKiem.Text + "%'" +

"or MaNhaCungCap like'%" + txtTimKiem.Text + "%'" +

"or MaNhanVien like'%" + txtTimKiem.Text + "%'", con);

SqlDataAdapter da = new SqlDataAdapter(cmd);

DataTable dt = new DataTable();

da.Fill(dt);

dgvHienThiThongTinPhieuNhap.DataSource = dt;

}

private void btnLamMoi\_Click(object sender, EventArgs e)

{

PhieuNhap\_Load(sender, e);

}

}

}

### **Form phiếu nhập sách:**

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

using System.Data.SqlClient;

namespace Menu

{

public partial class PhieuNhapSach : Form

{

SqlConnection con;

public PhieuNhapSach()

{

InitializeComponent();

con = new SqlConnection(@"Data Source=VUHOAINGOC\SQLEXPRESS;Initial Catalog=QuanLiThuVien;Integrated Security=True");

}

public void HienThiThongTin()

{

SqlCommand cmd = new SqlCommand("select \* from PhieuNhapSach", con);

SqlDataAdapter da = new SqlDataAdapter(cmd);

DataTable dt = new DataTable();

da.Fill(dt);

dgvShowThongTin.DataSource = dt;

}

private void PhieuNhapSach\_Load(object sender, EventArgs e)

{

if (con.State == ConnectionState.Closed)

{

con.Open();

}

HienThiThongTin();

SqlCommand cmd = new SqlCommand("select \* from PhieuNhap", con);

SqlDataAdapter da = new SqlDataAdapter(cmd);

DataTable dt = new DataTable();

da.Fill(dt);

cmbMaPhieuNhap.DataSource = dt;

cmbMaPhieuNhap.DisplayMember = "MaPhieuNhap";

cmbMaPhieuNhap.ValueMember = "MaPhieuNhap";

SqlCommand cmd1 = new SqlCommand("select \* from Sach", con);

SqlDataAdapter da1 = new SqlDataAdapter(cmd1);

DataTable dt1 = new DataTable();

da1.Fill(dt1);

cmbMaSach.DataSource = dt1;

cmbMaSach.DisplayMember = "TenSach";

cmbMaSach.ValueMember = "MaSach";

}

private void btnThem\_Click(object sender, EventArgs e)

{

if(cmbMaPhieuNhap.SelectedIndex == -1 || cmbMaSach.SelectedIndex == -1 || txtSoLuong.Text.Trim() == "")

{

MessageBox.Show("Vui lòng điền đầy đủ thông tin trước khi thêm!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Warning);

}

else

{

try

{

SqlCommand cmd = new SqlCommand("insert into PhieuNhapSach(MaPhieuNhap, MaSach, SoLuong) values(@MaPhieuNhap, @MaSach, @SoLuong) ", con);

cmd.Parameters.AddWithValue("@MaPhieuNhap", cmbMaPhieuNhap.SelectedValue);

cmd.Parameters.AddWithValue("@MaSach", cmbMaSach.SelectedValue);

cmd.Parameters.AddWithValue("@SoLuong", txtSoLuong.Text);

if (cmd.ExecuteNonQuery() > 0)

{

MessageBox.Show("Đã hoàn thành việc nhập sách!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Information);

}

else

MessageBox.Show("Việc nhập sách đã xảy ra gián đoạn! Vui lòng kiểm tra lại đầy đủ thông tin!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Information);

HienThiThongTin();

}

catch

{

MessageBox.Show("Mã phiếu nhập cho cuốn sách này đã tồn tại, vui lòng nhập mã phiếu khác!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Warning);

}

}

}

private void PhieuNhapSach\_FormClosed(object sender, FormClosedEventArgs e)

{

if(con.State == ConnectionState.Open)

{

con.Close();

}

}

private void btnSua\_Click(object sender, EventArgs e)

{

if (cmbMaPhieuNhap.SelectedIndex == -1 || cmbMaSach.SelectedIndex == -1 || txtSoLuong.Text.Trim() == "")

{

MessageBox.Show("Vui lòng điền đầy đủ thông tin trước khi sửa!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Warning);

}

else

{

try

{

int dongchon = dgvShowThongTin.CurrentRow.Index;

SqlCommand cmd = new SqlCommand("update PhieuNhapSach set MaPhieuNhap = @MaPhieuNhap," +

" MaSach = @MaSach, SoLuong = @SoLuong where MaPhieuNhap = @MaPhieuNhapCu", con);

cmd.Parameters.AddWithValue("@MaPhieuNhap", cmbMaPhieuNhap.SelectedValue);

cmd.Parameters.AddWithValue("@MaSach", cmbMaSach.SelectedValue);

cmd.Parameters.AddWithValue("@SoLuong", txtSoLuong.Text);

cmd.Parameters.AddWithValue("@MaPhieuNhapCu", dgvShowThongTin.Rows[dongchon].Cells["MaPhieuNhap"].Value.ToString());

if (cmd.ExecuteNonQuery() > 0)

{

MessageBox.Show("Sửa thông tin phiếu nhập sách thành công!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Information);

}

else

MessageBox.Show("Sửa thông tin phiếu nhập sách thất bại!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Information);

HienThiThongTin();

}

catch

{

MessageBox.Show("Tồn tại nhiều mã giống nhau nên không sửa thông tin của một trong các mã phiếu được!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Warning);

}

}

}

private void dgvShowThongTin\_SelectionChanged(object sender, EventArgs e)

{

int dongchon = dgvShowThongTin.CurrentRow.Index;

cmbMaPhieuNhap.SelectedValue = dgvShowThongTin.Rows[dongchon].Cells["MaPhieuNhap"].Value;

cmbMaSach.SelectedValue = dgvShowThongTin.Rows[dongchon].Cells["MaSach"].Value;

txtSoLuong.Text = dgvShowThongTin.Rows[dongchon].Cells["SoLuong"].Value.ToString();

}

private void btnXoa\_Click(object sender, EventArgs e)

{

int dongchon = dgvShowThongTin.CurrentRow.Index;

SqlCommand cmd = new SqlCommand("delete from PhieuNhapSach where MaPhieuNhap = @MaPhieuNhap ", con);

cmd.Parameters.AddWithValue("@MaPhieuNhap", dgvShowThongTin.Rows[dongchon].Cells["MaPhieuNhap"].Value.ToString());

if (cmd.ExecuteNonQuery() > 0)

{

MessageBox.Show("Xóa phiếu nhập sách thành công!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Information);

}

else

MessageBox.Show("Xóa phiếu nhập sách thất bại!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Information);

HienThiThongTin();

}

private void btnTimKiem\_Click(object sender, EventArgs e)

{

SqlCommand cmd = new SqlCommand("select \* from PhieuNhapSach where MaPhieuNhap like'%" + txtTimKiem.Text + "%'" +

"or MaSach like'%" + txtTimKiem.Text + "%'", con);

SqlDataAdapter da = new SqlDataAdapter(cmd);

DataTable dt = new DataTable();

da.Fill(dt);

dgvShowThongTin.DataSource = dt;

}

private void btnLamMoi\_Click(object sender, EventArgs e)

{

PhieuNhapSach\_Load(sender, e);

}

}

}

# **PHẦN V: KẾT LUẬN**

Trong thời gian tìm hiểu làm đồ án đề tài “Quản lí thư viện” nhóm em đã phát triển và hoàn thành cơ bản hệ thống quản lý thư viện. Hệ thống này được xây dựng trên công nghệ web mã nguồn mở ASP.NET MVC và SQL server. Bên cạnh đó, đề tài đã phân tích hệ thống thông tin, xây dựng mô hình chức năng nghiệp vụ, phân tích hệ thống về dữ liệu và xây dựng mô hình thực thể cho hệ thống. Từ đó tạo ra cơ sở dữ liệu tích hợp để xây dựng hệ thống quản lí mượn sách và trả sách. Trong quá trình làm đồ án còn có một số hạn chế về mặt kinh nghiệm, kiến thức nên đồ án khó tránh khỏi sai xót.

Trong thời gian chờ bảo vệ đồ án chúng em sẽ hoàn thiện nốt các phần còn lại của các chức năng đã đặt ra. Đề tài này thật sự hữu ích đối với chuyên ngành công nghệ thông tin nhờ một tài liệu tham khảo đề tài này có tính thực tế cao. Trong tương lai, nếu có điều kiện khách quan cho phép, chúng em sẽ tiếp tục hoàn thiện và phát triển để triển khai hệ thống vào thực tế, giúp cho hệ thống thư viện nhà trường tiết kiệm được thời gian, chi phí cho giáo viên, sinh viên giải quyết các công việc khác.