**BỘ GIÁO DỤC VÀ ĐÀO TẠO**

**TRƯỜNG ĐẠI HỌC QUY NHƠN**

170 An Dương Vương, TP. Quy Nhơn, Bình Định

Điện thoại: 02563 846 156 Fax: 02563 846 089 Web: www.qnu.edu.vn

*Trách nhiệm - Chuyên nghiệp - Chất lượng - Sáng tạo - Nhân văn*

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

****

**BÁO CÁO BÀI TẬP LỚN**

**HỌC PHẦN LẬP TRÌNH TRÊN WINDOWS**

**ĐỀ TÀI**

**QUẢN LÝ THƯ VIỆN**

**GIẢNG VIÊN HƯỚNG DẪN: ĐINH THỊ MỸ CẢNH**

**NGƯỜI THỰC HIỆN**

**Ung Minh Hoài - CNTT-K42A**

**Bình Định, ngày 29, tháng 04, năm 2022**

**Mục lục**

[**I. Giới thiệu** 3](#_Toc102857868)

[**1.** **Giới thiệu đề tài** 3](#_Toc102857869)

[**2.** **Ứng dụng thực tế** 3](#_Toc102857870)

[**II. Những nội dung thực hiện** 3](#_Toc102857871)

[**1.** **Trình bày bài toán** 3](#_Toc102857872)

[**2.** **Công nghệ, tổ chức và giải quyết bài toán** 3](#_Toc102857873)

[**3.** **Thiết kế cơ sở dữ liệu** 4](#_Toc102857874)

[**4.** **Thiết kế giao diện** 4](#_Toc102857875)

[**III. Đánh giá** 18](#_Toc102857876)

[**1.** **Kết quả đạt được** 18](#_Toc102857877)

[**2.** **Hạn chế và hướng phát triển** 19](#_Toc102857878)

[**IV. Tài liệu tham khảo** 19](#_Toc102857879)

1. **Giới thiệu**
2. **Giới thiệu đề tài**

Hiện nay, với xu hướng công nghệ thông tin ngày càng phát triển và hữu dụng trong cuộc sống, được con người khai thác một cách hiệu quả để biến nó thành công cụ hữu ích nhằm nâng cao chất lượng đời sống.

Trong đó thư viện trường học từ lâu đã trở thành một bộ phận không thể thiếu được trong nhà trường, đây là cơ sở vật chất trọng yếu phục vụ giảng dạy và học tập, đồng thời góp phần nâng cao chất lượng giáo dục toàn diện trong nhà trường. Tại các trường học, nhu cầu học tập ngày càng cao, đặc biệt là để đảm bào kiến thức có tính logic và chính xác thì việc tìm đến với sách là rất cần thiết, nhằm đáp ứng nhu cầu đó thì thư viện trường học đã tăng số lượng sách đáng kể, rất phong phú về thể loại sách cũng như số lượng. Công việc quản lý thư viện trở thành một chuỗi công việc vất vả và tốn nhiều công sức. Việc tin học hóa trong quản lý thư viện sẽ giúp việc quản lý trở nên đơn giản và đặc biệt là tính chính xác cao, truy vấn thông tin được nhanh chóng theo nhiều yêu cầu khác nhau.

1. **Ứng dụng thực tế**

Phần mềm quản lý thư viện với các chức năng Quản lý sách, Quản lý độc giả, Quản lý mượn trả sách, tra cứu tìm kiếm. Thống kê báo cáo nhằm đơn giản hóa công tác quản lý thư viện trong nhà trường. Sản phẩm được thiết kế đơn giản, dễ dùng, tiện lợi, tiết kiệm thời gian và công sức cho các cán bộ tác nghiệp cũng như công tác điều hành của ban lãnh đạo. Mỗi một phân hệ được thiết kế liên quan mật thiết như những mắc xích tạo nên tính chuyên nghiệp và khả năng tự động hóa cao.

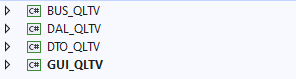
1. **Những nội dung thực hiện**
2. **Trình bày bài toán**

Phần mềm quản lý thư viện giúp cho người quản lý dễ dàng kiểm soát các người dùng, các đầu sách, danh sách độc giả, theo dõi mượn trả.

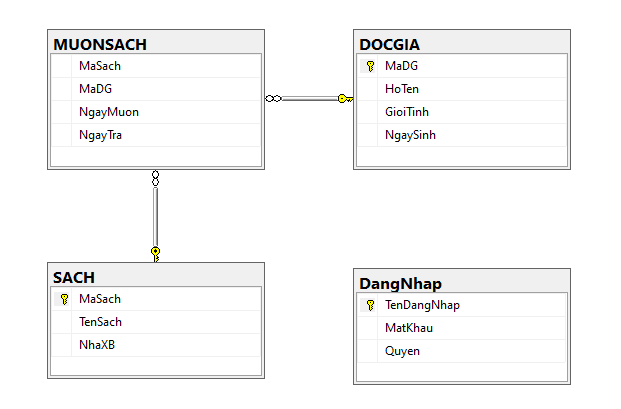
Người dùng có thể đăng ký , đăng nhập sau đó truy cập vào hệ thống để xem và tra cứu các đầu sách.

1. **Công nghệ, tổ chức và giải quyết bài toán**

* Công nghệ sử dụng Winform (Visual Studio 2019 C#)
* Sử dụng mô hình 3 layer



1. **Thiết kế cơ sở dữ liệu**



Hình 1. Cơ sở dữ liệu dưới dạng Diagram

1. **Thiết kế giao diện**

Trình bày một số chức năng chính, mỗi chức năng bao gồm mô tả, thao tác, hình ảnh minh họa.

* **Chuỗi kết nối**

namespace DAL\_QLTV

{

public class DBConnect

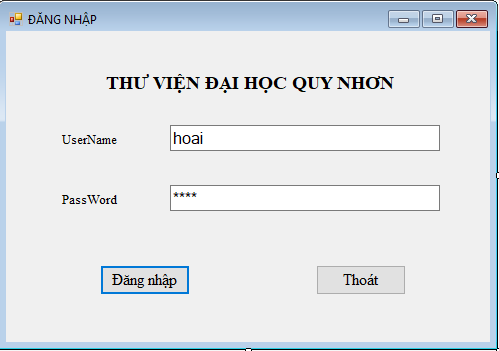
{

protected SqlConnection \_conn = new SqlConnection(@"Data Source=DESKTOP-F1OLJ2J\SQLEXPRESS;Initial Catalog=QLThuVien;Integrated Security=True");

}

}

* 1. **Màn hình “Đăng nhập”**
* Người dùng thực hiện đăng ký khi chưa có tài khoản. Sau đó đăng nhập để truy cập vào hệ thống.

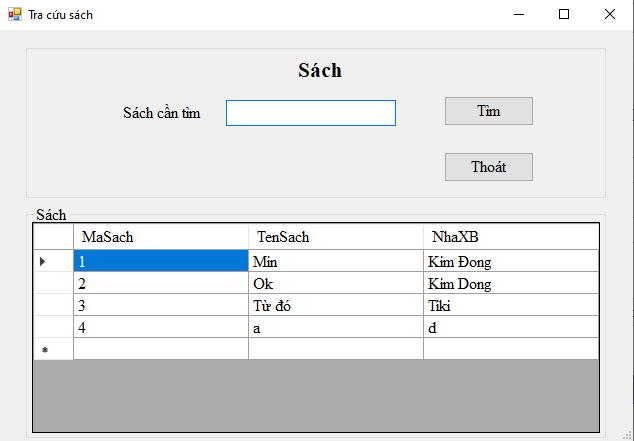


Hình 2. Giao diện đăng nhập

1. Các thành phần trong màn hình:

|  |  |  |
| --- | --- | --- |
| Tên | Kiểu | Mô tả |
| UserName | Textbox | Người dùng nhập UserName |
| PassWord | Textbox | Người dùng nhập PassWord |
| Đăng Nhập | Button | Nhấn nút “Đăng Nhập” khi đã điền đủ thông tin |
| Thoát | Button | Nhấn nút “Thoát” để thoát khỏi chương trình |

* Đối với tài khoản người dùng sẽ chuyển đến giao diện xem và tra cứu sách.

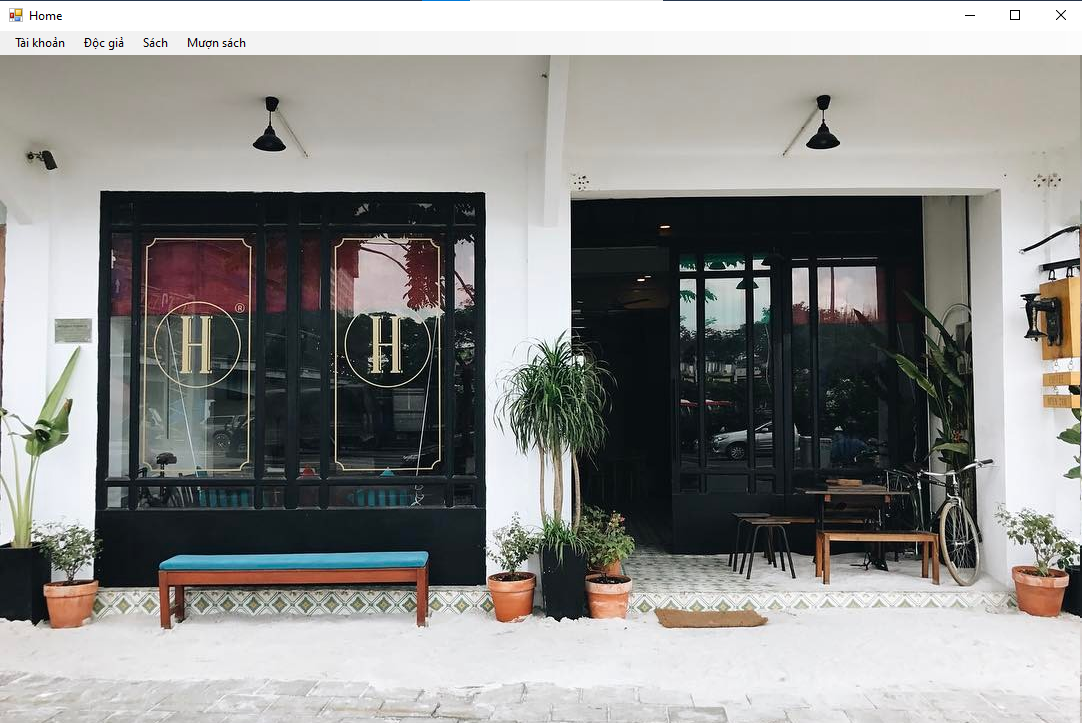


Hình 3. Giao diện tra cứu sách

1. Các thành phần trong màn hình:

|  |  |  |
| --- | --- | --- |
| Tên | Kiểu | Mô tả |
| Sách cần tìm | Textbox | Người dùng nhập vào tên sách cần tìm kiếm |
| Tìm | Button | Nhấn nút “Tìm” để thực hiện tìm kiếm. |
| Sách | DataGridView | Hiển thị danh sách sách |
| Thoát | Button | Nhấn nút “Thoát” để thoát khỏi chương trình |

* Đối với tài khoản người quản lý sẽ chuyển đến giao diện quản lý chung.



Hình 4. Giao diện quản lý

1. Câu lệnh cho nút “Đăng nhập”

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

{

string username = txtUserName.Text.Trim();

string password = txtPassWord.Text.Trim();

try

{

if(busTK.checkLogin(username, password) == 1){

frmMain f = new frmMain();

f.ShowDialog();

}

else if(busTK.checkLogin(username, password) == 2)

{

frmXemSach f = new frmXemSach();

f.ShowDialog();

}

else

{

MessageBox.Show("Tên đăng nhập/Mật khẩu không hợp lệ");

}

}

catch (Exception ex)

{

MessageBox.Show("Lỗi kết nối CSDL!");

return;

}

finally

{

}

}

1. Câu lệnh cho nút “Thoát”

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

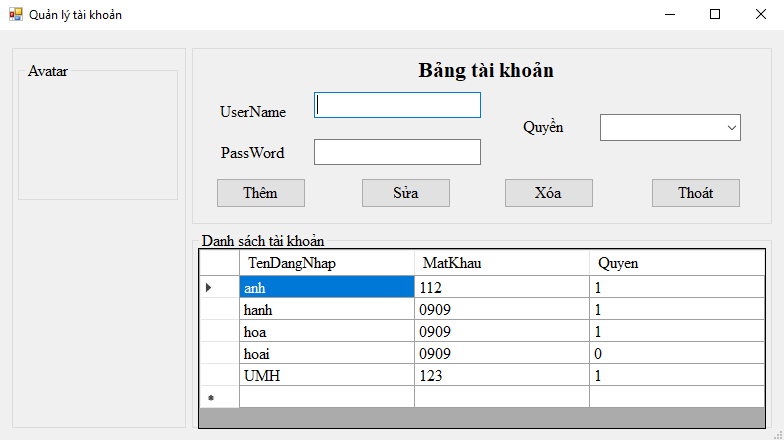
{

this.Close();

}

* 1. **Quản lý tài khoản**

Chức năng quản lý tài khoản dành cho người quản lý có thể thêm, sửa, xóa tài khoản.



Hình 5. Giao diện quản lý tài khoản

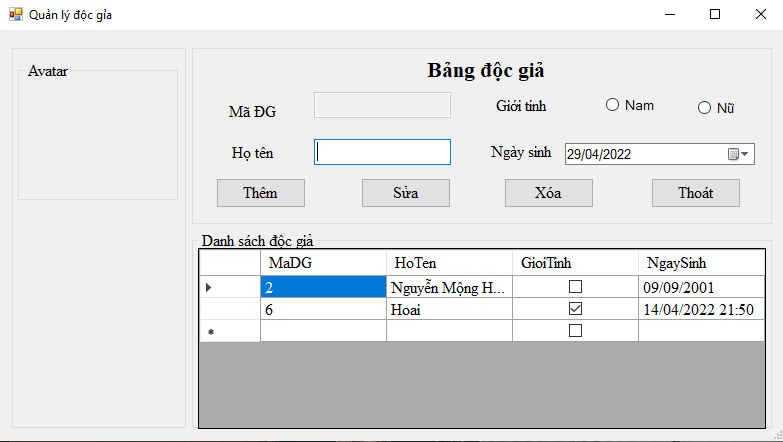
1. Các thành phần trong màn hình:

|  |  |  |
| --- | --- | --- |
| Tên | Kiểu | Mô tả |
| UserName | Textbox | Người dùng nhập UserName |
| PassWord | Textbox | Người dùng nhập PassWord |
| Quyền | Combobox | Chọn loại tài khoản |
| Danh sách tài khoản | DataGridView | Hiển thị danh sách tài khoản |
| Thêm, Xóa, Sửa, Thoát | Button | Các nút chức năng thêm, xóa, sửa tài khoản và thoát khỏi giao diện hiện tại |

* 1. **Quản lý độc giả**

Chức năng quản lý độc giả giúp người quản lý kiểm soát được thông tin các độc giả đã và đang mượn sách tại thư viện. Người quản lý hoàn toàn có thể thêm, sửa, xóa độc giả.

***Note:*** khi thực hiện xóa độc giả thì thông tin về mượn trả sách cũng sẽ được xóa.



Hình 6. Giao diện quản lý độc giả

1. Các thành phần trong màn hình:

|  |  |  |
| --- | --- | --- |
| Tên | Kiểu | Mô tả |
| Mã ĐG | Textbox | Xem mã độc giả |
| Họ tên | Textbox | Người dùng nhập họ tên |
| Giới tính | Radio Button | Chọn giới tính |
| Ngày sinh | DateTimePicker | Chọn ngày sinh |
| Danh sách độc giả | DataGridView | Hiển thị danh sách độc giả |
| Thêm, Xóa, Sửa, Thoát | Button | Các nút chức năng thêm, xóa, sửa độc giả và thoát khỏi giao diện hiện tại |

1. Câu lệnh các nút “Thêm”, “Xóa”, “Sửa”:

* GUI\_QLTV

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

{

if (txtHoTen.Text != "")

{

bool gt;

if (rbNam.Checked)

{

gt = true;

}

else

{

gt = false;

}

// Tạo DTo

DTO\_DocGia tv = new DTO\_DocGia(txtMaDG.Text, txtHoTen.Text, gt, dtpNgaySinh.Value); // Vì ID tự tăng nên để ID số gì cũng dc

// Them

if (busDG.themDocGia(tv))

{

MessageBox.Show("Thêm thành công");

dtgvDG.DataSource = busDG.getDocGia(); // refresh datagridview

}

else

{

MessageBox.Show("Thêm không thành công");

}

}

else

{

MessageBox.Show("Xin hãy nhập đầy đủ");

}

}

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

{

// Kiểm tra nếu có chọn table rồi

Int32 selectedCellCount =

dtgvDG.GetCellCount(DataGridViewElementStates.Selected);

if (selectedCellCount > 0)

{

if (txtMaDG.Text != "" && txtHoTen.Text != "")

{

bool gt;

if (rbNam.Checked)

{

gt = true;

}

else

{

gt = false;

}

// Tạo DTo

DTO\_DocGia tv = new DTO\_DocGia(txtMaDG.Text, txtHoTen.Text, gt, dtpNgaySinh.Value);

// Sửa

if (busDG.suaDocGia(tv))

{

MessageBox.Show("Sửa thành công");

dtgvDG.DataSource = busDG.getDocGia(); // refresh datagridview

}

else

{

MessageBox.Show("Sửa ko thành công");

}

}

else

{

MessageBox.Show("Xin hãy nhập đầy đủ");

}

}

else

{

MessageBox.Show("Hãy chọn thành viên muốn sửa");

}

}

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

{

// Kiểm tra nếu có chọn table rồi

Int32 selectedCellCount =

dtgvDG.GetCellCount(DataGridViewElementStates.Selected);

if (selectedCellCount > 0)

{

string tdn = txtMaDG.Text.Trim();

if (busMS.xoaMuonSach2(tdn))

{

if (busDG.xoaDocGia(tdn))

{

MessageBox.Show("Xóa thành công");

dtgvDG.DataSource = busDG.getDocGia(); // refresh datagridview

}

else

{

MessageBox.Show("Xóa ko thành công");

}

}

else

{

MessageBox.Show("Xóa không thành công");

}

}

}

* **DAL\_DOCGIA**

public bool themDocGia(DTO\_DocGia s)

{

try

{

// Ket noi

\_conn.Open();

// Query string - vì mình để TV\_ID là identity (giá trị tự tăng dần) nên ko cần fải insert ID

string SQL = "INSERT INTO DOCGIA(HoTen, GioiTinh, ngaysinh) VALUES (@ht, @gt,@ns)";

// Command (mặc định command type = text nên chúng ta khỏi fải làm gì nhiều).

SqlCommand cmd = new SqlCommand(SQL, \_conn);

cmd.Parameters.AddWithValue("@ht", s.HoTen);

cmd.Parameters.AddWithValue("@gt", s.GioiTinh);

cmd.Parameters.AddWithValue("@ns", s.NgaySinh);

// Query và kiểm tra

if (cmd.ExecuteNonQuery()>0)

return true;

\_conn.Close();

}

catch (Exception e)

{

}

finally

{

// Dong ket noi

\_conn.Close();

}

return false;

}

//Sửa

public bool suaDocGia(DTO\_DocGia dg)

{

try

{

// Ket noi

\_conn.Open();

int ma = Convert.ToInt32(dg.MaDG);

// Query string

string SQL = string.Format("UPDATE DOCGIA SET HoTen = @ht, GioiTinh = @gt, NgaySinh = @ns WHERE MaDG = '{0}'", ma);

// Command (mặc định command type = text nên chúng ta khỏi fải làm gì nhiều).

SqlCommand cmd = new SqlCommand(SQL, \_conn);

cmd.Parameters.AddWithValue("@ht", dg.HoTen);

cmd.Parameters.AddWithValue("@gt", dg.GioiTinh);

cmd.Parameters.AddWithValue("@ns", dg.NgaySinh);

// Query và kiểm tra

if (cmd.ExecuteNonQuery() > 0)

return true;

}

catch (Exception e)

{

}

finally

{

// Dong ket noi

\_conn.Close();

}

return false;

}

public bool xoaDocGia(string id)

{

try

{

// Ket noi

\_conn.Open();

// Query string - vì xóa chỉ cần ID nên chúng ta ko cần 1 DTO, ID là đủ

string SQL = string.Format("DELETE FROM DOCGIA WHERE MaDG = '{0}'", id);

// Command (mặc định command type = text nên chúng ta khỏi fải làm gì nhiều).

SqlCommand cmd = new SqlCommand(SQL, \_conn);

// Query và kiểm tra

if (cmd.ExecuteNonQuery() > 0)

return true;

}

catch (Exception e)

{

}

finally

{

// Dong ket noi

\_conn.Close();

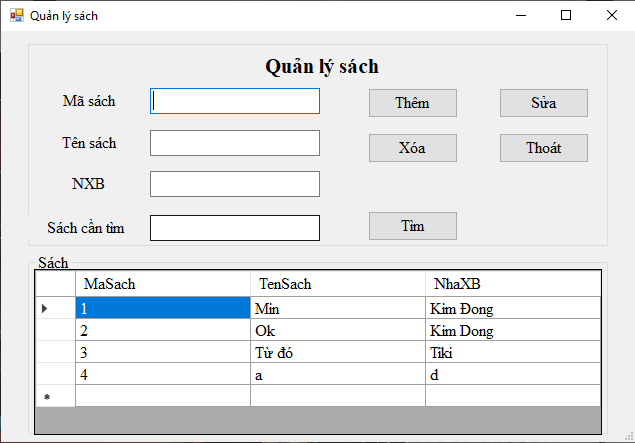
}

return false;

}

* 1. **Quản lý sách**

Chức năng quản lý sách giúp người quản lý thêm, sửa, xóa các đầu sách trong kho thư viện. Để hỗ trợ trong trường hợp có nhiều dữ liệu đầu sách việc tìm kiếm khó khăn thì ứng dụng đã cung cấp thêm chức năng tìm kiếm.



Hình 7. Giao diện quản lý sách

1. Các thành phần trong màn hình:

|  |  |  |
| --- | --- | --- |
| Tên | Kiểu | Mô tả |
| Mã sách | Textbox | Người dùng nhập mã sách |
| Tên sách | Textbox | Người dùng nhập tên sách |
| NXB | Textbox | Người dùng nhập nhà xuất bản |
| Sách cần tìm | Textbox | Nhập vào sách cần tìm |
| Sách | DataGridView | Hiển thị danh sách sách |
| Thêm, Xóa, Sửa, Tìm, Thoát | Button | Các nút chức năng thêm, xóa, sửa , tìm kiếm sách và thoát khỏi giao diện hiện tại |

1. Câu lệnh nút “Tìm”

* **GUI\_QLTV**

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

{

if (txtTim.Text != "")

{

dtgvSach.DataSource = busSach.timKiem(txtTim.Text);

}

}

* **DAL\_Sach:**

public DataTable timKiem(string s)

{

string sql = string.Format("SELECT \* FROM SACH WHERE TenSach Like N'%{0}'", s);

SqlDataAdapter search = new SqlDataAdapter(sql, \_conn);

DataTable dt = new DataTable();

search.Fill(dt);

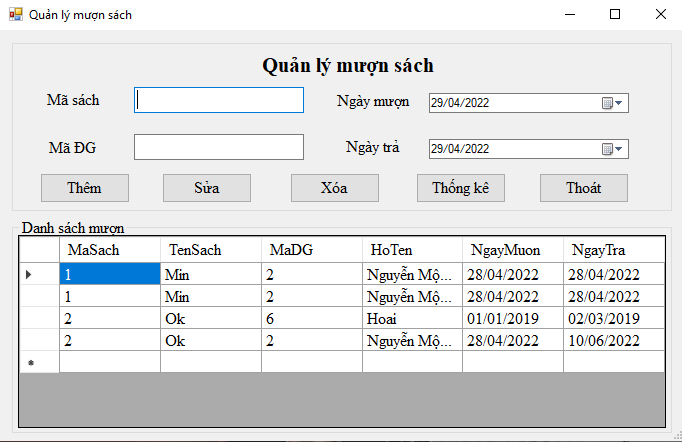
return dt;

//data.DataSource = dt;

}

* 1. **Quản lý mượn trả sách**

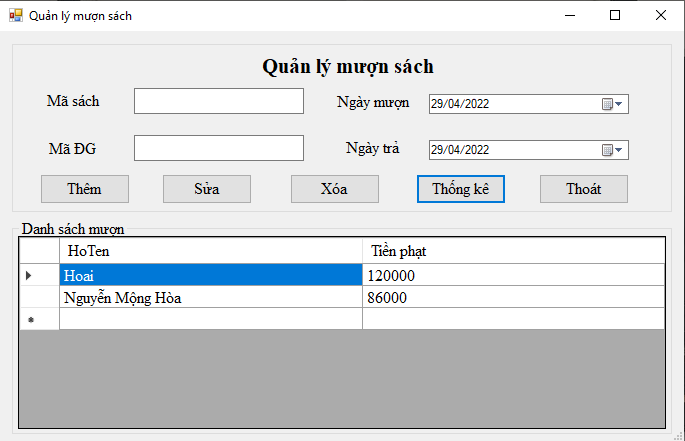
Chức năng quản lý mượn trả sách giúp thủ thư có thể quản lý được tình trạng mượn trả sách của người dùng. Hệ thống còn cung cấp thêm chức năng thống kê những độc giả quá hạn trả sách và tiền phạt dựa vào số ngày quá hạn nhân với 2.000(đồng)



Hình 8. Giao diện mượn trả sách

1. Các thành phần trong màn hình:

|  |  |  |
| --- | --- | --- |
| Tên | Kiểu | Mô tả |
| Mã sách | Textbox | Người dùng nhập mã sách |
| Mã ĐG | Textbox | Người dùng nhập mã độc giả |
| Ngày mượn | DateTimePicker | Nhập ngày mượn |
| Ngày trả | DateTimePicker | Nhập ngày trả |
| Danh sách mượn | DataGridView | Hiển thị danh sách mượn trả |
| Thêm, Xóa, Sửa, Thống kê, Thoát | Button | Các nút chức năng thêm, xóa, sửa, thống kế mượn trả và thoát khỏi giao diện hiện tại |



Hình 9. Giao diện thống kê

1. Câu lệnh nút “Thống kê”

* **GUI\_QLTV**

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

{

dtgvMuonSach.DataSource = busSach.getMuonSachQuaHan();

}

* **DAL\_MUONSACH**

public DataTable getMuonSachQuaHan()

{

SqlDataAdapter daSach = new SqlDataAdapter("Select HoTen, N'Tiền phạt' = DATEDIFF(day, b.NgayMuon, b.NgayTra)\*2000 from DOCGIA as a, MUONSACH as b where a.MaDG = b.MaDG and DATEDIFF(day, NgayMuon, NgayTra) > 7", \_conn);

DataTable dtSach3 = new DataTable();

daSach.Fill(dtSach3);

return dtSach3;

}

1. **Đánh giá**
2. **Kết quả đạt được**

Qua việc thực hành Project em đã nắm được thêm những kiến thức về cách tổ chức của một ứng dụng theo mô hình 3 lớp giúp thuận tiện hơn trong việc code, giúp rút ngắn số lượng code và ứng dụng cũng được trình bày một cách logic hơn.

Ngoài ra, cải thiện và nâng cao khả năng tự học của bản thân.

1. **Hạn chế và hướng phát triển**

* **Hạn chế:**

Vì thời gian ngắn, phầm mềm đã hoàn thành tuy nhiên vẫn còn nhiều hạn chế và thiếu.

Với tính chất phục vụ cho quá trình học tập, tìm hiểu nên chưa xây dựng đầy đủ các chức năng như một hệ thống đã có.

* **Hướng phát triển:**

Quản lý thêm nhiều thông tin độc giả hơn, SEO tốt hơn, Hỗ trợ gửi mail nhắc nhở khi trễ hạn trả sách hoặc các chương trình được tổ chức bởi thư viện.

1. **Tài liệu tham khảo**

Vận dụng các bài tập thực hành được học.