**TRƯỜNG ĐẠI HỌC SAO ĐỎ**

KHOA CÔNG NGHỆ THÔNG TIN



**BÁO CÁO PHÂN TÍCH THIẾT KẾ HƯỚNG ĐỐI TƯỢNG**

**ĐỀ TÀI:** **QUẢN LÝ THƯ VIỆN**

|  |  |  |
| --- | --- | --- |
| **Sinh viên thực hiện:** | 1.  2. | **Trương Văn Tuấn**  **Ngô Thị Hải Yến** |
|  |
| **Lớp:** | **DK10 - CNTT** | |

**Giảng viên hướng dẫn:** Nguyễn Thị Ánh Tuyết

**Khoa:** Công nghệ thông tin.

Hải Dương - 2022

**LỜI CẢM ƠN**

Sau quá trình học tập và rèn luyện tại khoa Công nghệ thông tin tại trường Đại học Sao Đỏ chúng em đã được trang bị các kiến thức cơ bản, các kỹ năng thực tế để có thể hoàn thành bài tập lớn môn học của mình.

Chúng em xin gửi lời cảm ơn chân thành đến cô Nguyễn Thị Ánh Tuyết đã quan tâm, hướng dẫn, truyền đạt những kiến thức và kinh nghiệm cho chúng em trong suốt thời gian học tập môn Phân tích thiết kế hướng đối tượng.

Trong quá trình làm bài tập lớn không tránh khỏi được những sai sót, chúng em mong nhận được sự góp ý của quý cô và các bạn để được hoàn thiện hơn.

TP. Chí Linh,…tháng ,…năm 2022.

**MỤC LỤC**

[MỞ ĐẦU 1](#_Toc103983500)

[CHƯƠNG 1: CƠ SỞ LÝ THUYẾT 3](#_Toc103983501)

[1.1. Microѕoft Viѕual Studio 3](#_Toc103983502)

[*1.1.1.* *Khái niệm Visual studio* 3](#_Toc103983503)

[*1.1.2.* *Tính năng của Visual Studio* 3](#_Toc103983504)

[1.2. SQL Server 5](#_Toc103983505)

[*1.2.1.* *Khái niệm* 5](#_Toc103983506)

[*1.2.2.* *Thành phần* 5](#_Toc103983507)

[*1.2.3.* *Mục đích sử dụng* 5](#_Toc103983508)

[*1.2.4.* *Phân loại* 5](#_Toc103983509)

[*1.2.5.* *Các phiên bản* 6](#_Toc103983510)

[1.3. Devexpress 7](#_Toc103983511)

[*1.3.1.* *Khái niệm* 7](#_Toc103983512)

[*1.3.2.* *Thành phần* 7](#_Toc103983513)

[*1.3.3.* *Cài đặt* 7](#_Toc103983514)

[1.4. EntityFramework 7](#_Toc103983515)

[*1.4.1.* *Khái niệm* 7](#_Toc103983516)

[*1.4.2.* *Lịch sử ra đời* 7](#_Toc103983517)

[*1.4.3.* *Cấu trúc* 8](#_Toc103983518)

[*1.4.4.* *Tính năng* 9](#_Toc103983519)

[*1.4.5.* *Các tình huống dùng Entity Framework* 9](#_Toc103983520)

[1.5. Mô hình 3 lớp 10](#_Toc103983521)

[*1.5.1* *Khái niệm* 10](#_Toc103983522)

[*1.5.2* *Các thành phần chính của mô hình 3 lớp* 11](#_Toc103983523)

[CHƯƠNG 2: PHÂN TÍCH THIẾT KỆ HỆ THỐNG 14](#_Toc103983524)

[2.1 Kiến trúc hệ thống 14](#_Toc103983525)

[2.2 Mô tả các thành phần trong hệ thống. 14](#_Toc103983526)

[2.3 Thuật toán lập sơ đồ logic: 16](#_Toc103983527)

[*2.3.1* *Bước 1: Xét yêu cầu phần mềm “Lập thẻ độc giả”* 16](#_Toc103983528)

[*2.3.2* *Bước 2: Xét yêu cầu phần mềm “Tiếp nhận sách mới”* 17](#_Toc103983529)

[*2.3.3* *Bước 3: Xét yêu cầu phần mềm “Tra cứu sách”* 18](#_Toc103983530)

[*2.3.4* *Bước 4: Xét yêu cầu phần mềm “Cho mượn sách”* 19](#_Toc103983531)

[*2.3.5* *Bước 5: Xét yêu cầu phần mềm “Nhận trả sách”* 20](#_Toc103983532)

[*2.3.6* *Bước 6: Xét yêu cầu phần mềm “Lập phiếu thu tiền phạt”* 22](#_Toc103983533)

[*2.3.7* *Bước 7: Xét yêu cầu phần mềm “Lập báo cáo”* 23](#_Toc103983534)

[2.3.8 Bước 8: Xét yêu cầu phần mềm “Thay đổi quy định” 25](#_Toc103983535)

[2.4 Phân thích một số quy trình trong thiết kế hệ thống 26](#_Toc103983536)

[*2.4.1* *Mô hình usecase mức tổng quát* 26](#_Toc103983537)

[*2.4.2* *Mô hình usecase mức dưới tổng quát* 26](#_Toc103983538)

[*2.4.3* *Phân rã use case quản độc giả* 26](#_Toc103983539)

[*2.4.4* *Phân rã use case quản lý sách* 27](#_Toc103983540)

[*2.4.5* *Phân rã use case quản lý mượn trả* 27](#_Toc103983541)

[*2.4.6* *Phân rã use case thống kê* 28](#_Toc103983542)

[*2.4.7* *Phân rã use case tìm kiếm* 28](#_Toc103983543)

[*2.4.8* *Quy trình quản lý mượn sách* 29](#_Toc103983544)

[*2.4.9* *Quy trình quản lý trả sách* 29](#_Toc103983545)

[2.5 Sơ đồ logic hoàn chỉnh: 30](#_Toc103983546)

[2.6 Danh sách các bảng dữ liệu (table) trong sơ đồ: 30](#_Toc103983547)

[2.7 Mô tả từng bảng dữ liệu 31](#_Toc103983548)

[CHƯƠNG 3: XÂY DỰNG CHƯƠNG TRÌNH 38](#_Toc103983549)

[3.1 Cài đặt dev express 38](#_Toc103983550)

[3.1.1 Chạy trình cài đặt 38](#_Toc103983551)

[3.1.2 Đăng ký sản phẩm 38](#_Toc103983552)

[3.1.3 Chọn các sản phẩm 38](#_Toc103983553)

[3.1.4 Chọn đường dẫn đích 39](#_Toc103983554)

[3.1.5 Sơ đồ liên kết các màn hình 40](#_Toc103983555)

[3.2 Danh sách các màn hình 41](#_Toc103983556)

[3.3 Mô tả các màn hình 43](#_Toc103983557)

[*3.3.1* *Màn hình chính* 43](#_Toc103983558)

[*3.3.2* *Màn hình đăng nhập* 45](#_Toc103983559)

[*3.3.3* *Màn hình độc giả* 46](#_Toc103983560)

[*3.3.4* *Màn hình Sách* 47](#_Toc103983561)

[*3.3.5* *Màn hình Loại Độc Giả* 49](#_Toc103983562)

[*3.3.6* *Màn hình Phiếu Thu Phạt Tiền* 50](#_Toc103983563)

[*3.3.7* *Màn hình Phiếu mượn sách* 51](#_Toc103983564)

[*3.3.8* *Màn hình Phiếu trả sách* 52](#_Toc103983565)

[3.4 Hoàn thiện logic 53](#_Toc103983566)

[*3.4.1* *Chức năng phân quyền* 53](#_Toc103983567)

[*3.4.2* *Chức năng đăng nhập* 54](#_Toc103983568)

[*3.4.3* *Chức năng chuyển frame phụ* 55](#_Toc103983569)

[*3.4.4* *Chức năng lấy dữ liệu lên bảng* 55](#_Toc103983570)

[*3.4.5* *Chức năng thêm sách* 56](#_Toc103983571)

[*3.4.6* *Chức năng lấy dữ liệu từ bảng lên ô text* 56](#_Toc103983572)

[*3.4.7* *Lấy danh sách phiếu phạt* 57](#_Toc103983573)

[*3.4.8* *Chức năng trả sách* 58](#_Toc103983574)

[*3.4.9* *Chức năng mượn sách* 59](#_Toc103983575)

[*3.5 Kết quả đạt được 60*](#_Toc103983576)

[*NHẬN XÉT VÀ KẾT LUẬN* 68](#_Toc103983577)

[*Ưu điểm* 68](#_Toc103983578)

[*Khuyết điểm* 68](#_Toc103983579)

[*Kết quả đạt được* 68](#_Toc103983580)

[*Hướng phát triển* 68](#_Toc103983581)

[*TÀI LIỆU THAM KHẢO* 68](#_Toc103983582)

[*1. Tiếng Việt 68*](#_Toc103983583)

[*2. Tiếng Anh 69*](#_Toc103983584)

**DANH MỤC HÌNH ẢNH**

[Hình 1.1:Giao diện phần mềm Microsoft Visual Studio 3](#_Toc103981289)

[Hình 1.2: Cấu trúc entity framework 8](#_Toc103981290)

[Hình 1.3: Sơ đồ database first 9](#_Toc103981291)

[Hình 1.4: Sơ đồ code First entity 10](#_Toc103981292)

[Hình 1.5: Sơ đồ Model First 10](#_Toc103981293)

[Hình 1.6: Minh họa mô hình 3 lớp 11](#_Toc103981294)

[Hình 1.7: Tầng presentation layer 12](#_Toc103981295)

[Hình 1.8: Tầng Bus logic layer 12](#_Toc103981296)

[Hình 1.9: Tầng Access layer 13](#_Toc103981297)

[Hình 2.1: Bảng độc giả 16](#_Toc103981298)

[Hình 2.2: Sơ đồ cơ sở dữ liệu tiếp nhận sách mới 18](#_Toc103981299)

[Hình 2.3: Sơ đồ cơ sở dữ liệu yêu cầu mượn sách tính đúng đắng 19](#_Toc103981300)

[Hình 2.4: Sơ đồ cơ sở dữ liệu yêu cầu mượn sách tính tiến hóa 20](#_Toc103981301)

[Hình 2.5: Sơ đồ cơ sở dữ liệu nhận trả sách tính đúng đắng 21](#_Toc103981302)

[Hình 2.6: Sơ đồ cơ sở dữ liệu nhận trả sách tính tiến hóa 22](#_Toc103981303)

[Hình 2.7: Sơ đồ cơ sở dữ liệu phiếu phạt tính đúng đắng 23](#_Toc103981304)

[Hình 2.8: Sơ đồ cơ sở dữ liệu báo cáo mượn sách tính đúng đắng 24](#_Toc103981305)

[Hình 2.9: Sơ đồ cơ sở dữ liệu sách trả trễ tính đúng đắng 25](#_Toc103981306)

[Hình 2.10: Biểu đồ use case mức tổng quát 26](#_Toc103981307)

[Hình 2.11: Biểu đồ use case mức dưới tổng quát 26](#_Toc103981308)

[Hình 2.12: Phân rã use case quản lý đôc giả 27](#_Toc103981309)

[Hình 2.13: Phân rã use case quản lý sách 27](#_Toc103981310)

[Hình 2.14: Phân rã use case quản lý mượn trả sách 28](#_Toc103981311)

[Hình 2.15: Phân rã use case thống kê 28](#_Toc103981312)

[Hình 2.16: Phân rã use case tìm kiếm 29](#_Toc103981313)

[Hình 2.17: Biểu đồ trạng thái cho lớp mượn sách 29](#_Toc103981314)

[Hình 2.18: Biểu đồ trạng thái cho lớp trả sách 30](#_Toc103981315)

[Hình 2.19: Minh họa sơ đồ cơ sở dữ liệu 30](#_Toc103981316)

[Hình 3.1:Cài đặt file dev express 38](#_Toc103981317)

[Hình 3.2: Đăng ký dev express 38](#_Toc103981318)

[Hình 3.3: Chọn sản phẩm dev express 39](#_Toc103981319)

[Hình 3.4: Chọn đường dẫn dev express 39](#_Toc103981320)

[Hình 3.5: Cài đặt dev express 40](#_Toc103981321)

[Hình 3.6: Sơ đồ liên kết các màn hình 41](#_Toc103981322)

[Hình 3.7: Giao diện thiết kế frmthongtinpm 43](#_Toc103981323)

[Hình 3.8: Giao diện thiết kế frmlogin 45](#_Toc103981324)

[Hình 3.9: Giao diện thiết kế frmdocgia 46](#_Toc103981325)

[Hình 3.10: Giao diện thiết kế frmsach 47](#_Toc103981326)

[Hình 3.11: Giao diện thiết kế frmdocgia 49](#_Toc103981327)

[Hình 3.12: Giao diện thiết kế frmDSphieuphat 50](#_Toc103981328)

[Hình 3.13: Giao diện mượn sách 51](#_Toc103981329)

[Hình 3.14: Giao diện thiết kế frmTTTraSach 52](#_Toc103981330)

[Hình 3.15: Giao diện đăng nhập 60](#_Toc103981331)

[Hình 3.16:Giao diện load nếu đăng nhập thành công 61](#_Toc103981332)

[Hình 3.17: Giao diện admin(quyền cao nhất) 61](#_Toc103981333)

[Hình 3.18: ADMIN (quyền quản lý độc giả) 61](#_Toc103981334)

[Hình 3.19: ADMIN (quyền quản lý sách) 61](#_Toc103981335)

[Hình 3.20: ADMIN (quyền quản lý mượn trả) 61](#_Toc103981336)

[Hình 3.21: Giao diện của độc giả 62](#_Toc103981337)

[Hình 3.22: Giao diện đổi mật khẩu 62](#_Toc103981338)

[Hình 3.23: Giao diện quản lí độc giả 63](#_Toc103981339)

[Hình 3.24: Giao diện thêm độc giả 63](#_Toc103981340)

[Hình 3.25: Giao diện loại độc giả 64](#_Toc103981341)

[Hình 3.26: Giao diện thu tiền phạt 64](#_Toc103981342)

[Hình 3.27: Giao diện quản lí sách 64](#_Toc103981343)

[Hình 3.28: Giao diện quản lí mượn sách 65](#_Toc103981344)

[Hình 3.29: Giao diện trả sách 65](#_Toc103981345)

[Hình 3.30: Giao diện thống kế sách mượn 66](#_Toc103981346)

[Hình 3.31: Giao diện tạo báo cáo 66](#_Toc103981347)

[Hình 3.32: Thống kê sách trả trễ 66](#_Toc103981348)

[Hình 3.33: Giao diện quản lí quy định 67](#_Toc103981349)

# **MỞ ĐẦU**

**Lý do chọn đề tài**

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 “Quản lý thư viện”. 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

Trong thời đại tri thức ngày nay, việc nâng cao chất lượng giáo dục là nhiệm vụ quan trọng và hàng đầu của nước ta. Song song với việc đào tạo, việc quản lý cũng không kém phần quan trọng đặc biệt là việc quản sách trong các thư viện . Hằng ngày một số lượng lớn sách trong các thư viện được sử dụng. Việc quản lý sách vốn dĩ đã khá khó khăn nhưng do nhu cầu đọc của chúng ta mỗi ngày càng tăng nên việc quản lý sách trong các thư viện cũng khó khăn hơn. Từ đó việc tạo ra Phần mềm Quản lý thư viện như một phần tất yếu. Phần mềm giúp giảo quyết phần nào khó khăn đặt ra ở trên. Phần mềm hỗ trợ các chức năng thông dụng mà nhiệm vụ của một thủ thư hay làm và hỗ trợ việc lưu trữ dữ liệu sách.

**Mục đích của đề tài**

* Giảm thiểu lượng công việc thủ công.
* Nâng cao hiệu quả trong công tác quản lí thư viện.
* Bảo mật thông tin và tiết kiệm thời gian.
* Tạo ra công cụ cho các thư viện dễ dàng quản lý thư viện tốt hơn.

**Yêu cầu**

* Giao diện và bố cục phần mềm sắp xếp hợp lí, rõ ràng, tiện lợi cho người dùng cũng như việc quản lí.
* Phần mềm có thể lưu trữ một lượng lớn thông tin về độc giả, sách, thể loại….
* Bộ phận quản lí có thể điều chỉnh, cập nhập các thông tin thư viện.
* Hệ thống cập nhật liên tục các thay đổi của thư viện.

**Ưu và nhược điểm của hệ thống hiện tại**

*Ưu điểm:*

* Không tốn kém cho việc đầu tư cơ sở vật chất.
* Nhân viên thư viện không cần phải hiểu biết về tin học nhiều cũng có thể làm được.

*Nhược điểm:*

* Thông tin ghi trên sổ sách, giấy tờ dễ bị mất mát, không lưu trữ được lâu dài.
* Quá trình tìm kiếm sách phục vụ cho việc mượn sách phải làm thủ công, do đó gây mất nhiều thời gian.
* Nhân viên phải tốn nhiều thời gian, công sức vào việc thống kê sách.
* Tất cả các công việc của nhân viên thư viện đều tiến hành một cách thủ công, không khoa học.

**Đối tượng nghiên cứu**

* Nghiên cứu cách thức quản lí sách, các qui trình quản lí thông tin thư viện trong hệ thống quản lí Trường Đại học Sao Đỏ.
* Hệ quản trị CSDL SQL Server và các ứng dụng
* Phương pháp xây dựng hệ quản trị CSDL SQL Server để làm phần mềm quản lý thư viện

**Phạm vi nghiên cứu**

* Quy trình xây dựng chương trình quản lí thư viện
* Tìm hiểu hệ quản trị CSDL SQL Server.

**Phương pháp nghiên cứu**

* Nghiên cứu tài liệu.
* Khảo sát thực tế.
* Xây dựng chương trình theo những yêu cầu đã đặt ra.

**Quy trình thực hiện các công việc chính**

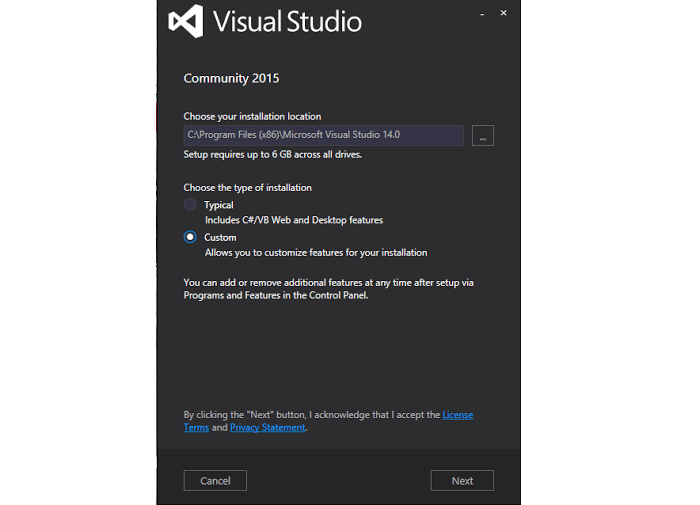
* Giao diện và bố cục phần mềm sắp xếp hợp lí, rõ ràng, tiện lợi cho người dùng cũng như việc quản lí.
* Phần mềm có thể lưu trữ một lượng lớn thông tin về độc giả, sách, thể loại.
* Bộ phận quản lí có thể điều chỉnh, cập nhập các thông tin thư viện.
* Hệ thống cập nhật liên tục các thay đổi của thư viện.

# **CHƯƠNG 1: CƠ SỞ LÝ THUYẾT**

1. **Microѕoft** **Viѕual Studio**
2. ***Khái niệm Visual studio***

Visual studio là một phần mềm hỗ trợ đắc lực hỗ trợ công việc lập trình website. Công cụ này được tạo lên và thuộc quyền sở hữu của ông lớn công nghệ Microsoft. Năm 1997, phần mềm lập trình nay có tên mã Project Boston. Nhưng sau đó, Microsoft đã kết hợp các công cụ phát triển, đóng gói thành sản phẩm duy nhất.

Visual Studio là hệ thống tập hợp tất cả những gì liên quan tới phát triển ứng dụng, bao gồm trình chỉnh sửa mã, trình thiết kế, gỡ lỗi. Tức là, bạn có thể viết code, sửa lỗi, chỉnh sửa thiết kế ứng dụng dễ dàng chỉ với 1 phần mềm Visual Studio mà thôi. Không dừng lại ở đó, người dùng còn có thể thiết kế giao diện, trải nghiệm trong Visual Studio như khi phát triển ứng dụng Xamarin, UWP bằng XAML hay Blend vậy.



Hình 1.1:Giao diện phần mềm Microsoft Visual Studio

Phần mềm Visual studio được chia thành 2 phiên bản Visual Studio Enterprise và Visual Studio Professional, các phiên bản cao cấp có tính phí này được sử dụng nhiều bởi các công ty chuyên về lập trình. Bên cạnh đó, Microsoft cũng cho ra mắt phiên bản Community (phiên bản miễn phí) của gói phần mềm, cung cấp cho người dùng những tính năng cơ bản nhất, phù hợp với các đối tượng lập trình không chuyên, mới tiếp cận tìm hiểu về công nghệ (đối tượng nghiên cứu, nhà phát triển cá nhân, hỗ trợ dự án mở, các tổ chức phi doanh nghiệp dưới 5 người dùng).

1. ***Tính năng của Visual Studio***

* *Đa nền tảng*

Phần mềm lập trình Visual Studio của Microsoft hỗ trợ sử dụng trên nhiều nền tảng khác nhau. Không giống như các trình viết code khác, Visual Studio sử dụng được trên cả Windows, Linux và Mac Systems. Điều này cực kỳ tiện lợi cho lập trình viên trong quá trình ứng dụng.

* *Đa ngôn ngữ lập trình*

Không chỉ hỗ trợ đa nền tảng, Visual Studio cũng cho phép sử dụng nhiều ngôn ngữ lập trình khác nhau từ C#, F#, C/C++, HTML, CSS, Visual Basic, JavaScript,… Bởi vậy, Visual Studio có thể dễ dàng phát hiện và thông báo cho bạn khi các chương trình có lỗi.

* *Hỗ trợ website*

Visual Studio code cũng hỗ trợ website, đặc biệt trong công việc soạn thảo và thiết kế web.

* *Kho tiện ích mở rộng phong phú*

Mặc dù Visual Studio có hệ thống các ngôn ngữ hỗ trợ lập trình khá đa dạng. Nhưng nếu lập trình viên muốn sử dụng một ngôn ngữ khác, bạn có thể dễ dàng tải xuống các tiện ích mở rộng. Tính năng hấp dẫn này được hoạt động như một phần chương trình độc lập nên không lo làm giảm hiệu năng của phần mềm.

* *Lưu trữ phân cấp*

Phần lớn các tệp dữ liệu đoạn mã của Visual Studio đều được đặt trong các thư mục tương tự nhau. Đồng thời, Visual Studio cũng cung cấp một số thư một cho các tệp đặc biệt để bạn lưu trữ an toàn, dễ tìm, dễ sử dụng hơn.

* *Kho lưu trữ an toàn*

Visual Studio, bạn có thể hoàn toàn yên tâm về tính lưu trữ, bởi phần mềm đã được kết nối GIT và một số kho lưu trữ an toàn được sử dụng phổ biến hiện nay.

* *Màn hình đa nhiệm*

Visual Studio sở hữu tính năng màn hình đa nhiệm, cho phép người dùng mở cùng lúc nhiều tập tin, thư mục dù chúng có thể không liên quan tới nhau.

* *Hỗ trợ viết code*

Khi sử dụng code vào trong lập trình, với Visual Studio, công cụ này có thể đề xuất tới các lập trình viên một số tùy chọn thay thế nhằm điều chỉnh đôi chút để đoạn code áp dụng thuận tiện hơn cho người dùng.

* *Hỗ trợ thiết bị đầu cuối*

Phần mềm Visual Studio cũng tích hợp các loại thiết bị đầu cuối, giúp người dùng không cần chuyển đổi giữa hai màn hình hay trở về thư mục gốc khi thực hiện một thao tác cần thiết nào đó.

* *Hỗ trợ Git*

Do kết nối với GitHub nên Visual Studio cho phép hỗ trợ sao chép, kéo thả trực tiếp. Các mã code này sau đó cũng có thể thay đổi và lưu lại trên phần mềm*.*

* *Intellisense*

Tính năng nhắc Intellisense được sử dụng hầu hết trong các phần mềm lập trình, bao gồm cả Visual Studio. Tuy nhiên, so với các trình viết mã, Visual Studio vẫn được đánh giá cao về tính chuyên nghiệp. Đặc biệt, tính năng này còn có thể phát hiện tất cả các đoạn mã không đầy đủ, nhắc lập trình viên, gợi ý sửa đổi, khai báo biến tự động trong trường hợp lập trình viên quên, giúp bổ sung cú pháp còn thiếu,…

* *Tính năng comment*

Một tính năng cũng khá hay ho, hỗ trợ cho người lập trình trong trường hợp “nhớ nhớ quên quên” đó là tính năng bình luận. Tính năng này cho phép lập trình viên để lại nhận xét, giúp dễ dàng ghi nhớ công việc cần hoàn thành, không bỏ sót công đoạn nào.

1. **SQL Server**
2. ***Khái niệm***

SQL Server hay còn gọi là Microsoft SQL Server, viết tắt là MS SQL Server. Đây là một phần mềm được phát triển bởi Microsoft dùng để lưu trữ dữ liệu dựa trên chuẩn RDBMS, và nó cũng là một hệ quản trị cơ sở dữ liệu quan hệ đối tượng (ORDBMS).

SQL Server cung cấp đầy đủ công cụ để quản lý, từ giao diện GUI cho đến việc sử dụng ngôn ngữ truy vấn SQL. Ngoài ra điểm mạnh của nó là Microsoft có khá nhiền nền tảng kết hợp hoàn hảo với SQL Server như ASP.NET, C# xây dựng Winform, bởi vì nó hoạt động hoàn toàn độc lập.

1. ***Thành phần***

Do hoạt động theo mô hình Client – Server nên nó được chia làm hai thành phần chính:

* Workstation: Được cài trên các thiết bị vận hành, nó là các phần mềm tương tác với máy chủ server. Ví dụ: SSMS, SSCM, Profiler, BIDS, SQLEM.
* Server: Được cài trên máy chủ chính (máy chủ tập trung). Đó là các dịch vụ như: SQL Server, SQL Server Agent, SSIS, SSAS, SSRS, SQL Browser, SQL Full Text Search.

1. ***Mục đích sử dụng***

Chung quy lại chúng ta sử dụng bất kì một hệ quản trị CSDL nào cũng để lưu trữ dữ liệu và SQL cũng không ngoại lệ. Tuy nhiên điều mà ta mong đợi ở nó là các tính năng giúp việc sử dụng hiêu quả hơn như:

* Cho phép tạo nhiều cơ sở dữ liệu
* Duy trì lưu trữ bền vững
* Bảo mật cao
* Phân tích dữ liệu bằng SSAS – SQL Server Analysis Services
* Tạo được báo cáo bằng SSRS – SQL Server Reporting Services
* Thực hiện quá trình ETL (Extract-Transform-Load) bằng SSIS – SQL Server Integration Services.

1. ***Phân loại***

* **SQL DDL (Data Definition Language)**

DDL là ngôn ngữ định nghĩa dữ liệu. Các lệnh DDL bao gồm:

Lệnh CREATE: Tạo một bảng, một View của bảng, hoặc đối tượng khác trong Database.

Lệnh ALTER: Sửa đổi một đối tượng Database đang tồn tại, ví dụ như một bảng.

Lệnh DROP: Xóa toàn bộ một bảng, một View của bảng hoặc đối tượng khác trong một Database.

* **SQL DML (Data Manipulation Language**)

DML là ngôn ngữ thao tác dữ liệu. Các lệnh DML bao gồm:

Lệnh SELECT: Lấy các bản ghi cụ thể từ một hoặc nhiều bảng.

Lệnh INSERT: Tạo một bản ghi.

Lệnh UPDATE: Sửa đổi các bản ghi.

Lệnh DELETE: Xóa các bản ghi.

* **SQL DCL (Data Control Language)**

DCL là ngôn ngữ điều khiển dữ liệu. Các lệnh DCL bao gồm:

Lệnh GRANT: Trao một quyền tới người dùng.

Lệnh REVOKE: Thu hồi quyền đã trao cho người dùng.

SQL ngày càng đóng vai trò quan trọng khi mà hiện nay Internet ngày càng phát triển. SQL thường được sử dụng để tạo ra các trang web động và các trang web này thường có nội dung được lấy ra từ cơ sở dữ liệu.

SQL có thể được dùng như sự kết hợp giữa cơ sở dữ liệu và trang web. Trong cơ sở dữ liệu trên máy chủ SQL sẽ thực hiện việc truy cập thông tin và kết quả hiển thị trên trang web khi người dùng yêu cầu. Và SQL cũng là công cụ để cập nhật thông tin cho CSDL đó.

1. ***Các phiên bản***

* Enterprise – là phiên bản cao cấp, có đầy đủ các tính năng
* Standard – gói chuẩn, tính năng tương đối đầy đủ, tuy nhiên vẫn kém hơn so với Enterprise
* Workgroup – dành cho các công ty làm việc từ xa với nhiều máy tách biệt
* Web – dành cho các ứng dụng website
* Developer – dành cho nhà phát triển, chứa đầy đủ tính năng nhưng chỉ phân quyền cho một người duy nhất, dễ dàng nâng cấp lên bản Enterprise mà không cần phải cài đặt lại từ đầu.
* Express – là bản cơ bản, sử dụng tối đa 1 CPU và 1GM RAM, dung lượng lưu trữ tối đa là 10GB. Bản này thường được sử dụng khi bạn học SQL Server ở trường.
* Compact – nhúng miễn phí vào các môi trường phát triển ứng dụng web. Kích thước tối đa của cơ sở dữ liệu là 4GB.
* Datacenter – là bản không giới hạn về bộ nhớ và hỗ trợ 25 bản cài đặt.
* Business Intelligence – được giới thiệu ở phiên bản SQL Server 2012, có đầy đủ các tính năng của bản Standard và một số tính năng như: Power View và PowerPivot.
* Enterprise Evaluation – bản dành để học tập và thử nghiệm phần mềm, chỉ được dùng trong 6 tháng.

1. **Devexpress**
2. ***Khái niệm***

Đối với những lập trình viên .NET thì DevEpress là một công cụ hết sức hữu dụng, cung cấp rất nhiều control trong Visual Studio. DevExpress không chỉ giúp thiết kế winform hay website đẹp hơn mà còn giúp cho việc lập trình được dễ dàng hơn, ta có thể thấy rõ nhất là trong việc tương tác với cơ sở dữ liệu.

DevExpress được ra mắt lần đầu tiên vào năm 2011 và được đông đảo lập trình viên .NET sử dụng. Từ đó đến này đã trải qua rất nhiều phiên bản với nhiều nâng cấp đáng kể. Phiên bản mới nhấtt hiện nay là v18.2. Với DevExpress ta có thể tự tạo cho mình một bộ Office riêng chỉ trong vòng một vài tiếng.

1. ***Thành phần***

* WinForms Controls: Cung cấp các control cho WinForms.
* ASP.NET Controls: Cung cấp các control cho WebForms.
* WPF Controls: Cung cấp các control cho WPF.
* Silverlight Controls: Cung cấp các control cho Silverlight.
* XtraCharts: Control cung cấp các loại biểu đồ.
* XtraReports: Cung cấp các control tạo báo cáo.
* XPO: Cung cấp môi trường làm việc với database.
* XAF: Một công nghệ mới giúp việc phát triển phần mềm một cách nhanh chóng.

1. ***Cài đặt***

Yêu cầu hệ thống:

* Microsoft .NET Framework từ 2.0 trở lên
* Microsoft Visual Studio
* SQL Server

1. **EntityFramework**
2. ***Khái niệm***

Entity Framework là một khung ORM(Object Relational Mapper) mã nguồn mở cho các ứng dụng .NET được Microsoft hỗ trợ. là 1 phần của .NET Framework. là một Open source ORM Framework.

*ORM là một công cụ tự động tạo ra các Objects trong ứng dụng từ tables,views, store procedures của một Relational Database.*

1. ***Lịch sử ra đời***

Entity Framework được cho ra mắt vào năm 2008. Nhờ sự ra đợi của nó việc truy vấn cơ sở dữ liệu quan hệ trở lên đơn giản, dễ học, giúp giảm bớt thời gian phát triển dự án xuống đáng kể.

Lịch sử phiên bản EF 6

EF 6 2013 .NET 4.0 & .NET 4.5, VS 2012

EF 5 2012 .NET 4.0, VS 2012

EF 4.3 2011 .NET 4.0, VS 2012

EF 4.0 2010 .NET 4.0, VS 2010

EF 1.0 (hoặc 3.5) 2008 .NET 3.5 SP1, VS 2008

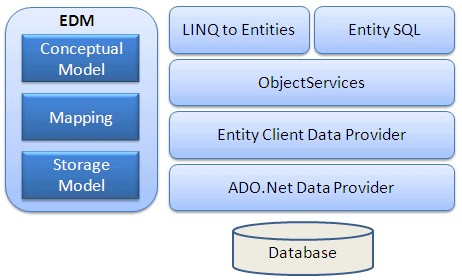
Lịch sử phiên bản lõi(core) của EF

EF Core 2.0 tháng 8 năm 2017 .NET Core 2.0, VS 2017

Lõi EF 1.1 Tháng 11 năm 2016 .NET Lõi 1.1

EF Core 1.0 tháng 6 năm 2016 .NET Core 1.0

1. ***Cấu trúc***



Hình 1.2: Cấu trúc entity framework

* **EDM (Entity Data Model):**

EDM bao gồm 3 phần :

Conceptual Model: Chứa các model class và các quan hệ của nó. Nó độc lập với thiết kế bảng CSDL của bạn.

Mapping: gồm có thông tin về cách Conceptual model kết nối với Storage model.

Storage Model: Đây là database design model bao gồm các bảng, views, stored procedures, và những quan hệ và các khóa.

* **LINQ to Entities:**

Là ngôn ngữ truy vấn sử dụng để viết các truy vấn tới object model. Trả về các thực thể được định nghĩa bên trong Conceptual model.

* **Entity SQL:**

Đây là một ngôn ngữ truy vấn khác, nó giống LINQ to Entities.

* **Object Service:**

Object service sẽ có trách nhiệm trong việc cụ thể hóa quá trình chuyển đổi dữ liệu trả về từ một entity client data provider tới một entity object structure.

* **Entity Client Data Provider:**

Nhiệm vụ chính của nó là chuyển đổi L2E hoặc những truy vấn Entity SQL vào một truy vấn SQL. Nó được hiểu bởi CSDL cơ bản. Giao tiếp với ADO.Net data provider lần lượt gửi và nhận dữ liệu từ CSDL.

* **ADO.Net Data Provider:**

*Thực hiện giao tiếp với CSDL bằng sử dụng chuẩn ADO.Net.*

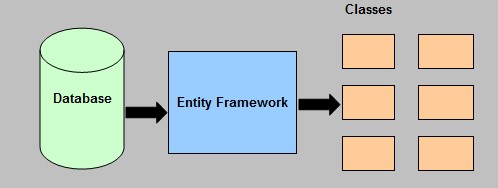
1. ***Tính năng***

Những tính năng đáng chú ý nhất của Entity framework:

* Entity framework hỗ trợ stored procedure.
* Nó làm việc với bất kỳ cơ sở dữ liệu quan hệ nào có Entity Framework hợp lệ.
* Entity Framework tạo truy vấn SQL từ LINQ to Entities.
* Nó được phát triển như một sản phẩm mã nguồn mở.
* Entity framework là một sản phẩm của Microsoft.
* EF cho phép tạo những câu lệnh thêm, xóa, cập nhật.
* Theo dõi những thay đổi của các đối tượng trong bộ nhớ.

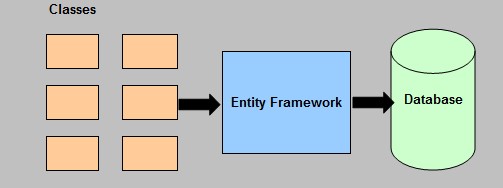
1. ***Các tình huống dùng Entity Framework***

**Database First:** Đầu tiên là nếu bạn đã có một CSDL hoặc muốn thiết kế CSDL trước khi làm các phần khác của ứng dụng.



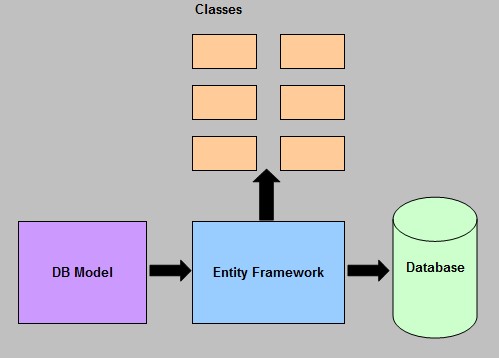
Hình 1.3: Sơ đồ database first

**Code First:** Thứ hai là bạn muốn tập trung vào các domain class rồi mới tạo CSDL từ các domain class đó.



Hình 1.4: Sơ đồ code First entity

**Model First:** Thứ ba là bạn muốn thiết kế schema của CSDL trên visual designer rồi mới tạo CSDL và các class.

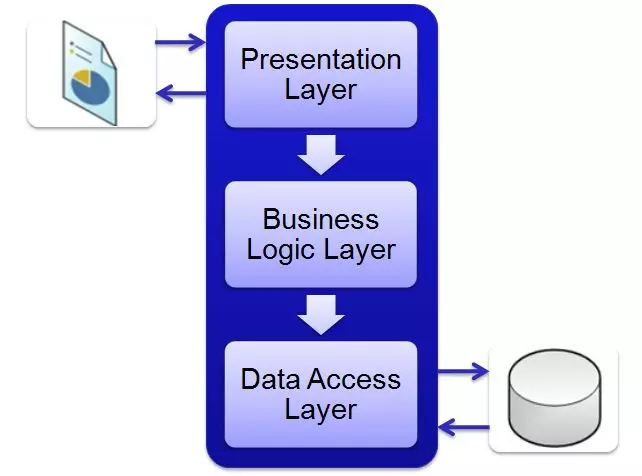


Hình 1.5: Sơ đồ Model First

1. **Mô hình 3 lớp**
2. ***Khái niệm***

Mô hình 3 lớp hay còn được gọi là mô hình Three Layer(3-Layer), mô hình này ra đời nhằm phân chia các thành phần trong hệ thống, các thành phần cùng chức năng sẽ được nhóm lại với nhau và phân chia công việc cho từng nhóm để dữ liệu không bị chồng chéo và chạy lộn xộn.

Mô hình này phát huy hiệu quả nhất khi bạn xây dựng một hệ thống lớn, việc quản lý code và xử lý dữ liệu lỗi dễ dàng hơn.



Hình 1.6: Minh họa mô hình 3 lớp

Ưu điểm:

* Việc phân chia thành từng lớp giúp cho code được tường minh hơn. Nhờ vào việc chia ra từng lớp đảm nhận các chức năng khác nhau và riêng biệt như giao diện, xử lý, truy vấn thay vì để tất cả lại một chỗ. Nhằm giảm sự kết dính.
* Dễ bảo trì khi được phân chia, thì một thành phần của hệ thống sẽ dễ thay đổi. Việc thay đổi này có thể được cô lập trong 1 lớp, hoặc ảnh hưởng đến lớp gần nhất mà không ảnh hưởng đến cả chương trình.
* Dễ phát triển, tái sử dụng: khi chúng ta muốn thêm một chức năng nào đó thì việc lập trình theo một mô hình sẽ dễ dàng hơn vì chúng ta đã có chuẩn để tuân theo. Và việc sử dụng lại khi có sự thay đổi giữa hai môi trường ( Winform sang Webfrom ) thì chỉ việc thay đổi lại lớp GUI.
* Dễ bàn giao. Nếu mọi người đều theo một quy chuẩn đã được định sẵn, thì công việc bàn giao, tương tác với nhau sẽ dễ dàng hơn và tiết kiệm được nhiều thời gian.
* Dễ phân phối khối lượng công việc. Mỗi một nhóm, một bộ phận sẽ nhận một nhiệm vụ trong mô hình 3 lớp. Việc phân chia rõ ràng như thế sẽ giúp các lập trình viên kiểm soát được khối lượng công việc của mình.

Lưu ý khi xây dựng mô hình 3 lớp:

* Cần một solution riêng cho project.
* Cần 3 project khác nhau để làm nên 3 lớp, tên Project đặt như sau:
* Lớp GUI: (VD: QuanLy\_GUI)
* Lớp Business: (VD: QuanLy\_BUS)
* Lớp Data Access: (VD: QuanLy\_DAL)
* Lớp DTO: (VD: QuanLy\_DTO)

1. ***Các thành phần chính của mô hình 3 lớp***

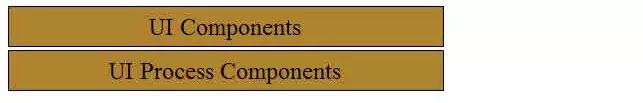
Mô hình 3-layer gồm có 3 phần chính:

* ***Presentation Layer (GUI)***

Lớp này có nhiệm vụ chính là giao tiếp với người dùng. Nó gồm các thành phần giao diện ( winform, webform, …) và thực hiện các công việc như nhập liệu, hiển thị dữ liệu, kiểm tra tính đúng đắn dữ liệu trước khi gọi lớp Business Logic Layer (BLL).

Có hai thành phần chính sau đây với những tác vụ cụ thể :

* UI Components : gồm các thành phần tạo nên giao diện của ứng dụng (GUI). Chúng chịu trách nhiệm thu nhận và hiển thị dữ liệu cho người dùng… Ví dụ : textbox, button, combobox, …
* UI Process Components : là thành phần chịu trách nhiệm quản lý các quá trình chuyển đổi giữa các UI… Ví dụ : Sắp xếp quá trình kiểm tra thông tin khách hàng:
* Hiển thị màn hình tra cứu ID.
* Hiển thị màn hình thông tin chi tiết khách hàng tương ứng.
* Hiển thị màn hình liên lạc với khách hàng.

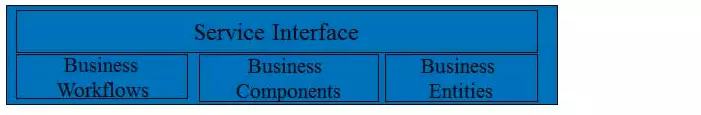


Hình 1.7: Tầng presentation layer

* ***Business Logic Layer (BLL)***

Đây là nơi đáp ứng các yêu cầu thao tác dữ liệu của GUI layer, xử lý chính nguồn dữ liệu từ Presentation Layer trước khi truyền xuống Data Access Layer và lưu xuống hệ quản trị CSDL.

Đây còn là nơi kiểm tra các ràng buộc, tính toàn vẹn và hợp lệ dữ liệu, thực hiện tính toán và xử lý các yêu cầu nghiệp vụ, trước khi trả kết quả về Presentation Layer.



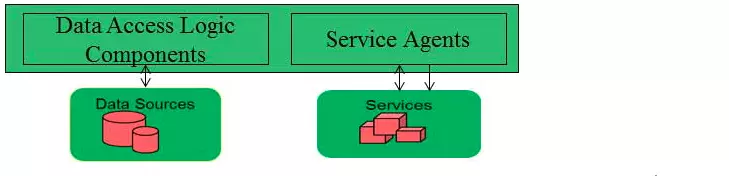
Hình 1.8: Tầng Bus logic layer

Lớp này gồm 4 thành phần:

* Service Interface : là thành phần giao diện lập trình mà lớp này cung cấp cho lớp Presentation sử dụng.
* Bussiness Workflows : chịu trách nhiệm xác định và điều phối các quy trình nghiệp vụ gồm nhiều bước và kéo dài. Những quy trình này phải được sắp xếp và thực hiện theo một thứ tự chính xác.
* Bussiness Components : chịu trách nhiệm kiểm tra các quy tắc nghiệp vụ, ràng buộc logic và thực hiện các công việc . Các thành phần này cũng thực hiện các dịch vụ mà Service Interface cung cấp và Business Workflows sẽ sử dụng nó.
* Bussiness Entities : thường được sử dụng như Data Transfer Objects ( DTO ) . Có thể sử dụng để truyền dữ liệu giữa các lớp (Presentation và Data Layer). Chúng thường là cấu trúc dữ liệu ( DataSets, XML,… ) hay các lớp đối tượng đã được tùy chỉnh. Ví dụ : tạo 1 class Student lưu trữ các dữ liệu về tên, ngày sinh, ID, lớp.
* ***Data Access Layer (DAL)***

Lớp này có chức năng giao tiếp với hệ quản trị CSDL như thực hiện các công việc liên quan đến lưu trữ và truy vấn dữ liệu ( tìm kiếm, thêm, xóa, sửa,…).

Data Access Logic Components : chịu trách nhiệm chính lưu trữ và truy xuất dữ liệu từ các nguồn dữ liệu (Data Sources) như XML, file system,… Hơn nữa còn tạo thuận lợi cho việc dễ cấu hình và bảo trì. Service Agents : giúp bạn gọi và tương tác với các dịch vụ từ bên ngoài một cách dễ dàng và đơn giản*.*

**

Hình 1.9: Tầng Access layer

* ***DTO Layer***

Lớp này chỉ là phụ thôi, đây là lớp định nghĩa các table trong database của bạn, định nghĩa cột của nó cũng như để ta gán data khi query lấy dữ liệu. Các bạn có thể hiểu nôm na là 1 dạng cơ bản ORM (Object Relation Mapping).

Để đảm bảo sự thống nhất bạn nên đặt tên theo nguyên tắc. Dưới đây là một số cách đặt tên cho các thành phần của 3 lớp, Ví dụ:

Một là, đặt lần lượt tên GUI - BUS - DAL.

Hai là, sử dụng tên có chứa tiền tố GUI - BLL - DAL - DTO.

Ba là, áp dụng tên có chứa GUI - BLL - DAL.

# **CHƯƠNG 2: PHÂN TÍCH THIẾT KỆ HỆ THỐ****NG**

1. **Kiến trúc hệ thống**

Ứng dụng kiến trúc mô hình 3 lớp để thiết kế phần mềm quản lý học sinh.

Kiến trúc này gồm có 3 thành phần: Presentation Layers, Business Logic Layers, và Data Access Layers.

* Presentation Layers: Lớp này làm nhiệm vụ giao tiếp với người dùng cuối để thu thập dữ liệu và hiển thị kết quả/dữ liệu thông qua các thành phần trong giao diện người sử dụng.
* Busines Logic Layers: Đây là layer xử lý chính các dữ liệu trước khi được đưa lên hiển thị trên màn hình hoặc xử lý các dữ liệu trước khi chuyển xuống Data Access Layer để lưu dữ liệu xuống cơ sở dữ liệu. Đây cũng là nơi đê kiểm tra ràng buộc, các yêu cầu nghiệp vụ, tính toán, xử lý các yêu cầu và lựa chọn kết quả trả về cho Presentation Layers.
* Data Access Layers: Lớp này thực hiện các nghiệp vụ liên quan đến lưu trữ và truy xuất dữ liệu của ứng dụng như đọc, lưu, cập nhật cơ sở dữ liệu.

**Cách thức vận hành như sau:**

* Đầu tiên User giao tiếp với Presentation Layers (GUI) để gửi đi thông tin và yêu cầu. Tại layer này, các thông tin sẽ được kiểm tra, nếu OK chúng sẽ được chuyển xuống Business Logic Layer (BUS).
* Tại BUS, các thông tin sẽ được nhào nặn, tính toán theo đúng yêu cầu đã gửi, nếu không cần đến Database thì BUS sẽ gửi trả kết quả về GUI, ngược lại nó sẽ đẩy dữ liệu (thông tin đã xử lý) xuống Data Access Layer (DAO).
* DAO sẽ thao tác với Database và trả kết quả về cho BUS, BUS kiểm tra và gửi nó lên GUI để hiển thị cho người dùng.
* Một khi gặp lỗi (các trường hợp không đúng dữ liệu) thì đang ở layer nào thì quăng lên trên layer cao hơn nó 1 bậc cho tới GUI thì sẽ quăng ra cho người dùng biết - Các dữ liệu được trung chuyển giữa các Layer thông qua một đối tượng gọi là Data Transfer Object (DTO), đơn giản đây chỉ là các Class đại diện cho các đối tượng được lưu trữ trong Database.

1. **Mô tả các thành phần trong hệ thống.**

|  |  |  |
| --- | --- | --- |
| **STT** | **Thành phần** | **Thành phần chi tiết** |
| 1 | GUI | Chuyên xử lý về giao diện, nếu thành công sẽ gửi thông tin yêu cầu xuống BUS (business logic layer).  Chứa các thư mục chính như: HelpGUI, Resource,fileReport (ReportBCTKTraTre, ReportMuonTL) và các form (frmADMIN, frmBCTKSachMuonTL, frmBCTKTraTre, frmChiTietPM, frmDocGia, frmDSPhieuPhat, frmLoaiDocGia, frmLogin, frmMain, frmSach, frmTacGia, frmThayDoiQuyDinh, frmTheLoai, frmThemDocGia, frmThemLoaiDocGia, frmThemPhieuMuon, frmThemSach, frmThemTacGia, frmThemTheLoai, frmThemTraSach, frmTTCaNhan, frmTTMuonSach, frmTTTraSach) |
| 2 | BUS | Thông tin sẽ được xử lý, nếu không cần đến Database thì sẽ gửi ngược lại GUI, ngược lại sẽ đẩy giữ liệu xuống DAO.  Chứa các lớp: ACCOUNTBUS, ADMINBUS, BCTKMUONSACHBUS, BCTKTRATREBUS, CHITIETPHANQUYENBUS, CHITIETPHIEUMUONBUS, CHITIETPHIEUTRABUS, CHITIETSACHBUS, CTBCTKMUONSACHBUS, CTBCTKTRATREBUS, CTPHIEUMUONBUS, DOCGIABUS, DSPHIEUPHATBUS, LOAIDOCGIABUS, PERMISSIONBUS, PHIEUMUONSACHBUS, SACHBUS, THAMSOBUS, THELOAIBUS, TACGIABUS, TINHTRANGBUS, TTDOCGIABUS. |
| 3 | DAO | DAO sẽ thao tác với Database và trả kết quả về cho BUS, BUS kiểm tra và gửi nó lên GUI để hiển thị cho người dùng.  Trong đó có chứa các lớp: ACCOUNTDAO, ADMINDAO, BCTKMUONSACHDAO, BCTKTRATREDAO, CHITIETPHANQUYENDAO, CHITIETPHIEUMUONDAO, CHITIETPHIEUTRADAO, CHITIETSACHDAO, CTBCTKMUONSACHDAO, CTBCTKTRATREDAO, CTPHIEUMUONDAO, DOCGIADAO, DSPHIEUPHATDAO, LOAIDOCGIADAO, PERMISSIONDAO, PHIEUMUONSACHDAO, SACHDAO, THAMSODAO, THELOAIDAO, TACGIADAO, TINHTRANGDAO, TTDOCGIADAO. |
| 4 | DTO | Bao gồm các lớp được sử dụng trong phần mềm: model.edmx (ACCOUNT, ADMIN, BCTKMUONSACH, BCTKTRATRE, CHITIETPHANQUUYEN, CTBCTKMUONSACH, CTBCTKTRATRE, CTPHIEUMUON, DOCGIA, LOAIDOCGIA, PERMISSION, PHIEUMUONSACH, PHIEUTHUTIENPHAT, PHIEUTRASACH, SACH, TACGIA, THAMSO, THELOAI, TINHTRANG) được tạo nên nhờ Entity Framework, một số lớp DTO tự tạo (TTDOCGIADTO, CHITIETPHIEUMUONDTO, CHITIETPHIEUTRADTO, CHITIETSACHDTO) |

1. **Thuật toán lập sơ đồ logic:**
2. ***Bước 1: Xét yêu cầu phần mềm “Lập thẻ độc giả”***

a. Thiết kế dữ liệu với tính đúng đắn

* Các thuộc tính mới: **HoTen, MaDocGia, NgaySinh, DiaChi, Email, NgayLapThe, NgayHetHan, IDAccount, SDT,TongNo**
* Thiết kế dữ liệu:



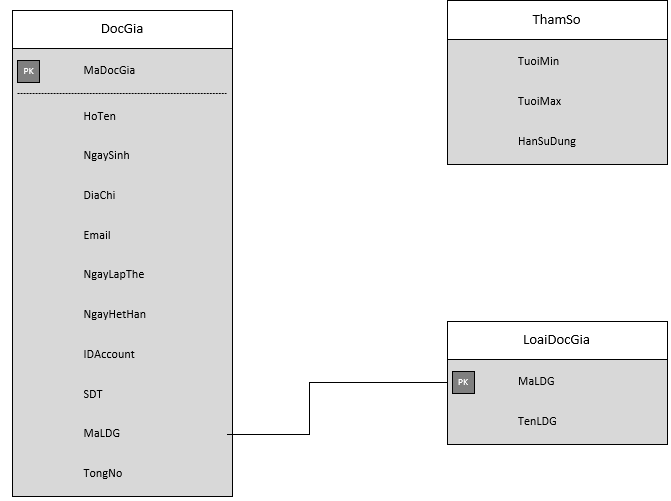
Hình 2.1: Bảng độc giả

* Các thuộc tính trừu tượng : **MaDocGia**
* Sơ đồ logic:



b.Thiết kế dữ liệu với tính tiến hóa

* Các thuộc tính mới: **MaLDG, TuoiMax, TuoiMin, HanSuDung**
* Thiết kế dữ liệu:



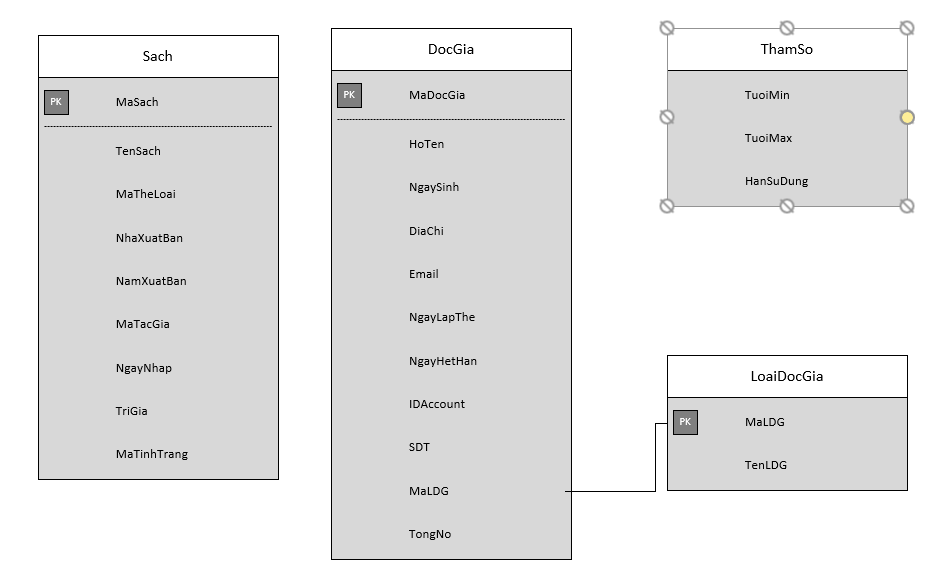
* Các thuộc tính trừu tượng: **MaLoaiDocGia, MaDocGia**
* Sơ đồ logic



1. ***Bước 2: Xét yêu cầu phần mềm “Tiếp nhận sách mới”***

a. Thiết kế dữ liệu với tính đúng đắn

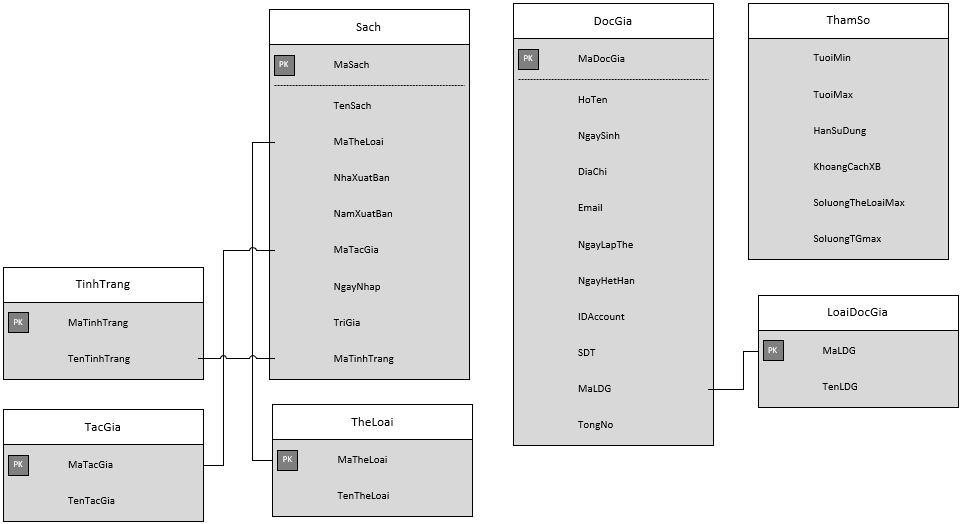
* Các thuộc tính mới: **TenSach, TheLoai, TacGia, NamXB, NhaXuatBan, NgayNhap, TriGia, MaTinhTrang**
* Thiết kế dữ liệu:



* Các thuộc tính trừu tượng : **MaSach**

b.Thiết kế dữ liệu với tính tiến hóa

* Các thuộc tính mới: **TenTheLoai, TenTacGia, KhoangCachXB, SoLuongTheLoaiMax, SoLuongTGmax**
* Thiết kế dữ liệu:



Hình 2.2: Sơ đồ cơ sở dữ liệu tiếp nhận sách mới

* Các thuộc tính trừu tượng: **MaTheLoai, MaTacGia, MaTinhTrang**

1. ***Bước 3: Xét yêu cầu phần mềm “Tra cứu sách”***

a. Thiết kế dữ liệu với tính đúng đắn

* Các thuộc tính mới: (**Không Có)**
* Thiết kế dữ liệu: (**Không Đổi)**
* Các thuộc tính trừu tượng : (**Không Có)**
* Sơ đồ logic: (**Không Đổi)**

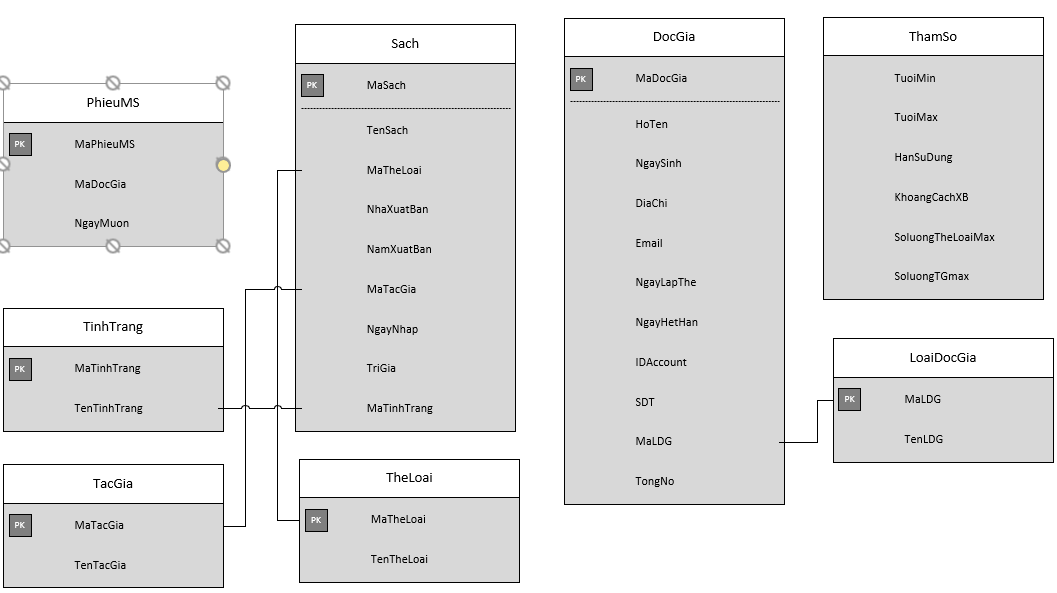
b.Thiết kế dữ liệu với tính tiến hóa

* Biểu mẫu liên quan: (**Không Có)**
* Các thuộc tính mới: (**Không Có)**
* Thiết kế dữ liệu: (**Không Đổi)**
* Các thuộc tính trừu tượng: (**Không Có)**
* Sơ đồ logic (**Không Đổi)**

1. ***Bước 4: Xét yêu cầu phần mềm “Cho mượn sách”***

a. Thiết kế dữ liệu với tính đúng đắn

* Các thuộc tính mới: **MaPhieuMS, MaDocGia , NgayMuon**
* Thiết kế dữ liệu:

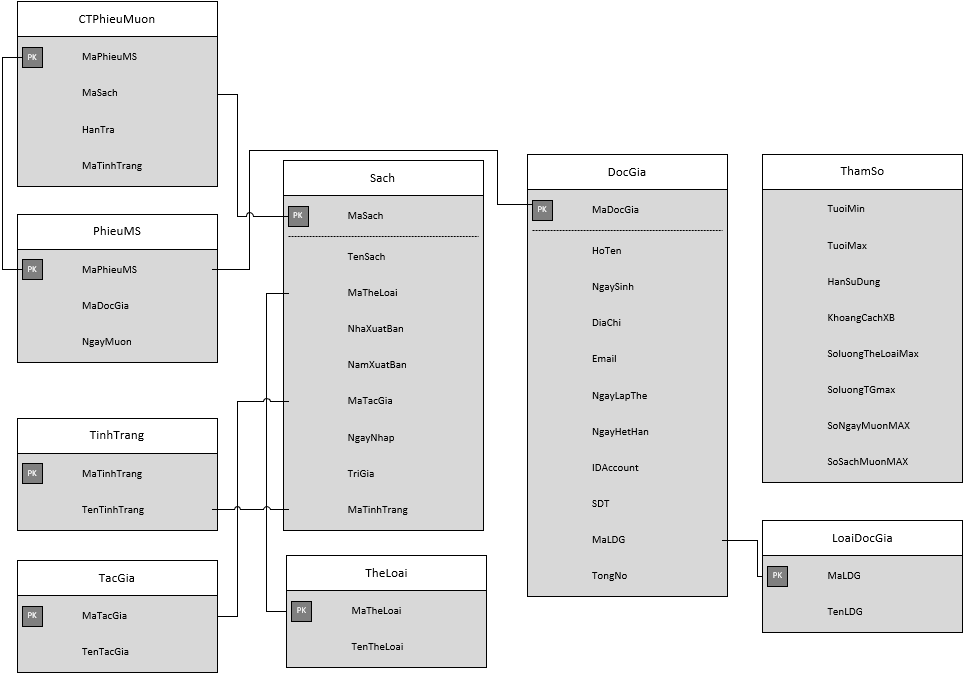


Hình 2.3: Sơ đồ cơ sở dữ liệu yêu cầu mượn sách tính đúng đắng

* Các thuộc tính trừu tượng : **MaPhieuMS**

b.Thiết kế dữ liệu với tính tiến hóa

* Các thuộc tính mới: **SoSachMuonMAX, SoNgayMuonMAX, HanTra, TinhTrangMuon**
* Thiết kế dữ liệu:



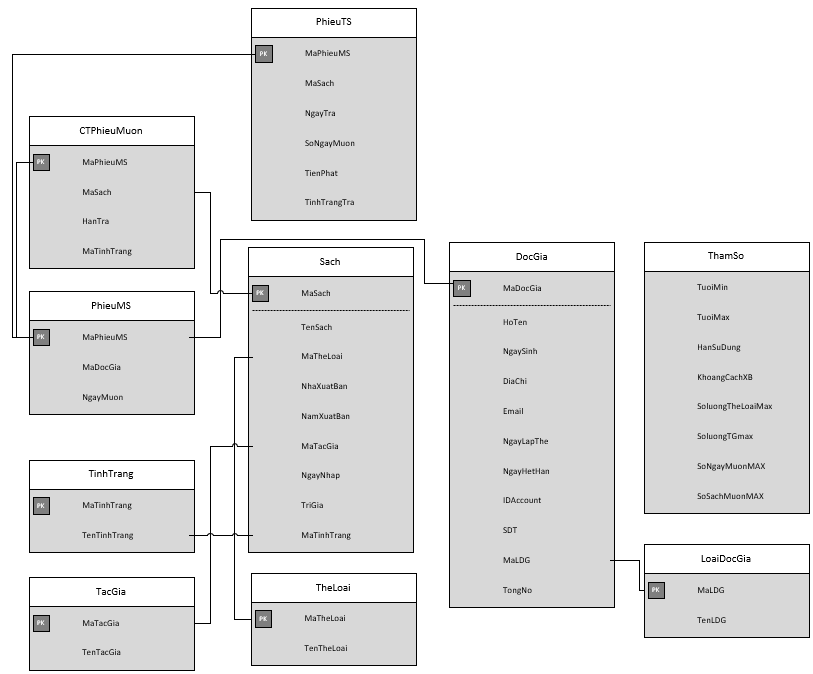
Hình 2.4: Sơ đồ cơ sở dữ liệu yêu cầu mượn sách tính tiến hóa

* Các thuộc tính trừu tượng: **MaPhieuMS**

1. ***Bước 5: Xét yêu cầu phần mềm “Nhận trả sách”***

a. Thiết kế dữ liệu với tính đúng đắn

* Các thuộc tính mới: **MaPhieuMS, MaSach, NgayTra, SoNgayMuon, TienPhat, TinhTrangTraSach**
* Thiết kế dữ liệu:

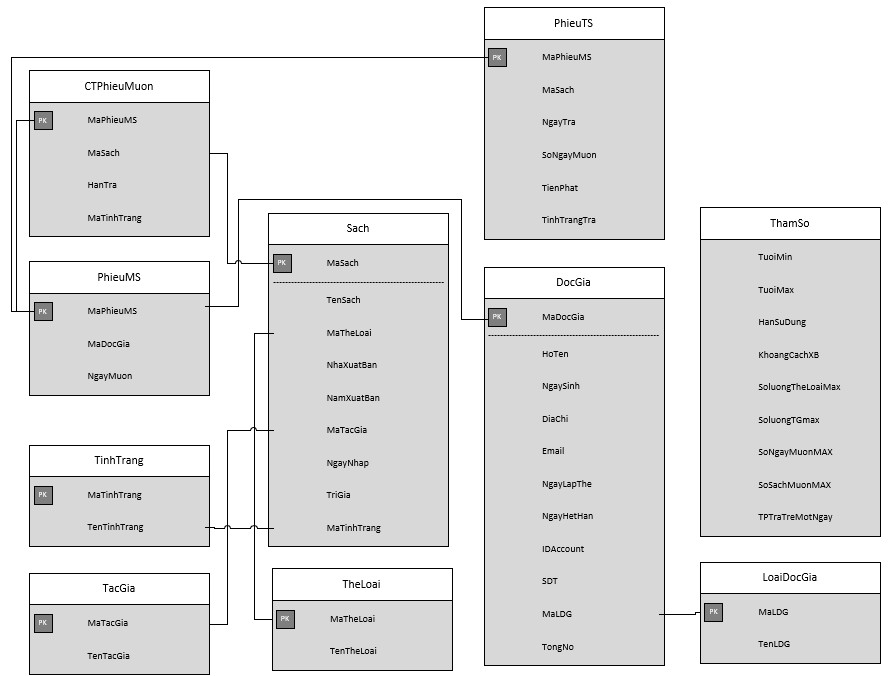


Hình 2.5: Sơ đồ cơ sở dữ liệu nhận trả sách tính đúng đắng

* Các thuộc tính trừu tượng : **MaPhieuMS**

b.Thiết kế dữ liệu với tính tiến hóa

* Các thuộc tính mới: **TPTraTreMotNgay**
* Thiết kế dữ liệu:



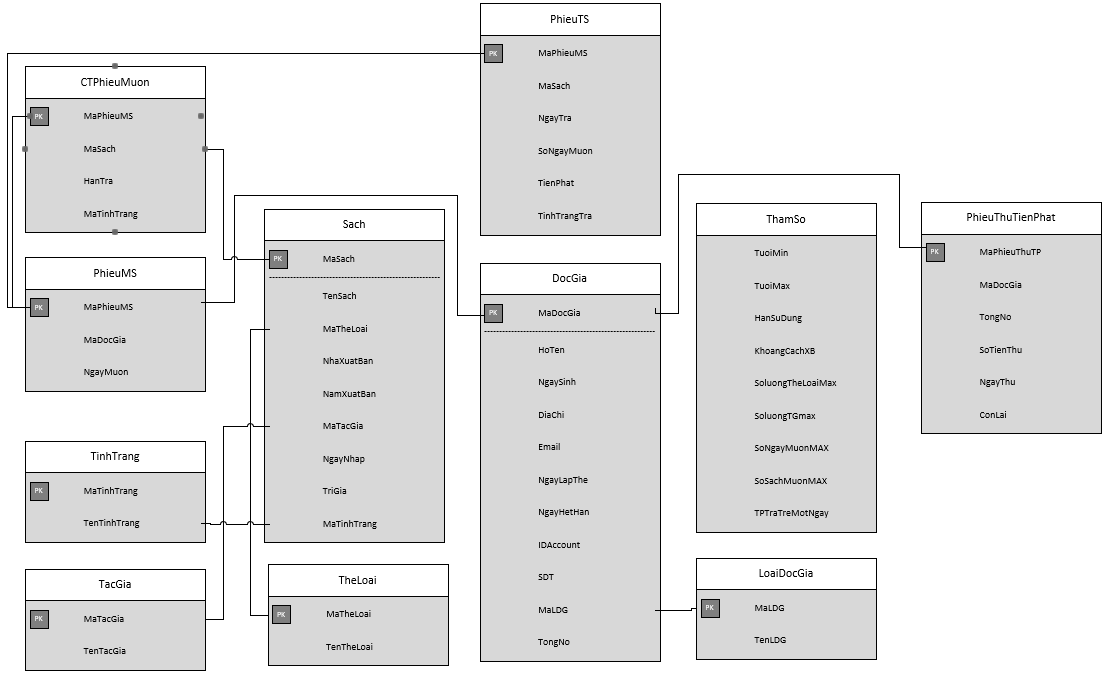
Hình 2.6: Sơ đồ cơ sở dữ liệu nhận trả sách tính tiến hóa

* Các thuộc tính trừu tượng: (**Không Có)**

1. ***Bước 6: Xét yêu cầu phần mềm “Lập phiếu thu tiền phạt”***

a. Thiết kế dữ liệu với tính đúng đắn

* Các thuộc tính mới: **MaPhieuThuTP, MaDocGia, TongNo, SoTienThu, NgayThu, ConLai**
* Thiết kế dữ liệu:



Hình 2.7: Sơ đồ cơ sở dữ liệu phiếu phạt tính đúng đắng

* Các thuộc tính trừu tượng : **MaPhieuThuTP**

b.Thiết kế dữ liệu với tính tiến hóa

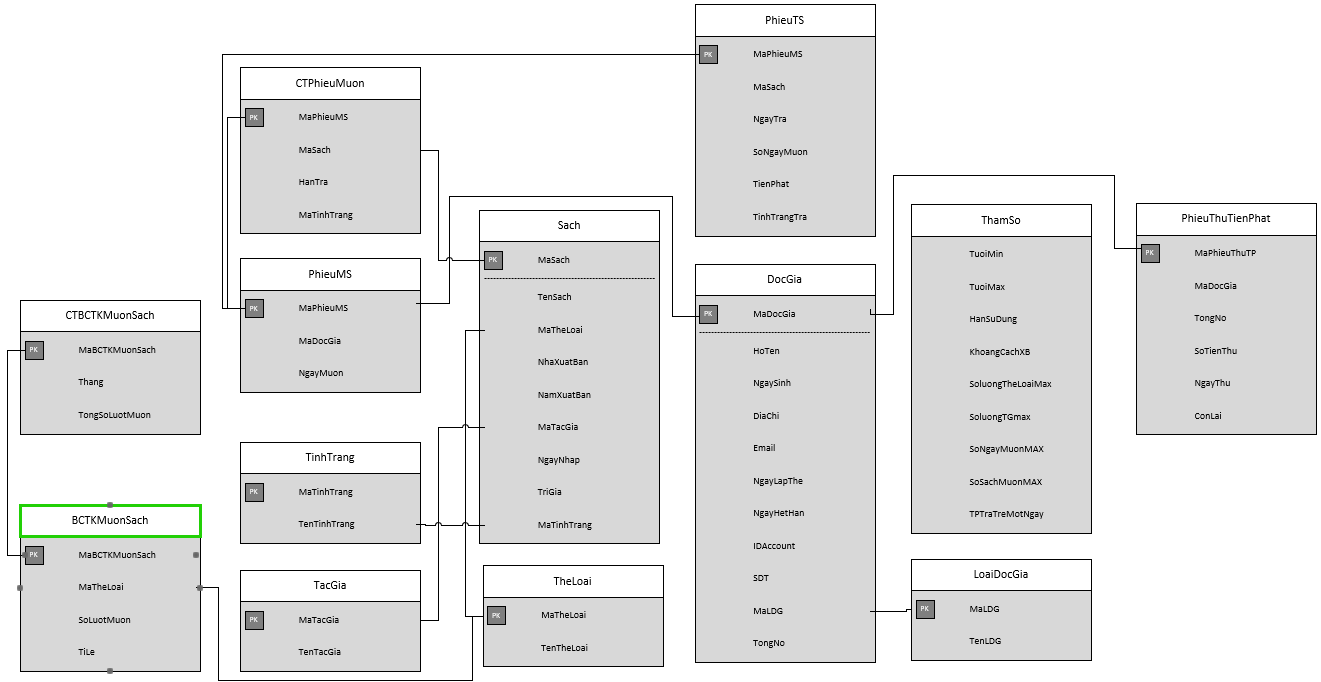
* Biểu mẫu liên quan: (**Không Có)**
* Các thuộc tính mới: (**Không Có)**
* Thiết kế dữ liệu: (**Không Đổi)**
* Các thuộc tính trừu tượng: (**Không Có)**

1. ***Bước 7: Xét yêu cầu phần mềm “Lập báo cáo”***

* **Báo cáo theo thể loại**

a. Thiết kế dữ liệu với tính đúng đắn

* Các thuộc tính mới: **MaBCTKMuonSach, MaTheLoai, SoLuotMuon, TiLe, Thang**
* Thiết kế dữ liệu:



Hình 2.8: Sơ đồ cơ sở dữ liệu báo cáo mượn sách tính đúng đắng

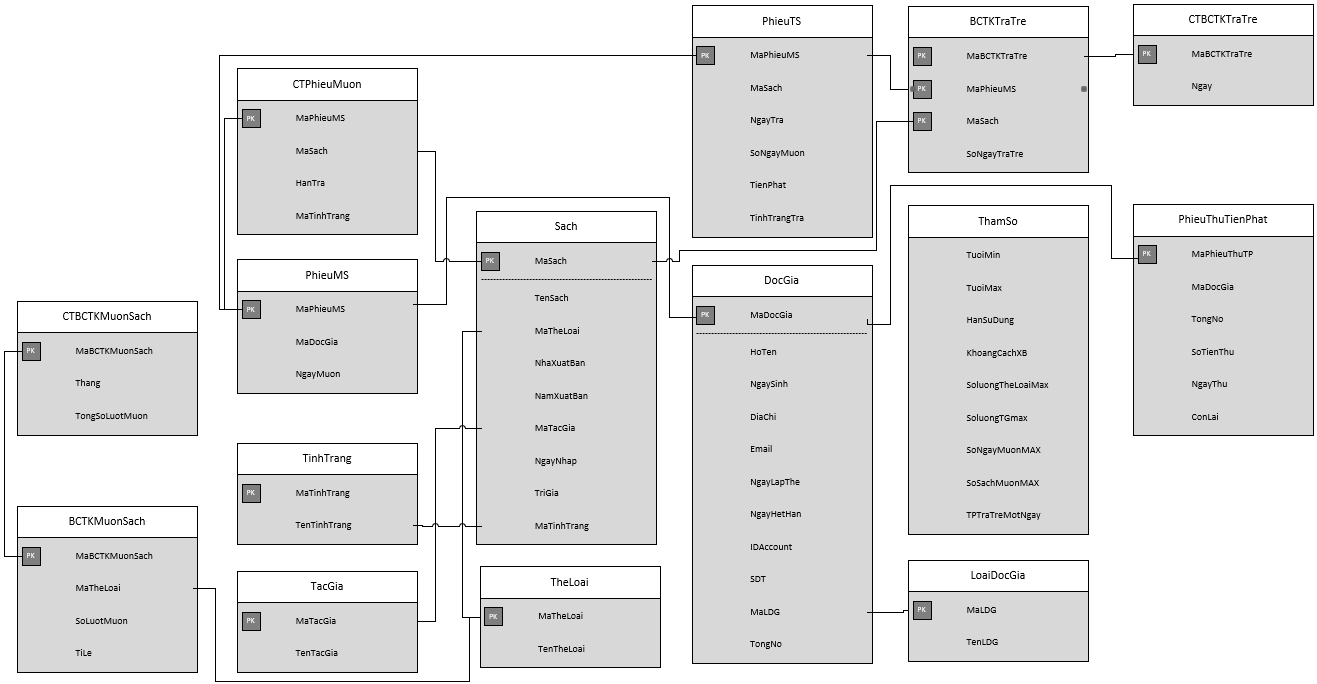
* Các thuộc tính trừu tượng : **MaBCTKMuonSach**

b.Thiết kế dữ liệu với tính tiến hóa

* Biểu mẫu liên quan: (**Không Có)**
* Sơ đồ luồng dữ liệu: (**Không Đổi)**
* Các thuộc tính mới: (**Không Có)**
* Thiết kế dữ liệu: (**Không Đổi)**
* Các thuộc tính trừu tượng: (**Không Có)**
* **Báo cáo theo sách trả trễ**

a. Thiết kế dữ liệu với tính đúng đắn

* Các thuộc tính mới: **MaBCTKTraTre, MaPhieuMS, MaSach, SoNgayTraTre, Ngay**
* Thiết kế dữ liệu:



Hình 2.9: Sơ đồ cơ sở dữ liệu sách trả trễ tính đúng đắng

* Các thuộc tính trừu tượng : **MaBCTKTraTre**

b.Thiết kế dữ liệu với tính tiến hóa

* Biểu mẫu liên quan: (**Không Có)**
* Sơ đồ luồng dữ liệu: (**Không Có)**
* Các thuộc tính mới: (**Không Có)**
* Thiết kế dữ liệu: (**Không Đổi)**
* Các thuộc tính trừu tượng: (**Không Có)**
* Sơ đồ logic (**Không Đổi)**

1. **Bước 8: Xét yêu cầu phần mềm “Thay đổi quy định”**

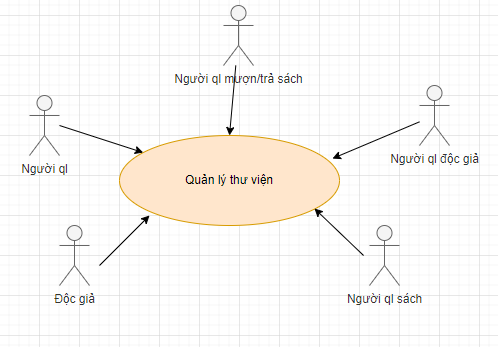
a. Thiết kế dữ liệu với tính đúng đắn

* Các thuộc tính mới: (**Không Có)**
* Thiết kế dữ liệu: (**Không Đổi)**
* Các thuộc tính trừu tượng : (**Không Có)**
* Sơ đồ logic: (**Không Đổi)**

b.Thiết kế dữ liệu với tính tiến hóa

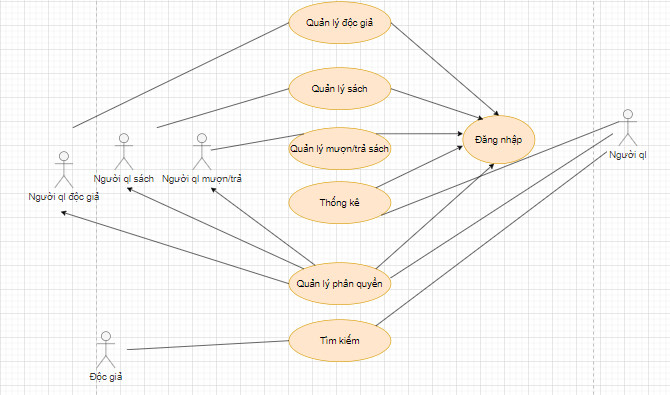
* Biểu mẫu liên quan: (**Không Có)**
* Các thuộc tính mới: (**Không Có)**
* Thiết kế dữ liệu: (**Không Đổi)**
* Các thuộc tính trừu tượng: (**Không Có)**
* Sơ đồ logic (**Không Đổi)**

1. **Phân thích một số quy trình trong thiết kế hệ thống**
2. ***Mô hình usecase mức tổng quát***



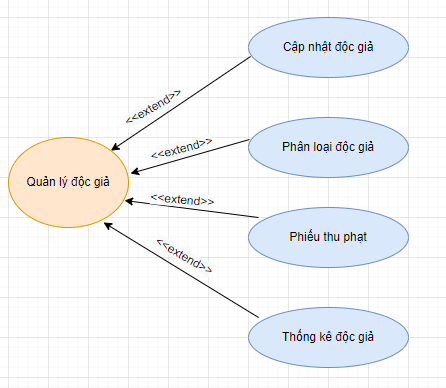
Hình 2.10: Biểu đồ use case mức tổng quát

1. ***Mô hình usecase mức dưới tổng quát***



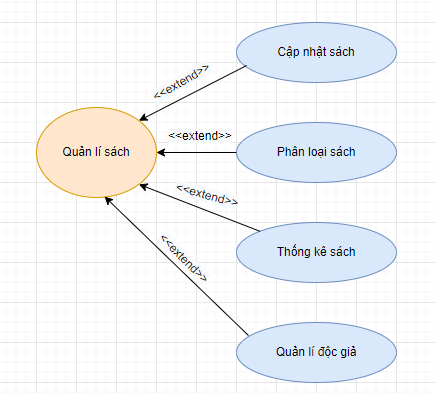
Hình 2.11: Biểu đồ use case mức dưới tổng quát

1. ***Phân rã use case quản độc giả***



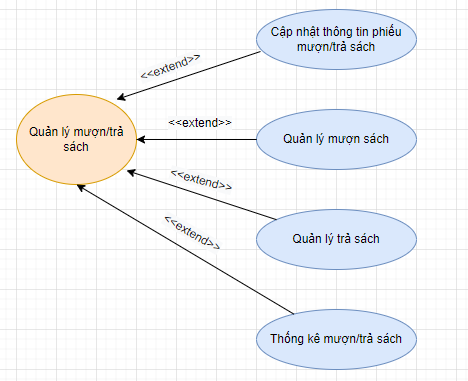
Hình 2.12: Phân rã use case quản lý đôc giả

1. ***Phân rã use case quản lý sách***



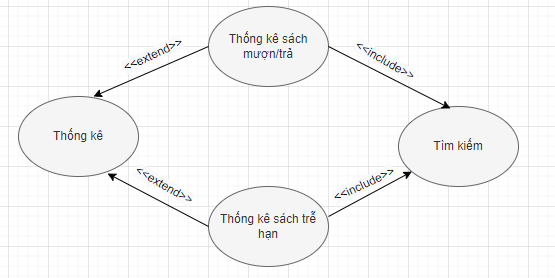
Hình 2.13: Phân rã use case quản lý sách

1. ***Phân rã use case quản lý mượn trả***



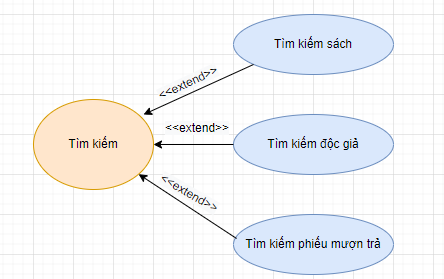
Hình 2.14: Phân rã use case quản lý mượn trả sách

1. ***Phân rã use case thống kê***



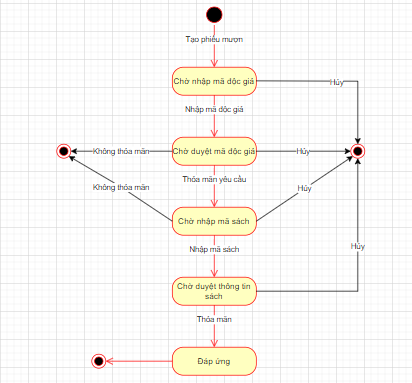
Hình 2.15: Phân rã use case thống kê

1. ***Phân rã use case tìm kiếm***



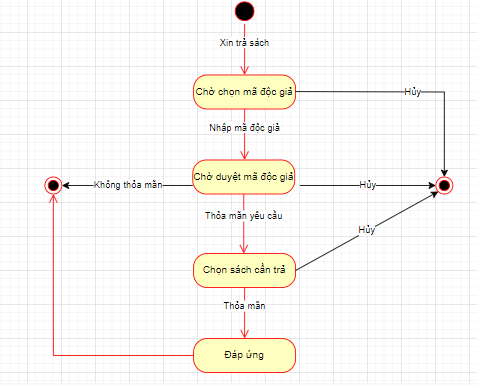
Hình 2.16: Phân rã use case tìm kiếm

1. ***Quy trình quản lý mượn sách***



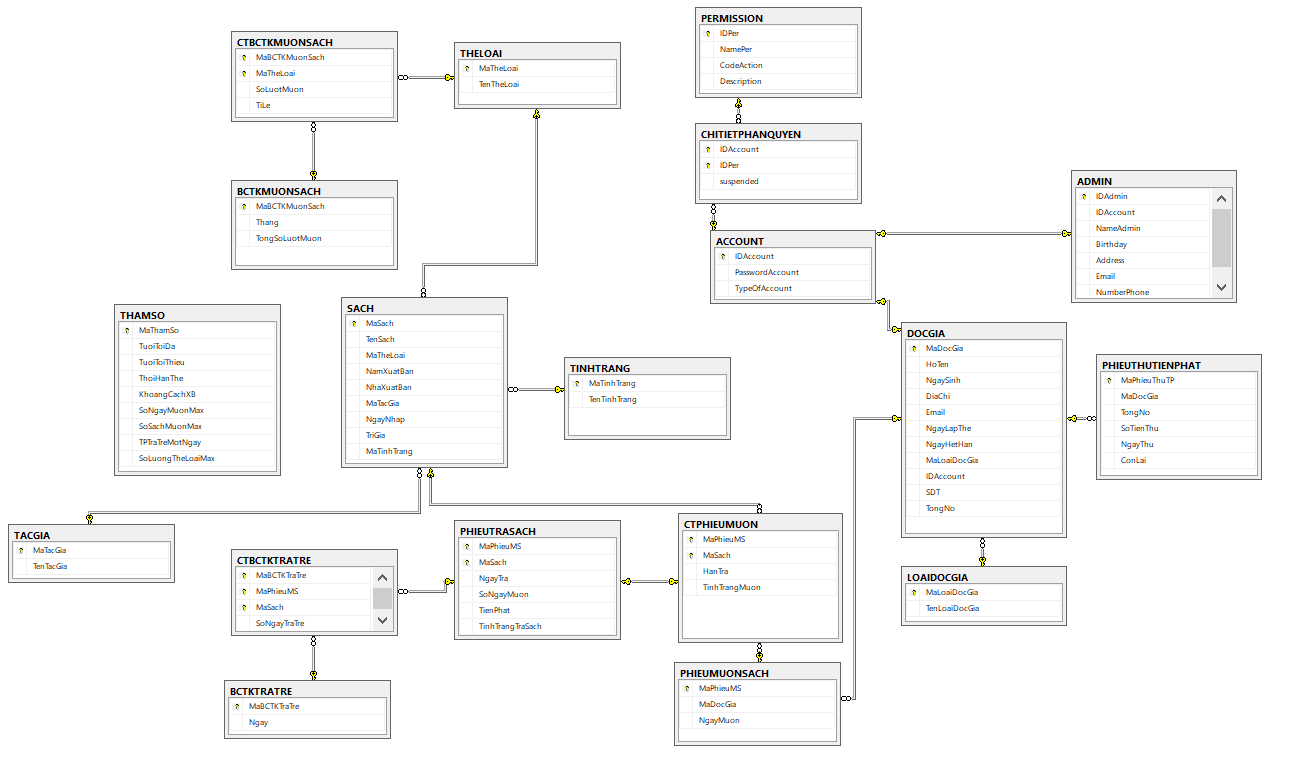
Hình 2.17: Biểu đồ trạng thái cho lớp mượn sách

1. ***Quy trình quản lý trả sách***



Hình 2.18: Biểu đồ trạng thái cho lớp trả sách

1. **Sơ đồ logic hoàn chỉnh:**



Hình 2.19: Minh họa sơ đồ cơ sở dữ liệu

1. **Danh sách các bảng dữ liệu (table) trong sơ đồ:**

|  |  |  |
| --- | --- | --- |
| **STT** | **Tên bảng dữ liệu** | **Diễn giải** |
| 1 | DOCGIA | Thông tin độc giả. |
| 2 | LOAIDOCGIA | Thông tin loại độc giả. |
| 3 | SACH | Thông tin về sách. |
| 4 | THELOAI | Danh sách thể loại. |
| 5 | TACGIA | Danh sách tác giả. |
| 6 | BCTKMUONSACH | Báo cáo tổng kết mượn sách theo thể loại trong một tháng |
| 7 | BCTKTRATRE | Báo cáo tổng kết số độc giả trả trễ |
| 8 | CHITIETPHANQUYEN | Phân rõ quyền người dùng |
| 9 | CTBCTKMUONSACH | Chi tiết nội dung báo cáo mượn sách theo thể loại |
| 10 | CTBCTKTRATRE | Chi tiết nội dung báo cáo việc trả trễ |
| 11 | CTPHIEUMUON | Danh sách thông tin việc mượn sách của độc giả |
| 12 | PERMISSION | Sự chấp nhận của người dùng |
| 13 | PHIEUMUONSACH | Thông tin phiếu mượn sách |
| 14 | PHIEUTRASACH | Thông tin phiếu trả sách |
| 15 | PHIEUTHUTIENPHAT | Thông tin về phiếu phạt tiền |
| 16 | THAMSO | Lưu trữ tên và giá trị các tham số có thể thay đổi |
| 17 | TINHTRANG | Tình trạng của sách |
| 18 | ACCOUNT | Thông tin người dùng |
| 19 | ADMIN | Thông tin về người quản lý |

1. **Mô tả từng bảng dữ liệu**
2. **Bảng DOCGIA**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Thuộc tính** | **Kiểu dữ liệu** | **Ràng buộc** | **Diễn giải** |
| 1 | MaDocGia | Varchar(50) | Khóa chính | Mã độc giả |
| 2 | HoTen | Nvarchar (100) |  | Họ và tên độc giả |
| 3 | NgaySinh | Date |  | Ngày sinh độc giả |
| 4 | DiaChi | Nvarchar (200) |  | Địa chỉ nơi ở độc giả |
| 5 | Email | Nvarchar (100) |  | Email độc giả |
| 6 | NgayLapThe | Date |  | Ngày lập thẻ độc giả |
| 7 | NgayHetHan | Date |  | Ngày hết hạn thẻ độc giả |
| 8 | MaLoaiDocGia | Varchar (50) |  | Mã loại độc giả |
| 9 | IDAccount | Varchar (50) |  | Mã tài khoản |
| 10 | SDT | Varchar (50) |  | Số điện thoại độc giả |
| 11 | TongNo | Decimal (18,0) |  | Tổng nợ của độc giả |

1. **Bảng LOAIDOCGIA**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Thuộc tính** | **Kiểu dữ liệu** | **Ràng buộc** | **Diễn giải** |
| 1 | MaLoaiDocGia | Varchar(50) | Khóa chính | Mã loại độc giả |
| 2 | TenLoaiDocGia | Nvarchar (100) |  | Tên loại độc giả |

1. **Bảng SACH**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Thuộc tính** | **Kiểu dữ liệu** | **Ràng buộc** | **Diễn giải** |
| 1 | MaSach | Varchar(50) | Khóa chính | Mã sách |
| 2 | TenSach | Nvarchar (100) |  | Tên sách |
| 3 | MaTheLoai | Varchar (50) |  | Mã thể loại sách |
| 4 | NamXuatBan | Int |  | Năm xuất bản quyển sách |
| 5 | NhaXuatBan | Nvarchar (100) |  | Nhà xuất bản sách |
| 6 | MaTacGia | Varchar (50) |  | Mã tác giả |
| 7 | NgayNhap | Date |  | Ngày nhập cuốn sách vào phần mềm |
| 8 | TriGia | Decimal (18,0) |  | Giá trị quyển sách |
| 9 | MaTinhTrang | Varchar (50) |  | Mã tình trạng cuốn sách |

1. **Bảng THELOAI**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Thuộc tính** | **Kiểu dữ liệu** | **Ràng buộc** | **Diễn giải** |
| 1 | MaTheLoai | Varchar(50) | Khóa chính | Mã thể loại sách |
| 2 | TenTheLoai | Nvarchar (100) |  | Tên thể loại sách |

1. **Bảng TACGIA**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Thuộc tính** | **Kiểu dữ liệu** | **Ràng buộc** | **Diễn giải** |
| 1 | MaTacGia | Varchar(50) | Khóa chính | Mã thể tác giả |
| 2 | TenTacGia | Nvarchar (100) |  | Tên thể tác giả |

1. **Bảng BCTKMUONSACH**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Thuộc tính** | **Kiểu dữ liệu** | **Ràng buộc** | **Diễn giải** |
| 1 | MaBCTKMuonSach | Varchar(50) | Khóa chính | Mã báo cáo mượn sách |
| 2 | Thang | Date |  | Tên thể loại sách |
| 3 | TongSoLuotMuon | Int |  | Tổng số lượt mượn của một thể loại sách |

1. **Bảng CTBCTKMUONSACH**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Thuộc tính** | **Kiểu dữ liệu** | **Ràng buộc** | **Diễn giải** |
| 1 | MaBCTKMuonSach | Varchar(50) | Khóa chính | Mã báo cáo tổng kết mượn sách |
| 2 | MaTheLoai | Varchar (50) | Khóa chính | Mã thể loại sách |
| 3 | SoLuotMuon | Int |  | Số lượt mượn |
| 4 | TiLe | Float |  | Tỉ lệ phần trăm mượn |

1. **Bảng BCTKTRATRE**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Thuộc tính** | **Kiểu dữ liệu** | **Ràng buộc** | **Diễn giải** |
| 1 | MaBCTKTraTre | Varchar(50) | Khóa chính | Mã báo cáo tổng kết trả trễ |
| 2 | Ngay | Date |  | Ngày cần báo cáo |

1. **Bảng CTBCTKTRATRE**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Thuộc tính** | **Kiểu dữ liệu** | **Ràng buộc** | **Diễn giải** |
| 1 | MaBCTKTraTre | Varchar(50) | Khóa chính | Mã báo cáo tổng kết trả trễ |
| 2 | MaPhieuMS | Varchar (50) | Khóa chính | Mã phiếu mượn sách |
| 3 | MaSach | Varchar (50) | Khóa chính | Mã cuốn sách trả trễ |
| 4 | SoNgayTra | Int |  | Số ngày trả quá hạn |

1. **Bảng CHITIETPHANQUYEN**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Thuộc tính** | **Kiểu dữ liệu** | **Ràng buộc** | **Diễn giải** |
| 1 | IDAccount | Varchar(50) | Khóa chính | Mã tài khoản |
| 2 | IDPer | Nvarchar (50) | Khóa chính | Mã phân quyền |
| 3 | Suspended | Bit |  | Gián đoạn |

1. **Bảng PERMISSION**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Thuộc tính** | **Kiểu dữ liệu** | **Ràng buộc** | **Diễn giải** |
| 1 | IDAPer | Varchar(50) | Khóa chính | Mã phân quyền |
| 2 | NamePer | Nvarchar (100) |  | Tên quyền |
| 3 | CodeAction | Nvarchar (100) |  | Mã hành động |
| 4 | Description | Nvarchar (200) |  | Mô tả hành động |

1. **Bảng ACCOUNT**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Thuộc tính** | **Kiểu dữ liệu** | **Ràng buộc** | **Diễn giải** |
| 1 | IDAccount | Varchar(50) | Khóa chính | Mã tài khoản |
| 2 | PasswordAccount | Nvarchar (100) |  | Mật khẩu tài khoản |
| 3 | TypeOfAccount | Varchar (50) |  | Loại tài khoản |

1. **Bảng ADMIN**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Thuộc tính** | **Kiểu dữ liệu** | **Ràng buộc** | **Diễn giải** |
| 1 | IDAdmin | Varchar(50) | Khóa chính | Mã quản lý |
| 2 | IDAccount | Varchar (50) |  | Mã tài khoản |
| 3 | NameAdmin | Nvarchar (100) |  | Tên quản lý |
| 4 | Birthday | Datetime |  | Ngày sinh |
| 5 | Address | Nvarchar (200) |  | Địa chỉ |
| 6 | Email | Varchar (100) |  | Email quản lý |
| 7 | NumberPhone | Varchar (50) |  | Số điện thoại quản lý |

1. **Bảng TINHTRANG**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Thuộc tính** | **Kiểu dữ liệu** | **Ràng buộc** | **Diễn giải** |
| 1 | MaTinhTrang | Varchar(50) | Khóa chính | Mã tình trạng |
| 2 | TenTinhTrang | Nvarchar (100) |  | Tên tình trạng |

1. **Bảng THAMSO**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Thuộc tính** | **Kiểu dữ liệu** | **Ràng buộc** | **Diễn giải** |
| 1 | MaThamSo | Int | Khóa chính | Mã tham số |
| 2 | TuoiToiDa | Int |  | Tuổi tối đa độc giả |
| 3 | TuoiToiThieu | Int |  | Tuổi tối thiểu độc giả |
| 4 | ThoiHanThe | Int |  | Thời hạn thẻ độc giả |
| 5 | KhoangCachXB | Int |  | Khoảng cách xuất bản |
| 6 | SoNgayMuonMAX | Int |  | Số ngày mượn tối đa |
| 7 | SoSachMuonMAX | Int |  | Số sách tối đa được mượn |
| 8 | TPTraTreMotNgay | Decimal (18,0) |  | Tiền phạt trễ một ngày |
| 9 | SoLuongTheLoaiMax | Int |  | Số lượng thể loại tối đa |

1. **Bảng PHIEUTRASACH**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Thuộc tính** | **Kiểu dữ liệu** | **Ràng buộc** | **Diễn giải** |
| 1 | MaPhieuMS | Varchar(50) | Khóa chính | Mã phiếu mượn |
| 2 | MaSach | Varchar (50) | Khóa chính | Mã sách trả |
| 3 | NgayTra | Date |  | Ngày trả |
| 4 | SoNgayMuonTre | Int |  | Số ngày trễ |
| 5 | TienPhat | Decimal(18,0) |  | Tiền phạt |
| 6 | TinhTrangTra | Nvarchar (200) |  | Tình trạng sách khi trả |

1. **Bảng PHIEUMUONSACH**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Thuộc tính** | **Kiểu dữ liệu** | **Ràng buộc** | **Diễn giải** |
| 1 | MaPhieuMS | Varchar(50) | Khóa chính | Mã phiếu mượn sách |
| 2 | MaDocGia | Varchar (50) |  | Mã độc giả mượn |
| 3 | NgayMuon | Date |  | Ngày mượn |

1. **Bảng CTPHIEUMUON**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Thuộc tính** | **Kiểu dữ liệu** | **Ràng buộc** | **Diễn giải** |
| 1 | MaPhieuMS | Varchar(50) | Khóa chính | Mã phiếu mượn sách |
| 2 | MaSach | Varchar (50) | Khóa chính | Mã sách mượn |
| 3 | HanTra | Datetime |  | Hạn trả sách |
| 4 | TinhTrangMuon | Nvarchar(200) |  | Tình trạng mượn |

1. **Bảng PHIEUTHUTIENPHAT**

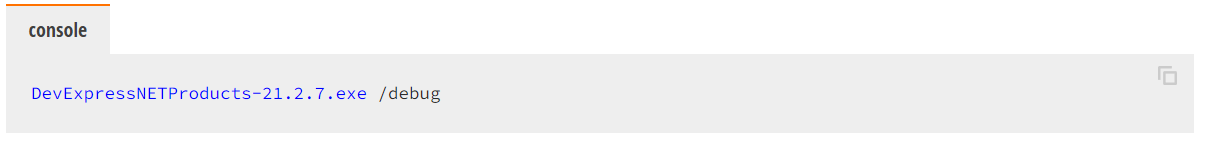
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Thuộc tính** | **Kiểu dữ liệu** | **Ràng buộc** | **Diễn giải** |
| 1 | MaPhieuThuTienPhat | Varchar(50) | Khóa chính | Mã phiếu thu tiền phạt |
| 2 | MaDocGia | Varchar (50) |  | Mã độc giả bị phạt |
| 3 | TongNo | Decimal (18,0) |  | Tổng nợ |
| 4 | SoTienThu | Decimal (18,0) |  | Số tiền thu |
| 5 | NgayThu | Date |  | Ngày thu |
| 6 | ConLai | Decimal (18,0) |  | Còn lại |

# **CHƯƠNG 3: XÂY DỰNG CHƯƠNG TRÌNH**

1. **Cài đặt dev express**
2. **Chạy trình cài đặt**

Chạy tệp DevExpressNETProducts-21.2.7 .exe mà đã tải xuống từ www.devexpress.com.

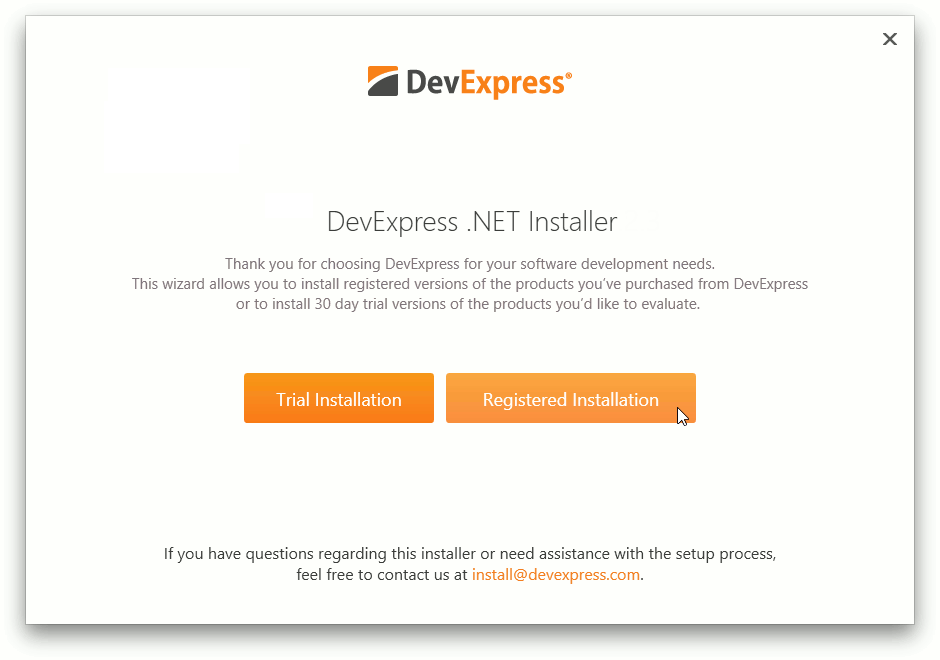
Có thể chạy công cụ trình cài đặt với các công tắc tương tự, như được mô tả trong tài liệu. Ví dụ, bạn có thể sử dụng debug để lấy tệp nhật ký của quá trình cài đặt.



Hình 3.1:Cài đặt file dev express

1. **Đăng ký sản phẩm**

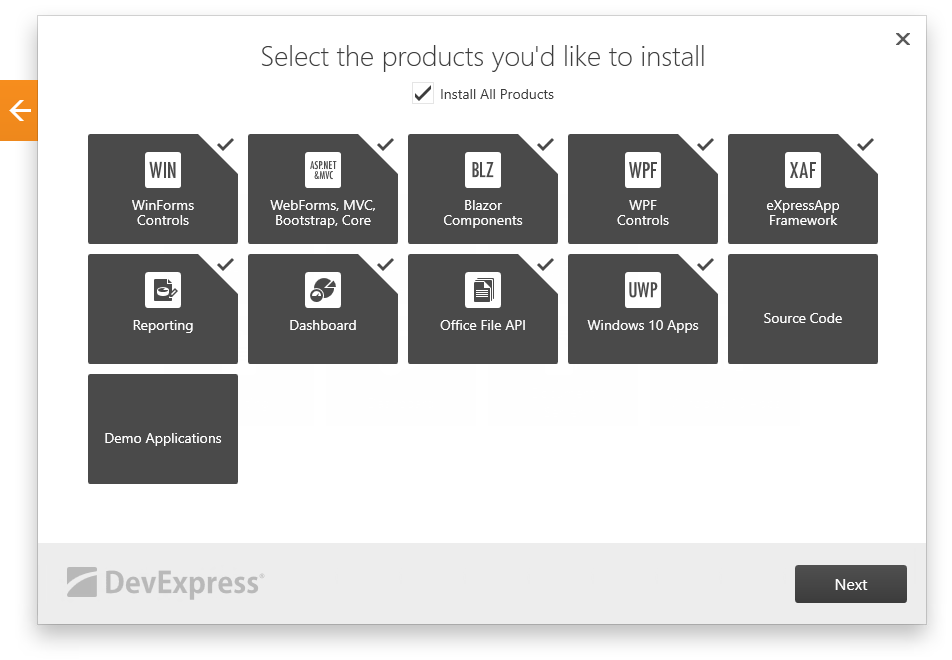
Sau khi bạn chạy trình cài đặt, nó sẽ hiển thị trang chào mừng cung cấp cho bạn tùy chọn cài đặt các sản phẩm DevExpress .NET ở chế độ Dùng thử hoặc Đã đăng ký. Lưu ý rằng có thể đăng ký sản phẩm của sau nếu bạn chọn Cài đặt dùng thử .



Hình 3.2: Đăng ký dev express

1. **Chọn các sản phẩm**

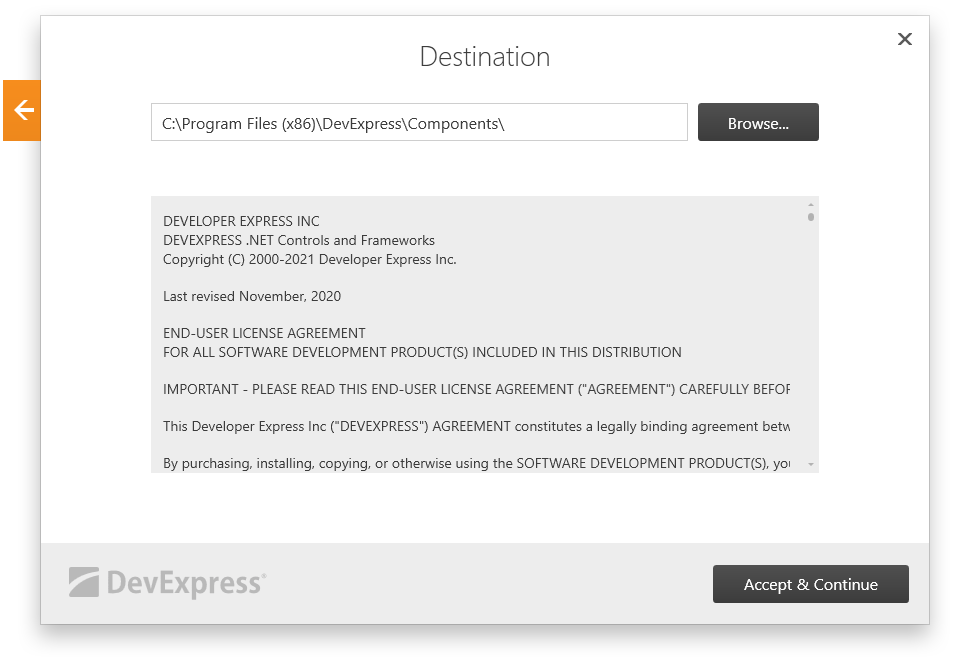
Trên trang tiếp theo, chọn các sản phẩm bạn muốn cài đặt. Chọn hoặc bỏ chọn tùy chọn Cài đặt Tất cả Sản phẩm để chọn cài đặt tất cả các sản phẩm hay chỉ các sản phẩm đã đăng ký. Nhấp vào Tiếp theo để sang trang tiếp theo. Lưu ý rằng có thể cài đặt các sản phẩm khác và sử dụng chúng ở chế độ Dùng thử nếu bạn không có giấy phép nhà phát hành.



Hình 3.3: Chọn sản phẩm dev express

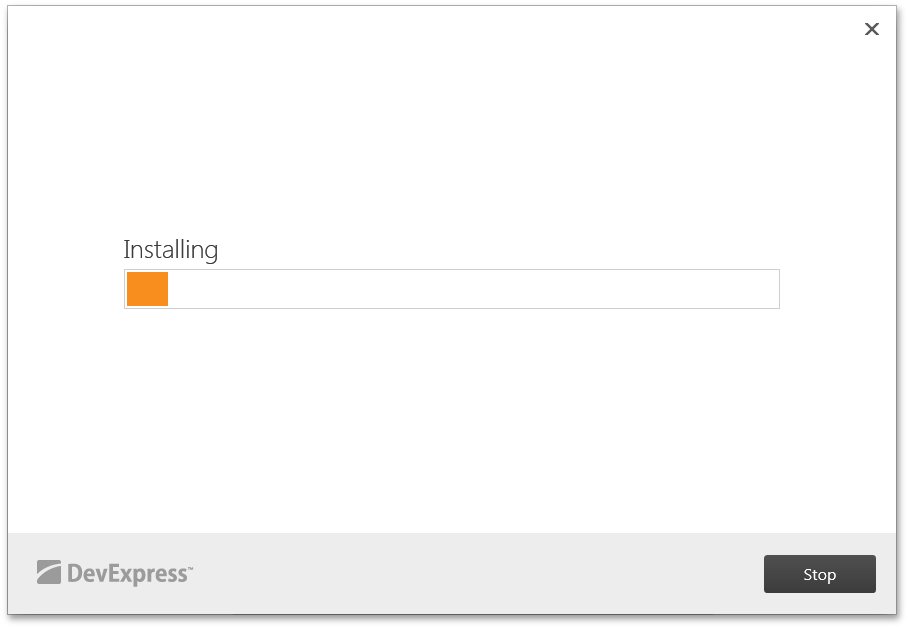
1. **Chọn đường dẫn đích**

Trên trang này, có thể chọn đường dẫn để cài đặt các sản phẩm DevExpress .NET. Nhấp vào Chấp nhận & Cài đặt nếu chấp nhận thỏa thuận cấp phép người dùng cuối để bắt đầu quá trình cài đặt.



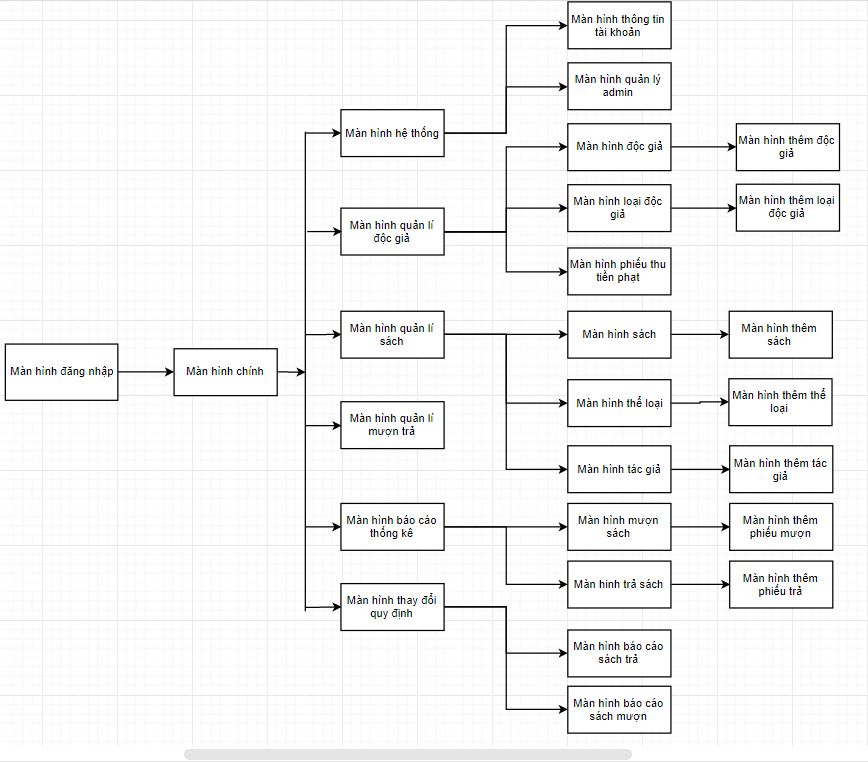
Hình 3.4: Chọn đường dẫn dev express

Thanh tiến trình hiển thị trong quá trình cài đặt để hiển thị tiến trình của quá trình cài đặt. Nhấp vào nút Dừng nếu bạn muốn làm gián đoạn hoặc hủy quá trình cài đặt. Nếu không, hãy đợi cho đến khi trình cài đặt thực hiện tất cả các hành động cần thiết và thông báo cho bạn rằng quá trình cài đặt đã hoàn tất.



Hình 3.5: Cài đặt dev express

1. **Sơ đồ liên kết các màn hình**

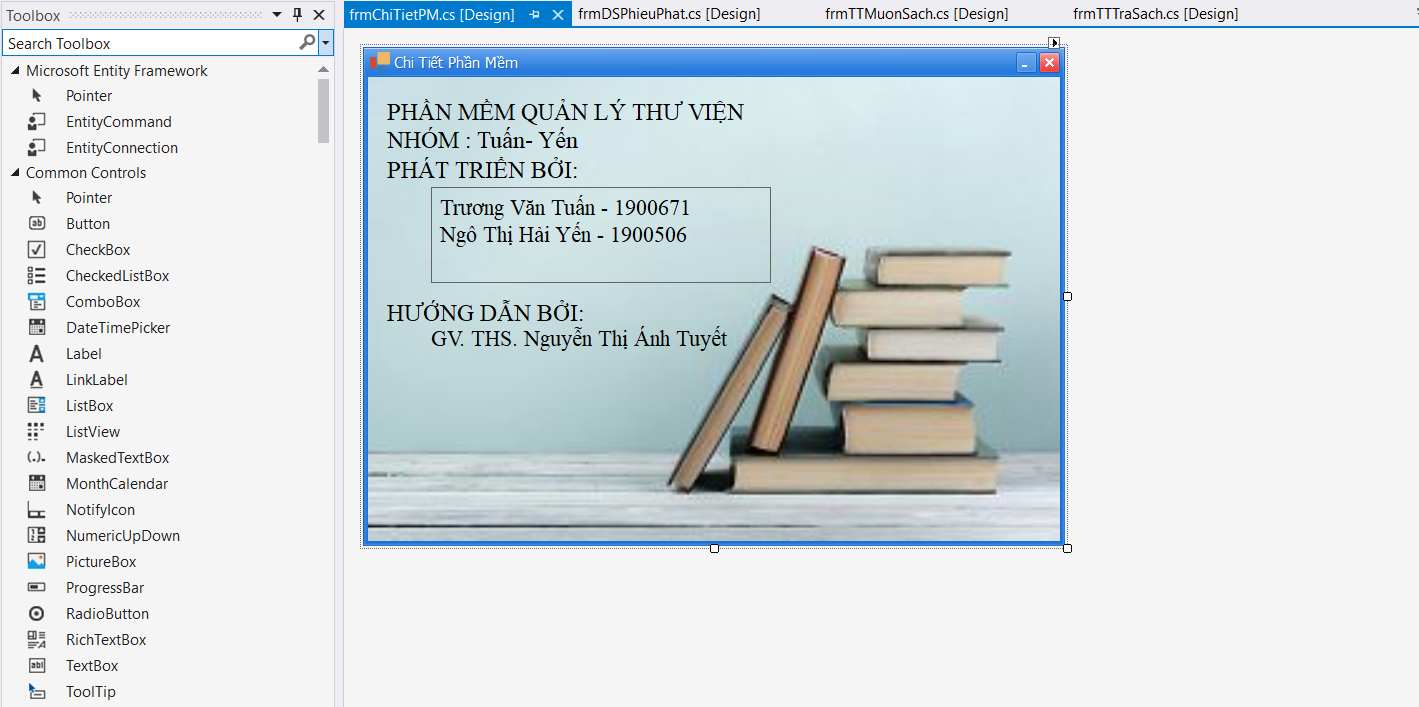


Hình 3.6: Sơ đồ liên kết các màn hình

1. **Danh sách các màn hình**

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Màn hình** | **Loại màn hình** | **Chức năng** |
| 1 | Màn hình đăng nhập | Nhập liệu | Đăng nhập vào phần mềm |
| 2 | Trang chủ | Màn hình chính | Cho phép người dùng thao tác các công việc |
| 3 | Màn hình hệ thống | Báo biểu | Thông tin người dùng và admin |
| 4 | Màn hình độc giả | Nhập liệu, Tra cứu | Dùng để xem danh sách độc giả, tra cứu thông tin và sửa chữa thông tin |
| 5 | Màn hình thêm độc giả | Nhập liệu | Thêm độc giả vào danh sách |
| 6 | Màn hình loại tác giả | Tra cứu, nhập liệu | Dùng để xem danh sách loại tác giả và sửa chữa |
| 7 | Màn hình thêm loại tác giả | Nhập liệu | Thêm loại độc giả |
| 8 | Màn hình Phiếu thu tiền phạt | Tra cứu, nhập liệu | Dùng để them phiếu phạt và tra cứu phiếu phạt |
| 9 | Màn hình Sách | Báo biểu, nhập liệu | Hiển thị danh sách sách và sửa chữa thông tin sách |
| 10 | Màn hình thêm sách | Nhập liệu | Thêm thông tin sách |
| 11 | Màn hình thể loại | Tra cứu, nhập liệu | Dùng để tra cứu thông tin thể loại sách và sửa chữa thể loại sách |
| 12 | Màn hình thêm thể loại | Nhập liệu | Dùng để them thể loại sách |
| 13 | Màn hình tác giả | Tra cứu, nhập liệu | Dùng để hiển thị thông tin tác giả và sửa chữa thông tin tác giả |
| 14 | Màn hình thêm tác giả | Nhập liệu | Dùng để thêm tác giả |
| 15 | Màn hình mượn sách | Tra cứu | Hiển thị các thông tin mượn sách |
| 16 | Màn hình thêm phiếu mượn | Nhập liệu | Dùng để thêm phiếu mượn sách |
| 17 | Màn hình trả sách | Tra cứu | Hiển thị các thông tin trả sách |
| 18 | Màn hình thêm phiếu trả | Nhập liệu | Dùng để thêm phiếu trả sách |
| 19 | Màn hình báo cáo mượn sách | Báo biểu | Hiển thị tổng kết thông tin mượn sách |
| 20 | Màn hình báo cáo trả sách | Báo biểu | Hiển thị tổng kết thông tin trả sách |
| 21 | Màn hình thông tin tài khoản | Báo biểu, nhập liệu | Hiển thị thông tin người dùng và sửa chữa thông tin |
| 22 | Màn hình quản lý admin | Báo biểu | Quản lý |
| 23 | Màn hình hệ thống | Báo biểu | Hiển thị các chức năng người dung |
| 24 | Màn hình quản lí độc giả | Báo biểu | Hiển thị các chức năng độc giả |
| 25 | Màn hình quản lý sách | Báo biểu | Hiển thị các chức năng sách |
| 26 | Màn hình quản lý mượn trả | Báo biểu | Hiển thị các chức năng cho phép mượn trả sách |
| 27 | Màn hình quản lý báo cáo thông kê | Báo biểu | Hiển thị các chức năng báo cáo thông kê |
| 28 | Màn hình thay đổi quy định | Báo biểu, nhập liệu | Hiển thị các quy định và cho phép thay đổi các quy định đó |

1. **Mô tả các màn hình**
2. **Màn hình chính**
3. **Giao diện**



Hình 3.7: Giao diện thiết kế frmthongtinpm

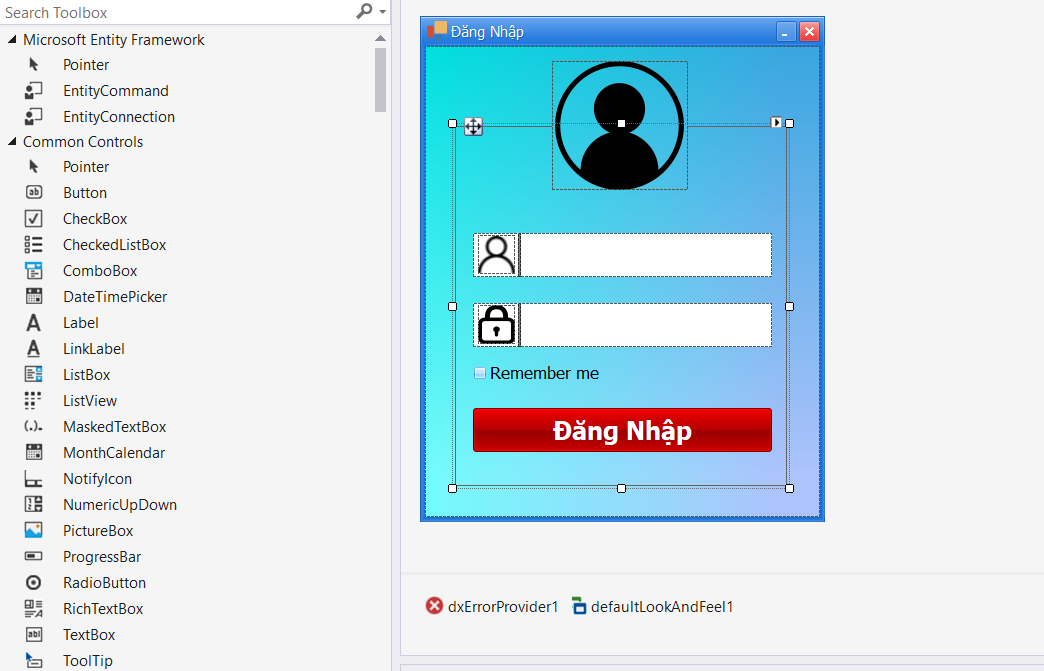
1. **Mô tả các đối tượng trên màn hình**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Tên | Kiểu | Ràng buộc | Chức năng |
| 1 | Hệ thống | Tab |  | Thông tin người dung và quản lý |
| 2 | Quản lý độc giả | Tab |  | Hiển thị màn hình chức năng độc giả |
| 3 | Quản lý Sách | Tab |  | Hiển thị màn hình chức năng sách |
| 4 | Quản lý mượn trả | Tab |  | Hiển thị màn hình chức năng mượn trả sách |
| 5 | Báo cáo thống kê | Tab |  | Hiển thị báo cáo thông kê mượn trả |
| 6 | Quy định | Tab |  | Hiển thị các quy định và cho phép thay đổi quy định |
| 7 | Tài khoản | Subitem |  | Thông tin tài khoản |
| 8 | Quản lý admin | Button item |  | Quản lý |
| 9 | Theme | Gallery bar item |  | Nền |
| 10 | Đăng xuất | Bar button |  | Thoát tài khoản |

1. **Danh sách các biến cố và xử lý tương ứng trên màn hình**

|  |  |  |
| --- | --- | --- |
| STT | Biến cố | Xử lý |
| 1 | Khi bấm vào tab Hệ thống | Hiển thị quản lý tài khoản, theme, quản lý admin, đăng xuất |
| 2 | Khi bấm vào tab Quản lý độc giả | Hiển thị các chức năng: Độc giả, Loại độc giả, Phiếu thu tiền phạt |
| 3 | Khi bấm vào tab Quản lý Sách | Hiển thị các chức năng: Sách, Thể loại, Tác giả |
| 4 | Khi bấm vào tab Quản lý mượn trả | Hiển thị chức năng mượn sách và trả sách |
| 5 | Khi bấm vào Báo cáo thống kê | Hiển thị báo cáo mượn và báo cáo trả |
| 6 | Khi bấm vào Quy định | Hiển thị các quy định |
| 7 | Khi bấm vào Tài khoản | Hiển thị thông tin tài khoản |
| 8 | Khi bấm vào Quản lý admin | Hiển thị chức năng quản lý |
| 9 | Khi bấm vào Theme | Hiển thị bảng skin |
| 10 | Khi bấm vào Đăng xuất | Thoát tài khoản |

1. **Màn hình đăng nhập**
2. **Giao diện**



Hình 3.8: Giao diện thiết kế frmlogin

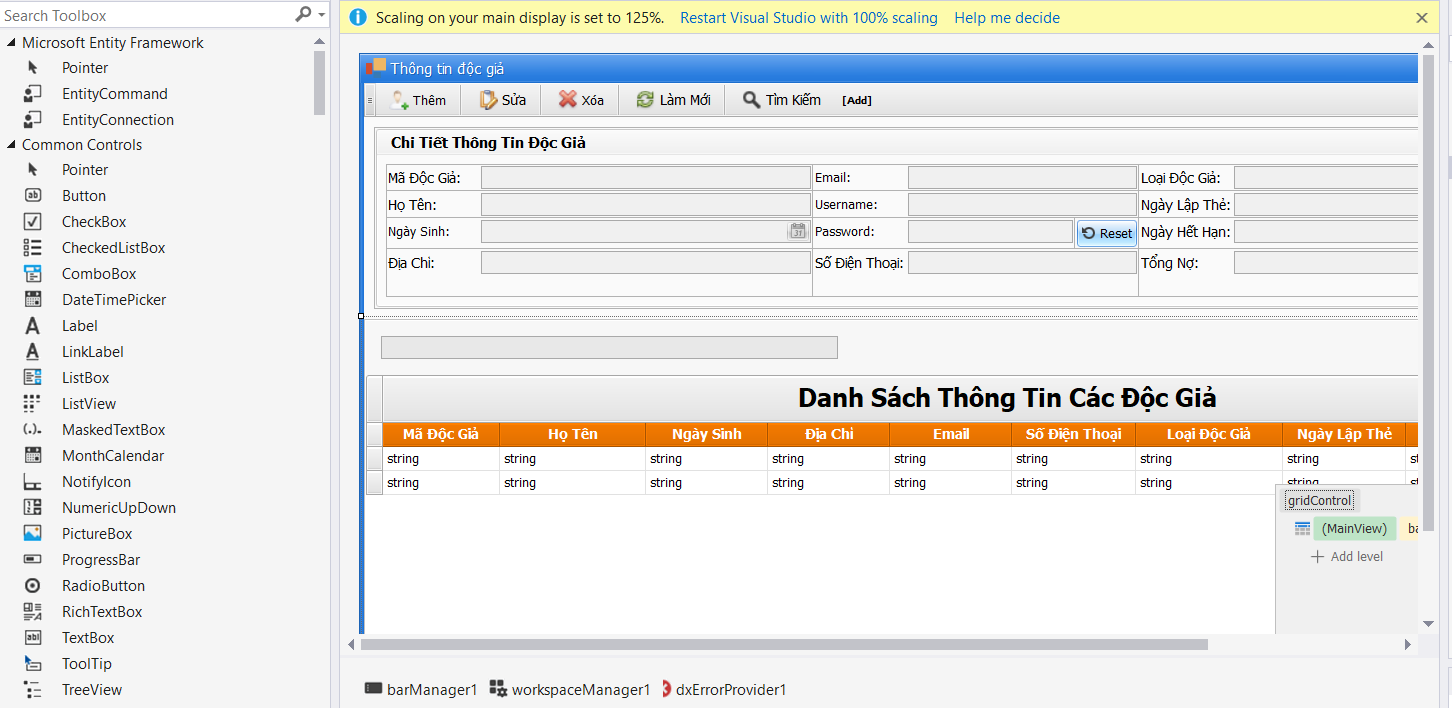
1. **Mô tả các đối tượng trên màn hình**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Tên | Kiểu | Ràng buộc | Chức năng |
| 1 | btnDangNhap | Button |  | Đăng nhập vào phần mềm |
| 2 | Ck\_remember | Checkedit |  | Lưu mật khẩu |
| 3 | txbTaiKhoan | Text |  | Nhập vào tài khoản |
| 4 | txbMatKhau | Text |  | Nhập vào mật khẩu |

1. **Danh sách các biến cố và xử lý tương ứng trên màn hình**

|  |  |  |
| --- | --- | --- |
| STT | Biến cố | Xử lý |
| 1 | Khi bấm vào btnDangNhap | Cho phép đăng nhập vào tài khoản người dung |
| 2 | Khi bấm vào Ck\_remember | Lưu trữ mật khẩu tiện cho lần sau |
| 3 | Khi bấm vào txbTaiKhoan | Người dung nhập vào tài khoản |
| 4 | Khi bấm vào txbMatKhau | Người dung nhập vào mật khẩu |

1. **Màn hình độc giả**
2. **Giao diện**



Hình 3.9: Giao diện thiết kế frmdocgia

1. **Mô tả các đối tượng trên màn hình**

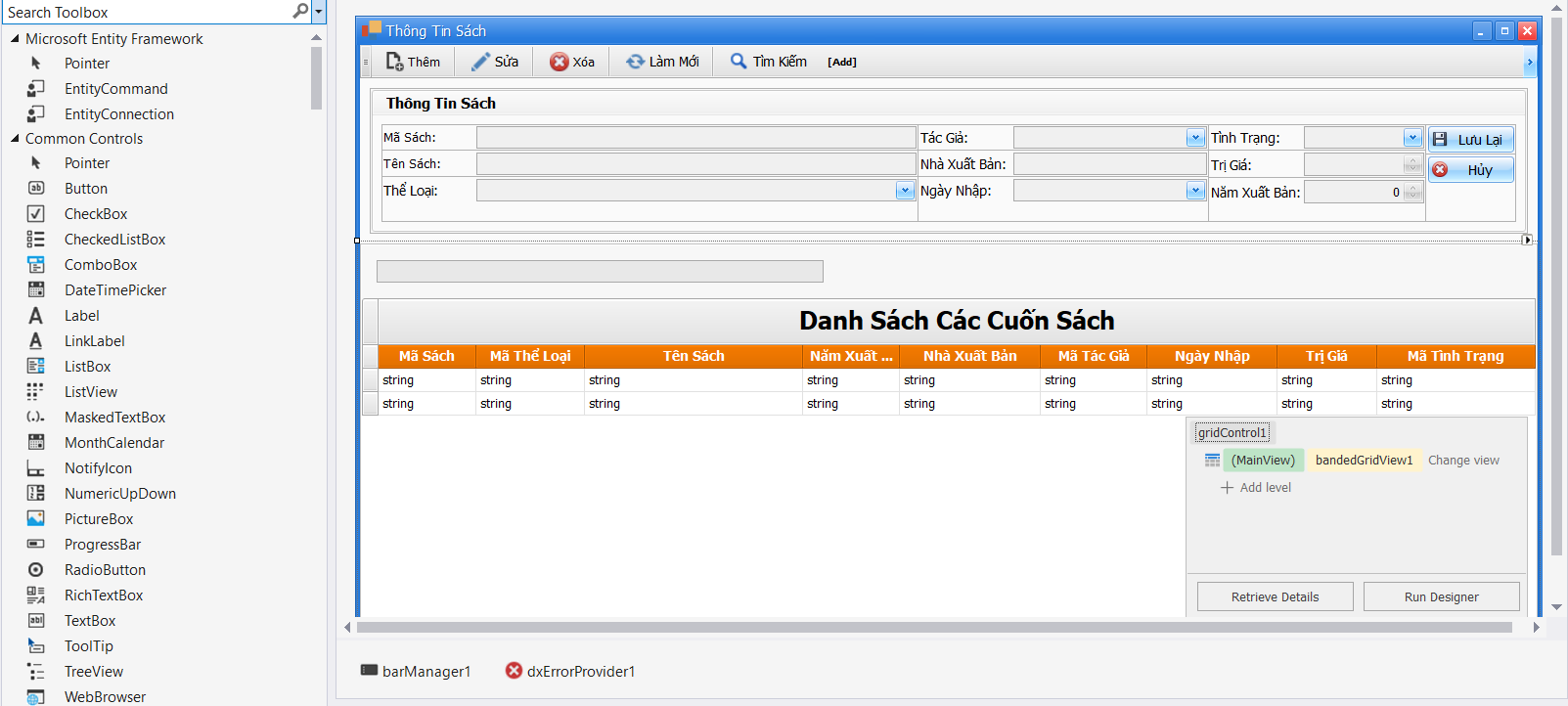
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Tên | Kiểu | Ràng buộc | Chức năng |
| 1 | btnThem | Button |  | Thêm độc giả |
| 2 | gridDocGia | Checkedit |  | Hiển thị danh sách độc giả |
| 3 | btnSua | Button |  | Sửa độc giả |
| 4 | btnXoa | Button |  | Xóa độc giả |
| 5 | btnLamMoi | Button |  | Tải lại CSDL |
| 6 | btnTimKiem | Button |  | Tìm độc giả |
| 7 | btnHuy | Button |  | Không Lưu khi sửa |
| 8 | btnLuuLai | button |  | Lưu khi sửa |

1. **Danh sách các biến cố và xử lý tương ứng trên màn hình**

|  |  |  |
| --- | --- | --- |
| STT | Biến cố | Xử lý |
| 1 | Khi bấm vào btnThem | Hiển thị form thêm độc giả |
| 2 | Khi bấm vào gridDocGia | Chọn vào một đối tượng trong list |
| 3 | Khi bấm vào btnSua | Cho phép sửa chữa thông tin độc giả đông thời hiển thị 2 phím lưu và hủy |
| 4 | Khi bấm vào btnXoa | Độc giả được chọn sẽ bị xóa |
| 5 | Khi bấm vào btnLamMoi | Tải lại CSDL sau khi người dung đã thao tác |
| 6 | Khi bấm vào btnTimKiem | Cho phép người dung tìm kiếm thông tin độc giả |
| 7 | Khi bấm vào btnHuy | Không lưu lại thao tác sửa của người dùng |
| 8 | btnLuuLai | Lưu lại việc sửa của người dùng |

1. **Màn hình Sách**

**a.Giao diện**



Hình 3.10: Giao diện thiết kế frmsach

**b.Mô tả các đối tượng trên màn hình**

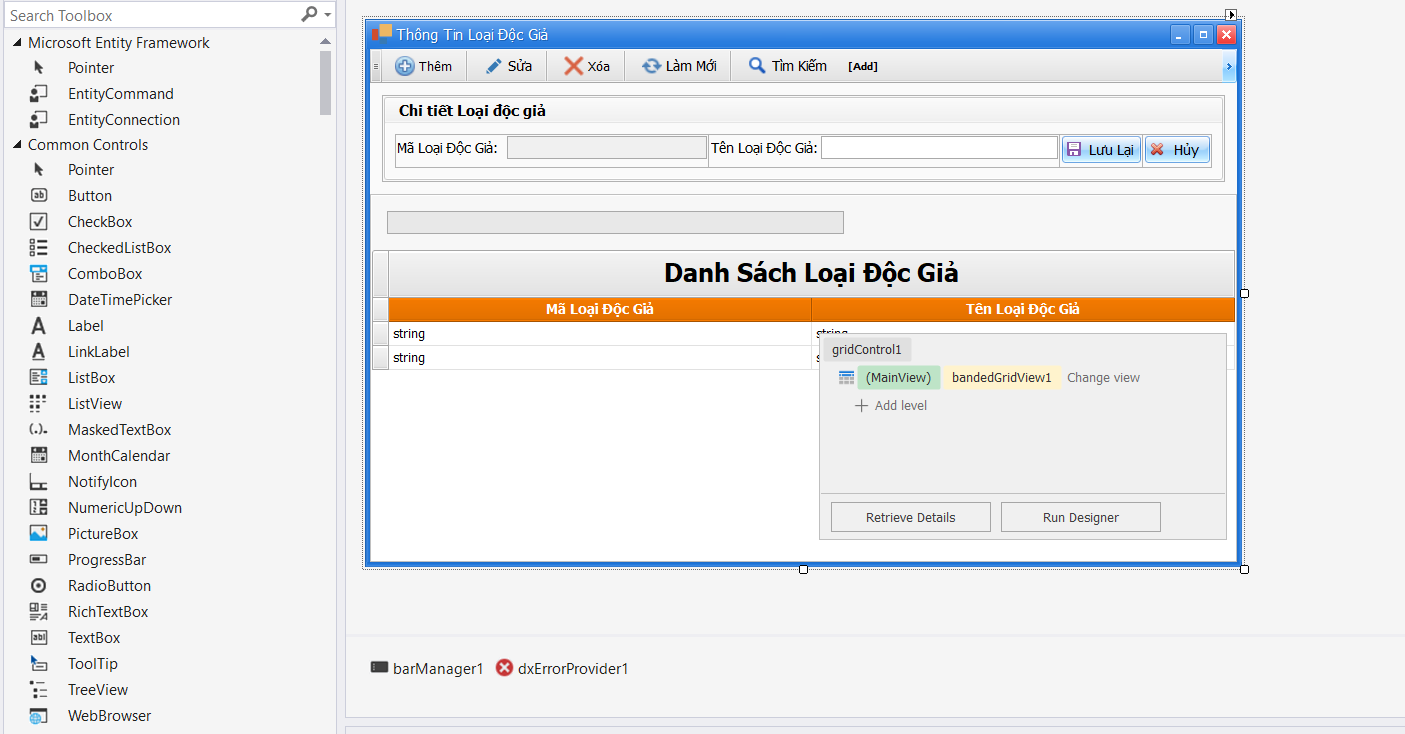
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Tên | Kiểu | Ràng buộc | Chức năng |
| 1 | label1 | Label |  | Hiển thị tên màn hình |
| 2 | gridSach | GridControl |  | Hiển thị danh sách sách, tra  cứu |
| 3 | txtMaSach | Textbox |  | Hiển thị/ nhập mã sách. |
| 4 | txtTenSach | Tetxbox |  | Hiển thị/ nhập tên sách. |
| 5 | cbTacGia | Combobox |  | Hiển thị/ chọn tác giả. |
| 6 | cbTheLoai | Combobox |  | Hiển thị/ chọn thể loại. |
| 7 | rdbTrong | Radiobox |  | Tình trạng sách đang trống  (chưa được mượn). |
| 8 | rdbBorrowed | Radiobox |  | Tình trạng sách đang được  mượn. |
| 9 | txtNXB | Textbox |  | Hiển thị/ nhập nhà xuất bản. |
| 10 | txtNamXB | Textbox |  | Hiển thị/ nhập năm xuất bản. |
| 11 | dtNgayNhap | Datetime |  | Hiển thị/ nhập ngày nhập  sách. |
| 12 | txtTriGia | Textbox |  | Hiển thị/ nhập trị giá sách. |
| 13 | btnImage | Button |  | Chọn file hình ảnh bìa sách. |

**c. Danh sách các biến cố và xử lý tương ứng trên màn hình**

|  |  |  |
| --- | --- | --- |
| STT | Biến cố | Xử lý |
| 1 | Chọn button Thêm | Xóa các giá trị trong textbox, bắt đầu thêm 1  cuốn sách. |
| 2 | Chọn button Sửa | Cho phép thay đổi các giá trị trong textbox của  cuốn sách đó. |
| 3 | Chọn button Lưu | Lưu lại thay đổi từ button Thêm và button Sửa |
| 4 | Chọn button Xóa | Xóa cuốn sách được chọn. |

1. **Màn hình Loại Độc Giả**

**a.Giao diện**



Hình 3.11: Giao diện thiết kế frmdocgia

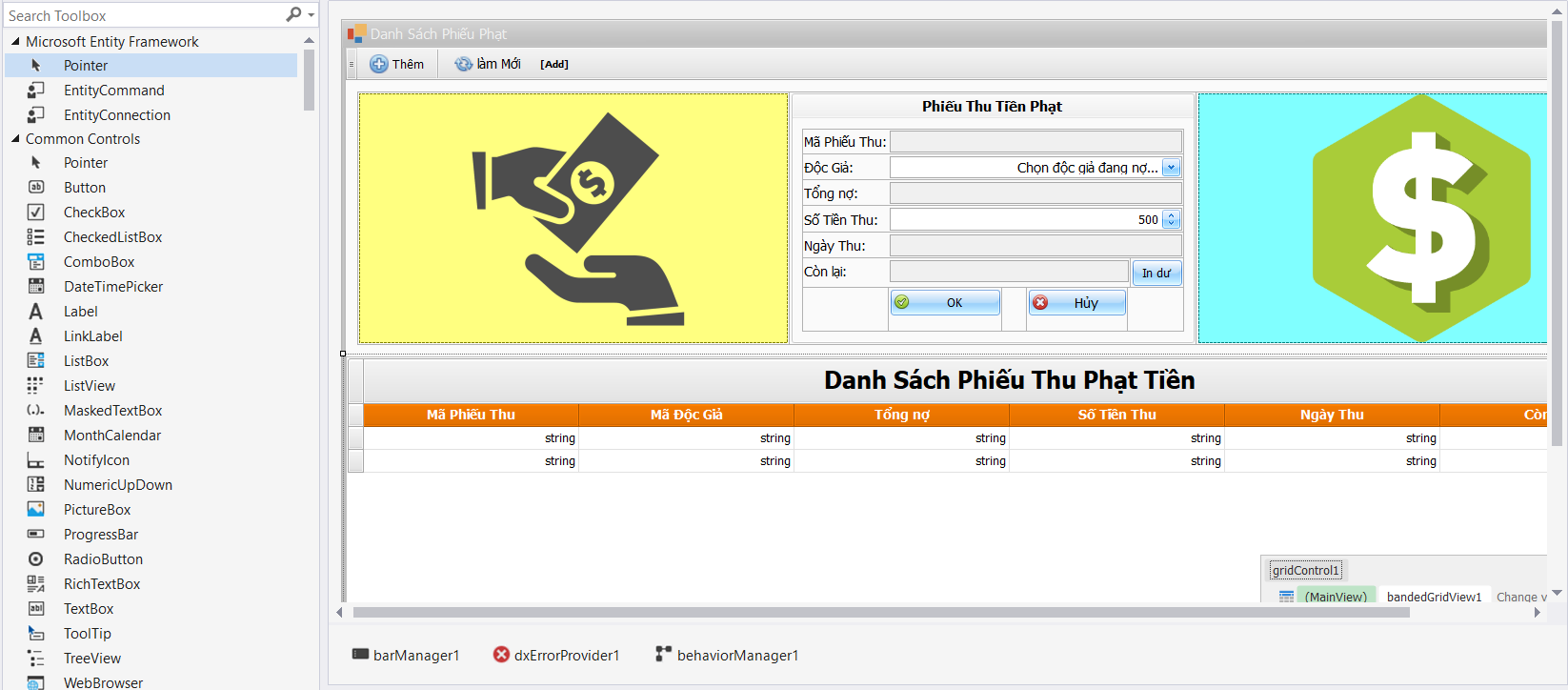
**b.Mô tả các đối tượng trên màn hình**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Tên | Kiểu | Ràng buộc | Chức năng |
| 1 | txtMaLDG | Textbox |  | Hiển thị mã loại đọc giả. |
| 2 | txtTenLDG | Textbox |  | Hiển thị/ nhập tên loại độc giả |
| 3 | gridLDG | gridControl |  | Hiển thị/ tra cứu danh sách loại độc giả. |

**c. Danh sách các biến cố và xử lý tương ứng trên màn hình**

|  |  |  |
| --- | --- | --- |
| STT | Biến cố | Xử lý |
| 1 | Chọn button Thêm | Xóa các giá trị trong textbox, bắt đầu thêm 1 loại đôc giả. |
| 2 | Chọn button Sửa | Cho phép thay đổi các giá trị trong textbox của loại đọc giả đó. |
| 3 | Chọn button Lưu | Lưu lại thay đổi từ button Thêm và button Sửa |
| 4 | Chọn button Xóa | Xóa loại đọc giả được chọn. |

1. **Màn hình Phiếu Thu Phạt Tiền**
2. **Giao diện**



Hình 3.12: Giao diện thiết kế frmDSphieuphat

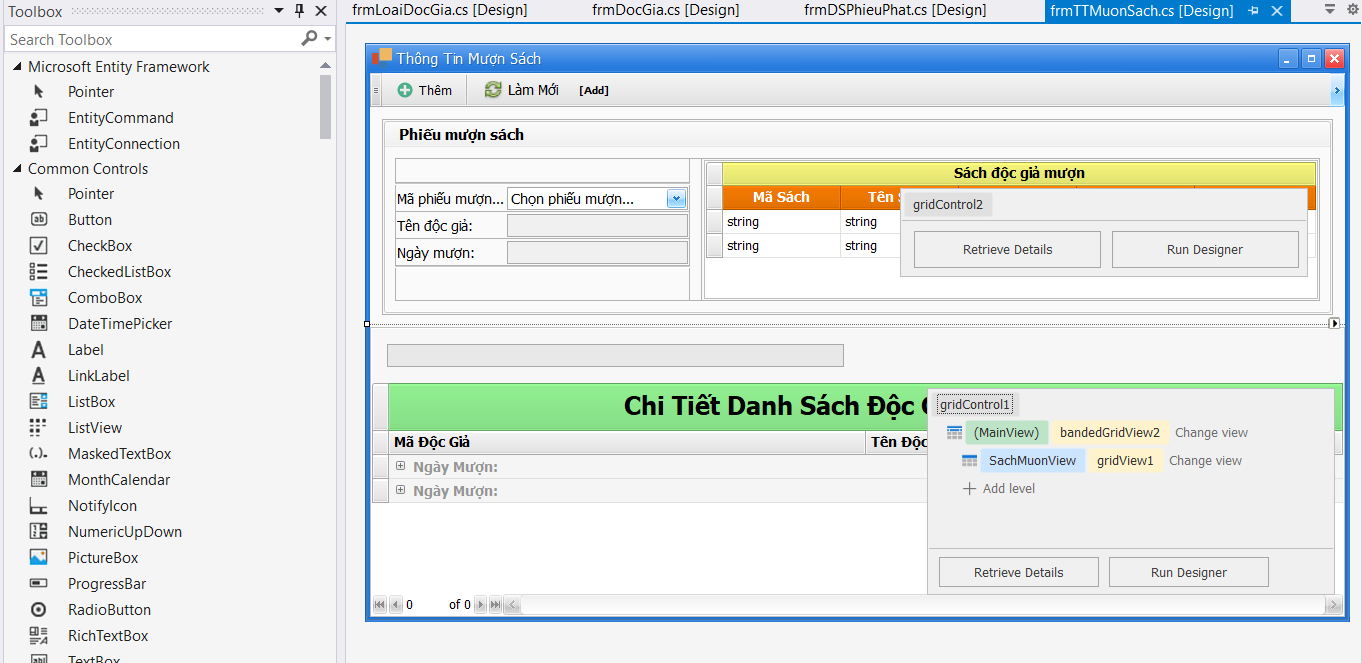
1. **Mô tả các đối tượng trên màn hình**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Tên | Kiểu | Ràng buộc | Chức năng |
| 1 | txtMaPP | Textbox |  | Hiển thị mã phiếu phạt. |
| 2 | cbMaDG | Combobox |  | Hiển thị/ chọn đọc giả. |
| 3 | gridPhat | gridControl |  | Hiển thị/ tra cứu danh sách phiếuphạt. |
| 4 | txtTenDG | Textbox |  | Hiển thị tên đọc giả. |
| 5 | txtSoTienThu | Textbox |  | Hiển thị/ thêm số tiền thu. |
| 6 | txtConlai | Textbox |  | Hiển thị số tiền còn lại. |
| 7 | txtMaNV | textbox |  | Hiển thị mã nhân viên. |

1. **Danh sách các biến cố và xử lý tương ứng trên màn hình**

|  |  |  |
| --- | --- | --- |
| STT | Biến cố | Xử lý |
| 1 | Chọn button Thêm | Xóa các giá trị trong textbox, bắt đầu thêm 1 phiếu phạt. |
| 2 | Chọn button Lưu | Lưu lại thay đổi từ button Thêm. |
| 3 | Chọn button Xóa | Xóa phiếu phạt được chọn. |

1. **Màn hình Phiếu mượn sách**
2. **Giao diện**



Hình 3.13: Giao diện mượn sách

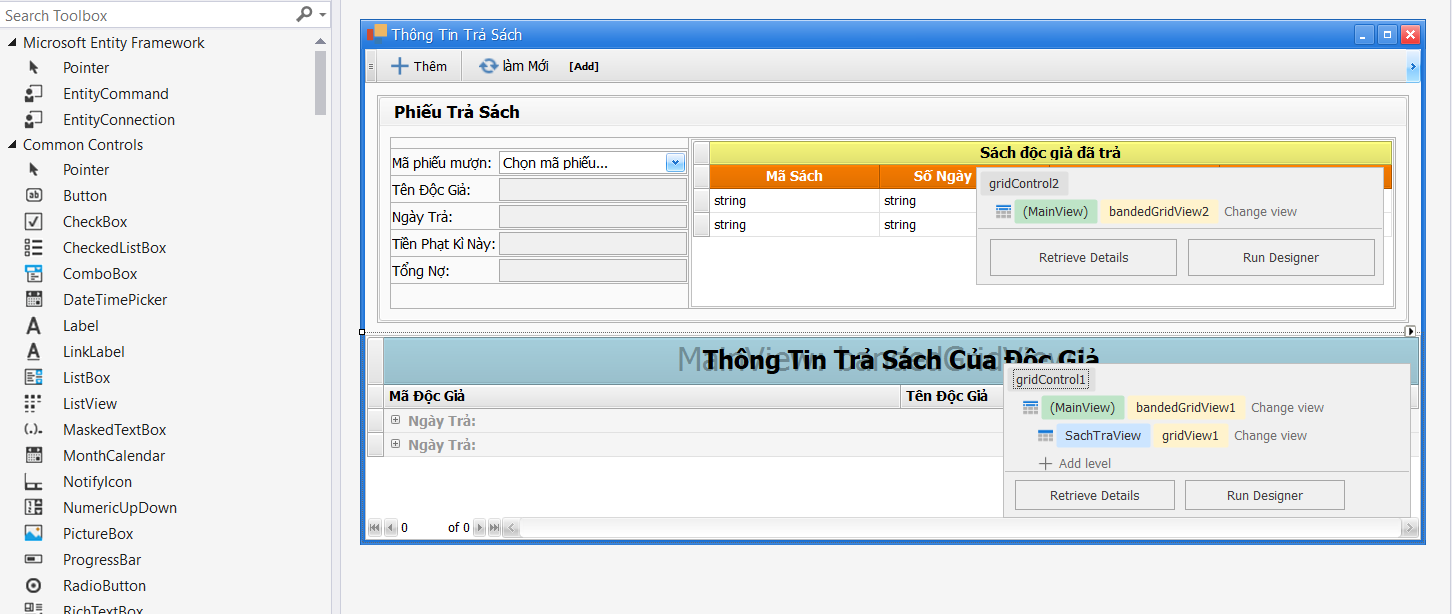
1. **Mô tả các đối tượng trên màn hình**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Tên | Kiểu | Ràng buộc | Chức năng |
| 1 | txtMaPM | Textbox |  | Hiển thị mã phiếu mượn |
| 2 | cbMaDG | Combobox |  | Hiển thị/ chọn đọc giả. |
| 3 | gridMuon | gridControl |  | Hiển thị/ tra cứu danh sách phiếu  mượn. |
| 4 | txtTenDG | Textbox |  | Hiển thị tên đọc giả. |
| 5 | dtNgayMuon | Datetimepicker |  | Hiển thị/ nhập ngày mượn. |
| 6 | dtHanTra | Datetimepicker |  | Hiển thị/ nhập ngày trả. |
| 7 | txtMaNV | Textbox |  | Hiển thị mã nhân viên. |
| 8 | gridCTMuon | gridControl |  | Hiển thị/ tra cứu danh sách sách trong  phiếu mượn. |

1. **Danh sách các biến cố và xử lý tương ứng trên màn hình**

|  |  |  |
| --- | --- | --- |
| STT | Biến cố | Xử lý |
| 1 | Chọn button Thêm | Xóa các giá trị trong textbox, bắt đầu thêm 1 phiếu mượn. |
| 2 | Chọn button Lưu | Lưu lại thay đổi từ button Thêm. |
| 3 | Chọn button Xóa | Xóa phiếu mượn được chọn. |
| 4 | Chọn button Thêm sách | Thêm sách đã chọn vào chi tiết phiếu mượn. |
| 5 | Chọn button Xóa sách | Xóa sách đã chọn ra khỏi chi tiết phiếu  mượn. |

1. **Màn hình Phiếu trả sách**
2. **Giao diện**



Hình 3.14: Giao diện thiết kế frmTTTraSach

1. **Mô tả các đối tượng trên màn hình**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Tên | Kiểu | Ràng buộc | Chức năng |
| 1 | txtMaPT | Textbox |  | Hiển thị mã phiếu trả |
| 2 | cbMaDG | Combobox |  | Hiển thị/ chọn đọc giả. |
| 3 | gridTra | gridControl |  | Hiển thị/ tra cứu danh sách phiếu trả. |
| 4 | txtTenDG | Textbox |  | Hiển thị tên đọc giả. |
| 5 | dtNgayTra | Datetimepicker |  | Hiển thị/ nhập ngày trả. |
| 6 | txtMaNV | Textbox |  | Hiển thị mã nhân viên. |
| 7 | gridCTTra | gridControl |  | Hiển thị/ tra cứu danh sách sách trong phiếu trả. |
| 8 | cbMaPM | Combobox |  | Hiển thị mã phiếu mượn. |

1. **Danh sách các biến cố và xử lý tương ứng trên màn hình**

|  |  |  |
| --- | --- | --- |
| STT | Biến cố | Xử lý |
| 1 | Chọn button Thêm | Xóa các giá trị trong textbox, bắt đầu thêm 1 phiếu trả. |
| 2 | Chọn button Lưu | Lưu lại thay đổi từ button Thêm. |
| 3 | Chọn button Xóa | Xóa phiếu trả được chọn. |
| 4 | Chọn button Thêm sách | Thêm sách đã chọn vào chi tiết phiếu trả. |
| 5 | Chọn button Xóa sách | Xóa sách đã chọn ra khỏi chi tiết phiếu trả. |

1. **Hoàn thiện logic**
2. **Chức năng phân quyền**

Khi người dùng đăng nhập vào phần mềm thành công, để thực hiện load các thành phần trong trang chính cần kiểm tra vai trò của của của người sử dụng.

void KiemTraPermisson()

{

account = ACCOUNTBUS.Instance.GetInfoAccount(idaccount).ToList()[0];

ShowNameAccount();

List<string> list = new List<string>();

list = CHITIETPHANQUYENBUS.Instance.GetListPerOfAccount(idaccount).ToList();

foreach (var item in list)

{

listper.Add(PERMISSIONBUS.Instance.GetInfoPermission(item)[0].CodeAction.ToString());

}

VisibleTab();

foreach (var item in listper)

{

switch (item)

{

case "DG":

VisibleTab();

return;

case "QL":

ShowTab();

return;

case "QLDG":

tab\_docgia.Visible = true;

break;

case "QLS":

tab\_sach.Visible = true;

break;

case "QLMT":

tab\_muontra.Visible = true;

break;

default:

break;

}

}

}

1. **Chức năng đăng nhập**

Khời tạo một danh sách listaccount có chưa thông tin đăng nhập. Sau đó lấy giá trị của người dùng nhập kiểm tra với listaccount nếu trùng thì cho đăng nhập thành công và chuyền dữ liệu của iduser load vào màn hình chính.

if (listaccount.Count == 0)

{

IOverlaySplashScreenHandle handle = SplashScreenManager.ShowOverlayForm(pl\_GiaoDien);

listaccount = /\*await\*/ ACCOUNTBUS.Instance.GetInfoAccount();

handle.Close();

}

for(int i=0;i<listaccount.Count;i++)

{

if(txbUsername.Text == listaccount[i].IDAccount && txbPassword.Text == listaccount[i].PasswordAccount)

{

IDUser = listaccount[i].IDAccount;

listaccount.Clear();

txbUsername.Select();

Hide();

SplashScreenManager.ShowDefaultSplashScreen("Đang mở...", "Phần Mềm Quản Lý Thư Viện");

Thread.Sleep(500);

FrmMain frmMain = new FrmMain(this, IDUser);

frmMain.Show();

SplashScreenManager.CloseDefaultSplashScreen();

return;

}

}

1. **Chức năng chuyển frame phụ**

Trong gia diện chính khi click vào frame khác sẽ chuyển sang kiểm tra frame đó đã được tồn tại. Sau đó chuyển giao diện xuống dưới frame chính với thuộc tính MdiParent và đóng giao diện hiện tại.

Form frm = this.KiemTraTonTai(typeof(frmSach));

if (frm != null)

frm.Activate();

else

{

SplashScreenManager.ShowDefaultWaitForm();

frmSach f = new frmSach

{

MdiParent = this

};

f.Show();

SplashScreenManager.CloseDefaultSplashScreen();

1. **Chức năng lấy dữ liệu lên bảng**

Sử dụng đối tượng db của lớp QuanLyThuVienEntities lấy dữ liệu từ bảng sách trong cơ sở dữ liệu trả về một danh sách sử dụng danh sách qua lớp Bus chuyển lên giao diện bảng để hiể thị

public List<SACH> GetListSachInfo()

{

List<SACH> listSachInfo = new List<SACH>();

using (QuanLyThuVienEntities db = new QuanLyThuVienEntities())

{

var data = ((from u in db.SACHes

select new { u.MaSach, u.MaTacGia, u.MaTheLoai, u.MaTinhTrang, u.NamXuatBan, u.NgayNhap, u.NhaXuatBan, u.TenSach, u.TriGia })).ToList();

foreach (var item in data)

{

SACH tts = new SACH();

tts.MaSach = item.MaSach;

tts.MaTacGia = item.MaTacGia;

tts.MaTheLoai = item.MaTheLoai;

tts.MaTinhTrang = item.MaTinhTrang;

tts.NamXuatBan = item.NamXuatBan;

tts.NgayNhap = item.NgayNhap;

tts.NhaXuatBan = item.NhaXuatBan;

tts.TriGia = item.TriGia;

tts.TenSach = item.TenSach;

listSachInfo.Add(tts);

}

}

return listSachInfo;

}

1. **Chức năng thêm sách**

public void AddInfoSach(string MaS, string TenS, string MaTL, int NamXB, string NhaXB, string MaTG, int TriGia, string MaTT)

{

using (QuanLyThuVienEntities db = new QuanLyThuVienEntities())

{

SACH s = new SACH

{

MaSach = MaS,

TenSach = TenS,

MaTheLoai = MaTL,

NamXuatBan = NamXB,

NhaXuatBan = NhaXB,

MaTacGia = MaTG,

TriGia = TriGia,

MaTinhTrang = MaTT

};

db.SACHes.Add(s);

db.SaveChanges();

}

}

1. **Chức năng lấy dữ liệu từ bảng lên ô text**

Sử dụng thư viện databinding tương tác dữ liệu và giao diện. Text được thay đổi khi thuộc tính ListBDSach được thay đổi. Vì vậy, textBox được cập nhật khi giá trị được thay đổi.

ListBDSach để load dữ liệu gridcontrol lên textbox không bị đơ khi thay đổi gridcontrol.

try

{

txbMaSach.DataBindings.Add("Text", ListBDSach, "MaSach", true, DataSourceUpdateMode.Never);

txbTenSach.DataBindings.Add("Text", ListBDSach, "TenSach", true, DataSourceUpdateMode.Never);

dateNgayNhap.DataBindings.Add("EditValue", ListBDSach, "NgayNhap", true, DataSourceUpdateMode.Never);

dateNamSX.DataBindings.Add("EditValue", ListBDSach, "NamXuatBan", true, DataSourceUpdateMode.Never);

glued\_TheLoai.DataBindings.Add("EditValue", ListBDSach, "MaTheLoai", true, DataSourceUpdateMode.Never);

txbNhaSX.DataBindings.Add("Text", ListBDSach, "NhaXuatBan",true, DataSourceUpdateMode.Never);

glued\_TacGia.DataBindings.Add("EditValue", ListBDSach, "MaTacGia", true, DataSourceUpdateMode.Never);

txbTriGia.DataBindings.Add("Text", ListBDSach, "TriGia", true, DataSourceUpdateMode.Never);

glued\_TinhTrang.DataBindings.Add("EditValue", ListBDSach, "MaTinhTrang", true, DataSourceUpdateMode.Never);

}

catch

{

}

1. **Lấy danh sách phiếu phạt**

void LoadDSPP()

{

layoutControl1.Visible = false;

btnThem.Enabled = true;

List <PHIEUTHUTIENPHAT> list = QLTV\_BUS.DSPHIEUPHATBUS.Instance.GetDSPhieuPhat();

gridControl1.DataSource = list.ToList();

try

{

PHIEUTHUTIENPHAT pttp = DSPHIEUPHATBUS.Instance.GetDSPhieuPhat()[DSPHIEUPHATBUS.Instance.GetDSPhieuPhat().Count - 1];

IdLast = pttp.MaPhieuThuTP;

}

catch { IdLast = ""; }

txb\_MaPhieuThu.Text = HelpGUI.AutoIncreaseID.Load\_AutoIncreaseID("PT", IdLast, 6);

glued\_DocGia.Properties.DataSource = null;

glued\_DocGia.EditValue = null;

btn\_OK.Enabled = false;

txb\_ConLai.EditValue = null;

txb\_TongNo.EditValue = null;

txb\_SoTienThu.EditValue = 500;

}

1. **Chức năng trả sách**

Khi người trả sách chương trình sẽ kiểm tra người dùng đã quá hạn trả sách hay chưa và tính tiền phát, tổng nợ hiện tại. Sau đó lưu trữ vào cơ sở dữ liệu.

bandedGridView1.MoveFirst();

decimal TienPhat = 0;

foreach (CHITIETPHIEUTRADTO item in list1)

{

TienPhat = 0;

if (DateTime.Now.Subtract(item.HanTra).Days > 0)

{

TienPhat = (decimal)DateTime.Now.Subtract(item.HanTra).Days \* THAMSOBUS.Instance.GetDSQuiDinh()[0].TPTraTreMotNgay;

}

CHITIETPHIEUTRABUS.Instance.AddPhieuTra(item.MaPhieuMS, item.MaSach, new DateTime(DateTime.Now.Year, DateTime.Now.Month, DateTime.Now.Day), DateTime.Now.Subtract(item.NgayMuon).Days, (decimal)DateTime.Now.Subtract(item.HanTra).Days \* THAMSOBUS.Instance.GetDSQuiDinh()[0].TPTraTreMotNgay, item.TinhTrangTraSach);

DOCGIABUS.Instance.UpdateTongNoDG(item.MaDocGia, TienPhat);

CTPHIEUMUONBUS.Instance.UpdateTinhTrangMuon(item.MaPhieuMS, item.MaSach, "Đã trả");

if (item.TinhTrangTraSach == "Hỏng")

{ SACHBUS.Instance.UpdateTinhtrangSach(item.MaSach, "TT003"); }

if (item.TinhTrangTraSach == "Mất")

{ SACHBUS.Instance.UpdateTinhtrangSach(item.MaSach, "TT004"); }

if (item.TinhTrangTraSach == "Bình thường")

{ SACHBUS.Instance.UpdateTinhtrangSach(item.MaSach, "TT001"); }

}

listtra.Clear();

listtra = new List<CHITIETPHIEUMUONDTO>();

//Load

listpm = CHITIETPHIEUMUONBUS.Instance.GetListPhieuMuon();

gridControl2.DataSource = new BindingList<CHITIETPHIEUMUONDTO>(listtra);

glued\_MaDocGia.Properties.DataSource = listpm.Where(x => x.TinhTrangMuon == "Còn hạn" || x.TinhTrangMuon == "Quá hạn").Select(x => new { x.MaDocGia, x.HoTen }).Distinct().ToList();

glued\_MaDocGia.Properties.ValueMember = "MaDocGia";

glued\_MaDocGia.Properties.DisplayMember = "MaDocGia";

dateEdit1.EditValue = new DateTime(DateTime.Now.Year, DateTime.Now.Month, DateTime.Now.Day);

1. **Chức năng mượn sách**

Trước khi mượn sách chương trình kiểm tra quy định trạng thái mượn sách quá số sách mượn sẽ không được mượn.Sau đó thêm thông tin sách mượn vào bảng phieumuon.

if(!dxErrorProvider1.HasErrors&&txbTenDocGia.EditValue!=null&&gridView2.RowCount>0)

{

if (Convert.ToInt32(seHanTra.EditValue) == 0)

{ XtraMessageBox.Show("Vui lòng chọn hạn trả!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Stop); }

else

{

if (XtraMessageBox.Show("Bạn đồng ý cho mượn sách", "Thông Báo", MessageBoxButtons.YesNo, MessageBoxIcon.Asterisk) == DialogResult.Yes)

{

int songaymuonmax = THAMSOBUS.Instance.GetDSQuiDinh().ToList()[0].SoNgayMuonMax;

int sosachmuonmax = THAMSOBUS.Instance.GetDSQuiDinh().ToList()[0].SoSachMuonMax;

if (KiemTraSoSachMuon(songaymuonmax, sosachmuonmax))

{

PHIEUMUONSACHBUS.Instance.AddPhieuMuon(txbMaPhieu.EditValue.ToString(), glued\_MaDocGia.EditValue.ToString(), Convert.ToDateTime(txbNgayMuon.EditValue));

LuuCTPHieuMuon();

this.Close();

}

else

{

XtraMessageBox.Show($"Bạn mượn tổng cộng {sosachmuontrongngay} sách trong {songaymuonmax} ngày\n\rKhông thỏa qui định: có {sosachmuonmax} sách mượn tối đa trong {songaymuonmax} ngày!",

"ERROR", MessageBoxButtons.OK, MessageBoxIcon.Error);

gridControl1.Focus();

}

}

}

}

else

{

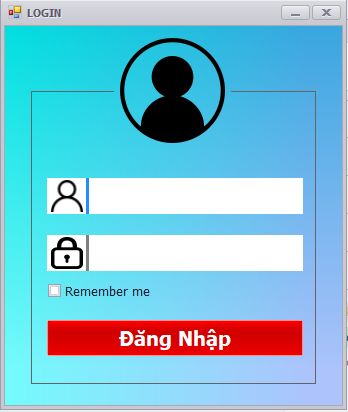
XtraMessageBox.Show("Có lỗi xảy ra! Vui lòng kiểm tra lại!", "ERROR", MessageBoxButtons.OK, MessageBoxIcon.Error);

glued\_MaDocGia.Focus();

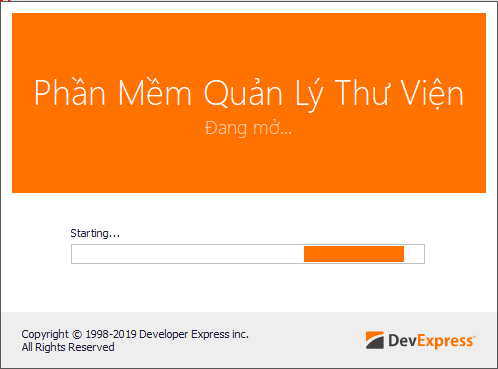
}

1. **Kết quả đạt được**

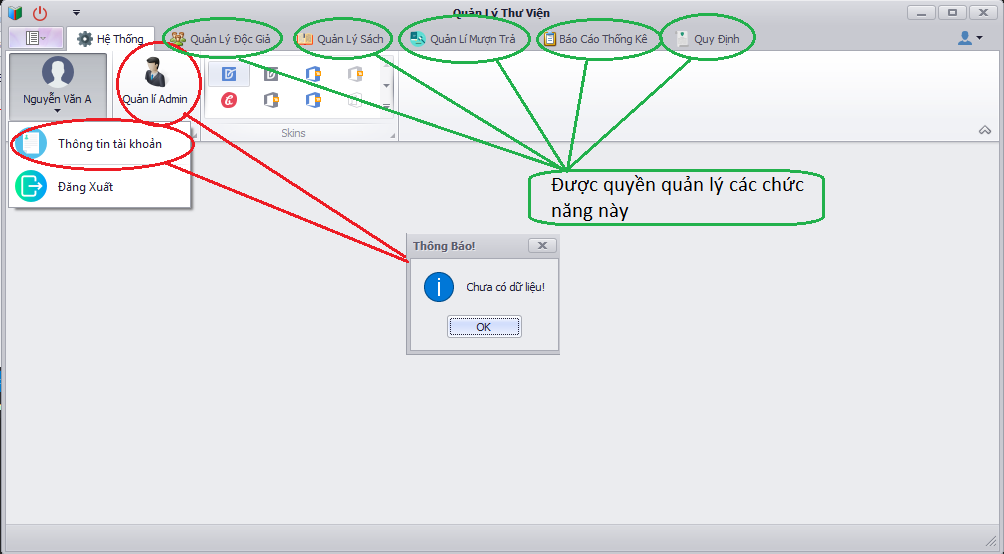
Để sử dụng phần cài đặt ứng dụng trên máy tính hệ điều hành Window .



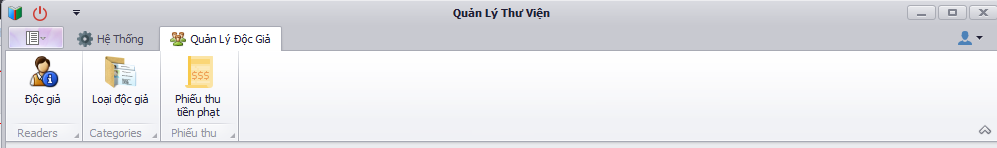
Hình 3.15: Giao diện đăng nhập



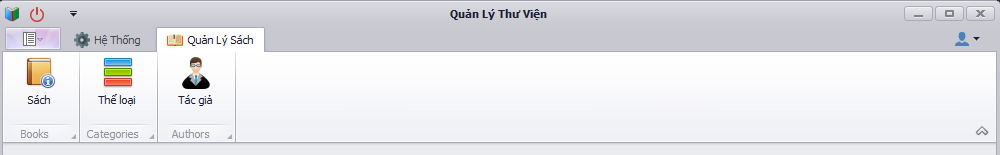
Hình 3.16:Giao diện load nếu đăng nhập thành công



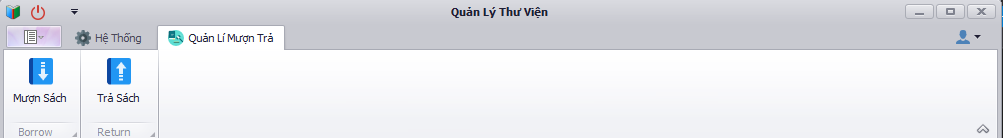
Hình 3.17: Giao diện admin(quyền cao nhất)



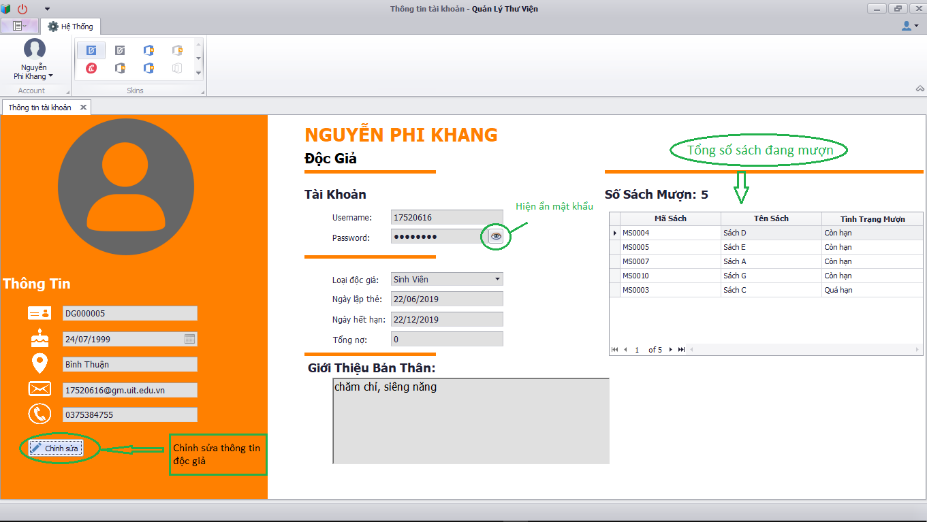
Hình 3.18: ADMIN (quyền quản lý độc giả)



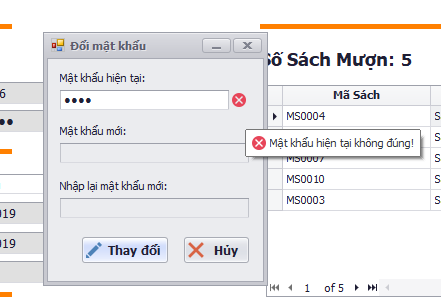
Hình 3.19: ADMIN (quyền quản lý sách)



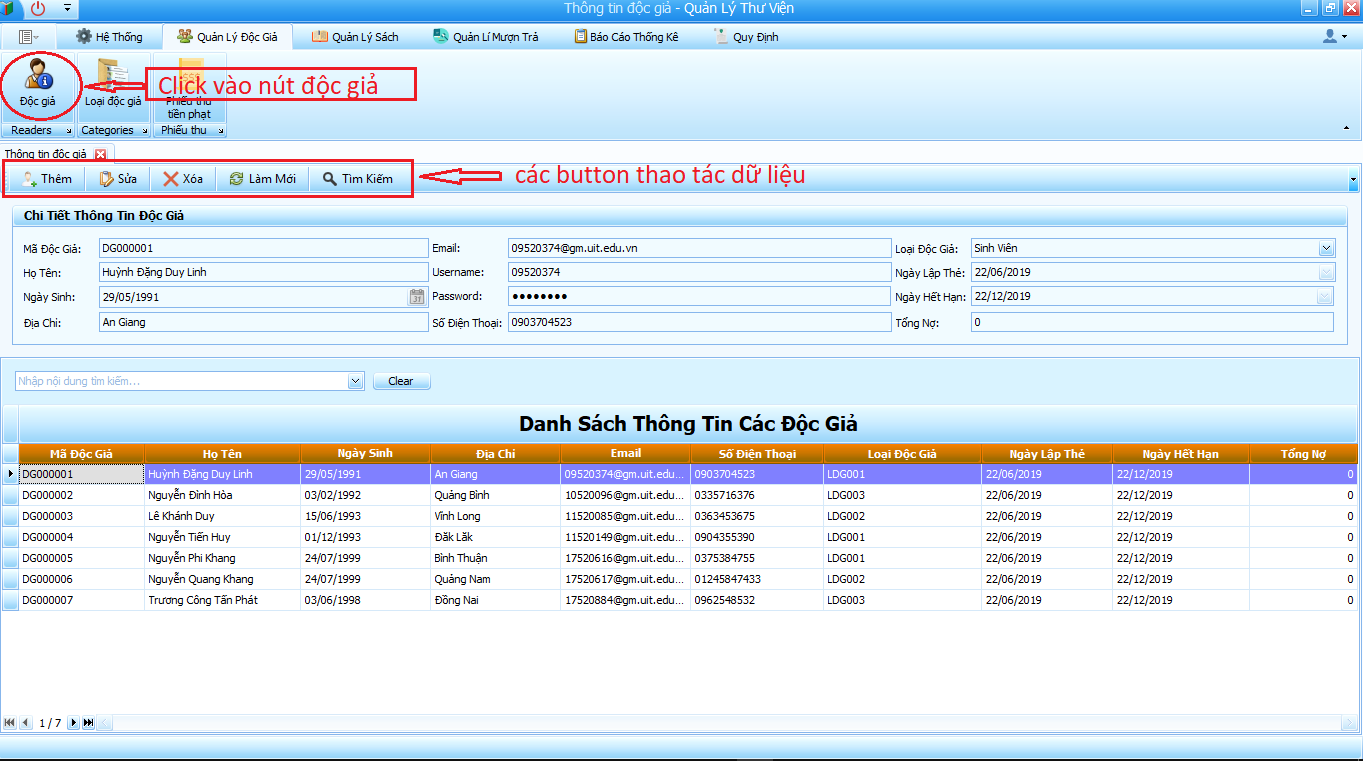
Hình 3.20: ADMIN (quyền quản lý mượn trả)



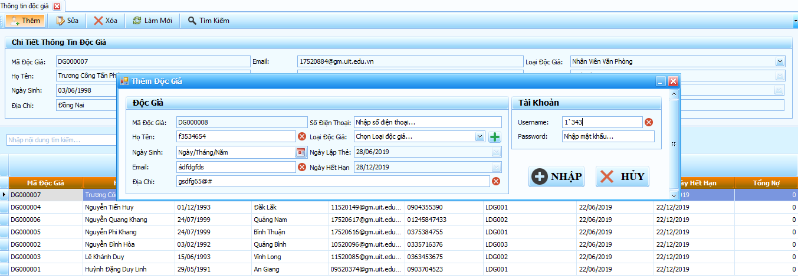
Hình 3.21: Giao diện của độc giả



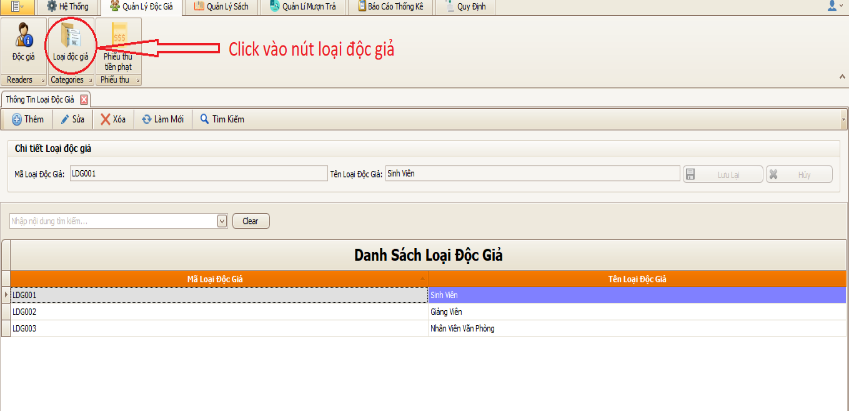
Hình 3.22: Giao diện đổi mật khẩu



Hình 3.23: Giao diện quản lí độc giả

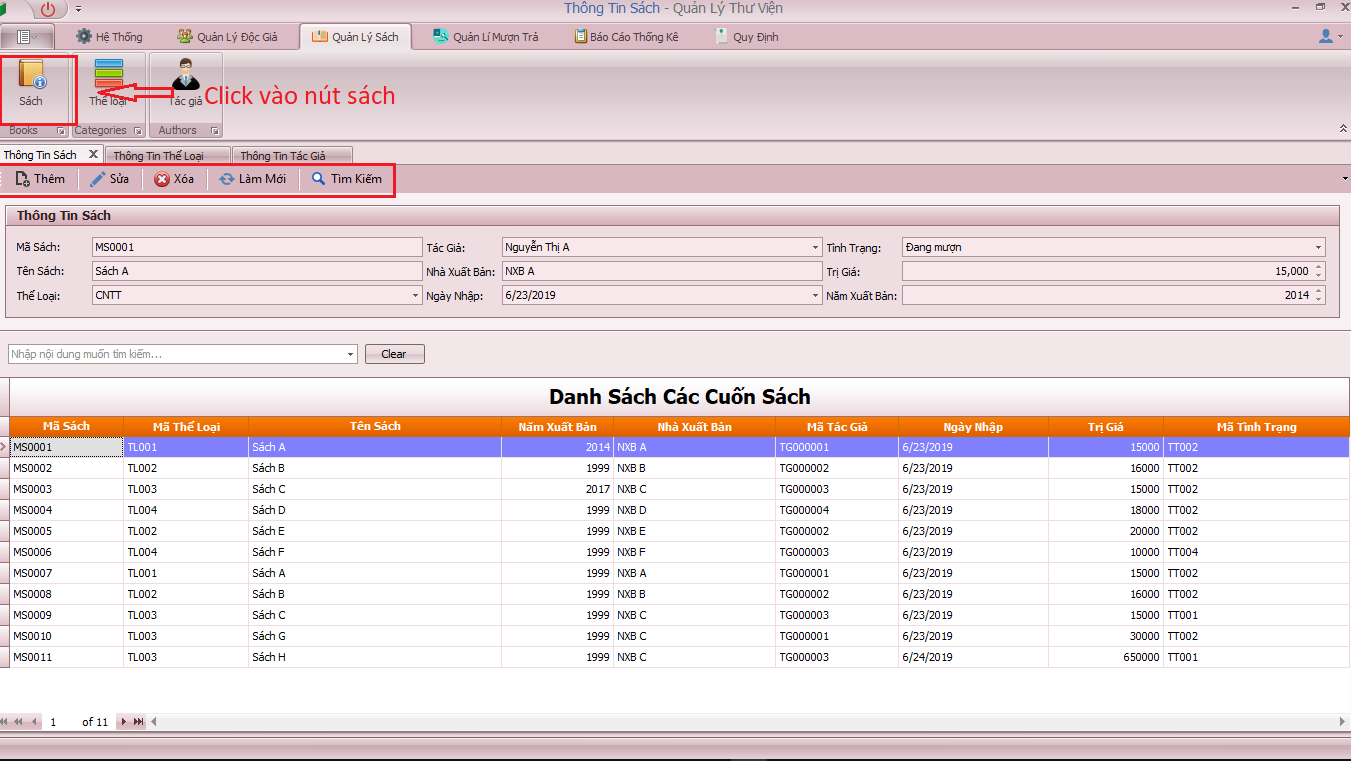


Hình 3.24: Giao diện thêm độc giả

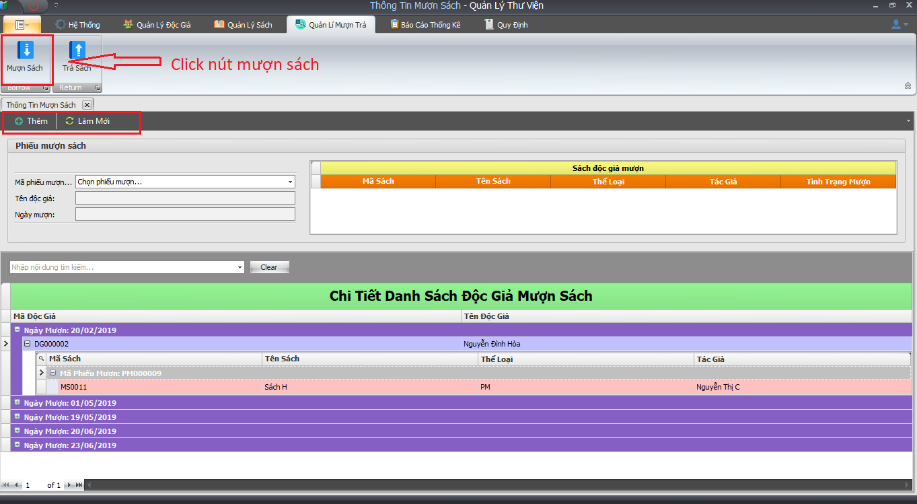


Hình 3.25: Giao diện loại độc giả

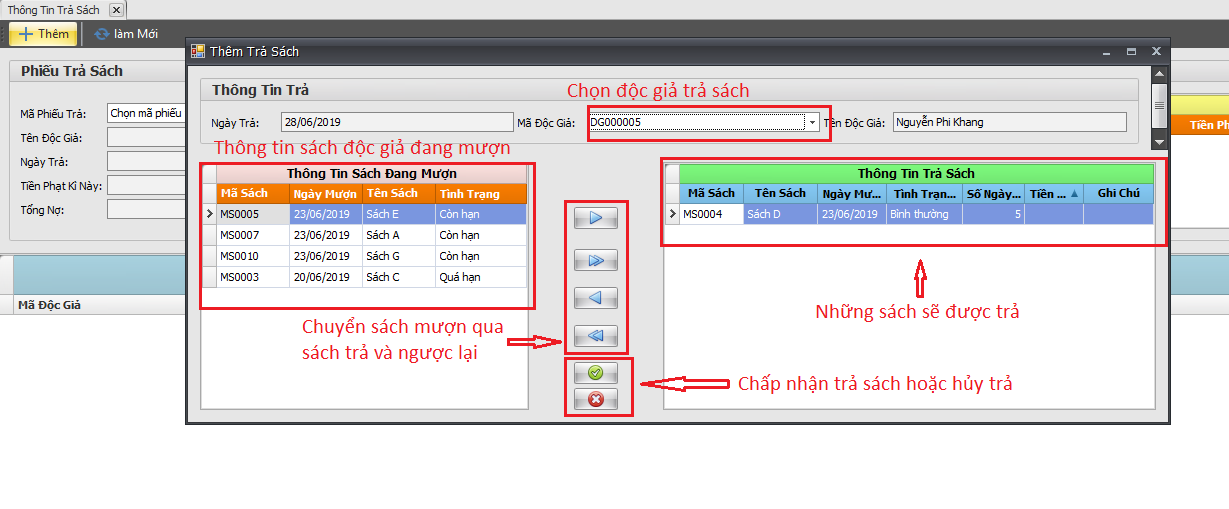
Hình 3.26: Giao diện thu tiền phạt



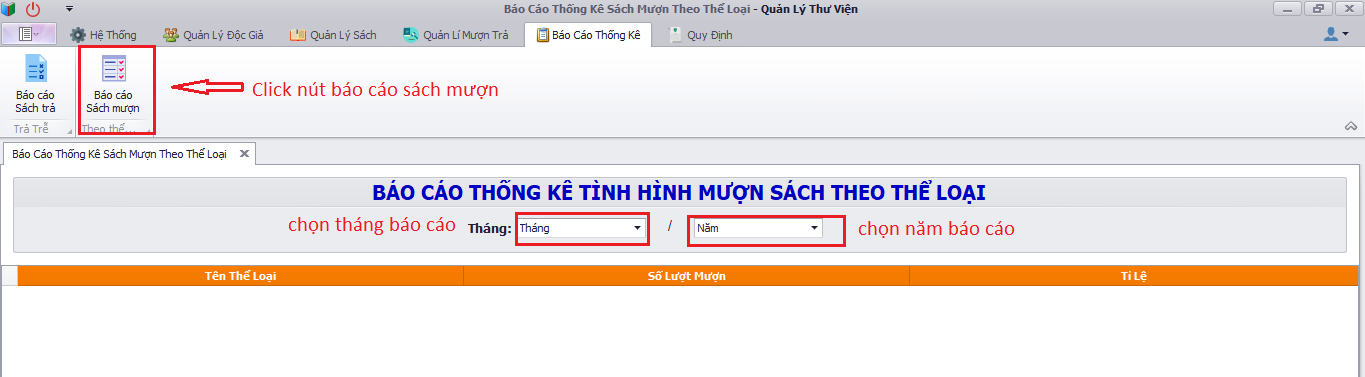
Hình 3.27: Giao diện quản lí sách



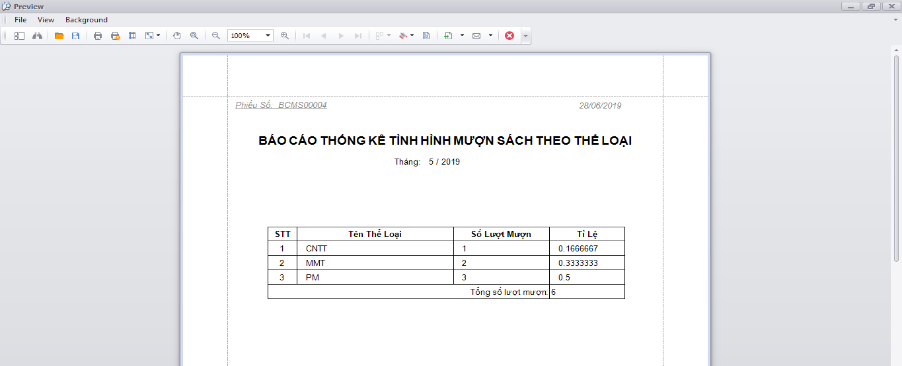
Hình 3.28: Giao diện quản lí mượn sách



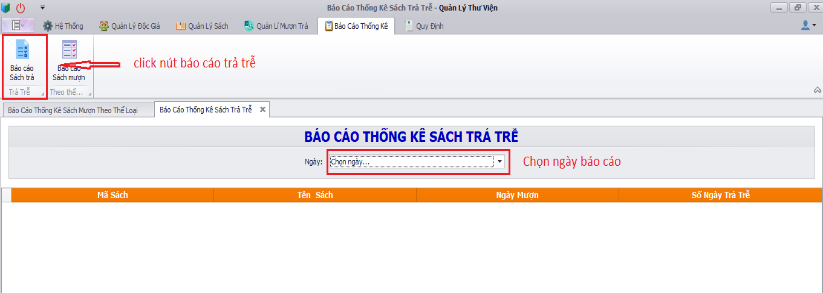
Hình 3.29: Giao diện trả sách



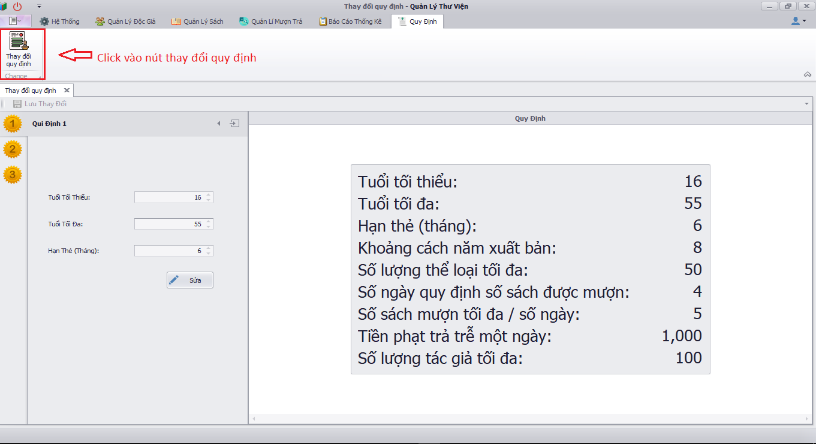
Hình 3.30: Giao diện thống kế sách mượn



Hình 3.31: Giao diện tạo báo cáo



Hình 3.32: Thống kê sách trả trễ



Hình 3.33: Giao diện quản lí quy định

**NHẬN XÉT VÀ KẾT LUẬN**

### Ưu điểm

* Độ tin cậy: Hệ thống có thể kiểm tra dữ liệu nhập vào và cho phép tự động thực hiện một số kiểm tra tự động đối chiếu và tìm những bản ghi có sự chênh lệch về dữ liệu, thông báo các số liệu nhập sai để có hướng khắc phục.
* Tính bảo mật: Hệ thống sử dụng phương pháp mã hoá dữ liệu lưu mật khẩu, người sử dụng chỉ có thể thay đổi dữ liệu sau khi đăng nhập vào hệ thống, có hỗ trợ thay đổi mật khẩu.
* Dễ sử dụng: Chương trình được thiết kế trên giao diện đồ họa với các cửa sổ, hộp thoại và hướng dẫn sử dụng hợp lý. Chương trình có hỗ trợ in ra file báo cáo.
* Tính mềm dẻo: Phần mềm có khả năng phát triển thêm như có thể mở rộng cho người dùng sử dụng file có sẵn để thêm danh sách học sinh vào hệ thống.
* Chương trình chỉ được triển khai trên người dùng là thủ thư và phải yêu cầu cài đặt SQL Server 2005 trở lên để sử dụng.

### Khuyết điểm

* Chưa tự động backup, restore dữ liệu.
* Chưa tự động hóa quá trình thu tiền phạt.
* Dữ liệu phải nhập bằng tay, không có thể lấy dữ liệu từ file tính ở ngoài.

### Kết quả đạt được

* Tạo ra được phần mềm quản lý thư viện với mô hình 3 lớp.
* Thực hiện được yêu cầu và quy định đã đặt ra từ ban đầu.
* Nếu kết nối sai CSDL thì có khả năng sẽ hiển thị sai

### Hướng phát triển

* Xây dựng thêm các chức năng chưa cài đặt.
* Kiểm tra ràng buộc dữ liệu chặt chẽ hơn ở chức năng cài đặt.
* Bổ sung thêm chức năng quản lý admin và thông tin admin.
* Phân quyền chi tiết từng chức năng hơn.
* Cải thiện giao diện, quản lý hệ thống lưu trữ dễ chuyển đổi.
* Đồng bộ hóa những thông tin giữa admin vs độc giả.

# **TÀI LIỆU THAM KHẢO**

1. **Tiếng Việt**

[1] Slide Bài giảng Link: <https://courses.uit.edu.vn/mod/folder/view.php?id=68196>

[2] Giáo trình Nhập môn công nghệ phần mềm

[3] Giáo trình C# Database Programming .

[4] SQL Server 2005 – 2008 – 2012 – 2014.

[5] Các bài học online trên youtube.

* <https://www.youtube.com/watch?v=Erk56uJcsj4>
* <https://www.youtube.com/watch?v=5jjKGiminpk>

1. **Tiếng Anh**

[1] Binding Class: <https://docs.microsoft.com/en-us/dotnet/api/system.windows.data.binding?redirectedfrom=MSDN&view=netframework-4.8>

[2] Connect sql to vs 2017: <https://stackoverflow.com/questions/43075071/error-connecting-to-a-remote-sql-server-database-visual-studio-2017>

[3] Devexpress: [https://documentation.devexpress.com/WindowsForms/114561/Controls-and- Libraries/Forms-and-User-Controls](https://documentation.devexpress.com/WindowsForms/114561/Controls-and-Libraries/Forms-and-User-Controls)

[4] EntityFramework: <https://www.entityframeworktutorial.net/what-is-entityframework.aspx>