|  |
| --- |
| **TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT TP. HCM**  **KHOA CÔNG NGHỆ THÔNG TIN**  ⧫⧫⧫⧫⧫    *Ts. Nguyễn Thành Sơn*  **Hướng dẫn đề tài**  **XÂY DỰNG PHẦN MỀM QUẢN LÝ**  **CỬA HÀNG BÁN SÁCH**  Báo cáo môn/nhóm:   211DBMS330284 – Hệ quản trị cơ sở dữ liệu – nhóm 01  DBMS Là Gì? Tầm Quan Trọng Của DBMS Với Doanh Nghiệp - Đề án 2020 - Tổng  Hợp Chia Sẻ Hình ảnh, Tranh Vẽ, Biểu Mẫu Trong Lĩnh Vực Giáo Dục  Nhóm số: 05  Học kỳ: I – Năm học: 2021 – 2022  Thành phố Hồ Chí Minh, tháng 11 năm 2021 |

**DANH SÁCH NHÓM THAM GIA**

**THỰC HIỆN ĐỀ TÀI VÀ VIẾT BÁO CÁO**

**Môn: Hệ quản trị cơ sở dữ liệu**

*HỌC KÌ I – NĂM HỌC 2021 – 2022*

*Nhóm số: 05*

*Tên đề tài: Xây dựng phần mềm quản lý cửa hàng bán sách*

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **HỌ VÀ TÊN** | **MSSV** | **SỐ ĐIỆN THOẠI** |
| 1 | Lê Thị Ngọc Diễm | 19110335 | 0936074295 |
| 2 | Nguyễn Thị Mỹ Linh (LD) | 19133032 | 0963577215 |
| 3 | Phạm Văn Thắng | 19110463 | 0868690043 |
| 4 | Lương Quốc Trung | 19110489 | 0935544787 |
| 5 | Đỗ Quốc Việt | 19110498 | 0393014871 |

***Nhận xét của giảng viên***

*Ngày 24 tháng 11 năm 2021*

*Giảng viên chấm điểm*

*Ts. Nguyễn Thành Sơn*

# LỜI MỞ ĐẦU

Ngày nay, Công nghệ thông tin đã và đang đóng vai trò quan trọng trong đời sống kinh tế, xã hội của nhiều quốc gia trên thế giới, là một phần không thể thiếu trong xã hội năng động và ngày càng hiện đại hoá. Công nghệ thông tin được ứng dụng vào hầu hết các lĩnh vực của đời sống và lĩnh vực hoạt động kinh doanh buôn bán không phải là một ngoại lệ. Việc ứng dụng công nghệ thông tin vào quản lý bán hàng không những tiết kiệm thời gian, tiết kiệm chi phí và nguồn nhân lực mà còn thể hiện được độ chính xác cao từ đó nâng cao hiệu quả công việc và tăng năng lực quản lý.

Xuất phát từ nhu cầu thực tế đó, trong công việc mua và bán sách, việc quản lý sách nhập và bán là một việc không thể thiếu. Nhằm thay thế việc quản lý bán hàng theo hình thức thô sơ, nhiều thủ tục, nhiều công đoạn,.. tại mỗi cửa hàng đạt hiệu quả không cao. Vì vậy, chúng em đã thực hiện báo cáo với đề tài “**Xây dựng hệ thống quản lý bán sách”.** Đây là một chương trình ứng dụng, do một bộ phận nhân viên trong nhà sách quản lý, với mục đích tiết kiệm thời gian và thuận tiện cho việc quản lý thông tin tại các nhà sách.

# LỜI CẢM ƠN

Lời đầu tiên, nhóm chúng em xin gửi lời cảm ơn đến thầy Nguyễn Thành Sơn – giảng viên bộ môn “Hệ quản trị cơ sở dữ liệu” trường Đại học Sư phạm Kỹ thuật thành phố Hồ Chí Minh đã tận tình giảng dạy, trang bị, cung cấp cho chúng em những kiến thức nền tảng, chuyên môn cần thiết giúp nhóm em hiểu sâu về lĩnh vực nghiên cứu và hoàn thành được đồ án này.

Tuy nhiên trong quá trình nghiên cứu đề tài, mặc dù đã cố gắng hoàn thành tốt công việc được phân công trong phạm vi và khả năng cho phép nhưng chắc chắn không tránh khỏi những thiếu sót. Rất mong nhận được sự góp ý của thầy để chương trình được hoàn chỉnh hơn, đồng thời bổ sung vốn kinh nghiệm cho nhóm trên con đường sắp tới.

Nhóm em xin chân thành cảm ơn!

*Tp. HCM, ngày 22 tháng 11 năm 2021*

*Nhóm Trưởng*

*My Linh*

*Nguyễn Thị Mỹ Linh*

**DANH MỤC CÁC TỪ VIẾT TẮT**

|  |  |
| --- | --- |
| **CÁC TỪ VIẾT TẮT** | **VIẾT ĐẦY ĐỦ** |
| CSDL | Cơ sở dữ liệu |

**MỤC LỤC**

[LỜI MỞ ĐẦU](#_heading=h.319y80a)

[LỜI CẢM ƠN](#_heading=h.1gf8i83)

[TỔNG QUAN VỀ ĐỀ TÀI 1](#_heading=h.40ew0vw)

[1. Giới thiệu đề tài 1](#_heading=h.2fk6b3p)

[2. Mục tiêu của đề tài 1](#_heading=h.upglbi)

[3. Các chức năng chính của đề tài 2](#_heading=h.3ep43zb)

[4. Mô tả giao diện đề tài 2](#_heading=h.1tuee74)

[5. Công nghệ sử dụng 2](#_heading=h.4du1wux)

[NỘI DUNG 3](#_heading=h.2szc72q)

[Chương 1: CƠ SỞ LÝ THUYẾT 3](#_heading=h.184mhaj)

[1.1. Sơ lược về cơ sở dữ liệu và SQL Server 3](#_heading=h.3s49zyc)

[1.1.1. Khái quát cở sở dữ liệu 3](#_heading=h.279ka65)

[1.1.2. Hệ quản trị cơ sở dữ liệu 4](#_heading=h.meukdy)

[1.1.3. Sơ lược về SQL Server 5](#_heading=h.36ei31r)

[1.2. Sơ lược về Winforms, ADO.NET trong C# 7](#_heading=h.1ljsd9k)

[1.2.1. Lập trình Winforms trong C# 7](#_heading=h.45jfvxd)

[1.2.2. ADO.NET trong C# 8](#_heading=h.2koq656)

[1.3. Sơ lược về mô hình ba lớp 9](#_heading=h.zu0gcz)

[1.3.1. Khái niệm 9](#_heading=h.3jtnz0s)

[1.3.2. Các thành phần chính 9](#_heading=h.1yyy98l)

[Chương 2: PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG 11](#_heading=h.4iylrwe)

[2.1. Các yêu cầu cụ thể 11](#_heading=h.2y3w247)

[2.1.1. Đối với người quản lý 11](#_heading=h.1d96cc0)

[2.1.2. Đối với nhân viên 11](#_heading=h.3x8tuzt)

[2.1.3. Đối với khách hàng 11](#_heading=h.2ce457m)

[2.2. Mô tả các chức năng của hệ thống 11](#_heading=h.rjefff)

[2.2.1. Chức năng Quản lý nhân viên 11](#_heading=h.3bj1y38)

[2.2.2. Chức năng Quản lý khách hàng thành viên 12](#_heading=h.1qoc8b1)

[2.2.3. Chức năng Quản lý kho sách 12](#_heading=h.4anzqyu)

[2.2.4. Chức năng Quản lý mã giảm giá 12](#_heading=h.2pta16n)

[2.2.5. Chức năng Quản lý hóa đơn 12](#_heading=h.14ykbeg)

[2.2.6. Chức năng Quản lý phiếu nhập 12](#_heading=h.3oy7u29)

[2.2.7. Chức năng Quản lý thu nhập 12](#_heading=h.243i4a2)

[2.2.8. Chức năng lập hóa đơn bán hàng 12](#_heading=h.j8sehv)

[2.3. Thiết kế cơ sở dữ liệu 12](#_heading=h.338fx5o)

[2.3.1. Xây dựng các thực thể 12](#_heading=h.1idq7dh)

[2.3.2. Mối quan hệ giữa các thực thể 13](#_heading=h.42ddq1a)

[2.3.3. Xây dựng mô hình ERD 14](#_heading=h.2hio093)

[2.3.4. Chuyển từ mô hình ERD sang mô hình quan hệ 14](#_heading=h.wnyagw)

[2.3.5. Mô hình Diagrams 15](#_heading=h.3gnlt4p)

[2.3.6. Chi tiết từng thực thể 15](#_heading=h.1vsw3ci)

[Chương 3: CÀI ĐẶT HỆ THỐNG 20](#_heading=h.2uxtw84)

[3.1. Cài đặt cơ sở dữ liệu và tạo các ràng buộc khóa chính, khóa ngoại và not null 20](#_heading=h.1a346fx)

[3.2. Cài đặt Trigger 26](#_heading=h.3u2rp3q)

[3.2.1. Trigger kiểm tra số lượng trên hóa đơn nhỏ hơn số lượng tồn kho 26](#_heading=h.2981zbj)

[3.2.2. Trigger cập nhật số lượng sách trong bảng Sách sau khi bán sách 26](#_heading=h.odc9jc)

[3.2.3. Trigger cập nhật sách trong trong bảng Sách sau khi cập nhật bán sách 26](#_heading=h.38czs75)

[3.2.4. Trigger kiểm tra số lượng sách nhập phải lớn hơn 0 27](#_heading=h.1nia2ey)

[3.2.5. Trigger cập nhật số lượng sách trong bảng Sách sau khi nhập sách 27](#_heading=h.47hxl2r)

[3.2.6. Trigger cập nhật sách trong trong bảng Sách sau khi cập nhật nhập sách 28](#_heading=h.2mn7vak)

[3.2.7. Kiểm tra ngày bắt đầu giảm giá phải trước ngày kết thúc giảm giá 28](#_heading=h.11si5id)

[3.2.8. Trigger kiểm tra phần trăm giảm nhỏ hơn 100 28](#_heading=h.3ls5o66)

[3.3. Cài đặt các Views 29](#_heading=h.20xfydz)

[3.3.1. View xem Chi tiết hóa đơn 29](#_heading=h.4kx3h1s)

[3.3.2. View xem Chi tiết phiếu nhập 29](#_heading=h.302dr9l)

[3.3.3. View lấy ra top 20 cuốn sách bán chạy nhất 29](#_heading=h.1f7o1he)

[3.3.4. View xem số lần mua hàng của mỗi thành viên 29](#_heading=h.3z7bk57)

[3.3.5. View lấy ra top 20 thành viên có giá trị mua hàng lớn nhất 30](#_heading=h.2eclud0)

[3.3.6. View xem lương nhân viên 30](#_heading=h.thw4kt)

[3.4. Cài đặt các Stored Procedures 30](#_heading=h.3dhjn8m)

[3.4.1. Stored Procedure xem, thêm, xóa, sửa dữ liêu các bảng 30](#_heading=h.1smtxgf)

[3.4.2. Stored procedure tìm kiếm nhân viên 32](#_heading=h.4cmhg48)

[3.4.3. Stored procedure tìm kiếm loại sách 33](#_heading=h.2rrrqc1)

[3.4.4. Stored procedure tìm kiếm sách 33](#_heading=h.16x20ju)

[3.4.5. Stored procedure lấy thông tin cho hóa đơn bán hàng 34](#_heading=h.3qwpj7n)

[3.4.6. Stored procedure tìm hóa đơn 36](#_heading=h.261ztfg)

[3.4.7. Stored procedure tìm phiếu nhập 37](#_heading=h.l7a3n9)

[3.5. Cài đặt các Functions 37](#_heading=h.356xmb2)

[3.5.1. Function tìm kiếm ID nhân viên 37](#_heading=h.1kc7wiv)

[3.5.2. Function tìm kiếm tên nhân viên 37](#_heading=h.44bvf6o)

[3.5.3. Function tìm kiếm username của nhân viên 37](#_heading=h.2jh5peh)

[3.5.4. Function Check Login 38](#_heading=h.ymfzma)

[3.5.5. Function lấy Password bằng ID 38](#_heading=h.3im3ia3)

[3.5.6. Function tính tổng doanh thu theo tháng 38](#_heading=h.1xrdshw)

[3.5.7. Funtion tính doanh thu đối với mỗi thành viên 38](#_heading=h.4hr1b5p)

[3.5.8. Funtion tính số lượng sách bán được 39](#_heading=h.2wwbldi)

[3.5.9. Function xem chi tiết hóa đơn 39](#_heading=h.1c1lvlb)

[3.5.10. Function xem chi tiết phiếu nhập 39](#_heading=h.3w19e94)

[3.6. Tạo User và phân quyền 39](#_heading=h.2b6jogx)

[3.6.1. Bảng cơ sở dữ liệu dùng để phân quyền 39](#_heading=h.qbtyoq)

[3.6.2. Phân quyền cho các user 40](#_heading=h.3abhhcj)

[Chương 4: THIẾT KẾ GIAO DIỆN 44](#_heading=h.1pgrrkc)

[KẾT LUẬN 50](#_heading=h.49gfa85)

[1. Kết quả đạt được 50](#_heading=h.2olpkfy)

[2. Hạn chế 50](#_heading=h.13qzunr)

[3. Hướng phát triển 50](#_heading=h.3nqndbk)

[TÀI LIỆU THAM KHẢO 51](#_heading=h.22vxnjd)

[PHỤ LỤC 52](#_heading=h.i17xr6)

[Hướng dẫn sử dụng chương trình 52](#_heading=h.320vgez)

# TỔNG QUAN VỀ ĐỀ TÀI

## 1. Giới thiệu đề tài

Một cửa hàng bán sách cần làm các công việc như quản lý nhân viên, quản lý sách, khách hàng thành viên, mối quan hệ với nhà cung cấp, đều chỉ mang cái nhìn tổng quát về công việc kinh doanh của cửa hàng. Người quản lý cửa hàng rất khó khăn trong việc nắm bắt các thông tin về nhà cung cấp, khách hàng, số lượng sách đã bán được, số lượng sách còn tồn trong kho, loại sách nào bán chạy để có thể đưa ra biện pháp kinh doanh kịp thời. Mỗi khi muốn thống kê, báo cáo về hoạt động kinh doanh của cửa hàng, các nhân viên lại phải tìm tòi, lùng sục từng hóa đơn, sổ sách mất nhiều thời gian, vì vậy người nhà quản lý khó nắm bắt được hoạt động kinh doanh thường xuyên của cửa hàng mình dẫn đến sự chậm trễ trong việc đưa ra chiến lược kinh doanh.

Bên cạnh đó sự phát triển không ngừng của công nghệ thông tin, việc ứng dụng công nghệ thông tin vào việc quản lý sẽ giúp cửa hàng trở nên dễ dàng và tiện lợi hơn rất nhiều so với việc làm thủ công bằng tay và giấy tờ.

## 2. Mục tiêu của đề tài

Xây dựng một hệ thống hữu hiệu cho các cửa hàng nói chung có thể quản lý hàng hóa một cách đơn giản và hiệu quả hơn;

Xây dựng một hệ thống quản lý đơn giản, nhanh gọn, chính xác và tốn ít thời gian, công sức nhất;

Các mục tiêu quan trọng của đề tài:

* Tính chính xác: Mọi thông tin của mọi cuốn sách phải đảm bảo chính xác 100% để không ảnh hưởng đến kết số lượng sách, giá sách....
* Tính nhanh gọn: việc tìm thông tin cũng như xử lý thông tin phải nhanh gọn, tiết kiệm thời gian
* Khả năng tìm kiếm, chỉnh sửa: tìm kiếm phải đạt tốc độ cao. Chính sửa đơn giản, hiệu quả. Giao diện đẹp, dễ thực hiện.
* Tính bảo mật: Độ bảo mật về thông tin cá nhân của phải cao. Hệ thống phải có hành vi ngăn chặn khi có sự xâm nhập bất hợp pháp từ bên ngoài.
* Tính ổn định: Hệ thống phải làm việc trơn chu, tránh sai sót về mặt dữ liệu và phải chiếm ít tài nguyên của máy.

## 3. Các chức năng chính của đề tài

Chương trình gồm có các chức năng chính:

- Quản lý nhân viên;

- Quản lý khách hàng thành viên;

- Quản lý kho sách;

- Quản lý mã giảm giá;

- Quản lý hóa đơn;

- Quản lý phiếu nhập;

- Quản lý thu nhập;

- Lập hóa đơn bán hàng;

## 4. Mô tả giao diện đề tài

- Giao diện đăng nhập: Gồm có Username và Password. Người dùng (người quản lý, nhân viên) khi muốn đăng nhập vào hệ thống cần có một tài khoản.

- Giao diện cho nhân viên: Gồm các from Lập hóa đơn bán hàng, Quản lý khách hàng thành viên, Quản lý kho sách, Quản lý mã giảm giá, Quản lý hóa đơn, Quản lý phiếu nhập.

- Giao diện cho người quản lý: Ngoài các from Lập hóa đơn bán hàng, Quản lý khách hàng thành viên, Quản lý kho sách, Quản lý mã giảm giá, Quản lý hóa đơn, Quản lý phiếu nhập giống như của nhân viên thì còn có from Quản lý nhân viên và Quản lý thu nhập.

## 5. Công nghệ sử dụng

Microsoft SQL Server Management Studio 18

ADO.NET và Windows Forms Application trên Microsoft Visual Studio 2019.

# NỘI DUNG

# Chương 1: CƠ SỞ LÝ THUYẾT

## 1.1. Sơ lược về cơ sở dữ liệu và SQL Server

### 1.1.1. Khái quát cở sở dữ liệu

Một CSDL (Database) là một tập hợp các dữ liệu có liên quan với nhau, chứa thông tin của một tổ chức nào đó (như một trường học, một ngân hàng, một công ty, một nhà máy,…) được lưu trữ trên các thiết bị nhớ để đáp ứng nhu cầu khai thác thông tin của nhiều người sử dụng với nhiều mục đích khác nhau.

#### 1.1.1.1. Các mức thể hiện của cở sở dữ liệu

Mức vật lý: CSDL vật lý của một hệ CSDL là tập hợp các tệp dữ liệu, tồn tại trên các thiết bị nhớ.

Mức khái niệm: Những người quản trị hệ CSDL không cần hiểu chi tiết ở mức vật lí nhưng họ cần phải biết những dữ liệu nào được lưu trong CSDL. Giữa những dữ liệu có các mối liên hệ nào,… CSDL có thể mô tả như một bảng, mỗi cột là một thuộc tính, mỗi hàng tương ứng với dữ liệu về một chủ thể. Mức hiểu CSDL như vậy gọi là mức khái niệm

Mức khung nhìn: Khi khai thác CSDL, một người dùng có thể không quan tâm đến toàn bộ thông tin chứa trong CSDL mà chỉ cần một phần thông tin nào đó phù hợp với nghiệp vụ hay mục đích sử dụng của mình. Thể hiện phù hợp của CSDL cho mỗi người dùng được coi là một khung nhìn của CSDL. Mức hiểu CSDL của người dùng thông qua khung nhìn được gọi là mức khung nhìn.

#### 1.1.1.2. Các yêu cầu cơ bản của cở sở dữ liệu

CSDL phải đảm bảo các tính chất sau:

- Tính cấu trúc: Dữ liệu trong CSDL được lưu trữ theo một cấu trúc xác định. Ví dụ, CSDL lớp có cấu trúc là bảng 50 dòng, 10 cột. Mỗi cột là một thuộc tính và mỗi dòng là một hồ sơ học sinh.

- Tính toàn vẹn: Các giá trị dữ liệu được lưu trữ trong CSDL phải thỏa mãn một số ràng buộc, tùy thuộc vào hoạt động của tổ chức mà CSDL phản ánh.

- Tính nhất quán: Sau những thao tác cập nhật dữ liệu và ngay cả khi có sự cố xảy ra (do phần cứng hay phần mềm) trong quá trình cập nhật, dữ liệu trong CSDL phải luôn đảm bảo tính đúng đắn.

- Tính an toàn và bảo mật thông tin: CSDL vẫn được bảo vệ an toàn, phải ngăn chặn được những truy xuất không được phép và phải khôi phục được CSDL khi có sự cố ở phần cứng hay phần mềm.

- Tính độc lập: Vì một CSDL phải phục vụ cho nhiều mục đích khai thác khác nhau nên dữ liệu phải độc lập với các ứng dụng, không phụ thuộc vào một vài bài toán cụ thể.

- Tính không dư thừa: CSDL thường không lưu trữ những dữ liệu trùng lặp hoặc những thông tin có thể dễ dàng suy diễn hay tính toán được từ những dữ liệu đã có. Sự trùng lặp thông tin vừa làm lãng phí bộ nhớ để lưu trữ vừa dễ dẫn đến tình trạng không nhất quán thông tin.

### 1.1.2. Hệ quản trị cơ sở dữ liệu

#### 1.1.2.1. Định nghĩa Hệ quản trị cơ sở dữ liệu

Hệ quản trị cơ sở dữ liệu (Database Management System - DBMS): Là một hệ thống phần mềm cho phép tạo lập cơ sở dữ liệu và điều khiển mọi truy cập đối với cơ sở dữ liệu đó.

Trên thị trường phần mềm hiện nay ở Việt Nam đã xuất hiện khá nhiều phần mềm hệ quản trị cơ sở dữ liệu như: Microsoft Access, Foxpro, DB2, SQL Server, Oracle,…

Hệ quản trị cơ sở dữ liệu quan hệ (Relational Database Management System - RDBMS) là một hệ quản trị cơ sở dữ liệu theo mô hình quan hệ.

#### 1.1.2.2. Chức năng của Hệ quản trị cơ sở dữ liệu

- Quản lý lượng dữ liệu lớn hiệu quả

- Quản lý những cơ sở dữ liệu tồn tại lâu dài

- Hỗ trợ quan sát dữ liệu

- Hỗ trợ định nghĩa cấu trúc dữ liệu và thao tác trên dữ liệu thông qua các ngôn ngữ cao cấp nhất định

- Cung cấp các truy cập chính xác cùng lúc với nhiều truy cập dữ liệu khác nhau.

- Đảm bảo tính độc lập và bất biến của dữ liệu dù cấu trúc mô hình dữ liệu có thay đổi.

- Bảo vệ dữ liệu khỏi các truy cập không được cấp phép

- Phục hồi dữ liệu, tránh làm mất dữ liệu trên hệ thống

#### 1.1.2.3. Vai trò của Hệ quản trị cơ sở dữ liệu

Cung cấp môi trường tạo lập cơ sở dữ liệu: Vai trò của các hệ quản trị cơ sở dữ liệu đó chính là cung cấp cho người dùng một ngôn ngữ định nghĩa dữ liệu để cho họ có thể tự mô tả, khai báo kiểu và cấu trúc của dữ liệu

Cung cấp cách cập nhật và khai thác dữ liệu: Các hệ quản trị cơ sở dữ liệu sql có nhiệm vụ cung cấp cho người dùng ngôn ngữ thao tác dữ liệu để họ có thể diễn tả các yêu cầu cũng như thực hiện các thao tác cập nhật, khai thác CSDL. Những thao tác dữ liệu mà người dùng có thể thực hiện là nhập, sửa và xóa dữ liệu. Còn các khai thác dữ liệu bao gồm tìm kiếm và kết xuất dữ liệu

Cung cấp các công cụ kiểm soát và điều khiển các truy cập vào cơ sở dữ liệu: Đây là một trong các vai trò rất quan trọng của hệ quản trị cơ sở dữ liệu. Mục đích là để đảm bảo thực hiện một số yêu cầu cơ bản của hệ quản trị cơ sở dữ liệu. Các yêu cầu đó gồm:

* Đảm bảo an ninh và kịp thời phát hiện, thực hiện các hành động để ngăn chặn sự truy cập bất hợp phát;
* Duy trì được tính nhất quán của dữ liệu;
* Tổ chức, điều khiển các truy cập dữ liệu;
* Có thể khôi phục lại được cơ sở dữ liệu khi xảy ra các sự cố về phần cứng hoặc phần mềm;
* Quản lý các mô tả dữ liệu.

### 1.1.3. Sơ lược về SQL Server

SQL Server chính là một hệ quản trị dữ liệu quan hệ sử dụng câu lệnh SQL để trao đổi dữ liệu giữa máy cài SQL Server và máy Client. Một Relational Database Management System – RDBMS gồm có: databases, database engine và các chương trình ứng dụng dùng để quản lý các bộ phận trong RDBMS và những dữ liệu khác.

#### 1.1.3.1. Lịch sử ra đời và các ấn bản của SQL server

Năm 1989, phiên bản đầu tiên của SQL Server 1.0 ra đời được dùng cho các hệ điều hành 16 bit và được phát triển cho tới ngày nay.

Cho tới khi SQL Server ra phiên bản 6.5 thì được thị trường chấp nhận rộng rãi. Một đột phá cải tiến cho SQL Server 7.0 khi được Microsoft viết lại một engine hoàn toàn mới. Đến khi SQL Server từ phiên bản 7.0 cải tiến lên 8.0 chủ yếu phát triển về tính năng thiết kế website.

Một vài ấn bản SQL Server:

- Enterprise: là một ấn bản chứa tất cả các đặc điểm nổi bật của SQL Server như: các công cụ cho phép tạo và quản lý phân cụm SQL Server, nhân bộ máy cơ sở dữ liệu và một số dịch vụ đi kèm. Nó có thể đánh địa chỉ 12 terabytes và quản lý cơ sở dữ liệu lên tới 524 petabytes.

- Standard: Ấn bản này có thể chạy tốt trên hệ thống lên tới 4 CPU và 2 GB RAM rất thích hợp cho các dịch vụ thiết kế web vừa và nhỏ.

- Developer: Ấn bản này giới hạn số lượng người kết nối với server nhưng có đầy đủ các tính năng của Enterprise Edition. Đây là phiên bản được sử dụng cho kiểm tra và phát triển ứng dụng phù hợp cho các cá nhân trong lĩnh vực web như: freelancer Việt Nam, website spa,…

- Workgroup: ấn bản SQL Server này có các chức năng lõi cơ sở dữ liệu nhưng không đi kèm các dịch vụ. Ở phiên bản 2012 không có ấn bản này.

- Express: Ấn bản này dễ dàng sử dụng và quản trị cơ sở dữ liệu đơn giản.

#### 1.1.3.2. Các thành phần cơ bản trong SQL Server

Các thành cơ bản trong SQL Server gồm có: Reporting Services, Database Engine, Integration Services, Notification Services, Full Text Search Service,… Tất cả kết hợp với nhau tạo thành một giải pháp hoàn chỉnh giúp cho việc phân tích và lưu trữ dữ liệu trở nên dễ dàng hơn.

*- Database Engine:* Đây là một engine có khả năng chứa dữ liệu ở các quy mô dưới dạng support và table. Ngoài ra, nó còn có khả năng tự điều chỉnh ví dụ: trả lại tài nguyên cho hệ điều hành khi một user log off và sử dụng thêm các tài nguyên của máy khi cần.

*- Integration Services:* Đây là tập hợp các đối tượng lập trình và các công cụ đồ họa cho việc sao chép, di chuyển và chuyển đổi dữ liệu. Khi bạn làm việc trong một công ty lớn thì dữ liệu được lưu trữ ở nhiều nơi khác nhau như được chứa trong: Oracle, SQL Server, DB2, Microsoft Access,… và bạn chắc chắn sẽ có nhu cầu di chuyển dữ liệu giữa các server này. Ngoài ra, bạn còn muốn định dạng dữ liệu trước khi lưu vào database. Chắc chắn Integration Services sẽ giúp bạn giải quyết được công việc này dễ dàng.

*- Analysis Services:* Đây là một dịch vụ phân tích dữ liệu rất hay của Microsoft. Công cụ này giúp việc phân tích dữ liệu hiệu quả và dễ dàng bằng cách dùng kỹ thuật khai thác dữ liệu – datamining và khái niệm hình khối nhiều chiều – multi dimendion cubes.

*- Notification Services:* Dịch vụ thông báo này là nền tảng cho sự phát triển và triển khai các ứng dụng soạn và gửi thông báo. Ngoài ra, dịch vụ này còn có chức năng gửi thông báo theo dịch thời đến hàng ngàn người đăng ký sử dụng trên nhiều loại thiết bị khác nhau.

*- Reporting Services:* Là một công cụ tạo, quản lý và triển khai báo cáo bao gồm: server và client. Ngoài ra, nó còn là nền tảng cho việc phát triển và xây dựng các ứng dụng báo cáo.

*- Full Text Search Service:* Là một thành phần đặc biệt trong việc truy vấn và đánh chỉ mục dữ liệu văn bản không cấu trúc được lưu trữ trong các cơ sở dữ liệu SQL Server.

*- Service Broker:* Là một môi trường lập trình cho việc tạo ra các ứng dụng trong việc nhảy qua các Instance.

## 1.2. Sơ lược về Winforms, ADO.NET trong C#

### 1.2.1. Lập trình Winforms trong C#

Winform là một ứng dụng có giao diện người dùng đồ họa (GUI) nằm trong .Net framework. Mục đích của nó là cung cấp một giao diện đồ họa để phát triển các ứng dụng cho máy tính dễ dàng hơn. Winforms (Windows Form) chỉ có thể sử dụng để phát triển các ứng dụng chạy trên Desktop chứ không thể chạy trên web. Các ứng dụng WinForms có thể chứa các loại điều khiển khác nhau như label, textbox, listbox,…

Winform cho phép lập trình các ứng dụng Windows nhờ vào tính tiện ích, dễ viết, có giao diện kéo thả đơn giản,… Các GUI được thiết kế bằng việc thêm vào Form các điều khiển phù hợp.

Vai trò của WinForm:

* Chứa các điều khiển
* Xử lý dữ liệu người dùng nhập
* Kết nối với cơ sở dữ liệu
* Hiển thị thông tin

Ưu điểm:

* Tốc độ xử lý nhanh
* Đảm bảo bảo mật, an toàn thông tin
* Có thể tương thích với nhiều phiên bản Windows khác nhau
* Thao tác được trên nhiều giao diện

Nhược điểm:

* Phần mềm chạy trên Windows nên muốn sử dụng phải cài đặt phần mềm trên máy tính
* Chỉ sử dụng được trên Desktop
* Đồ họa không cao nên giao diện phần mềm thiếu tính trực quan

### 1.2.2. ADO.NET trong C#

#### 1.2.2.1. Giới thiệu về công nghệ ADO.NET

ADO.NET là một phần của .NET Framework, nó được xem là “bộ thư viện lớp” chịu trách nhiệm xử lý dữ liệu trong ngôn ngữ MS.NET. ADO.NET được thiết kế với dạng dữ liệu “ngắt kết nối”, nghĩa là chúng ta có thể lấy cả một cấu trúc phức tạp của dữ liệu từ database, sau đó ngắt kết nối với database rồi mới thực hiện các thao tác cần thiết. Đây là một sự tiến bộ về mặt thiết kế bởi vì thiết kế ADO trước đây luôn cần duy trì một kết nối trong quá trình thao tác dữ liệu.

#### 1.2.2.2. Kiến trúc ADO.NET

Kiến trúc ADO.NET có hai phần

Phần kết nối: được sử dụng khi ta kết nối với Database và thao tác dữ liệu, yêu cầu phải thực hiện kết nối với Database khi đang thao tác. Các đối tượng của phần này là:

*- Connection:* Đối tượng quản lý đóng /mở kết nối tới Database. Có hai dạng Connection tuỳ theo nguồn dữ liệu là gì (OleDb hay SQL Server) đó là OleDbConnection và SqlConnection.

*- Command:* Đối tượng thực hiện các câu lệnh tương tác truy vấn, rút trích dữ liệu từ database khi đã thiết lập kết nối tới dữ liệu và trả về kết quả. Tương tự như Connection, Command cũng có hai dạng tùy theo nguồn dữ liệu là gì (OleDb hay SQL Server) đó là OleDbCommand và SqlCommand.

*- DataReader:* Đối tượng xử lý đọc dữ liệu, được thiết kế phù hợp cho các ứng dụng web. Chỉ xử lý một dòng dữ liệu tại một thời điểm. Phù hợp với ứng dụng web vì xử lý nhanh, nhẹ không chiếm bộ nhớ. Cũng có 2 dạng tùy theo dữ liệu nguồn: OleDbDataReader và SqlDataReader. Dữ liệu của đối tượng được tạo ra khi đối tượng Command thực hiện câu lệnh ExecuteReader().

*- DataAdapter:* Đây là đối tượng rất quan trọng của ADO.NET, nó là cầu nối của database và dataset (dataset là đối tượng ngắt kết nối), bởi vì đối tượng “ngắt kết nối” dataset không thể liên lạc trực tiếp với database nên nó cần một đối tượng trung gian lấy dữ liệu từ database cho nó. Và đó chính là DataAdapter. Vì DataAdapter khi thao tác với Database vẫn phải duy trì kết nối nên nó được liệt kê vào dạng “kết nối”, nhưng bản chất là phục vụ cho việc “ngắt kết nối”.

Phần “ngắt kết nối”: chỉ có một đối tượng chịu trách nhiệm ngắt kết nối đó chính là DataSet. DataSet không cần biết gì về Database thuộc kiểu gì, kết nối ra sao. Nhiệm vụ của DataSet là nhận dữ liệu về từ DataAdapter và xử lý nó. DataSet có thể được xem như 1 Database trong bộ nhớ gồm tất cả các bảng, quan hệ

... DataSet có nhiều đối tượng được xem là “con” tức là cấp thấp hơn đi kèm với nó như: DataTable (tương đương với 1 bảng trong database), cấp thấp hơn của DataTable có các đối tượng DataRow (tương đương với 1 dòng), DataColumn( tương đương với 1 cột), DataRelation (tương đương với các quan hệ). Ngoài ra còn có các đối tượng nhóm: vd DataTableCollection, DataRowCollection, DataColumnCollection. Việc sử dụng DataSet là một tiến bộ lớn của kiến trúc ADO.NET tuy nhiên với các ứng dụng Web, việc sử dụng DataSet không được khuyến khích vì đối tượng DataSet được xem là quá lớn, nặng nề khó thích hợp cho đường truyền trên web vốn rất hạn chế.

## 1.3. Sơ lược về mô hình ba lớp

### 1.3.1. 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.

### 1.3.2. Các thành phần chính

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).

*- Business Logic Layer (BLL)*: Layer này phân ra 2 thành nhiệm vụ:

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.

*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,…).

**1.3.3. Ưu điểm và nhược điểm**

**Ưu điểm:**

- Phân loại rõ ràng các lớp có các nhiệm vụ khác nhau. Từ đó ta có thể quản lý và duy trì project tốt hơn

- Dễ dàng phân loại các hành động tại Business

- Dễ dàng phân loại các hàm truy xuất tại Database, phân loại hàm theo table,…

- Ứng dụng được cho các project lớn ở bên ngoài

- Thao tác thay đổi dữ liệu có thể được cô lập trong 1 lớp hoặc chỉ ảnh hưởng đến lớp gần nhất. Điều đó đảm bảo không ảnh hưởng đến cả chương trình, tiết kiệm thời gian công sức.

**Nhược điểm:**

- Việc truyền dữ liệu giữa các tầng sẽ chậm hơn vì phải truyền giữa các tiến trình khác nhau (IPC), dữ liệu cần phải được đóng gói 🡪 truyền đi 🡪 mở gói trước khi có thể dùng được

- Việc phát triển ứng dụng phức tạp hơn.

# Chương 2: PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG

## 2.1. Các yêu cầu cụ thể

### 2.1.1. Đối với người quản lý

- Hỗ trợ người quản lý tìm kiếm các thông tin về các đầu sách, thông tin về hóa đơn, khách hàng, tình hình kho hàng sách…

- Hỗ trợ đắc lực cho người quản lý giúp người quản lý nhận định một cách tống thể nhất về tình hình hoạt động, kinh doanh của cửa hàng mình

- Hỗ trợ công việc tìm kiếm của người quản lý một cách tốt nhất và nhanh nhất có thể.

### 2.1.2. Đối với nhân viên

- Hỗ trợ nhân viên tìm kiếm các thông tin liên quan đến sách như số lượng, đơn giá, khuyến mại giảm giá để nhân viên kịp thời có thông tin tiếp thị cho khách hàng

- Hỗ trợ nhân viên tìm kiếm các loại sách, tên sách một cách nhanh chóng, thống kê số lượng sách còn tồn trong kho

- Giúp nhân viên tìm kiếm thông tin khách hàng, nhà cung cấp một cách nhanh chóng, chính xác

- Nhập số liệu một cách nhanh chóng, chính xác

- Đảm bảo công việc lưu trữ chuẩn xác, tìm kiếm các thông tin về khách hàng, nhà cung cấp nhanh chóng, chính xác

- Trợ giúp nhân viên trong việc thống kê doanh thu của cửa hàng.

### 2.1.3. Đối với khách hàng

- Xuất hóa đơn cho khách hàng mua hàng

- Tìm kiếm thông tin sách cho khách hàng nếu khách hàng có yêu cầu tìm kiếm và kiểm tra xem sách đó còn hàng hay không?

- Thay đổi thông tin khách hàng khi cần.

## 2.2. Mô tả các chức năng của hệ thống

### 2.2.1. Chức năng Quản lý nhân viên

* Cập nhật: Thêm, xóa, sửa thông tin nhân viên
* Tìm kiếm nhân viên theo mã nhân viên, tên
* Hiển thị danh sách các nhân viên

### 2.2.2. Chức năng Quản lý khách hàng thành viên

* Cập nhật: Thêm, xóa, sửa thông tin khách hàng thành viên
* Tìm kiếm khách hàng thành viên theo số điện thoại, tên
* Hiển thị danh sách các khách hàng thành viên

### 2.2.3. Chức năng Quản lý kho sách

- Cập nhật: Thêm, xóa, sửa thông tin sách, thông tin tác giả, thông tin nhà xuất bản

- Tìm kiếm sách theo tên, thể loại, tác giả

- Hiển thị danh sách những quyển sách trong kho sách

### 2.2.4. Chức năng Quản lý mã giảm giá

- Cập nhật: Thêm, xóa, sửa thông tin các mã giảm giá

- Tìm kiếm mã giảm giá theo ngày

- Hiển thị danh sách các mã giảm giá

### 2.2.5. Chức năng Quản lý hóa đơn

- Tìm kiếm hóa đơn theo số điện thoại thành viên, ngày mua

- Hiển thị và xem chi tiết hóa đơn

### 2.2.6. Chức năng Quản lý phiếu nhập

- Cập nhật: Thêm phiếu nhập

- Tìm kiếm phiếu nhập theo số điện thoại nhà xuất bản, ngày nhập

- Hiển thị và xem chi tiết phiếu nhập

### 2.2.7. Chức năng Quản lý thu nhập

- Hiển thị thu nhập theo tháng

### 2.2.8. Chức năng lập hóa đơn bán hàng

Khi có khách hàng đến mua sách và thanh toán thì nhân viên bán hàng sẽ sử  
dụng chức năng này để lập hóa đơn bán hàng

## 2.3. Thiết kế cơ sở dữ liệu

### 2.3.1. Xây dựng các thực thể

- Nhân viên: Mỗi nhân viên phải có một id, quyền đăng nhập vào hệ thống, username, password và các thông tin như họ tên, địa chỉ, số điện thoại, lương, hình ảnh, ngày tạo, ngày update

- Loại thành viên: Mỗi loại thành viên sẽ có một id, tên, mức ưu đãi, ngày tạo, ngày update

- Khách hàng thành viên: Khi mua hàng lần đầu tiên phải đăng kí là thành viên của cửa hàng và mỗi thành viên sẽ có một id và các thông tin như học tên, địa chỉ, số điện thoại, loại thành viên, ngày tạo, ngày update

- Sách: Mỗi quyển sách sẽ có một id và các thông tin như tên sách, thể loại, tên tác giả, nhà xuất bản, giá bán, số lượng sách trong kho, ngày tạo, ngày update

- Giảm giá: Mỗi mã giảm giá có một id, ngày bắt đầu, ngày kết thúc, phần trăm giảm giá, tiêu đề giảm giá, ngày tạo, ngày update

- Hóa đơn: Mỗi hóa đơn sẽ có một id và các thông tin như tên nhân viên, tên thành viên, ngày mua, mã giảm giá, giá trị hóa đơn, ngày tạo, ngày update

- Chi tiết hóa đơn: Chi tiết hóa đơn sẽ có id hóa đơn, tên sách, số lượng, giá bán, ngày tạo, ngày update

- Phiếu nhập: Mỗi phiếu nhập sẽ có một id và các thông tin như tên nhân viên, tên nhà xuất bản, ngày nhập, tổng tiền nhập, ngày tạo, ngày update

- Chi tiết phiếu nhập: Chi tiết phiếu nhập sẽ có id phiếu nhập, tên sách, số lượng, giá nhập, ngày tạo, ngày update

### 2.3.2. Mối quan hệ giữa các thực thể

Sau khi tạo xong các bảng dữ liệu ta tiến hành liên kết các bảng lại với nhau để dữ liệu từng bảng có thể kết nối và trở thành một hệ thống lưu trữ dữ liệu hoàn chỉnh:

TacGia liên kết 1-n với Sach qua ID\_TG

LoaiSach liên kết 1-n với Sach qua ID\_LoaiSach

GiamGia liên kết 1-n với HoaDon qua ID\_GG

Sach liên kết 1-n với ChiTietHoaDon qua ID\_Sach

Sach liên kết 1-n với ChiTietPhieuNhap qua ID\_Sach

NXB liên kết 1-n với Sach qua ID\_NXB

NXB liên kết 1-n với PhieuNhap qua ID\_NXB

PhieuNhap liên kết 1-n với ChiTietPhieuNhap qua ID\_PN

NhanVien liên kết 1-n với PhieuNhap qua ID\_NV

HoaDon liên kết 1-n với ChiTietHoaDon qua ID\_HD

NhanVien liên kết 1-n với HoaDon qua ID\_NV

ThanhVien liên kết 1-n với HoaDon qua ID\_TV

LoaiThanhVien liên kết 1-n với ThanhVien qua ID\_LoaiTV

### 2.3.3. Xây dựng mô hình ERD

Diagram, schematic

Description automatically generated

### 2.3.4. Chuyển từ mô hình ERD sang mô hình quan hệ

**NhanVien** (ID, Quyen, Username, Password, HoTen, DiaChi, SĐT, Luong, HinhAnh NgayTao, NgayUpdate)

**ThanhVien** (ID, HoTen, DiaChi, SĐT, NgayTao, NgayUpdate, ID\_LoaiTV)

**LoaiThanhVien** (ID, TenLoaiTV, MucUuDai, NgayTao, NgayUpdate)

**Sach** (ID, TenSach, ID\_LoaiSach, ID\_NXB, ID\_TacGia, GiaBan, SoLuongTonKho, NgayTao, NgayUpdate)

**LoaiSach** (ID, TenTheLoai, NgayTao, NgayUpdate)

**NXB** (ID, TenNXB, DiaChi, SĐT, NgayTao, NgayUpdate)

**TacGia** (ID, TenTacGia, NgayTao, NgayUpdate)

**GiamGia** (ID, TieuDe, PhanTramGG, NgayBatDau, NgayKetThuc, GhiChu, NgayTao, NgayUpdate)

**HoaDon** (ID, ID\_NV, ID\_TV, ID\_GG, NgayMua, ThanhTien, NgayTao, NgayUpdate)

**ChiTietHoaDon** (ID\_HoaDon, ID\_Sach, SoLuong, NgayTao, NgayUpdate)

**PhieuNhap** (ID, ID\_NV, ID\_NXB, NgayNhap, ThanhTien, NgayTao, NgayUpdate)

**ChiTietPhieuNhap** (ID\_PN, ID\_Sach, SoLuong, GiaNhap, NgayTao, NgayUpdate)

### 2.3.5. Mô hình Diagrams

Diagram, schematic

Description automatically generated

### 2.3.6. Chi tiết từng thực thể

**Bảng 2.3.6.1. NhanVien**: Nhân viên

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Tên thuộc tính** | **Kiểu dữ liệu** | **Unique** | **Not null** | **Khóa chính** | **Khóa ngoại** | **Quan hệ được tham chiếu** | **Mô tả** |
| ID | Bigint | x | x | x |  |  | ID nhân viên |
| Quyen | Char(10) |  | x |  |  |  | Quyền đăng nhập |
| Username | Varchar(50) | x | x |  |  |  | Tên đăng nhập |
| Password | Varchar(50) |  | x |  |  |  | Mật khẩu đăng nhập |
| HoTen | Nvarchar(50) |  | x |  |  |  | Họ tên nhân viên |
| DiaChi | Nvarchar(200) |  | x |  |  |  | Địa chỉ |
| SĐT | Varchar(11) |  | x |  |  |  | Số điện thoại |
| Luong | Bigint |  | x |  |  |  | Lương/tháng |
| Hình ảnh | Image |  |  |  |  |  | Hình ảnh |
| NgayTao | Datetime |  |  |  |  |  | Ngày tạo dữ liệu |
| NgayUpdate | Datetime |  |  |  |  |  | Ngày chỉnh sửa |

**Bảng 2.3.6.2. LoaiThanhVien:** Loại thành viên

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Tên thuộc tính** | **Kiểu dữ liệu** | **Unique** | **Not null** | **Khóa chính** | **Khóa ngoại** | **Quan hệ được tham chiếu** | **Mô tả** |
| ID | Bigint | x | x | x |  |  | ID loại thành viên |
| TenLoaiTV | Nvarchar(50) |  | x |  |  |  | Tên loại thành viên |
| MucUuDai | Float |  | x |  |  |  | Mức ưu đãi |
| NgayTao | Datetime |  | x |  |  |  | Ngày tạo dữ liệu |
| NgayUpdate | Datetime |  |  |  |  |  | Ngày chỉnh sửa |

**Bảng 2.3.6.3. ThanhVien**: Thành viên

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Tên thuộc tính** | **Kiểu dữ liệu** | **Unique** | **Not null** | **Khóa chính** | **Khóa ngoại** | **Quan hệ được tham chiếu** | **Mô tả** |
| ID | Bigint | x | x | x |  |  | ID thành viên |
| HoTen | Nvarchar(50) |  | x |  |  |  | Họ tên thành viên |
| DiaChi | Nvarchar(200) |  | x |  |  |  | Địa chỉ |
| SĐT | Varchar(11) |  | x |  |  |  | Số điện thoại |
| ID\_LoaiTV | Bigint |  | x |  | x | LoaiThanhVien (ID) | ID loại thành viên |
| NgayTao | Datetime |  | x |  |  |  | Ngày tạo dữ liệu |
| NgayUpdate | Datetime |  |  |  |  |  | Ngày chỉnh sửa |

**Bảng 2.3.6.4. Sach**: Sách

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Tên thuộc tính** | **Kiểu dữ liệu** | **Unique** | **Not null** | **Khóa chính** | **Khóa ngoại** | **Quan hệ được tham chiếu** | **Mô tả** |
| ID | Bigint | x | x | x |  |  | ID sách |
| TenSach | Nvarchar(200) |  | x |  |  |  | Tên sách |
| ID\_LoaiSach | Bigint |  | x |  | x | LoaiSach(ID) | ID loại sách |
| ID\_NXB | Bigint |  | x |  | x | NXB(ID) | ID nhà xuất bản |
| ID\_TacGia | Bigint |  | x |  | x | TacGia(ID) | ID tác giả |
| GiaBan | Bigint |  | x |  |  |  | Giá bán |
| SoLuongTonKho | Int |  | x |  |  |  | Số lượng tồn kho |
| NgayTao | Datetime |  | x |  |  |  | Ngày tạo dữ liệu |
| NgayUpdate | Datetime |  |  |  |  |  | Ngày chỉnh sửa |

**Bảng 2.3.6.5. LoaiSach**: Loại sách

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Tên thuộc tính** | **Kiểu dữ liệu** | **Unique** | **Not null** | **Khóa chính** | **Khóa ngoại** | **Quan hệ được tham chiếu** | **Mô tả** |
| ID | Bigint | x | x | x |  |  | ID loại sách |
| TenLoaiSach | Nvarchar(200) |  | x |  |  |  | Tên loại sách |
| NgayTao | Datetime |  | x |  |  |  | Ngày tạo dữ liệu |
| NgayUpdate | Datetime |  |  |  |  |  | Ngày chỉnh sửa |

**Bảng 2.3.6.6. TacGia**: Tác giả

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Tên thuộc tính** | **Kiểu dữ liệu** | **Unique** | **Not null** | **Khóa chính** | **Khóa ngoại** | **Quan hệ được tham chiếu** | **Mô tả** |
| ID | Bigint | x | x | x |  |  | ID tác giả |
| TenTacGia | Nvarchar(50) |  | x |  |  |  | Tên tác giả |
| NgayTao | Datetime |  | x |  |  |  | Ngày tạo dữ liệu |
| NgayUpdate | Datetime |  |  |  |  |  | Ngày chỉnh sửa |

**Bảng 2.3.6.7. NXB**: Nhà xuất bản

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Tên thuộc tính** | **Kiểu dữ liệu** | **Unique** | **Not null** | **Khóa chính** | **Khóa ngoại** | **Quan hệ được tham chiếu** | **Mô tả** |
| ID | Bigint | x | x | x |  |  | ID nhà xuất bản |
| TenNXB | Nvarchar(200) |  | x |  |  |  | Tên nhà xuất bản |
| DiaChi | Nvarchar(200) |  | x |  |  |  | Địa chỉ |
| SĐT | Nvarchar(11) |  | x |  |  |  | Số điện thoại |
| NgayTao | Datetime |  | x |  |  |  | Ngày tạo dữ liệu |
| NgayUpdate | Datetime |  |  |  |  |  | Ngày chỉnh sửa |

**Bảng 2.3.6.8. GiamGia**: Giảm giá

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Tên thuộc tính** | **Kiểu dữ liệu** | **Unique** | **Not null** | **Khóa chính** | **Khóa ngoại** | **Quan hệ được tham chiếu** | **Mô tả** |
| ID | Bigint | x | x | x |  |  | ID loại sách |
| TieuDe | Nvarchar(100) |  | x |  |  |  | Tiêu đề giảm giá |
| PhanTramGG | Float |  | x |  |  |  | Phần trăm giảm giá |
| NgayBatDau | Datetime |  | x |  |  |  | Ngày bắt đầu |
| NgayKetThuc | Datetime |  | x |  |  |  | Ngà̀y kết thúc |
| GhiChu | Nvarchar(100) |  | x |  |  |  | Ghi chú |
| NgayTao | Datetime |  | x |  |  |  | Ngày tạo dữ liệu |
| NgayUpdate | Datetime |  |  |  |  |  | Ngày chỉnh sửa |

**Bảng 2.3.6.9. HoaDon**: Hóa đơn

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Tên thuộc tính** | **Kiểu dữ liệu** | **Unique** | **Not null** | **Khóa chính** | **Khóa ngoại** | **Quan hệ được tham chiếu** | **Mô tả** |
| ID | Bigint | x | x | x |  |  | ID hóa đơn |
| ID\_NV | Bigint |  | x |  | x | NhanVien(ID) | ID nhân viên |
| ID\_TV | Bigint |  | x |  | x | ThanhVien(ID) | ID thành viên |
| ID\_GG | Bigint |  |  |  | x | GiamGia(ID) | ID mã giảm giá |
| NgayMua | Datetime |  | x |  |  |  | Ngày mua hàng |
| ThanhTien | Bigint |  | x |  |  |  | Thành tiền |
| NgayTao | Datetime |  |  |  |  |  | Ngày tạo dữ liệu |
| NgayUpdate | Datetime |  |  |  |  |  | Ngày chỉnh sửa |

**Bảng 2.3.6.10. ChiTietHoaDon**: Chi tiết hóa đơn

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Tên thuộc tính** | **Kiểu dữ liệu** | **Unique** | **Not null** | **Khóa chính** | **Khóa ngoại** | **Quan hệ được tham chiếu** | **Mô tả** |
| ID\_HD | Bigint | x | x | x | x | HoaDon(ID) | ID hóa đơn |
| ID\_Sach | Bigint |  | x | x | x | Sach(ID) | ID của sách bán |
| SoLuong | Int |  | x |  |  |  | Số lượng sách bán |
| NgayTao | Datetime |  |  |  |  |  | Ngày tạo dữ liệu |
| NgayUpdate | Datetime |  |  |  |  |  | Ngày chỉnh sửa |

**Bảng 2.3.6.11. PhieuNhap**: Phiếu nhập

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Tên thuộc tính** | **Kiểu dữ liệu** | **Unique** | **Not null** | **Khóa chính** | **Khóa ngoại** | **Quan hệ được tham chiếu** | **Mô tả** |
| ID | Bigint | x | x | x |  |  | ID phiếu nhập |
| ID\_NV | Bigint |  | x |  | x | NhanVien(ID) | ID nhân viên |
| ID\_NXB | Bigint |  | x |  | x | NXB(ID) | ID nhà xuất bản |
| NgayNhap | Datetime |  | x |  |  |  | Ngày nhập |
| ThanhTien | Bigint |  | x |  |  |  | Thành tiền |
| NgayTao | Datetime |  |  |  |  |  | Ngày tạo dữ liệu |
| NgayUpdate | Datetime |  |  |  |  |  | Ngày chỉnh sửa |

**Bảng 2.3.6.12.ChiTietPhieuNhap**: Chi tiết phiếu nhập

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Tên thuộc tính** | **Kiểu dữ liệu** | **Unique** | **Not null** | **Khóa chính** | **Khóa ngoại** | **Quan hệ được tham chiếu** | **Mô tả** |
| ID\_PN | Bigint | x | x | x | x | PhieuNhap(ID) | ID phiếu nhập |
| ID\_Sach | Bigint |  | x | x | x | Sach(ID) | ID của sách nhập |
| SoLuong | Int |  | x |  |  |  | Số lượng sách nhập |
| GiaNhap | Bigint |  | x |  |  |  | Giá nhập |
| NgayTao | Datetime |  |  |  |  |  | Ngày tạo dữ liệu |
| NgayUpdate | Datetime |  |  |  |  |  | Ngày chỉnh sửa |

# Chương 3: CÀI ĐẶT HỆ THỐNG

## 3.1. Cài đặt cơ sở dữ liệu và tạo các ràng buộc khóa chính, khóa ngoại và not null

CREATE DATABASE QuanLyCuaHangSach

GO

USE QuanLyCuaHangSach

GO

CREATE TABLE LoaiSach

(

ID BIGINT IDENTITY(1,1) NOT NULL,

TenLoaiSach NVARCHAR(200) NOT NULL,

NgayTao DATETIME NOT NULL,

NgayUpdate DATETIME NULL,

CONSTRAINT PK\_LoaiSach PRIMARY KEY (ID)

)

GO

CREATE TABLE TacGia

(

ID BIGINT IDENTITY(1,1) NOT NULL,

TenTacGia NVARCHAR(50) NOT NULL,

NgayTao DATETIME NOT NULL,

NgayUpdate DATETIME NULL,

CONSTRAINT PK\_TacGia PRIMARY KEY (ID)

)

GO

CREATE TABLE NXB

(

ID BIGINT IDENTITY(1,1) NOT NULL,

TenNXB NVARCHAR(200) NOT NULL,

DiaChi NVARCHAR(200) NOT NULL,

SDT VARCHAR(11) NOT NULL,

NgayTao DATETIME NOT NULL,

NgayUpdate DATETIME NULL,

CONSTRAINT PK\_NXB PRIMARY KEY (ID)

)

GO

CREATE TABLE Sach

(

ID BIGINT IDENTITY(1,1) NOT NULL,

TenSach NVARCHAR(200) NOT NULL,

ID\_LoaiSach BIGINT NOT NULL DEFAULT 1,

ID\_NXB BIGINT NOT NULL DEFAULT 1,

ID\_TacGia BIGINT NOT NULL DEFAULT 1,

GiaBan BIGINT NOT NULL,

SoLuongTonKho INT NOT NULL,

NgayTao DATETIME NOT NULL,

NgayUpdate DATETIME NULL,

CONSTRAINT PK\_Sach PRIMARY KEY (ID),

CONSTRAINT FK\_Sach\_LoaiSach FOREIGN KEY (ID\_LoaiSach) REFERENCES LoaiSach (ID)

ON DELETE SET DEFAULT

ON UPDATE CASCADE,

CONSTRAINT FK\_Sach\_NXB FOREIGN KEY (ID\_NXB) REFERENCES NXB (ID)

ON DELETE SET DEFAULT

ON UPDATE CASCADE,

CONSTRAINT FK\_Sach\_TacGia FOREIGN KEY (ID\_TacGia) REFERENCES TacGia (ID)

ON DELETE SET DEFAULT

ON UPDATE CASCADE

)

GO

CREATE TABLE LoaiThanhVien (

ID BIGINT IDENTITY(1,1) NOT NULL,

TenLoaiTV NVARCHAR(50) NOT NULL,

MucUuDai FLOAT NOT NULL DEFAULT 0,

NgayTao DATETIME NOT NULL,

NgayUpdate DATETIME NULL,

CONSTRAINT PK\_LoaiNhanVien PRIMARY KEY (ID)

)

GO

CREATE TABLE ThanhVien

(

ID BIGINT IDENTITY(1,1) NOT NULL,

HoTen NVARCHAR(50) NOT NULL,

DiaChi NVARCHAR(200) NOT NULL,

SDT VARCHAR(11) NOT NULL,

ID\_LoaiTV BIGINT DEFAULT 1,

NgayTao DATETIME NOT NULL,

NgayUpdate DATETIME NULL,

CONSTRAINT PK\_ThanhVien PRIMARY KEY(ID),

CONSTRAINT FK\_ThanhVien\_LoaiThanhVien FOREIGN KEY (ID\_LoaiTV) REFERENCES LoaiThanhVien(ID)

ON DELETE SET DEFAULT

ON UPDATE CASCADE

)

GO

CREATE TABLE NhanVien (

ID BIGINT IDENTITY(1,1) NOT NULL,

Quyen CHAR(10) NOT NULL,

Username VARCHAR(50) NOT NULL,

Password VARCHAR(50) NOT NULL,

HoTen NVARCHAR(50) NOT NULL,

DiaChi NVARCHAR(200) NOT NULL,

SDT VARCHAR(11) NOT NULL,

Luong BIGINT NOT NULL,

HinhAnh IMAGE NULL,

NgayTao DATETIME NOT NULL,

NgayUpdate DATETIME NULL,

CONSTRAINT PK\_NhanVien PRIMARY KEY (ID)

)

GO

CREATE TABLE PhieuNhap

(

ID BIGINT IDENTITY(1,1) NOT NULL,

ID\_NV BIGINT NOT NULL DEFAULT 1,

ID\_NXB BIGINT NOT NULL DEFAULT 1,

NgayNhap DATETIME NOT NULL,

ThanhTien BIGINT NOT NULL,

NgayTao DATETIME NULL,

NgayUpdate DATETIME NULL,

CONSTRAINT PK\_PhieuNhap PRIMARY KEY (ID),

CONSTRAINT FK\_PhieuNhap\_NhanVien FOREIGN KEY (ID\_NV) REFERENCES NhanVien (ID)

ON DELETE SET DEFAULT

ON UPDATE CASCADE,

CONSTRAINT FK\_PhieuNhap\_NXB FOREIGN KEY (ID\_NXB) REFERENCES NXB (ID)

ON DELETE SET DEFAULT

ON UPDATE CASCADE

)

GO

CREATE TABLE ChiTietPhieuNhap (

ID\_PN BIGINT NOT NULL DEFAULT 1,

ID\_Sach BIGINT NOT NULL DEFAULT 1,

SoLuong INT NOT NULL,

GiaNhap BIGINT NOT NULL,

NgayTao DATETIME NULL,

NgayUpdate DATETIME NULL,

CONSTRAINT PK\_ChiTietPhieuNhap PRIMARY KEY (ID\_PN, ID\_Sach),

CONSTRAINT FK\_ChiTietPhieuNhap\_PhieuNhap FOREIGN KEY (ID\_PN) REFERENCES PhieuNhap (ID)

ON DELETE SET DEFAULT

ON UPDATE CASCADE,

CONSTRAINT FK\_ChiTietPhieuNhap\_Sach FOREIGN KEY (ID\_Sach) REFERENCES Sach (ID)

)

GO

CREATE TABLE GiamGia (

ID BIGINT IDENTITY(1,1) NOT NULL,

TieuDe NVARCHAR(100) NOT NULL,

PhanTramGG FLOAT NOT NULL,

NgayBatDau DATETIME NOT NULL,

NgayKetThuc DATETIME NOT NULL,

GhiChu NVARCHAR(100) NULL,

NgayTao DATETIME NOT NULL,

NgayUpdate DATETIME NULL,

CONSTRAINT PK\_GiamGia PRIMARY KEY (ID)

)

GO

CREATE TABLE HoaDon

(

ID BIGINT IDENTITY(1,1) NOT NULL,

ID\_NV BIGINT NOT NULL DEFAULT 1,

ID\_TV BIGINT NOT NULL DEFAULT 1,

ID\_GG BIGINT DEFAULT 0,

NgayMua DATETIME NOT NULL,

ThanhTien BIGINT NOT NULL,

NgayTao DATETIME NULL,

NgayUpdate DATETIME NULL,

CONSTRAINT PK\_HoaDon PRIMARY KEY (ID),

CONSTRAINT FK\_HoaDon\_NhanVien FOREIGN KEY (ID\_NV) REFERENCES NHANVIEN(ID)

ON DELETE SET DEFAULT

ON UPDATE CASCADE,

CONSTRAINT FK\_HoaDon\_ThanhVien FOREIGN KEY (ID\_TV) REFERENCES THANHVIEN(ID)

ON DELETE SET DEFAULT

ON UPDATE CASCADE,

CONSTRAINT FK\_HoaDon\_GiamGia FOREIGN KEY (ID\_GG) REFERENCES GIAMGIA(ID)

ON DELETE SET DEFAULT

ON UPDATE CASCADE

)

GO

CREATE TABLE ChiTietHoaDon (

ID\_HD BIGINT NOT NULL DEFAULT 1,

ID\_Sach BIGINT NOT NULL DEFAULT 1,

SoLuong INT NOT NULL,

NgayTao DATETIME NULL,

NgayUpdate DATETIME NULL,

CONSTRAINT PK\_ChiTietHoaDon PRIMARY KEY (ID\_HD, ID\_Sach),

CONSTRAINT FK\_ChiTietHoaDon\_HoaDon FOREIGN KEY (ID\_HD) REFERENCES HoaDon (ID)

ON DELETE SET DEFAULT

ON UPDATE CASCADE,

CONSTRAINT FK\_ChiTietHoaDon\_Sach FOREIGN KEY (ID\_Sach) REFERENCES Sach (ID)

ON DELETE SET DEFAULT

ON UPDATE CASCADE,

)

GO

## 3.2. Cài đặt Trigger

### 3.2.1. Trigger kiểm tra số lượng trên hóa đơn nhỏ hơn số lượng tồn kho

CREATE TRIGGER trgSoLuongChiTietHoaDon ON ChiTietHoaDon

AFTER INSERT AS

DECLARE @sl INT, @sltk INT

SELECT @sl = ChiTietHoaDon.SoLuong, @sltk = Sach.SoLuongTonKho

FROM Inserted ChiTietHoaDon JOIN Sach ON ChiTietHoaDon.ID\_Sach = Sach.ID

IF (@sl > @sltk)

BEGIN

RAISERROR(N'Số lượng tồn kho không đủ', 16, 1)

ROLLBACK

RETURN

END

GO

sp\_SetTriggerOrder @triggername= 'trgSoLuongChiTietHoaDon', @order='First', @stmttype = 'INSERT'

GO

### 3.2.2. Trigger cập nhật số lượng sách trong bảng Sách sau khi bán sách

CREATE TRIGGER trgBanSach ON ChitietHoaDon

AFTER INSERT AS

BEGIN

UPDATE Sach

SET SoLuongTonKho = SoLuongTonKho - (

SELECT SoLuong

FROM inserted

WHERE ID\_Sach = Sach.ID

)

FROM Sach

JOIN inserted ON Sach.ID = inserted.ID\_Sach

END

GO

### 3.2.3. Trigger cập nhật sách trong trong bảng Sách sau khi cập nhật bán sách

CREATE TRIGGER trgCapNhatBanSach ON ChitietHoaDon

AFTER UPDATE AS

BEGIN

UPDATE Sach SET SoLuongTonKho = SoLuongTonKho -

(SELECT SoLuong FROM inserted WHERE ID\_Sach = Sach.ID) +

(SELECT SoLuong FROM deleted WHERE ID\_Sach = Sach.ID)

FROM Sach

JOIN deleted ON Sach.ID = deleted.ID\_Sach

END

GO

### 3.2.4. Trigger kiểm tra số lượng sách nhập phải lớn hơn 0

CREATE TRIGGER trgSoLuongChiTietPhieuNhap ON ChiTietPhieuNhap

AFTER INSERT AS

DECLARE @sl INT

SELECT @sl= ChiTietPhieuNhap.SoLuong

FROM Inserted ChiTietPhieuNhap

IF @sl<1

BEGIN

RAISERROR(N'Số lượng không hợp lệ', 16, 1)

ROLLBACK

RETURN

END

GO

### 3.2.5. Trigger cập nhật số lượng sách trong bảng Sách sau khi nhập sách

CREATE TRIGGER trgNhapSach ON ChitietPhieuNhap

AFTER INSERT AS

BEGIN

UPDATE Sach

SET SoLuongTonKho = SoLuongTonKho + (

SELECT SoLuong

FROM inserted

WHERE ID\_Sach = Sach.ID

)

FROM Sach

JOIN inserted ON Sach.ID = inserted.ID\_Sach

END

GO

### 3.2.6. Trigger cập nhật sách trong trong bảng Sách sau khi cập nhật nhập sách

CREATE TRIGGER trgCapNhatNhapSach ON ChitietPhieuNhap

AFTER UPDATE AS

BEGIN

UPDATE Sach SET SoLuongTonKho = SoLuongTonKho +

(SELECT SoLuong FROM inserted WHERE ID\_Sach = Sach.ID) +

(SELECT SoLuong FROM deleted WHERE ID\_Sach = Sach.ID)

FROM Sach

JOIN deleted ON Sach.ID = deleted.ID\_Sach

END

GO

### 3.2.7. Kiểm tra ngày bắt đầu giảm giá phải trước ngày kết thúc giảm giá

CREATE TRIGGER trgNgayBatDau ON GiamGia

AFTER INSERT AS

DECLARE @NgayBatDau datetime, @NgayKetThuc datetime

SELECT @NgayBatDau = ne.NgayBatDau, @NgayKetThuc = ne.NgayKetThuc

FROM inserted ne

IF(@NgayBatDau >= @NgayKetThuc)

BEGIN

ROLLBACK

RAISERROR(N'Ngày bắt đầu của giảm giá phải trước ngày kết thúc', 16, 1)

RETURN

END

GO

### 3.2.8. Trigger kiểm tra phần trăm giảm nhỏ hơn 100

CREATE TRIGGER trgPTGG ON GiamGia

AFTER INSERT AS

DECLARE @PhanTramGG FLOAT

SELECT @PhanTramGG=PhanTramGG FROM inserted

IF(@PhanTramGG>100)

BEGIN

ROLLBACK

RAISERROR(N'Phần trăm giảm giá không hợp lệ', 16, 1)

RETURN

END

## 3.3. Cài đặt các Views

### 3.3.1. View xem Chi tiết hóa đơn

CREATE VIEW viewChiTietHoaDon AS

SELECT ChiTietHoaDon.ID\_HD, NhanVien.HoTen as TenNV, ThanhVien.HoTen as TenTV, ThanhVien.SDT as SDT\_TV, TenSach, SoLuong, GiaBan, PhanTramGG, MucUuDai, ThanhTien = (ChiTietHoaDon.SoLuong \* Sach.GiaBan\*((100-(PhanTramGG + MucUuDai))/100))

FROM HoaDon, ChiTietHoaDon, NhanVien, ThanhVien, Sach, GiamGia, LoaiThanhVien

WHERE ChiTietHoaDon.ID\_HD = HoaDon.ID

and HoaDon.ID\_NV = NhanVien.ID

and HoaDon.ID\_TV = ThanhVien.ID

and ChiTietHoaDon.ID\_Sach = Sach.ID

and HoaDon.ID\_GG = GiamGia.ID

and ThanhVien.ID\_TV = LoaiThanhVien.ID

GO

### 3.3.2. View xem Chi tiết phiếu nhập

CREATE VIEW viewChiTietPhieuNhap AS

SELECT ChiTietPhieuNhap.ID\_PN, NhanVien.HoTen as TenNV, TenNXB, NXB.SDT as SDT\_NXB, TenSach, SoLuong, GiaNhap, ThanhTien = (ChiTietPhieuNhap.SoLuong \* ChiTietPhieuNhap.GiaNhap)

FROM PhieuNhap, ChiTietPhieuNhap, NhanVien, NXB, Sach

WHERE ChiTietPhieuNhap.ID\_PN = PhieuNhap.ID

and PhieuNhap.ID\_NV = NhanVien.ID

and PhieuNhap.ID\_NXB = NXB.ID

and ChiTietPhieuNhap.ID\_Sach = Sach.ID

GO

### 3.3.3. View lấy ra top 20 cuốn sách bán chạy nhất

CREATE VIEW viewTopSell AS

SELECT top(20) \* from funSoLuongSachBanDuoc ()

ORDER BY slBanDuoc DESC

GO

### 3.3.4. View xem số lần mua hàng của mỗi thành viên

CREATE VIEW viewSoLanMuaHang AS

SELECT ThanhVien.ID, ThanhVien.HoTen, count(HoaDon.ID\_TV) as SoLanMua

FROM ThanhVien, HoaDon

WHERE ThanhVien.ID = HoaDon.ID\_TV

GROUP BY ThanhVien.ID, ThanhVien.HoTen

GO

### 3.3.5. View lấy ra top 20 thành viên có giá trị mua hàng lớn nhất

CREATE VIEW viewDoanhThuThanhVien AS

SELECT TOP(20) \* from funCountTotalbyThanhVien ()

ORDER BY DoanhThu desc

GO

### 3.3.6. View xem lương nhân viên

CREATE View [dbo].[viewNhanVien] AS

SELECT NhanVien.ID, HoTen, Luong, Count(HoaDon.ID) as SoHD

FROM NhanVien LEFT OUTER JOIN (select \* from HoaDon where MONTH(HoaDon.NgayMua)= (SELECT MONTH(GETDATE()))) as HoaDon on NhanVien.ID = HoaDon.ID\_NV

WHERE Luong IS NOT NULL

GROUP BY NhanVien.ID, HoTen, Luong

GO

## 3.4. Cài đặt các Stored Procedures

### 3.4.1. Stored Procedure xem, thêm, xóa, sửa dữ liêu các bảng

Bảng NhanVien

- Xem bảng nhân viên:

CREATE PROCEDURE spGetNhanVien

AS

BEGIN

SELECT \* FROM NhanVien

END

GO

- Thêm nhân viên:

CREATE PROCEDURE spInsertNhanVien

@Quyen char(10),

@Username varchar(50),

@Password varchar(50),

@HoTen nvarchar(50),

@DiaChi nvarchar(200),

@SDT varchar(11),

@Luong bigint,

@HinhAnh image,

@NgayTao datetime

AS

BEGIN

INSERT INTO NhanVien (Quyen, Username, Password, HoTen, DiaChi, SDT, Luong, HinhAnh, NgayTao)

VALUES

(@Quyen, @Username, @Password, @HoTen, @DiaChi, @SDT, @Luong, @HinhAnh, @NgayTao)

END

GO

- Xóa nhân viên:

CREATE PROCEDURE spDeleteNhanVien

@ID bigint

AS

BEGIN

DELETE FROM NhanVien

WHERE ID = @ID

END

GO

- Cập nhật thông tin nhân viên:

CREATE PROCEDURE spUpdateNhanVien

@Quyen char(10),

@ID bigint,

@Username varchar(50),

@Password varchar(50),

@HoTen nvarchar(50),

@DiaChi nvarchar(200),

@SDT varchar(11),

@Luong bigint,

@HinhAnh image,

@NgayUpdate datetime

AS

BEGIN TRANSACTION NhanVienUpdate

WITH MARK N'UPDATE NHAN VIEN';

IF (@HoTen is null or @UserName is null or @Password is null)

begin

raiserror ('Ho ten, UserName. Password cua nhan vien khong duoc bo trong',16,1)

rollback

return

end

UPDATE NhanVien

SET

Quyen=@Quyen,

Username = @Username,

Password = @Password,

HoTen = @HoTen,

DiaChi = @DiaChi,

SDT = @SDT,

Luong = @Luong,

HinhAnh = @HinhAnh,

NgayUpdate = @NgayUpdate

WHERE ID = @ID

IF (@@ERROR <>0)

BEGIN

raiserror('Error',16,1)

rollback

return

END

COMMIT TRAN

GO

Cài đặt tương tự cho các bảng: ThanhVien, LoaiThanhVien, GiamGia, Sach, LoaiSach, TacGia, NXB, HoaDon, ChiTietHoaDon, PhieuNhap, ChiTietPhieuNhap.

### 3.4.2. Stored procedure tìm kiếm nhân viên

- Tìm kiếm nhân viên bằng ID

CREATE PROCEDURE spFindNhanVienByID

@ID bigint

AS

BEGIN

select \* from NhanVien where ID = @ID

END

- Tìm kiếm nhân viên bằng Username và Password

CREATE PROCEDURE spFindNhanVien

@Username NVARCHAR(50),

@Password NVARCHAR(50)

AS

BEGIN

select \* from nhanvien where Username = @Username and Password = @Password

END

### 3.4.3. Stored procedure tìm kiếm loại sách

- Tìm kiếm loại sách theo tên

CREATE PROCEDURE spFindLoaiSachByName

AS

BEGIN

SELECT TenLoaiSach FROM LoaiSach

END

GO

- Tìm kiếm loại sách theo tên và id

CREATE PROCEDURE spFindLoaiSachByIDAndName

AS

BEGIN

SELECT ID, TenLoaiSach FROM LoaiSach

END

GO

### 3.4.4. Stored procedure tìm kiếm sách

- Tìm kiếm sách theo loại sách

CREATE PROCEDURE spFindSachByNameLoaiSach

@ID\_LoaiSach bigint

AS

BEGIN

SELECT TenSach FROM Sach WHERE ID\_LoaiSach=@ID\_LoaiSach

END

- Tìm kiếm sách theo tên

CREATE PROCEDURE spFindSachByName

@TenSach nvarchar(200)

AS

BEGIN

SELECT TenSach,GiaBan, SoLuongTonKho FROM Sach WHERE TenSach LIKE '%' + @TenSach + '%'

END

- Tìm kiếm sách theo tên sách, tên loại sách, tên tác giả, tên nhà xuất bản

CREATE PROCEDURE spFindSach

@TenSach nvarchar(200),

@TenLoaiSach nvarchar(200),

@TenTacGia nvarchar(50),

@TenNXB nvarchar(200)

AS

BEGIN

SELECT \* FROM viewThongTinSach

WHERE TenSach LIKE '%' + @TenSach + '%'

OR TenLoaiSach LIKE '%' + @TenLoaiSach + '%'

OR TenTacGia LIKE '%' + @TenTacGia + '%'

OR TenNXB LIKE '%' + @TenNXB + '%'

END

### 3.4.5. Stored procedure lấy thông tin cho hóa đơn bán hàng

- Lấy id và tên nhân viên

CREATE PROCEDURE spGetNameAndIdNhanVien

AS

BEGIN

SELECT ID, HoTen FROM NhanVien

END

- Lấy id và tên thành viên

CREATE PROCEDURE spGetNameAndIdThanhVien

AS

BEGIN

SELECT ID, HoTen FROM ThanhVien

END

- Lấy id sách

CREATE PROCEDURE spGetIDSachByName

@TenSach nvarchar(200)

AS

BEGIN

SELECT ID FROM Sach WHERE TenSach=@TenSach

END

GO

- Lấy số lượng tồn kho của sách

CREATE PROCEDURE spGetSoSachTonKho

@TenSach nvarchar(200)

AS

BEGIN

SELECT SoLuongTonKho FROM Sach WHERE TenSach = @TenSach

END

GO

- Lấy thông tin giá bán của sách

CREATE PROCEDURE spGetGiaBanByName

@TenSach nvarchar(200)

AS

BEGIN

SELECT GiaBan FROM Sach WHERE TenSach = @TenSach

END

GO

- Lấy thông tin mức ưu đãi đối với mỗi thành viên

CREATE PROCEDURE spGetGiamGiaByTenThanhVien

@HoTen nvarchar(50)

AS

BEGIN

SELECT MucUuDai

FROM ThanhVien, LoaiThanhVien

WHERE ThanhVien.ID\_LoaiTV=LoaiThanhVien.ID AND ThanhVien.HoTen = @HoTen

END

CREATE PROCEDURE spGetGiamGiaByIDThanhVien

@ID bigint

AS

BEGIN

SELECT MucUuDai

FROM ThanhVien, LoaiThanhVien

WHERE ThanhVien.ID\_LoaiTV=LoaiThanhVien.ID AND ThanhVien.ID = @ID

END

GO

- Lấy thông tin giảm giá

CREATE PROCEDURE spGetGiamGiaByNgay

@ngay datetime

AS

BEGIN

SELECT ID, NgayBatDau, NgayKetThuc, PhanTramGG

FROM GiamGia

WHERE NgayBatDau<= @ngay AND @ngay<=NgayKetThuc

END

### 3.4.6. Stored procedure tìm hóa đơn

- Tìm kiếm hóa đơn theo ngày mua

CREATE PROCEDURE spFindHoaDonByNgay

@NgayMua DateTime

AS

BEGIN

SELECT \* FROM HoaDon WHERE NgayMua = @NgayMua

END

- Tìm kiếm hóa đơn bằng số điện thoại thành viên

CREATE PROCEDURE spFindHoaDonByThanhVien

@SDT VARCHAR(11)

AS

BEGIN

SELECT \* FROM HoaDon

WHERE ID in (SELECT HoaDon.ID FROM ThanhVien, HoaDon

WHERE ThanhVien.ID = HoaDon.ID\_TV AND ThanhVien.SDT = @SDT)

END

### 3.4.7. Stored procedure tìm phiếu nhập

- Tìm kiếm hóa đơn theo ngày mua

CREATE PROCEDURE spFindPhieuNhapByNgay

@NgayNhap DateTime

AS

BEGIN

SELECT \* FROM PhieuNhap WHERE NgayNhap = @NgayNhap

END

- Tìm kiếm hóa đơn bằng số điện thoại thành viên

CREATE PROCEDURE spFindPhieuNhapByNXB

@SDT VARCHAR(11)

AS

BEGIN

SELECT \* FROM PhieuNhap

WHERE ID in (SELECT PhieuNhap.ID FROM NXB, PhieuNhap

WHERE NXB.ID = PhieuNhap.ID\_NXB AND NXB.SDT = @SDT)

END

## 3.5. Cài đặt các Functions

### 3.5.1. Function tìm kiếm ID nhân viên

CREATE FUNCTION f\_TimKiemIDNV

(@ID varchar(10) )

RETURNS TABLE

RETURN (select \* from Nhanvien where ID like '%'+@ID+'%');

GO

### 3.5.2. Function tìm kiếm tên nhân viên

CREATE FUNCTION f\_TimKiemTenNV

(@HoTen nvarchar(50) )

RETURNS TABLE

RETURN (select \* from Nhanvien where HoTen like '%'+@HoTen+'%');

GO

### 3.5.3. Function tìm kiếm username của nhân viên

CREATE FUNCTION f\_TimKiemUserNameNV

(@UserName nvarchar(50) )

RETURNS TABLE

RETURN (select \* from Nhanvien where UserName like '%' + @UserName + '%' );

GO

### 3.5.4. Function Check Login

CREATE FUNCTION f\_checkLoginUseUsernameAndPassword

(

@Username NVARCHAR(50),

@Password NVARCHAR(50)

)

RETURNS TABLE

AS

RETURN

(

SELECT \* FROM NhanVien WHERE Username = @Username AND Password = @Password

)

### 3.5.5. Function lấy Password bằng ID

CREATE FUNCTION f\_getPassword

(@ID varchar(10) )

RETURNS TABLE

RETURN (select Password from Nhanvien where ID =@ID);

### 3.5.6. Function tính tổng doanh thu theo tháng

CREATE FUNCTION funCountTotalbyMonth ()

RETURNS TABLE AS

RETURN

select MONTH(NgayMua) AS 'Thang', SUM(ThanhTien) as 'Doanhthu'

from HoaDon

group by MONTH(NgayMua)

GO

### 3.5.7. Funtion tính doanh thu đối với mỗi thành viên

CREATE FUNCTION funCountTotalbyThanhVien ()

RETURNS TABLE AS

RETURN

select HoaDon.ID\_TV, ThanhVien.HoTen, sum(HoaDon.ThanhTien) as DoanhThu

from HoaDon, ThanhVien

where HoaDon.ID\_TV = ThanhVien.ID

group by HoaDon.ID\_TV, ThanhVien.HoTen

GO

### 3.5.8. Funtion tính số lượng sách bán được

CREATE FUNCTION funSoLuongSachBanDuoc ()

RETURNS TABLE AS

RETURN

select ChiTietHoaDon.ID\_Sach, TenSach, sum(ChiTietHoaDon.SoLuong) as slBanDuoc

from Sach, ChiTietHoaDon

where Sach.ID = ChiTietHoaDon.ID\_Sach

group by ChiTietHoaDon.ID\_Sach, Sach.TenSach

GO

### 3.5.9. Function xem chi tiết hóa đơn

CREATE FUNCTION funXemChiTietHoaDon (@ID bigint)

RETURNS TABLE AS

RETURN

( SELECT \* FROM viewChiTietHoaDon WHERE ID\_HD = @ID)

GO

### 3.5.10. Function xem chi tiết phiếu nhập

CREATE FUNCTION funXemChiTietPhieuNhap(@ID bigint)

RETURNS TABLE AS

RETURN

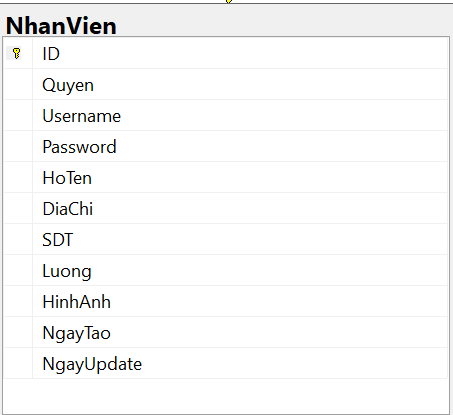
( SELECT \* FROM viewChiTietPhieuNhap WHERE ID\_PN = @ID)

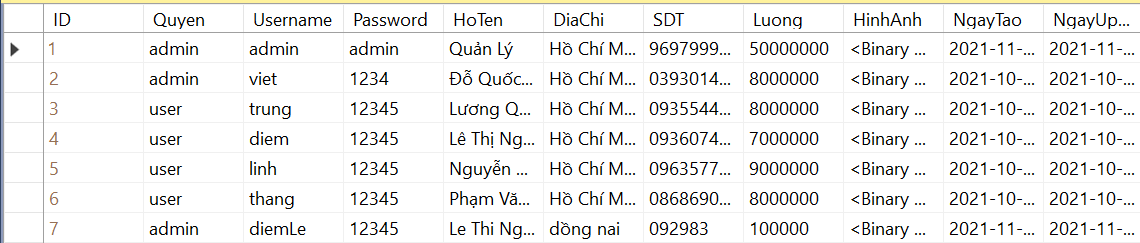
GO

## 3.6. Tạo User và phân quyền

### 3.6.1. Bảng cơ sở dữ liệu dùng để phân quyền

Đối với đồ án, người có quyền truy cập vào CSDL gồm có nhân viên và người quản lý cửa hàng, hai đối tượng này có quyền khác nhau trên CSDL, được quy định cụ thể trong bảng NhanVien





Đề tài này tạo ra 2 loại tài khoản: user (có quyền select trên các bảng ngoại trừ bảng nhân viên) và admin (có quyền select, insert, delete, exec trên tất cả các bảng).

### 3.6.2. Phân quyền cho các user

Đề tài đã phân rõ quyền đối với từng loại tài khoản và chỉ thuộc 1 trong 2 đối tượng là user và admin, nên thay vì gán quyền đối với từng tài khoản riêng biệt thì có thể tạo ra role, sau đó gán user vào role:

* Đối với tài khoản user gán vào roleUser:

Create Role roleUser

Grant select To roleUser

deny select ON nhanvien To roleUser

* Đối với tài khoản admin gán vào roleAdmin:

Create Role roleAdmin

Grant select, insert, update, delete,exec To roleAdmin

Khi một nhân viên được thêm vào bảng ta đồng thời sẽ tạo tài khoản login, tài khoản user và gán user vào role tương ứng với quyền đã cấp. Nhân viên có thể dùng tài khoản này để đăng nhập vào CSDL với username và password tương ứng.

CREATE TRIGGER trg\_addUser ON NhanVien AFTER INSERT AS

BEGIN

DECLARE @Quyen char(10)

DECLARE @Username varchar(50)

DECLARE @Password varchar(50)

DECLARE @ID bigint

Select @ID = ID FROM inserted WHERE NOT EXISTS (select ID from deleted)

Select @Quyen = Quyen FROM inserted WHERE ID=@ID

declare @t nvarchar(4000)

DECLARE @Roleuser varchar(50)

SET @Roleuser ='roleUser'

IF @Quyen = 'admin' set @Roleuser='roleAdmin'

Select @Username = Username FROM inserted WHERE ID=@ID

Select @Password = Password FROM inserted WHERE ID=@ID

SET @t = N'CREATE LOGIN ' + QUOTENAME(@username) + ' WITH PASSWORD = ' + QUOTENAME(@password, '''')

exec sys.sp\_executesql @t

set @t = N'CREATE USER ' + QUOTENAME(@username) + ' FOR LOGIN ' +QUOTENAME(@username)

exec sys.sp\_executesql @t

set @t = N'ALTER ROLE ' + QUOTENAME(@Roleuser) + ' ADD MEMBER ' + QUOTENAME(@username)

exec sys.sp\_executesql @t

END

Khi một nhân viên chỉnh sửa lại quyền, username, password thì sẽ thu hồi lại quyền, user và login của nhân viên này trước đó và tạo lại tài khoản login, tài khoản user và gán user vào role tương ứng với quyền đã cấp. Nhân viên chỉ có thể dùng tài khoản mới này để đăng nhập vào CSDL với username và password tương ứng.

CREATE TRIGGER trg\_UpdateUser ON NhanVien AFTER UPDATE AS

BEGIN

DECLARE @Quyen char(10)

DECLARE @Username varchar(50)

DECLARE @Password varchar(50)

DECLARE @ID bigint

Select @ID = inserted.ID FROM inserted, deleted WHERE inserted.Quyen<>deleted.Quyen or inserted.Username<>deleted.Username or inserted.Password<>deleted.Password

Select @Quyen = Quyen FROM inserted WHERE ID=@ID

declare @t nvarchar(4000)

DECLARE @Roleuser varchar(50)

set @t = N'ALTER ROLE ' + 'roleUser' + ' DROP MEMBER ' + QUOTENAME(@username)

exec sys.sp\_executesql @t

set @t = N'ALTER ROLE ' + 'roleAdmin' + ' DROP MEMBER ' + QUOTENAME(@username)

exec sys.sp\_executesql @t

SET @Roleuser ='roleUser'

IF @Quyen = 'admin' set @Roleuser='roleAdmin'

exec sys.sp\_executesql @t

Select @Username = Username FROM deleted WHERE ID=@ID

set @t = N'DROP USER ' + QUOTENAME(@username)

exec sys.sp\_executesql @t

SET @t = N'DROP LOGIN ' + QUOTENAME(@username)

exec sys.sp\_executesql @t

Select @Username = Username FROM inserted WHERE ID=@ID

Select @Password = Password FROM inserted WHERE ID=@ID

SET @t = N'CREATE LOGIN ' + QUOTENAME(@username) + ' WITH PASSWORD = ' + QUOTENAME(@password, '''')

exec sys.sp\_executesql @t

set @t = N'CREATE USER ' + QUOTENAME(@username) + ' FOR LOGIN ' +QUOTENAME(@username)

exec sys.sp\_executesql @t

set @t = N'ALTER ROLE ' + QUOTENAME(@Roleuser) + ' ADD MEMBER ' + QUOTENAME(@username)

exec sys.sp\_executesql @t

END

Khi một nhân viên bị xóa trên bảng Nhân Viên thì sẽ thu hồi lại quyền, user và login của nhân viên này trước đó. Nhân viên đó ki này không thể tiếp tục truy vào CSDL.

CREATE TRIGGER trg\_DeleteUser ON NhanVien AFTER DELETE AS

BEGIN

DECLARE @Username varchar(50)

DECLARE @ID bigint

Select @ID = ID from deleted

declare @t nvarchar(4000)

set @t = N'ALTER ROLE ' + 'roleUser' + ' DROP MEMBER ' + QUOTENAME(@username)

exec sys.sp\_executesql @t

set @t = N'ALTER ROLE ' + 'roleAdmin' + ' DROP MEMBER ' + QUOTENAME(@username)

exec sys.sp\_executesql @t

Select @Username = Username FROM deleted WHERE ID=@ID

set @t = N'DROP USER ' + QUOTENAME(@username)

exec sys.sp\_executesql @t

SET @t = N'DROP LOGIN ' + QUOTENAME(@username)

exec sys.sp\_executesql @t

END

# Chương 4: THIẾT KẾ GIAO DIỆN

Các form chính:

- Form login

Graphical user interface, application, Teams

Description automatically generated

- Form bán hàng

Graphical user interface, text, application, email

Description automatically generated

- Form quản lý hóa đơn

Graphical user interface, application

Description automatically generated

- Form quản lý phiếu nhập

Graphical user interface

Description automatically generated

- Form quản lý kho sáchGraphical user interface

Description automatically generated

- Form quản lý giảm giáGraphical user interface

Description automatically generated

- Form quản lý thành viên

Graphical user interface

Description automatically generated

- Form Doanh thu

Chart, bar chart

Description automatically generated

- Form quản lý nhân viên

Graphical user interface

Description automatically generated

Graphical user interface

Description automatically generated

Graphical user interface

Description automatically generated

Graphical user interface, application

Description automatically generated

# KẾT LUẬN

## 1. Kết quả đạt được

Sau thời gian nghiên cứu và thực hiện các yêu cầu chính của đề tài, cơ bản đã hoàn tất với những nội dung chủ yếu sau:

- Xây dựng được các chức năng cần thiết cho hệ thống quản lý nhà sách, có giao diện thân thiện, thuận tiện cho người dùng.

- Chức năng cập nhật (thêm, sửa, xóa), tìm kiếm được thực hiện nhanh chóng.

- Biết các sử dụng trigger, function, stored procedure trong hệ quản trị cơ sở dữ liệu và vận dụng vào lập trình Winform, ADO.NET trong C#.

## 2. Hạn chế

Vì sản phẩm được làm trong thời gian ngắn, kinh nghiệm còn hạn chế nên còn nhiều điểm chưa khắc phục được.

- Khả năng xử lý nghiệp vụ chưa linh hoạt nhạy bén.

- Chưa giải quyết trọn vẹn các vấn đề nảy sinh trong quá trình quản lý

- Một số chức năng của hệ thống chưa hoàn thiện.

## 3. Hướng phát triển

Nghiên cứu sâu hoàn thiện thêm các chức năng của sản phẩm, làm cho phần mềm có ích lĩnh vực quản lý.

Thêm các chức năng như cập nhật video, thông tin về sách. Có thể sử dụng song song bởi nhiều nhân viên và cùng một server.

# TÀI LIỆU THAM KHẢO

[1] <https://sites.google.com/site/chiase799/k12_b1>, truy cập ngày 10/11/2021

[2] <https://www.dtechvn.com/tong-quan-ve-sql-server.html>, truy cập ngày 13/11/2021

[3] <https://plctech.com.vn/winform-la-gi/>, truy cập ngày 01/11/2021

[4] <https://tranphuochung.wordpress.com/2009/12/23/ado-net-bai-1-gi%E1%BB%9Bi-thi%E1%BB%87u/>, truy cập ngày 02/11/2021

# PHỤ LỤC

## Hướng dẫn sử dụng chương trình

- Chức năng Đăng nhập: nhập uername, password (đã có trong database) 🡪 login để đăng nhập vào hệ thống

Graphical user interface, application

Description automatically generated

Giao diện sau khi đăng nhập (phân quyền rõ ràng giữa user với admin)

Graphical user interface, text, application

Description automatically generated

- Chức năng bán hàng: Nhân viên ấn nút tìm kiếm để tìm sách, sau đó chọn sách, số lượng sách để thêm vô hóa đơn. Nhân viên chọn khách hàng, hệ thống sẽ tự động tính tiền, tính giảm giá, sau khi ấn xuất hóa đơn sẽ lưu vô database.

Graphical user interface, text

Description automatically generated

- Chức năng Quản lý hóa đơn: Xem chi tiết hóa đơn: Chọn hóa đơn 🡪 View Details để xem chi tiết hóa đơn.

Graphical user interface

Description automatically generated

Tìm kiếm: Nhập số điện thoại thành viên vào thanh tìm kiếm và nhấn nút Tìm kiếm để tìm kiếm hóa đơn theo số điện thoại, hoặc chọn ngày mua và nhấn nút Tìm để tìm kiếm hóa đơn theo ngày mua, hoặc có thể kết hợp tìm kiếm bằng số điện thoại và ngày mua.

Graphical user interface, application

Description automatically generated- Chức năng quản lý Phiếu nhập:

Xem chi tiết phiếu nhập: tương tự như xem chi tiết hóa đơn, chọn phiếu nhập, sau đó nhấn View Details để xem chi tiết phiếu nhập.

Thêm phiếu nhập: chọn Add Receipt để thêm phiếu nhập 🡪 chọn tên NXB 🡪 chọn sách 🡪 chọn Save để lưu phiếu nhập và chi tiết phiếu nhập.

A screenshot of a computer

Description automatically generated

Tìm kiếm: Nhập số điện thoại nhà xuất bản vào thanh tìm kiếm và nhấn nút Tìm kiếm để tìm kiếm phiếu nhập theo số điện thoại, hoặc chọn ngày nhập và nhấn nút Tìm để tìm kiếm phiếu nhập theo ngày nhập, hoặc có thể kết hợp tìm kiếm bằng số điện thoại và ngày nhập.

Graphical user interface

Description automatically generated

- Chức năng Quản lý giảm giá, thêm xóa sửa giảm giá, click vô dòng giảm giá để xem chi tiết và chỉnh sửa

Graphical user interface

Description automatically generated

- Chức năng Quản lý sách: quản lý thể loại, tác giả, nhà xuất bản bằng các hành động như thêm, xóa, sửa và có chức năng xem top sell (top bán chạy)

Chọn Add 🡪 Nhập thông tin sách 🡪 Save: để thêm sách

Chọn Update 🡪 Sửa thông tin sách 🡪 Save: để cập nhật sách

Chọn sách để xóa 🡪 Hiển thị thông báo 🡪 Chọn Ok: để xóa sách

Nhập tên sách hoặc tên tác giả hoặc tên loại sách hoặc tên nhà xuất bản vào thanh tìm kiếm và nhấn Tìm kiếm để tìm kiếm sách

Graphical user interface

Description automatically generated

- Chức năng Quản lý thành viên, click vô thành viên để sửa hoặc xóa, và cũng có thể thêm thành viên

Graphical user interface

Description automatically generated

- Chức năng Thống kê doanh thu theo tháng của cửa hàng, doanh thu theo thành viên

Chart, bar chart, box and whisker chart

Description automatically generated

Graphical user interface

Description automatically generated

- Chức năng Quản lý nhân viên: Sẽ hiện thông tin tất cả nhân viên, có thể thực hiện thao tác thêm, sửa xóa trên form

Graphical user interface

Description automatically generated

Trong trường hợp chỉ cần sửa đổi lương nhân viên có thể tiến hành sửa nhanh trên form

Graphical user interface

Description automatically generated

Sẽ hiện thông tin tất cả nhân viên, có thể thực hiện thao tác thêm, sửa xóa trên form

Graphical user interface

Description automatically generated

Chức năng đổi mật khẩu: Nhập mã nhân viên, sau đó nhập lại password hiện đang sử dụng, password mới, xác nhận lại password nếu tất cả đều hợp lệ thì sẽ có thông báo thành công. Ngược lại trả ra lỗi

Graphical user interface, application

Description automatically generated