

**BỘ GIÁO DỤC VÀ ĐÀO TẠO
TRƯỜNG ĐẠI HỌC NGUYỄN TẤT THÀNH
KHOA CÔNG NGHỆ THÔNG TIN**



ĐỒ ÁN MÔN HỌC

**XÂY DỰNG HỆ THỐNG QUẢN LÝ CỬA HÀNG
BÁN ĐỒNG HỒ TREO TƯỜNG**

Giảng viên giảng dạy : ThS. Giang Hào Côn

Sinh viên thực hiện : Đặng Quốc Lai

MSSV : 1811545103

Lớp : 18DTH1A

Chuyên ngành : Kỹ thuật phần mềm

Môn học : Chuyên đề .NET

Khóa : 2018

TP. Hồ Chí Minh, tháng 9 năm 2020.

**BỘ GIÁO DỤC VÀ ĐÀO TẠO
TRƯỜNG ĐẠI HỌC NGUYỄN TẤT THÀNH
KHOA CÔNG NGHỆ THÔNG TIN**



ĐỒ ÁN MÔN HỌC

**XÂY DỰNG HỆ THỐNG QUẢN LÝ CỬA HÀNG
BÁN ĐỒNG HỒ TREO TƯỜNG**

Giảng viên giảng dạy : ThS. Giang Hào Côn
Sinh viên thực hiện : Đặng Quốc Lai
MSSV : 1811545103
Lớp : 18DTH1A
Chuyên ngành : Kỹ thuật phần mềm
Môn học : Chuyên đề .NET
Khóa : 2018

TP. Hồ Chí Minh, tháng 9 năm 2020.

NHIỆM VỤ ĐỒ ÁN MÔN HỌC

Họ và tên	: ĐẶNG QUỐC LAI	MSSV	: 1811545103
Chuyên ngành	: Kỹ thuật phần mềm	Lớp	: 18DTH1A
Email	: danglai.mail@gmail.com	SĐT	: 077.247.0922
Tên đề tài	: XÂY DỰNG HỆ THỐNG QUẢN LÝ CỬA HÀNG BÁN ĐỒNG HỒ TREO TƯỜNG		
Giảng viên giảng dạy	: ThS. Giang Hào Côn		
Thời gian thực hiện	: Từ ngày 09/07/2020 đến 29/09/2020 .		

Nhiệm vụ/nội dung (mô tả chi tiết nội dung, yêu cầu, phương pháp...):

- Thiết kế cơ sở dữ liệu cho phần mềm.
- Thiết lập được mô hình MVC để kết nối và thao tác với cơ sở dữ liệu SQL.
- Lập mô hình phân rã chức năng, mô hình dữ liệu quan hệ.
- Xây dựng và thiết kế giao diện cho các chức năng:
 - + Đăng nhập.
 - + Hiện thị danh sách sản phẩm trong cửa hàng.
 - + Chủ cửa hàng có thể thêm, sửa, xóa, quản lý các nhân viên trong cửa hàng.
 - + Và các màn hình ứng với các chức năng khác.
- Xây dựng báo cáo quản lý tồn kho sản phẩm.
- Đóng gói và xuất bản ứng dụng.
- Viết báo cáo đồ án, in và đóng cuốn theo biểu mẫu qui định.

Nội dung và yêu cầu đã được thông qua Bộ môn.

TP. HCM, ngày ... tháng ... năm 2020

TRƯỞNG BỘ MÔN
(Ký tên)

GIẢNG VIÊN GIẢNG DẠY
(Ký tên)

ThS. Phạm Văn Đăng

ThS. Giang Hào Côn

LỜI CẢM ƠN

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

Sinh viên thực hiện
(Ký tên)

ĐẶNG QUỐC LAI

NHẬN XÉT CỦA GIẢNG VIÊN HƯỚNG DẪN

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

TP. Hồ Chí Minh, ngày ... tháng ... năm 2020

GIẢNG VIÊN HƯỚNG DẪN

(Ký tên)

ThS. Giang Hào Côn

DANH SÁCH HÌNH

Hình 3.1 Mô hình phân rã chức năng	16
Hình 3.2 Mô hình quan hệ dữ liệu	17
Hình 3.3 Màn hình đăng nhập ban đầu	18
Hình 3.4 Màn hình đăng nhập sai	18
Hình 3.5 Màn hình thông báo đăng nhập sai	18
Hình 3.6 Màn hình chính đối với tài khoản admin	19
Hình 3.7 Màn hình chính đối với tài khoản nhân viên thường	19
Hình 3.8 Giao diện in hóa đơn khi chưa thêm sản phẩm vào giỏ hàng	20
Hình 3.9 Màn hình điền thông tin xuất hóa đơn	20
Hình 3.10 Màn hình in hóa đơn	21
Hình 3.11 Màn hình kiểm tra danh sách hóa đơn	22
Hình 3.12 Màn hình xem danh sách khách hàng	22
Hình 3.13 Màn hình quản lý nhân viên	23
Hình 3.14 Màn hình xem, sửa thông tin của nhân viên	23
Hình 3.15 Màn hình yêu cầu mật khẩu khi muốn thay đổi thông tin nhân viên	24
Hình 3.16 Màn hình đổi mật khẩu	24

DANH SÁCH TỪ VIẾT TẮT

STT	Từ viết tắt	Ý nghĩa
1	CSDL	Cơ sở dữ liệu
2	MVC	Model – Control - View

PHÂN CÔNG NHIỆM VỤ

Sinh viên 1	Sinh viên 2
MSSV: 1811545103	
Họ tên: Đặng Quốc Lai	
<p>Nhiệm vụ:</p> <ul style="list-style-type: none">- Thiết kế cơ sở dữ liệu cho phần mềm.- Thiết lập được mô hình MVC để kết nối và thao tác với cơ sở dữ liệu SQL.- Lập mô hình phân rã chức năng, mô hình dữ liệu quan hệ.- Xây dựng và thiết kế giao diện cho các chức năng:<ul style="list-style-type: none">+ Đăng nhập.+ Hiện thị danh sách sản phẩm trong cửa hàng.+ Chủ cửa hàng có thể thêm, sửa, xóa, quản lý các nhân viên trong cửa hàng.+ Và các màn hình ứng với các chức năng khác.- Xây dựng báo cáo quản lý tồn kho sản phẩm.- Đóng gói và xuất bản ứng dụng.- Viết báo cáo đồ án, in và đóng cuốn theo biểu mẫu qui định.	

MỤC LỤC

CHƯƠNG 1: GIỚI THIỆU.....	4
1.1. Cơ sở khoa học và tính thực tiễn của đề tài	4
1.2. Mục tiêu đề tài	4
1.3. Phương pháp nghiên cứu	4
1.4. Phạm vi nghiên cứu	4
1.5. Kết quả đạt được	5
CHƯƠNG 2: CƠ SỞ LÝ THUYẾT	6
2.1. Giới thiệu lập trình giao diện .NET Framework.....	6
2.1.1. Khái niệm lập trình giao diện	6
2.1.2. Khái niệm .NET Framework	6
2.1.3. Kiến trúc .NET Framework	6
2.1.4. Mục đích thiết kế .NET Framework	7
2.1.5. Những đặc tính của .NET Framework.....	7
2.2. Ngôn ngữ lập trình C#	8
2.2.1. C# là gì?	8
2.2.2. Vai trò C# trong .Net FrameWork	8
2.2.3. Quá trình dịch chương trình C#	8
2.2.4. Biến và Kiểu dữ liệu trong C#	9
2.2.5. Các loại ứng dụng C#	10
2.3. Các thành phần giao diện cơ bản	11
2.3.1. Label	11
2.3.2. Textbox	11
2.3.3. Checkbox:	11
2.3.4. CheckedListBox.....	12

2.3.5. RadioButton	12
2.3.6. ListBox.....	12
2.3.7. Button.....	13
2.4. Kết nối cơ sở dữ liệu.....	13
2.4.1. Khái niệm cơ sở dữ liệu:.....	13
2.4.2. Các mô hình kết nối cơ sở dữ liệu	13
2.4.3. ADO.NET	14
2.5. Xây dựng báo cáo bằng Crystal Reports	14
2.6. Đóng gói và xuất bản ứng dụng	15
CHƯƠNG 3: THIẾT KẾ VÀ CÀI ĐẶT PHẦN MỀM	16
3.1. Mô hình phân rã chức năng	16
3.2. Mô hình quan hệ dữ liệu	17
3.3. Các màn hình giao diện	18
3.3.1. Đăng nhập	18
3.3.2. Màn hình chính	19
3.3.3. Giao diện khi nhấn nút in.....	20
3.3.4. Màn hình điền thông tin để xuất hóa đơn	20
3.3.5. Màn hình in hóa đơn	21
3.3.6. Màn hình kiểm tra danh sách hóa đơn	22
3.3.7. Màn hình xem danh sách khách hàng	22
3.3.8. Màn hình quản lý nhân viên	23
3.3.9. Màn hình xem thông tin của nhân viên	23
3.4. Code các tính năng trong ứng dụng	24
3.4.1. Code cho button “Đăng nhập” ở form Đăng nhập.....	24
3.4.2. Code cho button xem danh sách sản phẩm	25
3.4.3. Phương thức addToPnMain	25

3.4.4. Code cho button xem danh sách hóa đơn	26
3.4.5. Code cho button xem danh sách khách hàng.....	26
3.4.6. Code cho button Lưu thông tin	26
3.4.7. Code cho button In hóa đơn.....	27
3.4.8. Code cho tính năng thêm mới nhân viên	28
3.4.9. Code lắng nghe sự kiện nhập tài khoản khi thêm nhân viên	28
3.4.10. Code lắng nghe sự kiện nhập số điện thoại ở form nhập khách hàng	29

CHƯƠNG 1: GIỚI THIỆU

1.1. Cơ sở khoa học và tính thực tiễn của đề tài

Ngày nay, khi cuộc sống con người ngày càng bận rộn thì thời gian là thứ vô cùng quý giá. Vì vậy, việc quản lý thời gian là một việc vô cùng cần thiết. Để quản lý thời gian tốt nhất, con người luôn dùng một vật dụng cơ bản là đồng hồ, đặc biệt, khi phạm vi sử dụng ở hộ gia đình thì một chiếc đồng hồ treo tường là thứ không thể thiếu. Vì nhu cầu mua đồng hồ treo tường tăng cao nên các cửa hàng bán đồng hồ treo tường, đặc biệt là cửa hàng nhiều chi nhánh chủ cửa hàng phải giám sát nhân viên, giám sát hàng hóa tồn kho và có thể thống kê doanh thu. Tuy nhiên nếu một người làm tất cả các công việc đó thì rất vất vả, vì vậy việc phát triển một phần mềm hỗ trợ chủ cửa hàng là rất cần thiết.

1.2. Mục tiêu đề tài

Xây dựng một phần mềm đầy đủ chức năng có thể giúp chủ cửa hàng quản lý thời gian nhân viên làm việc, tính lương nhân viên, quản lý hàng tồn kho, thống kê doanh thu từng chi nhánh, kiểm tra bảo hành cho sản phẩm,...

1.3. Phương pháp nghiên cứu

- Tham khảo quy trình ở các cửa hàng bán đồng hồ trong thực tế.
- Ghi chú lại các thông tin quan trọng.
- Phân tích kỹ các quy trình từ mua hàng, nhập hàng, theo dõi, tính giờ làm việc của nhân viên.
- Hỏi đáp các yêu cầu nghiệp vụ khác (nếu có).

1.4. Phạm vi nghiên cứu

- Nghiên cứu bắt đầu từ các cửa hàng đơn lẻ để tham khảo kỹ các thông tin về quy trình nhập hàng, xuất hàng, xuất hóa đơn.
- Sau khi có các nghiệp vụ về nhập xuất hàng, bắt đầu nghiên cứu đến các cửa hàng có hệ thống phân phối lớn, từ 2 – 3 chi nhánh khác nhau để khảo sát quy trình tính công cho nhân viên, đặc biệt các quy trình nghiệp vụ như cửa hàng chi nhánh gửi thông tin về cửa hàng chính để xuất hóa đơn đỏ khi khách hàng yêu cầu.
- Tiếp theo tìm hiểu đến các yêu cầu thực tế khác như: thêm khuyến mãi cho cửa hàng (theo thời gian thực) hoặc thêm khuyến mãi đối với khách hàng thân thiết, quá trình kiểm tra sản phẩm và thời hạn bảo hành.

1.5. Kết quả đạt được

Xây dựng được chức năng và cho các giao diện:

- + Đăng nhập.
- + Xem danh sách sản phẩm còn trong cửa hàng.
- + Xuất thông tin hóa đơn, In hóa đơn.
- + Gửi yêu cầu xuất hóa đơn đỏ về cho cửa hàng chính.
- + Nhân viên xem và sửa được thông tin của mình (như đổi mật khẩu, email, số điện thoại, xem được số ngày công mà mình đi làm).
- + Chủ cửa hàng có thể quản lý được giờ làm của nhân viên, kiểm tra nhân viên làm việc như thế nào, thống kê doanh thu và sản phẩm tồn của từng chi nhánh.
- + Nhân viên có thể kiểm tra thời gian bảo hành của sản phẩm.
- + Các nhân viên và chủ cửa hàng có thể trao đổi thông tin lẫn nhau ngay trên phần mềm và lưu vào cơ sở dữ liệu chung.
- + Cuối tháng, khi chủ cửa hàng yêu cầu, phần mềm sẽ thống kê doanh thu, tính thu, chi cho cửa hàng để quản lý.

CHƯƠNG 2: CƠ SỞ LÝ THUYẾT

2.1. Giới thiệu lập trình giao diện .NET Framework

2.1.1. Khái niệm lập trình giao diện

- Giao diện là điểm giao tiếp giữa hai đối tượng.
 - + Khi cả hai đối tượng đều là con người thì giao diện gọi là trung gian.
 - + Khi cả hai đối tượng giao tiếp đều là bộ phận phần mềm thì điểm giao tiếp giữa hai bên gọi là giao diện lập trình.
 - + Khi một bên là con người và bên kia là phần mềm thì điểm giao tiếp giữa hai bên gọi là giao diện điều khiển (giao diện đồ hoạ).
 - + Giao diện điều khiển mà không có đồ hoạ thì thường là giao diện dòng lệnh.
- Lập trình giao diện: là sử dụng ngôn ngữ lập trình tạo ra giao diện phần mềm ứng dụng hoặc giao diện website có thể truy cập trên internet.

2.1.2. Khái niệm .NET Framework

- Là một nền tảng lập trình tập hợp các thư viện lập trình có thể được cài thêm hoặc đã có sẵn trong các hệ điều hành Windows.
- .NET Framework cung cấp những giải pháp thiết yếu cho những yêu cầu thông thường của các chương trình điện toán như lập trình giao diện người dùng, truy cập dữ liệu, kết nối cơ sở dữ liệu, ứng dụng web, các giải thuật số học và giao tiếp mạng.
- .NET Framework quản lý việc thực thi các chương trình được viết dựa trên .NET Framework do đó người dùng cần phải cài .NET Framework để có thể chạy các chương trình được viết trên nền .NET.

2.1.3. Kiến trúc .NET Framework

Gồm 2 thành phần chính:

- Common Language Runtime (CLR):
 - + Gọi tắt là bộ thực thi được xây dựng trên các dịch vụ hệ điều hành. Chịu trách nhiệm thực hiện các ứng dụng và đảm bảo các phần liên quan đến ứng dụng đều được đáp ứng như quản lý bộ nhớ, an ninh bảo mật, tích hợp ngôn ngữ...

- + Bộ thực thi bao gồm nhiều dịch vụ hỗ trợ phát triển và triển khai ứng dụng cũng như cải thiện tính đáng tin cậy của ứng dụng.
- + CLR được gắn kèm với hệ điều hành. Điều này đảm bảo ứng dụng viết ra trên máy tính của bạn sẽ chạy trên các máy tính khác mà không cần cài đặt.
- NET Framework class library: là thành phần cung cấp thư viện lập trình như cho ứng dụng, cơ sở dữ liệu, dịch vụ web.
 - + Base class library: là thư viện các lớp cơ bản nhất, được dùng trong khi lập trình hay bản thân những người xây dựng .NET Framework cũng phải dùng nó để xây dựng các lớp cao hơn. Ví dụ các lớp trong thư viện này là String, Integer, Exception...
 - + ADO.NET và XLM: Bộ thư viện này gồm các lớp dùng để xử lý dữ liệu. ADO.NET thay thế ADO để trong việc thao tác với các dữ liệu thông thường. Các lớp đối tượng XML được cung cấp để bạn xử lý các dữ liệu theo định dạng mới như dạng XML.
- Web services: là các dịch vụ được cung cấp qua Web (hay Internet). Dịch vụ được coi là Web service không nhằm vào người dùng mà nhằm vào người xây dựng phần mềm. Web services còn dùng để cung cấp các dữ liệu hay một chức năng tính toán.
- Windows form: cung cấp các lớp đối tượng để xây dựng các ứng dụng Windows.

2.1.4. Mục đích thiết kế .NET Framework

- Sự hợp nhất thông qua các chuẩn Internet công cộng.
- Hỗ trợ nhiều ngôn ngữ.
- Nâng cao năng suất cho các nhà phát triển.
- Bảo vệ những sự đầu tư thông qua việc bảo mật đã được cải tiến.
- Tận dụng những dịch vụ của hệ điều hành.

2.1.5. Những đặc tính của .NET Framework

- Hỗ trợ các chuẩn dịch vụ Web XML
- Hiệu suất cho người phát triển

- Hỗ trợ những chuẩn Internet công cộng
- Bảo mật được cải thiện

2.2. Ngôn ngữ lập trình C#

2.2.1. C# là gì?

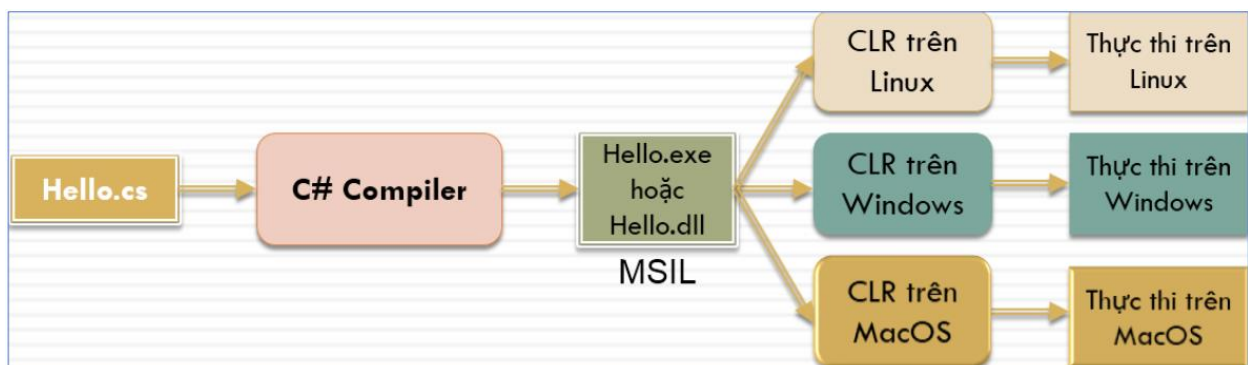
- C# (hay C sharp) là một ngôn ngữ lập trình đơn giản, được phát triển bởi đội ngũ kỹ sư của Microsoft vào năm 2000, trong đó người dẫn đầu là Anders Hejlsberg và Scott Wiltamuth.
- C# là ngôn ngữ lập trình hiện đại, hướng đối tượng và nó được xây dựng trên nền tảng của hai ngôn ngữ mạnh nhất là C++ và Java.
- C# được thiết kế cho Common Language Infrastructure (CLI), mà gồm Executable Code và Runtime Environment, cho phép chúng ta sử dụng các ngôn ngữ high-level đa dạng trên các nền tảng và cấu trúc máy tính khác nhau.

2.2.2. Vai trò C# trong .Net Framework

- NET runtime sẽ phổ biến và được cài trong máy client.
- C# tạo cơ hội cho tổ chức xây dựng các App Client/Server n-tier.
- Kết nối ADO.NET cho phép truy cập nhanh và dễ dàng với SQL Server, Oracle...
- Cách tổ chức .NET cho phép hạn chế những vấn đề phiên bản

2.2.3. Quá trình dịch chương trình C#

- Mã nguồn C# (tập tin *.cs) được biên dịch qua MSIL (file *.exe hoặc *.dll).
- MSIL được CLR thông dịch qua mã máy dùng kỹ thuật Just-in-time để tăng tốc độ.



Hình 2.1. Minh họa quá trình dịch chương trình C#

2.2.4. Biến và Kiểu dữ liệu trong C#

Kiểu	Biểu diễn	Dãy giá trị	Giá trị mặc định
bool	Giá trị Boolean	True hoặc False	False
byte	Kiểu unsigned integer (8 bit)	0 tới 255	0
char	Kiểu Unicode character (16 bit)	U +0000 tới U +ffff	'\0'
decimal	Kiểu thập phân (128 bit)	$(-7.9 \times 10^{28}$ tới $7.9 \times 10^{28}) / 10^0 - 28$	0.0M
double	Kiểu double (64 bit)	$(+/-)5.0 \times 10^{-324}$ tới $(+/-)1.7 \times 10^{308}$	0.0D
float	Kiểu float (32 bit)	-3.4×10^{38} tới $+3.4 \times 10^{38}$	0.0F
int	Kiểu integer (32 bit)	-2,147,483,648 tới 2,147,483,647	0
long	Kiểu signed integer (64 bit)	-9,223,372,036,854,775,808 tới 9,223,372,036,854,775,807	0L
sbyte	Kiểu signed integer (8 bit)	-128 tới 127	0
short	Kiểu signed integer (16 bit)	-32,768 tới 32,767	0
uint	Kiểu unsigned integer (32 bit)	0 tới 4,294,967,295	0
ulong	Kiểu unsigned integer (64 bit)	0 tới 18,446,744,073,709,551,615	0
ushort	Kiểu unsigned integer (16 bit)	0 tới 65,535	0

Bảng 2.1 Các kiểu dữ liệu trong C#

- Kiểu tham chiếu – reference type

- + Kiểu này không chứa dữ liệu thực sự mà nó chứa một sự tham khảo trong biến. Có thể tạo ra nhiều biến cùng tham chiếu đến một vùng nhớ.
- + Nếu dữ liệu trong vùng nhớ bị thay đổi bởi một biến nào đó, thì những biến khác cũng sẽ thay đổi giá trị một cách tự động.
- Kiểu object: là class nền tảng cơ bản cho mọi kiểu dữ liệu trong C# common type system (CTS). Object là một bí danh của class Object. Kiểu object có thể gán giá trị của mọi kiểu dữ liệu khác, kiểu giá trị, kiểu tham chiếu, kiểu định nghĩa trước hoặc do người dùng định nghĩa. Tuy nhiên trước khi gán giá trị, nó cần chuyển kiểu. Khi chuyển kiểu value sang object, đó được gọi là boxing và mặt khác, khi kiểu object được chuyển sang kiểu value được gọi là unboxing.
- Kiểu dynamic: có thể lưu bất kỳ kiểu giá trị nào trong biến kiểu dữ liệu động. Kiểm tra kiểu cho kiểu của biến dynamic diễn ra vào thời gian chạy. Tương tự kiểu object ngoại trừ kiểm tra kiểu cho kiểu biến object diễn ra tại thời điểm biên dịch, nhưng trái lại kiểm tra kiểu cho kiểu dynamic diễn ra vào thời gian chạy.
- Kiểu string: cho phép gán giá trị là chuỗi vào biến. Kiểu string là một bí danh của class String, nguồn gốc từ kiểu object. Giá trị gán sử dụng bằng cách quoted hoặc @quoted.

2.2.5. Các loại ứng dụng C#

- Chương trình Console
 - + Giao tiếp với người dùng bằng bàn phím
 - + Không có giao diện đồ họa (GUI)
- Chương trình Windows Form
 - + Giao tiếp với người dùng bằng bàn phím và mouse
 - + Có giao diện đồ họa và xử lý sự kiện
- Chương trình Web Form
 - + Kết hợp với ASP .NET, C# đóng vai trò xử lý bên dưới.
 - + Có giao diện đồ họa và xử lý sự kiện

2.3. Các thành phần giao diện cơ bản

2.3.1. Label

- Mục đích: hiển thị các thông tin không cho phép người dùng thay đổi dữ liệu.
- Thuộc tính:
 - + Name: tên label.
 - + Text: nội dung của label.
 - + TextAlign: căn lề.
- Phương thức:
 - + Hide(): ẩn label đi.
 - + Show(): hiện label.
 - + Click(): kích hoạt khi người dùng kích chuột vào label.
 - + DoubleClick(): kích hoạt khi người dùng nhấn đúp vào label.

2.3.2. Textbox

- Mục đích: nhận dữ liệu do người dùng nhập.
- Thuộc tính:
 - + Name: tên textbox.
 - + Multiline: true - cho phép nhập nhiều dòng.
 - + PasswordChar: ẩn dữ liệu bằng các ký tự được chỉ định.
 - + ReadOnly: true – không cho phép nhập liệu.
 - + Text: nội dung của textbox.
- Phương thức:
 - + TextChange(): theo dõi dữ liệu trong textbox khi bị thay đổi.
 - + Clear(): xóa nội dung textbox.
 - + Focus(): chỉ định con trỏ vào textbox.

2.3.3. Checkbox:

- Mục đích: yêu cầu người dùng nhập yes hoặc no.
- Thuộc tính:
 - + Appearance: hình dạng checkbox.

- + Checked: true – chọn.
- + CheckState: theo dõi trạng thái của điều khiển (Checked / Unchecked / Indeterminate: chọn / không chọn / vô định).

2.3.4. CheckedListBox

- Mục đích: hiển thị danh sách phân tử, mỗi phân tử có 1 ô checkbox bên cạnh.
- Thuộc tính:
 - + Items: Danh sách các phân tử trong checkedListBox.
 - + Appearance: hình dạng CheckedListBox.
 - + CheckOnClick: true – 1 click để chọn; false – 2 click để chọn.

2.3.5. RadioButton

- Mục đích: Là điều khiển cho phép người sử dụng có thể chọn tùy chọn trong danh sách, tuy nhiên trong khi CheckBox cho phép chọn nhiều tùy chọn cùng lúc thì RadioButton chỉ cho phép chọn 1 trong số các tùy chọn.
- Thuộc tính:
 - + Appearance: hình dạng radiobutton.
 - + Checked: true – trạng thái là đang chọn.

2.3.6. ListBox

- Mục đích: nhận thông tin từ người dùng bằng cách chọn giá trị từ danh sách.
- Thuộc tính:
 - + DataSource: dữ liệu cho listbox.
 - + DisplayMember: tên trường hiển thị trên listbox.
 - + DropDownStyle: kiểu trình bày danh sách phân tử.
 - + Items: tập các phân tử có trong điều khiển.
 - + ValueMember: giá trị ứng với khóa.
 - + SelectedText: lấy giá trị chuỗi đang chọn (dạng string).
 - + SelectedItem: lấy đối tượng đang chọn trong listbox.
 - + SelectedValue: lấy giá trị ứng với kiểu object đang chọn.
 - + SelectedIndex: lấy giá trị chỉ mục (kiểu int) ứng với phân tử đang chọn.
- Phương thức:
 - + SelectedIndexChanged: Xảy ra khi giá trị của thuộc tính SelectedIndex được sửa đổi.

- + `SelectedValueChanged`: Xảy ra khi giá trị của thuộc tính `SelectedValue` được sửa đổi

2.3.7. Button

- Mục đích: chờ đợi người dùng nhấn vào và thực hiện sự kiện.
- Thuộc tính:
 - + `FlatStyle`: kiểu đường viền của button.
 - + `Image`: hình ảnh hiển thị trên button.
 - + `Text`: nội dung bên trong button.
 - + `TextAlign`: canh lề cho text trong button.
- Phương thức:
 - + `Focus()`: đặt con trỏ vào điều khiển.
 - + `PerformClick()`: kích hoạt một sự kiện bấm cho một nút.
 - + `Click()`: xảy ra khi người dùng bấm vào điều khiển.
 - + `DoubleClick()`: Xảy ra khi người dùng bấm đúp vào điều khiển.

2.4. Kết nối cơ sở dữ liệu

2.4.1. Khái niệm cơ sở dữ liệu:

Cơ sở dữ liệu là một hệ thống các thông tin có cấu trúc, được lưu trữ trên các thiết bị lưu trữ nhằm thỏa mãn yêu cầu khai thác thông tin đồng thời của nhiều người sử dụng hay nhiều chương trình ứng dụng chạy cùng một lúc với những mục đích khác nhau.

Việc sử dụng hệ thống CSDL này sẽ khắc phục được những khuyết điểm của cách lưu trữ dưới dạng hệ thống tập tin, đó là:

- Giảm trùng lặp thông tin, đảm bảo tính nhất quán và toàn vẹn dữ liệu.
- Đảm bảo dữ liệu được truy xuất theo nhiều cách khác nhau, từ nhiều người khác nhau và nhiều ứng dụng khác nhau.
- Tăng khả năng chia sẻ thông tin.

2.4.2. Các mô hình kết nối cơ sở dữ liệu

- `Single-Tier Architecture`: Đây là loại đơn giản nhất, đơn lẻ cho một người dùng và tương đương với việc chạy một ứng dụng trên máy tính cá nhân. Tất cả các thành phần cần thiết để chạy ứng dụng được đặt trong nó. Giao diện người dùng, logic nghiệp vụ và lưu trữ dữ liệu đều nằm trên cùng một máy.

- Two-Tier Architecture: Ứng dụng người dùng (Windows Form, Asp.net hoặc Mobile apps), các giao tiếp chỉ xảy ra giữa Client – Server. Logic nghiệp vụ (dịch vụ web) có mặt trong cùng hệ thống, còn máy chủ và máy chủ cơ sở dữ liệu trong một máy chủ khác.
- Three-Tier Architecture Model-Control-View (MVC):
 - + Graphic User Interface (GUI): Lớp này là lớp hiển thị giao diện và các chức năng để người dùng cuối sử dụng.
 - + Business Logic Layer (BLL): Đây là lớp nhận các yêu cầu từ lớp GUI và truy xuất lên lớp Data để lấy thông tin và trả về GUI.
 - + Data Access Layer (DAL): Lớp này là lớp để truy xuất với CSDL, chỉ duy nhất lớp này được làm việc với database.
 - + Data Transfer Object Layer (DTO): Lớp này chỉ đóng vai trò phụ trong mô hình 3 lớp, dùng để định nghĩa các table trong database dạng đối tượng.

2.4.3. ADO.NET

- Công nghệ ADO.NET cho phép kết nối với các hệ thống cơ sở dữ liệu theo hai cách:
 - + ODBC (Open Database Connectivity).
 - + OLE DB (Object Linking and Embedding Database).
- ODBC và OLE DB là các trình điều khiển cơ sở dữ liệu được sử dụng để kết nối với nguồn dữ liệu. Bằng cách kết nối với cơ sở dữ liệu, chúng ta có thể truy xuất, thao tác hoặc cập nhật dữ liệu trong cơ sở dữ liệu.

2.5. Xây dựng báo cáo bằng Crystal Reports

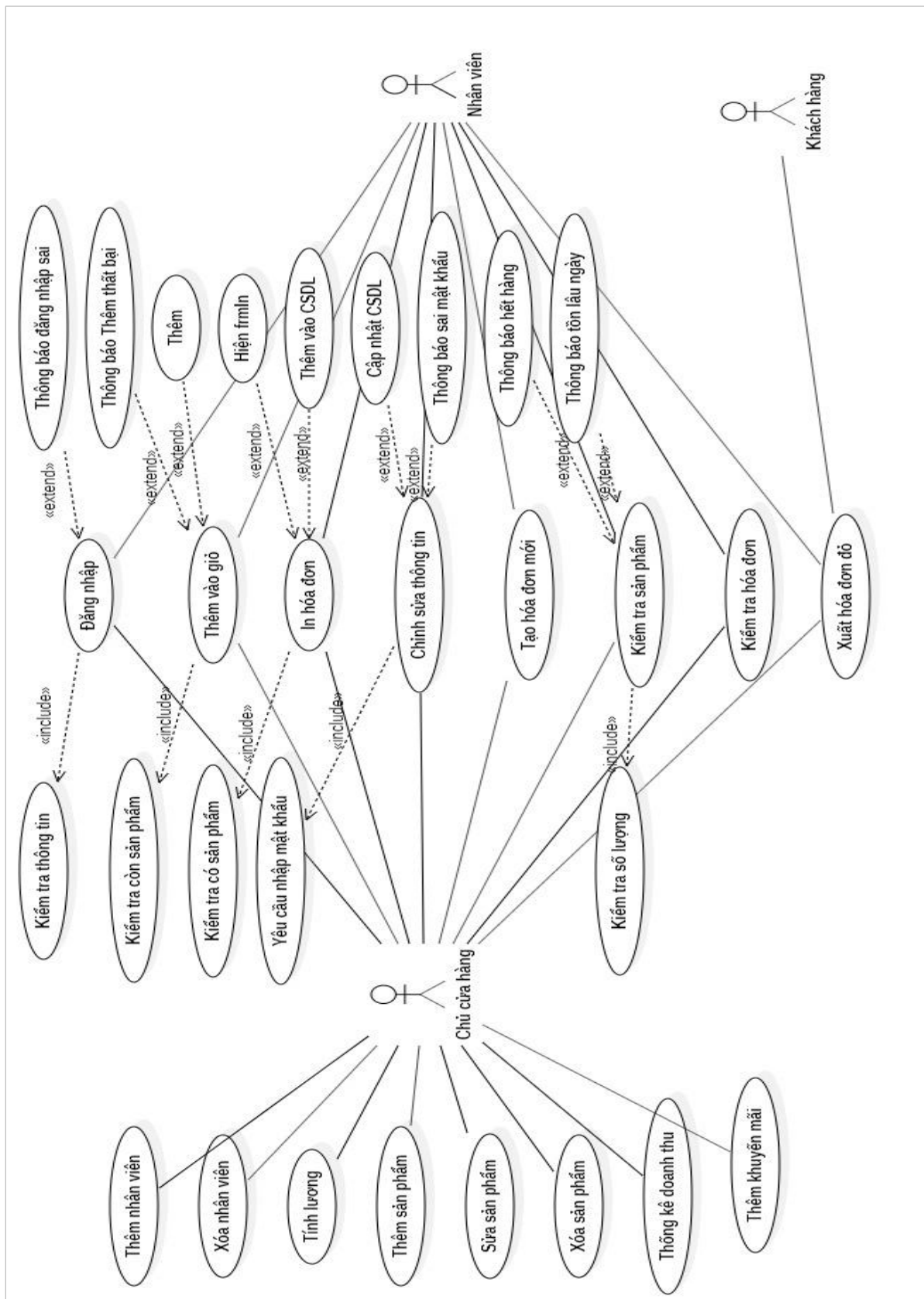
- Tải và cài đặt Crystal Reports cho phiên bản Visual Studio tương ứng.
- Ở trong Project cần xây dựng báo cáo, tạo 1 form mới để chứa thông tin báo cáo.
- Kéo thả Component ReportViewer từ Toolbox vào form báo cáo vừa tạo.
- Thiết kế giao diện ban đầu trên form báo cáo.
- Tạo 1 Dataset mô phỏng table (chứa các trường được lấy data từ cơ sở dữ liệu).
- Tạo 1 file Report (*.rdlc) để thiết kế báo cáo.
- Thiết kế báo cáo bằng cách kéo thả các trường trong dataset vào file report vừa tạo.
- Ở sự kiện form load của form bên trên, tiến hành các thao tác kết nối cơ sở dữ liệu và truyền giá trị, dạng tham số vào file report để có thể in báo cáo.

2.6. Đóng gói và xuất bản ứng dụng

- Các bạn cần tải và cài đặt gói Setup Projects.
- Mở Project, nhấn phải chuột vào Solution => Add => New Project.
- Chọn vào Visual Studio Installer => Setup Project. Sau đó đặt tên rồi chọn vị trí lưu cho File Exe sau này..
- Máy tính sẽ hiện 1 màn hình khác.
- Nhấn phải chuột vào Application Folder => Add => Project Outout.
- Chọn tên Project của bạn sau đó nhấn chọn Primary ouput => OK.
- Nhấn phải chuột vào Primary output mới tạo và chọn Create Shortcut to Primary output...
- Di chuyển Shortcut vừa tạo ở thư mục Application Folder vào thư mục User's Desktop rồi dán nó vào khoảng trống.
- Thiết lập icon cho File Exe: Nhấn phải chuột vào Shortcut chọn Properties Window...
- Vào mục Icon => Browse.
- Nhấp chọn vào thư mục Application Folder => Add File, chọn icon phù hợp.
- Chọn vào thư mục User's Programs Menu. Nhấn phải chuột vào khoảng trống bên phải chọn Create New Shortcut.
- Vào thư mục Application Folder => Primary output... => OK.
- Tương tự hãy thiết lập icon cho Shortcut như ở trên.
- Vào mục Properties => Author để chỉnh tên tác giả và mục Manufacturer để chỉnh tên nhà sản xuất.
- Rebuild lại File Setup.

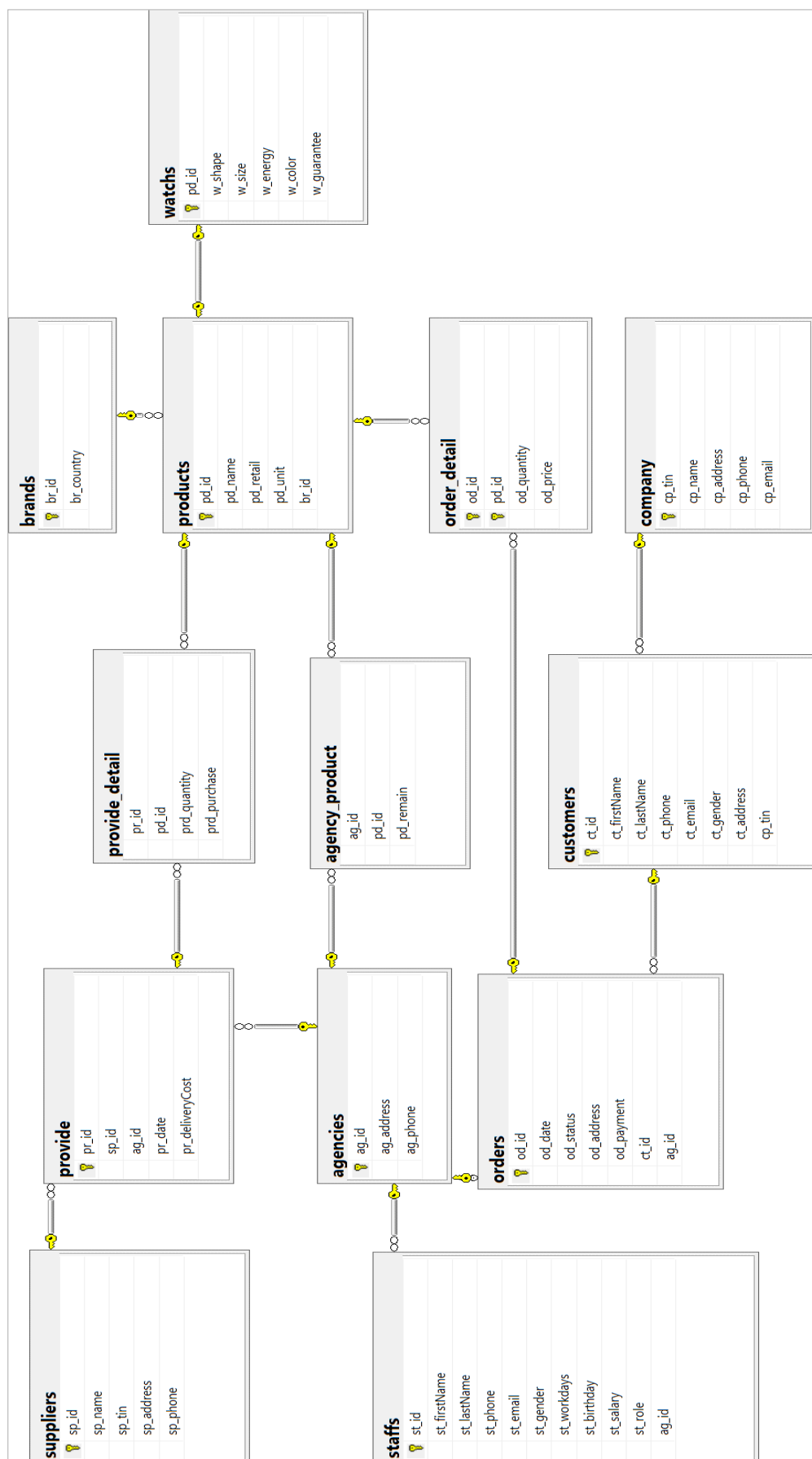
CHƯƠNG 3: THIẾT KẾ VÀ CÀI ĐẶT PHẦN MỀM

3.1. Mô hình phân rã chức năng



Hình 3.1 Mô hình phân rã chức năng

3.2. Mô hình quan hệ dữ liệu



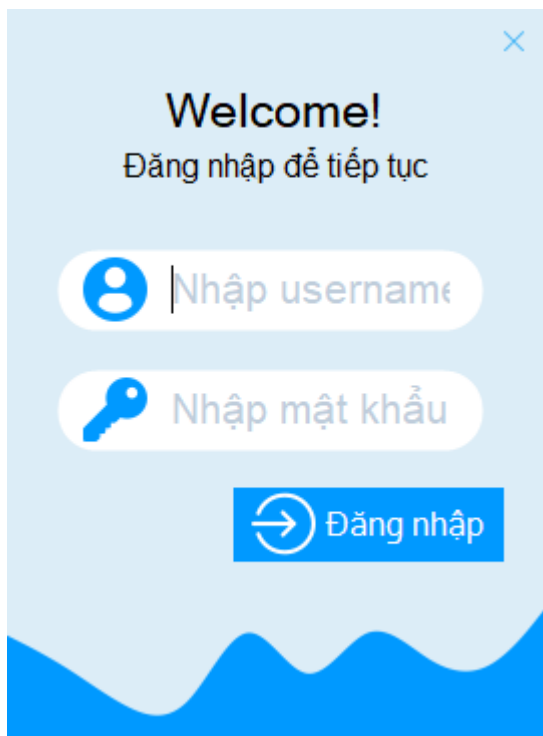
Hình 3.2 Mô hình quan hệ dữ liệu

3.3. Các màn hình giao diện

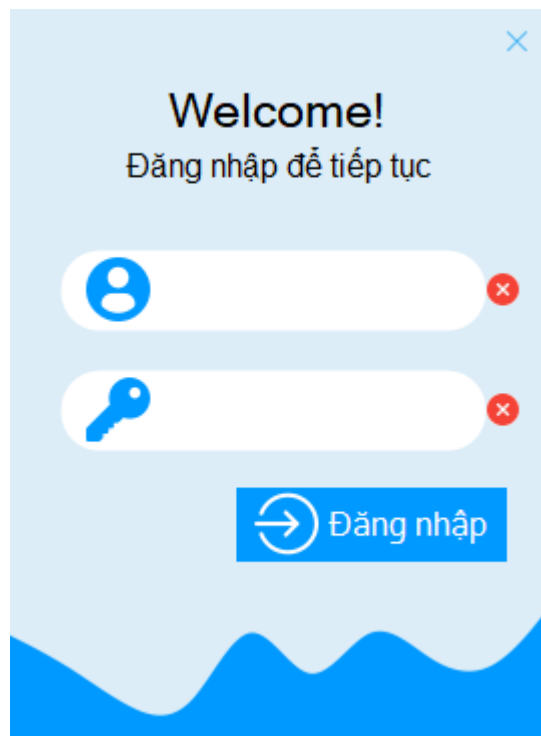
3.3.1. Đăng nhập

Nhập username/password lần lượt là:

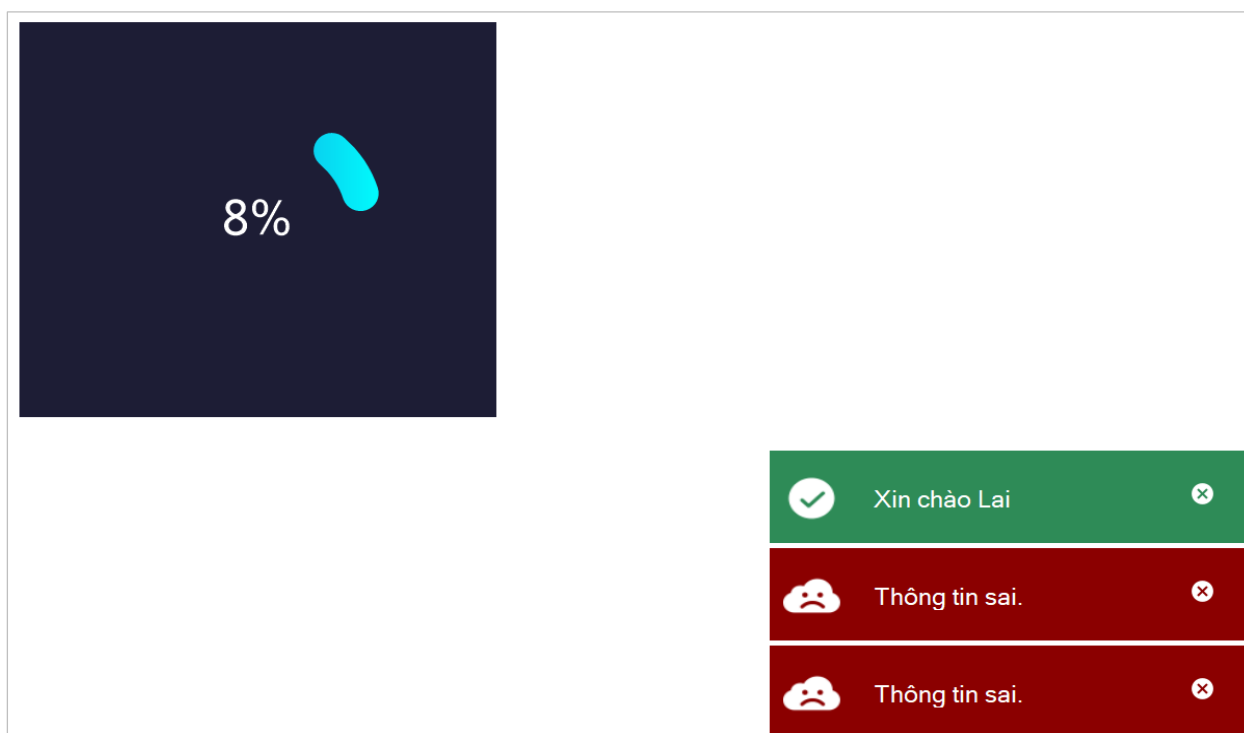
- admin/admin: để vào giao diện chủ cửa hàng
- dtminh/dtminh: để vào giao diện nhân viên.



Hình 3.3 Màn hình đăng nhập ban đầu



Hình 3.4 Màn hình đăng nhập sai



Hình 3.5 Màn hình thông báo đăng nhập sai

3.3.2. Màn hình chính

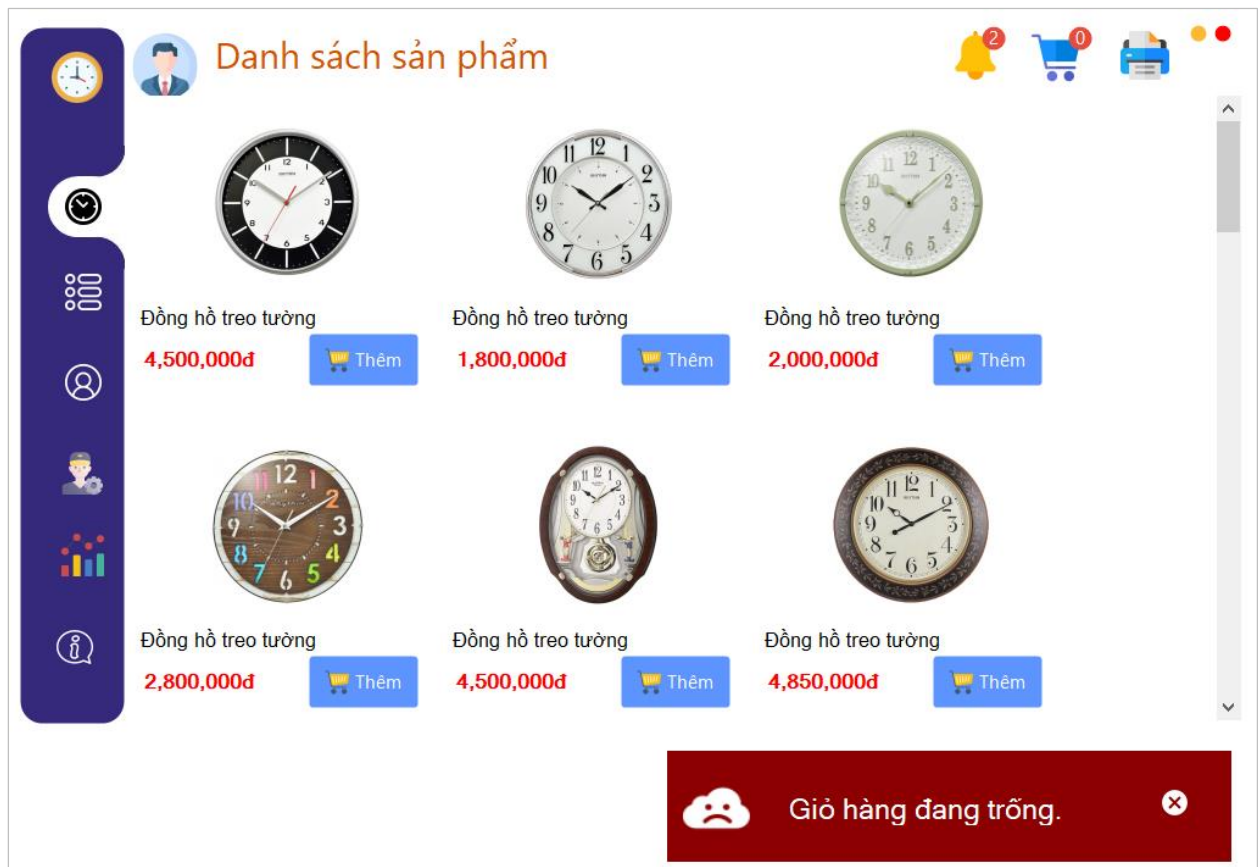


Hình 3.6 Màn hình chính đối với tài khoản admin



Hình 3.7 Màn hình chính đối với tài khoản nhân viên thường

3.3.3. Giao diện khi nhấn nút in



Hình 3.8 Giao diện in hóa đơn khi chưa thêm sản phẩm vào giỏ hàng

3.3.4. Màn hình điền thông tin để xuất hóa đơn


The screenshot shows a form titled 'NHẬP THÔNG TIN HÓA ĐƠN' (Enter Invoice Information). The form includes the following fields and controls:

- Số hóa đơn:** 000000000005
- Thời gian:** 08:27:22 17/09/2020
- Đơn vị:** Đặng Quốc Lai
- Chọn chi nhánh:** 123 Nguyễn Hữu Cảnh, G
- Số điện thoại:** 0896804466 (with a dropdown menu showing other numbers: 0813788390, 0896228105, 0896804466, 0898863448)
- Tên khách hàng:** (with a dropdown menu showing the same numbers as the phone number field)
- Địa chỉ:** (with a dropdown menu showing the same numbers as the phone number field)
- Ngày mua:** 08:27:11 17/09/2020 (with a refresh icon and a 'Reset' button)
- ĐT người nhận:** 0896804466
- Tên người nhận:** Huỳnh Thị Thúy Kiều
- Địa chỉ nhận:** Bình Định
- Ngày giao:** 08:27:11 17/09/2020

There is a green 'In' button and a red 'Reset' button in the center of the form.

Hình 3.9 Màn hình điền thông tin xuất hóa đơn

3.3.5. Màn hình in hóa đơn



Công ty TNHH TM DV Quốc Lai
Địa chỉ : 606 Quốc lộ 13, P. HBP, Q. Thủ Đức, TP.HCM
Mã số thuế : **0315706254**
Chi nhánh : 123 Nguyễn Hữu Cánh, Q. Bình Thạnh, TP. Hồ
Người lập: Đặng Quốc Lai

BIÊN NHẬN

Số hóa đơn: **000000000005**
Ngày lập: 08:27:11 17/09/2020

KHÁCH HÀNG

Họ tên **Huỳnh Thị Thúy Kiều**
Địa chỉ **Bình Định**
Điện thoại **0896804466**
Ngày mua 08:27:11 17/09/2020

Thông tin giao hàng

Họ tên **Huỳnh Thị Thúy Kiều**
Địa chỉ **Bình Định**
Điện thoại **0896804466**
Ngày giao 08:27:11 17/09/2020

STT	Pd_name	Số lượng	Đơn giá	Thành tiền
1	Đồng hồ treo tường CMG527NR03	1	2,000,000	2,000,000
2	Đồng hồ treo tường CMG544NR02	1	1,800,000	1,800,000

Tổng cộng: 3,800,000

TP. Hồ Chí Minh, ngày 17 tháng 09 năm 2020.

Khách hàng

Người lập hóa đơn

Huỳnh Thị Thúy Kiều

Đặng Quốc Lai

Current Page No.: 1

Total Page No.: 1

Zoom Factor: 100%

Hình 3.10 Màn hình in hóa đơn

3.3.6. Màn hình kiểm tra danh sách hóa đơn



Mã hóa đơn	Họ tên KH	Ngày mua	Ngày giao	Tổng tiền
000000000000	Huỳnh Thị Thúy Kiều	09/09/2020 5:36:1...	09/09/2020 5:36:11 PM	6,300,000
000000000001	Nguyễn Thị Tiên	15/09/2020 3:06:4...	15/09/2020 3:06:49 A...	12,600,000
000000000002	Huỳnh Thị Thúy Kiều	16/09/2020 9:34:3...	16/09/2020 9:34:00 A...	6,300,000
000000000003	Huỳnh Thị Thúy Kiều	16/09/2020 10:32:...	16/09/2020 10:32:00 ...	6,300,000
000000000004	Huỳnh Thị Thúy Kiều	16/09/2020 10:38:...	16/09/2020 10:38:00 ...	4,500,000

Hình 3.11 Màn hình kiểm tra danh sách hóa đơn

3.3.7. Màn hình xem danh sách khách hàng



Mã KH	Họ tên	Điện thoại	Địa chỉ
0000000	Huỳnh Thị Thúy Kiều	0896804466	Bình Định
0000001	Nguyễn Thị Tiên	0813788390	TP. Hồ Chí Minh
0000002	La Hoàng Long	0523769577	TP. Nha Trang
0000003	La Quán Trung	0307906979	Bình Định
0000004	Lưu Huyền Đức	896713483	TP. Hồ Chí Minh
0000005	Tào Mạnh Đức	0559224778	TP. Hồ Chí Minh
0000006	Ngô Nhật Huy	0596447968	TP. Nha Trang
0000007	Nguyễn Trọng Khánh	0898863448	Bình Định
0000008	Hà Tiểu Mai	0896228105	Phú Yên
0000009	Văn Tiến Luận	0898863448	Bình Định

Hình 3.12 Màn hình xem danh sách khách hàng

3.3.8. Màn hình quản lý nhân viên



Mã NV	Họ tên	Chi nhánh	Số ngày công	Lương CB	Tính lương
admin	Đặng Quốc Lai	000	10	100,000	1,000,000
dtminh	Đặng Thành ...	002	10	120,000	1,200,000
dttram	Đặng Thùy Tr...	001	23	200,000	4,600,000
hqhuy	Huỳnh Quan...	001	8	120,000	960,000
lhkhang	Lê Hoàng Kh...	000	10	150,000	1,500,000
ngmy	Ngô Hoàng ...	000	5	170,000	850,000
nqtrung	Nguyễn Qua...	002	10	120,000	1,200,000
ntdung	Nguyễn Thành...	001	4	120,000	480,000

Hình 3.13 Màn hình quản lý nhân viên

3.3.9. Màn hình xem thông tin của nhân viên



XIN CHÀO ĐẶNG QUỐC LAI

Tên đăng nhập: admin Số điện thoại: 0772470988

Họ: Đặng Quốc Email: danglai.mail@

Tên: Lai Giới tính: ☒ Nam

Đã làm việc: 10 ngày

[Đổi mật khẩu](#) [Lưu thông tin](#) [Clear](#)

Hình 3.14 Màn hình xem, sửa thông tin của nhân viên

Hình 3.15 Màn hình yêu cầu mật khẩu khi muốn thay đổi thông tin nhân viên

Hình 3.16 Màn hình đổi mật khẩu

3.4. Code các tính năng trong ứng dụng

3.4.1. Code cho button “Đăng nhập” ở form Đăng nhập

```
private void btnLogin_Click(object sender, EventArgs e){
    if (txtNotEmpty(txtUsername, "Tên đăng nhập") &
        txtNotEmpty(txtPassword, "Mật khẩu")) {
        // cả 2 ô nhập vào đều có dữ liệu
        string username = txtUsername.Text.Trim(),
            password = txtPassword.Text.Trim();
        Staffs_DTO s = Staffs.BLL.GetStaffByLogin(username, password);
        if (s == null) { // đăng nhập sai
            frmAlert frmAlert = new frmAlert();
            frmAlert.showAlert("Thông tin sai.", frmAlert.enmType.Error);
            txtUsername.Text = "";
            txtPassword.Text = "";
            txtUsername.Focus();
        }
        else{
```



```

frmAlert frmAlert = new frmAlert();
frmAlert.showAlert(
    "Xin chào " + s.St_lastName,
    frmAlert.enmType.Success);
this.Hide();
frmLoading fLoading = new frmLoading();
fLoading.ShowDialog();
frmMain f = new frmMain(s);
f.ShowDialog();
this.Close();
}
}
}

```

3.4.2. Code cho button xem danh sách sản phẩm

```

private void btnSanPham_Click(object sender, EventArgs e){
    lblTitle.Text = "Danh sách sản phẩm";
    flpSanPham.Show();
    pnMain.Hide();
}

```

3.4.3. Phương thức addToPnMain

```

private void addToPnMain(UserControl uc, string title){
    pnMain.Controls.Clear();
    pnMain.Controls.Add(uc);
    uc.Dock = DockStyle.Fill;

    lblTitle.Text = title;
    flpSanPham.Hide();
    pnMain.Show();
}

```

3.4.4. Code cho button xem danh sách hóa đơn

```
private void btnHoaDon_Click(object sender, EventArgs e){
    addToPnMain(localVariable.ucDanhSachHoaDon, "Danh sách hóa đơn");
}
```

3.4.5. Code cho button xem danh sách khách hàng

```
private void btnKhachHang_Click(object sender, EventArgs e){
    if (localVariable.ucDanhSachKhachHang == null){
        List<Customers_DTO> lstKhachHang = Customer_BLL.LoadCustomers();
        localVariable.ucDanhSachKhachHang = new
            UC_DanhSachKhachHang(lstKhachHang);
    }
    addToPnMain(localVariable.ucDanhSachKhachHang,
        "Danh sách khách hàng");
}
```

3.4.6. Code cho button Lưu thông tin

```
private void btnLuu_Click(object sender, EventArgs e){
    frmAlert f = new frmAlert();
    if (txtCurrentPassword.Text == staff.St_password){
        if (!isUpdateAll){ // chỉ cập nhật mật khẩu
            // gán lại mật khẩu trong đối tượng staff
            if (txtNewPassword.Text != txtNewPassword2.Text)
                f.showAlert("Mật khẩu không trùng nhau.", frmAlert.enmType.Error);
            else
                staff.St_password = txtNewPassword.Text;
        }
        frmAlert frm = new frmAlert();
        if (Staffs_BLL.updateStaff(staff)){
            frm.showAlert("Cập nhật thành công.", frmAlert.enmType.Success);
            this.Close();
        }
    }
}
```

```

else
    frm.showAlert("Cập nhật thất bại.", frmAlert.enmType.Warning);
}
else{
    f.showAlert("Mật khẩu sai.", frmAlert.enmType.Error);
    txtCurrentPassword.Text = txtNewPassword.Text = txtNewPassword2.Text = "";
    txtCurrentPassword.Focus();
}
}

```

3.4.7. Code cho button In hóa đơn

```

private void btnIn_Click(object sender, EventArgs e){
    string od_id = lblSoHoaDon.Text, od_dateDelivery = lblTime.Text, od_status =
"0", od_address = txtDiaChi2.Text, od_payment = "Tiền mặt", ct_id = "0000000",
    ag_id = cboChiNhanh.SelectedValue.ToString();
    int cartCount = Temp.cart.Count;
    // lọc dữ liệu từ bộ nhớ tạm Temp.cart để đẩy vào database
    for(int i = 0; i < cartCount; i++) {
        string price = Temp.cart[i].Od_price.ToString();
        price = price.Substring(0,price.IndexOf(".")+1);
        Order_detail_BLL.addOrder(
            od_id, od_dateDelivery, od_status, od_address, od_payment, st_id, ct_id,
            ag_id, Temp.cart[i].Pd_id, Temp.cart[i].Od_quantity.ToString(), price);
    }
    frmRptCart f = new frmRptCart(cboChiNhanh.Text, txtNgayMua.Text,
    lblSoHoaDon.Text, lblTenNhanVien.Text, txtTenKH1.Text, txtDiaChi1.Text,
    txtDT1.Text, txtTenKH2.Text, txtDiaChi2.Text, txtDT2.Text, txtNgayNhan.Text,
    Temp.cart);
    f.ShowDialog();
}

```

3.4.8. Code cho tính năng thêm mới nhân viên

```
private void btnThem_Click(object sender, EventArgs e){
    if (txtSt_id.Text==""||txtSt_firstName.Text == "" || txtSt_lastName.Text == ""
        || txtSt_phone.Text == "" || txtSt_email.Text == ""){
        frmAlert frm = new frmAlert();
        frm.showAlert("Thông tin không hợp lệ.", frmAlert.enmType.Error);
        return;
    }
    Staffs_DTO staff = new Staffs_DTO();
    staff.St_id = txtSt_id.Text;
    staff.St_firstName = txtSt_firstName.Text;
    staff.St_lastName = txtSt_lastName.Text;
    staff.St_phone = txtSt_phone.Text;
    staff.St_email = txtSt_email.Text;
    staff.St_gender = btnGioiTinh.Checked;
    if (Staffs_BLL.addStaff(staff)) {
        new frmAlert().showAlert("Thêm thành công.", frmAlert.enmType.Success);
        localVariable.lstStaff.Add(staff);
        Close();
    }
    else
        new frmAlert().showAlert("Thêm thất bại.", frmAlert.enmType.Error);
}
```

3.4.9. Code lắng nghe sự kiện nhập tài khoản khi thêm nhân viên

```
private void txtSt_id_TextChanged(object sender, EventArgs e){
    string st_id = txtSt_id.Text;
    if(st_id == ""){
        err.SetError(txtSt_id, "Tài khoản không được để trống");
        btnThem.Enabled = false;
    }
    else {
```

```

for(int i = 0; i < lstCount; i++) {
    if (localVariable.lstStaff[i].St_id == st_id) {
        err.SetError(txtSt_id, "Tài khoản đã tồn tại.");
        btnThem.Enabled = false;
        return;
    }
    else{
        err.SetError(txtSt_id, "");
        btnThem.Enabled = true;
    }
}
}
}
}

```

3.4.10. Code lắng nghe sự kiện nhập số điện thoại ở form nhập khách hàng

```

private void txtDT1_TextChanged(object sender, EventArgs e){
    txtDT2.Text = txtDT1.Text;
    if (txtDT1.Text.Length == 10){
        string text = txtDT1.Text;
        foreach (Customers_DTO c in lstCustomer)
            if (c.Ct_phone == text)
            {
                txtTenKH1.Text = c.Ct_firstName + " " + c.Ct_lastName;
                txtDiaChi1.Text = c.Ct_address;
                txtNgayMua.Text = lblTime.Text;
                return;
            }
    }
}
}

```

KẾT LUẬN

Kết quả đạt được

Xây dựng được chức năng và cho các giao diện:

- Đăng nhập.
- Xem danh sách sản phẩm còn trong cửa hàng.
- Xuất thông tin hóa đơn, In hóa đơn.
- Nhân viên xem và sửa được thông tin của mình (như đổi mật khẩu, email, số điện thoại, xem được số ngày công mà mình đi làm).
- Chủ cửa hàng có thể quản lý được giờ làm của nhân viên, kiểm tra nhân viên làm việc như thế nào, thống kê doanh thu và sản phẩm tồn của từng chi nhánh.

Hạn chế

- Chưa phát triển được toàn bộ tính năng cho sản phẩm.
- Chưa phát triển theo hướng kết nối với internet để trao đổi thông tin giữa các chi nhánh với nhau.
- Giao diện được canh chỉnh responsive cho nhiều loại màn hình, với màn hình có độ phân giải 1920 x 1080 giao diện bị nhỏ so với màn hình, tuy nhiên lại khá lớn với màn hình máy tính để bàn bình thường.
- Chủ cửa hàng chưa thể thêm sản phẩm cho từng chi nhánh.

Hướng phát triển

- Phát triển thành một phần mềm có thể quản lý cho nhiều loại sản phẩm khác nhau chứ không chỉ riêng đồng hồ.
- Phát triển tính năng giao tiếp giữa các chi nhánh trên web.
- Cải thiện lại giao diện để tương tác với nhiều loại màn hình hơn.

TÀI LIỆU THAM KHẢO

- [1] Calling parent form functions from a user control,
<https://www.codeproject.com/Articles/25479/Calling-parent-form-functions-from-a-user-control>, ngày tham khảo 07/09/2019.
- [2] ExecuteNonQuery: Connection property has not been initialize, <https://www.c-sharpcorner.com/forums/executenonquery-connection-property-has-not-been-initialize>, ngày tham khảo 07/09/2019.
- [3] Giang Hào Côn (2020), *Slide bài giảng môn Chuyên đề .NET*, Khoa CNTT, Trường ĐH. Nguyễn Tất Thành.
- [4] Pass Data from User Control To Windows Form,
<https://www.codeproject.com/Questions/657086/Pass-Data-from-User-Control-To-Windows-Form>, ngày tham khảo 17/09/2019.
- [5] Select, Insert, Update, Delete Using Stored Procedure in SQL Server 2008,
<https://www.c-sharpcorner.com/UploadFile/rohatash/select-insert-update-delete-using-stored-procedure-in-sql/>, ngày tham khảo 07/09/2019.