

TRƯỜNG ĐẠI HỌC KIÊN GIANG
KHOA THÔNG TIN & TRUYỀN THÔNG



BÁO CÁO
Niên luận cơ sở

Đề tài:

**XÂY DỰNG PHẦN MỀM
QUẢN LÝ KHO THUỶ SẢN**

Sinh viên: Nguyễn Nhật Đăng

MSSV: 21072006005

Lớp: Công nghệ thông tin - B021TT1

GVHD: Huỳnh Minh Trí

Tháng 10 năm 2023

LỜI CẢM ƠN

Mỗi dự án luôn tạo cho ta lại những kỉ niệm những khó khăn, nhưng dù thế ta vẫn có được sự đồng hành đó là sự giúp đỡ của thầy cô, cũng như bạn bè cùng nhau vượt qua những khó khăn. Với lòng trân trọng biết ơn sự giúp đỡ, em xin gửi đến thầy Huỳnh Minh Trí Trường Đại học Kiên Giang, thầy đã vận dụng kiến thức và tâm huyết của mình để truyền đạt vốn kiến thức quý báu cho chúng em trong suốt thời gian học tập tại trường và đặc biệt, trong học kỳ này.

Ngoài ra sự hỗ trợ của khoa "CÔNG NGHỆ THÔNG TIN" đã mang đến cho em một môn học mang lại sự hiểu biết cũng như là môn nền tảng để học tập về sau. Bên cạnh đó, trong quá trình làm bài do kiến thức của em còn hạn chế và còn nhiều điều mới mẻ chưa nắm được hết trong thời gian.

Do vậy, không tránh khỏi những thiếu sót là điều không thể tránh khỏi được, vì thế em rất mong nhận được những ý kiến đóng góp quý báu của quý thầy cô và các bạn học cùng lớp để kiến thức của em để từ đó bài tập sẽ được tốt hơn. Lời cuối em xin gửi đến thầy cô và các bạn lời cảm ơn chân thành và sâu sắc.

Kiên Giang, ngày.....tháng.....năm 2023

Sinh viên thực hiện

NHẬN XÉT CỦA NGƯỜI HƯỚNG DẪN

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

Tổng điểm:

....., ngày.... tháng.... năm.....

NGƯỜI HƯỚNG DẪN

(ký tên và ghi họ tên)

MỤC LỤC

CHƯƠNG 1: GIỚI THIỆU CHUNG	7
1.1 Giới thiệu tổng quan	7
1.2 Giới thiệu đề tài	8
1.2.1 Giới thiệu	8
1.2.2 Chức năng.....	9
1.2.3 Phân tích xử lý	10
1.3 Giới thiệu về cơ sở lý thuyết cho đề tài.....	11
1.3.1 cơ sở lý thuyết	11
1.3.2 Các giải pháp công nghệ.....	11
1.3.3 công cụ thực hiện.....	13
1.3.4 Database.....	14
CHƯƠNG 2: THIẾT KẾ CÁC MÔ HÌNH HỆ THỐNG.....	16
2.1. Mô hình quan niệm.....	16
2.2. Mô hình luận lý.....	16
2.3. Mô hình Vật lý.....	17
2.4. Mô hình liên kết các bảng trên máy tính	19
2.5. Mô hình xử lý (UML).....	20
CHƯƠNG 3: CÁC MỤC XỬ LÝ (DEMO).....	21
3.1 Đăng nhập hệ thống.....	21
3.2 Cập nhật: thêm, sửa, xóa	22
3.3 Tìm kiếm, hiển thị, xử lý thông tin theo yêu cầu đề tài.....	24
3.4 Thống kê, tổng hợp số liệu theo yêu cầu đề tài	25
3.5 Xử lý nhập xuất, xuất hàng.....	26
3.6 Xử lý in phiếu nhập xuất	27
CHƯƠNG 4: KẾT LUẬN VÀ KIẾN NGHỊ	30
4.1 Kết quả đạt được.....	30
4.2 Hướng phát triển.....	30
4.1 Kiến nghị	31

DANH MỤC HÌNH

Hình 1 mô hình quan niệm	16
Hình 2 Mô hình liên kết các bản	19
Hình 3 Mô hình sử lý.....	20
Hình 4 Giao diện đăng nhập.....	21
Hình 5 Giao diện xem xuất hàng	22
Hình 6 Giao diện sửa phiếu xuất	22
Hình 7 Giao diện tìm kiếm	24
Hình 8 Giao diện thống kê hàng tồn kho.....	25
Hình 9 Giao diện phiếu nhập xuất	27

DANH MỤC BẢNG

Bảng 1 Mô hình vật lý kho	17
Bảng 2 Mô hình vật lý thủy sản.....	17
Bảng 3 Mô hình vật lý ton kho	17
Bảng 4 Mô hình vật lý nhà cung cấp	17
Bảng 5 Mô hình vật lý khách hàng.....	17
Bảng 6 Mô hình vật lý đơn nhập	18
Bảng 7 Mô hình vật lý đơn xuất	18
Bảng 8 Mô hình vật lý chi tiết nhập	18
Bảng 9 Mô hình vật lý chi tiết xuất	18

CHƯƠNG 1: GIỚI THIỆU CHUNG

1.1 Giới thiệu tổng quan

Ngành thủy sản Việt Nam đang phát triển mạnh mẽ trong những năm gần đây. Theo thống kê của Bộ Nông nghiệp và Phát triển Nông thôn, tổng sản lượng thủy sản năm 2022 đạt 8,4 triệu tấn, tăng 3,1% so với năm 2021. Sự phát triển của ngành thủy sản kéo theo sự gia tăng nhu cầu về các dịch vụ hậu cần, trong đó có dịch vụ quản lý kho.

Hiện nay, phần lớn các doanh nghiệp thủy sản vẫn sử dụng các phương pháp quản lý kho truyền thống, như sử dụng sổ sách, giấy tờ hoặc các phần mềm quản lý đơn giản. Các phương pháp này mang lại nhiều hạn chế, như:

- Khó khăn trong việc theo dõi, kiểm soát hàng hóa.
- Dễ xảy ra sai sót, thất thoát.
- Tốn thời gian và công sức.

Với những lý do trên, việc xây dựng phần mềm quản lý kho thủy sản là cần thiết để đáp ứng nhu cầu quản lý kho một cách hiệu quả của các doanh nghiệp thủy sản.

Cụ thể, lợi ích của việc sử dụng phần mềm quản lý kho thủy sản bao gồm:

- Nâng cao hiệu quả quản lý
 - Giảm thiểu sai sót, thất thoát.
 - Tiết kiệm thời gian và công sức.
 - Tăng cường tính minh bạch, chính xác.
- Nâng cao khả năng cạnh tranh:
 - Tăng cường hiệu quả hoạt động của kho
 - Đáp ứng nhu cầu của khách hàng
 - Tạo lợi thế cạnh tranh

Mục tiêu của đề tài xây dựng phần mềm quản lý kho thủy sản là xây dựng một phần mềm đáp ứng các yêu cầu sau:

- Hỗ trợ quản lý toàn diện các hoạt động của kho thủy sản, bao gồm: nhập kho, xuất kho, tồn kho, kiểm kê, báo cáo.
- Đảm bảo tính chính xác, an toàn và bảo mật dữ liệu.
- Tăng cường hiệu quả hoạt động của kho thủy sản.

Phạm vi đề tài xây dựng phần mềm quản lý kho thủy sản được xác định như sau:

- Đối tượng: Phần mềm được xây dựng dành cho các doanh nghiệp thủy sản, bao gồm các doanh nghiệp sản xuất, chế biến, kinh doanh thủy sản.
- Tài nguyên: Phần mềm được xây dựng trên nền tảng lập trình ứng dụng windows, sử dụng các công nghệ lập trình như .Net c#, SQL.
- Chức năng: Phần mềm hỗ trợ quản lý toàn diện các hoạt động của kho thủy sản, bao gồm: nhập kho, xuất kho, tồn kho, kiểm kê, báo cáo.
- Dữ liệu: Dữ liệu được lưu trữ trên hệ thống máy chủ an toàn.

Phạm vi đề tài xây dựng phần mềm quản lý kho thủy sản được xác định như trên là phù hợp với mục tiêu và yêu cầu của đề tài. Phạm vi này sẽ giúp phần mềm đáp ứng được nhu cầu quản lý kho của các doanh nghiệp thủy sản, đồng thời đảm bảo tính hiệu quả và khả thi của đề tài.

Khả năng ứng dụng triển khai đề tài xây dựng phần mềm quản lý kho thủy sản. Phần mềm có thể được ứng dụng trong các doanh nghiệp thủy sản, bao gồm các doanh nghiệp sản xuất, chế biến, kinh doanh thủy sản. Phần mềm sẽ giúp các doanh nghiệp này nâng cao hiệu quả hoạt động của kho thông qua các giải pháp sau:

- Giảm thiểu sai sót: Phần mềm giúp doanh nghiệp thủy sản giảm thiểu sai sót trong quá trình nhập kho, xuất kho, kiểm kê.
- Tiết kiệm thời gian và công sức: Phần mềm giúp doanh nghiệp thủy sản tiết kiệm thời gian và công sức trong việc quản lý kho.
- Tăng cường tính minh bạch, chính xác: Phần mềm giúp doanh nghiệp thủy sản tăng cường tính minh bạch, chính xác trong việc quản lý kho.

1.2 Giới thiệu đề tài

1.2.1 Giới thiệu

Đề tài xây dựng phần mềm quản lý kho thủy sản sẽ xây dựng một phần mềm giúp quản lý toàn diện các hoạt động của kho thủy sản, bao gồm:

- Nhập kho: Theo dõi quá trình nhập kho, bao gồm việc kiểm tra chất lượng, số lượng, chủng loại hàng hóa, cập nhật thông tin hàng hóa vào hệ thống.
- Xuất kho: Theo dõi quá trình xuất kho, bao gồm việc kiểm tra thông tin đơn hàng, số lượng, chủng loại hàng hóa, cập nhật thông tin hàng hóa đã xuất kho.

- Tồn kho: Theo dõi tình trạng tồn kho, bao gồm số lượng, chủng loại, vị trí, hạn sử dụng của hàng hóa.
- Kiểm kê: Thực hiện kiểm kê định kỳ để đảm bảo tính chính xác của số lượng hàng hóa trong kho.
- Báo cáo: Cung cấp các báo cáo tổng hợp về hoạt động của kho, giúp nhà quản lý đưa ra các quyết định kinh doanh phù hợp.

1.2.2 Chức năng

Các chức năng chính của phần mềm

- Các chức năng chính của phần mềm quản lý kho thủy sản bao gồm:
 - Quản lý hàng hóa: Phần mềm sẽ quản lý thông tin về hàng hóa, bao gồm mã hàng, tên hàng, chủng loại, đơn vị tính, giá cả, hạn sử dụng,...
 - Quản lý nhập kho: Phần mềm sẽ hỗ trợ nhập kho hàng hóa, bao gồm việc kiểm tra chất lượng, số lượng, chủng loại hàng hóa, cập nhật thông tin hàng hóa vào hệ thống.
 - Quản lý xuất kho: Phần mềm sẽ hỗ trợ xuất kho hàng hóa, bao gồm việc kiểm tra thông tin đơn hàng, số lượng, chủng loại hàng hóa, cập nhật thông tin hàng hóa đã xuất kho.
 - Quản lý tồn kho: Phần mềm sẽ theo dõi tình trạng tồn kho của hàng hóa, bao gồm số lượng, chủng loại, vị trí, hạn sử dụng của hàng hóa.
 - Quản lý kiểm kê: Phần mềm sẽ hỗ trợ kiểm kê định kỳ để đảm bảo tính chính xác của số lượng hàng hóa trong kho.
 - Quản lý báo cáo: Phần mềm sẽ cung cấp các báo cáo tổng hợp về hoạt động của kho, giúp nhà quản lý đưa ra các quyết định kinh doanh phù hợp.

Các yêu cầu về tính năng của phần mềm

- Phần mềm quản lý kho thủy sản cần đáp ứng các yêu cầu về tính năng sau:
 - Chính xác: Phần mềm cần đảm bảo tính chính xác của dữ liệu, tránh sai sót trong quá trình nhập kho, xuất kho, kiểm kê.
 - An toàn: Phần mềm cần được trang bị các tính năng bảo mật để ngăn chặn việc truy cập trái phép hoặc sửa đổi dữ liệu.
 - Dễ sử dụng: Phần mềm cần có giao diện thân thiện, dễ sử dụng cho người dùng.

- Mở rộng: Phần mềm cần có khả năng mở rộng để đáp ứng nhu cầu của doanh nghiệp trong tương lai.

Đề tài xây dựng phần mềm quản lý kho thủy sản sẽ là một công cụ hữu ích giúp các doanh nghiệp thủy sản nâng cao hiệu quả hoạt động của kho.

1.2.3 Phân tích xử lý

Các hoạt động xử lý của phần mềm quản lý kho thủy sản gồm:

- Quản lý kho.
 - Quản lý thông kê hàng tồn trong kho.
 - Thêm sửa xoá thông tin của kho.
- Quản lý thủy sản.
 - Quản lý các thông tin của thủy sản bao gồm tên của thủy sản và giá xuất ra của thủy sản.
 - Thêm hàng hoá mới, cập nhật các thông tin của các loại thủy sản, xoá thủy sản.
- Quản lý khách hàng và nhà cung cấp.
 - Xem các thông tin của khách hàng và các nhà cung cấp (khách hàng là các công ty tổ chức cá nhân nhập hàng hoá từ kho nhà cung cấp là các công ty tổ chức cá nhân cung cấp thủy sản cho kho). Các thông tin bao gồm tên địa chỉ sdt.
 - Thêm khách hàng và nhà cung cấp mới, cập nhật thông tin của khách hàng và nhà cung cấp và xoá khách hàng và nhà cung cấp khỏi hệ thống.
- Quản lý nhập kho.
 - Xem thông tin của phiếu nhập kho (các phiếu nhập kho gồm thông tin nhà cung cấp nhập vào kho nào danh sách các hàng hoá được nhập).
 - Thêm phiếu nhập kho, chỉnh sửa thông tin phiếu nhập(chỉ những phiếu nhập chưa được nhập vào kho mới có thể chỉnh sửa), xoá phiếu nhập kho.
 - Thực hiện nhập kho(khi nhập kho các hàng trong phiếu nhập sẽ được chuyển vào hàng tồn kho và không thể chỉ sửa được nữa).
- Quản lý xuất kho.

- Xem thông tin của phiếu xuất kho (các phiếu xuất kho gồm thông tin khách hàng nhập vào kho nào danh sách các hàng hoá được nhập).
- Thêm phiếu xuất kho, chỉnh sửa thông tin phiếu xuất (chỉ những phiếu xuất chưa được xuất mới có thể chỉnh sửa), xoá phiếu xuất kho.
- Thực hiện xuất kho (khi xuất kiểm tra hàng tồn trong kho có đủ thay không nếu đủ mới có thể xuất khi xuất không thể sửa các thông tin được nữa).

1.3 Giới thiệu về cơ sở lý thuyết cho đề tài

1.3.1 cơ sở lý thuyết

Các cơ sở lý thuyết để xây dựng phần mềm:

- Cơ sở dữ liệu.
- Phân tích thiết kế hệ thống thông tin.
- Lập trình ứng dụng windows.

1.3.2 Các giải pháp công nghệ

1.3.2.1 .NET Framework

.NET Framework là một nền tảng lập trình và cũng là một nền tảng thực thi ứng dụng chủ yếu trên hệ điều hành Microsoft Windows được phát triển bởi Microsoft từ năm 2002, kết thúc phát triển vào năm 2022 ở phiên bản 4.8.1. Các chương trình được viết trên nền .NET Framework có thể chạy trên Windows, macOS, và Linux thông qua Mono.

.NET Framework bao gồm tập các thư viện lập trình lớn, và những thư viện này hỗ trợ việc xây dựng các chương trình phần mềm như lập trình giao diện; truy cập, kết nối cơ sở dữ liệu; ứng dụng web; các giải thuật, cấu trúc dữ liệu; giao tiếp mạng... CLR cùng với bộ thư viện này là 2 thành phần chính của .NET Framework.

.NET Framework là một nền tảng phát triển phần mềm mạnh mẽ và linh hoạt có thể được sử dụng để xây dựng nhiều loại ứng dụng khác nhau. Nó là một lựa chọn tuyệt vời cho các nhà phát triển muốn xây dựng các ứng dụng chất lượng cao và hiệu quả.

Tải xuống .NET Framework tại: <https://dotnet.microsoft.com/en-us/download>

1.3.2.2 ngôn ngữ lập trình C#

C# là một ngôn ngữ lập trình hướng đối tượng được phát triển bởi Microsoft. Nó là một phần của nền tảng .NET và có thể được sử dụng để xây dựng nhiều loại ứng dụng

khác nhau, bao gồm ứng dụng web, ứng dụng máy tính để bàn, ứng dụng di động và trò chơi.

C# là một ngôn ngữ mạnh mẽ và linh hoạt, có nhiều tính năng ưu việt, bao gồm:

- Tính năng đa nền tảng: Ứng dụng C# có thể chạy trên Windows, macOS và Linux.
- Mã nguồn mở: C# là một ngôn ngữ mã nguồn mở, có nghĩa là nó miễn phí để sử dụng và sửa đổi.
- Cộng đồng lớn: C# có một cộng đồng nhà phát triển lớn và tích cực có thể cung cấp hỗ trợ và trợ giúp.
- Hệ sinh thái phong phú: Có nhiều thư viện và công cụ có sẵn cho các nhà phát triển C#.
- Nền tảng trưởng thành: C# là một ngôn ngữ trưởng thành đã tồn tại hơn 20 năm

1.3.2.3 Entity Framework

Entity Framework là một công cụ ánh xạ đối tượng-quan hệ (ORM) mã nguồn mở được phát triển bởi Microsoft. Nó cho phép các nhà phát triển truy cập và thao tác dữ liệu trong cơ sở dữ liệu quan hệ bằng cách sử dụng các đối tượng lập trình hướng đối tượng.

Entity Framework hoạt động bằng cách tạo một lớp đối tượng cho mỗi bảng trong cơ sở dữ liệu. Các đối tượng này đại diện cho các hàng trong bảng và các thuộc tính của chúng đại diện cho các cột trong bảng.

Entity Framework có thể được sử dụng với nhiều loại cơ sở dữ liệu quan hệ, bao gồm SQL Server, Oracle, và MySQL.

Lợi ích của việc sử dụng Entity Framework

Entity Framework mang lại nhiều lợi ích cho các nhà phát triển, bao gồm:

- Tăng năng suất: Entity Framework giúp các nhà phát triển tiết kiệm thời gian và công sức bằng cách tự động tạo các câu lệnh SQL.
- Tăng độ bảo mật: Entity Framework giúp bảo vệ dữ liệu bằng cách cung cấp các tính năng bảo mật tích hợp.
- Tăng khả năng mở rộng: Entity Framework giúp các ứng dụng dễ dàng mở rộng bằng cách cung cấp các tính năng như phân tích và truy vấn dữ liệu.

1.3.3 công cụ thực hiện

1.3.3.1 Visual Studio 2019

Visual Studio 2019 là một môi trường phát triển tích hợp (IDE) được phát triển bởi Microsoft. Nó được sử dụng để phát triển phần mềm cho Windows, macOS và Linux. Visual Studio 2019 là phiên bản mới nhất của Visual Studio, được phát hành vào năm 2019.

Visual Studio 2019 bao gồm nhiều tính năng mới và cải tiến, bao gồm:

- Cải thiện hiệu suất: Visual Studio 2019 được cải thiện hiệu suất đáng kể, giúp các nhà phát triển xây dựng ứng dụng nhanh hơn.
- Tích hợp với Azure: Visual Studio 2019 được tích hợp chặt chẽ với Azure, giúp các nhà phát triển dễ dàng triển khai ứng dụng lên đám mây.
- Tính năng mới cho các ngôn ngữ lập trình: Visual Studio 2019 bổ sung nhiều tính năng mới cho các ngôn ngữ lập trình phổ biến, bao gồm C#, Visual Basic, và Python.

Visual Studio 2019 là một IDE mạnh mẽ và linh hoạt, có thể được sử dụng để phát triển nhiều loại ứng dụng khác nhau. Nó là một lựa chọn tuyệt vời cho các nhà phát triển phần mềm muốn xây dựng các ứng dụng chất lượng cao và hiệu quả.

Các tính năng chính của Visual Studio 2019

- Công cụ lập trình: Visual Studio 2019 cung cấp các công cụ lập trình mạnh mẽ cho các ngôn ngữ lập trình phổ biến, bao gồm C#, Visual Basic, và Python.
- Trình duyệt mã: Visual Studio 2019 có trình duyệt mã tích hợp giúp các nhà phát triển dễ dàng tìm kiếm và điều hướng mã.
- Trình gỡ lỗi: Visual Studio 2019 có trình gỡ lỗi mạnh mẽ giúp các nhà phát triển tìm và giải quyết lỗi trong mã.
- Trình xây dựng: Visual Studio 2019 có trình xây dựng tích hợp giúp các nhà phát triển xây dựng và triển khai ứng dụng.
- Tích hợp với các dịch vụ của Microsoft: Visual Studio 2019 được tích hợp chặt chẽ với các dịch vụ của Microsoft, chẳng hạn như Azure và GitHub.

Cách tải xuống và cài đặt Visual Studio 2019

Tải xuống Visual Studio 2019 tại:

https://my.visualstudio.com/Downloads?q=visual%20studio%202019&wt.mc_id=o~msft~vscom~older-downloads

1.3.4 Database

SQL Server là một hệ thống quản trị cơ sở dữ liệu quan hệ (RDBMS) được phát triển bởi Microsoft. Nó là một lựa chọn phổ biến để lưu trữ và quản lý dữ liệu cho nhiều ứng dụng khác nhau, bao gồm ứng dụng web, ứng dụng máy tính để bàn và ứng dụng di động.

Các thành phần của SQL Server

- Bảng: Một bảng là một tập hợp các dữ liệu có liên quan được lưu trữ trong cơ sở dữ liệu. Mỗi bảng bao gồm một số hàng và cột.
- Hàng: Một hàng là một bản ghi duy nhất trong bảng. Mỗi hàng đại diện cho một thực thể, chẳng hạn như một khách hàng, một sản phẩm hoặc một đơn đặt hàng.
- Cột: Một cột là một trường dữ liệu trong bảng. Mỗi cột chứa một loại dữ liệu cụ thể, chẳng hạn như số, chuỗi hoặc ngày tháng.
- Mối quan hệ: Các bảng trong SQL Server có thể được liên kết với nhau bằng các mối quan hệ. Mối quan hệ là một liên kết giữa các dữ liệu trong hai hoặc nhiều bảng.
- Trình điều khiển máy khách: Trình điều khiển máy khách là một chương trình phần mềm cho phép các ứng dụng giao tiếp với SQL Server.
- Trình quản lý cơ sở dữ liệu: Trình quản lý cơ sở dữ liệu là một công cụ phần mềm cho phép người dùng quản lý SQL Server, chẳng hạn như tạo, xóa và sửa đổi các bảng và mối quan hệ.

Các câu lệnh SQL

SQL là một ngôn ngữ truy vấn cơ sở dữ liệu được sử dụng để truy vấn, thêm, cập nhật và xóa dữ liệu từ SQL Server.

SQL Server mang lại nhiều lợi ích cho các nhà phát triển, bao gồm:

- Tính khả dụng: SQL Server có thể được lưu trữ trên máy chủ hoặc đám mây, giúp chúng có sẵn cho người dùng từ bất kỳ đâu.
- Tính bảo mật: SQL Server có thể được bảo vệ bằng mã hóa và các biện pháp bảo mật khác.

- Tính mở rộng: SQL Server có thể được mở rộng để đáp ứng nhu cầu của doanh nghiệp.
- Tính hiệu suất: SQL Server là một hệ thống hiệu quả có thể xử lý lượng lớn dữ liệu và truy vấn phức tạp.
- Tính dễ sử dụng: SQL Server có thể được sử dụng bởi các nhà phát triển và người dùng ở mọi cấp độ kinh nghiệm.

Các ứng dụng của SQL Server

- Các ứng dụng web: SQL Server được sử dụng để lưu trữ dữ liệu cho các ứng dụng web, chẳng hạn như các trang web thương mại điện tử và mạng xã hội.
- Các ứng dụng máy tính để bàn: SQL Server được sử dụng để lưu trữ dữ liệu cho các ứng dụng máy tính để bàn, chẳng hạn như các ứng dụng kế toán và CRM.
- Các ứng dụng di động: SQL Server được sử dụng để lưu trữ dữ liệu cho các ứng dụng di động, chẳng hạn như các ứng dụng ngân hàng và giao thông

Tải xuống sql server tại: <https://www.microsoft.com/en-us/sql-server/sql-server-downloads>

Don_nhap(id, ngay_nhap, trang_thai, id_kho, id_ncp).

Don_xuat(id, ngay_nhap, trang_thai, id_kho, id_kh).

Ct_nhap(id_dn, id_ts, so_luong, gia_nhap).

Ct_xuat(id_dn, id_ts, so_luong, gia_xuat).

2.3. Mô hình Vật lý

Kho(ID, Ten).

Tên thuộc tính	Kiểu dữ liệu	K thước	Ràng buộc toàn vẹn
ID (K)	Int		
Ten	Varchar	Len()=30	

Bảng 1 Mô hình vật lý kho

Thuy_san(ID, ten, gia_ban).

Tên thuộc tính	Kiểu dữ liệu	K thước	Ràng buộc toàn vẹn
ID (K)	Int		
Ten	Varchar	Len()=30	
gia_ban	int		

Bảng 2 Mô hình vật lý thủy sản

Ton_kho(Id_kho, id_ts, so_luong).

Tên thuộc tính	Kiểu dữ liệu	K thước	Ràng buộc toàn vẹn
Id_kho (K)	Int		Lookup(kho)
id_ts(K)	Int	Len()=30	Lookup(Thuy_san)
so_luong	int		

Bảng 3 Mô hình vật lý ton kho

Nha_cung_cap(id, ten, sdt, dia_chi).

Tên thuộc tính	Kiểu dữ liệu	K thước	Ràng buộc toàn vẹn
ID (K)	Int		
Ten	Varchar	Len()=30	
sdt	char	Len()=12	
dia_chi	Varchar	Len()=255	

Bảng 4 Mô hình vật lý nhà cung cấp

Khac_hang(id, ten, sdt, dia_chi).

Tên thuộc tính	Kiểu dữ liệu	K thước	Ràng buộc toàn vẹn
ID (K)	Int		
Ten	Varchar	Len()=30	
sdt	char	Len()=12	
dia_chi	Varchar	Len()=255	

Bảng 5 Mô hình vật lý khách hàng

Don_nhap(id, ngay_nhap, trang_thai, id_kho, id_ncp).

Tên thuộc tính	Kiểu dữ liệu	K thước	Ràng buộc toàn vẹn
ID (K)	Int		
ngay_nhap	date		
trang_thai	bool		
id_kho	int		Lookup(kho)
id_ncp	int		Lookup(nha cung cap)

Bảng 6 Mô hình vật lý đơn nhập

Don_xuat(id, ngay_nhap, trang_thai, id_kho, id_kh).

Tên thuộc tính	Kiểu dữ liệu	K thước	Ràng buộc toàn vẹn
ID (K)	Int		
ngay_nhap	date		
trang_thai	bool		
id_kho	int		Lookup(kho)
id_kh	int		Lookup(khac hang)

Bảng 7 Mô hình vật lý đơn xuất

Ct_nhap(id_dn, id_ts, so_luong, gia_nhap).

Tên thuộc tính	Kiểu dữ liệu	K thước	Ràng buộc toàn vẹn
id_dn (K)	Int		Lookup(don nhap)
id_ts(K)	date		Lookup(thuy san)
so_luong	int		
gia_nhap	int		

Bảng 8 Mô hình vật lý chi tiết nhập

Ct_xuat(id_dn, id_ts, so_luong, gia_xuat).

Tên thuộc tính	Kiểu dữ liệu	K thước	Ràng buộc toàn vẹn
id_dx (K)	Int		Lookup(don xuất)
id_ts(K)	date		Lookup(thuy san)
so_luong	int		
gia_nhap	int		

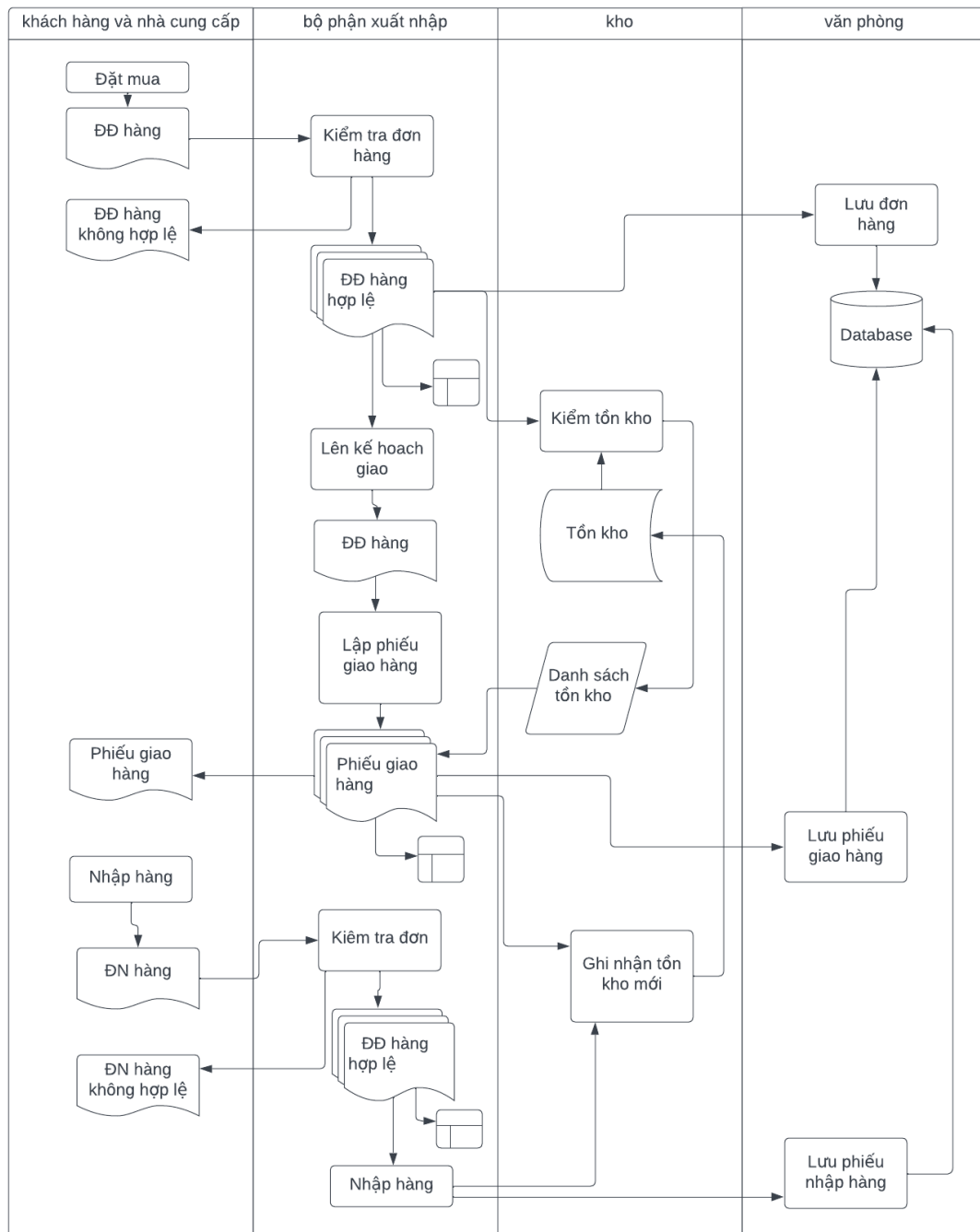
Bảng 9 Mô hình vật lý chi tiết xuất

2.4. Mô hình liên kết các bảng trên máy tính



Hình 2 Mô hình liên kết các bản

2.5. Mô hình xử lý (UML)

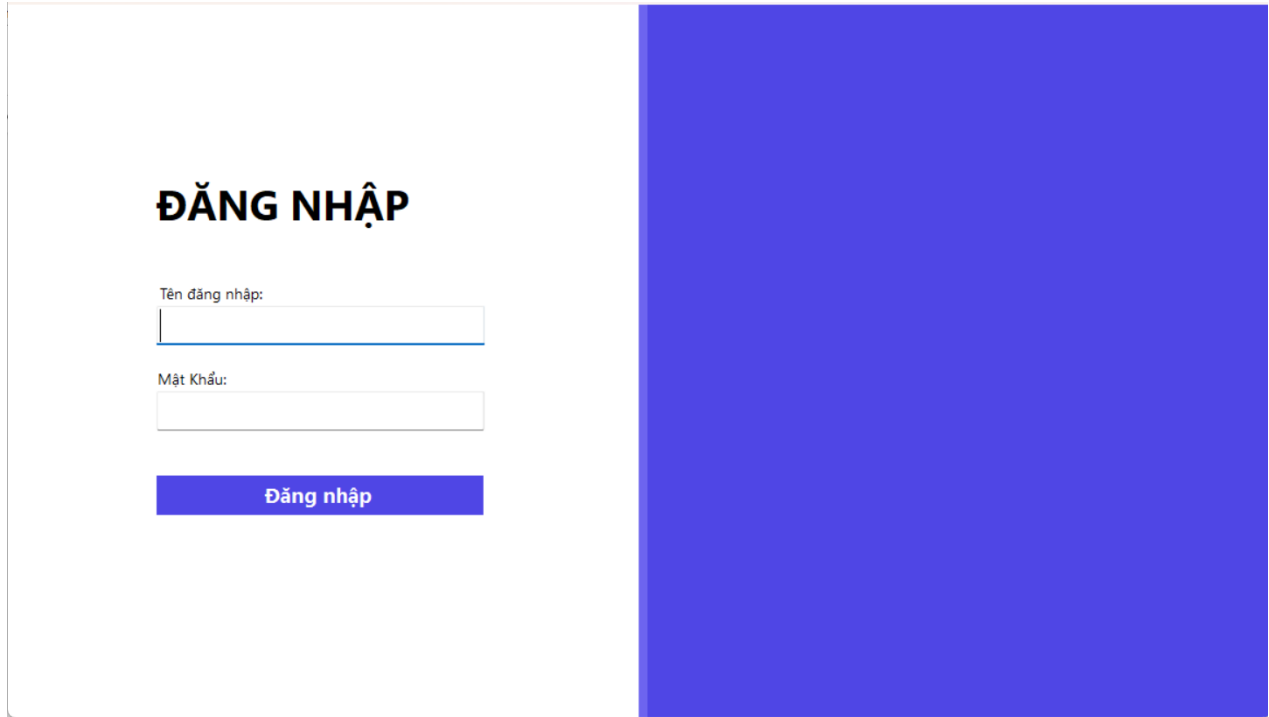


Hình 3 Mô hình xử lý

CHƯƠNG 3: CÁC MỤC XỬ LÝ (DEMO)

3.1 Đăng nhập hệ thống

Giao diện đăng nhập vào ứng dụng.

The image shows a login form on a white background. At the top, the text 'ĐĂNG NHẬP' is displayed in a bold, black, sans-serif font. Below this, there are two input fields. The first is labeled 'Tên đăng nhập:' and the second is labeled 'Mật Khẩu:'. Both labels are in a small, black, sans-serif font. The input fields are white with a thin blue border. Below the input fields is a blue button with the text 'Đăng nhập' in white, bold, sans-serif font. The entire form is positioned on the left side of a larger white area, with a solid blue rectangle to its right.

Hình 4 Giao diện đăng nhập

Code chính:

```
private void Login()
{
    string username = tUsername.Text;
    string password = Password.Text;

    var query =
        fRoot.getContext().accounts.SingleOrDefault(s => s.username ==
            username);

    if(query != null)
    {
        if(password == query.password.TrimEnd(' '))
        {
            fRoot.Logined(query.Id);
            Password.Text = "";
            return;
        }
    }

    Lerr.Text = "Tài khoản hoặc mật khẩu sai!";
}
```

3.2 Cập nhật: thêm, sửa, xóa

Giao diện thêm sửa xóa đơn hàng

The screenshot shows a web application interface for managing export orders. On the left is a blue sidebar with a 'LOGO' and navigation links: 'Trang chủ', 'Sản phẩm', 'Xuất hàng' (highlighted), 'Nhập hàng', 'Khách hàng', and 'Nhà cung cấp'. At the bottom of the sidebar is a 'Đăng xuất' button. The main area is titled 'Xuất hàng' and contains a search bar with a 'Tìm kiếm' button. Below the search bar is a table with two rows of data:

Item	Tổng số lượng(kg)	Tổng giá
an Số phiếu: 1001	20	1600000
teo Số phiếu: 1000	40	4800000

Below the table is a large empty box. To the right of the table is a summary panel titled 'Phiếu xuất' with the number '1001' and a print icon. It contains the following information:

- Khách hàng: an
- Kho: kho 01
- Tổng số lượng: 20
- Tổng giá: 1600000
- Ngày xuất: none
- Hàng xuất: Cá rô
- Giá: 80000
- Số lượng: 20

At the bottom right of the main area are two buttons: 'xóa' (red) and 'chỉnh sửa' (white).

Hình 5 Giao diện xem xuất hàng

Giao diện sửa

The screenshot shows the 'Sửa phiếu xuất' (Edit Export Order) interface. It features a blue sidebar on the left with the same navigation links as Figure 5. The main area is titled 'Sửa phiếu xuất' and contains a search bar. Below the search bar is a table with two rows of data:

Item	Tổng số lượng(kg)	Tổng giá
an Số phiếu: 1001	20	1600000
teo Số phiếu: 1000	40	4800000

Below the table is a large empty box. To the right of the table is a summary panel titled 'Sửa phiếu xuất' with the number '1001'. It contains the following information:

- Khách hàng: an
- Kho: kho 01
- Hàng xuất: Cá rô
- Số lượng: 20
- Giá: 80000
- Tên sản phẩm:

At the bottom right of the main area is a 'Lưu' (Save) button.

Hình 6 Giao diện sửa phiếu xuất

Code chính phần thêm:

```

private int Save() {
    if (String.IsNullOrEmpty(tKH.Text) || cKho.SelectedItem == null)
        return -1;

    var kh = root.getContext().KhachHangs.SingleOrDefault(e => e.ten_kh == tKH.Text);
    var kho = root.getContext().Khoes.SingleOrDefault(e => e.ten_kho ==
cKho.SelectedItem.ToString());

    if (kh == null || kho == null)
        return -1;

    var newPx = new models.PhieuXuat {
        Id_kho = kho.Id_kho,
        id_kh = kh.Id_kh,
        ngay_xuat = DateTime.Now,
        da_xuat = false
    };

    foreach (var item in fListHang.Controls) {
        LICreatePhieuXuat li = (LICreatePhieuXuat)item;
        if (li.IdTS != -1 && li.soluong > 0)
            newPx.TTPhieuXuats.Add(li.GetTTPhieuXuat());
    }

    root.getContext().PhieuXuats.Add(newPx);
    root.getContext().SaveChanges();
    return newPx.Id_px;
}

```

Code chính phần sửa:

```

private void Save() {
    var kh = root.getContext().KhachHangs.SingleOrDefault(e => e.ten_kh == tKH.Text);
    var kho = root.getContext().Khoes.SingleOrDefault(e => e.ten_kho ==
cKho.SelectedItem.ToString());

    if (kh == null || kho == null)
        return;
    var px = root.getContext().PhieuXuats.Find(Id);

    px.id_kh = kh.Id_kh;
    px.Id_kho = kho.Id_kho;

    foreach (var item in fListHang.Controls) {
        LIEditPhieuXuat li = (LIEditPhieuXuat)item;
        li.Save();
    }

    root.getContext().SaveChanges();
    root.GetExportController().Reload();
    root.MinicontrolClose();
}

```

Code chính phần xóa:

```

private void bDelete_Click(object sender, EventArgs ev) {
    DialogResult dialogResult = MessageBox.Show("Bạn có muốn xóa?", "delete",
MessageBoxButtons.YesNo);
    if (dialogResult != DialogResult.Yes) return;

    var px = root.getContext().PhieuXuats.Find(Id);
    var ttpx = root.getContext().TTPhieuXuats.Where(e => e.Id_px == Id);
}

```

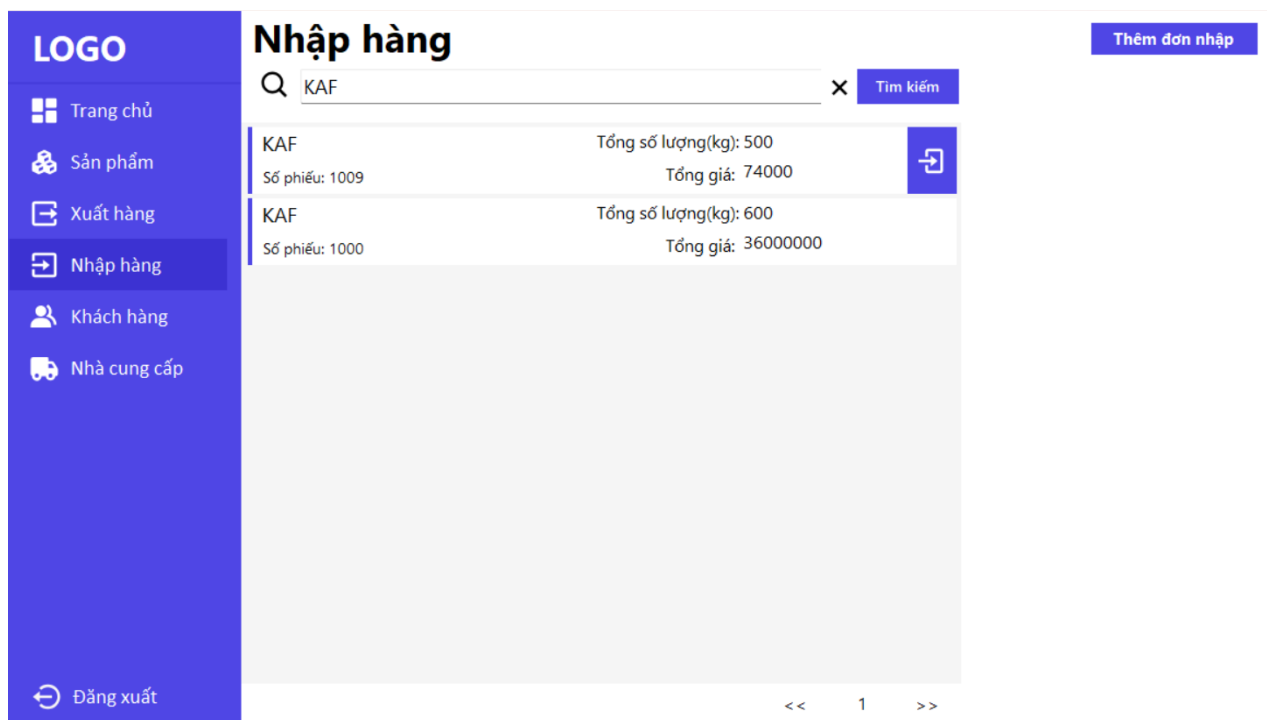
```

root.getContext().TTPhieuXuats.RemoveRange(ttpx);
root.getContext().PhieuXuats.Remove(px);
root.getContext().SaveChanges();
root.GetExportController().Reload();
}

```

3.3 Tìm kiếm, hiển thị, xử lý thông tin theo yêu cầu đề tài

Giao diện tìm kiếm đơn nhập



Hình 7 Giao diện tìm kiếm

Code chính phần tìm kiếm:

```

private void Search(object sender, EventArgs e) {
    if (String.IsNullOrEmpty(tSearch.Text)) {
        isSeachMode = false;
        bRemoveSearch.Visible = false;
        tSearch.Text = "";

        SetPageNumber();
        SetList(1);

        RenderList();
        return;
    }

    isSeachMode = true;
    bRemoveSearch.Visible = true;

    SetPageNumber();
    SetList(1);

    RenderList();
}

```


3.4 Thống kê, tổng hợp số liệu theo yêu cầu đề tài

Giao diện thống kê hàng tồn kho:

LOGO	Quản lý kho hàng		Thêm kho
Trang chủ	All	Q	Tổng lượng hàng tồn: 980
Sản phẩm	Cá lóc	Tồn kho(kg): 500	
Xuất hàng	kho 01		
Nhập hàng	Cá rô	Tồn kho(kg): 480	
Khách hàng	kho 01		
Nhà cung cấp			
Đăng xuất			

Hình 8 Giao diện thống kê hàng tồn kho

Code chính phần thống kê:

```
private void ReRender() {
    int controlsNumber = pUnList.Controls.Count;
    int count = 0;

    foreach (var item in List) {
        if (count < controlsNumber) {
            var control = (LITonKho)pUnList.Controls[count];
            control.NameTS = item.ThuySan.ten;
            control.At = item.Kho.ten_kho;
            control.SL = item.so_luong;
            control.ReRender();
            count++;
            continue;
        }
        LITonKho temp = new LITonKho {
            NameTS = item.ThuySan.ten,
            At = item.Kho.ten_kho,
            SL = item.so_luong
        };
        temp.ReRender();
        pUnList.Controls.Add(temp);
    }

    while (count < controlsNumber) {
        pUnList.Controls.RemoveAt(pUnList.Controls.Count-1);
        count++;
    }
    ResizeList();
}
```

3.5 Xử lý nhập xuất, xuất hàng

Code chính xử lý nhập:

```
public bool NhapKho(int id) {
    var pn = root.getContext().PhieuNhaps.Find(id);
    if (pn == null)
        return false;
    if (pn.da_nhap)
        return false;
    pn.ngay_nhap = DateTime.Now;
    pn.da_nhap = true;

    var ttpn = pn.TTPhieuNhaps.ToList();

    foreach(var item in ttpn) {
        var tonkho = root.getContext().TonKhoes.SingleOrDefault(e => e.id_ts ==
item.Id_ts && e.Id_kho == pn.id_kho);
        if(tonkho == null) {
            var newTK = new models.TonKho {
                id_ts = item.Id_ts,
                Id_kho = pn.id_kho,
                so_luong = item.so_luong
            };
            root.getContext().TonKhoes.Add(newTK);

            continue;
        }
        tonkho.so_luong += item.so_luong;
    }
    try {
        root.getContext().SaveChanges();
        return true;
    } catch (Exception ex) {
        root.Rollback();
        return false;
    }
}
```

Code chính xử lý xuất:

```
public bool xuấtKho(int id) {
    var px = root.getContext().PhieuXuats.Find(id);
    if (px == null)
        return false;
    if (px.da_xuat)
        return false;

    var ttpx = px.TTPhieuXuats.ToList();

    foreach (var item in ttpx) {
        var tonkho = root.getContext().TonKhoes.SingleOrDefault(e => e.id_ts ==
item.Id_ts && e.Id_kho == px.Id_kho);
        if (tonkho == null) {
            root.Rollback();
            throw new Exception("khong du hang");
        }
        if(tonkho.so_luong - item.so_luong < 0) {
            root.Rollback();
            throw new Exception("khong du hang");
        }
        if(tonkho.so_luong - item.so_luong == 0) {
```

```

        root.getContext().TonKhoes.Remove(tonkho);
        continue;
    }
    tonkho.so_luong -= item.so_luong;
}

px.ngay_xuat = DateTime.Now;
px.da_xuat = true;

try {
    root.getContext().SaveChanges();
} catch (Exception ex) {
    root.Rollback();
}
return true;
}

```

3.6 Xử lý in phiếu nhập xuất

Phiếu nhập, xuất:

PHIẾU XUẤT KHO

Số: 1000

Tên khách hàng: teo
 Ngày xuất: 10/20/2023 9:18:54 PM
 Xuất tại kho: kho 01

STT	Tên thủy sản	Số lượng	Đơn giá	Thành tiền
1	Cá lóc	20	80000	1600000
2	Cá rô	20	160000	3200000
Tổng giá				4800000

Nhà cung cấp

Bên nhận hàng

Hình 9 Giao diện phiếu nhập xuất

Code chính:

```

private void printDocument1_PrintPage(object sender,
System.Drawing.Printing.PrintPageEventArgs e) {
    var pn = root.getContext().PhieuXuats.Find(Id);

    Font font14 = new Font("Segoe UI", 14, FontStyle.Regular);

```

```

        e.Graphics.DrawString("PHIẾU XUẤT KHO", new Font("Segoe UI", 26, FontStyle.Bold),
Brushes.Black, new Point(265, 80));
        e.Graphics.DrawString("Số: " + pn.Id_px.ToString(), new Font("Segoe UI", 14,
FontStyle.Regular), Brushes.Black, new Point(377, 125));

        e.Graphics.DrawString("Tên khách hàng: " + pn.KhachHang.ten_kh, new Font("Segoe
UI", 16, FontStyle.Regular), Brushes.Black, new Point(50, 175));
        e.Graphics.DrawString("Ngày xuất: " + (pn.da_xuat ? pn.ngay_xuat.ToString() : "chưa
xuất"), new Font("Segoe UI", 16, FontStyle.Regular), Brushes.Black, new Point(50, 208));
        e.Graphics.DrawString("Xuất tại kho: " + pn.Kho.ten_kho, new Font("Segoe UI", 16,
FontStyle.Regular), Brushes.Black, new Point(50, 241));

        e.Graphics.DrawLine(new Pen(Brushes.Black), 50, 275, 800, 275);
        e.Graphics.DrawLine(new Pen(Brushes.Black), 50, 275, 50, 305);
        e.Graphics.DrawLine(new Pen(Brushes.Black), 100, 275, 100, 305);
        e.Graphics.DrawLine(new Pen(Brushes.Black), 800, 275, 800, 305);
        e.Graphics.DrawLine(new Pen(Brushes.Black), 680, 275, 680, 305);
        e.Graphics.DrawLine(new Pen(Brushes.Black), 560, 275, 560, 305);
        e.Graphics.DrawLine(new Pen(Brushes.Black), 440, 275, 440, 305);

        e.Graphics.DrawString("STT", font14, Brushes.Black, new Point(55, 277));
        e.Graphics.DrawString("Tên thủy sản", font14, Brushes.Black, new Point(105, 277));
        e.Graphics.DrawString("Số lượng", font14, Brushes.Black, new Point(445, 277));
        e.Graphics.DrawString("Đơn giá", font14, Brushes.Black, new Point(565, 277));
        e.Graphics.DrawString("Thành tiền", font14, Brushes.Black, new Point(685, 277));

        int sum = 0;
        int count = 0;
        foreach (var item in pn.TTPhieuXuats) {
            e.Graphics.DrawLine(new Pen(Brushes.Black), 50, 305 + (30 * count), 50, 335 +
(30 * count));
            e.Graphics.DrawLine(new Pen(Brushes.Black), 100, 305 + (30 * count), 100, 335
+ (30 * count));
            e.Graphics.DrawLine(new Pen(Brushes.Black), 800, 305 + (30 * count), 800, 335
+ (30 * count));
            e.Graphics.DrawLine(new Pen(Brushes.Black), 680, 305 + (30 * count), 680, 335
+ (30 * count));
            e.Graphics.DrawLine(new Pen(Brushes.Black), 560, 305 + (30 * count), 560, 335
+ (30 * count));
            e.Graphics.DrawLine(new Pen(Brushes.Black), 440, 305 + (30 * count), 440, 335
+ (30 * count));
            e.Graphics.DrawLine(new Pen(Brushes.Black), 50, 305 + (30 * count), 800, 305
+ (30 * count));

            e.Graphics.DrawString((count + 1).ToString(), font14, Brushes.Black, new
Point(55, 307 + (30 * count)));
            e.Graphics.DrawString(item.ThuySan.ten, font14, Brushes.Black, new Point(105,
307 + (30 * count)));
            e.Graphics.DrawString(item.so_luong.ToString(), font14, Brushes.Black, new
Point(445, 307 + (30 * count)));
            e.Graphics.DrawString(((int)(item.gia_xuat)).ToString(), font14,
Brushes.Black, new Point(565, 307 + (30 * count)));
            e.Graphics.DrawString(((int)(item.so_luong * item.gia_xuat)).ToString(),
font14, Brushes.Black, new Point(685, 307 + (30 * count)));

            sum += (int)(item.so_luong * item.gia_xuat);
            count++;
        }
        e.Graphics.DrawLine(new Pen(Brushes.Black), 50, 305 + (30 * count), 50, 335 + (30 *
count));
        e.Graphics.DrawLine(new Pen(Brushes.Black), 680, 305 + (30 * count), 680, 335 + (30
* count));

```

```

        e.Graphics.DrawLine(new Pen(Brushes.Black), 50, 305 + (30 * count), 800, 305 + (30
* count));
        e.Graphics.DrawLine(new Pen(Brushes.Black), 50, 335 + (30 * count), 800, 335 + (30
* count));
        e.Graphics.DrawLine(new Pen(Brushes.Black), 800, 305 + (30 * count), 800, 335 + (30
* count));

        e.Graphics.DrawString("Tổng giá", font14, Brushes.Black, new Point(55, 307 + (30 *
count)));
        e.Graphics.DrawString(sum.ToString(), font14, Brushes.Black, new Point(685, 307 +
(30 * count)));

        e.Graphics.DrawString("Nhà cung cấp", new Font("Segoe UI", 16, FontStyle.Regular),
Brushes.Black, new Point(60, 350 + (30 * count)));
        e.Graphics.DrawString("Bên nhận hàng", new Font("Segoe UI", 16, FontStyle.Regular),
Brushes.Black, new Point(620, 350 + (30 * count)));
    }

```

CHƯƠNG 4: KẾT LUẬN VÀ KIẾN NGHỊ

4.1 Kết quả đạt được

Trong quá trình tìm hiểu và khảo sát trên, chúng em đã đạt được một số kết quả sau:

Phân tích được quy trình hoạt động và các chức năng của hệ thống

Xây dựng được các chức năng cơ bản của một phần mềm quản lý kho thuỷ sản.

Các chức năng chính của phần mềm

- Quản lý hàng hoá.
- Thống kê hàng tồn kho.
- Quản lý nhập xuất kho.
- Thêm sửa xoá hàng hoá.
- Thêm sửa xoá phiếu nhập xuất.
- Quản lý các kho chứa hàng.
- Quản lý khách hàng xem thêm sửa xoá khách hàng.
- Quản lý nhà cung cấp em thêm sửa xoá nhà cung cấp.
- In phiếu nhập xuất hàng hoá.
- Thực thi nhập xuất kho.
- Đăng nhập vào hệ thống.
- Tiềm kiểm hàng hoá hoá đơn khách hàng và nhà cung cấp.

Ưu điểm:

- Giao diện đẹp.
- Dễ sử dụng.
- Có các chức năng cơ bản của phần mềm quản lý kho.

Hạn chế:

- Phần mềm còn đơn giản.
- Chức năng còn hạn chế.

4.2 Hướng phát triển

- Tiếp tục nguyên cứu, xây dựng để ứng dụng ngày càng hoàn thiện và có thể áp dụng vào thực tế.
- Phát triển thêm nhiều chức năng cho phần mềm.
- Cải thiện hiệu năng của ứng dụng.

- Tích hợp thực động hoá cho ứng dụng.
- Thêm trang thông kê cụ thể hơn cho ứng dụng.
- Cập nhật các tính năng sao lưu dữ liệu.
- Báo cáo tự động.
- Quản lý hàng hóa theo lô.
- Quản lý hàng hóa theo hạn sử dụng.
- Quản lý hàng hóa theo vị trí.
- Quản lý hàng hóa theo nhà cung cấp.
- Quản lý hàng hóa theo khách hàng.

4.1 Kiến nghị

Do vốn hiểu biết của chúng em còn nhiều hạn chế nên chương trình còn rất nhiều hạn chế và không thể ứng dụng vào thực tế. Chính vì vậy, chúng em mong thầy hướng dẫn cho đề tài của chúng em thêm.

TÀI LIỆU THAM KHẢO

Link website

https://my.visualstudio.com/Downloads?q=visual%20studio%202019&wt.mc_id=o~msft~vscom~older-downloads

<https://www.microsoft.com/en-us/sql-server/sql-server-downloads>

<https://dotnet.microsoft.com/en-us/download>

Kiên Giang, ngày 6 tháng 11 năm 2023

Sinh viên thực hiện

Nguyễn Nhật Đăng