BỘ CÔNG THƯƠNG

**TRƯỜNG ĐẠI HỌC KINH TẾ - KỸ THUẬT CÔNG NGHIỆP**

**KHOA CÔNG NGHỆ THÔNG TIN**

A blue and white logo

Description automatically generated

**BÁO CÁO ĐỒ ÁN 1**

**TÊN ĐỀ TÀI :**

**XÂY DỰNG PHẦN MỀM QUẢN LÝ BÁN HÀNG CHO CỬA HÀNG BÁN GIÀY DÉP**

**Giảng viên hướng dẫn: ThS. Phạm Thị Hồng Nhung**

**Nhóm SV thực hiện :**

|  |  |  |
| --- | --- | --- |
| **Mã sinh viên** | **Họ và Tên** | **Vị trí** |
| 21103101027 | Trần Đình Hoàng Phúc | Nhóm trưởng |
| 21103101032 | Nguyễn Đình Minh | Thành Viên |
| 21103101037 | Đào Trọng Tiệp | Thành Viên |
| 21103100986 | Lê Thái Tuấn | Thành Viên |

**HÀ NỘI, NĂM 2024**

BỘ CÔNG THƯƠNG

**TRƯỜNG ĐẠI HỌC KINH TẾ - KỸ THUẬT CÔNG NGHIỆP**

**KHOA CÔNG NGHỆ THÔNG TIN**

A blue and white logo

Description automatically generated

**BÁO CÁO ĐỒ ÁN 1**

**TÊN ĐỀ TÀI :**

**XÂY DỰNG PHẦN MỀM QUẢN LÝ BÁN HÀNG CHO CỬA HÀNG BÁN GIÀY DÉP**

**Giảng viên hướng dẫn: ThS. Phạm Thị Hồng Nhung**

**Nhóm SV thực hiện :**

|  |  |  |
| --- | --- | --- |
| **Mã sinh viên** | **Họ và Tên** | **Vị trí** |
| 21103101027 | Trần Đình Hoàng Phúc | Nhóm trưởng |
| 21103101032 | Nguyễn Đình Minh | Thành Viên |
| 21103101037 | Đào Trọng Tiệp | Thành Viên |
| 21103100986 | Lê Thái Tuấn | Thành Viên |

**HÀ NỘI, NĂM 2024**

# Mục Lục

[Mục Lục 1](#_Toc167787424)

[DANH MỤC HÌNH ẢNH 1](#_Toc167787425)

[LỜI NÓI ĐẦU 1](#_Toc167787426)

[LÝ DO CHỌN ĐỀ TÀI 3](#_Toc167787427)

[NHẬN XÉT CỦA GIẢNG VIÊN HƯỚNG DẪN 4](#_Toc167787428)

[DANH MỤC CÁC TỪ VIẾT TẮT 5](#_Toc167787429)

[CHƯƠNG 1. KHẢO SÁT VÀ MÔ TẢ HỆ THỐNG 6](#_Toc167787430)

[1.1 Khảo sát hiện trạng hệ thống. 6](#_Toc167787431)

[1.1.1 Tên đề tài 6](#_Toc167787432)

[1.1.2 Phương pháp nghiên cứu 6](#_Toc167787433)

[1.2 Địa điểm khảo sát 6](#_Toc167787434)

[1.3 Khảo sát hệ thống 6](#_Toc167787435)

[1.4 Mô tả hoạt động của hệ thống 7](#_Toc167787436)

[1.4.1 Quản lý nhập hàng 7](#_Toc167787437)

[1.4.2 Quản lý bán hàng 8](#_Toc167787438)

[1.4.3 Quản lý thông tin 9](#_Toc167787439)

[1.4.4 Báo cáo thống kê 10](#_Toc167787440)

[1.5 Mục tiêu quản lý 12](#_Toc167787441)

[CHƯƠNG 2. PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG 13](#_Toc167787442)

[2.1 Công cụ sử dụng 13](#_Toc167787443)

[2.2 Xây dựng mô hình phân cấp chức năng 14](#_Toc167787444)

[2.3 Các ký hiệu được sử dụng trong mô hình DFD 18](#_Toc167787445)

[2.3.1 Chức năng( tiến trình) 18](#_Toc167787446)

[2.3.2 Luồng dữ liệu 19](#_Toc167787447)

[2.3.3 Kho dữ liệu 19](#_Toc167787448)

[2.3.4 Tác nhân ngoài 20](#_Toc167787449)

[2.3.5 Tác nhân trong 20](#_Toc167787450)

[2.4 Xây dựng sơ đồ ngữ cảnh 21](#_Toc167787451)

[2.5 Xây dựng sơ đồ luồng dữ liệu mức đỉnh 21](#_Toc167787452)

[2.6 Xây dựng sơ đồ luồng dữ liệu mức dưới đỉnh 22](#_Toc167787453)

[2.7 Phân tích và thiết kế về dữ liệu 25](#_Toc167787454)

[CHƯƠNG 3: XÂY DỰNG PHẦN MỀM CHO BÀI TOÁN QUẢN LÝ 29](#_Toc167787455)

[3.1. Thiết kế CSDL cho dự án 29](#_Toc167787456)

[3.1.1 Thiết kế các bảng quan hệ 29](#_Toc167787457)

[3.1.2 Chuẩn hóa CSDL của dự án Các dạng chuẩn: 32](#_Toc167787458)

[3.1.3 Mô hình quan hệ (Database Diagrams) và Code SQL 33](#_Toc167787459)

[3.2 Xây dựng phần mềm quản lý hệ thống hoàn chỉnh của dự án 36](#_Toc167787460)

[3.2.1 Lựa chọn công cụ: Ngôn ngữ lập trình Winforms C# 36](#_Toc167787461)

[3.2.2 Thiết kế giao diện 37](#_Toc167787462)

[3.2.3 Kết quả kiểm thử chương trình 82](#_Toc167787463)

[CHƯƠNG 4: KẾT LUẬN 82](#_Toc167787464)

[4.1 Kết quả đạt được 82](#_Toc167787465)

[4.2 Hạn chế của đề tài 82](#_Toc167787466)

[4.3 Hướng phát triển 83](#_Toc167787467)

[TÀI LIỆU THAM KHẢO 83](#_Toc167787468)

# DANH MỤC HÌNH ẢNH

[Hình 2.1. Sơ đồ phân cấp chức năng của quản lý shop giày dép 17](#_Toc167787469)

[Hình 2.2. Sơ đồ mức ngữ cảnh 21](#_Toc167787470)

[Hình 2.3. Sơ đồ luồng dữ liệu mức đỉnh 22](#_Toc167787471)

[Hình 2.4. Sơ đồ luồng dữ liệu mức dưới đỉnh của chức năng quản lý nhập hàng. 23](#_Toc167787472)

[Hình 2.5. Sơ đồ luồng dữ liệu mức dưới đỉnh của chức năng quản lý bán hàng. 23](#_Toc167787473)

[Hình 2.6. Sơ đồ luồng dữ liệu mức dưới đỉnh của chức năng quản lý thông tin. 24](#_Toc167787474)

[Hình 2.7. Sơ đồ luồng dữ liệu mức dưới đỉnh của chức năng báo cáo thống kê. 24](#_Toc167787475)

[Hình 3.1 Sơ đồ Database Diagrams của hệ thống 33](#_Toc167787476)

[Hình 3.2 Giao diện Form đăng nhập 39](#_Toc167787477)

[Hình 3.3 Form giao diện chính 41](#_Toc167787478)

[Hình 3.4 Giao diện Form quản lý tài khoản nhân viên 44](#_Toc167787479)

[Hình 3.5 Giao diện Form quản lý nhà cung cấp 48](#_Toc167787480)

[Hình 3.6 Giao diện Form quản lý loại hàng hóa 52](#_Toc167787481)

[Hình 3.7 Giao diện Form quản lý hàng hóa 56](#_Toc167787482)

[Hình 3.8 Giao diện Form quản lý khách hàng 61](#_Toc167787483)

[Hình 3.9 Giao diện Form Phiếu nhập kho 65](#_Toc167787484)

[Hình 3.10 Giao diện Form thêm mới phiếu nhập 69](#_Toc167787485)

[Hình 3.11 Form phiếu xuất kho 73](#_Toc167787486)

[Hình 3.12 Giao diện form thêm mới phiếu xuất 78](#_Toc167787487)

[Hình 3.13 Form Báo cáo tồn kho 80](#_Toc167787488)

[Hình 3.14 Hiển thị màn hình thoát chương trình 81](#_Toc167787489)

# LỜI NÓI ĐẦU

Ngày nay khoa học kỹ thuật phát triển như vũ bão, lao động trí óc dần thay thế cho lao động chân tay bằng những ứng dụng khoa học kỹ thuật. Và góp phần đắc lực trong cuộc cách mạng khoa học này phải kể đến lĩnh vực công nghệ thông tin, công nghệ thông tin được ứng dụng trong nhiều lĩnh vực. Đặc biệt là trong ngành kinh tế, nó đóng góp một phần lớn vào quá trình hội nhập.

Như chúng ta đã biết, việc quản lý kinh doanh của rất nhiều của hàng nói chung và của hàng kinh doanh giày dép nói riêng vẫn tiến hành rất thủ công, việc ghi chép thông qua số sách gặp rất nhiều trở ngại, gây khó khăn cho người quản lý khi muốn xem xét tình trạng các mặt hàng còn hay hết. Từ thực tế như vậy mà nhóm chúng em đã tiến hành xây dựng một phần mềm quản lý hoạt động kinh doanh cho một cửa hàng giày dép.

Hệ thống là một chương trình phần mềm cho phép người sử dụng thực hiện một cách nhanh chóng, chính xác việc nhập dữ liệu, lưu trữ cập nhật thông tin các mặt hàng, thực hiện yêu cầu bảo cáo bán hàng hay báo cáo tài chính của cửa hàng với giao diện làm việc thân thiện tiện dụng đối với người sử dụng.

Mong rằng với phần mềm này chúng em có thể đóng góp được một phần nào đó vào sự phát triển kinh doanh của các cửa hàng giày dép nói riêng và các cửa hàng kinh doanh khác nói chung.

Với sự cố gắng và nổ lực chúng em đã cùng nhau xây dựng và hoàn thành để tài này. Xong vẫn còn nhiều thiếu xót chúng em rất mong nhận được sự đóng góp ý kiến của thầy và các bạn để cho đề tài của chúng em được hoàn thiện hơn.

Chúng em xin chân thành cảm ơn !

**Nhóm sinh viên thực hiện:**

|  |  |
| --- | --- |
| Trần Đình Hoàng Phúc (Nhóm trưởng) | Nguyễn Đình Minh |
| Đào Trọng Tiệp | Lê Thái Tuấn |

|  |  |
| --- | --- |
| **ĐH KINH TẾ KỸ THUẬT CÔNG NGHIỆP**  **KHOA CÔNG NGHỆ THÔNG TIN** | **Đồ án số: 13**  CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT NAM  **Độc lập – Tự do – Hạnh phúc** |

ĐỒ ÁN 1

Nhóm SV thực hiện: 1- Trần Đình Hoàng Phúc - NT 2- Nguyễn Đình Minh

3- Đào Trọng Tiệp 4- Lê Thái Tuấn

Tên đồ án: ***Xây dựng phần mềm quản lý bán hàng cho cửa hàng bán giầy dép***

**Yêu cầu:**

- Mục tiêu của đồ án:

* Củng cố kiến thức môn học phân tích thiết kế hệ thống
* Rèn luyện kỹ năng thiết kế chương trình và kỹ thuật lập trình.
* Rèn luyện kỹ năng thực hiện và trình bày báo cáo.

- Nhiệm vụ của đồ án:

* Khảo sát và phân tích thiết kế hệ thống dựa trên một hệ thống thực tế, gắn liền với bài toán quản lý kinh doanh tại một cửa hàng bán giầy dép. Tìm hiểu các yêu cầu, nhiệm vụ cần thực hiện của hệ thống, các mẫu biểu cần thiết khi xây dựng hệ thống.
* Thiết kế các Form, xây dựng các lớp thực thể, tạo các bảng cơ sở dữ liệu và nhập mỗi bảng tối thiểu10 bản ghi, kết nối cơ sở dữ liệu.
* Cập nhật dữ liệu từ các Form vào cơ sở dữ liệu (thêm, sửa, xóa).
* Thiết kế các Report hiển thị dữ liệu, đáp ứng nhu cầu tìm kiếm, thống kê theo thời gian (ngày, tháng, quý, năm) hoặc theo yêu cầu thực tế của bài toán.

- Kết quả cần đạt được:

* Hiểu và trình bày về hệ thống quản lý kinh doanh tại một cửa hàng bán giầy dép.
* Thiết kế và cài đặt chương trình quản lý kinh doanh tại một cửa hàng bán giầy dép với các chức năng quản lý cơ bản như cập nhật thông tin, tìm kiếm, thống kê, báo cáo...
* Viết báo cáo tổng hợp và trình bày báo cáo

Hà Nội, ngày 05 tháng 03 năm 2024

Giảng viên

Phạm Thị Hồng Nhung

# LÝ DO CHỌN ĐỀ TÀI

Cùng với sự phát triên không ngừng về kỹ thuật máy tính và mạng điện tử, công nghệ thông tin cũng được những công nghệ có đăng cấp cao và lần lượt chinh phục hết đỉnh cao này đến đỉnh cao khác. Mạng Internet là một trong những sản phâm có giá trị hết sức lớn lao và ngày càng trở nên một công cụ không thể thiếu, là nền tảng chính cho sự truyền tải, trao đổi thông tin trên toàn cầu.

Chúng em đã quyết định chọn để tài : “ **Xây dựng phần mềm quản lý bán giày dép** “ với một số lý do sau:

Nhu cầu cải thiện hiệu suất hoạt động Hiện nay, các cửa hàng đang gặp phải nhiều thách thức trong việc quản lý và cung cấp dữ liệu về sản phẩm của mình, Một hệ thống quản lý bán hàng hiệu quả sẽ giúp cải thiện hiệu suất hoạt động của cửa, giúp đảm bảo rằng chủ cửa hàng có thể dễ dàng truy cập và thay đổi mọi thứ về sản phẩm mà của mình.

Tăng cường trải nghiệm người dùng, Phần mềm quản lý bán hàng sẽ manglại một cách quản lý dễ dàng hơn cho . Bằng cách cung cấp các tính năng tiện ích như tra cứu sản phẩm trực tuyến, thay đổi thông tin sản phẩm và thông báo về doanh thu và các sản phẩm đã hết. Chúng tôi sẽ tạo ra một môi trường thuận lợi và thoải mái cho người dùng.

Cải thiện quản lý sản phẩm, phần mềm quản lý cửa hàng sẽ giúp thư viện theo dõi và quản lý sản phẩm một cách hiệu quả hơn, từ việc quản lý số lượng tồn kho, đến việc theo dõi đơn hàng của khách hàng, hệ thống sẽ cung cấp thông tin và chi tiết chính xác, giúp việc quản lý trở nên dễ dàng, thuận tiện và hiệu quả hơn.

# NHẬN XÉT CỦA GIẢNG VIÊN HƯỚNG DẪN

**NỘI DUNG NHẬN XÉT:**

**1. Về nội dung của đồ án:**

**............................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................**

**2. Về hình thức của đồ án:**

**.....................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................**

**.......................................................................................................................................................................**

**3. Những nhận xét khác:**

**.....................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................**

**.......................................................................................................................................................................**

**4. Đánh giá và cho điểm:**

**.....................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................**

**.......................................................................................................................................................................**

# DANH MỤC CÁC TỪ VIẾT TẮT

|  |  |
| --- | --- |
| **Từ** | **Nghĩa của từ** |
|  |  |
| BFD | Mô hình phân rã chức năng |
| DFD | Sơ đồ luồng dữ liệu |
| NCC | Nhà cung cấp |
| TT | Thông tin |
| DS | Danh sách |
| KH | Khách hàng |
| NV | Nhân viên |
| y.c | Yêu cầu |
| TB | Thông báo |
| HSDL | Hồ sơ dữ liệu |
| MH | Mặt hàng |
| CSDL | Cơ sở dữ liệu |
|  |  |
|  |  |
|  |  |

# 

# KHẢO SÁT VÀ MÔ TẢ HỆ THỐNG

## 1.1 Khảo sát hiện trạng hệ thống.

### 1.1.1 Tên đề tài

“Xây dựng phần mềm quản lý bán hàng cho cửa hàng bán giầy dép”

### 1.1.2 Phương pháp nghiên cứu

- Nghiên cứu lý thuyết

- Nghiên cứu thực nghiệm

## 1.2 Địa điểm khảo sát

- Cửa hàng kinh doanh buôn bán giầy dép Tùng Anh đặt tại: Giáp Hải, làng Thành, Tp. Bắc Giang, Bắc Giang, Vietnam chuyên buôn bán giày dép với nhiều mẫu mã. Cửa hàng nhận bán buôn, bán lẻ với chế độ ưu đãi cho những khách hàng mua hàng với số lượng lớn và dịch vụ hàng theo yêu cầu khách hàng.

## 1.3 Khảo sát hệ thống

- Hệ thống: quản lý việc bán hàng cho cửa hàng shop giày dép.

- Cửa hàng nhỏ gọn vài thành viên, tập trung nguồn giày dép chất lượng đủ các kiểu dáng phong cách khác phù hợp với nhu cầu khách hàng một cách tốt nhất, mang đến cho khách hàng những sản phẩm đẹp mắt và chất lượng tốt với giá cả phải chăng nhất.

- Các mặt hàng: sản phẩm các mẫu giày dép được cập nhật liên tục để phù hợp cho các thanh thiếu niên và những phụ nữ chung tuổi.

- Trong quá trình bán hàng thông tin được nhân viên nhập vào hệ thống :

* Phiếu nhập.
* Phiếu xuất.
* Sổ ghi thông tin nhà cung cấp, khách hàng, nhân viên.

- Hình thức bán trực tiếp tại cửa hàng.

**\*Cơ cấu tổ chức**

Chủ cửa hàng

Nhân viên

**\* Quản lý cửa hàng:**

- Quản lý nhân viên.

- Quyết định việc nhập hàng.

- Liên hệ nhà cung cấp.

**\* Nhân viên:**

- Nhân viên nhận hàng từ nhà cung cấp giao đến, kiểm tra hàng xem có đủ số lượng hàng mình đã đặt với nhà cung cấp hay chưa và hàng có bị lỗi hay không.

- Nếu hàng đã đủ như yêu cầu thì nhận hóa đơn hàng từ nhà cung cấp kiểm tra lại hóa đơn rồi báo với chủ cửa hàng để chủ cửa hàng thanh toán lại tiền cho phía nhà cung cấp.

**\*Kho:** Nơi chứa các sản phẩm hàng mà được shop nhập từ nguồn cung cấp và tổng hợp số lượng để bán ra cho khách hàng.

## 1.4 Mô tả hoạt động của hệ thống

### 1.4.1 Quản lý nhập hàng

- Nhân viên có chức vụ nhập dựa vào nhu cầu thị trường và đơn đặt hàng từ khách hàng để lên danh sách các hàng cần nhập rồi lập đơn hàng nhập, trình lên chủ cửa hàng xem xét và duyệt, đơn đặt hàng sau khi được duyệt sẽ được chủ cửa hàng gửi tới nhà cung cấp, nhà cung cấp xác nhận đơn hàng, những mặt hàng có thể đáp ứng được theo đơn đặt hàng. Hai bên sẽ thỏa thuận ngày giao hàng.

- Đến ngày giao hàng nhận hàng từ NCC, nhà cung cấp sẽ giao hàng và kèm theo hóa đơn hàng. Nhân viên sẽ kiểm tra hàng hóa và hóa đơn xem đúng theo đúng yêu cầu mình đặt hay chưa rồi đưa cho chủ cửa hàng để phê duyệt và thanh toán cho phía nhà cung cấp.

****

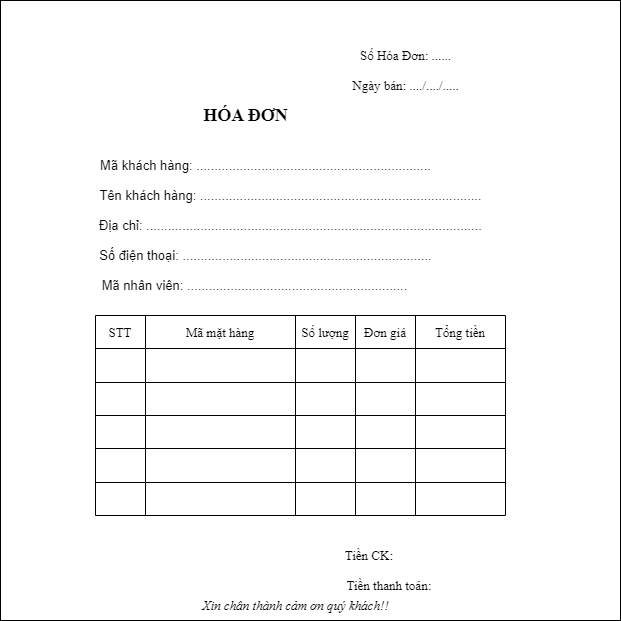
- Trong lúc kiểm tra hàng hóa với bên nhà cung cấp nếu phát hiện những mặt hàng lỗi nhân viên sẽ trả hàng về cho nhà cung cấp để họ giải quyết .

- Hàng sẽ được nhân viên chuyển vào kho, nhân viên sẽ cập nhật số lượng, cập nhật thêm những mặt hàng mới và xóa bỏ những mặt hàng không còn bán nữa.

### 1.4.2 Quản lý bán hàng

-Các mặt hàng sau khi được chuyển đến kho, nhân viên có nhiệm vụ sắp xếp phân loại các mặt hàng lên kệ, đầy đủ màu sắc lẫn size mẫu hàng.

-Khách hàng đến cửa hàng sẽ được nhân viên hướng dẫn, lựa chọn và tư vấn những mẫu mã sản phẩm mà mình thấy vừa size với khách hàng. Khi mà khách hàng đã vừa ý thì khách hàng sẽ mang những đồ mình lựa chọn ra quầy thanh toán. Nhân viên sẽ tiếp nhận, quét mã vạch và lập phiếu xuất để khách hàng thanh toán.



### 1.4.3 Quản lý thông tin

- Cửa hàng sẽ quản lý các thông tin của khách hàng, nhà cung cấp, nhân viên bao gồm: tên, địa chỉ, sđt.

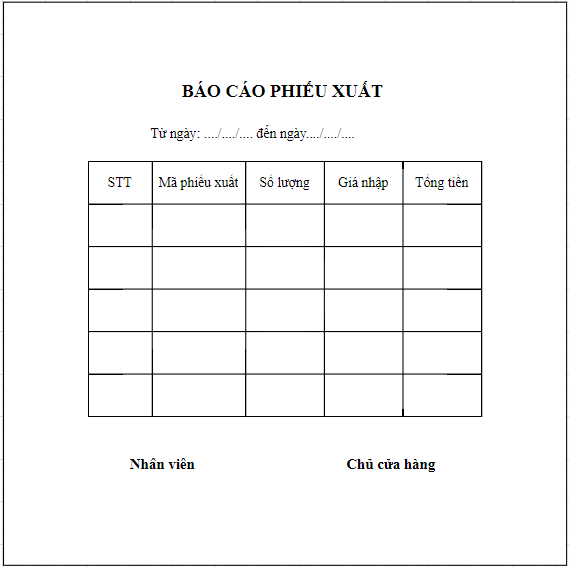
- Hàng sẽ được nhân viên chuyển vào kho, nhân viên sẽ cập nhật số lượng, cập nhật thêm những mặt hàng mới và xóa bỏ những mặt hàng không còn bán nữa. Thông tin mặt hàng sẽ được cửa hàng lưu lại.

### 1.4.4 Báo cáo thống kê

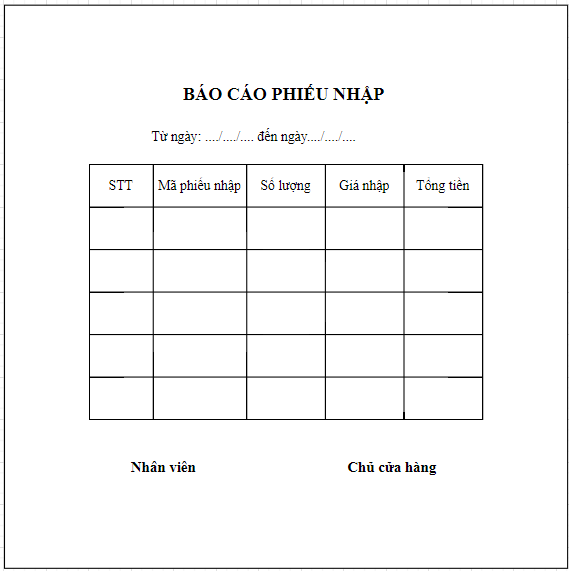
- Hàng tháng chủ cửa hàng sẽ gửi yêu cầu báo cáo thống kê. Nhân viên lập thống kê và báo cáo bày lên chủ cửa hàng xem xét và xử lý.

- Báo cáo thống kê bao gồm:

* Thống kê phiếu xuất



Thống kê phiếu nhập



Thống kê tồn kho



-Quản lý dựa trên các báo cáo nhận được và tình hình thực tế của cửa hàng sẽ đưa ra các yêu cầu tới từng bộ phận như: quyết định nhập hàng; tăng lương, thưởng cho nhân viên; hình thức khuyến mại bán hàng...

## 1.5 Mục tiêu quản lý

**-** Xây dựng một phần mềm giúp cho việc theo dõi cũng như quản lý kinh doanh cửa hàng trở lên nhanh hơn về thời gian chính xác hơn về xử lý dữ liệu.

-Phần mềm quản lý cửa hàng bán giày dép nhằm mục tiêu quản lý các hoạt động trong cửa hàng, nhằm đáp ứng nhu cầu quan trọng nhất trong kinh doanh là tăng lợi nhuận.

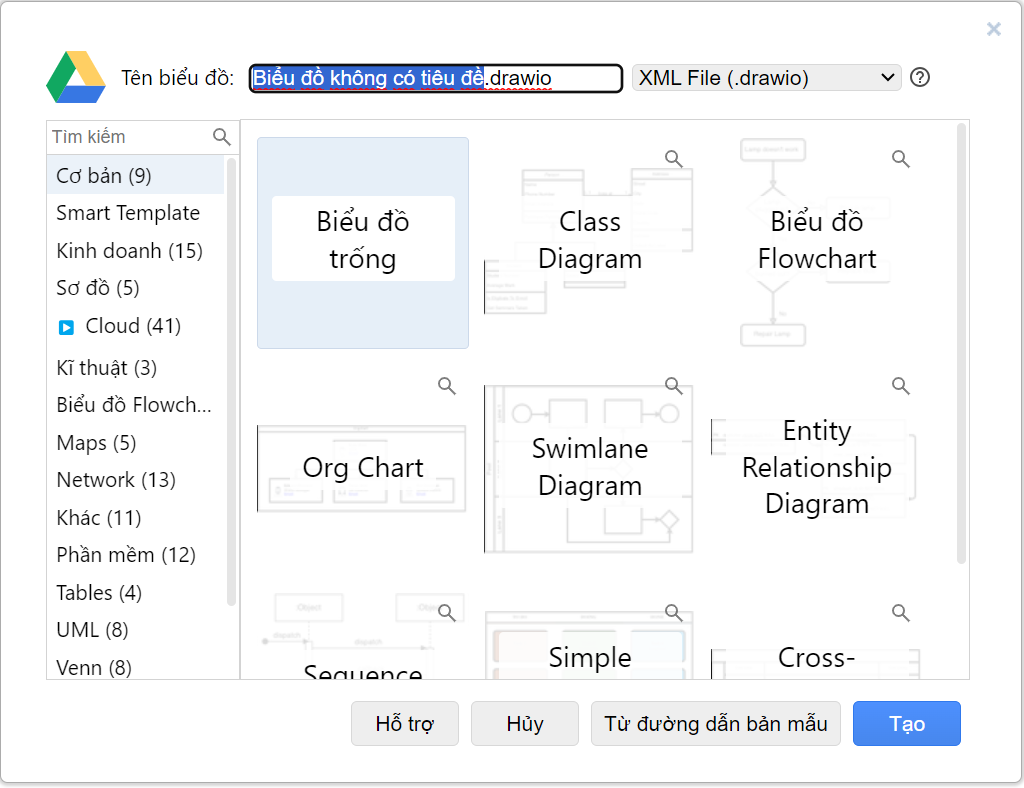
- Khi áp dụng công nghệ thông tin vào việc quản lý sẽ giảm thiểu công sức, thời gian, nhằm giảm thao tác thủ công, mang lại tính chính xác và hiệu quả trong công tác quản lý hoạt động kinh doanh và giúp nhà quản lý nắm bắt tình hình hoạt động của đại lý để có thể đưa ra nhưng quyết định nhanh chóng.

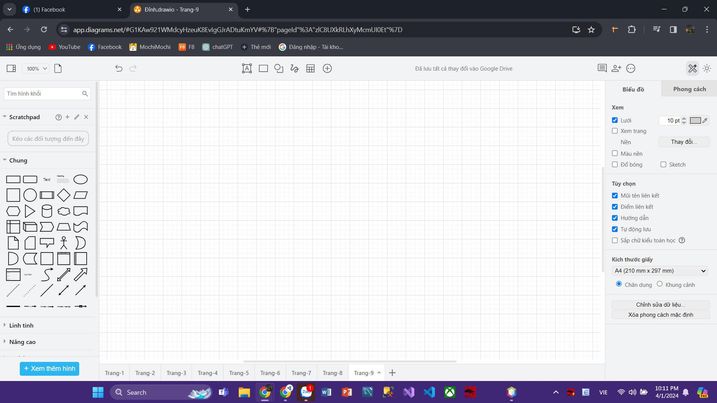
- Tổng hợp, báo cáo thống kê được thuận tiện nhanh chóng.

# PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG

## 2.1 Công cụ sử dụng

Draw.io được biết đến là một website cung cấp nền tảng cho người dùng vẽ các biểu đồ, mô hình, sơ đồ đơn giản. Đặc biệt, người dùng có thể sử dụng online không cần cài đặt vào máy, không bị giới hạn số lần sử dụng và hoàn toàn miễn phí.

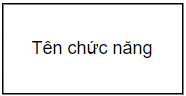




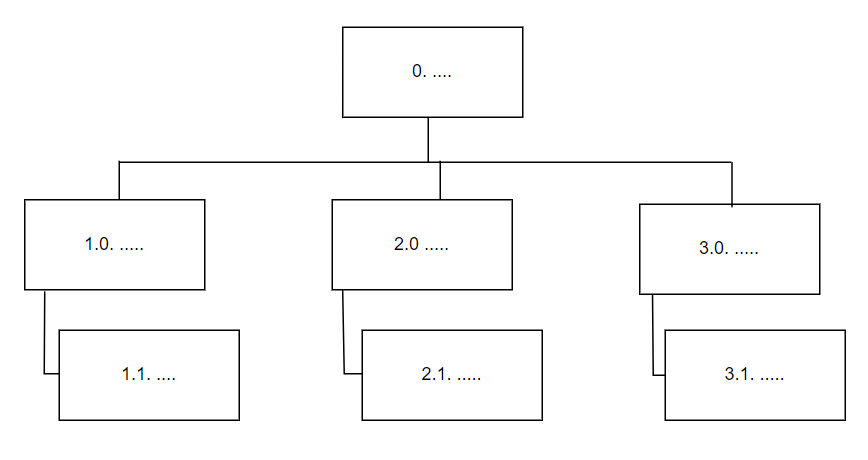
## 2.2 Xây dựng mô hình phân cấp chức năng

**\* Các ký pháp sử dụng trong mô hình:**

- Hình chữ nhật (có tên chức năng bên trong): dùng để mô tả một chức năng chính của hệ thống.



- Đường gấp khúc hình cây: dùng để nối một chức năng ở mức trên với các chức năng ở mức dưới.(trực tiếp phân chia chức năng này).



**\* Các nguyên tắc áp dụng trong mô hình:**

- Đặt tên chức năng: tên chức năng phải là một mệnh đề gồm: động từ + bổ ngữ

- Phân rã có thứ bậc các chức năng:

* Mỗi chức năng phải là một bộ phận thực sự của cùng một chức năng (cụ thể) ở mức trên( tính thực chất của mỗi chức năng).
* Việc thực hiện tất cả các chức năng ở mức dưới phải đảm bảo thực hiện được chức năng ở mức trên đã phân ra chúng( tính đầy đủ của chức năng con).

**\* Các cách xây dựng mô hình chức năng nghiệp vụ:**

- Cách 1: Theo phương pháp kiểm sát, tiếp cận thông tin từ trên xuống( top down).

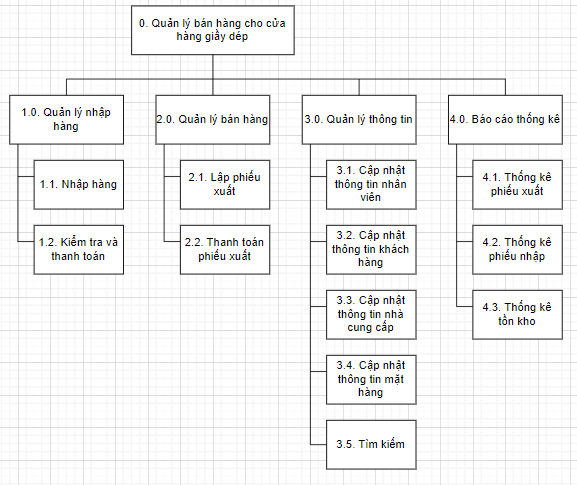
* Sơ đồ BFD được hình thành bằng cách phân chia từ chức năng lớn( tổng quát) thành các chức năng nhỏ hơn.
* Tiếp tục phân rã đến từng nhiệm vụ hoặc từng nhóm công việc nhỏ do từng cá nhân thực hiện.

- Cách 2: Mô hình được xây dựng theo chiều từ dưới lên( bottom up), thường được áp dụng đối với một hệ thống nhỏ, có thể biết ngay được các công việc chi tiết.

* Lần lượt nhóm dần các chức năng ở mức dưới theo từng nhóm, đặt tên chức năng mới cho nhóm chức năng chung này.
* Tiếp tục nhóm cho các chức năng mức trên, cho đến khi đạt được một chức năng chung duy nhất của hệ thống. [1]

**\* Danh sách hồ sơ dữ liệu:**

1. Danh sách mặt hàng
2. Danh sách nhà cung cấp
3. Danh sách nhân viên
4. Danh sách khách hàng
5. Bản báo cáo
6. Phiếu nhập
7. Phiếu xuất



Hình 2.1. Sơ đồ phân cấp chức năng của quản lý shop giày dép

**\* Mô tả các chức năng:**

-Mức 0: Có một chức năng duy nhất là quản lý shop giày dép

-Mức 1**:** Các chức năng chính:

1.0 Quản lý nhập hàng

2.0 Quản lý bán hàng

3.0 Quản lý thông tin

4.0 Báo cáo thống kê

**-**Mức 2: Các chức năng chi tiết của từng chức năng chính

1.0 Quản lý nhập hàng

1.1 Nhập hàng: Nhân viên nhập hàng từ nhà cung cấp

1.2 Kiểm tra và thanh toán: Nhân viên có nhiệm vụ kiểm tra hàng mà bên NCC giao đến và thanh toán.

2.0 Quản lý bán hàng

2.1 Lập phiếu xuất: tiếp nhận mặt hàng khách mua và thông tin khách hàng để tạo phiếu xuất

2.2 Thanh toán: khách nhận phiếu xuất và thanh toán cho nhân viên

3.0 Quản lý thông tin

3.1 Cập nhật thông tin NV: NV sẽ cung cấp đầy đủ thông tin cá nhân của mình cho cửa hàng, mọi thông tin của nv sẽ được lưu trữ lại.

3.2 Cập nhật thông tin KH: khách hàng sẽ cung cấp thông tin cho NV, nhân viên sẽ nhập và lưu lại thông tin cho KH.

3.3 Cập nhật thông tin NCC: bên NCC sẽ gửi thông tin cửa hàng, nhân viên sẽ nhập lại thông tin và lưu trữ lại thông tin.

3.4 Cập nhật thông tin mặt hàng: nhân viên sẽ cập nhật số lượng hàng có trong kho, thêm những mặt hàng mới và xóa những mặt hàng mà shop không còn bán nữa.

3.5 Tìm kiếm: mọi thông tin về NCC, KH, NV, MH đều có thể tìm kiếm trên hệ thống của cửa hàng.

4.0 Báo cáo thống kê

4.1 Thống kê phiếu xuất: lập báo cáo thống kê số lượng hàng bán.

4.2 Thống kê phiếu nhập: lập báo cáo thống kê số lượng hàng nhập.

4.3 Thống kê tồn kho: thống kê hàng còn tồn.

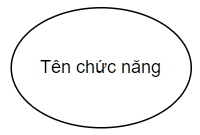
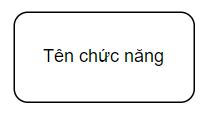
## 2.3 Các ký hiệu được sử dụng trong mô hình DFD

### 2.3.1 Chức năng( tiến trình)

- Định nghĩa: là một hoạt động có liên quan đến sự biến đổi hoặc tác động lên thông tin như thêm vào, xóa, thay đổi thông tin trong hệ thống.

- Cách đặt tên: động từ + bổ ngữ

- Ký hiệu: hình chữ nhật góc tròn hoặc hình eclipse



### 2.3.2 Luồng dữ liệu

- Định nghĩa: là luồng thông tin vào hoặc ra khỏi chức năng( ô xử lý)

- Cách đặt tên: danh từ + tính từ

- Ký hiệu: mũi tên bên trên có ghi thông tin di chuyển



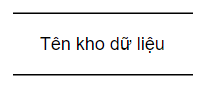
- Chú ý: các luồng thông tin khác nhau phải có tên gọi khác nhau. Luồng dữ liệu chỉ chứa thông tin logic chứ không phải tài liệu vật lý.

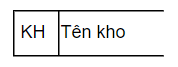
### 2.3.3 Kho dữ liệu

- Định nghĩa: là nơi biểu diễn thông tin cần cất giữ, để một hoặc nhiều chức năng sử dụng chúng.

- Cách đặt tên: danh từ + tính từ( mô tả nội dung dữ liệu trong kho)

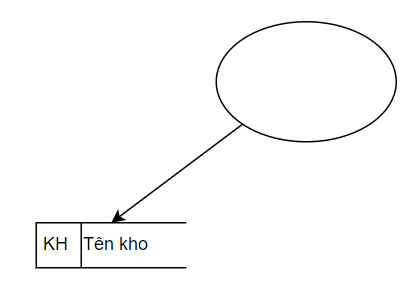
- Ký hiệu:(bên trong có danh từ + tính từ)



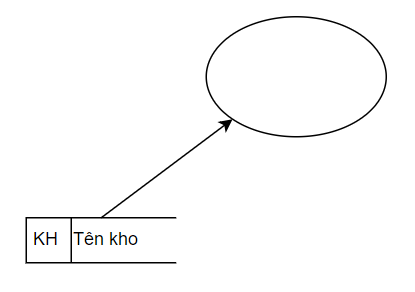


- Một số quy tắc:

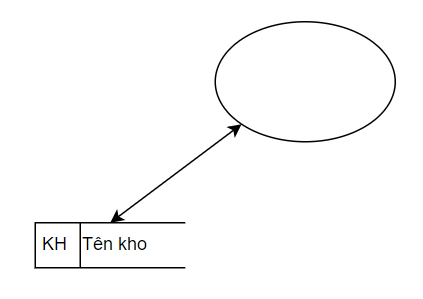
+ Đưa dữ liệu vào kho:



+ Lấy dữ liệu ra khỏi kho:



+ Cập nhật dữ liệu:

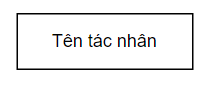


### 2.3.4 Tác nhân ngoài

- Định nghĩa: là một người hoặc một nhóm người nằm ngoài hệ thống nhưng có trao đổi thông tin trực tiếp với hệ thống. Sự có mặt của các nhân tố này trên sơ đồ chỉ ra giới hạn của hệ thống, định mối quan hệ của hệ thống với thế giới bên ngoài.

- Cách đặt tên: danh từ

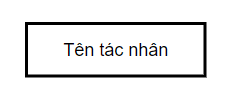
- Ký hiệu: hình chữ nhật( bên trong có danh từ)



### 2.3.5 Tác nhân trong

- Định nghĩa: là một chức năng con của hệ thống đang xét nhưng được trình bày ở một trang khác của mô hình.

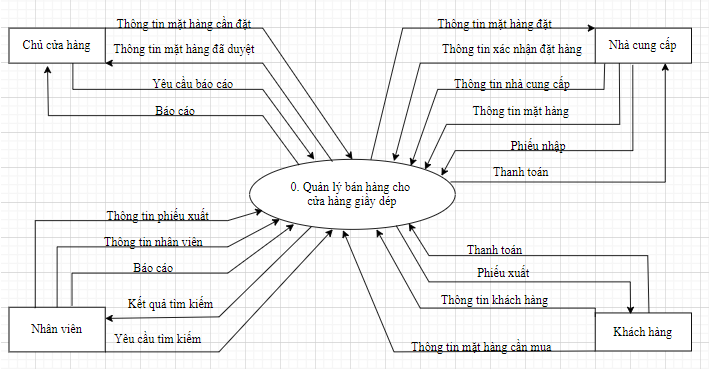
- Cách đặt tên: động từ + bổ ngữ

- Ký hiệu: hình chữ nhật nét đậm(bên trong có động từ + bổ ngữ) [1] 

## 2.4 Xây dựng sơ đồ ngữ cảnh

- Mô hình luồng dữ liệu mức khung cảnh gồm một chức năng duy nhất biểu thị toàn bộ hệ thống đang nghiên cứu, chức năng này được nối với mọi tác nhân ngoài của hệ thống.

- Các luồng dữ liệu giữa chức năng và tác nhân ngoài chỉ thông tin vào và ra của hệ thống. [1]



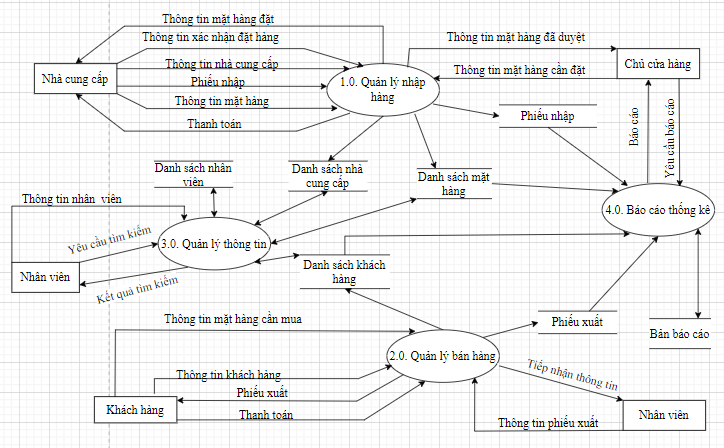
Hình 2.2. Sơ đồ mức ngữ cảnh

## 2.5 Xây dựng sơ đồ luồng dữ liệu mức đỉnh

- Với mức đỉnh các tác nhân ngoài của hệ thống ở mức khung cảnh được giữ nguyên với các luồng thông tin vào ra.

- Hệ thống được phân rã thành các chức năng mức đỉnh là các tiến trình chính bên trong hệ thống theo mô hình phân rã chức năng ở mức 1.

- Xuất hiện thêm các kho dữ liệu và luồng thông tin trao đổi giữa các chức năng mức đỉnh. [1]



Hình 2.3. Sơ đồ luồng dữ liệu mức đỉnh

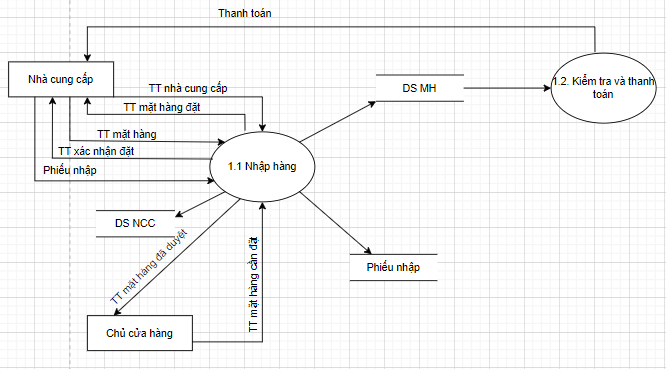
## 2.6 Xây dựng sơ đồ luồng dữ liệu mức dưới đỉnh

- Ở mức này thực hiện phân rã đối với mỗi chức năng của mức đỉnh.

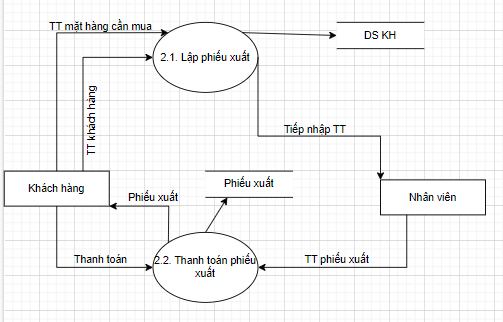
- Khi thực hiện mức phân rã này vẫn phải căn cứ vào mô hình phân rã chức năng để xác định các chức năng con sẽ xuất hiện trong mô hình luồng dữ liệu.

- Việc phân rã có thể tiếp tục cho đến khi đủ số mức cần thiết.

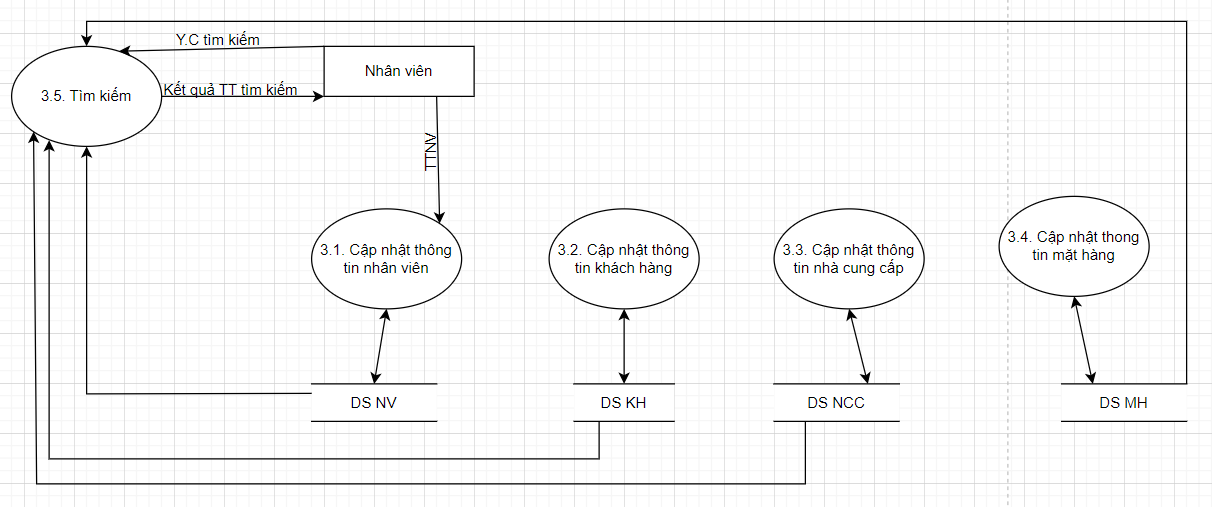
- Khi phân rã các chức năng cần phải đảm bảo tất cả các luồng thông tin vào ra ở chức năng mức cao hơn phải có mặt trong các chức năng mức thấp hơn và ngược lại. [1]



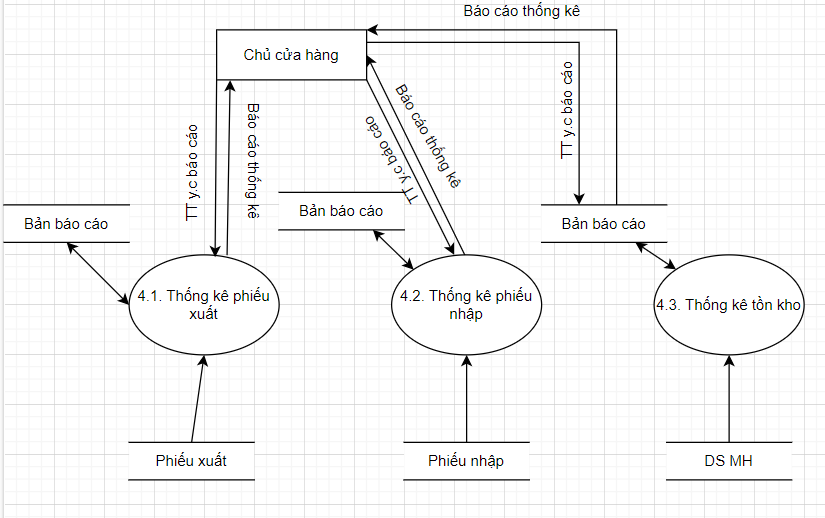
Hình 2.4. Sơ đồ luồng dữ liệu mức dưới đỉnh của chức năng quản lý nhập hàng.



Hình 2.5. Sơ đồ luồng dữ liệu mức dưới đỉnh của chức năng quản lý bán hàng.



Hình 2.6. Sơ đồ luồng dữ liệu mức dưới đỉnh của chức năng quản lý thông tin.



Hình 2.7. Sơ đồ luồng dữ liệu mức dưới đỉnh của chức năng báo cáo thống kê.

## 2.7 Phân tích và thiết kế về dữ liệu

**\*Mô hình thực thể mối quan hệ - Sơ đồ ER**

Từ trên đây ta xác định được các thực thể và liên kết sau:

a. Thực thể

NhaCungCap(MaNCC, TenNCC, DienThoai, DiaChi)

KhachHang(MaKH, TenKH, DienThoai, DiaChi)

NhanVien(MaNV, TenNV, MatKhau, DienThoai, DiaChi, HinhAnh)

HangHoa(MaHH, TenHH, TonKho, DonGia, HinhAnh)

LoaiHangHoa(MaLHH, TenLHH)

b. Xác định liên kết

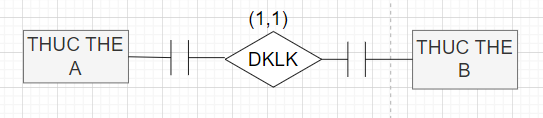
HangHoa- “Thuộc ”- LoaiHangHoa

NhanVien – “Nhập” – HangHoa – “Từ” – NhaCungCap

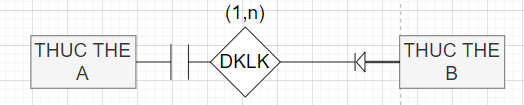
NhanVien – “Bán” – HangHoa – “Cho” – KhachHang

\*Quy ước:

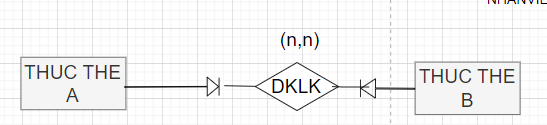
-Mối quan hệ 1-1:



-Mối quan hệ 1-n

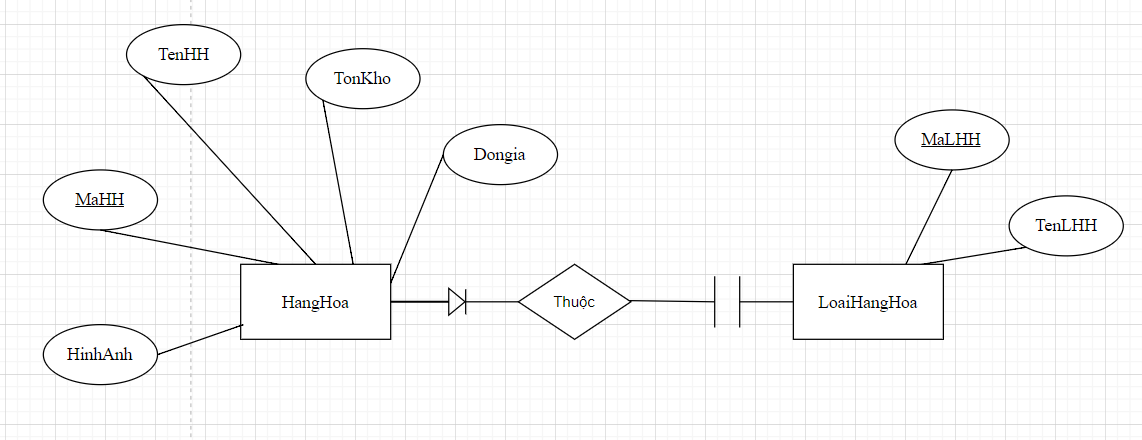


-Mối quan hệ n-n:

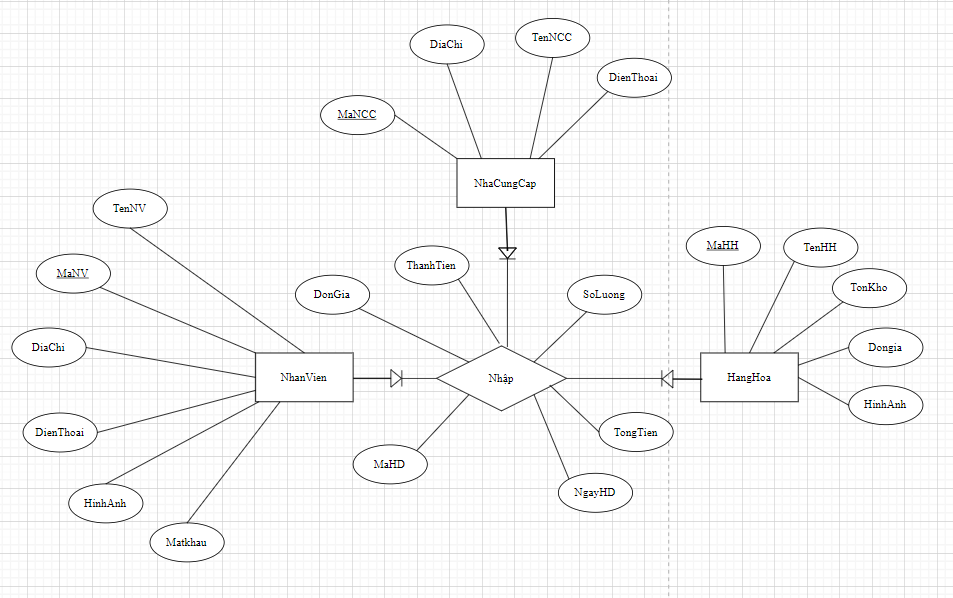


**\*Mô hình khái niệm dữ liệu :**

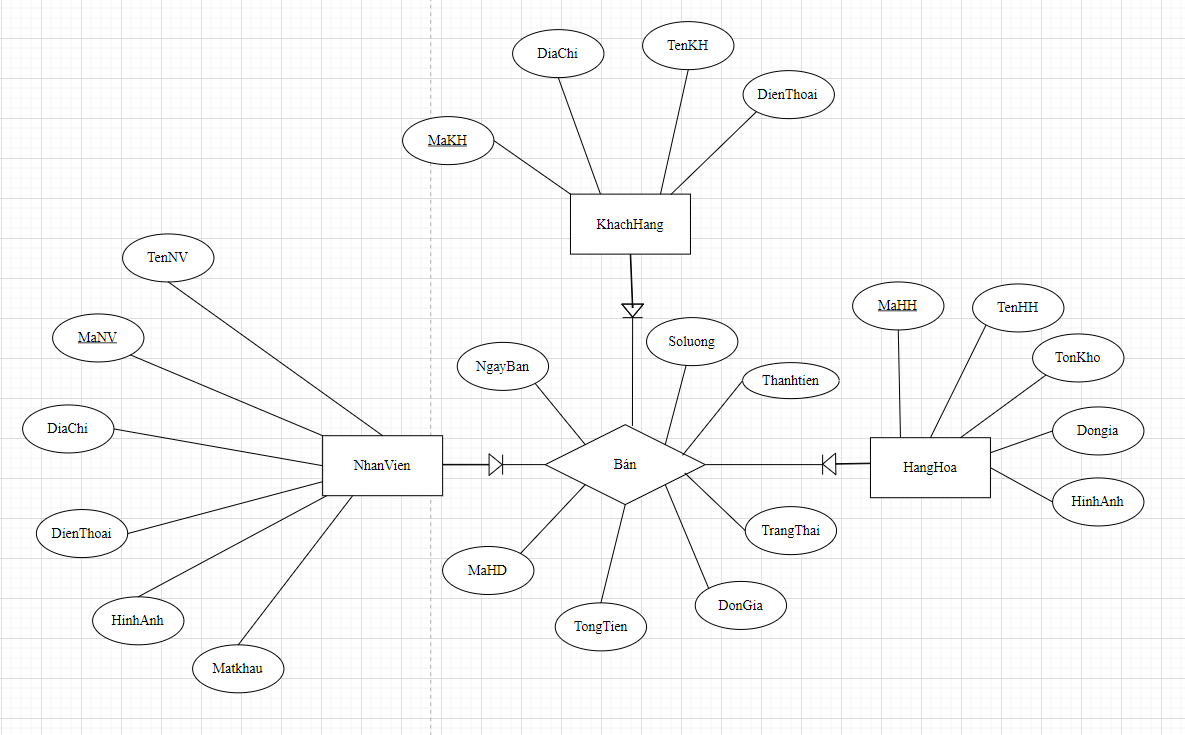
\*HangHoa -Thuộc – LoaiHangHoa



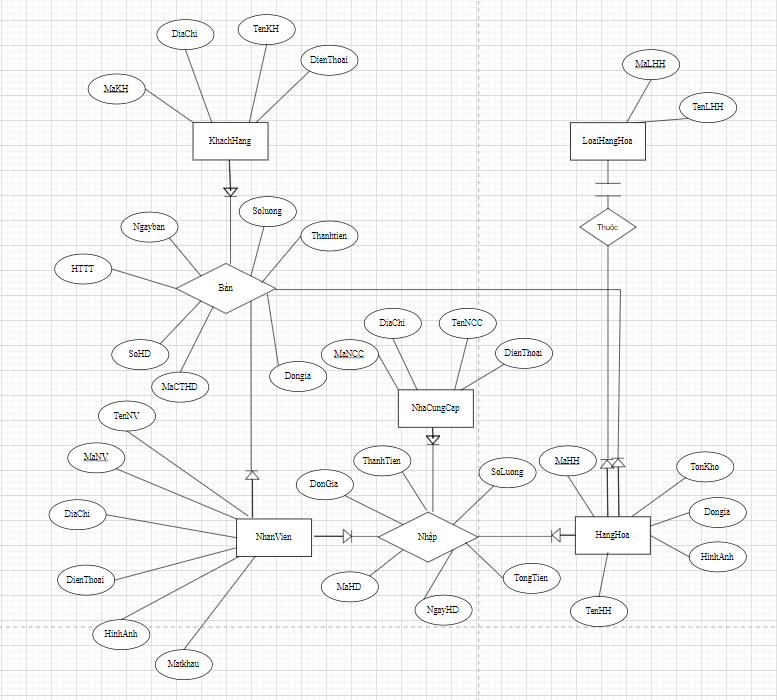
\*NhanVien- Nhập - HangHoa- Từ- NhaCungCap



\*NHANVIEN - Bán - MATHANG - Cho – KHACHHANG



\*ER



**\*Sau khi chuẩn hóa ta được mô hình quan hệ như sau:**

1. NhaCungCap(MaNCC, TenNCC, DienThoai, DiaChi)
2. KhachHang(MaKH, TenKH, DienThoai, DiaChi)
3. NhanVien(MaNV, TenNV, MatKhau, DienThoai, DiaChi, HinhAnh)
4. HangHoa(MaHH, TenHH, TonKho, DonGia, MaNCC, MaLHH, HinhAnh)
5. NhapKho(MaHD, NgayHD, MaNV, TongTien)
6. ChiTietNhapKho(MaHD, MaHH, DonGia, SoLuong, ThanhTien)
7. XuatKho(MaHD, MaKH, NgayBan, MaNV, TongTien, TrangThai)
8. ChiTietXuatKho(MaHD, MaHH, DonGia, SoLuong, ThanhTien)
9. LoaiHangHoa(MaLHH, TenLHH)

# CHƯƠNG 3: XÂY DỰNG PHẦN MỀM CHO BÀI TOÁN QUẢN LÝ

**Mục đích**

Nội dung chương 3 hướng dẫn cách thiết kế CSDL và xây dựng phần mềm hoàn chỉnh cho dự án:

• Hướng dẫn thiết kế các bảng dữ liệu quan hệ

• Chuẩn hóa CSDL vừa thiết kế

• Lựa chọn công cụ giải quyết bài toán

• Hướng dẫn thiết kế giao diện cho phần mềm

• Hướng dẫn cách thức kết nối CSDL

• Hướng dẫn một số thao tác cụ thể với CSDL

## 3.1. Thiết kế CSDL cho dự án

### 3.1.1 Thiết kế các bảng quan hệ

\*Mã hóa các thực thể thành bảng quan hệ: Mỗi thực thể ở mô hình ER thành 1 bảng quan hệ ở mô hình quan hệ

|  |  |
| --- | --- |
| **Thực thể** | **Bảng quan hệ** |
| NhaCungCap | NhaCungCap |
| NhanVien | NhanVien |
| KhachHang | KhachHang |
| HangHoa | HangHoa |
| NhapKho | NhapKho |
| ChiTietNhapKho | ChiTietNhapKho |
| XuatKho | XuatKho |
| ChiTietXuatKho | ChiTietXuatKho |
| LoaiHangHoa | LoaiHangHoa |

\*Bảng NhanVien dùng để lưu lại thông tin nhân viên

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tên trường** | **Kiểu dữ liệu** | **Cỡ dữ liệu** | **Ràng buộc** | **Diễn giải** |
| MaNV | varchar | 32 | Khóa chính | Mã nhân viên |
| TenNV | nvarchar | 128 | Not Null | Tên nhân viên |
| MatKhau | varchar | 256 | Not Null | Mật khẩu |
| DienThoai | varchar | 16 |  | Điện thoại |
| DiaChi | nvarchar | 128 |  | Địa chỉ |
| HinhAnh | nvarchar | Max | Not Null | Hình ảnh |

\*Bảng NhaCungCap dùng để lưu thông tin nhà cung cấp

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tên trường** | **Kiểu dữ liệu** | **Cỡ dữ liệu** | **Ràng buộc** | **Diễn giải** |
| MaNCC | varchar | 32 | Khóa chính | Mã nhà cung cấp |
| TenNCC | nvarchar | 128 | Not Null | Tên nhà cung cấp |
| DiaChi | nvarchar | 128 |  | Địa chỉ |
| DienThoai | varchar | 16 |  | Điện thoại |

\*Bảng KhachHang dùng để lưu lại thông tin khách hàng gồm

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tên trường** | **Kiểu dữ liệu** | **Cỡ dữ liệu** | **Ràng buộc** | **Diễn giải** |
| MaKH | varchar | 32 | Khóa chính | Mã khách hàng |
| TenKH | nvarchar | 128 | Not Null | Tên khách hàng |
| DienThoai | varchar | 16 |  | Điện thoại |
| DiaChi | nvarchar | 128 |  | Địa chỉ |

\*Bảng HangHoa dùng để lưu thông tin chi tiết mặt hàng

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tên trường** | **Kiểu dữ liệu** | **Cỡ dữ liệu** | **Ràng buộc** | **Diễn giải** |
| MaHH | varchar | 32 | Khóa chính | Mã hàng hóa |
| TenHH | nvarchar | 128 | Not Null | Tên hàng hóa |
| HinhAnh | nvarchar | Max |  | Hình ảnh |
| MaLHH | varchar | 32 | Khóa ngoại | Mã loại hàng hóa |
| MaNCC | varchar | 32 | Khóa ngoại | Mã nhà cung cấp |
| TonKho | decimal |  | Not Null | Tồn kho |
| Dongia | decimal |  | Not Null | Đơn giá |

|  |  |  |
| --- | --- | --- |
| **Tên trường** | **Cột khóa ngoại** | **Quan hệ với bảng** |
| MaNCC | MaNCC | NhaCungCap |
| MaLHH | MaLHH | LoaiHangHoa |

\*Bảng NhapKho dùng để lưu lại thông tin nhập kho gồm

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tên trường** | **Kiểu dữ liệu** | **Cỡ dữ liệu** | **Ràng buộc** | **Diễn giải** |
| MaHD | varchar | 32 | Khóa chính | Mã hóa đơn |
| NgayHD | datetime |  | Not Null | Ngày hóa đơn |
| MaNV | varchar | 32 | Khóa ngoại | Mã nhân viên |
| TongTien | decimal |  |  | Tổng tiền |

|  |  |  |
| --- | --- | --- |
| **Tên** | **Cột khóa ngoại** | **Quan hệ với bảng** |
| MaNV | MaNV | NhanVien |

\*Bảng ChiTietNhapKho dùng để lưu trữ thông tin chi tiết

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tên trường** | **Kiểu dữ liệu** | **Cỡ dữ liệu** | **Ràng buộc** | **Diễn giải** |
| MaHD | varchar | 32 | Khóa ngoại | Mã hóa đơn |
| MaHH | varchar | 32 | Khóa ngoại | Mã hàng hóa |
| DonGia | decimal |  | Not Null | Đơn giá |
| SoLuong | decimal |  | Not Null | Số lượng |
| ThanhTien | decimal |  | Not Null | Thành tiền |

|  |  |  |
| --- | --- | --- |
| **Tên trường** | **Cột khóa ngoại** | **Quan hệ với bảng** |
| MaHH | MaHH | HangHoa |
| MaHD | MaHD | NhapKho |

\*Bảng XuatKho dùng để lưu lại thông tin xuất kho gồm

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tên trường** | **Kiểu dữ liệu** | **Cơ sở dữ liệu** | **Ràng buộc** | **Diễn giải** |
| MaHD | varchar | 32 | Khóa chính | Mã hóa đơn |
| NgayBan | datetime |  |  | Ngày bán |
| TongTien | decimal |  | Not Null | Tổng tiền |
| MaNV | varchar | 32 | Khóa ngoại | Mã nhân viên |
| MaKH | varchar | 32 | Khóa ngoại | Mã khách hàng |
| TrangThai | nvarchar | 128 | Not Null | Trạng thái |

|  |  |  |
| --- | --- | --- |
| **Tên trường** | **Cột khóa ngoại** | **Quan hệ với bảng** |
| MaNV | MaNV | NhanVien |
| MaKH | MaKH | KhachHang |

\*Bảng ChiTietXuatKho dùng để lưu trữ thông tin chi tiết

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tên trường** | **Kiểu dữ liệu** | **Độ rộng dữ liệu** | **Ràng buộc** | **Diễn giải** |
| MaHD | varchar | 32 | Khóa ngoại | Mã hóa đơn |
| SoLuong | decimal |  | Not Null | Số lượng |
| Dongia | decimal |  | Not Null | Đơn giá |
| Thanhtien | decimal |  | Not Null | Tổng tiền |
| MaHH | varchar | 32 | Khóa ngoại | Mã hàng hóa |

|  |  |  |
| --- | --- | --- |
| **Tên trường** | **Cột khóa ngoại** | **Quan hệ với bảng** |
| MaHH | MaHH | HangHoa |
| MaHD | MaHD | XuatKho |

\*Bảng LoaiHangHoa lưu thông tin phân loại hàng hóa

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tên trường** | **Kiểu dữ liệu** | **Cơ sở dữ liệu** | **Ràng buộc** | **Diễn giải** |
| MaLHH | varchar | 32 | Khóa chính | Mã loại hàng hóa |
| TenLHH | Nvarchar | 128 | Not Null | Tên loại hàng hóa |

### 3.1.2 Chuẩn hóa CSDL của dự án Các dạng chuẩn:

**Dạng chuẩn thứ nhất (1NF – First Normal Form)**

Một lược đồ quan hệ R được gọi là ở dạng chuẩn một nếu và chỉ nếu toàn bộ các miền có mặt trong R đều chỉ chứa các giá trị nguyên tố.

Từ định nghĩa này cho ta thấy rằng bất kỳ quan hệ chuẩn hóa nào cũng ở dạng 1NF và tất nhiên điều đó đúng.

**Dạng chuẩn thứ 2 (2NF)**

Lược đồ quan hệ R ở dạng chuẩn thứ hai nếu nó ở dạng 1NF và nếu mỗi thuộc tính không khóa của R là phụ thuộc hàm đầy đủ vào khóa chính.

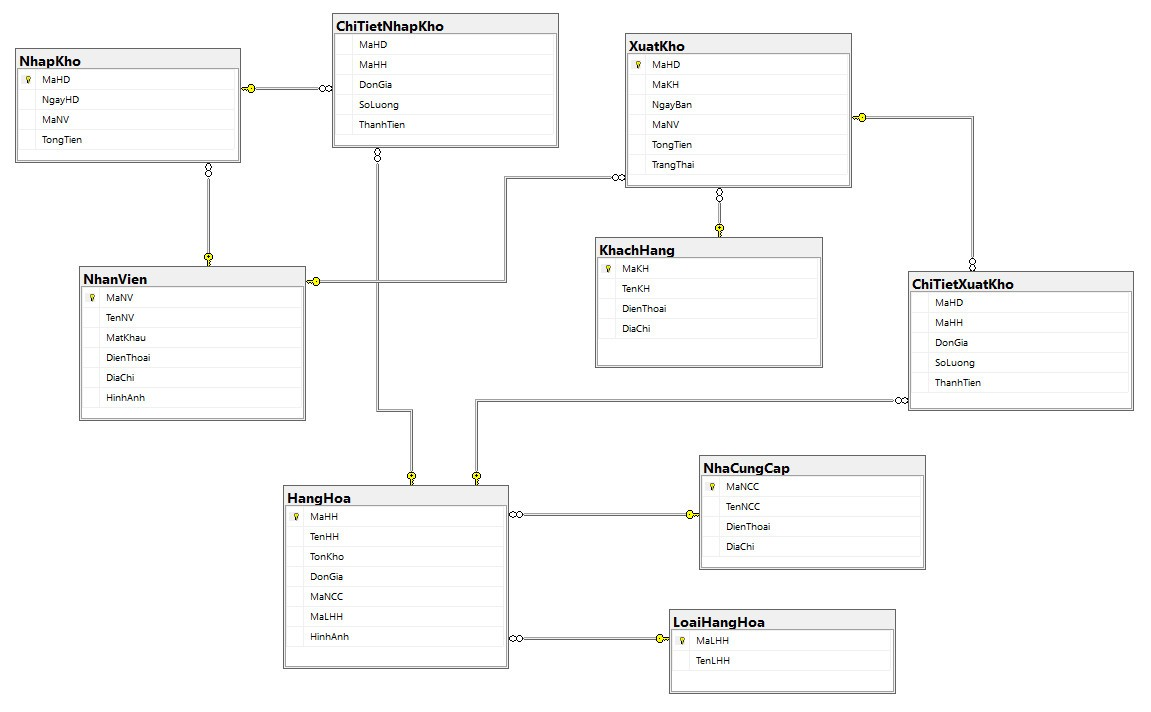
**Dạng chuẩn thứ 3 (3NF)**

Lược đồ quan hệ R là ở dạng chuẩn thứ ba (3NF) nếu nó là 2NF và mỗi thuộc tính không khóa của R là không phụ thuộc hàm bắc cầu vào khóa chính.

**Dạng chuẩn Boye- Codd (BCNF)**

Lược đồ quan hệ R với tập các phụ thuộc hàm được gọi là ở dạng chuẩn BCNF nếu X→A thỏa trên R, A∈X thì X là một khóa của R. Định lý: Nếu một lược đồ quan hệ R với tập phụ thuộc hàm F là ở BCNF thì nó là ở 3NF. Các bảng Cơ sở dữ liệu của dự án đã ở dạng chuẩn 3NF

### 3.1.3 Mô hình quan hệ (Database Diagrams) và Code SQL



Hình 3.1 Sơ đồ Database Diagrams của hệ thống

* **Code SQL:**

CREATE DATABASE dbQuanLyBanGiayDep

GO

USE dbQuanLyBanGiayDep

GO

CREATE TABLE NhanVien

(

MaNV VARCHAR(32) PRIMARY KEY NOT NULL,

TenNV NVARCHAR(128) NOT NULL,

MatKhau VARCHAR(256) NOT NULL,

DienThoai VARCHAR(16),

DiaChi NVARCHAR(128),

HinhAnh NVARCHAR(MAX)

)

GO

CREATE TABLE KhachHang

(

MaKH VARCHAR(32) PRIMARY KEY NOT NULL,

TenKH NVARCHAR(128) NOT NULL,

DienThoai VARCHAR(16),

DiaChi NVARCHAR(128)

)

GO

CREATE TABLE LoaiHangHoa

(

MaLHH VARCHAR(32) PRIMARY KEY NOT NULL,

TenLHH NVARCHAR(128) NOT NULL

)

GO

CREATE TABLE NhaCungCap

(

MaNCC VARCHAR(32) PRIMARY KEY NOT NULL,

TenNCC NVARCHAR(128) NOT NULL,

DienThoai VARCHAR(16),

DiaChi NVARCHAR(128)

)

GO

CREATE TABLE HangHoa

(

MaHH VARCHAR(32) PRIMARY KEY NOT NULL,

TenHH NVARCHAR(128) NOT NULL,

TonKho DECIMAL NOT NULL,

DonGia DECIMAL NOT NULL,

MaNCC VARCHAR(32) FOREIGN KEY REFERENCES NhaCungCap(MaNCC) ON DELETE CASCADE ON UPDATE CASCADE NOT NULL,

MaLHH VARCHAR(32) FOREIGN KEY REFERENCES LoaiHangHoa(MaLHH) ON DELETE CASCADE ON UPDATE CASCADE NOT NULL,

HinhAnh NVARCHAR(MAX)

)

GO

CREATE TABLE NhapKho

(

MaHD VARCHAR(32) PRIMARY KEY NOT NULL,

NgayHD DATETIME NOT NULL,

MaNV VARCHAR(32) FOREIGN KEY REFERENCES NhanVien(MaNV) ON DELETE CASCADE ON UPDATE CASCADE NOT NULL,

TongTien decimal NOT NULL,

)

GO

CREATE TABLE ChiTietNhapKho

(

MaHD VARCHAR(32) FOREIGN KEY REFERENCES NhapKho(MaHD) ON DELETE CASCADE ON UPDATE CASCADE NOT NULL,

MaHH VARCHAR(32) FOREIGN KEY REFERENCES HangHoa(MaHH) ON DELETE CASCADE ON UPDATE CASCADE NOT NULL,

DonGia decimal NOT NULL,

SoLuong decimal NOT NULL,

ThanhTien decimal NOT NULL

)

GO

CREATE TABLE XuatKho

(

MaHD VARCHAR(32) PRIMARY KEY NOT NULL,

MaKH VARCHAR(32) FOREIGN KEY REFERENCES khachhang(makh) ON DELETE CASCADE ON UPDATE CASCADE NOT NULL,

NgayBan DATETIME NOT NULL,

MaNV VARCHAR(32) FOREIGN KEY REFERENCES nhanvien(manv) ON DELETE CASCADE ON UPDATE CASCADE NOT NULL,

TongTien DECIMAL NOT NULL,

TrangThai NVARCHAR(128) NOT NULL

)

GO

CREATE TABLE ChiTietXuatKho

(

MaHD VARCHAR(32) FOREIGN KEY REFERENCES XuatKho(MaHD) ON DELETE CASCADE ON UPDATE CASCADE NOT NULL,

MaHH VARCHAR(32) FOREIGN KEY REFERENCES HangHoa(MaHH) ON DELETE CASCADE ON UPDATE CASCADE NOT NULL,

DonGia DECIMAL NOT NULL,

SoLuong DECIMAL NOT NULL,

ThanhTien DECIMAL NOT NULL

)

INSERT INTO NhanVien (MaNV,TenNV,MatKhau,DienThoai,DiaChi)VALUES(N'AD',N'ADMIN',N'123',N'0354125623',N'Hà Nội')

GO

INSERT INTO KhachHang (MaKH,TenKH,DienThoai,DiaChi)VALUES(N'KH001',N'Linh Lung',N'0245126325',N'Hà Nội')

INSERT INTO KhachHang (MaKH,TenKH,DienThoai,DiaChi)VALUES(N'KH002',N'Nguyễn Mạnh',N'0541236985',N'Hà Nội')

## 3.2 Xây dựng phần mềm quản lý hệ thống hoàn chỉnh của dự án

### 3.2.1 Lựa chọn công cụ: Ngôn ngữ lập trình Winforms C#

**Tổng quan về ngôn ngữ lập trình Winforms**

* Các ứng dụng của Windows rất dễ sử dụng, nhưng rất khó đối với người đã tạo lập ra chúng. Để đạt được tính dễ dùng đòi hỏi người lập trình phải bỏ ra rất nhiều công sức để cài đặt.
* Lập trình trên Windows khó và phức tạp hơn nhiều so với lập trình trên DOS. Tuy nhiên lập trình trên Windows sẽ giải quyết được một số vấn đề khó khăn trong môi trường DOS như xây dựng giao diện người dùng, quản lý bộ nhớ ảo, độc lập thiết bị vào ra, thâm nhập Internet, khả năng chia sẻ tài nguyên, ...
* Windows cung cấp các hàm để người lập trình thâm nhập các đặc trưng của hệ điều hành gọi là giao diện lập trình ứng dụng (Application Programming Interface – API). Những hàm này được đặt trong các thư viện liên kết động (Dynamic Link Library – DLL). Các chương trình ứng dụng sử dụng chúng thông qua các lời gọi hàm và chỉ chia sẻ được khi trong máy có cài đặt Windows.

**Các chức năng và phương thức Winform:**

Là một chương trình quản lý bán thức ăn nhanh nên các chức năng thêm, xóa, sửa, lưu là không thể thiếu. Ngoài chức năng quản lý, nhân viên có thể ghi hóa đơn cho khách hàng muốn đặt món và tìm kiếm hóa đơn cũ một cách dễ dàng.

### 3.2.2 Thiết kế giao diện

- Các công cụ chính tạo nên giao diện được sử dụng:

• Label: dùng để trình bày một chuỗi văn bản thông thường nhằm mục đích mô tả thêm thông tin cho đối tượng khác.

• Textbox: dùng để nhập dữ liệu đầu vào, ngoài ra còn có thể dùng để xuất dữ liệu.

• Combobox: Hiển thị như một Textbox kết hợp với một Listbox, cho phép người dùng lựa chọn các mục từ danh sách hoặc nhập giá trị mới.

• Button: là thành phần tương tác cho phép người dùng giao tiếp với một ứng dụng. • Datagridview: dùng để hiển thị và chỉnh sửa dữ liệu dạng bảng.

• Panel: là một container hay cơ chế nhóm control phù hợp cho giao diện chương trình.

* **Kết nối cơ sở dữ liệu SQL**

using System;

using System.Collections.Generic;

using System.Data.SqlClient;

using System.Data;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace PhanMemQuanLyBanGiayDep.Models

{

public class ConnectSQL

{

public static string GetConnectionString()

{

SqlConnectionStringBuilder stringBuilder = new SqlConnectionStringBuilder()

{

InitialCatalog = "dbQuanLyBanGiayDep",

DataSource = @"MSI\SQLEXPRESS01",

IntegratedSecurity = true,

MultipleActiveResultSets = true,

ConnectTimeout = 0,

MaxPoolSize = 500,

TrustServerCertificate = true

};

return stringBuilder.ConnectionString;

}

public static DataTable Load(string sql)

{

DataTable dt = new DataTable();

string conn\_string = GetConnectionString();

SqlConnection sqlConnect = new SqlConnection(conn\_string);

sqlConnect.Open();

SqlCommand cmd = new SqlCommand(sql, sqlConnect);

cmd.CommandType = CommandType.Text;

SqlDataAdapter da = new SqlDataAdapter(cmd);

da.Fill(dt);

return dt;

}

public static bool ExecuteNonQuery(string sql)

{

using (SqlConnection connection = new SqlConnection(GetConnectionString()))

{

try

{

connection.Open();

SqlCommand cmd = new SqlCommand(sql, connection);

cmd.ExecuteNonQuery();

connection.Close();

return true;

}

catch (Exception ex)

{

throw ex;

}

}

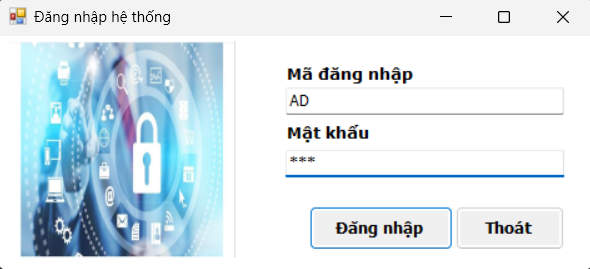
}

}

}

* **Thiết kế chi tiết các giao diện:**

**3.2.2.1 Giao diện form đăng nhập**

****

Hình 3.2 Giao diện Form đăng nhập

* **Thiết kế**

Button frmThoat

Button btnDangNhap

TextBox txtmatkhau

TextBox txtmanv

PictureBox pbKey

* **View code giao diện form đăng nhập**

namespace PhanMemQuanLyBanGiayDep

{

public partial class frmDangNhap : Form

{

DangNhapControllers db = new DangNhapControllers();

public static string manv;

public static string tennv;

public static string matkhau;

public frmDangNhap()

{

InitializeComponent();

}

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

{

DialogResult dr = MessageBox.Show("Bạn có muốn thoát chương trình ?", "Xác nhận", MessageBoxButtons.YesNo, MessageBoxIcon.Question);

if (dr == DialogResult.Yes)

{

this.Close();

}

else

return;

}

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

{

if (txtmanv.Text == "")

{

MessageBox.Show("Mã đăng nhập không được trống", "Thông báo",

MessageBoxButtons.OK, MessageBoxIcon.Error);

txtmanv.Focus();

return;

}

if (txtmatkhau.Text == "")

{

MessageBox.Show("Mật khẩu không được trống", "Thông báo",

MessageBoxButtons.OK, MessageBoxIcon.Error);

txtmatkhau.Focus();

return;

}

DataTable dt = new DataTable();

dt = db.DangNhap(txtmanv.Text.Trim(), txtmatkhau.Text.Trim());

if (dt == null || dt.Rows.Count > 0)

{

this.Hide();

frmMain frm = new frmMain();

manv = txtmanv.Text.Trim();

matkhau = txtmatkhau.Text.Trim();

tennv = db.TenTK(manv).Rows[0][0].ToString();

frm.ShowDialog();

this.Close();

}

else

{

MessageBox.Show("Không đúng tên người dùng hoặc mật khẩu", "Thông báo",

MessageBoxButtons.OK, MessageBoxIcon.Error);

txtmanv.Focus();

}

}

private void frmDangNhap\_Load(object sender, EventArgs e)

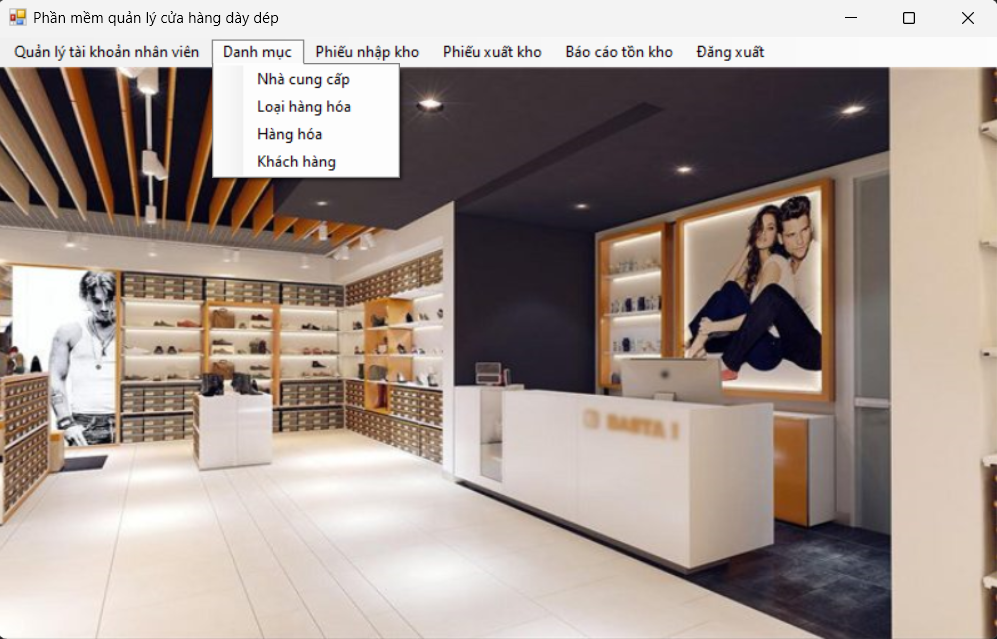
{

}

}

}

* + - 1. **Form giao diện chính**

****

Hình 3.3 Form giao diện chính

* **Thiết kết**

quảnLýTàiKhoảnNhânViênToolStripMenuItem\_Click

nhàCungCấpToolStripMenuItem\_Click

loạiHàngHóaToolStripMenuItem\_Click

hàngHóaToolStripMenuItem\_Click

kháchHàngToolStripMenuItem\_Click

phiếuNhậpKhoToolStripMenuItem\_Click

phiếuXuấtKhoToolStripMenuItem\_Click

báoCáoTồnKhoToolStripMenuItem\_Click

đăngXuấtToolStripMenuItem\_Click

*- Khi người dùng click vào từng menu item thì từng sự kiện ToolStripMenuItem\_Click sẽ được gọi tương ứng.*

* **View code giao diện chính**

namespace PhanMemQuanLyBanGiayDep.Views

{

public partial class frmMain : Form {

public frmMain()

{

InitializeComponent();

}

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

{

frmNhanVien frm = new frmNhanVien();

frm.ShowDialog();

}

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

{

frmNCC frm = new frmNCC();

frm.ShowDialog();

}

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

{

frmLHH frm = new frmLHH();

frm.ShowDialog();

}

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

{

frmHangHoa frm = new frmHangHoa();

frm.ShowDialog();

}

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

{

frmKhachHang frm = new frmKhachHang();

frm.ShowDialog();

}

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

{

frmNhapKho frm = new frmNhapKho();

frm.ShowDialog();

}

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

{

frmXuatKho frm = new frmXuatKho();

frm.ShowDialog();

}

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

{

frmBaoCaoTonKho frm = new frmBaoCaoTonKho();

frm.ShowDialog();

}

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

{

DialogResult dr = MessageBox.Show("Bạn có muốn thoát chương trình ?", "Xác nhận", MessageBoxButtons.YesNo, MessageBoxIcon.Question);

if (dr == DialogResult.Yes)

{

this.Close();

}

else

return;

}

private void frmMain\_Load(object sender, EventArgs e)

{ }

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

{

frmNhanVien frm = new frmNhanVien();

frm.ShowDialog();

}

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

{

frmNCC frm = new frmNCC();

frm.ShowDialog();

}

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

{

frmLHH frm = new frmLHH();

frm.ShowDialog();

}

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

{

frmHangHoa frm = new frmHangHoa();

frm.ShowDialog();

}

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

{

frmKhachHang frm = new frmKhachHang();

frm.ShowDialog();

}

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

{

frmNhapKho frm = new frmNhapKho();

frm.ShowDialog();

}

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

{

frmXuatKho frm = new frmXuatKho();

frm.ShowDialog();

}

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

{

frmBaoCaoTonKho frm = new frmBaoCaoTonKho();

frm.ShowDialog();

}

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

{

DialogResult dr = MessageBox.Show("Bạn có muốn thoát chương trình ?", "Xác nhận", MessageBoxButtons.YesNo, MessageBoxIcon.Question);

if (dr == DialogResult.Yes)

{

this.Close();

}

else

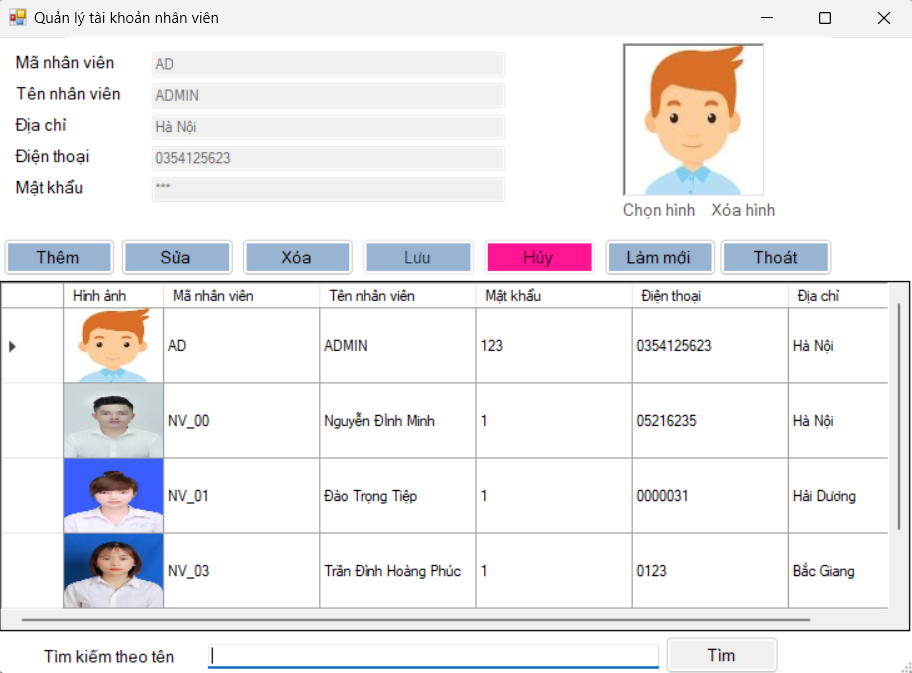
return;

}

}

}

* + - 1. **Giao diện Form quản lý tài khoản nhân viên**

****

Hình 3.4 Giao diện Form quản lý tài khoản nhân viên

* **Thiết kế**

btnThem\_Click: Button thêm nhân viên.

btnSua\_Click: Button sửa thông tin nhân viên bao gồm cả hình ảnh.

btnXoa\_Click: Button xóa thông tin nhân viên.

btnLuu\_Click: Button lưu ( Sau khi thêm hoặc sửa thông tin xong bấm nút Lưu để lưu thông tin nhân viên) .

btnHuy\_Click: Button Hủy quá trình lưu hoặc sửa thông tin nhân viên.

btnLamMoi\_Click: Button làm mới.

btnThoat\_Click: Button thoát sẽ thoát khỏi form quản lý tài khoản nhân viên.

btnTim\_Click: Button Tìm kiếm theo tên giúp tìm kiếm thông tin nhân viên theo tên dễ dàng.

* **View code giao diện quản lý tài khoản nhân viên**

namespace PhanMemQuanLyBanGiayDep.Views

{

public partial class frmNhanVien : Form

{

public frmNhanVien()

{

InitializeComponent();

}

DangNhapControllers dal = new DangNhapControllers();

private bool luu;

private void frmNhanVien\_Load(object sender, EventArgs e)

{

HienThiNhanVien();

boolcontrols(true);

}

private void HienThiNhanVien()

{

DataTable dt = new DataTable();

dt = dal.HienThi(txtTenNVSearch.Text.Trim());

gridview.Rows.Clear();

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

{

string MaNV = dt.Rows[i]["MaNV"].ToString();

string TenNV = dt.Rows[i]["TenNV"].ToString();

string MatKhau = dt.Rows[i]["MatKhau"].ToString();

string DienThoai = dt.Rows[i]["DienThoai"].ToString();

string DiaChi = dt.Rows[i]["DiaChi"].ToString();

string v = dt.Rows[i]["hinhAnh"].ToString();

if (v != "")

{

System.Drawing.Image img = ConvertStringtoImage(dt.Rows[i]["hinhAnh"].ToString());

gridview.Rows.Add(new object[] { img,MaNV, TenNV, MatKhau, DienThoai, DiaChi });

}

else

{

gridview.Rows.Add(new object[] { null, MaNV, TenNV, MatKhau, DienThoai, DiaChi });

foreach (var column in gridview.Columns)

{

if (column is DataGridViewImageColumn)

(column as DataGridViewImageColumn).DefaultCellStyle.NullValue = null;

}

}

}

foreach (DataGridViewRow r in gridview.Rows)

r.Height = 60;

if (gridview.Rows.Count == 0)

{

txtMaNV.Text = "";

txtTenNV.Text = "";

txtMatKhau.Text = "";

txtDienThoai.Text = "";

txtDiaChi.Text = "";

pchinhAnh.Image = null;

}

else

{

var row = this.gridview.Rows[0];

txtMaNV.Text = row.Cells[1].Value.ToString().Trim();

txtTenNV.Text = row.Cells[2].Value.ToString().Trim();

txtMatKhau.Text = row.Cells[3].Value.ToString().Trim();

txtDienThoai.Text = row.Cells[4].Value.ToString().Trim();

txtDiaChi.Text = row.Cells[5].Value.ToString().Trim();

DataTable dtimg = new DataTable();

dtimg = dal.HinhAnhNhanVien(txtMaNV.Text.Trim());

if (dtimg.Rows.Count > 0)

{

if (dtimg.Rows[0]["HinhAnh"].ToString() != "")

{

pchinhAnh.Image = ConvertStringtoImage(dtimg.Rows[0]["HinhAnh"].ToString());

}

else

{

dtHinhAnh = "";

pchinhAnh.Image = null;

}

}

}

}

public System.Drawing.Image ConvertStringtoImage(string commands)

{

byte[] photoarray = Convert.FromBase64String(commands);

MemoryStream ms = new MemoryStream(photoarray, 0, photoarray.Length);

ms.Write(photoarray, 0, photoarray.Length);

System.Drawing.Image image = System.Drawing.Image.FromStream(ms);

return image;

}

//Hinh ảnh

private void gridview\_CellClick(object sender, DataGridViewCellEventArgs e)

{

if (e.RowIndex >= 0)

{

DataGridViewRow row = this.gridview.Rows[e.RowIndex];

txtMaNV.Text = row.Cells[1].Value.ToString().Trim();

txtTenNV.Text = row.Cells[2].Value.ToString().Trim();

txtMatKhau.Text = row.Cells[3].Value.ToString().Trim();

txtDienThoai.Text = row.Cells[4].Value.ToString().Trim();

txtDiaChi.Text = row.Cells[5].Value.ToString().Trim();

DataTable dtimg = new DataTable();

dtimg = dal.HinhAnhNhanVien(txtMaNV.Text.Trim());

if (dtimg.Rows.Count > 0)

{

if (dtimg.Rows[0]["HinhAnh"].ToString() != "")

{

pchinhAnh.Image = ConvertStringtoImage(dtimg.Rows[0]["HinhAnh"].ToString());

}

else

{

dtHinhAnh = "";

pchinhAnh.Image = null;

}

}

}

}

private Boolean choseImage = false;

public static System.Drawing.Image img2;

private string dtHinhAnh;

private void btnChonHinh\_LinkClicked(object sender, LinkLabelLinkClickedEventArgs e)

{

OpenFileDialog OpenFile = new OpenFileDialog();

string filename;

OpenFile.Multiselect = false;

OpenFile.Filter = "Images (\*.png, \*.gif, \*.tif, \*.tiff, \*.bmp, \*.jpg, \*.jpeg, \*.jpe, \*.jfif)|\*.png;\*.gif;\*.tif;\*.tiff;\*.bmp;\*.jpg;\*.jpeg;\*.jpe;\*.jfif";

if (OpenFile.ShowDialog() == System.Windows.Forms.DialogResult.OK)

{

filename = OpenFile.FileName;

OpenFile.Dispose();

if (filename != "")

{

choseImage = true;

System.Drawing.Image img;

try

{

img = System.Drawing.Image.FromFile(filename);

if (img != null)

{

pchinhAnh.Image = img;

img2 = img;

ImageConverter converter = new ImageConverter();

var bytes = (byte[])converter.ConvertTo((Bitmap)pchinhAnh.Image, typeof(byte[]));

dtHinhAnh = Convert.ToBase64String(bytes);

}

}

catch (Exception)

{

throw;

}

}

}

}

private void btnXoaHinh\_LinkClicked(object sender, LinkLabelLinkClickedEventArgs e)

{

choseImage = true;

pchinhAnh.Image = null;

img2 = null;

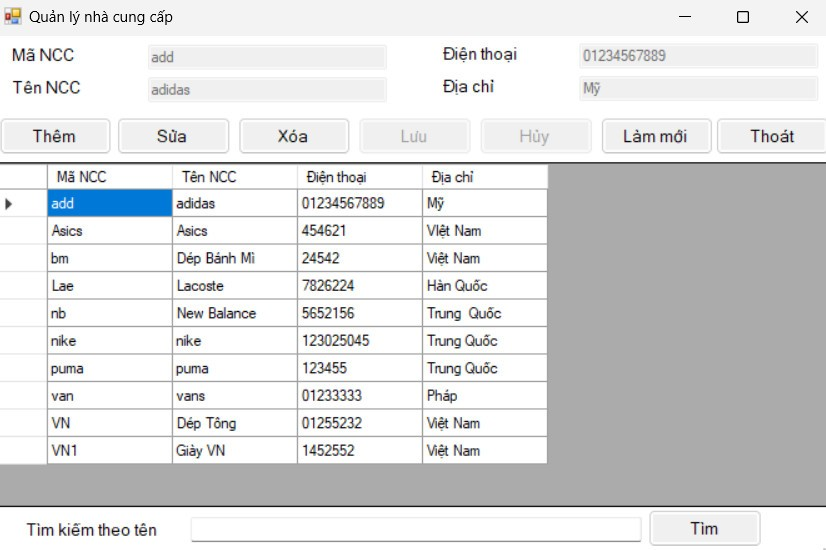
dtHinhAnh = "";

}

}

}

* + - 1. **Giao diện form quản lý nhà cung cấp**



Hình 3.5 Giao diện Form quản lý nhà cung cấp

* **Thiết kế**

btnThem\_Click: Button Thêm nhà cung cấp.

btnSua\_Click: Button sửa thông tin nhà cung cấp .

btnXoa\_Click: Button xóa thông tin nhà cung cấp.

btnLuu\_Click: Button lưu ( Sau khi thêm hoặc sửa thông tin xong bấm nút Lưu để lưu thông tin nhà cung cấp) .

btnHuy\_Click: Button Hủy quá trình lưu hoặc sửa thông tin nhà cung cấp.

btnLamMoi\_Click: Button làm mới khi nhập thông tin nhà cung cấp.

btnThoat\_Click: Button thoát sẽ thoát khỏi form quản lý nhà cung cấp.

btnTim\_Click: Button Tìm kiếm theo tên giúp tìm kiếm thông tin nhà cung cấp theo tên dễ dàng.

* **View code giao diện quản lý nhà cung cấp**

namespace PhanMemQuanLyBanGiayDep.Views

{

public partial class frmNCC : Form

{

NCCControllers dal = new NCCControllers();

private bool luu;

public frmNCC()

{

InitializeComponent();

}

private void frmNCC\_Load(object sender, EventArgs e)

{

HienThiNCC("");

boolcontrols(true);

if (gridview.Rows.Count == 0)

{

return;

}

var row = this.gridview.Rows[0];

txtMaNCC.Text = row.Cells[0].Value.ToString();

txtTenNCC.Text = row.Cells[1].Value.ToString();

txtDienThoai.Text = row.Cells[2].Value.ToString();

txtDiaChi.Text = row.Cells[3].Value.ToString();

}

private void HienThiNCC(string TenNV)

{

gridview.DataSource = dal.HienThi(TenNV);

gridview.Columns[0].HeaderText = "Mã NCC";

gridview.Columns[1].HeaderText = "Tên NCC";

gridview.Columns[2].HeaderText = "Điện thoại";

gridview.Columns[3].HeaderText = "Địa chỉ";

}

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

{

if (gridview.Rows.Count == 0)

{

return;

}

luu = false;

txtMaNCC.Enabled = false;

boolcontrols(false);

txtMaNCC.Enabled = false;

}

private void boolcontrols(bool iss)

{

btnThem.Enabled = iss;

btnSua.Enabled = iss;

btnXoa.Enabled = iss;

btnLuu.Enabled = !iss;

btnHuy.Enabled = !iss;

btnLamMoi.Enabled = iss;

btnThoat.Enabled = iss;

btnTim.Enabled = iss;

txtMaNCC.Enabled = !iss;

txtTenNCC.Enabled = !iss;

txtDiaChi.Enabled = !iss;

txtDienThoai.Enabled = !iss;

}

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

{

if (gridview.Rows.Count == 0)

{

return;

}

DialogResult dr = MessageBox.Show("Có chắc chắn xóa NCC này không ?", "Xác nhận", MessageBoxButtons.YesNo, MessageBoxIcon.Question);

if (dr == DialogResult.Yes)

{

dal.Xoa(gridview.Rows[gridview.CurrentCell.RowIndex].Cells[0].Value.ToString());

MessageBox.Show("Xóa thành công ", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Information, MessageBoxDefaultButton.Button1);

HienThiNCC("");

boolcontrols(true);

}

else

return;

}

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

{

if (txtMaNCC.Text == "")

{

MessageBox.Show("Mã NCC không được trống", "Thông báo",

MessageBoxButtons.OK, MessageBoxIcon.Error);

txtMaNCC.Focus();

return;

}

if (txtTenNCC.Text == "")

{

MessageBox.Show("Tên NCC không được trống", "Thông báo",

MessageBoxButtons.OK, MessageBoxIcon.Error);

txtTenNCC.Focus();

return;

}

if (luu == true)

{

try

{

dal.Them(txtMaNCC.Text.Trim(), txtTenNCC.Text.Trim(), txtDienThoai.Text.Trim(), txtDiaChi.Text.Trim());

MessageBox.Show("Thêm thành công.");

HienThiNCC("");

boolcontrols(true);

}

catch (Exception)

{

MessageBox.Show("Mã NCC đã tồn tại, vui lòng tạo mã khác.", "Thông báo",

MessageBoxButtons.OK, MessageBoxIcon.Error);

txtMaNCC.Focus();

return;

}

}

else

{

try

{

dal.Sua(txtMaNCC.Text.Trim(), txtTenNCC.Text.Trim(), txtDienThoai.Text.Trim(), txtDiaChi.Text.Trim());

MessageBox.Show("Sửa thành công.");

HienThiNCC("");

boolcontrols(true);

}

catch (Exception)

{

MessageBox.Show("Mã NCC đã tồn tại, vui lòng tạo mã khác.", "Thông báo",

MessageBoxButtons.OK, MessageBoxIcon.Error);

txtMaNCC.Focus();

return;

}

}

}

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

{

HienThiNCC("");

boolcontrols(true);

if (gridview.Rows.Count == 0)

{

return;

}

var row = this.gridview.Rows[0];

txtMaNCC.Text = row.Cells[0].Value.ToString();

txtTenNCC.Text = row.Cells[1].Value.ToString();

txtDienThoai.Text = row.Cells[2].Value.ToString();

txtDiaChi.Text = row.Cells[3].Value.ToString();

}

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

{

HienThiNCC("");

boolcontrols(true);

if (gridview.Rows.Count == 0)

{

return;

}

var row = this.gridview.Rows[0];

txtMaNCC.Text = row.Cells[0].Value.ToString();

txtTenNCC.Text = row.Cells[1].Value.ToString();

txtDienThoai.Text = row.Cells[2].Value.ToString();

txtDiaChi.Text = row.Cells[3].Value.ToString();

}

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

{

this.Close();

}

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

{

HienThiNCC(txtTenSearch.Text);

}

private void gridview\_CellClick(object sender, DataGridViewCellEventArgs e)

{

if (e.RowIndex >= 0)

{

DataGridViewRow row = this.gridview.Rows[e.RowIndex];

txtMaNCC.Text = row.Cells[0].Value.ToString();

txtTenNCC.Text = row.Cells[1].Value.ToString();

txtDienThoai.Text = row.Cells[2].Value.ToString();

txtDiaChi.Text = row.Cells[3].Value.ToString();

}

}

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

{

txtMaNCC.Text = "";

txtTenNCC.Text = "";

txtDienThoai.Text = "";

txtDiaChi.Text = "";

boolcontrols(false);

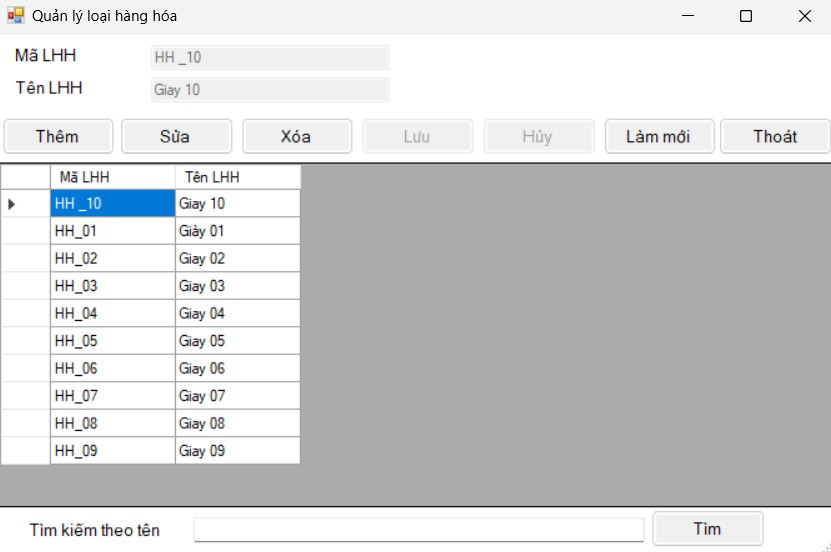
luu = true;

}

}

}

* + - 1. **Giao diện form quản lý loại loại hàng hóa**

****

Hình 3.6 Giao diện Form quản lý loại hàng hóa

* **Thiết kế**

btnThem\_Click: Button Thêm loại hàng hóa.

btnSua\_Click: Button sửa thông tin loại hàng hóa.

btnXoa\_Click: Button xóa thông tin loại hàng hóa.

btnLuu\_Click: Button lưu ( Sau khi thêm hoặc sửa thông tin xong bấm nút Lưu để lưu thông tin loại hàng hóa) .

btnHuy\_Click: Button Hủy quá trình lưu hoặc sửa thông tin loại hàng hóa.

btnLamMoi\_Click: Button làm mới khi nhập thông tin loại hàng hóa.

btnThoat\_Click: Button thoát sẽ thoát khỏi form quản lý loại hàng hóa.

btnTim\_Click: Button Tìm kiếm theo tên giúp tìm kiếm thông tin loại hàng hóa theo tên dễ dàng.

* **View code giao diện quản lý loại hàng hóa**

namespace PhanMemQuanLyBanGiayDep.Views

{

public partial class frmLHH : Form

{

LHHControllers dal = new LHHControllers();

private bool luu;

public frmLHH()

{

InitializeComponent();

}

private void HienThiLHH(string TenNV)

{

gridview.DataSource = dal.HienThi(TenNV);

gridview.Columns[0].HeaderText = "Mã LHH";

gridview.Columns[1].HeaderText = "Tên LHH";

}

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

{

txtMaLHH.Text = "";

txtTenLHH.Text = "";

boolcontrols(false);

luu = true;

}

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

{

if (gridview.Rows.Count == 0)

{

return;

}

luu = false;

txtMaLHH.Enabled = false;

boolcontrols(false);

txtMaLHH.Enabled = false;

}

private void boolcontrols(bool iss)

{

btnThem.Enabled = iss;

btnSua.Enabled = iss;

btnXoa.Enabled = iss;

btnLuu.Enabled = !iss;

btnHuy.Enabled = !iss;

btnLamMoi.Enabled = iss;

btnThoat.Enabled = iss;

btnTim.Enabled = iss;

txtMaLHH.Enabled = !iss;

txtTenLHH.Enabled = !iss;

}

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

{

if (gridview.Rows.Count == 0)

{

return;

}

DialogResult dr = MessageBox.Show("Có chắc chắn xóa LHH này không ?", "Xác nhận", MessageBoxButtons.YesNo, MessageBoxIcon.Question);

if (dr == DialogResult.Yes)

{

dal.Xoa(gridview.Rows[gridview.CurrentCell.RowIndex].Cells[0].Value.ToString());

MessageBox.Show("Xóa thành công ", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Information, MessageBoxDefaultButton.Button1);

HienThiLHH("");

boolcontrols(true);

}

else

return;

}

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

{

if (txtMaLHH.Text == "")

{

MessageBox.Show("Mã LHH không được trống", "Thông báo",

MessageBoxButtons.OK, MessageBoxIcon.Error);

txtMaLHH.Focus();

return;

}

if (txtTenLHH.Text == "")

{

MessageBox.Show("Tên LHH không được trống", "Thông báo",

MessageBoxButtons.OK, MessageBoxIcon.Error);

txtTenLHH.Focus();

return;

}

if (luu == true)

{

try

{

dal.Them(txtMaLHH.Text.Trim(), txtTenLHH.Text.Trim());

MessageBox.Show("Thêm thành công.");

HienThiLHH("");

boolcontrols(true);

}

catch (Exception)

{

MessageBox.Show("Mã LHH đã tồn tại, vui lòng tạo mã khác.", "Thông báo",

MessageBoxButtons.OK, MessageBoxIcon.Error);

txtMaLHH.Focus();

return;

}

}

else

{

try

{

dal.Sua(txtMaLHH.Text.Trim(), txtTenLHH.Text.Trim());

MessageBox.Show("Sửa thành công.");

HienThiLHH("");

boolcontrols(true);

}

catch (Exception)

{

MessageBox.Show("Mã LHH đã tồn tại, vui lòng tạo mã khác.", "Thông báo",

MessageBoxButtons.OK, MessageBoxIcon.Error);

txtMaLHH.Focus();

return;

}

}

}

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

{

HienThiLHH("");

boolcontrols(true);

if (gridview.Rows.Count == 0)

{

return;

}

var row = this.gridview.Rows[0];

txtMaLHH.Text = row.Cells[0].Value.ToString();

txtTenLHH.Text = row.Cells[1].Value.ToString();

}

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

{

HienThiLHH("");

boolcontrols(true);

if (gridview.Rows.Count == 0)

{

return;

}

var row = this.gridview.Rows[0];

txtMaLHH.Text = row.Cells[0].Value.ToString();

txtTenLHH.Text = row.Cells[1].Value.ToString();

}

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

{

this.Close();

}

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

{

HienThiLHH(txtTenSearch.Text);

}

private void gridview\_CellClick(object sender, DataGridViewCellEventArgs e)

{

if (e.RowIndex >= 0)

{

DataGridViewRow row = this.gridview.Rows[e.RowIndex];

txtMaLHH.Text = row.Cells[0].Value.ToString();

txtTenLHH.Text = row.Cells[1].Value.ToString();

}

}

private void frmLHH\_Load(object sender, EventArgs e)

{

HienThiLHH("");

boolcontrols(true);

if (gridview.Rows.Count == 0)

{

return;

}

var row = this.gridview.Rows[0];

txtMaLHH.Text = row.Cells[0].Value.ToString();

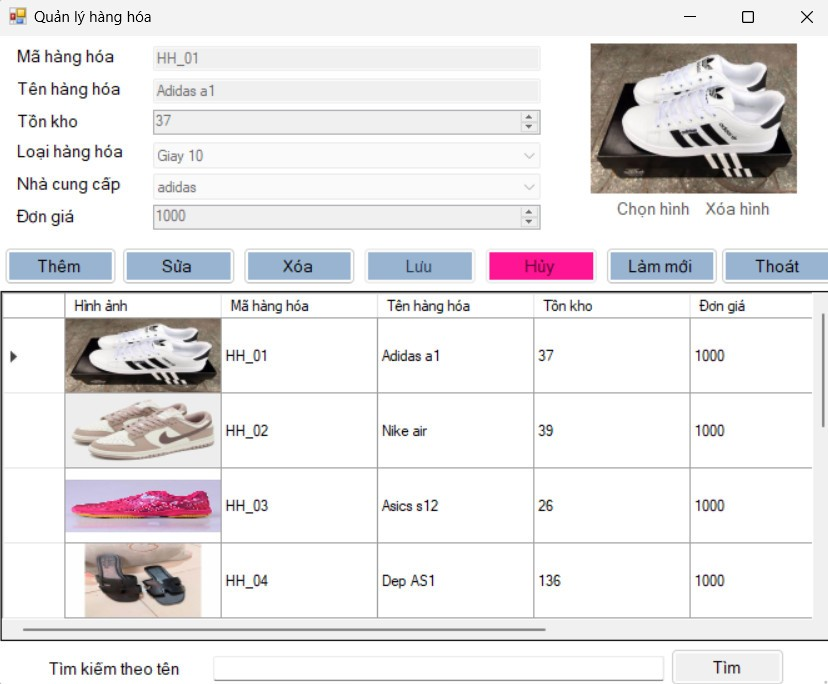
txtTenLHH.Text = row.Cells[1].Value.ToString();

}

}

}

* + - 1. **Giao diện form quản lý hàng hóa**



Hình 3.7 Giao diện Form quản lý hàng hóa

* **Thiết kế**

btnThem\_Click: Button Thêm hàng hóa.

btnSua\_Click: Button sửa thông tin hàng hóa.

btnXoa\_Click: Button xóa thông tin hàng hóa.

btnLuu\_Click: Button lưu ( Sau khi thêm hoặc sửa thông tin xong bấm nút Lưu để lưu thông tin hàng hóa) .

btnHuy\_Click: Button Hủy quá trình lưu hoặc sửa thông tin hàng hóa.

btnLamMoi\_Click: Button làm mới khi nhập thông tin hàng hóa.

btnThoat\_Click: Button thoát sẽ thoát khỏi form quản lý hàng hóa.

btnTim\_Click: Button Tìm kiếm theo tên giúp tìm kiếm thông tin hàng hóa theo tên.

* **View code giao diện quản lý hàng hóa**

namespace PhanMemQuanLyBanGiayDep.Views

{

public partial class frmHangHoa : Form

{

public frmHangHoa()

{

InitializeComponent();

}

HangHoaControllers db = new HangHoaControllers();

NCCControllers dbNCC = new NCCControllers();

LHHControllers dbLHH = new LHHControllers();

private bool luu;

private void frmHangHoa\_Load(object sender, EventArgs e)

{

HienThiHangHoa();

boolcontrols(true);

DanhSachLHH();

DanhSachNCC();

}

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

{

DanhSachLHH();

DanhSachNCC();

txtMaHH.Text = "";

txtTenHH.Text = "";

txtTonKho.Value = 1;

txtDonGia.Value = 1;

boolcontrols(false);

luu = true;

}

private void boolcontrols(bool iss)

{

btnThem.Enabled = iss;

btnSua.Enabled = iss;

btnXoa.Enabled = iss;

btnChonHinh.Enabled = !iss;

btnXoaHinh.Enabled = !iss;

btnLuu.Enabled = !iss;

btnHuy.Enabled = !iss;

btnLamMoi.Enabled = iss;

btnThoat.Enabled = iss;

btnTim.Enabled = iss;

txtMaHH.Enabled = !iss;

txtTenHH.Enabled = !iss;

txtTonKho.Enabled = !iss;

txtDonGia.Enabled = !iss;

cboMaNCC.Enabled = !iss;

cboMaLHH.Enabled = !iss;

}

//Hiển thị hàng hóa

private void HienThiHangHoa()

{

DataTable dt = new DataTable();

dt = db.HienThi(txtTenSearch.Text.Trim());

gridview.Rows.Clear();

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

{

string MaHH = dt.Rows[i]["MaHH"].ToString();

string TenHH = dt.Rows[i]["TenHH"].ToString();

string TonKho = dt.Rows[i]["TonKho"].ToString();

string DonGia = dt.Rows[i]["DonGia"].ToString();

string MaNCC = dt.Rows[i]["MaNCC"].ToString();

string MaLHH = dt.Rows[i]["MaLHH"].ToString();

string v = dt.Rows[i]["hinhAnh"].ToString();

if (v != "")

{

System.Drawing.Image img = ConvertStringtoImage(dt.Rows[i]["hinhAnh"].ToString());

gridview.Rows.Add(new object[] { img, MaHH, TenHH, TonKho, DonGia, MaNCC, MaLHH });

}

else

{

gridview.Rows.Add(new object[] { null, MaHH, TenHH, TonKho, DonGia, MaNCC, MaLHH });

foreach (var column in gridview.Columns)

{

if (column is DataGridViewImageColumn)

(column as DataGridViewImageColumn).DefaultCellStyle.NullValue = null;

}

}

}

foreach (DataGridViewRow r in gridview.Rows)

r.Height = 60;

if (gridview.Rows.Count == 0)

{

txtMaHH.Text = "";

txtTenHH.Text = "";

txtTonKho.Text = "";

txtDonGia.Text = "";

pchinhAnh.Image = null;

}

else

{

var row = this.gridview.Rows[0];

txtMaHH.Text = row.Cells[1].Value.ToString().Trim();

txtTenHH.Text = row.Cells[2].Value.ToString().Trim();

txtTonKho.Text = row.Cells[3].Value.ToString().Trim();

txtDonGia.Text = row.Cells[4].Value.ToString().Trim();

cboMaNCC.SelectedValue = row.Cells[5].Value.ToString().Trim();

cboMaLHH.SelectedValue = row.Cells[6].Value.ToString().Trim();

DataTable dtimg = new DataTable();

dtimg = db.HinhAnhHangHoa(txtMaHH.Text.Trim());

if (dtimg.Rows.Count > 0)

{

if (dtimg.Rows[0]["HinhAnh"].ToString() != "")

{

pchinhAnh.Image = ConvertStringtoImage(dtimg.Rows[0]["HinhAnh"].ToString());

}

else

{

dtHinhAnh = "";

pchinhAnh.Image = null;

}

}

}

}

public System.Drawing.Image ConvertStringtoImage(string commands)

{

byte[] photoarray = Convert.FromBase64String(commands);

MemoryStream ms = new MemoryStream(photoarray, 0, photoarray.Length);

ms.Write(photoarray, 0, photoarray.Length);

System.Drawing.Image image = System.Drawing.Image.FromStream(ms);

return image;

}

private void DanhSachLHH()

{

DataTable dt = dbLHH.HienThi("");

cboMaLHH.DataSource = dt;

cboMaLHH.DisplayMember = "TenLHH";

cboMaLHH.ValueMember = "MaLHH";

}

private void DanhSachNCC()

{

DataTable dt = dbNCC.HienThi("");

cboMaNCC.DataSource = dt;

cboMaNCC.DisplayMember = "TenNCC";

cboMaNCC.ValueMember = "MaNCC";

}

private void gridview\_CellClick(object sender, DataGridViewCellEventArgs e)

{

if (e.RowIndex >= 0)

{

DataGridViewRow row = this.gridview.Rows[e.RowIndex];

txtMaHH.Text = row.Cells[1].Value.ToString().Trim();

txtTenHH.Text = row.Cells[2].Value.ToString().Trim();

txtTonKho.Text = row.Cells[3].Value.ToString().Trim();

txtDonGia.Text = row.Cells[4].Value.ToString().Trim();

cboMaNCC.SelectedValue = row.Cells[5].Value.ToString().Trim();

cboMaLHH.SelectedValue = row.Cells[6].Value.ToString().Trim();

DataTable dtimg = new DataTable();

dtimg = db.HinhAnhHangHoa(txtMaHH.Text.Trim());

if (dtimg.Rows.Count > 0)

{

if (dtimg.Rows[0]["HinhAnh"].ToString() != "")

{

pchinhAnh.Image = ConvertStringtoImage(dtimg.Rows[0]["HinhAnh"].ToString());

}

else

{

dtHinhAnh = "";

pchinhAnh.Image = null;

}

}

}

}

private Boolean choseImage = false;

public static System.Drawing.Image img2;

private string dtHinhAnh;

//Chọn và xóa hình ảnh

private void btnChonHinh\_LinkClicked(object sender, LinkLabelLinkClickedEventArgs e)

{

OpenFileDialog OpenFile = new OpenFileDialog();

string filename;

OpenFile.Multiselect = false;

OpenFile.Filter = "Images (\*.png, \*.gif, \*.tif, \*.tiff, \*.bmp, \*.jpg, \*.jpeg, \*.jpe, \*.jfif)|\*.png;\*.gif;\*.tif;\*.tiff;\*.bmp;\*.jpg;\*.jpeg;\*.jpe;\*.jfif";

if (OpenFile.ShowDialog() == System.Windows.Forms.DialogResult.OK)

{

filename = OpenFile.FileName;

OpenFile.Dispose();

if (filename != "")

{

choseImage = true;

System.Drawing.Image img;

try

{

img = System.Drawing.Image.FromFile(filename);

if (img != null)

{

pchinhAnh.Image = img;

img2 = img;

ImageConverter converter = new ImageConverter();

var bytes = (byte[])converter.ConvertTo((Bitmap)pchinhAnh.Image, typeof(byte[]));

dtHinhAnh = Convert.ToBase64String(bytes);

}

}

catch (Exception)

{

throw;

}

}

}

}

private void btnXoaHinh\_LinkClicked(object sender, LinkLabelLinkClickedEventArgs e)

{

choseImage = true;

pchinhAnh.Image = null;

img2 = null;

dtHinhAnh = "";

}

private void txtDonGia\_ValueChanged(object sender, EventArgs e)

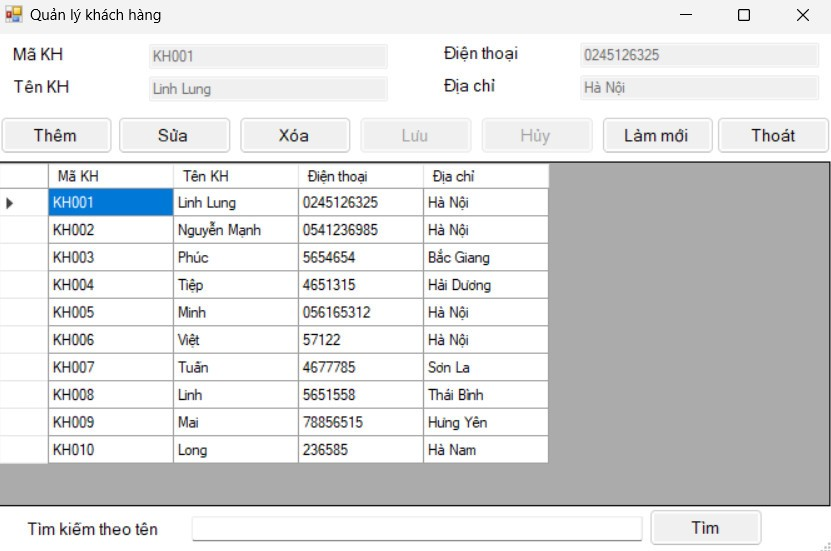
{

}

}

}

* + - 1. **Giao diện Form quản lý khách hàng**



Hình 3.8 Giao diện Form quản lý khách hàng

* **Thiết kế**

btnThem\_Click: Button Thêm khách hàng.

btnSua\_Click: Button sửa thông tin khách hàng.

btnXoa\_Click: Button xóa thông tin khách hàng.

btnLuu\_Click: Button lưu ( Sau khi thêm hoặc sửa thông tin xong bấm nút Lưu để lưu thông tin khách hàng) .

btnHuy\_Click: Button Hủy quá trình lưu hoặc sửa thông tin khách hàng.

btnLamMoi\_Click: Button làm mới khi nhập thông tin khách hàng.

btnThoat\_Click: Button thoát sẽ thoát khỏi form quản lý khách hàng.

btnTim\_Click: Button Tìm kiếm theo tên giúp tìm kiếm thông tin khách hàng theo tên.

* **View code giao diện quản lý khách hàng**

namespace PhanMemQuanLyBanGiayDep.Views

{

public partial class frmKhachHang : Form

{

KhachHangControllers dal = new KhachHangControllers();

private bool luu;

public frmKhachHang()

{

InitializeComponent();

}

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

{

txtMaKH.Text = "";

txtTenKH.Text = "";

txtDienThoai.Text = "";

txtDiaChi.Text = "";

boolcontrols(false);

luu = true;

}

private void HienThiKH(string TenNV)

{

gridview.DataSource = dal.HienThi(TenNV);

gridview.Columns[0].HeaderText = "Mã KH";

gridview.Columns[1].HeaderText = "Tên KH";

gridview.Columns[2].HeaderText = "Điện thoại";

gridview.Columns[3].HeaderText = "Địa chỉ";

}

private void frmKhachHang\_Load(object sender, EventArgs e)

{

HienThiKH("");

boolcontrols(true);

if (gridview.Rows.Count == 0)

{

return;

}

var row = this.gridview.Rows[0];

txtMaKH.Text = row.Cells[0].Value.ToString();

txtTenKH.Text = row.Cells[1].Value.ToString();

txtDienThoai.Text = row.Cells[2].Value.ToString();

txtDiaChi.Text = row.Cells[3].Value.ToString();

}

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

{

if (gridview.Rows.Count == 0)

{

return;

}

luu = false;

txtMaKH.Enabled = false;

boolcontrols(false);

txtMaKH.Enabled = false;

}

private void boolcontrols(bool iss)

{

btnThem.Enabled = iss;

btnSua.Enabled = iss;

btnXoa.Enabled = iss;

btnLuu.Enabled = !iss;

btnHuy.Enabled = !iss;

btnLamMoi.Enabled = iss;

btnThoat.Enabled = iss;

btnTim.Enabled = iss;

txtMaKH.Enabled = !iss;

txtTenKH.Enabled = !iss;

txtDiaChi.Enabled = !iss;

txtDienThoai.Enabled = !iss;

}

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

{

if (gridview.Rows.Count == 0)

{

return;

}

DialogResult dr = MessageBox.Show("Có chắc chắn xóa KH này không ?", "Xác nhận", MessageBoxButtons.YesNo, MessageBoxIcon.Question);

if (dr == DialogResult.Yes)

{

dal.Xoa(gridview.Rows[gridview.CurrentCell.RowIndex].Cells[0].Value.ToString());

MessageBox.Show("Xóa thành công ", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Information, MessageBoxDefaultButton.Button1);

HienThiKH("");

boolcontrols(true);

}

else

return;

}

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

{

if (txtMaKH.Text == "")

{

MessageBox.Show("Mã KH không được trống", "Thông báo",

MessageBoxButtons.OK, MessageBoxIcon.Error);

txtMaKH.Focus();

return;

}

if (txtTenKH.Text == "")

{

MessageBox.Show("Tên KH không được trống", "Thông báo",

MessageBoxButtons.OK, MessageBoxIcon.Error);

txtTenKH.Focus();

return;

}

if (luu == true)

{

try

{

dal.Them(txtMaKH.Text.Trim(), txtTenKH.Text.Trim(), txtDienThoai.Text.Trim(), txtDiaChi.Text.Trim());

MessageBox.Show("Thêm thành công.");

HienThiKH("");

boolcontrols(true);

}

catch (Exception)

{

MessageBox.Show("Mã KH đã tồn tại, vui lòng tạo mã khác.", "Thông báo",

MessageBoxButtons.OK, MessageBoxIcon.Error);

txtMaKH.Focus();

return;

}

}

else

{

try

{

dal.Sua(txtMaKH.Text.Trim(), txtTenKH.Text.Trim(), txtDienThoai.Text.Trim(), txtDiaChi.Text.Trim());

MessageBox.Show("Sửa thành công.");

HienThiKH("");

boolcontrols(true);

}

catch (Exception)

{

MessageBox.Show("Mã KH đã tồn tại, vui lòng tạo mã khác.", "Thông báo",

MessageBoxButtons.OK, MessageBoxIcon.Error);

txtMaKH.Focus();

return;

}

}

}

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

{

HienThiKH("");

boolcontrols(true);

if (gridview.Rows.Count == 0)

{

return;

}

var row = this.gridview.Rows[0];

txtMaKH.Text = row.Cells[0].Value.ToString();

txtTenKH.Text = row.Cells[1].Value.ToString();

txtDienThoai.Text = row.Cells[2].Value.ToString();

txtDiaChi.Text = row.Cells[3].Value.ToString();

}

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

{

HienThiKH("");

boolcontrols(true);

if (gridview.Rows.Count == 0)

{

return;

}

var row = this.gridview.Rows[0];

txtMaKH.Text = row.Cells[0].Value.ToString();

txtTenKH.Text = row.Cells[1].Value.ToString();

txtDienThoai.Text = row.Cells[2].Value.ToString();

txtDiaChi.Text = row.Cells[3].Value.ToString();

}

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

{

this.Close();

}

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

{

HienThiKH(txtTenSearch.Text);

}

private void gridview\_CellClick(object sender, DataGridViewCellEventArgs e)

{

if (e.RowIndex >= 0)

{

DataGridViewRow row = this.gridview.Rows[e.RowIndex];

txtMaKH.Text = row.Cells[0].Value.ToString();

txtTenKH.Text = row.Cells[1].Value.ToString();

txtDienThoai.Text = row.Cells[2].Value.ToString();

txtDiaChi.Text = row.Cells[3].Value.ToString();

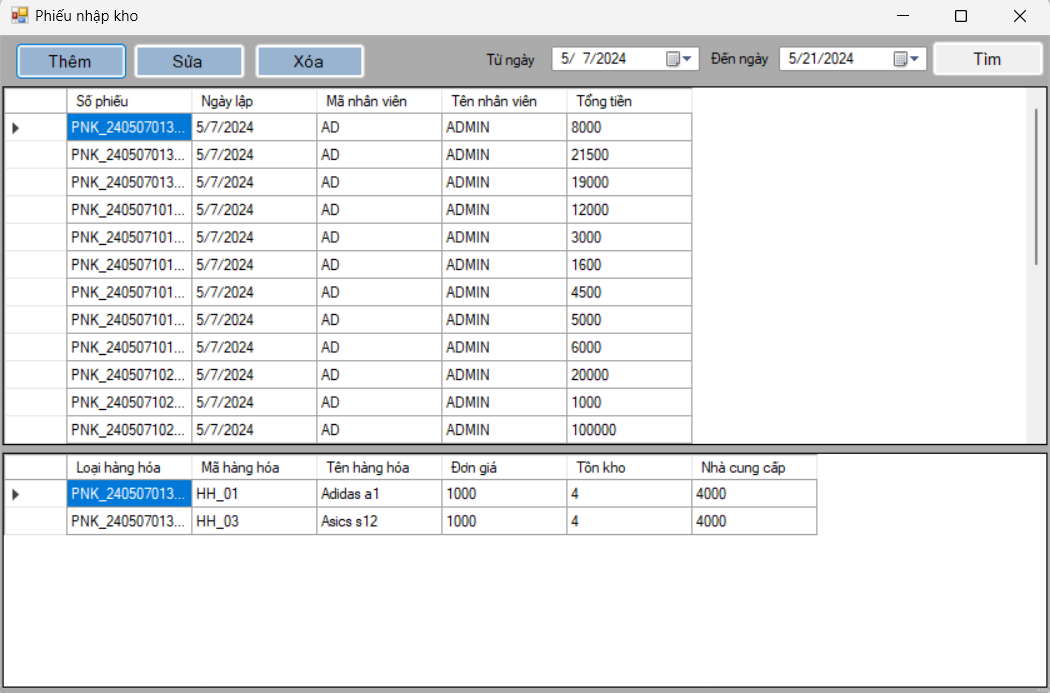
}

}

}

}

* + - 1. **Giao diện Form Phiếu nhập kho**



Hình 3.9 Giao diện Form Phiếu nhập kho

* **Thiết kế**

btnThem\_Click: Button Thêm phiếu nhập kho.

btnSua\_Click: Button sửa thông tin phiếu nhập kho.

btnXoa\_Click: Button xóa thông tin phiếu nhập kho.

btnTim\_Click: Button Tìm kiếm theo tên giúp tìm kiếm thông tin phiếu nhập kho theo ngày tháng năm.

* **View code giao diện phiếu nhập kho**

namespace PhanMemQuanLyBanGiayDep.Views

{

public partial class frmNhapKho : Form

{

public frmNhapKho()

{

InitializeComponent();

}

NhapKhoControllers db = new NhapKhoControllers();

DataTable dt;

DataTable dt2;

public static Boolean save;

public static string MaHD;

public static DateTime NgayHD;

public static string MaTK\_NV;

public static string TongTien;

private void frmNhapKho\_Load(object sender, EventArgs e)

{

HienThiNhapKho();

if (dtGVNhapKho.Rows.Count > 0)

{

var row = this.dtGVNhapKho.Rows[0];

HienThiChiTietNhapKho(row.Cells["MaHD"].Value.ToString());

}

else

{

HienThiChiTietNhapKho("");

}

}

private void HienThiNhapKho()

{

dt = db.HienThiNhapKho(dtNgayHD1.Value.ToString("yyyyMMdd"), dtNgayHD2.Value.ToString("yyyyMMdd"));

dtGVNhapKho.DataSource = dt;

dtGVNhapKho.Columns[0].HeaderText = "Số phiếu";

dtGVNhapKho.Columns[1].HeaderText = "Ngày lập";

dtGVNhapKho.Columns[2].HeaderText = "Mã nhân viên";

dtGVNhapKho.Columns[3].HeaderText = "Tên nhân viên";

dtGVNhapKho.Columns[4].HeaderText = "Tổng tiền";

}

private void HienThiChiTietNhapKho(string value)

{

dt2 = db.HienThiChiTietNhapKho(value);

dtGVChiTietNhapKho.DataSource = dt2;

dtGVChiTietNhapKho.Columns[0].HeaderText = "Loại hàng hóa";

dtGVChiTietNhapKho.Columns[1].HeaderText = "Mã hàng hóa";

dtGVChiTietNhapKho.Columns[2].HeaderText = "Tên hàng hóa";

dtGVChiTietNhapKho.Columns[3].HeaderText = "Đơn giá";

dtGVChiTietNhapKho.Columns[4].HeaderText = "Tồn kho";

dtGVChiTietNhapKho.Columns[5].HeaderText = "Nhà cung cấp";

}

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

{

save = true;

frmNhapKhoInfor frm = new frmNhapKhoInfor();

frm.Text = "Thêm mới";

frm.ShowDialog();

HienThiNhapKho();

if (dtGVNhapKho.Rows.Count > 0)

{

var row = this.dtGVNhapKho.Rows[0];

HienThiChiTietNhapKho(row.Cells["MaHD"].Value.ToString());

}

else

{

HienThiChiTietNhapKho("");

}

}

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

{

if (dtGVNhapKho.Rows.Count == 0)

{

return;

}

DataGridViewRow row = this.dtGVNhapKho.Rows[dtGVNhapKho.CurrentCell.RowIndex];

save = false;

frmNhapKhoInfor frm = new frmNhapKhoInfor();

MaHD = row.Cells[0].Value.ToString();

NgayHD = DateTime.Parse(row.Cells[1].Value.ToString());

MaTK\_NV = row.Cells[2].Value.ToString();

TongTien = row.Cells[3].Value.ToString();

frm.Text = "Sửa phiếu";

frm.ShowDialog();

HienThiNhapKho();

if (dtGVNhapKho.Rows.Count > 0)

{

var row2 = this.dtGVNhapKho.Rows[0];

HienThiChiTietNhapKho(row2.Cells["MaHD"].Value.ToString());

}

else

{

HienThiChiTietNhapKho("");

}

}

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

{

if (dtGVNhapKho.Rows.Count == 0)

{

return;

}

DialogResult dr = MessageBox.Show("Có chắc chắn xóa dòng dữ liệu này không ?", "Xác nhận", MessageBoxButtons.YesNo, MessageBoxIcon.Question);

if (dr == DialogResult.Yes)

{

string del = "";

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

{

del = del + " UPDATE HangHoa SET TonKho = TonKho - " + dtGVChiTietNhapKho.Rows[i].Cells["SoLuong"].Value.ToString() + " WHERE MaHH = '"+ dtGVChiTietNhapKho.Rows[i].Cells["MaHH"].Value.ToString() + "' ; ";

}

ConnectSQL.ExecuteNonQuery(del);

db.Xoa(dtGVNhapKho.Rows[dtGVNhapKho.CurrentCell.RowIndex].Cells[0].Value.ToString());

MessageBox.Show("Xóa thành công ", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Information, MessageBoxDefaultButton.Button1);

HienThiNhapKho();

if (dtGVNhapKho.Rows.Count > 0)

{

var row2 = this.dtGVNhapKho.Rows[0];

HienThiChiTietNhapKho(row2.Cells["SOPHIEU"].Value.ToString());

}

else

{

HienThiChiTietNhapKho("");

}

}

else

return;

}

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

{

HienThiNhapKho();

if (dtGVNhapKho.Rows.Count > 0)

{

var row2 = this.dtGVNhapKho.Rows[0];

HienThiChiTietNhapKho(row2.Cells["MaHD"].Value.ToString());

}

else

{

HienThiChiTietNhapKho("");

}

}

private void dtGVNhapKho\_CellClick(object sender, DataGridViewCellEventArgs e)

{

if (e.RowIndex >= 0)

{

DataGridViewRow row = this.dtGVNhapKho.Rows[e.RowIndex];

HienThiChiTietNhapKho(row.Cells[0].Value.ToString());

}

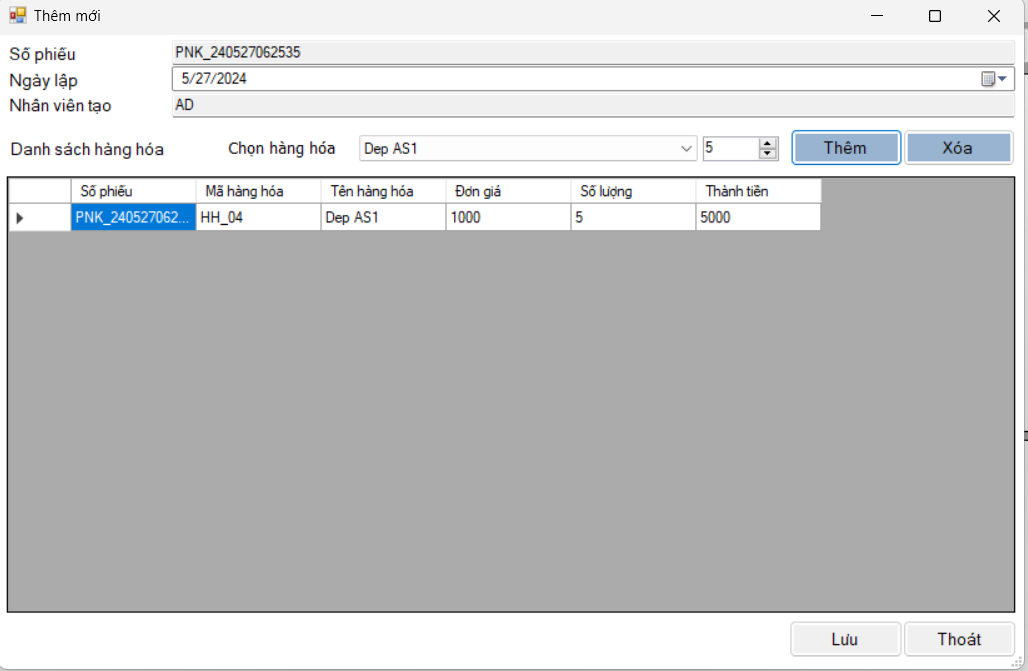
}

}

}

* + - 1. **Giao diện Form thêm mới phiếu nhập**

Khi bấm Button Thêm ở Form Phiếu nhập kho màn hìnhsẽ chuyển sang giao diện Form Thêm mới giúp thêm phiếu nhập mới.



Hình 3.10 Giao diện Form thêm mới phiếu nhập

* **Thiết kế:**

btnThem\_Click: Button Thêm mới phiếu nhập kho .

btnXoa\_Click: Button xóa thông tin phiếu nhập kho mới.

btnLuu\_Click: Button lưu ( Sau khi thêm hoặc sửa thông tin xong bấm nút Lưu để lưu phiếu nhập kho vừa tạo mới) .

btnThoat\_Click: Button thoát sẽ thoát khỏi form Thêm mới.

* **View code giao diện form thêm phiếu nhập mới**

namespace PhanMemQuanLyBanGiayDep.Views

{

public partial class frmNhapKhoInfor : Form

{

NhapKhoControllers db = new NhapKhoControllers();

HangHoaControllers dbhh = new HangHoaControllers();

DataTable dtct = new DataTable();

public Boolean ischeck = true;

public frmNhapKhoInfor()

{

InitializeComponent();

}

private void HienThiChiTietNhapKho()

{

dtct = db.HienThiChiTietNhapKho(txtMaHD.Text.Trim());

dtGVChiTietNhapKho.DataSource = dtct;

dtGVChiTietNhapKho.Columns[0].HeaderText = "Số phiếu";

dtGVChiTietNhapKho.Columns[1].HeaderText = "Mã hàng hóa";

dtGVChiTietNhapKho.Columns[2].HeaderText = "Tên hàng hóa";

dtGVChiTietNhapKho.Columns[3].HeaderText = "Đơn giá";

dtGVChiTietNhapKho.Columns[4].HeaderText = "Số lượng";

dtGVChiTietNhapKho.Columns[5].HeaderText = "Thành tiền";

}

private void frmNhapHangInfor\_Load(object sender, EventArgs e)

{

if (frmNhapKho.save == true)

{

HienThiHangHoa();

txtMaTK\_NV.Text = frmDangNhap.manv;

txtMaTK\_NV.ReadOnly = true;

txtMaHD.ReadOnly = true;

HienThiChiTietNhapKho();

GenCode();

}

else

{

HienThiHangHoa();

txtMaHD.Text = frmNhapKho.MaHD;

txtMaTK\_NV.ReadOnly = true;

HienThiChiTietNhapKho();

txtMaTK\_NV.Text = frmNhapKho.MaTK\_NV;

dtNgayHD.Value = frmNhapKho.NgayHD;

}

}

private void HienThiHangHoa()

{

DataTable dt = dbhh.HienThi("");

cboMaHH.DataSource = dt;

cboMaHH.DisplayMember = "TenHH";

cboMaHH.ValueMember = "MAHH";

}

private void GenCode()

{

string sTemp = "PNK\_" + DateTime.Now.ToString("yyMMddhhmmss");

txtMaHD.Text = sTemp;

}

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

{

if (cboMaHH.Text == "")

{

MessageBox.Show("Mã hàng hóa không được trống", "Thông báo",

MessageBoxButtons.OK, MessageBoxIcon.Error);

cboMaHH.Focus();

return;

}

if (nmSoLuong.Value == 0)

{

MessageBox.Show("Số lượng phải lớn hơn 0", "Thông báo",

MessageBoxButtons.OK, MessageBoxIcon.Error);

cboMaHH.Focus();

return;

}

ischeck = true;

if (dtGVChiTietNhapKho.Rows.Count > 0)

{

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

{

try

{

if (dtct.Rows[i]["MaHH"].ToString() == cboMaHH.SelectedValue.ToString())

{

dtct.Rows[i]["SoLuong"] = int.Parse(dtct.Rows[i]["SoLuong"].ToString()) + nmSoLuong.Value;

ischeck = false;

break;

}

}

catch (Exception)

{

throw;

}

}

}

if (ischeck == true)

{

DataTable DonGia = new DataTable();

DonGia = db.DonGiaHangHoa(cboMaHH.SelectedValue.ToString());

decimal x = decimal.Parse(DonGia.Rows[0][0].ToString()) \* nmSoLuong.Value;

DataRow toInsert = dtct.NewRow();

dtct.Rows.Add(new Object[]{

txtMaHD.Text.Trim(),

cboMaHH.SelectedValue.ToString(),

DonGia.Rows[0][1].ToString(),

DonGia.Rows[0][0].ToString(),

nmSoLuong.Value,

x});

}

dtct.AcceptChanges();

dtGVChiTietNhapKho.DataSource = dtct;

dtct.AcceptChanges();

}

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

{

dtGVChiTietNhapKho.Rows.RemoveAt(dtGVChiTietNhapKho.CurrentCell.RowIndex);

dtGVChiTietNhapKho.DataSource = dtct;

dtct.AcceptChanges();

}

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

{

if (dtGVChiTietNhapKho.Rows.Count == 0)

{

MessageBox.Show("Chưa nhập sách cần mượn", "Thông báo",

MessageBoxButtons.OK, MessageBoxIcon.Error);

return;

}

decimal c;

for (int i = 0; i < dtGVChiTietNhapKho.RowCount; i++)

{

c = db.KiemTraHangHoaTonKho(dtGVChiTietNhapKho.Rows[0].Cells[1].Value.ToString());

if (c < decimal.Parse(dtGVChiTietNhapKho.Rows[0].Cells[4].Value.ToString()))

{

MessageBox.Show("Thêm thất bại, số lượng tồn của hàng hóa là '" + dtGVChiTietNhapKho.Rows[0].Cells[1].Value.ToString() + "' còn lại là '" + c + "'", "Thông báo",

MessageBoxButtons.OK, MessageBoxIcon.Error);

return;

}

}

if (frmNhapKho.save == true)

{

db.ThemNhapKho(txtMaHD.Text.Trim(), dtNgayHD.Value.ToString("yyyy/MM/dd"), txtMaTK\_NV.Text.Trim(), 0);

db.XoaChiTietNhapKho(txtMaHD.Text.Trim());

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

{

db.ThemChiTietNhapKho(txtMaHD.Text.Trim(), dtct.Rows[i]["MaHH"].ToString(), decimal.Parse(dtct.Rows[i]["DonGia"].ToString()), decimal.Parse(dtct.Rows[i]["SOLUONG"].ToString()), decimal.Parse(dtct.Rows[i]["ThanhTien"].ToString()));

}

MessageBox.Show("Thêm thành công ", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Information, MessageBoxDefaultButton.Button1);

}

if (frmNhapKho.save == false)

{

db.XoaChiTietNhapKho(txtMaHD.Text.Trim());

db.SuaNhapKho(txtMaHD.Text.Trim(), dtNgayHD.Value.ToString("yyyy/MM/dd"), txtMaTK\_NV.Text.Trim(), 0);

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

{

db.ThemChiTietNhapKho(txtMaHD.Text.Trim(), dtct.Rows[i]["MaHH"].ToString(), decimal.Parse(dtct.Rows[i]["DonGia"].ToString()), decimal.Parse(dtct.Rows[i]["SOLUONG"].ToString()), decimal.Parse(dtct.Rows[i]["ThanhTien"].ToString()));

}

MessageBox.Show("Sửa thành công ", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Information, MessageBoxDefaultButton.Button1);

}

db.UpdateTongTien(txtMaHD.Text.Trim());

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

{

string s1 = "UPDATE HangHoa SET TonKho = TonKho + '" + decimal.Parse(dtGVChiTietNhapKho.Rows[i].Cells["SoLuong"].Value.ToString()) + "' WHERE MaHH = '" + dtGVChiTietNhapKho.Rows[i].Cells["MaHH"].Value.ToString() + "'";

ConnectSQL.ExecuteNonQuery(s1);

}

this.Close();

}

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

{

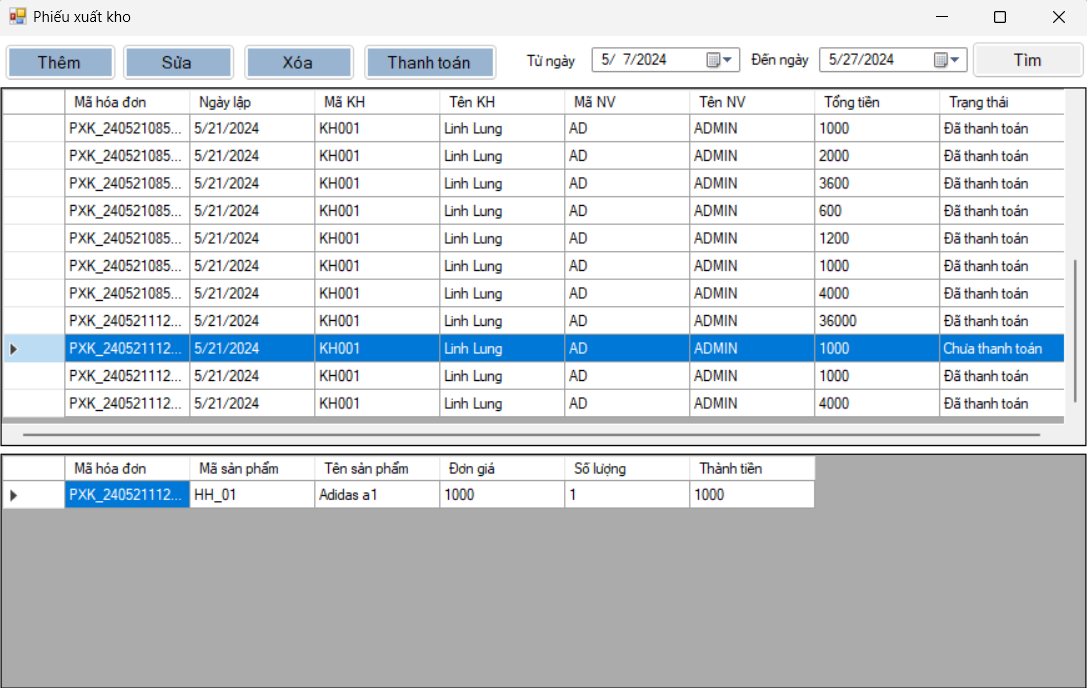
this.Close();

}

}

}

**3.2.2.10 Giao diện Form Phiếu xuất kho**

****

Hình 3.11 Form phiếu xuất kho

* **Thiết kế**

btnThem\_Click: Button Thêm phiếu xuất kho.

btnSua\_Click: Button sửa thông tin phiếu xuất kho.

btnXoa\_Click: Button xóa thông tin phiếu xuất kho.

btnTim\_Click: Button Tìm kiếm theo tên giúp tìm kiếm thông tin phiếu xuất kho theo ngày tháng năm

btnThanhToan\_Click: Button thanh toán tiền cho những đơn hàng chưa thanh toán

* **View code giao diện phiếu xuất kho**

namespace PhanMemQuanLyBanGiayDep.Views

{

public partial class frmXuatKho : Form

{

public frmXuatKho()

{

InitializeComponent();

}

XuatKhoControllers db = new XuatKhoControllers();

DataTable dt;

DataTable dt2;

public static Boolean save;

public static string mahd;

public static string makh;

public static DateTime ngayban;

public static string manv;

public static string tennv;

public static string tongtien;

public static string trangthai;

private void frmXuatKho\_Load(object sender, EventArgs e)

{

HienThiXuatKho();

if (dtGVXuatKho.Rows.Count > 0)

{

var row = this.dtGVXuatKho.Rows[0];

HienThiChiTietXuatKho(row.Cells["MaHD"].Value.ToString());

}

else

{

HienThiChiTietXuatKho("");

}

}

private void HienThiXuatKho()

{

dt = db.HienThiXuatKho(dtNgayHD1.Value.ToString("yyyyMMdd"), dtNgayHD2.Value.ToString("yyyyMMdd"));

dtGVXuatKho.DataSource = dt;

dtGVXuatKho.Columns[0].HeaderText = "Mã hóa đơn";

dtGVXuatKho.Columns[1].HeaderText = "Ngày lập";

dtGVXuatKho.Columns[2].HeaderText = "Mã KH";

dtGVXuatKho.Columns[3].HeaderText = "Tên KH";

dtGVXuatKho.Columns[4].HeaderText = "Mã NV";

dtGVXuatKho.Columns[5].HeaderText = "Tên NV";

dtGVXuatKho.Columns[6].HeaderText = "Tổng tiền";

dtGVXuatKho.Columns[7].HeaderText = "Trạng thái";

}

private void HienThiChiTietXuatKho(string value)

{

dt2 = db.HienThiChiTietXuatKho(value);

dtGVChiTietXuatKho.DataSource = dt2;

dtGVChiTietXuatKho.Columns[0].HeaderText = "Mã hóa đơn";

dtGVChiTietXuatKho.Columns[1].HeaderText = "Mã sản phẩm";

dtGVChiTietXuatKho.Columns[2].HeaderText = "Tên sản phẩm";

dtGVChiTietXuatKho.Columns[3].HeaderText = "Đơn giá";

dtGVChiTietXuatKho.Columns[4].HeaderText = "Số lượng";

dtGVChiTietXuatKho.Columns[5].HeaderText = "Thành tiền";

}

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

{

HienThiXuatKho();

if (dtGVXuatKho.Rows.Count > 0)

{

var row = this.dtGVXuatKho.Rows[0];

HienThiChiTietXuatKho(row.Cells["MaHD"].Value.ToString());

}

else

{

HienThiChiTietXuatKho("");

}

}

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

{

save = true;

frmXuatKhoInfor frm = new frmXuatKhoInfor();

frm.Text = "Thêm mới";

frm.ShowDialog();

HienThiXuatKho();

if (dtGVXuatKho.Rows.Count > 0)

{

var row = this.dtGVXuatKho.Rows[0];

HienThiChiTietXuatKho(row.Cells["MaHD"].Value.ToString());

}

else

{

HienThiChiTietXuatKho("");

}

}

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

{

if (dtGVXuatKho.Rows.Count == 0)

{

return;

}

DataGridViewRow row = this.dtGVXuatKho.Rows[dtGVXuatKho.CurrentCell.RowIndex];

save = false;

mahd = row.Cells[0].Value.ToString();

ngayban = DateTime.Parse(row.Cells[1].Value.ToString());

makh = row.Cells[2].Value.ToString();

manv = row.Cells[4].Value.ToString();

tennv = row.Cells[5].Value.ToString();

tongtien = row.Cells[6].Value.ToString();

trangthai = row.Cells[7].Value.ToString();

if (trangthai == "Đã thanh toán")

{

MessageBox.Show("Phiếu này đã được thanh toán, không được sửa.");

return;

}

frmXuatKhoInfor frm = new frmXuatKhoInfor();

frm.Text = "Sửa phiếu";

frm.ShowDialog();

HienThiXuatKho();

if (dtGVXuatKho.Rows.Count > 0)

{

var row2 = this.dtGVXuatKho.Rows[0];

HienThiChiTietXuatKho(row2.Cells["MaHD"].Value.ToString());

}

else

{

HienThiChiTietXuatKho("");

}

}

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

{

if (dtGVXuatKho.Rows.Count == 0)

{

return;

}

DataGridViewRow row = this.dtGVXuatKho.Rows[dtGVXuatKho.CurrentCell.RowIndex];

trangthai = row.Cells[7].Value.ToString();

if (trangthai == "Đã thanh toán")

{

MessageBox.Show("Phiếu này đã được thanh toán, không được xóa.");

return;

}

DialogResult dr = MessageBox.Show("Có chắc chắn xóa dòng dữ liệu này không ?", "Xác nhận", MessageBoxButtons.YesNo, MessageBoxIcon.Question);

if (dr == DialogResult.Yes)

{

db.Xoa(dtGVXuatKho.Rows[dtGVXuatKho.CurrentCell.RowIndex].Cells[0].Value.ToString());

MessageBox.Show("Xóa thành công ", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Information, MessageBoxDefaultButton.Button1);

HienThiXuatKho();

if (dtGVXuatKho.Rows.Count > 0)

{

var row2 = this.dtGVXuatKho.Rows[0];

HienThiChiTietXuatKho(row2.Cells["Mahd"].Value.ToString());

}

else

{

HienThiChiTietXuatKho("");

}

}

else

return;

}

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

{

if (dtGVXuatKho.Rows.Count == 0)

{

return;

}

DataGridViewRow row1 = this.dtGVXuatKho.Rows[dtGVXuatKho.CurrentCell.RowIndex];

trangthai = row1.Cells[7].Value.ToString();

if (trangthai == "Đã thanh toán")

{

MessageBox.Show("Phiếu này đã được thanh toán.");

return;

}

DataGridViewRow row = this.dtGVXuatKho.Rows[dtGVXuatKho.CurrentCell.RowIndex];

DialogResult ok = new DialogResult();

ok = MessageBox.Show("Bạn có muốn tính tiền cho khách hàng này Không ?", "Thông báo", MessageBoxButtons.YesNo, MessageBoxIcon.Question);

if (ok == DialogResult.Yes)

{

MessageBox.Show("TỔNG SỐ TIỀN THANH TOÁN LÀ " + row.Cells[6].Value.ToString(), "Thông báo",

MessageBoxButtons.OK, MessageBoxIcon.Asterisk);

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

{

db.UpdateTonKho(decimal.Parse(dtGVChiTietXuatKho.Rows[i].Cells["SoLuong"].Value.ToString()), dtGVChiTietXuatKho.Rows[i].Cells["MaHH"].Value.ToString());

}

db.UpdateTrangThaiSauThanhToan(row.Cells[0].Value.ToString());

HienThiXuatKho();

if (dtGVXuatKho.Rows.Count > 0)

{

var rowx = this.dtGVXuatKho.Rows[0];

HienThiChiTietXuatKho(rowx.Cells["MaHD"].Value.ToString());

}

else

{

HienThiChiTietXuatKho("");

}

}

else

{

return;

}

}

private void dtGVXuatKho\_CellClick(object sender, DataGridViewCellEventArgs e)

{

if (e.RowIndex >= 0)

{

DataGridViewRow row = this.dtGVXuatKho.Rows[e.RowIndex];

trangthai = row.Cells[7].Value.ToString();

HienThiChiTietXuatKho(row.Cells[0].Value.ToString());

}

}

private void dtGVChiTietXuatKho\_CellContentClick(object sender, DataGridViewCellEventArgs e)

{

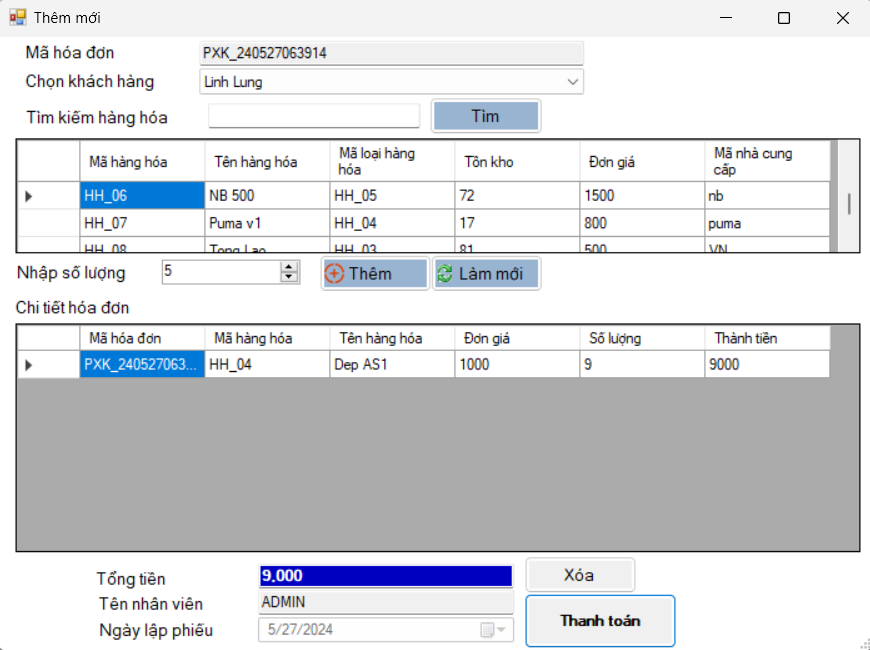
}

}

}

**3.2.2.11 Giao diện Form thêm mới phiếu xuất kho**

Khi bấm Button Thêm ở Form Phiếu xuất kho màn hìnhsẽ chuyển sang giao diện Form Thêm mới giúp thêm phiếu xuất mới.

****

Hình 3.12 Giao diện form thêm mới phiếu xuất

* **Thiết kế**

btnThem\_Click: Button Thêm mới phiếu xuất kho .

btnLamMoi\_Click: Button làm mới thông tin phiếu xuất vừa thêm.

btnXoa\_Click: Button xóa thông tin phiếu xuất kho mới.

btnLuu\_Click: Button lưu ( Sau khi thêm hoặc sửa thông tin xong bấm nút Lưu để lưu phiếu xuất kho vừa tạo mới) .

btnTim\_Click: Button Tìm kiếm hàng hóa giúp tìm kiếm thông tin hàng hóa vừa thêm.

btnThanhToan\_Click: Button thanh toán tiền cho những đơn hàng vừa thêm mới.

btnThoat\_Click: Button thoát sẽ thoát khỏi form Thêm mới

* **View code giao diện thêm mới phiếu xuất**

//Thêm phiếu

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

{

if (grdHangHoa.Rows.Count == 0)

{

MessageBox.Show("Không có hàng hóa để tạo phiếu", "Thông báo",

MessageBoxButtons.OK, MessageBoxIcon.Error);

return;

}

DataGridViewRow row = this.grdHangHoa.Rows[grdHangHoa.CurrentCell.RowIndex];

string MaHD = txtmahd.Text;

string MaHH = row.Cells[0].Value.ToString();

decimal DonGia = decimal.Parse(row.Cells[4].Value.ToString() + "");

decimal SoLuong = nmsoluong.Value;

decimal ThanhTien = DonGia \* SoLuong;

if (dbhd.CheckExitsXuatKho(txtmahd.Text) == 1)

{

dbhd.ThemXuatKho(MaHD, cbomakh.SelectedValue.ToString(), dtNgayHD.Value.ToString("yyyy/MM/dd"), frmDangNhap.manv, decimal.Parse(txtTongTien.Text));

}

else

{

dbhd.SuaXuatKho(MaHD, cbomakh.SelectedValue.ToString(), dtNgayHD.Value.ToString("yyyy/MM/dd"), frmDangNhap.manv, decimal.Parse(txtTongTien.Text));

}

decimal a = dbhd.KiemTraHangHoaTonKho(MaHD, MaHH);

decimal b = nmsoluong.Value;

if (dbhd.KiemTraHangHoaTonKho(MaHD, MaHH) < nmsoluong.Value)

{

MessageBox.Show("Tồn kho của sản phẩm này không đủ để xuất bán.");

return;

}

if (dbhd.CheckExits(MaHD, MaHH) == 0)

{

dbhd.CapNhatSoLuongTienChiTietXuatKho(MaHD, MaHH, SoLuong);

}

else

{

dbhd.ThemChiTietXuatKho(MaHD, MaHH, DonGia, SoLuong, ThanhTien);

}

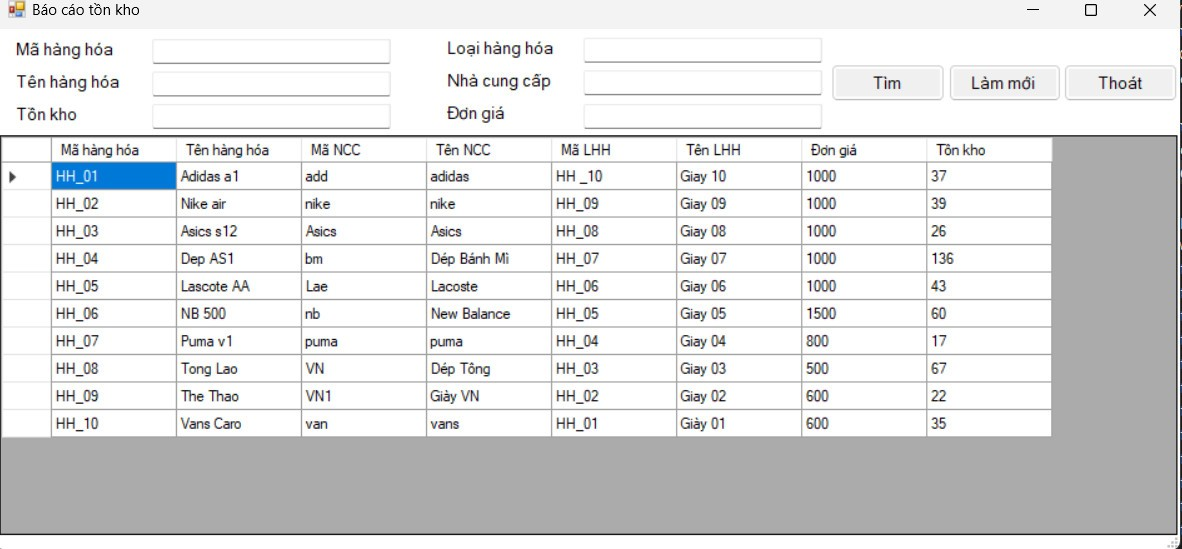
dbhd.UpdateTongTien(MaHD);

HienThiChiTietXuatKho();

TongTienXuatKho();

}

**3.2.2.12 Giao diện Form Báo cáo tồn kho**

****

Hình 3.13 Form Báo cáo tồn kho

* **Thiết kế**

btnTim\_Click: Button Tìm kiếm hàng hóa giúp tìm kiếm hàng hóa tồn kho.

btnLamMoi\_Click: Button làm mới thông tin vừa nhập.

btnThoat\_Click: Button thoát sẽ thoát khỏi form Báo cáo tồn kho.

* **View code giao diện báo cáo tồn kho**

namespace PhanMemQuanLyBanGiayDep.Views

{

public partial class frmBaoCaoTonKho : Form

{

public frmBaoCaoTonKho()

{

InitializeComponent();

}

BCTonKhoControllers db = new BCTonKhoControllers();

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

{

this.Close();

}

private void HienThiHangHoa()

{

gridview.DataSource = db.HienThi(txtMaHH.Text,txtTenHH.Text,txtTenNCC.Text,txtTenLHH.Text,txtTonKho.Text,txtDonGia.Text);

gridview.Columns[0].HeaderText = "Mã hàng hóa";

gridview.Columns[1].HeaderText = "Tên hàng hóa";

gridview.Columns[2].HeaderText = "Mã NCC";

gridview.Columns[3].HeaderText = "Tên NCC";

gridview.Columns[4].HeaderText = "Mã LHH";

gridview.Columns[5].HeaderText = "Tên LHH";

gridview.Columns[6].HeaderText = "Đơn giá";

gridview.Columns[7].HeaderText = "Tồn kho";

}

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

{

HienThiHangHoa();

}

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

{

txtMaHH.Text = "";

txtTenHH.Text = "";

txtDonGia.Text = "";

txtTonKho.Text = "";

txtTenLHH.Text = "";

txtTenNCC.Text = "";

HienThiHangHoa();

}

private void frmBaoCaoTonKho\_Load(object sender, EventArgs e)

{

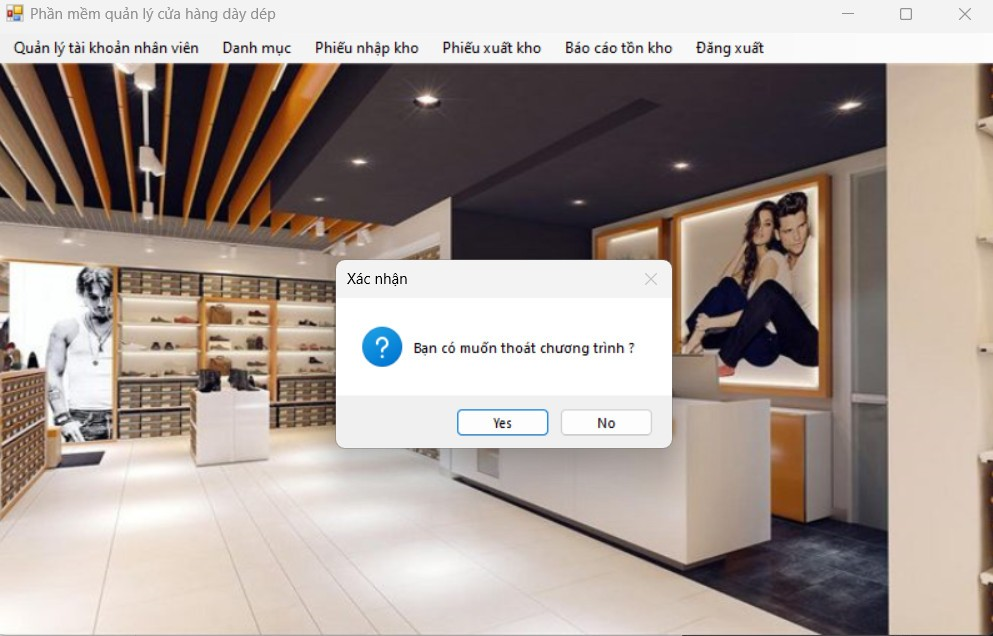
HienThiHangHoa();

}

}

}

**3.2.2.13 đăngXuấtToolStripMenuItem\_Click**

****

Hình 3.14 Hiển thị màn hình thoát chương trình

* Khi người dùng click vào menu item "Đăng xuất" thi sự kiện đăngXuấtToolStripMenuItem\_Click sẽ được gọi bấm yes để thoát chương trình

### 3.2.3 Kết quả kiểm thử chương trình

- Về cơ bản, phần mềm đã chạy thành công và chạy được trên các hệ thống của Windows.

- Giao diện tương đối dễ nhìn, phù hợp, thân thiện với người dùng.

# CHƯƠNG 4: KẾT LUẬN

## 4.1 Kết quả đạt được

* Xây dựng được giao diện thân thiện, dễ sử dụng cho việc quản lý cửa hàng bán giày dép.
* Kết nối CSDL để quản lý hàng hóa, tránh được rủi ro trong khâu nhập xuất.
* Hỗ trợ hoạt động trên nhiều môi trường Windows phổ biến.
* Khách hàng có thể tìm kiếm thông tin sản phẩm và thanh toán sản phẩm giày dép một cách nhanh chóng.
* Phần mềm có ấn tượng tốt, thân thiện với người dùng.

## 4.2 Hạn chế của đề tài

* Do thời gian khảo sát và ngôn ngữ lập trình mới tiếp cận nên một số chức năng còn sơ sài, chưa đầy đủ, chưa phát triển đầy đủ chức năng tìm kiếm,
* Chức năng quản lý nhà cung cấp chưa được tối ưu hóa.
* Chưa có các tính năng thống kê, báo cáo đầy đủ.
* Phạm vi triển khai chỉ dành riêng cho cửa hàng bán giày dép, chưa mở rộng cho các lĩnh vực khác.

## 4.3 Hướng phát triển

* Hoàn thiện phần mềm "Quản lý cửa hàng bán giày dép" để hỗ trợ quản lý hàng hóa, nhân viên, khách hàng, nhà cung cấp một cách hiệu quả.
* Mở rộng chức năng để quản lý các chi nhánh, nhân sự, lương, thu chi và các tính năng quản lý cấp cao khác.
* Bổ sung các tính năng còn thiếu, tối ưu hóa các chức năng hiện có, đảm bảo độ tin cậy, an toàn cho hệ thống.
* Xem xét khả năng mở rộng phần mềm cho các lĩnh vực kinh doanh khác ngoài giày dép.

# TÀI LIỆU THAM KHẢO

[1]. Nguyễn Văn Ba, *Phân tích thiết kế hệ thống thông tin* , Nhà xuất bản ĐHQG Hà Nội.

[2]. Ths.Lê Thị Kiều Oanh, Ths.Đào Thị Phương Anh, Tài liệu học tập *Cơ sở dữ liệu*, Trường Đại học Kinh tế Kỹ thuật Công nghiệp, 2019.

[3]. Phạn Hữu Khang - Phương Lan. Microsoft SQL Server 2008 - *Quản trị cơ sở dữ liệu*. NXB Lao động - Xã hội.

[4]. Thạc Bình Cường. *Phân Tích Và Thiết Kế Hệ Thống Thông Tin*. NXB Khoa Học Kỹ Thuật. 2004.

[5]. http://SinhVienIT.Net