**KHOA CÔNG NGHỆ THÔNG TIN**

**TRẦN MINH KHA**

**MSSV: 220576**

**DƯƠNG DUY KHANG**

**MSSV: 224167**

**TÊN ĐỀ TÀI**

**XÂY DỰNG PHẦN MỀM QUẢN LÝ CỬA HÀNG QUẦN ÁO**

**ĐỒ ÁN CƠ SỞ 01**

**Ngành: công nghệ thông tin  
Mã ngành: 7480201**

**CÁN BỘ HƯỚNG DẪN**

**LÊ ĐỨC THẮNG**

# LỜI CẢM ƠN

Được sự phân công của khoa Công Nghệ Thông Tin Trường Đại Học Nam Cần Thơ, và dưới sự hướng dẫn của thầy Lê Đức Thắng, bọn em đã hoàn thành đề tài “xây dựng phầm mềm quản lý quần áo”.

Chúng em, Trần Minh Kha và Dương Duy Khang, xin gửi lời cảm ơn chân thành đến Trường Đại học Nam Cần Thơ, đặc biệt là Khoa Công nghệ Thông tin, đã tạo điều kiện thuận lợi để chúng tôi thực hiện đồ án này. Chúng em xin bày tỏ lòng biết ơn sâu sắc đến Thầy Lê Đức Thắng, giảng viên hướng dẫn, đã tận tình hướng dẫn, và góp ý quý báu trong suốt quá trình thực hiện.

Chúng em đã cố gắng thực hiện đề tài một cách hoàn chỉnh nhất nhưng do trình độ hiểu biết và nhận thức còn chưa cao nên trong đồ án còn nhiều sự thiếu sót không thể tránh khỏi mong, bọn em mong nhận được sự đóng góp ý kiến của thầy để hoàn thiện đồ án được tốt hơn.

Em xin chân thành cảm ơn!

|  |  |
| --- | --- |
| Sinh viên thực hiện  Dương Duy Khang | *Cần Thơ, tháng 06, năm 2025*  Sinh viên thực hiện  Trần Minh Kha |

# LỜI CAM KẾT

Chúng tôi xin cam kết đồ án cơ sở 01 này được hoàn thành dựa trên kết quả nghiên cứu của chúng tôi và kết quả này chưa được sử dụng cho bất kỳ đồ án cơ sở nào khác.

|  |  |
| --- | --- |
| Sinh viên thực hiện  Dương Duy Khang | *Cần Thơ, tháng 06, năm 2025*  Sinh viên thực hiện  Trần Minh Kha |

# NHẬN XÉT ĐÁNH GIÁ CỦA GIÁO VIÊN HƯỚNG DẪN

*Cần Thơ, ngày….tháng…. năm 2025*

# NHẬN XÉT ĐÁNH GIÁ CỦA GIÁO VIÊN PHẢN BIỆN

*Cần Thơ, ngày….tháng…. năm 2025*

# MỤC LỤC

LỜI CẢM ƠN i

LỜI CAM KẾT ii

NHẬN XÉT ĐÁNH GIÁ CỦA GIÁO VIÊN HƯỚNG DẪN iii

NHẬN XÉT ĐÁNH GIÁ CỦA GIÁO VIÊN PHẢN BIỆN iv

MỤC LỤC v

DANH SÁCH HÌNH ix

CHƯƠNG 1 GIỚI THIỆU 1

1.1 Hoàn cảnh xuất phát đề tài 1

1.2 Sản phẩm nghiệm thu 2

CHƯƠNG 2 CƠ SỞ LÝ LUẬN VÀ PHƯƠNG PHÁP NGHIÊN CỨU 3

2.1 NGHIÊN CỨU LÝ THUYẾT CƠ SỞ. 3

2.1.1Tổng quan về hệ thống quản lý cửa hàng 3

2.1.2 Mô hình 3 tầng (3-Layer Architecture) 3

2.1.3 Công nghệ sử dụng 4

2.1.4 Các khái niệm và kỹ thuật lập trình áp dụng 5

2.1.5 Cơ sở lý thuyết về chức năng hệ thống 6

2.2 THIẾT KẾ THUẬT TOÁN 7

2.2.1 Thuật toán xử lý giao diện người dùng (UI/UX Algorithms) 7

2.2.2 Thuật toán xử lý logic kinh doanh (Business Logic Algorithms) 8

2.2.3 Thuật toán quản lý danh mục (Catalog Management Algorithms) 9

2.2.4 Thuật toán quản lý hệ thống (System Management Algorithms) 10

2.2.5 Thuật toán nhập hàng (Import Management Algorithms) 10

2.2.6 Thuật toán hóa đơn (Invoice Management Algorithms) 11

2.2.7 Thuật toán báo cáo (Reporting Algorithms) 11

2.2.8 Thuật toán xử lý lỗi (Error Handling Algorithms) 12

2.2.9 Thuật toán xử lý giao diện con (Sub-Form Handling Algorithms) 12

2.2.10 Thuật toán xử lý chuỗi (String Processing Algorithms) 12

CHƯƠNG 3 PHÂN TÍCH HỆ THỐNG 13

3.1 XÂY DỰNG CƠ SỞ DỮ LIỆU 13

3.1.1 Tổng quan 13

3.1.2 Thiết kế cơ sở dữ liệu 13

3.1.3 Ràng buộc toàn vẹn 14

3.1.4 Tối ưu hóa hiệu suất 14

3.1.5 Tích hợp với mô hình 3 tầng 15

3.1.6 Giới thiệu phần mã SQL 15

3.2 XÂY DỰNG PHẦN MỀM 17

3.2.1 Tổng quan 17

3.2.2 Công nghệ và công cụ sử dụng 17

3.2.3 Cấu trúc phần mềm (Mô hình 3 tầng) 18

3.2.4 Các chức năng chính và triển khai 18

3.2.5 Xử lý lỗi và tối ưu hóa 20

3.2.6 Tích hợp với cơ sở dữ liệu 21

3.3 MÔ HÌNH HÓA YÊU CẦU PHẦN MỀM 21

3.3.1 Mô hình dữ liệu mức quan niệm (CDM) 21

3.3.2 Tổng quan 21

3.3.3 Các thực thể và thuộc tính 22

3.3.4 Mối quan hệ giữa các thực thể 24

3.3.5 Mô hình dữ liệu mức Logic (LDM) 25

3.3.6 Cấu trúc các bảng và thuộc tính 26

3.3.7 Ràng buộc toàn vẹn 27

3.3.8 Tối ưu hóa 27

3.4 MÔ HÌNH DỮ LIỆU MỨC VẬT LÝ (PDM) 27

3.4.1 Tổng quan 28

3.4.2 Cấu trúc cơ sở dữ liệu vật lý 28

3.4.3 Tối ưu hóa hiệu suất 29

3.4.4 Tích hợp với phần mềm 29

3.5 MÔ HÌNH DỮ LIỆU CƠ SỞ DỮ LIỆU (RDM) 29

3.5.1 Tổng quan 30

3.5.2. Nguyên tắc cơ bản 30

3.5.3 Cấu trúc các bảng quan hệ 30

3.5.4 Mối quan hệ và ràng buộc 31

3.5.5 Tối ưu hóa 31

3.5.6 Ứng dụng 31

3.6 SƠ ĐỒ USE CASE 32

3.6.1 Tổng quan 32

3.6.2 Các Nhân vật (Actors) 32

3.6.3 Các Trường hợp Sử dụng (Use Cases) 33

3.6.4 Mối quan hệ (Relationships) 34

3.7 SƠ ĐỒ LUỒNG DỮ LIỆU CẤP 0 (DFD LEVEL 0) 35

3.7.1 Tổng quan 35

3.7.2 Thành phần chính của DFD cấp 0 35

3.7.3 Mô tả DFD cấp 0 36

3.8 SƠ ĐỒ LUỒNG DỮ LIỆU CẤP 1 (DFD LEVEL 1) 38

3.8.1 Tổng quan 38

3.8.2 Thành phần chính của DFD cấp 1 38

3.8.3 Mô tả DFD cấp 1 cho từng quá trình cấp 0 39

CHƯƠNG 4 GIAO DIỆN HỆ THỐNG 41

4.1 Trang đăng nhập 41

4.2 Trang chủ 42

4.3 Trang thông tin nhân viên 42

4.4 Trang thông tin chức vụ 43

4.5 Trang thông tin khách hàng 43

4.6 Trang thông tin nhà cung cấp 44

4.7 Trang thông tin khuyến mãi 44

4.8 Trang thông tin chất liệu 45

4.9 Trang thông tin thương hiệu 45

4.10 Trang thông tin loại sản phẩm 46

4.11 Trang thông tin sản phẩm 46

4.12 Trang phân quyền 47

4.13 Trang chọn tài khoản 47

4.14 Trang thông tin phiếu nhập 48

4.15 Trang thông tin hóa đơn 48

4.16 Trang thông tin doanh thu 49

4.17 Trang thống kê sản phẩm 49

4.18 Trang thống kê bán hàng 50

4.19 Trang chăm sóc khách hàng 50

KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN 51

TÀI LIỆU THAM KHẢO 52

# DANH SÁCH HÌNH

[Hình 3.1 Mô hình dữ liệu mức qua niệm (CDM) 21](#_Toc206007896)

[Hình 3.2 Mô hình dữ liệu mức Logic (LDM) 25](#_Toc206007897)

[Hình 3.3 Mô hình dữ liệu mức vật lý (pdm) 27](#_Toc206007898)

[Hình 3.4 Mô hình dữ liệu cơ sở dữ liệu (rdm) 29](#_Toc206007899)

[Hình 3.5 Sơ đồ Use case 32](#_Toc206007900)

[Hình 3.6 Sơ đồ luồng dữ liệu cấp 0 (dfd level 0) 35](#_Toc206007901)

[Hình 3.7 Sơ đồ luồng dữ liệu cấp 1 (dfd level 1) 38](#_Toc206007902)

[Hình 4.1 Trang đăng nhập 41](#_Toc206007903)

[Hình 4.2 Trang chủ 42](#_Toc206007904)

[Hình 4.3 Trang thông tin nhân viên 42](#_Toc206007905)

[Hình 4.4 Trang thông tin chức vụ 43](#_Toc206007906)

[Hình 4.5 Trang thông tin khách hàng 43](#_Toc206007907)

[Hình 4.6 Trang thông tin nhà cung cấp 44](#_Toc206007908)

[Hình 4.7 Trang thông tin khuyến mãi 44](#_Toc206007909)

[Hình 4.8 Trang thông tin chất chịu 45](#_Toc206007910)

[Hình 4.9 Trang thông tin thương hiệu 45](#_Toc206007911)

[Hình 4.10 Trang thông tin loại sản phẩm 46](#_Toc206007912)

[Hình 4.11 Trang thông tin sản phẩm 46](#_Toc206007913)

[Hình 4.12 Trang phân quyền 47](#_Toc206007914)

[Hình 4.13 Trang chọn tài khoản 47](#_Toc206007915)

[Hình 4.14 Trang thông tin phiếu nhập 48](#_Toc206007916)

[Hình 4.15 Trang thông tin hóa đơn 48](#_Toc206007917)

[Hình 4.16 Trang thông tin doanh thu 49](#_Toc206007918)

[Hình 4.17 Trang thống kê sản phẩm 49](#_Toc206007919)

[Hình 4.18 Trang thống kê bán hàng 50](#_Toc206007920)

[Hình 4.19 Trang chăm sóc khách hàng 50](#_Toc206007921)

# CHƯƠNG 1 GIỚI THIỆU

## 1.1 Hoàn cảnh xuất phát đề tài

**Tình huống thực tế:** Các cửa hàng quần áo tại Việt Nam, đặc biệt trong lĩnh vực thời trang và sản phẩm thủ công, đang phát triển mạnh nhờ xu hướng người tiêu dùng ưa chuộng các sản phẩm độc đáo, phản ánh bản sắc địa phương. Tuy nhiên, với quy mô vừa và nhỏ, những cửa hàng này đối mặt với nhiều thách thức trong quản lý kho, theo dõi doanh thu và chăm sóc khách hàng. Thương hiệu Y

ody từng nhận định rằng các phần mềm hiện có chỉ đáp ứng được 60-70% nhu cầu, làm nổi bật sự cần thiết của một giải pháp được tùy chỉnh riêng. Hiện tại, nhiều cửa hàng vẫn sử dụng phương pháp thủ công như ghi chép trên giấy hoặc Excel, dẫn đến mất thời gian, dễ xảy ra sai sót trong thống kê và khó khăn trong việc tra cứu thông tin nhanh chóng. Việc xây dựng một phần mềm quản lý cửa hàng quần áo sẽ giúp tự động hóa quy trình, giảm thiểu sai sót và nâng cao hiệu quả kinh doanh.

**Tình hình nghiên cứu:** Tại Việt Nam, nghiên cứu chủ yếu tập trung vào thương mại điện tử và các hệ thống quản lý quy mô lớn, trong khi bán lẻ offline quy mô nhỏ nhận được ít sự quan tâm. Các công trình từ Đại học Kinh tế Đà Nẵng, Đại học Bách khoa TP.HCM và nghiên cứu của Đặng Thái Bình chỉ ra rằng rào cản về tài chính và kỹ năng công nghệ của các doanh nghiệp vừa và nhỏ khiến WinForms trở thành giải pháp khả thi. Các phần mềm phổ biến như KiotViet, Sapo hay Nhanh.vn thường có chi phí cao, thiếu tính linh hoạt để đáp ứng nhu cầu đặc thù của quần áo. Trên thế giới, các nghiên cứu tập trung vào hệ thống POS dựa trên đám mây và trí tuệ nhân tạo, nhưng các giải pháp như Square hay Shopify cung cấp tính năng quản lý toàn diện lại chưa phù hợp với đặc thù thị trường Việt Nam, đặc biệt là các thương hiệu địa phương. Đề tài này hướng đến phát triển một phần mềm WinForms đơn giản, tiết kiệm chi phí và đáp ứng đúng nhu cầu của các cửa hàng quần áo tại Việt Nam, nhờ khả năng hoạt động ngoại tuyến, giao diện thân thiện và chi phí thấp.

**Dự kiến ứng dụng:** Phần mềm không chỉ hỗ trợ quản lý sản phẩm, khách hàng, hóa đơn và kho hàng mà còn có tiềm năng mở rộng trong tương lai. Những cải tiến như tích hợp công nghệ đám mây, ứng dụng trí tuệ nhân tạo, kết nối với các nền tảng thương mại điện tử như Shopee, hoặc tích hợp các phương thức thanh toán trực tuyến như Momo, ZaloPay sẽ giúp hỗ trợ phát triển bền vững. Những nâng cấp này sẽ giúp các cửa hàng quần áo nâng cao hiệu quả hoạt động, tăng khả năng cạnh tranh và thúc đẩy quá trình chuyển đổi số, đồng thời tùy chỉnh cho nhiều lĩnh vực khác nhau, bao gồm quản lý chuỗi cửa hàng và phân tích dữ liệu khách hàng.

## 1.2 Sản phẩm nghiệm thu

Sản phẩm là một phần mềm chạy trên PC, được phát triển bằng WinForms (C#) và sử dụng cơ sở dữ liệu SQL Server. Phần mềm cung cấp các chức năng quản lý sản phẩm, khách hàng, hóa đơn, báo cáo, nhân viên, doanh thu và thống kê được triển khai dưới dạng file thực thi (.exe) tương thích với hệ điều hành Windows 7/8/10/11. Giao diện thân thiện và khả năng hoạt động ngoại tuyến giúp phần mềm phù hợp với các cửa hàng quần áo có nguồn lực hạn chế, đồng thời đảm bảo hiệu quả trong quản lý hàng ngày.

# CHƯƠNG 2 CƠ SỞ LÝ LUẬN VÀ PHƯƠNG PHÁP NGHIÊN CỨU

## 2.1 NGHIÊN CỨU LÝ THUYẾT CƠ SỞ.

### **2.1.1****Tổng quan về hệ thống quản lý cửa hàng**

Hệ thống quản lý cửa hàng là một phần mềm hỗ trợ quản lý các hoạt động kinh doanh như quản lý danh mục (nhân viên, khách hàng, sản phẩm, v.v.), nhập hàng, lập hóa đơn, báo cáo doanh thu và thống kê. Mục tiêu là tự động hóa các quy trình, giảm thiểu sai sót, tăng hiệu quả quản lý và cung cấp thông tin kịp thời cho người dùng.

**Tầm quan trọng:**

- Tăng hiệu quả vận hành: Tự động hóa các tác vụ như tính toán hóa đơn, kiểm tra tồn kho.

- Hỗ trợ ra quyết định: Báo cáo doanh thu và thống kê giúp phân tích hiệu quả kinh doanh.

- Bảo mật và phân quyền: Đảm bảo chỉ người dùng được ủy quyền mới truy cập các chức năng nhạy cảm.

**Yêu cầu hệ thống:**

- Giao diện thân thiện, dễ sử dụng.

- Xử lý dữ liệu nhanh chóng, chính xác.

- Tích hợp cơ sở dữ liệu để lưu trữ và truy xuất thông tin.

- Đảm bảo tính bảo mật và phân quyền người dùng.

### 2.1.2 Mô hình 3 tầng (3-Layer Architecture)

Hệ thống được thiết kế theo mô hình 3 tầng (Presentation Layer, Business Logic Layer - BLL, Data Access Layer - DAL) với các lớp đối tượng truyền dữ liệu (DTO), đảm bảo tính mô-đun, dễ bảo trì và mở rộng.

**Presentation Layer (Tầng giao diện):**

- Được xây dựng bằng **WinForms** trong C#, cung cấp giao diện người dùng trực quan với các thành phần như DataGridView, Button, ComboBox, TextBox.

- Xử lý tương tác người dùng thông qua mô hình lập trình dựa trên sự kiện (event-driven programming).

- Liên kết dữ liệu (data binding) để hiển thị thông tin từ cơ sở dữ liệu lên giao diện.

**Business Logic Layer (BLL):**

- Chứa các quy tắc kinh doanh và logic xử lý, ví dụ: kiểm tra tính hợp lệ của dữ liệu, tạo mã tự động, tính toán tổng tiền hóa đơn, áp dụng mã khuyến mãi.

- Nhận dữ liệu từ giao diện thông qua các đối tượng DTO và chuyển đến DAL để xử lý cơ sở dữ liệu.

- Đảm bảo tính nhất quán và hợp lệ của dữ liệu trước khi lưu trữ.

**Data Access Layer (DAL):**

- Quản lý truy xuất dữ liệu từ cơ sở dữ liệu (thường sử dụng SQL Server) thông qua ADO.NET hoặc Entity Framework.

- Sử dụng các truy vấn SQL hoặc stored procedure để thực hiện các thao tác CRUD (Create, Read, Update, Delete).

- Đóng vai trò trung gian giữa BLL và cơ sở dữ liệu, đảm bảo tách biệt logic kinh doanh và truy xuất dữ liệu.

**Data Transfer Object (DTO):**

- Các lớp như NhanVienDTO, SanPhamDTO, HoaDonDTO được sử dụng để truyền dữ liệu giữa các tầng.

- Đảm bảo tính đóng gói, giảm sự phụ thuộc trực tiếp giữa giao diện và cơ sở dữ liệu.

- Ví dụ: NhanVienDTO chứa các thuộc tính như MaNhanVien, HoTen, SoDienThoai,…

### 2.1.3 Công nghệ sử dụng

**C# và WinForms:**

- C#: Ngôn ngữ lập trình hướng đối tượng mạnh mẽ, hỗ trợ phát triển ứng dụng Windows với hiệu suất cao.

- WinForms: Framework cho phép xây dựng giao diện người dùng đồ họa (GUI) với các thành phần như form, button, datagridview, hỗ trợ lập trình dựa trên sự kiện.

**Cơ sở dữ liệu SQL Server:**

- SQL (Structured Query Language) là một ngôn ngữ lập trình được sử dụng để quản lý và truy vấn dữ liệu trong cơ sở dữ liệu quan hệ. Nó cho phép người dùng thực hiện các thao tác như tạo, sửa đổi, xóa và truy xuất dữ liệu từ cơ sở dữ liệu.

- Lưu trữ dữ liệu quan hệ với các bảng như NhanVien, SanPham, HoaDon, PhieuNhap.

- Sử dụng stored procedure để tối ưu hóa truy vấn và đảm bảo bảo mật.

**ADO.NET:**

- Công nghệ kết nối và truy xuất dữ liệu từ SQL Server, sử dụng các lớp như SqlConnection, SqlCommand, SqlDataAdapter.

- Hỗ trợ xử lý dữ liệu hiệu quả và an toàn.

### 2.1.4 Các khái niệm và kỹ thuật lập trình áp dụng

**Lập trình dựa trên sự kiện (Event-Driven Programming)**

- Khái niệm: Hệ thống phản hồi các hành động của người dùng (như click nút, chọn dòng) thông qua các sự kiện (EventArgs).

- Ứng dụng:

+ Các sự kiện như btnThemNV\_Click, dgvNhanVien\_CellClick trong NhanVien.cs xử lý thao tác người dùng.

+ Sự kiện cbBaoCao\_SelectedIndexChanged trong BaoCao.cs để chọn loại báo cáo.

**Liên kết dữ liệu (Data Binding)**

- Khái niệm: Kỹ thuật đồng bộ dữ liệu giữa nguồn dữ liệu (List<DTO>, DataSet) và giao diện (DataGridView).

- Ứng dụng: Tự động cập nhật giao diện khi dữ liệu thay đổi, giảm mã thủ công.

**Kiểm tra tính hợp lệ (Validation Logic)**

Khái niệm: Kiểm tra dữ liệu đầu vào để đảm bảo tính hợp lệ trước khi xử lý.

Ứng dụng: Sử dụng IF-ELSE, string.IsNullOrEmpty để kiểm tra các trường như mã, số điện thoại, số lượng.

**Tạo mã tự động (Auto-Generation)**

- Khái niệm: Tạo mã duy nhất dựa trên quy tắc (ngày giờ hoặc số thứ tự).

- Ứng dụng: Tạo mã nhân viên, mã hóa đơn, mã phiếu nhập.

**Quản lý trạng thái (State Management)**

- Khái niệm: Quản lý trạng thái giao diện (ví dụ: bật/tắt trường nhập liệu) để đảm bảo tính nhất quán.

- Ứng dụng: Vô hiệu hóa mã nhân viên (txtMaNhanVien.Enabled = false) sau khi tạo.

**Xử lý lỗi (Error Handling)**

- Khái niệm: Sử dụng khối try-catch để bắt và xử lý ngoại lệ, ghi log lỗi.

- Ứng dụng: Đảm bảo hệ thống ổn định, thông báo lỗi thân thiện qua MessageBox.

**Quản lý giao diện con (Dialog Management)**

- Khái niệm: Sử dụng form con để thu thập dữ liệu với ShowDialog() và DialogResult.

- Ứng dụng: Form con như FormNhapNgay, FormNhapThangNam trong BaoCao.cs.

**Xử lý chuỗi (String Processing)**

- Khái niệm: Chuẩn hóa chuỗi đầu vào (loại bỏ khoảng trắng, kiểm tra định dạng).

- Ứng dụng: Sử dụng Trim(), ToUpper() trong NhanVien.cs, HoaDon.cs.

### 2.1.5 Cơ sở lý thuyết về chức năng hệ thống

- Quản lý danh mục: Dựa trên mô hình CRUD (Create, Read, Update, Delete) để quản lý các thực thể như nhân viên, sản phẩm, khách hàng.

- Tài khoản và phân quyền: Áp dụng mô hình Role-Based Access Control (RBAC) để phân quyền dựa trên vai trò (Quản lý, Nhân viên).

- Nhập hàng và hóa đơn: Dựa trên quy trình kinh doanh thực tế, đảm bảo cập nhật tồn kho và tính toán chính xác.

- Báo cáo: Sử dụng kỹ thuật phân tích dữ liệu để tổng hợp doanh thu và thống kê, hỗ trợ ra quyết định.

- Đăng xuất: Quản lý phiên người dùng để đảm bảo bảo mật.

**Ưu điểm của mô hình 3 tầng**

- Tách biệt trách nhiệm: Giao diện, logic kinh doanh và dữ liệu được tách biệt, dễ bảo trì và mở rộng.

- Tái sử dụng mã: Các lớp DTO và phương thức BLL/DAL có thể dùng lại cho nhiều chức năng.

- Bảo mật: DAL xử lý truy vấn SQL, giảm nguy cơ SQL Injection.

- Dễ kiểm thử: Mỗi tầng có thể được kiểm thử độc lập.

## 2.2 THIẾT KẾ THUẬT TOÁN

### 2.2.1 Thuật toán xử lý giao diện người dùng (UI/UX Algorithms)

**Xử lý sự kiện (Event Handling)**

- Áp dụng:

+ Trong NhanVien.cs, các sự kiện như thêm, sửa, xóa, chọn nhân viên xử lý tương tác người dùng.

+ Trong BaoCao.cs, sự kiện cbBaoCao\_SelectedIndexChanged phản hồi khi chọn loại báo cáo.

+ Tương tự cho các form KhachHang.cs, SanPham.cs, PhieuNhapHang.cs, HoaDon.cs.

- Kỹ thuật: Mô hình lập trình dựa trên sự kiện (event-driven programming) sử dụng EventArgs để quản lý hành động người dùng.

- Tích hợp 3 tầng:

+ Giao diện gọi phương thức từ BLL.

+ BLL chuyển NhanVienDTO đến DAL để lưu vào cơ sở dữ liệu.

- Mô tả thuật toán:

+ Đăng ký sự kiện cho các thành phần giao diện (button, combobox, datagridview).

+ Khi sự kiện kích hoạt, gọi phương thức BLL tương ứng, truyền DTO.

+ Cập nhật giao diện dựa trên kết quả từ BLL.

**Liên kết dữ liệu (Data Binding)**

- Áp dụng:

+ Sử dụng dgvNhanVien.DataSource trong NhanVien.cs và dgvBaoCao.DataSource trong BaoCao.cs để liên kết dữ liệu từ List<NhanVienDTO> hoặc DataSet.

+ Áp dụng cho các danh mục (KhachHang.cs, SanPham.cs) và chức năng hóa đơn, phiếu nhập.

- Kỹ thuật: Kỹ thuật liên kết dữ liệu (data binding) tự động đồng bộ dữ liệu và giao diện.

- Tích hợp 3 tầng:

+ Giao diện lấy danh sách List<DTO> từ BLL

+ BLL gọi DAL để truy vấn dữ liệu từ cơ sở dữ liệu.

- Mô tả thuật toán:

+ Gọi BLL để lấy danh sách DTO.

+ Gán danh sách vào DataGridView.DataSource.

+ Tự động cập nhật giao diện khi dữ liệu thay đổi.

### 2.2.2 Thuật toán xử lý logic kinh doanh (Business Logic Algorithms)

**Kiểm tra tính hợp lệ (Validation Logic)**

- Áp dụng:

+ Phương thức kiểm tra các trường hợp bắt buộc trong “nhân viên” kiểm tra các trường bắt buộc.

+ Áp dụng tương tự cho “Sản Phẩm, Khách Hàng, Phiếu Nhập Hàng, Hóa Đơn”.

- Kỹ thuật: Sử dụng IF-ELSE và string.IsNullOrEmpty để đảm bảo dữ liệu hợp lệ.

- Tích hợp 3 tầng:

+ Logic kiểm tra được thực hiện tại BLL.

+ Giao diện gọi BLL trước khi gửi DTO đến DAL.

- Mô tả thuật toán:

+ Kiểm tra các trường bắt buộc trong DTO.

+ Nếu lỗi, trả về thông báo qua MessageBox và dừng.

+ Nếu hợp lệ, tiếp tục xử lý (gửi đến DAL).

**Tạo mã tự động (Auto-Generation)**

- Áp dụng:

+ Phương thức “Thiết lập mã nhân viên mới” trong “Nhân viên” để tạo mã.

+ Tương tự cho sản phẩm, khách hàng, phiếu nhập, hóa đơn, mã khuyến mãi.

- Kỹ thuật: Thuật toán tăng dần kết hợp định dạng chuỗi dựa trên ngày giờ hoặc số thứ tự.

- Tích hợp 3 tầng:

+ BLL tạo mã và kiểm tra trùng lặp bằng cách gọi DAL.

+ DTO chứa mã được tạo để truyền giữa các tầng.

- Mô tả thuật toán:

+ Lấy thời gian hiện tại hoặc số thứ tự lớn nhất từ DAL.

+ Tạo mã theo định dạng.

+ Kiểm tra trùng lặp qua DAL; nếu trùng, tăng số thứ tự và thử lại.

**Quản lý trạng thái (State Management)**

- Áp dụng:

+ Trong NhanVien.cs, phương thức XoaCacTruong xóa trắng các trường nhập liệu, và txtMaNhanVien.Enabled = false vô hiệu hóa mã nhân viên.

+ Áp dụng cho HoaDon.cs, PhieuNhapHang.cs để vô hiệu hóa mã hóa đơn, mã phiếu nhập.

- Kỹ thuật: Quản lý trạng thái giao diện để ngăn lỗi người dùng.

- Tích hợp 3 tầng: Giao diện cập nhật trạng thái dựa trên kết quả từ BLL.

- Mô tả thuật toán:

+ Sau thao tác, đặt lại các trường nhập liệu về mặc định.

+ Vô hiệu hóa các trường không được sửa (ví dụ: mã nhân viên).

+ Cập nhật giao diện.

### 2.2.3 Thuật toán quản lý danh mục (Catalog Management Algorithms)

**Quản lý danh mục**

- Áp dụng: Các form NhanVien.cs, KhachHang.cs, SanPham.cs, ChucVu.cs, ChatLieu.cs, ThuongHieu.cs, NhaCungCap.cs, LoaiSanPham.cs, MaKhuyenMai.cs xử lý Thêm, Sửa, Xóa, Tìm kiếm.

- Kỹ thuật: Sử dụng sự kiện, liên kết dữ liệu, kiểm tra hợp lệ, tạo mã tự động.

- Tích hợp 3 tầng:

+ Giao diện gọi BLL (ví dụ: \_nhanVienBLL.AddNhanVien(nhanVienDTO)).

+ BLL xử lý logic và gọi DAL để truy vấn/lưu dữ liệu.

+ DTO truyền dữ liệu giữa các tầng (ví dụ: NhanVienDTO, SanPhamDTO).

**Mô tả thuật toán:**

+ Tải danh sách: BLL gọi DAL để lấy List<DTO>, gán vào DataGridView.

+ Thêm mới: Kiểm tra hợp lệ tại BLL, gửi DTO đến DAL để lưu, cập nhật giao diện.

+ Sửa: Lấy DTO từ dòng chọn, kiểm tra hợp lệ, gửi đến DAL, làm mới giao diện.

+ Xóa: Xác nhận xóa, gọi BLL để xóa qua DAL, cập nhật giao diện.

+ Tìm kiếm: Nhập từ khóa, BLL gọi DAL để tìm, hiển thị kết quả.

### 2.2.4 Thuật toán quản lý hệ thống (System Management Algorithms)

**Tài khoản và phân quyền**

- Áp dụng: Form DangNhap.cs xử lý đăng nhập, TaiKhoan.cs quản lý tài khoản và phân quyền (Quản lý, Nhân viên).

- Kỹ thuật: Xác thực và phân quyền dựa trên vai trò.

- Tích hợp 3 tầng:

+ Giao diện gọi taiKhoanBLL.Authenticate (taiKhoanDTO).

+ BLL kiểm tra logic và gọi\_taiKhoan

+ DAL.CheckLogin().

- Mô tả thuật toán:

+ Đăng nhập: Nhập thông tin, BLL xác thực qua DAL, lưu phiên nếu hợp lệ.

+ Phân quyền: Dựa trên vai trò trong TaiKhoanDTO, bật/tắt chức năng.

**Đăng xuất**

- Áp dụng: Sự kiện btnDangXuat\_Click trong MainForm.cs.

- Kỹ thuật: Quản lý trạng thái phiên.

- Mô tả thuật toán:

+ Xóa thông tin phiên.

+ Đóng form chính, mở form đăng nhập.

### 2.2.5 Thuật toán nhập hàng (Import Management Algorithms)

**Lập phiếu nhập hàng**

- Áp dụng: Form PhieuNhapHang.cs xử lý phiếu nhập.

- Kỹ thuật: Sử dụng DataGridView, kiểm tra hợp lệ, cập nhật tồn kho.

- Tích hợp 3 tầng:

+ Giao diện gửi PhieuNhapDTO và List<ChiTietPhieuNhapDTO> đến BLL.

+ BLL gọi DAL để lưu và cập nhật tồn kho.

- Mô tả thuật toán:

+ Chọn nhà cung cấp, tạo mã phiếu.

+ Thêm sản phẩm vào DataGridView.

+ Kiểm tra hợp lệ tại BLL, lưu qua DAL.

+ Cập nhật tồn kho.

### 2.2.6 Thuật toán hóa đơn (Invoice Management Algorithms)

**Lập hóa đơn**

- Áp dụng: Form HoaDon.cs xử lý hóa đơn.

- Kỹ thuật: Tính tổng, áp dụng khuyến mãi, cập nhật tồn kho.

- Tích hợp 3 tầng:

+ Giao diện gửi HoaDonDTO và List<ChiTietHoaDonDTO> đến BLL.

+ BLL xử lý logic (tính giảm giá) và gọi DAL.

- Mô tả thuật toán:

+ Chọn khách hàng, tạo mã hóa đơn.

+ Thêm sản phẩm vào DataGridView.

+ Kiểm tra hợp lệ, tính tổng tiền tại BLL.

+ Lưu qua DAL, cập nhật tồn kho.

### 2.2.7 Thuật toán báo cáo (Reporting Algorithms)

**Báo cáo doanh thu và thống kê**

- Áp dụng: Form BaoCao.cs xử lý báo cáo.

- Kỹ thuật: Sử dụng form con để nhập tham số, hiển thị trên DataGridView.

- Tích hợp 3 tầng:

+ Giao diện gọi \_baoCaoBLL.GetDoanhThu(startDate, endDate).

+ BLL gọi DAL để lấy dữ liệu từ stored procedure.

- Mô tả thuật toán:

+ Mở form con để nhập thời gian.

+ BLL gọi DAL để lấy List<BaoCaoDTO>.

+ Hiển thị trên dgvBaoCao.

### 2.2.8 Thuật toán xử lý lỗi (Error Handling Algorithms)

**Xử lý ngoại lệ**

- Áp dụng: Tất cả form sử dụng try-catch, ghi log qua System.Diagnostics.Debug.WriteLine.

- Kỹ thuật: Quản lý lỗi tập trung.

- Tích hợp 3 tầng: Lỗi từ DAL được BLL xử lý và trả về giao diện.

- Mô tả thuật toán:

+ Bọc mã trong try.

+ Trong catch, ghi log và hiển thị thông báo.

### 2.2.9 Thuật toán xử lý giao diện con (Sub-Form Handling Algorithms)

**Quản lý cửa sổ con**

- Áp dụng: Form con (FormNhapNgay, FormNhapThangNam, FormNhapNam) trong BaoCao.cs.

- Kỹ thuật: Sử dụng ShowDialog(), NumericUpDown, DateTimePicker.

- Mô tả thuật toán:

+ Mở form con, chờ kết quả (DialogResult.OK).

+ Lấy dữ liệu và xử lý.

### 2.2.10 Thuật toán xử lý chuỗi (String Processing Algorithms)

**Định dạng chuỗi**

- Áp dụng: Phương thức TaoNhanVienTuInput trong NhanVien.cs dùng Trim().

- Kỹ thuật: Xử lý chuỗi cơ bản.

- Tích hợp 3 tầng: BLL chuẩn hóa chuỗi trước khi gửi đến DAL.

- Mô tả thuật toán:

+ Loại bỏ khoảng trắng thừa.

+ Kiểm tra định dạng chuỗi.

# CHƯƠNG 3 PHÂN TÍCH HỆ THỐNG

## 3.1 XÂY DỰNG CƠ SỞ DỮ LIỆU

### 3.1.1 Tổng quan

Cơ sở dữ liệu là nền tảng của hệ thống "Quản lý phần mềm cửa hàng local brand", được thiết kế trên **SQL Server** để lưu trữ và quản lý dữ liệu một cách hiệu quả, hỗ trợ các chức năng như quản lý danh mục (Nhân viên, Khách hàng, Sản phẩm, Chức vụ, Chất liệu, Thương hiệu, Nhà cung cấp, Loại sản phẩm, Khuyến mãi), tài khoản và phân quyền, nhập hàng, hóa đơn, báo cáo doanh thu/thống kê, và đăng xuất. Cơ sở dữ liệu sử dụng mô hình quan hệ (Relational Database Model), được xây dựng bằng ngôn ngữ SQL với các bảng, ràng buộc, chỉ mục, và trigger để đảm bảo tính toàn vẹn, hiệu suất, và tự động hóa các tác vụ.

### 3.1.2 Thiết kế cơ sở dữ liệu

Cơ sở dữ liệu QuanLyCuaHangLocalBrand gồm **17 bảng**, được thiết kế dựa trên các thực thể sau:

**NHAN\_VIEN**: Lưu thông tin nhân viên (mã, họ tên, ngày sinh, giới tính, số điện thoại, mã chức vụ).

**CHUC\_VU**: Lưu danh mục chức vụ (mã, tên, mô tả).

**KHACH\_HANG**: Lưu thông tin khách hàng (mã, tên, ngày sinh, email, số điện thoại, địa chỉ, điểm tích lũy).

**NHA\_CUNG\_CAP**: Lưu thông tin nhà cung cấp (mã, tên, địa chỉ, điện thoại, email).

**SAN\_PHAM**: Lưu thông tin sản phẩm (mã, tên, số lượng tồn, đơn giá bán/nhập, màu sắc, kích cỡ, ngày sản xuất, mô tả, mã thương hiệu, mã loại sản phẩm, mã chất liệu, mã nhà cung cấp).

**HOA\_DON\_BAN** và **CHI\_TIET\_HOA\_DON\_BAN**: Quản lý hóa đơn bán (mã, ngày lập, tổng tiền, mã nhân viên, mã khuyến mãi, mã khách hàng, chi tiết sản phẩm).

**PHIEU\_NHAP** và **CHI\_TIET\_PHIEU\_NHAP**: Quản lý nhập hàng (mã, ngày nhập, tổng tiền, mã nhà cung cấp, mã nhân viên, chi tiết sản phẩm).

**KHUYEN\_MAI**: Lưu chương trình khuyến mãi (mã, tên, phần trăm, ngày bắt đầu/kết thúc, mô tả).

**LOAI\_SAN\_PHAM**: Lưu danh mục loại sản phẩm (mã, tên, mô tả).

**TAI\_KHOAN**: Quản lý tài khoản (tên đăng nhập, mật khẩu, quyền truy cập, mã chức vụ, mã nhân viên).

**THUONG\_HIEU**: Lưu danh mục thương hiệu (mã, tên, mô tả).

**CHAT\_LIEU**: Lưu danh mục chất liệu (mã, tên, mô tả).

**HINH\_ANH**: Lưu hình ảnh sản phẩm (mã, mã sản phẩm, đường dẫn, mô tả, loại hình ảnh).

### 3.1.3 Ràng buộc toàn vẹn

**Khóa chính (Primary Key)**: Đảm bảo mỗi bản ghi duy nhất (ví dụ: MaNhanVien, MaHoaDon).

**Khóa ngoại (Foreign Key)**: Liên kết các bảng (ví dụ: MaChucVu trong NHAN\_VIEN tham chiếu CHUC\_VU).

**Ràng buộc CHECK**: Kiểm tra dữ liệu hợp lệ, như SoLuongTon >= 0, PhanTramKhuyenMai từ 0-100, NgayKetThuc >= NgayBatDau.

**UNIQUE**: Ngăn trùng lặp (ví dụ: SoDienThoai, Email, TenDangNhap).

**ON DELETE CASCADE/NO ACTION**: Quản lý xóa dữ liệu liên quan (ví dụ: xóa HOA\_DON\_BAN sẽ xóa CHI\_TIET\_HOA\_DON\_BAN; không xóa SAN\_PHAM nếu còn trong CHI\_TIET\_HOA\_DON\_BAN).

### 3.1.4 Tối ưu hóa hiệu suất

**Chỉ mục (Index)**: Tạo NONCLUSTERED INDEX trên các cột thường truy vấn (MaSP, MaHoaDon, MaPhieuNhap) để tăng tốc độ truy vấn

**Trigger**: Tự động hóa tác vụ:

Tính ThanhTien trong, CHI\_TIET\_HOA\_DON\_BAN, CHI\_TIET\_PHIEU\_NHAP.

Cập nhật SoLuongTon trong SAN\_PHAM khi nhập/xuất kho, kiểm tra tồn kho trước khi bán.

Cập nhật TongTien cho HOA\_DON\_BAN, PHIEU\_NHAP, áp dụng khuyến mãi.

Cộng DiemTichLuy cho khách hàng dựa trên tổng tiền hóa đơn.

Kiểm tra hiệu lực khuyến mãi trong HOA\_DON\_BAN.

**Stored Procedure**: Sử dụng để xử lý truy vấn phức tạp (ví dụ: báo cáo doanh thu), tăng hiệu suất và bảo mật.

### 3.1.5 Tích hợp với mô hình 3 tầng

**Presentation Layer**: Giao diện WinForms gọi BLL để xử lý dữ liệu (ví dụ: \_nhanVienBLL.AddNhanVien(nhanVienDTO)), hiển thị dữ liệu qua DataGridView.

**BLL**: Xử lý logic kinh doanh (kiểm tra hợp lệ, tạo mã tự động), truyền DTO đến DAL.

**DAL**: Thực thi truy vấn SQL hoặc stored procedure để tương tác với cơ sở dữ liệu, trả về List<DTO> hoặc DataSet.

**DTO**: Truyền dữ liệu giữa các tầng (ví dụ: NhanVienDTO, HoaDonDTO).

### 3.1.6 Giới thiệu phần mã SQL

**Các bảng chính**

**CHUC\_VU**: Lưu thông tin chức vụ (Mã chức vụ, Tên chức vụ, Mô tả).

**THUONG\_HIEU**: Lưu thông tin thương hiệu (Mã thương hiệu, Tên thương hiệu, Mô tả).

**LOAI\_SAN\_PHAM**: Lưu loại sản phẩm (Mã loại, Tên loại, Mô tả).

**CHAT\_LIEU**: Lưu thông tin chất liệu (Mã chất liệu, Tên chất liệu, Mô tả).

**NHA\_CUNG\_CAP**: Lưu thông tin nhà cung cấp (Mã NCC, Tên, Địa chỉ, Điện thoại, Email).

**KHUYEN\_MAI**: Lưu thông tin khuyến mãi (Mã khuyến mãi, Tên, Phần trăm, Ngày bắt đầu, Ngày kết thúc, Mô tả).

**KHACH\_HANG**: Lưu thông tin khách hàng (Mã KH, Tên, Ngày sinh, Email, Số điện thoại, Địa chỉ).

**NHAN\_VIEN**: Lưu thông tin nhân viên (Mã NV, Họ tên, Ngày sinh, Giới tính, Số điện thoại, Mã chức vụ).

**TAI\_KHOAN**: Lưu tài khoản nhân viên (Mã NV, Tên đăng nhập, Mật khẩu, Quyền truy cập: Admin/User/Manager).

**SAN\_PHAM**: Lưu thông tin sản phẩm (Mã SP, Tên, Số lượng tồn, Giá bán, Giá nhập, Màu sắc, Kích cỡ, Ngày sản xuất, Mô tả, Mã thương hiệu, Mã loại SP, Mã chất liệu, Mã NCC).

**HOA\_DON\_BAN**: Lưu hóa đơn bán (Mã hóa đơn, Ngày lập, Tổng tiền, Mã NV, Mã khuyến mãi, Mã KH, Địa chỉ, Ghi chú).

**CHI\_TIET\_HOA\_DON\_BAN**: Lưu chi tiết hóa đơn bán (Mã hóa đơn, Mã SP, Số lượng bán, Đơn giá, Thành tiền).

**PHIEU\_NHAP**: Lưu phiếu nhập (Mã phiếu nhập, Ngày lập, Tổng tiền, Mã NCC, Mã NV, Địa chỉ, Ghi chú).

**CHI\_TIET\_PHIEU\_NHAP**: Lưu chi tiết phiếu nhập (Mã phiếu nhập, Mã SP, Số lượng, Đơn giá, Thành tiền).

**HINH\_ANH**: Lưu hình ảnh sản phẩm (Mã hình ảnh, Mã SP, Đường dẫn hình).

**Các ràng buộc**

**Khóa chính**: Mỗi bảng có khóa chính duy nhất (ví dụ: MaChucVu, MaSP, MaHoaDon...).

**Khóa ngoại**: Liên kết giữa các bảng (ví dụ: MaChucVu trong NHAN\_VIEN liên kết với CHUC\_VU).

**Ràng buộc dữ liệu**:

- Email hợp lệ (LIKE '%@%.%').

- Số lượng tồn, giá bán, giá nhập >= 0.

- Phần trăm khuyến mãi từ 0-100.

- Ngày kết thúc khuyến mãi >= Ngày bắt đầu.

- Ngày sinh, ngày lập hóa đơn, phiếu nhập <= ngày hiện tại.

- Giới tính: Nam, Nữ, Khác.

- Quyền truy cập: Admin, User, Manager.

**Chỉ mục (Index)**

Tạo các chỉ mục không phân cụm (NONCLUSTERED INDEX) để tối ưu truy vấn trên các cột thường xuyên tra cứu:

- MaSP (SAN\_PHAM)

- MaThuongHieu (SAN\_PHAM)

- MaHoaDon (CHI\_TIET\_HOA\_DON\_BAN)

- MaPhieuNhap (CHI\_TIET\_PHIEU\_NHAP)

- MaSP (HINH\_ANH)

**Trigger**

**TRG\_CheckThanhTien\_ChiTietHoaDonBan**: Tự động tính ThanhTien (Số lượng bán \* Đơn giá) trong CHI\_TIET\_HOA\_DON\_BAN.

**TRG\_CheckThanhTien\_ChiTietPhieuNhap**: Tự động tính ThanhTien (Số lượng \* Đơn giá) trong CHI\_TIET\_PHIEU\_NHAP.

**TRG\_UpdateSoLuongTon\_NhapKho**: Cập nhật SoLuongTon trong SAN\_PHAM khi thêm/sửa/xóa CHI\_TIET\_PHIEU\_NHAP.

**TRG\_UpdateSoLuongTon\_XuatKho**: Kiểm tra và cập nhật SoLuongTon khi thêm/sửa/xóa CHI\_TIET\_HOA\_DON\_BAN, báo lỗi nếu số lượng tồn không đủ.

**TRG\_UpdateTongTien\_HoaDonBan**: Cập nhật TongTien trong HOA\_DON\_BAN, tính toán dựa trên chi tiết hóa đơn và áp dụng khuyến mãi nếu có.

**TRG\_UpdateTongTien\_PhieuNhap**: Cập nhật TongTien trong PHIEU\_NHAP dựa trên chi tiết phiếu nhập.

**TRG\_CheckKhuyenMai**: Kiểm tra khuyến mãi còn hiệu lực khi thêm/sửa HOA\_DON\_BAN.

## 3.2 XÂY DỰNG PHẦN MỀM

### 3.2.1 Tổng quan

Phần mềm "Quản lý cửa hàng local brand" được phát triển trên nền tảng **C#** với **WinForms**, sử dụng mô hình 3 tầng (Presentation, Business Logic Layer - BLL, Data Access Layer - DAL) và **DTO** (Data Transfer Object) để quản lý các chức năng: danh mục (Nhân viên, Khách hàng, Sản phẩm, Chức vụ, Chất liệu, Thương hiệu, Nhà cung cấp, Loại sản phẩm, Khuyến mãi), tài khoản và phân quyền, nhập hàng, hóa đơn, báo cáo doanh thu/thống kê, và đăng xuất. Phần mềm tích hợp với cơ sở dữ liệu SQL Server (QuanLyCuaHangLocalBrand) để lưu trữ và truy xuất dữ liệu hiệu quả.

### 3.2.2 Công nghệ và công cụ sử dụng

**Ngôn ngữ lập trình**: C# (hướng đối tượng, mạnh mẽ, phù hợp cho ứng dụng desktop).

**Framework**: .NET Framework 4.5 trở lên, hỗ trợ WinForms và ADO.NET.

**Giao diện**: WinForms, cung cấp các thành phần giao diện như DataGridView, Button, TextBox, ComboBox.

**Cơ sở dữ liệu**: SQL Server, với 17 bảng, ràng buộc, chỉ mục, và trigger để đảm bảo toàn vẹn dữ liệu.

**Công nghệ truy xuất dữ liệu**: ADO.NET, sử dụng các lớp SqlConnection, SqlCommand, SqlDataAdapter để tương tác với cơ sở dữ liệu.

**Môi trường phát triển**: Visual Studio (hỗ trợ thiết kế giao diện, lập trình, và debug).

### 3.2.3 Cấu trúc phần mềm (Mô hình 3 tầng)

**Presentation Layer (Tầng giao diện):**

Xây dựng các form WinForms như NhanVien.cs, KhachHang.cs, SanPham.cs, HoaDon.cs, PhieuNhap.cs, BaoCao.cs, DangNhap.cs.

Sử dụng lập trình dựa trên sự kiện (event-driven programming) để xử lý các thao tác người dùng (thêm, sửa, xóa, tìm kiếm).

Liên kết dữ liệu (DataGridView.DataSource) để hiển thị danh sách từ List<DTO> hoặc DataSet.

**Business Logic Layer (BLL):**

Chứa các lớp như NhanVienBLL, SanPhamBLL, HoaDonBLL để xử lý logic kinh doanh: kiểm tra hợp lệ, tạo mã tự động, tính tổng tiền, áp dụng khuyến mãi, cập nhật điểm tích lũy.

Nhận dữ liệu từ giao diện qua DTO và gọi DAL để truy xuất cơ sở dữ liệu.

**Data Access Layer (DAL):**

Chứa các lớp như NhanVienDAL, SanPhamDAL, HoaDonDAL để thực hiện truy vấn SQL (CRUD) thông qua ADO.NET.

Sử dụng stored procedure và trigger để tối ưu hóa truy vấn và đảm bảo toàn vẹn dữ liệu.

**Data Transfer Object (DTO):**

Các lớp như NhanVienDTO, SanPhamDTO, HoaDonDTO, PhieuNhapDTO để truyền dữ liệu giữa các tầng, đảm bảo tính đóng gói và giảm sự phụ thuộc.

### 3.2.4 Các chức năng chính và triển khai

**Quản lý danh mục:**

Thực thể: Nhân viên, Khách hàng, Sản phẩm, Chức vụ, Chất liệu, Thương hiệu, Nhà cung cấp, Loại sản phẩm, Khuyến mãi.

Triển khai:

- Form như NhanVien.cs, SanPham.cs hiển thị danh sách trên DataGridView, hỗ trợ thêm, sửa, xóa, tìm kiếm.

- BLL kiểm tra hợp lệ (ví dụ: SoDienThoai duy nhất, PhanTramKhuyenMai từ 0-100) trước khi gọi DAL lưu dữ liệu.

- DAL thực thi truy vấn SQL (INSERT, UPDATE, DELETE, SELECT) vào các bảng như NHAN\_VIEN, SAN\_PHAM.

**Tài khoản và phân quyền:**

Thực thể: Tài khoản.

Triển khai:

+ Form DangNhap.cs xác thực người dùng qua \_taiKhoanBLL.Authenticate(taiKhoanDTO).

+ Phân quyền (Admin, User, Manager) giới hạn truy cập chức năng (ví dụ: chỉ Admin sửa danh mục).

+ DAL truy vấn bảng TAI\_KHOAN để kiểm tra TenDangNhap, MatKhau.

**Nhập hàng:**

Thực thể: Phiếu nhập, Chi tiết phiếu nhập.

Triển khai:

+ Form PhieuNhap.cs nhập chi tiết qua DataGridView, tạo mã tự động.

+ BLL kiểm tra số lượng, giá nhập; DAL lưu vào PHIEU\_NHAP, CHI\_TIET\_PHIEU\_NHAP và cập nhật SoLuongTon trong SAN\_PHAM qua trigger.

**Hóa đơn:**

Thực thể: Hóa đơn bán, Chi tiết hóa đơn bán.

Triển khai:

+ Form HoaDon.cs, DatHang.cs nhập chi tiết sản phẩm, tính tổng tiền, áp dụng khuyến mãi (MaKhuyenMai).

+ BLL kiểm tra tồn kho, tính TongTien và DiemTichLuy; DAL lưu vào HOA\_DON\_BAN, CHI\_TIET\_HOA\_DON\_BAN,.

+ Trigger tự động cập nhật ThanhTien, TongTien, và SoLuongTon.

**Báo cáo doanh thu/thống kê:**

Triển khai:

+ Form BaoCao.cs sử dụng form con (FormNhapNgay, FormNhapThangNam) để nhập tham số thời gian.

+ BLL gọi DAL thực thi stored procedure, trả về List<BaoCaoDTO> để hiển thị trên DataGridView.

**Đăng xuất:**

Triển khai: Sự kiện btnDangXuat\_Click trong MainForm.cs xóa phiên, mở form DangNhap.cs.

**Quản lý hình ảnh sản phẩm:**

Thực thể: Hình ảnh.

Triển khai:

+ Form SanPham.cs hiển thị đường dẫn hình ảnh từ bảng HINH\_ANH.

+ DAL lưu và truy xuất DuongDanHinh, liên kết với MaSP.

### 3.2.5 Xử lý lỗi và tối ưu hóa

**Xử lý lỗi:**

Sử dụng try-catch trong BLL và DAL để bắt ngoại lệ, ghi log qua System.Diagnostics.Debug.WriteLine, và hiển thị thông báo qua MessageBox.

Trigger kiểm tra tồn kho và hiệu lực khuyến mãi để ngăn lỗi dữ liệu.

**Tối ưu hóa:**

Tạo mã tự động: Tạo mã duy nhất (ví dụ: NV202506091920) tại BLL, kiểm tra trùng qua DAL.

Chuẩn hóa chuỗi: Sử dụng Trim(), ToUpper() tại BLL để xử lý đầu vào.

Giao diện con: Sử dụng ShowDialog() trong form con (FormNhapNgay) để nhập tham số báo cáo.

Liên kết dữ liệu: Tự động đồng bộ dữ liệu với giao diện qua DataGridView.DataSource.

### 3.2.6 Tích hợp với cơ sở dữ liệu

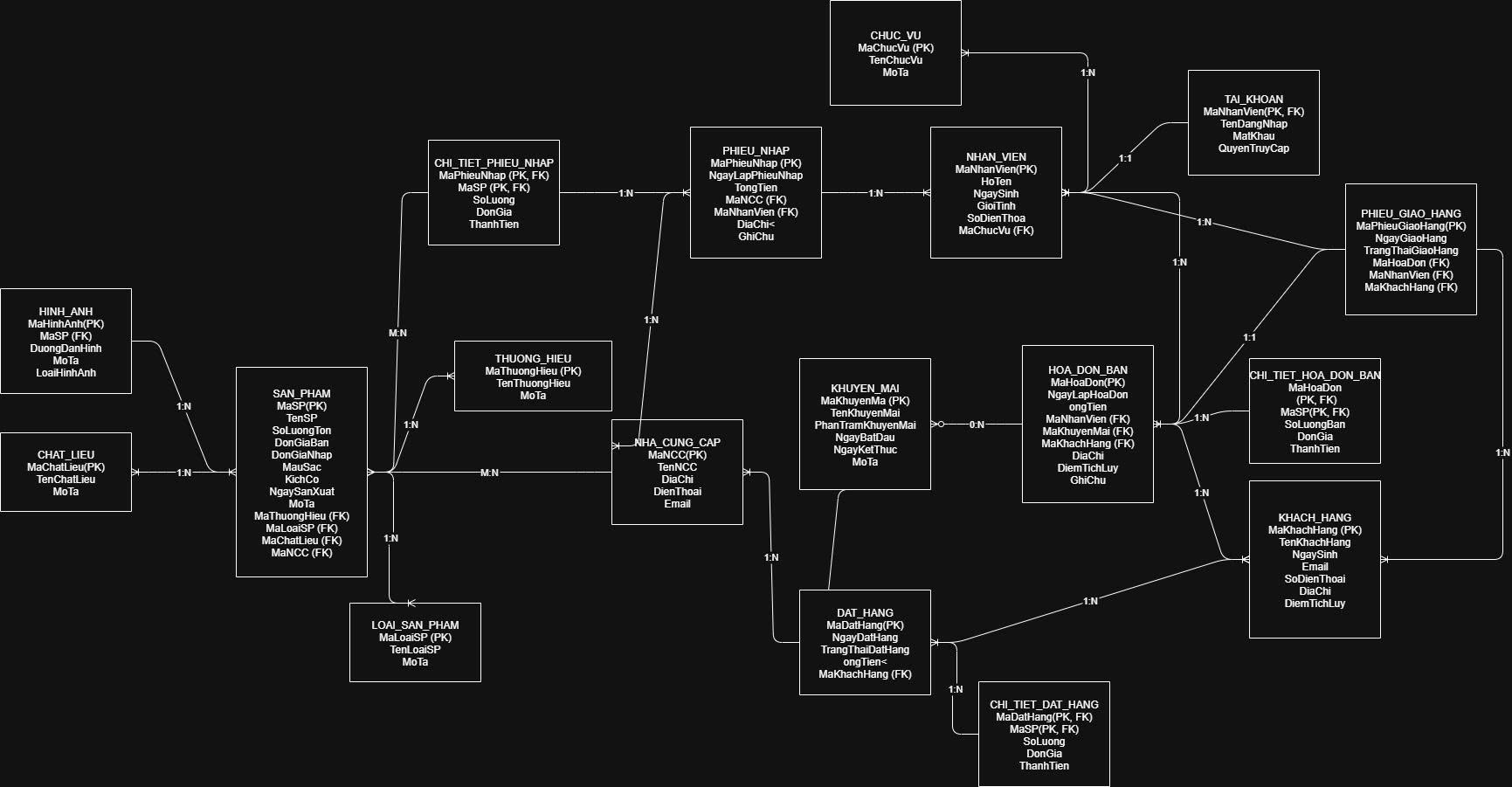
Phần mềm tích hợp với cơ sở dữ liệu QuanLyCuaHangLocalBrand (như NHAN\_VIEN, SAN\_PHAM, HOA\_DON\_BAN) thông qua ADO.NET.

DAL thực thi truy vấn SQL và stored procedure, sử dụng trigger để tự động cập nhật SoLuongTon, ThanhTien, TongTien, DiemTichLuy.

Chỉ mục (NONCLUSTERED INDEX) trên các cột như MaSP, MaHoaDon tăng tốc độ truy vấn.

## 3.3 MÔ HÌNH HÓA YÊU CẦU PHẦN MỀM

### 3.3.1 Mô hình dữ liệu mức quan niệm (CDM)



Hình 3. Mô hình dữ liệu mức qua niệm (CDM)

### 3.3.2 Tổng quan

Mô hình dữ liệu mức quan niệm (CDM) là biểu diễn trừu tượng của cơ sở dữ liệu hệ thống "Quản lý phần mềm cửa hàng local brand", được xây dựng để mô tả các thực thể chính, thuộc tính, và mối quan hệ giữa chúng mà không đi vào chi tiết kỹ thuật triển khai. CDM được thiết kế dựa trên mô hình thực thể-liên kết (Entity-Relationship Model - ERM), phục vụ như nền tảng cho việc thiết kế cơ sở dữ liệu quan hệ (QuanLyCuaHangLocalBrand) trên SQL Server. Mô hình này phản ánh các yêu cầu kinh doanh của cửa hàng local brand, bao gồm quản lý danh mục, tài khoản, giao dịch (nhập hàng, hóa đơn,).

### 3.3.3 Các thực thể và thuộc tính

**NHAN\_VIEN**:

Thuộc tính: Mã nhân viên (PK), Họ tên, Ngày sinh, Giới tính, Số điện thoại, Mã chức vụ.

Ý nghĩa: Lưu thông tin nhân viên làm việc tại cửa hàng.

**CHUC\_VU**:

Thuộc tính: Mã chức vụ (PK), Tên chức vụ, Mô tả.

Ý nghĩa: Định nghĩa các vai trò của nhân viên.

**KHACH\_HANG**:

Thuộc tính: Mã khách hàng (PK), Tên khách hàng, Ngày sinh, Email, Số điện thoại, Địa chỉ, Điểm tích lũy.

Ý nghĩa: Quản lý thông tin khách hàng và điểm thưởng.

**NHA\_CUNG\_CAP**:

Thuộc tính: Mã NCC (PK), Tên NCC, Địa chỉ, Điện thoại, Email.

Ý nghĩa: Lưu thông tin nhà cung cấp hàng hóa.

**SAN\_PHAM**:

Thuộc tính: Mã SP (PK), Tên SP, Số lượng tồn, Đơn giá bán, Đơn giá nhập, Màu sắc, Kích cỡ, Ngày sản xuất, Mô tả, Mã thương hiệu, Mã loại sản phẩm, Mã chất liệu, Mã nhà cung cấp.

Ý nghĩa: Quản lý thông tin sản phẩm trong cửa hàng.

**HOA\_DON\_BAN**:

Thuộc tính: Mã hóa đơn (PK), Ngày lập hóa đơn, Tổng tiền, Mã nhân viên, Mã khuyến mãi, Mã khách hàng.

Ý nghĩa: Lưu thông tin hóa đơn bán hàng.

**CHI\_TIET\_HOA\_DON\_BAN**:

Thuộc tính: Mã hóa đơn, Mã sản phẩm (PK), Số lượng bán, Đơn giá, Thành tiền.

Ý nghĩa: Chi tiết các sản phẩm trong hóa đơn.

**PHIEU\_NHAP**:

Thuộc tính: Mã phiếu nhập (PK), Ngày nhập, Tổng tiền, Mã NCC, Mã nhân viên.

Ý nghĩa: Ghi nhận phiếu nhập hàng từ nhà cung cấp.

**CHI\_TIET\_PHIEU\_NHAP**:

Thuộc tính: Mã phiếu nhập, Mã SP (PK), Số lượng, Đơn giá, Thành tiền.

Ý nghĩa: Chi tiết các sản phẩm trong phiếu nhập.

**KHUYEN\_MAI**:

Thuộc tính: Mã khuyến mãi (PK), Tên khuyến mãi, Phần trăm khuyến mãi, Ngày bắt đầu, Ngày kết thúc, Mô tả.

Ý nghĩa: Quản lý các chương trình khuyến mãi.

**LOAI\_SAN\_PHAM**:

Thuộc tính: Mã loại sản phẩm (PK), Tên loại sản phẩm, Mô tả.

Ý nghĩa: Phân loại sản phẩm theo nhóm.

**TAI\_KHOAN**:

Thuộc tính: Mã nhân viên (PK), Tên đăng nhập, Mật khẩu, Quyền truy cập, Mã chức vụ.

Ý nghĩa: Quản lý thông tin đăng nhập và phân quyền.

**THUONG\_HIEU**:

Thuộc tính: Mã thương hiệu (PK), Tên thương hiệu, Mô tả.

Ý nghĩa: Lưu danh mục thương hiệu sản phẩm.

**CHAT\_LIEU**:

Thuộc tính: Mã chất liệu (PK), Tên chất liệu, Mô tả.

Ý nghĩa: Lưu danh mục chất liệu sản phẩm.

### 3.3.4 Mối quan hệ giữa các thực thể

**NHAN\_VIEN - CHUC\_VU**:

Quan hệ 1-1 (một nhân viên thuộc một chức vụ).

Khóa ngoại: MaChucVu trong NHAN\_VIEN tham chiếu CHUC\_VU.

**NHAN\_VIEN - TAI\_KHOAN**:

Quan hệ 1-1 (một nhân viên có một tài khoản).

Khóa ngoại: MaNhanVien trong TAI\_KHOAN tham chiếu NHAN\_VIEN.

**NHAN\_VIEN/HOA\_DON\_BAN/PHIEU\_NHAP**:

Quan hệ 1-nhiều (một nhân viên lập nhiều hóa đơn, phiếu nhập).

Khóa ngoại: MaNhanVien trong các bảng này tham chiếu NHAN\_VIEN**/HOA\_DON\_BAN**:

Quan hệ 1-nhiều (một khách hàng thực hiện nhiều đơn, hóa đơn).

Khóa ngoại: MaKhachHang trong các bảng này tham chiếu KHACH\_HANG.

**NHA\_CUNG\_CAP - SAN\_PHAM/PHIEU\_NHAP**:

Quan hệ 1-nhiều (một nhà cung cấp cung cấp nhiều sản phẩm hoặc phiếu nhập).

Khóa ngoại: MaNCC trong SAN\_PHAM và PHIEU\_NHAP tham chiếu NHA\_CUNG\_CAP.

**SAN\_PHAM /HOA\_DON\_BAN/PHIEU NHAP (qua chi tiết)**:

Quan hệ nhiều-nhiều (một sản phẩm có thể xuất hiện trong nhiều đơn, hóa đơn, hoặc phiếu nhập).

Quan hệ trung gian:CHI\_TIET\_HOA\_DON\_BAN, CHI\_TIET\_PHIEU\_NHAP với khóa ngoại MaDatHang, MaHoaDon, MaPhieuNhap, và MaSP.

**KHUYEN\_MAI - HOA\_DON\_BAN**:

Quan hệ 1-nhiều (một khuyến mãi áp dụng cho nhiều hóa đơn).

Khóa ngoại: MaKhuyenMai trong HOA\_DON\_BAN tham chiếu KHUYEN\_MAI.

**THUONG\_HIEU/LOAI\_SAN\_PHAM/CHAT\_LIEU - SAN\_PHAM**: Quan hệ 1-nhiều (một thương hiệu, loại sản phẩm, hoặc chất liệu thuộc về nhiều sản phẩm).

Khóa ngoại: MaThuongHieu, MaLoaiSP, MaChatLieu trong SAN\_PHAM tham chiếu các bảng tương ứng.

### 3.3.5 Mô hình dữ liệu mức Logic (LDM)



Hình 3. Mô hình dữ liệu mức Logic (LDM)

Mô hình dữ liệu mức Logic (LDM) là bước chuyển đổi từ Mô hình dữ liệu mức quan niệm (CDM) sang mô hình quan hệ, cung cấp cấu trúc chi tiết của cơ sở dữ liệu QuanLyCuaHangLocalBrand trên SQL Server. LDM xác định các bảng, thuộc tính, khóa chính (PK), khóa ngoại (FK), và các ràng buộc toàn vẹn, phản ánh các yêu cầu kinh doanh của hệ thống "Quản lý phần mềm cửa hàng local brand", bao gồm quản lý danh mục, tài khoản, nhập hàng, hóa đơn, và báo cáo.

### 3.3.6 Cấu trúc các bảng và thuộc tính

**CHUC\_VU**: MaChucVu, TenChucVu, MoTa - Lưu danh mục chức vụ

**NHAN\_VIEN**: MaNhanVien, HoTen, NgaySinh, GioiTinh, SoDienThoai, MaChucVu - Lưu thông tin nhân viên

**TAI\_KHOAN**: MaNhanVien, TenDangNhap, MatKhau, QuyenTruyCap, MaChucVu - Quản lý tài khoản

**KHACH\_HANG**: MaKhachHang, TenKhachHang, NgaySinh, Email, SoDienThoai, DiaChi, DiemTichLuy - Lưu thông tin khách hàng

**NHA\_CUNG\_CAP**: MaNCC, TenNCC, DiaChi, DienThoai, Email - Lưu thông tin nhà cung cấp

**THUONG\_HIEU**: MaThuongHieu, TenThuongHieu, MoTa - Lưu danh mục thương hiệu

**LOAI\_SAN\_PHAM**: MaLoaiSP, TenLoaiSP, MoTa - Lưu danh mục loại sản phẩm

**CHAT\_LIEU**: MaChatLieu, TenChatLieu, MoTa - Lưu danh mục chất liệu

**SAN\_PHAM**: MaSP, TenSP, SoLuongTon, DonGiaBan, DonGiaNhap, MauSac, KichCo, NgaySanXuat, MoTa, MaThuongHieu, MaLoaiSP, MaChatLieu, MaNCC - Quản lý sản phẩm

**KHUYEN\_MAI**: MaKhuyenMai, TenKhuyenMai, PhanTramKhuyenMai, NgayBatDau, NgayKetThuc, MoTa - Quản lý khuyến mãi

**HOA\_DON\_BAN**: MaHoaDon, NgayLapHoaDon, TongTien, MaNhanVien, MaKhuyenMai, MaKhachHang - Lưu hóa đơn bán

**CHI\_TIET\_HOA\_DON\_BAN**: MaHoaDon, MaSP, SoLuongBan, DonGia, ThanhTien - Chi tiết hóa đơn

**PHIEU\_NHAP**: MaPhieuNhap, NgayLapPhieuNhap, TongTien, MaNCC, MaNhanVien - Lưu phiếu nhập

**CHI\_TIET\_PHIEU\_NHAP**: MaPhieuNhap, MaSP, SoLuong, DonGia, ThanhTien - Chi tiết phiếu nhập

### 3.3.7 Ràng buộc toàn vẹn

**Khóa chính (PK)**: Đảm bảo mỗi bản ghi duy nhất (ví dụ: MaNhanVien, MaHoaDon).

**Khóa ngoại (FK)**: Duy trì tính toàn vẹn tham chiếu giữa các bảng (ví dụ: MaChucVu trong NHAN\_VIEN tham chiếu CHUC\_VU).

**Ràng buộc CHECK**: Kiểm tra dữ liệu hợp lệ (ví dụ: SoLuongTon >= 0, PhanTramKhuyenMai từ 0-100, NgayKetThuc >= NgayBatDau).

**UNIQUE**: Ngăn trùng lặp (ví dụ: SoDienThoai, Email, TenDangNhap).

**ON DELETE CASCADE/NO ACTION**: Quản lý xóa dữ liệu (ví dụ: xóa HOA\_DON\_BAN sẽ xóa CHI\_TIET\_HOA\_DON\_BAN; không xóa SAN\_PHAM nếu còn trong CHI\_TIET\_HOA\_DON\_BAN).

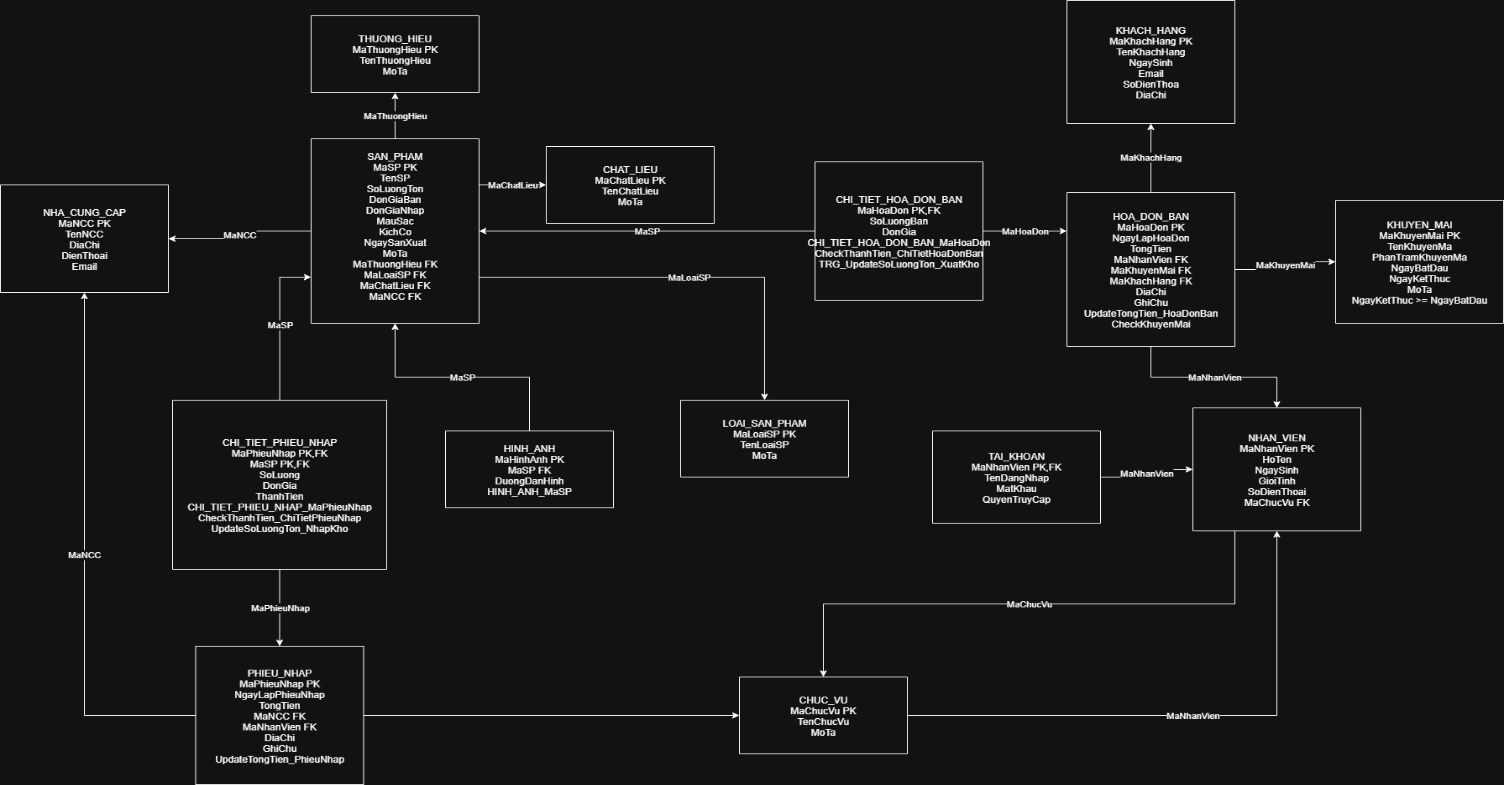
### 3.3.8 Tối ưu hóa

**Chỉ mục (Index)**: Đề xuất tạo NONCLUSTERED INDEX trên các cột thường truy vấn (ví dụ: MaSP, MaHoaDon, MaPhieuNhap) để tăng tốc độ truy vấn.

**Trigger**: Sử dụng để tự động hóa (ví dụ: cập nhật ThanhTien, SoLuongTon, TongTien, DiemTichLuy).

**Chuẩn hóa**: Áp dụng chuẩn 3NF để loại bỏ dư thừa dữ liệu, đảm bảo tính nhất quán.

## 3.4 MÔ HÌNH DỮ LIỆU MỨC VẬT LÝ (PDM)



Hình 3. Mô hình dữ liệu mức vật lý (pdm)

### 3.4.1 Tổng quan

Mô hình dữ liệu mức vật lý (PDM) là bước cuối trong thiết kế cơ sở dữ liệu, chuyển đổi từ LDM thành cấu trúc vật lý trên SQL Server cho QuanLyCuaHangLocalBrand. PDM định nghĩa bảng, kiểu dữ liệu, kích thước trường, chỉ mục, trigger, và tối ưu hóa, hỗ trợ hệ thống "Quản lý phần mềm cửa hàng local brand" trong quản lý danh mục, tài khoản, nhập hàng, hóa đơn, và báo cáo.

### 3.4.2 Cấu trúc cơ sở dữ liệu vật lý

**CHUC\_VU**: MaChucVu, TenChucVu, MoTa - Lưu danh mục chức vụ

**NHAN\_VIEN**: MaNhanVien, HoTen, NgaySinh, GioiTinh, SoDienThoai, MaChucVu - Lưu thông tin nhân viên

**TAI\_KHOAN**: MaNhanVien, TenDangNhap, MatKhau, QuyenTruyCap, MaChucVu - Quản lý tài khoản

**KHACH\_HANG**: MaKhachHang, TenKhachHang, NgaySinh, Email, SoDienThoai, DiaChi, DiemTichLuy - Lưu thông tin khách hàng

**NHA\_CUNG\_CAP**: MaNCC, TenNCC, DiaChi, DienThoai, Email - Lưu thông tin nhà cung cấp

**THUONG\_HIEU**: MaThuongHieu, TenThuongHieu, MoTa - Lưu danh mục thương hiệu

**LOAI\_SAN\_PHAM**: MaLoaiSP, TenLoaiSP, MoTa - Lưu danh mục loại sản phẩm

**CHAT\_LIEU**: MaChatLieu, TenChatLieu, MoTa - Lưu danh mục chất liệu

**SAN\_PHAM**: MaSP, TenSP, SoLuongTon, DonGiaBan, DonGiaNhap, MauSac, KichCo, NgaySanXuat, MoTa, MaThuongHieu, MaLoaiSP, MaChatLieu, MaNCC - Quản lý sản phẩm

**KHUYEN\_MAI**: MaKhuyenMai, TenKhuyenMai, PhanTramKhuyenMai, NgayBatDau, NgayKetThuc, MoTa - Quản lý khuyến mãi

**HOA\_DON\_BAN**: MaHoaDon, NgayLapHoaDon, TongTien, MaNhanVien, MaKhuyenMai, MaKhachHang - Lưu hóa đơn bán

**CHI\_TIET\_HOA\_DON\_BAN**: MaHoaDon, MaSP, SoLuongBan, DonGia, ThanhTien - Chi tiết hóa đơn

**PHIEU\_NHAP**: MaPhieuNhap, NgayLapPhieuNhap, TongTien, MaNCC, MaNhanVien - Lưu phiếu nhập

**CHI\_TIET\_PHIEU\_NHAP**: MaPhieuNhap, MaSP, SoLuong, DonGia, ThanhTien - Chi tiết phiếu nhập

### 3.4.3 Tối ưu hóa hiệu suất

Chỉ mục: NONCLUSTERED INDEX trên MaSP, MaHoaDon, MaPhieuNhap, MaDatHang để tăng tốc độ truy vấn

Trigger: Tự động tính ThanhTien, cập nhật SoLuongTon, TongTien, DiemTichLuy, kiểm tra khuyến mãi

Stored Procedure: Sử dụng để xử lý báo cáo doanh thu/thống kê, tăng hiệu suất

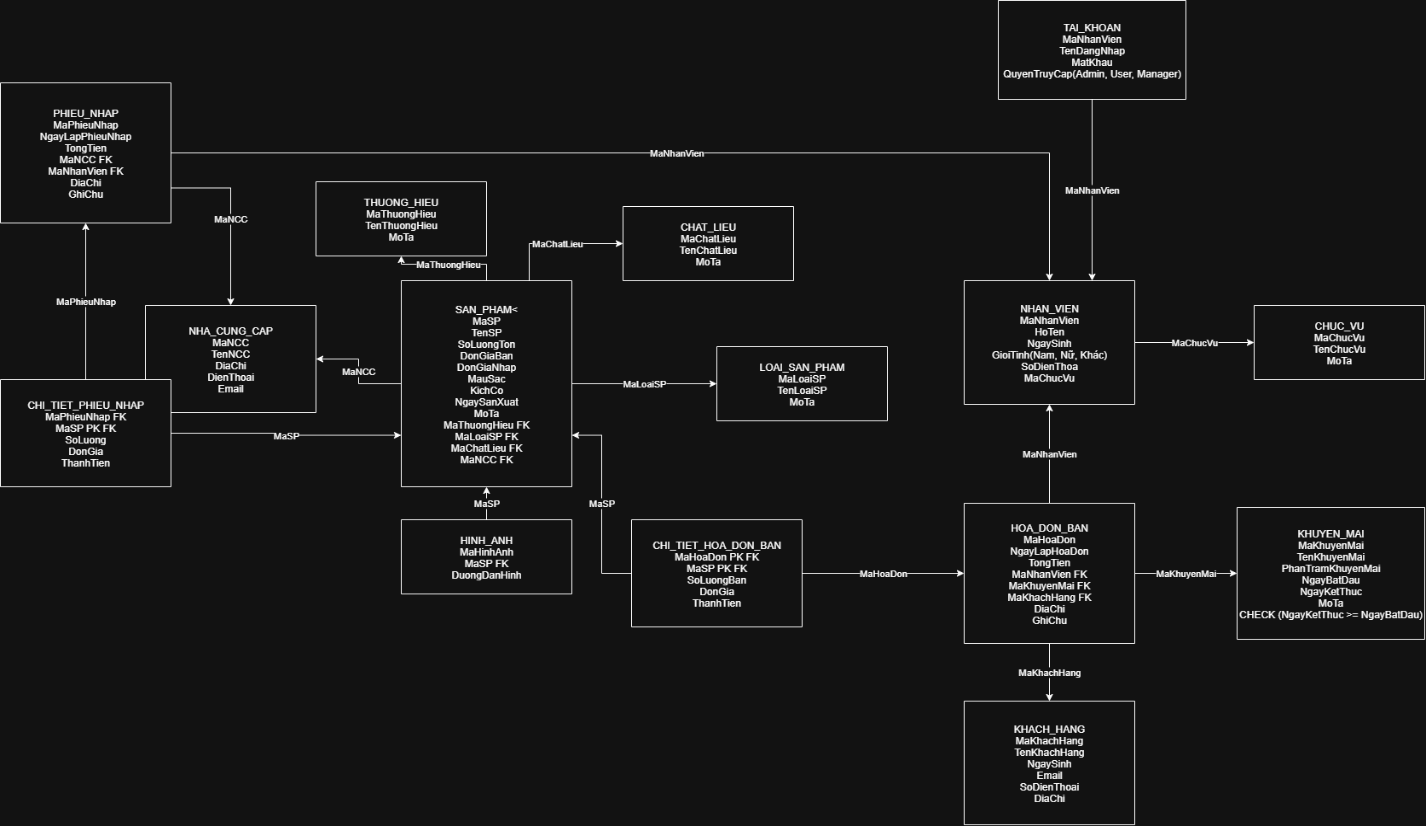
### 3.4.4 Tích hợp với phần mềm

PDM triển khai trên SQL Server, kết nối WinForms qua ADO.NET trong DAL

Bảng NHAN\_VIEN, SAN\_PHAM, HOA\_DON\_BAN truy xuất qua DAL và DTO

Trigger và chỉ mục hỗ trợ tự động hóa, tối ưu hóa truy vấn từ BLL

## 3.5 MÔ HÌNH DỮ LIỆU CƠ SỞ DỮ LIỆU (RDM)



Hình 3. Mô hình dữ liệu cơ sở dữ liệu (rdm)

### 3.5.1 Tổng quan

Mô hình dữ liệu mức vật lý (PDM) là bước cuối trong thiết kế cơ sở dữ liệu, chuyển đổi từ LDM thành cấu trúc vật lý trên SQL Server cho QuanLyCuaHangLocalBrand. PDM định nghĩa bảng, kiểu dữ liệu, kích thước trường, chỉ mục, trigger, và tối ưu hóa, hỗ trợ hệ thống "Quản lý phần mềm cửa hàng local brand" trong quản lý danh mục, tài khoản, nhập hàng, hóa đơn, và báo cáo.

### 3.5.2. Nguyên tắc cơ bản

Bảng: Tập hợp hàng và cột, đại diện thực thể hoặc mối quan hệ

Khóa chính: Cột duy nhất xác định hàng

Khóa ngoại: Liên kết bảng, duy trì tính toàn vẹn

Ràng buộc toàn vẹn: Bao gồm khóa, giá trị, duy nhất

Chuẩn hóa: Áp dụng 3NF để loại bỏ phụ thuộc truyền đạt

### 3.5.3 Cấu trúc các bảng quan hệ

**CHUC\_VU**: MaChucVu, TenChucVu, MoTa - Lưu danh mục chức vụ

**NHAN\_VIEN**: MaNhanVien, HoTen, NgaySinh, GioiTinh, SoDienThoai, MaChucVu - Lưu thông tin nhân viên

**TAI\_KHOAN**: MaNhanVien, TenDangNhap, MatKhau, QuyenTruyCap, MaChucVu - Quản lý tài khoản

**KHACH\_HANG**: MaKhachHang, TenKhachHang, NgaySinh, Email, SoDienThoai, DiaChi, DiemTichLuy - Lưu thông tin khách hàng

**NHA\_CUNG\_CAP**: MaNCC, TenNCC, DiaChi, DienThoai, Email - Lưu thông tin nhà cung cấp

**THUONG\_HIEU**: MaThuongHieu, TenThuongHieu, MoTa - Lưu danh mục thương hiệu

**LOAI\_SAN\_PHAM**: MaLoaiSP, TenLoaiSP, MoTa - Lưu danh mục loại sản phẩm

**CHAT\_LIEU**: MaChatLieu, TenChatLieu, MoTa - Lưu danh mục chất liệu

**SAN\_PHAM**: MaSP, TenSP, SoLuongTon, DonGiaBan, DonGiaNhap, MauSac, KichCo, NgaySanXuat, MoTa, MaThuongHieu, MaLoaiSP, MaChatLieu, MaNCC - Quản lý sản phẩm

**KHUYEN\_MAI**: MaKhuyenMai, TenKhuyenMai, PhanTramKhuyenMai, NgayBatDau, NgayKetThuc, MoTa - Quản lý khuyến mãi

**HOA\_DON\_BAN**: MaHoaDon, NgayLapHoaDon, TongTien, MaNhanVien, MaKhuyenMai, MaKhachHang - Lưu hóa đơn bán

**CHI\_TIET\_HOA\_DON\_BAN**: MaHoaDon, MaSP, SoLuongBan, DonGia, ThanhTien - Chi tiết hóa đơn

**PHIEU\_NHAP**: MaPhieuNhap, NgayLapPhieuNhap, TongTien, MaNCC, MaNhanVien - Lưu phiếu nhập

**CHI\_TIET\_PHIEU\_NHAP**: MaPhieuNhap, MaSP, SoLuong, DonGia, ThanhTien - Chi tiết phiếu nhập

### 3.5.4 Mối quan hệ và ràng buộc

Quan hệ 1-1: NHAN\_VIEN - TAI\_KHOAN, NHAN\_VIEN - CHUC\_VU

Quan hệ 1-nhiều: NHAN\_VIEN - HOA\_DON\_BAN/PHIEU\_NHAP, NHA\_CUNG\_CAP - SAN\_PHAM/PHIEU\_NHAP, KHUYEN\_MAI - HOA\_DON\_BAN

Quan hệ nhiều: HOA\_DON\_BAN/PHIEU\_NHAP (quaCHI\_TIET\_HOA\_DON\_BAN, CHI\_TIET\_PHIEU

\_NHAP)

Ràng buộc: CHECK (giá trị hợp lệ), UNIQUE (không trùng lặp), ON DELETE CASCADE/NO ACTION (quản lý xóa)

### 3.5.5 Tối ưu hóa

Chỉ mục: NONCLUSTERED INDEX trên MaSP, MaHoaDon, MaPhieuNhap

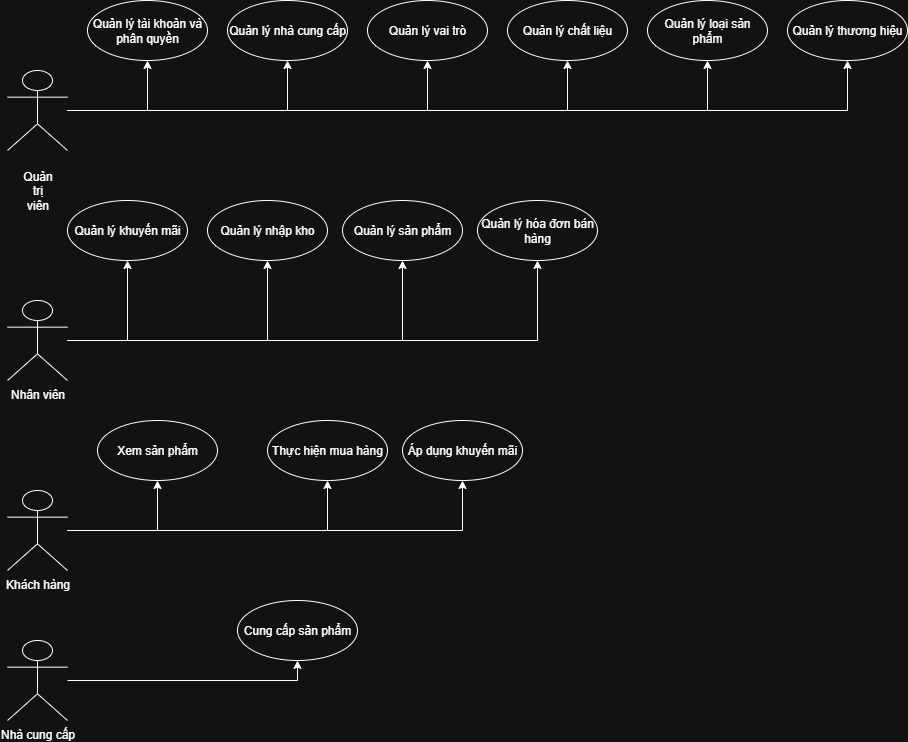
Trigger: Tính ThanhTien, cập nhật SoLuongTon, TongTien, DiemTichLuy, kiểm tra khuyến mãi

Stored Procedure: Dùng cho truy vấn phức tạp (báo cáo doanh thu)

### 3.5.6 Ứng dụng

RDM triển khai trên SQL Server, tích hợp WinForms qua mô hình 3 tầng (BLL, DAL, DTO). DAL dùng RDM để thực thi truy vấn SQL, đảm bảo quản lý dữ liệu hiệu quả.

## 3.6 SƠ ĐỒ USE CASE



Hình 3. Sơ đồ Use case

### 3.6.1 Tổng quan

Sơ đồ use case mô tả các tương tác giữa các nhân vật (actors) và các chức năng chính của phần mềm quản lý cửa hàng quần áo. Hệ thống được thiết kế để hỗ trợ quản trị viên (Admin), người dùng (User - nhân viên bán hàng), khách hàng (Customer), nhà cung cấp (Supplier), và quản lý (Manager) trong việc thực hiện các nhiệm vụ liên quan đến quản lý sản phẩm, giao dịch, và báo cáo. Sơ đồ được xây dựng theo mô hình trực quan, với các đường nối thể hiện mối quan hệ giữa nhân vật và các trường hợp sử dụng.

### 3.6.2 Các Nhân vật (Actors)

Quản trị viên (Admin): Người chịu trách nhiệm quản lý toàn diện hệ thống, bao gồm cấu hình danh mục, phân quyền, và giám sát dữ liệu.

Người dùng (User - Nhân viên bán hàng): Nhân viên trực tiếp thực hiện các giao dịch bán hàng, nhập kho, và hỗ trợ khách hàng.

Khách hàng (Customer): Người mua sản phẩm, tương tác với hệ thống thông qua người dùng.

Nhà cung cấp (Supplier): Cung cấp sản phẩm và thông tin liên quan cho cửa hàng.

Quản lý (Manager): Giám sát hoạt động và tạo báo cáo để hỗ trợ ra quyết định (mặc dù không trực tiếp thực hiện use case cụ thể).

### 3.6.3 Các Trường hợp Sử dụng (Use Cases)

Dưới đây là mô tả chi tiết từng trường hợp sử dụng và mối quan hệ với các nhân vật:

**Quản lý nhà cung cấp (Manage Suppliers):**

Mô tả: Quản trị viên thêm, sửa, xóa, và tìm kiếm thông tin về nhà cung cấp (tên, địa chỉ, liên hệ).

Nhân vật: Quản trị viên.

**Quản lý vai trò (Manage Roles):**

Mô tả: Quản trị viên tạo, chỉnh sửa, và phân bổ các vai trò (như Quản lý, Nhân viên) trong hệ thống.

Nhân vật: Quản trị viên.

**Quản lý chất liệu (Manage Materials):**

Mô tả: Quản trị viên quản lý danh mục chất liệu sản phẩm (ví dụ: cotton, lụa, v.v.).

Nhân vật: Quản trị viên.

**Quản lý loại sản phẩm (Manage Product Types):**

Mô tả: Quản trị viên quản lý các loại sản phẩm (ví dụ: áo, quần, váy).

Nhân vật: Quản trị viên.

**Quản lý thương hiệu (Manage Brands):**

Mô tả: Quản trị viên quản lý danh mục thương hiệu sản phẩm (ví dụ: Yody, Nike).

Nhân vật: Quản trị viên.

**Quản lý khuyến mãi (Manage Promotions):**

Mô tả: Quản trị viên tạo, chỉnh sửa, và theo dõi các chương trình khuyến mãi (giảm giá, ưu đãi).

Nhân vật: Quản trị viên.

**Quản lý nhập kho (Manage Inventory Imports):**

Mô tả: Người dùng xử lý việc nhập hàng từ nhà cung cấp, cập nhật số lượng tồn kho.

Nhân vật: Người dùng.

**Quản lý sản phẩm (Manage Products):**

Mô tả: Quản trị viên thêm, sửa, xóa, và quản lý thông tin sản phẩm (mã, tên, giá, chất liệu).

Nhân vật: Quản trị viên.

**Quản lý hóa đơn bán hàng (Manage Sales Invoices):**

Mô tả: Người dùng tạo và quản lý hóa đơn bán hàng cho khách hàng.

Nhân vật: Người dùng.

**Xem sản phẩm (View Products):**

Mô tả: Khách hàng hoặc người dùng xem danh sách sản phẩm có sẵn.

Nhân vật: Khách hàng, Người dùng.

**Thực hiện mua hàng (Make Purchase):**

Mô tả: Khách hàng thực hiện mua sản phẩm thông qua người dùng.

Nhân vật: Khách hàng.

**Áp dụng khuyến mãi (Apply Promotion):**

Mô tả: Người dùng áp dụng mã khuyến mãi khi tạo hóa đơn cho khách hàng.

Nhân vật: Người dùng.

**Cung cấp sản phẩm (Supply Products):**

Mô tả: Nhà cung cấp cung cấp sản phẩm và thông tin liên quan (số lượng, giá).

Nhân vật: Nhà cung cấp.

### 3.6.4 Mối quan hệ (Relationships)

Quản trị viên (Admin): Kết nối với các use case quản lý (Quản lý nhà cung cấp, Quản lý vai trò, Quản lý chất liệu, Quản lý loại sản phẩm, Quản lý thương hiệu, Quản lý khuyến mãi, Quản lý sản phẩm) thông qua các đường nối trực tiếp, thể hiện vai trò kiểm soát toàn diện.

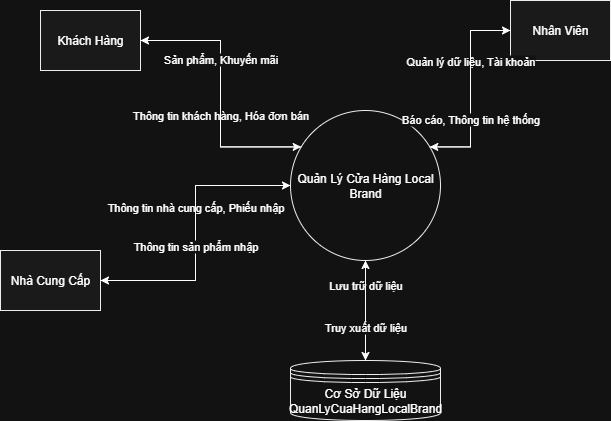
Người dùng (User): Kết nối với các use case giao dịch (Quản lý nhập kho, Quản lý hóa đơn bán hàng, Áp dụng khuyến mãi), hỗ trợ khách hàng và nhập dữ liệu.

Khách hàng (Customer): Kết nối với các use case mua sắm (Xem sản phẩm, Thực hiện mua hàng), tương tác gián tiếp qua người dùng.

Nhà cung cấp (Supplier): Kết nối với use case Cung cấp sản phẩm, cung cấp nguồn hàng cho hệ thống.

Quản lý (Manager): Không trực tiếp thực hiện use case cụ thể nhưng giám sát toàn bộ hoạt động, có thể liên quan gián tiếp qua Tạo báo cáo (nếu được mở rộng).

## 3.7 SƠ ĐỒ LUỒNG DỮ LIỆU CẤP 0 (DFD LEVEL 0)



Hình 3. Sơ đồ luồng dữ liệu cấp 0 (dfd level 0)

### 3.7.1 Tổng quan

Sơ đồ luồng dữ liệu cấp 0 (DFD Level 0) là biểu diễn tổng quát của hệ thống "Quản lý phần mềm cửa hàng local brand", thể hiện cách dữ liệu được nhập, xử lý, và xuất ra giữa các tác nhân bên ngoài và các quá trình chính bên trong hệ thống. Hệ thống được phát triển trên C# (WinForms) với mô hình 3 tầng, tích hợp cơ sở dữ liệu QuanLyCuaHangLocalBrand trên SQL Server. DFD cấp 0 được hoàn thiện vào 12:15 AM +07, ngày 02/08/2025.

### 3.7.2 Thành phần chính của DFD cấp 0

Quá trình (Processes): Các chức năng chính của hệ thống, được biểu diễn bằng hình ellipse hoặc hình tròn với số thứ tự (ví dụ: 1.0, 2.0).

Tác nhân (External Entities): Các nguồn hoặc đích của dữ liệu bên ngoài hệ thống, được biểu diễn bằng hình vuông (ví dụ: Admin, Khách hàng, Nhà cung cấp).

Luồng dữ liệu (Data Flows): Dữ liệu di chuyển giữa tác nhân và quá trình, được biểu diễn bằng mũi tên có nhãn (ví dụ: "Thông tin nhân viên", "Hóa đơn").

Kho dữ liệu (Data Stores): Nơi lưu trữ dữ liệu, được biểu diễn bằng hình chữ nhật mở (ví dụ: Cơ sở dữ liệu nhân viên, Sản phẩm).

### 3.7.3 Mô tả DFD cấp 0

**Các quá trình chính**

Quá trình 1.0: Quản lý danh mục

Mô tả: Xử lý thêm, sửa, xóa, và tìm kiếm thông tin danh mục (Nhân viên, Chức vụ, Khách hàng, Nhà cung cấp, Sản phẩm, Thương hiệu, Loại sản phẩm, Chất liệu, Khuyến mãi).

Luồng vào: "Thông tin nhân viên" từ Admin, "Thông tin sản phẩm" từ Nhà cung cấp.

Luồng ra: "Danh sách danh mục" đến Admin.

Kho dữ liệu: "Cơ sở dữ liệu danh mục" (bao gồm NHAN\_VIEN, SAN\_PHAM, v.v.).

Quá trình 2.0: Quản lý giao dịch

Mô tả: Xử lý nhập hàng, bán hàng, và tạo hóa đơn.

Luồng vào: "Yêu cầu nhập hàng" từ Nhà cung cấp, "Thông tin bán hàng" từ User.

Luồng ra: "Phiếu nhập" đến Nhà cung cấp, "Hóa đơn" đến Khách hàng, "Thông báo giao dịch" đến User.

Kho dữ liệu: "Cơ sở dữ liệu giao dịch" (bao gồm PHIEU\_NHAP, HOA\_DON\_BAN).

Quá trình 3.0: Quản lý tài khoản và phân quyền

Mô tả: Xác thực đăng nhập và phân quyền cho các tác nhân.

Luồng vào: "Thông tin đăng nhập" từ Admin, User, Manager.

Luồng ra: "Quyền truy cập" đến các tác nhân, "Thông báo đăng nhập" đến User.

Kho dữ liệu: "Cơ sở dữ liệu tài khoản" (TAI\_KHOAN).

Quá trình 4.0: Báo cáo và thống kê

Mô tả: Tạo báo cáo doanh thu/thống kê theo thời gian.

Luồng vào: "Yêu cầu báo cáo" từ Manager, "Dữ liệu giao dịch" từ Kho dữ liệu.

Luồng ra: "Báo cáo doanh thu" đến Manager.

Kho dữ liệu: "Cơ sở dữ liệu giao dịch" (HOA\_DON\_BAN).

**Các tác nhân (External Entities)**

Admin: Cung cấp và nhận thông tin danh mục, tài khoản.

User (Nhân viên bán hàng): Cung cấp và nhận thông tin giao dịch.

Manager: Nhận báo cáo doanh thu.

Nhà cung cấp: Cung cấp thông tin sản phẩm và nhận phiếu nhập.

Các kho dữ liệu (Data Stores)

Cơ sở dữ liệu danh mục: Lưu trữ thông tin NHAN\_VIEN, CHUC\_VU, KHACH\_HANG, NHA\_CUNG\_CAP, SAN\_PHAM, THUONG\_HIEU, LOAI\_SAN\_PHAM, CHAT\_LIEU, KHUYEN\_MAI.

Cơ sở dữ liệu giao dịch: Lưu trữ thông tin PHIEU\_NHAP, CHI\_TIET\_PHIEU\_NHAP, HOA\_DON\_BAN, CHI\_TIET\_HOA\_DON\_BAN,.

Cơ sở dữ liệu tài khoản: Lưu trữ thông tin TAI\_KHOAN.

**Luồng dữ liệu (Data Flows)**

Admin → Quá trình 1.0: "Thông tin nhân viên", "Thông tin danh mục".

Nhà cung cấp → Quá trình 2.0: "Thông tin sản phẩm", "Yêu cầu nhập hàng".

User → Quá trình 2.0/3.0: "Thông tin bán hàng".

Manager → Quá trình 5.0: "Yêu cầu báo cáo".

Quá trình 1.0 → Admin: "Danh sách danh mục".

Quá trình 2.0 → Nhà cung cấp: "Phiếu nhập".

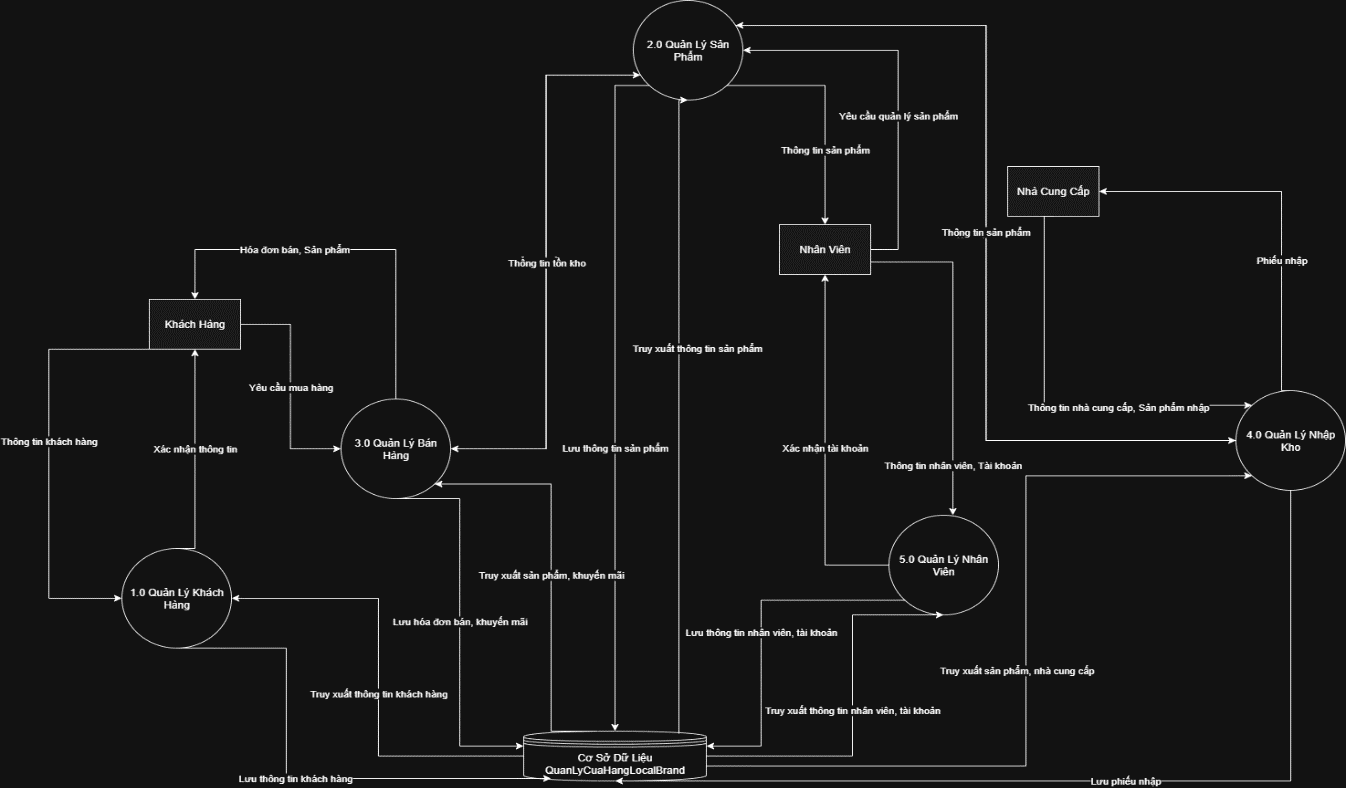
Quá trình 2.0 → Khách hàng: "Hóa đơn".

Quá trình 3.0 → User: "Quyền truy cập", "Thông báo đăng nhập".

Quá trình 4.0 → Manager: "Báo cáo doanh thu".

Quá trình 1.0/2.0/3.0/4.0/5.0 ↔ Kho dữ liệu: Dữ liệu hai chiều (đọc/ghi).

## 3.8 SƠ ĐỒ LUỒNG DỮ LIỆU CẤP 1 (DFD LEVEL 1)



Hình 3. Sơ đồ luồng dữ liệu cấp 1 (dfd level 1)

### 3.8.1 Tổng quan

Sơ đồ luồng dữ liệu cấp 1 (DFD Level 1) là sự phân tách chi tiết của các quá trình trong DFD cấp 0, mô tả cách dữ liệu được xử lý bên trong từng chức năng chính của hệ thống "Quản lý phần mềm cửa hàng local brand". Hệ thống được phát triển trên C# (WinForms) với mô hình 3 tầng, tích hợp cơ sở dữ liệu QuanLyCuaHangLocalBrand trên SQL Server.

### 3.8.2 Thành phần chính của DFD cấp 1

Quá trình con (Sub-processes): Các chức năng chi tiết bên trong mỗi quá trình cấp 0, được đánh số (ví dụ: 1.1, 1.2).

Tác nhân (External Entities): Giữ nguyên từ DFD cấp 0 (Admin, User, Manager, Khách hàng, Nhà cung cấp).

Luồng dữ liệu (Data Flows): Chi tiết hơn, thể hiện dữ liệu vào/ra giữa các quá trình con, tác nhân, và kho dữ liệu.

Kho dữ liệu (Data Stores): Cụ thể hóa các bảng trong cơ sở dữ liệu.

### 3.8.3 Mô tả DFD cấp 1 cho từng quá trình cấp 0

Quá trình 1.0: Quản lý danh mục

Quá trình con:

1.1: Quản lý nhân viên/chức vụ

Luồng vào: "Thông tin nhân viên" (Họ tên, Ngày sinh, v.v.) từ Admin.

Luồng ra: "Danh sách nhân viên" đến Admin.

Kho dữ liệu: NHAN\_VIEN, CHUC\_VU.

1.2: Quản lý sản phẩm/thương hiệu/loại sản phẩm/chất liệu

Luồng vào: "Thông tin sản phẩm" (Mã SP, Tên SP, v.v.) từ Admin/Nhà cung cấp.

Luồng ra: "Danh sách sản phẩm" đến Admin.

Kho dữ liệu: SAN\_PHAM, THUONG\_HIEU, LOAI\_SAN\_PHAM, CHAT\_LIEU.

1.3: Quản lý khách hàng/nhà cung cấp/khuyến mãi

Luồng vào: "Thông tin khách hàng" (Tên, Địa chỉ, v.v.) từ Admin, "Thông tin nhà cung cấp" từ Nhà cung cấp.

Luồng ra: "Danh sách khách hàng/nhà cung cấp/khuyến mãi" đến Admin.

Kho dữ liệu: KHACH\_HANG, NHA\_CUNG\_CAP, KHUYEN\_MAI.

Quá trình 2.0: Quản lý giao dịch

Quá trình con:

2.1: Nhập hàng

Luồng vào: "Yêu cầu nhập hàng" từ Nhà cung cấp, "Thông tin chi tiết nhập" từ User.

Luồng ra: "Phiếu nhập" đến Nhà cung cấp, "Thông báo nhập hàng" đến User.

Kho dữ liệu: PHIEU\_NHAP, CHI\_TIET\_PHIEU\_NHAP, SAN\_PHAM (cập nhật SoLuongTon).

2.2: Bán hàng và tạo hóa đơn

Luồng vào: "Thông tin bán hàng" (Mã SP, Số lượng) từ User, "Thông tin khuyến mãi" từ KHUYEN\_MAI.

Luồng ra: "Hóa đơn" đến Khách hàng, "Thông báo bán hàng" đến User.

Kho dữ liệu: HOA\_DON\_BAN, CHI\_TIET\_HOA\_DON\_BAN, SAN\_PHAM (cập nhật SoLuongTon).

Quá trình 4.0: Quản lý tài khoản và phân quyền

Quá trình con:

3.1: Đăng nhập

Luồng vào: "Thông tin đăng nhập" (Tên đăng nhập, Mật khẩu) từ Admin/User/Manager.

Luồng ra: "Quyền truy cập" đến tác nhân, "Thông báo đăng nhập" đến User.

Kho dữ liệu: TAI\_KHOAN.

3.2: Quản lý phân quyền

Luồng vào: "Yêu cầu phân quyền" từ Admin.

Luồng ra: "Cập nhật quyền truy cập" đến TAI\_KHOAN.

Kho dữ liệu: TAI\_KHOAN, CHUC\_VU.

Quá trình 4.0: Báo cáo và thống kê

Quá trình con:

4.1: Tạo báo cáo doanh thu

Luồng vào: "Yêu cầu báo cáo" (thời gian) từ Manager.

Luồng ra: "Báo cáo doanh thu" đến Manager.

Kho dữ liệu: HOA\_DON\_BAN, CHI\_TIET\_HOA\_DON\_BAN.

# CHƯƠNG 4 GIAO DIỆN HỆ THỐNG

## 4.1 Trang đăng nhập

Hình 4. Trang đăng nhập

## 4.2 Trang chủ

Hình 4. Trang chủ

## 4.3 Trang thông tin nhân viên

Hình 4. Trang thông tin nhân viên

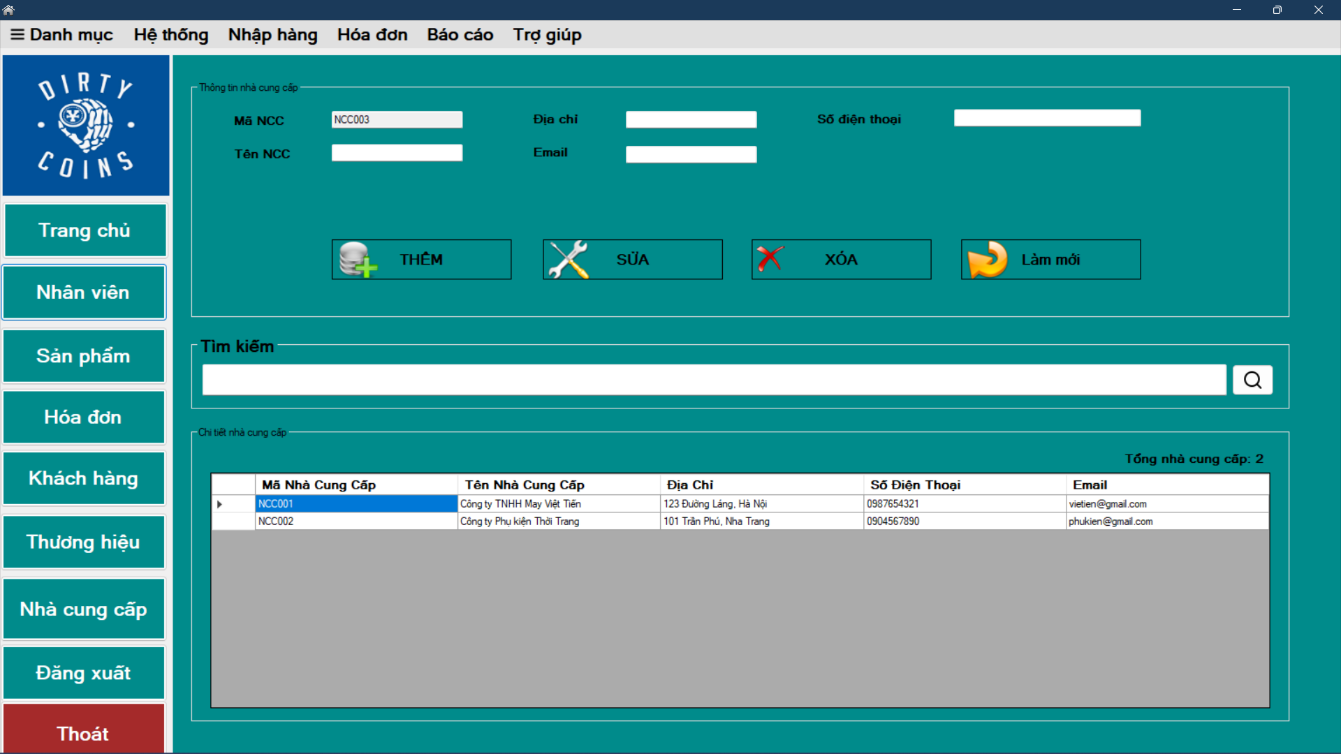
## 4.4 Trang thông tin chức vụ

Hình 4. Trang thông tin chức vụ

## 4.5 Trang thông tin khách hàng

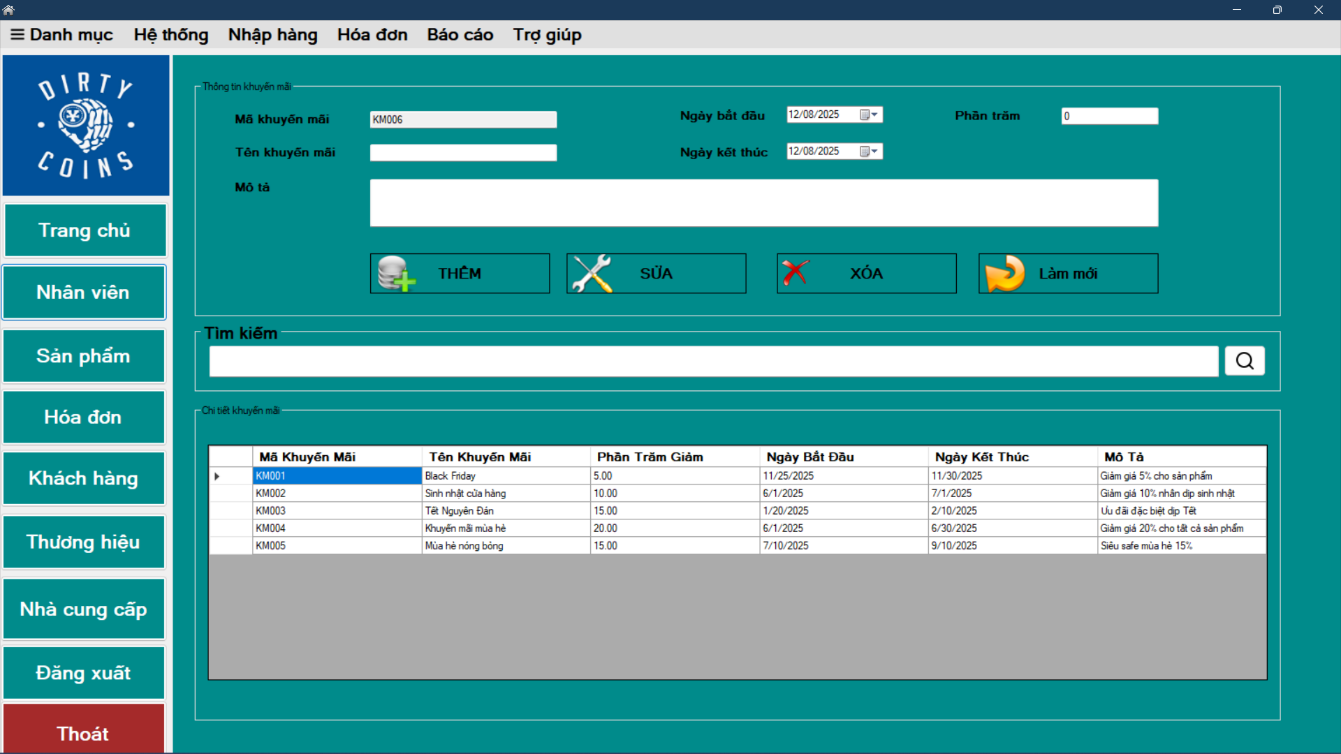
Hình 4. Trang thông tin khách hàng

## 4.6 Trang thông tin nhà cung cấp



Hình 4. Trang thông tin nhà cung cấp

## 4.7 Trang thông tin khuyến mãi

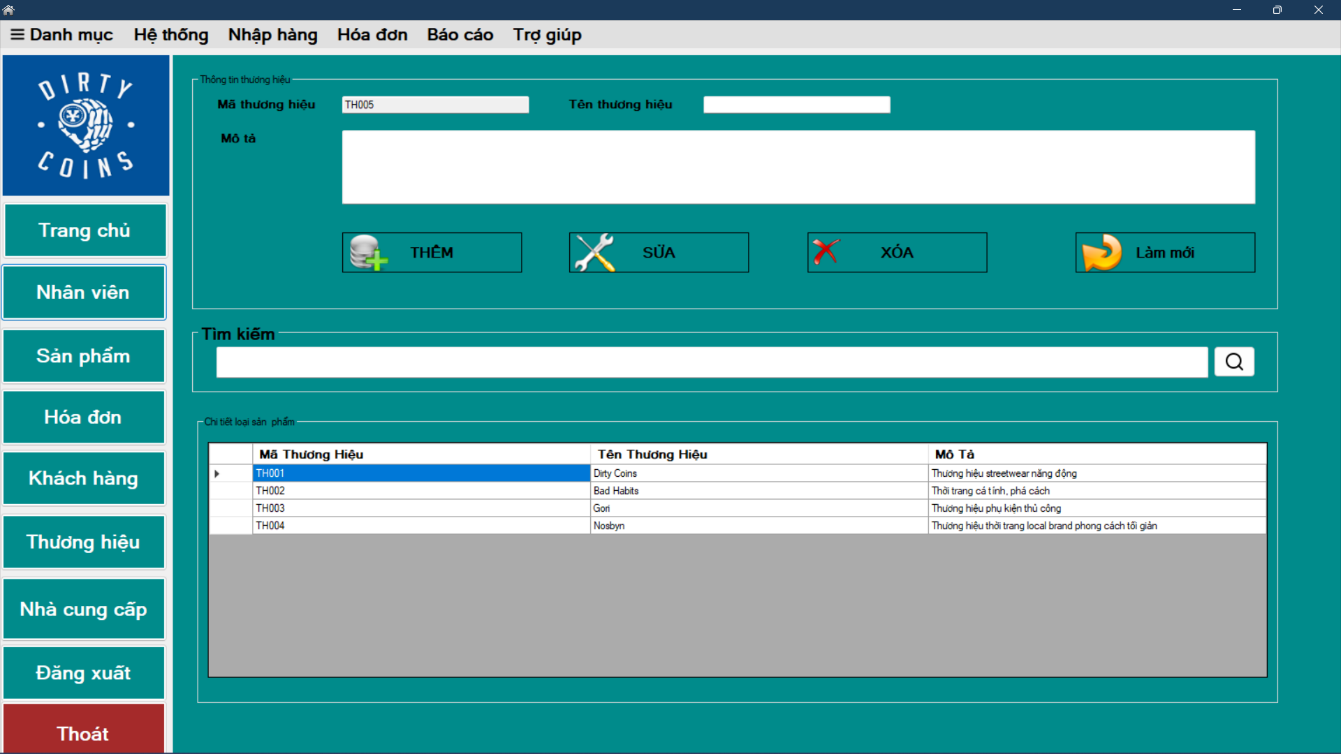


Hình 4. Trang thông tin khuyến mãi

## 4.8 Trang thông tin chất liệu

Hình 4. Trang thông tin chất chịu

## 4.9 Trang thông tin thương hiệu

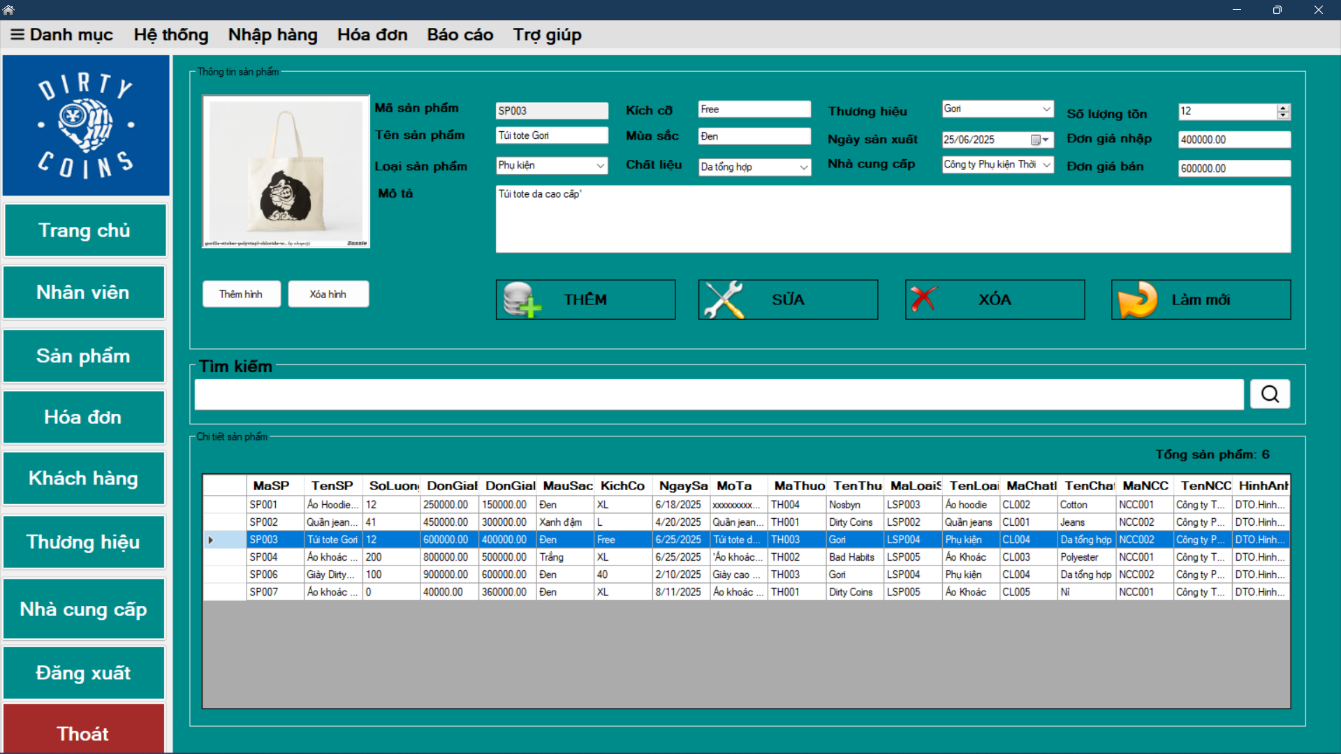


Hình 4. Trang thông tin thương hiệu

## 4.10 Trang thông tin loại sản phẩm

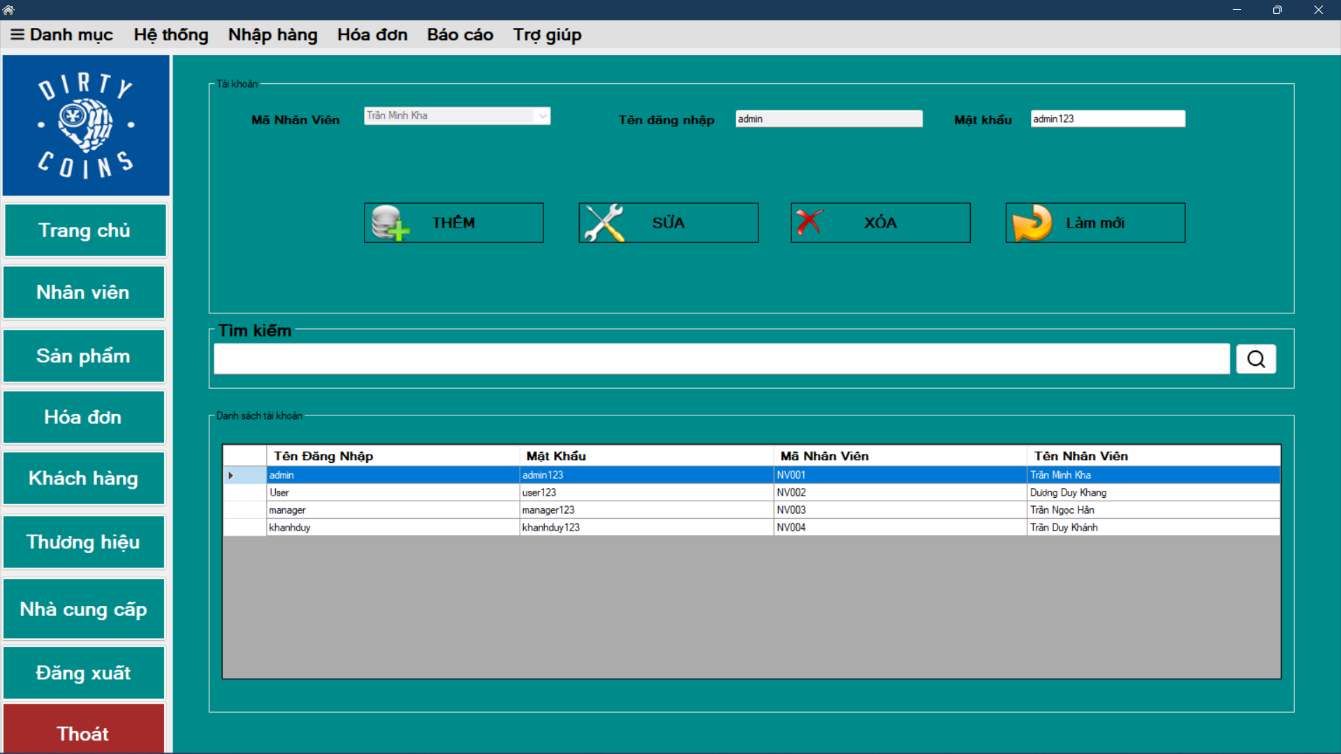
Hình 4. Trang thông tin loại sản phẩm

## 4.11 Trang thông tin sản phẩm



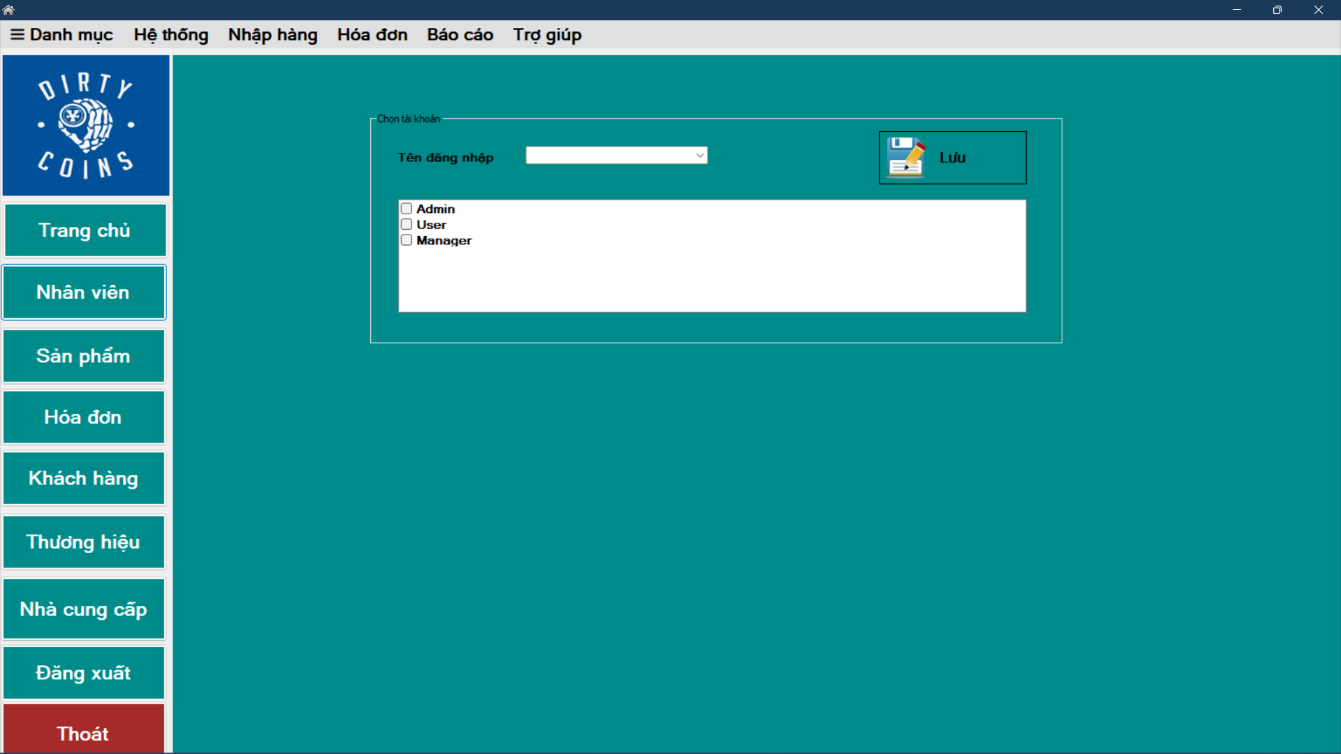
Hình 4. Trang thông tin sản phẩm

## 4.12 Trang phân quyền



Hình 4. Trang phân quyền

## 4.13 Trang chọn tài khoản

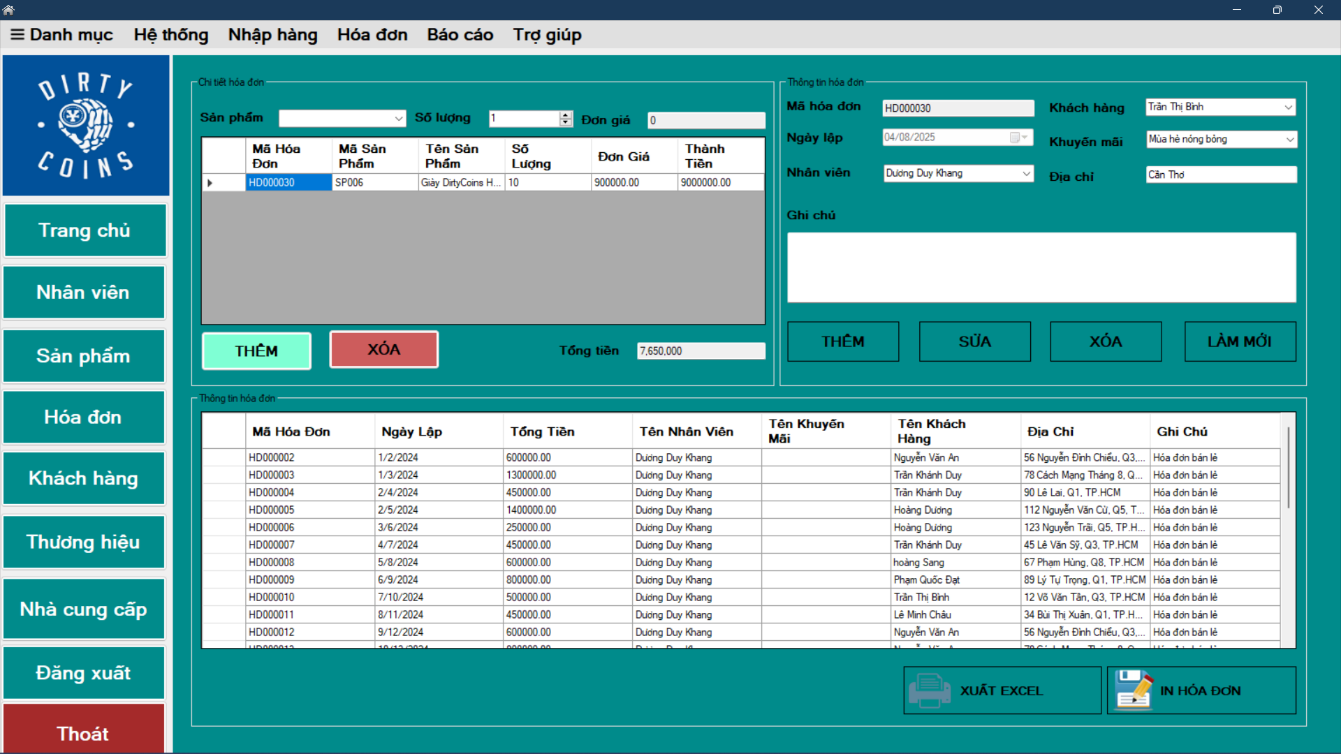


Hình 4. Trang chọn tài khoản

## 4.14 Trang thông tin phiếu nhập

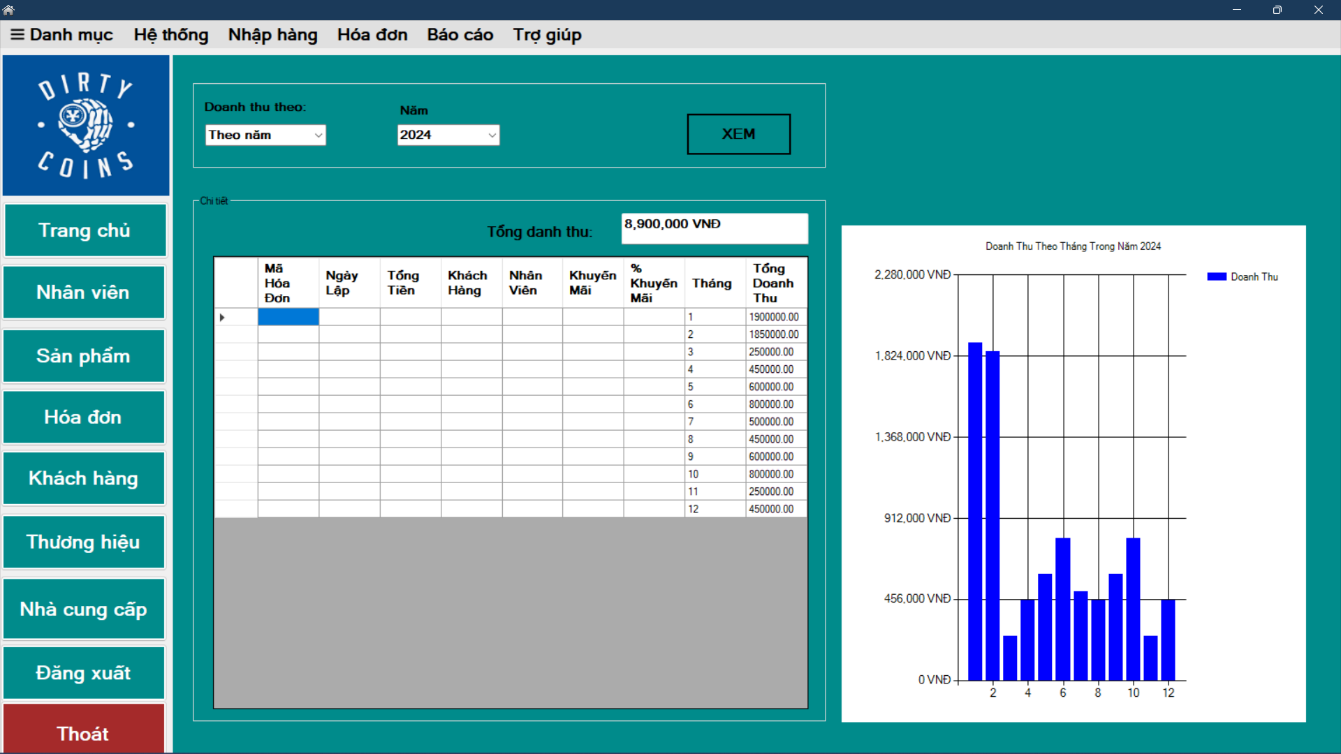
Hình 4. Trang thông tin phiếu nhập

## 4.15 Trang thông tin hóa đơn



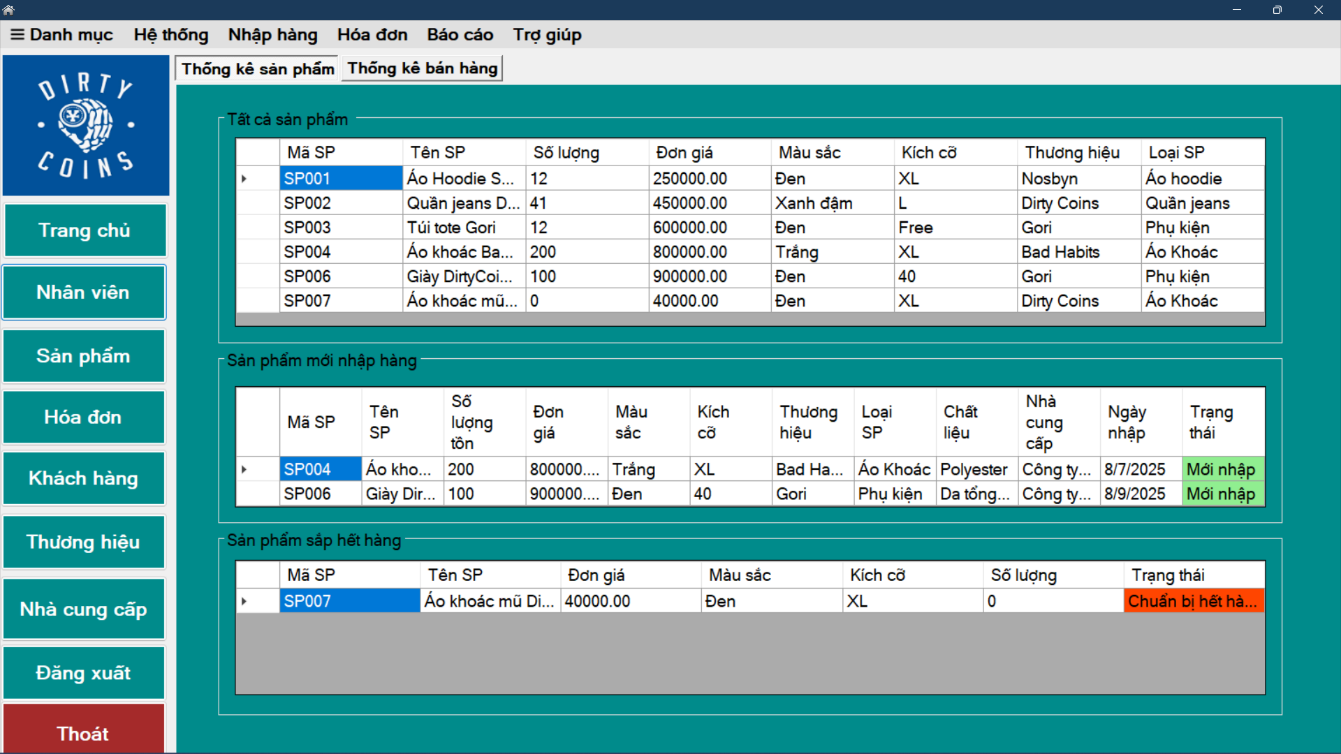
Hình 4. Trang thông tin hóa đơn

## 4.16 Trang thông tin doanh thu



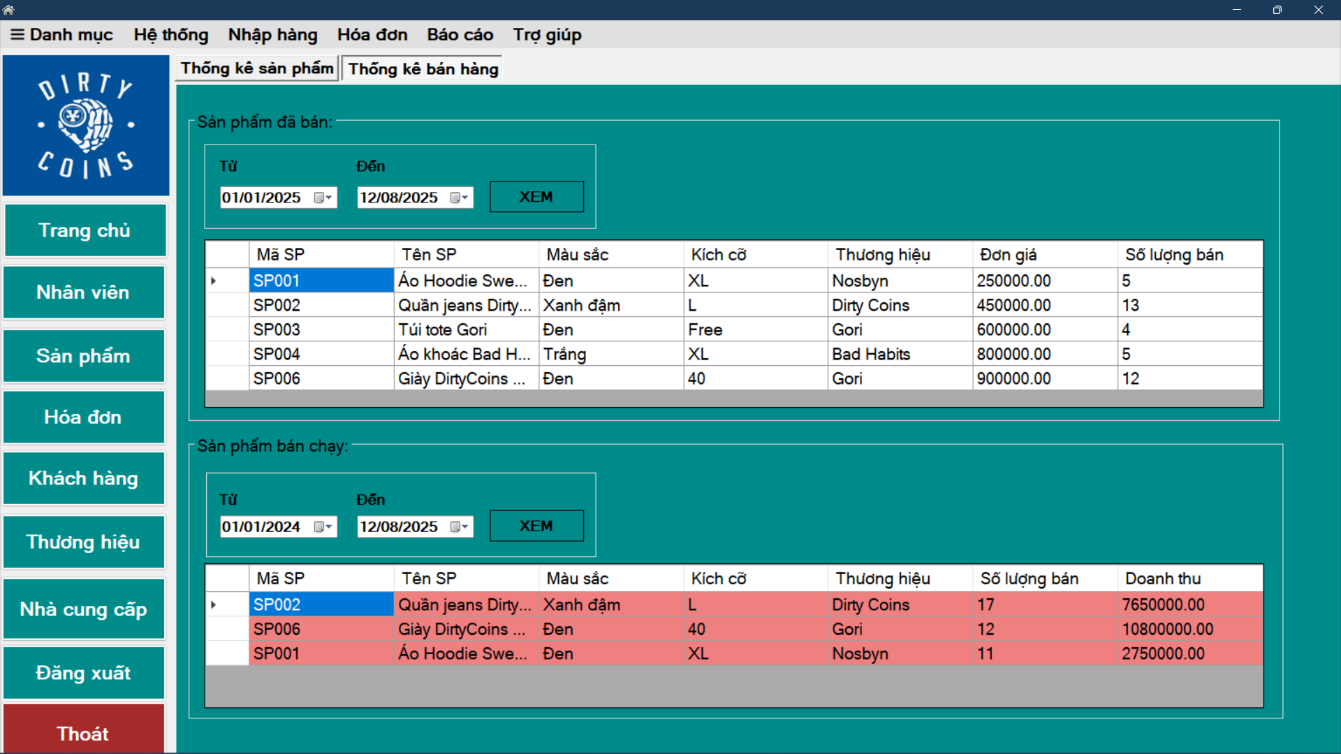
Hình 4. Trang thông tin doanh thu

## 4.17 Trang thống kê sản phẩm



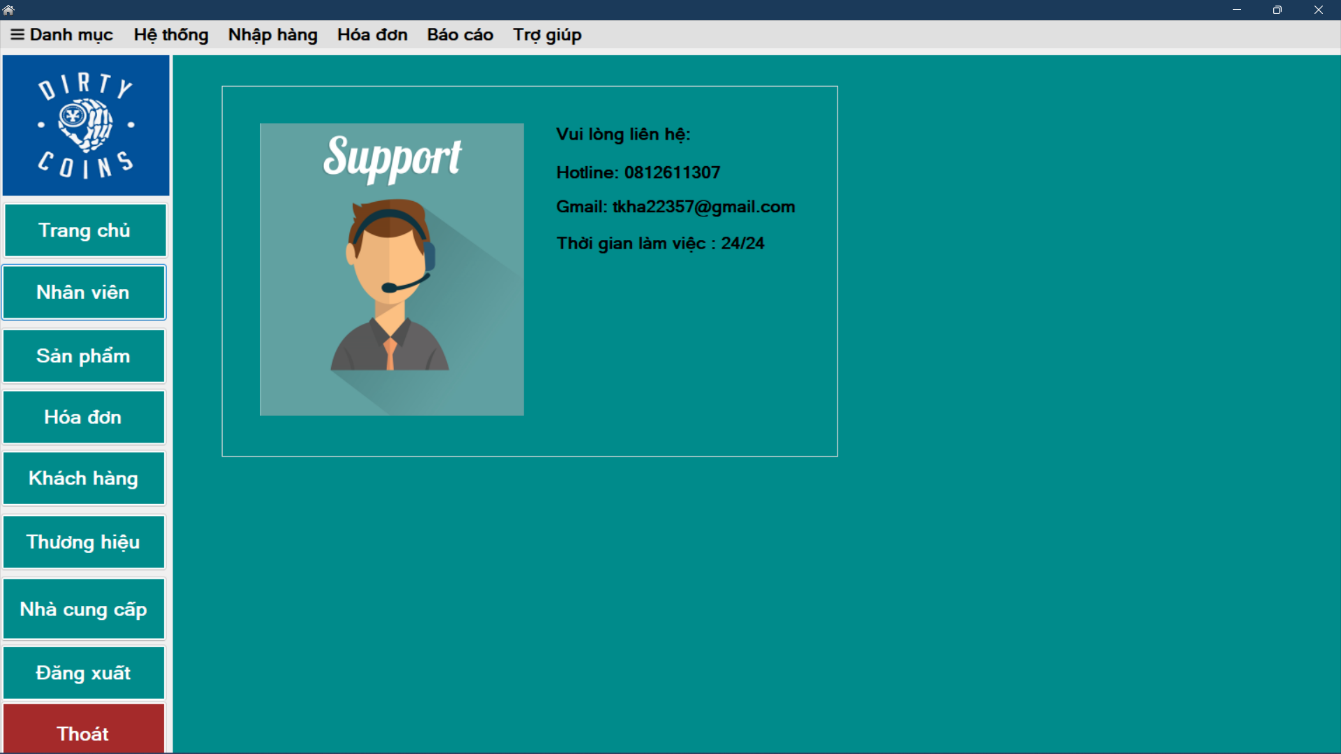
Hình 4. Trang thống kê sản phẩm

## 4.18 Trang thống kê bán hàng



Hình 4. Trang thống kê bán hàng

## 4.19 Trang chăm sóc khách hàng



Hình 4. Trang chăm sóc khách hàng

# KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

Phần mềm "Quản lý cửa hàng local brand", đáp ứng các yêu cầu cơ bản về quản lý danh mục, giao dịch và báo cáo. Được xây dựng trên nền tảng WinForms và SQL Server với mô hình 3 tầng, phần mềm mang lại giao diện thân thiện, hiệu suất ổn định và khả năng hoạt động ngoại tuyến, phù hợp với các cửa hàng quần áo quy mô nhỏ và vừa.

Nhằm hướng tới sự phát triển bền vững và nâng cao tính cạnh tranh, các định hướng cải tiến được đề xuất như sau:

**Tích hợp nền tảng đám mây:** Cho phép đồng bộ dữ liệu thời gian thực giữa các chi nhánh, hỗ trợ quản lý tập trung và mở rộng quy mô.

**Ứng dụng trí tuệ nhân tạo:** Triển khai phân tích dữ liệu khách hàng để dự đoán xu hướng tiêu dùng và đề xuất chiến lược kinh doanh hiệu quả.

**Kết nối thanh toán trực tuyến:** Hỗ trợ tích hợp với các cổng thanh toán như Momo, ZaloPay để tối ưu hóa trải nghiệm khách hàng.

**Phát triển giao diện web hoặc ứng dụng di động:** Mở rộng khả năng tiếp cận, cho phép quản lý từ xa và phục vụ khách hàng mọi lúc, mọi nơi.

**Nâng cấp báo cáo thông minh:** Thêm các công cụ trực quan hóa dữ liệu (biểu đồ, đồ thị) và phân tích đa chiều để hỗ trợ ra quyết định chiến lược.

Những định hướng này không chỉ giúp phần mềm thích nghi với xu hướng công nghệ mà còn tạo nền tảng vững chắc cho sự phát triển lâu dài trong ngành bán lẻ thời

# TÀI LIỆU THAM KHẢO

LÊ ĐỨC THẮNG. (2023). GIÁO TRÌNH PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG THÔNG TIN. Cần Thơ.

Tài liệu về lập trình Windows Form bằng C#. (2024, 11 29). Retrieved from VniTeach-Giáo viên 4.0: <https://listwr.com/ATEhim>

TRƯƠNG PHẠM HỒNG NGÂN. (n.d.). TÀI LIỆU HƯỚNG DẪN LẬP TRÌNH.

Và các giáo trình và tài liệu hướng dẫn trên lớp.