TRƯỜNG ĐẠI HỌC KINH TẾ

**KHOA THỐNG KÊ – TIN HỌC**

–––––––––––––––––––––––––––––––



**BÁO CÁO BÀI TẬP NHÓM**

**QUẢN LÝ CỬA HÀNG**

**PHỤ TÙNG Ô TÔ HÀ**

**Môn học: Quản trị cơ sở dữ liệu**

**Sinh viên thực hiện: Nhóm 07**

1. Chế Thị Nhã Quyên 45K14
2. Trần Đỗ Hòa 45K14
3. Nguyễn Thị Ngọc 45K14
4. Nguyễn Thị My La 45K14

**GVHD: Cao Thị Nhâm**

***Đà Nẵng, 10/2021***

# MỤC LỤC

[MỤC LỤC i](#_Toc89288714)

[TỈ LỆ PHÂN TRĂM CÔNG VIỆC iii](#_Toc89288715)

[DANH MỤC HÌNH ẢNH iv](#_Toc89288716)

[DANH MỤC BẢNG BIỂU v](#_Toc89288717)

[1. Thiết kế cơ sở dữ liệu 1](#_Toc89288718)

[1.1. Thiết kế cơ sở dữ liệu ở mức khái niệm 1](#_Toc89288719)

[1.2. Thiết kế cơ sở dữ liệu mức logic 5](#_Toc89288720)

[1.3. Thiết kế cơ sở dữ liệu mức vật lý 7](#_Toc89288721)

[2. Xây dựng cơ sở dữ liệu 9](#_Toc89288722)

[2.1. Bảng HANG 9](#_Toc89288723)

[2.2. Bảng LOAIHANG 10](#_Toc89288724)

[2.3. Bảng NHANVIEN 10](#_Toc89288725)

[2.4. Bảng KHACHHANG 10](#_Toc89288726)

[2.5. Bảng HOADON 11](#_Toc89288727)

[2.6. Bảng CHITIET\_HD 11](#_Toc89288728)

[2.7. Bảng PHIEUNHAP 12](#_Toc89288729)

[2.8. Bảng CHITIET\_PN 12](#_Toc89288730)

[2.9. Bảng NHACC 13](#_Toc89288731)

[3. Xác định và tạo các index cho các thuộc tính cần thiết 13](#_Toc89288732)

[4. Xây dựng cơ chế bảo mật tương ứng cho cơ sở dữ liệu 14](#_Toc89288733)

[5. Xây dựng cơ chế backup dữ liệu phù hợp cho cơ sở dữ liệu 17](#_Toc89288734)

[5.1. Full backup 17](#_Toc89288735)

[5.2. Differential backup 18](#_Toc89288736)

[5.3. Transaction logs backup 18](#_Toc89288737)

[6. Phát triển ứng dụng mang tên TESTDB – Sử dụng ngôn ngữ C# 20](#_Toc89288738)

[6.1. Form đăng nhập hệ thống 20](#_Toc89288739)

[6.2. Giao diện form chính 21](#_Toc89288740)

[6.3. Quản lý Hàng hóa 21](#_Toc89288741)

[6.4. Quản lý Hóa đơn 23](#_Toc89288742)

[6.5. Quản lý Phiếu nhập 24](#_Toc89288743)

[6.6. Quản lý Khách hàng 25](#_Toc89288744)

[6.7. Quản lý Nhà cung cấp 26](#_Toc89288745)

[6.8. Quản lý Nhân viên 27](#_Toc89288746)

[7. Đưa ra phương án giải quyết cho vấn đề lượng dữ liệu rất lớn, không đủ lưu trữ trong một ổ đĩa 28](#_Toc89288747)

[8. Đề phòng phương án hệ thống CHPT bị tấn công bằng SQL Injection. Đưa ra phương án và thực thi phương án. 28](#_Toc89288748)

# TỈ LỆ PHÂN TRĂM CÔNG VIỆC

|  |  |
| --- | --- |
| **THÀNH VIÊN** | **PHẦN TRĂM** |
| 1. Trần Đỗ Hòa | 27 % |
| 2. Nguyễn Thị My La | 24 % |
| 3. Nguyễn Thị Ngọc | 24 % |
| 4. Chế Thị Nhã Quyên | 25 % |

# DANH MỤC HÌNH ẢNH

[Hình 1. 1: Hóa đơn bán hàng 1](#_Toc89288749)

[Hình 1. 2: Mô hình ER – BÁN HÀNG 2](#_Toc89288750)

[Hình 1. 3: Hóa đơn nhập hàng 3](#_Toc89288751)

[Hình 1. 4: Mô hình ER – NHẬP HÀNG 5](#_Toc89288752)

[Hình 1. 5: Mô hình tích hợp ER 5](#_Toc89288753)

[Hình 1. 6: Sơ đồ cơ sở dữ liệu 6](#_Toc89288754)

[Hình 6. 1: Form đăng nhập hệ thống 20](#_Toc89288755)

[Hình 6. 2: Form thông báo đăng nhập sai 20](#_Toc89288756)

[Hình 6. 3: Form giao diện chính 21](#_Toc89288757)

[Hình 6. 4: Form thông tin hàng và loại hàng 22](#_Toc89288758)

[Hình 6. 5: Form thông tin Hóa đơn và chi tiết hóa đơn 23](#_Toc89288759)

[Hình 6. 6: Form thông tin phiếu nhập và chi tiết tiết phiếu nhập 25](#_Toc89288760)

[Hình 6. 7: Form thông tin Khách hàng 25](#_Toc89288761)

[Hình 6. 8: Form thông tin Nhà cung cấp 27](#_Toc89288762)

[Hình 6. 9: Form thông tin nhân viên 27](#_Toc89288763)

# DANH MỤC BẢNG BIỂU

[Bảng 1. 1: Bảng dữ liệu từ Hóa đơn bán hàng 2](#_Toc89288628)

[Bảng 1. 2: Bảng dữ liệu từ Hóa đơn nhập hàng 4](#_Toc89288629)

[Bảng 1. 3: Bảng hàng 7](#_Toc89288630)

[Bảng 1. 4: Bảng loại hàng 7](#_Toc89288631)

[Bảng 1. 5: Bảng nhân viên 7](#_Toc89288632)

[Bảng 1. 6: Bảng khách hàng 8](#_Toc89288633)

[Bảng 1. 7: Bảng hóa đơn 8](#_Toc89288634)

[Bảng 1. 8: Bảng chi tiết hóa đơn 8](#_Toc89288635)

[Bảng 1. 9: Bảng phiếu nhập 9](#_Toc89288636)

[Bảng 1. 10: Bảng chi tiết phiếu nhập 9](#_Toc89288637)

[Bảng 1. 11: Bảng nhà cung cấp 9](#_Toc89288638)

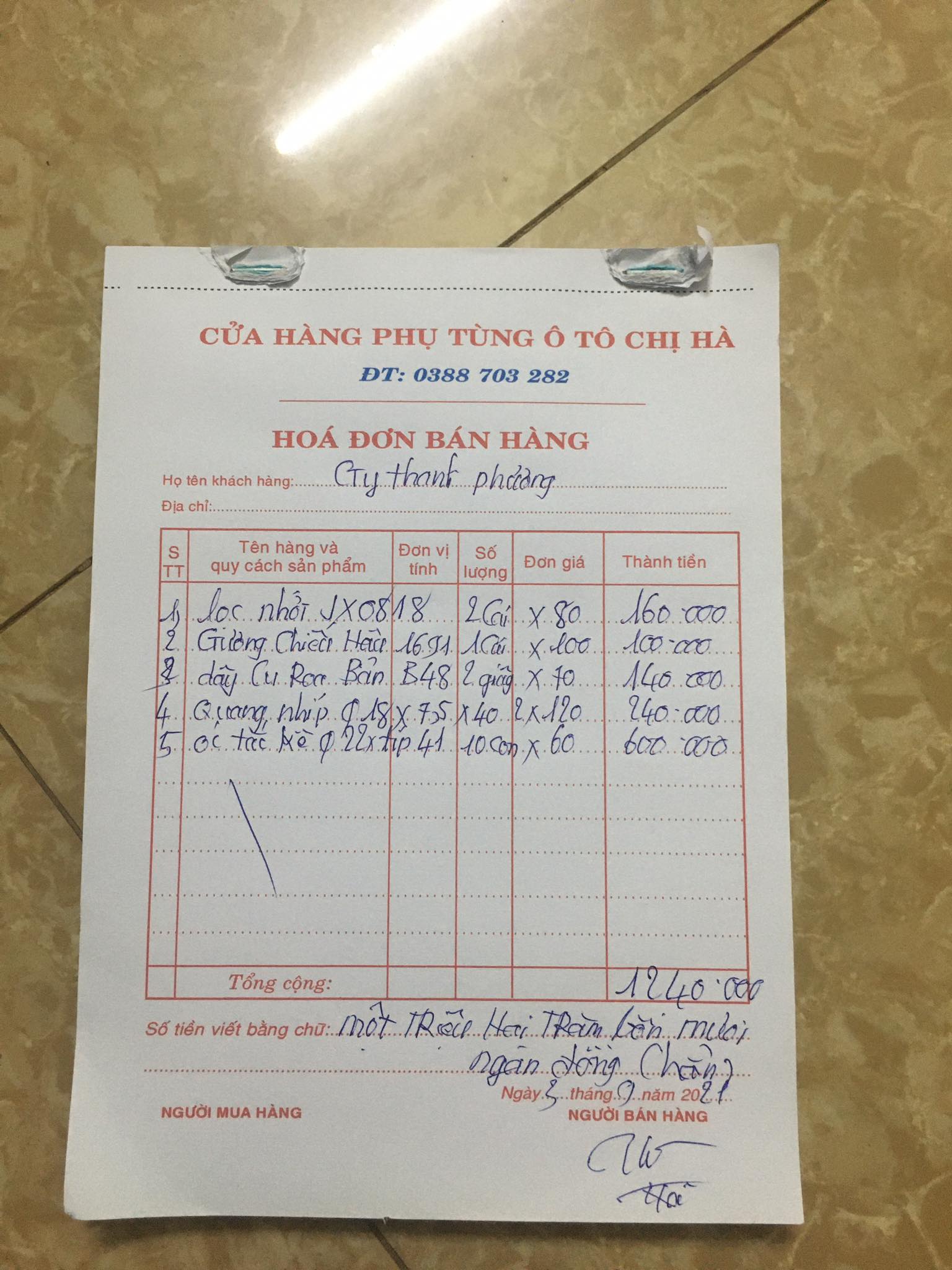
# 

## Thiết kế cơ sở dữ liệu

### Thiết kế cơ sở dữ liệu ở mức khái niệm

**Hóa đơn bán hàng**

* Các thực thể



Hình 1. 1: Hóa đơn bán hàng

|  |  |  |
| --- | --- | --- |
| Từ trong HSDL | Từ rõ nghĩa | Viết tắt |
| ~~Họ tên khách hàng~~ | ~~Tên khách hàng~~ | ~~TenKH~~ |
| ~~Địa chỉ~~ | ~~Địa chỉ~~ | ~~DiaChi~~ |
| ~~Tên hàng~~ | ~~Tên hàng~~ | ~~TenH~~ |
| ~~Quy cách hàng~~ | ~~Loại hàng~~ | ~~TenLH~~ |
| Đơn vị tính | Đơn vị | DonVi |
| Số lượng | Số lượng bán | SoLuongBan |
| ~~Đơn giá~~ | ~~Đơn giá bán~~ | ~~DonGiaBan~~ |
| Thành tiền | Thành tiền | ThanhTien |
| Ngày bán | Ngày bán | NgayBan |
| ~~Người bán hàng~~ | ~~Nhân viên~~ | ~~TenNV~~ |
| Tổng cộng | Tổng cộng | TongCong |

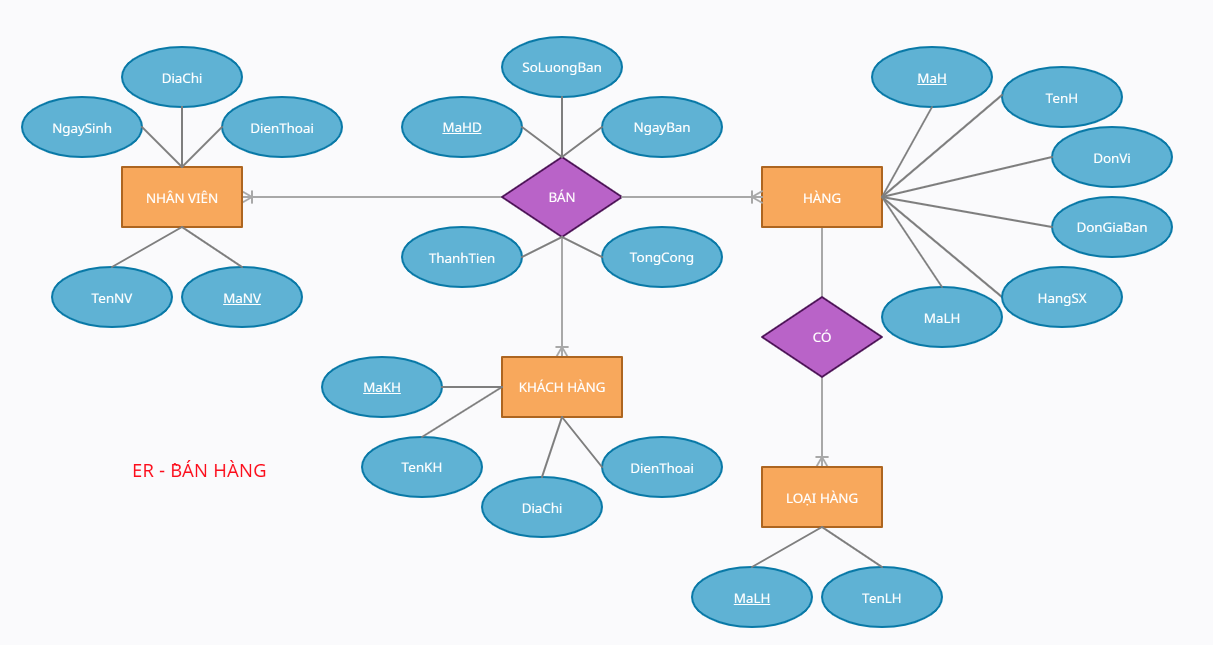
Bảng 1. : Bảng dữ liệu từ Hóa đơn bán hàng

* KHÁCH HÀNG (MaKH, TenKH, DiaChi, DienThoai)
* HÀNG (MaH, TenH, DonVi, DonGiaBan, HangSX, *MaLH*)
* LOẠI HÀNG (MaLH, TenLH)
* NHÂN VIÊN (MaNV, TenNV, NgaySinh, DiaChi, DienThoai)

*Bổ sung thêm:* MaKH, DienThoai, MaH, HangSX, MaLH, MaNV, NgaySinh, DiaChi, DienThoai

* Quan hệ
* BÁN (MaHD, NgayBan, SoLuongBan, ThanhTien, TongCong)

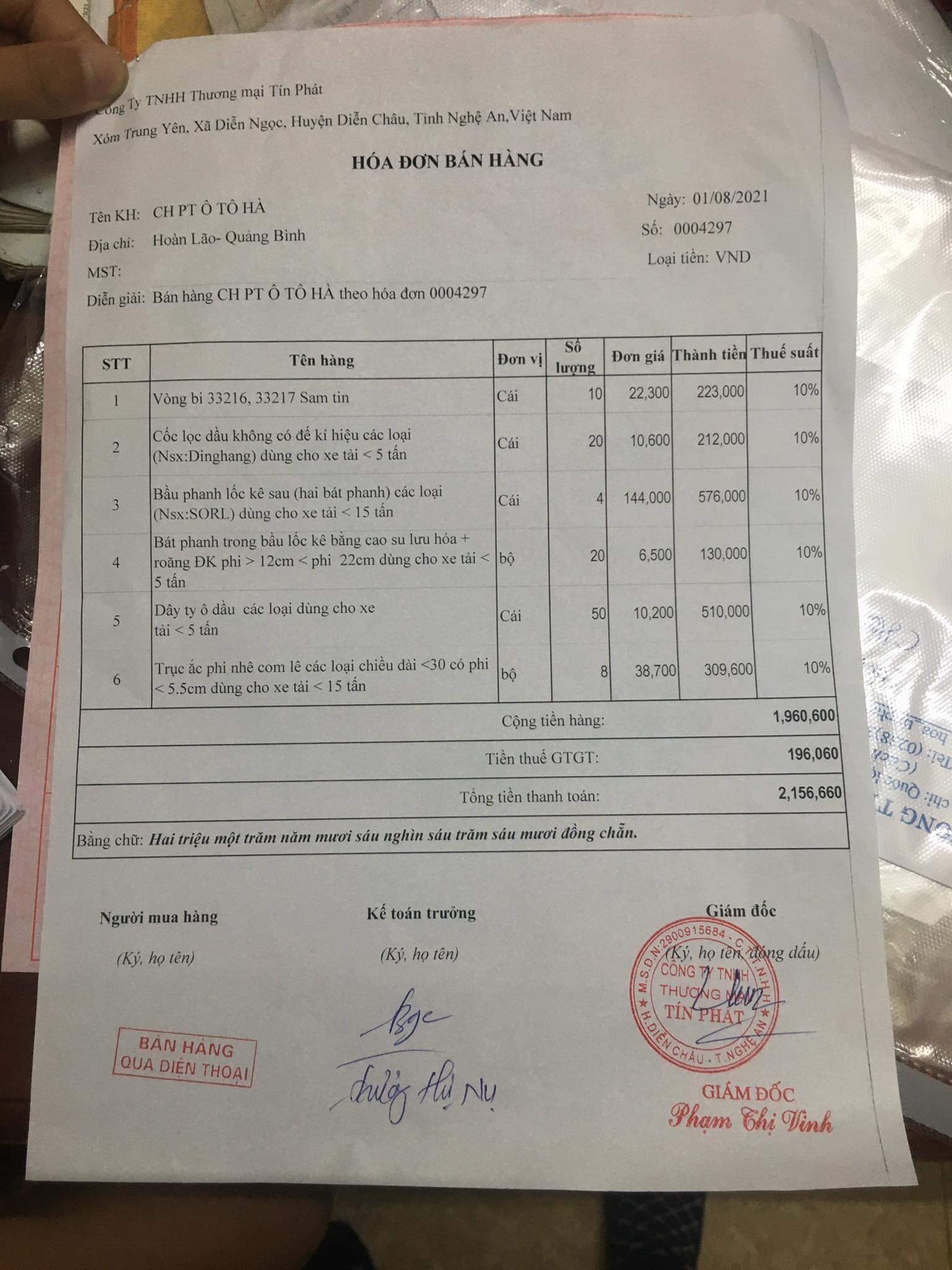
*Bổ sung thêm:* MaHD



Hình 1. : Mô hình ER – BÁN HÀNG

**Hóa đơn nhập hàng (Phiếu nhập hàng)**

* Các thực thể



Hình 1. : Hóa đơn nhập hàng

|  |  |  |
| --- | --- | --- |
| Từ trong HSDL | Từ rõ nghĩa | Viết tắt |
| Số | Mã hóa đơn nhập hàng /  Mã phiếu nhập hàng | MaPNH |
| ~~Tên hàng~~ | ~~Tên hàng~~ | ~~TenH~~ |
| ~~Quy cách hàng~~ | ~~Loại hàng~~ | ~~MaLH~~ |
| Ngày nhập hàng | Ngày nhập hàng | NgayNhap |
| ~~Người nhập hàng~~ | ~~Nhân viên~~ | ~~TenNV~~ |
| Nhà cung cấp | Nhà cung cấp | TenNCC |
| Địa chỉ nhà cung cấp | Địa chỉ nhà cung cấp | DiaChiNCC |
| ~~Đơn vị tính~~ | ~~Đơn vị~~ | ~~DonVi~~ |
| Số lượng | Số lượng nhập | SoLuongNhap |
| ~~Đơn giá~~ | ~~Đơn giá nhập~~ | ~~DonGiaNhap~~ |
| Thành tiền | Thành tiền | ThanhTien |
| Thuế GTGT | VAT | VAT |
| Cộng tiền hàng | Tổng tiền | TongTien |
| Tổng tiền thanh toán | Tổng cộng | TongCong |

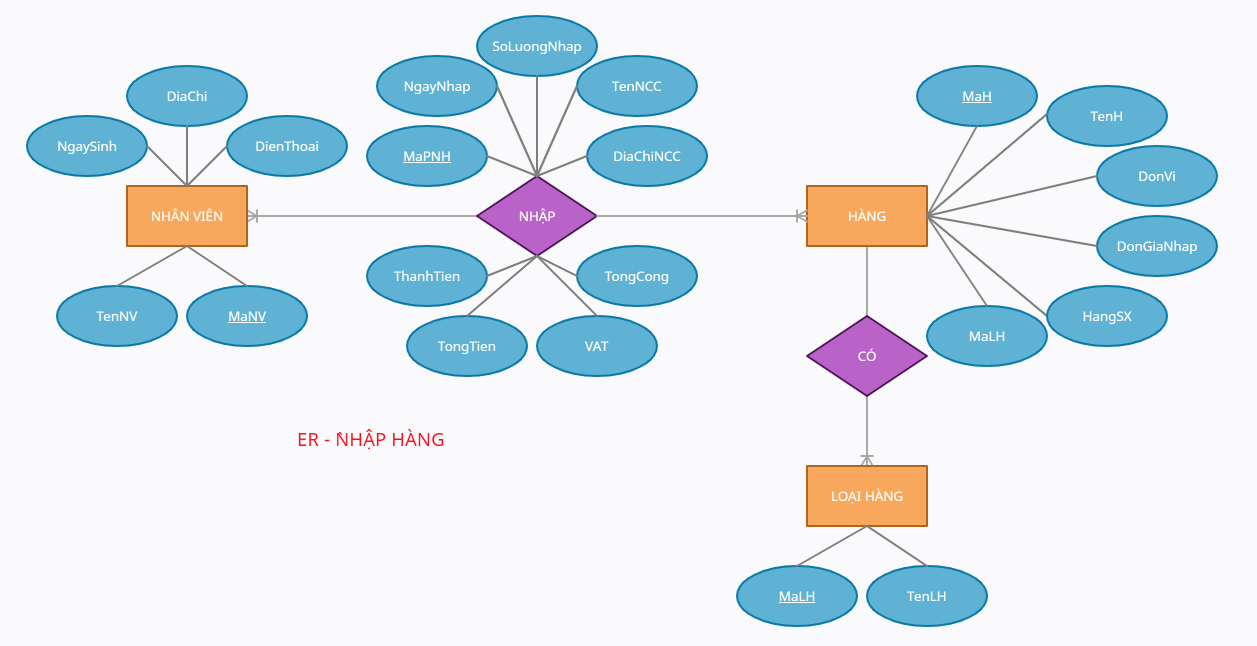
Bảng 1. : Bảng dữ liệu từ Hóa đơn nhập hàng

* HÀNG (MaH, TenH, DonVi, DonGiaNhap, HangSX, *MaLH*)
* LOẠI HÀNG (MaLH, TenLH)
* NHÂN VIÊN (MaNV, TenNV, NgaySinh, DiaChi, DienThoai)

*Bổ sung thêm:* MaKH, DienThoai, MaH, HangSX, MaLH, MaNV, NgaySinh, DiaChi, DienThoai

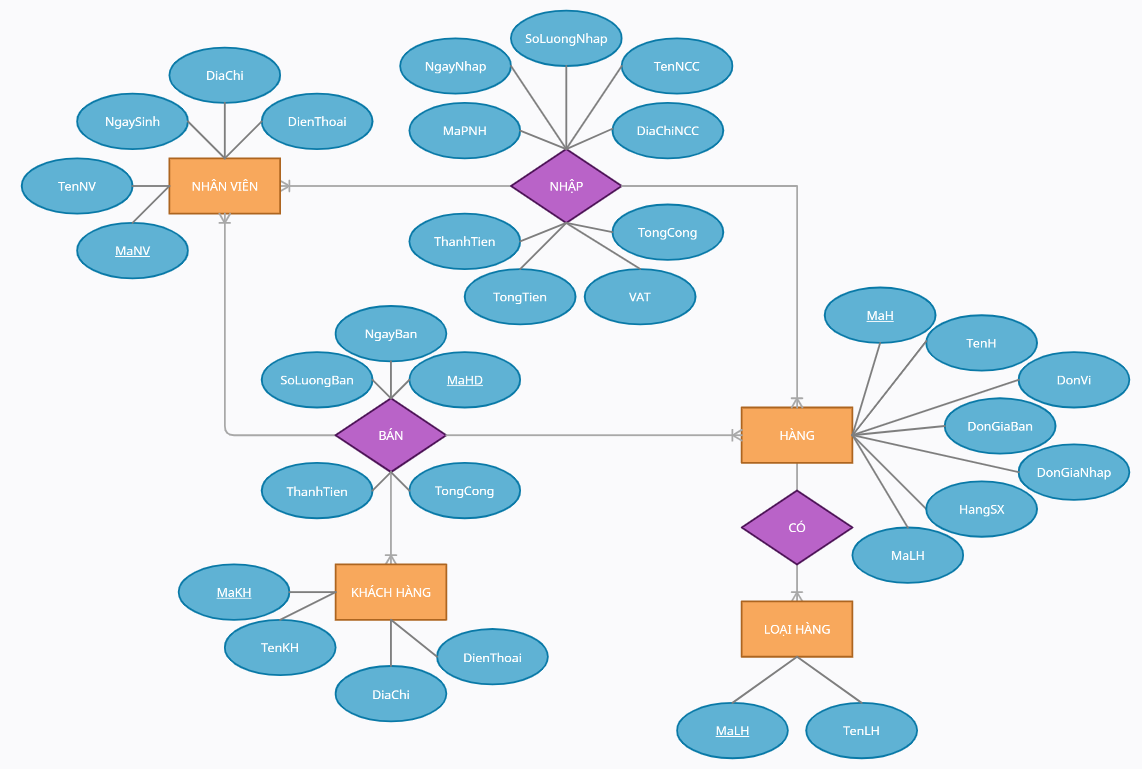
* Quan hệ

NHẬP (MaPNH, TenNCC, DiaChiNCC, NgayNhap, SoLuongNhap, ThanhTien, VAT, TongTien, TongCong)



Hình 1. : Mô hình ER – NHẬP HÀNG

**Tích hợp ER – NHẬP HÀNG và ER – BÁN HÀNG**



Hình 1. : Mô hình tích hợp ER

### Thiết kế cơ sở dữ liệu mức logic

Chuyển thực thể, thu được:

* HANG (MaH, TenH, DonVi, HangSX, DonGiaNhap, DonGiaBan, MaLH)
* LOAIHANG (MaLH, TenLH)
* NHANVIEN (MaNV, TenNV, NgaySinh, DiaChi, DienThoai)
* KHACHHANG (MaKH, TenKH, DiaChi, DienThoai)

Chuyển quan hệ, thu được:

* BAN (MaHD, NgayBan, SoLuongBan, ThanhTien, TongCong, MaNV, MaKH, MaH)
* NHAP (MaPNH, TenNCC, DiaChiNCC, NgayNhap, SoLuongNhap, ThanhTien, VAT, TongTien, TongCong, MaNV, MaH)

Chuẩn hóa quan hệ BAN, thu được:

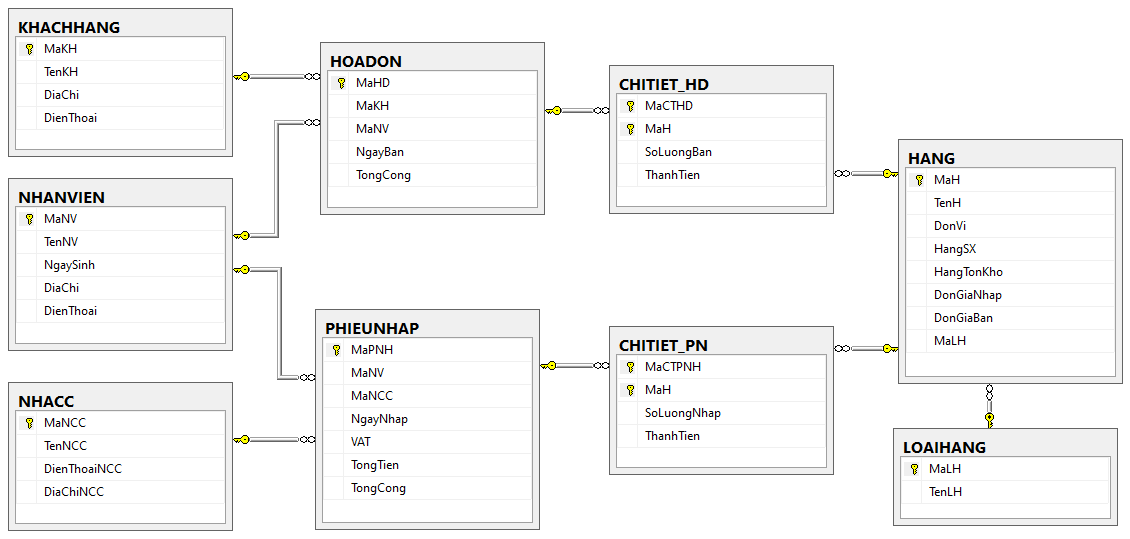
* HOADON (MaHD, *MaKH*, *MaNV*, NgayBan,TongCong)
* CHITIET\_HD (MaHD, MaH, SoLuongBan, ThanhTien)

Chuẩn hóa quan hệ NHAP, thu được:

* PHIEUNHAP (MaPNH, *MaNV*, *MaNCC*, NgayNhap, VAT, TongTien, TongCong)
* CHITIET\_PN (MaPNH, MaH, SoLuongNhap, ThanhTien)
* NHACC (MaNCC, TenNCC, DienThoaiNCC, DiaChiNCC)

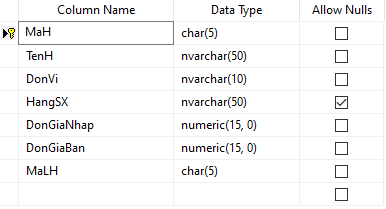
*Bổ sung thêm:* MaNCC, DienThoaiNCC

Sơ đồ ràng buộc quan hệ:

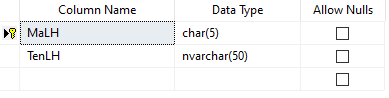


Hình 1. : Sơ đồ cơ sở dữ liệu

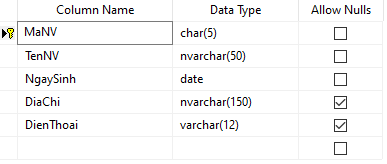
### Thiết kế cơ sở dữ liệu mức vật lý



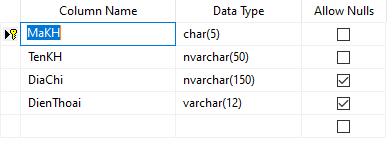
Bảng 1. : Bảng hàng



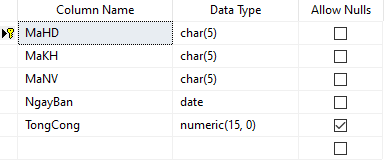
Bảng 1. : Bảng loại hàng



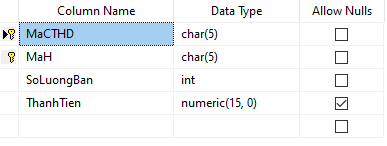
Bảng 1. : Bảng nhân viên



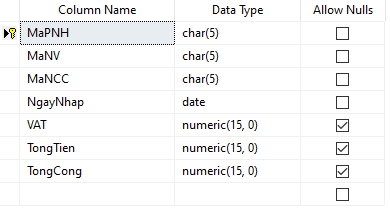
Bảng 1. : Bảng khách hàng



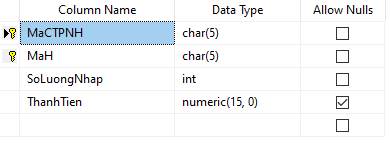
Bảng 1. : Bảng hóa đơn



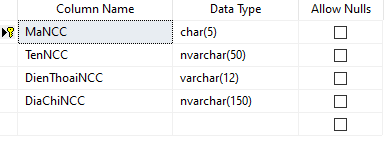
Bảng 1. : Bảng chi tiết hóa đơn



Bảng 1. : Bảng phiếu nhập



Bảng 1. : Bảng chi tiết phiếu nhập

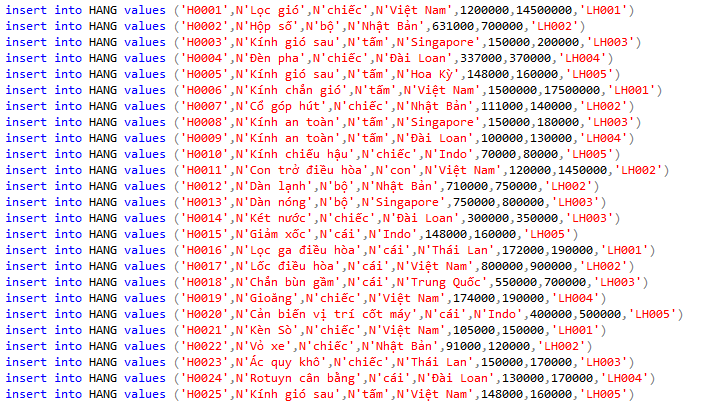


Bảng 1. : Bảng nhà cung cấp

## Xây dựng cơ sở dữ liệu

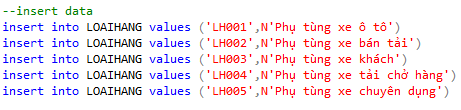
### Bảng HANG

Tạo dữ liệu cho bảng HANG

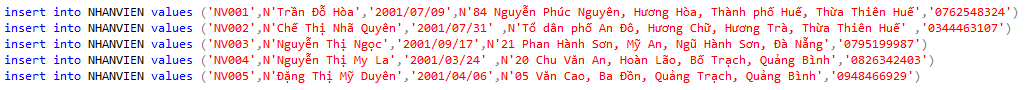


### Bảng LOAIHANG

Tạo dữ liệu cho bảng LOAIHANG

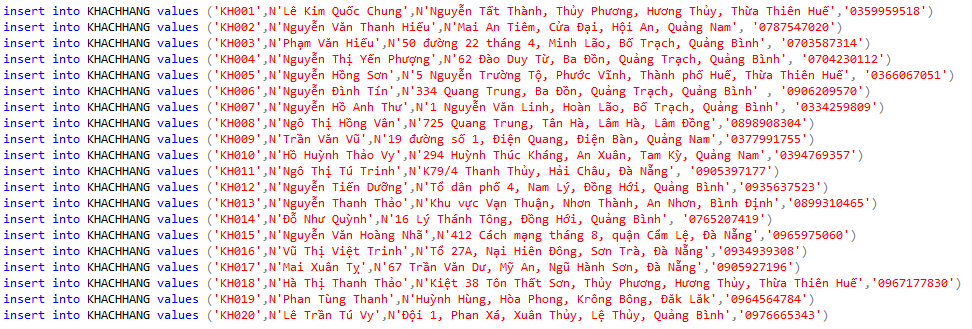


### Bảng NHANVIEN

Tạo dữ liệu cho bảng NHANVIEN

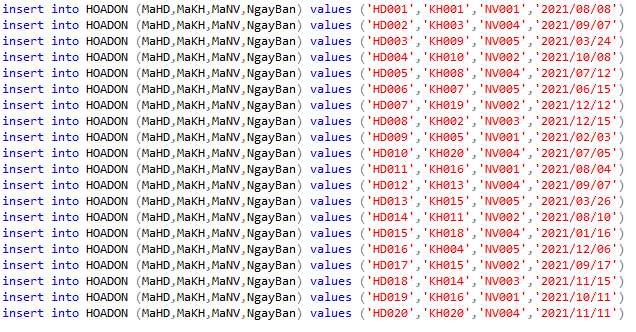
### Bảng KHACHHANG

Tạo dữ liệu cho bảng KHACHHANG



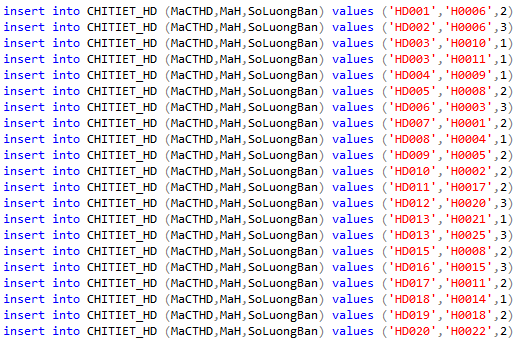
### Bảng HOADON

Tạo dữ liệu cho bảng HOADON



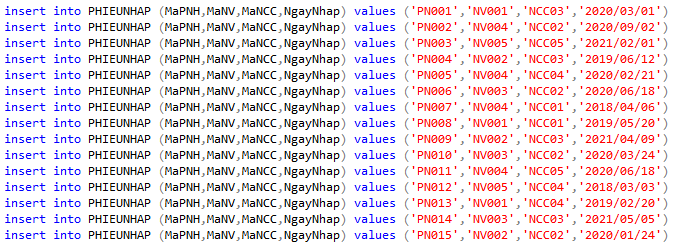
### Bảng CHITIET\_HD

Tạo dữ liệu cho bảng CHITIET\_HD



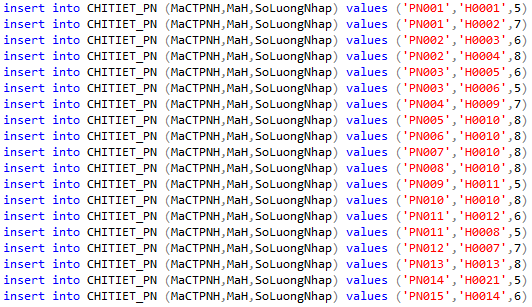
### Bảng PHIEUNHAP

Tạo dữ liệu cho bảng PHIEUNHAP



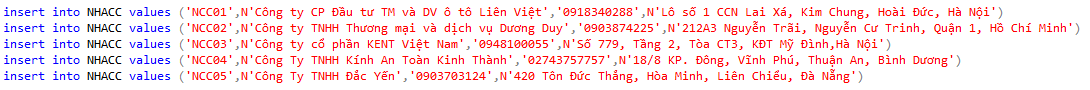
### Bảng CHITIET\_PN

Tạo dữ liệu cho bảng CHITIET\_PN



### Bảng NHACC

Tạo dữ liệu cho bảng NHACC



## Xác định và tạo các index cho các thuộc tính cần thiết

Index (chỉ mục) là một cấu trúc liên kết với một bảng hoặc một view dùng để tăng tốc độ truy vấn dữ liệu. Index chứa các khóa được tạo từ một hay nhiều cột và được lưu trữ dưới dạng Btree.

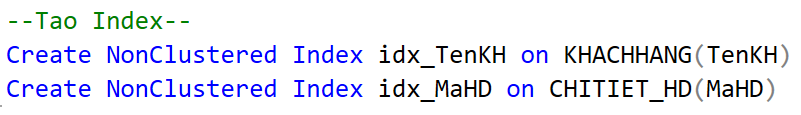
Index giúp tăng tốc các truy vấn select và các mệnh đề where, nhưng nó làm chậm dữ liệu nhập vào với các câu lệnh update và insert. Các chỉ mục có thể được tạo ra hoặc bỏ đi mà không ảnh hưởng đến dữ liệu.

Index chia thành 2 loại: Clustered index và Non-clustered index.

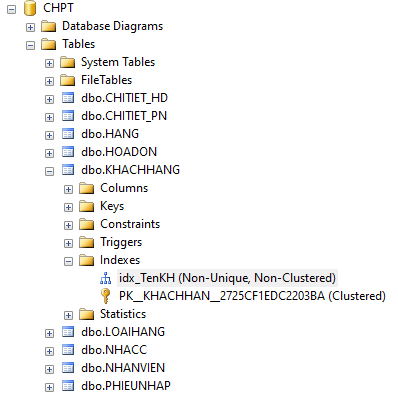
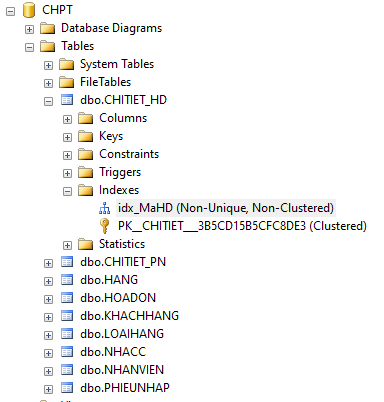
Ở đây, trong cơ sở dữ liệu CHPT của mình, nhóm em tạo Non-clustered index trên cột TenKH trong bảng KHACHHANG và trên cột MaHD ở bảng CHITIET\_HD. Bởi vì:

* Truy vấn đến cột TenKH và MaHD là rất nhiều trong quá trình hoạt động của cửa hàng.
* Những cột này là cột mà có giá trị NOT NULL

🡪 Vì vậy chọn các thuộc tính này để Index giúp hiệu năng truy vấn dữ liệu nhanh hơn.



Index thành công

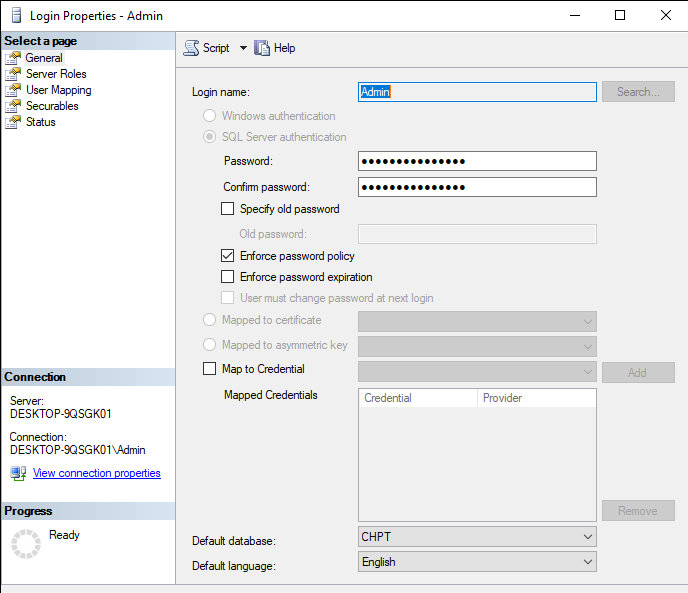
 

## Xây dựng cơ chế bảo mật tương ứng cho cơ sở dữ liệu

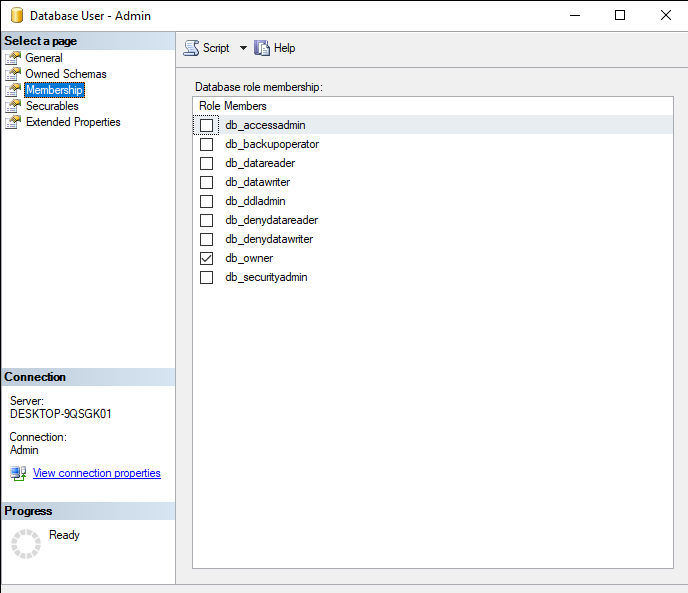
Cơ chế bảo mật:

* Authentication: xác thực người dùng có đươch phép kết nối với SQL Sever. SQL Server cho phép xác thực đăng nhập vào hệ thống bằng hai cơ chế là windows authentication và SQL Server authentication. Ở bài này, nhóm đã chọn cơ chế xác thực SQL Server Authentication khi tạo một login mới, vì vậy nhóm đã tạo ra tài khoản login name và password để đăng nhập của SQL Sever.
* Authorization: ủy quyền cho người sử dụng
* Các bước thực hiện như sau:

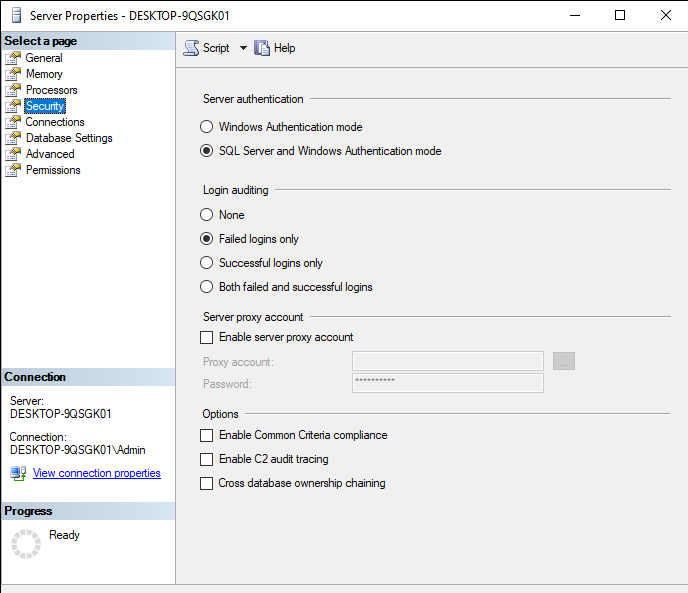
Bước 1: Tạo tài khoản Login có tên tài khoản là Admin



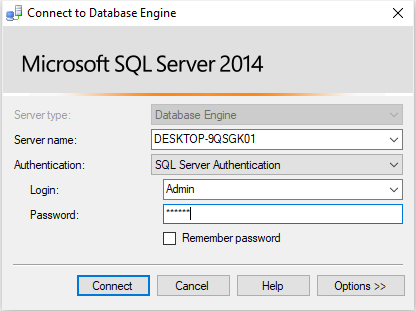
Bước 2: Phân quyền cho CSDL cho tài khoản vừa tạo



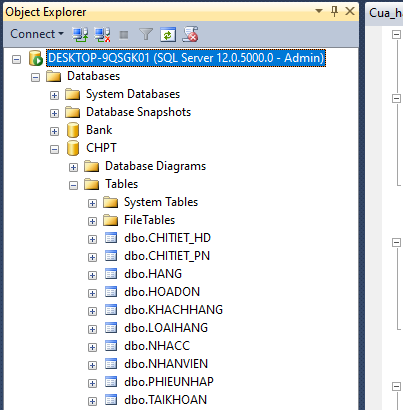
Bước 3: Thiết lập chế độ chứng thực Windows và SQL Server Authentication mode



Bước 4: Đăng nhập vào tài khoản vừa tạo ở trên



🡪 Đăng nhập thành công:



## Xây dựng cơ chế backup dữ liệu phù hợp cho cơ sở dữ liệu

Backup database là tiến trình bảo vệ CSDL giảm sự mất mát dữ liệu, được thực hiện để lưu dữ liệu, tạo ra một bản sao của tình trạng CSDL hiện tại và được thực hiện thường xuyên. Phòng các trường hợp máy chủ bị sự cố như hư ổ cứng hoặc dữ liệu bị mất do người dùng vô tình hoặc cố ý xoá,…

### Full backup

Loại backup này sẽ backup dữ liệu đầy đủ nhất, có thể phục hồi CSDL về bất kì thời điểm nào đó trong quá khứ, vì vậy máy chủ sẽ mất nhiều thời gian để thực hiện nếu database lớn. Đây là loại backup cơ bản và thường sử dụng trước các loại backup khác

### Differential backup

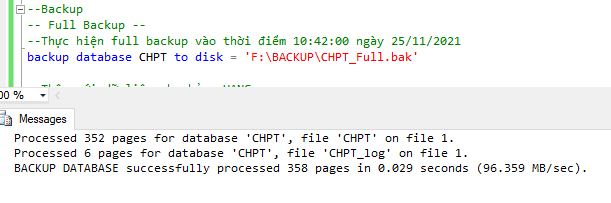
Loại backup này sẽ backup toàn bộ tất cả những thay đổi dữ liệu trên CSDL kể từ lần full backup gần nhất. Vì vậy, trước khi thực hiện different backup thì full backup phải được thực hiện trước. 🡪 Sử dụng loại backup này sẽ tiết kiệm được thời gian backup dữ liệu.

### Transaction logs backup

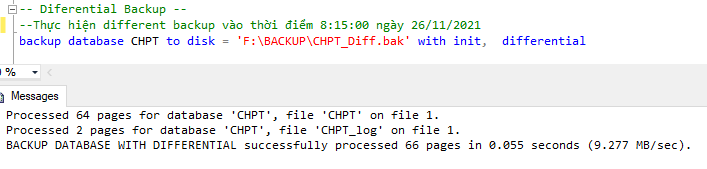
Loại backup này sẽ backup các transaction log kể từ lần full backup sau cùng hoặc log backup sau cùng. Nó cho phép phục hồi CSDL về một thời điểm bất kỳ trong quá khứ ví dụ như mỗi giờ trong ngày.

**Backup thủ công:**

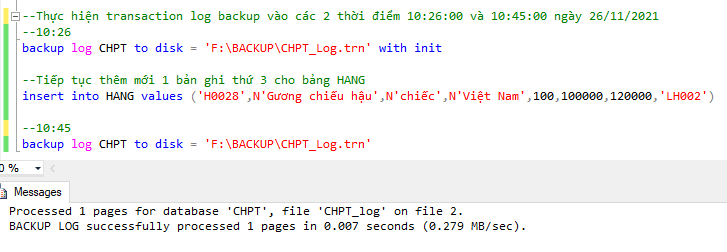
Full backup

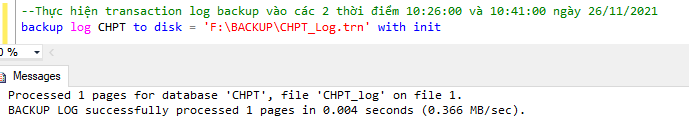


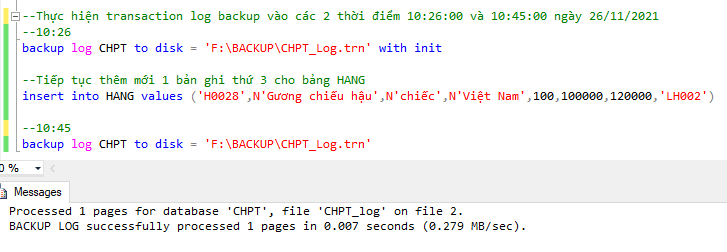
Diff backup



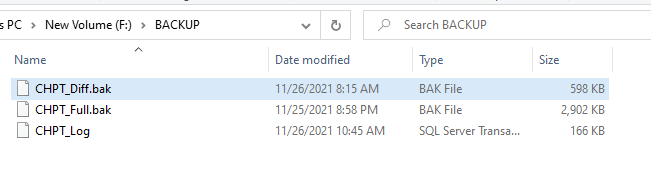
Transaction logs backup





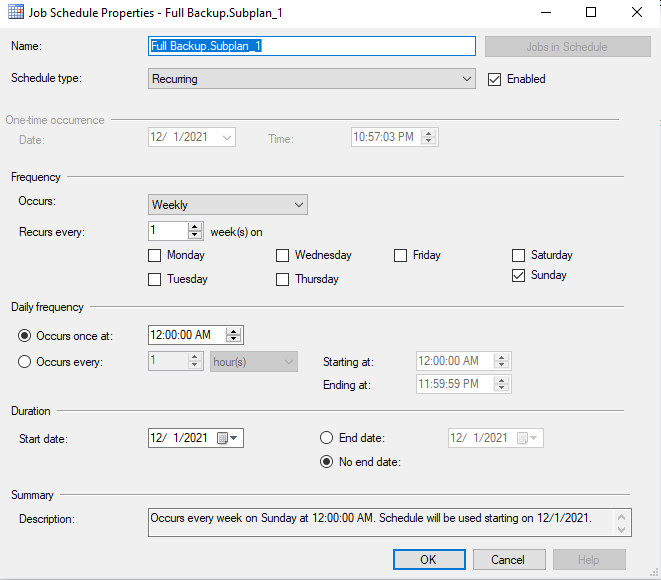


🡪 Các file nằm trong đường dẫn sau khi backup thành công:

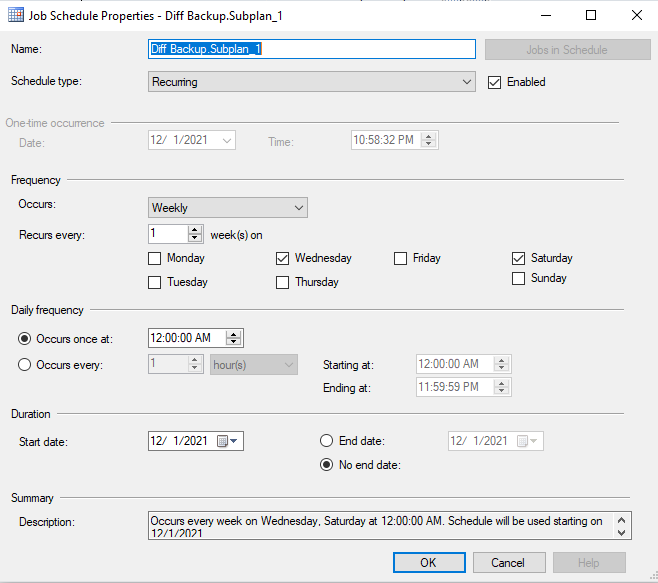


**Backup tự động:**

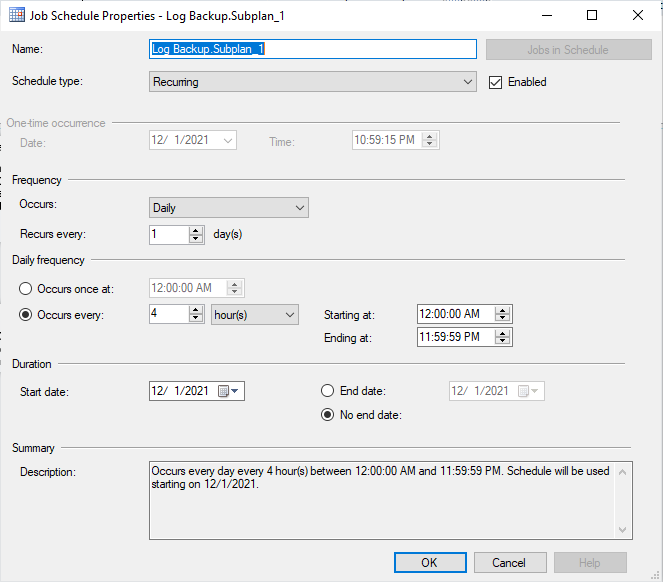
Full backup theo tuần, một tuần một lần vào chủ nhật lúc 12:00 AM



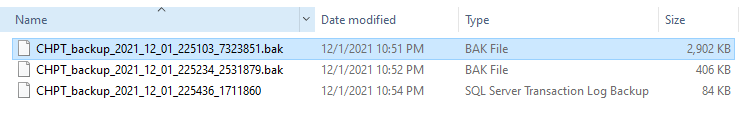
Diff backup một tuần hai lần vào thứ 4 và thứ 7 lúc 12:00 AM



Transaction logs backup mỗi ngày một lần cách nhau 4h đồng hồ

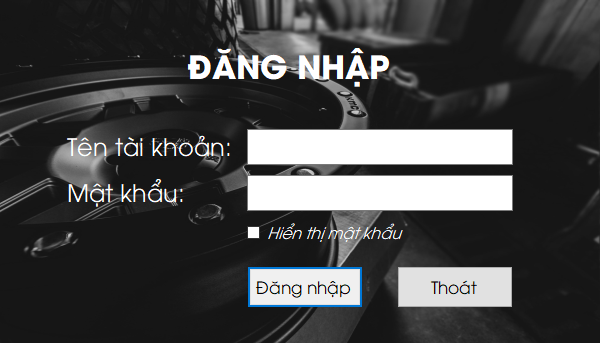


🡪 Các file nằm trong đường dẫn sau khi backup thành công:



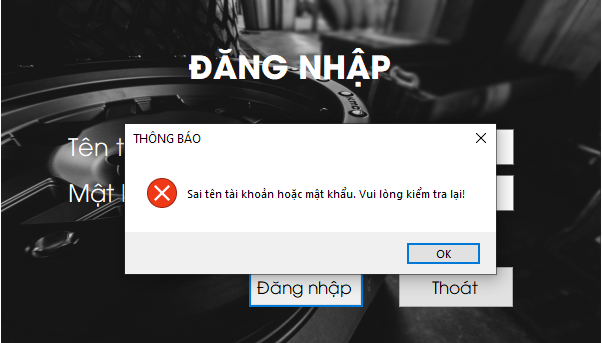
## Phát triển ứng dụng mang tên TESTDB – Sử dụng ngôn ngữ C#

### Form đăng nhập hệ thống



Hình 6. 1: Form đăng nhập hệ thống

Người dùng sử dụng tên đăng nhập và mật khẩu được cấp để đăng nhập khi muốn sử dụng các chứng năng có trong hệ thống. Nếu nhập tài khoản đúng thì đi đến giao diện chính của hệ thống. Nêu sai hệ thống sẽ thông báo *“Sai tên tài khoản hoặc mật khẩu. Vui lòng kiểm tra lại!”*, trường hợp này có thể nhập lại cho chính xác hoặc nhấp chuột vào nút *“Thoát”* để thoát khỏi hệ thống.



Hình 6. : Form thông báo đăng nhập sai

### Giao diện form chính

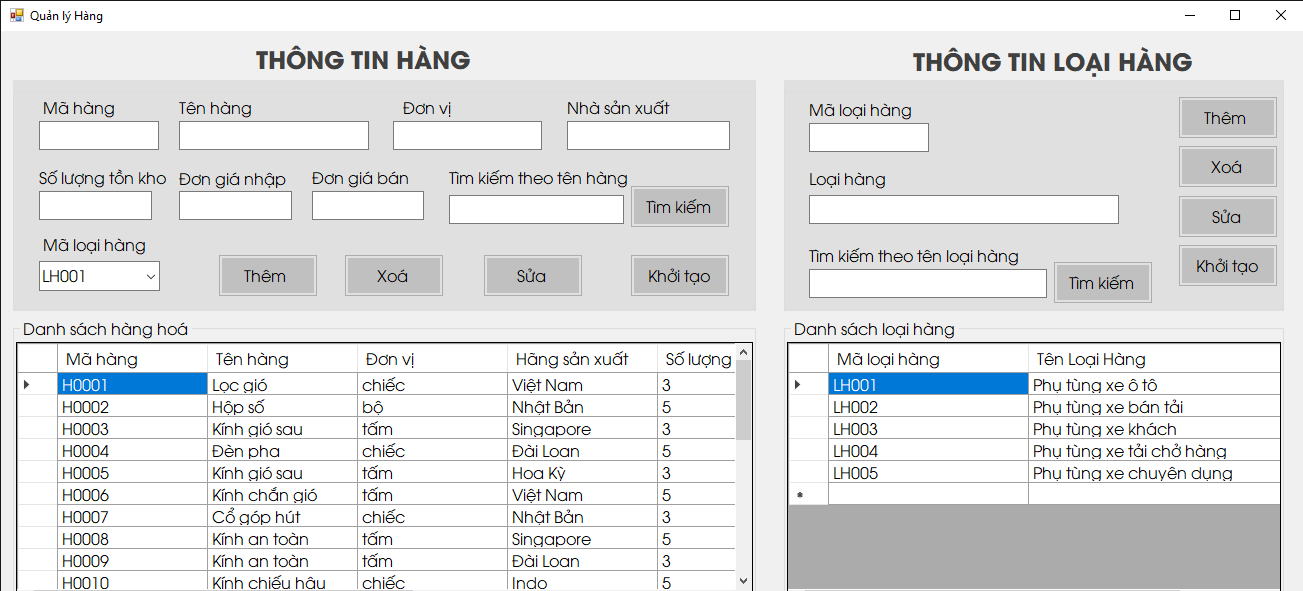


Hình 6. : Form giao diện chính

Tại đây, người dùng sẽ lựa chọn các danh mục mà mình muốn thực hiện các chức năng trên form bằng cách nhấn vào MenuStrip *Quản lý*. Sẽ hiện ra các danh mục bao gồm: *Hàng hóa, Hóa đơn, Phiếu nhập, Khách hàng, Nhà cung cấp, Nhân viên.* Nếu không sử dụng nữa, người dùng nhấn vào MenuStrip *Đăng xuất* để thoát khỏi ứng dụng.

### Quản lý Hàng hóa

Người dùng nhấn chọn **Hàng hóa,** hệ thống sẽ xuất hiện bảng gồm *Thông tin hàng* và *Thông tin loại hàng* như hình bên dưới:



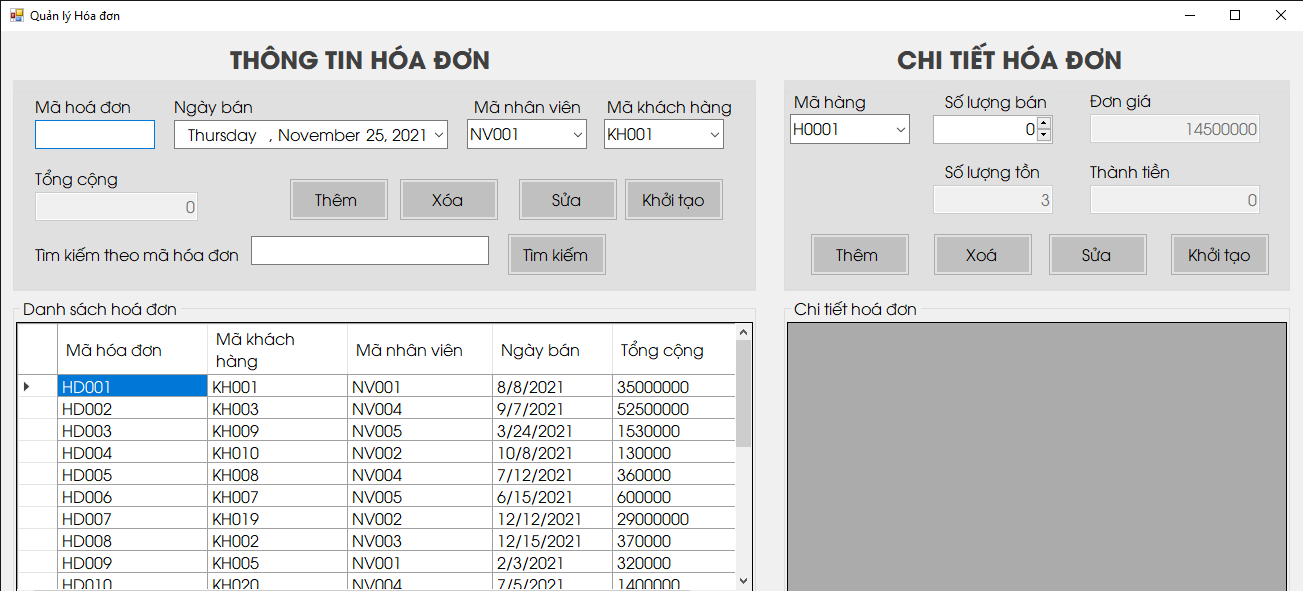
Hình 6. : Form thông tin hàng và loại hàng

Các công cụ và chức năng tương ứng:

* **Các textboox** *- Mã hàng, Tên hàng, Đơn vị, Nhà sản xuất, Số lượng tồn kho, Đơn giá nhập, Đơn giá bán, Tìm kiếm theo tên hàng (Hàng) và Mã loại hàng, Tên loại hàng, Tìm kiếm theo tên loại hàng (Loại hàng)*: Trước hết, người dùng cần phải nhập dữ liệu vào các textbox khi uốn thực hiện được các chức năng thêm, sửa, xóa, tìm kiếm.
* **Combobox** *- Mã loại hàng (Thông tin hàng)*: Người dùng chọn mã loại hàng đã có sẵn khi muốn thực hiện được các chức năng thêm, sửa, xóa
* **Button Thêm**: Sau khi nhập các giá trị vào textbox, nhấn chuột vào nút **Thêm** nếu muốn thêm một hàng hóa hay loại hàng:
* Nếu không nhập đầy đủ thông tin ở các ô trống thì hệ thống sẽ thông báo lỗi *“Vui lòng điền đầy đủ thông tin!”*
* Nếu nhập đầy đủ và hợp lệ thì hệ thống sẽ thông báo “*Thêm mới thành công*”.
* **Button Xóa**: Khi hết hoặc không bán hàng hóa, loại hàng đó nữa và muốn xóa chúng đi, người dùng chỉ cần nhấp chọn hàng hóa hoặc loại hàng cần xóa ở bảng DataGirdView rồi nhấn vào nút **Xóa**. Hệ thống hiện lên thông báo “*Xóa thành công*”.
* **Button Sửa**: Khi muốn sửa thông tin hàng hóa hoặc loại hàng, người dùng chọn hàng hóa hoặc loại hàng hóa cần sửa ở bảng DataGirdView. 🡪 Thông tin của hàng hóa sẽ hiển thị ở các textbox. Tiến hành việc sửa bằng cách nhập lại dữ liệu cần sửa vào các ô tương ứng rồi nhấn vào nút **Sửa**. Hệ thống hiện lên thông báo “Sửa thành công”. Không được sửa Mã hàng hay Mã loại hàng.
* **Button Tìm kiếm**: Khi muốn tìm kiếm một hàng hóa/loại hàng nào đó, người dùng nhập giá trị cần tìm vào textbox *Tìm kiếm theo tên hàng*, *Tìm kiếm theo tên loại hàng* sau đó nhấn nút **Tìm kiếm**
* **Button Khởi tạo**: Người dùng nhấn nút **Khởi tạo** để hủy bỏ thao tác được làm từ trước
* **Bảng DataGirdView** *- Danh sách hàng hóa và danh sách loại hàng*: Hiển thị thông tin dữ liệu của hàng và loại hàng

### Quản lý Hóa đơn

Người dùng nhấn chọn **Hóa đơn,** hệ thống sẽ xuất hiện bảng gồm *Thông tin hóa đơn* và *Chi tiết hóa đơn* như hình bên dưới:



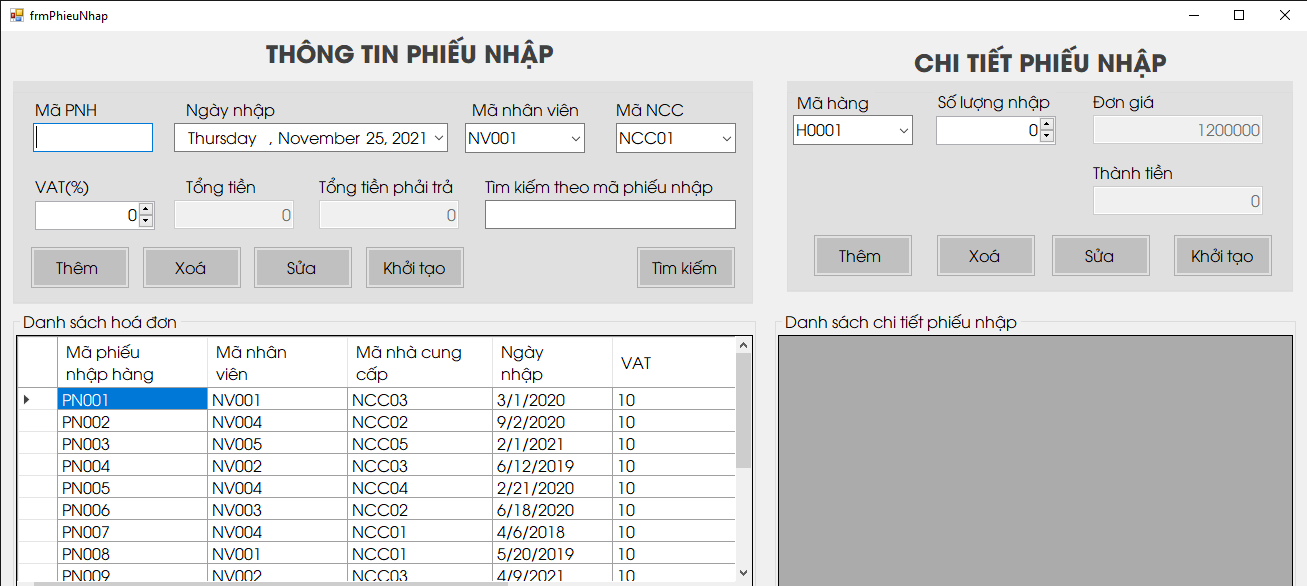
Hình 6. : Form thông tin Hóa đơn và chi tiết hóa đơn

Các công cụ và chức năng tương ứng:

* **Các textboox** *- Mã hóa đơn, Tìm kiếm theo mã hóa đơn (Hóa đơn)*: Người dùng cần phải nhập dữ liệu vào các textbox khi muốn thực hiện được các chức năng thêm, sửa, xóa, tìm kiếm.
* **Các combobox** *- Mã nhân viên, Mã khách hàng (Hóa đơn) và Mã hàng (Chi tiết Hóa đơn)*: Người dùng chọn mã nhân viên, Mã khách hàng, Mã hàng đã có sẵn khi muốn thực hiện được các chức năng thêm, sửa, xóa.
* **DateTimePicket** *- Ngày bán (Hóa đơn)*: Người dùng chọn ngày bán khi muốn thực hiện được các chức năng thêm, sửa, xóa.
* **NumericUpDown** *- Số lượng bán (Chi tiết Hóa đơn)*: Người dùng tăng, giảm hoặc nhập số lượng bán vào khi muốn thực hiện được các chức năng thêm, sửa, xóa.
* **Button Thêm**: Sau khi nhập các giá trị vào, nhấn chuột vào nút **Thêm** nếu muốn thêm một hóa đơn. Nếu muốn thêm chi tiết tiết hóa đơn, trước hết phải chọn vào hóa đơn tương ứng rồi mới thêm được chi tiết hóa đơn.
* **Button Xóa**: Muốn xóa hóa đơn, người dùng phải xóa chi tiết hóa đơn trước. Người dùng chỉ cần nhấp chọn chi tiết hóa đơn hoặc hóa đơn ở bảng DataGirdView rồi nhấn vào nút **Xóa**.
* **Button Sửa**: Người dùng chọn chi tiết hóa đơn/hóa đơn muốn sửa rồi nhấn nút **Sửa**. Không thể sửa Mã hóa đơn, Tổng Cộng, Đơn giá, Số lượng tồn, Thành tiền
* **Buttton Tìm kiếm**: Người dùng nhập giá trị cần tìm vào textbox *Tìm kiếm theo mã hóa đơn* sau đó nhấn nút **Tìm kiếm**
* **Button Khởi tạo**: Người dùng nhấn nút **Khởi tạo** để hủy bỏ thao tác được làm từ trước
* **Bảng DataGirdView** *- Danh sách hóa đơn và chi tiết hóa đơn*: Hiển thị thông tin của hóa đơn và chi tiết hóa đơn

### Quản lý Phiếu nhập

Người dùng nhấn chọn **Phiếu nhập,** hệ thống sẽ xuất hiện bảng gồm *Thông tin phiếu nhập* và *Chi tiết phiếu nhập* như hình bên dưới:

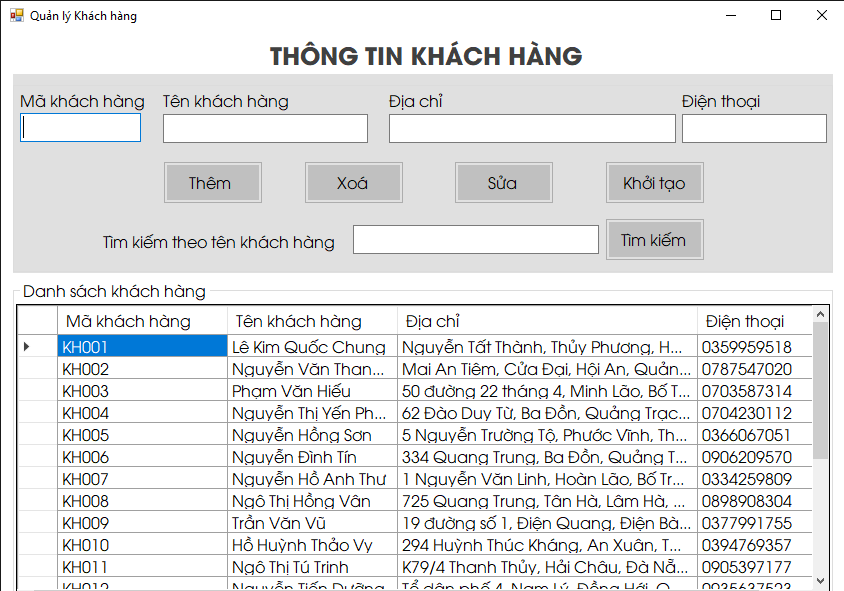


Hình 6. : Form thông tin phiếu nhập và chi tiết tiết phiếu nhập

Các chức năng thêm, sửa, xóa, khởi tạo và cũng như tìm kiếm của bảng phiếu nhập cũng thực hiện các thao tác tương tư như bảng Hóa đơn đã nêu ở trên.

### Quản lý Khách hàng

Người dùng nhấn chọn **Khách hàng,** hệ thống sẽ xuất hiện bảng gồm *Thông tin Khách hàng* hình bên dưới:



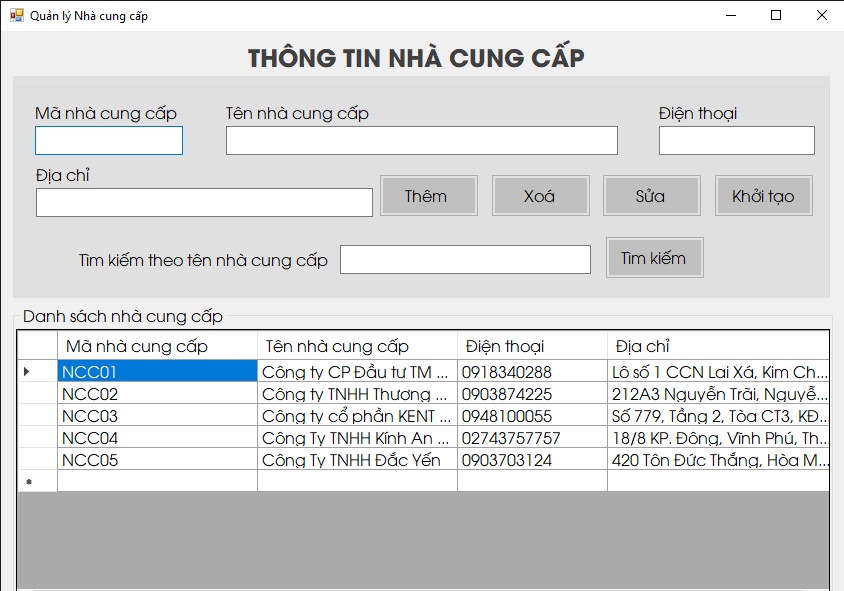
Hình 6. : Form thông tin Khách hàng

Các công cụ và chức năng tương ứng:

* **Các textboox** *- Mã khách hàng, Tên khách hàng, Địa chỉ, Điện thoại, Tìm kiếm theo tên khách hàng*: Người dùng cần phải nhập dữ liệu vào các textbox khi muốn thực hiện được các chức năng thêm, sửa, xóa, tìm kiếm.
* **Button Thêm**: Sau khi nhập các giá trị vào, nhấn chuột vào nút **Thêm** nếu muốn thêm một khách hàng.
* **Button Xóa**: Người dùng chỉ cần nhấp chọn khách hàng muốn xóa ở bảng DataGirdView rồi nhấn vào nút **Xóa**.
* **Button Sửa**: Người dùng chọn khách hàng muốn sửa rồi nhấn nút **Sửa**. Không thể sửa Mã khách hàng
* **Buttton Tìm kiếm**: Người dùng nhập giá trị cần tìm vào textbox *Tìm kiếm theo tên khách hàng* sau đó nhấn nút **Tìm kiếm**
* **Button Khởi tạo**: Người dùng nhấn nút **Khởi tạo** để hủy bỏ thao tác được làm từ trước.
* **Bảng DataGirdView** *- Danh sách hóa đơn và chi tiết hóa đơn*: Hiển thị thông tin của hóa đơn và chi tiết hóa đơn:
* Thêm, Sửa: dữ liệu tự cập nhật vào bảng
* Xóa: dữ liệu đã xóa không còn xuất hiện ở bảng
* Tìm kiếm: dữ liệu cần tìm sẽ hiển thị ở bảng

### Quản lý Nhà cung cấp

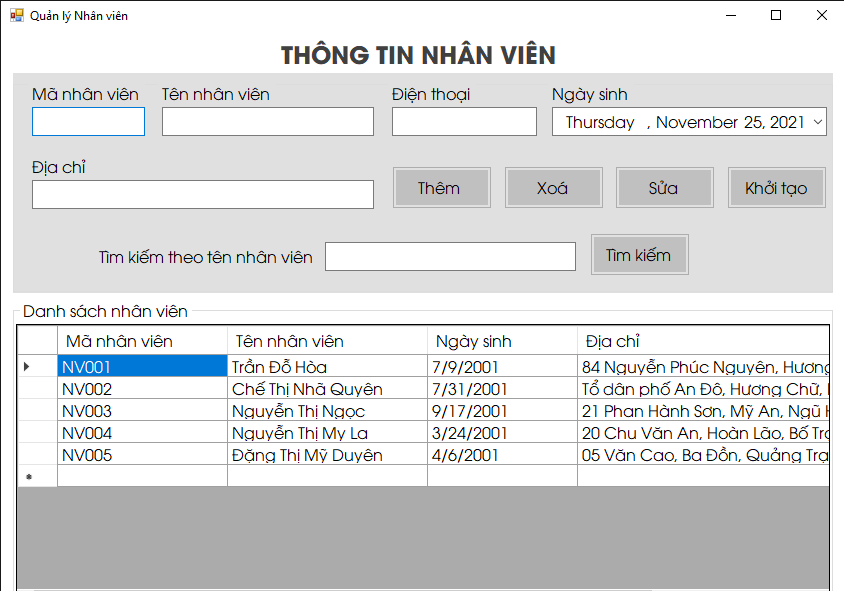
Người dùng nhấn chọn **Nhà cung cấp,** hệ thống sẽ xuất hiện bảng gồm *Thông tin Nhà cung cấp* như hình bên dưới. Các chức năng thêm, sửa, xóa, khởi tạo và cũng như tìm kiếm của bảng Nhà cung cấp cũng thực hiện các thao tác tương tư như bảng Khách hàng nêu ở trên.



Hình 6. 8: Form thông tin Nhà cung cấp

### Quản lý Nhân viên

Người dùng nhấn chọn **Nhân viên,** hệ thống sẽ xuất hiện bảng gồm *Thông tin Nhân viên* như hình bên dưới:



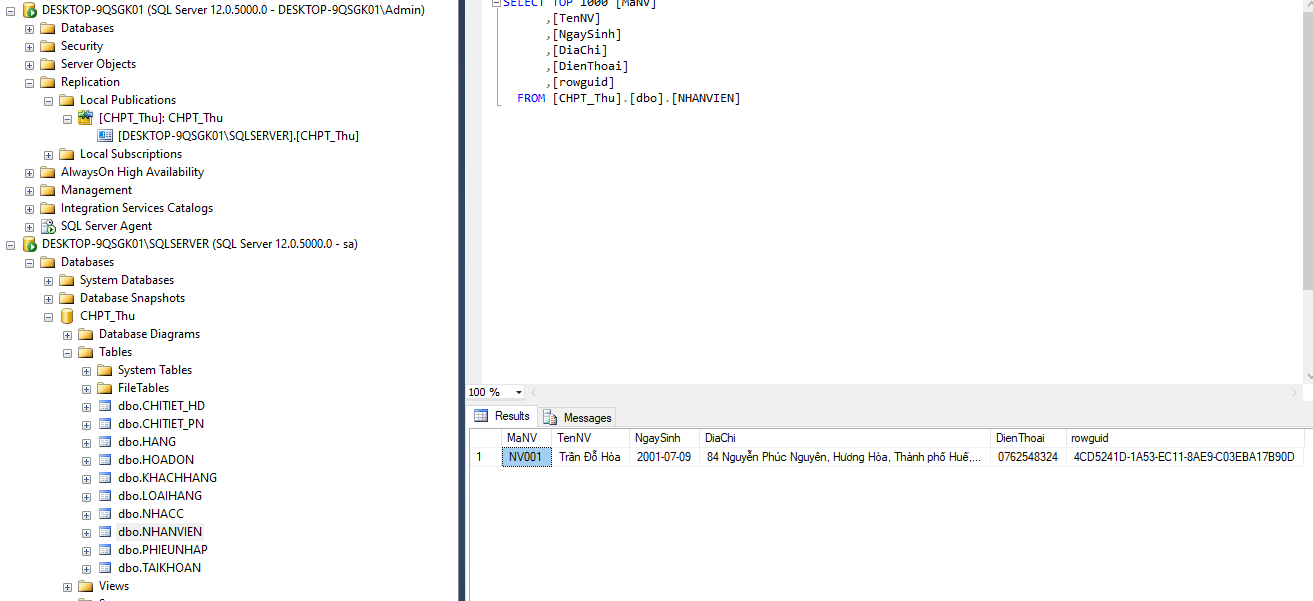
Hình 6. : Form thông tin nhân viên

Các chức năng thêm, sửa, xóa, khởi tạo và cũng như tìm kiếm của bảng Nhân viên cũng thực hiện các thao tác tương tư như bảng Khách hàng hay Nhà cung cấp đã nêu ở trên.

## Đưa ra phương án giải quyết cho vấn đề lượng dữ liệu rất lớn, không đủ lưu trữ trong một ổ đĩa

Phương án giải quyết: Phân tán dữ liệu. Đây là giải pháp chia nhỏ một database lớn thành nhiều database nhỏ.

Nhóm đã chọn phân tán dữ liệu theo chiều ngang, lấy bảng Nhân viên làm bảng gốc để phân tán. Dưới đây là hình ảnh phân tán dữ liệu NV001 sang một server con:



## Đề phòng phương án hệ thống CHPT bị tấn công bằng SQL Injection. Đưa ra phương án và thực thi phương án.

**SQL Injection** là một kỹ thuật lợi dụng những lỗ hổng về câu truy vấn của các ứng dụng. Được thực hiện bằng cách chèn thêm một đoạn [SQL](https://topdev.vn/blog/sql-la-gi/) để làm sai lệnh đi câu truy vấn ban đầu, từ đó có thể khai thác dữ liệu từ database. **SQL injection** có thể cho phép những kẻ tấn công thực hiện các thao tác như một người quản trị web, trên cơ sở dữ liệu của ứng dụng.

Sau khi tìm hiểu các phương án để đề phòng hệ thống bị tấn công bằng SQL Injection thì nhóm chọn phương án Prepared Statements (with Parameterized Queries). Vì:

* Khi sử dụng phương pháp này, các câu lệnh truy vấn được xác định sẵn rồi truyền từng tham số vào rồi truy vấn sau đó. Kiểu mã hóa này cho phép cơ sở dữ liệu phân biệt được đâu là dữ liệu và queries được người sử dụng đưa vào. Lý do quan trọng nhất để sử dụng truy vấn được tham số hóa là để tránh các cuộc tấn công SQL injection.

