|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| TRƯỜNG ĐẠI HỌC GIAO THÔNG VẬN TẢI  **KHOA CÔNG NGHỆ THÔNG TIN**    **BÁO CÁO BÀI TẬP LỚN**  **Lập trình trực quan**  **Nhóm 8**  **Đề tài: Quản lý bán kính mắt**  Giảng viên hướng dẫn : Cao Thị Luyên  Danh sách sinh viên:   |  |  | | --- | --- | | Họ tên | Mã sinh viên | | Trần Hữu Tâm | 222631136 | | Đoàn Hải Xuyên | 222631162 | | Phạm Công Đăng | 222631087 | | Trịnh Công Đức | 222601090 | | Trần Trường Thức | 222631147 | |

TRƯỜNG ĐẠI HỌC GIAO THÔNG VẬN TẢI

**KHOA CÔNG NGHỆ THÔNG TIN**



**BÁO CÁO BÀI TẬP LỚN**

**Lập trình trực quan**

**Nhóm 8**

**Đề tài: Quản lý bán kính mắt**

Giảng viên hướng dẫn : Cao Thị Luyên

Danh sách sinh viên:

|  |  |
| --- | --- |
| Họ tên | Mã sinh viên |
| Trần Hữu Tâm | 222631136 |
| Đoàn Hải Xuyên | 222631162 |
| Phạm Công Đăng | 222631087 |
| Trịnh Công Đức | 222601090 |
| Trần Trường Thức | 222631147 |

**Lời nói đầu**

Trong bối cảnh công nghệ ngày càng phát triển và nhu cầu quản lý các thông tin liên quan đến sức khỏe, đặc biệt là thị lực, việc xây dựng các phần mềm quản lý hiệu quả trở thành yếu tố quan trọng để nâng cao chất lượng dịch vụ. Đặc biệt, trong lĩnh vực quang học, việc quản lý bán kính mắt đóng vai trò then chốt trong việc đảm bảo khách hàng nhận được sản phẩm phù hợp và chính xác.

Báo cáo này trình bày về việc phát triển phần mềm WinForm phục vụ cho công tác quản lý bán kính mắt, nhằm hỗ trợ các cơ sở kinh doanh, cửa hàng kính mắt và các chuyên gia trong lĩnh vực quang học có thể dễ dàng quản lý thông tin liên quan đến khách hàng và các đơn hàng một cách hiệu quả. Phần mềm này không chỉ giúp lưu trữ và tra cứu thông tin về các loại kính mắt, bán kính, và các thông số kỹ thuật mà còn cung cấp các tính năng phân tích, báo cáo chi tiết giúp tối ưu hóa công việc quản lý.

Với mục tiêu giúp các doanh nghiệp, tổ chức cung cấp dịch vụ kính mắt hoạt động hiệu quả hơn, báo cáo sẽ đi vào phân tích các yêu cầu kỹ thuật, cấu trúc hệ thống, quy trình phát triển và các kết quả đạt được trong quá trình thực hiện dự án này.

Mục lục

[Chương 1: Giới thiệu đề tài: Quản lý bán kính 6](#_Toc182611016)

[1.1. Giới thiệu về đề tài 6](#_Toc182611017)

[1.2. Kế hoạch làm đề tài 6](#_Toc182611018)

[1.3. Các công nghệ sử dụng 10](#_Toc182611019)

[Chương 2: Mô hình hệ thống 12](#_Toc182611020)

[2.1. Các chức năng của hệ thống 12](#_Toc182611021)

[2.2. Các giao diện chính có trong hệ thống 12](#_Toc182611022)

[2.3. Mô hình lớp 25](#_Toc182611023)

[2.3.1. Lớp Account (Kế thừa từ lớp Form): 25](#_Toc182611024)

[2.3.2. Lớp Home (Kế thừa từ lớp Form) 26](#_Toc182611025)

[2.3.3. Lớp QuanLyDanhMucHangHoa (Kế thừa từ lớp Form) 27](#_Toc182611026)

[2.3.4. Lớp QuanLyHoaDonBan (Kế thừa từ lớp Form) 29](#_Toc182611027)

[2.3.5. Lớp QuanLyHoaDonNhap (Kế thừa từ lớp Form) 30](#_Toc182611028)

[2.3.6. Lớp QuanLyKhachHang (Kế thừa từ lớp Form) 31](#_Toc182611029)

[2.3.7. Lớp QuanLyNhaCungCap (Kế thừa từ lớp Form) 33](#_Toc182611030)

[2.3.8. Lớp QuanLyNhanVien (Kế thừa từ lớp Form) 34](#_Toc182611031)

[2.3.9. Lớp DoanhThu (Kế thừa từ lớp Form) 36](#_Toc182611032)

[2.4. Mô hình cơ sở dữ liệu 37](#_Toc182611033)

[2.5. Các tiện ích 39](#_Toc182611034)

[Chương 3: Phát triển hệ thống 41](#_Toc182611035)

[3.1. Lớp GUI 41](#_Toc182611036)

[3.1.1 Form Account: 41](#_Toc182611037)

[3.1.2 Form Home 42](#_Toc182611038)

[3.1.3. Form QuanLyDanhMucHangHoa 43](#_Toc182611039)

[3.1.4. Form QuanLyHoaDonBan 44](#_Toc182611040)

[3.1.5. Form QuanLyHoaDonNhap 45](#_Toc182611041)

[3.1.6. Form QuanLyKhachHang 47](#_Toc182611042)

[3.1.7. Form QuanLyNhaCungCap 49](#_Toc182611043)

[3.1.8. Form QuanLyNhanVien 50](#_Toc182611044)

[3.1.9. Form DoanhThu 52](#_Toc182611045)

[3.2. Lớp BUS 53](#_Toc182611046)

[3.2.1. Lớp Account (Xử lý Đăng nhập) 53](#_Toc182611047)

[3.2.2. Lớp Home (Form chính sau khi đăng nhập) 54](#_Toc182611048)

[3.2.3. Lớp QuanLyDanhMucHangHoa 55](#_Toc182611049)

[3.2.4. Lớp QuanLyHoaDonBan 56](#_Toc182611050)

[3.2.5. Lớp QuanLyHoaDonNhap 58](#_Toc182611051)

[3.2.6. Lớp QuanLyKhachHang 59](#_Toc182611052)

[3.2.7. Lớp QuanLyNhaCungCap 61](#_Toc182611053)

[3.2.8. Lớp QuanLyNhanVien 62](#_Toc182611054)

[3.2.9. Lớp DoanhThu 63](#_Toc182611055)

[3.3. Kết chương 64](#_Toc182611056)

[4. Kết luận 65](#_Toc182611057)

[4.1. Kết quả thu được 65](#_Toc182611058)

[4.2. Hạn chế và hướng phát triển của đề tài 65](#_Toc182611059)

[Danh mục tham khảo 66](#_Toc182611060)

# Chương 1: Giới thiệu đề tài Quản lý bán kính

## Giới thiệu về đề tài

**Mục đích và Ý nghĩa:**

Đề tài "Quản lý bán kính mắt" được xây dựng nhằm giải quyết các vấn đề trong quản lý kinh doanh tại các cửa hàng kính mắt, bao gồm quản lý thông tin sản phẩm, đơn hàng, khách hàng và các giao dịch. Hệ thống cung cấp giải pháp thay thế cho việc lưu trữ thủ công hoặc các phương pháp quản lý thủ công thiếu chính xác và không hiệu quả. Việc áp dụng hệ thống này giúp tăng cường tính nhất quán, giảm thiểu sai sót và tạo điều kiện thuận lợi cho việc chăm sóc khách hàng.

**Các chức năng chính:**

Hệ thống quản lý bán kính mắt sẽ bao gồm các chức năng chính sau:

* **Quản lý khách hàng**: Lưu trữ và tra cứu thông tin khách hàng (tên, tuổi, địa chỉ, lịch sử mua hàng, và các thông tin cần thiết cho chăm sóc khách hàng).
* **Quản lý sản phẩm kính mắt**: Bao gồm các thông tin về loại kính, giá bán, số lượng tồn kho và các đặc điểm riêng biệt như độ cận, độ lọc UV.
* **Quản lý đơn hàng và thanh toán**: Hỗ trợ ghi nhận thông tin đơn hàng, tính toán giá thành, quản lý tình trạng thanh toán, và cập nhật tồn kho.
* **Báo cáo doanh thu và thống kê**: Hệ thống có khả năng tạo các báo cáo về doanh thu theo ngày, tuần, tháng và các báo cáo thống kê sản phẩm bán chạy.

**Phạm vi và Đối tượng sử dụng**

Phần mềm được xây dựng chủ yếu để sử dụng tại các cửa hàng bán kính mắt hoặc các phòng khám mắt có bán kính. Đối tượng chính là nhân viên quản lý, nhân viên bán hàng và người quản lý cửa hàng.

## 1.2. Kế hoạch làm đề tài

Kế hoạch thực hiện đề tài bao gồm các bước chi tiết như sau:

**Giai đoạn 1: Khảo sát và thu thập yêu cầu**

1. **Mục tiêu**:
   * Hiểu rõ quy trình quản lý thông tin khách hàng và bán kính mắt tại cửa hàng kính mắt.
   * Thu thập các yêu cầu cụ thể từ chủ cửa hàng và nhân viên.
2. **Các hoạt động**:
   * **Khảo sát quy trình thực tế**:
     + Gặp gỡ và trao đổi với chủ cửa hàng để hiểu các bước trong việc tiếp nhận khách hàng, đo mắt, và quản lý thông tin.
     + Quan sát quá trình đo mắt, lưu trữ thông tin khách hàng, và xử lý đơn hàng.
   * **Xác định vấn đề và yêu cầu**:
     + Xác định các khó khăn trong việc quản lý thủ công, như mất mát dữ liệu, không theo dõi được lịch sử đo mắt.
     + Thu thập yêu cầu về các chức năng cần có: lưu trữ thông tin khách hàng, lưu thông số đo bán kính mắt, tìm kiếm nhanh thông tin, và tạo báo cáo đơn hàng.
   * **Tạo tài liệu yêu cầu**:
     + Ghi lại chi tiết các chức năng cần thiết và phạm vi hệ thống:
       - Quản lý thông tin khách hàng (thêm, sửa, xóa).
       - Quản lý thông số bán kính mắt theo từng lần đo.
       - Tìm kiếm thông tin khách hàng và lịch sử đo mắt.
       - Quản lý đơn hàng kính và báo cáo doanh thu.

**Giai đoạn 2: Phân tích và thiết kế hệ thống**

1. **Mục tiêu**:
   * Thiết kế hệ thống đơn giản nhưng hiệu quả, dễ dàng mở rộng.
   * Đảm bảo giao diện thân thiện, dễ sử dụng cho nhân viên cửa hàng.
2. **Các hoạt động**:
   * **Phân tích hệ thống**:
     + Xây dựng sơ đồ Use Case để minh họa các tác vụ chính như: đăng nhập, quản lý khách hàng, đo mắt, xử lý đơn hàng.
   * **Thiết kế cơ sở dữ liệu**:
     + Xây dựng mô hình ERD với các bảng chính:
       - **Khách hàng**: mã khách hàng, họ tên, số điện thoại, địa chỉ.
       - **Thông số đo mắt**: mã lần đo, mã khách hàng, ngày đo, thông số mắt trái/phải (SPH, CYL, AXIS), ghi chú.
       - **Đơn hàng kính**: mã đơn hàng, mã khách hàng, ngày đặt, loại kính, tổng tiền, trạng thái.
       - **Nhân viên**: mã nhân viên, tên, vai trò, thông tin liên hệ.
   * **Thiết kế giao diện người dùng (UI/UX)**:
     + Thiết kế mockup giao diện các form:
       - Form quản lý khách hàng (danh sách khách hàng, thêm/sửa/xóa).
       - Form đo mắt (thêm thông số đo mới, xem lịch sử đo).
       - Form quản lý đơn hàng (thêm đơn hàng, xem trạng thái).
       - Dashboard tổng quan (báo cáo doanh thu, danh sách đơn hàng chưa hoàn thành).
   * **Thiết kế sơ đồ lớp (Class Diagram)**:
     + Mô hình hóa các lớp đối tượng như: KhachHang, LanDo, DonHang, NhanVien.

**Giai đoạn 3: Phát triển hệ thống**

1. **Mục tiêu**:
   * Xây dựng hệ thống dựa trên Windows Forms, đáp ứng đầy đủ các chức năng đã thiết kế.
2. **Các hoạt động**:
   * **Tạo giao diện người dùng**:
     + Xây dựng các form:
       - Form đăng nhập với chức năng phân quyền (quản lý, nhân viên).
       - Form quản lý khách hàng: thêm mới, chỉnh sửa, và xóa thông tin khách hàng.
       - Form đo mắt: nhập thông số đo mới, xem lịch sử đo.
       - Form quản lý đơn hàng: xử lý đơn kính, cập nhật trạng thái (đã giao, đang chờ).
   * **Phát triển chức năng**:
     + Sử dụng ADO.NET hoặc Entity Framework để kết nối cơ sở dữ liệu SQL Server.
     + Lập trình các chức năng:
       - Thêm/sửa/xóa và tìm kiếm thông tin khách hàng.
       - Lưu thông số đo mắt kèm ngày đo và ghi chú.
       - Quản lý đơn hàng, bao gồm tính tổng tiền và trạng thái giao hàng.
       - Tạo báo cáo doanh thu theo tháng hoặc tuần.
   * **Tích hợp**:
     + Đảm bảo các phần của hệ thống hoạt động mượt mà, giao diện đồng nhất.

**Giai đoạn 4: Kiểm thử và hoàn thiện**

1. **Mục tiêu**:
   * Kiểm tra toàn bộ hệ thống để đảm bảo hoạt động ổn định, không lỗi.
   * Hoàn thiện các điều chỉnh dựa trên phản hồi.
2. **Các hoạt động**:
   * **Kiểm thử chức năng**:
     + Thử nghiệm từng form và tính năng (thêm khách hàng, lưu thông số đo, quản lý đơn hàng).
   * **Kiểm thử tích hợp**:
     + Đảm bảo các chức năng kết nối liền mạch (ví dụ: thêm khách hàng mới -> lưu lần đo -> tạo đơn hàng).
   * **Kiểm thử giao diện**:
     + Đảm bảo giao diện dễ hiểu, thân thiện với nhân viên cửa hàng.
   * **Thu thập phản hồi**:
     + Yêu cầu chủ cửa hàng và nhân viên sử dụng thử hệ thống.
     + Điều chỉnh giao diện hoặc tính năng theo góp ý.

**Giai đoạn 5: Triển khai và hướng dẫn sử dụng**

1. **Mục tiêu**:
   * Đưa hệ thống vào sử dụng chính thức tại cửa hàng kính mắt.
   * Đảm bảo nhân viên cửa hàng sử dụng hệ thống một cách hiệu quả.
2. **Các hoạt động**:
   * **Cài đặt hệ thống**:
     + Cài đặt phần mềm trên các máy tính tại cửa hàng.
     + Cấu hình cơ sở dữ liệu SQL Server để lưu trữ thông tin.
   * **Hướng dẫn sử dụng**:
     + Tạo tài liệu hướng dẫn chi tiết (PDF hoặc video).
     + Tổ chức buổi hướng dẫn ngắn cho nhân viên cửa hàng.
   * **Hỗ trợ sau triển khai**:
     + Xử lý các lỗi phát sinh trong quá trình sử dụng.
     + Cập nhật phần mềm theo yêu cầu.

## 1.3. Các công nghệ sử dụng

Để xây dựng và triển khai hệ thống quản lý bán kính mắt, chúng tôi sử dụng các công nghệ và công cụ sau:

1. **Ngôn ngữ lập trình**: C# - Ngôn ngữ lập trình mạnh mẽ và linh hoạt, được sử dụng rộng rãi trong phát triển phần mềm Windows.
2. **Nền tảng phát triển**: .NET Framework - Khung phát triển ứng dụng của Microsoft hỗ trợ mạnh mẽ cho việc xây dựng các ứng dụng desktop.
3. **Công cụ phát triển**: Visual Studio - Môi trường phát triển tích hợp (IDE) hỗ trợ đầy đủ các tính năng cho việc lập trình, gỡ lỗi, và triển khai ứng dụng.
4. **Cơ sở dữ liệu**: SQL Server - Hệ quản trị cơ sở dữ liệu phổ biến, hỗ trợ quản lý dữ liệu lớn và tích hợp tốt với các ứng dụng Windows Forms.
5. **Thư viện và công cụ hỗ trợ**:
   * Windows Forms: Thư viện của .NET Framework dùng để xây dựng giao diện người dùng.
   * ADO.NET: Thư viện của .NET Framework hỗ trợ kết nối và thao tác với cơ sở dữ liệu.
   * Report Viewer và Report.rdlc: Công cụ tạo và xem báo cáo trực quan.

# Chương 2: Mô hình hệ thống

## 2.1. Các chức năng của hệ thống

Hệ thống quản lý bán kính mắt sẽ bao gồm các chức năng chính sau:

* **Quản lý hàng hóa**: Thêm, sửa, và xóa thông tin sản phẩm. Sản phẩm có thể bao gồm các chi tiết như công dụng, loại kính, màu sắc, diop (độ kính), ...
* **Quản lý nhân viên**: Thêm, sửa, và xóa thông tin nhân viên. Các thông tin bao gồm tên, giới tính, ngày sinh, địa chỉ, điện thoại, vị trí công việc, email và mật khẩu.
* **Quản lý khách hàng**: Thêm, sửa, và xóa thông tin khách hàng như tên, địa chỉ, và số điện thoại.
* **Quản lý nhà cung cấp**: Thêm, sửa, và xóa nhà cung cấp cùng các chi tiết liên hệ.
* **Quản lý doanh thu**: Xem doanh thu theo từng tháng hoặc trong khoảng thời gian tùy chọn, phân tích doanh thu thuần bằng cách so sánh doanh thu bán hàng với chi phí nhập hàng.
* **Quản lý hóa đơn bán hàng**: Thêm, xóa hóa đơn bán hàng và xem chi tiết hóa đơn, bao gồm thông tin sản phẩm và giảm giá cho từng mục.
* **Quản lý hóa đơn nhập hàng**: Tương tự như hóa đơn bán hàng, quản lý hóa đơn nhập bao gồm thêm, xóa, và xem chi tiết hóa đơn nhập từ nhà cung cấp.

## 2.2. Các giao diện chính có trong hệ thống

Dưới đây là danh sách các giao diện (forms) chính trong hệ thống:

1. **Form Account (Đăng nhập)**: Giao diện cho phép người dùng đăng nhập vào hệ thống.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Tên đối tượng | Kiểu | Ý nghĩa | Ghi chú |
| 1 | username | Textbox | Nhập mã nhân viên |  |
| 2 | password | Textbox | Nhập mật khẩu |  |
| 3 | Login | Button | Tiến hành đăng nhập |  |
| 4 | label3 | Label | Hiển thị “Đăng nhập” |  |
| 5 | label1 | Label | Hiển thị “Tài khoản” |  |
| 6 | label2 | Label | Hiển thị “Mật khẩu” |  |
| 7 | pictureBox1 | Picturebox | Hiển thị icon username |  |
| 8 | pictureBox2 | Picturebox | Hiển thị icon password |  |
| 9 | checkBox1 | Checkbox | Lưu tài khoản vừa đăng nhập |  |
| 10 | linkLabel1 | Linklabel | Chuyển sang form quên mật khẩu |  |

**Phương thức chính**:

* **LoadSavedCredentials()**: Tải thông tin đăng nhập đã lưu nếu người dùng đã chọn tùy chọn "Lưu mật khẩu" trước đó.
* **Login\_Click**: Xử lý sự kiện khi nhấn nút đăng nhập, xác thực người dùng và chuyển đến form Home nếu đăng nhập thành công.
* **SaveCredentials(string username, string password)**: Lưu thông tin đăng nhập khi chọn "Lưu mật khẩu".
* **ClearSavedCredentials()**: Xóa thông tin đăng nhập đã lưu khi "Lưu mật khẩu" bị bỏ chọn hoặc khi người dùng đăng xuất.
* **AuthenticateUser(string userInput, string password)**: Xác thực thông tin đăng nhập của người dùng bằng cách truy vấn cơ sở dữ liệu.
* **GetUserRole(string maNV)**: Lấy vai trò (công việc) của người dùng dựa trên mã nhân viên.

1. **Form home (Trang chủ)**: Giao diện chính điều hướng đến các chức năng khác.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Tên đối tượng | Kiểu | Ý nghĩa | Ghi chú |
| 1 | lbTenNV | Label | Hiển thị tên nhân viên | Nội dung cập nhật từ thuộc tính TenNV |
| 2 | lbCV | Label | Hiển thị công việc nhân viên | Nội dung cập nhật từ thuộc tính CongViec |
| 3 | QLNV | Button | Mở form quản lý nhân viên | Chỉ hiển thị nếu nhân viên là quản lý |
| 4 | btndoanhthu | Button | Mở form quản lý doanh thu | Chỉ hiển thị nếu nhân viên là quản lý |
| 5 | button3 | Button | Mở form thêm nhân viên | Chỉ hiển thị nếu nhân viên là quản lý |
| 6 | button4 | Button | Mở form tìm kiếm nhân viên | Chỉ hiển thị nếu nhân viên là quản lý |
| 7 | button1 | Button | Mở form thêm hàng hóa |  |
| 8 | QLDMHH | Button | Mở form quản lý danh mục hàng hóa |  |
| 9 | QLHDB | Button | Mở form quản lý hóa đơn bán |  |
| 10 | QLHDN | Button | Mở form quản lý hóa đơn nhập |  |
| 11 | button2 | Button | Tìm kiếm sản phẩm |  |
| 12 | button7 | Button | Tìm kiếm khách hàng |  |
| 13 | QLKH | Button | Mở form quản lý khách hàng |  |
| 14 | QLNCC | Button | Mở form quản lý nhà cung cấp |  |
| 15 | rspass | Button | Mở form reset password |  |
| 16 | richTextBoxThongBao | richTextBox | Hiển thị thông báo |  |
| 17 | exit | Button | Đăng xuất, quay lại form đăng nhập |  |

**Phương thức chính**:

**Home\_Load**: Xử lý sự kiện khi form Home được tải, cập nhật giao diện và hiển thị các thông báo liên quan đến người dùng.

**UpdateUI**: Cập nhật thông tin giao diện người dùng (như tên nhân viên và công việc), hiển thị các nút chức năng tùy theo vai trò của người dùng (ví dụ: hiện thị nút quản lý nếu người dùng có vai trò là quản lý).

**LoadThongBao**: Tải và hiển thị các thông báo từ cơ sở dữ liệu. Nếu người dùng là quản lý, tất cả các thông báo sẽ được hiển thị; nếu không, chỉ các thông báo liên quan đến mã nhân viên mới được hiển thị.

**QLNV\_Click**: Mở form quản lý nhân viên, dùng khi người dùng có vai trò quản lý và chọn nút QLNV.

**QLNCC\_Click**: Mở form quản lý nhà cung cấp.

**QLKH\_Click**: Mở form quản lý khách hàng.

**QLHDN\_Click**: Mở form quản lý hóa đơn nhập.

**QLDMHH\_Click**: Mở form quản lý danh mục hàng hóa.

**QLHDB\_Click**: Mở form quản lý hóa đơn bán.

**exit\_Click**: Xử lý sự kiện đăng xuất, mở lại form đăng nhập và đóng form Home.

**LayEmailTuMaNV**: Lấy email của người dùng từ cơ sở dữ liệu dựa trên mã nhân viên, dùng cho mục đích đổi mật khẩu.

**rspass\_Click**: Xử lý sự kiện khi người dùng chọn đổi mật khẩu, sử dụng email lấy từ phương thức LayEmailTuMaNV và mở form đổi mật khẩu.

**LoadSavedCredentials** (trong ngữ cảnh form đăng nhập): Tải thông tin đăng nhập đã lưu nếu người dùng chọn "Lưu mật khẩu" trước đó.

**AuthenticateUser** (trong ngữ cảnh form đăng nhập): Xác thực thông tin đăng nhập của người dùng qua truy vấn cơ sở dữ liệu.

**GetUserRole** (trong ngữ cảnh form đăng nhập): Lấy vai trò (công việc) của người dùng dựa trên mã nhân viên.

1. **Form quản lý hàng hóa**: Quản lý thông tin hàng hóa.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Tên đối tượng | Kiểu | Ý nghĩa | Ghi chú |
| 1 | dataGridView1 | dataGridView | Hiển thị thông tin các sản phẩm |  |
| 2 | Them | Button | Mở form MainHangHoa để thêm sản phẩm và các chi tiết sản phẩm liên quan |  |
| 3 | Sua | Button | Mở form sửa hàng hóa |  |
| 4 | Xoa | Button | Xóa hàng hóa | Yêu cầu xác nhận trước khi xóa, nếu sản phẩm đã tồn tại hóa đơn liên quan thì không được xóa |
| 5 | button1 | Button | Mở form để xem báo cáo |  |
| 6 | btnXuatExcel | Button | Xuất danh sách sản phẩm ra file Excel |  |
| 7 | Exit | Button | Trở về form Home |  |

**Phương thức chính**:

* **loadData()**: Phương thức tải dữ liệu từ cơ sở dữ liệu và hiển thị vào DataGridView. Phương thức này sử dụng truy vấn SQL phức tạp với nhiều phép JOIN để lấy tên từ các bảng liên quan như loại kính, chất liệu, màu sắc, và nước sản xuất.
* **Them\_Click**: Sự kiện khi người dùng nhấn nút thêm hàng hóa mới. Phương thức này sẽ mở form MainHangHoa và cập nhật DataGridView khi có hàng hóa mới được thêm.
* **Sua\_Click**: Phương thức xử lý sự kiện khi người dùng nhấn nút sửa, lấy dữ liệu hàng hóa từ hàng được chọn và truyền vào form SuaHangHoa để sửa.
* **Xoa\_Click**: Xử lý việc xóa hàng hóa được chọn. Phương thức này yêu cầu người dùng xác nhận trước khi xóa hàng hóa từ cơ sở dữ liệu. Nếu hàng hóa tồn tại trong hóa đơn, sẽ hiển thị thông báo lỗi không cho phép xóa.
* **btnXuatExcel\_Click**: Xuất dữ liệu từ DataGridView ra file Excel (.xlsx) bằng thư viện ClosedXML. Phương thức này sẽ lưu file tại đường dẫn người dùng chọn và thông báo khi thành công.
* **dataGridView1\_CellFormatting**: Định dạng lại dữ liệu hiển thị trong các cột DonGiaNhap và DonGiaBan. Số liệu sẽ hiển thị tối đa hai chữ số thập phân nếu có phần lẻ, hoặc chỉ phần nguyên nếu không có phần lẻ.
* **Exit\_Click**: Đóng form QuanLyDanhMucHangHoa và quay về form Home, truyền lại thông tin của nhân viên để hiển thị đúng dữ liệu trên form Home.
* **button1\_Click**: Mở báo cáo HangHoaReport với thông tin tên nhân viên.

1. **Form quản lý nhân viên**: Quản lý thông tin nhân viên.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Tên đối tượng | Kiểu | Ý nghĩa | Ghi chú |
| 1 | dataGridView1 | dataGridView | Hiển thị thông tin các nhân viên |  |
| 2 | ThemNV | Button | Mở form thêm nhân viên | Quyền quản lý mới được thao tác |
| 3 | SuaNV | Button | Mở form sửa nhân viên | Quyền quản lý mới được thao tác |
| 4 | XoaNV | Button | Xóa nhân viên | Quyền quản lý mới được thao tác, xác nhận trước khi xóa, nếu tồn tại háo đơn liên quan thì không được xóa, không được tự xóa bản thân. |
| 5 | btnbaocao | Button | Mở form để xem báo cáo | Quyền quản lý mới được thao tác |
| 6 | exitNV | Button | Trở lại form Home |  |

**Phương thức chính**:

* **LoadData()**: Phương thức tải dữ liệu từ bảng NhanVien trong cơ sở dữ liệu và hiển thị vào DataGridView.
* **ThemNV\_Click**: Sự kiện khi người dùng nhấn nút "Thêm nhân viên", mở form ThemNV để thêm nhân viên mới.
* **SuaNV\_Click**: Sự kiện sửa nhân viên khi một hàng trong DataGridView được chọn. Phương thức này lấy thông tin nhân viên từ hàng được chọn và mở form SuaNV với dữ liệu sẵn có để người dùng sửa.
* **XoaNV\_Click**: Sự kiện xóa nhân viên được chọn, với hộp thoại xác nhận trước khi thực hiện. Nếu xóa thành công, DataGridView được cập nhật để phản ánh thay đổi.
* **exitNV\_Click**: Đóng form QuanLyNhanVien và quay lại form Home, đồng thời truyền lại thông tin của nhân viên hiện tại (gồm tên, công việc, và mã) để hiển thị đúng dữ liệu trên form Home.
* **btnbaocao\_Click**: Mở báo cáo NhanVienReport với thông tin tên nhân viên để xuất báo cáo danh sách nhân viên.

1. **Form quản lý khách hàng**: Quản lý thông tin khách hàng.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Tên đối tượng | Kiểu | Ý nghĩa | Ghi chú |
| 1 | dataGridView1 | dataGridView | Hiển thị thông tin khách hàng |  |
| 2 | themKH | Button | Mở form thêm khách hàng |  |
| 3 | suaKH | Button | Mở form sửa khách hàng |  |
| 4 | xoaKH | Button | Xóa khách hàng | Xác nhận trước khi xóa, nếu tồn tại hóa đơn liên quan thì không được xóa |
| 5 | btnbaocao | Button | Mở form để xem báo cáo |  |
| 6 | exit | Button | Trở lại form Home |  |

**Phương thức chính**:

* **LoadData()**: Phương thức tải dữ liệu từ bảng KhachHang trong cơ sở dữ liệu và hiển thị trên DataGridView. Cột MaKhach, TenKhach, DiaChi, và DienThoai được chọn từ bảng.
* **QuanLykhachhnag\_Activated**: Sự kiện cập nhật dữ liệu DataGridView mỗi khi form QuanLyKhachHang được kích hoạt lại.
* **ThemKH\_Click**: Sự kiện cho nút "Thêm khách hàng", mở form ThemKH để thêm một khách hàng mới.
* **SuaKH\_Click**: Sự kiện cho nút "Sửa khách hàng", cho phép người dùng chỉnh sửa thông tin khách hàng được chọn. Thông tin về mã khách hàng (MaKhach), tên khách hàng (TenKhach), địa chỉ (DiaChi), và điện thoại (DienThoai) từ hàng được chọn được chuyển đến form SuaKH để thực hiện chỉnh sửa.
* **XoaKH\_Click**: Sự kiện cho nút "Xóa khách hàng" với hộp thoại xác nhận trước khi xóa khách hàng. Sau khi xóa thành công, DataGridView được cập nhật lại.
* **exit\_Click**: Đóng form QuanLyKhachHang và quay lại form Home, truyền lại thông tin nhân viên (tên, công việc, mã) để hiển thị đúng trên form Home.
* **btnbaocao\_Click**: Mở báo cáo KhachHangReport để xuất thông tin khách hàng, với thông tin tên nhân viên được truyền vào để cá nhân hóa báo cáo.

1. **Form quản lý nhà cung cấp**: Quản lý thông tin nhà cung cấp.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Tên đối tượng | Kiểu | Ý nghĩa | Ghi chú |
| 1 | dataGridView1 | dataGridView | Hiển thị thông tin các nhà cung cấp |  |
| 2 | btnThem | Button | Mở form thêm nhà cung cấp |  |
| 3 | btnSua | Button | Mở form sửa nhà cung cấp |  |
| 4 | btnXoa | Button | Xóa nhà cung cấp | Yêu cầu xác nhận trước khi xóa, nếu tồn tại hóa đơn liên quan thì không được xóa |
| 5 | button2 | Button | Mở form để xem báo cáo |  |
| 6 | button1 | Button | Trở lại form Home |  |

**Phương thức chính**:

* **LoadData()**: Phương thức này kết nối với cơ sở dữ liệu và tải tất cả dữ liệu từ bảng NhaCungCap, sau đó hiển thị dữ liệu lên DataGridView.
* **QuanLyNhaCungCap Constructor**:
* Constructor mặc định khởi tạo form và gọi LoadData() để tải dữ liệu ngay khi form mở.
* Constructor có tham số (tên nhân viên, công việc, mã nhân viên) dùng để truyền thông tin nhân viên khi khởi tạo form.
* **btnThem\_Click**: Mở form NhaCungCap ở chế độ thêm (Mode = "Them"). Nếu người dùng thêm thành công (kết quả là DialogResult.OK), dữ liệu sẽ được tải lại để cập nhật DataGridView.
* **btnSua\_Click**: Mở form NhaCungCap ở chế độ sửa (Mode = "Sua"). Khi một hàng được chọn, mã nhà cung cấp (MaNCC) sẽ được truyền sang form NhaCungCap để sửa đổi thông tin. Sau khi sửa thành công, DataGridView sẽ được cập nhật.
* **btnXoa\_Click**: Thực hiện xóa nhà cung cấp khi có hàng được chọn trong DataGridView. Trước khi xóa, hiển thị một hộp thoại xác nhận. Nếu người dùng đồng ý, phương thức XoaNhaCungCap sẽ xóa nhà cung cấp và cập nhật lại DataGridView.
* **button1\_Click**: Đóng form QuanLyNhaCungCap và mở lại form Home, truyền thông tin nhân viên để form Home có thể hiển thị đúng.
* **button2\_Click**: Mở form báo cáo NhaCungCapReport để xuất thông tin nhà cung cấp, với thông tin tên nhân viên được truyền vào báo cáo.

1. **Form doanh thu**: Hiển thị doanh thu theo khoảng thời gian và theo tháng.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Tên đối tượng | Kiểu | Ý nghĩa | Ghi chú |
| 1 | dataGridView1 | dataGridView | Xem doanh thu | Mặc định là không có gì |
| 2 | dtpngaybatdau | Datetimepicker | Chọn thời gian bắt đầu |  |
| 3 | dtpngayketthuc | Datetimepicker | Chọn thời gian kết thúc |  |
| 4 | lblTongDoanhThuBan | Label | Hiển thị doanh thu bán | Trong khoảng thời gian được chọn |
| 5 | lblTongDoanhThuNhap | Label | Hiển thị doanh thu nhập | Trong khoảng thời gian được chọn |
| 6 | lblTongDoanhThuThuan | Label | Hiển thị doanh thu thuần | Trong khoảng thời gian được chọn |
| 7 | btnxemdoanhthu | Button | Xem doanh thu dựa trên khoảng thời gian đã chọn |  |
| 8 | btnxemdoanhthutheothang | Button | Mở form xem doanh thu theo tháng |  |
| 9 | button1 | Button | Mở form xem báo cáo doanh thu |  |
| 10 | btnexit | Button | Trở lại form Home |  |

**Phương thức chính**:

* **Khởi tạo form DoanhThu**:
  + Constructor mặc định không có tham số.
  + Constructor có tham số nhận thông tin nhân viên (TenNV, CongViec, MaNV) để truyền vào form và sử dụng trong báo cáo hoặc điều hướng.
* **btnxemdoanhthu\_Click**:
  + Sự kiện này thực hiện việc hiển thị doanh thu theo khoảng ngày đã chọn (tuNgay và denNgay).
  + Kết nối cơ sở dữ liệu, lấy dữ liệu từ bảng DoanhThu với điều kiện ngày trong khoảng được chọn.
  + Dữ liệu thu được sẽ được hiển thị trong dataGridView1, với các cột:
    - NgayThang: Ngày tháng doanh thu.
    - DoanhThuBan: Doanh thu bán hàng.
    - DoanhThuNhap: Chi phí nhập hàng.
    - DoanhThuThuần: Lợi nhuận thuần.
  + **Tính tổng doanh thu**:
    - totalDoanhThuBan: Tổng doanh thu bán.
    - totalDoanhThuNhap: Tổng chi phí nhập.
    - totalDoanhThuThuan: Tổng lợi nhuận.
  + Các giá trị tổng sẽ được hiển thị trong các Label tương ứng (lblTongDoanhThuBan, lblTongDoanhThuNhap, lblTongDoanhThuThuan).
  + Mở form ChiTietDoanhThu để hiển thị chi tiết doanh thu trong khoảng thời gian đã chọn.
* **btnexit\_Click**: Đóng form DoanhThu và quay lại form Home, truyền các thông tin nhân viên (TenNV, CongViec, MaNV).
* **btnxemdoanhthutheothang\_Click**: Mở form DoanhThuTheoThang để xem doanh thu theo tháng.
* **button1\_Click**: Mở form báo cáo DoanhThuReport để xuất báo cáo doanh thu, truyền tên nhân viên (TenNV) vào báo cáo.
* **DoanhThu\_Load** và các sự kiện không hoạt động khác (như panel1\_Paint, lblTongDoanhThuBan\_Click) là các sự kiện không thực hiện chức năng cụ thể trong đoạn mã hiện tại, có thể được dùng để xử lý thêm trong tương lai.

1. **Form quản lý hóa đơn bán**: Quản lý các hóa đơn bán, bao gồm thêm mới, xóa và xem chi tiết hóa đơn.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Tên đối tượng | Kiểu | Ý nghĩa | Ghi chú |
| 1 | dataGridView1 | dataGridView | Hiển thị thông tin các hóa đơn bán |  |
| 2 | btnThemHD | Button | Mở form thêm hóa đơn bán |  |
| 3 | btnXoaHD | Button | Xóa hóa đơn bán | Không yêu cầu xác nhận trước khi xóa |
| 4 | btnXemChiTiet | Button | Xem chi tiết hóa đơn bán |  |
| 5 | btnIn | Button | In hóa đơn |  |
| 6 | btnbaocao | Button | Mở form xem báo cáo hóa đơn bán |  |
| 7 | btnexit | Button | Trở lại form Home |  |

**Phương thức chính**:

* **Constructor QuanLyHoaDonBan**:
  + Khởi tạo form và gán thông tin nhân viên (TenNV, CongViec, MaNV) nếu có.
  + Gọi hàm LoadData() để tải dữ liệu hóa đơn từ cơ sở dữ liệu và hiển thị trong dataGridView1.
* **Phương thức LoadData()**:
  + Mở kết nối đến cơ sở dữ liệu và lấy dữ liệu từ bảng HoaDonBan cùng ChiTietHoaDonBan, tính tổng số lượng sản phẩm và tổng tiền của từng hóa đơn.
  + Dữ liệu được nạp vào dataGridView1 để hiển thị.
* **Sự kiện btnXemChiTiet\_Click**:
  + Kiểm tra xem có hàng nào được chọn trong dataGridView1 hay không.
  + Lấy SoHDB của hóa đơn được chọn, mở form ChiTietHoaDonBan để xem chi tiết hóa đơn.
* **Sự kiện btnThemHD\_Click**:
  + Mở form HoaDonBan cho phép người dùng thêm hóa đơn bán mới vào cơ sở dữ liệu.
* **Sự kiện btnXoaHD\_Click**:
  + Kiểm tra xem có hóa đơn nào được chọn để xóa không.
  + Sau khi xác nhận, xóa chi tiết và hóa đơn được chọn khỏi cơ sở dữ liệu trong một giao dịch (transaction). Nếu có lỗi, giao dịch sẽ được hủy bỏ.
* **Sự kiện btnIn\_Click**:
  + Lấy chi tiết của hóa đơn đã chọn và tạo báo cáo hóa đơn để in.
  + Sử dụng đối tượng PrintDocument và PrintDialog để hiển thị thông tin chi tiết hóa đơn, bao gồm tên khách hàng, ngày bán, tên nhân viên, sản phẩm, số lượng, và thành tiền.
* **Sự kiện btnbaocao\_Click**:
  + Mở form HoaDonBanReport để tạo báo cáo tổng quan các hóa đơn, có thể dùng cho báo cáo chi tiết hơn.
* **Sự kiện dataGridView1\_CellFormatting**:
  + Định dạng cột ThanhTien và TongTien để hiển thị số tiền với hai chữ số thập phân nếu cần thiết.
* **Sự kiện exit\_Click**:
  + Đóng form hiện tại và mở lại form Home, truyền thông tin nhân viên để tiếp tục sử dụng trong form đó.

1. **Form quản lý hóa đơn nhập**: Quản lý các hóa đơn nhập, bao gồm thêm mới, xóa và xem chi tiết hóa đơn.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Tên đối tượng | Kiểu | Ý nghĩa | Ghi chú |
| 1 | dataGridView1 | dataGridView | Hiển thị thông tin các hóa đơn nhập |  |
| 2 | btnthemhdn | Button | Mở form thêm hóa đơn nhập |  |
| 3 | btnxoahdn | Button | Xóa hóa đơn nhập | Không yêu cầu xác nhận trước khi xóa |
| 4 | btnchitiethdn | Button | Xem chi tiết hóa đơn nhập |  |
| 5 | btnbaocaohdn | Button | Xem báo cáo hóa đơn nhập |  |
| 6 | exit | Button | Trở lại form Home |  |

**Phương thức chính**:

* **LoadData()**: Phương thức này kết nối đến cơ sở dữ liệu SQL Server, lấy dữ liệu về các hóa đơn nhập (bao gồm các thông tin như số hóa đơn, ngày nhập, mã nhà cung cấp, tổng số lượng, tổng tiền) và hiển thị lên dataGridView1.
* **Sự kiện kích hoạt form (QuanLyHoaDonNhap\_Activated)**: Mỗi khi form được kích hoạt lại, phương thức LoadData() sẽ được gọi để cập nhật dữ liệu hiển thị mới nhất từ cơ sở dữ liệu.
* **Thêm chi tiết hóa đơn nhập**: Sự kiện button1\_Click sẽ mở form ThemChiTietHoaDonNhap2, cho phép thêm chi tiết cho một hóa đơn nhập mới.
* **Xóa hóa đơn nhập**: Sự kiện button3\_Click cho phép người dùng xóa hóa đơn nhập đã chọn sau khi xác nhận. Chương trình sử dụng transaction để đảm bảo xóa dữ liệu liên quan một cách an toàn. Nếu việc xóa thành công, dữ liệu được tải lại; nếu thất bại, hệ thống sẽ thông báo lỗi.
* **Xem chi tiết hóa đơn nhập**: Sự kiện button4\_Click cho phép người dùng xem chi tiết của một hóa đơn nhập đã chọn bằng cách mở form ChiTietHoaDonNhap.
* **Định dạng dữ liệu hiển thị**: Phương thức dataGridView1\_CellFormatting sẽ định dạng lại cột TongTien và DonGia để hiển thị giá trị dạng số tròn nếu không có phần thập phân hoặc hiển thị hai chữ số thập phân nếu có.
* **Quay lại trang chủ**: Sự kiện exit\_Click sẽ đóng form hiện tại và quay lại form Home với thông tin người dùng đã đăng nhập.
* **Xem báo cáo**: Sự kiện btnbaocao\_Click mở form HoaDonNhapReport để hiển thị báo cáo về hóa đơn nhập.

## 2.3. Mô hình lớp

Mô hình lớp của hệ thống bao gồm các lớp chính như sau:

### 2.3.1. Lớp Account (Kế thừa từ lớp Form):

* **Thuộc tính**:
  + **username**: TextBox nhận tên đăng nhập của người dùng.
  + **password**: TextBox nhận mật khẩu của người dùng.
  + **checkBox1**: CheckBox để lưu thông tin đăng nhập (tùy chọn "Remember Me").
* **Phương thức**:
  + **Account()**: Hàm khởi tạo, gọi InitializeComponent() để khởi tạo các thành phần trên form và gọi LoadSavedCredentials() để tải thông tin đăng nhập đã lưu.
  + **LoadSavedCredentials()**: Kiểm tra xem có thông tin đăng nhập nào đã được lưu trước đó hay không, nếu có sẽ tự động điền vào TextBox.
  + **Login\_Click(object sender, EventArgs e)**: Xử lý sự kiện khi nhấn nút đăng nhập, xác thực người dùng qua AuthenticateUser, lưu hoặc xóa thông tin đăng nhập nếu cần và chuyển đến giao diện chính (Home).
  + **SaveCredentials(string username, string password)**: Lưu tên đăng nhập và mật khẩu khi người dùng chọn tùy chọn "Remember Me".
  + **ClearSavedCredentials()**: Xóa thông tin đăng nhập đã lưu khi người dùng bỏ chọn "Remember Me" hoặc đăng xuất.
  + **AuthenticateUser(string userInput, string password)**: Xác thực thông tin đăng nhập của người dùng bằng cách truy vấn cơ sở dữ liệu.
  + **GetUserRole(string maNV)**: Lấy vai trò (công việc) của nhân viên từ cơ sở dữ liệu dựa trên mã nhân viên.

### 2.3.2. Lớp Home (Kế thừa từ lớp Form)

**Thuộc tính:**

* TenNV (string): Lưu trữ tên của nhân viên đăng nhập.
* CongViec (string): Lưu trữ chức vụ của nhân viên (ví dụ: Quản lý).
* MaNV (int): Mã số định danh của nhân viên.

**Phương thức:**

* **Home()**: Hàm khởi tạo, thiết lập các sự kiện và ẩn các nút giao diện ban đầu.
* **QLNV\_Click**: Mở form quản lý nhân viên. Chỉ dành cho vai trò quản lý.
* **QLNCC\_Click**: Mở form quản lý nhà cung cấp.
* **QLKH\_Click**: Mở form quản lý khách hàng.
* **QLHDN\_Click**: Mở form quản lý hóa đơn nhập.
* **QLDMHH\_Click**: Mở form quản lý danh mục hàng hóa.
* **QLHDB\_Click**: Mở form quản lý hóa đơn bán.
* **exit\_Click**: Đăng xuất, mở form đăng nhập Account và đóng form hiện tại.
* **Home\_Load**: Xử lý khi form Home tải lên. Thiết lập giao diện và tải thông báo.
* **UpdateUI**: Cập nhật giao diện cho form dựa trên chức vụ của nhân viên. Nếu là quản lý, các nút liên quan đến quản lý sẽ hiển thị.
* **button1\_Click**: Mở form thêm hàng hóa.
* **button9\_Click**: Mở lại form quản lý danh mục hàng hóa.
* **button12\_Click**: Mở form quản lý hóa đơn bán.
* **button8\_Click**: Mở form quản lý hóa đơn nhập.
* **btndoanhthu\_Click**: Mở form quản lý doanh thu, chỉ dành cho quản lý.
* **LoadThongBao**: Tải thông báo từ cơ sở dữ liệu và hiển thị trong richTextBoxThongBao. Nếu là quản lý, sẽ tải tất cả thông báo, nếu không chỉ tải thông báo dành cho nhân viên đó.
* **button4\_Click**: Mở form tìm kiếm nhân viên.
* **button3\_Click**: Mở form thêm nhân viên.
* **button7\_Click**: Mở form tìm kiếm khách hàng.
* **button2\_Click**: Mở form tìm kiếm sản phẩm.
* **button6\_Click**: Mở form tìm kiếm hóa đơn.
* **LayEmailTuMaNV**: Truy vấn lấy email của nhân viên từ mã nhân viên (MaNV) để sử dụng trong chức năng đổi mật khẩu.
* **rspass\_Click**: Mở form đổi mật khẩu với email của nhân viên.

### 2.3.3. Lớp QuanLyDanhMucHangHoa (Kế thừa từ lớp Form)

**Thuộc tính**

* **TenNV (string)**: Tên của nhân viên hiện tại, có thể được truyền vào từ form gọi để hiển thị thông tin hoặc sử dụng trong báo cáo.
* **CongViec (string)**: Chức vụ của nhân viên (quản lý hoặc nhân viên), xác định quyền truy cập vào các chức năng quản lý.
* **MaNV (int)**: Mã định danh nhân viên hiện tại, sử dụng để phân biệt các nhân viên khi thực hiện truy vấn hoặc thao tác cụ thể.

**Phương thức chính:**

* **QuanLyDanhMucHangHoa()**: Hàm khởi tạo mặc định, thiết lập giao diện ban đầu và đăng ký sự kiện cho các nút trên form.
* **QuanLyDanhMucHangHoa(string tenNV, string congViec, int maNV)**: Hàm khởi tạo có tham số, nhận thông tin nhân viên từ form gọi và gán vào các thuộc tính để hiển thị hoặc sử dụng khi cần thiết.

**Các chức năng chính:**

* **loadData()**:
  + Mở kết nối đến cơ sở dữ liệu, thực hiện truy vấn để lấy dữ liệu hàng hóa từ bảng DanhMucHangHoa.
  + Sử dụng các bảng liên kết để lấy thông tin chi tiết như LoaiKinh, GongMat, HinhDangMat, v.v. để hiển thị thông tin sản phẩm đầy đủ.
  + Gán dữ liệu vào dataGridView1 và đặt tên hiển thị cho các cột, giúp người dùng dễ dàng quản lý và theo dõi.
* **Them\_Click**:
  + Mở form MainHangHoa để thêm hàng hóa mới.
  + Sử dụng sự kiện HangHoaAdded để tự động cập nhật dữ liệu trong dataGridView1 khi có hàng hóa mới được thêm.
* **Sua\_Click**:
  + Kiểm tra xem người dùng đã chọn hàng hóa trong dataGridView1 hay chưa.
  + Nếu có, lấy dữ liệu từ hàng được chọn và truyền vào form SuaHangHoa để chỉnh sửa thông tin.
  + Sau khi chỉnh sửa, dữ liệu sẽ được cập nhật lại trong dataGridView1.
* **Exit\_Click**:
  + Đóng form QuanLyDanhMucHangHoa và quay lại form Home, giữ nguyên thông tin nhân viên hiện tại.
* **Xoa\_Click**:
  + Kiểm tra xem người dùng đã chọn hàng hóa chưa.
  + Nếu có, hiển thị hộp thoại xác nhận xóa.
  + Sau khi người dùng xác nhận, thực hiện lệnh xóa trong cơ sở dữ liệu và cập nhật dataGridView1.
* **btnXuatExcel\_Click**:
  + Hiển thị hộp thoại lưu file và cho phép người dùng xuất dữ liệu hàng hóa ra file Excel.
  + Sử dụng thư viện ClosedXML để lưu file, tiện cho việc chia sẻ hoặc lưu trữ dữ liệu.
* **dataGridView1\_CellFormatting**:
  + Tùy chỉnh định dạng của các cột giá trị (DonGiaNhap, DonGiaBan) trong dataGridView1.
  + Hiển thị dạng số nguyên nếu không có phần lẻ, nếu có sẽ hiển thị tối đa hai chữ số thập phân.
* **button1\_Click**:
  + Mở form báo cáo HangHoaReport sử dụng tên nhân viên hiện tại, giúp báo cáo dễ dàng truy vết theo nhân viên.

### 2.3.4. Lớp QuanLyHoaDonBan (Kế thừa từ lớp Form)

**Thuộc tính:**

* **TenNV (string):** Lưu trữ tên của nhân viên hiện tại.
* **CongViec (string):** Lưu trữ chức vụ của nhân viên.
* **MaNV (int):** Mã số định danh của nhân viên.

**Phương thức:**

* **QuanLyHoaDonBan() (Constructor):** Hàm khởi tạo mặc định, thực hiện khởi tạo giao diện và tải dữ liệu hóa đơn.
* **QuanLyHoaDonBan(string tenNV, string congViec, int maNV) (Constructor):** Hàm khởi tạo với các tham số chi tiết về nhân viên. Được sử dụng để khởi tạo giao diện và tải dữ liệu hóa đơn kèm theo thông tin nhân viên đăng nhập.
* **capnhat (Event Handler):** Phương thức xử lý sự kiện khi Form được kích hoạt, dùng để cập nhật dữ liệu hiển thị.
* **LoadData():** Phương thức truy vấn cơ sở dữ liệu và tải dữ liệu hóa đơn bán lên DataGridView. Hiển thị thông tin về mã hóa đơn, ngày bán, khách hàng, tổng số lượng sản phẩm, và thành tiền.
* **exit\_Click (Event Handler):** Đăng xuất khỏi Form hiện tại và mở lại Form Home.
* **btnXemChiTiet\_Click (Event Handler):** Mở Form chi tiết hóa đơn bán dựa trên hóa đơn đã chọn.
* **dataGridView1\_CellFormatting (Event Handler):** Định dạng dữ liệu hiển thị trong DataGridView, đặc biệt cho cột TongTien và ThanhTien với định dạng số thập phân.
* **btnThemHD\_Click (Event Handler):** Mở Form để thêm hóa đơn bán mới.
* **btnXoaHD\_Click (Event Handler):** Xóa hóa đơn bán được chọn sau khi có xác nhận từ người dùng.
* **btnIn\_Click (Event Handler):** Hiển thị hộp thoại in cho hóa đơn bán đã chọn. Truy vấn và hiển thị chi tiết hóa đơn, bao gồm các thông tin về khách hàng, nhân viên, sản phẩm, số lượng và tổng tiền hóa đơn.
* **btnbaocao\_Click (Event Handler):** Mở Form để tạo báo cáo hóa đơn bán.

### 2.3.5. Lớp QuanLyHoaDonNhap (Kế thừa từ lớp Form)

**Thuộc tính:**

* TenNV: Biến chuỗi lưu tên của nhân viên.
* CongViec: Biến chuỗi lưu công việc của nhân viên.
* MaNV: Biến số nguyên lưu mã nhân viên.

**Phương thức:**

* **QuanLyHoaDonNhap()** *(Constructor)*:
  + Khởi tạo lớp và gọi phương thức LoadData để tải dữ liệu hóa đơn nhập.
  + Gán sự kiện Activated cho lớp, để mỗi khi cửa sổ được kích hoạt, dữ liệu sẽ được tải lại.
* **QuanLyHoaDonNhap(string tenNV, string congViec, int maNV)** *(Constructor có tham số)*:
  + Tương tự như constructor trên, nhưng có thêm tham số đầu vào để khởi tạo các thông tin của nhân viên (TenNV, CongViec, MaNV).
* **LoadData()**:
  + Chức năng: Kết nối tới cơ sở dữ liệu, truy vấn các hóa đơn nhập và hiển thị lên dataGridView1.
  + Truy vấn lấy các thông tin gồm số hóa đơn, ngày tạo, mã nhân viên, mã nhà cung cấp, loại mặt hàng, tổng số lượng, và tổng tiền.
* **button1\_Click(object sender, EventArgs e)**:
  + Chức năng: Mở form ThemChiTietHoaDonNhap2 để thêm chi tiết hóa đơn nhập mới.
  + Truyền TenNV và MaNV để form mới có thể sử dụng thông tin nhân viên.
* **button3\_Click(object sender, EventArgs e)**:
  + Chức năng: Xóa hóa đơn nhập đã chọn trong dataGridView1.
  + Kiểm tra người dùng đã chọn một dòng dữ liệu chưa, hiển thị thông báo xác nhận xóa, rồi thực hiện xóa dữ liệu trong bảng HoaDonNhap và ChiTietHoaDonNhap trong một giao dịch.
* **button4\_Click(object sender, EventArgs e)**:
  + Chức năng: Mở form ChiTietHoaDonNhap để xem chi tiết của hóa đơn nhập đã chọn.
  + Truyền mã số hóa đơn (SoHDN) của dòng dữ liệu đang được chọn vào form chi tiết hóa đơn.
* **dataGridView1\_CellFormatting(object sender, DataGridViewCellFormattingEventArgs e)**:
  + Chức năng: Định dạng các cột trong dataGridView1, đặc biệt là cột TongTien và DonGia, để hiển thị số thập phân phù hợp.
* **exit\_Click(object sender, EventArgs e)**:
  + Chức năng: Đóng form hiện tại và mở form Home, truyền lại các thông tin TenNV, CongViec, và MaNV.
* **QuanLyHoaDonNhap\_Activated(object sender, EventArgs e)**:
  + Chức năng: Tự động gọi LoadData để cập nhật lại dữ liệu khi cửa sổ form được kích hoạt.
* **btnbaocao\_Click(object sender, EventArgs e)**:
  + Chức năng: Mở form HoaDonNhapReport để tạo báo cáo cho hóa đơn nhập và truyền thông tin TenNV để hiển thị trên báo cáo.

### 2.3.6. Lớp QuanLyKhachHang (Kế thừa từ lớp Form)

**Thuộc tính:**

* TenNV: Chuỗi lưu tên của nhân viên.
* CongViec: Chuỗi lưu công việc của nhân viên.
* MaNV: Số nguyên lưu mã nhân viên.

**Phương thức:**

* **QuanLyKhachHang()** *(Constructor)*:
  + Khởi tạo form quản lý khách hàng, gọi phương thức LoadData để tải dữ liệu khách hàng từ cơ sở dữ liệu.
  + Gán sự kiện Activated để mỗi khi form được kích hoạt, dữ liệu sẽ tự động được tải lại.
* **QuanLyKhachHang(string tenNV, string congViec, int maNV)** *(Constructor có tham số)*:
  + Tương tự constructor trên nhưng nhận thêm tham số để khởi tạo các thông tin của nhân viên (TenNV, CongViec, MaNV).
* **LoadData()**:
  + Kết nối đến cơ sở dữ liệu, thực hiện truy vấn thông tin khách hàng (MaKhach, TenKhach, DiaChi, DienThoai) từ bảng KhachHang và hiển thị lên dataGridView1.
* **themKH\_Click(object sender, EventArgs e)**:
  + Chức năng: Mở form ThemKH để thêm khách hàng mới.
* **suaKH\_Click(object sender, EventArgs e)**:
  + Chức năng: Sửa thông tin khách hàng được chọn từ dataGridView1.
  + Kiểm tra xem có dòng nào được chọn hay chưa. Nếu có, lấy thông tin khách hàng được chọn và mở form SuaKH, truyền dữ liệu của khách hàng để chỉnh sửa.
* **exit\_Click(object sender, EventArgs e)**:
  + Chức năng: Đóng form hiện tại và mở form Home, truyền thông tin TenNV, CongViec, và MaNV.
* **xoaKH\_Click(object sender, EventArgs e)**:
  + Chức năng: Xóa khách hàng được chọn từ dataGridView1.
  + Kiểm tra xem có khách hàng nào được chọn không. Nếu có, hiển thị thông báo xác nhận xóa, sau đó thực hiện xóa khách hàng trong cơ sở dữ liệu.
* **QuanLykhachhnag\_Activated(object sender, EventArgs e)**:
  + Chức năng: Gọi LoadData để cập nhật lại dữ liệu khi form QuanLyKhachHang được kích hoạt.
* **btnbaocao\_Click(object sender, EventArgs e)**:
  + Chức năng: Mở form KhachHangReport để tạo báo cáo khách hàng và truyền thông tin TenNV để hiển thị trên báo cáo.

### 2.3.7. Lớp QuanLyNhaCungCap (Kế thừa từ lớp Form)

**Thuộc tính:**

* TenNV: Chuỗi lưu tên của nhân viên.
* CongViec: Chuỗi lưu công việc của nhân viên.
* MaNV: Số nguyên lưu mã nhân viên.

**Phương thức:**

* **QuanLyNhaCungCap()** *(Constructor)*:
  + Khởi tạo form quản lý nhà cung cấp, gọi phương thức LoadData để tải dữ liệu nhà cung cấp từ cơ sở dữ liệu.
* **QuanLyNhaCungCap(string tenNV, string congViec, int maNV)** *(Constructor có tham số)*:
  + Tương tự constructor trên nhưng nhận thêm tham số để khởi tạo các thông tin của nhân viên (TenNV, CongViec, MaNV).
* **LoadData()**:
  + Kết nối đến cơ sở dữ liệu, thực hiện truy vấn thông tin nhà cung cấp từ bảng NhaCungCap và hiển thị lên dataGridView1.
* **ThemNCC\_Click(object sender, EventArgs e)**:
  + Chức năng: Xử lý sự kiện thêm nhà cung cấp, nhưng hiện tại phương thức này chưa được triển khai.
* **btnThem\_Click(object sender, EventArgs e)**:
  + Chức năng: Mở form NhaCungCap với chế độ thêm (Mode = "Them"). Nếu thêm thành công, gọi lại LoadData để cập nhật danh sách nhà cung cấp.
* **btnSua\_Click(object sender, EventArgs e)**:
  + Chức năng: Sửa thông tin nhà cung cấp được chọn từ dataGridView1.
  + Kiểm tra xem có dòng nào được chọn hay chưa. Nếu có, lấy MaNCC và mở form NhaCungCap ở chế độ sửa (Mode = "Sua"), truyền mã nhà cung cấp (MaNCC) để chỉnh sửa.
* **XoaNhaCungCap(int maNCC)**:
  + Chức năng: Thực hiện xóa nhà cung cấp có mã MaNCC trong cơ sở dữ liệu.
  + Kết nối đến cơ sở dữ liệu và xóa nhà cung cấp bằng truy vấn DELETE. Hiển thị thông báo nếu xóa thành công hoặc thông báo lỗi nếu có vấn đề xảy ra.
* **btnXoa\_Click(object sender, EventArgs e)**:
  + Chức năng: Xóa nhà cung cấp được chọn từ dataGridView1.
  + Kiểm tra xem có nhà cung cấp nào được chọn không. Nếu có, hiển thị thông báo xác nhận, sau đó gọi XoaNhaCungCap để thực hiện xóa nhà cung cấp và cập nhật lại danh sách sau khi xóa.
* **button1\_Click(object sender, EventArgs e)**:
  + Chức năng: Đóng form hiện tại và mở form Home, truyền các thông tin TenNV, CongViec, và MaNV.
* **button2\_Click(object sender, EventArgs e)**:
  + Chức năng: Mở form NhaCungCapReport để tạo báo cáo nhà cung cấp và truyền thông tin TenNV để hiển thị trên báo cáo.

### 2.3.8. Lớp QuanLyNhanVien (Kế thừa từ lớp Form)

**Thuộc tính:**

* TenNV: Chuỗi lưu tên của nhân viên hiện tại.
* CongViec: Chuỗi lưu công việc của nhân viên hiện tại.
* MaNV: Số nguyên lưu mã nhân viên hiện tại.

**Phương thức:**

* **QuanLyNhanVien()** *(Constructor)*:
  + Khởi tạo form quản lý nhân viên, gọi phương thức LoadData để tải dữ liệu nhân viên từ cơ sở dữ liệu.
* **QuanLyNhanVien(string tenNV, string congViec, int maNV)** *(Constructor có tham số)*:
  + Khởi tạo form quản lý nhân viên và khởi tạo các thông tin của nhân viên (TenNV, CongViec, MaNV), sau đó gọi phương thức LoadData để tải dữ liệu nhân viên từ cơ sở dữ liệu.
* **LoadData()**:
  + Kết nối đến cơ sở dữ liệu và thực hiện truy vấn để lấy thông tin từ bảng NhanVien.
  + Hiển thị dữ liệu trong dataGridView1.
* **dataGridView1\_CellContentClick(object sender, DataGridViewCellEventArgs e)**:
  + Phương thức xử lý sự kiện khi click vào một ô trong dataGridView1. Hiện tại chưa có nội dung xử lý cụ thể.
* **ThemNV\_Click(object sender, EventArgs e)**:
  + Mở form ThemNV để thêm nhân viên mới.
* **SuaNV\_Click(object sender, EventArgs e)**:
  + Sửa thông tin của nhân viên được chọn từ dataGridView1.
  + Kiểm tra xem có dòng nào được chọn không, nếu có, lấy các thông tin như mã nhân viên (MaNV), tên nhân viên (TenNV), giới tính, ngày sinh, điện thoại, địa chỉ, mã công việc để hiển thị và chỉnh sửa trong form SuaNV.
* **XoaNV\_Click(object sender, EventArgs e)**:
  + Xóa nhân viên được chọn từ dataGridView1.
  + Hiển thị hộp thoại xác nhận xóa, nếu người dùng đồng ý thì thực hiện câu lệnh DELETE để xóa nhân viên trong bảng NhanVien.
  + Sau khi xóa, gọi LoadData để cập nhật lại danh sách nhân viên trong dataGridView1.
* **exitNV\_Click(object sender, EventArgs e)**:
  + Đóng form hiện tại và mở form Home, đồng thời truyền các thông tin TenNV, CongViec, và MaNV.
* **btnbaocao\_Click(object sender, EventArgs e)**:
  + Mở form NhanVienReport để tạo báo cáo về nhân viên, truyền thông tin TenNV để hiển thị trong báo cáo.

### 2.3.9. Lớp DoanhThu (Kế thừa từ lớp Form)

**Thuộc tính:**

* TenNV: Chuỗi lưu tên của nhân viên hiện tại.
* CongViec: Chuỗi lưu công việc của nhân viên hiện tại.
* MaNV: Số nguyên lưu mã nhân viên hiện tại.

**Phương thức:**

* **DoanhThu()** *(Constructor)*:
  + Khởi tạo form doanh thu và thiết lập các thành phần giao diện.
* **DoanhThu(string tenNV, string congViec, int maNV)** *(Constructor có tham số)*:
  + Khởi tạo form và thiết lập thông tin người dùng (TenNV, CongViec, MaNV).
* **btnxemdoanhthu\_Click(object sender, EventArgs e)**:
  + Lấy dữ liệu doanh thu trong khoảng thời gian từ tuNgay đến denNgay từ cơ sở dữ liệu.
  + Thực hiện truy vấn để lấy các thông tin gồm ngày tháng, doanh thu bán, chi phí nhập, và doanh thu thuần.
  + Tính tổng doanh thu bán, tổng chi phí nhập, và tổng doanh thu thuần, rồi hiển thị kết quả lên các nhãn (lblTongDoanhThuBan, lblTongDoanhThuNhap, lblTongDoanhThuThuan).
  + Mở form ChiTietDoanhThu để hiển thị chi tiết doanh thu trong khoảng thời gian được chọn.
* **btnexit\_Click(object sender, EventArgs e)**:
  + Đóng form hiện tại và mở form Home, truyền các thông tin TenNV, CongViec, và MaNV.
* **btnxemdoanhthutheothang\_Click(object sender, EventArgs e)**:
  + Mở form DoanhThuTheoThang, cho phép người dùng xem doanh thu theo từng tháng.
* **button1\_Click(object sender, EventArgs e)**:
  + Mở form DoanhThuReport để tạo báo cáo doanh thu, truyền thông tin TenNV để hiển thị trong báo cáo.

## 2.4. Mô hình cơ sở dữ liệu

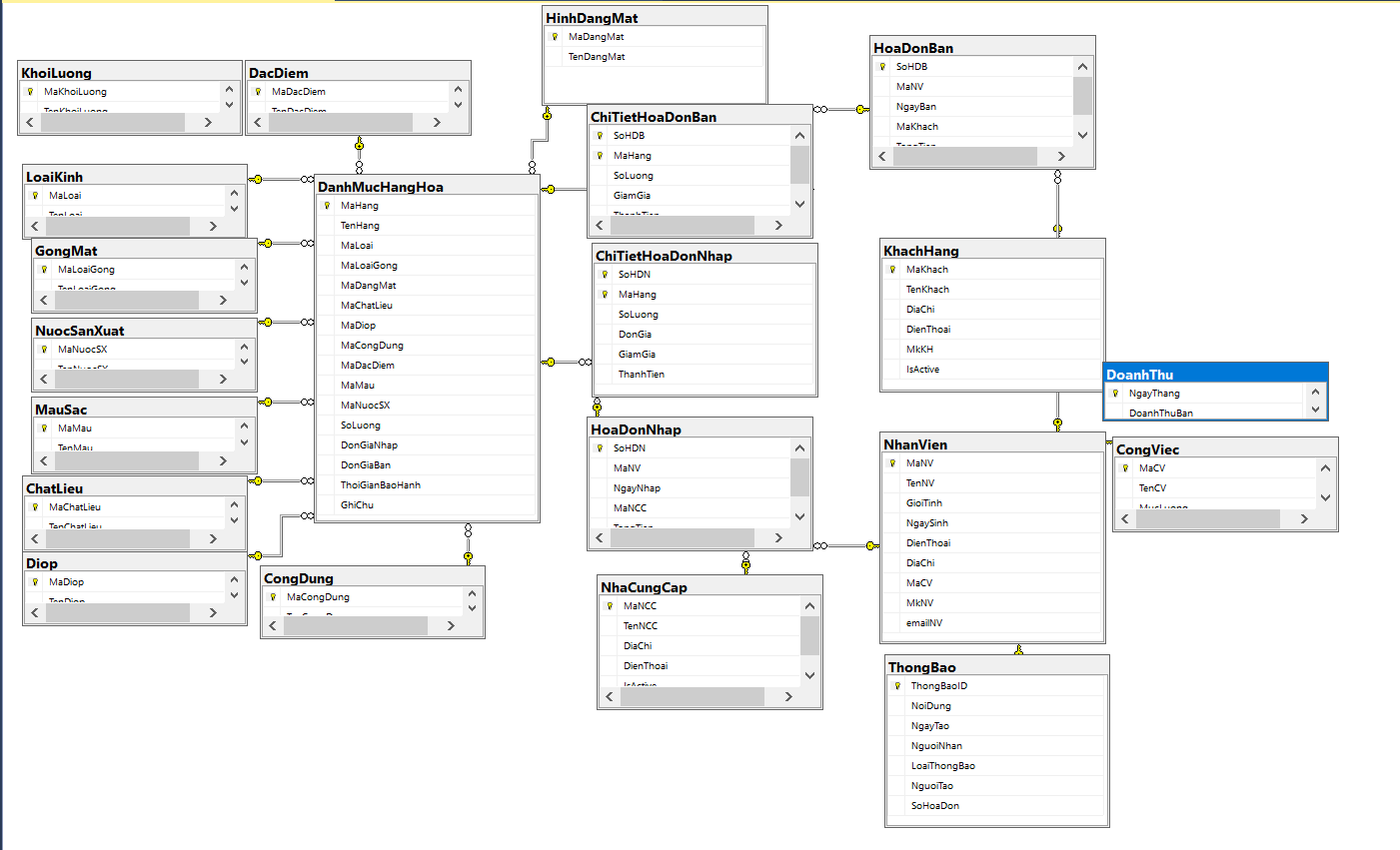
Cơ sở dữ liệu của hệ thống bao gồm các bảng chính như sau:

1. **KhoiLuong**:
   * Thuộc tính: MaKhoiLuong (Primary Key), TenKhoiLuong.
2. **LoaiKinh**:
   * Thuộc tính: MaLoai (Primary Key), TenLoai.
3. **HinhDangMat**:
   * Thuộc tính: MaDangMat (Primary Key), TenDangMat.
4. **ChatLieu**:
   * Thuộc tính: MaChatLieu (Primary Key), TenChatLieu.
5. **NuocSanXuat**:
   * Thuộc tính: MaNuocSX (Primary Key), TenNuocSX.
6. **MauSac**:
   * Thuộc tính: MaMau (Primary Key), TenMau.
7. **Diop**:
   * Thuộc tính: MaDiop (Primary Key), TenDiop.
8. **CongViec**:
   * Thuộc tính: MaCV (Primary Key), TenCV, MucLuong.
9. **CongDung**:
   * Thuộc tính: MaCongDung (Primary Key), TenCongDung.
10. **KhachHang**:
    * Thuộc tính: MaKhach (Primary Key), TenKhach, DiaChi, DienThoai.
11. **NhaCungCap**:
    * Thuộc tính: MaNCC (Primary Key), TenNCC, DiaChi, DienThoai.
12. **DacDiem**:
    * Thuộc tính: MaDacDiem (Primary Key), TenDacDiem.
13. **GongMat**:
    * Thuộc tính: MaLoaiGong (Primary Key), TenLoaiGong.
14. **NhanVien**:
    * Thuộc tính: MaNV (Primary Key), TenNV, GioiTinh, NgaySinh, DienThoai, DiaChi, MaCV, MkNV, emailNV.
15. **DanhMucHangHoa**:
    * Thuộc tính: MaHang (Primary Key), TenHang, MaLoai, MaLoaiGong, MaDangMat, MaChatLieu, MaDiop, MaCongDung, MaDacDiem, MaMau, MaNuocSX, SoLuong, DonGiaNhap, DonGiaBan, ThoiGianBaoHanh, GhiChu.
16. **HoaDonBan**:
    * Thuộc tính: SoHDB (Primary Key), MaNV, NgayBan, MaKhach, TongTien.
17. **HoaDonNhap**:
    * Thuộc tính: SoHDN (Primary Key), MaNV, NgayNhap, MaNCC, TongTien.
18. **ChiTietHoaDonBan**:
    * Thuộc tính: SoHDB (Primary Key), MaHang (Primary Key), SoLuong, GiamGia, ThanhTien.
19. **ChiTietHoaDonNhap**:
    * Thuộc tính: SoHDN (Primary Key), MaHang (Primary Key), SoLuong, DonGia, GiamGia, ThanhTien.
20. **DoanhThu**:
    * Thuộc tính: NgayThang (Primary Key), DoanhThuBan, DoanhThuNhap, DoanhThuThuần.
21. **ThongBao**:
    * Thuộc tính: ThongBaoID (Primary Key), NoiDung, NgayTao, NguoiNhan, LoaiThongBao, SoHoaDon.

**Mối quan hệ giữa các thực thể (các bảng):**

1. NhanVien và HoaDonBan có mối quan hệ Một-Nhiều (một nhân viên có thể tạo nhiều hóa đơn bán hàng).
2. NhanVien và HoaDonNhap có mối quan hệ Một-Nhiều (một nhân viên có thể tạo nhiều hóa đơn nhập hàng).
3. KhachHang và HoaDonBan có mối quan hệ Một-Nhiều (một khách hàng có thể có nhiều hóa đơn mua hàng).
4. NhaCungCap và HoaDonNhap có mối quan hệ Một-Nhiều (một nhà cung cấp có thể có nhiều hóa đơn cung cấp hàng hóa).
5. DanhMucHangHoa và ChiTietHoaDonBan có mối quan hệ Một-Nhiều (một sản phẩm có thể xuất hiện trong nhiều hóa đơn bán hàng).
6. DanhMucHangHoa và ChiTietHoaDonNhap có mối quan hệ Một-Nhiều (một sản phẩm có thể xuất hiện trong nhiều hóa đơn nhập hàng).

Sơ đồ diagram:



## 2.5. Các tiện ích

Có các button để tìm kiếm nhanh trên giao diện Home (Trang Chủ):

* **Tìm sản phẩm:**  Tìm kiếm sản phẩm dựa theo Tên sản phẩm, Màu sắc, Chất liệu, Hình dạng mắt. Sau đó hiển thị ra màn hình các thông tin của kính có đặc điểm đó
* **Tìm nhân viên:** Tìm kiếm nhân viên dựa theo Tên Nhân viên hoặc Mã nhân viên. Sau đó hiển thị ra các thông tin cá nhân của nhân viên đó
* **Tìm hóa đơn:** Tìm kiếm hóa đơn dựa trên mã hóa đơn từ đó hiển thị các thông tin về hóa đơn đó
* **Tìm khách hàng:** Tìm kiếm khách hàng theo mã khách hàng hoặc tên khách hàng. Sau đó hiển thị ra thông tin cá nhân của khách hàng đó và danh sách Tên hàng hóa, Số lượng, Ngày mua, Tổng tiền của khách hàng đó

# Chương 3: Phát triển hệ thống

## 3.1. Lớp GUI

### 3.1.1. Form Account:

**Form Account** là form đăng nhập cho hệ thống.

**Chức năng chính của form Account:**

* **LoadSavedCredentials()**: Tải thông tin tài khoản đã lưu (nếu có) và điền vào các ô nhập liệu. Nếu người dùng đã chọn lưu mật khẩu trước đó, các thông tin này sẽ tự động hiển thị.
* **Login\_Click**: Xử lý sự kiện khi người dùng nhấn nút "Login". Phương thức này sẽ:
  + Xác thực thông tin đăng nhập của người dùng bằng cách gọi AuthenticateUser(userInput, pass).
  + Nếu đăng nhập thành công, kiểm tra xem người dùng có chọn lưu mật khẩu không và lưu thông tin nếu cần.
  + Khởi tạo và mở form Home (màn hình chính sau đăng nhập) với các thông tin của nhân viên đã đăng nhập (bao gồm tên, công việc, mã nhân viên).
  + Ẩn form Account.
  + Nếu đăng nhập thất bại, hiển thị thông báo lỗi.
* **AuthenticateUser()**: Xác thực tài khoản của người dùng bằng cách truy vấn dữ liệu từ bảng NhanVien trong cơ sở dữ liệu. Nếu thông tin hợp lệ, phương thức này sẽ trả về các thông tin của nhân viên (tên, công việc, mã nhân viên) để mở form Home.
* **SaveCredentials()**: Lưu tên đăng nhập và mật khẩu vào cài đặt nếu người dùng chọn lưu thông tin.
* **ClearSavedCredentials()**: Xóa thông tin tài khoản đã lưu khi người dùng bỏ chọn hộp kiểm "Lưu thông tin".

**Mối quan hệ với các Form khác:**

* **Home**: Khi người dùng đăng nhập thành công, Account sẽ khởi tạo và mở form Home (màn hình chính của hệ thống).
* **ResetPassword**: Khi người dùng nhấp vào liên kết "Quên mật khẩu?", Account sẽ mở form ResetPassword để người dùng có thể đặt lại mật khẩu.

### 3.1.2. Form Home

**Form Home** là giao diện chính của hệ thống sau khi người dùng đăng nhập thành công. Nó cung cấp các chức năng quản lý khác nhau như quản lý nhân viên, nhà cung cấp, khách hàng, hóa đơn nhập, hóa đơn bán, và doanh thu.

**Chức năng chính của form Home:**

* **UpdateUI()**: Cập nhật giao diện người dùng, hiển thị tên và chức vụ của nhân viên đã đăng nhập. Nếu chức vụ là "Quản lý", nó sẽ hiển thị các nút điều khiển bổ sung như QLNV, btndoanhthu, button3, và button4.
* **LoadThongBao()**: Tải và hiển thị thông báo cho người dùng. Nếu người dùng là quản lý, hệ thống sẽ hiển thị tất cả thông báo; nếu không, chỉ hiển thị các thông báo dành riêng cho nhân viên đó.
* **LayEmailTuMaNV()**: Truy xuất email của nhân viên dựa trên mã nhân viên (MaNV). Email này được sử dụng trong các trường hợp như đặt lại mật khẩu.

**Mối quan hệ với các Form khác:**

* **QuanLyNhanVien**: Khi người dùng nhấn vào nút QLNV, form QuanLyNhanVien sẽ mở, cho phép quản lý nhân viên. Home truyền các thông tin TenNV, CongViec, và MaNV cho QuanLyNhanVien.
* **QuanLyNhaCungCap**: Khi người dùng nhấn vào nút QLNCC, hệ thống sẽ mở form QuanLyNhaCungCap với các thông tin về nhân viên.
* **QuanLyKhachHang**: Tương tự, nhấn vào QLKH sẽ mở form QuanLyKhachHang.
* **QuanLyHoaDonNhap** và **QuanLyHoaDonBan**: Nhấn vào QLHDN hoặc QLHDB sẽ mở form QuanLyHoaDonNhap hoặc QuanLyHoaDonBan để quản lý hóa đơn nhập và hóa đơn bán.
* **DoanhThu**: Nút btndoanhthu sẽ mở form DoanhThu để xem báo cáo doanh thu nếu người dùng có quyền.
* **ThemHangHoa**, **TimNhanVien**, **ThemNV**, **TimKhachHang**, **TimSanPham2**, **TimHoaDon**: Các nút khác trên form Home mở các form tương ứng để thêm hoặc tìm kiếm các thông tin như hàng hóa, nhân viên, khách hàng, sản phẩm, và hóa đơn.
* **DoiMatKhau**: Nút rspass sẽ mở form DoiMatKhau, truyền email của nhân viên vào form này để hỗ trợ đổi mật khẩu.

### 3.1.3. Form QuanLyDanhMucHangHoa

**Form QuanLyDanhMucHangHoa** là giao diện quản lý danh mục hàng hóa, cho phép thêm, sửa, xóa, và xuất danh sách hàng hóa ra file Excel. Nó hiển thị các thông tin liên quan như mã sản phẩm, loại kính, chất liệu, công dụng, màu sắc, và xuất xứ.

**Chức năng chính của form QuanLyDanhMucHangHoa**:

* **loadData()**: Kết nối và tải dữ liệu từ bảng DanhMucHangHoa cùng với các bảng liên quan như LoaiKinh, GongMat, ChatLieu, DacDiem, và NuocSanXuat. Dữ liệu sau đó được hiển thị trong DataGridView với các tiêu đề cột được tùy chỉnh để người dùng dễ dàng quản lý.
* **Them\_Click**: Mở form MainHangHoa để thêm hàng hóa mới vào danh mục. Khi hàng hóa mới được thêm thành công, loadData() sẽ được gọi lại để cập nhật danh sách.
* **Sua\_Click**: Nếu có hàng hóa được chọn trong DataGridView, các thông tin chi tiết sẽ được lấy và truyền vào form SuaHangHoa. Sau khi chỉnh sửa, danh sách sẽ được cập nhật lại.
* **Xoa\_Click**: Thực hiện xóa hàng hóa đã chọn. Hệ thống sẽ hiển thị hộp thoại xác nhận trước khi xóa. Nếu hàng hóa có liên kết với hóa đơn, hệ thống sẽ thông báo lỗi.
* **btnXuatExcel\_Click**: Cho phép xuất dữ liệu hiện tại trong DataGridView ra file Excel.
* **dataGridView1\_CellFormatting**: Định dạng các cột giá nhập và giá bán để hiển thị số nguyên khi không có phần thập phân, hoặc tối đa 2 chữ số thập phân khi có phần lẻ.
* **button1\_Click**: Mở form HangHoaReport để xem báo cáo hàng hóa chi tiết.

**Mối quan hệ với các Form khác**:

* **MainHangHoa**: Mở khi nhấn nút thêm hàng hóa, cho phép thêm hàng hóa mới vào danh mục.
* **SuaHangHoa**: Mở khi nhấn nút sửa hàng hóa, truyền dữ liệu từ hàng được chọn trong DataGridView để hỗ trợ chỉnh sửa thông tin.
* **Home**: Mở khi nhấn nút thoát khỏi form QuanLyDanhMucHangHoa, đồng thời truyền TenNV, CongViec, và MaNV về form Home.
* **HangHoaReport**: Mở khi nhấn nút báo cáo (button1\_Click) để xem báo cáo chi tiết về hàng hóa.

### 3.1.4. Form QuanLyHoaDonBan

**Form QuanLyHoaDonBan** là một giao diện quản lý hóa đơn bán hàng, với các chức năng như tải và hiển thị dữ liệu hóa đơn, thêm mới, xóa hóa đơn, xem chi tiết, và in hóa đơn. Các hóa đơn trong DataGridView được lấy từ cơ sở dữ liệu và cho phép người dùng quản lý hiệu quả các hóa đơn bán hàng.

**Chức năng chính của form QuanLyHoaDonBan:**

* **LoadData()**: Phương thức này kết nối với cơ sở dữ liệu để lấy thông tin hóa đơn từ bảng HoaDonBan và ChiTietHoaDonBan. Dữ liệu bao gồm các cột như SoHDB, MaNV, NgayBan, MaKhach, TongSoLuongSanPham, và ThanhTien. Dữ liệu sau đó được hiển thị trên DataGridView.
* **btnXemChiTiet\_Click**: Khi người dùng chọn một hàng và nhấn "Xem Chi Tiết", phương thức này sẽ lấy SoHDB từ hóa đơn được chọn và mở form ChiTietHoaDonBan để hiển thị chi tiết hóa đơn đó.
* **btnThemHD\_Click**: Mở form HoaDonBan, cho phép người dùng thêm mới một hóa đơn bán hàng. Form này được truyền TenNV và MaNV để ghi lại thông tin nhân viên tạo hóa đơn.
* **btnXoaHD\_Click**: Nếu người dùng chọn một hóa đơn và nhấn "Xóa Hóa Đơn", hệ thống sẽ xác nhận trước khi thực hiện xóa. Phương thức này sẽ xóa hóa đơn và các chi tiết liên quan trong bảng ChiTietHoaDonBan nếu không có lỗi.
* **btnIn\_Click**: Cho phép in hóa đơn. Khi người dùng chọn một hóa đơn và nhấn "In", thông tin hóa đơn sẽ được truy vấn từ cơ sở dữ liệu, sau đó hiển thị trong giao diện in với các chi tiết như NgayBan, TenKhach, TenNV, TenHang, SoLuong, và ThanhTien. Tổng tiền và lời cảm ơn cũng được in cuối hóa đơn.
* **dataGridView1\_CellFormatting**: Định dạng cột TongTien và ThanhTien để hiển thị số tiền dưới dạng không có phần thập phân khi là số nguyên hoặc tối đa 2 chữ số thập phân nếu có phần lẻ.
* **btnbaocao\_Click**: Mở form HoaDonBanReport, hiển thị báo cáo các hóa đơn bán hàng theo thông tin của TenNV (nhân viên đã đăng nhập).

**Mối quan hệ với các Form khác:**

* **ChiTietHoaDonBan**: Được mở khi nhấn "Xem Chi Tiết" để hiển thị chi tiết về sản phẩm và số lượng trong hóa đơn.
* **HoaDonBan**: Được mở khi nhấn "Thêm Hóa Đơn", cung cấp giao diện để nhập thông tin và lưu hóa đơn mới vào cơ sở dữ liệu.
* **Home**: Được mở khi nhấn nút thoát (exit\_Click). Form Home nhận các tham số TenNV, CongViec, và MaNV để duy trì thông tin đăng nhập của nhân viên.
* **HoaDonBanReport**: Được mở khi nhấn "Báo Cáo", hiển thị báo cáo hóa đơn bán hàng dựa trên thông tin nhân viên hiện tại.

### 3.1.5. Form QuanLyHoaDonNhap

**Form QuanLyHoaDonNhap** là một giao diện quản lý hóa đơn nhập hàng, với các chức năng như tải và hiển thị dữ liệu hóa đơn, thêm mới, xóa hóa đơn, xem chi tiết, và tạo báo cáo. Các hóa đơn được truy xuất từ cơ sở dữ liệu SQL Server và hiển thị trong DataGridView, giúp người dùng quản lý hiệu quả các hóa đơn nhập hàng.

**Chức năng chính của form QuanLyHoaDonNhap**

1. **LoadData()**:
   * Phương thức này kết nối với cơ sở dữ liệu để lấy thông tin hóa đơn từ bảng HoaDonNhap và ChiTietHoaDonNhap.
   * Dữ liệu bao gồm các cột SoHDN, NgayTao, MaNV, MaNCC, LoaiMatHang (số lượng loại mặt hàng), TongSoLuong, và TongTien.
   * Dữ liệu sau đó được hiển thị trên DataGridView.
2. **button1\_Click (Thêm chi tiết hóa đơn nhập)**:
   * Khi người dùng nhấn nút "Thêm Chi Tiết Hóa Đơn Nhập", form ThemChiTietHoaDonNhap2 sẽ mở, cho phép người dùng thêm chi tiết sản phẩm vào hóa đơn nhập mới. Form này nhận TenNV và MaNV để lưu thông tin nhân viên đã tạo hóa đơn.
3. **button3\_Click (Xóa hóa đơn nhập)**:
   * Nếu người dùng chọn một hóa đơn và nhấn "Xóa Hóa Đơn Nhập", hệ thống sẽ xác nhận trước khi thực hiện xóa.
   * Phương thức này sẽ xóa hóa đơn và các chi tiết liên quan trong bảng ChiTietHoaDonNhap nếu không có lỗi xảy ra.
   * Phương thức sử dụng SqlTransaction để đảm bảo dữ liệu được xóa một cách nhất quán từ cả hai bảng HoaDonNhap và ChiTietHoaDonNhap.
4. **button4\_Click (Xem chi tiết hóa đơn nhập)**:
   * Khi người dùng chọn một hàng và nhấn "Xem Chi Tiết", phương thức này sẽ lấy SoHDN từ hóa đơn được chọn và mở form ChiTietHoaDonNhap để hiển thị chi tiết hóa đơn, bao gồm thông tin sản phẩm, số lượng, đơn giá và thành tiền.
5. **dataGridView1\_CellFormatting**:
   * Định dạng cột TongTien và DonGia để hiển thị số tiền dưới dạng không có phần thập phân khi là số nguyên hoặc tối đa 2 chữ số thập phân nếu có phần lẻ.
   * Điều này giúp người dùng dễ dàng đọc và quản lý các giá trị tiền tệ trong bảng.
6. **btnbaocao\_Click (Tạo báo cáo)**:
   * Mở form HoaDonNhapReport, hiển thị báo cáo các hóa đơn nhập hàng theo thông tin của nhân viên hiện tại (TenNV).
   * Báo cáo bao gồm thông tin như NgayTao, SoHDN, MaNV, MaNCC, TongSoLuong, và TongTien cho từng hóa đơn.
7. **exit\_Click (Thoát)**:
   * Nút thoát cho phép quay lại form Home, truyền các tham số TenNV, CongViec, và MaNV để duy trì thông tin đăng nhập của nhân viên.

**Mối quan hệ với các Form khác:**

* **ThemChiTietHoaDonNhap2**: Được mở khi nhấn "Thêm Chi Tiết Hóa Đơn Nhập", cung cấp giao diện để nhập thông tin chi tiết sản phẩm và lưu vào cơ sở dữ liệu.
* **ChiTietHoaDonNhap**: Được mở khi nhấn "Xem Chi Tiết" để hiển thị các chi tiết về sản phẩm và số lượng trong hóa đơn nhập.
* **Home**: Được mở khi nhấn nút thoát (exit\_Click), duy trì thông tin đăng nhập của nhân viên hiện tại.
* **HoaDonNhapReport**: Được mở khi nhấn "Báo Cáo", hiển thị báo cáo hóa đơn nhập hàng dựa trên thông tin nhân viên hiện tại.

### 3.1.6. Form QuanLyKhachHang

**Form QuanLyKhachHang** là một giao diện quản lý thông tin khách hàng, bao gồm các chức năng chính như tải và hiển thị danh sách khách hàng, thêm mới, sửa, xóa, và tạo báo cáo khách hàng. Các thông tin khách hàng được truy xuất từ cơ sở dữ liệu SQL Server và hiển thị trên DataGridView, giúp người dùng quản lý khách hàng một cách hiệu quả.

**Chức năng chính của form QuanLyKhachHang**

1. **LoadData()**:
   * Phương thức này kết nối với cơ sở dữ liệu để lấy danh sách khách hàng từ bảng KhachHang.
   * Dữ liệu bao gồm các cột MaKhach, TenKhach, DiaChi, và DienThoai.
   * Dữ liệu sau đó được hiển thị trên DataGridView để người dùng có thể xem, chọn và thực hiện các thao tác khác trên danh sách khách hàng.
2. **themKH\_Click (Thêm khách hàng)**:
   * Khi người dùng nhấn nút "Thêm Khách Hàng", form ThemKH sẽ mở, cung cấp giao diện để nhập thông tin khách hàng mới và lưu vào cơ sở dữ liệu.
   * Sau khi form này được hoàn thành và đóng lại, dữ liệu mới sẽ được tải lại và hiển thị trên DataGridView.
3. **suaKH\_Click (Sửa khách hàng)**:
   * Khi người dùng chọn một khách hàng từ DataGridView và nhấn "Sửa Khách Hàng", phương thức này sẽ lấy mã khách hàng và các thông tin chi tiết từ hàng được chọn.
   * Sau đó, form SuaKH sẽ mở, hiển thị thông tin hiện tại của khách hàng và cho phép người dùng chỉnh sửa.
   * Sau khi chỉnh sửa hoàn tất, SuaKH sẽ lưu lại các thay đổi vào cơ sở dữ liệu.
4. **xoaKH\_Click (Xóa khách hàng)**:
   * Khi người dùng chọn một khách hàng từ DataGridView và nhấn "Xóa Khách Hàng", hệ thống sẽ yêu cầu xác nhận trước khi thực hiện xóa.
   * Phương thức này sẽ xóa khách hàng khỏi bảng KhachHang trong cơ sở dữ liệu nếu không có lỗi xảy ra.
   * Nếu xóa thành công, dữ liệu trên DataGridView sẽ được cập nhật lại để hiển thị danh sách mới.
5. **btnbaocao\_Click (Tạo báo cáo)**:
   * Mở form KhachHangReport, hiển thị báo cáo thông tin khách hàng dựa trên thông tin của nhân viên hiện tại (TenNV).
   * Báo cáo bao gồm các thông tin chi tiết như MaKhach, TenKhach, DiaChi, và DienThoai của từng khách hàng.
6. **exit\_Click (Thoát)**:
   * Khi người dùng nhấn nút thoát, form Home sẽ mở và truyền các tham số TenNV, CongViec, và MaNV để duy trì thông tin đăng nhập của nhân viên hiện tại.

**Mối quan hệ với các Form khác:**

* **ThemKH**: Được mở khi nhấn "Thêm Khách Hàng", cung cấp giao diện để nhập thông tin khách hàng mới và lưu vào cơ sở dữ liệu.
* **SuaKH**: Được mở khi nhấn "Sửa Khách Hàng", cung cấp giao diện để chỉnh sửa thông tin khách hàng đã có.
* **Home**: Được mở khi nhấn nút thoát (exit\_Click), duy trì thông tin đăng nhập của nhân viên hiện tại.
* **KhachHangReport**: Được mở khi nhấn "Báo Cáo", hiển thị báo cáo danh sách khách hàng hiện có.

### 3.1.7. Form QuanLyNhaCungCap

**Form QuanLyNhaCungCap** là một giao diện quản lý thông tin nhà cung cấp, bao gồm các chức năng chính như tải và hiển thị danh sách nhà cung cấp, thêm mới, sửa, xóa, và tạo báo cáo. Thông tin nhà cung cấp được truy xuất từ cơ sở dữ liệu SQL Server và hiển thị trên DataGridView, giúp người dùng quản lý nhà cung cấp hiệu quả.

**Chức năng chính của form QuanLyNhaCungCap**

1. **LoadData()**:
   * Phương thức này kết nối với cơ sở dữ liệu để lấy danh sách nhà cung cấp từ bảng NhaCungCap.
   * Dữ liệu sau đó được hiển thị trên DataGridView, cho phép người dùng xem và lựa chọn các nhà cung cấp để thao tác khác.
2. **btnThem\_Click (Thêm nhà cung cấp)**:
   * Khi người dùng nhấn nút "Thêm", form NhaCungCap sẽ mở với chế độ Mode được đặt là "Them", cho phép nhập thông tin nhà cung cấp mới.
   * Sau khi hoàn thành và đóng form NhaCungCap, LoadData() sẽ được gọi lại để cập nhật DataGridView với thông tin nhà cung cấp mới.
3. **btnSua\_Click (Sửa nhà cung cấp)**:
   * Khi người dùng chọn một nhà cung cấp từ DataGridView và nhấn "Sửa", phương thức này lấy mã nhà cung cấp (MaNCC) từ hàng được chọn.
   * Form NhaCungCap sẽ mở với Mode được đặt là "Sua" và MaNCC của nhà cung cấp đã chọn.
   * Người dùng có thể sửa thông tin nhà cung cấp. Sau khi lưu lại, LoadData() sẽ được gọi lại để cập nhật DataGridView.
4. **XoaNhaCungCap() (Xóa nhà cung cấp)**:
   * Phương thức này thực hiện thao tác xóa nhà cung cấp từ bảng NhaCungCap trong cơ sở dữ liệu dựa trên mã nhà cung cấp (MaNCC) đã chọn.
   * Sau khi xóa, phương thức LoadData() được gọi để cập nhật lại danh sách nhà cung cấp trong DataGridView.
5. **btnXoa\_Click (Nút xóa)**:
   * Khi người dùng chọn một nhà cung cấp từ DataGridView và nhấn "Xóa", hệ thống sẽ yêu cầu xác nhận trước khi thực hiện.
   * Nếu xác nhận, phương thức XoaNhaCungCap() sẽ xóa nhà cung cấp đã chọn và cập nhật lại DataGridView.
6. **button1\_Click (Thoát)**:
   * Khi người dùng nhấn nút thoát, form Home sẽ mở và truyền các tham số TenNV, CongViec, và MaNV để duy trì thông tin đăng nhập của nhân viên hiện tại.
   * Form QuanLyNhaCungCap sẽ đóng sau khi form Home mở.
7. **button2\_Click (Báo cáo)**:
   * Mở form NhaCungCapReport, hiển thị báo cáo danh sách nhà cung cấp dựa trên thông tin của nhân viên hiện tại (TenNV).
   * Báo cáo bao gồm thông tin chi tiết của các nhà cung cấp như MaNCC, TenNCC, DiaChi, và DienThoai.

**Mối quan hệ với các Form khác:**

* **NhaCungCap**: Được mở khi nhấn "Thêm" hoặc "Sửa", cho phép thêm mới hoặc chỉnh sửa thông tin nhà cung cấp.
* **Home**: Được mở khi nhấn nút thoát (button1\_Click), duy trì thông tin đăng nhập của nhân viên hiện tại.
* **NhaCungCapReport**: Được mở khi nhấn nút "Báo Cáo" (button2\_Click), hiển thị báo cáo danh sách nhà cung cấp hiện có.

### 3.1.8. Form QuanLyNhanVien

**Form QuanLyNhanVien** là giao diện quản lý thông tin nhân viên với các chức năng chính như tải danh sách nhân viên, thêm mới, sửa, xóa, và xuất báo cáo. Dữ liệu nhân viên được lấy từ cơ sở dữ liệu SQL Server và hiển thị trên DataGridView để người dùng dễ dàng thao tác.

**Chức năng chính của form QuanLyNhanVien**

1. **LoadData()**:
   * Phương thức này kết nối với cơ sở dữ liệu và truy vấn danh sách nhân viên từ bảng NhanVien.
   * Dữ liệu được hiển thị trên DataGridView, cho phép người dùng xem và chọn nhân viên để thực hiện các thao tác khác.
2. **ThemNV\_Click (Thêm nhân viên)**:
   * Khi người dùng nhấn nút "Thêm", form ThemNV sẽ mở lên để nhập thông tin cho nhân viên mới.
   * Sau khi form ThemNV đóng, dữ liệu DataGridView sẽ được cập nhật bằng cách gọi lại LoadData().
3. **SuaNV\_Click (Sửa nhân viên)**:
   * Khi người dùng chọn một nhân viên từ DataGridView và nhấn "Sửa", phương thức sẽ lấy thông tin từ hàng được chọn, bao gồm mã nhân viên (MaNV), tên, giới tính, ngày sinh, điện thoại, địa chỉ, và mã công việc.
   * Form SuaNV sẽ mở với thông tin nhân viên cần sửa đã được truyền vào, cho phép người dùng cập nhật dữ liệu.
   * Sau khi thông tin được lưu, LoadData() sẽ được gọi lại để cập nhật DataGridView.
4. **XoaNV\_Click (Xóa nhân viên)**:
   * Khi người dùng chọn một nhân viên và nhấn "Xóa", một hộp thoại xác nhận sẽ xuất hiện để yêu cầu người dùng xác nhận thao tác.
   * Nếu được xác nhận, hệ thống sẽ thực hiện lệnh xóa nhân viên khỏi bảng NhanVien dựa trên mã nhân viên (MaNV) đã chọn.
   * Sau khi xóa thành công, LoadData() sẽ được gọi lại để cập nhật DataGridView.
5. **exitNV\_Click (Thoát)**:
   * Khi người dùng nhấn nút thoát, form Home sẽ mở lên với các thông tin đăng nhập hiện tại của nhân viên, bao gồm TenNV, CongViec, và MaNV.
   * Form QuanLyNhanVien sẽ đóng lại khi form Home mở.
6. **btnbaocao\_Click (Báo cáo)**:
   * Khi nhấn nút "Báo Cáo", form NhanVienReport sẽ mở để hiển thị báo cáo danh sách nhân viên hiện tại.
   * Báo cáo này cung cấp thông tin tổng hợp của các nhân viên trong công ty, giúp quản lý dễ dàng nắm bắt thông tin.

**Mối quan hệ với các Form khác:**

* **ThemNV**: Được mở khi nhấn "Thêm" để nhập thông tin nhân viên mới.
* **SuaNV**: Được mở khi nhấn "Sửa", cho phép người dùng cập nhật thông tin nhân viên đã có trong cơ sở dữ liệu.
* **Home**: Được mở khi nhấn nút thoát (exitNV\_Click), lưu lại thông tin đăng nhập của nhân viên hiện tại.
* **NhanVienReport**: Được mở khi nhấn nút "Báo Cáo" (btnbaocao\_Click), cung cấp báo cáo danh sách nhân viên chi tiết.

### 3.1.9. Form DoanhThu

**Form DoanhThu** là giao diện quản lý và theo dõi tình hình doanh thu của công ty trong một khoảng thời gian nhất định. Người dùng có thể xem tổng doanh thu, chi phí nhập, lợi nhuận thuần, và chi tiết doanh thu theo các thời gian cụ thể.

**Chức năng chính của form DoanhThu**

1. **btnxemdoanhthu\_Click (Xem doanh thu theo khoảng thời gian)**:
   * Khi người dùng nhấn nút "Xem Doanh Thu", hệ thống sẽ lấy thông tin doanh thu từ bảng DoanhThu trong cơ sở dữ liệu SQL Server dựa trên khoảng thời gian người dùng chọn (từ dtpngaybatdau đến dtpngayketthuc).
   * Câu truy vấn SQL sử dụng điều kiện BETWEEN để lọc các bản ghi trong phạm vi thời gian được chọn.
   * Dữ liệu được hiển thị trên DataGridView, bao gồm các thông tin như ngày tháng, doanh thu bán, chi phí nhập và lợi nhuận thuần.
   * Sau khi tải dữ liệu, các tổng giá trị (doanh thu bán, chi phí nhập, lợi nhuận thuần) sẽ được tính toán và hiển thị ở các label (lblTongDoanhThuBan, lblTongDoanhThuNhap, lblTongDoanhThuThuan).
   * Sau khi hiển thị bảng, form ChiTietDoanhThu cũng sẽ mở ra để hiển thị chi tiết doanh thu theo khoảng thời gian đã chọn.
2. **btnexit\_Click (Thoát về trang chủ)**:
   * Khi nhấn nút thoát (btnexit), form Home sẽ mở lại với các thông tin đăng nhập của nhân viên hiện tại (TenNV, CongViec, MaNV).
   * Form DoanhThu sẽ đóng lại khi form Home mở.
3. **btnxemdoanhthutheothang\_Click (Xem doanh thu theo tháng)**:
   * Khi nhấn nút "Xem Doanh Thu Theo Tháng", form DoanhThuTheoThang sẽ được mở để người dùng có thể xem báo cáo doanh thu theo tháng, thay vì theo một khoảng thời gian cụ thể.
4. **button1\_Click (Xuất báo cáo doanh thu)**:
   * Khi nhấn nút "Báo Cáo Doanh Thu", form DoanhThuReport sẽ được mở để tạo và hiển thị báo cáo doanh thu cho nhân viên đang đăng nhập.
   * Báo cáo doanh thu này có thể chứa thông tin tổng hợp về tình hình doanh thu của công ty, giúp người quản lý có cái nhìn tổng quan về hiệu quả hoạt động kinh doanh.

## 3.2. Lớp BUS

### 3.2.1. Lớp Account (Xử lý Đăng nhập)

Lớp này chủ yếu xử lý các chức năng liên quan đến việc đăng nhập người dùng và quản lý thông tin đăng nhập. Cụ thể:

* **Chức năng đăng nhập:**
  + **Login\_Click(object sender, EventArgs e)**: Xử lý sự kiện khi người dùng nhấn nút đăng nhập, xác thực thông tin tài khoản và chuyển hướng đến form Home nếu đăng nhập thành công.
  + **AuthenticateUser(string userInput, string password)**: Kiểm tra tài khoản và mật khẩu của người dùng từ cơ sở dữ liệu và trả về thông tin người dùng nếu hợp lệ.
  + **GetUserRole(string maNV)**: Lấy thông tin công việc của nhân viên từ cơ sở dữ liệu thông qua mã nhân viên (maNV).
* **Lưu và xóa thông tin đăng nhập:**
  + **LoadSavedCredentials()**: Tải thông tin đăng nhập đã lưu (nếu có) vào các trường thông tin đăng nhập.
  + **SaveCredentials(string username, string password)**: Lưu thông tin đăng nhập nếu người dùng chọn lưu.
  + **ClearSavedCredentials()**: Xóa thông tin đăng nhập đã lưu.
* **Thao tác với checkbox:**
  + **checkBox1\_CheckedChanged(object sender, EventArgs e)**: Khi checkbox "Lưu thông tin đăng nhập" được chọn hay bỏ chọn, sẽ lưu hoặc xóa thông tin đăng nhập tương ứng.
* **Quên mật khẩu:**
  + **linkLabel1\_LinkClicked(object sender, LinkLabelLinkClickedEventArgs e)**: Mở form ResetPassword để người dùng có thể thay đổi mật khẩu khi quên.

### 3.2.2. Lớp Home (Form chính sau khi đăng nhập)

Lớp này quản lý giao diện người dùng sau khi người dùng đăng nhập thành công, hiển thị thông tin nhân viên, công việc và các chức năng theo vai trò của người dùng. Cụ thể:

* **Thuộc tính**:
  + TenNV, CongViec, MaNV: Lưu trữ thông tin về tên nhân viên, công việc và mã nhân viên của người dùng đăng nhập.
* **Chức năng chính**:
  + **Hiển thị các chức năng tùy thuộc vào vai trò người dùng**:
    - Trong phương thức UpdateUI(), nếu người dùng có công việc là "Quản lý", các chức năng quản lý nhân viên, khách hàng, hóa đơn sẽ được hiển thị.
  + **Điều hướng đến các form khác**:
    - Các nút như QLNV\_Click, QLNCC\_Click, QLKH\_Click, QLHDN\_Click, QLDMHH\_Click, QLHDB\_Click sẽ chuyển hướng đến các form tương ứng để quản lý nhân viên, nhà cung cấp, khách hàng, hóa đơn, danh mục hàng hóa, và hóa đơn bán.
    - Các chức năng này là cơ sở để xây dựng hệ thống quản lý cho doanh nghiệp, giúp quản lý các đối tượng và nghiệp vụ.
  + **Chức năng thông báo**:
    - Phương thức LoadThongBao() tải thông báo từ cơ sở dữ liệu, hiển thị thông báo đến nhân viên hoặc quản lý tùy theo vai trò, giúp hệ thống cung cấp các thông báo cần thiết đến người dùng.
  + **Cập nhật giao diện**:
    - UpdateUI() sử dụng thông tin về người dùng để cập nhật giao diện chính, bao gồm hiển thị tên nhân viên và công việc.
  + **Quản lý thông tin nhân viên và khách hàng**:
    - Các nút button3\_Click, button4\_Click, button7\_Click được sử dụng để thêm nhân viên, tìm nhân viên, tìm khách hàng hoặc sản phẩm trong hệ thống.
  + **Thay đổi mật khẩu**:
    - rspass\_Click cho phép người dùng thay đổi mật khẩu khi nhập đúng email, dựa trên mã nhân viên.

### 3.2.3. Lớp QuanLyDanhMucHangHoa

Lớp này quản lý giao diện và chức năng của danh mục hàng hóa trong hệ thống.

**Thuộc tính**

* **TenNV**, **CongViec**, **MaNV**: Lưu trữ thông tin về tên nhân viên, công việc và mã nhân viên của người dùng đăng nhập.

**Chức năng chính**

* **Hiển thị danh mục hàng hóa**
  + **Phương thức loadData()**:
    - Truy vấn cơ sở dữ liệu để lấy thông tin danh mục hàng hóa, kết hợp với các bảng liên quan như Loại Kính, Loại Gọng, Hình Dáng, Chất Liệu, v.v.
    - Hiển thị dữ liệu trong DataGridView với các tiêu đề cột tùy chỉnh.
* **Thêm mới hàng hóa**
  + **Sự kiện Them\_Click**:
    - Điều hướng đến form MainHangHoa và kích hoạt sự kiện cập nhật khi thêm hàng hóa thành công, đồng thời tải lại dữ liệu.
* **Sửa thông tin hàng hóa**
  + **Sự kiện Sua\_Click**:
    - Lấy thông tin hàng hóa từ dòng được chọn trong DataGridView.
    - Truyền thông tin đến form SuaHangHoa để chỉnh sửa và cập nhật dữ liệu sau khi sửa thành công.
* **Xóa hàng hóa**
  + **Sự kiện Xoa\_Click**:
    - Hiển thị hộp thoại xác nhận trước khi xóa.
    - Sử dụng câu lệnh DELETE để xóa dữ liệu hàng hóa trong cơ sở dữ liệu, với kiểm tra trạng thái sử dụng để đảm bảo an toàn dữ liệu.
* **Xuất dữ liệu ra Excel**
  + **Sự kiện btnXuatExcel\_Click**:
    - Lưu dữ liệu danh mục hàng hóa vào file Excel, cho phép người dùng xuất dữ liệu với định dạng thân thiện và dễ sử dụng.
* **Điều hướng trở về trang chủ**
  + **Sự kiện Exit\_Click**:
    - Điều hướng đến form Home, đồng thời giữ nguyên thông tin người dùng để cập nhật giao diện.
* **Báo cáo danh mục hàng hóa**
  + **Sự kiện button1\_Click**:
    - Điều hướng đến form HangHoaReport để hiển thị báo cáo danh mục hàng hóa.
* **Định dạng dữ liệu hiển thị**
  + **Phương thức dataGridView1\_CellFormatting**:
    - Tùy chỉnh hiển thị các cột giá (giá nhập và giá bán) để hiển thị phần nguyên hoặc tối đa 2 chữ số thập phân.

### 3.2.4. Lớp QuanLyHoaDonBan

Lớp này quản lý giao diện người dùng để thực hiện các chức năng quản lý hóa đơn bán. Chức năng chính bao gồm hiển thị, thêm, xóa, xem chi tiết, in hóa đơn, và tạo báo cáo. Các chức năng này hỗ trợ việc quản lý hóa đơn bán một cách hiệu quả và chuyên nghiệp.

**Thuộc tính:**

* **TenNV, CongViec, MaNV:** Lưu trữ thông tin nhân viên đăng nhập để hiển thị và kiểm soát quyền truy cập theo vai trò.

**Chức năng chính:**

**1. Hiển thị dữ liệu:**

* **Phương thức LoadData():**
  + Tải và hiển thị danh sách hóa đơn bán từ cơ sở dữ liệu vào DataGridView.
  + Thực hiện các truy vấn SQL để tính toán và nhóm dữ liệu như tổng số lượng sản phẩm, tổng tiền.

**2. Xem chi tiết hóa đơn:**

* **Phương thức btnXemChiTiet\_Click:**
  + Lấy mã hóa đơn từ dòng được chọn trong DataGridView.
  + Chuyển thông tin mã hóa đơn đến form ChiTietHoaDonBan để hiển thị chi tiết các sản phẩm trong hóa đơn.

**3. Thêm hóa đơn:**

* **Phương thức btnThemHD\_Click:**
  + Mở form HoaDonBan cho phép thêm mới hóa đơn.

**4. Xóa hóa đơn:**

* **Phương thức btnXoaHD\_Click:**
  + Xóa hóa đơn và các chi tiết liên quan trong cơ sở dữ liệu sau khi xác nhận từ người dùng.
  + Sử dụng giao dịch SQL (SqlTransaction) để đảm bảo tính toàn vẹn dữ liệu.

**5. In hóa đơn:**

* **Phương thức btnIn\_Click:**
  + Truy vấn dữ liệu chi tiết hóa đơn và in ra dưới dạng tài liệu.
  + Sử dụng PrintDocument để tạo và hiển thị giao diện in hóa đơn.

**6. Tạo báo cáo:**

* **Phương thức btnbaocao\_Click:**
  + Mở form HoaDonBanReport để hiển thị báo cáo tổng quan về hóa đơn bán, có thể theo nhân viên hoặc thời gian.

**7. Cập nhật giao diện:**

* **Phương thức capnhat:**
  + Tự động làm mới dữ liệu khi form kích hoạt (Active).
* **Định dạng dữ liệu hiển thị:**
  + Sử dụng sự kiện dataGridView1\_CellFormatting để định dạng số tiền trong cột TongTien và ThanhTien.

**8. Quay lại màn hình chính:**

* **Phương thức exit\_Click:**
  + Trở lại form chính (Home) với thông tin nhân viên được truyền lại.

### 3.2.5. Lớp QuanLyHoaDonNhap

Lớp này quản lý giao diện và chức năng liên quan đến hóa đơn nhập, cho phép người dùng xem, thêm, sửa, xóa, và báo cáo dữ liệu về hóa đơn nhập hàng. Giao diện và chức năng được điều chỉnh dựa trên vai trò và thông tin của nhân viên đăng nhập. Cụ thể:

**Thuộc tính**

* TenNV: Lưu trữ tên của nhân viên đăng nhập.
* CongViec: Lưu trữ vai trò hoặc công việc của nhân viên.
* MaNV: Lưu trữ mã nhân viên của người đăng nhập.

**Chức năng chính**

* **Tải và hiển thị dữ liệu hóa đơn nhập**:
  + **Phương thức LoadData()**:
    - Lấy dữ liệu hóa đơn nhập từ cơ sở dữ liệu, bao gồm thông tin ngày nhập, mã nhà cung cấp, loại mặt hàng, tổng số lượng, và tổng tiền.
    - Hiển thị dữ liệu lên DataGridView.
* **Thêm chi tiết hóa đơn nhập**:
  + **Nút button1\_Click**:
    - Điều hướng đến form ThemChiTietHoaDonNhap2 với thông tin nhân viên đăng nhập (TenNV, MaNV).
* **Xóa hóa đơn nhập**:
  + **Nút button3\_Click**:
    - Xác nhận xóa và thực hiện xóa hóa đơn nhập, bao gồm cả chi tiết hóa đơn nhập liên quan.
    - Sử dụng giao dịch SQL để đảm bảo tính toàn vẹn dữ liệu.
* **Xem chi tiết hóa đơn nhập**:
  + **Nút button4\_Click**:
    - Mở form ChiTietHoaDonNhap với mã hóa đơn nhập (SoHDN) được chọn.
* **Định dạng dữ liệu**:
  + **Phương thức dataGridView1\_CellFormatting**:
    - Định dạng cột TongTien và DonGia để hiển thị số thập phân gọn gàng hơn.
* **Xuất báo cáo**:
  + **Nút btnbaocao\_Click**:
    - Điều hướng đến form HoaDonNhapReport để xem và xuất báo cáo hóa đơn nhập.
* **Cập nhật giao diện**:
  + **Phương thức QuanLyHoaDonNhap\_Activated**:
    - Tự động làm mới dữ liệu khi form được kích hoạt.
* **Thoát về giao diện chính**:
  + **Nút exit\_Click**:
    - Quay về form Home và truyền thông tin nhân viên (TenNV, CongViec, MaNV).

### 3.2.6. Lớp QuanLyKhachHang

Lớp này quản lý giao diện và chức năng liên quan đến khách hàng, cho phép người dùng thêm, sửa, xóa, và xem báo cáo thông tin khách hàng. Giao diện và các chức năng được điều chỉnh phù hợp với vai trò và thông tin của nhân viên đăng nhập.

**Thuộc tính**

* TenNV: Lưu trữ tên của nhân viên đăng nhập.
* CongViec: Lưu trữ vai trò hoặc công việc của nhân viên đăng nhập.
* MaNV: Lưu trữ mã nhân viên đăng nhập.

**Chức năng chính**

* **Tải và hiển thị dữ liệu khách hàng**:
  + **Phương thức LoadData()**:
    - Lấy danh sách khách hàng từ cơ sở dữ liệu, bao gồm MaKhach, TenKhach, DiaChi, và DienThoai.
    - Hiển thị dữ liệu trong DataGridView.
* **Thêm khách hàng**:
  + **Nút themKH\_Click**:
    - Điều hướng đến form ThemKH để thêm thông tin khách hàng mới.
* **Sửa thông tin khách hàng**:
  + **Nút suaKH\_Click**:
    - Lấy thông tin khách hàng được chọn từ DataGridView.
    - Mở form SuaKH với các thông tin cần sửa (MaKhach, TenKhach, DiaChi, DienThoai).
* **Xóa khách hàng**:
  + **Nút xoaKH\_Click**:
    - Xác nhận xóa và thực hiện xóa khách hàng khỏi cơ sở dữ liệu.
    - Hiển thị thông báo nếu xóa thành công hoặc xảy ra lỗi.
* **Xuất báo cáo**:
  + **Nút btnbaocao\_Click**:
    - Mở form KhachHangReport để xem và xuất báo cáo danh sách khách hàng.
* **Thoát về giao diện chính**:
  + **Nút exit\_Click**:
    - Quay lại form Home và truyền thông tin đăng nhập (TenNV, CongViec, MaNV).
* **Cập nhật giao diện tự động**:
  + **Phương thức QuanLykhachhnag\_Activated**:
    - Làm mới dữ liệu khách hàng khi form được kích hoạt.

### 3.2.7. Lớp QuanLyNhaCungCap

Lớp này chịu trách nhiệm quản lý giao diện và các chức năng liên quan đến nhà cung cấp. Nó cho phép nhân viên thực hiện các tác vụ như thêm, sửa, xóa và xuất báo cáo thông tin nhà cung cấp.

**Thuộc tính**

* TenNV: Lưu trữ tên nhân viên đang đăng nhập.
* CongViec: Lưu trữ vai trò của nhân viên đăng nhập.
* MaNV: Lưu trữ mã nhân viên đang đăng nhập.

**Chức năng chính**

* **Tải và hiển thị dữ liệu nhà cung cấp**:
  + **Phương thức LoadData()**:
    - Lấy danh sách nhà cung cấp từ cơ sở dữ liệu và hiển thị trên DataGridView.
    - Các thông tin bao gồm: Mã nhà cung cấp, tên, địa chỉ, số điện thoại, và các thuộc tính khác (nếu có).
* **Thêm nhà cung cấp**:
  + **Nút btnThem\_Click**:
    - Mở form NhaCungCap ở chế độ "Thêm".
    - Nếu thông tin được thêm thành công, dữ liệu sẽ được làm mới.
* **Sửa thông tin nhà cung cấp**:
  + **Nút btnSua\_Click**:
    - Lấy thông tin nhà cung cấp từ hàng được chọn trong DataGridView.
    - Mở form NhaCungCap ở chế độ "Sửa", truyền mã nhà cung cấp (MaNCC) vào form.
    - Làm mới dữ liệu sau khi cập nhật thành công.
* **Xóa nhà cung cấp**:
  + **Nút btnXoa\_Click**:
    - Lấy mã nhà cung cấp từ hàng được chọn trong DataGridView.
    - Hiển thị hộp thoại xác nhận xóa.
    - Gọi phương thức XoaNhaCungCap(int maNCC) để thực hiện thao tác xóa trong cơ sở dữ liệu.
* **Xuất báo cáo**:
  + **Nút button2\_Click**:
    - Mở form NhaCungCapReport để xuất báo cáo danh sách nhà cung cấp.
* **Thoát về giao diện chính**:
  + **Nút button1\_Click**:
    - Quay lại form Home với thông tin của nhân viên đăng nhập (TenNV, CongViec, MaNV).

### 3.2.8. Lớp QuanLyNhanVien

Lớp này quản lý các chức năng liên quan đến thông tin nhân viên trong hệ thống. Sau khi người dùng đăng nhập, giao diện này cho phép người dùng thực hiện các thao tác như thêm, sửa, xóa nhân viên, và xuất báo cáo danh sách nhân viên.

**Thuộc tính**

* TenNV: Lưu trữ tên nhân viên đang đăng nhập.
* CongViec: Lưu trữ vai trò của nhân viên đăng nhập.
* MaNV: Lưu trữ mã nhân viên đăng nhập.

**Chức năng chính**

* **Tải và hiển thị dữ liệu nhân viên**:
  + **Phương thức LoadData()**:
    - Kết nối tới cơ sở dữ liệu và lấy danh sách nhân viên.
    - Dữ liệu được hiển thị trên DataGridView, giúp người dùng dễ dàng xem và quản lý thông tin nhân viên.
* **Thêm nhân viên**:
  + **Nút ThemNV\_Click**:
    - Mở form ThemNV để người dùng có thể thêm mới thông tin nhân viên.
    - Form này có thể bao gồm các trường như tên, giới tính, ngày sinh, số điện thoại, địa chỉ, và mã chức vụ.
* **Sửa thông tin nhân viên**:
  + **Nút SuaNV\_Click**:
    - Lấy thông tin từ hàng được chọn trong DataGridView.
    - Mở form SuaNV để người dùng có thể sửa thông tin của nhân viên đã chọn.
* **Xóa nhân viên**:
  + **Nút XoaNV\_Click**:
    - Lấy mã nhân viên từ hàng được chọn trong DataGridView.
    - Hiển thị hộp thoại xác nhận yêu cầu người dùng có muốn xóa nhân viên này hay không.
    - Nếu người dùng xác nhận, thực hiện việc xóa nhân viên khỏi cơ sở dữ liệu.
* **Xuất báo cáo nhân viên**:
  + **Nút btnbaocao\_Click**:
    - Mở form NhanVienReport để xuất báo cáo danh sách nhân viên theo yêu cầu.
* **Thoát về giao diện chính**:
  + **Nút exitNV\_Click**:
    - Quay lại form Home với thông tin của nhân viên đăng nhập (TenNV, CongViec, MaNV).

### 3.2.9. Lớp DoanhThu

Lớp này quản lý các chức năng liên quan đến doanh thu trong hệ thống, cho phép người dùng xem báo cáo doanh thu theo khoảng thời gian và theo tháng, cùng với khả năng xuất báo cáo chi tiết.

**Thuộc tính**

* TenNV: Lưu trữ tên nhân viên đăng nhập.
* CongViec: Lưu trữ vai trò của nhân viên đăng nhập.
* MaNV: Lưu trữ mã nhân viên đăng nhập.

**Chức năng chính**

* **Xem doanh thu theo khoảng thời gian**:
  + **Nút btnxemdoanhthu\_Click**:
    - Lấy dữ liệu từ cơ sở dữ liệu về doanh thu bán, chi phí nhập và lợi nhuận trong khoảng thời gian người dùng chọn (từ ngày TuNgay đến ngày DenNgay).
    - Dữ liệu được hiển thị trên DataGridView với các cột như "Ngày Tháng", "Doanh Thu", "Chi Phí Nhập", và "Lợi Nhuận".
    - Tính tổng doanh thu bán, chi phí nhập, và lợi nhuận trong khoảng thời gian đó và hiển thị trên giao diện.
* **Hiển thị chi tiết doanh thu**:
  + **Nút btnxemdoanhthu\_Click (Tiếp theo)**:
    - Sau khi hiển thị doanh thu theo khoảng thời gian, một form ChiTietDoanhThu sẽ được mở để người dùng có thể xem chi tiết doanh thu cho khoảng thời gian đã chọn.
* **Xem doanh thu theo tháng**:
  + **Nút btnxemdoanhthutheothang\_Click**:
    - Mở form DoanhThuTheoThang để người dùng có thể xem báo cáo doanh thu theo tháng.
* **Xuất báo cáo doanh thu**:
  + **Nút button1\_Click**:
    - Mở form DoanhThuReport và truyền thông tin tên nhân viên (TenNV) để xuất báo cáo doanh thu theo yêu cầu.
* **Thoát về giao diện chính**:
  + **Nút btnexit\_Click**:
    - Quay lại form Home với thông tin nhân viên đăng nhập (TenNV, CongViec, MaNV).

## 3.3. Kết chương

Trong quá trình phát triển ứng dụng quản lý bán hàng, chúng tôi đã xây dựng và hoàn thiện các form chính, bao gồm quản lý hóa đơn, khách hàng, nhà cung cấp, danh mục hàng hóa, và các báo cáo doanh thu. Mỗi form đều được thiết kế với giao diện thân thiện, dễ sử dụng, giúp người dùng thực hiện các thao tác một cách nhanh chóng và hiệu quả.

# 4. Kết luận

## 4.1. Kết quả thu được

**Kết quả thu được**

* **Tính năng đầy đủ**: Ứng dụng đã tích hợp đầy đủ các chức năng cần thiết cho việc quản lý bán hàng, từ việc tạo và cập nhật hóa đơn đến việc theo dõi doanh thu và tồn kho.
* **Giao diện người dùng**: Giao diện được thiết kế thân thiện, dễ sử dụng với các bảng điều khiển rõ ràng, giúp người dùng dễ dàng tìm kiếm và thao tác.
* **Báo cáo chi tiết**: Chức năng báo cáo cho phép người dùng xem và phân tích doanh thu theo thời gian, từ đó đưa ra các quyết định kinh doanh hợp lý.
* **Bảo mật thông tin**: Hệ thống đã xây dựng cơ chế xác thực người dùng, đảm bảo an toàn cho dữ liệu và thông tin nhạy cảm.

## 4.2. Hạn chế và hướng phát triển của đề tài

**Hạn chế**:

* Ứng dụng vẫn còn một số hạn chế về responsive, cần được kiểm tra và sửa chữa để nâng cao độ ổn định.
* Giao diện chưa thực sự đẹp và thân thiện với người sử dụng
* Di chuyển giữa các Form chưa nhanh

**Hướng phát triển**:

* **Cải thiện chức năng báo cáo**: Mở rộng các loại báo cáo, cho phép xuất dữ liệu ra file Excel hoặc PDF để phục vụ cho việc phân tích và trình bày.
* **Tích hợp phân quyền người dùng**: Bổ sung tính năng phân quyền để người quản lý có thể kiểm soát quyền truy cập của từng nhân viên, từ đó tăng cường bảo mật.
* **Tối ưu hóa trải nghiệm người dùng**: Cải thiện giao diện để thích ứng tốt hơn với các thiết bị di động, giúp người dùng có thể truy cập và sử dụng ứng dụng mọi lúc, mọi nơi.
* **Phát triển thêm tính năng**: Nghiên cứu và phát triển thêm các tính năng mới như quản lý tồn kho tự động, thông báo nhắc nhở, và phân tích xu hướng bán hàng để nâng cao hiệu quả quản lý.

# Danh mục tham khảo