

Bộ Giáo Dục Và Đào Tạo  
Trường Đại Học Ngoại Ngữ - Tin Học Thành Phố Hồ Chí Minh  
**Khoa Công Nghệ Thông Tin**



**MÔN HỌC: ĐỒ ÁN PHẦN MỀM**

**ĐỀ TÀI: XÂY DỰNG PHẦN MỀM QUẢN LÝ CÁC ĐẠI LÝ**

**Giáo Viên Hướng Dẫn:** ThS. Nguyễn Thanh Vũ

**Thành Viên :**

1. Trương Tân Nhã – MSSV: 20DH111319
2. Nguyễn Thị Kim Ngân – MSSV: 20DH111085
3. Nguyễn Thị Diễm Khang – MSSV: 20DH112027
4. Nguyễn Thị Diễm Quỳnh – MSSV: 20DH111839
5. Trương Thị Hường – MSSV: 20DH110293

*Tp. Hồ Chí Minh, ngày 27 tháng 05 năm 2023*

**PHIẾU CHẤM ĐIỂM MÔN THI VĂN ĐÁP**

 **Điểm phần trình bày – Điểm hệ 10 – Tỷ lệ điểm chiếm 30%**

	<b>CBCT1</b>	<b>CBCT2</b>
<b>Họ tên CBCT</b>	..... <b>Chữ ký:</b> .....	..... <b>Chữ ký:</b> .....
<b>Điểm</b>	..... <b>Bằng chữ:</b> .....	..... <b>Bằng chữ:</b> .....
<b>Nhận xét</b>		

 **Điểm quá trình – Điểm hệ 10 – Tỷ lệ điểm chiếm 70%**

**Họ tên CBCT:** ..... **Ký tên**

 **Điểm tổng kết:** ..... **Bằng chữ**

### ***Lời cảm ơn***

Trước tiên, nhóm chúng em xin chân thành cảm ơn Khoa Công nghệ thông tin Trường Đại học Ngoại Ngữ - Tin Học Thành Phố Hồ Chí Minh đã tạo điều kiện cho nhóm thực hiện tốt đề tài đồ án môn học này.

Nhóm chúng em xin chân thành cảm ơn cô Lê Thị Minh Nguyên, người đã nhiệt tình hướng dẫn và đóng góp ý kiến giúp cho nhóm hoàn thành đồ án. Đồng thời, nhóm chúng em cũng gửi lời cảm ơn đến các quý thầy cô trong khoa Công nghệ thông tin đã tận tình giảng dạy, trang bị cho nhóm những kiến thức quý báu trong các kì học qua, giúp nhóm có một nền tảng kiến thức vững chắc để hoàn thành đề tài này.

Mặc dù nhóm đã cố gắng hoàn thành đồ án môn học trong phạm vi và khả năng cho phép nhưng vẫn không tránh khỏi những thiếu sót. Nhóm chúng em mong nhận được sự thông cảm và tận tình đóng góp ý kiến của các quý thầy cô để hoàn thiện đồ án hơn nữa, cũng như nâng cao kiến thức bản thân.

Chúng em xin chân thành cảm ơn.

## *Mục lục*

---

<b>CHƯƠNG 1: MÔ TẢ ĐỀ TÀI.....</b>	<b>5</b>
1. Đặc tả hệ thống: .....	9
2. Trang web tham khảo: .....	10
<b>CHƯƠNG 2: SƠ ĐỒ CƠ SỞ DỮ LIỆU .....</b>	<b>11</b>
2.1. Sơ đồ theo mức quan niệm .....	11
2.2. Sơ đồ theo mức Logic.....	12
<b>CHƯƠNG 3: MÀN HÌNH GIAO DIỆN .....</b>	<b>13</b>
3.1. Màn hình Login cho tất cả các phòng nhân viên: .....	13
3.2. Sơ đồ liên kết các trang giao diện Kinh Doanh .....	13
3.2.1. Giao diện theo sơ đồ Kinh Doanh .....	14
3.3. Sơ đồ liên kết các trang giao diện Kế toán .....	24
3.3.1 Giao diện theo sơ đồ Kế toán.....	24
3.4. Sơ đồ liên kết các trang giao diện phòng Kho .....	31
3.4.1 Giao diện theo sơ đồ phòng Kho .....	31
3.5. Sơ đồ liên kết các trang giao diện Giao Hàng.....	43
3.5.1 Giao diện phòng giao hàng: .....	44
3.6. Sơ đồ liên kết các trang giao diện Đại lý .....	56
3.6.1 Giao diện theo sơ đồ Đại lý .....	57
3.7. Sơ đồ liên kết các trang giao diện Quản trị viên .....	63
3.7.1 Giao diện theo sơ đồ Quản trị viên .....	63
<b>CHƯƠNG 4: MÔ TẢ CHI TIẾT ỨNG DỤNG THEO MÔ HÌNH MVC.....</b>	<b>66</b>
4.1. MODELS.....	66
4.1.1. Model Diagram.....	67
4.2. CONTROLLERS .....	77
4.2.1. Module tổng quan .....	77
4.2.2. Chi tiết controller theo phòng ban .....	77
4.3. VIEWS.....	138
<b>CHƯƠNG 5: TỔNG KẾT.....</b>	<b>143</b>
5.1. Kết quả đạt được .....	143
5.2. Kết quả chưa đạt được.....	143
5.3. Hướng phát triển trong tương lai .....	143
<b>TÀI LIỆU THAM KHẢO .....</b>	<b>144</b>

### **Danh mục hình ảnh**

Hình 1: Sơ đồ dữ liệu mức quan niệm .....	11
Hình 2: Sơ đồ dữ liệu mức logic .....	12
Hình 3. Đăng nhập nhân viên .....	13
Hình 4. Trang chủ phòng kinh doanh 1 .....	14
Hình 5. Trang chủ phòng kinh doanh 2 .....	14
Hình 6. Quản lý đại lý .....	15
Hình 7. Thêm mới đại lý .....	15
Hình 8. Thêm loại đại lý .....	15
Hình 9. Cập nhật đại lý .....	16
Hình 10. Xác nhận xóa đại lý.....	16
Hình 11. Quản lý đơn hàng / Danh sách đơn hàng .....	16
Hình 12. Đơn hàng chưa xét duyệt .....	17
Hình 13. Đơn hàng đã xét duyệt .....	17
Hình 14. Thêm mới đơn hàng cho đại lý .....	17
Hình 15. Cập nhật xét duyệt đơn hàng.....	18
Hình 16. Danh sách chi tiết đơn hàng.....	18
Hình 17. Cập nhật thêm sản phẩm vào chi tiết đơn hàng .....	18
Hình 18. Cập nhật lại số lượng hoặc cần đổi sang sản phẩm khác của đơn hàng .....	19
Hình 19. Xóa thông tin sản phẩm trong chi tiết đơn hàng .....	19
Hình 20. Doanh số bán hàng.....	19
Hình 21. Quản lý sản phẩm.....	20
Hình 22. Thêm mới thông tin sản phẩm .....	20
Hình 23. Thêm mới thông tin sản phẩm .....	20
Hình 24. Xóa thông tin sản phẩm .....	21
Hình 25. Phiếu đề nghị xuất nhập kho (Danh sách chưa xét duyệt) .....	21
Hình 26. Danh sách đã xét duyệt phiếu .....	21
Hình 27. Thông tin phiếu đề nghị nhập kho .....	22
Hình 28. Thông tin phiếu đề nghị đã được xét duyệt .....	22
Hình 29. Báo cáo.....	23
Hình 30: Sơ đồ liên kết phòng kế toán.....	24
Hình 31. Giao diện trang chủ kế toán .....	24
Hình 32. Giao diện danh sách đơn hàng .....	25
Hình 33. Chi tiết đơn hàng.....	25
Hình 34. Cập nhật thông tin đơn hàng .....	26
Hình 35. In hóa đơn .....	26
Hình 36. Danh sách đại lý trong hệ thống .....	27
Hình 37. Danh sách đơn hàng đã giao nhưng chưa thanh toán của đại lý .....	27

Hình 38. Lập phiếu công nợ.....	28
Hình 39. Danh sách công nợ đại lý.....	28
Hình 40. Giao diện cập nhật thông tin phiếu công nợ .....	29
Hình 41. Chi tiết công nợ.....	29
Hình 42. Xóa phiếu công nợ đại lý .....	29
Hình 43. Doanh thu đơn hàng.....	30
Hình 44. Thống kê tổng lượng bán sản phẩm.....	30
Hình 45: Sơ đồ liên kết phòng kho .....	31
Hình 46. Danh sách đơn hàng mới cần giao .....	44
Hình 47. In danh sách đơn hàng mới cần giao.....	44
Hình 48. Chi tiết đơn hàng.....	45
Hình 49. Chọn chuyến giao cho đơn hàng.....	45
Hình 50. Danh sách các chuyến giao .....	45
Hình 51. Chi tiết chuyến giao .....	46
Hình 52. In danh sách các chuyến giao.....	46
Hình 53. Thêm chuyến giao.....	47
Hình 54. Chi tiết chuyến giao .....	47
Hình 55. Hóa đơn đơn hàng.....	48
Hình 56. In hóa đơn đơn hàng .....	48
Hình 57. Phiếu xuất kho theo đơn hàng.....	49
Hình 58. In phiếu xuất kho theo đơn hàng.....	49
Hình 59. Cập nhật chuyến giao.....	50
Hình 60. Xóa chuyến giao .....	50
Hình 61. Xóa chuyến giao .....	50
Hình 62. In danh sách đơn giao hàng.....	51
Hình 63. Danh sách chi tiết đơn hàng .....	51
Hình 64. Danh sách đơn chờ giao.....	52
Hình 65. Danh sách đơn hàng đang giao .....	52
Hình 66. Danh sách đơn hàng đã giao .....	52
Hình 67. Danh sách nhân viên giao hàng .....	53
Hình 68. Thêm nhân viên giao hàng.....	53
Hình 69. Cập nhật nhân viên giao.....	54
Hình 70. Xếp chuyến xe cho nhân viên giao hàng .....	54
Hình 71. Xóa nhân viên giao hàng.....	54
Hình 72. Danh sách phương tiện giao hàng.....	55
Hình 73. Thêm phương tiện giao hàng mới .....	55
Hình 74. Cập nhật phương tiện giao hàng .....	55
Hình 75. Xóa phương tiện giao hàng.....	56

Hình 76. Trang đăng nhập .....	57
Hình 77. Trang đăng ký .....	58
Hình 78. Trang chủ danh sách sản phẩm .....	58
Hình 79. Trang cập nhật thông tin đại lý .....	58
Hình 80. Trang đổi mật khẩu tài khoản đại lý .....	59
Hình 81. Trang thông tin sản phẩm .....	59
Hình 82. Trang danh sách đơn hàng .....	59
Hình 83. Trang danh sách chờ giao .....	60
Hình 84. Trang danh sách đang giao .....	60
Hình 85. Trang danh sách đã giao .....	60
Hình 86. Trang xóa đơn hàng .....	61
Hình 87. Trang chi tiết đơn hàng .....	61
Hình 88. Trang giờ hàng .....	61
Hình 89. Trang công nợ .....	62
Hình 90. Trang công nợ .....	62
Hình 91. Trang liên hệ .....	62
Hình 92. Trang chủ admin .....	63
Hình 93. Quản lý nhân viên .....	64
Hình 94. Thêm nhân viên.....	64
Hình 95. Cập nhật thông tin nhân viên .....	65
Hình 96. Thông tin chi tiết nhân viên .....	65
Hình 97. Xóa nhân viên .....	65
Hình 98: Module model .....	66
Hình 99. Sơ đồ Model Diagram sau khi đã được triển khai trên asp.net MVC.....	67
Hình 100. BaoCao.cs .....	68
Hình 101. BienBangKiemKe.cs.....	68
Hình 102. ChiTietBienBang.cs .....	69
Hình 103. ChiTietDonHang.cs .....	69
Hình 104. ChiTietKho.cs: .....	70
Hình 105. ChiTietPhieuNhapXuat.cs .....	70
Hình 106. ChucVu.cs .....	71
Hình 107. ChuyenGiao.cs .....	71
Hình 108. DaiLy.cs .....	72
Hình 109. DonHang.cs.....	72
Hình 110. Kho.cs .....	73
Hình 111. LoaiDL.cs.....	73
Hình 112. NhanVien.cs .....	74
Hình 113. NhanVienGH.cs .....	74

Hình 114. PhieuCongNo.cs.....	75
Hình 115. PhieuNhapXuat.cs.....	75
Hình 116. PhuongTienGH.cs .....	76
Hình 117. SanPham.cs .....	76
Hình 118: Module views.....	138
Hình 119. Module views phòng kinh doanh.....	138
Hình 120. Module views phòng kế toán .....	139
Hình 121. Module views phòng kho hàng .....	140
Hình 122: Module views phòng giao hàng .....	141
Hình 123: Module views quản trị viên .....	141
Hình 124. Module views đại lý (khách hàng).....	142
Hình 125: Module views đăng nhập nhân viên và đại lý.....	142

# CHƯƠNG 1: MÔ TẢ ĐỀ TÀI

## 1. Đặc tả hệ thống:

Hệ thống quản lý các đại lý hướng đến mục tiêu đáp ứng nhu cầu mua hàng của đại lý và nghiệp vụ quản lý cho các nhân viên. Hệ thống gồm 6 actor tham gia : Đại lý, nhân viên Kinh doanh,nhân viên Kế Toán, nhân viên Kho, nhân viên giao hàng và quản trị viên.

- Thiết kế 1 trang web đáp ứng nhu cầu thực hiện dịch vụ mua hàng của Đại lý, gồm các chức năng cơ bản sau:
  1. Trang chủ
  2. Quản lý giỏ hàng
  3. Quản lý người dùng: đăng ký, đăng nhập
  4. Quản lý mục sản phẩm
  5. Quản lý danh sách đơn đặt hàng cá nhân
  6. Xem danh sách công nợ
  7. Quản lý thông tin cá nhân
- Thiết kế 1 trang web cung cấp nghiệp vụ cho nhân viên kinh doanh, bao gồm các chức năng cơ bản sau:
  1. Trang chủ
  2. Quản lý đại lý
  3. Quản lý đơn hàng
  4. Xem doanh thu
  5. Quản lý sản phẩm
  6. Tiếp nhận và xét duyệt phiếu đề nghị nhập/xuất từ kho
  7. Báo cáo
- Thiết kế 1 trang web cung cấp nghiệp vụ cho nhân viên kế toán , bao gồm các chức năng cơ bản sau:
  1. Trang chủ
  2. Theo dõi đơn hàng
  3. Cập nhật tình trạng đơn hàng
  4. Quản lý công nợ

5. Xem danh sách đại lý

6. Xem doanh thu

7. Thống kê sản phẩm

➤ Thiết kế 1 trang web cung cấp nghiệp vụ cho nhân viên kho, bao gồm các chức năng cơ bản sau:

1. Trang chủ

2. Quản lý kho

3. Quản lý hàng hóa

4. Tiếp nhận đơn hàng đã xét duyệt từ đại lý

5. Quản lý phiếu nhập/xuất hàng hóa và phiếu xuất đơn hàng

6. Kiểm kê báo cáo

➤ Thiết kế 1 trang web cung cấp nghiệp vụ cho nhân viên giao hàng, bao gồm các chức năng cơ bản sau:

1. Tiếp nhận đơn hàng chờ giao

2. Quản lý chuyến giao hàng

3. Theo dõi tình trạng giao của đơn hàng

4. Quản lý nhân viên giao hàng

5. Quản lý phương tiện

➤ Thiết kế 1 trang web cung cấp nghiệp vụ cho quản trị viên, bao gồm các chức năng cơ bản sau:

1. Trang chủ

2. Quản lý nhân viên

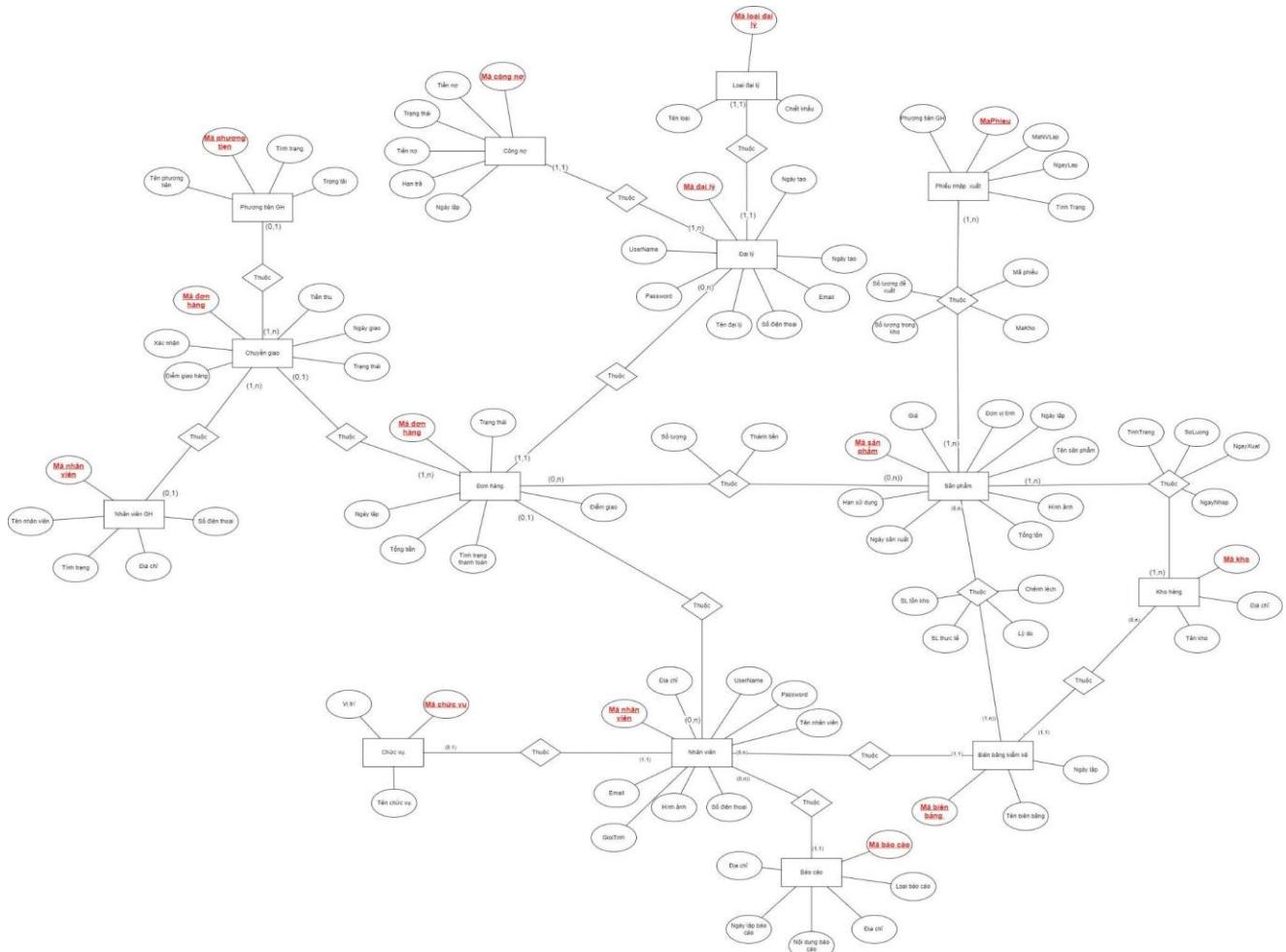
## 2. Trang web tham khảo:

✓ <http://QuanLyDL.somee.com>

✓ <http://www.QuanLyDL.somee.com>

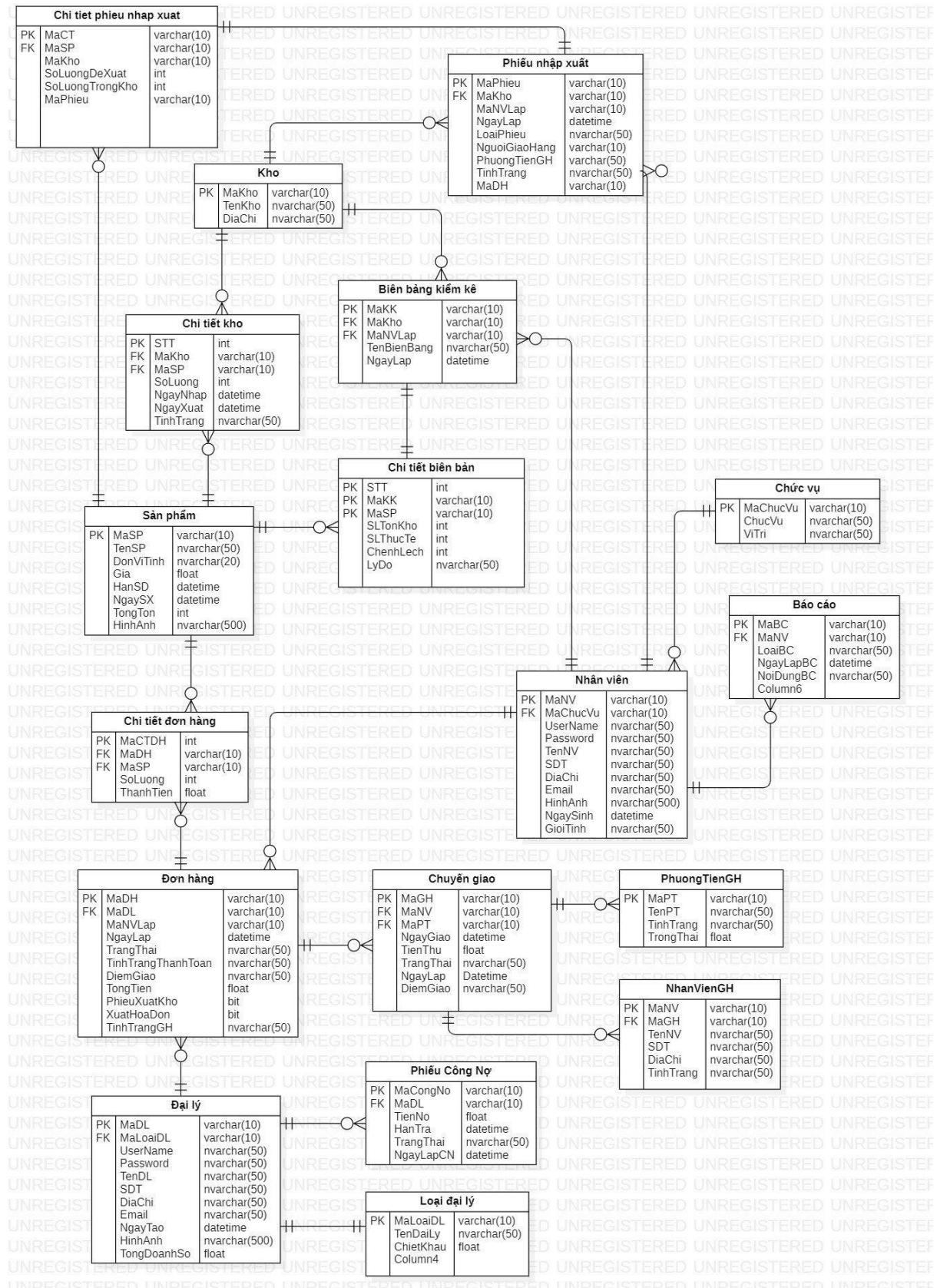
## CHƯƠNG 2: SƠ ĐỒ CƠ SỞ DỮ LIỆU

### **2.1. Sơ đồ theo mức quan niệm**



*Hình 1: Sơ đồ dữ liệu mức quan niệm*

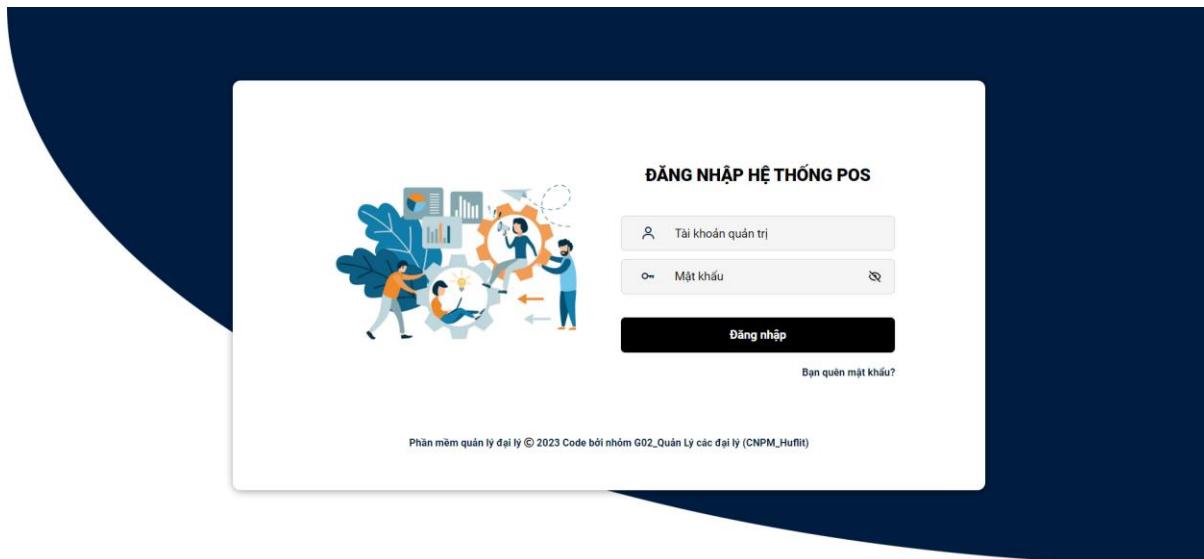
## 2.2. Sơ đồ theo mức Logic



Hình 2: Sơ đồ dữ liệu mức logic

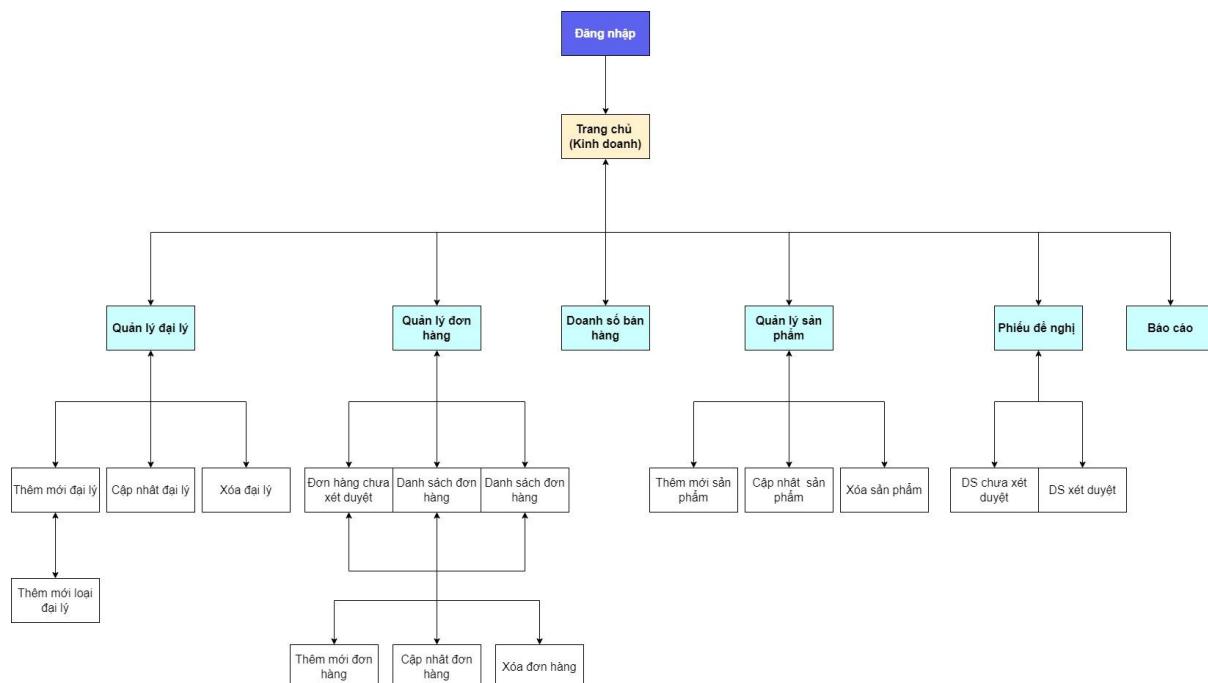
## ***CHƯƠNG 3: MÀN HÌNH GIAO DIỆN***

### **3.1. Màn hình Login cho tất cả các phòng nhân viên:**



*Hình 3. Đăng nhập nhân viên*

### **3.2. Sơ đồ liên kết các trang giao diện Kinh Doanh**



### 3.2.1. Giao diện theo sơ đồ Kinh Doanh

#### 3.2.1.1. Giao diện trang chủ

The screenshot shows the main dashboard of the business management system. On the left sidebar, there are several menu items: Tân Nhã (Welcome back), PHÒNG KINH DOANH (Business Office), Trang chủ (Home), Quản lý đại lý (Manage Agents), Quản lý đơn hàng (Manage Orders), Doanh số bán hàng (Sales Performance), Quản lý sản phẩm (Product Management), Phiếu đề nghị (Request Form), and Báo cáo (Report). The main content area is titled 'Bảng Điều Khiển' (Control Panel) and contains the following sections:

- THÔNG TIN ĐẠI LÝ**: Total number of managed agents: 7 major agents. Responsible agent: Kinh doanh.
- THÔNG TIN SẢN PHẨM**: Total number of products: 12. Current product stock in the system: 12.
- DON HÀNG THÀNH CÔNG**: Total number of successful orders: 1 order. All orders have been checked and settled.
- DON HÀNG ĐANG CHỜ**: Total number of pending orders: 0. Current number of pending orders: 0.
- BÁO CÁO THÁNG NÀY**: Number of reports generated this month: 3.
- PHIẾU ĐỀ NGHỊ XUẤT/ NHẬP KHO**: Total number of bills generated: 1. Includes export/import bills.

Below these sections is a table titled 'Tình trạng đơn hàng' (Order Status) with columns: ID đơn hàng, Mã đại lý, Mã nhân viên lập, Ngày lập, Trạng thái đơn hàng, Địa điểm giao hàng, Tổng tiền, and Trạng thái. One row is shown: DL40565, DL 558, 4/14/2023 11:47:08 AM, Đã xuất đến nhà, Đồng Nai, 94.787.010 vnđ, Đã thanh toán.

Hình 4. Trang chủ phòng kinh doanh 1

This screenshot shows the same main dashboard as above, but the central content area is now displaying a table titled 'Danh sách đại lý mới' (List of new agents) with the following data:

Mã đại lý	Loại đại lý	Tên đại lý	Địa chỉ	Số điện thoại	Địa chỉ email	Ngày tạo
DL150	Đại lý nhỏ	Điểm Kang abc	84a, Phó Đức Chính, Phường 01, Quận Bình Thạnh	01222112222	dq@gmail.com	4/14/2023 11:36:57 AM
DL6730	Đại lý nhỏ	Nguyễn Lý	Đồng Nai	0111111111	ly@gmail.com	4/6/2023 9:04:03 PM
DL225	Đại lý vừa	10.21.11.40	94, Trần Hưng Đạo	123456879	tannhat@gmail.com	12/24/2022 11:03:53 AM
DL390	Đại lý nhỏ	Điểm Kang abc	84a, Phó Đức Chính, Phường 01, Quận Bình Thạnh	01222112222	dq@gmail.com	12/14/2022 12:00:00 AM
DL802	Đại lý lớn	Điểm Quỳnh	Bình Chánh	0908455325	dqq@gmail.com	12/7/2022 12:00:00 AM
DL558	Đại lý nhỏ	Trương Hường	TP.HCM	0377480266	truongthihuong20020110@gmail.com	11/29/2022 12:00:00 AM
DL821	Đại lý nhỏ	Tân Khoa	155a, Phó Đức Chính, Phường 01, Quận Bình Thạnh	048456156656	truongtannha2702@gmail.com	11/25/2022 12:00:00 AM

Hình 5. Trang chủ phòng kinh doanh 2

### 3.2.1.2 Quản lý đại lý

Hình 6. Quản lý đại lý

Hình 7. Thêm mới đại lý

Hình 8. Thêm loại đại lý

*Hình 9. Cập nhật đại lý*
*Hình 10. Xác nhận xóa đại lý*

### 3.2.1.3 Quản lý đơn hàng

Mã đơn hàng	Mã đại lý	Mã nhân viên lập	Ngày lập	Trạng thái đơn hàng	Địa điểm giao hàng	Tổng tiền	Tinh trạng thanh toán	Tinh trạng xuất hóa đơn	Tinh trạng giao hàng	Chức năng
DH0565	DL558		4/14/2023 11:47:08 AM	Đã xét duyệt	Đồng Nai	94,787,010 vnđ	Đã thanh toán	True	True	Chờ giao

*Hình 11. Quản lý đơn hàng / Danh sách đơn hàng*

The screenshot shows the 'Danh Sách Đơn Hàng' (Order List) page. The left sidebar includes sections for 'PHÒNG KINH DOANH', 'Trang chủ', 'Quản lý đại lý', 'Quản lý đơn hàng', 'Doanh số bán hàng', and 'Quản lý sản phẩm'. The main content area displays a table titled 'Danh Sách Đơn Hàng' with columns: Mã đơn hàng, Mã đại lý, Mã nhân viên lập, Ngày lập, Trạng thái đơn hàng, Địa điểm giao hàng, Tổng tiền, Tình trạng thanh toán, Tình trạng xuất kho, Tình trạng xuất hóa đơn, Tình trạng giao hàng, and Chức năng. A search bar labeled 'Tim kiếm:' is at the top right. Below the table, it says 'Không có kết quả' and 'Hiện 0 đến 0 của 0 danh mục'. Buttons for 'Thêm mới đơn hàng', 'Làm mới danh sách', and 'Xuất Excel' are at the bottom.

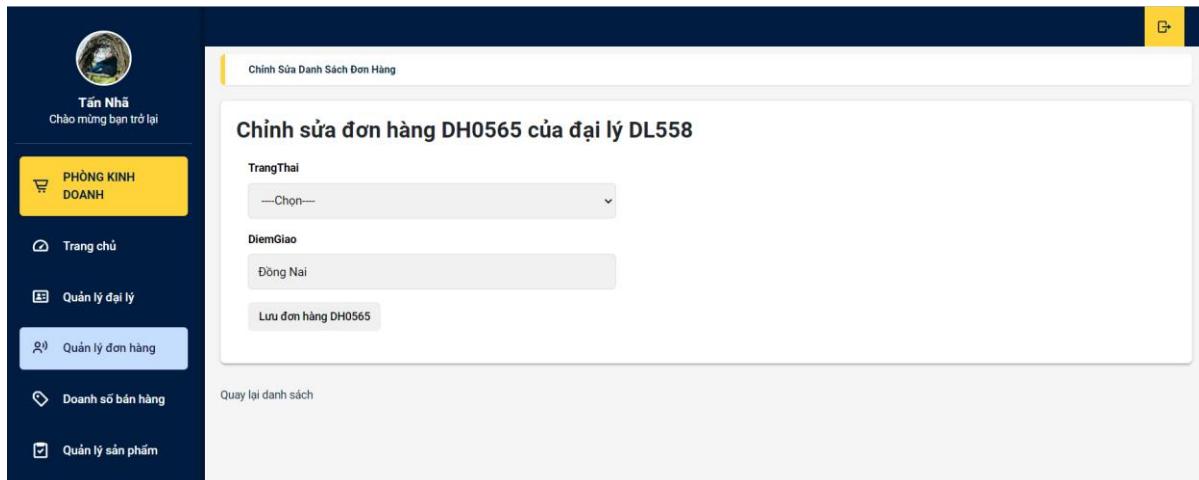
*Hình 12. Đơn hàng chưa xét duyệt*

The screenshot shows the same 'Danh Sách Đơn Hàng' (Order List) page, but now with one row of data. The order details are: DH0565, DL558, null, 4/14/2023 11:47:08 AM, Đã xét duyệt, Đồng Nai, 94.787,010 vnđ, Đã thanh toán, True, True, Chờ giao. The 'Chức năng' column contains three icons: a green checkmark, a red square, and an orange circle. The footer shows 'Hiện 1 đến 1 của 1 danh mục' and navigation buttons 'Lùi 1 Tiếp'.

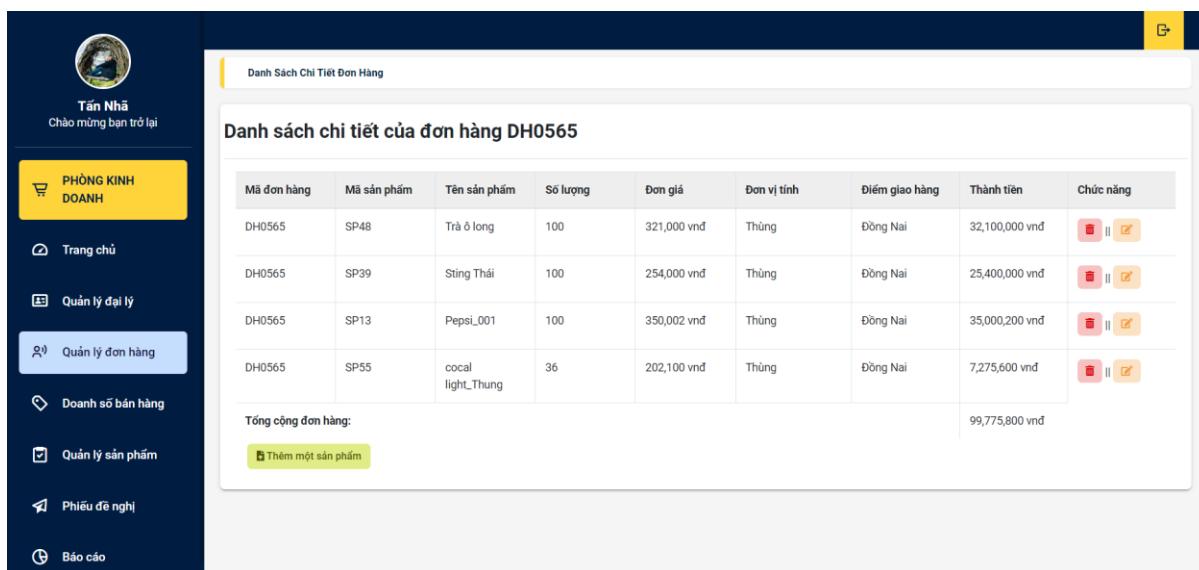
*Hình 13. Đơn hàng đã xét duyệt*

The screenshot shows a modal dialog box titled 'Thêm mới đơn hàng cho đại lý' (Add new order for agent). It has a dropdown 'Chọn mã đại lý' set to 'DL225 - LDL02 - 10.21.11.40'. Below it is a table with columns: Mã đơn hàng, Mã đại lý, and Mã lập. It shows one row: DH0565, DL558. The main area contains a section 'THÔNG TIN ĐƠN HÀNG' with fields: Tên sản phẩm (SP15 - Coca cola - SL: 90C), Số lượng (500), Thành tiền (211.500.000 đ). A note 'Tổng tiền: 211.500.000 đ' is displayed in red. Buttons 'Đơn hàng hợp lệ' (Valid order), 'Thêm SP' (Add item), 'Xóa SP' (Delete item), 'Thêm mới đơn hàng' (Add new order), and 'Hủy bỏ' (Cancel) are at the bottom. The background shows the same 'Danh Sách Đơn Hàng' page as in Figure 12.

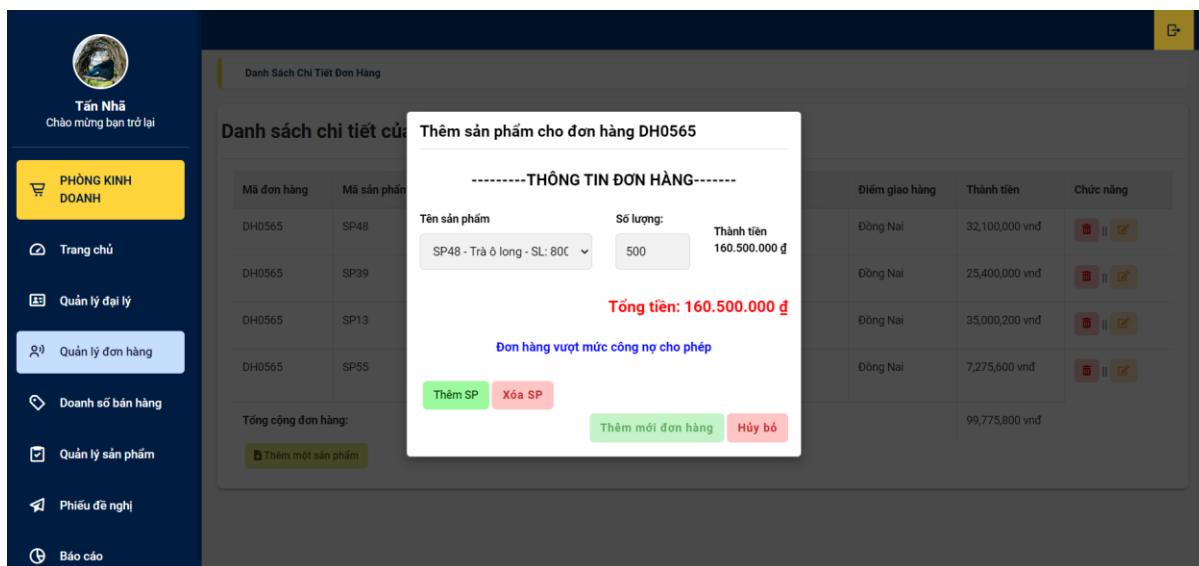
*Hình 14. Thêm mới đơn hàng cho đại lý*



Hình 15. Cập nhật xét duyệt đơn hàng



Hình 16. Danh sách chi tiết đơn hàng



Hình 17. Cập nhật thêm sản phẩm vào chi tiết đơn hàng

The screenshot shows a user interface for managing sales orders. On the left is a sidebar with a logo, the name 'Tân Nhã', a greeting 'Chào mừng bạn trở lại', and a yellow button labeled 'PHÒNG KINH DOANH'. Below these are links: 'Trang chủ', 'Quản lý đại lý', 'Quản lý đơn hàng' (highlighted in blue), 'Doanh số bán hàng', 'Quản lý sản phẩm', and 'Phiếu đề nghị'. The main content area has a title 'Chỉnh sửa chi tiết đơn hàng DH0565 của đại lý DL558' and a sub-section 'ChiTietDonHang'. It contains fields for 'MASP' (set to 'Pepsi\_001'), 'SoLuong' (set to '100'), and a 'Lưu chi tiết đơn hàng DH0565' button. At the bottom is a link 'Quay lại danh sách đơn hàng'.

Hình 18. Cập nhật lại số lượng hoặc càn đổi sang sản phẩm khác của đơn hàng

The screenshot shows a confirmation dialog box. The title is 'Bạn có muốn xóa sản phẩm này SP48'. Inside, there are fields for 'SoLuong' (50), 'ThanhTien' (16050000), and 'TenSP' (Trà ô long). Below these are buttons 'Xóa sản phẩm này' and 'Quay lại danh sách đơn hàng'.

Hình 19. Xóa thông tin sản phẩm trong chi tiết đơn hàng

### 3.1.1.4 Doanh số bán hàng

The screenshot shows a table titled 'Danh sách hóa đơn' (Sales Report). The table has columns: Mã đơn hàng, Mã đại lý, Mã nhân viên lập, Ngày lập, Điểm giao hàng, and Tổng tiền. One row is shown with values: DH0565, DL558, NV01, 4/14/2023 11:47:08 AM, Đồng Nai, and 5,137,627 vnđ. A summary row at the bottom shows 'Tổng cộng hóa đơn:' and '5,137,627 vnđ'.

Hình 20. Doanh số bán hàng

### 3.2.1.5 Quản lý sản phẩm

The screenshot shows a web-based application for managing products. On the left is a sidebar with a user profile picture, the name 'Tấn Nhã', and a greeting 'Chào mừng bạn trở lại'. Below this are several menu items: 'PHÒNG KINH DOANH' (highlighted in yellow), 'Trang chủ', 'Quản lý đại lý', 'Quản lý đơn hàng', 'Doanh số bán hàng', 'Quản lý sản phẩm' (highlighted in blue), 'Phiếu đề nghị', and 'Báo cáo'. The main content area is titled 'Quản lý sản phẩm' and contains a table with the following data:

Mã sản phẩm	Tên sản phẩm	Hình ảnh	Giá	Đơn vị tính	Ngày sản xuất	Hạn sử dụng	Tổng tồn	Chức năng
SP13	Pepsi_001		350.000 vnd	Thùng	4/11/2023 12:00:00 AM	4/22/2023 12:00:00 AM	840	<button>Chỉnh sửa</button>    <button>Xóa sản phẩm</button>
SP15	Coca cola		423.000 vnd	Thùng	4/5/2023 12:00:00 AM	4/12/2023 12:00:00 AM	450	<button>Chỉnh sửa</button>    <button>Xóa sản phẩm</button>
SP39	Sting Thái		254.000 vnd	Thùng	4/18/2023 12:00:00 AM	4/21/2023 12:00:00 AM	1445	<button>Chỉnh sửa</button>    <button>Xóa sản phẩm</button>

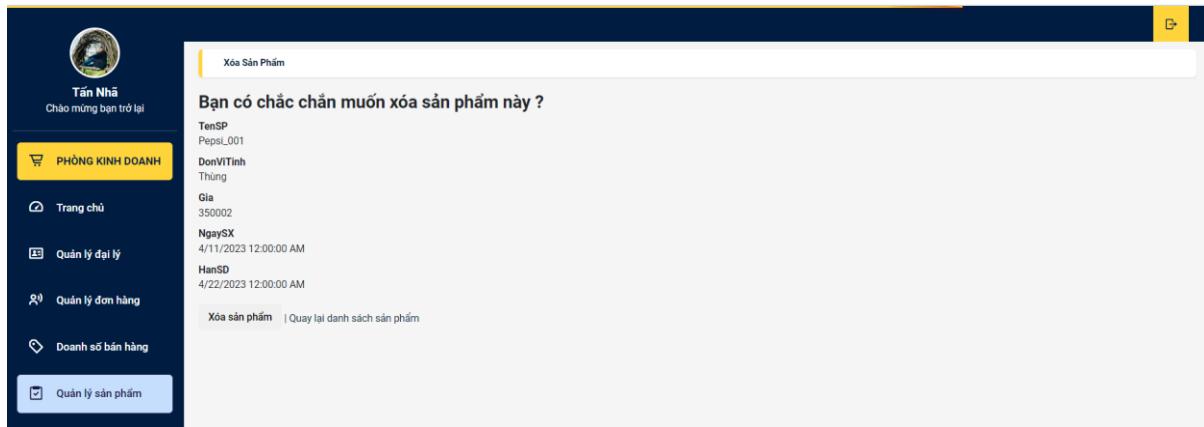
Hình 21. Quản lý sản phẩm

The screenshot shows the 'Thêm mới sản phẩm' (Create new product) screen. The sidebar on the left is identical to the one in Figure 21. The main form has the title 'Tạo mới sản phẩm' and fields for 'Ảnh sản phẩm' (with a 'Chọn ảnh' button), 'Tên sản phẩm' (text input), 'Đơn vị tính' (dropdown menu set to 'Lốc'), 'Giá sản phẩm' (text input), 'Ngày sản xuất:' (date input), and 'Hạn sử dụng:' (date input). At the bottom are 'Thêm mới sản phẩm' and 'Hủy bỏ' buttons.

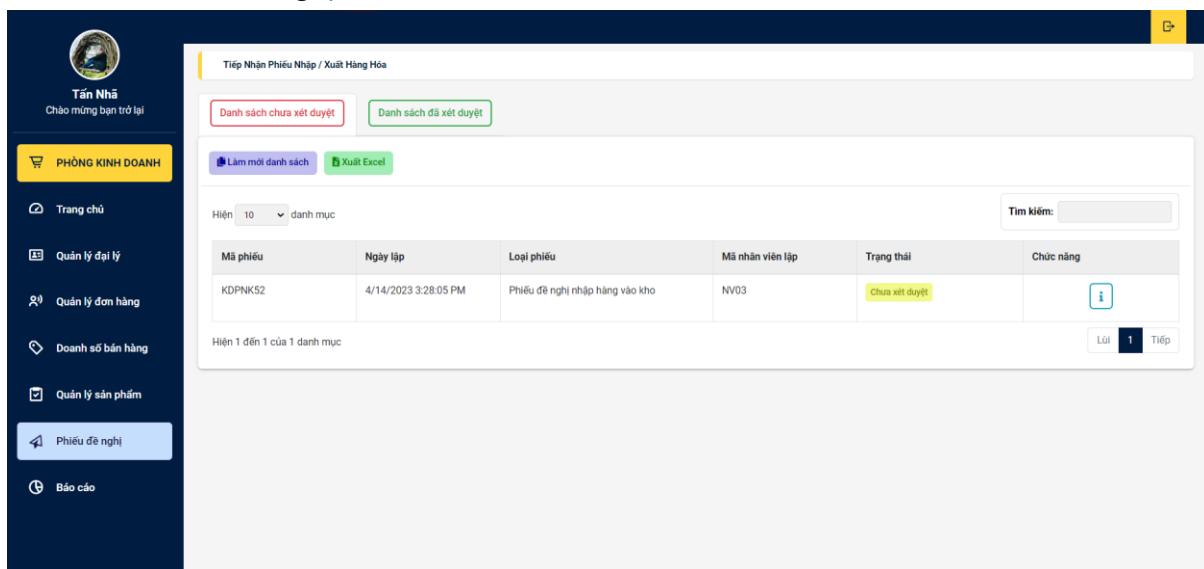
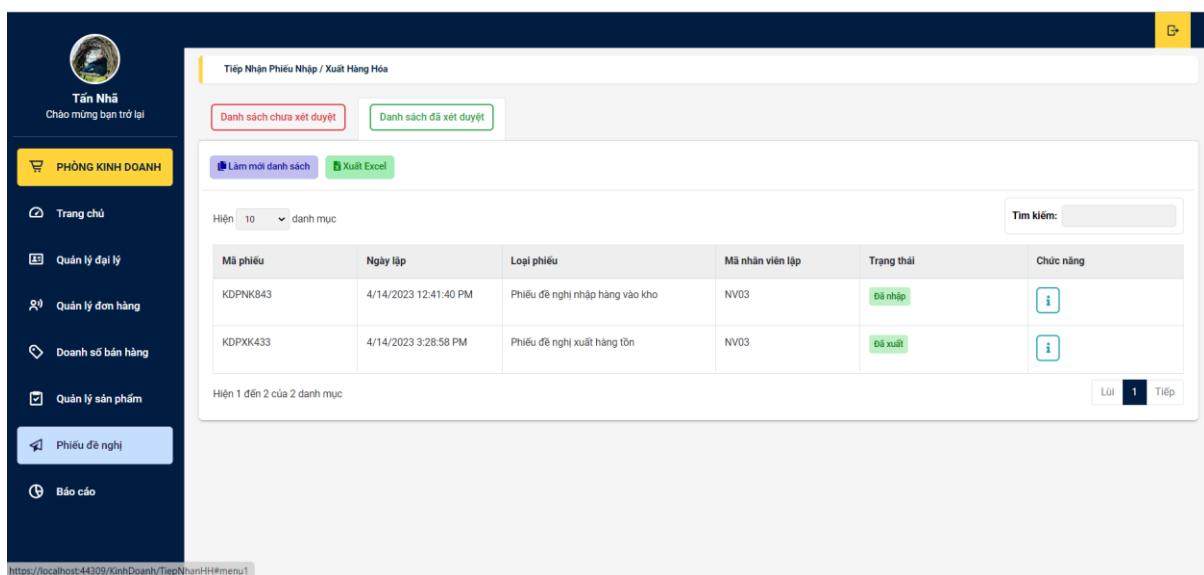
Hình 22. Thêm mới thông tin sản phẩm

The screenshot shows the 'Chỉnh sửa sản phẩm SP13' (Edit product SP13) screen. The sidebar is the same as before. The main form displays the product details: 'Tên SP' (Pepsi\_001), 'Đơn vị tính' (Thùng), 'Giá sản phẩm' (350002), 'Ngày sản xuất:' (4/11/2023 12:00:00 AM), and 'Hạn sử dụng:' (4/22/2023 12:00:00 AM). It also shows a placeholder image for a Pepsi can and a 'Chọn ảnh' button. At the bottom are 'Quay lại danh sách sản phẩm' and 'Lưu thông tin sản phẩm' buttons.

Hình 23. Thêm mới thông tin sản phẩm

*Hình 24. Xóa thông tin sản phẩm*

### 3.2.1.6 Phiếu đề nghị

*Hình 25. Phiếu đề nghị xuất nhập kho (Danh sách chưa xét duyệt)**Hình 26. Danh sách đã xét duyệt phiếu*

The screenshot shows the 'Tấn Nhã' travel agency management system interface. On the left is a sidebar with navigation links: Trang chủ, Quản lý đại lý, Quản lý đơn hàng, Quản lý sản phẩm, Phiếu đề nghị (highlighted in blue), and Báo cáo. The main content area is titled 'Chi Tiết Phiếu Đề Nghị Nhập Kho'. It displays a purchase requisition form with the following details:

**PHIẾU ĐỀ NGHỊ NHẬP KHO**

Ngày: 4/14/2023 3:28:05 PM  
Mã số: KDPNK52

- Kinh gửi: Trưởng phòng kinh doanh
- Họ tên người làm phiếu: Nguyễn Thị Diễm Khang
- Sản phẩm cần nhập kho:

Mã Kho	Mã sản phẩm	Tên sản phẩm	Giá tiền	Đơn vị tính	Trong kho		Cần nhập	
					Số lượng	Thành tiền	Số lượng	Thành tiền
K0444	SP13	Pepsi_001	350002	Thùng	50	17,500,100 VND	463	162,050,926 VND
K0444	SP15	Coca cola	423000	Thùng	50	21,150,000 VND	463	195,849,000 VND

Thủ kho  
(ký và ghi rõ họ tên)  
Nguyễn Thị Diễm Khang

*Hình 27. Thông tin phiếu đề nghị nhập kho*

The screenshot shows the same 'Tấn Nhã' travel agency management system interface as Figure 27. The main content area is titled 'Chi Tiết Phiếu Đề Nghị Nhập Kho' and displays a purchase requisition form with the following details:

**PHIẾU ĐỀ NGHỊ NHẬP KHO**

Ngày: 4/14/2023 12:41:40 PM  
Mã số: KDPNK843

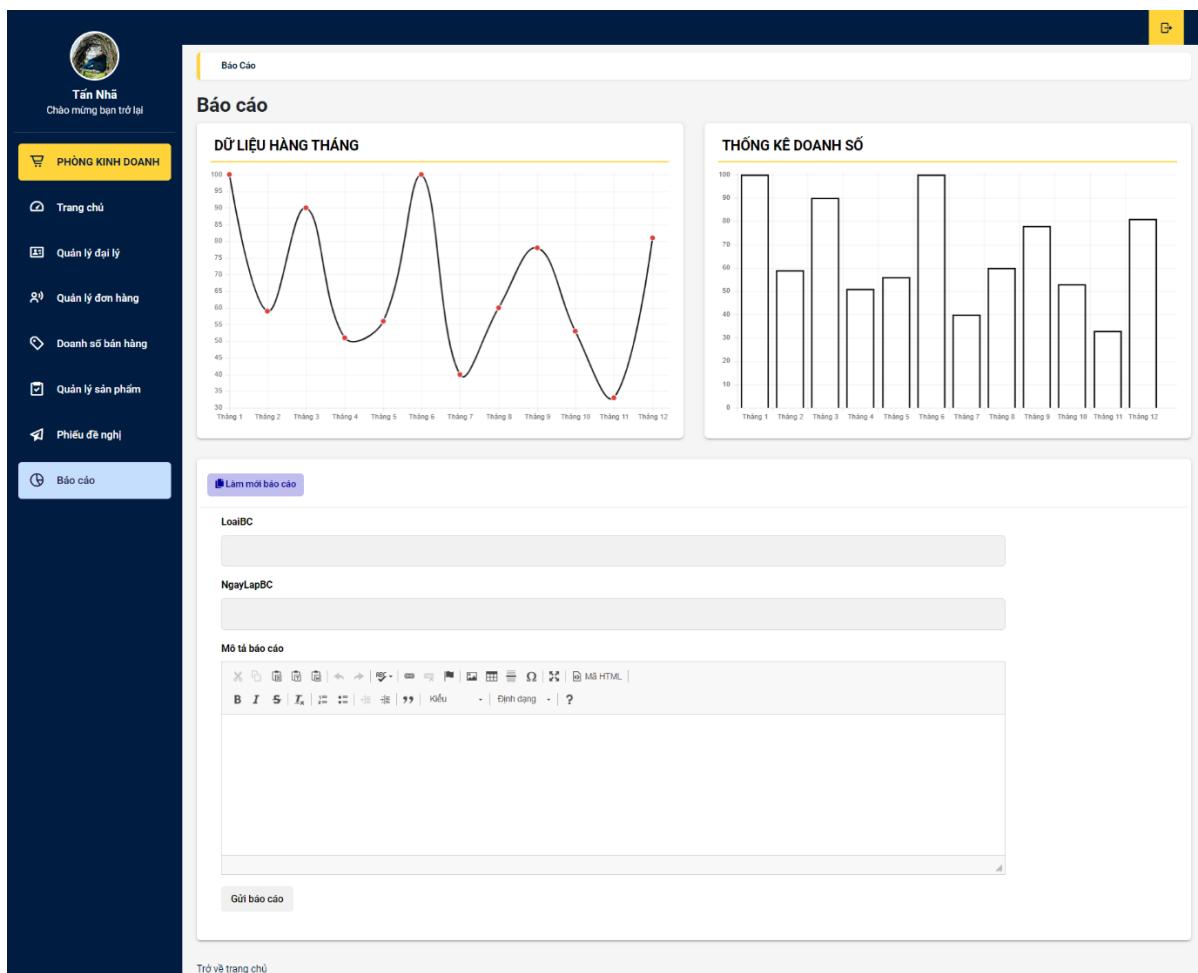
- Kinh gửi: Trưởng phòng kinh doanh
- Họ tên người làm phiếu: Nguyễn Thị Diễm Khang
- Sản phẩm cần nhập kho:

Mã Kho	Mã sản phẩm	Tên sản phẩm	Giá tiền	Đơn vị tính	Trong kho		Cần nhập	
					Số lượng	Thành tiền	Số lượng	Thành tiền
K0444	SP13	Pepsi_001	350002	Thùng	100	35,000,200 VND	413	144,550,826 VND
K080	SP55	cocal light_Thùng	202100	Thùng	90	18,189,000 VND	423	85,488,300 VND
K080	SP39	Sting Thái	254000	Thùng	20	5,080,000 VND	493	125,222,000 VND

Thủ kho  
(ký và ghi rõ họ tên)  
Nguyễn Thị Diễm Khang

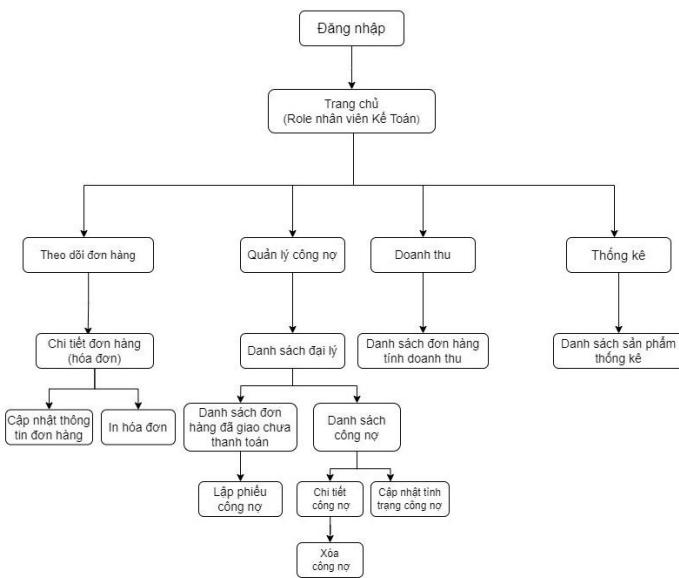
*Hình 28. Thông tin phiếu đề nghị đã được xét duyệt*

### 3.2.1.7 Báo cáo



Hình 29. Báo cáo

### 3.3. Sơ đồ liên kết các trang giao diện Kế toán



Hình 30: Sơ đồ liên kết phòng kế toán

#### 3.3.1 Giao diện theo sơ đồ Kế toán

##### 3.3.1.1 Trang chủ kế toán

Trang Chủ

Thứ Năm, 25/05/2023 - 20 giờ 57 phút 57 giây

ĐƠN HÀNG ĐẠI LÝ		TÌNH TRẠNG ĐẠI LÝ		PHIẾU CÔNG NỢ ĐẠI LÝ		DOANH THU	
Mã CN	Đại lý	Tình trạng	Số lượng	Mã PN	Tình trạng	Giá trị	Thời gian
CN3	Trương Hường-DL558	Chưa thanh toán	1	PN01	Công nợ gần đến hạn trả	144,523,900 vnđ	Tháng

Danh sách công nợ sắp tới hạn

Mã CN	Đại lý	Loại đại lý	Trạng thái	Ngày lập	Ngày trả	Thành tiền	
CN3	Trương Hường-DL558	LDL02	Chưa thanh toán	30/4/2023	25/5/2023	382,005,970 vnđ	Xem

Hình 31. Giao diện trang chủ kế toán

### 3.3.1.2 Theo dõi danh sách đơn hàng

Mã DH	Đại lý	Điểm đón giao	Tình trạng thanh toán	Ngày lập đơn	Tổng tiền	Tình trạng giao hàng	Hóa đơn	Chức năng
DH1653	Trương Hướng	132 Lê Đại Hành, Q10, TP HCM	Đã thanh toán	30/4/2023	4,479,300 vnd	<span style="background-color: green; color: white;">Đã giao</span>	<span style="background-color: green; color: white;">Đã xuất HD</span>	<span style="color: blue;">T</span>
DH1827	Trương Hướng	Đồng Nai	Chưa thanh toán	25/5/2023	210,780,000 vnd	<span style="background-color: green; color: white;">Đã giao</span>	<span style="background-color: yellow; color: black;">Chưa xuất HD</span>	<span style="color: blue;">T</span>
DH2416	Trương Hướng	Sư vạn hạnh q10	Đã thanh toán	30/4/2023	8,613,000 vnd	<span style="background-color: green; color: white;">Đã giao</span>	<span style="background-color: green; color: white;">Đã xuất HD</span>	<span style="color: blue;">T</span>
DH385	Nguyễn Lý	Đồng Nai	Đang nợ	11/4/2023	94,762,500 vnd	<span style="background-color: green; color: white;">Đã giao</span>	<span style="background-color: yellow; color: black;">Chưa xuất HD</span>	<span style="color: blue;">T</span>
DH4042	Trương Hướng	90a, Phố Đức Chính, Phường 01, Quận Bình Thạnh	Đã thanh toán	2/4/2023	9,648,000 vnd	<span style="background-color: green; color: white;">Đã giao</span>	<span style="background-color: yellow; color: black;">Chưa xuất HD</span>	<span style="color: blue;">T</span>
DH4342	Trương Hướng	132 Lê Đại Hành, Q10, TP HCM	Đã thanh toán	1/4/2023	450,000 vnd	<span style="background-color: green; color: white;">Đã giao</span>	<span style="background-color: green; color: white;">Đã xuất HD</span>	<span style="color: blue;">T</span>
DH4353	Trương Hướng	132 Lê Đại Hành, Q10, TP HCM	Đã thanh toán	2/4/2023	2,150,000 vnd	<span style="background-color: green; color: white;">Đã giao</span>	<span style="background-color: yellow; color: black;">Chưa xuất HD</span>	<span style="color: blue;">T</span>
DH5407	Nguyễn Lý	Đồng Nai	Đã thanh toán	6/4/2023	102,700,000 vnd	<span style="background-color: green; color: white;">Đã giao</span>	<span style="background-color: green; color: white;">Đã xuất HD</span>	<span style="color: blue;">T</span>
DH6666	Trương Hướng	132 Lê Đại Hành, Q10, TP HCM	Đã thanh toán	3/4/2023	3,210,000 vnd	<span style="background-color: green; color: white;">Đã giao</span>	<span style="background-color: green; color: white;">Đã xuất HD</span>	<span style="color: blue;">T</span>
DH6821	Trương Hướng	Sư vạn hạnh q10	Đã thanh toán	1/4/2023	5,636,070 vnd	<span style="background-color: green; color: white;">Đã giao</span>	<span style="background-color: green; color: white;">Đã xuất HD</span>	<span style="color: blue;">T</span>

Hiện 1 đến 10 của 17 danh mục

Lùi 1 2 Tiếp

Hình 32. Giao diện danh sách đơn hàng

### 3.3.1.3 Chi tiết đơn hàng (hóa đơn)

Mã CTDH	Mã SP	Tên sản phẩm	Số lượng	Thành tiền
78	SP48	Trà ô long	10	3,210,000 vnd
79	SP87	Sữa hạt	10	2,150,000 vnd
80	SP66	monster	10	4,210,000 vnd

Tổng tiền: 9,570,000 vnd

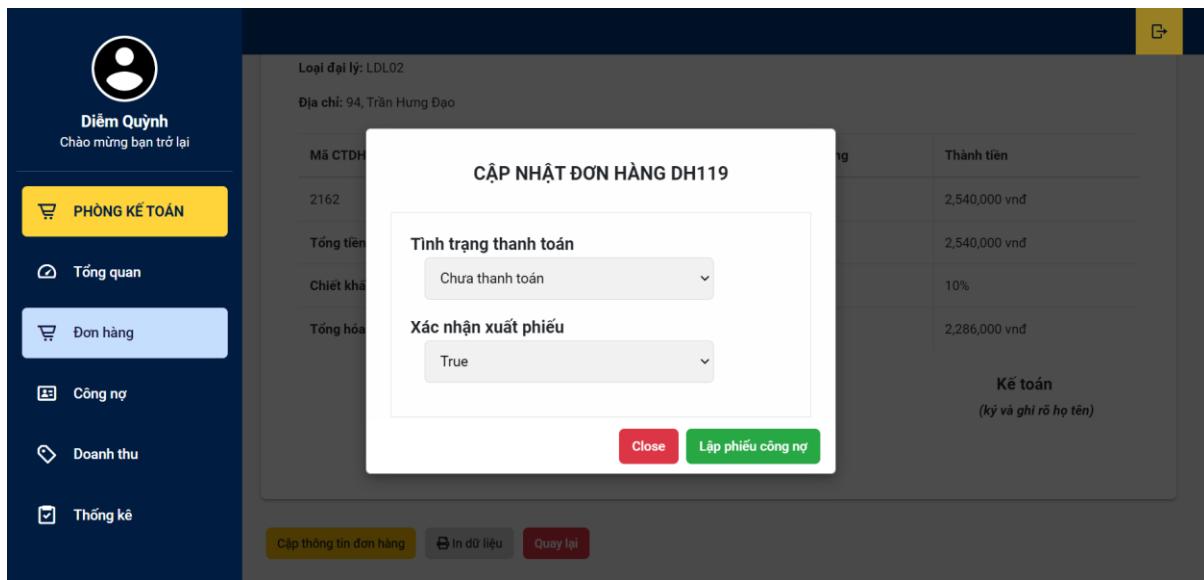
Chiết khấu: 10%

Tổng hóa đơn: 8,613,000 vnd

Kế toán  
(ký và ghi rõ họ tên)

Hình 33. Chi tiết đơn hàng

### 3.3.1.4 Cập nhật thông tin đơn hàng



Hình 34. Cập nhật thông tin đơn hàng

### 3.3.1.5 In hóa đơn



Hình 35. In hóa đơn

### 3.3.1.6 Danh sách đại lý

Danh Sách Đại Lý

Thứ Năm, 25/05/2023 - 22 giờ 57 phút 57 giây

Hiển thị 10 danh mục

Tìm kiếm:

Mã DL	Đại lý	Tổng công nợ	Trạng thái	Tổng doanh số	Chức năng
DL225	10.21.11.40 Đại lý nhô LDL03	00 vnd	Bình thường	00 vnd	<input checked="" type="checkbox"/> <input type="checkbox"/>
DL390	Diễn Kang abc Đại lý nhô LDL03	49,958,950 vnd	Bình thường	00 vnd	<input checked="" type="checkbox"/> <input checked="" type="checkbox"/> 3
DL558	Trương Hướng Đại lý nhô LDL03	00 vnd	Bình thường	233,520,300 vnd	<input checked="" type="checkbox"/> <input checked="" type="checkbox"/> 2
DL6730	Nguyễn Lý Đại lý nhô LDL03	00 vnd	Bình thường	102,700,000 vnd	<input checked="" type="checkbox"/> <input checked="" type="checkbox"/>
DL802	Diễm Quỳnh Đại lý nhô LDL03	00 vnd	Bình thường	00 vnd	<input checked="" type="checkbox"/> <input checked="" type="checkbox"/>
DL821	Tấn Khoa Đại lý nhô LDL03	00 vnd	Bình thường	00 vnd	<input checked="" type="checkbox"/> <input checked="" type="checkbox"/>

Hiện 1 đến 6 của 6 danh mục

Lùi 1 Tiếp

Hình 36. Danh sách đại lý trong hệ thống

### 3.3.1.7. Danh sách đơn hàng đã giao và chưa thanh toán của đại lý

Danh Sách Đơn Hàng Chưa Thanh Toán

Thứ Năm, 25/05/2023 - 21 giờ 25 phút 39 giây

THÔNG TIN ĐẠI LÝ

Mã đại lý: DL558- LDL02

Tên đại lý: Trương Hướng

Địa chỉ: Đồng Nai

Liên lạc: 0377480266 - truongthihuong20020110@gmail.com

THÔNG TIN ĐƠN HÀNG

All | Tim theo tháng

Mã DH	Ngày lập	Điểm giao hàng	Tổng tiền
DH4342	1/4/2023	132 Lê Đại Hành, Q10, TP HCM	450,000 vnd
DH4353	2/4/2023	132 Lê Đại Hành, Q10, TP HCM	2,150,000 vnd

Tổng cộng hóa đơn: 2,600,000 vnd

Thêm phiếu công nợ | In dữ liệu | Quay lại danh sách

Hình 37. Danh sách đơn hàng đã giao nhưng chưa thanh toán của đại lý

### 3.3.1.8. Lập phiếu công nợ

Danh Sách Đơn Hàng Chưa Thanh Toán

Thứ Năm, 25/05/2023 - 21 giờ 26 phút 27 giây

Đơn hàng	Tổng tiền
Đại Hành, Q10, TP.HCM	450,000 vnđ
Đại Hành, Q10, TP.HCM	2,150,000 vnđ
	2,600,000 vnđ

Hình 38. Lập phiếu công nợ

### 3.3.1.9. Danh sách công nợ đại lý

Danh Sách Công Nợ

Thứ Năm, 25/05/2023 - 21 giờ 27 phút 20 giây

Mã CN	Tên đại lý	Mã Đại lý	Tổng nợ	Ngày tạo phiếu	Hạn trả	Tình trạng	Thanh toán	Chức năng
CN5	Diễm Kang abc	DL390	1,110,000 vnđ	30/4/2023	26/6/2023	Còn hạn	Chưa thanh toán	
CN6	Diễm Kang abc	DL390	1,110,000 vnđ	30/4/2023	29/6/2023	Còn hạn	Chưa thanh toán	
CN7	Diễm Kang abc	DL390	47,738,950 vnđ	31/5/2023	30/6/2023	Còn hạn	Chưa thanh toán	

Hiện 1 đến 3 của 3 danh mục

Hình 39. Danh sách công nợ đại lý

### 3.3.1.10. Cập nhật thông tin phiếu công nợ

Danh Sách Công Nợ Chưa Thanh Toán

Thứ Năm, 25/05/2023 - 21 giờ 35 phút 05 giây

**PHIẾU CÔNG NỢ**

Ngày: 30/4/2023

Mã công nợ: CNS

Tên đại lý: Diêm Kang abc - LDL03

Địa chỉ: 84a, Phó Đức Chính, Phường 01, Quận Bình Thạnh

Phiếu công nợ tháng: 4

Mã đơn hàng	Ngày lập	Địa điểm giao hàng	Thành tiền
Tổng cộng tiền nợ:			00 vnd

**CẬP NHẬT THÔNG TIN PHIẾU CÔNG NỢ**

Hạn trả: 05/26/2023

Trạng thái thanh toán: Chưa thanh toán

**Cập nhật** **Quay lại danh sách**

Hình 40. Giao diện cập nhật thông tin phiếu công nợ

### 3.3.1.11 Chi tiết phiếu công nợ

Xuất Phiếu Công Nợ

Thứ Năm, 25/05/2023 - 21 giờ 33 phút 30 giây

**PHIẾU CÔNG NỢ**

Ngày: 4/30/2023 12:00:00 AM

Mã công nợ: CNS

Tên đại lý: Diêm Kang abc

Loại đại lý: LDL03

Địa chỉ: 84a, Phó Đức Chính, Phường 01, Quận Bình Thạnh

Phiếu công nợ tháng: 4

Hạn trả: 5/26/2023 12:00:00 AM

Thành tiền: 1,110,000 vnd

Mã đơn hàng	Ngày lập	Địa điểm giao hàng	Thành tiền
DH385	4/11/2023 3:19:40 AM	Đồng Nai	94,762,500 vnd
Tổng cộng tiền nợ:			94,762,500 vnd

**In dữ liệu** **Xóa phiếu công nợ** **Quay lại danh sách** **Gửi Mail**

Hình 41. Chi tiết công nợ

### 3.3.1.12 Xóa phiếu công nợ

Xuất Phiếu Công Nợ

Thứ Năm, 25/05/2023 - 21 giờ 35 phút 38 giây

**XÁC NHẬN XÓA PHIẾU CÔNG NỢ**

Bạn muốn xóa phiếu công nợ CNS của Diêm Kang abc - DL390 ?!

Tổng công nợ: 1,110,000 vnd

**Close** **Xóa**

Tên đại lý: Diêm Kang abc

Loại đại lý: LDL03

Địa chỉ: 84a, Phó Đức Chính, Phường 01, Quận Bình Thạnh

Phiếu công nợ tháng: 4

Hạn trả: 5/26/2023 12:00:00 AM

Thành tiền: 1,110,000 vnd

Mã đơn hàng	Ngày lập	Địa điểm giao hàng	Thành tiền
DH385	4/11/2023 3:19:40 AM	Đồng Nai	94,762,500 vnd

Hình 42. Xóa phiếu công nợ đại lý

### 3.3.1.13 Doanh thu

Báo Cáo Doanh Thu

Thứ Năm, 25/05/2023 - 21 giờ 36 phút 15 giây

Mã đơn hàng	Mã đại lý	Mã nhân viên lập	Ngày lập	Điểm giao hàng	Tổng tiền
DH1653	DL558	NV01	30/4/2023	132 Lê Đại Hành, Q10, TP HCM	4,479,300 vnd
DH1827	DL558	NV01	25/5/2023	Đồng Nai	210,780,000 vnd
DH2416	DL558	NV01	30/4/2023	Sư vạn hạnh q10	8,613,000 vnd
DH4042	DL558	NV01	2/4/2023	90a, Phố Đức Chính, Phường 01, Quận Bình Thạnh	9,648,000 vnd
DH5407	DL6730	NV01	6/4/2023	Đồng Nai	102,700,000 vnd
<b>Tổng cộng hóa đơn :</b>					<b>336,220,300 vnd</b>

Hình 43. Doanh thu đơn hàng

### 3.3.1.14 Thống kê

Thống Kê

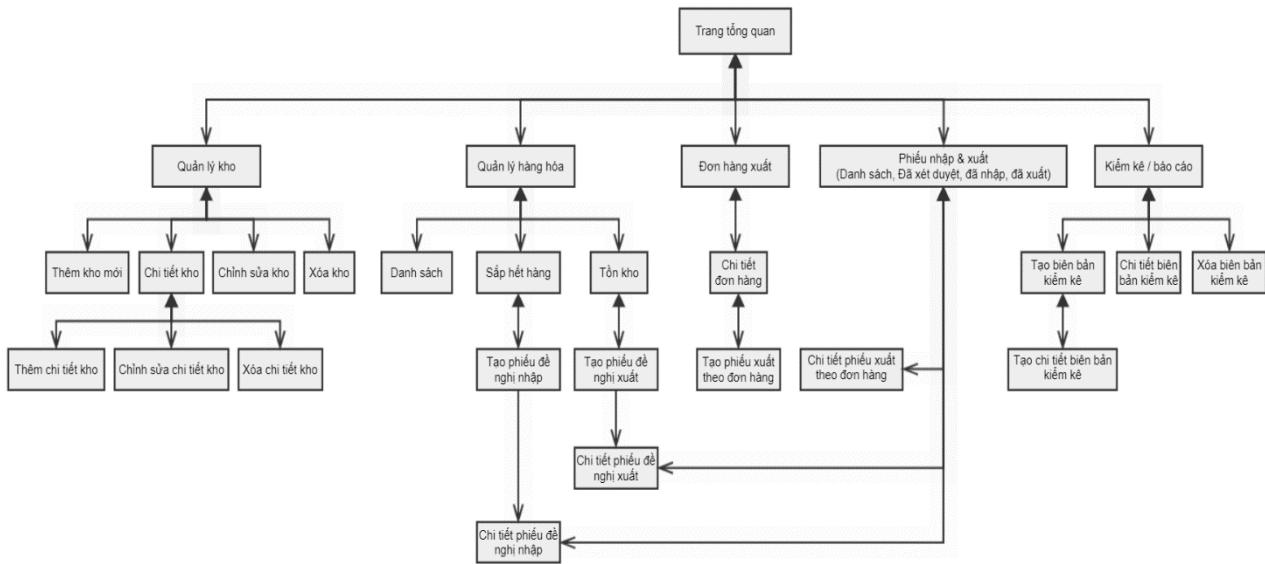
Thứ Năm, 25/05/2023 - 21 giờ 36 phút 39 giây

Mã SP	Tên sản phẩm	Đơn vị	Giá	Tổng hàng bán	Tổng tiền
SP13	Pepsi_01	Lon	350,000 vnd	100	35,000,000 vnd
SP15	Coca cola	Thùng	423,000 vnd	601	254,223,000 vnd
SP39	Sting Thái	Thùng	254,000 vnd	101	25,654,000 vnd
SP48	Trà ô long	Thùng	321,000 vnd	430	138,030,000 vnd
SP55	cocal light_Thùng	Lon	202,100 vnd	1000	202,100,000 vnd
SP641	Nha đam	Lon	45,000 vnd	0	0 vnd
SP66	monster	Thùng_Lon	421,000 vnd	10	4,210,000 vnd
SP68	Sting dầu dừa	Thùng_chai	122,000 vnd	0	0 vnd
SP717	Sữa vinamilk _socola	Thùng_chai	45,000 vnd	0	0 vnd
SP84	cocal ít đường _T	Thùng_Lon	411,230 vnd	0	0 vnd

Hiện 1 đến 10 của 12 danh mục

Hình 44. Thống kê tổng lượng bán sản phẩm

### 3.4. Sơ đồ liên kết các trang giao diện phòng Kho



Hình 45: Sơ đồ liên kết phòng kho

#### 3.4.1 Giao diện theo sơ đồ phòng Kho

##### 3.4.1.1 Trang chủ phòng kho

Giao diện trang chủ phòng kho:

- Tổng Quan:**
  - THÔNG TIN KHO:** Tổng 5 kho được quản lý. Chi trách nhiệm quản lý kho hàng: nhập xuất, kiểm kê, báo cáo hàng hóa.
  - SẮP HẾT HÀNG:** 0 sản phẩm. Số sản phẩm cảnh báo hết cần nhập thêm.
  - THÔNG TIN SẢN PHẨM:** Tổng 12 sản phẩm. Tổng số sản phẩm có trong các kho.
  - SẢN PHẨM TỒN KHO:** 0 sản phẩm. Tổng số sản phẩm tồn kho trong các kho.
  - PHIẾU NHẬP KHO:** Tổng 1 phiếu được tạo. Phiếu nhập hàng hóa vào kho.
  - PHIẾU XUẤT KHO:** Tổng 2 phiếu được tạo. Phiếu xuất hàng hóa ra khỏi kho.
- Danh sách kho hàng:**

Mã kho	Tên kho	Địa chỉ	Số lượng
K0444	Kho Sóng Thần 4	251/43 Hai Bà Trưng, Quận 1, TPHCM	5 sản phẩm
K077	Nguyễn Bình Khiêm 1	43/21/17a Hồ Xuân Hương, p3, Hà Nội	3 sản phẩm
K080	Hàng Không	123 DBP, P04, Tân Bình, TPHCM	6 sản phẩm
K0853	Kho Sóng Thần 1	84a, Phố Đức Chính, p2, Bình Thạnh, TPHCM	2 sản phẩm
K096	Nguyễn Bình Khiêm	4/27 Nguyễn Bình Khiêm, P. Võ Thị Sáu, Q1, TPHCM	3 sản phẩm

Hiện 10 danh mục

### 3.4.1.2 Quản lý kho

Mã kho	Tên kho	Địa chỉ	Số chi tiết kho	Các chức năng
KO2310	Ginger MacAvddy	27856 Lindbergh Trail	2	
KO2376	Corine Westney	7 Northland Crossing	0	
KO2417	Glenna Bartrup	38404 Hallows Drive	0	
KO2428	Leonarie Fenemore	804 Messerschmidt Point	0	
KO2437	Brooks Ramsdell	003 Schiller Street	0	
KO2485	Ted Mattaser	6983 Eagle Crest Point	0	
KO2745	Gabey Baynton	38535 Dennis Pass	1	
KO2828	Garwin Swallow	29 Walton Avenue	0	
KO2830	Alika Homig	1 Gerald Park	0	
KO2948	Penny Losano	49303 School Junction	0	

Hiện 1 đến 10 của 88 danh mục

Lùi 1 2 3 4 5 ... 9 Tiếp

### 3.4.1.3 Thêm mới kho hàng

Mã kho	Tên kho	Số lượng	Địa chỉ
KO444	Kho Sóng Thần 4	5 sản phẩm	251/43 Hai Bà trưng, quận 1, TPHCM
KO77	Nguyễn Bình Khiêm 1	3 sản phẩm	43/21/17a Hồ Xuân Hương, p3, Hà Nội
KO80	Hàng Không	6 sản phẩm	123 DBR P04, Tân Bình, TPHCM
KO853	Kho Sóng Thần 1	2 sản phẩm	84a, Phố Đức Chính , p2, Bình Thạnh, TPHCM
KO96	Nguyễn Bình Khiêm	3 sản phẩm	4/27 Nguyễn Bình Khiêm, P. Võ Thị Sáu, Q1, TPHCM

### 3.4.1.4 Chính sửa kho hàng

Chỉnh sửa thông tin kho "Kho Sóng Thần 4"

Mã kho (Mã định): KO444

Tên kho: Kho Sóng Thần 4

Địa chỉ: 251/43 Hai Bà Trưng, Quận 1, TP.HCM

Lưu thay đổi Trở lại

### 3.4.1.5 Xóa một kho

Xóa Kho

Bạn muốn xóa kho "Kho Sóng Thần 4"?

Mã kho: KO444

Tên Kho: Kho Sóng Thần 4

Địa chỉ: 251/43 Hai Bà Trưng, Quận 1, TP.HCM

Xóa vĩnh viễn Trở lại

### 3.4.1.6 Danh sách chi tiết kho

STT	Mã sản phẩm	Tên sản phẩm	Ngày nhập	Ngày xuất gần nhất	Số lượng	Tình trạng	Chức năng
52	SP13	Pepsi_001	4/6/2023 11:30:00 PM	4/14/2023 12:00:00 PM	513	Còn hàng	
53	SP15	Coca cola	4/6/2023 11:30:00 PM	4/10/2023 12:00:00 PM	500	Còn hàng	
54	SP48	Trà ô long	4/6/2023 11:30:00 PM	1/1/2000 12:00:00 AM	500	Còn hàng	
55	SP55	cocal light_Thung	4/6/2023 11:30:00 PM	4/12/2023 12:00:00 AM	532	Còn hàng	
56	SP55	cocal light_Thung	4/6/2023 11:31:00 PM	1/1/2000 12:00:00 AM	300	Còn hàng	

Hiện 10 danh mục

Tim kiếm:

Thêm chi tiết kho Phản phổi sản phẩm

Trở lại danh sách kho

Lùi 1 Tiếp

### 3.4.1.7 Phân phối sản phẩm

Chọn sản phẩm:

SP13 - Pepsi\_001 - 513

(\*) Mã SP - Tên SP - Số lượng

Phân phối ngay

Chọn kho nhận:

KO2310 - Ginger MacAvddy - 27856 Lindbergh Trail

(\*) Mã kho - Tên kho - Địa chỉ

Nhập số lượng (Nhỏ hơn sản phẩm đã chọn)

Nhập vào số lượng cần nhập vào kho này!

### 3.4.1.8 Tạo chi tiết kho

Tạo Chi Tiết Kho

Tạo chi tiết kho của kho KO444

Mã sản phẩm

SP13

Số lượng / thùng

Ngày nhập vào kho

mm/dd/yyyy -- --

Thêm chi tiết kho

Trả lại

Trợ giúp tìm kiếm sản phẩm và mã sản phẩm

- SP13 - Pepsi\_001
- SP15 - Coca cola
- SP39 - Sting Thái
- SP48 - Trà ô long
- SP55 - cocal light\_Thung
- SP641 - Nha đam
- SP66 - monster
- SP68 - Sting dầu dó
- SP717 - Sữa vinamilk \_socola
- SP84 - cocal ít đường \_T
- SP87 - Sữa hạt
- SP95 - NumberOne

### 3.4.1.9 Xóa chi tiết kho

Xóa Chi Tiết Kho

Bạn muốn xóa chi tiết kho 52 của kho KO444?

Ngày nhập: 4/6/2023 11:30:00 PM

Ngày xuất gần nhất: 4/14/2023 12:00:00 PM

Số lượng (thùng): 513

Tình trạng: Còn hàng

Tên kho: Kho Sóng Thần 4

Tên sản phẩm: Pepsi\_001

Xóa vĩnh viễn

Trả lại

### 3.4.1.10 Chỉnh sửa chi tiết kho

### 3.4.1.11 Quản lý hàng hóa

DANH SÁCH SẢN PHẨM							
Hiện	10	danh mục		Tim kiếm:			
Mã sản phẩm	Tên sản phẩm	Hình minh họa	Giá	Đơn vị tính	Ngày sản xuất	Hạn sử dụng	Tổng tồn
SP13	Pepsi_001		350002	Thùng	2/28/2023 12:00:00 AM	4/5/2023 12:00:00 AM	1373
SP15	Coca cola		423000	Thùng	4/4/2023 12:00:00 AM	4/19/2023 12:00:00 AM	913
SP356	Trà trứng 2T		12000	Thùng	5/20/2023 12:00:00 AM	8/18/2023 12:00:00 AM	
SP39	Sting Thái		254000	Thùng	4/5/2023 12:00:00 AM	5/23/2023 12:00:00 AM	1600

### 3.4.1.12 Danh sách sản phẩm sắp hết hàng

Mã Kho	Mã sản phẩm	Tên sản phẩm	Giá tiền	Đơn vị tính	Ngày xuất gần nhất	Số lượng	Tình trạng
KO444	SP13	Pepsi_001	350002	Thùng	4/11/2023 12:00:00 AM	50	Sắp hết hàng
KO444	SP15	Coca cola	423000	Thùng	11/25/2022 12:00:00 AM	50	Sắp hết hàng

### 3.4.1.13 Danh sách sản phẩm tồn kho

Mã Kho	Mã sản phẩm	Tên sản phẩm	Giá tiền	Đơn vị tính	Số lượng	Tình trạng
KO6918	SP9662	Joann Danbye	455460		1022	Tồn kho

### 3.4.1.14 Tạo phiếu đề nghị nhập kho

Mã Kho	Mã sản phẩm	Tên sản phẩm	Giá tiền	Số lượng còn lại	Đơn vị tính
KO444	SP13	Pepsi_001	350002	50	Thùng
KO444	SP15	Coca cola	423000	50	Thùng

### 3.4.1.15 Tạo phiếu đề nghị xuất kho

### 3.4.1.16 Danh sách đơn hàng đã xét duyệt

### 3.4.1.17 Chi tiết đơn hàng đã xét duyệt và chưa có hóa đơn

### 3.4.1.18 Chi tiết đơn hàng đã xét duyệt và có hóa đơn

Danh Sách Chi Tiết Đơn Hàng

Thứ Sáu, 14/04/2023 - 15 giờ 38 phút 03 giây

Mã đơn hàng số: DH0565  
Điểm giao hàng: Đồng Nai

Mã sản phẩm	Tên sản phẩm	Số lượng	Đơn giá	Đơn vị tính	Thành tiền
SP39	Sting Thái	2	254,000 vnđ	Thùng	508,000 vnđ
SP13	Pepsi_001	14	350,000 vnđ	Thùng	4,900,028 vnđ

Tổng cộng đơn hàng: 5,408,028 vnđ

+ Tạo phiếu xuất

Trở lại

**HÓA ĐƠN ĐƠN HÀNG**

Ngày: 4/14/2023 11:47:08 AM  
Mã đơn hàng: DH0565

Tên đại lý: Trương Hường  
Loại đại lý: LDL03  
Địa chỉ: Đồng Nai

Mã CTDH	Mã SP	Tên sản phẩm	Số lượng	Thành tiền
2154	SP39	Sting Thái	2	508,000 vnđ
2157	SP13	Pepsi_001	14	4,900,028 vnđ

Tổng tiền: 5,408,028 vnđ

Chiết khấu: 5%

Tổng hóa đơn: 5,137,627 vnđ

Kế toán  
(ký và ghi rõ họ tên)  
Nguyễn Thị Diễm Quỳnh

### 3.4.1.19 Tạo phiếu xuất kho theo đơn hàng

Tạo Phiếu Xuất Kho

Thứ Sáu, 14/04/2023 - 15 giờ 39 phút 17 giây

Tên đơn vị: Công ty TNHH nước giải khát NQK  
Theo mã đơn hàng số: DH119

**PHIẾU XUẤT KHO THEO ĐƠN HÀNG**

Ngày: 4/14/2023 3:38:55 PM  
Người làm phiếu: Diễm Khang

Kho xuất hàng:  
K096 - 4/27 Nguyễn Bình Khiêm, P. Võ Thị Sáu, Q1, TP.HCM

Đại lý nhận hàng: DL225 - 10.21.11.40  
Địa chỉ nhận hàng: 94, Trần Hưng Đạo

Danh sách sản phẩm:

Mã sản phẩm	Tên sản phẩm	Số lượng	Đơn giá	Đơn vị tính	Thành tiền
SP39	Sting Thái	10	254,000 vnđ	Thùng	2,540,000 vnđ

Tổng cộng đơn hàng: 2,540,000 vnđ

Thủ kho  
(ký và ghi rõ họ tên)  
Nguyễn Thị Diễm Khang

Lưu phiếu Hủy bỏ

### 3.4.1.20 Danh sách phiếu nhập xuất đã tạo

Mã phiếu	Ngày lập	Loại phiếu	Tình trạng	Lựa chọn
KDPNK52	4/14/2023 3:28:05 PM	Phiếu đề nghị nhập hàng vào kho	Chưa xét duyệt	<a href="#">Chi tiết</a>
KDPNK843	4/14/2023 12:41:40 PM	Phiếu đề nghị nhập hàng vào kho	Đã nhập	<a href="#">Chi tiết</a>
KDPXK433	4/14/2023 3:28:58 PM	Phiếu đề nghị xuất hàng tồn	Đã xét duyệt	<a href="#">Chi tiết</a>
PHXHD0481	4/14/2023 2:50:16 PM	Phiếu xuất kho theo đơn hàng số DH0555	Đã cập nhật	<a href="#">Chi tiết</a>
PHXHD0629	4/14/2023 11:53:14 AM	Phiếu xuất kho theo đơn hàng số DH0565	Đã cập nhật	<a href="#">Chi tiết</a>

### 3.4.1.21 Danh sách phiếu đã xét duyệt

Mã phiếu	Ngày lập	Loại phiếu	Tình trạng	Lựa chọn
KDPXK433	4/14/2023 3:28:58 PM	Phiếu đề nghị xuất hàng tồn	Đã xét duyệt	<a href="#">Chi tiết</a>

### 3.4.1.22 Danh sách phiếu đã nhập

Mã phiếu	Ngày lập	Loại phiếu	Tình trạng	Lựa chọn
KDPNK843	4/14/2023 12:41:40 PM	Phiếu đề nghị nhập hàng vào kho	Đã nhập	<a href="#">Chi tiết</a>

### 3.4.1.23 Danh sách phiếu đã xuất

Mã phiếu	Ngày lập	Loại phiếu	Tình trạng	Lựa chọn
KDPXK433	4/14/2023 3:28:58 PM	Phiếu đề nghị xuất hàng tồn	Đã xuất	<a href="#">Chi tiết</a>

### 3.4.1.24 Chi tiết phiếu đề nghị xuất kho

Mã Kho	Mã sản phẩm	Tên sản phẩm	Giá tiền	Đơn vị tính	Trong kho	Cần xuất		
K077	SP39	Sting Thái	254000	Thùng	1002	254,508,000 VNĐ	470	119,380,000 VNĐ
K0444	SP48	Trà ô long	321000	Thùng	1023	328,383,000 VNĐ	491	157,611,000 VNĐ

Thủ kho  
(ký và ghi rõ họ tên)  
Nguyễn Thị Diễm Khang

### 3.4.1.25 Chi tiết phiếu đề nghị nhập kho

**Chi Tiết Phiếu Đề Nghị Nhập Kho**

Thứ Sáu, 14/04/2023 - 15 giờ 28 phút 15 giây

Tình trạng: Chưa xét duyệt

Tên đơn vị: Công ty TNHH nước giải khát NQK

**PHIẾU ĐỀ NGHỊ NHẬP KHO**

Ngày: 4/14/2023 3:28:05 PM  
Mã số: KDPNK52

- Kinh gửi: Trưởng phòng kinh doanh  
- Họ tên người làm phiếu: Nguyễn Thị Diễm Khang

- Chi tiết nhập kho:

Mã Kho	Mã sản phẩm	Tên sản phẩm	Giá tiền	Đơn vị tính	Trong kho		Cần nhập	
					Số lượng	Thành tiền	Số lượng	Thành tiền
KO444	SP13	Pepsi_001	350002	Thùng	50	17,500,100 VND	463	162,050,926 VND
KO444	SP15	Coca cola	423000	Thùng	50	21,150,000 VND	463	195,849,000 VND

Thủ kho  
(ký và ghi rõ họ tên)

Nguyễn Thị Diễm Khang

### 3.4.1.26 Kiểm kê / báo cáo

**Báo Cáo Kho Hàng**

Thứ Sáu, 14/04/2023 - 15 giờ 44 phút 36 giây

**TỔNG SỐ KHO**  
5 kho

**TỔNG SẢN PHẨM**  
12 sản phẩm

**TỔNG PHIẾU XUẤT HÀNG**  
1 phiếu

**SẢN PHẨM SẮP HẾT HÀNG**  
2 sản phẩm

**SẢN PHẨM TỒN KHO**  
0 sản phẩm

**TỔNG PHIẾU NHẬP HÀNG**  
2 phiếu

**DANH SÁCH BIÊN BẢN KIỂM KÊ**

Hiện 10 danh mục

Mã biên bản	Mã nhân viên lập	Chức vụ	Tên biên bản	Ngày lập	Mã kho	Chức năng
BB482	NV03	Nhân viên kho	Biên bản kiểm kê lần n	4/7/2023 12:34:42 AM	KO444	
BB579	NV06	Nhân viên kho	Biên bản kiểm kê lần 1	4/7/2023 12:20:14 PM	KO77	
BB792	NV01	Trưởng phòng kinh doanh	Biên bản kiểm kê lần 1	3/30/2023 9:30:46 PM	KO444	
BB884	NV03	Nhân viên kho	Biên bản kiểm kê lần 2	4/1/2023 11:06:15 PM	KO444	

Hiện 1 đến 4 của 4 danh mục

Lùi 1 Tiếp

### 3.4.1.27 Tạo biên bản kiểm kê

Tạo Biên Bản

Tạo biên bản kiểm kê

Mã nhân viên lập: NV03

Tên kho: Kho Sóng Thần 4

Tên biên bản:

Hủy bỏ Tiếp tục

Tạo Chi Tiết Biên Bản

Tạo chi tiết biên bản kiểm kê

Tên sản phẩm: Pepsi\_001

Số lượng thực tế:

Chi tiết:

Lưu dữ liệu Trở lại Hủy bỏ

### 3.4.1.28 Chi tiết biên bản kiểm kê hàng hóa

Chi Tiết Biên Bản

BIÊN BẢN KIỂM KÊ HÀNG HÓA

Ngày: 4/7/2023 12:34:42 AM

Mã số: BB482\_5

Tên biên bản: Biên bản kiểm kê lần n

Nhân viên kiểm kê: NV03 - Diễm Khang

Thông tin kho:

- Mã kho: K0444
- Tên kho: Kho Sóng Thần 4
- Địa chỉ kho: 251/43 Hai Bà Trưng, quận 1, TPHCM

Thông tin sản phẩm:

Mã sản phẩm	Tên sản phẩm	Đơn vị tính	Đơn giá	Ngày sản xuất	Hạn sử dụng	Theo số kho		Theo kiểm kê		Chênh lệnh	
						Số lượng	Thành tiền	Số lượng	Thành tiền	Số lượng	Thành tiền
SP13	Pepsi_001	Thùng	350002	4/11/2023 12:00:00 AM	4/22/2023 12:00:00 AM	300	105,000,600	800	280,001,600	500	175,001,000

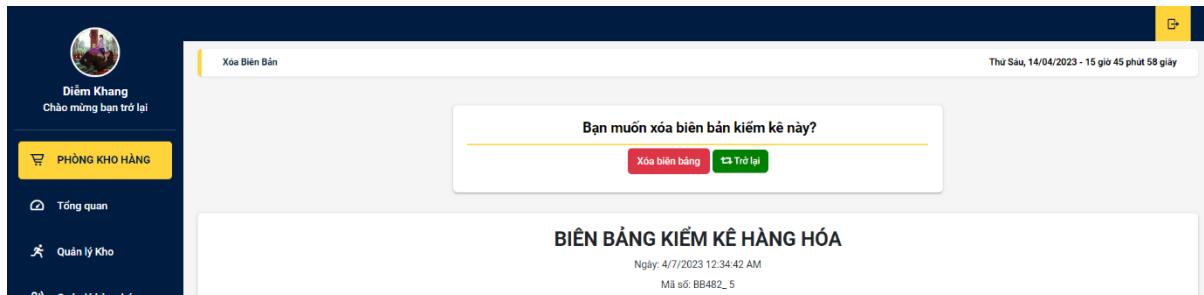
Chi tiết: Do bán đắt

Thủ kho  
(ký và ghi rõ họ tên)

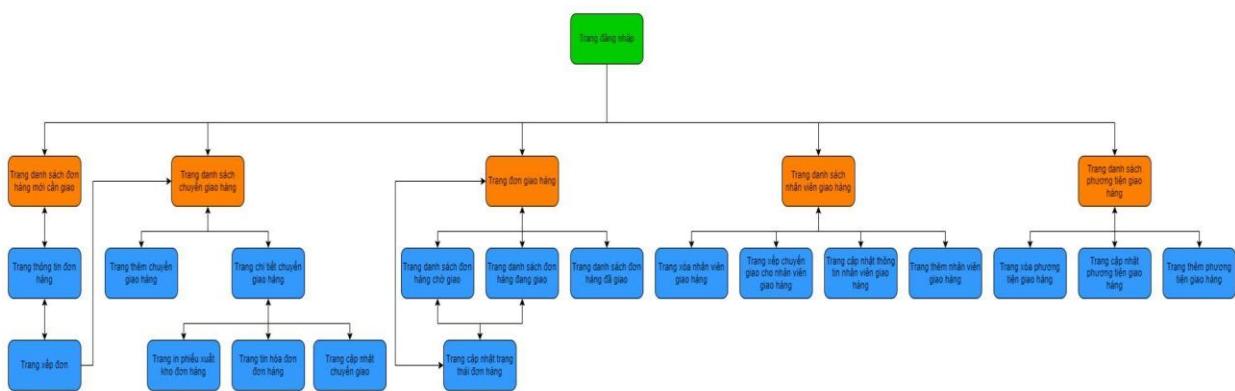
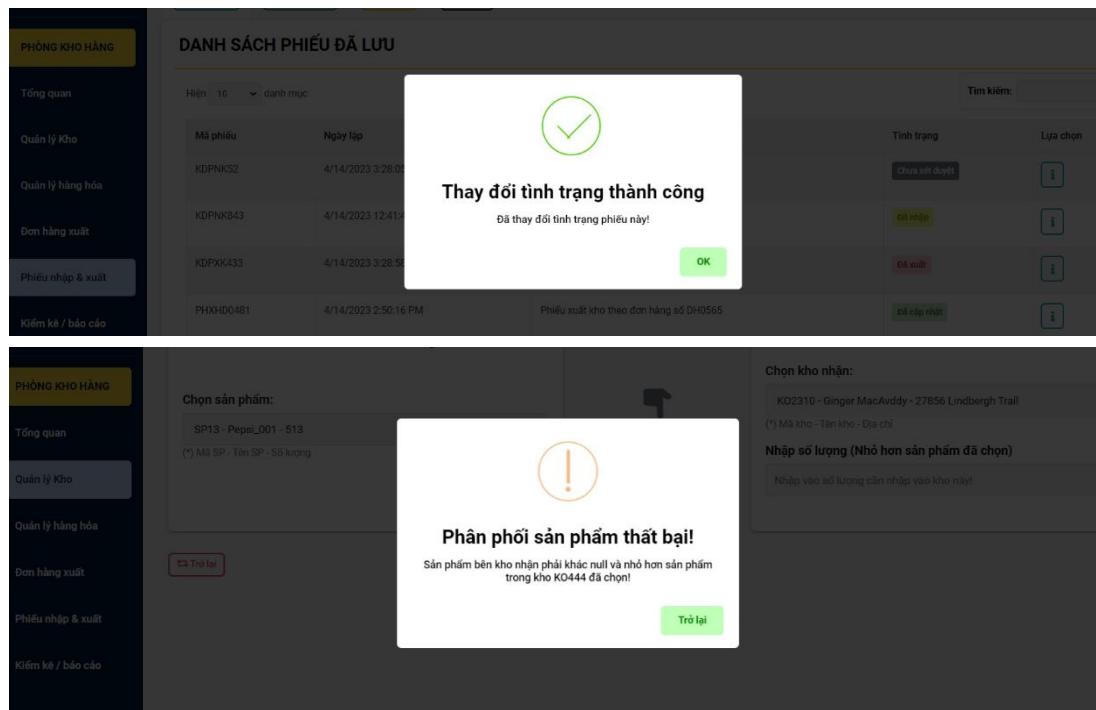
Nguyễn Thị Diễm Khang

Trở lại In phiếu

### 3.4.1.29 Xóa biên bản kiểm kê



### 3.4.1.30 Màn hình thông báo



## 3.5. Sơ đồ liên kết các trang giao diện Giao Hàng

### 3.5.1 Giao diện phòng giao hàng:

#### 3.5.1.1 Đơn hàng mới

Mã đơn hàng	Ngày lên đơn	Mã đại lý	Tên đại lý	Số điện thoại đại lý	Địa điểm giao hàng	Tình trạng thanh toán	Thành tiền	Xếp đơn
DH2751	5/5/2023 1:41:28 PM	DL558	Trương Hướng	0966253223	TPHCM	Đã thanh toán	6,319,685 vnd	
DH3372	5/5/2023 11:43:10 AM	DL802	Điêm Quỳnh	0908455325	Bình Chánh	Đã thanh toán	4,887,500 vnd	
DH6004	5/5/2023 11:35:58 AM	DL558	Trương Hướng	0966253223	TPHCM	Chưa thanh toán	6,897,019 vnd	
DH6524	5/5/2023 1:41:55 PM	DL1	Chaman	(371) 3909126	Suite 56	Chưa thanh toán	6,590,070 vnd	

Hiện 1 đến 4 của 4 danh mục

Lùi 1 Tiếp

Hình 46. Danh sách đơn hàng mới cần giao

Hình 47. In danh sách đơn hàng mới cần giao

Danh sách chi tiết của đơn hàng DH3372

Mã sản phẩm	Tên sản phẩm	Số lượng	Đơn giá	Đơn vị tính	Thành tiền
SP39	Sting Thái	10	254,000 vnđ	Thùng	2,540,000 vnđ
SP48	Trà ô long	10	321,000 vnđ	Thùng	3,210,000 vnđ

Tổng cộng đơn hàng: 5,750,000 vnđ

**Quay lại** **Xếp đơn**

Hình 48. Chi tiết đơn hàng

XẾP ĐƠN DH2751

Mã giao hàng	Tên nhân viên giao	Mã phương tiện	Ngày lập	Trạng thái	Khu vực giao	Ngày giao
GH362	51A-57694	5/5/2023 2:02:42 PM	Chờ giao			

**Làm mới danh sách**

**DANH SÁCH CÁC CHUYỂN GIAO HÀNG HIỆN CÓ**

Hiện 10 danh mục

**Quay lại**

Hình 49. Chọn chuyến giao cho đơn hàng

### 3.4.1.2. Chuyển giao hàng

Danh Sách Các Chuyển Giao Hàng

Mã giao hàng	Tên nhân viên giao	Mã phương tiện	Ngày lập	Trạng thái	Khu vực giao	Ngày giao	Chi tiết
GH362	Nguyễn Việt Trí	51A-57694	5/5/2023 2:02:42 PM	Chờ giao	TP HCM	5/20/2023 9:00:00 AM	
GH512	Đinh Mạnh An	51A-34456	5/5/2023 3:48:34 PM	Chờ giao	TP HCM	5/10/2023 4:00:00 PM	
GH612	Lê Anh Nguyên	51A-45363	5/5/2023 3:48:47 PM	Chờ giao			

Hiện 1 đến 3 của 3 danh mục

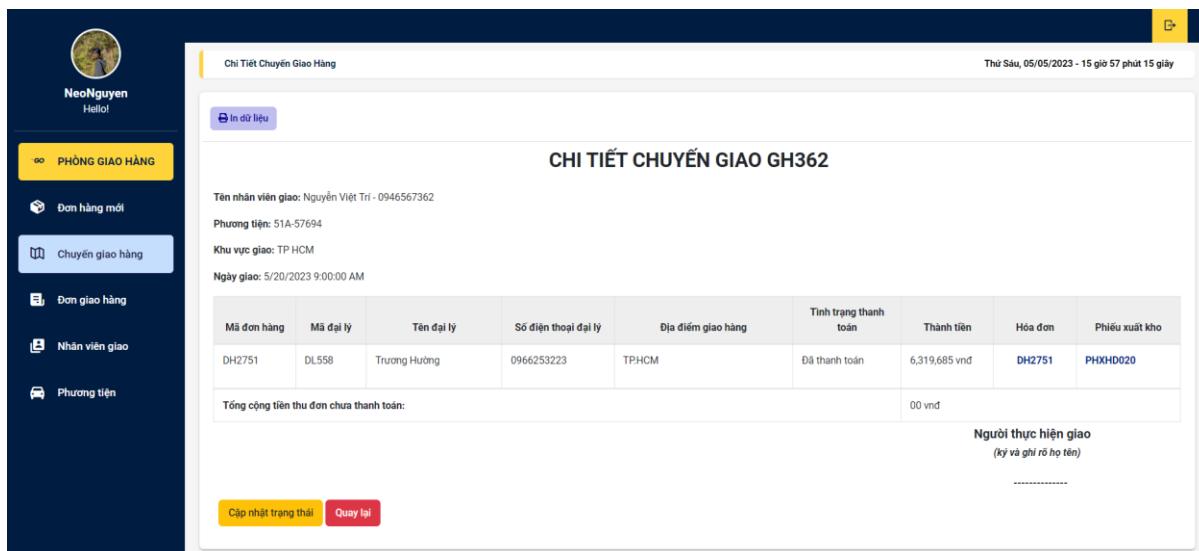
**Thêm chuyển giao hàng** **Làm mới danh sách** **In dữ liệu**

**DANH SÁCH CÁC CHUYỂN GIAO HÀNG**

Hiện 10 danh mục

**Quay lại**

Hình 50. Danh sách các chuyến giao



Hình 51. Chi tiết chuyến giao

DANH SÁCH CÁC CHUYỂN GIAO HÀNG

Hiện 10 danh mục

Mã giao hàng	Tên nhân viên giao	Mã phương tiện	Ngày lập	Trạng thái	Khu vực giao	Ngày giao	Chi tiết
GH362	Nguyễn Việt Trí	51A-57694	5/5/2023 2:02:42 PM	<button>Chờ giao</button>	TP HCM	5/20/2023 9:00:00 AM	
GH512	Bình Mạnh An	51A-34456	5/5/2023 3:48:34 PM	<button>Chờ giao</button>	TP HCM	5/10/2023 4:00:00 PM	
GH612	Lê Anh Nguyễn	51A-45363	5/5/2023 3:48:47 PM	<button>Chờ giao</button>			

Hiện 1 đến 3 của 3 danh mục

In 1 tờ giấy

Máy in đích Microsoft Print to PDF

Trang Tất cả

Màu Đen và trắng

Chế độ cài đặt khác

In Hủy

Hình 52. In danh sách các chuyến giao

Mã nhân viên	Tên nhân viên giao	Tình trạng
SHIP03	Nguyễn Việt Trí	Đang giao hàng
SHIP05	Đinh Mạnh An	Đang giao hàng
SHIP06	Lê Anh Nguyễn	Đang giao hàng
SHIP07	Đặng Văn Ngu	Nghỉ phép
SHIP285	Lê Minh Thủ	Sẵn sàng
SHIP527	Nguyễn Khắc	Sẵn sàng
SHIP893	Nguyễn Đình Trọng	Sẵn sàng

Mã phương tiện	Tên phương tiện	Tình trạng
51A-88974	Hyundai 576	Bán tri
51A-57694	Hyundai 95743	Sử dụng
51A-45363	Suzuki 600	Sử dụng
51A-54987	Hyundai Daichii	Sẵn sàng
51A-34456	Hyundai 584 A1	Sẵn sàng
51A-88-325	Hyundai 584 A1	Sẵn sàng

*Hình 53. Thêm chuyến giao*

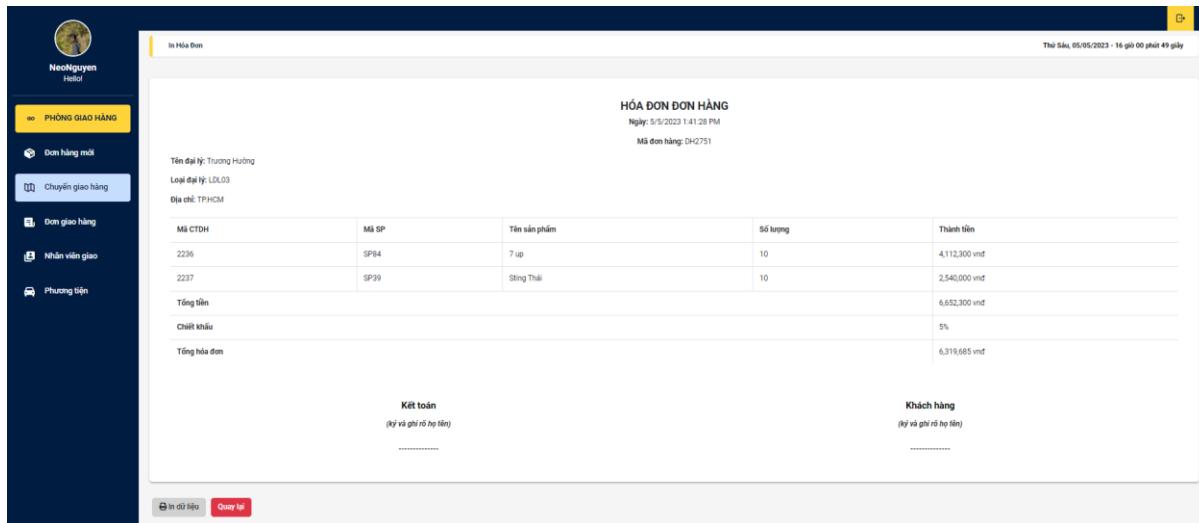
Mã đơn hàng	Mã đại lý	Tên đại lý	Số điện thoại đại lý	Địa điểm giao hàng	Tình trạng thanh toán	Thành tiền	Hóa đơn	Phiếu xuất kho
DH2751	DL558	Trương Hướng	0966253223	TPHCM	Đã thanh toán	6,319,685 vnđ	DH2751	PHXHD020

Tổng cộng tiền thu đơn chưa thanh toán: 00 vnđ

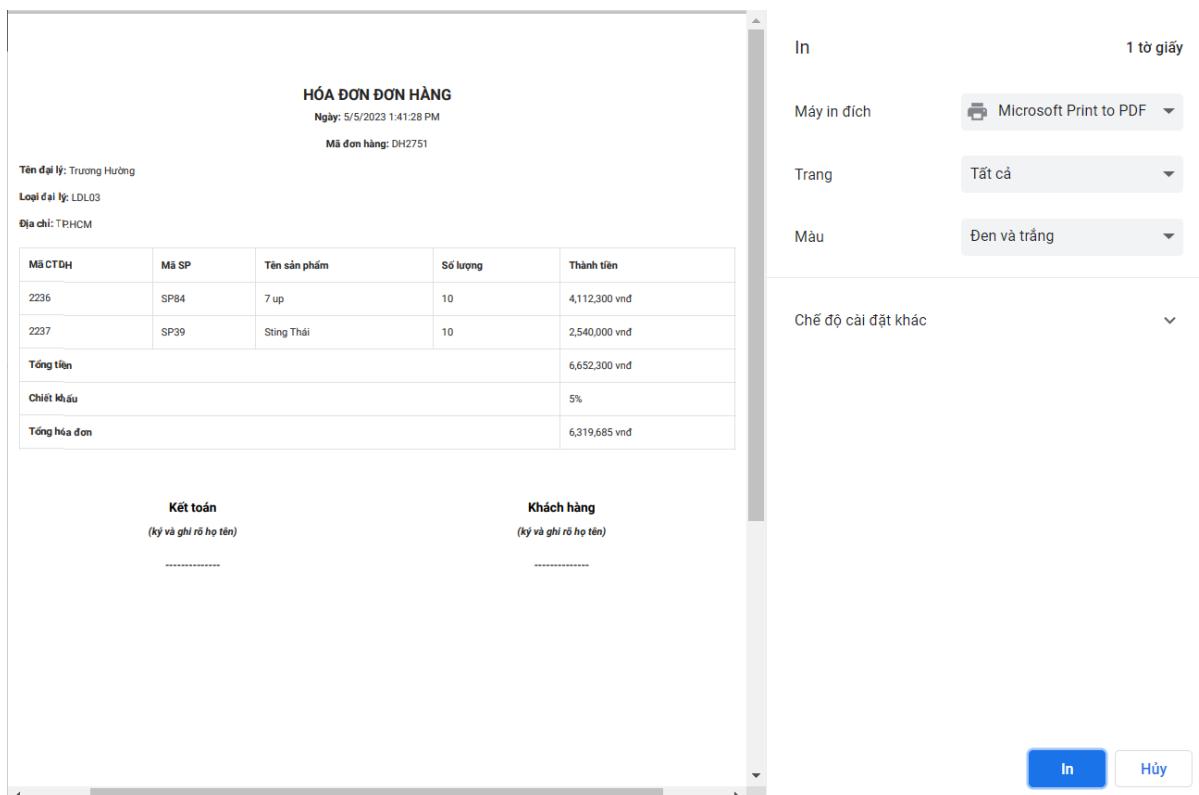
Người thực hiện giao  
(ký và ghi rõ họ tên)

Cập nhật trạng thái Quay lại

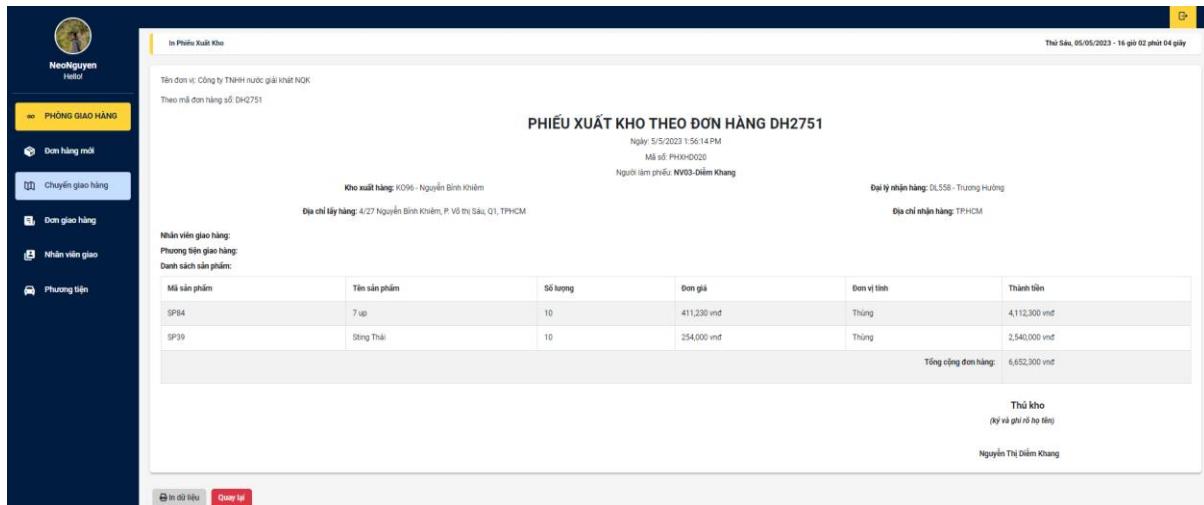
*Hình 54. Chi tiết chuyến giao*



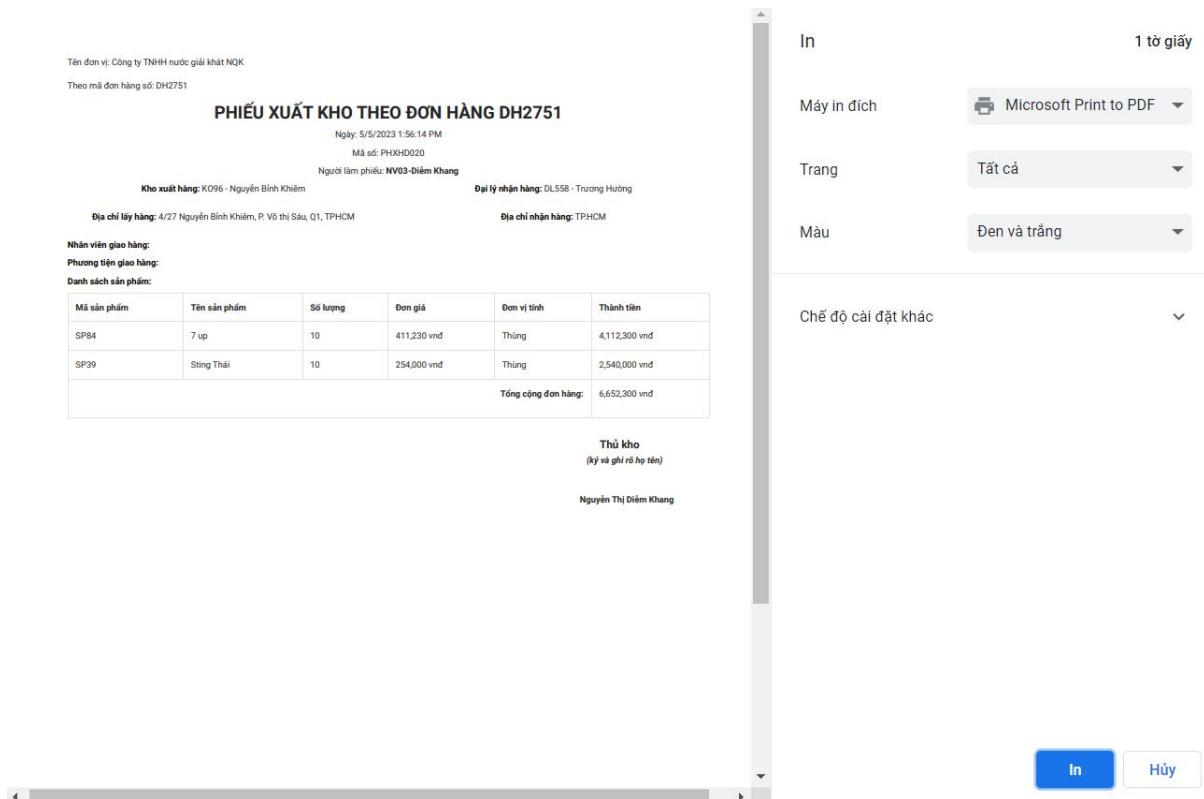
Hình 55. Hóa đơn đơn hàng



Hình 56. In hóa đơn đơn hàng



Hình 57. Phiếu xuất kho theo đơn hàng



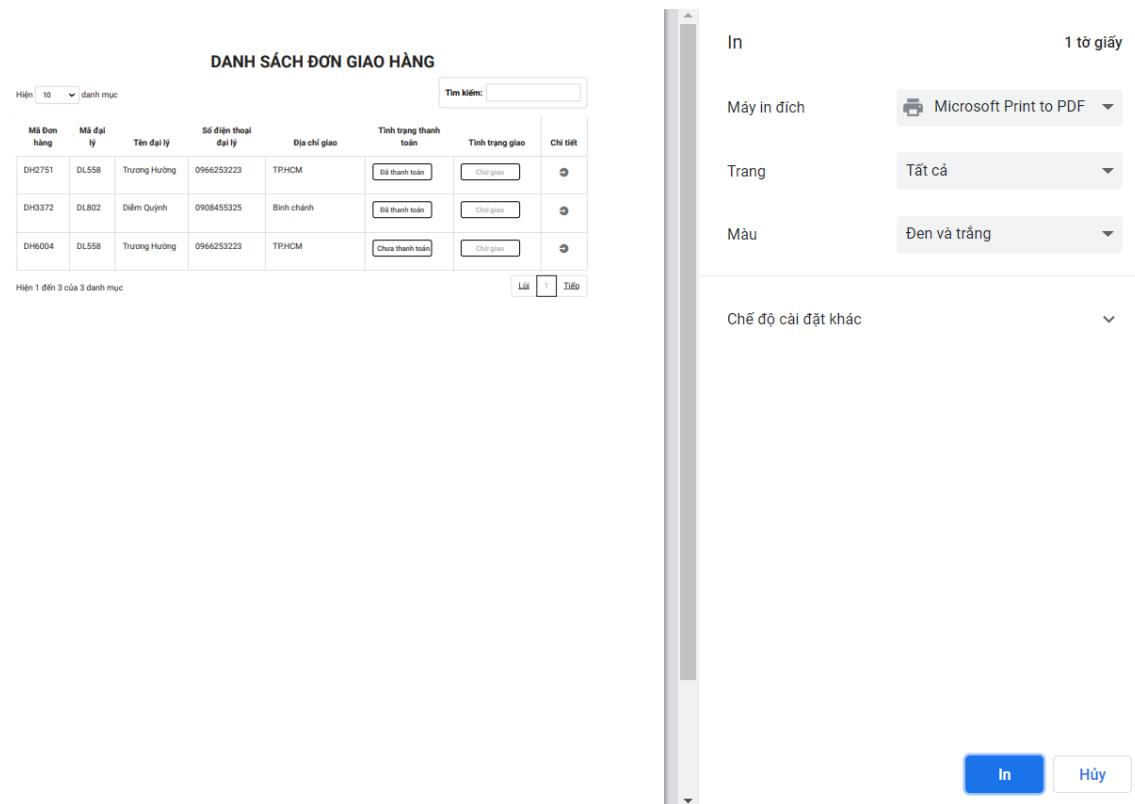
Hình 58. In phiếu xuất kho theo đơn hàng

*Hình 59. Cập nhật chuyển giao*
*Hình 60. Xóa chuyển giao*

### 3.4.1.3. Đơn giao hàng

Mã Đơn hàng	Mã đại lý	Tên đại lý	Số điện thoại đại lý	Địa chỉ giao	Tình trạng thanh toán	Tình trạng giao	Chi tiết
DH2751	DL558	Trương Hướng	0966253223	TPHCM	<span style="background-color: #2e7131; color: white;">Đã thanh toán</span>	<span style="background-color: #2e7131; color: white;">Chờ giao</span>	<span style="color: orange;">X</span>
DH3372	DL802	Điêm Quỳnh	0908455325	Bình Chánh	<span style="background-color: #2e7131; color: white;">Đã thanh toán</span>	<span style="background-color: #2e7131; color: white;">Chờ giao</span>	<span style="color: orange;">X</span>
DH6004	DL558	Trương Hướng	0966253223	TPHCM	<span style="background-color: #ff4d4d; color: white;">Chưa thanh toán</span>	<span style="background-color: #ff4d4d; color: white;">Chờ giao</span>	<span style="color: orange;">X</span>

*Hình 61. Xóa chuyển giao*



Hình 62. In danh sách đơn giao hàng

Mã sản phẩm	Tên sản phẩm	Số lượng	Đơn giá	Đơn vị tính	Thành tiền
SP84	7 up	10	411,230 vnđ	Thùng	4,112,300 vnđ
SP39	Sting Thái	10	254,000 vnđ	Thùng	2,540,000 vnđ
					Tổng cộng đơn hàng: 6,652,300 vnđ

Hình 63. Danh sách chi tiết đơn hàng

Mã Đơn hàng	Mã đại lý	Tên đại lý	Số điện thoại đại lý	Địa chỉ giao	Tình trạng thanh toán	Tình trạng giao	Chi tiết
DH2751	DL558	Trương Hường	0966253223	TPHCM	<span>Đã thanh toán</span>	<span>Chờ giao</span>	<span>...</span>
DH3372	DL802	Diễm Quýnh	0908455325	Bình chánh	<span>Đã thanh toán</span>	<span>Chờ giao</span>	<span>...</span>
DH6004	DL558	Trương Hường	0966253223	TPHCM	<span>Chưa thanh toán</span>	<span>Chờ giao</span>	<span>...</span>

Hiện 1 đến 3 của 3 danh mục

[Quay lại](#)

*Hình 64. Danh sách đơn chờ giao*

Mã Đơn hàng	Mã đại lý	Tên đại lý	Số điện thoại đại lý	Địa chỉ giao	Tình trạng thanh toán	Tình trạng giao	Chi tiết
DH2751	DL558	Trương Hường	0966253223	TPHCM	<span>Đã thanh toán</span>	<span>Đang giao</span>	<span>...</span>

Hiện 1 đến 1 của 1 danh mục

[Quay lại](#)

*Hình 65. Danh sách đơn hàng đang giao*

Mã Đơn hàng	Mã đại lý	Tên đại lý	Số điện thoại đại lý	Địa chỉ giao	Tình trạng thanh toán	Tình trạng giao	Chi tiết
DH3372	DL802	Diễm Quýnh	0908455325	Bình chánh	<span>Đã thanh toán</span>	<span>Đã giao</span>	<span>...</span>

Hiện 1 đến 1 của 1 danh mục

[Quay lại](#)

*Hình 66. Danh sách đơn hàng đã giao*

### 3.4.1.3. Nhân viên giao

Mã nhân viên	Tên nhân viên giao hàng	Số điện thoại	Địa chỉ	Tình trạng	Cập nhật
SHIP06	Lê Anh Nguyên	0923456233	235 Tô Hiến Thành, quận 10, TP HCM	Đang giao hàng - GH612	
SHIP03	Nguyễn Việt Trí	0946567362	7/23 Nam Kỳ Khởi Nghĩa, quận 1, TPHCM	Đang giao hàng - GH362	
SHIP05	Đinh Mạnh An	0913487327	54/3 Trường Chinh, quận Tân Bình, TP HCM	Sẵn sàng	
SHIP285	Lê Minh Thế	093223452	234 Trần Văn Đang, quận 11, TP HCM	Sẵn sàng	

Hình 67. Danh sách nhân viên giao hàng

Hình 68. Thêm nhân viên giao hàng

Cập Nhật Thông Tin Nhân Viên Giao

**NHÂN VIÊN GIAO HÀNG SHIP05**

Mã nhân viên: SHIP05

Tên nhân viên:

Đinh Mạnh An

SDT:

0913487327

Địa chỉ:

54/3 Trường Chinh, quận Tân Bình, TP HCM

Tình trạng làm việc:

Sẵn sàng

Lưu Quay lại

*Hình 69. Cập nhật nhân viên giao*

Xếp Chuyển Giao

**NHÂN VIÊN GIAO HÀNG SHIP05 - Đinh Mạnh An**

Xếp chuyển giao:

GH612

Lưu Quay lại

Làm mới danh sách

**DANH SÁCH CÁC CHUYẾN GIAO HÀNG CHƯA CÓ NHÂN VIÊN VẬN CHUYỂN**

Mã giao hàng	Tên nhân viên giao	Mã phương tiện	Ngày lập	Trạng thái	Khu vực giao	Ngày giao
GH612	Lê Anh Nguyên	51A-45363	5/5/2023 3:48:47 PM	Chờ giao		

Hiện 10 danh mục

Tim kiếm:

Lùi Tiếp

*Hình 70. Xếp chuyến xe cho nhân viên giao hàng*

Xóa Nhân Viên Giao Hàng

**XÓA NHÂN VIÊN GIAO HÀNG SHIP05**

Tên nhân viên: Đinh Mạnh An

Số điện thoại: 0913487327

Địa chỉ: 54/3 Trường Chinh, quận Tân Bình, TP HCM

Tình trạng: Sẵn sàng

Xóa Quay lại

*Hình 71. Xóa nhân viên giao hàng*

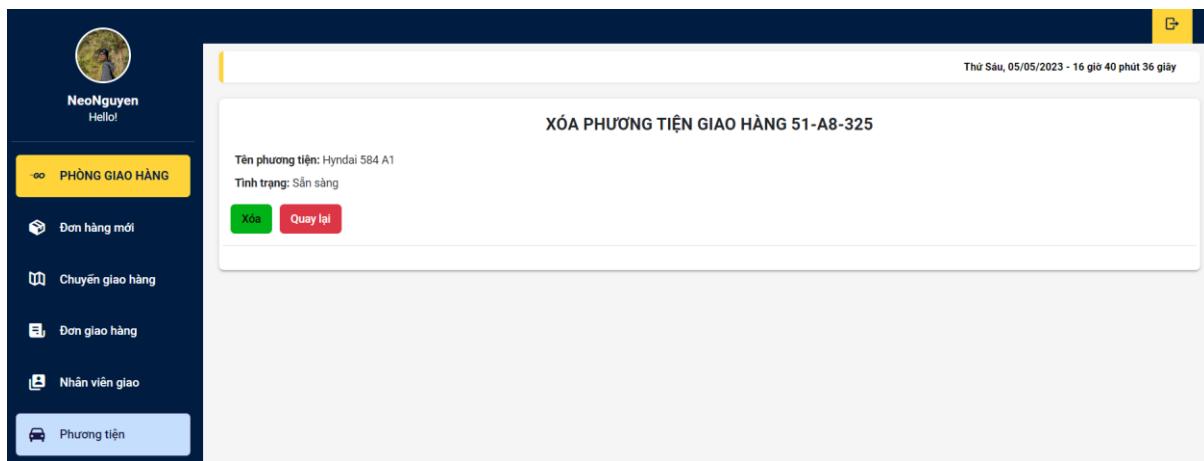
### 3.4.1.2. Phương tiện

Mã phương tiện	Tên phương tiện	Trọng tải	Tình trạng	Cập nhật
51A-88974	Hyundai 576	500	Bảo trì	
51-A8-325	Hyundai 584 A1	1000	Sẵn sàng	
51A-34456	Suzuki 500	500	Sẵn sàng	
51A-54987	Hyundai Daichi	1000	Sẵn sàng	
51A-45363	Suzuki 600	800	Sử dụng	

Hình 72. Danh sách phương tiện giao hàng

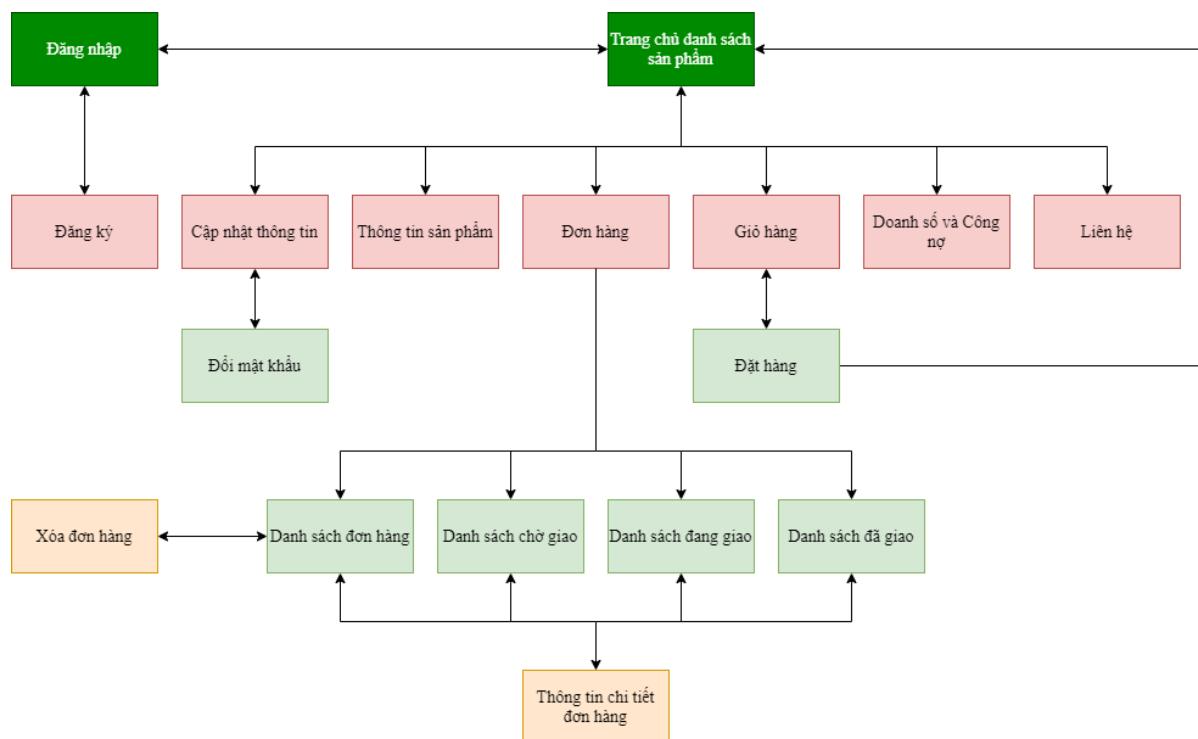
Hình 73. Thêm phương tiện giao hàng mới

Hình 74. Cập nhật phương tiện giao hàng



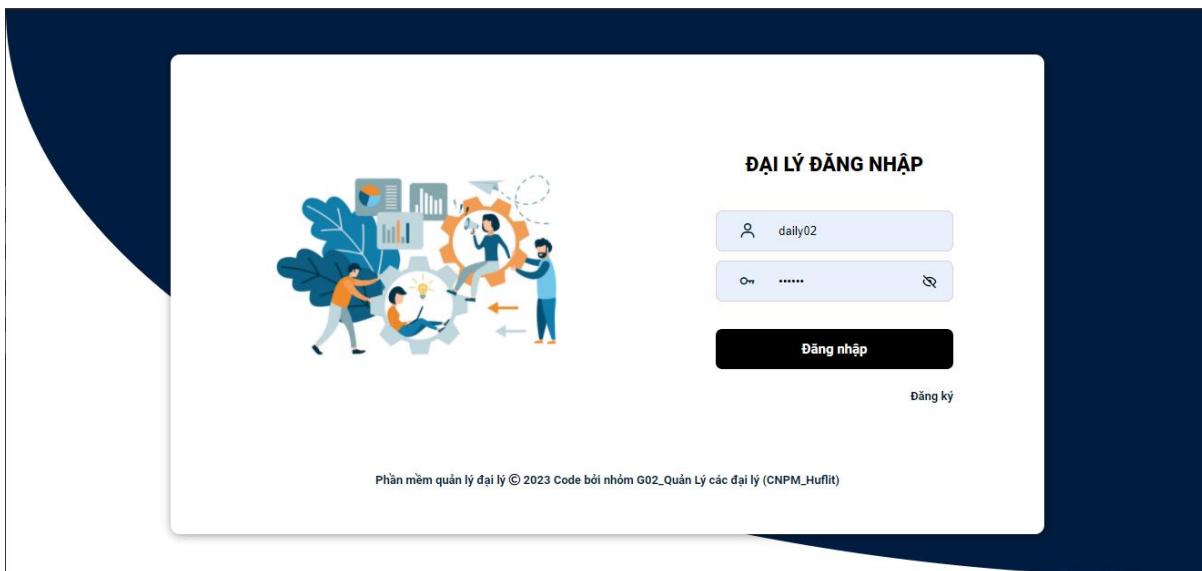
Hình 75. Xóa phương tiện giao hàng

### 3.6. Sơ đồ liên kết các trang giao diện Đại lý



### 3.6.1 Giao diện theo sơ đồ Đại lý

#### 3.6.1.1 Đăng nhập



Hình 76. Trang đăng nhập

### 3.6.1.2 Đăng ký

Hình 77. Trang đăng ký

### 3.6.1.3 Trang chủ danh sách sản phẩm

Hình 78. Trang chủ danh sách sản phẩm

### 3.6.1.4 Cập nhật thông tin

Hình 79. Trang cập nhật thông tin đại lý

### 3.6.1.5 Đổi mật khẩu

Hình 80. Trang đổi mật khẩu tài khoản đại lý

### 3.6.1.6 Thông tin sản phẩm

Hình 81. Trang thông tin sản phẩm

### 3.6.1.7 Danh sách đơn hàng

Hình 82. Trang danh sách đơn hàng

### 3.6.1.8 Danh sách chờ giao

Mã đơn hàng	Ngày lập	Trạng thái tiếp nhận	Tình trạng thanh toán	Tình trạng giao hàng	Chức năng
DH8333	4/15/2023 11:14:10 AM	Đã xét duyệt	Chưa thanh toán	Chờ giao hàng	
DH8717	4/15/2023 10:38:19 AM	Đã xét duyệt	Chưa thanh toán	Chờ giao hàng	

Hình 83. Trang danh sách chờ giao

### 3.6.1.9 Danh sách đang giao

Mã đơn hàng	Ngày lập	Trạng thái tiếp nhận	Tình trạng thanh toán	Tình trạng giao hàng	Chức năng
DH8333	4/15/2023 11:14:10 AM	Đã xét duyệt	Chưa thanh toán	Đang giao hàng	

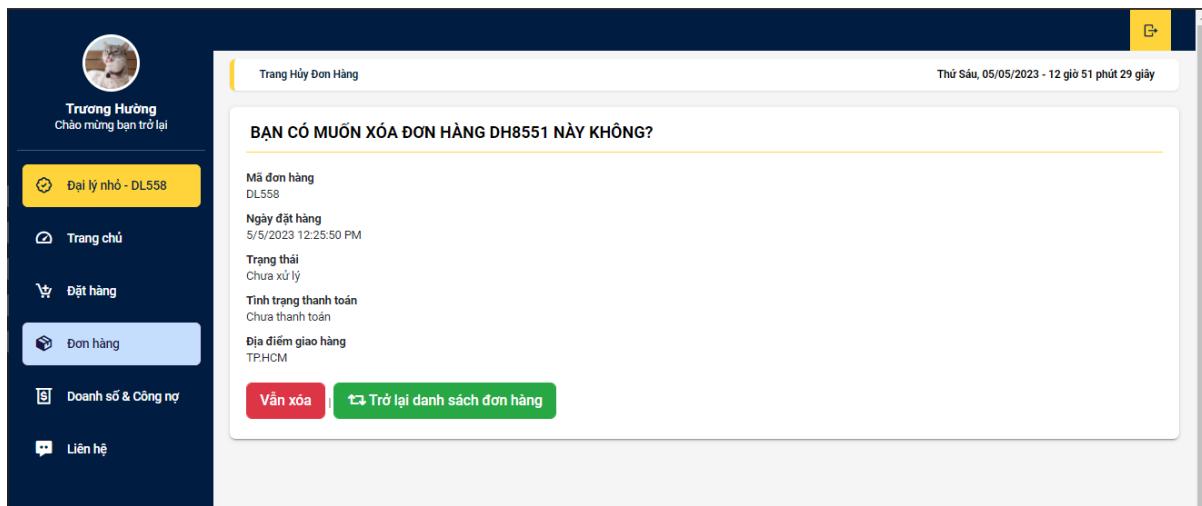
Hình 84. Trang danh sách đang giao

### 3.6.1.10 Danh sách đã giao

Mã đơn hàng	Ngày lập	Trạng thái tiếp nhận	Tình trạng thanh toán	Tình trạng giao hàng	Chức năng
DH3281	4/14/2023 3:30:51 PM	Đã xét duyệt	Đã thanh toán	Đã giao hàng	

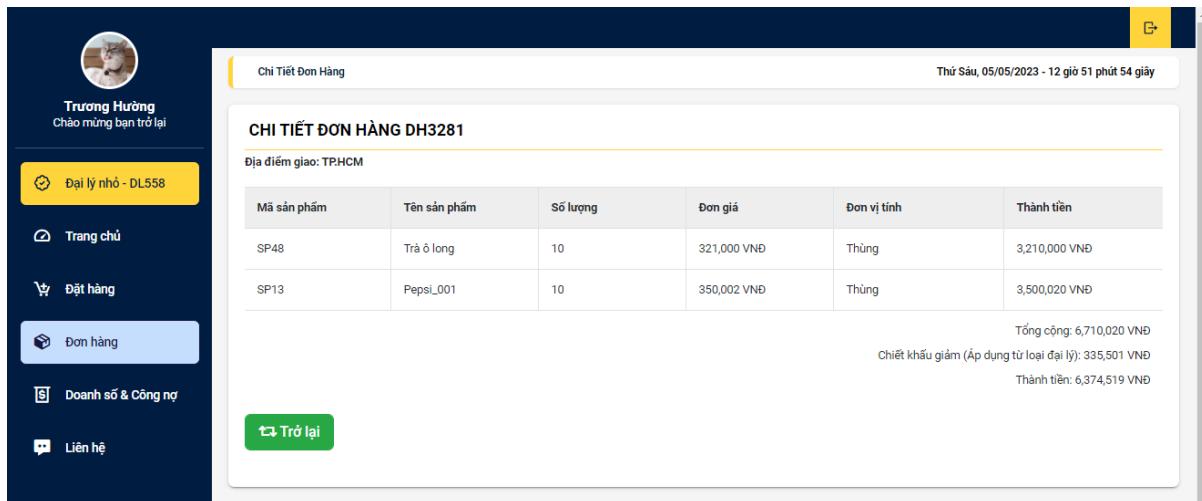
Hình 85. Trang danh sách đã giao

### 3.6.1.11 Xóa đơn hàng



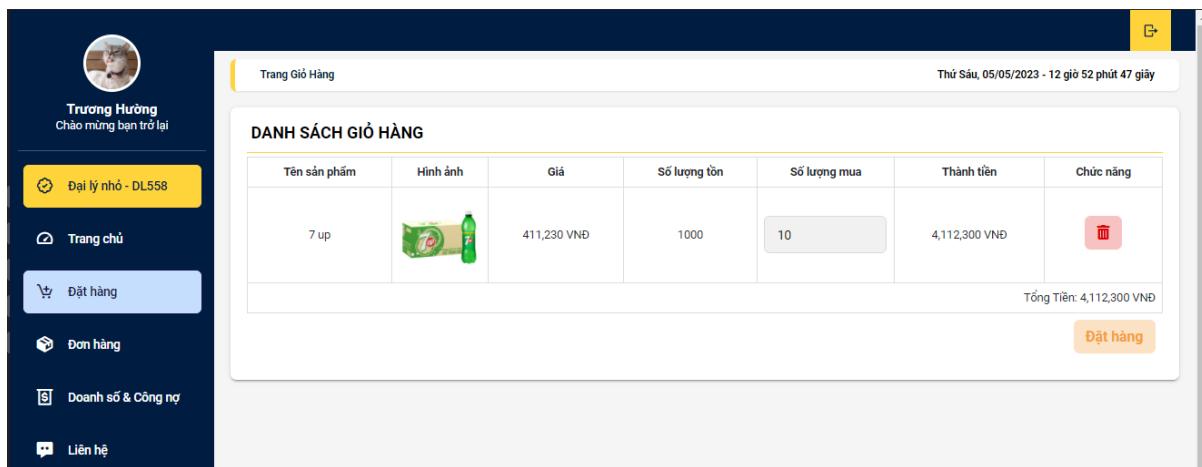
Hình 86. Trang xóa đơn hàng

### 3.6.1.12 Trang chi tiết đơn hàng



Hình 87. Trang chi tiết đơn hàng

### 3.6.1.13 Giỏ hàng



Hình 88. Trang giỏ hàng

### 3.6.1.14 Đặt hàng

Tên sản phẩm	Hình ảnh	Giá	Số lượng	Tổng tiền
7 up		411,230 VNĐ	10	4,112,300 VNĐ

Tổng: 4,112,300 VNĐ  
Chiết khấu giảm (Áp dụng từ loại đại lý): 205,615 VNĐ  
Thành tiền: 3,906,685 VNĐ

**Xác nhận**

Hình 89. Trang công nợ

### 3.6.1.15 Doanh số và Công nợ

Mã công nợ	Tiền nợ	Ngày lập	Ngày phải trả	Trạng thái thanh toán
CN3649	2,286,000 VNĐ	3/14/2023 12:00:00 AM	4/30/2023 12:00:00 AM	Đã thanh toán

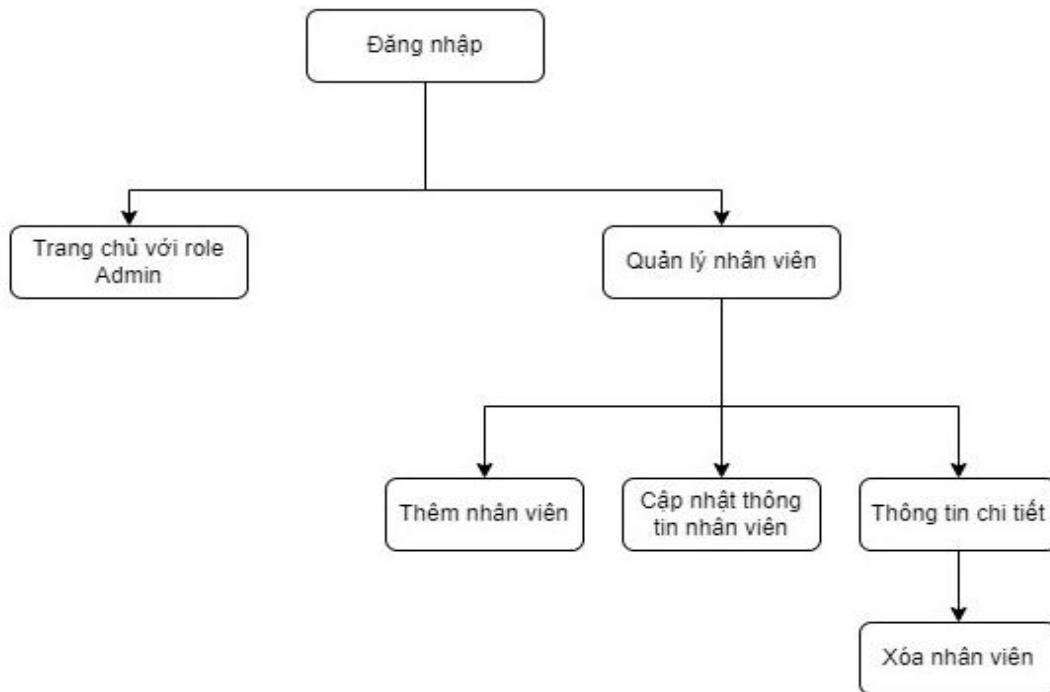
Hình 90. Trang công nợ

### 3.6.1.16 Trang liên hệ

Phòng ban	Số điện thoại	Email
Kinh doanh (Giải quyết vấn đề về đơn hàng)	(+84) 999999999	kinhdoanh@gmail.com
Kế toán (Giải quyết vấn đề về công nợ)	(+84) 111111111	ketoan@gmail.com

Hình 91. Trang liên hệ

### 3.7. Sơ đồ liên kết các trang giao diện Quản trị viên



#### 3.7.1 Giao diện theo sơ đồ Quản trị viên

##### 3.7.1.1 Trang chủ

The screenshot shows the Admin homepage with the following interface elements:

- Header:** Chào mừng bạn trở lại, AdminToanNang, QUẢN TRỊ VIÊN.
- Left sidebar:** QUẢN TRỊ VIÊN (highlighted), Tổng quan, Quản lý nhân viên.
- Main content:**
  - Employee Management:** NHÂN VIÊN, Tổng: 11 nhân viên.
  - Employee List:** Danh sách quản trị viên (Employee List). Includes a search bar (Tim kiếm) and a table with columns: Hình đại diện (Avatar), Mã nhân viên (Employee ID), Họ và tên (Name), Vị trí (Position), Username, and two small icons (edit and info).
  - Table Data:** NV04, AdminToanNang, Nhân Viên, user04; NV50, Dam Vinh Hung, Nhân Viên, user019.
  - Pagination:** Hiện 10 danh mục, 1/2 trang, Tiếp (Next).

Hình 92. Trang chủ admin

### 3.7.1.2. Quản lý nhân viên

Mã nhân viên	Chức vụ	Tên nhân viên	UserName	Chức năng
NV01	Kinh Doanh - NVKD	Tấn Nhã	user01	[Edit] [View]
NV02	Kế Toán - NVKT	Điểm Quynh	user02	[Edit] [View]
NV03	Kho - NVK	Điểm Khang	user03	[Edit] [View]
NV04	Admin - QTHT	AdminToanNang	user04	
NV05	Giao Hàng - NVGH	NeoNguyen	user05	[Edit] [View]
NV06	Kho - NVK	Nguyễn Tuấn Anh	user06	[Edit] [View]
NV118	Kho - NVK	Tú Anh	user0190	[Edit] [View]
NV50	Admin - QTHT	Dam Vinh Hung	user019	[Edit] [View]
NV571	Kinh Doanh - NVKD	Đàm Vinh Hưng	user017	[Edit] [View]
NV610	Kinh Doanh - NVKD	Đàm Vĩnh Hưng	user010	[Edit] [View]

Hiện 1 đến 10 của 11 danh mục

Lùi 1 2 Tiếp

Hình 93. Quản lý nhân viên

### 3.7.1.3. Thêm nhân viên

THÊM TÀI KHOẢN NHÂN VIÊN

Chọn hình ảnh [Choose File] 6662a96799...83c29fe7.jpg

Thông tin cá nhân

Tên nhân viên  
Tú Anh

Địa chỉ  
Bình Chánh

Địa chỉ  
Nữ

Ngày sinh  
05/28/2002

Số điện thoại  
0908923232

Email  
tqn89303@gmail.com

Username  
user011

Chức vụ  
NVKD

Password  
\*\*\*\*\*

Xác nhận password  
\*\*\*\*\*

Hợp lệ!

Hợp lệ!

Save Account

Quay lại danh sách

Hình 94. Thêm nhân viên

### 3.7.1.4. Cập nhật thông tin nhân viên

Hình 95. Cập nhật thông tin nhân viên

### 3.7.1.5. Thông tin chi tiết nhân viên

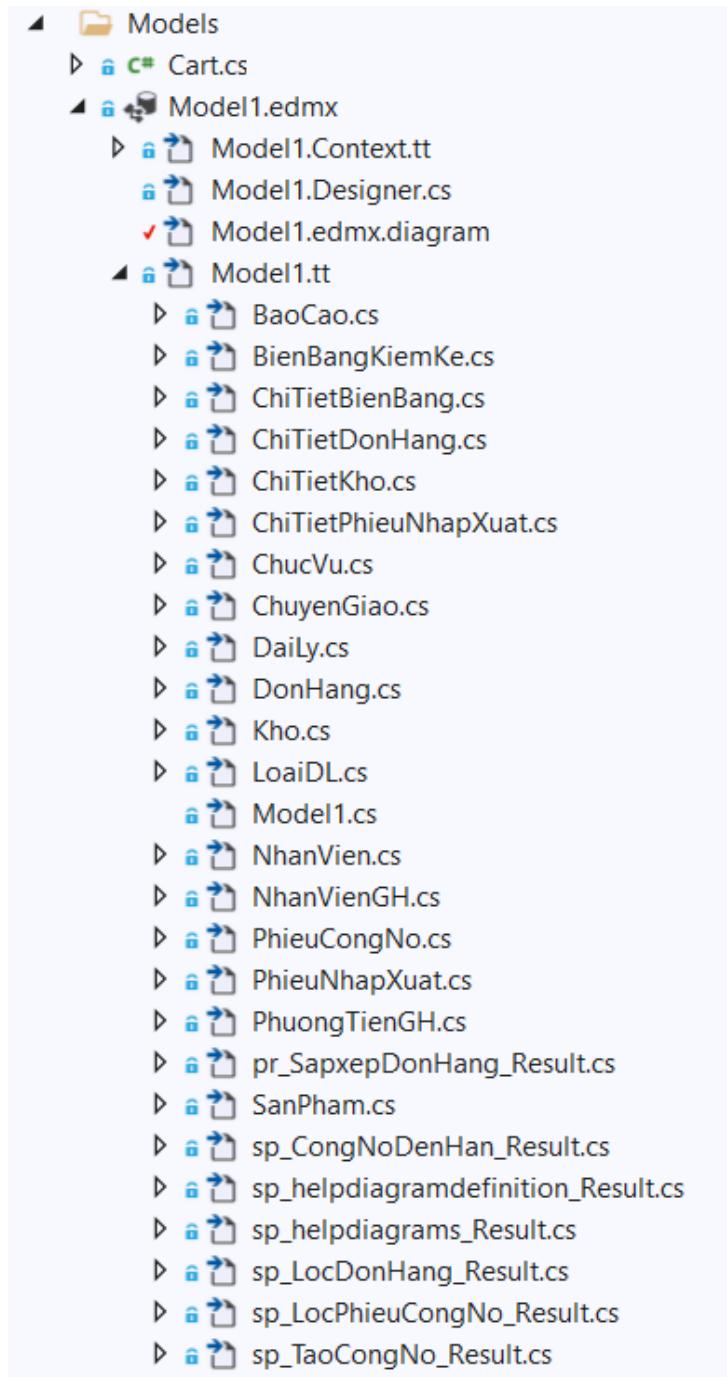
Hình 96. Thông tin chi tiết nhân viên

### 3.7.1.6. Xóa nhân viên

Hình 97. Xóa nhân viên

## CHƯƠNG 3: MÔ TẢ CHI TIẾT ỨNG DỤNG THEO MÔ HÌNH MVC

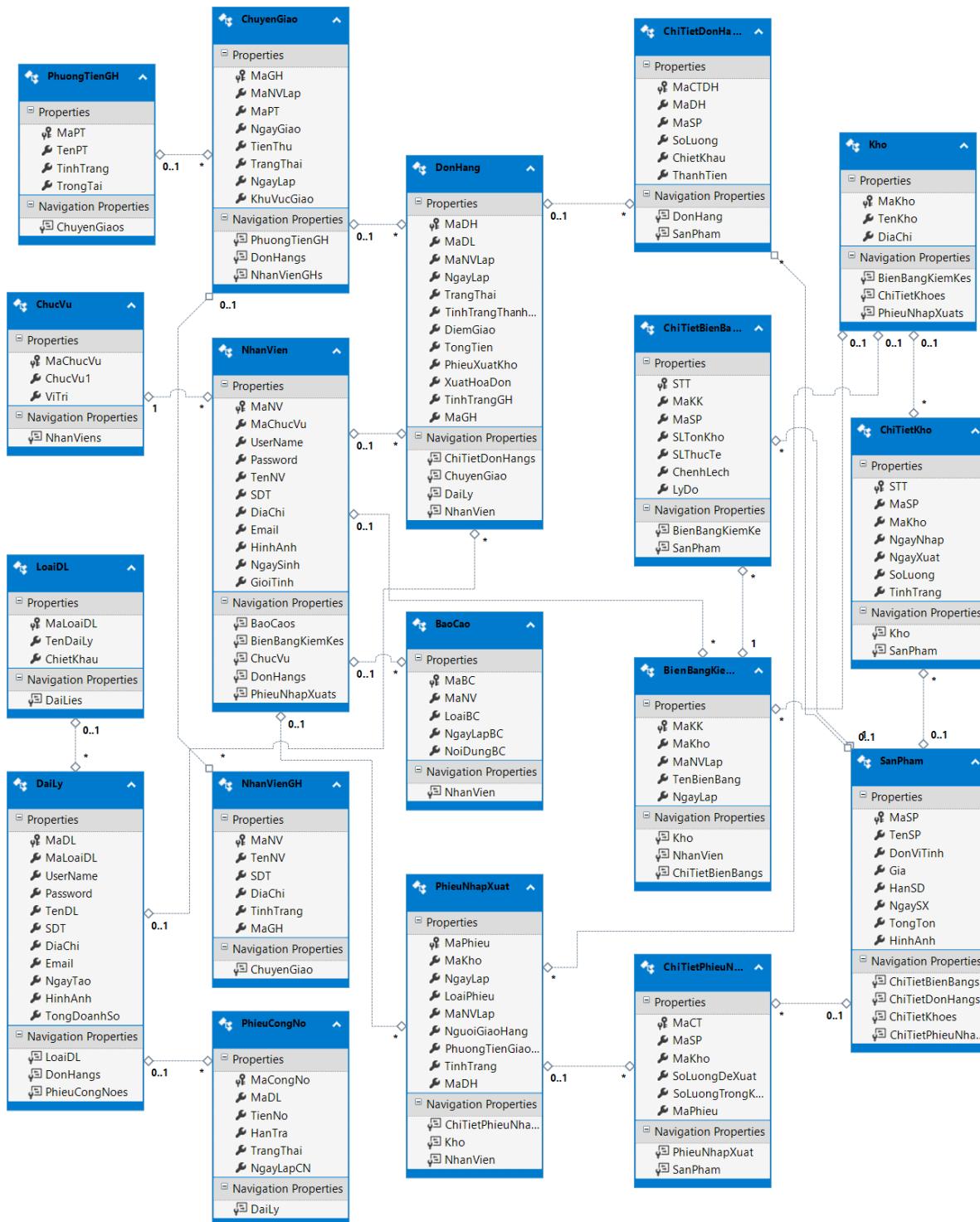
### 4.1. MODELS



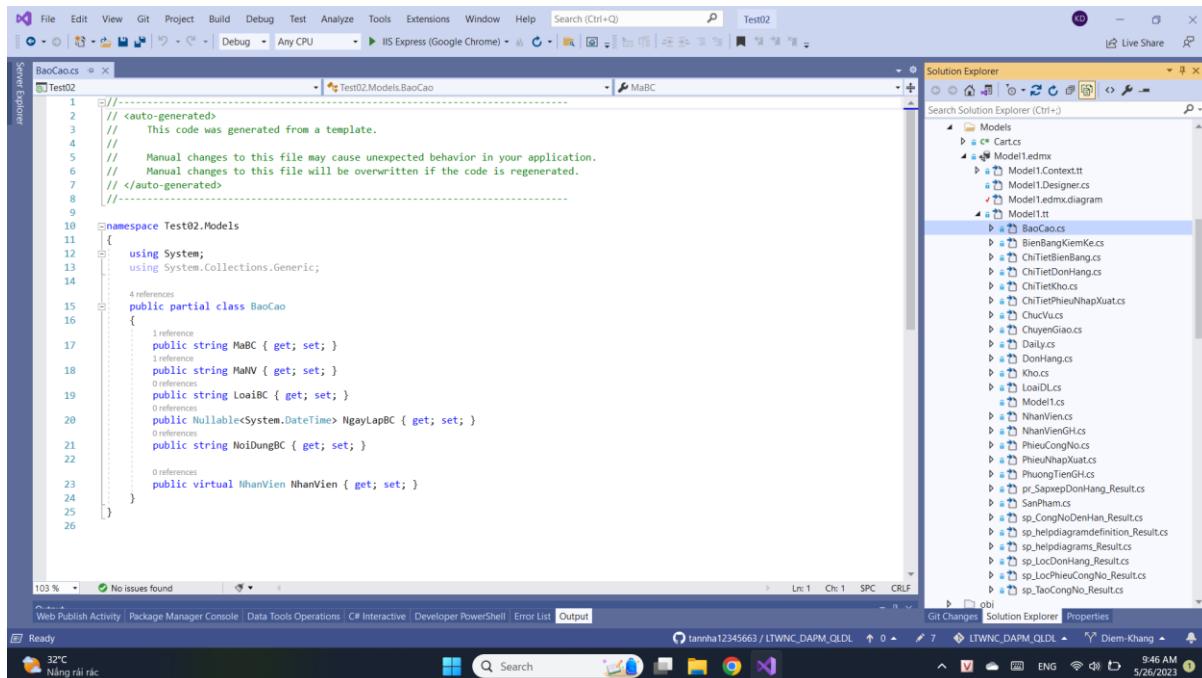
Hình 98: Module model

Model tổng thể của toàn bộ chức năng cũng như đáp ứng các yêu cầu nghiệp vụ đối với mỗi actor (Kinh Doanh, Kế Toán , Kho , Giao Hàng, Admin, Đại lý)

#### 4.1.1. Model Diagram

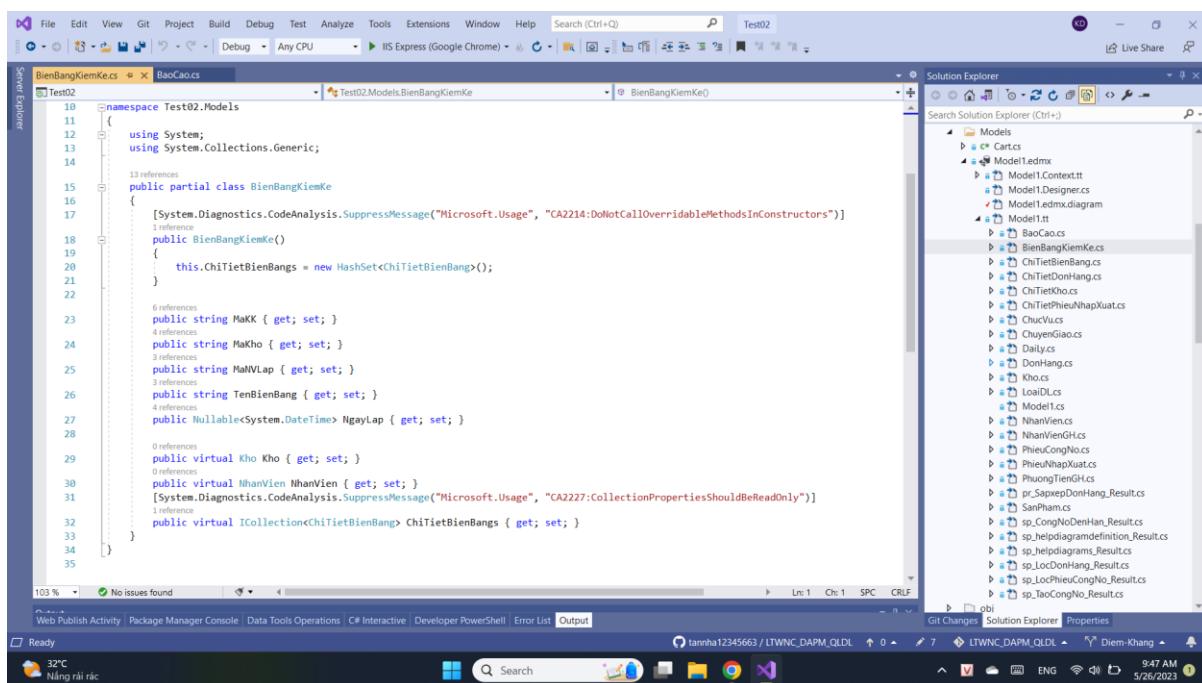


Hình 99. Sơ đồ Model Diagram sau khi đã được triển khai trên asp.net MVC



Hình 100. BaoCao.cs

BaoCao.cs: Thực hiện viết báo cáo kết quả , và một số vấn đề phát sinh được sử dụng cho các phòng Kinh Doanh



Hình 101. BienBangKiemKe.cs

BienBangKiemKe.cs: Mỗi biên bảng kiểm kê sẽ chưa lưu biên bản kiểm kê từ kho khi

```

4 // 
5 // Manual changes to this file may cause unexpected behavior in your application.
6 // Manual changes to this file will be overwritten if the code is regenerated.
7 // </auto-generated>
8 //
9
10 namespace Test02.Models
11 {
12     using System;
13     using System.Collections.Generic;
14
15     9 references
16     public partial class ChiTietBienBang
17     {
18         1 reference
19         public int STT { get; set; }
20         5 references
21         public string MaKK { get; set; }
22         2 references
23         public Nullable<int> SLTonKho { get; set; }
24         2 references
25         public Nullable<int> SLThucTe { get; set; }
26         1 reference
27         public Nullable<int> ChenhLech { get; set; }
28         1 reference
29         public string LyDo { get; set; }
30
31         0 references
32         public virtual BienBangKiemKe BienBangKiemKe { get; set; }
33         0 references
34         public virtual SanPham SanPham { get; set; }
35     }
36
37     0 references
38 }

```

Hình 102. ChiTietBienBang.cs

```

1 // 
2 // </auto-generated>
3 // This code was generated from a template.
4 //
5 // Manual changes to this file may cause unexpected behavior in your application.
6 // Manual changes to this file will be overwritten if the code is regenerated.
7 // </auto-generated>
8 //
9
10 namespace Test02.Models
11 {
12     using System;
13     using System.Collections.Generic;
14
15     17 references
16     public partial class ChiTietDonHang
17     {
18
19         7 references
20         public int MaCTDH { get; set; }
21         30 references
22         public string MaDH { get; set; }
23         1 reference
24         public string MaSP { get; set; }
25         19 references
26         public Nullable<int> Soluong { get; set; }
27         5 references
28         public Nullable<double> ChietKhau { get; set; }
29         12 references
30         public Nullable<double> ThanhTien { get; set; }
31
32         0 references
33         public virtual DonHang DonHang { get; set; }
34         0 references
35         public virtual SanPham SanPham { get; set; }
36     }
37
38     0 references
39 }

```

Hình 103. ChiTietDonHang.cs

ChiTietDonHang.cs: Thực hiện thêm xóa sửa cập nhật thông tin sản phẩm trong đơn hàng áp dụng cho phòng Kinh Doanh, Đại Lý

```

4 // Manual changes to this file may cause unexpected behavior in your application.
5 // Manual changes to this file will be overwritten if the code is regenerated.
6 // </auto-generated>
7 //
8 //
9
10 namespace Test02.Models
11 {
12     using System;
13     using System.Collections.Generic;
14
15     11 references
16     public partial class ChiTietKho
17     {
18         8 references
19         public int STT { get; set; }
20
21         16 references
22         public string MaSP { get; set; }
23
24         1 reference
25         public string MaKho { get; set; }
26
27         4 references
28         public Nullable<System.DateTime> NgayNhap { get; set; }
29
30         4 references
31         public Nullable<System.DateTime> NgayXuat { get; set; }
32
33         47 references
34         public Nullable<int> Soluong { get; set; }
35
36         35 references
37         public string TinhTrang { get; set; }
38
39         0 references
40         public virtual Kho Kho { get; set; }
41
42         0 references
43         public virtual SanPham SanPham { get; set; }
44     }
45
46     0 references
47 }

```

Hình 104. ChiTietKho.cs:

```

1 // </auto-generated>
2 // This code was generated from a template.
3 //
4 //
5 // Manual changes to this file may cause unexpected behavior in your application.
6 // Manual changes to this file will be overwritten if the code is regenerated.
7 // </auto-generated>
8 //
9
10 namespace Test02.Models
11 {
12     using System;
13     using System.Collections.Generic;
14
15     9 references
16     public partial class ChiTietPhieuNhapXuat
17     {
18         3 references
19         public string MaCT { get; set; }
20
21         6 references
22         public string MaSP { get; set; }
23
24         6 references
25         public string MaKho { get; set; }
26
27         4 references
28         public Nullable<int> SoluongDeXuat { get; set; }
29
30         4 references
31         public Nullable<int> SoluongTrongKho { get; set; }
32
33         0 references
34         public string MaPhieu { get; set; }
35
36         0 references
37         public virtual PhieuNhapXuat PhieuNhapXuat { get; set; }
38
39         0 references
40         public virtual SanPham SanPham { get; set; }
41     }
42
43     0 references
44 }

```

Hình 105. ChiTietPhieuNhapXuat.cs

```

1 //<auto-generated>
2 // This code was generated from a template.
3 //
4 // Manual changes to this file may cause unexpected behavior in your application.
5 // Manual changes to this file will be overwritten if the code is regenerated.
6 // </auto-generated>
7
8
9
10 namespace Test02.Models
11 {
12     using System;
13     using System.Collections.Generic;
14
15     public partial class ChucVu
16     {
17         [System.Diagnostics.CodeAnalysis.SuppressMessage("Microsoft.Usage", "CA2214:DoNotCallOverridableMethodsInConstructors")]
18         public ChucVu()
19         {
20             this.NhanViens = new HashSet<NhanVien>();
21         }
22
23         public string MaChucVu { get; set; }
24         public string ChucVu { get; set; }
25         public string ViTri { get; set; }
26
27         [System.Diagnostics.CodeAnalysis.SuppressMessage("Microsoft.Usage", "CA2227:CollectionPropertiesShouldBeReadOnly")]
28         public virtual ICollection<NhanVien> NhanViens { get; set; }
29     }
30 }

```

Hình 106. ChucVu.cs

ChucVu.cs: Thực hiện phân quyền người dùng giữa các user nhân viên tham giao trong hệ thống và phân role theo từng chức năng

```

1 //<auto-generated>
2 // This code was generated from a template.
3 //
4 // Manual changes to this file may cause unexpected behavior in your application.
5 // Manual changes to this file will be overwritten if the code is regenerated.
6 // </auto-generated>
7
8
9
10 namespace Test02.Models
11 {
12     using System;
13     using System.Collections.Generic;
14
15     public partial class ChuyenGiao
16     {
17         [System.Diagnostics.CodeAnalysis.SuppressMessage("Microsoft.Usage", "CA2214:DoNotCallOverridableMethodsInConstructors")]
18
19         public ChuyenGiao()
20         {
21             this.DonHangs = new HashSet<DonHang>();
22             this.NhanVienGhs = new HashSet<NhanVienGh>();
23         }
24
25         public string MaGH { get; set; }
26         public string MaNVlap { get; set; }
27         public string MaPT { get; set; }
28         public Nullable<System.DateTime> NgayGiao { get; set; }
29         public Nullable<double> TienThu { get; set; }
30         public string TranThai { get; set; }
31     }
32 }

```

Hình 107. ChuyenGiao.cs

```

namespace Test02.Models
{
    using System;
    using System.Collections.Generic;

    public partial class Daily
    {
        [System.Diagnostics.CodeAnalysis.SuppressMessage("Microsoft.Usage", "CA2214:DoNotCallOverridableMethodsInConstructors")]
        public Daily()
        {
            this.DonHangs = new HashSet();
            this.PhieuCongNoes = new HashSet<PhieuCongNo>();
        }

        public string MaDL { get; set; }
        public string MaLoadID { get; set; }
        public string Username { get; set; }
        public string Password { get; set; }
        public string TenDL { get; set; }
        public string SDT { get; set; }
        public string DiaChi { get; set; }
        public string Email { get; set; }
        public Nullable NgayLap { get; set; }
        public string HinhAnh { get; set; }
        public Nullable<bool> TongDoanSo { get; set; }

        [System.Diagnostics.CodeAnalysis.SuppressMessage("Microsoft.Usage", "CA2227:CollectionPropertiesShouldBeReadOnly")]
        public virtual ICollection<DonHang> DonHangs { get; set; }
        [System.Diagnostics.CodeAnalysis.SuppressMessage("Microsoft.Usage", "CA2227:CollectionPropertiesShouldBeReadOnly")]
        public virtual ICollection<PhieuCongNo> PhieuCongNoes { get; set; }
    }
}

```

Hình 108. DaiLy.cs

DaiLy.cs: Lưu trữ thông tin khách hàng phục vụ việc lên đơn hàng và kiểm tra thông tin áp dụng cho phòng Kinh Doanh, Đại Lý, Kế Toán, Giao hàng

```

namespace Test02.Models
{
    using System;
    using System.Collections.Generic;

    public partial class DonHang
    {
        [System.Diagnostics.CodeAnalysis.SuppressMessage("Microsoft.Usage", "CA2214:DoNotCallOverridableMethodsInConstructors")]
        public DonHang()
        {
            this.ChiTietDonHangs = new HashSet<ChiTietDonHang>();
        }

        public string MaOH { get; set; }
        public string MaDL { get; set; }
        public string MaVLap { get; set; }
        public string NgayLap { get; set; }
        public string TrangThai { get; set; }
        public string TinhTrangThanhToan { get; set; }
        public string DienLao { get; set; }
        public Nullable<double> TongTien { get; set; }
        public Nullable<bool> PhieuKuatkho { get; set; }
        public Nullable<string> KieuViem { get; set; }
        public string MaHG { get; set; }

        [System.Diagnostics.CodeAnalysis.SuppressMessage("Microsoft.Usage", "CA2227:CollectionPropertiesShouldBeReadOnly")]
        public virtual ICollection<ChiTietDonHang> ChiTietDonHangs { get; set; }
        [System.Diagnostics.CodeAnalysis.SuppressMessage("Microsoft.Usage", "CA2227:CollectionPropertiesShouldBeReadOnly")]
        public virtual ChuyenMiao ChuyenDiao { get; set; }
        public virtual Daily Daily { get; set; }
        public virtual KieuViem NhanVien { get; set; }
    }
}

```

Hình 109. DonHang.cs

DonHang.cs: Thực hiện thêm xóa sửa đơn hàng cho đại lý được áp dụng cho phòng Kinh Doanh , đại lý

```

1 //<auto-generated>
2 // This code was generated from a template.
3 //
4 // Manual changes to this file may cause unexpected behavior in your application.
5 // Manual changes to this file will be overwritten if the code is regenerated.
6 // </auto-generated>
7
8
9 namespace Test02.models
10 {
11     using System;
12     using System.Collections.Generic;
13
14     public partial class Kho
15     {
16         [System.Diagnostics.CodeAnalysis.SuppressMessage("Microsoft.Usage", "CA2214:DoNotCallOverridableMethodsInConstructors")]
17         public Kho()
18         {
19             this.BlenBangKiemKhes = new HashSet<BlenBangKiemKho>();
20             this.ChiTietKhoes = new HashSet<ChiTietKho>();
21             this.PhieuHapXuats = new HashSet<PhieuHapXuat>();
22         }
23
24         5 references
25         public string MaKho { get; set; }
26         1 reference
27         public string TenKho { get; set; }
28         3 references
29         public string DiaChi { get; set; }
30
31         [System.Diagnostics.CodeAnalysis.SuppressMessage("Microsoft.Usage", "CA2227:CollectionPropertiesShouldBeReadOnly")]
32         1 reference
33         public virtual ICollection<BlenBangKiemKho> BlenBangKiemKhes { get; set; }
34         [System.Diagnostics.CodeAnalysis.SuppressMessage("Microsoft.Usage", "CA2227:CollectionPropertiesShouldBeReadOnly")]
35         1 reference
36         public virtual ICollection<ChiTietKho> ChiTietKhoes { get; set; }
37         [System.Diagnostics.CodeAnalysis.SuppressMessage("Microsoft.Usage", "CA2227:CollectionPropertiesShouldBeReadOnly")]
38         1 reference
39         public virtual ICollection<PhieuHapXuat> PhieuHapXuats { get; set; }
40     }
41 }

```

Hình 110. Kho.cs

```

1 //<auto-generated>
2 // This code was generated from a template.
3 //
4 // Manual changes to this file may cause unexpected behavior in your application.
5 // Manual changes to this file will be overwritten if the code is regenerated.
6 // </auto-generated>
7
8
9 namespace Test02.Models
10 {
11     using System;
12     using System.Collections.Generic;
13
14     public partial class LoaiDL
15     {
16         [System.Diagnostics.CodeAnalysis.SuppressMessage("Microsoft.Usage", "CA2214:DoNotCallOverridableMethodsInConstructors")]
17         public LoaiDL()
18         {
19             this.Dailies = new HashSet<Daily>();
20         }
21
22         4 references
23         public string MaLoaiDL { get; set; }
24         0 references
25         public string TenDaily { get; set; }
26         6 references
27         public Nullable<double> ChietKhau { get; set; }
28
29         [System.Diagnostics.CodeAnalysis.SuppressMessage("Microsoft.Usage", "CA2227:CollectionPropertiesShouldBeReadOnly")]
30         1 reference
31         public virtual ICollection<Daily> Dailies { get; set; }
32     }
33 }

```

Hình 111. LoaiDL.cs

LoaiDL.cs: Thực hiện phân các cấp bậc đối với khách hàng có quy mô doanh số lớn và áp dụng chiết khấu dành riêng cho từng loại đại lý.

```

1 //<auto-generated>
2 // This code was generated from a template.
3 //
4 // Manual changes to this file may cause unexpected behavior in your application.
5 // Manual changes to this file will be overwritten if the code is regenerated.
6 // </auto-generated>
7
8
9
10 namespace Test02.Models
11 {
12     using System;
13     using System.Collections.Generic;
14
15     public partial class NhanVien
16     {
17         [System.Diagnostics.CodeAnalysis.SuppressMessage("Microsoft.Usage", "CA2214:DoNotCallOverridableMethodsInConstructors")]
18         public NhanVien()
19         {
20             this.BaoCao = new HashSet<BaoCao>();
21             this.BiemBangKiemKe = new HashSet<BiemBangKiemKe>();
22             this.DonHang = new HashSet<DonHang>();
23             this.PhuieuNhapXuat = new HashSet<PhieuNhapXuat>();
24         }
25
26         13 references
27         public string MaNV { get; set; }
28         public string MaChucVu { get; set; }
29         public string Username { get; set; }
30         public string Password { get; set; }
31         public string TenNV { get; set; }
32         public string SDT { get; set; }
33         public string DiaChi { get; set; }
34         public string Email { get; set; }
35         public string HinhAnh { get; set; }
36         public Nullable<System.DateTime> NgaySinh { get; set; }
37         public string GioiTinh { get; set; }
38
39         [System.Diagnostics.CodeAnalysis.SuppressMessage("Microsoft.Usage", "CA2227:CollectionPropertiesShouldBeReadOnly")]
40         public virtual ICollection<BaoCao> BaoCao { get; set; }
41         [System.Diagnostics.CodeAnalysis.SuppressMessage("Microsoft.Usage", "CA2227:CollectionPropertiesShouldBeReadOnly")]
42         public virtual ICollection<BiemBangKiemKe> BiemBangKiemKe { get; set; }
43         [System.Diagnostics.CodeAnalysis.SuppressMessage("Microsoft.Usage", "CA2227:CollectionPropertiesShouldBeReadOnly")]
44         public virtual ICollection<DonHang> DonHang { get; set; }
45         [System.Diagnostics.CodeAnalysis.SuppressMessage("Microsoft.Usage", "CA2227:CollectionPropertiesShouldBeReadOnly")]
46         public virtual ICollection<PhieuNhapXuat> PhieuNhapXuat { get; set; }
47     }
48 }

```

Hình 112. NhanVien.cs

NhanVien.cs: Thực hiện lưu trữ, quản lý thông tin nhân viên trong hệ thống

```

1 //<auto-generated>
2 // This code was generated from a template.
3 //
4 // Manual changes to this file may cause unexpected behavior in your application.
5 // Manual changes to this file will be overwritten if the code is regenerated.
6 // </auto-generated>
7
8
9
10 namespace Test02.Models
11 {
12     using System;
13     using System.Collections.Generic;
14
15     public partial class NhanVienGH
16     {
17         6 references
18         public string MaNV { get; set; }
19         0 references
20         public string TenNV { get; set; }
21         0 references
22         public string SDT { get; set; }
23         5 references
24         public string DiaChi { get; set; }
25         2 references
26         public string TinhTrang { get; set; }
27         0 references
28         public string MaGH { get; set; }
29
30         0 references
31         public virtual ChuyenGiao ChuyenGiao { get; set; }
32     }
33 }

```

Hình 113. NhanVienGH.cs

```

1 //-----
2 // <auto-generated>
3 //   This code was generated from a template.
4 //
5 //   Manual changes to this file may cause unexpected behavior in your application.
6 //   Manual changes to this file will be overwritten if the code is regenerated.
7 // </auto-generated>
8 //-----
9
10 namespace Test02.Models
11 {
12     using System;
13     using System.Collections.Generic;
14
15     public partial class NhanVienGH
16     {
17         6 references
18         public string MaNV { get; set; }
19         0 references
20         public string TenNV { get; set; }
21         0 references
22         public string SDT { get; set; }
23         5 references
24         public string DiaChi { get; set; }
25         2 references
26         public string TinhTrang { get; set; }
27         0 references
28         public string MagH { get; set; }
29
30         0 references
31         public virtual ChuyenGiao ChuyenGiao { get; set; }
32     }
33 }
34
35
36
37
38

```

Hình 114. PhieuCongNo.cs

```

10 namespace Test02.Models
11 {
12     using System;
13     using System.Collections.Generic;
14
15     public partial class PhieuNhapXuat
16     {
17         [System.Diagnostics.CodeAnalysis.SuppressMessage("Microsoft.Usage", "CA2214:DoNotCallOverridableMethodsInConstructors")]
18         0 references
19         public PhieuNhapXuat()
20         {
21             this.ChiTietPhieuNhapXuats = new HashSet<ChiTietPhieuNhapXuat>();
22         }
23
24         8 references
25         public string MaPhiieu { get; set; }
26         3 references
27         public string MaKho { get; set; }
28         5 references
29         public Nullable<System.DateTime> NgayLap { get; set; }
30         3 references
31         public string LoaiPhiieu { get; set; }
32         1 reference
33         public string MaNVlap { get; set; }
34         1 reference
35         public string NguoigiaoHang { get; set; }
36         1 reference
37         public string PhuongTienGiaoHang { get; set; }
38         5 references
39         public string TinhTrang { get; set; }
40         7 references
41         public string MaDH { get; set; }
42
43         [System.Diagnostics.CodeAnalysis.SuppressMessage("Microsoft.Usage", "CA2227:CollectionPropertiesShouldBeReadOnly")]
44         1 reference
45         public virtual ICollection<ChiTietPhieuNhapXuat> ChiTietPhieuNhapXuats { get; set; }
46         0 references
47         public virtual Kho Kho { get; set; }
48         1 reference
49         public virtual NhanVien NhanVien { get; set; }
50     }
51 }
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
79
80
81
82
83
84
85
86
87
88
89
89
90
91
92
93
94
95
96
97
98
99
99
100
101
102

```

Hình 115. PhieuNhapXuat.cs

```

PhuongTienGH.cs
Test02
7 // </auto-generated>
8
9
10 namespace Test02.Models
11 {
12     using System;
13     using System.Collections.Generic;
14
15     8 references
16     public partial class PhuongTienGH
17     {
18         [System.Diagnostics.CodeAnalysis.SuppressMessage("Microsoft.Usage", "CA2214:DoNotCallOverridableMethodsInConstructors")]
19         0 references
20         public PhuongTienGH()
21         {
22             this.ChuyenGiao = new HashSet<ChuyenGiao>();
23         }
24
25         7 references
26         public string MaPT { get; set; }
27
28         0 references
29         public string TenPT { get; set; }
30
31         6 references
32         public string TinhTrang { get; set; }
33
34         0 references
35         public Nullable<double> TrongTai { get; set; }
36
37         [System.Diagnostics.CodeAnalysis.SuppressMessage("Microsoft.Usage", "CA2227:CollectionPropertiesShouldBeReadOnly")]
38         1 reference
39         public virtual ICollection<ChuyenGiao> ChuyenGiao { get; set; }
40
41     }
42
43     7 references
44 }

```

Server Explorer, Solution Explorer, Git Changes, Solution Explorer, Properties.

Hình 116. PhuongTienGH.cs

```

SanPham.cs
Test02
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42

```

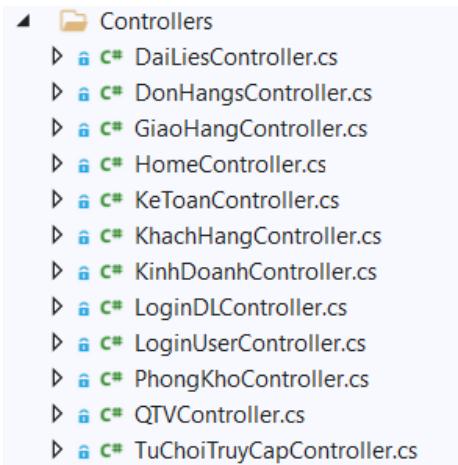
Server Explorer, Solution Explorer, Git Changes, Solution Explorer, Properties.

Hình 117. SanPham.cs

SanPham.cs: Thực hiện lưu trữ và quản lý thông tin sản phẩm đang tồn tại trong hệ thống

## 4.2. CONTROLLERS

### 4.2.1. Module tổng quan



Danh sách toàn bộ các controller phục vụ đáp ứng các yêu cầu nghiệp vụ trong việc quản lý đại lý.

### 4.2.2. Chi tiết controller theo phòng ban

#### 4.2.2.1. Phòng Kinh Doanh:

```

14  namespace Test02.Controllers
15  {
16      [Authentication (MaChucVu ="NVKD")]
17      public class KinhDoanhController : Controller
18      {
19
20          QuanLyDLEntities2 database = new QuanLyDLEntities2();
21          // GET: KinhDoanh
22          public ActionResult TrangChu()
23          {
24              return View();
25          }
26
27          public ActionResult QuanLyDL()
28          {
29              return View(database.Dailies.ToList().OrderByDescending(s=> s.NgayTao));
30          }
31          //Kiểm tra tên đại lý
32          [HttpPost]
33          public ActionResult KTTenDL(string TenDL)
34          {
35              var check = database.Dailies.Where(s => s.TenDL == TenDL).FirstOrDefault();
36              if(check != null)
37              {
38                  return Json(new { success = false}, JsonRequestBehavior.AllowGet);
39              }
39
40              else
41              {
42                  return Json(new { success = true }, JsonRequestBehavior.AllowGet);
43              }
44          }
}

```

Action Method :

\_TrangChu (View TrangChu)

\_QuanLyDL (View QuanLyDL)

\_KTTenDL (Trả về dạng chuỗi Json khi hàm gọi Ajax được thực thi từ phía client)

```

        [HttpPost]
        public ActionResult KTSDT(string SDT)
        {
            var check = database.Dailies.Where(s => s.SDT == SDT).FirstOrDefault();
            if (check != null)
            {
                return Json(new { success = false }, JsonRequestBehavior.AllowGet);
            }
            else
            {
                return Json(new { success = true }, JsonRequestBehavior.AllowGet);
            }
        }
        [HttpPost]
        public ActionResult KTEmail(string Email)
        {
            if (Email.IndexOf("@gmail.com") > 0)
            {
                var check = database.Dailies.Where(s => s.Email == Email).FirstOrDefault();
                if (check != null)
                {
                    return Json(new { success = false }, JsonRequestBehavior.AllowGet);
                }
                else
                {
                    return Json(new { success = true }, JsonRequestBehavior.AllowGet);
                }
            }
            else
            {
                return Json(new { success = false }, JsonRequestBehavior.AllowGet);
            }
        }
    }
}

```

#### Action Method :

- \_ KTSDT (trả về kết quả khi kiểm tra số điện thoại của đại lý theo dạng Json)
- \_ KTEmail (Trả về kết quả khi kiểm tra email của đại lý theo dạng Json)

```

        }
        public ActionResult ThemDL()
        {
            ViewBag.MaLoaiDL = new SelectList(database.LoaiDLs, "MaLoaiDL", "TenDaily");
            return View();
        }

        public ActionResult ChinhSuaDL(string id)
        {
            if (id == null)
            {
                return new HttpStatusCodeResult(System.Net.HttpStatusCode.BadRequest);
            }
            Daily daily = database.Dailies.Find(id);
            if (daily == null)
            {
                return HttpNotFound();
            }
            ViewBag.MaLoaiDL = new SelectList(database.LoaiDLs, "MaLoaiDL", "TenDaily", daily.MaLoaiDL);
            return View(daily);
        }
        [HttpPost]
        [ValidateAntiForgeryToken]
        public ActionResult ChinhSuaDL([Bind(Include = "MaDL,MaLoaiDL,UserName,Password,TenDL,SDT,DiaChi,Email,NgayTao")] Daily daily)
        {
            if (ModelState.IsValid)
            {
                database.Entry(daily).State = System.Data.Entity.EntityState.Modified;
                database.SaveChanges();
                return RedirectToAction("QuanLyDL");
            }
            ViewBag.MaLoaiDL = new SelectList(database.LoaiDLs, "MaLoaiDL", "TenDaily", daily.MaLoaiDL);
            return View(daily);
        }
        [HttpPost]
        public ActionResult ResetPass(string madl)
        {
            if (madl == null)
            {

```

#### Action Method:

- \_ ThemDL (View ThemDL)
- \_ ChinhSuaDL (View ChinhSuaDL )
- \_ Post : ChinhSuaDL (Trả về View ChinhSuaDL sau khi đã xử lý chỉnh sửa xong)

```

1 public ActionResult ResetPass(string madl)
2 {
3     if (madl == null)
4     {
5         return Json(new { success = false }, JsonRequestBehavior.AllowGet);
6     }
7     else
8     {
9         var resetpass = database.Dailies.Find(madl);
10        resetpass.Password = "123456";
11        database.Entry(resetpass).State = System.Data.Entity.EntityState.Modified;
12        database.SaveChanges();
13        TempData["madl"] = madl;
14        //Gửi email về tk
15        var daily = database.Dailies.Find(madl);
16        string content = System.IO.File.ReadAllText(Server.MapPath("~/TemplateMail/ResetPass.html"));
17        content = content.Replace("{{TenDL}}", daily.Email);
18        content = content.Replace("{{username}}", daily.UserName);
19        content = content.Replace("{{time}}", DateTime.Now.ToString());
20        //content = content.Replace("{{Total}}", tongtien);
21        //content = content.Replace("{{Thoigian}}", Convert.ToString(ttkh.CreateDate));
22        //content = content.Replace("{{Invoice}}", themhd.MaHD);
23        string subject = "Tài khoản đại lý " + daily.MaDL + " của bạn vừa được cập nhật !";
24        WebMail.Send(daily.Email, subject, content, null, null, true, null, null, null, null, null, null);
25        return Json(new { success = true }, JsonRequestBehavior.AllowGet);
26    }
27 }
28 [HttpPost]
29 [ValidateAntiForgeryToken]
30 public ActionResult ThemDL([Bind(Include = "MaLoaiDL,UserName,Password,TenDL,SDT,DiaChi,Email")] Daily daily)
31 {
32     if (ModelState.IsValid)
33     {
34         Random rd = new Random();
35         var madl = "DL" + rd.Next(0, 10) + rd.Next(0, 10) + rd.Next(0, 10) + rd.Next(0, 10);
36         daily.MaDL = madl;
37         daily.NgayTao = System.DateTime.Now;
38         daily.Password = "123456";
39         database.Dailies.Add(daily);
40     }
41 }

```

### Action Method:

\_ Post: ResetPass (Trả về dạng chuỗi Json liên quan đến view ChiTietDL)

```

1 [HttpPost]
2 [ValidateAntiForgeryToken]
3 public ActionResult ThemDL([Bind(Include = "MaLoaiDL,UserName,Password,TenDL,SDT,DiaChi,Email")] Daily daily)
4 {
5     if (ModelState.IsValid)
6     {
7         Random rd = new Random();
8         var madl = "DL" + rd.Next(0, 10) + rd.Next(0, 10) + rd.Next(0, 10) + rd.Next(0, 10);
9         daily.MaDL = madl;
10        daily.NgayTao = System.DateTime.Now;
11        daily.Password = "123456";
12        database.Dailies.Add(daily);
13        database.SaveChanges();
14        TempData["messageAlert"] = "Đã thêm mới";
15        TempData["TenDLthem"] = daily.TenDL;
16        //Gửi email về tk
17        string content = System.IO.File.ReadAllText(Server.MapPath("~/TemplateMail/AddAccount.html"));
18        content = content.Replace("{{email}}", daily.Email);
19        content = content.Replace("{{Password}}", daily.Password);
20        content = content.Replace("{{TenDL}}", daily.TenDL);
21        //content = content.Replace("{{Total}}", tongtien);
22        //content = content.Replace("{{Thoigian}}", Convert.ToString(ttkh.CreateDate));
23        //content = content.Replace("{{Invoice}}", themhd.MaHD);
24        string subject = "Đây là tin nhắn tự động từ hệ thống POS";
25        WebMail.Send(daily.Email, subject, content, null, null, true, null, null, null, null, null, null);
26        return RedirectToAction("QuanLyDL");
27    }
28
29 ViewBag.MaLoaiDL = new SelectList(database.LoaiDLs, "MaLoaiDL", "TenDaily", daily.MaLoaiDL);
30 return View(daily);
31 }
32 // GET: Dailies/Delete/5
33 public ActionResult Delete(string id)
34 {
35     if (id == null)
36     {
37         return new HttpStatusCodeResult(System.Net.HttpStatusCode.BadRequest);
38     }
39     Daily daily = database.Dailies.Find(id);
40     if (daily == null)
41     {
42     }
43 }

```

### Action Method:

\_ Post: ThemDL (Liên quan đến view ThemDL )

\_ Delete (Thực hiện xóa đại lý trên view Delete)

```

// GET: Dailies/Delete/5
public ActionResult Delete(string id)
{
    if (id == null)
    {
        return new HttpStatusCodeResult(System.Net.HttpStatusCode.BadRequest);
    }
    Daily daily = database.Dailies.Find(id);
    if (daily == null)
    {
        return HttpNotFound();
    }
    return View(daily);
}

// POST: Dailies/Delete/5
[HttpPost, ActionName("Delete")]
[ValidateAntiForgeryToken]
public ActionResult DeleteConfirmed(string id)
{
    Daily daily = database.Dailies.Find(id);
    database.Dailies.Remove(daily);
    database.SaveChanges();
    //Gửi email về tk
    string content = System.IO.File.ReadAllText(Server.MapPath("~/TemplateMail/DeleteAccount.html"));
    content = content.Replace("{{TenDL}}", daily.TenDL);
    content = content.Replace("{{username}}", daily.UserName);
    content = content.Replace("{{time}}", DateTime.Now.ToString());
    //content = content.Replace("{{Total}}", tongtien);
    //content = content.Replace("{{Thoigian}}", Convert.ToString(ttkh.CreateDate));
    //content = content.Replace("{{Invoice}}", themhd.MaHD);
    string subject = "BẢN ĐÃ XÓA TÀI KHOẢN ĐẠI LÝ";
    WebMail.Send(daily.Email, subject, content, null, null, null, true, null, null, null, null, null, null);
    return RedirectToAction("QuanLyDL");
}

```

### Action Method:

- \_ Post: Delete (Thực hiện xóa đại lý khi được submit và trả về view QuanLyDL khi hoàn tất)

```

public ActionResult QuanLyDH()
{
    return View(database.DonHangs.ToList().OrderBy(s => s.NgayLap));
}
[HttpPost]
public ActionResult CheckCongNoDL(string MaDL, string ThanhTien)
{
    var tongtiendh = 0;
    var daily = database.Dailies.Where(s => s.MaDL == MaDL).FirstOrDefault();
    tongtiendh += Convert.ToInt32(ThanhTien);
    if (daily.MaLoaiDL == "LDL01" && tongtiendh <= 50000000)
    {
        return Json(new { success = true }, JsonRequestBehavior.AllowGet);
    }
    else if (daily.MaLoaiDL == "LDL02" && tongtiendh <= 25000000)
    {
        return Json(new { success = true }, JsonRequestBehavior.AllowGet);
    }
    else if (daily.MaLoaiDL == "LDL03" && tongtiendh <= 10000000)
    {
        return Json(new { success = true }, JsonRequestBehavior.AllowGet);
    }
    else
    {
        return Json(new { success = false }, JsonRequestBehavior.AllowGet);
    }
}
//Thêm mới đơn hàng
public ActionResult ThemDH()
{
    ViewBag.MaDL = new SelectList(database.Dailies, "MaDL", "MaDL");
    return View();
}

```

### Action Method:

- \_ QuanLyDH (Trả về danh sách đơn hàng trên view QuanLyDH)
- \_ CheckCongNoDL (Trả về dạng chuỗi json khi có lời gọi Ajax gọi tới )

### \_ThemDH (Mở trang ThemDH trên view ThemDH)

```

}
//Thêm mới đơn hàng
public ActionResult ThemDH()
{
    ViewBag.MaDL = new SelectList(database.DaiLies, "MaDL", "MaDL");
    return View();
}
[HttpPost]
[ValidateAntiForgeryToken]
public ActionResult ThemDH([Bind(Include = "MaDH,MaDL,NgayLap,DiemGiao")] DonHang donHang)
{
    DataTable dh = new DataTable();
    if (Session["DonHang"] == null)
    {
        dh.Columns.Add("MaDH");
        dh.Columns.Add("MaDL");
        dh.Columns.Add("MaNVLap");
        dh.Columns.Add("NgayLap");
        dh.Columns.Add("TrangThai");
        dh.Columns.Add("TinhTrangThanhToan");
        dh.Columns.Add("DiemGiao");
        //Tạo xong lưu lại Session
        Session["DonHang"] = dh;
    }
    else
    {
        //Lấy thông tin từ Session
        dh = Session["DonHang"] as DataTable;
    }
    if (ModelState.IsValid)
    {
        var user = (Test02.Models.NhanVien)Session["user"];
        Random rd = new Random();
        var madh = "DH" + rd.Next(1, 1000);
        donHang.MaDH = madh;
        Session["tempdata"] = madh;
        Session["diemgiao"] = donHang.DiemGiao;
        donHang.NgayLap = System.DateTime.Now;
        donHang.MaNVLap = user.MaNV;
    }
}

```

Action Method Post: ThemDH (Thực hiện thêm đơn hơn khi view ThemDH gửi submit)

```

}
if (ModelState.IsValid)
{
    var user = (Test02.Models.NhanVien)Session["user"];
    Random rd = new Random();
    var madh = "DH" + rd.Next(1, 1000);
    donHang.MaDH = madh;
    Session["tempdata"] = madh;
    Session["diemgiao"] = donHang.DiemGiao;
    donHang.NgayLap = System.DateTime.Now;
    donHang.MaNVLap = user.MaNV;
    donHang.TinhTrangThanhToan = "Chưa thanh toán";
    donHang.TrangThai = "Chưa xử lý";
    //Lưu thông tin đơn hàng vào bảng DataTable
    DataRow dr = dh.NewRow();
    dr["MaDH"] = donHang.MaDH;
    dr["MaDL"] = donHang.MaDL;
    dr["MaNVLap"] = donHang.MaNVLap;
    dr["NgayLap"] = donHang.NgayLap;
    dr["TrangThai"] = donHang.TrangThai;
    dr["TinhTrangThanhToan"] = donHang.TinhTrangThanhToan;
    dr["DiemGiao"] = donHang.DiemGiao;
    dh.Rows.Add(dr);

    //database.DonHangs.Add(donHang);
    //database.SaveChanges();
    return RedirectToAction("ThemCTHD");
}

return View(donHang);
}
[HttpPost]
[ValidateAntiForgeryToken]
public ActionResult ThemDL(LoaiDL loaiDL)
{
    Random rd = new Random();
    var madl = "LDL" + rd.Next(1, 1000);
    loaiDL.MaLoaiDL = madl;
    database.LoaiDLs.Add(loaiDL);
}

```

Action Method Post: ThemDL : (Thực hiện thêm Đại lý khi view ThemDL gửi yêu cầu submit)

```
[HttpPost]
[ValidateAntiForgeryToken]
public ActionResult ThemDL(LoaiDL loaiDL)
{
    Random rd = new Random();
    var madl = "DL" + rd.Next(1, 1000);
    loaiDL.MaloaiDL = madl;
    database.LoaiDLs.Add(loaiDL);
    database.SaveChanges();
    TempData["messageAlert"] = "Đã thêm mới loại đại lý";
    return RedirectToAction("ThemDL");
}
//Thêm mới chi tiết đơn hàng bán tự động
public ActionResult ThemCTHD()
{
    ViewBag.MaDH = new SelectList(database.DonHangs, "MaDH", "MaDL");
    ViewBag.MaSP = new SelectList(database.SanPhams, "MaSP", "TenSP");
    return View();
}
[HttpPost]
[ValidateAntiForgeryToken]
public ActionResult ThemCTHD([Bind(Include = "MaSP,Soluong,DongGia,DiemGiao,DonViTinh")] ChiTietDonHang chiTietDonHang)
{
    if (ModelState.IsValid)
    {
        //Thêm thông tin đơn hàng sau khi đã điền chi tiết đơn hàng

        LuudhvaoDB();
        //Thêm chi tiết đơn hàng
        chiTietDonHang.MaDH = (string)Session["tempdata"];
        var dongia = database.SanPhams.Where(s => s.MaSP == chiTietDonHang.MaSP).FirstOrDefault();
        if (chiTietDonHang.ChietKhuu == null)
        {
            chiTietDonHang.ThanhTien = (dongia.Gia) * (chiTietDonHang.Soluong);
        }
        else
        {
            chiTietDonHang.ThanhTien = (chiTietDonHang.Soluong) * (dongia.Gia) * (chiTietDonHang.ChietKhuu);
        }
        var checkslp = database.SanPhams.Where(s => s.MaSP == chiTietDonHang.MaSP).FirstOrDefault();
    }
}
```

### Action Method:

- \_ ThemCTHD : Thực hiện mở trang ThemCTHD
- \_ Post: ThemCTHD: Thực hiện thêm chi tiết hóa đơn khi người dùng gửi yêu cầu submit từ trang ThemCTHD

```

        [HttpPost]
        [ValidateAntiForgeryToken]
        public ActionResult ThemCTHD([Bind(Include = "MaSP,SoLuong,DonGia,DiemGiao,DonViTinh")] ChiTietDonHang chiTietDonHang)
        {
            if (ModelState.IsValid)
            {
                //Thêm thông tin đơn hàng sau khi đã điền chi tiết đơn hàng

                LuudhvaoDB();
                //Thêm chi tiết đơn hàng
                chiTietDonHang.MaDH = (string)Session["tempdata"];
                var dongia = database.SanPhams.Where(s => s.MaSP == chiTietDonHang.MaSP).FirstOrDefault();
                if (chiTietDonHang.ChietKhau == null)
                {
                    chiTietDonHang.ThanhTien = (dongia.Gia) * (chiTietDonHang.SoLuong);
                }
                else
                {
                    chiTietDonHang.ThanhTien = (chiTietDonHang.SoLuong) * (dongia.Gia) * (chiTietDonHang.ChietKhau);
                }
                var checkslp = database.SanPhams.Where(s => s.MaSP == chiTietDonHang.MaSP).FirstOrDefault();
                if (chiTietDonHang.SoLuong > checkslp.TongTon)
                {
                    var madh = database.DonHangs.Where(s=> s.MaDH == chiTietDonHang.MaDH).FirstOrDefault();
                    TempData["messageAlert"] = "Không đủ số lượng để đặt";
                    database.DonHangs.Remove(madh);
                    database.SaveChanges();
                    return RedirectToAction("QuanLyDH");
                }
                else
                {
                    var ctk = database.ChiTietKhoses.Where(s => s.MaSP == chiTietDonHang.MaSP).ToList();
                    var slsp = (int)chiTietDonHang.SoLuong;
                    var checksl = -slsp;
                    foreach (var udk in ctk)
                    {
                        checksl += (int)udk.SoLuong;
                        if(checksl>=0)
                    }
                }
            }
            else
            {
                var ctk = database.ChiTietKhoses.Where(s => s.MaSP == chiTietDonHang.MaSP).ToList();
                var slsp = (int)chiTietDonHang.SoLuong;
                var checksl = -slsp;
                foreach (var udk in ctk)
                {
                    checksl += (int)udk.SoLuong;
                    if(checksl>=0)
                    {
                        udk.SoLuong = checksl;
                        if (udk.SoLuong < 1500)
                        {
                            udk.TinhTrang = "Sắp hết hàng";
                        }
                        database.Entry(udk).State = System.Data.Entity.EntityState.Modified;
                        database.SaveChanges();
                        break;
                    }
                    else if (checksl < 0)
                    {
                        udk.TinhTrang = "Hết hàng";
                        udk.SoLuong = 0;
                        database.Entry(udk).State = System.Data.Entity.EntityState.Modified;
                        database.SaveChanges();
                    }
                }
                database.ChiTietDonHangs.Add(chiTietDonHang);
                database.SaveChanges();
                //Thêm tổng tiền của đơn hàng
                var dh = database.DonHangs.Find(chiTietDonHang.MaDH);
                dh.TongTien = chiTietDonHang.ThanhTien;
                database.Entry(dh).State = System.Data.Entity.EntityState.Modified;
                database.SaveChanges();

                TempData["messageAlert"] = "Đã thêm mới đơn hàng";
                TempData["themmadh"] = chiTietDonHang.MaDH;
                return RedirectToAction("QuanLyDH");
            }
        }
    
```

```

        if(checksl>=0)
        {
            udk.SoLuong = checksl;
            if (udk.SoLuong < 1500)
            {
                udk.TinhTrang = "Sắp hết hàng";
            }
            database.Entry(udk).State = System.Data.Entity.EntityState.Modified;
            database.SaveChanges();
            break;
        }
        else if (checksl <0)
        {
            udk.TinhTrang = "Hết hàng";
            udk.SoLuong = 0;
            database.Entry(udk).State = System.Data.Entity.EntityState.Modified;
            database.SaveChanges();
        }
    }
    database.ChiTietDonHangs.Add(chiTietDonHang);
    database.SaveChanges();
    //Thêm tổng tiền của đơn hàng
    var dh = database.DonHangs.Find(chiTietDonHang.MaDH);
    dh.TongTien = chiTietDonHang.ThanhTien;
    database.Entry(dh).State = System.Data.Entity.EntityState.Modified;
    database.SaveChanges();

    TempData["messageAlert"] = "Đã thêm mới đơn hàng";
    TempData["themmadh"] = chiTietDonHang.MaDH;
    return RedirectToAction("QuanLyDH");
}

ViewBag.MaSP = new SelectList(database.SanPhams, "MaSP", "TenSP", chiTietDonHang.MaSP);
return View(chiTietDonHang);
}

//Version 2.0 mới
[HttpPost]
public ActionResult ThemDH02(DonHang dh)
{
    int id = int.Parse(Request["sl"]);
    if (id > 0)
    {
        var user = (Test02.Models.NhanVien)Session["user"];
        Random rd = new Random();
        var madh = "DH" + rd.Next(1, 1000);
        dh.MaDH = madh;
        dh.NgayLap = System.DateTime.Now;
        dh.MaNVLap = user.MaNV;
        dh.TinhTrangThanhToan = "Chưa thanh toán";
        dh.TrangThai = "Chưa xử lý";
        dh.TongTien = 0;
        var ldl = database.DaiLies.Where(s => s.MaDL == dh.MaDL).FirstOrDefault();
        dh.DiemGiao = ldl.DiaChi;
        dh.PhieuXuatKho = false;
        dh.XuatHoaDon = false;
        dh.TinhTrangGH = null;
        database.DonHangs.Add(dh);
        database.SaveChanges();
        var loadl = database.DaiLies.Find(dh.MaDL);
        var ck = database.LoaiDLs.Where(s => s.MaLoaiDL == loadl.MaLoaiDL).FirstOrDefault();

        for (int i = 1; i <= id; i++)
        {
            ChiTietDonHang ctdh = new ChiTietDonHang();
            ctdh.MaCTDH = rd.Next(0, 10000);
            ctdh.MaDH = dh.MaDH;
            ctdh.MaSP = Request["MaSP"+i];
            ctdh.SoLuong = Convert.ToInt32(Request["SoLuong" + i]);
            //ctdh.ChiếtKhau = ck.ChiếtKhau;
            var dg = database.SanPhams.Where(s => s.MaSP == ctdh.MaSP).FirstOrDefault();
            if (ctdh.ChiếtKhau == null)
            {
                ctdh.ThanhTien = (dg.Gia) * (ctdh.SoLuong);
            }
        }
    }
}

```

```

var dg = database.SanPhams.Where(s => s.MaSP == ctdh.MaSP).FirstOrDefault();
if (ctdh.ChietKhau == null)
{
    ctdh.ThanhTien = (dg.Gia) * (ctdh.SoLuong);
}
//else
//{
//    ctdh.ThanhTien = (ctdh.SoLuong) * (dg.Gia) * (ctdh.ChietKhau);
//}
//Kiểm tra số lượng sản phẩm tồn trong kho
var checkslp = database.SanPhams.Where(s => s.MaSP == ctdh.MaSP).FirstOrDefault();
if (ctdh.SoLuong > checkslp.TongTon)
{
    database.DonHangs.Remove(dh);
    TempData["messageAlert"] = "Không đủ số lượng để đặt";
    return RedirectToAction("QuanLyDH");
}
else
{
    //var ctk = database.ChiTietKhoes.Where(s => s.MaSP == ctdh.MaSP).ToList();
}
dh.TongTien += ctdh.ThanhTien;
database.ChiTietDonHangs.Add(ctdh);
database.SaveChanges();
}
dh.TongTien = dh.TongTien - (dh.TongTien * ck.ChietKhau);
if (ldl.MaLoaiDL == "LDL01")
{
    if (dh.TongTien > 50000000)
    {
        TempData["messageAlert"] = "error01";
        TempData["themmadh"] = dh.MaDH;
        TempData["madl"] = dh.MaDL;
        database.DonHangs.Remove(dh);
        database.SaveChanges();
        return RedirectToAction("QuanLyDH");
    }
}
else if (ldl.MaLoaiDL == "LDL02")
{
    if (dh.TongTien > 25000000)
    {
        TempData["messageAlert"] = "error01";
        TempData["themmadh"] = dh.MaDH;
        TempData["madl"] = dh.MaDL;
        database.DonHangs.Remove(dh);
        database.SaveChanges();
        return RedirectToAction("QuanLyDH");
    }
}
else if (ldl.MaLoaiDL == "LDL03")
{
    if (dh.TongTien > 10000000)
    {
        TempData["messageAlert"] = "error01";
        TempData["themmadh"] = dh.MaDH;
        TempData["madl"] = dh.MaDL;
        database.DonHangs.Remove(dh);
        database.SaveChanges();
        return RedirectToAction("QuanLyDH");
    }
}

```

```

        }

        TempData["messageAlert"] = "Đã thêm mới đơn hàng";
        TempData["themmadh"] = dh.MaDH;

        //Gửi email đã lên đơn hàng thành công
        //Lấy thông tin kh
        var ttkh = database.Dailies.Find(dh.MaDL);
        var tongtien = string.Format("{0:0,0 vnđ}", dh.TongTien);
        //Lấy thông tin IP
        string ipAddress = Request.UserHostAddress;
        string content = System.IO.File.ReadAllText(Server.MapPath("~/TemplateMail/NewOrder.html"));
        content = content.Replace("{{Username}}", ttkh.UserName);
        content = content.Replace("{{IdOrder}}", dh.MaDH);
        content = content.Replace("{{datetime}}", DateTime.Now.ToString());
        content = content.Replace("{{TenDL}}", ttkh.TenDL);
        content = content.Replace("{{MyIP}}", ipAddress);
        content = content.Replace("{{emailntt}}", ttkh.Email);

    }

    TempData["messageAlert"] = "Đã thêm mới đơn hàng";
    TempData["themmadh"] = dh.MaDH;

    //Gửi email đã lên đơn hàng thành công
    //Lấy thông tin kh
    var ttkh = database.Dailies.Find(dh.MaDL);
    var tongtien = string.Format("{0:0,0 vnđ}", dh.TongTien);
    //Lấy thông tin IP
    string ipAddress = Request.UserHostAddress;
    string content = System.IO.File.ReadAllText(Server.MapPath("~/TemplateMail/NewOrder.html"));
    content = content.Replace("{{Username}}", ttkh.UserName);
    content = content.Replace("{{IdOrder}}", dh.MaDH);
    content = content.Replace("{{datetime}}", DateTime.Now.ToString());
    content = content.Replace("{{TenDL}}", ttkh.TenDL);
    content = content.Replace("{{MyIP}}", ipAddress);
    content = content.Replace("{{emailntt}}", ttkh.Email);
    content = content.Replace("{{email}}", ttkh.Email);
    content = content.Replace("{{TinhTrang}}", dh.TrangThai);
    content = content.Replace("{{SDT}}", ttkh.SDT);
    content = content.Replace("{{Tongtien}}", tongtien);
    content = content.Replace("{{chietkhai}}", ck.ChietKhau.ToString());
    content = content.Replace("{{DCGH}}", ttkh.DiaChi);
    //content = content.Replace("{{Total}}", tongtien);
    //content = content.Replace("{{Thoigian}}", Convert.ToString(ttkh.CreateDate));
    //content = content.Replace("{{Invoice}}", themhd.MaHD);
    string subject = "Đại lý phân phối nước ngọt - Đơn hàng mới " + dh.MaDH;
    WebMail.Send(ttkh.Email, subject, content, null, null, true, null, null, null, null, null, null);
    return RedirectToAction("QuanLyDH");
}

TempData["messageAlert"] = "Không đủ số lượng để đặt";
return RedirectToAction("QuanLyDH");
}
public void LuudhvaoDB()
{
    DonHang donHang = new DonHang();

```

```

    public void LuudhvaoDB()
    {
        DonHang donHang = new DonHang();

        DataTable dh = Session["DonHang"] as DataTable;
        foreach (DataRow dr in dh.Rows)
        {
            donHang.MaDH = dr["MaDH"].ToString();
            donHang.MaDL = dr["MaDL"].ToString();
            donHang.MaNVLap = dr["MaNVLap"].ToString();
            var date = Convert.ToDateTime(dr["NgayLap"]);
            donHang.NgayLap = date;
            donHang.TrangThai = dr["TrangThai"].ToString();
            donHang.TinhTrangThanhToan = dr["TinhTrangThanhToan"].ToString();
            donHang.DiemGiao = dr["DiemGiao"].ToString();
        }
        database.DonHangs.Add(donHang);
        database.SaveChanges();
    }
}

```

#### Action Method:

- \_ LuudhvaoDB : Thực hiện thêm mới đơn hàng và sau đó sẽ trở về màn hình QuanLyDH

```

public ActionResult ChinhSuaCTDH(string id)
{
    if (id == null)
    {
        return new HttpStatusCodeResult(System.Net.HttpStatusCode.BadRequest);
    }
    var madh = database.ChiTietDonHangs.Where(s => s.MaSP == id).FirstOrDefault();
    if (madh == null)
    {
        return HttpNotFound();
    }
    Session["MaCTDH"] = madh.MaCTDH;
    Session["madh"] = madh.MaDH;
    ViewBag.MaSP = new SelectList(database.SanPhams, "MaSP", "TenSP");
    return View(madh);
}
// POST: ChiTietDonHangs/Edit/5
// To protect from overposting attacks, enable the specific properties you want to bind to, for
// more details see https://go.microsoft.com/fwlink/?LinkId=317598.
[HttpPost]
[ValidateAntiForgeryToken]
public ActionResult ChinhSuaCTDH([Bind(Include = "MaSP,SoLuong,DonGia,DiemGiao,DonViTinh")] ChiTietDonHang chiTietDonHang)
{
    if (ModelState.IsValid)
    {
        var dongia = database.SanPhams.Where(s => s.MaSP == chiTietDonHang.MaSP).FirstOrDefault();
        chiTietDonHang.MaCTDH = (int)Session["MaCTDH"];
        if (chiTietDonHang.ChietKhau == null)
        {
            chiTietDonHang.ThanhTien = (chiTietDonHang.SoLuong) * (dongia.Gia);
        }
        else
        {
            chiTietDonHang.ThanhTien = (chiTietDonHang.SoLuong) * (dongia.Gia) * (chiTietDonHang.ChietKhau);
        }
        chiTietDonHang.MaDH = (string)Session["madh"];
        TempData["messageAlert"] = "Đã cập nhật chi tiết đơn hàng";
        TempData["caphatdh"] = chiTietDonHang.MaDH;
    }
}

```

#### Action Method:

- \_ ChinhSuaCTDH: Hiển thị trang ChinhSuaCTDH
- \_ Post: ChinhSuaCTDH: Thực hiện quá trình chỉnh sửa trong chi tiết đơn hàng và trả về DonHang khi đã hoàn tất

```

    [HttpPost]
    [ValidateAntiForgeryToken]
    public ActionResult ChinhSuaCTDH([Bind(Include = "MaSP,SoLuong,DonGia,DiemGiao,DonViTinh")] ChiTietDonHang chiTietDonHang)
    {
        if (ModelState.IsValid)
        {
            var dongia = database.SanPhams.Where(s => s.MaSP == chiTietDonHang.MaSP).FirstOrDefault();
            chiTietDonHang.MaCTDH = (int)Session["MaCTDH"];
            if (chiTietDonHang.ChietKhai == null)
            {
                chiTietDonHang.ThanhTien = (chiTietDonHang.SoLuong) * (dongia.Gia);
            }
            else
            {
                chiTietDonHang.ThanhTien = (chiTietDonHang.SoLuong) * (dongia.Gia) * (chiTietDonHang.ChietKhai);
            }
            chiTietDonHang.MaDH = (string)Session["madh"];
            TempData["messageAlert"] = "Đã cập nhật chi tiết đơn hàng";
            TempData["capnhatdh"] = chiTietDonHang.MaDH;
            database.Entry(chiTietDonHang).State = System.Data.Entity.EntityState.Modified;
            database.SaveChanges();
            return RedirectToAction("DanhSachCTDH", "KinhDoanh", new { id = chiTietDonHang.MaDH });
        }
        ViewBag.MaSP = new SelectList(database.SanPhams, "MaSP", "TenSP", chiTietDonHang.MaSP);
        return View(chiTietDonHang);
    }

    //Hiển thị danh sách CTDH
    public ActionResult DanhSachCTDH(string id)
    {
        TempData["madh"] = id;
        return View(database.ChiTietDonHangs.ToList().Where(s => s.MaDH == id));
    }
    public ActionResult ThemCTHD1()
    {
        ViewBag.MaDH = new SelectList(database.DonHangs, "MaDH", "MaDL");
        ViewBag.MaSP = new SelectList(database.SanPhams, "MaSP", "TenSP");
        return View();
    }

    public ActionResult ThemCTHD1()
    {
        ViewBag.MaDH = new SelectList(database.DonHangs, "MaDH", "MaDL");
        ViewBag.MaSP = new SelectList(database.SanPhams, "MaSP", "TenSP");
        return View();
    }

    [HttpPost]
    [ValidateAntiForgeryToken]
    public ActionResult ThemCTHD2([Bind(Include = "MaDH,MaSP,SoLuong")] ChiTietDonHang chiTietDonHang)
    {
        if (ModelState.IsValid)
        {
            int sl = int.Parse(Request["s1"]);
            double total = 0;
            var dh = database.DonHangs.Find(chiTietDonHang.MaDH);
            var ldl = database.DaiLies.Where(s => s.MaDL == dh.MaDL).FirstOrDefault();
            var ck = database.LoaiDLs.Where(s => s.MaLoaiDL == ldl.MaLoaiDL).FirstOrDefault();
            Random rd = new Random();
            for (int i = 1; i <= sl; i++)
            {
                chiTietDonHang.MaSP = Request["MaSP" + i];
                var soluong = Convert.ToInt32(Request["SoLuong" + i]);
                var dg = database.SanPhams.Where(s => s.MaSP == chiTietDonHang.MaSP).FirstOrDefault();
                //Kiểm tra số lượng sản phẩm tồn trong kho
                var checkslp = database.SanPhams.Where(s => s.MaSP == chiTietDonHang.MaSP).FirstOrDefault();
                if (soluong > checkslp.TongTon)
                {
                    TempData["messageAlert"] = "Không đủ số lượng để đặt";
                    TempData["s1masp"] = chiTietDonHang.MaSP;
                    return RedirectToAction("DanhSachCTDH", "KinhDoanh", new { id = chiTietDonHang.MaDH });
                }
                var checkdh = database.ChiTietDonHangs.Where(s => s.MaDH == chiTietDonHang.MaDH && s.MaSP == chiTietDonHang.MaSP).FirstOrDefault();
                if (checkdh != null)
                {
                    chiTietDonHang = checkdh;
                    //Nếu trùng sản phẩm
                    chiTietDonHang.MaCTDH = checkdh.MaCTDH;
                }
            }
        }
    }
}

```

```

[HttpPost]
[ValidateAntiForgeryToken]
public ActionResult ThemCTHD2([Bind(Include = "MaDH,MaSP,Soluong")] ChiTietDonHang chiTietDonHang)
{
    if (ModelState.IsValid)
    {
        int sl = int.Parse(Request["sl"]);
        double total = 0;
        var dh = database.DonHangs.Find(chiTietDonHang.MaDH);
        var ldl = database.Dailies.Where(s => s.MaDL == dh.MaDL).FirstOrDefault();
        var ck = database.LoaiLs.Where(s => s.MaLoaiDL == ldl.MaLoaiDL).FirstOrDefault();
        Random rd = new Random();
        for (int i = 1; i <= sl; i++)
        {
            chiTietDonHang.MaSP = Request["MaSP" + i];
            var soluong = Convert.ToInt32(Request["Soluong" + i]);
            var dg = database.SanPhams.Where(s => s.MaSP == chiTietDonHang.MaSP).FirstOrDefault();
            //Kiểm tra số lượng sản phẩm tồn trong kho
            var checkslp = database.SanPhams.Where(s => s.MaSP == chiTietDonHang.MaSP).FirstOrDefault();
            if (soluong > checkslp.TongTon)
            {
                TempData["messageAlert"] = "Không đủ số lượng để đặt";
                TempData["slmasp"] = chiTietDonHang.MaSP;
                return RedirectToAction("DanhSachCTDH", "KinhDoanh", new { id = chiTietDonHang.MaDH });
            }
            var checkdh = database.ChiTietDonHangs.Where(s => s.MaDH == chiTietDonHang.MaDH && s.MaSP == chiTietDonHang.MaSP).FirstOrDefault();
            if (checkdh != null)
            {
                chiTietDonHang = checkdh;
                //Nếu trùng sản phẩm
                chiTietDonHang.MaCTDH = checkdh.MaCTDH;
                chiTietDonHang.Soluong = checkdh.Soluong + soluong;
                chiTietDonHang.ThanhTien = (dg.Gia) * (chiTietDonHang.Soluong);
                database.Entry(chiTietDonHang).State = System.Data.Entity.EntityState.Modified;
                database.SaveChanges();
            }
            else
            {
                chiTietDonHang.MaCTDH = rd.Next(0, 100000);
            }
            var checkdh = database.ChiTietDonHangs.Where(s => s.MaDH == chiTietDonHang.MaDH && s.MaSP == chiTietDonHang.MaSP).FirstOrDefault();
            if (checkdh != null)
            {
                chiTietDonHang = checkdh;
                //Nếu trùng sản phẩm
                chiTietDonHang.MaCTDH = checkdh.MaCTDH;
                chiTietDonHang.Soluong = checkdh.Soluong + soluong;
                chiTietDonHang.ThanhTien = (dg.Gia) * (chiTietDonHang.Soluong);
                database.Entry(chiTietDonHang).State = System.Data.Entity.EntityState.Modified;
                database.SaveChanges();
            }
            else
            {
                chiTietDonHang.MaCTDH = rd.Next(0, 100000);
                chiTietDonHang.Soluong = soluong;
                chiTietDonHang.ThanhTien = (dg.Gia) * (soluong);
                database.ChiTietDonHangs.Add(chiTietDonHang);
                database.SaveChanges();
            }
        }
        //Kiểm tra tổng tiền hiện tại đơn hàng
        var checktongtien = database.ChiTietDonHangs.Where(s => s.MaDH == chiTietDonHang.MaDH).ToList();
        foreach(var item01 in checktongtien)
        {
            total += (double) item01.ThanhTien;
        }
        dh.TongTien = total;
        if (ldl.MaLoaiDL == "LDL01")
        {
            if (dh.TongTien > 50000000)
            {
                TempData["messageAlert"] = "error01";
                TempData["themmadh"] = chiTietDonHang.MaDH;
                TempData["madl"] = dh.MaDL;
                return RedirectToAction("QuanLyDH");
            }
        }
        else if (ldl.MaLoaiDL == "LDL02")
    }
}

```

```

        TempData["messageAlert"] = "error01";
        TempData["themmadh"] = chiTietDonHang.MaDH;
        TempData["madl"] = dh.MaDL;
        return RedirectToAction("QuanLyDH");
    }
}
else if (ldl.MaLoaiDL == "LDL02")
{
    if (dh.TongTien > 250000000)
    {
        TempData["messageAlert"] = "error01";
        TempData["themmadh"] = dh.MaDH;
        TempData["madl"] = dh.MaDL;
        return RedirectToAction("QuanLyDH");
    }
}
else if (ldl.MaLoaiDL == "LDL03")
{
    if (dh.TongTien > 100000000)
    {
        TempData["messageAlert"] = "error01";
        TempData["themmadh"] = dh.MaDH;
        TempData["madl"] = dh.MaDL;
        return RedirectToAction("QuanLyDH");
    }
}
dh.TongTien = dh.TongTien - (dh.TongTien * ck.ChietKhau);
//Cập nhật lại tổng tiền của đơn hàng
//dh.TongTien += chitietDonHang.ThanhTien;
database.Entry(dh).State = System.Data.Entity.EntityState.Modified;
database.SaveChanges();
TempData["messageAlert"] = "Đã cập nhật chi tiết đơn hàng";
TempData["capnhatdh"] = chiTietDonHang.MaDH;
TempData["messageAlert"] = "Đã cập nhật chi tiết đơn hàng";
return RedirectToAction("DanhSachCTDH", "KinhDoanh", new { id= chiTietDonHang.MaDH });
}

ViewBag.MaSP = new SelectList(database.SanPhams, "MaSP", "TenSP", chiTietDonHang.MaSP);
return View(chiTietDonHang);
}

//CHECK TỔNG TIỀN TRƯỚC
[HttpPost]
public ActionResult CheckTongTien(string MaSP, int SoLuong)
{
    var total = 0;
    var checksp = database.SanPhams.Where(s => s.MaSP == MaSP).FirstOrDefault();
    var tongton = checksp.TongTon;
    total = (int)(checksp.Gia * SoLuong);
    return Json(new { data=total,tongton}, JsonRequestBehavior.AllowGet);
}

[HttpPost]
public ActionResult CheckCongNo(string MaDH, string ThanhTien)
{
    var tongtiendh = 0;
    var checkdh = database.DonHangs.Where(s => s.MaDH == MaDH).FirstOrDefault();
    var daily = database.Dailies.Where(s => s.MaDL == checkdh.MaDL).FirstOrDefault();
    var ctdh = database.ChiTietDonHangs.Where(s => s.MaDH == MaDH);
    foreach(var item in ctdh)
    {
        var sp = database.SanPhams.Where(s => s.MaSP == item.MaSP).FirstOrDefault();
        tongtiendh += Convert.ToInt32(sp.Gia * item.SoLuong);
    }
    tongtiendh += Convert.ToInt32(ThanhTien);
    if (daily.MaLoaiDL == "LDL01" && tongtiendh <=500000000)
    {
        return Json(new { success=true }, JsonRequestBehavior.AllowGet);
    }
    else if (daily.MaLoaiDL == "LDL02" && tongtiendh <= 250000000)
    {
        return Json(new { success = true }, JsonRequestBehavior.AllowGet);
    }
    else if (daily.MaLoaiDL == "LDL03" && tongtiendh <= 100000000)
    {
        return Json(new { success = true }, JsonRequestBehavior.AllowGet);
    }
    else
    {
        return Json(new { success = false }, JsonRequestBehavior.AllowGet);
    }
}
//Chỉnh sửa thông tin đơn hàng

```

**Action Method:**

\_ CheckTongTien : Thực hiện kiểm tra tổng tiền và trả kết quả theo dạng Json

\_ Post: CheckCongNo: Thực hiện kiểm tra Công nợ và kết quả trả về theo dạng Json

```
//Chỉnh sửa thông tin đơn hàng
public ActionResult ChinhSuaDH(string id)
{
    if (id == null)
    {
        return new HttpStatusCodeResult(System.Net.HttpStatusCode.BadRequest);
    }
    DonHang donHang = database.DonHangs.Find(id);
    if (donHang == null)
    {
        return HttpNotFound();
    }
    return View(donHang);
}
[HttpPost, ActionName("ChinhSuaDH")]
[ValidateAntiForgeryToken]
public ActionResult ChinhSuaDH(DonHang donHang)
{
    ChiTietDonHang ct = new ChiTietDonHang();
    if (ModelState.IsValid)
    {
        database.Entry(donHang).State = System.Data.Entity.EntityState.Modified;
        database.SaveChanges();
        return RedirectToAction("QuanLyDH");
    }
    return View(donHang);
}
```

Action Method:

\_ ChinhSuaDH: Mở trang chỉnh sửa đơn hàng (View ChinhSuaDH)

\_ Post: ChinhSuaDH: Thực hiện chỉnh sửa đơn hàng và trả về màn hình QuanLyDH khi hoàn tất

```
//Xóa chi tiết đơn hàng
// GET: ChiTietDonHangs/Delete/5
public ActionResult XoaCTDH(string id)
{
    var madh = database.ChiTietDonHangs.Where(s => s.MaDH == id).FirstOrDefault();
    if (madh == null)
    {
        return new HttpStatusCodeResult(HttpStatusCode.BadRequest);
    }
    return View(madh);
}
// POST: ChiTietDonHangs/Delete/5
[HttpPost, ActionName("XoaCTDH")]
[ValidateAntiForgeryToken]
public ActionResult XoaCTDH1(string id)
{
    var madh = database.ChiTietDonHangs.Where(s => s.MaDH == id).FirstOrDefault();
    database.ChiTietDonHangs.Remove(madh);
```

Action Method:

\_ XoaCTDH: Mở trang XoaCTDH xác nhận lại trước khi xóa

\_ Post: XoaCTDH1 : thực hiện xóa một sản phẩm trong chi tiết đơn hàng và trả về trang đơn hàng DonHang khi hoàn tất

```

[HttpPost, ActionName("XoaCTDH")]
[ValidateAntiForgeryToken]
public ActionResult XoaCTDH(string id)
{
    var madh = database.ChiTietDonHangs.Where(s => s.MaDH == id).FirstOrDefault();

    database.ChiTietDonHangs.Remove(madh);
    database.SaveChanges();

    var dh = database.DonHangs.Find(id);
    var ldl = database.DaiLies.Where(s => s.MaDL == dh.MaDL).FirstOrDefault();
    var ck = database.LoaiDLs.Where(s => s.MaLoaiDL == ldl.MaLoaiDL).FirstOrDefault();
    double total = 0;
    var checktongtien = database.ChiTietDonHangs.Where(s => s.MaDH == id).ToList();
    foreach (var item01 in checktongtien)
    {
        total += (double)item01.ThanhTien;
    }

    dh.TongTien = total - (total*ck.ChietKhau);

    database.Entry(dh).State = System.Data.Entity.EntityState.Modified;
    database.SaveChanges();
    return RedirectToAction("QuanLyDH");
}
//Xóa đơn hàng
public ActionResult XoaDH(string id)
{
    if (id == null)
    {
        return new HttpStatusCodeResult(System.Net.HttpStatusCode.BadRequest);
    }
    DonHang donHang = database.DonHangs.Find(id);
    if (donHang == null)
    {
        return HttpNotFound();
    }
    return View(donHang);
}

```

#### Action Method:

- \_ XoaDH : Thực hiện mở trang XoaDH

```

        [HttpPost, ActionName("XoaDH")]
        [ValidateAntiForgeryToken]
        public ActionResult XoaDHConfirm(string id)
        {
            DonHang donHang = database.DonHangs.Find(id);
            var daily = database.Dailies.Find(donHang.MaDL);
            TempData["capnhattrang"] = donHang.MaDH;
            database.DonHangs.Remove(donHang);
            database.SaveChanges();
            TempData["messageAlert"] = "Đã xóa đơn hàng";

            //Gửi email về tk
            string content = System.IO.File.ReadAllText(Server.MapPath("~/TemplateMail/CancleOrder.html"));
            content = content.Replace("{Username}", daily.UserName);
            content = content.Replace("{MaDH}", donHang.MaDH);
            content = content.Replace("{datetimetime}", DateTime.Now.ToString());
            content = content.Replace("{tinhtrangTT}", donHang.TrangThai);
            content = content.Replace("{TrangThaiGH}", donHang.TinhTrangGH);
            //content = content.Replace("{Total}", tongtien);
            //content = content.Replace("{Thoigian}", Convert.ToString(ttkh.CreateDate));
            //content = content.Replace("{Invoice}", themhd.MaHD);
            string subject = "ĐƠN HÀNG "+donHang.MaDH+" ĐÃ ĐƯỢC HỦY";
            WebMail.Send(daily.Email, subject, content, null, null, null, true, null, null, null, null, null, null);
            return RedirectToAction("QuanLyDH");
        }

        public ActionResult QuanLySP()
        {
            return View(database.SanPhams.ToList().OrderByDescending(s=> s.NgaySX));
        }

        public ActionResult ThongTinSP(string id)
        {
            if (id == null)

```

### Action Method:

\_QuanLySP: mở trang QuanLySP

\_ThongTinSP: Mở trang thông tin sản phẩm (View ThongTinSP)

```

public ActionResult ThemSP()
{
    return View();
}

// POST: SanPhams/Create
// To protect from overposting attacks, enable the specific properties you want to bind to, for
// more details see https://go.microsoft.com/fwlink/?LinkId=317598.

//Hàm lưu ảnh (Khi chuyển sang phòng khác nhớ đổi SanPham thành tên đối tượng cần được lưu ảnh )
public void LuuAnh(SanPham sp, HttpPostedFileBase Hinhanh)
{
    Hinhanh
    //Lưu file (Kiểm tra trùng file)
    Hinhanh.SaveAs(fullDuongDan);
    sp.Hinhanh = urlTuongdoi + Path.GetFileName(fullDuongDan);
}

[HttpPost]
[ValidateAntiForgeryToken]
public ActionResult ThemSP([Bind(Include = "TenSP,DonViTinh,Gia,HanSD,NgaySX,Hinhanh")] SanPham sanPham, HttpPostedFileBase Hinhanh)
{
    if (ModelState.IsValid)
    {
        LuuAnh(sanPham, Hinhanh);
        Random rd = new Random();
        var themSP = "SP" + rd.Next(0,9)+rd.Next(0,9)+rd.Next(0,9);

        sanPham.MaSP = themSP;
        database.SanPhams.Add(sanPham);
        database.SaveChanges();
        TempData["messageAlert"] = "Đã thêm mới sản phẩm";
        TempData["maspTT"] = sanPham.MaSP;
        return RedirectToAction("QuanLySP");
    }

    return View(sanPham);
}

// GET: SanPhams/Edit/5
public ActionResult ChinhSuaSP(string id)

```

### Action Method:

\_ThemSP: mở trang ThemSP

\_Post: ThemSP: Thực hiện thêm sản phẩm và trả về trang QuanLySP

```
// GET: SanPhams/Edit/5
public ActionResult ChinhSuaSP(string id)
{
    if (id == null)
    {
        return new HttpStatusCodeResult(System.Net.HttpStatusCode.BadRequest);
    }
    SanPham sanPham = database.SanPhams.Find(id);

    if (sanPham == null)
    {
        return HttpNotFound();
    }
    return View(sanPham);
}

// POST: SanPhams/Edit/5
// To protect from overposting attacks, enable the specific properties you want to bind to, for
// more details see https://go.microsoft.com/fwlink/?LinkId=317598.
[HttpPost]
[ValidateAntiForgeryToken]
public ActionResult ChinhSuaSP(SanPham sanPham, HttpPostedFileBase Hinhanh)
{
    if (ModelState.IsValid)
    {
        if(Hinhanh == null)
        {
            var hac = database.SanPhams.AsNoTracking().Where(s=>s.MaSP==sanPham.MaSP).FirstOrDefault();
            sanPham.Hinhanh = hac.Hinhanh;
            database.Entry(sanPham).State = System.Data.Entity.EntityState.Modified;
            database.SaveChanges();
            TempData["messageAlert"] = "Đã cập nhật sản phẩm";
            TempData["maspTT"] = sanPham.MaSP;
        }
        else
        {
            LuuAnh(sanPham, Hinhanh);
            database.Entry(sanPham).State = System.Data.Entity.EntityState.Modified;
        }
    }
    return RedirectToAction("QuanLySP");
}

```

### Action Method:

\_ChinhSuaSP: thực hiện mở trang chỉnh sửa ChinhSuaSP

\_ Post: ChinhSuaSP: Thực hiện chỉnh sửa thông tin sản phẩm và trả lại view

QuanLySP

```
[HttpPost]
[ValidateAntiForgeryToken]
public ActionResult ChinhSuaSP(SanPham sanPham, HttpPostedFileBase Hinhanh)
{
    if (ModelState.IsValid)
    {
        if(Hinhanh == null)
        {
            var hac = database.SanPhams.AsNoTracking().Where(s=>s.MaSP==sanPham.MaSP).FirstOrDefault();
            sanPham.Hinhanh = hac.Hinhanh;
            database.Entry(sanPham).State = System.Data.Entity.EntityState.Modified;
            database.SaveChanges();
            TempData["messageAlert"] = "Đã cập nhật sản phẩm";
            TempData["maspTT"] = sanPham.MaSP;
        }
        else
        {
            LuuAnh(sanPham, Hinhanh);
            database.Entry(sanPham).State = System.Data.Entity.EntityState.Modified;
            database.SaveChanges();
            TempData["messageAlert"] = "Đã cập nhật sản phẩm";
            TempData["maspTT"] = sanPham.MaSP;
        }
    }
    return RedirectToAction("QuanLySP");
}
return View(sanPham);
}

// GET: SanPhams/Delete/5
public ActionResult XoaSP(string id)
{
    if (id == null)
    {
        return new HttpStatusCodeResult(System.Net.HttpStatusCode.BadRequest);
    }
    SanPham sanPham = database.SanPhams.Find(id);
    if (sanPham == null)
    {
        return HttpNotFound();
    }
}
```

### Action Method:

### \_ XoaSP: Thực mở trang xác nhận XoaSP

```

3     public ActionResult XoaSP(string id)
4     {
5         if (id == null)
6         {
7             return new HttpStatusCodeResult(System.Net.HttpStatusCode.BadRequest);
8         }
9         SanPham sanPham = database.SanPhams.Find(id);
10        if (sanPham == null)
11        {
12            return HttpNotFound();
13        }
14        return View(sanPham);
15    }
16    [HttpPost]
17    [ValidateAntiForgeryToken]
18    public ActionResult xoasp(string id)
19    {
20        try
21        {
22            SanPham sanPham = database.SanPhams.Find(id);
23            database.SanPhams.Remove(sanPham);
24            database.SaveChanges();
25
26            TempData["messageAlert"] = "Đã xóa sản phẩm";
27            TempData["maspTT"] = sanPham.MaSP;
28            return RedirectToAction("QuanLySP");
29        }
30        catch
31        {
32            return Content("Không xóa được dữ liệu");
33        }
34    }
35}

```

### Action Method:

\_ Post: XoaSP: Thực hiện xóa sản phẩm và trả lại thông báo view QuanLySP

```

public ActionResult QuanLyHD()
{
    return View(database.DonHangs.ToList().OrderByDescending(s=>s.NgayLap));
}
public ActionResult QuanLyKho()
{
    return View();
}
//Tiếp nhận phiếu đề nghị nhập/ xuất hàng hóa
public ActionResult TiepNhanHH()
{
    var dstiephan = database.PhiếuNhậpXuats.ToList();
    return View(dstiephan);
}
//Phiếu nhập kho
public ActionResult ChiTietPhieuNhapKho(string id)
{
    if (id == null)
    {
        return new HttpStatusCodeResult(HttpStatusCode.BadRequest);
    }
    TempData["maPhieuNhap"] = id;
    Session["KinhGuu"] = "Trường phòng kinh doanh";
    Session["HoTen"] = "Nguyễn Thị Diễm Khang";
    PhieuNhapXuat phieuNhapXuat = database.PhiếuNhậpXuats.Find(id);
    if (phieuNhapXuat == null)
    {
        return HttpNotFound();
    }
    return View(phieuNhapXuat);
}

```

### Action Method:

\_ QuanLyDH: Thực hiện mở trang QuanLyDH

\_ TiepNhanHH: Thực hiện mở trang TiepNhanHH

\_ ChiTietPhieuNhapKho: Thực hiện mở trang ChiTietPhieuNhapKho

\_ ChiTietPhieuXuatKho: Thực hiện mở trang ChiTietPhieuXuatKho

```

//Phiếu xuất kho
public ActionResult ChiTietPhieuXuatKho(string id)
{
    if (id == null)
    {
        return new HttpStatusCodeResult(HttpStatusCode.BadRequest);
    }
    TempData["maPhieuXuat"] = id;
    Session["KinhGu"] = "Trường phòng kinh doanh";
    Session["HoTen"] = "Nguyễn Thị Diễm Khang";
    PhieuNhapXuat phieuNhapXuat = database.PhieuNhapXuats.Find(id);
    if (phieuNhapXuat == null)
    {
        return HttpNotFound();
    }
    return View(phieuNhapXuat);
}
//Xét duyệt phiếu
public ActionResult XetDuyetPhieu(string MaPhieu)
{
    var check = database.PhieuNhapXuats.Where(s => s.MaPhieu == MaPhieu).FirstOrDefault();
    if(check != null)
    {
        check.TinhTrang = "Đã xét duyệt";
        database.Entry(check).State = System.Data.Entity.EntityState.Modified;
        database.SaveChanges();
        return Json(new { success = true, MaPhieu=check.MaPhieu }, JsonRequestBehavior.AllowGet);
    }
    return Json(new { success = false }, JsonRequestBehavior.AllowGet);
}

// Báo cáo
public ActionResult BaoCao()
{
    return View();
}
[HttpPost]
[ValidateInput(false)]
public ActionResult BaoCao(BaoCao baocao)
{
    if (ModelState.IsValid)
    {

        Random rd = new Random();
        var mabc = "BC" + rd.Next(1, 1000);
        baocao.MaBC = mabc;
        var manv = (Test02.Models.NhanVien)Session["user"];
        baocao.MaNV = manv.MaNV;
        database.BaoCaos.Add(baocao);
        database.SaveChanges();
        return RedirectToAction("TrangChu");
    }

    return View(baocao);
}
public void Hamrandom( )
{
}
}
public ActionResult TuChoiTruyCap()
{
    return View();
}

```

Action Method:

\_ BaoCao: Thực hiện mở trang BaoCao

- \_ Post: BaoCao: Thực hiện cập nhật thêm mới báo cáo và trả lại TrangChu khi hoàn tất
- \_ TuChoiTruyCap : Thực hiện mở trang TuChoiTruyCap

#### **4.2.2.2. Phòng Kế Toán:**

```

1 reference
public double maxno(string maloai)
{
    double maxcn = 0;
    if (maloai == "LDL01")
    {
        maxcn = 500000000;
    }
    else if (maloai == "LDL02")
    {
        maxcn = 250000000;
    }
    else
    {
        maxcn = 100000000;
    }
    return maxcn;
}

1 reference
public int SoTinhTrangDaiLy()
{
    int countdl = 0;
    List<Daily> dly = database.Dailies.ToList();
    foreach(var item in dly)
    {
        double? a = database.sp_SumCN(item.MaDL).FirstOrDefault();
        double maxcndl = maxno(item.MaLoaiDL);
        double dangercn = maxcndl - (maxcndl * 0.2);
        if (a >= dangercn)
        {
            countdl++;
        }
    }
    return countdl;
}

```

Action method:

maxno : trả về số nợ tối đa cho từng loại đại lý

SoTinhTrangDaiLy: trả về số lượng đại lý có mức tổng công nợ đạt gần tới mức công nợ tối đa cho phép

```
0 references
public ActionResult TrangChuKeToan()
{
    ViewBag.tongdathanhtoan = database.sp_DHtheotinhtrang("Đã thanh toán").FirstOrDefault();
    ViewBag.tongchuathanhtoan = database.sp_DHtheotinhtrang("Chưa thanh toán").FirstOrDefault();
    ViewBag.tongdonno = database.sp_DHtheotinhtrang("Đang nợ").FirstOrDefault();
    ViewBag.donthanhtoan = database.sp_DonHangtrongngay("Đã thanh toán").FirstOrDefault();
    ViewBag.donchuathanhtoan = database.sp_DonHangtrongngay("Chưa thanh toán").FirstOrDefault();
    ViewBag.donno = database.sp_DonHangtrongngay("Đang nợ").FirstOrDefault();

    int dem = SoTinhTrangDaiLy();
    Session["demdl"] = dem;
    Session["doanhthu"] = database.sp_DoanhThutrongthang().FirstOrDefault();
    List<sp_CongNoDenHan_Result> cnhan = database.sp_CongNoDenHan().ToList();
    Session["cndenhan"] = cnhan;
    return View();
}
```

**Action Method**

- TrangChuKeToan: trả về view trang chủ cho role kế toán

```
0 references
public ActionResult QLDonHang()
{
    BindDropDownList();
    Tinhhangdonhang();
    List<sp_LocDonHang_Result> dh = database.sp_LocDonHang("All", "All", "Đã xét duyệt").ToList();
    return View(dh);
}
[HttpPost]
0 references
public ActionResult QLDonHang(string thangs, string tinhtrangdh)
{
    BindDropDownList();
    Tinhhangdonhang();
    List<sp_LocDonHang_Result> dh = database.sp_LocDonHang(thangs, tinhtrangdh, "Đã xét duyệt").ToList();
    return View(dh);
}
```

**Action Method:**

- QLDonHang: hiển thị danh sách tất cả đơn hàng
- QLDonHang (HttpPost) : hiển thị danh sách đơn hàng được lọc theo tháng và tình trạng thanh toán

```

0 references
public ActionResult XuatHDBH(String id)
{
    if (id == null)
    {
        return new HttpStatusCodeResult(System.Net.HttpStatusCode.BadRequest);
    }
    DonHang donHang = database.DonHangs.Find(id);
    Session["madaily"] = id;
    if (donHang == null)
    {
        return HttpNotFound();
    }
    return View(donHang);
}

0 references
public ActionResult ChinhhsuaHD(DonHang donHang)
{
    if (ModelState.IsValid)
    {
        database.Entry(donHang).State = System.Data.Entity.EntityState.Modified;
        database.SaveChanges();
        TempData["edithd"] = "oke";
        return RedirectToAction("XuatHDBH/" + donHang.MaDH);
    }
    return View(donHang);
}

```

### Action Method

- XuatHDBH: hiển thị thông tin chi tiết của một đơn hàng
- ChinhhsuaHD: thực hiện cập nhật thông tin đơn hàng

```

0 references
public ActionResult DSDaiLy()
{
    database.sp_capnhat();
    return View(database.Dailies.ToList().OrderBy(s => s.MaDL));
}

```

### Action Method:

- DSDaiLy: hiển thị danh sách đại lý

```

0 references
public ActionResult CongnoDL(string id)
{
    BindDropDownList();
    List<DonHang> dh = Get_DonHang("All", "Chưa thanh toán", "Đã xét duyệt").Where(s=>s.MaDL==id && s.TinhTrangGH=="Đã giao").ToList();
    ViewBag.TongCN = database.sp_TienDHNODaily(id, "All").FirstOrDefault();
    Session["Madly"] = id;
    Session["thangtrongdonhang"] = KtraThang(dh);
    Session["KtraDH"] = KtraDH(dh);
    return View(dh);
}
[HttpPost]
0 references
public ActionResult CongnoDL(string id, string thangs)
{
    BindDropDownList();
    List<DonHang> dh = Get_DonHang(thangs, "Chưa thanh toán", "Đã xét duyệt").Where(s => s.MaDL == id && s.TinhTrangGH == "Đã giao").ToList();
    ViewBag.TongCN = database.sp_TienDHNODaily(id, thangs).FirstOrDefault();
    Session["KtraDH"] = KtraDH(dh);
    Session["Madly"] = id;
    Session["thangtrongdonhang"] = KtraThang(dh);
    return View(dh);
}

```

**Action Method:**

- CongnoDL: hiển thị danh sách tất cả đơn hàng tính công nợ cho một đại lý cụ thể
- CongnoDL(HttpPost) : hiển thị danh sách các đơn hàng tính công nợ của một đại lý được lọc theo tháng

```

0 references
public ActionResult TaoCongno( string MaDL, PhieuCongNo pphieuCongNo, DateTime Hantra, DateTime NgayLapCN )
{
    System.Threading.Thread.Sleep(5000);
    try
    {
        var c = database.Dailies.Where(s => s.MaDL == MaDL).ToList().FirstOrDefault();
        string thanglapcn = NgayLapCN.Month.ToString();
        var tienno = database.sp_TienDHNODaiLy(MaDL, thanglapcn).FirstOrDefault();
        database.sp_CongnoDLY(MaDL, tienno, Hantra, NgayLapCN, thanglapcn);
    }
    catch
    {
        TempData["taocn"] = "loicn";
        return RedirectToAction("DSDaily");
    }
    TempData["taocn"] = "success";
    return RedirectToAction("DSDaily");
}

```

**Action Method:**

- TaoCongno: thực hiện lập công nợ cho đại lý

```

public ActionResult DSCongnoNo(String id)
{
    Session["madailycn"] = id;
    BindDropDownList();
    TinhTangCongno();
    List<sp_LocPhieuCongNo_Result> dscongno = database.sp_LocPhieuCongNo("All", "All", id).ToList();
    return View(dscongno);
}

[HttpPost]
0 references
public ActionResult DSCongnoNo(String id, string thangs, string tinhtrangcn)
{
    BindDropDownList();
    TinhTangCongno();
    var dscongno = database.PhieuCongNoes.Where(s => s.MaDL == id).ToList();
    List<sp_LocPhieuCongNo_Result> cn = database.sp_LocPhieuCongNo(thangs, tinhtrangcn, id).ToList();
    Session["madailycn"] = id;
    return View(cn);
}

```

#### Action Method:

- DSCongnoNo: hiển thị danh sách tất cả phiếu công nợ của một đại lý cụ thể
- DSCongnoNo(HttpPost) : hiển thị danh sách phiếu công nợ của đại lý được lọc theo tháng và tình trạng thanh toán

```

0 references
public ActionResult SendMailCN(string id)
{
    var cn = database.PhieuCongNoes.Where(t => t.MaCongNo == id).FirstOrDefault();
    var daily = database.DaiLies.Where(t => t.MaDL == cn.MaDL).FirstOrDefault();
    string content = System.IO.File.ReadAllText(Server.MapPath("~/TemplateMail/Doino.html"));

    content = content.Replace("{{Username}}", daily.TenDL);
    content = content.Replace("{{MaCN}}", cn.MaCongNo);
    content = content.Replace("{{Tienno}}", cn.TienNo.ToString());
    content = content.Replace("{{NgayLap}}", cn.NgayLapCN.ToString());
    content = content.Replace("{{Han}}", cn.HanTra.ToString());
    string subject = "Thư báo công nợ chưa thanh toán";

    WebMail.Send(daily.Email, subject, content, null, null, null, true, null, null, null, null, null, null);
    TempData["baono"] = "thanhcong";
    return RedirectToAction("ChitietCN", "KeToan", new { id = id });
}

```

#### Action Method:

- SenMailCN: thực hiện gửi mail nhắc tới hạn trả tiền công nợ cho đại lý

```

0 references
public ActionResult ChinhSuCN(string id)
{
    return View(database.PhieuCongNoes.Where(s => s.MaCongNo == id).FirstOrDefault());
}

[HttpPost]
0 references
public ActionResult ChinhSuCN(string id, PhieuCongNo phieuCongNo)
{
    if (ModelState.IsValid)
    {
        database.Entry(phieuCongNo).State = System.Data.Entity.EntityState.Modified;
        List<DonHang> donhang = database.DonHangs.Where(s => s.MaDL == phieuCongNo.MaDL
            && s.TinhTrangThanhToan == "Đang nợ"
            && s.TrangThai == "Đã xét duyệt" && s.TinhTrangGH == "Đã giao").ToList();
        if (phieuCongNo.TrangThai == "Đã thanh toán")
        {
            foreach(var item in donhang)
            {
                item.TinhTrangThanhToan = "Đã thanh toán";
            }
        }
        database.SaveChanges();
        TempData["chinhSuacn"] = "chinhthanhtcong";
        return RedirectToAction("DSDaily");
    }
    return View(phieuCongNo);
}

```

#### Action Method:

- ChinhSuCN: hiển thị view cập nhật thông tin phiếu công nợ
- ChinhSuCN(HttpPost): thực hiện cập nhật thông tin phiếu công nợ

```

0 references
public ActionResult XoaCN(string id, PhieuCongNo phieuCongNo)
{
    System.Threading.Thread.Sleep(5000);

    try
    {
        database.sp_XoaCN(id);
        TempData["huyXoaCN"] = "daxoa";
        return RedirectToAction("DSCongNo/" + id);
    }
    catch
    {
        TempData["huyXoaCN"] = "khongxoa";
        return RedirectToAction("DSCongNo/" + id);
    }
}

0 references
public ActionResult ChitietCN(string id)
{
    var phieucn = database.PhieuCongNoes.Where(s => s.MaCongNo == id).FirstOrDefault();
    List<DonHang> donhang = new List<DonHang>();
    if (phieucn.TrangThai == "Chưa thanh toán")
    {
        donhang = database.DonHangs.Where(s => s.TinhTrangThanhToan == "Đang nợ"
            && s.TrangThai == "Đã xét duyệt" && s.TinhTrangGH == "Đã giao").ToList();
    }
    else
    {
        donhang = database.DonHangs.Where(s => s.TinhTrangThanhToan == "Đã thanh toán" && s.TrangThai == "Đã xét duyệt"
            && s.TinhTrangGH == "Đã giao").ToList();
    }
    Session["donhangcheck"] = donhang;
    return View(phieucn);
}

```

#### Action Method:

- XoaCN: thực hiện xóa phiếu công nợ của một đại lý cụ thể
- ChitietCN: hiển thị thông tin chi tiết phiếu công nợ của đại lý

```

0 references
public ActionResult Doanhthu()
{
    BindDropDownList();
    List<DonHang> dhthanhtoan = database.DonHangs.Where(s => s.TrangThai == "Đã xét duyệt"
    && s.TinhTrangThanhToan == "Đã thanh toán" && s.TinhTrangGH == "Đã giao").ToList();
    return View(dhthanhtoan);
}
[HttpPost]
0 references
public ActionResult Doanhthu(string thangs)
{
    BindDropDownList();
    List<DonHang> dhthanhtoan = database.DonHangs.Where(s => s.TrangThai == "Đã xét duyệt"
    && s.TinhTrangThanhToan == "Đã thanh toán" && s.TinhTrangGH == "Đã giao").ToList();
    List<DonHang> dhchon = LayDonHang_TheoThang(thangs,dhthanhtoan);

    return View(dhchon);
}

0 references
public ActionResult ThongKe()
{
    List<SanPham> sp = database.SanPhams.OrderBy(s => s.MaSP).ToList();
    return View(sp);
}

```

#### Action Method:

- Doanhthu: hiển thị danh sách tất cả các đơn hàng tính doanh thu
- Doanhthu(HttpPost): hiển thị danh sách đơn hàng tính doanh thu được lọc theo tháng
- ThongKe: hiển thị danh sách tất cả sản phẩm được thống kê số tiền bán được

#### 4.2.2.3. Phòng Kho Hàng:

Thư viện được sử dụng trong controller phòng kho hàng:

```

1  using System;
2  using System.Collections.Generic;
3  using System.Data;
4  using System.Linq;
5  using System.Net;
6  using System.Web;
7  using System.Web.Mvc;
8  using System.Web.Routing;
9  using Test02.App_Start;
10 using Test02.Models;
11

```

Action Method <Test02>: Hiển thị view trang chủ kho hàng:

The screenshot shows the Visual Studio IDE interface. On the left is the code editor with the following C# code for the 'Test02' action method:

```

19 public ActionResult Test02(){
20     //tính tổng số kho
21     var total = database.Khoes.ToList().Count;
22     TempData["tongKho"] = total;
23     //tính tổng sản phẩm sắp hết hàng
24     var dssphh = database.ChiTietKhoes.ToList();
25     var total2 = 0;
26     foreach (var item in dssphh){
27         if (item.TinhTrang == "Sắp hết hàng"){
28             total2++;
29         }
30     }
31     TempData["tongSPHH"] = total2;
32     //tính tổng số sản phẩm
33     var totalsp = database.SanPhams.ToList().Count;
34     TempData["Tongsp1"] = totalsp;
35     //tính tổng số Phiếu nhập xuất hàng
36     var tongpn = 0; var tongpx = 0;
37     var pnx = database.PheuNhapXuats.ToList();
38     foreach (var item in pnx){
39         string str = item.MaPhiieu.Substring(0, 4);
40         if (str == "KDPX"){
41             tongpx++;
42         }
43         else if (str == "KDPN"){
44             tongpn++;
45         }
46         else{
47             tongpx++;
48         }
49     }
50     TempData["Tongpn"] = tongpn;
51     TempData["Tongpx"] = tongpx;
52     //tính tổng sản phẩm tồn kho
53     var total3 = 0;
54     foreach (var item in dssphh){
55         if (item.TinhTrang == "Tồn kho"){
56             total3++;
57         }
58     }
59     TempData["TongSPTK"] = total3;
60     return View(database.Khoes.ToList());
61 }

```

On the right is the Solution Explorer window, which lists the project 'Test02' with its files. The 'Controllers' folder contains several controller files like 'DaiLieController.cs', 'DonHangController.cs', etc. The 'Views' folder contains 'Base', 'ChuyenGiaos', and 'DaiLie' subfolders, with 'Create.cshtml' being the currently selected file.

### Action Method <QuanLyDL>:

- Hiển thị danh sách sản phẩm liên quan đến module Quản lý hàng hóa
- Liên quan đến View danh sách sản phẩm, SP sắp hết hàng / tồn kho

```

76    public ActionResult QuanLyDL()
77    {
78        //tính tổng Sp sắp hết hàng / hết hàng
79        var dssphh = database.ChitietKhoes.ToList();
80        var total2 = 0;
81        foreach (var item in dssphh){
82            if (item.TinhTrang == "Sắp hết hàng" || item.TinhTrang == "Hết hàng"){
83                total2++;
84            }
85            TempData["tongSPHH"] = total2;
86            //tính tổng sản phẩm tồn kho
87            var total3 = 0;
88            foreach (var item in dssphh){
89                if (item.TinhTrang == "Tồn kho") {
90                    total3++;
91                }
92            }
93            TempData["TongSPTK"] = total3;
94            //Kiểm tra đã tạo phiếu chưa
95            var total5 = 0; //đếm trong ct phiếu
96            var total4 = 0; //đếm trong ct kho
97            var total6 = 0; //đếm trong ct phiếu
98            var total7 = 0; //đếm trong ct kho
99            foreach (var ctp in database.ChitietPhiieuNhapXuats){
100                foreach(var item in database.ChitietKhoes){
101                    if (item.TinhTrang == "Sắp hết hàng" || item.TinhTrang == "Hết hàng"){
102                        if (item.MaKho == ctp.MaKho && item.MaSP == ctp.MaSP && item.SoLuong == ctp.SoLuongTrongKho){
103                            total5++;
104                            total4++;
105                            break;
106                        }
107                    }
108                    if (item.TinhTrang == "Tồn kho") {
109                        if (item.MaKho == ctp.MaKho && item.MaSP == ctp.MaSP && item.SoLuong == ctp.SoLuongTrongKho){
110                            total6++;
111                            total7++;
112                            break;
113                        }
114                    }
115                }
116            }
117            //check sắp hết hàng / hết hàng
118            if(total4 == total5 && (total4 != 0 && total5 != 0)){ TempData["checkTrungSPHH"] = true;}else TempData["checkTrungSPHH"] = false;
119            //check tồn kho
120            if (total6 == total7 && (total6 != 0 && total7 != 0)){TempData["checkTrungSPTK"] = true;}
121            else TempData["checkTrungSPTK"] = false;
122        }
123        return View();
124    }

```

### Action Method <QuanLyKho>: Hiển thị danh sách kho liên quan đến module Quản lý kho:

```
146     // Quản lý kho
147     public ActionResult QuanLyKho()
148     {
149         return View(database.Khoes.ToList().OrderByDescending(s => s.ChiTietKhoes.Count()));
150     }
151
152     [HttpPost]
153     public ActionResult QuanLyKho(string Masp)
154     {
155         var masp = Masp.ToUpper();
156         var kholes = new List<Kho>();
157         var chitietkho = database.ChiTietKhoes.Where(s => s.MaSP == masp).ToList();
158         if(chitietkho.Count() == 0)
159         {
160             TempData["AlertMessage"] = "NoFind";
161             return View(kholes);
162         }
163         else
164         {
165             foreach (var item in database.Khoes.ToList())
166             {
167                 foreach(var item2 in chitietkho)
168                 {
169                     if(item.MaKho == item2.MaKho)
170                     {
171                         kholes.Add(item);
172                         break;
173                     }
174                 }
175             }
176         }
177     }
178 }
```

#### Action Method <ThemKho>:

- Thực hiện chức năng thêm kho mới
- Liên quan đến View thêm kho hàng

```

386 [HttpPost]
387 [ValidateAntiForgeryToken]
388 public ActionResult ThemKho([Bind(Include = "MaKho,TenKho,DiaChi")] Kho kho)
389 {
390     //if (ModelState.IsValid)
391     //{
392         // Random rd = new Random();
393         // var makho = "KO" + rd.Next(1, 1000);
394         // kho.MaKho = makho;
395         // database.Khoes.Add(kho);
396         // database.SaveChanges();
397         // TempData["AlertMessage"] = "Đã thêm";
398         // TempData["MaCTKkk"] = makho;
399         // return RedirectToAction("QuanLyKho");
400     //}
401     if (kho.TenKho == null || kho.DiaChi == null)
402     {
403         TempData["AlertMessage"] = "check null";
404         return RedirectToAction("Themkho");
405     }
406     else
407     {
408         Random rd = new Random();
409         var makho = "KO" + rd.Next(1, 1000);
410         kho.MaKho = makho;
411         database.Khoes.Add(kho);
412         database.SaveChanges();
413         TempData["AlertMessage"] = "Đã thêm";
414         TempData["MaCTKkk"] = makho;
415         return RedirectToAction("QuanLyKho");
416     }
417     return View(kho);
418 }

```

- Ngoài ra còn có các action method phục vụ chức năng chỉnh sửa kho, xóa kho khi kho đó không còn hoạt động:

```

445 0 references
446 public ActionResult ChinhSuaKho(string id){...}
447
448 // POST: Khoes/Edit/5
449 // To protect from overposting attacks, enable the specific properties you want to bind to, for
450 // more details see https://go.microsoft.com/fwlink/?LinkId=317598.
451 [HttpPost]
452 [ValidateAntiForgeryToken]
453 0 references
454 public ActionResult ChinhSuaKho([Bind(Include = "MaKho,TenKho,DiaChi")] Kho kho){...}
455
456 // GET: Khoes/Delete/5
457 0 references
458 public ActionResult XoaKho(string id){...}
459
460 // POST: Khoes/Delete/5
461 [HttpPost, ActionName("XoaKho")]
462 [ValidateAntiForgeryToken]
463 0 references
464 public ActionResult XoaKhoConfirmed(string id){...}
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517

```

Action Method <ChiTietKho>: Hiển thị danh sách chi tiết kho của 1 kho bất kỳ liên quan đến module Quản lý kho hàng.

```

204     // Chi tiết kho
205     public ActionResult ChiTietKho(string id)
206     {
207         TempData["makho"] = id;
208         return View(database.ChiTietKhoes.ToList().Where(s => s.MaKho == id));
209     }

```

Action Method <CreateCTKho>: Thực hiện chức năng tạo chi tiết kho mới:

```

199     [HttpPost]
200     [ValidateAntiForgeryToken]
201     public ActionResult CreateCTKho([Bind(Include = "MaCTKho,MaSP,MaKho,NgayNhap,NgayXuat,SoLuong,TinhTrang")] ChiTietKho chiTietKho)
202     {
203
204         if (ModelState.IsValid)
205         {
206             chiTietKho.MaKho = (string)Session["makho"];
207             //Random rd = new Random();
208             //var mactk = "CTK" + rd.Next(1, 1000);
209             //chiTietKho.MaCTKho = mactk;
210             if (chiTietKho.SoLuong <= 100 && chiTietKho.SoLuong > 0)
211             {
212                 chiTietKho.TinhTrang = "Sắp hết hàng";
213             }
214             else if (chiTietKho.SoLuong > 100 && chiTietKho.SoLuong < 1000)
215             {
216                 chiTietKho.TinhTrang = "Còn hàng";
217             }
218             //var ktra = (System.DateTime.Now - chiTietKho.NgayXuat);
219             else if(chiTietKho.SoLuong >= 1000)
220             {
221                 chiTietKho.TinhTrang = "Tồn kho";
222             }
223             else if (chiTietKho.SoLuong == 0)
224             {
225                 chiTietKho.TinhTrang = "Hết hàng";
226             }
227             else
228             {
229                 TempData["AlertMessage"] = "error null";
230                 return RedirectToAction("CreateCTKho",
231                                     new { controller = "PhongKho", action = "CreateCTKho", Id = chiTietKho.MaKho });
232             }
233             if(chiTietKho.NgayXuat == null)
234             {
235                 DateTime ngayDf = new DateTime(2000, 01, 01);
236                 chiTietKho.NgayXuat = ngayDf;
237             }
238
239             database.ChiTietKhoes.Add(chiTietKho);
240             database.SaveChanges();
241             updateTongTon();
242             TempData["AlertMessage"] = "Đã thêm";
243             TempData["MaCTKho"] = chiTietKho.STT;
244             return RedirectToAction("ChiTietKho",
245                                     new { controller = "PhongKho", action = "ChiTietKho", Id = chiTietKho.MaKho });
246         }
247     }
248     return View();
249 }

```

- Ngoài ra còn có các action method phục vụ chức năng chỉnh sửa chi tiết kho hoặc xóa chi tiết kho:

```

274     // GET: ChiTietKhoes/Edit/5
275     public ActionResult EditCTKho(int id){...}
276
277     // POST: ChiTietKhoes/Edit/5
278     // To protect from overposting attacks, enable the specific properties you want to bind to, for
279     // more details see https://go.microsoft.com/fwlink/?LinkId=317598.
280     [HttpPost]
281     [ValidateAntiForgeryToken]
282     public ActionResult EditCTKho([Bind(Include = "MaCTKho,MaSP,MaKho,NgayNhap,NgayXuat,SoLuong,TinhTrang")] ChiTietKho chiTietKho){...}
283
284     // GET: ChiTietKhoes/Delete/5
285     public ActionResult DeleteCTKho(int id){...}
286
287     // POST: ChiTietKhoes/Delete/5
288     [HttpPost, ActionName("DeleteCTKho")]
289     [ValidateAntiForgeryToken]
290     public ActionResult DeleteCTKhoConfirmed(int id){...}

```

Action Method <TaoPhieuNhapKho>: Thực hiện chức năng tạo phiếu đề nghị nhập kho khi có sản phẩm sắp hết hàng hoặc đã hết hàng:

```

519 [HttpPost]
520 [ValidateAntiForgeryToken]
521 public ActionResult TaoPhiieuNhapKho([Bind(Include = "MaPhiieu,MaKho,NgayLap,LoaiPhiieu,MaNVLap")] PhiieuNhapXuat phieuNhapXuat)
522 {
523     if (ModelState.IsValid)
524     {
525         Random rd = new Random();
526         var maphiieu = "KDPNK" + rd.Next(1, 1000);
527         phieuNhapXuat.MaPhiieu = maphiieu;
528         Session["phieunhap"] = maphiieu;
529         phieuNhapXuat.NgayLap = System.DateTime.Now;
530         phieuNhapXuat.LoaiPhiieu = "Phiếu đề nghị nhập hàng vào kho";
531         Session["KinhGui"] = " ";
532         Session["HoTen"] = " ";
533         var user = (Test02.Models.NhanVien)HttpContext.Session["user"];
534         phieuNhapXuat.MaNVLap = user.MaNV;
535         phieuNhapXuat.TinhTrang = "Chưa xét duyệt";
536
537         foreach (var item in database.ChiTietKhoes)
538         {
539             if(item.TinhTrang == "Sắp hết hàng" || item.TinhTrang == "Hết hàng")
540             {
541                 ChiTietPhiieuNhapXuat ctphieu = new ChiTietPhiieuNhapXuat();
542                 ctphieu.MaCT = "CT" + rd.Next(100, 1000);
543                 ctphieu.MaPhiieu = maphiieu;
544                 ctphieu.MaKho = item.MaKho;
545                 ctphieu.MaSP = item.MaSP;
546                 ctphieu.SoLuongTrongKho = item.SoLuong;
547                 ctphieu.SoLuongDeXuat = 513 - item.SoLuong;
548                 database.ChiTietPhiieuNhapXuats.Add(ctphieu);
549             }
550         }
551         database.PhiếuXuats.Add(phieuNhapXuat);
552         database.SaveChanges();
553         TempData["AlertMessage"] = "Đã thêm";
554         TempData["MaPNKho"] = maphiieu;
555         return RedirectToAction("ChiTietPhiieuNhapKho", new RouteValueDictionary(
556             new { controller = "PhongKho", action = "ChiTietPhiieuNhapKho", Id = maphiieu }));
557     }
}

```

Action Method <GiaiQuyetTonKho>: Thực hiện chức năng tạo phiếu đề nghị xuất kho khi có sản phẩm tồn kho:

```

630 [HttpPost]
631 [ValidateAntiForgeryToken]
632 public ActionResult GiaiQuyetTonKho([Bind(Include = "MaPhiieu,MaKho,NgayLap,LoaiPhiieu")] PhiieuNhapXuat phieuNhapXuat)
633 {
634     if (ModelState.IsValid)
635     {
636         Random rd = new Random();
637         var maphieux = "KDPXK" + rd.Next(1, 1000);
638         phieuNhapXuat.MaPhiieu = maphieux;
639         Session["phieuXuat"] = maphieux;
640         phieuNhapXuat.NgayLap = System.DateTime.Now;
641         phieuNhapXuat.LoaiPhiieu = "Phiếu đề nghị xuất hàng tồn";
642         Session["KinhGui"] = " ";
643         Session["HoTen"] = " ";
644         var user = (Test02.Models.NhanVien)HttpContext.Session["user"];
645         phieuNhapXuat.MaNVLap = user.MaNV;
646         phieuNhapXuat.TinhTrang = "Chưa xét duyệt";
647         //var ctphieu = database.ChiTietPhiieuNhapXuats.;
648
649         foreach (var item in database.ChiTietKhoes)
650         {
651             if (item.TinhTrang == "Tồn kho")
652             {
653                 ChiTietPhiieuNhapXuat ctphieu = new ChiTietPhiieuNhapXuat();
654                 ctphieu.MaCT = "CT" + rd.Next(100, 1000);
655                 ctphieu.MaPhiieu = maphieux;
656                 ctphieu.MaKho = item.MaKho;
657                 ctphieu.MaSP = item.MaSP;
658                 ctphieu.SoLuongTrongKho = item.SoLuong;
659                 ctphieu.SoLuongDeXuat = item.SoLuong - 532;
660                 database.ChiTietPhiieuNhapXuats.Add(ctphieu);
661             }
662         }
663         database.PhiếuXuats.Add(phieuNhapXuat);
664         database.SaveChanges();
665         TempData["AlertMessage"] = "Đã thêm";
666         TempData["MaPNKho"] = maphieux;
667         return RedirectToAction("ChiTietPhiieuXuatKho", new RouteValueDictionary(
668             new { controller = "PhongKho", action = "ChiTietPhiieuXuatKho", Id = maphieux }));
669     }
670     return View(phieuNhapXuat);
671 }

```

Action Method <BaoCao>: Thực hiện chức năng thống kê và hiển thị danh sách biên bảng kiểm kê/ báo cáo:

```

725     public ActionResult BaoCao()
726     {
727         //Chức vụ nhân viên lập biên bảng kiểm kê
728         TempData["tencv1"] = TempData["tencv"];
729         //tính tổng số kho
730         var totalkho = database.Khoes.ToList().Count;
731         TempData["TongKho"] = totalkho;
732         //tính tổng số sản phẩm
733         var totalsp = database.SanPhams.ToList().Count;
734         TempData["Tongsp"] = totalsp;
735         //tính tổng số Phiếu nhập xuất hàng
736         var tongpn = 0;
737         var tongpx = 0;
738         var pnx = database.PhieuNhapXuats.ToList();
739         foreach(var item in pnx) {
740             string str = item.MaPhieu.Substring(0, 4);
741             if(str == "KDPX") {
742                 tongpx = tongpx + 1;
743             }
744             else if (str == "KDPN") {
745                 tongpn = tongpn + 1;
746             }
747         }
748         TempData["Tongpn"] = tongpn;
749         TempData["Tongpx"] = tongpx;
750         //tính tổng sản phẩm sắp hết hàng
751         var dssphh = database.ChiTietKhoes.ToList();
752         var total2 = 0;
753         foreach (var item in dssphh){
754             if (item.TinhTrang == "Sắp hết hàng"){
755                 total2 = total2 + 1;
756             }
757             else if (item.TinhTrang == "Hết hàng"){
758                 total2 = total2 + 1;
759             }
760         }
761         TempData["TongSPHH"] = total2;
762         //tính tổng sản phẩm tồn kho
763         var total3 = 0;
764         foreach (var item in dssphh) {
765             if (item.TinhTrang == "Tồn kho"){
766                 total3 = total3 + 1;
767             }
768         }
769         TempData["TongSPTK"] = total3;
770         //var bienBangKiemKes = database.BienBangKiemKes.Include(b => b.NhanVien).Include(b => b.SanPham);
771         return View(database.BienBangKiemKes.ToList().OrderByDescending(s => s.NgayLap));
772     }
773 }
```

### Action Method <CreateBBKK>: Thực hiện chức năng tạo biên bảng kiểm kê

```

786 [HttpPost]
787 [ValidateAntiForgeryToken]
788 public ActionResult CreateBBKK([Bind(Include = "MaKK,Makho,ManVLap,TenBienBang,NgayLap")] BienBangKiemKe bienBangKiemKe)
789 {
790     DataTable bb = new DataTable();
791     if (Session["BBKK"] == null)
792     {
793         bb.Columns.Add("MaKK");
794         bb.Columns.Add("MaKho");
795         bb.Columns.Add("ManVLap");
796         bb.Columns.Add("NgayLap");
797         bb.Columns.Add("TenBB");
798         //bb.Columns.Add("ManVLap");
799         //Tạo xong lưu lại Session
800         Session["BBKK"] = bb;
801     }
802     else
803     {
804         //Lấy thông tin từ Session
805         bb = Session["BBKK"] as DataTable;
806     }
807     if (ModelState.IsValid)
808     {
809         Random rd = new Random();
810         var mabb = "BB" + rd.Next(1, 1000);
811         bienBangKiemKe.MaKK = mabb;
812         Session["tempdata"] = mabb;
813         bienBangKiemKe.NgayLap = System.DateTime.Now;
814         if(bienBangKiemKe.TenBienBang != null)
815         {
816             DataRow dr = bb.NewRow();
817             dr["MaKK"] = bienBangKiemKe.MaKK;
818             dr["MaVLap"] = bienBangKiemKe.ManVLap;
819             dr["MaKho"] = bienBangKiemKe.Makho;
820             dr["TenBB"] = bienBangKiemKe.TenBienBang;
821             dr["NgayLap"] = bienBangKiemKe.NgayLap;
822             bb.Rows.Add(dr);
823             //database.BienBangKiemKes.Add(bienBangKiemKe);
824             //database.SaveChanges();
825
826             return RedirectToAction("CreateCTBBKK");
827         }
828         else
829         {
830             TempData["AlertMessage"] = "null";
831             return RedirectToAction("CreateBBKK");
832         }
833     }
834
835     ViewBag.Makho = new SelectList(database.Khoes, "MaKho", "TenKho", bienBangKiemKe.Makho);
836     ViewBag.ManVLap = new SelectList(database.NhanViens, "MaNV", "MaNV", bienBangKiemKe.ManVLap);
837     return View(bienBangKiemKe);
838 }
```

Action Method <CreateCTBBKK>: Thực hiện chức năng tạo chi tiết biên bảng kiểm kê

```

851  [HttpPost]
852  [ValidateAntiForgeryToken]
853  public ActionResult CreateCTBBKK([Bind(Include = "STT,MaKK,MaSP,SLTonKho,SLThucTe,ChenhLech,LyDo")] ChiTietBienBang chiTietBienBang)
854  {
855      if (ModelState.IsValid)
856      {
857          if(chiTietBienBang.SLThucTe != null && chiTietBienBang.LyDo != null)
858          {
859              chiTietBienBang.MaKK = (string)Session["tempdata"];
860              Random rd = new Random();
861              //var mactbb = "CTBB" + rd.Next(1, 1000);
862              chiTietBienBang.STT = rd.Next(1, 1000);
863              BienBangKiemKe bienBangKiemKe = new BienBangKiemKe();
864
865              DataTable bb = Session["BBKK"] as DataTable;
866
867              foreach (DataRow dr in bb.Rows)
868              {
869                  bienBangKiemKe.MaKK = dr["MaKK"].ToString();
870                  bienBangKiemKe.Makho = dr["Makho"].ToString();
871                  bienBangKiemKe.MaNVLap = dr["MaNVLap"].ToString();
872                  var date = Convert.ToDateTime(dr["NgayLap"]);
873                  bienBangKiemKe.NgayLap = date;
874                  bienBangKiemKe.TenBienBang = dr["TenBB"].ToString();
875              }
876              var makhofromds = database.ChiTietKhoes.ToList();
877              var slton = 0;
878              foreach (var item in makhofromds)
879              {
880                  if (item.Makho == bienBangKiemKe.Makho & item.MaSP == chiTietBienBang.MaSP)
881                  {
882                      slton += (int)item.SoLuong;
883                  }
884              }
885              chiTietBienBang.SLTonKho = slton;
886              chiTietBienBang.ChenhLech = Math.Abs((int)chiTietBienBang.SLTonKho - (int)chiTietBienBang.SLThucTe) ;
887
888              database.BienBangKiemKes.Add(bienBangKiemKe);
889              database.SaveChanges();
890              database.ChiTietBienBangs.Add(chiTietBienBang);
891              database.SaveChanges();
892              TempData["AlertMessage"] = "Đã thêm";
893              return RedirectToAction("BaoCao");
894          }
895      }
896      else
897      {
898          TempData["AlertMessage"] = "null";
899          return RedirectToAction("CreateCTBBKK");
900      }
901  }
902  ViewBag.MaKK = new SelectList(database.BienBangKiemKes, "MaKK", "MaKK", chiTietBienBang.MaKK);
903  ViewBag.MaSP = new SelectList(database.SanPhams, "MaSP", "TenSP", chiTietBienBang.MaSP);
904  return View(chiTietBienBang);
905 }
```

Action Method <CreatePXuat>: Thực hiện chức năng tạo phiếu xuất kho theo đơn hàng đã có hóa đơn

```

972     public ActionResult CreatePXuat(string id)
973     {
974         TempData["madhx"] = id;
975         TempData["ngaylap"] = System.DateTime.Now;
976         var maDH = id;
977         var donhang = database.DonHangs.Where(s => s.MaDH == maDH).FirstOrDefault();
978         List<ChiTietDonHang> ctdonhang = database.ChiTietDonHangs.Where(s => s.MaDH == maDH).ToList();
979         var kho = database.Khoes.ToList();
980
981         List<Kho> khoutien = new List<Kho>(); //set danh sách kho ưu tiên
982         int i = 0; //Biến đếm
983         //kiểm tra kho có chứa sản phẩm của đơn hàng không?
984         foreach (var item in kho)
985         {
986             //tạo danh sách chi tiết kho của kho item
987             List<ChiTietKho> ctk1 = database.ChiTietKhoes.Where(s=> s.Makho == item.Makho).ToList();
988             foreach(var itemdh in ctdonhang)
989             {
990                 foreach(var item2 in ctk1)
991                 {
992                     //Kiểm tra sản phẩm có số lượng trong kho phải >= số lượng sản phẩm của đơn hàng
993                     if(item2.MaSP == itemdh.MaSP && item2.SoLuong >= itemdh.SoLuong){
994                         //Nếu thỏa thì tăng i lên 1
995                         i += 1;
996                         break;
997                     }
998                 }
999             }
999             //Nếu i = số lượng đơn hàng thì add kho item vào ds kho ưu tiên
1000             if (i >= ctdonhang.Count())
1001             {
1002                 i = 0; //trả lại i=0 để xét tiếp kho tiếp theo
1003                 khoutien.Add(item);
1004             }
1005             else i = 0;
1006         }
1007         foreach (var item in database.DonHangs){
1008             if (maDH == item.MaDH){
1009                 TempData["maDL"] = item.Daily.MaDL;
1010                 TempData["tenDL"] = item.Daily.TenDL;
1011                 TempData["diemgiao"] = item.DiemGiao;
1012             }
1013         }
1014         ViewBag.Makho = new SelectList(khoutien, "Makho", "Makho");
1015         if(khoutien.Count() == 0)
1016         {
1017             ViewBag.Makho = " ";
1018             TempData["AlertMessage"] = "noKho";
1019             return RedirectToAction("ChiTietDonHang", new RouteValueDictionary(
1020                 new { controller = "PhongKho", action = "ChiTietDonHang", Id = Session["madhxk"] }));
1021         }
1022         ViewBag.MaNVLap = new SelectList(database.NhanViens, "MaNV", "MaChucVu");
1023         ViewBag.MaDH = new SelectList(database.DonHangs, "MaDH", "MaDH");
1024         return View();
1025     }

```

Action Method:

- <DSDonHang>: Hiển thị danh sách đơn hàng đã xét duyệt
- <ChiTietDonHang>: Hiển thị chi tiết đơn hàng, nhận vào mã đơn hàng

```

1108     //đơn hàng
1109     public ActionResult DSDonHang()
1110     {
1111         return View();
1112     }
1113     public ActionResult ChiTietDonHang(string id)
1114     {
1115         Session["madhxk"] = id;
1116         TempData["madh"] = id;
1117         return View(database.ChiTietDonHangs.ToList().Where(s => s.MaDH == id));
1118     }

```

Action Method <NhapKho>: Thực hiện chức năng hiển thị tất cả phiếu nhập xuất đã tạo:

```
641 0 references
642 public ActionResult NhapKho()
643 {
644     return View(database.PhieuNhapXuats.ToList().OrderByDescending(s => s.NgayLap));
645 } // GET: PhieuNhapXuats/Create
```

Action Method <PhanPhoiSanPham>: Thực hiện chức năng phân phối sản phẩm giữa 2 kho:

```

1278     //Phân phối sản phẩm
1279     public ActionResult PhanPhoiSanPham(string id)
1280     {
1281         Session["PPMaKho"] = id;
1282         return View(database.ChiTietKhoes.ToList().Where(s => s.MaKho == id));
1283     }
1284
1285     [HttpPost]
1286     [ValidateAntiForgeryToken]
1287     public ActionResult PhanPhoiSanPham(string id, int? STT, string MaKhoNhan, int? SLNhan)
1288     {
1289         var ctkho = database.ChiTietKhoes.Where(s => s.STT == STT).FirstOrDefault();
1290
1291         if (ctkho.SoLuong == 0)
1292         {
1293             TempData["AlertMessage"] = "Invalid1";
1294             return RedirectToAction("PhanPhoiSanPham");
1295         }else if (SLNhan == null || SLNhan > ctkho.SoLuong)
1296         {
1297             TempData["AlertMessage"] = "Invalid2";
1298             return RedirectToAction("PhanPhoiSanPham");
1299         }
1300         //trừ số lượng của sản phẩm trong kho cần xuất
1301         ctkho.SoLuong -= SLNhan;
1302         database.Entry(ctkho).State = System.Data.EntityState.Modified;
1303         database.SaveChanges();
1304         //Tạo chi tiết kho mới của kho nhận
1305         ChiTietKho chiTietKho = new ChiTietKho();
1306         chiTietKho.MaSP = ctkho.MaSP;
1307         chiTietKho.Makho = Makholhan;
1308         chiTietKho.NgayNhap = System.DateTime.Now;
1309         DateTime ngayDf = new DateTime(2000, 01, 01);
1310         chiTietKho.NgayXuat = ngayDf;
1311         chiTietKho.SoLuong = SLNhan;
1312         if (chiTietKho.SoLuong <= 100 && chiTietKho.SoLuong > 0)
1313         {
1314             chiTietKho.TinhTrang = "Sắp hết hàng";
1315         }
1316         else if (chiTietKho.SoLuong > 100 && chiTietKho.SoLuong < 1000)
1317         {
1318             chiTietKho.TinhTrang = "Còn hàng";
1319         }
1320         else if (chiTietKho.SoLuong >= 1000)
1321         {
1322             chiTietKho.TinhTrang = "Tồn kho";
1323         }
1324         else if (chiTietKho.SoLuong == 0)
1325         {
1326             chiTietKho.TinhTrang = "Hết hàng";
1327         }
1328         else
1329         {
1330             chiTietKho.TinhTrang = "SP phân phối";
1331         }
1332         database.ChiTietKhoes.Add(chiTietKho);
1333         database.SaveChanges();
1334         TempData["AlertMessage"] = "Valid";
1335         return RedirectToAction("PhanPhoiSanPham");
1336     }
1337 }
1338 }
```

Action Method <updateTongTon>: Thực hiện chức năng cập nhật tổng tồn, không có view để hiển thị:

```

148     5 references
149     public ActionResult updateTongTon()
150     {
151         //set tổng tồn từng sp theo từng chi tiết kho
152         var chiTietKhos = database.ChiTietKhoses.ToList();
153         var sanPhams = database.SanPhams.ToList();
154         foreach (var sp in sanPhams)
155         {
156             sp.TongTon = 0;
157             foreach (var ctk in chiTietKhos)
158             {
159                 if (sp.MaSP == ctk.MaSP)
160                 {
161                     sp.TongTon += ctk.SoLuong;
162                 }
163                 database.Entry(sp).State = System.Data.Entity.EntityState.Modified;
164                 database.SaveChanges();
165             }
166             return View();
167         }
168     }

```

### Action Method khi thêm kho mới:

- <CheckTenKhoAvailability>: Thực hiện chức năng cập kiểm tra tên kho có bị trùng với tên kho đã lưu hay không.
- <CheckDiaChiKhoAvailability>: Thực hiện chức năng cập kiểm tra địa chỉ kho có bị trùng với địa chỉ các kho đã lưu hay không.

```

375     0 references
376     public JsonResult CheckTenKhoAvailability(string namekho)
377     {
378         System.Threading.Thread.Sleep(200);
379         var tenKho = database.Kholes.Where(s => s.TenKho == namekho).SingleOrDefault();
380         if(tenKho != null)
381         {
382             return Json(1);
383         }
384         else
385         {
386             return Json(0);
387         }
388     }
389     0 references
390     public JsonResult CheckDiaChiAvailability(string diachi)
391     {
392         System.Threading.Thread.Sleep(200);
393         var diaChi = database.Kholes.Where(s => s.DiaChi == diachi).SingleOrDefault();
394         if (diaChi != null)
395         {
396             return Json(1);
397         }
398         else
399         {
400             return Json(0);
401         }

```

Action Method <updateStatusPhieu>: Thực hiện chức năng cập nhật trạng thái phiếu nhập xuất:

```

1165 //cập nhật trạng thái phiếu để nghị khi đã xét duyệt
1166 [HttpPost]
1167 [ValidateAntiForgeryToken]
1168 public ActionResult UpdateStatusPhieu(string id)
{
    PhieuNhapXuat phieu = database.PhiếuNhapXuats.Find(id);
    var ctpheu = database.ChiTietPhiếuNhapXuats.Where(s => s.MaPhiếu == id).ToList();
    string str = phieu.MaPhiếu.Substring(0, 4);
    if (str == "KDPX")
    {
        foreach (var item in ctpheu)
        {
            var ctkho = database.ChiTietKhoes.Where(s => s.MaKho == item.MaKho && s.MaSP == item.MaSP && s.SoLuong == item.SoLuongTrongKho).FirstOrDefault();
            if (ctkho == null)
            {
                break;
            }
            ctkho.SoLuong -= item.SoLuongDeXuat;
            ctkho.TìnhTrạng = "Còn hàng";
            updateTongTon();
        }
        phieu.TìnhTrạng = "Đã xuất";
        database.Entry(phieu).State = System.Data.Entity.EntityState.Modified;
        database.SaveChanges();
    }
    else if(str == "KDPN")
    {
        foreach (var item in ctpheu)
        {
            var ctkho = database.ChiTietKhoes.Where(s => s.MaKho == item.MaKho && s.MaSP == item.MaSP && s.SoLuong == item.SoLuongTrongKho).FirstOrDefault();
            if(ctkho == null)
            {
                break;
            }
            ctkho.SoLuong += item.SoLuongDeXuat;
            ctkho.TìnhTrạng = "Còn hàng";
            updateTongTon();
        }
        phieu.TìnhTrạng = "Đã nhập";
        database.Entry(phieu).State = System.Data.Entity.EntityState.Modified;
        database.SaveChanges();
    }
}
 TempData["AlertMessage"] = "status";
 var phieuXu = database.PhiếuNhapXuats.ToList();
 return RedirectToAction("NhậpKho");
}

```

#### 4.2.2.4. Quản trị viên:

```

0 references
public ActionResult HomePage()
{
    var lstnv = database.NhanViens.Where(s => s.MaChucVu == "QTHT").ToList();
    return View(lstnv);
}

0 references
public ActionResult QLNhanVien()
{
    return View(database.NhanViens.ToList().OrderByDescending(s => s.MaNV));
}

```

Action Method:

- HomePage: hiển thị trang chủ quản trị viên
- QLNhanVien: hiển thị danh sách nhân viên trong hệ thống

```

0 references
public JsonResult CheckUsernameAvailability(string userdata)
{
    System.Threading.Thread.Sleep(200);
    var SeachData = database.NhanViens.Where(x => x.UserName == userdata).SingleOrDefault();

    if (SeachData != null)
    {
        return Json(1);
    }
    else
    {
        return Json(0);
    }
}

0 references
public JsonResult CheckEmailAvailability(string usermail)
{
    System.Threading.Thread.Sleep(200);
    var SeachData = database.NhanViens.Where(x => x.Email == usermail).SingleOrDefault();

    if (SeachData != null)
    {
        return Json(1);
    }
    else
    {
        return Json(0);
    }
}

```

#### Action Method:

- CheckEmailAvailability: kiểm tra trùng email nhân viên
- CheckUsernameAvailability: kiểm tra trùng tên user đăng nhập nhân viên

```

0 references
public JsonResult CheckSDTAvailability(string userSDT)
{
    System.Threading.Thread.Sleep(200);
    var SeachData = database.NhanViens.Where(x => x.SDT == userSDT).SingleOrDefault();

    if (SeachData != null)
    {
        return Json(1);
    }
    else
    {
        return Json(0);
    }
}

0 references
public JsonResult KtraPass(string pass1, string pass2)
{
    System.Threading.Thread.Sleep(200);
    if (pass1 != pass2)
    {
        return Json(1);
    }
    else
    {
        return Json(0);
    }
}

```

#### Action Method:

- CheckSDTAvailability: kiểm tra trùng số điện thoại nhân viên
- KtraPass: kiểm tra xác nhận nhập mật khẩu

```

0 references
public ActionResult ThemNV()
{
    ViewBag.MaChucVu = new SelectList(database.ChucVus, "MaChucVu", "MaChucVu");
    return View();
}

[HttpPost]
0 references
public ActionResult ThemNV(NhanVien nhanVien,HttpPostedFileBase HinhAnh)
{
    if (ModelState.IsValid)
    {
        if (HinhAnh != null)
        {
            var fileName = Path.GetFileName(HinhAnh.FileName);
            //Tạo đường dẫn tới file
            var path = Path.Combine(Server.MapPath("~/Data/Images"), fileName);
            //Lưu tên
            nhanVien.HinhAnh = fileName;
            //Save vào Images Folder
            HinhAnh.SaveAs(path);
        }
        else
        {
            nhanVien.HinhAnh = "account.png";
        }
        Random rd = new Random();
        var manv = "NV" + rd.Next(1, 1000);
        nhanVien.MaNV = manv;
        database.NhanViens.Add(nhanVien);
        database.SaveChanges();
        TempData["thongbao"] = "add";
        return RedirectToAction("QLNhanVien");
    }
    return View(nhanVien);
}

```

#### Action Method:

- ThemNV: hiển thị view thêm mới nhân viên
- ThemNV (HttpPost): thực hiện thêm nhân viên vào hệ thống

```

0 references
public ActionResult ChinhSuaNV(String id,NhanVien nhanVien)
{
    ViewBag.MaChucVu = new SelectList(database.ChucVus, "MaChucVu", "MaChucVu", nhanVien.MaChucVu);
    return View(database.NhanViens.Where(s => s.MaNV == id).FirstOrDefault());
}

[HttpPost]
0 references
public ActionResult ChinhSuaNV(NhanVien nv, HttpPostedFileBase Hinhanh, string imgnv)
{
    if (ModelState.IsValid)
    {
        if (Hinhanh != null)
        {
            var fileName = Path.GetFileName(Hinhanh.FileName);
            var path = Path.Combine(Server.MapPath("~/Data/Images"), fileName);
            nv.Hinhanh = fileName;
            //Save vào Images Folder
            Hinhanh.SaveAs(path);

        }
        else
        {
            nv.Hinhanh = imgnv;
        }
        database.Entry(nv).State = System.Data.Entity.EntityState.Modified;
        database.SaveChanges();
        TempData["thongbao"] = "edit";
        return RedirectToAction("QLNhanVien");
    }
    return View(nv);
}

```

#### Action Method:

- ChinhSuaNV: hiển thị view cập nhật thông tin nhân viên
- ChinhSuaNV(HttpPost): thực hiện cập nhật thông tin nhân viên

```

0 references
public ActionResult ChitietNV(string id)
{
    var nhanvien = database.NhanViens.Where(s => s.MaNV == id).FirstOrDefault();
    return View(nhanvien);
}

```

```

0 references
public ActionResult XoaNV(string id,NhanVien nhanvien)
{
    nhanvien = database.NhanViens.Where(s => s.MaNV == id).FirstOrDefault();
    database.NhanViens.Remove(nhanvien);
    database.SaveChanges();
    TempData["thongbao"] = "delete";
    return RedirectToAction("QLNhanVien");
}

```

#### Action Method:

- ChitietNV: hiển thị thông tin chi tiết nhân viên
- XoaNV: thực hiện xóa nhân viên

#### 4.2.2.1. Đại lý (khách hàng):

```
public ActionResult PageSanPham(int? page, string query)
{
    TempData["timkiem"] = query;
    Session["page"] = page;
    Session["query"] = query;
    int pageSize = 6;
    int pageNum = (page ?? 1);
    if (query == null || query == "")
    {
        return View(db.SanPhams.ToList().OrderByDescending(s => s.TenSP).ToPagedList(pageNum, pageSize));
    }

    if (query.Length == 1)
    {
        var data = db.SanPhams.Where(s => s.MaSP.Substring(0, 1) == query || s.TenSP.Substring(0, 1) == query).ToList().OrderByDescending(s => s.TenSP).ToPagedList(pageNum, pageSize);
        return View(data);
    }
    else
    {
        var data = db.SanPhams.Where(s => s.MaSP.Contains(query) || s.TenSP.Contains(query)).ToList().OrderByDescending(s => s.TenSP).ToPagedList(pageNum, pageSize);
        return View(data);
    }
}
```

#### Action Method:

- PageSanPham: Thực hiện mở trang PageSanPham

```
// Chi tiết sản phẩm
public ActionResult ChiTietSP(string id)
{
    TempData["masp"] = id;
    return View(db.SanPhams.Where(s => s.MaSP == id).FirstOrDefault());
}

// Giỏ hàng
public ActionResult GioHangDL()
{
    if (Session["Cart"] == null)
        return View();
    Cart cart = Session["Cart"] as Cart;
    return View(cart);
}

// Tạo mới giỏ hàng
public Cart GetCart()
{
    Cart cart = Session["Cart"] as Cart;
    if (cart == null || Session["Cart"] == null)
    {
        cart = new Cart();
        Session["Cart"] = cart;
    }
    return cart;
}
```

#### Action Method:

- ChiTietSP: Thực hiện mở trang ChiTietSP
- GioHangDL: Thực hiện mở trang GioHangDL
- GetCart: Thực hiện tạo mới giỏ hàng

```

//Thêm vào giỏ hàng
public ActionResult ThemVaoGH(string id)
{
    var sp = db.SanPhams.SingleOrDefault(s => s.MaSP == id);

    if (sp.TongTon == 0 || sp.TongTon == null)
    {
        TempData["hethang"] = "Sản phẩm đã hết hàng!!!";
    }
    else if (sp.TongTon > 0)
    {
        TempData["Themvaogiohang"] = "thanhcong";
        GetCart().Add(sp);
    }
    return RedirectToAction("PageSanPham", new { page = Session["page"], query = Session["query"] });
}

public ActionResult ThemVaoGHViewChiTietSP(string id)
{
    var sp = db.SanPhams.SingleOrDefault(s => s.MaSP == id);

    if (sp.TongTon == 0 || sp.TongTon == null)
    {
        TempData["hethang"] = "Sản phẩm đã hết hàng!!!";
    }
    else if (sp.TongTon > 0)
    {
        TempData["Themvaogiohang"] = "thanhcong";
        GetCart().Add(sp);
    }
    return RedirectToAction("ChiTietSP/" + id, "KhachHang");
}

```

### Action Method:

- ThemVaoGioHang: Thực hiện thêm sản phẩm vào giỏ hàng và trả lại trang PageSanPham
- ThemVaoGHViewChiTietSP: Thực hiện thêm sản phẩm vào giỏ hàng và trả lại trang ChiTietSP

```

[HttpPost]
public ActionResult CapNhatSL(string MaSP, int soluong)
{
    var cart = GetCart();
    var maSP = db.SanPhams.FirstOrDefault(s => s.MaSP == MaSP);

    cart.CapNhatSL(MaSP, soluong);

    double tongTien = cart.TongTien();
    double thanhTien = cart.ThanhTien(maSP.MaSP);

    var result = new
    {
        tongTien = "Tổng tiền: " + tongTien.ToString("#,#0") + " VND",
        thanhTien = thanhTien.ToString("#,#0") + " VND",
    };
    return Json(result);
}

//Xóa sản phẩm khỏi giỏ hàng
public ActionResult XoaSP(string id)
{
    TempData["XoaSP"] = "Xóa sản phẩm thành công!!!";
    Cart cart = Session["Cart"] as Cart;
    cart.XoaSP(id);
    Session["Cart"] = cart;
    return RedirectToAction("GioHangDL", "KhachHang");
}

```

### Action Method:

- CapNhatSL: Thực hiện cập nhật số lượng sản phẩm trong giỏ hàng
- XoaSP: Thực hiện xóa sản phẩm ra khỏi giỏ hàng

```

public double TinhCongNo(string MaDL)
{
    List<PhieuCongNo> phieuCongNos = db.PhieuCongNoes.Where(s => s.MaDL == MaDL && s.TrangThai == "Chưa thanh toán").ToList();

    double tongCN = 0;
    foreach(var i in phieuCongNos)
    {
        tongCN += (double)i.TienNo;
    }
    return tongCN;
}

public double MaxNo(string maLoaiDL)
{
    double congNo = 0;
    if (maLoaiDL == "LDL01")
    {
        congNo = 500000000;
    }
    else if (maLoaiDL == "LDL02")
    {
        congNo = 250000000;
    }
    else
    {
        congNo = 100000000;
    }
    return congNo;
}

```

**Action Method:**

- TinhCongNo: Thực hiện tính công nợ trả về tổng công nợ
- MaxNo: Thực hiện tính số tiền nợ tối đa trả về số tiền nợ tối đa

```

//Trang đặt hàng
public ActionResult PageDatHang()
{
    Cart cart = Session["Cart"] as Cart;

    var daily = (Test02.Models.Daily)HttpContext.Session["userDL"];

    if (cart == null || cart.Items.Count() <= 0)
    {
        TempData["GHNLL"] = "Vui lòng chọn sản phẩm!!!";
        return RedirectToAction("GioHangDL", "KhachHang");
    }

    else if(cart.Items != null)
    {
        double tongphu = cart.TongTien();
        double giam = tongphu * (double)daily.LoaiDL.ChietKhau;
        double thanhtien = tongphu - giam;

        double tongTienCN = TinhCongNo(daily.MaDL);
        double maxCN = MaxNo(daily.MaLoaiDL);
        double tong = tongphu + tongTienCN;

        var gioHang = cart.Items.ToList();
        foreach (var item in gioHang)
        {
            if (item.soluong > item.idSP.TongTon || item.soluong < 0)
            {
                TempData["ErrorSL"] = "Số lượng không hợp lệ";
                return RedirectToAction("GioHangDL", "KhachHang");
            }
        }

        if (tong >= maxCN)
        {
            TempData["MaxCN"] = "Công nợ vượt quá mức quy định, vui lòng thanh toán công nợ trước khi đặt hàng";
            return RedirectToAction("GioHangDL", "KhachHang");
        }
        else
        {
            TempData["giam"] = string.Format("{0:0,0}", giam);
            TempData["thanhtien"] = string.Format("{0:0,0}", thanhtien);
            return View(cart);
        }
    }
    return View();
}

```

**Action Method:**

- PageDatHang: Thực hiện mở trang PageDatHang

```

//Đặt hàng
public ActionResult DatHang(FormCollection form)
{
    Random rd = new Random();
    Cart cart = Session["Cart"] as Cart;
    DonHang donHang = new DonHang();
    var daiLy = (Test02.Models.DaiLy)HttpContext.Session["userDL"];
    //Tính tổng tiền
    double tongphu = cart.TongTien();
    //Tính chiết khấu
    double giam = tongphu * (double)daiLy.LoaiDL.ChietKhau;
    //Tính thành tiền
    double thanhtien = tongphu - giam;

    //Thêm vào bảng đơn hàng
    var maDH = "DH" + rd.Next(0, 9) + rd.Next(0, 9) + rd.Next(0, 9) + rd.Next(0, 9);

    donHang.MaDH = maDH;
    donHang.MaDL = daiLy.MaDL;
    donHang.MaNVLap = null;
    donHang.NgayLap = DateTime.Now;
    donHang.TrangThai = "Chưa xử lý";
    donHang.TinhTrangThanhToan = "Chưa thanh toán";
    donHang.DiemGiao = daiLy.DiaChi;
    donHang.TongTien = (float)thanhtien;
    donHang.PhieuXuatKho = false;
    donHang.XuatHoaDon = false;
    db.DonHangs.Add(donHang);

    //Thêm vào bảng chi tiết đơn hàng
    Random maCTDH = new Random();
    foreach (var item in cart.Items)
    {
        var idCTDH = maCTDH.Next(1, 9) + maCTDH.Next(10, 100);
        ChiTietDonHang chiTietDonHang = new ChiTietDonHang();
        chiTietDonHang.MaCTDH = idCTDH;
        chiTietDonHang.MaDH = maDH;
        chiTietDonHang.MaSP = item.idSP.MaSP;
        chiTietDonHang.SoLuong = item.soLuong;
        chiTietDonHang.ThanhTien = item.soLuong * item.idSP.Gia.Value;
        db.ChiTietDonHangs.Add(chiTietDonHang);
    }
    db.SaveChanges();
    cart.XoaSauKhiDat();
    TempData["muahangthanhcong"] = "Đặt hàng thành công";
    return RedirectToAction("PageSanPham", "KhachHang");
}

```

#### Action Method:

- DatHang: Thực hiện đặt hàng và trả về trang PageSanPham

```
// Danh sách đơn hàng
public ActionResult DonHangDL()
{
    var daily = (Test02.Models.Daily)HttpContext.Session["userDL"];
    var dsDH = db.DonHangs.Where(s => s.MaDL == daily.MaDL).ToList();
    return View(dsDH);
}

public ActionResult DSChoGiao()
{
    var daily = (Test02.Models.Daily)HttpContext.Session["userDL"];
    var dsDH = db.DonHangs.Where(s => s.MaDL == daily.MaDL && s.TinhTrangGH == "Chờ giao").ToList();
    return View(dsDH);
}

public ActionResult DSDangGiao()
{
    var daily = (Test02.Models.Daily)HttpContext.Session["userDL"];
    var dsDH = db.DonHangs.Where(s => s.MaDL == daily.MaDL && s.TinhTrangGH == "Đang giao").ToList();
    return View(dsDH);
}

public ActionResult DSDHDaGiao()
{
    var daily = (Test02.Models.Daily)HttpContext.Session["userDL"];
    var dsDH = db.DonHangs.Where(s => s.MaDL == daily.MaDL && s.TinhTrangGH == "Đã giao").ToList();
    return View(dsDH);
}
```

**Action Method:**

- DonHangDL: Thực hiện mở trang DonHangDL
- DSChoGiao: Thực hiện mở trang DSChoGiao
- DSDangGiao: Thực hiện mở trang DSDangGiao
- DSDHDaGiao: Thực hiện mở trang DSDHDaGiao

```
// Chi tiết đơn hàng
public ActionResult DetailsDH(string id)
{
    TempData["madh"] = id;
    var maDH = db.DonHangs.FirstOrDefault(s => s.MaDH == id);
    TempData["DiaDiemGH"] = maDH.DiemGiao;
    return View(db.ChiTietDonHangs.Where(s => s.MaDH == id).ToList());
}

// Xóa đơn hàng
public ActionResult DeleteDH(string id)
{
    if (id == null)
    {
        return new HttpStatusCodeResult(System.Net.HttpStatusCode.BadRequest);
    }
    DonHang donHang = db.DonHangs.Find(id);
    if (donHang == null)
    {
        return HttpNotFound();
    }
    return View(donHang);
}

[HttpPost, ActionName("DeleteDH")]
[ValidateAntiForgeryToken]
public ActionResult DeleteDHConfirmed(string id)
{
    DonHang donHang = db.DonHangs.Find(id);
    db.DonHangs.Remove(donHang);
    db.SaveChanges();
    return RedirectToAction("DonHangDL");
}
```

**Action Method:**

- DetailsDH: Thực hiện mở trang DetailsDH
- DeleteDH: Thực hiện mở trang DeleteDH

- DeleteDHConfirmed: Thực hiện xóa đơn hàng và trả về trang DonHangDL

```

public ActionResult CongNo()
{
    var daily = (Test02.Models.Daily)HttpContext.Session["userDL"];
    TempData["daily"] = daily.MaDL;
    var dsCongNo = db.PhieuCongNoes.Where(s => s.MaDL == daily.MaDL).ToList();

    var conNo = db.PhieuCongNoes.Where(s => s.TrangThai == "Chưa thanh toán" && s.MaDL == daily.MaDL);

    if (daily.MaLoaiDL == "LDL01")
    {
        ViewBag.CongNo = 500000000;
    }
    else if (daily.MaLoaiDL == "LDL02")
    {
        ViewBag.CongNo = 250000000;
    }
    else
    {
        ViewBag.CongNo = 100000000;
    }

    double tong = 0;
    foreach (var i in conNo)
    {
        tong += (double)i.TienNo;
    }
    TempData["TongCongNo"] = tong;
    return View(dsCongNo);
}

```

#### Action Method CongNo: Thực hiện mở trang CongNo

```

public ActionResult LienHe()
{
    return View();
}

public JsonResult CheckEmailAvailability(string email)
{
    System.Threading.Thread.Sleep(200);

    var mailCus = db.Dailies.Where(x => x.Email == email).SingleOrDefault();
    if (mailCus != null)
    {
        return Json(1);
    }
    else
    {
        return Json(0);
    }
}

```

#### Action Method:

- LienHe: Thực hiện mở trang LienHe
- CheckEmailAvailability: Thực hiện kiểm tra email và trả về thông báo

```

public JsonResult CheckNoPhoneAvailability(string noPhone)
{
    System.Threading.Thread.Sleep(200);

    var noPhoneCus = db.Dailies.Where(x => x.SDT == noPhone).SingleOrDefault();
    if (noPhoneCus != null)
    {
        return Json(1);
    }
    else
    {
        return Json(0);
    }
}

```

Action Method CheckNoPhoneAvailability: Thực hiện kiểm tra số điện thoại và trả về thông báo

```

public ActionResult ThongTinDL(string id)
{
    //var maDL = (Test02.Models.DaiLy)HttpContext.Session["userDL"];
    if (id == null)
    {
        return new HttpStatusCodeResult(HttpStatusCode.BadRequest);
    }
    Daily daily = db.Dailies.Find(id);
    if (daily == null)
    {
        return HttpNotFound();
    }
    return View(daily);
}

//Đổi hình ảnh đại lý
[HttpPost]
[ValidateAntiForgeryToken]
public ActionResult ThongTinDL(Daily daily, HttpPostedFileBase Hinhanh, string id)
{
    Daily info = db.Dailies.Find(id);

    var totalEmail = db.Dailies.Count(s => s.Email == daily.Email);
    var totalNoPhone = db.Dailies.Count(s => s.SDT == daily.SDT);

    if ((daily.SDT != info.SDT && totalNoPhone >= 1) ||
        (daily.Email != info.Email && totalEmail >= 1))
    {
        TempData["errorEmailPhone"] = "Không thể cập nhật, vì số điện thoại và email này đã được sử dụng";
        return View(daily);
    }

    else
    {
        LuuAnh(daily, Hinhanh);
        update.TenDL = daily.TenDL;
        update.Email = daily.Email;
        update.SDT = daily.SDT;
        update.DiaChi = daily.DiaChi;
        update.Hinhanh = daily.Hinhanh;
        Session["userDL"] = info;
        db.SaveChanges();
        TempData["UpdateTC"] = "Cập nhật đại lý thành công";
        return RedirectToAction("ThongTinDL");
    }
}
return RedirectToAction("ThongTinDL");
}

```

#### Action Method

- [HttpGet ThongTinDL](#): Thực hiện mở trang ThongTinDL
- [HttpPost ThongTinDL](#): Thực hiện cập nhật thông tin DL và trả về trang ThongTinDL

```

public ActionResult CapNhatMatKhau(string id)
{
    //var maDL = (Test02.Models.DaiLy)HttpContext.Session["userDL"];
    if (id == null)
    {
        return new HttpStatusCodeResult(HttpStatusCode.BadRequest);
    }
    Daily daily = db.Dailies.Find(id);
    if (daily == null)
    {
        return HttpNotFound();
    }
    return View(daily);
}

[HttpPost]
[ValidateAntiForgeryToken]
public ActionResult CapNhatMatKhau(Daily daily, string matkhau, string confirmPass, string id)
{
    //var maDL = (Test02.Models.DaiLy)HttpContext.Session["userDL"];

    var maDL = db.Dailies.Find(id);
    if (matkhau != maDL.Password)
    {
        TempData["errorPass"] = "Mật khẩu không chính xác";
        return RedirectToAction("CapNhatMatKhau");
    }

    if (confirmPass != daily.Password)
    {
        TempData["errorConfirm"] = "Xác thực mật khẩu không chính xác";
        return RedirectToAction("CapNhatMatKhau");
    }

    if (ModelState.IsValid)
    {
        Daily update = db.Dailies.Find(id);
        //db.Entry(daily).State = (System.Data.Entity.EntityState)System.Data.EntityState.Modified;
        update.Password = daily.Password;
        db.SaveChanges();
        TempData["Pass"] = "Đổi mật khẩu thành công!!!";
        return RedirectToAction("ThongTinDL/" + maDL.MaDL);
    }
    return View(daily);
}

```

#### Action Method:

- `HttpGet CapNhatMatKhau`: Thực hiện mở trang `CapNhatMatKhau`
- `HttpPost CapNhatMatKhau`: Thực hiện cập nhật mật khẩu

```

public void LuuAnh(Daily daily, HttpPostedFileBase Hinhanh)
{
    #region Hình ảnh
    if (Hinhanh == null)
    {
        daily.Hinhanh = daily.Hinhanh;
    }
    else
    {
        //Xác định đường dẫn lưu file : Url tương đối => tuyệt đối
        var urlTuongdoi = "/Data/ImagesDL/";
        var urlTuyetDoi = Server.MapPath(urlTuongdoi); // Lấy đường dẫn lưu file trên server

        //Check trùng tên file => Đổi tên file = tên file cũ (ko kèm đuôi)
        //Ảnh.jpg => ảnh + "-" + 1 + ".jpg" => ảnh-1.jpg

        string fullDuongDan = urlTuyetDoi + Hinhanh.FileName;
        int i = 1;
        while (System.IO.File.Exists(fullDuongDan) == true)
        {
            // 1. Tách tên và đuôi
            var ten = Path.GetFileNameWithoutExtension(Hinhanh.FileName);
            var duoi = Path.GetExtension(Hinhanh.FileName);
            // 2. Sử dụng biến i để chạy và cộng vào tên file mới
            fullDuongDan = urlTuyetDoi + ten + "-" + i + duoi;
            i++;
            // 3. Check lại
        }
    #endregion
    //Lưu file (Kiểm tra trùng file)
    Hinhanh.SaveAs(fullDuongDan);
    daily.Hinhanh = urlTuongdoi + Path.GetFileName(fullDuongDan);
}
}

```

Action Method LuuAnh: Thực hiện lưu ảnh vào thư mục hệ thống

#### 4.2.2.6. Phòng Giao Hàng:

```

[Authentication(MaChucVu=="NVGH")]
public class GiaoHangController : Controller
{
    QuanLyDLEntities2 database = new QuanLyDLEntities2();
    // GET: GiaoHang

    // Danh sách đơn hàng mới cần xét duyệt
    public ActionResult DonHangMoi()
    {
        return View(database.DonHangs.ToList().OrderByDescending(s => s.MaDH));
    }

    // Hiển thị chi tiết đơn hàng
    public ActionResult ChiTietDonHang(string id)
    {
        TempData["madh02"] = id;
        TempData["gh_ctdonhang02"] = database.ChiTietDonHangs.ToList().Where(s => s.MaDH == id);
        return View(TempData["gh_ctdonhang02"]);
    }
}

```

Action Method:

- DonHangMoi: Hiển thị danh sách đơn hàng
- ChiTietDonHang: Hiển thị chi tiết đơn hàng dựa trên mã đơn hàng

```

----//Hiển thị danh sách đơn hàng giao
----public ActionResult DonGiaoHang()
----{
----    //tính tổng đơn
----    var tongdonchogiao = 0;
----    var tongdondanggiao = 0;
----    var tongdondagiao = 0;
----    var dh = database.DonHangs.ToList();
----    foreach (var item in dh)
----    {
----        if (item.XuatHoaDon == true
----            && item.PhieuXuatKho == true
----            && item.TrangThai == "Đã xét duyệt"
----            && item.TinhTrangGH == "Chờ giao"
----            && item.MaGH != null)
----        {
----            tongdonchogiao = tongdonchogiao + 1;
----        }
----        else if (item.XuatHoaDon == true
----            && item.PhieuXuatKho == true
----            && item.TrangThai == "Đã xét duyệt"
----            && item.TinhTrangGH == "Đang giao"
----            && item.MaGH != null)
----        {
----            tongdondanggiao = tongdondanggiao + 1;
----        }
----        else if (item.XuatHoaDon == true
----            && item.PhieuXuatKho == true
----            && item.TrangThai == "Đã xét duyệt"
----            && item.TinhTrangGH == "Đã giao"
----            && item.MaGH != null)
----        {
----            tongdondagiao = tongdondagiao + 1;
----        }
----    }
----    TempData["TongDonChoGiao"] = tongdonchogiao;
----    TempData["TongDonDangGiao"] = tongdondanggiao;
----    TempData["TongDonDaGiao"] = tongdondagiao;

----    return View(database.DonHangs.ToList().OrderByDescending(s => s.MaDH));
----}

```

Action Method DonGiaoHang: Hiển thị danh sách đơn giao hàng

```

----//-----LẤP CÁC CHUYỀN GIAO (gom nhiều đơn hàng)-----
----//Danh sách các chuyên giao
----public ActionResult DanhSachCacChuyenGiao(ChuyenGiao chuyengiao)
----{
----    TempData["ngaylapcg"] = System.DateTime.Now;
----    TempData["dschuyengiao"] = database.ChuyenGiaos.ToList().OrderByDescending(s => s.MaGH);
----    return View(TempData["dschuyengiao"]);
----}

```

### Action Method DanhSachCacChuyenGiao: Hiển thị danh sách đơn đã giao

```

...//Hiển thị danh sách đơn đã giao
...public ActionResult DonDaGiao()
...
...{
...    return View(database.DonHangs.ToList().OrderByDescending(s => s.MaDH));
...}

...//Hiển thị danh sách đơn đã giao
...public ActionResult DonDangGiao()
...
...{
...    return View(database.DonHangs.ToList().OrderByDescending(s => s.MaDH));
...}

...//Danh sách đơn hàng hoàn
...public ActionResult DonChoGiao()
...
...{
...    return View(database.DonHangs.ToList().OrderByDescending(s => s.MaDH));
...}

...//Chỉnh sửa chuyến giao
...public ActionResult ChinhSuaChuyenGiao(string id)
...
...{
...    return View(database.ChuyenGaos.Where(s => s.MaGH == id).FirstOrDefault());
...}

...[HttpPost, ActionName("ChinhSuaChuyenGiao")]
...public ActionResult ChinhSuaChuyenGiao(ChuyenGiao chuyengiao, DonHang donhang,
...NhanVienGH shipper, PhuongTienGH phuongtien)
...
...{
...    donhang = database.DonHangs.Where(s => s.MaGH == chuyengiao.MaGH).FirstOrDefault();
...    shipper = database.NhanVienGHs.Where(s => s.MaGH == shipper.MaGH).FirstOrDefault();
...    phuongtien = database.PhuongTienGHs.Where(s => s.MaPT == phuongtien.MaPT).FirstOrDefault();
...    database.Entry(chuyengiao).State = System.Data.Entity.EntityState.Modified;
...    database.Entry(donhang).State = System.Data.Entity.EntityState.Modified;
...    database.Entry(shipper).State = System.Data.Entity.EntityState.Modified;
...    database.Entry(phuongtien).State = System.Data.Entity.EntityState.Modified;
...    if (chuyengiao.TrangThai == "Chờ giao")
...    {
...        donhang.TinhTrangGH = Convert.ToString("Chờ giao");
...        shipper.TinhTrang = Convert.ToString("Đang giao hàng");
...        phuongtien.TinhTrang = Convert.ToString("Sử dụng");
...    }
...    else if (chuyengiao.TrangThai == "Đang giao")
...    {
...        donhang.TinhTrangGH = Convert.ToString("Đang giao");
...        shipper.TinhTrang = Convert.ToString("Đang giao hàng");
...        phuongtien.TinhTrang = Convert.ToString("Sử dụng");
...    }
...    else if (chuyengiao.TrangThai == "Đã giao")
...    {
...        donhang.TinhTrangGH = Convert.ToString("Đã giao");
...        shipper.TinhTrang = Convert.ToString("Sẵn sàng");
...        phuongtien.TinhTrang = Convert.ToString("Sẵn sàng");
...    }
...    database.SaveChanges();
...    return RedirectToAction("DanhSachCacChuyenGiao");
...}

```

### Action Method:

- **DonDaGiao:**  
Hiển thị danh sách đơn đã giao
- **DonDangGiao:**  
Hiển thị danh sách đơn đang giao
- **DonChoGiao:**  
Hiển thị danh sách đơn chờ giao

- ChinhSuaChuyenGiao: Chính sửa chuyến giao đồng thời cập nhật trạng thái nhân viên giao hàng, phương tiện giao hàng và tình trạng đơn hàng có mã chuyến giao tương ứng

```

.... //Them-chuyen-giao
.... public ActionResult ThemChuyenGiaoHang()
.... {
....     ViewBag.PhuongTien = new SelectList(database.PhuongTienGHs.Where(s => s.TinhTrang == "Sẵn sàng"), "MaPT", "MaPT");
....     ViewBag.NVGiao = new SelectList(database.NhanVienGHs, "MaNV", "TenNV");
....     TempData["messageAlert"] = "Đã thêm chuyến giao mới!";
....     return View();
.... }

.... [HttpPost]
.... public ActionResult ThemChuyenGiaoHang([Bind(Include = "MaNVLap, MaNVGiao, MaPT, NgayLap, TrangThai, NgayGiao")]
.... ChuyenGiao chuyenggiao, NhanVienGH nhanviengiao, PhieuNhapXuat phieunhapxuat, PhuongTienGH phuongtien)
.... {
....     if (ModelState.IsValid)
....     {
....         ViewBag.ngaylap = TempData["ngaylapcg"] = System.DateTime.Now;
....         Random rd = new Random();
....         var cg = "GH" + rd.Next(1, 1000);
....         chuyenggiao.TienThu = 0;
....         chuyenggiao.MaGH = cg;
....         chuyenggiao.TrangThai = Convert.ToString("Chờ giao");
....         chuyenggiao.NgayLap = System.DateTime.Now;
....         phuongtien = database.PhuongTienGHs.Where(s => s.MaPT == chuyenggiao.MaPT).FirstOrDefault();
....         phuongtien.TinhTrang = Convert.ToString("Sử dụng");
....         database.ChuyenGaos.Add(chuyenggiao);
....         database.SaveChanges();
....     }
....     return RedirectToAction("DanhSachCacChuyenGiao");
.... }

```

Action Method ThemChuyenGiao: Thêm mới chuyến giao

```

.... //Xoa-chuyen-giao
.... public ActionResult XoaChuyenGiao(String id)
.... {
....     return View(database.ChuyenGaos.Where(s => s.MaGH == id).FirstOrDefault());
.... }
.... [HttpPost]
.... public ActionResult XoaChuyenGiao(String id, ChuyenGiao chuyengiao, NhanVienGH shipper, PhuongTienGH phuongtien)
.... {
....     chuyengiao = database.ChuyenGaos.Where(s => s.MaGH == id).FirstOrDefault();
....     database.ChuyenGaos.Remove(chuyengiao);
....     database.Entry(shipper).State = System.Data.Entity.EntityState.Modified;
....     database.Entry(phuongtien).State = System.Data.Entity.EntityState.Modified;
....     shipper.TinhTrang = Convert.ToString("Sẵn sàng");
....     phuongtien.TinhTrang = Convert.ToString("Sẵn sàng");
....     database.SaveChanges();
....     TempData["AlertMessage"] = "Xóa thành công";
....     return RedirectToAction("DanhSachCacChuyenGiao");
.... }

```

Action Method XoaChuyenGiao: Xóa chuyến giao

```

    ... //Chi-tiet-chuyen-giao
    ... public ActionResult ChiTietChuyenGiaoHang(string id)
    ... {
    ...     TempData["magh01"] = id;
    ...     return View(database.DonHangs.ToList().Where(s => s.MaGH == id));
    ... }

```

Action Method ChiTietChuyenGiaoHang: hiển thị chi tiết chuyến giao hàng tương ứng với mã chuyến giao được chọn

```

    ... //Cap-nhat-trang-thai-don-hang, cap-nhat-tong-tien-thu-khi-tinh-trang-don-hang-la "Chua-thanh-toan"
    ... public ActionResult CapNhatTrangThaiDonHang(string id)
    ... {
    ...     TempData["madh01"] = id;
    ...     ViewBag.ChuyenGiao = new SelectList(database.ChuyenGaos.Where(s => s.TrangThai == "Chờ giao"), "MaGH", "MaGH");
    ...     //ViewBag.ChuyenGiao = new SelectList(database.ChuyenGaos, "MaGH", "MaGH");
    ...     return View(database.DonHangs.Where(s => s.MaDH == id).FirstOrDefault());
    ... }

    [HttpPost, ActionName("CapNhatTrangThaiDonHang")]
    [ValidateAntiForgeryToken]
    public ActionResult CapNhatTrangThaiDonHang(DonHang donhang, ChuyenGiao chuyengiao)
    {
        ...
        if (ModelState.IsValid)
        {
            ...
            chuyengiao = database.ChuyenGaos.Where(s => s.MaGH == donhang.MaGH).FirstOrDefault();
            if (donhang.TinhTrangThanhToan == "Chưa thanh toán")
            {
                ...
                var tien = donhang.TongTien;
                chuyengiao.TienThu += tien;
                ...
            }
            else-if (donhang.TinhTrangThanhToan == "Đã thanh toán")
            {
                ...
                chuyengiao.TienThu = chuyengiao.TienThu;
            }
            database.Entry(donhang).State = System.Data.Entity.EntityState.Modified;
            database.Entry(chuyengiao).State = System.Data.Entity.EntityState.Modified;
            database.SaveChanges();
            TempData["messageAlert"] = "Xếp đơn thành công";
        }
        ...
        return RedirectToAction("DanhSachCacChuyenGiao");
    }
}

```

Action Method CapNhatTrangThaiDonHang: cập nhật trạng thái đơn hàng

```

    ... //Danh-sach-shipper
    ... public ActionResult DanhSachShipper()
    ... {
    ...     return View(database.NhanVienGhs.ToList().OrderByDescending(s => s.MaNV));
    ... }

```

Action Method DanhSachShipper: hiển thị danh sách nhân viên giao hàng

```

    public ActionResult ThemShipper()
    {
        .....
        return View();
    }

    //Them Shipper
    [HttpPost]
    public ActionResult ThemShipper(NhanVienGH nhanVien)
    {
        Random rd = new Random();
        var manv = "SHIP" + rd.Next(1, 1000);
        nhanVien.MaNV = manv;
        nhanVien.TinhTrang = Convert.ToString("Sẵn sàng");
        database.NhanVienGHs.Add(nhanVien);
        database.SaveChanges();
        TempData["AlertMessage"] = "Thêm nhân viên giao hàng thành công";
        return RedirectToAction("DanhSachShipper");
    }
}

```

#### Action Method:

- ThemShipper: Hiển thị giao diện thêm nhân viên giao hàng
- [HttpPost] ActionResult ThemShipper: thực hiện truy xuất cơ sở dữ liệu để thêm thông tin nhân viên giao hàng mới, đồng thời cập nhật tình trạng nhân viên giao hàng mới là “Sẵn sàng”

```

    //chinh sua shipper
    public ActionResult ChinhSuaShipper(String id)
    {
        .....
        return View(database.NhanVienGHs.Where(s => s.MaNV == id).FirstOrDefault());
    }

    [HttpPost]
    public ActionResult ChinhSuaShipper(String id, NhanVienGH nhanVien)
    {
        .....
        database.Entry(nhanVien).State = System.Data.Entity.EntityState.Modified;
        database.SaveChanges();
        TempData["AlertMessage"] = "Chỉnh sửa thành công";
        return RedirectToAction("DanhSachShipper");
    }
}

```

#### Action Method:

- ChinhSuaShipper: Hiển thị giao diện chỉnh sửa nhân viên giao hàng theo mã nhân viên được chọn
- [HttpPost] ActionResult ChinhSuaShipper: thực hiện truy xuất cơ sở dữ liệu để chỉnh sửa thông tin nhân viên giao hàng tương ứng với mã nhân viên được chọn

```

... //xep xe cho shipper
public ActionResult XepXe(String id)
{
    ... ViewBag.ChuyenGiao = new SelectList(database.ChuyenGiaos.Where(s => s.TrangThai == "Chờ giao"), "MaGH", "MaGH");
    ... return View(database.NhanVienGHs.Where(s => s.MaNV == id).FirstOrDefault());
}

[HttpPost]
public ActionResult XepXe(String id, NhanVienGH nhanVien)
{
    ... database.Entry(nhanVien).State = System.Data.Entity.EntityState.Modified;
    ... nhanVien.TinhTrang = Convert.ToString("Đang giao hàng");
    ... database.SaveChanges();
    ... TempData["AlertMessage"] = "Xếp xe thành công";
    ... return RedirectToAction("DanhSachShipper");
}

```

### Action Method:

- XepXe: Hiển thị giao diện các thông tin chuyến giao đang có và lựa chọn chuyến giao cho nhân viên giao hàng

```

... //xoá shipper
public ActionResult XoaShipper(String id)
{
    ...
    ... return View(database.NhanVienGHs.Where(s => s.MaNV == id).FirstOrDefault());
}

[HttpPost]
public ActionResult XoaShipper(String id, NhanVienGH nhanvien)
{
    ... nhanvien = database.NhanVienGHs.Where(s => s.MaNV == id).FirstOrDefault();
    ... database.NhanVienGHs.Remove(nhanvien);
    ... database.SaveChanges();
    ... TempData["AlertMessage"] = "Xóa thành công";
    ... return RedirectToAction("DanhSachShipper");
}

```

Action Method: XoaShipper: Hiển thị giao diện xóa nhân viên giao hàng tương ứng với mã nhân viên giao hàng được chọn

```

... //Danh sách phương tiện giao hàng
public ActionResult DanhSachPhuongTienGiaoHang()
{
    ...
    ... return View(database.PhuongTienGHs.ToList().OrderByDescending(s => s.MaPT));
}

```

Action Method DanhSachPhuongTienGiaoHang: Hiển thị giao diện danh sách phương tiện giao hàng đang có.

```

----//Them phuong.tien.giao.hang
----public ActionResult ThemPhuongTienGiaoHang()
----{
----|----return View();
----}

----[HttpPost]
----public ActionResult ThemPhuongTienGiaoHang(PhuongTienGH phuongtien)
----{
----|----Random rd = new Random();
----|----var mapt = "51A-" + rd.Next(1, 1000);
----|----phuongtien.MaPT = mapt;
----|----phuongtien.TinhTrang = Convert.ToString("Sẵn sàng");
----|----database.PhuongTienGHs.Add(phuongtien);
----|----database.SaveChanges();
----|----return RedirectToAction("Danh Sach PhuongTienGiaoHang");
----}

```

Action Method ThemPhuongTienGiaoHang: Hiển thị giao diện thêm phương tiện giao hàng.

```

----//Chinh.sua.phuong.tien.giao.hang
----public ActionResult ChinhSuaPhuongTienGiaoHang(String id)
----{
----|----return View(database.PhuongTienGHs.Where(s => s.MaPT == id).FirstOrDefault());
----}
----[HttpPost]
----public ActionResult ChinhSuaPhuongTienGiaoHang(String id, PhuongTienGH pt)
----{
----|----database.Entry(pt).State = System.Data.Entity.EntityState.Modified;
----|----database.SaveChanges();
----|----return RedirectToAction("Danh Sach PhuongTienGiaoHang");
----}

```

Action Method ChinhSuaPhuongTienGiaoHang: Hiển thị giao diện chỉnh sửa phương tiện giao hàng tương ứng với mã phương tiện giao hàng được chọn.

```

----//Xoa.phuong.tien.giao.hang
----public ActionResult XoaPhuongTienGiaoHang(String id)
----{
----|----return View(database.PhuongTienGHs.Where(s => s.MaPT == id).FirstOrDefault());
----}
----[HttpPost]
----public ActionResult XoaPhuongTienGiaoHang(String id, PhuongTienGH pt)
----{
----|----pt = database.PhuongTienGHs.Where(s => s.MaPT == id).FirstOrDefault();
----|----database.PhuongTienGHs.Remove(pt);
----|----database.SaveChanges();
----|----return RedirectToAction("Danh Sach PhuongTienGiaoHang");
----}

```

Action Method XoaPhuongTienGiaoHang: Hiển thị giao diện xóa phương tiện giao hàng tương ứng với mã phương tiện giao hàng được chọn

```

---//In-phieu-xuat-kho
---public ActionResult InPhieuXuatKho(string id)
---
{
---    if (id == null)
    {
---        return new HttpStatusCodeResult(HttpStatusCode.BadRequest);
    }

---    PhieuNhapXuat phieuNhapXuat = database.PhieuNhapXuats.Find(id);
---    TempData["maPhieuXuattheodonhang"] = id;
---    var maDH = phieuNhapXuat.MaDH;
---    TempData["ctmadh"] = phieuNhapXuat.MaDH;
---    TempData["nvlap"] = phieuNhapXuat.MaNVLap;
---    TempData["tennvlap"] = phieuNhapXuat.NhanVien.TenNV;
---    TempData["ngaylap"] = phieuNhapXuat.NgayLap;
---    TempData["tenNGiao"] = phieuNhapXuat.NguoiGiaoHang;
---    TempData["phuongtien"] = phieuNhapXuat.PhuongTienGiaoHang;
---    foreach (var item in database.DonHangs)
    {
---        if (maDH == item.MaDH)
        {
---            TempData["maDL"] = item.MaDL;
---            TempData["tenDL"] = item.DaiLy.TenDL;
---            TempData["diemgiao"] = item.DiemGiao;
        }
    }
---    if (phieuNhapXuat == null)
    {
---        return HttpNotFound();
    }
---    return View(phieuNhapXuat);
}

```

Action Method InPhieuXuatKho: truy xuất dữ liệu từ cơ sở dữ liệu và lấy dữ liệu phiếu xuất kho có mã đơn hàng tương ứng với mã đơn hàng được chọn, trả về giao diện phiếu xuất kho

```

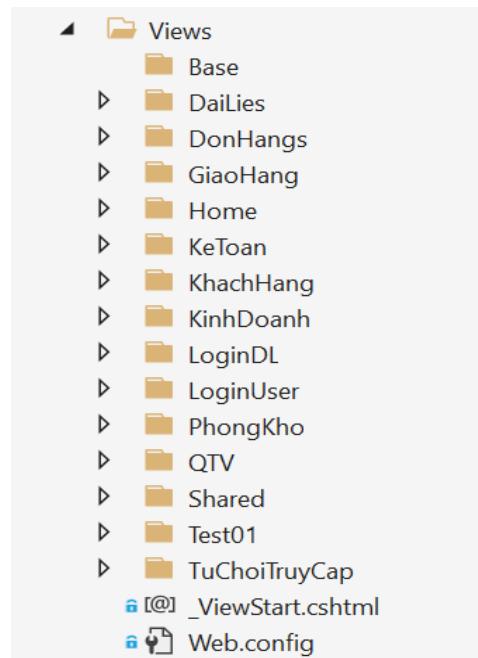
---//In-hoa-don
---public ActionResult InHoaDon(string id)
{
---    DonHang donHang = database.DonHangs.Find(id);
---    ChuyenGiao chuyengiao = database.ChuyenGiaos.Where(s => s.MaGH == id).FirstOrDefault();
---    TempData["chuyengiao"] = chuyengiao;
---    TempData["ctmadh"] = id;
---    if (id == null)
    {
---        return new HttpStatusCodeResult(System.Net.HttpStatusCode.BadRequest);
    }
---    Session["madaily"] = id;
---    if (donHang == null)
    {
---        return HttpNotFound();
    }
---    return View(donHang);
}

```

Action Method:

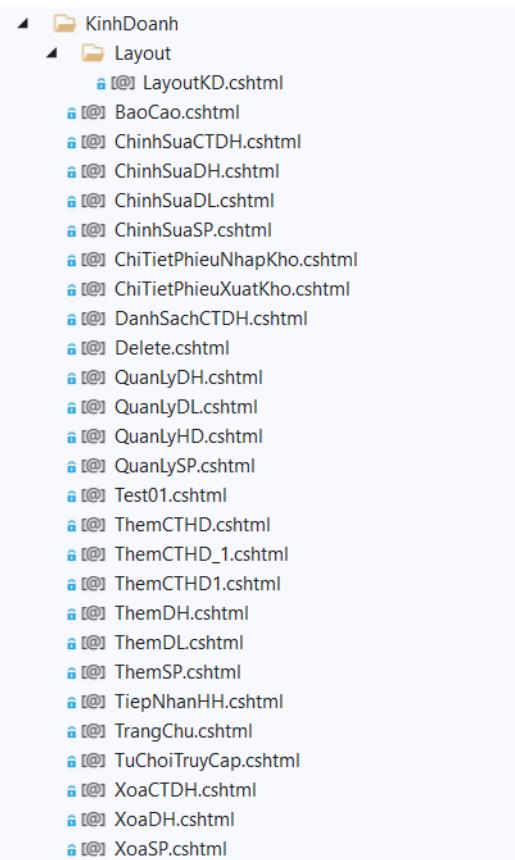
- InHoaDon: truy xuất dữ liệu từ cơ sở dữ liệu và lấy dữ liệu hóa đơn có mã đơn hàng tương ứng với mã đơn hàng được chọn, trả về giao diện hóa đơn của đơn hàng

### 4.3. VIEWS



Hình 118: Module views

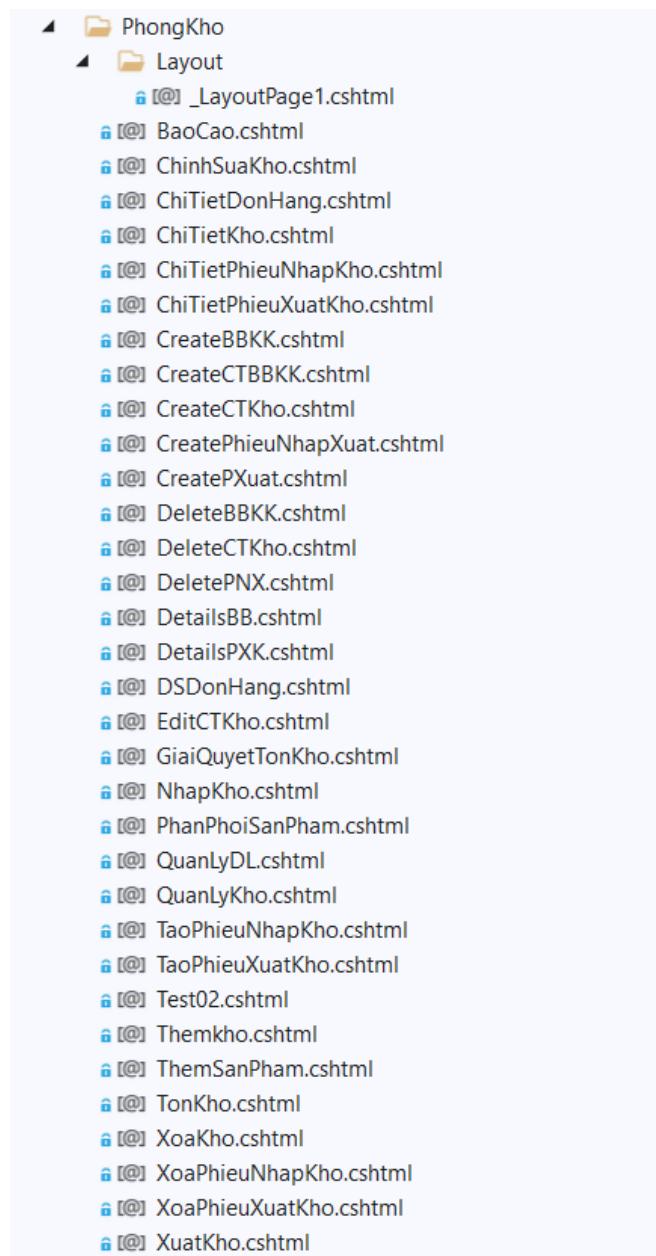
Danh mục view hỗ trợ cho vai trò nghiệp vụ cho từng phòng Kinh Doanh, Kế toán, Kho, Giao Hàng, Đại lý.



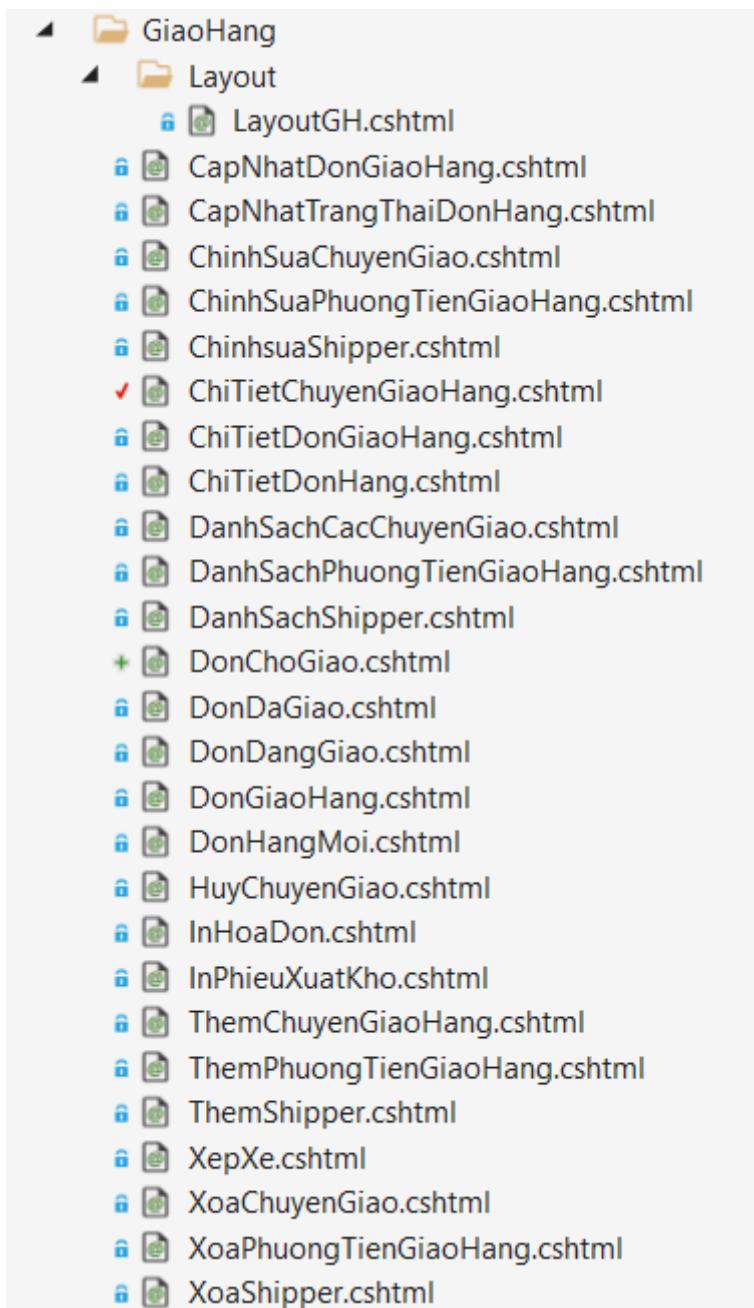
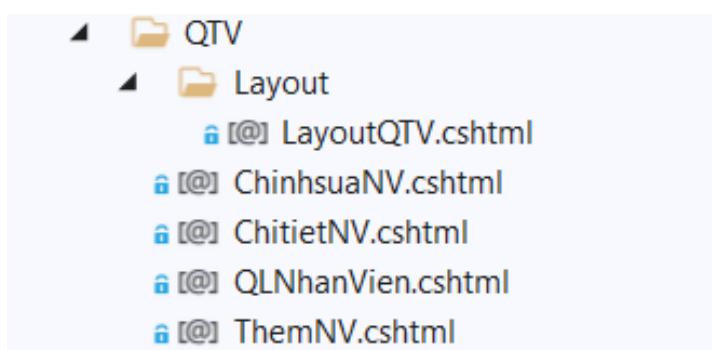
Hình 119. Module views phòng kinh doanh

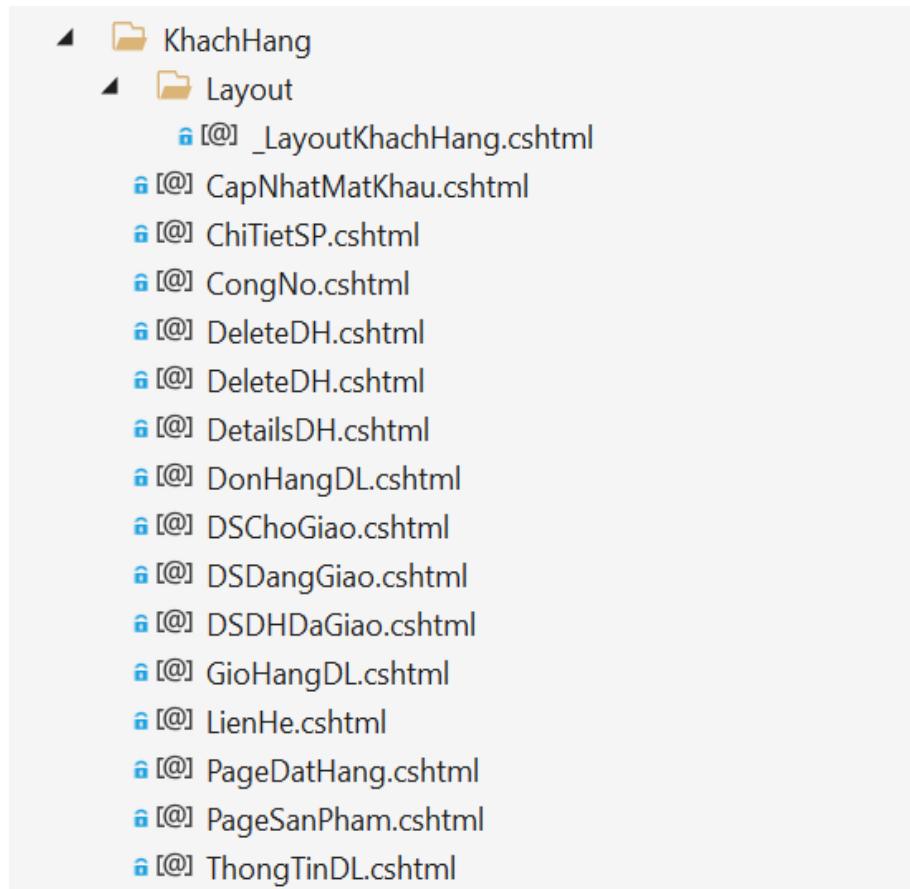


Hình 120. Module views phòng kế toán

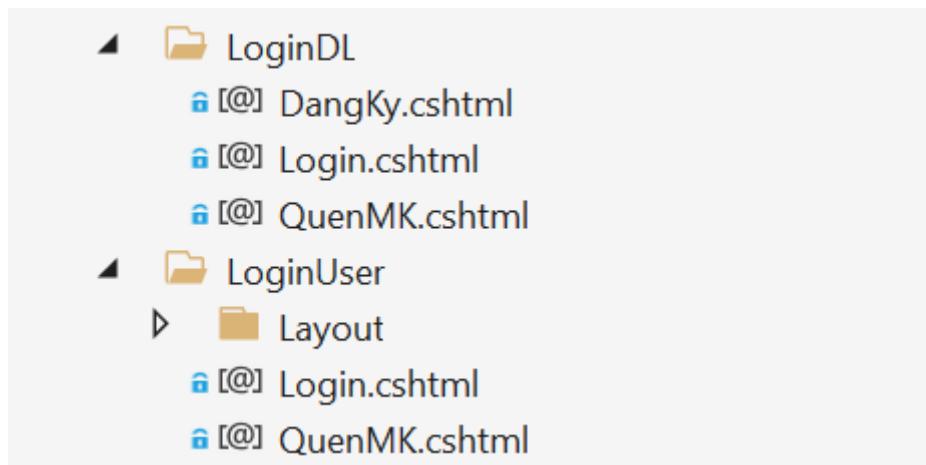


Hình 121. Module views phòng kho hàng

*Hình 122: Module views phòng giao hàng**Hình 123: Module views quản trị viên*



Hình 124. Module views đại lý (khách hàng)



Hình 125: Module views đăng nhập nhân viên và đại lý

## CHƯƠNG 4: TỔNG KẾT

### 5.1. Kết quả đạt được

Tăng hiệu quả quản lý: Phần mềm tạo ra một kênh thông tin giữa các bộ phận trong công ty, giúp quản lý nhân viên, đại lý và sản phẩm một cách nhanh chóng và thuận tiện hơn, từ việc đặt hàng, quản lý kho hàng, phân phối, tài chính, đến báo cáo, theo dõi hoạt động kinh doanh.

Tăng sự chuyên nghiệp trong cung cấp sản phẩm: Với phần mềm quản lý đại lý, công ty có thể cung cấp đến khách hàng những thông tin chính xác, đầy đủ và kịp thời về sản phẩm, tình trạng cập nhật mới nhất từ hàng. Từ đó giúp tăng sự tin tưởng và uy tín của công ty trong mắt các đại lý.

Tăng doanh số bán hàng: Sở hữu phần mềm quản lý đại lý giúp công ty quản lý được mọi hoạt động kinh doanh, quyền lợi về giá cả.

Tiết kiệm chi phí và thời gian: Website đặt hàng của đại lý giúp đặt hàng nhanh chóng, theo dõi tình trạng giao hàng. Từ đó giúp giảm thiểu chi phí chào hàng và thời gian, tăng lợi nhuận phát sinh cho công ty.

Tóm lại, phần mềm quản lý đại lý là một giải pháp hiệu quả để giúp các công ty phân phối trở nên chuyên nghiệp hơn, tiết kiệm chi phí và giảm thiểu thời gian thiểu xót.

### 5.2. Kết quả chưa đạt được

- Giao diện một số chức năng vẫn chưa hoàn thiện, thu hút người dùng
- Một số chức năng trên website vẫn gặp vài lỗi
- Mã hóa dữ liệu nhằm tăng tính bảo mật.
- Một số thông báo vẫn chưa hiển thị hoặc hiển thị không rõ lỗi

### 5.3. Hướng phát triển trong tương lai

- Thiết kế và hoàn thiện giao diện người dùng
- Hoàn thiện các chức năng đang gặp lỗi, hiển thị thêm các thông báo với nội dung chính xác
- Mã hóa dữ liệu nhằm tăng tính bảo mật.
- Phát triển thêm một số chức năng như cho phép đại lý thanh toán online, ...

## TÀI LIỆU THAM KHẢO

- [1] Microsoft, Microsoft SQL documentation  
<https://docs.microsoft.com/vi-vn/sql/?view=sql-server-ver19>
- [2] Microsoft, ASP.NET documentation  
<https://docs.microsoft.com/vi-vn/aspnet/core/?view=aspnetcore-5.0>
- [3] Stack Exchange, stackoverflow.com  
<https://stackoverflow.com/>
- [4] Microsoft Sql Server, Auto-Backup-Sql-Server  
<https://www.sql.edu.vn/microsoft-sql-server/auto-backup-sql-server/>

## BIỂU MẪU ĐÁNH GIÁ LÀM VIỆC NHÓM

Tên thành viên	Phần công việc được giao	Mức độ hoàn thành	Mức độ tích cực	Ngày bắt đầu	Hạn nộp	Nhận xét
Trương Tân Nhã	1. Thực hiện mục phân tích yêu cầu đề bài 2. Thực hiện phân chia nội dung công việc cho các thành viên trong nhóm 3. Thực hiện code trang Actor Nhân viên Kinh doanh 4. Nghiên cứu và áp dụng một số kỹ thuật mới vào đồ án (JS) 5. Tổng hợp nội dung và làm báo cáo về Actor Nhân viên Kinh Doanh 6. Triển khai cơ sở dữ liệu và web upload lên hosting 7. Đánh giá và chất lượng làm việc của các thành viên trong nhóm	95%	Tốt	14/02/2023	26/05/2023	
Nguyễn Thị Diễm Khang	1. Thực hiện mục phân tích yêu cầu đề bài ERD, requirement về vai trò Kho hàng 2. Thực hiện code trang Actor Nhân viên Kho hàng 3. Nghiên cứu và áp dụng một số kỹ thuật mới vào đồ án (JS)	95%	Tốt	14/02/2023	26/05/2023	Có sự nhiệt huyết hết mình và cầu tiến trong quá trình làm việc, chịu chăm học nỗ lực nhiều với đồ án môn học với nhóm

	4. Tổng hợp nội dung và làm báo cáo về Actor Nhân viên kho hàng 5. Triển khai cơ sở dữ liệu và web upload lên hosting 6. Tổng hợp nội dung file word và viết báo cáo				
Nguyễn Thị Diễm Quỳnh	1 Thực hiện mục phân tích yêu cầu đề bài ERD, requirement về vai trò Kế toán 2. Thực hiện code trang Actor Nhân viên Kế toán 3. Nghiên cứu và áp dụng một số kỹ thuật mới vào đồ án (JS) , có áp dụng một số store procedure giúp hệ thống nhanh , giảm lượng code trên asp.net 4. Tổng hợp nội dung và làm báo cáo về Actor Nhân viên Kế toán, Admin 5. Góp ý và xây dựng bài có tính thực tế cao 6. Tổng hợp nội dung file word và viết báo cáo	95%	Tốt	14/02/2023	26/05/2023  Có ý tưởng nhiều và khả năng giải quyết vấn đề nhanh chóng, lanh lợi , nhạy bén trong việc học hỏi ghi chép kỹ lưỡng . Tuy nhiên cần phải cải thiện khả năng làm việc nhóm và lắng nghe nhiều với các thành viên nhiều hơn.
Trương Thị Hường	1 Thực hiện mục phân tích yêu cầu đề bài ERD, requirement về vai trò Đại lý 2. Thực hiện code nghiệp vụ trang Actor Nhân viên Đại lý 3. Nghiên cứu và áp dụng một số kỹ thuật	95%	Tốt	14/02/2023	26/05/2023  Chịu chăm và có cố gắng làm bài, là thành viên đóng vai trò, Support, hậu cần giúp đỡ gánh vác các thành viên trong nhóm. Tuy nhiên cần

	mới vào đồ án (JS) , có áp dụng một số store procedure giúp hệ thống nhanh , giảm lượng code trên asp.net  4. Tổng hợp nội dung và làm báo cáo về Actor Nhân viên Đại lý 5. Đóng góp và xây dựng bài. 6. Tổng hợp nội dung file word và viết báo cáo				phải tự học nhiều hơn và cũng như nghiên cứu sâu nhiều hơn ngoài thực tế.
Nguyễn Thị Kim Ngân	1 Thực hiện mục phân tích yêu cầu đề bài ERD, requirement về vai trò Giao hàng 2. Thực hiện code nghiệp vụ trang Actor Nhân viên Giao hàng 3. Nghiên cứu và áp dụng một số kỹ thuật mới vào đồ án (JS) , có áp dụng một số store procedure giúp hệ thống nhanh , giảm lượng code trên asp.net 4. Tổng hợp nội dung và làm báo cáo về Actor Nhân viên Giao hàng 5. Đóng góp và xây dựng bài. 6. Tổng hợp nội dung file word và viết báo cáo	95%	Khá-tốt	14/02/2023 26/05/2023	Là thành viên đặc biệt cũng khá nổi trội trong việc xử lý nhiều tình huống cũng như gỡ được nhiều debug cho team và đưa ra phương hướng giải quyết tốt nhất. Tuy nhiên cần phải chú tâm cũng như cố gắng nỗ lực nhiều hơn trong việc học để thuận lợi cho đôi bên

HẾT.