



**TRƯỜNG ĐẠI HỌC NGOẠI NGỮ - TIN HỌC THÀNH PHỐ HỒ CHÍ MINH**

**KHOA CÔNG NGHỆ THÔNG TIN**

□□□

**BÁO CÁO**

**CÁC HỆ QUẢN TRỊ CƠ SỞ DỮ LIỆU**

# **THIẾT KẾ VÀ CÀI ĐẶT HỆ THỐNG CƠ SỞ DỮ LIỆU QUẢN LÝ CỬA HÀNG HOA BẰNG ORACLE**

**Giảng viên hướng dẫn: Ths. Nguyễn Thanh Trung**

**Sinh viên thực hiện: Nhóm 3**

- |                         |            |
|-------------------------|------------|
| 1. Phan Ngọc Loan       | 23DH111888 |
| 2. Nguyễn Thanh Thúy Vy | 23DH114141 |
| 3. Nguyễn Đăng Khôi     | 23DH111709 |

*Thành phố Hồ Chí Minh, ngày 4 tháng 11 năm 2025*

## LỜI CẢM ƠN

Trước hết, chúng em xin gửi lời cảm ơn chân thành đến Trường Đại học Ngoại ngữ - Tin học Thành phố Hồ Chí Minh, cùng với Khoa Công nghệ Thông tin đã tạo điều kiện để chúng em được học tập và tiếp cận môn học Các hệ quản trị cơ sở dữ liệu, cũng như triển khai đề tài Phân tích, thiết kế và cài đặt cơ sở dữ liệu cho hệ thống cơ sở dữ liệu quản lý cửa hàng tạp hóa.

Đặc biệt, nhóm em xin bày tỏ lòng biết ơn sâu sắc đến Thầy Nguyễn Thanh Trung, giảng viên hướng dẫn của nhóm. Thầy đã dành nhiều thời gian và tâm huyết để chỉ dẫn, hỗ trợ, cũng như đóng góp những ý kiến quý báu, giúp nhóm em hoàn thiện và nâng cao chất lượng của đề tài.

Nhóm em cũng xin gửi lời cảm ơn đến các bạn sinh viên đã đồng hành, chia sẻ kinh nghiệm và hỗ trợ trong suốt quá trình làm việc nhóm, góp phần giúp sản phẩm ngày càng hoàn thiện hơn.

Cuối cùng, thay mặt cả nhóm, em xin kính chúc Thầy thật nhiều sức khỏe, niềm vui và thành công trong sự nghiệp trồng người.

# MỤC LỤC

LỜI CẢM ƠN.....	II
MỤC LỤC.....	III
DANH MỤC VIẾT TẮT .....	V
DANH MỤC HÌNH ẢNH.....	VII
DANH MỤC BẢNG.....	VIII
LỜI MỞ ĐẦU.....	IX
CHƯƠNG 1. KHẢO SÁT HIỆN TRẠNG & XÁC ĐỊNH YÊU CẦU .....	1
1.1. Khảo sát hiện trạng.....	1
1.1.1. Giới thiệu tổng quan.....	1
1.1.2. Cơ cấu tổ chức hệ thống.....	1
1.1.3. Đánh giá hiện trạng.....	3
1.2. Mô tả nội dung cần giải quyết của đề tài .....	4
1.2.1. Phạm vi, quy trình và nghiệp vụ.....	4
1.2.2. Các chứng từ, báo cáo và số liệu .....	5
1.2.3. Hiệu năng xử lý dữ liệu trong hệ thống.....	5
1.2.4. An toàn dữ liệu .....	5
1.2.5. Xử lý truy xuất đồng thời.....	5
1.2.6. Lý do sử dụng hệ quản trị Oracle.....	5
CHƯƠNG 2. CƠ SỞ LÝ THUYẾT .....	7
2.1. Kiến trúc oracle .....	7
2.1.1. Oracle Instance .....	7
2.1.2. Oracle Database.....	8
2.1.3. Mục đích của kiến trúc này .....	8
2.1.4. Liên hệ với đề tài .....	9
2.2. Quản trị user và profile trong oracle.....	10

2.2.1. Khái niệm User.....	10
2.2.2. Khái niệm Profile.....	10
2.2.3. Liên hệ với đề tài .....	11
<b>2.3. Ngôn ngữ pl/sql.....</b>	<b>12</b>
2.3.1. Khái niệm.....	12
2.3.2. Ví dụ minh họa.....	12
2.3.3. Liên hệ với đề tài .....	13
<b>2.4. Cơ chế sao lưu và phục hồi sau sự cố .....</b>	<b>13</b>
2.4.1. Giới thiệu về backup.....	13
2.4.2. Restore và Recovery .....	14
2.4.3. Redo Log và Archivelog.....	15
2.4.4. Liên hệ với đề tài .....	15
<b>2.5. Quản lý giao tác/giao dịch .....</b>	<b>15</b>
2.5.1. Khái niệm Transaction .....	15
2.5.2. Commit / Rollback.....	16
2.5.3. Liên hệ với đề tài .....	16
<b>2.6. Xử lý đồng thời.....</b>	<b>16</b>
2.6.1. Giới thiệu.....	16
2.6.2. Isolation Level trong Oracle.....	17
2.6.3. Liên hệ với đề tài .....	17
<b>CHƯƠNG 3. PHÂN TÍCH HỆ THỐNG .....</b>	<b>19</b>
<b>3.1. Phân tích chức năng.....</b>	<b>19</b>
3.1.1. Mô hình phân cấp chức năng của hệ thống (sơ đồ BFD) .....	19
3.1.2. Mô tả chi tiết chức năng.....	19
<b>3.2. Phân tích dữ liệu.....</b>	<b>22</b>
3.2.1. Mô hình thực thể kết hợp (ERD) .....	22
3.2.2. Quy định về dữ liệu.....	23
3.2.3. Các sơ đồ luồng dữ liệu (DFD).....	23
<b>CHƯƠNG 4. THIẾT KẾ VÀ CÀI ĐẶT HỆ THỐNG.....</b>	<b>25</b>

<b>4.1. Mô hình dữ liệu quan hệ (Relational Data Model) .....</b>	<b>25</b>
<b>4.2. Từ điển dữ liệu (data dictionary).....</b>	<b>26</b>
4.2.1. Thực thể CHINHANH: .....	26
4.2.2. Thực thể KHO: .....	26
4.2.3. Thực thể LOAIHOA:.....	26
4.2.4. Thực thể HOA:.....	26
4.2.5. Thực thể KHACHHANG: .....	28
4.2.6. Thực thể CHUCVU: .....	28
4.2.7. Thực thể NHANVIEN: .....	28
4.2.8. Thực thể DONHANG: .....	29
4.2.9. Thực thể CHITIETDONHANG: .....	29
4.2.10. Thực thể DONNHAPHANG: .....	30
4.2.11. Thực thể CHITIETNHAPHANG:.....	30
<b>4.3. cài đặt hệ thống cơ sở dữ liệu bằng oracle.....</b>	<b>30</b>
4.3.1. Quản lý lưu trữ dữ liệu.....	30
4.3.2. Cài đặt Store Procedure, Trigger .....	37
4.3.3. Minh họa vấn đề xử lý đồng thời.....	59
4.3.4. Thiết kế và cài đặt giao tác (Transaction).....	60
4.3.5. Thiết kế và cài đặt mô hình phân quyền của hệ thống .....	62
<b>CHƯƠNG 5. KẾT LUẬN .....</b>	<b>67</b>
<b>5.1. Kết quả đạt được .....</b>	<b>67</b>
<b>5.2. Kết quả chưa đạt được.....</b>	<b>67</b>
<b>5.3. Hướng phát triển và mở rộng trong tương lai .....</b>	<b>68</b>
<b>BẢNG PHÂN CÔNG.....</b>	<b>9</b>

## DANH MỤC VIẾT TẮT

STT	Từ viết tắt	Tên đầy đủ	Nghĩa tiếng việt
1	RB	Ràng buộc	Ràng buộc

2	PK	Primary key	Khóa chính
3	FK	Forgein key	Khóa ngoại
4	PL/SQL	Procedural Language/Structured Query Language	Ngôn ngữ Thủ tục/Ngôn ngữ Truy vấn có Cấu trúc
5	NCC	Nhà Cung Cấp	

## **DANH MỤC HÌNH ẢNH**

Hình 1.1 Sơ đồ cơ cấu tổ chức quản lý cửa hàng hoa.....	3
Hình 3.1 Sơ đồ chức năng kinh doanh hệ thống quản lý cửa hàng hoa.....	19
Hình 3.2 Mô hình thực thể kết hợp của hệ thống quản lý cửa hàng hoa .....	22
Hình 4.1 Mô hình dữ liệu quan hệ (relational data model).....	25

## DANH MỤC BẢNG

Bảng 4.1 – Mô tả thực thể CHINHANH .....	26
Bảng 4.2 – Mô tả thực thể KHO .....	26
Bảng 4.3 – Mô tả thực thể LOAIHOA .....	26
Bảng 4.4 – Mô tả thực thể HOA .....	26
Bảng 4.5 – Mô tả thực thể KHACHHANG .....	28
Bảng 4.6 – Mô tả thực thể CHUCVU .....	28
Bảng 4.7 – Mô tả thực thể NHANVIEN .....	28
Bảng 4.8 – Mô tả thực thể DONHANG .....	29
Bảng 4.9 – Mô tả thực thể CHITIETDONHANG .....	29
Bảng 4.10 – Mô tả thực thể DONNHAPHANG .....	30
Bảng 4.11 – Mô tả thực thể CHITIETNHAPHANG .....	30



# LỜI MỞ ĐẦU

## Về mục tiêu

Việc áp dụng công nghệ vào quản lý cửa hàng hoa đang trở thành xu hướng tất yếu trong bối cảnh cạnh tranh ngày càng gay gắt giữa các doanh nghiệp bán lẻ. Thay vì quản lý thủ công bằng sổ sách hoặc file excel rời rạc, các hệ thống quản lý cửa hàng hoa có thể giúp giải quyết được nhiều khó khăn trong quá trình vận hành: từ quản lý sản phẩm, đơn hàng, khách hàng, đến nhân viên và doanh thu. Điều này không chỉ giúp tiết kiệm thời gian và chi phí mà còn nâng cao hiệu quả kinh doanh, đồng thời cải thiện trải nghiệm khách hàng.

Đề tài “Quản lý cửa hàng hoa bằng Oracle” hướng đến việc tận dụng sức mạnh của hệ quản trị cơ sở dữ liệu Oracle để xây dựng một hệ thống quản lý chặt chẽ, chính xác, an toàn và linh hoạt. Hệ thống này hỗ trợ nhà quản lý cửa hàng theo dõi tình trạng hàng hóa, kiểm soát số lượng tồn kho, quản lý đơn đặt hàng và đưa ra các báo cáo doanh thu kịp thời, từ đó góp phần nâng cao hiệu suất kinh doanh và khả năng cạnh tranh trên thị trường.

## Về ý nghĩa

Trong thời đại Cách mạng Công nghiệp 4.0, việc sử dụng công nghệ thông tin để quản lý hoạt động kinh doanh là một yếu tố rất quan trọng. Đối với cửa hàng hoa, phần mềm quản lý dựa trên cơ sở dữ liệu Oracle sẽ mang lại nhiều lợi ích thiết thực:

- Tiết kiệm thời gian và tối ưu hóa hoạt động: Hệ thống cho phép quản lý tự động các khâu như nhập hàng, bán hàng, xử lý đơn đặt online, xuất hóa đơn, và cập nhật tình trạng tồn kho tức thì, thay thế cách quản lý thủ công vốn dễ sai sót.
- Cải thiện trải nghiệm khách hàng: Người mua hoa có thể nhanh chóng biết được tình trạng sản phẩm (hoa còn hàng hay hết hàng), giá bán, các chương trình khuyến mãi... ngay khi đặt hàng. Điều này tăng sự thuận tiện, tạo niềm tin và sự hài lòng cho khách hàng.
- Hỗ trợ quản lý hiệu quả: Cơ sở dữ liệu Oracle giúp cửa hàng dễ dàng thống kê doanh thu, chi phí, lợi nhuận theo từng giai đoạn. Các báo cáo này hỗ trợ nhà quản lý trong việc đưa ra các chiến lược kinh doanh hợp lý.
- Nâng cao khả năng cạnh tranh: Hệ thống có thể kết hợp cùng các tính năng thông minh như gợi ý sản phẩm, quản lý chương trình khuyến mãi, hoặc phân tích thói quen khách hàng để tạo ra lợi thế cạnh tranh và giữ chân khách hàng lâu dài.

Kết luận, việc xây dựng hệ thống Quản lý cửa hàng hoa bằng Oracle không chỉ giúp giải quyết các vấn đề quản lý hiện tại mà còn mở ra cơ hội nâng cao chất lượng dịch vụ, tăng hiệu quả kinh doanh và khả năng cạnh tranh cho doanh nghiệp. Đây là bước tiến quan trọng trong việc ứng dụng công nghệ thông tin vào lĩnh vực bán lẻ, giúp các cửa hàng hoa vận hành chuyên nghiệp hơn và đáp ứng tốt hơn nhu cầu của khách hàng.

## Về đối tượng sử dụng

- **Nhà quản lý cửa hàng hoa:** Dễ dàng theo dõi hoạt động kinh doanh, quản lý nhân viên, kiểm soát doanh thu – chi phí và đưa ra quyết định kịp thời.
- **Khách hàng mua hoa trực tiếp và trực tuyến:** Có thể tìm kiếm sản phẩm, xem thông tin hoa, tình trạng còn hàng, đặt hàng và thanh toán nhanh chóng.
- **Doanh nghiệp, đối tác cung cấp hoa và phụ kiện:** Kết nối để nhập hàng, theo dõi đơn nhập, quản lý công nợ và hợp đồng.
- **Các nhà đầu tư hoặc chủ sở hữu:** Có thể xem báo cáo tổng quan về tình hình kinh doanh, lợi nhuận, xu hướng bán hàng để đánh giá hiệu quả đầu tư.

# CHƯƠNG 1. KHẢO SÁT HIỆN TRẠNG & XÁC ĐỊNH YÊU CẦU

## CẦU

### 1.1. KHẢO SÁT HIỆN TRẠNG

#### 1.1.1. Giới thiệu tổng quan

Hệ thống quản lý cửa hàng hoa là một giải pháp hiện đại và thiết thực, thể hiện rõ cách công nghệ 4.0 được áp dụng vào lĩnh vực bán lẻ và dịch vụ tiêu dùng. Nghiệp vụ doanh nghiệp trong lĩnh vực quản lý cửa hàng hoa là quá trình tổ chức, điều hành và thực hiện các hoạt động liên quan đến bán hoa và phụ kiện cho khách hàng. Nó bao gồm các quy trình và hoạt động nhằm đáp ứng nhu cầu đa dạng của khách hàng, cung cấp sản phẩm hoa tươi, dịch vụ giao hàng, đồng thời quản lý thông tin liên quan đến đơn hàng, khách hàng và kho hàng.

Một hệ thống quản lý cửa hàng hoa cần xây dựng các chức năng hoạt động như sau:

**Bán hàng:** Đảm bảo việc bán hoa và phụ kiện được minh bạch về giá cả, khuyến mãi, chiết khấu và các dịch vụ đi kèm (gói hoa, thiệp chúc, giao hàng).

**Quản lý đơn đặt hàng:** Đảm bảo việc đặt hoa (trực tiếp tại cửa hàng hoặc trực tuyến) được xử lý chính xác, nhanh chóng và theo dõi được trạng thái đơn hàng.

**Quản lý thông tin khách hàng:** Lưu trữ và quản lý dữ liệu khách hàng (thông tin cá nhân, lịch sử mua hàng, sở thích) để phục vụ chăm sóc khách hàng và xây dựng chương trình khuyến mãi.

**Quản lý sản phẩm và tồn kho:** Theo dõi tình trạng hoa tươi, phụ kiện, cập nhật số lượng tồn kho, nhập hàng và hạn sử dụng để tránh tình trạng thiếu hàng hoặc hư hỏng.

**Quản lý nhân viên và doanh thu:** Hỗ trợ quản lý lịch làm việc, phân công nhiệm vụ cho nhân viên, đồng thời theo dõi doanh thu – chi phí, lập báo cáo định kỳ. Quy trình hoạt động nghiệp vụ trong quản lý cửa hàng hoa có thể được mô tả như sau: khách hàng tiến hành đặt hoa trực tiếp tại cửa hàng hoặc thông qua website/ứng dụng trực tuyến. Hệ thống sẽ ghi nhận đơn đặt hàng, xử lý tùy theo tình trạng tồn kho và yêu cầu của khách hàng. Sau khi xác nhận, hệ thống sẽ hướng dẫn khách hàng thanh toán và lưu thông tin hóa đơn. Tiếp đến, cửa hàng sẽ chuẩn bị sản phẩm (bó hoa, lẵng hoa, giỏ hoa...) và thực hiện giao hàng theo yêu cầu. Đồng thời, hệ thống sẽ lưu lại toàn bộ dữ liệu liên quan để hỗ trợ báo cáo, chăm sóc khách hàng và hoạch định chiến lược kinh doanh.

#### 1.1.2. Cơ cấu tổ chức hệ thống

Hệ thống quản lý cửa hàng hoa có cơ cấu tổ chức chặt chẽ nhằm đảm bảo hoạt động hiệu quả và mang lại dịch vụ tốt nhất cho khách hàng. Đứng đầu cửa hàng là Ban Giám đốc, trực tiếp giám sát và điều hành toàn bộ hoạt động kinh doanh của cửa hàng hoa. Dưới Ban Giám đốc là các bộ phận chức năng bao gồm bộ phận kinh doanh và bộ phận vận hành cửa hàng, mỗi bộ phận được quản lý bởi trưởng bộ phận để đảm bảo công việc diễn ra suôn sẻ và hiệu quả.

### **1.1.2.1. Bộ phận Kinh Doanh**

Giám đốc kinh doanh: Chịu trách nhiệm quản lý toàn bộ hoạt động kinh doanh, xây dựng chiến lược bán hàng, chăm sóc khách hàng, quản lý sản phẩm và dịch vụ đi kèm (gói hoa, thiệp chúc, dịch vụ giao hàng).

Trưởng phòng kinh doanh: Hỗ trợ giám đốc trong việc điều hành, giám sát công việc của nhân viên kinh doanh và nhân viên bán hàng, đảm bảo doanh thu đạt chỉ tiêu.

Chuyên viên kinh doanh: Thực hiện việc tìm kiếm khách hàng tiềm năng, duy trì mối quan hệ với khách hàng hiện tại, xây dựng chương trình khuyến mãi để thu hút khách hàng.

Nhân viên bán hàng: Trực tiếp tư vấn, giới thiệu sản phẩm hoa và phụ kiện, xử lý đơn đặt hàng, lập hóa đơn và hỗ trợ khách hàng trong suốt quá trình mua hàng.

### **1.1.2.2. Bộ phận Vận Hành Cửa Hàng**

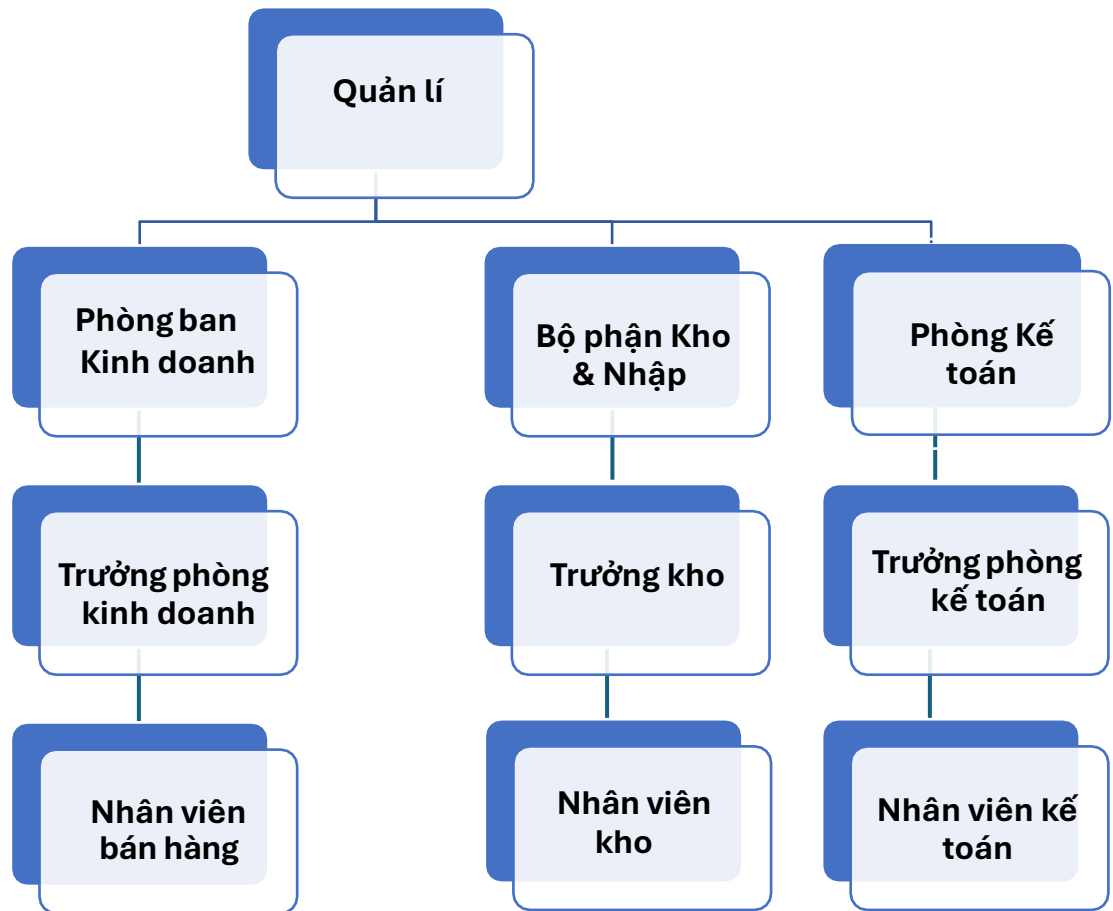
Giám đốc vận hành: Quản lý toàn bộ hoạt động vận hành cửa hàng, bao gồm nhập hàng, kiểm soát tồn kho, chất lượng sản phẩm và quy trình giao hàng.

Trưởng phòng vận hành: Hỗ trợ giám đốc vận hành trong việc giám sát nhân sự, phân công nhiệm vụ, đảm bảo quy trình vận hành được thực hiện đúng tiến độ.

Chuyên viên vận hành: Chịu trách nhiệm cập nhật thông tin về sản phẩm mới, tình trạng hàng tồn kho, phối hợp với bộ phận bán hàng để điều chỉnh số lượng nhập hàng và cung ứng kịp thời theo nhu cầu thực tế.

Nhân viên kho và giao hàng: Đảm bảo việc nhập – xuất hàng đúng quy trình, bảo quản hoa tươi, đồng thời vận chuyển sản phẩm đến khách hàng đúng thời gian và địa điểm yêu cầu.

Cơ cấu tổ chức của hệ thống quản lý cửa hàng hoa được mô tả như sau:



Hình 1.1 Sơ đồ cơ cấu tổ chức quản lý cửa hàng hoa

### 1.1.3. Đánh giá hiện trạng

Hệ thống quản lý cửa hàng hoa truyền thống hiện nay chủ yếu dựa vào phương pháp thủ công như ghi chép sổ sách hoặc sử dụng bảng tính đơn giản. Điều này dẫn đến nhiều bất cập khi quy mô cửa hàng ngày càng mở rộng, số lượng sản phẩm đa dạng và nhu cầu khách hàng ngày càng cao. Vì vậy, việc đánh giá hiện trạng của hệ thống quản lý cửa hàng hoa cần được xem xét dựa trên các khía cạnh như lưu trữ dữ liệu, an toàn dữ liệu và khả năng truy xuất dữ liệu.

Hệ thống quản lý cũ thường gặp phải các vấn đề:

- Trùng lặp dữ liệu: Thông tin khách hàng, đơn hàng, sản phẩm bị ghi trùng hoặc nhập nhiều lần, gây lãng phí bộ nhớ và khó khăn trong việc quản lý.
- Dữ liệu chưa chuẩn hóa: Việc thiếu chuẩn hóa cơ sở dữ liệu dẫn đến tình trạng

mất mát dữ liệu, khó cập nhật đồng bộ, và có thể xuất hiện sai lệch trong báo cáo kinh doanh.

- Khó kiểm soát tồn kho: Việc theo dõi số lượng hoa, phụ kiện, hạn sử dụng và tình trạng nhập – xuất hàng không chính xác, gây ra tình trạng thiếu hàng hoặc dư thừa, làm tăng chi phí.
- Bảo mật kém: Dữ liệu dễ bị thất lạc hoặc rò rỉ do chưa có cơ chế kiểm soát truy cập và phân quyền rõ ràng cho nhân viên.
- Truy vấn chậm và phức tạp: Khi số lượng đơn hàng và khách hàng tăng, việc tìm kiếm và tổng hợp dữ liệu (ví dụ: thống kê doanh thu, danh sách khách hàng thân thiết, số lượng hoa bán chạy) trở nên khó khăn, mất nhiều thời gian và giảm hiệu suất kinh doanh.

Để giải quyết các vấn đề trên, hệ thống quản lý cửa hàng hoa mới cần:

- Chuẩn hóa cơ sở dữ liệu: Thiết kế theo các dạng chuẩn để đảm bảo tính toàn vẹn, tránh dư thừa và dễ dàng mở rộng.
- Tối ưu hóa lưu trữ và truy xuất dữ liệu: Sử dụng Oracle để quản lý tập trung, giúp truy vấn nhanh chóng, chính xác và hỗ trợ báo cáo tức thời.
- Tăng cường bảo mật: Phân quyền rõ ràng cho từng vai trò (quản lý, nhân viên bán hàng, nhân viên kho) để bảo vệ dữ liệu.
- Hỗ trợ phân tích và ra quyết định: Cung cấp các báo cáo doanh thu, phân tích xu hướng bán hàng, đánh giá hiệu quả tồn kho nhằm nâng cao năng lực quản lý và tăng khả năng cạnh tranh.

## **1.2. MÔ TẢ NỘI DUNG CẦN GIẢI QUYẾT CỦA ĐỀ TÀI**

### **1.2.1. Phạm vi, quy trình và nghiệp vụ**

Đề tài sẽ tập trung chủ yếu vào phạm vi quản lý hoạt động kinh doanh của cửa hàng hoa. Hệ thống được xây dựng nhằm đáp ứng các quy trình và nghiệp vụ tối thiểu phải có trong quản lý, bao gồm:

- Quản lý thông tin sản phẩm (hoa, phụ kiện, giá bán, tình trạng tồn kho).
- Quản lý đơn đặt hàng (trực tiếp tại cửa hàng và trực tuyến).
- Quản lý khách hàng và lịch sử mua hàng.
- Quản lý thanh toán và hóa đơn.

Ngoài ra, hệ thống còn bổ sung thêm các quy trình và nghiệp vụ khác để tạo nên sự chặt chẽ, nâng cao hiệu quả quản lý, bao gồm:

- Quản lý nhập hàng, theo dõi tồn kho.
- Quản lý nhân viên, phân quyền người dùng.
- Thống kê doanh thu và lập báo cáo bán hàng theo nhiều tiêu chí.

### **1.2.2. Các chứng từ, báo cáo và số liệu**

Hệ thống sẽ quản lý các chứng từ và số liệu liên quan đến quá trình bán hàng và nhập hàng trong cửa hàng hoa, bao gồm các thông tin như sau:

- Thông tin khách hàng: Họ tên, số điện thoại, địa chỉ, lịch sử mua hàng.
- Thông tin sản phẩm hoa: Loại hoa, giá bán, số lượng tồn kho, ngày nhập hàng, hạn sử dụng.
- Thông tin đơn hàng: Danh sách sản phẩm, số lượng, giá trị đơn hàng, trạng thái giao hàng.
- Doanh thu bán hàng: Thống kê theo ngày, tháng, quý, năm.
- Thông tin hóa đơn và thanh toán: Lưu trữ chi tiết các giao dịch của khách hàng.
- Báo cáo phân tích: Báo cáo sản phẩm bán chạy, khách hàng thân thiết, chi phí nhập hàng và lợi nhuận.

### **1.2.3. Hiệu năng xử lý dữ liệu trong hệ thống**

Hệ thống quản lý cửa hàng hoa cần xử lý liên tục các thao tác nghiệp vụ bán hàng, nhập hàng và quản lý tồn kho mỗi ngày. Do đó, hệ thống phải đảm bảo:

- Cung cấp đầy đủ thông tin về sản phẩm, số lượng tồn kho và cập nhật ngay lập tức khi phát sinh giao dịch bán hàng hoặc nhập hàng.
- Thực hiện tốt các thao tác CRUD (Create – Read – Update – Delete) trong cơ sở dữ liệu, đảm bảo tính chính xác và tốc độ xử lý.
- Hỗ trợ người dùng (quản lý, nhân viên bán hàng, nhân viên kho) thao tác nhanh chóng, dễ dàng và mượt mà trên hệ thống.

Đảm bảo hệ thống có khả năng mở rộng, đáp ứng nhu cầu tăng trưởng dữ liệu khi cửa hàng phát triển quy mô lớn hơn hoặc có nhiều chi nhánh

### **1.2.4. An toàn dữ liệu**

Hệ thống cần đảm bảo tính bảo mật và toàn vẹn dữ liệu, tránh mất mát hoặc sai lệch khi thực hiện giao dịch. Đồng thời, hệ thống cung cấp cơ chế Backup và Restore để khôi phục khi xảy ra sự cố. Quyền truy cập dữ liệu được phân quyền rõ ràng, tuân thủ nguyên tắc “đúng người, đúng việc”, đảm bảo chỉ những nhân sự có thẩm quyền mới được phép thao tác trực tiếp trên dữ liệu.

### **1.2.5. Xử lý truy xuất đồng thời**

Hệ thống phải hỗ trợ nhiều người dùng (quản lý, nhân viên bán hàng, nhân viên kho) thao tác cùng lúc mà không gây ra xung đột dữ liệu. Điều này bảo đảm hoạt động bán hàng và nhập hàng có thể diễn ra song song, không làm chậm hệ thống hoặc ảnh hưởng đến trải nghiệm người dùng

### **1.2.6. Lý do sử dụng hệ quản trị Oracle**

Oracle được lựa chọn làm hệ quản trị cơ sở dữ liệu cho đề tài Quản lý cửa hàng hoa vì những lý do sau:

- Hỗ trợ kiến trúc đa người dùng, phù hợp với môi trường cửa hàng có nhiều

nhân viên thao tác đồng thời.

- Cung cấp các tính năng nâng cao như phân quyền, giới hạn dung lượng, ghi log và cơ chế Locking, giúp đảm bảo an toàn dữ liệu và tránh xung đột trong quá trình giao dịch.
- Ngôn ngữ PL/SQL cho phép tối ưu hóa truy vấn, thuận lợi trong phát triển và chỉnh sửa hệ thống.
- Có cơ chế Backup & Restore mạnh mẽ, hỗ trợ công tác quản lý và bảo trì dữ liệu.
- Được đánh giá cao về bảo mật, hiệu suất và khả năng quản lý dữ liệu, bảo đảm hệ thống quản lý cửa hàng hoạt động ổn định, an toàn và hiệu quả.



## **CHƯƠNG 2. CƠ SỞ LÝ THUYẾT**

### **2.1. KIẾN TRÚC ORACLE**

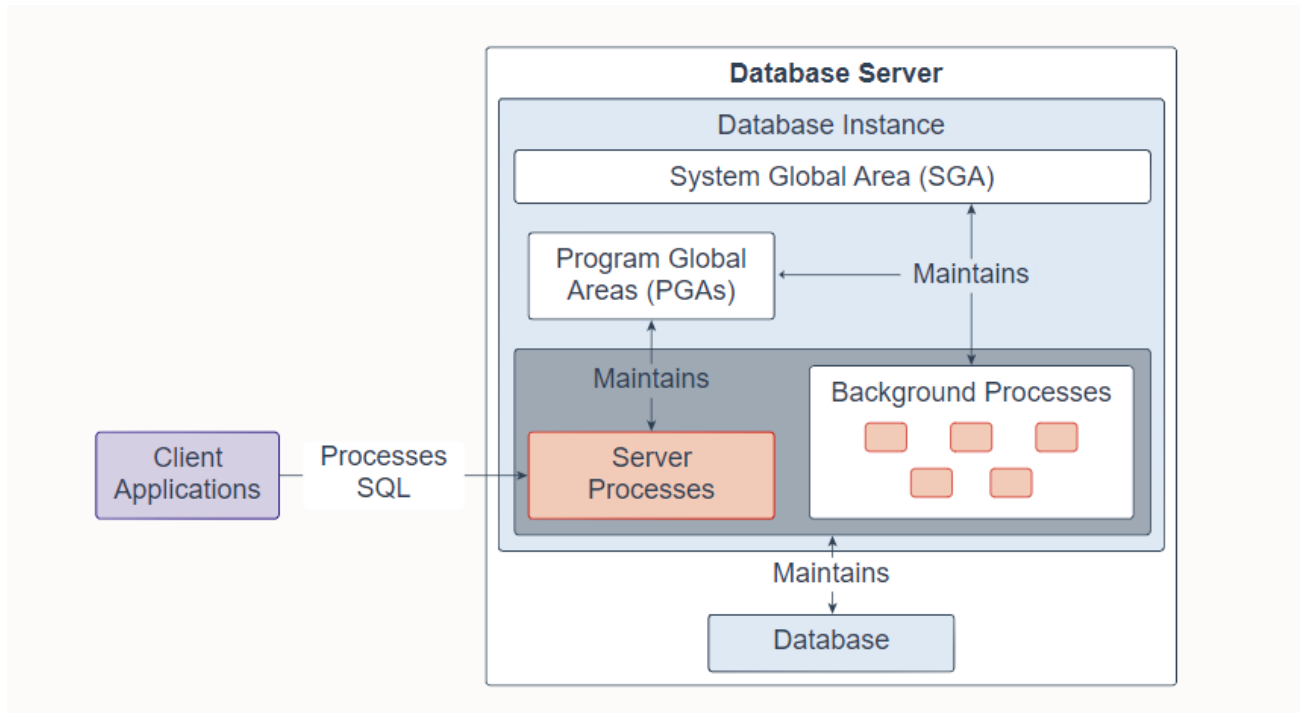
Oracle là hệ quản trị cơ sở dữ liệu quan hệ hướng đối tượng (ORDBMS) có kiến trúc đặc thù, bao gồm hai thành phần cốt lõi: Oracle Instance và Oracle Database. Việc tách rời hai thành phần này giúp Oracle đạt hiệu năng cao, đảm bảo an toàn dữ liệu và hỗ trợ nhiều user hoạt động đồng thời.

#### **2.1.1. Oracle Instance**

Oracle Instance là phần “hoạt động” của hệ thống, bao gồm Memory (SGA: System Global Area) và các Background Processes. Instance chịu trách nhiệm tiếp nhận và xử lý câu lệnh SQL từ người dùng, lưu dữ liệu tạm thời và ghi log các thay đổi trước khi lưu xuống đĩa.

Các thành phần quan trọng của Instance gồm:

- Shared Pool: lưu trữ câu lệnh SQL đã phân tích và các thông tin metadata nhằm giảm thời gian xử lý.
- Database Buffer Cache: vùng nhớ lưu tạm dữ liệu được đọc hoặc chuẩn bị ghi xuống Datafiles.
- Redo Log Buffer: ghi lại các thay đổi để đảm bảo khả năng phục hồi (recovery).
- DBWR (Database Writer Process): chịu trách nhiệm ghi dữ liệu từ Buffer Cache xuống Datafiles.
- LGWR (Log Writer Process): ghi thay đổi từ Redo Buffer vào Redo Log Files.



Hình 2.1 Kiến trúc Oracle Instance & Database

### 2.1.2. Oracle Database

Oracle Database là phần “lưu trữ vật lý” của hệ thống. Đây là nơi chứa tất cả dữ liệu của ứng dụng:

- Datafiles: nơi lưu trữ bảng dữ liệu, chỉ mục (index), thông tin khách hàng, sản phẩm,...
- Control files: quản lý thông tin cấu trúc của database.
- Redo Log Files: lưu lại mọi thay đổi để hỗ trợ phục hồi sau sự cố.

Nhờ cơ chế lưu trữ tách biệt, Oracle có khả năng xử lý khối lượng lớn dữ liệu và vẫn đảm bảo sự toàn vẹn khi nhiều thao tác được thực hiện cùng lúc.

### 2.1.3. Mục đích của kiến trúc này

Kiến trúc kết hợp giữa Instance và Database mang lại nhiều ưu điểm, đặc biệt trong các hệ thống lớn hoặc có nhiều người sử dụng:

- Hiệu năng cao: dữ liệu được lưu tạm trong bộ nhớ (Buffer Cache) giúp việc đọc/ghi diễn ra nhanh hơn nhiều so với truy cập đĩa trực tiếp.

- Đảm bảo an toàn dữ liệu: mọi thay đổi đều được ghi vào Redo Log trước khi lưu thực sự, giúp khôi phục được dữ liệu khi xảy ra sự cố.
- Hỗ trợ nhiều user đồng thời: tiến trình như Latch, Lock, và các isolation level giúp ngăn xung đột dữ liệu khi nhiều nhân viên thao tác cùng lúc.
- Tối ưu cho hệ thống dữ liệu lớn: Oracle được thiết kế để hoạt động tốt trên những hệ thống có khối lượng giao dịch cao, nhiều nghiệp vụ diễn ra song song.

#### **2.1.4. Liên hệ với đề tài**

Trong hệ thống quản lý tạp hóa, dữ liệu phát sinh liên tục: hóa đơn bán hàng, số lượng tồn kho, thông tin khách hàng và điểm tích lũy. Các thao tác này đều được xử lý theo luồng:

- Nhân viên bán hàng thực hiện thao tác INSERT/UPDATE (ví dụ lập hóa đơn).
- Câu lệnh SQL được Instance tiếp nhận và phân tích trong Shared Pool.
- Dữ liệu cập nhật được lưu tạm vào Buffer Cache.
- Thay đổi được ghi vào Redo Log Buffer để đảm bảo sau này có thể phục hồi.
- LGWR ghi log thay đổi xuống Redo Log Files nhằm đảm bảo transaction không bị mất.
- DBWR ghi dữ liệu từ Buffer Cache xuống Datafiles để lưu trữ lâu dài.

Nhờ luồng xử lý này, hệ thống đảm bảo:

- dữ liệu bán hàng không bị mất khi mất điện hoặc lỗi hệ thống,
- thông tin tồn kho luôn được cập nhật nhất quán,
- thao tác của nhiều nhân viên vẫn hoạt động trơn tru mà không gây xung đột dữ liệu,
- tốc độ xử lý hóa đơn nhanh nhờ dữ liệu được lưu tạm trong bộ nhớ.

## **2.2. QUẢN TRỊ USER VÀ PROFILE TRONG ORACLE**

### **2.2.1. Khái niệm User**

Trong Oracle, User là đối tượng đại diện cho một tài khoản đăng nhập vào hệ quản trị cơ sở dữ liệu. Mỗi user được sử dụng để xác định danh tính người dùng và phạm vi quyền hạn mà họ được phép thao tác. Một user có thể sở hữu các đối tượng dữ liệu như bảng (table), view, thủ tục (procedure), trigger, hoặc có thể được cấp quyền truy cập vào các đối tượng thuộc user khác thông qua cơ chế phân quyền (GRANT).

Ví dụ:

User USER\_KINHDOANH có thể được cấp quyền xem dữ liệu bán hàng (SELECT), trong khi User USER\_KHO được cấp quyền thêm và cập nhật số lượng tồn (INSERT, UPDATE).

### **2.2.2. Khái niệm Profile**

Profile trong Oracle là một tập hợp các chính sách (policies) được áp dụng cho user nhằm kiểm soát hành vi sử dụng cơ sở dữ liệu và giới hạn tài nguyên mà mỗi user có thể tiêu thụ. Profile thường được sử dụng để tăng cường bảo mật tài khoản và bảo vệ hệ thống trước các hành vi truy cập bất thường hoặc lạm dụng tài nguyên.

Các tham số phổ biến của một profile bao gồm:

- PASSWORD\_LIFE\_TIME: thời gian hiệu lực của mật khẩu trước khi buộc phải thay đổi.
- FAILED\_LOGIN\_ATTEMPTS: số lần nhập sai mật khẩu tối đa cho phép.
- PASSWORD\_LOCK\_TIME: thời gian user bị khóa khi vượt quá số lần đăng nhập sai.
- INACTIVE\_TIME: thời gian tối đa user được phép không hoạt động trước khi session tự động kết thúc.
- CPU\_PER\_SESSION, LOGICAL\_READS\_PER\_SESSION: giới hạn tài nguyên mà user được phép sử dụng trong một phiên làm việc.

Ví dụ:

Một profile có thể quy định rằng mật khẩu của nhân viên phải thay đổi sau 60 ngày, không được nhập sai quá 3 lần, và session sẽ tự động đóng sau 15 phút không hoạt động. Các quy định này giúp đảm bảo an toàn dữ liệu và hạn chế truy cập trái phép.

### **2.2.3. Liên hệ với đề tài**

Trong hệ thống quản lý tập hóa, mỗi phòng ban hoặc nhóm người dùng có nhiệm vụ riêng và cần giới hạn quyền truy cập khác nhau. Để đảm bảo bảo mật và tránh truy cập sai dữ liệu, hệ thống cần:

1. Tạo user riêng cho từng bộ phận, ví dụ:
  - User\_KinhDoanh: xem báo cáo bán hàng, lập hóa đơn.
  - User\_Kho: cập nhật tồn kho, nhập hàng.
  - User\_KeToan: xem doanh thu, kiểm tra hóa đơn.
2. Gắn profile phù hợp cho từng user, chẳng hạn:
  - Giới hạn thời gian đăng nhập, tránh truy cập ngoài giờ.
  - Quy định số lần nhập sai mật khẩu để tăng bảo mật.
  - Giới hạn tài nguyên (session, CPU) để tránh gây nghẽn hệ thống.
3. Chia quyền hợp lý: mỗi user chỉ được làm đúng chức năng của bộ phận mình, giúp giảm rủi ro thao tác sai hoặc truy cập trái phép dữ liệu quan trọng (doanh thu, tồn kho, thông tin khách hàng).

Nhờ việc quản trị user và profile, hệ thống quản lý tập hóa đảm bảo an toàn dữ liệu, phân quyền rõ ràng và giảm thiểu lỗi do người dùng gây ra.

## **2.3. NGÔN NGỮ PL/SQL**

### **2.3.1. Khái niệm**

PL/SQL (Procedural Language / SQL) là phần mở rộng theo hướng thủ tục của SQL do Oracle phát triển.

Nếu SQL chỉ cho phép thực thi các câu lệnh truy vấn tĩnh, thì PL/SQL bổ sung thêm đầy đủ đặc điểm của một ngôn ngữ lập trình, bao gồm:

- Cấu trúc BLOCK (khối lệnh) gồm khai báo – thực thi – xử lý ngoại lệ.
- Hỗ trợ biến, hằng, cursor, vòng lặp, câu điều kiện.
- Cơ chế EXCEPTION để bắt lỗi và xử lý an toàn.
- Cho phép tạo Procedure, Function, Trigger, Package, giúp tự động hóa xử lý dữ liệu.

### **2.3.2. Ví dụ minh họa**

Trigger rút gọn dùng để tự động cập nhật số lượng tồn kho khi có giao dịch bán hàng:

```
CREATE OR REPLACE TRIGGER trg_update_tonkho
```

```
AFTER INSERT ON ChiTietHD
```

```
FOR EACH ROW
```

```
BEGIN
```

```
    UPDATE SanPham
```

```
    SET SoLuongTon = SoLuongTon - :NEW.SoLuong
```

```
    WHERE MaSP = :NEW.MaSP;
```

```
END;
```

```
/
```

Giải thích: Khi cửa hàng lập hóa đơn và thêm một dòng vào bảng *ChiTietHD*, trigger này tự kích hoạt và trừ tồn kho của sản phẩm tương ứng. Các thao tác này diễn ra hoàn toàn tự động mà không cần nhân viên hay phần mềm phải gọi lệnh bổ sung.

### **2.3.3. Liên hệ với đề tài**

Trong hệ thống quản lý cửa hàng tạp hóa, dữ liệu thay đổi liên tục theo các nghiệp vụ như bán hàng, nhập hàng, áp dụng khuyến mãi hay cộng/trừ điểm cho khách hàng. Những thao tác này nếu xử lý thủ công sẽ:

- dễ xảy ra sai sót,
- mất thời gian,
- thiếu nhất quán dữ liệu,
- khó kiểm soát khi dữ liệu lớn.

Do đó, đề tài đã sử dụng PL/SQL để tự động hóa nhiều chức năng cốt lõi. Cụ thể:

- Trigger cập nhật tồn kho khi phát sinh bán hàng.
- Procedure giảm giá 5% cho hóa đơn trên 500.000đ.
- Procedure tích điểm khách hàng và trừ điểm khi thanh toán.
- Trigger cảnh báo tồn kho thấp để tránh thiếu hàng.

Những thành phần PL/SQL này giúp hệ thống hoạt động chính xác, tự động và nhất quán hơn.

## **2.4. CƠ CHẾ SAO LƯU VÀ PHỤC HỒI SAU SỰ CỐ**

### **2.4.1. Giới thiệu về backup**

Backup (sao lưu dữ liệu) là quá trình tạo bản sao của toàn bộ hoặc một phần cơ sở dữ liệu để lưu trữ và sử dụng khi cần phục hồi. Trong Oracle, cơ chế backup rất đa dạng và linh hoạt. Có ba dạng sao lưu phổ biến:

#### 2.4.1.1. Full Backup (Sao lưu toàn bộ)

Full backup là việc sao lưu toàn bộ dữ liệu trong cơ sở dữ liệu tại một thời điểm cụ thể.

Đặc điểm:

- Là bản sao hoàn chỉnh của hệ thống.
- Phục hồi nhanh vì chỉ cần khôi phục duy nhất bản sao này và áp dụng redo log.
- Thường dùng theo chu kỳ (daily, weekly) tùy mức độ quan trọng của dữ liệu.

#### 2.4.1.2. Incremental Backup (Sao lưu tăng dần)

Incremental backup chỉ sao lưu phần dữ liệu đã thay đổi kể từ lần sao lưu trước đó.

Ưu điểm:

- Giảm kích thước file backup.
- Giảm thời gian sao lưu.
- Phù hợp với hệ thống có nhiều giao dịch phát sinh theo ngày như bán hàng tại tạp hóa.

Nhược điểm: phục hồi sẽ lâu hơn vì phải ghép nhiều bản incremental cùng nhau.

#### 2.4.1.3. Logical Backup

Logical backup (dạng xuất dữ liệu bằng EXP/IMP hoặc Data Pump) thường dùng cho:

- Sao lưu cấu trúc bảng, dữ liệu, views, procedures, triggers.
- Chuyển dữ liệu giữa các database khác nhau.
- Phục hồi lỗi logic, sai sót do người dùng.

### 2.4.2. Restore và Recovery

#### 2.4.2.1. Khái niệm Restore

Restore là quá trình đưa file dữ liệu từ bản backup quay về vị trí ban đầu.

Tại bước này, cơ sở dữ liệu được đưa về trạng thái đúng bằng thời điểm backup.



#### 2.4.2.2. Khái niệm Recovery

Recovery là quá trình áp dụng các redo log (nhật ký giao dịch) lên bản restore nhằm đưa dữ liệu trở về thời điểm gần nhất trước khi xảy ra lỗi.

#### 2.4.3. Redo Log và Archivelog

##### 2.4.3.1. Redo Log

Redo log là nhật ký ghi lại mọi thay đổi về dữ liệu (INSERT, UPDATE, DELETE).

Khi xảy ra sự cố như mất điện hoặc crash hệ thống, Oracle dựa vào redo log để khôi phục dữ liệu chưa kịp ghi xuống đĩa.

##### 2.4.3.2. Chế độ Archivelog

Ở chế độ Archivelog, các redo log cũ được tự động sao lưu lại dưới dạng archive log thay vì bị ghi đè.

Điều này cho phép database:

Khôi phục đến bất kỳ thời điểm nào (Point-in-Time Recovery),

Không bị mất giao dịch ngay cả khi backup full được thực hiện lâu trước đó.

#### 2.4.4. Liên hệ với đề tài

Trong cửa hàng tạp hóa, dữ liệu hóa đơn, tồn kho và khách hàng thay đổi liên tục, nên nếu bị mất sẽ ảnh hưởng trực tiếp đến doanh thu và hoạt động bán hàng. Vì vậy, cần sử dụng full backup, incremental backup, cùng với redo log và archivelog để có thể khôi phục dữ liệu khi xảy ra sự cố như mất điện, lỗi hệ thống hoặc xóa nhầm. Nhờ đó, hệ thống luôn an toàn và dữ liệu được đảm bảo chính xác.

### 2.5. QUẢN LÝ GIAO TÁC/GIAO DỊCH

#### 2.5.1. Khái niệm Transaction

Transaction là một đơn vị xử lý gồm một hoặc nhiều thao tác SQL được thực hiện như một khối thống nhất. Một transaction phải tuân theo bốn tính chất ACID:

- Atomicity: hoặc thực hiện toàn bộ, hoặc không thực hiện gì.
- Consistency: dữ liệu luôn ở trạng thái hợp lệ trước và sau giao tác.
- Isolation: các giao tác không ảnh hưởng lẫn nhau khi chạy đồng thời.
- Durability: khi giao tác hoàn tất (commit), dữ liệu được lưu bền vững.

### **2.5.2. Commit / Rollback**

Commit: lưu vĩnh viễn mọi thay đổi của transaction.

Rollback: hủy toàn bộ thay đổi và đưa dữ liệu về trạng thái trước khi transaction bắt đầu.

### **2.5.3. Liên hệ với đề tài**

Trong hệ thống bán hàng, việc lập một hóa đơn được xem như một transaction. Toàn bộ thao tác thêm hóa đơn, thêm chi tiết, trừ tồn kho... phải hoàn thành trọn vẹn. Nếu có lỗi như sản phẩm hết tồn hoặc nhập sai dữ liệu, hệ thống sẽ rollback để tránh tình trạng hóa đơn dở dang hoặc tồn kho bị trừ sai.

## **2.6. XỬ LÝ ĐỒNG THỜI**

### **2.6.1. Giới thiệu**

Xử lý đồng thời (concurrency control) là cơ chế cho phép nhiều người dùng hoặc nhiều giao tác truy cập và thao tác lên cơ sở dữ liệu cùng một thời điểm. Khi có nhiều thao tác diễn ra đồng thời, hệ thống rất dễ phát sinh xung đột và gây ra sai lệch dữ liệu nếu không được kiểm soát.

Một số vấn đề thường gặp khi nhiều user thao tác cùng lúc:

- Lost update: hai giao tác cùng sửa một bản ghi, bản cập nhật sau ghi đè bản trước.
- Dirty read: đọc phải dữ liệu chưa commit từ giao tác khác.
- Non-repeatable read: đọc cùng một dữ liệu nhưng ra hai kết quả khác nhau do giao tác khác đã thay đổi bản ghi.
- Phantom read: kết quả truy vấn thay đổi vì giao tác khác chèn thêm bản ghi mới.

Vì vậy, hệ quản trị CSDL cần cơ chế khóa (lock) và mức cô lập (isolation level) để đảm bảo các giao tác song song nhưng vẫn giữ được tính đúng đắn của dữ liệu.

### **2.6.2. Isolation Level trong Oracle**

Oracle cung cấp các mức cách ly (isolation level) để kiểm soát cách giao tác đọc – ghi dữ liệu:

- READ COMMITTED (mặc định): ngăn dirty read, đảm bảo chỉ đọc dữ liệu đã commit.
- SERIALIZABLE: hạn chế non-repeatable read và phantom read, đảm bảo giao tác hoạt động như chạy tuần tự.
- READ ONLY: chỉ cho phép đọc, không cho phép cập nhật trong giao tác.

Các mức này giúp hệ thống lựa chọn cách “cô lập” giao tác phù hợp với mức độ an toàn dữ liệu mong muốn.

### **2.6.3. Liên hệ với đề tài**

Trong cửa hàng tạp hóa, nhiều nhân viên có thể lập hóa đơn và bán cùng một sản phẩm tại cùng thời điểm. Nếu không kiểm soát đồng thời, số lượng tồn kho có nguy cơ bị cập nhật sai như bị trừ hai lần, hoặc bán ghi sau ghi đè bản ghi trước. Để giải quyết vấn đề này, hệ thống áp dụng các cơ chế sau:

#### **2.6.3.1. Sử dụng Transaction Control**

Mỗi lần tạo hóa đơn và thêm chi tiết hóa đơn được xem như một transaction.

- Nếu toàn bộ thao tác thêm hóa đơn, trừ tồn kho đều thành công → COMMIT.
- Nếu có lỗi (ví dụ bán vượt tồn kho) → ROLLBACK toàn bộ để tránh tồn kho bị trừ sai.

Điều này đảm bảo tính “hoặc tất cả, hoặc không gì” (Atomicity).

#### **2.6.3.2. Khóa bản ghi (Row-level Locking)**

Khi một giao tác đang cập nhật số lượng tồn của sản phẩm, Oracle tự động khóa dòng dữ liệu đó.

→ Giao tác khác phải đợi cho đến khi khóa được giải phóng, tránh tình trạng hai nhân viên cập nhật cùng lúc lên cùng một bản ghi.

### 2.6.3.3. Isolation Level của Oracle

Hệ thống chạy ở mức cách ly READ COMMITTED, đảm bảo:

- không đọc dữ liệu chưa commit,
- không cập nhật dựa trên dữ liệu tạm,  
→ giúp giao tác không bị ảnh hưởng bởi thay đổi chưa hoàn tất của người dùng khác.

### 2.6.3.4. Trigger kiểm tra tồn kho

Trigger đảm bảo rằng trước khi trừ tồn kho, hệ thống kiểm tra số lượng hiện có:

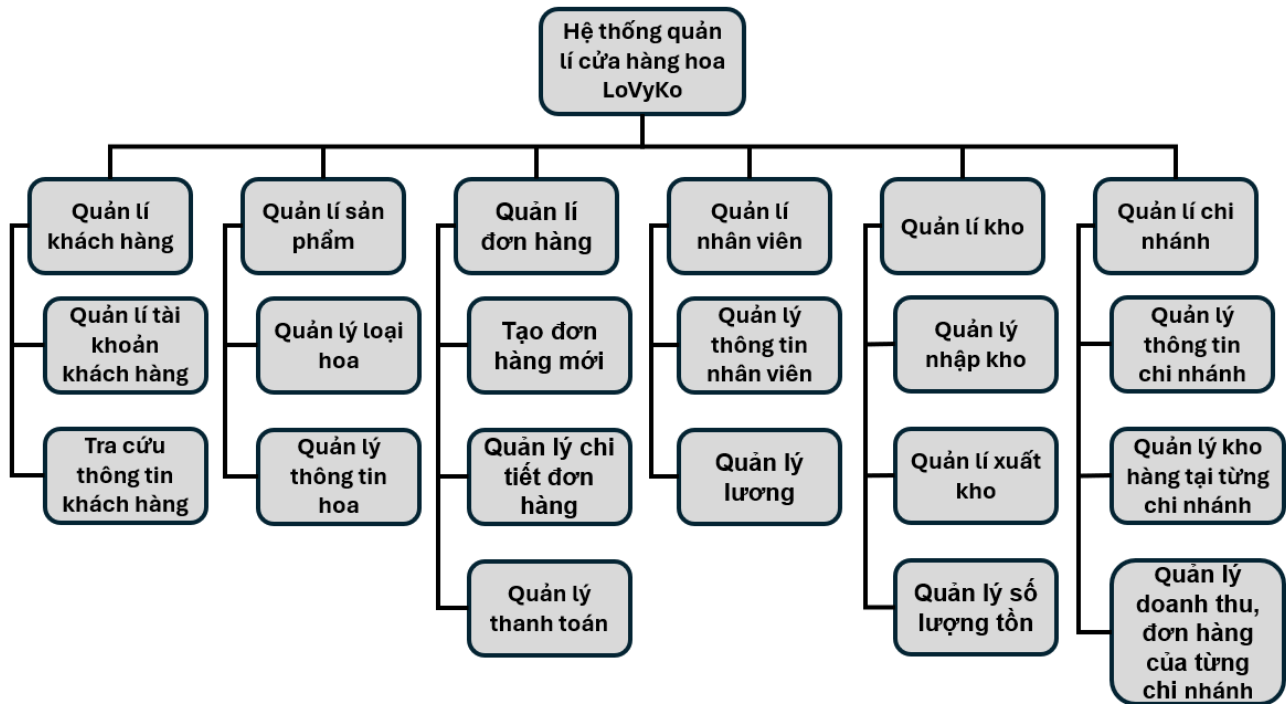
- nếu đủ hàng → cho phép bán
- nếu không đủ → báo lỗi và rollback.

Điều này ngăn hoàn toàn việc bán vượt tồn kho khi nhiều nhân viên thao tác cùng lúc.

## CHƯƠNG 3. PHÂN TÍCH HỆ THỐNG

### 3.1. PHÂN TÍCH CHỨC NĂNG

#### 3.1.1. Mô hình phân cấp chức năng của hệ thống (sơ đồ BFD)



Hình 3.1 Sơ đồ chức năng kinh doanh hệ thống quản lý cửa hàng hoa

#### 3.1.2. Mô tả chi tiết chức năng

##### 3.1.2.1. Quản lý sản phẩm

Chức năng này quản lý toàn bộ thông tin về hoa và sản phẩm kinh doanh, bao gồm danh mục hoa, giá bán, tình trạng tồn kho, hình ảnh.

Quản lý loại hoa

- Cho phép tạo, sửa, xóa các danh mục loại hoa (hoa hồng, hoa lan, hoa tulip, hoa cưới, hoa sự kiện...).
- Mỗi loại hoa có mã loại, tên loại, mô tả.
- Gắn loại hoa với từng sản phẩm cụ thể để dễ tra cứu và lọc dữ liệu.
- Hỗ trợ báo cáo thống kê doanh thu theo loại hoa.

Quản lý thông tin hoa

- Lưu trữ đầy đủ thông tin: mã hoa, tên hoa, giá bán, đơn vị tính (bông, bó, giỏ), hình ảnh, mô tả chi tiết.
- Cập nhật tình trạng sản phẩm: còn hàng, sắp hết, tạm ngừng kinh doanh.
- Quản lý khuyến mãi (giảm giá %, combo sản phẩm).
- Cho phép nhập giá nhập để tính lợi nhuận biên.
- Liên kết trực tiếp với chức năng quản lý kho → số lượng tồn sẽ tự động giảm khi có đơn hàng.

### **3.1.2.2. Quản lý nhân viên**

Quản lý thông tin nhân viên

- Lưu mã NV, họ tên, ngày sinh, giới tính, số điện thoại, địa chỉ, chức vụ.
- Quản lý hợp đồng, ngày vào làm, thời gian thử việc.
- Quản lý lịch làm việc theo ca.

Quản lý lương

- Hệ thống tính lương dựa trên: hệ số lương, số ca làm, số ngày công, doanh số bán hàng (nếu có hoa hồng).
- Ghi nhận các khoản thưởng, phạt, phụ cấp.
- Xuất báo cáo lương hàng tháng.

Phân quyền tài khoản

- Mỗi nhân viên có một tài khoản đăng nhập hệ thống.
- Phân quyền theo vai trò:
  - Quản trị viên (Admin): toàn quyền hệ thống.
  - Quản lý chi nhánh: quản lý nhân viên, kho, đơn hàng của chi nhánh.
  - Nhân viên bán hàng: tạo đơn hàng, quản lý khách hàng.
  - Nhân viên kho: quản lý nhập, xuất, tồn kho.
- Đảm bảo nguyên tắc bảo mật dữ liệu (ai có quyền gì chỉ được xem và chỉnh sửa trong phạm vi đó).

### **3.1.2.3. Quản lý tồn kho**

Đây là chức năng quan trọng để đảm bảo số lượng hoa luôn chính xác, tránh thất thoát.

Quản lý nhập kho

- Ghi nhận phiếu nhập kho: ngày nhập, nhà cung cấp, sản phẩm, số lượng, giá nhập, nhân viên phụ trách.
- Cập nhật số lượng tồn kho theo sản phẩm.
- Kiểm tra trùng lặp và cảnh báo khi nhập sản phẩm đã có tồn kho lớn.

Quản lý xuất kho

- Xuất kho khi có đơn hàng bán ra hoặc điều chuyển chi nhánh.
- Ghi nhận phiếu xuất kho: ngày xuất, lý do xuất, số lượng, nhân viên thực hiện.
- Giảm số lượng tồn kho tương ứng.

Quản lý số lượng tồn

- Hệ thống luôn cập nhật số lượng hoa còn lại trong kho theo thời gian thực.

- Cảnh báo sản phẩm sắp hết, tồn kho lâu ngày hoặc hoa đã hỏng.
- Báo cáo tồn kho theo chi nhánh, theo loại hoa.

#### **3.1.2.4. Quản lý khách hàng**

Quản lý tài khoản khách hàng

- Hệ thống cho phép khách hàng đăng ký tài khoản với các thông tin: họ tên, email, số điện thoại, mật khẩu, địa chỉ.
- Khách hàng có thể đăng nhập, đăng xuất, thay đổi mật khẩu, cập nhật thông tin cá nhân.
- Hệ thống kiểm tra ràng buộc (mật khẩu đủ mạnh, email không trùng lặp).
- Có cơ chế khôi phục mật khẩu qua email/OTP.
- Quản lý trạng thái tài khoản (hoạt động, bị khóa do vi phạm, chưa kích hoạt).

Tra cứu thông tin khách hàng

- Nhân viên có thể tìm kiếm khách hàng bằng tên, số điện thoại, email hoặc mã khách hàng.
- Hiện thị thông tin cơ bản: họ tên, số điện thoại, địa chỉ, số lần mua hàng.
- Liên kết với chức năng đơn hàng → có thể xem lịch sử mua sắm, tần suất đặt hoa, giá trị trung bình mỗi đơn hàng.
- Hỗ trợ phân loại khách hàng (khách hàng thân thiết, khách hàng VIP) dựa trên doanh thu mang lại.
- Cung cấp dữ liệu cho bộ phận chăm sóc khách hàng và marketing.

#### **3.1.2.5. Quản lý đơn hàng**

Tạo đơn hàng mới

- Đơn hàng có thể được tạo bởi nhân viên tại cửa hàng hoặc khách hàng đặt online.
- Thông tin cần nhập: khách hàng, danh sách sản phẩm (hoa), số lượng, giá bán, địa chỉ giao hàng, phương thức thanh toán.
- Hệ thống kiểm tra tồn kho trước khi chấp nhận đơn hàng.
- Mã đơn hàng được tạo tự động, đảm bảo duy nhất.

Quản lý chi tiết đơn hàng

- Lưu danh sách sản phẩm kèm số lượng, đơn giá, thành tiền.
- Cho phép chỉnh sửa đơn hàng (thêm/bớt sản phẩm) trước khi xác nhận.
- Cập nhật trạng thái đơn hàng: chờ xác nhận → đã xác nhận → đang giao → hoàn tất/hủy.
- Lưu vết thay đổi đơn hàng để kiểm soát sai sót.

Quản lý thanh toán

- Hỗ trợ nhiều phương thức: tiền mặt, chuyển khoản, ví điện tử (Momo, ZaloPay...).
- Xác nhận đã thanh toán và tự động phát hành hóa đơn điện tử.
- Lưu thông tin thanh toán: mã giao dịch, số tiền, ngày thanh toán.
- Trường hợp đơn hàng bị hủy → hệ thống quản lý hoàn tiền hoặc ghi chú lý do.

#### **3.1.2.6. Quản lý chi nhánh**

Đây là chức năng mở rộng khi hệ thống có nhiều cửa hàng (multi-branch).

### Quản lý thông tin chi nhánh

- Lưu thông tin chi nhánh: mã, tên, địa chỉ, số điện thoại, quản lý phụ trách.
- Quản lý danh sách nhân viên thuộc từng chi nhánh.

### Quản lý kho hàng tại từng chi nhánh

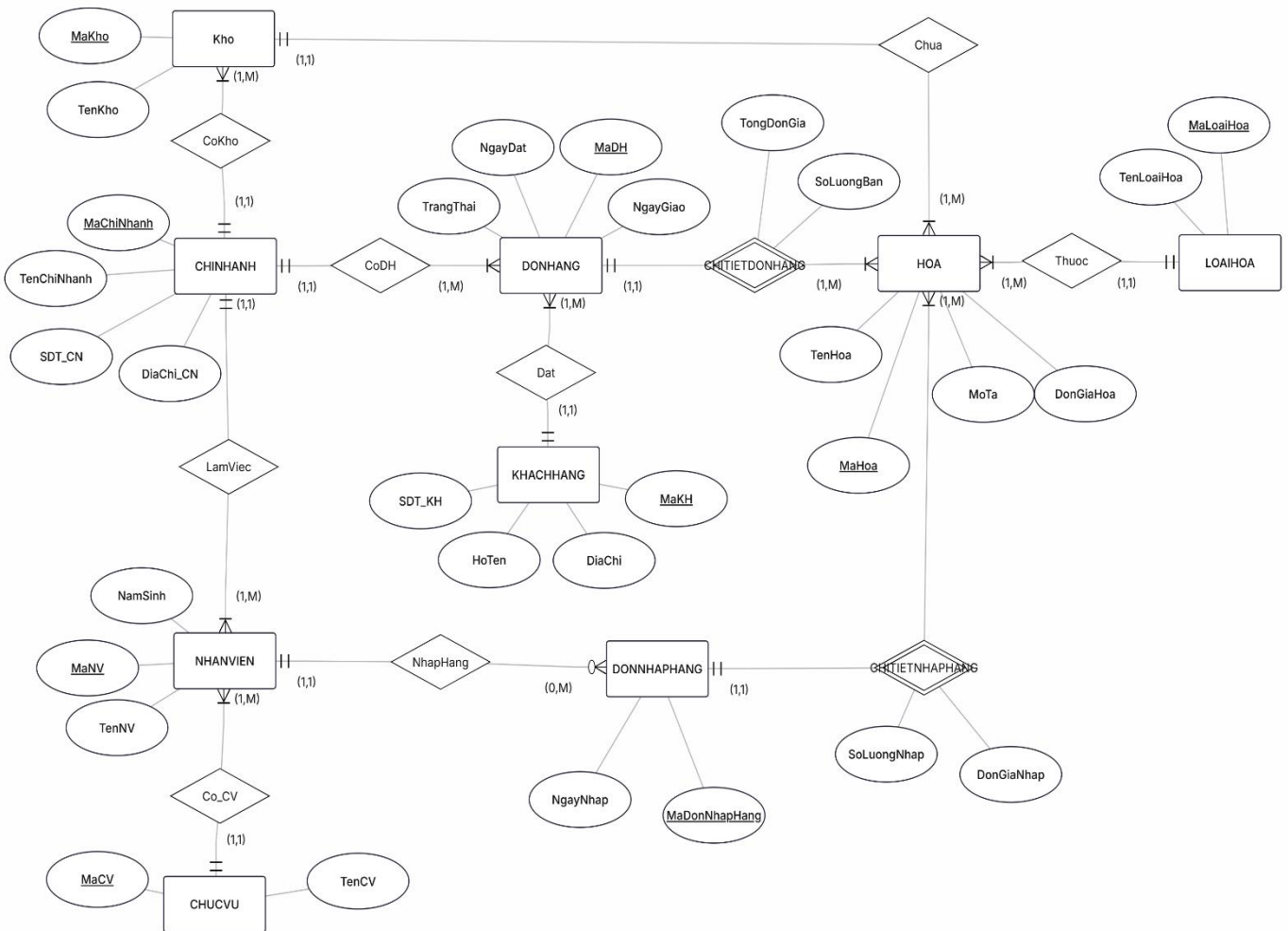
- Mỗi chi nhánh có kho riêng, hệ thống tách biệt số lượng tồn của từng kho.
- Quản lý nhập – xuất kho riêng cho từng chi nhánh.
- Cho phép điều chuyển hàng giữa các chi nhánh.

### Quản lý doanh thu, đơn hàng của từng chi nhánh

- Theo dõi số lượng đơn hàng, doanh thu hàng ngày, hàng tháng.
- Báo cáo sản phẩm bán chạy theo chi nhánh.
- So sánh hiệu quả kinh doanh giữa các chi nhánh.

## 3.2. PHÂN TÍCH DỮ LIỆU

### 3.2.1. Mô hình thực thể kết hợp (ERD)



Hình 2.2 Mô hình thực thể kết hợp của hệ thống quản lý cửa hàng hoa



**3.2.2. Quy định về dữ liệu****3.2.3. Các sơ đồ luồng dữ liệu (DFD)**

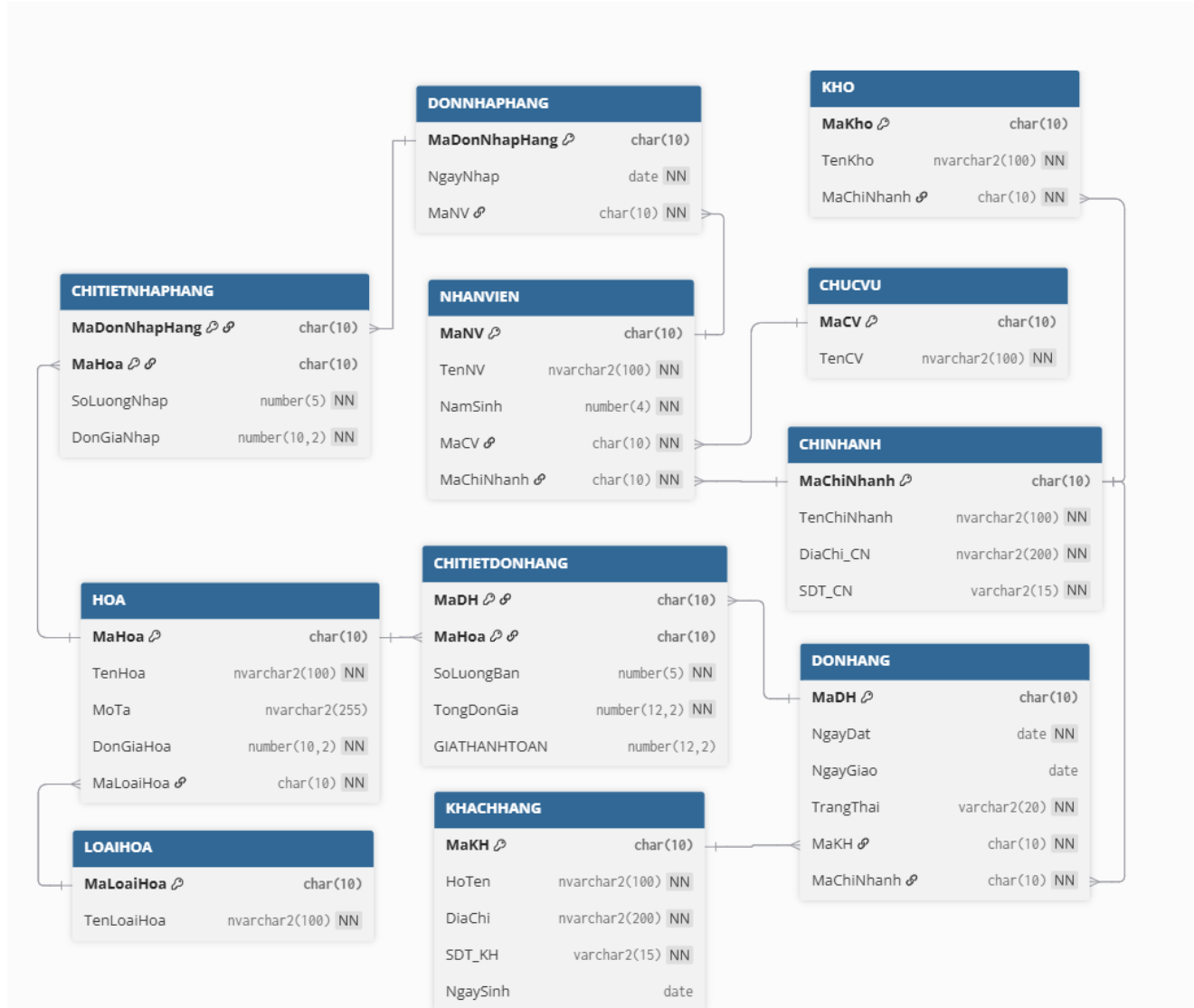
<b>STT</b>	<b>Tên Ràng Buộc Toàn Vẹn</b>	<b>Data Logic</b>	<b>Business Logic</b>
<b>RB01</b>	SĐT chi nhánh và khách hàng bắt đầu bằng 0, có 10 hoặc 11 chữ số	✓	
<b>RB02</b>	Số lượng bán phải $\geq 1$	✓	
<b>RB03</b>	Đơn giá hoa phải $> 0$	✓	
<b>RB04</b>	Đơn giá bán trong đơn hàng phải $> 0$	✓	
<b>RB05</b>	Đơn giá nhập phải $> 0$	✓	
<b>RB06</b>	Ngày giao phải lớn hơn hoặc bằng ngày đặt đơn hàng.	✓	
<b>RB07</b>	Trạng thái đơn hàng chỉ nhận (“CHO XAC NHAN”, “ĐA XAC NHAN”, “DANG GIAO”, “HOAN TAT”, “HUY”)	✓	
<b>RB08</b>	Giảm giá 10% cho khách nếu đặt vào hôm sinh nhật		✓
<b>RB09</b>	Tăng giá 15% cho các mặt hàng hoa khi đặt hàng vào các ngày lễ quan trọng: Các ngày lễ: 14/2, 8/3, 20/10, 1/1		✓
<b>RB10</b>	Một khách hàng không được có quá 5 đơn hàng chưa thanh toán.		✓
<b>RB11</b>	Hoa tồn kho quá 5 ngày kể từ ngày nhập sẽ bị đánh dấu “Xả hàng”.		✓

**CHƯƠNG 3: PHÂN TÍCH HỆ THỐNG**

<b>RB12</b>	Một chi nhánh phải có ít nhất 1 nhân viên bán hàng và 1 nhân viên kho.		✓
<b>RB13</b>	Nhân viên không được làm việc ở 2 chi nhánh cùng lúc.		✓
<b>RB14</b>	Tên hoa không được để trống và không được trùng trong cùng một chi nhánh.	✓	
<b>RB15</b>	Số lượng sản phẩm bán ra trong chi tiết đơn hàng $\leq$ số lượng tồn kho tại thời điểm đó.		✓
<b>RB16</b>	Ngày sinh của khách hàng (nếu có) phải nhỏ hơn ngày hiện tại và khách hàng phải $\geq 15$ tuổi.	✓	✓
<b>RB17</b>	Đơn hàng đã hủy thì không được phép cập nhật thêm chi tiết sản phẩm hay thay đổi thông tin thanh toán.		✓
<b>RB18</b>	Nếu khách hàng hủy đơn hàng trong vòng 2 phút sau khi đặt thì không bị tính phí, sau 2 phút thì tính phí hủy 10%		✓
<b>RB17</b>	Báo cáo doanh thu chi nhánh chỉ tính với đơn hàng “HOAN THANH” + “Đã thanh toán”.		✓
<b>RB18</b>	Khách hàng sẽ được giảm 10% cho đơn hàng đầu tiên		✓
<b>RB19</b>	Khách hàng VIP (Chi $\geq 20$ triệu/năm) sẽ được hưởng ưu đãi 15% tất cả đơn hàng		✓

## CHƯƠNG 4. THIẾT KẾ VÀ CÀI ĐẶT HỆ THỐNG

### 4.1. MÔ HÌNH DỮ LIỆU QUAN HỆ (RELATIONAL DATA MODEL)



Hình 3.1 Mô hình dữ liệu quan hệ (relational data model)

**4.2. TỪ ĐIỂN DỮ LIỆU (DATA DICTIONARY)****4.2.1. Thực thể CHINHANH:**

Bảng 4.1 – Mô tả thực thể CHINHANH

STT	Thuộc tính	Kiểu dữ liệu	Kích thước	Ràng buộc
1	MaChiNhanh	VARCHAR2	10	PRIMARY KEY
2	TenChiNhanh	NVARCHAR2	100	NOT NULL
3	DiaChi_CN	NVARCHAR2	200	NOT NULL
4	SDT_CN	VARCHAR2	15	NOT NULL, CHECK ^0[0-9]{9,10}\$

**4.2.2. Thực thể KHO:**

Bảng 4.2 – Mô tả thực thể KHO

STT	Thuộc tính	Kiểu dữ liệu	Kích thước	Ràng buộc
1	MaKho	VARCHAR2	10	PRIMARY KEY
2	TenKho	NVARCHAR2	100	NOT NULL
3	MaChiNhanh	VARCHAR2	10	FOREIGN KEY → CHINHANH(MaChiNhanh)

**4.2.3. Thực thể LOAIHOA:**

Bảng 4.3 – Mô tả thực thể LOAIHOA

STT	Thuộc tính	Kiểu dữ liệu	Kích thước	Ràng buộc
1	MaLoaiHoa	VARCHAR2	10	PRIMARY KEY
2	TenLoaiHoa	NVARCHAR2	100	UNIQUE, NOT NULL

**4.2.4. Thực thể HOA:**

Bảng 4.4 – Mô tả thực thể HOA

STT	Thuộc tính	Kiểu dữ liệu	Kích thước	Ràng buộc
1	MaHoa	VARCHAR2	10	PRIMARY KEY
2	TenHoa	NVARCHAR2	100	UNIQUE, NOT NULL

---

**CHƯƠNG 3: PHÂN TÍCH HỆ THỐNG**

3	MoTa	NVARCHAR2	255	NULL
4	DonGiaHoa	NUMBER	(10,2)	NOT NULL
5	MaLoaiHoa	VARCHAR2	10	FOREIGN KEY → LOAIHOA(MaLoaiHoa)

#### 4.2.5. Thực thể KHACHHANG:

Bảng 4.5 – Mô tả thực thể KHACHHANG

STT	Thuộc tính	Kiểu dữ liệu	Kích thước	Ràng buộc
1	MaKH	VARCHAR2	10	PRIMARY KEY
2	HoTen	NVARCHAR2	100	NOT NULL
3	DiaChi	NVARCHAR2		NOT NULL
4	SDT_KH	VARCHAR2		NOT NULL, CHECK ^0[0-9]{9}\$
5	NgaySinh	DATE		NOT NULL

#### 4.2.6. Thực thể CHUCVU:

Bảng 4.6 – Mô tả thực thể CHUCVU

STT	Thuộc tính	Kiểu dữ liệu	Kích thước	Ràng buộc
1	MaCV	VARCHAR2	10	PRIMARY KEY
2	TenCV	NVARCHAR2	100	NOT NULL

#### 4.2.7. Thực thể NHANVIEN:

Bảng 4.7 – Mô tả thực thể NHANVIEN

STT	Thuộc tính	Kiểu dữ liệu	Kích thước	Ràng buộc
1	MaNV	VARCHAR2	10	PRIMARY KEY
2	TenNV	NVARCHAR2	100	NOT NULL
3	NamSinh	DATE	—	NOT NULL
4	MaCV	VARCHAR2	10	FOREIGN KEY → CHUCVU(MaCV)

5	MaChiNhanh	VARCHAR2	10	FOREIGN KEY → CHINHANH(MaChiNhanh)
---	------------	----------	----	---------------------------------------

**4.2.8. Thực thể DONHANG:**

Bảng 4.8 – Mô tả thực thể DONHANG

STT	Thuộc tính	Kiểu dữ liệu	Kích thước	Ràng buộc
1	MaDH	VARCHAR2	10	PRIMARY KEY
2	TrangThai	NVARCHAR2	50	NULL
3	NgayDat	DATE	—	NOT NULL
4	NgayGiao	DATE	—	NULL
5	MaKH	VARCHAR2	10	FOREIGN KEY → KHACHHANG(MaKH)
6	MaNV	VARCHAR2	10	FOREIGN KEY → NHANVIEN(MaNV)
7	MaChiNhanh	VARCHAR2	10	FOREIGN KEY → CHINHANH(MaChiNhanh)

**4.2.9. Thực thể CHITIETDONHANG:**

Bảng 4.9 – Mô tả thực thể CHITIETDONHANG

STT	Thuộc tính	Kiểu dữ liệu	Kích thước	Ràng buộc
1	MaDH	VARCHAR2	10	PRIMARY KEY (kết hợp), FOREIGN KEY → DONHANG(MaDH)
2	MaHoa	VARCHAR2	10	PRIMARY KEY (kết hợp), FOREIGN KEY → HOA(MaHoa)
3	SoLuong	NUMBER	—	NULL
4	TongDonGia	NUMBER	(10,2)	NULL

5	GiaThanhToan	NUMBER	(10,2)	NULL
---	--------------	--------	--------	------

**4.2.10. Thực thể DONNHAPHANG:**

Bảng 4.10 – Mô tả thực thể DONNHAPHANG

STT	Thuộc tính	Kiểu dữ liệu	Kích thước	Ràng buộc
2	NgayNhap	DATE	—	NOT NULL
3	MaNV	VARCHAR2	10	FOREIGN KEY → NHANVIEN(MaNV)

**4.2.11. Thực thể CHITIETNHAPHANG:**

Bảng 4.11 – Mô tả thực thể CHITIETNHAPHANG

STT	Thuộc tính	Kiểu dữ liệu	Kích thước	Ràng buộc
1	MaDonNhapHang	VARCHAR2	10	PRIMARY KEY (kết hợp), FOREIGN KEY → DONNHAPHANG(MaDonNhapHang)
2	MaHoa	VARCHAR2	10	PRIMARY KEY (kết hợp), FOREIGN KEY → HOA(MaHoa)
4	SoLuongNhap	NUMBER	—	NULL
5	DonGiaNhap	NUMBER	(10,2)	NOT NULL

**4.3. CÀI ĐẶT HỆ THỐNG CƠ SỞ DỮ LIỆU BẰNG ORACLE****4.3.1. Quản lý lưu trữ dữ liệu****4.3.1.1. Script tạo các table**

-- 1. CHINHANH

CREATE TABLE CHINHANH (

MaChiNhanh CHAR(10) PRIMARY KEY,



---

#### CHƯƠNG 4: THIẾT KẾ VÀ CÀI ĐẶT HỆ THỐNG

```
TenChiNhanh NVARCHAR2(100) NOT NULL,  
DiaChi_CN NVARCHAR2(200) NOT NULL,  
SDT_CN VARCHAR2(15) NOT NULL,  
CONSTRAINT CHK_CN_SDT CHECK (REGEXP_LIKE(SDT_CN, '^0[0-9]{9,10}$'))  
);
```

-- 2. KHO (thuộc chi nhánh)

```
CREATE TABLE KHO (  
    MaKho CHAR(10) PRIMARY KEY,  
    TenKho NVARCHAR2(100) NOT NULL,  
    MaChiNhanh CHAR(10) NOT NULL,  
    CONSTRAINT FK_KHO_CN FOREIGN KEY (MaChiNhanh) REFERENCES CHIN-  
    HANH(MaChiNhanh)  
);
```

-- 3. CHUCVU

```
CREATE TABLE CHUCVU (  
    MaCV CHAR(10) PRIMARY KEY,  
    TenCV NVARCHAR2(100) NOT NULL  
);
```

-- 4. NHANVIEN (làm việc tại chi nhánh)

```
CREATE TABLE NHANVIEN (  
    MaNV CHAR(10) PRIMARY KEY,  
    TenNV NVARCHAR2(100) NOT NULL,  
    NamSinh NUMBER(4) NOT NULL,  
    MaCV CHAR(10) NOT NULL,  
    MaChiNhanh CHAR(10) NOT NULL,
```

---

#### CHƯƠNG 4: THIẾT KẾ VÀ CÀI ĐẶT HỆ THỐNG

```
CONSTRAINT FK_NV_CV FOREIGN KEY (MaCV) REFERENCES CHUCVU(MaCV),  
CONSTRAINT FK_NV_CN FOREIGN KEY (MaChiNhanh) REFERENCES CHIN-  
HANH(MaChiNhanh)  
);
```

-- 5. LOAIHOA

```
CREATE TABLE LOAIHOA (  
    MaLoaiHoa CHAR(10) PRIMARY KEY,  
    TenLoaiHoa NVARCHAR2(100) NOT NULL  
);
```

-- 6. HOA

```
CREATE TABLE HOA (  
    MaHoa CHAR(10) PRIMARY KEY,  
    TenHoa NVARCHAR2(100) NOT NULL,  
    MoTa NVARCHAR2(255),  
    DonGiaHoa NUMBER(10,2) NOT NULL,  
    MaLoaiHoa CHAR(10) NOT NULL,  
    CONSTRAINT FK_HOA_LOAI FOREIGN KEY (MaLoaiHoa) REFERENCES  
    LOAIHOA(MaLoaiHoa),  
    CONSTRAINT CHK_HOA_DONGIAHOA CHECK (DonGiaHoa > 0)  
);
```

-- 7. KHACHHANG

```
CREATE TABLE KHACHHANG (  
    MaKH CHAR(10) PRIMARY KEY,  
    HoTen NVARCHAR2(100) NOT NULL,  
    DiaChi NVARCHAR2(200) NOT NULL,
```

---

#### CHƯƠNG 4: THIẾT KẾ VÀ CÀI ĐẶT HỆ THỐNG

```
SDT_KH VARCHAR2(15) NOT NULL,  
NgaySinh DATE,  
CONSTRAINT CHK_KH_SDT CHECK (REGEXP_LIKE(SDT_KH, '^0[0-9]{9}$'))  
);
```

-- 8. DONHANG (THÊM MaChiNhanh + FK)

```
CREATE TABLE DONHANG (  
    MaDH CHAR(10) PRIMARY KEY,  
    NgayDat DATE DEFAULT SYSDATE NOT NULL,  
    NgayGiao DATE,  
    TrangThai VARCHAR2(20) DEFAULT 'CHO XAC NHAN' NOT NULL,  
    MaKH CHAR(10) NOT NULL,  
    MaChiNhanh CHAR(10) NOT NULL, -- MỐI: Đơn hàng thuộc chi nhánh nào  
    CONSTRAINT FK_DH_KH FOREIGN KEY (MaKH) REFERENCES KHACHHANG(MaKH),  
    CONSTRAINT FK_DH_CN FOREIGN KEY (MaChiNhanh) REFERENCES CHINHANH(MaChiNhanh),  
    CONSTRAINT CHK_DH TRANGTHAI CHECK (TrangThai IN ('CHO XAC NHAN','DA XAC NHAN','DANG GIAO','HOAN THANH','HUY')),  
    CONSTRAINT CHK_DH_NGAYGIAO CHECK (NgayGiao IS NULL OR NgayGiao >= NgayDat)  
);
```

-- 9. CHITIETDONHANG

```
CREATE TABLE CHITIETDONHANG (  
    MaDH CHAR(10),  
    MaHoa CHAR(10),  
    SoLuongBan NUMBER(5) NOT NULL,  
    TongDonGia NUMBER(12,2) NOT NULL,  
    GIATHANHTOAN NUMBER(12,2),
```

---

#### CHƯƠNG 4: THIẾT KẾ VÀ CÀI ĐẶT HỆ THỐNG

```
CONSTRAINT PK_CT_DH PRIMARY KEY (MaDH, MaHoa),  
CONSTRAINT FK_CT_DH_DH FOREIGN KEY (MaDH) REFERENCES DONHANG(MaDH)  
ON DELETE CASCADE,  
CONSTRAINT FK_CT_DH_HOA FOREIGN KEY (MaHoa) REFERENCES HOA(MaHoa),  
CONSTRAINT CHK_CTDH_SOLUONG CHECK (SoLuongBan >= 1),  
CONSTRAINT CHK_CTDH_TONGDG CHECK (TongDonGia >= 0)  
);
```

-- 10. DONNHAPHANG

```
CREATE TABLE DONNHAPHANG (  
    MaDonNhapHang CHAR(10) PRIMARY KEY,  
    NgayNhap DATE DEFAULT SYSDATE NOT NULL,  
    MaNV CHAR(10) NOT NULL,  
    CONSTRAINT FK_DNH_NV FOREIGN KEY (MaNV) REFERENCES NHANVIEN(MaNV)  
);
```

-- 11. CHITIETNHAPHANG

```
CREATE TABLE CHITIETNHAPHANG (  
    MaDonNhapHang CHAR(10),  
    MaHoa CHAR(10),  
    SoLuongNhap NUMBER(5) NOT NULL,  
    DonGiaNhap NUMBER(10,2) NOT NULL,  
    CONSTRAINT PK_CT_NH PRIMARY KEY (MaDonNhapHang, MaHoa),  
    CONSTRAINT FK_CT_NH_DNH FOREIGN KEY (MaDonNhapHang) REFERENCES  
DONNHAPHANG(MaDonNhapHang) ON DELETE CASCADE,  
    CONSTRAINT FK_CT_NH_HOA FOREIGN KEY (MaHoa) REFERENCES HOA(MaHoa),  
    CONSTRAINT CHK_CTNH_SOLUONG CHECK (SoLuongNhap >= 1),  
    CONSTRAINT CHK_CTNH_DONGIANHAP CHECK (DonGiaNhap > 0)
```

);

#### **4.3.1.2. Script insert các dữ liệu**

-- CHINHANH

```
INSERT INTO CHINHANH VALUES ('CN01', N'Chi nhánh Hà Nội', N'123 Trần Phú, Hà Nội',  
'0241234567');
```

```
INSERT INTO CHINHANH VALUES ('CN02', N'Chi nhánh TP.HCM', N'456 Lê Lợi, TP.HCM',  
'0287654321');
```

-- KHO

```
INSERT INTO KHO VALUES ('K01', N'Kho Hà Nội', 'CN01');
```

```
INSERT INTO KHO VALUES ('K02', N'Kho TP.HCM', 'CN02');
```

-- CHUCVU

```
INSERT INTO CHUCVU VALUES ('CV01', N'Quản lý');
```

```
INSERT INTO CHUCVU VALUES ('CV02', N'Nhân viên bán hàng');
```

```
INSERT INTO CHUCVU VALUES ('CV03', N'Nhân viên kho');
```

```
INSERT INTO CHUCVU VALUES ('CV04', N'Nhân viên kế toán');
```

-- NHANVIEN

```
INSERT INTO NHANVIEN VALUES ('NV01', N'Nguyễn Văn An', 1990, 'CV01', 'CN01');
```

```
INSERT INTO NHANVIEN VALUES ('NV02', N'Lê Thị Bình', 1995, 'CV02', 'CN02');
```

-- LOAIHOA

```
INSERT INTO LOAIHOA VALUES ('LH01', N'Hoa Hồng');
```

```
INSERT INTO LOAIHOA VALUES ('LH02', N'Hoa Cúc');
```

-- HOA

```
INSERT INTO HOA VALUES ('H01', N'Hoa Hồng Đỏ', N'Hoa hồng tươi loại 1', 15000, 'LH01');
```

---

#### CHƯƠNG 4: THIẾT KẾ VÀ CÀI ĐẶT HỆ THỐNG

```
INSERT INTO HOA VALUES ('H02', N'Hoa Cúc Trắng', N'Hoa cúc đẹp', 10000, 'LH02');
```

```
-- KHACHHANG
```

```
INSERT INTO KHACHHANG (MaKH, HoTen, DiaChi, SDT_KH, NgaySinh)
```

```
VALUES ('KH01', N'Nguyễn Văn A', N'123 Nguyễn Huệ, Phường Bến Nghé, Quận 1, TP. Hồ Chí Minh', '0909123456', TO_DATE('1990-01-15', 'YYYY-MM-DD'));
```

```
INSERT INTO KHACHHANG (MaKH, HoTen, DiaChi, SDT_KH, NgaySinh)
```

```
VALUES ('KH02', N'Trần Thị B', N'45 Nguyễn Thị Thập, Phường Tân Phong, Quận 7, TP. Hồ Chí Minh', '0912345678', TO_DATE('1985-06-20', 'YYYY-MM-DD'));
```

```
INSERT INTO KHACHHANG (MaKH, HoTen, DiaChi, SDT_KH, NgaySinh)
```

```
VALUES ('KH03', N'Lê Văn C', N'210 Cách Mạng Tháng 8, Phường 10, Quận 3, TP. Hồ Chí Minh', '0987654321', TO_DATE('1992-09-10', 'YYYY-MM-DD'));
```

```
INSERT INTO KHACHHANG (MaKH, HoTen, DiaChi, SDT_KH, NgaySinh)
```

```
VALUES ('KH04', N'Phạm Thị D', N'78 Điện Biên Phủ, Phường 15, Quận Bình Thạnh, TP. Hồ Chí Minh', '0932123456', TO_DATE('1995-03-05', 'YYYY-MM-DD'));
```

```
INSERT INTO KHACHHANG (MaKH, HoTen, DiaChi, SDT_KH, NgaySinh)
```

```
VALUES ('KH05', N'Hoàng Văn E', N'36 Lê Văn Việt, Phường Hiệp Phú, TP. Thủ Đức, TP. Hồ Chí Minh', '0945123456', TO_DATE('1988-12-25', 'YYYY-MM-DD'));
```

```
-- DONHANG (CẬP NHẬT: thêm MaChiNhanh)
```

```
INSERT INTO DONHANG (MaDH, NgayDat, NgayGiao, TrangThai, MaKH, MaChiNhanh)
```

```
VALUES ('DH01', TO_DATE('2025-11-01', 'YYYY-MM-DD'), TO_DATE('2025-11-03', 'YYYY-MM-DD'), N'HOÀN THÀNH', 'KH01', 'CN02');
```

```
INSERT INTO DONHANG (MaDH, NgayDat, NgayGiao, TrangThai, MaKH, MaChiNhanh)
```

---

## CHƯƠNG 4: THIẾT KẾ VÀ CÀI ĐẶT HỆ THỐNG

```
VALUES ('DH02', TO_DATE('2025-11-02','YYYY-MM-DD'), NULL, N'DA XAC NHAN', 'KH02', 'CN02');
```

```
-- CHITIETDONHANG
```

```
INSERT INTO CHITIETDONHANG VALUES ('DH01', 'H01', 10, 150000, 150000); -- 10 * 15000
```

```
INSERT INTO CHITIETDONHANG VALUES ('DH02', 'H02', 5, 50000, 50000); -- 5 * 10000
```

```
-- DONNHAPHANG
```

```
INSERT INTO DONNHAPHANG VALUES ('N01', TO_DATE('2025-10-25','YYYY-MM-DD'), 'NV01');
```

```
INSERT INTO DONNHAPHANG VALUES ('N02', TO_DATE('2025-10-26','YYYY-MM-DD'), 'NV02');
```

```
-- CHITIETNHAPHANG
```

```
INSERT INTO CHITIETNHAPHANG VALUES ('N01', 'H01', 50, 12000);
```

```
INSERT INTO CHITIETNHAPHANG VALUES ('N02', 'H02', 60, 8000);
```

### 4.3.2. Cài đặt Store Procedure, Trigger

#### 4.3.2.1. Cài đặt Trigger

##### 4.3.2.1.1. Trigger giảm 10% vào ngày lễ và tăng 15% vào ngày lễ (14/2, 8/3, 20/10, 1/1)

```
CREATE OR REPLACE TRIGGER TRG_GIA_THANHTOAN
```

```
BEFORE INSERT OR UPDATE ON CHITIETDONHANG
```

```
FOR EACH ROW
```

```
DECLARE
```

```
    v_ngaydat DATE;
```

```
    v_ngaysinh DATE;
```

```
    v_dongiahua NUMBER(10,2);
```

```
    v_tongdongia NUMBER(12,2);
```

```
BEGIN
```

-- Lấy ngày đặt và ngày sinh

```
SELECT dh.NgayDat, kh.NgaySinh
INTO v_ngaydat, v_ngaysinh
FROM DONHANG dh
JOIN KHACHHANG kh ON dh.MaKH = kh.MaKH
WHERE dh.MaDH = :NEW.MaDH;
```

-- Lấy đơn giá hoa

```
SELECT DonGiaHoa INTO v_dongiahoa
FROM HOA
WHERE MaHoa = :NEW.MaHoa;
```

-- Tính tổng đơn giá gốc

```
v_tongdongia := :NEW.SoLuongBan * v_dongiahoa;
:NEW.TongDonGia := v_tongdongia;
```

-- BƯỚC 1: Tăng 15% nếu là ngày lễ

```
IF TO_CHAR(v_ngaydat, 'MM-DD') IN ('02-14', '03-08', '10-20', '01-01') THEN
    v_tongdongia := ROUND(v_tongdongia * 1.15, 2);
END IF;
```

-- BƯỚC 2: Giảm 10% nếu là sinh nhật

```
IF v_ngaysinh IS NOT NULL
    AND TO_CHAR(v_ngaydat, 'MM-DD') = TO_CHAR(v_ngaysinh, 'MM-DD') THEN
    v_tongdongia := ROUND(v_tongdongia * 0.9, 2);
END IF;
```



-- Gán giá thành toán

:NEW.GIATHANHTOAN := v\_tongdonggia;

EXCEPTION

WHEN NO\_DATA\_FOUND THEN

:NEW.TongDonGia := 0;

:NEW.GIATHANHTOAN := 0;

WHEN OTHERS THEN

RAISE\_APPLICATION\_ERROR(-20002, 'Lỗi tính giá thành: ' || SQLERRM);

END;

/

### **Lệnh test**

--Trường hợp 1: Đặt hàng vào đúng ngày sinh nhật → Được giảm 10%

-- Tạo đơn hàng vào đúng ngày sinh nhật KH01 (15/01)

INSERT INTO DONHANG VALUES ('DH03', TO\_DATE('2025-01-15','YYYY-MM-DD'),  
NULL, N'DA XAC NHAN', 'KH01', 'CN01');

-- Thêm chi tiết đơn hàng

INSERT INTO CHITIETDONHANG (MaDH, MaHoa, SoLuongBan, TongDonGia)

VALUES ('DH03', 'H01', 5, 15000); -- Giá gốc 15000

-- Kiểm tra kết quả

SELECT MaDH, MaHoa, SoLuongBan, TongDonGia, giathanhtoan

FROM CHITIETDONHANG

WHERE MaDH = 'DH03';

---

#### CHƯƠNG 4: THIẾT KẾ VÀ CÀI ĐẶT HỆ THỐNG

--Trường hợp 2: Đặt hàng KHÔNG vào ngày sinh nhật → Không giảm giá

-- Tạo đơn hàng vào ngày thường (ví dụ 08/11/2025) cho KH01

```
INSERT INTO DONHANG VALUES ('DH04', TO_DATE('2025-11-08','YYYY-MM-DD'),  
NULL, N'DA XAC NHAN', 'KH01', 'CN01');
```

-- Thêm chi tiết đơn hàng

```
INSERT INTO CHITIETDONHANG (MaDH, MaHoa, SoLuongBan, TongDonGia)  
VALUES ('DH04', 'H01', 3, 15000);
```

-- Kiểm tra kết quả

```
SELECT MaDH, MaHoa, SoLuongBan, TongDonGia, giathanhtoan  
FROM CHITIETDONHANG  
WHERE MaDH = 'DH04';
```

-----  
--Trường hợp 3: Đặt hàng vào ngày 14/02 → Tăng 15%

-- Tạo đơn hàng vào 14/02/2025

```
INSERT INTO DONHANG VALUES ('DH05', TO_DATE('2025-02-14','YYYY-MM-DD'),  
NULL, N'DA XAC NHAN', 'KH01', 'CN01');
```

-- Thêm chi tiết đơn hàng (giá gốc 15000)

```
INSERT INTO CHITIETDONHANG (MaDH, MaHoa, SoLuongBan, TongDonGia)  
VALUES ('DH05', 'H01', 5, 15000);
```

-- Kiểm tra kết quả

```
SELECT MaDH, MaHoa, SoLuongBan, TongDonGia, giathanhtoan  
FROM CHITIETDONHANG
```

---

#### CHƯƠNG 4: THIẾT KẾ VÀ CÀI ĐẶT HỆ THỐNG

WHERE MaDH = 'DH05';

-----  
-- Trường hợp 4. Khách hàng có sinh nhật trùng vào ngày lễ (14/2) ta có GiaThanhToan =  
TongDonGia + 15% - 10%

INSERT INTO KHACHHANG (MaKH, HoTen, DiaChi, SDT\_KH, NgaySinh)  
VALUES ('KH07', 'Nguyễn Văn Tèo', '123 Hai Bà Trưng', '0909111222', TO\_DATE('1990-02-14', 'YYYY-MM-DD'));

INSERT INTO DONHANG VALUES ('DH07', TO\_DATE('2025-02-14', 'YYYY-MM-DD'),  
NULL, N'DA XAC NHAN', 'KH07', 'CN01');

INSERT INTO CHITIETDONHANG (MaDH, MaHoa, SoLuongBan, TongDonGia)  
VALUES ('DH07', 'H01', 2, 15000);

SELECT MaDH, MaHoa, SoLuongBan, TongDonGia, giathanhtoan  
FROM CHITIETDONHANG  
WHERE MaDH = 'DH07';

-----  
-- Tổng kết kiểm tra

-- Xem tất cả chi tiết đơn hàng để so sánh

SELECT  
    dh.MaDH,  
    TO\_CHAR(dh.NgayDat, 'YYYY-MM-DD') AS NgayDat,  
    kh.HoTen,  
    TO\_CHAR(kh.NgaySinh, 'MM-DD') AS NgaySinh,  
    ct.MaHoa,

h.TenHoa,

h.DonGiaHoa,

ct.SoLuongBan,

ct.TongDonGia,

ct.GIATHANHTOAN,

CASE

WHEN TO\_CHAR(dh.NgayDat, 'MM-DD') IN ('02-14','03-08','10-20','01-01') THEN  
'Tăng 15%'

WHEN TO\_CHAR(dh.NgayDat, 'MM-DD') = TO\_CHAR(kh.NgaySinh, 'MM-DD')  
THEN 'Giảm 10%'

ELSE 'Bình thường'

END AS KhuyenMai

FROM DONHANG dh

JOIN KHACHHANG kh ON dh.MaKH = kh.MaKH

JOIN CHITIETDONHANG ct ON dh.MaDH = ct.MaDH

JOIN HOA h ON ct.MaHoa = h.MaHoa

WHERE dh.MaDH IN ('DH03', 'DH04', 'DH05', 'DH07')

ORDER BY dh.MaDH;

**4.3.2.1.2. Nếu khách hàng hủy đơn hàng trong vòng 2' sau khi đặt thì không bị tính phí, sau 2' thì tính phí hủy.**

ALTER TABLE DONHANG ADD (

ThoiGianHuy DATE,

PhiHuy NUMBER(12,2) DEFAULT 0,

LyDoHuy NVARCHAR2(255)

);

---

CREATE OR REPLACE TRIGGER TRG\_TINH\_PHI\_HUY\_DONHANG

BEFORE UPDATE OF TrangThai ON DONHANG

FOR EACH ROW

WHEN (NEW.TrangThai = 'HUY' AND OLD.TrangThai != 'HUY')

DECLARE

    v\_tong\_giathanhtoan NUMBER(12,2) := 0;

    v\_thoigian\_dat    DATE;

    v\_phut\_chenh\_lech  NUMBER;

BEGIN

    -- Lấy tổng GIATHANHTOAN của đơn hàng

    SELECT NVL(SUM(GIATHANHTOAN), 0)

    INTO v\_tong\_giathanhtoan

    FROM CHITIETDONHANG

    WHERE MaDH = :NEW.MaDH;

    -- Thời gian đặt hàng

    v\_thoigian\_dat := :OLD.NgayDat;

    -- Tính số phút chênh lệch từ lúc đặt đến lúc hủy (dùng SYSDATE)

    v\_phut\_chenh\_lech := ROUND((SYSDATE - v\_thoigian\_dat) \* 24 \* 60, 2);

    -- Gán thời gian hủy

    :NEW.ThoiGianHuy := SYSDATE;

    -- Quy tắc tính phí hủy

    IF v\_phut\_chenh\_lech <= 2 THEN

        :NEW.PhiHuy := 0;

        :NEW.LyDoHuy := 'Hủy trong 2 phút - Miễn phí';

    ELSE

        :NEW.PhiHuy := ROUND(v\_tong\_giathanhtoan \* 0.10, 2);

---

#### CHƯƠNG 4: THIẾT KẾ VÀ CÀI ĐẶT HỆ THỐNG

```
:NEW.LyDoHuy := 'Hủy sau ' || v_phut_chenh_lech || ' phút - Phí 10%';  
END IF;  
  
EXCEPTION  
  
  WHEN NO_DATA_FOUND THEN  
  
    :NEW.PhiHuy := 0;  
  
    :NEW.LyDoHuy := 'Không có chi tiết đơn hàng';  
  
  WHEN OTHERS THEN  
  
    RAISE_APPLICATION_ERROR(-20010, 'Lỗi tính phí hủy: ' || SQLERRM);  
  
END;  
  
/  
  
Lệnh test  
-- Bật output để xem thời gian  
SET SERVEROUTPUT ON;  
  
-- Test 1: Tạo đơn hàng mới  
INSERT INTO DONHANG (MaDH, NgayDat, TrangThai, MaKH, MaChiNhanh)  
VALUES ('DH99', SYSDATE, 'DA XAC NHAN', 'KH01', 'CN02');  
  
INSERT INTO CHITIETDONHANG (MaDH, MaHoa, SoLuongBan, TongDonGia)  
VALUES ('DH99', 'H01', 10, 150000);  
  
-- Trigger TRG_GIA_THANHTOAN sẽ tự chạy → GIATHANHTOAN = 150000 (giả sử  
không lỗi, không sinh nhật)  
  
-- Hủy ngay lập tức (trong vòng 2 phút)  
DBMS_OUTPUT.PUT_LINE('Test 1: Hủy trong vòng 2 phút');
```

---

#### CHƯƠNG 4: THIẾT KẾ VÀ CÀI ĐẶT HỆ THỐNG

```
UPDATE DONHANG SET TrangThai = 'HUY' WHERE MaDH = 'DH99';
```

```
-- Xem kết quả
```

```
SELECT MaDH, NgayDat, ThoiGianHuy,  
       ROUND((ThoiGianHuy - NgayDat)*24*60, 2) AS Phut_Chenh_Lech,  
       PhiHuy, LyDoHuy, TrangThai  
FROM DONHANG WHERE MaDH = 'DH99';
```

```
-----  
-- Test 2: Tạo đơn hàng mới và chờ hơn 2 phút rồi hủy
```

```
INSERT INTO DONHANG (MaDH, NgayDat, TrangThai, MaKH, MaChiNhanh)  
VALUES ('DH98', SYSDATE, 'DA XAC NHAN', 'KH01', 'CN02');
```

```
INSERT INTO CHITIETDONHANG (MaDH, MaHoa, SoLuongBan, TongDonGia)  
VALUES ('DH98', 'H01', 20, 300000); -- 20 bó x 15k
```

```
-- Chờ hơn 2 phút (bạn chạy lệnh này sau 2 phút 10 giây chẳng hạn)
```

```
-- Hoặc giả lập bằng cách sửa NgayDat cũ hơn (chỉ để test nhanh):
```

```
UPDATE DONHANG SET NgayDat = SYSDATE - 3/1440 WHERE MaDH = 'DH98'; --  
trừ 3 phút
```

```
DBMS_OUTPUT.PUT_LINE('Test 2: Hủy sau hơn 2 phút');
```

```
UPDATE DONHANG SET TrangThai = 'HUY' WHERE MaDH = 'DH98';
```

```
-- Xem kết quả
```

```
SELECT MaDH, NgayDat, ThoiGianHuy,  
       ROUND((ThoiGianHuy - NgayDat)*24*60, 2) AS Phut_Chenh_Lech,
```

PhiHuy, LyDoHuy

FROM DONHANG WHERE MaDH = 'DH98';

**4.3.2.1.3. Khách hàng VIP (Chi >= 20 triệu/năm) sẽ được hưởng ưu đãi 15% tất cả đơn hàng.**

ALTER TABLE KHACHHANG ADD (

LoaiKH VARCHAR2(10) DEFAULT 'THUONG' CHECK (LoaiKH IN ('THUONG', 'VIP')),

TongChiTieu NUMBER(15,2) DEFAULT 0,

NgayCapNhatVIP DATE

);

CREATE OR REPLACE PROCEDURE PROC\_CAPNHAT\_KHACHHANG\_VIP

IS

BEGIN

UPDATE KHACHHANG kh

SET LoaiKH = CASE

WHEN (

SELECT NVL(SUM(ct.GIATHANHTOAN), 0)

FROM DONHANG dh

JOIN CHITIETDONHANG ct ON dh.MaDH = ct.MaDH

WHERE dh.MaKH = kh.MaKH

AND dh.TrangThai = 'HOAN THANH'

AND dh.NgayDat >= ADD\_MONTHS(TRUNC(SYSDATE,'YYYY'), -12)

-- 12 tháng gần nhất

) >= 20000000 THEN 'VIP'

ELSE 'THUONG'



```
END,

TongChiTieu = (
    SELECT NVL(SUM(ct.GIATHANHTOAN), 0)
    FROM DONHANG dh
    JOIN CHITIETDONHANG ct ON dh.MaDH = ct.MaDH
    WHERE dh.MaKH = kh.MaKH
    AND dh.TrangThai = 'HOAN THANH'
    AND dh.NgayDat >= ADD_MONTHS(TRUNC(SYSDATE,'YYYY'), -12)
),

NgayCapNhatVIP = SYSDATE

WHERE kh.MaKH IN (
    SELECT MaKH FROM DONHANG WHERE TrangThai = 'HOAN THANH'
);

COMMIT;

DBMS_OUTPUT.PUT_LINE('Cập nhật VIP thành công cho tất cả khách hàng!');

END;

/
```

### **Lệnh test**

-- 1. Tạo khách hàng mới

```
INSERT INTO KHACHHANG (MaKH, HoTen, DiaChi, SDT_KH, NgaySinh)
VALUES ('KH99', N'Nguyễn Thị VIP', N'Quận 1', '0909999999', SYSDATE);
```

-- 2. Tạo 1 đơn hàng lớn (đủ 20 triệu trong năm nay)

```
INSERT INTO DONHANG (MaDH, NgayDat, TrangThai, MaKH, MaChiNhanh)
VALUES ('DH999', SYSDATE, 'HOAN THANH', 'KH99', 'CN02');
```

```
INSERT INTO CHITIETDONHANG (MaDH, MaHoa, SoLuongBan, TongDonGia)
VALUES ('DH999', 'H01', 1334, 20010000); -- ~20 triệu
```

-- 3. Cập nhật trạng thái VIP

```
EXEC PROC_CAPNHAT_KHACHHANG_VIP;
```

-- Kiểm tra: KH99 giờ là VIP

```
SELECT MaKH, HoTen, TongChiTieu, LoaiKH FROM KHACHHANG WHERE MaKH =
'KH99';
```

-- → LoaiKH = 'VIP', TongChiTieu  $\approx$  20,010,000

-- 4. Tạo đơn hàng mới → tự động được giảm 15%

```
INSERT INTO DONHANG (MaDH, NgayDat, TrangThai, MaKH, MaChiNhanh)
VALUES ('DH888', SYSDATE, 'DA XAC NHAN', 'KH99', 'CN02');
```

```
INSERT INTO CHITIETDONHANG (MaDH, MaHoa, SoLuongBan, TongDonGia)
VALUES ('DH888', 'H01', 10, 150000); -- giá gốc 150,000
```

-- Xem kết quả: GIATHANHTOAN phải  $\approx$  127,500 ( $150,000 \times 0.85$ )

```
SELECT MaDH, MaHoa, SoLuongBan, TongDonGia, GIATHANHTOAN
FROM CHITIETDONHANG WHERE MaDH = 'DH888';
```

#### **4.3.2.2. Cài đặt Store Procedure**

##### **4.3.2.2.1. Xem thống kê doanh thu đơn hàng theo chi nhánh và theo thời gian**

```
CREATE OR REPLACE PROCEDURE PROC_THONGKE_DOANHTHU_CHINHANH (
    p_tu_ngay IN DATE,
    p_den_ngay IN DATE
)
```

IS

-- Biến để lưu tổng hợp

v\_tong\_dh     NUMBER;

v\_tong\_doanhthu NUMBER(15,2);

BEGIN

-- Kiểm tra đầu vào

IF p\_tu\_ngay IS NULL OR p\_den\_ngay IS NULL THEN

    RAISE\_APPLICATION\_ERROR(-20001, 'Cả hai tham số ngày đều phải được cung cấp.');

END IF;

IF p\_den\_ngay < p\_tu\_ngay THEN

    RAISE\_APPLICATION\_ERROR(-20002, 'Ngày kết thúc phải lớn hơn hoặc bằng ngày bắt đầu.');

END IF;

-- In tiêu đề

DBMS\_OUT-

PUT.PUT\_LINE('=====');  
=====');

DBMS\_OUTPUT.PUT\_LINE('THỐNG KÊ DOANH THU THEO CHI NHÁNH');

DBMS\_OUTPUT.PUT\_LINE('Từ ngày: ' || TO\_CHAR(p\_tu\_ngay, 'DD/MM/YYYY') ||

    ' đến ngày: ' || TO\_CHAR(p\_den\_ngay, 'DD/MM/YYYY'));

DBMS\_OUT-

PUT.PUT\_LINE('=====');  
=====');

DBMS\_OUTPUT.PUT\_LINE(RPAD('Mã CN', 10) ||

    RPAD('Tên Chi Nhánh', 30) ||

    RPAD('Số ĐH', 10) ||

    RPAD('Doanh Thu', 15));

```
DBMS_OUTPUT.PUT_LINE('-----');

-- Duyệt từng chi nhánh và tính doanh thu
FOR rec IN (
    SELECT
        cn.MaChiNhanh,
        cn.TenChiNhanh,
        COUNT(dh.MaDH) AS SoDonHang,
        NVL(SUM(ct.GIATHANHTOAN), 0) AS DoanhThu
    FROM CHINHANH cn
    LEFT JOIN DONHANG dh ON cn.MaChiNhanh = dh.MaChiNhanh
        AND dh.NgayDat BETWEEN p_tu_ngay AND p_den_ngay
        AND dh.TrangThai = 'HOAN THANH' -- Chi tính đơn hoàn thành
    LEFT JOIN CHITIETDONHANG ct ON dh.MaDH = ct.MaDH
    GROUP BY cn.MaChiNhanh, cn.TenChiNhanh
    ORDER BY DoanhThu DESC
) LOOP
    DBMS_OUTPUT.PUT_LINE(
        RPAD(NVL(rec.MaChiNhanh, ''), 10) ||
        RPAD(NVL(rec.TenChiNhanh, ''), 30) ||
        RPAD(rec.SoDonHang, 10) ||
        TO_CHAR(rec.DoanhThu, 'FM999,999,990.00')
    );

    -- Cộng dồn tổng
    v_tong_dh := v_tong_dh + rec.SoDonHang;
    v_tong_doanhthu := v_tong_doanhthu + rec.DoanhThu;
END LOOP;
```

```
-- In dòng tổng kết
DBMS_OUTPUT.PUT_LINE('-----');
DBMS_OUTPUT.PUT_LINE(
    RPAD('TỔNG CỘNG', 40) ||
    RPAD(v_tong_dh, 10) ||
    TO_CHAR(v_tong_doanhthu, 'FM999,999,990.00')
);
DBMS_OUTPUT.PUT_LINE('=====
=====');

EXCEPTION

    WHEN OTHERS THEN

        RAISE_APPLICATION_ERROR(-20003, 'Lỗi trong PROC_THONGKE_DO-
        ANHTHU_CHINHANH: ' || SQLERRM);

END;

/
```

### **Lệnh gọi**

-- Bật output để xem kết quả

SET SERVEROUTPUT ON;

-- Ví dụ 1: Doanh thu tháng 11/2025 (từ 01/11 đến 08/11)

```
EXEC PROC_THONGKE_DOANHTHU_CHINHANH(TO_DATE('2025-11-01','YYYY-MM-
DD'), TO_DATE('2025-11-08','YYYY-MM-DD'));
```

-- Ví dụ 2: Doanh thu toàn bộ tháng 2/2025 (có ngày lễ 14/2)

```
EXEC PROC_THONGKE_DOANHTHU_CHINHANH(TO_DATE('2025-02-01','YYYY-MM-
DD'), TO_DATE('2025-02-28','YYYY-MM-DD'));
```

-- Ví dụ 3: Doanh thu năm 2025

```
EXEC PROC_THONGKE_DOANHTHU_CHINHANH(TO_DATE('2025-01-01','YYYY-MM-DD'), TO_DATE('2025-12-31','YYYY-MM-DD'));
```

```
select * from Donhang;
```

#### **4.3.2.2.2. Procedure Thêm một khách hàng mới vào hệ thống cửa hàng hoa một cách an toàn, có kiểm tra, có ghi log, và tự động lưu dữ liệu.**

-- Create Table để ghi Log

```
CREATE TABLE LOG_KHACHHANG (  
    LogID      NUMBER GENERATED BY DEFAULT AS IDENTITY PRIMARY KEY,  
    MaKH       CHAR(10),  
    HanhDong   VARCHAR2(50),  
    ThoiGian   DATE      DEFAULT SYSDATE,  
    NguoiThucHien VARCHAR2(30) DEFAULT USER  
);
```

-----  
-- drop procedure ThemKhachHang;

-- Procedure Thêm một khách hàng mới vào hệ thống cửa hàng hoa một cách an toàn, có kiểm tra, có ghi log, và tự động lưu dữ liệu.

```
CREATE OR REPLACE PROCEDURE ThemKhachHang (  
    p_MaKH   IN CHAR,  
    p_HoTen  IN NVARCHAR2,  
    p_DiaChi IN NVARCHAR2,  
    p_SDT    IN VARCHAR2,  
    p_NgaySinh IN DATE  
) AS  
    v_count NUMBER;
```

BEGIN

-- 1. Kiểm tra mã KH trùng

SELECT COUNT(\*) INTO v\_count FROM KHACHHANG WHERE MaKH = p\_MaKH;

IF v\_count > 0 THEN

    RAISE\_APPLICATION\_ERROR(-20001, 'Mã khách hàng đã tồn tại!');

END IF;

-- 2. Kiểm tra SĐT hợp lệ

IF NOT REGEXP\_LIKE(p\_SDT, '^0[0-9]{9}\$') THEN

    RAISE\_APPLICATION\_ERROR(-20002,  
        'Số điện thoại phải có 10 số, bắt đầu bằng 0!');

END IF;

-- 3. Thêm dữ liệu

INSERT INTO KHACHHANG (MaKH, HoTen, DiaChi, SDT\_KH, NgaySinh)

VALUES (p\_MaKH, p\_HoTen, p\_DiaChi, p\_SDT, p\_NgaySinh);

-- 4. Ghi log

INSERT INTO LOG\_KHACHHANG (MaKH, HanhDong)

VALUES (p\_MaKH, 'THÊM MỚI');

COMMIT;

DBMS\_OUTPUT.PUT\_LINE('Thêm khách hàng thành công: ' || p\_MaKH);

EXCEPTION

WHEN OTHERS THEN

    ROLLBACK;

    RAISE\_APPLICATION\_ERROR(-20003, 'Lỗi: ' || SQLERRM);

END;

/

### **Lệnh thêm**

-- Thêm Khách Hàng

SET SERVEROUTPUT ON;

BEGIN

ThemKhachHang(

p\_MaKH => 'KH09',

p\_HoTen => N'Đỗ Tài',

p\_DiaChi => N'12 Nguyễn Văn Cừ, Quận 5, TP.HCM',

p\_SDT => '0909456123',

p\_NgaySinh => TO\_DATE('1997-02-18', 'YYYY-MM-DD')

);

END;

/

select \* from LOG\_KHACHHANG;

select \* from Khachhang;

#### **4.3.2.2.3. Procedure Xóa khách hàng chỉ khi chưa có đơn hàng**

-- Xóa khách hàng chỉ khi chưa có đơn hàng

CREATE OR REPLACE PROCEDURE XoaKhachHang (

p\_MaKH IN CHAR

) AS

v\_count\_dh NUMBER;

BEGIN

-- Kiểm tra khách hàng có đơn hàng chưa

SELECT COUNT(\*) INTO v\_count\_dh

FROM DONHANG

WHERE MaKH = p\_MaKH;

IF v\_count\_dh > 0 THEN



```
        RAISE_APPLICATION_ERROR(-20004, 'Không thể xóa! Khách hàng có đơn hàng.');
```

END IF;

-- Xóa khách hàng

```
DELETE FROM KHACHHANG WHERE MaKH = p_MaKH;
```

IF SQL%ROWCOUNT = 0 THEN

```
        RAISE_APPLICATION_ERROR(-20005, 'Không tìm thấy khách hàng: ' || p_MaKH);
```

END IF;

-- Ghi log xóa

```
INSERT INTO LOG_KHACHHANG (MaKH, HanhDong)
VALUES (p_MaKH, 'XÓA');
```

COMMIT;

DBMS\_OUTPUT.PUT\_LINE('Xóa khách hàng thành công: ' || p\_MaKH);

EXCEPTION

WHEN OTHERS THEN

```
        ROLLBACK;
```

```
        RAISE_APPLICATION_ERROR(-20006, 'Lỗi xóa: ' || SQLERRM);
```

END;

/

### **Lệnh xóa**

```
-- Xóa khách hàng chỉ khi chưa có đơn hàng
```

```
CREATE OR REPLACE PROCEDURE XoaKhachHang (
    p_MaKH IN CHAR
) AS
```

```
v_count_dh NUMBER;

BEGIN

-- Kiểm tra khách hàng có đơn hàng chưa
SELECT COUNT(*) INTO v_count_dh
FROM DONHANG
WHERE MaKH = p_MaKH;

IF v_count_dh > 0 THEN
    RAISE_APPLICATION_ERROR(-20004, 'Không thể xóa! Khách hàng có đơn hàng.');
```

END IF;

```
-- Xóa khách hàng
DELETE FROM KHACHHANG WHERE MaKH = p_MaKH;

IF SQL%ROWCOUNT = 0 THEN
    RAISE_APPLICATION_ERROR(-20005, 'Không tìm thấy khách hàng: ' || p_MaKH);
END IF;

-- Ghi log xóa
INSERT INTO LOG_KHACHHANG (MaKH, HanhDong)
VALUES (p_MaKH, 'XÓA');
```

COMMIT;

```
DBMS_OUTPUT.PUT_LINE('Xóa khách hàng thành công: ' || p_MaKH);
EXCEPTION

WHEN OTHERS THEN

    ROLLBACK;
```

---

#### CHƯƠNG 4: THIẾT KẾ VÀ CÀI ĐẶT HỆ THỐNG

```
        RAISE_APPLICATION_ERROR(-20006, 'Lỗi xóa: ' || SQLERRM);
END;
/

-----\

-- Xóa Khách hàng chưa có đơn hàng
SET SERVEROUTPUT ON;

BEGIN

    XoaKhachHang(p_MaKH => 'KH08');
END;
/

-----

-- Xóa khách hàng đã có đơn hàng
SET SERVEROUTPUT ON;

BEGIN

    XoaKhachHang(p_MaKH => 'KH01');
END;
/

4.3.2.2.3. Procedure thống kê Top 10 khách hàng chi tiêu nhiều nhất
-- =====
-- 3. PROCEDURE: Top 10 khách hàng chi tiêu nhiều nhất (có năm)
-- Dùng để in báo cáo, tặng quà VIP
-- =====

CREATE OR REPLACE PROCEDURE PROC_TOP10_KHACHHANG (
    p_nam IN NUMBER DEFAULT EXTRACT(YEAR FROM SYSDATE)
)
IS
```

BEGIN

DBMS\_OUT-  
PUT.PUT\_LINE('=====');

DBMS\_OUTPUT.PUT\_LINE(' TOP 10 KHÁCH HÀNG CHI TIÊU NHIỀU NHẤT NĂM ' ||  
p\_nam);

DBMS\_OUT-  
PUT.PUT\_LINE('=====');

FOR rec IN (  
SELECT  
ROW\_NUMBER() OVER (ORDER BY NVL(SUM(ct.GIATHANHTOAN),0) DESC) AS  
STT,  
kh.MaKH,  
kh.HoTen,  
kh.SDT\_KH,  
TO\_CHAR(NVL(SUM(ct.GIATHANHTOAN),0), 'FM999,999,990') || ' đ' AS TongChiTieu  
FROM KHACHHANG kh  
LEFT JOIN DONHANG dh ON kh.MaKH = dh.MaKH  
AND dh.TrangThai = 'HOAN THANH'  
AND EXTRACT(YEAR FROM dh.NgayDat) = p\_nam  
LEFT JOIN CHITIETDONHANG ct ON dh.MaDH = ct.MaDH  
GROUP BY kh.MaKH, kh.HoTen, kh.SDT\_KH  
ORDER BY TongChiTieu DESC  
) LOOP  
EXIT WHEN rec.STT > 10;  
DBMS\_OUTPUT.PUT\_LINE(  
RPAD(rec.STT, 4) ||  
RPAD(rec.MaKH, 12) ||  
RPAD(rec.HoTen, 30) ||

```
RPAD(rec.SDT_KH, 15) ||  
rec.TongChiTieu  
);  
END LOOP;  
END;  
/
```

**Lệnh gọi**

```
EXEC PROC_TOP10_KHACHHANG(2025);
```

**4.3.3. Minh họa vấn đề xử lý đồng thời**

**Tình huống:** Hai nhân viên cùng cập nhật số lượng tồn kho của cùng một loại hoa

• Mô tả:

- Sản phẩm: Hoa Hồng Đỏ – Mã SP: H01
- Số lượng tồn ban đầu: 100 bông
- Nhân viên A thực hiện kiểm kho và cập nhật xuất 10 bông.
- Cùng lúc đó, nhân viên B kiểm kho và cập nhật nhập thêm 5 bông.
- Cả A và B đều đọc số lượng tồn = 100, vì hai giao tác xảy ra gần như đồng thời.

Diễn biến thực tế xảy ra:

A cập nhật:  $100 - 10 = 90$

B cập nhật:  $100 + 5 = 105$

- Kết quả cuối cùng bị ghi sai, tùy giao tác nào ghi sau:

Nếu A ghi sau → tồn còn 95

Nếu B ghi sau → tồn còn 105

=> Dữ liệu gặp vấn đề Lost Update (cập nhật bị mất).

• **Phân tích :**

SLTon=100	T1	T2
t1	Read(SLTon = 100)	
t2		Read(SLTon = 100)
t3	SLTon = 100 - 10 = 90	
t4	Write(SLTon = 90)	
t5		SLTon = 100 + 5 = 105
t6		Write(SLTon = 105)

➤ Kết quả cuối cùng: **SLTon = 105** → sai vì mất cập nhật của A.

• **Giải pháp : Sử dụng mức cô lập SERIALIZABLE**

SLTon=100	T1	T2
t1	Read(SLTon = 100)	
t2	SLTon = 90	
t3	Write(SLTon = 90)	
t4		Read(SLTon = 90)
t5		SLTon = 90 + 5 = 95
t6		Write(SLTon = 95)

➤ **Kết quả đúng cuối cùng: SLTon = 95**

➤ Không xảy ra lost update.

#### 4.3.4. Thiết kế và cài đặt giao tác (Transaction)

Cài đặt Transaction trong Oracle

Procedure: TAO\_DON\_HANG

CREATE OR REPLACE PROCEDURE TAO\_DON\_HANG (

p\_maKH IN CHAR,

p\_maHoa IN CHAR,

p\_soluong IN NUMBER,

p\_maChiNhanh IN CHAR

) AS

v\_dongia NUMBER(10,2);

```
v_tongDG    NUMBER(12,2);
v_maDH      CHAR(10);
BEGIN
    -- 1. Lấy đơn giá hoa
    SELECT DonGiaHoa INTO v_dongia
    FROM HOA
    WHERE MaHoa = p_maHoa;

    -- 2. Tính tổng đơn giá
    v_tongDG := v_dongia * p_soluong;

    -- 3. Tạo đơn hàng
    v_maDH := 'DH' || TO_CHAR(SEQ_DON.NEXTVAL, 'FM0000'); -- tạo mã đơn hàng
    dạng DH0001
    INSERT INTO DONHANG (MaDH, MaKH, MaChiNhanh)
    VALUES (v_maDH, p_maKH, p_maChiNhanh);

    -- 4. Thêm chi tiết đơn hàng
    INSERT INTO CHITIETDONHANG (MaDH, MaHoa, SoLuongBan, TongDonGia, Gi-
    aThanhToan)
    VALUES (v_maDH, p_maHoa, p_soluong, v_tongDG, v_tongDG); -- giả sử không giảm
    giá

    COMMIT;
EXCEPTION
    WHEN OTHERS THEN
        ROLLBACK;
```

RAISE;

END;

/

#### 4.3.5. Thiết kế và cài đặt mô hình phân quyền của hệ thống

Trong hệ thống quản lý cửa hàng hoa, việc phân quyền người dùng là cần thiết nhằm đảm bảo tính bảo mật dữ liệu và giới hạn quyền truy cập phù hợp cho từng bộ phận. Oracle hỗ trợ cơ chế phân quyền dựa trên User – Role – Privilege, cho phép quản trị viên thiết lập các quyền truy cập chi tiết cho từng nhóm người dùng.

Việc thiết kế mô hình phân quyền được thực hiện dựa trên phân tích nghiệp vụ của cửa hàng, trong đó mỗi bộ phận đảm nhiệm một vai trò khác nhau: Bán hàng, Quản lý, Kho. Hệ thống cần đảm bảo rằng nhân viên bán hàng chỉ được phép lập hóa đơn; nhân viên kho chỉ được phép xem và cập nhật tồn kho; quản lý được phép xem toàn bộ dữ liệu và phê duyệt các chức năng quan trọng.

Dưới đây là bảng Mô hình phân quyền của hệ thống quản lý cửa hàng hoa

Bảng 4.3.5. Mô hình phân quyền của hệ thống quản lý cửa hàng hoa

STT	Bộ phận	Chức Vụ	Quyền được cấp	Phạm vi
1	Bán hàng	Nhân viên	SELECT, INSERT, UPDATE	HoaDon, ChiTietHD, KhachHang, SanPham
2		Quản lý bán hàng	SELECT, INSERT, UPDATE, DELETE	HoaDon, ChiTietHD, SanPham, KhachHang, NhanVien
3	Kho	Nhân viên kho	SELECT, UPDATE	SanPham, LoHang, ChiTietPN, TonKho (SoLuongTon)
4		Quản lý kho	SELECT, INSERT,	SanPham,



			UPDATE	LoHang, PhieuNhap, ChiTietPN
5	Quản lý chung	Chủ cửa hàng	ALL PRIVILEGES	Tất cả các bảng trong hệ thống
6	Kế toán	Nhân viên kế toán	SELECT, INSERT, UPDATE	HoaDon, ChiTietHD, PhieuNhap, ChiTietPN, CongNo-KhachHang, CongNoNhaCung-Cap, KháchHang
7	Kế toán	Quản lý kế toán	SELECT, INSERT, UPDATE, DELETE	HoaDon, ChiTietHD, PhieuNhap, ChiTietPN, CongNo-KhachHang, CongNoNhaCung-Cap, KháchHang, BaoCaoDoanhThu, BaoCaoCongNo

#### 4.3.5.1. Cài đặt phân quyền

Dựa trên mô hình phân quyền đã được thiết kế cho hệ thống quản lý cửa hàng hoa, phần này trình bày chi tiết các bước cài đặt kỹ thuật để hiện thực hóa mô hình đó.

- Tạo Role cho từng bộ phận

```
CREATE ROLE ROLE_BANHANG_NV;
```

```
CREATE ROLE ROLE_BANHANG_QL;
```

```
CREATE ROLE ROLE_KHO_NV;
```

```
CREATE ROLE ROLE_KHO_QL;
```

```
CREATE ROLE ROLE_CHUCUAHANG;
```

```
CREATE ROLE ROLE_KE_TOAN_NV;
```

```
CREATE ROLE ROLE_KE_TOAN_QL;
```

Cấp quyền cho từng Role

**Nhân viên bán hàng**

GRANT SELECT, INSERT, UPDATE ON HoaDon TO ROLE\_BANHANG\_NV;

GRANT SELECT, INSERT, UPDATE ON ChiTietHD TO ROLE\_BANHANG\_NV;

GRANT SELECT ON KháchHang TO ROLE\_BANHANG\_NV;

GRANT SELECT ON SanPham TO ROLE\_BANHANG\_NV;

**Quản lý bán hàng**

GRANT SELECT, INSERT, UPDATE, DELETE ON HoaDon TO ROLE\_BANHANG\_QL;

GRANT SELECT, INSERT, UPDATE, DELETE ON ChiTietHD TO ROLE\_BANHANG\_QL;

GRANT SELECT, UPDATE ON SanPham TO ROLE\_BANHANG\_QL;

**Nhân viên kho**

GRANT SELECT, UPDATE ON SanPham TO ROLE\_KHO\_NV;

GRANT SELECT, UPDATE ON LoHang TO ROLE\_KHO\_NV;

GRANT SELECT, UPDATE ON ChiTietPN TO ROLE\_KHO\_NV;

**Quản lý kho**

GRANT SELECT, INSERT, UPDATE ON SanPham TO ROLE\_KHO\_QL;

GRANT SELECT, INSERT, UPDATE ON LoHang TO ROLE\_KHO\_QL;

GRANT SELECT, INSERT, UPDATE ON PhieuNhap TO ROLE\_KHO\_QL;

**Nhân viên kế toán**

GRANT SELECT, INSERT, UPDATE ON HoaDon TO ROLE\_KE\_TOAN\_NV;

GRANT SELECT, INSERT, UPDATE ON ChiTietHD TO ROLE\_KE\_TOAN\_NV;

GRANT SELECT, INSERT, UPDATE ON PhieuNhap TO ROLE\_KE\_TOAN\_NV;

GRANT SELECT, INSERT, UPDATE ON ChiTietPN TO ROLE\_KE\_TOAN\_NV;

GRANT SELECT ON KhachHang TO ROLE\_KE\_TOAN\_NV;

### **Quản lý kế toán**

GRANT SELECT, INSERT, UPDATE, DELETE ON HoaDon TO ROLE\_KE\_TOAN\_QL;

GRANT SELECT, INSERT, UPDATE, DELETE ON ChiTietHD TO  
ROLE\_KE\_TOAN\_QL;

GRANT SELECT, INSERT, UPDATE, DELETE ON PhieuNhap TO  
ROLE\_KE\_TOAN\_QL;

GRANT SELECT, INSERT, UPDATE, DELETE ON ChiTietPN TO  
ROLE\_KE\_TOAN\_QL;

GRANT SELECT, INSERT, UPDATE, DELETE ON BaoCaoDoanhThu TO  
ROLE\_KE\_TOAN\_QL;

### **Chủ cửa hàng**

GRANT ALL PRIVILEGES TO ROLE\_CHUCUAHANG;

#### **○ Tạo User và gán Role tương ứng**

CREATE USER NV\_BANHANG01 IDENTIFIED BY 123;

GRANT ROLE\_BANHANG\_NV TO NV\_BANHANG01;

CREATE USER KHO\_NV01 IDENTIFIED BY 123;

GRANT ROLE\_KHO\_NV TO KHO\_NV01;

CREATE USER QL\_BANHANG IDENTIFIED BY 123;

GRANT ROLE\_BANHANG\_QL TO QL\_BANHANG;

CREATE USER KT\_NV01 IDENTIFIED BY 123;

GRANT ROLE\_KE\_TOAN\_NV TO KT\_NV01;

CREATE USER QL\_KE\_TOAN IDENTIFIED BY 123;

GRANT ROLE\_KE\_TOAN\_QL TO QL\_KE\_TOAN;

CREATE USER CHUCUAHANG IDENTIFIED BY 123;

GRANT ROLE\_CHUCUAHANG TO CHUCUAHANG;

## CHƯƠNG 5. KẾT LUẬN

### 5.1. KẾT QUẢ ĐẠT ĐƯỢC

Sau quá trình thiết kế và cài đặt hệ thống cơ sở dữ liệu cho Shop Hoa, nhóm đã hoàn thành các phần quan trọng, giúp hệ thống vận hành ổn định và kiểm soát tốt nghiệp vụ.

STT	Những điều đã thực hiện
1	Trigger giảm 10% vào ngày lễ và tăng 15% vào ngày lễ (14/2, 8/3, 20/10, 1/1)
2	Xem thông kê doanh thu đơn hàng theo chi nhánh và theo thời gian
3	Procedure Thêm một khách hàng mới vào hệ thống cửa hàng hoa một cách an toàn, có kiểm tra, có ghi log, và tự động lưu dữ liệu.
4	Procedure Xóa khách hàng chỉ khi chưa có đơn hàng
5	SĐT chi nhánh và khách hàng bắt đầu bằng 0, có 10 hoặc 11 chữ số
6	Số lượng bán phải $\geq 1$
7	Đơn giá hoa phải $> 0$
8	Đơn giá bán trong đơn hàng phải $> 0$
9	Đơn giá nhập phải $> 0$
10	Ngày giao phải lớn hơn hoặc bằng ngày đặt đơn hàng.
11	Trạng thái đơn hàng chỉ nhận (“CHO XAC NHAN”, “DA XAC NHAN”, “DANG GIAO”, “HOAN TAT”, “HUY”)
12	Ngày sinh của khách hàng (nếu có) phải nhỏ hơn ngày hiện tại và khách hàng phải $\geq 15$ tuổi.
13	Khách hàng VIP (Chi $\geq 20$ triệu/năm) sẽ được hưởng ưu đãi 15% tất cả đơn hàng
14	Nếu khách hàng hủy đơn hàng trong vòng 24h sau khi đặt thì không bị tính phí, sau 24h thì tính phí hủy.
15	Xem được thông kê doanh thu đơn hàng theo chi nhánh và theo thời gian.
16	Procedure thống kê Top 10 khách hàng chi tiêu nhiều nhất

### 5.2. KẾT QUẢ CHƯA ĐẠT ĐƯỢC

STT	Những điều chưa làm được
1	Một khách hàng không được có quá 5 đơn hàng chưa thanh toán.
2	Hoa tồn kho quá 5 ngày kể từ ngày nhập sẽ bị đánh dấu “Xả hàng”.
3	Một chi nhánh phải có ít nhất 1 nhân viên bán hàng và 1 nhân viên kho.
4	Nhân viên không được làm việc ở 2 chi nhánh cùng lúc.
5	Tên hoa không được để trống và không được trùng trong cùng một chi nhánh.

6	Đơn hàng đã hủy thì không được phép cập nhật thêm chi tiết sản phẩm hay thay đổi thông tin thanh toán.
7	Báo cáo doanh thu chi nhánh chỉ tính với đơn hàng “HOAN THANH” + “Đã thanh toán”.
8	Khách hàng sẽ được giảm 10% cho đơn hàng đầu tiên
9	Backup-Restore

### **5.3. HƯỚNG PHÁT TRIỂN VÀ MỞ RỘNG TRONG TƯƠNG LAI**

Trong tương lai, hệ thống quản lý shop hoa có thể được phát triển và mở rộng theo nhiều hướng nhằm đáp ứng nhu cầu thực tế, nâng cao hiệu quả vận hành và gia tăng trải nghiệm của người dùng. Một số định hướng chính bao gồm:

#### **1. Mở rộng chức năng quản lý bán hàng và chăm sóc khách hàng**

- Tích hợp hệ thống tích điểm – hạng thành viên, tự động tính điểm dựa trên tổng chi tiêu.
- Tự động gửi email/SMS nhắc đơn hàng, chúc mừng sinh nhật, hoặc gợi ý sản phẩm theo lịch sử mua hàng.
- Áp dụng AI gợi ý hoa theo sở thích và theo dịp (sinh nhật, lễ tình nhân, lễ tốt nghiệp...).

#### **2. Xây dựng ứng dụng web/mobile dành cho khách hàng**

- Cho phép khách hàng đặt hoa online, theo dõi tình trạng đơn hàng theo thời gian thực.
- Cho phép thanh toán qua Ví điện tử (Momo, ZaloPay), ngân hàng, hoặc thẻ.
- Hiện thị bản đồ giao hàng khi shipper đang vận chuyển.

#### **3. Hỗ trợ quản lý nhiều chi nhánh (Multi-Branch)**

- Mở rộng mô hình cơ sở dữ liệu để:
  - Quản lý tồn kho từng chi nhánh.
  - Chuyển hàng giữa các chi nhánh (Transfer Order).
  - Đồng bộ dữ liệu theo thời gian thực.
- Tích hợp dashboard theo dõi doanh thu – tồn kho của từng chi nhánh hoặc toàn hệ thống.

#### **4. Tự động hóa quy trình nhập hàng và tồn kho**

- Kết nối với nhà cung cấp để:
  - Tự động gửi phiếu đặt hoa khi lượng tồn kho xuống thấp.
  - Theo dõi chất lượng hoa tươi theo ngày nhập – ngày hết hạn.
- Sử dụng cảm biến IoT (nếu mở rộng thực tế) để:

- Theo dõi nhiệt độ kho lạnh.
- Cảnh báo khi điều kiện bảo quản không đạt chuẩn.

## **5. Nâng cấp hệ thống báo cáo và dự đoán**

- Áp dụng Machine Learning để:
  - Dự đoán lượng hoa cần nhập theo mùa, dịp lễ.
  - Dự báo doanh thu và chi phí hoạt động.
- Tạo thêm các báo cáo nâng cao:
  - Báo cáo lợi nhuận theo dòng hoa.
  - Báo cáo tỉ lệ hư hỏng hoa tươi.
  - Dashboard phân tích khách hàng thân thiết.

## **6. Nâng cấp bảo mật và phân quyền hệ thống**

- Sử dụng JWT hoặc OAuth2.0 để bảo vệ API.
- Thêm tính năng ghi log truy cập, phát hiện bất thường.
- Phân quyền chi tiết (RBAC):
  - Chủ cửa hàng
  - Quản lý chi nhánh
  - Nhân viên bán hàng
  - Nhân viên kho

## **7. Triển khai hệ thống thực tế (Deployment)**

- Triển khai CSDL và hệ thống backend lên Cloud (AWS, Azure, GCP).
- Tối ưu hệ thống:
  - Caching (Redis)
  - Load balancing
  - Backup & Recovery tự động hằng ngày.
- Xây dựng cơ chế auto scale cho mùa cao điểm (Valentine, 8/3, 20/10...).

## **8. Tích hợp hệ thống giao hàng và mạng xã hội**

- Kết nối API với đơn vị vận chuyển (GHTK, GHN, AhaMove).
- Tự động đồng bộ sản phẩm lên Facebook Shop, Zalo Shop, giúp mở rộng kinh doanh.
- Tích hợp chatbot tư vấn tự động.

## BẢNG PHÂN CÔNG

CHƯƠNG	Nội dung thực hiện	Phan Ngọc Loan	Nguyễn Thanh Thúy Vy	Khôi
1	Giới thiệu tổng quan		X	X
	Cơ cấu tổ chức hệ thống	X	X	
	Đánh giá hiện trạng		X	X
	Phạm vi, quy trình và nghiệp vụ		X	X
	Các chứng từ báo cáo và số liệu		X	X
	Hiệu năng xử lý dữ liệu trong hệ thống		X	X
	An toàn dữ liệu		X	
	Xử lý truy xuất đồng thời		X	
	Lý do sử dụng hệ quản trị Oracle		X	
	Mô hình phân cấp chức năng của hệ thống (sơ đồ BFD)	X		
	Mô tả chi tiết chức năng	X	X	



3	Mô hình thực thể kết hợp (ERD)	X		
	Quy định về dữ liệu		X	
	Các sơ đồ luồng dữ liệu (DFD)	X		
4	Mô hình dữ liệu quan hệ	X	X	
	Từ điển dữ liệu (data dictionary)	X	X	
	Quản lý lưu trữ dữ liệu	X	X	
	Thiết kế cài đặt Stored Procedure			X
	Thiết kế cài đặt Trigger	X		

