

**TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT TP. HỒ CHÍ MINH KHOA**

**CÔNG NGHỆ THÔNG TIN**

**BỘ MÔN HỆ THỐNG THÔNG TIN**

-----



**ĐỀ TÀI CUỐI KỲ MÔN HỆ QUẢN TRỊ CSDL**

**TÊN ĐỀ TÀI:**

**QUẢN LÝ QUÁN CÀ PHÊ**

**GVHD:** *TS. Nguyễn Thành Sơn*

**Lớp HP:** *DBMS330284\_22\_2\_03CLC*

**Nhóm thực hiện:** *Nhóm 01*

**Học kỳ:** *2*

**Năm học:** *2022 - 2023*

*Thành phố Hồ Chí Minh, tháng 5, năm 2023*



## DANH SÁCH SINH VIÊN NHÓM THỰC HIỆN

HỌC KÌ 1 NĂM HỌC 2022-2023

Nhóm 01

*Đề tài: Quản lý quán cà phê*

21110202	Bùi Quốc Khang (Chuyển ca từ DBSY230184_22_1_01CLC)
21110273	Phạm Hùng Phong
21110842	Đặng Minh Nhật (Chuyển ca từ DBSY230184_22_1_01CLC)
21110849	Đặng Phú Quý

## **NHẬN XÉT CỦA GV**

Tp. Hồ Chí Minh, ngày ....., tháng 11, năm 2022

Giảng viên chấm điểm

# MỤC LỤC

<b>CHƯƠNG 1: TỔNG QUAN VỀ HỆ THỐNG</b>	<b>1</b>
1. 1	
1.1 . Mô tả bài toán	1
1.2 . Nghiệp vụ của bài toán	3
1.3 . Mô tả chức năng của bài toán	5
1.4. Tính lương nhân viên:	6
<b>CHƯƠNG 2: PHÂN TÍCH THIẾT KẾ HỆ THỐNG</b>	<b>8</b>
1. 8	
2. 9	
3. 10	
4. 12	
5. 17	
6. 18	
<b>CHƯƠNG 3: THIẾT KẾ CÁC CHỨC NĂNG</b>	<b>22</b>
1. 22	
2. 22	
2.1. Danh mục nhân viên	23
2.2. Danh mục bảng phân ca	23
2.3. Danh mục ca làm việc	23
2.4. Danh mục hóa đơn	23
2.5. Danh mục sản phẩm	24
2.6. Danh mục phiếu chi	24
2.8. Danh mục nhà cung cấp	25
3. 25	
3.1. Hàm tìm kiếm khách hàng	26
3.2. Tạo khách hàng mới	26
3.3. Trigger bắt lỗi khi thêm khách hàng mới	27
4. 27	
4.1. Xuất hóa đơn	28
4.2. Tìm kiếm hóa đơn	29
5. 29	
5.1. Tìm kiếm nguyên liệu	30
5.2. Thêm nguyên liệu	30
6. 30	
6.1. Thêm sản phẩm	31

6.2.	Sửa sản phẩm	32
6.3.	Xóa sản phẩm	33
6.4.	Tìm kiếm sản phẩm	34
6.5.	Trigger bắt lỗi trùng tên khi thêm và chỉnh sửa sản phẩm	34
<b>7.</b>	<b>34</b>	
7.1.	Doanh thu theo ngày, tháng, năm	35
7.2.	Chi phí theo tháng, năm	35
<b>8.</b>	<b>36</b>	
8.1.	Thêm nhân viên	37
8.2.	Xóa nhân viên	38
8.3.	Tìm kiếm nhân viên	39
8.4.	Tính lương nhân viên	41
8.5.	Sửa thông tin nhân viên	41
8.6.	Trigger bắt lỗi khi thêm hoặc sửa thông tin nhân viên	42
<b>9.</b>	<b>42</b>	
9.1.	Thêm phiếu chi	44
<b>10.</b>	<b>43</b>	
10.1.	Thêm ca cho nhân viên	45
<b>CHƯƠNG 4: TẠO USER VÀ PHÂN QUYỀN</b>		<b>48</b>
<b>CHƯƠNG 5: THIẾT KẾ GIAO DIỆN HỆ THỐNG</b>		<b>50</b>
<b>1.</b>	<b>48</b>	
<b>2.</b>	<b>48</b>	
<b>3.</b>	<b>49</b>	
3.1.	Danh mục sản phẩm	52
3.2.	Danh mục kho	52
3.3.	Danh mục hóa đơn	54
3.4.	Danh mục hóa đơn ứng dụng	55
3.5.	Danh mục nhân viên	56
3.6.	Danh mục ca làm việc	57
3.7.	Danh mục bảng ca làm việc	58
3.8.	Danh mục phiếu chi	59
3.9.	Danh mục lương nhân viên	60

## LỜI CẢM ƠN

Kính thưa thầy Nguyễn Thành Sơn,

Nhóm em xin gửi lời cảm ơn chân thành đến thầy vì đã hướng dẫn và giúp đỡ chúng em trong quá trình hoàn thiện đồ án môn Hệ quản trị cơ sở dữ liệu.

Nhờ vào sự chỉ dẫn và giúp đỡ của thầy, nhóm em đã có thể hoàn thành đồ án một cách hiệu quả và đạt được kết quả như mong đợi. Chúng em rất biết ơn thầy đã dành thời gian và tâm huyết để truyền đạt kiến thức và kinh nghiệm cho nhóm.

Thầy đã giúp chúng em có được những kiến thức và kỹ năng cần thiết để làm việc với cơ sở dữ liệu một cách hiệu quả, và nhóm em sẽ sử dụng những kiến thức này để phát triển bản thân và đóng góp vào công việc sau này.

Một lần nữa, nhóm em xin chân thành cảm ơn thầy Nguyễn Thành Sơn vì sự giúp đỡ và hướng dẫn của thầy trong quá trình học tập của nhóm.

Trân trọng,

Bùi Quốc Khang,

Phạm Hùng Phong,

Đặng Minh Nhật,

Đặng Phú Quý.

## LỜI NÓI ĐẦU

Xin chào thầy và các bạn,

Đồ án quản lý quán cà phê là kết quả của nhóm em sau một thời gian học tập và thực hành môn học Hệ quản trị cơ sở dữ liệu. Đây là một đồ án đầy thử thách và hứa hẹn đem lại cho chúng ta những kiến thức và kinh nghiệm quý giá.

Trong quá trình hoàn thành đồ án, nhóm em đã đối mặt với rất nhiều khó khăn và thách thức. Tuy nhiên, với sự hướng dẫn và hỗ trợ tận tình của giáo viên hướng dẫn cùng với sự nỗ lực không ngừng của chúng em, đồ án đã được hoàn thành một cách thật sự tốt đẹp.

Đồ án quản lý quán cà phê là một ứng dụng thực tiễn, nhằm mục đích giúp chủ quán cà phê có thể quản lý hoạt động của quán một cách dễ dàng và hiệu quả hơn. Bằng việc áp dụng những kiến thức về hệ quản trị cơ sở dữ liệu, chúng em đã xây dựng được một hệ thống quản lý hoàn chỉnh, bao gồm các chức năng quản lý khách hàng, quản lý nhân viên, quản lý thực đơn và các hoạt động kinh doanh khác.

Nhóm em hy vọng rằng đồ án của chúng em sẽ đem lại cho quý thầy cô và các bạn sinh viên những kiến thức và trải nghiệm thực tế hữu ích. Chúng em cũng mong muốn rằng đồ án này sẽ có thể truyền cảm hứng và khơi gợi những ý tưởng sáng tạo cho các bạn sinh viên khác trong việc ứng dụng hệ quản trị cơ sở dữ liệu vào thực tế. Cảm ơn quý thầy cô và các bạn sinh viên đã theo dõi và đóng góp ý kiến cho đồ án của chúng em.



## **CHƯƠNG 1: TỔNG QUAN VỀ HỆ THỐNG**

### **1. Đặc tả đề tài**

#### **1.1. Mô tả bài toán**

Một chuỗi quán cà phê cần có một hệ thống để quản lý. Chương trình quản lý được xây dựng cho hai đối tượng người dùng là người quản lý và nhân viên với các quyền khác nhau. Đối với người dùng là nhân viên, họ sẽ có quyền truy cập vào các chức năng như bán hàng, xem doanh thu, xem ca làm việc. Còn với người dùng là người quản lý chi nhánh, họ sẽ có quyền truy cập vào các chức năng như quản lý kho, quản lý doanh thu, quản lý nhân viên, quản lý ca làm việc. Đối với hệ thống quản lý chuỗi quán cà phê này cần lưu trữ những dữ liệu sau:

Chuỗi quán cà phê gồm nhiều chi nhánh, các chi nhánh phân biệt với nhau bằng mã chi nhánh, mỗi chi nhánh gồm các thông tin như mã chi nhánh, tên chi nhánh, địa chỉ chi nhánh, mã người quản lý. Một nhân viên chỉ làm việc tại một chi nhánh. Một chi nhánh có nhiều nhân viên. Một nhân viên có thể quản lý một chi nhánh hoặc không quản lý chi nhánh nào.

Mỗi chi nhánh có nhiều nhân viên, một nhân viên chỉ làm một công việc, một công việc có thể có nhiều nhân viên cùng làm. Các công việc phân biệt với nhau bằng mã công việc, mỗi công việc gồm các thông tin như mã công việc, tên công việc, lương. Các nhân viên phân biệt với nhau bằng mã nhân viên, mỗi nhân viên gồm các thông tin như mã nhân viên, họ tên, ngày sinh, giới tính, địa chỉ, số điện thoại, mã công việc, thưởng, ngày tuyển dụng, mã chi nhánh.

Khách hàng đến quán được chia thành hai nhóm khách hàng, các nhóm khách hàng phân biệt với nhau bằng mã nhóm khách hàng. Một nhóm khách hàng có thể có nhiều khách hàng, một khách hàng chỉ thuộc một nhóm khách hàng. Các khách hàng phân biệt với nhau bằng mã khách hàng. Mỗi khách hàng có các thông tin như mã khách hàng, họ tên, số điện thoại, mã loại khách hàng.

Sản phẩm được làm ra từ các nguyên liệu. Một sản phẩm được làm ra từ nhiều nguyên liệu, một nguyên liệu có thể dùng trong nhiều sản phẩm. Các nguyên liệu phân biệt nhau bằng mã nguyên liệu. Mỗi nguyên liệu có các thông tin như mã nguyên liệu,

tên nguyên liệu, mã chi nhánh, số lượng. Số lượng nguyên liệu ở mỗi chi nhánh là khác nhau.

Sản phẩm được chia thành nhiều loại, các loại sản phẩm phân biệt với nhau bằng mã loại sản phẩm. Một loại sản phẩm có thể có nhiều sản phẩm, một sản phẩm chỉ thuộc một loại sản phẩm. Các sản phẩm phân biệt với nhau bằng mã sản phẩm. Mỗi sản phẩm có các thông tin như mã sản phẩm, tên sản phẩm, đơn giá, tình trạng, mã loại sản phẩm. Để quản lý nguyên liệu cần có thông tin của nhà sản xuất nguyên liệu, cũng như lên đơn đặt hàng nguyên liệu đảm bảo luôn cung cấp đủ nguyên liệu để chế biến sản phẩm. Một nhà sản xuất có thể sản xuất ra nhiều nguyên liệu, một nguyên liệu cũng có thể được sản xuất từ nhiều nhà sản xuất. Các nhà sản xuất phân biệt với nhau bằng mã nhà sản xuất. Mỗi nhà sản xuất có các thông tin như mã nhà sản xuất, tên nhà sản xuất, địa chỉ, số điện thoại. Đơn nhập hàng được phân biệt bởi mã đơn nhập hàng, mỗi đơn nhập hàng có các thông tin như mã đơn nhập hàng, ngày nhập hàng, ngày hết hạn, trị giá đơn nhập hàng, mã nhà sản xuất.

Các đơn hàng được phân biệt với nhau qua mã đơn hàng, mỗi đơn hàng gồm các thông tin như mã đơn hàng, ngày đặt hàng, mã khách hàng, mã bàn, trạng thái, trị giá đơn hàng.

Để tăng doanh thu cho quán thì còn mô hình bán qua các ứng dụng (Shopee Food, Grab Food, Baemin, ...). Các ứng dụng được phân biệt với nhau qua mã ứng dụng. Mỗi ứng dụng có các thông tin như mã ứng dụng, tên ứng dụng, chiết khấu. Và các đơn hàng này được gọi là đơn hàng qua ứng dụng (khác với các đơn hàng nói trên). Các đơn hàng qua ứng dụng này cũng phân biệt theo mã đơn hàng, mỗi đơn hàng qua ứng dụng có các thông tin như mã đơn hàng, ngày đặt hàng, mã ứng dụng, trạng thái, trị giá đơn hàng.

Các nhân viên sẽ được chia ca để làm việc. Một nhân viên có thể làm nhiều ca, một ca có thể có nhiều nhân viên. Các ca làm việc được phân biệt với nhau qua mã ca, mỗi ca có các thông tin như mã ca, ngày, giờ bắt đầu, giờ kết thúc.

Bảng thống kê tài chính sẽ là điều tất yếu để duy trì các hoạt động khác của quán bao gồm mã giao dịch, loại giao dịch (doanh thu, chi phí), ngày giao dịch, số tiền, mô tả.

## 1.2. Nghiệp vụ của bài toán

Bài toán quản lý quán cà phê là một nghiệp vụ quản lý kinh doanh trong lĩnh vực dịch vụ ăn uống. Nó liên quan tới việc quản lý nhân viên, quản lý hàng hóa, quản lý doanh thu và điều hành quán cà phê. Nhân viên là bộ mặt của quán và giao tiếp với khách hàng từ lúc bước vào cho đến khi khách ra về. Vì vậy chúng ta cần xây dựng tài liệu đào tạo nghiệp vụ cho nhân viên về quy trình tác nghiệp, văn hóa phục vụ, cách ứng xử, xử lý tình huống, ...

### - Quản lý nhân viên:

Khi tuyển dụng nhân sự, quán cần đưa ra các yêu cầu cũng như các điều kiện (ví dụ: làm ca đêm, không đi trễ quá số buổi quy định, xin nghỉ trước số ngày quy định, ...) mà nhân sự đồng ý trong suốt quá trình làm việc. Quán cần chi trả cho nhân sự mức lương hợp lý. Nhân viên đã được tuyển dụng sẽ trải qua đợt đào tạo về nghiệp vụ, quy trình ở vị trí của bản thân. Người quản lý cần theo dõi và quan sát các nhân sự của mình, chỉ ra những điểm còn thiếu sót để nhân sự khắc phục và lời khen cho những điểm tốt của nhân sự để nhân sự lấy đó làm động lực.

### - Quy trình thực hiện khi khách đến quán:

Đầu tiên, nhân viên sẽ chào khách khi khách bước vào quán. Sau đó nhân viên đứng quầy sẽ bấm tạo đơn hàng trên chương trình. Tiếp theo nhân viên chỉ vào menu hỏi khách muốn sử dụng món nào, và nếu cảm thấy khách phân vân thì có thể tư vấn cho khách về các sản phẩm. Sau khi khách chọn được món thì nhân viên tiến hành thêm các món vào đơn hàng vừa tạo. Tiếp theo, nhân viên hỏi khách có thể thành viên của quán chưa, nếu chưa thì xin thông tin của khách để tạo thẻ thành viên. Sau đó, nhân viên nhập số điện thoại của khách hàng vào để tích điểm. Cuối cùng, nhân viên xuất hóa đơn để đưa cho khách thanh toán và đưa cho khách một tấm thẻ số (tượng trưng cho số bàn đã tạo đơn trên chương trình). Khi này, nhân viên pha chế bắt đầu làm những món có trong đơn hàng của khách. Sau khi hoàn thành các món trong đơn hàng, nhân viên phục vụ đưa món ra cho khách hàng. Nhân viên cần phải kiểm tra đầy đủ các món trong đơn, vật dụng đi kèm với món rồi mới đem đến cho khách hàng.

### - Quy trình thực hiện khi có đơn hàng qua ứng dụng (Shopee Food, Grab Food, Baemin, ...):

Nhân viên quán sẽ có một chiếc Ipad đăng nhập vào các ứng dụng. Khi có đơn hàng được đặt qua các ứng dụng, Ipad sẽ nhận được thông báo của các đơn hàng. Nhân viên sẽ bấm tạo đơn ở mục đơn hàng qua ứng dụng trên chương trình, thêm các món có trong đơn hàng qua ứng dụng. Sau đó, nhân viên xuất hóa đơn và đưa cho người giao hàng của ứng dụng đó.

- Quản lý hàng hóa, quản lý doanh thu và điều hành quán cà phê chia thành các giai đoạn:

Ở giai đoạn đầu, quán phải tìm được đối tác, nhà sản xuất cung cấp hàng hóa uy tín từ đầu vì có chính sách giá cả, giao hàng, chứng từ rõ ràng để tiện cho việc kiểm soát. Quán cần thống kê hàng hóa nhập vào và có kho hàng để lưu trữ, bảo quản và kiểm soát hàng hóa. Quán cần kiểm tra hàng hóa theo hóa đơn nhập hàng và nhập vào chương trình quản lý quán cà phê.

Tiếp theo, ở giai đoạn vận hành, chúng ta cần xây dựng chi phí của từng sản phẩm và lên giá thành cho các sản phẩm đó (ví dụ: 1kg cà phê chế biến được 45 ly). Ngoài ra, chúng ta cũng cần xây dựng quy trình làm việc, nghiệp vụ cho các nhân viên thu ngân, pha chế, phục vụ, bảo vệ.

Cuối cùng, ở giai đoạn kiểm soát, chúng ta cần kiểm soát về nghiệp vụ của các nhân viên, dựa trên những đánh giá của khách hàng về chất lượng phục vụ, trích xuất camera các ca làm việc để quan sát và đưa ra đánh giá về từng nhân sự. Ngoài ra, quán cần phải kiểm soát được thu chi hàng ngày. Nhân viên thu ngân sẽ được phân quyền để xem bảng doanh thu theo ngày để tổng kết ca. Nhân viên sẽ kiểm tra xem số tiền thu được có đúng với doanh thu của ca đó không rồi mới giao ca cho nhân viên khác. Người quản lý sẽ được phân quyền để chi tiền cho số tiền nhập hàng. Kho hàng của quán cần được kiểm soát chặt chẽ. Kho hàng cần được kiểm tra định kỳ hàng tuần để đối chiếu với số liệu thực tế ghi nhận ở chương trình quản lý quán. Người quản lý cần phải nắm các bảng báo cáo phân tích như báo cáo doanh thu (theo ngày, tháng, năm), báo cáo chi phí (chi tiết các khoản chi), báo cáo thất thoát (bị đổ vỡ, hư hỏng, mất mát). Chủ của chuỗi quán cà phê sẽ dựa trên các báo cáo để nắm rõ doanh thu, chi phí, lương nhân sự, ... để có thể đưa ra điều chỉnh sao cho có lợi nhất.

### 1.3. Mô tả chức năng của bài toán

Đối với nhân viên:

**Chức năng bán hàng:** cho phép nhân viên chọn số bàn, tạo đơn hàng cho bàn, thêm món, xóa món, xuất hóa đơn thanh toán.

**Quản lý khách hàng:** cho phép nhân viên tạo thẻ thành viên cho khách hàng, nhập số điện thoại ở bước thanh toán để tích điểm cho khách hàng.

**Xem doanh thu:** cho phép nhân viên có thể xem được doanh thu bán trong ca làm việc của mình.

**Xem bảng ca làm việc:** cho phép nhân viên xem bảng phân ca làm việc trong tuần.

Đối với người quản lý:

**Quản lý kho:** cho phép người quản lý biết được số lượng của từng nguyên liệu còn trong kho hàng, ngày nhập hàng của nguyên liệu, hạn sử dụng.

**Quản lý nhân viên:** cho phép người quản lý nhập, thêm, xóa, sửa thông tin của nhân viên. Người quản lý có thể xem các thông tin để liên lạc được với nhân viên như số điện thoại.

**Quản lý ca làm việc của nhân viên:** Các nhân viên sẽ đăng ký ca với người quản lý và người quản lý sẽ thêm ca làm việc vào bảng phân ca cho các nhân viên. Người quản lý có quyền thêm, xóa, sửa ca làm việc của các nhân viên.

**Quản lý doanh thu:** cho phép quán cà phê tính toán doanh thu theo ngày, tuần, tháng hoặc năm, dựa trên thông tin về đơn hàng và giao dịch tài chính.

**Quản lý tài chính:** cho người quản lý tạo ra các giao dịch tài chính, bao gồm mã giao dịch, loại giao dịch (doanh thu hoặc chi phí), ngày giao dịch, số tiền giao dịch và mô tả giao dịch.

#### 1.4. Tính lương nhân viên:

Tính lương nhân viên trong quán cà phê có thể bao gồm các yếu tố sau:

1. **Lương cơ bản:** đây là số tiền nhân viên được trả cho việc làm công việc cố định mà họ được giao. Đây là số tiền cố định mà nhân viên được trả hàng tháng (tính theo giờ).
2. **Thời gian làm việc:** đây là số giờ mà nhân viên đã làm việc trong tháng.
3. **Lương theo giờ:** được tính theo công thức: Lương cơ bản x giờ làm việc.
4. **Tiền thưởng:** đây là khoản tiền thưởng mà nhân viên được trả cho các ca làm việc vào ngày lễ, những ngày có nhân viên xin nghỉ đột xuất (mỗi ca làm việc này nhân viên sẽ được cộng một đơn vị vào ca thưởng) hoặc doanh thu quán đạt được chỉ tiêu nhất định (ở đây chỉ tiêu của quán đề ra sẽ được thưởng thêm 5% doanh thu). Công thức tính tiền thưởng được tính như sau: tiền thưởng = doanh thu x 5% + ca thưởng x 4 giờ x 2 x lương cơ bản (quán quy định mỗi ca làm là 4 giờ).

**Tổng lương = Lương theo giờ + Tiền thưởng**

VD:

Giả sử quán cà phê có 2 loại nhân viên: nhân viên thu ngân và nhân viên phục vụ. Cả 2 loại nhân viên đều được trả lương theo giờ làm việc.

Lương nhân viên thu ngân: 25.000 đồng/giờ

Lương nhân viên phục vụ: 23.000 đồng/giờ

Vào tháng này, nhân viên thu ngân A đã làm việc 41 ca và 5 ca thưởng, còn nhân viên phục vụ B đã làm việc 32 ca và 3 ca thưởng.

**Do đó, lương của nhân viên thu ngân A sẽ là:**

Lương theo giờ = Số giờ làm việc x Lương/giờ

= 41 ca x 4 giờ x 25.000 đồng/giờ

= 4.100.000 đồng

Tiền thưởng = ca thưởng x 4 giờ x 2 x Lương cơ bản

= 5 ca thưởng x 4 giờ x 2 x 25.000 đồng/giờ

= 1.000.000 đồng

Tổng lương = Lương theo giờ + Tiền thưởng = 4.100.000 + 1.000.000 = 5.100.000 đồng

**Tương tự, lương của nhân viên phục vụ B sẽ là:**

$$\begin{aligned}\text{Lương theo giờ} &= \text{Số giờ làm việc} \times \text{Lương/giờ} \\ &= 32 \text{ ca} \times 4 \text{ giờ} \times 23.000 \text{ đồng/giờ} \\ &= 2.944.000 \text{ đồng}\end{aligned}$$

$$\begin{aligned}\text{Tiền thưởng} &= \text{ca thưởng} \times 4 \text{ giờ} \times \text{Lương cơ bản} \times 2 \\ &= 3 \text{ ca} \times 4 \text{ giờ} \times 23.000 \text{ đồng/giờ} \\ &= 276.000 \text{ đồng}\end{aligned}$$

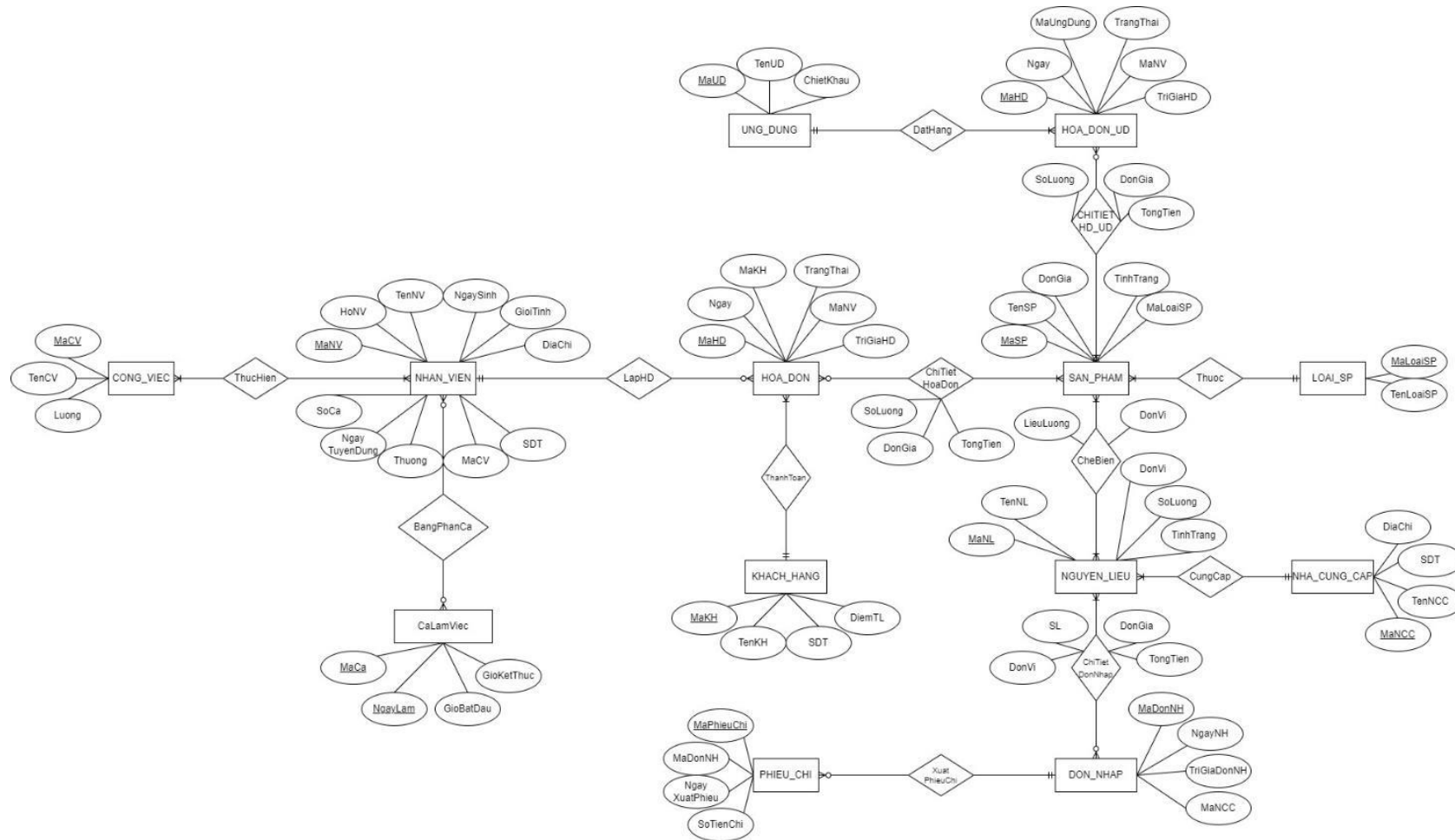
$$\text{Tổng lương} = \text{Lương theo giờ} + \text{Tiền thưởng} = 2.944.000 + 276.000 = 3.220.000 \text{ đồng}$$

**Trong trường hợp doanh thu quán đạt chỉ tiêu thì nhân viên trong quán được thưởng thêm 5% doanh thu của quán.**

## CHƯƠNG 2: PHÂN TÍCH THIẾT KẾ HỆ THỐNG

### 1. Thiết kế cơ sở dữ liệu mức quan niệm

Từ mô tả về dữ liệu cần có ở phần mô tả của bài toán ta hình thành được sơ đồ thực thể kết hợp (ERD).





## 2. Thiết kế cơ sở dữ liệu mức logic

Từ sơ đồ thực thể kết hợp (ERD), ta có các lược đồ quan hệ:

- CongViec(MaCV, TenCV, Luong)
- NhanVien(MaNV, HoNV, TenNV, NgaySinh, GioiTinh, DiaChi, NgayTuyenDung, SoCa, Thuong, MaCV, SDT)
- BangPhanCa(MaCa, MaNV, NgayLam)
- CaLamViec(MaCa, NgayLam, GioBatDau, GioKetThuc)
- HoaDon(MaHD, Ngay, MaKH, MaNV, TrangThai, TriGiaHD)
- ChiTietHoaDon(MaHD, MaSP, SoLuong, TongTien, DonGia)
- KhachHang(MaKH, TenKH, SDT, DiemTL)
- SanPham(MaSP, TenSP, DonGia, TinhTrang, MaLoaiSP)
- LoaiSanPham(MaLoaiSP, TenLoaiSP)
- ChiTietHoaDonUngDung (MaHD UD, MaSP, SoLuong, DonGia, TongTien)
- HoaDonUngDung (MaHD, Ngay, MaUngDung, TrangThai, MaNV, TriGiaHD)
- UngDung(MaUngDung, TenUngDung, ChietKhauPhanTram)
- CheBien(MaSP, MaNL, LieuLuong, DonVi)
- NguyenLieu(MaNL, TenNL, MaNCC, SoLuong, DonVi, TinhTrang)
- ChiTietDonNhapHang(MaNL, MaDonNH, DonGia, SL, TongTien)
- NhaCungCap(MaNCC, TenNCC, DiaChi, SDT)
- DonNhapHang(MaDonNH, NgayNH, TriGiaDonNH, MaNCC)
- PhieuChi(MaPhieuChi, MaDonNH, NgayXuatPhieu, SoTienChi)

### 3. Các ràng buộc cần có

S T T	BẢNG	RÀNG BUỘC
1	CongViec	Khóa chính MaCV, CHECK Luong > 0
2	NhanVien	Khóa chính MaNV, Khóa ngoại MaCV đến bảng CongViec, NOT NULL HoNV, NOT NULL TenNV, NOT NULL SDT, CHECK NgaySinh (>=18 tuổi), CHECK SDT (=10), CHECK NgayTuyenDung (>0)
3	KhachHang	Khoá chính MaKH, NOT NULL TenKH, NOT NULL SDT, CHECK SDT (=10)
4	NhaCungCap	Khoá chính MaNCC, NOT NULL TenNCC, NOT NULL SDT, CHECK SDT(=10)
5	DonNhapHang	Khoá chính MaDonNH, Khoá ngoại MaNCC đến bảng NhaCungCap.  CHECK NgayNH (>=0), NOT NULL TriGiaDonNH
6	NguyenLieu	Khoá chính MaNL, Khoá ngoại MaNCC đến bảng NhaCungCap,  NOT NULL TenNL, CHECK SL(>0),
7	ChiTietNhapHang	Khóa chính (MaDonNH, MaNL), Khóa ngoại MaDonNH đến bảng DonNhapHang, Khóa ngoại MaNL đến bảng NguyenLieu,  CHECK SL(>0),
8	LoaiSanPham	Khoá chính MaLoaiSP, NOT NULL TenLoaiSP
9	SanPham	Khoá chính MaSP, Khoá ngoại MaLoaiSP đến bảng LoaiSanPham,  NOT NULL TenSP, NOT NULL DonGia
10	UngDung	Khoá chính MaUngDung,  NOT NULL TenUngDung, NOT NULL ChietKhauPhanTram

11	HoaDon	Khoá chính MaHD, Khoá ngoại MaKH đến bảng KháchHang, Khoá ngoại MaNV đến bảng NhanVien, CHECK NgayDatHang ( $\geq 0$ ), NOT NULL TriGiaHD
12	ChiTietHoaDon	Khoá chính (MaHD, MaSP), Khoá ngoại MaHD đến bảng HoaDon, Khoá ngoại MaSP đến bảng SanPham, CHECK SL( $>0$ )
13	HoaDonUngDung	Khoá chính MaHD_UD, Khoá ngoại MaUngDung đến bảng UngDung, Khoá ngoại MaNV đến bảng MaNV, CHECK NgayDatHang ( $\geq 0$ ), NOT NULL TriGiaDH
14	ChiTietHoaDonUngDung	Khóa chính (MaHD_UD, MaSP), Khóa ngoại MaHD_UD đến bảng HoaDonUngDung, Khóa ngoại MaSP đến bảng SanPham, CHECK SL( $>0$ )
15	CaLamViec	Khóa chính (MaCa, NgayLam), CHECK NgayLam( $\geq 0$ )
16	BangPhanCa	Khóa chính (MaCa, NgayLam, MaNV), Khóa ngoại (MaCa, NgayLam) đến bảng CaLamViec, Khóa ngoại MaNV đến bảng NhanVien
17	PhieuChi	Khóa chính MaPhieuChi, Khóa ngoại MaDonNH đến bảng DonNhapHang, CHECK NgayXuatPhieu( $\geq 0$ )
18	CheBien	Khóa chính (MaSP, MaNL), Khóa ngoại MaSP đến bảng SP, Khóa ngoại MaNL đến bảng NguyenLieu

#### 4. Cài đặt các CSDL và các ràng buộc

##### *Bảng công việc*

```
CREATE TABLE CongViec(  
    MaCV nchar(10) CONSTRAINT PK_CongViec PRIMARY KEY,  
    TenCV nvarchar(50) NOT NULL,  
    Luong float check (Luong > 0)  
)
```

##### *Bảng nhân viên*

```
CREATE TABLE NhanVien(  
    MaNV nchar(10) CONSTRAINT PK_NhanVien PRIMARY KEY,  
    HoNV nvarchar(10) NOT NULL,  
    TenNV nvarchar(10) NOT NULL,  
    NgaySinh date check (DATEDIFF(year, NgaySinh, GETDATE())>=18),  
    GioiTinh nvarchar(3),  
    DiaChi nvarchar(100),  
    SDT nchar(11) check (len(SDT)=10),  
    MaCV nchar(10) CONSTRAINT FK_NhanVien_CV FOREIGN KEY REFERENCES  
    CongViec(MaCV),  
    SoCa int,  
    Thuong int,  
    NgayTuyenDung date check (DATEDIFF(day, NgayTuyenDung, GETDATE())>=0)  
)
```

##### *Bảng khách hàng*

```
CREATE TABLE KhachHang(  
    MaKH nchar(10) CONSTRAINT PK_KhachHang PRIMARY KEY,  
    TenKH nvarchar(50) NOT NULL,  
    SDT nchar(11) NOT NULL check (len(SDT)=10),  
    DiemTL int  
)
```

##### *Bảng nhà cung cấp*

```
CREATE TABLE NhaCungCap(  
    MaNCC nchar(10) CONSTRAINT PK_NhaCungCap PRIMARY KEY,  
    TenNCC nvarchar(50) NOT NULL,  
    DiaChi nchar(100),  
    SDT nchar(10) NOT NULL check (len(SDT)=10)  
)
```

##### *Bảng đơn nhập hàng*

```
CREATE TABLE DonNhapHang(  
    MaDonNH nchar(10) CONSTRAINT PK_DonNhapHang PRIMARY KEY,  
    NgayNH date check (DATEDIFF(day, NgayNH, GETDATE())>=0),  
    TriGiaDonNH float NOT NULL,  
    MaNCC nchar(10) CONSTRAINT FK_DonNhapHang_NCC FOREIGN KEY REFERENCES  
    NhaCungCap(MaNCC),  
    TinhTrang nvarchar(50)  
)
```

### ***Bảng nguyên liệu***

```
CREATE TABLE NguyenLieu(  
    MaNL nchar(10) CONSTRAINT PK_NguyenLieu PRIMARY KEY,  
    TenNL nvarchar(50) NOT NULL,  
    MaNCC nchar(10) CONSTRAINT FK_NguyenLieu_NCC FOREIGN KEY REFERENCES  
    NhaCungCap(MaNCC),  
    SL int check (SL>0),  
    DonVi nchar(10),  
    TinhTrang nchar(10)  
)
```

### ***Bảng chi tiết nhập hàng***

```
CREATE TABLE ChiTietNhapHang(  
    MaDonNH nchar(10) CONSTRAINT FK_ChiTietNH_DonNH FOREIGN KEY REFERENCES  
    DonNhapHang(MaDonNH),  
    MaNL nchar(10) CONSTRAINT FK_ChiTietNH_NL FOREIGN KEY REFERENCES  
    NguyenLieu(MaNL),  
    DonGia float,  
    SL int check (SL>0),  
    DonVi nchar(10),  
    TongTien float  
    CONSTRAINT PK_ChiTietNhapHang PRIMARY KEY (MaDonNH, MaNL)  
)
```

### ***Bảng loại sản phẩm***

```
CREATE TABLE LoaiSanPham(  
    MaLoaiSP nchar(10) CONSTRAINT PK_LoaiSanPham PRIMARY KEY,  
    TenLoaiSP nvarchar(50) NOT NULL  
)
```

### ***Bảng sản phẩm***

```
CREATE TABLE SanPham(  
    MaSP nchar(10) CONSTRAINT PK_MaSP PRIMARY KEY,  
    TenSP nvarchar(50) NOT NULL,  
    DonGia float NOT NULL,  
    TinhTrang nchar(10) DEFAULT N'Hết hàng',  
    MaLoaiSP nchar(10) CONSTRAINT FK_SanPham_LoaiSP FOREIGN KEY REFERENCES  
    LoaiSanPham  
)
```

### ***Bảng ứng dụng***

```
CREATE TABLE UngDung(  
    MaUngDung nchar(10) CONSTRAINT PK_UngDung PRIMARY KEY,  
    TenUngDung nvarchar(50) NOT NULL,  
    ChietKhauPhanTram int NOT NULL  
)
```

### ***Bảng hoá đơn***

```
CREATE TABLE HoaDon(  
    MaHD nchar(10) CONSTRAINT PK_HoaDon PRIMARY KEY,  
    NgayDatHang date check (DATEDIFF(year, NgayDatHang, GETDATE())>=0),  
    MaKH nchar(10) CONSTRAINT FK_HoaDon_KH FOREIGN KEY REFERENCES  
    KhachHang(MaKH),  
    MaNV nchar(10) CONSTRAINT FK_HoaDon_NV FOREIGN KEY REFERENCES NhanVien(MaNV),  
    TriGiaHD int NOT NULL  
)
```

### ***Bảng chi tiết hoá đơn***

```
CREATE TABLE ChiTietHoaDon(  
    MaHD nchar(10) CONSTRAINT FK_ChiTietHD_HD FOREIGN KEY REFERENCES  
    HoaDon(MaHD),  
    MaSP nchar(10) CONSTRAINT FK_ChiTietHD_SP FOREIGN KEY REFERENCES  
    SanPham(MaSP),  
    SL int check (SL > 0),  
    DonGia float,  
    TongTien float  
    CONSTRAINT PK_ChiTietHD PRIMARY KEY (MaHD, MaSP)  
)
```

### ***Bảng hóa đơn ứng dụng***

```
CREATE TABLE HoaDonUngDung(  
    MaHD_UD nchar(10) CONSTRAINT PK_HoaDonUngDung PRIMARY KEY,  
    NgayDatHang date check (DATEDIFF(day, NgayDatHang, GETDATE())>=0),  
    MaUngDung nchar(10) CONSTRAINT FK_HoaDonUngDung_UD FOREIGN KEY REFERENCES  
    UngDung(MaUngDung),  
    MaNV nchar(10) CONSTRAINT FK_HoaDonUngDung_NV FOREIGN KEY REFERENCES  
    NhanVien(MaNV),  
    TriGiaDH int NOT NULL  
)
```

### ***Bảng chi tiết hoá đơn qua ứng dụng***

```
CREATE TABLE ChiTietHoaDonUngDung(  
    MaHD_UD nchar(10) CONSTRAINT FK_ChiTietHDUD_HDUD FOREIGN KEY REFERENCES  
    HoaDonUngDung(MaHD_UD),  
    MaSP nchar(10) CONSTRAINT FK_ChiTietHDUD_SP FOREIGN KEY REFERENCES  
    SanPham(MaSP),  
    SL int check (SL>0),  
    DonGia float,  
    TongTien float  
    CONSTRAINT PK_ChiTietHDUD PRIMARY KEY (MaHD_UD, MaSP)  
)
```

### ***Bảng các ca làm việc***

```
CREATE TABLE CaLamViec(  
    MaCa nchar(10),  
    NgayLam date check (DATEDIFF(day, NgayLam, GETDATE())>=0),  
    GioBatDau nchar(10),  
    GioKetThuc nchar(10),  
    CONSTRAINT PK_CaLamViec PRIMARY KEY (MaCa, NgayLam)  
)
```

### ***Bảng phân ca làm cho nhân viên***

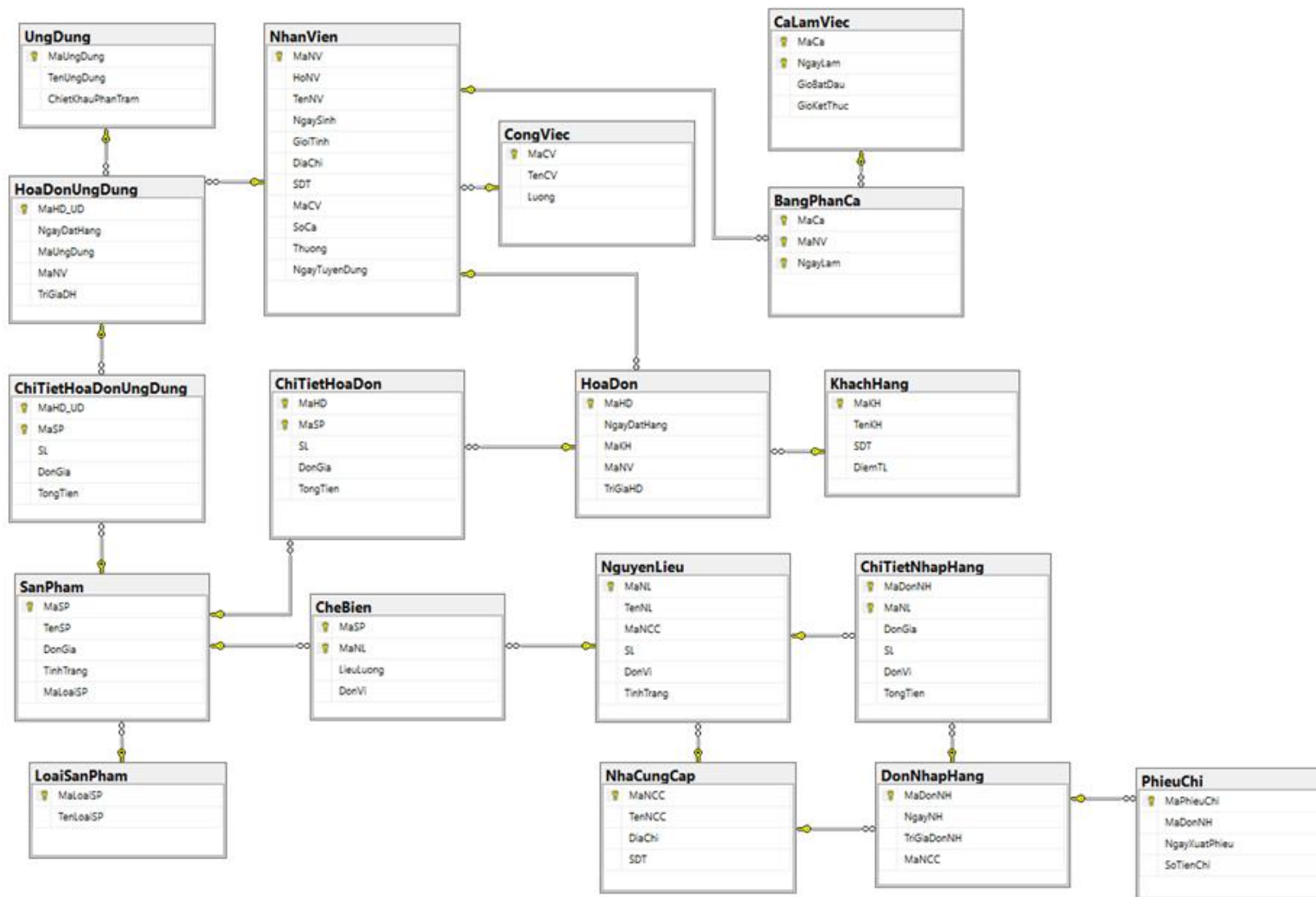
```
CREATE TABLE BangPhanCa(  
    MaCa nchar(10),  
    MaNV nchar(10) CONSTRAINT FK_PhanCa_NV FOREIGN KEY REFERENCES NhanVien(MaNV),  
    NgayLam date check (DATEDIFF(day, NgayLam, GETDATE())>=0),  
    CONSTRAINT PK_BangPhanCa PRIMARY KEY (MaCa, MaNV, NgayLam),  
    CONSTRAINT FK_PhanCa_Ca FOREIGN KEY (MaCa, NgayLam) REFERENCES CaLamViec(MaCa,  
    NgayLam)  
)
```

### ***Bảng phiếu chi***

```
CREATE TABLE PhieuChi(  
    MaPhieuChi nchar(10) CONSTRAINT PK_PHIEUCHI PRIMARY KEY,  
    MaDonNH nchar(10) CONSTRAINT FK_PhieuChi_DonNH FOREIGN KEY REFERENCES  
    DonNhapHang(MaDonNH),  
    NgayXuatPhieu date check (DATEDIFF(day, NgayXuatPhieu, GETDATE())>=0),  
    SoTienChi float  
)
```

### ***Bảng chế biến***

```
CREATE TABLE CheBien(  
    MaSP nchar(10) CONSTRAINT FK_CheBien_SP FOREIGN KEY REFERENCES SanPham(MaSP),  
    MaNL nchar(10) CONSTRAINT FK_CheBien_NL FOREIGN KEY REFERENCES  
    NguyenLieu(MaNL),  
    LieuLuong int,  
    DonVi nchar(10)  
    CONSTRAINT PK_CheBien PRIMARY KEY (MaSP, MaNL)  
)
```





## 5. Các view

### a. Xem ca làm việc của nhân viên trong ngày

```
CREATE VIEW V_CaLamTrongNgay AS
SELECT clv.*, nv.TenNV, nv.MaCV, cv.TenCV
FROM dbo.BangPhanCa pc
      JOIN dbo.NhanVien nv ON pc.MaNV = nv.MaNV
      JOIN dbo.CaLamViec clv ON pc.MaCa = clv.MaCa
      JOIN dbo.CongViec cv ON nv.MaCV = cv.MaCV
WHERE clv.NgayLam = CONVERT(DATE, GETDATE())
```

### b. Xem danh sách sản phẩm

```
--View danh sách sản phẩm
CREATE VIEW V_DanhSachSanPham AS
SELECT MaSP, TenSP, DonGia, TinhTrang
FROM dbo.SanPham
GO
```

### c. Xem số lượng sản phẩm đã bán trong ngày

```
CREATE VIEW V_SoLuongSanPhamDaBanTrongNgay AS
SELECT sp.MaSP, SUM(cthd.SL) AS SoLuongDaBanTaiQuay, SUM(cthdUD.SL)
AS SoLuongDaBanQuaUD
FROM dbo.SanPham sp
      JOIN dbo.ChiTietHoaDon cthd ON sp.MaSP = cthd.MaSP
      JOIN dbo.HoaDon hd ON cthd.maHD = hd.MaHD
      JOIN dbo.ChiTietHoaDonUngDung cthdUD ON sp.MaSP = cthdUD.MaSP
      JOIN dbo.HoaDonUngDung hdUD ON cthdUD.MaHD_UD = hdud.MaHD_UD
WHERE hd.NgayDatHang = CONVERT(DATE, GETDATE()) AND hdud.NgayDatHang = CONVERT(DATE,
GETDATE())
GROUP BY sp.MaSP
```

### d. Xem thông tin nhân viên

```
CREATE VIEW V_ThongTinNhanVien AS
SELECT nv.TenNV, nv.NgaySinh, nv.SDT, nv.SoCa, nv.NgayTuyenDung, nv.Thuong,
cv.TenCV, cv.Luong
FROM dbo.NhanVien nv JOIN dbo.CongViec cv ON nv.MaCV = cv.MaCV
```

### e. Xem phiếu chi trong ngày

```
CREATE VIEW V_PhiếuChiTrongNgay AS
SELECT *
FROM dbo.PhiếuChi pc
WHERE pc.NgayXuatPhiếu = CONVERT(DATE, GETDATE())
```

### f. Xem đơn nhập hàng chưa giao

```
CREATE VIEW vi_DonNhapHang AS
SELECT *
FROM dbo.DonNhapHang dnh
WHERE TinhTrang = N'Chưa giao'
```

## 6. Các Trigger

### a. Trigger đặt trạng thái hoá đơn

```
CREATE TRIGGER set_HoaDon_TrangThai
ON HoaDon
FOR INSERT
AS
BEGIN
    UPDATE HoaDon
    SET TrangThai = 'Chưa thanh toán'
    WHERE MaHD IN (SELECT MaHD FROM INSERTED)
END;
```

### b. Trigger thay đổi số lượng của nguyên liệu mỗi khi bán được sản phẩm (được tính theo công thức).

```
CREATE TRIGGER TG_CheBienSanPham
ON ChiTietHoaDon
AFTER Insert
AS
    Begin
        -- Xử lý mã NL CF
        Declare @newMaSP nchar(10), @newSL int
        Select @newMaSP=ne.MaSP, @newSL=ne.SL
        From inserted ne
        IF (@newMaSP = 'SP01' OR @newMaSP = 'SP02' OR @newMaSP = 'SP03')
            Begin
                UPDATE NguyenLieu
                SET SL = (SL*1000 - @newSL*(select TOP 1 LieuLuong
                    from CheBien
                    where MaSP=@newMaSP))/1000
                WHERE MaNL = 'CF'
            End
        -- Xử lý mã NL DG
        IF (@newMaSP = 'SP01' OR @newMaSP = 'SP02' OR @newMaSP = 'SP03' OR
@newMaSP = 'SP04' OR @newMaSP = 'SP05' OR @newMaSP = 'SP06' OR @newMaSP = 'SP07' OR
@newMaSP = 'SP08' OR @newMaSP = 'SP09' OR @newMaSP = 'SP10' OR @newMaSP = 'SP11')
            Begin
                UPDATE NguyenLieu
                SET SL = (SL*1000 - @newSL*(select TOP 1 LieuLuong
                    from CheBien
                    where MaSP=@newMaSP))/1000
                WHERE MaNL = 'DG'
            End
        -- Xử lý mã NL SD
        IF (@newMaSP = 'SP02')
            Begin
                UPDATE NguyenLieu
                SET SL = (SL*1000 - @newSL*(select TOP 1 LieuLuong
                    from CheBien
                    where MaSP=@newMaSP))/1000
                WHERE MaNL = 'SD'
            End
        -- Xử lý mã NL ST
        IF (@newMaSP = 'SP03')
            Begin
                UPDATE NguyenLieu
                SET SL = (SL*1000 - @newSL*(select TOP 1 LieuLuong
                    from CheBien
```

```

                                where MaSP=@newMaSP))/1000
                                WHERE MaNL = 'ST'
                                End
                                -- Xử lý mã NL SRD
                                IF (@newMaSP = 'SP04' OR @newMaSP = 'SP08')
                                Begin
                                UPDATE NguyenLieu
                                SET SL = (SL*1000 - @newSL*(select TOP 1 LieuLuong
                                                                from CheBien
                                                                where MaSP=@newMaSP))/1000

                                WHERE MaNL = 'SRD'
                                End
                                -- Xử lý mã NL SRD
                                IF (@newMaSP = 'SP07' OR @newMaSP = 'SP11')
                                Begin
                                UPDATE NguyenLieu
                                SET SL = (SL*1000 - @newSL*(select TOP 1 LieuLuong
                                                                from CheBien
                                                                where MaSP=@newMaSP))/1000

                                WHERE MaNL = 'SRT'
                                End
                                -- Xử lý mã NL SRD
                                IF (@newMaSP = 'SP05' OR @newMaSP = 'SP09')
                                Begin
                                UPDATE NguyenLieu
                                SET SL = (SL*1000 - @newSL*(select TOP 1 LieuLuong
                                                                from CheBien
                                                                where MaSP=@newMaSP))/1000

                                WHERE MaNL = 'SRV'
                                End
                                -- Xử lý mã NL SRD
                                IF (@newMaSP = 'SP06' OR @newMaSP = 'SP10')
                                Begin
                                UPDATE NguyenLieu
                                SET SL = (SL*1000 - @newSL*(select TOP 1 LieuLuong
                                                                from CheBien
                                                                where MaSP=@newMaSP))/1000

                                WHERE MaNL = 'SRVQ'
                                End
                                End
                                End

```

**c. Kiểm tra xem nguyên liệu trong kho còn đủ để đáp ứng số lượng sản phẩm khách hàng yêu cầu hay không**

```

CREATER TRIGGER TG_SPHetHang
ON ChiTietHoaDon
AFTER INSERT
AS
    BEGIN
        Declare @newMaSP nchar(10), @newSL int, @MaNLMin nchar(10), @SLMin
int, @LieuLuongCan float
        Select @newMaSP = ne.MaSP, @newSL = ne.SL
        From inserted ne
        --
        Select @SLMin=min(NguyenLieu.SL)
        From NguyenLieu, CheBien
        Where NguyenLieu.MaNL = CheBien.MaNL
        and CheBien.MaSP = @newMaSP
        --
    END

```

```

        Select TOP 1 @MaNLMin = NguyenLieu.MaNL, @LieuLuongCan =
CheBien.LieuLuong
        From NguyenLieu, CheBien
        Where NguyenLieu.MaNL = CheBien.MaNL
        and CheBien.MaSP = @newMaSP
        and NguyenLieu.SL = @SLMin
        --
        IF (@newSL*@LieuLuongCan - @SLMin*1000 > 0)
            BEGIN
                UPDATE SanPham
                SET TinhTrang = 'Hết hàng'
                WHERE MaSP=@newMaSP
                ROLLBACK
            END
        END
END

```

#### d. Kiểm tra trùng lặp số điện thoại của khách hàng

```

CREATE TRIGGER TG_TrungSDT
ON dbo.KhachHang
AFTER INSERT, UPDATE
AS
BEGIN
    -- Kiểm tra số điện thoại vừa thêm có bị trùng lặp
    IF EXISTS (
        SELECT *
        FROM inserted i
        WHERE EXISTS (
            SELECT *
            FROM dbo.KhachHang k
            WHERE k.SDT = i.SDT AND k.MaKH <> i.MaKH
        )
    )
    BEGIN
        -- Nếu trùng thì rollback
        --PRINT N'Số điện thoại đã tồn tại'
        ROLLBACK;
    END
END

```

#### e. Kiểm tra tên sản phẩm có bị trùng không

```

CREATE TRIGGER TG_TrungTenSP
ON dbo.SanPham
AFTER INSERT, UPDATE
AS
BEGIN
    -- Kiểm tra tên sản phẩm vừa thêm có bị trùng lặp
    IF EXISTS (
        SELECT *
        FROM inserted i
        WHERE EXISTS (
            SELECT *
            FROM dbo.SanPham sp
            WHERE sp.TenSP = i.TenSP AND sp.MaSP <> i.MaSP
        )
    )
    BEGIN
        -- Nếu trùng thì rollback
        RAISERROR ('Tên sản phẩm bị trùng', 16, 1)
    END
END

```

```
ROLLBACK;  
END  
END
```

## CHƯƠNG 3: THIẾT KẾ CÁC CHỨC NĂNG

### 1. Kết nối cơ sở dữ liệu

```
public class MY_DB
{
    SqlConnection con = new SqlConnection(@"Data Source=LAPTOP-TSVFN4HJ;Initial
Catalog=QUANLYQUANCOFFEE_Cur;User Id="+ GLOBAL.username + ";Password=" +
GLOBAL.password + ";");

    public SqlConnection getConnection
    {
        get
        {
            return con;
        }
    }
    SqlConnection conAdmin = new SqlConnection(@"Data Source=LAPTOP-TSVFN4HJ;Initial
Catalog=QUANLYQUANCOFFEE_Cur;Integrated Security=True");
    public SqlConnection getConnectionAdmin
    {
        get
        {
            return conAdmin;
        }
    }

    // open the connection
    public void openConnection()
    {
        if (con.State == ConnectionState.Closed)
        {
            con.Open();
        }
    }

    public void openConnectionAdmin()
    {
        if (conAdmin.State == ConnectionState.Closed)
        {
            conAdmin.Open();
        }
    }

    public void closeConnection()
    {
        if (con.State == ConnectionState.Open)
        {
            con.Close();
        }
    }

    public void closeConnectionAdmin()
    {
        if (conAdmin.State == ConnectionState.Open)
        {
            conAdmin.Close();
        }
    }
}
```

### 2. Xem thông tin danh mục

## 2.1. Danh mục nhân viên

```
CREATE VIEW [dbo].[v_LoadEmployee] AS
SELECT MaNV, HoNV, TenNV, NgaySinh, GioiTinh, DiaChi, SDT, CongViec.TenCV, SoCa,
Thuong, NgayTuyenDung, HinhAnh
FROM NhanVien, CongViec
WHERE NhanVien.MaCV = CongViec.MaCV
```

Thực thi trên C#:

```
public DataTable getAllEmployee()
{
    SqlCommand cmd = new SqlCommand("SELECT * FROM v_LoadEmployee", db.getConnection());
    SqlDataAdapter adapter = new SqlDataAdapter(cmd);
    DataTable dataTable = new DataTable();
    adapter.Fill(dataTable);
    return dataTable;
}
```

## 2.2. Danh mục bảng phân ca

```
CREATE VIEW [dbo].[v_BangPhanCa]
AS
SELECT NhanVien.MaNV, NhanVien.HoNV, NhanVien.TenNV, BangPhanCa.MaCa,
BangPhanCa.NgayTrongTuan
FROM BangPhanCa, NhanVien
WHERE BangPhanCa.MaNV = NhanVien.MaNV
```

Thực thi trên C#:

```
private void displayDivision()
{
    SqlCommand cmd = new SqlCommand("SELECT * FROM v_BangPhanCa", db.getConnection());
    SqlDataAdapter adapter = new SqlDataAdapter(cmd);
    DataTable dataTable = new DataTable();
    adapter.Fill(dataTable);
    current_DataGridView = 2;
    dataGridView_Division.DataSource = null;
    dataGridView_Division.DataSource = dataTable;
}
```

## 2.3. Danh mục ca làm việc

```
CREATE VIEW [dbo].[v_CaLamViec]
AS
SELECT *
FROM CaLamViec
```

Thực thi trên C#:

```
private void displayShift()
{
    SqlCommand cmd = new SqlCommand("SELECT * FROM v_CaLamViec", db.getConnection());
    SqlDataAdapter adapter = new SqlDataAdapter(cmd);
    DataTable dataTable = new DataTable();
    adapter.Fill(dataTable);
    current_DataGridView = 1;
    dataGridView_Division.DataSource = null;
    dataGridView_Division.DataSource = dataTable;
}
```

## 2.4. Danh mục hóa đơn

```
CREATE VIEW [dbo].[vi_HoaDonView] AS
SELECT MaHD, NgayDatHang, MaKH, MaNV, TriGiaHD
FROM HoaDon;
```

Thực thi trên C#:

```
public void load_tabPage_Receipt()
{
    SqlCommand cmd = new SqlCommand("SELECT * FROM vi_HoaDonView", db.getConnection());
}
```

```

SqlDataAdapter da = new SqlDataAdapter(cmd);
DataTable dt = new DataTable();
da.Fill(dt);
dataGridView3.DataSource = dt;

//Đổi tên
dataGridView3.Columns["MaHD"].HeaderText = "Mã hoá đơn";
dataGridView3.Columns["NgayDatHang"].HeaderText = "Ngày đặt hàng";
dataGridView3.Columns["MaKH"].HeaderText = "Mã khách hàng";
dataGridView3.Columns["MaNV"].HeaderText = "Mã nhân viên";
dataGridView3.Columns["TriGiaHD"].HeaderText = "Trị giá hoá đơn";
}

```

## 2.5. Danh mục sản phẩm

```

CREATE VIEW [dbo].[view_danhSachSanPham] AS
SELECT *
FROM SanPham

```

Thực thi trên C#:

```

public DataTable getDSSanPham()
{
    SqlCommand command = new SqlCommand("Select * from view_danhSachSanPham",
mydb.getConnection());
    DataTable table = new DataTable();
    SqlDataAdapter adapter = new SqlDataAdapter(command);
    adapter.Fill(table);
    return table;
}

```

## 2.6. Danh mục phiếu chi

```

CREATE VIEW [dbo].[DonNhapHangView]
AS
SELECT DNH.MaDonNH, DNH.NgayNH, DNH.MaNCC, DNH.TinhTrang, PC.MaPhieuChi,
PC.NgayXuatPhieu, CTH.MaNL, CTH.DonGia, CTH.SL, CTH.DonVi, PC.SoTienChi
FROM DonNhapHang DNH
LEFT JOIN PhieuChi PC ON DNH.MaDonNH = PC.MaDonNH
LEFT JOIN ChiTietNhapHang CTH ON DNH.MaDonNH = CTH.MaDonNH;

```

Thực thi trên C#:

```

private void load_tabPage_Payment()
{
    SqlCommand cmd = new SqlCommand("SELECT * FROM DonNhapHangView", db.getConnection());
    SqlDataAdapter adapter = new SqlDataAdapter(cmd);
    DataTable dt = new DataTable();
    adapter.Fill(dt);
    dataGridViewPayment.DataSource = dt;
    //Header text
    dataGridViewPayment.Columns["MaDonNH"].HeaderText = "Mã Đơn Nhập Hàng";
    dataGridViewPayment.Columns["NgayNH"].HeaderText = "Ngày nhập hàng";
    dataGridViewPayment.Columns["NgayNH"].DefaultCellStyle.Format = "dd/MM/yyyy";
    dataGridViewPayment.Columns["MaNCC"].HeaderText = "Mã Nhà Cung Cấp";
    dataGridViewPayment.Columns["TinhTrang"].HeaderText = "Tình trạng";
    dataGridViewPayment.Columns["MaPhieuChi"].HeaderText = "Mã phiếu chi";
    dataGridViewPayment.Columns["NgayXuatPhieu"].HeaderText = "Ngày Xuất Phiếu";
    dataGridViewPayment.Columns["NgayXuatPhieu"].DefaultCellStyle.Format = "dd/MM/yyyy";
    dataGridViewPayment.Columns["MaNL"].HeaderText = "Mã nguyên liệu";
    dataGridViewPayment.Columns["DonGia"].HeaderText = "Đơn giá";
    dataGridViewPayment.Columns["SL"].HeaderText = "Số lượng";
    dataGridViewPayment.Columns["DonVi"].HeaderText = "Đơn vị";
    dataGridViewPayment.Columns["SoTienChi"].HeaderText = "Số tiền chi";
}

```

## 2.7. Danh mục nguyên liệu

```

CREATE VIEW [dbo].[view_NguyenLieu] AS

```



```
SELECT *
FROM NguyenLieu
```

Thực thi trên C#:

```
public DataTable xemNguyenLieu()
{
    SqlCommand command = new SqlCommand("select * from view_NguyenLieu", mydb.getConnection());
    DataTable dt = new DataTable();
    SqlDataAdapter adapter = new SqlDataAdapter(command);
    adapter.Fill(dt);
    return dt;
}
```

## 2.8. Danh mục nhà cung cấp

```
CREATE VIEW [dbo].[view_nhaCungCap] AS
SELECT *
FROM dbo.NhaCungCap
```

Thực thi trên C#:

```
private void themnguyenlieuForm_Load(object sender, EventArgs e)
{
    DataTable tbNCC = new DataTable();
    SqlCommand command = new SqlCommand("select * from view_nhaCungCap", mydb.getConnection());
    SqlDataAdapter adapter = new SqlDataAdapter(command);
    adapter.Fill(tbNCC);

    DataTable tbNguyenLieu = new DataTable();
    tbNguyenLieu = nguyenLieu.getNguyenLieu();

    cbNCC.DataSource = tbNCC;
    cbNCC.DisplayMember = "TenNCC";
    cbNCC.Text = "";

    cbTenNL.DataSource = tbNguyenLieu;
    cbTenNL.DisplayMember = "TenNL";
    cbTenNL.Text = "";

    cbTinhTrang.Items.Add("Còn hàng");
    cbTinhTrang.Items.Add("Hết hàng");

    tbMaNL.Enabled = false;
    cbTenNL.DropDownStyle = ComboBoxStyle.DropDownList;
    cbNCC.Enabled = false;
    tbDonVi.Enabled = false;
}
```

## 3. Quản lý khách hàng

### 3.1. Hàm tìm kiếm khách hàng

```
CREATE FUNCTION [dbo].[SearchTenKHBySDT] (@SDT nchar(11))
RETURNS nvarchar(50)
AS
BEGIN
    DECLARE @TenKH nvarchar(50);
    SELECT @TenKH = TenKH FROM KháchHang WHERE SDT = @SDT;
    RETURN @TenKH;
END
```

Thực thi trên C#:

```
private void btncheckcustomer_Click(object sender, EventArgs e)
{
    string sdt = textBoxcartsdtkh.Text.Trim();
```

```

db.openConnection();
SqlCommand cmd = new SqlCommand("SELECT dbo.SearchTenKHBySDT(@SDT)", db.getConnection);
cmd.Parameters.AddWithValue("@SDT", sdt);
object result = cmd.ExecuteScalar();

if (result != DBNull.Value && result != null)
{
    string tenKH = (string)result;
    // Display the TenKH in the label
    textBoxcarttenkhachhang.Text = tenKH;
    db.closeConnection();
}
else
{
    // If the KhachHang doesn't exist, open the NewKhachHangForm to create a new one
    NewKhachHangForm newKhachHangForm = new NewKhachHangForm(sdt);
    newKhachHangForm.ShowDialog();
    db.closeConnection();
}
}

```

### 3.2. Tạo khách hàng mới

```

CREATE PROCEDURE [dbo].[InsertNewKhachHang]
    @MaKH nchar(10),
    @TenKH nvarchar(50),
    @SDT nchar(11)
AS
BEGIN
    IF EXISTS (SELECT 1 FROM KhachHang WHERE SDT = @SDT)
    BEGIN
        RETURN;
    END
    INSERT INTO KhachHang (MaKH, TenKH, SDT, DiemTL)
    VALUES (@MaKH, @TenKH, @SDT, 0)
END

```

Thực thi trên C#:

```

private void saveButton_Click(object sender, EventArgs e)
{
    string MaKH = textBoxMaKH.Text;
    string TenKH = textBoxTenKH.Text;
    string SDT = textBoxSDT.Text;

    if (string.IsNullOrEmpty(MaKH))
    {
        MessageBox.Show("Please enter a valid value for 'MaKH'.");
        return;
    }

    db.openConnection();
    SqlCommand cmd = new SqlCommand("InsertNewKhachHang", db.getConnection);
    cmd.CommandType = CommandType.StoredProcedure;
    cmd.Parameters.Add("@MaKH", SqlDbType.NChar).Value = MaKH;
    cmd.Parameters.Add("@TenKH", SqlDbType.NVarChar).Value = TenKH;
    cmd.Parameters.Add("@SDT", SqlDbType.NChar).Value = SDT;
}

```

```

if (cmd.ExecuteNonQuery() > 0)
{
    MessageBox.Show("Thêm thành công!", "Add Customer", MessageBoxButtons.OK,
    MessageBoxIcon.Information);
    db.closeConnection();
}
else
{
    MessageBox.Show("Thêm thất bại", "Add Customer", MessageBoxButtons.OK,
    MessageBoxIcon.Error);
    db.closeConnection();
}
}

```

### 3.3. Trigger bắt lỗi khi thêm khách hàng mới

```

CREATE TRIGGER trg_InsertNewKhachHang
ON KhachHang
FOR INSERT, UPDATE
AS
BEGIN
    -- check MaKH
    IF EXISTS (SELECT * FROM inserted WHERE TRIM(MaKH) = ' ')
    BEGIN
        RAISERROR('Mã KH không được để trống', 16, 1)
        ROLLBACK TRANSACTION
        RETURN
    END
    IF NOT EXISTS (SELECT * FROM KhachHang WHERE MaKH IN (SELECT MaKH FROM
    inserted))
    BEGIN
        RAISERROR('Mã KH đã tồn tại', 16, 1)
        ROLLBACK TRANSACTION
        RETURN
    END
    -- check ten KH
    IF EXISTS (SELECT * FROM inserted WHERE TRIM(TenKH) = ' ')
    BEGIN
        RAISERROR('Tên KH không được để trống', 16, 1)
        ROLLBACK TRANSACTION
        RETURN
    END
    -- check SDT
    IF EXISTS (SELECT * FROM inserted WHERE TRIM(SDT) = ' ')
    BEGIN
        RAISERROR('Mã KH không được để trống', 16, 1)
        ROLLBACK TRANSACTION
        RETURN
    END
    IF NOT EXISTS (SELECT * FROM KhachHang WHERE SDT IN (SELECT SDT FROM
    inserted))
    BEGIN
        RAISERROR('SDT đã tồn tại', 16, 1)
        ROLLBACK TRANSACTION
        RETURN
    END
END;

```

## 4. Quản lý hóa đơn

### 4.1. Xuất hóa đơn

```

CREATE PROCEDURE [dbo].[proc_AddHoaDon]
    @MaHD nchar(10),
    @NgayDatHang date,
    @MaKH nchar(10),
    @MaNV nchar(10),
    @TriGiaHD int
AS
BEGIN
    INSERT INTO HoaDon (MaHD, NgayDatHang, MaKH, MaNV, TriGiaHD)
    VALUES (@MaHD, @NgayDatHang, @MaKH, @MaNV, @TriGiaHD);
END;

```

Thực thi trên C#:

```

private void addButton_Click(object sender, EventArgs e)
{
    MY_DB db = new MY_DB();
    // Get the parameter values from the text boxes
    string maHD = maHDTextBox.Text;
    DateTime ngayDatHang = DateTime.Today;

    string maKH = maKHTextBox.Text;
    string maNV = maNVTextBox.Text;
    int triGiaHD;
    if (int.TryParse(triGiaHDTextBox.Text, out triGiaHD) == false)
    {
        MessageBox.Show("TriGiaHD must be a valid integer.");
        return;
    }
    // Set up the stored procedure name and parameter names
    string storedProcedureName = "proc_AddHoaDon";
    string[] parameterNames = new string[] { "@MaHD", "@NgayDatHang", "@MaKH", "@MaNV",
"@TriGiaHD" };
    string returnParameterName = "@ReturnValue";

    using (SqlCommand command = new SqlCommand(storedProcedureName, db.getConnection))
    {
        command.CommandType = System.Data.CommandType.StoredProcedure;

        // Add the parameters
        command.Parameters.Add(parameterNames[0], System.Data.SqlDbType.NChar, 10).Value = maHD;
        command.Parameters.Add(parameterNames[1], System.Data.SqlDbType.Date).Value = ngayDatHang;
        command.Parameters.Add(parameterNames[2], System.Data.SqlDbType.NChar, 10).Value = maKH;
        command.Parameters.Add(parameterNames[3], System.Data.SqlDbType.NChar, 10).Value = maNV;
        command.Parameters.Add(parameterNames[4], System.Data.SqlDbType.Int).Value = triGiaHD;

        // Open the connection
        db.openConnection();

        // Execute the command
        int rowsAffected = command.ExecuteNonQuery();

        // Check the result
        if (rowsAffected > 0)
        {
            MessageBox.Show("Xuất thành công.");
        }
    }
}

```

```

    }
    else
    {
        MessageBox.Show("Xuất thất bại.");
    }
}
}

```

## 4.2. Tìm kiếm hóa đơn

```

CREATE PROCEDURE [dbo].[proc_TimKiemHoaDon]
    @NgayBatDau DATE,
    @NgayKetThuc DATE
AS
BEGIN
    SELECT *
    FROM HoaDon
    WHERE NgayDatHang BETWEEN
        @NgayBatDau AND @NgayKetThuc
END

```

Thực thi trên C#:

```

private void btnsearchReceipt_Click(object sender, EventArgs e)
{
    SqlCommand command = new SqlCommand("proc_TimKiemHoaDon", db.getConnection());
    command.CommandType = CommandType.StoredProcedure;
    command.Parameters.AddWithValue("@NgayBatDau", dateTimePicker1receipt.Value);
    command.Parameters.AddWithValue("@NgayKetThuc", dateTimePicker2receipt.Value);
    db.openConnection();
    DataTable table = new DataTable();
    using (SqlDataAdapter adapter = new SqlDataAdapter(command))
    {
        adapter.Fill(table);
    }
    dataGridView3.DataSource = table;
    db.closeConnection();
}

```

## 5. Quản lý kho nguyên liệu

### 5.1. Tìm kiếm nguyên liệu

```

CREATE FUNCTION [dbo].[func_getIngreListByString] (@string NVARCHAR(50))
RETURNS @IngreList TABLE (MaNL VARCHAR(10), TenNL NVARCHAR(50), MaNCC
VARCHAR(10), SoLuong INT, DonVi NVARCHAR(10), TinhTrang NVARCHAR(10))
AS
BEGIN
    INSERT INTO @IngreList
    SELECT *
    FROM dbo.NguyenLieu
    WHERE CONCAT(MaNL, TenNL, MaNCC, DonVi, TinhTrang) LIKE N'%' + @string + '%'

    RETURN
END

```

Thực thi trên C#:

```

public DataTable timNguyenLieu(string str)
{
    string fnName = "func_getIngreListByString";
}

```

```

SqlCommand command = new SqlCommand("Select * from " + fnName + " (@string)",
mydb.getConnection());
command.Parameters.Add("@string", SqlDbType.NChar).Value = str;
DataTable table = new DataTable();
SqlDataAdapter adapter = new SqlDataAdapter(command);
adapter.Fill(table);
return table;
}

```

## 5.2. Thêm nguyên liệu

```

CREATE PROCEDURE [dbo].[proc_themNguyenLieu]
@MaNL nchar(10),
@TenNL nvarchar(50),
@MaNCC nchar(10),
@SoLuong int,
@DonVi nchar(10),
@TinhTrang nchar(10)
AS
BEGIN
    BEGIN TRY
        Thêm mới sản phẩm
        INSERT INTO dbo.NguyenLieu (MaNL, TenNL, MaNCC, SL, DonVi, TinhTrang)
        VALUES (@MaNL, @TenNL, @MaNCC, @SoLuong, @DonVi, @TinhTrang)
    END TRY
    BEGIN CATCH
        DECLARE @err NVARCHAR(MAX)
        SELECT @err = N'Lỗi ' + ERROR_MESSAGE()
        RAISERROR(@err, 16, 1)
    END CATCH
END

```

Thực thi trên C#:

```

public bool themNguyenLieu(string maNL, string tenNL, string maNCC, int soLuong, string donVi, string
tinhTrang)
{
    SqlCommand command = new SqlCommand("proc_themNguyenLieu", mydb.getConnection());
    command.CommandType = CommandType.StoredProcedure;
    command.Parameters.Add("@MaNL", SqlDbType.NChar).Value = maNL;
    command.Parameters.Add("@TenNL", SqlDbType.NVarChar).Value = tenNL;
    command.Parameters.Add("@MaNCC", SqlDbType.NChar).Value = maNCC;
    command.Parameters.Add("@SoLuong", SqlDbType.Int).Value = soLuong;
    command.Parameters.Add("@DonVi", SqlDbType.NChar).Value = donVi;
    command.Parameters.Add("@TinhTrang", SqlDbType.NChar).Value = tinhTrang;

    mydb.openConnection();

    if (command.ExecuteNonQuery() == 1)
    {
        mydb.closeConnection();
        return true;
    }
    else
    {
        mydb.closeConnection();
        return false;
    }
}

```

## 6. Quản lý sản phẩm

### 6.1. Thêm sản phẩm

```

CREATE PROCEDURE [dbo].[proc_themSanPhamMoi]
@MaSP nchar(10),

```

```

        @TenSP nvarchar(50),
        @DonGia float,
        @TinhTrang nchar(10),
        @MaLoaiSP nchar(10),
        @TenLoaiSP nvarchar(50),
        @HinhAnh image
    AS
    BEGIN
        BEGIN TRANSACTION
        BEGIN TRY
            -- Kiểm tra xem loại sản phẩm đã tồn tại hay chưa
            IF NOT EXISTS (SELECT * FROM LoaiSanPham WHERE MaLoaiSP =
@MaLoaiSP)
                BEGIN
                    -- Nếu chưa tồn tại, thêm mới loại sản phẩm
                    INSERT INTO LoaiSanPham (MaLoaiSP, TenLoaiSP)
                    VALUES (@MaLoaiSP, @TenLoaiSP)
                END

            -- Thêm mới sản phẩm
            INSERT INTO SanPham (MaSP, TenSP, DonGia, TinhTrang, MaLoaiSP,
HinhAnh)
            VALUES (@MaSP, @TenSP, @DonGia, @TinhTrang, @MaLoaiSP,
@HinhAnh)

            COMMIT TRAN
        END TRY

        BEGIN CATCH
            ROLLBACK
            DECLARE @err NVARCHAR(MAX)
            SELECT @err = N'Lỗi' + ERROR_MESSAGE()
            RAISERROR(@err, 16, 1)
        END CATCH
    END

```

Thực thi trên C#:

```

public bool themSanPham(string MaSP, string TenSP, float DonGia, string TinhTrang, string MaLoaiSP, string
TenLoaiSP, MemoryStream HinhAnh)
{
    SqlCommand command = new SqlCommand("proc_themSanPhamMoi", mydb.getConnection());
    command.CommandType = CommandType.StoredProcedure;
    command.Parameters.Add("@MaSP", SqlDbType.NChar).Value = MaSP;
    command.Parameters.Add("@TenSP", SqlDbType.NVarChar).Value = TenSP;
    command.Parameters.Add("@DonGia", SqlDbType.Float).Value = DonGia;
    command.Parameters.Add("@TinhTrang", SqlDbType.NChar).Value = TinhTrang;
    command.Parameters.Add("@MaLoaiSP", SqlDbType.NChar).Value = MaLoaiSP;
    command.Parameters.Add("@TenLoaiSP", SqlDbType.NVarChar).Value = TenLoaiSP;
    command.Parameters.Add("@HinhAnh", SqlDbType.Image).Value = HinhAnh.ToArray();

    mydb.openConnection();

    if (command.ExecuteNonQuery() > 0)
    {
        mydb.closeConnection();
        return true;
    }
}

```

```

    }
    else
    {
        mydb.closeConnection();
        return false;
    }
}

```

## 6.2. Sửa sản phẩm

```

CREATE PROCEDURE [dbo].[proc_suaSanPham]
    @MaSP nchar(10),
    @TenSP nvarchar(50),
    @DonGia float,
    @TinhTrang nchar(10),
    @HinhAnh image
AS
BEGIN
    BEGIN TRY
        -- Thêm mới sản phẩm
        UPDATE dbo.SanPham SET MaSP = @MaSP, TenSP = @TenSP, DonGia =
@DonGia,
                                TinhTrang = @TinhTrang, HinhAnh =
@HinhAnh
        WHERE MaSP = @MaSP
    END TRY
    BEGIN CATCH
        DECLARE @err NVARCHAR(MAX)
        SELECT @err = N'Lỗi' + ERROR_MESSAGE()
        RAISERROR(@err, 16, 1)
    END CATCH
END

```

Thực thi trên C#:

```

public bool suaSanPham(string MaSP, string TenSP, float DonGia, string TinhTrang, MemoryStream HinhAnh)
{
    SqlCommand command = new SqlCommand("proc_suaSanPham", mydb.getConnection());
    command.CommandType = CommandType.StoredProcedure;
    command.Parameters.Add("@MaSP", SqlDbType.NChar).Value = MaSP;
    command.Parameters.Add("@TenSP", SqlDbType.NVarChar).Value = TenSP;
    command.Parameters.Add("@DonGia", SqlDbType.Float).Value = DonGia;
    command.Parameters.Add("@TinhTrang", SqlDbType.NChar).Value = TinhTrang;
    command.Parameters.Add("@HinhAnh", SqlDbType.Image).Value = HinhAnh.ToArray();

    mydb.openConnection();

    if (command.ExecuteNonQuery() > 0)
    {
        mydb.closeConnection();
        return true;
    }
    else
    {
        mydb.closeConnection();
        return false;
    }
}

```

## 6.3. Xóa sản phẩm

```

CREATE PROCEDURE [dbo].[proc_xoaSanPham]
    @MaSP nchar(10)
AS
BEGIN

```



```

BEGIN TRANSACTION
    BEGIN TRY
        --Xoá nguyên liệu tiêu tốn theo @MaSP trong bảng CheBien
        DELETE FROM    dbo.CheBien WHERE CheBien.MaSP = @MaSP

        --Xoá sản phẩm theo @MaSP trong bảng SanPham
        DELETE FROM    dbo.SanPham WHERE SanPham.MaSP = @MaSP
        COMMIT TRAN
    END TRY

    BEGIN CATCH
        ROLLBACK
        DECLARE @err NVARCHAR(MAX)
        SELECT @err = N'Lỗi' + ERROR_MESSAGE()
        RAISERROR(@err, 16, 1)
    END CATCH
END

```

Thực thi trên C#:

```

public bool xoaSanPham(string maSP)
{
    SqlCommand command = new SqlCommand("dbo.proc_xoaSanPham", mydb.getConnection());
    command.CommandType = CommandType.StoredProcedure;
    command.Parameters.Add("@MaSP", SqlDbType.NChar).Value = maSP;

    mydb.openConnection();

    if (command.ExecuteNonQuery() > 0)
    {
        mydb.closeConnection();
        return true;
    }
    else
    {
        mydb.closeConnection();
        return false;
    }
}

```

#### 6.4. Tìm kiếm sản phẩm

```

CREATE FUNCTION [dbo].[func_getProductListByString] (@string NVARCHAR(50))
RETURNS @ProductList TABLE (MaSP VARCHAR(10), TenSP NVARCHAR(50), DonGia INT,
TinhTrang NVARCHAR(10), MaLoaiSP VARCHAR(10), HinhAnh IMAGE)
AS
BEGIN
    INSERT INTO @ProductList
    SELECT MaSP, TenSP, DonGia, TinhTrang, MaLoaiSP, HinhAnh
    FROM    dbo.SanPham
    WHERE   CONCAT(MaSP, TenSP, TinhTrang, MaLoaiSP) LIKE '%' + @string + '%'
    RETURN
END

```

Thực thi trên C#:

```

public DataTable getDSSanPhamByString(string str)
{
    string fnName = "func_getProductListByString";
}

```

```

SqlCommand command = new SqlCommand("Select * from " + fnName + " (@string)",
mydb.getConnection());
command.Parameters.Add("@string", SqlDbType.NChar).Value = str;
DataTable table = new DataTable();
SqlDataAdapter adapter = new SqlDataAdapter(command);
adapter.Fill(table);
return table;
}

```

## 6.5. Trigger bắt lỗi trùng tên khi thêm và chỉnh sửa sản phẩm

```

CREATE TRIGGER TG_TrungTenSP
ON dbo.SanPham
AFTER INSERT, UPDATE
AS
BEGIN
    -- Kiểm tra tên sản phẩm vừa thêm có bị trùng lặp
    IF EXISTS (
        SELECT *
        FROM inserted i
        WHERE EXISTS (
            SELECT *
            FROM dbo.SanPham sp
            WHERE sp.TenSP = i.TenSP AND sp.MaSP <> i.MaSP
        )
    )
    BEGIN
        -- Nếu trùng thì rollback
        RAISERROR ('Tên sản phẩm bị trùng', 16, 1)
        ROLLBACK;
    END
END

```

## 7. Thống kê doanh thu và chi phí

### 7.1. Doanh thu theo ngày, tháng, năm

```

CREATE FUNCTION [dbo].[func_tinhDoanhThuNgay](@ngay INT, @thang INT, @nam INT)
RETURNS FLOAT
AS
BEGIN
    DECLARE @doanhThu FLOAT = 0;
    SELECT @doanhThu = COALESCE(SUM(TriGiaHD), 0)
    FROM HoaDon
    WHERE DAY(NgayDatHang) = @ngay AND MONTH(NgayDatHang) = @thang AND
YEAR(NgayDatHang) = @nam;
    RETURN @doanhThu;
END;

CREATE FUNCTION [dbo].[func_tinhDoanhThuThang](@thang INT, @nam INT) RETURNS
float
BEGIN
    DECLARE @doanhThu float = 0;
    SELECT @doanhthu = COALESCE(SUM(TriGiaHD), 0)
    FROM HoaDon
    WHERE MONTH(NgayDatHang) = @thang AND YEAR(NgayDatHang) = @nam;
    RETURN @doanhThu;
END;

CREATE FUNCTION [dbo].[func_tinhDoanhThuNam](@nam INT) RETURNS float
BEGIN

```

```

DECLARE @doanhThu float = 0;
SELECT @doanhthu = COALESCE(SUM(TriGiaHD), 0)
FROM HoaDon
WHERE YEAR(NgayDatHang) = @nam;
RETURN @doanhThu;
END;

```

## 7.2. Chi phí theo tháng, năm

```

CREATE FUNCTION [dbo].[func_tinhTongChiPhiThang](@thang INT, @nam INT) RETURNS float
BEGIN
    DECLARE @chiphi float;
    SELECT @chiphi = SUM(SoTienChi)
    FROM PhieuChi
    WHERE MONTH(NgayXuatPhieu) = @thang AND YEAR(NgayXuatPhieu) = @nam;
    RETURN @chiphi;
END;

CREATE FUNCTION [dbo].[func_tinhTongChiPhiNam](@nam INT) RETURNS float
BEGIN
    DECLARE @chiphi float;
    SELECT @chiphi = SUM(SoTienChi)
    FROM PhieuChi
    WHERE YEAR(NgayXuatPhieu) = @nam;
    RETURN @chiphi;
END;

```

Thực thi trên C# của doanh thu và chi phí:

```

public void load_tabPage_Statics()
{
    db.openConnection();
    SqlCommand cmd = new SqlCommand("SELECT dbo.func_tinhDoanhThuNgay(@ngay, @thang, @nam)", db.getConnection());
    cmd.Parameters.AddWithValue("@ngay", int.Parse(DateTime.Today.ToString("dd")));
    cmd.Parameters.AddWithValue("@thang", int.Parse(DateTime.Today.ToString("MM")));
    cmd.Parameters.AddWithValue("@nam", int.Parse(DateTime.Today.ToString("yyyy")));
    object result = cmd.ExecuteScalar();
    double doanhthutheothang = (double)result; // change data type to double
    string doanhthuString = doanhthutheothang.ToString(); // convert to string
    labeldoanhthutheongay.Text = "Doanh thu theo ngày: " + doanhthuString + " VND"; // set label text
    db.closeConnection();

    db.openConnection();
    cmd = new SqlCommand("SELECT dbo.func_tinhDoanhThuThang(@thang, @nam)", db.getConnection());
    cmd.Parameters.AddWithValue("@thang", int.Parse(DateTime.Today.ToString("MM")));
    cmd.Parameters.AddWithValue("@nam", int.Parse(DateTime.Today.ToString("yyyy")));
    result = cmd.ExecuteScalar();
    doanhthutheothang = (double)result; // change data type to double
    doanhthuString = doanhthutheothang.ToString(); // convert to string
    labeldoanhthutheothang.Text = "Doanh thu theo tháng: " + doanhthuString + " VND"; // set label text
    db.closeConnection();

    db.openConnection();
    cmd = new SqlCommand("SELECT dbo.func_tinhDoanhThuNam(@nam)", db.getConnection());
    cmd.Parameters.AddWithValue("@nam", int.Parse(DateTime.Today.ToString("yyyy")));
    result = cmd.ExecuteScalar();
}

```

```

doanhthutheothang = (double)result; // change data type to double
doanhthuString = doanhthutheothang.ToString(); // convert to string
labeldoanhthutheonam.Text = "Doanh thu theo năm: " + doanhthuString + " VND"; // set label text
db.closeConnection();

db.openConnection();
cmd = new SqlCommand("SELECT dbo.func_tinhTongChiPhiThang(@thang, @nam)",
db.getConnection());
cmd.Parameters.AddWithValue("@thang", int.Parse(DateTime.Today.ToString("MM")));
cmd.Parameters.AddWithValue("@nam", int.Parse(DateTime.Today.ToString("yyyy")));
result = cmd.ExecuteScalar();
doanhthutheothang = (double)result; // change data type to double
doanhthuString = doanhthutheothang.ToString(); // convert to string
labelchiphithang.Text = "Chi phí theo tháng: " + doanhthuString + " VND"; // set label text
db.closeConnection();

db.openConnection();
cmd = new SqlCommand("SELECT dbo.func_tinhTongChiPhiNam(@nam)", db.getConnection());
cmd.Parameters.AddWithValue("@nam", int.Parse(DateTime.Today.ToString("yyyy")));
result = cmd.ExecuteScalar();
doanhthutheothang = (double)result; // change data type to double
doanhthuString = doanhthutheothang.ToString(); // convert to string
labelchiphitheonam.Text = "Chi phí theo năm: " + doanhthuString + " VND"; // set label text
db.closeConnection();
}

```

## 8. Quản lý nhân viên

### 8.1. Thêm nhân viên

```

CREATE PROC [dbo].[proc_AddEmployee]
@manv nchar(10), @ho nvarchar(10), @ten nvarchar(10), @ns date, @gt nvarchar(3),
@dc nvarchar(100),
@sdt nchar(11), @macv nchar(10), @soca int, @thuong int, @ntd date, @ha image
AS
INSERT INTO NhanVien (MaNV, HoNV, TenNV, NgaySinh, GioiTinh, DiaChi, SDT,
MaCV, SoCa, Thuong, NgayTuyenDung, HinhAnh)
VALUES (@manv, @ho, @ten, @ns, @gt, @dc, @sdt, @macv, @soca, @thuong, @ntd,
@ha)

```

Thực thi trên C#:

```

public bool Insert(string id, string ho, string ten, DateTime bd, string gd, string adr, string phn, ComboBox cv, int
soca, int thuong, DateTime hd, MemoryStream pic)
{
    try
    {
        SqlCommand cmd = new SqlCommand("EXEC proc_AddEmployee @manv, @ho, @ten, @ns, @gt,
@dc, @sdt, @macv, @soca, @thuong, @ntd, @ha", db.getConnection());

        cmd.Parameters.Add("@manv", SqlDbType.NChar, 10).Value = id;
        cmd.Parameters.Add("@ho", SqlDbType.NVarChar, 10).Value = ho;
        cmd.Parameters.Add("@ten", SqlDbType.NVarChar, 10).Value = ten;
        cmd.Parameters.Add("@ns", SqlDbType.Date).Value = bd;
        cmd.Parameters.Add("@gt", SqlDbType.NVarChar, 3).Value = gd;
        cmd.Parameters.Add("@dc", SqlDbType.NVarChar, 100).Value = adr;
        cmd.Parameters.Add("@sdt", SqlDbType.NVarChar, 11).Value = phn;
        cmd.Parameters.Add("@macv", SqlDbType.NChar, 10).Value = cv.SelectedValue;
        cmd.Parameters.Add("@soca", SqlDbType.Int).Value = soca;
        cmd.Parameters.Add("@thuong", SqlDbType.Int).Value = thuong;
        cmd.Parameters.Add("@ntd", SqlDbType.Date).Value = hd;
        cmd.Parameters.Add("@ha", SqlDbType.Image).Value = pic.ToArray();
    }
}

```

```

db.openConnection();
try
{
    cmd.ExecuteNonQuery();
    db.closeConnection();
    return true;
}
catch
{
    db.closeConnection();
    return false;
}
}
catch
{
    return false;
}
}

```

## 8.2. Xóa nhân viên

```

CREATE PROCEDURE [dbo].[proc_DeleteEmployee]
    @MaNV nchar(10)
AS
BEGIN
    SET NOCOUNT ON;

    BEGIN TRANSACTION;
    BEGIN TRY
        -- Xóa dòng trong bảng OrderDetails có OrderID tương ứng
        DELETE FROM BangPhanCa WHERE MaNV = @MaNV;

        -- Xóa dòng trong bảng Orders có OrderID tương ứng
        DELETE FROM NhanVien WHERE MaNV = @MaNV;
    END TRY
    BEGIN CATCH
        DECLARE @err NVARCHAR(MAX)
        SELECT @err = N'Lỗi ' + ERROR_MESSAGE()
        RAISERROR(@err, 16, 1)
        ROLLBACK TRANSACTION;
        THROW;
    END CATCH
    COMMIT TRANSACTION;
END

```

Thực thi trên C#:

```

private void remove_Employee()
{
    try
    {
        DialogResult result = MessageBox.Show("Bạn có chắc chắn muốn xóa nhân viên này không?", "Remove employee", MessageBoxButtons.YesNo, MessageBoxIcon.Error);
        if (result == DialogResult.Yes)
        {
            string ID = dataGridView_Employees.CurrentRow.Cells[0].Value.ToString();

            SqlCommand cmd = new SqlCommand("EXEC proc_DeleteEmployee @manv", db.getConnection);
            cmd.Parameters.Add("@manv", SqlDbType.NChar, 10).Value = ID;

            db.openConnection();
            if (cmd.ExecuteNonQuery() == -1)
            {
                MessageBox.Show("Xóa thành công!", "Remove employee", MessageBoxButtons.OK, MessageBoxIcon.Information);
            }
        }
    }
}

```

```

        db.closeConnection();
        reset_Employee();
    }
    else
    {
        MessageBox.Show("Xóa thất bại!", "Remove employee", MessageBoxButtons.OK,
        MessageBoxIcon.Error);
        db.closeConnection();
    }
}
}
catch
{
    MessageBox.Show("Xóa thất bại!", "Remove employee", MessageBoxButtons.OK,
    MessageBoxIcon.Error);
}
}

```

### 8.3. Tìm kiếm nhân viên

```

CREATE PROC [dbo].[pro_SearchByMaNV]
    @MaNV nchar(10)
AS
BEGIN
    SELECT *
    FROM v_LoadEmployee
    WHERE MaNV = @MaNV
END

CREATE PROC [dbo].[pro_SearchBySDT]
    @SDT nchar(11)
AS
BEGIN
    SELECT *
    FROM v_LoadEmployee
    WHERE SDT = @SDT
END

CREATE PROC [dbo].[pro_SearchByTenCV]
    @TenCV nvarchar(50)
AS
BEGIN
    SELECT *
    FROM v_LoadEmployee
    WHERE TenCV = @TenCV
END

CREATE PROC [dbo].[pro_SearchByTenNV]
    @TenNV nvarchar(10)
AS
BEGIN
    SELECT *
    FROM v_LoadEmployee
    WHERE TenNV = @TenNV
END

```

Thực thi trên C#:

```

private void search_Employee()
{
    if (txt_Search_Employee.Text == "")
    {
        load_tabPage_Employees();
    }
    else

```

```

{
    SqlCommand cmd = new SqlCommand("", db.getConnection);

    if (comboBox_Search_Employee.Text == "Tim theo Mã NV")
    {
        cmd.CommandText = "EXEC pro_SearchByMaNV @manv";
        cmd.Parameters.Add("@manv", SqlDbType.NChar, 10).Value = txt_Search_Employee.Text;
    }
    else if (comboBox_Search_Employee.Text == "Tim theo Tên")
    {
        cmd.CommandText = "EXEC pro_SearchByTenNV @tennv";
        cmd.Parameters.Add("@tennv", SqlDbType.NChar, 10).Value = txt_Search_Employee.Text;
    }
    else if (comboBox_Search_Employee.Text == "Tim theo SĐT")
    {
        cmd.CommandText = "EXEC pro_SearchBySDT @sdt";
        cmd.Parameters.Add("@sdt", SqlDbType.NChar, 11).Value = txt_Search_Employee.Text;
    }
    else if (comboBox_Search_Employee.Text == "Tim theo Công Việc")
    {
        cmd.CommandText = "EXEC pro_SearchByTenCV @tencv";
        cmd.Parameters.Add("@tencv", SqlDbType.NVarChar, 50).Value = txt_Search_Employee.Text;
    }
    DataTable dt = employee.getEmployeeWithCommand(cmd);
    if (dt.Rows.Count > 0)
    {
        dataGridView_Employees.DataSource = dt;
    }
    else
    {
        MessageBox.Show("Không tìm thấy!", "Search employee", MessageBoxButtons.OK,
        MessageBoxIcon.Error);
    }
}
}

```

#### 8.4. Tính lương nhân viên

```

CREATE FUNCTION func_tinhLuongNV(@thang INT, @nam INT) RETURNS TABLE
AS RETURN (
    SELECT nv.MaNV, nv.HoNV, nv.TenNV, nv.SoCa,
    (nv.SoCa*cv.Luong*4 + nv.Thuong*(cv.Luong*2)) +
    CASE WHEN dbo.func_tinhDoanhThuThang(@thang, @nam) > 100000000
    THEN dbo.func_tinhDoanhThuThang(@thang, @nam)*0.01
    ELSE 0
    END AS Luong
    FROM NhanVien nv
    INNER JOIN CongViec cv ON nv.MaCV = cv.MaCV
)

```

Code C# gọi danh mục:

```

private void tinhluongForm_Load(object sender, EventArgs e)
{
    MY_DB db = new MY_DB();
    db.openConnection();

    SqlCommand cmd = new SqlCommand("SELECT * FROM func_tinhLuongNV(@thang, @nam)",
    db.getConnection);

    cmd.Parameters.AddWithValue("@thang", int.Parse(DateTime.Today.ToString("MM")));
    cmd.Parameters.AddWithValue("@nam", int.Parse(DateTime.Today.ToString("yyyy")));
}

```

```
// Create a data adapter to fill a dataset with the results
SqlDataAdapter da = new SqlDataAdapter(cmd);
DataSet ds = new DataSet();
da.Fill(ds);

dataGridViewluong.DataSource = ds.Tables[0];
db.closeConnection();
}
```

## 8.5. Sửa thông tin nhân viên

```
CREATE PROC [dbo].[proc_EditEmployee]
    @manv nchar(10), @ho nvarchar(10), @ten nvarchar(10), @ngaysinh date,
    @gioitinh nvarchar(3), @diachi nvarchar(100), @sdt nchar(11), @macv
nchar(10),
    @soca int, @thuong int, @ngaytuyendung date, @hinhanh image
AS
BEGIN
    BEGIN TRY
        UPDATE NhanVien
        SET
            HoNV = @ho, TenNV = @ten, NgaySinh = @ngaysinh,
            GioiTinh = @gioitinh, DiaChi = @diachi, SDT = @sdt, MaCV = @macv,
            SoCa = @soca, Thuong = @thuong, NgayTuyenDung = @ngaytuyendung,
            HinhAnh = @hinhanh
        WHERE MaNV = @manv
    END TRY
    BEGIN CATCH
        DECLARE @err NVARCHAR(MAX)
        SELECT @err = N'Lỗi ' + ERROR_MESSAGE()
        RAISERROR(@err, 16, 1)
    END CATCH
END
```

Code C# gọi danh mục:

```
public bool Update(string id, string ho, string ten, DateTime bd, string gd, string adr, string phn, ComboBox cv, int
soca, int thuong, DateTime hd, MemoryStream pic)
{
    try
    {
        SqlCommand cmd = new SqlCommand("EXEC proc_EditEmployee @manv, @ho, @ten, @ns, @gt,
@dc, @sdt, @macv, @soca, @thuong, @ntd, @ha", db.getConnection());

        cmd.Parameters.Add("@manv", SqlDbType.NChar, 10).Value = id;
        cmd.Parameters.Add("@ho", SqlDbType.NVarChar, 10).Value = ho;
        cmd.Parameters.Add("@ten", SqlDbType.NVarChar, 10).Value = ten;
        cmd.Parameters.Add("@ns", SqlDbType.Date).Value = bd;
        cmd.Parameters.Add("@gt", SqlDbType.NVarChar, 3).Value = gd;
        cmd.Parameters.Add("@dc", SqlDbType.NVarChar, 100).Value = adr;
        cmd.Parameters.Add("@sdt", SqlDbType.NVarChar, 11).Value = phn;
        cmd.Parameters.Add("@macv", SqlDbType.NChar, 10).Value = cv.SelectedValue;
        cmd.Parameters.Add("@soca", SqlDbType.Int).Value = soca;
        cmd.Parameters.Add("@thuong", SqlDbType.Int).Value = thuong;
        cmd.Parameters.Add("@ntd", SqlDbType.Date).Value = hd;
        cmd.Parameters.Add("@ha", SqlDbType.Image).Value = pic.ToArray();
        db.openConnection();
    }
    try
    {
        cmd.ExecuteNonQuery();
        db.closeConnection();
        return true;
    }
    catch
}
```



```

    {
        db.closeConnection();
        return false;
    }
}
catch
{
    return false;
}
}

```

## 8.6. Trigger bắt lỗi khi thêm hoặc sửa thông tin nhân viên

```

CREATE TRIGGER trg_CheckNhanVien
ON NhanVien
FOR INSERT, UPDATE
AS
BEGIN
    -- check MaNV
    IF EXISTS (SELECT * FROM inserted WHERE TRIM(MaNV) = ' ')
    BEGIN
        RAISERROR('Mã nhân viên không được để trống', 16, 1)
        ROLLBACK TRANSACTION
        RETURN
    END
    IF NOT EXISTS (SELECT * FROM NhanVien WHERE MaNV IN (SELECT MaNV FROM
inserted))
    BEGIN
        RAISERROR('Mã nhân viên đã tồn tại', 16, 1)
        ROLLBACK TRANSACTION
        RETURN
    END
    -- check ho ten nhan vien
    IF EXISTS (SELECT * FROM inserted WHERE TRIM(TenNV) = '' )
    BEGIN
        RAISERROR('Tên nhân viên không được để trống', 16, 1)
        ROLLBACK TRANSACTION
        RETURN
    END
    IF EXISTS (SELECT * FROM inserted WHERE TRIM(HoNV) = '')
    BEGIN
        RAISERROR('Họ nhân viên không được để trống', 16, 1)
        ROLLBACK TRANSACTION
        RETURN
    END
    IF EXISTS (SELECT 1 FROM inserted i INNER JOIN NhanVien n ON i.MaNV != n.MaNV
AND TRIM(i.HoNV) = TRIM(n.HoNV) AND TRIM(i.TenNV) = TRIM(n.TenNV))
    BEGIN
        RAISERROR('Họ tên nhân viên đã tồn tại', 16, 1)
        ROLLBACK TRANSACTION
        RETURN
    END
    -- check ngay sinh
    IF EXISTS (
        SELECT * FROM inserted WHERE
datediff(year,inserted.NgaySinh,getdate())<(18)
    )
    BEGIN
        RAISERROR ('Nhân viên phải trên 18 tuổi', 16, 1)
        ROLLBACK TRANSACTION
        RETURN
    END
    -- check dia chi

```

```

IF EXISTS (SELECT * FROM inserted WHERE TRIM(DiaChi) = '' )
BEGIN
    RAISERROR('Địa chỉ không được để trống', 16, 1)
    ROLLBACK TRANSACTION
    RETURN
END
-- check so dien thoai
IF EXISTS (SELECT * FROM inserted WHERE TRIM(SDT) = '' )
BEGIN
    RAISERROR('Số điện thoại không được để trống', 16, 1)
    ROLLBACK TRANSACTION
    RETURN
END
IF NOT EXISTS (SELECT * FROM inserted WHERE len(TRIM(SDT)) = (10))
BEGIN
    RAISERROR('Số điện thoại gồm 10 chữ số', 16, 1)
    ROLLBACK TRANSACTION
    RETURN
END
IF EXISTS (SELECT 1 FROM inserted i INNER JOIN NhanVien n ON i.MaNV != n.MaNV
AND TRIM(i.SDT) = TRIM(n.SDT))
BEGIN
    RAISERROR('Số điện thoại đã tồn tại', 16, 1)
    ROLLBACK TRANSACTION
    RETURN
END
-- check ngay tuyen dung
IF NOT EXISTS (
    SELECT * FROM inserted WHERE
    (datediff(day,[NgayTuyenDung],getdate())>=(0))
)
BEGIN
    RAISERROR ('Ngày tuyển dụng không thể là trong tương lai', 16, 1)
    ROLLBACK TRANSACTION
    RETURN
END
END
END

```

## 9. Quản lý phiếu chi

### 9.1. Thêm phiếu chi

```

CREATE PROCEDURE [dbo].[proc_InsertDonNhapHangAndPhieuChiAndChiTietNhapHang]
    @MaDonNH nchar(10),
    @NgayNH date,
    @TriGiaDonNH float,
    @MaNCC nchar(10),
    @TinhTrang nvarchar(50),
    @MaPhieuChi nchar(10),
    @NgayXuatPhieu date,
    @SoTienChi float,
    @MaNL nchar(10),
    @DonGia float,
    @SL int,
    @DonVi nchar(10),
    @TongTien float
AS
BEGIN
    BEGIN TRANSACTION;

    BEGIN TRY

        -- Insert into DonNhapHang table
        INSERT INTO DonNhapHang (MaDonNH, NgayNH, TriGiaDonNH, MaNCC, TinhTrang)
    
```

```

VALUES (@MaDonNH, @NgayNH, @TriGiaDonNH, @MaNCC, @TinhTrang)

-- Insert into PhieuChi table
INSERT INTO PhieuChi (MaPhieuChi, MaDonNH, NgayXuatPhieu, SoTienChi)
VALUES (@MaPhieuChi, @MaDonNH, @NgayXuatPhieu, @SoTienChi)

-- Insert into ChiTietNhapHang table
INSERT INTO ChiTietNhapHang (MaDonNH, MaNL, DonGia, SL, DonVi, TongTien)
VALUES (@MaDonNH, @MaNL, @DonGia, @SL, @DonVi, @TongTien)

END TRY
BEGIN CATCH
    ROLLBACK TRANSACTION;
    THROW;
END CATCH;

COMMIT TRANSACTION;
END

```

Code C# thực thi:

```

private void btnaddPayment_Click(object sender, EventArgs e)
{
    MY_DB db = new MY_DB();
    SqlCommand command = new
SqlCommand("proc_InsertDonNhapHangAndPhieuChiAndChiTietNhapHang", db.getConnection());
    command.CommandType = CommandType.StoredProcedure;
    float TongTien = float.Parse(textBoxDonGia.Text) * float.Parse(textBoxSL.Text);
    // Add parameters for the DonNhapHang table
    command.Parameters.AddWithValue("@MaDonNH", textBoxMaDonNH.Text);
    command.Parameters.AddWithValue("@NgayNH", DateTime.Today);
    command.Parameters.AddWithValue("@TriGiaDonNH", TongTien);
    command.Parameters.AddWithValue("@MaNCC", textBoxMaNCC.Text);
    command.Parameters.AddWithValue("@TinhTrang", "Đang giao");

    // Add parameters for the PhieuChi table
    command.Parameters.AddWithValue("@MaPhieuChi", textBoxMaPhieuChi.Text);
    command.Parameters.AddWithValue("@NgayXuatPhieu", DateTime.Today);
    command.Parameters.AddWithValue("@SoTienChi", TongTien);

    // Add parameters for the ChiTietNhapHang table
    command.Parameters.AddWithValue("@MaNL", textBoxMaNL.Text);
    command.Parameters.AddWithValue("@DonGia", float.Parse(textBoxDonGia.Text));
    command.Parameters.AddWithValue("@SL", int.Parse(textBoxSL.Text));
    command.Parameters.AddWithValue("@DonVi", textBoxDonVi.Text);
    command.Parameters.AddWithValue("@TongTien", TongTien);

    db.openConnection();
    int rowsAffected = command.ExecuteNonQuery();

    // Check the result
    if (rowsAffected > 0)
    {
        MessageBox.Show("Xuất thành công.");
    }
    else
    {
        MessageBox.Show("Xuất thất bại.");
    }
    db.closeConnection();
}

```

## 10. Quản lý phân ca

### 10.1. Thêm ca cho nhân viên

```
CREATE PROC [dbo].[proc_PhanCa]
    @maca nchar(10),
    @manv nchar(10),
    @dayOfWeek nchar(10)
AS
BEGIN
    BEGIN TRY
        INSERT INTO BangPhanCa
            VALUES(@maca, @manv, @dayOfWeek)
    END TRY
    BEGIN CATCH
        DECLARE @err NVARCHAR(MAX)
        SELECT @err = N'Lỗi ' + ERROR_MESSAGE()
        RAISERROR(@err, 16, 1)
    END CATCH
END
```

Code C# thực thi:

```
private void shift()
{
    SqlCommand cmd = new SqlCommand("EXEC proc_PhanCa @maca, @manv, @dayOfWeek",
db.getConnection());
    cmd.Parameters.Add("@maca", SqlDbType.NChar, 10).Value = comboBox_shift.Text;
    cmd.Parameters.Add("@manv", SqlDbType.NChar, 10).Value = txt_ID.Text;
    cmd.Parameters.Add("@dayOfWeek", SqlDbType.NChar, 10).Value = comboBox_dayOfWeek.Text;
    db.openConnection();
    try
    {
        cmd.ExecuteNonQuery();
        db.closeConnection();
        MessageBox.Show("Phân ca thành công!", "Phân ca", MessageBoxButtons.OK,
MessageBoxIcon.Information);
    }
    catch (SqlException sqlEx)
    {
        db.closeConnection();
        MessageBox.Show("Phân ca thất bại!\n\nLỗi: " + sqlEx.Message, "Phân ca", MessageBoxButtons.OK,
MessageBoxIcon.Exclamation);
    }
    catch (Exception ex)
    {
        db.closeConnection();
        MessageBox.Show("Phân ca thất bại!\n\nLỗi: " + ex.Message, "Phân ca", MessageBoxButtons.OK,
MessageBoxIcon.Exclamation);
    }
}
```

### 10.2. Xóa ca làm của nhân viên

```
CREATE PROC [dbo].[proc_XoaPhanCa]
    @manv nchar(10),
    @maca nchar(10),
    @ngay nchar(10)
AS
BEGIN
    BEGIN TRY
        DELETE FROM BangPhanCa
        WHERE MaNV = @manv and
            MaCa = @maca and
            NgayTrongTuan = @ngay
    END TRY
```

```

BEGIN CATCH
DECLARE @err NVARCHAR(MAX)
SELECT @err = N'Lỗi ' + ERROR_MESSAGE()
RAISERROR(@err, 16, 1)
END CATCH
END

```

Code C# thực thi:

```

private void btn_remove_Click(object sender, EventArgs e)
{
    SqlCommand cmd = new SqlCommand("EXEC proc_XoaPhanCa @manv, @maca, @ngay",
db.getConnection());
    cmd.Parameters.Add("@manv", SqlDbType.NChar, 10).Value = txt_IdEmployee.Text;
    cmd.Parameters.Add("@maca", SqlDbType.NChar, 10).Value = comboBox_IdShift.Text;
    cmd.Parameters.Add("@ngay", SqlDbType.NChar, 10).Value = comboBox_dayOfWeek.Text;
    db.openConnection();
    try
    {
        cmd.ExecuteNonQuery();
        db.closeConnection();
        MessageBox.Show("Xóa thành công!", "Xóa phân ca", MessageBoxButtons.OK,
MessageBoxIcon.Information);
        this.Close();
    }
    catch (SqlException sqlEx)
    {
        db.closeConnection();
        MessageBox.Show("Xóa thất bại!\n\nLỗi: " + sqlEx.Message, "Xóa phân ca", MessageBoxButtons.OK,
MessageBoxIcon.Error);
    }
    catch (Exception ex)
    {
        db.closeConnection();
        MessageBox.Show("Xóa thất bại!\n\nLỗi: " + ex.Message, "Xóa phân ca", MessageBoxButtons.OK,
MessageBoxIcon.Error);
    }
}
}

```

## CHƯƠNG 4: TẠO USER VÀ PHÂN QUYỀN

Ứng với hai nhóm người dùng (Nhân viên và quản lý) là các role riêng

+ Đối với Role Staff (dành cho nhân viên):

```
CREATE ROLE Staff
--Gán các quyền trên table cho role Staff
GRANT SELECT, REFERENCES ON BangPhanCa TO Staff
GRANT SELECT, REFERENCES ON CaLamViec TO Staff
GRANT SELECT, REFERENCES ON CheBien TO Staff
GRANT SELECT, REFERENCES ON ChiTietHoaDon TO Staff
GRANT SELECT, REFERENCES ON ChiTietHoaDonUngDung TO Staff
GRANT SELECT, REFERENCES ON ChiTietNhapHang TO Staff
GRANT SELECT, REFERENCES ON CongViec TO Staff
GRANT SELECT, INSERT, REFERENCES ON DonNhapHang TO Staff
GRANT SELECT, INSERT, REFERENCES ON HoaDon TO Staff
GRANT SELECT, INSERT, REFERENCES ON HoaDonUngDung TO Staff
GRANT SELECT, INSERT, REFERENCES ON KhachHang TO Staff
GRANT SELECT, REFERENCES ON LoaiSanPham TO Staff
GRANT SELECT, REFERENCES ON NguyenLieu TO Staff
GRANT SELECT, REFERENCES ON NhaCungCap TO Staff
GRANT SELECT, REFERENCES ON NhanVien TO Staff
GRANT SELECT, INSERT, REFERENCES ON PhieuChi TO Staff
GRANT SELECT, REFERENCES ON SanPham TO Staff
GRANT SELECT, REFERENCES ON UngDung TO Staff
    Gán quyền thực thi trên các procedure, function cho role Staff
GRANT EXECUTE TO Staff
GRANT SELECT TO Staff

DENY EXECUTE ON proc_AddEmployee to Staff;
DENY EXECUTE ON proc_DeleteEmployee to Staff;
DENY EXECUTE ON proc_EditEmployee to Staff;
DENY EXECUTE ON proc_suaCheBien to Staff;
DENY EXECUTE ON proc_suaNguyenLieu to Staff;
DENY EXECUTE ON proc_suaSanPham to Staff;
DENY EXECUTE ON proc_xoaSanPham to Staff;
DENY EXECUTE ON proc_xoaPhanCa to Staff;
```

+ Đối với Role Manager ta sử dụng server role sysadmin.

- Trigger tạo tài khoản

```
CREATE TRIGGER [dbo].[Trigger_CreateSQLAccount] ON [dbo].[DANGNHAP]
AFTER INSERT
AS
DECLARE @userName nvarchar(30), @passWord nvarchar(10), @manv nvarchar(10)
SELECT @userName=n1.TenDangNhap, @passWord=n1.MatKhau, @manv=n1.maNV
FROM inserted n1
BEGIN
    DECLARE @sqlString nvarchar(2000), @macv nvarchar(10)
    -----
    SET @sqlString= 'CREATE LOGIN [' + @userName + '] WITH PASSWORD=''' +
@passWord
+''', DEFAULT_DATABASE=[QUANLYQUANCOFFEE_Cur], CHECK_EXPIRATION=OFF,
CHECK_POLICY=OFF'
    EXEC (@sqlString)
    -----
    SET @sqlString= 'CREATE USER ' + @userName + ' FOR LOGIN ' + @userName
    EXEC (@sqlString)
    -----
    SELECT @macv = MaCV
```

```

        FROM NhanVien
        WHERE MaNV = @manv
        if (@macv = 'cv01')
            SET @sqlString = 'ALTER SERVER ROLE sysadmin' + ' ADD MEMBER '
+ @userName;
        else
            SET @sqlString = 'ALTER ROLE Staff ADD MEMBER ' + @userName;
        EXEC (@sqlString)
    END

```

- Procedure xóa nhân viên

```

CREATE PROCEDURE [dbo].[proc_DeleteEmployee]
    @MaNV nchar(10)
AS
BEGIN
    SET NOCOUNT ON;
    DECLARE @username varchar(15);
    SELECT @username=TenDangNhap FROM DANGNHAP WHERE maNV=@maNV
    DECLARE @sql varchar(100)
    DECLARE @SessionID INT;
    SELECT @SessionID = session_id
    FROM sys.dm_exec_sessions
    WHERE login_name = @username;
    IF @SessionID IS NOT NULL
    BEGIN
        SET @sql = 'kill ' + Convert(NVARCHAR(20), @SessionID)
        exec (@sql)
    END
    BEGIN TRANSACTION;
    BEGIN TRY
        DELETE FROM HoaDon WHERE MaNV = @MaNV;
        DELETE FROM BangPhanCa WHERE MaNV = @MaNV;
        DELETE FROM NhanVien WHERE MaNV = @MaNV;
        --
        SET @sql = 'DROP USER ' + @username
        exec (@sql)
        --
        SET @sql = 'DROP LOGIN ' + @username
        exec (@sql)
        --
        DELETE FROM DANGNHAP WHERE maNV=@maNV;
    END TRY
    BEGIN CATCH
        DECLARE @err NVARCHAR(MAX)
        SELECT @err = N'Lỗi ' + ERROR_MESSAGE()
        RAISERROR(@err, 16, 1)
        ROLLBACK TRANSACTION;
        THROW;
    END CATCH
    COMMIT TRANSACTION;
END

```

## CHƯƠNG 5: THIẾT KẾ GIAO DIỆN HỆ THỐNG

### Môi trường cài đặt và công nghệ sử dụng

- Hệ cơ sở dữ liệu Microsoft SQL Server Management Studio 19.
- Chương trình Winform được viết bằng Microsoft Visual Studio 2022 - trên nền tảng ASP.NET Web Application (.NET Framework) C#.

### Giao diện và chức năng của chương trình:

#### 1. Form Home

Cà phê Sơn - Phần mềm quản lý quán cà phê

Home Warehouse Product Receipt Statistic Employees Payment

Cà phê Đá xay Trà Soda Thức uống khác Bánh Full Menu

CAFE LATTE 35,000 VND

CAFE AMERICANO 35,000 VND

CAFE CAPPUCCINO 40,000 VND

CAFE ESPRESSO 35,000 VND

CAFE PHIN ĐEN 35,000 VND

CAFE PHIN SỮA 35,000 VND

**CART**

Số điện thoại:

Tên khách hàng:

Check Customer

Tên sản phẩm	Giá	SL	Tổng giá
--------------	-----	----	----------

NEW Xuất hoá đơn

#### 2. Form đăng nhập

Cà phê Sơn - Phần mềm quản lý quán cà phê

Tài khoản:

Mật khẩu:

☐ Nhớ tài khoản *Vui lòng điền đủ thông tin tài khoản mật khẩu*

**Đăng nhập**

[Quên mật khẩu?](#) [Giới thiệu](#)



- Tạo tài khoản:

Code C#:

```
private void buttonCreate_Click(object sender, EventArgs e)
{
    mydb.openConnectionAdmin();
    SqlCommand cmd = new SqlCommand("proc_ThemTaiKhoan", mydb.getConnectionAdmin());
    cmd.CommandType = CommandType.StoredProcedure;
    cmd.Parameters.AddWithValue("@manv", comboBoxMaNV.SelectedValue.ToString());
    cmd.Parameters.AddWithValue("@tk", textBoxUsername.Text);
    cmd.Parameters.AddWithValue("@mk", textBoxPassword.Text);

    if (cmd.ExecuteNonQuery() > 0)
    {
        MessageBox.Show("Tạo tài khoản thành công");
        mydb.closeConnectionAdmin();
    }
    else
    {
        mydb.closeConnectionAdmin();
    }
}
```

Code SQL:

```
CREATE procedure [dbo].[proc_ThemTaiKhoan] @tk NCHAR(10), @mk NCHAR(10), @manv
NCHAR(10)
AS
BEGIN
    INSERT INTO DANGNHAP VALUES (@tk, @mk, @manv)
END
```

- Kiểm tra đăng nhập:

Code C#:

```
GLOBAL.username = textBoxUser.Text;
GLOBAL.password = textBoxPassword.Text;
SqlCommand cmd = new SqlCommand("SELECT dbo.checkLogin(@user, @pass)",
mydb.getConnectionAdmin());
cmd.Parameters.AddWithValue("@user", textBoxUser.Text);
cmd.Parameters.AddWithValue("@pass", textBoxPassword.Text);
mydb.openConnectionAdmin();

bool count = (bool)cmd.ExecuteScalar();

mydb.closeConnectionAdmin();
```

Code SQL:

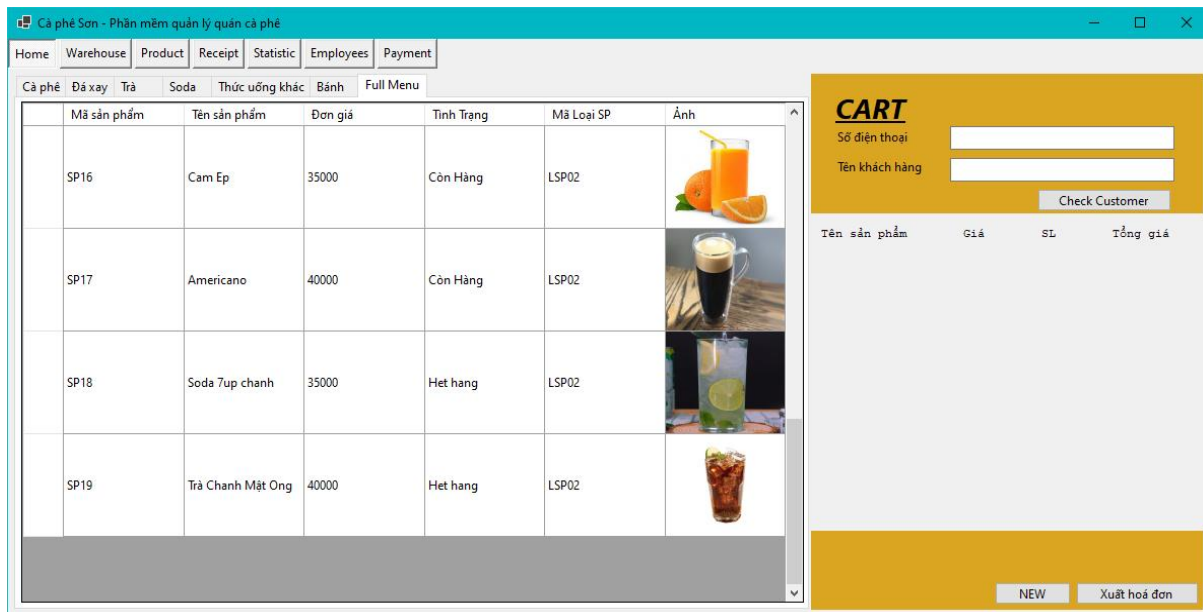
```
CREATE FUNCTION [dbo].[checkLogin] (@username NVARCHAR(MAX), @password
NVARCHAR(MAX))
RETURNS BIT
AS
BEGIN
    DECLARE @result BIT;

    SELECT @result = CAST(COUNT(*) AS BIT)
    FROM DANGNHAP
    WHERE TenDangNhap = @username AND MatKhau = @password;

    RETURN @result;
END;
```

### 3. Các Form Danh mục

### 3.1. Danh mục sản phẩm



Code C# gọi danh mục:

```
public DataTable getDSSanPham()
{
    //Lấy danh mục sản phẩm từ view
    SqlCommand command = new SqlCommand("Select * from view_danhSachSanPham",
mydb.getConnection);
    DataTable table = new DataTable();
    SqlDataAdapter adapter = new SqlDataAdapter(command);
    adapter.Fill(table);
    return table;
}
```

Code tạo view trên SQL:

```
CREATE VIEW [dbo].[view_danhSachSanPham] AS
SELECT *
FROM SanPham
```

### 3.2. Danh mục kho

Cà phê Sơn - Phần mềm quản lý quán cà phê

Home Warehouse Product Receipt Statistic Employees Payment

KHO NGUYÊN LIỆU  SEARCH

	MaNL	TenNL	MaNCC	SL	DonVi	TinhTrang
▶	BANANA	Chuối	STC	100	Kg	Còn hàng
	CAM	CAM	PNC	100	Kg	Còn hàng
	CF	Cà phê	TNC	10	kg	Còn hàng
	DG	Đường	SSS	10	kg	Còn hàng
	SD	Sữa đặc	SSS	10	lít	Còn hàng
	SRD	Syrup Đào	STC	10	lít	Còn hàng
	SRT	Syrup Táo	STC	10	lít	Còn hàng
	SRV	Syrup Vải	STC	10	lít	Còn hàng
	SRVQ	Syrup Việt Quất	STC	10	lít	Còn hàng
	ST	Sữa tươi	SSS	10	lít	Còn hàng

Code C# gọi danh mục:

```
public DataTable xemNguyenLieu()
{
    //Lấy danh sách nguyên liệu trong kho
    SqlCommand command = new SqlCommand("select * from view_NguyenLieu", mydb.getConnection);
    DataTable dt = new DataTable();
    SqlDataAdapter adapter = new SqlDataAdapter(command);
    adapter.Fill(dt);
    return dt;
}
```

Code tạo view trên SQL:

```
CREATE VIEW [dbo].[view_NguyenLieu] AS
SELECT *
FROM NguyenLieu
```

### 3.3. Danh mục hóa đơn

Cà phê Sơn - Phần mềm quản lý quán cà phê

Home Warehouse Product Receipt Statistic Employees Payment

HOÁ ĐƠN From 20/04/2023 to 20/04/2023 SEARCH REFRESH

	Mã hoá đơn	Ngày đặt hàng	Mã khách hàng	Mã nhân viên	Trị giá hoá đơn
▶	1	4/19/2023	1	1	79000
	11	4/19/2023	1	2	37000
	12	4/19/2023	1	2	50000
	13	4/19/2023	1	1	45000
	14	4/19/2023	1	1	35000
	15	4/19/2023	1	2	45000
	16	4/19/2023	1	1	72000
	2	4/19/2023	2	2	124000
	3	4/19/2023	1	1	30000
	34209	4/20/2023	10	1	112000
	4	4/19/2023	1	1	37000
	5	4/19/2023	2	1	15000
	6	4/19/2023	1	2	32000
	7	4/19/2023	1	1	35000
	8	4/19/2023	1	1	35000
	9	4/19/2023	1	1	35000
*					

Code C# gọi danh mục:

```
public void load_tabPage_Receipt()
{
    //Lấy danh mục hóa đơn
    SqlCommand cmd = new SqlCommand("SELECT * FROM vi_HoaDonView", db.getConnection);
    SqlDataAdapter da = new SqlDataAdapter(cmd);
    DataTable dt = new DataTable();
    da.Fill(dt);
    dataGridView3.DataSource = dt;

    //Đổi tên
    dataGridView3.Columns["MaHD"].HeaderText = "Mã hoá đơn";
    dataGridView3.Columns["NgayDatHang"].HeaderText = "Ngày đặt hàng";
    dataGridView3.Columns["MaKH"].HeaderText = "Mã khách hàng";
    dataGridView3.Columns["MaNV"].HeaderText = "Mã nhân viên";
    dataGridView3.Columns["TriGiaHD"].HeaderText = "Trị giá hoá đơn";
}
```

Code tạo view trên SQL:

```
CREATE VIEW [dbo].[vi_HoaDonView] AS
SELECT MaHD, NgayDatHang, MaKH, MaNV, TriGiaHD
FROM HoaDon;
```

### 3.4. Danh mục hóa đơn ứng dụng

	MaHD_UD	NgayDatHang	MaUngDung	MaNV
▶	1	5/9/2023	BM	1
*	2	5/10/2023	GF	2








Code C# gọi danh mục:

```
db.openConnection();
SqlDataAdapter da1 = new SqlDataAdapter("SELECT * FROM vi_HoaDonUngDung", db.getConnection());
DataTable dt1 = new DataTable();
da1.Fill(dt1);
dgvHDUD.DataSource = dt1;
db.closeConnection();
```

Code tạo view trên SQL:

```
CREATE VIEW vi_HoaDonUngDung AS
SELECT MaHD_UD, NgayDatHang, MaUngDung, MaNV, TriGiaHD
FROM HoaDonUngDung;
```

### 3.5. Danh mục nhân viên

Cà phê Sơn - Phần mềm quản lý quán cà phê												
<div> <a href="#">Home</a> <a href="#">Warehouse</a> <a href="#">Product</a> <a href="#">Receipt</a> <a href="#">Statistic</a> <a href="#">Employees</a> <a href="#">Payment</a> </div>												
<div> <a href="#">Thêm</a> <a href="#">Xóa</a> <a href="#">Hiển Thị</a> <a href="#">Phân ca</a> </div> <div> <div>Tìm theo Mã NV</div> <div></div> <div>Tìm Kiếm</div> </div>												
	Mã NV	Họ	Tên	Ngày Sinh	Giới Tính	Địa Chỉ	Số Điện Thoại	Công Việc	Số lượng ca	Tiền Thưởng	Ngày Tuyển Dụng	Hình Ảnh
▶	1	Dang Nhat	Nhat	01/01/2003	Nam	TD	0938951673	Quản Lý	10	0	1/1/2022	
	11	Phu Khang	Khang	19/04/2003	Nam	Thu Duck	0912322222	Quản Lý	0	0	4/19/2023	
	2	Dang Phong	Phong	01/01/2003	Nam	TD	0938951673	Quản Lý	20	0	1/1/2022	
	3	Dang Nhi	Nhi	01/01/2003	Nam	TD	0938951673	Quản Lý	30	0	1/1/2022	
	4	Dang Quy	Quy	01/01/2003	Nam	TD	0938951673	Quản Lý	40	0	1/1/2022	
	5	Dang Phat	Phat	01/01/2003	Nam	TD	0938951673	Quản Lý	50	0	1/1/2022	
	6	Dang Thinh	Thinh	01/01/2003	Nam	TD	0938951673	Quản Lý	60	0	1/1/2022	

Code C# gọi danh mục:

```
public DataTable getAllEmployee()
{
    SqlCommand cmd = new SqlCommand("SELECT * FROM v_LoadEmployee", db.getConnection);
    SqlDataAdapter adapter = new SqlDataAdapter(cmd);
    DataTable dataTable = new DataTable();
    adapter.Fill(dataTable);
    return dataTable;
}
```

Code tạo view trên SQL:

```
CREATE VIEW [dbo].[v_LoadEmployee] AS
SELECT MaNV, HoNV, TenNV, NgaySinh, GioiTinh, DiaChi, SDT, CongViec.TenCV, SoCa,
Thuong, NgayTuyenDung, HinhAnh
FROM NhanVien, CongViec
WHERE NhanVien.MaCV = CongViec.MaCV
```

### 3.6. Danh mục ca làm việc

Phân ca

Mã NV:  Mã Ca:  Thứ:  Ngày làm: 20/04/2023

	MaCa	NgàyTrongTuan	GioBatDau	GioKetThuc
▶	ca1	Chu Nhật	0h	4h
	ca1	Thu 2	0h	4h
	ca1	Thu 3	0h	4h
	ca1	Thu 4	0h	4h

Code C# gọi danh mục:

```
private void displayShift()
{
    SqlCommand cmd = new SqlCommand("SELECT * FROM v_CaLamViec", db.getConnection);
    SqlDataAdapter adapter = new SqlDataAdapter(cmd);
    DataTable dataTable = new DataTable();
    adapter.Fill(dataTable);
    current_DataGridView = 1;
    dataGridView_Division.DataSource = dataTable;
}
```

Code tạo view trên SQL:

```
CREATE VIEW [dbo].[v_CaLamViec]
AS
SELECT *
FROM CaLamViec
```

### 3.7. Danh mục bảng ca làm việc

Phân ca

Mã NV:  Mã Ca:  Thứ:  Ngày làm: 20/04/2023

	MaCa	NgàyTrongTuan	MaNV	HoNV	TenNV	NgàyLam
▶	ca1	Chu Nhat	1	Dang Nhat	Nhat	4/20/2023
	ca1	Thu 2	2	Dang Phong	Phong	3/21/2023
	ca2	Thu 2	3	Dang	Nhi	3/21/2023
	ca3	Thu 2	4	Dang	Quy	3/21/2023

Code C# gọi danh mục:

```
private void displayDivision()
{
    SqlCommand cmd = new SqlCommand("SELECT * FROM v_BangPhanCa", db.getConnection());
    SqlDataAdapter adapter = new SqlDataAdapter(cmd);
    DataTable dataTable = new DataTable();
    adapter.Fill(dataTable);
    current_DataGridView = 2;
    dataGridView_Division.DataSource = dataTable;
}
```

Code tạo view trên SQL:

```
CREATE VIEW [dbo].[v_BangPhanCa]
AS
SELECT NhanVien.MaNV, NhanVien.HoNV, NhanVien.TenNV, BangPhanCa.MaCa,
BangPhanCa.NgayTrongTuan, BangPhanCa.NgayLam
FROM BangPhanCa, NhanVien
WHERE BangPhanCa.MaNV = NhanVien.MaNV
```



### 3.8. Danh mục phiếu chi

Cà phê Sơn - Phần mềm quản lý quán cà phê											
Home	Warehouse	Product	Receipt	Statistic	Employees	Payment					
PHIẾU CHI										REFRESH	XUẤT
	Mã Đơn Nhập Hàng	Ngày nhập hàng	Mã Nhà Cung Cấp	Tình trạng	Mã phiếu chi	Ngày Xuất Phiếu	Mã nguyên liệu	Đơn giá	Số lượng	Đơn vị	Số tiền chi
▶	3	20/04/2023	SSS	Đang giao	3	20/04/2023	CF	10000	10	Gram	1000000
	4	20/04/2023	SSS	Đang giao	4	20/04/2023	SRD	25000	5	Chai	1250000
	5	20/04/2023	STC	Đang giao	5	20/04/2023	SRT	35000	5	Chai	1750000
	6	20/04/2023	TNC	Đang giao	6	20/04/2023	SRD	25000	100	Chai	2500000
	7	20/04/2023	SSS	Đang giao	7	20/04/2023	DG	15000	10	Kg	150000
	8	20/04/2023	SSS	Đang giao	8	20/04/2023	SRV	40000	100	Chai	4000000
*											

Code C# gọi danh mục:

```
private void load_tabPage_Payment()
{
    SqlCommand cmd = new SqlCommand("SELECT * FROM DonNhapHangView", db.getConnection);
    SqlDataAdapter adapter = new SqlDataAdapter(cmd);
    DataTable dt = new DataTable();
    adapter.Fill(dt);
    dataGridViewPayment.DataSource = dt;
}
```

Code tạo view trên SQL:

```
CREATE VIEW [dbo].[DonNhapHangView]
AS
SELECT DNH.MaDonNH, DNH.NgayNH, DNH.MaNCC, DNH.TinhTrang, PC.MaPhieuChi,
PC.NgayXuatPhieu, CTH.MaNL, CTH.DonGia, CTH.SL, CTH.DonVi, PC.SoTienChi
FROM DonNhapHang DNH
LEFT JOIN PhieuChi PC ON DNH.MaDonNH = PC.MaDonNH
LEFT JOIN ChiTietNhapHang CTH ON DNH.MaDonNH = CTH.MaDonNH;
```

### 3.9. Danh mục lương nhân viên

tinhluongForm					
	MaNV	HoNV	TenNV	SoCa	Luong
▶	1	Dang Nhat	Nhat	10	600000000
	11	Phu	Khang	0	0
	2	Dang Phong	Phong	20	1200000000
	3	Dang	Nhi	30	1800000000
	4	Dang	Quy	40	2400000000
	5	Dang	Phat	50	3000000000
	6	Dang	Thinh	60	3600000000
	7	Phong	Quoc	0	0
	8	Dang	Khoa	80	4800000000
*					

Code C# gọi hàm:

```
private void tinhluongForm_Load(object sender, EventArgs e)
{
    MY_DB db = new MY_DB();
    db.openConnection();

    SqlCommand cmd = new SqlCommand("SELECT * FROM func_tinhLuongNV(@thang, @nam)",
    db.getConnection());

    cmd.Parameters.AddWithValue("@thang", int.Parse(DateTime.Today.ToString("MM")));
    cmd.Parameters.AddWithValue("@nam", int.Parse(DateTime.Today.ToString("yyyy")));

    // Create a data adapter to fill a dataset with the results
    SqlDataAdapter da = new SqlDataAdapter(cmd);
    DataSet ds = new DataSet();
    da.Fill(ds);
}
```

```
dataGridViewluong.DataSource = ds.Tables[0];  
db.closeConnection();  
}
```

Code function trên SQL:

```
CREATE FUNCTION func_tinhLuongNV(@thang INT, @nam INT) RETURNS TABLE  
AS RETURN (  
SELECT nv.MaNV, nv.HoNV, nv.TenNV, nv.SoCa,  
(nv.SoCa*cv.Luong*4 + nv.Thuong*(cv.Luong*2)) +  
CASE WHEN dbo.func_tinhDoanhThuThang(@thang, @nam) > 100000000  
THEN dbo.func_tinhDoanhThuThang(@thang, @nam)*0.01  
ELSE 0  
END AS Luong  
FROM NhanVien nv  
INNER JOIN CongViec cv ON nv.MaCV = cv.MaCV  
)
```