

CHƯƠNG I- ĐẶC TẢ HỆ THỐNG

1.1 Mục đích, yêu cầu

1.1.1 Mục đích

Chương trình quản lý tiêm chủng là một chương trình xoay quanh các đối tượng cần quản lý là vaccine, nhân viên trung tâm tiêm chủng và người tham gia tiêm chủng, khi mỗi người dân đến tiêm toàn bộ quá trình đi khám và tiêm quá trình khám, thông tin được cập nhật trên hồ sơ điện tử.

Mục tiêu của đề tài là xây dựng một phần mềm để thực hiện các thao tác trên, trợ giúp việc quản lý tại trung tâm tiêm chủng được nhanh chóng, dễ dàng hơn cũng như đáp ứng được nhu cầu ngày càng cao của người tham gia tiêm chủng. Trong tình hình dịch như hiện nay điều này rất có ý nghĩa, giúp người dân giảm thời gian chờ đợi cũng như hạn chế tiếp xúc, tránh lây lan dịch bệnh.

1.1.2 Yêu cầu

Chương trình thực hiện những yêu cầu sau:

- Cho phép nhân viên đăng nhập và quản lý hệ thống
- Quản lý thông tin về hồ sơ nhân viên
- Quản lý thông tin hồ sơ khách hàng
- Quản lý việc nhập vaccine (có xuất phiếu nhập giúp dễ dàng quản lý)
- Quản lý việc tiêm vaccine: xuất vaccine để tiêm, liều lượng, ngày tiêm nhắc lại, thông tin khách hàng, hóa đơn
- Cập nhật tình trạng các loại vaccine: tên vaccine, nhà sản xuất, ngày sản xuất, hạn sử dụng, số lô, số lượng có sẵn v.v.v...

1.2. Mô tả hệ thống

1.2.1. Cơ cấu tổ chức và sự phân công chia trách nhiệm:

Quản trị: Xem toàn bộ danh sách nhân viên, sửa thông tin nhân viên, quản lý vaccine trong kho, quản lý lịch sử các phiếu tiêm được xuất ra.

Các nhân viên của trung tâm: Chia ra 3 bộ phận:

- Bác sĩ: là người thực hiện khám và tiêm, có quyền truy xuất các thông tin vaccine, quản lý và tạo phiếu tiêm cũng như tra cứu lịch sử tiêm

của khách hàng, thống kê khách hàng sắp đến hạn tiêm và gửi email cho khách hàng, lưu lại và các phiếu này do quản trị quản lý.

- Thu ngân: là người có quyền truy xuất các thông tin vaccine, quản lý và tạo phiếu tiêm, tra cứu lịch sử tiêm của khách hàng, cũng như có quyền tạo hóa đơn thanh toán phiếu tiêm.

- Quản kho: là người có quyền quản lý, nhập xuất vaccine cũng như tra cứu lịch sử tiêm của khách hàng.

1.2.2. Quy trình xử lý:

1. Khảo sát hệ thống.
2. Phân cấp các chức năng trong hệ thống.
3. Xây dựng luồng dữ liệu.
4. Xây dựng các mối quan hệ dữ liệu trong hệ thống.
5. Xây dựng mô hình dữ liệu.
6. Chuẩn hóa mô hình dữ liệu.
7. Thiết kế các bảng vật lý.

1.2.3. Quy tắc nghiệp vụ:

Quy trình xử lý tại trung tâm:

• **Bước 1:** Khách hàng đăng kí tiêm vaccine tại quầy thu ngân. Khách hàng đem phiếu tiêm vừa được tạo rồi đến gặp bác sĩ để xin chữ ký. Nếu vaccine trong kho hết thì không thể tạo được phiếu tiêm mà khách hàng cần quay lại vào lần sau. Nếu khách hàng muốn khám tổng quan trước thì đến gặp bác sĩ khám và tư vấn. Sau đó bác sĩ sẽ tạo phiếu tiêm và ký tên.

• **Bước 2:** Khách hàng được bác sĩ khám, tư vấn và xác nhận phiếu tiêm trước khi tiêm tại phòng khám. Sau đó, khách hàng đem phiếu tiêm đã được xác nhận ra quầy thu ngân để thanh toán và lấy hóa đơn.

• **Bước 3:** Sau khi hoàn thành các thủ tục, y tá tiến hành tiêm vaccine cho khách hàng tại khu vực tiêm.

• **Bước 4:** Theo dõi 30 phút sau khi tiêm và được nhân viên y tế kiểm tra sức khỏe và hẹn ngày tiêm nhắc lại (nếu có) trước khi ra về. Khi gần đến hạn tiêm, trung tâm sẽ gửi mail để nhắc nhở khách hàng.

Tổng tiền cần thanh toán trong hóa đơn:

$$\text{Tổng tiền} = \sum \text{vaccine} * (1 - \text{chiết khấu})$$

1.3. Đặc tả chức năng cần xây dựng

Xây dựng hệ thống dựa trên thực tế các trung tâm tiêm chủng trên cả nước. Hệ thống phải có khả năng đáp ứng những yêu cầu cơ bản như: quản lý được nguồn vào ra của vaccine (bằng cách lưu lịch sử các phiếu nhập, phiếu tiêm của khách hàng), thông báo những vaccine còn lại số lượng ít trong kho cũng như sắp hết hạn sử dụng, tính tổng tiền khách hàng cần trả cho mỗi lượt vào tiêm và cập nhật lại số lượng vaccine sau mỗi lần hoàn thành phiếu tiêm cho khách hàng, tạo phiếu tiêm ban đầu cho khách hàng.

1.3.1. Chức năng xem danh sách vacxin đang có

- Có mục tìm kiếm vaccine theo tên, theo loại
- Hiện thị mã vaccine, nhà sản xuất, ngày sản xuất, hạn sử dụng, số lô, số lượng có sẵn, đơn giá, mã loại vaccine và mã quản kho.

1.3.2. Chức năng tạo và hoàn thành phiếu tiêm cho khách hàng

- Thu ngân nhập các thông tin cá nhân của khách hàng, thông tin người giám hộ (nếu có), thông tin vaccine tiêm, ngày tiêm, mã bác sĩ tiêm.
- Chuyển phiếu tiêm đến cho bác sĩ, bác sĩ điền các lưu ý vào mục ghi chú trong phiếu tiêm và tiến hành tiêm cho khách hàng. Sau đó thì nhấn hoàn thành để lưu phiếu tiêm vào danh sách phiếu tiêm và cập nhật số lượng vaccine.

1.3.3. Chức năng hóa đơn cho khách hàng

- Sau khi tiêm xong, khách hàng sẽ đem phiếu tiêm đến lại quầy thu ngân để tiến hành thanh toán.
- Sau khi hoàn thành, hóa đơn sẽ được lưu lại để quản lý.

1.3.4. Chức năng xem lại danh sách phiếu tiêm và hóa đơn đã được xử lý (chỉ dành cho quản trị)

Giúp người quản trị quản lý được ai đã thực hiện những công việc gì, để trong quá trình làm việc xảy ra vấn đề gì có thể dễ dàng xử lý.

1.3.5. Chức năng thống kê

- Quản trị: Thống kê doanh thu, top vaccine dùng nhiều, ...
- Bác sĩ: Thống kê khách hàng đến hạn tiêm

1.3.6. Chức năng quản lý nhân viên

- Thêm, sửa, xóa tài khoản nhân viên.
- Phân quyền cho các nhân viên theo chức vụ.

1.3.7. Chức năng xuất báo cáo, hóa đơn, phiếu tiêm

- Cho phép xuất phiếu tiêm, hóa đơn, báo cáo ra định dạng pdf ,excel, word,.. hoặc in ra bản cứng.

1.3.8. Chức năng gửi email cho khách hàng

- Cho phép bác sĩ gửi email thông báo tới các khách hàng khi sắp đến hạn tiêm mũi tiếp theo.

1.4. Phân chia công việc nhóm

Phụ trách	Công việc
Trần Bảo Trung	<ul style="list-style-type: none">- Phân quyền cho các tài khoản đăng nhập- Chức năng thanh toán hóa đơn- Thiết kế các thống kê cho bác sĩ- Tính năng xuất hóa đơn, phiếu tiêm,...
Đinh Thị Thu Uyên	<ul style="list-style-type: none">- Thiết kế tạo phiếu tiêm- Chức năng quản lý khách hàng- Chức năng nhập xuất vaccine- Thống nhất cơ sở dữ liệu cho cả nhóm
Phạm Đình Khương Duy	<ul style="list-style-type: none">- Chức năng quản lý tài khoản trung tâm- Chức năng quản lý người giám hộ- Thiết kế các thống kê cho quản trị- Chức năng tra cứu vaccine
Hồ Nguyễn Nguyễn	<ul style="list-style-type: none">- Chức năng quản lý lịch sử tiêm khách hàng- Xây dựng tính năng gửi email cho khách hàng- Nhập dữ liệu mẫu cho demo

CHƯƠNG II- PHÂN TÍCH VÀ CÀI ĐẶT DỮ LIỆU HỆ THỐNG

2.1. Mô tả hệ thống dưới dạng mối quan hệ dữ liệu

* Đặc tả vấn đề: Một trung tâm tiêm chủng có nhu cầu tạo 1 phần mềm giúp việc quản lí trở nên dễ dàng và tiết kiệm thời gian.

* Đối với khách hàng, chỉ cần mã khách hàng là có thể xem các thông tin liên quan đến khách hàng đó trong chức năng lịch sử tiêm. Khách hàng không có tài khoản (tên đăng nhập, mật khẩu) như các thành viên làm việc cho trung tâm tiêm chủng.

Khách hàng mới đăng kí thông tin tại quầy thu ngân trong chức năng tạo phiếu tiêm với các thông tin: Tên khách hàng, ngày sinh, giới tính, tiền sử bệnh, ngày tiêm, mã bác sĩ, mã vaccine, mũi thứ, liều lượng, nhắc lại sau bao nhiêu tháng.

Sau khi đã được đăng kí thông tin, những lần tiêm tiếp theo khách hàng chỉ cần điền các thông tin sau để tạo phiếu tiêm: mã vaccine, mũi thứ, liều lượng, nhắc lại sau bao nhiêu tháng.

Thu ngân hoặc bác sĩ đều có thể tạo phiếu tiêm cho khách hàng nhưng chỉ có thu ngân mới có thể thực hiện chức năng thanh toán và in hóa đơn cho khách hàng.

* Đối với đối tượng thu ngân, quản lý kho, quản trị viên sẽ đăng kí tài khoản với các thông tin như sau: mã thành viên, họ và tên, ngày sinh, địa chỉ, số điện thoại. Riêng đối với đối tượng bác sĩ sẽ có thêm thông tin bằng cấp, chuyên khoa.

Với từng chức vụ khác nhau, mỗi nhân viên sẽ có những sự khác nhau nhất định trong các chức năng của phần mềm, cụ thể như sau:

+ Giống: Tất cả các tài khoản sau khi đăng nhập đều có thể tra cứu thông tin vắc xin trung tâm đang quản lý, lịch sử tiêm của khách hàng.

+ Khác:

- Thu ngân: Có thêm các chức năng tạo phiếu tiêm, thanh toán hóa đơn của khách hàng.

- Bác sĩ: Có thêm chức năng tạo phiếu tiêm giống như với đối tượng thu ngân, gửi email cho khách hàng đến hạn tiêm.

- Quản kho: Trong phần quản lý vắc xin có thêm chức năng quản lý nhập kho.

Nếu vacxin thuộc loại chưa có trong danh sách sẽ thêm loại đó vào danh sách các loại vacxin với các thông tin: mã loại vaccine, tên loại vaccine

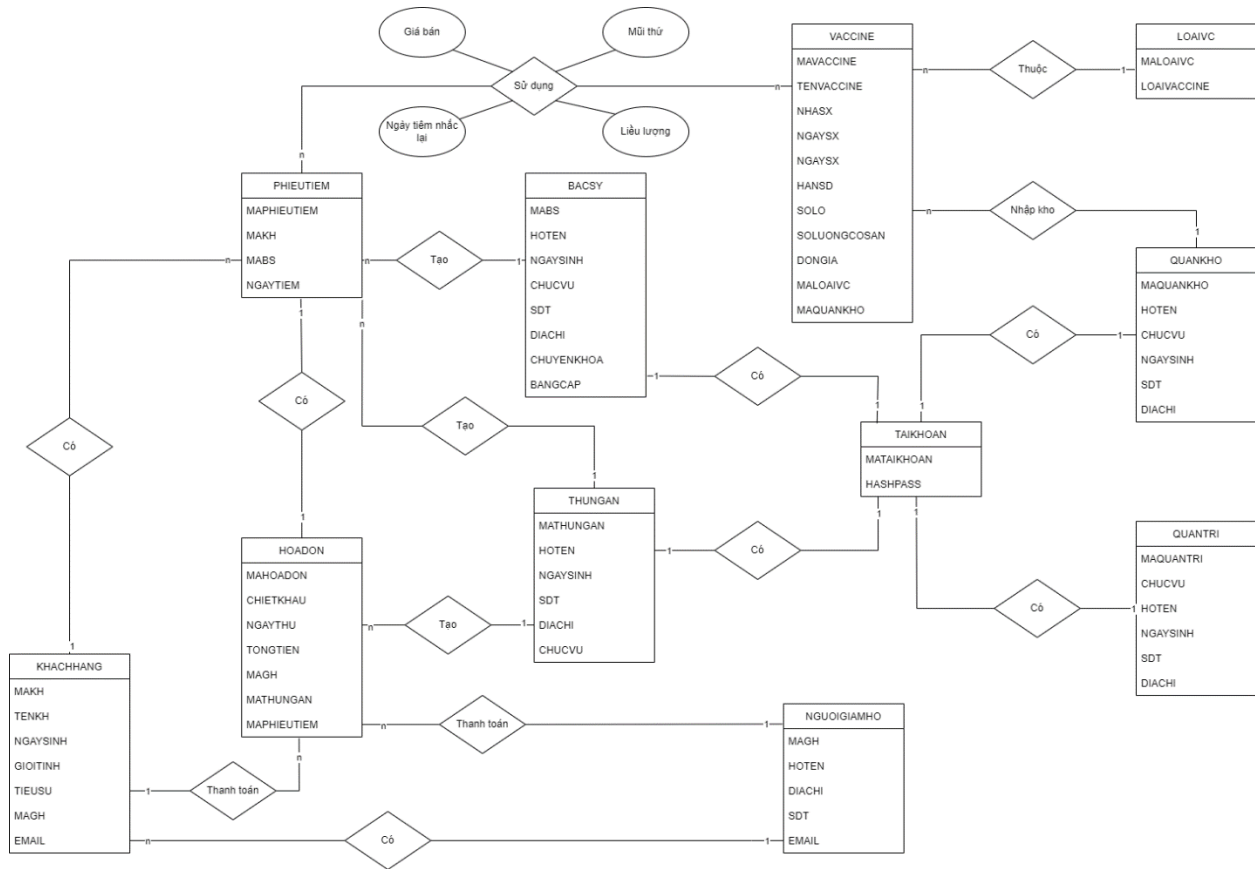
Nếu vacxin thuộc loại đã có, nhập thông tin vaccine với các thông tin: Tên vaccine, mã vaccine (bỏ trống nếu là vaccine mới), nhà sản xuất, số hiệu lô, số lượng, ngày sản xuất, đơn giá, hạn sử dụng, loại vaccine

- Quản trị viên: Có thể xem được tất cả các chức năng quản lý thông tin của tất cả các nhân viên trong trung tâm. Nhưng không thể thanh toán hóa đơn cho khách hàng (chỉ có thu ngân mới có thể thực hiện).

Quản trị viên có thể xem thống kê doanh thu, thống kê vaccine của trung tâm tiêm chủng.

Ngoài ra quản trị viên có thêm chức năng tạo tài khoản cho tất cả các thành viên trong trung tâm tiêm chủng với các thông tin tương ứng với từng đối tượng.

2.2. Xây dựng mô hình liên kết thực thể



Mô hình liên kết thực thể

2.3. Cách chuyển đổi từ mô hình liên kết thực thể sang mô hình quan hệ

Mô Hình E-R	Mô Hình CSDL Quan Hệ
Một Tập Thực Thể	Một Lược Đồ Quan Hệ
Mỗi Thuộc Tính	Một Thuộc Tính
Mỗi Thuộc Tính Nhận Diện	Khóa Chính
Mỗi mối kết hợp	Khóa Ngoại / LỆ Quan Hệ mới (trong mkh M-N)

Sơ đồ chuyển đổi từ mô hình ER sang mô hình quan hệ

Các quy tắc chuyển đổi:

- Quy tắc 1: Với kiểu liên kết 1:1
 - + Cách 1: Chuyển khóa chính của lược đồ quan hệ này sang làm khóa ngoại của lược đồ quan hệ kia và ngược lại
 - + Cách 2: Nhập 2 kiểu thực thể và mối liên kết thành một lược đồ quan hệ, chọn khóa chính cho phù hợp
 - Quy tắc 2: Với kiểu liên kết 1:n
 - Chuyển khóa chính của lược đồ quan hệ bên 1 sang làm khóa ngoại của lược đồ quan hệ bên n
 - Quy tắc 3: Với kiểu liên kết n:n
 - Chuyển mỗi liên kết thành một lược đồ quan hệ có thuộc tính là thuộc tính của mỗi liên kết, thêm các thuộc tính khóa chính
- Từ các quy tắc trên, ta xây dựng được lược đồ quan hệ theo các bước như sau:

Bước 1: Chuyển các tập thực thể thành quan hệ

KHACHHANG(MAKH, TENKH, NGAYSINH, GIOITINH, TIEUSU, MAGH, EMAIL)

PHIEUTIAM(MAPHIEUTIAM, MAKH, MABS, NGAYTIEM)

VACCINE(MAVACCINE, TENVACCINE, NHASX, NGAYSX, HANSĐ, SOLO, SOLUONGCOSAN, ĐONGIA, MALOAIVC, MAQUANKHO)

LOAIVC(MALOAIVC, LOAIVACCINE)

QUANKHO(MAQUANKHO, HOTEN, CHUCVU, NGAYSINH, SĐT, DIACHI)

BACSY(MABS, HOTEN, NGAYSINH, CHUCVU, SĐT, DIACHI, CHUYENKHOA, BANGCAP)

TAIKHOAN(MATAIKHOAN, HASHPASS)

THUNGAN(MATHUNGAN, HOTEN, NGAYSINH, SĐT, DIACHI, CHUCVU)

HOADON(MAHOADON, CHIETKHAU, NGAYTHU, TONGTIEN, MAGH, MATHUNGAN, MAPHIEUTIAM)

NGUOIGIAMHO(MAGH, HOTEN, DIACHI, SĐT, EMAIL)

QUANTRI(MAQUANTRI,CHUCVU,HOTEN,NGAYSINH,SDT,DIACHI)

Bước 2: Chuyển các mối liên kết

Mối kết hợp PHIEUTIEU – VACCINE chuyển thành một quan hệ mới:

CHITIEUTIEU(MAPHIEUTIEU, MAVACCINE, GIABAN, MUI THU, NGAYTIEMNHACLAU, LIEULUONG)

2.4. Phân tích nhu cầu sử dụng thông tin

- KHACHHANG(MAKH, TENKH, NGAYSINH, GIOITINH, TIEUSU, MAGH, EMAIL)

- + Thuộc 1NF vì miền giá trị của các thuộc tính chỉ chứa giá trị đơn, không phân chia được và giá trị của mỗi thuộc tính cũng là một giá trị đơn lấy từ miền giá trị của nó

- + Thuộc 2NF vì thỏa mãn 1NF, không tồn tại phụ thuộc hàm vào một phần của khóa nên không vi phạm 2NF

- + Thuộc 3NF vì thỏa mãn 2NF, không có phụ thuộc hàm có nguồn là thuộc tính không khóa, đích là thuộc tính không khóa

- + Thuộc BCNF vì thỏa mãn 3NF, không có thuộc tính nào phụ thuộc hàm vào thuộc tính không khóa

- PHIEUTIEU(MAPHIEUTIEU, MAKH, MABS, NGAYTIEM)

- + Thuộc 1NF vì miền giá trị của các thuộc tính chỉ chứa giá trị đơn, không phân chia được và giá trị của mỗi thuộc tính cũng là một giá trị đơn lấy từ miền giá trị của nó

- + Thuộc 2NF vì thỏa mãn 1NF, không tồn tại phụ thuộc hàm vào một phần của khóa nên không vi phạm 2NF

- + Thuộc 3NF vì thỏa mãn 2NF, không có phụ thuộc hàm có nguồn là thuộc tính không khóa, đích là thuộc tính không khóa

- + Thuộc BCNF vì thỏa mãn 3NF, không có thuộc tính nào phụ thuộc hàm vào thuộc tính không khóa

- VACCINE(MAVACCINE, TENVACCINE, NHASX, NGAYSX, HANSĐ, SOLO, SOLUONGCOSAN, DONGIA, MALOAIVC, MAQUANKHO)

- + Thuộc 1NF vì miền giá trị của các thuộc tính chỉ chứa giá trị đơn, không phân chia được và giá trị của mỗi thuộc tính cũng là một giá trị đơn lấy từ miền giá trị của nó

- + Thuộc 2NF vì thỏa mãn 1NF, không tồn tại phụ thuộc hàm vào một phần của khóa nên không vi phạm 2NF

- + Thuộc 3NF vì thỏa mãn 2NF, không có phụ thuộc hàm có nguồn là thuộc tính không khóa, đích là thuộc tính không khóa
- + Thuộc BCNF vì thỏa mãn 3NF, không có thuộc tính nào phụ thuộc hàm vào thuộc tính không khóa
 - LOAIVC(MALOAIVC, LOAIVACCINE)
 - + Thuộc 1NF vì miền giá trị của các thuộc tính chỉ chứa giá trị đơn, không phân chia được và giá trị của mỗi thuộc tính cũng là một giá trị đơn lấy từ miền giá trị của nó
 - + Thuộc 2NF vì thỏa mãn 1NF, không tồn tại phụ thuộc hàm vào một phần của khóa nên không vi phạm 2NF
 - + Thuộc 3NF vì thỏa mãn 2NF, không có phụ thuộc hàm có nguồn là thuộc tính không khóa, đích là thuộc tính không khóa
 - + Thuộc BCNF vì thỏa mãn 3NF, không có thuộc tính nào phụ thuộc hàm vào thuộc tính không khóa
 - QUANKHO(MAQUANKHO, HOTEN, CHUCVU, NGAYSINH, SDT, DIACHI)
 - + Thuộc 1NF vì miền giá trị của các thuộc tính chỉ chứa giá trị đơn, không phân chia được và giá trị của mỗi thuộc tính cũng là một giá trị đơn lấy từ miền giá trị của nó
 - + Thuộc 2NF vì thỏa mãn 1NF, không tồn tại phụ thuộc hàm vào một phần của khóa nên không vi phạm 2NF
 - + Thuộc 3NF vì thỏa mãn 2NF, không có phụ thuộc hàm có nguồn là thuộc tính không khóa, đích là thuộc tính không khóa
 - + Thuộc BCNF vì thỏa mãn 3NF, không có thuộc tính nào phụ thuộc hàm vào thuộc tính không khóa
 - BACSY(MABS, HOTEN, NGAYSINH, CHUCVU, SDT, DIACHI, CHUYENKHOA, BANGCAP)
 - + Thuộc 1NF vì miền giá trị của các thuộc tính chỉ chứa giá trị đơn, không phân chia được và giá trị của mỗi thuộc tính cũng là một giá trị đơn lấy từ miền giá trị của nó
 - + Thuộc 2NF vì thỏa mãn 1NF, không tồn tại phụ thuộc hàm vào một phần của khóa nên không vi phạm 2NF
 - + Thuộc 3NF vì thỏa mãn 2NF, không có phụ thuộc hàm có nguồn là thuộc tính không khóa, đích là thuộc tính không khóa

+ Thuộc BCNF vì thỏa mãn 3NF, không có thuộc tính nào phụ thuộc hàm vào thuộc tính không khóa

○ TAIKHOAN(MATAIKHOAN,HASHPASS)

+ Thuộc 1NF vì miền giá trị của các thuộc tính chỉ chứa giá trị đơn, không phân chia được và giá trị của mỗi thuộc tính cũng là một giá trị đơn lấy từ miền giá trị của nó

+ Thuộc 2NF vì thỏa mãn 1NF, không tồn tại phụ thuộc hàm vào một phần của khóa nên không vi phạm 2NF

+ Thuộc 3NF vì thỏa mãn 2NF, không có phụ thuộc hàm có nguồn là thuộc tính không khóa, đích là thuộc tính không khóa

+ Thuộc BCNF vì thỏa mãn 3NF, không có thuộc tính nào phụ thuộc hàm vào thuộc tính không khóa

○ THUNGAN(MATHUNGAN, HOTEN, NGAYSINH, SDT, DIACHI, CHUCVU)

+ Thuộc 1NF vì miền giá trị của các thuộc tính chỉ chứa giá trị đơn, không phân chia được và giá trị của mỗi thuộc tính cũng là một giá trị đơn lấy từ miền giá trị của nó

+ Thuộc 2NF vì thỏa mãn 1NF, không tồn tại phụ thuộc hàm vào một phần của khóa nên không vi phạm 2NF

+ Thuộc 3NF vì thỏa mãn 2NF, không có phụ thuộc hàm có nguồn là thuộc tính không khóa, đích là thuộc tính không khóa

+ Thuộc BCNF vì thỏa mãn 3NF, không có thuộc tính nào phụ thuộc hàm vào thuộc tính không khóa

○ HOADON(MAHOADON, CHIETKHAU, NGAYTHU, TONGTIEN, MAGH, MATHUNGAN, MAPHIEUTIEM)

+ Thuộc 1NF vì miền giá trị của các thuộc tính chỉ chứa giá trị đơn, không phân chia được và giá trị của mỗi thuộc tính cũng là một giá trị đơn lấy từ miền giá trị của nó

+ Thuộc 2NF vì thỏa mãn 1NF, không tồn tại phụ thuộc hàm vào một phần của khóa nên không vi phạm 2NF

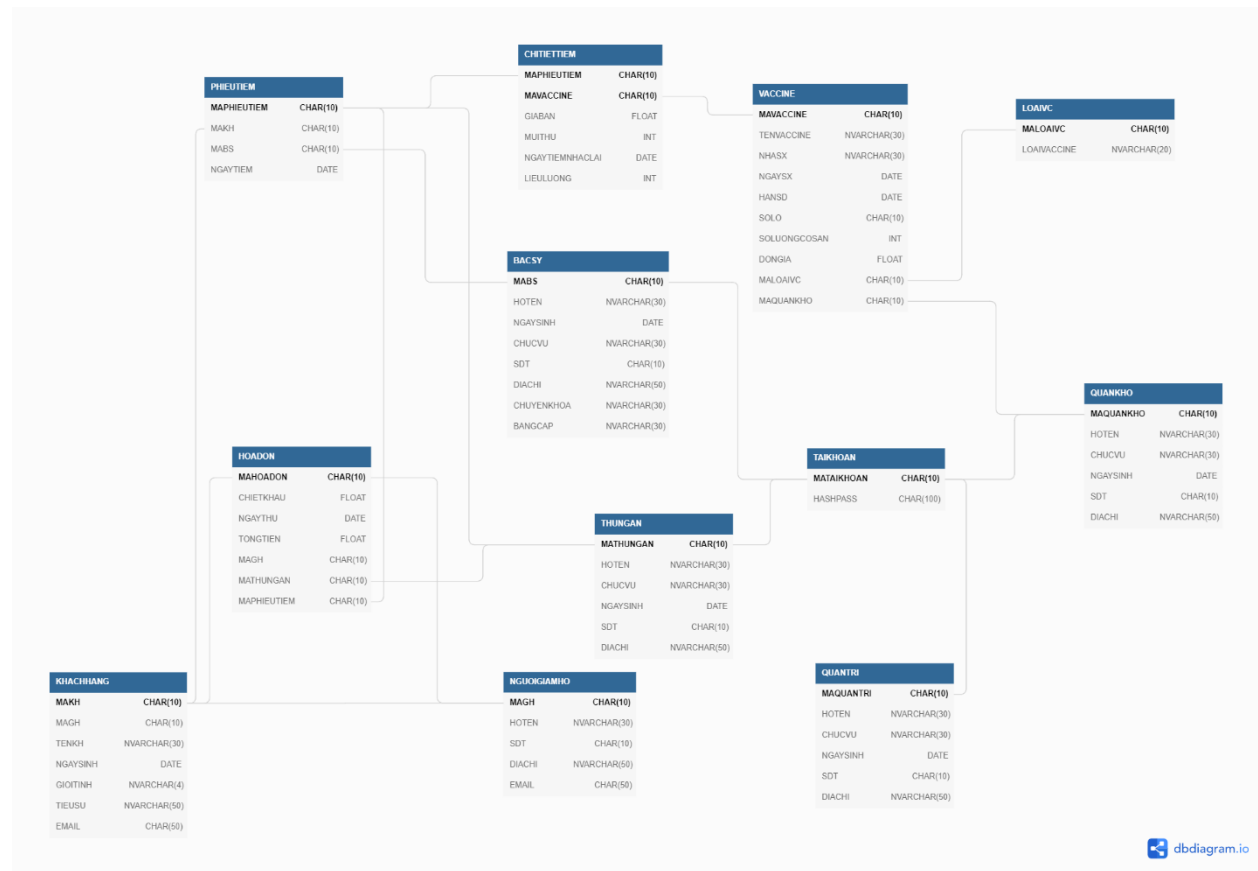
+ Thuộc 3NF vì thỏa mãn 2NF, không có phụ thuộc hàm có nguồn là thuộc tính không khóa, đích là thuộc tính không khóa

+ Thuộc BCNF vì thỏa mãn 3NF, không có thuộc tính nào phụ thuộc hàm vào thuộc tính không khóa

○ NGUOIGIAMHO(MAGH,HOTEN,DIACHI,SDT,EMAIL)

- + Thuộc 1NF vì miền giá trị của các thuộc tính chỉ chứa giá trị đơn, không phân chia được và giá trị của mỗi thuộc tính cũng là một giá trị đơn lấy từ miền giá trị của nó
- + Thuộc 2NF vì thỏa mãn 1NF, không tồn tại phụ thuộc hàm vào một phần của khóa nên không vi phạm 2NF
- + Thuộc 3NF vì thỏa mãn 2NF, không có phụ thuộc hàm có nguồn là thuộc tính không khóa, đích là thuộc tính không khóa
- + Thuộc BCNF vì thỏa mãn 3NF, không có thuộc tính nào phụ thuộc hàm vào thuộc tính không khóa
- QUANTRI(MAQUANTRI,CHUCVU,HOTEN,NGAYSINH,SDT,DIACHI)
 - + Thuộc 1NF vì miền giá trị của các thuộc tính chỉ chứa giá trị đơn, không phân chia được và giá trị của mỗi thuộc tính cũng là một giá trị đơn lấy từ miền giá trị của nó
 - + Thuộc 2NF vì thỏa mãn 1NF, không tồn tại phụ thuộc hàm vào một phần của khóa nên không vi phạm 2NF
 - + Thuộc 3NF vì thỏa mãn 2NF, không có phụ thuộc hàm có nguồn là thuộc tính không khóa, đích là thuộc tính không khóa
 - + Thuộc BCNF vì thỏa mãn 3NF, không có thuộc tính nào phụ thuộc hàm vào thuộc tính không khóa
- CHITIETTIEM(MAPHIEUTIEM, MAVACCINE, GIABAN, MUITHU, NGAYTIEMNHACLAI, LIEULUONG)
 - + Thuộc 1NF vì miền giá trị của các thuộc tính chỉ chứa giá trị đơn, không phân chia được và giá trị của mỗi thuộc tính cũng là một giá trị đơn lấy từ miền giá trị của nó
 - + Thuộc 2NF vì thỏa mãn 1NF, không tồn tại phụ thuộc hàm vào một phần của khóa nên không vi phạm 2NF
 - + Thuộc 3NF vì thỏa mãn 2NF, không có phụ thuộc hàm có nguồn là thuộc tính không khóa, đích là thuộc tính không khóa
 - + Thuộc BCNF vì thỏa mãn 3NF, không có thuộc tính nào phụ thuộc hàm vào thuộc tính không khóa

2.5. Vẽ mô hình quan hệ



2.6. Đặc tả các bảng

Chúng ta có tổng cộng 12 bảng là: CHITTIETTIEM, PHIEUTTIEM, VACCINE, QUANKHO, LOAIVC, BACSY, HOADON, THUNGAN, TAIKHOAN, NGUOIGIAMHO, KHACHHANG, QUANTRI

PHIEUTTIEM					
Trường	Kiểu	Index	NULL	Default	Chú thích
<u>MAPHIEUTTIEM</u>	CHAR(10)	PRIMARY	NO		Mã PT
MAKH	CHAR(10)	REFERENCES	YES		Mã KH
MABS	CHAR(10)	REFERENCES	YES		Mã BS
NGAYTIEM	DATE		YES		Ngày tiêm

CHITIETTIEM					
<i>Trường</i>	<i>Kiểu</i>	<i>Index</i>	<i>NULL</i>	<i>Default</i>	<i>Chú thích</i>
<u>MAPHIEUTTIEM</u>	CHAR(10)	REFERENCES	NO		Mã PT
<u>MAVACCINE</u>	CHAR(10)	REFERENCES	NO		Mã VC
GIABAN	FLOAT		YES		Giá bán
MUTHU	INT		YES		Mũi thứ
NGAYTIEMNHACLAI	DATE		YES		Ngày tiêm nhắc lại
LIEULUONG	INT		YES		Liều lượng

VACCINE					
<i>Trường</i>	<i>Kiểu</i>	<i>Index</i>	<i>NULL</i>	<i>Default</i>	<i>Chú thích</i>
<u>MAVACCINE</u>	CHAR(10)	PRIMARY	NO		Mã VC
TENVACCINE	NVARCHAR(30)		YES		Tên VC
NHASX	NVARCHAR(30)		YES		Nhà SX
NGAYSX	DATE		YES		Ngày SX
HANSĐ	DATE		YES		Hạn sử dụng
SOLO	CHAR(10)		YES		Số lô
SOLUONGCOSAN	INT		YES		Số lượng có sẵn
DONGIA	FLOAT		YES		Đơn giá

MALOAIVC	CHAR(10)	REFERENCES	YES		Mã loại vắc xin
MAQUANKHO	CHAR(10)	REFERENCES	YES		Mã quản kho

QUANKHO					
<i>Trường</i>	<i>Kiểu</i>	<i>Index</i>	<i>NULL</i>	<i>Default</i>	<i>Chú thích</i>
<u>MAQUANKHO</u>	CHAR(10)	PRIMARY	NO		Mã quản kho
HOTEN	NVARCHAR(30)		YES		Họ tên
CHUCVU	NVARCHAR(30)		YES		Chức vụ
NGAYSINH	DATE		YES		Ngày sinh
SDT	CHAR(10)		YES		Số điện thoại
DIACHI	NVARCHAR(50)		YES		Địa chỉ

LOAIVC					
<i>Trường</i>	<i>Kiểu</i>	<i>Index</i>	<i>NULL</i>	<i>Default</i>	<i>Chú thích</i>
<u>MALOAIVC</u>	CHAR(10)	PRIMARY	NO		Mã loại vắc xin
LOAIVACCINE	NVARCHAR(20)		YES		Loại vắc xin

BACSY					
<i>Trường</i>	<i>Kiểu</i>	<i>Index</i>	<i>NULL</i>	<i>Default</i>	<i>Chú thích</i>
<u>MABS</u>	CHAR(10)	PRIMARY	NO		Mã bác sĩ
HOTEN	NVARCHAR(30)		YES		Họ tên
NGAYSINH	DATE		YES		Ngày sinh
CHUCVU	NVARCHAR(30)		YES		Chức vụ
SDT	CHAR(10)		YES		Số điện thoại

DIACHI	NVARCHAR(50)		YES		Địa chỉ
CHUYENKHOA	NVARCHAR(30)		YES		Chuyên khoa
BANGCAP	NVARCHAR(30)		YES		Bằng cấp

TAIKHOAN					
<i>Trường</i>	<i>Kiểu</i>	<i>Index</i>	<i>NULL</i>	<i>Default</i>	<i>Chú thích</i>
<u>MATAIKHOAN</u>	CHAR(10)	PRIMARY	NO		Mã tài khoản
HASHPASS	CHAR(100)		YES		Mã hash mật khẩu

THUNGAN					
<i>Trường</i>	<i>Kiểu</i>	<i>Index</i>	<i>NULL</i>	<i>Default</i>	<i>Chú thích</i>
<u>MATHUNGAN</u>	CHAR(10)	PRIMARY	NO		Mã thu ngân
HOTEN	NVARCHAR(30)		YES		Họ tên
CHUCVU	NVARCHAR(30)		YES		Chức vụ
NGAYSINH	DATE		YES		Ngày sinh
SDT	CHAR(10)		YES		Số điện thoại
DIACHI	NVARCHAR(50)		YES		Địa chỉ

HOADON					
<i>Trường</i>	<i>Kiểu</i>	<i>Index</i>	<i>NULL</i>	<i>Default</i>	<i>Chú thích</i>
<u>MAHOADON</u>	CHAR(10)	PRIMARY	NO		Mã hóa đơn
CHIETKHAU	FLOAT		YES		Chiết khấu
NGAYTHU	DATE		YES		Ngày thu
TONGTIEN	FLOAT		YES		Tổng tiền
MAGH	CHAR(10)	REFERENCES	YES		Mã giám hộ
MATHUNGAN	CHAR(10)	REFERENCES	YES		Mã thu ngân
MAPHIEUTIEM	CHAR(10)	REFERENCES	YES		Mã phiếu tiêm

NGUOIGIAMHO					
<i>Trường</i>	<i>Kiểu</i>	<i>Index</i>	<i>NULL</i>	<i>Default</i>	<i>Chú thích</i>
<u>MAGH</u>	CHAR(10)	PRIMARY	NO		Mã giám hộ
HOTEN	NVARCHAR(30)		YES		Họ tên
SDT	CHAR(10)		YES		Số điện thoại
DIACHI	NVARCHAR(50)		YES		Địa chỉ
EMAIL	CHAR(50)		YES		Email

KHACHHANG					
<i>Trường</i>	<i>Kiểu</i>	<i>Index</i>	<i>NULL</i>	<i>Default</i>	<i>Chú thích</i>
<u>MAKH</u>	CHAR(10)	PRIMARY	NO		Mã thu ngân
MAGH	CHAR(10)	REFERENCES	YES		Mã giám hộ
TENKH	NVARCHAR(30)		YES		Tên khách hàng
NGAYSINH	DATE		YES		Ngày sinh
GIOITINH	NVARCHAR(4)		YES		Giới tính
TIEUSU	NVARCHAR(50)		YES		Tiểu sử bệnh
EMAIL	CHAR(50)		YES		Email

QUANTRI					
<i>Trường</i>	<i>Kiểu</i>	<i>Index</i>	<i>NULL</i>	<i>Default</i>	<i>Chú thích</i>
<u>MAQUANTRI</u>	CHAR(10)	PRIMARY	NO		Mã quản trị
HOTEN	NVARCHAR(30)		YES		Họ tên
CHUCVU	NVARCHAR(30)		YES		Chức vụ
NGAYSINH	DATE		YES		Ngày sinh
SDT	CHAR(10)		YES		Số điện thoại
DIACHI	NVARCHAR(50)		YES		Địa chỉ

CHƯƠNG III: KHAI THÁC CƠ SỞ DỮ LIỆU

3.1. Cài đặt cơ sở dữ liệu

3.1.1. Các câu lệnh cài đặt cơ sở dữ liệu

Cài đặt bảng khách hàng

```
CREATE TABLE KHACHHANG(  
    MAKH CHAR(10) NOT NULL,  
    TENKH NVARCHAR(30),  
    NGAYSINH DATE,  
    EMAIL CHAR(50),  
    GIOITINH NVARCHAR(4),  
    TIEUSU NVARCHAR(50),  
    MAGH CHAR(10)  
)  
CREATE TABLE KHACHHANG ADD CONSTRAINT KH PRIMARY KEY (MAKH)  
CREATE TABLE KHACHHANG ADD CONSTRAINT KH_GH FOREIGN KEY (MAGH)  
REFERENCES NGUOIGIAMHO(MAGH)
```

Cài đặt bảng người giám hộ

```
CREATE TABLE NGUOIGIAMHO(  
    MAGH CHAR(10) NOT NULL,  
    HOTEN NVARCHAR(30),  
    DIACHI NVARCHAR(30),  
    SDT CHAR(10),  
    EMAIL CHAR(50),  
)  
CREATE TABLE NGUOIGIAMHO ADD CONSTRAINT NGH PRIMARY KEY (MAGH)
```

Cài đặt bảng hóa đơn

```
CREATE TABLE HOADON(  
    MAHOADON CHAR(10) ,  
    CHIETKHAU FLOAT,  
    NGAYTHU DATE,  
    TONGTIEN FLOAT,  
    MAGH CHAR(10) REFERENCES NGUOIGIAMHO(MAGH),  
    MATHUNGAN CHAR(10) REFERENCES THUNGAN(MATHUNGAN),  
    MAPHIEUTIEM CHAR(10) REFERENCES PHIEUTIEM(MAPHIEUTIEM),  
)
```

```

CREATE TABLE HOADON ADD CONSTRAINT MHD PRIMARY KEY (MAHOADON)
CREATE TABLE HOADON ADD CONSTRAINT MHD_GH FOREIGN KEY (MAGH)
REFERENCES NGUOIGIAMHO(MAGH)
CREATE TABLE HOADON ADD CONSTRAINT MHD_TN FOREIGN KEY
(MATHUNGAN) REFERENCES THUNGAN(MATHUNGAN)
CREATE TABLE HOADON ADD CONSTRAINT MHD_PT FOREIGN KEY
(MAPHIEUTIEM) REFERENCES PHIEUTIEM(MAPHIEUTIEM)

```

Cài đặt bảng phiếu tiêm

```

CREATE TABLE PHIEUTIEM(
MAPHIEUTIEM CHAR(10) PRIMARY KEY,
MAKH CHAR(10) REFERENCES KHACHHANG(MAKH) ,
MABS CHAR(10), REFERENCES BACSI(MABS),
NGAYTIEM DATE
)
CREATE TABLE PHIEUTIEM ADD CONSTRAINT PT_KH FOREIGN KEY (MAKH)
REFERENCES KHACHHANG(MAKH)
CREATE TABLE PHIEUTIEM ADD CONSTRAINT PT_BS FOREIGN KEY (MABS)
REFERENCES BACSY(MABS)

```

Cài đặt bảng chi tiết tiêm

```

CREATE TABLE CHITIETTIEM(
MAPHIEUTIEM CHAR(10),
MAVACCINE CHAR(10),
GIABAN FLOAT,
MUITHU INT,
NGAYTIEMNHACLAIR DATE,
LIEULUONG INT,
TIEMNHACLAIR BIT
)
CREATE TABLE CHITIETTIEM ADD CONSTRAINT ctt PRIMARY KEY
(MAPHIEUTIEM, MAVACCINE)
CREATE TABLE CHITIETTIEM ADD CONSTRAINT CTT_PT FOREIGN KEY
(MAPHIEUTIEM) REFERENCES PHIEUTIEM(MAPHIEUTIEM)
CREATE TABLE CHITIETTIEM ADD CONSTRAINT CTT_VC FOREIGN KEY
(MAVACCINE) REFERENCES VACCINE(MAVACCINE)

```

Cài đặt bảng bác sỹ

```

CREATE TABLE BACSY(

```

```

MABS CHAR(10) PRIMARY KEY,
CHUCVU NVARCHAR(30),
HOTEN NVARCHAR(30),
NGAYSINH DATE,
SĐT CHAR(10),
DIACHI NVARCHAR(50),
CHUYENKHOA NVARCHAR(30),
BANGCAP NVARCHAR(30)
)
CREATE TABLE BACSY ADD CONSTRAINT BS_TK FOREIGN KEY (MABS)
REFERENCES TAIKHOAN(MATAIKHOAN)

```

Cài đặt bảng thu ngân

```

CREATE TABLE THUNGAN(
MATHUNGAN CHAR(10) PRIMARY KEY,
CHUCVU NVARCHAR(30),
HOTEN NVARCHAR(30),
NGAYSINH DATE,
SĐT CHAR(10),
DIACHI NVARCHAR(50)
)
CREATE TABLE THUNGAN ADD CONSTRAINT TN_TK FOREIGN KEY
(MATHUNGAN) REFERENCES TAIKHOAN(MATAIKHOAN)

```

Cài đặt bảng quản kho

```

CREATE TABLE QUANKHO(
MAQUANKHO CHAR(10),
CHUCVU NVARCHAR(30),
HOTEN NVARCHAR(30),
NGAYSINH DATE,
SĐT CHAR(10),
DIACHI NVARCHAR(50)
)
CREATE TABLE QUANKHO ADD CONSTRAINT QK FOREIGN KEY (MAQUANKHO)
REFERENCES TAIKHOAN(MATAIKHOAN)

```

Cài đặt bảng quản trị

```

CREATE TABLE QUANTRI(
MAQUANTRI CHAR(10),

```

```

CHUCVU NVARCHAR(30),
HOTEN NVARCHAR(30),
NGAYSINH DATE,
SĐT CHAR(10),
DIACHI NVARCHAR(50)
)
CREATE TABLE QUANTRI ADD CONSTRAINT QT FOREIGN KEY (MAQUANTRI)
REFERENCES TAIKHOAN(MATAIKHOAN)

```

Cài đặt bảng tài khoản

```

CREATE TABLE TAIKHOAN(
MATAIKHOAN CHAR(10) PRIMARY KEY,
HASHPASS CHAR(100)
)

```

Cài đặt bảng vaccine

```

CREATE TABLE VACCINE(
MAVACCINE CHAR(10) PRIMARY KEY,
TENVACCINE NVARCHAR(20),
NHASX NVARCHAR(30),
NGAYSX DATE,
HANSX DATE,
SOLO CHAR(10),
SOLUONGCOSAN INT,
DONGIA FLOAT,
MAQUANKHO CHAR(10),
MALOAIVC CHAR(10)
)
CREATE TABLE VACCINE ADD CONSTRAINT VC_LVC FOREIGN KEY
(MALOAIVC) REFERENCES LOAIVC(MALOAIVC)
CREATE TABLE VACCINE ADD CONSTRAINT VC_QK FOREIGN KEY
(MAQUANKHO) REFERENCES QUANKHO(MAQUANKHO)

```

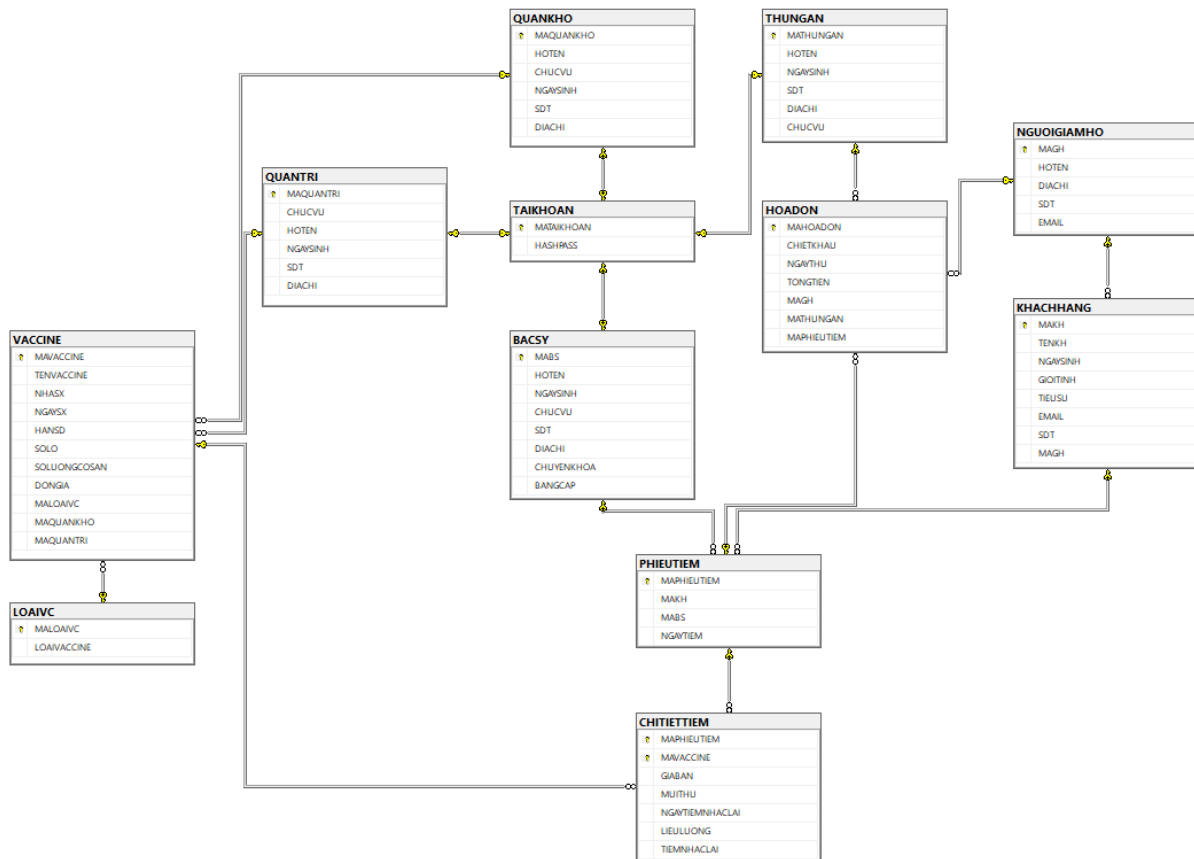
Cài đặt bảng loại vaccine

```

CREATE TABLE LOAIVC(
MALOAIVC CHAR(10),
LOAIVACCINE NVARCHAR(20)
)
CREATE TABLE LOAIVC ADD CONSTRAINT LVC PRIMARY KEY MALOAIVC

```

3.1.2. Mô hình dữ liệu (Diagram)



3.2. Khai thác cơ sở dữ liệu

3.2.1. Trần Bảo Trung

3.2.1.1. Truy vấn cơ bản

1. Lấy tất cả tên vaccine, tên loại vaccine

- Đại số quan hệ

π TENVACCINE, LOAIVACCINE (σ VACCINE.MALOAIVC=LOAIVC.MALOAIVC (VACCINE x LOAIVC))

- Câu lệnh truy vấn

```
SELECT TENVACCINE, LOAIVACCINE FROM VACCINE, LOAIVC
WHERE VACCINE.MALOAIVC=LOAIVC.MALOAIVC
```

2. Cho biết tất cả vaccine có số lượng có sẵn nhỏ hơn 100

- Đại số quan hệ

π MAVACCINE, TENVACCINE (σ SOLUONGCOSAN < 100 (VACCINE))

-Câu lệnh truy vấn

```
SELECT MAVACCINE, TENVACCINE FROM VACCINE  
WHERE SOLUONGCOSAN < 100
```

3. Cho biết mã và tên tất cả vaccine

- Đại số quan hệ

$\pi_{MAVACCINE, TENVACCINE, TENLOAIVC} (\sigma_{VACCINE.MALOAIVC = LOAIVC.MALOAIVCAND} (VACCINE \times LOAIVC))$

- Câu lệnh truy vấn

```
SELECT MAVACCINE, TENVACCINE, TENLOAIVC FROM VACCINE, LOAIVC  
WHERE VACCINE.MALOAIVC = LOAIVC.MALOAIVC
```

4. Lấy tất cả tên vaccine, mã vaccine có số lô là 18

- Đại số quan hệ

$\pi_{MAVACCINE, TENVACCINE} (\sigma_{SOLO = 18} (VACCINE))$

- Câu lệnh truy vấn

```
SELECT MAVACCINE, TENVACCINE FROM VACCINE  
WHERE SOLO = 18
```

5. Cho biết tất cả các vaccine có số lô là 25 và có số lượng còn lại nhỏ hơn 100

- Đại số quan hệ

$\pi_{MAVACCINE, TENVACCINE} (\sigma_{SOLO = 25 \text{ AND } SOLUONGCONLAI < 100} (VACCINE))$

- Câu lệnh truy vấn

```
SELECT MAVACCINE, TENVACCINE FROM VACCINE  
WHERE SOLO = 25  
AND SOLUONGCONLAI < 100
```

6. Cho biết tên các vaccine có loại vaccine là ‘Sởi’

- Đại số quan hệ

$\pi_{TENVACCINE, TENLOAIVC} (\sigma_{VACCINE.MALOAIVC = LOAIVC.MALOAIVCAND \text{ AND } TENLOAIVC = 'SỞI'} (VACCINE \times LOAIVC))$

- Câu lệnh truy vấn

```
SELECT TENVACCINE, TENLOAIVC FROM VACCINE, LOAIVC  
WHERE VACCINE.MALOAIVC = LOAIVC.MALOAIVC  
AND TENLOAIVC = N'SỞI'
```

7. Cho biết mã vaccine, tên vaccine của các vaccine có trong phiếu tiêm có mã là ‘PT001’

- Đại số quan hệ

$\pi_{\text{MAVACCINE, VACCINE.TENVACCINE}} (\sigma_{\text{VACCINE.MAVACCINE = CHITIETTIEM.MAVACCINE AND MAPHIEUTTIEM='PT001'}} (\text{VACCINE x CHITIETTIEM}))$

- Câu lệnh truy vấn

```
SELECT VACCINE.MAVACCINE, VACCINE.TENVACCINE FROM VACCINE, CHITIETTIEM
WHERE VACCINE.MAVACCINE = CHITIETTIEM.MAVACCINE
AND MAPHIEUTTIEM = 'PT001'
```

8. Cho biết tên, giá bán của vaccine có trong phiếu tiêm có mã là 'PT001'

- Đại số quan hệ

$\pi_{\text{VACCINE.TENVACCINE, CHITIETTIEM.GIABAN}} (\sigma_{\text{VACCINE.MAVACCINE = CHITIETTIEM.MAVACC AND VC.TENVC='ASTRA'}} (\text{VACCINE x CHITIETTIEM}))$

- Câu lệnh truy vấn

```
SELECT VACCINE.TENVACCINE, GIABAN FROM VACCINE, CHITIETTIEM
WHERE VACCINE.MAVACCINE = CHITIETTIEM.MAVACCINE
AND MAPHIEUTTIEM='PT001'
```

9. Cho biết tổng số vaccine còn lại của loại vaccine có tên là 'Covid19'

- Đại số quan hệ

$\exists_{\text{SUM(SOLUONGCOSAN)}} (\sigma_{\text{VACCINE.MALOAVC = LOAIVC.MALOAVC AND LOAIVACCINE='COVID19'}} (\text{VACCINE x LOAIVC}))$

- Câu lệnh truy vấn

```
SELECT SUM(SOLUONGCOSAN) AS SOLUONGCOSAN FROM VACCINE, LOAIVC
WHERE VACCINE.MALOAVC=LOAIVC.MALOAVC
AND LOAIVACCINE='Covid19'
```

10. Cho biết tên quản kho nhập vaccine có số lô là 27

- Đại số quan hệ

$\pi_{\text{QUANKHO.HOTEN}} (\sigma_{\text{VACCINE.MAQUANKHO = QUANKHO.MAQUANKHO AND SOLO=27}} (\text{VACCINE x QUANKHO}))$

- Câu lệnh truy vấn

```
SELECT HOTEN FROM VACCINE, QUANKHO
WHERE VACCINE.MAQUANKHO=QUANKHO.MAQUANKHO
AND SOLO=27
```

11. Cho biết tên quản kho và số điện thoại của quản kho có chức vụ là 'TRƯỞNG PHÒNG'

- Đại số quan hệ

$\pi_{HOTEN,SDT}(\sigma_{CHUCVU='TRƯỞNG PHÒNG' } (QUANKHO))$

- Câu lệnh truy vấn

```
SELECT HOTEN, SDT FROM QUANKHO
WHERE CHUCVU=N'TRƯỞNG PHÒNG'
```

12. Cho biết tất cả số lô vaccine quản kho 'QK001' nhập

- Đại số quan hệ

$\pi_{VACCINE.SOLO}(\sigma_{VACCINE.MAQUANKHO = QUANKHO.MAQUANKHO \text{ AND } QUANKHO.MAQUANKHO='QK001'} (VACCINE \times QUANKHO))$

- Câu lệnh truy vấn

```
SELECT SOLO FROM VACCINE, QUANKHO
WHERE QUANKHO.MAQUANKHO=VACCINE.MAQUANKHO
AND QUANKHO.MAQUANKHO='QK001'
```

13. Cho biết số vaccine có loại vaccine là 'Thủy đậu'

- Đại số quan hệ

$\mathfrak{Z}_{COUNT(MAVACCINE)}(\sigma_{VACCINE.MALOAIVC = LOAIVC.MALOAIVC \text{ AND } LOAIVACCINE='THỦY ĐẬU'} (VACCINE \times LOAIVC))$

- Câu lệnh truy vấn

```
SELECT COUNT(MAVACCINE) AS SOVACCINE FROM VACCINE, LOAIVC
WHERE VACCINE.MALOAIVC=LOAIVC.MALOAIVC
AND LOAIVACCINE=N'Thủy đậu'
```

14. Cho biết những người quản kho ở Hồ Chí Minh

- Đại số quan hệ:

$\sigma_{(DIACHI = '%HA NOI')} (QUANKHO)$

- Câu lệnh truy vấn

```
SELECT * FROM QUANKHO WHERE DIACHI LIKE N'%HA NOI%'
```

15. Kiểm tra xem có bao nhiêu quản kho đã có tài khoản

- Đại số quan hệ

$\mathfrak{Z}_{COUNT(TAIKHOAN.MATAIKHOAN)}(\sigma_{TAIKHOAN.MATAIKHOAN= QUANKHO.MAQUANKHO \text{ AND } (QUANKHO \times TAIKHOAN)})$

- Câu lệnh truy vấn

```
SELECT COUNT(MAQUANKHO) FROM QUANKHO, TAIKHOAN
WHERE QUANKHO.MAQUANKHO=TAIKHOAN.MATAIKHOAN
```

16. Tính tổng số tiền nhập vaccine của số lô 100

- Đại số quan hệ

$\Sigma_{\text{SUM(DONGIA)}}(\sigma_{\text{SOLO}=100} \text{ VACCINE})$

- Câu lệnh truy vấn

```
SELECT SUM(DONGIA) FROM VACCINE
WHERE SOLO=100
```

17. Cho biết các phiếu tiêm có ngày tiêm nhắc lại hẹn ngày 20-10-2022

- Đại số quan hệ:

$\sigma_{(\text{NGAYTIEMNHACLAI} = '2022-10-20')} (\text{CHITIETTIEM})$

- Câu lệnh truy vấn:

```
SELECT * FROM CHITIETTIEM WHERE NGAYTIEMNHACLAI = '2022-10-20'
```

18. Cho biết phiếu tiêm có ngày tiêm nhắc lại là '22-12-2022'

- Đại số quan hệ

$\sigma_{(\text{NGAYTIEMNHACLAI} = '2022-12-22')} (\text{CHITIETTIEM})$

- Câu lệnh truy vấn

```
SELECT * FROM CHITIETTIEM
WHERE NGAYTIEMNHACLAI = '2022-12-22'
```

19. Cho biết các vaccine có ngày sản xuất từ 12/03/2021 đến 12/03/2022

- Đại số quan hệ

$\sigma_{('2021-03-12' < \text{NGAYSX} = '2022-02-12')} (\text{VACCINE})$

- Câu lệnh truy vấn

```
SELECT TENVACCINE FROM VACCINE WHERE NGAYSX BETWEEN '2021-03-12' AND '2022-03-12'
```

20. Cho biết tên và liều lượng của các vaccine có mã phiếu tiêm là 'PT002'

- Đại số quan hệ

$\pi_{\text{VACCINE.TENVACCINE, CHITIETTIEM.LIEULUONG}}(\sigma_{\text{VACCINE.MAVACCINE} = \text{CHITIETTIEM.MAVC AND MAPHIEUTIEM} = 'PT002'} (\text{VACCINE x CHITIETTIEM}))$

- Câu lệnh truy vấn

```
SELECT VACCINE.TENVACCINE, LIEULUONG FROM VACCINE, CHITIETTIEM
WHERE VACCINE.MAVACCINE = CHITIETTIEM.MAVACCINE
AND MAPHIEUTIEM = 'PT002'
```

3.2.1.2 Truy vấn nâng cao

1. Đưa ra tên loại vaccine và đơn giá cao nhất trong từng loại Vaccin

```
SELECT LOAIVACCINE, MAX(DONGIA) AS 'GIACAONHAT' FROM VACCINE, LOAIVC
WHERE VACCINE.MALOAIVC=LOAIVC.MALOAIVC
GROUP BY LOAIVACCINE
```

2. Đưa ra loại vaccine có số lượng có sẵn nhỏ hơn 100

```
SELECT LOAIVACCINE, SUM(SOLUONGCOSAN) AS 'SOLUONGCOSAN' FROM VACCINE,
LOAIVC
WHERE VACCINE.MALOAIVC=LOAIVC.MALOAIVC
GROUP BY LOAIVACCINE
HAVING SUM(SOLUONGCOSAN)<100
```

3. Đếm số lượng vaccine theo từng mã loại vaccine

```
SELECT LOAIVC.MALOAIVC, SUM (VACCINE.SOLUONGCOSAN) AS SL FROM LOAIVC
LEFT JOIN VACCINE ON VACCINE.MALOAIVC = LOAIVC.MALOAIVC
GROUP BY LOAIVC.MALOAIVC
```

4. Cho biết 5 vaccine có số lượng ít nhất

```
SELECT TOP(5) MAVACCINE, TENVACCINE, SOLUONGCOSAN FROM VACCINE
ORDER BY SOLUONGCOSAN ASC
```

5. Xuất các vùng xuất xứ vaccine kèm số lượng

```
SELECT NHASX, SUM(SOLUONGCOSAN) AS SOLUONG FROM VACCINE
GROUP BY NHASX
```

6. Cho biết các phiếu tiêm có tổng thanh toán lớn hơn 200000

```
SELECT MAPHIEUTIEM , SUM(GIABAN) FROM CHITIIETTIEM
GROUP BY MAPHIEUTIEM
HAVING SUM(GIABAN)>200000
```

7. Đưa ra đơn giá nhỏ nhất, lớn nhất của vaccine

```
SELECT MAX(DONGIA) AS DONGILONNHAT, MIN(DONGIA) AS DONGIANHONHAT
FROM VACCINE
```

8. Đưa ra các vaccine cách hạn sử dụng 50 ngày

```
SELECT MAVACCINE, TENVACCINE, DATEDIFF(DAY, GETDATE(), HANSĐ) AS
'SONGAYCONLAI' FROM VACCINE
WHERE DATEDIFF(DAY, GETDATE(), HANSĐ)<50
```

9. Đưa ra mã quản kho, tên quản kho nhập số lô 18

```
SELECT MAQUANKHO, HOTEN FROM VACCINE, QUANKHO
WHERE QUANKHO.MAQUANKHO=VACCINE.MAQUANKHO
AND VACCINE.SOLO='18'
```

- 10.Đưa ra phiếu tiêm có ngày tiêm nhắc lại cách ngày hiện tại 20 ngày

```
SELECT MAPHIEUTIEM, DATEDIFF(DAY, GETDATE(), NGAYTIEMNHACLAI) AS
'NGAYTIEMNHACLAI' FROM CHITIIETTIEM
WHERE DATEDIFF(DAY, GETDATE(), NGAYTIEMNHACLAI)<20
```

3.2.1.3. Truy vấn TSQL

1. Lấy tất cả lịch sử tiêm của khách hàng theo mã khách hàng

```
CREATE PROC sp_GetAllLichSuTiem (@MAKH CHAR(10))
AS
BEGIN
```

```

SELECT pt.NGAYTIEM, vc.TENVACCINE, lvc.LOAIVACCINE, vc.NHASX,
ctt.MUITHU, ctt.LIEULUONG, ctt.NGAYTIEMNHACLAIR, vc.DONGIA
FROM KHACHHANG kh INNER JOIN PHIEUTIAM pt INNER JOIN CHITIETIAM ctt
INNER JOIN VACCINE vc INNER JOIN LOAIVC lvc
ON lvc.MALOAIVC = vc.MALOAIVC
ON vc.MAVACCINE = ctt.MAVACCINE
ON ctt.MAPHIEUTIAM = pt.MAPHIEUTIAM
ON pt.MAKH = kh.MAKH
WHERE kh.MAKH = @MAKH
END

```

2. Lấy tất cả vaccine

```

CREATE PROC sp_GetAllVaccine
AS
BEGIN
SELECT vc.MAVACCINE, vc.TENVACCINE, vc.NHASX, vc.NGAYSX,
vc.HANSO, vc.SOLO, vc.SOLUONGCOSAN, vc.DONGIA, lvc.LOAIVACCINE
FROM VACCINE vc INNER JOIN LOAIVC lvc ON lvc.MALOAIVC = vc.MALOAIVC
END

```

3. Lấy thông tin bệnh nhân dựa trên mã phiếu tiêm

```

CREATE PROC spRP_GetInfoBenhNhan(@maPhieuTiem CHAR(10))
AS
BEGIN
SELECT *
FROM PHIEUTIAM INNER JOIN KHACHHANG ON KHACHHANG.MAKH =
PHIEUTIAM.MAKH
WHERE MAPHIEUTIAM = @maPhieuTiem
END

```

4. Lấy thông tin quản kho theo mã quản kho

```

CREATE PROC sp_GetQuanKhoInfo( @MAQUANKHO CHAR(10))
AS
BEGIN
SELECT MAQUANKHO, HOTEN, NGAYSINH, SDT, DIACHI, CHUCVU FROM QUANKHO
WHERE MAQUANKHO = @MAQUANKHO
END

```

5. Tìm kiếm vaccine theo mã vaccine

```

CREATE PROC sp_SearchByMaVC (@Value char(10))
AS
BEGIN
SELECT vc.MAVACCINE, vc.TENVACCINE, vc.NHASX, vc.NGAYSX, vc.HANSO,
vc.SOLO, vc.SOLUONGCOSAN, vc.DONGIA, lvc.LOAIVACCINE
FROM VACCINE vc INNER JOIN LOAIVC lvc ON lvc.MALOAIVC = vc.MALOAIVC
WHERE vc.MAVACCINE LIKE N'%' + @Value + '%'
END

```

6. Tìm kiếm vaccine theo tên vaccine

```

CREATE PROC sp_SearchByTenVC (@Value NVARCHAR(100))

```

```

AS
BEGIN
    SELECT vc.MAVACCINE, vc.TENVACCINE, vc.NHASX, vc.NGAYSX, vc.HANSĐ,
vc.SOLO,      vc.SOLUONGCOSAN, vc.DONGIA, lvc.LOAIVACCINE
    FROM VACCINE vc INNER JOIN LOAIVC lvc ON lvc.MALOAIVC = vc.MALOAIVC
    WHERE vc.TENVACCINE LIKE N'%' + @Value + '%'
END

```

7. Tìm kiếm vaccine theo loại vaccine

```

CREATE PROC sp_SearchByLoaiVC (@Value NVARCHAR(100))
AS
BEGIN
    SELECT vc.MAVACCINE, vc.TENVACCINE, vc.NHASX, vc.NGAYSX, vc.HANSĐ,
vc.SOLO,      vc.SOLUONGCOSAN, vc.DONGIA, lvc.LOAIVACCINE
    FROM VACCINE vc INNER JOIN LOAIVC lvc ON lvc.MALOAIVC = vc.MALOAIVC
    WHERE lvc.LOAIVACCINE LIKE N'%' + @Value + '%'
END

```

8. Tạo thủ tục xóa loại vaccine và xóa luôn tất cả các vaccine thuộc loại vaccine đó

```

CREATE TRIGGER trg_xoaLVC ON LOAIVC INSTEAD OF DELETE AS
BEGIN
    DECLARE @MALOAIVC CHAR(10)
    SET @MALOAIVC= (SELECT MALOAIVC FROM Deleted)
    DELETE FROM LOAIVC WHERE MALOAIVC=@MALOAIVC
    DELETE FROM VACCINE WHERE MALOAIVC=@MALOAIVC
END

```

9. Kiểm tra vaccine có còn trong kho không theo mã vaccine

```

CREATE PROC sp_IsVCInStock (@MAVACCINE CHAR(10))
AS
BEGIN
    DECLARE @returnVal int
    IF EXISTS (SELECT MAVACCINE FROM VACCINE WHERE MAVACCINE= @MAVACCINE)
    BEGIN
        SET @returnVal = 1
    END
    ELSE SET @returnVal = 0
    RETURN @returnVal
END

```

10. Tìm kiếm vaccine theo một từ khóa bất kì

```

CREATE PROC sp_SearchAllVaccine (@Value NVARCHAR(100))
AS
BEGIN
    SELECT vc.MAVACCINE, vc.TENVACCINE, vc.NHASX, vc.NGAYSX, vc.HANSĐ,
vc.SOLO, vc.SOLUONGCOSAN, vc.DONGIA, lvc.LOAIVACCINE
    FROM VACCINE vc INNER JOIN LOAIVC lvc ON lvc.MALOAIVC = vc.MALOAIVC
    WHERE vc.MAVACCINE LIKE N'%' + @Value + '%'
    OR vc.TENVACCINE LIKE N'%' + @Value + '%'

```

```

OR vc.NHASX LIKE N'%' + @Value + '%'
OR vc.NGAYSX LIKE N'%' + @Value + '%'
OR vc.HANSX LIKE N'%' + @Value + '%'
OR vc.SOLO LIKE N'%' + @Value + '%'
OR vc.SOLUONGCOSAN LIKE N'%' + @Value + '%'
OR lvc.LOAIIVACCINE LIKE N'%' + @Value + '%'
OR vc.DONGIA LIKE N'%' + @Value + '%'

```

END

3.2.2. Phạm Đình Khương Duy

3.2.2.1. Truy vấn cơ bản

- Đưa ra mã thu ngân, tên thu ngân viết hóa đơn
- Đại số quan hệ

$$\pi_{MATHUNGAN, HOTEN}(\sigma_{THUNGAN.MATHUNGAN=HOADON.MATHUNGAN}(THUNGAN \times HOADON))$$

- Câu lệnh truy vấn

```

SELECT THUNGAN. MATHUNGAN, THUNGAN.HOTEN FROM THUNGAN, HOADON
WHERE THUNGAN.MATHUNGAN=HOADON.MATHUNGAN

```

- Đưa ra mã khách hàng, tên khách hàng có họ ‘Phạm’

- Đại số quan hệ

$$\pi_{MAKH, TENKH}(\sigma_{TENKH='Phạm\%', (KHACHHANG)})$$

- Câu lệnh truy vấn

```

SELECT MAKH, TENKH FROM KHACHHANG WHERE TENKH LIKE N'Phạm%'

```

- Cho biết các thu ngân có chức vụ là ‘Kế toán’

- Đại số quan hệ

$$\sigma_{CHUCVU='KẾ TOÁN'}(THUNGAN)$$

- Câu lệnh truy vấn

```

SELECT * FROM THUNGAN WHERE CHUCVU= 'Kế toán'

```

- Cho biết các thu ngân có họ Nguyễn và chức vụ = ‘Thủ quỹ’

- Đại số quan hệ

$$\sigma_{(ChucVu = 'Thủ quỹ' \text{ AND } hoTen = 'Nguyễn\%')}(THUNGAN)$$

- Câu lệnh truy vấn

```

SELECT * FROM THUNGAN WHERE HOTEN LIKE N'Nguyễn%' AND CHUCVU = 'Thủ quỹ'

```

5. Cho biết các thu ngân có địa chỉ ở Nghệ An

- Đại số quan hệ

$\sigma_{\text{DiaChi} = \% \text{Nghệ An}} (\text{THUNGAN})$

- Câu lệnh truy vấn

`SELECT * FROM THUNGAN WHERE DIACHI LIKE N'%Nghệ An'`

6. Cho biết những người giám hộ ở Hồ Chí Minh

- Đại số quan hệ:

$\sigma_{(\text{DIACHI} = \% \text{TP HCM})} (\text{NGUOIGIAMHO})$

- Câu lệnh truy vấn

`SELECT * FROM NGUOIGIAMHO WHERE DIACHI LIKE N'%TP HCM%'`

7. Cho biết tất cả số lô có nhập loại vaccine ‘Covid19’

- Đại số quan hệ

$\pi_{\text{SOLO}}(\sigma_{\text{VACCINE.MALOAIVC} = \text{LOAIVC.MALOAIVC AND LOAIVACCINE} = \text{'COVID19'}} (\text{VACCINE x LOAIVC}))$

- Câu lệnh truy vấn

`SELECT SOLO AS SOVACCINE FROM VACCINE, LOAIVC
WHERE VACCINE.MALOAIVC=LOAIVC.MALOAIVC
AND LOAIVACCINE=N'Covid19'`

8. Cho biết các hóa đơn do thu ngân mã ‘TN001’ nhập

- Đại số quan hệ:

$\sigma_{(\text{MATHUNGAN} = \text{'TN001'})} (\text{HOADON})$

- Câu lệnh truy vấn

`SELECT * FROM HOADON WHERE MATHUNGAN = 'TN001'`

9. Cho biết tên những khách hàng có mã người giám hộ là ‘GH001’

- Đại số quan hệ:

$\pi_{\text{TENKH}}(\sigma_{(\text{MAGH} = \text{'GH001'})} (\text{KHACHHANG}))$

- Câu lệnh truy vấn

`SELECT TENKH FROM KHACHHANG WHERE MAGH= 'GH001'`

10. Đưa ra mã thu ngân, tên thu ngân, năm sinh của thu ngân sinh năm 2001

- Đại số quan hệ:

$\pi_{\text{MATHUNGAN}, \text{HOTEN}}(\sigma_{\text{YEAR(NGAYSINH)} = 2001} (\text{THUNGAN}))$

- Câu lệnh truy vấn:

`SELECT MATHUNGAN, HOTEN FROM THUNGAN WHERE YEAR(NGAYSINH)=2001`

11. Tìm mã khách hàng, tên khách hàng, tên người giám hộ các khách hàng tên “Duy”

- Đại số quan hệ:

$\pi_{MAKH, TENKH, TENGH}(\sigma_{KHACHHANG.MAGH=NGUOIGH.MAGH \text{ AND } HOTEN='Duy'}(KHACHHANG \times GIAMHO))$

- Câu lệnh truy vấn:

`SELECT MAKH, TENKH, TENGH FROM KHACHHANG, NGUOIGIAMHO
WHERE KHACHHANG.MAGH = NGUOIGIAMHO.MAGH
AND TENKH LIKE 'Duy'`

12. Tìm mã thu ngân, tên thu ngân có chữ cái 'x' trong tên

- Đại số quan hệ:

$\pi_{MATHUNGAN, HOTEN}(\sigma_{hoTen='%x\%'}(THUNGAN))$

- Câu lệnh truy vấn:

`SELECT MATHUNGAN, HOTEN FROM THUNGAN WHERE HOTEN LIKE 'x%'`

13. Đưa ra các thu ngân có ngày sinh trong khoảng 1-1-2001 đến 1-10-2001

- Đại số quan hệ:

$(\sigma_{NGAYSINH \text{ BETWEEN } '2001-1-1' \text{ AND } '2001-10-1'}(THUNGAN))$

- Câu lệnh truy vấn:

`SELECT * FROM THUNGAN
WHERE NGAYSINH BETWEEN '2001-1-1' AND '2001-10-1'`

14. Đưa ra mã Vacxin, Tên Vacxin, đơn giá Vacxin có đơn giá > 200

- Đại số quan hệ:

$\pi_{maVacxin, tenVacxin, donGia}(\sigma_{donGia > 200}(VACCINE))$

- Câu lệnh truy vấn:

`SELECT MAVACXIN, TENVACXIN, DONGIA FROM VACCINE WHERE DONGIA > 200`

15. Đưa ra những phiếu tiêm tiêm trong tháng 5

- Đại số quan hệ:

$(\sigma_{MONTH(NGAYTIEM)=5}(PHIEUTIEM))$

- Câu lệnh truy vấn:

`SELECT * FROM PHIEUTIEM WHERE MONTH(NGAYTIEM)=5`

16. Cho biết số vaccine có loại vaccine là ‘Sởi’

- Đại số quan hệ

$\exists \text{COUNT(MAVACCINE)} (\sigma_{\text{VACCINE.MALOAIVC} = \text{LOAIVC.MALOAIVC AND LOAIVACCINE} = \text{'SỞI'}} (\text{VACCINE} \times \text{LOAIVC}))$

- Câu lệnh truy vấn

```
SELECT COUNT(MAVACCINE) AS SOVACCINE FROM VACCINE, LOAIVC
WHERE VACCINE.MALOAIVC=LOAIVC.MALOAIVC
AND LOAIVACCINE=N'Sởi'
```

17. Đưa ra mã hóa đơn của thu ngân có mã thu ngân là 'TN001':

- Đại số quan hệ:

$\pi_{\text{NgàyThu}} (\sigma_{\text{MaNV} = \text{'NV01'}} (\text{HOADON}))$

- Câu lệnh truy vấn:

```
SELECT MAHOADON FROM HOADON WHERE MATHUNGAN='TN001'
```

18. Đưa ra các thu ngân là nam :

- Đại số quan hệ:

$\sigma_{(\text{GIOITINH} = \text{'NAM'})} (\text{THUNGAN})$

- Câu lệnh truy vấn:

```
SELECT * FROM THUNGAN WHERE GIOITINH = 'NAM'
```

19. Đưa ra nhân viên sinh vào Ngày 7 trong bảng Nhân Viên

- Đại số quan hệ:

$(\sigma_{\text{Day}(\text{NGAYSINH})=7} (\text{THUNGAN}))$

- Câu lệnh truy vấn:

```
SELECT * FROM THUNGAN WHERE day(NGAYSINH) =7
```

20. Đưa ra mã hóa đơn của thu ngân có tên thu ngân là 'Khương Duy':

- Đại số quan hệ:

$\pi_{\text{MAHOADON}} (\sigma_{\text{HOADON.MATHUNGAN} = \text{THUNGAN.MATHUNGAN AND HOTEN} = \text{'Khương Duy'}} (\text{HOADON} \times \text{THUNGAN}))$

- Câu lệnh truy vấn:

```
SELECT MAHOADON FROM HOADON, THUNGAN
WHERE HOADON.THUNGAN= THUNGAN.MATHUNGAN
HOTEN=N'Khương Duy'
```

3.2.2.2. Truy vấn nâng cao

1. Đưa ra hóa đơn có tổng tiền cao nhất

```
SELECT TOP 1 * FROM HOA DON ORDER BY TONGTIEN desc
```

2. Đếm tổng số thu ngân đã có tài khoản

```
SELECT COUNT(MATHUNGAN) AS 'SOTHUNGAN'  
FROM THUNGAN, TAIKHOAN  
WHERE THUNGAN.MATHUNGAN=TAIKHOAN.MATAIKHOAN
```

3. Đưa ra mã giám hộ, tên giám hộ có hóa đơn với tổng tiền lớn hơn 20000

```
SELECT * FROM NGUOIGIAMHO INNER JOIN HOADON  
ON NGUOIGIAMHO.MAGH = HOADON.MAGH WHERE TONGTIEN >20000
```

4. Cho biết tổng số phiếu tiêu của khách hàng có mã là 'KH001'

```
SELECT COUNT(MAPHIEUTIEU) AS 'SOPHIEUTIEU'  
FROM PHIEUTIEU  
WHERE MAKH= 'KH001'
```

5. Cho biết tổng tiền của phiếu tiêu có mã bác sĩ là 'BS001'

```
SELECT MABS, SUM(TONGTIEN) AS TONGTIEN  
FROM PHIEUTIEU, HOADON  
WHERE PHIEUTIEU.MAPHIEUTIEU=HOADON.MAPHIEUTIEU  
AND MABS= 'BS001'  
GROUP BY MABS
```

6. Cho biết số thu ngân là nữ

```
SELECT COUNT(MATHUNGAN) AS SOTHUNGAN  
FROM THUNGAN  
WHERE GIOITINH=N'Nữ'
```

7. Cho biết các mã phiếu tiêu của bác sĩ có họ 'Phạm'

```
SELECT MAPHIEUTIEU FROM PHIEUTIEU, BACSI  
WHERE PHIEUTIEU.MABS = BACSI.MABS  
AND BACSI.HOTEN LIKE N'Phạm%'
```

8. Cho biết các phiếu tiêu có ngày tiêu từ 1-1-2022 đến 15-5-2022

```
SELECT * FROM PHIEUTIEU  
WHERE NGAYTIEU BETWEEN '2022-01-01' AND '2022-05-15'
```

9. Đưa ra Mã Bác sĩ, Tên Bác sĩ đã tiêu cho phiếu có mã phiếu tiêu là "PT001"

```
SELECT BACSI.MABS, BACSI.HOTEN FROM PHIEUTIEU, BACSI  
WHERE PHIEUTIEU.MABS = BACSI.MABS  
AND maPhieu='PT001'
```

10. Đưa ra mã khách hàng, tên khách hàng có số lượng phiếu tiêu >2

```

SELECT KHACHHANG.MAKH, TENKH ,COUNT(MAPHIEUTIEM)
FROM KHACHHANG, PHIEUTIEM
WHERE KHACHHANG.MAKH=PHIEUTIEM.MAKH
GROUP BY KHACHHANG.MAKH,TENKH
HAVING COUNT(MAPHIEUTIEM)>2

```

3.2.2.3 Truy vấn TSQL

1. Lấy thông tin thu ngân

```

CREATE PROC sp_GetThuNganInfo( @MATHUNGAN CHAR(10))
AS
BEGIN
    SELECT MATHUNGAN, HOTEN, NGAYSINH, SDT, DIACHI, CHUCVU FROM THUNGAN WHERE
MATHUNGAN=@MATHUNGAN
END

```

2. Lấy thông tin phiếu tiêm và thông tin bác sĩ

```

CREATE PROC sp_GetPhieuTiemsInfo
AS
BEGIN
    SELECT pt.MAPHIEUTIEM, kh.MAKH, kh.TENKH, kh.NGAYSINH, kh.GIOITINH,
kh.TIEUSU, pt.NGAYTIEM, bs.HOTEN AS TENBS
    FROM BACSY bs INNER JOIN PHIEUTIEM pt INNER JOIN KHACHHANG kh
    ON kh.MAKH = pt.MAKH
    ON pt.MABS = bs.MABS
    ORDER BY pt.MAPHIEUTIEM DESC
END

```

3. Lấy thông tin hóa đơn

```

CREATE PROC sp_GetAllHoaDonInfo
AS
BEGIN
    SELECT hd.MAHOADON, hd.TONGTIEN, hd.CHIETKHAU, hd.NGAYTHU, ngh.HOTEN AS
NGUOIGH, ngh.DIACHI, ngh.SDT, tn.HOTEN AS THUNGAN
    FROM THUNGAN tn INNER JOIN HOADON hd INNER JOIN NGUOIGIAMHO ngh
    ON ngh.MAGH = hd.MAGH
    ON hd.MATHUNGAN = tn.MATHUNGAN
END

```

4. Kiểm tra tài khoản khách hàng đã thanh toán chưa

```

CREATE PROC sp_CheckPaymentStatus (@MAPHIEUTIEM CHAR(10))
AS
BEGIN
    DECLARE @returnVal int
    IF EXISTS (SELECT * FROM PHIEUTIEM pt INNER JOIN HOADON hd ON
hd.MAPHIEUTIEM = pt.MAPHIEUTIEM WHERE pt.MAPHIEUTIEM = @MAPHIEUTIEM)
    BEGIN
        SET @returnVal = 1
    END

```

```

END
ELSE SET @returnVal = 0
RETURN @returnVal
END

```

5. Tìm kiếm khách hàng theo tên

```

CREATE PROC sp_SearchTenKhachHang (@Value NVARCHAR(100))
BEGIN
    SELECT MAKH, TENKH FROM KHACHHANG
    WHERE TENKH LIKE N'%' + @Value + '%'
END

```

6. Lấy tên thu ngân bằng mã thu ngân

```

CREATE PROC spRP_GetTenThuNgan (@maTN CHAR(10))
AS
BEGIN
    SELECT HOTEN AS 'HoTenTN'
    FROM THUNGAN
    WHERE MATHUNGAN = @maTN
END

```

7. Tìm kiếm hóa đơn theo mã hóa đơn

```

CREATE PROC sp_SearchHoaDon (@Value CHAR(10))
AS
BEGIN
    SELECT * FROM HOADON
    WHERE MAHOADON = @Value
END

```

8. Lấy thông tin phiếu tiêm từ mã hóa đơn

```

CREATE PROC sp_GetMaPhieuTiemFromHD (@MAHOADON CHAR(10))
AS
BEGIN
    SELECT MAPHIEUTIEM FROM HOADON WHERE MAHOADON = @MAHOADON
END

```

9. Thêm mã giám hộ vào khách hàng

```

CREATE PROC sp_AddMaGHtoKH (@MAGH CHAR(10))
AS
BEGIN
    UPDATE KHACHHANG SET MAGH = @MAGH WHERE MAKH IN (
        SELECT kh.MAKH
        FROM NGUOIGIAMHO gh INNER JOIN HOADON hd INNER JOIN PHIEUTIEM pt INNER
JOIN KHACHHANG kh
        ON kh.MAKH = pt.MAKH ON pt.MAPHIEUTIEM = hd.MAPHIEUTIEM ON hd.MAGH =
gh.MAGH
        WHERE gh.MAGH = @MAGH )
END

```

10. Giám số lượng vaccine có sẵn sau khi tạo chitiettiem

```

CREATE TRIGGER trg_TaoChiTietTiem ON CHITIETTIEM AFTER INSERT AS
BEGIN
    DECLARE @DaSuDung FLOAT
    SET @DaSuDung = (
        SELECT Inserted.LIEULUONG
        FROM Inserted
    )

    SET @DaSuDung = ROUND(@DaSuDung, 0) --lam tron len

    UPDATE dbo.VACCINE
    SET SOLUONGCOSAN = SOLUONGCOSAN - @DaSuDung
    FROM Inserted INNER JOIN dbo.VACCINE vc ON vc.MAVACCINE =
Inserted.MAVACCINE
    WHERE vc.MAVACCINE = Inserted.MAVACCINE
END

```

3.2.3. Hồ Nguyễn Nguyên

3.2.3.1. Truy vấn cơ bản

1. Cho biết các bác sĩ có chuyên khoa là ‘Răng hàm mặt’

- Đại số quan hệ

$$\sigma_{\text{CHUYENKHOA}=\text{'RĂNG HÀM MẶT'}}(\text{BACSI})$$

- Câu lệnh truy vấn

```
SELECT * FROM BACSY WHERE CHUYENKHOA= N'Răng hàm mặt'
```

2. Đưa ra mã bác sĩ, Tên bác sĩ của bác sĩ có bằng cấp là Đại học

- Đại số quan hệ:

$$\pi_{\text{MABS}, \text{HOTEN}}(\sigma_{\text{BANGCAP}=\text{'ĐẠI HỌC'}}(\text{BACSI}))$$

- Câu lệnh truy vấn:

```
SELECT MABS, TENBS FROM BACSY WHERE BANGCAP= N'Đại học'
```

3. Đưa ra mã bác sĩ, Tên bác sĩ trong phiếu tiêm có mã là ‘PT001’

- Đại số quan hệ

$$\pi_{\text{MABS}, \text{HOTEN}}(\sigma_{\text{BACSY.MABS} = \text{PHIEUTIAM.MABS AND MAPHIEUTIAM}=\text{'PT001'}}(\text{BACSY x PHIEUTIAM}))$$

- Câu lệnh truy vấn

```

SELECT BACSY.MABS, HOTEN FROM BACSY, PHIEUTIAM
WHERE BACSY.MABS=PHIEUTIAM.MABS
AND PHIEUTIAM.MAPHIEUTIAM= 'PT001'

```

4. Cho biết tên giám hộ và số điện thoại của giám hộ có mã giám hộ là 'GH001'

- Đại số quan hệ

$\pi_{\text{HOTEN,SDT}}(\sigma_{\text{MAGH}='GH001'} \text{NGUOIGIAMHO})$

- Câu lệnh truy vấn

```
SELECT HOTEN, SDT FROM NGUOIGIAMHO
WHERE MAGH=N'GH001'
```

5. Cho biết tất cả mã khách hàng, tên khách hàng có người giám hộ tên là 'Hồ Nguyễn Nguyên'

- Đại số quan hệ

$\pi_{\text{MAKH, TENKH}}(\sigma_{\text{KHACHHANG.MAGH} = \text{NGUOIGIAMHO.MAGH AND NGUOIGIAMHO.HOTEN}=' \text{Hồ Nguyễn Nguyên}' } (\text{NGUOIGIAMHO} \times \text{KHACHHANG}))$

- Câu lệnh truy vấn

```
SELECT MAKH, TENKH FROM NGUOIGIAMHO, KHACHHANG
WHERE KHACHHANG.MAGH=GIAMHO.MAGH
AND NGUOIGIAMHO.HOTEN= N'Hồ Nguyễn Nguyên'
```

6. Cho biết tên, đơn giá của vaccine có trong phiếu tiêm có mã là 'PT005'

- Đại số quan hệ

$\pi_{\text{VACCINE.TENVACCINE, VACCINE.DONGIA}}(\sigma_{\text{VACCINE.MAVACCINE} = \text{CHITIETTIEM.MAVC AND MAPHIEUTIEM}='PT005'} (\text{VACCINE} \times \text{CHITIETTIEM}))$

- Câu lệnh truy vấn

```
SELECT VACCINE.TENVACCINE, DONGIA FROM VACCINE, CHITIETTIEM
WHERE VACCINE.MAVACCINE = CHITIETTIEM.MAVACCINE
AND MAPHIEUTIEM='PT005'
```

7. Đếm số bác sĩ là nam và tham gia khám:

- Đại số quan hệ:

$\exists_{\text{COUNT(MABS)}}(\sigma_{\text{BACSY.MABS=PHIEUTIEM.MABS AND GIOITINH}='NAM'} (\text{BACSY} \times \text{PHIEUTIEM}))$

- Câu lệnh truy vấn

```
SELECT COUNT(MABS) FROM BACSY, PHIEU TIEM
WHERE BACSY.MABS= PHIEUTIEM.MABS
AND GIOITINH='Nam'
```

8. Đưa ra mã bác sĩ, tên bác sĩ có trong bảng TAIKHOAN

- Đại số quan hệ

$\pi_{MABS, HOTEN}(\sigma_{BACSY.MABS= TAIKHOAN.MATAIKHOAN}(BACSY \times TAIKHOAN))$

- Câu lệnh truy vấn

```
SELECT THUNGAN, MATHUNGAN, THUNGAN.HOTEN FROM THUNGAN, HOADON
WHERE THUNGAN.MATHUNGAN=HOADON.MATHUNGAN
```

9. Đưa ra các bác sĩ có ngày sinh trong khoảng 1-1-2001 đến 1-10-2001

- Đại số quan hệ:

$(\sigma_{NGAYSINH \text{ BETWEEN '2001-1-1' AND '2001-10-1' }}(BACSY))$

- Câu lệnh truy vấn:

```
SELECT * FROM BACSY
WHERE NGAYSINH BETWEEN '2001-1-1' AND '2001-10-1'
```

10. Cho biết tất cả mã phiếu tiêm do bác sĩ ‘Hồ Nguyễn Nguyên’ tiêm

- Đại số quan hệ

$\pi_{VACCINE.SOLO}(\sigma_{VACCINE.MAQUANKHO = QUANKHO.MAQUANKHO \text{ AND } QUANKHO.MAQUANKHO='QK001'}(VACCINE \times QUANKHO))$

- Câu lệnh truy vấn

```
SELECT SOLO FROM VACCINE, QUANKHO
WHERE QUANKHO.MAQUANKHO=VACCINE.MAQUANKHO
AND QUANKHO.MAQUANKHO='QK001'
```

11. Cho biết tất cả phiếu tiêm có sử dụng vaccine ‘Astra’

- Đại số quan hệ

$\pi_{MAPT}(\sigma_{VACCINE.MAVACCINE = PHIEUTIEM.MAVACCINE \text{ AND } TENVACCINE='ASTRA'}(VACCINE \times PHIEUTIEM))$

- Câu lệnh truy vấn

```
SELECT MAPT FROM VACCINE, PHIEUTIEM
WHERE VACCINE.MAVACCINE=PHIEUTIEM.MAVACCINE
AND TENVACCINE=N'Astra'
```

12. Đưa ra mã phiếu tiêm của bác sĩ có mã tên là ‘Hồ Nguyễn Nguyên’:

- Đại số quan hệ:

$\pi_{MAPHIEUTIEM}(\sigma_{PHIEUTIEM.MABS=BACSY.MABS \text{ AND } BACSY.HOTEN='NGUYEN'}(PHIEUTIEM \times BACSY))$

- Câu lệnh truy vấn:

```
SELECT MAHOADON FROM PHIEUTIEM, BACSY
```

WHERE PHIEUTIEM.MABS=BACSY.MABS

AND BACSY.HOTEN= N'Hồ Nguyễn Nguyên'

13. Đưa ra thông tin vacxin hết hạn trong năm nay

- Đại số quan hệ:

$\sigma_{\text{YEAR}(\text{HSD}) = \text{YEAR}(\text{GETDATE}())}(\text{Vacxin})$

- Câu lệnh truy vấn

SELECT* FROM VACCINE WHERE YEAR(HANSĐ) = YEAR(GETDATE())

14. Đưa ra mã khách hàng đã có phiếu tiêm hoặc dưới 30 tuổi

- Đại số quan hệ:

$\pi_{\text{maKH}}(\sigma_{\text{tuoi} < 30}(\text{KhachHang})) \cap \pi_{\text{maKH}}(\sigma(\text{KhachHang}))$

- Câu lệnh truy vấn

SELECT MAKH FROM KHACHHANG WHERE GIOITINH =N'NỮ' OR MAKH IN (SELECT MAKH FROM PHIEUTIEM)

15. Đưa ra các người giám hộ có giới tính là "Nữ" và giám hộ cho khách hàng có tên là 'Nguyễn'

- Đại số quan hệ:

$\pi_{\text{MAGH}, \text{TENKH}}(\sigma_{\text{NGUOIGIAMHO.MAGH} = \text{KHACHHAGN.MAGH AND GIOITINH} = \text{'Nữ' AND TENKH} = \text{'%Nguyễn'}}(\text{GIAMHO, KHACHHANG}))$

- Câu lệnh truy vấn:

SELECT GIAMHO.MAGH, GIAMHO.HOTEN FROM GIAMHO, KHACHHANG
WHERE NGUOIGIAMHO.MAGH=KHACHHANG.MAGH
AND GIOITINH LIKE N'Nữ'
AND TENKH LIKE N'%Nguyễn'

16. Cho biết tất cả số lô có nhập loại vaccine ‘Cảm cúm’

- Đại số quan hệ

$\pi_{\text{SOLO}}(\sigma_{\text{VACCINE.MALOAIVC} = \text{LOAIVC.MALOAIVC AND LOAIVACCINE} = \text{'CẢM CÚM'}}(\text{VACCINE x LOAIVC}))$

- Câu lệnh truy vấn

SELECT SOLO AS SOVACCINE FROM VACCINE, LOAIVC
WHERE VACCINE.MALOAIVC=LOAIVC.MALOAIVC
AND LOAIVACCINE=N'Cảm cúm'

17. Cho biết tên thu ngân viết phiếu tiêm có mã phiếu tiêm là ‘PT005’

- Đại số quan hệ

$\pi_{\text{THUNGAN.HOTEN}}(\sigma_{\text{HOADON.MATHUNGAN} = \text{THUNGAN.MATHUNGAN AND MAPHIEUTIE} = \text{'PT005'}}(\text{HOADON x THUNGAN}))$

- Câu lệnh truy vấn

```
SELECT HOTEN FROM HOADON, THUNGAN
WHERE HOADON.MATHUNGAN=THUNGAN.MATHUNGAN
AND MAPHIEUTIE = 'PT005'
```

18. Cho biết số phiếu tiêm có mã loại vaccine là ‘LVC001’

- Đại số quan hệ

$\exists_{\text{COUNT(MAPHIEUTIE)}}(\sigma_{\text{VACCINE.MAVACCINE} = \text{CHITIEETIE.MAVACCINE AND MALOAIVC} = \text{'LVC001'}}(\text{VACCINE x CHITIEETIE}))$

- Câu lệnh truy vấn

```
SELECT COUNT(MAPHIEUTIE) AS SOPHIEUTIE FROM VACCINE, CHITIEETIE
WHERE VACCINE.MAVACCINE=CHITIEETIE.MAVACCINE
AND MALOAIVC='LVC001'
```

19. Cho biết những người bác sĩ ở Hồ Chí Minh

- Đại số quan hệ:

$\sigma_{(\text{DIACHI} = \text{'%TP HCM'})}(\text{BACSY})$

- Câu lệnh truy vấn

```
SELECT * FROM BACSY WHERE DIACHI LIKE N'%TP HCM%'
```

20. Tính tổng số tiền của vaccine có trong phiếu tiêm có mã là ‘PT005’

- Đại số quan hệ

$\exists_{\text{SUM(DONGIA)}}(\sigma_{\text{CHITIEETIE.MAVACCINE} = \text{VACCINE.MAVACCINE AND MAPHIEUTIE} = \text{'PT005'}}(\text{CHITIEETIE x VACCINE}))$

- Câu lệnh truy vấn

```
SELECT SUM(DONGIA) AS TONGTIE FROM CHITIEETIE, VACCINE
WHERE CHITIEETIE.MAVACCINE=VACCINE.MAVACCINE
AND MAPHIEUTIE = 'PT005'
```

3.2.3.2. Truy vấn nâng cao

1. Tính tổng số phiếu tiêm mỗi bác sĩ có chức vụ là ‘Trưởng khoa’

```
SELECT BACSY.MABS, BACSY.HOTEN, COUNT(PHIEUTIE.MAPHIEUTIE)
FROM BACSY JOIN PHIEUTIE ON PHIEUTIE.MABS=BACSY.MABS
WHERE BACSY.CHUCVU=N'TRƯỞNG KHOA'
GROUP BY BACSY.MABS, BACSY.HOTEN
```

2. Xuất các địa chỉ của bác sĩ kèm số lượng bác sĩ

```
SELECT DIACHI, SUM(MABS) AS SOBACSY FROM BACSY
```

GROUP BY DIACHI

3. Đưa ra mã loại, tên loại, số lượng Vacxin theo từng loại

```
SELECT LOAIVC.MALOAIVC, LOAIVACCINE, COUNT(LOAIVC.MALOAIVC)
AS SOLUONG FROM VACCINE
JOIN LOAIVC ON LOAIVC.MALOAIVC = VACCINE.MALOAIVC
GROUP BY LOAIVC.MALOAIVC, LOAIVACCINE
```

4. Cho biết tổng số khách hàng họ Nguyễn

```
SELECT COUNT(MAKH) AS 'SOLUONG' FROM KHACHHANG
WHERE TENKHACHHANG LIKE N'NGUYỄN%'
```

5. Đưa ra danh sách bác sĩ là TP và mã phiếu tiêm mà bác sĩ tiêm nếu có

```
SELECT BACSY.MABS, BACSY.HOTEN, MAPHIEUTIEM
FROM BACSY LEFT JOIN PHIEUTIEM ON BACSY.MABS=PHIEUTIEM.MABS
WHERE BACSY.MABS=N'TRƯỜNG PHÒNG'
```

6. Đưa ra mã quản kho, tên quản kho nhập số lô 25

```
SELECT MAQUANKHO, HOTEN FROM VACCINE, QUANKHO
WHERE QUANKHO.MAQUANKHO=VACCINE.MAQUANKHO
AND VACCINE.SOLO='25'
```

7. Đếm số lượng vaccine theo từng mã loại vaccine

```
SELECT LOAIVC.MALOAIVC, SUM (VACCINE.SOLUONGCOSAN) AS SL FROM LOAIVC
LEFT JOIN VACCINE ON VACCINE.MALOAIVC = LOAIVC.MALOAIVC
GROUP BY LOAIVC.MALOAIVC
```

8. Lấy tất cả thông tin về vaccine và loại vaccine

```
SELECT vc.MAVACCINE, vc.TENVACCINE, vc.NHASX, vc.NGAYSX, vc.HANSO, vc.SOLO,
vc.SOLUONGCOSAN, vc.DONGIA, lvc.LOAIVACCINE
FROM VACCINE vc INNER JOIN LOAIVC lvc ON lvc.MALOAIVC = vc.MALOAIVC
```

9. Đưa ra các vaccine cách hạn sử dụng 100 ngày

```
SELECT MAVACCINE, TENVACCINE, DATEDIFF(DAY, GETDATE(), HANSO) AS
'SONGAYCONLAI' FROM VACCINE
WHERE DATEDIFF(DAY, GETDATE(), HANSO)<100
```

10. Cho biết tổng tiền của phiếu tiêm có bác sĩ là 'Trưởng khoa'

```
SELECT MABS, SUM(TONGTIEN) AS TONGTIEN
FROM PHIEUTIEM, HOADON, BACSY
WHERE PHIEUTIEM.MAPHIEUTIEM=HOADON.MAPHIEUTIEM
AND HOADON.MABS=BACSY.MABS
AND CHUCVU= N'Trưởng Khoa'
GROUP BY MABS
```

3.2.3.3. Truy vấn TSQL

1. Tạo trigger xóa phiếu tiêm

```
CREATE TRIGGER XOAPHIEUTIRM
ON PHIEUTIEM
FOR DELETE
```

```

AS
BEGIN
    DECLARE @MAPHIEUTIEM CHAR(8)
    SELECT @MAPHIEUTIEM = DELETED.MAPHIEUTIEM FROM DELETED
    DELETE CHITIETIEM WHERE @MAPHIEUTIEM = MAPHIEUTIEM
END

```

2. Tìm kiếm bệnh nhân và Phiếu khám, trong ngày với các thông tin Mã khách hàng, Tên khách hàng, Địa chỉ, Tên Người giám hộ.

```

CREATE PROC sp_TimKHTrongNgay
(
    @maKH NVARCHAR(3),
    @hoTenKH NVARCHAR(30),
    @ngayHomNay DATE,
    @diaChi NVARCHAR(150),
    @tenGH NVARCHAR(30)
)
AS
BEGIN
    SELECT KHACHHANG.MAKH, KHACHHANG.TENKH, PHIEUTIEM.MaPhIEUTIEM, PK.KetLuan
    FROM KHACHHANG JOIN PhieuTIEM AS PK ON KHACHHANG.MAKHACHHANG =
    PHIEUTIEM.MAKHACHHANG WHERE
        ( BN.MAKH LIKE '%' + @maKH + '%' OR @maKH = '' ) AND
        ( BN.TENKH LIKE '%' + @TENKH + '%' OR @TENKH = '' ) AND
        ( BN.TenGIAMHO LIKE '%' + @tenGH + '%' OR @tenGH = '' ) AND
        ( BN.DiaChi LIKE '%' + @diaChi + '%' OR @diaChi = '' ) AND
        PHIEUTIEM.NGAYTIEM = @ngayHomNay
    ORDER BY BN.MaKH ASC
END

```

3. Tạo hàm chỉnh sửa vaccine

```

CREATE PROC sp_ChinhSuaVaccine(@MAVACCINE CHAR(10), @TENVACCINE NVARCHAR(20),
@NHASX NVARCHAR(30), @NGAYSX DATE, @HANS DATE,
    @SOLO CHAR(10), @SOLUONGCOSAN INT, @DONGIA FLOAT, @LOAIVC NVARCHAR(20),
@MAQUANKHO CHAR(10))
AS
BEGIN
    DECLARE @MALOAIVC CHAR(10)
    IF EXISTS (SELECT LOAIVACCINE FROM LOAIVC WHERE LOAIVACCINE=@LOAIVC)
    BEGIN
        SELECT @MALOAIVC=MALOAIVC FROM LOAIVC WHERE LOAIVACCINE=@LOAIVC
    END
    ELSE
    BEGIN
        DECLARE @LASTMALVC CHAR(10)
        DECLARE @NEXTMALVC CHAR(10)
    END

```

```

SELECT @LASTMALVC = MALOAIVC FROM (SELECT TOP(1) MALOAIVC FROM
LOAIVC ORDER BY MALOAIVC DESC) AS LASTMALVC
SELECT @NEXTMALVC=CONCAT('LVC00', SUBSTRING(@LASTMALVC, 6,10)+1)
PRINT @NEXTMALVC
INSERT INTO LOAIVC (MALOAIVC, LOAIVACCINE)
VALUES (@NEXTMALVC, @LOAIVC)
SELECT @MALOAIVC = MALOAIVC FROM LOAIVC WHERE
LOAIVACCINE=@LOAIVC
END

UPDATE VACCINE
SET
    TENVACCINE = @TENVACCINE,
    NHASX = @NHASX,
    NGAYSX = @NGAYSX,
    HANSX = @HANSX,
    SOLO = @SOLO,
    SOLUONGCOSAN = @SOLUONGCOSAN ,
    DONGIA = @DONGIA,
    MALOAIVC = @MALOAIVC,
    MAQUANKHO = @MAQUANKHO
WHERE MAVACCINE = @MAVACCINE
END

```

4. Lấy tất cả phiếu tiêm trong ngày

```

CREATE PROC sp_GetAllPhieuTiemTrongNgay
AS
BEGIN
    SELECT * FROM PHIEUTIEM
    WHERE NGAYTIEM=GETDATE()
END

```

5. Lấy tất cả thông tin tài khoản

```

CREATE PROC sp_getAllTaiKhoan( @maTK CHAR(10), @pass Char(100))
AS
BEGIN
    IF EXISTS (SELECT * FROM QUANTRI WHERE MAQUANTRI = @maTK) AND EXISTS
    (SELECT * FROM TAIKHOAN WHERE MATAIKHOAN=@maTK AND HASHPASS=@pass)

    BEGIN
        SELECT MATAIKHOAN FROM TAIKHOAN
    END
    ELSE
    BEGIN
        SELECT * FROM TAIKHOAN WHERE 1=0
    END
END

```

6. Xóa tài khoản

```

CREATE TRIGGER trg_XoaTaiKhoan ON TAIKHOAN INSTEAD OF DELETE AS

```

```

BEGIN
    DECLARE @MATK CHAR(10)
    SET @MATK = (SELECT MATAIKHOAN FROM DELETED)
    DELETE FROM TAIKHOAN WHERE MATAIKHOAN=@MATK
    DELETE FROM THUNGAN WHERE MATHUNGAN=@MATK
    DELETE FROM BACSY WHERE MABS=@MATK
    DELETE FROM QUANKHO WHERE MAQUANKHO=@MATK
    DELETE FROM QUANTRI WHERE MAQUANTRI = @MATK
END

```

7. Lấy thông tin toàn bộ phiếu tiêm

```

CREATE PROC sp_GetPhieuTiemReportInfo (@MAPT CHAR(10))
AS
BEGIN
    DECLARE @tbBSPT TABLE (MAPT CHAR(10), NGAYTIEM DATE, TENBS NVARCHAR(50),
    MAKH CHAR(10))

    INSERT INTO @tbBSPT SELECT pt.MAPHIEUTIEM, pt.NGAYTIEM, bs.HOTEN,
    pt.MAKH FROM PHIEUTIEM pt INNER JOIN BACSY bs ON bs.MABS = pt.MABS WHERE
    pt.MAPHIEUTIEM = @MAPT

    SELECT bspt.NGAYTIEM, kh.MAKH, kh.TENKH, kh.NGAYSINH, kh.GIOITINH,
    kh.TIEUSU, bspt.TENBS FROM KHACHHANG kh INNER JOIN @tbBSPT bspt INNER JOIN
    CHITIEETTIEM ctt INNER JOIN VACCINE vc
    ON vc.MAVACCINE = ctt.MAVACCINE
    ON bspt.MAPT = ctt.MAPHIEUTIEM
    ON kh.MAKH = bspt.MAKH
END

```

8. Lấy thông tin hóa đơn theo mã hóa đơn

```

CREATE PROC spRP_getInforHoaDon (@maHD CHAR(10))
AS
BEGIN
    SELECT
        CHIETKHAU AS 'ChietKhaU',
        NGAYTHU AS 'NgayThu',
        TONGTIEN AS 'TongTien',
        MAPHIEUTIEM AS 'MaPhieuTiem',
        MATHUNGAN AS 'MaThuNgan'
    FROM HOADON
    WHERE MAHOADON = @maHD
END

```

9. Lấy tên bệnh nhân bằng mã phiếu tiêm

```

CREATE PROC spRP_GetTenBenhNhan(@maPhieuTiem CHAR(10))
AS
BEGIN
    SELECT DISTINCT TENKH AS 'TenBN'
    FROM PHIEUTIEM INNER JOIN KHACHHANG ON KHACHHANG.MAKH = PHIEUTIEM.MAKH
    WHERE MAPHIEUTIEM = @maPhieuTiem

```

END

10.Xóa các phiếu tiêm và hóa đơn của bác sĩ theo mã bác sĩ dùng con trỏ

```
CREATE PROC XOABACSY(@MABS char (10))
AS
BEGIN
    DECLARE @maPhieuTIEM CHAR (10)
    --tạo cursor để xóa các phiếu tiêm mà bác sĩ đó làm
    DECLARE PT_cursor CURSOR FORWARD_ONLY FOR SELECT MaPhieuTIEM FROM
    PHIEUTIEM WHERE MaBS = @MABS
    OPEN PT_cursor
    FETCH FIRST FROM TRO INTO @MA
    WHILE @@FETCH_STATUS = 0
    BEGIN
        DELETE PHIEUTIEM WHERE MaPHIEUTIEM = @maPhieuTIEM
        DELETE HOADON WHERE HOADON.MAPHIEUTIEM = @MAPHIEUTIEM
    END
    CLOSE PT_cursor
    DEALLOCATE PT_cursor
END
```

3.2.4. Đình Thị Thu Uyên

3.2.4.1. Truy vấn cơ bản

1. Cho biết mã vaccine, tên vaccine, tên loại vaccine của các vaccine có trong phiếu tiêm có mã là 'PT002'

- Đại số quan hệ

$$\pi_{\text{MAVACCINE, VACCINE.TENVACCINE, LOAIVC.LOAIVACCINE}} (\sigma_{\text{VACCINE.MAVACCINE = CHITIETTIEM.MAVACCINE AND VACCINE.MALOAIVC = LOAIVC.MALOAIVC AND MAPHIEUTIEM = 'PT001'}} (\text{VACCINE} \times \text{CHITIETTIEM} \times \text{LOAIVC}))$$

- Câu lệnh truy vấn

```
SELECT VACCINE.MAVACCINE, VACCINE.TENVACCINE, LOAIVC.LOAIVACCINE FROM
VACCINE, CHITIETTIEM, LOAIVC
WHERE VACCINE.MAVACCINE = CHITIETTIEM.MAVACCINE
AND VACCINE.MALOAIVC=LOAIVC.MALOAIVC
AND MAPHIEUTIEM = 'PT001'
```

2. Cho biết tên những vaccine phòng bệnh ho gà

- Đại số quan hệ:

$$\pi_{\text{TENVACCINE}} (\sigma_{\text{VACCINE.MALOAIVC=LOAIVC.MALOAIVC AND LOAIVACCINE= '%ho gà%'}} (\text{VACCINE} \times \text{LOAIVC}))$$

- Câu lệnh truy vấn:

```
SELECT TENVACCINE FROM VACCINE, LOAIVC
```

WHERE VACCINE.MALOAIVC=LOAIVC.MALOAIVC
AND LOAIVACCINE LIKE N'%ho gà%'

3. Cho biết tên bác sĩ và số điện thoại của bác sĩ có chức vụ là 'TRƯỞNG PHÒNG'

- Đại số quan hệ

$\pi_{HOTEN,SDT}(\sigma_{CHUCVU='TRƯỞNG PHÒNG' BACSY})$

- Câu lệnh truy vấn

SELECT HOTEN, SDT FROM BACSY
WHERE CHUCVU=N'TRƯỞNG PHÒNG'

4. Cho biết những khách bác sĩ sinh vào năm 2001

- Đại số quan hệ

$\sigma_{Year(NGAYSINH) = 2001} (BACSY)$

- Câu lệnh truy vấn

SELECT * FROM BACSY WHERE YEAR(NGAYSINH) = 2001

5. Cho biết các phiếu tiêm có ngày tiêm nhắc lại vào ngày 20-10-2022

- Đại số quan hệ:

$\sigma_{(ngayHen = '2020-10-20')} (PT_ChiTiet)$

- Câu lệnh truy vấn:

SELECT * FROM dbo.PT_ChiTiet WHERE NgayHenTiem = '2022-10-20'

6. Cho biết tổng số vaccine còn lại của loại vaccine có tên là 'Ho gà'

- Đại số quan hệ

$\exists_{SUM(SOLUONGCOSAN)}(\sigma_{VACCINE.MALOAIVC = LOAIVC.MALOAIVC \text{ AND } LOAIVACCINE='HO GÀ'} (VACCINE \times LOAIVC))$

- Câu lệnh truy vấn

SELECT SUM(SOLUONGCOSAN) AS SOLUONGCOSAN FROM VACCINE, LOAIVC
WHERE VACCINE.MALOAIVC=LOAIVC.MALOAIVC
AND LOAIVACCINE=N'Ho gà'

7. Tính tổng đơn giá vaccine của vaccine có số lô 100

- Đại số quan hệ

$\exists_{SUM(DONGIA)}(\sigma_{SOLO=100} VACCINE)$

- Câu lệnh truy vấn

SELECT SUM(DONGIA) FROM VACCINE
WHERE SOLO=100

8. Đếm số bác sĩ có bằng cấp là tiến sĩ

- Đại số quan hệ

$\mathfrak{J}_{\text{COUNT(MABS)}}(\sigma_{\text{BANGCAP}=\text{'TIẾN SĨ'}} \text{BACSY})$

- Câu lệnh truy vấn

```
SELECT COUNT(MABS) FROM BACSY
WHERE BANGCAP = N'Tiến sĩ'
```

9. Cho biết tất cả số lô có nhập loại vaccine ‘Ho gà’

- Đại số quan hệ

$\pi_{\text{SOLO}}(\sigma_{\text{VACCINE.MALOIVC}=\text{LOAIVC.MALOIVC AND LOAIVACCINE}=\text{'HO GÀ'}} (\text{VACCINE x LOAIVC}))$

- Câu lệnh truy vấn

```
SELECT SOLO AS SOVACCINE FROM VACCINE, LOAIVC
WHERE VACCINE.MALOIVC=LOAIVC.MALOIVC
AND LOAIVACCINE=N'HO GÀ'
```

10. Đưa ra mã giám hộ, tên giám hộ của khách hàng có họ ‘Đinh’

- Đại số quan hệ

$\pi_{\text{MAGH,TENKH}}(\sigma_{\text{KHACHHANG.MAGH}=\text{GIAMHO.MAGH AND TENKH}=\text{'ĐINH\%'}} (\text{KHACHHANG x GIAMHO}))$

- Câu lệnh truy vấn

```
SELECT MAKH, TENKH FROM KHACHHANG, GIAMHO
WHERE KHACHHANG.MAGH=GIAMHO.MAGH
AND TENKH LIKE N'Đinh%'
```

11. Đếm số người giám hộ ở Đắc Lắc

- Đại số quan hệ:

$\mathfrak{J}_{\text{COUNT}}(\sigma_{\text{DIACHI}=\text{'\%Đắc Lắc'}} (\text{NGUOIGIAMHO}))$

- Câu lệnh truy vấn

```
SELECT COUNT(MAGH) FROM NGUOIGIAMHO WHERE DIACHI LIKE N'%Đắc Lắc'
```

12. Đếm số thủ kho là nữ và tham gia nhập vaccine:

- Đại số quan hệ:

$\mathfrak{J}_{\text{COUNT}}(\sigma_{\text{QUANKHO.MAQUANKHO}=\text{VACCINE.MAQUANKHO AND GIOITINH}=\text{'NỮ'}} (\text{QUANKHO x VACCINE}))$

- Câu lệnh truy vấn

```
SELECT COUNT(MAQUANKHO) FROM QUANKHO, VACCINE
WHERE QUANKHO.MAQUANKHO= VACCINE.MAQUANKHO
```


AND GIOITINH = N'NỮ'

13. Đưa ra các khách hàng có ngày sinh trong khoảng 1-1-2022 đến 10-05-2022

- Đại số quan hệ:

σ NGAYSINH BETWEEN '2022-01-01' AND '2022-05-10' (KHACHHANG)

- Câu lệnh truy vấn:

```
SELECT * FROM KHACHHANG
WHERE NGAYSINH BETWEEN '2022-01-01' AND '2022-05-10'
```

14. Cho biết mã hóa đơn của khách hàng có tên khách hàng là 'Uyên'

- Đại số quan hệ:

π MAHOADON (σ KHACHHANG.MAKH=HOADON.MAKHACHHANG AND TENKH = '%Uyên') (KHACHHANG x HOADON))

- Câu lệnh truy vấn

```
SELECT MAHOADON FROM KHACHHANG, HOADON
WHERE KHACHHANG.MAKH = HOADON.MAKH
AND MAGH= 'GH001'
```

15. Tính tổng số tiền của khách hàng có tên là 'Uyên'

- Đại số quan hệ

Σ SUM(GIABAN) (σ CHITIETTIEM.MAKH=KHACHHANG.MAKH AND TENKH='%UYÊN' (VACCINE x KHACHHANG))

- Câu lệnh truy vấn

```
SELECT SUM(GIABAN) FROM CHITIETTIEM, KHACHHANG
WHERE KHACHHANG.MAKH= CHITIETTIEM.MAKH
AND TENKH= N'Uyên'
```

16. Cho biết các vaccine có số lô là 25 và số lượng còn lại nhỏ hơn 200

- Đại số quan hệ

π MAVACCINE, TENVACCINE (σ SOLO = 25 AND SOLUONGCONLAI < 200 (VACCINE))

- Câu lệnh truy vấn

```
SELECT MAVACCINE, TENVACCINE FROM VACCINE
WHERE SOLO = 25
AND SOLUONGCONLAI < 200
```

17. Đưa ra mã hóa đơn được viết bởi thu ngân có mã thu ngân là 'TN001':

- Đại số quan hệ:

$\pi_{\text{NgàyThu}} (\sigma_{\text{MaNV}='NV01'} (\text{HOADON}))$

- Câu lệnh truy vấn:

```
SELECT MAHOADON FROM HOADON WHERE MATHUNGAN='TN001'
```

18. Cho biết các quản kho có họ Đinh và chức vụ = 'Thủ quỹ'

- Đại số quan hệ

$\sigma_{(\text{ChucVu} = \text{'Thủ quỹ'} \text{ AND } \text{hoTen} = \text{'Đinh\%'})} (\text{QUANKHO})$

- Câu lệnh truy vấn

```
SELECT * FROM QUANKHO WHERE HOTEN LIKE N'Đinh%' AND CHUCVU = N'Thủ quỹ'
```

19. Kiểm tra xem có bao nhiêu quản trị đã có tài khoản

- Đại số quan hệ

$\exists \text{COUNT}(\text{TAIKHOAN.MATAIKHOAN})(\sigma_{\text{TAIKHOAN.MATAIKHOAN} = \text{QUANTRI.MAQUANTRI AND } (\text{QUANTRI x TAIKHOAN}))$

- Câu lệnh truy vấn

```
SELECT COUNT(MAQUANTRI) FROM QUANTRI, TAIKHOAN
WHERE QUANTRI.MAQUANTRI=TAIKHOAN.MATAIKHOAN
```

20. Đếm xem có bao nhiêu phiếu tiêm có sử dụng vaccine 'Astra'

- Đại số quan hệ

$\exists \text{COUNT}(\sigma_{\text{VACCINE.MAVACCINE} = \text{PHIEUTIEM.MAVACCINE AND } \text{TENVACCINE} = \text{'ASTRA'}} (\text{VACCINE x PHIEUTIEM}))$

- Câu lệnh truy vấn

```
SELECT COUNT(MAPT) FROM VACCINE, PHIEUTIEM
WHERE VACCINE.MAVACCINE=PHIEUTIEM.MAVACCINE
AND TENVACCINE=N'Astra'
```

3.2.4.2. Truy vấn nâng cao

1. Xuất các vùng xuất xử vaccine kèm số lượng

```
SELECT NHASX, SUM(SOLUONGCOSAN) AS SOLUONG FROM VACCINE
GROUP BY NHASX
```

1. Đếm số vaccine theo loại vaccine

```
SELECT LOAIVC.LOAIVACCINE, SOLUONG
FROM LOAIVC, (SELECT VC.MALOAIVC AS 'MALOAI', SUM(SOLUONGCOSAN) AS 'SoLuong'
FROM VACCINE vc INNER JOIN LOAIVC lvc ON lvc.MALOAIVC = vc.MALOAIVC
GROUP BY VC.MALOAIVC) AS MALVC_SL
WHERE LOAIVC.MALOAIVC=MALVC_SL.MALOAI
ORDER BY LOAIVC.LOAIVACCINE ASC
```

3. Cho biết các khách hàng có tổng thanh toán lớn hơn 200000

```
SELECT MAKH , SUM(GIABAN) FROM CHITIENTIEM
GROUP BY MAKH
HAVING SUM(GIABAN)>200000
```

4. Tính tổng số phiếu tiêm của từng bác sĩ

```
SELECT MABS, COUNT(MABS)
FROM PHIEUTIEM
GROUP BY MABS
```

5. Thêm trường tổng số hóa đơn vào bảng thu ngân

```
CREATE TABLE THUNGAN
ADD TONGHOADON INT
GO
SELECT * FROM THUNGAN
UPDATE THUNGAN
SET TONGHOADON = (SELECT COUNT(MAHOADON)
                  FROM HOADON
                  GROUP BY MATHUNGAN
                  HAVING HOADON.MATHUNGAN=THUNGAN.MATHUNGAN)
```

6. Xóa thu ngân k viết hóa đơn nào

```
DELETE THUNGAN
WHERE MATHUNGAN NOT IN (SELECT MATHUNGAN FROM HOADON)
```

7. Đưa ra danh sách bác sĩ là trưởng phòng và mã phiếu tiêm mà bác sĩ tiêm nếu có

```
SELECT BACSY.MABS, BACSY.HOTEN, PHIEUTIEM.MAPHIEUTIEM
FROM BACSY LEFT JOIN PHIEUTIEM ON PHIEUTIEM.MABS=BACSY.MABS
WHERE BACSY.CHUCVU=N'TRƯỞNG PHÒNG'
```

8. Đưa ra các vaccine mới sản xuất trong vòng 50 ngày

```
SELECT MAVACCINE, TENVACCINE, DATEDIFF(DAY, NGAYSX, GETDATE()) AS 'SONGAY'
FROM VACCINE
WHERE DATEDIFF(DAY, NGAYSX, GETDATE())<50
```

9. Cho biết tổng tiền của phiếu tiêm có mã bác sĩ là 'BS002'

```
SELECT MABS, SUM(TONGTIEN) AS TONGTIEN
FROM PHIEUTIEM, HOADON
WHERE PHIEUTIEM.MAPHIEUTIEM=HOADON.MAPHIEUTIEM
AND MABS= 'BS002'
```

10. Đưa ra bác sĩ tiêm nhiều phiếu tiêm nhất

```
SELECT *
FROM BACSY, ( SELECT TOP(1)BS.MABS, COUNT(PHIEUTIEM.MAPHIEUTIEM) AS
SOPHIEUTIEM
FROM BACSY AS BS, PHIEUTIEM
WHERE BS.MABS=PHIEUTIEM.MABS
GROUP BY BS.MABS
ORDER BY COUNT(PHIEUTIEM.MAPHIEUTIEM) DESC
```

```

) AS BS1
WHERE BACSY.MABS=BS1.MABS

```

3.2.4.3 Truy vấn TSQL

1. Tạo thủ tục thêm tài khoản

```

CREATE PROC sp_InsertTaiKhoan (@MATK CHAR(10), @HOTEN NVARCHAR(30),
@NGAYSINH DATE, @SDT CHAR(10), @DIACHI NVARCHAR(50), @CHUCVU NVARCHAR(30),
@CHUYENKHOA NVARCHAR(30), @BANGCAP NVARCHAR(30))
AS
BEGIN
    IF @CHUCVU = N'BÁC SỸ'
        BEGIN
            INSERT INTO BACSY (MABS, HOTEN,NGAYSINH, SDT, DIACHI,
CHUCVU, CHUYENKHOA, BANGCAP)
            VALUES (@MATK, @HOTEN, @NGAYSINH, @SDT, @DIACHI, @CHUCVU,
@CHUYENKHOA, @BANGCAP)
        END
    IF @CHUCVU = N'THU NGÂN'
        BEGIN
            INSERT INTO THUNGAN (MATHUNGAN, HOTEN,NGAYSINH, SDT,
DIACHI, CHUCVU)
            VALUES (@MATK, @HOTEN, @NGAYSINH, @SDT, @DIACHI, @CHUCVU)
        END
    IF @CHUCVU = N'QUẢN KHO'
        BEGIN
            INSERT INTO QUANKHO (MAQUANKHO, HOTEN,NGAYSINH, SDT,
DIACHI, CHUCVU)
            VALUES (@MATK, @HOTEN, @NGAYSINH, @SDT, @DIACHI, @CHUCVU)
        END
    IF @CHUCVU = N'QUẢN TRỊ'
        BEGIN
            INSERT INTO QUANTRI (MAQUANTRI, HOTEN,NGAYSINH, SDT,
DIACHI, CHUCVU)
            VALUES (@MATK, @HOTEN, @NGAYSINH, @SDT, @DIACHI, @CHUCVU)
        END
END

```

2. Tạo thủ tục sửa thông tin tài khoản

```

CREATE PROC sp_UpdateTaiKhoanInfo (@MATK CHAR(10), @HOTEN NVARCHAR(30),
@NGAYSINH DATE, @SDT CHAR(10), @DIACHI NVARCHAR(50), @CHUCVU NVARCHAR(30),
@CHUYENKHOA NVARCHAR(30), @BANGCAP NVARCHAR(30))
AS
BEGIN
    IF @CHUCVU = N'BÁC SỸ'
        BEGIN
            UPDATE BACSY
            SET

```

```

        HOTEN=@HOTEN,
        NGAYSINH=@NGAYSINH,
        SDT = @SDT,
        DIACHI=@DIACHI,
        CHUYENKHOA=@CHUYENKHOA,
        BANGCAP=@BANGCAP
        WHERE MABS=@MATK;
    END
    IF @CHUCVU = N'THU NGÂN'
    BEGIN
        UPDATE THUNGAN
        SET
            HOTEN=@HOTEN,
            NGAYSINH=@NGAYSINH,
            SDT = @SDT,
            DIACHI=@DIACHI
        WHERE MATHUNGAN=@MATK;
    END
    IF @CHUCVU = N'QUẢN KHO'
    BEGIN
        UPDATE QUANKHO
        SET
            HOTEN=@HOTEN,
            NGAYSINH=@NGAYSINH,
            SDT = @SDT,
            DIACHI=@DIACHI
        WHERE MAQUANKHO=@MATK;
    END
    IF @CHUCVU = N'QUẢN TRỊ'
    BEGIN
        UPDATE QUANTRI
        SET
            HOTEN=@HOTEN,
            NGAYSINH=@NGAYSINH,
            SDT = @SDT,
            DIACHI=@DIACHI
        WHERE MAQUANTRI=@MATK;
    END
END
END

```

3. Tạo thủ tục đếm số vaccine theo loại vaccine

```

CREATE PROC sp_CountVCTheoLoaiVC
AS
BEGIN
    SELECT LOAIVC.LOAIVACCINE, SOLUONG
    FROM LOAIVC, (SELECT VC.MALOAIVC AS 'MALOAI', SUM(SOLUONGCOSAN) AS
'SoLuong' FROM VACCINE vc INNER JOIN LOAIVC lvc ON lvc.MALOAIVC = vc.MALOAIVC
    GROUP BY VC.MALOAIVC) AS MALVC_SL

```

```

WHERE LOAIVC.MALOAIVC=MALVC_SL.MALOAI
ORDER BY LOAIVC.LOAIVACCINE ASC
END

```

4. Tạo thủ tục lấy số vaccine được dùng nhiều nhất trong một thời gian

```

CREATE PROC sp_GetMostUsedVaccineINTIME (@NgayDau DATE, @NgayCuoi DATE)
AS
BEGIN
    SELECT MAVACCINE AS 'MaVC', COUNT(MAVACCINE) AS 'SoLuong'
    FROM PHIEUTIAM INNER JOIN CHITIETTIEM ON CHITIETTIEM.MAPHIEUTIAM =
    PHIEUTIAM.MAPHIEUTIAM
    WHERE CHITIETTIEM.MAPHIEUTIAM IN
        (SELECT DISTINCT MAPHIEUTIAM FROM HOADON WHERE NGAYTHU BETWEEN
@NgayDau AND @NgayCuoi)
    GROUP BY MAVACCINE
    ORDER BY SoLuong DESC
END

```

5. Tạo thủ tục lấy vaccine sắp hết hạn

```

CREATE PROC sp_GetVaccineINTIME
AS
BEGIN
    SELECT MAVACCINE, TENVACCINE, DATEDIFF(DAY, GETDATE(), HANSĐ) AS
'SONGAYCONLAI' FROM VACCINE
    WHERE DATEDIFF(DAY, GETDATE(), HANSĐ)<50
END

```

6. Tạo thủ tục lấy thông tin khách hàng sắp đến ngày tiêm lại

```

CREATE PROC sp_GetKhachHangINTIME
AS
BEGIN
    SELECT KHACHHANG.MAKH, TENKH, DATEDIFF(DAY, GETDATE(), NGAYTIEMNHACLAI)
AS 'SONGAYTIEMLAI' FROM KHACHHANG, CHITIETTIEM, PHIEUTIAM
    WHERE KHACHHANG.MAKH=PHIEUTIAM.MAKH
    AND CHITIETTIEM.MAPHIEUTIAM=PHIEUTIAM.MAPHIEUTIAM
    AND DATEDIFF(DAY, GETDATE(), NGAYTIEMNHACLAI) >0
    AND DATEDIFF(DAY, GETDATE(), NGAYTIEMNHACLAI) <100
END

```

7. Tạo thủ tục lấy thông tin hóa đơn trong một thời gian

```

CREATE PROC sp_GetHoaDonINTIME_TK (@NgayDau DATE, @NgayCuoi DATE)
AS
BEGIN
    SELECT NGAYTHU AS "Ngày", TONGTIEN AS "Tien"
    FROM HOADON
    WHERE NGAYTHU BETWEEN @NgayDau AND @NgayCuoi
    ORDER BY NGAYTHU ASC

```

END

8. Tạo thủ tục lấy doanh thu theo ngày

```
CREATE PROC sp_GetDoanhThuTheoNgayINTIME(@NgayDau DATE, @NgayCuoi DATE)
AS
BEGIN
    SELECT NGAYTHU AS "Ngày", SUM(TONGTIEN) AS "Tien"
    FROM HOADON
    WHERE NGAYTHU BETWEEN @NgayDau AND @NgayCuoi
    GROUP BY(NGAYTHU)
    ORDER BY NGAYTHU ASC
END
```

9. Tạo trigger xóa phiếu tiêm

```
CREATE TRIGGER trg_xoaPT ON PHIEUTIEM INSTEAD OF DELETE AS
BEGIN
    DECLARE @deletedMAPT CHAR(10)
    SET @deletedMAPT = (SELECT MAPHIEUTIEM FROM Deleted)

    DELETE FROM dbo.HOADON WHERE MAPHIEUTIEM = @deletedMAPT
    DELETE FROM dbo.CHITIETTIEM WHERE MAPHIEUTIEM = @deletedMAPT

    DELETE FROM dbo.PHIEUTIEM WHERE MAPHIEUTIEM = @deletedMAPT
END
```

10. Tìm kiếm vaccine theo nhà sản xuất

```
CREATE PROC sp_SearchByNhaSX (@Value NVARCHAR(100))
AS
BEGIN
    SELECT vc.MAVACCINE, vc.TENVACCINE, vc.NHASX, vc.NGAYSX, vc.HANSX,
    vc.SOLO, vc.SOLUONGCOSAN, vc.DONGIA, lvc.LOAIVACCINE
    FROM VACCINE vc INNER JOIN LOAIVC lvc ON lvc.MALOAIVC = vc.MALOAIVC
    WHERE vc.NHASX LIKE N'%' + @Value + '%'
END
```

CHƯƠNG IV: CÀI ĐẶT ỨNG DỤNG

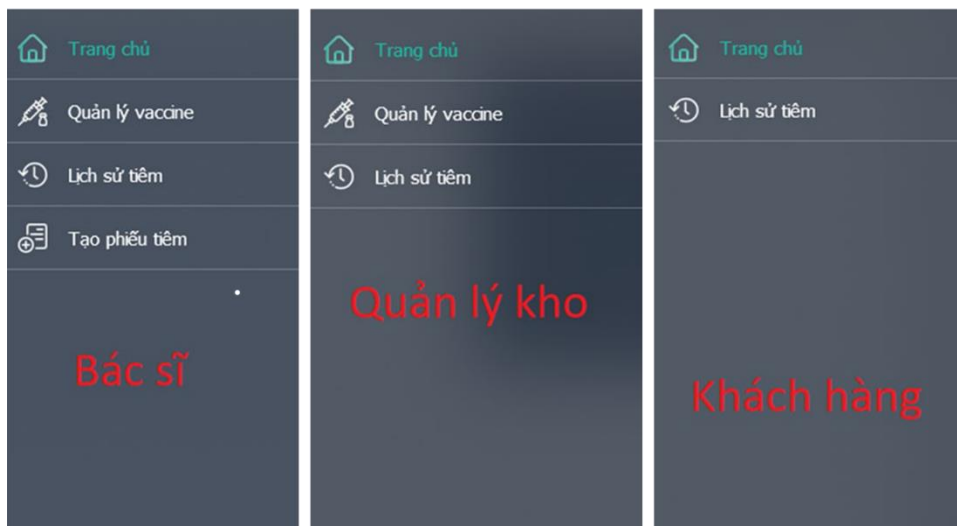
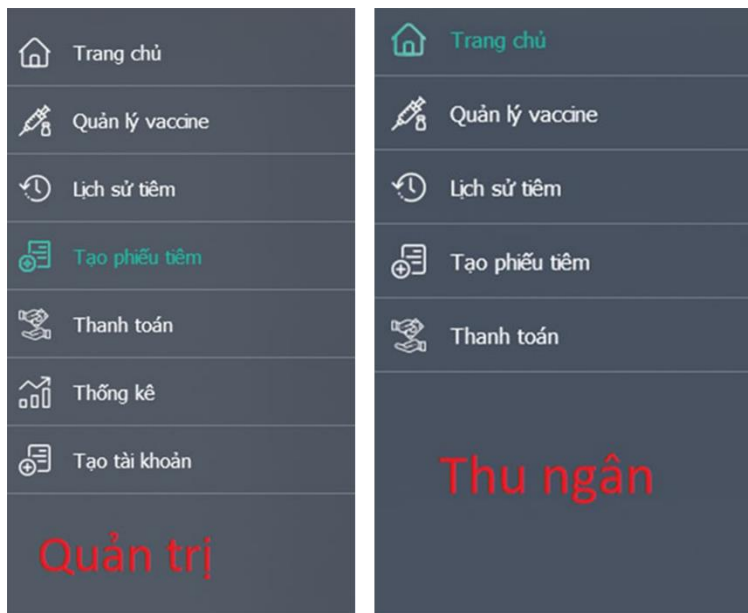
4.1. Thiết kế giao diện

4.1.1. Yêu cầu thiết kế

- Chương trình thiết kế phải đảm bảo những yêu cầu sau:
 - + Giao diện dễ hiểu dễ sử dụng.
 - + Phân quyền người dùng để đảm bảo tính bảo mật của hệ thống
 - + Đảm bảo tìm kiếm thông tin dễ dàng

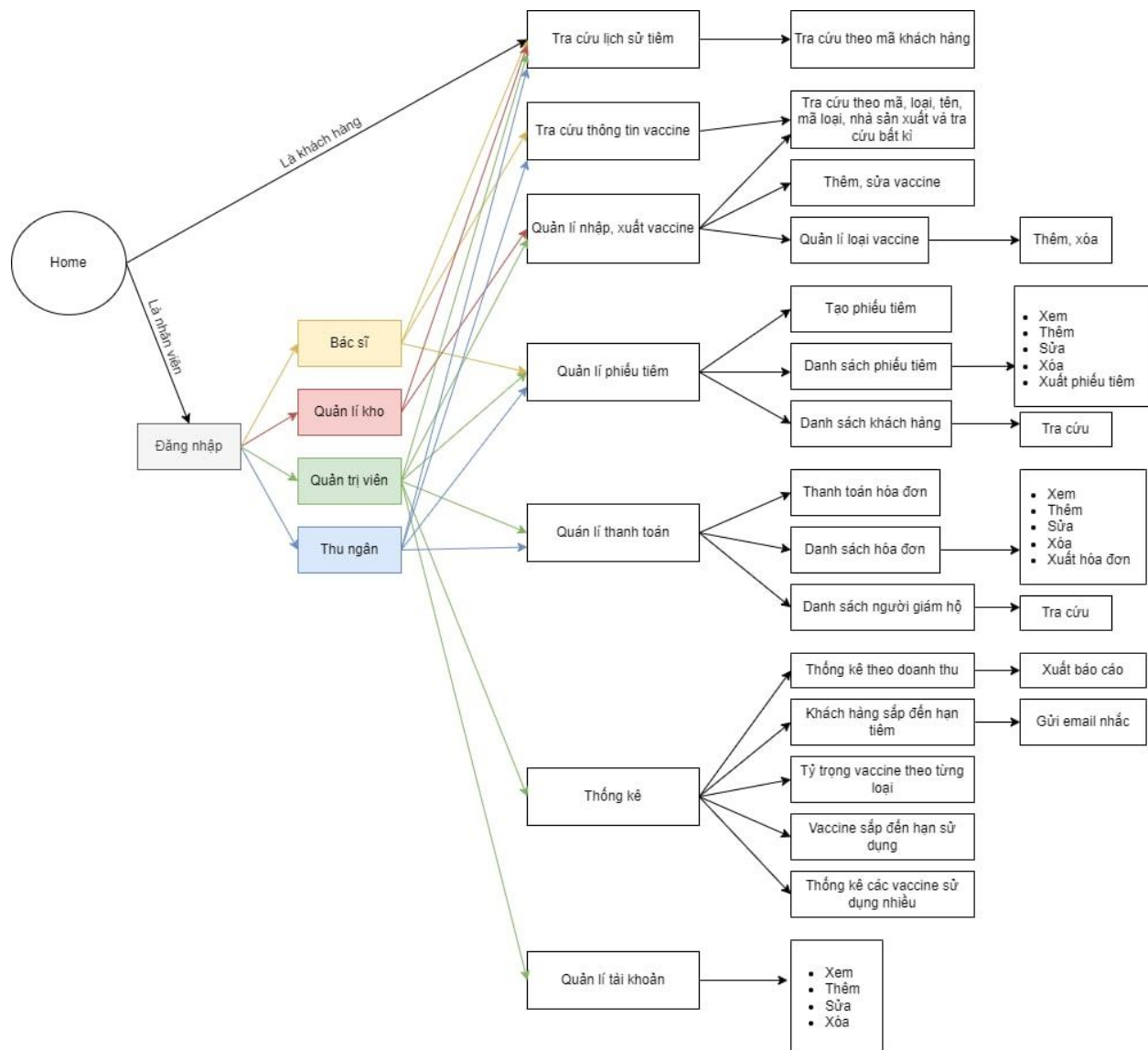
4.1.2. Phân quyền

- Khách hàng: là người không cần đăng nhập tài khoản, tuy nhiên có thể tra cứu lịch sử tiêm chủng qua mã khách hàng.
- Bác sĩ: là người có quyền truy xuất các thông tin vaccine, quản lý và tạo phiếu tiêm cũng như tra cứu lịch sử tiêm của khách hàng, hiển thị thống kê khách hàng đến hạn tiêm cũng như gửi email thông báo cho khách hàng.
- Quản lý kho: là người có quyền quản lý, nhập xuất vaccine cũng như tra cứu lịch sử tiêm của khách hàng.
- Thu ngân: là người có quyền truy xuất các thông tin vaccine, quản lý và tạo phiếu tiêm cũng như có quyền tạo hóa đơn thanh toán phiếu tiêm.
- Quản trị: là người có mọi quyền đối với hệ thống, bao gồm các quyền của quản lý và các quyền cập nhật danh sách người dùng, cấp quyền cho tài khoản xóa tài khoản, cập nhật tham số hệ thống. Ngoài ra, quản trị còn có thể thông kê doanh thu, top vaccine được sử dụng trong khoảng thời gian chỉ định,...



Menu hiển thị theo người dùng

4.1.3. Mô hình giao diện tổng thể của hệ thống



4.2. Các chức năng chính

- Chương trình bao gồm các chức năng chính sau:

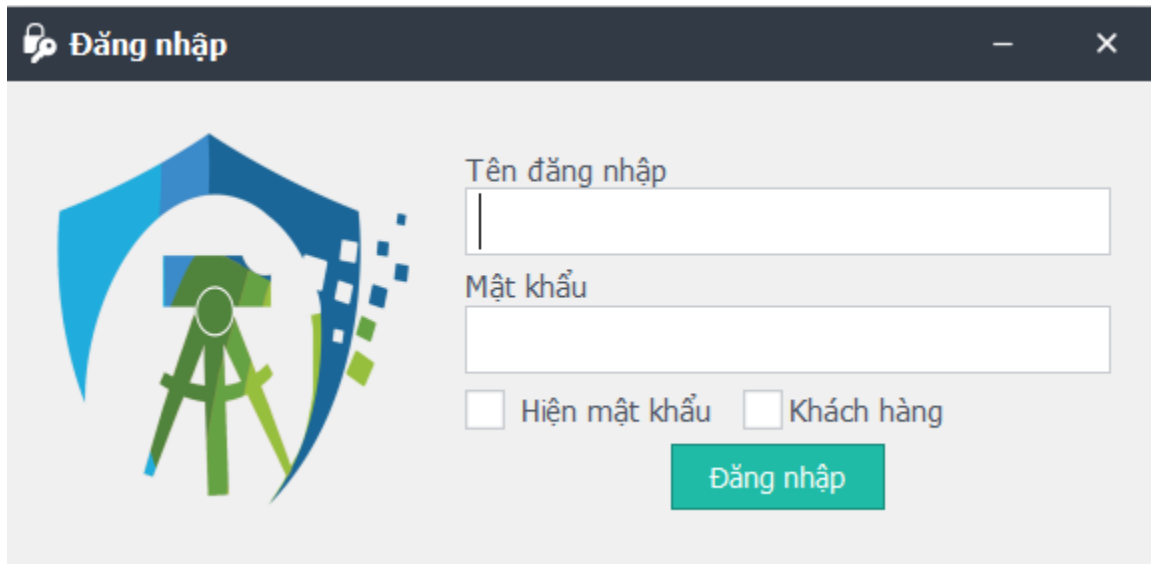
4.2.1. Giao diện Đăng nhập

a. Chức năng đăng nhập

+ Người dùng đăng nhập với username và password của mình, sẽ được cấp quyền tương ứng để truy cập hệ thống.

+ Truy cập với quyền khách hàng không cần đăng nhập qua checkbox.

b. Thiết kế giao diện



The screenshot shows a login window with a dark blue header bar containing the title 'Đăng nhập' and standard window controls. The main area has a light gray background. On the left is a logo consisting of a blue shield-like shape with a green stylized figure inside. On the right, there are two text input fields labeled 'Tên đăng nhập' and 'Mật khẩu'. Below these fields are two checkboxes: 'Hiện mật khẩu' and 'Khách hàng'. At the bottom right is a green button with the text 'Đăng nhập'.

Giao diện đăng nhập

c. Các lệnh SQL phục vụ cho chức năng

- Stored procedure Infor_TK: Kiểm tra tồn tại tài khoản có tham số tên đăng nhập @user và mật khẩu @hashpass và đưa ra thông tin nếu có.

```
CREATE PROC [dbo].[Infor_TK]
@user CHAR(10),
@hashpass CHAR(100)
AS
BEGIN
    SELECT * FROM TAIKHOAN
    WHERE MATAIKHOAN = @user
    AND HASHPASS = @hashpass
END
```

4.2.2. Giao diện quản lý người dùng

a. Chức năng quản lý người dùng

+ Chức năng này chỉ người quản trị được sử dụng để cập, sửa, xóa tài khoản người dùng. Khi nhấn vào chức năng này, danh sách tất cả cả tài khoản hiện thị trên datagridview. Khi nhấn vào một dòng trên datagridview này, thông tin tài khoản đó được đổ ra các textbox. Nếu quản trị muốn cập tài khoản thì nhập đầy đủ các thông tin tài khoản đó, sau đó nhấn vào button thêm tài khoản để cập tài khoản.

+ Nếu chỉ đăng nhập với quyền thông thường (quyền user), thì trong giao diện này chỉ hiện thông tin của user đó. Còn quyền thêm, xóa hay cấp quyền sẽ không được phép thực hiện.

+ Khi quản trị muốn sửa thông tin tài khoản thì nhấn vào dòng chứa thông tin tài khoản muốn sửa đổi ở datagridview, sau đó nhập thông tin sửa đổi vào các textbox và nhấn button Sửa tài khoản để cập nhật lại thông tin tài khoản. Ở phần mật khẩu, do mật khẩu ở đã được chuyển thành mã hash nên không thể lấy được mật khẩu lúc đầu, do vậy nếu để trống mục này thì mật khẩu sẽ không thay đổi, còn nếu nhập mật khẩu, thì nó sẽ tự tạo mã hash mới tương ứng với mật khẩu đó lưu vào cơ sở dữ liệu. Tuy nhiên, ở sửa tài khoản thì hai thông tin là chức vụ và mã thành viên sẽ không được chỉnh sửa, nếu vẫn muốn chỉnh sửa thì phải xóa tài khoản đó và thêm lại từ đầu. Nếu muốn xóa tài khoản người dùng, nhấn vào dòng chứa thông tin tài khoản muốn xóa tài khoản, sau đó nhấn Xóa để tiến hành xóa tài khoản.

b. Thiết kế giao diện

The screenshot shows a web application interface for account management. The interface is divided into three main sections: a sidebar, a top header, and a main content area.

Sidebar: Contains navigation links for Trang chủ, Quản lý vaccine, Lịch sử tiêm, Tạo phiếu tiêm, Thanh toán, Thống kê, and Tạo tài khoản.

Top Header: Displays 'Xin chào Quản trị' and 'Đăng xuất'.

Main Content Area:

- Thông tin tài khoản:** A form for adding or editing an account. It includes fields for Chức vụ (Bác sĩ), Họ và tên, Ngày sinh (18/05/2022), Mã thành viên, Mật khẩu, Địa chỉ, Số điện thoại, Bằng cấp, and Chuyên khoa. There are buttons for 'Thêm tài khoản', 'Sửa tài khoản', 'Xóa tài khoản', and 'Đặt lại'.
- Danh sách tài khoản:** A table listing accounts with columns: MA, HOTEN, NGAYSINH, SDT, DIACHI, and CHUCVU. The table contains 15 rows of data, including accounts for 'Trần Văn Tùng', 'Trung Bảo Lươn', 'Bảo Lươn Trung', 'Bác Sĩ Nguyễn', 'Phạm Đình Mạnh', 'Lươn Bảo Trung', 'Quản trị', 'Hồ Thu Ngân', 'Phan Giám Sát', 'Thu Ngân Nguyễn', and 'Mai Thu Ngân'.

Giao diện quản lý tài khoản

c. Các lệnh SQL phục vụ cho chức năng

- Stored procedure `sp_getAllTaiKhoan`: SELECT toàn bộ các tài khoản có trong hệ thống có tham số `@maTK`, `@pass`. Hai tham số này nhằm mục đích xác thực đây là tài khoản quản trị, ở đây tham số là `@pass` nhưng đây là một mã hash của mật khẩu người dùng và trong cơ sở dữ liệu cũng lưu mật khẩu ở dưới dạng mã hash như vậy để tránh lộ lọt cơ sở dữ liệu thì hacker cũng không thể suy ra được mật khẩu thật sự của người dùng.

```
CREATE PROC [dbo].[sp_getAllTaiKhoan] ( @maTK CHAR(10), @pass Char(100))
AS
BEGIN
    IF EXISTS (SELECT * FROM QUANTRI WHERE MAQUANTRI = @maTK) AND EXISTS
    (SELECT * FROM TAIKHOAN WHERE MATAIKHOAN=@maTK AND HASHPASS=@pass)
        BEGIN
            SELECT MATAIKHOAN FROM TAIKHOAN
        END
    ELSE
        BEGIN
            SELECT MATAIKHOAN FROM TAIKHOAN WHERE 1=0
        END
    END
```

- Stored procedure `sp_GetHashpassFromMa`: được sử dụng để mã hash của mật khẩu của tài khoản có mã tài khoản là `@maUser`. Ngoài ra, chúng ta phải cung cấp thêm hai tham số mã quản trị `@maQT` và mật khẩu quản trị `@passQT` để xác thực đây là quản trị có quyền lấy thông tin, chỉnh sửa tài khoản.

```
CREATE PROC [dbo].[sp_GetHashpassFromMa] ( @maQT CHAR(10), @passQT Char(100),
@maUser CHAR(10))
AS
BEGIN
    IF EXISTS (SELECT * FROM QUANTRI WHERE MAQUANTRI = @maQT) AND EXISTS
    (SELECT * FROM TAIKHOAN WHERE MATAIKHOAN=@maQT AND HASHPASS=@passQT)
        BEGIN
            SELECT HASHPASS FROM TAIKHOAN where MATAIKHOAN = @maUser
        END
    ELSE
        BEGIN
            SELECT HASHPASS FROM TAIKHOAN WHERE 1=0
        END
    END
```

- Stored procedure `sp_CheckExistanceTaiKhoan`: được sử dụng để kiểm tra có tồn tại tài khoản với mã tài khoản là `@MATK` không.

```
CREATE PROC [dbo].[sp_heckExistanceTaiKhoan] ( @maTK CHAR(10))
```

```

AS
BEGIN
    SELECT * FROM TAIKHOAN WHERE MATAIKHOAN = @MaTK
END

```

- Stored procedure sp_GetGeneralInfoTaiKhoan: SELECT các thông tin chung nhất của tài khoản có tham số @MATK

```

CREATE PROC [dbo].[SP_GetAllGeneralInfoTaiKhoan] (@maTK CHAR(10), @pass Char(100))
AS
BEGIN
    IF EXISTS (SELECT * FROM QUANTRI WHERE MAQUANTRI = @maTK) AND EXISTS (SELECT *
    FROM TAIKHOAN WHERE MATAIKHOAN=@maTK AND HASHPASS=@pass)
        BEGIN
            SELECT MATHUNGAN AS MA, HOTEN, NGAYSINH, SDT, DIACHI, CHUCVU FROM THUNGAN
            UNION
            SELECT MABS AS MA, HOTEN, NGAYSINH, SDT, DIACHI, CHUCVU FROM BACSY
            UNION
            SELECT MAQUANKHO AS MA, HOTEN, NGAYSINH, SDT, DIACHI, CHUCVU FROM QUANKHO
            UNION
            SELECT MAQUANTRI AS MA, HOTEN, NGAYSINH, SDT, DIACHI, CHUCVU FROM QUANTRI
        END
    ELSE
        BEGIN
            SELECT MAQUANTRI AS MA, HOTEN, NGAYSINH, SDT, DIACHI, CHUCVU FROM QUANTRI Where
            1=0
        END
    END
END

```

- Stored procedure sp_GetQuanKhoInfo: SELECT toàn bộ thông tin của quản lý kho có tham số @MAQUANKHO

```

CREATE PROC [dbo].[sp_GetQuanKhoInfo]( @MAQUANKHO CHAR(10))
AS
BEGIN
    SELECT MAQUANKHO, HOTEN, NGAYSINH, SDT, DIACHI, CHUCVU FROM QUANKHO WHERE
    MAQUANKHO = @MAQUANKHO
END

```

- Stored procedure sp_GetQuanTriInfo: SELECT toàn bộ thông tin của quản trị viên có tham số @MAQUANTRI

```

CREATE PROC [dbo].[sp_GetQuanTriInfo] (@MAQUANTRI CHAR(10))
AS
BEGIN
    SELECT MAQUANTRI, HOTEN, NGAYSINH, SDT, DIACHI, CHUCVU FROM QUANTRI WHERE
    MAQUANTRI=@MAQUANTRI
END

```

- Stored procedure sp_GetBacSyInfo: SELECT toàn bộ thông tin của bác sĩ có tham số @MABS

```
CREATE PROC [dbo].[sp_GetBacSyInfo] (@MABS CHAR(10))
AS
BEGIN
    SELECT MABS, HOTEN, NGAYSINH, SDT, DIACHI, CHUCVU, CHUYENKHOA, BANGCAP FROM
    BACSY WHERE MABS=@MABS
END
```

- Stored procedure sp_GetThuNganInfo: SELECT toàn bộ thông tin của nhân viên thu ngân có tham số @MATHUNGAN

```
CREATE PROC [dbo].[sp_GetThuNganInfo] ( @MATHUNGAN CHAR(10))
AS
BEGIN
    SELECT MATHUNGAN, HOTEN, NGAYSINH, SDT, DIACHI, CHUCVU FROM THUNGAN WHERE
    MATHUNGAN=@MATHUNGAN
END
```

- Stored procedure sp_InsertTaiKhoan : được sử dụng khi thêm một tài khoản với các tham số nhận giá trị từ các textbox nhập vào . Ngoài ra người dùng cũng cần phải truyền hai tham số @maTK, @pass, đây là mã tài khoản và mã băm mật khẩu của người quản trị có quyền thêm tài khoản.

```
CREATE PROC [dbo].[sp_InsertTaiKhoan] (@maQT CHAR(10), @passQT Char(100), @MATK
CHAR(10), @HOTEN NVARCHAR(30), @NGAYSINH DATE, @SDT CHAR(10), @DIACHI
NVARCHAR(50), @CHUCVU NVARCHAR(30), @CHUYENKHOA NVARCHAR(30), @BANGCAP
NVARCHAR(30), @PASSWORD char(100))
AS
BEGIN
    IF EXISTS (SELECT * FROM QUANTRI WHERE MAQUANTRI = @maQT) AND EXISTS
    (SELECT * FROM TAIKHOAN WHERE MATAIKHOAN=@maQT AND HASHPASS=@passQT)
    BEGIN
        INSERT INTO TAIKHOAN (MATAIKHOAN, HASHPASS)
        VALUES (@MATK, @PASSWORD)
        IF @CHUCVU = N'BÁC SỸ'
        BEGIN
            INSERT INTO BACSY (MABS, HOTEN, NGAYSINH, SDT, DIACHI, CHUCVU, CHUYENKHOA, BANGCAP)
            VALUES (@MATK, @HOTEN, @NGAYSINH, @SDT, @DIACHI, @CHUCVU, @CHUYENKHOA, @BANGCAP)
        END
        IF @CHUCVU = N'THU NGÂN'
        BEGIN
            INSERT INTO THUNGAN (MATHUNGAN, HOTEN, NGAYSINH, SDT, DIACHI, CHUCVU)
            VALUES (@MATK, @HOTEN, @NGAYSINH, @SDT, @DIACHI,
            @CHUCVU)
        END
        IF @CHUCVU = N'QUẢN LÝ KHO'
        BEGIN
```

```

INSERT INTO QUANKHO (MAQUANKHO, HOTEN,NGAYSINH, SDT, DIACHI, CHUCVU)
                VALUES (@MATK, @HOTEN, @NGAYSINH, @SDT, @DIACHI,
@CHUCVU)

                END
        IF @CHUCVU = N'QUẢN TRỊ'
        BEGIN
INSERT INTO QUANTRI (MAQUANTRI, HOTEN,NGAYSINH, SDT, DIACHI, CHUCVU)
                VALUES (@MATK, @HOTEN, @NGAYSINH, @SDT, @DIACHI,
@CHUCVU)

                END
        END
END

```

- Stored procedure sp_UpdateTaiKhoanInfo: được sử dụng khi sửa thông tin một tài khoản với các tham số nhận giá trị từ các textbox nhập vào tại tài khoản có @MaTK là mã tài khoản được click tại một dòng trong datagridview. Ở đây, vẫn cần thêm hai tham số là @MaQT và @passQT để xác thực quản trị viên đang công việc cập nhật dữ liệu.

```

CREATE PROC [dbo].[sp_UpdateTaiKhoanInfo] (@maQT CHAR(10), @passQT Char(100),@MATK
CHAR(10), @HOTEN NVARCHAR(30), @NGAYSINH DATE, @SDT CHAR(10), @DIACHI
NVARCHAR(50), @CHUCVU NVARCHAR(30), @CHUYENKHOA NVARCHAR(30), @BANGCAP
NVARCHAR(30), @PASSWORD CHAR(100))
AS
BEGIN
        IF EXISTS (SELECT * FROM QUANTRI WHERE MAQUANTRI = @maQT) AND EXISTS
        (SELECT * FROM TAIKHOAN WHERE MATAIKHOAN=@maQT AND HASHPASS=@passQT)
        BEGIN
                UPDATE TAIKHOAN
                SET
                        HASHPASS=@PASSWORD
                WHERE MATAIKHOAN=@MATK
                IF @CHUCVU = N'BÁC SỸ'
                BEGIN
                        UPDATE BACSY
                        SET
                                HOTEN=@HOTEN,
                                NGAYSINH=@NGAYSINH,
                                SDT = @SDT,
                                DIACHI=@DIACHI,
                                CHUYENKHOA=@CHUYENKHOA,
                                BANGCAP=@BANGCAP
                        WHERE MABS=@MATK;
                END
                IF @CHUCVU = N'THU NGÂN'
                BEGIN
                        UPDATE THUNGAN
                        SET
                                HOTEN=@HOTEN,

```



```

        NGAYSINH=@NGAYSINH,
        SDT = @SDT,
        DIACHI=@DIACHI
    WHERE MATHUNGAN=@MATK;
END
IF @CHUCVU = N'QUẢN LÍ KHO'
BEGIN
    UPDATE QUANKHO
    SET
        HOTEN=@HOTEN,
        NGAYSINH=@NGAYSINH,
        SDT = @SDT,
        DIACHI=@DIACHI
    WHERE MAQUANKHO=@MATK;
END
IF @CHUCVU = N'QUẢN TRỊ'
BEGIN
    UPDATE QUANTRI
    SET
        HOTEN=@HOTEN,
        NGAYSINH=@NGAYSINH,
        SDT = @SDT,
        DIACHI=@DIACHI
    WHERE MAQUANTRI=@MATK;
END
END
END

```

- Stored procedure sp_DeleteTaiKhoan: được sử dụng khi xóa thông tin một tài khoản với các tham số nhận giá trị từ các textbox nhập vào tại tài khoản có @username là username được click tại một dòng trong datagridview.

```

CREATE PROC [dbo].[sp_DeleteTaiKhoan] (@MATK CHAR(10))
AS
BEGIN
    DELETE FROM TAIKHOAN WHERE MATAIKHOAN=@MATK
    DELETE FROM THUNGAN WHERE MATHUNGAN=@MATK
    DELETE FROM BACSY WHERE MABS=@MATK
    DELETE FROM QUANKHO WHERE MAQUANKHO=@MATK
    DELETE FROM QUANTRI WHERE MAQUANTRI = @MATK
END

```

4.2.3. Giao diện Lịch sử tiêu

a. Chức năng Lịch sử tiêu

+ Chức năng này có thể được sử dụng được cho mọi đối tượng kể cả khách hàng.

+ Chức năng này giúp khách hàng vào trung tâm tiêm chủng có thể tự mình tra cứu lịch sử tiêm của bản thân mình. Ngoài ra, nhân viên có thể tra cứu lịch sử tiêm của khách hàng để đưa ra thông tin tư vấn cho khách hàng.

+ Chức năng này cần nhập mã khách hàng và ấn vào button Tìm kiếm để tra cứu. Nhân viên có thể tra cứu mã khách hàng có trong giao diện tạo phiếu tiêm.

b. Thiết kế giao diện

NGAYTIEM	TENVACCINE	LOAIVACCINE	NHASX	MUITHU	LIEULUONG	NGAYTIEMNHACLAIR	DONGIA
7/18/2022	Astra	covid	Anh	3	1	10/18/2022	100001
5/18/2022	BCG	Lao	Việt Nam	1	1	8/18/2022	120000

Giao diện lịch sử tiêm

c. Các lệnh SQL phục vụ cho chức năng

- Stored procedure sp_GetAllLichSuTiem: được sử dụng để Select toàn bộ lịch sử tiêm vaccine của một khách hàng có tham số @MAKH

```
CREATE PROC [dbo].[sp_GetAllLichSuTiem] (@MAKH CHAR(10))
AS
BEGIN
    SELECT pt.NGAYTIEM, vc.TENVACCINE, lvc.LOAIVACCINE, vc.NHASX, ctt.MUITHU,
    ctt.LIEULUONG, ctt.NGAYTIEMNHACLAIR, vc.DONGIA
    FROM KHACHHANG kh INNER JOIN PHIEUTIEM pt INNER JOIN CHITIETTIEM ctt INNER
    JOIN VACCINE vc INNER JOIN LOAIVC lvc
    ON lvc.MALOAIVC = vc.MALOAIVC
```

```

ON vc.MAVACCINE = ctt.MAVACCINE
ON ctt.MAPHIEUTIEM = pt.MAPHIEUTIEM
ON pt.MAKH = kh.MAKH
WHERE kh.MAKH = @MAKH
END

```

4.2.4. Giao diện Quản lý vaccine

a. Chức năng Quản lý vaccine

+ Giao diện quản lý vaccine tùy theo người dùng. Đối với người dùng quản trị hoặc nhập kho sẽ có mục để nhập xuất kho, cũng như tra cứu thông tin vaccine. Các đối tượng khác chỉ hiển thị tra cứu thông tin vaccine.

+ Thông tin các vaccine được hiển thị lên datagridview. Muốn thêm loại vaccine mới, người dùng nhập đầy đủ thông tin vào các textbox và để trống textbox mã vaccine. Ở vị trí loại vaccine, người dùng sẽ chọn các loại các vaccine ở trong combobox. Để thêm, xóa loại vaccine, ta ấn vào button danh sách loại VC, phần mềm sẽ mở một form mới để quản lý các loại vaccine. Sau khi nhập đầy đủ thông tin, ta ấn vào button Thêm vaccine, phần mềm sẽ tự động tạo mã vaccine mới cho vaccine vừa nhập và hiển thị lên datagridview.

+ Để chỉnh sửa vaccine, người dùng có thể nhập mã vaccine và điền các thông tin khác cho vaccine đó, sau đó ấn vào button chỉnh sửa vaccine để chỉnh sửa. Phần mềm sẽ tìm vaccine theo mã vaccine, nếu tìm thành công sẽ sửa theo thông tin vaccine theo thông tin mình vừa nhập. Ngoài ra, chúng ta có thể tìm vaccine trong kho bằng cách điền thông tin vào ô textbox tìm kiếm và ấn vào button tìm kiếm. Ở đây, phần mềm cho ta tìm kiếm theo các thông tin như tất cả thông tin, mã vaccine, nhà sản xuất, tên vaccine hoặc loại vaccine qua việc click vào các radiobutton. Phần mềm sẽ hiển thị thông tin tìm kiếm ở trên datagridview, khi tìm được vaccine cần chỉnh sửa, ta ấn vào hàng chứa vaccine cần chỉnh sửa đó, phần mềm sẽ tự động điền thông tin vào các ô textbox ở trên về thông tin vaccine được chọn. Từ đó ta có thể chỉnh sửa thông tin bằng cách ấn vào button chỉnh sửa vaccine như bình thường.

b. Thiết kế giao diện

The interface shows a sidebar with navigation options: Trang chủ, Quản lý vaccine, Lịch sử tiêm, Tạo phiếu tiêm, Thanh toán, Thống kê, and Tạo tài khoản. The main area is titled 'Xin chào Quản trị' and 'Đăng xuất'. It contains a 'Nhap kho' form with fields for vaccine name, manufacturer, production date, expiration date, lot number, quantity, unit price, and vaccine type. A dropdown menu for vaccine types is open, showing options like LVC001 - covid, LVC002 - Sởi, LVC003 - Viêm gan B, LVC004 - Thủy đậu, LVC005 - Cúm, LVC006 - Lao, LVC007 - Uốn ván, and LVC008 - Sởi. Below the form is a table of vaccine information.

MAVACCINE	TENVACCINE	NHASX	NGAYSX	DONGIA	LOAIVACCINE
VC001	Astra	Anh	7/5/2020	997	covid
VC002	Covaxin	Mỹ	5/5/2020	500	covid
VC003	rubella	Bi	5/7/2022	397	Sởi
VC004	VAT	Nha Trang	4/17/2019	498	Uốn ván
VC005	vaxgrip	Trung Qu...	7/13/2018	195	Cúm
VC006	Engerix-B	Việt Nam	8/9/2017	199	Viêm gan B
VC007	BCG	Việt Nam	10/5/2018	299	Lao
VC008	Varivax	Ostraylia	6/6/2019	98	Mắc
VC009	H1N1 vaccine	Thái Lan	5/8/2022	654	Cúm
VC010	Indirab	Ấn Độ	2/3/2021	299	Dại
VC011	Verorab	Pháp	5/24/2022	435	Dại

Giao diện quản lý vaccine cho đối tượng quản lý kho, quản trị

The interface shows a sidebar with navigation options: Trang chủ, Quản lý vaccine, Lịch sử tiêm, Tạo phiếu tiêm, Thanh toán, Thống kê, and Tạo tài khoản. The main area is titled 'Xin chào Quản trị' and 'Đăng xuất'. It contains a 'Cập nhật loại vaccine' form with fields for vaccine type code and name. A dropdown menu for vaccine types is open, showing options like LVC001 - covid, LVC002 - Sởi, LVC003 - Viêm gan B, LVC004 - Thủy đậu, LVC005 - Cúm, LVC006 - Lao, LVC007 - Uốn ván, and LVC008 - Sởi. Below the form is a table of vaccine information.

HANSO	SOLO	SOLUONGCOSAN	DONGIA	LOAIVACCINE
8/5/2023	1	997	100001	covid
5/5/2023	2	500	90000	covid
8/5/2024	2	397	70000	Sởi
4/17/2024	2	498	80000	Uốn ván
7/13/2024	3	197	110000	Cúm
8/9/2024	1	199	55000	Viêm gan B
10/10/20...	1	298	120000	Lao
6/6/2024	3	99	200000	Thủy đậu
9/13/2022	3	654	21000	Cúm

Form thêm xóa loại vaccine

Trang chủ

Quản lý vaccine

Lịch sử tiêm

Tạo phiếu tiêm

Thanh toán

Xin chào Hồ Thu Ngân

Đăng xuất

Tra cứu thông tin vaccine

☒ Tất cả

☐ Mã

☐ Tên

☐ Loại

☐ Nhà sx

Tìm kiếm

	MAVACCINE	TENVACCINE	NHASX	NGAYSX	HANSĐ	SOLO	SOLUONGCOSAN	DONGIA	LOAIIVACCINE
→	VC001	Astra	Anh	7/5/2020	8/5/2023	1	997	100001	covid
	VC002	Covaxin	Mỹ	5/5/2020	5/5/2023	2	500	90000	covid
	VC003	rubella	Bi	5/7/2022	8/5/2024	2	397	70000	Sởi
	VC004	VAT	Nha Trang	4/17/2019	4/17/2024	2	498	80000	Uốn ván
	VC005	vaxigrip	Trung Qu...	7/13/2018	7/13/2024	3	197	110000	Cúm
	VC006	Engerix-B	Việt Nam	8/9/2017	8/9/2024	1	199	55000	Viêm gan B
	VC007	BCG	Việt Nam	10/5/2018	10/10/20...	1	298	120000	Lao
	VC008	Varivax	Mỹ	6/6/2019	6/6/2024	3	99	200000	Thủy đậu
	VC009	H1N1 vaccine	Thái Lan	5/8/2022	9/13/2022	3	654	21000	Cúm

Giao diện quản lý vaccine cho các đối tượng khác

c. Các lệnh SQL phục vụ cho chức năng

- Stored procedure `sp_GetAllVaccine`: liệt kê thông tin tất cả các vaccine có trong hệ thống

```
CREATE PROC [dbo].[sp_GetAllVaccine]
AS
BEGIN
    SELECT vc.MAVACCINE, vc.TENVACCINE, vc.NHASX, vc.NGAYSX, vc.HANSX, vc.SOLO,
    vc.SOLUONGCOSAN, vc.DONGIA, lvc.LOAIIVACCINE
    FROM VACCINE vc INNER JOIN LOAIVC lvc ON lvc.MALOAIVC = vc.MALOAIVC
END
```

- Stored procedure `sp_SearchAllVaccine`: được sử dụng để tìm kiếm vaccine có thông tin của một trường giống với tham số `@Value` được truyền vào

```
CREATE PROC [dbo].[sp_SearchAllVaccine] (@Value NVARCHAR(100))
AS
BEGIN
    SELECT vc.MAVACCINE, vc.TENVACCINE, vc.NHASX, vc.NGAYSX, vc.HANSX, vc.SOLO,
    vc.SOLUONGCOSAN, vc.DONGIA, lvc.LOAIIVACCINE
    FROM VACCINE vc INNER JOIN LOAIVC lvc ON lvc.MALOAIVC = vc.MALOAIVC
    WHERE vc.MAVACCINE LIKE N'%' + @Value + '%'
    OR vc.TENVACCINE LIKE N'%' + @Value + '%'
    OR vc.NHASX LIKE N'%' + @Value + '%'
```

```

OR vc.NGAYSX LIKE N'%' + @Value + '%'
OR vc.HANSX LIKE N'%' + @Value + '%'
OR vc.SOLO LIKE N'%' + @Value + '%'
OR vc.SOLUONGCOSAN LIKE N'%' + @Value + '%'
OR lvc.LOAIVACCINE LIKE N'%' + @Value + '%'
OR vc.DONGIA LIKE N'%' + @Value + '%'
END

```

- Stored procedure sp_SearchByLoaiVC: được sử dụng để tìm kiếm vaccine có thông tin có tên loại vaccine trùng với tham số truyền vào @Value

```

CREATE PROC [dbo].[sp_SearchByLoaiVC] (@Value NVARCHAR(100))
AS
BEGIN
    SELECT vc.MAVACCINE, vc.TENVACCINE, vc.NHASX, vc.NGAYSX, vc.HANSX, vc.SOLO,
    vc.SOLUONGCOSAN, vc.DONGIA, lvc.LOAIVACCINE
    FROM VACCINE vc INNER JOIN LOAIVC lvc ON lvc.MALOAIVC = vc.MALOAIVC
    WHERE lvc.LOAIVACCINE LIKE N'%' + @Value + '%'
END

```

- Stored procedure sp_SearchByMaVC: được sử dụng để tìm kiếm vaccine có thông tin có mã vaccine trùng với tham số truyền vào @Value

```

CREATE PROC [dbo].[sp_SearchByMaVC] (@Value char(10))
AS
BEGIN
    SELECT vc.MAVACCINE, vc.TENVACCINE, vc.NHASX, vc.NGAYSX, vc.HANSX, vc.SOLO,
    vc.SOLUONGCOSAN, vc.DONGIA, lvc.LOAIVACCINE
    FROM VACCINE vc INNER JOIN LOAIVC lvc ON lvc.MALOAIVC = vc.MALOAIVC
    WHERE vc.MAVACCINE LIKE N'%' + @Value + '%'
END

```

- Stored procedure sp_SearchByNhaSX: được sử dụng để tìm kiếm vaccine có thông tin có tên nhà sản xuất trùng với tham số truyền vào @Value

```

CREATE PROC [dbo].[sp_SearchByNhaSX] (@Value NVARCHAR(100))
AS
BEGIN
    SELECT vc.MAVACCINE, vc.TENVACCINE, vc.NHASX, vc.NGAYSX, vc.HANSX, vc.SOLO,
    vc.SOLUONGCOSAN, vc.DONGIA, lvc.LOAIVACCINE
    FROM VACCINE vc INNER JOIN LOAIVC lvc ON lvc.MALOAIVC = vc.MALOAIVC
    WHERE vc.NHASX LIKE N'%' + @Value + '%'
END

```

- Stored procedure sp_SearchByTenVC: được sử dụng để tìm kiếm vaccine có thông tin có tên vaccine trùng với tham số truyền vào @Value

```

CREATE PROC [dbo].[sp_SearchByTenVC] (@Value NVARCHAR(100))

```

```

AS
BEGIN
    SELECT vc.MAVACCINE, vc.TENVACCINE, vc.NHASX, vc.NGAYSX, vc.HANSX, vc.SOLO,
    vc.SOLUONGCOSAN, vc.DONGIA, lvc.LOAIVACCINE
    FROM VACCINE vc INNER JOIN LOAIVC lvc ON lvc.MALOAIVC = vc.MALOAIVC
    WHERE vc.TENVACCINE LIKE N'%' + @Value + '%'
END

```

- Stored procedure sp_GetAllLoaiVC: được sử dụng để liệt kê tất cả các loại vaccine có trong cơ sở dữ liệu

```

CREATE PROC [dbo].[sp_GetAllLoaiVC]
AS
BEGIN
    SELECT * FROM LOAIVC
END

```

- Stored procedure sp_InsertVaccine: được sử dụng để thêm một vaccine vào cơ sở dữ liệu

```

CREATE PROC [dbo].[sp_InsertVaccine] (@MAVACCINE CHAR(10), @TENVACCINE
NVARCHAR(20), @NHASX NVARCHAR(30), @NGAYSX DATE, @HANSX DATE,
@SOLO CHAR(10), @SOLUONGCOSAN INT, @DONGIA FLOAT, @MALOAIVC CHAR(10), @MAQUANKHO
CHAR(10), @MAQUANTRI CHAR(10))
AS
BEGIN
    if (TRIM(@MAQUANKHO) = '')
    BEGIN
        set @MAQUANKHO = NULL
    END
    if (TRIM(@MAQUANTRI) = '')
    BEGIN
        set @MAQUANTRI = NULL
    END
    IF EXISTS (SELECT LOAIVACCINE FROM LOAIVC WHERE MALOAIVC=@MALOAIVC)
    BEGIN
        INSERT INTO VACCINE
        (
            MAVACCINE,
            TENVACCINE,
            NHASX,
            NGAYSX,
            HANSX,
            SOLO,
            SOLUONGCOSAN,
            DONGIA,
            MALOAIVC,
            MAQUANKHO,
            MAQUANTRI
        )
    END

```

```

VALUES
(
    @MAVACCINE,
    @TENVACCINE,
    @NHASX,
    @NGAYSX,
    @HANSD,
    @SOLO,
    @SOLUONGCOSAN,
    @DONGIA,
    @MALOAIVC,
    @MAQUANKHO,
    @MAQUANTRI
)
END
END

```

- Stored procedure sp_ChinhSuaVaccine: được sử dụng để cập nhật vaccine một vaccine đã có trong cơ sở dữ liệu

```

CREATE PROC [dbo].[sp_ChinhSuaVaccine](@MAVACCINE CHAR(10), @TENVACCINE
NVARCHAR(20), @NHASX NVARCHAR(30), @NGAYSX DATE, @HANSD DATE,
@SOLO CHAR(10), @SOLUONGCOSAN INT, @DONGIA FLOAT, @MALOAIVC NVARCHAR(20),
@MAQUANKHO CHAR(10), @MAQUANTRI CHAR(10))
AS
BEGIN
    if(TRIM(@MAQUANKHO) = '')
    BEGIN
        set @MAQUANKHO = NULL
    END
    if(TRIM(@MAQUANTRI) = '')
    BEGIN
        set @MAQUANTRI = NULL
    END
    UPDATE VACCINE
    SET
        TENVACCINE = @TENVACCINE,
        NHASX = @NHASX,
        NGAYSX = @NGAYSX,
        HANSD = @HANSD,
        SOLO = @SOLO,
        SOLUONGCOSAN = @SOLUONGCOSAN ,
        DONGIA = @DONGIA,
        MALOAIVC = @MALOAIVC,
        MAQUANKHO = @MAQUANKHO,
        MAQUANTRI = @MAQUANTRI
    WHERE MAVACCINE = @MAVACCINE
END

```


- Stored procedure sp_IsExistLVC: được sử dụng để kiểm tra có tồn tại loại vaccine trong cơ sở dữ liệu không

```
CREATE PROC [dbo].[sp_IsExistLVC] (@MALVC CHAR(10))
AS
BEGIN
    DECLARE @returnVal int
    IF EXISTS (SELECT LOAIVACCINE FROM LOAIVC WHERE MALVC = @MALVC)
    BEGIN
        SET @returnVal = 1
    END
    ELSE SET @returnVal = 0
    RETURN @returnVal
END
```

- Stored procedure sp_GetLastestMaLVC: được sử dụng lấy mã loại vaccine mới nhất có trong cơ sở dữ liệu. Proc này nhằm sinh ra mã loại vaccine tiếp theo.

```
CREATE PROC [dbo].[sp_IsExistLVC] (@MALVC CHAR(10))
AS
BEGIN
    SELECT TOP(1) MALOAIVC FROM LOAIVC ORDER BY MALOAIVC DESC
END
```

- Stored procedure sp_InsertLoaiVC: được sử dụng để chèn thêm loại vaccine mới, với hai tham số truyền vào là @MALOAIVC và @LOAIVACCINE

```
CREATE PROC [dbo].[sp_InsertLoaiVC](
    @MALOAIVC CHAR(10),
    @LOAIVACCINE NVARCHAR(20)
)
AS
BEGIN
    IF EXISTS (SELECT MALOAIVC FROM LOAIVC WHERE MALOAIVC = @MALOAIVC)
    BEGIN
        SELECT * FROM VACCINE
    END
    ELSE
    BEGIN
        INSERT INTO LOAIVC (MALOAIVC, LOAIVACCINE)
        VALUES (@MALOAIVC, @LOAIVACCINE)
    END
END
```

- Stored procedure sp_XoaLoaiVC: được sử dụng để xóa lại vaccine trong cơ sở dữ liệu với tham số truyền vào là @MALOAIVC

```
CREATE PROC [dbo].[sp_XoaLoaiVC] (@MALOAIVC CHAR(10))
AS
BEGIN
    DELETE FROM LOAIVC
```

WHERE MALOAIVC=@MALOAIVC
END

4.2.5. Giao diện Tạo phiếu tiêm

a. Chức năng Tạo phiếu tiêm

- Giao diện tạo phiếu tiêm này được sử dụng cho chủ yếu cho đối tượng là bác sĩ. Ngoài ra, thu ngân cũng như người quản trị cũng có thể sử dụng chức năng này. Bác sĩ khám, kiểm tra tổng quát khách hàng, sau đó tiến hành tạo phiếu tiêm. Phiếu tiêm này sẽ được chuyển tiếp cho thu ngân để tiến hành thanh toán.

- Bắt đầu form, phần mềm sẽ load danh sách các khách hàng và sẽ hiển thị trên datagridview ở phía dưới cùng. Chúng ta có thể tìm kiếm khách hàng có trong datagridview qua một thanh tìm kiếm ở phía trên datagridview. Khi click vào một hàng của datagridview, phần mềm sẽ hiển thị thông tin của khách hàng đó ở trên các textbox ở vị trí thông tin phiếu tiêm. Khi thông tin khách hàng chưa có trong cơ sở dữ liệu, thì khi ta nhập thông tin khách hàng mới trên textbox và tiến hành tạo phiếu tiêm thì thông tin khách hàng cũng sẽ được lưu vào cơ sở dữ liệu. Khi nhập thông tin khách hàng thì chúng ta sẽ để trống mục mã khách hàng, phần mềm sẽ tự tạo mã khách hàng cho chúng ta. Để tạo phiếu tiêm, chúng ta phải nhập đầy đủ thông tin khách hàng, mã bác sĩ tạo phiếu tiêm cũng như vaccine sử dụng. Khi nhập thông tin vaccine như mã vaccine, mũi thứ, liều lượng thì ta sẽ ấn button + để thêm vaccine đó. Khi ta nhập mã vaccine, hệ thống tự gợi ý mã vaccine cho chúng ta. Nếu cần tiêm nhắc lại, ta click vào ô checkbox để chọn thời gian nhắc lại. Khi ta thêm vaccine thì thông tin vaccine như tên vaccine, giá bán sẽ hiển thị trên datagridview. Chúng ta có thể thêm vaccine khác nữa hoặc ấn button trừ để xóa loại vaccine đó. Khi tạo thành công phiếu tiêm, hệ thống sẽ hỏi chúng ta có muốn xuất phiếu tiêm hay không, ấn yes để xuất phiếu tiêm hoặc ấn no để không xuất.

- Chúng ta có thể quản lý phiếu tiêm vaccine qua form cập nhật danh sách phiếu tiêm qua button xem danh sách.

- Trong form cập nhật danh sách phiếu tiêm, thông tin phiếu tiêm sẽ được tải và hiển thị theo thứ tự từ mới nhất trở ngược lại lên datagridview. Khi click vào từng hàng của phiếu tiêm, danh sách vaccine đã được sử dụng tại phiếu tiêm đó sẽ hiển thị. Chúng ta có thể click đúp vào một ô để sửa thông tin hoặc ấn phím delete trên bàn phím để xóa phiếu tiêm đó. Để hủy bỏ việc chỉnh sửa hoặc xóa ấn vào button đặt lại, còn để lưu chỉnh sửa thì ấn vào button lưu. Để xuất phiếu tiêm, ấn vào hàng

phiếu tiêm cần xuất trong datagridview và ấn button xuất. Khi đã chỉnh sửa trong form mà ấn thoát thì hệ thống sẽ hiển thị dialog xem có muốn lưu thông tin hay không trước khi thoát.

b. Thiết kế giao diện

Thông tin phiếu tiêm

Tên khách hàng: Mã khách hàng:

Ngày sinh: Giới tính: Ngày tiêm:

Tiền sử bệnh: Bác sỹ:

Email: Số điện thoại:

Thao tác

Danh sách vaccine sử dụng

Mã vaccine: Mũi thứ: Liều lượng: Nhắc lại sau: tháng

	MAVACCINE	TENVACCINE	GIABAN	MUITHU	LIEULUONG	NGAYNHACLAIR
→	VC006	Engerix-B	55000	1	2	2022-07-27
	VC003	Rubella (MR)	70000	2	2	9999-01-01

	MAKH	TENKH	NGAYSINH	GIOITINH	TIEUSU	EMAIL	SDT
	KH008	Hoàng Thành H...	8/16/2001	Nam	Không	hoangthanhhung...	091872947
	KH007	Nghiêm Quang...	6/19/2003	Nam	Không	ngiemhuy@yah...	033989383
	KH006	Trần Hiếu Tài	6/19/2002	Nam	Không	hieutai1906@gm...	03394983
→	KH005	Nguyễn Quang...	8/14/2001	Nam	Thần kinh	nguyenhung25@...	084626586
	KH004	Nguyễn Tùng Lâm	3/12/2003	Nam	Không	tunglam1203@g...	099128312
	KH003	Hồ Nguyễn Ngu...	10/5/2001	Nam	Không	honguyennguye...	095533211
	KH002	Nguyễn Mạnh C...	6/1/2006	Nam	Cao huyết áp	nguyenmanhcuo...	097104070
	KH001	Đào Đình Huy	5/11/1999	Nam	Không	dinghuy@outlook...	091248748

Giao diện tạo phiếu tiêm

Cập nhật danh sách phiếu tiêm

Danh sách phiếu tiêm

MAPHIEUTIEM	MAKH	TENKH	NGAYSINH	GIOITINH	TIEUSU	NGAYTIEM	TENBS
PT009	KH008	Hoàng Thành Hùng	8/16/2001	Nam	Không	5/18/2022	Bảo Lươn Trung
PT008	KH001	Đào Đình Huy	5/11/1999	Nam	Không	7/5/2022	Phạm Đình Mạnh
PT007	KH008	Hoàng Thành Hùng	8/16/2001	Nam	Không	7/18/2022	Bác Sĩ Nguyễn
PT006	KH007	Nghiêm Quang Huy	6/19/2003	Nam	Không	5/26/2022	Bảo Lươn Trung
PT005	KH006	Trần Hiếu Tài	6/19/2002	Nam	Không	6/8/2022	Bác Sĩ Nguyễn
PT004	KH005	Nguyễn Quang Hùng	8/15/2001	Nam	Không	3/11/2022	Trung Bảo Lươn
PT003	KH004	Nguyễn Tùng Lâm	3/12/2003	Nam	Không	5/7/2022	Bảo Lươn Trung
PT002	KH003	Hồ Nguyễn Nguyễn	10/5/2001	Nam	Không	4/5/2022	Trung Bảo Lươn
→ PT001	KH002	Nguyễn Mạnh Cường	6/1/2006	Nam	Cao huyết áp	6/5/2022	Trần Văn Tùng

Danh sách vaccine đã sử dụng

MAVACCINE	TENVACCINE	LOAVACCINE	NHASX	MUITHU	LIEULUONG	NGAYTIEMNHA...	DONGIA
→ VC001	Astra	covid	Anh	3	1	1/5/2023	100001
VC003	rubella	Sởi	Bì	1	3	12/5/2022	70000

Thoát

?

Đặt lại

Đã lưu

Xuất

Form cập nhật danh sách phiếu tiêm

Trung tâm Tiêm chủng MTAVC
 Địa chỉ: 236 HQV, Bắc Từ Liêm, Hà Nội
 Hotline: 18001357

PHIẾU TIÊM CHỦNG

Mã khách hàng: KH002

PT001

Họ tên khách hàng: Nguyễn Mạnh Cường

Tiền sử bệnh: Cao huyết áp

Ngày sinh: 6/1/2006

Giới tính: Nam

STT	Mã Vaccine	Tên Vaccine	Liều lượng	Đơn giá
1	VC001	Astra	1	100001
2	VC003	rubella	3	70000

Ngày 5 tháng 6 năm 2022

Bác sỹ

Trần Văn Tùng

This application was created using the trial version of the XtraReports.

Mẫu hóa đơn phiếu tiêm được xuất

c. Các lệnh SQL phục vụ cho chức năng

- Stored procedure sp_GetVCConLai: được sử dụng để lấy số lượng vaccine còn lại có mã vaccine là @MAVACCINE

```
CREATE PROC [dbo].[sp_GetVCConLai] (@MAVACCINE NVARCHAR(100))
AS
BEGIN
    SELECT SOLUONGCOSAN FROM VACCINE WHERE MAVACCINE = @MAVACCINE
END
```

- Stored procedure sp_GetDonGiaVC: được sử dụng để lấy giá tiền của vaccine có mã vaccine là @MAVACCINE

```
CREATE PROC [dbo].[sp_GetDonGiaVC] (@MAVACCINE NVARCHAR(100))
AS
BEGIN
    SELECT DONGIA FROM VACCINE WHERE MAVACCINE = @MAVACCINE
END
```

- Stored procedure sp_GetTenVC: được sử dụng để lấy tên của vaccine có mã vaccine là @MAVACCINE

```
CREATE PROC [dbo].[sp_GetTenVC] (@MAVACCINE NVARCHAR(100))
AS
BEGIN
    SELECT TENVACCINE FROM VACCINE WHERE MAVACCINE = @MAVACCINE
END
```

- Stored procedure sp_IsVCInStock: được sử dụng để kiểm tra tồn tại vaccine trong cơ sở dữ liệu hay không

```
CREATE PROC [dbo].[sp_IsVCInStock] (@MAVACCINE CHAR(10))
AS
BEGIN
    DECLARE @returnVal int
    IF EXISTS (SELECT MAVACCINE FROM VACCINE WHERE MAVACCINE = @MAVACCINE)
    BEGIN
        SET @returnVal = 1
    END
    ELSE SET @returnVal = 0
    RETURN @returnVal
END
```

- Stored procedure sp_GetAllVaccine: liệt kê thông tin tất cả các vaccine có trong hệ thống

```
CREATE PROC [dbo].[sp_GetAllVaccine]
AS
BEGIN
    SELECT vc.MAVACCINE, vc.TENVACCINE, vc.NHASX, vc.NGAYSX, vc.HANSX, vc.SOLO,
    vc.SOLUONGCOSAN, vc.DONGIA, lvc.LOAIVACCINE
```

```

FROM VACCINE vc INNER JOIN LOAIVC lvc ON lvc.MALOAIVC = vc.MALOAIVC
END

```

- Stored procedure sp_GetAllKH: liệt kê thông tin tất cả các khách hàng có trong hệ thống

```

CREATE PROC [dbo].[sp_GetAllKH]
AS
BEGIN
    SELECT * FROM KHACHHANG
END

```

- Stored procedure sp_IsExistsMaKH: được sử dụng để kiểm tra tồn tại khách hàng trong cơ sở dữ liệu hay không

```

CREATE PROC [dbo].[sp_IsExistsMaKH] (@MAKH CHAR(10))
AS
BEGIN
    DECLARE @returnVal int
    IF EXISTS (SELECT MAKH FROM KHACHHANG WHERE MAKH = @MAKH)
    BEGIN
        SET @returnVal = 1
    END
    ELSE SET @returnVal = 0
    RETURN @returnVal
END

```

- Stored procedure sp_InsertKH: thêm khách hàng vào trong cơ sở dữ liệu với các tham số @MAKH ,@TENKH ,@NGAYSINH, ...

```

CREATE PROC [dbo].[sp_InsertKH] (@MAKH CHAR(10), @TENKH NVARCHAR(50), @NGAYSINH DATE, @GIOITINH NVARCHAR(4), @TIEUSU NVARCHAR(250), @MAGH CHAR(10))
AS
BEGIN
    INSERT INTO dbo.KHACHHANG
    (
        MAKH,
        TENKH,
        NGAYSINH,
        GIOITINH,
        TIEUSU,
        MAGH
    )
    VALUES
    (
        @MAKH,
        @TENKH,
        @NGAYSINH,
        @GIOITINH,
        @TIEUSU,
        @MAGH
    )

```

```
)  
END
```

- Stored procedure sp_InsertPhieuTiem: được sử dụng để thêm phiếu tiêm vào cơ sở dữ liệu với các tham số @MAPHIEUTIEM,@NGAYTIEM,...

```
CREATE PROC [dbo].[sp_InsertPhieuTiem] (@MAPHIEUTIEM CHAR(10), @NGAYTIEM DATE,  
@MAKH CHAR(10), @MABS CHAR(10))  
AS  
BEGIN  
    INSERT INTO dbo.PHIEUTIEM  
    (  
        MAPHIEUTIEM,  
        NGAYTIEM,  
        MAKH,  
        MABS  
    )  
    VALUES  
    (  
        @MAPHIEUTIEM,  
        @NGAYTIEM,  
        @MAKH,  
        @MABS  
    )  
END
```

- Stored procedure sp_InsertCTT: được sử dụng để thêm thông tin chi tiết tiêm

```
CREATE PROC [dbo].[sp_InsertCTT] (@MAPHIEUTIEM CHAR(10),@MAVACCINE CHAR(10),  
@GIABAN INT, @MUITHU INT, @NGAYNHACLAIR DATE, @LIEULUONG FLOAT)  
AS  
BEGIN  
    INSERT INTO dbo.CHITIETIEM  
    (  
        MAPHIEUTIEM,  
        MAVACCINE,  
        GIABAN,  
        MUITHU,  
        NGAYTIEMNHACLAIR,  
        LIEULUONG  
    )  
    VALUES  
    (  
        @MAPHIEUTIEM,  
        @MAVACCINE,  
        @GIABAN,  
        @MUITHU,  
        @NGAYNHACLAIR,  
        @LIEULUONG  
    )  
END
```

- Stored procedure sp_GetVCFromPhieuTiem: được sử dụng để SELECT các thông tin của vaccine được sử dụng trong phiếu tiêm với tham số truyền vào là @MAPHIEUTIEM

```
CREATE PROC [dbo].[sp_GetVCFromPhieuTiem] (@MAPHIEUTIEM CHAR(10))
AS
BEGIN
    SELECT vc.MAVACCINE, vc.TENVACCINE, lvc.LOAIVACCINE, vc.NHASX, ctt.MUITHU,
    ctt.LIEULUONG, ctt.NGAYTIEMNHACLAIR, vc.DONGIA
    FROM PHIEUTIEM pt INNER JOIN CHITIETTIEM ctt INNER JOIN VACCINE vc INNER JOIN
    LOAIVC lvc
    ON lvc.MALOAIVC = vc.MALOAIVC
    ON vc.MAVACCINE = ctt.MAVACCINE
    ON ctt.MAPHIEUTIEM = pt.MAPHIEUTIEM
    WHERE ctt.MAPHIEUTIEM = @MAPHIEUTIEM
END
```

- Stored procedure sp_GetPhieuTiemInfo: được sử dụng để liệt kê tất cả các phiếu tiêm theo thứ tự từ mới nhất đến cũ

```
CREATE PROC [dbo].[sp_GetPhieuTiemInfo]
AS
BEGIN
    SELECT pt.MAPHIEUTIEM, kh.MAKH, kh.TENKH, kh.NGAYSINH, kh.GIOITINH, kh.TIEUSU,
    pt.NGAYTIEM, bs.HOTEN AS TENBS
    FROM BACSY bs INNER JOIN PHIEUTIEM pt INNER JOIN KHACHHANG kh
    ON kh.MAKH = pt.MAKH
    ON pt.MABS = bs.MABS
    ORDER BY pt.MAPHIEUTIEM DESC
END
```

- Trigger trg_TaoChiTietTiem: được sử dụng để tự động giảm số lượng vaccine có sẵn sau khi tạo CHITIETTIEM. Ngoài ra trigger này để kiểm tra xem khách hàng đã tiêm vaccine đó trước chưa, nếu tiêm rồi thì đánh dấu đã tiêm nhắc lại cho CHITIETTIEM đó.

```
CREATE TRIGGER [dbo].[trg_TaoChiTietTiem] ON [dbo].[CHITIETTIEM] AFTER INSERT AS
BEGIN
    DECLARE @DaSuDung FLOAT
    SET @DaSuDung = (
        SELECT Inserted.LIEULUONG
        FROM Inserted
    )

    SET @DaSuDung = ROUND(@DaSuDung, 0)

    UPDATE dbo.VACCINE
    SET SOLUONGCOSAN = SOLUONGCOSAN - @DaSuDung
```



```

FROM Inserted INNER JOIN dbo.VACCINE vc ON vc.MAVACCINE = Inserted.MAVACCINE
WHERE vc.MAVACCINE = Inserted.MAVACCINE

UPDATE dbo.CHITIETTIEM
SET CHITIETTIEM.TIEMNHACLAIR=0
FROM inserted, CHITIETTIEM, PHIEUTIEU, (SELECT TOP(1) MAPHIEUTIEU, MAKH FROM
PHIEUTIEU ORDER BY MAPHIEUTIEU DESC) AS PT
where inserted.MAVACCINE=CHITIETTIEM.MAVACCINE
AND PT.MAKH=PHIEUTIEU.MAKH
AND PHIEUTIEU.MAPHIEUTIEU=CHITIETTIEM.MAPHIEUTIEU
AND CHITIETTIEM.MAPHIEUTIEU != PT.MAPHIEUTIEU
END

```

4.2.6. Giao diện Thanh toán

a. Chức năng Thanh toán

- Chức năng này được sử dụng cho thu ngân, quản trị cũng có thể sử dụng nhưng sẽ bị hạn chế chức năng. Chức năng nhằm kiểm tra phiếu tiêm đã thanh toán chưa, nếu chưa thanh toán thì có thể tiến hành thanh toán, sau khi thanh toán xong thì sẽ tự động xuất hóa đơn. Chúng ta có thể sửa xóa hóa đơn cũng như xuất hóa đơn sau.

- Khi mở form, hệ thống sẽ load danh sách tất cả người giám hộ, và tiến hành hiển thị trên datagridview. Form được chia thành hai khu vực, khu vực đầu tiên là thông tin người thanh toán, khu vực thứ hai là thông tin phiếu tiêm cần thanh toán cũng như các thông tin về chi tiết thanh toán. Tương tự các giao diện khác, khi click vào hàng của datagridview thì thông tin hàng đó sẽ được đổ lên các textbox ở khu vực người thanh toán. Ở khu vực thanh toán, nếu là người mới thì nhập đầy đủ thông tin và không nhập mã giám hộ, còn nếu mã người giám hộ tồn tại thì hệ thống sẽ lấy thông tin có trong cơ sở dữ liệu.

- Ở phần thông tin hóa đơn, người dùng sẽ nhập thông tin phiếu tiêm để kiểm tra xem phiếu tiêm đã được thanh toán chưa, nếu chưa thanh toán thì hệ thống sẽ hiển thị danh sách vaccine đã sử dụng ở phiếu tiêm đó và hiển thị tổng tiền ở textbox của chi tiết thanh toán. Ta có thể nhập chiết khấu, hệ thống sẽ tính toán tiền khách phải trả. Khi nhập số tiền khách trả, hệ thống sẽ đưa ra tiền trả lại cho khách. Sau khi nhập đủ thông tin, ta ấn vào button thanh toán. Khi thanh toán thành công thì hệ thống sẽ hiển thị giao diện xuất hóa đơn.

Để sửa thông tin hóa đơn như tên người giám hộ, số điện thoại,... thì ta ấn vào Danh sách hóa đơn, ta cũng click đúp để chỉnh sửa thông tin hoặc click vào button xuất để xuất hóa đơn tương tự như phần Xem danh sách của mục tạo phiếu tiêm.

b. Thiết kế giao diện

Thanh toán

Xin chào Quản trị Đăng xuất

Thông tin người thanh toán

Mã giám hộ: GH004

Họ và tên: Nguyễn Quang Hùng

Địa chỉ: Vĩnh Phúc

Số điện thoại: 092142356

Email: ngHung123@gmail.com

Nhập để tìm kiếm... Tìm kiếm

MAGH	HOTEN	DIACHI	SDT	EMAIL
GH005	Hoàng Th...	Nghệ An	08738472	
→ GH004	Nguyễn Q...	Vĩnh Phúc	092142356	ngHung12...
GH003	Nguyễn T...	Lạng Sơn	098272345	
GH002	Hồ Nguyễn...	Tp HCM	0934346906	
GH001	Nguyễn M...	Hà Nội	09978970	

Thông tin hoá đơn

Mã phiếu tiêm: PT010 Kiểm tra

Ngày thanh toán: 24/05/2022

Tên khách hàng: Nguyễn Quang Hùng

Danh sách vaccine sử dụng

MAV...	TEN...	LOAI...	N...	M...	LIEU...	NGAYTI...	DO...
→ VC0...	vaxig...	Cúm ...	Tr...	3	2	8/24/2022	110...

Chi tiết thanh toán

Tổng tiền: 220000 VND

Chiết khấu: 0

Phải trả: 220000 VND

Khách trả: 880000 VND

Trả lại: 660000 VND

Thao tác

Thanh toán

Danh sách hoá đơn

Đặt lại

Giao diện thanh toán hóa đơn

Cập nhật danh sách hoá đơn

Danh sách hoá đơn

MAHOADON	TONGTIEN	CHIETKHAU	NGAYTHU	NGUOIGH	DIACHI	SDT	THUNGAN
→ HD007	200000	0.2	4/30/2022	Hoàng Thành H...	Nghệ An	08738472	Mai Thu Ngân
HD006	100001	0.2	6/23/2022	Hoàng Thành H...	Nghệ An	08738472	Mai Thu Ngân
HD005	160000	0.3	5/3/2022	Nguyễn Quang...	Vĩnh Phúc	092142356	Thu Ngân Nguyễn
HD004	110000	0	5/7/2022	Nguyễn Tùng L...	Lạng Sơn	098272345	Thu Ngân Nguyễn
HD003	100001	0.2	7/5/2022	Hồ Nguyễn Ngu...	Tp HCM	0934346906	Thu Ngân Nguyễn
HD001	310001	0.1	7/5/2022	Nguyễn Mạnh	Hà Nội	09978970	Hồ Thu Ngân

Hướng dẫn

- Nhấn đúp vào một ô để sửa thông tin (Chỉ cho phép sửa Ngày thu, Tên NGH, Địa chỉ NGH, SDT NGH)
- Nhấn phím Delete để xoá 1 bản ghi

Danh sách vaccine sử dụng

MAVACCINE	TENVACCINE	THUYD	MY	1	1	7/8/2022	200000
→ VC008	Varivax	Thủy đậu	Mỹ				

Thoát ? Đặt lại Đã lưu Xuất

Form cập nhật danh sách hóa đơn



TRUNG TÂM TIÊM CHỦNG MTA

HÓA ĐƠN THANH TOÁN

Ngày lập: 7/5/2022 1



HD001

Người thanh toán: Nguyễn Mạnh

Điện Thoại: 099 78970

Địa Chỉ: Hà Nội

Khách Hàng: Nguyễn Mạnh Cường

Bác sĩ: Trần Văn Tùng

Mã Vaccine	Tên VACCINE	Thành tiền (VNĐ)
VC001	Astra	100001
VC003	rubella	70000
Tổng Tiền (VNĐ):		310001
Chiết Khấu (%):		0.1
Thành Tiền (VNĐ):		279001

NHÂN VIÊN THU NGÂN

Hồ Thu Ngân

c. Các lệnh SQL phục vụ cho chức năng

- Stored procedure `sp_GetVCFromPhieuTiem`: được sử dụng để SELECT các thông tin của vaccine được sử dụng trong phiếu tiêm với tham số truyền vào là `@MAPHIEUTIEM`

```
CREATE PROC [dbo].[sp_GetVCFromPhieuTiem] (@MAPHIEUTIEM CHAR(10))
AS
BEGIN
    SELECT vc.MAVACCINE, vc.TENVACCINE, lvc.LOAIVACCINE, vc.NHASX, ctt.MUITHU,
    ctt.LIEULUONG, ctt.NGAYTIEMNHACLAIR, vc.DONGIA
    FROM PHIEUTIEM pt INNER JOIN CHITIETIEM ctt INNER JOIN VACCINE vc INNER JOIN
    LOAIVC lvc
        ON lvc.MALOAIVC = vc.MALOAIVC
        ON vc.MAVACCINE = ctt.MAVACCINE
        ON ctt.MAPHIEUTIEM = pt.MAPHIEUTIEM
    WHERE ctt.MAPHIEUTIEM = @MAPHIEUTIEM
END
```

- Stored procedure `sp_GetAllNGH`: được sử dụng để SELECT tất cả những người giám hộ trong cơ sở dữ liệu

```
CREATE PROC [dbo].[sp_GetAllNGH] (@MAPHIEUTIEM CHAR(10))
AS
BEGIN
    SELECT * FROM NGUOIGIAMHO
END
```

- Stored procedure `sp_GetTenKHFromPhieuTiem`: được sử dụng để lấy tên khách hàng với tham số truyền vào là `@MAPHIEUTIEM`

```
CREATE PROC [dbo].[sp_GetTenKHFromPhieuTiem] (@MAPHIEUTIEM CHAR(10))
AS
BEGIN
    SELECT kh.TENKH FROM KHACHHANG kh INNER JOIN PHIEUTIEM pt
        ON pt.MAKH = kh.MAKH
        WHERE pt.MAPHIEUTIEM = @MAPHIEUTIEM
END
```

- Stored procedure `sp_InsertNGH`: được sử dụng thêm thông tin người giám hộ vào cơ sở dữ liệu với các tham số `@MAGH`, `@HOTEN`,...

```
CREATE PROC [dbo].[sp_InsertNGH] (@MAGH CHAR(10), @HOTEN NVARCHAR(50), @DIACHI
NVARCHAR(100), @SDT VARCHAR(20))
AS
BEGIN
    INSERT INTO dbo.NGUOIGIAMHO
    (
        MAGH,
```

```

        HOTEN,
        DIACHI,
        SDT
    )
VALUES
(
    @MAGH, -- MAGH - char(10)
    @HOTEN, -- HOTEN - nvarchar(50)
    @DIACHI, -- DIACHI - nvarchar(100)
    @SDT -- SDT - varchar(20)
)
END

```

- Stored procedure sp_AddMaGHtoKH: được sử dụng thêm, cập nhật mã giám hộ cho các khách hàng

```

CREATE PROC [dbo].[sp_AddMaGHtoKH] (@MAGH CHAR(10))
AS
BEGIN
    UPDATE KHACHHANG SET MAGH = @MAGH WHERE MAKH IN (
        SELECT kh.MAKH
        FROM NGUOIGIAMHO gh INNER JOIN HOADON hd INNER JOIN PHIEUTIEP pt INNER
JOIN KHACHHANG kh
        ON kh.MAKH = pt.MAKH ON pt.MAPHIEUTIEP = hd.MAPHIEUTIEP ON hd.MAGH =
gh.MAGH
        WHERE gh.MAGH = @MAGH )
END

```

- Stored procedure sp_InsertHD: được sử dụng để thêm hóa đơn vào cơ sở dữ liệu với các tham số @MAHOADON, @CHIETKHAU,...

```

CREATE PROC [dbo].[sp_InsertHD] (@MAHOADON CHAR(10), @CHIETKHAU FLOAT, @NGAYTHU
DATE, @TONGTIEN BIGINT, @MAGH CHAR(10), @MATHUNGAN CHAR(10), @MAPHIEUTIEP
CHAR(10))
AS
BEGIN
    INSERT INTO dbo.HOADON
    (
        MAHOADON,
        CHIETKHAU,
        NGAYTHU,
        TONGTIEN,
        MAGH,
        MATHUNGAN,
        MAPHIEUTIEP
    )
VALUES
(
    @MAHOADON,
    @CHIETKHAU,
    @NGAYTHU,
    @TONGTIEN,

```

```

        @MAGH,
        @MATHUNGAN,
        @MAPHIEUTIEM
    )
END

```

- Stored procedure sp_CheckPaymentStatus: được sử dụng để kiểm phiếu tiền đã được thanh toán hay chưa, sử dụng tham số @MAPHIEUTIEM

```

CREATE PROC [dbo].[sp_CheckPaymentStatus] (@MAPHIEUTIEM CHAR(10))
AS
BEGIN
    DECLARE @returnVal int
    IF EXISTS (SELECT * FROM dbo.PHIEUTIEM pt INNER JOIN dbo.HOADON hd ON
hd.MAPHIEUTIEM = pt.MAPHIEUTIEM WHERE pt.MAPHIEUTIEM = @MAPHIEUTIEM)
    BEGIN
        SET @returnVal = 1
    END
    ELSE SET @returnVal = 0
    RETURN @returnVal
END

```

- Stored procedure sp_GetAllHoaDonInfo: được sử dụng để lấy danh sách các hóa đơn có trong cơ sở dữ liệu

```

CREATE PROC [dbo].[sp_GetAllHoaDonInfo]
AS
BEGIN
    SELECT hd.MAHOADON, hd.TONGTIEN, hd.CHIETKHAU, hd.NGAYTHU, ngh.HOTEN AS
NGUOIGH, ngh.DIACHI, ngh.SDT, tn.HOTEN AS THUNGAN
    FROM dbo.THUNGAN tn INNER JOIN dbo.HOADON hd INNER JOIN dbo.NGUOIGIAMHO ngh
    ON ngh.MAGH = hd.MAGH
    ON hd.MATHUNGAN = tn.MATHUNGAN
END

```

- Stored procedure sp_GetDeleteHoaDon: được sử dụng để xóa hóa đơn có tham số @MAHOADON trong cơ sở dữ liệu

```

CREATE PROC [dbo].[sp_DeleteHoaDon] (@MAHOADON CHAR(10))
AS
BEGIN
    DELETE FROM dbo.HOADON WHERE MAHOADON = @MAHOADON
END

```

- Stored procedure sp_UpdateHoaDonInfo: được sử cập nhật hóa đơn có mã là @MAHOADON. Proc cần truyền những tham số khác như @NGAYTHU, @NGUOIGH,...

```

CREATE PROC [dbo].[sp_UpdateHoaDonInfo] (@MAHOADON CHAR(10), @NGAYTHU DATE,
@NGUOIGH NVARCHAR(50), @DIACHI NVARCHAR(100), @SDT VARCHAR(20))
AS

```

```

BEGIN
    UPDATE dbo.HOADON SET NGAYTHU = @NGAYTHU WHERE MAHOADON = @MAHOADON
    UPDATE dbo.NGUOIGIAMHO SET HOTEN = @NGUOIGH, DIACHI = @DIACHI, SĐT = @SĐT
WHERE MAGH IN
    (
        SELECT HD.MAGH FROM dbo.NGUOIGIAMHO ngh INNER JOIN dbo.HOADON hd ON
        hd.MAGH = ngh.MAGH WHERE hd.MAHOADON = @MAHOADON
    )
END

```

- Trigger trg_xoaPT: được sử dụng xóa hóa đơn và các chi tiết tiêm của phiếu tiêm trước khi xóa phiếu tiêm

```

CREATE TRIGGER trg_xoaPT ON PHIEUTIEM INSTEAD OF DELETE AS
BEGIN
    DECLARE @deletedMAPT CHAR(10)
    SET @deletedMAPT = (SELECT MAPHIEUTIEM FROM Deleted)

    DELETE FROM dbo.HOADON WHERE MAPHIEUTIEM = @deletedMAPT
    DELETE FROM dbo.CHITIETTIEM WHERE MAPHIEUTIEM = @deletedMAPT

    DELETE FROM dbo.PHIEUTIEM WHERE MAPHIEUTIEM = @deletedMAPT
END

```

4.2.7. Giao diện Thống kê

a. Chức năng Thống kê

- Đối với quản trị, phần mềm thống kê theo thống kê doanh thu, thống kê theo tỷ trọng vaccine từng loại, thống kê theo top vaccine được dùng nhiều nhất và thống kê số lượng vắc xin đã hoặc sắp hết hạn theo từng loại.

- Ở thống kê doanh thu, mặc định phần mềm hiển thị biểu đồ doanh thu trong 2 tháng gần nhất. Ta có thể thay đổi thời gian hiển thị thống kê của biểu đồ qua hai dateEdit hiển thị thời gian bắt đầu và kết thúc thống kê. Ở thống kê này, có một textbox hiển thị tổng doanh thu trong khoảng thời gian đó. Ta có thể xuất báo cáo về doanh thu bằng cách ấn vào button Xuất báo cáo.

- Ở thống kê tỷ trọng các loại vaccine, biểu đồ có dạng hình tròn, trong biểu đồ hiển thị tên của loại đó cũng như tỷ trọng của loại đó.

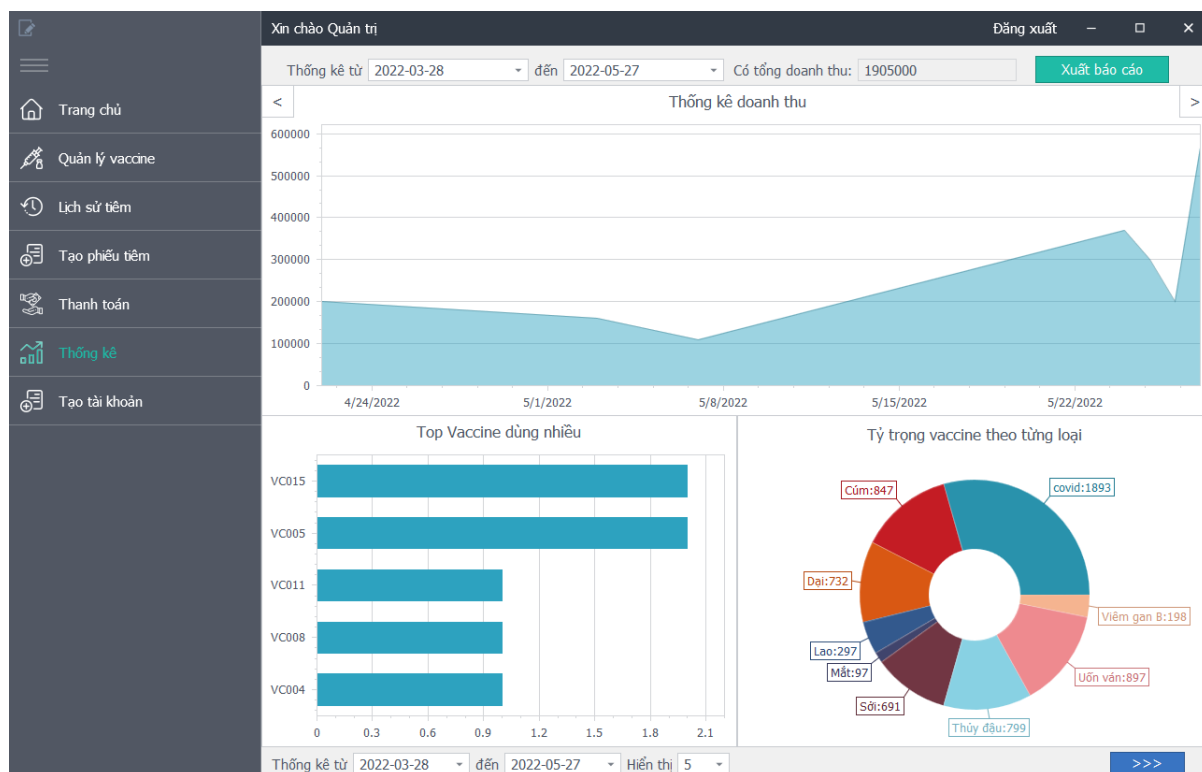
- Ở thống kê top vaccine dùng nhiều, cũng tương tự như biểu đồ thống kê doanh thu, biểu đồ hiển thị mặc định top vaccine dùng nhiều trong vòng 2 tháng gần nhất, chúng ta có thể thay đổi giá trị đó qua hai dateEdit ở dưới biểu đồ. Ngoài ra,

thống kê còn cho phép ta chọn số lượng vaccine sẽ hiện thị qua một combobox ở dưới biểu đồ.

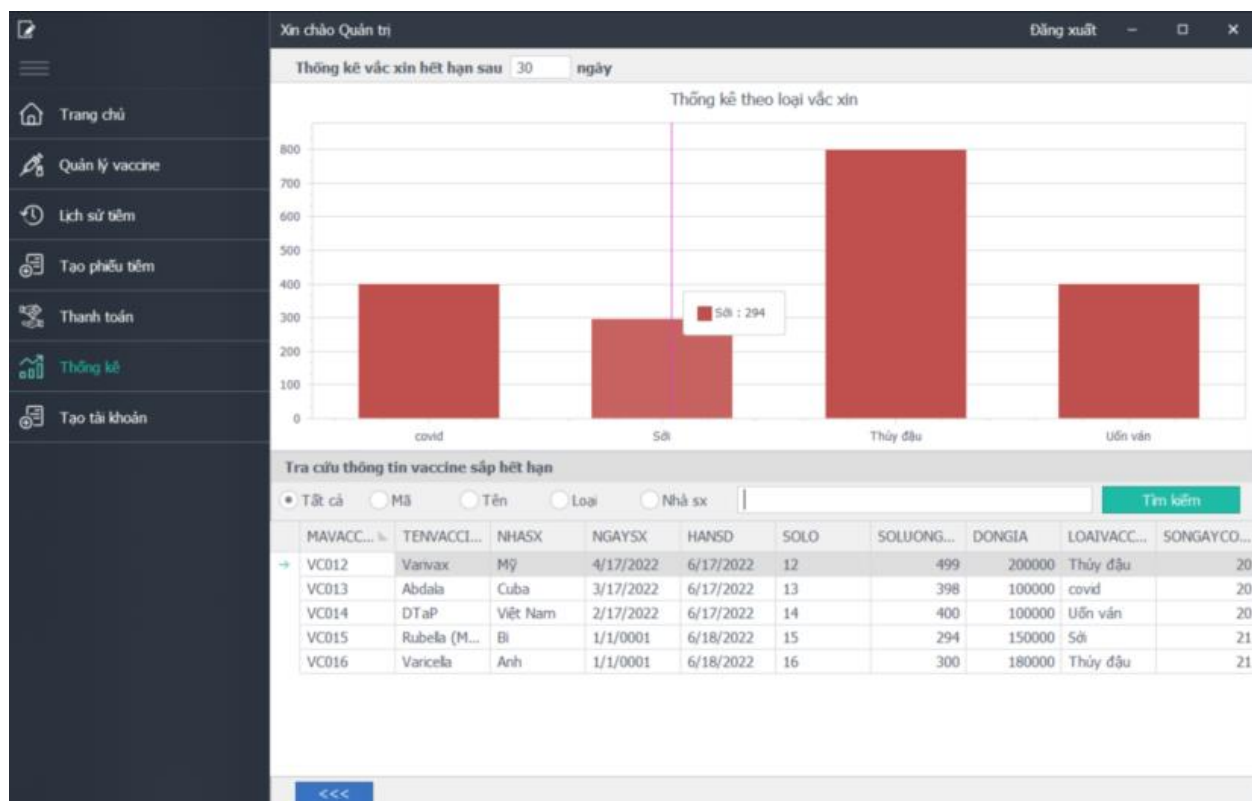
- Ở thống kê vắc xin đã hoặc sắp hết hạn, mặc định phần mềm sẽ hiển thị trong biểu đồ và trong gridview những vắc xin sắp hoặc đã hết hạn sau 30 ngày tiếp theo. Chúng ta có thể thay đổi thời gian để số liệu trong biểu đồ loại vắc xin và ở bảng chi tiết các vắc xin đã hoặc sắp hết hạn sẽ được cập nhật lại. Chúng ta cũng có thể tìm kiếm vắc xin đã hoặc sắp hết hạn trong gridview thông qua mã vắc xin, tên vắc xin, loại vắc xin, nhà sản xuất hay tất cả vắc xin. Hệ thống sẽ tính số ngày còn lại của vắc xin còn sử dụng được trong cột SONGAYCONLAI, còn lại các thông tin hiển thị khác trong gridview sẽ tương tự như khi tra cứu thông tin vắc xin trong chức năng quản lý vắc xin.

- Đối với bác sĩ, phần mềm thống kê các khách hàng sắp đến thời gian tiêm nhắc lại. Từ đó có thể xuất danh sách hoặc gửi email cho khách hàng để thông báo.

b. Thiết kế giao diện



Giao diện thống kê 1 cho quản trị viên



Giao diện thống kê 2 cho quản trị viên



TRUNG TÂM TIÊM CHỦNG MTA

BÁO CÁO THỐNG KÊ

Thống Kê Doanh Thu

Thời gian từ: 19/03/2022 đến 18/05/2022

Ngày	Doanh Thu (VNĐ)
4/30/2022	200000
5/3/2022 1	160000
5/7/2022 1	110000
Tổng doanh thu:	470000

This application was created using the trial version of the XtraReports.

Mẫu hóa báo cáo thống kê doanh thu

Xin chào Trần Văn Tùng
Đăng xuất

Thông kê khách hàng đến thời gian tiêm

	<input type="checkbox"/>	MAKH	TENKH	GIOITINH	SDT	EMAIL	MAVACCINE	NGAYTIEMNHA...
→	<input type="checkbox"/>	KH014	Phạm Thị Xuân	Nữ	098182722	thixuan1406@g...	VC015	9/15/2022
	<input type="checkbox"/>	KH004	Nguyễn Tùng L...	Nam	099128312	tunglam1203@...	VC005	8/7/2022
	<input type="checkbox"/>	KH001	Đào Đình Huy	Nam	091248748	dinghuy@outlo...	VC007	8/5/2022
	<input type="checkbox"/>	KH018	Phùng Đức Dũng	Nam			VC015	7/13/2022
	<input type="checkbox"/>	KH010	Phí Minh Phương	Nam	093187483	phuongphi74@...	VC011	6/17/2022

Hạn tiêm từ 2022-05-16 đến 2022-09-15

Thông kê khách hàng đến thời gian tiêm cho bác sĩ

←
📁
!
🗑️
✉️
🕒
🔍
📄
🗑️
⋮

Trung tâm tiêm chủng
Hộp thư đến x

trungtamtiemchung155@gmail.com
tới tôi ▾

Xin chào Trần Bảo Trung

Email này được gửi từ trung tâm tiêm chủng MTAVC

Lịch tiêm nhắc lại vắc xin Astra vào ngày 6/9/2022

Mong bạn thu xếp thời gian để tới trung tâm. Xin cảm ơn.

↩️ Trả lời

➡️ Chuyển tiếp

Nội dung email được gửi tới khách hàng

**Thống Kê Khách Hàng Đến Hạn Tiêm**

Thời gian từ: 2022-01-03 đến 2022-10-13

Mã KH	Tên Khách hàng	SĐT	Mã VC	Ngày đến hạn
KH003	Hồ Nguyễn Nguyên	095533211	VC001	10/5/2022
KH016	Dương Trung Hiếu		VC013	9/22/2022
KH014	Phạm Thị Xuân	098182722	VC015	9/15/2022
KH012	Vũ Ngọc Khánh	092172834	VC013	8/26/2022
KH002	Nguyễn Mạnh Cường	097104070	VC001	8/26/2022
KH004	Nguyễn Tùng Lâm	099128312	VC005	8/7/2022
KH001	Đào Đình Huy	091248748	VC007	8/5/2022
KH002	Nguyễn Mạnh Cường	097104070	VC002	7/26/2022
KH018	Phùng Đức Dũng		VC015	7/26/2022
KH010	Phí Minh Phương	093187483	VC011	6/17/2022
KH005	Nguyễn Quang Hùng	084626586	VC004	4/11/2022

Xuất danh sách khách hàng đến thời gian tiêm cho bác sĩ

c. Các lệnh SQL phục vụ cho chức năng

- Stored procedure sp_GetHoaDonINTIME_TK: được sử dụng để liệt kê ngày thanh toán và tổng tiền của các hóa đơn được thanh toán trong khoảng thời gian từ ngày @NgàyDau đến ngày @NgàyCuoi

```

CREATE PROC [dbo].[sp_GetHoaDonINTIME_TK] (@NgayDau DATE, @NgayCuoi DATE)
AS
BEGIN
    SELECT NGAYTHU AS "Ngày", TONGTIEN AS "Tien"
    FROM dbo.HOADON
    WHERE NGAYTHU BETWEEN @NgayDau AND @NgayCuoi
    ORDER BY NGAYTHU ASC
END

```

- Stored procedure sp_CountVCTheoLoaiVC: được sử dụng để liệt kê tổng số vaccine của mỗi loại vaccine

```

CREATE PROC [dbo].[sp_CountVCTheoLoaiVC]
AS
BEGIN
    SELECT LOAIVC.LOAIVACCINE, SOLUONG
    FROM LOAIVC, (SELECT VC.MALOAIVC AS 'MALOAI', SUM(SOLUONGCOSAN) AS
'SoLuong'
FROM VACCINE vc INNER JOIN LOAIVC lvc ON lvc.MALOAIVC = vc.MALOAIVC
GROUP BY VC.MALOAIVC) AS MALVC_SL
WHERE LOAIVC.MALOAIVC=MALVC_SL.MALOAI
ORDER BY LOAIVC.LOAIVACCINE ASC
END

```

- Stored procedure sp_GetMostUsedVaccineINTIME: được sử dụng để liệt kê số vaccine được sử dụng trong khoảng thời gian từ @NgayDau đến @NgayCuoi và được sắp xếp theo thứ tự giảm dần số lượng.

```

CREATE PROC [dbo].[sp_GetMostUsedVaccineINTIME] (@NgayDau DATE, @NgayCuoi DATE)
AS
BEGIN
    SELECT MAVACCINE AS 'MaVC', COUNT(MAVACCINE) AS 'SoLuong'
    FROM dbo.PHIEUTIEM INNER JOIN dbo.CHITIETIEM ON
CHITIETIEM.MAPHIEUTIEM = PHIEUTIEM.MAPHIEUTIEM
WHERE CHITIETIEM.MAPHIEUTIEM IN
(SELECT DISTINCT MAPHIEUTIEM FROM dbo.HOADON WHERE NGAYTHU BETWEEN
@NgayDau AND @NgayCuoi)
GROUP BY MAVACCINE
ORDER BY SoLuong DESC
END

```

- Stored procedure sp_GetDoanhThuTheoNgayINTIME: được sử dụng ngày và tổng doanh thu của ngày đó trong khoảng thời gian @NgayDau đến @NgayCuoi

```

CREATE PROC [dbo].[sp_GetDoanhThuTheoNgayINTIME] (@NgayDau DATE, @NgayCuoi DATE)
AS
BEGIN
    SELECT NGAYTHU AS "Ngày", SUM(TONGTIEN) AS "Tien"
    FROM dbo.HOADON

```

```

WHERE NGAYTHU BETWEEN @NgayDau AND @NgayCuoi
GROUP BY(NGAYTHU)
ORDER BY NGAYTHU ASC
END

```

- Stored procedure sp_GetTenLoaiVCTheoMaLoai: được sử dụng để SELECT tên của loại vaccine có mã loại vaccine là @MaLoaiVC

```

CREATE PROC [dbo].[sp_GetTenLoaiVCTheoMaLoai](@MaLoaiVC CHAR(10))
AS
BEGIN
    SELECT LOAIVACCINE AS 'TenLoaiVC'
    FROM dbo.LOAIVC
    WHERE MALOAIVC = @MaLoaiVC
END

```

- Stored procedure sp_GetTenVCTuMaVC: được sử dụng để lấy tên của vaccine từ mã vaccine là tham số đầu vào @MaVC

```

CREATE PROC [dbo].[sp_GetTenVCTuMaVC](@MaVC CHAR(10))
AS
BEGIN
    SELECT TENVACCINE AS 'TenVC'
    FROM dbo.VACCINE
    WHERE MAVACCINE = @MaVC
END

```

- Stored procedure sp_GetLoaiVaccineSHH: được sử dụng để lấy các loại vắc xin đã hoặc sắp hết hạn theo số lượng sau số ngày cho trước với tham số đầu vào là @SoNgay

```

ALTER PROC [dbo].[sp_GetLoaiVaccineSHH] (@SoNgay INT)
AS
BEGIN
    SELECT vc.MALOAIVC AS 'MALOAI', lvc.LOAIVACCINE, SUM(SOLUONGCOSAN) AS
    "SOLUONGCOSAN"
    FROM VACCINE vc JOIN LOAIVC lvc ON lvc.MALOAIVC = vc.MALOAIVC
    WHERE DATEDIFF(DAY, GETDATE(), HANSD) < @SoNgay
    GROUP BY lvc.LOAIVACCINE, vc.MALOAIVC
END

```

- Stored procedure sp_GetVaccineSHH: được sử dụng để lấy tất cả các vắc xin đã hoặc sắp hết hạn theo số lượng sau số ngày cho trước với tham số đầu vào là @SoNgay

```

CREATE PROC [dbo].[sp_GetVaccineSHH] (@SoNgay INT)
AS
BEGIN
    SELECT vc.MAVACCINE, TENVACCINE, vc.NHASX, vc.NGAYSX, vc.HANSX, vc.SOLO,
    vc.SOLUONGCOSAN, vc.DONGIA, lvc.LOAIVACCINE, DATEDIFF(DAY, GETDATE(), vc.HANSX) AS
    'SONGAYCONLAI'

```

```

FROM VACCINE vc JOIN LOAIVC lvc ON lvc.MALOAIVC = vc.MALOAIVC
WHERE DATEDIFF(DAY, GETDATE(), HANSĐ) < @SoNgày
GROUP BY vc.MAVACCINE, TENVACCINE, vc.NHASX, vc.NGAYSX, vc.HANSĐ, vc.SOLO,
vc.SOLUONGCOSAN, vc.DONGIA, lvc.LOAIVACCINE, vc.HANSĐ
ORDER BY SONGAYCONLAI ASC
END

```

- Stored procedure `sp_SearchAllVaccineSHH`: được sử dụng để tìm kiếm vaccine có thông tin của một trường giống với tham số `@Value` được truyền vào và sẽ hết hạn sau số ngày được truyền vào bởi tham số `@SoNgày`

```

CREATE PROC [dbo].[sp_SearchAllVaccineSHH] (@Value NVARCHAR(100), @SoNgày INT)
AS
BEGIN
    SELECT vc.MAVACCINE, TENVACCINE, vc.NHASX, vc.NGAYSX, vc.HANSĐ, vc.SOLO,
    vc.SOLUONGCOSAN, vc.DONGIA, lvc.LOAIVACCINE, DATEDIFF(DAY, GETDATE(), vc.HANSĐ) AS
    'SONGAYCONLAI'
    FROM VACCINE vc INNER JOIN LOAIVC lvc ON lvc.MALOAIVC = vc.MALOAIVC
    WHERE (vc.MAVACCINE LIKE N'%' + @Value + '%'
    OR vc.TENVACCINE LIKE N'%' + @Value + '%'
    OR vc.NHASX LIKE N'%' + @Value + '%'
    OR vc.NGAYSX LIKE N'%' + @Value + '%'
    OR vc.HANSĐ LIKE N'%' + @Value + '%'
    OR vc.SOLO LIKE N'%' + @Value + '%'
    OR vc.SOLUONGCOSAN LIKE N'%' + @Value + '%'
    OR lvc.LOAIVACCINE LIKE N'%' + @Value + '%'
    OR vc.DONGIA LIKE N'%' + @Value + '%')
    AND DATEDIFF(DAY, GETDATE(), HANSĐ) < @SoNgày
    GROUP BY vc.MAVACCINE, TENVACCINE, vc.NHASX, vc.NGAYSX, vc.HANSĐ, vc.SOLO,
    vc.SOLUONGCOSAN, vc.DONGIA, lvc.LOAIVACCINE, vc.HANSĐ
    ORDER BY SONGAYCONLAI ASC
END

```

- Stored procedure `sp_SearchByLoaiVCSHH`: được sử dụng để tìm kiếm vaccine có thông tin có tên loại vaccine trùng với tham số truyền vào `@Value` và sẽ hết hạn sau số ngày được truyền vào bởi tham số `@SoNgày`

```

CREATE PROC [dbo].[sp_SearchByLoaiVCSHH] (@Value NVARCHAR(100), @SoNgày INT)
AS
BEGIN
    SELECT vc.MAVACCINE, TENVACCINE, vc.NHASX, vc.NGAYSX, vc.HANSĐ, vc.SOLO,
    vc.SOLUONGCOSAN, vc.DONGIA, lvc.LOAIVACCINE, DATEDIFF(DAY, GETDATE(), vc.HANSĐ) AS
    'SONGAYCONLAI'
    FROM VACCINE vc INNER JOIN LOAIVC lvc ON lvc.MALOAIVC = vc.MALOAIVC
    WHERE lvc.LOAIVACCINE LIKE N'%' + @Value + '%' AND DATEDIFF(DAY, GETDATE(),
    HANSĐ) < @SoNgày
    GROUP BY vc.MAVACCINE, TENVACCINE, vc.NHASX, vc.NGAYSX, vc.HANSĐ, vc.SOLO,
    vc.SOLUONGCOSAN, vc.DONGIA, lvc.LOAIVACCINE, vc.HANSĐ
    ORDER BY SONGAYCONLAI ASC
END

```

- Stored procedure `sp_SearchByMaVCSHH`: được sử dụng để tìm kiếm vaccine có thông tin có mã vaccine trùng với tham số truyền vào `@Value` và sẽ hết hạn sau số ngày được truyền vào bởi tham số `@SoNgày`

```

CREATE PROC [dbo].[sp_SearchByMaVCSHH] (@Value char(10), @SoNgay INT)
AS
BEGIN
    SELECT vc.MAVACCINE, TENVACCINE,vc.NHASX, vc.NGAYSX, vc.HANSĐ, vc.SOLO,
    vc.SOLUONGCOSAN, vc.DONGIA,lvc.LOAIVACCINE,DATEDIFF(DAY, GETDATE(), vc.HANSĐ) AS
    'SONGAYCONLAI'
    FROM VACCINE vc INNER JOIN LOAIVC lvc ON lvc.MALOAIVC = vc.MALOAIVC
    WHERE vc.MAVACCINE LIKE N'%' + @Value + '%' AND DATEDIFF(DAY, GETDATE(),
    HANSĐ)<@SoNgay
    GROUP BY vc.MAVACCINE, TENVACCINE,vc.NHASX, vc.NGAYSX, vc.HANSĐ, vc.SOLO,
    vc.SOLUONGCOSAN, vc.DONGIA,lvc.LOAIVACCINE,vc.HANSĐ
    ORDER BY SONGAYCONLAI ASC
END

```

- Stored procedure sp_SearchByNhaSXSHH: được sử dụng để tìm kiếm vaccine có thông tin có tên nhà sản xuất trùng với tham số truyền vào @Value và sẽ hết hạn sau số ngày được truyền vào bởi tham số @SoNgay

```

CREATE PROC [dbo].[sp_SearchByNhaSXSHH] (@Value NVARCHAR(100), @SoNgay INT)
AS
BEGIN
    SELECT vc.MAVACCINE, TENVACCINE,vc.NHASX, vc.NGAYSX, vc.HANSĐ, vc.SOLO,
    vc.SOLUONGCOSAN, vc.DONGIA,lvc.LOAIVACCINE,DATEDIFF(DAY, GETDATE(), vc.HANSĐ) AS
    'SONGAYCONLAI'
    FROM VACCINE vc INNER JOIN LOAIVC lvc ON lvc.MALOAIVC = vc.MALOAIVC
    WHERE vc.NHASX LIKE N'%' + @Value + '%' AND DATEDIFF(DAY, GETDATE(),
    HANSĐ)<@SoNgay
    GROUP BY vc.MAVACCINE, TENVACCINE,vc.NHASX, vc.NGAYSX, vc.HANSĐ, vc.SOLO,
    vc.SOLUONGCOSAN, vc.DONGIA,lvc.LOAIVACCINE,vc.HANSĐ
    ORDER BY SONGAYCONLAI ASC
END

```

- Stored procedure sp_SearchByTenSHH: được sử dụng để tìm kiếm vaccine có thông tin có tên vaccine trùng với tham số truyền vào @Value và sẽ hết hạn sau số ngày được truyền vào bởi tham số @SoNgay

```

CREATE PROC [dbo].[sp_SearchByTenSHH] (@Value NVARCHAR(100), @SoNgay INT)
AS
BEGIN
    SELECT vc.MAVACCINE, TENVACCINE,vc.NHASX, vc.NGAYSX, vc.HANSĐ, vc.SOLO,
    vc.SOLUONGCOSAN, vc.DONGIA,lvc.LOAIVACCINE,DATEDIFF(DAY, GETDATE(), vc.HANSĐ) AS
    'SONGAYCONLAI'
    FROM VACCINE vc INNER JOIN LOAIVC lvc ON lvc.MALOAIVC = vc.MALOAIVC
    WHERE vc.TENVACCINE LIKE N'%' + @Value + '%' AND DATEDIFF(DAY, GETDATE(),
    HANSĐ)<@SoNgay
    GROUP BY vc.MAVACCINE, TENVACCINE,vc.NHASX, vc.NGAYSX, vc.HANSĐ, vc.SOLO,
    vc.SOLUONGCOSAN, vc.DONGIA,lvc.LOAIVACCINE,vc.HANSĐ
    ORDER BY SONGAYCONLAI ASC
END

```


- Stored procedure sp_GetKhachHangDenHanINTIME: được sử dụng để lấy thông tin khách hàng đến hạn tiêm trong với hai tham số hạn tiêm từ ngày @NgayDau đến hạn tiêm là @NgayCuoi

```
CREATE PROC [dbo].[sp_GetKhachHangDenHanINTIME] (@NgayDau DATE, @NgayCuoi DATE)
AS
BEGIN
    SELECT KH.MAKH,
    TENKH, KH.GIOITINH, KH.SDT, KH.EMAIL, CTT.MAVACCINE, CTT.NGAYTIEMNHACLAI FROM KHACHHANG
as KH
    INNER JOIN PHIEUTIEM AS PT ON KH.MAKH=pt.MAKH
    INNER JOIN CHITIETIEM AS CTT ON CTT.MAPHIEUTIEM=PT.MAPHIEUTIEM
    WHERE NGAYTIEMNHACLAI BETWEEN @NgayDau AND @NgayCuoi AND TIEMNHACLAI=1
    GROUP BY
    KH.MAKH, TENKH, KH.GIOITINH, KH.SDT, KH.EMAIL, CTT.MAVACCINE, CTT.NGAYTIEMNHACLAI
    ORDER BY DATEDIFF(DAY, GETDATE(), NGAYTIEMNHACLAI) DESC
END
```