**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.

*1.2.1 Yêu cầu*

Chương trình thực hiện những yêu cầu sau:

- Cho phép nhân viên đăng nhập và quản lí 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 lí 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, 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 lí 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:

Tổng tiền =

**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 lí 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

- Hiển 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 lí

*1.3.4. Chức năng xem lại danh sách phiếu tiêm và hóa đơn đã được xử lí ( 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 lí.

**1.4. Phân chia công việc nhóm**

|  |  |
| --- | --- |
| Phụ trách | Công việc |
| Trần Bảo Trung | Xử lí phần nhân viên 1, chức vụ, hoàn thiện báo cáo chương II kèm truy vấn |
| Đinh Thị Thu Uyên | Xử lí phần nhân viên 2, đăng nhập, hoàn thiện báo cáo chương II kèm truy vấn |
| Phạm Đình Khương Duy | Xử lí phần khách hàng, phiếu tiêm, phiếu nhập  Xây dựng mô hình cho trung tâm, hoàn thiện  báo cáo chương I kèm truy vấn |
| Hồ Nguyễn Nguyên | Xử lí phần kho vaccine, loại vaccine, vaccine kèm hoàn thiện báo cáo chương II kèm truy vấn |

**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.

- 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

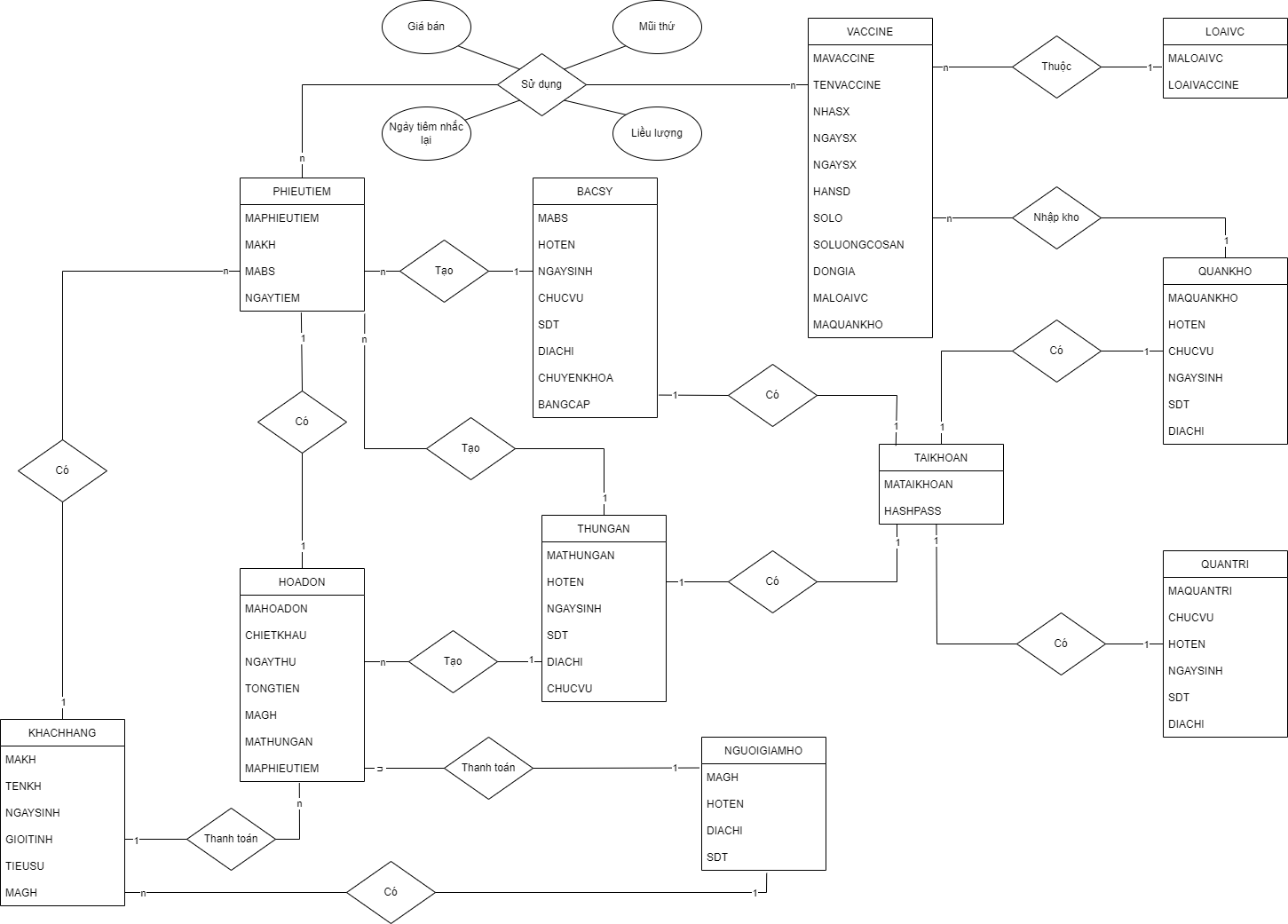
Nếu vacxin thuộc loại đã có, nhập thông tin vaccine với các thông tin: Tên vaccine, mã vaccine (bỏ trống nếu là vaccine mới), nhà sản xuất, số hiệu lô, số lượng, ngày sản xuất, đơn giá, hạn sử dụng, loại vaccine

- Quản trị viên: Có thể xem được tất cả các chức năng quản lí 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.

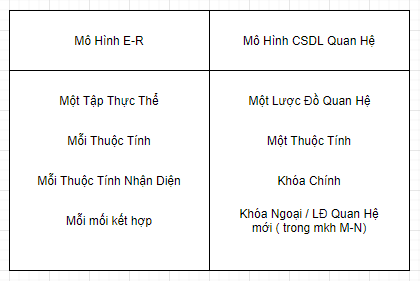
Ngoài ra quản trị viên có thêm chức năng tạo tài khoản cho tất cả các thành viên trong trung tâm tiêm chủng với các thông tin tương ứng với từng đối tượng.

**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**



*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 mối liên kết thành một lược đồ quan hệ có thuộc tính là thuộc tính của mối 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)

PHIEUTIEM(MAPHIEUTIEM, MAKH, MABS, NGAYTIEM)

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

LOAIVC(MALOAIVC,LOAIVACCINE)

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

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

TAIKHOAN(MATAIKHOAN,HASHPASS)

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

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

NGUOIGIAMHO(MAGH,HOTEN,DIACHI,SDT)

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

**Bước 2:** Chuyển các mối liên kết

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

CHITIETTIEM(MAPHIEUTIEM, MAVACCINE, GIABAN, MUITHU, NGAYTIEMNHACLAI, LIEULUONG)

**2.4. Phân tích nhu cầu sử dụng thông tin**

* + KHACHHANG(MAKH, TENKH, NGAYSINH, GIOITINH, TIEUSU, MAGH)

+ 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

* + PHIEUTIEM(MAPHIEUTIEM, 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, HANSD, 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)

+ 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)

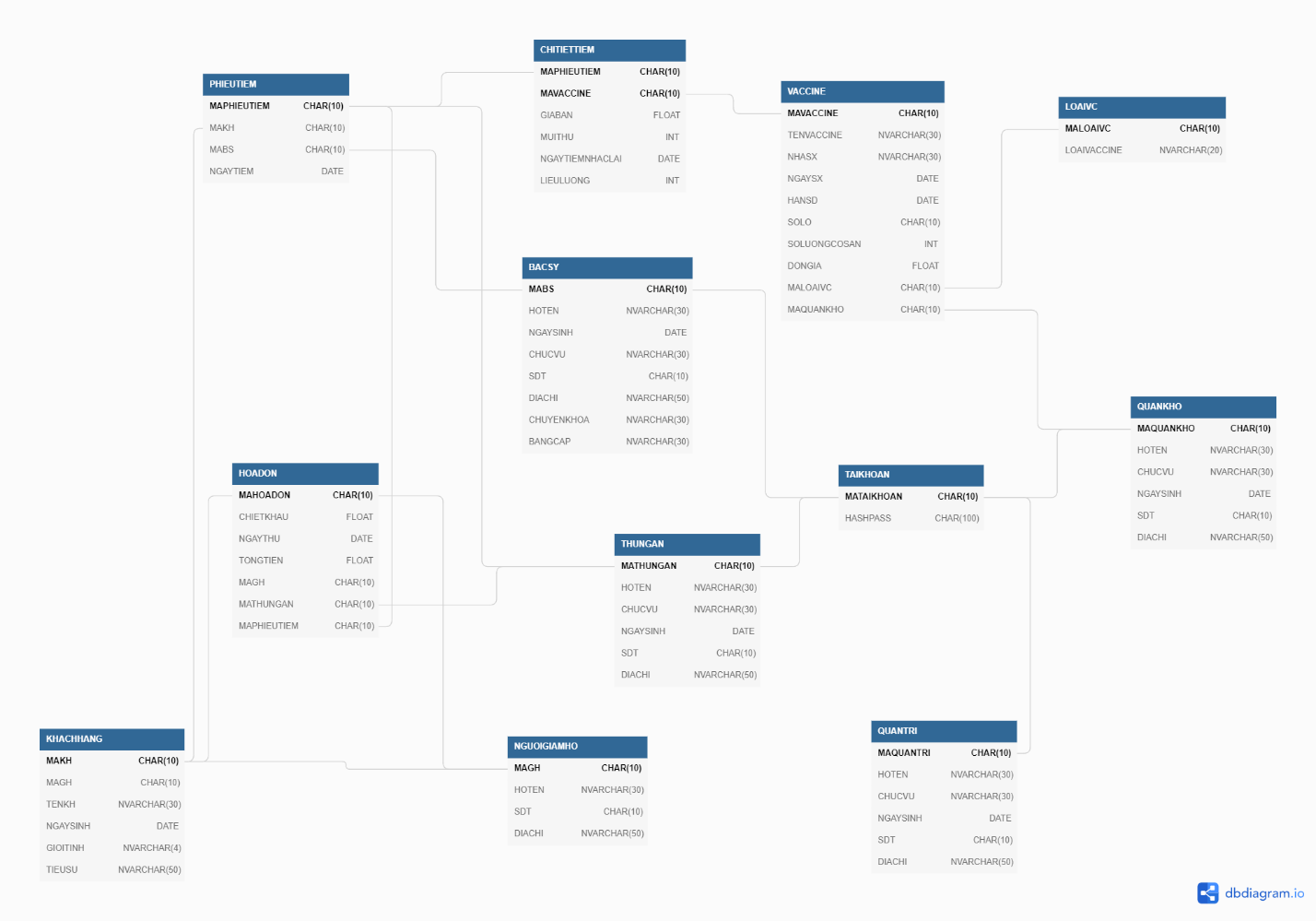
+ Thuộc 1NF vì miền giá trị của các thuộc tính chỉ chứa giá trị đơn, không phân chia được và giá trị của mỗi thuộc tính cũng là một giá trị đơn lấy từ miền giá trị của nó

+ 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: CHITIETTIEM, PHIEUTIEM, VACCINE, QUANKHO, LOAIVC, BACSY, HOADON, THUNGAN, TAIKHOAN, NGUOIGIAMHO, KHACHHANG, QUANTRI

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **CHITIETTIEM** | | | | | |
| *Trường* | *Kiểu* | *Index* | *NULL* | *Default* | *Chú thích* |
| **MAPHIEUTIEM** | **CHAR(10)** | **REFERENCES** | **NO** |  | **Mã PT** |
| **MAVACCINE** | **CHAR(10)** | **REFERENCES** | **NO** |  | **Mã VC** |
| GIABAN | FLOAT |  | YES |  | Giá bán |
| MUITHU | INT |  | YES |  | Mũi thứ |
| NGAYTIEMNHACLAI | DATE |  | YES |  | Ngày tiêm nhắc lại |
| LIEULUONG | INT |  | YES |  | Liều lượng |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **PHIEUTIEM** | | | | | |
| *Trường* | *Kiểu* | *Index* | *NULL* | *Default* | *Chú thích* |
| **MAPHIEUTIEM** | **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 |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **VACCINE** | | | | | |
| *Trường* | *Kiểu* | *Index* | *NULL* | *Def-ault* | *Chú thích* |
| **MAVACCINE** | **CHAR(10)** | **PRIMARY** | **NO** |  | Mã VC |
| TENVACCINE | NVARCHAR(30) |  | YES |  | Tên VC |
| NHASX | NVARCHAR(30) |  | YES |  | Nhà SX |
| NGAYSX | DATE |  | YES |  | Ngày SX |
| HANSD | 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** | | | | | |
| *Trường* | *Kiểu* | *Index* | *NULL* | *Default* | *Chú thích* |
| **MAQUANKHO** | **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** | | | | | |
| *Trường* | *Kiểu* | *Index* | *NULL* | *Default* | *Chú thích* |
| **MALOAIVC** | **CHAR(10)** | **PRIMARY** | **NO** |  | **Mã loại vắc xin** |
| LOAIVACCINE | NVARCHAR(20) |  | YES |  | Loại vắc xin |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **BACSY** | | | | | |
| *Trường* | *Kiểu* | *Index* | *NULL* | *Default* | *Chú thích* |
| **MABS** | **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** | | | | | |
| *Trường* | *Kiểu* | *Index* | *NULL* | *Default* | *Chú thích* |
| **MATAIKHOAN** | **CHAR(10)** | **PRIMARY** | **NO** |  | **Mã tài khoản** |
| HASHPASS | CHAR(100) |  | YES |  | Mã hash mật khẩu |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **THUNGAN** | | | | | |
| *Trường* | *Kiểu* | *Index* | *NULL* | *Default* | *Chú thích* |
| **MATHUNGAN** | **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 |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **KHACHHANG** | | | | | |
| *Trường* | *Kiểu* | *Index* | *NULL* | *Default* | *Chú thích* |
| **MAKH** | **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 |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **NGUOIGIAMHO** | | | | | |
| *Trường* | *Kiểu* | *Index* | *NULL* | *Default* | *Chú thích* |
| **MAGH** | **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 |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **HOADON** | | | | | |
| *Trường* | *Kiểu* | *Index* | *NULL* | *Default* | *Chú thích* |
| **MAHOADON** | **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** |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **QUANTRI** | | | | | |
| *Trường* | *Kiểu* | *Index* | *NULL* | *Default* | *Chú thích* |
| **MAQUANTRI** | **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,

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)

)

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,

NGAYTIEMNHACLAI DATE,

LIEULUONG INT

)

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,

HANSD 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(

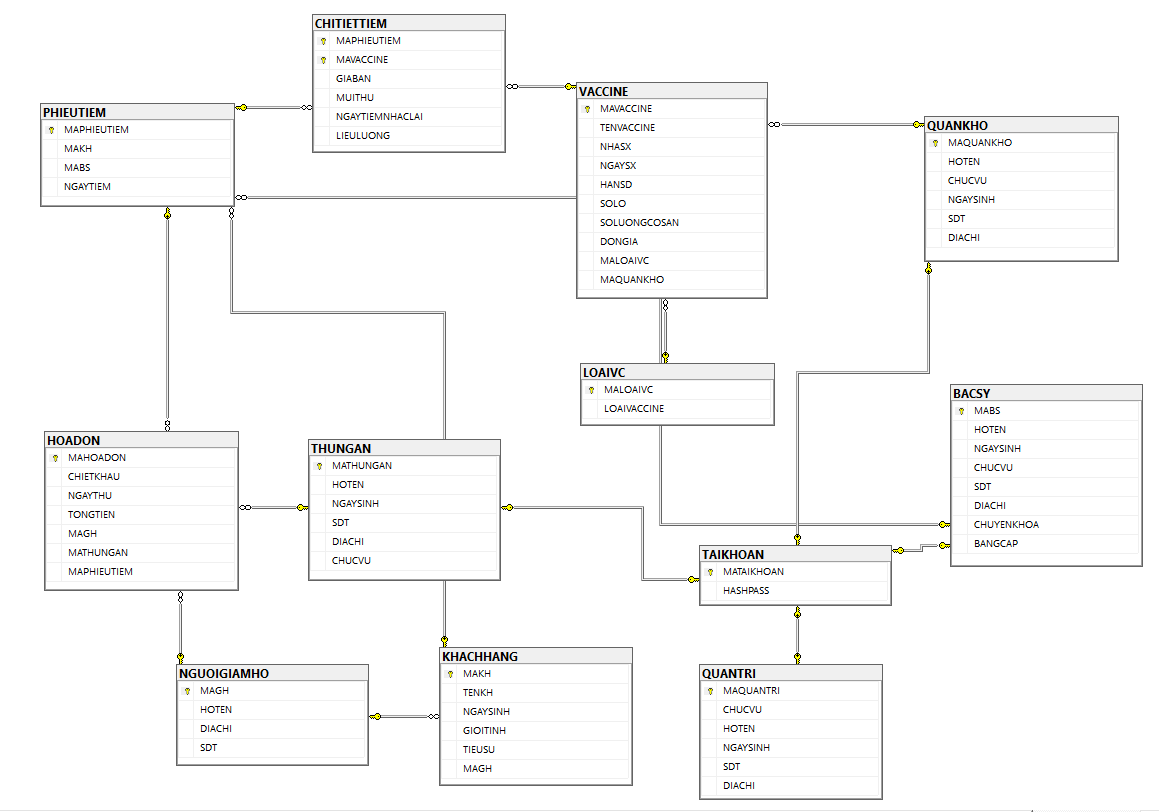
MALOAIVC CHAR(10),

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**

* + 1. **Trần Bảo Trung**
       1. **Truy vấn cơ bản**

1. Lấy tất cả tên vaccine, tên loại vaccine

* Đại số quan hệ

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

* Câu lệnh truy vấn

SELECT TENVACCINE, LOAIVACCINE FROM VACCINE, LOAIVC

WHERE VACCINE.MALOAIVC=LOAIVC.MALOAIVC

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

* Đại số quan hệ

MAVACCINE, TENVACCINE ( SOLUONGCOSAN < 100 (VACCINE))

-Câu lệnh truy vấn

SELECT MAVACCINE, TENVACCINE FROM VACCINE

WHERE SOLUONGCOSAN < 100

1. Cho biết mã và tên tất cả vaccine

* Đại số quan hệ

MAVACCINE, TENVACCINE, TENLOAIVC ( VACCINE.MALOAIVC = LOAIVC.MALOAIVCAND (VACCINE x LOAIVC))

* Câu lệnh truy vấn

SELECT MAVACCINE, TENVACCINE, TENLOAIVC FROM VACCINE, LOAIVC

WHERE VACCINE.MALOAIVC= LOAIVC.MALOAIVC

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

* Đại số quan hệ

MAVACCINE, TENVACCINE ( SOLO = 18 (VACCINE))

* Câu lệnh truy vấn

SELECT MAVACCINE, TENVACCINE FROM VACCINE

WHERE SOLO = 18

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

* Đại số quan hệ

MAVACCINE, TENVACCINE ( SOLO = 25 AND SOLUONGCONLAI < 100 (VACCINE))

* Câu lệnh truy vấn

SELECT MAVACCINE, TENVACCINE FROM VACCINE

WHERE SOLO = 25

AND SOLUONGCONLAI < 100

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

* Đại số quan hệ

TENVACCINE, TENLOAIVC ( VACCINE.MALOAIVC = LOAIVC.MALOAIVCAND AND TENLOAIVC=’SỞI’ (VACCINE x LOAIVC))

* Câu lệnh truy vấn

SELECT TENVACCINE, TENLOAIVC FROM VACCINE, LOAIVC

WHERE VACCINE.MALOAIVC = LOAIVC.MALOAIVC

AND TENLOAIVC = N’SỞI’

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

* Đại số quan hệ

MAVACCINE, VACCINE.TENVACCINE ( VACCINE.MAVACCINE = CHITIETTIEM.MAVACCINE AND MAPHIEUTIEM =’PT001’ (VACCINE x CHITIETTIEM))

* Câu lệnh truy vấn

SELECT VACCINE.MAVACCINE, VACCINE.TENVACCINE FROM VACCINE, CHITIETTIEM

WHERE VACCINE.MAVACCINE = CHITIETTIEM.MAVACCINE

AND MAPHIEUTIEM = 'PT001'

1. Cho biết tên, giá bán của vaccine có trong phiếu tiêm có mã là ‘PT001’

* Đại số quan hệ

VACCINE.TENVACCINE, CHITIETTIEM.GIABAN ( VACCINE.MAVACCINE = CHITIETTIEM.MAVC AND VC.TENVC=’ASTRA’ (VACCINE x CHITIETTIEM))

* Câu lệnh truy vấn

SELECT VACCINE.TENVACCINE, GIABAN FROM VACCINE, CHITIETTIEM

WHERE VACCINE.MAVACCINE = CHITIETTIEM.MAVACCINE

AND MAPHIEUTIEM='PT001'

1. Cho biết tổng số vaccine còn lại của loại vaccine có tên là ‘Covid19’

* Đại số quan hệ

ℑ SUM(SOLUONGCOSAN)( VACCINE.MALOAIVC = LOAIVC.MALOAIVC AND LOAIVACCINE=’COVID19’ (VACCINE x LOAIVC))

* Câu lệnh truy vấn

SELECT SUM(SOLUONGCOSAN) AS SOLUONGCOSAN FROM VACCINE, LOAIVC

WHERE VACCINE.MALOAIVC=LOAIVC.MALOAIVC

AND LOAIVACCINE=N'Covid19'

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

* Đại số quan hệ

QUANKHO.HOTEN( VACCINE.MAQUANKHO = QUANKHO.MAQUANKHO AND SOLO=27 (VACCINE x QUANKHO))

* Câu lệnh truy vấn

SELECT HOTEN FROM VACCINE, QUANKHO

WHERE VACCINE.MAQUANKHO=QUANKHO.MAQUANKHO

AND SOLO=27

1. Cho biết tên quản kho và số điện thoại của quản kho có chức vụ là ‘TRƯỞNG PHÒNG’

* Đại số quan hệ

HOTEN,SDT( CHUCVU=’TRƯỞNG PHÒNG’ QUANKHO)

* Câu lệnh truy vấn

SELECT HOTEN, SDT FROM QUANKHO

WHERE CHUCVU=N'TRƯỞNG PHÒNG'

1. Cho biết tất cả số lô vaccine quản kho ‘QK001’ nhập

-Đại số quan hệ

VACCINE.SOLO( VACCINE.MAQUANKHO = QUANKHO.MAQUANKHO AND QUANKHO.MAQUANKHO=’QK001’ (VACCINE x QUANKHO))

* Câu lệnh truy vấn

SELECT SOLO FROM VACCINE, QUANKHO

WHERE QUANKHO.MAQUANKHO=VACCINE.MAQUANKHO

AND QUANKHO.MAQUANKHO='QK001'

1. Cho biết số vaccine có loại vaccine là ‘Thủy đậu’

* Đại số quan hệ

ℑ COUNT(MAVACCINE)( VACCINE.MALOAIVC = LOAIVC.MALOAIVC AND LOAIVACCINE=’THỦY ĐẬU’ (VACCINE x LOAIVC))

* Câu lệnh truy vấn

SELECT COUNT(MAVACCINE) AS SOVACCINE FROM VACCINE, LOAIVC

WHERE VACCINE.MALOAIVC=LOAIVC.MALOAIVC

AND LOAIVACCINE=N'Thủy đậu'

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

- Đại số quan hệ:

(DIACHI = ‘%HA NOI’)  (QUANKHO)

- Câu lệnh truy vấn

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

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

* Đại số quan hệ

ℑ COUNT(TAIKHOAN.MATAIKHOAN)( TAIKHOAN.MATAIKHOAN= QUANKHO.MAQUANKHO AND (QUANKHO x TAIKHOAN))

* Câu lệnh truy vấn

SELECT COUNT(MAQUANKHO) FROM QUANKHO, TAIKHOAN

WHERE QUANKHO.MAQUANKHO=TAIKHOAN.MATAIKHOAN

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

* Đại số quan hệ

ℑ SUM(DONGIA)( SOLO=100 VACCINE)

* Câu lệnh truy vấn

SELECT SUM(DONGIA) FROM VACCINE

WHERE SOLO=100

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

* Đại số quan hệ:

(NGAYTIEMNHACLAI = ‘2022-10-20’)  (CHITIETTIEM)

**-** Câu lệnh truy vấn:

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

1. Cho biết phiếu tiêm có ngày tiêm nhắc lại là ’22-12-2022’

* Đại số quan hệ

(NGAYTIEMNHACLAI = ‘2022-12-22’)  (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

(‘2021-03-12’ <NGAYSX = ‘2022-02-12’)  (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

VACCINE.TENVACCINE, CHITIETTIEM.LIEULUONG ( VACCINE.MAVACCINE = CHITIETTIEM.MAVC AND MAPHIEUTIEM=’PT002’ (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 Vacxin

SELECT LOAIVACCINE, MAX(DONGIA) AS 'GIACAONHAT' FROM VACCINE, LOAIVC

WHERE VACCINE.MALOAIVC=LOAIVC.MALOAIVC

GROUP BY LOAIVACCINE

1. 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

1. 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

1. Cho biết 5 vaccine có số lượng ít nhất

SELECT TOP(5) MAVACCINE, TENVACCINE, SOLUONGCOSAN FROM VACCINE

ORDER BY SOLUONGCOSAN ASC

1. Xuất các vùng xuất xử vaccine kèm số lượng

SELECT NHASX, SUM(SOLUONGCOSAN) AS SOLUONG FROM VACCINE

GROUP BY NHASX

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

SELECT MAPHIEUTIEM , SUM(GIABAN) FROM CHITIETTIEM

GROUP BY MAPHIEUTIEM

HAVING SUM(GIABAN)>200000

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

SELECT MAX(DONGIA) AS DONGILONNHAT, MIN(DONGIA) AS DONGIANHONHAT

FROM VACCINE

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

SELECT MAVACCINE, TENVACCINE, DATEDIFF(DAY, GETDATE(), HANSD) AS 'SONGAYCONLAI' FROM VACCINE

WHERE DATEDIFF(DAY, GETDATE(), HANSD)<50

1. 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'

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

SELECT MAPHIEUTIEM, DATEDIFF(DAY, GETDATE(), NGAYTIEMNHACLAI) AS 'NGAYTIEMNHACLAI' FROM CHITIETTIME

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.NGAYTIEMNHACLAI, 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

2. Lấy tất cả vaccine

CREATE PROC sp\_GetAllVaccine

AS

BEGIN

SELECT vc.MAVACCINE, vc.TENVACCINE, vc.NHASX, vc.NGAYSX, vc.HANSD, 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 PHIEUTIEM INNER JOIN KHACHHANG ON KHACHHANG.MAKH = PHIEUTIEM.MAKH

WHERE MAPHIEUTIEM = @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.HANSD, 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.HANSD, 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.HANSD, 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.HANSD, 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.HANSD 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

**3.2.2. Phạm Đình Khương Duy**

**3.2.2.1. Truy vấn cơ bản**

1. Đưa ra mã thu ngân, tên thu ngân viết hóa đơn

- Đại số quan hệ

- Câu lệnh truy vấn

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

WHERE THUNGAN.MATHUNGAN=HOADON.MATHUNGAN

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

- Đại số quan hệ

- Câu lệnh truy vấn

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

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

- Đại số quan hệ

CHUCVU=’KẾ TOÁN’ (THUNGAN)

* Câu lệnh truy vấn

SELECT \* FROM THUNGAN WHERE CHUCVU= 'Kế toán'

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

- Đại số quan hệ

(ChucVu = ‘Thủ quỹ’ 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

DiaChi = ‘%Nghệ An’ (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:

(DIACHI = ‘%TP HCM’)  (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

SOLO( VACCINE.MALOAIVC = LOAIVC.MALOAIVC AND LOAIVACCINE=’COVID19’ (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:

(MATHUNGAN = ‘TN001’)  (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:

TENKH ( (MAGH = ‘GH001’)  (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:

* 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:

* Câu lệnh truy vấn:

SELECT MAKH, TENKH, TENGH FROM KHACHHANG, NGUOIGIAMHO

WHERE KHACHHANG.MAGH = NGUOIGIAMHO.MAGH

AND TENKH LIKE N'%Duy'

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

- Đại số quan hệ:

* Câu lệnh truy vấn:

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

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

- Đại số quan hệ:

* 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:

* 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:

* 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

ℑ COUNT(MAVACCINE)( VACCINE.MALOAIVC = LOAIVC.MALOAIVC AND LOAIVACCINE=’SỞI’ (VACCINE x 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:

NgayThu ( MaNV=’NV01’ (HOADON))

* Câu lệnh truy vấn:

SELECT MAHOADON FROM HOADON WHERE MATHUNGAN=’TN001’

18. Đưa ra các thu ngân là nam :

- Đại số quan hệ:

(GIOITINH=’NAM’)  (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:

* 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:

MAHOADON ( HOADON.MATHUNGAN = THUNGAN.MATHUNGAN AND HOTEN=’Khương Duy’ (HOADON x 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 tiêm của khách hàng có mã là ‘KH001’

SELECT COUNT(MAPHIEUTIEM) AS 'SOPHIEUTIEM'

FROM PHIEUTIEM

WHERE MAKH= 'KH001'

5. Cho biết tổng tiền của phiếu tiêm có mã bác sĩ là ‘BS001’

SELECT MABS, SUM(TONGTIEN) AS TONGTIEN

FROM PHIEUTIEM, HOADON

WHERE PHIEUTIEM.MAPHIEUTIEM=HOADON.MAPHIEUTIEM

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 tiêm của bác sĩ có họ ‘Phạm’

SELECT MAPHIEUTIEM FROM PHIEUTIEM, BACSI

WHERE PHIEUTIEM.MABS = BACSI.MABS

AND BACSI.HOTEN LIKE N'Phạm%'

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

SELECT \* FROM PHIEUTIEM

WHERE NGAYTIEM BETWEEN '2022-01-01' AND '2022-05-15'

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

SELECT BACSI.MABS, BACSI.HOTEN FROM PHIEUTIEM, BACSI

WHERE PHIEUTIEM.MABS = BACSI.MABS

AND maPhieu='PT001'

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

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. Giam so luong vaccine co san sau khi tao 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

CHUYENKHOA=’RĂNG HÀM MẶT’ (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:

* 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

MABS, HOTEN ( BACSY.MABS = PHIEUTIEM.MABS AND MAPHIEUTIEM=’PT001’ (BACSY x PHIEUTIEM))

* Câu lệnh truy vấn

SELECT BACSY.MABS, HOTEN FROM BACSY, PHIEUTIEM

WHERE BACSY.MABS=PHIEUTIEM.MABS

AND PHIEUTIEM.MAPHIEUTIEM='PT001'

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

* Đại số quan hệ

HOTEN,SDT( MAGH=’GH001’ 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

MAKH, TENKH( KHACHHANG.MAGH = NGUOIGIAMHO.MAGH AND NGUOIGIAMHO.HOTEN=’Hồ Nguyễn Nguyên’ (NGUOIGIAMHO x 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

VACCINE.TENVACCINE, VACCINE.DONGIA ( VACCINE.MAVACCINE = CHITIETTIEM.MAVC AND MAPHIEUTIEM=’PT005’ (VACCINE x 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:

ℑ COUNT(MABS)( BACSY.MABS=PHIEUTIEM.MABS AND GIOITINH=’NAM’ ( BACSY x 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

- 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:

* 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

VACCINE.SOLO( VACCINE.MAQUANKHO = QUANKHO.MAQUANKHO AND QUANKHO.MAQUANKHO=’QK001’ (VACCINE x QUANKHO))

* Câu lệnh truy vấn

SELECT SOLO FROM VACCINE, QUANKHO

WHERE QUANKHO.MAQUANKHO=VACCINE.MAQUANKHO

AND QUANKHO.MAQUANKHO='QK001'

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

- Đại số quan hệ

MAPT( VACCINE.MAVACCINE = PHIEUTIEM.MAVACCINE AND TENVACCINE=’ASTRA’ (VACCINE x 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:

MAPHIEUTIEM ( PHIEUTIEM.MABS=BACSY.MABS AND BACSY.HOTEN=’NGUYÊN’ (PHIEUTIEM x 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:

σ­YEAR(HSD) = YEAR(GETDATE())­­(Vacxin)

* Câu lệnh truy vấn

SELECT\* FROM VACCINE WHERE YEAR(HANSD) = YEAR(GETDATE())

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

- Đại số quan hệ:

π­maKH­(σ­tuoi < 30­(KhachHang)) π­maKH­(σ(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 'Nguyên'

- Đại số quan hệ:

MAGH,TENGH( NGUOIGIAMHO.MAGH = KHACHHAGN.MAGH AND GIOITINH=’Nữ’ AND TENKH=’%Nguyên’ (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'%Nguyên'

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

- Đại số quan hệ

SOLO( VACCINE.MALOAIVC = LOAIVC.MALOAIVC AND LOAIVACCINE=’CẢM CÚM’ (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

THUNGAN.HOTEN( HOADON.MATHUNGAN = THUNGAN.MATHUNGAN AND MAPHIEUTIEM=’PT005’(HOADON x THUNGAN))

* Câu lệnh truy vấn

SELECT HOTEN FROM HOADON, THUNGAN

WHERE HOADON.MATHUNGAN=THUNGAN.MATHUNGAN

AND MAPHIEUTIEM = 'PT005'

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

* Đại số quan hệ

ℑ COUNT(MAPHIEUTIEM)( VACCINE.MAVACCINE = CHITIETTIEM.MAVACCINE AND MALOAIVC=’LVC001’ (VACCINE x CHITIETTIEM))

* Câu lệnh truy vấn

SELECT COUNT(MAPHIEUTIEM) AS SOPHIEUTIEM FROM VACCINE, CHITIETTIEM

WHERE VACCINE.MAVACCINE=CHITIETTIEM.MAVACCINE

AND MALOAIVC='LVC001'

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

- Đại số quan hệ:

(DIACHI = ‘%TP HCM’)  (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

ℑ SUM(DONGIA)( CHITIETTIEM.MAVACCINE = VACCINE.MAVACCINE AND MAPHIEUTIEM=’PT005’ (CHITIETTIEM x VACCINE))

* Câu lệnh truy vấn

SELECT SUM(DONGIA) AS TONGTIE FROM CHITIETTIEM, VACCINE

WHERE CHITIETTIEM.MAVACCINE=VACCINE.MAVACCINE

AND MAPHIEUTIEM = ‘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(PHIEUTIEM.MAPHIEUTIEM)

FROM BACSY JOIN PHIEUTIEM ON PHIEUTIEM.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'TRƯỞNG 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.HANSD, 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(), HANSD) AS 'SONGAYCONLAI' FROM VACCINE

WHERE DATEDIFF(DAY, GETDATE(), HANSD)<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 CHITIETTIEM 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, @HANSD 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)

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,

HANSD = @HANSD,

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 CHITIETTIEM 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 xoá 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. Đinh 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

MAVACCINE, VACCINE.TENVACCINE, LOAIVC.LOAIVACCINE ( VACCINE.MAVACCINE = CHITIETTIEM.MAVACCINE AND VACCINE.MALOAIVC = LOAIVC.MALOAIVC AND MAPHIEUTIEM =’PT001’ (VACCINE x CHITIETTIEM x 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:

TENVACCINE ( VACCINE.MALOAIVC=LOAIVC.MALOAIVC AND LOAIVACCINE= ‘%ho gà%’ (VACCINE x 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

HOTEN,SDT( 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

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:

(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

ℑ SUM(SOLUONGCOSAN)( VACCINE.MALOAIVC = LOAIVC.MALOAIVC AND LOAIVACCINE=’HO GÀ’ (VACCINE x 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

ℑ SUM(DONGIA)( 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

ℑ COUNT(MABS)( BANGCAP=’TIÊN SĨ’ 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

SOLO( VACCINE.MALOAIVC = LOAIVC.MALOAIVC AND LOAIVACCINE=’HO GÀ’ (VACCINE x LOAIVC))

- Câu lệnh truy vấn

SELECT SOLO AS SOVACCINE FROM VACCINE, LOAIVC

WHERE VACCINE.MALOAIVC=LOAIVC.MALOAIVC

AND LOAIVACCINE=N'HO GÀ'

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

- Đại số quan hệ

- 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:

ℑ COUNT() (DIACHI = ‘%Đắc Lắc’)  (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:

ℑ COUNT()( QUANKHO.MAQUANKHO=VACCINE.MAQUANKHO AND GIOITINH=’NỮ’ ( QUANKHOx 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:

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:

MAHOADON ( 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

ℑ SUM(GIABAN)(  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

MAVACCINE, TENVACCINE ( 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:

NgayThu ( MaNV=’NV01’ (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

(ChucVu = ‘Thủ quỹ’ AND hoTen = ‘Đinh%’)  (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

ℑ COUNT(TAIKHOAN.MATAIKHOAN)( TAIKHOAN.MATAIKHOAN= QUANTRI.MAQUANTRI AND (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

ℑ COUNT()( VACCINE.MAVACCINE = PHIEUTIEM.MAVACCINE AND TENVACCINE=’ASTRA’ (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 CHITIETTIEM

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

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 PHIEUTIEM INNER JOIN CHITIETTIEM ON CHITIETTIEM.MAPHIEUTIEM = PHIEUTIEM.MAPHIEUTIEM

WHERE CHITIETTIEM.MAPHIEUTIEM IN

(SELECT DISTINCT MAPHIEUTIEM 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(), HANSD) AS 'SONGAYCONLAI' FROM VACCINE

WHERE DATEDIFF(DAY, GETDATE(), HANSD)<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, PHIEUTIEM

WHERE KHACHHANG.MAKH=PHIEUTIEM.MAKH

AND CHITIETTIEM.MAPHIEUTIEM=PHIEUTIEM.MAPHIEUTIEM

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 "Ngay", 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 "Ngay", 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.HANSD, 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**

* 1. **Thiết kế giao diện**
     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

* + 1. **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.
* 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.

Graphical user interface

Description automatically generated

Menu hiển thị theo người dùng

* + 1. **Các chức năng chính**
* Chương trình bao gồm các chức năng chính sau:

**4.1.3.1. 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**

**Graphical user interface

Description automatically generated**

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

**4.1.3.2. Giao diện quản lí người dùng**

**a. Chức năng quản lý người dùng**

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

**b. Thiết kế giao diện**

**A screenshot of a computer

Description automatically generated**

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 khảu 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 tài 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

- 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 khảu 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 LÍ 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

**4.1.3.3. Giao diện Lịch sử tiêm**

**a. Chức năng Lịch sử tiêm**

**+** 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**

**Graphical user interface, application

Description automatically generated**

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 thạm 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.NGAYTIEMNHACLAI, 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

**4.1.3.4. 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 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**

**Graphical user interface, application, table

Description automatically generated**

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

**Graphical user interface, application

Description automatically generated**

Form thêm xóa loại vaccine

**Table

Description automatically generated with low confidence**

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.HANSD, vc.SOLO, vc.SOLUONGCOSAN, vc.DONGIA, lvc.LOAIVACCINE

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.HANSD, 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.HANSD 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.HANSD, 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.HANSD, 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.HANSD, 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.HANSD, 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, @HANSD 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,

HANSD,

SOLO,

SOLUONGCOSAN,

DONGIA,

MALOAIVC,

MAQUANKHO,

MAQUANTRI

)

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

**4.1.3.5. 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 hàng 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 hàng 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. 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 phiêu 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**

**Graphical user interface, application, table

Description automatically generated**

Giao diện tạo phiếu tiêm

Graphical user interface, table

Description automatically generated

Form cập nhật danh sách phiếu tiêm

Table

Description automatically generated

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.HANSD, 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, -- MAKH - char(10)

@TENKH, -- TENKH - nvarchar(20)

@NGAYSINH, -- NGAYSINH - date

@GIOITINH, -- GIOITINH - nvarchar(4)

@TIEUSU, -- TIEUSU - nvarchar(250)

@MAGH -- MAGH - char(10)

)

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, -- MAPHIEUTIEM - char(10)

@NGAYTIEM, -- NGAYTIEM - date

@MAKH, -- MAKH - char(10)

@MABS -- MABS - char(10)

)

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, @NGAYNHACLAI DATE, @LIEULUONG FLOAT)

AS

BEGIN

INSERT INTO dbo.CHITIETTIEM

(

MAPHIEUTIEM,

MAVACCINE,

GIABAN,

MUITHU,

NGAYTIEMNHACLAI,

LIEULUONG

)

VALUES

( @MAPHIEUTIEM, -- MAPHIEUTIEM - char(10)

@MAVACCINE, -- MAVACCINE - char(10)

@GIABAN, -- GIABAN - int

@MUITHU, -- MUITHU - int

@NGAYNHACLAI, -- NGAYTIEMNHACLAI - date

@LIEULUONG -- LIEULUONG - float

)

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.NGAYTIEMNHACLAI, 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\_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

- Trigger trg\_TaoChiTietTiem: được sử dụng để tự động giảm 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

**4.1.3.6. 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 chi 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 phiểu tiêm đó và hiển thị tổng tiển 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**

**Graphical user interface, application

Description automatically generated**

Giao diện thanh toán hóa đơn

Graphical user interface, table

Description automatically generated

Form cập nhật danh sách hóa đơn

Table

Description automatically generated

Mẫu hóa đơn thanh toán

**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.NGAYTIEMNHACLAI, 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\_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 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

- 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), @MAPHIEUTIEM CHAR(10))

AS

BEGIN

INSERT INTO dbo.HOADON

(

MAHOADON,

CHIETKHAU,

NGAYTHU,

TONGTIEN,

MAGH,

MATHUNGAN,

MAPHIEUTIEM

)

VALUES

( @MAHOADON, -- MAHOADON - char(10)

@CHIETKHAU, -- CHIETKHAU - float

@NGAYTHU, -- NGAYTHU - date

@TONGTIEN, -- TONGTIEN - bigint

@MAGH, -- MAGH - char(10)

@MATHUNGAN, -- MATHUNGAN - char(10)

@MAPHIEUTIEM -- MAPHIEUTIEM - char(10)

)

END

- Stored procedure sp\_ CheckPaymentStatus: được sử dụng để kiểm phiếu tiêm 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, SDT = @SDT 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

**4.1.3.7. Giao diện Thống kê**

**a. Chức năng Thống kê**

**-** Phần mềm thống kê theo thống kê doanh thu, thống kê theo tỷ trọng vaccine từng loại và thống kê theo top vaccine được dùng nhiều nhất.

- Ở thống kê doanh thu, mặc định phần mềm hiển thị biểu đồ doanh thu trong 2 thàng 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 thi qua một combobox ở dưới biểu đồ.

**b. Thiết kế giao diện**

**Graphical user interface, chart, application

Description automatically generated**

Giao diện thống kê

Table

Description automatically generated

Mẫu hóa báo cáo thống kê doanh thu

**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 @NgayDau đến ngày @NgayCuoi

CREATE PROC [dbo].[sp\_GetHoaDonINTIME\_TK] (@NgayDau DATE, @NgayCuoi DATE)

AS

BEGIN

SELECT NGAYTHU AS "Ngay", 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.CHITIETTIEM ON CHITIETTIEM.MAPHIEUTIEM = PHIEUTIEM.MAPHIEUTIEM

WHERE CHITIETTIEM.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 "Ngay", 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