

BÁO CÁO ĐỒ ÁN THỰC HÀNH

MÔN HỆ QUẢN TRỊ CƠ SỞ DỮ LIỆU

GVHD: Tiết Gia Hồng, Hồ Thị Hoàng Vy

MỤC LỤC

Contents

MỤC LỤC	2
BẢNG THIẾT KẾ THỰC THỂ KẾT HỢP	4
I. Lược đồ thực thể kết hợp	4
Chú thích	4
Business Rules	4
Hình lược đồ ER	5
II. Ràng buộc dữ liệu bổ sung.....	5
III. Mô hình cơ sở dữ liệu quan hệ	8
IV. Các loại người dùng	9
XÁC ĐỊNH CHỨC NĂNG HỆ THỐNG VÀ TÌNH HUỐNG TRANH CHẤP	9
I. Các chức năng của hệ thống	9
1. Chức năng cho DỪNG CHUNG	9
2. Phân hệ Quản trị viên	10
3. Phân hệ Nhân viên	10
4. Phân hệ Nha sĩ	10
5. Phân hệ Khách hàng.....	11
II. Cấu trúc thư mục SQL	11
Lưu ý	11
III. Ứng dụng QLNhaKhoa	12
IV. Xác định tình huống tranh chấp	13
V. Thiết kế giao diện chức năng	14
1. Đăng nhập	14
2. Phân hệ Khách hàng.....	15
3. Phân hệ Nha sĩ	20
4. Phân hệ Nhân viên	26
5. Phân hệ Quản trị viên	34
CÀI ĐẶT TÌNH HUỐNG TRANH CHẤP	38
I. Sinh viên thực hiện: 21127149 – Huỳnh Minh Quang	38
1. Tình huống 1: Admin thêm thông tin thuốc nhưng xảy ra lỗi phải rollback. Trước đó, nhân viên vào xem thuốc.....	38
II. Sinh viên thực hiện: 21127296 – Đặng Hà Huy	41

2. Tình huống 2: Nhân viên đang tìm thuốc bằng tên thì admin thay đổi tên của một thuốc bất kì có trong danh sách sẽ hiển thị.	41
III. Sinh viên thực hiện: 21127135 – Diệp Hữu Phúc.....	46
3. Tình huống 3: Nhân viên tìm thuốc có số lượng tồn kho nhỏ hơn X thì admin thêm thuốc mới cũng thỏa điều kiện đó.	46
4. Tình huống 4: Admin cập nhật số lượng tồn kho của thuốc X nhưng chưa hoàn thành update thì nha sĩ cập nhật số lượng của thuốc X trong một toa thuốc bất kì.	50
IV. Sinh viên thực hiện: 21127004 – Trần Nguyễn An Phong	55
5. Tình huống 5: Nhân viên muốn cập nhật thông tin tài khoản của mình nhưng chưa hoàn thành thì đồng thời cũng có một admin muốn cập nhật thông tin của nhân viên đó.	55
6. Tình huống 6: Nha sĩ trong lúc cập nhật thông tin toa thuốc X thì lại thực hiện xóa toa thuốc này.	57
XỬ LÝ TÌNH HUỐNG TRANH CHẤP.....	61
I. Sinh viên thực hiện: 21127149 – Huỳnh Minh Quang	61
1. Tình huống 1: Admin thêm thông tin thuốc nhưng xảy ra lỗi phải rollback. Trước đó, nhân viên vào xem thuốc.....	61
II. Sinh viên thực hiện: 21127296 – Đặng Hà Huy	65
2. Tình huống 2: Nhân viên đang tìm thuốc bằng tên thì admin thay đổi tên của một thuốc bất kì có trong danh sách sẽ hiển thị.	65
III. Sinh viên thực hiện: 21127135 – Diệp Hữu Phúc.....	70
3. Tình huống 3: Nhân viên tìm thuốc có số lượng tồn kho nhỏ hơn X thì admin thêm thuốc mới cũng thỏa điều kiện đó.	70
4. Tình huống 4: Admin cập nhật số lượng tồn kho của thuốc X nhưng chưa hoàn thành update thì nha sĩ cập nhật số lượng của thuốc X trong một toa thuốc bất kì.	74
IV. Sinh viên thực hiện: 21127004 – Trần Nguyễn An Phong	79
5. Tình huống 5: Nhân viên muốn cập nhật thông tin tài khoản của mình nhưng chưa hoàn thành thì đồng thời cũng có một admin muốn cập nhật thông tin của nhân viên đó.	79
6. Tình huống 6: Nha sĩ trong lúc cập nhật thông tin toa thuốc X thì lại thực hiện xóa toa thuốc này.	82

BẢNG THIẾT KẾ THỰC THỂ KẾT HỢP

I. Lược đồ thực thể kết hợp

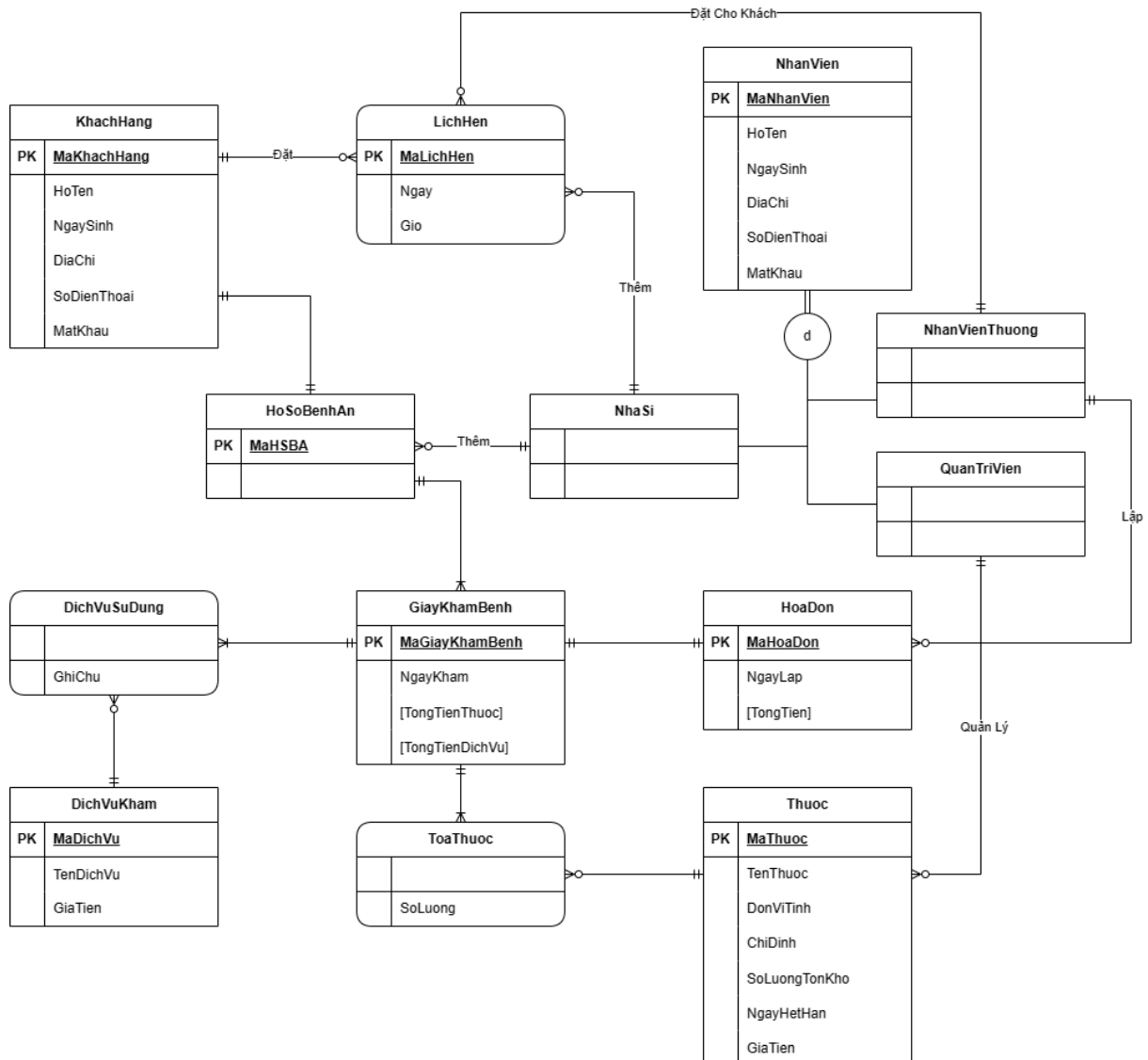
Chú thích

- **Có (thể ...) nhiều:** 0 – many.
- **Ít nhất một:** 1 – many.
- **Tối đa một:** 0 – 1.
- **Chỉ/duy nhất:** 1 – 1.
- Nhân viên bao gồm nhân viên thường, nha sĩ, quản trị viên.

Business Rules

- Mỗi khách hàng có thể đặt nhiều lịch hẹn. Mỗi lịch hẹn thuộc về một khách hàng.
- Mỗi khách hàng chỉ có một hồ sơ bệnh án. Mỗi hồ sơ bệnh án chỉ thuộc về một khách hàng.
- Mỗi nha sĩ có thể thêm nhiều hồ sơ bệnh án. Mỗi hồ sơ bệnh án được thêm bởi một nha sĩ.
- Mỗi nha sĩ có thể có nhiều lịch hẹn. Mỗi lịch hẹn chỉ thuộc về một nha sĩ.
- Mỗi nha sĩ có thể thêm nhiều lịch cá nhân. Mỗi lịch cá nhân chỉ được thêm bởi một nha sĩ.
- Mỗi hồ sơ bệnh án có thể có nhiều giấy khám bệnh. Mỗi giấy khám bệnh chỉ thuộc về một hồ sơ bệnh án.
- Mỗi giấy khám bệnh có duy nhất một toa thuốc. Mỗi toa chỉ thuộc về một giấy khám bệnh.
- Mỗi giấy khám bệnh có duy nhất một hóa đơn thanh toán. Mỗi hóa đơn thanh toán chỉ thuộc về một giấy khám bệnh.
- Mỗi toa thuốc có ít nhất một loại thuốc. Mỗi loại thuốc có thể thuộc về nhiều toa thuốc.
- Mỗi quản trị viên có thể quản lý nhiều loại thuốc. Mỗi loại thuốc chỉ được quản lý bởi một quản trị viên.
- Mỗi nhân viên có thể đặt nhiều lịch hẹn. Mỗi lịch hẹn được đặt bởi một nhân viên (trường hợp nhân viên đặt hẹn khám cho bệnh nhân).
- Mỗi nhân viên có thể lập nhiều hóa đơn. Mỗi hóa đơn chỉ được lập bởi một nhân viên.
- Mỗi giấy khám bệnh có thể có một hoặc nhiều phiếu dịch vụ sử dụng. Mỗi phiếu dịch vụ sử dụng thuộc về một giấy khám bệnh.
- Mỗi phiếu dịch vụ sử dụng có duy nhất 1 dịch vụ khám. Mỗi dịch vụ khám có thể thuộc về nhiều phiếu dịch vụ sử dụng.
- Mỗi toa thuốc có thể có một hoặc nhiều phiếu thuốc sử dụng. Mỗi phiếu thuốc sử dụng thuộc về một toa thuốc.
- Mỗi phiếu thuốc sử dụng chứa một loại thuốc. Mỗi loại thuốc có thể nằm trong nhiều phiếu thuốc sử dụng.

Hình lược đồ ER



II. Ràng buộc dữ liệu bổ sung

- Hóa đơn thanh toán cho một giấy khám bệnh là tổng tiền của tất cả dịch vụ và một toa thuốc thuộc về giấy khám bệnh đó.

R1	Thêm	Xóa	Sửa
HoaDon	+	-	+ (TongTien)
ToaThuoc	+	-	+ (TongTien)
DichVuSuDung	+	-	-
DichVuKham	-	-	+ (GiaTien)

2. Chỉ có thể ra hóa đơn thanh toán nếu tìm thấy hồ sơ và giấy khám bệnh của bệnh nhân.

R2	Thêm	Xóa	Sửa
HoaDon	+	-	-

3. Quản trị viên chỉ có thể thêm một loại thuốc với số lượng thêm vào ít nhất là 1 và số lượng tồn kho của một loại thuốc phải là một số nguyên dương.

R3	Thêm	Xóa	Sửa
Thuoc	+	-	+ (SoLuongTonKho)

4. Quản trị viên chỉ có thể thực hiện giảm số lượng tồn kho của một loại thuốc.

R4	Thêm	Xóa	Sửa
Thuoc	-	-	+ (SoLuongTonKho)

5. Quản trị viên chỉ có thể xóa thuốc khi ngày hiện tại đã vượt qua ngày hết hạn của thuốc.

R5	Thêm	Xóa	Sửa
Thuoc	-	+	-

6. Chỉ có thể đặt hẹn với một nha sĩ nếu không có lịch nào khác với nha sĩ đó bị trùng thời gian.

R6	Thêm	Xóa	Sửa
LichHen	+	-	+ (Ngay, Gio, MaNhaSi)

7. Chỉ có thể ra toa thuốc nếu các loại thuốc trên toa thuốc còn tồn kho.

R7	Thêm	Xóa	Sửa
ThuocSuDung	+	-	-

8. Nha sĩ chỉ có thể thêm lịch cá nhân hoặc sửa lịch cá nhân của mình nếu không có lịch hẹn nào khác của nha sĩ đó trùng thời gian.

R8	Thêm	Xóa	Sửa
LichHen	+	-	+ (Ngay, Gio)

9. Nha sĩ chỉ có thể sửa lịch cá nhân do chính nha sĩ thêm vào.

R9	Thêm	Xóa	Sửa
LichHen	+	-	+ (MaNhanVienDatLich)

10. Giới tính của một nhân viên là “Nam” hoặc “Nữ”.

R10	Thêm	Xóa	Sửa
NhanVien	+	-	+ (GioiTinh)

11. Giới tính của một bệnh nhân là “Nam” hoặc “Nữ”.

R11	Thêm	Xóa	Sửa
KhachHang	+	-	+ (GioiTinh)

12. Số điện thoại của nhân viên là một chuỗi số có từ 9 đến 11 kí tự.

R12	Thêm	Xóa	Sửa
NhanVien	+	-	+ (SoDienThoai)

13. Ngày hẹn của một lịch hẹn phải là ngày hợp lệ trong tháng.

R13	Thêm	Xóa	Sửa
LichHen	+	-	+ (Ngay)

14. Giờ hẹn của một lịch hẹn phải nằm trong khung giờ làm việc của bệnh viện từ 8h đến 17h.

R14	Thêm	Xóa	Sửa
LichHen	+	-	+ (Gio)

15. Mật khẩu của một khách hàng phải có độ dài ít nhất là 8 và phải bao gồm chữ cái và số.

R15	Thêm	Xóa	Sửa
KhachHang	+	-	+ (MatKhau)

16. Số điện thoại của khách hàng là một chuỗi số có từ 9 đến 11 kí tự.

R16	Thêm	Xóa	Sửa
-----	------	-----	-----

KhachHang	+	-	+ (SoDienThoai)
-----------	---	---	-----------------

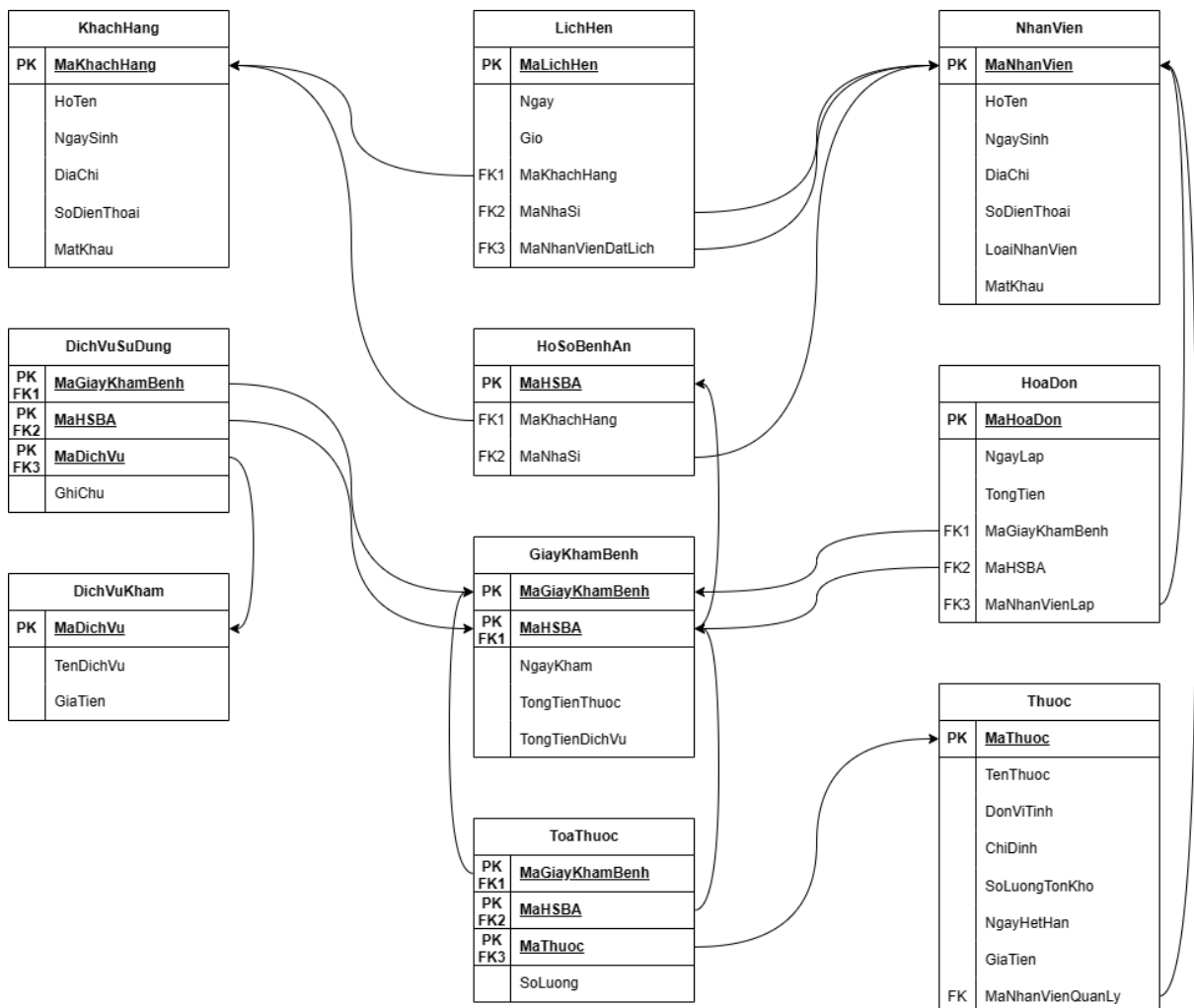
17. Đơn vị tính của một loại thuốc là “viên”, “gói” hoặc “hộp”.

R17	Thêm	Xóa	Sửa
Thuoc	+	-	+ (DonViTinh)

18. Ngày hết hạn của thuốc phải là một ngày hợp lệ trong năm và không được sớm hơn ngày thêm loại thuốc vào kho.

R18	Thêm	Xóa	Sửa
Thuoc	+	-	+ (NgayHetHan)

III. Mô hình cơ sở dữ liệu quan hệ



IV. Các loại người dùng

Các loại người dùng	Vai trò và phân quyền
Khách hàng (Bệnh nhân)	<ul style="list-style-type: none"> - Vai trò: <ul style="list-style-type: none"> + Được thực hiện bởi tất cả người dùng chưa đăng nhập hoặc đăng nhập với quyền của khách hàng. - Quyền: <ul style="list-style-type: none"> + Đặt lịch hẹn cho bản thân. + Xem và chỉnh sửa hồ sơ của họ. + Truy cập và xem giấy khám bệnh và thuốc của bản thân.
Nha sĩ	<ul style="list-style-type: none"> - Vai trò: <ul style="list-style-type: none"> + Được thực hiện bởi nha sĩ (nhân viên y tế). - Quyền: <ul style="list-style-type: none"> + Thêm và quản lý hồ sơ bệnh án của bệnh nhân. + Xem danh sách lịch hẹn của mình. + Thêm lịch cá nhân và cập nhật lịch cá nhân của nha sĩ. + Thêm giấy khám bệnh cho hồ sơ bệnh án của bệnh nhân. + Thêm toa thuốc và dịch vụ sử dụng vào hồ sơ bệnh nhân.
Quản trị viên	<ul style="list-style-type: none"> - Vai trò: <ul style="list-style-type: none"> + Được thực hiện bởi người quản lý hệ thống hoặc quản trị viên. - Quyền: <ul style="list-style-type: none"> + Quản lý loại thuốc, bao gồm bổ sung, sửa đổi và xóa loại thuốc. + Quản lý người dùng, bao gồm bổ sung, sửa đổi và xóa tài khoản khách hàng, nha sĩ và nhân viên.
Nhân viên	<ul style="list-style-type: none"> - Vai trò: <ul style="list-style-type: none"> + Được thực hiện bởi nhân viên y tế hoặc nhân viên quầy đăng ký lịch hẹn. - Quyền: <ul style="list-style-type: none"> + Quản lý và đặt lịch khám cho bệnh nhân. + Thêm lịch hẹn vào hồ sơ bệnh án của bệnh nhân. + Thêm và quản lý hoá đơn thanh toán cho đơn thuốc và dịch vụ.

XÁC ĐỊNH CHỨC NĂNG HỆ THỐNG VÀ TÌNH HUỐNG TRANH CHẤP

I. Các chức năng của hệ thống

1. Chức năng cho DÙNG CHUNG

STT	Chức năng	Mô tả hoạt động
ALL1	Đăng nhập	Đăng nhập vào hệ thống Quản lý Nha khoa dựa vào tài khoản và mật khẩu.
ALL2	Đăng ký	Đăng ký tài khoản, mật khẩu và điền thông tin theo yêu cầu.
ALL3	Xem thông tin tài khoản	Xem các thông tin của tài khoản như tên, số điện thoại,...

ALL4	Cập nhật thông tin tài khoản	Cập nhật, chỉnh sửa thông tin tài khoản trên hệ thống.
------	------------------------------	--------------------------------------------------------

2. Phân hệ Quản trị viên

STT	Chức năng	Mô tả hoạt động
AD1	Tạo tài khoản đăng nhập	Admin sẽ tạo tài khoản đăng nhập sẵn cho admin, nhân viên, nha sĩ theo yêu cầu và quy tắc dành cho mật khẩu khởi tạo.
AD2	Thêm, xóa, sửa tài khoản	Admin có thể tạo mới, xóa, hoặc sửa tài khoản trong hệ thống.
AD3	Khóa, kích hoạt tài khoản	Admin có thể khóa tài khoản hiện đang có trong hệ thống khi có vấn đề xảy ra, và tái kích hoạt khi đã được giải quyết.
AD4	Cấp quyền thao tác trên dữ liệu	Cấp quyền thao tác cho từng phân hệ người dùng trên cơ sở dữ liệu.
AD5	Xem, thêm, chỉnh sửa, xóa thông tin thuốc	Admin có thể thêm, xóa, sửa thông tin thuốc được cung cấp tại phòng khám.

3. Phân hệ Nhân viên

STT	Chức năng	Mô tả hoạt động
NV1	Đặt, thay đổi lịch khám cho khách hàng	Nhân viên có thể tiếp nhận yêu cầu đặt lịch, đăng ký khám từ khách hàng, và sau đó nhập vào hệ thống.
NV2	Xem hồ sơ bệnh án của khách hàng	Nhân viên có thể truy cập vào hồ sơ bệnh án của khách hàng để xem các thông tin chung của khách hàng như họ tên, địa chỉ, ngày sinh, số điện thoại và các nha sĩ thực hiện khám cho bệnh nhân.
NV3	Xem giấy khám bệnh của khách hàng	Nhân viên có thể xem các giấy khám bệnh của khách hàng để kiểm tra các thông tin về ngày khám, tiền thuốc, tiền khám của khách hàng.
NV4	Xem dịch vụ sử dụng của khách hàng	Nhân viên có thể xem danh sách các dịch vụ của khách hàng đã sử dụng.
NV5	Xem toa thuốc của khách hàng	Nhân viên có thể xem danh sách toa thuốc của các giấy khám bệnh của khách hàng.
NV6	Xem thuốc	Nhân viên có thể xem danh mục thuốc gồm: mã thuốc, tên thuốc, đơn vị tính, chỉ định, số lượng tồn trong kho và ngày hết hạn.
NV7	Lập, chỉnh sửa hóa đơn cho khách hàng	Nhân viên có thể lập hóa đơn cho một giấy khám bệnh của một khách hàng và chỉnh sửa hóa đơn này nếu hóa đơn chưa được thanh toán.

4. Phân hệ Nha sĩ

STT	Chức năng	Mô tả hoạt động
NS1	Xem, thêm, chỉnh sửa hồ sơ bệnh án	Nha sĩ có thể thêm, sửa hồ sơ bệnh án của bệnh nhân.
NS2	Xem, thêm, chỉnh sửa giấy khám bệnh	Nha sĩ có thể thêm, sửa giấy khám bệnh của bệnh nhân.
NS3	Xem, thêm, chỉnh sửa, xóa toa thuốc	Nha sĩ có thể thêm, sửa toa thuốc của bệnh nhân.
NS4	Xem, thêm, chỉnh sửa, xóa dịch vụ sử dụng	Nha sĩ có thể thêm, sửa dịch vụ sử dụng của bệnh nhân.
NS5	Xem, thêm, chỉnh sửa, xóa lịch hẹn cá nhân	Nha sĩ có thể thêm, xóa, sửa lịch hẹn cá nhân (do khách hàng chủ động đặt trực tiếp với nha sĩ).

5. Phân hệ Khách hàng

STT	Chức năng	Mô tả hoạt động
KH1	Xem, thêm lịch hẹn cho bản thân	Khách hàng có thể tự đặt lịch hẹn thông qua hệ thống đặt lịch hẹn, và xem lại các lịch hẹn đã đặt.
KH2	Xem hồ sơ bệnh án của bản thân	Khách hàng xem hồ sơ bệnh án của mình.

II. Cấu trúc thư mục SQL

Nhằm phục vụ cho sự dễ dàng trong việc quản lý, hợp tác, và cải tiến trong quá trình cài đặt cơ sở dữ liệu, source code SQL được chia thành nhiều thành phần riêng lẻ nhau.

- **createDB.sql** – Script xây dựng database và tạo các bảng.
- **funcs** – Thư mục chứa script function cho mỗi bảng, và **misc.sql** cho function dùng chung.
- **storedProcs** – Thư mục chứa script stored procedure cho mỗi bảng, và **misc.sql** cho stored procedure dùng chung.
- **permissions.sql** – Script phân quyền cho database.
- **populateData** – Thư mục chứa dữ liệu cho mỗi bảng.
- **concurrency** – Thư mục chứa script cài đặt, và giải quyết các tình huống tranh chấp.

Và, đơn giản hóa việc chạy tất cả thành phần xây dựng cơ sở dữ liệu với **setupDB.bat**. Đây là batch file sử dụng SQLCMD để gọi chạy các script SQL theo thứ tự đã được sắp xếp sẵn.

Lưu ý

1. **Chú ý bật SQL Server** trước khi chạy **setupDB.bat**.
2. Việc chạy thủ công từng script là hoàn toàn có thể, tuy nhiên vẫn nên tuân theo thứ tự đã được tính toán trong **setupDB.bat**. Dưới đây là ví dụ minh họa cho thấy đã thực hiện xong.

```

C:\Windows\system32\cmd.exe
INVALID MAGIAYKHAMBENH
Msg 50000, Level 16, State 1, Server DESKTOP-0VDG7HL, Procedure USP_HOADON_INS, Line 11
INVALID MAGIAYKHAMBENH
Msg 50000, Level 16, State 1, Server DESKTOP-0VDG7HL, Procedure USP_HOADON_INS, Line 11
INVALID MAGIAYKHAMBENH
Msg 50000, Level 16, State 1, Server DESKTOP-0VDG7HL, Procedure USP_HOADON_INS, Line 11
INVALID MAGIAYKHAMBENH

(1 rows affected)
---- Granting Permissions...
Changed database context to 'Nhom3_QLNhaKhoa'.
Msg 15025, Level 16, State 1, Server DESKTOP-0VDG7HL, Line 5
The server principal 'KH001' already exists.
Msg 15025, Level 16, State 1, Server DESKTOP-0VDG7HL, Line 10
The server principal 'KH002' already exists.
Msg 15025, Level 16, State 1, Server DESKTOP-0VDG7HL, Line 15
The server principal 'KH003' already exists.
Msg 15025, Level 16, State 1, Server DESKTOP-0VDG7HL, Line 4
The server principal 'NV001' already exists.
Msg 15025, Level 16, State 1, Server DESKTOP-0VDG7HL, Line 9
The server principal 'NV002' already exists.
Msg 15025, Level 16, State 1, Server DESKTOP-0VDG7HL, Line 4
The server principal 'NS001' already exists.
Msg 15025, Level 16, State 1, Server DESKTOP-0VDG7HL, Line 9
The server principal 'NS002' already exists.
Msg 15025, Level 16, State 1, Server DESKTOP-0VDG7HL, Line 4
The server principal 'AD001' already exists.
Msg 15025, Level 16, State 1, Server DESKTOP-0VDG7HL, Line 9
The server principal 'AD002' already exists.
Press any key to continue . . .
  
```

III. Ứng dụng QLNhaKhoa

QLNhaKhoa được cài đặt bằng Windows Forms.

- **QLNhaKhoa** – Thư mục chứa source code WinForms.
- **QLNK_Release** – Thư mục chứa bản release (.exe) của ứng dụng.

Để sử dụng app, lưu ý bật **SQL Server**, và hoàn thành việc thiết lập cơ sở dữ liệu như đã miêu tả ở [Cấu trúc thư mục SQL](#), sau đó vào thư mục **QLNK_Release** và chạy **QLNhaKhoa.exe**.

Tất cả stored procedure được dùng khi cài đặt và giải quyết các tình huống tranh chấp đều được sử dụng trong app. Do đó, để demo các tình huống trên app, thực hiện chạy file chứa proc tranh chấp muốn demo để thay thế các proc thông thường. Sau đó, dựa theo **tran1.sql** và **tran2.sql** tương ứng để thực hiện các chức năng liên quan.

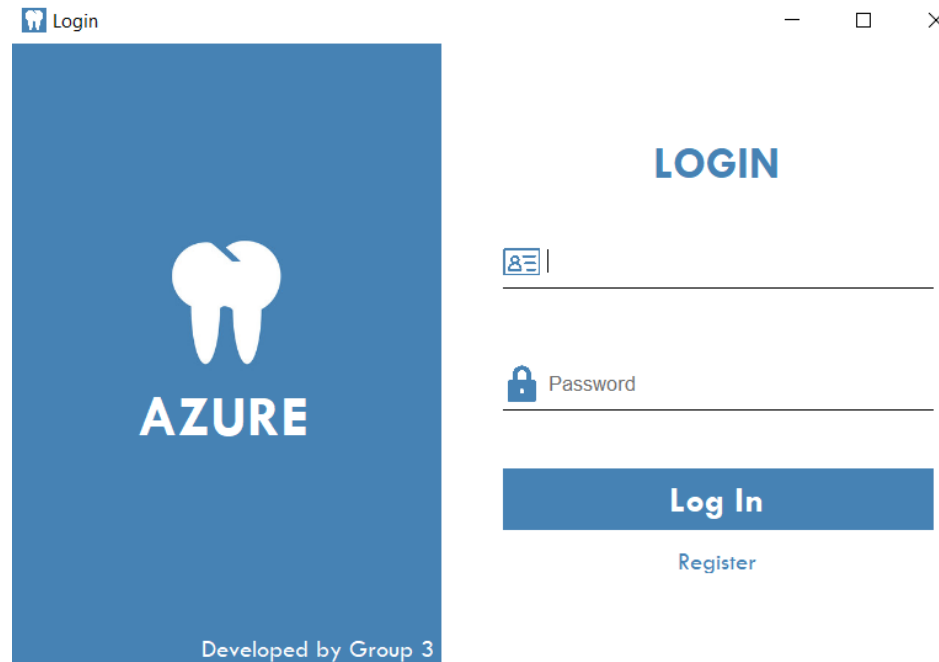
Tên đăng nhập và mật khẩu của mỗi phân hệ có thể được tìm thấy trong các file .sql tương ứng (**data_KhachHang**, **data_NhanVien**, **data_NhaSi**, **data_QuanTriVien**). Tùy vào tài khoản đăng nhập, app sẽ có 4 kiểu giao diện tương ứng với loại tài khoản, được thể hiện rõ ở phần [Thiết kế giao diện chức năng](#).

IV. Xác định tình huống tranh chấp

STT	Chức năng 1	Người dùng	Chức năng 2	Người dùng	Lỗi tranh chấp
1	Thêm thuốc	Quản trị viên	Xem thuốc	Nhân viên	Dirty Read Admin thêm thông tin thuốc nhưng xảy ra lỗi phải rollback. Trước đó, nhân viên vào xem thuốc.
2	Tìm thuốc bằng tên	Nhân viên	Thay đổi tên thuốc	Quản trị viên	Unrepeatable Read Nhân viên đang tìm thuốc bằng tên thì admin thay đổi tên của một thuốc bất kì có trong danh sách sẽ hiển thị.
3	Tìm thuốc có số lượng tồn kho nhỏ hơn X	Nhân viên	Thêm thuốc có số lượng tồn kho nhỏ hơn X	Quản trị viên	Phantom Read Nhân viên tìm thuốc có số lượng tồn kho nhỏ hơn X thì admin thêm thuốc mới cũng thỏa điều kiện đó.
4	Chỉnh sửa số lượng tồn kho của thuốc X	Quản trị viên	Cập nhật toa thuốc có liên quan tới thuốc X	Nha sĩ	Lost Update Admin cập nhật số lượng tồn kho của thuốc X nhưng chưa hoàn thành update thì nha sĩ cập nhật số lượng của thuốc X trong một toa thuốc bất kì.
5	Cập nhật thông tin tài khoản của mình	Nhân viên	Chỉnh sửa thông tin tài khoản	Quản trị viên	Conversion Deadlock Nhân viên cập nhật thông tin tài khoản của mình nhưng chưa hoàn thành thì admin cập nhật thông tin của nhân viên đó.
6	Cập nhật toa thuốc X	Nha sĩ	Xóa toa thuốc X	Nha sĩ	Cycle Deadlock Nha sĩ trong lúc cập nhật thông tin toa thuốc X thì lại thực hiện xóa toa thuốc này.

V. Thiết kế giao diện chức năng


1. Đăng nhập




Login

— □ ×

LOGIN

 |

 Password

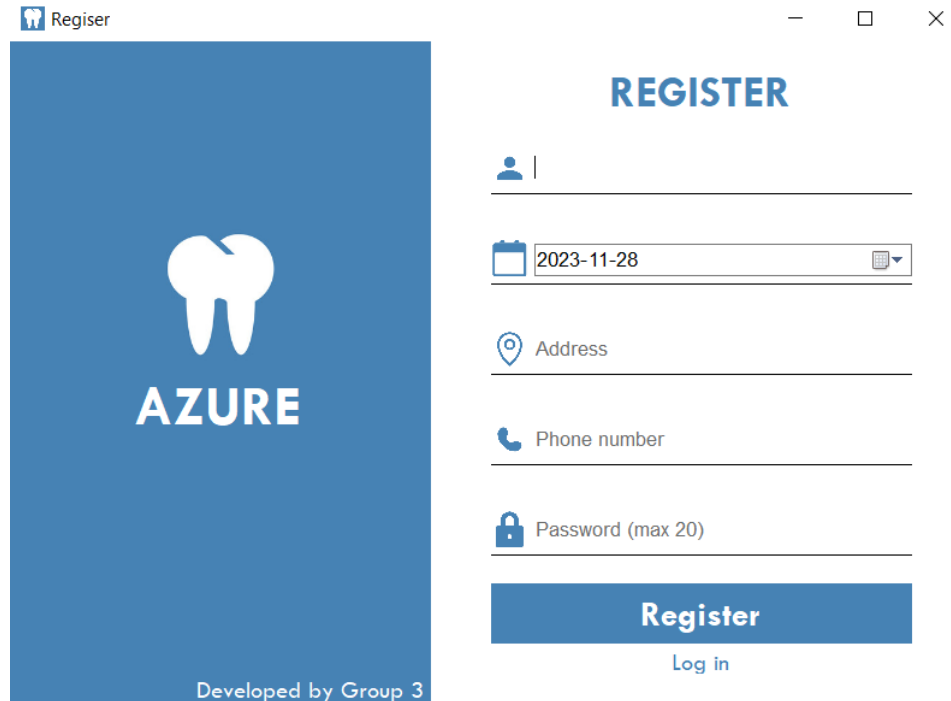
Log In

[Register](#)

Developed by Group 3

2. Phân hệ Khách hàng

2.1. Đăng ký



The screenshot shows a web application window titled "Regiser" (sic) with standard window controls. The left sidebar is blue with a white tooth icon and the text "AZURE" and "Developed by Group 3". The main content area is titled "REGISTER" and contains the following fields:


- A user selection field with a person icon and a dropdown arrow.
- A date field with a calendar icon, showing "2023-11-28", and a dropdown arrow.
- An "Address" field with a location pin icon.
- A "Phone number" field with a phone icon.
- A "Password (max 20)" field with a lock icon.

At the bottom of the form are two buttons: a blue "Register" button and a "Log in" link.

2.2. Cập nhật thông tin

Customer Dashboard

Customer ID: KH005



Bùi Hữu Quỳnh

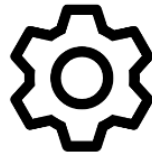
Account

Appointment

Medical record

Sign out

Developed by Group 3



ID

KH005

Name

Bùi Hữu Quỳnh

Phone number

90865834129

Date of birth

1998-12-10

Password

•••

Address

408, Nguyễn Thị Minh Khai, phường 3, quận 1, Thành phố Hồ Chí Minh

Update

2.3. Đặt lịch hẹn



Make an appointment



Make an appointment

Dentist

Date

Time (8h - 17h)

2.4. Cập nhật lịch hẹn

Customer Dashboard
 Customer ID: KH005

Bùi Hữu Quỳnh

Account

Appointment

Medical record

Sign out

Developed by Group 3

Appointment

Time (8h - 17h)

16:24

Date

2023-12-01

Employee ID

NV001

Appointment ID

LH027

Dentist

Vũ Anh Mai

Make an appointment

Update


Refresh

	MALICHHEN	NGAY	GIO	MAKHACHHANG	MANHASI	MANVDATLICH
▶	LH027	12/1/2023	984	KH005	NS011	NV001

2.5. Xem hồ sơ bệnh án

Customer Dashboard

Customer ID: KH005



Bùi Hữu Quỳnh


Account

Appointment

Medical record

Sign out

Developed by Group 3



Dentist ID:
NS013

Name:
Phan An Phúc

Customer ID:
KH005

Name:
Bùi Hữu Quỳnh

Phone:
90865834129


Date of birth:
1998-12-10

Address:
408, Nguyễn Thị Minh Khai, phường 3, quận 1, Thành phố Hồ Chí Minh

3. Phân hệ Nha sĩ

3.1. Cập nhật thông tin

Dentist ID: NS001


Nguyễn Hoàng Anh

Account

Appointment


Patient record

Medical certificate

Prescription

Services

Developed by Group 3



ID

NS001

Name

Nguyễn Hoàng Anh

Phone number

70275365445

Date of birth

1995-01-29

Password

•••

Address

294, Lý Thái Tổ, phường 1, quận 3, Thành phố Hồ Chí Minh

Update

3.2. Thêm và cập nhật hồ sơ bệnh án

Dentist ID: NS001


Nguyễn Hoàng Anh

Account

Appointment

Patient record

Medical certificate

Prescription

Services

Developed by Group 3

Customer Medical Record

Record ID
HS011

Customer ID
Nguyễn Hoàng Quỳnh


Dentist
Nguyễn Hoàng Anh

Add Delete Refresh

MAHSBA	MAKHACHHANG	MANHASI
▶ HS011	KH019	NS001

3.3. Thêm và cập nhật lịch hẹn cá nhân

Dentist ID: NS022


Nguyễn Anh Linh

Account

Appointment

Patient record

Medical certificate

Prescription

Services

Developed by Group 3

Appointment

Time (8h - 17h)

10:23

Date

1983-11-24

Employee ID

Employee ID

Appointment ID

LH008

Customer

Hoàng Gia Tiến

Make an appointment

Update

Delete

Refresh

	MALICHHEN	NGAY	GIO	MAKHACHHANG	MANHASI	MANVDATLICH
▶	LH008	11/24/1983	623	KH012	NS022	
	LH013	3/26/1986	752	KH028	NS022	NV003

3.4. Thêm và cập nhật giấy khám bệnh

Dentist ID: NS022

Nguyễn Anh Linh

Account

Appointment

Patient record

Medical certificate

Prescription

Services

Developed by Group 3

Medical Certificate

Certificate ID
GK001

Date of examination
1983-10-10

Customer Record ID
HS001

Add Refresh

	MAHSBA	MAGIAYKHAMBENH	NGAYKHAM	TONGTIENTHUOC	TONGTIENDICHVU
▶	HS001	GK001	10/10/1983	3000	0
	HS001	GK002	6/1/1996	0	0
	HS001	GK003	6/25/1996	54000	34000
	HS002	GK001	7/8/1997	75000	0
	HS002	GK002	11/15/1992	15000	200
	HS003	GK001	10/27/2001	13000	0
	HS003	GK002	5/26/1992	255000	200
	HS004	GK001	3/6/1999	149000	0
	HS005	GK001	4/28/2002	45000	0
	HS005	GK002	4/9/2000	318000	0
	HS006	GK001	10/6/1983	0	0

3.5. Thêm và cập nhật dịch vụ sử dụng

Dentist ID: NS022

Nguyễn Anh Linh

Account

Appointment

Patient record

Medical ceritificate

Prescription

Services

Dental Services

Service ID

Record ID

Certificate ID

DV003

HS002

GK002

Service Name

Price

TRÁM RĂNG SỬA

200

Note

Note

Add

Update

Delete

Refresh

	MADICHVU	TENDICHVU	GIATIEN	MAHSBA	MAGIAYKHAMBENH	GHICHU
	DV007	NIỀNG RĂNG INVI...	34000	HS001	GK003	
▶	DV003	TRÁM RĂNG SỬA	200	HS002	GK002	
	DV001	KHÁM TỔNG QUÁT	0	HS003	GK001	
	DV003	TRÁM RĂNG SỬA	200	HS003	GK002	
	DV001	KHÁM TỔNG QUÁT	0	HS005	GK001	
*						

Developed by Group 3

3.6. Thêm và cập nhật toa thuốc

Dentist ID: NS022

Nguyễn Anh Linh

Account

Appointment

Patient record

Medical certificate

Prescription

Services

Developed by Group 3

Customer Prescription

Customer record: Bùi Hữu Quỳnh

Certificate ID: GK001

Medication: Daflon

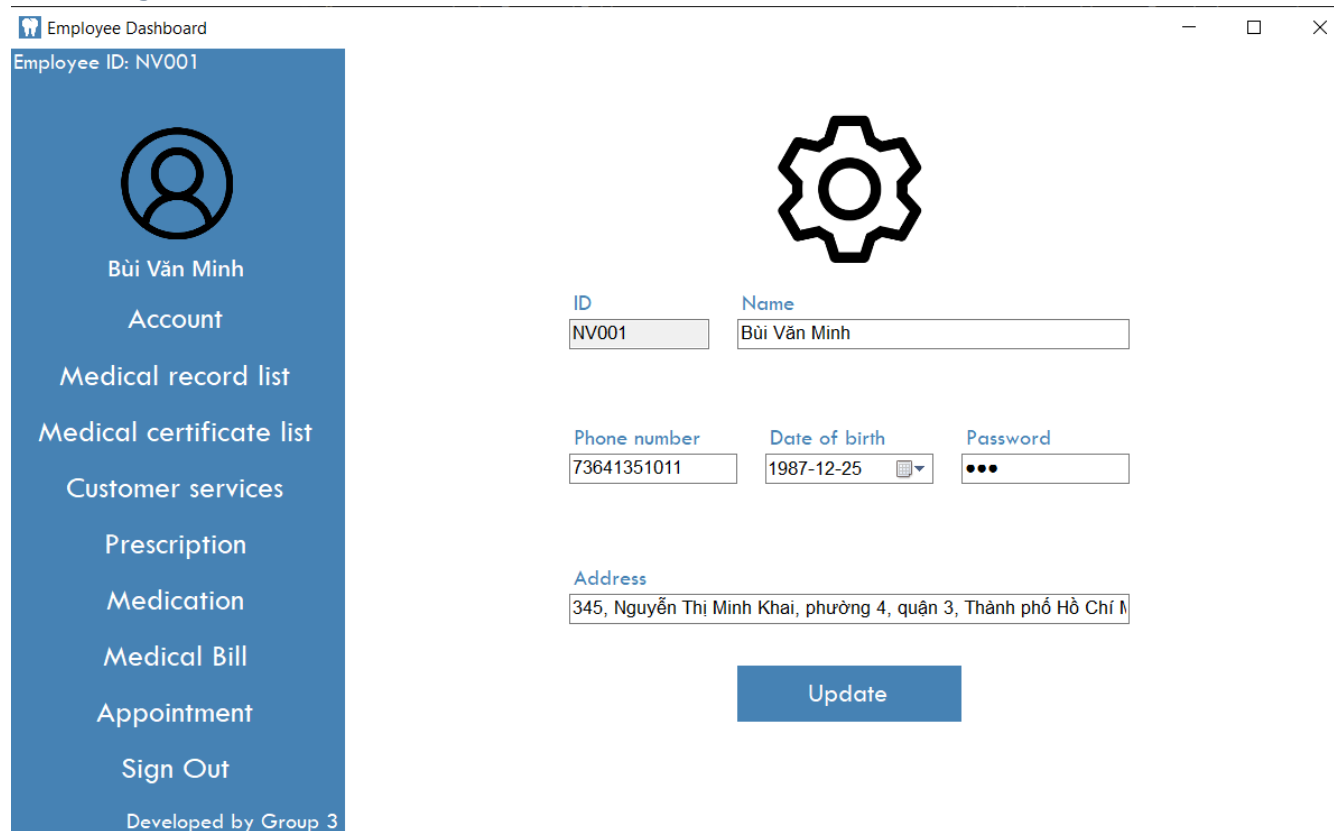
Quantity: 1

Add Update Delete Refresh

	MAHSBA	MAGIAYKHAMBENH	MATHUOC	SOLUONG
▶	HS001	GK001	TH009	1
	HS001	GK003	TH009	18
	HS002	GK001	TH014	5
	HS002	GK002	TH009	5
	HS003	GK001	TH018	1
	HS003	GK002	TH011	15
	HS003	GK002	TH015	14
	HS004	GK001	TH012	4
	HS004	GK001	TH019	9
	HS005	GK001	TH004	3
	HS005	GK002	TH004	20

4. Phân hệ Nhân viên

4.1. Cập nhật thông tin



The screenshot displays the 'Employee Dashboard' for user 'Employee ID: NV001'. On the left is a blue sidebar menu with icons and text for: Account, Medical record list, Medical certificate list, Customer services, Prescription, Medication, Medical Bill, Appointment, Sign Out, and 'Developed by Group 3'. The main content area features a large gear icon at the top. Below it are input fields for 'ID' (NV001) and 'Name' (Bui Văn Minh). Further down are fields for 'Phone number' (73641351011), 'Date of birth' (1987-12-25 with a calendar icon), and 'Password' (masked with dots). An 'Address' field contains the text '345, Nguyễn Thị Minh Khai, phường 4, quận 3, Thành phố Hồ Chí Minh'. A blue 'Update' button is positioned at the bottom center of the form area.

ID	Name	Phone number	Date of birth	Password	Address
NV001	Bui Văn Minh	73641351011	1987-12-25	•••	345, Nguyễn Thị Minh Khai, phường 4, quận 3, Thành phố Hồ Chí Minh

4.2. Đặt và cập nhật lịch hẹn

Employee ID: NV001

Bùi Văn Minh

Account

Medical record list

Medical certificate list

Customer services

Prescription

Medication

Medical Bill

Appointment

Developed by Group 3

Appointment

Appointment ID
LH001

Time (8h - 17h)
10:24

Date
2001-04-26

Dentist
Đỗ Gia Khánh

Customer
Lý Minh Tiến

Make an appointment
Update
Refresh


Enter appointment ID to ...
Search

	MALICHHEN	NGAY	GIO	MAKHACHHANG	MANHASI	MANVDATLICH
▶	LH001	4/26/2001	624	KH018	NS029	
	LH002	9/19/1986	899	KH019	NS026	
	LH003	2/3/2002	796	KH006	NS002	
	LH004	2/13/1986	989	KH010	NS009	
	LH005	1/1/2000	746	KH011	NS025	
	LH006	10/29/1996	1014	KH022	NS027	
	LH007	2/16/2002	687	KH006	NS012	
	LH008	11/24/1983	623	KH012	NS022	
	LH009	7/17/1998	675	KH016	NS014	
	LH010	11/14/1984	583	KH030	NS019	
	LH011	1/1/1996	863	KH011	NS039	NV003

4.3. Xem thuốc

Employee Dashboard

Employee ID: NV001



Bùi Văn Minh

Account

Medical record list

Medical certificate list

Customer services

Prescription

Medication

Medical Bill

Appointment

Sign Out

Developed by Group 3

Medication

Med ID: TH001 Emp ID: AD001 Unit: Hộp In stock: 69

Med Name: Augmentin 875mg/125mg Price: 10000

Expiration Date: 2025-12-22 Prescribe: Uống sau khi ăn


Refresh

Enter med ID to search

	MATHUOC	TENTHUOC	DONVITINH	CHIDINH	SOLUONGTON	NGAYHETHAN	GIATIEN	MANVQUANL
▶	TH001	Augmentin ...	Hộp	Uống sau kh...	69	12/22/2025	10000	AD001
	TH002	Angut 300mg	Viên	Uống sau kh...	46	4/25/2025	19000	AD010
	TH003	Co-Diovan 8...	Viên	Không dành ...	63	11/26/2025	14000	AD008
	TH004	Dopamine ...	Gói	Không dành ...	27	4/22/2024	15000	AD003
	TH005	Curam 875m...	Hộp	Uống sau kh...	24	5/21/2024	5000	AD007
	TH006	Dorocardyl ...	Hộp	Không dành ...	48	11/19/2025	8000	AD007
	TH007	Dưỡng huyế...	Viên	Uống trước ...	62	10/20/2024	18000	AD002
	TH008	Enterogermi...	Gói	Uống sau kh...	99	9/21/2025	19000	AD010
	TH009	Daflon	Viên	Không dành ...	75	4/20/2025	3000	AD006
	TH010	Bilobil forte ...	Hộp	Không dành ...	71	11/5/2024	14000	AD009
	TH011	Bupivacaine ...	Viên	Không dành ...	17	1/18/2025	3000	AD009

4.4. Lập và cập nhật hóa đơn

Employee ID: NV001



Bùi Văn Minh
Account

Medical record list

Medical certificate list

Customer services

Prescription

Medication

Medical Bill

Appointment

Developed by Group 3

Medical Bill

Bill ID: HD001

Customer Certificate: GK002

Total: 255200

Date: 2022-07-14

Customer Record: HS003


Add Refresh

Enter bill ID to search Search

	MAHOADON	NGAYLAP	TONGTIEN	MAHSBA	MAGIAYKHAMBENH	MANVLAP
▶	HD001	7/14/2022	255200	HS003	GK002	NV006
	HD002	5/26/2022	75000	HS002	GK001	NV004
	HD003	1/12/2022	13000	HS003	GK001	NV002
	HD004	11/11/2022	318000	HS005	GK002	NV007
	HD005	3/19/2022	45000	HS005	GK001	NV004
	HD006	4/22/2022	45000	HS005	GK001	NV001
	HD007	4/15/2022	0	HS001	GK002	NV006
	HD008	8/24/2022	75000	HS002	GK001	NV007
	HD009	8/20/2022	318000	HS005	GK002	NV006
	HD010	2/19/2022	149000	HS004	GK001	NV006
	HD011	7/29/2022	75000	HS002	GK001	NV005

4.5. Xem hồ sơ bệnh án

Employee ID: NV001



Bùi Văn Minh

Account

Medical record list

Medical certificate list

Customer services

Prescription

Medication

Medical Bill

Appointment

Developed by Group 3

Customer Medical Record

Record ID

Customer ID

Dentist ID


Refresh

Search

MAHSBA	MAKHACHHANG	MANHASI
▶ HS001	KH005	NS013
HS002	KH004	NS024
HS003	KH018	NS028
HS004	KH016	NS018
HS005	KH012	NS010
HS006	KH026	NS030
HS007	KH023	NS003
HS008	KH007	NS025
HS009	KH003	NS020
HS010	KH025	NS009
HS011	KH019	NS001

4.6. Xem giấy khám bệnh

Employee ID: NV001



Bùi Văn Minh

Account

Medical record list

Medical certificate list

Customer services

Prescription

Medication

Medical Bill

Appointment

Developed by Group 3

Medical Certificate

Certificate ID

GK001

Customer Record ID

HS001

Date of examination

1983-10-10

Med total

3000

Services total

0


Refresh

Enter certificate ID to se... Search

	MAHSBA	MAGIAYKHAMBENH	NGAYKHAM	TONGTIENTHUOC	TONGTIENDICHVU
▶	HS001	GK001	10/10/1983	3000	0
	HS001	GK002	6/1/1996	0	0
	HS001	GK003	6/25/1996	54000	34000
	HS002	GK001	7/8/1997	75000	0
	HS002	GK002	11/15/1992	15000	200
	HS003	GK001	10/27/2001	13000	0
	HS003	GK002	5/26/1992	255000	200
	HS004	GK001	3/6/1999	149000	0
	HS005	GK001	4/28/2002	45000	0
	HS005	GK002	4/9/2000	318000	0
	HS006	GK001	10/6/1983	0	0

4.7. Xem dịch vụ sử dụng

Employee ID: NV001



Bùi Văn Minh

Account

Medical record list

Medical certificate list

Customer services

Prescription

Medication

Medical Bill

Appointment

Developed by Group 3

Dental Services

Service ID

DV007

Price

34000

Service Name

NIỀNG RĂNG INVISALIGN ĐƠN GIẢN

Refresh

Enter service ID to search

Search

MADICHVU	TENDICHVU	GIATIEN
DV007	NIỀNG RĂNG INVISALIGN ĐƠN GIẢN	34000
DV003	TRÁM RĂNG SỮA	200
DV001	KHÁM TỔNG QUÁT	0
DV003	TRÁM RĂNG SỮA	200
DV001	KHÁM TỔNG QUÁT	0

4.8. Xem toa thuốc

Employee ID: NV001

Bùi Văn Minh

Account

Medical record list

Medical certificate list

Customer services

Prescription

Medication

Medical Bill

Appointment

Developed by Group 3

Customer Prescription

Customer Record ID

HS001

Medical Certificate ID

GK001

Med ID

TH009

Quantity

1

Medication Name

Daflon

Refresh

Enter record ID to search

Search

	MATHUOC	TENTHUOC	MAHSBA	SOLUONG	MAGIAYKHAMBENH
▶	TH009	Daflon	HS001	1	GK001
	TH009	Daflon	HS001	18	GK003
	TH014	Dimedrol	HS002	5	GK001
	TH009	Daflon	HS002	5	GK002
	TH018	Dorithricin	HS003	1	GK001
	TH011	Bupivacaine WPW Spi...	HS003	15	GK002
	TH015	Adrenalin	HS003	14	GK002
	TH012	Atropin sulfat	HS004	4	GK001
	TH019	Aerius	HS004	9	GK001
	TH004	Dopamine Hydrochlo...	HS005	3	GK001
	TH004	Dopamine Hydrochlo...	HS005	20	GK002

5. Phân hệ Quản trị viên

5.1. Thêm thuốc



Add medication

— □ ×

Add Medication

Med Name

Unit

Expiration Date

2023-12-02



Instock

Instock

Price

Price

Employee

Trần Hoàng Anh


Prescription


Prescription

Add

Cancel

5.2. Cập nhật thuốc

 Admin Dashboard
Admin ID: AD001


 Trần Hoàng Anh
 Account
 Account Manager
 Medication
 Sign Out
 Developed by Group 3

Medication

Med ID

TH001

Emp ID

Trần Hoàng Anh

Unit

Hộp

In stock

69

Med Name

Augmentin 875mg/125mg

Price

10000

Expiration Date

2025-12-22

Prescription

Uống sau khi ăn

Add

Update


Delete

Search

Refresh

	MATHUOC	TENTHUOC	DONVITINH	CHIDINH	SOLUONGTON	NGAYHETHAN	GIATIEN	MANVQUANL
▶	TH001	Augmentin ...	Hộp	Uống sau kh...	69	12/22/2025	10000	AD001
	TH002	Angut 300mg	Viên	Uống sau kh...	46	4/25/2025	19000	AD010
	TH003	Co-Diovan 8...	Viên	Không dành ...	63	11/26/2025	14000	AD008
	TH004	Dopamine ...	Gói	Không dành ...	27	4/22/2024	15000	AD003
	TH005	Curam 875m...	Hộp	Uống sau kh...	24	5/21/2024	5000	AD007
	TH006	Dorocardyl ...	Hộp	Không dành ...	48	11/19/2025	8000	AD007
	TH007	Dưỡng huyế...	Viên	Uống trước ...	62	10/20/2024	18000	AD002
	TH008	Enterogermi...	Gói	Uống sau kh...	99	9/21/2025	19000	AD010
	TH009	Daflon	Viên	Không dành ...	75	4/20/2025	3000	AD006
	TH010	Bilobil forte ...	Hộp	Không dành ...	71	11/5/2024	14000	AD009
	TH011	Bupivacaine ...	Viên	Không dành ...	17	1/18/2025	3000	AD009
	TH012	Atropin sulfat	Hộp	Không dành ...	11	9/16/2025	8000	AD001
	TH013	Dorocardyl ...	Hộp	Uống trước ...	62	3/26/2025	15000	AD005

5.3. Thêm tài khoản nhân viên, nha sĩ


 Add an account

— □ ×

Add account

Name

Birthdate

2023-12-02 

Phone number

Phone number

Address

Address

Password

Password


Emp type


▼

Create

Cancel

5.4. Cập nhật tài khoản nhân viên, nha sĩ

 Admin Dashboard
Admin ID: AD001


Trần Hoàng Anh
Account
Account Manager
Medication
Sign Out
Developed by Group 3

Account Manager

Account ID

NS001

Name

Nguyễn Hoàng Anh

Birthday

1995-01-29

Phone number

70275365445

Employee type

Dentist

Address

294, Lý Thái Tổ, phường 1, quận 3, Thành phố Hồ Chí Minh

Create

Update

Delete

Refresh

	MANHANVIEN	HOTEN	NGAYSINH	DIACHI	SODT	LOAINHANVIEN
▶	NS001	Nguyễn Hoàng A...	1/29/1995	294, Lý Thái Tổ, p...	70275365445	1
	NS002	Hồ Văn Tường Mi...	9/4/1995	157, Lý Thái Tổ, p...	36421252076	1
	NS003	Đỗ Tấn Tiến	7/19/1981	238, Nguyễn Thị ...	00031140059	1
	NS004	Trần Văn Minh	5/9/2002	465, Võ Thị Sáu, p...	63941748189	1
	NS005	Trần Thiên Tường	12/27/1986	299, Nguyễn Thị ...	03640384262	1
	NS006	Hoàng Hoàng Khôi	7/9/1980	213, Nguyễn Thị ...	74280455393	1
	NS007	Hoàng Tấn Thư	8/11/2001	206, Nguyễn Thị ...	90368347745	1
	NS008	Phan Thiên Khánh	10/26/2000	404, Nguyễn Thị ...	00609618013	1
	NS009	Phan Hoàng Tiến	9/5/1991	269, Lê Hồng Pho...	42616816627	1
	NS010	Nguyễn Hoàng T...	5/7/1996	305, Lê Hồng Pho...	79025070349	1
	NS011	Vũ Anh Mai	7/27/1983	287, Võ Thị Sáu, p...	22463544176	1

CÀI ĐẶT TÌNH HUỐNG TRANH CHẤP

I. Sinh viên thực hiện: 21127149 – Huỳnh Minh Quang

1. Tình huống 1: Admin thêm thông tin thuốc nhưng xảy ra lỗi phải rollback. Trước đó, nhân viên vào xem thuốc.

ERR01: Dirty read T1 (User = Quản trị viên): Thêm thuốc. T2 (User = Nhân viên): Xem thuốc.			
USP_THUOC_INS	Khóa	USP_THUOC_SEARCH	Khóa
Input: Tên thuốc, đơn vị tính, chỉ định, số lượng tồn kho, ngày hết hạn, giá tiền, mã nhân viên quản lý, mã thuốc. Output: 0 nếu insert thành công, hoặc mã lỗi.		Input: Mã thuốc, tên thuốc, đơn vị tính, chỉ định, số lượng tồn kho, ngày hết hạn, giá tiền, mã nhân viên quản lý. (Tất cả Input đều không bắt buộc). Output: Những thuốc thỏa điều kiện tìm kiếm.	
SET TRANSACTION ISOLATION LEVEL READ COMMITTED		SET TRANSACTION ISOLATION LEVEL READ UNCOMMITTED	
BEGIN TRAN			
B1: Kiểm tra tính hợp lệ của đơn vị tính (viên, gói, hộp), số lượng tồn kho (≥ 0), giá tiền (≥ 0), mã nhân viên quản lý (AD###). IF @DONVITINH != N'VIÊN' AND @DONVITINH != N'GÓI' AND @DONVITINH != N'HỘP' BEGIN RAISERROR(N'INVALID DONVITINH, MUST BE EITHER VIÊN, HỘP, OR GÓI', 16, 1) ROLLBACK TRAN RETURN -1 END IF @SOLUONGTONKHO < 0 BEGIN RAISERROR('INVALID SOLUONGTONKHO', 16, 1)			

ROLLBACK TRAN RETURN -2 END IF @GIATIEN < 0 BEGIN RAISERROR('INVALID GIATIEN', 16, 1) ROLLBACK TRAN RETURN -3 END			
B2: Tìm mã thuốc được thêm gần đây nhất, và tạo mã thuốc mới. SELECT @MATHUOC = MATHUOC FROM THUOC WHERE MATHUOC = (SELECT MAX(MATHUOC) FROM THUOC) SET @MATHUOC = dbo.F_MAKE_ID('TH', @MATHUOC)	S(THUOC) //Xin khóa đọc trên bảng THUOC thành công.		
B3: Insert dữ liệu vào bảng THUOC. INSERT INTO THUOC VALUES (@MATHUOC, @TENTHUOC, @DONVITINH, @CHIDINH, @SOLUONGTONKHO, @NGAYHETHAN, @GIATIEN, @MANVQUANLY)	X(THUOC) //Xin khóa ghi trên bảng THUOC thành công.		
WAITFOR DELAY '00:00:10'			
		BEGIN TRAN	
		B1: Khai báo và khởi tạo Dynamic SQL. DECLARE @SQL NVARCHAR(MAX) SET @SQL = N'SELECT * FROM THUOC WHERE 1 = 1'	
		B2: Kiểm tra yêu cầu tìm kiếm trên từng thuộc tính và điều chỉnh Dynamic SQL tương ứng. IF @MATHUOC IS NOT NULL	

		<pre> SET @SQL += N' AND MATHUOC LIKE "%"+@MATHUOC+"%" IF @TENTHUOC IS NOT NULL SET @SQL += N' AND TENTHUOC LIKE "%"+@TENTHUOC+"%" IF @DONVITINH IS NOT NULL SET @SQL += N' AND DONVITINH LIKE "%"+@DONVITINH+"%" IF @CHIDINH IS NOT NULL SET @SQL += N' AND CHIDINH LIKE "%"+@CHIDINH+"%" IF @SOLUONGTONKHO IS NOT NULL SET @SQL += N' AND SOLUONGTONKHO <= @SOLUONGTONKHO' IF @NGAYHETHAN IS NOT NULL SET @SQL += N' AND NGAYHETHAN <= @NGAYHETHAN' IF @GIATIEN IS NOT NULL SET @SQL += N' AND GIATIEN <= GIATIEN' IF @MANVQUANLY IS NOT NULL SET @SQL += N' AND MANVQUANLY LIKE "%"+@MANVQUANLY+"%" </pre>	
		<p>B3: Thực thi Dynamic SQL tìm những thuốc thỏa điều kiện.</p> <pre> EXEC sp_executesql @SQL, N'@MATHUOC VARCHAR(5), @TENTHUOC NVARCHAR(25), @DONVITINH NVARCHAR(5), @CHIDINH NVARCHAR(50), @SOLUONGTONKHO INT, @NGAYHETHAN DATE, @GIATIEN INT, @MANVQUANLY VARCHAR(5)', @MATHUOC, @TENTHUOC, @DONVITINH, @CHIDINH, @SOLUONGTONKHO, </pre>	<p>S(THUOC) //Không cần xin khóa.</p>

		@NGAYHETHAN, @GIATIEN, @MANVQUANLY	
		COMMIT TRAN RETURN 0	
<p>B4: Kiểm tra nhân viên được giao quản lý có phải là admin.</p> <pre>IF LEFT(@MANVQUANLY, 2) != 'AD' BEGIN RAISERROR('NVQUANLY MUST BE AN ADMIN', 16, 1) ROLLBACK TRAN RETURN -4 END</pre> <p>//Do nhân viên không phải admin gây lỗi nên toàn bộ Transaction bị ROLLBACK, dẫn đến T2 bị Dirty Read.</p>			

II. Sinh viên thực hiện: 21127296 – Đặng Hà Huy

2. Tình huống 2: Nhân viên đang tìm thuốc bằng tên thì admin thay đổi tên của một thuốc bất kì có trong danh sách sẽ hiển thị.

ERR02: Unrepeatable Read T1 (User = Nhân viên): Tìm thuốc bằng tên. T2 (User = Quản trị viên): Thay đổi tên thuốc.			
USP_THUOC_SEARCH	Khóa	USP_THUOC_UPD	Khóa
<u>Input:</u> Mã thuốc, tên thuốc, đơn vị tính, chỉ định, số lượng tồn kho, ngày hết hạn, giá tiền, mã nhân viên quản lý. (Tất cả Input đều không bắt buộc). <u>Output:</u> Những thuốc thỏa điều kiện tìm kiếm.		<u>Input:</u> Mã thuốc, tên thuốc, đơn vị tính, chỉ định, số lượng tồn kho, ngày hết hạn, giá tiền, mã nhân viên quản lý, người update. <u>Output:</u> 0 nếu update thành công, hoặc mã lỗi.	
SET TRANSACTION ISOLATION LEVEL READ COMMITTED		SET TRANSACTION ISOLATION LEVEL READ COMMITTED	
BEGIN TRAN			

<p>B1: Khai báo và khởi tạo Dynamic SQL. DECLARE @SQL NVARCHAR(MAX) SET @SQL = N' FROM THUOC WHERE 1 = 1'</p>			
<p>B2: Kiểm tra yêu cầu tìm kiếm trên từng thuộc tính và điều chỉnh Dynamic SQL tương ứng. IF @MATHUOC IS NOT NULL SET @SQL += N' AND MATHUOC LIKE "% "+@MATHUOC+"% " IF @TENTHUOC IS NOT NULL SET @SQL += N' AND TENTHUOC LIKE "% "+@TENTHUOC+"% " IF @DONVITINH IS NOT NULL SET @SQL += N' AND DONVITINH LIKE "% "+@DONVITINH+"% " IF @CHIDINH IS NOT NULL SET @SQL += N' AND CHIDINH LIKE "% "+@CHIDINH+"% " IF @SOLUONGTONKHO IS NOT NULL SET @SQL += N' AND SOLUONGTONKHO <= @SOLUONGTONKHO' IF @NGAYHETHAN IS NOT NULL SET @SQL += N' AND NGAYHETHAN <= @NGAYHETHAN' IF @GIATIEN IS NOT NULL SET @SQL += N' AND GIATIEN <= GIATIEN' IF @MANVQUANLY IS NOT NULL SET @SQL += N' AND MANVQUANLY LIKE "% "+@MANVQUANLY+"% "'</p>			
<p>B3: Tạo thêm DSQL đếm số lượng thuốc thỏa điều kiện, và hoàn thiện DSQL ban đầu. DECLARE @SQLCHK NVARCHAR(MAX), @RC INT SET @SQLCHK = N'SELECT @RC = COUNT(*)' + @SQL</p>			

SET @SQL = N'SELECT *' + @SQL			
<p>B4: Thực thi DSQL đếm và kiểm tra số lượng thuốc trả về.</p> <pre>EXEC sp_executesql @SQLCHK, N'@MATHUOC VARCHAR(5), @TENTHUOC NVARCHAR(25), @DONVITINH NVARCHAR(5), @CHIDINH NVARCHAR(50), @SOLUONGTONKHO INT, @NGAYHETHAN DATE, @GIATIEN INT, @MANVQUANLY VARCHAR(5), @RC INT OUTPUT', @MATHUOC, @TENTHUOC, @DONVITINH, @CHIDINH, @SOLUONGTONKHO, @NGAYHETHAN, @GIATIEN, @MANVQUANLY, @RC OUTPUT</pre> <pre>IF @RC = 0 BEGIN RAISERROR('NO RESULT FOUND!', 16, 1) ROLLBACK TRAN RETURN -1 END</pre>	<p>S(THUOC) //Xin khóa đọc trên bảng THUOC thành công.</p>		
WAITFOR DELAY '00:00:10'			
		BEGIN TRAN	
		<p>B1: Kiểm tra tính hợp lệ của mã thuốc, các thuộc tính có thể update, và người update.</p> <pre>IF NOT EXISTS (SELECT * FROM THUOC WHERE MATHUOC = @MATHUOC) BEGIN RAISERROR('INVALID MATHUOC', 16, 1) ROLLBACK TRAN RETURN -1 END</pre>	<p>S(THUOC) //Xin khóa đọc trên bảng THUOC thành công.</p>

		<pre> IF @DONVITINH != N'VIÊN' AND @DONVITINH != N'GÓI' AND @DONVITINH != N'HỘP' BEGIN RAISERROR(N'INVALID DONVITINH, MUST BE EITHER VIÊN, HỘP, OR GÓI', 16, 1) ROLLBACK TRAN RETURN -2 END IF @SOLUONGTONKHO < 0 BEGIN RAISERROR('INVALID SOLUONGTONKHO', 16, 1) ROLLBACK TRAN RETURN -3 END IF @GIATIEN < 0 BEGIN RAISERROR('INVALID GIATIEN', 16, 1) ROLLBACK TRAN RETURN -4 END IF @TENTHUOC IS NULL OR @CHIDINH IS NULL OR @NGAYHETHAN IS NULL OR @NGUOIUPDATE IS NULL BEGIN RAISERROR('INVALID DATA', 16, 1) ROLLBACK TRAN RETURN -5 END IF @NGUOIUPDATE NOT LIKE 'AD%' OR @NGUOIUPDATE != @MANVQUANLY BEGIN RAISERROR('ONLY ADMIN MANAGING THIS THUOC CAN UPDATE IT', 16, 1) </pre>	
--	--	----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	--

		ROLLBACK TRAN RETURN -6 END	
		B2: Update các thuộc tính của thuốc với mã thuốc phù hợp. UPDATE THUOC SET TENTHUOC = @TENTHUOC, DONVITINH = @DONVITINH, CHIDINH = @CHIDINH, SOLUONGTONKHO = @SOLUONGTONKHO, NGAYHETHAN = @NGAYHETHAN, GIATIEN = @GIATIEN, MANVQUANLY = @MANVQUANLY WHERE MATHUOC = @MATHUOC	U,X(THUOC) //Xin khóa update và khóa ghi trên bảng THUOC thành công.
		COMMIT TRAN RETURN 0	
<p>B5: Thực thi DSQL tìm những thuốc thỏa điều kiện.</p> <p>EXEC sp_executesql @SQL, N'@MATHUOC VARCHAR(5), @TENTHUOC NVARCHAR(25), @DONVITINH NVARCHAR(5), @CHIDINH NVARCHAR(50), @SOLUONGTONKHO INT, @NGAYHETHAN DATE, @GIATIEN INT, @MANVQUANLY VARCHAR(5)', @MATHUOC, @TENTHUOC, @DONVITINH, @CHIDINH, @SOLUONGTONKHO, @NGAYHETHAN, @GIATIEN, @MANVQUANLY</p> <p>//Do B2 của T2 đã cập nhật một thuốc thỏa điều kiện tìm kiếm của T1 nên lần đọc ở B5 sẽ có kết quả khác so với B4, dẫn đến T1 bị Unrepeatable Read.</p> <p>//Ví dụ: T1 tìm thuốc tên <i>Augmentin</i>, ở B4 trả về 1 dòng thỏa. Tuy nhiên, T2 cập nhật thuốc này sang</p>	<p>S(THUOC) //Xin khóa đọc trên bảng THUOC thành công.</p>		

tên ABC ở B2. Cuối cùng, khi T1 thực hiện B5 không thu được kết quả nào.			
COMMIT TRAN RETURN 0			

III. Sinh viên thực hiện: 21127135 – Diệp Hữu Phúc

3. Tình huống 3: Nhân viên tìm thuốc có số lượng tồn kho nhỏ hơn X thì admin thêm thuốc mới cũng thỏa điều kiện đó.

ERR03: Phantom Read T1 (User = Nhân viên): Tìm thuốc có số lượng tồn kho nhỏ hơn X. T2 (User = Quản trị viên): Thêm thuốc có số lượng tồn kho nhỏ hơn X.			
USP_THUOC_SEARCH	Khóa	USP_THUOC_INS	Khóa
Input: Mã thuốc, tên thuốc, đơn vị tính, chỉ định, số lượng tồn kho, ngày hết hạn, giá tiền, mã nhân viên quản lý. (Tất cả Input đều không bắt buộc). Output: Những thuốc thỏa điều kiện tìm kiếm.		Input: Tên thuốc, đơn vị tính, chỉ định, số lượng tồn kho, ngày hết hạn, giá tiền, mã nhân viên quản lý, mã thuốc. Output: 0 nếu insert thành công, hoặc mã lỗi.	
SET TRANSACTION ISOLATION LEVEL READ COMMITTED		SET TRANSACTION ISOLATION LEVEL READ COMMITTED	
BEGIN TRAN			
B1: Khai báo và khởi tạo Dynamic SQL. DECLARE @SQL NVARCHAR (MAX) SET @SQL = N' FROM THUOC WHERE 1 = 1'			
B2: Kiểm tra yêu cầu tìm kiếm trên từng thuộc tính và điều chỉnh Dynamic SQL tương ứng. IF @MATHUOC IS NOT NULL SET @SQL += N' AND MATHUOC LIKE "%"+@MATHUOC+"%" IF @TENTHUOC IS NOT NULL SET @SQL += N' AND TENTHUOC LIKE "%"+@TENTHUOC+"%"			

<pre> IF @DONVITINH IS NOT NULL SET @SQL += N' AND DONVITINH LIKE "%"+@DONVITINH+"%" IF @CHIDINH IS NOT NULL SET @SQL += N' AND CHIDINH LIKE "%"+@CHIDINH+"%" IF @SOLUONGTONKHO IS NOT NULL SET @SQL += N' AND SOLUONGTONKHO <= @SOLUONGTONKHO' IF @NGAYHETHAN IS NOT NULL SET @SQL += N' AND NGAYHETHAN <= @NGAYHETHAN' IF @GIATIEN IS NOT NULL SET @SQL += N' AND GIATIEN <= GIATIEN' IF @MANVQUANLY IS NOT NULL SET @SQL += N' AND MANVQUANLY LIKE "%"+@MANVQUANLY+"%" </pre>			
<p>B3: Tạo thêm DSQL đếm số lượng thuốc thỏa điều kiện, và hoàn thiện DSQL ban đầu.</p> <pre> DECLARE @SQLCHK NVARCHAR(MAX), @RC INT SET @SQLCHK = N'SELECT @RC = COUNT(*)' + @SQL SET @SQL = N'SELECT *' + @SQL </pre>			
<p>B4: Thực thi DSQL đếm và kiểm tra số lượng thuốc trả về.</p> <pre> EXEC sp_executesql @SQLCHK, N'@MATHUOC VARCHAR(5), @TENTHUOC NVARCHAR(25), @DONVITINH NVARCHAR(5), @CHIDINH NVARCHAR(50), @SOLUONGTONKHO INT, @NGAYHETHAN DATE, @GIATIEN INT, @MANVQUANLY VARCHAR(5), @RC INT OUTPUT', @MATHUOC, @TENTHUOC, @DONVITINH, @CHIDINH, @SOLUONGTONKHO, </pre>	<p>S(THUOC) //Xin khóa đọc trên bảng THUOC thành công.</p>		

@NGAYHETHAN, @GIATIEN, @MANVQUANLY, @RC OUTPUT IF @RC = 0 BEGIN RAISERROR('NO RESULT FOUND!', 16, 1) ROLLBACK TRAN RETURN -1 END			
WAITFOR DELAY '00:00:10'			
		BEGIN TRAN	
		B1: Kiểm tra tính hợp lệ của đơn vị tính (viên, gói, hộp), số lượng tồn kho (≥ 0), giá tiền (≥ 0), mã nhân viên quản lý (AD###). IF @DONVITINH != N'VIÊN' AND @DONVITINH != N'GÓI' AND @DONVITINH != N'HỘP' BEGIN RAISERROR(N'INVALID DONVITINH, MUST BE EITHER VIÊN, HỘP, OR GÓI', 16, 1) ROLLBACK TRAN RETURN -1 END IF @SOLUONGTONKHO < 0 BEGIN RAISERROR('INVALID SOLUONGTONKHO', 16, 1) ROLLBACK TRAN RETURN -2 END IF @GIATIEN < 0 BEGIN RAISERROR('INVALID GIATIEN', 16, 1) ROLLBACK TRAN RETURN -3 END	

		<pre> IF LEFT(@MANVQUANLY, 2) != 'AD' BEGIN RAISERROR('NVQUANLY MUST BE AN ADMIN', 16, 1) ROLLBACK TRAN RETURN -4 END </pre>	
		<p>B2: Tìm mã thuốc được thêm gần đây nhất, và tạo mã thuốc mới.</p> <pre> SELECT @MATHUOC = MATHUOC FROM THUOC WHERE MATHUOC = (SELECT MAX(MATHUOC) FROM THUOC) SET @MATHUOC = dbo.F_MAKE_ID('TH', @mATHUOC) </pre>	<p>S(THUOC) //Xin khoá đọc trên bảng THUOC thành công.</p>
		<p>B3: Insert dữ liệu vào bảng THUOC.</p> <pre> INSERT INTO THUOC VALUES (@MATHUOC, @TENTHUOC, @DONVITINH, @CHIDINH, @SOLUONGTONKHO, @NGAYHETHAN, @GIATIEN, @MANVQUANLY) </pre>	<p>X(THUOC) //Xin khóa ghi trên bảng THUOC thành công.</p>
		<pre> COMMIT TRAN RETURN 0 </pre>	
<p>B5: Thực thi DSQL tìm những thuốc thỏa điều kiện.</p> <pre> EXEC sp_executesql @SQL, N'@MATHUOC VARCHAR(5), @TENTHUOC NVARCHAR(25), @DONVITINH NVARCHAR(5), @CHIDINH NVARCHAR(50), @SOLUONGTONKHO INT, @NGAYHETHAN DATE, @GIATIEN INT, @MANVQUANLY VARCHAR(5)', </pre>	<p>S(THUOC) //Xin khóa đọc trên bảng THUOC thành công.</p>		

<p>@MATHUOC, @TENTHUOC, @DONVITINH, @CHIDINH, @SOLUONGTONKHO, @NGAYHETHAN, @GIATIEN, @MANVQUANLY</p> <p>//Do B3 của T2 đã thêm một thuốc thỏa điều kiện tìm kiếm của T1 nên lần đọc ở B5 sẽ có nhiều kết quả hơn so với B4, dẫn đến T1 bị Phantom Read.</p> <p>//Ví dụ: T1 tìm mọi thuốc có số lượng tồn kho (sltk) <= 30, ở B4 trả về 6 dòng thỏa. Tuy nhiên, T2 thêm thuốc ABC có sltk = 10 ở B3. Cuối cùng, khi T1 thực hiện B5 thu được 7 kết quả.</p>			
<p>COMMIT TRAN</p> <p>RETURN 0</p>			

4. Tình huống 4: Admin cập nhật số lượng tồn kho của thuốc X nhưng chưa hoàn thành update thì nha sĩ cập nhật số lượng của thuốc X trong một toa thuốc bất kì.

<p>ERR04: Lost Update</p> <p>T1 (User = Quản trị viên): Chính sửa số lượng tồn kho của thuốc X.</p> <p>T2 (User = Nha sĩ): Cập nhật toa thuốc có liên quan tới thuốc X.</p>			
<p>USP_THUOC_UPD</p> <p>Input: Mã thuốc, tên thuốc, đơn vị tính, chỉ định, số lượng tồn kho, ngày hết hạn, giá tiền, mã nhân viên quản lý, người update.</p> <p>Output: 0 nếu update thành công, hoặc mã lỗi.</p> <p>SET TRANSACTION ISOLATION LEVEL READ COMMITTED</p> <p>BEGIN TRAN</p> <p>B1: Kiểm tra tính hợp lệ của mã thuốc, các thuộc tính có thể update, và người update.</p>	<p>Khóa</p>	<p>USP_TOATHUOC_UPD</p> <p>Input: Mã hồ sơ bệnh án, mã giấy khám bệnh, mã thuốc, số lượng.</p> <p>Output: 0 nếu update thành công, hoặc mã lỗi.</p> <p>SET TRANSACTION ISOLATION LEVEL READ COMMITTED</p>	<p>Khóa</p>
	S(THUOC)		

<pre> IF NOT EXISTS (SELECT * FROM THUOC WHERE MATHUOC = @MATHUOC) BEGIN RAISERROR('INVALID MATHUOC', 16, 1) ROLLBACK TRAN RETURN -1 END IF @DONVITINH != N'VIÊN' AND @DONVITINH != N'GÓI' AND @DONVITINH != N'HỘP' BEGIN RAISERROR(N'INVALID DONVITINH, MUST BE EITHER VIÊN, HỘP, OR GÓI', 16, 1) ROLLBACK TRAN RETURN -2 END IF @SOLUONGTONKHO < 0 BEGIN RAISERROR('INVALID SOLUONGTONKHO', 16, 1) ROLLBACK TRAN RETURN -3 END IF @GIATIEN < 0 BEGIN RAISERROR('INVALID GIATIEN', 16, 1) ROLLBACK TRAN RETURN -4 END IF @TENTHUOC IS NULL OR @CHIDINH IS NULL OR @NGAYHETHAN IS NULL OR @NGUOIUPDATE IS NULL BEGIN RAISERROR('INVALID DATA', 16, 1) ROLLBACK TRAN RETURN -5 END </pre>	<pre> //Xin khóa đọc trên bảng THUOC thành công. </pre>		
----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	---------------------------------------------------------	--	--

<pre>IF @NGUOIUPDATE NOT LIKE 'AD%' OR @NGUOIUPDATE != @MANVQUANLY BEGIN RAISERROR('ONLY ADMIN MANAGING THIS THUOC CAN UPDATE IT', 16, 1) ROLLBACK TRAN RETURN -6 END</pre>			
WAITFOR DELAY '00:00:10'			
		BEGIN TRAN	
		<p>B1: Kiểm tra sự tồn tại của toa thuốc.</p> <pre>IF NOT EXISTS (SELECT * FROM TOATHUOC WHERE MAHSBA = @MAHSBA AND MAGIAYKHAMBENH = @MAGIAYKHAMBENH AND MATHUOC = @MATHUOC) BEGIN RAISERROR('INVALID TOATHUOC', 16, 1) ROLLBACK TRAN RETURN -1 END</pre>	<p>S(TOA THUOC) //Xin khóa đọc trên bảng TOATHUOC thành công.</p>
		<p>B2: Kiểm tra tính hợp lệ của mã thuốc, và số lượng yêu cầu có vượt quá số lượng tồn kho.</p> <pre>IF NOT EXISTS (SELECT * FROM THUOC WHERE MATHUOC = @MATHUOC AND SOLUONGTONKHO >= @SOLUONG) BEGIN RAISERROR('INVALID MATHUOC, OR SOLUONGTONKHO IS INSUFFICIENT', 16, 1) ROLLBACK TRAN RETURN -2 END</pre>	<p>S(THUOC) //Xin khóa đọc trên bảng THUOC thành công.</p>
		<p>B3: Lấy số lượng hiện tại ở toa thuốc.</p> <pre>DECLARE @DIFFSOLUONG INT, @DIFFCOST INT</pre>	<p>S(TOA THUOC)</p>

		<code>SELECT @DIFFSOLUONG = SOLUONG FROM TOATHUOC WHERE MAHSBA = @MAHSBA AND MAGIAYKHAMBENH = @MAGIAYKHAMBENH AND MATHUOC = @MATHUOC</code>	//Xin khóa đọc trên bảng TOATHUOC thành công.
		<p>B4: Tính chênh lệch giữa số lượng hiện tại và số lượng muốn cập nhật. Tính chênh lệch giá tiền dựa vào chênh lệch số lượng.</p> <code>SET @DIFFSOLUONG = @SOLUONG - @DIFFSOLUONG</code> <code>SET @DIFFCOST = (SELECT GIATIEN FROM THUOC WHERE MATHUOC = @MATHUOC) * @DIFFSOLUONG</code>	<p>S(THUOC)</p> <p>//Xin khóa đọc trên bảng THUOC thành công.</p>
		<p>B5: Update số lượng cho toa thuốc tương ứng.</p> <code>UPDATE TOATHUOC SET SOLUONG = @SOLUONG WHERE MAHSBA = @MAHSBA AND MAGIAYKHAMBENH = @MAGIAYKHAMBENH AND MATHUOC = @MATHUOC</code>	<p>U,X(TOA THUOC)</p> <p>//Xin khóa update và khóa ghi trên bảng TOATHUOC thành công.</p>
		<p>B6: Update số lượng tồn kho dựa vào chênh lệch cho thuốc tương ứng.</p> <code>UPDATE THUOC SET SOLUONGTONKHO -= @DIFFSOLUONG WHERE MATHUOC = @MATHUOC</code>	<p>U,X(THUOC)</p> <p>//Xin khóa update và khóa ghi trên bảng THUOC thành công.</p>
		<p>B7: Update tổng tiền thuốc dựa vào chênh lệch cho giấy khám bệnh tương ứng.</p> <code>UPDATE GIAYKHAMBENH SET TONGTIENTHUOC += @DIFFCOST WHERE MAHSBA = @MAHSBA AND MAGIAYKHAMBENH = @MAGIAYKHAMBENH</code>	<p>U,X(GIAY KHAMBENH)</p> <p>//Xin khóa update và khóa ghi trên bảng GIAY KHAMBENH thành công.</p>

		COMMIT TRAN RETURN 0	
<p>B2: Update các thuộc tính của thuốc với mã thuốc phù hợp.</p> <p>UPDATE THUOC SET TENTHUOC = @TENTHUOC, DONVITINH = @DONVITINH, CHIDINH = @CHIDINH, SOLUONGTONKHO = @SOLUONGTONKHO, NGAYHETHAN = @NGAYHETHAN, GIATIEN = @GIATIEN, MANVQUANLY = @MANVQUANLY WHERE MATHUOC = @MATHUOC</p> <p>//Do B6 của T2 cập nhật số lượng tồn kho (sltk) của thuốc trong khi T1 chưa thực hiện cập nhật tại B2, nên khi B2 hoàn thành sẽ ghi đè lên thông tin ở B6 của T2, dẫn đến T2 bị Lost Update.</p> <p>//Ví dụ: T1 cập nhật sltk của thuốc TH009 từ 63 lên 100. Đồng thời, T2 thay đổi số lượng của toa thuốc TH009 (của giấy khám bệnh GK001 thuộc hồ sơ HS001) từ 13 lên 20. Theo đúng, kết quả sltk của TH009 phải là $100 - 7 = 93$. Tuy nhiên, do T2 bị Lost Update nên sltk vẫn đạt 100.</p>	<p>U,X THUOC)</p> <p>//Xin khóa update và khóa ghi trên bảng THUOC thành công.</p>		
COMMIT TRAN RETURN 0			

IV. Sinh viên thực hiện: 21127004 – Trần Nguyễn An Phong

5. Tình huống 5: Nhân viên muốn cập nhật thông tin tài khoản của mình nhưng chưa hoàn thành thì đồng thời cũng có một admin muốn cập nhật thông tin của nhân viên đó.

ERR05: Conversion Deadlock T1 (User = Nhân viên): Cập nhật thông tin tài khoản của mình. T2 (User = Quản trị viên): Chính sửa thông tin tài khoản của nhân viên trên.			
USP_NHANVIEN_UPD Input: Mã nhân viên, họ tên, địa chỉ, số điện thoại, loại nhân viên, mật khẩu. Output: 0 nếu update thành công, hoặc mã lỗi.	Khóa	USP_NHANVIEN_UPD Input: Mã nhân viên, họ tên, địa chỉ, số điện thoại, loại nhân viên, mật khẩu. Output: 0 nếu update thành công, hoặc mã lỗi.	Khóa
SET TRANSACTION ISOLATION LEVEL REPEATABLE READ		SET TRANSACTION ISOLATION LEVEL REPEATABLE READ	
BEGIN TRAN			
B1: Kiểm tra tính hợp lệ của mã nhân viên. IF NOT EXISTS (SELECT * FROM NHANVIEN WHERE MANHANVIEN = @MANHANVIEN) BEGIN RAISERROR('INVALID MANHANVIEN', 16, 1) ROLLBACK TRAN RETURN -1 END	S(NHANVIEN) //Xin khóa đọc trên bảng NHANVIEN thành công.		
B2: Kiểm tra tính hợp lệ của các thuộc tính có thể update. IF @LOAINHANVIEN != 0 AND @LOAINHANVIEN != 1 AND @LOAINHANVIEN != 2 BEGIN RAISERROR('INVALID LOAINHANVIEN', 16, 1) ROLLBACK TRAN RETURN -2 END			

<pre>IF @HOTEN IS NULL OR @DIACHI IS NULL OR @SODT IS NULL OR @MATKHAU IS NULL BEGIN RAISERROR('INVALID DATA', 16, 1) ROLLBACK TRAN RETURN -3 END</pre>			
WAITFOR DELAY '00:00:10'			
		BEGIN TRAN	
		<p>B1: Kiểm tra tính hợp lệ của mã nhân viên.</p> <pre>IF NOT EXISTS (SELECT * FROM NHANVIEN WHERE MANHANVIEN = @MANHANVIEN) BEGIN RAISERROR('INVALID MANHANVIEN', 16, 1) ROLLBACK TRAN RETURN -1 END</pre>	<p>S(NHANVIEN) //Xin khóa đọc trên bảng NHANVIEN thành công.</p>
		<p>B2: Kiểm tra tính hợp lệ của các thuộc tính có thể update.</p> <pre>IF @LOAINHANVIEN != 0 AND @LOAINHANVIEN != 1 AND @LOAINHANVIEN != 2 BEGIN RAISERROR('INVALID LOAINHANVIEN', 16, 1) ROLLBACK TRAN RETURN -2 END</pre> <pre>IF @HOTEN IS NULL OR @DIACHI IS NULL OR @SODT IS NULL OR @MATKHAU IS NULL BEGIN RAISERROR('INVALID DATA', 16, 1) ROLLBACK TRAN</pre>	

		RETURN -3 END	
		WAITFOR DELAY '00:00:10'	
B3: Update nhân viên với mã nhân viên phù hợp. UPDATE NHANVIEN SET HOTEN = @HOTEN, DIACHI = @DIACHI, SODT = @SODT, LOAINHANVIEN = @LOAINHANVIEN, MATKHAU = @MATKHAU WHERE MANHANVIEN = @MANHANVIEN	U,X(NHAN VIEN) //Xin khóa update và khóa ghi trên bảng NHANVIEN thất bại do T2 giữ khóa đọc đến cuối TRAN.		
CHỜ			
		B3: Update nhân viên với mã nhân viên phù hợp. UPDATE NHANVIEN SET HOTEN = @HOTEN, DIACHI = @DIACHI, SODT = @SODT, LOAINHANVIEN = @LOAINHANVIEN, MATKHAU = @MATKHAU WHERE MANHANVIEN = @MANHANVIEN	U,X(NHAN VIEN) //Xin khóa update và khóa ghi trên bảng NHANVIEN thất bại do T1 giữ khóa đọc đến cuối TRAN.
//Xảy ra Conversion Deadlock		CHỜ	

6. Tình huống 6: Nha sĩ trong lúc cập nhật thông tin toa thuốc X thì lại thực hiện xóa toa thuốc này.

ERR06: Cycle Deadlock

T1 (User = Nha sĩ): Cập nhật toa thuốc X.

T2 (User = Nha sĩ): Xóa toa thuốc X.

USP_TOATHUOC_UPD	Khóa	USP_TOATHUOC_DEL	Khóa
Input: Mã hồ sơ bệnh án, mã giấy khám bệnh, mã thuốc, số lượng. Output: 0 nếu update thành công, hoặc mã lỗi.		Input: Mã hồ sơ bệnh án, mã giấy khám bệnh, mã thuốc. Output: 0 nếu update thành công, hoặc mã lỗi.	
SET TRANSACTION ISOLATION LEVEL READ COMMITTED		SET TRANSACTION ISOLATION LEVEL READ COMMITTED	
BEGIN TRAN			
B1: Kiểm tra sự tồn tại của toa thuốc. IF NOT EXISTS (SELECT * FROM TOATHUOC WHERE MAHSBA = @MAHSBA AND MAGIAYKHAMBENH = @MAGIAYKHAMBENH AND MATHUOC = @MATHUOC) BEGIN RAISERROR('INVALID TOATHUOC', 16, 1) ROLLBACK TRAN RETURN -1 END	S(TOA THUOC) //Xin khóa đọc trên bảng TOATHUOC thành công.		
B2: Kiểm tra tính hợp lệ của mã thuốc, và số lượng yêu cầu có vượt quá số lượng tồn kho. IF NOT EXISTS (SELECT * FROM THUOC WHERE MATHUOC = @MATHUOC AND SOLUONGTONKHO >= @SOLUONG) BEGIN RAISERROR('INVALID MATHUOC, OR SOLUONGTONKHO IS INSUFFICIENT', 16, 1) ROLLBACK TRAN RETURN -2 END	S(THUOC) //Xin khóa đọc trên bảng THUOC thành công.		
B3: Lấy số lượng hiện tại ở toa thuốc. DECLARE @DIFFSOLUONG INT, @DIFFCOST INT SELECT @DIFFSOLUONG = SOLUONG FROM TOATHUOC WHERE MAHSBA = @MAHSBA	S(TOA THUOC) //Xin khóa đọc trên bảng TOATHUOC thành công.		

AND MAGIAYKHAMBENH = @MAGIAYKHAMBENH AND MATHUOC = @MATHUOC			
B4: Tính chênh lệch giữa số lượng hiện tại và số lượng muốn cập nhật. Tính chênh lệch giá tiền dựa vào chênh lệch số lượng. SET @DIFFSOLUONG = @SOLUONG - @DIFFSOLUONG SET @DIFFCOST = (SELECT GIATIEN FROM THUOC WHERE MATHUOC = @MATHUOC) * @DIFFSOLUONG	S(THUOC) //Xin khóa đọc trên bảng THUOC thành công.		
WAITFOR DELAY '00:00:05'			
		BEGIN TRAN	
		B1: Kiểm tra sự tồn tại của toa thuốc. IF NOT EXISTS (SELECT * FROM TOATHUOC WHERE MAHSBA = @MAHSBA AND MAGIAYKHAMBENH = @MAGIAYKHAMBENH AND MATHUOC = @MATHUOC) BEGIN RAISERROR('INVALID TOATHUOC', 16, 1) ROLLBACK TRAN RETURN -1 END	S(TOA THUOC) //Xin khóa đọc trên bảng TOATHUOC thành công.
		B2: Lấy số lượng hiện tại ở toa thuốc. DECLARE @SOLUONG INT, @COST INT SELECT @SOLUONG = SOLUONG FROM TOATHUOC WHERE MAHSBA = @MAHSBA AND MAGIAYKHAMBENH = @MAGIAYKHAMBENH AND MATHUOC = @MATHUOC	S(TOA THUOC) //Xin khóa đọc trên bảng TOATHUOC thành công.
		B3: Tính giá tiền hiện tại của toa thuốc dựa vào số lượng. SET @COST = (SELECT GIATIEN FROM THUOC WHERE MATHUOC = @MATHUOC)	S(THUOC) // Xin khóa đọc trên bảng

		* @SOLUONG	THUOC thành công.
		WAITFOR DELAY '00:00:05'	
B5: Update số lượng cho toa thuốc tương ứng. UPDATE TOATHUOC SET SOLUONG = @SOLUONG WHERE MAHSBA = @MAHSBA AND MAGIAYKHAMBENH = @MAGIAYKHAMBENH AND MATHUOC = @MATHUOC	U,X(TOA THUOC) //Xin khóa update và khóa ghi trên bảng TOATHUOC thành công.		
WAITFOR DELAY '00:00:05'			
		B4: Update giảm tổng tiền thuốc, dựa vào giá toa thuốc, cho giấy khám bệnh tương ứng. UPDATE GIAYKHAMBENH SET TONGTIENTHUOC -= @COST WHERE MAHSBA = @MAHSBA AND MAGIAYKHAMBENH = @MAGIAYKHAMBENH	U,X(GIAY KHAMBENH) //Xin khóa update và khóa ghi trên bảng GIAY KHAMBENH thành công.
		WAITFOR DELAY '00:00:05'	
B6: Update tổng tiền thuốc dựa vào chênh lệch cho giấy khám bệnh tương ứng. UPDATE GIAYKHAMBENH SET TONGTIENTHUOC += @DIFFCOST WHERE MAHSBA = @MAHSBA AND MAGIAYKHAMBENH = @MAGIAYKHAMBENH	U,X(GIAY KHAMBENH) //Xin khóa update và khóa ghi trên bảng GIAY KHAMBENH thất bại do T2 đang giữ khóa.		
CHỜ			
		B5: Xóa toa thuốc tương ứng.	U,X(TOA THUOC)

		DELETE FROM TOATHUOC WHERE MAHSBA = @MAHSBA AND MAGIAYKHAMBENH = @MAGIAYKHAMBENH AND MATHUOC = @MATHUOC	//Xin khóa update và khóa ghi trên bảng TOATHUOC thất bại do T1 đang giữ khóa.
//Xảy ra Cycle Deadlock		CHỜ	

XỬ LÝ TÌNH HUỐNG TRANH CHẤP

I. Sinh viên thực hiện: 21127149 – Huỳnh Minh Quang

- Tình huống 1: Admin thêm thông tin thuốc nhưng xảy ra lỗi phải rollback. Trước đó, nhân viên vào xem thuốc.

Lỗi Dirty Read xảy ra do mức cô lập của T2 là **READ UNCOMMITTED**, cho phép đọc trên bảng đã có khóa ghi. Do đó, để khắc phục cần nâng mức cô lập của T2 lên **READ COMMITTED**.

ERR01: Dirty read T1 (User = Quản trị viên): Thêm thuốc. T2 (User = Nhân viên): Xem thuốc.			
USP_THUOC_INS	Khóa	USP_THUOC_SEARCH	Khóa
Input: Tên thuốc, đơn vị tính, chỉ định, số lượng tồn kho, ngày hết hạn, giá tiền, mã nhân viên quản lý, mã thuốc. Output: 0 nếu insert thành công, hoặc mã lỗi.		Input: Mã thuốc, tên thuốc, đơn vị tính, chỉ định, số lượng tồn kho, ngày hết hạn, giá tiền, mã nhân viên quản lý. (Tất cả Input đều không bắt buộc). Output: Những thuốc thỏa điều kiện tìm kiếm.	
SET TRANSACTION ISOLATION LEVEL READ COMMITTED		SET TRANSACTION ISOLATION LEVEL READ COMMITTED	
BEGIN TRAN			

<p>B1: Kiểm tra tính hợp lệ của đơn vị tính (viên, gói, hộp), số lượng tồn kho (≥ 0), giá tiền (≥ 0), mã nhân viên quản lý (AD###).</p> <pre> IF @DONVITINH != N'VIÊN' AND @DONVITINH != N'GÓI' AND @DONVITINH != N'HỘP' BEGIN RAISERROR(N'INVALID DONVITINH, MUST BE EITHER VIÊN, HỘP, OR GÓI', 16, 1) ROLLBACK TRAN RETURN -1 END IF @SOLUONGTONKHO < 0 BEGIN RAISERROR('INVALID SOLUONGTONKHO', 16, 1) ROLLBACK TRAN RETURN -2 END IF @GIATIEN < 0 BEGIN RAISERROR('INVALID GIATIEN', 16, 1) ROLLBACK TRAN RETURN -3 END IF LEFT(@MANVQUANLY, 2) != 'AD' BEGIN RAISERROR('NVQUANLY MUST BE AN ADMIN', 16, 1) ROLLBACK TRAN RETURN -4 END </pre>			
<p>B2: Tìm mã thuốc được thêm gần đây nhất, và tạo mã thuốc mới.</p> <pre> SELECT @MATHUOC = MATHUOC FROM THUOC </pre>	<p>S(THUOC) //Xin khóa đọc trên bảng</p>		

WHERE MATHUOC = (SELECT MAX(MATHUOC) FROM THUOC) SET @MATHUOC = dbo.F_MAKE_ID('TH', @MATHUOC)	THUOC thành công.		
B3: Insert dữ liệu vào bảng THUOC. INSERT INTO THUOC VALUES (@MATHUOC, @TENTHUOC, @DONVITINH, @CHIDINH, @SOLUONGTONKHO, @NGAYHETHAN, @GIATIEN, @MANVQUANLY)	X(THUOC) //Xin khóa ghi trên bảng THUOC thành công.		
WAITFOR DELAY '00:00:10'			
		BEGIN TRAN	
		B1: Khai báo và khởi tạo Dynamic SQL. DECLARE @SQL NVARCHAR(MAX) SET @SQL = N'SELECT * FROM THUOC WHERE 1 = 1'	
		B2: Kiểm tra yêu cầu tìm kiếm trên từng thuộc tính và điều chỉnh Dynamic SQL tương ứng. IF @MATHUOC IS NOT NULL SET @SQL += N' AND MATHUOC LIKE "%"+@MATHUOC+"%" IF @TENTHUOC IS NOT NULL SET @SQL += N' AND TENTHUOC LIKE "%"+@TENTHUOC+"%" IF @DONVITINH IS NOT NULL SET @SQL += N' AND DONVITINH LIKE "%"+@DONVITINH+"%" IF @CHIDINH IS NOT NULL SET @SQL += N' AND CHIDINH LIKE "%"+@CHIDINH+"%" IF @SOLUONGTONKHO IS NOT NULL SET @SQL += N' AND SOLUONGTONKHO <= @SOLUONGTONKHO'	

		<pre> IF @NGAYHETHAN IS NOT NULL SET @SQL += N' AND NGAYHETHAN <= @NGAYHETHAN' IF @GIATIEN IS NOT NULL SET @SQL += N' AND GIATIEN <= GIATIEN' IF @MANVQUANLY IS NOT NULL SET @SQL += N' AND MANVQUANLY LIKE "%"+@MANVQUANLY+"%" </pre>	
		<p>B3: Thực thi Dynamic SQL tìm những thuốc thỏa điều kiện.</p> <pre> EXEC sp_executesql @SQL, N'@MATHUOC VARCHAR(5), @TENTHUOC NVARCHAR(25), @DONVITINH NVARCHAR(5), @CHIDINH NVARCHAR(50), @SOLUONGTONKHO INT, @NGAYHETHAN DATE, @GIATIEN INT, @MANVQUANLY VARCHAR(5)', @MATHUOC, @TENTHUOC, @DONVITINH, @CHIDINH, @SOLUONGTONKHO, @NGAYHETHAN, @GIATIEN, @MANVQUANLY </pre>	<p>S(THUOC) //Xin khóa đọc trên bảng THUOC thất bại do T1 đang giữ khóa.</p>
		CHỖ	
<p>B4: Kiểm tra nhân viên được giao quản lý có phải là admin.</p> <pre> IF LEFT(@MANVQUANLY, 2) != 'AD' BEGIN RAISERROR('NVQUANLY MUST BE AN ADMIN', 16, 1) ROLLBACK TRAN RETURN -4 END </pre> <p>//Do nhân viên không phải admin gây lỗi nên toàn bộ Transaction bị ROLLBACK.</p>			

		<p>B3: Thực thi Dynamic SQL tìm những thuốc thỏa điều kiện.</p> <pre>EXEC sp_executesql @SQL, N'@MATHUOC VARCHAR(5), @TENTHUOC NVARCHAR(25), @DONVITINH NVARCHAR(5), @CHIDINH NVARCHAR(50), @SOLUONGTONKHO INT, @NGAYHETHAN DATE, @GIATIEN INT, @MANVQUANLY VARCHAR(5)', @MATHUOC, @TENTHUOC, @DONVITINH, @CHIDINH, @SOLUONGTONKHO, @NGAYHETHAN, @GIATIEN, @MANVQUANLY</pre>	<p>S(THUOC) //Xin khóa đọc trên bảng THUOC thành công.</p>
		<p>COMMIT TRAN</p> <p>RETURN 0</p>	

II. Sinh viên thực hiện: 21127296 – Đặng Hà Huy

2. Tình huống 2: Nhân viên đang tìm thuốc bằng tên thì admin thay đổi tên của một thuốc bất kì có trong danh sách sẽ hiển thị.

Lỗi Unrepeatable Read xảy ra do mức cô lập của T1 là **READ COMMITTED**, tức T1 sẽ nhả khóa đọc ngay sau khi kết thúc việc đọc nên T2 có thể chen ngang vào để xin khóa ghi. Do đó, để T1 giữ khóa đọc đến khi giao tác kết thúc cần nâng mức cô lập của T1 lên **REPEATABLE READ**.

<p>ERR02: Unrepeatable Read T1 (User = Nhân viên): Tìm thuốc bằng tên. T2 (User = Quản trị viên): Thay đổi tên thuốc.</p>			
USP_THUOC_SEARCH	Khóa	USP_THUOC_UPD	Khóa
<p>Input: Mã thuốc, tên thuốc, đơn vị tính, chỉ định, số lượng tồn kho, ngày hết hạn, giá tiền, mã nhân viên quản lý. (Tất cả Input đều không bắt buộc).</p> <p>Output: Những thuốc thỏa điều kiện tìm kiếm.</p>		<p>Input: Mã thuốc, tên thuốc, đơn vị tính, chỉ định, số lượng tồn kho, ngày hết hạn, giá tiền, mã nhân viên quản lý, người update.</p> <p>Output: 0 nếu update thành công, hoặc mã lỗi.</p>	

SET TRANSACTION ISOLATION LEVEL REPEATABLE READ		SET TRANSACTION ISOLATION LEVEL READ COMMITTED	
BEGIN TRAN			
B1: Khai báo và khởi tạo Dynamic SQL. DECLARE @SQL NVARCHAR(MAX) SET @SQL = N' FROM THUOC WHERE 1 = 1'			
B2: Kiểm tra yêu cầu tìm kiếm trên từng thuộc tính và điều chỉnh Dynamic SQL tương ứng. IF @MATHUOC IS NOT NULL SET @SQL += N' AND MATHUOC LIKE "%"+@MATHUOC+"%" IF @TENTHUOC IS NOT NULL SET @SQL += N' AND TENTHUOC LIKE "%"+@TENTHUOC+"%" IF @DONVITINH IS NOT NULL SET @SQL += N' AND DONVITINH LIKE "%"+@DONVITINH+"%" IF @CHIDINH IS NOT NULL SET @SQL += N' AND CHIDINH LIKE "%"+@CHIDINH+"%" IF @SOLUONGTONKHO IS NOT NULL SET @SQL += N' AND SOLUONGTONKHO <= @SOLUONGTONKHO' IF @NGAYHETHAN IS NOT NULL SET @SQL += N' AND NGAYHETHAN <= @NGAYHETHAN' IF @GIATIEN IS NOT NULL SET @SQL += N' AND GIATIEN <= GIATIEN' IF @MANVQUANLY IS NOT NULL SET @SQL += N' AND MANVQUANLY LIKE "%"+@MANVQUANLY+"%"			
B3: Tạo thêm DSQL đếm số lượng thuốc thỏa điều kiện, và hoàn thiện DSQL ban đầu.			

<pre> DECLARE @SQLCHK NVARCHAR(MAX), @RC INT SET @SQLCHK = N'SELECT @RC = COUNT(*) + @SQL SET @SQL = N'SELECT *' + @SQL </pre>			
<p>B4: Thực thi DSQL đếm và kiểm tra số lượng thuốc trả về.</p> <pre> EXEC sp_executesql @SQLCHK, N'@MATHUOC VARCHAR(5), @TENTHUOC NVARCHAR(25), @DONVITINH NVARCHAR(5), @CHIDINH NVARCHAR(50), @SOLUONGTONKHO INT, @NGAYHETHAN DATE, @GIATIEN INT, @MANVQUANLY VARCHAR(5), @RC INT OUTPUT', @MATHUOC, @TENTHUOC, @DONVITINH, @CHIDINH, @SOLUONGTONKHO, @NGAYHETHAN, @GIATIEN, @MANVQUANLY, @RC OUTPUT IF @RC = 0 BEGIN RAISERROR('NO RESULT FOUND!', 16, 1) ROLLBACK TRAN RETURN -1 END </pre>	<p>S(THUOC) //Xin khóa đọc trên bảng THUOC thành công.</p>		
<pre> WAITFOR DELAY '00:00:10' </pre>			
		<pre> BEGIN TRAN </pre>	
		<p>B1: Kiểm tra tính hợp lệ của mã thuốc, các thuốc tính có thể update, và người update.</p> <pre> IF NOT EXISTS (SELECT * FROM THUOC WHERE MATHUOC = @MATHUOC) BEGIN RAISERROR('INVALID MATHUOC', 16, 1) ROLLBACK TRAN RETURN -1 END </pre>	<p>S(THUOC) //Xin khóa đọc trên bảng THUOC thành công.</p>

		<pre> IF @DONVITINH != N'VIÊN' AND @DONVITINH != N'GÓI' AND @DONVITINH != N'HỘP' BEGIN RAISERROR(N'INVALID DONVITINH, MUST BE EITHER VIÊN, HỘP, OR GÓI', 16, 1) ROLLBACK TRAN RETURN -2 END IF @SOLUONGTONKHO < 0 BEGIN RAISERROR('INVALID SOLUONGTONKHO', 16, 1) ROLLBACK TRAN RETURN -3 END IF @GIATIEN < 0 BEGIN RAISERROR('INVALID GIATIEN', 16, 1) ROLLBACK TRAN RETURN -4 END IF @TENTHUOC IS NULL OR @CHIDINH IS NULL OR @NGAYHETHAN IS NULL OR @NGUOIUPDATE IS NULL BEGIN RAISERROR('INVALID DATA', 16, 1) ROLLBACK TRAN RETURN -5 END IF @NGUOIUPDATE NOT LIKE 'AD%' OR @NGUOIUPDATE != @MANVQUANLY BEGIN </pre>	
--	--	-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	--

		RAISERROR('ONLY ADMIN MANAGING THIS THUOC CAN UPDATE IT', 16, 1) ROLLBACK TRAN RETURN -6 END	
		B2: Update các thuộc tính của thuốc với mã thuốc phù hợp. UPDATE THUOC SET TENTHUOC = @TENTHUOC, DONVITINH = @DONVITINH, CHIDINH = @CHIDINH, SOLUONGTONKHO = @SOLUONGTONKHO, NGAYHETHAN = @NGAYHETHAN, GIATIEN = @GIATIEN, MANVQUANLY = @MANVQUANLY WHERE MATHUOC = @MATHUOC	U,X(THUOC) //Xin khóa update và khóa ghi trên bảng THUOC thất bại do T1 đang giữ khóa đọc.
		CHỜ	
B5: Thực thi DSQL tìm những thuốc thỏa điều kiện. EXEC sp_executesql @SQL, N'@MATHUOC VARCHAR(5), @TENTHUOC NVARCHAR(25), @DONVITINH NVARCHAR(5), @CHIDINH NVARCHAR(50), @SOLUONGTONKHO INT, @NGAYHETHAN DATE, @GIATIEN INT, @MANVQUANLY VARCHAR(5)', @MATHUOC, @TENTHUOC, @DONVITINH, @CHIDINH, @SOLUONGTONKHO, @NGAYHETHAN, @GIATIEN, @MANVQUANLY	S(THUOC) //Vẫn còn giữ khóa đọc trên bảng THUOC đã xin ở B4.		
COMMIT TRAN RETURN 0			
		B2: Update các thuộc tính của thuốc với mã thuốc phù hợp. UPDATE THUOC SET TENTHUOC = @TENTHUOC, DONVITINH = @DONVITINH, CHIDINH = @CHIDINH, SOLUONGTONKHO =	U,X(THUOC) //Xin khóa update và khóa ghi trên

		@SOLUONGTONKHO, NGAYHETHAN = @NGAYHETHAN, GIATIEN = @GIATIEN, MANVQUANLY = @MANVQUANLY WHERE MATHUOC = @MATHUOC	bảng THUOC thành công.
		COMMIT TRAN RETURN 0	

III. Sinh viên thực hiện: 21127135 – Diệp Hữu Phúc

3. Tình huống 3: Nhân viên tìm thuốc có số lượng tồn kho nhỏ hơn X thì admin thêm thuốc mới cũng thỏa điều kiện đó.

Lỗi Phantom Read xảy ra do mức cô lập của T1 là **READ COMMITTED**, tức T1 sẽ không khóa phạm vi các dòng thỏa điều kiện, và nhà khóa đọc ngay sau khi kết thúc việc đọc, nên T2 có thể chen ngang vào để xin khóa ghi. Do đó, để T1 khóa phạm vi các dòng thỏa điều kiện, đồng thời giữ khóa đọc đến khi giao tác kết thúc, cần nâng mức cô lập của T1 lên **SERIALIZABLE**.

ERR03: Phantom Read T1 (User = Nhân viên): Tìm thuốc có số lượng tồn kho nhỏ hơn X. T2 (User = Quản trị viên): Thêm thuốc có số lượng tồn kho nhỏ hơn X.			
USP_THUOC_SEARCH	Khóa	USP_THUOC_INS	Khóa
Input: Mã thuốc, tên thuốc, đơn vị tính, chỉ định, số lượng tồn kho, ngày hết hạn, giá tiền, mã nhân viên quản lý. (Tất cả Input đều không bắt buộc). Output: Những thuốc thỏa điều kiện tìm kiếm.		Input: Tên thuốc, đơn vị tính, chỉ định, số lượng tồn kho, ngày hết hạn, giá tiền, mã nhân viên quản lý, mã thuốc. Output: 0 nếu insert thành công, hoặc mã lỗi.	
SET TRANSACTION ISOLATION LEVEL SERIALIZABLE		SET TRANSACTION ISOLATION LEVEL READ COMMITTED	
BEGIN TRAN			
B1: Khai báo và khởi tạo Dynamic SQL. DECLARE @SQL NVARCHAR(MAX) SET @SQL = 'N' FROM THUOC WHERE 1 = 1'			

<p>B2: Kiểm tra yêu cầu tìm kiếm trên từng thuộc tính và điều chỉnh Dynamic SQL tương ứng.</p> <pre> IF @MATHUOC IS NOT NULL SET @SQL += N' AND MATHUOC LIKE "% "+@MATHUOC+"%" IF @TENTHUOC IS NOT NULL SET @SQL += N' AND TENTHUOC LIKE "% "+@TENTHUOC+"%" IF @DONVITINH IS NOT NULL SET @SQL += N' AND DONVITINH LIKE "% "+@DONVITINH+"%" IF @CHIDINH IS NOT NULL SET @SQL += N' AND CHIDINH LIKE "% "+@CHIDINH+"%" IF @SOLUONGTONKHO IS NOT NULL SET @SQL += N' AND SOLUONGTONKHO <= @SOLUONGTONKHO' IF @NGAYHETHAN IS NOT NULL SET @SQL += N' AND NGAYHETHAN <= @NGAYHETHAN' IF @GIATIEN IS NOT NULL SET @SQL += N' AND GIATIEN <= GIATIEN' IF @MANVQUANLY IS NOT NULL SET @SQL += N' AND MANVQUANLY LIKE "% "+@MANVQUANLY+"%" </pre>			
<p>B3: Tạo thêm DSQL đếm số lượng thuốc thỏa điều kiện, và hoàn thiện DSQL ban đầu.</p> <pre> DECLARE @SQLCHK NVARCHAR(MAX), @RC INT SET @SQLCHK = N'SELECT @RC = COUNT(*) + @SQL SET @SQL = N'SELECT *' + @SQL </pre>			
<p>B4: Thực thi DSQL đếm và kiểm tra số lượng thuốc trả về.</p>	<p>S(THUOC) //Xin khóa đọc trên bảng</p>		

<pre>EXEC sp_executesql @SQLCHK, N'@MATHUOC VARCHAR(5), @TENTHUOC NVARCHAR(25), @DONVITINH NVARCHAR(5), @CHIDINH NVARCHAR(50), @SOLUONGTONKHO INT, @NGAYHETHAN DATE, @GIATIEN INT, @MANVQUANLY VARCHAR(5), @RC INT OUTPUT', @MATHUOC, @TENTHUOC, @DONVITINH, @CHIDINH, @SOLUONGTONKHO, @NGAYHETHAN, @GIATIEN, @MANVQUANLY, @RC OUTPUT IF @RC = 0 BEGIN RAISERROR('NO RESULT FOUND!', 16, 1) ROLLBACK TRAN RETURN -1 END</pre>	THUOC thành công.		
WAITFOR DELAY '00:00:10'			
		BEGIN TRAN	
		<p>B1: Kiểm tra tính hợp lệ của đơn vị tính (viên, gói, hộp), số lượng tồn kho (≥ 0), giá tiền (≥ 0), mã nhân viên quản lý (AD###).</p> <pre>IF @DONVITINH != N'VIÊN' AND @DONVITINH != N'GÓI' AND @DONVITINH != N'HỘP' BEGIN RAISERROR(N'INVALID DONVITINH, MUST BE EITHER VIÊN, HỘP, OR GÓI', 16, 1) ROLLBACK TRAN RETURN -1 END IF @SOLUONGTONKHO < 0 BEGIN RAISERROR('INVALID SOLUONGTONKHO', 16, 1)</pre>	

		ROLLBACK TRAN RETURN -2 END IF @GIATIEN < 0 BEGIN RAISERROR('INVALID GIATIEN', 16, 1) ROLLBACK TRAN RETURN -3 END IF LEFT(@MANVQUANLY, 2) != 'AD' BEGIN RAISERROR('NVQUANLY MUST BE AN ADMIN', 16, 1) ROLLBACK TRAN RETURN -4 END	
		B2: Tìm mã thuốc được thêm gần đây nhất, và tạo mã thuốc mới. SELECT @MATHUOC = MATHUOC FROM THUOC WHERE MATHUOC = (SELECT MAX(MATHUOC) FROM THUOC) SET @MATHUOC = dbo.F_MAKE_ID('TH', @MATHUOC)	S(THUOC) //Xin khoá đọc trên bảng THUOC thành công.
		B3: Insert dữ liệu vào bảng THUOC. INSERT INTO THUOC VALUES (@MATHUOC, @TENTHUOC, @DONVITINH, @CHIDINH, @SOLUONGTONKHO, @NGAYHETHAN, @GIATIEN, @MANVQUANLY)	X(THUOC) //Xin khóa ghi trên bảng THUOC thất bại do T1 đang giữ khóa đọc.
		CHỖ	

<p>B5: Thực thi DSQL tìm những thuốc thỏa điều kiện.</p> <p>EXEC sp_executesql @SQL, N'@MATHUOC VARCHAR(5), @TENTHUOC NVARCHAR(25), @DONVITINH NVARCHAR(5), @CHIDINH NVARCHAR(50), @SOLUONGTONKHO INT, @NGAYHETHAN DATE, @GIATIEN INT, @MANVQUANLY VARCHAR(5)', @MATHUOC, @TENTHUOC, @DONVITINH, @CHIDINH, @SOLUONGTONKHO, @NGAYHETHAN, @GIATIEN, @MANVQUANLY</p>	<p>S(THUOC) //Vẫn còn giữ khóa đọc trên bảng THUOC đã xin ở B4.</p>		
<p>COMMIT TRAN RETURN 0</p>			
		<p>B3: Insert dữ liệu vào bảng THUOC.</p> <p>INSERT INTO THUOC VALUES (@MATHUOC, @TENTHUOC, @DONVITINH, @CHIDINH, @SOLUONGTONKHO, @NGAYHETHAN, @GIATIEN, @MANVQUANLY)</p>	<p>X(THUOC) //Xin khóa ghi trên bảng THUOC thành công.</p>
		<p>COMMIT TRAN RETURN 0</p>	

4. Tình huống 4: Admin cập nhật số lượng tồn kho của thuốc X nhưng chưa hoàn thành update thì nha sĩ cập nhật số lượng của thuốc X trong một toa thuốc bất kì.

Lỗi Lost Update xảy ra do khóa đọc (Shared Lock) của T1 trên bảng A không ngăn được T2 chen ngang vào xin khóa update và ghi trên bảng này. Do đó, để khắc phục T1 cần cài khóa update (Update Lock) trên bảng A để cấm T2 xin khóa update và ghi trên cùng bảng.

ERR04: Lost Update

T1 (User = Quản trị viên): Chỉnh sửa số lượng tồn kho của thuốc X.

T2 (User = Nha sĩ): Cập nhật toa thuốc có liên quan tới thuốc X.

USP_THUOC_UPD	Khóa	USP_TOATHUOC_UPD	Khóa
Input: Mã thuốc, tên thuốc, đơn vị tính, chỉ định, số lượng tồn kho, ngày hết hạn, giá tiền, mã nhân viên quản lý, người update. Output: 0 nếu update thành công, hoặc mã lỗi.		Input: Mã hồ sơ bệnh án, mã giấy khám bệnh, mã thuốc, số lượng. Output: 0 nếu update thành công, hoặc mã lỗi.	
SET TRANSACTION ISOLATION LEVEL READ COMMITTED		SET TRANSACTION ISOLATION LEVEL READ COMMITTED	
BEGIN TRAN			
B1: Kiểm tra tính hợp lệ của mã thuốc, các thuộc tính có thể update, và người update. IF NOT EXISTS (SELECT * FROM THUOC WITH (UPDLOCK) WHERE MATHUOC = @MATHUOC) BEGIN RAISERROR('INVALID MATHUOC', 16, 1) ROLLBACK TRAN RETURN -1 END IF @DONVITINH != N'VIÊN' AND @DONVITINH != N'GÓI' AND @DONVITINH != N'HỘP' BEGIN RAISERROR(N'INVALID DONVITINH, MUST BE EITHER VIÊN, HỘP, OR GÓI', 16, 1) ROLLBACK TRAN RETURN -2 END IF @SOLUONGTONKHO < 0 BEGIN RAISERROR('INVALID SOLUONGTONKHO', 16, 1) ROLLBACK TRAN RETURN -3	U(THUOC) //Xin khóa update trên bảng THUOC thành công.		

<pre> END IF @GIATIEN < 0 BEGIN RAISERROR('INVALID GIATIEN', 16, 1) ROLLBACK TRAN RETURN -4 END IF @TENTHUOC IS NULL OR @CHIDINH IS NULL OR @NGAYHETHAN IS NULL OR @NGUOIUPDATE IS NULL BEGIN RAISERROR('INVALID DATA', 16, 1) ROLLBACK TRAN RETURN -5 END IF @NGUOIUPDATE NOT LIKE 'AD%' OR @NGUOIUPDATE != @MANVQUANLY BEGIN RAISERROR('ONLY ADMIN MANAGING THIS THUOC CAN UPDATE IT', 16, 1) ROLLBACK TRAN RETURN -6 END </pre>			
WAITFOR DELAY '00:00:10'			
		BEGIN TRAN	
		<pre> B1: Kiểm tra sự tồn tại của toa thuốc. IF NOT EXISTS (SELECT * FROM TOATHUOC WHERE MAHSBA = @MAHSBA AND MAGIAYKHAMBENH = @MAGIAYKHAMBENH AND MATHUOC = @MATHUOC) BEGIN RAISERROR('INVALID TOATHUOC', 16, 1) ROLLBACK TRAN RETURN -1 </pre>	<p>S(TOA THUOC) //Xin khóa đọc trên bảng TOATHUOC thành công.</p>

		END	
		<p>B2: Kiểm tra tính hợp lệ của mã thuốc, và số lượng yêu cầu có vượt quá số lượng tồn kho.</p> <pre> IF NOT EXISTS (SELECT * FROM THUOC WHERE MATHUOC = @MATHUOC AND SOLUONGTONKHO >= @SOLUONG) BEGIN RAISERROR('INVALID MATHUOC, OR SOLUONGTONKHO IS INSUFFICIENT', 16, 1) ROLLBACK TRAN RETURN -2 END </pre>	<p>S(THUOC) //Xin khóa đọc trên bảng THUOC thành công.</p>
		<p>B3: Lấy số lượng hiện tại ở toa thuốc.</p> <pre> DECLARE @DIFFSOLUONG INT, @DIFFCOST INT SELECT @DIFFSOLUONG = SOLUONG FROM TOATHUOC WHERE MAHSBA = @MAHSBA AND MAGIAYKHAMBENH = @MAGIAYKHAMBENH AND MATHUOC = @MATHUOC </pre>	<p>S(TOA THUOC) //Xin khóa đọc trên bảng TOATHUOC thành công.</p>
		<p>B4: Tính chênh lệch giữa số lượng hiện tại và số lượng muốn cập nhật. Tính chênh lệch giá tiền dựa vào chênh lệch số lượng.</p> <pre> SET @DIFFSOLUONG = @SOLUONG - @DIFFSOLUONG SET @DIFFCOST = (SELECT GIATIEN FROM THUOC WHERE MATHUOC = @MATHUOC) * @DIFFSOLUONG </pre>	<p>S(THUOC) //Xin khóa đọc trên bảng THUOC thành công.</p>
		<p>B5: Update số lượng cho toa thuốc tương ứng.</p> <pre> UPDATE TOATHUOC SET SOLUONG = @SOLUONG WHERE MAHSBA = @MAHSBA AND MAGIAYKHAMBENH = @MAGIAYKHAMBENH AND MATHUOC = @MATHUOC </pre>	<p>U,X(TOA THUOC) //Xin khóa update và khóa ghi trên bảng TOATHUOC thành công.</p>

		<p>B6: Update số lượng tồn kho dựa vào chênh lệch cho thuốc tương ứng.</p> <p>UPDATE THUOC SET SOLUONGTONKHO -= @DIFFSOLUONG WHERE MATHUOC = @MATHUOC</p>	<p>U,X(THUOC)</p> <p>//Xin khóa update và khóa ghi trên bảng THUOC thất bại do T1 đang giữ khóa update.</p>
		CHỜ	
<p>B2: Update các thuộc tính của thuốc với mã thuốc phù hợp.</p> <p>UPDATE THUOC SET TENTHUOC = @TENTHUOC, DONVITINH = @DONVITINH, CHIDINH = @CHIDINH, SOLUONGTONKHO = @SOLUONGTONKHO, NGAYHETHAN = @NGAYHETHAN, GIATIEN = @GIATIEN, MANVQUANLY = @MANVQUANLY WHERE MATHUOC = @MATHUOC</p>	<p>U,X(THUOC)</p> <p>//Với khóa update đã xin được ở B1, xin khóa ghi trên bảng THUOC thành công.</p>		
<p>COMMIT TRAN</p> <p>RETURN 0</p>			
		<p>B6: Update số lượng tồn kho dựa vào chênh lệch cho thuốc tương ứng.</p> <p>UPDATE THUOC SET SOLUONGTONKHO -= @DIFFSOLUONG WHERE MATHUOC = @MATHUOC</p>	<p>U,X(THUOC)</p> <p>//Xin khóa update và khóa ghi trên bảng THUOC thành công.</p>
		<p>B7: Update tổng tiền thuốc dựa vào chênh lệch cho giấy khám bệnh tương ứng.</p> <p>UPDATE GIAYKHAMBENH SET TONGTIENTHUOC += @DIFFCOST WHERE MAHSBA = @MAHSBA AND MAGIAYKHAMBENH = @MAGIAYKHAMBENH</p>	<p>U,X(GIAY KHAMBENH)</p> <p>//Xin khóa update và khóa ghi trên bảng GIAY KHAMBENH thành công.</p>

		COMMIT TRAN RETURN 0	
--	--	-------------------------	--

IV. Sinh viên thực hiện: 21127004 – Trần Nguyễn An Phong

5. Tình huống 5: Nhân viên muốn cập nhật thông tin tài khoản của mình nhưng chưa hoàn thành thì đồng thời cũng có một admin muốn cập nhật thông tin của nhân viên đó.

Lỗi Conversion Deadlock xảy ra do T1 và T2 thực hiện cùng một giao tác có mức cô lập là **REPEATABLE READ**, tức khóa đọc xin trên bảng A sẽ được giữ đến khi giao tác kết thúc, dẫn đến nếu T1 và T2 cùng xin khóa đọc rồi sau đó cùng xin khóa ghi sẽ từ chối nhau. Do đó, để khắc phục cần phải hạ mức cô lập của giao tác xuống **READ COMMITTED**.

ERR05: Conversion Deadlock T1 (User = Nhân viên): Cập nhật thông tin tài khoản của mình. T2 (User = Quản trị viên): Chính sửa thông tin tài khoản của nhân viên trên.			
USP_NHANVIEN_UPD	Khóa	USP_NHANVIEN_UPD	Khóa
Input: Mã nhân viên, họ tên, địa chỉ, số điện thoại, loại nhân viên, mật khẩu. Output: 0 nếu update thành công, hoặc mã lỗi.		Input: Mã nhân viên, họ tên, địa chỉ, số điện thoại, loại nhân viên, mật khẩu. Output: 0 nếu update thành công, hoặc mã lỗi.	
SET TRANSACTION ISOLATION LEVEL READ COMMITTED		SET TRANSACTION ISOLATION LEVEL READ COMMITTED	
BEGIN TRAN			
B1: Kiểm tra tính hợp lệ của mã nhân viên. IF NOT EXISTS (SELECT * FROM NHANVIEN WHERE MANHANVIEN = @MANHANVIEN) BEGIN RAISERROR('INVALID MANHANVIEN', 16, 1) ROLLBACK TRAN RETURN -1 END	S(NHANVIEN) //Xin khóa đọc trên bảng NHANVIEN thành công.		
B2: Kiểm tra tính hợp lệ của các thuộc tính có thể update.			

<pre> IF @LOAINHANVIEN != 0 AND @LOAINHANVIEN != 1 AND @LOAINHANVIEN != 2 BEGIN RAISERROR('INVALID LOAINHANVIEN', 16, 1) ROLLBACK TRAN RETURN -2 END IF @HOTEN IS NULL OR @DIACHI IS NULL OR @SODT IS NULL OR @MATKHAU IS NULL BEGIN RAISERROR('INVALID DATA', 16, 1) ROLLBACK TRAN RETURN -3 END </pre>			
WAITFOR DELAY '00:00:10'			
		BEGIN TRAN	
		<p>B1: Kiểm tra tính hợp lệ của mã nhân viên.</p> <pre> IF NOT EXISTS (SELECT * FROM NHANVIEN WHERE MANHANVIEN = @MANHANVIEN) BEGIN RAISERROR('INVALID MANHANVIEN', 16, 1) ROLLBACK TRAN RETURN -1 END </pre>	<p>S(NHANVIEN) //Xin khóa đọc trên bảng NHANVIEN thành công.</p>
		<p>B2: Kiểm tra tính hợp lệ của các thuộc tính có thể update.</p> <pre> IF @LOAINHANVIEN != 0 AND @LOAINHANVIEN != 1 AND @LOAINHANVIEN != 2 BEGIN RAISERROR('INVALID LOAINHANVIEN', 16, 1) ROLLBACK TRAN </pre>	

		RETURN -2 END IF @HOTEN IS NULL OR @DIACHI IS NULL OR @SODT IS NULL OR @MATKHAU IS NULL BEGIN RAISERROR('INVALID DATA', 16, 1) ROLLBACK TRAN RETURN -3 END	
		WAITFOR DELAY '00:00:10'	
B3: Update nhân viên với mã nhân viên phù hợp. UPDATE NHANVIEN SET HOTEN = @HOTEN, DIACHI = @DIACHI, SODT = @SODT, LOAINHANVIEN = @LOAINHANVIEN, MATKHAU = @MATKHAU WHERE MANHANVIEN = @MANHANVIEN	U,X(NHAN VIEN) //Xin khóa update và khóa ghi trên bảng NHANVIEN thành công.		
COMMIT TRAN RETURN 0			
		B3: Update nhân viên với mã nhân viên phù hợp. UPDATE NHANVIEN SET HOTEN = @HOTEN, DIACHI = @DIACHI, SODT = @SODT, LOAINHANVIEN = @LOAINHANVIEN, MATKHAU = @MATKHAU WHERE MANHANVIEN = @MANHANVIEN	U,X(NHAN VIEN) //Xin khóa update và khóa ghi trên bảng NHANVIEN thành công.
		COMMIT TRAN RETURN 0	

6. Tình huống 6: Nha sĩ trong lúc cập nhật thông tin toa thuốc X thì lại thực hiện xóa toa thuốc này.

Lỗi Cycle Deadlock xảy ra do T1 và T2 ghi cùng các bảng nhưng lại thiếu sự đồng bộ trong thứ tự ghi các bảng này. Do đó, để khắc phục cần phải sắp xếp thứ tự thực hiện các thao tác ghi giữa T1 và T2 khớp với nhau.

ERR06: Cycle Deadlock T1 (User = Nha sĩ): Cập nhật toa thuốc X. T2 (User = Nha sĩ): Xóa toa thuốc X.			
USP_TOATHUOC_UPD	Khóa	USP_TOATHUOC_DEL	Khóa
Input: Mã hồ sơ bệnh án, mã giấy khám bệnh, mã thuốc, số lượng. Output: 0 nếu update thành công, hoặc mã lỗi.		Input: Mã hồ sơ bệnh án, mã giấy khám bệnh, mã thuốc. Output: 0 nếu update thành công, hoặc mã lỗi.	
SET TRANSACTION ISOLATION LEVEL READ COMMITTED		SET TRANSACTION ISOLATION LEVEL READ COMMITTED	
BEGIN TRAN			
B1: Kiểm tra sự tồn tại của toa thuốc. IF NOT EXISTS (SELECT * FROM TOATHUOC WHERE MAHSBA = @MAHSBA AND MAGIAYKHAMBENH = @MAGIAYKHAMBENH AND MATHUOC = @MATHUOC) BEGIN RAISERROR('INVALID TOATHUOC', 16, 1) ROLLBACK TRAN RETURN -1 END	S(TOA THUOC) //Xin khóa đọc trên bảng TOATHUOC thành công.		
B2: Kiểm tra tính hợp lệ của mã thuốc, và số lượng yêu cầu có vượt quá số lượng tồn kho. IF NOT EXISTS (SELECT * FROM THUOC WHERE MATHUOC = @MATHUOC AND SOLUONGTONKHO >= @SOLUONG) BEGIN RAISERROR('INVALID MATHUOC, OR SOLUONGTONKHO IS INSUFFICIENT', 16, 1) ROLLBACK TRAN	S(THUOC) //Xin khóa đọc trên bảng THUOC thành công.		

RETURN -2 END			
B3: Lấy số lượng hiện tại ở toa thuốc. DECLARE @DIFFSOLUONG INT, @DIFFCOST INT SELECT @DIFFSOLUONG = SOLUONG FROM TOATHUOC WHERE MAHSBA = @MAHSBA AND MAGIAYKHAMBENH = @MAGIAYKHAMBENH AND MATHUOC = @MATHUOC	S(TOA THUOC) //Xin khóa đọc trên bảng TOATHUOC thành công.		
B4: Tính chênh lệch giữa số lượng hiện tại và số lượng muốn cập nhật. Tính chênh lệch giá tiền dựa vào chênh lệch số lượng. SET @DIFFSOLUONG = @SOLUONG - @DIFFSOLUONG SET @DIFFCOST = (SELECT GIATIEN FROM THUOC WHERE MATHUOC = @MATHUOC) * @DIFFSOLUONG	S(THUOC) //Xin khóa đọc trên bảng THUOC thành công.		
WAITFOR DELAY '00:00:05'			
		BEGIN TRAN	
		B1: Kiểm tra sự tồn tại của toa thuốc. IF NOT EXISTS (SELECT * FROM TOATHUOC WHERE MAHSBA = @MAHSBA AND MAGIAYKHAMBENH = @MAGIAYKHAMBENH AND MATHUOC = @MATHUOC) BEGIN RAISERROR('INVALID TOATHUOC', 16, 1) ROLLBACK TRAN RETURN -1 END	S(TOA THUOC) //Xin khóa đọc trên bảng TOATHUOC thành công.
		B2: Lấy số lượng hiện tại ở toa thuốc. DECLARE @SOLUONG INT, @COST INT SELECT @SOLUONG = SOLUONG FROM TOATHUOC WHERE MAHSBA = @MAHSBA	S(TOA THUOC) //Xin khóa đọc trên bảng

		AND MAGIAYKHAMBENH = @MAGIAYKHAMBENH AND MATHUOC = @MATHUOC	TOATHUOC thành công.
		B3: Tính giá tiền hiện tại của toa thuốc dựa vào số lượng. SET @COST = (SELECT GIATIEN FROM THUOC WHERE MATHUOC = @MATHUOC) * @SOLUONG	S(THUOC) // Xin khóa đọc trên bảng THUOC thành công.
		WAITFOR DELAY '00:00:05'	
B5: Update số lượng cho toa thuốc tương ứng. UPDATE TOATHUOC SET SOLUONG = @SOLUONG WHERE MAHSBA = @MAHSBA AND MAGIAYKHAMBENH = @MAGIAYKHAMBENH AND MATHUOC = @MATHUOC	U,X(TOA THUOC) //Xin khóa update và khóa ghi trên bảng TOATHUOC thành công.		
		WAITFOR DELAY '00:00:05'	
		B4: Xóa toa thuốc tương ứng. DELETE FROM TOATHUOC WHERE MAHSBA = @MAHSBA AND MAGIAYKHAMBENH = @MAGIAYKHAMBENH AND MATHUOC = @MATHUOC	U,X(TOA THUOC) //Xin khóa update và khóa ghi trên bảng TOATHUOC thất bại do T1 đang giữ khóa.
		CHỜ	
B6: Update tổng tiền thuốc dựa vào chênh lệch cho giấy khám bệnh tương ứng. UPDATE GIAYKHAMBENH SET TONGTIENTHUOC += @DIFFCOST WHERE	U,X(GIAY KHAMBENH) //Xin khóa update và khóa ghi trên		

MAHSBA = @MAHSBA AND MAGIAYKHAMBENH = @MAGIAYKHAMBENH	bảng GIAY KHAMBENH thành công.		
B7: Update số lượng tồn kho, dựa vào chênh lệch cho thuốc tương ứng. UPDATE THUOC SET SOLUONGTONKHO -= @DIFFSOLUONG WHERE MATHUOC = @MATHUOC	U,X(THUOC) //Xin khóa update và khóa ghi trên bảng THUOC thành công.		
COMMIT TRAN RETURN 0			
		B4: Xóa toa thuốc tương ứng. DELETE FROM TOATHUOC WHERE MAHSBA = @MAHSBA AND MAGIAYKHAMBENH = @MAGIAYKHAMBENH AND MATHUOC = @MATHUOC	U,X(TOA THUOC) //Xin khóa update và khóa ghi trên bảng TOATHUOC thành công.
		WAITFOR DELAY '00:00:05'	
		B5: Update giảm tổng tiền thuốc, dựa vào giá toa thuốc, cho giấy khám bệnh tương ứng. UPDATE GIAYKHAMBENH SET TONGTIENTHUOC -= @COST WHERE MAHSBA = @MAHSBA AND MAGIAYKHAMBENH = @MAGIAYKHAMBENH	U,X(GIAY KHAMBENH) //Xin khóa update và khóa ghi trên bảng GIAY KHAMBENH thành công.
		B6: Update tăng số lượng tồn kho, dựa vào số lượng, cho thuốc tương ứng. UPDATE THUOC SET SOLUONGTONKHO += @SOLUONG WHERE MATHUOC = @MATHUOC	U,X(THUOC) //Xin khóa update và khóa ghi trên bảng THUOC thành công.

		COMMIT TRAN RETURN 0	
--	--	-------------------------	--