**d**

**BỘ MÔN HỆ THỐNG THÔNG TIN – KHOA CÔNG NGHỆ THÔNG TIN**

Nhóm sinh viên thực hiện: Nhóm 5

1. 19127041 – Lê Thị Phương Linh
2. 19127173 – Hoàng Lê Khanh
3. 19127361 – Nguyễn Thị Ngọc Diệu

**ĐẠI HỌC KHOA HỌC TỰ NHIÊN THÀNH PHỐ HỒ CHÍ MINH**

BÁO CÁO ĐỒ ÁN THỰC HÀNH

MÔN HỆ QUẢN TRỊ CƠ SỞ DỮ LIỆU

GVHD

TS. PHẠM THỊ BẠCH HUỆ

THS. TIẾT GIA HỒNG

Logo

Description automatically generated

# **THÔNG TIN NHÓM**

Số thứ tự nhóm: **Nhóm 5**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | MSSV | Họ tên | Công việc | % Hoàn thành |
| 1 | **19127041** | **Lê Thị Phương Linh** |  | **100 %** |
| 2 | **19127173** | **Hoàng Lê Khanh** |  | **100 %** |
| 3 | **19127361** | **Nguyễn Thị Ngọc Diệu** |  | **100 %** |

# **CÀI ĐẶT TÌNH HUỐNG TRANH CHẤP**

# Sinh viên thực hiện: Lê Thị Phương Linh

## Tình huống 1:

|  |  |  |  |
| --- | --- | --- | --- |
| **ERR01: Phantom Read**  T1 (User = đối tác): thực hiện xem doanh thu của mình (đối với đơn hàng đã giao).  T2 (User = tài xế): Thực hiện cập nhật trạng thái “Đã giao hàng” cho đơn hàng của đối tác 🡪 doanh thu của đối tác tăng | | | |
| USP\_DT\_XEMDOANHTHU | **Khóa** | USP\_TX\_DAGIAOHANG | **Khóa** |
| ***Input:***  ***Output***: Tổng doanh thu, tổng số lượng đơn hàng kèm danh sách đơn hàng đã giao. | ***Input***: Trạng thái đơn hàng, mã đơn hàng, thời gian thực hiện giao hàng  ***Output***: Thêm thành công trạng thái mới của đơn hàng. |
| SET TRANSACTION ISOLATION  LEVEL **REPEATEABLE READ** |  |
| BEGIN TRAN |  |  |  |
| B1: Tính tổng doanh thu và tổng đơn hàng với trạng thái đã giao  SELECT SUM(PHI\_SAN\_PHAM - PHI\_VAN\_CHUYEN) AS DOANH\_THU, COUNT(\*) AS TONG\_LUONG\_DON  FROM UV\_DHDOITAC  WHERE TRANG\_THAI = N'Đã giao hàng'  \* NOTE: UV\_DHDOITAC là một view hiển thị **danh sách đơn hàng** của chính đối tác đó. | **S(UV\_DHDOITAC)**  //Xin khoá đọc trên view UV\_DHDOITAC với điều kiện trạng thái đơn hàng là “Đã giao hàng” |  |  |
| WAITFOR DELAY '00:00:05' |  |  |  |
|  |  | BEGIN TRAN |  |
|  |  | B1: Kiểm tra đơn hàng có tồn tại / có phải là đơn hàng mà tài xế đã nhận hay không  IF @MaDH NOT IN (SELECT MADH  FROM UV\_DHTAIXE)  BEGIN  PRINT @MaDH + N' không phải của tài xế!!'  ROLLBACK TRAN  RETURN 1  END  \* NOTE: UV\_DHTAIXE là một view hiển thị **danh sách đơn hàng** mà tài xế đã nhận. | **S(UV\_DHTAIXE)**  //Xin khoá đọc trên view UV\_DHTAIXE |
|  |  | B2: Kiểm tra đơn hàng có phải đã được giao rồi hay không  IF EXISTS (SELECT \*  FROM UV\_TTDHTAIXE  WHERE TEN\_TRANG\_THAI = @TrangThai AND MADH = @MaDH)  BEGIN  PRINT @MaDH + N' đã được giao trước đó!!'  ROLLBACK TRAN  RETURN 1  END  \* NOTE: UV\_TTDHTAIXE là một view hiển thị **danh sách trạng thái đơn hàng** mà tài xế đã nhận. | **S(UV\_TTDHTAIXE)**  //Xin khoá đọc trên view UV\_TTDHTAIXE |
|  |  | B3: Thêm trạng thái mới của đơn hàng  INSERT UV\_TTDHTAIXE  VALUES(@TrangThai, @MaDH,@ThoiGian) | **X(UV\_TTDHTAIXE)**  //Xin khoá ghi trên view UV\_TTDHTAIXE |
|  |  | B4: Cập nhật thông tin đơn hàng  UPDATE UV\_DHTAIXE  SET TRANG\_THAI = @TrangThai  WHERE MADH = @MaDH | **X(UV\_DHTAIXE)**  //Xin khoá ghi trên view UV\_DHTAIXE |
|  |  | COMMIT |  |
| B2: Hiển thị danh sách đơn hàng  SELECT MADH, MAKH, PHI\_SAN\_PHAM, PHI\_GIAM, PHI\_SAN\_PHAM - PHI\_GIAM AS THANH\_TIEN, HINH\_THUC\_THANH\_TOAN  FROM UV\_DHDOITAC  WHERE TRANG\_THAI = N'Đã giao hàng' | **S(UV\_DHDOITAC)**  //Xin khoá đọc trên view UV\_DHDOITAC với điều kiện trạng thái đơn hàng là “Đã giao hàng” |  |  |
| COMMIT |  |  |  |

## Tình huống 2:

|  |  |  |  |
| --- | --- | --- | --- |
| **ERR02: *Cycle Deadlock***  T1 (User = đối tác): cập nhật thông tin của đối tác.  T2 (User = đối tác): cập nhật thông tin của đối tác. | | | |
| USP\_DT\_TTCASE1 | **Khóa** | USP\_DT\_TTCASE2 | **Khóa** |
| ***Input:*** Tên người đại diện, mã chi nhánh, địa chỉ của chi nhánh (đường)  ***Output***: Deadlock | ***Input***: Tên người đại diện, mã chi nhánh, địa chỉ của chi nhánh (đường)  ***Output***: Deadlock |
| BEGIN TRAN |  |  |  |
| B1: Kiểm tra chi nhánh  IF @MaCN NOT IN (SELECT MACN  FROM UV\_CN\_DOITAC)  BEGIN  PRINT @MaCN + N' không phải chi nhánh của đối tác!!'  ROLLBACK TRAN  RETURN 1  END  \* NOTE: UV\_CN\_DOITAC là một view hiển thị **danh sách chi nhánh** của chính đối tác đó. | **S(UV\_DHDOITAC)**  //Xin khóa đọc trên view UV\_CN\_DOITAC |  |  |
| B2: Cập nhật tên người đại diện của đối tác  UPDATE UV\_TTDOITAC  SET NGUOI\_DAI\_DIEN = @NguoiDaiDien | **X(UV\_DHDOITAC)**  //Xin khóa ghi trên view UV\_TTDOITAC |  |  |
| WAITFOR DELAY '00:00:05' |  |  |  |
|  |  | BEGIN TRAN |  |
|  |  | B1: Kiểm tra chi nhánh  IF @MaCN NOT IN (SELECT MACN  FROM UV\_CN\_DOITAC)  BEGIN  PRINT @MaCN + N' không phải chi nhánh của đối tác!!'  ROLLBACK TRAN  RETURN 1  END  \* NOTE: UV\_CN\_DOITAC là một view hiển thị **danh sách chi nhánh** của chính đối tác đó. | **S(UV\_DHDOITAC)**  //Xin khóa đọc trên view UV\_CNDOITAC |
|  |  | B2: Cập nhật địa chỉ của chi nhánh  UPDATE UV\_CN\_DOITAC  SET DUONG = @Duong  WHERE MACN = @MaCN | **X(UV\_CN\_DOITAC)**  //Xin khóa ghi trên view UV\_CN\_DOITAC |
|  |  | WAITFOR DELAY '00:00:05' |  |
| B3: Cập nhật địa chỉ của chi nhánh  UPDATE UV\_CN\_DOITAC  SET DUONG = @Duong  WHERE MACN = @MaCN | **X(UV\_CN\_DOITAC)**  //Xin khóa ghi trên view UV\_CN\_DOITAC |  |  |
|  |  |  |  |
|  |  | B2: Cập nhật tên người đại diện của đối tác  UPDATE UV\_TTDOITAC  SET NGUOI\_DAI\_DIEN = @NguoiDaiDien | **X(UV\_DHDOITAC)**  //Xin khóa ghi trên view UV\_TTDOITAC |
| COMMIT |  | COMMIT |  |
|  |  |  |  |

## Tình huống 3:

|  |  |  |  |
| --- | --- | --- | --- |
| **ERR03: *Phantom Read***  T1 (User = khách hàng): khách hàng tìm kiếm sản phẩm theo tên.  T2 (User = đối tác): thêm một sản phẩm mới cùng tên với sản phẩm khách hàng đang tìm. | | | |
| USP\_KH\_TIMSP | **Khóa** | USP\_DT\_THEMSP | **Khóa** |
| ***Input:*** Tên sản phẩm  ***Output***: Số lượng tìm được và danh sách sản phẩm tương ứng | ***Input***: Mã chi nhánh, mã sản phẩm, số lượng tồn, giá bán  ***Output***: Thêm sản phẩm thành công |
| SET TRANSACTION ISOLATION  LEVEL **REPEATEABLE READ** |  |  |  |
| BEGIN TRAN |  |  |  |
| B1: Tính số lượng sản phẩm đã tìm được  SELECT COUNT(\*) AS SL\_TIM\_KIEM  FROM UV\_XEMSP\_KH  WHERE TENSP = @TenSP  \* NOTE: UV\_XEMSP\_KH là một view hiển thị **danh sách sản phẩm** đang được bán. | **S(UV\_XEMSP\_KH)**  //Xin khoá đọc trên view UV\_XEMSP\_KH với điều kiện: theo tên sản phẩm mà KH tìm |  |  |
| WAITFOR DELAY '00:00:05' |  |  |  |
|  |  | BEGIN TRAN |  |
|  |  | B1: Kiểm tra sản phẩm có trong danh sách sản phẩm mà hệ thống cho phép bán  IF @MaSP NOT IN (SELECT MASP  FROM SAN\_PHAM)  BEGIN  PRINT @MaSP + N' không tồn tại!!'  ROLLBACK TRAN  RETURN 1  END  \* NOTE: SAN\_PHAM là bảng **danh sách sản phẩm** mà hệ thống cho phép. | **S(SAN\_PHAM)**  //Xin khoá đọc trên bảng SAN\_PHAM |
|  |  | B2: Kiểm tra chi nhánh  IF @MaCN NOT IN (SELECT MACN  FROM UV\_CN\_DOITAC  WHERE MAHD IS NOT NULL)  BEGIN  PRINT @MaCN + N' không thuộc quản lý của đối tác / chưa được ký hợp đồng!!'  ROLLBACK TRAN  RETURN 1  END  \* NOTE: UV\_CN\_DOITAC là một view hiển thị **danh sách chi nhánh** của chính đối tác đó. | **S(UV\_CN\_DOITAC)**  //Xin khoá đọc trên view UV\_CN\_DOITAC |
|  |  | B3: Thêm sản phẩm  INSERT PHAN\_PHOI  VALUES (@MaSP, @MaCN, @SL\_Ton, @Gia) | **X(PHAN\_PHOI)**  //Xin khoá ghi trên bảng PHAN\_PHOI |
|  |  | COMMIT |  |
| B2: Hiển thị sản phẩm  SELECT \* FROM UV\_XEMSP\_KH  WHERE TENSP = @TenSP | **S(UV\_XEMSP\_KH)**  //Xin khoá đọc trên view UV\_XEMSP\_KH với điều kiện: theo tên sản phẩm mà KH tìm |  |  |
| COMMIT |  |  |  |

## Tình huống 4:

|  |  |  |  |
| --- | --- | --- | --- |
| **ERR04: *Unrepeatable read***  T1 (User = khách hàng): khách hàng kiểm tra thành tiền của sản phẩm (trước khi đặt hàng).  T2 (User = đối tác): thực hiện cập nhật giá cho sản phẩm | | | |
| USP\_KH\_CTDH | **Khóa** | USP\_DT\_THEMSP | **Khóa** |
| ***Input:*** Mã sản phẩm, mã chi nhánh phân phối, số lượng mua  ***Output***: Đơn giá và thành tiền tương ứng | ***Input***: Mã chi nhánh, mã sản phẩm, số lượng tồn, giá bán  ***Output***: Thêm sản phẩm thành công |
| SET TRANSACTION ISOLATION  LEVEL **READ COMMITED** |  | SET TRANSACTION ISOLATION  LEVEL **READ COMMITED** |  |
| BEGIN TRAN |  |  |  |
| B1: Kiểm tra sản phẩm  IF @MaSP NOT IN (SELECT MASP  FROM SAN\_PHAM)  BEGIN  PRINT @MaSP + N' không tồn tại!!'  ROLLBACK TRAN  RETURN 1  END | **S(SAN\_PHAM)**  //Xin khoá đọc trên bảng SAN\_PHAM |  |  |
| B2: Kiểm tra chi nhánh  IF @MaCN NOT IN (SELECT MACN  FROM PHAN\_PHOI)  BEGIN  PRINT @MaCN + N' không tồn tại / chưa ký hợp đồng!!'  ROLLBACK TRAN  RETURN 1  END  \* NOTE: PHAN\_PHOI là bảng danh sách sản phẩm ứng với từng chi nhánh đã được ký hợp đồng mua bán. | **S(PHAN\_PHOI)**  //Xin khoá đọc trên bảng PHAN\_PHOI |  |  |
| B3: Thông tin sản phẩm kèm số lượng và đơn giá  SELECT MASP, TENSP, @SoLuong AS SO\_LUONG\_MUA, GIA  FROM UV\_XEMSP\_KH  WHERE MASP = @MaSP AND MACN = @MaCN  \* NOTE: UV\_XEMSP\_KH là một view hiển thị **danh sách sản phẩm** đang được bán. | **S(UV\_XEMSP\_KH)**  //Xin khoá đọc trên view UV\_XEMSP\_KH với điều kiện theo sản phẩm và chi nhánh |  |  |
| WAITFOR DELAY '00:00:05' |  |  |  |
|  |  | BEGIN TRAN |  |
|  |  | B1: Kiểm tra sản phẩm có trong danh sách sản phẩm mà hệ thống cho phép bán  IF @MaSP NOT IN (SELECT MASP  FROM SAN\_PHAM)  BEGIN  PRINT @MaSP + N' không tồn tại!!'  ROLLBACK TRAN  RETURN 1  END  \* NOTE: SAN\_PHAM là bảng **danh sách sản phẩm** mà hệ thống cho phép. | **S(SAN\_PHAM)**  //Xin khoá đọc trên bảng SAN\_PHAM |
|  |  | B2: Kiểm tra chi nhánh  IF @MaCN NOT IN (SELECT MACN  FROM UV\_CN\_DOITAC  WHERE MAHD IS NOT NULL)  BEGIN  PRINT @MaCN + N' không thuộc quản lý của đối tác / chưa được ký hợp đồng!!'  ROLLBACK TRAN  RETURN 1  END  \* NOTE: UV\_CN\_DOITAC là một view hiển thị **danh sách chi nhánh** của chính đối tác đó. | **S(UV\_CN\_DOITAC)**  //Xin khoá đọc trên view UV\_CN\_DOITAC |
|  |  | B3: Cập nhật giá của sản phẩm  UPDATE UV\_SPDOITAC SET GIA = @GiaMoi WHERE MASP = @MaSP AND MACN = @MaCN | **X(UV\_SPDOITAC)**  //Xin khoá ghi trên bảng UV\_SPDOITAC |
|  |  | COMMIT |  |
| B2: Hiển thị thành tiền của sản phẩm  SELECT GIA \* @SoLuong AS THANH\_TIEN  FROM UV\_XEMSP\_KH  WHERE MASP = @MaSP AND MACN = @MaCN | **S(UV\_XEMSP\_KH)**  //Xin khoá đọc trên view UV\_XEMSP\_KH với điều kiện theo sản phẩm và chi nhánh |  |  |
| COMMIT |  |  |  |

# II. Sinh viên thực hiện: Hoàng Lê Khanh

## 1.     Tình huống 1:

|  |  |  |
| --- | --- | --- |
| **ERROR 01: *Dirty read***  T1 (User = tài xế): **Thêm đơn hàng vào phiếu giao hàng**  T2 (User = tài xế): **Xem đơn hàng đã nhận giao** | | |
| **USP\_THEMPHIEUGIAOHANGTX** | **Khóa** | **USP\_KIEMTRADONHANGTX** | **Khóa** |
| ***Input:***  @MADH CHAR(8),  @NGAYGIAO DATE | ***Input***:  @MADHCANTIM CHAR(8) |  |
| SET TRAN ISOLATION LEVEL READ COMMITTED (DEFAULT) | SET TRAN ISOLATION LEVEL READ UNCOMMITTED |
| BEGIN TRAN |  |  |  |
| B1: Thêm thông tin phiếu giao hàng của tài xế về đơn hàng đã nhận.  INSERT PHIEU\_GIAO\_HANG VALUES ((SELECT TT.MATX FROM UV\_TTTAIXE TT), @MADH, @NGAYGIAO) | S(PHIEU\_GIAO\_HANG)  // Ghi không cần cấp khóa |  |  |
| WAITFOR DELAY '0:0:05' |  |  |  |
|  |  | BEGIN TRAN |  |
|  |  | B1: Kiểm tra thông tin mã đơn hàng có trống không.  IF @MADHCANTIM IS NULL  BEGIN  PRINT N'MÃ ĐƠN HÀNG TÌM KIẾM KHÔNG ĐƯỢC ĐỂ TRỐNG'  ROLLBACK TRAN  RETURN  END |  |
|  |  | B2: Kiểm tra thông tin đơn hàng có thuộc đơn hàng có thể nhận của tài xế không.  IF NOT EXISTS(SELECT \*  FROM UV\_DHTAIXE  WHERE MADH = @MADHCANTIM)  BEGIN  PRINT @MADHCANTIM + N'ĐƠN HÀNG NÀY KHÔNG THUỘC KHU VỰC CỦA TÀI XẾ'  ROLLBACK TRAN  RETURN  END | S(UV\_DHTAIXE)  //Đọc không cần cấp khóa |
|  |  | B3: Tìm thông tin đơn hàng trong danh sách đơn hàng đã nhận.  SELECT \*  FROM UV\_DHTAIXEDANHAN  WHERE MADH = @MADHCANTIM | S(UV\_DHTAIXEDANHAN)  //Đọc không cần cấp khóa |
|  |  | B4: Bắt lỗi hệ thống (nếu có)  BEGIN CATCH  PRINT N'LỖI HỆ THỐNG ' + ERROR\_MESSAGE()  ROLLBACK TRAN  END CATCH |  |
| B2: Kiểm tra thông tin đơn hàng được thêm vào thuộc vào đơn tài xế có thể nhận không.  IF NOT EXISTS(SELECT \*  FROM UV\_DHTAIXE  WHERE MADH = @MADH)  BEGIN  PRINT @MADH + N' ĐƠN HÀNG NÀY KHÔNG THUỘC KHU VỰC CỦA TÀI XẾ'  ROLLBACK TRAN  RETURN 0  END | S(UV\_DHTAIXE)  //Cấp khóa đọc |  |  |
| B3: Kiếm tra thông tin ngày giao hàng  IF @NGAYGIAO IS NULL OR @NGAYGIAO < (SELECT MAX(THOI\_GIAN) FROM TRANG\_THAI\_DON\_HANG WHERE @MADH = MADH)  BEGIN  PRINT N'KIỂM TRA LẠI NGÀY GIAO HÀNG (KHÔNG ĐƯỢC TRỐNG/ NGÀY GIAO KHÔNG HỢP LỆ)'  ROLLBACK TRAN  RETURN 0  END | S(TRANG\_THAI\_DON\_HANG)  //Cấp khóa đọc |  |  |
| B4: Bắt lỗi hệ thống (nếu có)  BEGIN CATCH  PRINT N'LỖI HỆ THỐNG ' + ERROR\_MESSAGE()  ROLLBACK TRAN  END CATCH |  |  |  |
| COMMIT |  | COMMIT |  |

## 2.     Tình huống 2:

|  |  |  |  |
| --- | --- | --- | --- |
| **ERROR 02: *Unrepeatable read***  T1 (User = khách hàng): **Tìm thông tin sản phẩm bằng tên sản phẩm.**  T2 (User = đối tác): **Chỉnh sửa tên sản phẩm.** | | | |
| **USP\_TIMSPTEN** | **Khóa** | **USP\_DOITENSP** | **Khóa** |
| ***Input:***  @TENSP NVARCHAR(100) | ***Input***:  @TENCANCAPNHAT NVARCHAR(100),  @TENCAPNHAT NVARCHAR(100) |
| SET TRAN ISOLATION LEVEL READ COMMITTED (DEFAULT) | SET TRAN ISOLATION LEVEL READ COMMITTED (DEFAULT) |
| BEGIN TRAN |  |  |  |
| B1: Kiểm tra xem sản phẩm có tồn tại trong danh sách sản phẩm ứng với tên cần tìm.  IF NOT EXISTS(SELECT \*  FROM SAN\_PHAM  WHERE TENSP = @TENSP)  BEGIN  PRINT @TENSP + N' KHÔNG TỒN TẠI'  ROLLBACK TRAN  RETURN  END | S(SAN\_PHAM)  //Cấp khóa đọc |  |  |
| WAITFOR DELAY '0:0:05' |  |  |  |
|  |  | BEGIN TRAN |  |
|  |  | B1: Kiểm tra xem sản phẩm có thuộc đối tác không.  IF NOT EXISTS (SELECT \*  FROM UV\_SPDOITAC  WHERE TENSP = @TENCANCAPNHAT)  BEGIN  PRINT  @TENCANCAPNHAT + N'SẢN PHẨM NÀY KHÔNG TỒN TẠI'  ROLLBACK TRAN  RETURN 0  END | S(UV\_SPDOITAC)  //Cấp khóa đọc |
|  |  | B2: Kiểm tra tên cập nhật có trống hay không.  IF @TENCAPNHAT IS NULL  BEGIN  PRINT  N'TÊN CẬP NHẬT KHÔNG ĐƯỢC TRỐNG'  ROLLBACK TRAN  RETURN 0  END |  |
|  |  | B3: Kiểm tra tên cập nhật không được trùng tên trước đó.  IF @TENCAPNHAT = @TENCANCAPNHAT  BEGIN  PRINT  N'TÊN CẬP NHẬT KHÔNG ĐƯỢC TRÙNG VỚI TÊN CŨ'  ROLLBACK TRAN  RETURN 0  END |  |
|  |  | B4: Bắt lỗi hệ thống (nếu có)  UPDATE UV\_SPDOITAC SET TENSP = @TENCAPNHAT WHERE @TENCANCAPNHAT = TENSP | //Không cấp khóa để cập nhật |
| B2: Tìm sản phẩm ứng với tên sản phẩm nhận vào.  SELECT \* FROM SAN\_PHAM WHERE TENSP = @TENSP | S(SAN\_PHAM)  //Cấp khóa đọc | B5: Bắt lỗi hệ thống (nếu có)  BEGIN CATCH  PRINT N‘LỖI HỆ THỐNG ' + ERROR\_MESSAGE()  ROLLBACK TRAN  END CATCH |  |
| B3: Bắt lỗi hệ thống (nếu có)  BEGIN CATCH  PRINT N’LỖI HỆ THỐNG ' + ERROR\_MESSAGE()  ROLLBACK TRAN  END CATCH |  |  |  |
| COMMIT |  | COMMIT |  |

# III. Sinh viên thực hiện: Ngọc Diệu

## 1.     Tình huống 1:

|  |  |  |  |
| --- | --- | --- | --- |
| **ERROR 01: Lost Update**  T1 (User = Khách hàng):  Thực hiện đặt hàng (tạo mới một chi tiết cho đơn hàng) với sản phẩm A tại chi nhánh X  T2 (User = Đối tác): Cập nhật thêm số lượng tồn cho sản phẩm A tại chi nhánh X | | | |
| **USP\_THEMCTDH** | **Khóa** | **USP\_THEMSLSP** | **Khóa** |
| ***Input:***  @MADH CHAR(8),  @MASP CHAR(8),  @MACN CHAR(8),  @SO\_LUONG INT | ***Input***:  @MASP CHAR(8),  @MACN CHAR(8),  @SO\_LUONG INT |
| READ COMMITTED (DEFAULT) | READ COMMITTED (DEFAULT) |
| BEGIN TRAN |  |  |  |
| B1: Kiểm tra sản phẩm có tồn tại trong danh sách hay không  IF NOT EXISTS (SELECT \*  FROM UV\_XEMSP\_KH  WHERE MASP = @MASP AND MACN = @MACN )  BEGIN  PRINT N 'SẢN PHẨM KHÔNG CÓ Ở CHI NHÁNH NÀY'  ROLLBACK TRAN  RETURN 1  END | **S( UV\_XEMSP\_KH)** |  |  |
| B2: Kiểm tra số lượng có hợp lệ không  IF @SO\_LUONG <= 0  BEGIN  PRINT N 'SỐ LƯỢNG PHẢI LỚN HƠN 0'  ROLLBACK TRAN  RETURN 1  END |  |  |  |
| B3: Kiểm tra số lượng tồn có đáp ứng đủ cho số lượng khách đặt không  DECLARE @SLT INT  SET @SLT = (SELECT SO\_LUONG\_TON  FROM PHAN\_PHOI  WHERE MASP = @MASP AND MACN = @MACN)  IF @SLT < @SO\_LUONG  BEGIN  PRINT N'SỐ LƯỢNG TỒN KHÔNG ĐỦ'  ROLLBACK TRAN  RETURN 1  END | S(PHAN\_PHOI) |  |  |
| WAITFOR DELAY '0:0:05' |  |  |  |
|  |  | BEGIN TRAN |  |
|  |  | B1: Kiểm tra sản phẩm có tồn tại trong danh sách hay không  IF NOT EXISTS (SELECT \*  FROM UV\_SPDOITAC  WHERE MASP = @MASP AND MACN = @MACN )  BEGIN  PRINT N 'SẢN PHẨM KHÔNG CÓ Ở CHI NHÁNH NÀY'  ROLLBACK TRAN  RETURN 1  END | **S(UV\_SPDOITAC)** |
|  |  | B2: Kiểm tra số lượng có hợp lệ không  IF @SO\_LUONG <= 0  BEGIN  PRINT N 'SỐ LƯỢNG PHẢI LỚN HƠN 0'  ROLLBACK TRAN  RETURN 1  END |  |
|  |  | B3: Đọc số lượng tồn hiện tại từ bảng PHAN\_PHOI  DECLARE @SLT INT  SET @SLT = (SELECT SO\_LUONG\_TON  FROM PHAN\_PHOI  WHERE MASP = @MASP AND MACN = @MACN) | **S(PHAN\_PHOI)** |
|  |  | WAITFOR DELAY '0:0:05' |  |
| B4: Chèn chi tiết đơn hàng vào dbo.CHI\_TIET\_DON\_HANG  INSERT dbo.CHI\_TIET\_DON\_HANG (MADH, MASP, SO\_LUONG) VALUES (@MADH, @MASP, @SO\_LUONG) | **X(CHI\_TIET\_DON\_HANG)** |  |  |
| B5: Cập nhật số lượng tồn cho bảng PHAN\_PHOI  UPDATE PHAN\_PHOI  SET SO\_LUONG\_TON = @SLT - @SO\_LUONG  WHERE MASP = @MASP AND MACN = @MACN | **X(PHAN\_PHOI)** |  |  |
| B4: Bắt lỗi hệ thống (nếu có)  BEGIN CATCH  PRINT N'LỖI HỆ THỐNG ' + ERROR\_MESSAGE()  ROLLBACK TRAN  RETURN 1  END CATCH |  |  |  |
| COMMIT |  |  |  |
|  |  | B4: Cập nhật số lượng tồn cho bảng PHAN\_PHOI  UPDATE PHAN\_PHOI  SET SO\_LUONG\_TON = @SLT + @SO\_LUONG  WHERE MASP = @MASP AND MACN = @MACN | **X(PHAN\_PHOI)** |
|  |  | B4: Bắt lỗi hệ thống (nếu có)  BEGIN CATCH  PRINT N'LỖI HỆ THỐNG ' + ERROR\_MESSAGE()  ROLLBACK TRAN  RETURN 1  END CATCH |  |
|  |  | COMMIT |  |

## 2.     Tình huống 2:

|  |  |  |  |
| --- | --- | --- | --- |
| **ERROR02: Conversion Deadlock**  T1 (User = Đối tác): Cập nhật thông tin đối tác (sửa tên người đại diện)  T2 (User = Đối tác):  Cập nhật thông tin đối tác (sửa địa chỉ - tên đường) | | | |
| **USP\_DT\_NDD** | **Khóa** | **USP\_DT\_Duong** | **Khóa** |
| ***Input:***  @NGUOI\_DAI\_DIEN nvarchar(30) | ***Input***:  @MASP CHAR(8),  @MACN CHAR(8),  @SO\_LUONG INT |
| SET TRANSACTION ISOLATION  LEVEL **REPEATABLE READ** | SET TRANSACTION ISOLATION  LEVEL **REPEATABLE READ** |
| BEGIN TRAN |  |  |  |
| B1: Kiểm tra người đại diện muốn đổi có trùng với người đại diện hiện tại không  DECLARE @NGUOI\_DAI\_DIEN\_HT nvarchar(30)  SET @NGUOI\_DAI\_DIEN\_HT = (SELECT NGUOI\_DAI\_DIEN  FROM UV\_TTDOITAC)  IF (@NGUOI\_DAI\_DIEN = @NGUOI\_DAI\_DIEN\_HT)  BEGIN  PRINT N'Người đại diện muốn đổi trùng với người đại diện hiện tại’  ROLLBACK TRAN  RETURN  END | **S(UV\_TTDOITAC)**  **//** Xin khóa đọc |  |  |
| WAITFOR DELAY '0:0:05' |  |  |  |
|  |  | BEGIN TRAN |  |
|  |  | B1: Kiểm tra tên đường muốn đổi có trùng với tên đường hiện tại không  DECLARE @DUONG\_HT nvarchar(30)  SET @DUONG\_HT = (SELECT DUONG  FROM UV\_TTDOITAC)  IF (@DUONG = @DUONG\_HT)  BEGIN  PRINT N'Tên đường muốn đổi trùng với tên đường hiện tại'  ROLLBACK TRAN  RETURN  END | **S( UV\_TTDOITAC)**  **//**Xin khóa đọc |
|  |  | WAITFOR DELAY '0:0:05' |  |
| B2: Cập nhật tên người đại diện  UPDATE UV\_TTDOITAC  SET NGUOI\_DAI\_DIEN = @NGUOI\_DAI\_DIEN | **X(UV\_TTDOITAC)**  **//** Xin khóa ghi |  |  |
| B3: Bắt lỗi hệ thống và in ra lỗi (nếu có)  BEGIN CATCH  DECLARE @ErrorMsg VARCHAR(2000)  SELECT @ErrorMsg = N'Lỗi: ' + ERROR\_MESSAGE()  RAISERROR(@ErrorMsg, 16,1)  ROLLBACK TRAN  RETURN  END CATCH |  |  |  |
| COMMIT |  |  |  |
|  |  | B2: Cập nhật tên đường  UPDATE UV\_TTDOITAC  SET DUONG = @DUONG | **X(UV\_TTDOITAC)**  // Xin khóa ghi |
|  |  | B3: Bắt lỗi hệ thống và in ra lỗi (nếu có)  BEGIN CATCH  DECLARE @ErrorMsg VARCHAR(2000)  SELECT @ErrorMsg = N'Lỗi: ' + ERROR\_MESSAGE()  RAISERROR(@ErrorMsg, 16,1)  ROLLBACK TRAN  RETURN  END CATCH |  |
|  |  | COMMIT |  |