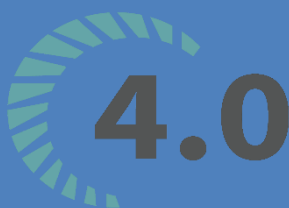


BỘ MÔN HỆ THỐNG THÔNG TIN – KHOA CÔNG NGHỆ THÔNG TIN
ĐẠI HỌC KHOA HỌC TỰ NHIÊN THÀNH PHỐ HỒ CHÍ MINH, ĐẠI HỌC QUỐC GIA TP HCM

MÔN HỆ QUẢN TRỊ CƠ SỞ DỮ LIỆU



Sinh viên thực hiện: 19127366 – Long Mỹ Du (Nhóm trưởng)
19127304 – Trần Khải Trúc
19127649 – Tô Thanh Tuấn

GV phụ trách: Hồ Thị Hoàng Vy
Tiết Gia Hồng

ĐỒ ÁN MÔN HỌC: DA-01 - HỆ QUẢN TRỊ CƠ SỞ DỮ LIỆU
HỌC KỲ I – NĂM HỌC 2021-2022



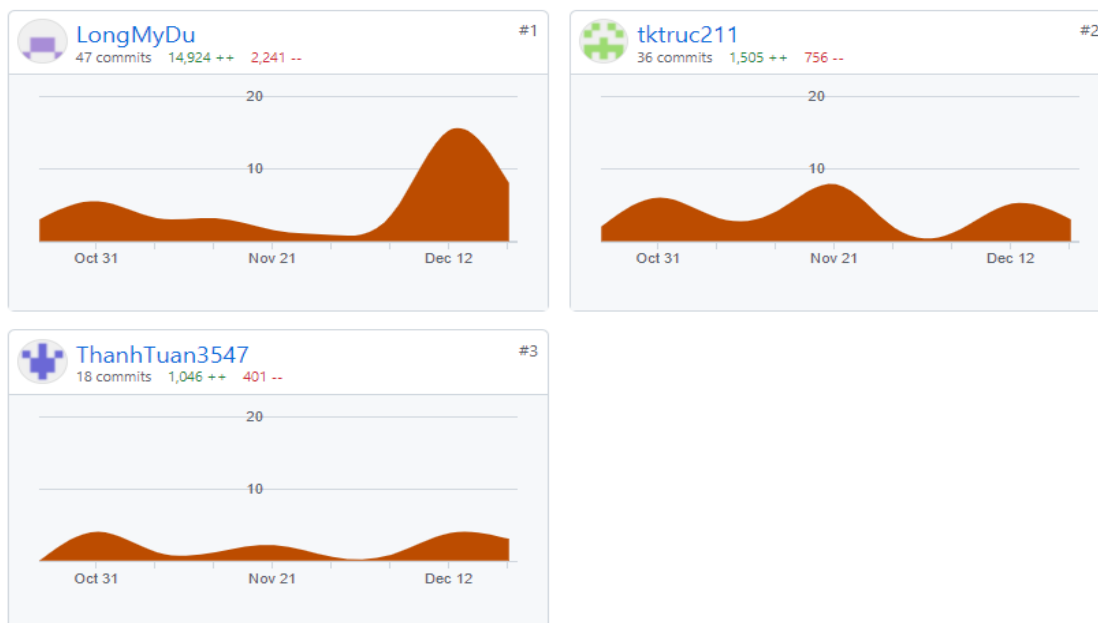
BẢNG THÔNG TIN CHI TIẾT NHÓM

Mã nhóm:	Nhóm 10	
Số lượng:	3	
Github đồ án:	https://github.com/LongMyDu/HQTCSDL-TH-Project01-Group10.git	
MSSV	Họ tên	Email
19127366	Long Mỹ Du	lmdu19@clc.fitus.edu.vn
19127304	Trần Khải Trúc	tktruc19@clc.fitus.edu.vn
19127649	Tô Thanh Tuấn	tttuan19@clc.fitus.edu.vn

Đánh giá cá nhân

MSSV	Họ tên	Thiết kế csdl	Phân quyền	Tranh chấp DT	Xử lý TCDT	Demo + giao diện	Tổng
19127366	Long Mỹ Du	4	1	10	7	12	34
19127304	Trần Khải Trúc	3	2	10	7	12	34
19127649	Tô Thanh Tuấn	3	2	10	6	11	32
Tổng % nhóm đạt							100%

Báo cáo từ github





Bảng phân công & đánh giá hoàn thành công việc			
Công việc thực hiện	Người thực hiện	Mức độ hoàn thành	Đánh giá của nhóm
Thiết kế cơ sở dữ liệu Phân quyền cho admin Viết trigger cho tài khoản Tạo các procedure chức năng Thiết kế giao diện hệ thống GUI Viết báo cáo Tìm và giải quyết tình huống tranh chấp: Dirty Read, Unrepeatable Read, Phantom Read. Cài đặt hệ thống: Front-end	Long Mỹ Du	100%	10/10
Thiết kế cơ sở dữ liệu Phân quyền cho đối tác Viết trigger cho ràng buộc loại tài khoản chi phí đơn hàng Tạo các procedure chức năng Thiết kế giao diện hệ thống GUI Viết báo cáo Tìm và giải quyết tình huống tranh chấp: Lost Update, Dirty Read. Cài đặt hệ thống: Back-end	Trần Khải Trúc	100%	10/10
Thiết kế cơ sở dữ liệu Phân quyền cho khách hàng, nhân viên, tài xế Viết trigger cho đơn hàng Tạo các procedure chức năng Thiết kế giao diện hệ thống GUI Viết báo cáo Tìm và giải quyết tình huống tranh chấp: Unrepeatable Read, Phantom Read. Cài đặt hệ thống: Back-end.	Tô Thanh Tuấn	100%	10/10



YÊU CẦU ĐỒ ÁN- BÀI TẬP

Loại bài tập	<input type="checkbox"/> Lý thuyết <input checked="" type="checkbox"/> Thực hành <input checked="" type="checkbox"/> Đồ án <input type="checkbox"/> Bài tập
Ngày bắt đầu	20/10/2021
Ngày kết thúc	19/12/2021

A. Yêu cầu của đồ án

1. Thiết kế cơ sở dữ liệu
2. Phân quyền người dùng
3. Xác định tình huống tranh chấp
4. Hướng giải quyết
5. Thiết kế giao diện hệ thống (GUI)
6. Cài đặt hệ thống



B.Kết quả

Mục lục

A.	Yêu cầu của đồ án	3
B.	Kết quả	4
I.	Đặc tả cơ sở dữ liệu	5
II.	Mô hình ER	7
III.	Ràng buộc dữ liệu	8
1.	Tài khoản người dùng	8
2.	Sản phẩm thuộc đơn hàng	8
3.	Loại tài khoản	9
4.	Phí sản phẩm	9
IV.	Phân quyền người dùng	10
V.	Tranh chấp đồng thời	12
1.	Lost Update	12
	Tình huống 1	12
	Tình huống 2	12
2.	Dirty Read	18
	Tình huống 1	22
	Tình huống 2	22
3.	Unrepeatable Read	28
	Tình huống 1	29
	Tình huống 2	29
4.	Phantom Read	39
	Tình huống 1	39
	Tình huống 2	44

I. Đặc tả cơ sở dữ liệu

Hệ thống đặt và chuyển hàng online

1. TAIKHOAN (TenTaiKhoan, MatKhu, PhanLoai, TinhTrangKhoa)

Tên từ: Mỗi tài khoản gồm một tên tài khoản duy nhất, có một mật khẩu, thuộc một phân loại người dùng hệ thống. TinhTrangKhoa cho biết tài khoản bị khóa hay không.

2. DOITAC (MaDoiTac, TenDoiTac, Nguo Dai Dien, ThanhPho, Quan, SoChiNhanh, SoDonHangMoiNgay, LoaiHang, DiaChiKinhDoanh, SoDT, Email, TenTaiKhoan)

Tên từ: Mỗi đối tác có một mã số duy nhất, một tên, một người đại diện, có một địa chỉ kinh doanh thuộc một quận của thành phố, kinh doanh một loại hàng, có một số điện thoại, một email và một tài khoản của hệ thống. SoChiNhanh cho biết số lượng chi nhánh của đối tác đó và SoDonHangMoiNgay là số lượng đơn hàng đối tác cần vận chuyển trong mỗi ngày.

3. HOPDONG (MaHopDong, MaSoThue, ThanhToanPhiKichHoat, PhiHoaHong, ThoiGianHieuLuc, SoChiNhanh, TinhTrangDuyet, MaDoiTac)

Tên từ: Mỗi hợp đồng có một mã hợp đồng duy nhất, có một mã số thuế thuộc một đối tác, có thời gian hiệu lực. Phí hoa hồng phải nộp được tính bằng doanh số bán x 10%. Thuộc tính ThanhToanPhiKichHoat cho biết đối tác đã thanh toán phí kích hoạt của hợp đồng hay chưa để xếp vào hàng chờ duyệt. TinhTrangDuyet cho biết nhân viên đã duyệt hợp đồng này hay chưa. SoChiNhanh là số lượng chi nhánh đăng ký trong hợp đồng.

4. CHINHANH (MaChiNhanh, DiaChi, MaHopDong)

Tên từ: Mỗi chi nhánh có một mã số duy nhất, có một địa chỉ và thuộc về một hợp đồng.

5. SANPHAM (MaSP, TenSP, Gia, SoLuongTon, MaChiNhanh)

Tên từ: Mỗi sản phẩm có một mã sản phẩm duy nhất, có một tên của sản phẩm, có một giá và được cung cấp bởi một chi nhánh. Có số lượng tồn của sản phẩm được cập nhật sau khi bán.

6. KHACHHANG (MaKH, HoTen, SoDT, DiaChi, Email, TenTaiKhoan)

Tên từ: Mỗi khách hàng có một mã số duy nhất, có một họ tên, một số điện thoại, một địa chỉ, một email và có một tài khoản của hệ thống.

7. TAIXE (MaTaiXe, HoTen, CMND, SoDT, DiaChi, BienSoXe, KhuVucHoatDong, Email, TaiKhoanNganHang, TenTaiKhoan)

Tên từ: Mỗi tài xế có một mã số duy nhất, có một họ tên, một chứng minh nhân dân, một số điện thoại, một địa chỉ, một biển số xe, có một khu vực hoạt động, một email, một tài khoản ngân hàng, và có một tài khoản của hệ thống.

8. DONHANG (MaDonHang, NgayLap, HinhThucThanhToan, DiaChiGiaoHang, PhiSP, PhiVC, TinhTrangVanChuyen, MaKH, MaChiNhanh, MaTaiXe)

Tên từ: Mỗi đơn hàng có một mã đơn hàng duy nhất, có ngày lập đơn hàng, một địa chỉ giao đơn hàng, có một hình thức để thanh toán đơn hàng. PhiSP và PhiVC là tiền mua sản phẩm và chi phí vận chuyển. TinhTrangVanChuyen được tài xế cập nhật để khách hàng theo dõi. Đơn hàng của một khách hàng được cung cấp bởi một chi nhánh và được một tài xế vận chuyển.

9. CHITIETDONHANG (MaSP, MaDonHang, SoLuong, Gia)

Tên từ: Mỗi dòng của chi tiết đơn hàng thể hiện số lượng và giá của một sản phẩm thuộc một đơn hàng.

10. NHANVIEN (MaNhanVien, HoTen, TenTaiKhoan)

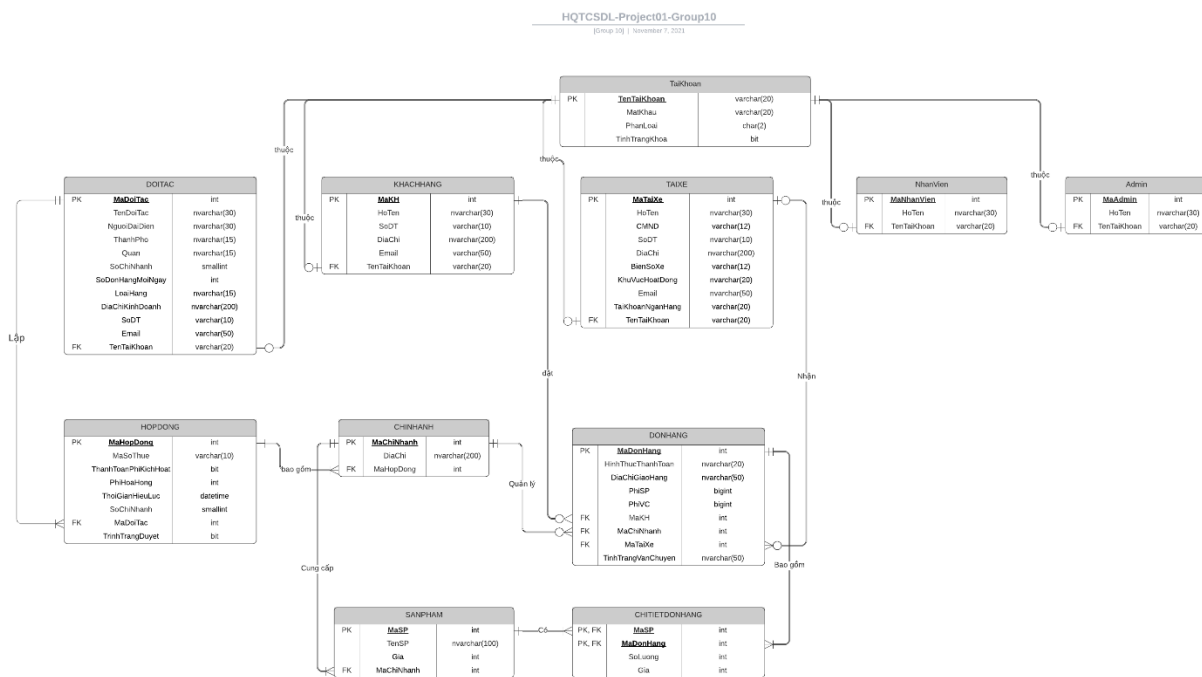
Tên từ: Mỗi nhân viên có một mã nhân viên duy nhất, một họ tên và có một tài khoản thuộc hệ thống.

11. ADMIN (MaAdmin, HoTen, TenTaiKhoan)

Tên từ: Mỗi admin có một mã số duy nhất, có một họ tên và một tài khoản của hệ thống.



II. Mô hình ER



III. Ràng buộc dữ liệu

1. Tài khoản người dùng

a. Ràng buộc: Mỗi tài khoản chỉ được thuộc về một người dùng

Bảng tầm ảnh hưởng:

Bảng	Thêm	Xóa	Sửa
Doitac	+	-	+(TenTaiKhoan)
TaiXe	+	-	+(TenTaiKhoan)
KhachHang	+	-	+(TenTaiKhoan)
NhanVien	+	-	+(TenTaiKhoan)
Admin	+	-	+(TenTaiKhoan)

b. Ràng buộc: Mỗi tài khoản phải cùng loại với người dùng sử dụng tài khoản đó

Bảng tầm ảnh hưởng:

Bảng	Thêm	Xóa	Sửa
TaiKhoan	-	-	+(PhanLoai)
Doitac	+	-	+(TenTaiKhoan)
TaiXe	+	-	+(TenTaiKhoan)
KhachHang	+	-	+(TenTaiKhoan)
NhanVien	+	-	+(TenTaiKhoan)
Admin	+	-	+(TenTaiKhoan)

2. Sản phẩm thuộc đơn hàng

Ràng buộc: Mỗi đơn hàng chỉ được có các sản phẩm thuộc cùng một chi nhánh

Bảng tầm ảnh hưởng:

Bảng	Thêm	Xóa	Sửa
DonHang	-	-	+(MaChiNhanh)
ChiTietDonHang	+	-	+(MaDonHang, MaSP)

3. Loại tài khoản

Mô tả:

Phân loại tài khoản trong bảng TaiKhoan bao gồm: DT (đối tác), KH (khách hàng), TX (tài xế), NV (nhân viên), AD (admin)

Nội dung:

Thêm Constraint CK_TaiKhoan_PhanLoai trên bảng TaiKhoan

4. Phí sản phẩm

Ràng buộc:

Phí sản phẩm (PhiSP) trong bảng đơn hàng (DonHang) bằng tổng giá trị các sản phẩm thuộc đơn hàng đó trong bảng chi tiết đơn hàng (ChiTietDonHang)

Bảng tầm ảnh hưởng:

Bảng	Thêm	Xóa	Sửa
ChiTietDonHang	+	+	+(SoLuong,Gia)



IV. Phân quyền người dùng

Vai trò Thực thể	Đối tác	Khách Hàng	Tài Xế	Nhân Viên	Admin
TAIKHOAN	Xem (TenTaiKhoan, TinhTrangKhoa) Sửa (TenTaiKhoan, MatKhou)	Xem (TenTaiKhoan, TinhTrangKhoa) Sửa (TenTaiKhoan, MatKhou)	Xem (TenTaiKhoan, TinhTrangKhoa) Sửa (TenTaiKhoan, MatKhou)	Xem (TenTaiKhoan, TinhTrangKhoa) Sửa (TenTaiKhoan, MatKhou)	Xem Thêm Xóa Sửa Cấp quyền
DOITAC	Xem Sửa	Xem (TenDoiTac, Thanhpho, Quan, LoaiHang, SoDT, DiaChiKinhDoanh)	x	Xem	x
HOPDONG	Xem Thêm	x	x	Xem Sửa (PhiHoaHong, ThoiGianHieuLuc TinhTrangDuyet)	x
CHINHANH	Xem Thêm Xóa Sửa (DiaChi)	Xem (DiaChi)	x	x	x
KHACHHANG	x	Xem Sửa (Hoten, Sodt, Diachi, Email)	Xem (MaKH, HoTen, SoDT)	x	x



DONHANG	Xem Sửa (TinhTrangVanC huyen)	Xem Thêm Sửa (HinhThucThanh Toan, DiaChiGiaoHang, MaChiNhanh)	Xem Sửa (TinhTrangVanC huyen)	x	x
CHITIETDON HANG	Xem	Xem Thêm Xóa Sửa (MaSP, SoLuong)	x	x	x
SANPHAM	Xem Thêm Xóa Sửa	Xem (MaSp, Tensp, Gia, MaChiNhanh)	x	x	x
TAIXE	x	x	Xem Sửa	x	x
NHANVIEN	x	x	x	x	Xem Thêm Xóa Sửa Cấp quyền
ADMIN	x	x	x	x	Xem Thêm Xóa Sửa Cấp quyền

V. Tranh chấp đồng thời

1. Lost Update

Tình huống 1

Transaction 1: Bán được 3 sản phẩm '100001' và cập nhật lượng tồn của sản phẩm.

Transaction 2: Thêm vào 5 sản phẩm '100001' và cập nhật lượng tồn của sản phẩm.

Transaction 1	Transaction 2	Giải thích
Set @soluongton = (Select SoLuongTon from SanPham where MaSP = @masp)		T1 lấy số lượng tồn của sản phẩm '100001'.
	Set @soluongton = (Select SoLuongTon from SanPham where MaSP = @masp)	T2 lấy số lượng tồn của sản phẩm '100001'.
	Set @soluongton = @soluongton + @soluongthem	T2 thêm vào 5 sản phẩm.
Set @soluongton = @soluongton - @soluongban		T1 bán được 3 sản phẩm.
	Update SanPham Set SoLuongTon = @soluongton Where MaSP = @masp	T2 cập nhật số lượng tồn của sản phẩm.
Update SanPham Set SoLuongTon = @soluongton Where MaSP = @masp		T1 cập nhật số lượng tồn của sản phẩm.
	Commit	
Commit		

Giải thích

Xem số lượng tồn ban đầu của sản phẩm '100001' là 687.

```
USE DB_QLDatChuyenHang
GO

Select * from SANPHAM
```

	MaSP	TenSP	Gia	SoLuongTon	MaChiNhanh
5	100000	Sweater 211	61000	897	100044
6	100001	Dao rọc giấy 914	20000	687	100030
7	100002	USB siêu nhỏ 286	86000	953	100004

- Chạy transaction 1 trước và chạy transaction 2 ngay sau đó.
- Ta thấy transaction 2 sẽ hoàn thành trước và số lượng tồn lúc này là 692 ($= 687 + 5$)
- Sau đó vài giây, transaction 1 hoàn thành và kết quả số lượng tồn là 684 ($= 687 - 3$)

DB_QLDatChuyenHang

Execute

Tran_1_LostUpdate...03SAKS\tktru (52)

-- Lost Update
-- Transaction 1: Bán được 3 sản phẩm '100001' và cập nhật số l

USE DB_QLDatChuyenHang
GO

exec dbo.Ban_SanPham_SoLuong '100001', '3';

Select * from SANPHAM

100 %

Results

Messages

	MaSP	TenSP	Gia	SoLuongTon	MaChiNhanh
5	100000	Sweater 211	61000	897	100044
6	100001	Dao rọc giấy 914	20000	684	100030
7	100002	USB siêu nhỏ 286	86000	953	100004

Tran_2_LostUpdate...03SAKS\tktru (54)

-- Lost Update
-- Transaction 2: Thêm vào 5 sản phẩm '100001' và cập nhật số lư

USE DB_QLDatChuyenHang
GO

exec dbo.Them_SanPham_SoLuong '100001', '5';

Select * from SANPHAM

100 %

Results

Messages

	MaSP	TenSP	Gia	SoLuongTon	MaChiNhanh
5	100000	Sweater 211	61000	897	100044
6	100001	Dao rọc giấy 914	20000	692	100030
7	100002	USB siêu nhỏ 286	86000	953	100004

- Tuy nhiên, kết quả đúng phải là $687 + 5 - 3 = 689$
- Do transaction 1 không đọc dữ liệu cập nhật từ transaction 2 → **Lost Update**



Demo lỗi trên giao diện

- Xem số lượng tồn ban đầu của sản phẩm '100001' là 678.

Quản lý sản phẩm

Lọc theo	Mã SP	Tên SP	Giá	SL Tồn	Mã CN
Chi nhánh 100030	100001	Dao rọc giấy 914	20000	678	100030
Lọc	100085	Dây chuyền 988	29000	810	100030

- Transaction 1 (bên trái) đặt hàng 3 sản phẩm.
- Transaction 2 (bên phải) thêm vào 5 số lượng của sản phẩm.

Mã SP: 100001 Tên SP: Dao rọc giấy 914 Giá: 20000 Mã CN: 100030 Số lượng: 3

Tổng tiền: 60000

Địa chỉ: 21/1 Nguyễn Thái Học Q1

Hình thức thanh toán: Tiền mặt, Momo, Visa

Đặt hàng

Cập nhật sản phẩm

Mã SP: 100001 Tên sản phẩm: Dao rọc giấy 914

Mã SP: 100001 Số lượng tồn tăng thêm: 5

Mã SP: Giá bán mới:

Giảm giá: % cho tất cả sản phẩm

Thêm sản phẩm

Tên sản phẩm: Giá: SL tồn: Mã chi nhánh:

Xác nhận

- Transaction 1 (bên trái) cập nhật số lượng tồn là 675.
- Transaction 2 (bên phải) cập nhật số lượng tồn là 683.

Danh sách sản phẩm					
Lọc theo	Mã SP	Tên SP	Giá	SL Tồn	Mã CN
Đối tác	100001	Dao rọc giấy 914	20000	675	100030
Chi nhánh	100085	Dây chuyền 988	29000	810	100030
Giá					
Lọc					

Quản lý sản phẩm					
Lọc theo	Mã SP	Tên SP	Giá	SL Tồn	Mã CN
Chi nhánh	100001	Dao rọc giấy 914	20000	683	100030
Lọc	100085	Dây chuyền 988	29000	810	100030



Giải quyết tranh chấp

Thiết lập mức cô lập cho cả 2 transaction là REPEATABLE READ → Dẫn tới **Deadlock**

Transaction 1	Transaction 2	Giải thích
Set @soluongton = (Select SoLuongTon from SanPham where MaSP = @masp)		T1 xin cấp S-lock trên bảng SanPham. SQL cấp S-lock trên bảng SanPham cho T1.
	Set @soluongton = (Select SoLuongTon from SanPham where MaSP = @masp)	T2 xin cấp S-lock trên bảng SanPham. SQL cấp S-lock trên bảng SanPham cho T2.
	Set @soluongton = @soluongton + @soluongthem	
Set @soluongton = @soluongton - @soluongban		
	Update SanPham Set SoLuongTon = @soluongton Where MaSP = @masp (CHỜ)	T2 xin cấp X-lock trên bảng SanPham. SQL không cấp X-lock trên bảng SanPham cho T2 vì T1 đang giữ S-lock.
Update SanPham Set SoLuongTon = @soluongton Where MaSP = @masp (CHỜ)		T1 xin cấp X-lock trên bảng SanPham. SQL không cấp X-lock trên bảng SanPham cho T1 vì T2 đang giữ S-lock.
		T1 chờ T2 trả S-lock, T2 chờ T1 trả S-lock. → Deadlock



Giải quyết deadlock

Thiết lập mức cô lập cho cả 2 transaction là REPEATABLE READ và sử dụng UPDATE LOCK khi đọc.

Transaction 1	Transaction 2	Giải thích
Set @soluongton = (Select SoLuongTon from SanPham with (Updlock) where MaSP = @masp)		T1 xin cấp U-lock trên bảng SanPham. SQL cấp U-lock cho T1.
	Set @soluongton = (Select SoLuongTon from SanPham with (Updlock) where MaSP = @masp) (CHỜ)	T2 xin cấp U-lock trên bảng SanPham. SQL không cấp U-lock cho T2 vì T1 đang giữ U-lock.
Set @soluongton = @soluongton - @soluongban		
Update SanPham Set SoLuongTon = @soluongton Where MaSP = @masp		T1 xin cấp X-lock trên bảng SanPham. SQL cấp X-lock cho T1.
commit tran		T1 trả khóa X-lock trên bảng SanPham.
	Set @soluongton = (Select SoLuongTon from SanPham with (Updlock) where MaSP = @masp)	T2 xin cấp U-lock trên bảng SanPham. SQL cấp U-lock cho T2.
	Set @soluongton = @soluongton + @soluongthem	
	Update SanPham Set SoLuongTon = @soluongton Where MaSP = @masp	T2 xin cấp X-lock trên bảng SanPham. SQL cấp X-lock cho T2.
	commit tran	

Tình huống 2

Transaction 1: Thay đổi giá bán của sản phẩm '100066' thành 120.000đ.

Transaction 2: Giảm giá 10% tất cả sản phẩm của chi nhánh '100043'.

Transaction 1	Transaction 2	Giải thích
Set @gia = (Select Gia from SanPham where MaSP = @masp)		T1 lấy giá của sản phẩm '100066'
	Set @gia = (Select Gia from SanPham where MaChiNhanh = @macn)	T2 lấy giá của các sản phẩm thuộc chi nhánh '100043'
	Set @gia = @gia * (100% - @giamgia%)	T2 giảm giá 10% các sản phẩm của chi nhánh
Set @gia = @giamoi		T1 đổi giá của sản phẩm '100066'
	Update SanPham Set Gia = @gia Where MaChiNhanh = @macn	T2 cập nhật giá của các sản phẩm thuộc chi nhánh
Update SanPham Set Gia = @gia Where MaSP = @masp		T1 cập nhật giá của sản phẩm '100066'
	Commit	
Commit		



- Xem giá ban đầu của sản phẩm '100066' là 100.000đ.

```
USE DB_QLDatChuyenHang
GO

Select * from SANPHAM
```

	MaSP	TenSP	Gia	SoLuongTon	MaChiNhanh
70	100065	Kính cường lực 148	73000	475	100038
71	100066	Nhẫn bạc 516	100000	174	100043
72	100067	Gương mini 439	92000	894	100002

- Chạy transaction 1 trước và chạy transaction 2 ngay sau đó.
- Ta thấy transaction 2 sẽ hoàn thành trước và kết quả giá là 90.000 (giảm giá 10% với 100.000).
- Sau đó vài giây, transaction 1 hoàn thành và kết quả giá là 120.000đ.

```
-- Transaction 1: Thay đổi giá bán của sản phẩm '100066'

USE DB_QLDatChuyenHang
GO

exec dbo.CapNhat_SANPHAM_gia '100066', '120000';

Select * from SANPHAM
```

	MaSP	TenSP	Gia	SoLuongTon	MaChiNhanh
70	100065	Kính cường lực 148	73000	475	100038
71	100066	Nhẫn bạc 516	120000	174	100043
72	100067	Gương mini 439	92000	894	100002

```
-- Transaction 2: Giảm giá 10% tất cả sản phẩm của chi nhánh 100043

USE DB_QLDatChuyenHang
GO

exec dbo.CapNhat_SANPHAM_GiamGiaDongLoat '100043', '10';

Select * from SANPHAM
Where MaChiNhanh = '100043'
```

	MaSP	TenSP	Gia	SoLuongTon	MaChiNhanh
1	100066	Nhẫn bạc 516	90000	174	100043
2	100087	Kính cường lực 592	26730	24	100043

- Do cả 2 transaction không đọc dữ liệu cập nhật của nhau → **Lost Update**.



Demo lỗi trên giao diện

- Xem giá ban đầu của sản phẩm '100066' là 100.000đ.

Quản lý sản phẩm

Lọc theo	Mã SP	Tên SP	Giá	SL Tồn	Mã CN
Chi nhánh 100043	100066	Nhẫn bạc 516	100000	174	100043
Lọc	100087	Kính cường lực 592	17536	24	100043

- Transaction 1 (bên trên) thay đổi giá bán của sản phẩm '100066' thành 120.000đ.
- Transaction 2 (bên dưới) giảm giá 10% cho toàn bộ sản phẩm thuộc chi nhánh '100043'.

Mã SP:	100066	Giá bán mới:	120000	Thay đổi giá bán
Giảm giá		% cho tất cả sản phẩm		Giảm giá
Mã SP:		Giá bán mới:		Thay đổi giá bán
Giảm giá	10	% cho tất cả sản phẩm		Giảm giá

- Transaction 1 (bên trên) cập nhật giá sản phẩm là 120.000đ.
- Transaction 2 (bên dưới) cập nhật giá sản phẩm là 90.000đ (giảm giá 10%).

Lọc theo	Mã SP	Tên SP	Giá	SL Tồn	Mã CN
Chi nhánh 100043	100066	Nhẫn bạc 516	120000	174	100043
Lọc	100087	Kính cường lực 592	15782	24	100043

Lọc theo	Mã SP	Tên SP	Giá	SL Tồn	Mã CN
Chi nhánh 100043	100066	Nhẫn bạc 516	90000	174	100043
Lọc	100087	Kính cường lực 592	15782	24	100043

Giải quyết tranh chấp

Thiết lập mức cô lập cho cả 2 transaction là REPEATABLE READ → Dẫn tới **Deadlock**

Transaction 1	Transaction 2	Giải thích
Set @gia = (Select Gia from SanPham where MaSP = @masp)		T1 xin cấp S-lock trên bảng SanPham SQL cấp S-lock cho T1.
	Set @gia = (Select Gia from SanPham where MaChiNhanh = @macn)	T2 xin cấp S-lock trên bảng SanPham SQL cấp S-lock cho T2.
	Set @gia = @gia * (100% - @giamgia%)	
Set @gia = @giamoi		
	Update SanPham Set Gia = @gia Where MaChiNhanh = @macn (CHỜ)	T2 xin cấp X-lock trên bảng SanPham. SQL không cấp X-lock cho T2 vì T1 đang giữ S-lock.
Update SanPham Set Gia = @gia Where MaSP = @masp (CHỜ)		T1 xin cấp X-lock trên bảng SanPham. SQL không cấp X-lock cho T1 vì T đang giữ S-lock.
		T1 chờ T2 trả S-lock, T2 chờ T1 trả S-lock → Deadlock



Giải quyết deadlock

Thiết lập mức cô lập cho cả 2 transaction là REPEATABLE READ và sử dụng UPDATE LOCK khi đọc.

Transaction 1	Transaction 2	Giải thích
Set @gia = (Select Gia from SanPham with (Updlock) where MaSP = @masp)		T1 xin cấp U-lock trên bảng SanPham SQL cấp U-lock trên bảng SanPham cho T1
	Set @gia = (Select Gia from SanPham with (Updlock) where MaChiNhanh = @macn) (CHỜ)	T2 xin cấp U-lock trên bảng SanPham SQL không cấp U-lock cho T2 vì T1 đang giữ U-lock
Set @gia = @giamoi		
Update SanPham Set Gia = @gia Where MaSP = @masp		T1 xin cấp X-lock trên bảng SanPham SQL cấp X-lock trên bảng SanPham cho T1
Commit		T1 trả khóa X-lock trên bảng SanPham
	Set @gia = (Select Gia from SanPham with (Updlock) where MaChiNhanh = @macn)	T2 xin cấp U-lock trên bảng SanPham SQL cấp U-lock trên bảng SanPham cho T2
	Set @gia = @gia * (100% - @giamgia%)	
	Update SanPham Set Gia = @gia Where MaChiNhanh = @macn	T2 xin cấp X-lock trên bảng SanPham SQL cấp X-lock trên bảng SanPham cho T2
	Commit	

2. Dirty Read

Tình huống 1

Transaction 1: Bán 8 sản phẩm '100028', sau đó bị hủy do lỗi số lượng tồn.

Transaction 2: Xem sản phẩm '100028' thuộc chi nhánh '100023'.

Transaction 1	Transaction 2	Giải thích
Set @soluongton = (Select SoLuongTon from SanPham where MaSP = @masp)		T1 lấy số lượng tồn của sản phẩm '100028'
Set @soluongton = @soluongton - @soluongban		T1 bán 8 sản phẩm '100028'
Update SanPham Set SoLuongTon = @soluongton Where MaSP = @masp		T1 cập nhật số lượng tồn của sản phẩm '100028'
	Select * from SanPham where MaSP = @masp	T2 xem thông tin sản phẩm '100028'
Rollback		T1 lỗi khi lượng tồn < 0, hủy cập nhật

- Xem số lượng tồn ban đầu của sản phẩm '100028' là 6.

```
USE DB_QLDatChuyenHang
GO

Select * from SANPHAM
```

	MaSP	TenSP	Gia	SoLuongTon	MaChiNhanh
32	100027	Gương mini 743	6000	91	100028
33	100028	Khay làm đá 047	73000	6	100023
34	100029	Đồ qot 565	24000	95	100018



- Chạy transaction 1 và cập nhật số lượng tồn của sản phẩm.
- Sau đó, chạy transaction 2 khi transaction 1 chưa thực hiện commit. Transaction 2 đọc thông tin sản phẩm và kết quả số lượng tồn của sản phẩm lúc này là -2 ($= 6 - 8$).
- Lúc này, transaction 1 kiểm tra số lượng tồn không hợp lệ nên trả về lỗi và rollback.

```
Tran_1_DirtyRead.s...C03SAKS\tktru (54) X
-- Dirty Read
-- Transaction 1: Bán 8 sản phẩm '100028', sau đó bị hủy do lỗi số lượng

USE DB_QLDatChuyenHang
GO

exec dbo.Ban_SanPham_SoLuong '100028', '8';

Select * from SANPHAM
```

100 %

Results Messages

(1 row affected)

Msg 50000, Level 16, State 1, Procedure Ban_SanPham_SoLuong, Line 43 [Batch Start Line 5]
Sản phẩm không đủ số lượng.

Msg 3902, Level 16, State 1, Procedure Ban_SanPham_SoLuong, Line 48 [Batch Start Line 5]
The COMMIT TRANSACTION request has no corresponding BEGIN TRANSACTION.

(104 rows affected)

- Vì transaction 1 đã rollback nên số lượng tồn vẫn giữ nguyên ban đầu là 6.
- Tuy nhiên transaction 2 đã đọc số lượng tồn trước đó nên cho ra kết quả sai

→ Dirty Read

Tran_1_DirtyRead.s...C03SAKS\tktru (54)						Tran_2_DirtyRead.s...C03SAKS\tktru (55)					
-- Dirty Read -- Transaction 1: Bán 8 sản phẩm '100028', sau đó						-- Dirty Read -- Transaction 2: Xem thông tin sản phẩm '100028'					
USE DB_QLDatChuyenHang GO						USE DB_QLDatChuyenHang GO					
exec dbo.Ban_SanPham_SoLuong '100028', '8';						exec dbo.XemTatCa_SANPHAM_ThuocChiNhanh '100023';					
Select * from SANPHAM											
Results Messages						Results Messages					
MaSP	TenSP	Gia	SoLuongTon	MaChiNhanh		MaSP	TenSP	Gia	SoLuongTon	MaChiNhanh	
32	100027	Gương mini 743	6000	91	100028	1	100021	Gương mini 140	54000	329	100023
33	100028	Khay làm đá 047	73000	6	100023	2	100028	Khay làm đá 047	73000	-2	100023
34	100029	Đồ gọt 565	24000	95	100018	3	100098	Mô hình trang trí 93	46000	510	100023

Demo lỗi trên giao diện

- Xem số lượng tồn ban đầu của sản phẩm '100028' là 6.

Danh sách sản phẩm

Lọc theo	Mã SP	Tên SP	Giá	SL Tồn	Mã CN
Đối tác	100021	Gương mini 140	54000	329	100023
Chi nhánh	100028	Khay làm đá 047	73000	6	100023
Giá	100098	Mô hình trang trí 93	46000	510	100023

- Transaction 1 (bên trái) đặt hàng và cập nhật số lượng tồn sản phẩm không hợp lệ.
- Transaction 2 (bên phải) xem sản phẩm và đọc được số lượng tồn không hợp lệ.

100028 Khay làm đá 047 73000 100023 0

Tổng tiền: 584000

Địa chỉ: 21/1 Nguyễn

Hình thức thanh toán: Tiền mặt, Momo, Visa

Đặt hàng

localhost:8080
Đơn hàng không hợp lệ
RequestError: The COMMIT TRANSACTION request has no corresponding BEGIN TRANSACTION.

Quản lý sản phẩm

Lọc theo	Mã SP	Tên SP	Giá	SL Tồn	Mã CN
Chi nhánh	100021	Gương mini 140	54000	329	100023
Lọc	100028	Khay làm đá 047	73000	-2	100023
	100098	Mô hình trang trí 93	46000	510	100023

- Sau khi hiển thị thông báo lỗi, transaction 1 rollback nên số lượng tồn không cập nhật.

Lọc theo	Mã SP	Tên SP	Giá	SL Tồn	Mã CN
Đối tác	100021	Gương mini 140	54000	329	100023
Chi nhánh	100028	Khay làm đá 047	73000	6	100023
Giá	100098	Mô hình trang trí 93	46000	510	100023

Lọc

localhost:8080
Đơn hàng không hợp lệ
RequestError: The COMMIT TRANSACTION request has no corresponding BEGIN TRANSACTION.



Giải quyết tranh chấp

Thiết lập mức cô lập READ COMMITTED cho transaction 2.

Transaction 1	Transaction 2	Giải thích
Set @soluongton = (Select SoLuongTon from SanPham where MaSP = @masp)		
Set @soluongton = @soluongton - @soluongban		
Update SanPham Set SoLuongTon = @soluongton Where MaSP = @masp		T1 xin cấp khóa X-lock trên bảng SanPham. SQL cấp khóa X-lock cho T1.
	Select * from SanPham where MaSP = @masp (CHỜ)	T2 xin cấp khóa S-lock trên bảng SanPham. SQL không cấp S-lock cho T2 vì T1 đang giữ X-lock
Rollback		
	Select * from SanPham where MaSP = @masp	T1 đã trả X-lock. SQL cấp S-lock cho T2.
	Commit	

Tình huống 2

Transaction 1: Giảm giá 10% cho tất cả sản phẩm được cung cấp bởi chi nhánh 1.

Transaction 2: Đọc thông tin các sản phẩm được cung cấp bởi chi nhánh 1.

Transaction 1	Transaction 2
<pre>update SANPHAM set Gia = convert(bigint, Gia * (100-10)/100) where MaChiNhanh = 1</pre>	
	<pre>Select * from SANPHAM Where MaChiNhanh = 1</pre>
	<pre>commit tran</pre>
<pre>raiserror('Đã xảy ra sự cố', 16, 1); rollback tran</pre>	

Giải thích:

Dữ liệu ban đầu:

Results

Messages

	MaSP	TenSP	Gia	SoLuongTon	MaChiNhanh
1	1	Hạt hạnh nhân hữu cơ 1kg	450000	100	1
2	2	Khô Heo Cháy Tỏi DTFood Đặc Biệt Thơm Ngon	85000	100	1
3	3	Thùng 20 gói Mì Rong Biển Ottogi 120gx20	250000	200	1
4	4	Mì Trộn Xốt Tương Đen Hàn Quốc Ottogi 135Gr	35000	50	1

Thực thi transaction 1 giảm giá 10% các sản phẩm thuộc chi nhánh 1. Sau đó, vì xảy ra sự cố nên transaction 1 bị rollback, và giá của các sản phẩm được giữ nguyên. Thế nhưng, trước khi transaction 1 rollback, transaction 2 đọc giá đã được giảm của các sản phẩm thuộc chi nhánh 1:



Results		Messages			
	MaSP	TenSP	Gia	SoLuongTon	MaChiNhanh
1	1	Hạt hạnh nhân hữu cơ	405000	110	1
2	2	Khô Heo Cháy Tỏi DTFood Đặc Biệt Thơm Ngon	76500	100	1
3	3	Thùng 20 gói Mì Rong Biển Ottogi 120gx20	225000	200	1
4	4	Mì Trộn Xốt Tương Đen Hàn Quốc Ottogi 135Gr	31500	50	1

Demo lỗi trên giao diện:

Thực hiện giảm giá 10% các sản phẩm của chi nhánh 1:

Cập nhật sản phẩm

Mã SP: Tên sản phẩm:

Mã SP: Số lượng tồn tăng thêm:

Mã SP: Giá bán mới:

Giảm giá % cho tất cả sản phẩm

localhost:8080 says

Giảm giá chi nhánh thất bại

Mặc dù việc thực hiện giảm giá thất bại, nhưng khi transaction 2 xem danh sách sản phẩm của chi nhánh 1 thì giá của sản phẩm vẫn là giá đã giảm:

Danh sách sản phẩm

Lọc theo		Mã SP	Tên SP	Giá	SL Tồn	Mã CN
Đối tác	<input type="text"/>	1	Hạt hạnh nhân hữu cơ 1kg	405000	100	1 <input data-bbox="1409 1375 1430 1402" type="button" value="+"/>
Chi nhánh	<input type="text" value="1"/>	2	Khô Heo Cháy Tỏi DTFood Đặc Biệt Thơm Ngon	76500	91	1 <input data-bbox="1409 1438 1430 1465" type="button" value="+"/>
Giá	<input type="text"/>	3	Thùng 20 gói Mì Rong Biển Ottogi 120gx20	225000	198	1 <input data-bbox="1409 1501 1430 1528" type="button" value="+"/>
<input type="button" value="Lọc"/>		4	Mì Trộn Xốt Tương Đen Hàn Quốc Ottogi 135Gr	31500	50	1 <input data-bbox="1409 1564 1430 1591" type="button" value="+"/>



Giải quyết tranh chấp:

Thiết lập mức cô lập READ COMMITTED cho transaction 2.

Transaction 1	Transaction 2	Giải thích
<pre>update SANPHAM set Gia = convert(bigint, Gia * (100-10)/100) where MaChiNhanh = 1</pre>		T1 xin cấp khóa X-lock trên bảng SANPHAM. SQL: cấp khóa X-lock trên bảng SANPHAM cho T1.
	<pre>Select * from SANPHAM Where MaChiNhanh = 1 (CHỜ)</pre>	T2 xin cấp khóa S-lock trên bảng SANPHAM. SQL: không cấp S-lock cho T2 vì T1 đang giữ X-lock.
<pre>raiserror('Đã xảy ra sự cố', 16, 1); rollback tran</pre>		
	<pre>Select * from SANPHAM Where MaChiNhanh = 1</pre>	T1 đã trả X-lock, SQL cấp S-lock cho T2.
	<pre>commit tran</pre>	

3. Unrepeatable Read

Tình huống 1

Transaction 1: Đăng nhập vào tài khoản: '_hopeful' mật khẩu: 'tOxd4cLQkFyI'

Transaction 2 : Đổi mật khẩu tài khoản : '_hopeful' thành '123456'

Transaction 1	Transaction 2
<pre> declare @MK varchar(20), @TK varchar(20) select @TK = '_hopeful', @MK = 'tOxd4cLQkFyI' if (LEN(@TK) > 20) begin raiserror(N'Tên đăng nhập không tồn tại', 16,1) rollback tran end </pre>	
<pre> else if (LEN(@MK) > 20) begin raiserror(N'Sai mật khẩu', 16,1) rollback tran end </pre>	
<pre> else if not exists(select * from TaiKhoan where TaiKhoan.TenTaiKhoan = @TK and TaiKhoan.MatKhau = @MK) begin raiserror (N'Sai tên đăng nhập hoặc mật khẩu', 16,1) rollback tran end </pre>	
<pre> else print (N'Đăng nhập thành công') </pre>	
	<pre> declare @TK varchar(20), @MK varchar(20), @MKMoi varchar(20), @MKMoiLan2 varchar(20) select @TK = '_hopeful', @MK = 'tOxd4cLQkFyI', @MKMoi = '123456', @MKMoiLan2 = '123456' if not exists(select * from TaiKhoan where taikhoan.TenTaiKhoan = @TK and TaiKhoan.MatKhau = @MK) </pre>



	<pre>begin raiserror(N'Tài khoản hoặc mật khẩu không chính xác',16,1) rollback end</pre>
	<pre>else if (@MKMoi = null or len(@MKMoi) > 20) begin raiserror (N'Mật khẩu mới không hợp lệ',16,1) rollback end</pre>
	<pre>else if (@MKMoi != @MKMoiLan2) begin raiserror (N'Nhập sai mật khẩu mới',16,1) rollback end</pre>
	<pre>else begin update TaiKhoan set MatKhau = @MKMoi where TaiKhoan.TenTaiKhoan = @TK print N'Đổi mật khẩu thành công' end</pre>
<pre>if not exists (select * from TaiKhoan where TaiKhoan.TenTaiKhoan = @TK and TaiKhoan.MatKhau = @MK) begin raiserror(N'Không tìm thấy thông tin tài khoản này',16,1) rollback tran end</pre>	

Giải thích:

T1: Nhập đúng tài khoản và mật khẩu, hệ thống báo đăng nhập thành công nhưng T2 đã đổi mật khẩu trước khi T1 truy vấn được thông tin tài khoản '_hopeful'



Messages

Đăng nhập thành công

Msg 50000, Level 16, State 1, Procedure DangNhap, Line 37 [Batch Start Line 2]

Không tìm thấy thông tin tài khoản này

T2: Đổi mật khẩu thành công của tài khoản '_hopeful'

Messages

(1 row affected)

Đổi mật khẩu thành công

Completion time: 2021-11-22T18:22:44.6459429+07:00

Demo trên giao diện:

Trường hợp nhập sai mật khẩu: server báo tài khoản hoặc mật khẩu không đúng.

TuDuTu

localhost:8080 says
Tài khoản hoặc mật khẩu không đúng

OK

Đăng nhập

Tên tài khoản

Mật khẩu

[Change password](#)

LOGIN

Transaction 1 đăng nhập vào tài khoản thành công nhưng do mật khẩu đã bị thay đổi trong quá trình đăng nhập nên server không thể tìm thấy thông tin tài khoản của tài khoản này.

The screenshot shows a login window titled "Đăng nhập". It has two input fields: "Tên tài khoản" (Username) with the value "tuan" and "Mật khẩu" (Password) with masked characters "....". Below the password field is a link "Change password". A green "LOGIN" button is at the bottom right. An error message box is displayed on top, stating "localhost:8080 says" and "Không tìm thấy thông tin tài khoản này" (No account information found). An "OK" button is on the error box.

Transaction 2 thay đổi mật khẩu trong khi tài khoản đang được đăng nhập ở một tab khác.

The screenshot shows a password change window titled "Đổi mật khẩu". It has four input fields: "Tên tài khoản" (Username) with "tuan", "Mật khẩu cũ" (Old password) with "....", "Mật khẩu mới" (New password) with ".....", and "Nhập lại mật khẩu mới" (Repeat new password) with ".....". A green button labeled "Đổi mật khẩu" is at the bottom right. An error message box is displayed on top, stating "localhost:8081 says" and "Change password successfully!". An "OK" button is on the error box. In the top left corner, the text "TuDuTu" is visible.

Giải quyết tranh chấp:

Thiết lập mức cô lập cho transaction 1 là REPEATABLE READ:

Transaction 1	Transaction 2	Giải thích
<pre>begin tran if exists(select * from TaiKhoan where TaiKhoan.TenTaiKhoan = @TK and TaiKhoan.MatKhai = @MK) begin print (N'Đăng nhập thành công')</pre>		<p>T1 xin cấp khóa S-lock trên bảng TaiKhoan SQL: cấp khóa S-lock trên bảng TaiKhoan cho T1 và T1 giữ S-lock tới hết giao tác</p>
	<pre>if not exists(select * from TaiKhoan where taikhoan.TenTaiKhoan = @TK and TaiKhoan.MatKhai = @MK) begin raiserror(N'Tài khoản hoặc mật khẩu không chính xác',16,1) rollback end</pre>	
	<pre>else begin update TaiKhoan set MatKhai = @MKMoi where TaiKhoan.TenTaiKhoan = @TK print N'Đổi mật khẩu thành công' commit tran end (chờ)</pre>	<p>T2 xin cấp khóa X-lock trên bảng TaiKhoan SQL: không cấp khóa X-lock cho T2 vì T1 đang giữ S-lock.</p>
<pre>if not exists (select * from TaiKhoan where TaiKhoan.TenTaiKhoan = @TK and TaiKhoan.MatKhai = @MK)</pre>		



<pre>begin raiserror(N'Không tìm thấy thông tin tài khoản này',16,1) rollback Tran end</pre>		
<pre>Truy xuất thông tin người đăng nhập commit tran</pre>		T1 trả khóa S-lock trên bảng TaiKhoan.
	<pre>else begin update TaiKhoan set MatKau = @MKMoi where TaiKhoan.TenTaiKhoan = @TK print N'Đổi mật khẩu thành công' commit tran end</pre>	SQL: cấp khóa X-lock trên bảng TaiKhoan cho T2.

Tình huống 2

Transaction 1: Tìm kiếm sản phẩm có từ khóa “hữu cơ”.

Transaction 2: Thay đổi tên sản phẩm 1 từ “Hạt hạnh nhân hữu cơ” thành “Hạt hạnh nhân”.

Transaction 1	Transaction 2
<pre> if exists (Select * From SANPHAM SP Where SP.TenSP LIKE ('%' + @TenSP + '%')) begin Print (N'Tồn tại sản phẩm') </pre>	
	<pre> if not exists (select* from SANPHAM SP where SP.MaSP = @MaSP) begin raiserror('Không tìm thấy sản phẩm.', 16, 1) rollback tran end </pre>
	<pre> else begin update SANPHAM set [TenSP] = @TenMoi where [MaSP] = @MaSP commit tran end </pre>
<pre> Select * From SANPHAM SP Where SP.TenSP LIKE ('%' + @TenSP + '%') end </pre>	
<pre> commit tran </pre>	

Giải thích:

Dữ liệu ban đầu:

	MaSP	TenSP	Gia	SoLuongTon	MaChiNhanh
1	1	Hạt hạnh nhân hữu cơ	450000	110	1

Thực thi transaction 1 để tìm kiếm sản phẩm có chứa từ khóa “hữu cơ”. Trong lúc transaction 1 đang thực thi, transaction 2 thay đổi tên của sản phẩm 1 từ “Hạt hạnh nhân hữu cơ” thành “Hạt hạnh nhân”.

Transaction 1 bị “unrepeatable read” khi tìm thấy sản phẩm có chứa từ khóa “hữu cơ” ở lần đọc đầu tiên (bên trong điều kiện của câu lệnh if):

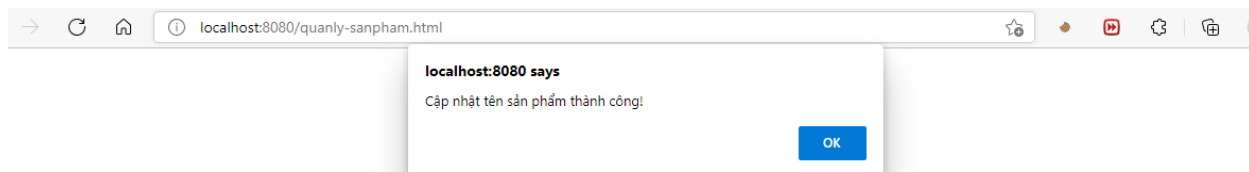
Results	Messages
Tồn tại kết quả	
(0 row(s) affected)	

Nhưng khi đọc lần 2 thì không xuất ra được thông tin sản phẩm nào:

Results	Messages

Demo trên giao diện

Thực hiện tìm kiếm sản phẩm có từ khóa “hữu cơ”. Trong lúc transaction 1 tìm kiếm thì thực hiện đổi tên sản phẩm ở transaction 2:

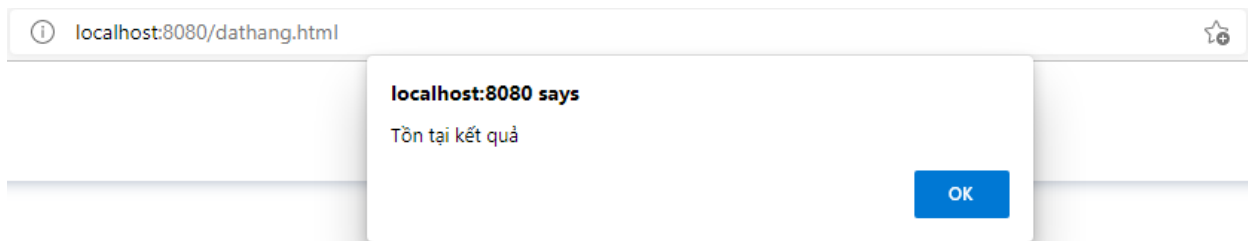


Cập nhật sản phẩm

Mã SP: Tên sản phẩm:

[Cập nhật tên sản phẩm](#)

Khi transaction 1 tìm kiếm kết thúc thì có thông báo tồn tại kết quả:



Nhưng danh sách kết quả thì không hiện thông tin sản phẩm nào:

Danh sách sản phẩm

hữu cơ

Q

Lọc theo

Mã SP Tên SP Giá SL Tồn Mã CN

Đối tác

Chi nhánh

Giá

Lọc

Giải quyết tranh chấp

Thiết lập mức cô lập cho transaction 1 là REPEATABLE READ.

Transaction 1	Transaction 2	Giải thích
<pre>if exists (Select * From SANPHAM SP Where SP.TenSP LIKE ('%' + @TenSP + '%')) begin Print (N'Tồn tại sản phẩm')</pre>		T1 xin cấp khóa S-lock trên bảng SANPHAM SQL: cấp khóa S-lock trên bảng SANPHAM cho T1 và T1 giữ S-lock tới hết giao tác
	<pre>if not exists (select* from SANPHAM SP where SP.MaSP = @MaSP)</pre>	



	<pre>begin raiserror('Không tìm thấy sản phẩm.', 16, 1) rollback tran end</pre>	
	<pre>else begin update SANPHAM set [TenSP] = @TenMoi where [MaSP] = @MaSP commit tran end (CHỜ)</pre>	T2 xin cấp khóa X-lock trên bảng SANPHAM SQL: không cấp khóa X-lock cho T2 vì T1 đang giữ S-lock.
<pre>Select * From SANPHAM SP Where SP.TenSP LIKE ('%' + @TenSP + '%') end</pre>		
<pre>commit tran</pre>		T1 trả khóa S-lock trên bảng SANPHAM.
	<pre>begin update SANPHAM set [TenSP] = @TenMoi where [MaSP] = @MaSP commit tran end</pre>	SQL: cấp khóa X-lock trên bảng SANPHAM cho T2.

4. Phantom Read

Tình huống 1

Transaction 1: Khách hàng xem các mặt hàng của chi nhánh 1

Transaction 2: Đối tác thêm vào chi nhánh 1 một mặt hàng

Transaction 1	Transaction 2
<pre> declare @MaChiNhanh int select @MaChiNhanh = 1 if not exists (select * from CHINHANH where CHINHANH.MaChiNhanh = 1) begin raiserror(N'Không có chi nhánh này', 16,1) rollback end </pre>	
<pre> declare @TongSoSP int set @TongSoSP = (select count (sp.MaSP) from SANPHAM sp where sp.MaChiNhanh = @MaChiNhanh) print N'Có ' + cast (@TongSoSP as nvarchar(10))+N' sản phẩm' </pre>	
	<pre> Declare @MaSP int, @TenSP nvarchar(100), @Gia bigint, @SoLuongTon int = NULL, @MaChiNhanh int = NULL select @MaSP = 5, @TenSP = N'Bánh oreo ngon giòn 150g', @Gia = 15000, @SoLuongTon = 100, @MaChiNhanh = 1 if @Gia < 0 begin raiserror('Giá sản phẩm không hợp lệ.', 16, 1) rollback tran end </pre>
	<pre> else begin </pre>

	<code>insert into SANPHAM (TenSP, Gia, SoLuongTon, MaChiNhanh) values (@TenSP, @Gia, @SoLuongTon, @MaChiNhanh)</code>
<code>select * from SANPHAM sp where sp.MaChiNhanh = @MaChiNhanh</code>	

Kết quả:

Transaction 1: Đã được có tất cả 4 sản phẩm thuộc chi nhánh 1 nhưng xuất ra thông tin 5 sản phẩm do trước khi truy vấn thông tin của các sản phẩm thuộc chi nhánh 1 thì T2 đã thêm vào một sản phẩm

<code>Có 4 sản phẩm</code>
<code>(5 rows affected)</code>
<code>Completion time: 2021-11-22T18:17:43.3922993+07:00</code>

	MaSP	TenSP	Gia	SoLuongTon	MaChiNhanh
1	1	Hạt hạnh nhân hữu cơ 1kg	450000	100	1
2	2	Khô Héo Cháy Tỏi DTFood Đặc Biệt Thơm Ngon	85000	100	1
3	3	Thùng 20 gói Mì Rong Biển Ottogi 120gx20	250000	200	1
4	4	Mì Trộn Xốt Tương Đen Hàn Quốc Ottogi 135Gr	35000	50	1
5	5	Bánh oreo ngon giòn 150g	15000	100	1

Transaction 2: Thêm vào một sản phẩm thuộc chi nhánh 1 làm cho kết quả đọc của T1 trước và sau không cùng số bộ dữ liệu.

Demo lỗi trên giao diện:

Transaction 1 tìm các sản phẩm thuộc chi nhánh 1 : Database trả về tổng số sản phẩm đọc được của chi nhánh 1 là 9, nhưng thực tế lại trả ra thông tin của 10 sản phẩm. Do trước khi truy xuất thông tin các sản phẩm thì một sản phẩm mới đã được thêm vào.

Danh sách sản phẩm

localhost:8080 says
Tổng số sản phẩm là: 9

OK

Lọc theo
Đối tác
Chi nhánh
1
Giá
Lọc

Mã SP	Tên SP	Giá	SL Tồn	Mã CN
1	Hạt hạnh nhân hữu cơ 1kg	450000	100	1
2	Khô Héo Chấy Tỏi DTFood Đặc Biệt Thơm Ngon	85000	100	1
3	Thùng 20 gói Mì Rong Biển Ottogi 120gx20	250000	200	1
4	Mì Trộn Xốt Tương Đen Hàn Quốc Ottogi 135Gr	35000	50	1
5	Bánh oreo ngon giòn 150g	15000	100	1
100101	redbull 330ml	15000	1000	1
100102	snack BBQ 25g	12000	1000	1
100103	Kẹo chuppachip	2000	1000	1
100104	Nước ép Vườn nhà 500ml	10000	1000	1
100105	Nước ép cam nguyên chất 500ml	10000	1000	1

Transaction 2 thêm vào thành công một sản phẩm thuộc chi nhánh 1 sau khi Transaction 1 đếm số sản phẩm của chi nhánh 1, nhưng trước khi transaction 1 đọc thông tin của các sản phẩm:

TuDuTu

localhost:8081/quanly-sanpham.html

Thêm sản phẩm

localhost:8081 says
Thêm sản phẩm thành công!

OK

Tên sản phẩm
Giá
SL tồn
Mã chi nhánh
1

Xác nhận



Giải quyết tranh chấp:

Thực hiện cài đặt mức cô lập SERIALIZABLE trên transaction 1:

Transaction 1	Transaction 2	Giải thích
<pre>if @MaChiNhanh != NULL and not exists (select * from CHINHANH CN where CN.MaChiNhanh = @MaChiNhanh) begin raiserror('Không tìm thấy chi nhánh.', 16, 1); rollback tran end</pre>		
<pre>declare @TongSoSP int set @TongSoSP = (select count (sp.MaSP) from SANPHAM sp where sp.MaChiNhanh = @MaChiNhanh)</pre>		T1 xin S-lock trên bảng SanPham SQL: Cấp S-lock trên bảng SanPham cho T1 và ngăn không cho thêm dòng dữ liệu
<pre>print N'Có ' + cast (@TongSoSP as nvarchar(10))+N' sản phẩm'</pre>		
	<pre>insert into SANPHAM (MaSP, TenSP, Gia, SoLuongTon, MaChiNhanh) values (@MaSP, @TenSP, @Gia, @SoLuongTon, @MaChiNhanh) (chờ)</pre>	T2 xin X-lock trên bảng SanPham SQL: không cấp X- lock cho T2 vì T1 ngăn không cho thêm dữ liệu trên bảng SanPham
<pre>select * from SANPHAM sp where sp.MaChiNhanh = @MaChiNhanh select @TongSoSP as Tongso</pre>		
<pre>commit tran</pre>		T1 trả S-lock trên bảng SanPham



	<pre>insert into SANPHAM (MaSP, TenSP, Gia, SoLuongTon, MaChiNhanh) values (@MaSP, @TenSP, @Gia, @SoLuongTon, @MaChiNhanh)</pre>	SQL cấp X-lock cho T2.
--	--	------------------------

Tình huống 2

Transaction 1: Xem tất cả đơn hàng thuộc chi nhánh 1 trong tình trạng “Chờ xác nhận”.

Transaction 2: Thêm đơn hàng mới thuộc chi nhánh 1 cũng trong tình trạng “Chờ xác nhận”.

Transaction 1	Transaction 2
<pre>if @MaChiNhanh != NULL and not exists (select * from CHINHANH CN where CN.MaChiNhanh = @MaChiNhanh) begin raiserror('Không tìm thấy chi nhánh.', 16, 1); rollback tran end</pre>	
<pre>Set @SoDonHang = (Select count(*) From DONHANG Where MaChiNhanh = @MaChiNhanh and TinhTrangVanChuyen = @TinhTrang)</pre>	
<pre>Print N'Tổng số đơn hàng trong tình trạng "Chờ xác nhận": ' + CAST(@SoDonHang AS VARCHAR(10))</pre>	
	<pre>insert into DONHANG (HinhThucThanhToan, DiaChiGiaoHang, PhiVC, MaKH, MaChiNhanh, TinhTrangVanChuyen) values (@HinhThucThanhToan, @DiaChiGiaoHang, @PhiVC, @MaKH, @MaChiNhanh, N'Chờ xác nhận')</pre>
	<pre>commit tran</pre>
<pre>Select * From DONHANG Where MaChiNhanh = @MaChiNhanh and TinhTrangVanChuyen = @TinhTrang</pre>	



commit tran

Giải thích:

Dữ liệu ban đầu:

	MaDonHang	NgayLap	HinhThucThanhToan	DiaChiGiaoHang	PhiSP	PhiVC	MaKH	MaChiNhanh	MaTaiXe	TinhTrangVanChuyen
1	7	2021-10-21 00:00:00.000	Tiền mặt	182 Nguyễn Huệ, Q.1	NULL	20000	1	1	NULL	Chờ xác nhận
2	8	2021-10-03 00:00:00.000	MoMo	64 Hàn Hải Nguyên, Q.11	NULL	20000	2	1	NULL	Chờ xác nhận

Thực thi transaction 1 để thống kê số lượng và xem thông tin các đơn hàng đang trong tình trạng “Chờ xác nhận”. Trong lúc transaction 1 đang thực thi, thực thi transaction 2 để thêm 1 đơn hàng mới cũng với tình trạng “Chờ xác nhận”.

Kết quả: Transaction 1 bị “phantom read” khi thống kê tổng số đơn hàng đang trong tình trạng “Chờ xác nhận” là 2, nhưng lại xuất ra 3 dòng kết quả:

Tổng số đơn hàng trong tình trạng “Chờ xác nhận”: 2

	MaDonHang	NgayLap	HinhThucThanhToan	DiaChiGiaoHang	PhiSP	PhiVC	MaKH	MaChiNhanh	MaTaiXe	TinhTrangVanChuyen
1	7	2021-10-21 00:00:00.000	Tiền mặt	182 Nguyễn Huệ, Q.1	NULL	20000	1	1	NULL	Chờ xác nhận
2	8	2021-10-03 00:00:00.000	MoMo	64 Hàn Hải Nguyên, Q.11	NULL	20000	2	1	NULL	Chờ xác nhận
3	9	2021-11-19 11:05:47.260	Tiền mặt	402 Nguyễn Thị Minh Khai, Q.1	NULL	30000	1	1	NULL	Chờ xác nhận

Demo trên giao diện:

Thực thi transaction 1 để lọc ra những đơn hàng thuộc chi nhánh 1 và có tình trạng là “Chờ xác nhận”.

Trong lúc đó, thực hiện đặt hàng các sản phẩm thuộc chi nhánh 1 ở transaction 2:



Giỏ hàng

localhost:8080 says
Đặt hàng thành công!

OK

Mã SP			Mã CN	Số lượng
2	Khô Heo Chấy Tỏi DTFood Đặc Biệt Thơm Ngon	85000	1	1
4	Mì Trộn Xốt Tương Đen Hàn Quốc Ottogi 135Gr	35000	1	2

Tổng tiền: 155000

Địa chỉ

843 Đường 3/2, Quận 5, TP.HCM

Hình thức thanh toán

Tiền mặt
Momo
Visa

Activate Windows
Go to Settings to activate Windows.

Đặt hàng

Khi transaction 1 kết thúc việc lọc đơn hàng, hệ thống thông báo tổng số đơn hàng thỏa mãn điều kiện là 3:

localhost:8081/xem-donhang.html

localhost:8081 says
Tổng số đơn hàng: 3

OK

Nhưng danh sách đơn hàng lại hiển thị thông tin của 4 đơn hàng:

Danh sách đơn hàng

Lọc theo

Chi nhánh

1

Tình trạng

Chờ xác nhận

Lọc

Mã đơn	Mã khách hàng	Tổng tiền	Tình trạng	Mã chi nhánh
1	1	340000	Chờ xác nhận	1
2	2	450000	Chờ xác nhận	1
3	1	100000	Chờ xác nhận	1
100099	100099	155000	Chờ xác nhận	1



Giải quyết tranh chấp:

Thiết lập mức cô lập SERIALIZABLE trên transaction 1.

Transaction 1	Transaction 2	Giải thích
<pre>if @MaChiNhanh != NULL and not exists (select * from CHINHANH CN where CN.MaChiNhanh = @MaChiNhanh) begin raiserror('Không tìm thấy chi nhánh.', 16, 1); rollback tran end</pre>		
<pre>Set @SoDonHang = (Select count(*) From DONHANG Where MaChiNhanh = @MaChiNhanh and TinhTrangVanChuyen = @TinhTrang)</pre>		T1 xin S-lock trên bảng DONHANG SQL: Cấp S-lock trên bảng DONHANG cho T1 và ngăn không cho thêm dòng dữ liệu
<pre>Print N'Tổng số đơn hàng trong tình trạng "Chờ xác nhận": ' + CAST(@SoDonHang AS VARCHAR(10))</pre>		
	<pre>insert into DONHANG (HinhThucThanhToan, DiaChiGiaoHang, PhiVC, MaKH, MaChiNhanh, TinhTrangVanChuyen) values (@HinhThucThanhToan, @DiaChiGiaoHang, @PhiVC, @MaKH, @MaChiNhanh, N'Chờ xác nhận') (CHỜ)</pre>	T2 xin X-lock trên bảng DONHANG SQL: không cấp X-lock cho T2 vì T1 ngăn không cho thêm dữ liệu trên bảng DONHANG
<pre>Select * From DONHANG Where MaChiNhanh = @MaChiNhanh and TinhTrangVanChuyen = @TinhTrang</pre>		



<code>commit tran</code>		T1 trả S-lock trên bảng đơn hàng
	<pre>insert into DONHANG (HinhThucThanhToan, DiaChiGiaoHang, PhiVC, MaKH, MaChiNhanh, TinhTrangVanChuyen) values (@HinhThucThanhToan, @DiaChiGiaoHang, @PhiVC, @MaKH, @MaChiNhanh, N'Chờ xác nhận')</pre>	SQL cấp X-lock cho T2.