**BÁO CÁO ĐỒ ÁN THỰC HÀNH**

MÔN **HỆ QUẢN TRỊ CƠ SỞ DỮ LIỆU**

GVHD: Cô **Tiết Gia Hồng**

**Nhóm 7 - Lớp: 19HTTT2**

**25/12/2021**

Mục lục

[PHÂN CÔNG 4](#_Toc91153933)

[ **Yêu cầu đồ án** 4](#_Toc91153934)

[ **Phần trăm tham gia** 5](#_Toc91153935)

[ **Phần trăm hoàn thành công việc** 7](#_Toc91153936)

[KẾT QUẢ ĐỒ ÁN 7](#_Toc91153937)

[**I.** **Lược đồ thực thể kết hợp** 7](#_Toc91153938)

[**II.** **Ràng buộc dữ liệu bổ sung** 11](#_Toc91153939)

[**III.** **Mô hình cơ sở dữ liệu quan hệ** 12](#_Toc91153940)

[**IV.** **Các chức năng của hệ thống** 14](#_Toc91153941)

[**1.** **Chức năng cho DÙNG CHUNG** 14](#_Toc91153942)

[**2.** **Phân hệ đối tác** 14](#_Toc91153943)

[**3.** **Phân hệ khách hàng** 15](#_Toc91153944)

[**4.** **Phân hệ tài xế** 16](#_Toc91153945)

[**5.** **Phân hệ nhân viên** 17](#_Toc91153946)

[**6.** **Phân hệ quản trị** 18](#_Toc91153947)

[**V.** **Xác định tình huống tranh chấp** 18](#_Toc91153948)

[VI. Thiết kế giao diện chức năng 20](#_Toc91153949)

[**1.** **Homepage** 20](#_Toc91153950)

[**2.** **Chức năng: Đăng nhập** 22](#_Toc91153951)

[**3.** **Chức năng: Đăng ký (Đối tác)** 23](#_Toc91153952)

[**4.** **Chức năng: Lập hợp đồng (Đối tác)** 24](#_Toc91153953)

[**5.** **Chức năng: Quản lý sản phẩm và chi nhánh (Đối tác)** 25](#_Toc91153954)

[**6.** **Chức năng: Theo dõi đơn hàng (Đối tác)** 26](#_Toc91153955)

[**7.** **Chức năng: Đăng ký (Khách hàng)** 27](#_Toc91153956)

[**8.** **Chức năng: Đặt hàng (Khách hàng)** 28](#_Toc91153957)

[**9.** **Chức năng: Theo dõi đơn hàng (Khách hàng)** 29](#_Toc91153958)

[**10.** **Chức năng: Đăng ký làm tài xế** 30](#_Toc91153959)

[**11.** **Chức năng: Nhận đơn hàng (Tài xế)** 32](#_Toc91153960)

[**12.** **Chức năng: Chỉnh sửa tình trang đơn hàng (Tài xế)** 33](#_Toc91153961)

[**13.** **Chức năng: Theo dõi thu nhập (Tài xế)** 34](#_Toc91153962)

[**14.** **Chức năng: Xét duyệt hợp đồng (Nhân viên)** 35](#_Toc91153963)

[**15.** **Chức năng: Gia hạn hợp đồng (Nhân viên)** 37](#_Toc91153964)

[**16.** **Chức năng: Cập nhật thông tin các tài khoản và Thêm xóa sửa tài khoản nhân viên và admin (Quản trị)** 38](#_Toc91153965)

[**17.** **Chức năng: Khóa và kích hoạt tài khoản (Quản trị)** 39](#_Toc91153966)

[**18.** **Chức năng: Cấp quyền thao tác trên dữ liệu & giao diện cho người dùng (Quản trị)** 40](#_Toc91153967)

[VII. Cài đặt tình huống tranh chấp 40](#_Toc91153968)

[**1.** **Tình huống 1: Kiểm tra hơp đồng** 41](#_Toc91153969)

[**2.** **Tình huống 2: Xem và Nhận đơn hàng** 42](#_Toc91153970)

[**3.** **Tình huống 3: Hai tài xế cùng nhận chung một đơn hàng** 44](#_Toc91153971)

[**4.** **Tình huống 4: Khách hàng đặt hàng thất bại nhưng đối tác vẫn thấy đơn hàng** 46](#_Toc91153972)

[**5.** **Tình huống 5: Khách hàng đang xem sản phẩm thì thêm 1 sản phẩm khác vào** 49](#_Toc91153973)

[**6.** **Tình huống 6: Đối tác lập hợp đồng xong đọc thông báo, Nhân viên gửi thông báo tới đối tác xong xem hợp đồng** 50](#_Toc91153974)

[VIII. Xử lý tình huống tranh chấp 51](#_Toc91153975)

[**1.** **Tình huống 1: Kiểm tra hơp đồng** 51](#_Toc91153976)

[**2.** **Tình huống 2: Nhận và nhận đơn hàng** 53](#_Toc91153977)

[**3.** **Tình huống 3: Hai tài xế cùng nhận chung một đơn hàng** 56](#_Toc91153978)

[**4.** **Tình huống 4: Khách hàng đặt hàng thất bại nhưng đối tác vẫn thấy đơn hàng** 59](#_Toc91153979)

[**5.** **Tình huống 5: Khách hàng đang xem sản phẩm thì thêm 1 sản phẩm khác vào** 62](#_Toc91153980)

[**6.** **Tình huống 6: Đối tác lập hợp đồng xong đọc thông báo, Nhân viên gửi thông báo tới đối tác xong xem hợp đồng** 63](#_Toc91153981)

[IX. THAM KHẢO 64](#_Toc91153982)

# **PHÂN CÔNG**

* **Yêu cầu đồ án**

|  |  |  |
| --- | --- | --- |
| **STT** | **Công việc** | **%** |
| 1 | Phân tích nghiệp vụ và thiết kế cơ sở dữ liệu phù hợp | **100** |
| 2 | Xác định và xây dựng các chức năng phù hợp Web | **100** |
| 3 | Xác định vấn đề tranh chấp dữ liệu liệu trong hệ thống và hướng giải quyết | **100** |
| 4 | Phân quyền người dùng trong hệ thống | **100** |
| 5 | Report | **100** |
| **Total** | | **100** |

* **Phần trăm tham gia**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **MSSV** | **Họ tên** | **Phụ trách** | **%** |
| 1 | 19127097 | Nguyễn Ngọc Phương Anh | Phân quyền người dùng trong hệ thống  Xác định và xây dựng các chức năng phù hợp Web  Xác định vấn đề tranh chấp dữ liệu liệu trong hệ thống và hướng giải quyết | **33.5** |
| 2 | 19127449 | Phùng Anh Khoa | Phân tích nghiệp vụ và thiết kế cơ sở dữ liệu phù hợp  Xác định và xây dựng các chức năng phù hợp Web  Xác định vấn đề tranh chấp dữ liệu liệu trong hệ thống và hướng giải quyết | **33.5** |
| 3 | 19127095 | Ngô Huy Anh | Xác định và xây dựng các chức năng phù hợp Web  Xác định vấn đề tranh chấp dữ liệu liệu trong hệ thống và hướng giải quyết  Report | **33** |
| **Total** | | | | **100** |

* **Phần trăm hoàn thành công việc**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **MSSV** | **Họ tên** | **Công việc** | **%** |
| 1 | 19127097 | Nguyễn Ngọc Phương Anh | Xây dựng các chức năng phù hợp cho Web  Report | **100** |
| 2 | 19127449 | Phùng Anh Khoa | Xây dựng các chức năng phù hợp cho Web  Report | **100** |
| 3 | 19127095 | Ngô Huy Anh | Xây dựng các chức năng phù hợp cho Web  Report | **100** |

# **KẾT QUẢ ĐỒ ÁN**

## **Lược đồ thực thể kết hợp**

**Mô hình ER**

Diagram, schematic

Description automatically generated

Link: [drawio](https://drive.google.com/file/d/1_XrvMzxg7E2i1r17TEqbDfGWjahNi1wr/view?usp=sharing)

**Mô tả bảng – thuộc tính**

Bao gồm 12 bảng:

* Đối tác (**Mã số thuế**, Tên đối tác, Người đại diện, Thành phố, Quận, Số chi nhánh, Số lượng đơn hàng mỗi ngày, Loại hàng vận chuyển, Địa chỉ kinh doanh, SĐT, Email)
* Mỗi đối tác có một mã số thuế để phân biệt với các đối tác khác
* Khai báo số lượng đơn hàng mỗi ngày để công ty biết được giới hạn vận chuyển của đối tác
* Khai báo số chi nhánh để tránh trường hợp đối tác thêm quá số lượng chi nhánh
* Hợp đồng (**Mã hợp đồng**, Mã số thuế, Số chi nhánh đăng ký, Phần trăm hoa hồng, Thời gian hiệu lực, Tình trạng)
* Mỗi hợp đồng có một mã hợp đồng để phân biệt với các hợp đồng khác
* Mỗi hợp đồng có một mã số thuế để biết được hợp đồng này của đối tác nào
* Mỗi đối tác chỉ có một hợp đồng duy nhất
* Sau khi đăng ký xong sẽ lấy 1 triệu tiền phí
* Mỗi tháng cần đóng phí hoa hồng: Số tiền kiếm được hàng tháng \* phần trăm hoa hồng
* Đơn hàng (**Mã đơn hàng**, Tên đơn hàng, Ngày đặt, Tình trạng, Khu vực, Phí vận chuyển, Phí sản phẩm, Hình thức thanh toán, Địa chỉ giao hàng, Mã khách hàng, Mã số thuế, Mã tài xế)
* Mỗi đơn hàng cần mã đơn hàng để phân biệt với các đơn hàng khác
* Cần ngày đặt để biết được số đơn hàng mỗi ngày của đối tác, tránh tình trạng đối tả vận chuyển nhiều hơn mức có thể
* Cần Tình trạng để khách hàng biết được đơn hàng đang giao, đã giao, hay chưa giao. Đối tác và tài xế có thể chỉnh giá trị này
* Cần khu vực để tài xế chọn hàng trong khu vực hoạt động của tài xế
* Cần phí vận chuyển để khách hàng biết được đơn giá đơn hàng phải trả
* Cần phí vận chuyển để khách hàng biết phải trả và để tài xế biết được số tiền nhận được
* Cần mã khách hàng để biết đơn hàng của khách hàng nào
* Cần mã số thuế để biết đơn hàng của đối tác nào để lấy giá sản phẩm cung cấp từ đối tác đó
* Cần mã tài xế để biết tài xế nào nhận đơn hàng này và để tài xế có thể chỉnh tình trạng của đơn
* Khách hàng (**Mã khách hàng**, Họ tên, SĐT, Địa chỉ, Email)
* Cần mã khách hàng để phân biệt với các khách hàng khác
* Tài xế (**Mã tài xế**, Họ tên, CMND, SĐT, Biển số xe, Địa chỉ, Khu vực hoạt động, Email, STK, Ngân hàng)
* Cần mã tài xế để phân biệt với các tài xế khác
* Cần khu vực hoạt động để nhận đơn hàng nằm trong khu vực có thể nhận được
* Chi nhánh (**Mã chi nhánh**, Mã số thuế, Tên chi nhánh, Địa chỉ)
* Cần mã chi nhánh để phân biệt cách chi nhánh với nhau
* Cần mã số thuế để biết chi nhánh thuộc về đối tác nào
* Thông báo (**Mã thông báo**, Nội dung, Mã số thuế)
* Cần mã thông báo để phân biệt các thông báo với nhau
* Cần mã số thuế để biết được thông báo này gửi tới đối tác nào
* Nhân viên (**Mã nhân viên**, Họ tên, SĐT, Địa chỉ, Email)
* Cần mã nhân viên để phân biệt các nhân viên với nhau
* Admin (**Mã Admin**, Họ tên, SĐT, Địa chỉ, Email)
* Cần mã Admin để phân biệt các Admin với nhau
* Sản phẩm (**Mã sản phẩm**, Mã chi nhánh, Số lượng tồn, Giá cả, TenSP)
* Cần mã sản phẩm để phân biệt các sản phẩm với nhau
* Cần mã chi nhánh để biết được chi nhánh nào cung cấp sản phẩm này
* DHSP (**Mã sản phẩm**, **Mã đơn hàng**, Số lượng, Giá)
* Cần mã sản phẩm, đơn hàng để phân biệt các Đơn hàng sản phẩm
* Cần số lượng, giá để tính được phí sản phẩm phải trả ở bảng đơn hàng
* Tài khoản (**Mã người dùng**, Tên người dùng, Mật khẩu, Loại người dùng, Tình trạng)
* Cần mã người dùng để phân biệt người dùng
* Cần loại người dùng để biết người dùng này là gì (Admin, khách hàng, nhân viên, đối tác, tài xế) giúp việc truy vấn thuận tiện hơn
* Cần tình trạng để biết tài khoản này có được sử dụng không

## **Ràng buộc dữ liệu bổ sung**

* **Bảng Đơn hàng (DonHang):**
* Đọc từ bảng Đối tác: Mỗi đối tác có số lượng đơn giao nhất định cho mỗi ngày, nếu đã đủ thì không giao thêm đơn nào nữa.
* Cột TinhTrang: chỉ nhận giá trị ‘Đang giao’, ‘Đã giao’ và ‘Chờ’
* Phí sản phẩm = tổng của các đơn giá ở bảng DHSP
* **Bảng Hợp đồng (HopDong):**
* Cột TinhTrang: chỉ nhận giá trị ‘Chờ duyệt’ ,‘Đã duyệt’ và huỷ
* Phần trăm hoa hồng >= 0 và <=100
* **Bảng Chi nhánh (ChiNhanh):**
* Đọc từ bảng Đối tác: Thêm, chỉnh chi nhánh. Mỗi đối tác có ghi số lượng chi nhánh khi đăng ký.
* **Bảng Đơn hàng sản phẩm (DHSP):**
* Đọc số sản phẩm từ bảng Sản phẩm: Mỗi sản phẩm có 1 số lượng nhất định, nếu thêm sửa số lượng mà vượt quá số hiện tại thì huỷ.
* Đơn giá = Giá sản phẩm \* số lượng
* **Bảng Tài xế (TaiXe):**
* Cột CMND: là duy nhất
* **Bảng Tài khoản (TaiKhoan):**
* Cột TinhTrang: Chỉ nhận giá trị Khoá hoặc mở
* Cột LoaiNguoiDung: nhận giá trị 1,2,3,4,5 tượng trưng cho 5 loại người dùng (Quản trị, Đối tác, Khách Hàng, Tài xế, Nhân viên)
* **Bảng Chi tiết hợp đồng (HopDong):**
* Chi nhánh phải thuộc về đối tác mà đăng ký hợp đồng này
* Số chi nhánh đăng ký cho hợp đồng không lớn hơn số chi nhánh đăng ký đã ghi khi đăng ký hợp đồng

## **Mô hình cơ sở dữ liệu quan hệ**

Chart

Description automatically generatedTable

Description automatically generatedTable

Description automatically generated

Link: [Google Sheets](https://docs.google.com/spreadsheets/d/1dyUnkMach_zofEf7NJmoPfrPLfDUmjCa/edit#gid=469722269)

## **Các chức năng của hệ thống**

### **Chức năng cho DÙNG CHUNG**

|  |  |  |
| --- | --- | --- |
| **STT** | **Chức năng** | **Mô tả hoạt động** |
| DC1 | Đăng nhập | Kết nối tới server bằng tài khoản và mật khẩu đã tạo. |

### **Phân hệ đối tác**

|  |  |  |
| --- | --- | --- |
| **STT** | **Chức năng** | **Mô tả hoạt động** |
| DT1 | Đăng ký làm đối tác | Đối tác nhập: Tên đối tác, người đại diện, thành phố, quận, số chi nhánh, số lượng đơn hàng mỗi ngày, loại hàng vận chuyển, địa chỉ kinh doanh, số điện thoại, email. Hệ thống tiếp nhận vào tạo tài khoản nếu hợp lệ. |
| DT2 | Lập hợp đồng | Đối tác nhập: Mã số thuế của đối tác, người đại diện, số chi nhánh đăng ký, địa chỉ các chi nhánh để đăng ký hợp đồng. Hệ thống áp dụng mức phí hoa hồng hiện tại. |
| DT3 | Tái kích hoạt hợp đồng | Đối tác đồng ý tái kích hoạt hợp đồng. Hệ thống cập nhật thời gian hiệu lực và phần trăm hoa hồng theo qui định hiện tại của công ty. |
| DT4 | Quản lý sản phẩm | Cho phép đối tác thêm – xóa – sửa thông tin sản phẩm và chi nhánh có cung cấp sản phẩm này. Hệ thống kiểm tra tính hợp lệ và thông báo thêm – xoá – sửa thành công/thất bại. |
| DT5 | Quản lý đơn hàng | Cho phép đối tác xem thông tin đơn hàng và cập nhật tình trạng của đơn hàng. |

### **Phân hệ khách hàng**

|  |  |  |
| --- | --- | --- |
| **STT** | **Chức năng** | **Mô tả hoạt động** |
| KH1 | Đăng ký làm khách hàng | Khách hàng nhập: Họ tên, số điện thoại, địa chỉ, email để tạo tài khoản đăng nhập. |
| KH2 | Đặt hàng | * Khách hàng xem danh sách đối tác. * Chọn đối tác, hệ thống hiển thị danh sách sản phẩm của đối tác đó. * Chọn sản phẩm, số lượng tương ứng, hình thức thanh toán và địa chỉ giao hàng. * Hệ thống tiến hành tính phí gồm: phí sản phẩm và phí vận chuyển. * Khách hàng xác nhận đồng ý, đơn hàng sẽ được chuyển đến đối tác và tài xế |
| KH3 | Theo dõi đơn hàng | Khách hàng theo dõi quá trình vận chuyển đơn hàng (Gồm các tình trạng: Chờ/ Đang giao/ Đã giao). |

### **Phân hệ tài xế**

|  |  |  |
| --- | --- | --- |
| **STT** | **Chức năng** | **Mô tả hoạt động** |
| TX1 | Đăng ký làm tài xế | Tài xế nhập: Họ tên, CMND, điện thoại, địa chỉ, biển số xe, khu vực hoạt động, email, thông tin tài khoản ngân hàng để nhận tiền. Hệ thống kiểm tra thông tin và yêu cầu người dùng nộp một khoản phí thuế chân. Hệ thống tạo tài khoản sau khi người dùng thanh toán thành công. |
| TX2 | Tiếp nhận đơn hàng | Hệ thống hiển thị danh sách đơn hàng theo khu vực mà tài xế đã đăng ký. Tài xế chọn đơn hàng sẽ giao. |
| TX3 | Cập nhật đơn hàng | Tài xế cập nhật tình trạng đơn hàng lên hệ thống. |
| TX4 | Theo dõi thu nhập | Hệ thống hiển thị danh sách các đơn hàng tài xế đã nhận và phí vận chuyển tài xế được nhận ứng với từng đơn hàng |

### **Phân hệ nhân viên**

|  |  |  |
| --- | --- | --- |
| **STT** | **Chức năng** | **Mô tả hoạt động** |
| NV1 | Xét duyệt hợp đồng | Nhân viên xem những đơn hợp đồng xin duyệt và quyết định duyệt/huỷ bỏ. Hệ thống tiếp nhận quyết định, gửi thông báo huỷ hợp đồng (nếu huỷ bỏ); gửi thông báo hiệu lực của hợp đồng đến đối tác (nếu duyệt). |
| NV2 | Thông báo gia hạn | Nhân viên xem những hợp đồng đã được duyệt và gửi thông báo (trên app) đến người dùng khi hợp đồng sắp hết hạn. |

### **Phân hệ quản trị**

|  |  |  |
| --- | --- | --- |
| **STT** | **Chức năng** | **Mô tả hoạt động** |
| QT1 | Cập nhật thông tin tài khoản | Cập nhật thông tin tài khoản |
| QT2 | Quản lý nhân viên/admin | Thêm – xóa –sửa tài khoản admin, nhân viên |
| QT3 | Mở/Khoá tài khoản | Quản trị có thể khoá tài khoản hoặc kích hoạt lại tài khoản người dùng. |
| QT4 | Cấp quyền thao tác dữ liệu | Quản trị được phép thay đổi quyền thao tác trên dữ liệu của người dùng khác |
| QT5 | Cấp quyền thao tác trên giao diện | Quản trị được phép thay đổi quyền thao tác trên giao diện của người dùng khác |

## **Xác định tình huống tranh chấp**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **STT** | **Chức năng 1** | **Người dùng** | **Chức năng 2** | **Người dùng** | **Lỗi tranh chấp** |
| **1** | **Kiểm tra và xuất số hợp đồng hết hạn** | **Nhân viên A** | **Gia hạn một hợp đồng** | **Nhân viên B** | **Unreapearable data** |
| **2** | **Gửi thông báo và gia hạn hợp đồng** | **Nhân viên** | **Lập hợp đồng và đọc thông báo** | **Đối tác** | **Deadlock cycle** |
| **3** | **Đối tác xem danh sách sản phẩm** | **Đối tác A** | **Đối tác thêm 1 sản phẩm mới vào danh sách** | **Đối tác B** | **Phantom read** |
| **4** | **Tài xế xem danh sách đơn hàng được nhận và chọn nhận đơn X** | **Tài xế A** | **Tài xế xem danh sách đơn hàng được nhận và chọn nhận đơn X** | **Tài xế B** | **Converson deadlock** |
| **5** | **Đặt hàng** | **Khách hàng** | **Xem số lượng đơn hàng trong ngày** | **Đối tác** | **Dirty read** |
| **6** | **Xem và nhận đơn hàng ‘01’** | **Tài xế A** | **Xem và nhận đơn hàng ‘01’** | **Tài xế B** | **Lost update** |

# **Thiết kế giao diện chức năng**

## **Homepage**

Diagram

Description automatically generated

Diagram

Description automatically generated

## **Chức năng: Đăng nhập**

Graphical user interface, application

Description automatically generated

## **Chức năng: Đăng ký (Đối tác)**

Graphical user interface

Description automatically generated

## **Chức năng: Lập hợp đồng (Đối tác)**

Text

Description automatically generated

## **Chức năng: Quản lý sản phẩm và chi nhánh (Đối tác)**

Table

Description automatically generated

## **Chức năng: Theo dõi đơn hàng (Đối tác)**

Table

Description automatically generated with medium confidence

## **Chức năng: Đăng ký (Khách hàng)**

Graphical user interface, application

Description automatically generated

## **Chức năng: Đặt hàng (Khách hàng)**

Graphical user interface, table

Description automatically generated

## **Chức năng: Theo dõi đơn hàng (Khách hàng)**

Graphical user interface, application, table

Description automatically generated

## **Chức năng: Đăng ký làm tài xế**

Graphical user interface, application, website

Description automatically generated

Diagram

Description automatically generated

## **Chức năng: Nhận đơn hàng (Tài xế)**

Table

Description automatically generated with medium confidence

## **Chức năng: Chỉnh sửa tình trang đơn hàng (Tài xế)**

Table

Description automatically generated

## **Chức năng: Theo dõi thu nhập (Tài xế)**

Graphical user interface

Description automatically generated with medium confidence

## **Chức năng: Xét duyệt hợp đồng (Nhân viên)**

Diagram

Description automatically generated

Graphical user interface, text, application, chat or text message

Description automatically generatedTable

Description automatically generated

**Gửi mail thông báo**

## **Chức năng: Gia hạn hợp đồng (Nhân viên)**

Graphical user interface, application

Description automatically generatedTable

Description automatically generated

Gửi mail thông báo

## **Chức năng: Cập nhật thông tin các tài khoản và Thêm xóa sửa tài khoản nhân viên và admin (Quản trị)**

**A picture containing table

Description automatically generated**

## **Chức năng: Khóa và kích hoạt tài khoản (Quản trị)**

Table

Description automatically generated

## **Chức năng: Cấp quyền thao tác trên dữ liệu & giao diện cho người dùng (Quản trị)**

Table

Description automatically generated

# **Cài đặt tình huống tranh chấp**

## **Tình huống 1: Kiểm tra hơp đồng**

|  |  |  |  |
| --- | --- | --- | --- |
| **ERR01: Unrepeatable Data**  T1 (User = Nhân viên A): Kiểm tra Hợp đồng đã hết hạn  T2 (User = Nhân viên B): Gia hạn 1 hợp đồng | | | |
| **sp\_HopDongHetHan** | **Khóa** | **sp\_GiaHanHopDong** | **Khóa** |
| ***Input:***  ***Output***: | ***Input***: @*MaHopDong int, @new date*  ***Output***: |
| SET TRANSACTION ISOLATION  LEVEL **READ UNCOMMITTED** | SET TRANSACTION ISOLATION  LEVEL **READ UNCOMMITTED** |
| BEGIN TRAN |  |  |  |
| B1: Kiểm tra những hợp đồng đã hết hạn  SELECT \*  FROM HopDong  WHERE ThoiGianHieuLuc < GETDATE() | // Không cần xin khoá |  |  |
| WAITFOR DELAY '00:00:05' |  |  |  |
|  |  | BEGIN TRAN |  |
|  |  | B1:Kiểm tra hợp đồng có tồn tại hay không  IF NOT EXISTS(select \* from HopDong where MaHopDong = @MaHopDong)  BEGIN  PRINT N'HopDong' + CAST(@MaHopDong AS VARCHAR(10)) + N' Không Tồn Tại'  ROLLBACK TRAN  RETURN 1  END |  |
|  |  | B2: Gia hạn hợp đồng  UPDATE HopDong SET ThoiGianHieuLuc = @newWHERE MaHopDong = @MaHopDong | **X(HopDong)**  // Xin khoá ghi trên bảng Hợp đồng |
| B2: Xuất số lượng hợp đồng hết hạn  SELECT @HDHH = COUNT(\*)  FROM HopDong  WHERE ThoiGianHieuLuc < GETDATE() | **S(HopDong)**  // Xin khoá đọc trên bảng Hợp đồng |  |  |
| COMMIT |  |  |  |
|  |  | COMMIT |  |

## **Tình huống 2: Xem và Nhận đơn hàng**

|  |  |  |  |
| --- | --- | --- | --- |
| **ERR01: Converson Deadlock**  T1 (User = Tài xế 1): Đọc và nhận đơn hàng  T2 (User = Tài xế 2): Đọc và nhận đơn hàng | | | |
| **SP\_NhanDonHang\_deadlock** | **Khóa** | **SP\_NhanDonHang\_deadlock** | **Khóa** |
| ***Input:*** @MaTaiXe int, @MaDonHang int | ***Input***: @MaTaiXe int, @MaDonHang int |
| SET TRANSACTION ISOLATION  LEVEL **REPEATABLE READ** | SET TRANSACTION ISOLATION  LEVEL **REPEATABLE READ** |
| BEGIN TRAN |  |  |  |
| IF NOT EXISTs(select \* from TaiXe where MaTaiXe = @MaTaiXe)  BEGIN  PRINT N'Tài Xế' + CAST(@MaTaiXe AS VARCHAR(10)) + N' Không Tồn Tại'  ROLLBACK TRAN  RETURN 1  END | **S(TaiXe)**  // Xin khoá đọc trên bảng tài xế |  |  |
| IF NOT EXISTS(select \* from DonHang,TaiXe where DonHang.KhuVuc = Taixe.KhuVucHoatDong and DonHang.MaTaiXe is null and Taixe.MaTaiXe = @MaTaiXe and TinhTrang = N'Chờ' and MaDonHang = @MaDonHang)  BEGIN  PRINT N'Đặt hàng không khả thi'  ROLLBACK TRAN  RETURN 1  END | **S(TaiXe)**  **S(DonHang)**  // Xin khoá đọc trên bảng tài xế và đơn hàng |  |  |
| WAITFOR DELAY '00:00:05' |  |  |  |
|  |  | IF NOT EXISTs(select \* from TaiXe where MaTaiXe = @MaTaiXe)  BEGIN  PRINT N'Tài Xế' + CAST(@MaTaiXe AS VARCHAR(10)) + N' Không Tồn Tại'  ROLLBACK TRAN  RETURN 1  END | **S(TaiXe)**  // Xin khoá đọc trên bảng tài xế |
|  |  | IF NOT EXISTS(select \* from DonHang,TaiXe where DonHang.KhuVuc = Taixe.KhuVucHoatDong and DonHang.MaTaiXe is null and Taixe.MaTaiXe = @MaTaiXe and TinhTrang = N'Chờ' and MaDonHang = @MaDonHang)  BEGIN  PRINT N'Đặt hàng không khả thi'  ROLLBACK TRAN  RETURN 1  END | **S(TaiXe)**  **S(DonHang)**  // Xin khoá đọc trên bảng tài xế và đơn hàng |
|  |  | WAITFOR DELAY '00:00:05' |  |
| update DonHang set TinhTrang = N'Đang Giao',MaTaiXe = @MaTaiXe where @MaDonHang = MaDonHang | **X(DonHang)** |  |  |
|  |  | update DonHang set TinhTrang = N'Đang Giao',MaTaiXe = @MaTaiXe where @MaDonHang = MaDonHang | **X(DonHang)** |
| COMMIT |  |  |  |
|  |  | COMMIT |  |

## **Tình huống 3: Hai tài xế cùng nhận chung một đơn hàng**

|  |  |  |  |
| --- | --- | --- | --- |
| **ERR01: Lost update**  T1 (User = Tài xế A): Xem và nhận đơn hàng  T2 (User = Tài xế B): Xem và nhận đơn hàng | | | |
| **sp\_NhanDonHang** | **Khóa** | **sp\_NhanDonHang** | **Khóa** |
| ***Input:*** *@MaTaiXe int,*  *@MaDonHang int*  ***Output***: | ***Input***: *@MaTaiXe int, @MaDonHang int*  ***Output***: |
| SET TRANSACTION ISOLATION  LEVEL **READ COMMITTED** | SET TRANSACTION ISOLATION  LEVEL **READ COMMITTED** |
| BEGIN TRAN |  |  |  |
| B1: Kiểm tra tài xế có tồn tại hay không  IF NOT EXISTS (select \* from TaiXe where MaTaiXe = @MaTaiXe)  BEGIN  PRINT N'Tài Xế' + CAST(@MaTaiXe AS VARCHAR(10)) + N' Không Tồn Tại'  ROLLBACK TRAN  RETURN 1  END | **S(Tài Xế)**  //Xin khoá đọc trên bảng Tài xế |  |  |
| B2: Kiểm tra đơn hàng có thuộc khu vực của tài xế  IF NOT EXISTS  (select \* from DonHang,TaiXe  where DonHang.KhuVuc = KhuVucHoatDong  and Taixe.MaTaiXe = @MaTaiXe  and TinhTrang = N'Chờ' and DonHang.MaTaiXe is null  and MaDonHang = @MaDonHang)  BEGIN  PRINT N'Đặt hàng không khả thi'  ROLLBACK TRAN  RETURN 1  END | **S(Đơn hàng)**  //Xin khoá đọc trên bảng Tài xế |  |  |
| WAITFOR DELAY '00:00:05' |  |  |  |
|  |  | BEGIN TRAN |  |
|  |  | B1: Kiểm tra tài xế có tồn tại hay không  IF NOT EXISTS (select \* from TaiXe where MaTaiXe = @MaTaiXe)  BEGIN  PRINT N'Tài Xế' + CAST(@MaTaiXe AS VARCHAR(10)) + N' Không Tồn Tại'  ROLLBACK TRAN  RETURN 1  END | **S(DonHang)**  //Xin khoá đọc trên bảng đơn hàng |
|  |  | B2: Kiểm tra đơn hàng có thuộc khu vực của tài xế  IF NOT EXISTS  (select \* from DonHang,TaiXe  where DonHang.KhuVuc = KhuVucHoatDong  and Taixe.MaTaiXe = @MaTaiXe  and TinhTrang = N'Chờ'  and MaDonHang = @MaDonHang)  BEGIN  PRINT N'Đặt hàng không khả thi'  ROLLBACK TRAN  RETURN 1  END | **S(Đơn hàng)**  //Xin khoá đọc trên bảng Tài xế |
|  |  | WAITFOR DELAY '00:00:05' |  |
| B3: Đặt hàng  update DonHang set TinhTrang = N'Đang Giao' where @MaDonHang = MaDonHang  update DonHang set MaTaiXe = @MaTaiXe where @MaDonHang = MaDonHang | **X(DonHang)**  //Xin khoá ghi trên bảng đơn hàng |  |  |
|  |  | B3: Đặt hàng  update DonHang set TinhTrang = N'Đang Giao' where @MaDonHang = MaDonHang  update DonHang set MaTaiXe = @MaTaiXe where @MaDonHang = MaDonHang | **X(DonHang)**  //Xin khoá ghi trên bảng đơn hàng |
| COMMIT |  |  |  |
|  |  | COMMIT |  |

## **Tình huống 4: Khách hàng đặt hàng thất bại nhưng đối tác vẫn thấy đơn hàng**

|  |  |  |  |
| --- | --- | --- | --- |
| **ERR01: Dirty Read**  T1 (User = Khách Hàng): Đặt hàng  T2 (User = Đối tác): Xem số lượng đơn hàng trong ngày | | | |
| **sp\_DatHang** | **Khóa** | **sp\_XemSoLuongDonHang** | **Khóa** |
| ***Input:*** @MaSoThue *int*, @MaKhachHang *int*,  @KhuVuc nvarchar(50)  ***Output***: |  | ***Input***: @MaSoThue *int*  ***Output***: |
| SET TRANSACTION ISOLATION  LEVEL **READ COMMITTED** | SET TRANSACTION ISOLATION  LEVEL **READ UNCOMMITTED** |
| BEGIN TRAN |  |  |  |
| B1: Kiểm tra thông tin về đối tác  IF NOT EXISTS (select \* from DoiTac where MaSoThue = @MaSoThue)  BEGIN  PRINT N'Đối tác' + CAST(@MaSoThue AS VARCHAR(10)) + N' Không Tồn Tại'  ROLLBACK TRAN  RETURN 1  END |  |  |  |
| INSERT INTO DonHang (MaSoThue, MaKhachHang, TinhTrang, KhuVuc, NgayDat)  values  (@MaSoThue, @MaKhachHang, N'Chờ', @KhuVuc, getdate()) | **X(Donhang)**  //xin khoá ghi trên bảng đơn hàng |  |  |
| WAITFOR DELAY '00:00:05' |  |  |  |
|  |  | BEGIN TRAN |  |
|  |  | IF NOT EXISTS(select \* from DoiTac where MaSoThue = @MaSoThue)  BEGIN  PRINT N'Đối tác' + CAST(@MaSoThue AS VARCHAR(10)) + N' Không Tồn Tại'  ROLLBACK TRAN  RETURN 1  END | //Không xin khoá |
|  |  | select \* from DonHang where MaSoThue = @MaSoThue  and DAY(DonHang.NgayDat) = DAY(getdate())  and MONTH (DonHang.NgayDat) = MONTH(getdate())  and YEAR(DonHang.NgayDat) = YEAR(getdate()) |  |
| IF @@error<>0  //Khi insert đến bảng DonHang, Thì số lượng đơn hàng quá số lượng đơn mà đối tác có thể cung cấp, nên buộc phải huỷ đơn của khách hàng  ROLLBACK |  |  |  |
|  |  | COMMIT TRAN |  |
| COMMIT TRAN |  |  |  |

## **Tình huống 5: Khách hàng đang xem sản phẩm thì thêm 1 sản phẩm khác vào**

|  |  |  |  |
| --- | --- | --- | --- |
| **ERR03: *Phantom read***  T1 (User = Khách hàng A): thực hiện xem danh sách sản phẩm của đối tác B  T2 (User = Đối tác B): thực hiện thêm 1 sản phẩm vào danh sách sản phẩm | | | |
| **sp\_XemSanPham** | **Khóa** | **sp\_ThemSanPham** | **Khóa** |
| ***Input:*** @MaSoThue *int*  ***Output***: | ***Input***: @MaSP *int*,  @MaChiNhanh *int*,  @SoLuongTon int,  @GiaCa int,  @TenSP varchar(50)  ***Output***: |
| SET TRANSACTION ISOLATION  LEVEL **READ COMMITTED** | SET TRANSACTION ISOLATION  LEVEL **REPEATEABLE READ** |
| BEGIN TRAN |  |  |  |
| B1: Xem danh sách các sản phẩm do đối tác B cung cấp  SELECT \* FROM SanPham  WHERE MaChiNhanh IN (SELECT MaChiNhanh FROM ChiNhanh WHERE MaSoThue=@MaSoThue) | **S(SanPham)**  **S(ChiNhanh)**  //Xin khoá đọc trên bảng sản phẩm và bảng chi nhánh |  |  |
| WAITFOR DELAY '00:00:20' |  |  |  |
|  |  | B1: Thêm 1 sản phẩm vào danh sách sản phẩm  INSERT SanPham(MaSP, MaChiNhanh, SoLuongTon, GiaCa, TenSP) VALUES (@MaSP, @MaChiNhanh, @SoLuongTon, @GiaCa, @TenSP) | //Không khoá ghi trên bảng sản phẩm |
| B2: Xem lại danh sách các sản phẩm do đối tác B cung cấp  SELECT \* FROM SanPham  WHERE MaChiNhanh IN (SELECT MaChiNhanh FROM ChiNhanh WHERE MaSoThue=@MaSoThue) | **S(SanPham)**  **S(ChiNhanh)**  //Xin khoá đọc trên bảng sản phẩm và bảng chi nhánh |  |  |
| COMMIT |  |  |  |
|  |  | COMMIT |  |

## **Tình huống 6: Đối tác lập hợp đồng xong đọc thông báo, Nhân viên gửi thông báo tới đối tác xong xem hợp đồng**



|  |  |  |  |
| --- | --- | --- | --- |
| **ERR04: *Cycle Deadlock***  T1 (User = Đối tác A): Lập hợp đồng xong đọc thông báo  T2 (User = Nhân viên B): Gửi thông báo chúc mừng tới đối tác xong xem hợp đồng | | | |
| **sp\_LapHopDong\_DocThongBao** | **Khóa** | **sp\_ThongBao\_HopDong** | **Khóa** |
| ***Input:*** @MaSoThue int  ***Output***: | ***Input***  @NoiDung nvarchar(50), @MaSoThue int  ***Output***: |
| SET TRANSACTION ISOLATION  LEVEL REPEATABLE READ | SET TRANSACTION ISOLATION  LEVEL REPEATABLE READ |
| BEGIN TRAN |  |  |  |
| INSERT INTO HopDong(MaSoThue) VALUES (@MaSoThue) | **X(HopDong)** |  |  |
| WAITFOR DELAY '00:00:05' |  |  |  |
|  |  | INSERT INTO ThongBao(NoiDung,MasoThue) VALUES (@NoiDung,@MaSoThue) | **X(ThongBao)** |
|  |  | WAITFOR DELAY '00:00:05' |  |
| SELECT NoiDung  FROM ThongBao  WHERE MaSoThue = @MaSoThue | **S(ThongBao)** |  |  |
|  |  | select \* from hopdong | **S(HopDong)** |
| COMMIT |  |  |  |
|  |  | COMMIT |  |

# **Xử lý tình huống tranh chấp**

## **Tình huống 1: Kiểm tra hơp đồng**

|  |  |  |  |
| --- | --- | --- | --- |
| **ERR01: Unrepeatable Data**  T1 (User = Nhân viên A): Kiểm tra Hợp đồng đã hết hạn  T2 (User = Nhân viên B): Gia hạn 1 hợp đồng | | | |
| **sp\_HopDongHetHan** | **Khóa** | **sp\_GiaHanHopDong** | **Khóa** |
| ***Input:***  ***Output***: | ***Input***: @*MaHopDong int, @new date*  ***Output***: |
| SET TRANSACTION ISOLATION  LEVEL **REPEATABLE READ** | SET TRANSACTION ISOLATION  LEVEL **REPEATABLE READ** |
| BEGIN TRAN |  |  |  |
| B1: Kiểm tra những hợp đồng đã hết hạn  SELECT \*  FROM HopDong  WHERE ThoiGianHieuLuc < GETDATE() | **S(HopDong)**  //Xin khoá đọc trên bảng hợp đồng |  |  |
| WAITFOR DELAY '00:00:05' |  |  |  |
|  |  | BEGIN TRAN |  |
|  |  | B1:Kiểm tra hợp đồng có tồn tại hay không  IF NOT EXISTS(select \* from HopDong where MaHopDong = @MaHopDong)  BEGIN  PRINT N'HopDong' + CAST(@MaHopDong AS VARCHAR(10)) + N' Không Tồn Tại'  ROLLBACK TRAN  RETURN 1  END | **S(HopDong)**  //Xin khoá đọc trên bảng hợp đồng |
|  |  | B2: Gia hạn hợp đồng  UPDATE HopDong SET ThoiGianHieuLuc = @newWHERE MaHopDong = @MaHopDong | **X(HopDong)**  // Xin khoá ghi trên bảng Hợp đồng |
| B2: Xuất số lượng hợp đồng hết hạn  SELECT @HDHH = COUNT(\*)  FROM HopDong  WHERE ThoiGianHieuLuc < GETDATE() | **S(HopDong)**  // Xin khoá đọc trên bảng Hợp đồng |  |  |
| COMMIT |  |  |  |
|  |  | COMMIT |  |

## **Tình huống 2: Nhận và nhận đơn hàng**

|  |  |  |  |
| --- | --- | --- | --- |
| **ERR01: Converson Deadlock**  T1 (User = Tài xế A): Xem và nhận đơn hàng  T2 (User = Tài xế B): Xem và nhận đơn hàng | | | |
| **sp\_NhanDonHang** | **Khóa** | **sp\_NhanDonHang** | **Khóa** |
| ***Input:*** *@MaTaiXe int, @MaDonHang int*  ***Output***: | ***Input***: *@MaTaiXe int, @MaDonHang int*  ***Output***: |
|  |  |
| BEGIN TRAN |  |  |  |
| B1: Kiểm tra tài xế có tồn tại hay không  IF NOT EXISTS (select \* from TaiXe where MaTaiXe = @MaTaiXe)  BEGIN  PRINT N'Tài Xế' + CAST(@MaTaiXe AS VARCHAR(10)) + N' Không Tồn Tại'  ROLLBACK TRAN  RETURN 1  END | **S(Tài Xế)**  //Xin khoá đọc trên bảng Tài xế |  |  |
| B2: Kiểm tra đơn hàng có thuộc khu vực của tài xế  IF NOT EXISTS  (select \* from DonHang,TaiXe  where DonHang.KhuVuc = KhuVucHoatDong  and Taixe.MaTaiXe = @MaTaiXe  and TinhTrang = N'Chờ'  and DonHang.MaTaiXe is null  and MaDonHang = @MaDonHang)  BEGIN  PRINT N'Đặt hàng không khả thi'  ROLLBACK TRAN  RETURN 1  END | **S(Đơn hàng)**  //Xin khoá đọc trên bảng Tài xế |  |  |
| WAITFOR DELAY '00:00:05' |  |  |  |
|  |  | BEGIN TRAN |  |
|  |  | B1: Kiểm tra tài xế có tồn tại hay không  IF NOT EXISTS (select \* from TaiXe where MaTaiXe = @MaTaiXe)  BEGIN  PRINT N'Tài Xế' + CAST(@MaTaiXe AS VARCHAR(10)) + N' Không Tồn Tại'  ROLLBACK TRAN  RETURN 1  END | **S(DonHang)**  //Xin khoá đọc trên bảng đơn hàng |
|  |  | B2: Kiểm tra đơn hàng có thuộc khu vực của tài xế  IF NOT EXISTS  (select \* from DonHang,TaiXe  where DonHang.KhuVuc = KhuVucHoatDong  and Taixe.MaTaiXe = @MaTaiXe  and TinhTrang = N'Chờ'  and DonHang.MaTaiXe is null  and MaDonHang = @MaDonHang)  BEGIN  PRINT N'Đặt hàng không khả thi'  ROLLBACK TRAN  RETURN 1  END | **S(Đơn hàng)**  //Xin khoá đọc trên bảng Tài xế |
|  |  | WAITFOR DELAY '00:00:05' |  |
| B3: Đặt hàng  IF (exists(select \* from donhang where madonhang = @MaDonHang and TinhTrang = N'Chờ'))  BEGIN  update DonHang set TinhTrang = N'Đang Giao', MaTaiXe = @MaTaiXe where @MaDonHang = MaDonHang  END  ELSE  BEGIN  PRINT N'không khả thi'  ROLLBACK TRAN  RETURN 1  END | **X(DonHang)**  //Xin khoá ghi trên bảng đơn hàng |  |  |
|  |  | B3: Đặt hàng  IF (exists(select \* from donhang where madonhang = @MaDonHang and TinhTrang = N'Chờ'))  BEGIN  update DonHang set TinhTrang = N'Đang Giao', MaTaiXe = @MaTaiXe where @MaDonHang = MaDonHang  END  ELSE  BEGIN  PRINT N'không khả thi'  ROLLBACK TRAN  RETURN 1  END | **X(DonHang)**  //Xin khoá ghi trên bảng đơn hàng |
| COMMIT |  |  |  |
|  |  | COMMIT |  |

## **Tình huống 3: Hai tài xế cùng nhận chung một đơn hàng**

|  |  |  |  |
| --- | --- | --- | --- |
| **ERR01: Lost update**  T1 (User = Tài xế A): Xem và nhận đơn hàng  T2 (User = Tài xế B): Xem và nhận đơn hàng | | | |
| **sp\_NhanDonHang** | **Khóa** | **sp\_NhanDonHang** | **Khóa** |
| ***Input:*** *@MaTaiXe int, @MaDonHang int*  ***Output***: | ***Input***: *@MaTaiXe int, @MaDonHang int*  ***Output***: |
|  |  |
| BEGIN TRAN |  |  |  |
| B1: Kiểm tra tài xế có tồn tại hay không  IF NOT EXISTS (select \* from TaiXe where MaTaiXe = @MaTaiXe)  BEGIN  PRINT N'Tài Xế' + CAST(@MaTaiXe AS VARCHAR(10)) + N' Không Tồn Tại'  ROLLBACK TRAN  RETURN 1  END | **S(Tài Xế)**  //Xin khoá đọc trên bảng Tài xế |  |  |
| B2: Kiểm tra đơn hàng có thuộc khu vực của tài xế  IF NOT EXISTS  (select \* from DonHang,TaiXe  where DonHang.KhuVuc = KhuVucHoatDong  and Taixe.MaTaiXe = @MaTaiXe  and TinhTrang = N'Chờ'  and DonHang.MaTaiXe is null  and MaDonHang = @MaDonHang)  BEGIN  PRINT N'Đặt hàng không khả thi'  ROLLBACK TRAN  RETURN 1  END | **S(Đơn hàng)**  //Xin khoá đọc trên bảng Tài xế |  |  |
| WAITFOR DELAY '00:00:05' |  |  |  |
|  |  | BEGIN TRAN |  |
|  |  | B1: Kiểm tra tài xế có tồn tại hay không  IF NOT EXISTS (select \* from TaiXe where MaTaiXe = @MaTaiXe)  BEGIN  PRINT N'Tài Xế' + CAST(@MaTaiXe AS VARCHAR(10)) + N' Không Tồn Tại'  ROLLBACK TRAN  RETURN 1  END | **S(DonHang)**  //Xin khoá đọc trên bảng đơn hàng |
|  |  | B2: Kiểm tra đơn hàng có thuộc khu vực của tài xế  IF NOT EXISTS  (select \* from DonHang,TaiXe  where DonHang.KhuVuc = KhuVucHoatDong  and Taixe.MaTaiXe = @MaTaiXe  and TinhTrang = N'Chờ'  and DonHang.MaTaiXe is null  and MaDonHang = @MaDonHang)  BEGIN  PRINT N'Đặt hàng không khả thi'  ROLLBACK TRAN  RETURN 1  END | **S(Đơn hàng)**  //Xin khoá đọc trên bảng Tài xế |
|  |  | WAITFOR DELAY '00:00:05' |  |
| B3: Đặt hàng  IF (exists(select \* from donhang where madonhang = @MaDonHang and TinhTrang = N'Chờ'))  BEGIN  update DonHang set TinhTrang = N'Đang Giao', MaTaiXe = @MaTaiXe where @MaDonHang = MaDonHang  END  ELSE  BEGIN  PRINT N'không khả thi'  ROLLBACK TRAN  RETURN 1  END | **X(DonHang)**  //Xin khoá ghi trên bảng đơn hàng |  |  |
|  |  | B3: Đặt hàng  IF (exists(select \* from donhang where madonhang = @MaDonHang and TinhTrang = N'Chờ'))  BEGIN  update DonHang set TinhTrang = N'Đang Giao', MaTaiXe = @MaTaiXe where @MaDonHang = MaDonHang  END  ELSE  BEGIN  PRINT N'không khả thi'  ROLLBACK TRAN  RETURN 1  END | **X(DonHang)**  //Xin khoá ghi trên bảng đơn hàng |
| COMMIT |  |  |  |
|  |  | COMMIT |  |

## **Tình huống 4: Khách hàng đặt hàng thất bại nhưng đối tác vẫn thấy đơn hàng**

|  |  |  |  |
| --- | --- | --- | --- |
| **ERR01: Dirty Read**  T1 (User = Khách Hàng): Đặt hàng  T2 (User = Đối tác): Xem số lượng đơn hàng trong ngày | | | |
| **sp\_DatHang** | **Khóa** | **sp\_XemSoLuongDonHang** | **Khóa** |
| ***Input:*** @MaSoThue *int*, @MaKhachHang *int*,  @KhuVuc nvarchar(50)  ***Output***: |  | ***Input***: @MaSoThue *int*  ***Output***: |
| SET TRANSACTION ISOLATION  LEVEL **REPEATABLE READ** | SET TRANSACTION ISOLATION  LEVEL **REPEATABLE READ** |
| BEGIN TRAN |  |  |  |
| B1: Kiểm tra thông tin về đối tác  IF NOT EXISTS (select \* from DoiTac where MaSoThue = @MaSoThue)  BEGIN  PRINT N'Đối tác' + CAST(@MaSoThue AS VARCHAR(10)) + N' Không Tồn Tại'  ROLLBACK TRAN  RETURN 1  END | **S(DoiTac)**  //xin khoá đọc trên bảng đơn hàng |  |  |
| B1:2 Đặt hàng  INSERT INTO DonHang (MaSoThue, MaKhachHang, TinhTrang, KhuVuc, NgayDat)  values  @MaSoThue, @MaKhachHang, N'Chờ', @KhuVuc, getdate()) | **X(Donhang)**  //xin khoá ghi trên bảng đơn hàng |  |  |
| WAITFOR DELAY '00:00:05' |  |  |  |
|  |  | BEGIN TRAN |  |
|  |  | IF NOT EXISTS(select \* from DoiTac where MaSoThue = @MaSoThue)  BEGIN  PRINT N'Đối tác' + CAST(@MaSoThue AS VARCHAR(10)) + N' Không Tồn Tại'  ROLLBACK TRAN  RETURN 1  END | **S(DoiTac)**  //xin khoá đọc trên bảng đơn hàng |
|  |  | select \* from DonHang where MaSoThue = @MaSoThue  and DAY(DonHang.NgayDat) = DAY(getdate())  and MONTH (DonHang.NgayDat) = MONTH(getdate())  and YEAR(DonHang.NgayDat) = YEAR(getdate()) | **S(Donhang)**  //xin khoá đọc trên bảng đơn hàng |
| IF @@error<>0  //Khi insert đến bảng DonHang, Thì số lượng đơn hàng quá số lượng đơn mà đối tác có thể cung cấp, nên buộc phải huỷ đơn của khách hàng  ROLLBACK |  |  |  |
|  |  | COMMIT TRAN |  |
| COMMIT TRAN |  |  |  |

## **Tình huống 5: Khách hàng đang xem sản phẩm thì thêm 1 sản phẩm khác vào**

|  |  |  |  |
| --- | --- | --- | --- |
| **ERR03: *Phantom read***  T1 (User = Khách hàng A): thực hiện xem danh sách sản phẩm của đối tác B  T2 (User = Đối tác B): thực hiện thêm 1 sản phẩm vào danh sách sản phẩm | | | |
| **sp\_XemSanPham** | **Khóa** | **sp\_ThemSanPham** | **Khóa** |
| ***Input:*** @MaSoThue *int*  ***Output***: | ***Input***: @MaSP *int*,  @MaChiNhanh varchar(10),  @SoLuongTon int,  @GiaCa int,  @TenSP varchar(50)  ***Output***: |
| SET TRANSACTION ISOLATION  LEVEL **SERIALIZABLE** |  |
| BEGIN TRAN |  |  |  |
| B1: Xem danh sách các sản phẩm do đối tác B cung cấp  SELECT \* FROM SanPham  WHERE MaChiNhanh IN (SELECT MaChiNhanh FROM ChiNhanh WHERE MaSoThue=@MaSoThue) | **S(SanPham)**  **S(ChiNhanh)**  //Xin khoá đọc trên bảng sản phẩm và bảng chi nhánh |  |  |
| WAITFOR DELAY '00:00:20' |  |  |  |
|  |  | B1: Thêm 1 sản phẩm vào danh sách sản phẩm  INSERT SanPham(MaSP, MaChiNhanh, SoLuongTon, GiaCa, TenSP) With(writelock) VALUES (@MaSP, @MaChiNhanh, @SoLuongTon, @GiaCa, @TenSP) | **X(SanPham)**  //sinh khoá ghi trên bảng sản phẩm |
| B2: Xem lại danh sách các sản phẩm do đối tác B cung cấp  SELECT \* FROM SanPham  WHERE MaChiNhanh IN (SELECT MaChiNhanh FROM ChiNhanh WHERE MaSoThue=@MaSoThue) | **S(SanPham)**  **S(ChiNhanh)**  //Xin khoá đọc trên bảng sản phẩm và bảng chi nhánh |  |  |
| COMMIT |  |  |  |
|  |  | COMMIT |  |

## **Tình huống 6: Đối tác lập hợp đồng xong đọc thông báo, Nhân viên gửi thông báo tới đối tác xong xem hợp đồng**



|  |  |  |  |
| --- | --- | --- | --- |
| **ERR04: *Cycle Deadlock***  T1 (User = Đối tác A): Lập hợp đồng xong đọc thông báo  T2 (User = Nhân viên B): Gửi thông báo chúc mừng tới đối tác xong xem hợp đồng | | | |
| **sp\_LapHopDong\_DocThongBao** | **Khóa** | **sp\_ThongBao\_HopDong** | **Khóa** |
| ***Input:*** @MaSoThue int  ***Output***: | ***Input***  @NoiDung nvarchar(50), @MaSoThue int  ***Output***: |
| SET TRANSACTION ISOLATION  LEVEL READ UNCOMMITED | SET TRANSACTION ISOLATION  LEVEL READ UNCOMMITED |
| BEGIN TRAN |  |  |  |
| INSERT INTO HopDong(MaSoThue) VALUES (@MaSoThue) | **X(HopDong)** |  |  |
| WAITFOR DELAY '00:00:05' |  |  |  |
|  |  | INSERT INTO ThongBao(NoiDung,MasoThue) VALUES (@NoiDung,@MaSoThue) | **X(ThongBao)** |
|  |  | WAITFOR DELAY '00:00:05' |  |
| SELECT NoiDung  FROM ThongBao  WHERE MaSoThue = @MaSoThue | **S(ThongBao)** |  |  |
|  |  | select \* from hopdong | **S(HopDong)** |
| COMMIT |  |  |  |
|  |  | COMMIT |  |

# **THAM KHẢO**