

ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN
KHOA HỆ THÔNG THÔNG TIN



ĐỒ ÁN MÔN HỆ QUẢN TRỊ CƠ SỞ DỮ LIỆU
XÂY DỰNG HỆ THÔNG QUẢN LÝ NHÀ SÁCH HẢI AN

GIẢNG VIÊN HƯỚNG DẪN: NGUYỄN HỒ DUY TRI

NHÓM: 1

THÀNH VIÊN:

- NGUYỄN QUỐC TRANG – 21521556
- NGUYỄN TRIỆU VY – 21522812
- HUỲNH LÊ PHONG – 21520086

Tp. Hồ Chí Minh, tháng 5 năm 2023

LỜI CẢM ƠN

Trên hành trình học hỏi và trau dồi tri thức luôn có sự giúp đỡ dù ít hay nhiều, trực tiếp hay gián tiếp của người thầy. Với lòng biết ơn sâu sắc, nhóm 1 tại em xin gửi lời cảm ơn đến với tập thể quý thầy cô trường Đại học Công nghệ Thông tin - Đại học Quốc gia TP.HCM và quý thầy cô Khoa Hệ thống Thông tin đã truyền đạt những kiến thức giúp nhóm tại em thực hiện đề tài này.

Đặc biệt Nhóm 1 xin gửi lời cảm ơn chân thành đối với thầy Nguyễn Hồ Duy Tri - giảng viên dạy thực hành môn Hệ Quản Trị Cơ Sở Dữ Liệu đã nhiệt tình hỗ trợ, góp ý và hướng dẫn cho nhóm trong suốt quá trình học và thực hiện đồ án môn học. Nếu không có sự chỉ bảo tận tâm của thầy, nhóm tại em sẽ gặp nhiều khó khăn trong quá trình thực hiện đồ án.

Với mục đích hiểu rõ và sâu về cách thức phân tích và thiết kế khi làm một hệ thống quản lý, cũng như tìm hiểu thêm về việc quản lý một hiệu sách, nhóm tại em đã chọn thực hiện đồ án “Quản lý hiệu sách”. Dựa trên những kiến thức được thầy cung cấp, cùng với việc tự học và tự tìm hiểu thêm các công cụ và kiến thức mới, nhóm đã cố gắng thực hiện đồ án một cách tốt nhất. Tuy nhiên, trong quá trình làm không thể tránh khỏi những thiếu sót. Chính vì vậy, nhóm em mong nhận được những nhận xét và góp ý của thầy để bổ sung và hoàn chỉnh đồ án tốt hơn, đồng thời cũng rút ra được thêm kinh nghiệm cho bản thân và của nhóm.

Cuối cùng, nhóm em xin chúc thầy luôn khỏe mạnh để tiếp tục sứ mệnh của nghề giáo là truyền đạt kiến thức cho các bạn sinh viên.

Xin chân thành cảm ơn thầy Duy Tri !

Nhóm 1

NHẬN XÉT CỦA GIÁO VIÊN

MỤC LỤC

PHẦN I. TỔNG QUAN ĐỒ ÁN	2
1. Phát biểu bài toán.....	2
1.1. Tóm tắt nội dung đề tài (bản tiếng Anh)	2
1.2. Lí do chọn đề tài.....	3
2. Xác định và phân tích yêu cầu	3
2.1. Xác định yêu cầu.....	3
2.2. Phân tích yêu cầu.....	4
3. Thiết kế mô hình quan hệ	7
3.1. Mô hình ERD (Thực thể mối kết hợp).....	7
3.2. Mô hình quan hệ	8
3.3. Bảng thuyết minh quan hệ và thuộc tính	8
3.4. Mô tả ràng buộc toàn vẹn	15
PHẦN II. XÂY DỰNG VÀ QUẢN LÝ GIAO TÁC.....	23
1. Qui định hệ thống giao tác.....	23
2. Xây dựng và mô tả Trigger trong đồ án môn học	24
3. Xây dựng và mô tả Procedure, Function trong đồ án môn học	45
3.1 Procedure	45
3.2 Function	54
PHẦN III. XỬ LÝ ĐỒNG THỜI	62
1. Tổng quan	62
a. Giao tác trong Oracle	62
b. Vấn đề trong môi trường truy xuất đồng thời	62
c. Các phương thức khóa cơ bản	63
d. Mức cô lập.....	63
2. Mô tả kịch bản gây mất nhất quán dữ liệu và giải pháp đề ra trong đồ án môn học	65
a. Trường hợp Lost update	65
b. Trường hợp Dirty Read.....	78
c. Trường hợp Non-repeatable Read	78

d. Trường hợp Phantom Read	85
PHẦN IV. THIẾT KẾ GIAO DIỆN	95
1. Thiết kế giao diện	95
1.1. Danh sách các giao diện	95
1.2. Giao diện đăng nhập.....	97
1.3. Giao diện trang chủ	97
1.4. Giao diện trang tài khoản	99
1.5. Giao diện trang hóa đơn	100
1.6. Giao diện trang chi tiết hóa đơn.....	101
1.7. Giao diện trang sách.....	102
1.8. Giao diện trang quản lý.....	103
1.9. Giao diện trang quản lý nhân viên.....	104
1.10. Giao diện trang quản lý khách hàng	106
1.11. Giao diện trang quản lý sách.....	108
1.12. Giao diện trang quản lý tài khoản	110
1.13. Giao diện trang quản lý phiếu nhập.....	111
1.14. Giao diện trang chi tiết phiếu nhập	112
1.15. Giao diện trang quản lý nhà cung cấp	113
1.16. Giao diện trang thống kê	114
2. Thiết kế Report	116
2.1. Giao diện report in hóa đơn.....	116
PHẦN V. KẾT LUẬN	117
1. Bảng phân công công việc	117
2. Môi trường phát triển và môi trường triển khai.....	118
2.2. Môi trường phát triển	118
2.3. Môi trường triển khai.....	118
3. Kết quả đạt được	118
3.1. Kết quả đạt được.....	118
3.2. Khó khăn	119
4. Hướng phát triển	119
5. Tài liệu tham khảo.....	119

PHẦN I. TỔNG QUAN ĐỒ ÁN

1. Phát biểu bài toán

1.1. Tóm tắt nội dung đề tài (bản tiếng Anh)

PROJECT CONTENT

With the trend of science and technology in development, the demand for people is increasing. Therefore, managing a large number of consumers becomes complex and challenging, especially for large enterprises. Without management tools and data storage support, data loss or mistakes can limit business activities and revenue. This is necessary to address the issues of managing a large amount of information for businesses. As a result, our group decided to develop a management system for the Hai An bookstore, a famous bookstore in Ho Chi Minh City.

Building a management system for the Hai An bookstore that meets the following requirement:

- Allow for the addition, deletion, editing, refreshing, and searching of data.
- Allow for the management of employee accounts with their account information and user permissions when logging in to use the system and recording the sales statistics of employees.
- Allow for the management information of books, employees, customers, and customers' invoices.
- Allow for the management of purchase orders and purchase order details, as well as supplier information.
- Allow for the printing of invoices and purchase orders in the bookstore.
- All information is stored in the database, allowing for easy retrieval, backup, and recovery to provide a synchronised, fast, and reliable system.

1.2. Lí do chọn đề tài

Với xu hướng khoa học - công nghệ đang trên đà phát triển, nhu cầu của con người ngày càng tăng cao. Các trung tâm mua sắm, nhà hàng, trung tâm giáo dục,... không ngừng tăng lên để đáp ứng được yêu cầu người dùng. Tuy nhiên, đối với các doanh nghiệp lớn, việc quản lý số lượng khách hàng đông đảo có thể trở nên phức tạp và khó khăn. Để giải quyết vấn đề này cần có 1 hệ thống quản lý hiệu quả.

Trước khi có hệ thống quản lý, các cửa hàng như nhà sách, siêu thị,... đối mặt với nhiều thách thức trong việc quản lý thông tin người dùng, nhân viên, nhà sản xuất và sản phẩm. Khi không có công cụ hỗ trợ, việc lưu trữ dữ liệu được thực hiện trên giấy rất dễ bị mất mát hoặc sai sót, khiến cho hoạt động và doanh thu của doanh nghiệp bị hạn chế. Hơn nữa, không gian nhỏ và nhân viên có hạn cũng tạo ra nhiều bất tiện trong việc phục vụ khách hàng và quản lý sản phẩm.

Với những khó khăn trên, việc xây dựng một hệ thống quản lý để thuận tiện cho việc quản lý, giải quyết các vấn đề của doanh nghiệp là rất cần thiết. Đối với các doanh nghiệp mới mở, nếu không biết cách quản lý sẽ khó để phát triển trong tương lai. Chính vì vậy, nhóm em quyết định xây dựng 1 hệ thống quản lý, cụ thể là của nhà sách Hải An - hiệu sách nổi tiếng tại TP. Hồ Chí Minh. Đây cũng là lý do chọn đề tài của nhóm em.

2. Xác định và phân tích yêu cầu

2.1. Xác định yêu cầu

- Tài khoản: Lưu trữ thông tin của nhân viên hay quản lý hỗ trợ cho việc phân quyền các chức năng khi đăng nhập vào hệ thống
- Nhân viên: Thêm, xóa, sửa, tìm kiếm nhân viên, hỗ trợ lập in các hóa đơn , phiếu nhập
- Sách: Lưu trữ các thông tin sách, và các chức năng như thêm, xóa, sửa, tìm kiếm sách giúp cho việc quản lý dễ dàng hơn

- Nhà cung cấp: sách được những nhà xuất bản cung cấp thông qua những phiếu nhập, mỗi nhà cung cấp có thể cung cấp nhiều loại sách.
- Khách hàng: lưu trữ thông tin khách hàng khi đến mua sách, có các chức năng như thêm, xóa, sửa, tìm kiếm khách hàng
- Hoá đơn: hỗ trợ lưu thông tin các hóa đơn giúp khách hàng có đầy đủ thông tin về các giao dịch mà mình với nhà sách và nhân viên có thể lưu lại ở hệ thống.
- Thẻ loại: hỗ trợ tìm kiếm các loại sách theo yêu cầu để giúp khách hàng có thể dễ dàng lựa chọn những thẻ loại, tên sách mà khách hàng muốn tìm.
- Phiếu nhập: Lưu thông tin và kiểm soát được ngày nhập, tổng tiền của lượng sách khi được nhập vào kho
- Chi tiết hóa đơn: Lưu trữ thông tin của từng mã hóa đơn từ đó có thể dễ dàng xem chi tiết của từng mã hóa đơn chính xác và dễ dàng
- Chi tiết phiếu nhập: Lưu trữ thông tin của từng mã phiếu nhập từ đó có thể dễ dàng xem chi tiết từng mã phiếu nhập chính xác và dễ dàng

2.2. Phân tích yêu cầu

2.2.1. Yêu cầu chức năng

- **Chức năng lưu trữ dữ liệu**
 - Lưu trữ tất cả thông tin trong cơ sở dữ liệu, giúp truy xuất, sao lưu và phục hồi dễ dàng, đồng bộ, nhanh chóng và đáng tin cậy.
- **Chức năng thêm, xóa, sửa, tìm kiếm, thống kê, tính toán**
 - Quản lý tài khoản: Hiệu sách cần quản lý thông tin từng tài khoản và phân quyền cho từng loại tài khoản khi truy cập vào hệ thống. Cần có các chức năng thêm, xóa, sửa, tìm kiếm các thông tin cơ bản của tài khoản và báo cáo thống kê doanh thu bán hàng của từng nhân viên.
 - Quản lý nhân viên: Hiệu sách cần quản lý thông tin từng nhân viên và phân quyền cho từng loại tài khoản của nhân viên theo các chức vụ khi truy cập vào hệ thống. Cần có các chức năng như thêm, xóa, sửa, tìm kiếm các thông tin cơ bản của nhân viên.

- Quản lý sách: Hiệu sách cần quản lý thông tin của các quyển sách. Cần có các chức năng như thêm, xóa, sửa, tìm kiếm các thông tin của sách trong hệ thống.
- Quản lý hóa đơn: Hiệu sách cần phải quản lý hóa đơn cho các khách hàng. Hệ thống cần có chức năng thêm, xóa, sửa, tìm kiếm cho từng giao dịch. Hiệu sách cần thể hiện những thông tin về hóa đơn như mã hóa đơn, ngày của hóa đơn, mã nhân viên, mã khách hàng, tổng tiền.
- Quản lý khách hàng: Hiệu sách cần phải quản lý thông tin của khách hàng. Hệ thống cần có chức năng thêm, xóa, sửa, tìm kiếm các thông tin của khách hàng.
- Quản lý thẻ loại: Hiệu sách quản lý các thông tin về thẻ loại như mã thẻ loại, tên thẻ loại, ghi chú. Và các chức năng như thêm, xóa, sửa, tìm kiếm.
- Quản lý nhà cung cấp: Hiệu sách quản lý các thông tin về nhà cung cấp như mã nhà cung cấp, tên nhà cung cấp, địa chỉ, số điện thoại. Và các chức năng như thêm, xóa, sửa, tìm kiếm.
- Quản lý phiếu nhập: Hiệu sách cần phải quản lý phiếu nhập khi thực hiện nhập sách. Hệ thống cần có chức năng thêm, xóa, sửa, tìm kiếm cho từng giao dịch. Hiệu sách cần thể hiện những thông tin về phiếu nhập như mã phiếu nhập, ngày nhập, tổng tiền, mã nhân viên.
- Quản lý chi tiết hóa đơn: Hiệu sách lưu thông tin của từng mã hóa đơn. Hệ thống có các chức năng như thêm, xóa, sửa, tìm kiếm, tính tiền cho mỗi mã hóa đơn.
- Quản lý chi tiết phiếu nhập: Hiệu sách lưu thông tin của từng mã phiếu nhập. Hệ thống có các chức năng như thêm, xóa, sửa, tìm kiếm, tính tiền cho mỗi mã phiếu nhập.

• **Chức năng phân quyền**

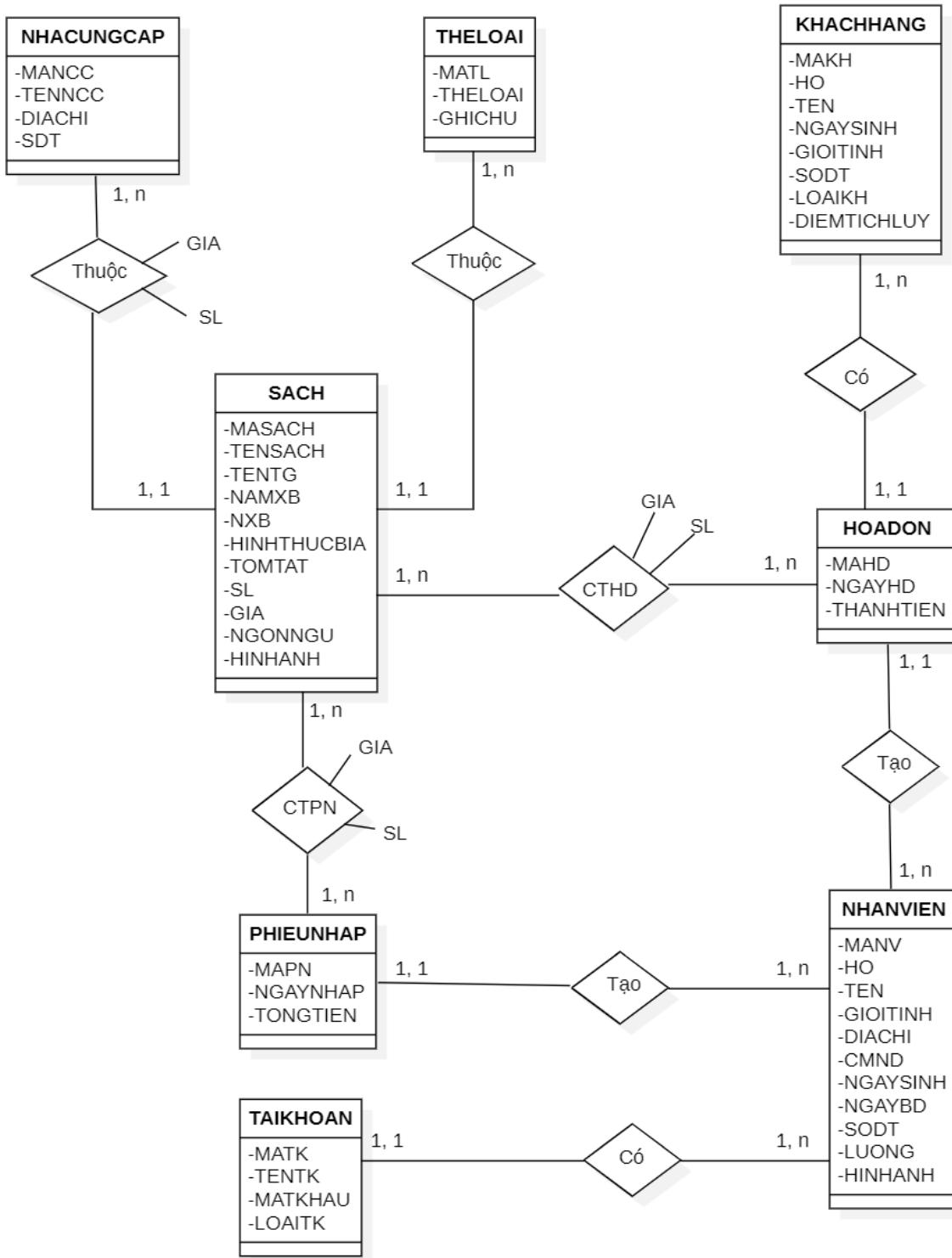
- Bộ phận quản lý hệ thống phân quyền cho từng nhân viên ở các bộ phận khác nhau.
- ADMIN và quản lý được toàn quyền xử lý và giám sát hệ thống.

2.2.2. Yêu cầu phi chức năng

- **Độ bảo mật:** Hệ thống quản lý hiệu sách cần đảm bảo tính bảo mật đối với thông tin liên quan đến khách hàng, nhân viên và tài khoản của hiệu sách.
- **Hiệu suất:** Hệ thống quản lý hiệu sách cần đáp ứng được yêu cầu về hiệu suất, đảm bảo thời gian xử lý nhanh và độ trễ thấp.
- **Thân thiện với người dùng:** Hệ thống quản lý hiệu sách cần có giao diện thân thiện, giúp người dùng dễ dàng sử dụng hệ thống.
- **Khả năng tương thích:** Hệ thống cần đảm bảo khả năng tương thích với môi trường ứng dụng khác nhau.
- **Dễ dàng bảo trì, cập nhật:** Hệ thống cần được thiết kế để có khả năng dễ dàng bảo trì, nâng cấp và cập nhật khi có yêu cầu mới.
- **Độ ổn định, chính xác:** Hệ thống cần đảm bảo độ ổn định cao, đảm bảo hoạt động liên tục mà không gặp sự cố, thể hiện chính xác các số liệu báo cáo thống kê tự động.

3. Thiết kế mô hình quan hệ

3.1. Mô hình ERD (Thực thể mối kết hợp)



3.2. Mô hình quan hệ

NHACUNGCAP (MaNCC, TenNCC, DiaChi, SDT)

THELOAI (MaTL, TheLoai, GhiChu)

SACH (MaSach, TenSach, MaTL, TenTG, NamXB, NXB, HinThucBia, TomTat, SL, Gia, MaNCC, NgonNgu, HinAnh)

KHACHHANG (MaKH, Ho, Ten, NgaySinh, GioiTinh, SoDT, LoaiKH, DiemTichLuy)

NHANVIEN (MaNV, Ho, Ten, GioiTinh, DiaChi, CMND, NgaySinh, NgayBD, SoDT, Luong, HinAnh)

TAIKHOAN (MaTK, TenTK, MatKhau, LoaiTK, MaNV)

HOADON (MaHD, NgayHD, MaNV, MaKH, ThanhTien)

CTHD (MaHD, MaSach, SL)

PHIEUNHAP (MaPN, NgayNhap, TongTien, MaNV)

CTPN (MaSach, MaPN, SL, Gia)

3.3. Bảng thuyết minh quan hệ và thuộc tính

BẢNG THUỘC TÍNH

- Quan hệ NHACUNGCAP

NHACUNGCAP				
Thuộc tính	Kiểu dữ liệu	Ràng buộc	Giá trị mặc định	Ý nghĩa
<u>MaNCC</u>	Number	Khóa chính, not null	Tự động	Mã nhà cung cấp
TenNCC	Varchar2(100)	Not null		Tên nhà cung cấp
DiaChi	Varchar2(100)	Not null		Địa chỉ nhà cung cấp

SDT	Varchar2(12)			Số điện thoại nhà cung cấp
-----	--------------	--	--	-------------------------------

- Quan hệ THELOAI

THELOAI				
Thuộc tính	Kiểu dữ liệu	Ràng buộc	Giá trị mặc định	Ý nghĩa
<u>MaTL</u>	Number	Khóa chính, not null	Tự động	Mã thể loại
TheLoai	Varchar2(30)	Not null		Tên thể loại
GhiChu	Varchar2(1000)			Ghi chú

- Quan hệ SACH

SACH				
Thuộc tính	Kiểu dữ liệu	Ràng buộc	Giá trị mặc định	Ý nghĩa
<u>MaSach</u>	Number	Khóa chính, not null	Tự động	Mã sách
TenSach	Varchar2(80)	Not null		Tên sách
MaTL	Number	Khóa ngoại tham chiếu đến khóa chính MaTL của bảng THELOAI, not null		Mã thể loại
TenTG	Varchar2(150)			Tên tác giả
NamXB	Date			Năm xuất bản
NXB	Varchar2(60)			Nhà xuất bản
HinhThucBia	Varchar2(15)			Hình thức bìa

TomTat	Varchar2 (1000)			Tóm tắt
SL	Number			Số lượng tồn
Gia	Number			Giá bán sách
MaNCC	Number	Khóa ngoại tham chiếu đến khóa chính MaNCC của bảng NHACUNGCAP, not null		Mã nhà cung cấp
NgonNgu	Varchar2(35)			Ngôn ngữ sách
HinhAnh	Varchar2(500)			Hình bìa sách

- Quan hệ KHACHHANG

KHACHHANG				
Thuộc tính	Kiểu dữ liệu	Ràng buộc	Giá trị mặc định	Ý nghĩa
<u>MaKH</u>	Number	Khóa chính, not null	Tự động	Mã khách hàng
Ho	Varchar2(20)	Not null		Họ khách hàng
Ten	Varchar2(20)	Not null		Tên khách hàng
NgaySinh	Date	Not null		Ngày sinh khách hàng
GioiTinh	Varchar2(10)	Not null		Giới tính khách hàng
SoDT	Varchar2(10)			Số điện thoại
LoaiKH	Varchar2(20)			Loại khách hàng
DiemTichLuy	Number		0	Điểm tích lũy

- Quan hệ NHANVIEN

NHANVIEN				
Thuộc tính	Kiểu dữ liệu	Ràng buộc	Giá trị mặc định	Ý nghĩa
<u>MaNV</u>	Number	Khóa chính, not null	Tự động	Mã nhân viên
Ho	Varchar2(20)	Not null		Họ nhân viên
Ten	Varchar2(20)	Not null		Tên nhân viên
GioiTinh	Varchar2(15)			Giới tính nhân viên
DiaChi	Varchar2(100)	Not null		Địa chỉ nhà
CMND	Varchar2(15)	Not null		Chứng minh nhân dân
NgaySinh	Date	Not null		Ngày sinh nhân viên
NgayBD	Date	Not null		Ngày vào làm
SoDT	Varchar2(10)	Not null		Số điện thoại
Luong	Number	Not null		Lương
HinhAnh	Varchar2(500)			Hình ảnh

- Quan hệ TAIKHOAN

TAIKHOAN				
Thuộc tính	Kiểu dữ liệu	Ràng buộc	Giá trị mặc định	Ý nghĩa
<u>MaTK</u>	Number	Khóa chính, not null	Tự động	Mã tài khoản
TenTK	Varchar2(30)			Tên tài khoản

MatKhau	Varchar2(30)			Mật khẩu tài khoản
LoaiTK	Varchar2(30)			Loại tài khoản
MaNV	Number	Khóa ngoại tham chiếu đến khóa chính MaNV của bảng NHANVIEN, not null		Mã nhân viên

- Quan hệ HOADON

HOADON				
Thuộc tính	Kiểu dữ liệu	Ràng buộc	Giá trị mặc định	Ý nghĩa
<u>MaHD</u>	Number	Khóa chính, not null	Tự động	Mã hóa đơn
NgayHD	Date	Not null		Ngày hóa đơn
MaNV	Number	Khóa ngoại tham chiếu đến khóa chính MaNV của bảng NHANVIEN, not null		Mã nhân viên
MaKH	Number	Khóa ngoại tham chiếu đến khóa chính MaKH của bảng KHACHHANG		Mã khách hàng

ThanhTien	Number		0	Thành tiền của hóa đơn
-----------	--------	--	---	---------------------------

- Quan hệ PHIEUNHAP

PHIEUNHAP				
Thuộc tính	Kiểu dữ liệu	Ràng buộc	Giá trị mặc định	Ý nghĩa
<u>MaPN</u>	Number	Khóa chính, not null	Tự động	Mã phiếu nhập
NgayNhap	Date	Not null		Ngày nhập hàng
TongTien	Number		0	Tổng tiền của phiếu nhập
MaNV	Number	Khóa ngoại tham chiếu đến khóa chính MaNV của bảng NHANVIEN, not null		Mã nhân viên

- Quan hệ CTPN

CTPN				
Thuộc tính	Kiểu dữ liệu	Ràng buộc	Giá trị mặc định	Ý nghĩa
<u>MaSach</u>	Number	Khóa chính, khóa ngoại tham chiếu đến khóa chính MaSach của bảng SACH, not null		Mã sách

<u>MaPN</u>	Number	Khóa chính, khóa ngoại tham chiếu đến khóa chính MaPN của bảng PHIEUNHAP, not null	Tự động	Mã phiếu nhập
SL	Number	Not null		Số lượng
Gia	Number	Not null		Giá nhập sách

- Quan hệ CTHD

CTHD				
Thuộc tính	Kiểu dữ liệu	Ràng buộc	Giá trị mặc định	Ý nghĩa
<u>MaHD</u>	Number	Khóa chính, khóa ngoại tham chiếu đến khóa chính MaHD của bảng HOADON, not null	Tự động	Mã hóa đơn
<u>MaSach</u>	Number	Khóa chính, khóa ngoại tham chiếu đến khóa chính MaSach của bảng SACH, not null		Mã sách
SL	Number	Not null		Số lượng
Gia	Number	Not null		Giá bán sách

3.4. Mô tả ràng buộc toàn vẹn

3.4.1. Ràng buộc toàn vẹn có bối cảnh trên 1 quan hệ

- **Ràng buộc toàn vẹn khóa chính, unique**

- **Ràng buộc 1:** Mã nhà cung cấp của quan hệ Nhà cung cấp là duy nhất
 - Bối cảnh: NHACUNGCAP
 - Nội dung: $\forall ncc1, ncc2 \in NHACUNGCAP: ncc1.MaNCC \neq ncc2.MaNCC$
 - Bảng tầm ảnh hưởng:

RB1	Thêm	Xóa	Sửa
NHACUNGCAP	+	-	+(MaNCC)

- **Ràng buộc 2:** Mã thể loại sách của quan hệ Thể loại là duy nhất
 - Bối cảnh: THELOAI
 - Nội dung: $\forall tl1, tl2 \in THELOAI: tl1.MaTL \neq tl2.MaTL$
 - Bảng tầm ảnh hưởng:

RB2	Thêm	Xóa	Sửa
THELOAI	+	-	+(MaTL)

- **Ràng buộc 3:** Mã sách của quan hệ Sách là duy nhất
 - Bối cảnh: SACH
 - Nội dung: $\forall s1, s2 \in SACH: s1.MaSach \neq s2.MaSach$
 - Bảng tầm ảnh hưởng:

RB3	Thêm	Xóa	Sửa
SACH	+	-	+(MaSach)

- **Ràng buộc 4:** Mã khách hàng của quan hệ Khách hàng là duy nhất
 - Bối cảnh: KHACHHANG
 - Nội dung: $\forall kh1, kh2 \in KHACHHANG: kh1.MaKH \neq kh2.MaKH$
 - Bảng tầm ảnh hưởng:

RB4	Thêm	Xóa	Sửa
KHACHHANG	+	-	+(MaKH)
RB5	Thêm	Xóa	Sửa
NHANVIEN	+	-	+(MaNV)

- **Ràng buộc 5:** Mã nhân viên của quan hệ Nhân viên là duy nhất
 - Bối cảnh: NHANVIEN
 - Nội dung: $\forall nv1, nv2 \in NHANVIEN: nv1.MaNV \neq nv2.MaNV$
 - Bảng tầm ảnh hưởng:

RB5	Thêm	Xóa	Sửa
NHANVIEN	+	-	+(MaNV)
RB6	Thêm	Xóa	Sửa
TAIKHOAN	+	-	+(MaTK)

- **Ràng buộc 6:** Mã tài khoản của quan hệ Tài khoản là duy nhất
 - Bối cảnh: TAIKHOAN
 - Nội dung: $\forall tk1, tk2 \in TAIKHOAN: tk1.MaTK \neq tk2.MaTK$
 - Bảng tầm ảnh hưởng:

RB6	Thêm	Xóa	Sửa
TAIKHOAN	+	-	+(MaTK)
RB7	Thêm	Xóa	Sửa
HOADON	+	-	+(MaHD)

- **Ràng buộc 7:** Mã hóa đơn của quan hệ Hóa đơn là duy nhất
 - Bối cảnh: HOADON
 - Nội dung: $\forall hd1, hd2 \in HOADON: hd1.MaHD \neq hd2.MaHD$
 - Bảng tầm ảnh hưởng:

RB7	Thêm	Xóa	Sửa
HOADON	+	-	+(MaHD)
RB8	Thêm	Xóa	Sửa
PHIEUNHAP	+	-	+(MaPN)

- **Ràng buộc 8:** Mã phiếu nhập hàng của quan hệ Phiếu nhập là duy nhất
 - Bối cảnh: PHIEUNHAP
 - Nội dung: $\forall pn1, pn2 \in PHIEUNHAP: pn1.MaPN \neq pn2.MaPN$
 - Bảng tầm ảnh hưởng:

RB8	Thêm	Xóa	Sửa
PHIEUNHAP	+	-	+(MaPN)

- **Ràng buộc 9:** Với mọi tài khoản, tên tài khoản là duy nhất
 - Bối cảnh: TAIKHOAN
 - Nội dung: $\forall tk1, tk2 \in TAIKHOAN: tk1.TenTK \neq tk2.TenTK$
 - Bảng tầm ảnh hưởng:

RB9	Thêm	Xóa	Sửa
TAIKHOAN	+	-	+(TenTK)

- **Ràng buộc liên thuộc tính**

- **Ràng buộc 10:** Với mọi nhân viên, ngày bắt đầu làm (NgayBD) luôn lớn hơn ngày sinh (NgaySinh) của nhân viên đó.
 - Bối cảnh: NHANVIEN
 - Nội dung: $\forall nv \in NHANVIEN: nv.NgayBD > nv.NgaySinh$
 - Bảng tầm ảnh hưởng:

RB10	Thêm	Xóa	Sửa
NHANVIEN	+	-	+(MaBD, NgaySinh)

- **Ràng buộc 11:** Với mọi cuốn sách, kiểm tra Giá và Số lượng luôn lớn hơn hoặc bằng 0.
 - Bối cảnh: SACH
 - Nội dung: $\forall s \in SACH: s.Gia \geq 0 \wedge s.SL \geq 0$
 - Bảng tầm ảnh hưởng:

RB11	Thêm	Xóa	Sửa
SACH	+	-	+(Gia, SL)

3.4.2. Ràng buộc toàn vẹn có bối cảnh trên nhiều quan hệ

- **Ràng buộc toàn vẹn khóa ngoại**

- **Ràng buộc 12:** Mỗi cuốn sách phải thuộc một thể loại sách.

- Bối cảnh: SACH, THELOAI
- Nội dung: $\forall s \in SACH, \exists tl \in THELOAI: s.MaTL = tl.MaTL$
- Bảng tầm ảnh hưởng:

RB12	Thêm	Xóa	Sửa
SACH	+	-	+(MaTL)
THELOAI	-	+	-

- **Ràng buộc 13:** Mỗi cuốn sách phải thuộc một nhà cung cấp.

- Bối cảnh: SACH, NHACUNGCAP
- Nội dung: $\forall s \in SACH, \exists ncc \in NHACUNGCAP: s.MaNCC = ncc.MaNCC$
- Bảng tầm ảnh hưởng:

RB13	Thêm	Xóa	Sửa
SACH	+	-	+(MaNCC)
NHACUNGCAP	-	+	-

- **Ràng buộc 14:** Mỗi tài khoản phải thuộc một nhân viên.

- Bối cảnh: TAIKHOAN, NHANVIEN
- Nội dung: $\forall tk \in TAIKHOAN, \exists nv \in NHANVIEN: tk.MaTK = nv.MaNV$
- Bảng tầm ảnh hưởng:

RB14	Thêm	Xóa	Sửa
TAIKHOAN	+	-	+(MaNV)
NHANVIEN	-	+	-

- **Ràng buộc 15:** Mỗi hóa đơn phải có Mã nhân viên tạo hóa đơn đó.

- Bối cảnh: HOADON, NHANVIEN
- Nội dung: $\forall hd \in HOADON, \exists nv \in NHANVIEN: hd.MaNV = nv.MaNV$
- Bảng tầm ảnh hưởng:

RB15	Thêm	Xóa	Sửa
HOADON	+	-	+(MaNV, MaKH)
NHANVIEN	-	+	-

- **Ràng buộc 16:** Mỗi phiếu nhập hàng phải lưu thông tin nhân viên nhập.

- Bối cảnh: PHIEUNHAP, NHANVIEN
- Nội dung: $\forall pn \in PHIEUNHAP, \exists nv \in NHANVIEN: pn.MaNV = nv.MaNV$
- Bảng tầm ảnh hưởng:

RB16	Thêm	Xóa	Sửa
PHIEUNHAP	+	-	+(MaNV)
NHANVIEN	-	+	-

- **Ràng buộc 17:** Mỗi chi tiết phiếu nhập hàng phải lưu Mã sách và Mã phiếu nhập hàng.

- Bối cảnh: PHIEUNHAP, SACH, CTPN
- Nội dung: $\forall ctpn \in CTPN, \exists s \in SACH, \exists pn \in PHIEUNHAP: ctpn.MaSach = s.MaSach \wedge ctpn.MaPN = pn.MaPN$
- Bảng tầm ảnh hưởng:

RB17	Thêm	Xóa	Sửa
PHIEUNHAP	+	-	+(MaSach, MaPN)
SACH	-	+	-
CTPN	-	+	-

- **Ràng buộc 18:** Mỗi chi tiết hóa đơn phải lưu Mã hóa đơn và Mã sách.

- Bối cảnh: CTHD, SACH, HOADON

- Nội dung: $\forall cthd \in CTHD, \exists s \in SACH, \exists hd \in HOADON:$
 $cthd.MaSach = s.MaSach \wedge cthd.MaHD = hd.MaHD$
- Bảng tầm ảnh hưởng:

RB18	Thêm	Xóa	Sửa
CTHD	+	-	+(MaSach, MaHD)
SACH	-	+	-
HOADON	-	+	-

- **Ràng buộc liên thuộc tính liên quan hệ**

- **Ràng buộc 19:** Ngày bán hàng (NgayHD) của một nhân viên phải lớn hơn hoặc bằng ngày nhân viên đó vào làm (NgayBD).

- Bối cảnh: NHANVIEN, HOADON
- Nội dung: $\forall nv \in NHANVIEN, \forall hd \in HOADON: nv.MaNV = hd.MaNV \wedge nv.NgayBD \leq hd.NgayHD$
- Bảng tầm ảnh hưởng:

RB19	Thêm	Xóa	Sửa
NHANVIEN	-	-	+(NgayBD)
HOADON	+	-	+(NgayHD, MaNV)

- **Ràng buộc 20:** Ngày nhập hàng (NgayNhap) vào kho hàng của một nhân viên phải lớn hơn hoặc bằng ngày nhân viên đó vào làm (NgayBD).

- Bối cảnh: NHANVIEN, PHIEUNHAP
- Nội dung: $\forall nv \in NHANVIEN, \forall pn \in PHIEUNHAP: nv.MaNV = pn.MaPN \wedge nv.NgayBD \leq pn.NgayNhap$
- Bảng tầm ảnh hưởng:

RB20	Thêm	Xóa	Sửa
NHANVIEN	-	-	+(NgayBD)
HOADON	+	-	+(NgayNhap, MaNV)

- **Ràng buộc 21:** Ngày mua hàng (NgayHD) của một khách hàng phải lớn hơn hoặc bằng ngày sinh của khách hàng đó (NgaySinh).

- Bối cảnh: KHACHHANG, HOADON
- Nội dung: $\forall kh \in KHACHHANG, \forall hd \in HOADON: kh.MaKH = hd.MaKH \wedge kh.NgaySinh \leq hd.NgayHD$
- Bảng tầm ảnh hưởng:

RB21	Thêm	Xóa	Sửa
KHACHHANG	-	-	+(NgaySinh)
HOADON	+	-	+(NgayHD, MaKH)

- **Ràng buộc 22:** Giá (Gia) của 1 chi tiết phiếu nhập phải nhỏ hơn giá bán của sách

- Bối cảnh: CTPN, SACH
 - Nội dung:
- $$\forall ct \in CTPN, \exists s \in SACH: ct.MaSach = s.MaSach \wedge ct.Gia < s.Gia$$
- Bảng tầm ảnh hưởng:

RB22	Thêm	Xóa	Sửa
CTPN	+	-	+(MaSach, Gia)
SACH	-	-	+(Gia)

• Ràng buộc do thuộc tính tổng hợp

- **Ràng buộc 23:** Tổng tiền (TongTien) của 1 Phiếu nhập là tổng giá (SL*Gia) của các chi tiết phiếu nhập thuộc phiếu nhập đó.

- Bối cảnh: PhieuNhap, CTPN
- Nội dung:

$$\forall pn \in PhieuNhap: pn.TongTien = \sum_{(ct \in CTPN: ct.MaPN = pn.MaPN)} (ct.SL * ct.Gia)$$

- Bảng tầm ảnh hưởng:

RB23	Thêm	Xóa	Sửa
PhieuNhap	- (1)	-	+(TongTien)
CTPN	+	+	+(MaPN, SL, Gia)

(1) Đối với thao tác thêm mới 1 PhieuNhap chưa có CTPN, kiểm tra

ThanhTien = 0

- **Ràng buộc 24:** Điểm tích lũy (DiemTichLuy) của 1 khách hàng là tổng giá trị (SL * GIA) của các CTHD, với GIA là thuộc tính trong bảng SACH ứng với các chi tiết hóa đơn của khách hàng đó

 - Bối cảnh: KHACHHANG, HOADON, CTHD, SACH

 - Nội dung:

$$\forall kh \in KHACHHANG, kh.DiemTichLuy = \sum_{(hd \in HOADON, ct \in CTHD, s \in SACH: kh.MaKH = hd.MaKH \wedge hd.MaDH = ct.MaHD \wedge ct.Sach = s.MaSach)} (ct.SL * s.Gia)$$

$$(ct.SL * s.Gia)$$

 - Bảng tầm ảnh hưởng:

RB24	Thêm	Xóa	Sửa
KHACHHANG	- (1)	-	+(DiemTichLuy)
HOADON	-	-	+(MaKH)
CTHD	+	+	+(SL)
SACH	-	-(*)	+(Gia)

➤ (1) Đối với thao tác thêm mới 1 khách hàng, kiểm tra DiemTichLuy = 0

PHẦN II. XÂY DỰNG VÀ QUẢN LÝ GIAO TÁC

1. Qui định hệ thống giao tác

- Tự động tăng MaNCC lên 1 đơn vị so với MaNCC lớn nhất trong bảng NHACUNGCAP khi thực hiện thêm mới một nhà cung cấp.
- Tự động tăng MaTL lên 1 đơn vị so với MaTL lớn nhất trong bảng THELOAI khi thực hiện thêm mới một thể loại.
- Tự động tăng MaSach lên 1 đơn vị so với MaSach lớn nhất trong bảng SACH khi thực hiện thêm mới một sách.
- Tự động tăng MaKH lên 1 đơn vị so với MaKH lớn nhất trong bảng KHACHHANG khi thực hiện thêm mới một khách hàng.
- Tự động tăng MaNV lên 1 đơn vị so với MaNV lớn nhất trong bảng NHANVIEN khi thực hiện thêm mới một nhân viên.
- Tự động tăng MaTK lên 1 đơn vị so với MaTK lớn nhất trong bảng TAIKHOAN khi thực hiện thêm mới một tài khoản.
- Tự động tăng MaHD lên 1 đơn vị so với MaHD lớn nhất trong bảng HOADON khi thực hiện thêm mới một hóa đơn.
- Tự động tăng MaPN lên 1 đơn vị so với MaPN lớn nhất trong bảng PHIEUNHAP khi thực hiện thêm mới một phiếu nhập.
- Tự động cập nhật lại THANHTIEN của một hóa đơn và DIEMTICHLUY và LOAIKH của khách hàng tương ứng sau khi:
 - Insert một CTHD mới
 - Update giá sách của SACH thuộc CTHD của HOADON đó
 - Update CTHD thuộc HOADON đó
 - Delete một CTHD thuộc HOADON đó
- Tự động cập nhật lại TongTien của một PhieuNhap sau khi update (MaPN, Gia) hoặc delete một CTPN thuộc PhieuNhap đó.
- Tự động phân loại khách hàng và cập nhật lại điểm tích lũy của khách hàng khi update (MAKH) trong hóa đơn của khách hàng:
 - Điểm tích lũy < 500000: loại khách hàng “Thường”

- Điểm tích lũy ≥ 500000 : loại khách hàng “Thân Thiết”
 - Điểm tích lũy ≥ 2000000 : loại khách hàng “VIP”
- Tự động cập nhật lại số lượng sách trong bảng SACH khi update hoặc delete CTHD tương ứng.
- Tự động cập nhật lại số lượng sách trong bảng SACH khi update hoặc delete CTPN tương ứng

2. Xây dựng và mô tả Trigger trong đồ án môn học

STT	TRIGGER	EVENT	TABLE
2.1	TRG19_NGBD_NGHD_NHANVIEN_up <u>Mã lệnh PL/SQL</u> <pre> create or replace trigger TRG19_NGBD_NGHD_NHANVIEN_up before update of NgayBD on NHANVIEN for each row declare cursor curNGHD is select NGAYHD from HOADON where MANV = :new.MANV; v_NGAYHD date; begin open curNGHD; LOOP FETCH curNGHD into v_NGAYHD; EXIT WHEN curNGHD%NOTFOUND; if (v_NGAYHD < :new.NGAYBD) then raise_application_error(-20001, 'Ngay bat dau lam khong hop le'); end if; END LOOP; end; </pre>	UPDATE	NHANVIEN

	Mô tả mã lệnh:			
	<ul style="list-style-type: none"> Tạo một biến cursor tên là "curNGHD", lấy các bản ghi trong bảng "HOADON" có trường MANV bằng với giá trị của MANV trong bản ghi mới. Tạo một biến date tên là "v_NGAYHD". Mở cursor "curNGHD" để lấy các bản ghi. Sử dụng vòng lặp LOOP để duyệt qua từng bản ghi lấy được trong cursor "curNGHD". Trong mỗi lần duyệt qua, tìm giá trị của trường "NGAYHD" trong bảng "HOADON" và lưu trữ vào biến "v_NGAYHD". Nếu "v_NGAYHD" nhỏ hơn "NGAYBD" (trường "NGAYBD" là trường trong bảng "NHANVIEN"), thì sẽ gây ra lỗi thông qua hàm "raise_application_error". Lỗi sẽ có mã -20001 và thông báo "Ngay bat dau lam khong hop le". 			
2.2	TRG19_NGBD_NGHD_HOADON_in_up	INSERT, UPDATE	HOADON	
	Mã lệnh PL/SQL			
	<pre> create or replace trigger TRG19_NGBD_NGHD_HOADON_in_up before insert or update of MaNV, NgayHD on HOADON for each row declare cursor curNGBD is select NGAYBD from NHANVIEN where MANV = :new.MANV; v_NGAYBD date; begin open curNGBD; LOOP FETCH curNGBD into v_NGAYBD; EXIT WHEN curNGBD%NOTFOUND; if (:new.NGAYHD < v_NGAYBD) then raise_application_error(-20002, 'NgayHD khong hop le'); end if; end loop; end; / </pre>			

	<pre> end if; END LOOP; End; </pre>		
Mô tả mã lệnh:			
	<ul style="list-style-type: none"> • Tạo một biến cursor tên "curNGBD", lấy giá trị của trường "NGAYBD" trong bảng "NHANVIEN" có trường "MANV" bằng với giá trị của "MANV" trong bản ghi mới. • Tạo một biến date tên "v_NGAYBD". • Mở cursor "curNGBD" để lấy các bản ghi. • Sử dụng vòng lặp LOOP để duyệt qua từng bản ghi lấy được trong cursor "curNGBD". • Trong mỗi lần duyệt qua, tìm giá trị của trường "NGAYBD" trong bảng "NHANVIEN" và lưu trữ vào biến "v_NGAYBD". • Nếu giá trị của trường "NGAYHD" trong bảng "HOADON" nhỏ hơn giá trị của "NGAYBD" trong bảng "NHANVIEN" tương ứng (với cùng một nhân viên), trigger sẽ gây ra lỗi thông qua hàm "raise_application_error". Lỗi sẽ có mã - 20002 và thông báo "NgayHD khong hop le". 		
2.3	<p style="text-align: center;">TRG20_NGBD_NGNHAP_NHANVIEN_up</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="padding: 5px; width: 50%;">UPDATE</td> <td style="padding: 5px; width: 50%;">NHANVIEN</td> </tr> </table> <p>Mã lệnh PL/SQL</p> <pre> create or replace trigger TRG20_NGBD_NGNHAP_NHANVIEN_up before update of NgayBD on NHANVIEN for each row declare cursor curNGNHAP is select NGAYNHAP from PHIEUNHAP where MANV = :new.MANV; v_NGAYNHAP date; begin open curNGNHAP; </pre>	UPDATE	NHANVIEN
UPDATE	NHANVIEN		

	<pre> LOOP FETCH curNGNHAP into v_NGAYNHAP; EXIT WHEN curNGNHAP%NOTFOUND; if (v_NGAYNHAP < :new.NGAYBD) then raise_application_error(-20003, 'Ngay bat dau lam khong hop le'); end if; END LOOP; end; </pre>			
	<p>Mô tả mã lệnh:</p> <ul style="list-style-type: none"> Tạo một biến cursor tên "curNGNHAP", lấy các bản ghi trong bảng "PHIEUNHAP" có trường "MANV" bằng với giá trị của "MANV" trong bản ghi mới. Tạo một biến date tên "v_NGAYNHAP". Mở cursor "curNGNHAP" để lấy các bản ghi. Sử dụng vòng lặp LOOP để duyệt qua từng bản ghi lấy được trong cursor "curNGNHAP". Trong mỗi lần duyệt qua, tìm giá trị của trường "NGAYNHAP" trong bảng "PHIEUNHAP" và lưu trữ vào biến "v_NGAYNHAP". Nếu "v_NGAYNHAP" nhỏ hơn trường "NGAYBD" trong bảng "NHANVIEN" tương ứng, trigger sẽ gây ra lỗi thông qua hàm "raise_application_error". Lỗi sẽ có mã -20003 và thông báo "Ngay bat dau lam khong hop le". 			
2.4	<table border="1"> <tr> <td style="text-align: center;">TRG20_NGBD_NGNHAP_PHIEUNHAP_in_up</td> <td style="text-align: center;">INSERT, UPDATE</td> <td style="text-align: center;">PHIEUNHAP</td> </tr> </table> <p>Mã lệnh PL/SQL</p> <pre> create or replace trigger TRG20_NGBD_NGNHAP_PHIEUNHAP_in_up before insert or update of MaNV, NgayNhap on PHIEUNHAP for each row declare </pre>	TRG20_NGBD_NGNHAP_PHIEUNHAP_in_up	INSERT, UPDATE	PHIEUNHAP
TRG20_NGBD_NGNHAP_PHIEUNHAP_in_up	INSERT, UPDATE	PHIEUNHAP		

```

cursor curNGBD is select NGAYBD from NHANVIEN where MANV =
:new.MANV;
v_NGAYBD date;
begin
open curNGBD;
LOOP
  FETCH curNGBD into v_NGAYBD;
  EXIT WHEN curNGBD%NOTFOUND;
  if (:new.NGAYNHAP < v_NGAYBD) then
    raise_application_error(-20004, 'NgayNhập không hợp lệ');
  end if;
END LOOP;
end;

```

Mô tả mã lệnh:

Mã lệnh này có chức năng kiểm tra trước khi thêm mới hoặc cập nhật bản ghi trong bảng "PHIEUNHAP". Trigger sẽ thực hiện truy vấn để lấy NGAYBD của NHANVIEN tương ứng với MANV và kiểm tra trường "NGAYNHAP" của bản ghi mới hoặc sửa đổi có hợp lệ hay không. Nếu "NGAYNHAP" nhỏ hơn "NGAYBD", sẽ gây ra lỗi và thông báo lỗi cho người dùng.

Với mỗi dòng được thêm mới hoặc cập nhật trong bảng "PHIEUNHAP", trigger sẽ thực hiện các bước trong khối code sau:

- Một con trỏ (cursor) được khai báo để thực hiện truy vấn lấy NGAYBD trong bảng NHANVIEN với điều kiện MANV bằng với giá trị của trường "MANV" trong bản ghi mới hoặc sửa đổi "MANV".
- Khai báo biến "v_NGAYBD" để lưu trữ kết quả của truy vấn lấy NGAYBD.
- Mở con trỏ curNGBD và sử dụng vòng lặp để duyệt qua kết quả trả về từ truy vấn lấy NGAYBD. Nếu không tìm thấy bản ghi nào thì thoát khỏi vòng lặp.

	<ul style="list-style-type: none"> Trong vòng lặp, trigger sẽ kiểm tra trường "NGAYNHAP" của bản ghi mới hoặc sửa đổi với giá trị của "NGAYBD". Nếu "NGAYNHAP" nhỏ hơn "NGAYBD", sẽ gây ra lỗi thông qua hàm "raise_application_error". Lỗi sẽ có mã -20004 và thông báo "NgayNhap khong hop le". 		
2.5	<p style="text-align: center;">TRG21_NGSINHKH_NGHD_KHACHHANG_up</p> <p>Mã lệnh PL/SQL</p> <pre>create or replace trigger TRG21_NGSINHKH_NGHD_KHACHHANG_up before update of NgaySinh on KHACHHANG for each row declare cursor curNGHD is select NGAYHD from HOADON where MAKH = :new.MAKH; v_NGAYHD date; begin open curNGHD; LOOP FETCH curNGHD into v_NGAYHD; EXIT WHEN curNGHD%NOTFOUND; if (v_NGAYHD < :new.NGAYSINH) then raise_application_error(-20005, 'Ngay Sinh khong hop le'); end if; END LOOP; end;</pre> <p>Mô tả mã lệnh:</p> <p>Mã lệnh này có chức năng kiểm tra trước khi cập nhật bản ghi trong bảng "KHACHHANG". Trigger sẽ thực hiện truy vấn để lấy NGAYHD của HOADON tương ứng với MAKH và kiểm tra trường "NGAYSINH" của bản ghi mới hoặc sửa đổi có hợp</p>	UPDATE	KHACHHANG

	<p>lệ hay không. Nếu "NGAYSINH" lớn hơn "NGAYHD", sẽ gây ra lỗi và thông báo lỗi cho người dùng.</p> <p>Với mỗi dòng được cập nhật trong bảng "KHACHHANG", trigger sẽ thực hiện các bước trong khối code sau:</p> <ul style="list-style-type: none"> Một con trỏ (cursor) được khai báo để thực hiện truy vấn lấy NGAYHD trong bảng HOADON với điều kiện MAKH bằng với giá trị của trường "MAKH" trong bản ghi mới hoặc sửa đổi "MAKH". Khai báo biến "v_NGAYHD" để lưu trữ kết quả của truy vấn lấy NGAYHD. Mở con trỏ curNGHD và sử dụng vòng lặp để duyệt qua kết quả trả về từ truy vấn lấy NGAYHD. Nếu không tìm thấy bản ghi nào thì thoát khỏi vòng lặp. Trong vòng lặp, trigger sẽ kiểm tra trường "NGAYSINH" trong bản ghi mới hoặc sửa đổi với giá trị của "NGAYHD". Nếu "NGAYSINH" lớn hơn "NGAYHD", sẽ gây ra lỗi thông qua hàm "raise_application_error". Lỗi sẽ có mã -20005 và thông báo "Ngay Sinh khong hop le". 		
2.6	<p>TRG21_NGSINHKH_NGHD_HOADON_in_up</p> <table border="1"> <tr> <td style="text-align: center;">INSERT, UPDATE</td><td style="text-align: center;">HOADON</td></tr> </table> <p>Mã lệnh PL/SQL</p> <pre>create or replace trigger TRG21_NGSINHKH_NGHD_HOADON_in_up before insert or update of NgayHD on HOADON for each row declare cursor curNGSINH is select NGAYSINH from KHACHHANG where MAKH = :new.MAKH; v_NGAYSINH date; begin open curNGSINH; LOOP FETCH curNGSINH into v_NGAYSINH; EXIT WHEN curNGSINH%NOTFOUND; END LOOP; if v_NGAYSINH > :new.NgayHD then raise_application_error(-20005, 'Ngay Sinh khong hop le'); end if; end;</pre>	INSERT, UPDATE	HOADON
INSERT, UPDATE	HOADON		

	<pre> if (:new.NGAYHD < v_NGAYSINH) then raise_application_error(-20006, 'NgayHD khong hop le'); end if; END LOOP; end; </pre>		
Mô tả mã lệnh:			
<p>Mã lệnh này có chức năng kiểm tra trước khi thêm mới hoặc cập nhật bản ghi trong bảng "HOADON". Trigger sẽ thực hiện truy vấn để lấy NGAYSINH của KHACHHANG tương ứng với MAKH và kiểm tra trường "NGAYHD" của bản ghi mới hoặc sửa đổi có hợp lệ hay không. Nếu "NGAYHD" nhỏ hơn "NGAYSINH", sẽ gây ra lỗi và thông báo lỗi cho người dùng.</p> <p>Với mỗi dòng được thêm mới hoặc cập nhật trong bảng "HOADON", trigger sẽ thực hiện các bước trong khối code sau:</p> <ul style="list-style-type: none"> Một con trỏ (cursor) được khai báo để thực hiện truy vấn lấy NGAYSINH trong bảng KHACHHANG với điều kiện MAKH bằng với giá trị của trường "MAKH" trong bản ghi mới hoặc sửa đổi "MAKH". Khai báo biến "v_NGAYSINH" để lưu trữ kết quả của truy vấn lấy NGAYSINH. Mở con trỏ curNGSINH và sử dụng vòng lặp để duyệt qua kết quả trả về từ truy vấn lấy NGAYSINH. Nếu không tìm thấy bản ghi nào thì thoát khỏi vòng lặp. Trong vòng lặp, trigger sẽ kiểm tra trường "NGAYHD" của bản ghi mới hoặc sửa đổi với giá trị của "NGAYSINH". Nếu "NGAYHD" nhỏ hơn "NGAYSINH", sẽ gây ra lỗi thông qua hàm "raise_application_error". Lỗi sẽ có mã -20006 và thông báo "NgayHD khong hop le". 			
2.7	TRG_ThanhTien_HOADON_in	INSERT	HOADON

	<p>Mã lệnh PL/SQL</p> <pre>create or replace trigger TRG22_ThanhTien_HOADON_in before insert on HOADON for each row begin if (:new.ThanhTien != 0) then raise_application_error(-20007, 'Khi them moi 1 HOADON, ThanhTien = 0 (hệ thống sẽ tự động tính lại khi hoàn tất insert CTHD thuoc HD nay)'); end if; end;</pre>			
2.8	<p>Mô tả mã lệnh:</p> <p>Mã lệnh này có chức năng đảm bảo giá trị của trường "ThanhTien" trong bảng "HOADON" phải là 0 khi thêm mới bản ghi, vì hệ thống sẽ tính lại giá trị này khi hoàn tất thêm mới các bản ghi tương ứng trong bảng chi tiết hóa đơn (CTHD).</p> <table border="1"> <tr> <td style="text-align: center;">TRG_DiemTL_KHACHHANG_in</td> <td style="text-align: center;">INSERT</td> <td style="text-align: center;">KHACHHANG</td> </tr> </table>	TRG_DiemTL_KHACHHANG_in	INSERT	KHACHHANG
TRG_DiemTL_KHACHHANG_in	INSERT	KHACHHANG		
	<p>Mã lệnh PL/SQL</p> <pre>create or replace trigger TRG25_DiemTL_KHACHHANG_in before insert on KHACHHANG for each row begin if (:new.DIEMTICHLUY != 0) then raise_application_error(-20008, 'Khi them moi 1 KHACHHANG, DIEMTICHLUY = 0 (hệ thống sẽ tự động tính lại khi hoàn tất insert CTHD thuoc HOADON cua KHACHHANG nay)'); end if; end;</pre>			

	<u>Mô tả mã lệnh:</u> Mã lệnh này có chức năng kiểm tra định dạng dữ liệu trước khi thêm mới bản ghi trong bảng "KHACHHANG". Trigger sẽ kiểm tra liệu trường "DIEMTICHLUY" trong bản ghi mới có bằng 0 hay không. Nếu không bằng 0, sẽ gây ra lỗi và thông báo lỗi cho người dùng.		
2.9	TRG_AUTO_SL_ThanhTien_DiemTL_LoaiKH_CTHD_up	UPDATE	CTHD
<u>Mã lệnh PL/SQL</u>			
<pre>create or replace trigger TRG_AUTO_SL_ThanhTien_DiemTL_CTHD_up after update on CTHD for each row declare v_GiaSach number; v_GiaCTHDNew number; v_MaKH KHACHHANG.MAKH%type; v_DTL number; begin select Gia into v_GiaSach from Sach where MaSach = :new.MaSach; update SACH set SL = SL - :new.SL where MaSach = :new.MaSach; v_GiaCTHDNew := :new.SL * v_GiaSach; update HOADON set ThanhTien = ThanhTien + v_GiaCTHDNew</pre>			

	<pre> where MaHD = :new.MaHD; select MaKH into v_MaKH from HOADON where MaHD = :new.MaHD; if (v_MaKH is not null) then update KHACHHANG set DIEMTICHLUY = DIEMTICHLUY + v_GiaCTHDNew where MaKH = v_MaKH; select DIEMTICHLUY into v_DTL from KHACHHANG where MAKH = v_MaKH; if (v_DTL >= 2000000) then update KHACHHANG set LOAIKH = 'VIP' where MAKH = v_MaKH; elsif (v_DTL >= 500000) then update KHACHHANG set LOAIKH = 'Thân Thiết' where MAKH = v_MaKH; elsif (v_DTL < 500000) then update KHACHHANG set LOAIKH = 'Thường' where MAKH = v_MaKH; end if; end if; </pre>
--	---

	<p>end;</p> <p>Mô tả mã lệnh:</p> <p>Mã lệnh này có chức năng tự động cập nhật thông tin cho các bảng "Sach", "HOADON" và "KHACHHANG" sau khi có bản ghi được cập nhật trong bảng "CTHD". Cụ thể, trigger sẽ cập nhật số lượng sách tồn lại, giá trị hoá đơn, điểm tích lũy và loại khách hàng tương ứng với hoá đơn trong bản ghi mới nhất của bảng "CTHD".</p> <p>Với mỗi dòng trong bảng "CTHD" được cập nhật, trigger sẽ thực hiện các bước trong khối code sau:</p> <ul style="list-style-type: none"> • Tạo các biến "v_GiaSach" (kiểu số), "v_GiaCTHDNew" (kiểu số), "v_MaKH" (kiểu "MAKH" của bảng "KHACHHANG") và "v_DTL" (kiểu số). • Lấy giá sách từ bảng "Sach" với mã sách tương ứng trong bản ghi mới. • Cập nhật số lượng sách tồn lại trong bảng "Sach". • Tính giá trị mới của trường "ThanhTien" trong bảng "HOADON" tương ứng với mã hoá đơn trong bản ghi mới nhất của bảng "CTHD". • Cập nhật giá trị mới của trường "ThanhTien" trong bảng "HOADON". • Lấy mã khách hàng từ bảng "HOADON". • Cập nhật điểm tích lũy cho khách hàng trong bảng "KHACHHANG". • Lấy điểm tích lũy của khách hàng sau khi cập nhật. • Dựa trên số điểm tích lũy mới của khách hàng, cập nhật loại khách hàng tương ứng trong bảng "KHACHHANG". 		
2.10	<p>TRG_AUTO_SL_ThanhTien_DiemTL_LoaiKH_CTHD_de_up</p>	<p>DELETE UPDATE</p>	CTHD
<p>Mã lệnh PL/SQL</p> <pre>create or replace trigger TRG_AUTO_ThanhTien_DiemTL_CTHD_de_up after delete or update on CTHD for each row declare v_GiaSach number; v_GiaCTHDOld number;</pre>			

	<pre> v_MaKH KHACHHANG.MAKH%type; v_DTL number; begin select Gia into v_GiaSach from Sach where MaSach = :old.MaSach; update SACH set SL = SL + :old.SL where MaSach = :old.MaSach; v_GiaCTHDOld := :old.SL * v_GiaSach; update HOADON set ThanhTien = ThanhTien - v_GiaCTHDOld where MaHD = :old.MaHD; select MaKH into v_MaKH from HOADON where MaHD = :old.MaHD; if (v_MaKH is not null) then update KHACHHANG set DIEMTICHLUY = DIEMTICHLUY - v_GiaCTHDOld where MaKH = v_MaKH; select DIEMTICHLUY into v_DTL from KHACHHANG </pre>
--	--

	<pre> where MAKH = v_MaKH; if (v_DTL >= 2000000) then update KHACHHANG set LOAIKH = 'VIP' where MAKH = v_MaKH; elsif (v_DTL >= 500000) then update KHACHHANG set LOAIKH = 'Thân Thiết' where MAKH = v_MaKH; elsif (v_DTL < 500000) then update KHACHHANG set LOAIKH = 'Thường' where MAKH = v_MaKH; end if; end if; end; </pre> <p>Mô tả mã lệnh:</p> <p>Mã lệnh này có chức năng tự động cập nhật thông tin cho các bảng "Sach", "HOADON" và "KHACHHANG" sau khi có bản ghi bị xóa hoặc cập nhật trong bảng "CTHD". Cụ thể, trigger sẽ cập nhật số lượng sách tồn lại, giá trị hoá đơn, điểm tích lũy và loại khách hàng tương ứng với hoá đơn trong bản ghi xóa hoặc cập nhật của bảng "CTHD".</p> <p>Với mỗi dòng trong bảng "CTHD" được xóa hoặc cập nhật, trigger sẽ thực hiện các bước trong khối code sau:</p> <ul style="list-style-type: none"> Tạo các biến "v_GiaSach" (kiểu số), "v_GiaCTHDOld" (kiểu số), "v_MaKH" (kiểu "MAKH" của bảng "KHACHHANG") và "v_DTL" (kiểu số). Lấy giá sách từ bảng "Sach" với mã sách tương ứng trong bản ghi bị xóa hoặc cập nhật.
--	---

	<ul style="list-style-type: none"> Cập nhật số lượng sách tồn lại trong bảng "Sach". Tính giá trị hiện tại của trường "ThanhTien" trong bảng "HOADON" tương ứng với mã hoá đơn trong bản ghi xóa hoặc cập nhật. Cập nhật giá trị mới của trường "ThanhTien" trong bảng "HOADON". Lấy mã khách hàng từ bảng "HOADON". Cập nhật điểm tích lũy cho khách hàng trong bảng "KHACHHANG". Lấy điểm tích lũy của khách hàng sau khi cập nhật. Dựa trên số điểm tích lũy mới của khách hàng, cập nhật loại khách hàng tương ứng trong bảng "KHACHHANG". 		
2.11	<p style="text-align: center;">TRG23_GIANHAP_GIABAN_CTPN_in_up</p>	INSERT, UPDATE	CTPN
<p>Mã lệnh PL/SQL</p> <pre>create or replace trigger TRG23_GIANHAP_GIABAN_CTPN before insert or update of MASACH, GIA on CTPN for each row declare cursor curGiaBanSach is select Gia from SACH where MASACH = :new.MASACH; v_GiaBan number; begin open curGiaBanSach; LOOP FETCH curGiaBanSach into v_GiaBan; EXIT WHEN curGiaBanSach%NOTFOUND; if (:new.Gia > v_GiaBan) then raise_application_error(-20009, 'Gia Nhập không phù hợp'); end if; END LOOP; end;</pre> <p>Mô tả mã lệnh:</p>			

	<p>Mã lệnh này có chức năng đảm bảo giá nhập của sách trong bảng "CTPN" không được lớn hơn giá bán của sách trong bảng "SACH". Trigger sẽ sử dụng con trỏ để lấy giá bán tương ứng với mã sách trong bản ghi mới/cũ trong bảng "CTPN", và so sánh với giá bán từ bảng "SACH". Nếu giá nhập lớn hơn giá bán, trigger sẽ gây ra lỗi và ngăn chặn thêm mới hoặc cập nhật bản ghi trong bảng "CTPN".</p> <p>Với mỗi dòng trong bảng "CTPN" được thêm mới hoặc cập nhật, trigger sẽ thực hiện các bước trong khối code sau:</p> <ul style="list-style-type: none"> Tạo một con trỏ "curGiaBanSach" để lấy giá bán của sách từ bảng "SACH" với mã sách tương ứng trong bản ghi mới/cũ. Tạo biến "v_GiaBan" để lưu giá bán của sách. Mở con trỏ. Với mỗi giá bán trong danh sách các giá bán lấy được từ bảng "SACH": <ul style="list-style-type: none"> Kiểm tra nếu giá nhập của sách trong bản ghi mới/cũ lớn hơn giá bán của sách, sẽ gây ra lỗi thông qua hàm "raise_application_error". Lỗi sẽ có mã -20009 và thông báo "Gia Nhập khong phu hop". Thoát vòng lặp. 		
2.12	<p>TRG_AUTO_checkGIA_ThanhTien_DIEMTL_LOAIKH_SACH_upGIA</p> <p>Mã lệnh PL/SQL</p> <pre>create or replace trigger TRG_AUTO_checkGIA_ThanhTien_DIEMTL_LOAIKH_SACH_upGIA after update of GIA on SACH for each row declare cursor curMaHD is select MaHD, SL from CTHD where MaSach = :new.MaSach; cursor curCTNhaphSach is select Gia from CTPN where MASACH = :new.MASACH; v_GiaNhaph number; v_MaHD HOADON.MaHD%type;</pre>	UPDATE	SACH

	<pre> v_SL CTHD.SL%type; v MAKH KHACHHANG.MAKH%type; v_DTL KHACHHANG.DIEMTICHLUY%type; begin open curCTNhapSach; LOOP FETCH curCTNhapSach into v_GiaNhap; EXIT WHEN curCTNhapSach%NOTFOUND; if (:new.Gia < v_GiaNhap) then raise_application_error(-20010, 'Gia ban khong phu hop'); end if; END LOOP; open curMaHD; LOOP FETCH curMaHD into v_MaHD, v_SL; EXIT WHEN curMaHD%NOTFOUND; update HOADON set ThanhTien = ThanhTien + v_SL * :new.Gia where MaHD = v_MaHD; update HOADON set ThanhTien = ThanhTien - v_SL * :old.Gia where MaHD = v_MaHD; select MAKH into v_MaKH from HOADON where MaHD = v_MaHD; </pre>
--	---

```
if (v_MaKH is not null) then
    update KHACHHANG
    set DIEMTICHLUY = DIEMTICHLUY + v_SL * :new.Gia
    where MAKH = v_MAKH;

    update KHACHHANG
    set DIEMTICHLUY = DIEMTICHLUY - v_SL * :old.Gia
    where MAKH = v_MAKH;

    select DIEMTICHLUY into v_DTL
    from KHACHHANG
    where MAKH = v_MAKH;

if (v_DTL >= 2000000) then
    update KHACHHANG
    set LOAIKH = 'VIP'
    where MAKH = v_MaKH;
elsif (v_DTL >= 500000) then
    update KHACHHANG
    set LOAIKH = 'Thân Thiết'
    where MAKH = v_MaKH;
elsif (v_DTL < 500000) then
    update KHACHHANG
    set LOAIKH = 'Thường'
    where MAKH = v_MaKH;
end if;
end if;
```

```

    END LOOP;
end;

```

Mô tả mã lệnh:

Mã lệnh này có chức năng là cập nhật lại giá trị trường "ThanhTien" trong bảng "HOADON" sau khi có bản ghi trong bảng "SACH" được cập nhật trường "GIA". Trigger sẽ sử dụng hai con trỏ để lấy danh sách giá nhập của sách trong bảng "CTPN", và danh sách mã hoá đơn và số lượng của mã sách tương ứng trong bản ghi mới. Với mỗi giá nhập, trigger sẽ kiểm tra nếu giá bán của sách trong bảng "SACH" mới cập nhật nhỏ hơn giá nhập, sẽ gây ra lỗi và ngăn chặn việc cập nhật. Sau đó, trigger sẽ lấy danh sách các mã hoá đơn và số lượng tương ứng để cập nhật lại trường "ThanhTien" trong bảng "HOADON".

Với mỗi dòng được cập nhật trường GIA của bảng "SACH", trigger sẽ thực hiện các bước trong khối code sau:

- Tạo một con trỏ "curCTNhaphSach" để lấy giá nhập của sách trong bảng "CTPN" tương ứng với mã sách trong bản ghi mới.
- Tạo một con trỏ "curMaHD" để lấy danh sách các mã hoá đơn và số lượng của mã sách tương ứng trong bản ghi.
- Tạo các biến "v_GiaNhaph" (kiểu số), "v_MaHD" (kiểu "MaHD" của bảng "HOADON") và "v_SL" (kiểu số).
- Mở con trỏ "curCTNhaphSach".
- Với mỗi giá nhập trong danh sách các giá nhập lấy được từ bảng "CTPN":
 - Kiểm tra nếu giá bán của sách trong bản ghi mới nhỏ hơn giá nhập của sách, sẽ gây ra lỗi thông qua hàm "raise_application_error". Lỗi sẽ có mã -20010 và thông báo "Gia ban khong phu hop".
- Thoát vòng lặp.
- Mở con trỏ "curMaHD".
- Với mỗi mã hoá đơn và số lượng tương ứng trong danh sách mã số mã hoá đơn sao khi cập nhật giá bán cho sách:

	<ul style="list-style-type: none"> ○ Cập nhật trường "ThanhTien" trong bảng "HOADON" với giá trị mới của giá bán. ○ Cập nhật trường "ThanhTien" trong bảng "HOADON" với giá trị cũ của giá bán. ● Thoát vòng lặp. 		
2.13	TRG_TongTienNhapSach_PhieuNhap_in	INSERT	PHIEUNHAP
<u>Mã lệnh PL/SQL</u>			
<pre>create or replace trigger TRG_TongTienNhapSach_PhieuNhap_in before insert on PHIEUNHAP for each row begin if (:new.TongTien != 0) then raise_application_error(-20011, 'Khi them moi 1 PhieuNhap tongtien = 0 (hệ thống sẽ tự động tính lại khi hoàn tất insert)'); end if; end;</pre>			
<u>Mô tả mã lệnh:</u>			
<p>Mã lệnh này có chức năng kiểm tra định dạng dữ liệu trước khi thêm mới bản ghi trong bảng "PHIEUNHAP". Trigger sẽ kiểm tra liệu trường "TongTien" trong bản ghi mới có bằng 0 hay không. Nếu không bằng 0, sẽ gây ra lỗi và thông báo lỗi cho người dùng.</p>			
2.14	TRG_AUTO_SL_TongTienNhapSach_CTPN_up	UPDATE	CTPN
<u>Mã lệnh PL/SQL</u>			
<pre>create or replace trigger TRG_AUTO_SL_TongTienNhapSach_CTPN_up after update on CTPN for each row begin update PHIEUNHAP</pre>			

	<pre> set TongTien = TongTien + :new.SL * :new.Gia where MAPN = :new.MaPN; update SACH set SL = SL + :new.SL where MaSach = :new.MaSach; end; </pre> <p>Mô tả mã lệnh:</p> <p>Mã lệnh này có chức năng là cập nhật lại trường "TongTien" trong bảng "PHIEUNHAP" sau khi có bản ghi trong bảng "CTPN" được cập nhật. Trigger sẽ lấy thông tin số lượng sách và giá nhập mới của bản ghi, tính toán giá trị mới của trường "TongTien" và cập nhật lại.</p>		
2.15	<p>TRG_AUTO_SL_TongTienNhapSach_CTPN_de_up</p>	<p>DELETE, UPDATE</p>	CTPN
	<p>Mã lệnh PL/SQL</p> <pre> create or replace trigger TRG_AUTO_SL_TongTienNhapSach_CTPN_de_up after delete or update on CTPN for each row declare v_SL number; v_TT number; begin select SL into v_SL from SACH where MASACH = :old.MaSach; select TongTien into v_TT from PHIEUNHAP where MAPN = :old.MaPN; </pre>		

	<pre> if (v_SL >= :old.SL) then update SACH set SL = SL - :old.SL where MaSach = :old.MaSach; end if; if (v_TT >= :old.SL * :old.Gia) then update PHIEUNHAP set TongTien = TongTien - :old.SL * :old.Gia where MAPN = :old.MaPN; end if; end; </pre> <p>Mô tả mã lệnh:</p> <p>Mã lệnh này có chức năng cập nhật lại trường "TongTien" trong bảng "PHIEUNHAP" sau khi có bản ghi trong bảng "CTPN" bị xóa hoặc cập nhật. Trigger sẽ lấy thông tin số lượng sách và giá nhập cũ của bản ghi cũ, tính toán giá trị cũ của trường "TongTien" và cập nhật lại.</p>
--	--

3. Xây dựng và mô tả Procedure, Function trong đồ án môn học

3.1 Procedure

STT	Procedure
3.1.1	<p>Tên Procedure: sp_ThemCTPN</p> <p>Mã lệnh PL/SQL:</p> <pre> create or replace procedure sp_ThemCTPN(v_MaPN_in IN CTPN.MaPN%type, v_MaSach_in IN CTPN.MaSach%type, v_SL_in IN CTPN.SL%type, v_Gia_in IN CTPN.GIA%type) </pre>

```

as
    v_sl number;
    v_TongTien number;
begin
    select SL into v_sl
    from SACH
    where MaSach = v_MaSach_in;

    select TongTien into v_TongTien
    from PHIEUNHAP
    where MAPN = v_MaPN_in;

    insert into CTPN(MaPN, MaSach, SL, Gia) values(v_MaPN_in, v_MaSach_in,
    v_SL_in, v_Gia_in);

    update PHIEUNHAP
    set TongTien = v_TongTien + v_SL_in * v_Gia_in
    where MAPN = v_MaPN_in;

    update SACH
    set SL = v_sl + v_SL_in
    where MaSach = v_MaSach_in;
    commit;
end;

```

Mô tả mã lệnh:

- Đoạn mã lệnh PL/SQL trên tạo ra một thủ tục có tên là "sp_ThemCTPN" dùng để thêm thông tin của một chi tiết phiếu nhập (CTPN) vào trong cơ sở dữ liệu.
- Các thông số đầu vào của thủ tục gồm: mã phiếu nhập, mã sách, số lượng, giá.

	<ul style="list-style-type: none"> Thủ tục này truy xuất số lượng sách và tổng tiền của phiếu nhập dựa trên mã sách và mã phiếu nhập, sau đó thêm thông tin chi tiết phiếu nhập vào bảng CTPN và cập nhật số lượng sách trong bảng SACH và tổng tiền của phiếu nhập trong bảng PHIEUNHAP. Cuối cùng, thực hiện commit để lưu các thay đổi vào cơ sở dữ liệu.
3.1.2	<p>Tên procedure: sp_ThemCTHD</p> <p>Mã lệnh PL/SQL:</p> <pre>create or replace procedure sp_ThemCTHD(v_MaHD_in IN CTHD.MaHD%type, v_MaSach_in IN CTHD.MaSach%type, v_SL_in IN CTHD.SL%type) as v_MaKH KHACHHANG.MAKH%type; v_GiaSach SACH.GIA%type; v_valueCTHD number; v_DTL KHACHHANG.DIEMTICHLUY%type; begin insert into CTHD(MaHD, MaSach, SL) values(v_MaHD_in, v_MaSach_in, v_SL_in); update SACH set SL = SL - v_SL_in where MaSach = v_MaSach_in; select Gia into v_GiaSach from SACH where MaSach = v_MaSach_in; v_valueCTHD := v_SL_in * v_GiaSach;</pre>

```
update HOADON
set ThanhTien = ThanhTien + v_valueCTHD
where MAHD = v_MAHD_in;

select MAKH into v_MaKH
from HOADON
where MAHD = v_MaHD_in;

if (v_MaKH is not null) then
    update KHACHHANG
    set DIEMTICHLUY = DIEMTICHLUY + v_valueCTHD
    where MAKH = v_MaKH;

select DIEMTICHLUY into v_DTL
from KHACHHANG
where MAKH = v_MaKH;

if (v_DTL >= 2000000) then
    update KHACHHANG
    set LOAIKH = 'VIP'
    where MAKH = v_MaKH;
elseif (v_DTL >= 500000) then
    update KHACHHANG
    set LOAIKH = 'Thân Thiết'
    where MAKH = v_MaKH;
elseif (v_DTL < 500000) then
    update KHACHHANG
```

	<pre> set LOAIKH = 'Thường' where MAKH = v_MaKH; end if; end if; commit; end; </pre> <p>Mô tả mã lệnh:</p> <ul style="list-style-type: none"> Procedure trên có tên là sp_ThemCTHD, nhiệm vụ của nó là thêm một chi tiết hóa đơn mới vào trong bảng CTHD. Đầu vào của procedure bao gồm ba tham số: v_MaHD_in là mã hóa đơn, v_MaSach_in là mã sách và v_SL_in là số lượng sách mua. Trong thân procedure, các hành động được thực hiện bao gồm: <ul style="list-style-type: none"> Thực hiện lệnh insert để thêm chi tiết hóa đơn vào trong bảng CTHD. Cập nhật lại số lượng sách trong bảng SACH bằng lệnh update. Lấy giá của sách từ bảng SACH và tính giá trị của chi tiết hóa đơn bằng cách nhân số lượng sách với giá của sách. Cập nhật lại tổng giá trị của hóa đơn trong bảng HOADON bằng lệnh update dựa trên mã hóa đơn. Lấy mã khách hàng từ bảng HOADON và cập nhật điểm tích lũy của khách hàng tương ứng trong bảng KHACHHANG. Sau đó, lấy điểm tích lũy đó và kiểm tra xem khách hàng đó thuộc loại nào (thường, thân thiết hay VIP) bằng cách so sánh với các mốc điểm tích lũy đã xác định trước đó. Nếu đạt được mốc điểm tích lũy nào đó thì cập nhật lại loại khách hàng của khách hàng đó trong bảng KHACHHANG. Cuối cùng, sử dụng lệnh commit để lưu trữ thay đổi vào cơ sở dữ liệu.
3.1.3	<p>Tên Procedure: sp_XoaNhanVien</p> <p>Mã lệnh PL/SQL:</p>

```

create or replace procedure sp_XoaNHANVIEN(
    v_MaNV_in NHANVIEN.MANV%type
)
as
begin
    delete from CTHD where MaHD in (select MaHD from HOADON where MaNV =
v_MaNV_in);
    delete from HOADON where MaNV = v_MaNV_in;
    delete from CTPN where MaPN in (select MaPN from PHIEUNHAP where MaNV =
v_MaNV_in);
    delete from PHIEUNHAP where MaNV = v_MaNV_in;
    delete from TAIKHOAN where MANV = v_MaNV_in;
    delete from NHANVIEN where MaNV = v_MaNV_in;
    commit;
end;

```

Mô tả mã lệnh:

Procedure trên có tên là sp_XoaNHANVIEN, nhiệm vụ của nó là xóa toàn bộ thông tin của một nhân viên dựa trên mã nhân viên được đưa vào đầu vào của procedure.

Trong thân procedure, các hành động được thực hiện bao gồm:

- Xóa hết các chi tiết hóa đơn có liên quan đến nhân viên đó bằng cách sử dụng lệnh delete kết hợp với câu lệnh where để xác định những chi tiết hóa đơn cần xóa. Điều kiện để xóa là mã nhân viên truyền vào phải trùng với mã nhân viên của hóa đơn đó.
- Tiếp đó, xóa hết các phiếu nhập cũng như các chi tiết phiếu nhập liên quan đến nhân viên bằng cách sử dụng lệnh delete kết hợp với câu lệnh where. Câu lệnh where xác định những phiếu nhập cần xóa bằng cách so sánh mã nhân viên truyền vào với mã nhân viên của các phiếu nhập đó.
- Xóa tài khoản của nhân viên đó trong bảng TAIKHOAN bằng lệnh delete.

	<ul style="list-style-type: none"> Cuối cùng, xóa nhân viên đó khỏi bảng NHANVIEN bằng lệnh delete với điều kiện là mã nhân viên truyền vào. Sau mỗi lệnh delete, của lượng bản ghi đã bị xóa có thể được lưu trữ vào cơ sở dữ liệu bằng câu lệnh commit.
3.1.4	<p>Tên Procedure: sp_XoaKHACHHANG</p> <p>Mã lệnh PL/SQL:</p> <pre>create or replace procedure sp_XoaKHACHHANG(v_MaKH_in KHACHHANG.MAKH%type) as begin delete from CTHD where MaHD in (select MaHD from HOADON where MaKH = v_MaKH_in); delete from HOADON where MaKH = v_MaKH_in; delete from KHACHHANG where MAKH = v_MaKH_in; commit; end;</pre> <p>Mô tả mã lệnh:</p> <p>Procedure trên có tên là sp_XoaKHACHHANG, nhiệm vụ của nó là xóa toàn bộ thông tin của một khách hàng dựa trên mã khách hàng được đưa vào đầu vào của procedure. Trong thân procedure, các hành động được thực hiện bao gồm:</p> <ul style="list-style-type: none"> Xóa hết các chi tiết hóa đơn có liên quan đến khách hàng đó bằng cách sử dụng lệnh delete kết hợp với câu lệnh where để xác định những chi tiết hóa đơn cần xóa. Điều kiện để xóa là mã khách hàng truyền vào phải trùng với mã khách hàng của hóa đơn đó. Tiếp đó, xóa hết các hóa đơn liên quan đến khách hàng bằng cách sử dụng lệnh delete kết hợp với câu lệnh where. Câu lệnh where xác định những hóa đơn cần xóa bằng cách so sánh mã khách hàng truyền vào với mã khách hàng của các hóa đơn đó.

	<ul style="list-style-type: none"> Xóa khách hàng đó trong bảng KHACHHANG bằng lệnh delete với điều kiện là mã khách hàng truyền vào. Sau mỗi lệnh delete, của lượng bản ghi đã bị xóa có thể được lưu trữ vào cơ sở dữ liệu bằng câu lệnh commit.
3.1.5	<p>Tên Procedure: sp_XoaHOADON</p> <p>Mã lệnh PL/SQL:</p> <pre>create or replace procedure sp_XoaHOADON(v_MaHD_in HOADON.MAHD%type) as begin delete from CTHD where MaHD = v_MaHD_in; delete from HOADON where MaHD = v_MaHD_in; commit; end;</pre> <p>Mô tả mã lệnh:</p> <p>Procedure trên có tên là sp_XoaHOADON, nhiệm vụ của nó là xóa toàn bộ thông tin của một hóa đơn dựa trên mã hóa đơn được đưa vào đầu vào của procedure.</p> <p>Trong thân procedure, các hành động được thực hiện bao gồm:</p> <ul style="list-style-type: none"> Xóa toàn bộ chi tiết hóa đơn trong bảng CTHD liên quan đến hóa đơn cần xóa bằng lệnh delete kết hợp với điều kiện là mã hóa đơn truyền vào. Xóa toàn bộ thông tin của hóa đơn trong bảng HOADON bằng lệnh delete kết hợp với điều kiện là mã hóa đơn truyền vào. Sau khi xóa thành công, sử dụng lệnh commit để lưu lại sự thay đổi vào cơ sở dữ liệu.
3.1.6	<p>Tên Procedure: sp_XoaPHIEUNHAP</p> <p>Mã lệnh PL/SQL:</p> <pre>create or replace procedure sp_XoaPHIEUNHAP(v_MaPN_in PHIEUNHAP.MAPN%type)</pre>

	<pre>) as begin delete from CTPN where MaPN = v_MaPN_in; delete from PHIEUNHAP where MaPN = v_MaPN_in; commit; end; </pre> <p>Mô tả mã lệnh:</p> <p>Procedure trên có tên là sp_XoaPhieuNhap, nhiệm vụ của nó là xóa toàn bộ thông tin của một phiếu nhập dựa trên mã phiếu nhập được đưa vào đầu vào của procedure.</p> <p>Trong thân procedure, các hành động được thực hiện bao gồm:</p> <ul style="list-style-type: none"> • Xóa toàn bộ chi tiết phiếu nhập trong bảng CTPN liên quan đến phiếu nhập cần xóa bằng lệnh delete kết hợp với điều kiện là mã phiếu nhập truyền vào. • Xóa toàn bộ thông tin của phiếu nhập trong bảng PHIEUNHAP bằng lệnh delete kết hợp với điều kiện là mã phiếu nhập truyền vào. • Sau khi xóa thành công, sử dụng lệnh commit để lưu lại sự thay đổi vào cơ sở dữ liệu.
3.1.7	<p>Tên Procedure: sleep</p> <p>Mã lệnh PL/SQL:</p> <pre> create or replace NONEDITIONABLE PROCEDURE sleep (in_time number) AS v_now date; BEGIN SELECT SYSDATE INTO v_now FROM DUAL; LOOP </pre>

	<pre> EXIT WHEN v_now + (in_time * (1/86400)) <= SYSDATE; END LOOP; end; <u>Mô tả mã lệnh:</u> Procedure trên có tên là sleep, nhiệm vụ của nó là tạm dừng thực thi thủ tục trong một khoảng thời gian được truyền vào. Trong thân procedure, các hành động được thực hiện bao gồm: </pre> <ul style="list-style-type: none"> • Khai báo biến v_now kiểu dữ liệu là date để lưu giá trị của hàm SYSDATE. • Sử dụng câu lệnh select SYSDATE into v_now from dual để gán giá trị của hàm SYSDATE vào biến v_now. • Sử dụng vòng lặp loop để tạm dừng thực thi thủ tục trong khoảng thời gian được truyền vào, bằng cách so sánh thời điểm bây giờ với thời điểm lúc thực hiện phương thức cộng thêm thời gian delay. • Để tính thời gian delay, sử dụng biến in_time truyền vào và chuyển đổi về đơn vị giây bằng cách nhân với 1/86400 (một ngày có 86400 giây). • Khi thời gian tạm dừng đã đủ, vòng lặp sẽ kết thúc và thực thi tiếp các câu lệnh phía sau. • Procedure này không thực hiện bất kỳ thao tác nào trên cơ sở dữ liệu, vì vậy không cần thực hiện commit.
--	---

3.2 Function

STT	Function
3.2.1	<p>Tên function: insert_new_MaHD</p> <p>Mã lệnh PL/SQL:</p> <pre> create or replace function insert_new_MaHD return number is v_MaHD_new HOADON.MAHD%type; </pre>

	<pre> begin select MAX(MaHD) into v_MaHD_new from HOADON; return v_MaHD_new + 1; end; </pre> <p>Mô tả mã lệnh:</p> <p>Function trên có tên là insert_new_MaHD, nhiệm vụ của nó là trả về số liệu kiểu số nguyên (number) để tạo mã hóa đơn mới (MaHD) khi thêm hóa đơn vào bảng HOADON.</p> <p>Trong thân function, các hành động được thực hiện bao gồm:</p> <ul style="list-style-type: none"> Khởi tạo biến v_MaHD_new kiểu dữ liệu là MAHD của bảng HOADON để lưu giá trị của mã hóa đơn mới. Sử dụng câu lệnh select MAX(MaHD) into v_MaHD_new from HOADON để lấy giá trị lớn nhất trong trường MaHD của bảng HOADON. Trả về giá trị của biến v_MaHD_new cộng thêm 1, đây chính là mã hóa đơn mới. Function này không thực hiện bất kỳ thao tác nào trên cơ sở dữ liệu, vì vậy không cần thực hiện commit.
3.2.2	<p>Tên function: insert_new_MaPN</p> <p>Mã lệnh PL/SQL:</p> <pre> create or replace function insert_new_MaPN return number is v_MaPN_new PHIEUNHAP.MAPN%type; begin select MAX(MaPN) into v_MaPN_new from PHIEUNHAP; </pre>

	<pre> return v_MaPN_new + 1; end; Mô tả mã lệnh: tương tự mô tả trên </pre>
3.2.3	<p>Tên function: insert_new_MaNV</p> <p>Mã lệnh PL/SQL:</p> <pre> create or replace function insert_new_MaNV return number is v_MaNV_new NHANVIEN.MANV%type; begin select MAX(MaNV) into v_MaNV_new from NHANVIEN; return v_MaNV_new + 1; end; </pre> <p>Mô tả mã lệnh: tương tự mô tả trên</p>
3.2.4	<p>Tên function: insert_new_MaKH</p> <p>Mã lệnh PL/SQL:</p> <pre> create or replace function insert_new_MaKH return number is v_MaKH_new KHACHHANG.MAKH%type; begin select MAX(MaKH) into v_MaKH_new from KHACHHANG; return v_MaKH_new + 1; </pre>

	<p>end;</p> <p>Mô tả mã lệnh: tương tự mô tả trên</p>
3.2.5	<p>Tên function: insert_new_MaSach</p> <p>Mã lệnh PL/SQL:</p> <pre>create or replace function insert_new_MaSach return number is v_MaSach_new SACH.MASACH%type; begin select MAX(MaSach) into v_MaSach_new from SACH; return v_MaSach_new + 1; end;</pre> <p>Mô tả mã lệnh: tương tự mô tả trên</p>
3.2.6	<p>Tên function: insert_new_MaTK</p> <p>Mã lệnh PL/SQL:</p> <pre>create or replace function insert_new_MaTK return number is v_MaTK_new TAIKHOAN.MATK%type; begin select MAX(MaTK) into v_MaTK_new from TAIKHOAN; return v_MaTK_new + 1; end;</pre> <p>Mô tả mã lệnh: tương tự mô tả trên</p>

3.2.7	<p>Tên function: insert_new_MaTL</p> <p>Mã lệnh PL/SQL:</p> <pre>create or replace function insert_new_MaTL return number is v_MaTL_new THELOAI.MATL%type; begin select MAX(MaTL) into v_MaTL_new from THELOAI; return v_MaTL_new + 1; end;</pre> <p>Mô tả mã lệnh: tương tự mô tả trên</p>
3.2.8	<p>Tên function: insert_new_Mancc</p> <p>Mã lệnh PL/SQL:</p> <pre>create or replace function insert_new_Mancc return number is v_Mancc_new NHACUNGCAP.MANCC%type; begin select MAX(Mancc) into v_Mancc_new from NHACUNGCAP; return v_Mancc_new + 1; end;</pre> <p>Mô tả mã lệnh: tương tự mô tả trên</p>
3.2.9	<p>Tên function: fn_DoanhThuTheoThangNam</p> <p>Mã lệnh PL/SQL:</p>

```

create or replace function fn_DoanhThuTheoThangNam(
    f_month_in number,
    f_year_in number
)
return number
is
    v_total number;
begin
    select sum(ThanhTien) into v_total
    from HOADON
    where extract(Month from NGAYHD) = f_month_in and extract(Year from
NGAYHD) = f_Year_in;

    if (v_total is null) then
        v_total := 0;
    end if;

    return v_total;
end;

```

Mô tả mã lệnh:

Function trên có tên là fn_DoanhThuTheoThangNam, nhiệm vụ của nó là trả về doanh thu từ các hóa đơn trong tháng và năm được truyền vào.

Trong thân function, các hành động được thực hiện bao gồm:

- Khai báo biến v_total kiểu dữ liệu là number để lưu tổng giá trị ThanhTien của các hóa đơn.
- Sử dụng câu lệnh select sum(ThanhTien) into v_total from HOADON để tính tổng giá trị ThanhTien của các hóa đơn trên bảng HOADON.

	<ul style="list-style-type: none"> Sử dụng hàm extract để lọc các hóa đơn trong tháng và năm được truyền vào dựa trên trường NGAYHD của bảng HOADON. Kiểm tra nếu giá trị của biến v_total là null thì gán giá trị bằng 0. Trả về giá trị của biến v_total, tức là doanh thu từ các hóa đơn trong tháng và năm được truyền vào. Function này không thực hiện bất kỳ thao tác nào trên cơ sở dữ liệu, vì vậy không cần thực hiện commit.
3.2.10	<p>Tên function: fn_ChiPhiNhapSachTheoThangNam</p> <p>Mã lệnh PL/SQL:</p> <pre>create or replace function fn_ChiPhiNhapSachTheoThangNam(f_month_in number, f_year_in number) return number is v_total number; begin select sum(TongTien) into v_total from PHIEUNHAP where extract(Month from NGAYNHAP) = f_month_in and extract(Year from NGAYNHAP) = f_Year_in; if (v_total is null) then v_total := 0; end if; return v_total; end;</pre>

Mô tả mã lệnh:

Function trên có tên là fn_ChiPhiNhapSachTheoThangNam, nhiệm vụ của nó là trả về tổng chi phí nhập sách trong tháng và năm được truyền vào.

Trong thân function, các hành động được thực hiện bao gồm:

- Khai báo biến v_total kiểu dữ liệu là number để lưu tổng giá trị của trường TongTien trên bảng PHIEUNHAP.
- Sử dụng câu lệnh select sum(TongTien) into v_total from PHIEUNHAP để tính tổng giá trị của trường TongTien trên bảng PHIEUNHAP.
- Sử dụng hàm extract để lọc các phiếu nhập sách trong tháng và năm được truyền vào dựa trên trường NGAYNHAP của bảng PHIEUNHAP.
- Kiểm tra nếu giá trị của biến v_total là null thì gán giá trị bằng 0.
- Trả về giá trị của biến v_total, tức là tổng chi phí nhập sách trong tháng và năm được truyền vào.
- Function này không thực hiện bất kỳ thao tác nào trên cơ sở dữ liệu, vì vậy không cần thực hiện commit.

PHẦN III. XỬ LÝ ĐỒNG THỜI

1. Tổng quan

a. Giao tác trong Oracle

Một giao tác là một đơn vị thao tác luân lý bao gồm một hoặc nhiều câu lệnh SQL, được thực thi bởi một người dùng đơn. Trong Oracle, một giao tác bắt đầu bằng việc thực thi câu lệnh SQL đầu tiên của người dùng. Và kết thúc khi một trong những điều sau xảy ra:

- Người dùng sử dụng lệnh COMMIT hoặc ROLLBACK mà không một điểm đánh dấu SAVEPOINT.
- Người dùng chạy một câu lệnh DDL chẳng hạn như CREATE, DROP, RENAME, hay ALTER.
- Người dùng ngắt kết nối với Oracle. Giao tác hiện tại sẽ được commit.
- Xử lý của người dùng bị ngắt một cách bất thường. Giao tác hiện tại sẽ bị rollback.

Sau khi một giao tác kết thúc, giao tác tiếp theo sẽ bắt đầu với câu lệnh SQL kế tiếp. Các câu lệnh kiểm soát giao tác:

- Ghi nhận vĩnh viễn những thay đổi được thực hiện trong giao tác (COMMIT).
- Quay ngược lại những thay đổi của giao tác, tính từ lúc giao tác bắt đầu hoặc từ một điểm savepoint (ROLLBACK).
- Đặt một điểm mà có thể rollback (SAVEPOINT)
- Thiết đặt thuộc tính cho giao tác (SET TRANSACTION)
 - SET TRANSACTION ISOLATION LEVEL {READ COMMITTED | SERIALIZABLE};
 - SET TRANSACTION READ {ONLY|WRITE};
- Điều chỉnh khi nào Oracle tiến hành commit (SET AUTOCOMMIT)
 - SET AUTOCOMMIT ON: commit ngay những thay đổi xuống cơ sở dữ liệu sau khi Oracle thực thi thành công các câu lệnh INSERT, UPDATE hoặc DELETE
 - SET AUTOCOMMIT OFF (mặc định): ngăn không để Oracle commit tự động, người dùng phải commit thủ công.

b. Vấn đề trong môi trường truy xuất đồng thời

Mất dữ liệu cập nhật (Lost Update): Tình trạng này xảy ra khi có nhiều hơn một giao tác cùng thực hiện cập nhật trên 1 đơn vị dữ liệu. Khi đó, tác dụng của giao tác cập nhật thực hiện sau sẽ đè lên tác dụng của thao tác cập nhật trước.

Đọc dữ liệu chưa commit (Uncommitted data, Dirty read): Xảy ra khi một giao tác thực hiện đọc trên một đơn vị dữ liệu mà đơn vị dữ liệu này đang bị cập nhật bởi một giao tác khác nhưng việc cập nhật chưa được xác nhận.

Giao tác đọc không thể lặp lại (Non-repeatable read): Tình trạng này xảy ra khi một giao tác T1 vừa thực hiện xong thao tác đọc trên một đơn vị dữ liệu (nhưng chưa commit) thì giao tác khác (t2) lại thay đổi (ghi) trên đơn vị dữ liệu này. Điều này làm cho lần đọc sau đó của T1 không còn nhìn thấy dữ liệu ban đầu nữa.

Bóng ma (Phantom read): Là tình trạng mà một giao tác đang thao tác trên một tập dữ liệu nhưng giao tác khác lại chèn thêm các dòng dữ liệu vào tập dữ liệu mà giao tác kia quan tâm.

c. Các phương thức khóa cơ bản

Shared Lock (S):

- Shared Lock \Leftrightarrow Read Lock
- Khi đọc 1 đơn vị dữ liệu, hệ quản trị tự động thiết lập Shared Lock trên đơn vị dữ liệu đó (trừ trường hợp sử dụng No Lock)
- Shared Lock có thể được thiết lập trên 1 bảng, 1 trang, 1 khóa hay trên 1 dòng dữ liệu.
- Nhiều giao tác có thể đồng thời giữ Shared Lock trên cùng 1 đơn vị dữ liệu.
- Không thể thiết lập Exclusive Lock trên đơn vị dữ liệu đang có Shared Lock.
- Shared Lock thường được giải phóng ngay sau khi sử dụng xong dữ liệu được đọc, trừ khi có thiết lập giữ shared lock cho đến hết giao tác.

Exclusive Lock (X):

- Exclusive Lock \Leftrightarrow Write Lock
- Khi thực hiện thao tác ghi (insert, update, delete) trên 1 đơn vị dữ liệu, hệ quản trị tự động thiết lập Exclusive Lock trên đơn vị dữ liệu đó
- Exclusive Lock luôn được giữ đến hết giao tác.
- Tại 1 thời điểm, chỉ có tối đa 1 giao tác được quyền giữ Exclusive Lock trên 1 đơn vị dữ liệu.
- Không thể thiết lập Exclusive Lock trên đơn vị dữ liệu đang có Shared Lock.

d. Mức cô lập

Read uncommitted:

❖ Đặc điểm:

- Không thiết lập Shared Lock trên những đơn vị dữ liệu cần đọc. Do đó không phải chờ khi đọc dữ liệu (kể cả khi dữ liệu đang bị lock bởi giao tác khác)
- (Vẫn tạo Exclusive Lock trên đơn vị dữ liệu được ghi, Exclusive Lock được giữ cho đến hết giao tác)

❖ Ưu điểm

- Tốc độ xử lý nhanh
- Không cản trở những giao tác khác thực hiện việc cập nhật dữ liệu

❖ Khuyết điểm

Có khả năng xảy ra mọi vấn đề xử lý đồng thời:

- Dirty Read
- Non-repeatable Read
- Phantom Read
- Lost update

Read Committed:

❖ Đặc điểm

- Đây là mức cài đặt mặc định của Oracle/SQL Server
- Tạo Shared Lock trên đơn vị dữ liệu được đọc, Shared Lock được giải phóng ngay sau khi đọc xong dữ liệu
- Tạo Exclusive Lock trên đơn vị dữ liệu được ghi, Exclusive Lock được giữ cho đến hết giao tác

❖ Ưu điểm

- Giải quyết vấn đề Dirty Reads
- Shared Lock được giải phóng ngay, không cần phải giữ cho đến hết giao tác nên không cần trả nhiều đến thao tác cập nhật của giao tác khác

❖ Khuyết điểm

- Chưa giải quyết được vấn đề Non-repeatable Read, Phantom Read, Lost Update
- Phải chờ nếu đơn vị dữ liệu cần đọc đang được giữ khóa ghi (xlock)

Repeatable Read

❖ Đặc điểm

- Tạo Shared Lock trên đơn vị dữ liệu được đọc và giữ shared lock này đến hết giao tác → các giao tác khác phải chờ đến khi giao tác này kết thúc nếu muốn cập nhật, thay đổi giá trị trên đơn vị dữ liệu này.
- (Repeatable Read = Read Committed + Giải quyết Non-repeatable Read)
- Tạo Exclusive Lock trên đơn vị dữ liệu được ghi, Exclusive Lock được giữ cho đến hết giao tác.

❖ Ưu điểm

- Giải quyết vấn đề Dirty Read và Non-repeatable Read

❖ Khuyết điểm

- Chưa giải quyết được vấn đề Phantom Read, do vẫn cho phép insert những dòng dữ liệu thỏa điều kiện thiết lập shared lock
- Phải chờ nếu đơn vị dữ liệu cần đọc đang được giữ khóa ghi (xlock)

- Shared lock được giữ đến hết giao tác → cản trở việc cập nhật dữ liệu của các giao tác khác

Serializable

❖ Đặc điểm

- Tạo Shared Lock trên đơn vị dữ liệu được đọc và giữ shared lock này đến hết giao tác → Các giao tác khác phải chờ đến khi giao tác này kết thúc nếu muốn cập nhật, thay đổi giá trị trên đơn vị dữ liệu này.
- Không cho phép Insert những dòng dữ liệu thỏa mãn điều kiện thiết lập Shared Lock (sử dụng Key Range Lock) → Serializable = Repeatable Read + Giải quyết Phantom Read
- Tạo Exclusive Lock trên đơn vị dữ liệu được ghi, Exclusive Lock được giữ cho đến hết giao tác.

❖ Ưu điểm

- Giải quyết thêm được vấn đề Phantom Read

❖ Khuyết điểm

- Phải chờ nếu đơn vị dữ liệu cần đọc đang được giữ khóa ghi (xlock)
- Cản trở nhiều đến việc cập nhật dữ liệu của các thao tác khác

Sử dụng các mức cô lập

Trong Oracle, để ngăn ngừa các vấn đề về truy xuất đồng thời, ta xem xét sử dụng các mức cô lập sau:

	Dirty Read	Lost Update	Non-repeatable Read	Phantom Read
Read Uncommitted	Cho phép xảy ra	Cho phép xảy ra	Cho phép xảy ra	Cho phép xảy ra
Read Committed (Oracle)	Không được	Cho phép xảy ra	Cho phép xảy ra	Cho phép xảy ra
Repeatable Read	Không được	Không được	Không được	Cho phép xảy ra
Serializable (Oracle)	Không được	Không được	Không được	Không được

Tất cả thông tin phần Tổng quan trên được trích dẫn từ tài liệu: trường ĐH CNTT, Khoa HTTT, môn học Hệ quản trị cơ sở dữ liệu, Bài thực hành số 4 – Nguyễn Hồ Duy Tri

2. Mô tả kịch bản gây mất nhất quán dữ liệu và giải pháp để ra trong đồ án môn học

a. Trường hợp Lost update

- ❖ Mô tả tình huống: Khi nhân viên bán hàng thực hiện thêm dữ liệu một chi tiết hóa đơn sau khi bán hết tất cả số lượng sách số 1 (22 cuốn), cùng lúc đó một nhân viên kho tiến hành thêm một chi tiết phiếu nhập với cùng một mã sách (Sách số 1) với nhân viên bán hàng, có thể xảy ra mất dữ liệu.
- ❖ Ta hãy xem xét và đánh giá lại lại hai store procedure được cài đặt ở phần II.3.1 sau với việc nhân viên bán hàng gặp sự cố bị delay “sleep(10)” trước khi cập nhật lại số lượng sách trong quá trình thực thi procedure sp_ThemCTHD

sp_ThemCTHD	Sp_ThemCTPN
<pre> create or replace procedure sp_ThemCTHD(v_MaHD_in IN CTHD.MaHD%type, v_MaSach_in IN CTHD.MaSach%type, v_SL_in IN CTHD.SL%type) as v_MaKH KHACHHANG.MAKH%type; v_GiaSach SACH.GIA%type; v_valueCTHD number; v_DTL KHACHHANG.DIEMTICHLUY%type; v_sl number; begin select SL into v_sl from SACH where MASACH = v_MaSach_in; insert into CTHD(MaHD, MaSach, SL) values(v_MaHD_in, v_MaSach_in, v_SL_in); update SACH set SL = v_sl - v_SL_in where MaSach = v_MaSach_in; select Gia into v_GiaSach from SACH where MaSach = v_MaSach_in; v_valueCTHD := v_SL_in * v_GiaSach; update HOADON set ThanhTien = ThanhTien + v_valueCTHD where MAHD = v_MAHD_in; select MAKH into v_MaKH </pre>	<pre> create or replace procedure sp_ThemCTPN(v_MaPN_in IN CTPN.MaPN%type, v_MaSach_in IN CTPN.MaSach%type, v_SL_in IN CTPN.SL%type, v_Gia_in IN CTPN.GIA%type) as v_sl number; v_TongTien number; begin select SL into v_sl from SACH where MaSach = v_MaSach_in; select TongTien into v_TongTien from PHIEUNHAP where MAPN = v_MaPN_in; insert into CTPN(MaPN, MaSach, SL, Gia) values(v_MaPN_in, v_MaSach_in, v_SL_in, v_Gia_in); update PHIEUNHAP set TongTien = v_TongTien + v_SL_in * v_Gia_in where MAPN = v_MaPN_in; update SACH set SL = v_sl + v_SL_in where MaSach = v_MaSach_in; commit; end; </pre>

```

from HOADON
where MAHD = v_MaHD_in;

if (v_MaKH is not null) then
    update KHACHHANG
    set DIEMTICHLUY =
DIEMTICHLUY + v_valueCTHD
    where MAKH = v_MaKH;

select DIEMTICHLUY into v_DTL
from KHACHHANG
where MAKH = v_MaKH;

if (v_DTL >= 2000000) then
    update KHACHHANG
    set LOAIKH = 'VIP'
    where MAKH = v_MaKH;
elseif (v_DTL >= 500000) then
    update KHACHHANG
    set LOAIKH = 'Thân Thiết'
    where MAKH = v_MaKH;
elseif (v_DTL < 500000) then
    update KHACHHANG
    set LOAIKH = 'Thường'
    where MAKH = v_MaKH;
end if;
end if;
sleep(10);
commit;
end;

```

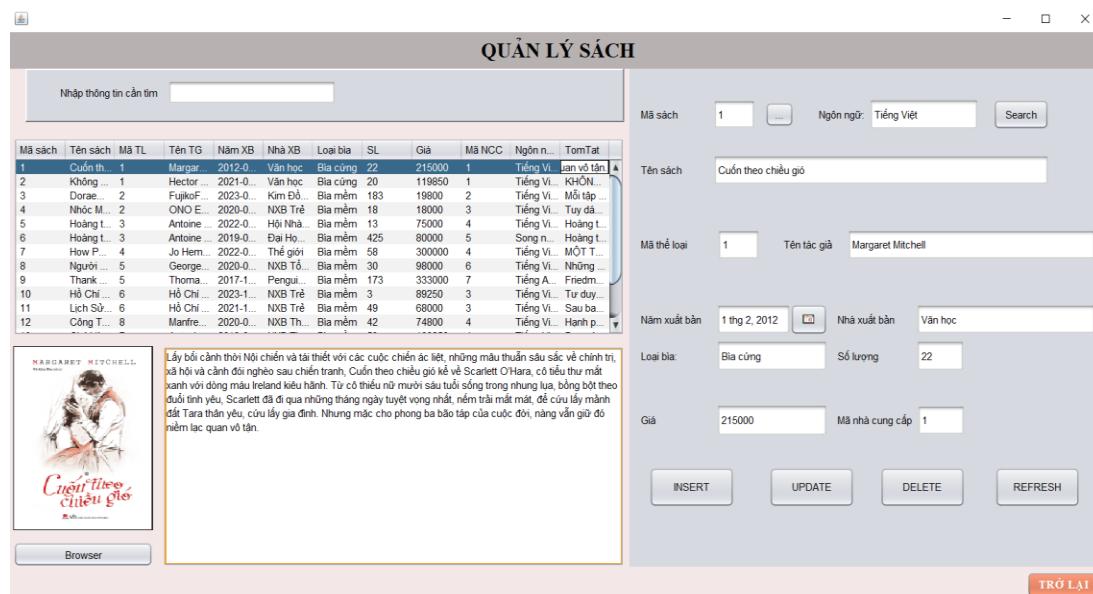
Mô tả tình huống trên hệ quản trị Oracle:

Transaction T1	Transaction T2
(nhân viên bán hàng thêm một chi tiết hóa đơn vào hóa đơn số 369 với mã sách bằng 1)	(Nhân viên kho thêm một chi tiết phiếu nhập vào phiếu nhập số 11 với mã sách bằng 1)
<pre> SQL> Select SL from SACH where MaSach = 1; SL - 22 </pre>	
SET TRANSACTION ISOLATION LEVEL READ COMMITTED;	

Exec sp_ThemCTHD(368, 1, 22); <pre>SQL> SET TRANSACTION ISOLATION LEVEL READ COMMITTED; Transaction set. SQL> Exec sp_ThemCTHD(368, 1, 22); PL/SQL procedure successfully completed.</pre>	
	SET TRANSACTION ISOLATION LEVEL READ COMMITTED; Exec sp_ThemCTPN(11, 1, 100, 10000); <pre>SQL> SET TRANSACTION ISOLATION LEVEL READ COMMITTED; Transaction set. SQL> exec sp_ThemCTPN(11, 1, 100, 10000); PL/SQL procedure successfully completed.</pre>
Select SL from SACH where MaSach = 1; <pre>SQL> select SL from SACH where MASACH = 1; SL ----- 122</pre>	

Mức chương trình:

➊ Màn hình 1: nhân viên bán hàng xem thông tin số lượng sách của sách số 1



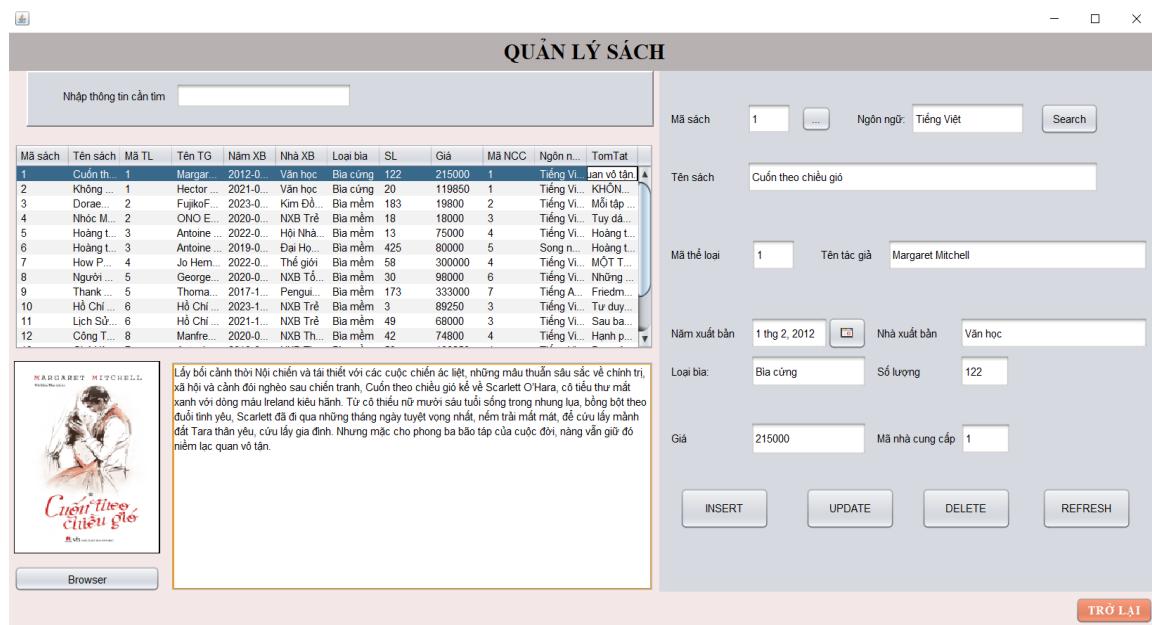
- ⊕ Màn hình 1: Nhân viên bán hàng thêm một chi tiết vào hóa đơn số 368 với mã sách số 1 và số lượng 22 cuốn (bán toàn bộ sách số 1)

Mã hóa đơn	Mã sách	Tên sách	Số lượng	Giá
368	1	Cuốn theo chiều gió	22	215000

- ⊕ Màn hình 2: Nhân viên kho thêm một chi tiết vào phiếu nhập số 11 với mã sách số 1 và số lượng 100 cuốn

Mã phiếu nhập	Mã sách	Tên sách	Số lượng	Giá
11	1	Cuốn theo chiều gió	100	10000

- ⊕ Màn hình 1: nhân viên bán hàng kiểm tra thông tin sách một lần nữa



- ❖ Kết quả hiển thị cả 2 mức đều là 122 cuốn sách, Kết quả đúng phải là 100 cuốn
→ mất dữ liệu
- ❖ Nguyên nhân: Khi transaction T1 thực hiện cập nhật lại số lượng sách sau khi tạo chi tiết hóa đơn nhưng chưa commit dữ liệu cùng lúc đó Transaction T2 cập nhật ghi đè làm mất dữ liệu.

Tình huống 2:

- ❖ Mô tả tình huống 2: Khi nhân viên bán hàng thực hiện thêm dữ liệu một chi tiết hóa đơn với số lượng sách số 1 được bán đi 22 cuốn (sách số 1 chỉ có 22 cuốn), cùng lúc đó một nhân viên bán hàng khác lại tiến hành thêm một chi tiết hóa đơn vẫn là sách số 1 với số lượng là 22 cuốn, có thể xảy ra mất dữ liệu
- ❖ **Mô tả tình huống trên hệ quản trị Oracle**

Transaction T1 (nhân viên bán hàng 1 thêm một chi tiết hóa đơn vào hóa đơn số 368 với mã sách bằng 1)	Transaction T2 (nhân viên bán hàng 2 thêm một chi tiết hóa đơn vào hóa đơn số 3 với mã sách bằng 1)						
<pre>SQL> SELECT SL FROM SACH WHERE MASACH =1;</pre> <table border="1"> <tr> <td>SL</td> </tr> <tr> <td>-----</td> </tr> <tr> <td>22</td> </tr> </table>	SL	-----	22	<pre>SQL> SELECT SL FROM SACH WHERE MASACH =1;</pre> <table border="1"> <tr> <td>SL</td> </tr> <tr> <td>-----</td> </tr> <tr> <td>22</td> </tr> </table>	SL	-----	22
SL							

22							
SL							

22							
SET TRANSACTION ISOLATION LEVEL READ COMMITTED;							

Exec sp_ThemCTHD(368, 1, 22);

```
SQL> SET TRANSACTION ISOLATION LEVEL READ COMMITTED;
Transaction set.

SQL> Exec sp_ThemCTHD(368, 1, 22);
PL/SQL procedure successfully completed.
```

SET TRANSACTION ISOLATION LEVEL
READ COMMITTED;
Exec sp_ThemCTHD(3, 1, 22);

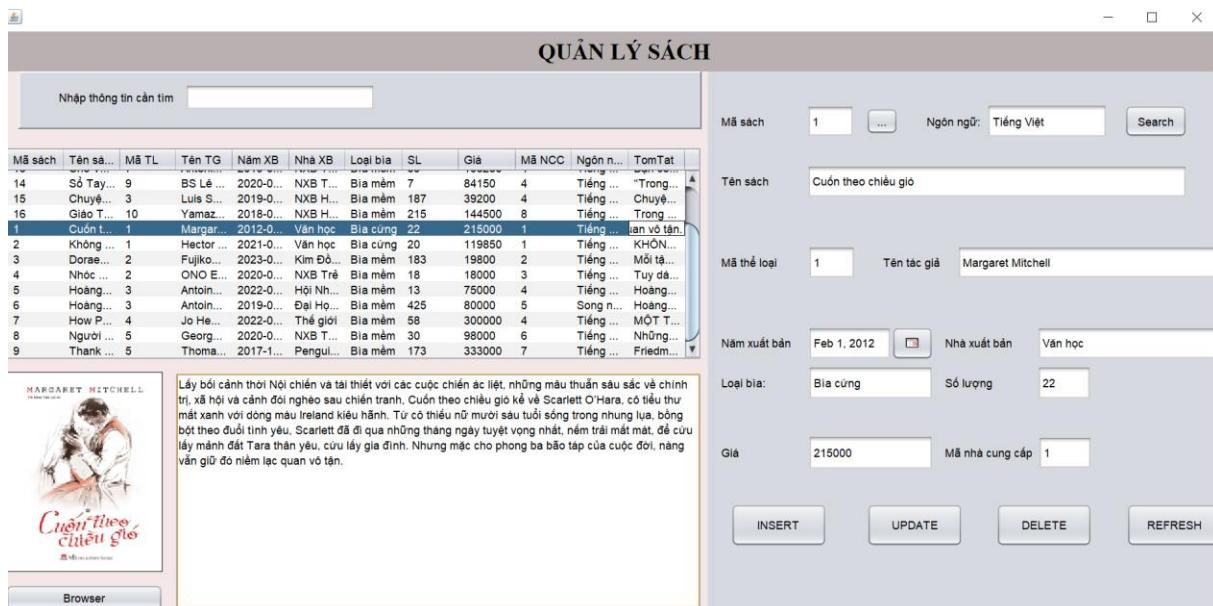
```
SQL> exec sp_ThemCTHD(3,1,22);
PL/SQL procedure successfully completed.
```

Select SL from SACH where MaSach = 1;

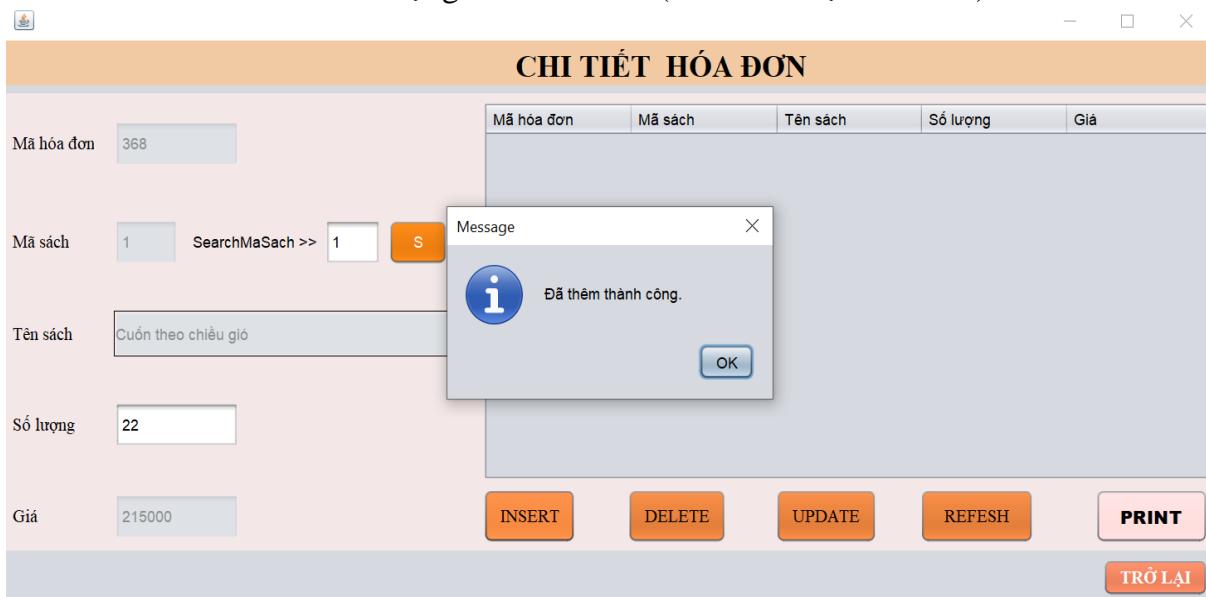
```
SQL> SELECT SL FROM SACH WHERE MASACH =1;
SL
-----
0
```

Mức chương trình

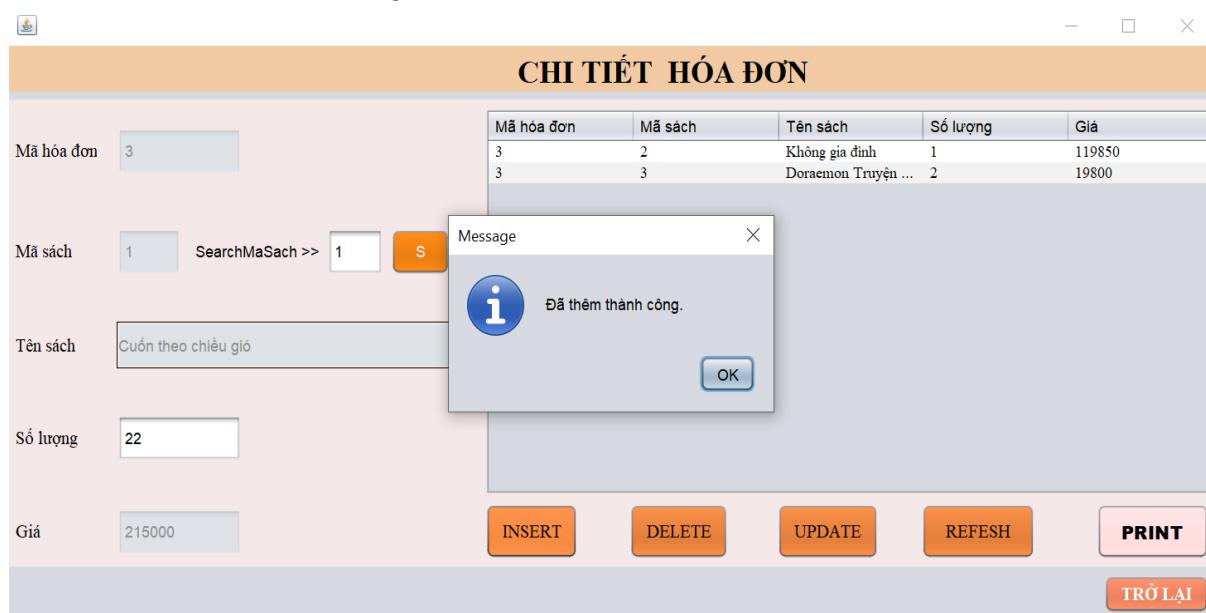
⊕ Màn hình 1 và 2: cả 2 nhân viên đều xem thông tin số lượng sách của sách số 1



- ✚ Màn hình 1: Nhân viên bán hàng 1 thêm một chi tiết hóa đơn vào hóa đơn số 368 với mã sách số 1 và số lượng 22 cuốn sách (bán toàn bộ sách số 1)



- ✚ Màn hình 2: Nhân viên bán hàng 2 thêm một chi tiết hóa đơn vào hóa đơn số 3 với mã sách số 1 và số lượng 22 cuốn sách (bán toàn bộ sách số 1)



- ✚ Màn hình 1 và 2: Kiểm tra thông tin sách một lần nữa

DANH SÁCH TẤT CẢ SÁCH

Mã sách	Tên sách	Mã thẻ loại	Tên tác giả	Năm xuất ...	Nhà xuất b...	Loại bìa	Số lượng	Giá	Mã NCC	Ngôn ngữ	Tóm tắt
10	Hồ Chí Mi...	6	Hồ Chí Minh	2023-10-06	NXB Trẻ	Bìa mềm	3	89250	3	Tiếng Việt	Tư duy qu...
11	Lịch Sử N...	6	Hồ Chí Minh	2021-10-04	NXB Trẻ	Bìa mềm	49	68000	3	Tiếng Việt	Sau bao n...
12	Công Thức...	8	Manfred F ...	2020-08-04	NXB Thế ...	Bìa mềm	42	74800	4	Tiếng Việt	Hạnh phu...
13	Chó Vàng M...	7	Antonio Fi...	2019-05-04	NXB Thế ...	Bìa mềm	59	106250	4	Tiếng Việt	Bản cõi n...
14	Sở Tay Án ...	9	BS Lê Thị ...	2020-05-10	NXB Thế ...	Bìa mềm	7	84150	4	Tiếng Việt	"Trong qu...
15	Chuyện C...c...	3	Luis Sepúl...	2019-04-10	NXB Hội N...	Bìa mềm	187	39200	4	Tiếng Việt	Chuyện c...
16	Giáo Trình...	10	Yamazaki ...	2018-04-10	NXB Hồng...	Bìa mềm	215	144500	8	Tiếng Việt	Trong quâ...
1	Cuốn theo...c...	1	Margaret ...	2012-02-01	Văn học	Bìa cứng	0	215000	1	Tiếng Việt	c quan vô tận.
2	Không gian ...	1	Hector Malot	2021-03-02	Văn học	Bìa cứng	20	119850	1	Tiếng Việt	KHÔNG G...
3	Doraemon...	2	Fujiko Fujio	2023-02-05	Kim Đồng	Bìa mềm	183	19800	2	Tiếng Việt	Mỗi tập tr...
4	Nhóc Mikko...	2	ONO Eriko	2020-05-05	NXB Trẻ	Bìa mềm	18	18000	3	Tiếng Việt	Tuy dẩn ...
5	Hoàng tử bé	3	Antoine D...	2022-01-12	Hội Nhà Văn	Bìa mềm	13	75000	4	Tiếng Việt	Hoàng tử ...



Lấy bối cảnh thời Nội chiến và tái thiết với các cuộc chiến ác liệt, những mâu thuẫn sâu sắc về chính trị, xã hội và cảnh đời nghèo sau chiến tranh. Cuốn theo chiều gió kể về Scarlett O'Hara, cô tiểu thư mắt xanh với dòng máu Ireland kiêu hãnh. Từ cô thiếu nữ mười sáu tuổi sống trong nhung lụa, bồng bột theo đuổi tình yêu, Scarlett đã đi qua những tháng ngày tuyệt vọng nhất, ném trái mắt, để cứu lấy mảnh đất Tara thân yêu, cứu lấy gia đình. Nhưng mặc cho phong ba bão táp của cuộc đời, nàng vẫn giữ được niềm lạc quan vô tận.

- ❖ Kết quả hiển thị cả 2 đều bán được sách số 1 với số lượng là 22 cuốn. Kết quả đúng phải là hệ thống sẽ hiện thông báo lỗi => mất dữ liệu
- ❖ Nguyên nhân: Khi transaction T1 thực hiện cập nhật lại số lượng sách sau khi tạo chi tiết hóa đơn nhưng chưa commit dữ liệu cùng lúc đó transaction T2 cập nhật nên ghi đè làm mất dữ liệu

✚ Tình huống 3:

- ❖ Mô tả tình huống 3: Khi nhân viên kho thực hiện thêm dữ liệu một chi tiết phiếu nhập cho mã phiếu nhập 11 với mã sách 1, cùng lúc đó một nhân viên kho khác tiến hành thêm một chi tiết phiếu nhập cho mã phiếu nhập 12 với cùng một mã sách 1, có thể xảy ra mất dữ liệu.

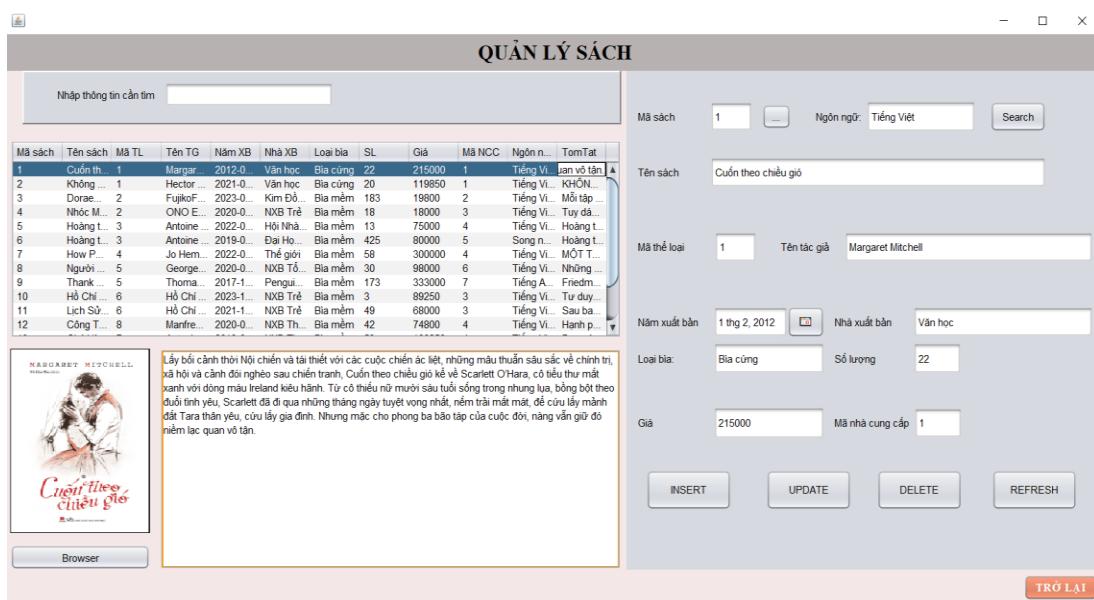
❖ Mô tả tình huống trên hệ quản trị Oracle:

Transaction T1	Transaction T2
(Nhân viên kho thứ nhất thêm một chi tiết phiếu nhập vào phiếu nhập số 11 với mã sách bằng 1)	(Nhân viên kho thứ hai thêm một chi tiết phiếu nhập vào phiếu nhập số 12 với mã sách bằng 1)
<pre>SQL> Select SL from SACH where MaSach = 1;</pre> <pre>SL ----- 22</pre>	

<p>SET TRANSACTION ISOLATION LEVEL READ COMMITTED; Exec sp_ThemCTPN(11, 1, 10, 165385);</p> <pre>SQL> SET TRANSACTION ISOLATION LEVEL READ COMMITTED; Transaction set. SQL> exec sp_ThemCTPN(11, 1, 10, 165385); PL/SQL procedure successfully completed.</pre>	
	<p>SET TRANSACTION ISOLATION LEVEL READ COMMITTED; Exec sp_ThemCTPN(12, 1, 20, 165385);</p> <pre>SQL> SET TRANSACTION ISOLATION LEVEL READ COMMITTED; Transaction set. SQL> exec sp_ThemCTPN(12, 1, 20, 165385); PL/SQL procedure successfully completed.</pre>
<p>Select SL from SACH where MaSach = 1;</p> <pre>SQL> select SL from SACH where MaSach = 1; SL ----- 42</pre>	

❖ Mức chương trình:

⊕ Màn hình 1: nhân viên kho xem thông tin số lượng sách của sách số 1



⊕ Màn hình 1: Nhân viên kho thứ nhất thêm một chi tiết vào phiếu nhập số 11 với mã sách số 1 và số lượng 10 cuốn

The screenshot shows a Windows application window titled "CHI TIẾT PHIẾU NHẬP". On the left, there are input fields for "Mã phiếu nhập" (Purchase Slip ID) set to 11, "Mã sách" (Book ID) set to 1, "Tên sách" (Book Name) set to "Cuốn theo chiều gió", "Số lượng" (Quantity) set to 10, and "Giá" (Price) set to 165385. To the right is a table with columns: Mã phiếu nhập, Mã sách, Tên sách, Số lượng, and Giá. A single row is selected with values: 11, 1, Cuốn theo chiều gió, 10, and 165385. At the bottom, there are four orange buttons: INSERT, DELETE, UPDATE, and REFRESH, along with a red "TRỞ LẠI" (Back) button.

⊕ Màn hình 2: Nhân viên kho thứ hai thêm một chi tiết vào phiếu nhập số 12 với mã sách số 1 và số lượng 20 cuốn

The screenshot shows a Windows application window titled "CHI TIẾT PHIẾU NHẬP". On the left, there are input fields for "Mã phiếu nhập" (Purchase Slip ID) set to 12, "Mã sách" (Book ID) set to 1, "Tên sách" (Book Name) set to "Cuốn theo chiều gió", "Số lượng" (Quantity) set to 20, and "Giá" (Price) set to 165385. To the right is a table with columns: Mã phiếu nhập, Mã sách, Tên sách, Số lượng, and Giá. A single row is selected with values: 12, 1, Cuốn theo chiều gió, 20, and 165385. At the bottom, there are four orange buttons: INSERT, DELETE, UPDATE, and REFRESH, along with a red "TRỞ LẠI" (Back) button.

⊕ Màn hình 1: Nhân viên kho thứ nhất kiểm tra thông tin sách một lần nữa

DANH SÁCH TẤT CẢ SÁCH

Mã sách	Tên sách	Mã thể loại	Tên tác giả	Năm xuất b...	Nhà xuất bản	Loại bìa	Số lượng	Giá	Mã NCC	Ngôn ngữ	Tóm tắt
1	Cuốn theo ...	1	Margaret Mi...	2012-02-01	Văn học	Bìa cứng	42	215000	1	Tiếng Việt	[c quan vô tận]
2	Không gia đ...	1	Hector Malot	2021-03-02	Văn học	Bìa cứng	20	119850	1	Tiếng Việt	KHÔNG GI...
3	Doraemon ...	2	FujikoFFujio	2023-02-05	Kim Đồng	Bìa mềm	183	19800	2	Tiếng Việt	Mỗi tập truy...
4	Nhóc Mikol ...	2	ONO Eriko	2020-05-05	NXB Trẻ	Bìa mềm	18	18000	3	Tiếng Việt	Tuy dâng n...
5	Hoàng tử bé	3	Antoine De ...	2022-01-12	Hội Nhà Văn	Bìa mềm	13	75000	4	Tiếng Việt	Hoàng tử b...
6	Hoàng tử bé	3	Antoine De ...	2019-04-05	Đại Học Q...	Bìa mềm	425	80000	5	Song ngữ A...	Hoàng tử b...
7	How Psych...	4	Jo Hemmin...	2022-06-05	Thế giới	Bìa mềm	58	300000	4	Tiếng Việt	MỘT TRO...
8	Người Giàu...	5	George S ...	2020-08-12	NXB Tổng ...	Bìa mềm	30	98000	6	Tiếng Việt	Những tran...
9	Thank You f...	5	Thomas L ...	2017-10-16	Penguin Bo...	Bìa mềm	173	333000	7	Tiếng Anh	Friedman r...
10	Hồ Chí Min...	6	Hồ Chí Minh	2023-10-06	NXB Trẻ	Bìa mềm	3	89250	3	Tiếng Việt	Tư duy quâ...
11	Lịch Sử Nữ...	6	Hồ Chí Minh	2021-10-04	NXB Trẻ	Bìa mềm	49	68000	3	Tiếng Việt	Sau bao nă...
12	Công Thức...	8	Manfred F ...	2020-08-04	NXB Thế G...	Bìa mềm	42	74800	4	Tiếng Việt	Hạnh phúc ...

Lấy bối cảnh thời Nội chiến và tái thiết với các cuộc chiến ác liệt, những mâu thuẫn sâu sắc về chính trị, xã hội và cảnh đời nghèo sau chiến tranh, Cuốn theo chiều gió kể về Scarlett O'Hara, cô tiểu thư mắt xanh với dòng máu Ireland kiêu hãnh. Từ cô thiếu nữ mười sáu tuổi sống trong nhung lụa, bồng bột theo đuổi tình yêu, Scarlett đã đi qua những tháng ngày tuyệt vọng nhất, nếm trải mất mát, để cứu lấy mảnh đất Tara thân yêu, cứu lấy gia đình. Nhưng mặc cho phong ba bão táp của cuộc đời, nàng vẫn giữ được niềm lạc quan vĩnh cửu.



- ❖ Kết quả hiển thị cả 2 mức đều là 42 cuốn sách, Kết quả đúng phải là 52 cuốn
→ mất dữ liệu
- ❖ Nguyên nhân: Khi transaction T1 thực hiện cập nhật lại số lượng sách sau khi tạo chi tiết phiếu nhập nhưng chưa commit dữ liệu cùng lúc đó Transaction T2 cập nhật ghi đè làm mất dữ liệu.
- ❖ Giải pháp ngăn chặn:
→ Cách 1:

Đối với stored procedure sp_ThemCTHD:

- Vấn đề: Trong trường hợp có nhiều giao dịch cùng cập nhật dữ liệu trên bảng **SACH** và **HOADON**, có xuất hiện tình huống lost update.
- Giải pháp:
 - Sử dụng cơ chế khóa dòng (row-level locking) để đảm bảo rằng sẽ chỉ có một giao dịch truy xuất và cập nhật dữ liệu của mỗi dòng cụ thể. Bằng cách đó, ta sẽ tránh được việc bị lost update.
 - Sử dụng hint **FOR UPDATE** để khóa dòng của bảng **SACH** trước khi thực hiện câu lệnh cập nhật và khóa bảng **HOADON** bằng cách thêm giá trị **FOR UPDATE** vào vé WHERE của câu lệnh **SELECT**. Chi tiết cụ thể:

- Trong câu lệnh **SELECT SL INTO v_sl FROM SACH WHERE MaSach = v_MaSach_in FOR UPDATE**, ta sử dụng hint **FOR UPDATE** để khóa dòng của bảng **SACH**.
- Trong câu lệnh **SELECT Gia INTO v_GiaSach FROM SACH WHERE MaSach = v_MaSach_in FOR UPDATE**, ta sử dụng hint **FOR UPDATE** để khóa dòng của bảng **SACH**.
- Tăng mức độ cô lập của giao dịch để phòng tránh các vấn đề liên quan đến đọc giá trị dirty (dirty read), do đó ta sử dụng mức độ đọc committed (read committed).
- Sử dụng lock mode **SHARE** cho giao dịch để khóa bảng và đảm bảo rằng các bảng này không bị thay đổi bởi bất kỳ giao dịch khác trong quá trình cập nhật.

Đối với stored procedure **sp_ThemCTPN**:

- ❖ Vấn đề: Trong trường hợp có nhiều giao dịch cùng cập nhật dữ liệu trên bảng **SACH** và **PHIEUNHAP**, có xuất hiện tình huống lost update.
- ❖ Giải pháp:
 - Sử dụng cơ chế khóa dòng (row-level locking) để đảm bảo rằng sẽ chỉ có một giao dịch truy xuất và cập nhật dữ liệu của mỗi dòng cụ thể. Bằng cách đó, ta sẽ tránh được việc bị lost update.
 - Sử dụng hint **FOR UPDATE** để khóa dòng của bảng **SACH** và **PHIEUNHAP** trước khi thực hiện câu lệnh update dữ liệu. Chi tiết cụ thể:
 - Trong câu lệnh **SELECT SL INTO v_sl FROM SACH WHERE MaSach = v_MaSach_in FOR UPDATE**, ta sử dụng hint **FOR UPDATE** để khóa dòng của bảng **SACH**.
 - Trong câu lệnh **SELECT TongTien INTO v_TongTien FROM PHIEUNHAP WHERE MAPN = v_MaPN_in FOR UPDATE**, ta sử dụng hint **FOR UPDATE** để khóa dòng của bảng **PHIEUNHAP**.
 - Tăng mức độ cô lập của giao dịch để phòng tránh các vấn đề liên quan đến đọc giá trị dirty (dirty read), do đó ta sử dụng mức độ đọc committed (read committed).
 - Sử dụng lock mode **SHARE** cho giao dịch để khóa bảng và đảm bảo rằng các bảng này không bị thay đổi bởi bất kỳ giao dịch khác trong quá trình cập nhật.

Các công việc xử lý dữ liệu được thực hiện ở mức độ stored procedure của DBMS

Lưu ý rằng, việc sử dụng hint **FOR UPDATE** sẽ làm tăng tải cho database và giảm hiệu suất của hệ thống nếu được sử dụng quá nhiều.

➔ **Cách 2:** Thay mức cõ lập mặc định (Read committed) thành Serializable để giải quyết tất cả vấn đề Lost update

b. Trường hợp Dirty Read

Trong hệ quản trị cơ sở dữ liệu Oracle mức cõ lập mặc định là Read Committed nên không bao giờ xảy ra trường hợp Dirty Read

c. Trường hợp Non-repeatable Read

- ❖ Mô tả tình huống: Khi một nhân viên quản lý thực hiện kiểm tra số lượng sách của một mã sách trong hệ thống nhưng chưa commit, cùng lúc đó một nhân viên bán hàng lại thực hiện xóa một hóa đơn có chi tiết hóa đơn đã bán mã sách đó và hệ thống cập nhật lại số lượng sách trong kho. Nhân viên quản lý kiểm tra số lượng sách một lần nữa thì không còn thấy dữ liệu ban đầu nữa.
- ❖ Nguyên nhân: Do trong khi nhân viên quản lý thực hiện tra cứu lần đầu chưa commit mà nhân viên bán hàng đã thực hiện xóa hóa đơn làm cho hệ thống tự động cập nhật lại số lượng sách của cách mã sách có trong hóa đơn thông qua trigger, trong đó có mã hóa đơn mà nhân viên quản lý đang kiểm tra, sau khi nhân viên bán hàng commit dữ liệu thì nhân viên quản lý lại tra cứu số lượng sách mã sách đó một lần nữa, dữ liệu lúc này đã bị ghi lại và không còn thấy dữ liệu cũ nữa.
- ❖ Với store procedure sp_XoaHoaDon:

```
create or replace procedure sp_XoaHoaDon(
    v_MaHD_in HOADON.MAHD%type
)
as
begin
    sleep(10);
    delete from CTHD where MaHD = v_MaHD_in;
    delete from HOADON where MaHD = v_MaHD_in;
    commit;
end;
```

➔ Ta có mô tả quá trình sau:

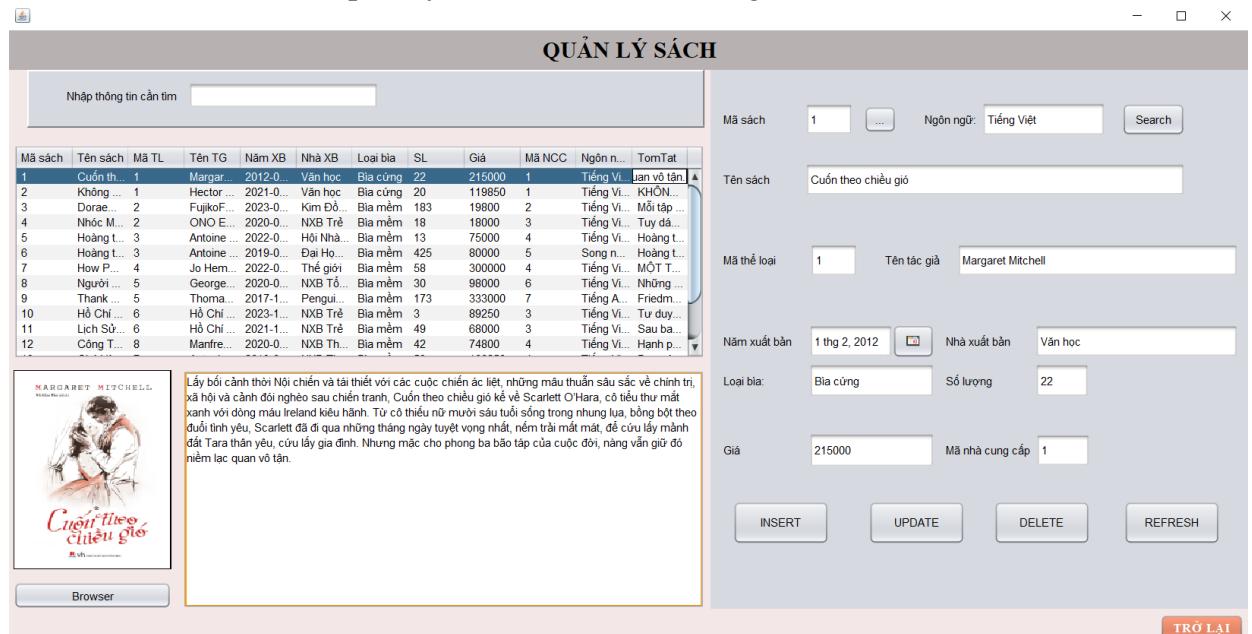
Mô tả tình huống trên hệ quản trị Oracle:

T1 (Nhân viên quản lý thực hiện kiểm tra số lượng sách của sách số 1)	T2 (Nhân viên bán hàng thực hiện xóa hóa đơn số 1 có chứa CTHD chứa mã sách số 1)
SET TRANSACTION ISOLATION LEVEL READ COMMITTED; Select SL from SACH where MaSach = 1;	

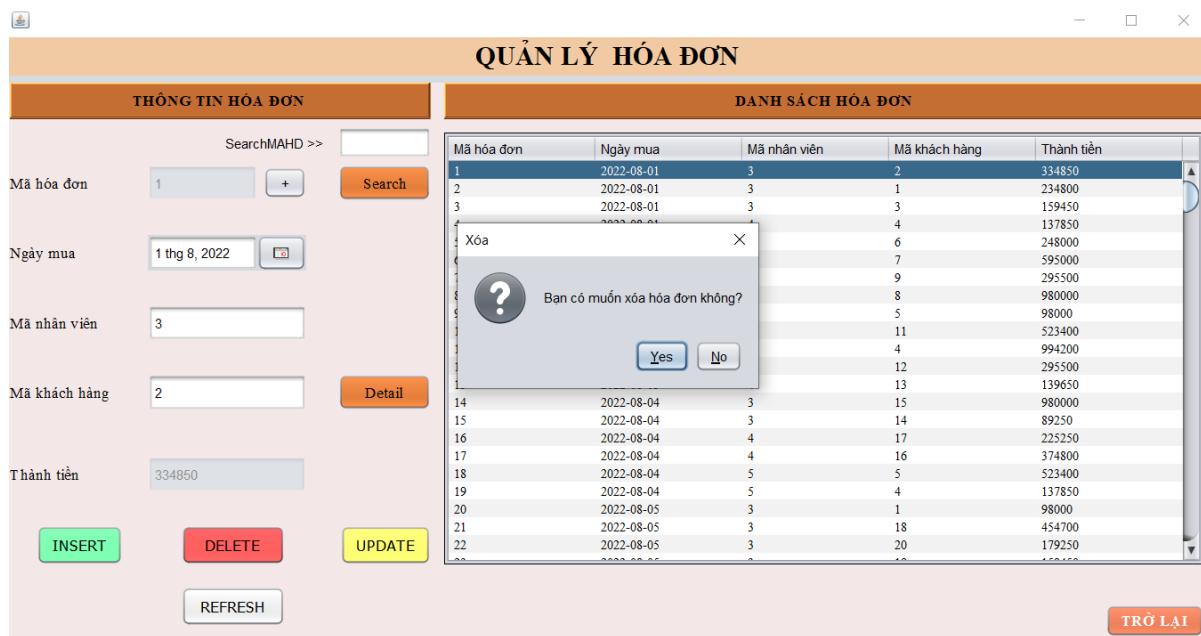
<pre>SQL> select SL from SACH where MASACH = 1;</pre> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="text-align: center;">SL</th></tr> </thead> <tbody> <tr> <td style="text-align: center;">22</td></tr> </tbody> </table>	SL	22	
SL			
22			
	<p style="color: #C00000;">SET TRANSACTION ISOLATION LEVEL READ COMMITTED; EXEC sp_XoaHoaDon(1);</p> <pre>SQL> exec sp_XoaHoaDon(1);</pre> <p>PL/SQL procedure successfully completed.</p>		
<p style="color: #C00000;">Select SL from SACH where MaSach = 1;</p> <pre>SQL> select SL from SACH where MASACH = 1;</pre> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="text-align: center;">SL</th></tr> </thead> <tbody> <tr> <td style="text-align: center;">23</td></tr> </tbody> </table>	SL	23	
SL			
23			

Mô tả tình huống trên chương trình

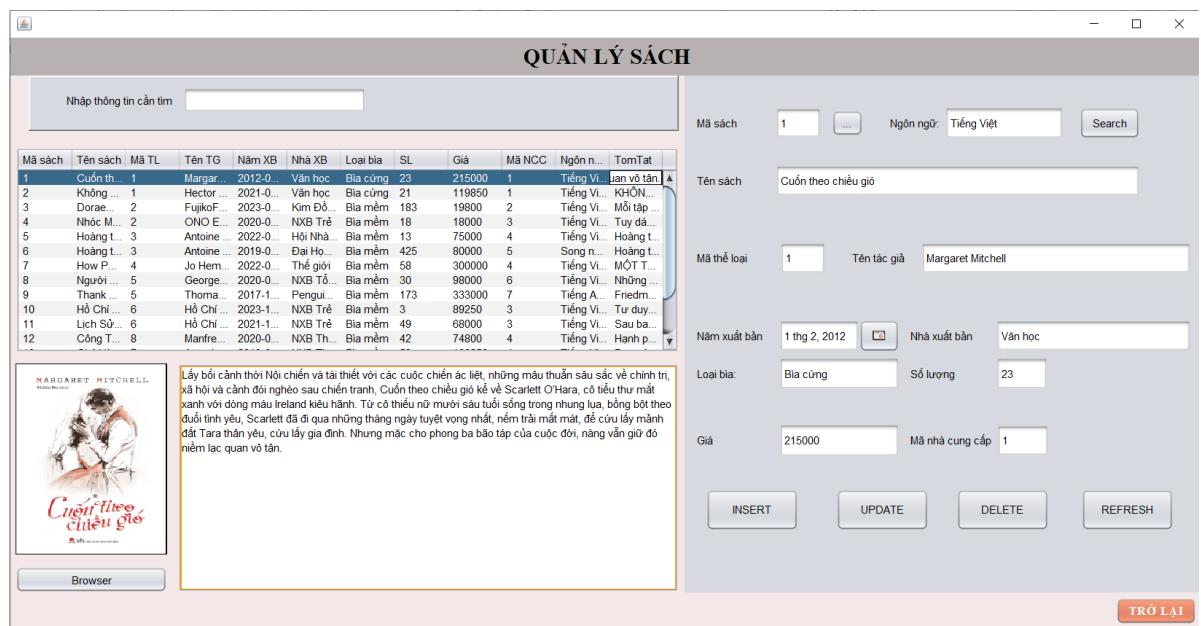
- Màn hình 1: Nhân viên quản lý kiểm tra xem số lượng còn lại của sách số 1



- Màn hình 2: Nhân viên bán hàng cùng lúc đó thực hiện xóa hóa đơn số 1 có chưa chi tiết bán sách số 1



- ✚ Màn hình 1: Nhân viên quản lý thực hiện refresh xem lại lần nữa thì thấy số lượng sách số 1 tăng lên 23



- ✚ Tình huống 2:

- ❖ Mô tả tình huống 2: Khi nhân viên quản lý thực hiện kiểm tra số lượng sách của một mã sách trong hệ thống nhưng chưa commit, cùng lúc đó một nhân viên kho lại thực hiện thêm một phiếu nhập có chi tiết phiếu nhập là số lượng mã sách đó và hệ thống cập nhật lại số lượng sách trong kho. Nhân

viên quản lý kiểm tra số lượng sách một lần nữa thì không còn thấy dữ liệu ban đầu

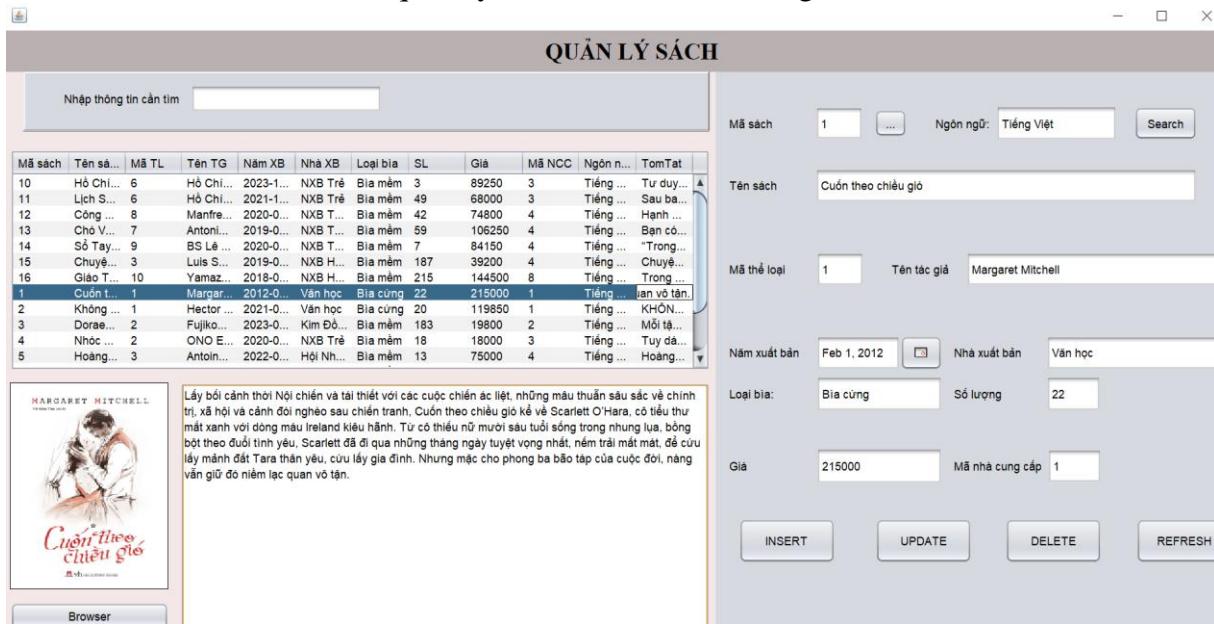
- ❖ Nguyên nhân: Do trong khi nhân viên quản lý thực hiện tra cứu lần đầu chưa commit mà nhân viên kho đã thực hiện thêm chi tiết phiếu nhập làm cho hệ thống tự động cập nhật lại số lượng sách của mã sách có trong chi tiết phiếu nhập, sau khi nhân viên kho commit dữ liệu thì nhân viên quản lý tra cứu số lượng sách của mã sách đó một lần nữa, dữ liệu lúc này đã bị ghi lại và không còn thấy dữ liệu cũ nữa

Mô tả tình huống trên hệ quản trị Oracle:

<p>T1 (Nhân viên quản lý thực hiện kiểm tra số lượng sách của sách số 1)</p> <pre style="background-color: black; color: white; padding: 10px;">SET TRANSACTION ISOLATION LEVEL READ COMMITTED; Select SL from SACH where MaSach = 1; SQL> select SL from SACH where MASACH = 1; SL ----- 22</pre>	<p>T2 (Nhân viên kho thực hiện thêm một chi tiết phiếu nhập có số lượng của mã sách số 1)</p> <pre style="background-color: black; color: white; padding: 10px;">SET TRANSACTION ISOLATION LEVEL READ COMMITTED; EXEC SQL> exec sp_ThemCTPN(11,1,100,10000); PL/SQL procedure successfully completed. sp_ThemCTPN(11,1,100,10000) ;</pre>
<pre style="background-color: black; color: white; padding: 10px;">Select SL from SACH where MaSach = 1; SQL> select sl from sach where masach=1; SL ----- 122</pre>	

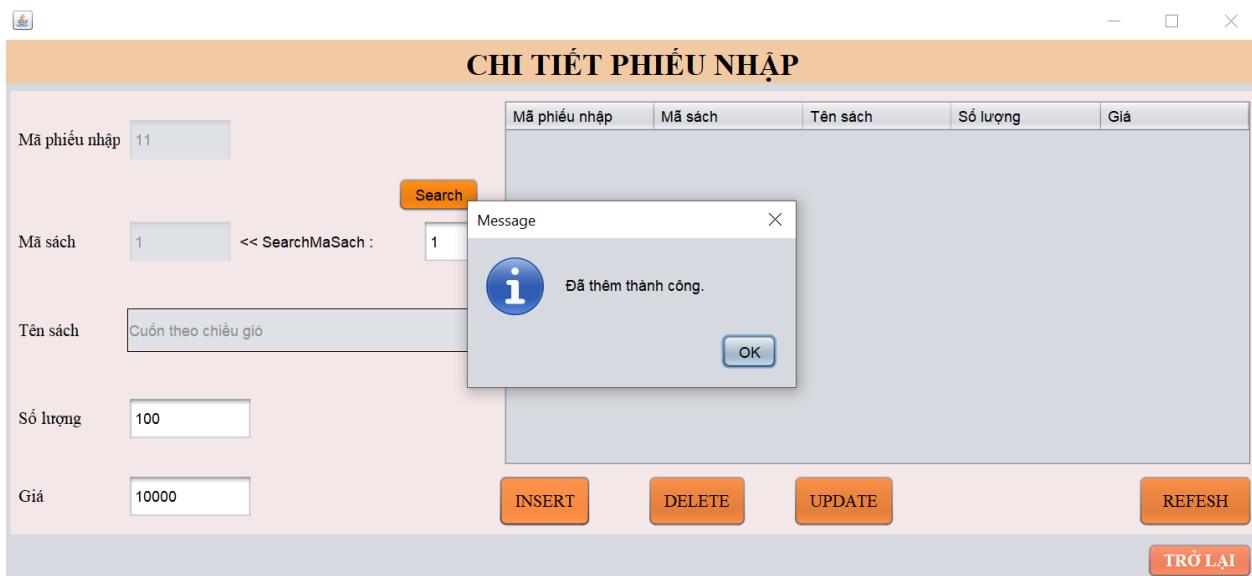
Mô tả tình huống trên chương trình

 Màn hình 1: nhân viên quản lý kiểm tra xem số lượng của sách số 1



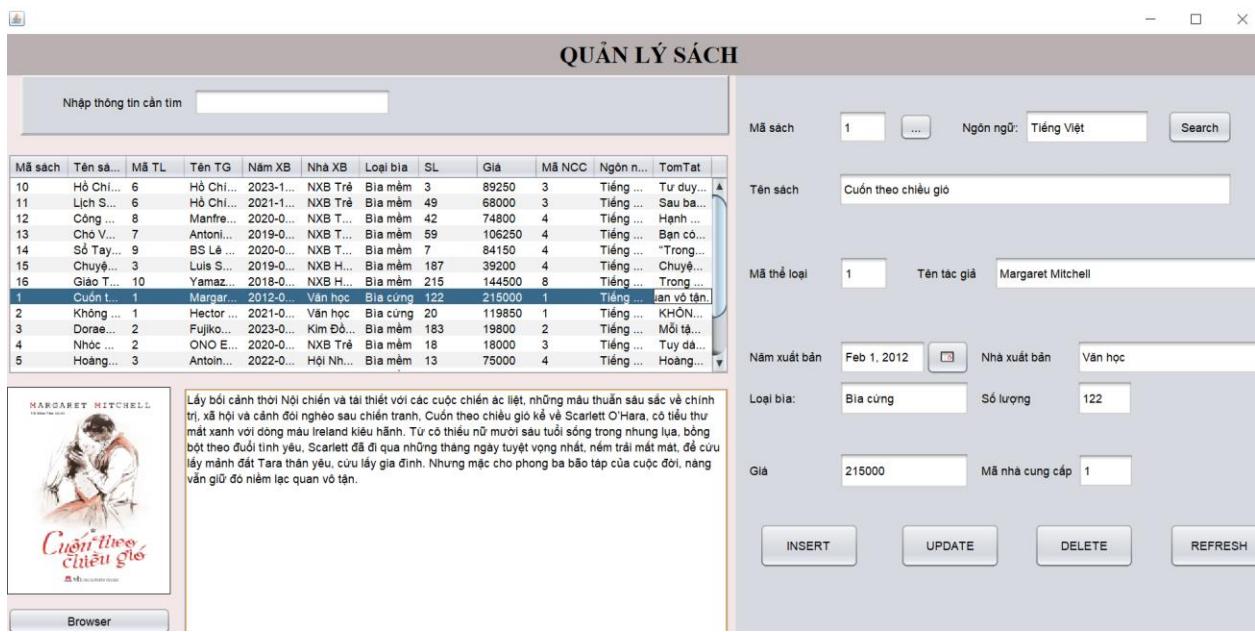
The screenshot shows a Windows application window titled "QUẢN LÝ SÁCH". On the left is a grid table with columns: Mã sách, Tên sá..., Mã TL, Tên TG, Năm XB, Nhà XB, Loại bìa, SL, Giá, Mã NCC, Ngôn n..., TomTat. The table contains 16 rows of book data. The 17th row, which corresponds to the selected book, has a blue background. To the right of the table are search and filter fields: Mã sách (1), Ngôn ngữ (Tiếng Việt), Tên sách (Cuốn theo chiều gió), Mã thể loại (1), Tên tác giả (Margaret Mitchell), Năm xuất bản (Feb 1, 2012), Nhà xuất bản (Văn học), Loại bìa (Bìa cứng), Số lượng (22), Giá (215000), and Mã nhà cung cấp (1). At the bottom are buttons for INSERT, UPDATE, DELETE, and REFRESH.

 Màn hình 2: nhân viên khi thực hiện thêm chi tiết phiếu nhập có số lượng của mã sách số 1 là 100



The screenshot shows a Windows application window titled "CHI TIẾT PHIẾU NHẬP". It has input fields for: Mã phiếu nhập (11), Mã sách (1), Tên sách (Cuốn theo chiều gió), Số lượng (100), and Giá (10000). Below these fields are buttons for INSERT, DELETE, UPDATE, REFRESH, and TRỞ LẠI. A modal dialog box titled "Message" with an info icon displays the message "Đã thêm thành công." (Added successfully). The background table header is visible: Mã phiếu nhập, Mã sách, Tên sách, Số lượng, Giá.

 Màn hình 1: nhân viên quản lý thực hiện refresh xem lại lần nữa thì thấy số lượng sách số 1 tăng lên 122



Tình huống 3:

- ❖ Mô tả tình huống 3: Khi nhân viên quản lý thực hiện kiểm tra số lượng sách của một mã sách trong hệ thống nhưng chưa commit, cùng lúc đó một nhân viên bán hàng lại thực hiện thêm một chi tiết hóa đơn có chi tiết hóa đơn là số lượng mã sách đó và hệ thống cập nhật lại số lượng sách trong kho. Nhân viên quản lý kiểm tra số lượng sách một lần nữa thì không còn thấy dữ liệu ban đầu
- ❖ Nguyên nhân: Do trong khi nhân viên quản lý thực hiện tra cứu lần đầu chưa commit mà nhân viên bán hàng đã thực hiện thêm chi tiết hóa đơn làm cho hệ thống tự động cập nhật lại số lượng sách của mã sách có trong chi tiết hóa đơn, sau khi nhân viên bán hàng commit dữ liệu thì nhân viên quản lý tra cứu số lượng sách của mã sách đó một lần nữa, dữ liệu lúc này đã bị ghi lại và không còn thấy dữ liệu cũ nữa

Mô tả tình huống trên hệ quản trị Oracle:

T1 (Nhân viên quản lý thực hiện kiểm tra số lượng sách của sách số 2)	T2 (Nhân viên bán hàng thực hiện thêm một chi tiết hóa đơn có số lượng của mã sách số 2)
SET TRANSACTION ISOLATION LEVEL READ COMMITTED;	

Select SL from SACH where MaSach = 2;

```
SQL> select sl from sach where masach=2;

SL
-----
20
```

SET TRANSACTION ISOLATION LEVEL READ COMMITTED;
EXEC sp_ThemCTHD(368,2,10);

```
SQL> exec sp_ThemCTHD(368,2,10);
```

```
PL/SQL procedure successfully completed.
```

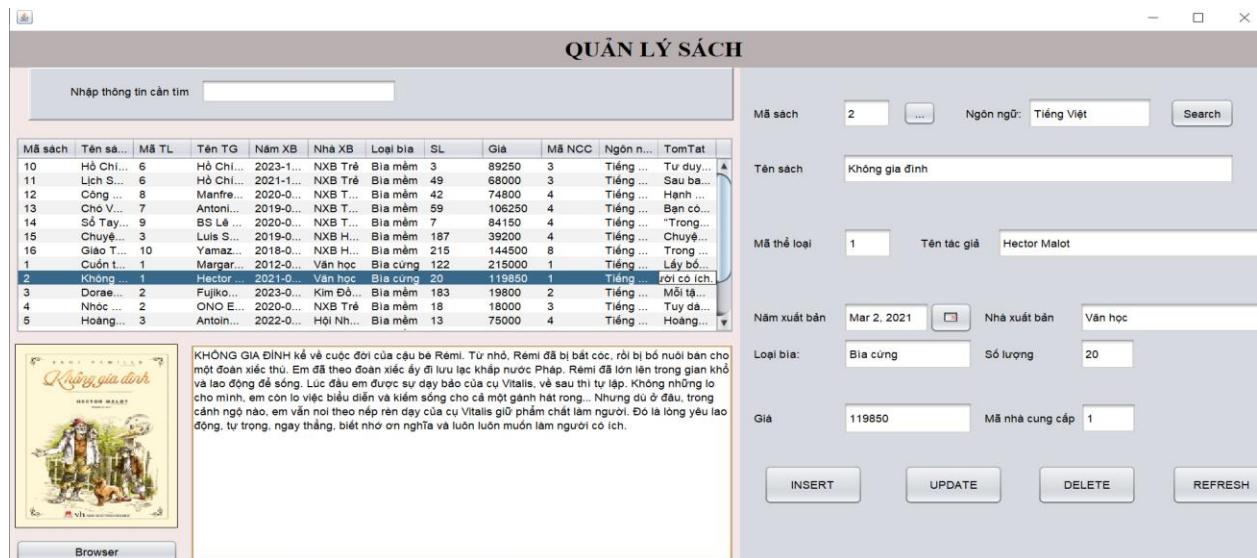
Select SL from SACH where MaSach = 2;

```
SQL> select sl from sach where masach=2;

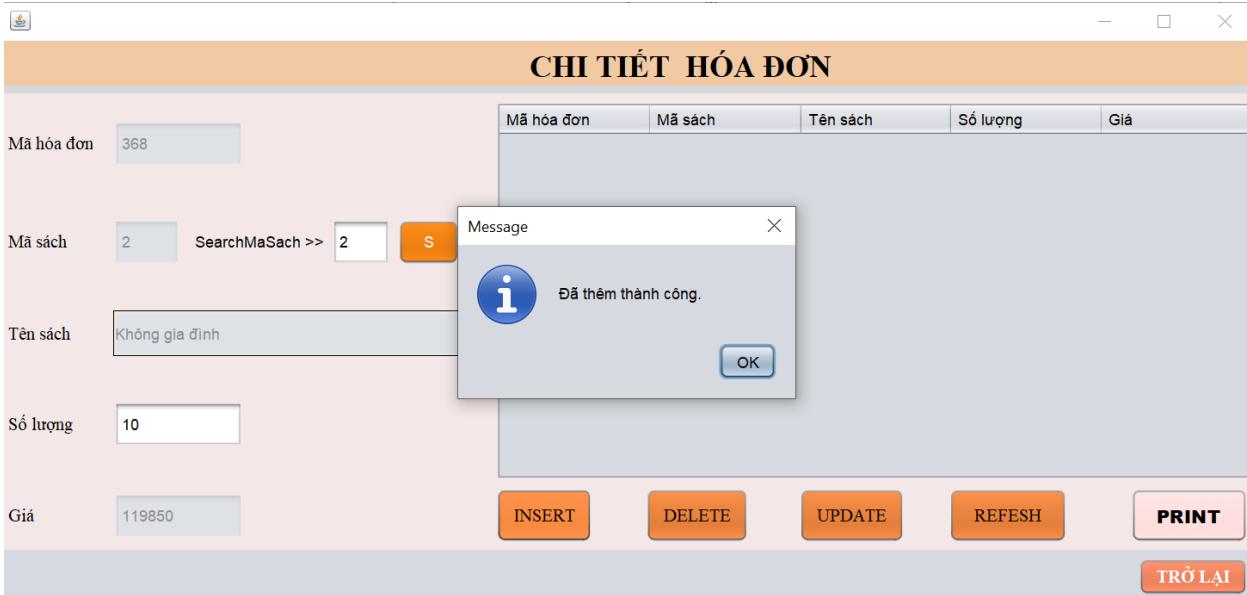
SL
-----
10
```

Mô tả trên hình huống trên chương trình:

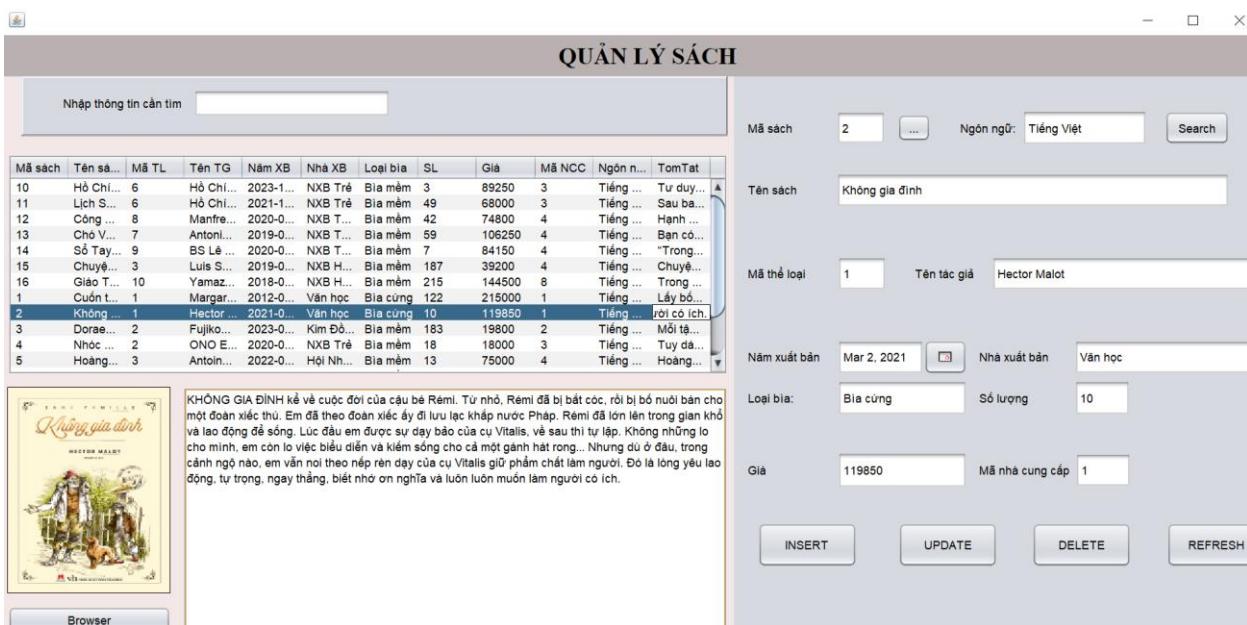
- Màn hình 1: Nhân viên quản lý kiểm tra xem số lượng của sách số 2



- Màn hình 2: Nhân viên bán hàng thêm vào chi tiết hóa đơn của hóa đơn mã 368 có số lượng của sách số 2 là 10



- Màn hình 1: Nhân viên quản lý thực hiện refresh xem lại lần nữa thì thấy số lượng sách số 2 giảm



- ❖ Giải pháp ngăn chặn: Thay mức cõi lập mặc định (Read committed) thành Serializable để giải quyết tất cả vấn đề Non-repeatable Read cho T1.

d. Trường hợp Phantom Read

- ❖ Mô tả tình huống: Khi nhân viên quản lý đang kiểm tra chi tiết hóa đơn của hóa đơn số 1, cùng lúc đó nhân viên bán hàng thực hiện thêm một record chi tiết hóa đơn mới trong hóa đơn số 1. Kết quả khi nhân viên quản lý kiểm tra lại lần nữa thì thấy số lượng CTHD tăng lên 1 dòng.
- ❖ Nguyên nhân: Do trong khi nhân viên quản lý thực hiện tra cứu lần đầu chưa commit mà nhân viên bán hàng đã thực hiện thêm một chi tiết hóa đơn làm cho số lượng chi tiết trong hóa đơn tăng lên, sau khi nhân viên bán hàng commit dữ liệu thì nhân viên quản lý lại xem chi tiết trong mã hóa đơn đó một lần nữa, dữ liệu lúc này đã bị thêm vào nhiều hơn dòng dữ liệu cũ.
- ❖ Ta có mô tả quá trình xảy ra vấn đề phantom read như sau:

<p>T1 (Nhân viên quản lý thực hiện kiểm tra (xem) chi tiết trong hóa đơn số 1)</p> <pre>SET TRANSACTION ISOLATION LEVEL READ COMMITTED; Select * from CTHD where MAHD = 1; SQL> SET TRANSACTION ISOLATION LEVEL READ COMMITTED; Transaction set. SQL> select * from CTHD where MAHD = 1; MAHD MASACH SL ----- ----- 1 1 1 1 2 1</pre>	<p>T2 (Nhân viên bán hàng thực hiện thêm một chi tiết mới (insert) vào hóa đơn số 1)</p> <pre>Exec sp_ThemCTHD(1, 5, 1); SQL> exec sp_ThemCTHD(1, 5, 1); PL/SQL procedure successfully completed.</pre>
	<p>Commit;</p>
<p>Select * from CTHD where MAHD = 1; SQL> select * from CTHD where MAHD = 1; MAHD MASACH SL ----- ----- 1 1 1 1 2 1 1 5 1</p>	

Commit;	
---------	--

Mô tả tình huống trên chương trình

- +[+] Màn hình 1: Nhân viên quản lý thực hiện kiểm tra (xem) danh sách chi tiết của hóa đơn số 1

The screenshot shows a Windows application window titled "CHI TIẾT HÓA ĐƠN". On the left, there are input fields for "Mã hóa đơn" (Invoice ID) set to 1, "Mã sách" (Book ID) set to 1, and "Tên sách" (Book Name) set to "Cuốn theo chiều gió". Below these are buttons for "SearchMaSach >>" and "S". To the right is a table listing the details of the invoice:

Mã hóa đơn	Mã sách	Tên sách	Số lượng	Giá
1	1	Cuốn theo chiều gió	1	215000
1	2	Không già định	1	119850

At the bottom of the window are several buttons: "INSERT", "DELETE", "UPDATE", "REFRESH", "PRINT", and "TRỞ LẠI".

- +[+] Màn hình 2: Nhân viên bán hàng thực hiện thêm mới một chi tiết thuộc hóa đơn số 1

The screenshot shows the same "CHI TIẾT HÓA ĐƠN" window. The "Mã sách" field now contains 5, and the "Tên sách" field contains "Hoàng tử bé". The table now includes a new row for book ID 5:

Mã hóa đơn	Mã sách	Tên sách	Số lượng	Giá
1	1	Cuốn theo chiều gió	1	215000
1	2	Không già định	1	119850
1	5	Hoàng tử bé	1	75000

The other fields and buttons remain the same as in the previous screenshot.

- +[+] Màn hình 1: Nhân viên quản lý thực hiện refresh lại xem lần nữa thì thấy xuất hiện thêm dòng chi tiết mới

The screenshot shows a software window titled "CHI TIẾT HÓA ĐƠN". On the left, there are four input fields: "Mã hóa đơn" (Invoice ID) with value "1", "Mã sách" (Book ID) with a search button "SearchMaSach >>" and a button "S", "Tên sách" (Book Name), and "Số lượng" (Quantity). Below these is a "Giá" (Price) field. On the right, a table displays invoice details:

Mã hóa đơn	Mã sách	Tên sách	Số lượng	Giá
1	1	Cuốn theo chiều gió	1	215000
1	2	Không già đanh	1	119850
1	5	Hoàng tử bé	1	75000

At the bottom are several buttons: "INSERT", "DELETE", "UPDATE", "REFRESH", "PRINT", and "TRỞ LẠI" (Back).

Tình huống 3:

- ❖ Mô tả tình huống 3: Khi nhân viên quản lý thực hiện kiểm tra số lượng sách của một mã sách trong hệ thống nhưng chưa commit, cùng lúc đó một nhân viên bán hàng lại thực hiện thêm một chi tiết hóa đơn có chi tiết hóa đơn là số lượng mã sách đó và hệ thống cập nhật lại số lượng sách trong kho. Nhân viên quản lý kiểm tra số lượng sách một lần nữa thì không còn thấy dữ liệu ban đầu
- ❖ Nguyên nhân: Do trong khi nhân viên quản lý thực hiện tra cứu lần đầu chưa commit mà nhân viên bán hàng đã thực hiện thêm chi tiết hóa đơn làm cho hệ thống tự động cập nhật lại số lượng sách của mã sách có trong chi tiết hóa đơn, sau khi nhân viên bán hàng commit dữ liệu thì nhân viên quản lý tra cứu số lượng sách của mã sách đó một lần nữa, dữ liệu lúc này đã bị ghi lại và không còn thấy dữ liệu cũ nữa

Mô tả tình huống trên hệ quản trị Oracle:

T1 (Nhân viên quản lý thực hiện kiểm tra số lượng sách của sách số 2)	T2 (Nhân viên bán hàng thực hiện thêm một chi tiết hóa đơn có số lượng của mã sách số 2)
SET TRANSACTION ISOLATION LEVEL READ COMMITTED;	

<p>Select SL from SACH where MaSach = 2;</p> <pre>SQL> select sl from sach where masach=2; SL ----- 20</pre>	
	<p>SET TRANSACTION ISOLATION LEVEL READ COMMITTED; EXEC sp_ThemCTHD(368,2,10);</p> <pre>SQL> exec sp_ThemCTHD(368,2,10); PL/SQL procedure successfully completed.</pre>
<p>Select SL from SACH where MaSach = 2;</p> <pre>SQL> select sl from sach where masach=2; SL ----- 10</pre>	

✚ Tình huống 2:

- ❖ Mô tả tình huống: Khi nhân viên quản lý đang kiểm tra chi tiết phiếu nhập của phiếu nhập số 1, cùng lúc đó nhân viên kho thực hiện thêm một record chi tiết phiếu nhập mới trong phiếu nhập số 1. Kết quả khi nhân viên quản lý kiểm tra lại lần nữa thì thấy số lượng CTPN tăng lên 1 dòng.
- ❖ Nguyên nhân: Do trong khi nhân viên quản lý thực hiện tra cứu lần đầu chưa commit mà nhân viên kho đã thực hiện thêm một chi tiết phiếu nhập làm cho số lượng chi tiết trong phiếu nhập tăng lên, sau khi nhân viên kho commit dữ liệu thì nhân viên quản lý lại xem chi tiết trong mã phiếu nhập đó một lần nữa, dữ liệu lúc này đã bị thêm vào nhiều hơn dòng dữ liệu cũ.

Mô tả tình huống trên hệ quản trị Oracle:

<p>T1 (Nhân viên quản lý thực hiện kiểm tra (xem) chi tiết trong phiếu nhập số 1)</p>	<p>T2 (Nhân viên kho thực hiện thêm một chi tiết mới (insert) vào phiếu nhập số 1)</p>
---	--

SET TRANSACTION ISOLATION LEVEL
READ COMMITTED;
Select * from CTPN where MAPN = 1;

```
SQL> select * from CTPN where MaPN=1;

  MAPN    MASACH      SL      GIA
-----  -----  -----
    1          1      30    165385
    1          2      70    92192
```

Exec sp_ThemCTPN(1,3,10
,10000);

```
SQL> exec sp_ThemCTPN(1,3,10,10000);

PL/SQL procedure successfully completed.
```

Select * from CTPN where MAPN = 1;

```
SQL> select * from CTPN where MaPN=1;

  MAPN    MASACH      SL      GIA
-----  -----  -----
    1          3      10    10000
    1          1      30    165385
    1          2      70    92192
```

Mô tả tình huống trên chương trình

✚ Màn hình 1: Nhân viên quản lý thực hiện kiểm tra chi tiết phiếu nhập số 1

The screenshot shows a Windows application window titled "CHI TIẾT PHIẾU NHẬP". On the left, there are input fields for "Mã phiếu nhập" (Purchase Slip ID) set to 1, "Mã sách" (Book ID) with a dropdown menu showing "SearchMaSach", and "Tên sách" (Book Name). Below these are "Số lượng" (Quantity) and "Giá" (Price) fields. To the right is a table grid with columns: Mã phiếu nhập, Mã sách, Tên sách, Số lượng, and Giá. The grid contains two rows of data: row 1 has book IDs 1 and 2, names "Cuốn theo chiều gió" and "Không gia đình", quantity 30, and price 165385; row 2 has book ID 1, name "Không gia đình", quantity 70, and price 92192. At the bottom are buttons for INSERT, DELETE, UPDATE, REFRESH, and a red "TRỞ LẠI" (Back) button.

Mã phiếu nhập	Mã sách	Tên sách	Số lượng	Giá
1	1	Cuốn theo chiều gió	30	165385
	2	Không gia đình	70	92192

✚ Màn hình 2: Nhân viên bán hàng thực hiện thêm một chi tiết phiếu nhập mới vào phiếu nhập số 1

This screenshot is similar to the first one but includes a modal "Message" dialog box in the center. The dialog has an "i" icon and the text "Đã thêm thành công." (Successfully added). The main window shows the purchase slip grid with the new row added: book ID 3, name "Nhật Bản Dài - Tập 18 - Nobita Du Hành Biển Phương Nam", quantity 10, and price 10000. The "INSERT" button is highlighted.

Mã phiếu nhập	Mã sách	Tên sách	Số lượng	Giá
1	1	Cuốn theo chiều gió	30	165385
	2	Không gia đình	70	92192
	3	Nhật Bản Dài - Tập 18 - Nobita Du Hành Biển Phương Nam	10	10000

- ⊕ Màn hình 3: Nhân viên quản lý thực hiện refresh lại xem lần nữa thì thấy xuất hiện thêm dòng chi tiết mới

The screenshot shows a Windows application window titled "CHI TIẾT PHIẾU NHẬP". On the left, there are input fields for "Mã phiếu nhập" (Purchase Slip ID) set to 1, "Mã sách" (Book ID), "Tên sách" (Book Name), "Số lượng" (Quantity), and "Giá" (Price). Below these fields are four orange buttons: "Search", "INSERT", "DELETE", and "UPDATE". To the right of the input fields is a grid table with columns: "Mã phiếu nhập", "Mã sách", "Tên sách", "Số lượng", and "Giá". The table contains three rows of data:

Mã phiếu nhập	Mã sách	Tên sách	Số lượng	Giá
1	3	Doraemon Truyện ...	10	10000
1	1	Cuốn theo chiều gió	30	165385
1	2	Không già định	70	92192

At the bottom right of the window are two more orange buttons: "REFRESH" and "TRỞ LẠI" (Back).

⊕ Tình huống 3:

- ❖ Mô tả tình huống 3 : Khi nhân viên quản lý đang kiểm tra danh sách khách hàng của quản lý khách hàng , cùng lúc đó nhân viên bán hàng thực hiện thêm một khách hàng mới trong danh sách khách hàng. Kết quả khi nhân viên quản lý kiểm tra lại lần nữa thì thấy số lượng khách hàng tăng lên 1.
- ❖ Nguyên nhân: Do trong khi nhân viên quản lý thực hiện tra cứu lần đầu chưa commit mà nhân viên bán hàng đã thực hiện thêm một khách hàng mới làm cho số lượng khách hàng trong danh sách khách hàng tăng lên, sau khi nhân viên bán hàng commit dữ liệu thì nhân viên quản lý lại xem danh sách khách hàng một lần nữa, dữ liệu lúc này đã bị thêm vào nhiều hơn dòng dữ liệu cũ.

Mô tả tình huống trên hệ quản trị Oracle:

T1 (Nhân viên quản lý thực hiện kiểm tra (xem) danh sách khách hàng của quản lý khách hàng)	T2 (Nhân viên bán hàng thực hiện thêm 1 khách hàng mới vào danh sách)
SET TRANSACTION ISOLATION LEVEL READ COMMITTED;	

<pre>Select * from KhachHang</pre> <table border="1" style="width: 100%; border-collapse: collapse; font-family: monospace;"> <thead> <tr> <th>MAKH HO</th><th>TEN</th><th>NGAYSINH</th><th>GI</th></tr> <tr> <th>SODT</th><th>LOAIKH</th><th>DIEMTICHLUY</th><th></th></tr> </thead> <tbody> <tr> <td>40 Võ Ngữ Lĩnh</td><td>Xuân</td><td>03-DEC-01</td><td>Nữ</td></tr> <tr> <td>0567892163</td><td>Thị trấn Thủ Đức</td><td>1865200</td><td></td></tr> </tbody> </table> <p>40 rows selected.</p>	MAKH HO	TEN	NGAYSINH	GI	SODT	LOAIKH	DIEMTICHLUY		40 Võ Ngữ Lĩnh	Xuân	03-DEC-01	Nữ	0567892163	Thị trấn Thủ Đức	1865200		
MAKH HO	TEN	NGAYSINH	GI														
SODT	LOAIKH	DIEMTICHLUY															
40 Võ Ngữ Lĩnh	Xuân	03-DEC-01	Nữ														
0567892163	Thị trấn Thủ Đức	1865200															
	<pre>ALTER SESSION SET NLS_DATE_FORMAT ='DD/MM/YYYY'; Commit; insert into KHACHHANG(MaKH, Ho, Ten, NgaySinh, GioiTinh, SoDT) values (MAKH_SEQ4.nextval, N'Võ ', N'Xuân', '03/12/2001', N'Nữ', '0567892163'); Commit;</pre> <pre>SQL> ALTER SESSION SET NLS_DATE_FORMAT = 'DD/MM/YY'; Session altered. SQL> commit; Commit complete. SQL> insert into KHACHHANG(MaKH, Ho, Ten, NgaySinh, GioiTinh, SoDT) values 4.nextval, N'Võ ', N'Xuân', '03/12/2001', N'Nữ', '0567892163'); 1 row created. SQL> commit; Commit complete.</pre>																
<pre>Select * from KhachHang</pre> <div style="background-color: black; color: white; padding: 5px; text-align: center;"> 41 rows selected. </div>																	

Mô tả tình huống trên chương trình

- ⊕ Màn hình 1: Nhân viên quản lý kiểm tra danh sách khách hàng của quản lý khách hàng

The screenshot shows a Windows application window titled "QUẢN LÝ KHÁCH HÀNG". It has two main sections: "THÔNG TIN KHÁCH HÀNG" (Customer Information) on the left and "DANH SÁCH KHÁCH HÀNG" (Customer List) on the right.

THÔNG TIN KHÁCH HÀNG:

- Mã khách hàng:
- +
-
- Họ:
- Tên:
- Ngày sinh:
- Giới tính: Nam Nữ
- Số ĐT:
- Loại khách hàng: Thân thiết Thường VIP
- Điểm tích lũy:

DANH SÁCH KHÁCH HÀNG:

MAKH	Họ	Tên	Ngày sinh	Giới tính	Số ĐT	Loại KH	Điểm TL
23	Lê Nhựt	Trường	2002-04-12	Nam	08251873...	VIP	3057650
24	Phạm Trọ...	Tuân	2003-02-13	Nam	01632738...	Thân Thiết	675100
25	Trần Hồng	Nhung	2003-11-18	Nữ	01729373...	VIP	3712500
26	Trần Ngọc	Như	2000-02-03	Nữ	01728391...	VIP	2575700
27	Nguyễn T...	Phát	1998-11-02	Nam	07897481...	Thân Thiết	1876100
28	Đỗ Thị	Hà	2000-12-17	Nữ	01111821...	VIP	2195450
29	Nguyễn Hà	My	2005-09-12	Nữ	08251877...	Thân Thiết	1518400
30	Lê Anh	Duy	1999-02-13	Nam	01678938...	Thân Thiết	1087200
31	Nguyễn ...	Ngân	1994-11-18	Nữ	01729377...	Thân Thiết	1942800
32	Nguyễn Á...	Hồng	2001-02-03	Nữ	05678391...	Thân Thiết	1787550
33	Nguyễn ...	Phúc	1997-11-18	Nam	01722343...	VIP	3079350
34	Lê Bá Nhật	Long	2000-02-13	Nam	01234391...	Thân Thiết	1964250
35	Nguyễn ...	Việt	1996-11-12	Nam	03456481...	VIP	2048600
36	Trương ...	Châu	2000-02-17	Nữ	01112221...	Thân Thiết	1707250
37	Nguyễn T...	My	2004-09-12	Nữ	08251899...	VIP	2444300
38	Nguyễn V...	Chương	1977-12-13	Nam	01678944...	Thân Thiết	1035450
39	Nguyễn V...	Hoàng	1974-11-08	Nam	01723477...	VIP	3138050
40	Võ Ngọc ...	Xuân	2001-12-03	Nữ	05678921...	Thân Thiết	1865200

- ⊕ Màn hình 2: Nhân viên bán hàng thực hiện thêm mới một khách hàng vào danh sách

The screenshot shows a Windows application window titled "QUẢN LÝ KHÁCH HÀNG". It has two main sections: "THÔNG TIN KHÁCH HÀNG" (Customer Information) on the left and "DANH SÁCH KHÁCH HÀNG" (Customer List) on the right.

THÔNG TIN KHÁCH HÀNG:

- Mã khách hàng: 41
- Họ: Võ
- Tên: Xuân
- Ngày sinh: Dec 3, 2001
- Giới tính: Nam Nữ
- Số ĐT: 0567892163
- Loại khách hàng: Thân thiết Thường VIP
- Điểm tích lũy: 0

DANH SÁCH KHÁCH HÀNG:

MAKH	Họ	Tên	Ngày sinh	Giới tính	Số ĐT	Loại KH	Điểm TL
23	Lê Nhựt	Trường	2002-04-12	Nam	08251873...	VIP	3057650
24	Phạm Trọ...	Tuân	2003-02-13	Nam	01632738...	Thân Thiết	675100
25	Trần Hồng	Nhung	2003-11-18	Nữ	01729373...	VIP	3712500
26	Trần Ngọc	Như	2000-02-03	Nữ	01728391...	VIP	2575700
27	Nguyễn T...	Phát	1998-11-02	Nam	07897481...	Thân Thiết	1876100
28	Đỗ Thị	Hà	2000-12-17	Nữ	01111821...	VIP	2195450
29	Nguyễn Hà	My	2005-09-12	Nữ	08251877...	Thân Thiết	1518400
30	Lê Anh	Duy	1999-02-13	Nam	01678938...	Thân Thiết	1087200
31	Nguyễn ...	Ngân	1994-11-18	Nữ	01729377...	Thân Thiết	1942800
32	Nguyễn Á...	Hồng	2001-02-03	Nữ	05678391...	Thân Thiết	1787550
33	Nguyễn ...	Phúc	1997-11-18	Nam	01722343...	VIP	3079350
34	Lê Bá Nhật	Long	2000-02-13	Nam	01234391...	Thân Thiết	1964250
35	Nguyễn ...	Việt	1996-11-12	Nam	03456481...	VIP	2048600
36	Trương ...	Châu	2000-02-17	Nữ	01112221...	Thân Thiết	1707250
37	Nguyễn T...	My	2004-09-12	Nữ	08251899...	VIP	2444300
38	Nguyễn V...	Chương	1977-12-13	Nam	01678944...	Thân Thiết	1035450
39	Nguyễn V...	Hoàng	1974-11-08	Nam	01723477...	VIP	3138050
40	Võ Ngọc ...	Xuân	2001-12-03	Nữ	05678921...	Thân Thiết	1865200

A modal dialog box titled "Message" displays the message: "Khách hàng được thêm vào thành công!" (Customer added successfully!).

- ✚ Màn hình 3: Nhân viên quản lý thực hiện refresh lại thì thấy danh sách khách hàng tăng lên thêm 1 người

The screenshot shows a Windows application window titled "QUẢN LÝ KHÁCH HÀNG". The left panel is labeled "THÔNG TIN KHÁCH HÀNG" and contains input fields for: Mã khách hàng (41), Họ (Võ), Tên (Xuân), Ngày sinh (Dec 3, 2001), Giới tính (Nam), Số ĐT (0567892163), Loại khách hàng (Thân thiết), and Điểm tích lũy (0). Below these are buttons for INSERT, DELETE, UPDATE, and REFRESH. The right panel is labeled "DANH SÁCH KHÁCH HÀNG" and displays a table of customer records. A new row is added at the bottom of the table, corresponding to the search results in the left panel.

MAKH	Họ	Tên	Ngày sinh	Giới tính	Số ĐT	Loại KH	Điểm TL
24	Phạm Trọ...	Tuấn	2003-02-13	Nam	01632738...	Thân Thiết	675100
25	Trần Hồng	Nhung	2003-11-18	Nữ	01729373...	VIP	3712500
26	Trần Ngọc	Như	2000-02-03	Nữ	01728391...	VIP	2575700
27	Nguyễn T...	Phát	1998-11-02	Nam	07897481...	Thân Thiết	1876100
28	Đỗ Thị	Hà	2000-12-17	Nữ	01111821...	VIP	2195450
29	Nguyễn Hả	My	2005-09-12	Nữ	08251877...	Thân Thiết	1518400
30	Lê Anh	Duy	1999-02-13	Nam	01678938...	Thân Thiết	1087200
31	Nguyễn ...	Ngân	1994-11-18	Nữ	01729377...	Thân Thiết	1942800
32	Nguyễn Á...	Hồng	2001-02-03	Nữ	05678391...	Thân Thiết	1787550
33	Nguyễn ...	Phúc	1997-11-18	Nam	01722343...	VIP	3079350
34	Lê Bá Nhát	Long	2000-02-13	Nam	01234391...	Thân Thiết	1964250
35	Nguyễn ...	Việt	1996-11-12	Nam	03456481...	VIP	2048600
36	Trương ...	Châu	2000-02-17	Nữ	01112221...	Thân Thiết	1707250
37	Nguyễn T...	My	2004-09-12	Nữ	08251899...	VIP	2444300
38	Nguyễn V...	Chương	1977-12-13	Nam	01678944...	Thân Thiết	1035450
39	Nguyễn V...	Hoàng	1974-11-08	Nam	01723477...	VIP	3138050
40	Võ Ngọc ...	Xuân	2001-12-03	Nữ	05678921...	Thân Thiết	1865200
41	Võ	Xuân	2001-12-03	Nữ	05678921...	Thân thiết	0

- ❖ Giải pháp ngăn chặn: Thay mức cõi lập mặc định (Read committed) thành Serializable để giải quyết tất cả vấn đề Phantom Read cho T1.

PHẦN IV. THIẾT KẾ GIAO DIỆN

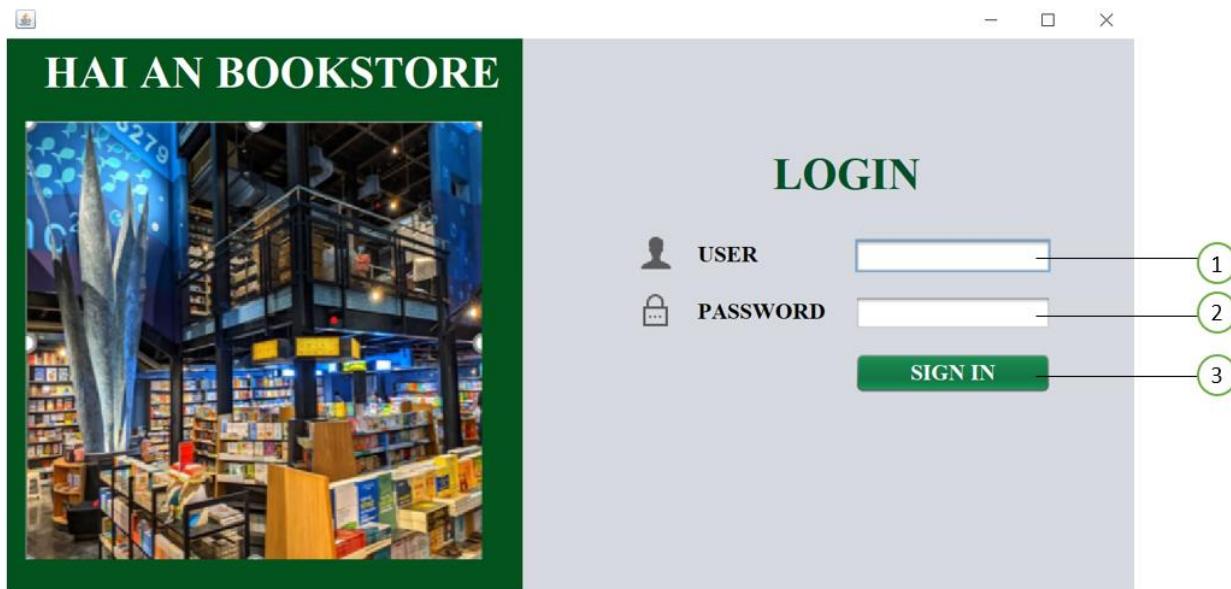
1. Thiết kế giao diện

1.1. Danh sách các giao diện

STT	Giao Diện	Chức Năng
1	Đăng nhập	Đăng nhập hệ thống
2	Trang chủ	Hiển thị trang chủ và các chức năng có trên trang
3	Tài khoản	Hiển thị thông tin tài khoản, thống kê bán hàng

4	Hóa đơn	Quản lý thông tin hóa đơn
5	Chi tiết hóa đơn	Quản lý thông tin chi tiết hóa đơn
6	In hóa đơn	Hiển thị thông tin của hóa đơn
7	Sách	Hiển thị thông tin sách và mô tả nội dung chính của sách
8	Quản lý	Hiển thị các loại quản lý
9	Quản lý nhân viên	Quản lý thông tin nhân viên
10	Quản lý khách hàng	Quản lý thông tin khách hàng
11	Quản lý sách	Quản lý thông tin sách
12	Quản lý tài khoản	Quản lý thông tin tài khoản
13	Quản lý phiếu nhập	Quản lý thông tin phiếu nhập
14	Quản lý nhà cung cấp	Quản lý thông tin nhà cung cấp
15	Thống kê	Hiển thị các thông tin thống kê như sách sắp hết, xếp hạng nhân viên, top sách mua nhiều, top khách hàng

1.2. Giao diện đăng nhập



Mô tả giao diện

STT	Tên	Kiểu	Chức năng
1	USER	JTextField	Nhập tên tài khoản
2	PASSWORD	JTextField	Nhập mật khẩu tài khoản
3	SIGN IN	JButton	Thực hiện thao tác đăng nhập

1.3. Giao diện trang chủ



Mô tả giao diện

STT	Tên	Kiểu	Chức năng
1	TÀI KHOẢN	JButton	Nhấn để vào trang tài khoản
2	HÓA ĐƠN	JButton	Nhấn để vào trang hóa đơn
3	SÁCH	JButton	Nhấn để vào trang sách
4	QUẢN LÝ	JButton	Nhấn để vào trang quản lý
5	ĐĂNG XUẤT	JButton	Nhấn để đăng xuất tài khoản

1.4. Giao diện trang tài khoản

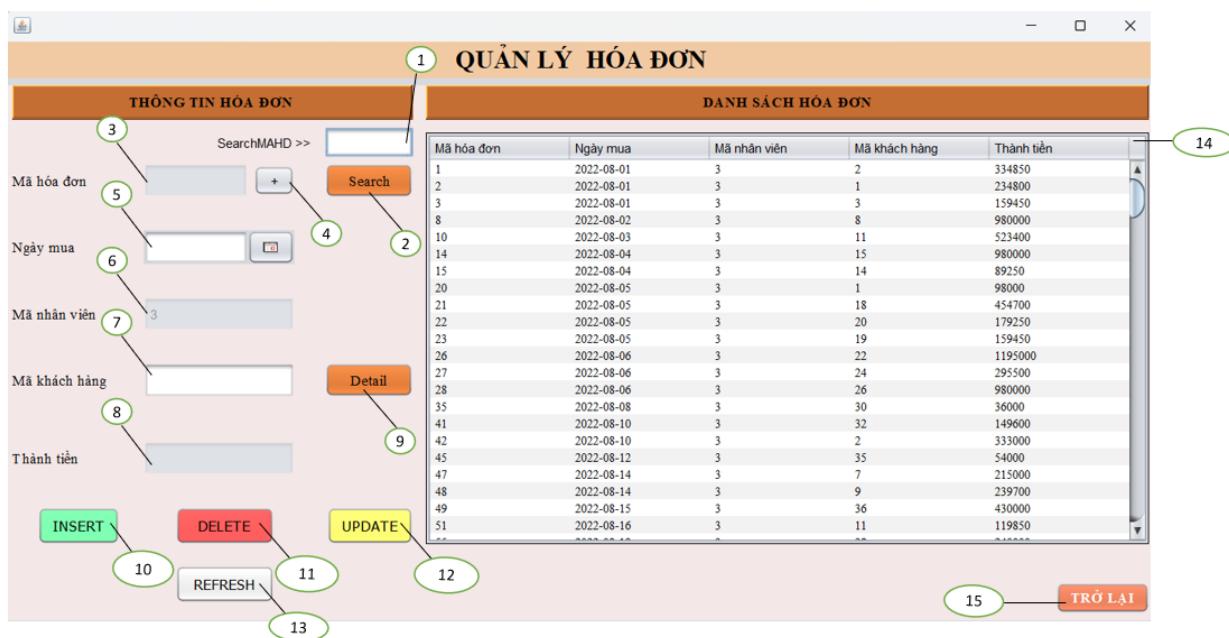


Mô tả giao diện

STT	Tên	Kiểu	Chức năng
1	MaTK	JTextField	Hiển thị mã tài khoản
2	MaNV	JTextField	Hiển thị mã nhân viên
3	Name	JTextField	Hiển thị tên
4	GioiTinh	JTextField	Hiển thị giới tính
5	SĐT	JTextField	Hiển thị số điện thoại
6	CMND	JTextField	Hiển thị chứng minh nhân dân
7	NgaySinh	JTextField	Hiển thị ngày sinh
8	DiaChi	JTextField	Hiển thị địa chỉ
9	NgayBD	JTextField	Hiển thị ngày bắt đầu
10	Luong	JTextField	Hiển thị lương
11	RANK	JTextField	Hiển thị mức xếp hạng
12	Số lượng hóa đơn	JTextField	Hiển thị số lượng hóa đơn

13	Doanh thu	JTextField	Hiển thị doanh thu
14	TRỞ LẠI	JButton	Nhấn để quay trở lại trang chủ

1.5. Giao diện trang hóa đơn

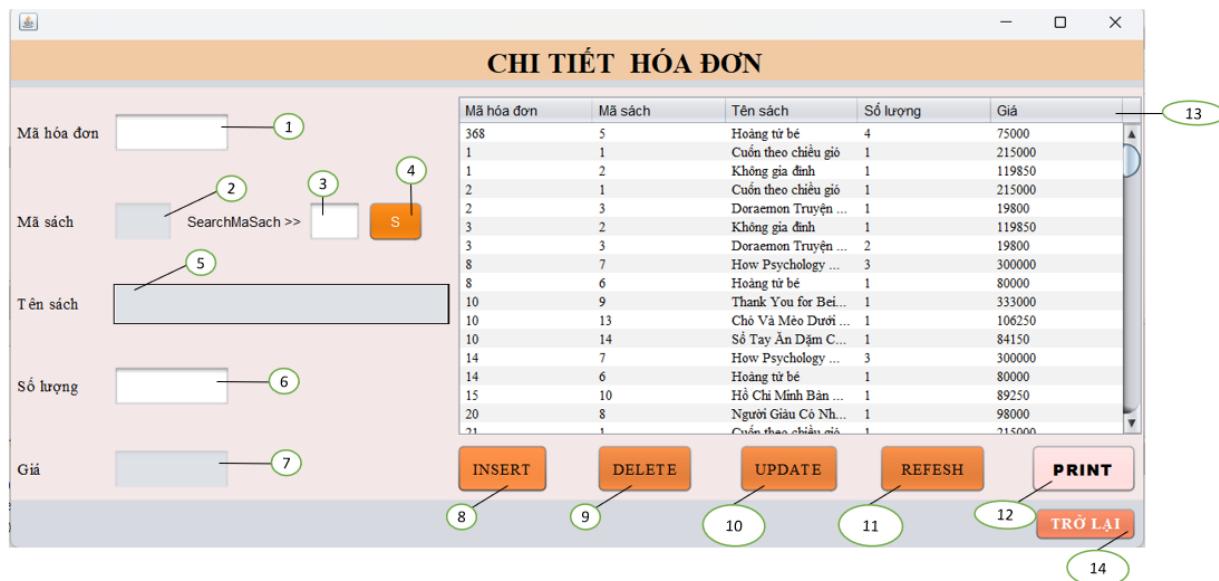


Mô tả giao diện

STT	Tên	Kiểu	Chức năng
1	SearchMAHD	JTextField	Nhập mã hóa đơn để tìm kiếm
2	Search	JButton	Nhấn để tìm kiếm
3	Mã hóa đơn	JTextField	Hiển thị mã hóa đơn
4	+	JButton	Nhấn để tự động hiển thị mã hóa đơn mới nhất

5	Ngày mua	JDateChooser	Nhập ngày mua
6	Mã nhân viên	JTextField	Hiển thị mã nhân viên
7	Mã khách hàng	JTextField	Nhập mã khách hàng
8	Thành tiền	JTextField	Hiển thị tổng tiền
9	Detail	JButton	Nhấn để vào trang chi tiết hóa đơn
10	INSERT	JButton	Nhấn để thực hiện chức năng thêm
11	DELETE	JButton	Nhấn để thực hiện chức năng xóa
12	UPDATE	JButton	Nhấn để thực hiện chức năng sửa
13	REFRESH	JButton	Nhấn để thực hiện chức năng làm mới
14	DANH SÁCH HÓA ĐƠN	JTable	Hiển thị thông tin của hóa đơn
15	TRỞ LẠI	JButton	Nhấn để trở lại trang chủ

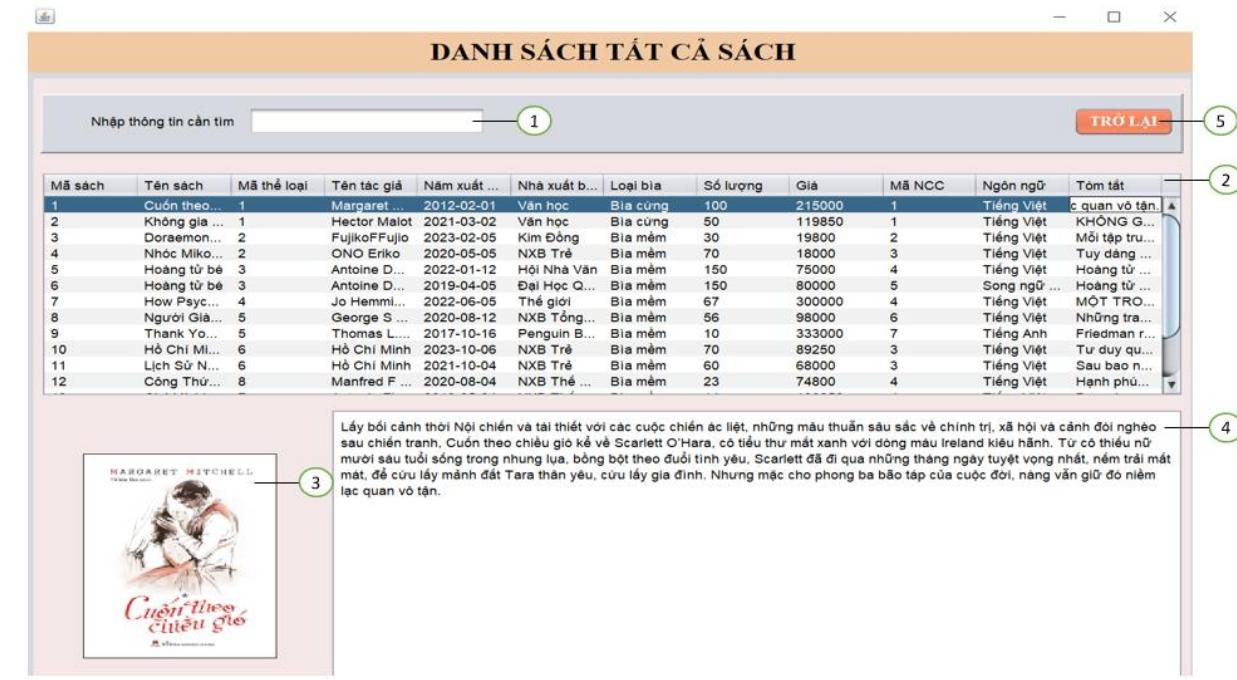
1.6. Giao diện trang chi tiết hóa đơn



Mô tả giao diện

STT	Tên	Kiểu	Chức năng
1	Mã hóa đơn	JTextField	Nhập mã hóa đơn
2	Mã sách	JButton	Hiển thị mã sách
3	SearchMaSach	JTextField	Nhập mã sách
4	S	JButton	Nhấn để tìm mã sách
5	Tên sách	JDateChooser	Hiển thị tên sách
6	Số lượng	JTextField	Nhập số lượng
7	Giá	JTextField	Hiển thị giá
8	INSERT	JButton	Nhấn để thực hiện chức năng thêm
9	DELETE	JButton	Nhấn để thực hiện chức năng xóa
10	UPDATE	JButton	Nhấn để thực hiện chức năng sửa
11	REFRESH	JButton	Nhấn để thực hiện chức năng làm mới
12	PRINT	JButton	Nhấn để in hóa đơn
13	BẢNG CTHD	JTable	Hiển thị thông tin của CTHD
14	TRỞ LẠI	JButton	Nhấn để trở lại trang hóa đơn

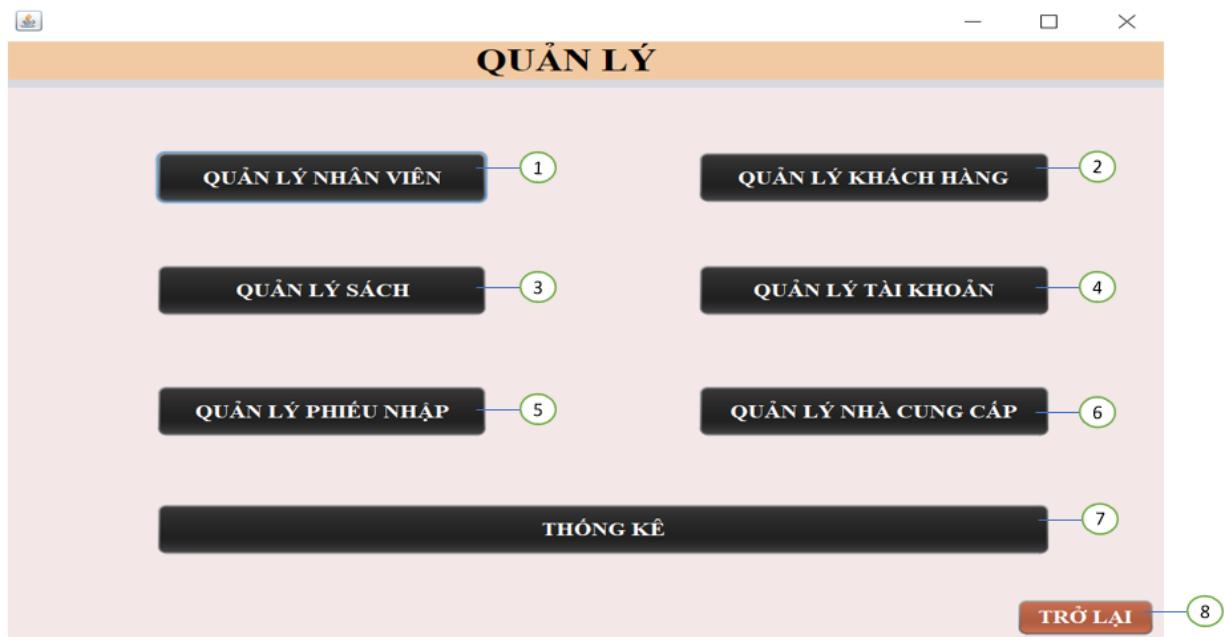
1.7. Giao diện trang sách



Mô tả giao diện

STT	Tên	Kiểu	Chức năng
1	Nhập thông tin cần tìm	JTextField	Nhập thông tin sách cần tìm
2	Bảng thông tin sách	JTable	Hiển thị thông tin của sách
3	Hình ảnh	JLabel	Hiển thị hình ảnh của sách
4	Tóm tắt	JScrollPane	Hiển thị tóm tắt của sách
5	TRỞ LẠI	JButton	Nhấn để trở lại trang chủ

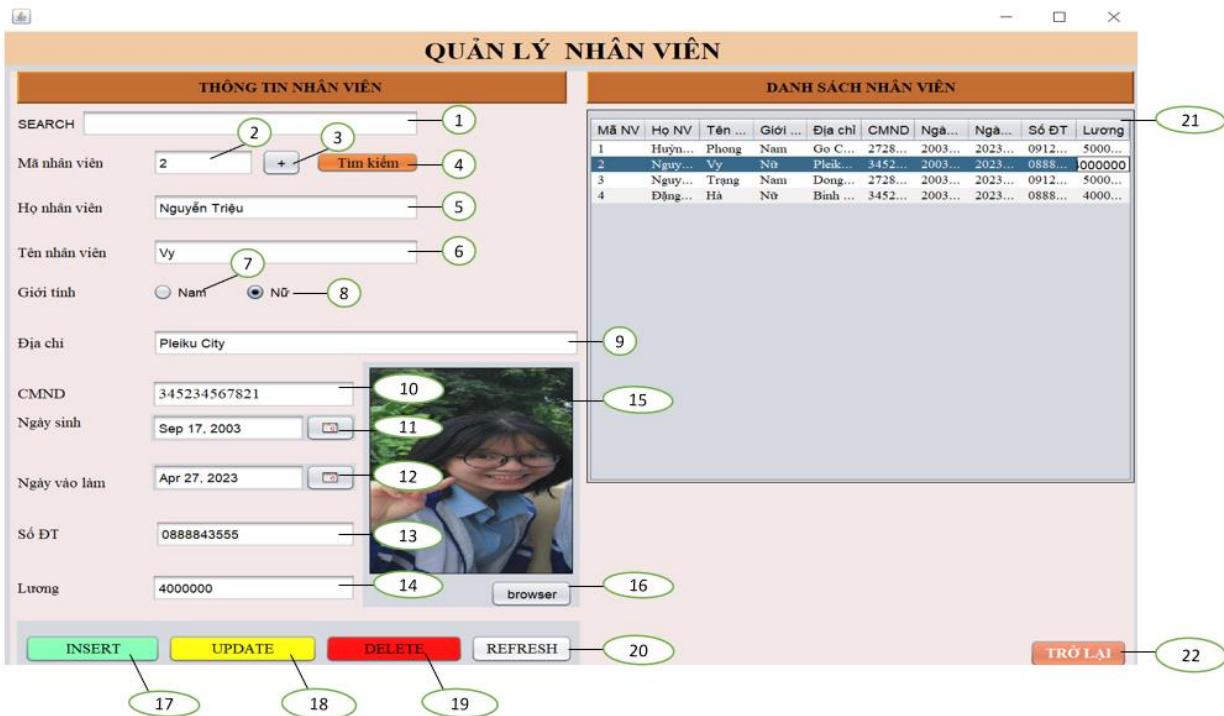
1.8. Giao diện trang quản lý



Mô tả giao diện

STT	Tên	Kiểu	Chức năng
1	Quản lý nhân viên	JButton	Nhấn để vào trang quản lý nhân viên
2	Quản lý khách hàng	JButton	Nhấn để vào trang quản lý khách hàng
3	Quản lý sách	JButton	Nhấn để vào trang quản lý sách
4	Quản lý tài khoản	JButton	Nhấn để vào trang quản lý tài khoản
5	Quản lý phiếu nhập	JButton	Nhấn để vào trang quản lý phiếu nhập
6	Quản lý nhà cung cấp	JButton	Nhấn để vào trang quản lý nhà cung cấp
7	Thông kê	JButton	Nhấn để vào trang thống kê
8	TRỞ LẠI	JButton	Nhấn để trở lại trang chủ

1.9. Giao diện trang quản lý nhân viên

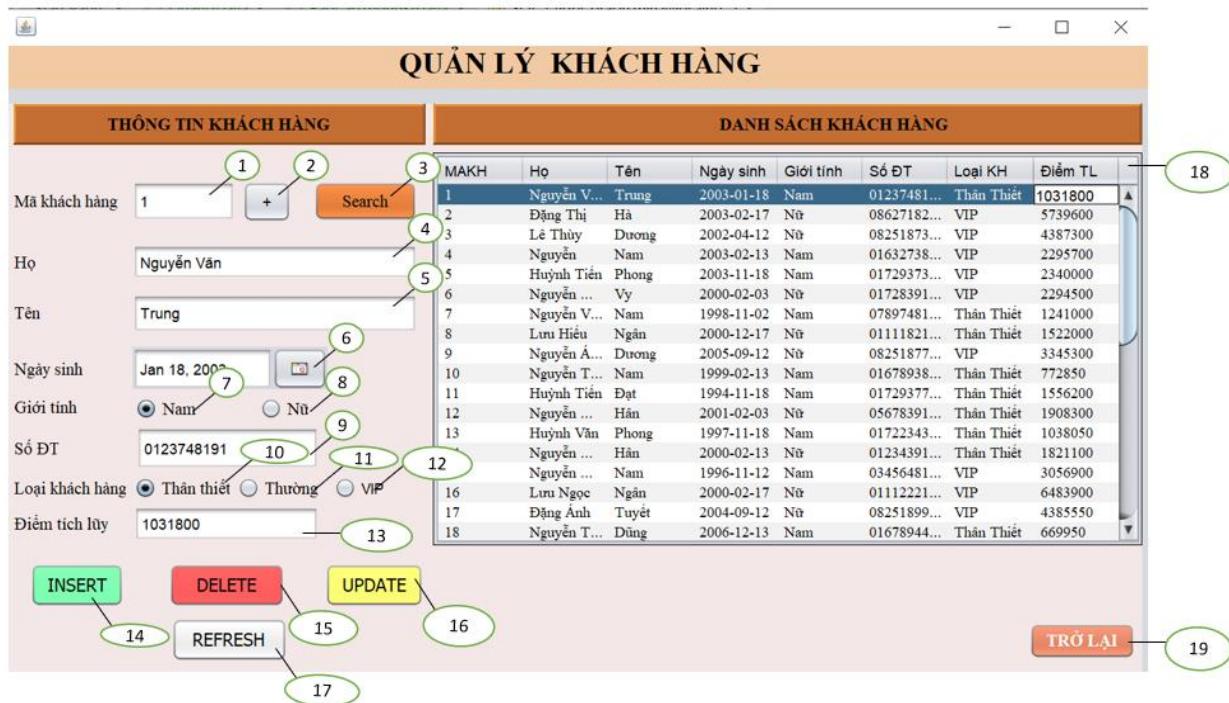


Mô tả giao diện

STT	Tên	Kiểu	Chức năng
1	SEARCH	JTextField	Nhập thông tin nhân viên để tìm kiếm
2	Mã nhân viên	JTextField	Nhập mã nhân viên
3	+	JButton	Nhấn vào để hiển thị mã nhân viên mới nhất
4	Tìm kiếm	JButton	Nhấn vào để tìm kiếm nhân viên
5	Họ nhân viên	JTextField	Nhập họ nhân viên
6	Tên nhân viên	JTextField	Nhập tên nhân viên
7	Nam	JRadioButton	Nhấn để chọn giới tính nam
8	Nữ	JRadioButton	Nhấn để chọn giới tính nữ
9	Địa chỉ	JTextField	Nhập địa chỉ
10	CMND	JTextField	Nhập CMND
11	Ngày sinh	JDateChooser	Nhập ngày sinh

12	Ngày vào làm	JDateChooser	Nhập ngày vào làm
13	Số ĐT	JTextField	Nhập số điện thoại
14	Lương	JTextField	Nhập thị lương
15	Hình ảnh	JLabel	Hiển thị hình ảnh nhân viên
16	browser	JButton	Nhấn để vào thư mục và chọn ảnh
17	INSERT	JButton	Nhấn để thực hiện chức năng thêm
18	DELETE	JButton	Nhấn để thực hiện chức năng xóa
19	UPDATE	JButton	Nhấn để thực hiện chức năng sửa
20	REFRESH	JButton	Nhấn để thực hiện chức năng làm mới
21	DANH SÁCH NHÂN VIÊN	JTable	Hiển thị thông tin của nhân viên
22	TRỞ LẠI	JButton	Nhấn để trở lại trang quản lý

1.10. Giao diện trang quản lý khách hàng

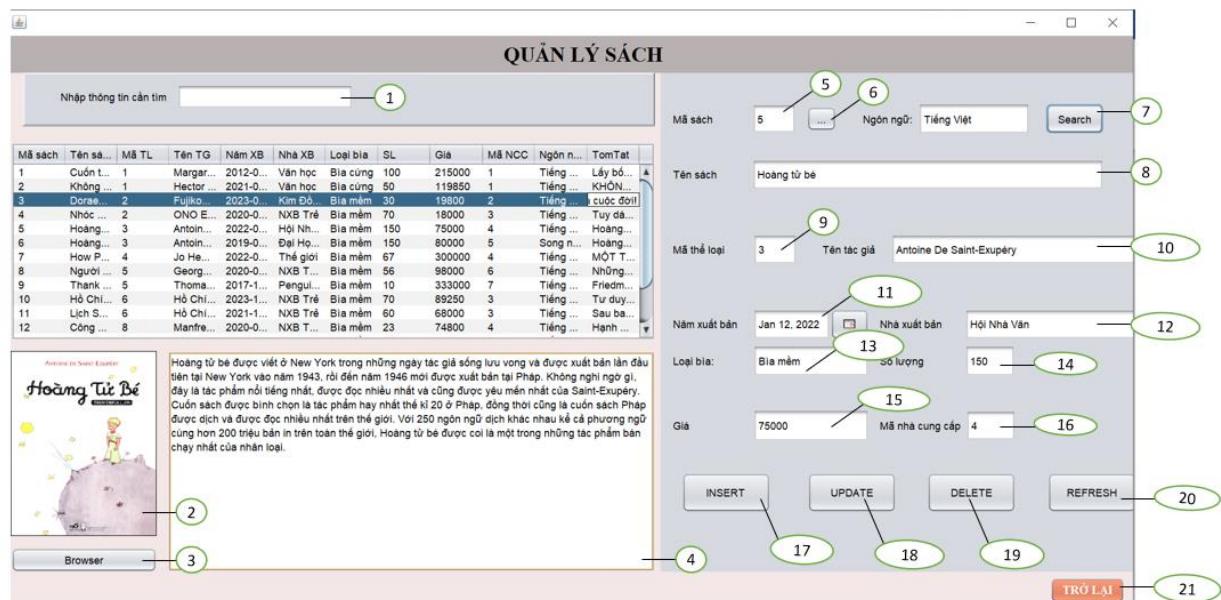


Mô tả giao diện

STT	Tên	Kiểu	Chức năng
1	Mã khách hàng	JTextField	Nhập mã khách hàng
2	+	JButton	Nhấn vào để hiển thị mã khách hàng mới nhất
3	Search	JButton	Nhấn vào để tìm mã khách hàng
4	Họ	JTextField	Nhập họ
5	Tên	JTextField	Nhập tên
6	Ngày sinh	JDateChooser	Nhập ngày sinh
7	Nam	JRadioButton	Nhấn để chọn giới tính nam
8	Nữ	JRadioButton	Nhấn để chọn giới tính nữ
9	Số ĐT	JTextField	Nhập số điện thoại

10	Thân thiết	JRadioButton	Nhấn để chọn loại khách hàng thân thiết
11	Thường	JRadioButton	Nhấn để chọn loại khách hàng thường
12	VIP	JRadioButton	Nhấn để chọn loại khách hàng VIP
13	Điểm tích lũy	JTextField	Nhập điểm tích lũy
14	INSERT	JButton	Nhấn để thực hiện chức năng thêm
15	DELETE	JButton	Nhấn để thực hiện chức năng xóa
16	UPDATE	JButton	Nhấn để thực hiện chức năng sửa
17	REFRESH	JButton	Nhấn để thực hiện chức năng làm mới
18	DANH SÁCH KHÁCH HÀNG	JTable	Hiển thị thông tin của nhân viên
19	TRỞ LẠI	JButton	Nhấn để trở lại trang quản lý

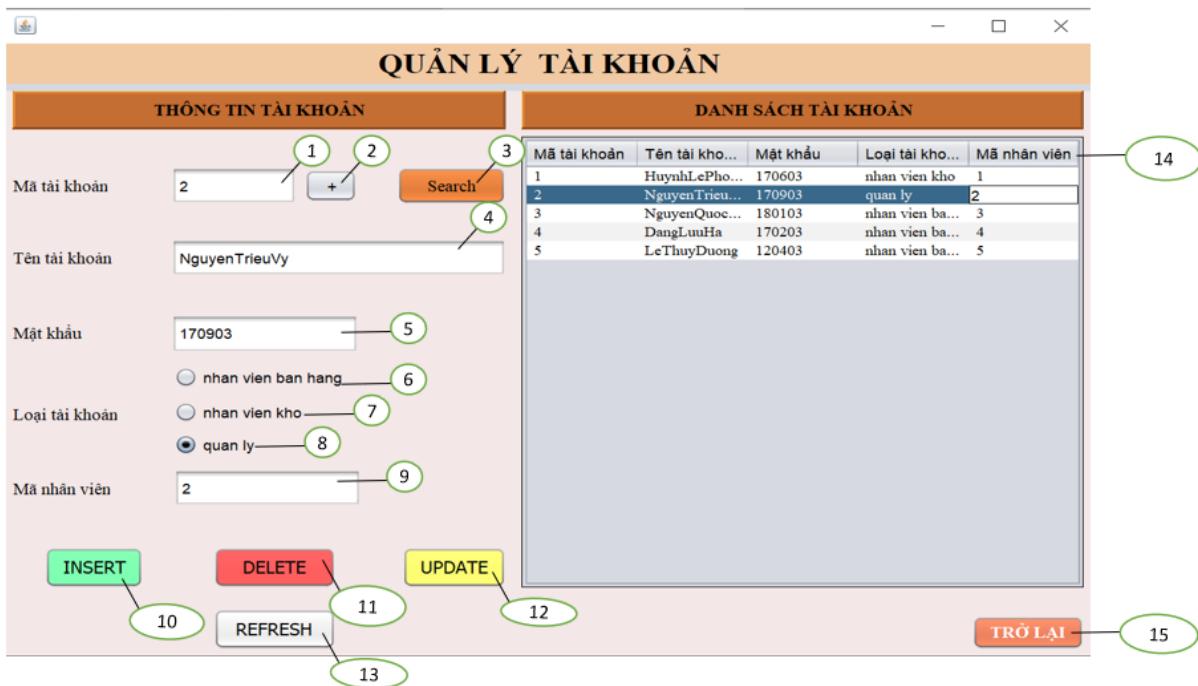
1.11. Giao diện trang quản lý sách



Mô tả giao diện

STT	Tên	Kiểu	Chức năng
1	Nhập thông tin cần tìm	JTextField	Nhập thông tin sách cần tìm
2	Hình ảnh	JLabel	Hiển thị hình ảnh sách
3	Browser	JButton	Nhấn để chọn hình ảnh từ thư mục
4	Tóm tắt	JTextArea	Hiển thị tóm tắt nội dung của sách
5	Mã sách	JTextField	Nhập mã sách
6	+	JButton	Nhấn vào để hiển thị mã sách mới nhất
7	Search	JRadioButton	Nhấn để tìm kiếm mã sách
8	Tên sách	JTextField	Nhập tên sách
9	Mã thể loại	JTextField	Nhập mã thể loại
10	Tên tác giả	JTextField	Nhập tên tác giả
11	Năm xuất bản	JDateChooser	Nhập năm xuất bản
12	Nhà xuất bản	JTextField	Nhập nhà xuất bản
13	Loại bìa	JTextField	Nhập loại bìa
14	Số lượng	JTextField	Nhập số lượng
15	Giá	JTextField	Nhập giá
16	Mã nhà cung cấp	JTextField	Nhập mã nhà cung cấp
17	INSERT	JButton	Nhấn để thực hiện chức năng thêm
18	DELETE	JButton	Nhấn để thực hiện chức năng xóa
19	UPDATE	JButton	Nhấn để thực hiện chức năng sửa
20	REFRESH	JButton	Nhấn để thực hiện chức năng làm mới
21	TRỞ LẠI	JButton	Nhấn để trở lại trang quản lý

1.12. Giao diện trang quản lý tài khoản



Mô tả giao diện

STT	Tên	Kiểu	Chức năng
1	Mã tài khoản	JTextField	Nhập mã tài khoản
2	+	JButton	Nhấn để hiển thị mã tài khoản mới nhất
3	Search	JButton	Nhấn để tìm kiếm mã tài khoản
4	Tên tài khoản	JTextField	Nhập tên tài khoản
5	Mật khẩu	JTextField	Nhập mật khẩu
6	Nhan vien ban hang	JRadioButton	Nhấn để chọn loại nhân viên bán hàng
7	Nhan vien kho	JRadioButton	Nhấn để chọn loại nhân viên kho
8	Quan ly	JRadioButton	Nhấn để chọn loại nhân viên quản lý
9	Mã nhân viên	JTextField	Nhập mã nhân viên
10	INSERT	JButton	Nhấn để thực hiện chức năng thêm
11	DELETE	JButton	
12	UPDATE	JButton	
13	REFRESH	JButton	
14			
15			

11	DELETE	JButton	Nhấn để thực hiện chức năng xóa
12	UPDATE	JButton	Nhấn để thực hiện chức năng sửa
13	REFRESH	JButton	Nhấn để thực hiện chức năng làm mới
14	DANH SÁCH TÀI KHOẢN	JTable	Hiển thị thông tin của các tài khoản
15	TRỞ LẠI	JButton	Nhấn để trở lại trang quản lý

1.13. Giao diện trang quản lý phiếu nhập

QUẢN LÝ PHIẾU NHẬP

Mã phiếu nhập	Ngày nhập	Mã nhân viên	Tổng tiền
1	2023-05-10	1	9571150
2	2023-05-11	1	13183470
3	2023-05-12	2	5100000
4	2023-05-13	3	15810790
5	2023-05-14	3	27000000
6	2023-05-15	4	33672000
7	2023-05-16	4	28492234
8	2023-05-17	2	37025624
9	2023-05-18	1	32914680
10	2023-05-19	2	7113900

Mô tả giao diện

STT	Tên	Kiểu	Chức năng
1	Mã phiếu nhập	JTextField	Nhập mã phiếu nhập
2	+	JButton	Nhấn để hiển thị mã phiếu nhập mới nhất
3	SearchMaPN	JTextField	Nhập mã phiếu nhập

4	S	JButton	Nhấn để tìm kiếm mã phiếu nhập
5	Ngày nhập	JDateChooser	Nhập ngày nhập
6	Mã nhân viên	JComboBox	Chọn mã nhân viên
7	Tổng tiền	JTextField	Nhập tổng tiền
8	Detail	JButton	Nhấn để hiển thị trang chi tiết phiếu nhập
9	INSERT	JButton	Nhấn để thực hiện chức năng thêm
10	DELETE	JButton	Nhấn để thực hiện chức năng xóa
11	UPDATE	JButton	Nhấn để thực hiện chức năng sửa
12	REFRESH	JButton	Nhấn để thực hiện chức năng làm mới
13	DANH SÁCH PHIẾU NHẬP	JTable	Hiển thị thông tin của các phiếu nhập
14	TRỞ LẠI	JButton	Nhấn để trở lại trang quản lý

1.14. Giao diện trang chi tiết phiếu nhập

The screenshot shows the 'CHI TIẾT PHIẾU NHẬP' (Detail Receipt) window. The window title is 'CHI TIẾT PHIẾU NHẬP'. On the left side, there are several input fields and buttons:

- 'Mã phiếu nhập' (Receipt ID) with value '1' (labeled 1)
- 'Mã sách' (Book ID) with value '1' (labeled 2)
- 'Tên sách' (Book Name) with value 'Cuốn theo chiều gió' (labeled 5)
- 'Số lượng' (Quantity) with value '30' (labeled 6)
- 'Giá' (Price) with value '165385' (labeled 7)
- A 'Search' button (labeled 3)
- A 'REFRESH' button (labeled 11)
- A 'TRỞ LẠI' (Return) button (labeled 13)

In the center, there is a table titled 'Danh sách phiếu nhập' (Receipt List) with the following data:

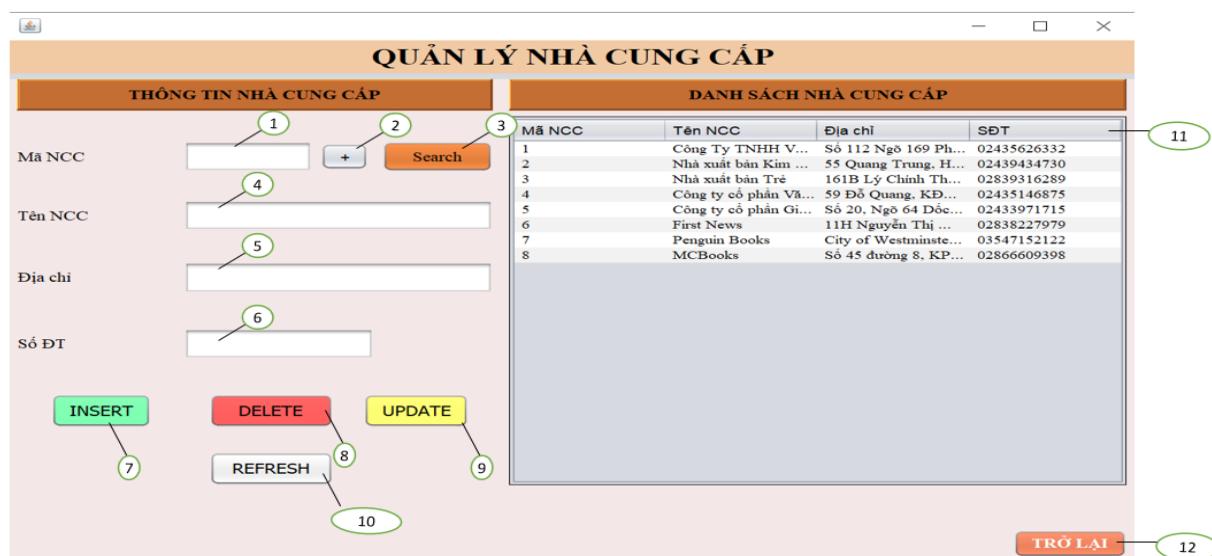
Mã phiếu nhập	Mã sách	Tên sách	Số lượng	Giá
1	1	Cuốn theo chiều gió	30	165385
1	2	Không gia đình	50	92192

At the bottom, there are buttons for 'INSERT' (labeled 8), 'DELETE' (labeled 9), 'UPDATE' (labeled 10), and another 'REFRESH' button (labeled 12).

Mô tả giao diện

STT	Tên	Kiểu	Chức năng
1	Mã phiếu nhập	JTextField	Hiển thị mã phiếu nhập
2	Mã sách	JTextField	Hiển thị mã sách
3	SearchMaSach	JTextField	Nhập mã sách
4	Search	JButton	Nhấn để tìm kiếm mã sách
5	Tên sách	JTextField	Hiển thị tên sách
6	Số lượng	JTextField	Nhập số lượng sách
7	Giá	JTextField	Nhập giá sách
8	INSERT	JButton	Nhấn để thực hiện chức năng thêm
9	DELETE	JButton	Nhấn để thực hiện chức năng xóa
10	UPDATE	JButton	Nhấn để thực hiện chức năng sửa
11	REFRESH	JButton	Nhấn để thực hiện chức năng làm mới
12	DANH SÁCH CTPN	JTable	Hiển thị thông tin của các chi tiết phiếu nhập
13	TRỞ LẠI	JButton	Nhấn để trở lại trang quản lý phiếu nhập

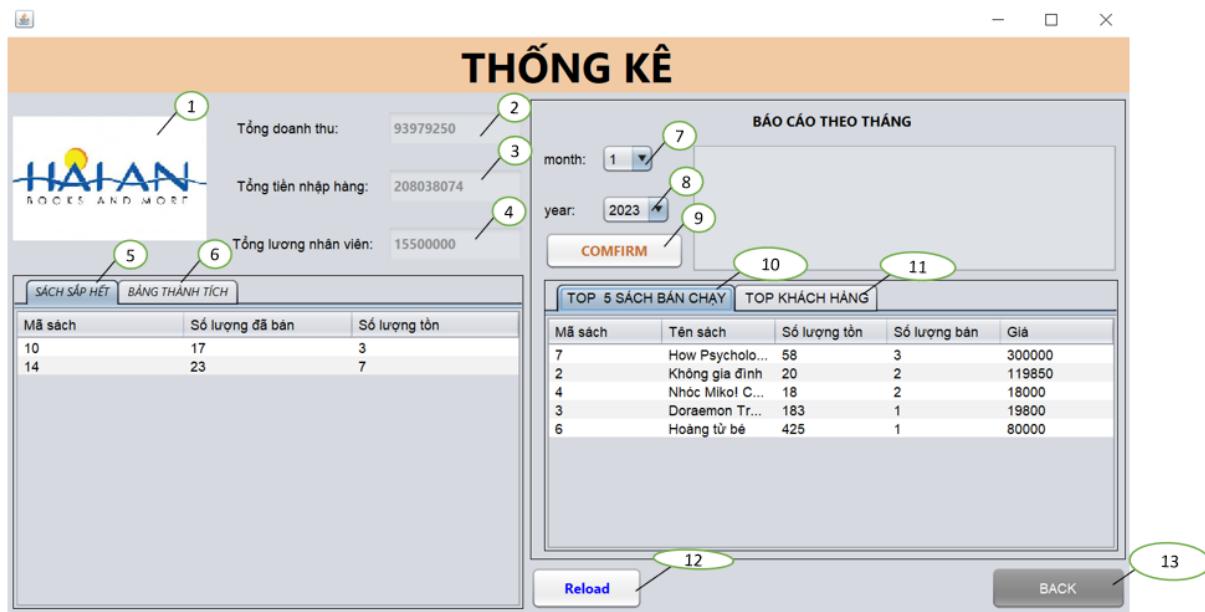
1.15. Giao diện trang quản lý nhà cung cấp



Mô tả giao diện

STT	Tên	Kiểu	Chức năng
1	Mã NCC	JTextField	Nhập mã nhà cung cấp
2	+	JButton	Nhấn để hiển thị mã nhà cung cấp mới nhất
3	Search	JButton	Nhập mã nhà cung cấp
4	Tên NCC	JTextField	Nhập tên nhà cung cấp
5	Địa chỉ	JTextField	Nhập địa chỉ
6	Số ĐT	JTextField	Nhập số điện thoại
7	INSERT	JButton	Nhấn để thực hiện chức năng thêm
8	DELETE	JButton	Nhấn để thực hiện chức năng xóa
9	UPDATE	JButton	Nhấn để thực hiện chức năng sửa
10	REFRESH	JButton	Nhấn để thực hiện chức năng làm mới
11	DANH SÁCH NHÀ CUNG CẤP	JTable	Hiển thị thông tin của các nhà cung cấp
12	TRỞ LẠI	JButton	Nhấn để trở lại trang quản lý

1.16. Giao diện trang thống kê



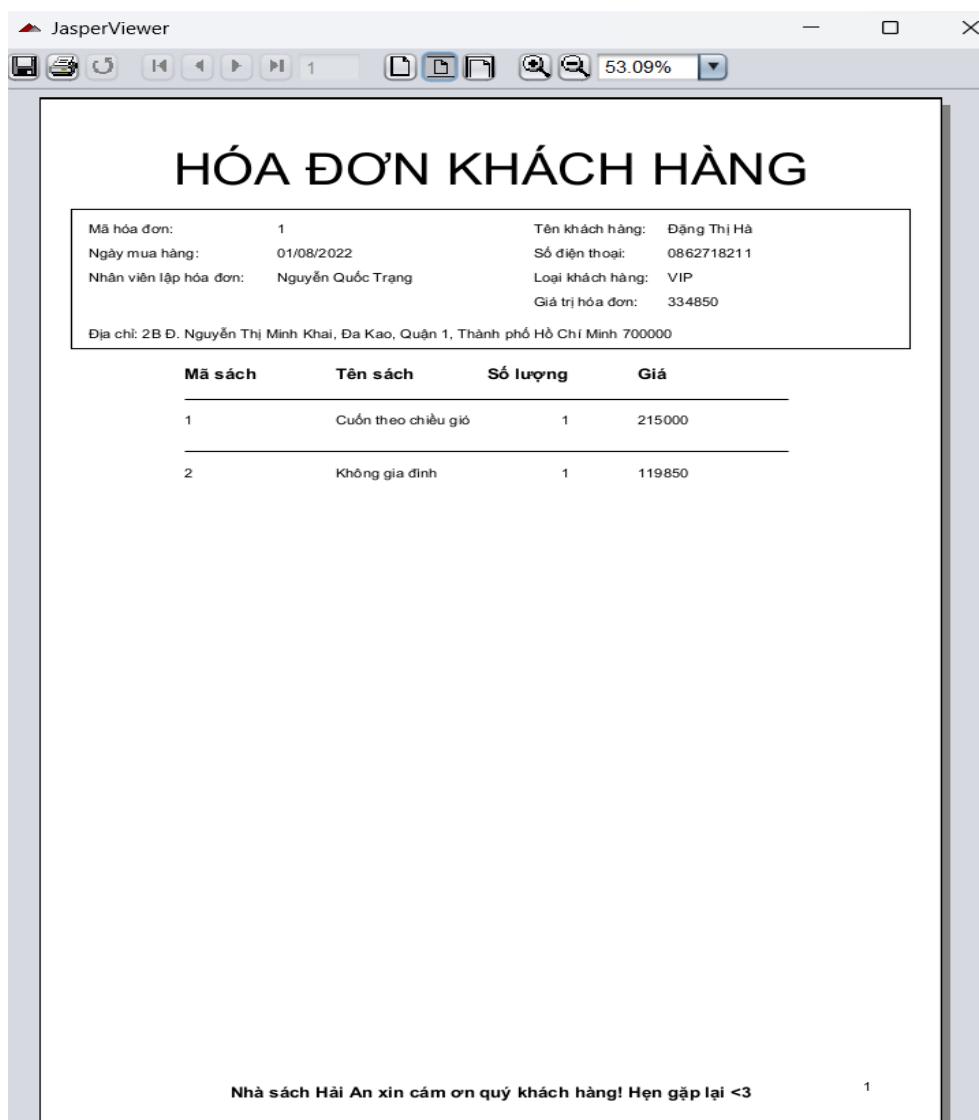
Mô tả giao diện

STT	Tên	Kiểu	Chức năng
1	Hình ảnh	JLabel	Hiển thị hình ảnh
2	Tổng doanh thu	JTextField	Hiển thị tổng doanh thu của hiệu sách
3	Tổng tiền nhập hàng	JTextField	Hiển thị tổng tiền nhập hàng của hiệu sách
4	Tổng lương nhân viên	JTextField	Hiển thị tổng lương của nhân viên
5	Sách sắp hết	JTabbedPane	Hiển thị danh sách các sách sắp hết
6	Bảng thành tích	JTabbedPane	Hiển thị xếp hạng của các nhân viên
7	Month	JComboBox	Nhấn để chọn tháng cần thống kê
8	Year	JComboBox	Nhấn để chọn năm cần thống kê
9	CONFIRM	JButton	Nhấn để xác nhận

10	TOP 5 SÁCH BÁN CHẠY	JTabbedPane	Hiển thị danh sách 5 sách bán chạy
11	TOP KHÁCH HÀNG	JTabbedPane	Hiển thị danh sách khách hàng mua nhiều nhất
12	Reload	JButton	Nhấn để làm mới lại
13	BACK	JButton	Nhấn để trở lại trang quản lý

2. Thiết kế Report

2.1. Giao diện report in hóa đơn



PHẦN V. KẾT LUẬN

1. Bảng phân công công việc

	Trạng	Vy	Phong
Bảng tóm tắt nội dung bằng tiếng anh		x	
Phát biểu bài toán		x	
Xác định yêu cầu			x
Phân tích thiết kế		x	x
Mô tả ràng buộc toàn vẹn	x		
Xây dựng giao tác			
Phần II.2 TRIGGER	2.1 → 2.5		x
	2.6 → 2.10		x
	2.11 → 2.15	x	
Xây dựng giao tác Procedure, Function		x	x
Xử lý đồng thời			
Lost update	TH 1	x	
	TH 2		x
	TH 3		x
Phantom Read	TH 1		x
	TH 2		x
	TH 3	x	
Non- Repeatable Read	TH 1		x
	TH 2		x
	TH 3	x	
Giao diện			
Thiết kế giao diện		x	
Viết chương trình Java	x		x

Tổng kết			
Kết luận báo cáo	X		
File word báo cáo			X
File ppt báo cáo		X	

2. Môi trường phát triển và môi trường triển khai

2.2. Môi trường phát triển

- Hệ điều hành: Microsoft Windows 10
- Hệ quản trị cơ sở dữ liệu: Oracle
- Công cụ xây dựng ứng dụng: NetBean
- Ngôn ngữ: Java

2.3. Môi trường triển khai

- Hệ điều hành: Microsoft Windows 7 trở lên
- Cần cài đặt hệ quản trị cơ sở dữ liệu Oracle

3. Kết quả đạt được

3.1. Kết quả đạt được

- Biết xây dựng và thiết kế cơ sở dữ liệu
- Biết xây dựng các Trigger, Store Procedure, Function
- Biết các trường hợp gây mất nhất quán dữ liệu, từ đó đưa ra hướng giải quyết các tình huống
- Biết cách xây dựng ứng dụng, giao diện, report
- Xây dựng hệ thống gồm có các chức năng sau:
 - Quản lý tài khoản
 - Quản lý nhân viên
 - Quản lý khách hàng
 - Quản lý kho
 - Quản lý phiếu nhập – chi tiết phiếu nhập

- Quản lý hóa đơn – chi tiết hóa đơn
- Quản lý nhà cung cấp
- Thống kê bán hàng
- Doanh thu bán hàng

3.2. Khó khăn

- Vì thời gian thực hiện đề tài còn giới hạn vì thế kết quả của đề tài còn một số hạn chế:
 - + Chưa thực hiện được chức năng đặt dịch vụ cho khách hàng
 - + Kỹ năng, kinh nghiệm làm đồ án chưa nhiều
 - + Chưa phải sản phẩm hoàn thiện nhất

4. Hướng phát triển

- Mở rộng thêm chức năng đặt dịch vụ cho khách hàng
- Phát triển ứng dụng đa ngôn ngữ
- Xây dựng ứng dụng trên web và các thiết bị di động

5. Tài liệu tham khảo

- Đồ án xây dựng hệ thống rạp chiếu phim ([Đồ án tham khảo](#))
- ChatGPT
- <https://www.studocu.com/vn/document/truong-dai-hoc-phenikaa/cong-nghe-thong-tin/nhom-5-du-an-he-thong-quan-li-nha-sach/27462652>
- [Slide PTTK tham khảo](#)
- [PTTK Hệ thống quản lý thư viện](#)
- Slide Bài thực hành số 4 do thầy Nguyễn Hồ Duy Tri cung cấp ([slide Bài TH4](#))
- Slide Bài thực hành số 6 thầy Nguyễn Hồ Duy Tri cung cấp ([Slide Bài TH6](#))

-
- https://www.programcreek.com/java-api-examples/?class=java.sql.Connection&method=TRANSACTION_SERIALIZABLE