**BỘ GIÁO DỤC VÀ ĐÀO TẠO**

**TRƯỜNG ĐH NGOẠI NGỮ TIN HỌC TP.HCM**

**KHOA CÔNG NGHỆ THÔNG TIN**

**------- 🙢 🙠 -------**

**BÁO CÁO ĐỒ ÁN MÔN HỆ QUẢN TRỊ CSDL**

ĐỀ TÀI:

**QUẢN LÝ MUA HÀNG**

**PHÂN HỆ   
YÊU CẦU BÁO GIÁ & MUA HÀNG**

**Giảng viên hướng dẫn :** Th.S Lê Minh Nguyện

**Sinh viên thực hiện :** Nguyễn Phát Nghị -16DH110117

Nguyễn Duy -16DH110097

**Lớp :** TT1601

**TP. HỒ CHÍ MINH, Ngày 24 tháng 12 năm 2019**

# *Mục Lục*

[*Mục Lục* 1](#_Toc28078298)

[*Danh sách hình ảnh* 2](#_Toc28078299)

[*Mở Đầu* 3](#_Toc28078300)

[1.Mục tiêu: 3](#_Toc28078301)

[2.Ý nghĩa: 3](#_Toc28078302)

[*Chương 1: Khảo sát hiện trạng và xác định yêu cầu* 4](#_Toc28078303)

[1.Hiện Trạng 4](#_Toc28078304)

[*Chương 2: Phân tích – thiết kế dữ liệu* 8](#_Toc28078305)

[1. Lược đồ cơ sở dữ liệu: 8](#_Toc28078306)

[2. Phân tích ràng buộc dữ liệu: 8](#_Toc28078307)

[*Chương 3: Cơ sở lý thuyết trong hệ quản trị CSDL* 13](#_Toc28078308)

[23.Mô hình client/server: 13](#_Toc28078309)

[24.Cấu trúc quản lý file vật lý trong SQL SERVER: 15](#_Toc28078310)

[25.Giao tác 16](#_Toc28078311)

[26.Xử lý đồng thời 17](#_Toc28078312)

[27.Phân quyền bảo mật 21](#_Toc28078313)

[*Chương 4: Triển khai thực nghiệm* 25](#_Toc28078314)

[1.Kết quả đạt được: 25](#_Toc28078315)

[2.Hướng phát triển: 30](#_Toc28078316)

[*Tài Liệu Tham Khảo* 31](#_Toc28078317)

# *Danh sách hình ảnh*

[Hình 1 Quỳ trình mua hàng 4](#_Toc28078320)

[Hình 2 Lược đồ CSDL 8](#_Toc28078321)

[Hình 3 Mô hình Client/Server 13](#_Toc28078322)

[Hình 4 Cấu trúc quản lý file vật lý 15](#_Toc28078323)

[Hình 5 Các cấp độ cô lập 19](#_Toc28078324)

[Hình 6 Cơ chế chứng thực 21](#_Toc28078325)

[Hình 7 Cơ chế phân quyền 22](#_Toc28078326)

[Hình 8 Danh sách yêu cầu báo giá 25](#_Toc28078327)

[Hình 9 Chi tiết yêu cầu báo giá 26](#_Toc28078328)

[Hình 10 Thêm yêu cầu báo giá 26](#_Toc28078329)

[Hình 11 Danh sách đơn mua hàng 27](#_Toc28078330)

[Hình 12 Thêm mới đơn mua hàng 28](#_Toc28078331)

[Hình 13 Thêm mới đơn mua hàng từ yêu cầu báo giá 28](#_Toc28078332)

[Hình 14 Thêm từ kế hoạch mua 29](#_Toc28078333)

# *Mở Đầu*

## 1.Mục tiêu:

Một quy trình mua hàng cơ bản gồm nhận yêu cầu mua hàng từ nội bộ công ty,  
gửi yêu cầu báo giá đến nhà cung cấp, tính toán đơn giá rồi tiến hành đặt và mua hàng. Nếu bạn quản lý tất cả các quá trình này chỉ bằng một nghiệp vụ thì có thể dễ dàng tìm hiểu về tiến độ công việc, và cũng không cần lặp đi lặp lại thực hiện cùng một nội dung công việc nữa..

## 2.Ý nghĩa:

Yêu cầu báo giá có thể kiểm tra trực quan báo giá đúng với từng nhà cùng cấp và có thể quyết định chọn mua với mức giá thích hợp nhất. Có thể chuyển báo giá đã lưu làm đơn đặt mua hàng và xử lý công việc nhập nhập phiếu đặt hàng một cách đơn giản nhất.

Nhập mặt hàng muốn mua và đơn giá yêu cầu cho nơi nhập hàng muốn yêu cầu báo giá và có thể tạo đơn theo yêu cầu báo giá. Sử dụng tính năng gửi E-mail là có thể gửi hàng loạt yêu cầu báo giá cho từng nhà cung cấp, sau đó người phụ trách của nhà cung cấp nhận yêu cầu báo giá thì có thể cập nhật giá và gửi lại. Cũng như có thể kiểm tra xem đã nhận được E-mail từ nhà cung cấp hay chưa.

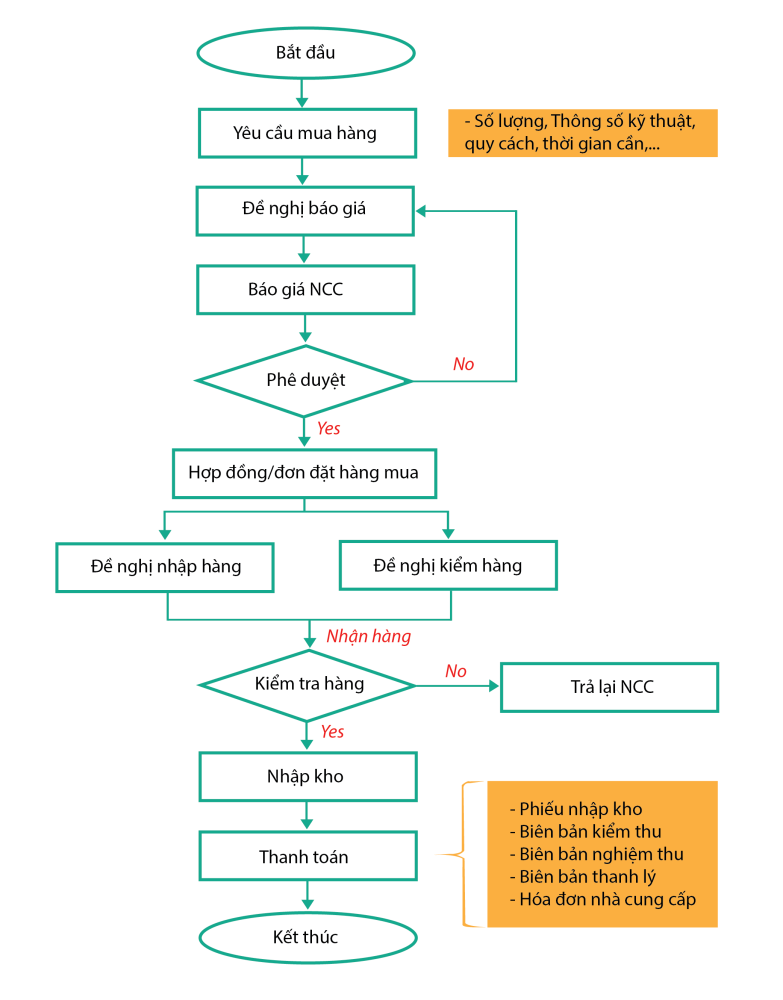
Có thể đối chiếu chi tiết đơn giá của từng mặt hạng theo nhà cung cấp cụ thể, từ đó chọn ra giá tốt nhất trong các báo giá mà nhà cung cấp gửi đến.

Sau khi có yêu cầu báo giá, có thể tạo đơn mua hàng một cách đơn giản thông qua yêu cầu báo giá đã được phản hồi dựa trên E-mail.

# *Chương 1: Khảo sát hiện trạng và xác định yêu cầu*

## Hiện Trạng

Mỗi doanh nghiệp hoạt động sản xuất kinh doanh khác nhau sẽ có quy trình mua hàng khác nhau, nhưng về cơ bản quy trình mua hàng của các công ty thường theo các bước sau:



Hình 1 Quỳ trình mua hàng

* **Lập “Yêu cầu mua hàng”**

- Các phòng ban khi có nhu cầu mua hàng hóa, dụng cụ, tài sản, nguyên vật liệu… sẽ làm yêu cầu gửi cho phòng mua hàng để tiến hành mua hàng đáp ứng nhu cầu sản xuất kinh doanh, các phiếu yêu cầu này do trưởng phòng hay người có trách nhiệm đã duyệt.

- Khi có “Yêu cầu mua hàng”, phòng mua hàng tiến hành phân công cho nhân viên mua hàng, tìm kiếm nhà cung cấp, báo giá các mặt hàng.

* **Lập “Đề nghị báo giá”**

- Từ “Yêu cầu mua hàng”, Phòng mua hàng tiến hành lập “Đề nghị báo giá” gửi các nhà cung cấp truyền thống hoặc các nhà cung cấp mới, đã tìm kiếm được theo các điều kiện các phòng ban đã yêu cầu.

* **Theo dõi “Báo giá của NCC”**

- Nhận các báo giá từ các nhà cung cấp

- Đánh giá nhà cung cấp có khả năng đáp ứng tốt nhất các tiêu chí và điều kiện đã xây dựng

- Lựa chọn nhà cung cấp cuối cùng

- Căn cứ vào báo giá và yêu cầu được phê duyệt, Phòng Mua hàng tiến hành lập và theo dõi “Hợp đồng / Đơn đặt hàng mua”. Tùy theo đặc thù của doanh nghiệp, tính chất mua bán mà lập Đơn hàng hay Hợp đồng mua hay là chứng từ xác nhận.

* **Phê duyệt Báo giá của Nhà cung cấp**

- Các báo giá của NCC sẽ được BLĐ xét duyệt dựa trên các thông tin sau:

+ So sánh báo giá và các điều kiện mua hàng cùng một mặt hàng của các nhà cung cấp khác nhau

+ So sánh báo giá mới với báo giá cũ cho cùng một mặt hàng của các nhà cung cấp khác nhau

* **Lập “Hợp đồng/ Đơn hàng mua”**

- Khi chọn được nhà cung cấp sẽ tiến hành lập hợp đồng, trên hợp đồng ghi nhận thông tin của báo giá, điều khoản thanh toán, lịch giao hàng. Gửi đơn đặt hàng hoặc hợp đồng cho nhà cung cấp và thực hiện ký kết đơn hàng hoặc hợp đồng giữa hai bên.

- Trường hợp việc ký hợp đồng có nhiều lần thực hiện, thì sẽ lập đơn hàng cho từng lần thực hiện. Thông tin trên đơn hàng cũng tương tự các điều khoản trên hợp đồng

- Chuyển “Hợp đồng/ Đơn hàng mua” cho các bộ phận liên quan theo dõi: Kế toán căn cứ thanh toán, theo dõi công nợ, Bộ phận kho theo dõi quá trình nhập hàng về kho.

* **Lập “Đề nghị nhập hàng” và “Đề nghị kiểm hàng”**

- Để chuẩn bị cho khâu nhập hàng theo lịch, phòng mua hàng lập “Đề nghị nhập hàng” và “Đề nghị kiểm hàng” gửi các phòng ban liên quan theo dõi thực hiện.

* **Nhập kho**

- Khi hàng được vận chuyển đến kho, các thông tin trên Hợp đồng/ Đơn hàng mua (số lượng, thông số kỹ thuật, quy cách…) sẽ làm căn cứ để bộ phận Kho kiểm tra. Các mặt hàng không đạt đúng tiêu chuẩn sẽ phản hồi cho Phòng mua hàng và Phòng mua hàng tiếp nhận và thực hiện các bước trả lại NCC. Các mặt hàng đạt tiêu chuẩn sẽ được tiến hành nhập kho.

- Khi hàng nhập kho, bộ phận kho sẽ nhập số lượng, còn phòng mua hàng sẽ bổ sung thông tin về giá

* **Thanh toán**

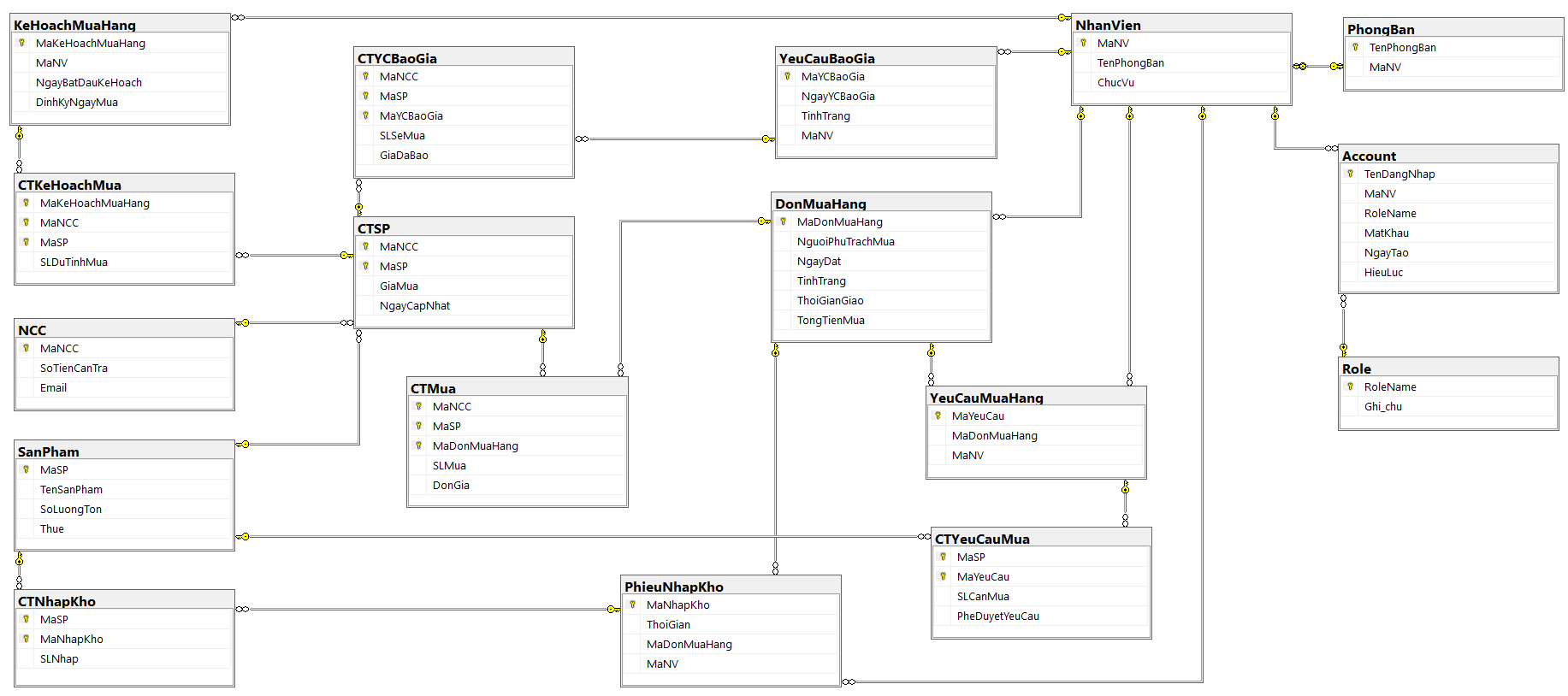
- Căn cứ vào điều khoản trên hợp đồng và các giấy tờ biên bản liên quan, phòng mua hàng sẽ lập bộ hồ sơ thanh toán

- Phòng Kế toán tiếp nhận và kiểm tra lại, nếu hợp lệ thì tiến hành thanh toán cho Nhà cung cấp nếu không phản hồi lại phòng Mua hàng bổ sung/ chỉnh sửa.

1. **Yêu Cầu**
   1. Yêu cầu báo giá
      * Yêu cầu báo giá trước khi đến đặt hàng đến chỗ mua hàng
      * Gửi yêu cầu báo giá hàng loạt đến nhiều nhà cung cấp
      * Có thể xem được đã nhận được Email yêu cầu từ nhà cung cấp
      * Có thể đặt giới hạn trong thời gian báo giá
      * Tạo ô ghi chú trong biểu mẫu yêu cầu báo giá
      * So sánh đơn giá của từng báo giá
      * Nhập mặt hàng muốn mua và đơn giá yêu cầu báo giá cho nơi nhập hàng và toạ đơn yêu cầu báo giá
      * Người phụ trách của nhà cung cấp đã nhận được yêu cầu báo giá thì có thể cập nhật giá cả và gửi lại theo thời gian thực
      * Có thể chọn và cập nhật giá tốt nhất trong các báo giá mà nhà cung cấp gửi đến đơn giá đã cho thấy.
      * Có thể liên kết với yêu cầu báo giá của các dữ liệu đã nhập ở kế hoạch đặt mua
      * Thông qua kiểm tra báo giá có thể cập nhật báo giá tốt nhất vào đơn đặt mua hàng
   2. Mua hàng
      * Có thể kiểm tra số lượng chưa nhập kho so với số lượng đặt
      * Có thể tiến hành mua riêng từng mặt hàng
      * Khi lập phiếu mua hàng số lượng tồn kho cập nhật tự động
      * Có thể kiểm tra số lượng và mặt hàng chưa tạo phiếu nhập với đơn mua hàng dựa trên tính năng liên kết với đơn đặt mua
      * Thông qua tính năng áp dụng số lượng còn lại thì có thể nhập bổ sung các mặt hàng hoặc số lượng chưa được nhập từ những đơn hàng trước
      * Dễ dàng nắm bắt tình huống tiến hành mua đối chiếu đặt mua
      * Có thể bao gồm mua hàng thực tế của từng người phụ trách mua thì có thể kiểm tra và và cập nhật tình hình mua hàng đa dạng.

# *Chương 2: Phân tích – thiết kế dữ liệu*

## 1. Lược đồ cơ sở dữ liệu:



Hình 2 Lược đồ CSDL

## 2. Phân tích ràng buộc dữ liệu:

1. Ràng buộc trên *1 quan hệ*
   1. Số tiền cần trả trong Nhà Phân Phối luôn dương

|  |  |  |  |
| --- | --- | --- | --- |
|  | Insert | Delete | Update |
| NhaPhanPhoi | + | - | + (SoTienCanTra) |

* 1. Định dạng Email của Nhà Phân Phối phải có ký tự ‘@’

|  |  |  |  |
| --- | --- | --- | --- |
|  | Insert | Delete | Update |
| NhaPhanPhoi | + | - | + (Email) |

* 1. Số lượng tồn của Sản Phẩm luôn dương

|  |  |  |  |
| --- | --- | --- | --- |
|  | Insert | Delete | Update |
| SanPham | + | - | + (SoLuongTon) |

* 1. Giá mua của chi tiết sản phẩm luôn dương

|  |  |  |  |
| --- | --- | --- | --- |
|  | Insert | Delete | Update |
| CTSP | + | - | + (GiaMua) |

* 1. Tổng tiền mua của đơn hàng luôn dương

|  |  |  |  |
| --- | --- | --- | --- |
|  | Insert | Delete | Update |
| DonMuaHang | + | - | + (TongTienMua) |

* 1. Số lượng dự tính mua của Chi Tiết Kế Hoạch Mua luôn dương

|  |  |  |  |
| --- | --- | --- | --- |
|  | Insert | Delete | Update |
| CTKeHoachMua | + | - | + (SLDuTinhMua) |

* 1. Số lượng cần mua của Chi Tiết Yêu Cầu Mua luôn dương

|  |  |  |  |
| --- | --- | --- | --- |
|  | Insert | Delete | Update |
| CTYeuCauMua | + | - | + (SLCanMua) |

* 1. Số lượng nhập của Chi Tiết Nhập Kho luôn dương

|  |  |  |  |
| --- | --- | --- | --- |
|  | Insert | Delete | Update |
| CTNhapKho | + | - | + (SLNhap) |

* 1. Số lượng sẽ mua và Giá Đã Báo của CTYC Báo Giá luôn dương

|  |  |  |  |
| --- | --- | --- | --- |
|  | Insert | Delete | Update |
| CTYCBaoGia | + | - | + (SLSeMua, GiaDaBao) |

1. Ràng buộc trên *nhiều quan hệ*
   1. TenPhongBan trong bảng NhanVien phải tồn tại trong bảng PhongBan

|  |  |  |  |
| --- | --- | --- | --- |
|  | Insert | Delete | Update |
| NhanVien | + | - | + (TenPhongBan) |
| PhongBan | - | + | + (TenPhongBan) |

* 1. MaNV trong bảng PhongBan phải tồn tại trong bảng NhanVien

|  |  |  |  |
| --- | --- | --- | --- |
|  | Insert | Delete | Update |
| PhongBan | + | - | + (MaNV) |
| NhanVien | - | + | + (MaNV) |

* 1. MaNV trong bảng Account phải tồn tại trong bảng NhanVien

|  |  |  |  |
| --- | --- | --- | --- |
|  | Insert | Delete | Update |
| Account | + | - | + (MaNV) |
| NhanVien | - | + | + (MaNV) |

* 1. RoleName trong bảng Account phải tồn tại trong bảng Role

|  |  |  |  |
| --- | --- | --- | --- |
|  | Insert | Delete | Update |
| Account | + | - | + (RoleName) |
| Role | - | + | +(RoleName) |

* 1. MaDonMuaHang trong bảng YeuCauMuaHang phải tồn tại trong bảng DonMuaHang

|  |  |  |  |
| --- | --- | --- | --- |
|  | Insert | Delete | Update |
| YeuCauMuaHang | + | - | +(MaDonMuaHang) |
| DonMuaHang | - | + | +(MaDonMuaHang) |

* 1. MaNV trong bảng YeuCauMuaHang phải tồn tại trong bảng NhanVien

|  |  |  |  |
| --- | --- | --- | --- |
|  | Insert | Delete | Update |
| YeuCauMuaHang | + | - | + (MaNV) |
| NhanVien | - | + | +(MaNV) |

* 1. NguoiPhuTrachMua (MaNV) trong bảng DonMuaHang phải tồn tại trong bảng NhanVien

|  |  |  |  |
| --- | --- | --- | --- |
|  | Insert | Delete | Update |
| DonMuaHang | + | - | + (NguoiPhuTrachMua) |
| NhanVien | - | + | +(MaNV) |

* 1. MaDonMuaHang trong bảng CTMua phải tồn tại trong DonMuaHang

|  |  |  |  |
| --- | --- | --- | --- |
|  | Insert | Delete | Update |
| CTMua | + | - | +(MaDonMuaHang) |
| DonMuaHang | - | + | +(MaDonMuaHang) |

* 1. MaNPP và MaSP trong bảng CTMua phải tồn tại trong bảng CTSP

|  |  |  |  |
| --- | --- | --- | --- |
|  | Insert | Delete | Update |
| CTMua | + | - | +(MaNPP, MaSP) |
| CTSP | - | + | +(MaNPP, MaSP) |

* 1. MaSP trong bảng CTYeuCauMua phải tồn tại trong bảng SanPham

|  |  |  |  |
| --- | --- | --- | --- |
|  | Insert | Delete | Update |
| CTYeuCauMua | + | - | +(MaSP) |
| SanPham | - | + | +(MaSP) |

* 1. MaYeuCau trong CTYeuCauMua phải tồn tại trong bảng YeuCauMuaHang

|  |  |  |  |
| --- | --- | --- | --- |
|  | Insert | Delete | Update |
| CTYeuCauMua | + | - | +(MaYeuCau) |
| YeuCauMuaHang | - | + | +(MaYeuCau) |

* 1. MaDonMuaHang trong bảng PhieuNhapKho phải tồn tại trong bảng DonMuaHang

|  |  |  |  |
| --- | --- | --- | --- |
|  | Insert | Delete | Update |
| PhieuNhapKho | + | - | +(MaDonMuaHang) |
| DonMuaHang | - | + | +(MaDonMuaHang) |

* 1. MaSP trong bảng CTNhapKho phải tồn tại trong bảng SanPham

|  |  |  |  |
| --- | --- | --- | --- |
|  | Insert | Delete | Update |
| CTNhapKho | + | - | +(MaSP) |
| SanPham | - | + | +(MaSP) |

* 1. MaNhapKho trong bảng CTNhapKho phải tồn tại trong bảng PhieuNhapKho

|  |  |  |  |
| --- | --- | --- | --- |
|  | Insert | Delete | Update |
| CTNhapKho | + | - | +(MaNhapKho) |
| PhieuNhapKho | - | + | +(MaNhapKho) |

* 1. MaNV trong bảng YeuCauBaoGia phải tồn tại trong bảng NhanVien

|  |  |  |  |
| --- | --- | --- | --- |
|  | Insert | Delete | Update |
| YeuCauBaoGia | + | - | +(MaNV) |
| NhanVien | - | + | +(MaNV) |

* 1. MaNPP và MaSP trong bảng CTYCBaoGia phải tồn tại trong bảng CTSP

|  |  |  |  |
| --- | --- | --- | --- |
|  | Insert | Delete | Update |
| CTYCBaoGia | + | - | +(MaNPP, MaSP) |
| CTSP | - | + | +(MaNPP, MaSP) |

* 1. MaYCBaoGia trong bảng CTYCBaoGia phải tồn tại trong bảng YeuCauBaoGia

|  |  |  |  |
| --- | --- | --- | --- |
|  | Insert | Delete | Update |
| CTYCBaoGia | + | - | +(MaYCBaoGia) |
| YeuCauBaoGia | - | + | +(MaYCBaoGia) |

* 1. MaNPP trong bảng CTSP phải tồn tại trong bảng NhaPhanPhoi

|  |  |  |  |
| --- | --- | --- | --- |
|  | Insert | Delete | Update |
| CTSP | + | - | +(MaNPP) |
| NhaPhanPhoi | - | + | +(MaNPP) |

* 1. MaSP trong bảng CTSP phải tồn tại trong bảng SanPham

|  |  |  |  |
| --- | --- | --- | --- |
|  | Insert | Delete | Update |
| CTSP | + | - | +(MaSP) |
| SanPham | - | + | +(MaSP) |

* 1. MaNV trong bảng KeHoachMuaHang phải tồn tại trong bảng NhanVien

|  |  |  |  |
| --- | --- | --- | --- |
|  | Insert | Delete | Update |
| KeHoachMuaHang | + | - | +(MaNV) |
| NhanVien | - | + | +(MaNV) |

* 1. MaKeHoachMuaHang trong bảng CTKeHoachMua phải tồn tại trong bảng KeHoachMuaHang

|  |  |  |  |
| --- | --- | --- | --- |
|  | Insert | Delete | Update |
| CTKeHoachMua | + | - | +(MaKeHoachMuaHang) |
| KeHoachMuaHang | - | + | +(MaKeHoachMuaHang) |

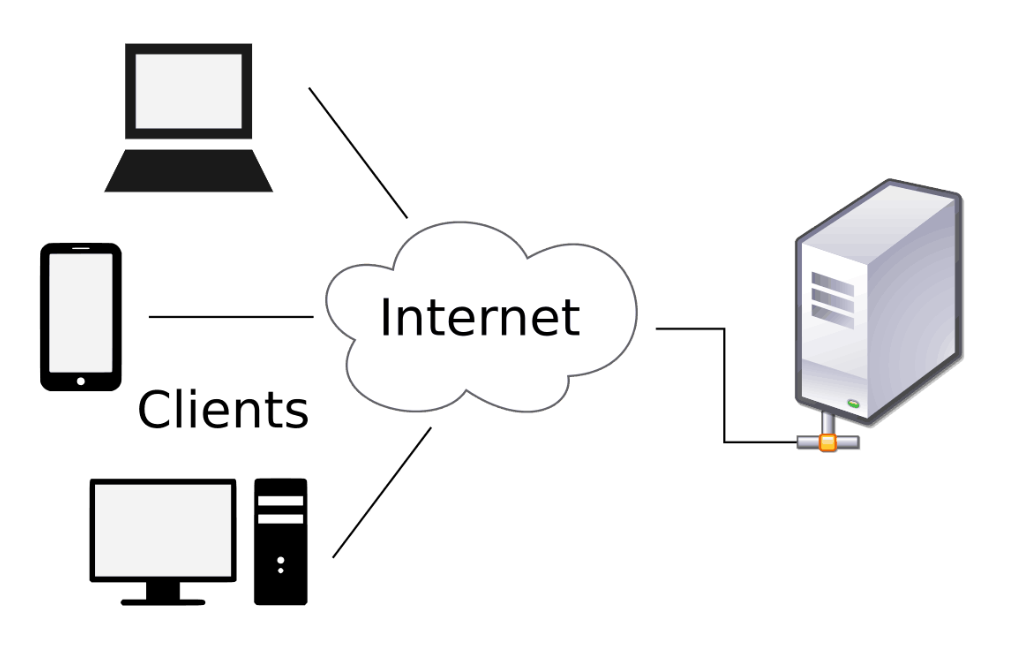
* 1. MaNPP và MaSP trong bảng CTKeHoachMua phải tồn tại trong bảng CTSP

|  |  |  |  |
| --- | --- | --- | --- |
|  | Insert | Delete | Update |
| CTKeHoachMua | + | - | +(MaNPP, MaSP) |
| CTSP | - | + | +(MaNPP, MaSP) |

# *Chương 3: Cơ sở lý thuyết trong hệ quản trị CSDL*

## Mô hình client/server:

Trong mô hình client server thì máy khách là các máy tính, các thiết bị điện tử như máy in, máy fax,….các máy khách client gửi yêu cầu đến máy chủ server. máy chủ server tiếp nhận yêu cầu, xử lý các yêu cầu đó và trả về kết quả



Hình 3 Mô hình Client/Server

**Ưu điểm của mô hình Client-server**

Ưu điểm của mô hình client server là với mô hình client server thì mọi thứ dường như đều nằm trên bàn của người sử dụng, nó có thể truy cập dữ liệu từ xa (bao gồm các công việc như gửi và nhận file, tìm kiếm thông tin,…) với nhiều dịch vụ đa dạng mà mô hình cũ không thể làm được. Mô hình client/server cung cấp một nền tảng lý tưởng cho phép tích hợp các kỹ thuật hiện đại như mô hình thiết kế hướng đối tượng, hệ chuyên gia, hệ thông tin địa lý (GIS)…

**Nhược điểm của mô hình Client-server**

Một trong những vấn đề nảy sinh trong mô hình này đó là tính an toàn và bảo mật thông tin trên mạng. Do phải trao đổi dữ liệu giữa 2 máy ở 2 khu vực khác nhau cho nên dễ dàng xảy ra hiện tượng thông tin truyền trên mạng bị lộ.

**Mô hình Client-server hoạt động như thế nào**

– Client Trong mô hình client/server, người ta còn định nghĩa cụ thể cho một máy client là một máy trạm mà chỉ được sử dụng bởi 1 người dùng với để muốn thể hiện tính độc lập cho nó. Máy client có thể sử dụng các hệ điều hành bình thường như Win9x, DOS, OS/2… Bản thân mỗi một client cũng đã được tích hợp nhiều chức năng trên hệ điều hành mà nó chạy, nhưng khi được nối vào một mạng LAN, WAN theo mô hình client/server thì nó còn có thể sử dụng thêm các chức năng do hệ điều hành mạng (NOS) cung cấp với nhiều dịch vụ khác nhau (cụ thể là các [dịch vụ máy chủ](https://vdodata.vn/thue-may-chu-chay-game-server/) trên mạng này cung cấp), ví dụ như nó có thể yêu cầu lấy dữ liệu từ một server hay gửi dữ liệu lên server đó…

– Thực tế trong các ứng dụng của mô hình client/server, các chức năng hoạt động chính là sự kết hợp giữa client và server với sự chia sẻ tài nguyên, dữ liệu trên cả hai máy Vai trò của client Trong mô hình client/server, client được coi như là người sử dụng các dịch vụ trên mạng do một hoặc nhiều máy chủ cung cấp và server được coi như là người cung cấp dịch vụ để trả lời các yêu cầu của các clients. Điều quan trọng là phải hiểu được vai trò hoạt động của nó trong một mô hình cụ thể, một máy client trong mô hình này lại có thể là server trong một mô hình khác. Ví dụ cụ thể như một máy trạm làm việc như một client bình thường trong mạng LAN nhưng đồng thời nó có thể đóng vai trò như một máy in chủ (printer server) cung cấp dịch vụ in ấn từ xa cho nhiều người khác (clients) sử dụng. Client được hiểu như là bề nổi của các dịch vụ trên mạng, nếu có thông tin vào hoặc ra thì chúng sẽ được hiển thị trên máy client.

– Server được định nghĩa như là một máy tính nhiều người sử dụng (multiuser computer). Vì một server phải quản lý nhiều yêu cầu từ các client trên mạng cho nên nó hoạt động sẽ tốt hơn nếu hệ điều hành của nó là đa nhiệm với các tính năng hoạt động độc lập song song với nhau như hệ điều hành UNIX, WINDOWS…

– Server cung cấp và điều khiển các tiến trình truy cập vào tài nguyên của hệ thống. Các ứng dụng chạy trên server phải được tách rời nhau để một lỗi của ứng dụng này khô ng làm hỏng ứng dụng khác. Tính đa nhiệm đảm bảo một tiến trình không sử dụng toàn bộ tài nguyên hệ thống.

**Vai trò của Server**

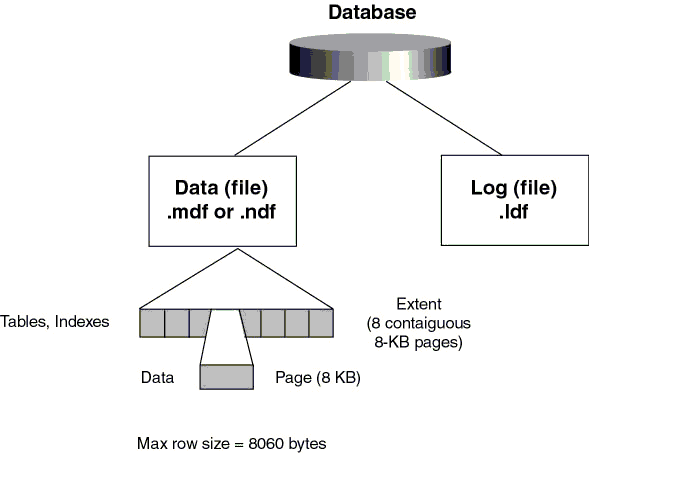
Như chúng ta đã bàn ở trên, server như là một nhà cung cấp dịch vụ cho các client yêu cầu tới khi cần, các dịch vụ như cơ sở dữ liệu, in ấn, truyền file,… Các ứng dụng server cung cấp các dịch vụ mang tính chức năng để hỗ trợ cho các hoạt động trên các máy clients có hiệu quả hơn. Sự hỗ trợ của các dịch vụ này có thể là toàn bộ hoặc chỉ một phần thông qua IPC. Để đảm bảo tính an toàn trên mạng cho nên server này còn có vai trò như là một nhà quản lý toàn bộ quyền truy cập dữ liệu của các máy clients, nói cách khác đó là vai trò quản trị mạng. Có rất nhiều cách hiện nay nhằm quản trị có hiệu quả, một trong những cách đang được sử dụng là dùng tên Login và mật khẩu.

## Cấu trúc quản lý file vật lý trong SQL SERVER:

Mỗi một database trong SQL Server đều chứa ít nhất một data file chính (primary), có thể có thêm một hay nhiều data file phụ (Secondary) và một transaction log file.

* **Primary data file** (thường có phần mở rộng **.mdf**) : đây là file chính chứa data và những system tables.
* **Secondary data file** (thường có phần mở rộng **.ndf**) : đây là file phụ thường chỉ sử dụng khi database được phân chia để chứa trên nhiều dĩa.
* **Transaction log file** (thường có phần mở rộng **.ldf**) : đây là file ghi lại tất cả những thay đổi diễn ra trong một database và chứa đầy đủ thông tin để có thể roll back hay roll forward khi cần.

Data trong SQL Server được chứa thành từng **Page** 8KB và 8 page liên tục tạo thành một **Extent** như sau:



Hình 4 Cấu trúc quản lý file vật lý

Trước khi SQL Server muốn lưu data vào một table nó cần phải dành riêng một khoảng trống trong data file cho table đó. Những khoảng trống đó chính là các extents. Có 2 loại Extents: **Mixed Extents** (loại hỗn hợp) dùng để chứa data của nhiều tables trong cùng một Extent và **Uniform Extent** (loại thuần nhất) dùng để chứa data của một table.

## Giao tác

1. **Khái niệm giao tác**

**Một giao tác (transaction)** là một hành động hay một chuỗi các hành động truy cập vào CSDL hoặc làm thay đổi nội dung CSDL, giao tác được đưua ra bởi một người sử dụng hay chương trình ứng dụng.

Như vậy một giao tác là một đơn vị công việc trên CSDL xét trên tính logic. Một giao tác có thể là toàn bộ một chương trình (viết trong ngôn ngữ thao tác dữ liệu bậc cao hay trong ngôn ngữ lập trình như SQL, COBOL,...), nó cũng có thể là một phần của chương trình hay một câu lệnh đơn lẻ như câu lệnh INSERT, UPDATE.

Trong ngữ cảnh CSDL, có thể xem sự thực hiện một chương trình ứng dụng như một dãy các **giao tác** và xen kẽ giữa chúng là những thao tác không đòi hỏi xử lí CSDL (thao tác CPU). Để cho việc tiện theo dõi, mình sẽ minh họa hai lược đồ quan hệ **Products** và **Categories** đơn giản như sau:

**Products** (id, category\_id, name, price)

**Category** (id, name)

* Lược đồ **Products** gồm các thuộc tính về mã sản phẩm (**id**), mã thể loại (**category\_id**), tên sản phẩm (**name**) và giá của sản phẩm (**price**)
* Lược đồ **Categories** dùng để tham chiếu mã thể loại ứng với từng sản phẩm bao gồm các thuộc tính về mã thể loại (**id**), tên thể loại (**name**)

Một giao tác đơn giản là nâng giá của một sản phẩm có mã số là x*x* thêm 10$%$ giá hiện tại của sản phẩm đó. Chúng ta có thể viết giao tác này như sau:

Read(product.id = x, price)

New\_price = price \* 1.1

Write(product.id = x, new\_price)

Ở đây chúng ta quy ước dùng **read(x)** để chỉ thao tác đọc một mục dữ liệu có tên là x*x* ở CSDL vào một biến của chương trình. Để cho đơn giản việc trình bày chúng ta coi như biến chương trình này cũng có tên là x*x*. Tương tự như vậy **write(x)** dùng để chỉ thao tác ghi giá trị biến x*x* của chương trình vào mục dữ liệu x*x* của CSDL.

Việc thưc hiện một lệnh **read(x)** sẽ bao gồm các bước sau:

* Tìm địa chỉ của khối có chứa mục dữ liệu x*x*
* Sao chép khối đó vào vùng đệm của bộ nhớ chính (nếu nó chưa có sẵn trong một vùng đệm của bộ nhớ chính)
* Sao chép mục dữ liệu x*x* từ vùng đệm vào biến có tên là x*x* của chương trình

Việc thưc hiện một lệnh **write(x)** sẽ bao gồm các bước sau:

* Tìm địa chỉ của khối có chứa mục dữ liệu x*x*
* Sao chép khối đó vào vùng đệm của bộ nhớ chính (nếu nó chưa có sẵn trong một vùng đệm của bộ nhớ chính)
* Sao chép mục dữ liệu x*x* từ biến chương trình có tên là x*x* vào chỗ chính xác của nó trong vùng đệm
* Lưu trữ khối vừa cập nhập trong vùng đệm vào lại thiết bị nhớ để sau này có thể truy xuất (có thể lưu ngay tức thời hoặc tại thời điểm muộn hơn)

Trong ví dụ trên lệnh **read(**[**product.id**](http://product.id/)**= x, price)** dùng để chỉ thao tác đọc dữ liệu ở thuộc tính ***price*** của bản ghi có giá trị khóa chính id=x, lệnh **write(**[**product.id**](http://product.id/)**= x, new\_price)** dùng để chỉ giao tác ghi lại giá trị của **new\_price** vào cho bản ghi có khóa chính là x. Giữa hai thao tác đọc và ghi dữ liệu nới trên có một thao tác không truy cập CSDL, đó là thao tác new\_price = price \* 1.1.

1. **Các tính chất của giao tác**
   * **Tính nguyên tố (Atomicity):** toàn bộ thao tác trong cùng một giao tác đều được thực hiện hoặc không một thao tác nào trong chúng được thực hiện. Một giao tác là một đơn vị công việc không thể phân chia.
   * **Tính nhất quán (Consistency):** một giao tác phải chuyển CSDL từ một trạng thái nhất quán này sang một trạng thái nhất quán khác.
   * **Tính cô lập (Isolation):** Các giao tác phải được thực hiện một cách độc lập với nhau. Nói một cách khác những tác động của giao tác này sẽ không thể thấy được đối với những giao tác khác, khi giao tác này chưa thực hiện xong hoàn toàn.
   * **Tính bền vững (Durability):** những thay đổi của CSDL do tác động của một giao tác thành công là bền vững, không bị mất đi.

## Xử lý đồng thời

**Tại sao có tranh chấp dữ liệu?**

* Các tranh chấp dữ liệu sẽ phát sinh khi cùng lúc có nhiều người dùng đọc và cập nhật cùng dữ liệu giống nhau
* Sử dụng một trong các kỹ thuật sau để quản lý việc nhiều người dùng cùng cập nhật:
  + Pessimistic concurrency
  + Optimistic concurrency
  + "Last in wins"

**Pessimistic concurrency**

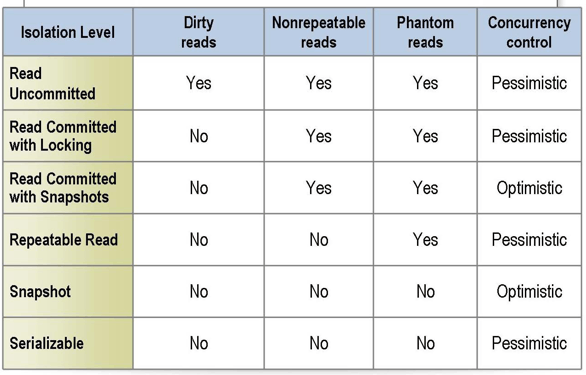
* Khóa ngay tại thời điểm đọc
  + Ví dụ: khi user A đọc một mẩu tin thì database engine lập tức khóa ngay mẩu tin này
  + Các user khác không thể cập nhật được mà phải chờ
  + Khi user A kết thúc cập nhật thì database engine mới mở khóa và các user khác mới có thể cập nhật được
* Đặc điểm
  + Bảo đảm tại một thời điểm chỉ cho phép có một cập nhật
  + Thích hợp với các ứng dụng có dữ liệu quan trọng cần thay đổi tập trung. Ví dụ các ứng dụng cho phép đặt chổ
  + Thời gian chiếm giữ càng ngắn càng tốt
  + Không thích hợp với mô hình disconnect vì các kết nối được duy trì đủ để đọc hoặc cập nhật => database engine không thể khóa dữ liệu trong thời gian dài được

**Optimistic concurrency**

* Khóa tại thời điểm ghi
  + Ví dụ: khi các user A và B cùng đọc một mẩu tin thì database engine không khóa mẩu tin này. Nhưng:
  + Giả sử user A sửa và ghi thành công (trước user B)
  + Sau đó thì đến user B ghi, database engine kiểm tra và phát hiện mẩu tin đã bị sửa trước đó => user B ghi không thành công (First in wins)
* Đặc điểm
  + Tại một thời điểm có thể có nhiều người dùng cập nhật
  + Thích hợp với các ứng dụng có dữ liệu ít thay đổi tập trung
  + Không hao tốn chi phí cho việc khóa dữ liệu
  + Thích hợp với mô hình disconnect vì người dùng có thể làm việc lâu dài với dữ liệu đã cache và sau khi kết thúc làm việc thì cập nhật lại CSDL
  + ADO.NET cho phép chọn lựa “Last in wins” hoặc “First in wins”

**Các cấp độ cô lập (isolation level)**

* Khi tạo Transaction có thể xác định cấp độ cô lập để xác lập ảnh hưởng giữa các Transaction
* Các Transaction cùng truy xuất dữ liệu giống nhau tại cùng thời điểm có thể gây ra các lỗi dị biệt như:
* **Dirty read:** đọc dữ liệu chưa được Commit
  + Transaction thứ nhất (T1) đang cập nhật nhưng chưa Commit
  + T2 có thể đọc thấy dữ liệu chưa được Commit của T1
* **Non-repeatable read:** đọc không nhất quán
  + T1 đang đọc dữ liệu đã được Commit
  + T2 có thể cập nhật dữ liệu này
  + T1 đọc lại và thấy có sự thay đổi
* **Phantom read:** đọc không bình thường
  + T1 đang đọc dữ liệu đã được Commit và dữ liệu được đọc theo điều kiện
  + T2 có thể cập nhật dữ liệu liên quan đến điều kiện (mà T2 đang đọc)
  + T1 đọc lại và thấy có sự thay đổi



Hình 5 Các cấp độ cô lập

**Read Uncommited**

* Là cấp độ cô lập thấp nhất trong giao tác
* Các thao tác đọc không yêu cầu khóa dữ liệu => có thể đọc thấy dữ liệu đang bị thay đổi trong các giao tác khác
* Read Uncommited chỉ bảo đảm dữ liệu đọc không bị hỏng vật lý

**Read Commited**

* Là cấp độ cô lập mặc định trong SQL Server 2005
* Các thao tác đọc sẽ bị treo (dead lock) khi dữ liệu đang bị thay đổi bởi các giao tác khác
* Khắc phục dị biệt Dirty Read

**Repeatable Read**

* Giao tác sử dụng cấp độ cô lập này sẽ khóa các dữ liệu mà nó đang truy vấn
* Các thao tác khác sẽ không cập nhật được dữ liệu đang bị khóa (dead lock)
* Khắc phục dị biệt Dirty Read và Repeatable Read

**Serializable**

* Được xem là mức cô lập cao nhất
* Giao tác sử dụng cấp độ cô lập này sẽ khóa các dữ liệu mà nó đang làm việc (có thể khóa toàn bộ table)
* Khắc phục dị biệt Dirty Read, Repeatable Read và Phantom Read

**Hai cấp độ cô lập mới**

* Read Commited with snapshots: sử dụng row versioning và Optimistic
* Snapshots: giải quyết được các lỗi dị biệt

**Hướng dẫn sử dụng các cấp độ cô lập**

* Các tác nhân ảnh hưởng đến việc chọn cấp độ cô lập
  + **Data Integrity**: ràng buộc dữ liệu => các trường hợp lỗi dị biệt
  + **Concurrency Conflicts**: đụng độ giữa các user
  + **Performance Overhead**: chi phí thực hiện
* Cấp độ cô lập **Read Uncommited**
  + Ứng dụng không yêu cầu dữ liệu đọc được phải chính xác tuyệt đối
  + Thao tác dữ liệu phải kết thúc càng nhanh càng tốt
* Cấp độ cô lập **Read Commited with Locking**
  + Ứng dụng không yêu cầu việc truy cập dữ liệu phải diễn ra trong một thời gian dài. Và khi có yêu cầu đọc thì dữ liệu đọc là phải chính xác tuyệt đối
  + Thích hợp với việc truy cập dữ liệu tuần tự
* Cấp độ cô lập **Read Commited with Snapshots**
  + Ứng dụng có yêu cầu việc truy cập dữ liệu sẽ diễn ra trong một thời gian dài
  + Dữ liệu truy cập nhất quán kể từ khi bắt đầu đọc
* Cấp độ cô lập **Repeatable Read**
  + Ứng dụng có yêu cầu đọc dữ liệu để xem, tính toán và luôn bảo đảm tính nhất quán. Có thể sau khi xem, tính toán thì quyết định cập nhật
  + Khi xem và tính toán dữ liệu kết thúc thì các giao tác khác mới được cập nhật dữ liệu
* Cấp độ cô lập **Snapshot**
  + Ứng dụng có yêu cầu đọc dữ liệu để xem, tính toán và luôn bảo đảm tính nhất quán. Có thể sau khi xem, tính toán thì không quyết định cập nhật
  + Các giao tác khác vẫn được xem và cập nhật dữ liệu
* Cấp độ cô lập **Serializable**
  + Ứng dụng có yêu cầu đọc dữ liệu để xem, tính toán và luôn bảo đảm tính nhất quán
  + Các giao tác khác không được cập nhật dữ liệu

## Phân quyền bảo mật

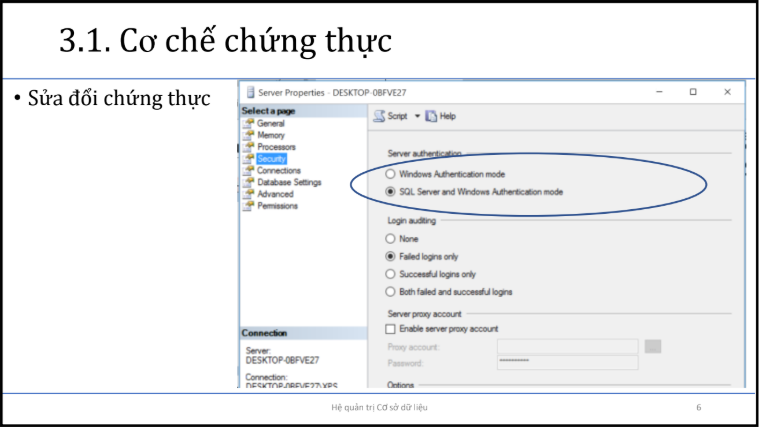
**An toàn dữ liệu:** nhằm chống lại sự xâm nhập trái phép từ bên

trong và bên ngoài hệ thống. Các tác nhân tác động từ bên ngoài

gây hư hỏng hệ quản trị cơ sở dữ liệu, sai lệch, mất mát dữ liệu,

cấu trúc dữ liệu,...

1. Cơ chế chứng thực

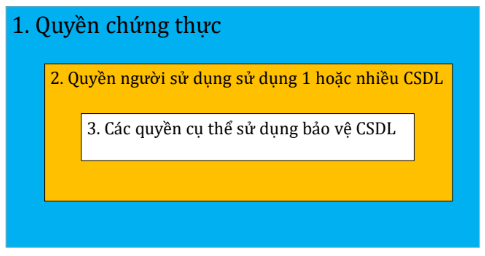


Hình 6 Cơ chế chứng thực

Hai chế độ chứng thực:

* **Mixed security Mode:** Kết nối với SQL server bằng cách dùng Windows Authentication hoặc SQL server Authentication
* **Windows authentication mode:** Chỉ có thể kết nối với SQL server bằng Windows Authentication

1. Cơ chế phân quyền



Hình 7 Cơ chế phân quyền

• Kiểm soát user có thể làm được gì trên database bằng sử dụng:

* Các Database role
* Cấp quyền cho user thao tác trên object và statement

• Các cơ chế cấp quyền:

* Dùng login account được tạo sẵn bởi hệ thống và được gán role default như: sa được gán sysadmin fixed server role.
* Chỉ định 1 login account là user của một Database: mặc định có quyền thuộc public database role.
* Sử dụng role/cơ chế role bao trùm: Sysadmin bao trùm db\_ower
* Sử dụng các lệnh cấp quyền cho user thao tác trên object và statement như grant, deny, revoke

ROLES:

• Role là một công cụ cho phép cấp quyền cho một nhóm User thay vì thực hiện trên từng user.

• Có 2 loại Roles:

* Fixed role
* User-defined Database role

• Hoặc có thể phân biệt:

* Role mức server
* Role mức Database

o User Defined Roles: bạn phải là thành viên của db\_securiadmin, hay db\_owner, hay sysadmin, … để có thể tạo role.

o Dùng T-SQL tạo role:

* Bước 1: Định nghĩa một role (Một user\_defined database role được định nghĩa trong một DataBase)
* sp\_addrole role\_name, role\_owner
* Bước 2: Gán quyền về statement và object cho role
* Bước 3: Gán các role là thành viên của role

o Xóa role: Sp\_droprole rolename

**Các thao tác về quyền**

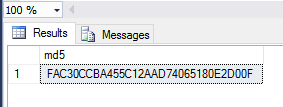
1. **Lệnh Cấp Quyền (grant):** Nếu bạn cấp quyền cho User và User là thành viên của một Role. Các quyền mà User có được hợp lại từ hai phía.
   * Granting Statement Permission :
     + GRANT { ALL | statement [ ,...n ] } TO user\_name [,...n ]
   * Granting Object Permission :
     + GRANT{{ ALL | permission [ ,...n ]}[(column\_name [ ,...n ] ) ] ON { table | view | stored\_procedure |extended\_procedure | user\_defined\_function }} TO user\_name [ ,...n ]
2. **Lệnh chối từ (deny):** Ngăn User sử dụng quyền và không cho phép User có cơ hội thừa hưởng do là thành viên của một Role.
   * Denying Statement Permission :
     + DENY { ALL | statement [ ,...n ] } TO user\_name [,...n ]
   * Denying Object Permission :
     + DENY {{ALL | permission [ ,...n ] } [ ( column\_name[ ,...n ] ) ] ON { table | view | stored\_procedure |extended\_procedure | user\_defined\_function }} TO user\_name [ ,...n ]
3. **Lệnh hủy (revoke):** Hủy quyền đã cấp grant hay từ chối deny
   * Revoking Statement Permission :
     + REVOKE {ALL | statement [ ,...n ] } FROM user\_name [ ,...n ]
   * Revoking Object Permission :
     + REVOKE{{ALL | permission[ ,...n ]}[(column\_name [ ,...n ] ) ] ON { table | view | stored\_procedure |extended\_procedure | user\_defined\_function }} {TO | FROM} user\_name [ ,...n ]
4. Cơ chế mã hóa
   1. Mã hoá một chiều MD5 trong SQL Server

Thuật toán mã hóa md5 là mã hóa một chiều, không có thể dịch ngược lại. Và bất kỳ ký tự mã hóa trong md5 cho kết quả tối đa không quá 32 ký tự.

Mã hóa chuỗi “http://laptrinhvb.net” bằng md5 bằng câu lệnh SQL như sau:

**SELECT CONVERT(VARCHAR(32), HASHBYTES(‘MD5’, ‘HTTP://LAPTRINHVB.NET’), 2) AS MD5**

kết quả trả về:



* 1. Mã hoá đối xứng hai chiều trong SQL Server

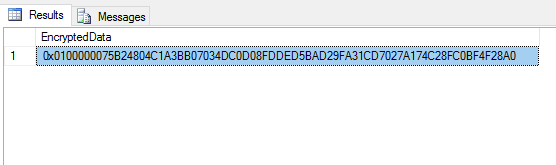
Mã hóa hai chiều, có nghĩa là khi mã hóa một chuỗi dữ liệu sẽ tạo khóa cho nó. Khi các các cần giải mã, chỉ cần nhập khóa vào là có thể giải mã được.

* **Hàm mã hóa 2 chiều EncryptByPassPhrase**

Mã hóa chuỗi “laptrinhvb.net” với khóa là “123”:

**SELECT ENCRYPTEDDATA = ENCRYPTBYPASSPHRASE(‘123’, ‘LAPTRINHVB.NET’ )**

Kết quả trả về:

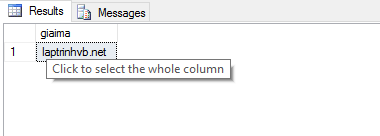


* **Hàm giải mã 2 chiều DecryptByPassPhrase trong SQL server.**

Giải mã đoạn chuỗi ví dụ vừa mã hóa ở trên.

**SELECT CONVERT(VARCHAR(100),DECRYPTBYPASSPHRASE(‘123’, 0X0100000075B24804C1A3BB07034DC0D08FDDED5BAD29FA31CD7027A174C28FC0BF4F28A0)) AS GIAIMA**

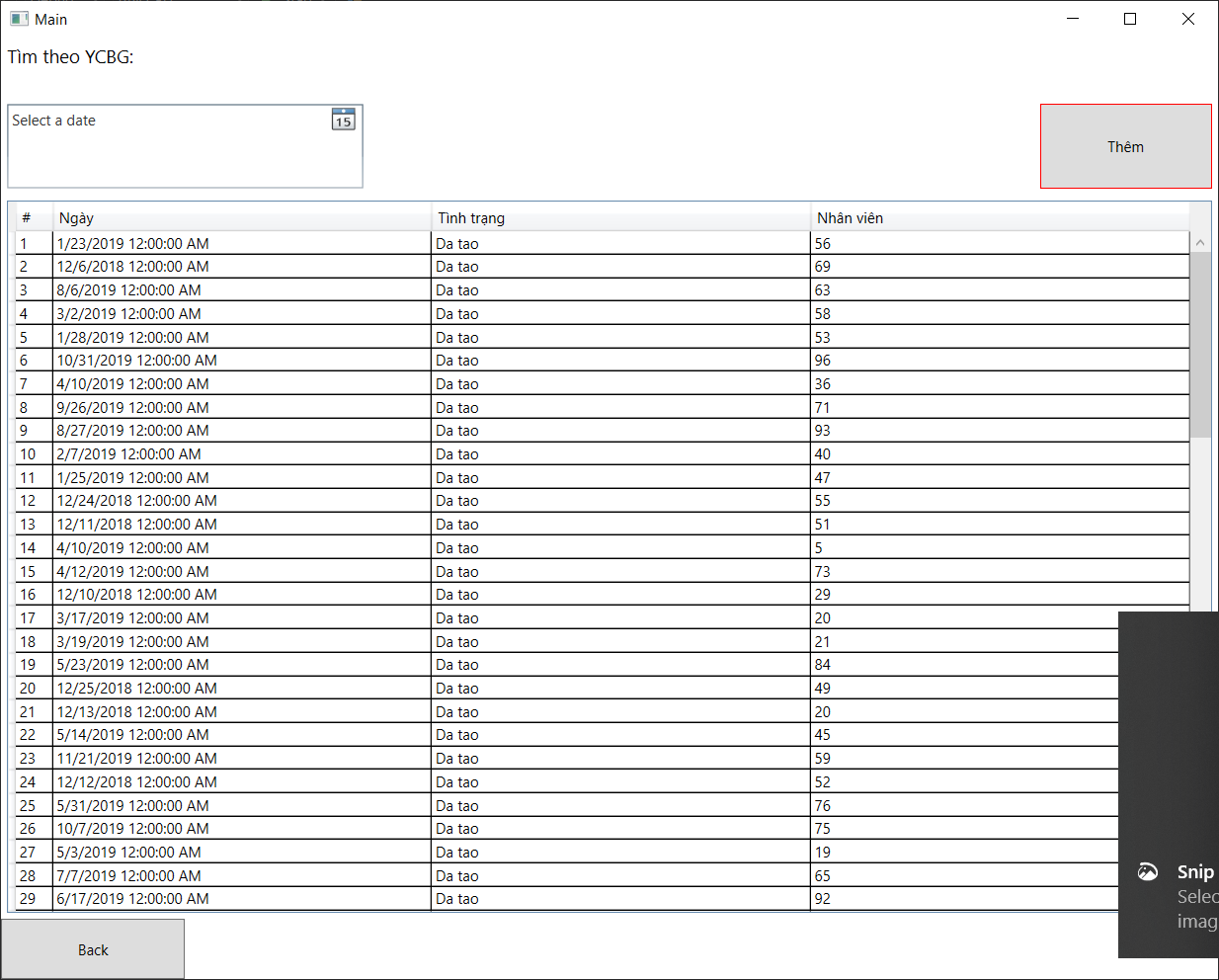
Kết quả trả về:



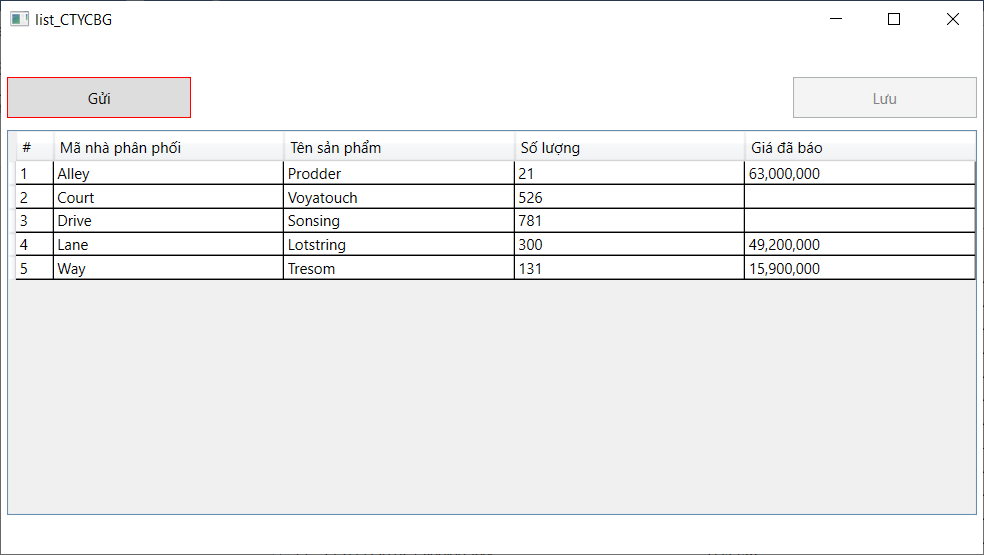
# *Chương 4: Triển khai thực nghiệm*

## Kết quả đạt được:

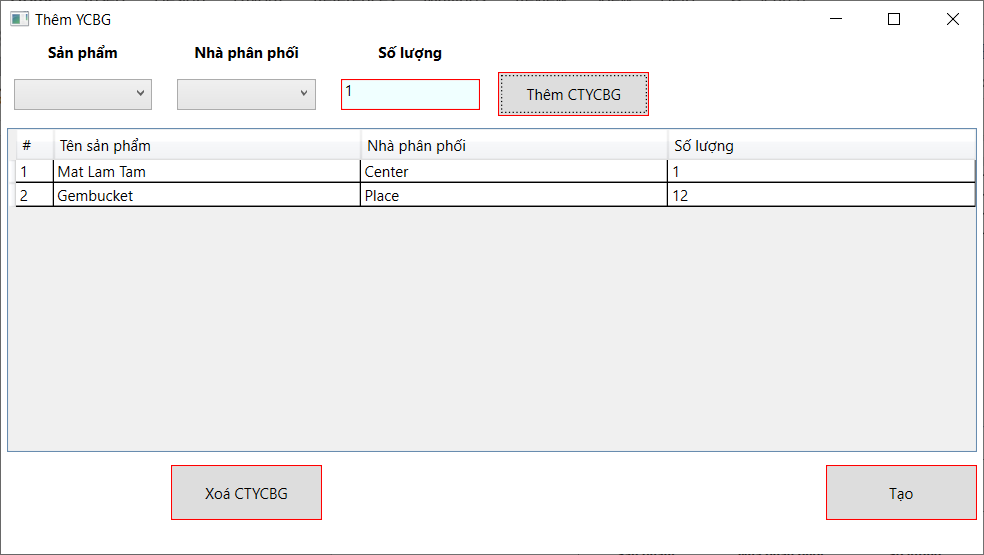
* Cài đặt được tất cả các ràng buộc toàn vẹn
* Các giao tác trên các tình huống:
  + Thêm yêu cầu báo giá
  + Thêm chi tiết yêu cầu báo giá
  + Sửa chi tiết yêu cầu báo giá
  + Sửa yêu cầu báo giá
  + Xoá chi tiết yêu cầu báo giá
* Cài đặt các con trỏ liên quan
* Phân quyền:
  + Admin
  + Quản lý
* Giao diện:



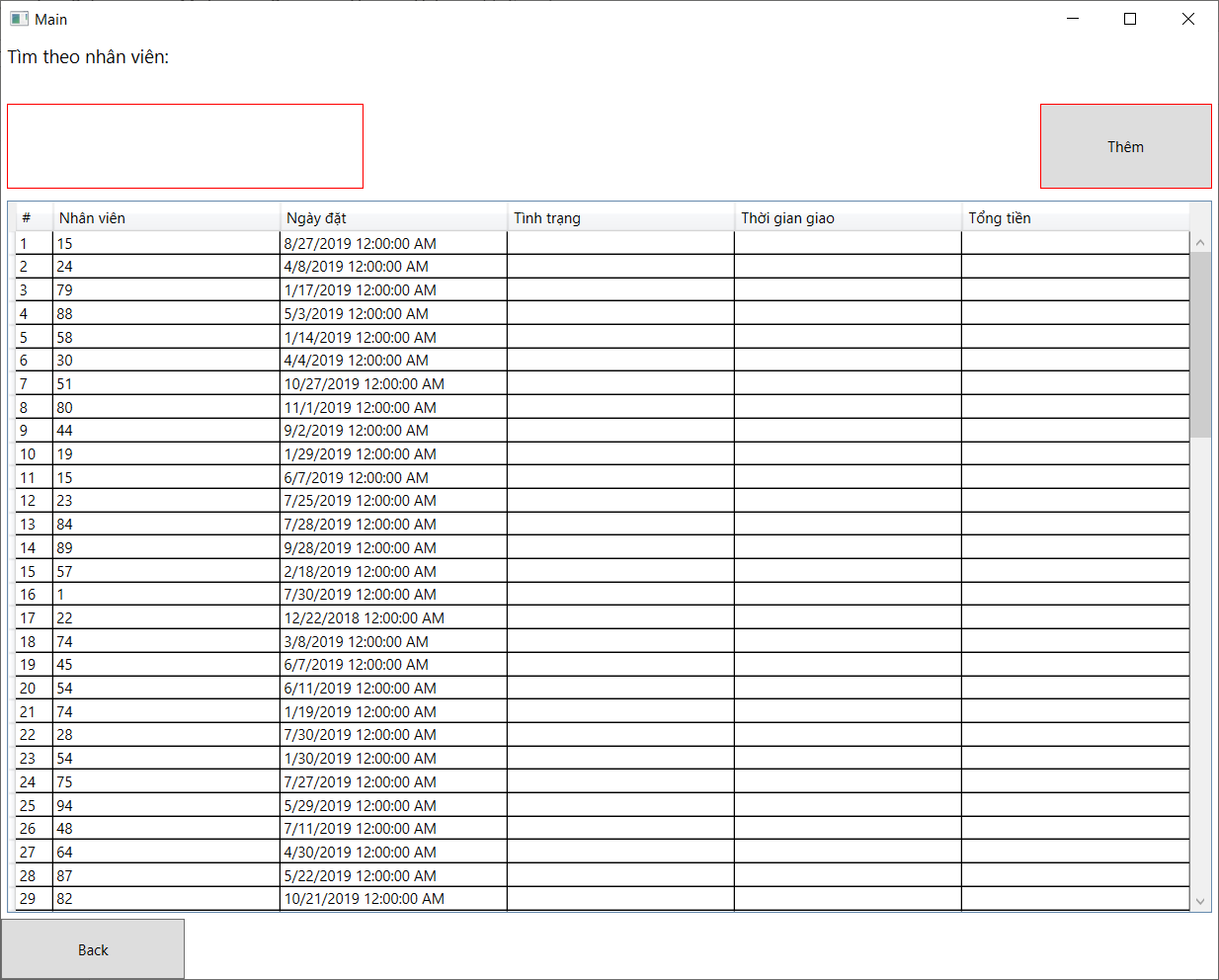
Hình 8 Danh sách yêu cầu báo giá



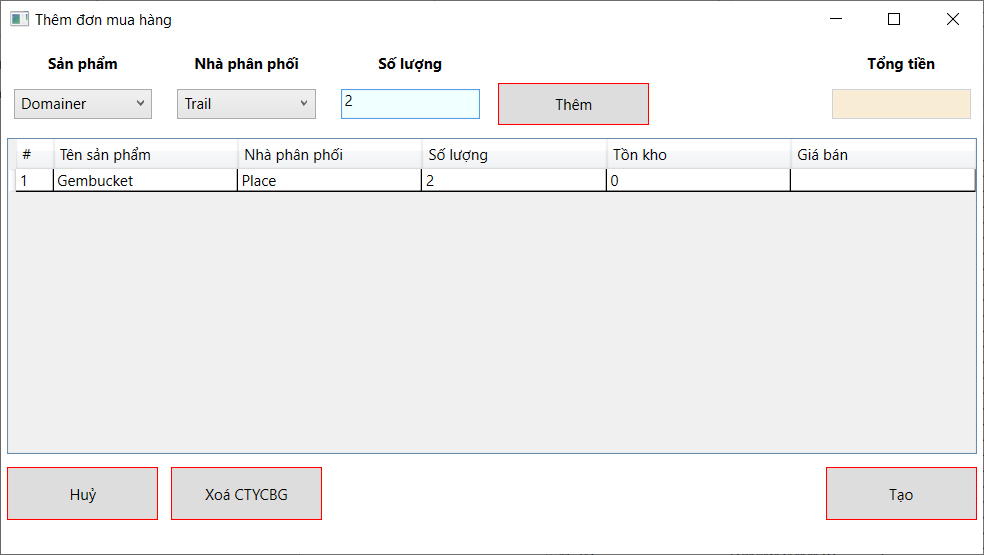
Hình 9 Chi tiết yêu cầu báo giá



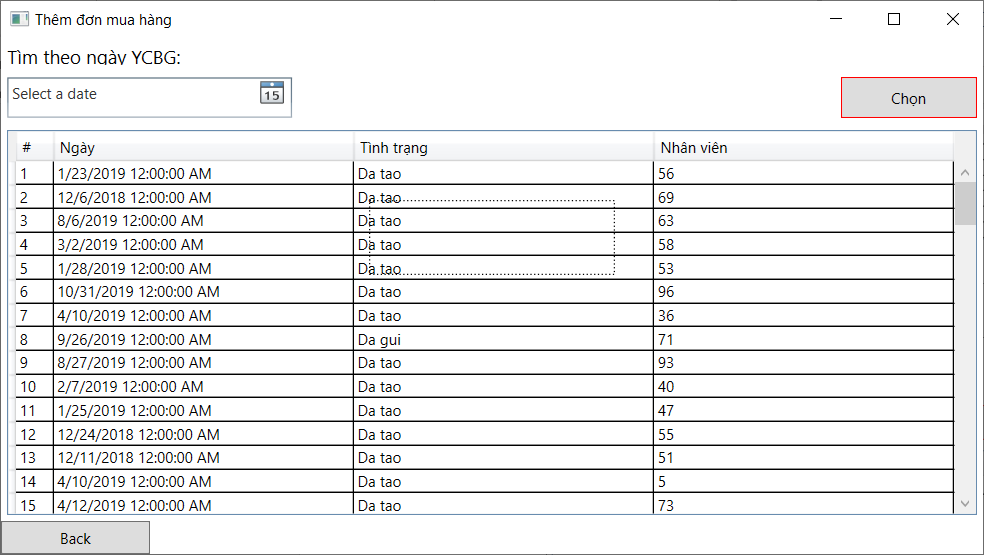
Hình 10 Thêm yêu cầu báo giá



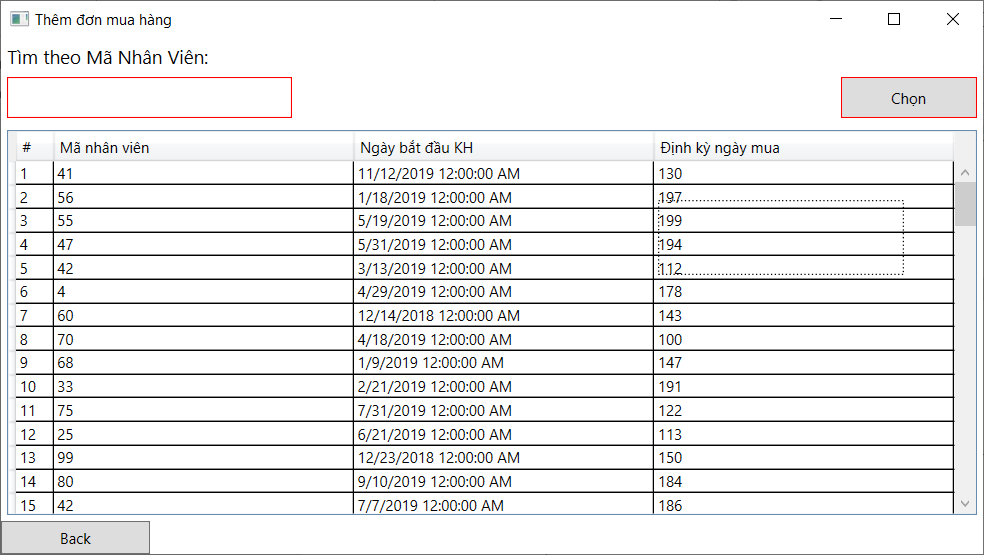
Hình 11 Danh sách đơn mua hàng



Hình 12 Thêm mới đơn mua hàng



Hình 13 Thêm mới đơn mua hàng từ yêu cầu báo giá



Hình 14 Thêm từ kế hoạch mua

## Hướng phát triển:

* Có thể gửi yêu cầu báo giá bằng nhiều ngôn ngữ
* Có thể đặt giới hạn thời gian trong yêu cầu báo giá
* Có thể thay đổi biểu mẫu gửi đi
* Hiển thị phần mềm bằng nhiều ngôn ngữ
* Phát triển trên nền tảng di dộng
* Có thể định trước ngày giao hàng
* Tự động tạo yêu cầu báo giá trong quá trình sản xuất
* Gửi yêu cầu báo giá dựa trên file excel

# *Tài Liệu Tham Khảo*

[1] www.sqlshack.com

[2] www.red-gate.com/simple-talk/sql/database-administration/sql-server-storage-internals-101/

[3] techtalk.vn/sqlserver-ma-hoa-du-lieu-mot-chieu-md5-va-ma-hoa-doi-xung-hai-chieu-trong-sqlserver.html

[4] docs.microsoft.com/

[4] www.w3schools.com