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

**ĐẠI HỌC KHOA HỌC TỰ NHIÊN THÀNH PHỐ HỒ CHÍ MINH, ĐẠI HỌC QUỐC GIA TP HCM**

Sinh viên thực hiện: 19127118-Tạ Công Điền

19127348-Bùi Công Danh

19127518-Lê Hoàng Thịnh Phước

19127362-Lê Xuân Dĩnh

GV phụ trách: Hồ Thị Hoàng Vy

Đồ án môn học  - CƠ SỞ DỮ LIỆU NÂNG CAO

HỌC KỲ I – NĂM HỌC 2021-2022

**môn cơ sở dữ liệu nâng cao**

**BẢNG THÔNG TIN CHI TIẾT NHÓM**

|  |  |
| --- | --- |
| **Mã nhóm:** | ADB1\_8 |
| **Tên nhóm:** | Nhóm 4 |
| **Số lượng:** | **4** |
| **MSSV** | **Họ tên** |
| 19127118 | Tạ Công Điền |
| 19127518 | Lê Hoàng Thịnh Phước |
| 19127362 | Lê Xuân Dĩnh |
| 19127348 | Bùi Công Danh |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Bảng phân công & đánh giá hoàn thành công việc** | | | | |
| **Công việc thực hiện** | **Người thực hiện** | **Mức độ hoàn thành** | **Đánh giá của nhóm** |
| Viết quy trình nghiệp vụ | Bùi Công Danh | 100% | 9/10 |
| Viết mô tả dữ liệu ràng buộc liên quan | Tạ Công Điền | 100% | 8/10 |
| Viết danh sách chức năng +tần suất giao dịch | Lê Hoàng Thịnh Phước | 100% | 8/10 |
| Tạo cơ sở dữ liệu mức vật lí, logic, quan niệm | Tạ Công Điền | 100% | 9/10 |
| Generate dữ liệu | Lê Xuân Dĩnh | 100% | 9/10 |
| Viết câu truy vấn | Lê Hoàng Thịnh Phước | 100% | 8/10 |
| Lập index cho database | Lê Xuân Dĩnh | 100% | 8/10 |
| Viết trigger cho database | Lê Xuân Dĩnh | 100% | 8/10 |
| Phân quyền cho người dùng | Tạ Công Điền | 100% | 8/10 |
| Viết procedure demo | Bùi Công Danh  Tạ Công Điền | 100% | 9/10 |
| Làm giao diện demo tính năng | Bùi Công Danh(chính)  Tạ Công Điền(phụ)  Lê Xuân Dĩnh(phụ) | 100% | 9/10 |
| Viết báo cáo | Tạ Công Điền  Bùi Công Danh | 100% | 9/10 |

**YÊU CẦU ĐỒ ÁN- BÀI TẬP**

|  |  |
| --- | --- |
| **Loại bài tập** | **Lý thuyết 🗹 Thực hành 🗹 Đồ án Bài tập** |
| **Ngày bắt đầu** | **01/12/2021** |
| **Ngày kết thúc** | **02/01/2022** |

1. Mục lục

[**I. Mô tả các quy trình dữ liệu 1**](#_Toc92055007)

[**I.1 Quy trình nghiệp vụ . 1**](#_Toc92055008)

[**I.2 Mô tả dữ liệu và các ràng buộc liên quan. 3**](#_Toc92055009)

[**I.3.Danh sách chức năng dự kiến kèm tần suất giao dịch tương ứng 3**](#_Toc92055010)

[**II. Thiết kế cơ sở dữ liệu 4**](#_Toc92055011)

[**II.1.Thiết kế cơ sở dữ liệu ở mức quan niệm. 4**](#_Toc92055012)

[**II.2 Thiết kế dữ liệu ở mức logic 8**](#_Toc92055013)

[**II.3 Thiết kế dữ liệu ở mức độ vật lí 10**](#_Toc92055014)

[**a.Thuộc tính suy dẫn 10**](#_Toc92055015)

[**b.Mô hình cài đặt trong SQL SERVER 12**](#_Toc92055016)

[**III.Các đề xuất sử dụng để cải thiện hiệu quả truy vấn (sử dụng index). 13**](#_Toc92055017)

[**IV.Một số phần cài đặt trong Script 24**](#_Toc92055018)

[**IV.1 Trigger 24**](#_Toc92055019)

[**IV.2 Phân quyền người dùng 25**](#_Toc92055020)

[**V.Link Source code và video 26**](#_Toc92055021)

[**VI.Phần trăm công việc từng người và ảnh báo cáo trên GITHUB 26**](#_Toc92055022)

[**VII.Phụ lục ( Cách sử dụng sqlgen để tạo data) 27**](#_Toc92055023)

[**VII. Tài liệu tham khảo 28**](#_Toc92055024)

1. Kết quả đạt được

# I. Mô tả các quy trình dữ liệu

## I.1 Quy trình nghiệp vụ .

Các loại sản phẩm đều có: mã sản phẩm, tên sản phẩm, giá, mô tả, loại sản phẩm, tình trạng, thuộc loại hàng nào, số lượng tồn, mức giảm giá. Một loại hoa có thể thuộc nhiều loại hàng khác nhau.

Các loại hoa sẽ thuộc danh loại mục khác nhau, bao gồm: hoa sinh nhật, hoa theo chủ đề, hoa tươi, màu sắc, hoa đặc biệt, hoa cưới. Riêng đối với các sản phẩm khác không phải là hoa thì thuộc thể loại quà tặng kèm. Một loại hoa có thể thuộc nhiều danh mục khác nhau.

**Phân hệ khách hàng:**

Mỗi lần khách mua hàng, cần lưu lại thông tin đơn bán hàng. Thông tin hóa đơn gồm: số đơn bán hàng, ngày lập hóa đơn, họ và tên, điện thoại, email, địa chỉ, hình thức đặt hàng, nhân viên thanh toán. Đối với người nhận hàng cần lưu thông tin người nhận, địa chỉ nhận hàng, và thời gian giao hàng.

Khi khách đã chọn lựa xong sản phẩm sẽ đến mục giỏ hàng để kiểm tra lại các sản phẩm đã chọn, phí tạm tính, giảm giá và tổng thành tiền, đã bao gồm 10% thuế giá trị gia tăng (VAT). Khách hàng cũng có thể sử dụng phiếu voucher để nhận thêm nhiều ưu đãi từ cửa hàng. Thông tin phiếu giảm giá bao gồm: mã phiếu, thời gian hết hạn, mức giảm giá, nội dung.

Có 4 hình thức thanh toán:

* 1. Thanh toán trực tiếp tại cửa hàng.
  2. Thanh toán tại nơi khác, phụ thu phí là 30.000đ.
  3. Thanh toán trực tiếp khi nhận hàng.
  4. Thanh toán qua thẻ ngân hàng hoặc các ví điện tử.

Đối với các đơn hàng đặt qua website (online), cửa hàng sẽ giảm giá thêm 20.000đ cho mỗi lần đặt.

Khách hàng sẽ được giảm giá 3% cho lần thứ 2 đặt hàng. Ngoài ra, khách hàng còn được giảm 5% cho lần đặt hàng ONLINE thứ 5 và 10% cho lần thứ 6.

Khách hàng có thể tạo tài khoản khi sử dụng dịch vụ. Hệ thống lưu lại thông tin khách hàng khi đăng ký gồm: email, mật khẩu, họ tên, số điện thoại.

Khách hàng có thể xem, thay đổi thông tin cá nhân, và xem lại chi tiết các đơn đặt hàng trước đó.

**Phân hệ quản trị:**

Kho hàng là nơi quản lý các sản phẩm của cửa hàng. Mỗi kho hàng có một mã kho duy nhất dùng để phẩn biệt các kho khác, tên kho hàng, địa chỉ kho hàng, quản trị viên quản lý kho.

Vào cuối mỗi ngày, quản trị viên sẽ tiến hành kiểm tra hàng hóa. Quản trị viên sẽ liên hệ với nhà cung ứng để nhập hàng nếu hết hàng trong kho. Ứng với mỗi lần nhập hàng sẽ có 1 phiếu nhập hàng, bao gồm mã phiếu nhập, ngày nhập, do quản trị viên nào phụ trách. Khi nhập hàng, cần phải lưu lại tên mặt hàng, số lượng nhập. Cần phải tính xem tổng chi phí nhập hàng là bao nhiêu. Khi cửa hàng có nhu cầu xuất hàng, cần phải có 1 phiếu xuất hàng gồm các thông tin: mã đơn, ngày lập, quản trị viên phụ trách, tổng chi phí xuất. Cần lưu thông tin số lượng các mặt hàng và số lượng mặt hàng cần xuất.

Quản trị viên thực hiện các thao tác thêm, xóa, cập nhật sản phẩm, lưu vết giá sản phẩm đối với những thay đổi của cửa hàng.

**Phân hệ quản lý:**

Quản lý cần thống kê xem số lượng các mặt hàng bán ra, doanh thu cửa hàng theo tháng, theo năm; dựa trên doanh thu theo từng năm để có thể thiết lập giảm giá cho các mặt hàng tương ứng, hoặc ưu đãi cho khách hàng. Ngoài ra, quản lý cần phải xem hiệu suất làm việc của nhân viên để có thể đưa ra các mức thưởng/phạt.

**Phân hệ nhân viên:**

Nhân viên khi làm việc cần phải thực hiện điểm danh. Nếu nhân viên vắng nhiều ngày theo quy định thì tùy thuộc vào số lượng ngày đã nghĩ mà có các mức xử phạt nhất định. Các nhân viên thực hiện tốt việc bán hàng sẽ có thưởng thêm vào cuối tháng. Nhân viên có thể xem lại lịch sử lương của mình.

Nhân viên có thể xem các đơn hàng quản lý bởi nhân viên đó và thống kê số lượng đơn hàng.

## I.2 Mô tả dữ liệu và các ràng buộc liên quan.

* Các dữ liệu liên quan MA (UID phân biệt của các thực thể: kiểu dữ liệu char).
* Các dữ liệu ngày của các phiếu nhập,phiếu xuất ,hóa đơn,ngày đăng ký tài khoản : kiểu dữ liệu date lấy ràng buộc là lấy date hiện tại.
* Các dữ liệu ngày sinh: Datetime
* Các dữ liệu địa chỉ ,hình thức thanh toán, mô tả có kiểu dữ liệu là nvarchar(30)
* Dữ liệu nhân biết tài khoản có khóa hay không, có kiểu dữ liệu bit mặc định là 0 (tức không khóa).
* Các kiểu dữ liệu liên quan đến mức giảm giá: float.
* Các kiểu dữ liệu liên quan giá tiền :int, phải >=0
* Ngoài ra một số dữ liệu còn lại thường sử dụng kiểu dữ liệu varchar.

## I.3.Danh sách chức năng dự kiến kèm tần suất giao dịch tương ứng

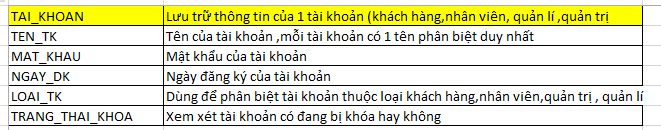
* Đăng nhập : Để cho khách hàng, quản lí, quản trị, nhân viên đăng nhập vào app (15000/ngày).
* Đối với khách hàng:
* Tìm kiếm sản phẩm: Khách hàng tìm kiếm sản phẩm mình cần dùng trên app (5000 lần/ngày).
* Mua hàng: Khách hàng mua những hoa mình cần (1000 lần/ngày).
* Thanh toán : Khách hàng thanh toán tiền hàng cho shop (1000 lần/ngày).
* Xem lịch sử mua hàng: Khách hàng xem những sản phẩm mình đã mua (400 lần/ngày)
* Đối với quản trị:
* Xem, cập nhật, xóa sản phẩm: Quản trị xem cập nhật xóa sản phẩm trong kho hàng (50 lần/ngày)
* Theo dõi tồn kho: kiểm tra số lượng sản phẩm tồn kho của kho hàng(1 lần/ngày)
* Theo dõi lịch sử nhập xuất hàng : Quản trị xem lịch sử nhập xuất hàng (10 lần/ngày).
* Đối với quản lí:
* Thống kê doanh thu của shop: Quản lí sẽ thống kê doanh thu của shop theo từng tháng (1 lần/tháng)
* Thống kế số lượng hàng: Quản lí sẽ thống kê số lượng hàng đang còn lại trong shop (1 lần/ngày).
* Xem hiệu suất làm việc của nhân viên: Mỗi ngày hoặc mỗi tháng quản lí sẽ xem hiệu suất làm việc của nhân viên (10 lần/tháng).
* Đối với nhân viên:
* Nhân viên điểm danh (150 lần/ ngày).
* Nhân viên xem số lương dơn hàng mà mình bán được (40 lần /ngày).
* Nhân viên xem lịch sử lương của bản thân (150 lần /ngày).

# II. Thiết kế cơ sở dữ liệu

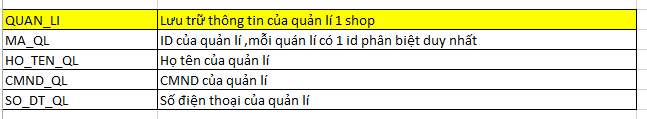
## II.1.Thiết kế cơ sở dữ liệu ở mức quan niệm.

Dựa theo phân tích các yêu cầu bài toán, ta có thể mô tả cho các thực thể như sau:

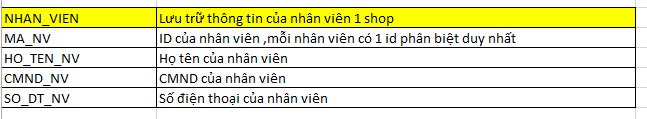
Thực thể tài khoản:



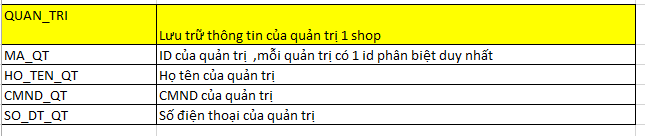
Thực thể quản lí:



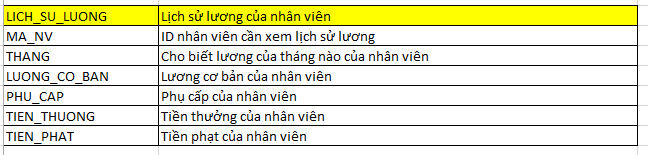
Thực thể nhân viên:



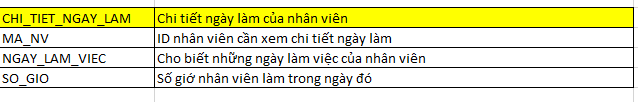
Thực thể quản trị:



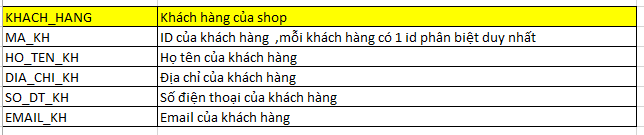
Thực thể lịch sử lương:



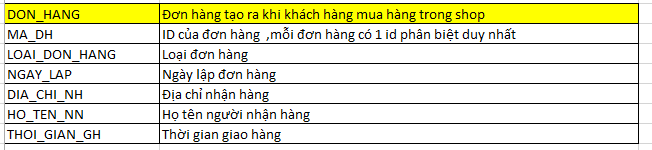
Thực thể chi tiết ngày làm



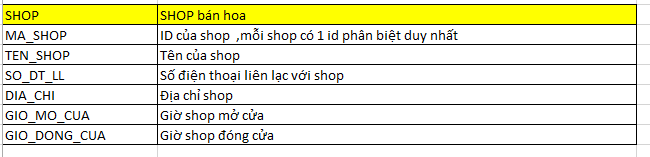
Thực thể khách hàng



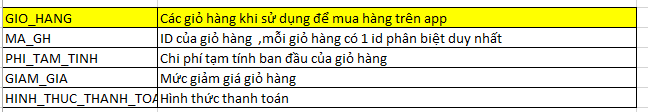
Thực thể đơn hàng



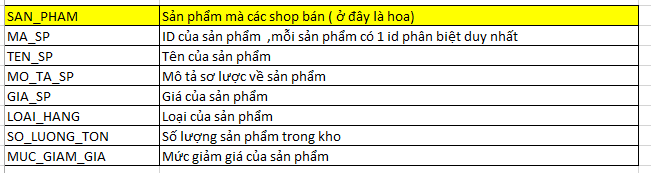
Thực thể shop



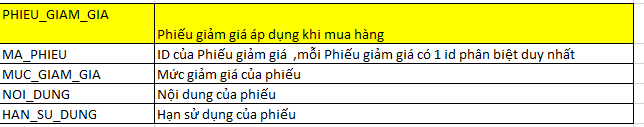
Thực thể giỏ hàng



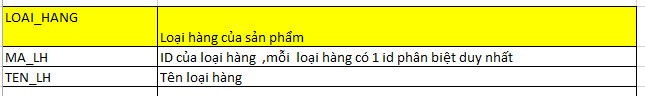
Thực thể sản phẩm



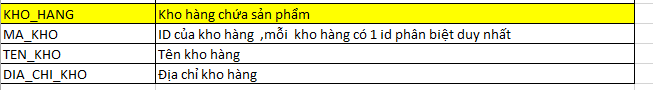
Thực thể phiếu giảm giá



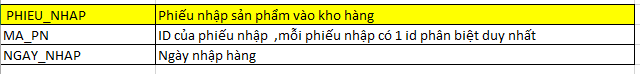
Thực thể loại hàng



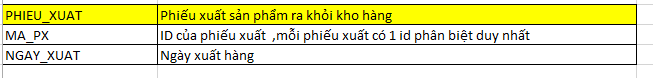
Thực thể kho hàng

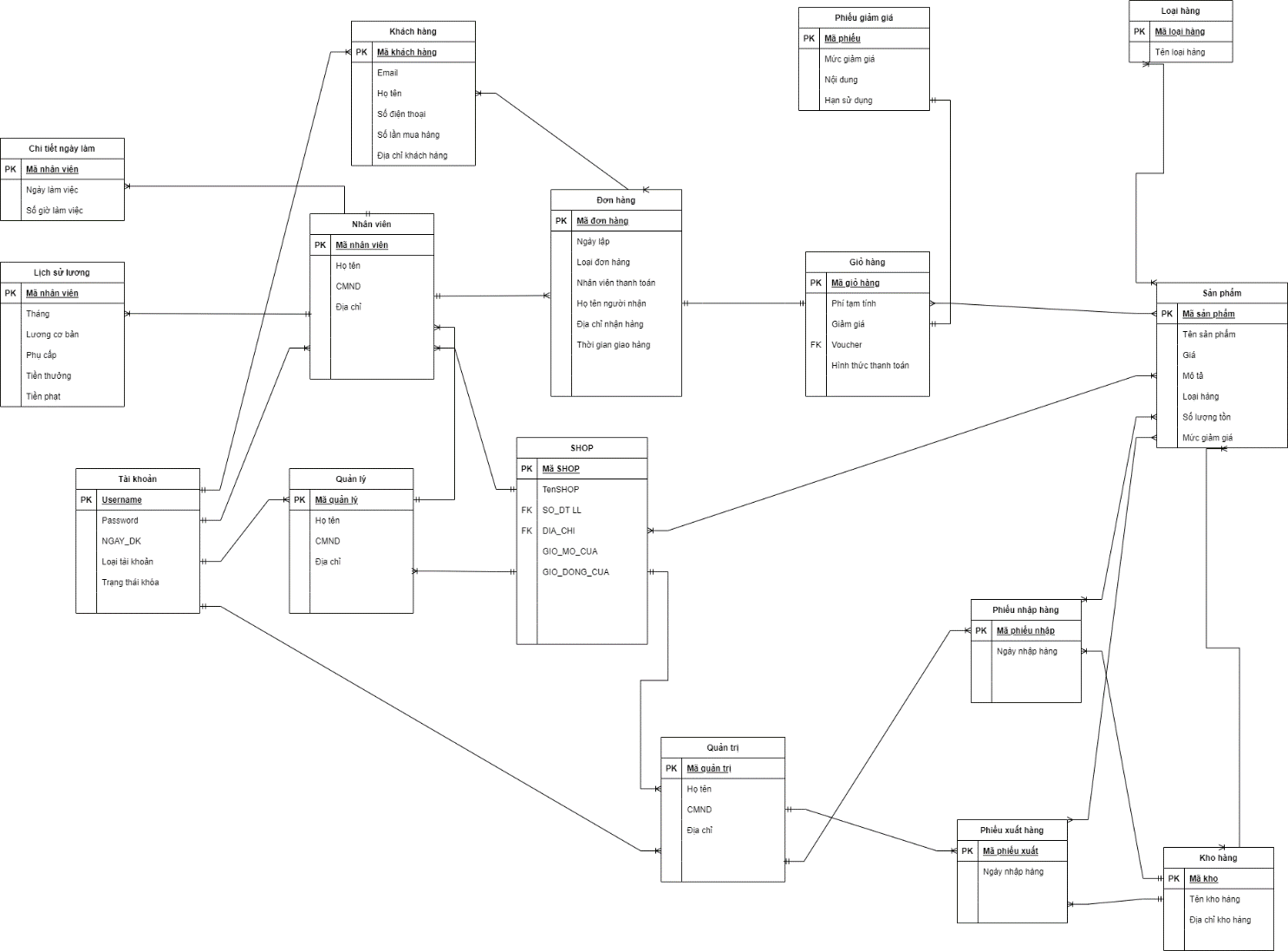


Phiếu nhập



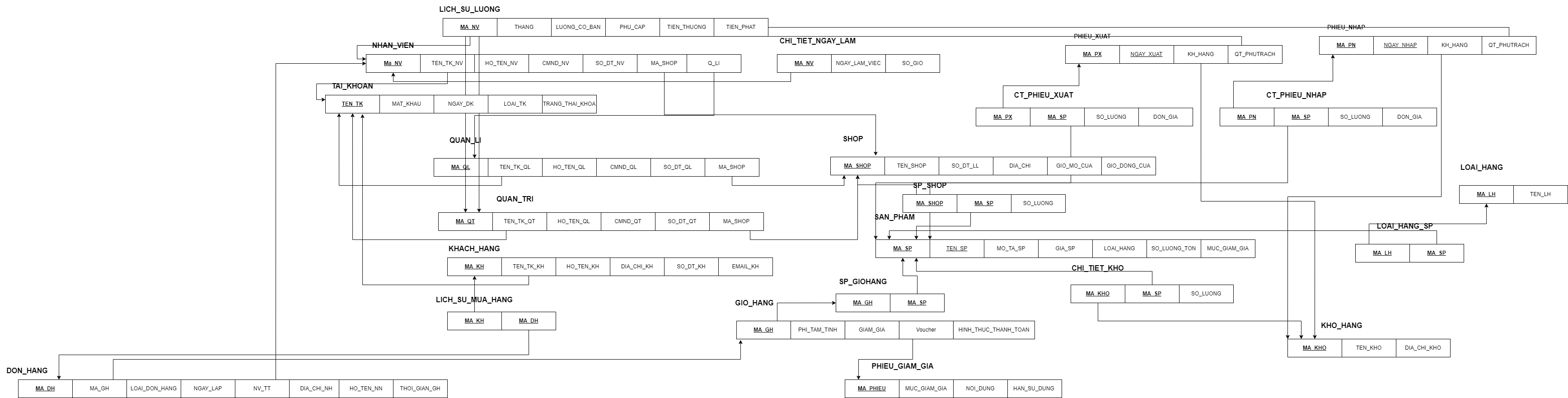
Phiếu xuất

Sau đây là mô hình ERD của thực thể :



## II.2 Thiết kế dữ liệu ở mức logic

Sử dụng lược đồ cơ sở dữ liệu ở trên ta thêm khóa ngoại và tạo ra các bảng trung gian là LICH\_SU\_MUA\_HANG, SP\_GIO\_HANG, SP\_SHOP, LOAI\_HANG\_SP, CHI\_TIET\_KHO, CT\_PHIEU\_NHAP, CT\_PHIEUXUAT thì ta được mô hình sau:



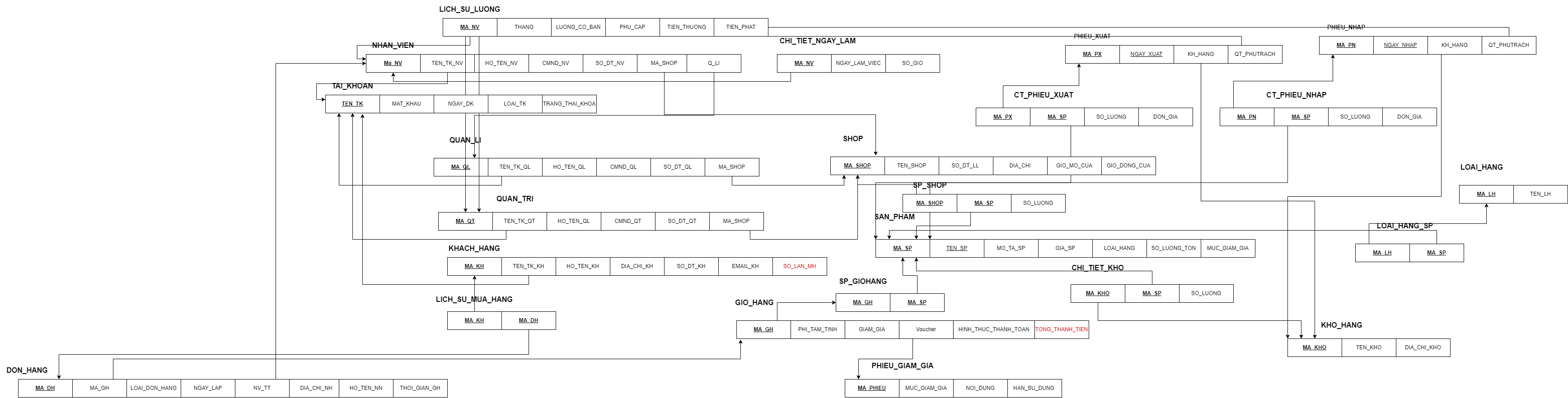
Ta xét thấy lược đồ trên đã đạt dạng chuẩn BCNF.

## II.3 Thiết kế dữ liệu ở mức độ vật lí

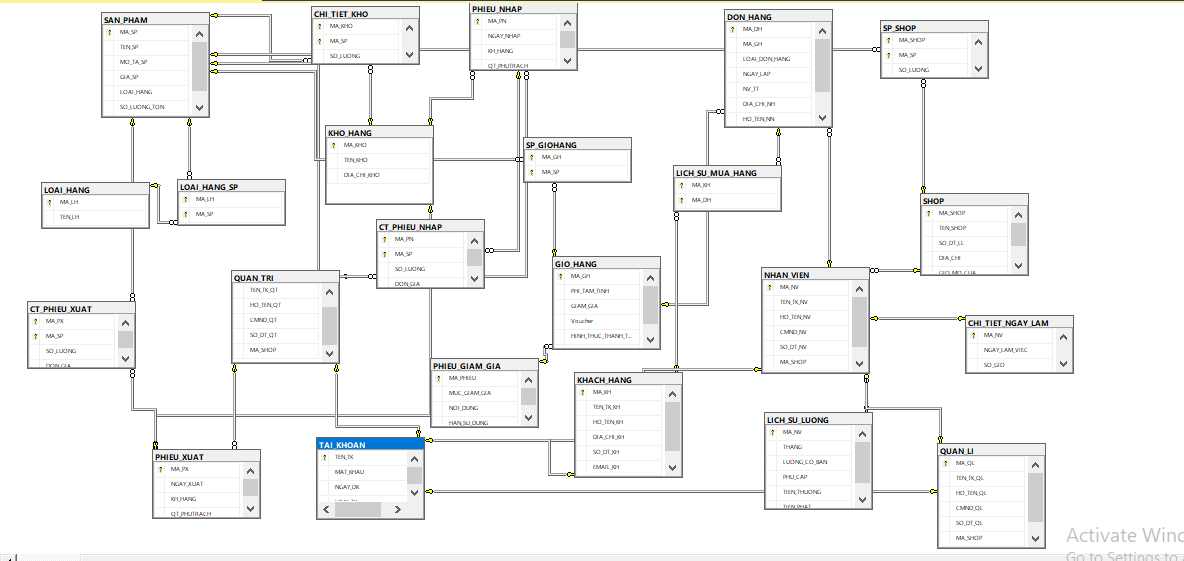
### a.Thuộc tính suy dẫn

Ta xây dựng thêm một số thuộc tính suy dẫn để tăng hiệu năng trong hệ thống:

* Giỏ hàng (GIO\_HANG): tạo thêm thuộc tính TONG\_THANH\_TIEN bằng tổng các dòng GIA\_SP của sản phẩm để nhanh chóng lấy ra tổng thành tiền của sản phẩm.
* Khách hàng (KHACH\_HANG): tạo thêm thuộc tính SO\_LAN\_MH bằng tổng các dòng của LICH\_SU\_MUA\_HANG để nhanh chóng lấy ra số lần mua hàng của khách hàng

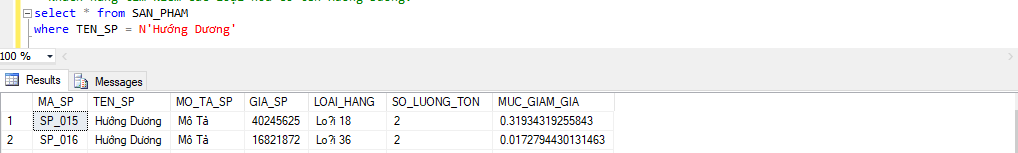


### b.Mô hình cài đặt trong SQL SERVER

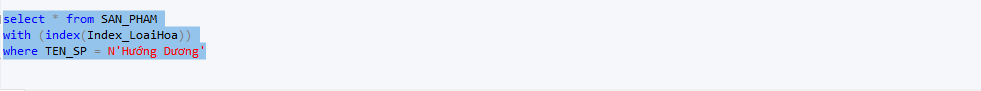


# III.Các đề xuất sử dụng để cải thiện hiệu quả truy vấn (sử dụng index).

**TH1:** Khách hàng muốn tìm kiếm các loại hoa Hướng Dương, thông thường chúng ta sẽ dùng như sau.

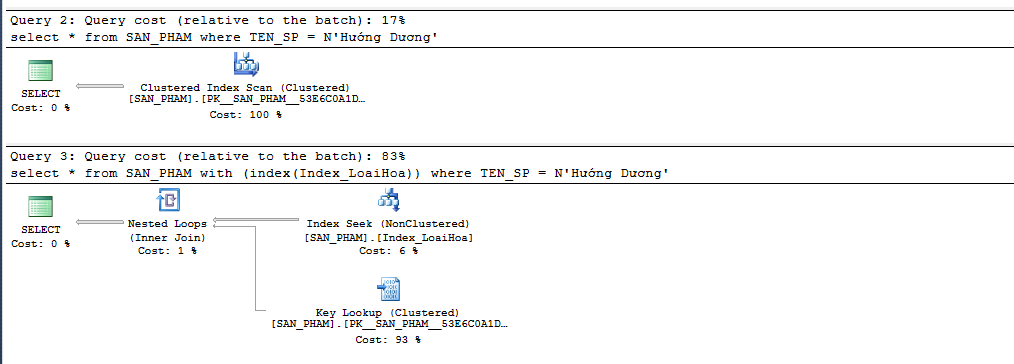


Khi sử dụng nonclustered index:



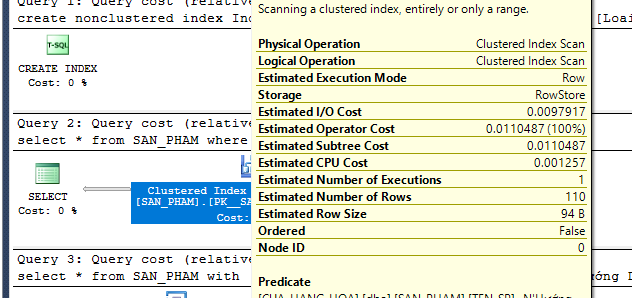
Với lượng dữ liệu ít, chúng ta khó xem xét hiệu quả của nonclustered index. Tuy nhiên khi dữ liệu nhiều, chúng ta sẽ thấy 1 cách rõ ràng.

Để so sánh 2 câu truy vấn khi không và khi sử dụng nonclustered index.

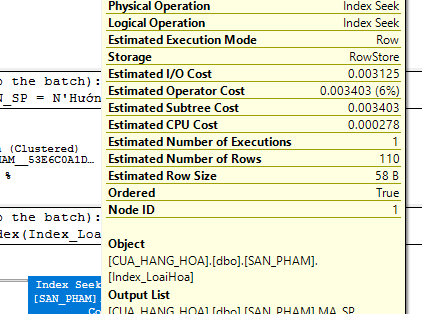


Về thời gian thực thi:

Ở câu lệnh thông thường

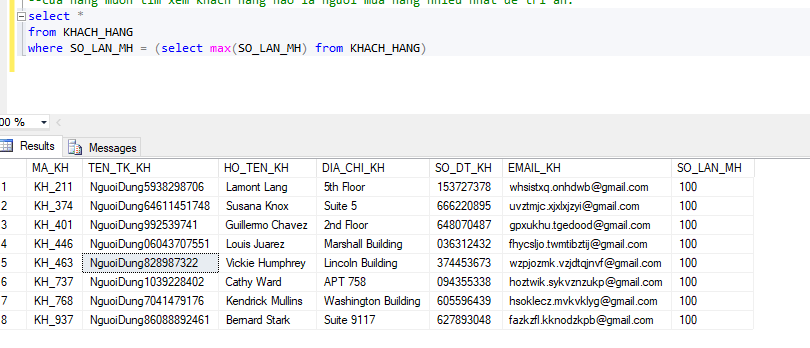


Và ở câu lệnh có sử dụng nonclustered index

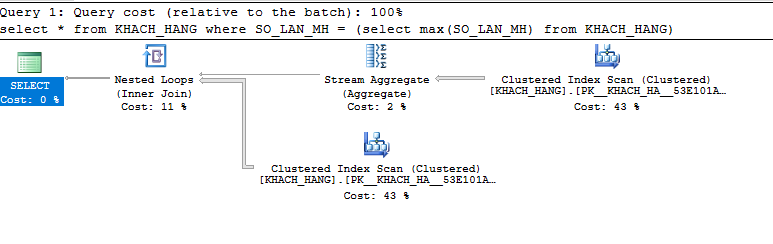


Sau khi xem execution plan ở trường hợp trên, ta thấy khi truy vấn mà chỉ cài khóa chính (clustered index), hiệu suất được cải thiện đáng kể vì query cost chỉ chiếm 17%, nhưng khi thêm nonclusterd index thì hiệu suất lại kém hơn.

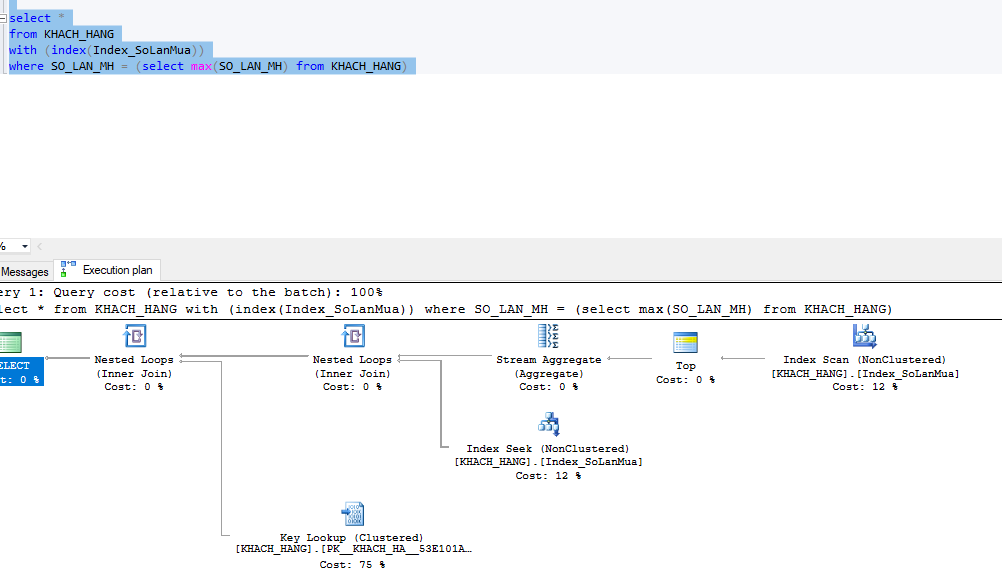
**TH2:** Cửa hàng muốn tìm những khách hàng đạt mốc mua hàng 100 lần để tri ân.



Với câu lệnh thông thường, ta được kết quả như sau:



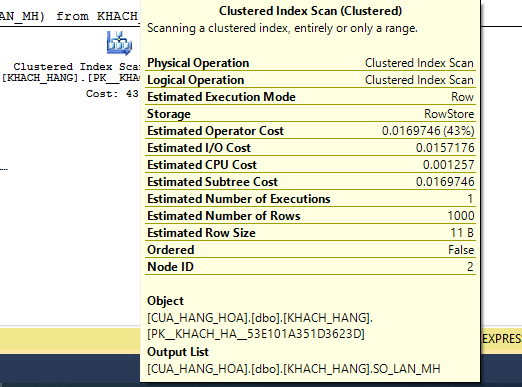
Khi sử dụng nonclustered index, ta được kết quả sau

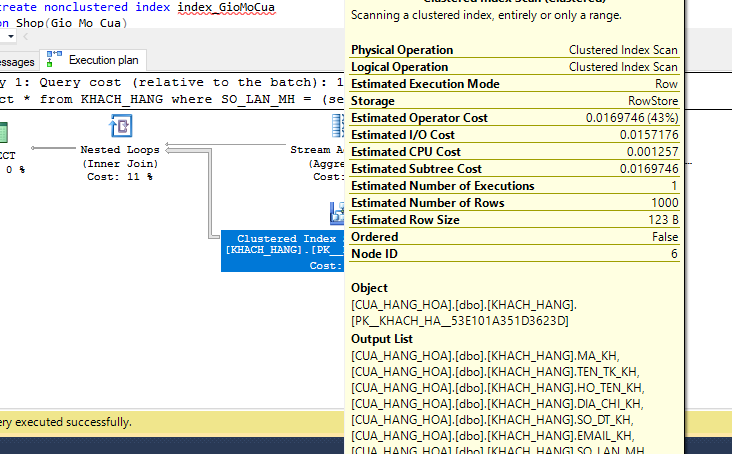


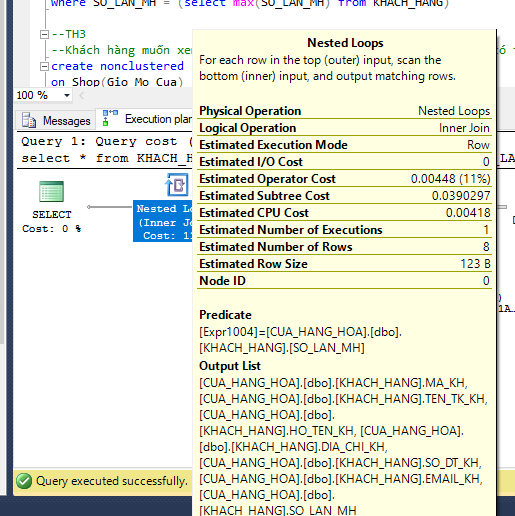
Với hiệu suất trên, ta dễ dàng thấy khi sử dụng nonclustered index, sẽ tối ưu hơn nhiều.

Khi so sánh thời gian thực thi, ta được kết quả sau:

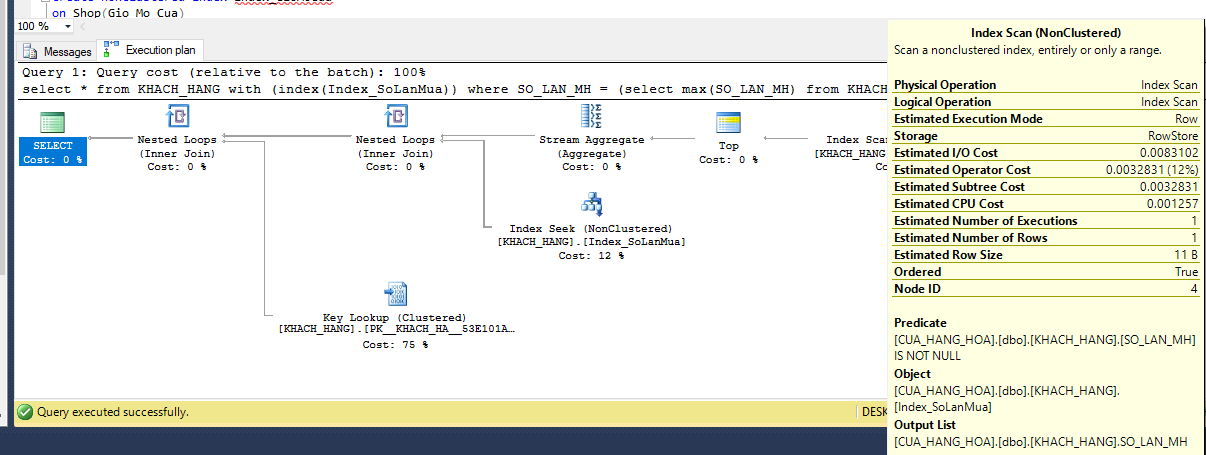
Ở câu lệnh thông thường:

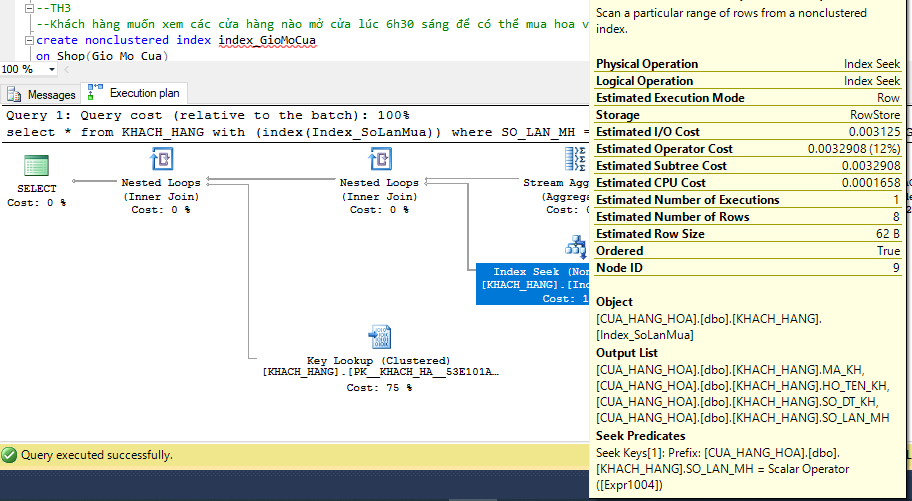


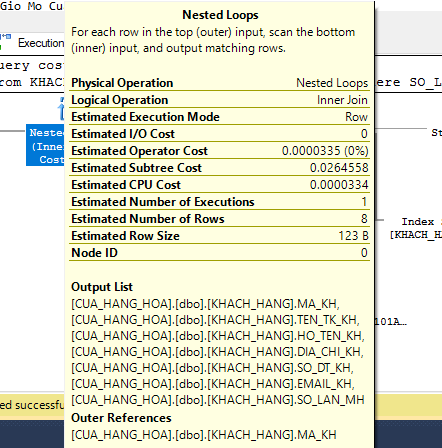




Và khi sử dụng nonclustered index, ta được kết quả như sau:

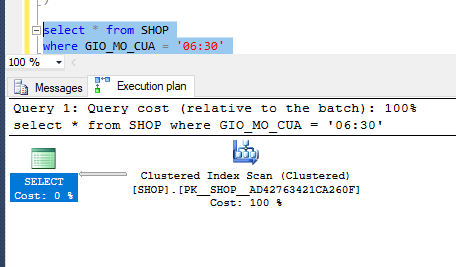




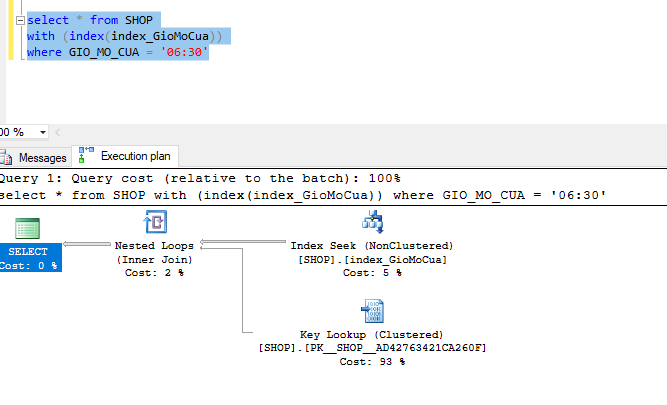


So với 0.00448+0.169746+0.0169746 = 0.174226 (s) của câu lệnh thông thường với 0.0032831+0.0032908+0.0000335=0.0066074 (s) của câu lệnh sử dụng nonclustered index, có sự khác biệt rất lớn được thể hiện ở đây

**TH3:** Khách hàng muốn xem các cửa hàng nào mở cửa lúc 6h30 sáng

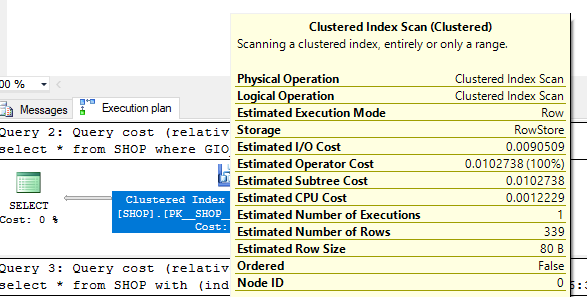


Khi sử dụng nonclustered index, ta được kết quả như sau:



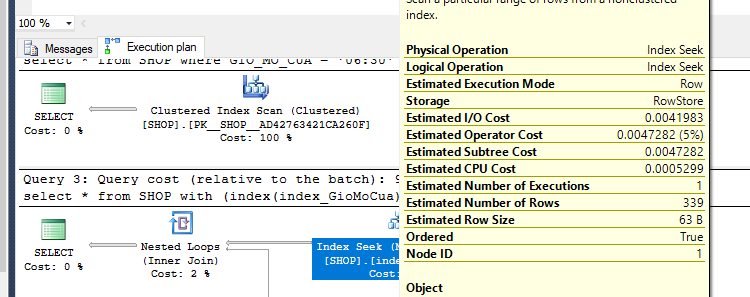
So về thời gian thực thi

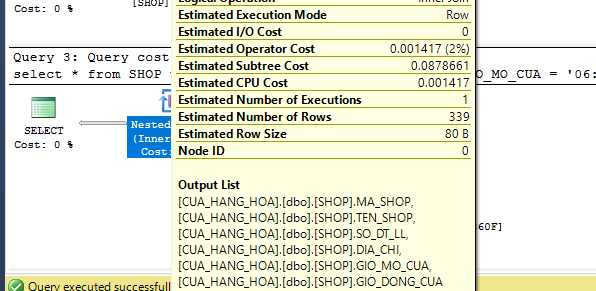
Ở câu lệnh thông thường, ta được kết quả sau:



Thời gian khoản 0.01 (s)

Và ở câu lệnh sử dụng nonclustered index

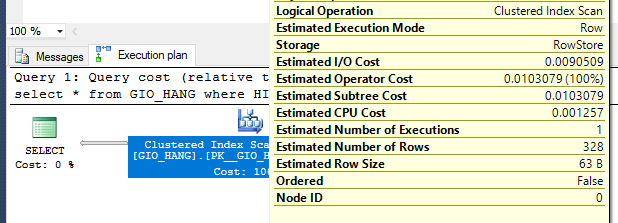




Chỉ tốn tầm 0.0014+0.0047 = 0.0061 (s) ở câu lệnh sử dụng nonclustered index

**TH4:** Người quản lí muốn xem những giỏ hàng có hình thức thanh toán tiền mặt.

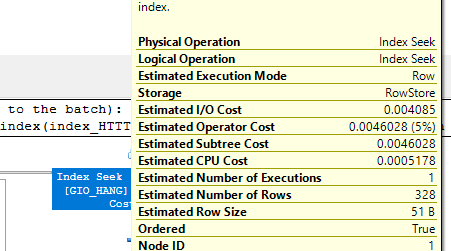
ở câu truy vấn không có nonclustered index.



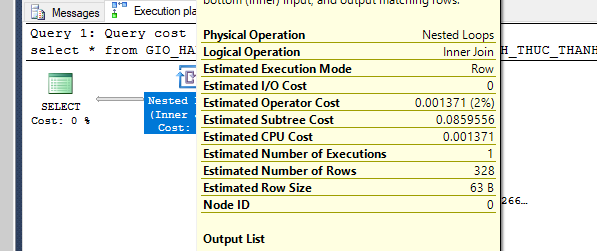
Kết quả thực thi là 0.01 (s)

Và ở câu truy vấn có sử dụng nonclustered index

Ở nonclustered index seek, ta có kết quả



Và ở vòng nested loop, ta có



Tổng thời gian thực thi của câu lệnh sử dụng nonclustered index khoảng 0.001+0.004 = 0.005 (s)

Trong khi đó, câu lệnh bình thường thực thi tới 0.01 (s)

Gấp đôi so với sử dụng nonclustered index.

Ngoài ra để tăng hiệu quả truy vấn ta đặt 1 sô index ở một vài thuộc tính ít cập nhật sửa đổi:

* Ở bảng KHACH\_HANG: Ta đặt index trên cột TEN\_TK\_KH để quá trình đăng nhập diễn ra nhanh hơn.
* Ở bảng QUAN\_TRI: Ta đặt index trên cột TEN\_TK\_QT để quá trình đăng nhập diễn ra nhanh hơn.
* Ở bảng QUAN\_LI: Ta đặt index trên cột TEN\_TK\_QL để quá trình đăng nhập diễn ra nhanh hơn.
* Ở bảng NHAN\_VIEN: Ta đặt index trên cột TEN\_TK\_NV để quá trình đăng nhập diễn ra nhanh hơn.

# IV.Một số phần cài đặt trong Script

## IV.1 Trigger

Thực hiện cài đặt các trigger sau

* Số lần mua hàng(MAKH) = Số dòng lịch sử mua hàng (MAKH), ta cài đặt khi insert/update bảng KHACH\_HANG.
* PHI\_TAM\_TINH(MAGH) = Tổng GIA\_SP(MA\_SP) trong (MA\_GH), ta thực hiện khi insert/update bảng GIO\_HANG.
* TONG\_THANH\_TIEN = PHI\_TAM\_TINH \* GIAM\_GIA, ta thực hiện khi insert, update bảng GIO\_HANG

## IV.2 Phân quyền người dùng

Ta sử dụng các mô tả (business rules) để tiến hành phân quyền người dùng.

Phân hệ khách hàng

|  |  |
| --- | --- |
| **Bảng** | **Quyền** |
| SHOP | SELECT |
| SP\_SHOP | SELECT |
| SAN\_PHAM | SELECT |
| GIO\_HANG | SELECT |

Phân hệ quản trị

|  |  |
| --- | --- |
| **Bảng** | **Quyền** |
| PHIEU\_NHAP | SELECT,UPDATE |
| CT\_PHIEU\_NHAP | SELECT,UPDATE |
| PHIEU\_XUAT | SELECT,UPDATE |
| CT\_PHIEU\_XUAT | SELECT,UPDATE |
| KHO\_HANG | SELECT |
| CHI\_TIET\_KHO | SELECT |

Phân hệ quản lí

|  |  |
| --- | --- |
| **Bảng** | **Quyền** |
| NHAN\_VIEN | SELECT,UPDATE,DELETE |
| LICH\_SU\_LUONG | SELECT,UPDATE |
| CHI\_TIET\_NGAY\_LAM | SELECT,UPDATE |
| SHOP | SELECT,UPDATE |
| SP\_SHOP | SELECT,UPDATE |

Phân hệ nhân viên

|  |  |
| --- | --- |
| **Bảng** | **Quyền** |
| NHAN\_VIEN | SELECT,UPDATE |
| LICH\_SU\_LUONG | SELECT |
| CHI\_TIET\_NGAY\_LAM | INSERT |

# V.Link Source code và video

[Link GitHub chứa source code giao diện](https://github.com/buicongdanh/ADB1_8_Project2)

[Link GoogleDrive chứa video demo](https://drive.google.com/drive/folders/1-0g9jhecRJCpXmcQcf8ZN6YrqKyOzUGe?fbclid=IwAR2ZcMYN-IEag6TQJ4xSWjxoHHd-Zts2iDr3ceoDn650hRbzauxRzVcLS7I)

# VI.Phần trăm công việc từng người và ảnh báo cáo trên GITHUB

**Ảnh báo cáo GITHUB**

Graphical user interface, application

Description automatically generated

**Bảng phân công công việc:**

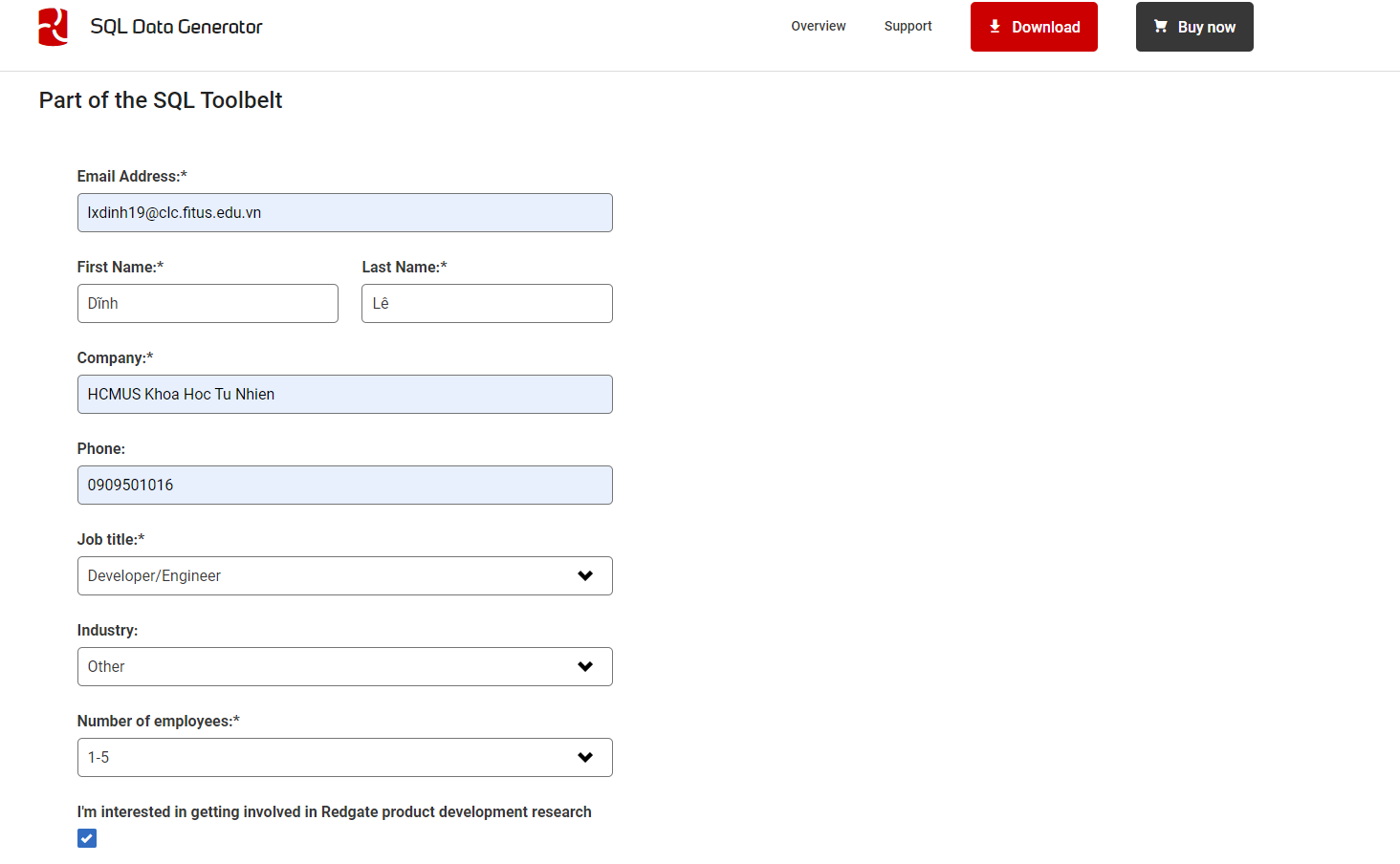
|  |  |
| --- | --- |
| Bùi Công Danh | 29% |
| Lê Xuân Dĩnh | 27% |
| Lê Hoàng Thịnh Phước | 15% |
| Tạ Công Điền | 29% |

# VII.Phụ lục ( Cách sử dụng sqlgen để tạo data)

Hướng dẫn fake data bằng công cụ hỗ trợ SQL Data Generator tool.

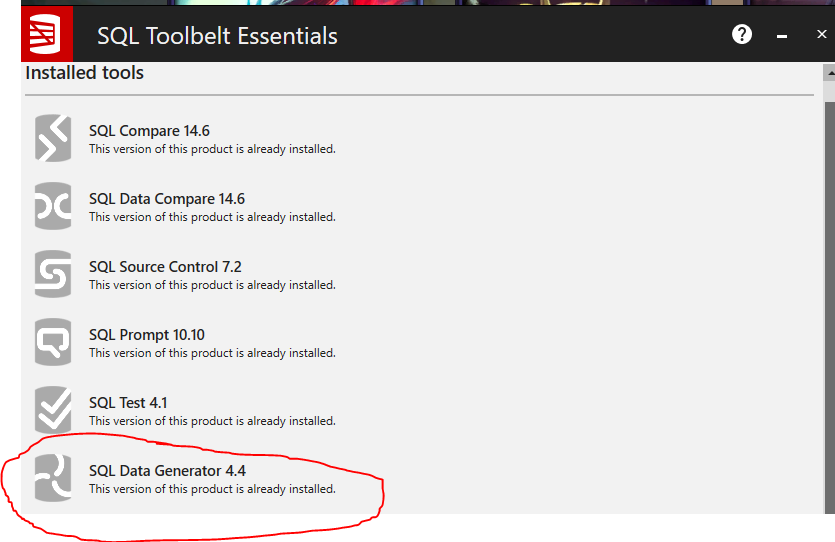
**CÀI ĐẶT**

Đầu tiên, ta tạo 1 account trên Red-Gate [tại link này](https://www.red-gate.com/products/sql-development/sql-data-generator/trial/).



Sau khi tạo xong, ta được tải 1 file SQL tool có tên SQL ToolbeltEssentials.exe

Thực hiện cài đặt từ file đó.



Chọn SQL Data Generator => Continue để hoàn tất quá trình cái đặt.

# VII. Tài liệu tham khảo

***1.***[***https://quantrimang.com/xu-ly-ngoai-le-try-catch-finally-trong-c-155848***](https://quantrimang.com/xu-ly-ngoai-le-try-catch-finally-trong-c-155848)

***2.***[***https://docs.microsoft.com/en-us/sql/relational-databases/partitions/partitioned-tables-and-indexes?view=sql-server-ver15***](https://docs.microsoft.com/en-us/sql/relational-databases/partitions/partitioned-tables-and-indexes?view=sql-server-ver15)

***3.*** [***https://hoayeuthuong.com/***](https://hoayeuthuong.com/)

***4. https://timoday.edu.vn/lap-trinh-co-so-du-lieu-voi-c/***