**ĐẠI HỌC ĐÀ NẴNG**

**TRƯỜNG ĐẠI HỌC KINH TẾ**

**Khoa Thống Kê – Tin Học**

=====🙥🕮🙧=====



**BÁO CÁO BÀI TẬP NHÓM**

**Học phần: Quản Trị Cơ Sở Dữ Liệu**

**Giáo viên hướng dẫn: Cao Thị Nhâm**

**Sinh viên thực hiện: Nhóm 14**

* + - * 1. **Nguyễn Thị Hiền Lương 43K21**
        2. **Trương Phú Tiên 43K21**

***Đà Nẵng, ngày 10 tháng 11 năm 2019***

**MỤC LỤC**

[1 Hóa đơn được chọn 3](#_Toc24295298)

[2 Thiết kế cơ sở dữ liệu 4](#_Toc24295299)

[1.1. Thiết kế cơ sở dữ liệu mức khái niệm 4](#_Toc24295300)

[1.2. Thiết kế cơ sở dữ liệu mức logic 5](#_Toc24295301)

[1.3. Thiết kế cơ sở dữ liệu mức vật lí 6](#_Toc24295302)

[3 Xây dựng cơ sở dữ liệu 8](#_Toc24295303)

[4 Xác định và tạo index cho các thuộc tính 9](#_Toc24295304)

[5 Xây dựng cơ chế bảo mật tương ứng cho cơ sở dữ liệu vừa tạo 10](#_Toc24295305)

[6 Xây dựng cơ chế backup dữ liệu phù hợp cho cơ sở dữ liệu đã tạo 11](#_Toc24295306)

[7 Phát triển ứng dụng (Dùng phần mềm C#) 13](#_Toc24295307)

[7.1 Form đăng nhập 13](#_Toc24295308)

[7.2 Thành phần và chức năng của Form quản lý 14](#_Toc24295310)

[7.3 Form quản lý Khách hàng 15](#_Toc24295311)

[7.4 Form quản lý Nhân viên 16](#_Toc24295312)

[7.5 Form quản lý Hàng hóa 17](#_Toc24295313)

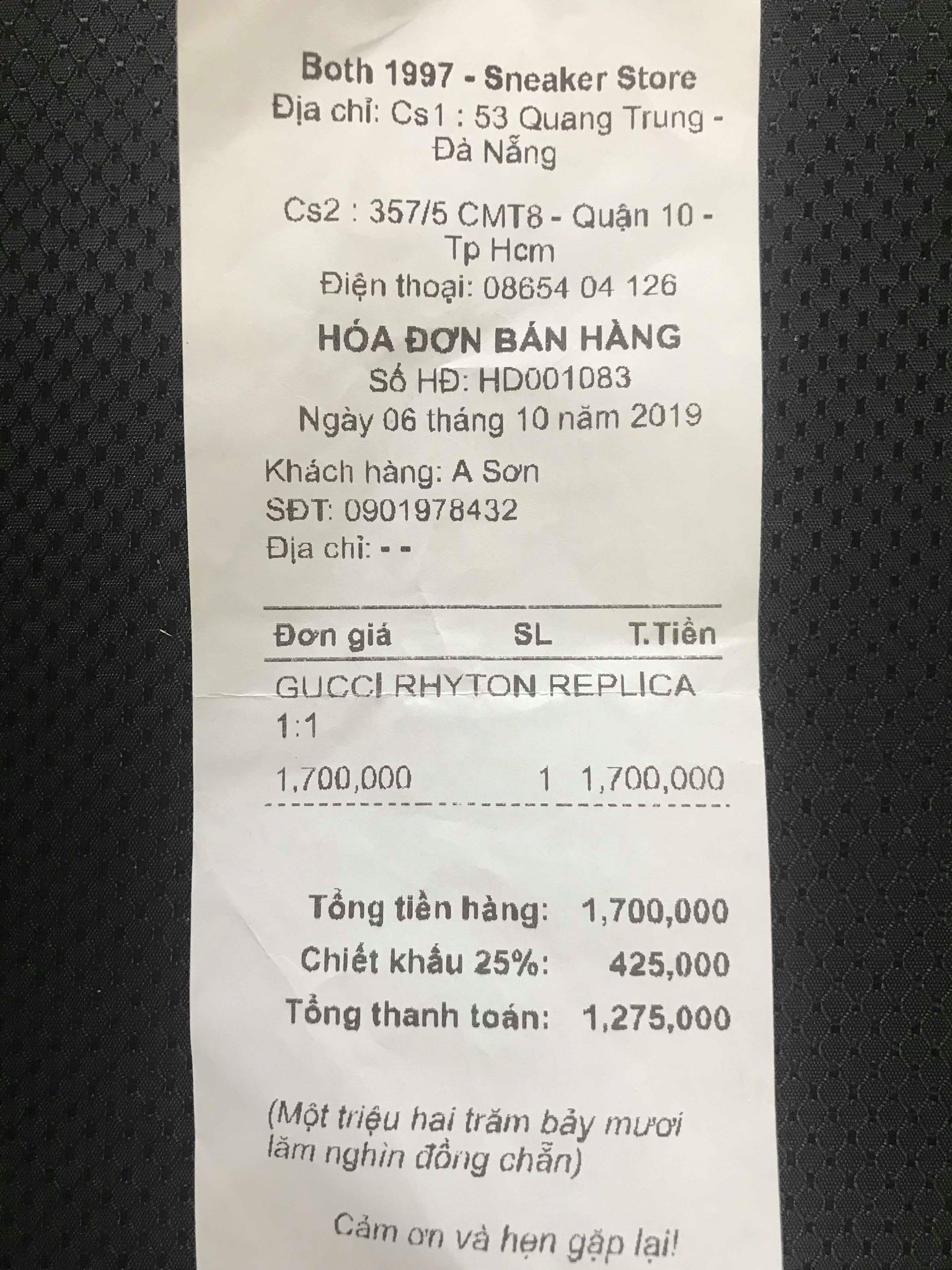
[7.6 Form quản lý Hóa đơn 18](#_Toc24295314)

[7.7 Form quản lý Chi tiết hóa đơn 19](#_Toc24295315)

[8 Phương án giải quyết vấn đề dữ liệu lớn 20](#_Toc24295316)

[9 Đề phòng phương án hệ thống bị tấn công bằng SQL Injection. 20](#_Toc24295317)

# Hóa đơn được chọn



Hình 1.1. Hóa đơn bán quần áo

Đây là hóa đơn mua hàng của một cửa hàng bán giày thể thao mang tên Both 1997 – Sneaker Store tại Cơ sở 1 : 53 Quang Trung, Đà Nẵng

# Thiết kế cơ sở dữ liệu

## Thiết kế cơ sở dữ liệu mức khái niệm

Xây dựng ER cho hóa đơn bán hàng

|  |  |
| --- | --- |
| **Từ trong hóa đơn** | **Viết tắt** |
| Mã khách hàng | MaKH |
| Tên khách hàng | TenKH |
| Số điện thoại khách hàng | SDT |
| Địa chỉ khách hàng | DiaChi |
| Mã nhân viên | MaNV |
| Tên nhân viên | TenNV |
| Username | Username |
| Password | Pass |
| Mã hóa đơn | MaHD |
| Ngày bán hàng | NgayBan |
| Tổng tiền hàng (Chưa chiết khấu) | TongTienHang |
| Chiết khấu | CK |
| Tổng thanh toán | TongThanhToan |
| Mã hàng hóa | MaHH |
| Tên hàng hóa | TenHH |
| Số Lượng | SL |
| Đơn giá | DonGia |
| Số lượng bán | SLBan |

* ***Các thực thể***
* HANGHOA (**MaHH**, TenHH, SL, DonGia)
* KHACHHANG (**MaKH**, TenKH, SDT, DiaChi)
* NHANVIEN (**MaNV**, TenNV, Username, Pass)
* ***Quan hệ***
* BAN (**MaHD**, ThanhTien, TongTienHang, NgayBan, CK, SLBan , TongThanhToan)

Hình 2.1**:** Sơ đồ ERD của hệ thống

**HANG HOA**

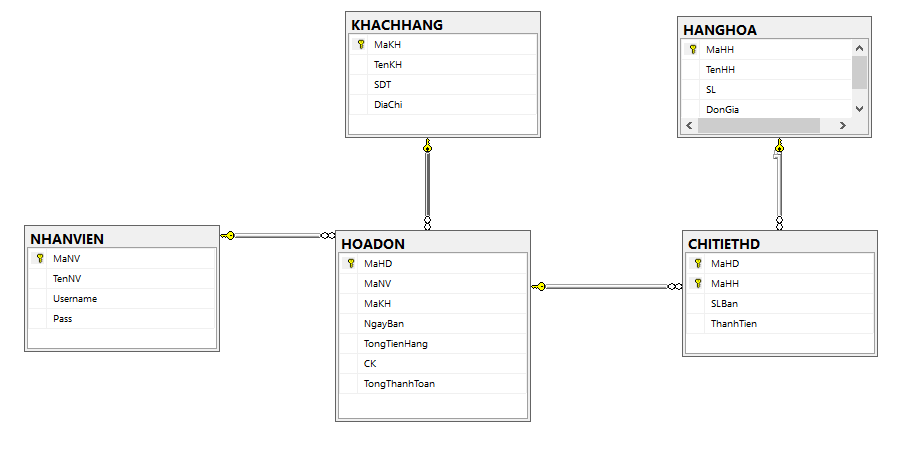
**KHACH HANG**

**BÁN**

**NHAN VIEN**

## Thiết kế cơ sở dữ liệu mức logic

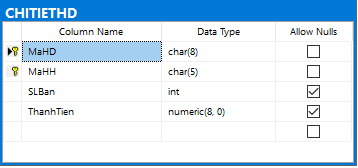
* **Chuyển thực thể, ta có:**
* HANGHOA (**MaHH**, TenHH, DonGia, SL)
* KHACHHANG (**MaKH**, TenKH, SDT, DiaChi)
* NHANVIEN (**MaNV**, TenNV, Username, Pass)
* **Chuyển quan hệ, ta có:**
* BAN (**MaKH**, **MaHH**, **MaHD**, SLBan, ThanhTien, TongTienHang, CK, TongThanhToan, NgayBan)
* **Chuẩn hóa quan hệ BAN:**
* HOADON (**MaHD**, MaKH, NgayBan, TongTienHang, CK, TongThanhToan)
* ChiTietHD (**MaHD**, **MaHH**, SLBan, ThanhTien)
* **Sơ đồ các ràng buộc quan hệ**



Hình 2.2. Sơ đồ ràng buộc các quan hệ

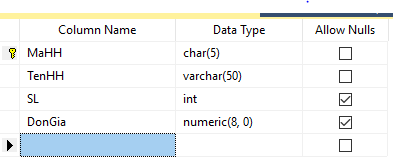
## Thiết kế cơ sở dữ liệu mức vật lí

**Bảng Chi tiết hóa đơn**



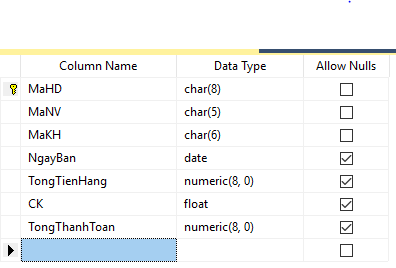
Hình 2.3. Bảng Chi Tiết Hóa Đơn

**Bảng Hàng hóa**



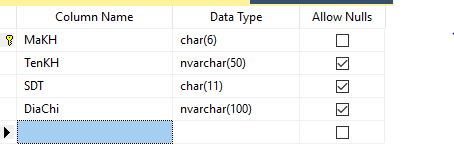
Hình 2.4. Bảng Hàng hóa

**Bảng Hóa đơn**



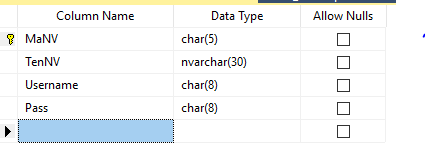
Hình 2.5. Bảng Hóa Đơn

**Bảng Khách hàng**



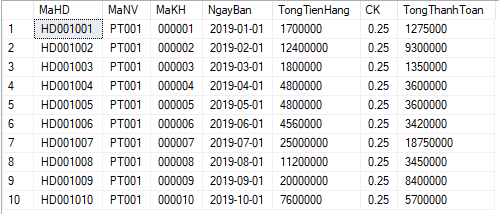
Hình 2.5. Bảng Khách hàng

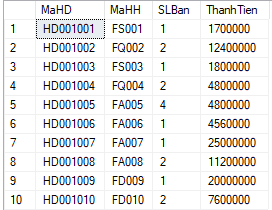
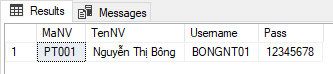
**Bảng Nhân viên**



Hình 2.6. Bảng Nhân viên

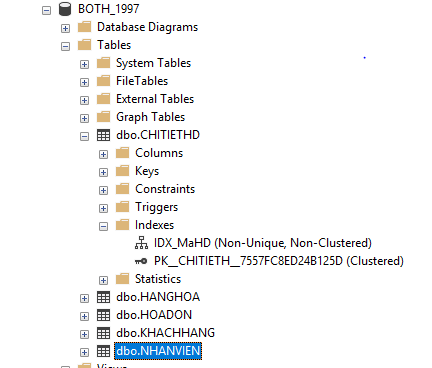
# 3 Xây dựng cơ sở dữ liệu





# 4 Xác định và tạo index cho các thuộc tính

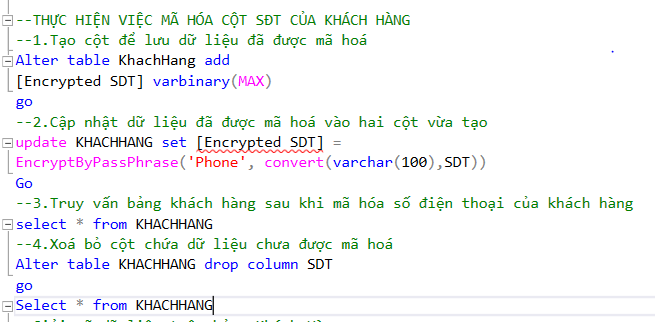
* Tạo index cho các thuộc tính trong cơ sở dữ liệu giúp tăng tốc độ truy vấn dữ liệu trong các bảng. Tuy nhiên, nó cũng làm chậm cho tốc độ update, insert, delete dữ liệu trong cơ sở dữ liệu.
* Do đó, khi thực hiện tạo index cho các thuộc tính trong database chúng ta chỉ nên tạo index cho các cột phù hợp trong bảng.
* Trong cơ sở dữ liệu của mình, nhóm đã tạo Non-Clustered index cho cột MaHD trong bảng ChiTietHD.
* Kết quả, sau khi tạo index thành công:



Hình 4.1. Kết quả Index

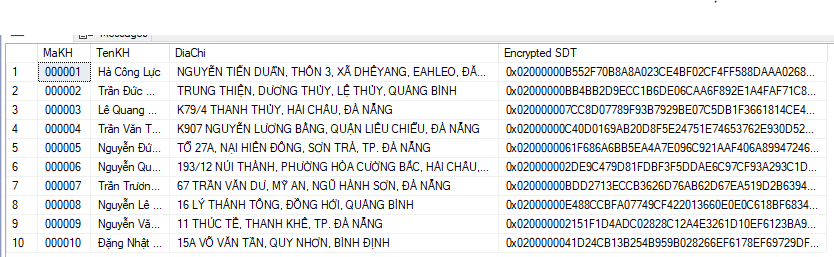
# Xây dựng cơ chế bảo mật tương ứng cho cơ sở dữ liệu vừa tạo

* Mã hoá là một phương pháp quan trọng nhằm bảo mật dữ liệu. Những dữ liệu nhạy cảm như số CMT, số thẻ tín dụng, mật khẩu và thông tin cá nhân như số điện thoại khách hàng, cần phải được bảo vệ trước vô vàn mối nguy hiểm tấn công hiện nay.
* Nhóm đã sử dụng phương pháp mã hóa đối xứng 2 chiều cho cơ sở dữ liệu vừa tạo bằng hàm mã hóa 2 chiều EncryptByPassPhrase trong SQL server.
* Thực hiện việc mã hóa thông tin của cột số điện thoại trong bảng Khách Hàng thông qua phương pháp mã hóa bằng mật khẩu:



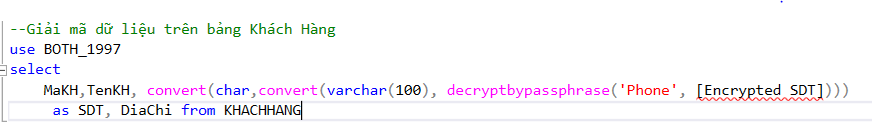
Hình 5.1. Thực hiện mã hóa

* Sau khi mã hóa xong, truy vấn dữ liệu trong bảng KHACHHANG, ta được kết quả sau:

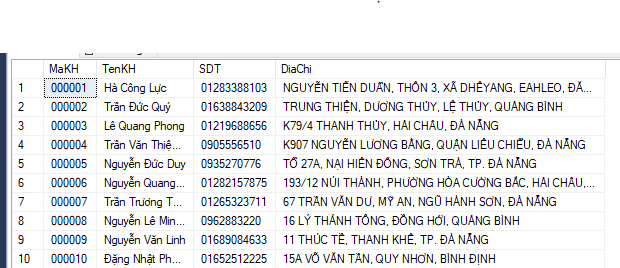


Hình 5.2. Kết quả mã hóa

* Giải mã dữ liệu trên bảng thông qua hàm Decryptbypassphrase



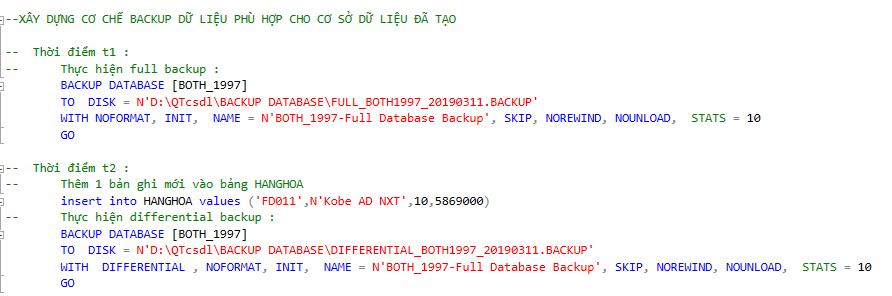
* Kết quả sau khi giải mã thành công cột SĐT trong bảng KHACHHANG:



Hình 5.3. Kết quả giải mã dữ liệu

# Xây dựng cơ chế backup dữ liệu phù hợp cho cơ sở dữ liệu đã tạo

* Các loại sao lưu trong SQL Server cung cấp:
* ***Full Database Backups:*** Copy tất cả các file dữ liệu trong một database. Tất cả những user data và database objects như system tables, indexes, user-defined tables đều được backup.
* ***Differential Database Backups:*** Copy những thay đổi trong tất cả các file dữ liệu kể từ lần backup gần nhất.
* ***File or File Group Backups:*** Copy một data file đơn hay một nhóm file.
* ***Differential File or File Group Backups:*** Tương tự như *differential database backup* nhưng chỉ copy những thay đổi trong data file đơn hay một file group.
* ***Transaction Log Backups:*** Ghi nhận một cách thứ tự tất cả các giao dịch (transaction) chứa trong file transaction log kể từ lần transaction log backup gần nhất. Loại sao lưu này cho phép ta phục hồi dữ liệu trở ngược lại vào một thời điểm nào đó trong quá khứ mà vẫn đảm bảo tính đồng nhất (consistent).
* Nhóm đã sử dụng 3 loại backup dữ liệu: full backup, different backup, transaction log backup.
* Sử dụng tiện ích Jobs trong SQL Server Agent để lập lịch Backup dữ liệu.
* Tiện ích Jobs trong SQL Server Agent giúp cho việc tạo lịch backup dữ liệu được thực hiện tự động một cách dễ dàng. SQL Server Agent là module trong SQL Server dùng để tự động thực hiện các tác vụ về bảo trì và giám sát hệ thống.
* Xây dựng kịch bản sao lưu và khôi phục dữ liệu cho cơ sở dữ liệu đã tạo như sau:
* Thực hiện Full backup dữ liệu vào 1 giờ sáng mỗi ngày.
* Thực hiện Differential backup: vào các thời điểm 10h, 14h, 18h, 22h (4 lần/ngày).
* Thực hiện transaction log backup 15 phút một lần vào các thời điểm 5′, 20′, 35′, và 50′ của mỗi giờ (4 lần/giờ).
* Ví dụ về thực hiện backup dữ liệu trong cơ sở dữ liệu đã tạo:



Hình 6.1. Backup dữ liệu

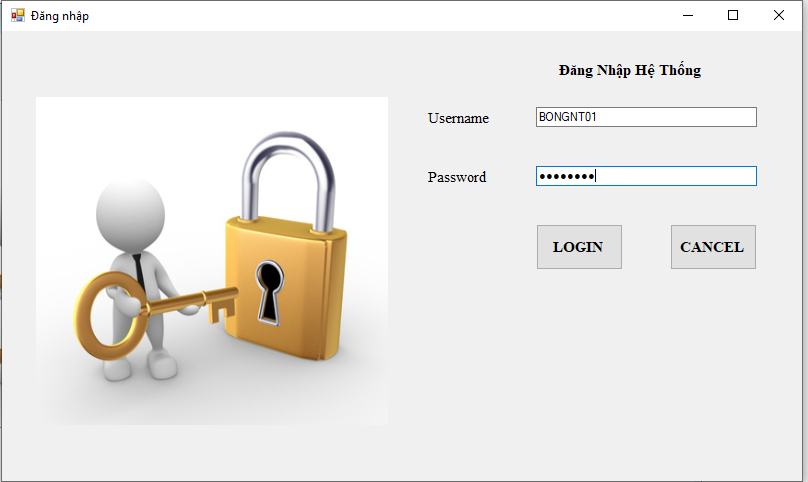
* Nếu backup thành công thì khi vào đường dẫn file kiểm tra, ta sẽ thấy các file sau:



Hình 6.2. Kết quả backup dữ liệu

# Phát triển ứng dụng (Dùng phần mềm C#)

## 7.1 Form đăng nhập



Hình 7.1. Form đăng nhập

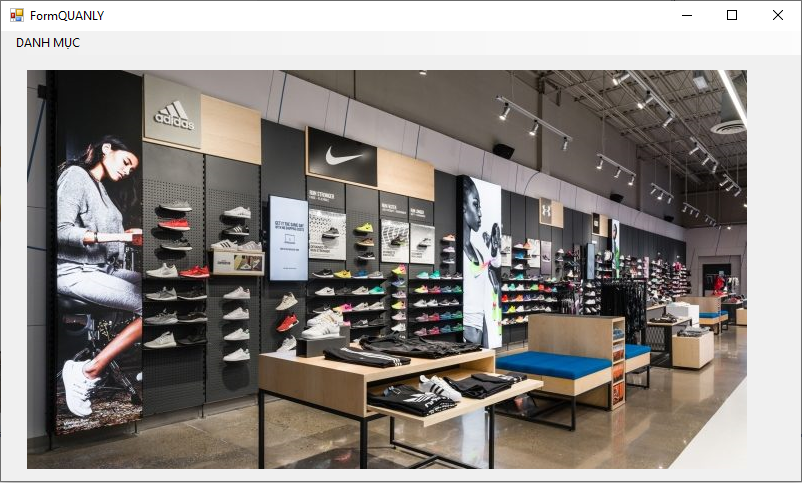
## 7\

Các thành phần và chức năng của Form đăng nhập

|  |  |
| --- | --- |
| Thành phần | Chức năng |
| username | Tên đăng nhập và mật khẩu có quyền truy cập được quy định ở bảng TaiKhoan trong CSDL. Người dùng sẽ nhập tên đăng nhập và mật khẩu vào các ô. |
| password |
| Nút cancel | Đóng form |
| Nút login | Khi click vào nút đăng nhập, nếu đúng sẽ được phép truy cập vào chương trình, ngược lại sẽ được thông báo tên đăng nhập và mật khẩu không đúng. |

Bảng 7.1. Thành phần và chức năng của Form đăng nhập

7.2 Thành phần và chức năng của Form quản lý



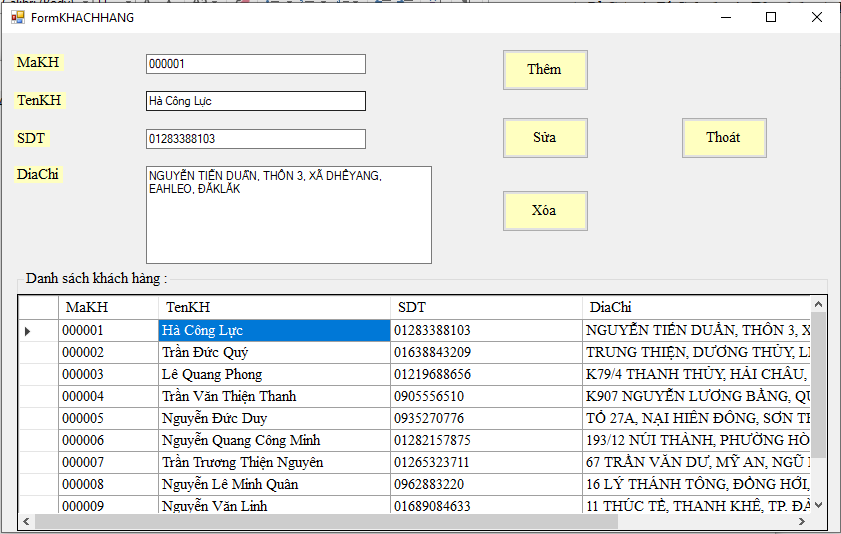
Hình 7.2. Form quản lý

Thành phần và chức năng của Form quản lý

|  |  |
| --- | --- |
| Thành phần | Chức năng |
| Danh mục | Lựa chọn các Form quản lý (Hàng hóa, Hóa đơn, Chi tiết hóa đơn, Khách hàng). |

Bảng 7.2. Thành phần và chức năng của Form quản lý

## 7.3 Form quản lý Khách hàng



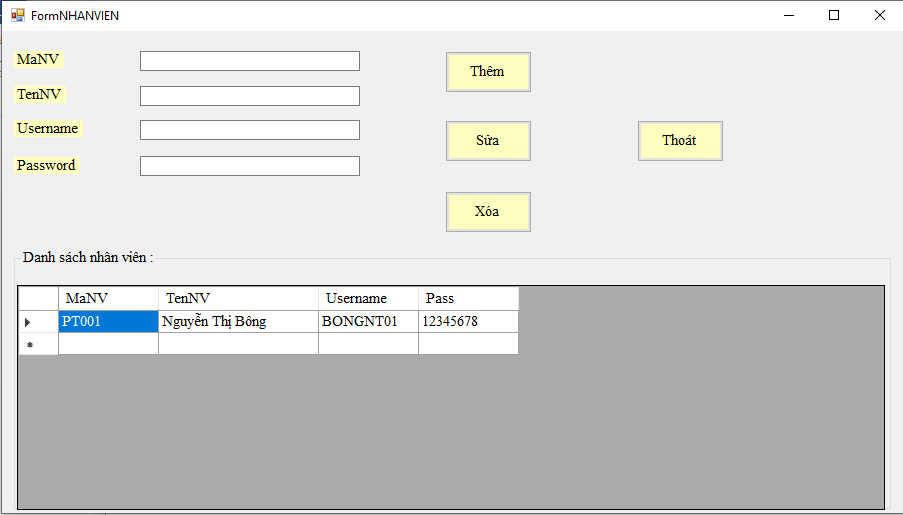
Hình 7.3. Form quản lý Khách Hàng

Thành phần và chức năng của Form quản lý Khách hàng

|  |  |
| --- | --- |
| Thành phần | Chức năng |
| Các textbox để nhập giá trị | Người dùng nhập các giá trị vào các ô để thực hiện các chức năng (Thêm, Sửa, Xóa). |
| Nút Thêm | Nhập vào mã khách và các giá trị tưởng ứng để thêm mới bản ghi vào bảng KhachHang |
| Nút Sửa | Cho phép sửa tên, số điện thoại và điểm đã tích lũy. |
| Nút Xóa | Nhập vào mã khách và thực hiện xóa bản ghi tương ứng. |
| Nút Thoát | Đóng cửa sổ. |

Bảng 7.3. Thành phần và chức năng của Form quản lý Khách hàng

## 7.4 Form quản lý Nhân viên



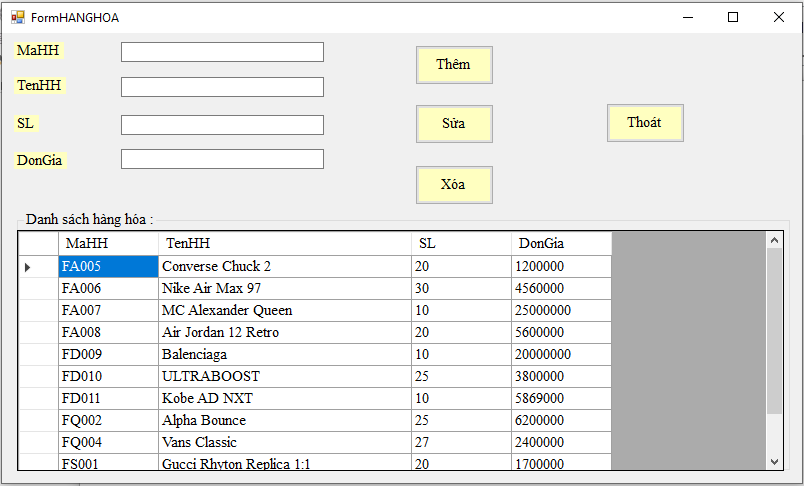
Hình 7.4. Form quản lý Nhân viên

Thành phần và chức năng của Form quản lý Nhân viên

|  |  |
| --- | --- |
| Thành phần | Chức năng |
| Các textbox để nhập giá trị | Người dùng nhập các giá trị vào các ô để thực hiện các chức năng (Thêm, Sửa, Xóa). |
| Nút Thêm | Nhập vào mã nhân viên và các giá trị tưởng ứng để thêm mới bản ghi vào bảng NHANVIEN |
| Nút Sửa | Cho phép sửa mã hàng hóa, tên nhân viên, tên đăng nhập và mật khẩu |
| Nút Xóa | Nhập vào mã nhân viên và thực hiện xóa bản ghi tương ứng. |
| Nút Thoát | Đóng cửa sổ. |

Bảng 7.4 Thành phần và chức năng của form quản lý Nhân viên

## 7.5 Form quản lý Hàng hóa



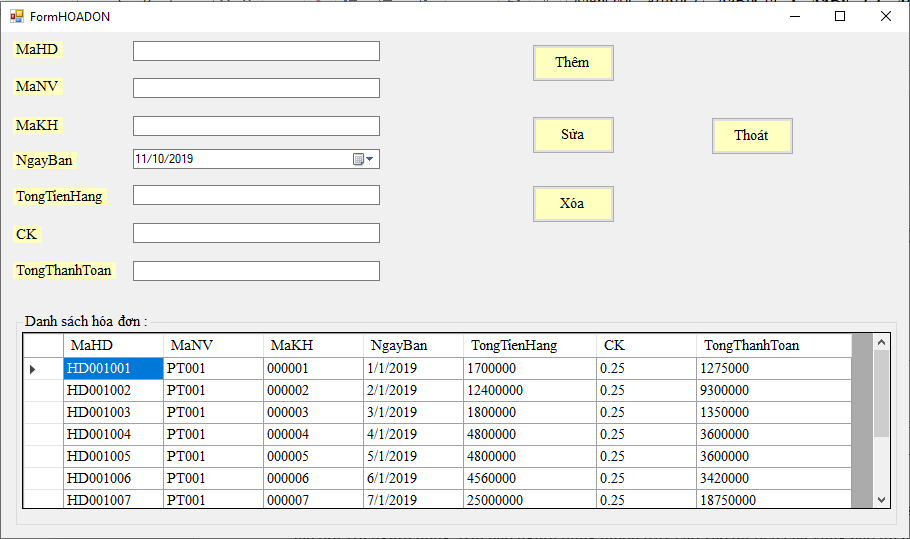
Hình 7.5. Form quản lý hàng hóa

Thành phần và chức năng của Form quản lý hàng hóa

|  |  |
| --- | --- |
| Thành phần | Chức năng |
| Các textbox để nhập giá trị (Mã hàng hóa, Tên hàng, Đơn giá, Số lượng) | Người dùng nhập các giá trị vào các ô để thực hiện các chức năng (Thêm, Sửa, Xóa). |
| Nút Thêm | Nhập tất cả các giá trị vào các textbox rồi nhấn Thêm, một bản ghi sẽ được ghi mới vào bảng HANGHOA |
| Nút Sửa | Nhập các giá trị vào textbox rồi nhấn Sửa, bảng ghi tương ứng với mã Hàng sẽ được thay đổi.  Đồng thời giá của Hàng hóa trong bảng ChiTietHD sẽ được cập nhật lại. |
| Nút Xóa | Nhập mã Hàng vào textbox tương ứng rồi nhấn Xóa, bản ghi tương ứng sẽ bị xóa đi.  Đồng thời các mặt hàng ở bảng ChiTietHD tương ứng sẽ bị xóa. |
| Nút Thoát | Đóng cửa sổ. |

Bảng 7.5. Thành phần và chức năng của form Quản lý hàng hóa

## 7.6 Form quản lý Hóa đơn



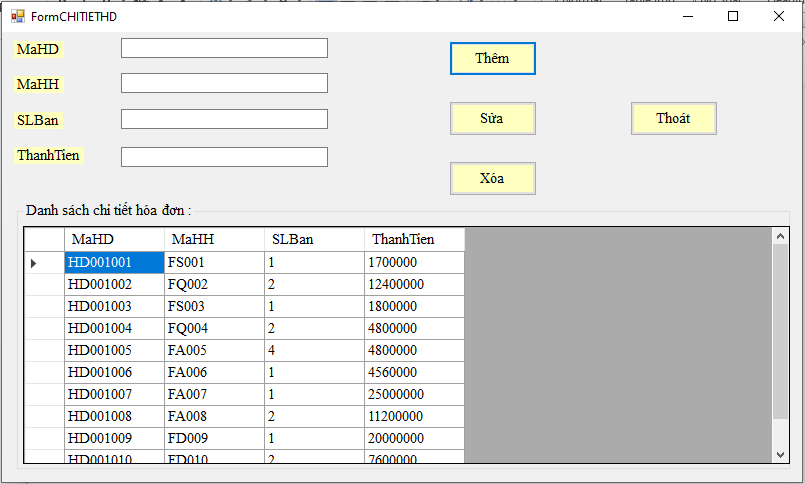
Hình 7.6. Form quản lý Hóa đơn

Thành phần và chức năng của Form Quản lý Hóa đơn

|  |  |
| --- | --- |
| Thành phần | Chức năng |
| Các textbox để nhập giá trị (Mã hóa đơn, Mã nhân viên, Ngày bán, Tổng tiền hàng, Chiết khấu, Tổng thanh toán) | Người dùng nhập các giá trị vào các ô để thực hiện các chức năng (Thêm, Sửa, Xóa). |
| Nút Thêm | Người dùng nhập vào Mã hóa đơn, Mã khách, Ngày bán, Tổng tiền hàng, Chiết khấu, để thêm vào bảng HoaDon. Tổng thanh toán sẽ được tính từ bảng ChiTietHD. Các giá trị còn lại sẽ được Update tương ứng. |
| Nút Sửa | Tương tự như nút Thêm, nút Sửa chỉ thay đổi giá trị của Mã khách hàng, Ngày bán |
| Nút Xóa | Nhập vào mã hóa đơn sẽ thực hiện Xóa bảng ghi tương ứng.  Đồng thời xóa ở bảng ChiTietHD các bản ghi tương ứng có mã Hóa đơn đó. |
| Nút Thoát | Đóng cửa sổ. |

Bảng 7.6. Thành phần và chức năng của Form quản lý Hóa đơn

## 7.7 Form quản lý Chi tiết hóa đơn



Hình 7.7. Form Quản lý Chi tiết hóa đơn

Thành phần và chức năng của Form Quản lý Chi tiết hóa đơn

|  |  |
| --- | --- |
| Thành phần | Chức năng |
| Các textbox để nhập giá trị | Người dùng nhập các giá trị vào các ô để thực hiện các chức năng (Thêm, Sửa, Xóa). |
| Nút Thêm | Nhập vào Mã hóa đơn, Mã hàng hóa và Số lượng bán để thực hiện thêm bản ghi vào bảng ChiTietHD.  Kiểm tra số lượng hàng được thêm có phù hợp với lượng hàng hiện tại ở bảng HANG không?  Giá bán được lấy từ bảng HANGHOA, sau đó Update Thành tiền tương ứng.  Đồng thời, Update lại tổng tiền và các giá trị ở bảng HOADON tương ứng. |
| Nút Sửa | Nhập vào Mã hóa đơn, sửa các giá trị Mã hàng hóa và số lượng bán  Tương tự, Giá bán được cập nhật lại từ bảng HANGHOA và Update lại Thành tiền. Đồng thời, các giá trị tương ứng ở bảng HoaDon sẽ được update lại. |
| Nút Xóa | Nhập vào Mã hóa đơn và Mã hàng hóa, Xóa bảng ghi tương ứng.  Đồng thời, Update lại tổng tiền và các giá trị ở bảng HOADON tương ứng. |
| Nút Thoát | Đóng cửa sổ. |

Bảng 7.7. Thành phần và chức năng của Form Quản lý Chi tiết hóa đơn

# Phương án giải quyết vấn đề dữ liệu lớn

* Vấn đề cần giải quyết: Giả sử lượng dữ liệu rất lớn, không đủ lưu trong một ổ đĩa.
* Phương án: Sử dụng cơ sở dữ liệu phân tán

Tổng hợp tất cả các tập dữ liệu có quan hệ logic với nhau, lưu trữ chúng trên các máy tính của một mạng máy tính để hệ thống biết địa chỉ IP của tất cả các máy lưu trữ. (Yêu cầu dữ liệu phải sắp xếp theo thứ tự và có chỉ mục cụ thể)

Hệ thống phần mềm sẽ cho phép quản lý CSDL phân tán và đảm bảo tính trong suốt về sự phân tán đối với người dùng. Khi nào người dùng muốn truy cập vào dữ liệu của vùng nào thì hệ thống sẽ kết nối tới địa chỉ IP của máy đang chứa dữ liệu đó, tới server cơ sở dữ liệu đang chạy trong đó.

* Ưu điểm của CSDL phân tán:
* Phù hợp với cấu trúc của tổ chức.
* Nâng cao khả năng chia sẻ và tính tự trị địa phương.
* Nâng cao tính sẵn sàng.
* Nâng cao tính tin cậy.
* Nâng cao hiệu năng.
* Dễ mở rộng.
* Nhược điểm của CSDL phân tán:
* Thiết kế CSDL phức tạp hơn.

(Trên đây là phương án đề xuất, nhưng nhóm 14 vẫn chưa thể áp dụng được vào cơ sở dữ liệu của mình.)

# Đề phòng phương án hệ thống bị tấn công bằng SQL Injection

**SQL Injection** là một kỹ thuật lợi dụng những lỗ hổng về câu truy vấn lấy dữ liệu của những website không an toàn trên web, đây là một kỹ thuật tấn công rất phổ biến và sự thành công của nó cũng tương đối cao.

Để bảo vệ bản thân trước SQL Injection, ta có thể thực hiện các biện pháp sau :

* **Lọc dữ liệu từ người dùng:** Cách phòng chống này tương tự như XSS. Ta sử dụng filter để lọc các kí tự đặc biệt (; ” ‘) hoặc các từ khoá (SELECT, UNION) do người dùng nhập vào. Nên sử dụng thư viện/function được cung cấp bởi framework. Viết lại từ đầu vừa tốn thời gian vừa dễ sơ sót.
* **Không cộng chuỗi để tạo SQL:** Sử dụng parameter thay vì cộng chuỗi. Nếu dữ liệu truyền vào không hợp pháp, SQL Engine sẽ tự động báo lỗi, ta không cần dùng code để check.
* **Không hiển thị exception, message lỗi:** Hacker dựa vào message lỗi để tìm ra cấu trúc database. Khi có lỗi, ta chỉ hiện thông báo lỗi chứ đừng hiển thị đầy đủ thông tin về lỗi, tránh hacker lợi dụng.
* **Phân quyền rõ ràng trong DB:** Nếu chỉ truy cập dữ liệu từ một số bảng, hãy tạo một account trong DB, gán quyền truy cập cho account đó chứ đừng dùng account root hay sa. Lúc này, dù hacker có inject được sql cũng không thể đọc dữ liệu từ các bảng chính, sửa hay xoá dữ liệu.
* **Backup dữ liệu thường xuyên:** Các cụ có câu “cẩn tắc vô áy náy”. Dữ liệu phải thường xuyên được backup để nếu có bị hacker xoá thì ta vẫn có thể khôi phục được.