BỘ CÔNG THƯƠNG

**TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP THỰC PHẨM TP. HCM**

**KHOA CÔNG NGHỆ THÔNG TIN**

---------------------------



**Quản lý Cafe**

**SINH VIÊN THỰC HIỆN**

1. 2001190451, Trần Hữu Danh, Lớp 10DHTH2

2. 2001190548, Trương chí Hoàng,Lớp 10DHTH3

3. 2001190474, Võ Thanh Điền, Lớp 10DHTH8

4.2001190441, Huỳnh Hoàng Chương ,Lớp 10DHTH8

**GVHD: Nguyễn Văn Lễ**

**BÁO CÁO TIỂU LUẬN**

Môn học: Hệ quản trị cơ sở dữ liệu

TP. HỒ CHÍ MINH, tháng 08 năm 2022

**MỤC LỤC**

[CHƯƠNG 1: GIỚI THIỆU 2](#_Toc111345450)

[CHƯƠNG 2: thiết kế VÀ CÀI ĐẶT cơ sở dữ liệu 3](#_Toc111345451)

[2.1 Diagram 3](#_Toc111345452)

[2.1.1 Account 3](#_Toc111345453)

[2.1.2 Food 3](#_Toc111345454)

[2.1.3 TABLEF 4](#_Toc111345455)

[2.1.4 CATEGORY 4](#_Toc111345456)

[2.1.5 Bill 4](#_Toc111345457)

[2.1.6 VITRI 4](#_Toc111345458)

[2.1.7 GIOITINH 5](#_Toc111345459)

[2.1.8 NHANVIEN 5](#_Toc111345460)

[2.2 Các Ràng Buộc 6](#_Toc111345461)

[2.3 Nhập dữ liệu 8](#_Toc111345462)

[2.4 Các fuction,proc,.. 10](#_Toc111345463)

[CHƯƠNG 3: CÀI ĐẶT ỨNG DỤNG MINH HỌA 24](#_Toc111345464)

[3.1 Giao diện (admin) 24](#_Toc111345465)

[3.1.1 Form đăng nhập 24](#_Toc111345466)

[3.1.2 Form quản lý nhân viên 25](#_Toc111345467)

[3.1.3 Form thay đổi thông tin tài khoản 25](#_Toc111345468)

[3.1.4 Form quản lý nhân viên 26](#_Toc111345469)

[3.1.5 Form quản lý bàn 26](#_Toc111345470)

[3.2 Chức năng danh mục 26](#_Toc111345471)

[3.3 Chức năng thống kê 27](#_Toc111345472)

[3.4 Chức năng tìm kiếm 28](#_Toc111345473)

[3.5 Chức năng nghiệp vụ 29](#_Toc111345474)

[3.6 Cài đặt chức năng phân quyền người dùng 30](#_Toc111345475)

[DANH MỤC TÀI LIỆU THAM KHẢO 32](#_Toc111345476)

1. GIỚI THIỆU

Hiện nay, Công Nghệ Thông Tin của quốc gia, đặc biệt là các nước đang phát triển, tiến hành công nghiệp hóa hiện đại hóa như nước ta. Sự bùng nổ thông tin và sự phát triển mạnh mẽ của công nghệ kỹ thuật số, muốn phát triển thì phải áp dụng tin học hóa vào tất cả các ngành lĩnh vực.

Bên cạnh đó cùng với sự phát triển nhanh chóng về phần cứng và phần mềm, đã hổ trợ cho con người trong công việc hiệu quả hơn. Các phần mềm hiện nay ngày càng dễ sử dụng và hỗ trợ con người trong việc xử lý công việc nhanh hơn.

Do vậy mà việc phát triển phần mềm đòi hỏi sự chính xác, xử lý nhiều việc, nhanh chóng và thân thiện. Ngoài ra người dùng còn đòi hỏi về tính bảo mật cao, tương tích tốt để người sử dụng có thể sử dụng an toàn,…. Các phần mềm giúp tiết kiệm về mặt thời gian, công sức, tăng độ chính xác và hiệu quả trong công việc.

Cụ thể, trong việc quản lý quán cafe. Nếu không sử dụng phần mềm thì việc quản lý sẽ khó khăn đối với nhiều quán khi mà lượng nhân viên và công việc quá nhiều sẽ dẫn đến sai sót trong công việc. Với sự giúp đỡ của phần mềm thì có thể kiểm soát chặt chẽ vì nó sẽ được chia thành nhiều khâu: quản lý nhân viên, quản lý hóa đơn,… Các công việc này đòi hỏi nhiều thời gian và sự chính xác. Ngoài ra việc lưu trữ cũng sẽ dễ dàng hơn tránh bị thất lạc, tốn kém,…

Tại các quán café hiện nay với lượng khách ngày càng tăng, để có thể phục vụ một cách hiệu quả hơn, thì cửa hàng đã tin học hóa các khâu quản lý. Đặt biệt là trong công tác và quản lý hàng hóa. Bởi vì công tác thủ công mà quán đang thực hiện sẽ bộc lộ nhiều hạn chế như:

* Kiểm tra hàng hóa sẽ mất nhiều thời gian và kém chính xác.
* Lưu trữ thông về nhập xuất hàng hóa, về thu chi cần nhiều loại giấy tờ cồng kềnh và kém hiệu quả.
* Cập nhật thêm thông tin hàng ngày tốn nhiều thời gian.
* Khó khăn trong quá trình thực hiện báo cáo thống kê, nhất là khi có sự cố đột xuất.

Trước tình hình đó thì vấn đề được nhóm chúng em đặt ra là cần xây dựng một hệ thống có thể đáp ứng những nhu cầu sau:

* Lưu trữ được lượng lớn thông tin không cần ghi giấy.
* Cập nhật dữ liệu nhanh chóng
* Quản lý nhân viên
* Thống kê lương của nhân viên
* In hóa đơn bán hàng một cách nhanh chóng
* Tính tiền

1. thiết kế VÀ CÀI ĐẶT cơ sở dữ liệu

- Với đề tài nhóm đã chọn, thiết kế cơ sở dữ liệu khoảng 5 bảng dữ liệu chính. mô tả thuộc tính cho từng bảng dữ liệu.

* + 1. Diagram

**Graphical user interface, diagram

Description automatically generated**

* + - 1. Account

Bảng chứa thông tin đăng nhập của các quản trị viên và nhân viên.

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên thuộc tính** | **Giải thích** | **Kiểu dữ liệu** | **Khóa** |
| UserID | Tên đăng nhập | Nvarchar(25) | Primary key |
| UserPassword | Mật khẩu | Nvarchar(25) |  |
| DISPLAYNAME | Tên hiển thị | Nvarchar(25) |  |
| TYPEACCOUNT | Kiểu Tài khoản | Nvarchar(10) |  |

* + - 1. Food

Thông tin món ăn và thông tin tên loại

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên thuộc tính** | **Giải thích** | **Kiểu dữ liệu** | **Khóa** |
| NAME | Tên món | Nvarchar(40) | Primary key |
| NAMECATEGORY | Tên loại | Nvarchar(40) | Foreign key |
| PRICE | Giá bán | Float |  |

* + - 1. TABLEF

Thông tin bàn

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên thuộc tính** | **Giải thích** | **Kiểu dữ liệu** | **Khóa** |
| NAME | Tên bàn | Nvarchar(40) | Primary key |
| STT | Số thứ tự | Nvarchar(40) |  |
| TOTAL | Thành tiền | Float |  |

* + - 1. CATEGORY

Thông tin loại món ăn

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên thuộc tính** | **Giải thích** | **Kiểu dữ liệu** | **Khóa** |
| NAME | Tên loại món ăn | Nvarchar(40) | Primary key |

* + - 1. Bill

Bảng chứa thông tin về hóa đơn

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên thuộc tính** | **Giải thích** | **Kiểu dữ liệu** | **Khóa** |
| ID | Mã hóa đơn | Varchar(10) | Primary key |
| NAMETABLE | Tên bàn | Nvarchar(40) | Foreign key |
| NAMEFOOD | Tên món | Nvarchar(40) | Foreign key |
| COUNTS | Số lượng | Int |  |

* + - 1. VITRI

Chứa các vị trí làm việc của nhân viên

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên thuộc tính** | **Giải thích** | **Kiểu dữ liệu** | **Khóa** |
| MaVT | Mã vị trí | Int identity(1,1) | Primary key |
| TenVT | Tên vị trí | Nvarchar(15) |  |

* + - 1. GIOITINH

Chứa thông tin giới tính

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên thuộc tính** | **Giải thích** | **Kiểu dữ liệu** | **Khóa** |
| MaPhai | Mã phái | Bit | Primary key |
| Phai | Tên phái | Nvarchar(15) |  |

* + - 1. NHANVIEN

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên thuộc tính** | **Giải thích** | **Kiểu dữ liệu** | **Khóa** |
| MaNV | Mã nhân viên | Int identity(1,1) | Primary key |
| TenNV | Tên nhân viên | Nvarchar(30) |  |
| NgaySinh | Ngày sinh | Datetime |  |
| MaPhai | Mã phái | Bit | Foreign key |
| DiaChi | Địa chỉ | Nvarchar(50) |  |
| SDT | Số điện thoại | Varchar(15) |  |
| MaVT | Mã vị trí | Int | Foreign key |
| Luong | Lương | Float |  |

* + 1. Các Ràng Buộc

**A picture containing text

Description automatically generated**

**--Tạo trigger updateTableF**

**create trigger trigger\_updateTableF on TABLEF**

**for insert, update**

**as**

**begin**

**if((select TOTAL from inserted)>=0)**

**commit tran**

**else**

**rollback tran**

**end**

**--Tạo trigger updateAccount**

**create trigger trigger\_updateAccount on ACCOUNT**

**for insert, update**

**as**

**begin**

**if((select TYPEACCOUNT from inserted) = 'ADMIN' or (select TYPEACCOUNT from inserted) = 'CASHIER' )**

**commit tran**

**else**

**rollback tran**

**end**

* + 1. Nhập dữ liệu

**Text

Description automatically generated**

**A picture containing table

Description automatically generated**

* + 1. Các fuction,proc,..

**Table

Description automatically generatedTable

Description automatically generated**

**create function ktDN (@username nvarchar(50), @pass nvarchar(50))**

**returns table**

**as**

**return (select ACC.USERNAME, ACC.PASSWORD from ACCOUNT ACC**

**where acc.USERNAME = @username and acc.PASSWORD = @pass)**

**create function loginDN (@username nvarchar(50), @pass nvarchar(50))**

**returns table**

**as**

**return (select \* from ACCOUNT ACC**

**where acc.USERNAME = @username and acc.PASSWORD = @pass)**

**--Tạo function getTable**

**create function getTable ()**

**returns table**

**as**

**return (select \* from TABLEF)**

**--Tạo function getBillDK**

**create function getBillDK (@tblName nvarchar(50))**

**returns table**

**as**

**return (select \* from BILL where NAMETABLE = @tblName)**

**create function getCategory ()**

**returns table**

**as**

**return (select \* from CATEGORY)**

**--Tạo function getFood**

**create function getFood (@categoryName nvarchar(50))**

**returns table**

**as**

**return (select \* from FOOD where NAMECATEGORY = @categoryName)**

**--Tạo procedure updatePreTable**

**create proc updatePreTable @TableName nvarchar(50)**

**as**

**update TABLEF**

**set STT = 'DATTRUOC'**

**where NAME = @TableName**

**--Tạo procedure openTable**

**create proc openTable @TableName nvarchar(50)**

**as**

**update TABLEF**

**set STT = 'ONLINE'**

**where NAME = @TableName**

**create function getThongKeFood ()**

**returns table**

**as**

**return (select \* from FOOD)**

**create function getNV ()**

**returns table**

**as**

**return (select \* from NHANVIEN)**

**create proc InsertCategory @Name nvarchar(50)**

**as**

**Insert into CATEGORY values (@Name)**

**create proc DeleteCategory @Name nvarchar(50)**

**as**

**Delete CATEGORY where NAME = @Name**

**create function SearchCategory (@Name nvarchar(50))**

**returns table**

**as**

**return (select \* from CATEGORY where NAME = @Name)**

**select \* from dbo.SearchCategory ('Kem')**

**--Tạo function getNV1**

**create function getNV1 ()**

**returns table**

**as**

**return (SELECT MANV,TENNV,NGAYSINH,MAPHAI,DIACHI,SDT,MAVT,LUONG FROM NHANVIEN)**

**--Tạo function getVT**

**create function getVT ()**

**returns table**

**as**

**return (SELECT \* FROM VITRI)**

**--Tạo function getGT**

**create function getGT ()**

**returns table**

**as**

**return (SELECT \* FROM GIOITINH)**

**--Tạo procedure InsertNV**

**create proc InsertNV @TENNV nvarchar(50), @NGAYSINH datetime, @MAPHAI bit, @DIACHI nvarchar(50), @SDT nvarchar(10),@MAVT int, @LUONG float**

**as**

**Insert into NHANVIEN values (@TENNV,@NGAYSINH,@MAPHAI,@DIACHI,@SDT,@MAVT,@LUONG)**

**--Tạo procedure DeleteNV**

**create proc DeleteNV @TENNV nvarchar(50)**

**as**

**Delete NHANVIEN where TENNV = @TENNV**

**--Tạo procedure UpdateNV**

**create proc UpdateNV @manv nchar(50), @TENNV nvarchar(50), @NGAYSINH datetime, @MAPHAI bit, @DIACHI nvarchar(50), @SDT nvarchar(10),@MAVT int, @LUONG float**

**as**

**update NHANVIEN**

**set TENNV=@TENNV,NGAYSINH = @NgaySinh , MAPHAI = @MaPhai,DIACHI = @DIACHI,SDT=@SDT,MAVT = @MAVT,LUONG = @LUONG**

**where MANV=@manv**

**--Tạo function getAccount**

**create function getAccount ()**

**returns table**

**as**

**return (select \* from ACCOUNT)**

**--Tạo procedure InsertAccount**

**create proc InsertAccount @USERNAME nvarchar(50), @DISPLAYNAME nvarchar(50), @PASSWORD nvarchar(50), @TYPEACCOUNT nvarchar(50)**

**as**

**Insert into ACCOUNT values (@USERNAME,@DISPLAYNAME,@PASSWORD,@TYPEACCOUNT)**

**--Tạo procedure DeleteAccount**

**create proc DeleteAccount @USERNAME nvarchar(50)**

**as**

**Delete ACCOUNT where USERNAME = @USERNAME**

**--Tạo procedure UpdateAccount**

**create proc UpdateAccount @USERNAME nvarchar(50), @DISPLAYNAME nvarchar(50), @PASSWORD nvarchar(50), @TYPEACCOUNT nvarchar(50)**

**as**

**update ACCOUNT**

**set DISPLAYNAME = @DISPLAYNAME , PASSWORD = @PASSWORD, TYPEACCOUNT = @TYPEACCOUNT**

**where USERNAME = @USERNAME**

**exec UpdateAccount 'user2', N'Thu Ngân 2', 'user2', 'ADMIN'**

**--Tạo function getTableF**

**create function getTableF ()**

**returns table**

**as**

**return (SELECT \* FROM TABLEF)**

**--Tạo procedure InsertTableF**

**create proc InsertTableF @NAME nvarchar(50), @STT nvarchar(50), @TOTAL float**

**as**

**Insert into TABLEF values(@NAME,@STT,@TOTAL)**

**--Tạo procedure DeleteTableF**

**create proc DeleteTableF @NAME nvarchar(50)**

**as**

**Delete TABLEF where NAME = @NAME**

**--Tạo function getBill**

**create function getBill ()**

**returns table**

**as**

**return (SELECT \* FROM BILL)**

**--Tạo procedure InsertBill**

**create proc InsertBill @NAMETABLE nvarchar(50), @NAMEFOOD nvarchar(50), @COUNTS int**

**as**

**Insert into BILL values(@NAMETABLE,@NAMEFOOD,@COUNTS)**

**--Tạo function getCountsBill**

**create function getCountsBill (@NAMETABLE nvarchar(50), @NAMEFOOD nvarchar(50))**

**returns table**

**as**

**return (SELECT COUNTS FROM BILL where NAMETABLE = @NAMETABLE and NAMEFOOD = @NAMEFOOD)**

**--Tạo procedure DeleteBill**

**create proc DeleteBill @NAMEFOOD nvarchar(50), @NAMETABLE nvarchar(50)**

**as**

**Delete BILL where NAMEFOOD = @NAMEFOOD and NAMETABLE = @NAMETABLE**

**--Tạo function getFoodPrice**

**create function getFoodPrice (@NAME nvarchar(50))**

**returns table**

**as**

**return (select PRICE from FOOD where NAME = @NAME)**

**--**

**create proc updatecount @NAMEFOOD nvarchar(50), @NAMETABLE nvarchar(50),@Cadd int**

**as**

**update BILL**

**set COUNTS = COUNTs + @Cadd**

**where NAMEFOOD = @NAMEFOOD and NAMETABLE =@NAMETABLE**

**--Tạo proc UpdateBill**

**create proc UpdateBill @NAMEFOOD nvarchar(50), @NAMETABLE nvarchar(50)**

**as**

**Update TABLEF**

**set TOTAL = TOTAL - (select \* from dbo.getFoodPrice(@NAMEFOOD))**

**where NAME = @NAMETABLE**

**--Tạo function getIDBill**

**create function getIDBill (@NAMEFOOD nvarchar(50), @NAMETABLE nvarchar(50))**

**returns table**

**as**

**return (select id from BILL where NAMEFOOD = @NAMEFOOD AND NAMETABLE = @NAMETABLE)**

**--Tạo proc UpdateBillCount**

**create proc UpdateBillCount @COUNTS int, @ID int**

**as**

**Update BILL**

**set COUNTS = COUNTS - @COUNTS**

**where ID = @ID**

**--Tạo function getTableF\_Food**

**create function getTableF\_Food (@NAMETABLE nvarchar(50))**

**returns table**

**as**

**return (select \* from TABLEF,BILL where NAMETABLE = @NAMETABLE AND TABLEF.NAME = BILL.NAMETABLE)**

**--Tạo procedure UpdateTableNull**

**create proc UpdateTableNull @NAMETABLE nvarchar(50)**

**as**

**Update TABLEF**

**set STT = N'TRONG',TOTAL=0**

**where NAME = @NAMETABLE**

**exec UpdateTableNull N'Bàn 01'**

**select \* from dbo.getTableF()**

**--Tạo procedure DeleteBillTable**

**create proc DeleteBillTable @NAMETABLE nvarchar(50)**

**as**

**Delete BILL where NAMETABLE = @NAMETABLE**

**--Tạo procedure UpdateAccount**

**create proc UpdateAccountInfo @PASSWORD nvarchar(50), @DISPLAYNAME nvarchar(50), @USERNAME nvarchar(50)**

**as**

**Update ACCOUNT**

**set PASSWORD = @PASSWORD, DISPLAYNAME=@DISPLAYNAME**

**where USERNAME=@USERNAME**

**--Tạo procedure UpdateChangeTable**

**create proc UpdateChangeTable @CURRENTTABLE nvarchar(50), @TABLECHANGE nvarchar(50)**

**as**

**Update TABLEF**

**set NAME = @TABLECHANGE**

**where NAME = @CURRENTTABLE**

**----Tạo cursor hiển thị danh sách nhân viên**

**declare dsnv cursor dynamic**

**for**

**select TENNV,NGAYSINH,MAPHAI,DIACHI,SDT,MAVT,LUONG**

**from NHANVIEN**

**open dsnv**

**declare @TENNV nvarchar(50),@NGAYSINH date, @MAPHAI nvarchar(10), @DIACHI nvarchar(50), @SDT nvarchar(10), @MAVT nvarchar(10), @LUONG float**

**fetch next from dsnv into @TENNV,@NGAYSINH, @MAPHAI, @DIACHI, @SDT, @MAVT, @LUONG**

**while(@@FETCH\_STATUS = 0)**

**begin**

**print @TENNV + ' ' + @NGAYSINH + convert(nvarchar,@NGAYSINH) + @MAPHAI + ' ' + @DIACHI + ' ' + @SDT + ' ' + @MAVT + ' ' + @LUONG + convert(nvarchar,@LUONG)**

**fetch next from dsnv into @TENNV,@NGAYSINH, @MAPHAI, @DIACHI, @SDT, @MAVT, @LUONG**

**end**

**close dsnv**

**deallocate dsnv**

**----Tạo cursor hiển thị danh sách tài khoản**

**declare dsaccount cursor dynamic**

**for**

**select USERNAME, DISPLAYNAME, PASSWORD, TYPEACCOUNT**

**from ACCOUNT**

**open dsaccount**

**declare @USERNAME nvarchar(50), @DISPLAYNAME nvarchar(50), @PASSWORD nvarchar(50), @TYPEACCOUNT nvarchar(50)**

**fetch next from dsaccount into @USERNAME, @DISPLAYNAME, @PASSWORD, @TYPEACCOUNT**

**while(@@FETCH\_STATUS = 0)**

**begin**

**print @USERNAME + ' ' + @DISPLAYNAME + ' ' + @PASSWORD + ' ' + @TYPEACCOUNT**

**fetch next from dsaccount into @USERNAME, @DISPLAYNAME, @PASSWORD, @TYPEACCOUNT**

**end**

**close dsaccount**

**deallocate dsaccount**

**----Tạo cursor hiển thị danh sách FOOD**

**declare dsfood cursor dynamic**

**for**

**select NAME, NAMECATEGORY, PRICE**

**from FOOD**

**open dsfood**

**declare @FNAME nvarchar(50), @NAMECATEGORY nvarchar(50), @PRICE float**

**fetch next from dsfood into @FNAME, @NAMECATEGORY, @PRICE**

**while(@@FETCH\_STATUS = 0)**

**begin**

**print @FNAME + ' ' + @NAMECATEGORY + ' ' + convert(nvarchar,@PRICE)**

**fetch next from dsfood into @FNAME, @NAMECATEGORY, @PRICE**

**end**

**close dsfood**

**deallocate dsfood**

**----Tạo cursor hiển thị danh sách CATEGORY**

**declare dscategory cursor dynamic**

**for**

**select NAME**

**from CATEGORY**

**open dscategory**

**declare @CATEGORYNAME nvarchar(50)**

**fetch next from dscategory into @CATEGORYNAME**

**while(@@FETCH\_STATUS = 0)**

**begin**

**print @CATEGORYNAME**

**fetch next from dscategory into @CATEGORYNAME**

**end**

**close dscategory**

**deallocate dscategory**

**---backup**

**DECLARE @NAME VARCHAR(50)**

**DECLARE @PATH VARCHAR(256) -**

**DECLARE @FILENAME VARCHAR(256)**

**DECLARE @FILEDATE VARCHAR(20)**

**SET @PATH = 'C:\Users\DanhTran\Desktop\Do\_An\_QLCaFe\QuanLiCaPhe\Backup and Restore'**

**SELECT @FILEDATE = CONVERT(VARCHAR(20),GETDATE(),112)**

**--ĐỊNH NGHĨA CURSOR**

**DECLARE DB\_CURSOR\_BACKUP CURSOR FOR**

**SELECT NAME**

**FROM MASTER.DBO.SYSDATABASES**

**OPEN DB\_CURSOR\_BACKUP**

**FETCH NEXT FROM DB\_CURSOR\_BACKUP INTO @NAME**

**WHILE @@FETCH\_STATUS = 0**

**BEGIN**

**SET @FILENAME = @PATH + @NAME + '\_' + @FILEDATE + '.BAK'**

**BACKUP DATABASE @NAME TO DISK = @FILENAME**

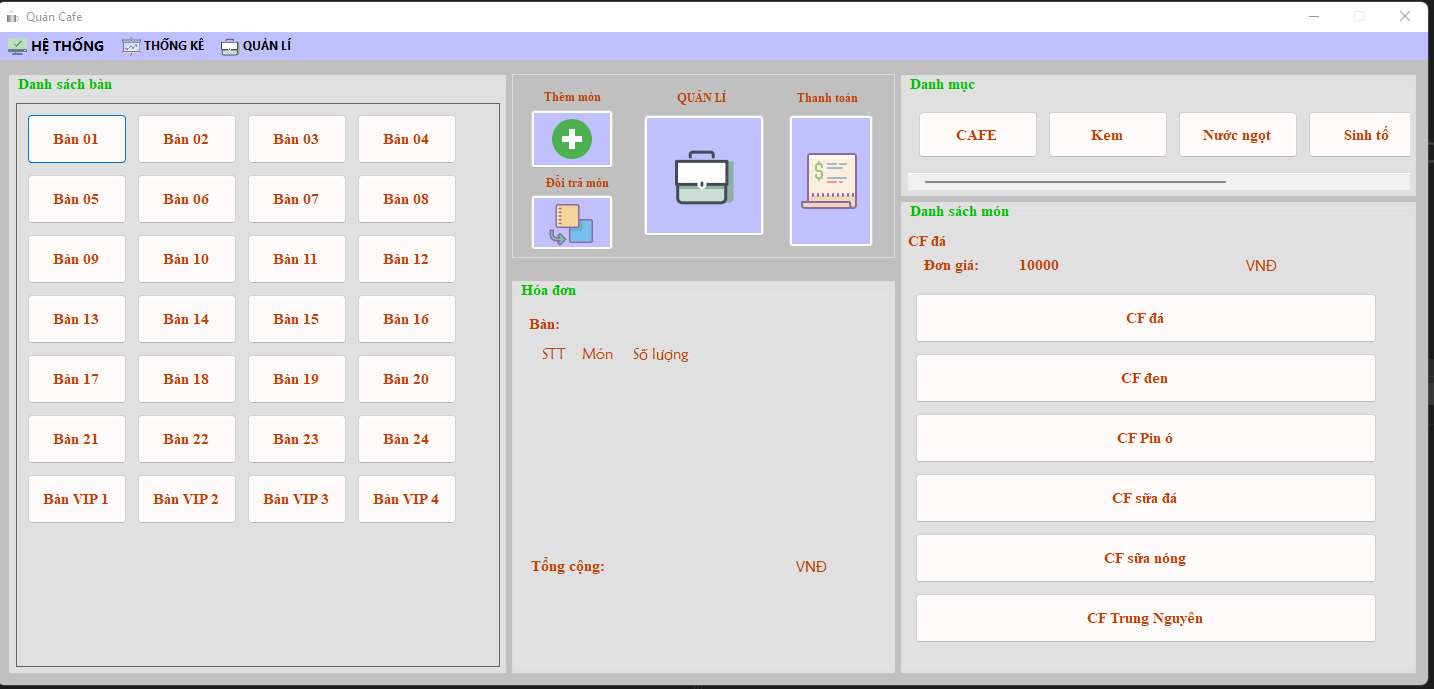
**FETCH NEXT FROM DB\_CURSOR\_BACKUP INTO @NAME**

**END**

**CLOSE DB\_CURSOR\_BACKUP**

**DEALLOCATE DB\_CURSOR\_BACKUP**

1. CÀI ĐẶT ỨNG DỤNG MINH HỌA
   * 1. Giao diện (admin)

****

* + - 1. Form đăng nhập

Đây là form chạy đầu tiên của phần mềm.

A picture containing text, window

Description automatically generated

* + - 1. Form quản lý nhân viên

Form giúp lưu trữ thông tin nhân viên và cập nhật thông tin khi có nhân viên mới vào làm.

A picture containing text, indoor, screenshot

Description automatically generated

* + - 1. Form thay đổi thông tin tài khoản

Có thể giúp người dùng thay đổi mật khẩu tài khoản để bảo mật tài khoản.

A screenshot of a computer

Description automatically generated with medium confidence

* + - 1. Form quản lý nhân viên

Form giúp lưu trữ thông tin nhân viên và cập nhật thông tin khi có nhân viên mới vào làm.

**A screenshot of a computer

Description automatically generated**

* + - 1. Form quản lý bàn

Khi mở rộng quán người dùng có thể thêm danh sách các bàn mới vào đây.

A screenshot of a computer

Description automatically generated with medium confidence

* + 1. Chức năng danh mục

Graphical user interface

Description automatically generated with medium confidence

* + 1. Chức năng thống kê

**Graphical user interface, application

Description automatically generated**

**Graphical user interface

Description automatically generated**

* + 1. Chức năng tìm kiếm

**Graphical user interface

Description automatically generated with medium confidence**

* + 1. Chức năng nghiệp vụ

**A screenshot of a computer

Description automatically generated with medium confidence**

**Graphical user interface, application

Description automatically generated**

**Graphical user interface, application, Word

Description automatically generated**

* + 1. Cài đặt chức năng phân quyền người dùng

**Table

Description automatically generated**

# 

# DANH MỤC TÀI LIỆU THAM KHẢO

* SÁCH  
  BG THUC HANH HQT CSDL 2018.pdf

BG\_HQTCSDL.pdf

* NGUỒN TỪ INTERNET VÀ WEBSITE