

TRƯỜNG CAO ĐẲNG NGHỀ THÀNH PHỐ HỒ CHÍ MINH

KHOA CÔNG NGHỆ THÔNG TIN



BÁO CÁO THỰC TẬP TỐT NGHIỆP

NGÀNH ỨNG DỤNG PHẦN MỀM

ĐỀ TÀI :Phần mềm quản lý quán coffe

SINH VIÊN : Nguyễn Quốc Phong

SINH VIÊN : Quách Minh Phúc

LỚP : T15UDPM

GVHD : Thầy Nguyễn Quốc Huy

TP. HỒ CHÍ MINH, 03/2018

**NHẬN XÉT CỦA GIÁO VIÊN HƯỚNG DẪN**

***Tp.HCM, Ngày Tháng Năm 2018***

LỜI CẢM ƠN

Em xin chân thành cảm ơn trường Cao Đẳng Nghề Hồ Chí Minh và Khoa Công Nghệ Thông Tin - Ứng dụng phần mềm đã tạo điều kiện cho em hoàn thành tốt đợt thực tập Khoá T15. Nhờ đợt thực tập này đã giúp cho em rất nhiều kinh nghiệm quý báu. Nhưng kinh nghiệm đó sẽ giúp em hoàn thiện hơn trong cộng việc và môi trường làm việc sau này.  
 Em cũng xin cám ơn quý thầy cô trong khoa Công nghệ thông tin - Ứng dụng phần mềm đã tận tình giảng dạy và trang bị cho em nhưng kiến thức cần thiết. Chính nhưng kiến thức này sẽ hỗ trợ em rất nhiều trong việc hoàn thành Khoá thực tập T15.

Mặc dù em đã cố gắng hoàn thành báo cáo thực tập Khoá T15. Tuy nhiên sẽ có những điểm thiếu sót, rất mong nhận được sự thông cảm, chia sẻ của quý thầy cô để giúp em hoàn thành tốt Khoá thực tập này.  
 Một lần nữa em xin chân thành cảm ơn !

Hồ Chí Minh, 28/1/2018

Sinh viên thực tập

Nguyễn Quốc Phong

Quách Minh Phúc

**Lời cám ơn**

**Công ty TNHH Thương mại và quảng cáo Lữ Nhạc**

Em xin chân thành cảm ơn Ban giám đốc Công ty TNHH Thương mại và quảng cáo Lữ Nhạc đã tận tình hướng dẫn em trong quá trình thực tập tại công ty.

Để có được kết quả này, em xin chân thành cám ơn Thầy Nguyễn Quốc Huy đã quan tâm hướng dẫn cho em trong quá trình hoàn thành đề tài này.

Vì thời gian có hạn cũng như kinh nghiệm còn thiếu nên đề tài của em khong tránh khỏi những thiếu sót. Em rất mong nhận được sự chỉ bảo, đóng góp ý kiến của thầy để em có điều kiện bổ sung và có thể hoàn thành tốt đề tài của mình hơn.  
 Một lần nữa em xin chân thành cảm ơn !

Hồ Chí Minh, 28/1/2018

Sinh viên thực tập

Nguyễn Quốc Phong

Quách Minh Phúc

**Mục Lục**

CHƯƠNG 1:CHUẨN BỊ DỮ LIỆU

1.QUÁ TRÌNH LÊN Ý TƯỞNG……………………………………………………

2.XÂY DỰNG CƠ SỞ DỮ LIỆU…………………………………………………..

3.THÊM DỮ LIỆU MÓN NƯỚC VÀO CƠ SỞ DỮ LIỆU………………………..

4.THIẾT KẾ CÁC FROM…………………………………………………………..

5.SỬ DỤNG FRAMEWORK ENTITY…………………………………………….

CHƯƠNG 2:LẬP TRÌNH PHẦN MỀM

1.LẬP TRÌNH FROM ĐĂNG NHẬP………………………………………………

2.LẬP TRÌNH FROM QUẢN LÝ………………………………………………….

3.LẬP TRÌNH FROM ADMIN……………………………………………………..

4.LẬP TRÌNH FROM ACCOUNT…………………………………………………

Ý tưởng

Trong thời buổi công nghệ hoá hiện đại hoá ,có một phần mềm để thay

thế sổ sách giúp chúng ta quản lý công việc,thời gian dễ dàng hơn.Trong lúc làm việc ở quán Coffe để kiếm them thu nhập em nhận thấy còn rất nhiều quán sử dụng sổ sách để ghi chép hoá đơn và thống kê hàng ngày nên em suy nghĩ nếu có một phần mềm có thể giúp ta làm những việc trên thì tốt hơn nhiều nên em quyết định viết một phần mềm là Quản lý quán Coffe.

Phần mềm này giúp ta :

-Dễ dàng quản lý hoá đơn,bàn ghé,….

-Thuận tiện trong việc thêm món,tính tiền,..

-Thuận lợi di chuyển hoá giữa các hoá đơn.

-Thống kê hoá đơn vào cuối ngày,tháng,

-Thêm sửa menu dễ dàng hơn.

**XÂY DỤNG CƠ SỞ DỮ LIỆU**

create database Cafe

go

use Cafe

go

create table Tablecoffe--Table

(

id int identity primary key,

name nvarchar(100)NOT NULL default N'chưa đặt tên',

[status] int not null default 0 -- trống || có người

)

go

create table Account--Account

(

Username nvarchar(100) primary key,

DisplayName nvarchar(100)NOT NULL default N'Nhân Viên',

[PassWord] nvarchar(1000)NOT NULL default 0,

[Type] int NOT NULL default 0--1:chủ && 0:NV

)

go

create table Category--Category

(

id int identity primary key,

name nvarchar(100)NOT NULL default N'chưa đặt tên',

)

go

create table Drinks—Drinks và những thứ khác

(

id int identity primary key,

name nvarchar(100)NOT NULL default N'Chưa đặt tên',

idcategory int NOT NULL foreign key references Category(id),

price float NOT NULL default 0

)

go

create table Bill--Bill

(

id int identity primary key,

DateCheckIn date NOT NULL default getdate(),

DateCheckOut date NOT NULL,

idTable int NOT NULL foreign key references Tablecoffe(id),

[status] int not null default 0,--thanh toán or chưa

totalprice int

)

go

create table BillInfo--BillInfo

(

id int identity primary key,

idBill int NOT NULL foreign key references Bill(id),

idDrinks int NOT NULL foreign key references Drinks(id),

CountItem int NOT NULL default 0

)

go

insert into Account values (N'admin',N'admin',N'admin',0)

insert into Account values (N'staff',N'staff',N'staff',1)

go

**THÊM DỮ LIỆU VÀO CƠ SỞ DỮ LIỆU**

------------------Thêm category

go

insert dbo.Category values ('coffe')

insert dbo.Category values ('soda')

insert dbo.Category values ('ice blended')

insert dbo.Category values ('yaourt')

insert dbo.Category values ('smoothies')

insert dbo.Category values ('milk shake')

insert dbo.Category values ('virgin')

insert dbo.Category values ('fruit tea')

insert dbo.Category values ('smoothies fruit')

insert dbo.Category values ('milk tea')

insert dbo.Category values ('juice')

insert dbo.Category values ('flower tea')

insert dbo.Category values ('mix drinks')

insert dbo.Category values ('different')

------------------Thêm nước

go

insert drinks values (N'Black Coffee',1,14)

insert drinks values (N'Milk coffee',1,18)

insert drinks values (N'Coffe sữa lắc',1,22)

insert drinks values (N'Ca cao sữa',1,25)

insert drinks values (N'Bạc xỉu 3 tầng',1,23)

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

insert drinks values (N'Blue Curacao',2,33)

insert drinks values (N'Berry',2,33)

insert drinks values (N'Passion',2,33)

insert drinks values (N'Mint',2,33)

insert drinks values (N'Lavender',2,33)

insert drinks values (N'Blue+Passion',2,33)

insert drinks values (N'Rose+Rasberry',2,33)

insert drinks values (N'Grape+Cinnamon',2,33)

insert drinks values (N'Berry+Apple',2,33)

insert drinks values (N'Berry+Coconut',2,33)

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

insert drinks values (N'Chocolate Coffee',3,40)

insert drinks values (N'Caramel Coffee',3,40)

insert drinks values (N'Green Tea Coffee',3,40)

insert drinks values (N'Cookie Chocolate',3,40)

insert drinks values (N'Cookie Berry',3,40)

insert drinks values (N'Cookie Mint',3,40)

insert drinks values (N'Apple Ice',3,40)

insert drinks values (N'Tiramisu Ice',3,40)

insert drinks values (N'Matcha Ice',3,40)

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

insert drinks values (N'Chanh dây',4,30)

insert drinks values (N'Đào',4,30)

insert drinks values (N'Xoài',4,30)

insert drinks values (N'vãi',4,30)

insert drinks values (N'Dâu',4,30)

insert drinks values (N'Phúc bồn tử',4,30)

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

insert drinks values (N'Bơ',5,35)

insert drinks values (N'Dừa',5,35)

insert drinks values (N'Mãng cầu',5,35)

insert drinks values (N'Dâu',5,35)

insert drinks values (N'Việt quất',5,35)

insert drinks values (N'Chanh tuyết',5,35)

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

insert drinks values (N'Chanh dây',6,30)

insert drinks values (N'Đào',6,30)

insert drinks values (N'Xoài',6,30)

insert drinks values (N'vãi',6,30)

insert drinks values (N'Dâu',6,30)

insert drinks values (N'Phúc bồn tử',6,30)

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

insert drinks values (N'Dừa',7,35)

insert drinks values (N'Matcha',7,35)

insert drinks values (N'Bạc hà',7,35)

insert drinks values (N'Dâu',7,35)

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

insert drinks values (N'Bơ',8,35)

insert drinks values (N'Dừa',8,35)

insert drinks values (N'Mãng cầu',8,35)

insert drinks values (N'Dâu',8,35)

insert drinks values (N'Việt quất',8,35)

insert drinks values (N'Chanh tuyết',8,35)

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

insert drinks values (N'Chanh dây',9,30)

insert drinks values (N'Đào',9,30)

insert drinks values (N'Xoài',9,30)

insert drinks values (N'vãi',9,30)

insert drinks values (N'Dâu',9,30)

insert drinks values (N'Phúc bồn tử',9,30)

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

insert drinks values (N'Matcha',10,25)

insert drinks values (N'Tiramisu',10,25)

insert drinks values (N'Đào',10,25)

insert drinks values (N'chanh dây',10,25)

insert drinks values (N'dưa lưới',10,25)

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

insert drinks values (N'Chanh dây',11,30)

insert drinks values (N'Đào',11,30)

insert drinks values (N'Xoài',11,30)

insert drinks values (N'vãi',11,30)

insert drinks values (N'Dâu',11,30)

insert drinks values (N'Phúc bồn tử',11,30)

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

insert drinks values (N'Hoa hồng',12,30)

insert drinks values (N'Hoa cúc',12,30)

insert drinks values (N'Hoa Quế',12,30)

insert drinks values (N'Hoa bách hợp',12,30)

insert drinks values (N'Hoa đào',12,30)

insert drinks values (N'Hoa Cam',12,30)

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

insert drinks values (N'Chanh dây',13,30)

insert drinks values (N'Đào',13,30)

insert drinks values (N'Xoài',13,30)

insert drinks values (N'vãi',13,30)

insert drinks values (N'Dâu',13,30)

insert drinks values (N'Phúc bồn tử',13,30)

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

insert drinks values (N'Thuốc mèo',14,2)

insert drinks values (N'Thuốc 3 số',14,2)

insert drinks values (N'Gói mèo',14,20)

insert drinks values (N'Gói 3 số',14,25)

insert drinks values (N'Khăn lạnh',14,2)

-------------------------------Thêm Bill

go

-------------------------------Thêm Bàn

declare @i int = 1

while @i <= 10

begin

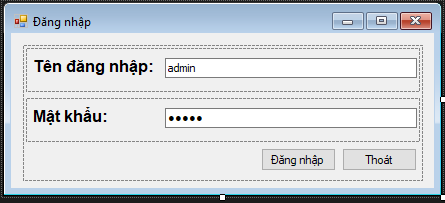
insert dbo.Tablecoffe (name) values (N'Bàn '+CAST(@i as nvarchar(100)))--cast để chuyển @i sang kiểu nvarchar

set @i=@i+1

end

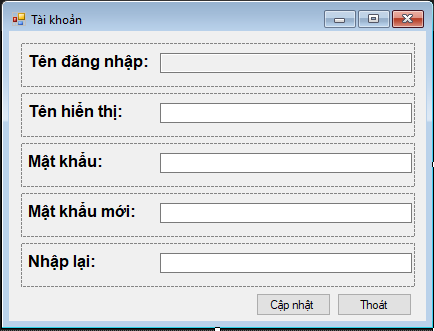
go

**THIẾT KẾ CÁC FROM**



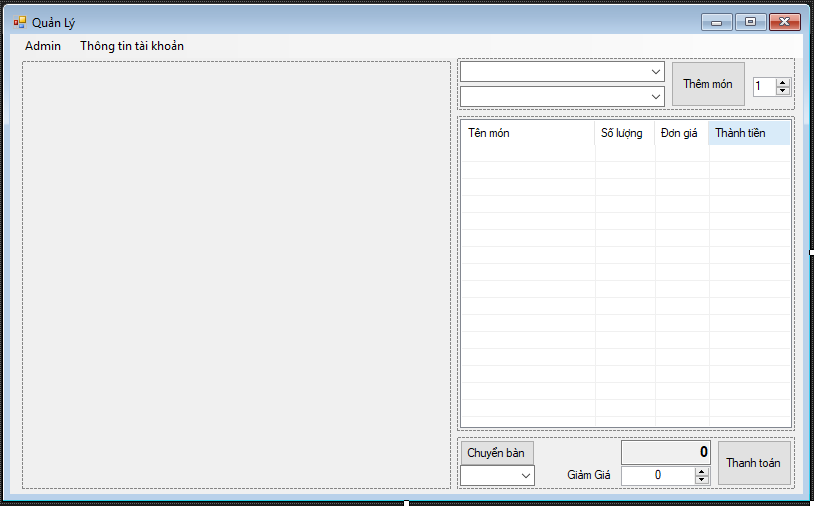
FROM đăng nhập

From này dùng để đăng nhập khi người dùng nhập đúng tên đăng nhập và mật khẩu



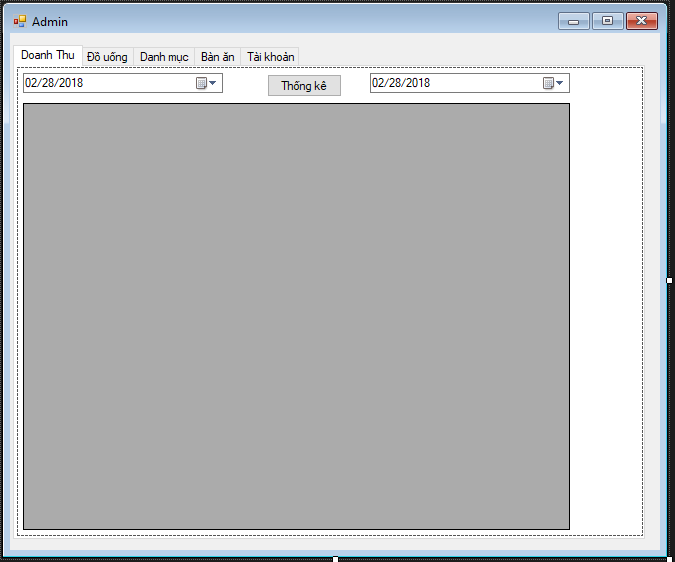
FROM tài khoản

From này dùng để thay đổi hoặc cập nhật tên hiển thị hoặc mật khẩu



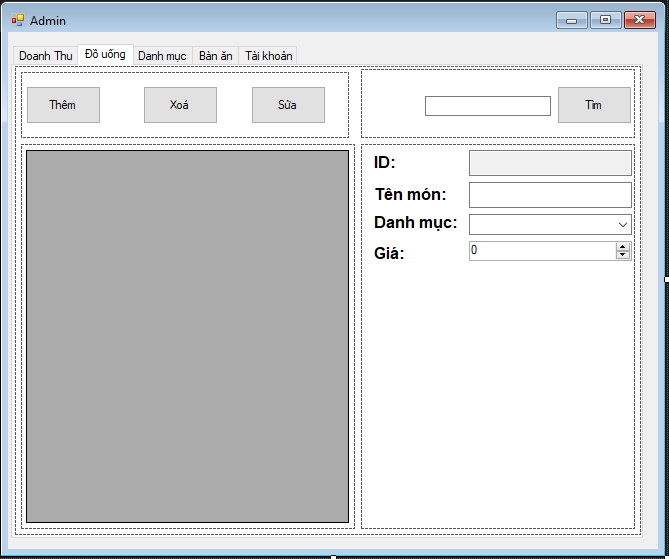
FROM Quản lý Coffe

From này chứa các bàn và quản lý chính như thêm bớt,tính tiền,chuyển bàn



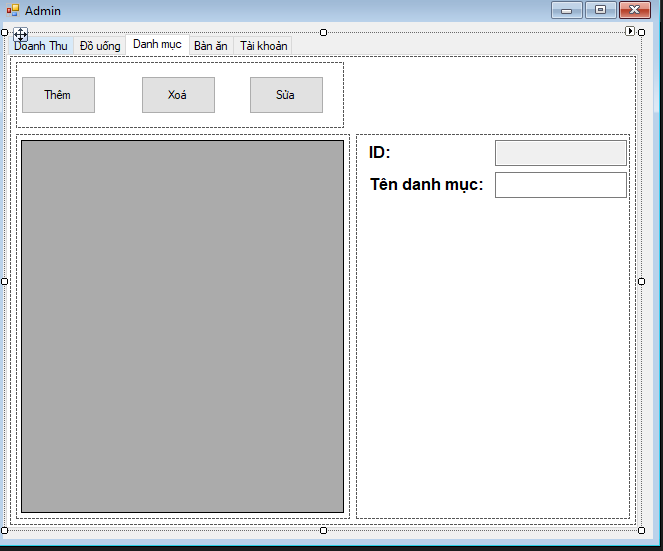
FROM ADMIN:Doanh thu

From admin chứa 5 tab chứa các tác dụng riêng.tab này dùng để thống kê hoá đơn



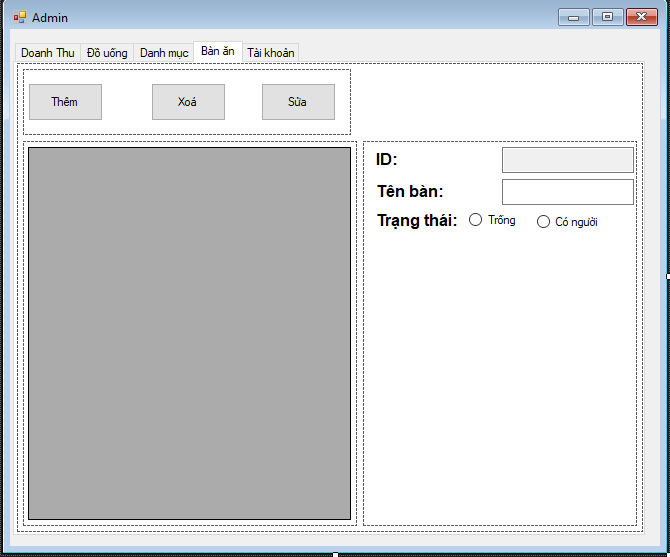
FROM ADMIN:Đồ uống

Dùng để quản lý các món đồ uống



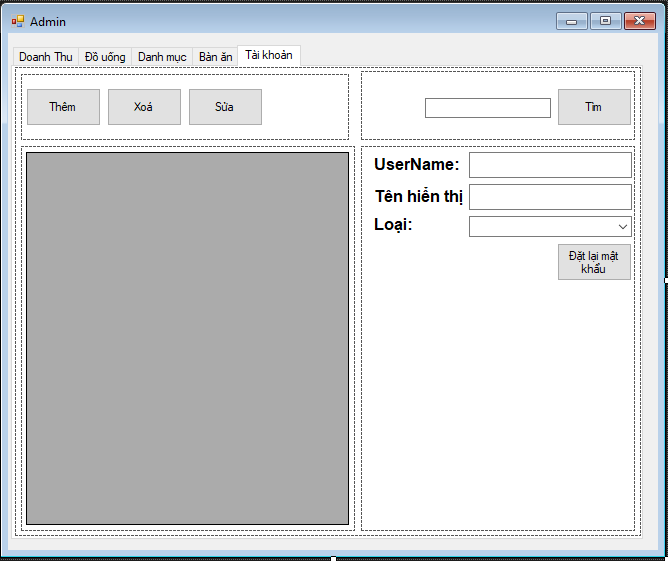
FROM ADMIN:DANH MỤC

Dùng để quản lý các danh mục của món nước



FROM ADMIN:BÀN ĂN

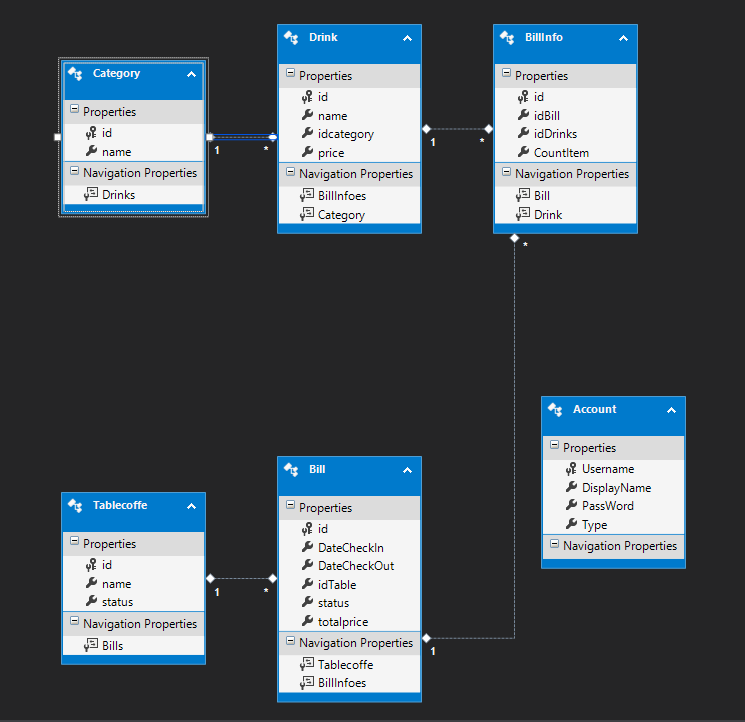
Dùng để quản lý số lượng bàn



FROM ADMIN:TÀI KHOẢN

Dùng để quản lý tài khoản của ADMIN

**ÁP DỤNG CÔNG NGHỆ ENTITY FRAMEWORK**



**LẬP TRÌNH FROM ĐĂNG NHẬP**

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

namespace QuanLyCF

{

public partial class fLogin : Form

{

public fLogin()

{

InitializeComponent();

}

CafeEntities db = new CafeEntities();

private void btnLogin\_Click(object sender, EventArgs e)

{

var check = db.Accounts.Where(c => c.Username.Equals(txbUserName.Text) && c.PassWord.Equals(txbPassWord.Text)).FirstOrDefault();

if (check != null)

{

this.Hide();

fManager f = new fManager(check.Username,check.Type);

f.ShowDialog();

this.Show();

}

else MessageBox.Show("Sai tài khoản hoặc mật khẩu !!!");

}

private void btnExit\_Click(object sender, EventArgs e)

{

if (MessageBox.Show("Bạn có muốn thoát chương trình không ?","Thông Báo",MessageBoxButtons.YesNo) == DialogResult.Yes)

{

Application.Exit();

}

}

}

}

**LẬP TRÌNH FROM QUẢN LÝ**

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Globalization;

using System.Linq;

using System.Text;

using System.Threading;

using System.Threading.Tasks;

using System.Windows.Forms;

namespace QuanLyCF

{

public partial class fManager : Form

{

string acc;

int type;

public fManager(string acc,int type)

{

this.acc = acc;

this.type = type;

InitializeComponent();

loadcategory();

loadswtichtable();

loadtable();

checkAdmin(type);

}

CafeEntities db = new CafeEntities();

#region void

void insertBill(int id)

{

Bill bill = new Bill();

bill.DateCheckIn = DateTime.Now;

bill.DateCheckOut = DateTime.Now;

bill.idTable = id;

bill.status = 1;

db.Bills.Add(bill);

db.SaveChanges();

}

int UncheckBill(int id)

{

Bill kq = db.Bills.Where(c => c.idTable == id && c.status == 1).SingleOrDefault();

try

{

int idtable = kq.id;

if (idtable != 0)

return idtable;

}

catch (Exception)

{

}

return -1;

}

void loadswtichtable()

{

cbSwitchTable.DataSource = db.Tablecoffes.ToList();

cbSwitchTable.DisplayMember = "name";

cbSwitchTable.ValueMember = "id";

}

void insertBillInfo(int idbill, int iddrinks, int count)

{

try

{

BillInfo kq = db.BillInfoes.Where(c => c.idBill == idbill && c.idDrinks == iddrinks).SingleOrDefault();

int drinkscount = kq.CountItem;

int billinfoexit = kq.id;

int newcount = 0;

if (billinfoexit > 0)//nếu đã có item đó thì tăng thêm hoặc xoá đi

{

newcount = drinkscount + count;

if (newcount > 0)

{

kq.CountItem = drinkscount + count;

db.SaveChanges();

}

else

{

db.BillInfoes.Remove(kq);

db.SaveChanges();

}

}

}

catch (Exception)

{

BillInfo billinfo = new BillInfo();

billinfo.idBill = idbill;

billinfo.idDrinks = iddrinks;

billinfo.CountItem = count;

db.BillInfoes.Add(billinfo);

db.SaveChanges();

}

}

int MaxIdBill()

{

try

{

return (int)db.Bills.Max(c => c.id);

}

catch

{

return 1;

}

}

void loadcategory()

{

cbCategory.DataSource = db.Categories.ToList();

cbCategory.DisplayMember = "name";

}

void loaddrinks(int id)

{

cbDrinks.DataSource = db.Drinks.Where(c => c.idcategory == id).ToList();

cbDrinks.DisplayMember = "name";

}

void showbill(int id)

{

float totalPrice = 0;

var result = from a in db.BillInfoes

from b in db.Bills

from c in db.Drinks

where b.idTable == id && a.idDrinks == c.id && a.idBill == b.id && b.status == 1

select new { c.name, a.CountItem, c.price };

lsvBill.Items.Clear();

foreach (var item in result)//add item vào listview

{

ListViewItem lvitem = new ListViewItem();

lvitem.Text = item.name.ToString();

lvitem.SubItems.Add(item.CountItem.ToString());

lvitem.SubItems.Add(item.price.ToString());

lvitem.SubItems.Add((item.price \* item.CountItem).ToString());

totalPrice += (float)(item.price \* item.CountItem);

lsvBill.Items.Add(lvitem);

}

CultureInfo culture = new CultureInfo("vi-VN");

//Thread.CurrentThread.CurrentCulture = culture; dùng để chuyển đổi đơn vị tiền tệ

txbTotalPrice.Text = totalPrice.ToString("c", culture);

}

void loadtable()

{

flpTable.Controls.Clear();

List<Tablecoffe> table = db.Tablecoffes.ToList();

foreach (Tablecoffe item in table)//Thêm button hiển thị bàn

{

if (UncheckBill(item.id) == -1)

item.status = 1;

else item.status = 0;

db.SaveChanges();

Button btn = new Button();

btn.Width = 100;

btn.Height = 100;

string a;

if (item.status == 0)

a = "Có người";

else a = "Không có người";

btn.Text = item.name + Environment.NewLine + a;

btn.Name = item.name;

btn.Click += Btn\_Click;

btn.Tag = item;//chứa dữ liệu của button đó

if (item.status == 0)

{

btn.BackColor = Color.Aqua;

}

else btn.BackColor = Color.Green;

flpTable.Controls.Add(btn);

}

}

void checkAdmin(int type)

{

adminToolStripMenuItem.Enabled = type == 0;

}

#endregion

#region event

private void Btn\_Click(object sender, EventArgs e)

{

int tableID = ((sender as Button).Tag as Tablecoffe).id;//lấy dữ liệu từ button để sử dụng

lsvBill.Tag = (sender as Button).Tag;

showbill(tableID);

}

private void cbCategory\_SelectedIndexChanged(object sender, EventArgs e)

{

int id = 0;

ComboBox cb = sender as ComboBox;

if (cb.SelectedItem == null)

return;

Category selected = cb.SelectedItem as Category;

id = selected.id;

loaddrinks(id);

}

private void btnCheckOut\_Click(object sender, EventArgs e)//tính tiền

{

Tablecoffe table = lsvBill.Tag as Tablecoffe;

int idbill = UncheckBill(table.id);

double totalprice = Convert.ToDouble(txbTotalPrice.Text.Split(',')[0]);

if(idbill!=-1)

{

if(MessageBox.Show(string.Format("Bạn có muốn thanh toán hoá đơn cho {0}\nTổng tiền là {1},000",table.name,totalprice),"Thông báo",MessageBoxButtons.OKCancel)==DialogResult.OK)

{

Bill bill = db.Bills.Where(c => c.id == idbill).SingleOrDefault();

bill.status = 0;

bill.DateCheckOut = DateTime.Now;

bill.totalprice = (int)totalprice;

table.status = 1;

db.SaveChanges();

showbill(table.id);

loadtable();

}

}

}

private void btnSwitchTable\_Click(object sender, EventArgs e)

{

Tablecoffe table = lsvBill.Tag as Tablecoffe;

int idbill = UncheckBill(table.id);

int tableswitch = int.Parse(cbSwitchTable.SelectedValue.ToString());

int idbillswitch = UncheckBill(tableswitch);

List <BillInfo> listbillinfo = db.BillInfoes.Where(c => c.idBill == idbill).ToList();

if (idbillswitch==-1)

{

insertBill(tableswitch);

foreach (BillInfo item in listbillinfo)

{

insertBillInfo(MaxIdBill(),item.idDrinks,item.CountItem);

}

}

else

{

foreach (BillInfo item in listbillinfo)

{

insertBillInfo(idbillswitch, item.idDrinks, item.CountItem);

}

}

Bill bill = db.Bills.Where(c => c.id == idbill).SingleOrDefault();

bill.status = 0;

db.SaveChanges();

showbill(table.id);

loadtable();

}

private void btnAddFood\_Click(object sender, EventArgs e)//thêm item

{

Tablecoffe table = lsvBill.Tag as Tablecoffe;

int idBill = UncheckBill(table.id);

int drinksID = (cbDrinks.SelectedItem as Drink).id;

int count = (int)nmDrinksCount.Value;

if (idBill == -1)//nếu bàn chưa có người thì add vào

{

insertBill(table.id);

insertBillInfo(MaxIdBill(), drinksID, count);

table.status = 1;

db.SaveChanges();

}

else//thêm item vào bàn có người

{

insertBillInfo(idBill, drinksID, count);

}

showbill(table.id);

loadtable();

}

private void adminToolStripMenuItem\_Click(object sender, EventArgs e)

{

fAdmin f = new fAdmin();

f.ShowDialog();

}

private void đăngXuấtToolStripMenuItem\_Click(object sender, EventArgs e)

{

this.Close();

}

private void thôngTinCáNhânToolStripMenuItem\_Click(object sender, EventArgs e)

{

fAccount f = new fAccount(acc);

f.ShowDialog();

}

#endregion

}

}

**LẬP TRÌNH FROM TÀI KHOẢN**

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

namespace QuanLyCF

{

public partial class fAccount : Form

{

string acc;

public fAccount(string acc)

{

this.acc = acc;

InitializeComponent();

changeAccount(acc);

}

CafeEntities db = new CafeEntities();

void changeAccount(string acc)

{

Account account = db.Accounts.Where(c => c.Username == acc).SingleOrDefault();

txbDisplayName.Text = account.DisplayName;

txbNewPass.Text = null;

txbPassWord.Text = null;

txbReEnterPass.Text = null;

txbUserName.Text = account.Username;

}

private void btnUpdate\_Click(object sender, EventArgs e)

{

Account account = db.Accounts.Where(c => c.Username == acc).SingleOrDefault();

account.DisplayName = txbDisplayName.Text;

if (txbPassWord.Text == account.PassWord)

{

if (txbReEnterPass.Text == txbNewPass.Text)

{

account.PassWord = txbNewPass.Text;

}

else

{

MessageBox.Show("sai mật khẩu nhập lại");

}

}

else MessageBox.Show("sai mật khẩu");

db.SaveChanges();

MessageBox.Show("cập nhật thành công");

}

}

}

LẬP TRÌNH FROM ADMIN

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

namespace QuanLyCF

{

public partial class fAdmin : Form

{

public fAdmin()

{

InitializeComponent();

load();

}

CafeEntities db = new CafeEntities();

private void btViewBill\_Click(object sender, EventArgs e)

{

loadBill();

}

private void btViewDrinks\_Click(object sender, EventArgs e)

{

loadDrinks();

}

void load()

{

loadBill();

loadDrinks();

loadCategory();

loadtable();

loadaccount();

}

void loadBill()

{

var result =

from b in db.Bills

from t in db.Tablecoffes

where b.DateCheckIn <= dtpFromDate.Value && b.DateCheckOut <= dtpToDate.Value && b.status == 0 && t.id == b.idTable

select new { TênBàn = t.name, TổngTiền = b.totalprice, NgàyVào = b.DateCheckIn, NgàyRa = b.DateCheckOut };

dtgBill.DataSource = result.ToList();

}

void loadDrinks()

{

var result = db.Drinks.Select(c => new { c.id, c.idcategory, c.name, c.price }).ToList();

dtgDrinks.DataSource = result;

tbDrinksID.DataBindings.Add(new Binding("text", dtgDrinks.DataSource, "id"));

tbDrinksName.DataBindings.Add(new Binding("text", dtgDrinks.DataSource, "name"));

nmdDrinksPrice.DataBindings.Add(new Binding("value", dtgDrinks.DataSource, "price"));

}

void loadCategory()

{

var result = db.Categories.Select(c=>new { c.id,c.name}).ToList();

dtgCategory.DataSource = result;

tbCategoryID.DataBindings.Add(new Binding("text", dtgCategory.DataSource, "id"));

tbCategoryName.DataBindings.Add(new Binding("text", dtgCategory.DataSource, "name"));

}

void loadtable()

{

var result = db.Tablecoffes.Select(c => new { c.id, c.name, c.status }).ToList();

dtgTable.DataSource = result;

tbTableID.DataBindings.Add(new Binding("text", dtgTable.DataSource, "id"));

tbTableName.DataBindings.Add(new Binding("text", dtgTable.DataSource, "name"));

}

void loadaccount()

{

var result = db.Accounts.Select(c=>new { c.Username,c.DisplayName,c.PassWord,c.Type}).ToList();

dtgAccount.DataSource = result;

//tbUserName.DataBindings.Add(new Binding("text", dtgTable.DataSource, "username"));

//tbDisplayName.DataBindings.Add(new Binding("text", dtgTable.DataSource, "displayname"));

}

private void tbDrinksID\_TextChanged(object sender, EventArgs e)

{

int a = Convert.ToInt32(tbDrinksID.Text);

var takecategoryid = db.Drinks.Where(c => c.id == a).SingleOrDefault();

cbDrinksCategory.DataSource = db.Categories.ToList();

cbDrinksCategory.DisplayMember = "Name";

cbDrinksCategory.SelectedItem = db.Categories.Where(c => c.id == takecategoryid.idcategory).SingleOrDefault();

}

private void btAddDrinks\_Click(object sender, EventArgs e)

{

int a = Convert.ToInt32(tbDrinksID.Text);

var takecategoryid = db.Drinks.Where(c => c.id == a).SingleOrDefault();

Drink drink = new Drink();

drink.idcategory = takecategoryid.idcategory;

drink.name = tbDrinksName.Text;

drink.price = Convert.ToDouble(nmdDrinksPrice.Value);

db.Drinks.Add(drink);

db.SaveChanges();

MessageBox.Show("thêm thành công");

}

private void btEditDrinks\_Click(object sender, EventArgs e)

{

int a = Convert.ToInt32(tbDrinksID.Text);

var takecategoryid = db.Drinks.Where(c => c.id == a).SingleOrDefault();

Drink drink = db.Drinks.Where(c => c.id == a).SingleOrDefault();

drink.idcategory = takecategoryid.idcategory;

drink.name = tbDrinksName.Text;

drink.price = Convert.ToDouble(nmdDrinksPrice.Value);

db.SaveChanges();

MessageBox.Show("Sửa thành công");

}

private void btDeleteDrinks\_Click(object sender, EventArgs e)

{

int a = Convert.ToInt32(tbDrinksID.Text);

Drink drink = db.Drinks.Where(c => c.id == a).SingleOrDefault();

db.Drinks.Remove(drink);

db.SaveChanges();

MessageBox.Show("Xoá thành công");

}

private void tbTableID\_TextChanged(object sender, EventArgs e)

{

int a = Convert.ToInt32(tbTableID.Text);

Tablecoffe table = db.Tablecoffes.Where(c => c.id == a).SingleOrDefault();

if (table.status == 1)

rdbtYes.Checked = true;

else rdbtNo.Checked = true;

}

private void tbUserName\_TextChanged(object sender, EventArgs e)

{

//var takeusername = db.Accounts.Where(c => c.Username == tbUserName.Text).SingleOrDefault();

//tbType.DataSource = db.Accounts.ToList();

//tbType.DisplayMember = "type";

//tbType.SelectedItem = db.Accounts.Where(c => c.Type == takeusername.Type).SingleOrDefault();

}

private void btCategoryAdd\_Click(object sender, EventArgs e)

{

Category category = new Category();

category.name = tbCategoryName.Text;

db.Categories.Add(category);

db.SaveChanges();

MessageBox.Show("Thêm thành công");

}

private void btCategoryDelete\_Click(object sender, EventArgs e)

{

int a = Convert.ToInt32(tbCategoryID.Text);

Category category = db.Categories.Where(c => c.id == a).SingleOrDefault();

db.Categories.Remove(category);

db.SaveChanges();

MessageBox.Show("Xoá thành công");

}

private void btCategoryEdit\_Click(object sender, EventArgs e)

{

int a = Convert.ToInt32(tbCategoryID.Text);

Category category = db.Categories.Where(c => c.id == a).SingleOrDefault();

category.name = tbCategoryName.Text;

db.SaveChanges();

MessageBox.Show("Sửa thành công");

}

private void btTableAdd\_Click(object sender, EventArgs e)

{

Tablecoffe table = new Tablecoffe();

table.name = tbTableName.Text;

if (rdbtYes.Checked)

table.status = 1;

else table.status = 0;

db.Tablecoffes.Add(table);

db.SaveChanges();

MessageBox.Show("Thêm thành công");

}

private void btTableDelete\_Click(object sender, EventArgs e)

{

int a = Convert.ToInt32(tbTableID.Text);

Tablecoffe table = db.Tablecoffes.Where(c => c.id == a).SingleOrDefault();

db.Tablecoffes.Remove(table);

db.SaveChanges();

MessageBox.Show("Xoá thành công");

}

private void btTableEdit\_Click(object sender, EventArgs e)

{

int a = Convert.ToInt32(tbTableID.Text);

Tablecoffe table = db.Tablecoffes.Where(c => c.id == a).SingleOrDefault();

table.name = tbTableName.Text;

if (rdbtYes.Checked)

table.status = 1;

else table.status = 0;

db.SaveChanges();

MessageBox.Show("sửa thành công");

}

private void btFindDrinks\_Click(object sender, EventArgs e)

{

dtgDrinks.DataSource = db.Drinks.Where(c => c.name.Contains(tbFindDrinks.Text)).ToList();

//tbDrinksID.DataBindings.Add(new Binding("text", dtgDrinks.DataSource, "id"));

//tbDrinksName.DataBindings.Add(new Binding("text", dtgDrinks.DataSource, "name"));

//nmdDrinksPrice.DataBindings.Add(new Binding("value", dtgDrinks.DataSource, "price"));

}

}

}

# **KẾT LUẬN**

ƯU ĐIỂM:Giúp quản lý quán coffe tốt hơn,sử dụng công nghệ mới entity framework

KHUYẾT ĐIỂM:Khó bảo trì và sửa chữa khi hư hoặc khi làm việc nhóm

Sau một thời gian tìm hiểu và học hỏi trên mạng cũng như từ các anh chị của công ty mà chúng em thực tập. Và một số tài liệu về việc sử dụng Entity Framework của trang web stackoverflow.com,từ đó nhóm chúng em đã hoàn thiện được phần mềm”QUẢN LÝ COFFE” cho công ty TNHH thương mại và quảng cáo Lữ Nhạc với kết quả là phần mềm trên.

Trong những tháng đi thực tập tại Công ty TNHH thương mại và quảng cáo Lữ nhạc đã giúp em tiếp thu thêm rất nhiều kiến thức có ích cho công việc ems au này, học hỏi được những cái hay mới lạ và bổ ích và đặc biệt là chính tay bọn em làm ra một phần mềm .Đó là niềm vui, niềm tự hào của chúng em khi được trao đổi, học hỏi kinh nghiệm học tập làm việc và thực tập tại công ty TNHH thương mại và quảng cáo Lữ Nhạc.  
Xin chân thành cảm ơn!!!