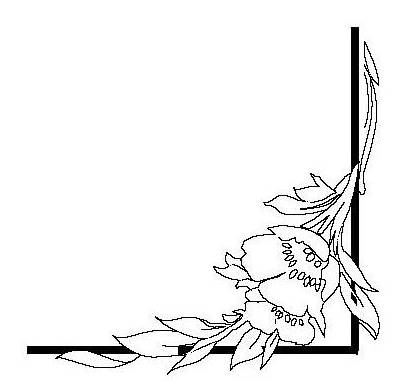
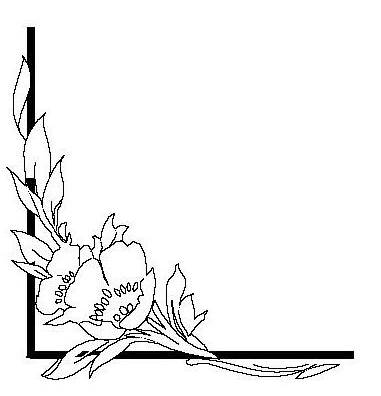
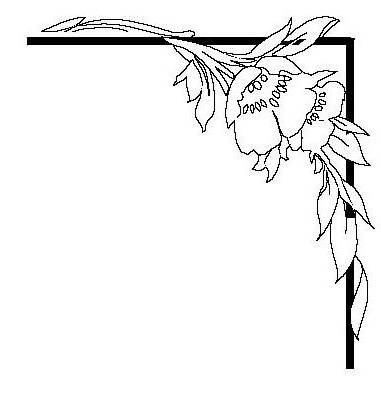
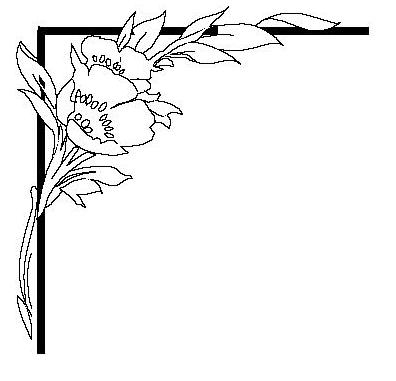
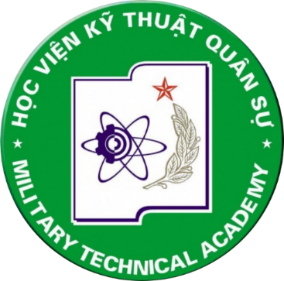
**HỌC VIỆN KỸ THUẬT QUÂN SỰ**



-----🙞🙜🕮🙞🙜-----



**BÁO CÁO BÀI TẬP LỚN MÔN HỌC C#**

Đề tài:

**xây dựng phần mềm Quản lý**

**Thu Ngân Phòng trà**

**Giảng Viên : Hồ Nhật Quang**

**Học viên thực hiện : Nguyễn Gia Hiếu**

**Lớp : Bảo đảm an toàn thông tin**

Hà Nội, Tháng 03/2023

**MỤC LỤC**

[**A.** **TÀI LIỆU ĐẶC TẢ YÊU CẦU PHẦN MỀM** 1](#_Toc129179293)

[**I.** **Giới thiệu** 1](#_Toc129179294)

[**1.** **Mục tiêu** 1](#_Toc129179295)

[**2.** **Phạm vi** 1](#_Toc129179296)

[**3.** **Tham khảo** 1](#_Toc129179297)

[**4.** **Mô tả tài liệu** 1](#_Toc129179298)

[**II.** **Tổng quan về phần mềm** 1](#_Toc129179299)

[**1.** **Yêu cầu chung về phần mềm** 1](#_Toc129179300)

[**2.** **Chức năng của phần mềm** 2](#_Toc129179301)

[**3.** **Đối tượng người dùng** 2](#_Toc129179302)

[**4.** **Các ràng buộc** 3](#_Toc129179303)

[**5.** **Giả định và phụ thuộc** 4](#_Toc129179304)

[**III.** **Quy trình nghiệp vụ** 4](#_Toc129179305)

[**IV.** **Đặc tả yêu cầu phần mềm** 8](#_Toc129179306)

1. **TÀI LIỆU ĐẶC TẢ YÊU CẦU PHẦN MỀM**
2. **Giới thiệu**
3. **Mục tiêu**

Chương trình quản lý hóa đơn quán ăn là một chương trình xoay quanh các đối tượng cần quản lý là món ăn , bàn ăn, hóa đơn, giúp nhân viên thu ngân, người quản lý dễ dành quản lý các bàn ăn, hóa đơn và thống kê doanh thu.

Mục tiêu của đề tài là xây dựng một phần mềm để thực hiện các thao tác trên, trợ giúp việc quản lý và nhân viên thu ngân tại quán ăn được nhanh chóng, dễ dàng hơn, thuận tiện hơn cũng như đáp ứng được nhu cầu ngày càng cao của khách hàng giúp giảm thiểu thời gian đặt chỗ và thanh toán..

1. **Phạm vi**

Phần mềm hỗ trợ khoản thu ngân được cài đặt trên hệ thống máy tính tại quầy thu ngân, giúp nhân viên thu ngân thanh toán hóa đơn của khách hàng nhanh chóng, chính xác và dễ dàng thực hiện việc in hóa đơn cho khách hàng. Đồng thời phục vụ cho người quản lý thống kê, kiểm soát doanh thu của quán.

1. **Tham khảo**

HowKTeam.com

Youtube.com

1. **Mô tả tài liệu**

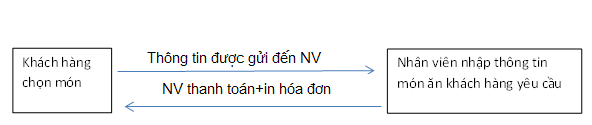
|  |  |  |
| --- | --- | --- |
| **Phần** | **Mô tả** | **Mục đích** |
| 1. Giới thiệu | Mô tả một cách khái quát nhất, phạm vi, mục đích tài liệu cần đạt được. | Đưa ra các công việc cần thực hiện của tài liệu. |
| 1. Tổng quan về phần mềm | Khái quát phần mềm, đưa ra mô hình của phần mềm. | Khái quát phần mềm, đối tượng sử dụng, mô hình ngữ cảnh. |
| 1. Phân tích quy trình nghiệp vụ | Các chức năng của hệ thống. | Phân tích quy trình nghiệp vụ một cách cụ thể nhất. |
| 1. Đặc tả yêu cầu | Các yêu cầu về chức năng và hiệu năng của phần mềm. | Trình bày các yêu cầu mà phần mềm cần phải thực thi. |

1. **Tổng quan về phần mềm**
2. **Yêu cầu chung về phần mềm**

Hệ thống phần mềm bao gồm 2 phần: Quản lý việc thanh toán hóa đơn và Quản lý thông tin của quán ăn trong CSDL.

1. *Quản lý việc thanh toán hóa đơn*

* Nhập thông tin món khách hàng yêu cầu (tên món, số lượng).
* Thanh toán, giảm giá (nếu có) và in hóa đơn cho khách hàng.



1. *Quản lý thông tin trong Cơ sở dữ liệu*

* Lưu trữ danh sách các thức uống 🡪 đối chiếu với thức uống khách hàng yêu cầu để thanh toán hóa đơn chi tiết và chính xác.
* Lưu trữ các hóa đơn đã được thanh toán theo thời gian 🡪 phục vụ cho việc báo cáo thông tin sau này.

1. **Chức năng của phần mềm**
2. *Thanh toán hóa đơn cho khách hàng*

* Khi khách hàng yêu cầu món trong menu của quán. Nhân viên tiếp nhận yêu cầu và thêm món khách vừa yêu cầu vào hóa đơn của bàn đó.
* Nếu khách hàng có nhu cầu chuyển bàn, nhân viên sẽ thực hiện chuyển hóa đơn của bàn hiện tại sang 1 bàn mới không có người, hoặc hoán đổi hóa đơn của 2 bàn đã có người.
* Nếu khách đặt trước bàn thì bàn ăn sẽ chuyển thành status là đã đặt trước.
* Nếu khách đi với nhóm quá đông người và tình cờ gặp nhau ở quán có thể ghép hai bàn các bàn ăn lại với nhau và hệ thống sẽ thông báo là những bàn khác đã được gộp và sẽ không thể xếp người vào vị trí đó nữa.
* Nếu khách muốn trả tiền cho người khác thì sẽ gộp bill lại với nhau.
* Nếu nhân viên nhầm lẫn trong việc gọi món của khách thì sẽ có chức năng tách món ấy ra khỏi bill và trả về đúng bàn khác.
* Vào các ngày lễ hoặc các dịp đặc biệt do quán quy định, nhân viên có thể giảm giá cho hóa đơn.
* Khi khách hàng có yêu cầu thanh toán 🡪 nhân viên sẽ lập hóa đơn. Hóa đơn này được lập thành 2 bản, 1 bản được in ra cho khách hàng, 1 bản được lưu lại. Khách hàng nhận hóa đơn và thanh toán tiền cho nhân viên.

1. *Quản lý việc báo cáo doanh thu*

* Phần mềm cho phép xem danh sách hóa đơn theo thời gian.
* Xuất báo cáo tài chính ra word
* Có biểu đồ cột doanh thu theo từng bàn để người quản lý dễ nhìn và trực quan hơn

1. *Chức năng khác*

* Người dùng có thể đổi mật khẩu và thông tin hiển thị của tài khoản.

1. **Đối tượng người dùng**

Phần mềm được ứng dụng cho 2 đối tượng: NV thu ngân và người quản lý:

* Khách hàng không trực tiếp sử dụng phần mềm, nhưng là người đưa ra các món cho NV thu ngân.
* NV thu ngân:

+ Tiếp nhận danh sách món khách hàng yêu cầu và thanh toán hóa đơn cho khách hàng. Chuyển bàn cho khách hàng khi có yêu cầu.

* Người quản lý: có thể sử dụng toàn bộ các chức năng của nhân viên, ngoài ra người quản lý còn có các quyền sau:

+ Thêm, sửa, xóa danh mục món và từng món. Định giá cho món.

+ Thêm, sửa, xóa bàn trong quán.

+ Kiểm soát doanh thu

+ Phân quyền người sử dụng phần mềm.

+ Quản lý tài khoản của nhân viên.

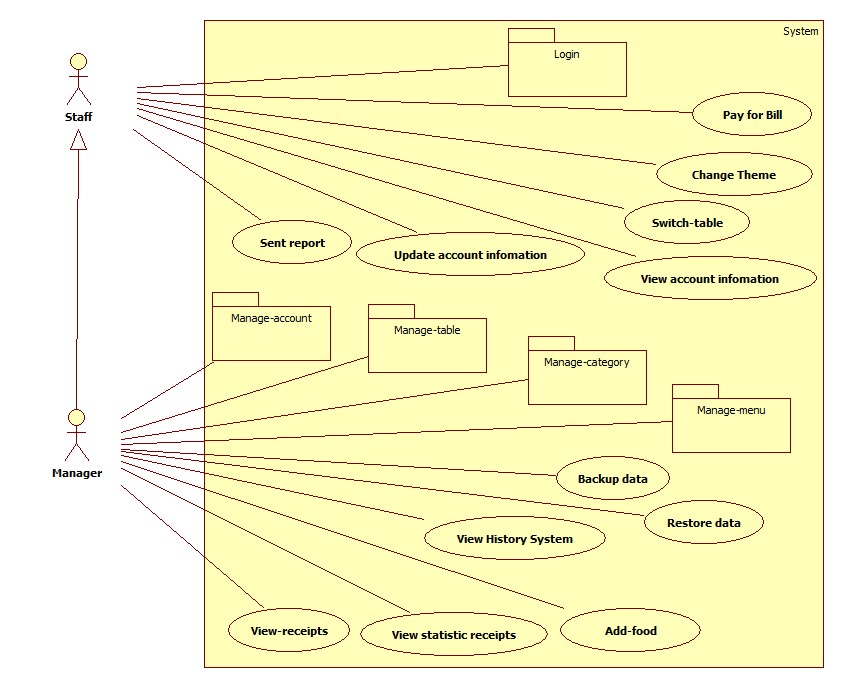
1. **Các ràng buộc**

* Hệ thống ràng buộc bởi yêu cầu của khách hàng: Khi khách hàng có yêu cầu thanh toán hóa đơn🡪 hệ thống mới cho phép thanh toán theo yêu cầu của khách hàng.
* Ràng buộc về dữ liệu đầu vào so với dữ liệu trong CSDL: món khách hàng yêu cầu được NV thu ngân chọn danh mục món sau đó chọn món 🡪 tất cả thông tin về món được hiển thị lên giao diện.
* Ràng buộc với ứng dụng bên ngoài: để gửi thông tin chi tiết của hóa đơn đến cho khách hàng🡪 yêu cầu phải có máy in kết nối với hệ thống máy tính hiện tại🡪 in hóa đơn cho khách hàng.

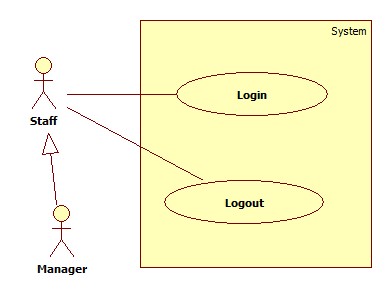
1. **Giả định và phụ thuộc**

Phần mềm chỉ được cài đặt trên hệ thống máy tính của quán cafe tại quầy thu ngân, khi đó hệ thống phần mềm mới kết nối được CSDL của quán. Hoặc nếu không cài đặt trên hệ thống máy tính của quán mà cài đặt trên máy tính khác 🡪 buộc phải có CSDL của quán thì mới thực hiện đầy đủ được các thao tác nghiệp vụ.

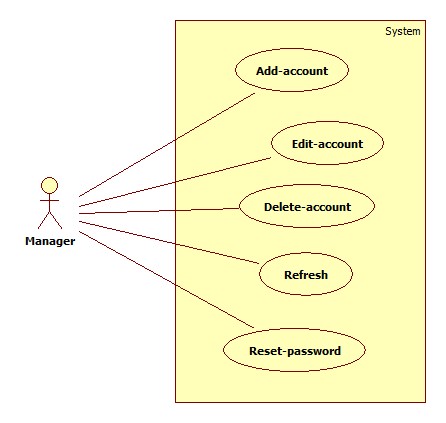
1. **Quy trình nghiệp vụ**

****

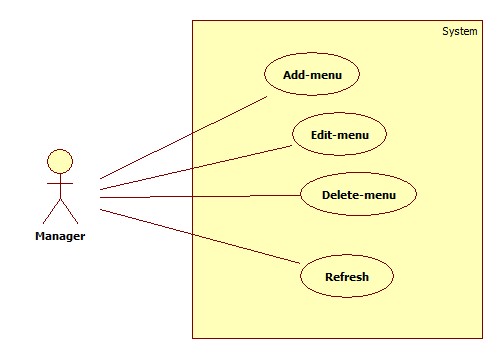
*Hình 4:* Sơ đồ use case tổng thể hệ thống



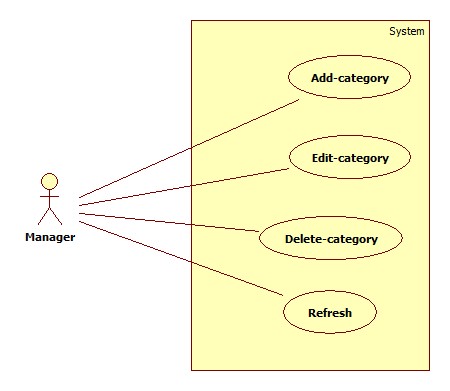
*Hình 5:* Sơ đồ phân rả package login

**

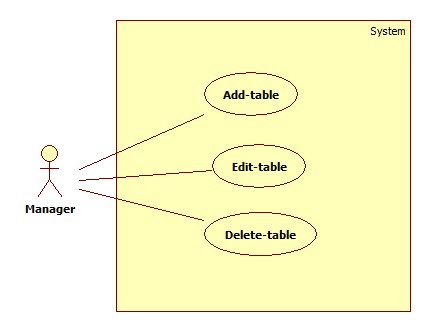
*Hình 6:* Sơ đồ phân rả package Manage-account



*Hình7:* Sơ đồ phân rả package Manage-menu



*Hình 8* Sơ đồ phân rả package Manage-category



*Hình 9:* Sơ đồ phân rả package Manage-table

1. **Đặc tả yêu cầu phần mềm**
   1. **Bảng dữ liệu**

**Tài Khoản**

|  |
| --- |
| ***Account*** |
| *Column name* | *Data Type* | *Not Null* | *Index* | *Chú thích* |
| Username | nvarchar(100) | 🗸 | Primary key | Tên đăng nhập |
| DisplayName | nvarchar(100) | 🗸 |  | Tên hiển thị |
| PassWord | nvarchar(100) | 🗸 |  | Mật khẩu |
| Type | int | 🗸 |  | Admin hoặc staff |

**Danh Mục**

|  |
| --- |
| ***FoodCategory*** |
| *Column name* | *Data Type* | *Not Null* | *Index* | *Chú thích* |
| id | int | 🗸 | Primary key | ID |
| name | nvarchar(100) | 🗸 |  | Tên Danh mục Đồ Ăn |

**Đồ Ăn**

|  |
| --- |
| ***Food*** |
| *Column name* | *Data Type* | *Not Null* | *Index* | *Chú thích* |
| ID | int | 🗸 | Primary key | ID |
| name | nvarchar(100) | 🗸 |  | Tên Đồ Ăn |
| idCategory | int | 🗸 | Foreign key | ID Danh Muc |
| price | float | 🗸 |  | Giá Đồ Ăn |

**Thông Tin Hoá Đơn**

|  |
| --- |
| ***BillInfo*** |
| *Column name* | *Data Type* | *Not Null* | *Index* | *Chú thích* |
| id | int | 🗸 | Primary key | ID |
| idCategory | int | 🗸 | Foreign key | ID Hoá Đơn |
| idFood | int | 🗸 | Foreign key | ID Đồ Ăn |
| count | int | 🗸 |  | Số lượng |

**Hoá Đơn**

|  |
| --- |
| ***Bill*** |
| *Column name* | *Data Type* | *Not Null* | *Index* | *Chú thích* |
| ID | int | 🗸 | Primary key | ID |
| DateCheckIn | date | 🗸 |  | Ngày Vào |
| DateCheckOut | date |  |  | Ngày Ra |
| idTable | int | 🗸 | Foreign key | ID bàn |
| status | int | 🗸 |  | Trạng Thái |
| discount | int |  |  | Giảm giá |
| totalPrice | float |  |  | Tổng Tiền ăn của đơn giá |
| timeCheckIn | time | 🗸 |  | Giờ khách vào |
| timeCheckOut | time |  |  | Giờ khách ra |

**Bàn Ăn**

|  |
| --- |
| ***TableFood*** |
| *Column name* | *Data Type* | *Not Null* | *Index* | *Chú thích* |
| ID | int | 🗸 | Primary key | ID |
| name | nvarchar(100) | 🗸 |  | Tên Bàn Ăn |
| status | nvarchar(100) | 🗸 |  | Trạng Thái Bàn |

* 1. **Chương Trình**
     + 1. **Các procudure**

--================================== sửa món ăn ======================================

ALTER PROC [dbo].[USP\_UpdateFood]

@name NVARCHAR(100), @idCategory NVARCHAR(100), @price FLOAT,@id INT

AS

BEGIN

UPDATE dbo.Food SET name=@name, idCategory=@idCategory, price=@price WHERE id = @id

END

--

GO

--===========================================sửa account====================================

ALTER PROC [dbo].[USP\_UpdateAccount]

@userName NVARCHAR(100), @displayName NVARCHAR(100), @password NVARCHAR(100), @newPassword NVARCHAR(100)

AS

BEGIN

DECLARE @isRightPass INT = 0

SELECT @isRightPass = COUNT(\*) FROM dbo.Account WHERE USERName = @userName AND PassWord = @password

IF (@isRightPass = 1)

BEGIN

IF (@newPassword = NULL OR @newPassword = '')

BEGIN

UPDATE dbo.Account SET DisplayName = @displayName WHERE UserName = @userName

END

ELSE

UPDATE dbo.Account SET DisplayName = @displayName, PassWord = @newPassword WHERE UserName = @userName

end

END

--==================================đổi bàn ====================================

GO

ALTER PROC [dbo].[USP\_SwichTable]

@idtable1 INT, @idTable2 INT

AS BEGIN

DECLARE @idFirsrBill int

DECLARE @idSeconrdBill INT

DECLARE @status1 NVARCHAR(100)

DECLARE @status2 NVARCHAR(100)

SELECT @status1=status FROM dbo.TableFood WHERE id = @idtable1

SELECT @status2=status FROM dbo.TableFood WHERE id = @idtable2

SELECT @idFirsrBill = id FROM dbo.Bill WHERE idTable= @idtable1 AND status=0

SELECT @idSeconrdBill = id FROM dbo.Bill WHERE idTable= @idTable2 AND status=0

IF(@idFirsrBill IS NULL)

BEGIN

INSERT INTO dbo.Bill(DateCheckIn,DateCheckOut,idTable,status,discount)

VALUES(GETDATE(),NULL,@idtable1,0,0 )

SELECT @idFirsrBill = MAX(id) FROM dbo.Bill WHERE idTable = @idtable1 AND status=0

END

IF(@idSeconrdBill IS NULL)

BEGIN

INSERT INTO dbo.Bill(DateCheckIn,DateCheckOut,idTable,status,discount)

VALUES

(GETDATE(), NULL, @idtable2, 0, 0 )

SELECT @idSeconrdBill = MAX(id) FROM dbo.Bill WHERE idTable = @idtable2 AND status=0

END

SELECT id INTO IDBillInfoTable FROM dbo.BillInfo WHERE idBill = @idSeconrdBill

UPDATE dbo.BillInfo SET idBill= @idSeconrdBill WHERE idBill= @idFirsrBill

UPDATE dbo.BillInfo SET idBill=@idFirsrBill WHERE id IN (SELECT \* FROM dbo.IDBillInfoTable)

DROP TABLE dbo.IDBillInfoTable

IF (@status1=N'Có người')

BEGIN

IF (@status2=N'Trống')

BEGIN

UPDATE dbo.TableFood SET status = N'Trống' WHERE id=@idtable1

UPDATE dbo.TableFood SET status = N'Có người' WHERE id=@idtable2

DELETE dbo.Bill WHERE id=@idFirsrBill

END

END

END

Go

--=====================================tao data cho table hóa đơn==========================

ALTER PROCEDURE [dbo].[USP\_Report]

@idBill INT

AS

BEGIN

DELETE dbo.report

INSERT INTO dbo.report (tenMon, donGia, SL, thanhTien)

SELECT f.name AS N'tenMon', CONCAT(FORMAT(f.price, 'N0', 'vi-VN'), ',00') AS N'donGia', bi.count AS N'SL', CONCAT(FORMAT(f.price \* bi.count, 'N0', 'vi-VN'), ',00') AS N'tTien'

FROM dbo.Food AS f, dbo.BillInfo AS bi

WHERE bi.idBill = @idBill AND f.id = bi.idFood;

END

GO

--======================================gộp bill=======================================

ALTER PROC [dbo].[USP\_PoolTable]

@idTable1 INT, @idTable2 INT

AS BEGIN

DECLARE @idFirsrBill INT-- = 1023

DECLARE @idSeconrdBill INT --=1022

PRINT @idFirsrBill

SELECT @idFirsrBill = dbo.Bill.id FROM dbo.Bill WHERE idTable = @idTable1 AND status=0

SELECT @idSeconrdBill = dbo.Bill.id FROM dbo.Bill WHERE idTable = @idTable2 AND status=0

PRINT @idFirsrBill

PRINT @idSeconrdBill

IF(@idSeconrdBill IS NULL)

BEGIN

INSERT INTO dbo.Bill

(

DateCheckIn,

DateCheckOut,

idTable,

status,

discount

)

VALUES

( GETDATE(), -- DateCheckIn - date

NULL, -- DateCheckOut - date

@idTable2,-- idTable - int

0, -- status - int

0 -- discount - int

)

SELECT @idSeconrdBill= MAX(id) FROM dbo.Bill

END

IF(@idFirsrBill IS NULL)

BEGIN

INSERT INTO dbo.Bill

(

DateCheckIn,

DateCheckOut,

idTable,

status,

discount

)

VALUES

( GETDATE(), -- DateCheckIn - date

NULL, -- DateCheckOut - date

@idTable1,-- idTable - int

0, -- status - int

0 -- discount - int

)

SELECT @idFirsrBill= MAX(id) FROM dbo.Bill

END

PRINT @idFirsrBill

PRINT @idSeconrdBill

-- BẮT ĐẦU XỬ LÝ

SELECT idBill,idFood,count INTO tem000 FROM dbo.BillInfo WHERE idBill= @idFirsrBill

UPDATE dbo.tem000 SET idBill=@idSeconrdBill

SELECT \* FROM dbo.tem000

DELETE dbo.BillInfo WHERE idBill = @idFirsrBill

UPDATE dbo.TableFood SET status= N'Trống' WHERE id= @idTable1

UPDATE dbo.Bill SET status =1 WHERE id = @idFirsrBill

PRINT @idFirsrBill

PRINT @idSeconrdBill

--KỊCH HAY BẮT ĐẦU

DECLARE tempCursor CURSOR FOR SELECT \* FROM dbo.tem000

PRINT @idFirsrBill

PRINT @idSeconrdBill

OPEN tempCursor

DECLARE @idBill INT

DECLARE @idFood INT

DECLARE @count INT

FETCH NEXT FROM tempCursor INTO @idBill, @idFood, @count

WHILE @@FETCH\_STATUS = 0

BEGIN

EXEC dbo.UPR\_UpdateBillInfo @idBill =@idBill ,@idFood = @idFood, @count = @count

FETCH NEXT FROM tempCursor INTO @idBill, @idFood, @count

END

CLOSE tempCursor

DEALLOCATE tempCursor

DROP TABLE dbo.tem000

END

--================Thanh toán================================================================

GO

ALTER PROC [dbo].[USP\_Payment]

@idBill INT, @discount INT,@totalPrice FLOAT , @tableName NVARCHAR(100)

AS

BEGIN

SELECT id INTO Temp FROM tableFood WHERE status=CONCAT(N'Đang được gộp với ',@tableName)

UPDATE dbo.Bill SET TimeCheckOut = CONVERT(varchar(5), GETDATE(), 108) ,dateCheckOut = GETDATE(), status = 1, discount = @discount, totalPrice = @totalPrice WHERE id = @idBill

DECLARE tempCursor CURSOR FOR SELECT \* FROM dbo.Temp

--con trỏ để sửa những bàn được gộp quay về trạng thái ban đầu

OPEN tempCursor

DECLARE @idTable INT

FETCH NEXT FROM tempCursor INTO @idTable

WHILE @@FETCH\_STATUS = 0

BEGIN

DELETE bill WHERE idTable= @idTable AND status =0

UPDATE tableFood SET status = N'Trống'WHERE id=@idTable

FETCH NEXT FROM tempCursor INTO @idTable

END

CLOSE tempCursor

DEALLOCATE tempCursor

DROP TABLE Temp

END

GO

--===================================đăng nhập=======================================

ALTER PROC [dbo].[USP\_Login]

@userName nvarchar(100), @passWord nvarchar(100)

AS

BEGIN

SELECT \* FROM dbo.Account WHERE UserName = @userName AND PassWord = @passWord

END

GO

--===================================thêm món ăn mới==================================

ALTER PROC [dbo].[USP\_InsertFood]

@name NVARCHAR(100), @idCategory NVARCHAR(100), @price FLOAT

AS

BEGIN

INSERT INTO dbo.Food

(

name,

idCategory,

price

)

VALUES

( @name, -- name - nvarchar(100)

@idCategory, -- idCategory - int

@price -- price - float

)

END

GO

--===================================thêm món mới===================================

ALTER PROC [dbo].[USP\_InsertBillInfo]

@idBill INT, @idFood INT, @count INT

AS

BEGIN

DECLARE @isExitsBillInfo INT

DECLARE @foodCount INT = 1

SELECT @isExitsBillInfo = id, @foodCount = b.count

FROM dbo.BillInfo AS b

WHERE idBill = @idBill AND idFood = @idFood

IF (@isExitsBillInfo > 0)

BEGIN

DECLARE @newCount INT = @foodCount + @count

IF (@newCount > 0)

UPDATE dbo.BillInfo SET count = @foodCount + @count WHERE idFood = @idFood AND idBill = @idBill

ELSE

DELETE dbo.BillInfo WHERE idBill = @idBill AND idFood = @idFood

END

ELSE

BEGIN

INSERT dbo.BillInfo

( idBill, idFood, count )

VALUES ( @idBill, -- idBill - int

@idFood, -- idFood - int

@count -- count - int

)

END

END

GO

--===================================thêm bill mới===================================

ALTER PROC [dbo].[USP\_InsertBill]

@idTable INT

AS

BEGIN

INSERT dbo.Bill

( DateCheckIn ,

DateCheckOut ,

idTable ,

status,

discount

)

VALUES ( GETDATE() , -- DateCheckIn - date

NULL , -- DateCheckOut - date

@idTable , -- idTable - int

0 , -- status - int

0

)

END

GO

--===================================lấy toàn bộ bàn ===================================

ALTER PROC [dbo].[USP\_GetTableList]

AS SELECT \* FROM dbo.TableFood

G

--======================lấy dữ liệu cho data set============================

ALTER PROC [dbo].[USP\_GetResultReport]

@checkIn date, @checkOut date

AS

BEGIN

SELECT SUM(totalPrice) AS totalFinal,SUM(totalPrice/(1-discount\*0.01)) AS total,SUM(totalPrice/(1-discount\*0.01))-SUM(totalPrice) AS totalDiscount ,@checkIn AS dateCheckIn,@checkOut AS dateCheckOut

FROM dbo.Bill

WHERE status=1 AND DateCheckIn >= @checkIn AND DateCheckOut<=@checkOut

END

GO

--===================================lấy hóa bill cũ===================================

ALTER PROC [dbo].[USP\_GetNumBillByDate]

@checkIn date, @checkOut date

AS

BEGIN

SELECT COUNT(\*)

FROM dbo.Bill AS b,dbo.TableFood AS t

WHERE DateCheckIn >= @checkIn AND DateCheckOut <= @checkOut AND b.status = 1

AND t.id = b.idTable

END

GO

--=================================== lấy bill===================================

ALTER PROC [dbo].[USP\_GetListFoodByBill]

@idBill INT

AS

BEGIN

SELECT f.name AS N'Tên món',CONCAT(FORMAT(f.price, 'N0', 'vi-VN'), ',00') AS N'Đơn giá', bi.count AS N'SL',CONCAT(FORMAT(f.price\*bi.count, 'N0', 'vi-VN'),',00') AS N'T.Tiền'

FROM dbo.Food AS f,dbo.BillInfo AS bi

WHERE bi.idBill= @idBill AND f.id= bi.idFood

END

GO

--============================= lấy danh sách của page===============================

ALTER PROC [dbo].[USP\_GetListBillByDateAndPage]

@checkIn date, @checkOut date, @page int

AS

BEGIN

DECLARE @pageRows INT = 10

DECLARE @selectRows INT = @pageRows

DECLARE @exceptRows INT = (@page - 1) \* @pageRows

;WITH BillShow AS( SELECT t.name, b.DateCheckIn, b.TimeCheckIn, b.TimeCheckOut ,CONCAT(FORMAT(b.totalPrice, 'N0', 'vi-VN'), ',00 VNĐ') AS totalPrice, CONCAT(b.discount,'%') AS disCount, b.id

FROM dbo.Bill AS b,dbo.TableFood AS t

WHERE DateCheckIn >= @checkIn AND DateCheckOut <= @checkOut AND b.status = 1 AND t.id = b.idTable)

SELECT TOP (@selectRows) \* FROM BillShow WHERE id NOT IN (SELECT TOP (@exceptRows) id FROM BillShow)

END

GO

--============================lấy danh sách bill theo ngày=============================

ALTER PROC [dbo].[USP\_GetListBillByDate]

@checkIn date, @checkOut date

AS

BEGIN

SELECT t.name, b.DateCheckIn, b.TimeCheckIn, b.TimeCheckOut ,CONCAT(FORMAT(b.totalPrice, 'N0', 'vi-VN'), ',00 VNĐ') AS totalPrice, CONCAT(b.discount,'%') AS disCount, b.id

FROM dbo.Bill b, dbo.TableFood t

WHERE t.id = b.idTable AND b.status=1 AND DateCheckIn >= @checkIn AND DateCheckOut<=@checkOut

END

GO

--========================lấy thông tin account bởi name=======================

ALTER PROC [dbo].[USP\_GetAccountByUserName]

@userName nvarchar(100)

AS

BEGIN

SELECT \* FROM dbo.Account WHERE UserName = @userName

END

GO

--========================lấy dữ liệu để vẽ biểu đồ===============================

ALTER PROC [dbo].[USP\_GetDataForBieuDo]

@checkIn date, @checkOut date

AS

BEGIN

SELECT idTable,name, SUM(totalPrice) AS totalFinal,@checkIn AS 'DateCheckIn',@checkOut AS 'DateCheckOut'

FROM dbo.Bill, dbo.tableFood WHERE Bill.idTable=tableFood.id AND Bill.status=1 AND DateCheckIn >= @checkIn AND DateCheckOut<=@checkOut

GROUP BY idTable,name

END

GO

--=================================== sửa thông tin bàn===================================

ALTER PROC [dbo].[USP\_EditTable]

@tableName NVARCHAR(100), @statusTable NVARCHAR(100), @idTable INT

AS

BEGIN

UPDATE dbo.TableFood SET name=@tableName,status= @statusTable WHERE id= @idTable

END

GO

--===================================xóa bàn===================================

ALTER PROC [dbo].[USP\_DeleteTable]

@idTable INT

AS

BEGIN

DELETE dbo.Bill WHERE idTable=@idTable

DELETE dbo.TableFood WHERE id = @idTable

END

GO

--=================================== gộp bill===================================

ALTER PROC [dbo].[USP\_CombineTable]

@idTable1 INT, @idTable2 INT

AS BEGIN

DECLARE @idFirsrBill INT-- = 1023

DECLARE @idSeconrdBill INT --=1022

SELECT @idFirsrBill = dbo.Bill.id FROM dbo.Bill WHERE idTable = @idTable1 AND status=0

SELECT @idSeconrdBill = dbo.Bill.id FROM dbo.Bill WHERE idTable = @idTable2 AND status=0

IF(@idSeconrdBill IS NULL)

BEGIN

INSERT INTO dbo.Bill( DateCheckIn, DateCheckOut, idTable, status, discount)

VALUES ( GETDATE(),NULL, @idTable2, 0,0 )

SELECT @idSeconrdBill= MAX(id) FROM dbo.Bill

END

IF(@idFirsrBill IS NULL)

BEGIN

INSERT INTO dbo.Bill ( DateCheckIn, DateCheckOut, idTable, status, discount )

VALUES ( GETDATE(), NULL, @idTable1, 0, 0 )

SELECT @idFirsrBill= MAX(id) FROM dbo.Bill

END

-- BẮT ĐẦU XỬ LÝ

SELECT idBill,idFood,count INTO tem000 FROM dbo.BillInfo WHERE idBill= @idFirsrBill

UPDATE dbo.tem000 SET idBill=@idSeconrdBill

SELECT \* FROM dbo.tem000

DELETE dbo.BillInfo WHERE idBill = @idFirsrBill

DECLARE @tenBan2 NVARCHAR(100)

DECLARE @tenBan1 NVARCHAR(100)

SELECT @tenBan1= CONCAT(N'Đang được gộp với ',name) FROM tableFood WHERE id=@idTable1

SELECT @tenBan2 =CONCAT(N'Đang được gộp với ',name) FROM tableFood WHERE id=@idTable2

PRINT @tenBan1

PRINT @tenBan2

UPDATE dbo.TableFood SET status= @tenBan2 WHERE id= @idTable1 OR status=@tenban1

--UPDATE dbo.Bill SET status =1 WHERE id = @idFirsrBill

--KỊCH HAY BẮT ĐẦU

DECLARE tempCursor CURSOR FOR SELECT \* FROM dbo.tem000

OPEN tempCursor

DECLARE @idBill INT

DECLARE @idFood INT

DECLARE @count INT

FETCH NEXT FROM tempCursor INTO @idBill, @idFood, @count

WHILE @@FETCH\_STATUS = 0

BEGIN

EXEC dbo.UPR\_UpdateBillInfo @idBill =@idBill ,@idFood = @idFood, @count = @count

FETCH NEXT FROM tempCursor INTO @idBill, @idFood, @count

END

CLOSE tempCursor

DEALLOCATE tempCursor

DROP TABLE dbo.tem000

END

GO

--===================================thêm bàn mới===================================

ALTER PROC [dbo].[USP\_AddTable]

@tableName NVARCHAR(100)

AS

BEGIN

INSERT INTO dbo.TableFood

(

name,

status

)

VALUES

( @tableName, -- name - nvarchar(100)

N'Trống' -- status - nvarchar(100)

)

END

GO

--=================================== thêm bill mới ===================================

ALTER PROC [dbo].[UPR\_UpdateBillInfo]

@idBill INT , @idFood INT , @count INT

AS

BEGIN

DECLARE @existID INT = 0

DECLARE @countOld INT = 0

SELECT @existID= COUNT(\*) FROM dbo.BillInfo AS a WHERE a.idBill=@idBill AND a.idFood= @idFood

IF @existID >0

BEGIN

SELECT @countOld= a.count FROM dbo.BillInfo AS a WHERE a.idBill=@idBill AND a.idFood= @idFood

UPDATE dbo.BillInfo SET count = @countOld+@count WHERE idBill=@idBill AND idFood= @idFood

END

ELSE

BEGIN

INSERT INTO dbo.BillInfo(idBill,idFood,count)

VALUES( @idBill, @idFood, @count )

END

END

GO

--============================ xem bàn đã thanh toán chưa==============================

ALTER PROC [dbo].[dbo\_USPCheckFoodInBillInf]

@idFood INT, @idBill INT

AS

BEGIN

SELECT\*FROM dbo.BillInfo WHERE idFood=@idFood AND idBill= @idBill

END

* + - 1. **Các Trigger**

1. --================== thay đổi status bàn khi thêm món==================

ALTER TRIGGER UTG\_UpdateBillInfo

ON dbo.BillInfo FOR INSERT, UPDATE

AS

BEGIN

DECLARE @idBill INT

SELECT @idBill = idBill FROM Inserted

DECLARE @idTable INT

SELECT @idTable = idTable FROM dbo.Bill WHERE id = @idBill AND status = 0

UPDATE dbo.TableFood SET status = N'Có người' WHERE id = @idTable

END

GO

--==================thay đổi status của bàn khi thanh toán==================

ALTER TRIGGER UTG\_UpdateBill

ON dbo.Bill FOR UPDATE

AS

BEGIN

DECLARE @idBill INT

SELECT @idBill = id FROM Inserted

DECLARE @idTable INT

SELECT @idTable = idTable FROM dbo.Bill WHERE id = @idBill

DECLARE @count int = 0

SELECT @count = COUNT(\*) FROM dbo.Bill WHERE idTable = @idTable AND status = 0

IF (@count = 0)

UPDATE dbo.TableFood SET status = N'Trống' WHERE id = @idTable

END

GO

--==================thay xóa món khỏi bill nếu sl=0==================

ALTER TRIGGER UTG\_DeleteBillInfo

ON dbo.BillInfo FOR DELETE

AS

BEGIN

DECLARE @idBillInfo INT

DECLARE @idBill INT

SELECT @idBillInfo = id, @idBill = Deleted.idBill FROM Deleted

DECLARE @idTable INT

SELECT @idTable = idTable FROM dbo.Bill WHERE id = @idBill

DECLARE @count INT = 0

SELECT @count = COUNT(\*) FROM dbo.BillInfo AS bi, dbo.Bill AS b WHERE b.id = bi.idBill AND b.id = @idBill AND b.status = 0

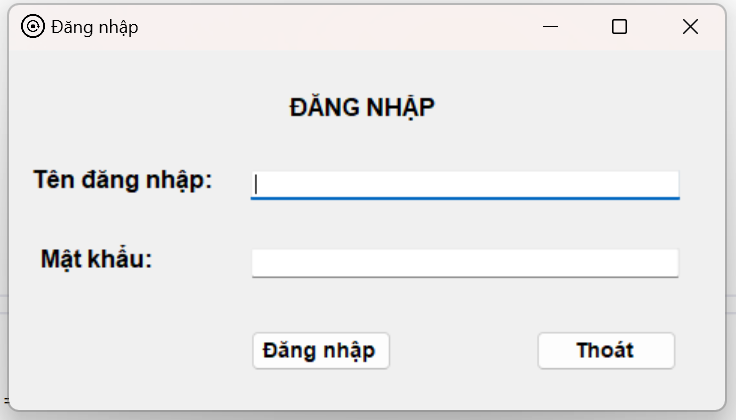
IF (@count = 0)

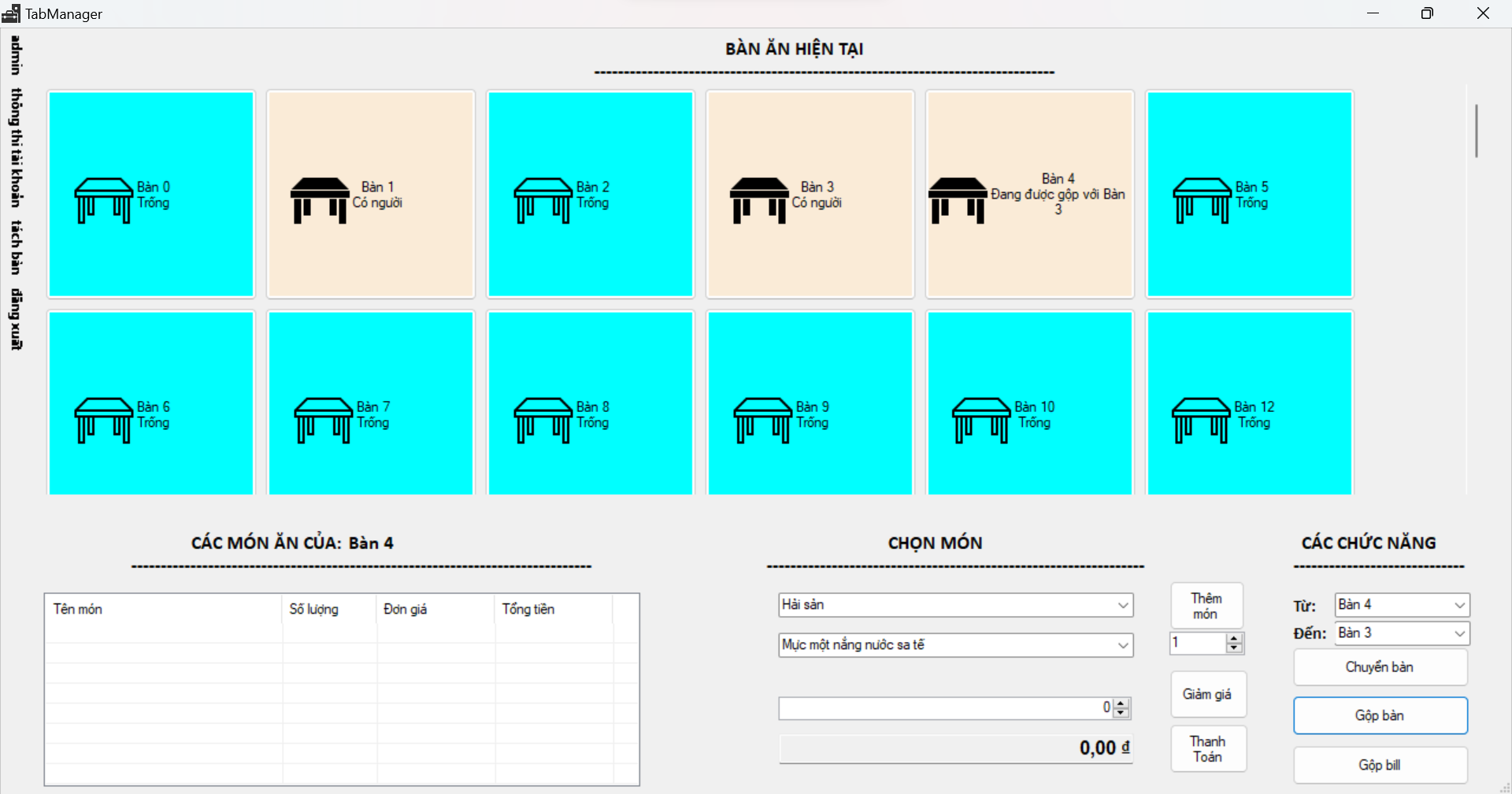
UPDATE dbo.TableFood SET status = N'Trống' WHERE id = @idTable

END

1. **Chương trình**

Khi mở chương trình sẽ phải đăng nhập





Khi chưa có cơ sở dữ liệu mình có thể nhấn nút đăng ký để đăng ký 1 tài khoản cơ sở dữ liệu

