|  |  |
| --- | --- |
| logomobile.png | **BỘ GIÁO DỤC VÀ ĐÀO TẠO**  **TRƯỜNG ĐẠI HỌC CÔNG NGHỆ TP. HCM** |

**BÀI TẬP LỚN**

Nghành: **CÔNG NGHỆ THÔNG TIN**

Lớp: **18DTHB3**

Sinh Viên Thực Hiện :  **Lê Tuấn Đạt-**

**Lê Tiến Anh-1811061252**

**Lê Hải Đăng-1811061887**

**Giảng Viên Hướng Dẫn : Gv.Thanh Văn**

TP.Hồ Chí Minh ,2021

**BỘ GIÁO DỤC VÀ ĐÀO TẠO**

**TRƯỜNG ĐẠI HỌC CÔNG NGHỆ TP. HCM**

**BÀI TẬP LỚN**

Nghành: **CÔNG NGHỆ THÔNG TIN**

Lớp: **18DTHB3**

Sinh Viên Thực Hiện : **Lê Tuấn Đạt-**

**Lê Tiến Anh-1811061252**

**Lê Hải Đăng-1811061887**

MỤC LỤC

Contents

[LỜI MỞ ĐẦU 5](#_Toc68702686)

[CHƯƠNG 1. TỔNG QUAN 6](#_Toc68702687)

[**1.1 Tổng quan về vấn đề nghiên cứu** 6](#_Toc68702688)

[*1.1.1 Khảo sát thực trạng* 6](#_Toc68702689)

[**1.2 Nhiệm vụ đồ án** 6](#_Toc68702690)

[*1.2.1 Đối tượng hướng đến* 6](#_Toc68702691)

[*1.2.2 Phạm vi nghiên cứu* 6](#_Toc68702692)

[*1.2.3 Thông tin tổng quan* 6](#_Toc68702693)

[*1.2.4 Phát biểu vấn đề* 6](#_Toc68702694)

[*1.2.5 Mục tiêu* 6](#_Toc68702695)

[*1.2.6 Mô tả* 6](#_Toc68702696)

[*1.2.7 Lợi ít mang lại* 6](#_Toc68702697)

[*1.2.8 Các bước thực hiện để hoàn thanh đồ án* 7](#_Toc68702698)

[CHƯƠNG 2. CƠ SỞ LÝ THUYẾT 7](#_Toc68702699)

[**2.1 Các khái niệm và cơ chế hoạt động** 7](#_Toc68702700)

[*2.1.1 C# WindownForm* 7](#_Toc68702701)

[*2.1.1.1 Môi trường lập trình* 7](#_Toc68702702)

[*2.1.1.2 Ngôn ngữ lập trình* 7](#_Toc68702703)

[2.1.1.3. Mô hình entity Famework 8](#_Toc68702704)

[*2.1.2. Tổng quan về hệ quản trị cơ sở dữ liệu SQL Server* 13](#_Toc68702705)

[*2.1.3 Yêu cầu hệ thống* 19](#_Toc68702706)

[*2.1.4 Yêu cầu chức năng* 19](#_Toc68702707)

[*2.1.5 Tồng quan DEVEXPRESS* 19](#_Toc68702708)

[**2.2 Mô hình giải phát** 20](#_Toc68702709)

[*2.2.1. Sơ đồ phân rã chức năng (BFD – Business Function Diagram)* 20](#_Toc68702710)

[20](#_Toc68702711)

[20](#_Toc68702712)

[20](#_Toc68702713)

[2.2.3. Sơ đồ thực thể kết hợp (ERD – Entity Relationship Diagram) 21](#_Toc68702714)

[*2.2.4. Mô hình dữ liệu quan hệ* 22](#_Toc68702715)

[**CHƯƠNG 3. KẾT QUẢ THỰC NGHIỆM** 23](#_Toc68702716)

[**3.1. Các thành phần chức năng của hệ thống** 23](#_Toc68702717)

[3.2. Thiết kế giao diện hệ thống 24](#_Toc68702718)

[3.2.1. Form Đăng nhập 24](#_Toc68702719)

[3.2.2. Form Giao diện chính của phần mềm 25](#_Toc68702720)

[3.2.3. From Doanh Thu 26](#_Toc68702721)

[**3.2.4. Form Thức Ăn** 27](#_Toc68702722)

[**3.2.5. Form Danh Mục** 28](#_Toc68702723)

[**3.2.6. Form Bàn Ăn** 29](#_Toc68702724)

[**3.2.7. Form Tài Khoản** 30](#_Toc68702725)

[**3.2.8. Form Quản Lý Nhân Viên** 31](#_Toc68702726)

[**3.2.9. Form Thông tin cá nhân** 32](#_Toc68702727)

[**Chương 4. Tổng kết.** 33](#_Toc68702728)

[*4.1. Kết quả đạt được* 33](#_Toc68702729)

[***4.2. Đánh giá phần mền*** 33](#_Toc68702730)

[*4.2.1. Kết quả đạt được* 33](#_Toc68702731)

[*4.2.2. Những mặt chưa làm được:* 33](#_Toc68702732)

[*4.3. Hướng phát triển và mở rộng đề tài* 33](#_Toc68702733)

[Tài liệu tham khảo 35](#_Toc68702734)

# LỜI MỞ ĐẦU

Công nghệ thông tin trong giai đoạn hiện nay đang phát triển như vũ báo ở mọi lĩnh vực hoạt động khắp nơi trên thế giới. Điều xảy ra trong vòng vài năm qua ở Việt Nam là sự đầu tư ô ạt vào công nghệ. Tin học đã và đang là một trong các vấn đề không thể thiếu đối với bất kỳ công ty nào. Đặc biệt tin học ngày càng giữ vai trò quan trọng trong vấn đề quản lý tại các tổ chức nhất là tại các lĩnh vực thu thập thông tin. Tuy ngành công nghệ thông tin ở nước ta mới chỉ phát triển trong vài năm trở lại đây và đang từng bước phát triển nhưng những bước phát triển đó đã cho thấy tiềm năng lớn lao trong lĩnh vực công nghệ thông tin ở nước ta, trong đó phát triển hệ thống thông tin đang là thế mạnh của đất nước. Hệ thống thông tin giúp công việc quản lý dễ dàng hơn, nâng cao hiệu quả sản xuất, tiết kiệm được lượng lớn thời gian và công sức.

Qua thời gian tìm hiểu thực tế để xác định đề tài đồ án, chúng em nhận thấy hệ thống Quán Lý Coffee là một trong những vấn đề rất quan trọng trong việc quản lý quán và nhân viên. Bởi tài sản cố định là thành phần không thể thiếu để trong một quán coffee có thể hoạt động và phát triển.

Do chúng em chưa có kinh nghiệm nghiên cứu, thực hành nên đề tài thực hiện còn nhiều thiếu sót. Chung em rất mong nhận được ý kiến đóng góp của Thầy để bài tập này được hoàn hảo hơn.

Chúng em xin chân thành cảm ơn!

Sinh viên thực hiện

**Lê Tuấn Đạt**

**Nguyễn Hải Đăng**

**Lê Tiến Anh**

# CHƯƠNG 1. TỔNG QUAN

## **1.1 Tổng quan về vấn đề nghiên cứu**

### *1.1.1 Khảo sát thực trạng*

Hiện nay, việc một một quán cà phê đòi hỏi phải có nhiều yếu: tài chính, vật chất, địa lý thuận lợi để giúp cho việc buôn bán phát triển. Bên cạnh đó thì việc quản lý quán cà phê của mình như thể nào là một câu hỏi lơn cần giải quyết. Vậy việc quản lý quán cà phê như thể nào bằng cách nào? Quản lý về cái gì? Quản lý như thế nào đc gọi là tiện lợi?? Ít tốn công sức nhưng lại cho ra kết quả hiệu quả.

## **1.2 Nhiệm vụ đồ án**

Từ các vấn đề trên các phần mềm quản lý được tạo ra nhằm hy vọng đem đến sự thuận tiện cho người sử dụng cũng như giúp cho chủ quán có thể xem xét một các tổng quát tình hình kinh doanh của quán mình.

### *1.2.1 Đối tượng hướng đến*

Phần mềm hướng đến tất cả người có nhu cầu kinh doanh quán cà phê cần phần mềm quản lí quán.

### *1.2.2 Phạm vi nghiên cứu*

Phạm vi quay quanh các nghiệp dụ bán hàng, tuân theo nghiệp vụ quản lý của một quán cà phê, đảm bảo đầy đủ các chức năng chính của phần mềm và khai thác các tính năng mới có liên quan đến quản lý quán cà phê nhằm giúp cho phần mềm thêm tiện tích.

### *1.2.3 Thông tin tổng quan*

Đơn vị sử dụng: các quán cà phê

Tên dự án: Phầm mềm quản lý quán coffee

### *1.2.4 Phát biểu vấn đề*

Mở ra một hướng đi mới trong việc quản lý một quán cà phê tiện ít hơn. Tổng hợp các chức năng cần thiết để quản lí một quán cà phê góp phần giúp ích cho việc kinh doanh và quản lý thu chi, nhân viên, hàng hóa... một cách nhanh chóng và hiệu quả.

### *1.2.5 Mục tiêu*

Hổ trợ các chủ kình doanh quán cà phê có thể quán lý doanh thu, nhân viên, hàng hóa một cách nhanh chóng và hiệu quả hơn.

### *1.2.6 Mô tả*

Xây đựng một hệ thống mới phù hợp, dễ hiểu, dễ sử dụng cho người dùng phần mềm.

Giúp quản lý hệ thống của quán trở nên tối ưu hơn, công tác quản lý dễ dàng và tiện lợi hơn.

### *1.2.7 Lợi ít mang lại*

Tạo sự tiện dụng, nhanh chóng dễ quản lý cho người chủ kinh doanh.

Tiết kiệm được thời gian và công sức.

### *1.2.8 Các bước thực hiện để hoàn thanh đồ án*

Lập kế hoạch phát triển hệ thống.

Phân thích hệ thống.

Thiết kế.

Cài đặt.

Kiểm tra.

Biên soạn tài liệu và hướng dẫn sử dụng.

# CHƯƠNG 2. CƠ SỞ LÝ THUYẾT

## **2.1 Các khái niệm và cơ chế hoạt động**

### *2.1.1 C# WindownForm*

### *2.1.1.1 Môi trường lập trình*

**Microsoft Visual Studio** là một môi trường phát triển tích hợp (IDE) từ Microsoft. Nó được sử dụng để phát triển chương trình máy tính cho Microsoft Windows, cũng như các trang web, các ứng dụng web và các dịch vụ web. Visual Studio sử dụng nền tảng phát triển phần mềm của Microsoft như Windows API, Windows Forms, Windows Presentation Foundation, Windows Store và Microsoft Silverlight. Nó có thể sản xuất cả hai ngôn ngữ máy và mã số quản lý.

Visual Studio bao gồm một trình soạn thảo mã hỗ trợ IntelliSense cũng như cải tiến mã nguồn. Trình gỡ lỗi tích hợp hoạt động cả về trình gỡ lỗi mức độ mã nguồn và gỡ lỗi mức độ máy. Công cụ tích hợp khác bao gồm một mẫu thiết kế các hình thức xây dựng giao diện ứng dụng, thiết kế web, thiết kế lớp và thiết kế giản đồ cơ sở dữ liệu. Nó chấp nhận các plug-in nâng cao các chức năng ở hầu hết các cấp bao gồm thêm hỗ trợ cho các hệ thống quản lý phiên bản (như Subversion) và bổ sung thêm bộ công cụ mới như biên tập và thiết kế trực quan cho các miền ngôn ngữ cụ thể hoặc bộ công cụ dành cho các khía cạnh khác trong quy trình phát triển phần mềm.

Visual Studio hỗ trợ nhiều ngôn ngữ lập trình khác nhau và cho phép trình biên tập mã và gỡ lỗi để hỗ trợ (mức độ khác nhau) hầu như mọi ngôn ngữ lập trình. Các ngôn ngữ tích hợp gồm có C, C++ và C++/CLI (thông qua Visual C++), VB.NET (thông qua Visual Basic.NET), C# (thông qua Visual C#) và F# (như của Visual Studio 2010). Hỗ trợ cho các ngôn ngữ khác như J++/J#, Python và Ruby thông qua dịch vụ cài đặt riêng rẽ. Nó cũng hỗ trợ XML/XSLT, HTML/XHTML, JavaScript và CSS.

Microsoft cung cấp phiên bản "Express" (đối với phiên bản Visual Studio 2013 trở về trước) và "Community" (đối với bản Visual Studio 2015 trở về sau) là phiên bản miễn phí của Visual Studio.

### *2.1.1.2 Ngôn ngữ lập trình*

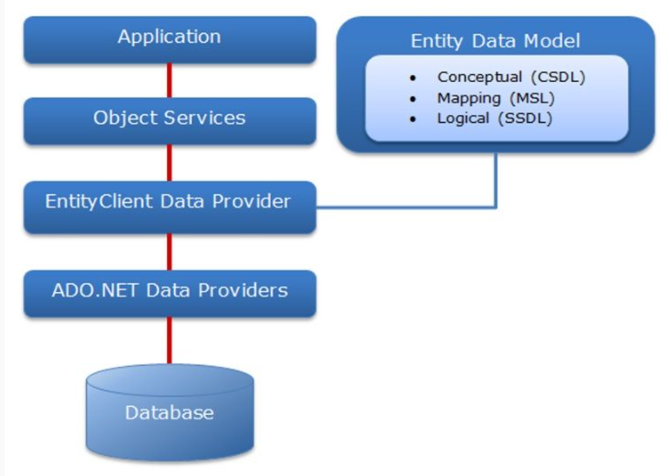
- C# (hay C sharp) là một ngôn ngữ lập trình đơn giản, được phát triển bởi đội ngũ kỹ sư của Microsoft vào năm 2000, trong đó người dẫn đầu là Anders Hejlsberg và Scott Wiltamuth.

- C# là ngôn ngữ lập trình hiện đại, hướng đối tượng và nó được xây dựng trên nền tảng của hai ngôn ngữ mạnh nhất là C++ và Java.

- C# được thiết kế cho Common Language Infrastructure (CLI), mà gồm Executable Code và Runtime Environment, cho phép chúng ta sử dụng các ngôn ngữ high-level đa dạng trên các nền tảng và cấu trúc máy tính khác nhau.

- C# với sự hỗ trợ mạnh mẽ của .NET Framework giúp cho việc tạo một ứng dụng Windows Forms hay WPF (Windows Presentation Foundation), . . . trở nên rất dễ dàng.

### 2.1.1.3. Mô hình entity Famework



hình 2.1.1.3 Mô hình entity famework

**-Application:**

+ Application(ứng dụng) là tầng chứa giao diện tương tác với người dùng, có thể là ứng dụng web hoặc là Windown Forms. Tầng này thường sử dụng Repository pattern để làm việc với cơ sở dữ liệu thông qua Entity Famework.

**- Object Services**

+ Object services là tầng chứa quá trình tương tác giữa ứng dụng và database, hay nói cách khác nó là nơi chủ yếu để truy cập dữ liệu từ database và trả ngược về giao diện.

Object services cung cấp các tiện ích để truy vết các thay đổi và cập nhật thay đổi trở lại database.

**-EntityClient Data Provide**

+ Đây là tầng cung cấp các kết nối , diễn dịch các truy vấn thực thể thành truy vấn nguồn dữ liệu , trả về data reader để Entity Famework dùng để chuyển dữ liệu thực thể thành các đối tượng.Phần này kết nối ADO.NET Data Provider để gửi hoặc lấy dữ liệu từ database

**-ADO.NET Data Provider**

+ Đây là tầng thấp nhất để dịch các truy vấn LinQ to entity thành các câu lệnh SQL và thực thi các câu lệnh trong hệ thống DBMS. Tầng này kết nối với database để sử dụng ADO.NET.

**- Entity Data Model(EDM)**

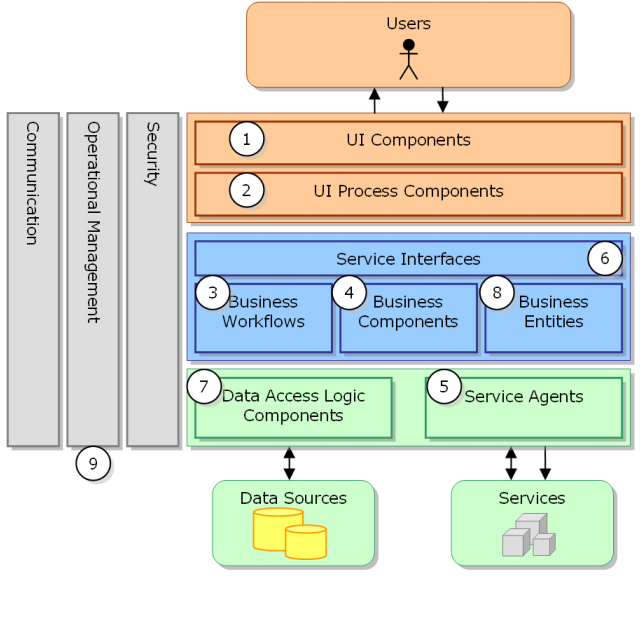
EDM gồm 3 phần chính : mô hình khái niệm(CSDL-Conceptual schema definition language),mô hình ánh xạ(MSL - mapping specfication language) và mô hình lưu trữ (SSDL - store schema definition language). Entity Data Model khác với EntityClient Data provider ở chỗ nó sử dụng LinQ là ngôn ngữ truy vấn tương tác với database.

+ Mô hình CSDL: Mô hình chứa các lớp mô hình và mối quan hệ giữa các lớp này . Nó độc lập với các mô hình quan hệ trong bản database.

+ Mô Hình ánh xạ (MSL): Mô hình ánh xạ gồm thông tin về cách mô hình khái niệm được ánh xạ đến mô hình lưu trữ

+ Mô hình lưu trữ(SSDL): Mô hình lưu trữ là một mô hình thiết kế database bao gồm các table , view ,stored procedure, mối quan hệ chúng các bản và khóa .Mô hình này thể hiện gần giống mô hình quan hệ các bảng trong database.

2.1.1.4 Mô hình 3 lóp



Mô hình 3-layer gồm có 3 phần chính:

*Presentation Layer (GUI)*

* Lớp này có nhiệm vụ chính là giao tiếp với người dùng. Nó gồm các thành phần giao diện ( winform, webform, …) và thực hiện các công việc như nhập liệu, hiển thị dữ liệu, kiểm tra tính đúng đắn dữ liệu trước khi gọi lớp Business Logic Layer (BLL).

*Business Logic Layer (BLL) Layer này phân ra 2 thành nhiệm vụ:*

* Đây là nơi đáp ứng các yêu cầu thao tác dữ liệu của GUI layer, xử lý chính nguồn dữ liệu từ Presentation Layer trước khi truyền xuống Data Access Layer và lưu xuống hệ quản trị CSDL.
* Đây còn là nơi kiểm tra các ràng buộc, tính toàn vẹn và hợp lệ dữ liệu, thực hiện tính toán và xử lý các yêu cầu nghiệp vụ, trước khi trả kết quả về Presentation Layer.

*Data Access Layer (DAL)*

* Lớp này có chức năng giao tiếp với hệ quản trị CSDL như thực hiện các công việc liên quan đến lưu trữ và truy vấn dữ liệu ( tìm kiếm, thêm, xóa, sửa,…).

**Presentation Layer (GUI)**

Có hai thành phần chính sau đây với những tác vụ cụ thể :

* **UI Components** : gồm các thành phần tạo nên giao diện của ứng dụng (GUI). Chúng chịu trách nhiệm thu nhận và hiển thị dữ liệu cho người dùng… Ví dụ : textbox, button, combobox, …
* **UI Process Components** : là thành phần chịu trách nhiệm quản lý các quá trình chuyển đổi giữa các UI… Ví dụ : Sắp xếp quá trình kiểm tra thông tin khách hàng:
  1. Hiển thị màn hình tra cứu ID.
  2. Hiển thị màn hình thông tin chi tiết khách hàng tương ứng.
  3. Hiển thị màn hình liên lạc với khách hàng.

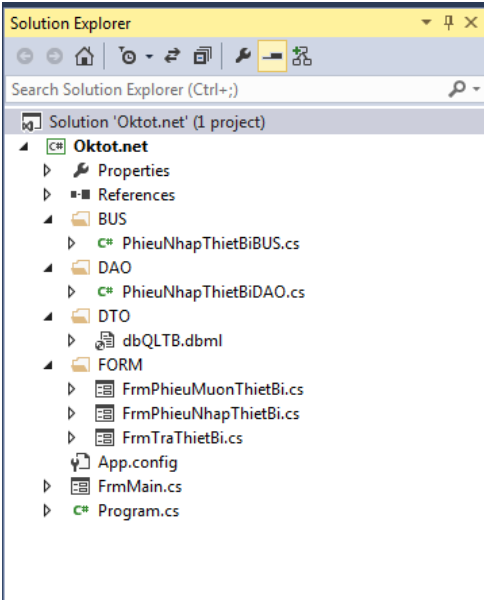
**Bussiness Layer (BLL)**

Lớp này gồm 4 thành phần:

* **Service Interface** : là thành phần giao diện lập trình mà lớp này cung cấp cho lớp Presentation sử dụng.
* **Bussiness Workflows** : chịu trách nhiệm xác định và điều phối các quy trình nghiệp vụ gồm nhiều bước và kéo dài. Những quy trình này phải được sắp xếp và thực hiện theo một thứ tự chính xác.
* **Bussiness Components** : chịu trách nhiệm kiểm tra các quy tắc nghiệp vụ, ràng buộc logic và thực hiện các công việc . Các thành phần này cũng thực hiện các dịch vụ mà Service Interface cung cấp và Business Workflows sẽ sử dụng nó.
* **Bussiness Entities** : thường được sử dụng như Data Transfer Objects ( DTO ) . Bạn có thể sử dụng để truyền dữ liệu giữa các lớp (Presentation và Data Layer). Chúng thường là cấu trúc dữ liệu ( DataSets, XML,… ) hay các lớp đối tượng đã được tùy chỉnh. Ví dụ : tạo 1 class Student lưu trữ các dữ liệu về tên, ngày sinh, ID, lớp.

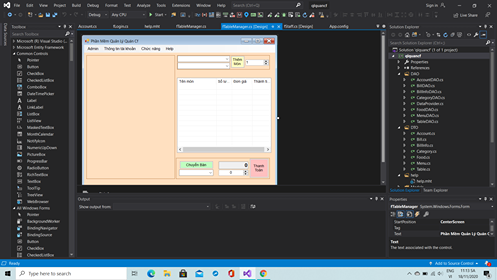
**Data Layer (DAL)**

* **Data Access Logic Components** : chịu trách nhiệm chính lưu trữ và truy xuất dữ liệu từ các nguồn dữ liệu (Data Sources) như XML, file system,… Hơn nữa còn tạo thuận lợi cho việc dễ cấu hình và bảo trì. Service Agents : giúp bạn gọi và tương tác với các dịch vụ từ bên ngoài một cách dễ dàng và đơn giản.



Hình ảnh tham khảo

* Đầu tiên là GUI gồm các button, texbox, ... mà người dùng sẽ tương tác với màn hình giao diện này.

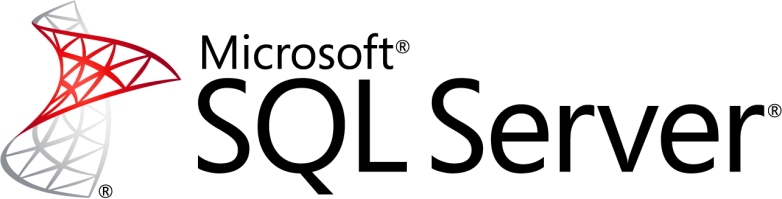


Lớp DTO, chứa những dữ liệu được xây dựng dưới dạng lớp đối tượng

Các nghiệp vụ xử lý chính sẽ được đặt ở lớp BUS (hay là BLL) gồm các nghiệp vụ insert, update, delete,..

Và cuối cùng là lớp DAO ( hay là DAL ). Truy vấn đến cơ sở dữ liệu

### *2.1.2. Tổng quan về hệ quản trị cơ sở dữ liệu SQL Server*



Có rất nhiều hệ quản trị cơ sở dữ liệu mạnh mẽ như: Oracle, My SQL… nhưng chúng em sẽ sử dụng SQL Server để thực hiện đề tài*“Xây dựng phần mềm quản lý điểm sinh viên”* này.

Hệ quản trị cơ sở dữ liệu Microsoft SQL Server (MSSQL) là một trong những hệ quản trị cơ sở dữ liệu thông dụng nhất hiện nay. Đây là hệ quản trị cơ sở dữ liệu thường được sử dụng trong các hệ thống trung bình, với ưu điểm có các công cụ quản lý mạnh mẽ giúp cho việc quản lý và bảo trì hệ thống dễ dàng, hỗ trợ nhiều phương pháp lưu trữ, phân vùng và đánh chỉ mục phục vụ cho việc tối ưu hóa hiệu năng.

SQL Server luôn được Microsoft cải tiến để nâng cao hiệu năng, tính sẵn sàng của hệ thống, khả năng mở rộng và bảo mật, cung cấp nhiều công cụ cho người phát triển ứng dụng được tích hợp với bộ Visual Studio do Microsoft cung cấp. SQL Server có 4 dịch vụ lớn là Database Engine, Intergration Service, Reporting service và Analysis Services.

* DataBase Engine: được phát triển để thực thi tốt hơn với việc hỗ trợ cả dữ liệu có cấu trúc và dữ liệu phi cấu trúc (XML).
* Khả năng sẵn sàng của hệ thống được nâng cao, hỗ trợ các chức năng: Database mirroring (cơ sở dữ liệu gương), failover clustering , snapshots và khôi phục dữ liệu nhanh.
* Việc quản lý chỉ mục được thực hiện song song với việc hoạt động của hệ thống. Người dùng có thể thêm chỉ mục, xây dựng lại chỉ mục hay xóa một chỉ mục đi trong khi hệ thống vẫn được sử dụng.
* Chức năng phân vùng dữ liệu được hỗ trợ: Người dùng có thể phân vùng các bảng và chỉ mục cũng như quản lý phân vùng dữ liệu một cách dễ dàng. Việc hỗ trợ phân vùng dữ liệu giúp nâng cao hiệu năng hoạt động của hệ thống.
* Dịch vụ đồng bộ hóa dữ liệu được mở rộng với việc hỗ trợ mô hình đồng bộ hóa ngang hàng. Đây là dịch vụ giúp đồng bộ hóa dữ liệu giữa các máy chủ dữ liệu, giúp mở rộng khả năng của hệ thống.
* Dịch vụ tích hợp (Integration Service) thiết kế lại cho phép người dùng tích hợp dữ liệu và phân tích dữ liệu từ nhiều nguồn khác nhau. Hỗ trợ việc quản lý chất lượng dữ liệu và làm sạch dữ liệu, một công việc quan trọng trong tiến trình ETL.
* Dịch vụ phân tích dữ liệu (Analysis Service): cung cấp khung nhìn tích hợp và thống nhất về dữ liệu cho người dùng, hỗ trợ việc phân tích dữ liệu.
* Công cụ khai phá dữ liệu (Data mining) được tích hợp hỗ trợ nhiều thuật toán khai phá dữ liệu, hỗ trợ cho việc phân tích, khai phá dữ liệu và xây dựng các hệ thống hỗ trợ ra quyết định cho người quản lý.
* Dịch vụ xây dựng quản lý báo cáo (Reporting Service) dựa trên nền tảng quản trị doanh nghiệp thông minh và được quản lý qua dịch vụ web. Báo cáo có thể được xây dựng dễ dàng với ngôn ngữ truy vấn MDX. Thông qua các công cụ trên Business Intelligent, người dùng dễ dàng truy cập báo cáo và trích xuất ra nhiều định dạng khác nhau thông qua trình duyệt web.

*\* Tổng quan về MD5*

Trong [mật mã học](https://vi.wikipedia.org/wiki/M%E1%BA%ADt_m%C3%A3_h%E1%BB%8Dc), **MD5** (viết tắt của [tiếng Anh](https://vi.wikipedia.org/wiki/Ti%E1%BA%BFng_Anh) **Message-Digest algorithm 5**, giải thuật Tiêu hóa tin 5) là một [hàm băm mật mã học](https://vi.wikipedia.org/wiki/H%C3%A0m_b%C4%83m_m%E1%BA%ADt_m%C3%A3_h%E1%BB%8Dc) được sử dụng phổ biến với giá trị Hash dài 128-[bit](https://vi.wikipedia.org/wiki/Bit). Là một chuẩn [Internet](https://vi.wikipedia.org/wiki/Internet) ([RFC 1321](http://tools.ietf.org/html/1321)), MD5 đã được dùng trong nhiều ứng dụng bảo mật, và cũng được dùng phổ biến để kiểm tra tính toàn vẹn của [tập tin](https://vi.wikipedia.org/w/index.php?title=T%E1%BA%ADp_tin_m%C3%A1y_t%C3%ADnh&action=edit&redlink=1). Một bảng băm MD5 thường được diễn tả bằng một số [hệ thập lục phân](https://vi.wikipedia.org/wiki/H%E1%BB%87_th%E1%BA%ADp_l%E1%BB%A5c_ph%C3%A2n) 32 ký tự.

MD5 được thiết kế bởi [Ronald Rivest](https://vi.wikipedia.org/w/index.php?title=Ronald_Rivest&action=edit&redlink=1) vào năm [1991](https://vi.wikipedia.org/wiki/1991) để thay thế cho hàm băm trước đó, [MD4](https://vi.wikipedia.org/w/index.php?title=MD4&action=edit&redlink=1). Vào năm [1996](https://vi.wikipedia.org/wiki/1996), người ta phát hiện ra một lỗ hổng trong MD5; trong khi vẫn chưa biết nó có phải là lỗi nghiêm trọng hay không, những chuyên gia mã hóa bắt đầu đề nghị sử dụng những giải thuật khác, như [SHA-1](https://vi.wikipedia.org/w/index.php?title=H%C3%A0m_b%C4%83m_SHA&action=edit&redlink=1) (khi đó cũng bị xem là không an toàn). Trong năm [2004](https://vi.wikipedia.org/wiki/2004), nhiều lỗ hổng hơn bị khám phá khiến cho việc sử dụng giải thuật này cho mục đích bảo mật đang bị đặt nghi vấn.

ỨNG DỤNG:

Các đồng hóa MD5 được dùng rộng rãi trong các phần mềm trên toàn thế giới để đảm bảo việc truyền tập tin được nguyên vẹn. Ví dụ, máy chủ tập tin thường cung cấp một [checksum](https://vi.wikipedia.org/wiki/Gi%C3%A1_tr%E1%BB%8B_t%E1%BB%95ng_ki%E1%BB%83m) MD5 được tính toán trước cho tập tin, để người dùng có thể so sánh với checksum của tập tin đã tải về. Những hệ điều hành dựa trên nền tảng [Unix](https://vi.wikipedia.org/wiki/Unix) luôn kèm theo tính năng MD5 sum trong các gói phân phối của họ, trong khi người dùng Windows sử dụng ứng dụng của hãng thứ ba.

Tuy nhiên, hiện nay dễ dàng tạo ra xung đột MD5, một người có thể tạo ra một tập tin để tạo ra tập tin thứ hai với cùng một checksum, do đó kỹ thuật này không thể chống lại một vài dạng giả mạo nguy hiểm. Ngoài ra, trong một số trường hợp checksum không thể tin tưởng được (ví dụ, nếu nó được lấy từ một lệnh như tập tin đã tải về), trong trường hợp đó MD5 chỉ có thể có chức năng kiểm tra lỗi: nó sẽ nhận ra một lỗi hoặc tải về chưa xong, rất dễ xảy ra khi tải tập tin lớn.

MD5 được dùng rộng rãi để lưu trữ [mật khẩu](https://vi.wikipedia.org/wiki/M%E1%BA%ADt_kh%E1%BA%A9u_truy_nh%E1%BA%ADp). Để giảm bớt sự dễ thương tổn đề cập ở trên, ta có thể thêm [salt](https://vi.wikipedia.org/w/index.php?title=Salt_(m%E1%BA%ADt_m%C3%A3_h%E1%BB%8Dc)&action=edit&redlink=1) vào mật khẩu trước khi băm chúng. Một vài hiện thực có thể áp dụng vào hàm băm hơn một lần-xem [làm mạnh thêm khóa](https://vi.wikipedia.org/w/index.php?title=L%C3%A0m_m%E1%BA%A1nh_th%C3%AAm_kh%C3%B3a&action=edit&redlink=1).

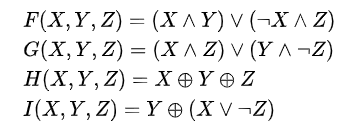
GIẢI THUẬT:

MD5 chuyển một đoạn thông tin chiều dài thay đổi thành một kết quả chiều dài không đổi 128 bit. Mẩu tin đầu vào được chia thành từng đoạn 512 bit; mẩu tin sau đó được [độn](https://vi.wikipedia.org/w/index.php?title=%C4%90%E1%BB%99n_(m%E1%BA%ADt_m%C3%A3_h%C3%B3a)&action=edit&redlink=1) sao cho chiều dài của nó chia chẵn cho 512. Công việc độn vào như sau: đầu tiên một bit đơn, 1, được gắn vào cuối mẩu tin. Tiếp theo là một dãy các số zero sao cho chiều dài của mẩu tin lên tới 64 bit ít hơn so với bội số của 512. Những bit còn lại được lấp đầy bằng một số nguyên 64-bit đại diện cho chiều dài của mẩu tin gốc.

Giải thuật MD5 chính hoạt động trên trạng thái 128-bit, được chia thành 4 từ 32-bit, với ký hiệu *A*, *B*, *C* và *D*. Chúng được khởi tạo với những hằng số cố định. Giải thuật chính sau đó sẽ xử lý các khối tin 512-bit, mỗi khối xác định một trạng thái. Quá trình xử lý khối tin bao gồm bốn giai đoạn giống nhau, gọi là *vòng*; mỗi vòng gồm có 16 tác vụ giống nhau dựa trên hàm phi tuyến *F*, [cộng mô đun](https://vi.wikipedia.org/w/index.php?title=C%E1%BB%99ng_m%C3%B4_%C4%91un&action=edit&redlink=1), và dịch trái. Hình 1 mô tả một tác vụ trong một vòng. Có 4 khả năng cho hàm *F*; mỗi cái được dùng khác nhau cho mỗi vòng:

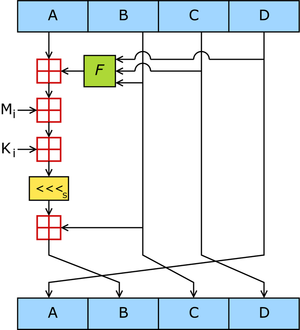
{\displaystyle F(X,Y,Z)=(X\wedge {Y})\vee (\neg {X}\wedge {Z})}

{\displaystyle G(X,Y,Z)=(X\wedge {Z})\vee (Y\wedge \neg {Z})}

{\displaystyle H(X,Y,Z)=X\oplus Y\oplus Z} 

{\displaystyle I(X,Y,Z)=Y\oplus (X\vee \neg {Z})}

{\displaystyle \oplus ,\wedge ,\vee ,\neg } lần lượt chỉ phép [XOR](https://vi.wikipedia.org/w/index.php?title=XOR&action=edit&redlink=1), [AND](https://vi.wikipedia.org/wiki/Ph%C3%A9p_giao), [OR](https://vi.wikipedia.org/w/index.php?title=Ph%C3%A9p_h%E1%BB%99i&action=edit&redlink=1) và [NOT](https://vi.wikipedia.org/w/index.php?title=Ph%C3%A9p_ph%E1%BB%A7_%C4%91%E1%BB%8Bnh&action=edit&redlink=1).



Hình 1.  Một thao tác MD5—MD5 bao gồm 64 tác vụ thế này, nhóm trong 4 vòng 16 tác vụ

MÃ GIẢ:

//*Chú ý: Tất cả các biến đều là biến không dấu 32 bit và bao phủ mô đun 2^32 khi tính toán*

**var** *int*[64] r, k

//*r xác định số dịch chuyển mỗi vòng*

r[ 0..15]:= {7, 12, 17, 22, 7, 12, 17, 22, 7, 12, 17, 22, 7, 12, 17, 22}

r[16..31]:= {5, 9, 14, 20, 5, 9, 14, 20, 5, 9, 14, 20, 5, 9, 14, 20}

r[32..47]:= {4, 11, 16, 23, 4, 11, 16, 23, 4, 11, 16, 23, 4, 11, 16, 23}

r[48..63]:= {6, 10, 15, 21, 6, 10, 15, 21, 6, 10, 15, 21, 6, 10, 15, 21}

//*Sử dụng phần nguyên nhị phân của sin của số nguyên làm hằng số:*

**for** i **from** 0 **to** 63

k[i]:= floor(abs(sin(i + 1)) × (2 **pow** 32))

//*Khởi tạo biến:*

**var** *int* h0:= 0x67452301

**var** *int* h1:= 0xEFCDAB89

**var** *int* h2:= 0x98BADCFE

**var** *int* h3:= 0x10325476

//*Tiền xử lý:*

**append** "1" bit **to** message

**append** "0" bits **until** message length in bits ≡ 448 (mod 512)

**append** bit (bit, not byte) length of unpadded message **as** *64-bit little-endian integer* **to** message

//*Xử lý mẩu tin trong đoạn 512-bit tiếp theo:*

**for each** *512-bit* chunk **of** message

break chunk into sixteen 32-bit little-endian words w[i], 0 ≤ i ≤ 15

//*Khởi tạo giá trị băm cho đoạn này:*

**var** *int* a:= h0

**var** *int* b:= h1

**var** *int* c:= h2

**var** *int* d:= h3

//*Vòng lặp chính:*

**for** i **from** 0 **to** 63

**if** 0 ≤ i ≤ 15 **then**

f:= (b **and** c) **or** ((**not** b) **and** d)

g:= i

**else if** 16 ≤ i ≤ 31

f:= (d **and** b) **or** ((**not** d) **and** c)

g:= (5×i + 1) **mod** 16

**else if** 32 ≤ i ≤ 47

f:= b **xor** c **xor** d

g:= (3×i + 5) **mod** 16

**else if** 48 ≤ i ≤ 63

f:= c **xor** (b **or** (**not** d))

g:= (7×i) **mod** 16

temp:= d

d:= c

c:= b

b:= b + **leftrotate**((a + f + k[i] + w[g]), r[i])

a:= temp

//*Thêm bảng băm của đoạn vào kết quả:*

h0:= h0 + a

h1:= h1 + b

h2:= h2 + c

h3:= h3 + d

**var** *int* digest:= h0 **append** h1 **append** h2 **append** h3 //*(expressed as little-endian)*

//*định nghĩa hàm dịch trái*

**leftrotate** (x, c)

return (x << c) **or** (x >> (32-c));

*Ghi chú: Thay vì hàm hóa*[*RFC 1321*](http://tools.ietf.org/html/1321)*gốc như trên, phần sau có thể được dùng để tăng độ hiệu quả (hữu ích nếu ngôn ngữ assembly được dùng - còn không, chương trình dịch sẽ tự động tối ưu hóa đoạn mã ở trên):*

(0 ≤ i ≤ 15): f:= d **xor** (b **and** (c **xor** d))

(16 ≤ i ≤ 31): f:= c **xor** (d **and** (b **xor** c))

CÁC BẢNG BĂM MD5:

Bảng băm MD5 128 bit (16 byte) (còn được gọi là *message digests*) được biểu diễn bằng chuỗi 32 số [thập lục phân](https://vi.wikipedia.org/w/index.php?title=Th%E1%BA%ADp_l%E1%BB%A5c_ph%C3%A2n&action=edit&redlink=1). Sau đây cho thấy đầu vào [ASCII](https://vi.wikipedia.org/wiki/ASCII) 43 byte và bảng băm MD5 tương ứng:

MD5("[The quick brown fox jumps over the lazy dog](https://vi.wikipedia.org/wiki/The_quick_brown_fox_jumps_over_the_lazy_dog)")

= 9e107d9d372bb6826bd81d3542a419d6

Thậm chí một sự thay đổi nhỏ trong mẩu tin cũng dẫn đến thay đổi hoàn toàn bảng băm, do [hiệu ứng thác](https://vi.wikipedia.org/w/index.php?title=Hi%E1%BB%87u_%E1%BB%A9ng_th%C3%A1c&action=edit&redlink=1). Ví dụ, thay d thành e:

MD5("The quick brown fox jumps over the lazy **e**og")

= ffd93f16876049265fbaef4da268dd0e

Bảng băm của một chuỗi rỗng là:

MD5("")

= d41d8cd98f00b204e9800998ecf8427e

### *2.1.3 Yêu cầu hệ thống*

- Hệ thống cơ sở dữ liệu đủ lớn để đáp ứng số lượng người dùng ngày càng tăng.

- Máy chủ có khả năng lưu trữ lâu dài và chính xác, đảm bảo bảo mật thông tin cho người dùng.

- Thông tin người dùng được đồng bộ và quản lý chặt chẽ.

- Hệ thống có khả năng đáp ứng lượng truy cập lớn.

### *2.1.4 Yêu cầu chức năng*

- Hệ thống phải cập nhật dữ liệu thông tin người dùng thường xuyên.

- Cập nhật nhanh các yêu cầu thông báo hỗ trợ của người dùng.

### *2.1.5 Tồng quan DEVEXPRESS*

 DevExpress là một Framework được viết cho nền tảng .NET Framework. Nó cung cấp các control và công nghệ để phục vụ cho quá trình phát triển phần mềm. Thành phần của DevExpress gồm:

* WinForms Controls: Cung cấp các control cho WinForms.
* ASP.NET Controls: Cung cấp các control cho WebForms.
* WPF Controls: Cung cấp các control cho WPF.
* Silverlight Controls: Cung cấp các control cho Silverlight.
* XtraCharts: Control cung cấp các loại biểu đồ.
* XtraReports: Cung cấp các control tạo báo cáo.
* XPO: Cung cấp môi trường làm việc với database.
* XAF: Một công nghệ mới giúp việc phát triển phần mềm một cách nhanh chóng.

       Trải qua hàng loạt phiên bản, DevExpress đã từng bước được nâng cấp, hoàn thiện và thêm mới rất nhiều chức năng. Với phiên bản DevExpress 11.2 hiện tại, bạn đã được cung cấp những công cụ, môi trường tuyệt vời để biến những ý tưởng của bạn thành hiện thực một cách nhanh chóng, dễ dàng.

## **2.2 Mô hình giải phát**

### *2.2.1. Sơ đồ phân rã chức năng (BFD – Business Function Diagram)*

**QUẢN LÍ QUÁN CAFE ROPTOP**

Tra cứu

Xử lý

Quản lý hệ thống

Cập nhật danh mục

Tìm kiếm món

Lập hóa đơn

Quản lý bàn

Đăng nhập

Tìm kiếm nhân viên

Thanh toán

Quản lý nhân viên

Đăng xuất

Gọi món

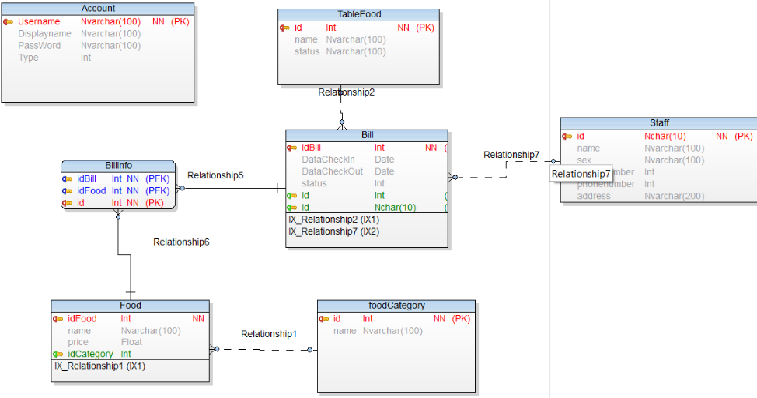
Thêm món

Quản lý danh sách thực đơn

Thống kê doanh thu

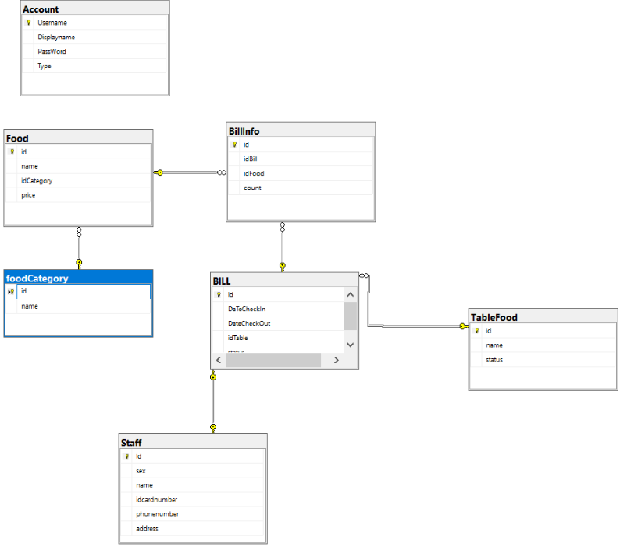
### 

### 2.2.3. Sơ đồ thực thể kết hợp (ERD – Entity Relationship Diagram)



### 

### *2.2.4. Mô hình dữ liệu quan hệ*



# 

# **CHƯƠNG 3. KẾT QUẢ THỰC NGHIỆM**

## **3.1. Các thành phần chức năng của hệ thống**

* **Chức năng đăng nhập hệ thống:**
* Đăng nhập tài khoản để xử lý chương trình.
* **Menu:** chứa các chức năng chính của chương trình.
* *Hệ thống:* trở về trang chủ, đăng xuất khỏi trang quản lý, thoát khỏi chương trình.

## 3.2. Thiết kế giao diện hệ thống

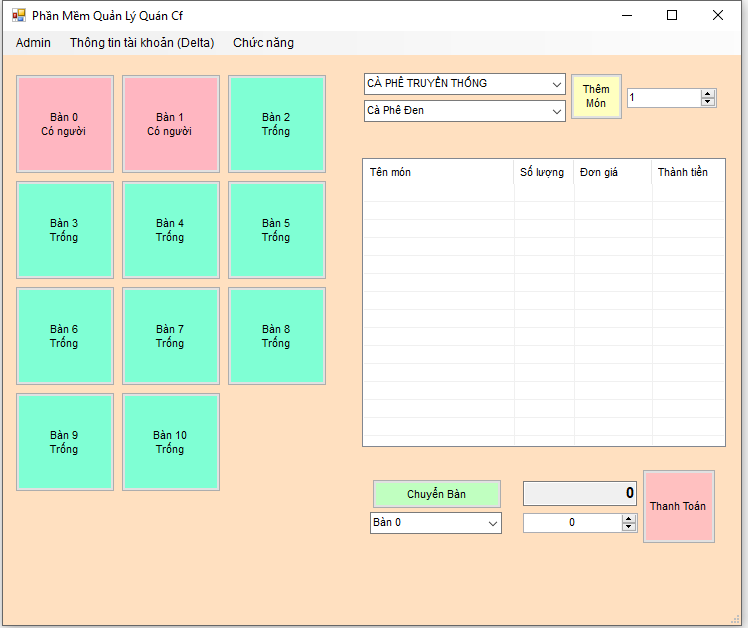
### 3.2.1. Form Đăng nhập

Đây là form chạy đầu tiên của phần mềm



**-**Form đăng nhập: Hiển thị thông tin gồm tên đăng nhập và mật khẩu yêu cầu người dùng đăng nhập và tiếp tục sử dụng hoặc thoát.

### 3.2.2. Form Giao diện chính của phần mềm



**-**Form giao diện chính của phần mền: Sau khi đăng nhập người dùng có thể sử dụng phần mền.

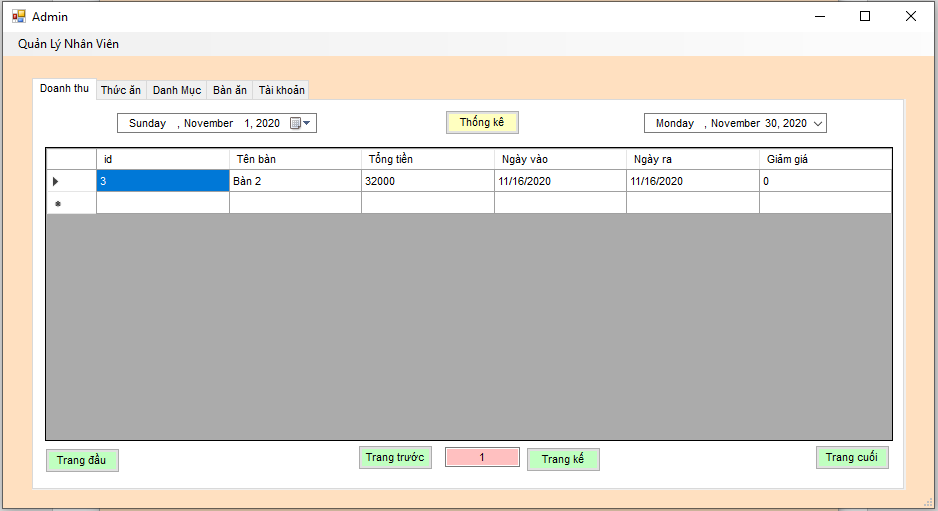
-Form giao diện gồm 3 tab: Admin, thông tin tài khoản, chức năng.

-Nút thêm món: Lựa chọn món khách gọi và nhấn vào thêm món để đưa vào danh sách.

-Nút Thanh Toán: Cho phép nhân viên thanh toán khi khách yêu cầu thanh toán.

-Nút Chuyển Bàn: Cho Phép nhân viên chuyển bàn theo yêu cầu của khách.

### 3.2.3. From Doanh Thu



-Form Doanh thu bao gồm: ID, Tên bàn, Tổng tiền, Ngày vào, Ngày ra và giảm giá.

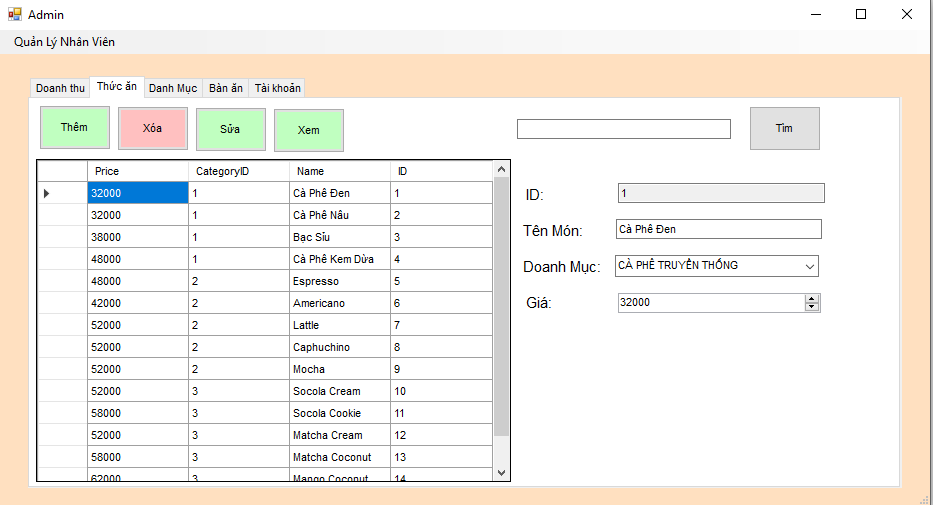
-Nút Trang đầu: Cho phép người dùng đi từ trang cuối đén trang đầu mà không cần phải qua từng trang.

-Nút Trang cuối: Ngược lại với Nút Trang đầu.

-Nút Trang trước: Cho phép người dùng đi đến 1 trang ở trước.

-Nút Trang kế: Cho phép người dùng đi đến trang kế tiếp.

### **3.2.4. Form Thức Ăn**



**-**Form bao gồm: Price, CategoryID, Name, ID.

-Form gồm các chức năng Thêm, Xóa, Sửa, Xem và Tìm.

-Nút Tìm: cho phép người dùng tìm kiếm thức ăn cần tìm một cách nhanh.

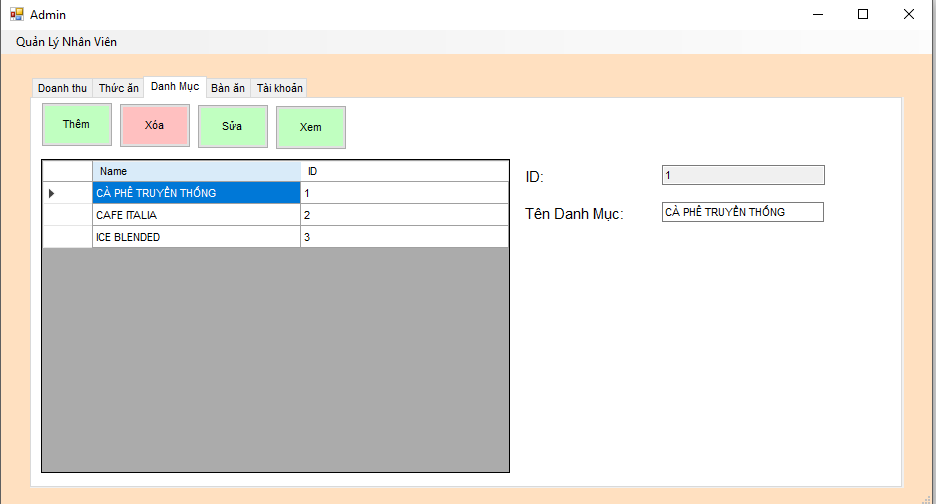
-Nút Thêm: Thêm một món vào danh sách.

-Nút Xóa: Xóa một món khỏi danh sách.

-Nút Sửa: Sửa bất kì các món có trong danh sách.

-Nút Xem: Cho phép người dùng xem thông tin của món.

### **3.2.5. Form Danh Mục**



-Form bao gồm thông tin: Name, ID.

-Form gồm các chức năng Thêm, Xóa, Sửa và Xem.

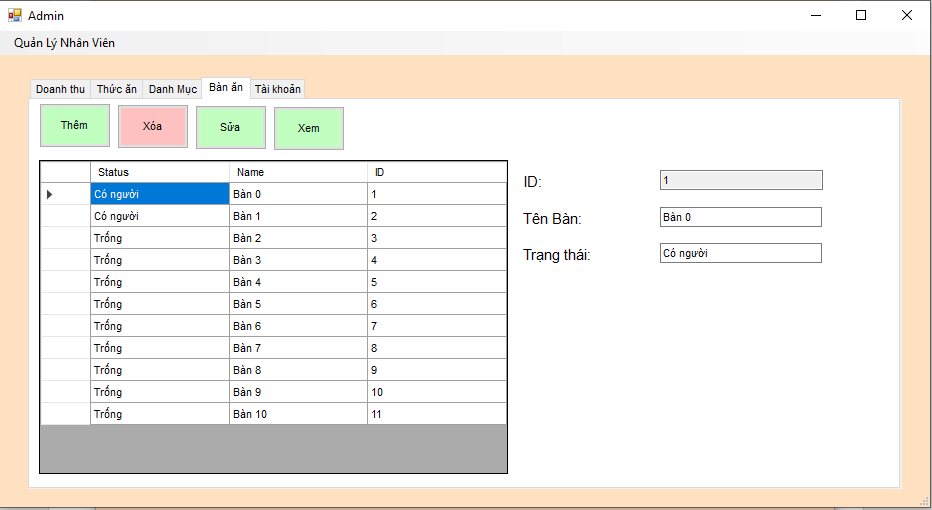
-Nút Thêm: Cho phép người dùng thêm tên một món ăn vào danh sách.

-Nút Xóa: Xóa một món khỏi danh sách.

-Nút Sửa: Sửa bất kì các món có trong danh sách.

-Nút Xem: Cho phép người dùng xem thông tin của món.

### **3.2.6. Form Bàn Ăn**



-Form bao gồm thông tin: Satus, Nam và ID.

-Form gồm các chức năng Thêm, Xóa, Sửa và Xem.

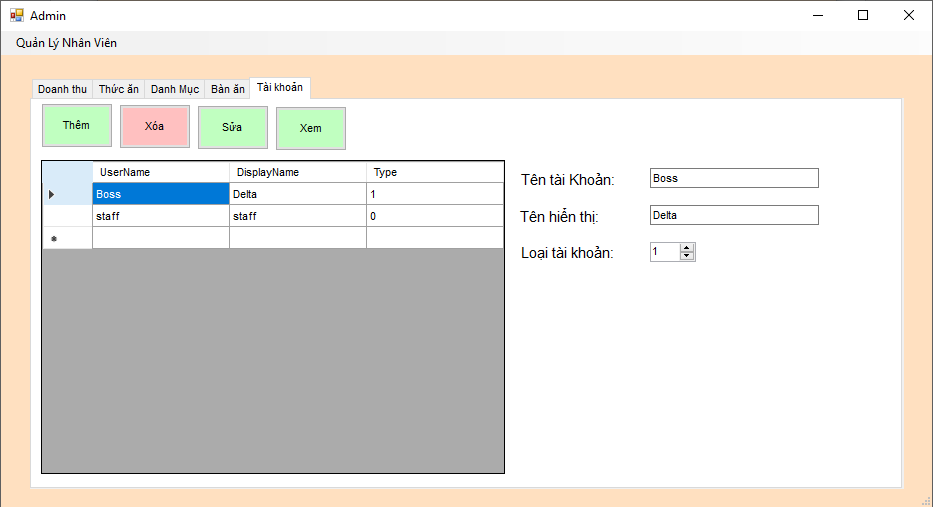
-Nút Thêm: Cho phép người dùng thêm tên một bàn vào danh sách.

-Nút Xóa: Xóa một bàn khỏi danh sách.

-Nút Sửa: Sửa bất kì các bàn có trong danh sách.

-Nút Xem: Cho phép người dùng xem thông tin của bàn.

### **3.2.7. Form Tài Khoản**



-Form gồm thông tin: UserName, DisplayName và Type.

-Form gồm các chức năng Thêm, Xóa, Sửa và Xem.

-Nút Thêm: Cho phép người dùng thêm tên một món ăn vào danh sách.

-Nút Xóa: Xóa một món khỏi danh sách.

-Nút Sửa: Sửa bất kì các món có trong danh sách.

-Nút Xem: Cho phép người dùng xem thông tin của món.

### **3.2.8. Form Quản Lý Nhân Viên**

****

-Form bao gồm thông tin: Mã Nhân Viên, Tên Nhân Viên, Giới tính, CMND, Số điện thoại và Địa chỉ.

-Form gồm các chức năng Thêm, Xóa, Sửa, Xem và Tìm.

-Nút Thêm: Cho phép người dùng thêm tên một một nhân viên vào danh sách.

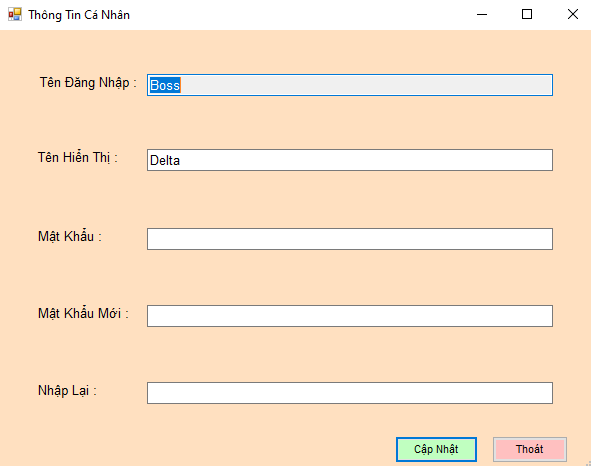
-Nút Xóa: Xóa một nhân viên khỏi danh sách.

-Nút Sửa: Sửa bất kì một nhân viên có trong danh sách.

-Nút Xem: Cho phép người dùng xem thông tin của một nhân viên.

-Nút Tìm: Cho phép người dùng tìm một nhân viên bất kì một cách nhanh chóng.

### **3.2.9. Form Thông tin cá nhân**



-Form gồm thông tin: Tên đăng nhập, Tên hiển thị, Mật khẩu, Mật khẩu mới và Nhập lại.

-Nút Cập Nhật: Điền thông tin theo yêu cầu vào các ô cần thiết và cập nhật giúp người dùng thay đổi mật khẩu.

-Nút Thoát: Thoát ra đơn giản.

# **Chương 4. Tổng kết.**

## *4.1. Kết quả đạt được*

- Xây dựng phần mềm quản lý nói chung và phần mềm quản lý điểm nói riêng không chỉ là việc xây dựng đơn thuần mà đòi hỏi trước đó phải tiến hành một cách có hệ thống các giai đoạn khảo sát, phân tích thiết kế.

- Đứng trước xu thế phát triển của công nghệ thông tin như vũ bão hiện nay thì việc xây dựng phần mềm quản lý điểm trong các trường học điều thiết yếu và mang tính thực tế cao. Qua việc nghiên cứu xây dựng đề tài này, chúng em đã phần nào củng cố được các kiến thức về lập trình và nắm được cách làm thế nào để xây dựng được một phần mềm. Từ đó, chúng em có thể xây dựng được những phần mềm tương tự như: quản lý thư viện, quản lý nhân sự, quản lý khách sạn…

## ***4.2. Đánh giá phần mền***

## *4.2.1. Kết quả đạt được*

- Nội dung lý thuyết được củng cố như:

+ Phân tích, thiết kế hệ thống cơ sở dữ liệu

+ Làm việc trên môi trường .NET, đặc biệt là ngôn ngữ c#.

+ Vận dụng được kỹ thuật thiết kế phần mềm theo mô hình 3 lớp.

+ Sử dụng tốt các chức năng cơ bản của các phần mềm hỗ trợ như:

Microsoft Word, StarUML,...

- Phần mềm cơ bản đã hoạt động tốt, ổn định, đáp ứng những nhu cầu, chức năng

nghiệp vụ cơ bản.

- Giao diện bắt mắt nhưng đơn giản, các ô hiển thị các chức năng to, rõ ràng,...

- Hoàn thành đồ án đúng thời gian quy định.

## *4.2.2. Những mặt chưa làm được:*

- Phần mềm còn thiếu sót, vẫn còn xuất hiện tình trạng lỗi,...

- Chức năng sao lưu cơ sở dữ liệu chưa lập được lịch sao lưu tự động

- Vẫn có các chức năng chỉ nằm ở mức ý tưởng.

- Giao diện vẫn còn đơn sơ,...

## *4.3. Hướng phát triển và mở rộng đề tài*

-Biết tháng đó lãi là bao nhiêudựa trên công thức có sẵn.

-Tiếp tục trao dồi kiến thức đặt biệt là cơ sở dữ liệu và kỹ thuật lập trình để

giải quyết các khuyết điểm trên

- Quản lý xuất nhập kho

- Nguyên liệu tồn kho mang tính chính xác hơn

- Thống kê được doanh thu (lãi) thực tế theo định kỳ, để chủ quán có hướng

điều chỉnh chi tiêu, và kế họach bán quán cho phù hợp.

- Chúng em xin chân thành cảm ơn!

# Tài liệu tham khảo

* Dương Thành Phết. Slide bài giảng Lập trình Java.
* Dương Thành Phết. Bài tập thực hành Lập trình Java.
* Văn Như Bích - Võ Hoàng Khang. Giáo trình Cơ sở dữ liệu.
* Văn Như Bích - Võ Hoàng Khang. Bài tập thực hành Cơ sở dữ liệu.
* Văn Như Bích - Võ Hoàng Khang. Slide bài giảng Phân tích và thiết kế hệ thống thông tin.
* Tài liệu tham khảo tập hợp từ nhiều đồ án và nhiều trang website.