

ĐẠI HỌC KINH TẾ QUỐC DÂN
TRƯỜNG CÔNG NGHỆ



BÁO CÁO ĐỀ ÁN

***Đề tài: Xây dựng phần mềm quản lý bán hàng quán cà phê
bằng C#***

Sinh viên thực hiện: Hoàng Phan Khánh Huyền

Mã sinh viên: 11222875

Giảng viên hướng dẫn: Nguyễn Phương Nam

Hà Nội, tháng 4 năm 2025

MỤC LỤC

LỜI NÓI ĐẦU.....	2
CHƯƠNG 1: TỔNG QUAN VỀ ĐỀ TÀI.....	3
1.1. Tổng quan về tình hình nghiên cứu.....	3
1.2. Lý do lựa chọn đề tài.....	4
1.3. Mục tiêu đề tài.....	4
1.4. Nhiệm vụ nghiên cứu.....	4
1.5. Đối tượng và khách thể nghiên cứu.....	5
1.6. Phương tiện sử dụng.....	5
1.7. Phạm vi đề tài.....	5
1.8. Đóng góp của đề tài.....	5
CHƯƠNG 2. CƠ SỞ LÝ THUYẾT CỦA ĐỀ TÀI.....	6
2.1. Mô hình 3 lớp (Three-Layer).....	6
2.2. Giao tiếp giữa các thành phần trong hệ thống.....	8
2.3. Công nghệ và công cụ sử dụng.....	9
CHƯƠNG 3. PHÂN TÍCH YÊU CẦU HỆ THỐNG.....	14
3.1. Yêu cầu chức năng.....	14
3.2. Yêu cầu phi chức năng.....	14
3.3. Quy trình nghiệp vụ.....	15
CHƯƠNG 4. THIẾT KẾ HỆ THỐNG.....	16
4.1. Biểu đồ Use Case.....	16
4.2. Biểu đồ hoạt động.....	18
4.3. Thiết kế cơ sở dữ liệu.....	20
4.4. Thiết kế giao diện.....	22
CHƯƠNG 5. PHÁT TRIỂN HỆ THỐNG.....	28
5.1. Cấu trúc thư mục.....	28
5.2. Sử dụng mô hình 3 lớp cho dự án.....	29
5.3. Mã nguồn của phần mềm.....	30
CHƯƠNG 6. KẾT LUẬN.....	35
6.1. Kết quả đạt được.....	35
6.2. Các hạn chế.....	35
6.3. Hướng phát triển.....	35
TÀI LIỆU THAM KHẢO.....	36

LỜI NÓI ĐẦU

Song hành với sự phát triển mạnh mẽ của khoa học và công nghệ, mọi lĩnh vực trong đời sống đều có sự góp mặt của một hoặc nhiều thành tựu nghiên cứu. Công nghệ thông tin đang dần trở thành một phần không thể thiếu trong cuộc sống hiện đại. Tại Việt Nam, chuyển đổi số đã đến “từng ngõ, từng nhà, từng người”. Doanh thu lĩnh vực công nghệ thông tin (kinh tế số ICT) ước đạt 1.928.311 tỷ đồng, tăng trưởng 26% so với cùng kỳ năm 2023. Ứng dụng công nghệ vào quy trình vận hành của một doanh nghiệp là sự cấp thiết. Đây không chỉ là một lĩnh vực đơn thuần mà còn là động lực to lớn đang thúc đẩy và nâng tầm nền kinh tế và tri thức của một đất nước.

Người tiêu dùng luôn mong muốn có được trải nghiệm tốt nhất nhằm thỏa mãn nhu cầu cá nhân. Vậy nên, thái độ làm việc chuyên nghiệp, sự tận tình, nhanh nhẹn và minh bạch chính là mấu chốt phải có ở một công ty hay doanh nghiệp nằm trong ngành dịch vụ. Cụ thể hơn, để giải đáp cho sự tò mò cá nhân, em đã quyết định lựa chọn chủ đề “Xây dựng phần mềm quản lý bán hàng quán cà phê bằng C#” cho đề án chuyên ngành Khoa học máy tính. Việc áp dụng phần mềm vào quá trình kinh doanh của một quán cà phê góp phần nâng cao trải nghiệm sử dụng dịch vụ của khách hàng đối với quán cà phê, làm tăng tính chuyên nghiệp và chính xác trong khâu xử lý đơn hàng. Không thể chắc chắn rằng nó sẽ cải thiện được mọi khuyết điểm của quy trình nghiệp vụ truyền thống từ xưa nay, tuy nhiên phần mềm sẽ phần nào xóa bỏ những nhược điểm đang tồn tại trong quy trình vận hành cũ đó.

Cuối cùng, em xin gửi lời cảm ơn đến thầy Nguyễn Phương Nam đã chỉ dẫn em và đưa ra những lời khuyên hữu ích trong quá trình thực hiện đề án chuyên ngành! Mong thầy sẽ góp ý và bổ sung thêm để sản phẩm này được hoàn thiện hơn!

CHƯƠNG 1: TỔNG QUAN VỀ ĐỀ TÀI

1.1. Tổng quan về tình hình nghiên cứu

1.1.1. Kinh doanh thương mại là gì?

Kinh doanh thương mại là hoạt động mua bán, trao đổi hàng hóa và dịch vụ nhằm mục đích tạo ra lợi nhuận. Đây là quá trình liên tục của việc tìm kiếm cơ hội, đánh giá rủi ro, huy động nguồn lực và đưa ra quyết định để tạo ra giá trị cho khách hàng và thu về lợi nhuận cho doanh nghiệp. Trong nền kinh tế hiện đại, kinh doanh thương mại không chỉ đơn thuần là việc mua đi bán lại, mà còn bao gồm nhiều hoạt động phức tạp như nghiên cứu thị trường, phát triển sản phẩm, xây dựng thương hiệu, quản lý tài chính và nhân sự.

Theo quan điểm hiện đại, kinh doanh thương mại bền vững cần cân bằng ba yếu tố cốt lõi: lợi nhuận kinh tế, trách nhiệm xã hội và bảo vệ môi trường. Đặc biệt trong thời đại số hóa, công nghệ thông tin đóng vai trò then chốt trong việc tối ưu hóa quy trình kinh doanh, nâng cao hiệu quả và mở rộng phạm vi hoạt động.

1.1.2. Kinh doanh quán cà phê

Kinh doanh quán cà phê là một mô hình kinh doanh dịch vụ phổ biến tại Việt Nam và trên toàn thế giới. Mô hình này không chỉ đơn thuần cung cấp đồ uống mà còn tạo ra không gian giao lưu, làm việc hoặc thư giãn cho khách hàng. Theo báo cáo của Mordor Intelligence, quy mô thị trường cà phê Việt Nam ước tính đạt 511,03 triệu USD vào năm 2024 và dự kiến sẽ đạt 763,46 triệu USD vào năm 2029, tăng trưởng với tốc độ CAGR là 8,13% trong giai đoạn dự báo (2024-2029), cho thấy tiềm năng phát triển mạnh mẽ của ngành này.

Tuy nhiên, kinh doanh quán cà phê cũng đối mặt với nhiều thách thức như cạnh tranh gay gắt, chi phí vận hành cao và yêu cầu ngày càng khắt khe từ phía người tiêu dùng. Việc quản lý hiệu quả quán cà phê đòi hỏi chủ doanh nghiệp phải theo dõi và kiểm soát nhiều yếu tố như hàng tồn kho, doanh thu, chi phí, lịch làm việc của nhân viên và chất lượng dịch vụ.

Trong bối cảnh chuyển đổi số, các quán cà phê tiên tiến đã và đang áp dụng các giải pháp công nghệ thông tin vào quản lý. Việc áp dụng các giải pháp công nghệ phần mềm trong quản lý doanh nghiệp giúp tối ưu hóa vận hành và giảm chi phí.

Điều này cho thấy tầm quan trọng của việc ứng dụng công nghệ thông tin vào quản lý quán cà phê trong thời đại hiện nay.

1.2. Lý do lựa chọn đề tài

Sự ra đời của công nghệ đòi hỏi các công việc trong cuộc sống cần được số hoá để chạy đua với xu hướng và phát triển đi lên. Tại Việt Nam, số lượng các quán cà phê ngày càng tăng, muốn cạnh tranh được với các thương hiệu khác, họ cần một hệ thống hỗ trợ tốt trong quy trình vận hành và quản lý. Một phần mềm quản lý quán cà phê hiệu quả sẽ giúp chủ quán tối ưu hóa các hoạt động kinh doanh, tiết kiệm thời gian, giảm thiểu sai sót và nâng cao trải nghiệm của khách hàng.

1.3. Mục tiêu đề tài

Phần mềm hỗ trợ trong việc quản lý các nghiệp vụ: tạo hóa đơn, quản lý tài khoản nhân viên, quản lý các món có trong thực đơn, thống kê doanh thu,... nhằm đáp ứng nhu cầu quản lý hiệu quả của các chủ doanh nghiệp trong lĩnh vực này. Cụ thể, đề tài hướng đến các mục tiêu sau:

- Thiết kế và xây dựng một hệ thống phần mềm có giao diện thân thiện, dễ sử dụng, phù hợp với người dùng không chuyên về công nghệ thông tin.
- Phát triển các tính năng cốt lõi giúp tự động hóa các quy trình nghiệp vụ, giảm thiểu sai sót và nâng cao hiệu quả quản lý.
- Giúp bản thân có thêm kỹ năng và tư duy thông qua quá trình xây dựng phần mềm.

1.4. Nhiệm vụ nghiên cứu

Với quyết định lựa chọn đề tài này, phần mềm ra đời đi kèm với sứ mệnh cao cả đó là sự thuận tiện cho người sử dụng cũng như các chủ quán cà phê. Thông qua những tính năng đã được tích hợp, chủ quán có thể xem xét tổng quát tình hình kinh doanh và quản lý chất lượng của quán.

Phần mềm còn đem lại nhiều lợi ích khác. Nhưng điều quan trọng là gia tăng doanh thu, cải thiện nguồn vốn vào ra cho quán, nâng cao chất lượng sản phẩm và dịch vụ đối với khách hàng.

1.5. Đối tượng và khách thể nghiên cứu

- Đối tượng nghiên cứu: Phần mềm quản lý bán hàng quán cà phê
- Khách thể nghiên cứu:
 - + Chủ quán cà phê
 - + Nhân viên quản lý
 - + Nhân viên thu ngân

1.6. Phương tiện sử dụng

- Microsoft Visual Studio 2022 - môi trường phát triển tích hợp (IDE) cho C#
- SQL Server Management Studio - công cụ quản lý cơ sở dữ liệu
- Draw.io - thiết kế giao diện và luồng công việc

1.7. Phạm vi đề tài

- Phạm vi nội dung: Đề tài tập trung xoay quanh các quy trình quản lý cốt lõi của kinh doanh quán cà phê: quản lý đơn hàng, thống kê doanh thu, quản lý thực đơn.
- Phạm vi không gian: Phần mềm được phát triển để hướng tới các quán cà phê nhỏ lẻ tại Việt Nam.
- Phạm vi thời gian: Quá trình xây dựng phần mềm được thực hiện trong vòng 4 tháng, từ tháng 1 năm 2025 đến tháng 4 năm 2025.

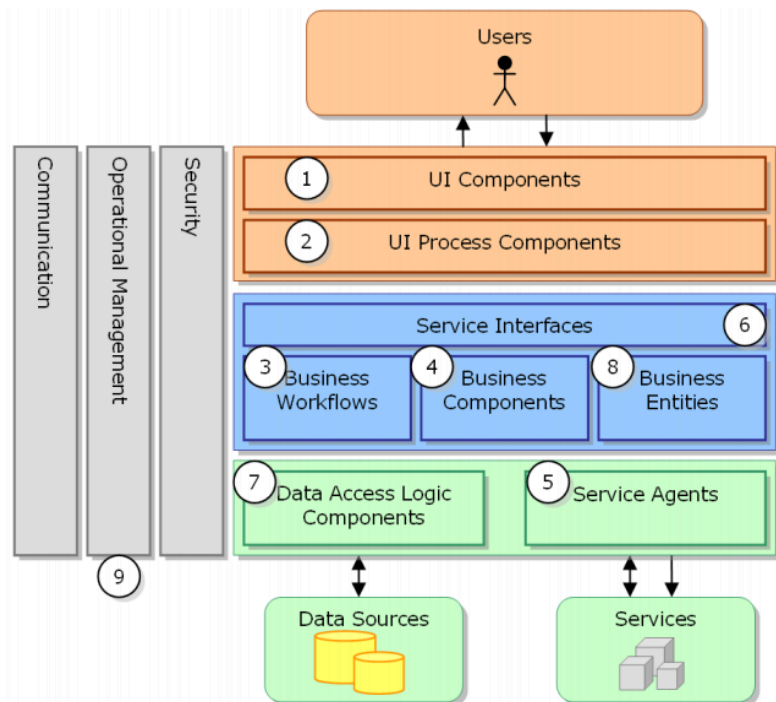
1.8. Đóng góp của đề tài

- Cung cấp một giải pháp phần mềm hoàn chỉnh giúp các chủ quán cà phê tối ưu hóa quy trình quản lý, nâng cao hiệu quả kinh doanh.
- Giảm thiểu thời gian và công sức cho các hoạt động quản lý thủ công, giúp chủ doanh nghiệp tập trung vào việc cải thiện chất lượng dịch vụ và phát triển kinh doanh.
- Cung cấp các báo cáo phân tích chi tiết, giúp chủ doanh nghiệp đưa ra quyết định kinh doanh dựa trên dữ liệu.

CHƯƠNG 2. CƠ SỞ LÝ THUYẾT CỦA ĐỀ TÀI

2.1. Mô hình 3 lớp (Three-Layer)

Mô hình 3 lớp được ra đời để phân chia các thành phần trong hệ thống theo chức năng của chúng, cụ thể là chúng ta sẽ chia chúng làm 3 lớp khác nhau:



2.1.1. Presentation Layer: Lớp này có nhiệm vụ chính giao tiếp với người dùng. Nó gồm các thành phần giao diện 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).

Lớp Presentation bao gồm 2 thành phần:

- 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:
 - + Hiển thị màn hình tra cứu ID
 - + Hiển thị màn hình thông tin chi tiết khách hàng tương ứng
 - + Hiển thị màn hình liên lạc với khách hàng.

Đây là lớp cao nhất của tầng kiến trúc, cũng là tầng quan trọng nhất bởi vì những gì người dùng nhìn thấy có ảnh hưởng đến số lượng người dùng phần mềm đó. Vì vậy, giao diện người dùng cần có một thiết kế tốt.

2.1.2. Business Logic Layer: Lớp này phân ra làm 2 nhiệm vụ:

- + Đáp ứng các yêu cầu thao tác dữ liệu của lớp Presentation, xử lý chính nguồn dữ liệu từ lớp Presentation trước khi truyền xuống lớp Data Access và lưu xuống hệ quản trị cơ sở dữ liệu.
- + Kiểm tra các ràng buộc, tính toàn vẹn và hợp lệ của dữ liệu, thực hiện tính toán và xử lý các yêu cầu nghiệp vụ, trước khi trả kết quả về lớp Presentation.

Lớp Business Logic bao gồm 4 thành phần:

- Service Interface: là thành phần giao diện lập trình mà lớp này cung cấp cho lớp Presentation sử dụng.
- Business 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.

Ví dụ: Thực hiện mua một đơn hàng trên Shopee qua nhiều bước: nhấn mua nếu còn hàng còn trong kho, áp dụng mã giảm giá (nếu có), tính toán tổng chi phí, cho phép chọn phương thức thanh toán và lựa chọn tốc độ giao hàng.

- Business 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.

Ví dụ: Ở ví dụ phía trên, bạn sẽ cần một Business Component để kiểm tra mặt hàng có còn hàng không? Hay một component để tính tổng chi phí phải trả,...

- Business 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.

2.1.3. Data Access Layer: Lớp này có chức năng giao tiếp với hệ quản trị cơ sở dữ liệu như thực hiện các công việc liên quan đến lưu trữ và truy vấn dữ liệu (tìm kiếm, thêm, xóa, sửa,...).

Lớp Data Access bao gồm 2 thành phần:

- Data Access Logic Components: chịu trách nhiệm chính cho việc 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 chúng ta gọi và tương tác với các dịch vụ từ bên ngoài một cách dễ dàng và đơn giản.

2.1.4. Ưu điểm

- Dễ dàng cập nhật và thay đổi giao diện người dùng
- Việc bảo trì một thành phần chỉ ảnh hưởng đến phần nhỏ của chương trình mà không tác động đến toàn bộ
- Nhờ sự tương tự mà có thể sử dụng lại mã nguồn ta đã viết để xây dựng một tính năng hoặc một ứng dụng mới

2.1.5. Nhược điểm

- Phức tạp hơn kiến trúc client - server thông thường
- Khó xây dựng hơn dẫn đến tốn nhiều thời gian hơn
- Người dùng cần có nền tảng kiến thức về các khái niệm hướng đối tượng

2.2. Giao tiếp giữa các thành phần trong hệ thống

2.2.1. Tương tác giữa Presentation Layer và Business Logic Layer

Presentation Layer đóng vai trò trung gian giữa người dùng và hệ thống. Khi người dùng thực hiện thao tác trên giao diện (ví dụ như nhập dữ liệu, nhấn nút), lớp này sẽ gửi yêu cầu đến Business Logic Layer để xử lý dữ liệu trước khi thực thi chúng trong cơ sở dữ liệu. Quy trình được mô tả chi tiết hơn như sau:

- Dữ liệu được thu thập từ các thành phần UI như TextBox, ComboBox, DataGridView, ... khi người dùng nhập dữ liệu trên giao diện.
- Presentation Layer gửi yêu cầu đến Business Logic Layer để gọi các phương thức tương ứng trong Business Logic, truyền dữ liệu đầu vào.
- Business Logic kiểm tra tính hợp lệ của dữ liệu đầu vào (ví dụ: không để trống, định dạng hợp lệ) và xử lý logic nghiệp vụ đối với dữ liệu (ví dụ: tính tổng giá trị đơn hàng, kiểm tra số lượng tồn kho).
- Trả lại kết quả về giao diện:

- + Nếu dữ liệu hợp lệ, gửi yêu cầu tiếp tục xuống Data Access Layer để làm việc với cơ sở dữ liệu.
- + Nếu xảy ra lỗi (dữ liệu sai, thiếu thông tin), hiển thị thông báo lỗi trên giao diện người dùng.
- Hiển thị kết quả trả về lên giao diện

2.2.2. Tương tác giữa Business Logic Layer và Data Access Layer

Business Logic Layer có nhiệm vụ kiểm tra dữ liệu và gọi phương thức từ Data Access Layer để thực thi các thao tác (truy xuất, thêm, sửa, xóa) trên dữ liệu trong cơ sở dữ liệu.

- Business Logic Layer sau khi kiểm tra dữ liệu hợp lệ, chúng bắt đầu gọi phương thức từ Data Access Layer để tương tác với dữ liệu.
- Data Access Layer kết nối đến SQL Server và tiến hành các truy vấn trên cơ sở dữ liệu. Có thể sử dụng ADO.NET hoặc Entity Framework để thực thi. Sau đó, dữ liệu được trả về hoặc trả về trạng thái xử lý dữ liệu cho Business Logic Layer.
- Business Logic Layer xử lý kết quả trả về từ Data Access Layer:
 - + Nếu thành công, dữ liệu được hiển thị lên giao diện của người dùng.
 - + Nếu thất bại, hiển thị thông báo lỗi hoặc ghi log để xử lý.

2.3. Công nghệ và công cụ sử dụng

2.3.1. Ngôn ngữ lập trình C#

C# (C-Sharp) là một ngôn ngữ lập trình, được Microsoft công bố lần đầu vào tháng 6 năm 2000 và chính thức ra mắt phiên bản đầu tiên vào tháng 1 năm 2002. Ngôn ngữ này được thiết kế bởi Anders Hejlsberg, một kỹ sư phần mềm tài năng người Đan Mạch, người từng tạo ra Turbo Pascal và Delphi. Với nguồn cảm hứng từ C++, Java, Python và Visual Basic, C# mang đến sự kết hợp hoàn hảo giữa sức mạnh, tính đơn giản và an toàn. Sự hậu thuẫn từ Microsoft đã giúp C# nhanh chóng trở thành một trong những ngôn ngữ lập trình hàng đầu thế giới.

Những lý do nên sử dụng C# để phát triển ứng dụng quản lý bán hàng:

- Khả năng chạy mượt mà trên nhiều nền tảng:
Nhờ vào .NET Framework và .NET Core, C# cho phép phát triển các ứng dụng

quản lý bán hàng có thể chạy ổn định trên nhiều hệ điều hành như Windows, macOS, Linux và cả các thiết bị di động. Điều này giúp các doanh nghiệp dễ dàng mở rộng và triển khai ứng dụng trên phạm vi rộng lớn.

- Tương thích hoàn hảo với hệ sinh thái Microsoft:
C# hoạt động tối ưu cùng các công cụ như SQL Server, Visual Studio, và Windows Forms, tạo điều kiện thuận lợi cho việc phát triển, kiểm thử và bảo trì ứng dụng. Các công nghệ này không chỉ giúp đẩy nhanh quá trình triển khai mà còn đảm bảo chất lượng sản phẩm ở mức cao nhất.
- Lập trình hướng đối tượng - Tối ưu hóa tổ chức mã nguồn:
C# còn được trang bị tính năng hướng đối tượng (OOP), giúp lập trình viên dễ dàng tổ chức và quản lý mã nguồn một cách rõ ràng. Điều này đặc biệt hữu ích khi phát triển các module nghiệp vụ phức tạp, giúp việc bảo trì và nâng cấp ứng dụng diễn ra nhanh chóng, hiệu quả.
- Hệ sinh thái phong phú và thư viện đa dạng:
Với sự hỗ trợ đến từ một hệ sinh thái rộng lớn bao gồm vô số thư viện và framework mạnh mẽ, thời gian phát triển được giảm bớt mà vẫn đảm bảo hiệu suất và tính ổn định của hệ thống. Ví dụ, các thư viện như Entity Framework hay SignalR cung cấp sẵn các giải pháp hữu ích để xử lý dữ liệu và giao tiếp thời gian thực.
- Tài nguyên học tập dồi dào:
Nhờ sự phổ biến của C#, bạn có thể dễ dàng tìm kiếm các tài liệu hướng dẫn, khóa học trực tuyến và sự hỗ trợ từ các cộng đồng lập trình viên. Từ đó, công cuộc giải quyết các vấn đề kỹ thuật và tiếp cận những xu hướng phát triển mới trở nên thuận tiện hơn.
- Hiệu suất cao và bảo trì dễ dàng
Các ứng dụng viết bằng C# thường đạt hiệu suất xử lý vượt trội nhờ các tối ưu trong ngôn ngữ và công cụ hỗ trợ. Đồng thời, với cấu trúc rõ ràng và tính an toàn, C# giúp lập trình viên thực hiện bảo trì và nâng cấp phần mềm một cách nhanh chóng và bảo mật.

C# không chỉ là một ngôn ngữ lập trình hiện đại mà còn là lựa chọn lý tưởng để phát triển các ứng dụng quản lý bán hàng. Với khả năng đa nền tảng, tích hợp hoàn hảo với hệ sinh thái Microsoft và hàng loạt lợi ích thực tế khác, C# không chỉ tối ưu

hóa hiệu suất mà còn mang lại sự linh hoạt và tính ổn định cho các dự án phần mềm. Đây chính là công cụ mạnh mẽ giúp bạn tiến xa hơn trong thế giới lập trình.

2.3.2. Môi trường phát triển Visual Studio

Ra đời từ năm 1997, Microsoft Visual Studio đã khẳng định vị thế là một môi trường phát triển tích hợp (IDE) toàn diện và mạnh mẽ, được hàng triệu lập trình viên trên toàn thế giới tin dùng. Được mệnh danh là “trình soạn thảo mã phổ biến nhất thế giới,” Visual Studio không chỉ là công cụ viết mã, biên dịch và gỡ lỗi, mà còn cung cấp bộ công cụ đa dạng giúp xây dựng các phần mềm hiện đại từ ứng dụng desktop, web, di động cho đến các dịch vụ đám mây.

Giao diện thân thiện cùng khả năng tùy biến cao là điểm sáng lớn của Visual Studio, giúp các lập trình viên dễ dàng quản lý dự án, kiểm soát mã nguồn và sửa lỗi nhanh chóng thông qua hệ thống gỡ lỗi tích hợp. Một trong những tính năng nổi bật là IntelliSense, công nghệ tự động hoàn thành mã thông minh dựa trên ngữ cảnh, cho phép lập trình viên làm việc nhanh chóng và chính xác hơn. Visual Studio cũng hỗ trợ nhiều ngôn ngữ lập trình phổ biến như C++, C#, Python, Visual Basic và JavaScript, giúp người dùng tự do phát triển ứng dụng theo nhu cầu dự án.

Điểm đặc biệt của Visual Studio nằm ở khả năng hỗ trợ phát triển đa nền tảng. Lập trình viên có thể sử dụng các công cụ như Xamarin hoặc .NET Core để xây dựng ứng dụng cho Windows, macOS, Linux, Android và iOS. Điều này không chỉ mở rộng phạm vi triển khai phần mềm mà còn giúp các dự án tiếp cận nhiều đối tượng người dùng hơn. Các tính năng như tích hợp Git và Azure DevOps cũng tạo điều kiện thuận lợi cho việc làm việc nhóm, từ quản lý mã nguồn đến phối hợp thực hiện các dự án lớn.

Không chỉ dừng lại ở việc hỗ trợ phát triển ứng dụng, Visual Studio còn tích hợp nhiều công cụ hữu ích khác như thiết kế giao diện, xây dựng cơ sở dữ liệu, và tiện ích mở rộng giúp cá nhân hóa môi trường làm việc theo từng yêu cầu cụ thể. Nhờ những tính năng vượt trội này, Visual Studio đã trở thành lựa chọn không thể thiếu đối với các lập trình viên chuyên nghiệp và những người mới bắt đầu bước chân vào lĩnh vực lập trình.

2.3.3. Hệ quản trị cơ sở dữ liệu SQL Server

SQL Server là một hệ quản trị cơ sở dữ liệu quan hệ (RDBMS) do Microsoft phát triển, từ năm 1988 cho đến nay, SQL Server đã trở thành một trong những

phương tiện quản lý dữ liệu phổ biến và được ưa chuộng trong các doanh nghiệp với khả năng xử lý dữ liệu lớn và hiệu suất cao.

Các chức năng chính của SQL Server bao gồm:

- Quản lý dữ liệu: Cho phép tạo, sửa đổi và xóa các bảng dữ liệu, chỉ mục, cũng như thiết lập các mối quan hệ giữa các bảng nhằm đảm bảo cấu trúc dữ liệu được tổ chức hợp lý.
- Quản lý giao dịch: Hỗ trợ các giao dịch theo tiêu chuẩn ACID, đảm bảo tính nhất quán, cách ly và độ bền của dữ liệu trong quá trình thực hiện các thao tác.
- Bảo mật dữ liệu: Cung cấp các cơ chế kiểm soát truy cập, mã hóa và xác thực người dùng để bảo vệ dữ liệu khỏi truy cập trái phép.
- Sao lưu và phục hồi: Cho phép thực hiện sao lưu dữ liệu định kỳ và phục hồi nhanh chóng khi gặp sự cố, bảo vệ dữ liệu khỏi mất mát.
- Phân tích và tạo báo cáo: Tích hợp các công cụ như SQL Server Reporting Services (SSRS) và SQL Server Analysis Services (SSAS) giúp phân tích dữ liệu, tạo báo cáo chi tiết và trực quan hóa thông tin kinh doanh.

SQL Server là một hệ thống quản trị cơ sở dữ liệu mạnh mẽ, cung cấp đầy đủ các công cụ từ quản lý dữ liệu cơ bản đến phân tích và tạo báo cáo nâng cao. Với khả năng xử lý hiệu quả, tính bảo mật cao và khả năng tích hợp với các hệ thống hiện đại, SQL Server giúp lập trình viên và các doanh nghiệp dễ dàng xây dựng, duy trì và khai thác dữ liệu để mang lại giá trị kinh doanh tối ưu.

2.3.4. Crystal Report

Phần mềm Crystal Report được phát triển bởi Business Objects - một công ty phát triển phần mềm có trụ sở tại Pháp và là một phần của tập đoàn công ty SAP AG. Crystal Report cung cấp nhiều lựa chọn các công cụ để tạo báo cáo cho người dùng. Nguồn dữ liệu được đưa vào báo cáo là do người dùng lựa chọn, họ có thể sử dụng các định dạng dữ liệu khác nhau như: Microsoft Excel (.xls), SQL Server (.sql),... Khi cơ sở dữ liệu đã được kết nối đến để tạo báo cáo, ta có thể chỉnh sửa những dữ liệu được đưa lên báo cáo, chỉnh sửa đầu trang, chân trang, tên cột,...

Với một bộ công cụ tạo báo cáo mạnh mẽ như vậy, Crystal Report đã mang lại nhiều lợi ích thiết thực cho doanh nghiệp và các lập trình viên:

- Giao diện trực quan: Crystal Reports cung cấp giao diện thiết kế báo cáo thân thiện, cho phép người dùng dễ dàng kéo thả các thành phần, định dạng dữ liệu và xem trước báo cáo ngay trong quá trình tạo.
- Kết nối dữ liệu đa dạng: Phần mềm hỗ trợ kết nối với nhiều nguồn dữ liệu khác nhau như SQL Server, Oracle, Excel, Access, hay các file văn bản. Điều này giúp tích hợp dữ liệu từ nhiều hệ thống thành một báo cáo duy nhất.
- Tùy chỉnh cao: Với khả năng sử dụng công thức, biểu thức và các hàm tích hợp, người dùng có thể tùy chỉnh dữ liệu hiển thị theo yêu cầu cụ thể của doanh nghiệp.
- Xuất báo cáo đa định dạng: Crystal Reports cho phép xuất báo cáo ra nhiều định dạng khác nhau như PDF, Excel, Word, HTML... giúp việc chia sẻ thông tin trở nên linh hoạt và thuận tiện.
- Khả năng mở rộng: Crystal Reports có thể được tích hợp vào các ứng dụng khác, chẳng hạn như nhúng vào các phần mềm do Visual Studio phát triển, từ đó tăng cường khả năng báo cáo trong các giải pháp phần mềm toàn diện.

Trong môi trường kinh doanh hiện nay, việc thu thập, phân tích và trình bày dữ liệu một cách hiệu quả đóng vai trò quan trọng trong việc hỗ trợ các quyết định chiến lược. Crystal Reports, với khả năng mạnh mẽ trong việc tạo ra các báo cáo chi tiết và trực quan, đã trở thành công cụ không thể thiếu trong nhiều lĩnh vực khác nhau. Từ tài chính, kế toán đến quản lý nhân sự, bán hàng và marketing, Crystal Reports giúp doanh nghiệp chuyển hóa dữ liệu thô thành thông tin giá trị, hỗ trợ quá trình ra quyết định một cách chính xác và kịp thời. Sự linh hoạt trong thiết kế báo cáo, khả năng kết nối với nhiều nguồn dữ liệu và tính năng tùy chỉnh cao khiến Crystal Reports trở thành lựa chọn ưu tiên cho các doanh nghiệp mong muốn nâng cao hiệu quả quản lý và hoạt động kinh doanh.

CHƯƠNG 3. PHÂN TÍCH YÊU CẦU HỆ THỐNG

3.1. Yêu cầu chức năng

- Quản lý thực đơn:
 - + Thêm, sửa, xóa món
 - + Xem danh sách món
 - + Tìm món trong thực đơn
- Quản lý tài khoản:
 - + Thêm, sửa, xóa tài khoản
 - + Xem danh sách tài khoản
- Quản lý bàn ăn:
 - + Thêm, sửa, xóa bàn ăn
 - + Xem danh sách bàn ăn
- Quản lý danh mục:
 - + Thêm, sửa, xóa danh mục
 - + Xem danh sách danh mục
- Xem danh sách hóa đơn trong khoảng thời gian
- Tạo hoá đơn:
 - + Thêm, xóa món trong hóa đơn
 - + Thanh toán hóa đơn
 - + Chuyển hóa đơn sang bàn khác
- Cập nhật thông tin tài khoản cá nhân

3.2. Yêu cầu phi chức năng

- Hiệu năng:
 - + Hệ thống phản hồi nhanh chóng, không gây trễ.
 - + Xử lý được số lượng lớn đơn hàng cùng lúc.
- Bảo mật:
 - + Bảo vệ thông tin khách hàng, nhân viên và dữ liệu kinh doanh.
 - + Phân quyền truy cập hệ thống.

- Tính khả dụng:
 - + Giao diện người dùng thân thiện, dễ sử dụng.
 - + Hệ thống hoạt động ổn định, ít xảy ra lỗi.
 - + Hệ thống có thể chạy trên nhiều cấu hình máy tính khác nhau.
- Khả năng mở rộng:
 - + Hệ thống có thể dễ dàng thêm các chức năng mới.
- Bảo trì:
 - + Mã nguồn rõ ràng, dễ hiểu, dễ bảo trì.
 - + Có tài liệu hướng dẫn sử dụng và tài liệu kỹ thuật.
- Tính tương thích:
 - + Hệ thống tương thích với các hệ điều hành Windows phổ biến.

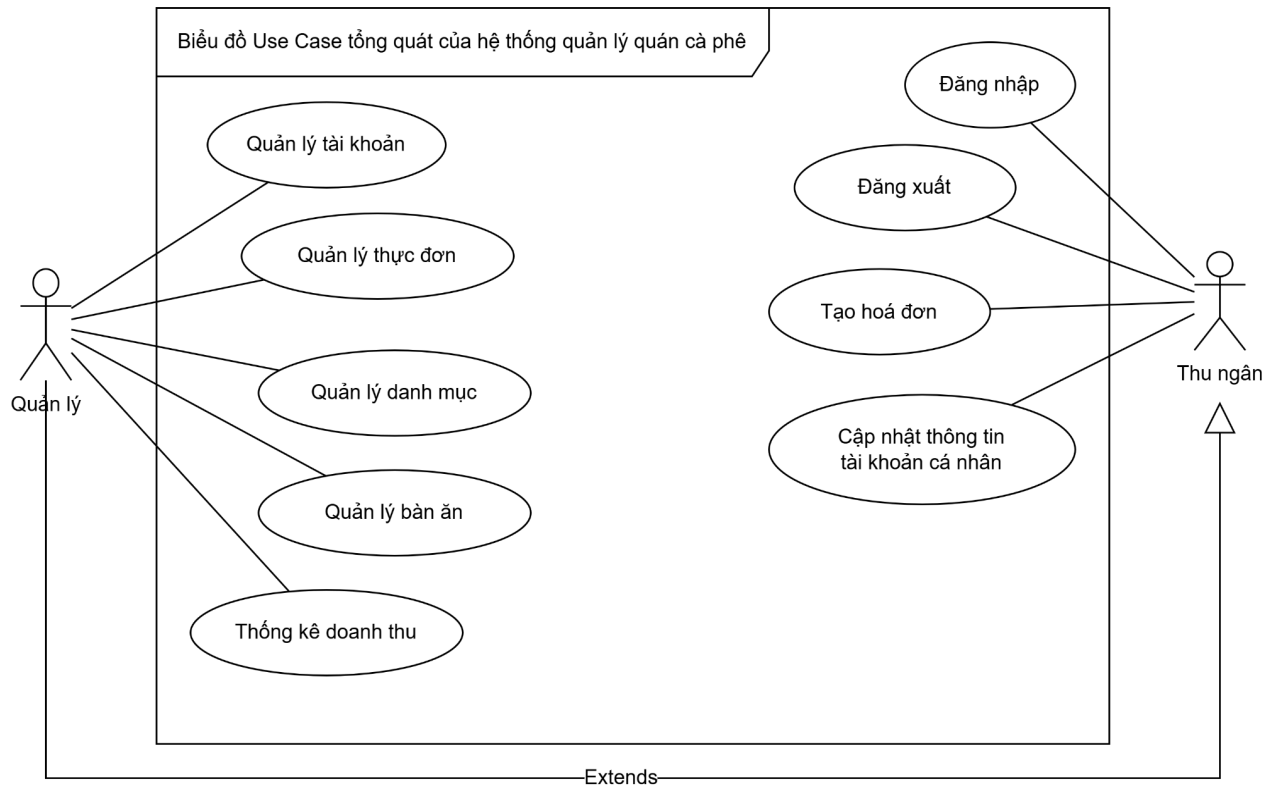
3.3. Quy trình nghiệp vụ

- Khách hàng đến quầy thu ngân để gọi món sau đó tìm chỗ ngồi.
- Khi khách hàng gọi món, nhân viên thu ngân thực thi lệnh trên phần mềm để thêm món vào hóa đơn, bấm chọn bàn của khách hàng trên hệ thống.
- Nếu trong thời gian ở quán, khách có nhu cầu chuyển đổi vị trí ngồi thì nhân viên cần cập nhật tại chức năng chuyển bàn.
- Khách hàng sử dụng xong dịch vụ và muốn rời đi thì cần tới quầy thu ngân để thanh toán hóa đơn, báo cho nhân viên vị trí bàn mà mình đã sử dụng. Sau đó, nhân viên sẽ thao tác thanh toán trên màn hình máy tính, nhập số phần trăm giảm giá (nếu có) và nhấn thanh toán để hiển thị số tiền khách phải trả. Bàn vừa được sử dụng sẽ chuyển trạng thái từ “Có người” sang “Trống”. Cuối cùng, trong danh sách hóa đơn, hóa đơn vừa thanh toán thành công sẽ được ghi vào trong cơ sở dữ liệu của hệ thống.
- Quản trị viên có quyền thêm, sửa, xóa, xem thông tin của thực đơn, tài khoản nhân viên và bàn ăn. Ngoài ra, họ có thể xem danh sách hóa đơn trong phần danh sách hóa đơn khi đăng nhập tài khoản quản trị viên.

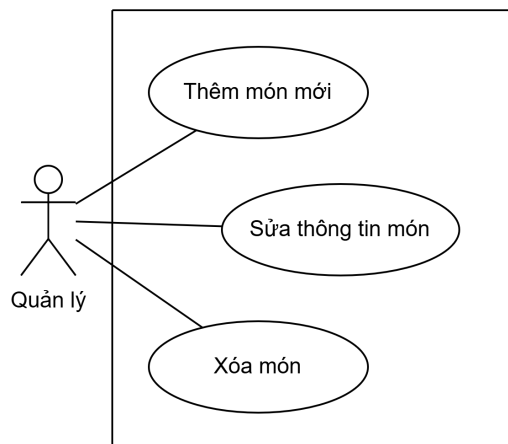
CHƯƠNG 4. THIẾT KẾ HỆ THỐNG

4.1. Biểu đồ Use Case

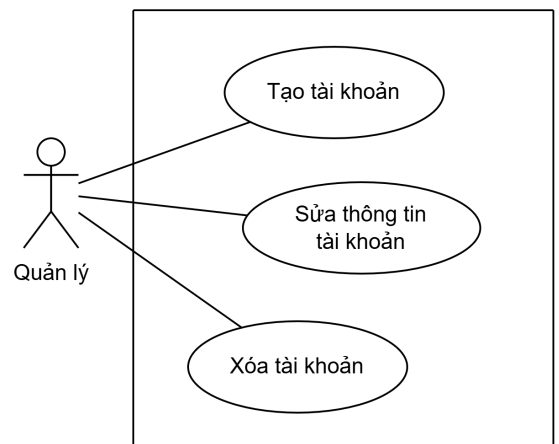
4.1.1. Biểu đồ Use Case tổng quát:



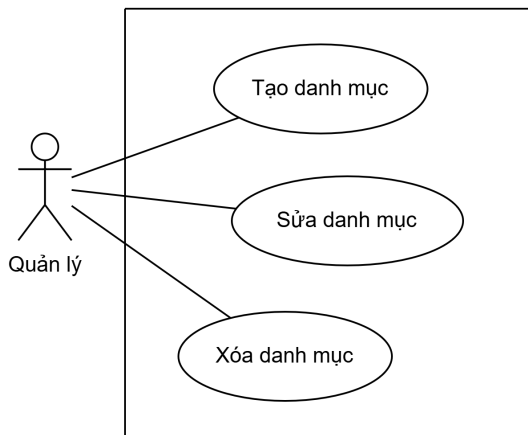
4.1.2. Các biểu đồ Use Case phân rã chức năng của hệ thống:



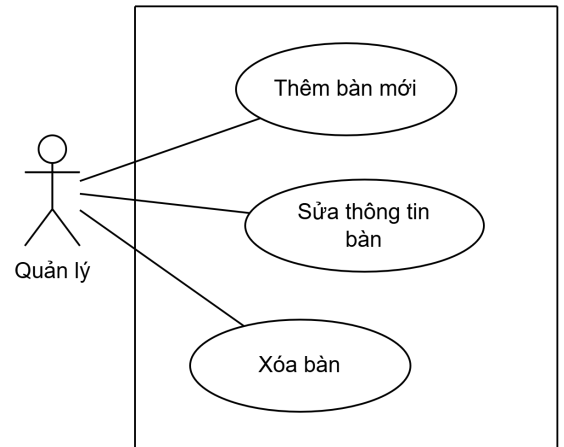
Biểu đồ Use Case phân rã chức năng quản lý thực đơn



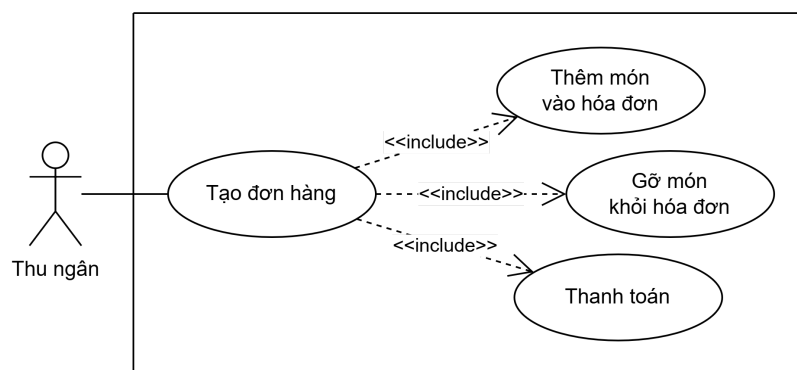
Biểu đồ Use Case phân rã chức năng quản lý tài khoản



Biểu đồ Use Case phân rã chức năng quản lý danh mục



Biểu đồ Use Case phân rã chức năng quản lý bàn ăn



Biểu đồ Use Case phân rã chức năng tạo đơn hàng

4.1.3. Đặc tả Use Case

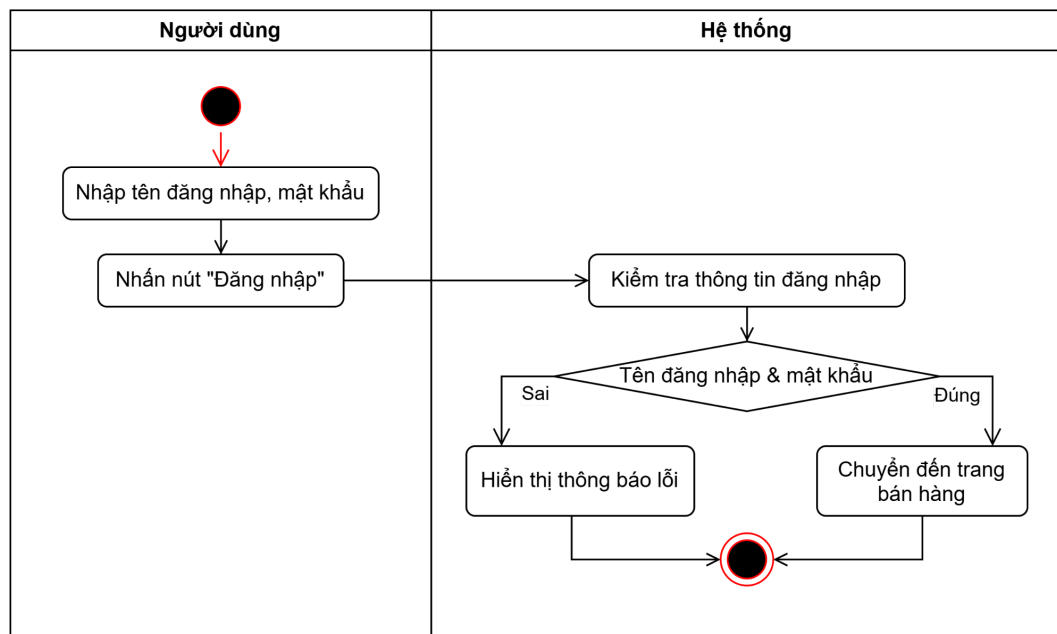
Phần mềm quản lý bao gồm các chức năng: bán hàng, quản lý thực đơn, quản lý bàn, quản lý tài khoản nhân viên, xem danh sách hóa đơn.

- **Bán hàng:** Chức năng chính của phần mềm, đảm nhiệm vai trò tạo hóa đơn khi khách hàng đặt món, bao gồm thông tin về bàn, danh sách món ăn, số lượng và giá tiền. Cho phép thực hiện thanh toán hóa đơn, tính toán tổng tiền, áp dụng giảm giá (nếu có) và chuyển trạng thái bàn từ “Có người” thành “Trống”.
- **Quản lý thực đơn:** Bao gồm thông tin của tất cả các món ăn/đồ uống theo danh mục (cho phép thao tác thêm thông tin, sửa thông tin, xóa thông tin, lưu thông tin), bao gồm tên món, giá, danh mục của món.
- **Quản lý bàn:** Lưu thông tin của các bàn gồm tên và trạng thái của bàn. Cho phép khách hàng có thể chuyển bàn và tự động cập nhật trạng thái bàn khi không còn ai sử dụng. Quản trị viên có thể thêm, sửa, xóa thông tin của bàn.

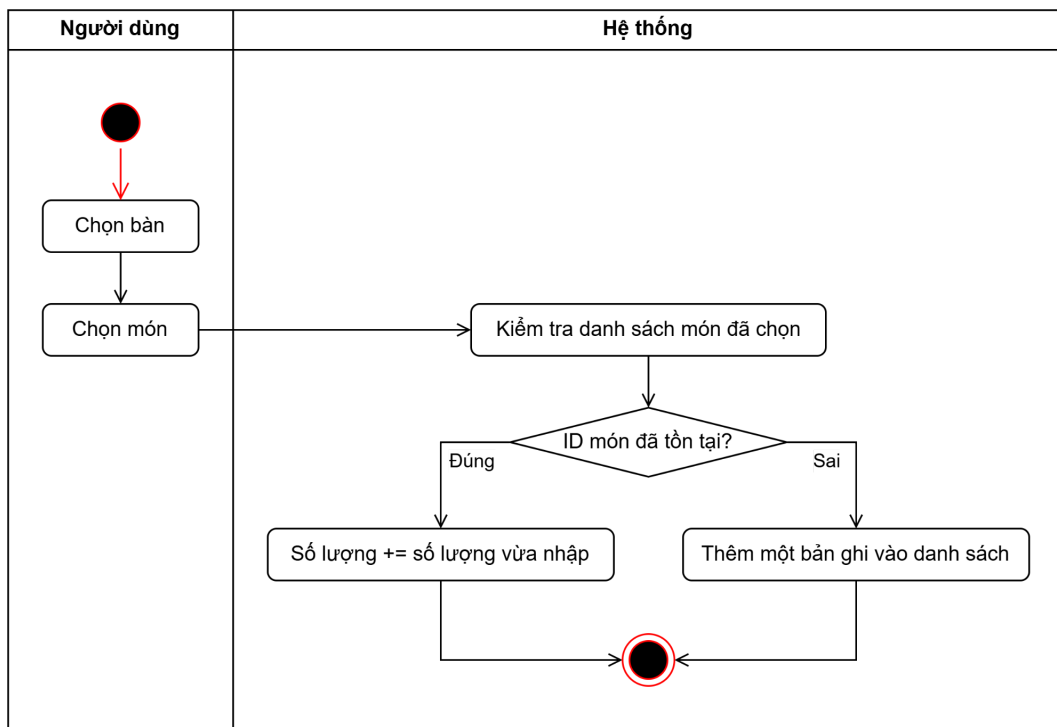
- Quản lý tài khoản nhân viên: Lưu trữ thông tin tài khoản của nhân viên bao gồm tên, tên đăng nhập, mật khẩu, chức vụ (cho phép mỗi cá nhân thao tác sửa thông tin, lưu thông tin).
- Xem danh sách hóa đơn: Hiển thị danh sách các hóa đơn đã thanh toán theo ngày, tháng, năm.

4.2. Biểu đồ hoạt động

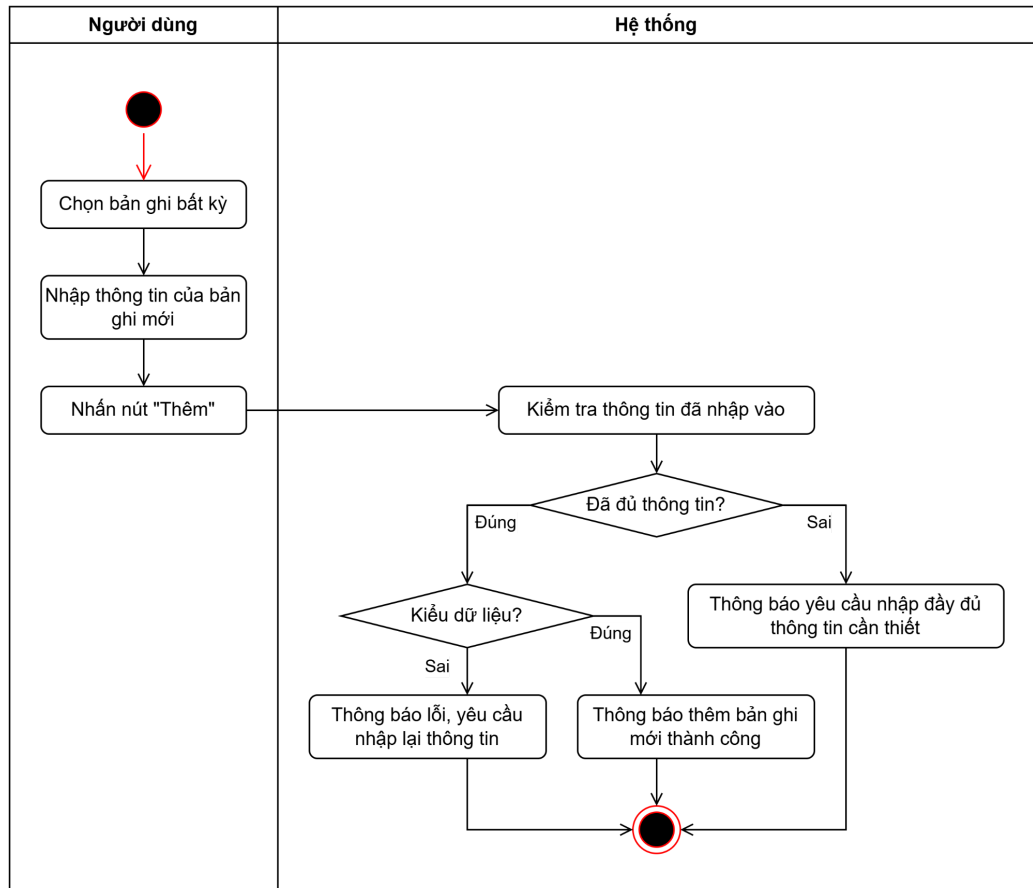
4.2.1. Chức năng đăng nhập



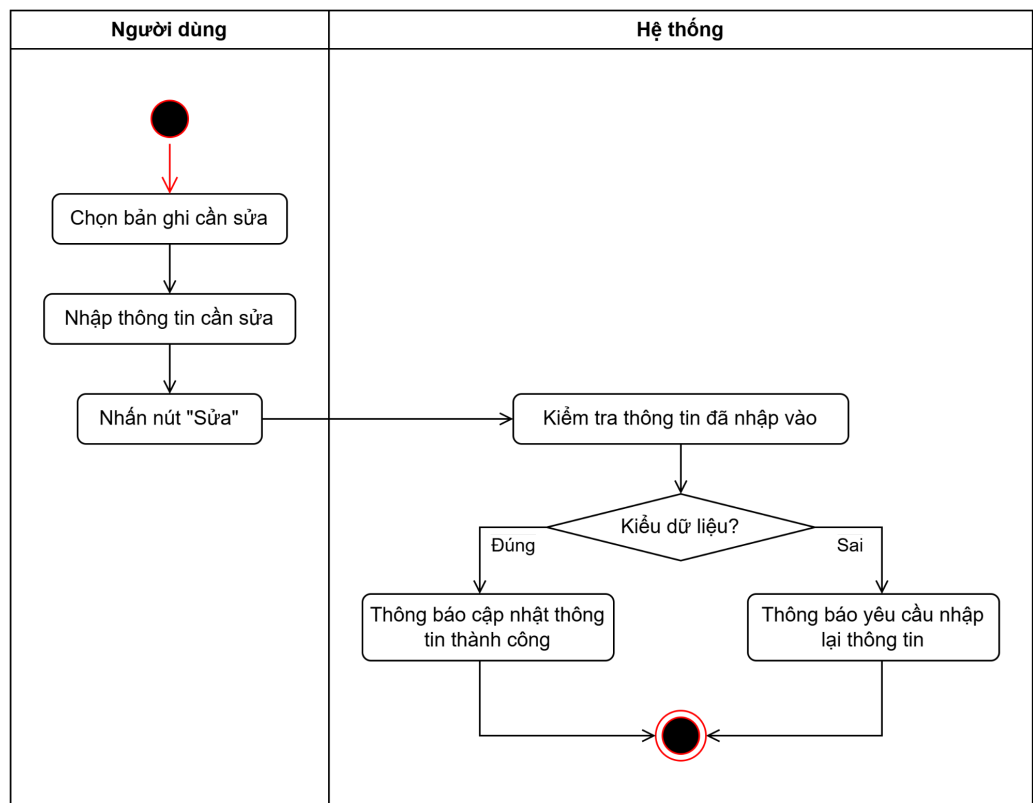
4.2.2. Chức năng thêm món vào hóa đơn



4.2.3. Chức năng thêm bản ghi mới

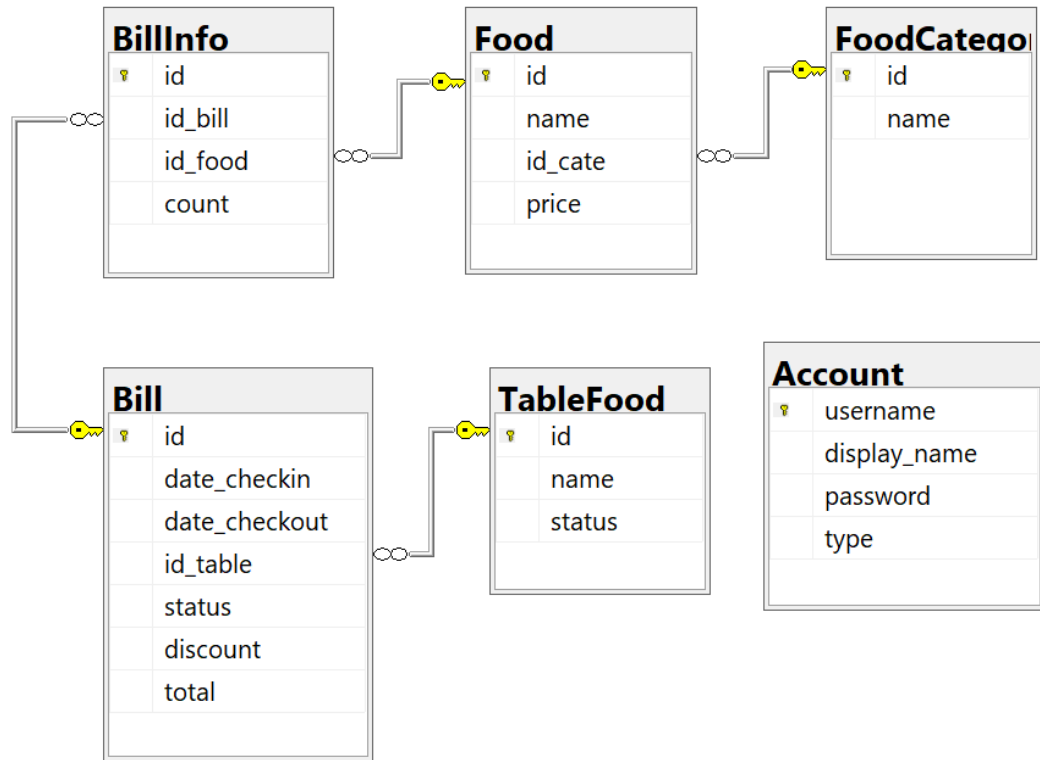


4.2.4. Chức năng sửa thông tin bản ghi



4.3. Thiết kế cơ sở dữ liệu

4.3.1. Mô hình cơ sở dữ liệu



- Account (Tài khoản người dùng)

STT	Tên thuộc tính	Kiểu dữ liệu	Khoá chính	Được phép để trống	Khoá ngoại
1	username	varchar(100)	X		
2	display_name	nvarchar(100)			
3	password	varchar(1000)			
4	type	int			

- Food (Món)

STT	Tên thuộc tính	Kiểu dữ liệu	Khoá chính	Được phép để trống	Khoá ngoại
1	id	int	X		
2	name	nvarchar(100)			

3	id_cate	int			X
4	price	int			

- FoodCategory (Danh mục món)

STT	Tên thuộc tính	Kiểu dữ liệu	Khoá chính	Được phép để trống	Khoá ngoại
1	id	int	X		
2	name	nvarchar(100)			

- Bill (Hoá đơn)

STT	Tên thuộc tính	Kiểu dữ liệu	Khoá chính	Được phép để trống	Khoá ngoại
1	id	int	X		
2	date_checkin	date			
3	date_checkout	date			
4	id_table	int			X
5	status	int			
6	discount	int			
7	total	int			

- BillInfo (Chi tiết hoá đơn)

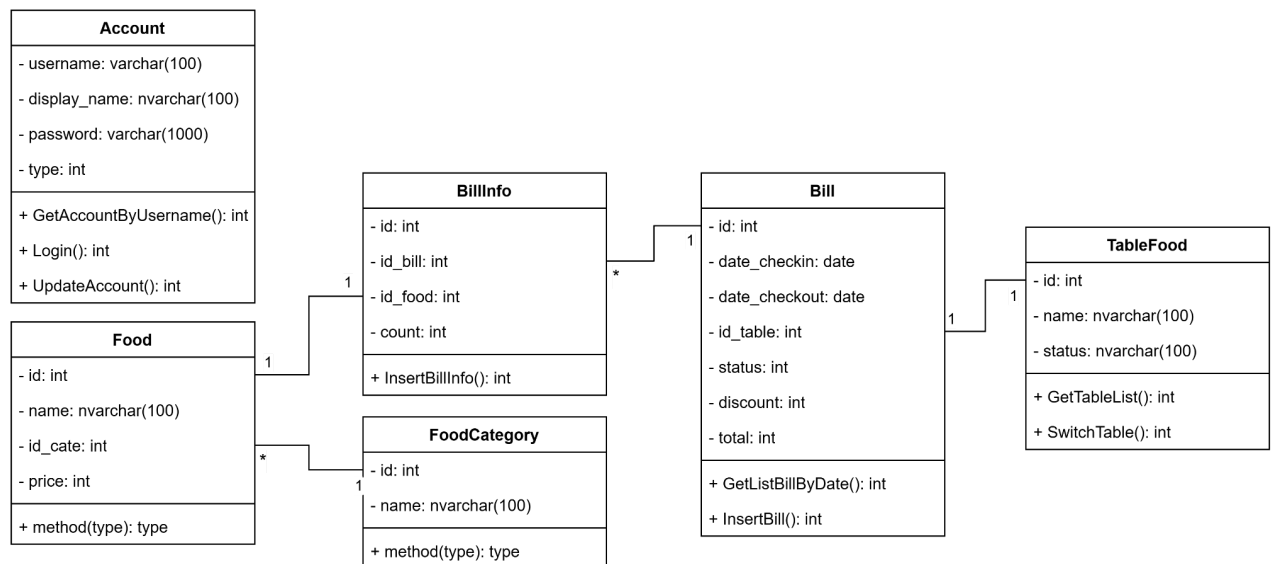
STT	Tên thuộc tính	Kiểu dữ liệu	Khoá chính	Được phép để trống	Khoá ngoại
1	id	int	X		
2	id_bill	int			X
3	id_food	int			X

4	count	int			
---	-------	-----	--	--	--

- TableFood (Bàn)

STT	Tên thuộc tính	Kiểu dữ liệu	Khoá chính	Được phép để trống	Khoá ngoại
1	id	int	X		
2	name	nvarchar(100)			
3	status	nvarchar(100)			

4.3.2. Mối quan hệ giữa các bảng



- Bảng BillInfo và Food liên kết thông qua id_food (mã món). Một bản ghi trong BillInfo tương ứng với một món mà khách hàng đã gọi.
- Bảng Food và FoodCategory liên kết thông qua id_cate (mã danh mục). Một danh mục món bao gồm nhiều món (đồ ăn, đồ uống) khác nhau.
- Bảng Bill và BillInfo liên kết thông qua id_bill (mã hoá đơn). Một hoá đơn gồm nhiều món khác nhau.
- Bảng Bill và TableFood liên kết thông qua id_table. Mỗi hoá đơn tương ứng với một bàn khách chọn.

4.4. Thiết kế giao diện

4.4.1. Giao diện đăng nhập

Khi chạy chương trình, giao diện đăng nhập hiện ra. Người dùng sử dụng tên đăng nhập và mật khẩu đã được cung cấp gõ vào TextBox tương ứng.

- Nếu thông tin đăng nhập chính xác, người dùng được chuyển đến giao diện bán hàng.
- Nếu nhập sai thông tin đăng nhập, sau khi ấn nút Đăng nhập thì thông báo “Sai tên đăng nhập hoặc mật khẩu!”.

Tên đăng nhập và mật khẩu của nhân viên được đặt theo tên chức vụ và kèm theo số thứ tự trong danh sách tài khoản đã đăng ký. Ví dụ: Trong cơ sở dữ liệu của chương trình đã có 2 tài khoản thu ngân với tên đăng nhập lần lượt là *staff1* và *staff2* thì khi tạo tài khoản thu ngân mới, tên đăng nhập của tài khoản đó sẽ là *staff3*.

Tài khoản của quản trị viên được cấp từ ban đầu như sau:

- Tên đăng nhập: *admin1*
- Mật khẩu: *admin123*

Tài khoản của một nhân viên thu ngân bất kì:

- Tên tài khoản: *staff1*
- Mật khẩu: *staff123*

Thao tác thực hiện:

- Tại khung TextBox bên trên, bên cạnh Label “Tên đăng nhập:”, người dùng nhập vào tên đăng nhập đã được cung cấp, chẳng hạn như trên hình là “admin1”.

- Tại khung TextBox ngay bên dưới, nhập mật khẩu tương ứng với tên tài khoản đã được cấp, ở đây ta nhập “admin123”.
- Nhấn nút “Đăng nhập” ở phía dưới cùng của cửa sổ Đăng nhập.

4.4.2. Giao diện bán hàng

Sau khi đăng nhập, giao diện bán hàng hiện ra với các chức năng: chọn bàn (khung bên trái), danh sách những món được gọi và chức năng thanh toán (khung bên phải). Tại đây, thu ngân sẽ chọn bàn khi có khách đến và thực hiện tạo hóa đơn cho bàn đã chọn, thực hiện chuyển bàn khi khách hàng có nhu cầu và số phần trăm giảm giá (nếu có). Cuối cùng là tiến hành thanh toán cho khách.

Tên món	Số lượng	Đơn giá	Thành tiền
Chanh dây đá xay	2	35000	70000
Trà ổi hồng	1	20000	20000

Trà:

Số lượng:

Bàn:

Tổng tiền: **90000**

Thao tác thực hiện:

- Click chọn bàn mà khách hàng yêu cầu ở bên trái.
- Ở bên phải, lựa chọn món mà khách hàng yêu cầu bằng cách nhấn vào ComboBox bên trái, phía dưới danh sách món đã chọn để chọn danh mục của món và nhấn tiếp vào ComboBox bên cạnh đó để chọn món.
- Thay đổi số lượng món được gọi tại NumericUpDown. Mặc định được đặt là 1.
- Nhấn nút “Thêm món” ở bên cạnh số lượng để thêm món vào hóa đơn.

- Sau khi dùng món xong, khách hàng tiến đến quầy thu ngân và nhân viên sẽ gõ số phần trăm giảm giá rồi nhấn nút “Thanh toán” ở góc dưới bên phải để hoàn thành một đơn hàng.

4.4.3. Giao diện thông tin tài khoản

Người dùng có thể thay đổi tên hiển thị và mật khẩu tài khoản của chính mình.



Thông tin cá nhân

THÔNG TIN TÀI KHOẢN CỦA BẠN

Tên đăng nhập: admin1

Tên hiển thị: Hoàng Phan

Mật khẩu hiện tại:

Mật khẩu mới:

Nhập lại mật khẩu mới:

Cập nhật thông tin

Thao tác thực hiện:

- Trên thanh công cụ phía trên của cửa sổ bán hàng, click vào mục “Thông tin tài khoản” rồi chọn “Thông tin cá nhân”.
- Giao diện thông tin cá nhân mở ra với các trường thông tin có thể thay đổi bao gồm: tên hiển thị, mật khẩu hiện tại, mật khẩu mới, nhập lại mật khẩu mới. Lưu ý: không được phép thay đổi tên đăng nhập.
- Người dùng nhấp chuột chọn vào TextBox bên cạnh nhãn tiêu đề của trường dữ liệu cần thay đổi. Rồi nhập thông tin mới vào.
- Để thay đổi mật khẩu, cần nhập đúng mật khẩu hiện tại đang sử dụng và nhập tiếp mật khẩu mới vào 2 TextBox cuối cùng.
- Nhấn nút Cập nhật thông tin nằm dưới cùng của cửa sổ để hoàn thành.

4.4.4. Giao diện quản trị viên

Người dùng đăng nhập với vai trò quản trị viên sẽ sử dụng được chức năng Quản trị viên để thực hiện một số thay đổi đối với các bảng có trong cơ sở dữ liệu. Tài khoản của nhân viên thu ngân thì không thể truy cập được chức năng này.

Quản trị viên Thông tin tài khoản (Trần Ngọc Hương Giang)

Dưới đây là giao diện của quản trị viên với thẻ doanh thu đang được mở. Trong thẻ này, ta có thể xem danh sách những hóa đơn từ ngày này đến ngày khác.

Thao tác thực hiện:

- Trên thanh công cụ phía trên cùng của cửa sổ, nhấp chuột vào dòng chữ “Quản trị viên”. Cửa sổ mới hiện ra với thẻ “Doanh thu” đang được mở.
- Chọn ngày bắt đầu tại DateTimePicker ở phía trên bên phải, chọn ngày kết thúc ngay bên dưới.
- Nhấn nút “Thống kê” để hệ thống bắt đầu hiển thị danh sách các hóa đơn trong khoảng thời gian đã chọn.

Trang quản trị viên

Doanh thu Món Danh mục Bàn ăn Tài khoản Bảo cáo

DANH SÁCH HÓA ĐƠN

	Tên bàn	Tổng tiền	Ngày vào	Ngày ra	Giảm giá
▶	Bàn 1	68000	01/04/2025	01/04/2025	20
	Bàn 11	63000	09/04/2025	09/04/2025	30
*					

Từ: 01/04/2025

Đến: 30/04/2025

Thống kê

Nếu muốn sửa đổi thông tin thực đơn, danh mục món, bàn, tài khoản thì chọn trên thanh tab phía trên cùng. Ở đây, ta sẽ thay đổi thông tin món trong thực đơn.

Thao tác thực hiện:

- Di chuyển con trỏ đến thẻ cần mở rồi nhấp chuột.

- Danh sách sẽ hiển thị toàn bộ những dữ liệu đang tồn tại trong bảng đó.
- Quản trị viên có thể thay đổi thông tin của một bản ghi bất kỳ trừ trường thông tin ID để nhận dạng. Chẳng hạn như trong hình phía dưới, ta có thể thực hiện thao tác thay đổi tên của món, danh mục và đơn giá của món đó.
- Nhập vào những thay đổi mới rồi nhấn nút “Sửa” ở phía dưới bên phải.
- Nếu muốn thêm mới, ta cũng nhập đầy đủ các trường thông tin cho bản ghi mới rồi nhấn nút “Thêm”.
- Để xóa một bản ghi nào đó, ta nhấp chuột chọn vào bản ghi đó ở danh sách bên trái của sổ, kiểm tra thông tin của bản ghi ở bên phải. Sau đó, ta nhấn nút “Xóa”.
- Nút “Xem” được sử dụng để hiển thị toàn bộ những bản ghi có trong bảng đó.

Trang quản trị viên

Doanh thu Món Danh mục Bàn ăn Tài khoản Báo cáo

DANH SÁCH SẢN PHẨM

	ID	Name	CateID	Price
▶	1	Cà phê sữa đá	1	25000
	4	Cà phê trứng	1	40000
	5	Trà vải	2	25000
	6	Nước ép dưa hấu	4	30000
	7	Bạc xỉu	1	35000
	8	Trà nhài	2	30000
	11	Sinh tố xoài	3	38000
	12	Chanh dây đá xay	3	35000
	13	Trà ổi hồng	2	20000
	14	Trà sữa thái đỏ	2	15000
	15	Trà đào cam sả	2	35000
	16	Xoài lắc	7	25000
	17	Hương dương	7	20000
	18	Kem dưa lưới	8	30000

ID:

Tên:

Danh mục:

Giá:

CHƯƠNG 5. PHÁT TRIỂN HỆ THỐNG

Phần mềm Quản lý quán cà phê được xây dựng dựa trên mô hình kiến trúc 3 lớp đã nêu tại Chương 2.

5.1. Cấu trúc thư mục

QuanLyCafeTest (Thư mục gốc)

- DAO (Data Access Layer - Tầng truy cập dữ liệu)
 - + AccountDAO.cs
 - + BillDAO.cs
 - + BillInfoDAO.cs
 - + CategoryDAO.cs
 - + DataProvider.cs
 - + FoodDAO.cs
 - + MenuDAO.cs
 - + TableDAO.cs
- DTO (Business Logic Layer - Tầng logic nghiệp vụ)
 - + Account.cs
 - + Bill.cs
 - + BillInfo.cs
 - + Category.cs
 - + Food.cs
 - + Menu.cs
 - + Table.cs
- Presentation Layer (Tầng giao diện)
 - + account_profile.cs (Form quản lý thông tin tài khoản)
 - + admin.cs (Form quản lý admin)
 - + login.cs (Form đăng nhập)
 - + table_management.cs (Form quản lý bàn)
 - + Program.cs (Điểm khởi chạy ứng dụng)
- App.config (Cấu hình kết nối tới cơ sở dữ liệu, chứa chuỗi kết nối connectionString)
- CrystalReport1.rpt (File báo cáo, dùng để xuất danh sách hóa đơn)

- QuanLyCafeDataSet.xsds (Định nghĩa cấu trúc cơ sở dữ liệu, các bảng như Account, Bill, Table, ...)

5.2. Sử dụng mô hình 3 lớp cho dự án

5.2.1. Data Access Layer (Tầng truy cập dữ liệu) - Thư mục DAO

- Chức năng: Giao tiếp với cơ sở dữ liệu, thực hiện các thao tác truy vấn (SELECT, INSERT, UPDATE, DELETE).
- Các file chính:
 - + AccountDAO.cs: Xử lý truy vấn liên quan đến tài khoản (đăng nhập, cập nhật thông tin).
 - + BillDAO.cs, BillInfoDAO.cs: Quản lý hóa đơn và chi tiết hóa đơn.
 - + TableDAO.cs: Quản lý bàn.
 - + DataProvider.cs: Cung cấp kết nối đến cơ sở dữ liệu SQL Server.
- Ví dụ: File BillDAO.cs chứa các phương thức như lấy danh sách hóa đơn.

5.2.2. Business Logic Layer (Tầng logic nghiệp vụ) - Thư mục DTO

- Chức năng: Định nghĩa các đối tượng dữ liệu (DTO - Data Transfer Object) và xử lý logic nghiệp vụ, làm trung gian giữa tầng giao diện và tầng truy cập dữ liệu.
- Các file chính:
 - + Account.cs, Bill.cs, Table.cs,: Định nghĩa các thuộc tính của đối tượng (ví dụ: Bill.cs chứa Id, DateCheckIn, Discount,...).
- Ví dụ: DTO Bill.cs được sử dụng để truyền dữ liệu hóa đơn từ tầng DAO lên tầng giao diện.

5.2.3. Presentation Layer (Tầng giao diện)

- Chức năng: Tầng này hiển thị giao diện người dùng và xử lý tương tác với người dùng.
- Các file chính:
 - + login.cs: Form đăng nhập.
 - + admin.cs: Form quản lý dành cho admin.
 - + table_management.cs: Form tạo đơn hàng (chuyển bàn, thanh toán,...).
 - + account_profile.cs: Form quản lý thông tin tài khoản cá nhân.
 - + Program.cs: Điểm khởi chạy ứng dụng, khởi tạo form chính.

- Ví dụ: Form table_management.cs hiển thị danh sách bàn, gọi các phương thức từ tầng DAO để cập nhật trạng thái.

5.3. Mã nguồn của phần mềm

Trong báo cáo này, em sẽ giải thích phần mã nguồn của tệp FoodDAO.cs, table_mananer.cs và những tệp liên quan để tránh dài dòng. Các file khác cũng tương tự như vậy.

- Phương thức GetFoodByCategoryID(int id) (FoodDAO.cs) để lấy toàn bộ thông tin của các món nằm trong danh mục có id_cate = id. Kết quả trả về là một danh sách các đối tượng Food.

Dòng	Mã nguồn
1	public List<Food> GetFoodByCategoryID(int id)
2	{
3	List<Food> list = new List<Food>();
4	string query = "select * from Food where id_cate = " + id;
5	DataTable data = DataProvider.Instance.ExecuteQuery(query);
6	foreach (DataRow item in data.Rows)
7	{
8	Food food = new Food(item);
9	list.Add(food);
10	}
11	return list;
12	}

- + Dòng 3: Khởi tạo một danh sách lưu trữ các đối tượng Food được lấy từ cơ sở dữ liệu.
- + Dòng 4: Tạo câu truy vấn trong SQL Server.
- + Dòng 5: Gọi phương thức ExecuteQuery từ file DataProvider.cs, phương thức này thực thi câu truy vấn và trả về một DataTable (một bảng dữ liệu).
- + Dòng 6: Duyệt qua từng bản ghi trong DataTable vừa trả về.
- + Dòng 8: Tạo một đối tượng Food mới với dữ liệu của bản ghi đang được duyệt.
- + Dòng 9: Thêm bản ghi dữ liệu đó vào danh sách các đối tượng Food.
- Phương thức phía trên khi sử dụng trong table_manager.cs để tải lên danh sách món thuộc một danh mục món có ID là id và hiển thị tên của toàn bộ các món có trong danh mục đó lên ComboBox.

Dòng	Mã nguồn
------	----------

1	void LoadFoodListByCateID(int id)
2	{
3	List<Food> listFood = FoodDAO.Instance.GetFoodByCategoryID(id);
4	cboFoods.DataSource = listFood;
5	cboFoods.DisplayMember = "Name";
6	}

- + Dòng 3: Gọi GetFoodByCategoryID(id) trả về danh sách món có id_cate = id.
- + Dòng 4: Gán danh sách listFood vừa tạo làm nguồn dữ liệu cho cboFoods.
- + Dòng 5: Chọn tên làm thuộc tính để hiển thị trên cboFoods.

- Trong file FoodDAO.cs còn có thêm một phương thức tương tự để hiển thị toàn bộ dữ liệu của món, đó là phương thức GetListFood(). Nó được gọi đến tại form admin.cs để những tài khoản quản trị viên có thể xem được toàn bộ và tiến hành những truy vấn (SELECT, UPDATE, DELETE, INSERT).

The screenshot shows a web application interface for managing products. At the top, there's a navigation bar with tabs: 'Doanh thu', 'Món', 'Danh mục', 'Bàn ăn', 'Tài khoản', and 'Báo cáo'. The 'Món' tab is selected. Below the navigation bar, the title 'DANH SÁCH SẢN PHẨM' is displayed. On the left, there's a table listing products with columns: ID, Name, CateID, and Price. The table contains 18 rows of data. On the right, there's a form for editing a product. It includes fields for 'ID' (set to 1), 'Tên' (set to 'Cà phê sữa đá'), 'Danh mục' (set to 'Cà phê'), and 'Giá' (set to 25000). There's a 'Tìm kiếm' button and four action buttons at the bottom: 'Thêm', 'Sửa', 'Xóa', and 'Xem'.

ID	Name	CateID	Price
1	Cà phê sữa đá	1	25000
4	Cà phê trứng	1	40000
5	Trà vải	2	25000
6	Nước ép dưa hấu	4	30000
7	Bạc xiu	1	35000
8	Trà nhài	2	30000
11	Sinh tố xoài	3	38000
12	Chanh dây đá xay	3	35000
13	Trà ổi hồng	2	20000
14	Trà sữa thái đỏ	2	15000
15	Trà đào cam sả	2	35000
16	Xoài lắc	7	25000
17	Hương dương	7	20000
18	Kem dưa lưới	8	30000

- Để hiển thị toàn bộ bàn hiện có, ta cần viết một phương thức LoadTable() nằm trong table_manager.cs. Nó sẽ tải danh sách các bàn từ cơ sở dữ liệu và hiển thị chúng dưới dạng các Button trên giao diện nằm trong một FlowLayoutPanel.


Dòng	Mã nguồn
1	void LoadTable()
2	{
3	flpTable.Controls.Clear();


```

4      List<Table> tableList = TableDAO.Instance.LoadTableList();
5      foreach (Table item in tableList)
6      {
7          Button btn = new Button() { Width = TableDAO.TableWidth,
Height = TableDAO.TableHeight };
8          btn.Text = item.Name + Environment.NewLine + item.Status;
9          btn.Click += btn_Click;
10         btn.Tag = item;
11         switch (item.Status)
12         {
13             case "Trống":
14                 btn.BackColor = Color.Azure;
15                 btn.ForeColor = Color.DarkSlateGray;
16                 break;
17             default:
18                 btn.BackColor = Color.DarkSlateGray;
19                 btn.ForeColor = Color.Azure;
20                 break;
21         }
22         flpTable.Controls.Add(btn);
23     }
24 }

```

- + Dòng 3: Xóa tất cả các control (các nút bàn) hiện có trong để chuẩn bị hiển thị danh sách bàn mới. Đảm bảo giao diện không bị trùng lặp hoặc hiển thị sai.
- + Dòng 7: Tạo một nút để đại diện cho một bàn có trong danh sách và thiết lập kích thước của nút đó.
- + Dòng 8: Hiển thị nội dung của mỗi nút gồm tên bàn và trạng thái của bàn đó.
- + Dòng 9: Gắn sự kiện Click cho nút đang khởi tạo.
- + Dòng 10: Lưu dữ liệu item (bàn đang duyệt) cho nút tương ứng.
- + Dòng 11: Kiểm tra trạng thái bàn để thay đổi màu nền của nút.
- + Dòng 22: Thêm nút vừa tạo vào FlowLayoutPanel.

 Phần mềm Quản lý quán cà phê



- Trong quá trình tạo một hóa đơn, ta cần nhìn được danh sách món đã gọi trước đó. Chính vì vậy, phương thức ShowBill(int id) viết ra để hiển thị danh sách món đã gọi tại bàn đó lên một ListView, đồng thời tính và hiển thị tổng tiền hóa đơn lên một TextBox.

Tên món	Số lượng	Đơn giá	Thành tiền
Nước ép dưa hấu	1	30000	30000
Trà ổi hồng	2	20000	40000
Bạc xỉu	1	35000	35000

Cà phê

Bạc xỉu

Số lượng:

1

Thêm món

Chuyển bàn

Giảm giá

Bàn 1

0

Thanh toán

Tổng tiền:

105000

Dòng	Mã nguồn
1	void ShowBill(int id) {
2	lsvBill.Items.Clear();
3	List<QuanLyCafe.DTO.Menu> listBillInfo =
	MenuDAO.Instance.GetListMenuByTable(id);
4	int totalPrice = 0;
5	foreach (QuanLyCafe.DTO.Menu item in listBillInfo) {
6	ListViewItem lsvItem = new
	ListViewItem(item.Name.ToString());
7	lsvItem.SubItems.Add(item.Count.ToString());
8	lsvItem.SubItems.Add(item.Price.ToString());
9	lsvItem.SubItems.Add(item.Total.ToString());
10	totalPrice += item.Total;
11	lsvBill.Items.Add(lsvItem); }
12	txtTotalPrice.Text = totalPrice.ToString(); }

- + Dòng 3: Gọi phương thức GetListMenuByTable từ lớp MenuDAO (nằm trong thư mục DAO - Data Access Layer) để hiển thị lên danh sách các món đã gọi của bàn có ID là id.
- + Dòng 4: Khai báo giá trị ban đầu của tổng tiền là 0.

- + Dòng 5: Duyệt qua từng đối tượng Menu trong danh sách listBillInfo để hiển thị thông tin từng món lên ListView.
- + Dòng 6: Tạo một ListViewItem mới (một hàng trong ListView). Cột đầu tiên của hàng này hiển thị tên món (item.Name), ví dụ: "Cà phê đen".
- + Dòng 7 - 8 - 9: Tạo 3 cột phụ để hiển thị số lượng, đơn giá, thành tiền cho từng món.
- + Dòng 10: Tính tổng tiền = tổng tiền hiện tại + thành tiền của món mới thêm vào hóa đơn.
- + Dòng 11: Thêm bản ghi thông tin của món mới vào ListView để hiển thị.
- + Dòng 12: Chuyển định dạng của tổng tiền thành Text để hiển thị lên TextBox.

CHƯƠNG 6. KẾT LUẬN

6.1. Kết quả đạt được

6.1.1. Về kiến thức

- Học hỏi thêm được mô hình 3 lớp trong xây dựng phần mềm hoặc Website.
- Biết sử dụng C# để tạo nên một phần mềm đơn giản.
- Áp dụng những kiến thức phân tích và thiết kế hệ thống để xây dựng quy trình nghiệp vụ và các chức năng cốt lõi nhất của hệ thống.
- củng cố lại về hệ quản trị cơ sở dữ liệu SQL Server và cách thức để tạo nên cơ sở dữ liệu cho riêng mình.

6.1.2. Về phần mềm

- Hoàn thiện chức năng cơ bản nhất: tạo hóa đơn và thanh toán. Ngoài ra, còn những tính năng như thêm, sửa, xóa thông tin của các bản ghi có trong cơ sở dữ liệu.
- Giao diện cơ bản, không hiện đại như những phần mềm khác hiện nay nhưng dễ dàng làm quen và thực thi các lệnh.

6.2. Các hạn chế

- Do lượng kiến thức còn hạn chế, dẫn đến việc triển khai gặp một chút khó khăn và chưa đạt được kết quả tốt nhất.
- Phần mềm không sử dụng thêm các thư viện nên thiết kế giao diện trông sẽ không hiện đại so với các phần mềm được sử dụng rộng rãi trên thị trường hiện nay.
- Vẫn còn thiếu sót nhiều tính năng: quản lý nhân viên, quản lý kho hàng, quản lý công thức pha chế,... để đưa vào sử dụng thực tiễn.
- Trong cơ sở dữ liệu, mật khẩu tài khoản chưa được mã hóa.

6.3. Hướng phát triển

- Trong tương lai, cụ thể là khóa luận tốt nghiệp ở kỳ học sắp tới đây, em sẽ bổ sung thêm những tính năng mới để gia tăng tính ứng dụng của phần mềm.
- Sử dụng bổ sung nhiều thư viện khác giúp cải thiện giao diện người dùng, dễ làm quen và thao tác hơn.
- Tìm hiểu các chuẩn mã hóa mật khẩu để tạo nên lớp rào chắn tốt hơn cho các thông tin nhạy cảm.

TÀI LIỆU THAM KHẢO

- Hồ, U. T. T. (2019, 09 24). *Đề tài: Xây dựng phần mềm quản lý quán cà phê, HOT, 9đ*. SlideShare. Retrieved April 12, 2025, from https://www.slideshare.net/slideshow/luan-van-xay-dung-phan-mem-quan-li-q-uan-ca-phe-hot-9d/175765522?utm_source=chatgpt.com#8
- Lê, N. S. (2020, 7 26). *Tìm Hiểu Về Mô Hình 3 Lớp (3 Layer)*. Viblo. <https://viblo.asia/p/tim-hieu-ve-mo-hinh-3-lop-3-layer-Qbq5Qmyz5D8>
- Ngọc, A. (2024, 07 10). *Chuyển đổi số đi đến 'từng ngõ, từng nhà, từng người', kinh tế số đạt gần 2 triệu tỉ đồng*. Tuổi trẻ online. <https://tuoitre.vn/chuyen-doi-so-di-den-tung-ngo-tung-nha-tung-nguoi-kinh-te-so-dat-gan-2-trieu-ti-dong-20240710153834629.htm>
- Phu Xuan University. (2021, 6 5). *PHU XUAN University*. Công nghệ thông tin: Mô Hình 3 Tier và 3 layer. <https://phuxuan.edu.vn/cong-nghe-thong-tin-mo-hinh-3-tier-va-3-layer/>
- Team, K. (2016, 11 22). *KTeam*. Lập trình phần mềm Quản lý quán cafe với C# Winform. <https://howkteam.vn/course/lap-trinh-phan-mem-quan-ly-quan-cafe-voi-c-winform-24>
- <https://tuoitre.vn/chuyen-doi-so-di-den-tung-ngo-tung-nha-tung-nguoi-kinh-te-so-dat-gan-2-trieu-ti-dong-20240710153834629.htm>