**Mục lục**

[DANH MỤC CÁC TỪ VIẾT TẮT 9](#_Toc153304379)

[LỜI MỞ ĐẦU 1](#_Toc153304380)

[1. Tính câp thiết của đề tài: 1](#_Toc153304381)

[2. Mục đích nghiên cứu: 1](#_Toc153304382)

[3. Nhiệm vụ nghiên cứu: 1](#_Toc153304383)

[4. Phương pháp nghiên cứu: 2](#_Toc153304384)

[5. Các kết quả đạt được của đề tài: 3](#_Toc153304385)

[6. Kết cấu của bài tiểu luận: 3](#_Toc153304386)

[CHƯƠNG 1: CƠ SỞ LÝ THUYẾT 4](#_Toc153304387)

[1.1 Mô hình hệ thống thực tế: 4](#_Toc153304388)

[1.2 Mô tả nghiệp vụ các chức năng: 4](#_Toc153304389)

[1.3 Yêu cầu phi chức năng: 5](#_Toc153304390)

[1.4 Công nghệ được sử dụng: 5](#_Toc153304391)

[1.5 Mô hình ba lớp (Three-Layer): 11](#_Toc153304392)

[CHƯƠNG 2: PHÂN TÍCH THIẾT KẾ HỆ THỐNG PHẦN MỀM 15](#_Toc153304394)

[2.1 Sơ đồ Usecase tổng quan: 15](#_Toc153304395)

[2.2 Đặc tả usecase: 15](#_Toc153304397)

[2.2.1 Đăng nhập: 15](#_Toc153304398)

[2.2.2 Bán hàng/ lập hóa đơn: 16](#_Toc153304401)

[2.2.3 Quản lý hóa đơn: 19](#_Toc153304404)

[2.2.4 Quản lý sản phẩm: 20](#_Toc153304407)

[2.2.5 Quản lý nhân viên: 25](#_Toc153304411)

[2.2.6 Quản lý doanh thu: 26](#_Toc153304413)

[2.2.7 Quản lý lương: 27](#_Toc153304416)

[2.3 Sơ đồ lớp: 29](#_Toc153304417)

[2.4 Sơ đồ tuần tự: 30](#_Toc153304419)

[2.4.1 Đăng nhập: 30](#_Toc153304420)

[2.4.2 Bán hàng/ lập hóa đơn: 30](#_Toc153304422)

[2.4.3 Quản lý hóa đơn: 31](#_Toc153304424)

[2.4.4 Quản lý sản phẩm: 31](#_Toc153304426)

[2.4.4.1 Thêm sản phẩm: 31](#_Toc153304427)

[2.4.4.2 Sửa thông tin sản phẩm: 32](#_Toc153304429)

[2.4.4.3 Xóa sản phẩm: 32](#_Toc153304431)

[2.4.4.4 Thống kê doanh thu: 33](#_Toc153304433)

[2.4.5 Quản lý nhân viên: 34](#_Toc153304435)

[2.4.5.1 Thêm nhân viên: 34](#_Toc153304436)

[2.4.5.2 Sửa thông tin nhân viên: 34](#_Toc153304438)

[2.4.6 Quản lý lương: 35](#_Toc153304441)

[2.5 Thiết kế CSDL: 36](#_Toc153304442)

[2.5.1 Bảng nRole: 36](#_Toc153304443)

[2.5.2 Bảng Account: 36](#_Toc153304445)

[2.5.3 Bảng NhanVien: 36](#_Toc153304447)

[2.5.4 Bảng Bill: 36](#_Toc153304449)

[2.5.5 Bảng BillChiTiet: 37](#_Toc153304451)

[2.5.6 Bảng NuocUong: 37](#_Toc153304453)

[2.5.7 Bảng Salary: 38](#_Toc153304455)

[CHƯƠNG 3: XÂY DỰNG PHẦN MỀM: 40](#_Toc153304457)

[3.1 Xây dựng Solution: 40](#_Toc153304458)

[3.2 Phân tích giao diện phần mềm: 42](#_Toc153304463)

[3.2.1 Form đăng nhập: 42](#_Toc153304464)

[3.2.2 Form Main: 45](#_Toc153304467)

[3.2.3 Form home: 48](#_Toc153304471)

[3.2.4 Form Bill: 55](#_Toc153304478)

[3.2.5 Form Manage: 56](#_Toc153304481)

[3.2.5.1 Tab thống kê: 57](#_Toc153304483)

[3.2.5.2 Tab nhân viên: 57](#_Toc153304484)

[3.2.5.3 Tab kho hàng: 59](#_Toc153304488)

[3.2.6 Form Salary: 61](#_Toc153304493)

[KẾT LUẬN 63](#_Toc153304494)

[ Kết quả đạt được: 63](#_Toc153304495)

[ Hạn chế: 63](#_Toc153304496)

[ Hướng phát triển: 63](#_Toc153304497)

[TÀI LIỆU THAM KHẢO 64](#_Toc153304498)

**DANH MỤC HÌNH ẢNH**

**CHƯƠNG 1: CƠ SỞ LÝ THUYẾT**

[Hình 1.1 Sơ đồ Three-Layer 12](#_Toc153304662)

**CHƯƠNG 2: PHÂN TÍCH THIẾT KẾ HỆ THỐNG**

[Hình 2. 1 Sơ đồ Usecase tổng quát 15](#_Toc153304689)

[Hình 2. 2 Phân rã Usecase đăng nhập 15](#_Toc153304690)

[Hình 2. 3 Phân rã Usecase bán hàng/lập hóa đơn 17](#_Toc153304691)

[Hình 2. 4 phân rã Usecase quản lý hóa đơn 19](#_Toc153304692)

[Hình 2. 5 Phân rã Usecase Quản lý sản phẩm 20](#_Toc153304693)

[Hình 2. 6 Phân rã Usecase quản lý nhân viên 22](#_Toc153304694)

[Hình 2. 7 Phân rã Usecase quản lý doanh thu 26](#_Toc153304695)

[Hình 2. 8 Phân ra UseCase quản lý lương 27](#_Toc153304696)

[Hình 2. 9 Sơ đồ lớp 29](#_Toc153304697)

[Hình 2. 10 Sơ đồ tuần tự đăng nhập 30](#_Toc153304698)

[Hình 2. 11 Sơ đồ tuần tự bán/lập hóa đơn 30](#_Toc153304699)

[Hình 2. 12 Sơ đồ tuần tự xóa hóa đơn 31](#_Toc153304700)

[Hình 2. 13 Sơ đồ tuần tự thêm sản phẩm 32](#_Toc153304701)

[Hình 2. 14 Sơ đồ tuần tự sửa sản phẩm 32](#_Toc153304702)

[Hình 2. 15 Sơ đồ tuần tự xóa sản phẩm 32](#_Toc153304703)

[Hình 2. 16 Sơ đồ tuần tự thống kê doanh thu 33](#_Toc153304704)

[Hình 2. 17 Sơ đồ tuần tự thêm nhân viên 34](#_Toc153304705)

[Hình 2. 18 Sơ đồ tuần tự sửa nhân viên 34](#_Toc153304706)

[Hình 2. 19 Sơ đồ tuần tự xóa nhân viên 35](#_Toc153304707)

[Hình 2. 20 Sơ đồ tuần tự thanh toán lương 35](#_Toc153304708)

[Hình 2. 21 Sơ đồ CSDL 39](#_Toc153304709)

**CHƯƠNG 3: XÂY DỰNG PHẦN MỀM**

[Hình 3. 1 Cấu trúc Solution 40](#_Toc153304710)

[Hình 3. 2 File DAO 41](#_Toc153304711)

[Hình 3. 3 File DTO 41](#_Toc153304712)

[Hình 3. 4 Cấu trúc các form 41](#_Toc153304713)

[Hình 3. 5 Form đăng nhập 42](#_Toc153304714)

[Hình 3. 6 Thông báo đăng nhập sai 44](#_Toc153304715)

[Hình 3. 7 Form main 45](#_Toc153304716)

[Hình 3. 8 Xác nhận đăng xuất 46](#_Toc153304717)

[Hình 3. 9 Form Home 48](#_Toc153304718)

[Hình 3. 10 Form hóa đơn 50](#_Toc153304719)

[Hình 3. 11 Xác nhận thanh toán 51](#_Toc153304720)

[Hình 3. 12 Thông báo thanh toán thành công 51](#_Toc153304721)

[Hình 3. 13 Xác nhận in hóa đơn 51](#_Toc153304722)

[Hình 3. 14 Hóa đơn sau khi in 52](#_Toc153304723)

[Hình 3. 15 Form Bill 55](#_Toc153304724)

[Hình 3. 16 Lọc danh sách hóa đơn 56](#_Toc153304725)

[Hình 3. 17 Form mange 56](#_Toc153304726)

[Hình 3. 18 Tab thống kê 57](#_Toc153304727)

[Hình 3. 19 Tab nhân viên 57](#_Toc153304728)

[Hình 3. 20 Form thêm tài khoản 58](#_Toc153304729)

[Hình 3. 21 Form Danh sách tài khoản 59](#_Toc153304730)

[Hình 3. 22 Tab kho hàng 59](#_Toc153304731)

[Hình 3. 23 Form Thêm sản phẩm 60](#_Toc153304732)

[Hình 3. 24 Form sửa sản phẩm 60](#_Toc153304733)

[Hình 3. 25 Xác nhận xóa hàng 61](#_Toc153304734)

[Hình 3. 26 Form Salary 61](#_Toc153304735)

[Hình 3. 27 Xác nhận lương 62](#_Toc153304736)

**DANH MỤC BẢNG**

[Bảng 2. 1 Đặc tả usecase đăng nhập 16](#_Toc153304765)

[Bảng 2. 2 Đặc tả usecase bán hàng/lập hóa đơn 18](#_Toc153304766)

[Bảng 2. 3 Đặc tả usecase xóa hóa đơn đơn 20](#_Toc153304767)

[Bảng 2. 4 Đặc tả usecase quản lý hàng hóa 24](#_Toc153304768)

[Bảng 2. 5 Đặc tả usecase quản lý nhân viên 25](#_Toc153304769)

[Bảng 2. 6 Đặc tả usecase quản lý doanh thu 27](#_Toc153304770)

[Bảng 2. 7 Bảng đặc tả Usecase Quản lý lương 28](#_Toc153304771)

[Bảng 2. 8 Bảng dữ liệu nRole 36](#_Toc153304772)

[Bảng 2. 9 Bảng dữ liệu Account 36](#_Toc153304773)

[Bảng 2. 10 Bảng dữ liệu NhanVien 36](#_Toc153304774)

[Bảng 2. 11 Bảng dữ liệu Bill 37](#_Toc153304775)

[Bảng 2. 12 Bảng dữ liệu BillChiTiet 37](#_Toc153304776)

[Bảng 2. 13 Bảng dữ liệu NuocUong 38](#_Toc153304777)

[Bảng 2. 14 Bảng dữ liệu Salary 38](#_Toc153304778)

**DANH MỤC CÁC TỪ VIẾT TẮT**

|  |  |  |
| --- | --- | --- |
| **STT** | **Ký hiệu viết tắt** | **Chữ viết đầy đủ** |
| 1 | IDE | Integrated development environment |
| 2 | UC | Usecase |
| 3 | CSDL | Cơ sở dữ liệu |

**LỜI MỞ ĐẦU**

1. **Tính câp thiết của đề tài:**

Hiện nay mô hình kinh doanh tư nhân phát triển rất nhanh không chỉ ở những thành phố lớn mà ở các khu vực trải dài cả nước đã và đang bùng nổ một cách nhanh chống, một trong số đó chính là mô hình kinh doanh quán cà phê. Theo thống kê hiện nay, Việt Nam có khoảng hơn 500.000 quán cà phê (bao gồm cả chuỗi quán, quán cà phê lẻ, quán cà phê vỉa hè…) tập trung chủ yếu ở các thành phố lớn như thành phố Hồ Chí Minh, Hà Nội, Đà Nẵng… [1]

Với sự phát triển của công nghệ, quá trình trao đổi giữa người mua và

người bán diễn ra đơn giản, nhanh gọn cùng với sự chính xác cao nên như cầu để đáp ứng nền tảng công nghệ để phục vụ cho hệ thống kinh doanh của mình cũng ngày càng tăng cao.

Đề tài “Xây dựng phần quản lý quán cà phê C# Winform” được xây dựng thông qua quá trình nghiên cứu với mong muốn đáp ứng các yêu cầu của các tư nhân hỗ trợ trong việc kinh doanh của mình.

1. **Mục đích nghiên cứu:**

Hệ thông của một quán cà phê không chỉ đơn giản với việc bán và mua, kinh doanh quán cafe hiện nay đang là loại hình kinh doanh phổ biến được rất nhiều nhà đầu tư lựa chọn để khởi nghiệp. Việc đầu tư mở một quán cafe đã gian nan thì để quản lý quán cafe đi vào hoạt động tốt sẽ càng khó hơn. Vì vậy bất kỳ quán cafe hay cơ sở kinh doanh nào muốn hoạt động hiệu quả đều cần phải có một sơ đồ quản lý quán cafe rõ ràng. [2]

Mục đích nghiên cứu đề tài là đem lại một trải nghiệm tiện lợi giúp chủ tư nhân có thể dễ dàng , hướng tới xây dựng một hệ thống quán cà phê thông minh và nhanh chóng trong tất cả quy trình quản lý.

1. **Nhiệm vụ nghiên cứu:**

Đề tài thực hiện nghiên cứu quy trình quản lý sản phẩm, quản lý hóa đơn, quản lý nhân sự , thống kê doanh thu. Như các phần mềm quản lý khác, phần mềm này đáp ứng đủ các tiêu chí về thêm, xóa ,sửa các đối tượng nhất định, các tính năng tìm kiếm được xây dựng theo logic.

Với quy trình quản lý sản phẩm: các chức năng hiển thị tên, số lượng trong kho của sản phẩm, ngày nhập và hạn sử dụng được đưa đến cho người sử dụng thấy và có thể dễ dang tính toán và xử lý theo yêu cầu riêng của bản thân. Các số liệu người dùng có thể tự cập nhật mà không cần sự hỗ trợ của nghiệp vụ từ tác giả phần mềm.

Tương tự quy trình quản lý hóa đơn đem đến cho người dùng thấy được tổng thể quá trình kinh doanh trong một khoảng thời gian cụ thể để có những giải pháp phù hợp, cái nhìn rõ ràng minh bạch trong quá trình kinh doanh đối với những trường hợp yêu cầu đối chứng của khách hàng.

Quản lý nhân sự giúp nắm bắt được dễ dàng tình hình nhân sự, dễ dàng cung cấp tài khoản quản lý với vai trò riêng để phục vụ cho quán.

Thống kê doanh thu cho người xem thấy được số liệu trong kinh doanh thông qua các biểu đồ các dạng như: biểu đồ cột, biểu đồ bánh,... tùy theo yêu cầu của tư nhân đối với tác giả/tổ chức xây dựng phần mềm, từ thống kê này giúp người dùng dễ dàng theo dõi lợi nhuận kinh doanh cũng như các nguồn hàng được yêu thích để điều chỉnh các số liệu sao cho hợp lý.

1. **Phương pháp nghiên cứu:**

Thông qua quá trình tìm hiểu từ các quán cà phê nhỏ trong khu vực, thực tế các quán nhỏ lẻ cũng đã có những phần mềm chuyên biệt phục vụ cho quá trình kinh doanh. Các nguồn tài liệu về phần mềm quản lý kinh doanh cũng rất phong phú và đa dạng từ tài liệu chữ, đến hình ảnh.

Sau thời gian nghiên cứu và tìm hiểu, phần mềm này dược xây dựng theo mô hình 3 lớp (three-layer) sử dụng Winform C# và Sql Sever, để thực hiện tạo ra giao diện người dùng cũng như CSDL để tương tác với phần mềm vầ truy xuất, lưu trữ dữ liệu hệ thống.

1. **Các kết quả đạt được của đề tài:**

* Giao diện rõ ràng, đẹp thân thiện với người dùng;
* Xử lý logic CSDL;
* Các chức năng cơ bản được xây dựng đầy đủ
* Có xây dựng những chú thích giúp người dùng dễ dàng sử dụng

1. **Kết cấu của bài tiểu luận:**

Bố cục của bài báo cáo “Xây dựng phần quản lý quán cà phê C# Winform” bao gồm phần Mở đầu, Kết luận , Tải liệu tham khảo và 3 chương:

Chương 1: Cơ sở lý thuyết

Chương 2: Phân tích thiết kế hệ thông

Chương 3: Xây dựng hệ thống

**CHƯƠNG 1: CƠ SỞ LÝ THUYẾT**

* 1. **Mô hình hệ thống thực tế:**

Quán cà phê chuyên bán các loại cà phê nổi tiếng từ các quốc gia các nhau như: Capuccino, Latte,...Nhắm đến tệp khách hàng yêu thích cà phê đến để thưởng thức và trải nghiệm. Quán trang bị một quầy thu ngân và hoạt động theo mô hình của quán ăn tự phục vụ, sơ lược về mô hình này “Quán ăn tự phục vụ được kết hợp giữa không gian độc đáo của quán cà phê và đặc trưng của nhà hàng buffet. Với mô hình này, thực khách sau khi đã lựa chọn, lấy đồ ăn/thức uống tại khu vực phục vụ và tự mang đến bàn để thưởng thức sau tiến hành thanh toán ở quầy thu ngân.” [3] Do đó cần trang bị một hệ thống quản lý để điều hành các hoạt động chính của quán như: quản lý sản phẩm, quản lý nhân viên, quản lý hóa đơn, thanh toán,...

* 1. **Mô tả nghiệp vụ các chức năng:**

Ứng dụng quản lý quán cà phê bao gồm 6 chức năng:

* Đăng nhập: Chức năng giúp người dùng truy cập hệ thống thông qua tài khoản và mật khẩu, các tài khoản mật khẩu có phân quyền riêng biệt với vị trí của mỗi cá nhân cụ thể như: nhân viên, quản trị viên.
* Trang chủ/Order: Chức năng giúp người dùng thực hiện quá trình thanh toán chọn cà phê, lập hóa đơn, chọn món cho khách hàng.
* Quản lý hóa đơn: lịch sử hóa đơn, chi tiết hóa đơn và thao tác lọc xem hóa đơn theo ngày.
* Quản lý tác vụ: chỉ mở khi tài khoản là quản trị viên

+ Quản lý nhân viên: Xem thông tin nhân viên, thêm/xóa/sửa vào danh sách dữ liệu sinh viên, tạo thêm tài khoản cho nhân viên cũng như tài khoản quản trị viên;

+ Quản lý doanh thu: hiển thị tổng số hóa đơn, lợi nhuận, tổng tiền lọc theo ngày. Biểu đồ hình ảnh một cách trực quan cho người dùng quan sát;

+ Quản lý sản phẩm: thêm/xóa/sửa sản phẩm, kiểm tra số lượng còn của sản phẩm, ngày nhập hàng, ngày sản xuất của sản phẩm.

+ Quản lý lương: thanh toán/ kiểm tra thời gian làm việc của nhân viên với tổng số lương phù hợp.

* 1. **Yêu cầu phi chức năng:**
* Thiết kế giao diện có tính nhất quán, màu sắc đại diện thương hiệu
* Cơ sở dữ liệu đủ lớn để hoạt động lâu dài;
* Hướng dẫn, thân thiện đảm bảo cho người dùng hiểu cách sử dụng nhanh chống;
* Hiệu năng: hoạt động tốt trên các nền tảng màn hình máy tính 12 inch trở lên
  1. **Công nghệ được sử dụng:** 
     1. **Khái quát ngôn ngữ lập trình C#:**

C# được tạo ra khi nào? 2002. Đúng vậy, ngôn ngữ mã hóa C# chỉ mới 17 tuổi. Nếu là con người, nó sẽ có thể lái xe hoặc bỏ phiếu! Nhưng không có gì trẻ con về ngôn ngữ lập trình này. Trên thực tế, C# được coi là ngôn ngữ lập trình máy tính trưởng thành và được sử dụng rộng rãi trên toàn thế giới. [4]

C# (phát âm là "See Sharp") là ngôn ngữ lập trình hiện đại, hướng đối tượng và an toàn kiểu. C# cho phép các nhà phát triển xây dựng nhiều loại ứng dụng an toàn và mạnh mẽ chạy trong .NET. C# có nguồn gốc từ họ ngôn ngữ C và sẽ quen thuộc ngay với các lập trình viên C, C++, Java và JavaScript.

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.

Ưu điểm của C#:

* Ngôn ngữ lập trình thuần hướng đối tượng;
* Từ họ ngôn ngữ C nên dễ dàng đối với các lập trình viên muốn học tập;
* Có khả năng kết nối với database dễ dàng;
* Các dự án được build/code trên IDE Visual Studio mạnh mẽ hiện đại mảu MicroSoft
* Hỗ trợ mạnh mẽ xây dựng WebApp với c# hoặc ASP.NET

Nhược điểm C#:

* Cần máy tính cấu hình cao để chạy một cách hiệu quả vì IDE Visual Studio khá tốn nhiều tài nguyên;
* Tốc độ chạy chậm hơn so với các ngôn ngữ lập trình khác như C++ và Assembly;
* Là ngôn ngữ lập trình hướng đối tượng nên yêu cầu lập trình viên phải có nền tảng vững về OOP (Object-oriented programming).

Yêu cầu của hệ điều hành để thiết lập:

* Windows Server 2016 (phiên bản 1709);
* Windows 10 Cập nhật Kỷ niệm (phiên bản 1607) (x86 và x64);
* Windows 10 Creators Update (phiên bản 1703) (x86 và x64);
* Windows 10 Fall Creators Update (phiên bản 1709) (x86 và x64);
* Windows Server 2012 R2 (x64);
* Windows 8.1 (x86 và x64);
* Windows Server 2012 (x64);
* Windows Server 2008 R2 Gói Dịch vụ 1 (x64);
* Windows 7 Gói Dịch vụ 1 (x86 và x64).
  + 1. **Khái quát về trình quản trị cơ sở dữ liệu MS Sql Sever 2022:**

SQL Server là một hệ thống quản lý cơ sở dữ liệu quan hệ (RDBMS) được phát triển bởi Microsoft. Nó chủ yếu được thiết kế và phát triển để cạnh tranh với cơ sở dữ liệu MySQL và Oracle. SQL Server hỗ trợ ANSI SQL, là ngôn ngữ SQL (Ngôn ngữ truy vấn có cấu trúc) tiêu chuẩn. Tuy nhiên, SQL Server đi kèm với việc triển khai ngôn ngữ SQL riêng, T-SQL (Transact-SQL).

T-SQL là Ngôn ngữ riêng của Microsoft được gọi là Transact-SQL. Nó cung cấp thêm khả năng khai báo biến, xử lý ngoại lệ, thủ tục lưu trữ, v.v.

SQL Server Management Studio (SSMS) là công cụ giao diện chính cho SQL Server và nó hỗ trợ cả môi trường 32 bit và 64 bit.. [5]

Ưu điểm của Sql Sever:

* Bạn có thể sử dụng nhiều phiên bản MS SQL khác nhau trên cùng 1 máy;
* Bạn có thể phát triển và duy trì riêng biệt các môi trường thử nghiệm khác nhau;
* Xây dựng và duy trì các loại máy chủ dự phòng;
* Hạn chế tối đa các vấn đề rủi ro trên cơ sở dữ liệu.

Nhược điểm của SQL Server

* Microsoft SQL Server chỉ hoạt động trên Windows;
* Bạn cần phải thanh toán chi phí thêm để có thể khởi chạy nhiều Database khác nhau trên Microsoft SQL Server.

SQL Server hiện có nhiều phiên bản khác nhau. Dưới đây là danh sách các phiên bản cùng với tính năng của từng phiên bản:

* Enterprise - bản cao cấp nhất với đầy đủ tính năng.
* Standard - ít tính năng hơn Enterprise, sử dụng khi không cần dùng tới các tính năng nâng cao.
* Workgroup - phù hợp cho các công ty lớn với nhiều văn phòng làm việc từ xa.
* Web - thiết kế riêng cho các ứng dụng web.
* Developer - tương tự như Enterprise nhưng chỉ cấp quyền cho một người dùng duy nhất để phát triển, thử nghiệm, demo. Có thể dễ dàng nâng cấp lên bản Enterprise mà không cần cài lại.
* Express - bản này chỉ dùng ở mức độ đơn giản, tối đa 1 CPU và bộ nhớ 1GB, kích thước tối đa của cơ sở dữ liệu là 10GB.
* Compact - nhúng miễn phí vào các môi trường phát triển ứng dụng web. Kích thước tối đa của cơ sở dữ liệu là 4GB.
* Datacenter - thay đổi lớn trên SQL Server 2008 R2 chính là bản Datacenter Edition. Không giới hạn bộ nhớ và hỗ trợ hơn 25 bản cài.
* Business Intelligence - Business Intelligence Edition mới được giới thiệu trên SQL Server 2012. Phiên bản này có các tính năng của bản Standard và hỗ trợ một số tính năng nâng cao về BI như Power View và PowerPivot nhưng không hỗ trợ những tính năng nâng cao về mức độ sẵn sàng như AlwaysOn Availability Groups…
* Enterprise Evaluation - bản SQL Server Evaluation Edition là lựa chọn tuyệt vời để dùng được mọi tính năng và có được bản cài miễn phí của SQL Server để học tập và phát triển. Phiên bản này có thời gian hết hạn là 6 tháng từ ngày cài. [6]

Yêu cầu về thiết bị:

* OS: Windows 7 (32 bit hoặc 64 bit) trở lên.
* CPU: AMD Opteron, AMD Athlon 64, Intel Xeon với hỗ trợ Intel EM 64T, Intel Pentium IV với hỗ trợ EM 64T.
* Tốc độ CPU: Tối thiểu 1,4 GHz. Khuyến nghị 2,0 GHz
* RAM: Tối thiểu 1GB
* Ổ đĩa trống: Tối thiểu 10GB
* Local Disk: Hỗ trợ các ổ đĩa có kích thước chuẩn là 512 byte và 4KB
* .NET Framework: Phiên bản 3.5 SP1 trở lên
  + 1. **Giới thiệu Diagram.net:**

Diagram.net là một công cụ vẽ sơ đồ mạnh mẽ, hỗ trợ nhiều hình khối, chạy trực tuyến mà không cần cài đặt, miễn phí và không bị hạn chế về số lượng biểu đồ. Bạn có thể tạo sơ đồ về mạng, điện, phác thảo vị trí các căn phòng trong nhà, hoặc vẽ các quy trình kinh doanh, vận hành, sản xuất. Những người làm công nghệ cũng sẽ đánh giá cao Diagram.net vì nó cho phép vẽ nhanh chóng các sơ đồ thiết kế phần mềm, phần cứng và hệ thống.

Diagram.net có thư viện mẫu đa dạng giúp bạn bắt đầu một cách nhanh chóng mà không cần phải vẽ lại từ đầu. Điều này tiết kiệm thời gian và giúp tăng sự hiệu quả trong quá trình tạo và chỉnh sửa sơ đồ của bạn. Đặc biệt, việc chạy trực tuyến mà không yêu cầu cài đặt làm cho Diagram.net trở thành một công cụ thuận tiện và linh hoạt cho mọi người, từ người dùng cá nhân đến các nhóm làm việc trên dự án lớn.

* + 1. **Giới thiệu về IDE Visual Studio:**

Là một môi trường phát triển tích hợp đến từ Microsoft, Visual Studio đầu tiên được ra mắt vào năm 1997, với tên gọi là Visual Studio 97 đây là nỗ lực đầu tiên của Microsoft sử dụng một môi trường phát triển cho nhiều ngôn ngữ lập trình khác nhau.

Visual Studio là một công cụ dành cho nhà phát triển mạnh mẽ mà bạn có thể sử dụng để hoàn thành toàn bộ chu trình phát triển ở một nơi. Đó là một môi trường phát triển tích hợp (IDE) toàn diện mà bạn có thể sử dụng để viết, chỉnh sửa, gỡ lỗi và xây dựng mã, sau đó triển khai ứng dụng của mình. Ngoài việc chỉnh sửa và gỡ lỗi mã, Visual Studio còn bao gồm các trình biên dịch, công cụ hoàn thiện mã, kiểm soát nguồn, tiện ích mở rộng và nhiều tính năng khác để nâng cao mọi giai đoạn của quy trình phát triển phần mềm. [7]

Visual Studio điểm mạnh ở những tính năng sau:

* Hỗ trợ đa ngôn ngữ: Giúp phát hiện lỗi và tham chiếu ngôn ngữ chéo một cách thuận lợi.
* Intelli-Sense: Tự động phát hiện và thực thi cú pháp biến, khai báo biến để giúp người dùng tránh những đoạn mã bị bỏ sót.
* Hỗ trợ đa nền tảng: Hoạt động trên Windows, Linux và Mac.
* Tiện ích mở rộng và Hỗ trợ: Các extension giúp tăng tốc quá trình phát triển ứng dụng mà không làm suy giảm hiệu suất của editor.
* Kết nối với Repository: Tích hợp mượt mà với Git và các Repository khác.
* Code editor đa chức năng: Cho phép bookmark, Quick Navigation, Incremental Search, Regex Search, Multi-item Clipboard, và Task-list.
* Web-Support: Xây dựng và hỗ trợ ứng dụng web.
* Hỗ trợ Terminal tích hợp: Giúp người dùng không phải chuyển đổi giữa hai màn hình.
* Hỗ trợ Git tích hợp: Thuận tiện lấy tài nguyên từ Github Repo trực tuyến và ngược lại.
* Debugger mạnh mẽ: Cho phép kiểm tra trạng thái của chương trình và phát hiện lỗi.
* Thiết kế đa dạng: Cung cấp visual designer cho WPF, Windows Forms, Class designer, Web designer, Mapping Designer và Data Designer, giúp phát triển ứng dụng một cách hiệu quả.

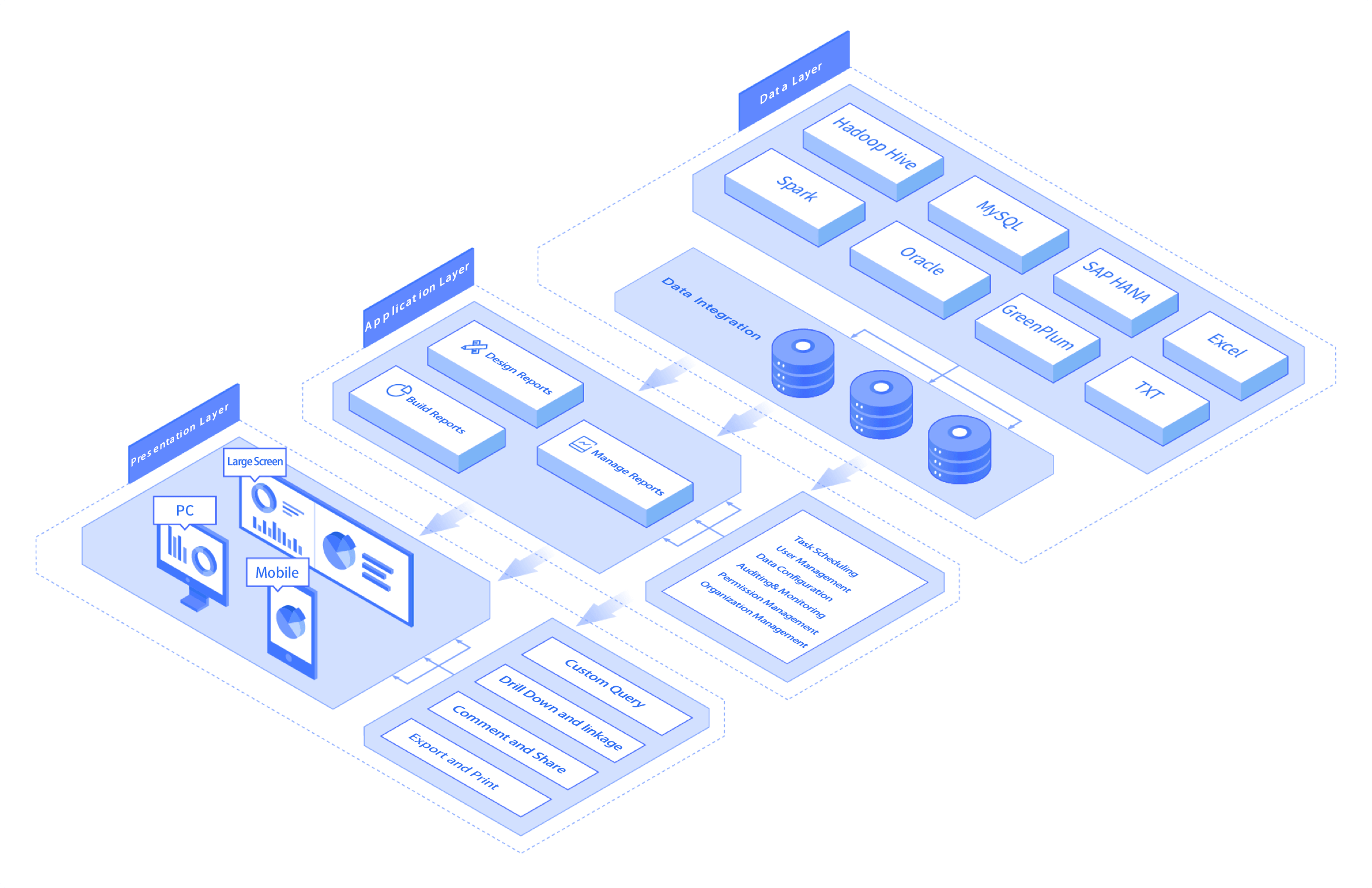
Ưu điểm của Visual Studio:

* Trình cài đặt dựa trên khối lượng công việc - chỉ cài đặt những gì mà lập trình viên cần;
* Các công cụ và tính năng mã hóa mạnh mẽ để xây dựng ứng dụng;
* Hỗ trợ nhiều ngôn ngữ - mã bằng C++, C#, JavaScript, TypeScript, Python,..
* Phát triển đa nền tảng - xây dựng ứng dụng cho mọi nền tảng;
* Tích hợp hỗ trợ mạnh mẽ cộng đồng Github;
* Phát triển có sự hỗ trợ của AI - viết mã hiệu quả hơn với sự hỗ trợ của AI.

Nhược điểm của Visual Studio:

* Giao diện hiện đại nhưng khá khó sử dụng cho người mới học lập trình vì cần khá nhiều bước thiết lập và cài đặt.
* Visual Studio là một IDE nặng nên cần sử dụng nhiều tài nguyên để khởi động và vận hành nên yêu cầu về cấu hình máy tính cũng cao hơn so với các IDE khác.
  1. **Mô hình ba lớp (Three-Layer):**

Kiến trúc 3 lớp là một mô hình client/server, trong đó, thành phần giao diện người dùng (UI), quy tắc xử lý (BR hoặc BL - business rules hoặc business logic), và lưu trữ dữ liệu được phát triển như những module độc lập. Điều đặc biệt của kiến trúc này là sự phân chia rõ ràng và độc lập giữa các lớp chức năng, góp phần quản lý dự án một cách hiệu quả.



Hình 1.1 Sơ đồ Three-Layer

Mô hình gồm 3 lớp:

* Cấp Giao Diện Người Dùng (GUI): Lớp này đảm nhận vai trò hiển thị giao diện và cung cấp các chức năng cho người dùng cuối. Nó tương tác trực tiếp với người dùng và chuyển các yêu cầu tương ứng đến Lớp Business (BUS).
* Lớp Nghiệp Vụ (BUS): Đây là lớp chịu trách nhiệm nhận các yêu cầu từ Lớp GUI. Nó thực hiện các xử lý nghiệp vụ và sau đó truy xuất Lớp Truy Cập Dữ Liệu để lấy thông tin. Sau khi có kết quả, nó trả về thông tin này cho GUI để hiển thị cho người dùng.
* Lớp Truy Cập Dữ Liệu: Lớp này đóng vai trò làm giao tiếp với Cơ Sở Dữ Liệu (CSDL). Nó là lớp duy nhất được phép tương tác trực tiếp với cơ sở dữ liệu và thực hiện các truy vấn để lấy, cập nhật, hoặc xóa dữ liệu từ CSDL.
* (Lớp DTO - Không Bắt Buộc): Lớp này có thể được sử dụng như một lớp đơn giản để định nghĩa cấu trúc dữ liệu tương ứng với các bảng trong CSDL. Nó cung cấp một cách để biểu diễn dữ liệu khi truy vấn và lấy dữ liệu từ CSDL, giống như một dạng cơ bản của ánh xạ đối tượng-quan hệ.

Mỗi lớp trong kiến trúc này có một trách nhiệm cụ thể và tách biệt:

* GUI Layer: Tập trung vào hiển thị giao diện và tương tác người dùng.
* Business Layer (BUS): Thực hiện xử lý nghiệp vụ và tương tác với Lớp Truy Cập Dữ Liệu để lấy thông tin.
* Data Access Layer: Chịu trách nhiệm truy cập trực tiếp vào CSDL và thực hiện các thao tác liên quan đến dữ liệu.
* DTO Layer (Tùy Chọn): Nếu được sử dụng, nó giúp định nghĩa cấu trúc dữ liệu và cung cấp một lớp trung gian giữa CSDL và ứng dụng.
* Các lớp này tạo ra một kiến trúc lớp giúp tăng tính linh hoạt, tái sử dụng mã, và quản lý hiệu quả các thành phần khác nhau của ứng dụng.

Ưu điểm của mô hình 3 lớp:

* Phân loại rõ ràng các lớp: Mô hình này giúp phân chia các thành phần thành 3 lớp với nhiệm vụ riêng biệt, bao gồm Presentation Layer (giao diện người dùng), Business Layer (logic xử lý) và Data Access Layer (truy xuất dữ liệu). Việc phân loại rõ ràng này giúp quản lý và bảo trì dự án dễ dàng hơn.
* Quản lý hành động tại Business Layer: Bằng cách phân chia rõ ràng chức năng và nhiệm vụ tại Business Layer, mô hình 3 lớp giúp dễ dàng phân loại các hành động liên quan đến xử lý logic của ứng dụng. Điều này giúp giữ cho Business Layer sạch sẽ và dễ quản lý.
* Phân loại hàm truy xuất tại Data Access Layer: Mô hình 3 lớp cho phép phân loại các hàm truy xuất dữ liệu theo các bảng cơ sở dữ liệu tương ứng. Điều này giúp dễ dàng quản lý và tìm kiếm các hàm truy xuất dữ liệu khi cần thiết.
* Ứng dụng cho các dự án lớn: Mô hình 3 lớp được sử dụng phổ biến trong các dự án quy mô lớn. Điều này là do mô hình này giúp tạo cấu trúc rõ ràng, giúp quản lý mã nguồn và phát triển dự án một cách hiệu quả. [8]

Nhược điểm của mô hình 3 lớp:

* Đòi hỏi đầu tư nhiều thời gian và nỗ lực để nắm bắt, yêu cầu sự hiểu biết sâu rộng khi áp dụng và tổ chức file một cách đúng đắn.
* Không phù hợp cho việc phát triển các ứng dụng nhỏ, vì có thể tạo ra sự phức tạp không cần thiết.
* Dễ dàng bị phân tán, đặt ra thách thức trong việc duy trì sự nhất quán và hiệu suất của dự án.

**CHƯƠNG 2: PHÂN TÍCH THIẾT KẾ HỆ THỐNG PHẦN MỀM**

1. **Sơ đồ Usecase tổng quan:**

A diagram of a person's relationship

Description automatically generated

Hình 2. 1 Sơ đồ Usecase tổng quát

1. **Đặc tả usecase:**

**2.2.1 Đăng nhập:**

**A diagram of a diagram

Description automatically generated**

Hình 2. 2 Phân rã Usecase đăng nhập

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên UC** | **Đăng nhập** | | |
| Tác nhân | Nhân viên/Quản trị viên | | |
| Mô tả | Các actor sử dụng tài khoản và mật khẩu của mình để đăng nhập vào phần mềm | | |
|  |
| Điều kiện tiên quyết | Có tài khoản | | |  |
| Luồng sự kiện |  | **Tác nhân** | **Hệ thống phản hồi** |  |
| 1 | Truy cập phần mềm |  |  |
| 2 |  | Hiển thị form đăng nhập |  |
| 3 | Nhập thông tin |  |  |
| 4 | Nhấn nút đăng nhập |  |  |
| 5 |  | Hệ thống tiếp nhận thông tin gửi đên server database kiểm tra. Hợp lệ, hiển thị màn hình chức năng cho người dùng sử dụng |  |
| Luồng thay thế |  |  | Hệ thống báo sai thông tin đăng nhập yêu cầu nhập lại |  |
| Điều kiện thoát | Sau khi người dùng đăng nhập thành công; Người dùng chọn nút thoát | | |  |

Bảng 2. 1 Đặc tả usecase đăng nhập

**2.2.2 Bán hàng/ lập hóa đơn:**

**A diagram of a diagram

Description automatically generated**

Hình 2. 3 Phân rã Usecase bán hàng/lập hóa đơn

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên UC** | **Bán hàng/Lập hóa đơn** | | |
| Tác nhân | Nhân viên/Quản trị viên | | |
| Mô tả | Các tác nhân chọn sản phẩm bán cho khách hàng để lập hóa đơn thanh toán cho khách hàng | | |
|  |
| Điều kiện tiên quyết | Đã đăng nhập thành công vào hệ thống | | |  |
| Luồng sự kiện |  | **Tác nhân** | **Hệ thống phản hồi** |  |
| 1 | Đăng nhập thành công |  |  |
| 2 |  | Hệ thống hiện bản chọn chức năng |  |
| 3 | Nhấn nút trang chủ/order |  |  |
| 4 |  | Màn hình bán hàng/lập hóa đơn hiện ra |  |
| 5 |  | Tải danh sách sản phẩm từ database hiển thị giao diện phù hợp cho người dùng tương tác. |  |
| 6 | Chọn sản phẩm cần bán |  |  |
| 7 |  | Hiển thị hóa đơn thay đổi theo lựa chọn của tác nhân |  |
| 8 | Chọn thanh toán và xác nhận thanh toán |  |  |
|  | 9 |  | Cập nhật hóa đơn vào database |  |
| Luồng thay thế |  | | Nếu người dùng không xác nhận thanh toán thì trả về lại trang chủ |  |
| Điều kiện thoát | Sau khi người dùng đăng nhập thành công; Người dùng chọn nút thoát | | |  |

Bảng 2. 2 Đặc tả usecase bán hàng/lập hóa đơn

**2.2.3 Quản lý hóa đơn:**

**A diagram of a diagram

Description automatically generated**

Hình 2. 4 phân rã Usecase quản lý hóa đơn

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên UC** | **Quản lý hóa đơn** | | |
| Tác nhân | Nhân viên/Quản trị viên | | |
| Mô tả | Các tác nhân chọn xem danh sách hóa đơn | | |
|  |
| Điều kiện tiên quyết | Đã đăng nhập thành công vào hệ thống | | |  |
| Luồng sự kiện |  | **Tác nhân** | **Hệ thống phản hồi** |  |
| 1 | Đăng nhập thành công |  |  |
| 2 |  | Hệ thống hiện bản chọn chức năng |  |
| 3 | Nhấn nút lịch sử hóa đơn |  |  |
| 4 |  | Màn hình quản lý hóa đơn hiện ra |  |
| 5 |  | Tải danh sách hóa đơn từ database hiển thị bảng hóa đơn. |  |
| 6 | Chọn ngày tháng hiển thị danh sách theo ngày |  |  |
| 7 |  | Hiển thị hóa đơn theo lựa chọn |  |
| 8 | Chọn mã hóa đơn và chọn xóa để xóa hóa đơn khỏi danh sách |  |  |
|  | 9 |  | Cập nhật hóa đơn vào database |  |
| Luồng thay thế |  | | Nếu người dùng không xác nhận xóa thì trở lại |  |
| Điều kiện thoát | Sau khi người dùng đăng nhập thành công; Người dùng chọn chức năng khác | | |  |

Bảng 2. 3 Đặc tả usecase xóa hóa đơn đơn

**2.2.4 Quản lý sản phẩm:**

**A diagram of a diagram

Description automatically generated**

Hình 2. 5 Phân rã Usecase Quản lý sản phẩm

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên UC** | **Quản lý sản phẩm** | | |
| Tác nhân | Quản trị viên | | |
| Mô tả | Quản trị viên quản lý kho hàng | | |
|  |
| Điều kiện tiên quyết | Đã đăng nhập thành công vào hệ thống bằng tài khoản có quyền admin | | |  |
| Luồng sự kiện 1 |  | **Tác nhân** | **Hệ thống phản hồi** |  |
| 1 | Đăng nhập thành công |  |  |
| 2 |  | Hệ thống hiện bản chọn chức năng |  |
| 3 | Nhấn nút quản lý sản phẩm |  |  |
| 4 |  | Màn hình quản lý sản phẩm với các chức năng và bảng danh sách hiện ra |  |
| Luồng sự kiện 2 | 1 | Chọn button thêm |  |  |
| 2 |  | Cửa sổ biểu mẫu để nhập thông tin hàng mới hiện ra |  |
| 3 | Nhập thông tin hàng |  |  |
| 4 | Nhấn button thêm và xác nhận "có" |  |  |
| 5 |  | Cập nhật hàng mới vào database |  |
| Luồng sự kiện 3 | 1 | Chọn hàng trên bảng hàng |  |  |
| 2 | Chọn button sửa |  |  |
| 3 |  | Cửa sổ biểu mẫu để nhập thông tin hàng cần sửa hiện ra |  |
| 4 | Nhập thông tin cần sửa |  |  |
| 5 | Nhấn button thêm và xác nhận "có" |  |  |
| 6 |  | Cập nhật nhân viên mới vào database |  |
| Luồng sự kiện 4 | 1 | Chọn hàng trên bảng hàng |  |  |
| 2 | Nhấn button xóa và xác nhận có |  |  |
| 3 |  | Xóa hàng khỏi database |  |
| Luồng thay thế | Nếu người dùng không xác nhận có ở các luồng chính thì trở lại | | |  |
| Điều kiện thoát | Sau khi người dùng đăng nhập thành công; Người dùng chọn chức năng khác | | |  |

Hình 2. 6 Phân rã Usecase quản lý nhân viên

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên UC** | **Quản lý nhân viên** | | |
| Tác nhân | Quản trị viên | | |
| Mô tả | Quản trị viên quản lý nhân sự thêm/xóa/sửa và quản lý tài khoản | | |
|  |
| Điều kiện tiên quyết | Đã đăng nhập thành công vào hệ thống bằng tài khoản có quyền admin | | |  |
| Luồng sự kiện 1 |  | **Tác nhân** | **Hệ thống phản hồi** |  |
| 1 | Đăng nhập thành công |  |  |
| 2 |  | Hệ thống hiện bản chọn chức năng |  |
| 3 | Nhấn nút quản lý nhân viên |  |  |
| 4 |  | Màn hình quản lý nhân viên với các chức năng và bảng danh sách nhân viên hiện ra |  |
| Luồng sự kiện 2 | 1 | Chọn button thêm |  |  |
| 2 |  | Cửa sổ biểu mẫu để nhập thông tin hàng mới hiện ra |  |
| 3 | Nếu đã có tài khoản nhập thông tin nhân viên |  |  |
| 4 | Nhấn button thêm và xác nhận "có" |  |  |
| 5 |  | Cập nhật nhân viên mới vào database |  |
| Luồng sự kiện 3 | 1 | Chọn hàng trên bảng hàng |  |  |
| 2 | Chọn button sửa |  |  |
| 3 |  | Cửa sổ biểu mẫu để nhập thông tin hàng cần sửa hiện ra |  |
| 4 | Nhập thông tin cần sửa |  |  |
| 5 | Nhấn button thêm và xác nhận "có" |  |  |
| 6 |  | Cập nhật nhân viên mới vào database |  |
| Luồng sự kiện 4 | 1 | Chọn nhân viên trên bảng hàng |  |  |
| 2 | Nhấn button xóa và xác nhận có |  |  |
| 3 |  | Xóa nhân viên khỏi database |  |
| Luồng thay thế | Nếu chưa có tài khoản thì đưa về người dùng đến cửa sổ xem danh sách tài khoản để thêm hoặc xóa tài khoản | | |  |
| Điều kiện thoát | Sau khi người dùng đăng nhập thành công; Người dùng chọn chức năng khác | | |  |

Bảng 2. 4 Đặc tả usecase quản lý hàng hóa

**2.2.5 Quản lý nhân viên:**

**A diagram of a diagram

Description automatically generated**

Bảng 2. 5 Đặc tả usecase quản lý nhân viên

**2.2.6 Quản lý doanh thu:**

A diagram of a diagram

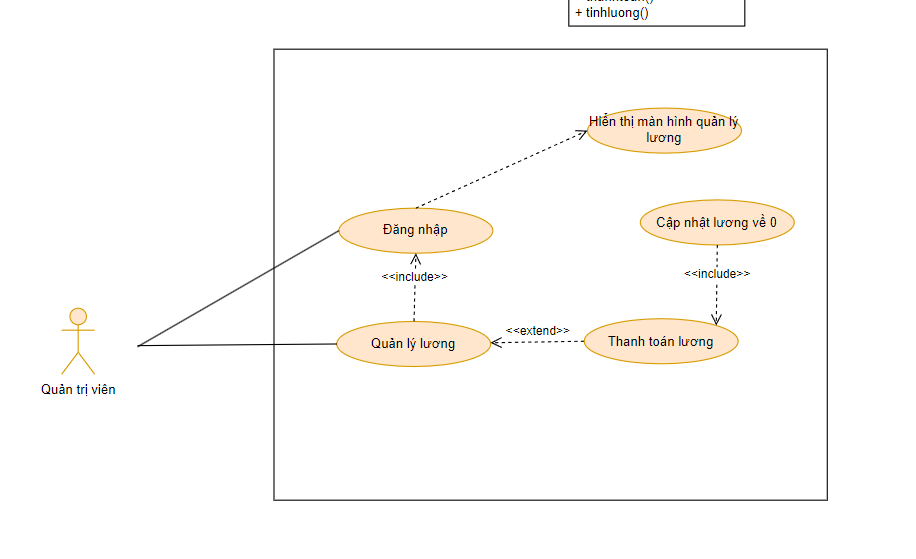
Description automatically generated

Hình 2. 7 Phân rã Usecase quản lý doanh thu

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên UC** | **Quản lý doanh thu** | | |
| Tác nhân | Quản trị viên | | |
| Mô tả | Quản lý doanh thu kiểm tra lợi nhuận, kiểm tra thu chi bằng biểu đồ trực quan | | |
|  |
| Điều kiện tiên quyết | Đã đăng nhập thành công vào hệ thống bằng tài khoản quản trị viên | | |  |
| Luồng sự kiện |  | **Tác nhân** | **Hệ thống phản hồi** |  |
| 1 | Đăng nhập thành công |  |  |
| 2 |  | Hệ thống hiện bản chọn chức năng |  |
| 3 | Chọn chức năng thống kê |  |  |
| 4 |  | Màn hình quản lý doanh thu hiện ra |  |
| 5 | Chọn ngày "từ" "đến" để xem thống kê giữa những ngày đó |  |  |
| 6 | Nhấn button hiển thị |  |  |
| 7 |  | Lợi nhuận, hóa đơn tổng,biểu đồ cột tròn hiện ra. |  |
| Điều kiện thoát | Sau khi người dùng đăng nhập thành công; Người dùng chọn chức năng khác | | |  |

Bảng 2. 6 Đặc tả usecase quản lý doanh thu

### **2.2.7 Quản lý lương:**

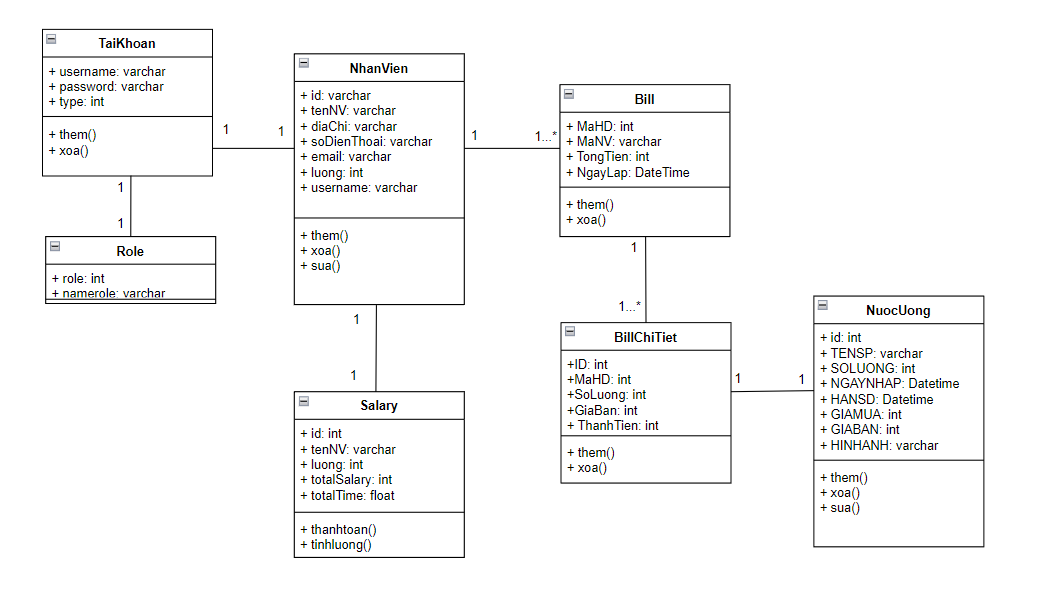


Hình 2. 8 Phân ra UseCase quản lý lương

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên UC** | **Quản lý lương** | | |
| Tác nhân | Quản trị viên | | |
| Mô tả | Quản lý lương kiểm tra tổng lương, tổng thời gian làm việc của nhân viên. | | |
| Điều kiện tiên quyết | Đã đăng nhập thành công vào hệ thống bằng tài khoản có quyền admin | | |  |
| Luồng sự kiện 1 |  | **Tác nhân** | **Hệ thống phản hồi** |  |
| 1 | Đăng nhập thành công |  |  |
| 2 |  | Hệ thống hiện bảng chọn chức năng |  |
| 3 | Nhấn nút quản lý lương |  |  |
| 4 |  | Màn hình quản lý lương với các chức năng và bảng danh sách hiện ra |  |
| Luồng sự kiện 2 | 1 | Chọn button “thanh toán” |  |  |
| 2 |  | Cửa sổ xác nhận thanh toán lương hiện ra |  |
| 4 | Nhấn button thêm và xác nhận "có" |  |  |
| 5 |  | Cập nhật lương nhân viên về 0 |  |
| Luồng thay thế | Nếu người dùng không xác nhận có ở các luồng chính thì trở lại | | |  |
| Điều kiện thoát | Sau khi người dùng đăng nhập thành công; Người dùng chọn chức năng khác | | |  |

Bảng 2. 7 Bảng đặc tả Usecase Quản lý lương

1. **Sơ đồ lớp:**



Hình 2. 9 Sơ đồ lớp

1. **: Sơ đồ tuần tự:**

**2.4.1 Đăng nhập:**

**A diagram of a diagram

Description automatically generated**

Hình 2. 10 Sơ đồ tuần tự đăng nhập

**2.4.2 Bán hàng/ lập hóa đơn:**

**A diagram of a project

Description automatically generated**

Hình 2. 11 Sơ đồ tuần tự bán/lập hóa đơn

**2.4.3 Quản lý hóa đơn:**

**- Xóa hóa đơn:**

**A diagram of a diagram

Description automatically generated**

Hình 2. 12 Sơ đồ tuần tự xóa hóa đơn

**2.4.4 Quản lý sản phẩm:**

**2.4.4.1 Thêm sản phẩm:**

A diagram of a product

Description automatically generated

Hình 2. 13 Sơ đồ tuần tự thêm sản phẩm

**2.4.4.2 Sửa thông tin sản phẩm:**

**A diagram of a product

Description automatically generated**

Hình 2. 14 Sơ đồ tuần tự sửa sản phẩm

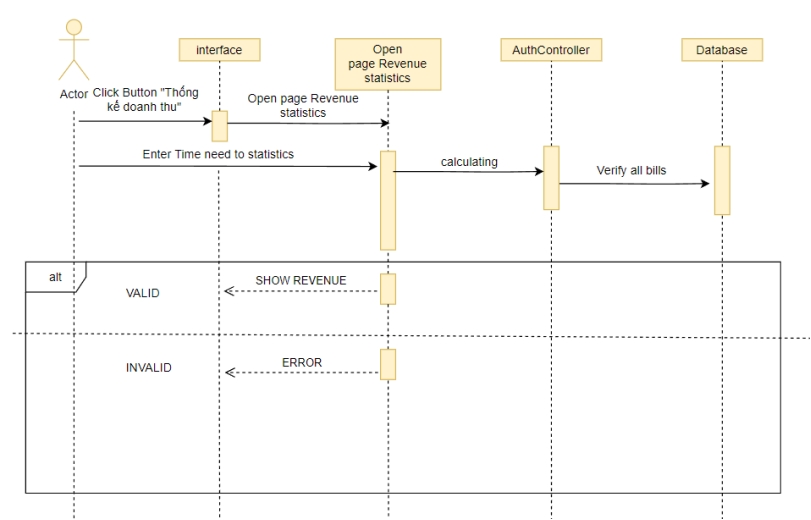
**2.4.4.3 Xóa sản phẩm:**

A diagram of a product

Description automatically generated

Hình 2. 15 Sơ đồ tuần tự xóa sản phẩm

**2.4.4.4 Thống kê doanh thu:**



Hình 2. 16 Sơ đồ tuần tự thống kê doanh thu

**2.4.5 Quản lý nhân viên:**

**2.4.5.1 Thêm nhân viên:**

A diagram of a software project

Description automatically generated

Hình 2. 17 Sơ đồ tuần tự thêm nhân viên

**2.4.5.2 Sửa thông tin nhân viên:**

A diagram of a process

Description automatically generated

Hình 2. 18 Sơ đồ tuần tự sửa nhân viên

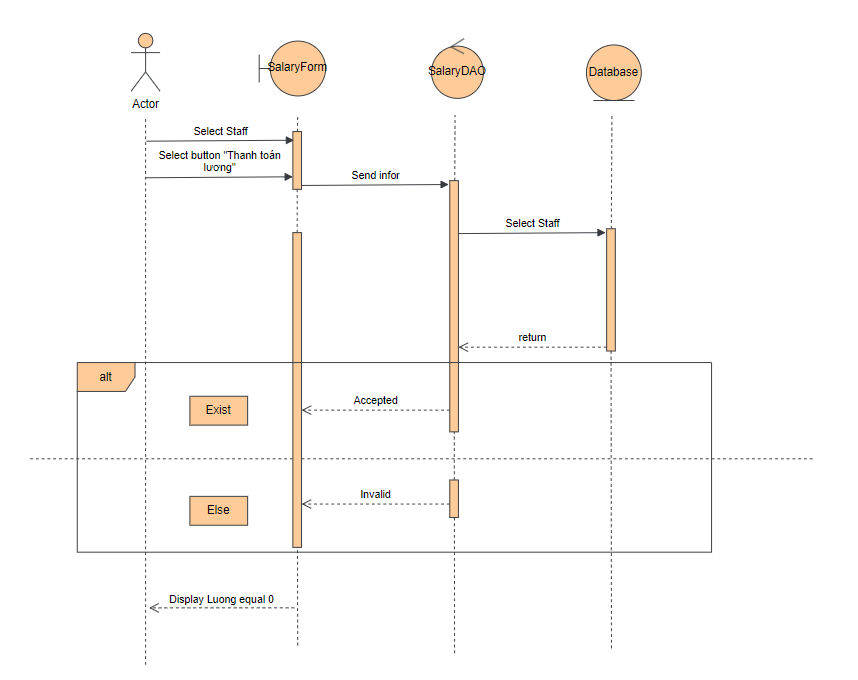
**2.4.5.3 Xóa nhân viên:**

A diagram of a software process

Description automatically generated

Hình 2. 19 Sơ đồ tuần tự xóa nhân viên

### **2.4.6 Quản lý lương:**



Hình 2. 20 Sơ đồ tuần tự thanh toán lương

1. **Thiết kế CSDL:**

**2.5.1 Bảng nRole:**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **STT** | **Tên thuộc tính** | **Kiểu dữ liệu** | **Kích thước** | **Not null/ Null** | **Khóa chính** | **Khóa ngoại** |
| 1 | id | int |  | X | X |  |
| 2 | nameRole | varchar | 100 | X |  |  |

Bảng 2. 8 Bảng dữ liệu nRole

**2.5.2 Bảng Account:**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **STT** | **Tên thuộc tính** | **Kiểu dữ liệu** | **Kích thước** | **Not null/ Null** | **Khóa chính** | **Khóa ngoại** |
| 1 | username | varchar | 100 | X | X |  |
| 2 | password | varchar | 1000 |  |  |  |
| 3 | type | int |  |  |  | X |

Bảng 2. 9 Bảng dữ liệu Account

**2.5.3 Bảng NhanVien:**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **STT** | **Tên thuộc tính** | **Kiểu dữ liệu** | **Kích thước** | **Not null/ Null** | **Khóa chính** | **Khóa ngoại** |
| 1 | id | varchar | 100 | X | X |  |
| 2 | tenNV | varchar | 255 | X |  |  |
| 3 | diaChi | varchar | MAX |  |  |  |
| 4 | soDienThoai | varchar | 20 |  |  |  |
| 5 | email | varchar | 225 |  |  |  |
| 6 | luong | int |  | X |  |  |
| 7 | username | varchar | 100 | X |  | X |

Bảng 2. 10 Bảng dữ liệu NhanVien

**2.5.4 Bảng Bill:**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **STT** | **Tên thuộc tính** | **Kiểu dữ liệu** | **Kích thước** | **Not null/ Null** | **Khóa chính** | **Khóa ngoại** |
| 1 | MaHD | int |  | X | X |  |
| 2 | MaNV | varchar | 100 | X |  | X |
| 3 | TongTien | int |  |  |  |  |
| 4 | NgayLap | datetime |  | X |  |  |

Bảng 2. 11 Bảng dữ liệu Bill

**2.5.5 Bảng BillChiTiet:**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **STT** | **Tên thuộc tính** | **Kiểu dữ liệu** | **Kích thước** | **Not null/ Null** | **Khóa chính** | **Khóa ngoại** |
| 1 | ID | int |  | X |  | X |
| 2 | MaHD | int |  | X |  | X |
| 3 | SoLuong | int |  | X |  |  |
| 4 | GiaBan | int |  | X |  | X |
| 5 | ThanhTien | int |  | X |  |  |

Bảng 2. 12 Bảng dữ liệu BillChiTiet

**2.5.6 Bảng NuocUong:**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **STT** | **Tên thuộc tính** | **Kiểu dữ liệu** | **Kích thước** | **Not null/ Null** | **Khóa chính** | **Khóa ngoại** |
| 1 | id | int |  | X |  | X |
| 2 | TENSP | varchar | 100 | X |  |  |
| 3 | SOLUONG | int |  | X |  |  |
| 4 | NGAYNHAP | DateTime |  | X |  |  |
| 5 | HANSD | DateTime |  | X |  |  |
| 6 | GIAMUA | int |  | X |  |  |
| 7 | GIABAN | int |  | X |  |  |
| 8 | HINHANH | varchar | MAX |  |  |  |

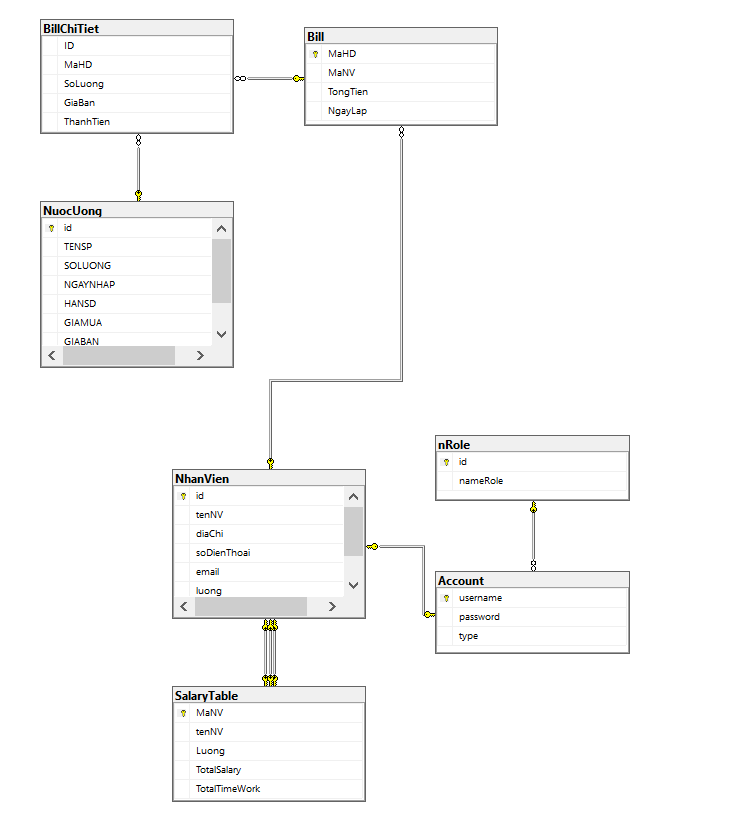
Bảng 2. 13 Bảng dữ liệu NuocUong

### **2.5.7 Bảng Salary:**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **STT** | **Tên thuộc tính** | **Kiểu dữ liệu** | **Kích thước** | **Not null/ Null** | **Khóa chính** | **Khóa ngoại** |
| 1 | id | int |  | X |  | X |
| 2 | tenNV | varchar | 100 | X |  | X |
| 3 | Luong | int |  | X |  | X |
| 4 | TotalSalary | int |  |  |  |  |
| 5 | TotalWorkTime | float |  |  |  |  |

Bảng 2. 14 Bảng dữ liệu Salary

Sơ đồ Database:



Hình 2. 21 Sơ đồ CSDL

**CHƯƠNG 3: XÂY DỰNG PHẦN MỀM:**

**3.1 Xây dựng Solution:**

Gồm 1 project và 3 file được tạo

A screenshot of a computer

Description automatically generated

Hình 3. 1 Cấu trúc Solution

* PresentLayer tất cả source code của dự án và form giao diện của dự án
* DAO (Data Access Object): được sử dụng chứa các lớp thực hiện các thao tác liên quan đến truy cập và tương tác với cơ sở dữ liệu. Nhiệm vụ chính của DAO là cung cấp một giao diện chung để truy cập dữ liệu mà không làm ảnh hưởng đến mã nguồn giao diện người dùng (User Interface Layer).

A screenshot of a computer

Description automatically generated

Hình 3. 2 File DAO

* DTO (Data Transfer Object): được sử dụng để chuyển dữ liệu giữa các lớp khác nhau trong ứng dụng, đóng gói dữ liệu và chuyển nó giữa các lớp một cách hiệu quả, thường được sử dụng để truyền dữ liệu giữa lớp Logic và lớp Truy cập Dữ liệu trong kiến trúc 3 lớp.  
  A black rectangular object with white text

  Description automatically generated

Hình 3. 3 File DTO

* File Form login được đặt riêng vì không chứa các quan hệ kế thừa hay nằm trong bất kỳ form tác vụ nào để tránh các lỗi không mong muốn xảy ra. Còn lại là các Form và user-control tự tạo. A screenshot of a computer

  Description automatically generated

Hình 3. 4 Cấu trúc các form

**3.2 Phân tích giao diện phần mềm:**

**3.2.1 Form đăng nhập:**

A screenshot of a login screen

Description automatically generated

Hình 3. 5 Form đăng nhập

Form đăng nhập gồm:

* 2 Text box username và password cho người dùng nhập thông tin tài khoản và mật khẩu
* 1 button login để đăng nhập sau khi nhập xong

Source code button login:

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

{

string userName = textBox1.Text;

string passWord = textBox2.Text;

if (Login(userName, passWord))

{

frmLogin frmLogin = new frmLogin();

string tenNV = AccountDAO.Instance.GetTenNhanvien(userName);

string id = AccountDAO.Instance.GetIdNhanvien(userName);

int userType = AccountDAO.Instance.GetUserType(userName);

if (userType == 0)

{

this.Hide();

frmMain frm = new frmMain();

frm.HienthiNhanVien(tenNV);

frm.HienthiIdNhanVien(id);

frm.SetLoginForm(this

frm.Show();

}

else

{

this.Hide();

frmMain frm = new frmMain();

frm.HienthiNhanVien(tenNV);

frm.HienthiIdNhanVien(id);

frm.BlockTabAdmin();

frm.SetLoginForm(this

frm.Show();

}

}

else

{

MessageBox.Show("Sai tên tài khoản hoặc mật khẩu!",

"Thông báo",

MessageBoxButtons.OK,

MessageBoxIcon.Error);

}

Giải thích: 2 biến chuỗi userName và passWord lấy dữ liệu string từ 2 textbox đưa vào hàm GetTenNhanVien và GetUserType từ lớp truy cập CSDL AccountDAO -> xét điều kiện để phân quyền truy cập mở các form chức năng.

Khi người dùng nhập sai thông tin tài khoản hiện một message box thông báo “Sai tên tài khoản hoặc mật khẩu!”:   
A screenshot of a login screen

Description automatically generated

Hình 3. 6 Thông báo đăng nhập sai

**3.2.2 Form Main:**

**A hands holding a cup of coffee

Description automatically generated**

A blue and black logo

Description automatically generated

Hình 3. 7 Form main

Sau khi đăng nhập phần mềm sẽ chạy một panel chào mừng người dùng;

Form main gồm:

* 4 button:

+ Trang chủ/Order: Mở Form Home;

+ Lịch sử hóa đơn: Mở Form Bill;

+ Quản lý tác vụ: Mở Form Manage;

+ Đăng xuất để thoát Form main và trở về Form đăng nhập.

* 3 label thể hiện thông tin nhân viên và mã nhân viên trực ca.
* 1 timer để hiện thời gian thực trên Form main.

Khi click vào đăng xuất một messagebox hiện ra xác nhận đăng xuất:

A computer screen shot of a computer screen

Description automatically generated

Hình 3. 8 Xác nhận đăng xuất

Source code button Đăng xuất:

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

{

DialogResult cc = MessageBox.Show(

"Xác nhận đăng xuất khỏi hệ thống?",

"Thông báo",

MessageBoxButtons.YesNo,

MessageBoxIcon.Question

);

if (cc == DialogResult.Yes)

{

// Xóa ký tự trong TextBox khi nhấn btnLogout

if (loginForm != null)

{

loginForm.ClearTextBoxes();

}

if (loginForm == null || loginForm.IsDisposed)

{

loginForm = new frmLogin();

loginForm.Show();

}

else

{

loginForm.Show();

}

// Đóng frmMain

this.Close();

}

else if (cc == DialogResult.No) { }

}

Source code button Trang chủ/Order:

//Button Trang chủ

private void button1\_Click(object sender, EventArgs e)

{

OpenChildForm(homeform);

if (homeform == null || homeform.IsDisposed)

{

homeform = new frmHome();

homeform.LaytenThuNgan(lblTenNhanVienTrucCa.Text);

homeform.LayIdthuNgan(lblMaNhanVienTrucCa.Text);

homeform.MdiParent = this;

homeform.Show();

}

else

{

homeform.Show();

}

pictureBoxlogo.Dispose();

}

**3.2.3 Form home:**

A screenshot of a computer

Description automatically generated

Hình 3. 9 Form Home

Form home là form con của Form main nằm trong nó gồm :

* Một Group box chứa menu các loại cà phê;
* Một panel chứa form hóa đơn;
* 3 button :

+ Button Clear: xóa hàng hóa đơn từng hàng;

+ Button Clear All: Xóa tất cả sản phẩm hiện có trên hóa đơn;

+ Thanh toán để thanh toán hóa đơn;

+Button “Tải lại” để reload lại menu.

Form hóa đơn khi có hàng sẽ tạo một gridview chứa các món đang được chọn để thanh toán:

A screenshot of a computer

Description automatically generated

Hình 3. 10 Form hóa đơn

Khi nhấn vào thanh toán một messagebox xác nhận thanh toán hiện ra:

A screenshot of a computer

Description automatically generated

Hình 3. 11 Xác nhận thanh toán

Sau khi chọn “Yes” một messagebox thông báo thành công hiện ra:

A screenshot of a computer

Description automatically generated

Hình 3. 12 Thông báo thanh toán thành công

Sau đó một messagebox in hóa đơn hiện ra:

A screenshot of a computer

Description automatically generated

Hình 3. 13 Xác nhận in hóa đơn

Hóa đơn được in:

A screenshot of a computer screen

Description automatically generated

Hình 3. 14 Hóa đơn sau khi in

Source button Thanh toán:

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

{

bool exceptionOccurred = false;

int MaHD = 0;

DialogResult cc = MessageBox.Show(

"Xác nhận thanh toán?",

"Thông báo",

MessageBoxButtons.YesNo,

MessageBoxIcon.Question);

if (cc == DialogResult.Yes)

{

try

{

string nguoiThuNgan = txtMaNvThuNgan.Text;

int maHD;

BillDAO.Instance.CreateBill(out maHD, nguoiThuNgan);

Console.WriteLine("MaHD sau khi tạo: " + maHD);

MaHD = maHD;

// Lấy thông tin từ GridView và thêm vào BillChiTiet

foreach (DataGridViewRow row in dgvHoaDon.Rows)

{

int id = Convert.ToInt32(row.Cells["id"].Value);

int soLuong = Convert.ToInt32(row.Cells["Soluong"].Value);

int thanhTien = 0;

BillDAO.Instance.CreateBillChiTiet(id, maHD, soLuong, out thanhTien);

row.Cells["ThanhTien"].Value = thanhTien;

}

MessageBox.Show("Thanh toán thành công!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Information);

}

catch (Exception ex)

{

BillDAO.Instance.DeleteBill(MaHD);

exceptionOccurred = true;

MessageBox.Show("Số lượng hàng còn lại không đủ!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Error);

}

if (!exceptionOccurred)

{

// Các dòng lệnh muốn thực hiện dù có exception hay không

NgayLap();

DialogResult result = MessageBox.Show("Có muốn in hóa đơn không?", "Thông báo", MessageBoxButtons.YesNo, MessageBoxIcon.Question);

if (result == DialogResult.Yes)

{

PrintPanel(pnHoaDon);

MessageBox.Show("In hóa đơn thanh toán thành công!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Information);

}

else

{

ResetformHoadon();

}

}

}

else

{

}

}

**3.2.4 Form Bill:**

A screenshot of a computer

Description automatically generated

Hình 3. 15 Form Bill

Form Bill bao gồm:

* 2 gridview chứa Hóa đơn và chi tiết của hóa đơn;
* 2 button Hiển thị và xóa;
* 2 DataTimePicker : để biểu diễn thời gian;
* Một text box để lấy mã hóa đơn.

Bảng dữ liệu trả sau khi click hiển thị:

Ví dụ: chọn xem danh sách hóa đơn từ 19-11-2023 đến 20-11-2023:

A screenshot of a computer

Description automatically generated

Hình 3. 16 Lọc danh sách hóa đơn

**3.2.5 Form Manage:**

A screenshot of a computer

Description automatically generated

Hình 3. 17 Form mange

Form manage gồm:

* 3 tab thống kê, nhân viên, kho hàng;
* 3 panel chứa: tổng hóa đơn, tổng tiền hóa đơn, lợi nhuận;
* 2 Datatimepicker và 1 button hiển thị

**3.2.5.1 Tab thống kê:**

Khi chọn ngày để hiển thị một các panel thể hiện các thông số hiện lên, biểu đồ thống kê gồm biểu đồ cột và biểu đồ tròn:

Ví dụ từ ngày 17-11-2023 đến ngày 20-11-2023:

A screenshot of a computer

Description automatically generated

Hình 3. 18 Tab thống kê

**3.2.5.2 Tab nhân viên:**

A screenshot of a computer

Description automatically generated

Hình 3. 19 Tab nhân viên

Gồm 6 textbox chứa các thông tin của nhân viên 1 numeric lương và một text box tìm kiếm nhân viên:

* Gridview show thông tin nhân viên;
* 6 button gồm:

+ Thêm: thêm nhân viên lấy dữ liệu từ text box trên;

+ Sửa: thông tin có trên text box;

+ Xóa: xóa dữ liệu có trên text box;

+ Tìm: Tìm nhân viên theo tên trên text box;

+ Thêm tài khoản: quản lý tài khoản:

A screenshot of a login screen

Description automatically generated

Hình 3. 20 Form thêm tài khoản

A screenshot of a computer

Description automatically generated

Hình 3. 21 Form Danh sách tài khoản

+ Reset: reset gridview và các thông tin hiện hữu trên text box.

**3.2.5.3 Tab kho hàng:**

A screenshot of a computer

Description automatically generated

Hình 3. 22 Tab kho hàng

Tab kho hàng gồm:

* 1 gridview data kho hàng
* 2 text box thể hiện giá trị đang được chọn và 1 text box tìm kiếm
* 5 button:

+ Thêm hàng: Khi click vào một biểu mẫu thêm hàng được hiện ra bên dưới

A screenshot of a computer

Description automatically generated

Hình 3. 23 Form Thêm sản phẩm

+ Sửa thông tin: Khi click vào biểu mẫu sửa thông tin hiện ra:

A screenshot of a computer

Description automatically generated

Hình 3. 24 Form sửa sản phẩm

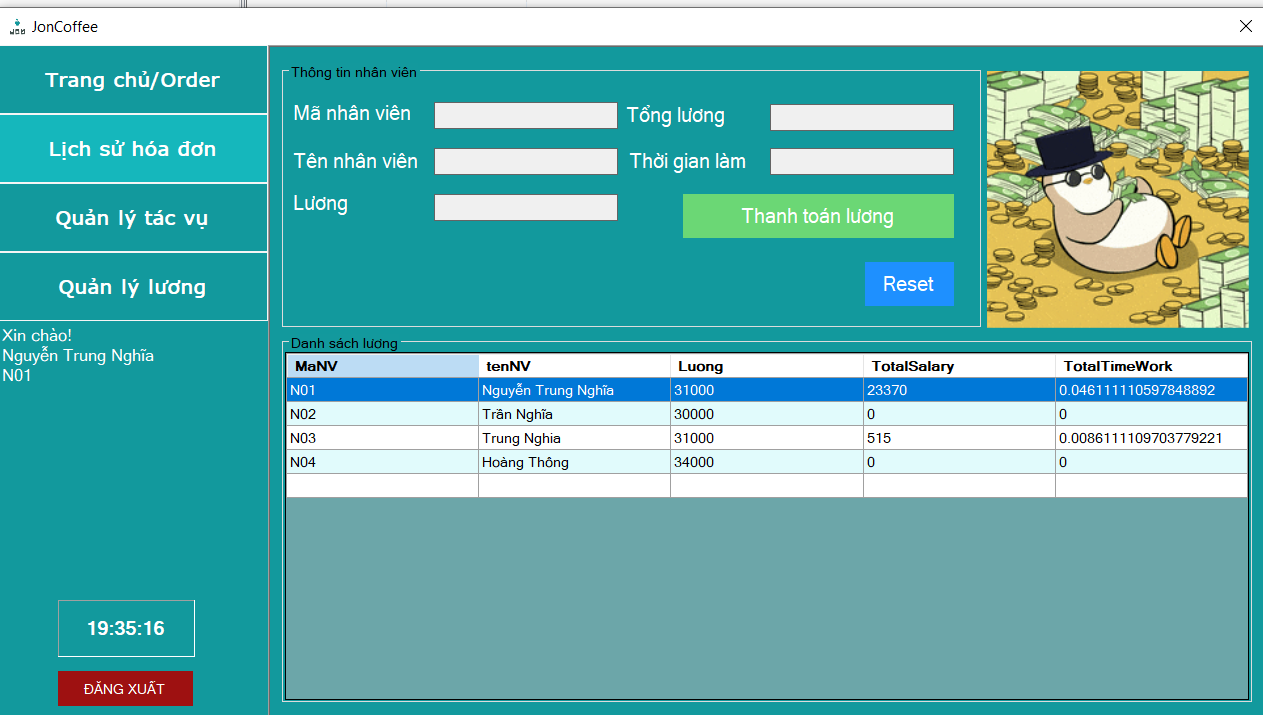
+ Xóa hàng: khi click một messagebox xác nhận xóa hàng:

A screenshot of a computer

Description automatically generated

Hình 3. 25 Xác nhận xóa hàng

### **3.2.6 Form Salary:**



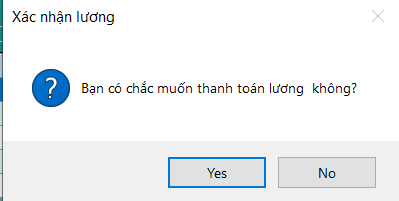
Hình 3. 26 Form Salary

Form Salary gồm:

- 5 textBox để hiện thị dữ liệu nhân viên được chọn và không thể thay đổi giá trị;

- 2 button:

+ Thanh toán lương: khi chọn gửi về một xác nhận thanh toán:



Hình 3. 27 Xác nhận lương

+ Reset: Xóa các giá trị của các textBox;

Source code button “Thanh toán lương”:

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

{

DialogResult result = MessageBox.Show("Bạn có chắc muốn thanh toán lương " + txtTenNV.Text + " không?", "Xác nhận lương", MessageBoxButtons.YesNo, MessageBoxIcon.Question);

string maNV = txtidNV.Text;

if (result == DialogResult.Yes)

{

if (SalaryDAO.Instance.ResetSalaryToZero(maNV))

{

MessageBox.Show("Đã trả lương cho " + txtTenNV.Text + " thành công.", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Information);

LoadAllSalary();

}

else

{

MessageBox.Show("Vui lòng chọn một nhân viên muốn thanh toán bảng bên dưới!", "Thông báo", MessageBoxButtons.OK, MessageBoxIcon.Error);

}

}

}

**KẾT LUẬN**

* **Kết quả đạt được:**

- Giao diện thân thiện hiện đại, thiết kế có tính khoa học đặc trưng theo màu sắc riêng của thương hiệu;

- Tốc độ phản hồi đáp ứng yêu cầu trong quá trình kinh doanh

- Ứng dụng được các kĩ thuật lập trình để giải quyết các bài toán các logic trong quá trình phát triển phần mềm, ví dụ như: Exception, Async/Await, Stored Procedure,... tránh các bug các lỗi như SQL injection.

- Học hỏi được nhiều mô hình xây dựng phát triển, trải nghiệm được nhiều UI để chọn lựa Winform là phiên bản cuối cùng để thực hiện;

- Hiểu sâu hơn về ngôn ngữ C# cũng như cách sử dụng các công cụ MS SQL Sever, Visual studio, Github,...

* **Hạn chế:**

- Cách cấu trúc file chưa tốt chưa tối ưu được mô hình 3 lớp;

- Chưa có các chức năng:

+ Thanh toán trực tuyến;

+ Chương trình khuyến mãi,...

* **Hướng phát triển:**

- Tìm ra thêm nhiều lỗi trong quá trình sử dụng để cải tiến cho phần mềm ngày một hoàn thiện

- Thêm những chức năng như: thanh toán trực tiếp, chương trình khuyến mãi, chat nội bộ,...

- Nâng cao trải nghiệm người dùng trong các hoạt ảnh chuyển tranh, thêm các chi tiết hoạt ảnh sinh động làm cho phần mềm hiện đại hơn;

- Responsive cho phần mềm tối ưu hơn trên mọi loại màn hình thiết bị

**TÀI LIỆU THAM KHẢO**

|  |  |
| --- | --- |
| [1] | H. N. D. -. P. s. C. Thương, “Phát triển thị trường tiêu thụ cà phê nội địa là xu hướng tất yếu trong giai đoạn hiện nay,” *SỞ CÔNG THƯƠNG ĐẮK LẮK,* 2023. |
| [2] | congdongnhahang.vn, *Sơ đồ quản lý quán cafe là gì? Các bộ phận trong sơ đồ quản lý quán cafe,* 2022. |
| [3] | cukcuk.vn/, *Những điều cần biết về kinh doanh quán ăn theo mô hình tự phục vụ,* 2022. |
| [4] | J. Friedman, *When Was C# Created? A Brief History,* 18/3/2020. |
| [5] | R. Peterson, “Guru99,” 11 10 2023. [Trực tuyến]. Available: https://www.guru99.com/sql-server-introduction.html. |
| [6] | K. C. v. T. thông, “cntt.hdu.edu.vn,” trường đại học Hồng Đức, 22 8 2022. [Trực tuyến]. Available: http://cntt.hdu.edu.vn/cac-phien-ban-ms-sql-server.html. |
| [7] | Microsoft, “Microsoft,” learn.microsoft.com, 24 10 2023. Available: https://learn.microsoft.com/en-us/visualstudio/get-started/visual-studio-ide?view=vs-2022. |
| [8] | N. L. Minh, “Mô hình 3 lớp là gì? Các thành phần chính của mô hình 3 lớp,” 13 6 2023. |