|  |  |
| --- | --- |
|  |  |
| TRẦN HỮU VINH | BỘ CÔNG THƯƠNG  TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP HÀ NỘI  --------------------------------------- |
|  |
| ĐỒ ÁN TỐT NGHIỆP ĐẠI HỌC |
| NGÀNH CÔNG NGHỆ THÔNG TIN |
| XÂY DỰNG PHẦN MỀM QUẢN LÝ BÁN HÀNG MẶT HÀNG  ĐỒ TRANG TRÍ TRÊN HỆ ĐIỀU HÀNH WINDOWS |
|  |
|  |
| CBHD :TS.Vũ Đình Minh |
| CÔNG NGHỆ THÔNG TIN | Sinh viên : Trần Hữu Vinh |
| Mã số sinh viên: 2019601953 |
|  |
|  |
|  |
| Hà Nội – Năm 2024 |
|  |

# LỜI CẢM ƠN

Sau 4 năm học tập và rèn luyện tại Trường Đại học Công Nghiệp Hà Nội, đồ án tốt nghiệp này là một dấu ấn quan trọng đánh dấu việc em - một sinh viên đã hoàn thành nhiệm vụ của mình trên ghế giảng đường Đại học. Đầu tiên, với tình cảm sâu sắc, chân thành, cho phép em được bày tỏ lòng biết ơn sâu sắc đến Gia đình, các anh chị, bạn bè đã luôn động viên, khích lệ, tạo điều kiện giúp đỡ em trong suốt quá trình học tập và nghiên cứu đề tài.

Em cũng xin gửi tới các thầy các cô khoa Công Nghệ Thông Tin Trường Đại học Công Nghiệp Hà Nội lời chào trân trọng, lời chúc sức khỏe và lời cảm ơn sâu sắc. Với sự quan tâm, dạy dỗ, chỉ bảo tận tình chu đáo của thầy cô, đến nay em đã có thể hoàn thành đồ án tốt nghiệp, đề tài: “Xây dựng phần mềm quản lý bán hàng mặt hàng đồ trang trí trên hệ điều hành Windows”.

Đặc biệt em xin gửi lời cảm ơn chân thành đến thầy giáo - TS.Vũ Đình Minh người đã tận tình giúp và hướng dẫn em hoàn thành đề tài đồ án.

Đồng thời, em xin bày tỏ lòng biết ơn đến lãnh đạo Trường Đại học Công Nghiệp Hà Nội, các Khoa, Phòng ban chức năng đã tạo điều kiện cho em được học tập tại nơi mà em yêu thích, cho em bước vào đời sống thực tế và áp dụng những kiến thức em đã học tại trường và môi trường làm việc mới của em. Qua quá trình học tập em đã tích lũy được rất nhiều kiến thức để chuẩn bị cho công việc sau này cũng như để phát triển thêm bản thân.

Trong quá trình hoàn thành đồ án tốt nghiệp không thể tránh khỏi thiếu sót, kính mong có sự góp ý từ thầy cô.

*Em xin chân thành cảm ơn!*

Trần Hữu Vinh

# LỜI NÓI ĐẦU

Trong thời đại số hóa ngày nay, việc tích hợp công nghệ thông tin vào trong các lĩnh vực và ngành nghề khác đã và đang được coi là xu hướng của tương lai bởi vì tính tiện lợi và năng suất mà nó mang lại. Bên cạnh đó, gia đình em cũng đang quản lý kinh doanh các mặt hàng đồ trang trí theo cách truyền thống, tuy nhiên, điều này khá bất tiện vì việc tính nhầm sổ sách hay thực sự gặp khó khăn khi cần tìm những ghi chép cũ. Chính điều này đã thúc đẩy em tập trung nghiên cứu và phát triển một trang phần mềm riêng, với mục tiêu tạo ra một giao diện thân thiện, dễ sử dụng để hỗ trợ cho chính công việc trong gia đình mình.

Đồ án của em tập trung vào việc xây dựng một hệ thống truy cứu thông tin nhanh nhạy, chính xác và dễ dàng tiếp cận cho người dùng, đi kèm với các dịch vụ (service) tăng tính tự động hóa nhằm giảm thiểu nỗ lực của người dùng nhưng vẫn đem lại kết quả vượt trội.

Em đã tiến hành nghiên cứu sâu về yêu cầu, xây dựng cấu trúc hệ thống và triển khai các công nghệ tiên tiến để đảm bảo tính linh hoạt, an toàn và hiệu quả cho phần mềm.

Trong báo cáo này, em sẽ trình bày về quá trình phát triển, kiến trúc hệ thống, chi tiết các chức năng được tích hợp và cách thức kiểm thử, đánh giá hiệu suất của sản phẩm.

Để hoàn thành được đồ án tốt nghiệp này, em xin được gửi lời cảm ơn chân thành đến các thầy cô trong khoa Công nghệ thông tin Trường Đại học Công Nghiệp Hà Nội đã tận tình giảng dạy và trang bị kiến thức cho em trong suốt thời gian em học tập tại trường. Ngoài ra, em xin cám ơn Thầy giáo trực tiếp hướng dẫn đề tài này – Tiến Sĩ Vũ Đình Minh, giảng viên Khoa Công nghệ thông tin Trường Đại học Công Nghiệp Hà Nội – đã tận tụy hết lòng chỉ dẫn và hỗ trợ tận tình để giúp em hoàn thành được đồ án.

Mục Lục

[LỜI CẢM ƠN 2](#_Toc154005338)

[LỜI NÓI ĐẦU 3](#_Toc154005339)

[DANH MỤC CÁC CHỮ VIẾT TẮT 8](#_Toc154005340)

[DANH MỤC BẢNG 8](#_Toc154005341)

[DANH MỤC CÁC HÌNH VẼ 10](#_Toc154005342)

[CHƯƠNG 1: TỔNG QUAN VỀ ĐỀ TÀI 13](#_Toc154005343)

[1.1. Tên đề tài 13](#_Toc154005344)

[1.2. Lý do chọn đề tài 13](#_Toc154005345)

[1.3. Mục tiêu của đề tài 13](#_Toc154005346)

[1.4. Đối tượng và phạm vi 14](#_Toc154005347)

[1.5. Kết quả dự kiến đạt được 15](#_Toc154005348)

[1.6. Tổng kết chương 1 15](#_Toc154005349)

[CHƯƠNG 2: TÌM HIỂU CÔNG NGHỆ XÂY DỰNG WEBSITE 15](#_Toc154005350)

[2.1. Tổng quan về Spring Framework 15](#_Toc154005351)

[2.1.1. Spring Framework là gì? 15](#_Toc154005352)

[2.1.2. Spring Framework hoạt động như thế nào? 16](#_Toc154005353)

[2.1.3. Ưu nhược điểm của Spring Framework 17](#_Toc154005354)

[2.1.3.1. Ưu điểm 17](#_Toc154005355)

[2.1.3.2. Nhược điểm 18](#_Toc154005356)

[2.1.4. Ứng dụng của Spring Framework 19](#_Toc154005357)

[2.2. Tổng quan về Angular 20](#_Toc154005358)

[2.2.1. Angular là gì? 20](#_Toc154005359)

[2.2.2. Angular hoạt động như thế nào? 20](#_Toc154005360)

[2.2.3. Ưu nhược điểm của Angular 21](#_Toc154005361)

[2.2.3.1. Ưu điểm 21](#_Toc154005362)

Danh mục hình vẽ

Danh mục bảng biểu

# MỞ ĐẦU

## 1. Tên đề tài

Đề tài “**Xây dựng phần mềm quản lý bán hàng mặt hàng đồ trang trí trên hệ điều hành Windows**”.

## 2. Lý do chọn đề tài

Phần mềm quản lý bán hàng có rất nhiều ứng dụng trong các lĩnh vực kinh doanh hàng hóa, bao gồm:

Cung cấp các trình quản lý: Điều này đặc biệt cần thiết trong các ngành kinh doanh hàng hóa bởi sự đa dạng và số lượng hàng hàng hóa luôn cần một sự quản lý độ chính xác cao, đi kèm đó là tốc độ phản hổi. Việc xây dựng phần mềm nhằm cung cấp một phương tiện quản lý đáp ứng được các nhu cầu trên.

Hỗ trợ tra cứu thông tin: Phần mềm giúp tra cứu và phân loại một lượng thông tin lớn một cách nhanh chóng và chính xác.

Báo cáo: Đây được coi là một trong những chức năng hữu ích nhất trong các trình quản lý, giúp cho người dùng có một cái nhìn tổng quát về các sản phẩm mặt hàng và tình trạng của chúng.

Thống kê công nợ: Việc tính toán sổ sách theo hướng truyền thống vẫn được nhiều hộ gia đình kinh doanh áp dụng, tuy nhiên về hiệu suất công việc thì không cao. Vì vậy, việc xây dựng một phần mềm đảm nhiệm các công việc tính toán luôn là điều cần thiết vì tính chính xác gần như tuyệt đối.

Hiện đại hóa ngành bán lẻ: Việc xây dựng một phần mềm quản lý chuyên nghiệp không chỉ là việc cải thiện trải nghiệm của người dùng mà còn là cách để ngành bán lẻ hiện đại hóa hoạt động kinh doanh của mình, tạo ra sự minh bạch và đáng tin cậy.

Tính ứng dụng và thực tiễn: Đề tài này có tính ứng dụng cao và gần gũi với thực tế. Việc xây dựng phần mềm không chỉ là một dự án thực hành mà còn là một cơ hội để áp dụng kiến thức lý thuyết vào thực tế công việc.

## 3. Mục tiêu của đề tài

Đề tài: Xây dựng phần mềm quản lý bán hàng mặt hàng đồ trang trí trên hệ điều hành Windows đáp ứng được những mục tiêu:

Trình quản lý: Phần mềm tạo ra một giao diện và không gian giúp quản lý thông tin hàng hóa, nhà cung cấp và cả người dùng.

Tra cứu thông tin nhanh: Tạo ra một giao diện trực quan và dễ sử dụng kèm theo đó là thuật toán giúp tăng tốc độ tìm kiếm và độ chính xác.

Cung cấp thông tin chính xác và toàn diện: Xác định và cập nhật thông tin chi tiết về những sản phẩm mới.

Tích hợp nhập và xuất các loại báo cáo định dạng excel: Cho phép nhập vào danh sách hàng với số lượng lớn (định dạng excel) giúp tăng năng suất lên nhiều lần, và xuất báo cáo về hàng hóa, doanh thu giúp chủ cửa hàng có cái nhìn toàn diện và dễ dàng về công việc kinh doanh.

Thống kê công nợ: Đưa ra thống kê về công nợ cũng như doanh thu của cửa hàng giúp chủ cửa hàng nắm bắt được tình hình kinh doanh nhanh chóng.

Kiểm soát an toàn thông tin: Bảo đảm an toàn và bảo mật thông tin cá nhân của người dùng và thông tin liên quan theo các tiêu chuẩn an toàn dữ liệu.

## 4. Đối tượng và phạm vi

Đối tượng:

Quản lý cửa hàng và chủ cửa hàng: Những người sử dụng để trực tiếp quản lý, cập nhật và tra cứu thông tin.

Phạm vi:

Phần mềm dành cho hộ kinh doanh gia đình được thiết kế và phát triển để đáp ứng nhu cầu thực tiễn về việc quản lý và số hóa những tài liệu hiện có nhằm tăng sự chính xác và tốc độ.

Xuất báo cáo dành cho khách hàng: Dành cho các đối tượng là khách hàng thân thiết muốn nắm bắt thông tin và kiểm tra các khoản thanh toán cũng như hàng hóa đã được giao.

## 5. Kết quả dự kiến đạt được

Xây dựng phần mềm quản lý bán hàng mặt hàng đồ trang trí trên hệ điều hành Windows.

## 6. Cấu trúc của đề tài

Trong báo cáo đồ án này, cấu trúc của đề tài được chia thành 3 chương:

* Chương 1: Tổng quan về công nghệ sử dụng trong dự án
* Chương 2: Phân tích hệ thống
* Chương 3: Kết luận

**CHƯƠNG 1**

**TỔNG QUAN VỀ DỰ ÁN PHẦN MỀM**

Giới thiệu chung: chương 1 trình bày tổng quan về dự án phát triển phần mềm quản lí tồn kho trên windows

* 1. **GIỚI THIỆU VỀ DỰ ÁN PHÁT TRIỂN PHẦN MỀM QUẢN LÝ TỒN KHO TRÊN HỆ ĐIỀU HÀNH WINDOWS**
     1. **Giới thiệu chung**
* Phần mềm quản lý hoạt động trên hệ điều hành windows
* Sử dụng tại gia đình (Thôn 6, Yên Sở, Hoài Đức, Hà Nội)
* Các tác nhân ngoài của hệ thống:
  + Chủ cửa hàng: Theo dõi tình trạng nhập/bán hàng, doanh thu hàng tháng/năm, khách hàng, nhà cung cấp, hàng tồn, công nợ.
  + Quản lý: Theo dõi tình trạng nhập/bán hàng, thông tin khách hàng.
    1. **Khảo sát và thu thập yêu cầu**

Hiện nay, không khó để tìm được những phần mềm quản lý bán hàng tại các kho hàng hay những cơ sở kinh doanh lớn trên khắp cả nước, em có thể tham khảo được rất nhiều các chức năng để hình thành nên phần mềm quản lý bán hàng dành cho hệ điều hành Windows. Kèm theo đó, em cũng tham khảo chính nhu cầu từ gia đình để rút ra được những chức năng cần thiết giúp tối ưu và quản lý công việc kinh doanh. Dưới đây là một số chức năng chính của phần mềm:

* Quản lý thông tin hàng hóa
* Quản lý nhập hàng
* Quản lý bán hàng
* Quản lý công nợ
* Xuất báo cáo excel
* Quản lý thông tin người dùng
* Service gửi email thông báo
  1. **CÔNG CỤ, PHƯƠNG PHÁP, KỸ THUẬT VẬN DỤNG**
* Các công cụ sử dụng trong dự án:
  + Visual studio:
  + PgAdmin4
  + Github
  + MS Server
  + Ngôn ngữ lập trình: C#
  + Framework: windows forms

### Giới thiệu về .NET Core

.NET Core là phiên bản cải tiến của .NET Framework, là một nền tảng phát triển đa năng, mã nguồn mở, miễn phí được duy trì bởi Microsoft. Nó là một framework đa nền tảng chạy trên các hệ điều hành Windows, macOS và Linux.

.NET Core có thể được sử dụng để xây dựng các loại ứng dụng khác nhau như thiết bị di động, máy tính để bàn, web, đám mây, IoT, máy học, microservice, trò chơi, v.v.

.NET Core có các điểm cốt lõi sau:

* Khung nguồn mở: .NET Core là một khung nguồn mở được Microsoft duy trì và có sẵn trên GitHub theo giấy phép MIT và Apache 2. Đây là một dự án thuộc .NET Foundation. Bạn có thể xem, tải xuống hoặc đóng góp cho mã nguồn qua các kho lưu trữ GitHub.
* Đa nền tảng: .NET Core chạy trên Windows, macOS và Linux, có các thời gian chạy khác nhau cho mỗi hệ điều hành nhưng tạo ra cùng một đầu ra.
* Nhất quán trên các kiến trúc: .NET Core thực thi mã với cùng hành vi trên các kiến trúc tập lệnh khác nhau, bao gồm x64, x86 và ARM.
* Hỗ trợ nhiều loại ứng dụng: .NET Core cho phép phát triển và chạy nhiều loại ứng dụng khác nhau như thiết bị di động, desktop, web, đám mây, IoT, máy học, microservices và trò chơi.
* Hỗ trợ nhiều ngôn ngữ lập trình: .NET Core hỗ trợ các ngôn ngữ lập trình C#, F# và Visual Basic. Bạn có thể sử dụng các IDE yêu thích như Visual Studio 2017/2019, Visual Studio Code, Sublime Text, Vim, và nhiều hơn nữa.
* Kiến trúc mô-đun: .NET Core hỗ trợ kiến trúc mô-đun thông qua các gói NuGet. Các tính năng khác nhau có thể được thêm vào dự án khi cần thiết, giảm dung lượng bộ nhớ và tăng tốc hiệu suất.
* Công cụ CLI: .NET Core bao gồm các công cụ CLI (Giao diện dòng lệnh) để phát triển và tích hợp liên tục.
* Triển khai linh hoạt: Ứng dụng .NET Core có thể được triển khai trên toàn người dùng, toàn hệ thống hoặc với Docker Container.
* Khả năng tương thích: .NET Core tương thích với .NET Framework và Mono API thông qua .NET Standard.
* Hiệu suất cao: .NET Core được tối ưu hóa về hiệu suất với các tính năng như biên dịch Just-In-Time (JIT), dịch mã thành hướng dẫn máy trong thời gian chạy để cải thiện tốc độ thực thi.
* Nền tảng hợp nhất: Từ .NET Core 3.1 trở đi, Microsoft đã hợp nhất các nền tảng .NET, tập hợp .NET Core, .NET Framework và Xamarin thành một nền tảng duy nhất gọi là “.NET” bắt đầu từ .NET 5, nhằm cung cấp tính nhất quán API và hành vi thời gian chạy trên các loại ứng dụng khác nhau.

### Giới thiệu về Windows Forms

Windows Forms (WinForms) là [thư viện lớp đồ họa](https://vi.wikipedia.org/wiki/Giao_di%E1%BB%87n_%C4%91%E1%BB%93_h%E1%BB%8Da_ng%C6%B0%E1%BB%9Di_d%C3%B9ng) (GUI) [mã nguồn mở](https://vi.wikipedia.org/wiki/Ph%E1%BA%A7n_m%E1%BB%81m_ngu%E1%BB%93n_m%E1%BB%9F) và miễn phí được bao gồm như một phần của Microsoft.NET Framework hoặc Mono Framework, cung cấp nền tảng để viết các ứng dụng khách phong phú cho máy tính để bàn, [máy tính xách tay](https://vi.wikipedia.org/wiki/M%C3%A1y_t%C3%ADnh_x%C3%A1ch_tay) và [máy tính bảng](https://vi.wikipedia.org/wiki/M%C3%A1y_t%C3%ADnh_b%E1%BA%A3ng). Mặc dù nó được coi là sự thay thế cho Thư viện lớp nền tảng Microsoft Foundation của C ++ trước đây và phức tạp hơn, nhưng nó không cung cấp mô hình tương đương và chỉ hoạt động như một nền tảng cho tầng giao diện người dùng trong một giải pháp nhiều tầng.

Vì dễ code do có hỗ trợ toolbox kéo thả và có nhiều component có sẵn, WinForm rất phù hợp để làm các phần mềm quản lý, tính tiền, thống kê… . Đây cũng là loại ứng dụng mà các công ty/doanh nghiệp vừa và nhỏ cần. Ngoài ra, chỉ cần sử dụng component như TelerikUI hoặc DevExpress , WinForms có thể tạo ra các giao diện hiện đại và có tính thẩm mĩ cao, từ đó phần mềm phát triền bằng Winforms có các lợi thế:

* Tốc độ xử lý dữ liệu nhanh chóng
* Đảm bảo an toàn, bảo mật thông tin
* Có thể chạy trên các phiên bản Windows khác nhau.
* Thao tác trên nhiều giao diện

### Giới thiệu về Git và Github

Git là tên gọi là một hệ thống quản lý phiên bản phân tán phổ biến nhất hiện nay (Distributed Version Control System – DVCS). DVCS là hệ thống giúp mỗi máy tính có thể lưu trữ nhiều phiên bản khác nhau của một mã nguồn được nhân bản (clone) từ một kho chứa mã nguồn (repository), mỗi thay đổi vào mã nguồn trên máy tính sẽ có thể ủy thác (commit) rồi đưa lên máy chủ nơi đặt kho chứa chính. Và một máy tính khác (nếu họ có quyền truy cập) cũng có thể clone lại mã nguồn từ kho chứa hoặc clone lại một tập hợp các thay đổi mới nhất trên máy tính kia. Trong Git, thư mục làm việc trên máy tính gọi là Working Tree.

Git giúp người dùng lưu lại các phiên bản của những lần thay đổi vào mã nguồn và dễ dàng cho việc khôi phục lại sau này. Do đó, việc backup lại phiên làm việc trước đó khi phát hiện lỗi trở nên đơn giản. Ngoài ra, khi các thành viên cùng team làm việc với nhau hoàn toàn có thể theo dõi online được các thay đổi của các thành viên khác ở từng phiên bản làm việc mà không nhất thiết phải ngồi ngay cạnh nhau, họ cũng có thể đối chiếu được những thay đổi đó rồi gộp (merge) thành một phiên bản hoàn chỉnh chứa những thay đổi của mọi thành viên trước khi đưa các thay đổi lên một kho chứa mã nguồn.

Cơ chế lưu trữ phiên bản của Git là Git sẽ tạo ra một “ảnh chụp” (snapshot) trên mỗi tập tin và thư mục sau khi commit, từ đó cho phép tái sử dụng lại một ảnh chụp nào đó mà có thể hiểu đó là một phiên bản. Đây cũng chính là lợi thế của Git so với các DVCS khác khi nó không “lưu cứng” dữ liệu mà sẽ lưu với dạng snapshot.

GitHub là một dịch vụ lưu trữ trên web dành cho các dự án có sử dụng hệ thống kiểm soát Git revision.

Khi nhắc đến Git, mọi người thường nghĩ ngay đến Github và thường coi chúng là một. Tuy nhiên 2 thuật ngữ này khác hẳn nhau, trong khi Git là tên gọi của một mô hình hệ thống, các máy tính có thể clone lại mã nguồn từ một repository (kho lưu trữ), còn GitHub là tên của một công ty cung cấp dịch vụ máy chủ repository (kho lưu trữ) công cộng, mỗi người có thể truy cập vào website trang chủ để tạo tài khoản trên đó và tạo ra kho chứa source (nguồn dữ liệu) của riêng mình khi làm việc.

Git mang đến nhiều lợi thế cho công việc lập trình:

* Git dễ sử dụng, an toàn và nhanh chóng.
* Quản lý source code dễ dàng chuyên nghiệp
* Có thể giúp quy trình làm việc code theo nhóm đơn giản hơn rất nhiều bằng việc kết hợp các phân nhánh (branch).
* Hạn chế được lỗi xảy ra trong quá trình code trong 1 team
* Khi gặp lỗi có thể dễ dàng Backup lại phiên bản trước
* Code không giới hạn khoảng cách giữa các thành viên trong team, bạn có thể làm việc ở bất cứ đâu vì chỉ cần clone mã nguồn từ kho chứa hoặc clone một phiên bản thay đổi nào đó từ kho chứa, hoặc một nhánh nào đó từ kho chứa.
* Dễ dàng trong việc deployment sản phẩm.

### Giới thiệu về Postgres

PostgreSQL, thường được viết tắt là "Postgres," là một hệ thống quản lý cơ sở dữ liệu mạnh mẽ và mã nguồn mở. Nó là một hệ thống quản lý cơ sở dữ liệu quan hệ (RDBMS) tiên tiến, được phát triển từ dự án POSTGRES từ Đại học California vào những năm đầu của thập kỷ 1980. PostgreSQL là một giải pháp mạnh mẽ, có khả năng xử lý các nhiệm vụ lưu trữ, truy vấn và xử lý dữ liệu phức tạp.

Được đánh giá cao về tính mở rộng, bảo mật và khả năng tùy chỉnh, PostgreSQL hỗ trợ nhiều loại dữ liệu khác nhau, từ cơ bản như văn bản và số đến dạng dữ liệu phức tạp như hình ảnh, âm thanh và dữ liệu địa lý. Nó cũng cung cấp một loạt các tính năng mở rộng như hỗ trợ cho các hàm và kiểu dữ liệu người dùng tự định nghĩa, truy vấn đa chiều, và các tính năng nâng cao về bảo mật.

PostgreSQL thường được sử dụng trong các ứng dụng web, hệ thống thông tin doanh nghiệp, và các dự án khoa học công nghệ, nhờ vào tính linh hoạt, hiệu suất cao và tính ổn định. Đồng thời, với việc là một hệ thống mã nguồn mở, nó thu hút một cộng đồng lớn các nhà phát triển cùng đóng góp vào sự phát triển và cải thiện của nó.

**KẾT LUẬN CHƯƠNG 1**

Trongchương 1 em đã trình bày về tổng quan dự án phần mềm quản lí tồn kho chạy trên hệ điều hành windows bao gồm các yêu cầu chức năng/phi chức năng, các công cụ, phương pháp, kĩ thuật được sử dụng để hiện thực hóa dự án.

**CHƯƠNG 2**

**PHÂN TÍCH YÊU CẦU PHẦN MỀM**

Giới thiệu chương: chương 2 đưa ra phân tích chi tiết về phần mềm, bao gồm các tác nhân của hệ thống và các yêu cầu chức năng/phi chức năng của hệ thống.

**2.1. CÁC TÁC NHÂN CỦA HỆ THỐNG**

Phần mềm được thiết kế với quy mô kinh doanh hộ gia đình nhỏ và mục đích quản lý và tương tác nội bộ nên tác nhân bao gồm:

* Chủ cửa hàng
* Người quản lý

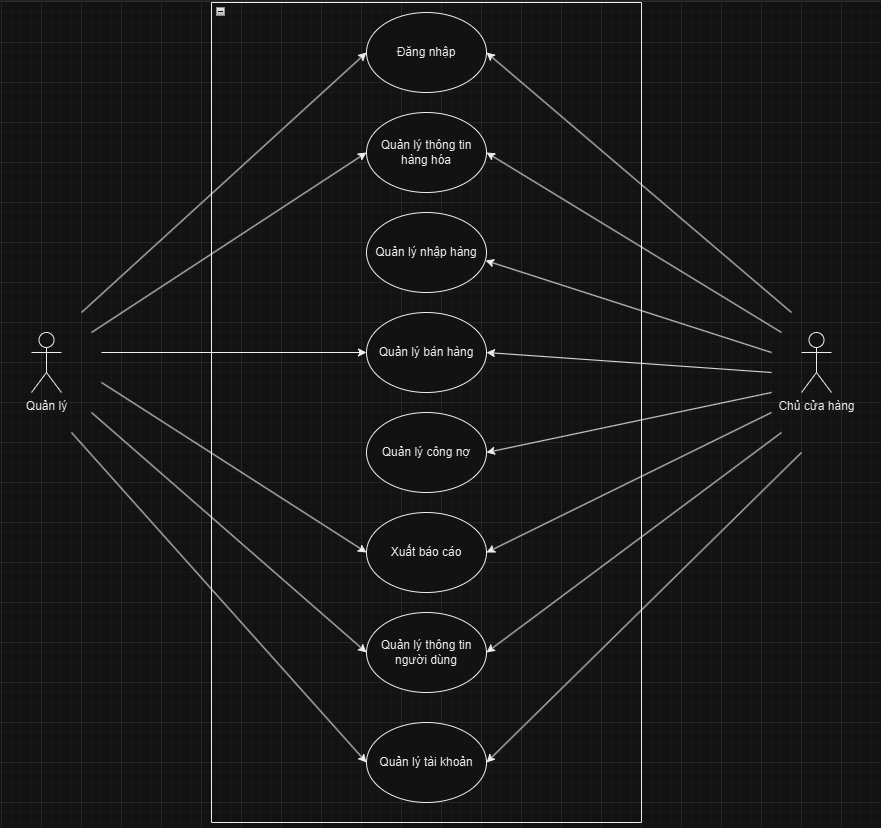
**2.2. CÁC YÊU CẦU CHỨC NĂNG**

Chủ cửa hàng và người quản lý là hai tác nhân duy nhất tương tác với phần mềm. Chủ cửa hàng có thẩm quyền cao hơn khi sử dụng được mọi chức năng của phần mềm:

* quản lý tài khoản cá nhân.
* quản lý hàng hóa.
* quản lý thông tin khách hàng.
* quản lý thông tin nhà cung cấp.
* quản lý công nợ, xuất báo cáo)

Người quản lý chỉ được cấp phép sử dụng các chức năng:

* quản lý tài khoản cá nhân.
* quản lý hàng hóa.
* quản lý thông tin khách hàng.
* xuất báo cáo.

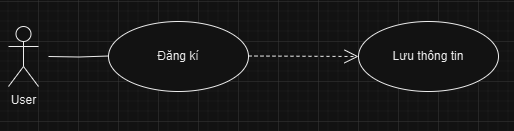


Hình 2.1. Biểu đồ use case tổng quát

**2.2.1. Giới thiệu về các yêu cầu chức năng hệ thống**

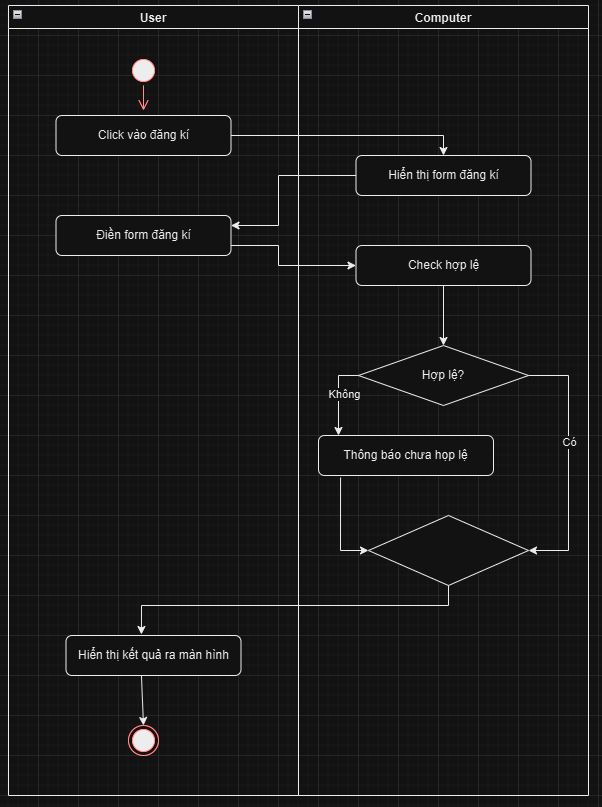
**2.2.1.2. Yêu cầu chức năng đăng kí tài khoản**

Trước khi có thể đăng nhập và sử dụng các chức năng khác của phần mềm, người dùng cần đăng kí tài khoản.



Hình 2.1: Biểu đồ minh họa UC đăng kí tài khoản (Register)

Dưới đây là quy trình đăng kí tài khoản thông thường của một user. Quy trình được minh họa bằng biểu đồ Activity Diagram.



Hình 2.2: Biểu đồ minh họa quy trình hoạt động của UC

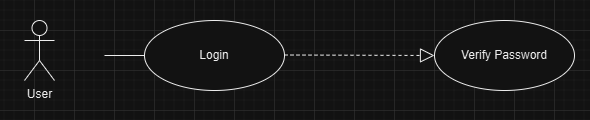
Bảng 2.1**.** Đặc tả chức năng Đăng kí tài khoản

|  |  |  |  |
| --- | --- | --- | --- |
| Mã Use case | UC01 | Tên Use case | UC đăng kí |
| Tác nhân | User | | |
| Mô tả | Người dùng thực hiện đăng kí tài khoản để đăng nhập và sử dụng các chức năng khác của hệ thống. | | |
| Sự kiện kích hoạt chức năng | Click đăng kí trên màn hình chính. | | |
| Tiền điều kiện | Không | | |
| Luồng sự kiện chính | |  |  |  | | --- | --- | --- | | **#** | **Thực hiện bởi** | **Hành động** | | 1 | Người dùng | Click vào đăng kí | | 2 | Hệ thống | Hiển thị form đăng kí | | 3 | Người dùng | Nhập form và nhấn đăng kí | | 4 | Hệ thống | Phản hồi trạng thái đăng kí | | | |
| Luồng sự kiện thay thế | Không | | |
| Hậu điều kiện | Không | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **#** | **Trường dữ liệu** | **Mô tả** | **Bắt buộc (Y/N)** | **Điều kiện hợp lệ** | **Ví dụ** |
|  | Username | Tên đăng nhập | Y | Không trùng | Username chưa được đăng kí |
|  | Password | Mật khẩu | Y | Bảo mật | Chứa các kí tự chữ, số, kí tự đặc biệt |
|  | Gmail | Email đăng kí | Y | Đúng format | abc@gmail.com |

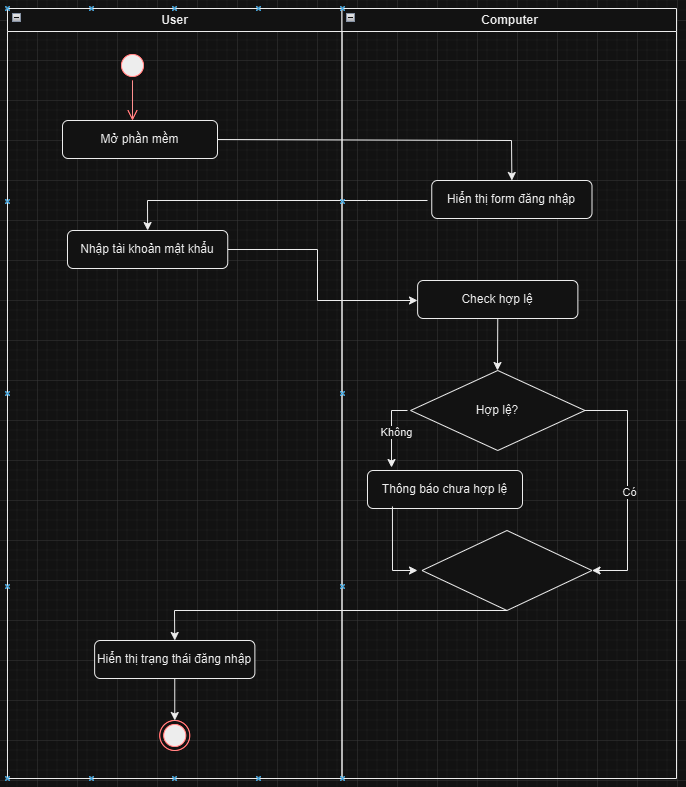
**2.2.1.3. Yêu cầu chức năng đăng nhập**

Trước khi sử dụng các chức năng của hệ thống, người dùng cần đăng nhập.



Hình 2.3: Biểu đồ minh họa UC đăng nhập (Login)

Dưới đây là quy trình đăng nhập thông thường của một user. Quy trình được minh họa bằng biểu đồ Activity Diagram.



Hình 2.4: Biểu đồ minh họa quy trình hoạt động của UC

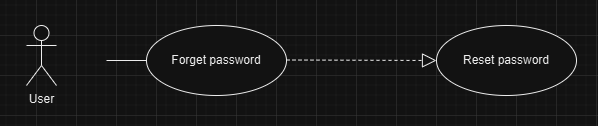
Bảng 2.2**.** Đặc tả chức năng Đăng nhập

|  |  |  |  |
| --- | --- | --- | --- |
| Mã Use case | UC02 | Tên Use case | UC đăng nhập |
| Tác nhân | User | | |
| Mô tả | Người dùng thực hiện đăng nhập để sử dụng các chức năng khác của hệ thống. | | |
| Sự kiện kích hoạt chức năng | Click đăng nhập trên màn hình chính. | | |
| Tiền điều kiện | Đã có tài khoản | | |
| Luồng sự kiện chính | |  |  |  | | --- | --- | --- | | **#** | **Thực hiện bởi** | **Hành động** | | 1 | Người dùng | Vào phần mềm | | 2 | Hệ thống | Hiển thị form đăng nhập | | 3 | Người dùng | Nhập thông tin tài khoản mật khẩu, chọn đăng nhập | | 4 | Hệ thống | Phản hồi trạng thái đăng nhập | | | |
| Luồng sự kiện thay thế | |  |  |  | | --- | --- | --- | | **#** | **Thực hiện bởi** | **Hành động** | | 4a | Hệ thống | Thông báo không hợp lệ, yêu cầu đăng nhập lại | | | |
| Hậu điều kiện | Không | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **#** | **Trường dữ liệu** | **Mô tả** | **Bắt buộc (Y/N)** | **Điều kiện hợp lệ** | **Ví dụ** |
|  | Username | Tên đăng nhập | Y | Nhập đúng tên đăng nhập |  |
|  | Password | Mật khẩu | Y | Nhập đúng mật khẩu ứng với tên đăng nhập |  |

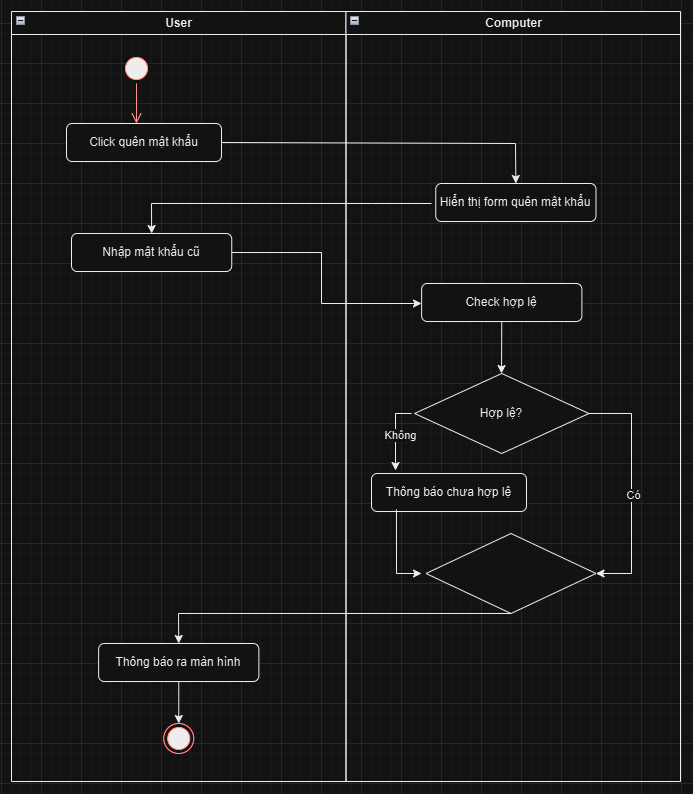
**2.2.1.4. Yêu cầu chức năng quên mật khẩu**

Cập nhật lại mật khẩu đăng nhập khi người dùng quên mật khẩu.



Hình 2.5: Biểu đồ minh họa UC đăng nhập (Forget password)

Dưới đây là quy trình quên mật khẩu thông thường của một user. Quy trình được minh họa bằng biểu đồ Activity Diagram.



Hình 2.4: Biểu đồ minh họa quy trình hoạt động của UC

Bảng 2.3**.** Đặc tả chức năng Quên mật khẩu

|  |  |  |  |
| --- | --- | --- | --- |
| Mã Use case | UC03 | Tên Use case | UC quên mật khẩu |
| Tác nhân | User | | |
| Mô tả | Người dùng đặt lại mật khẩu đăng nhập khi quên | | |
| Sự kiện kích hoạt chức năng | Click quên mật khẩu trên màn hình chính | | |
| Tiền điều kiện | Đã có tài khoản | | |
| Luồng sự kiện chính | |  |  |  | | --- | --- | --- | | **#** | **Thực hiện bởi** | **Hành động** | | 1 | Người dùng | Click quên mật khẩu | | 2 | Hệ thống | Hiển thị form quên mật khẩu | | 3 | Người dùng | Nhập thông email lấy lại mật khẩu và mật khẩu mới, sau đó chọn submit | | 4 | Hệ thống | Phản hồi kêt quả | | | |
| Luồng sự kiện thay thế | |  |  |  | | --- | --- | --- | | **#** | **Thực hiện bởi** | **Hành động** | | 4a | Hệ thống | Thông báo không hợp lệ, yêu cầu nhập lại | | | |
| Hậu điều kiện | Không | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **#** | **Trường dữ liệu** | **Mô tả** | **Bắt buộc (Y/N)** | **Điều kiện hợp lệ** | **Ví dụ** |
|  | Username | Tên đăng nhập | Y | Nhập đúng tên đăng nhập |  |
|  | Password | Mật khẩu | Y | Nhập đúng mật khẩu ứng với tên đăng nhập |  |