**Bộ giáo dục và đào tạo**

**Trường Đại học Sài Gòn**

--🙢🕮🙠--



**Báo cáo: Đồ án giữa kì**

**Môn: Công nghệ phần mềm**

**Đề tài: Sales Management Application(Ứng dụng quản lý bán hàng)**

Sinh viên thực hiện:

*3121411226 – Trần Quang Vinh.(nhóm trưởng)*

*3121411045 – Đặng Nguyễn Quốc Dương*

*3121411060 – Nguyễn Trí Đức*

*3121411100 – Phạm Quang Khiêm*

Giáo viên hướng dẫn: *Thầy Đỗ Như Tài*

***TPHCM, Ngày 4, tháng 5, năm 2025***

***Chương 1: Tổng quan đề tài***

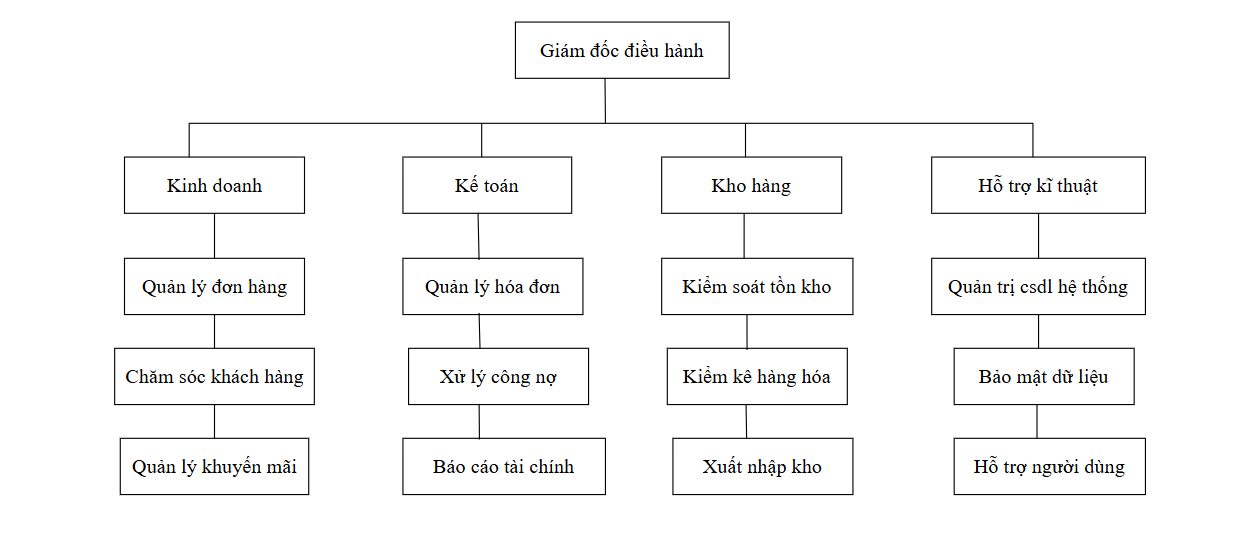
* 1. **Khảo sát hiện trạng hệ thống**

1. **Đối nội**
2. **Cơ cấu tổ chức nội bộ.**

Phần mềm quản lý bán hàng hoạt động như một hệ thống thống nhất, hỗ trợ quản lý các quy trình kinh doanh và vận hành nội bộ. Cơ cấu tổ chức bao gồm:

* **Ban quản lý & điều hành:** Ra quyết định, quản lý hệ thống và giám sát hoạt động.
* **Bộ phận kinh** **doanh:** Quản lý đơn hàng, khách hàng, giá cả và chương trình khuyến mãi.
* **Bộ phận kho hàng:** Kiểm soát hàng tồn kho, nhập xuất kho, kiểm kê hàng hóa.
* **Bộ phận kế toán:** Xử lý hóa đơn, thanh toán, công nợ và báo cáo tài chính.
* **Bộ phận kỹ thuật (IT):** Đảm bảo phần mềm vận hành ổn định, hỗ trợ kỹ thuật và bảo mật dữ liệu.
* **Bộ phận chăm sóc khách hàng:** Hỗ trợ khách hàng, giải quyết khiếu nại và tư vấn sản phẩm.

1. **Sơ đồ cơ cấu tổ chức nội bộ**

****

1. **Đối ngoại**
2. **Tổ chức và môi trường của tổ chức về mặt đối ngoại**

Phần mềm quản lý bán hàng không chỉ hoạt động nội bộ mà còn tương tác với các yếu tố bên ngoài như:

* **Khách hàng:** Giao tiếp qua đơn hàng, hóa đơn, chương trình khuyến mãi, chăm sóc khách hàng.
* **Nhà cung cấp:** Liên kết nhập hàng, quản lý hợp đồng, kiểm soát chất lượng sản phẩm.
* **Cơ quan thuế & tài chính:** Đáp ứng các yêu cầu về kế toán, hóa đơn điện tử, báo cáo thuế.
* **Ngân hàng & cổng thanh toán:** Hỗ trợ giao dịch tài chính, thanh toán trực tuyến.
* **Đối tác vận chuyển:** Kết nối với đơn vị giao hàng để theo dõi đơn hàng, tối ưu vận chuyển**.**

1. **Môi trường kinh doanh ảnh hưởng đến phần mềm.**

* **Yếu tố công nghệ:** Cập nhật xu hướng công nghệ như AI, Big Data, IoT để nâng cao hiệu quả.
* **Yếu tố pháp lý:** Tuân thủ quy định về bảo mật dữ liệu (GDPR, Nghị định 52/2013/NĐ-CP).
* **Yếu tố thị trường:** Cạnh tranh, nhu cầu khách hàng, xu hướng tiêu dùng.
  1. **Quy trình nghiệp vụ**

Phần mềm quản lý bán hàng hỗ trợ nhiều nghiệp vụ chính, bao gồm:

|  |  |
| --- | --- |
| Nghiệp vụ | Quy trình liên quan |
| Quản lý đơn hàng | Tạo đơn hàng, xử lý thanh toán, in hóa đơn, theo dõi đơn hàng |
| Quản lý danh mục và sản phẩm | Nhập hàng, xuất hàng, kiểm kê hàng tồn kho theo nhà cung cấp |
| Quản lý khách hàng | Lưu trữ thông tin khách hàng, lịch sử mua hàng, chương trình khuyến mãi |
| Quản lý nhà cung cấp | Quản lý thông tin nhà cung cấp, hợp đồng, đặt hàng nhập kho |
| Quản lý tài chính - kế toán | Theo dõi doanh thu, công nợ, lập báo cáo tài chính |
| Báo cáo & thống kê | Xuất báo cáo doanh thu, báo cáo tồn kho, phân tích dữ liệu bán hàng |

* **Nghiệp vụ dưới góc nhìn của người làm quản lý**

Từ góc nhìn quản lý, các nghiệp vụ quan trọng nhất bao gồm:

* + Giám sát doanh thu và lợi nhuận: Theo dõi tình hình bán hàng, hiệu quả kinh doanh.
  + Kiểm soát tồn kho: Đảm bảo hàng hóa đủ đáp ứng nhu cầu nhưng không bị tồn đọng.
  + Quản lý khách hàng & chăm sóc khách hàng: Xây dựng quan hệ bền vững với khách hàng.
  + Ra quyết định chiến lược: Dựa trên báo cáo dữ liệu để điều chỉnh chính sách kinh doanh.

|  |  |  |
| --- | --- | --- |
| Nghiệp vụ | Các công đoạn | Bộ phận liên quan |
| Quản lý đơn hàng | Tạo đơn hàng → Thanh toán → In hóa đơn → Cập nhật doanh thu | Người quản lý |
| Nhập hàng | Kiểm tra hàng cần nhập → Đặt hàng từ nhà cung cấp → Nhận hàng → Cập nhật tồn kho |
| Kiểm kê kho | Kiểm tra số lượng thực tế → So sánh với dữ liệu → Điều chỉnh nếu cần |
| Quản lý khách hàng | Lưu thông tin → Theo dõi lịch sử mua hàng → Chăm sóc khách hàng |
| Báo cáo & thống kê | Thu thập dữ liệu → Tổng hợp → Phân tích → Xuất báo cáo | Kế toán |

* **Các câu hỏi phỏng vấn khách hàng để lấy yêu cầu xây dựng quy trình nghiệp vụ**

**Câu hỏi chung về quy trình hiện tại**

1. Anh/chị có thể mô tả quy trình bán hàng của cửa hàng/doanh nghiệp không?

2. Hiện tại, anh/chị gặp những khó khăn gì trong quá trình quản lý bán hàng?

3. Những công đoạn nào đang tốn nhiều thời gian nhất?

**Câu hỏi về quản lý sản phẩm & nhập hàng**

4. Cách anh/chị theo dõi lượng hàng tồn kho hiện nay như thế nào?

5. Tần suất kiểm kê kho hàng là bao lâu? Có gặp khó khăn gì trong việc này không?

6. Khi phát sinh đơn hàng lớn, việc nhập kho và xử lý đơn hàng có bị chậm trễ không?

**Câu hỏi về chăm sóc khách hàng**

7. Anh/chị có lưu trữ thông tin khách hàng không? Nếu có, làm cách nào?

8. Anh/chị có theo dõi lịch sử mua hàng của khách không? Nếu có, cách nào?

**Câu hỏi về báo cáo & tài chính**

9. Anh/chị có cần báo cáo doanh thu, lợi nhuận theo ngày/tháng/năm không?

10. Khi cần kiểm tra công nợ khách hàng, anh/chị mất bao lâu để tổng hợp dữ liệu?

11. Hiện tại, việc theo dõi dòng tiền trong doanh nghiệp có gặp vấn đề gì không?

**Câu hỏi mở rộng về quy trình hiện tại**

1. Trong quy trình bán hàng hiện nay, có bước nào anh/chị thấy dư thừa hoặc có thể tự động hóa được không?
2. Anh/chị có đang sử dụng phần mềm hỗ trợ nào không? Nếu có, phần nào thấy chưa đáp ứng được nhu cầu?

**Câu hỏi chi tiết về quản lý kho & sản phẩm**

1. Khi có sản phẩm sắp hết hàng, hệ thống (nếu có) có cảnh báo không? Anh/chị muốn có tính năng này không?
2. Việc quản lý đơn nhập hàng từ nhà cung cấp hiện tại đang làm thủ công hay có quy trình rõ ràng?
3. Anh/chị có gặp khó khăn trong việc quản lý nhiều nhóm sản phẩm, mã sản phẩm hoặc lô hàng khác nhau không?

**Câu hỏi sâu về khách hàng**

1. Có bao giờ anh/chị muốn phân loại khách hàng theo mức độ mua hàng (thân thiết, tiềm năng, mới...) không?
2. Anh/chị có thực hiện chương trình khuyến mãi, tích điểm, giảm giá theo từng loại khách không?

**Câu hỏi về phân quyền và quản lý nhân viên**

1. Anh/chị có cần hệ thống phân quyền cho từng nhân viên? (VD: nhân viên bán hàng không được chỉnh sửa đơn hàng,...)
2. Anh/chị có theo dõi hiệu suất làm việc của từng nhân viên không? Có cần báo cáo theo nhân viên?
   1. **Đặc tả yêu cầu chức năng và phi chức năng**

* **Đặc tả yêu cầu chức năng:**

1. **Chức năng quản lý tài khoản khách hàng**

* Hệ thống cho phép người dùng đăng ký tài khoản mới bằng cách cung cấp các thông tin cá nhân như họ tên, email, số điện thoại, mật khẩu,...
* Hệ thống cho phép người dùng đăng nhập để truy cập vào các chức năng mua sắm.
* Hệ thống cho phép người dùng xem danh sách sản phẩm, xem chi tiết sản phẩm, và thêm sản phẩm vào giỏ hàng.
* Hệ thống cho phép người dùng tạo đơn hàng bao gồm:
* Chọn nhiều sản phẩm cho một đơn hàng.
* Chọn phương thức thanh toán.
* Nhập thông tin giao hàng chính xác.
* Hệ thống lưu trữ thông tin đơn hàng vào cơ sở dữ liệu và cập nhật trạng thái đơn hàng ban đầu là "Waiting".

1. **Chức năng quản lý danh mục và sản phẩm**
2. **Quản lý danh mục**

Hệ thống cho phép admin:

* Thêm danh mục sản phẩm mới.
* Chỉnh sửa thông tin danh mục.
* Xóa danh mục sản phẩm.
* Tìm kiếm danh mục theo mã danh mục, tên danh mục và nhà cung cấp.

1. **Quản lý sản phẩm**

Hệ thống cho phép admin:

* Thêm sản phẩm mới với các thông tin: mã sản phẩm, tên sản phẩm, thông số kỹ thuật, giá bán, hình ảnh.
* Mỗi sản phẩm có thể có nhiều đơn vị hàng hóa tương ứng với mã IMEI riêng.
* Cập nhật thông tin sản phẩm.
* Xóa sản phẩm khỏi hệ thống.
* Tìm kiếm sản phẩm theo mã sản phẩm, tên sản phẩm hoặc nhà cung cấp.

1. **Chức năng xử lý đơn hàng(Admin)**

Hệ thống hiển thị danh sách các đơn hàng từ khách hàng với trạng thái "Waiting".

* **Admin có thể:**
* Kiểm tra, xác nhận đơn hàng → chuyển trạng thái thành "Confirming".
* Xử lý các trường hợp khách hàng bom hàng ≥ 2 lần → xóa tài khoản và cấm mua sắm vĩnh viễn.
* Sau khi xác nhận đơn hàng, hệ thống cho phép admin tạo hóa đơn xuất hàng bao gồm:
* Mã phiếu xuất.
* Mã nhân viên phụ trách.
* Thông tin khách hàng.
* Danh sách sản phẩm chi tiết (mã IMEI, giá gốc, giá khuyến mãi nếu có)
* **Mã khuyến mãi:**

Hệ thống cho phép áp dụng mã giảm giá trong hóa đơn nếu phù hợp với các chương trình như:

* Khuyến mãi sinh nhật khách hàng.
* Kỷ niệm ngày thành lập công ty.
* Tri ân khách hàng thân thiết.

1. **Chức năng nhập hàng và quản lý kho**

*Admin có thể:*

* Kiểm tra tồn kho sản phẩm.
* Tạo phiếu nhập hàng mới khi số lượng sản phẩm thấp.
* Cập nhật thông tin nhà cung cấp và danh mục hàng nhập.

*Hệ thống hỗ trợ:*

* Nhập số lượng lớn sản phẩm theo từng danh mục.
* Gán mã IMEI riêng biệt cho từng sản phẩm.
* Sau khi nhập, sản phẩm được thêm vào hệ thống và có sẵn để khách hàng đặt mua.

1. **Chức năng thống kê & báo cáo**

Hệ thống cung cấp chức năng thống kê doanh thu theo:

* Thời gian (ngày, tháng, năm).
* Danh mục sản phẩm.
* Tổng số lượng đơn hàng và sản phẩm bán ra.
* Dữ liệu doanh thu được trình bày dưới dạng báo cáo biểu đồ trực quan, giúp admin dễ dàng phân tích và đưa ra chiến lược kinh doanh.

1. **Chức năng gửi Email khách hàng**

Hệ thống tích hợp tính năng gửi email trực tiếp từ hệ thống đến khách hàng, hỗ trợ các nội dung:

* Xác nhận đơn hàng thành công.
* Gửi thông báo về khuyến mãi, chương trình ưu đãi.
* Gửi email chăm sóc khách hàng và hỗ trợ hậu mãi.
* **Đặc tả yêu cầu phi chức năng:**

1. **Hiệu năng (Performance):**

* Tốc độ phản hồi: Hệ thống phải phản hồi các thao tác người dùng trong vòng 3 giây cho các chức năng phổ biến như đăng nhập, tìm kiếm sản phẩm, và thêm vào giỏ hàng.
* Xử lý đơn hàng: Hệ thống phải có khả năng xử lý ít nhất 1000 đơn hàng/ngày mà không bị gián đoạn.
* Tìm kiếm sản phẩm: Kết quả tìm kiếm phải hiển thị trong vòng 2 giây sau khi người dùng nhập từ khóa.

1. **Khả năng mở rộng (Scalability)**

* Hệ thống được thiết kế để có thể mở rộng dễ dàng khi có thêm người dùng, sản phẩm hoặc tính năng.
* Có thể tích hợp với các dịch vụ điện toán đám mây hoặc hệ thống ERP trong tương lai mà không cần thay đổi toàn bộ kiến trúc hệ thống.

1. **Bảo mật (Security)**

* Mã hóa mật khẩu người dùng bằng thuật toán hash (ví dụ: SHA-256 hoặc bcrypt).
* Phân quyền truy cập rõ ràng giữa admin và khách hàng.
* Truyền dữ liệu nhạy cảm phải thông qua giao thức HTTPS.
* Hệ thống lưu lại log truy cập và hoạt động của người dùng để phục vụ mục đích giám sát và bảo trì.

1. **Khả năng sử dụng (Usability)**

* Giao diện người dùng phải thân thiện, dễ sử dụng, phù hợp với người dùng phổ thông.
* Hệ thống nên có phiên bản tương thích với thiết bị di động (responsive).
* Có sẵn hướng dẫn sử dụng cho admin và khách hàng mới.

1. **Độ tin cậy (Reliability)**

* Hệ thống phải duy trì hoạt động ổn định 24/7, trừ khi có bảo trì định kỳ.
* Trong trường hợp mất kết nối tạm thời, hệ thống phải có khả năng tự động khôi phục dữ liệu sau khi kết nối được thiết lập lại.
* Quá trình xử lý đơn hàng và lưu trữ dữ liệu cần được đảm bảo an toàn và toàn vẹn (không bị mất, lỗi dữ liệu).

1. **Khả năng bảo trì & cập nhật (Maintainability & Updatability)**

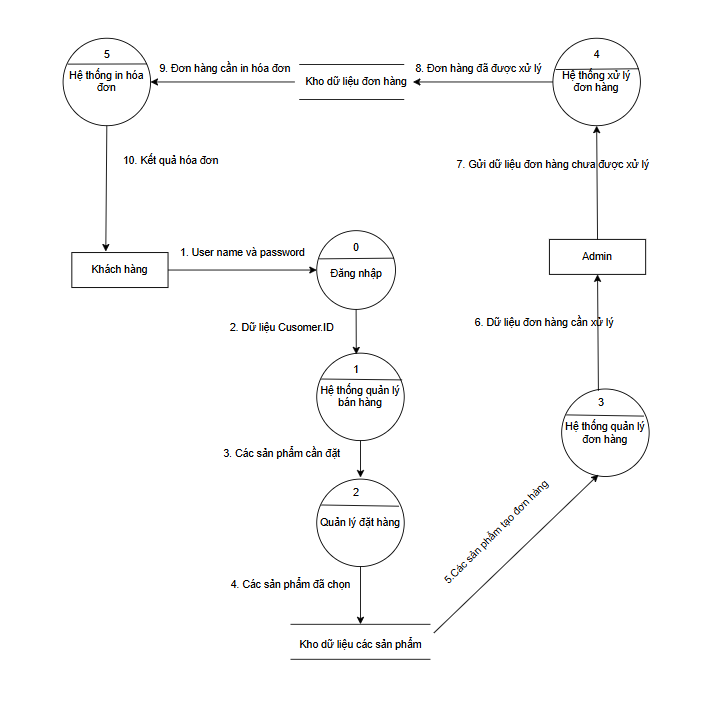
* Hệ thống phải được lập trình theo mô hình phân lớp (Model-View-Controller hoặc tương tự) để dễ bảo trì và cập nhật.
* Mỗi chức năng phải được đóng gói riêng biệt để dễ dàng nâng cấp hoặc thay thế module mà không ảnh hưởng đến toàn hệ thống.
* Có thể thực hiện kiểm thử tự động một phần trong quá trình phát triển và cập nhật.
  1. **Đề xuất giải pháp**

|  |  |  |
| --- | --- | --- |
| **Thành phần** | **Giải pháp** | **Lý do chọn** |
| Giao diện người dùng (Frontend) | Java Swing (phiên bản desktop) hoặc ReactJS (nếu muốn mở rộng lên web) | Giao diện linh hoạt, dễ phát triển, khả năng tương tác tốt |
| Xử lý logic nghiệp vụ (Backend) | Java (hoặc Python) theo mô hình MVC | Dễ bảo trì, phân chia rõ ràng giữa các tầng chức năng |
| Cơ sở dữ liệu | Microsoft SQL Server | Bảo mật tốt, hiệu năng ổn định, hỗ trợ dữ liệu lớn |
| Lưu trữ hình ảnh | Lưu tại thư mục cục bộ kết hợp đường dẫn trong DB | Tránh phình to CSDL, dễ thao tác hình ảnh |
| Báo cáo - thống kê | Thư viện JFreeChart (Java) hoặc Matplotlib (Python) | Tạo biểu đồ trực quan cho admin |
| Gửi Email | JavaMail API hoặc SMTP tích hợp Python | Gửi xác nhận đơn hàng, khuyến mãi cho khách hàng |
| Quản lý IMEI | Mỗi sản phẩm có danh sách IMEI riêng | Đảm bảo kiểm soát từng máy chính xác |

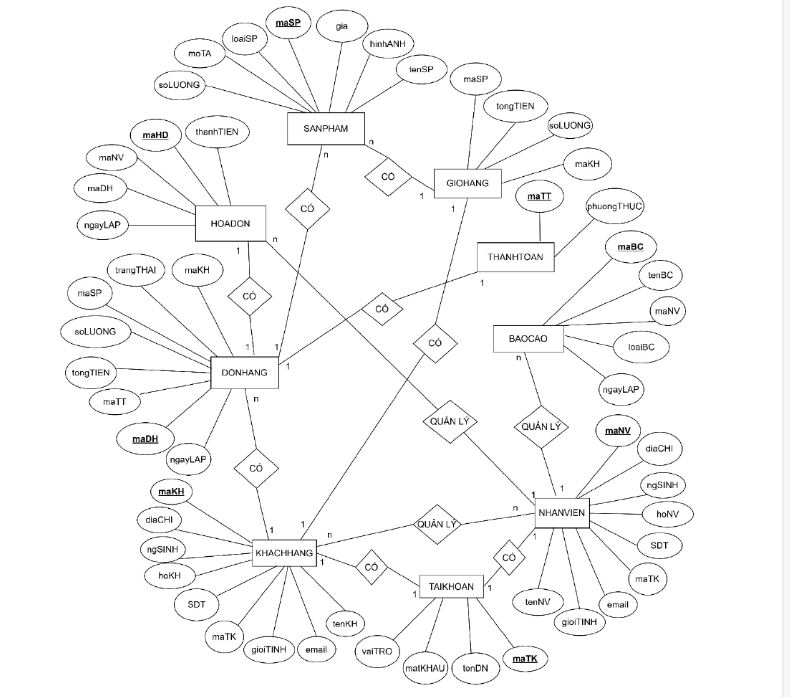
* **Mở rộng trong tương lai**
* Kết nối hệ thống với dịch vụ đám mây để lưu trữ dữ liệu lớn.
* Phát triển ứng dụng mobile sử dụng cùng cơ sở dữ liệu.
* Tích hợp AI gợi ý sản phẩm, dự đoán xu hướng bán hàng.
* Kết nối API vận chuyển để xử lý giao hàng tự động.

***Chương 2: Phân tích hệ thống***

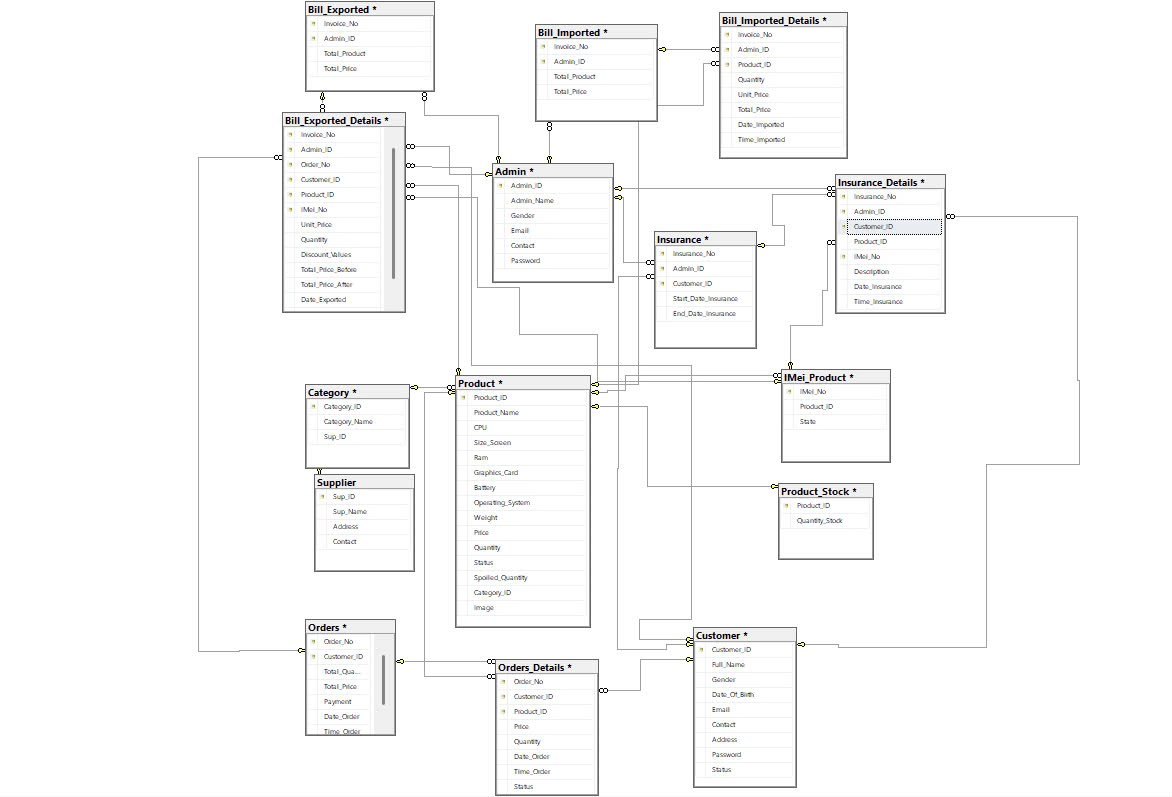
* 1. **Sơ đồ mức ngữ cảnh quy trình nghiệp vụ**

****

* **Sơ đồ thực thể kết hợp(ERD)**

****

* **Mapping table**

****

II. Use Case:

1. Xác định Use Csase:

* + - 1. Các module chính
* Quản lý nhập kho
* Quản lý hãng Laptop.
* Quản lý nhà cung cấp
* Quản lý hoá đơn
* Lập hoá đơn
* Quản lý khách hang
* Thống kê
  + - 1. Yêu cầu chức năng nghiệp vụ - Use Case:

1. Nhân viên bán hàng:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **STT** | **Công việc** | **Loại công việc** | **Quy đinh/công thức liên quan** | **Biểu mẫu liên quan** | **Ghi chú** |
| 1 | Thêm khách hàng | Lưu trữ | Mỗi khách hàng có 1 sđt duy nhất. |  |  |
| 2 | In hoá đơn | Lưu trữ | - Mỗi hoá đơn đều ghi nhận lại khách hàng mua, nhân viên nào lập hoá đơn  - Đảm bảo số lượng bán của loại Laptop phải bé hơn hoặc bằng số lượng còn trong kho. |  |  |
| 3 | Tra cứu Laptop | Tra cứu | - Tra cứu theo các ký tự đầu của tên Laptop |  |  |

1. Bộ phận - Quản lý

Quản lý có thể thực hiện các use case của Nhân viên

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **STT** | **Công việc** | **Loại công việc** | **Quy đinh/công thức liên quan** | **Biểu mẫu liên quan** | **Ghi chú** |
| 1 | Thêm Hãng Laptop | Lưu trữ |  |  |  |
|  | Thêm Nhà cung cấp |  |  |  |  |
|  | Thêm Loại Laptop |  |  |  |  |
|  | Thêm Laptop |  |  |  |  |
|  | Nhập Laptop vào kho |  |  |  |  |
|  | Quản lý nhân viên |  |  |  |  |
|  | Tra cứu khách hàng |  |  |  |  |
|  | Thống kê |  |  |  |  |
|  | Thống kê tổng doanh thu |  |  |  |  |

* + - 1. Yêu cầu chức năng hệ thống:

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Nội dung | Mô tả chi tiết | Ghi chú |
| 1 | Sao lưu, back up, phục hồi thông tin | Sao lưu thông tin những thông tin đã bị xoá và phục hôi khi cần thiết (Để khôi phục cần thao tác trực tiếp vào db) |  |
| 2 | Phân quyền sử dụng | Quản lý: Có thể thực hiện hết tất cả các chức năng nghiệp vụ của các user còn lại  Nhân viên bán hàng:  Chỉ thực hiện đúng chức năng nghiệp vụ đã được mô tả |  |

* + - 1. Yêu cầu phi chức năng:

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Nội dung | Tiêu chuẩn | Mô tả chi tiết |
| 1 | Tốc độ tìm kiếm nhanh và chính xác | Hiệu quả |  |
| 2 | Tiết kiệm được thời gian,  thu hẹp không gian lưu trữ, tránh thất lạc dữ liệu. | Hiệu quả | Tiết kiệm thời gian so với quản lý thủ công. Dữ liệu được sao lưu trên máy, có thể dễ dàng phục hồi |
| 3 | Giao diện đơn giản, dễ thao tác | Tiện dụng | Quy tắc 3 click |

2. Sơ đồ Use Case tổng quát:

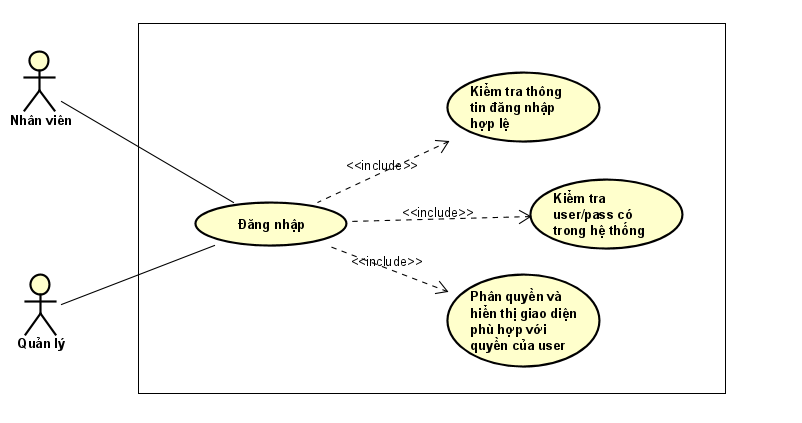
A diagram of a person with text

AI-generated content may be incorrect.

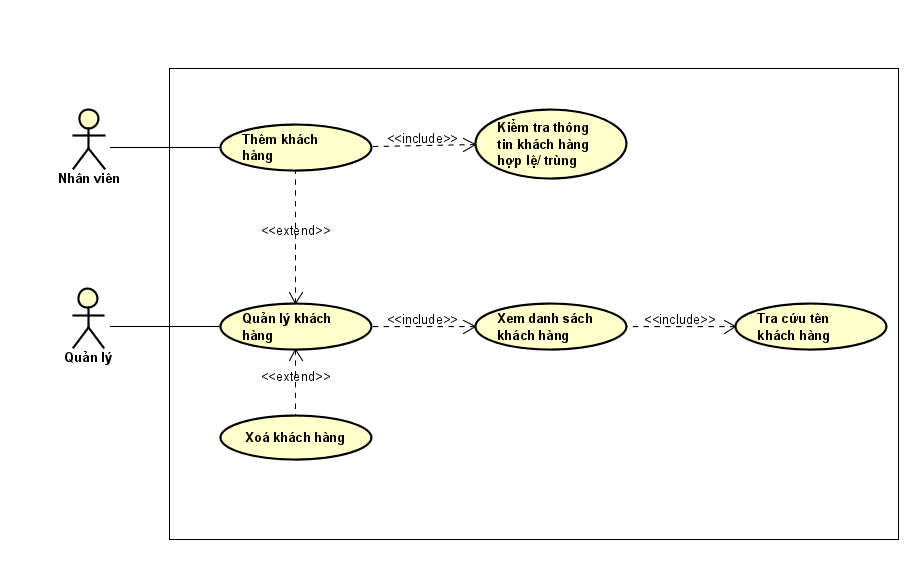
Ghi chú: Quản lý có toàn quyền sử dụng các chức năng của Nhân viên bán hàng.

3. Mô tả các Use Case:

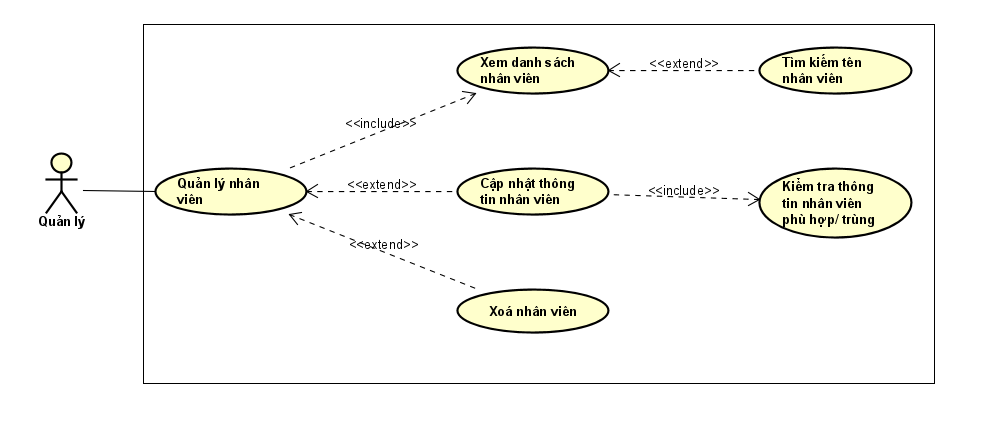
1. UC Đăng nhập:



1. UC Quản lý nhân viên



1. UC Quản lý khách hàng

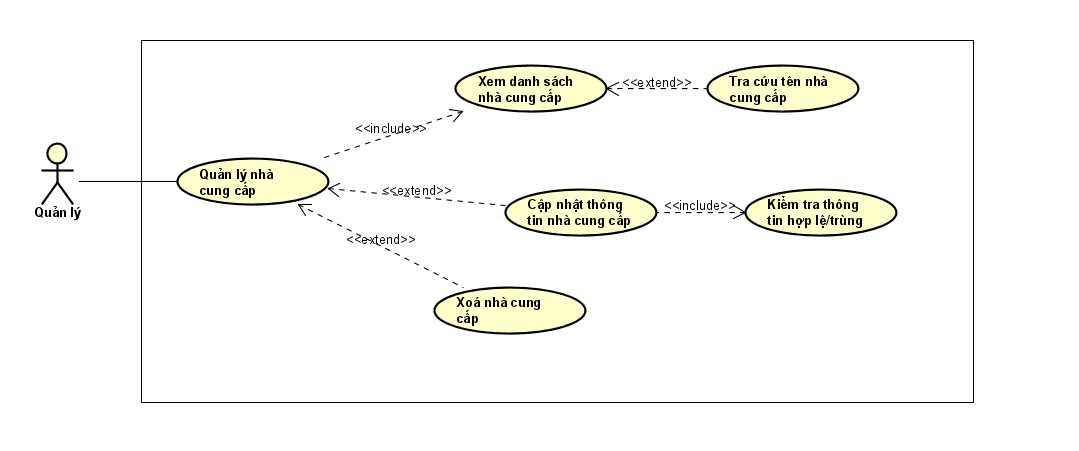


1. UC Quản lý danh mục Laptop

A diagram of a diagram

AI-generated content may be incorrect.

1. UC Quản lý nhà cung cấp



1. UC Quản lý loại Laptop

A diagram of a computer

AI-generated content may be incorrect.

1. UC Quản lý Laptop

A diagram of a company

AI-generated content may be incorrect.

1. UC Nhập kho

A diagram of a diagram

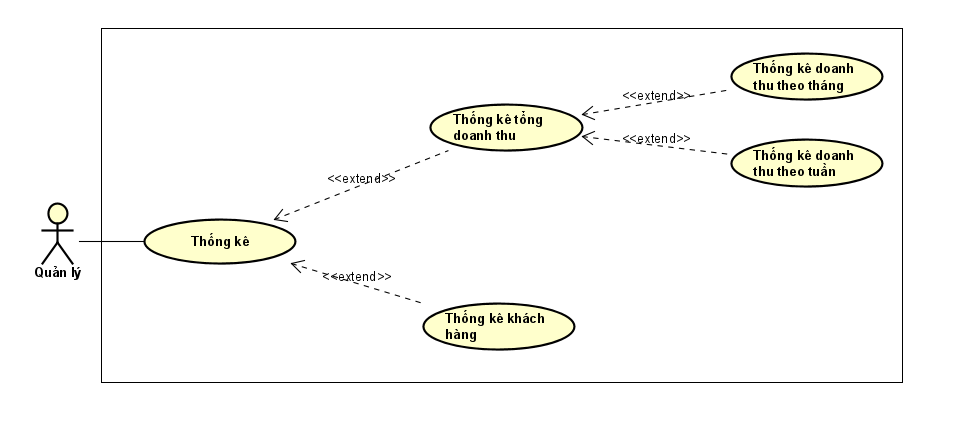
AI-generated content may be incorrect.

1. UC In hóa đơn

A diagram of a diagram

AI-generated content may be incorrect.

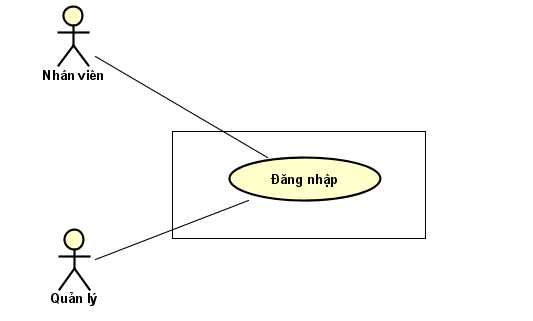
1. UC Thống kê



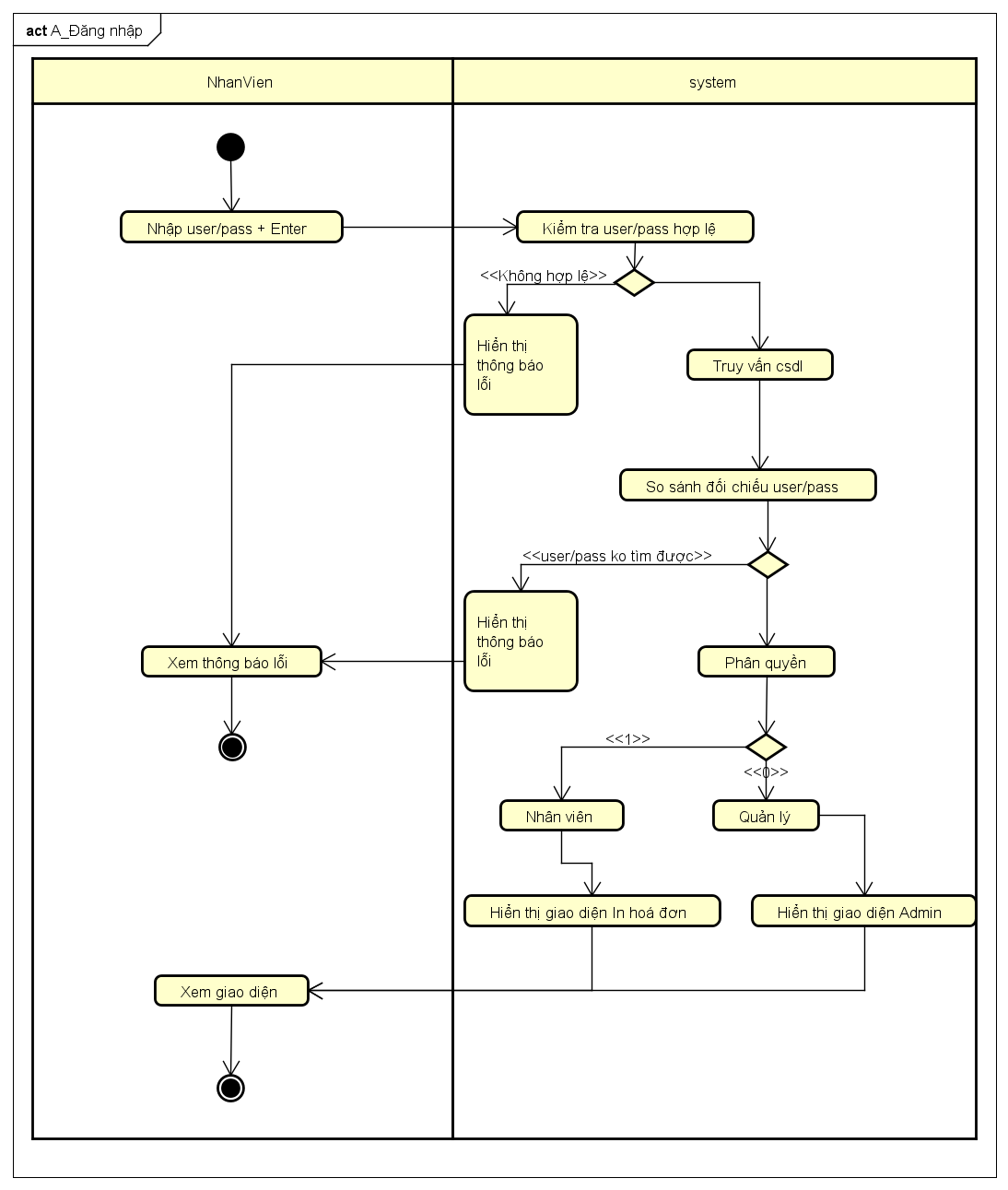
III/ Phân tích use case:

1. Phân tích thiết kế chức năng (UC – AC):

1.1. Đăng nhập

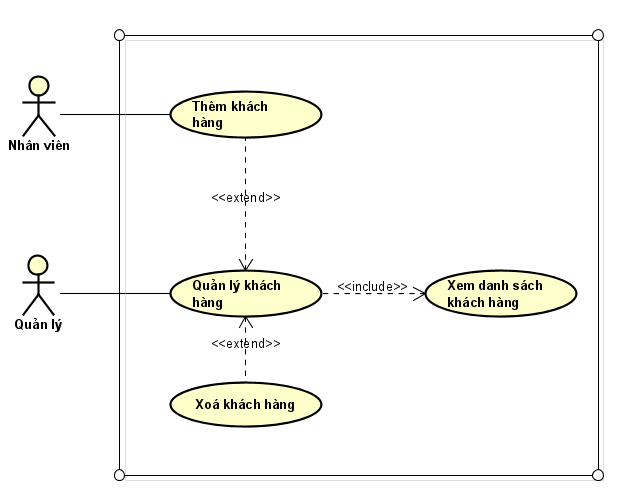


Mô tả: Nhân viên và quản lý trước khi sử dụng phần mềm phải đăng nhập vào hệ thống.



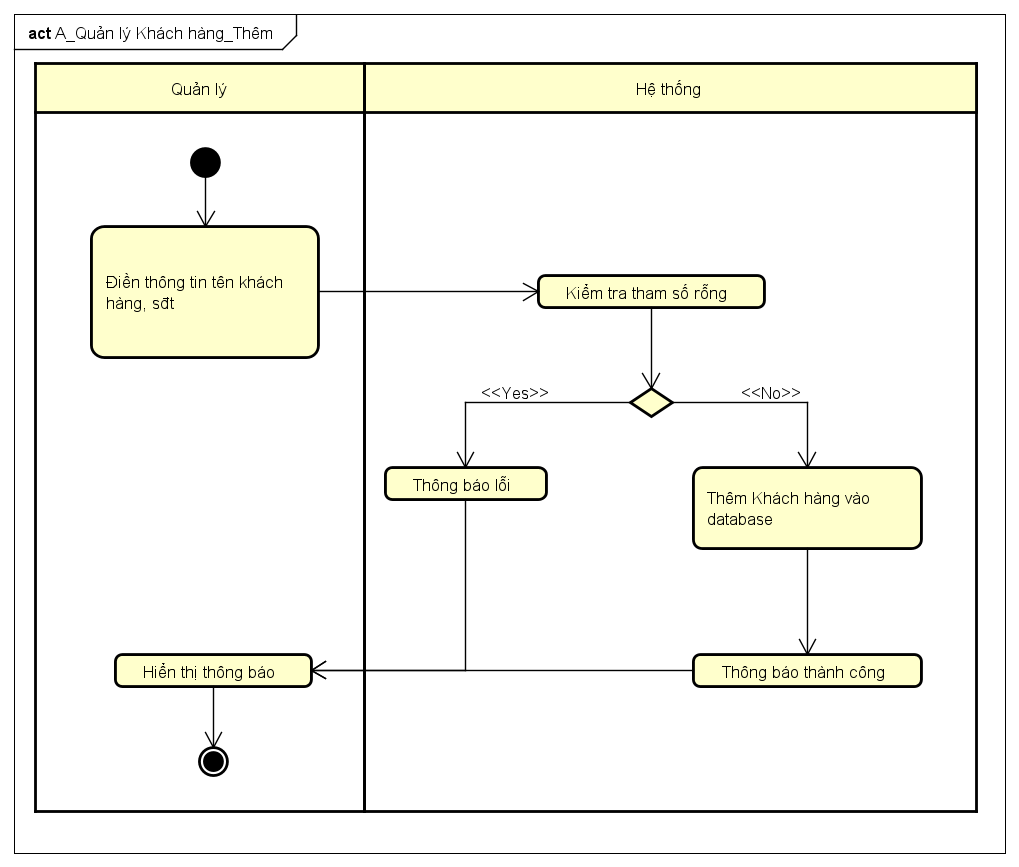
|  |  |  |
| --- | --- | --- |
| **Basic flow** | **Nhân viên, quản lý** | **Hệ thống** |
|  | 1. Nhân viên, quản lý nhập username/password và nhấn Đăng nhập |  |
|  | 2. Kiểm tra username/password hợp lệ, không bị rỗng |
|  | 3. Nếu tham số username/password hợp lệ, lấy đối tượng user trong database theo tham số username/password. |
|  | 4. Nếu user tồn tại, thực hiện phân quyền |
|  | 5. Nếu user có phân quyền là 0, mở trang admin |
| 6. Quản lý xem trang tương ứng |  |
| **Alternative flow 1** |  | 5.1. Nếu user có phân quyền là 1, mở trang In hóa đơn |
|  | 6.1.Nhân viên xem trang tương ứng |  |
| **Except flow 1** |  | 3.1 Tham số username/password không hợp lệ |
|  |  | 4.1 Hiển thị thông báo lỗi |
| 5.1. Người dùng xem thông báo lỗi |  |
| **Except flow 2** |  | 4.2. User không tồn tại hoặc username/password không đúng |
|  |  | 4.2. Chuyển sang bước thông báo lỗi 4.1 |

1.2. Quản lý Khách hàng



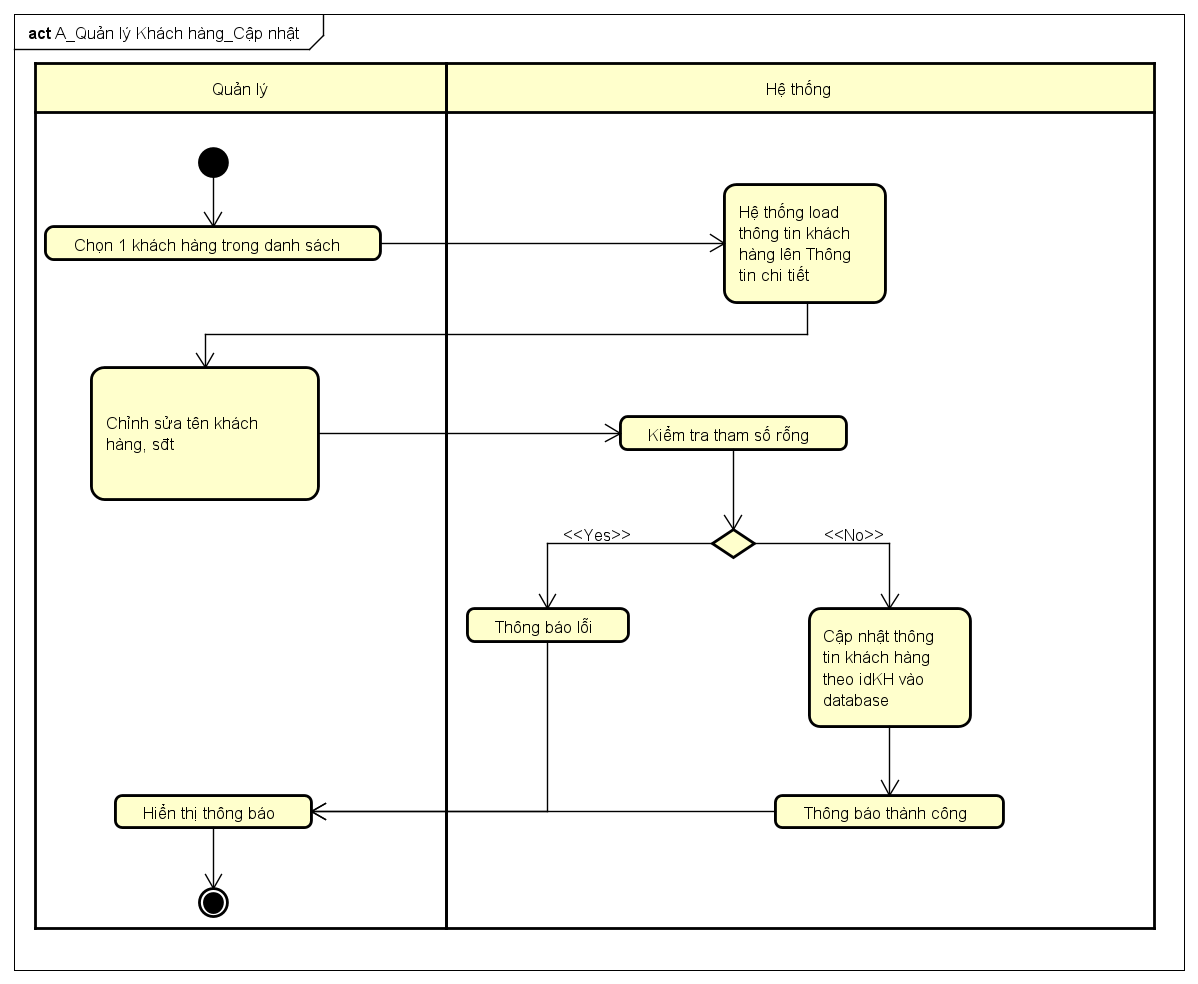
Mô tả: Nhân viên và quản lý đều có quyền thêm khác hàng vào hệ thống, quản lý còn có quyền xóa khách hàng.

A) Thêm khách hàng



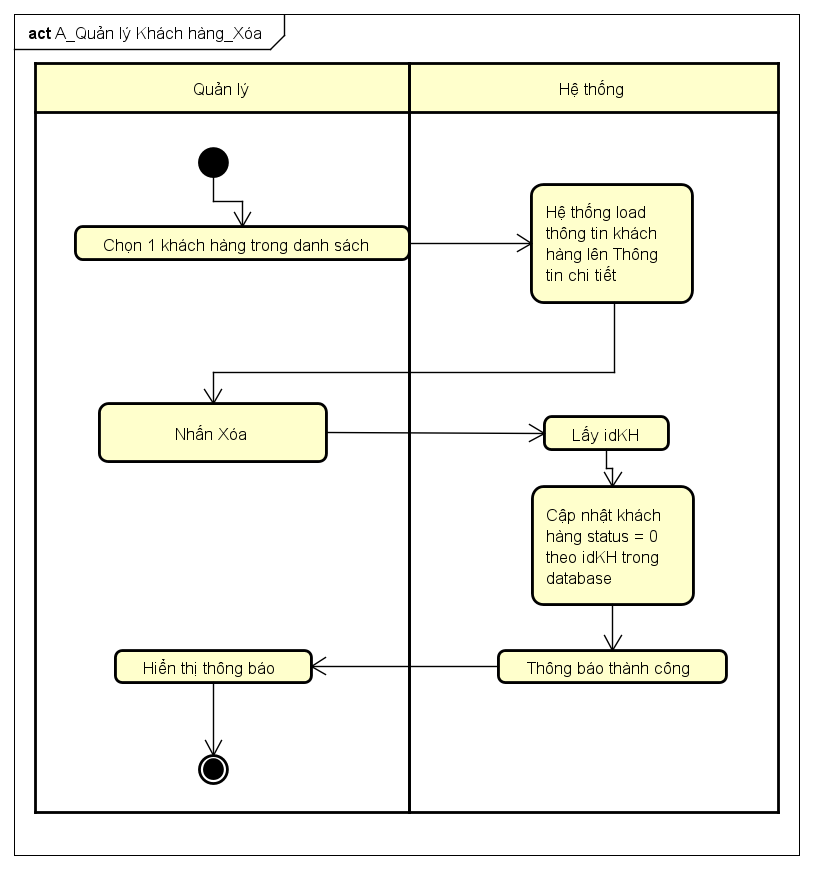
|  |  |  |
| --- | --- | --- |
| **Basic flow** | **Quản lý, nhân viên** | **Hệ thống** |
|  | 1.Quản lý, nhân viên nhập thông tin khách hàng: tên, số điện thoại |  |
|  | 2. Kiểm tra tham số hợp lệ, không bị rỗng |
|  | 3. Thêm thông tin khách hàng vào database |
|  | 4. Hiển thị thông báo thành công |
| 5. Quản lý, nhân viên xem trang thông báo |  |
| **Except flow** |  | 3.1 Tham số không hợp lệ |
|  |  | 4.1 Hiển thị thông báo lỗi |
| 5.1. Người dùng xem thông báo lỗi |  |

B) Cập nhật khách hàng



|  |  |  |
| --- | --- | --- |
| **Basic flow** | **Quản lý** | **Hệ thống** |
|  | 1.Quản lý chọn thông tin khách hàng |  |
|  | 2. Hiện thị thông tin khách hàng được chọn |
| 3.Chỉnh sửa thông tin khách hàng |  |
|  | 4. Kiểm tra thông số hợp lệ (không bị rỗng) |
|  | 5. Nếu tham số hợp lệ cập nhật thông tin khách hàng theo idKH vào hệ thống |
|  | 6. Hiển thị thông báo thành công |
| 7.Quản lý xem trang thông báo |  |
| **Except flow** |  | 5.1 Tham số không hợp lệ |
|  |  | 6.1 Hiển thị thông báo lỗi |
| 7.1. Người dùng xem thông báo lỗi |  |

C) Xóa khách hàng



|  |  |  |
| --- | --- | --- |
| **Basic flow** | **Quản lý** | **Hệ thống** |
|  | 1.Quản lý chọn khách hàng cần xóa |  |
|  | 2. Hiển thị thông tin chi tiết khách được chọn |
| 3. Nhấn xóa |  |
|  | 4. Lấy idKH của khách hàng |
|  | 5. Cập nhật status = 0 theo idKH đã chọn trong database |
|  | 6. Thông báo xóa thành công |
| 7. Quản lý xem trang thông báo |  |

1.3. Quản lý danh mục Laptop

A diagram of a diagram

AI-generated content may be incorrect.

Mô tả: Quản lý có quyền xem, thêm, cập nhật, xóa hãng Laptop

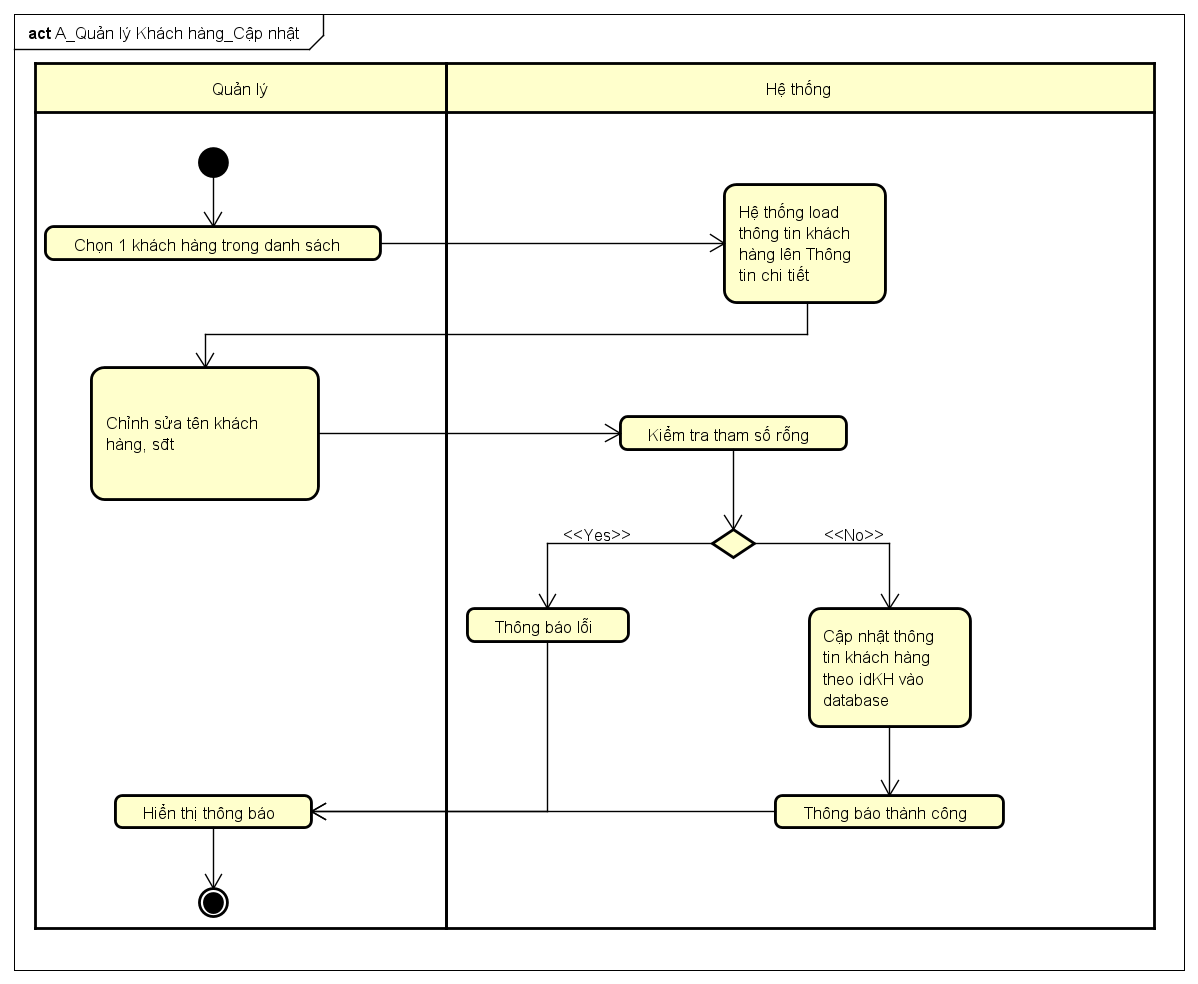
A) Thêm hãng Laptop

A diagram with text and words

AI-generated content may be incorrect.

|  |  |  |
| --- | --- | --- |
| **Basic flow** | **Quản lý** | **Hệ thống** |
|  | 1.Quản lý điền thông tin hãng Laptop cần thêm và nhấn Thêm |  |
|  | 2. Kiểm tra tham số hợp lệ, không bị rỗng |
|  | 3. Nếu tham số hợp lệ thêm thông tin hãng Laptop vào database |
|  | 4. Hiển thị thông báo thành công |
| 5. Quản lý xem trang thông báo |  |
| **Except flow** |  | 3.1 Tham số không hợp lệ |
|  |  | 4.1 Hiển thị thông báo lỗi |
| 5.1. Người dùng xem thông báo lỗi |  |

B) Cập nhật hàng Laptop



|  |  |  |
| --- | --- | --- |
| **Basic flow** | **Quản lý** | **Hệ thống** |
|  | 1.Quản lý nhấn chọn thông tin hãng Laptop cần chỉnh sửa |  |
|  | 2. Hiện thị thông tin hãng Laptop được chọn |
| 3. Quản lý chỉnh sửa thông tin hãng Laptop |  |
|  | 4. Kiểm tra tham số hợp lệ (không bị rỗng) |
|  | 5. Nếu hợp lệ thì cập nhật thông tin hãng Laptop theo id\_DanhMucLapTop vào hệ thống |
|  | 6. Thông báo thành công |
| 7. Người dùng xem thông báo thành công |  |
| **Except flow** |  | 5.1 Tham số không hợp lệ |
|  |  | 6.1 Hiển thị thông báo lỗi |
| 7.1. Người dùng xem thông báo lỗi |  |

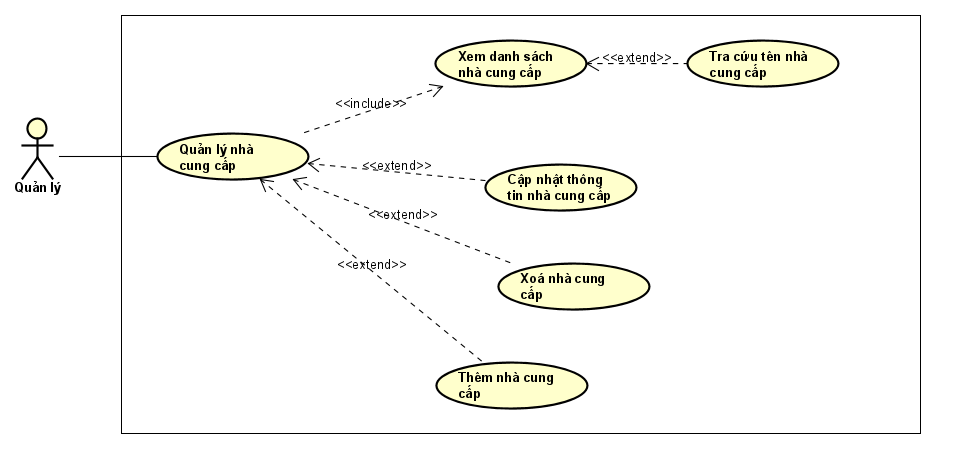
C) Xóa hãng Laptop

A diagram of a flowchart

AI-generated content may be incorrect.

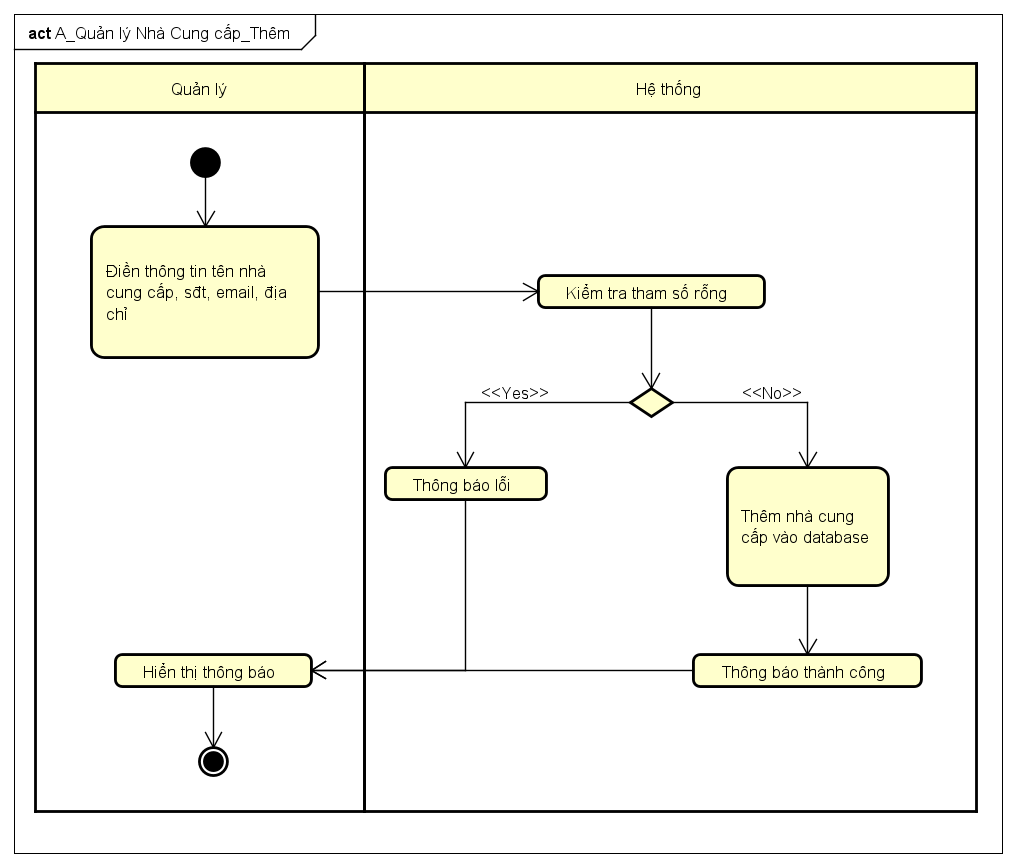
|  |  |  |
| --- | --- | --- |
| **Basic flow** | **Quản lý** | **Hệ thống** |
|  | 1.Quản lý chọn hãng Laptop cần xóa |  |
|  | 2. Load và hiển thị thông tin chi tiết hàng Laptop được chọn |
| 3. Nhấn xóa |  |
|  | 4. Lấy idHangGiay của hãng Laptop |
|  | 5. Cập nhật status = 0 theo id\_DanhMucLapTop đã chọn trong database |
|  | 6. Thông báo xóa thành công |
| 7. Quản lý xem trang thông báo |  |

1.5. Quản lý Nhà cung cấp



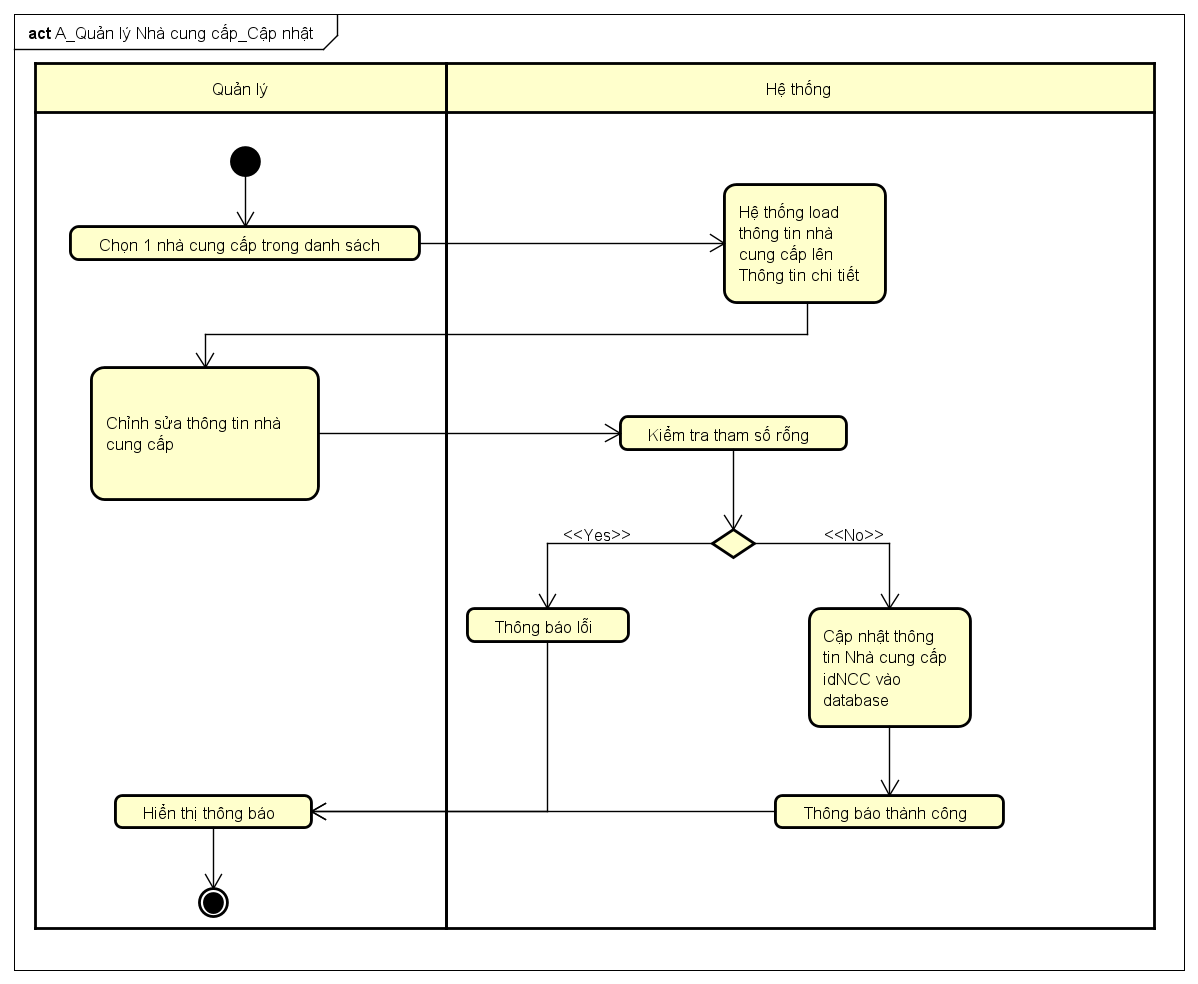
Mô tả: Quản lý có quyền xem danh sách nhà cung cấp, thêm, cập nhập, xóa thông tin các nhà cung cấp

A) Thêm nhà cung cấp



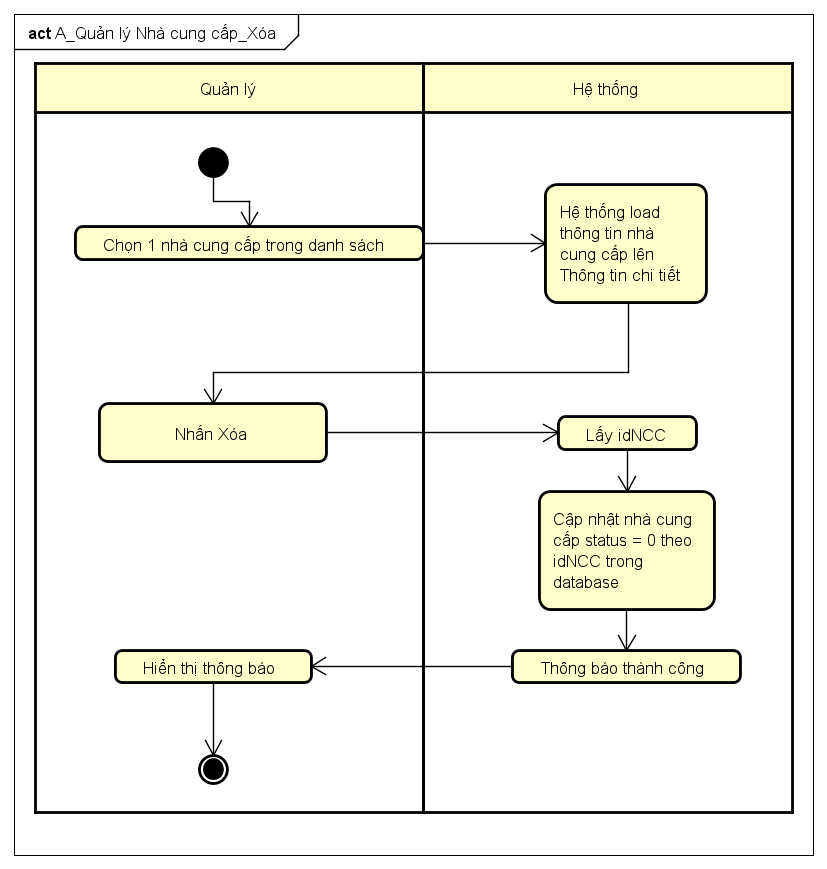
|  |  |  |
| --- | --- | --- |
| **Basic flow** | **Quản lý** | **Hệ thống** |
|  | 1.Quản lý điền thông tin nhà cung cấp: tên, số điện thoại, email, địa chỉ và nhấn Thêm |  |
|  | 2. Kiểm tra tham số hợp lệ, không bị rỗng |
|  | 3. Nếu tham số hợp lệ thêm thông tin nhà cung cấp vào database |
|  | 4. Hiển thị thông báo thành công |
| 5. Quản lý xem trang thông báo |  |
| **Except flow** |  | 3.1 Tham số không hợp lệ |
|  |  | 4.1 Hiển thị thông báo lỗi |
| 5.1. Người dùng xem thông báo lỗi |  |

B) Cập nhật nhà cung cấp



|  |  |  |
| --- | --- | --- |
| **Basic flow** | **Quản lý** | **Hệ thống** |
|  | 1.Quản lý nhấn chọn thông tin nhà cung cấp cần chỉnh sửa |  |
|  | 2. Load và hiện thị thông tin hãng Laptop được chọn |
| 3. Quản lý chỉnh sửa thông tin nhà cung cấp |  |
|  | 4. Kiểm tra tham số hợp lệ (không bị rỗng) |
|  | 5. Nếu hợp lệ thì cập nhật thông tin nhà cung cấp theo idNCC vào hệ thống |
|  | 6. Thông báo thành công |
| 7. Người dùng xem thông báo thành công |  |
| **Except flow** |  | 5.1 Tham số không hợp lệ |
|  |  | 6.1 Hiển thị thông báo lỗi |
| 7.1. Người dùng xem thông báo lỗi |  |

C) Xóa nhà cung cấp



|  |  |  |
| --- | --- | --- |
| **Basic flow** | **Quản lý** | **Hệ thống** |
|  | 1.Quản lý chọn nhà cung cấp cần xóa |  |
|  | 2. Load và hiển thị thông tin chi tiết nhà cung cấp được chọn |
| 3. Nhấn xóa |  |
|  | 4. Lấy idNCC của nhà cung cấp |
|  | 5. Cập nhật status = 0 theo idNCC đã chọn trong database |
|  | 6. Thông báo xóa thành công |
| 7. Quản lý xem trang thông báo |  |

1.6. Quản lý Loại Laptop

A diagram of a computer network

AI-generated content may be incorrect.

Mô tả: Quản lý có quyền xem danh sách loại Laptop, thêm, cập nhật, xóa thông tin các loại Laptop

A) Thêm loại Laptop

A diagram of a workflow

AI-generated content may be incorrect.

|  |  |  |
| --- | --- | --- |
| **Basic flow** | **Quản lý** | **Hệ thống** |
|  | 1.Quản lý điền thông tin loại Laptop: size, màu sắc, số lượng, giá bán và nhấn Thêm |  |
|  | 2. Kiểm tra tham số hợp lệ, không bị rỗng |
|  | 3. Nếu tham số hợp lệ thêm thông tin loại Laptop vào database |
|  | 4. Hiển thị thông báo thành công |
| 5. Quản lý xem trang thông báo |  |
| **Except flow** |  | 3.1 Tham số không hợp lệ |
|  |  | 4.1 Hiển thị thông báo lỗi |
| 5.1. Người dùng xem thông báo lỗi |  |

B) Cập nhật loại Laptop

A diagram with text and words

AI-generated content may be incorrect.

|  |  |  |
| --- | --- | --- |
| **Basic flow** | **Quản lý** | **Hệ thống** |
|  | 1.Quản lý nhấn chọn loại Laptop cần chỉnh sửa |  |
|  | 2. Load và hiện thị thông tin loại Laptop được chọn |
| 3. Quản lý chỉnh sửa thông tin loại Laptop: số lượng, giá bán |  |
|  | 4. Kiểm tra tham số hợp lệ (không bị rỗng) |
|  | 5. Nếu hợp lệ thì cập nhật thông tin theo id\_LoaiLapTop vào hệ thống |
|  | 6. Thông báo thành công |
| 7. Người dùng xem thông báo thành công |  |
| **Except flow** |  | 5.1 Tham số không hợp lệ |
|  |  | 6.1 Hiển thị thông báo lỗi |
| 7.1. Người dùng xem thông báo lỗi |  |

C) Xóa loại Laptop

A diagram with text and black circles

AI-generated content may be incorrect.

|  |  |  |
| --- | --- | --- |
| **Basic flow** | **Quản lý** | **Hệ thống** |
|  | 1.Quản lý chọn loại Laptop trong danh sách cần xóa |  |
|  | 2. Load và hiển thị thông tin chi tiết loại Laptop được chọn |
| 3. Nhấn xóa |  |
|  | 4. Lấy idLoaigiay |
|  | 5. Cập nhật Loại Laptop status = 0 theo id\_LoaiLapTop đã chọn trong database |
|  | 6. Thông báo xóa thành công |
| 7. Quản lý xem trang thông báo |  |

1.7 Quản lý Laptop

A diagram of a software company

AI-generated content may be incorrect.

Mô tả: Quản lý có quyền xem, thêm, cập nhật, xóa Laptop

A) Thêm Laptop

A diagram with text and words

AI-generated content may be incorrect.

|  |  |  |
| --- | --- | --- |
| **Basic flow** | **Quản lý** | **Hệ thống** |
|  | 1.Quản lý điền thông tin Laptop: số lượng, giá bán và nhấn Thêm |  |
|  | 2. Kiểm tra tham số hợp lệ, không bị rỗng |
|  | 3. Nếu tham số hợp lệ thêm thông tin Laptop vào database |
|  | 4. Hiển thị thông báo thành công |
| 5. Quản lý xem trang thông báo |  |
| **Except flow** |  | 3.1 Tham số không hợp lệ |
|  |  | 4.1 Hiển thị thông báo lỗi |
| 5.1. Người dùng xem thông báo lỗi |  |

B) Cập nhật Laptop

A diagram of a flowchart

AI-generated content may be incorrect.

|  |  |  |
| --- | --- | --- |
| **Basic flow** | **Quản lý** | **Hệ thống** |
|  | 1.Quản lý nhấn chọn Laptop cần chỉnh sửa |  |
|  | 2. Load và hiện thị thông tin Laptop được chọn |
| 3. Quản lý chỉnh sửa thông tin loại Laptop: số lượng, giá bán |  |
|  | 4. Kiểm tra tham số hợp lệ (không bị rỗng) |
|  | 5. Nếu hợp lệ thì cập nhật thông tin theo idLaptop vào hệ thống |
|  | 6. Thông báo thành công |
| 7. Người dùng xem thông báo thành công |  |
| **Except flow** |  | 5.1 Tham số không hợp lệ |
|  |  | 6.1 Hiển thị thông báo lỗi |
| 7.1. Người dùng xem thông báo lỗi |  |

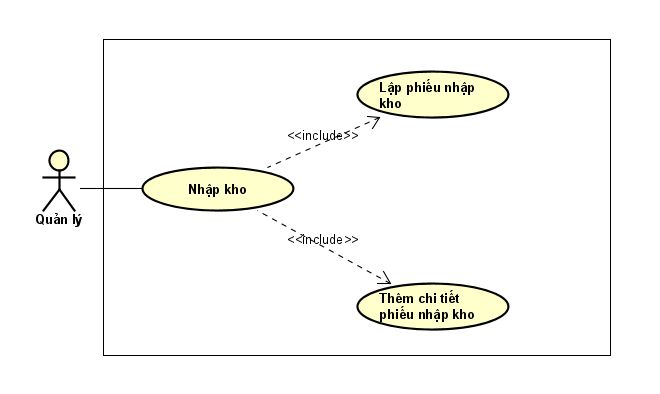
C) Xóa Laptop

A diagram of a computer

AI-generated content may be incorrect.

|  |  |  |
| --- | --- | --- |
| **Basic flow** | **Quản lý** | **Hệ thống** |
|  | 1.Quản lý chọn Laptop trong danh sách cần xóa |  |
|  | 2. Load và hiển thị thông tin chi tiết Laptop được chọn |
| 3. Nhấn xóa |  |
|  | 4. Lấy idGiay |
|  | 5. Cập nhật Laptop status = 0 theo idLaptop đã chọn trong database |
|  | 6. Thông báo xóa thành công |
| 7. Quản lý xem trang thông báo |  |

1.8. Nhập kho



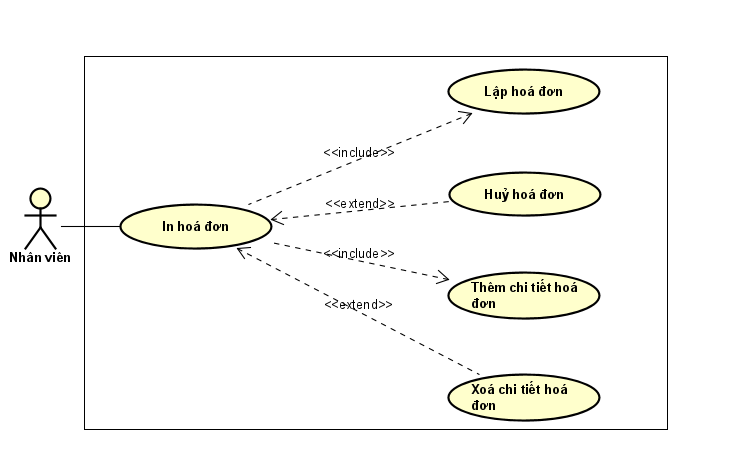
Mô tả: Quản lý tạo phiếu nhập kho sau đó thêm chi tiết từng loại Laptop vào phiếu nhập kho

A screenshot of a diagram

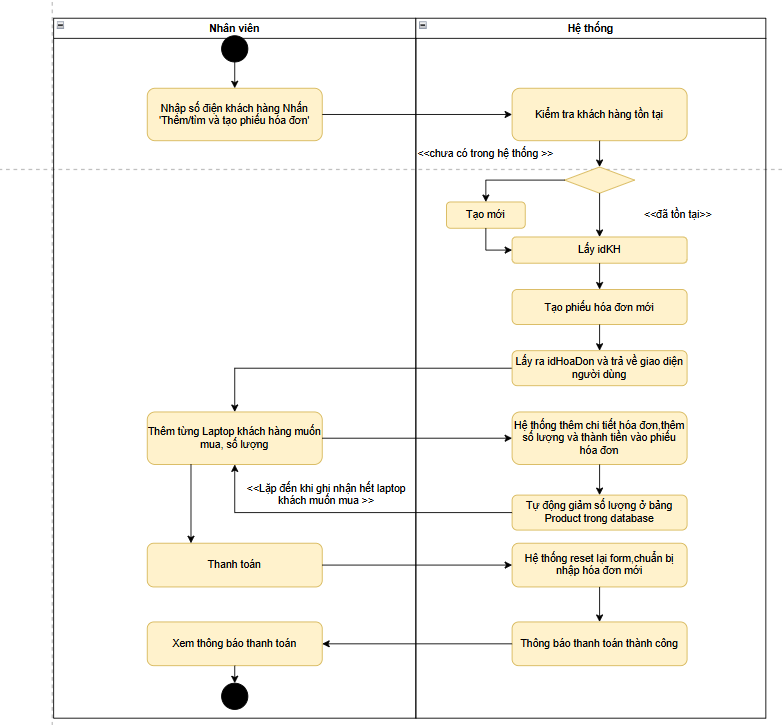
AI-generated content may be incorrect.

|  |  |  |
| --- | --- | --- |
| **Basic flow** | **Quản lý** | **Hệ thống** |
|  | 1.Quản lý nhấn vào tạo phiếu nhập kho |  |
|  | 2.Thực thi tạo phiếu nhập kho |
|  | 3.Trả ra thông tin hóa đơn vừa tạo ra |
| 4.Người dùng nhấn vào Nhập chi tiết nhập kho |  |
|  | 5.Hiển thị ra giao diện của nhập hóa đơn chi tiết nhập kho |
| 6.Người dùng nhập vào bảng hiện ra: Laptop ,size,số lượng,giá và nhấn Thêm |  |
|  | 7.Thêm Laptop vào CSDL |
|  | 8. Cộng dồn số lượng,thành tiền vào phiếu nhập kho chính |
|  | 9. Tự cập nhật số lượng Laptop vào trong bảng Giay trong database |
| 10.Nhấn x để kết thúc Thêm |  |

1.9. In hóa đơn

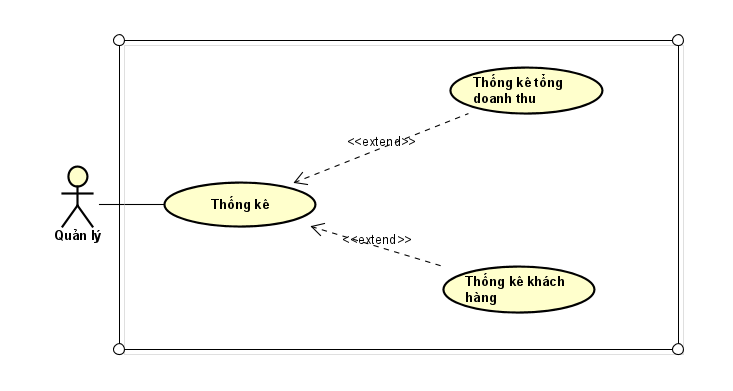


Mô tả: Nhân viên có các quyền lập hóa đơn mới, hủy hóa đơn, thêm, xóa chi tiết hóa đơn

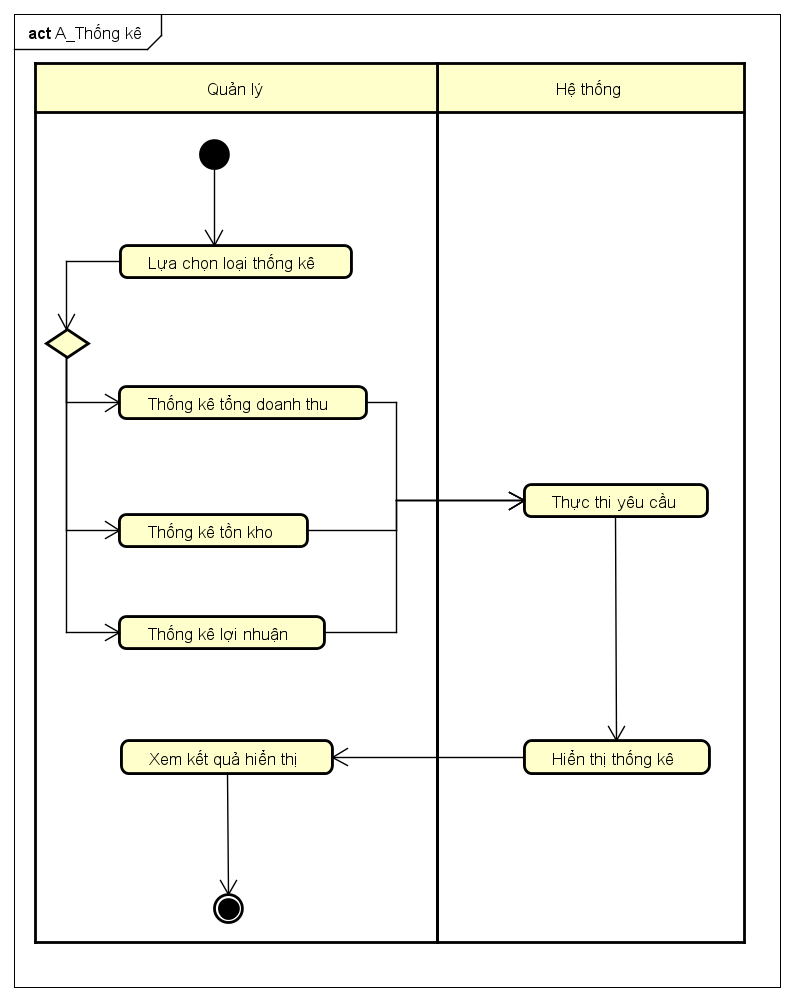


|  |  |  |
| --- | --- | --- |
| **Basic Flow** | **Nhân Viên** | **Hệ thống** |
|  | 1.Nhập sđt/tên khách hàng và nhấn “Thêm/tìm vào tạo phiếu hóa đơn |  |
|  | 2.Kiểm tra khách hàng nhập có tồn tại hay không |
|  | 3.1. Nếu đã tồn tại thì sang bước 4 |
|  | 3.2 Nếu chưa có trong hệ thống thì thực hiện tạo mới và chuyển sang bước 4 |
|  | 4.Lấy IDKH |
|  | 5.Tạo phiếu hóa đơn mới |
|  | 6.Lấy ra ID hóa đơn và trả về giao diện của người dùng |
| 7.Thêm từng Laptop,số lượng khách hàng muốn mua |  |
|  | 8.Hệ thống thêm vào chi tiết hóa đơn ,thêm số lượng và thành tiền vào phiếu hóa đơn |
|  | 9.Tự động giảm số lượng theo tên Laptop trong bảng Laptop của database |
| 10. Nhấn vào Thanh Toán |  |
|  | 11.Hệ thống reset lại form, chuẩn bị nhập hóa đơn mới |
|  | 12.Thông báo thanh toán thành công |
| 12.Xem thông báo thanh toán |  |

1.10. Thống kê



Mô tả: Quản lý thống kê tổng danh thu của cửa hàng và top những nhà cung cấp, danh mục và sản phẩm được ưa chuộng mua nhiều nhất bởi khách hàng



|  |  |  |
| --- | --- | --- |
| **Basic Flow** | **Quản lý** | **Hệ thống** |
|  | 1.Lựa chọn loại thống kê qua chọn 1 trong các bước 2, 3 ,4 |  |
| 2.Thống kê tổng doanh thu |  |
| 3.Thống kê tồn kho |  |
| 4.Thống kê lợi nhuận |  |
|  | 5.Nhận yêu cầu và thực thi yêu cầu |
|  | 6.Hiển thị thống kê |
| 7.Xem kết quả hiển thị |  |

***Chương 3: Quy trình phát triển phần mềm(quy trình Agile)***

1. **Giới thiệu tổng quan quy trình Agile**
2. **Khái niệm**

* Agile (viết tắt của Agile Software Development) là một phương thức phát triển phần mềm linh hoạt, được thực hiện bằng cách sử dụng các bước lặp ngắn từ 1 đến 4 tuần. Mục tiêu của Agile là giúp rút ngắn thời gian phát triển sản phẩm, đưa sản phẩm đến với tay khách hàng càng sớm càng tốt.
* Về bản chất, Agile giống như một phương pháp luận, một triết lý dựa trên nguyên tắc phân đoạn vòng lặp (iterative) và tăng trưởng (incremental) nên sở hữu tính linh hoạt cao. Tính chất này đi ngược lại với các phương pháp quản lý dự án truyền thống – vốn dĩ triển khai các giai đoạn một cách tuyến tính và vô cùng bị động trước các thay đổi bất ngờ.

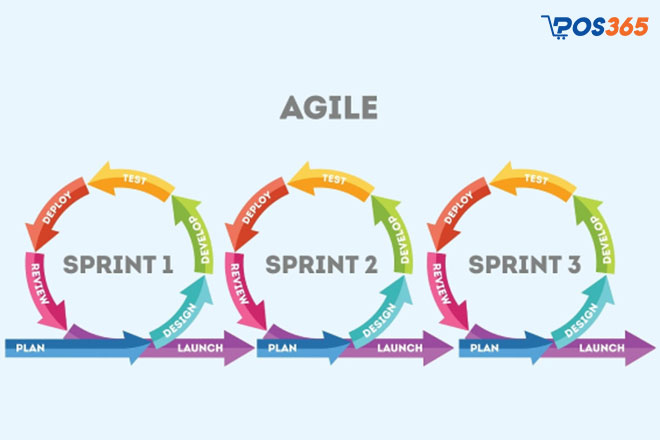
1. **Nguồn gốc**

* Trước khi có sự xuất hiện của Agile, mô hình Waterfall được xem như tiêu chuẩn vàng để phát triển phần mềm. Thông thường, quy trình sẽ bắt đầu bằng việc một nhà phân tích kinh doanh viết tài liệu yêu cầu nghiệp vụ, mô tả những yêu cầu mà họ cần từ phần mềm. Những tài liệu này thường rất dài và chi tiết, chứa đựng tất cả mọi thông tin từ chiến lược tổng thể đến các thông số kỹ thuật. Sau khi tất cả các tài liệu hoàn tất, các nhà phát triển mới bắt đầu code phần mềm, hoàn thiện nó và test. Tất cả quy trình này có thể mất đến một vài năm để kết thúc và để phần mềm được đưa vào sử dụng.
* Việc phát triển các phần mềm bắt đầu thay đổi với sự phát triển của mạng Internet. Đây cũng là lúc mà hiệu quả của phương pháp Waterfall bị đặt một dấu chấm hỏi. Nó làm tiêu tốn quá nhiều thời gian và công sức của nhóm thực hiện, trong khi lại thiếu đi tính khả thi khi nó chỉ hoàn toàn được xây dựng từ các nghiên cứu mà thiếu sự phản hồi thực tế từ người dùng.
* Vào năm 2001, một nhóm gồm 17 nhà phát triển phần mềm kinh nghiệm, bao gồm Kent Beck, Martin Fowler, Ron Jeffries, Ken Schwaber và Jeff Sutherland, đã cùng nhau nghiên cứu và xuất bản Tuyên ngôn Agile với 4 giá trị quan trọng và 12 nguyên tắc hướng dẫn cách tiếp cận lặp đi lặp lại và tập trung vào con người để phát triển phần mềm. Tuyên ngôn Agile được xem như là niềm tin của họ vào cách vận hành hoàn hảo mà quá trình phát triển phần mềm hiện đại nên có.

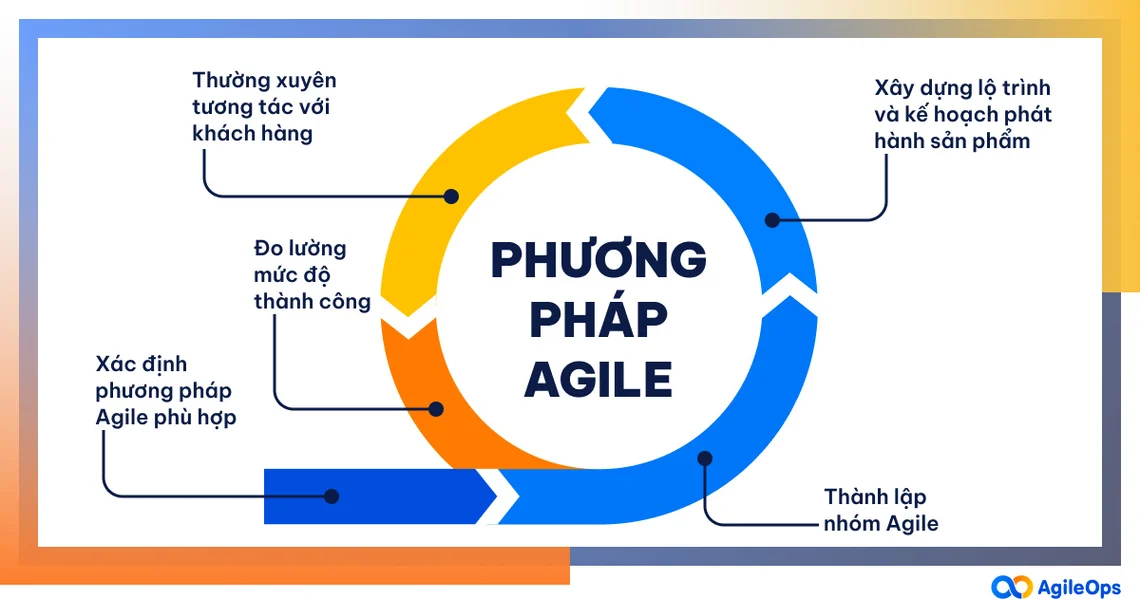
1. **Đặc trưng**

* **Tính lặp (Iterative):** Dự án sẽ được thực hiện trong các phân đoạn lặp đi lặp lại, thường có khung thời gian ngắn (từ 1-4 tuần). Trong mỗi phân đoạn đó, nhóm phát triển dự án sẽ thực hiện đầy đủ các công việc cần thiết như lập kế hoạch, phân tích yêu cầu, thiết kế, triển khai, kiểm thử để cho ra các phần nhỏ của sản phẩm.
* **Tính tăng trưởng và tiến hóa (Incremental & Evolutionary):** Cuối các phân đoạn, nhóm cho ra các phần nhỏ của sản phẩm cuối cùng, thường là đầy đủ, có khả năng chạy tốt, được kiểm thử cẩn thận và có thể sử dụng. Theo thời gian, phân đoạn này tiếp nối phân đoạn kia, các phần chạy được này sẽ được tích lũy, lớn dần lên cho tới khi toàn bộ yêu cầu của khách hàng được thỏa mãn.
* **Tính thích nghi (adaptive):** Do các phân đoạn chỉ kéo dài trong một khoảng thời gian ngắn và việc lập kế hoạch cũng được điều chỉnh liên tục, nên các thay đổi trong quá trình phát triển (yêu cầu thay đổi, thay đổi công nghệ, thay đổi định hướng về mục tiêu,…) đều có thể được đáp ứng theo cách thích hợp.
* **Nhóm tự tổ chức và liên chức năng:** Các cấu trúc nhóm này tự phân công công việc mà không dựa trên các mô tả cứng nhắc về chức danh hay một sự phân cấp rõ ràng. Nhóm tự tổ chức đã đủ các kỹ năng cần thiết để có thể được trao quyền tự ra quyết định, tự quản lý và tổ chức lấy công việc của chính mình để đạt được hiệu quả cao nhất.
* **Quản lý tiến trình thực nghiệm (Empirical Process Control):** Các nhóm Agile ra các quyết định dựa trên các dữ liệu thực tiễn (data-driven) thay vì tính toán lý thuyết hay các tiền giả định. Agile rút ngắn vòng đời phản hồi để dễ dàng thích nghi và gia tăng tính linh hoạt nhờ đó có thể kiểm soát được tiến trình, và nâng cao năng suất lao động.
* **Giao tiếp trực diện (face-to-face communication):** Agile không phản đối việc tài liệu hóa, nhưng đánh giá cao hơn việc giao tiếp trực diện thay vì thông qua giấy tờ. Agile khuyến khích nhóm phát triển trực tiếp nói chuyện để hiểu rõ hơn về những gì khách hàng thực sự cần. Trong giao tiếp giữa nội bộ nhóm, Agile khuyến khích trực tiếp trao đổi và thống nhất với nhau về thiết kế của hệ thống và cùng nhau triển khai thành các chức năng theo yêu cầu.
* **Phát triển dựa trên giá trị (value-based development):** Một trong các nguyên tắc cơ bản của Agile là “sản phẩm chạy tốt chính là thước đo của tiến độ”. Nhóm Agile thường cộng tác trực tiếp và thường xuyên với khách hàng để biết yêu cầu nào có độ ưu tiên cao hơn, mang lại giá trị hơn sớm nhất có thể cho dự án.

1. **Sơ đồ hoạt động của quy trình Agile**



*Hình 2.1: Quy trình Agile được thực hiện theo Sprint*

****

*Hình 2.2: Thể hiện phương pháp Agile*

1. **Ứng dụng quy trình Agile cho phần mềm Quản lý bán hàng (Phần Admin)**
2. **Giai đoạn khởi đầu – Xây dựng Product Backlog**

* Product Owner (PO) phối hợp với các bên liên quan như admin, nhân viên kinh doanh, nhà quản lý để liệt kê toàn bộ tính năng mà hệ thống cần có.
* Các yêu cầu này được sắp xếp theo mức độ ưu tiên, đưa vào danh sách gọi là Product Backlog.
* **Các Epic chính (chức năng cấp cao):**

|  |  |
| --- | --- |
| **Mã Epic** | **Chức năng** |
| EP01 | Quản lý danh mục và sản phẩm |
| EP02 | Xử lý đơn hàng |
| EP03 | Quản lý kho và nhập hàng |
| EP04 | Thống kê và báo cáo |
| EP05 | Gửi email tự động cho khách hàng |
| EP06 | Bảo mật, hiệu năng, khả năng mở rộng |

1. **Sprint Planning – Lập kế hoạch Sprint 1**

* Mục tiêu: Hoàn thiện chức năng quản lý danh mục và sản phẩm.
* Thời gian: Sprint 1 kéo dài 2 tuần.
* **Ví dụ các User Story trong Sprint 1:**
* Là admin, tôi muốn thêm danh mục sản phẩm mới để dễ dàng quản lý các nhóm hàng hóa.
* Là admin, tôi muốn chỉnh sửa hoặc xóa danh mục để cập nhật dữ liệu chính xác.
* Là admin, tôi muốn thêm sản phẩm với ảnh, thông số kỹ thuật và giá bán.

→ Kết quả kỳ vọng: Module quản lý danh mục và sản phẩm hoàn thiện và chạy được.

1. **Thực hiện Sprint (Iterative Development)**

Nhóm được chia nhỏ:

* Frontend: Java Swing (desktop).
* Backend: Java theo mô hình MVC.

|  |  |
| --- | --- |
| **Ngày** | **Công việc** |
| Ngày 1 | Thiết kế UI danh mục & sản phẩm |
| Ngày 2–4 | Code các chức năng CRUD (thêm/sửa/xóa) |
| Ngày 5–6 | Tích hợp frontend ↔ backend |
| Ngày 7–10 | Viết test case, kiểm thử nội bộ, sửa lỗi |

1. **Daily Scrum – Họp mỗi ngày 15 phút**

Trả lời 3 câu hỏi:

* Hôm qua tôi đã làm gì?
* Hôm nay tôi sẽ làm gì?
* Tôi gặp trở ngại gì không?

→ Mục tiêu: Nắm bắt tiến độ, hỗ trợ nhau, ngăn chặn tắc nghẽn sớm.

1. **Sprint Review – Trình bày kết quả cho PO**

Nhóm demo:

* Giao diện thêm/xóa/sửa danh mục, sản phẩm.
* Tìm kiếm sản phẩm theo tên hoặc mã.
* Dữ liệu lưu trữ ổn định trên SQL Server.

**Phản hồi từ PO:** "Thêm tính năng xem trước ảnh sản phẩm trước khi lưu."

→ Tính năng này sẽ được đưa trở lại vào Product Backlog.

1. **Sprint Retrospective – Cải tiến quy trình**

* **Tốt:** Nhóm giao tiếp tốt, chia task hợp lý.
* **Cần cải thiện:** Thời gian kiểm thử gấp → dành thêm 1.5 ngày cho test trong Sprint sau.

1. **Kế hoạch các Sprint tiếp theo**

|  |  |  |
| --- | --- | --- |
| **Sprint** | **Mục tiêu chính** | **Kết quả mong đợi** |
| Sprint 2 | Xử lý đơn hàng, tạo hóa đơn, áp dụng mã KM | Xác nhận đơn, tạo phiếu xuất, áp dụng khuyến mãi |
| Sprint 3 | Nhập hàng và quản lý kho | Gán IMEI, nhập hàng loạt, kiểm tra tồn kho |
| Sprint 4 | Gửi Email & bảo mật hệ thống | Gửi mail đơn hàng, mã hóa mật khẩu, phân quyền |
| Sprint 5 | Tổng kiểm thử và triển khai sản phẩm | Test toàn bộ hệ thống, viết hướng dẫn người dùng |
| Sprint 6 | Tạo phiếu nhập hàng | Tạo phiếu nhập, cập nhật kho, gán mã IMEI tự động |
| Sprint 7 | Xử lý order khách hàng & tạo phiếu xuất | Xác nhận đơn, xuất hàng, ghi nhận IMEI, cập nhật tồn kho |
| Sprint 8 | Thống kê & báo cáo nâng cao | Biểu đồ doanh thu, lọc theo danh mục, xuất báo cáo PDF/Excel |

1. **Sprint 6: Tạo phiếu nhập hàng và cập nhật hàng hóa về cửa hàng.**

|  |  |
| --- | --- |
| **Nội dung** | **Chi tiết** |
| Mục tiêu | Cho phép admin tạo phiếu nhập hàng khi sản phẩm sắp hết kho |
| Chức năng phát triển | - Tạo phiếu nhập (chọn danh mục, nhà cung cấp, số lượng, đơn giá)  - Gán mã IMEI tự động cho sản phẩm mới nhập  - Cập nhật kho sau khi nhập hàng thành công |
| User Story | - Là admin, tôi muốn tạo phiếu nhập để bổ sung kho khi sản phẩm sắp hết.  - Là admin, tôi muốn mỗi sản phẩm nhập có mã IMEI riêng để dễ theo dõi. |
| Kết quả mong đợi | Hệ thống có thể tạo phiếu nhập, cập nhật số lượng tồn kho theo danh mục. |

1. **Sprint 7: Xử lý đơn đặt hàng của khách và tạo phiếu xuất hàng**

|  |  |
| --- | --- |
| **Nội dung** | **Chi tiết** |
| Mục tiêu | Xử lý các đơn hàng từ khách hàng và tạo phiếu xuất hàng kèm thông tin IMEI |
| Chức năng phát triển | - Danh sách đơn hàng trạng thái "Waiting" - Xác nhận hoặc từ chối đơn hàng - Tạo phiếu xuất hàng (gồm mã IMEI, giá gốc, giá khuyến mãi) - Trừ hàng tồn kho tương ứng |
| User Story | - Là admin, tôi muốn xác nhận đơn hàng và xuất kho tương ứng. - Là admin, tôi muốn theo dõi đơn hàng bị bom từ khách hàng. |
| Kết quả mong đợi | Đơn hàng được xử lý, tạo phiếu xuất hợp lệ, trừ kho, và lưu thông tin IMEI. |

1. **Sprint 8: Thống kê và báo cáo nâng cao**

|  |  |
| --- | --- |
| **Nội dung** | **Chi tiết** |
| Mục tiêu | Cung cấp báo cáo chi tiết về hoạt động kinh doanh và kho hàng |
| Chức năng phát triển | - Thống kê doanh thu theo ngày/tháng/năm  - Thống kê theo danh mục, sản phẩm bán chạy  - Biểu đồ hóa dữ liệu bằng thư viện JFreeChart  - Xuất báo cáo PDF hoặc Excel |
| User Story | - Là admin, tôi muốn xem biểu đồ doanh thu để đánh giá hiệu quả kinh doanh.  - Là admin, tôi muốn lọc báo cáo theo từng danh mục hoặc sản phẩm. |
| Kết quả mong đợi | |  | | --- | |  |  |  | | --- | | Biểu đồ doanh thu rõ ràng, báo cáo dễ xuất, hỗ trợ quyết định chiến lược. | |

1. **Công cụ hỗ trợ Agile**

|  |  |
| --- | --- |
| **Mục đích** | **Công cụ đề xuất** |
| Quản lý công việc | Trello, Jira, GitHub Projects |
| Quản lý mã nguồn | Git + GitHub/GitLab/Bitbucket |
| Tự động hóa (CI/CD) | GitHub Actions, Jenkins |
| Giao tiếp nhóm | Zalo nhóm, Slack, Google Meet |

1. **Ứng dụng quy trình Agile cho phần mềm Quản lý bán hàng (Phần Khách hàng)**

## **Sprint 1: Giỏ hàng và Đặt hàng**

|  |  |  |
| --- | --- | --- |
| **STT** | **User Story** | **Mức độ ưu tiên** |
| 1 | Hiển thị danh sách sản phẩm hiện có | Cao |
| 2 | Thêm sản phẩm vào giỏ hàng | Cao |
| 3 | Sửa/xóa sản phẩm khỏi giỏ hàng | Cao |
| 4 | Chọn hình thức thanh toán trước khi bấm nút 'Đặt hàng' | Cao |
| 5 | Gửi đơn hàng, chuyển sang trạng thái 'chờ xác nhận' | Cao |

## Kế hoạch thực hiện theo từng ngày

|  |  |
| --- | --- |
| Ngày | Công việc |
| Ngày 1 | Thiết kế UI: danh sách sản phẩm và giao diện giỏ hàng |
| Ngày 2 | Code hiển thị sản phẩm từ CSDL (trạng thái 'Còn hàng') |
| Ngày 3–4 | Thêm/Sửa/Xóa sản phẩm khỏi giỏ hàng (lưu local hoặc DB) |
| Ngày 5 | Thêm màn hình chọn hình thức thanh toán (radio button) |
| Ngày 6–7 | Gửi đơn hàng → lưu vào bảng Đơn hàng với trạng thái 'Waiting' |
| Ngày 8–10 | Test, sửa lỗi, demo nội bộ |

1. **Sprint 2: Xác nhận đơn hàng và gửi Email**

|  |  |  |
| --- | --- | --- |
| **STT** | **User Story** | **Mức độ ưu tiên** |
| 1 | Là admin, tôi muốn xác nhận đơn hàng của khách hàng để chuyển sang trạng thái đang xử lý. | Cao |
| 2 | Là hệ thống, tôi muốn tự động gửi email xác nhận đơn hàng sau khi khách hàng đặt hàng thành công. | Cao |
| 3 | Là khách hàng, tôi muốn nhận email xác nhận đơn hàng để yên tâm và có bằng chứng giao dịch. | Trung bình |

* **Mục tiêu Sprint 2:**
* Xây dựng quy trình xác nhận đơn hàng từ phía admin.
* Gửi email xác nhận tự động (nội dung có mã đơn hàng, sản phẩm, tổng tiền, phương thức thanh toán).
* Tích hợp với thư viện gửi email (JavaMail API hoặc SMTP).
* **Kế hoạch thực hiện theo từng ngày**

|  |  |
| --- | --- |
| **Ngày** | **Công việc** |
| Ngày 1 | Thiết kế UI màn hình admin để quản lý đơn hàng chờ xác nhận |
| Ngày 2 | Code chức năng xác nhận đơn hàng và cập nhật trạng thái đơn hàng (→ "Confirming") |
| Ngày 3 | Thiết kế nội dung email xác nhận đơn hàng (mã đơn, sản phẩm, tổng tiền...) |
| Ngày 4–5 | Tích hợp chức năng gửi email sử dụng JavaMail API hoặc SMTP |
| Ngày 6 | Kiểm thử quá trình đặt hàng → xác nhận → gửi email (end-to-end) |
| Ngày 7–8 | Bổ sung thông báo popup khi xác nhận thành công hoặc có lỗi |
| Ngày 9–10 | Sửa lỗi, kiểm thử lại, demo nội bộ cho Product Owner |

1. **Ưu và nhược điểm của mô hình Agile**

|  |  |  |
| --- | --- | --- |
| **Tiêu chí** | **Ưu điểm** | **Nhược điểm** |
| **Linh hoạt (Flexibility)** | Dễ dàng thích ứng với thay đổi yêu cầu trong suốt quá trình phát triển. | Có thể dẫn đến thay đổi quá thường xuyên, ảnh hưởng đến tiến độ nếu không kiểm soát tốt. |
| **Phản hồi khách hàng** | Liên tục lấy ý kiến khách hàng sau mỗi Sprint giúp sản phẩm phù hợp thực tế hơn. | Cần sự tham gia thường xuyên từ khách hàng – đôi khi khách hàng không có đủ thời gian. |
| **Tiến độ và chất lượng** | Phát hiện lỗi sớm, cải thiện dần chất lượng qua từng vòng phát triển. | Dễ bị phân tán nếu nhóm không có kinh nghiệm hoặc không quản lý tốt Product Backlog. |
| **Giao tiếp trong nhóm** | Tăng cường trao đổi nội bộ, chia sẻ kiến thức giữa các thành viên. | Thiếu tài liệu chính thức rõ ràng nếu chỉ tập trung vào giao tiếp miệng. |
| **Đáp ứng thị trường nhanh** | Sản phẩm có thể đưa ra thị trường sớm, thăm dò và cải tiến theo nhu cầu thực tế. | Không phù hợp với các dự án có yêu cầu cố định và quy mô lớn cần kiểm soát chặt chẽ. |
| **Quản lý rủi ro** | Kiểm tra và điều chỉnh thường xuyên giúp giảm rủi ro tổng thể. | Nếu không có PO rõ ràng và cam kết, nhóm có thể đi sai hướng hoặc thiếu định hướng dài hạn. |
| **Tài liệu dự án** | Tập trung vào phần mềm chạy được hơn là tài liệu dư thừa. | Các bên liên quan (ví dụ: nhà đầu tư) có thể cần tài liệu chi tiết để đánh giá và duyệt dự án. |

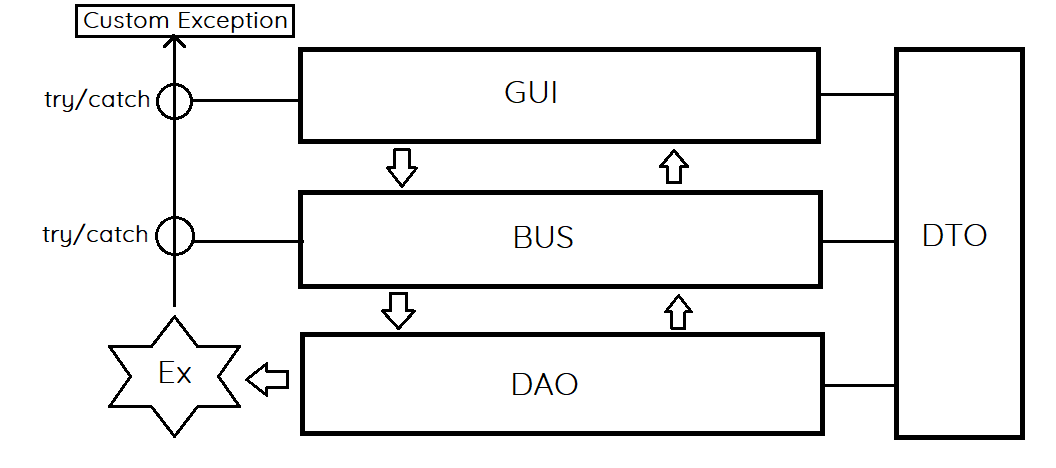
***Chương 4: Kiến trúc phần mềm và mô hình hóa dữ liệu***

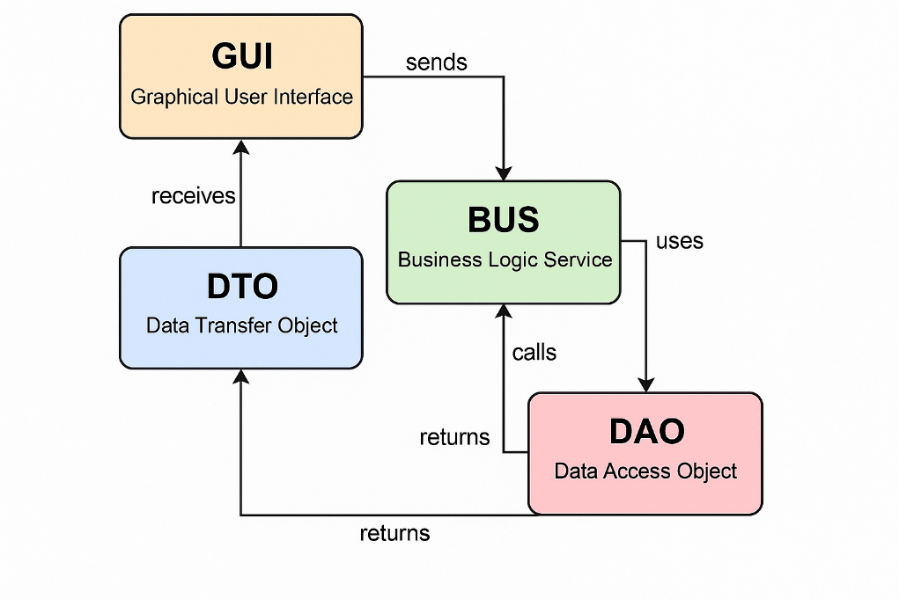
1. **Kiến trúc phần mềm**
2. **Ngôn ngữ lập trình Java**

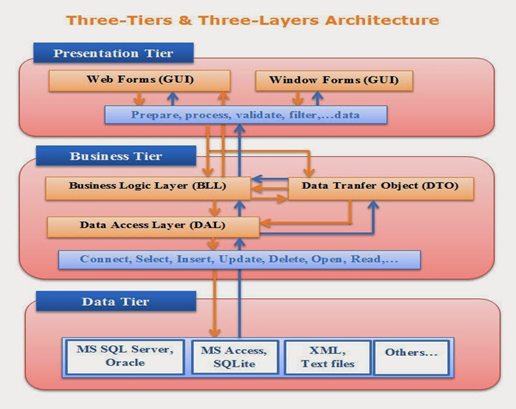
* Java là một ngôn ngữ lập trình bậc cao, hướng đối tượng được phát triển bởi Sun Microsystems (nay thuộc Oracle). Được ra mắt lần đầu vào năm 1995, Java nhanh chóng trở thành một trong những ngôn ngữ lập trình phổ biến nhất trên thế giới nhờ khả năng đa nền tảng, bảo mật cao, dễ sử dụng, và có một hệ sinh thái phong phú.
* **Đặc điểm nổi bật của Java:**
* Hướng đối tượng (OOP - Object-Oriented Programming) Java tuân theo mô hình lập trình hướng đối tượng, giúp tổ chức mã nguồn thành các đối tượng tái sử dụng, dễ bảo trì, và quản lý hiệu quả.
* Độc lập nền tảng – "Viết một lần, chạy mọi nơi" (WORA - Write Once, Run Anywhere) Nhờ vào Máy ảo Java (JVM - Java Virtual Machine), Java có thể chạy trên nhiều hệ điều hành khác nhau mà không cần sửa đổi mã nguồn.
* Quản lý bộ nhớ tự động – "Garbage Collection" Java có hệ thống Thu gom rác (Garbage Collector) tự động giải phóng bộ nhớ không còn sử dụng, giúp giảm thiểu lỗi quản lý bộ nhớ.
* Thư viện phong phú & Framework đa dạng Java sở hữu một hệ sinh thái rộng lớn với hàng nghìn thư viện hỗ trợ phát triển ứng dụng web, di động, AI, Big Data, v.v.
* Bảo mật cao Java cung cấp các tính năng bảo mật mạnh mẽ như kiểm soát truy cập, mã hóa dữ liệu, giúp bảo vệ ứng dụng khỏi các mối nguy hại.
* Hỗ trợ đa luồng (Multithreading) Java cho phép xử lý song song nhiều tác vụ, giúp tăng hiệu suất cho ứng dụng cần xử lý nhiều công việc cùng lúc.

1. **Kiến trúc phân lớp(Layered Architecture)**

* Mô hình **BUS, DAO, DTO, GUI** thường được sử dụng trong lập trình phần mềm doanh nghiệp, đặc biệt khi xây dựng ứng dụng theo kiến trúc phân lớp (Layered Architecture). Đây là một cách tiếp cận tuân theo nguyên lý của lập trình hướng đối tượng (OOP), giúp hệ thống có cấu trúc rõ ràng, dễ bảo trì, mở rộng và tái sử dụng.
* Một hệ thống phần mềm được chia thành nhiều lớp, mỗi lớp đảm nhận một nhiệm vụ cụ thể, tạo nên sự tách biệt và linh hoạt trong thiết kế. Nhờ vào tính đóng gói, mỗi lớp chỉ cung cấp các chức năng cần thiết, giấu đi chi tiết triển khai bên trong, giúp tăng tính bảo mật và ổn định cho ứng dụng.
* Tính kế thừa cho phép các lớp con sử dụng lại các phương thức và thuộc tính từ lớp cha, giảm sự trùng lặp mã, đồng thời hỗ trợ mở rộng chức năng mà không cần thay đổi cấu trúc gốc. Nhờ đó, mô hình này giúp xây dựng hệ thống phần mềm có tính linh hoạt cao, dễ dàng cập nhật và nâng cấp.
* Ngoài ra, tính đa hình đảm bảo rằng cùng một phương thức có thể hoạt động theo những cách khác nhau tùy thuộc vào lớp đối tượng đang sử dụng. Điều này giúp mã nguồn dễ đọc, dễ bảo trì, và cho phép mở rộng ứng dụng một cách linh hoạt mà không cần sửa đổi nhiều đoạn mã cũ.
* Cuối cùng, tính trừu tượng giúp lập trình viên tập trung vào các khái niệm cốt lõi, thay vì chi tiết triển khai. Nó cho phép định nghĩa các phương thức và lớp trừu tượng, từ đó các lớp con có thể triển khai chi tiết phù hợp với từng trường hợp sử dụng thực tế.
* Nhờ các đặc điểm của OOP, mô hình **BUS, DAO, DTO, GUI** không chỉ giúp tổ chức hệ thống một cách khoa học, mà còn đảm bảo ứng dụng dễ mở rộng, dễ bảo trì và có hiệu suất hoạt động cao trong môi trường doanh nghiệp.



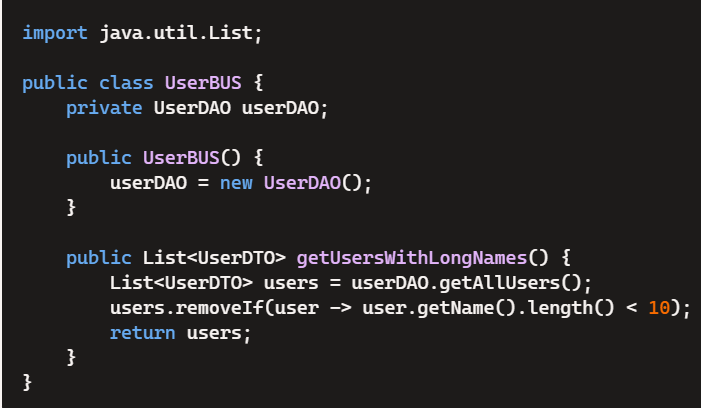




*Hình 3: Mô phỏng cách hoạt động của mô hình phân lớp*

* **BUS (Business Logic Layer) – Lớp xử lý nghiệp vụ**
* Chứa các quy tắc nghiệp vụ, xử lý logic trước khi gửi dữ liệu đến GUI hoặc DAO:
* Đóng vai trò trung gian, đảm bảo ứng dụng hoạt động đúng với yêu cầu của doanh nghiệp.
* Không tương tác trực tiếp với cơ sở dữ liệu, mà lấy dữ liệu từ DAO, xử lý rồi trả về GUI.

Ví dụ:



*Hình 4: Miêu tả lớp BUS hoạt động*

* **DAO (Data Access Object) – Lớp truy xuất dữ liệu**
* Chịu trách nhiệm giao tiếp với cơ sở dữ liệu, thực hiện các thao tác CRUD (Create, Read, Update, Delete)
* Tách biệt hoàn toàn logic xử lý với hệ thống lưu trữ dữ liệu, giúp bảo trì dễ dàng hơn.
* Cung cấp các phương thức lấy dữ liệu từ database và chuyển nó thành DTO để sử dụng.

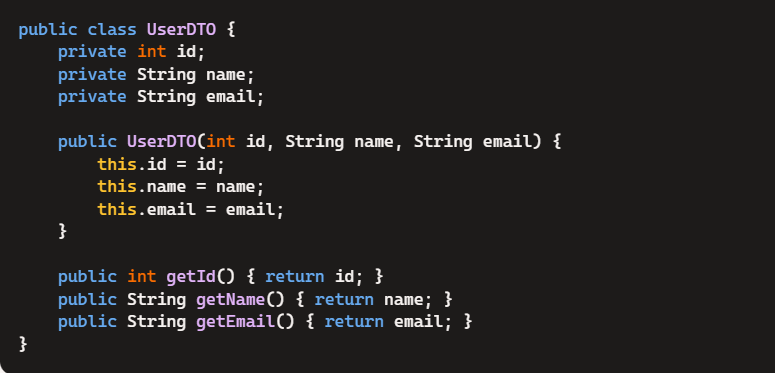
Ví dụ:



*Hình 5: Miêu tả lớp DAO hoạt động*

* **DTO (Data Transfer Object) – Đối tượng truyền dữ liệu:**
* Đóng gói dữ liệu, giúp trao đổi thông tin giữa các lớp mà không cần tương tác trực tiếp với cơ sở dữ liệu.
* Không chứa logic xử lý, chỉ có thuộc tính (fields), constructor, getter/setter.
* Giúp giảm sự phụ thuộc giữa các lớp và làm cho việc truyền dữ liệu dễ dàng hơn.

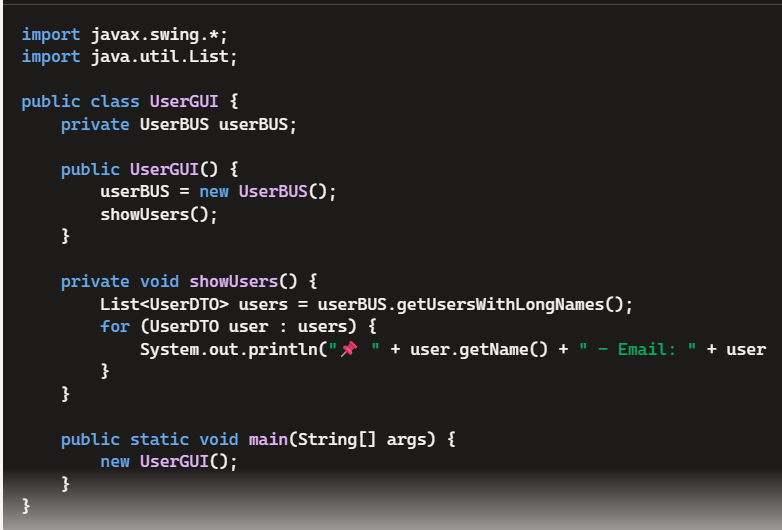
Ví dụ:



*Hình 6: Miêu tả lớp DTO hoạt động*

* **GUI (Graphical User Interface) – Lớp giao diện**
* Hiển thị thông tin và tương tác với người dùng, giúp người dùng có thể nhập, xem, chỉnh sửa dữ liệu.
* Không chứa logic xử lý, chỉ nhận dữ liệu từ BUS để hiển thị.
* Có thể là ứng dụng web (HTML, CSS, JavaScript) hoặc giao diện đồ họa (Java Swing, JavaFX).

Ví dụ:



*Hình 7: Miêu tả lớp GUI hoạt động*