**TÀI LIỆU MÔ TẢ THIẾT KẾ PHẦN MỀM**

***<<QUẢN LÝ BÁN HÀNG CHO CỬA HÀNG QUẦN ÁO>>***

**SDD\_Nhóm 3\_QLBHCCHQA.docx**

**Hà Nội, [tháng 5/2025]**

**Bảng ghi nhận sự thay đổi của tài liệu**

| **Thời gian thay đổi** | **Nội dung thay đổi** | **Lý do** | **Phiên bản bị thay đổi** | **Mô tả sự thay đổi** | **Phiên bản mới** |
| --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

Mục Lục :

[1. GIỚI THIỆU 5](#_Toc199187681)

[1.1. Mục đích tài liệu 5](#_Toc199187682)

[1.2. Phạm vi tài liệu 5](#_Toc199187683)

[1.3. Thuật ngữ và từ viết tắt 5](#_Toc199187684)

[1.4. Tài liệu tham khảo 6](#_Toc199187685)

[1.5. Mô tả tài liệu 6](#_Toc199187686)

[2. KHẢO SÁT HIỆN TRẠNG PHẦN MỀM 6](#_Toc199187687)

[2.1. Khảo sát hiện trạng 6](#_Toc199187688)

[2.2. Yêu cầu hệ thống mới 9](#_Toc199187689)

[3. THIẾT KẾ HỆ THỐNG PHẦN MỀM 11](#_Toc199187690)

[3.1. Biểu đồ usecase 11](#_Toc199187691)

[3.2. Biểu đồ activity 17](#_Toc199187692)

[3.3. Biểu đồ sequence 19](#_Toc199187693)

[3.4. Biểu đồ class 21](#_Toc199187694)

[3.5. Biểu đồ component 24](#_Toc199187695)

[3.6. Biểu đồ deployment 26](#_Toc199187696)

[4. THIẾT KẾ DỮ LIỆU 27](#_Toc199187697)

[4.1. Mô tả dữ liệu 27](#_Toc199187698)

[4.2. Thiết kế ER 28](#_Toc199187699)

[4.3. Thiết kế dữ liệu 29](#_Toc199187700)

[5. THIẾT KẾ GIAO DIỆN 31](#_Toc199187701)

[6. CÁC MỤC NÂNG CAO KHÔNG BẮT BUỘC 32](#_Toc199187702)

[6.1. Đánh giá rủi ro 32](#_Toc199187703)

[6.2. Kế hoạch kiểm thử 33](#_Toc199187704)

[6.3. Lộ trình triển khai 33](#_Toc199187705)

# 1. GIỚI THIỆU

## 1.1. Mục đích tài liệu

Tài liệu này cung cấp mô tả chi tiết về thiết kế của Hệ thống Quản lý Bán hàng cho Cửa hàng Quần áo, làm cơ sở để các nhà phát triển triển khai, kiểm thử và bảo trì hệ thống.

## 1.2. Phạm vi tài liệu

Tài liệu là công cụ giao tiếp cho nhóm phát triển, cung cấp thông số kỹ thuật chi tiết về kiến trúc hệ thống, thiết kế dữ liệu và giao diện người dùng. Nó hỗ trợ triển khai, kiểm thử, triển khai và bảo trì hệ thống.

## 1.3. Thuật ngữ và từ viết tắt

| **Thuật ngữ** | **Định nghĩa** | **Giải thích** |
| --- | --- | --- |
| CNTT | Công nghệ thông tin | Information Technology |
| CNPM | Công nghệ phần mềm | Software Engineering |
| CSDL | Cơ sở dữ liệu | Nơi lưu trữ thông tin và cho phép truy cập |
| PK | Primary Key | Khóa chính |
| UK | Unique Key | Khóa duy nhất |
| SRS | Đặc tả yêu cầu phần mềm | Tài liệu mô tả yêu cầu chức năng và phi chức năng |
| JWT | JSON Web Token | Công nghệ xác thực người dùng |
| API | Giao diện lập trình ứng dụng | Giao diện cho các hệ thống giao tiếp |
| MVC | Model-View-Controller | Mô hình kiến trúc phần mềm tách biệt dữ liệu, giao diện và logic |
| MTTR | Thời gian trung bình để sửa lỗi | Mean Time To Repair |
|  |  |  |

## 1.4. Tài liệu tham khảo

|  |  |  |
| --- | --- | --- |
| Tên tài liệu | Nguồn | Ngày phát hành |
| IEEE 1016-2009 | IEEE | 2009 |
| SRS\_Quản lý bán hàng cho cửa hàng quần áo | Nhóm 3 | 25/04/2025 |
| Tài liệu khảo sát | Nhóm 3 | 15/04/2025 |

## 1.5. Mô tả tài liệu

Tài liệu SDD bao gồm các phần:

* **Phần 1**: Tổng quan về tài liệu, mục đích và vai trò trong quá trình phát triển.
* **Phần 2**: Phân tích hiện trạng, yêu cầu và ràng buộc thiết kế.
* **Phần 3**: Thiết kế hệ thống với các biểu đồ usecase, activity, sequence, class, component, deployment.
* **Phần 4**: Thiết kế dữ liệu với mô tả thực thể, ERD và lược đồ CSDL.
* **Phần 5**: Thiết kế giao diện với wireframe và nguyên tắc UX/UI.
* **Phần 6**: Các mục nâng cao về đánh giá rủi ro, kiểm thử và lộ trình triển khai.

# 2. KHẢO SÁT HIỆN TRẠNG PHẦN MỀM

## 2.1. Khảo sát hiện trạng

**Tổng quan hệ thống hiện tại**:  
Cửa hàng quần áo hiện sử dụng phương pháp quản lý thủ công, kết hợp với một số công cụ không đồng bộ như Microsoft Excel và sổ sách giấy. Hệ thống này phục vụ các hoạt động bán hàng, quản lý kho, khách hàng và tài chính, nhưng thiếu tích hợp và tự động hóa, dẫn đến nhiều bất cập trong vận hành. Dựa trên khảo sát thực tế tại cửa hàng và phỏng vấn các bên liên quan (nhân viên bán hàng, quản lý kho, thu ngân, quản lý cửa hàng), các quy trình và vấn đề chính được xác định như sau:

**Quy trình bán hàng hiện tại**:

1. **Khách hàng chọn sản phẩm**:
   * Tại cửa hàng: Khách hàng xem sản phẩm trực tiếp, nhân viên bán hàng tư vấn và ghi lại thông tin đơn hàng trên giấy hoặc Excel.
   * Trực tuyến: Cửa hàng có một trang web đơn giản (hoặc fanpage trên mạng xã hội) để khách hàng đặt hàng qua tin nhắn. Nhân viên phải nhập thủ công thông tin đơn hàng vào Excel.
2. **Xác nhận đơn hàng**:
   * Nhân viên bán hàng kiểm tra thông tin khách hàng (tên, số điện thoại, địa chỉ giao hàng) và yêu cầu nhân viên kho kiểm tra tồn kho.
   * Quá trình này thường mất 5-10 phút do phải liên lạc qua điện thoại hoặc trao đổi trực tiếp.
3. **Kiểm tra và xuất kho**:
   * Nhân viên kho tìm kiếm thông tin tồn kho trong sổ sách hoặc file Excel, xác nhận số lượng và lập phiếu xuất kho thủ công.
   * Nếu sản phẩm không đủ, nhân viên kho thông báo cho quản lý để liên hệ nhà cung cấp, gây chậm trễ.
4. **Thanh toán và hóa đơn**:
   * Khách hàng thanh toán bằng tiền mặt, chuyển khoản ngân hàng, hoặc ví điện tử (Momo, ZaloPay), nhưng không có hệ thống tích hợp để xác minh thanh toán tự động.
   * Thu ngân tạo hóa đơn giấy hoặc gửi hóa đơn qua tin nhắn, mất thời gian nhập liệu và dễ sai sót.
5. **Giao hàng**:
   * Đơn hàng tại cửa hàng được giao trực tiếp cho khách.
   * Đơn hàng trực tuyến được chuyển qua đơn vị vận chuyển (Giao Hàng Nhanh, Giao Hàng Tiết Kiệm), nhưng không có hệ thống theo dõi trạng thái giao hàng.
6. **Cập nhật trạng thái đơn hàng**:
   * Nhân viên bán hàng cập nhật trạng thái đơn hàng (hoàn tất, hủy, đang giao) vào Excel hoặc sổ sách, nhưng thường không đồng bộ giữa các bộ phận.

**Các vấn đề chính**:

* **Hiệu suất thấp**:
  + Việc ghi chép thủ công và kiểm tra kho mất trung bình 10-15 phút mỗi đơn hàng, đặc biệt khi cửa hàng đông khách.
  + Quy trình kiểm tra tồn kho không thời gian thực, dẫn đến tình trạng bán quá số lượng tồn kho hoặc từ chối đơn hàng không cần thiết.
* **Sai sót dữ liệu**:
  + Nhập liệu thủ công gây ra lỗi như trùng lặp mã sản phẩm, sai thông tin khách hàng, hoặc không khớp số liệu tồn kho.
  + Theo khảo sát, khoảng 15% đơn hàng gặp lỗi nhập liệu, dẫn đến trả hàng hoặc khiếu nại từ khách.
* **Thiếu tích hợp kênh bán hàng**:
  + Kênh bán hàng trực tuyến và tại cửa hàng hoạt động độc lập, không đồng bộ tồn kho hoặc thông tin đơn hàng.
  + Ví dụ, sản phẩm có thể hiển thị "còn hàng" trên website nhưng thực tế đã hết tại kho.
* **Quản lý khách hàng kém**:
  + Không có hệ thống lưu trữ thông tin khách hàng (lịch sử mua sắm, sở thích, điểm tích lũy), khiến cửa hàng không thể triển khai chương trình khách hàng thân thiết.
  + Khách hàng thường phàn nàn về việc không nhận được ưu đãi hoặc thông tin khuyến mãi kịp thời.
* **Báo cáo chậm trễ**:
  + Việc tổng hợp doanh thu, tồn kho, hoặc chi phí phải thực hiện thủ công vào cuối ngày hoặc cuối tuần, mất 2-3 giờ mỗi lần.
  + Báo cáo thường thiếu chính xác do lỗi nhập liệu hoặc không cập nhật kịp thời.
* **Khó mở rộng**:
  + Hệ thống thủ công không đáp ứng được khi cửa hàng mở thêm chi nhánh hoặc tăng quy mô bán hàng trực tuyến.
  + Nhân viên phải làm việc quá tải khi lưu lượng đơn hàng tăng (ví dụ, trong mùa cao điểm như Black Friday).

**So sánh với đối thủ cạnh tranh**:

* Các chuỗi bán lẻ lớn (Zara, H&M, Uniqlo) sử dụng hệ thống quản lý bán hàng tích hợp với các tính năng:
  + **Tồn kho thời gian thực**: Cập nhật số lượng sản phẩm trên tất cả kênh bán hàng.
  + **Tích hợp đa kênh**: Đồng bộ đơn hàng từ website, ứng dụng di động và cửa hàng vật lý.
  + **Quản lý khách hàng**: Hệ thống CRM lưu trữ lịch sử mua sắm, gửi ưu đãi cá nhân hóa qua email/SMS.
  + **Thanh toán linh hoạt**: Hỗ trợ nhiều phương thức thanh toán (thẻ, ví điện tử, Apple Pay).
  + **Báo cáo tự động**: Tạo báo cáo doanh thu, tồn kho, hoặc xu hướng mua sắm chỉ trong vài giây.
* Cửa hàng hiện tại thiếu các tính năng này, dẫn đến trải nghiệm khách hàng kém hơn và hiệu quả vận hành thấp hơn so với đối thủ.

## 2.2. Yêu cầu hệ thống mới

Sau khi tìm hiểu, chúng em rút ra được Hệ thống sẽ gồm nhiều phân hệ các chức năng khác nhau. Các phân hệ chính bao gồm:

* **Quản lý hệ thống**:
  + Cho phép tạo, chỉnh sửa, xóa tài khoản người dùng (khách hàng, nhân viên).
  + Phân quyền rõ ràng theo vai trò (quản lý, nhân viên bán hàng, kho, thu ngân) [REQ-001, REQ-002].
  + Hỗ trợ đăng nhập bằng email/mật khẩu hoặc Google Sign-In [REQ-015].
* **Quản lý sản phẩm**:
  + Lưu trữ thông tin sản phẩm (mã, tên, kích cỡ, màu sắc, giá, số lượng, danh mục) [REQ-004].
  + Cho phép thêm, sửa, xóa sản phẩm với kiểm tra mã duy nhất [REQ-024, REQ-025].
  + Hỗ trợ tìm kiếm và lọc sản phẩm theo nhiều tiêu chí (giá, màu, kích cỡ) [REQ-026].
* **Quản lý đơn hàng**:
  + Tự động hóa quy trình đặt hàng, xác nhận, và cập nhật trạng thái (chờ xử lý, đang giao, hoàn tất, hủy) [REQ-033].
  + Đồng bộ đơn hàng giữa kênh trực tuyến và tại cửa hàng [REQ-034].
  + Gửi thông báo trạng thái đơn hàng qua email/SMS [REQ-035].
* **Quản lý kho**:
  + Cập nhật tồn kho thời gian thực sau mỗi giao dịch [REQ-036].
  + Quản lý nhập/xuất kho với phiếu nhập/xuất tự động [REQ-037].
  + Cảnh báo khi sản phẩm sắp hết hàng (dưới ngưỡng 10% tồn kho) [REQ-038].
* **Quản lý khách hàng**:
  + Lưu trữ thông tin khách hàng (tên, địa chỉ, số điện thoại, lịch sử mua sắm) [REQ-039].
  + Quản lý chương trình khách hàng thân thiết với tích điểm và ưu đãi [REQ-040].
  + Cho phép khách hàng xem lịch sử đơn hàng và trạng thái tích điểm [REQ-041].
* **Quản lý thanh toán**:
  + Tích hợp các phương thức thanh toán (Momo, VNPay, thẻ, tiền mặt) [REQ-007].
  + Tự động xác minh thanh toán và tạo hóa đơn điện tử [REQ-008].
  + Lưu trữ lịch sử giao dịch để đối chiếu [REQ-009].
* **Báo cáo**:
  + Tạo báo cáo doanh thu, tồn kho, và chi phí theo ngày/tuần/tháng [REQ-011].
  + Hỗ trợ xuất báo cáo dưới dạng PDF hoặc Excel [REQ-012].
  + Cung cấp biểu đồ trực quan (doanh thu, sản phẩm bán chạy) [REQ-013].

**Yêu cầu phi chức năng**:

* **Hiệu suất**: Phản hồi tìm kiếm dưới 2 giây, xử lý 100 giao dịch đồng thời [REQ-055, REQ-056].
* **Độ tin cậy**: Độ khả dụng 99.9%, MTTR dưới 4 giờ [REQ-057, REQ-058].
* **Bảo mật**: Mã hóa dữ liệu bằng HTTPS, sao lưu hàng ngày, ngăn chặn SQL Injection [REQ-060, REQ-078].
* **Khả dụng**: Hỗ trợ đa nền tảng (Windows, macOS, iOS, Android) qua trình duyệt Chrome, Firefox, Edge [REQ-069].
* **Dễ sử dụng**: Thời gian đào tạo nhân viên dưới 30 phút, giao diện trực quan [REQ-053].

**2.3. Ràng buộc thiết kế**

* **Công nghệ bắt buộc**: Sử dụng ReactJS, Node.js/Express, MySQL [REQ-063].
* **Hiệu suất**: Thời gian phản hồi tìm kiếm dưới 2 giây, xử lý 100 người dùng đồng thời [REQ-055, REQ-056].
* **Bảo mật**: Mã hóa HTTPS, sao lưu hàng ngày, khóa tài khoản sau 5 lần đăng nhập sai [REQ-060, REQ-017].
* **Độ tin cậy**: Độ khả dụng 99.9%, MTTR dưới 4 giờ [REQ-057, REQ-058].
* **Khả dụng**: Hỗ trợ đa nền tảng (PC, di động) và trình duyệt phổ biến [REQ-069].
* **Dễ sử dụng**: Giao diện đơn giản, thời gian đào tạo nhân viên dưới 30 phút [REQ-053].
* **Tích hợp**: Hỗ trợ API thanh toán (Momo, VNPay) và dịch vụ thông báo (email, SMS) [REQ-007, REQ-035].
* **Ngôn ngữ**: Hệ thống sử dụng tiếng Việt làm ngôn ngữ chính, hỗ trợ tiếng Anh cho khách quốc tế [REQ-070].

# 3. THIẾT KẾ HỆ THỐNG PHẦN MỀM

## 3.1. Biểu đồ usecase

**Biểu đồ Usecase Tổng quát**:

**A diagram of a flowchart

AI-generated content may be incorrect.**

**Mô tả Usecase chi tiết**:

**1.Đăng nhập**

* **Tác nhân**: Khách hàng, Nhân viên bán hàng, Nhân viên kho, Thu ngân, Quản lý, Nhân viên hỗ trợ.
* **Mục đích**: Xác thực người dùng để truy cập hệ thống đúng theo vai trò.
* **Chi tiết**: Hệ thống hỗ trợ đăng nhập qua email/mật khẩu hoặc Google. Tài khoản bị khóa sau 5 lần sai.

**2.Đăng ký**

* **Tác nhân**: Khách hàng.
* **Mục đích**: Tạo tài khoản để mua sắm trực tuyến.
* **Chi tiết**: Khách hàng nhập email, mật khẩu, số điện thoại… và nhận email xác nhận để kích hoạt tài khoản

**3.Tìm kiếm sản phẩm**

* **Tác nhân**: Khách hàng.
* **Mục đích**: Hỗ trợ khách hàng tra cứu nhanh sản phẩm theo các tiêu chí như tên, giá, kích cỡ, màu sắc.

**4.Quản lý sản phẩm**

* **Tác nhân**: Nhân viên bán hàng, Quản lý.
* **Mục đích**: Thêm, sửa, xóa sản phẩm và cập nhật tồn kho.
* **Chi tiết**: Có xác minh mã sản phẩm duy nhất và log lại hành động chỉnh sửa.

**5.Quản lý đơn hàng**

* **Tác nhân**: Khách hàng, Nhân viên bán hàng, Nhân viên kho.
* **Mục đích**: Tạo và xử lý đơn hàng từ khi đặt đến giao hàng.
* **Chi tiết**: Hệ thống cập nhật trạng thái đơn hàng và tồn kho tự động.

**6.Thanh toán**

* **Tác nhân**: Khách hàng, Thu ngân, Hệ thống thanh toán.
* **Mục đích**: Xử lý thanh toán đơn hàng bằng nhiều hình thức: tiền mặt, ví điện tử (Momo, VNPay…).
* **Chi tiết**: Sau khi thanh toán, tạo hóa đơn điện tử và cập nhật trạng thái đơn hàng.

**7.Gửi thông báo**

* **Tác nhân**: Hệ thống.
* **Mục đích**: Gửi tự động thông báo qua Email/SMS cho khách hàng về đơn hàng, bảo hành, khuyến mãi.

**8.Nhận thông báo**

* **Tác nhân**: Khách hàng.
* **Mục đích**: Nhận các thông báo từ hệ thống liên quan đến đơn hàng và khuyến mãi.

**9.Quản lý kho**

* **Tác nhân**: Nhân viên kho, Quản lý, Nhà cung cấp.
* **Mục đích**: Kiểm tra, nhập/xuất kho, cảnh báo tồn kho thấp.
* **Chi tiết**: Tồn kho cập nhật theo thời gian thực và có cảnh báo nếu dưới mức quy định.

**10.Quản lý khách hàng**

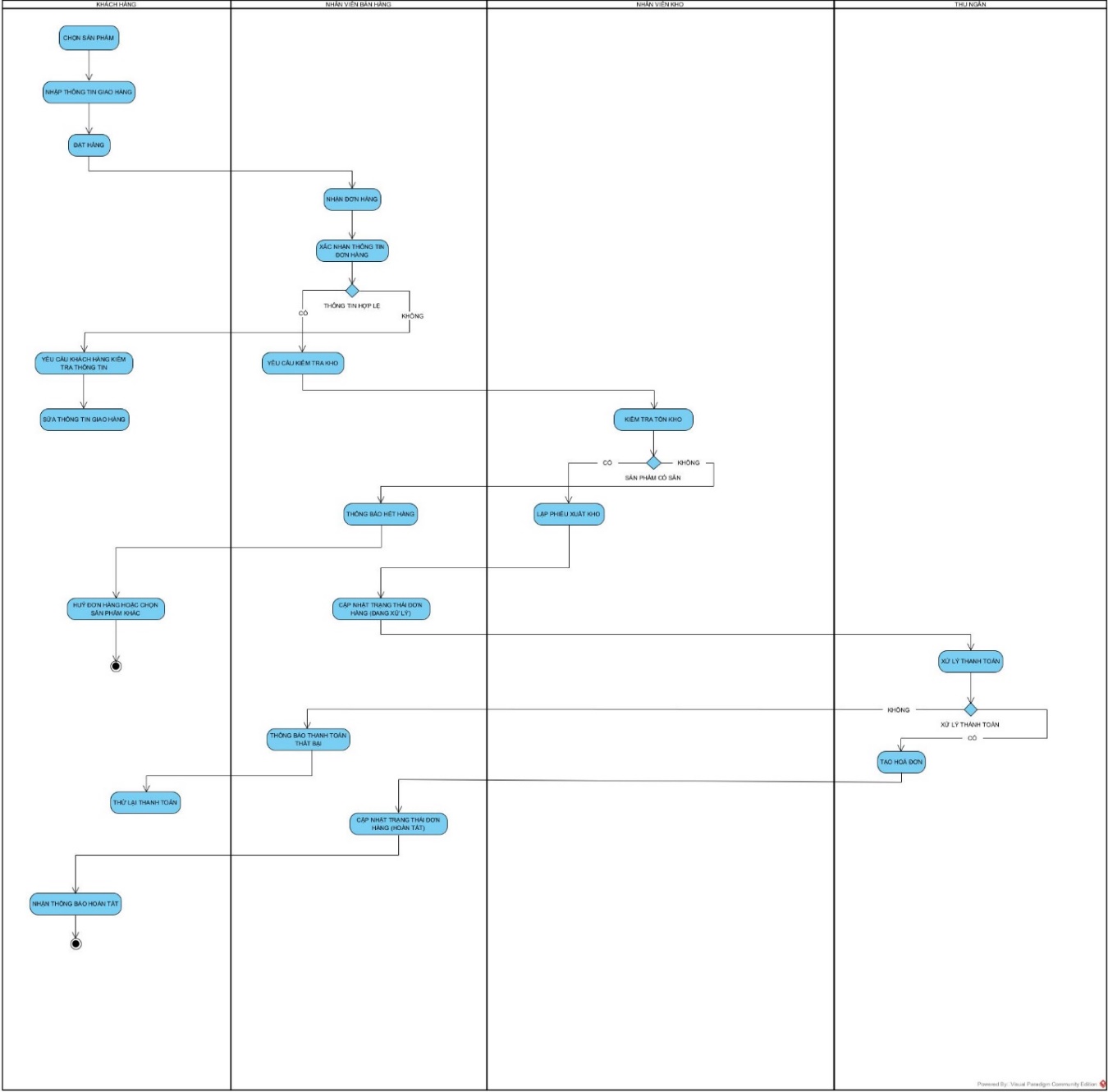
* **Tác nhân**: Quản lý.
* **Mục đích**: Theo dõi lịch sử mua hàng, điểm tích lũy và phân loại khách hàng.
* **Chi tiết**: Quản lý thông tin và hành vi tiêu dùng để phục vụ chương trình khách hàng thân thiết.

**11.Báo cáo**

* **Tác nhân**: Quản lý.
* **Mục đích**: Tạo báo cáo doanh thu, tồn kho, hiệu quả bán hàng.
* **Chi tiết**: Hệ thống hỗ trợ xuất ra PDF hoặc Excel với biểu đồ trực quan.

## 3.2. Biểu đồ activity

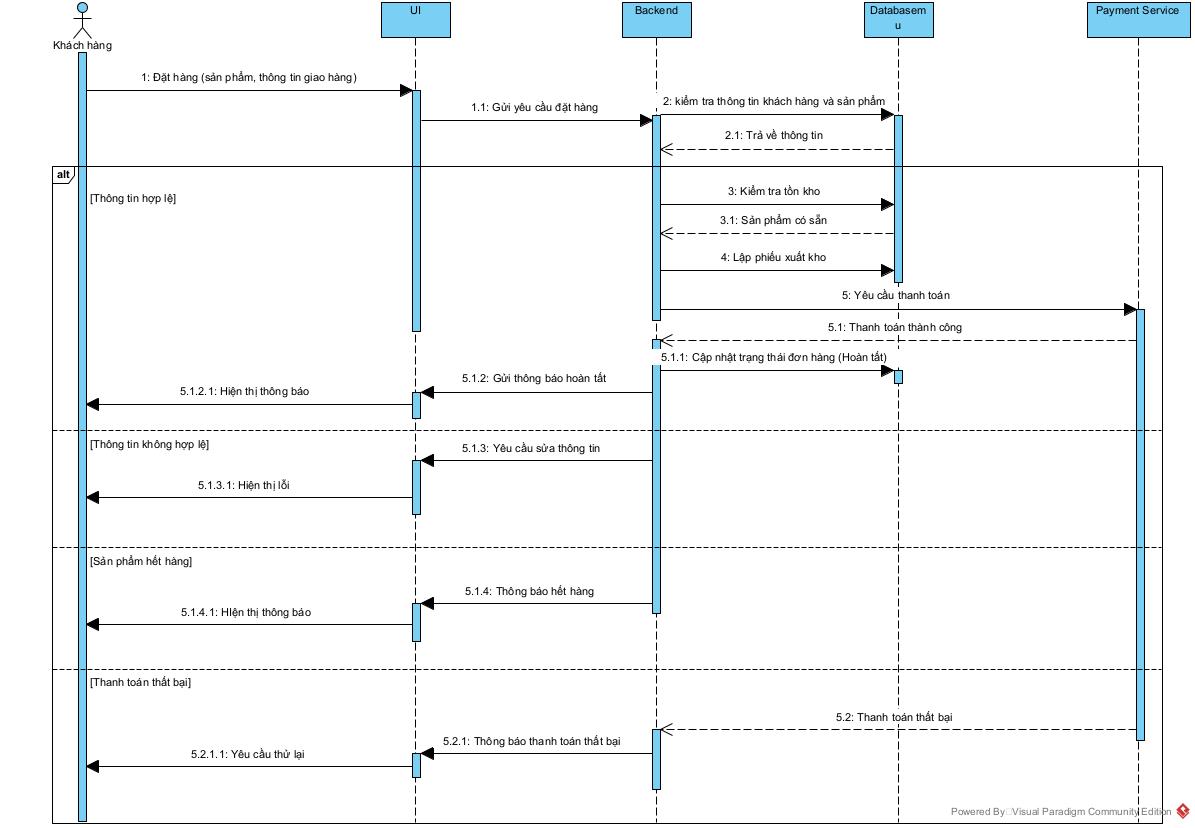
**Biểu đồ Activity cho Quản lý Đơn hàng:**

****

* **Mô tả:** Mô tả quy trình xử lý đơn hàng từ khi khách hàng đặt hàng đến khi hoàn tất hoặc hủy, bao gồm các bước chọn sản phẩm, xác nhận thông tin, kiểm tra kho, thanh toán, và cập nhật trạng thái.
* **Luồng chính:** 
  1. Khách hàng:
     + Chọn sản phẩm.
     + Nhập thông tin giao hàng (tên, địa chỉ, số điện thoại).
     + Đặt hàng qua UI (trực tuyến hoặc tại cửa hàng).
  2. Nhân viên bán hàng:
     + Nhận đơn hàng.
     + Xác nhận thông tin đơn hàng.
     + Kiểm tra thông tin khách hàng hợp lệ:
       - Nếu hợp lệ, chuyển sang bước kiểm tra kho.
       - Nếu không hợp lệ, yêu cầu khách hàng chỉnh sửa thông tin.
  3. Kho hàng:
     + Kiểm tra tồn kho:
       - Nếu đủ hàng, lập phiếu xuất kho.
       - Nếu không đủ hàng, thông báo cho quản lý để xử lý nhập thêm.
  4. Thu ngân:
     + Xử lý thanh toán (tiền mặt, Momo, VNPay).
     + Tạo hóa đơn điện tử.
     + Cập nhật trạng thái đơn hàng (hoàn tất) sau khi thanh toán thành công .
  5. Hệ thống gửi thông báo trạng thái đơn hàng qua email/SMS cho khách hàng.
* **Luồng phụ:** 
  1. Hủy đơn hàng: Nếu khách hàng hủy trước khi thanh toán, hệ thống cập nhật trạng thái "hủy" và gửi thông báo hủy cho khách hàng.
  2. Thanh toán thất bại: Nếu thanh toán không thành công (C6), hệ thống yêu cầu Thu ngân hoặc khách hàng thử lại với phương thức khác.
  3. Thông báo quản lý: Nếu tồn kho không đủ, quản lý quyết định nhập thêm hàng hoặc thông báo khách hàng về tình trạng hết hàng.
* **Yêu cầu liên quan:** 
  1. [REQ-033]: Hỗ trợ đặt hàng trực tuyến và tại cửa hàng.
  2. [REQ-034]: Đồng bộ trạng thái đơn hàng sau thanh toán.
  3. [REQ-035]: Gửi thông báo trạng thái đơn hàng qua email/SMS.
  4. [REQ-036]: Kiểm tra tồn kho thời gian thực trước khi xác nhận đơn hàng.
  5. [REQ-007]: Hỗ trợ nhiều phương thức thanh toán (Momo, VNPay, tiền mặt).

## 3.3. Biểu đồ sequence

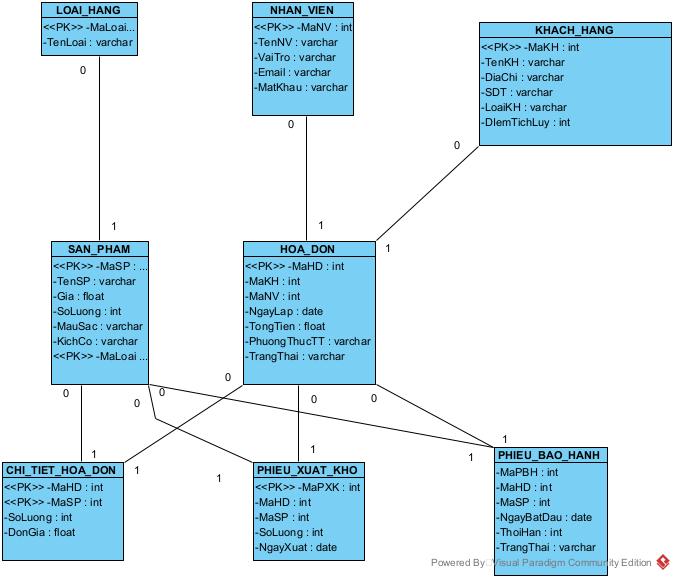
**Biểu đồ Sequence cho Xử lý Đơn hàng:**

****

* **Tác nhân:** Khách hàng, UI (Giao diện người dùng), Backend, Database, Payment Service, Email/SMS Service.
* **Luồng chính:** 
  1. Khách hàng chọn sản phẩm, nhập thông tin giao hàng (tên, địa chỉ, số điện thoại) và gửi yêu cầu đặt hàng qua UI (trực tuyến hoặc tại cửa hàng).
  2. UI gửi yêu cầu đặt hàng đến Backend qua API.
     + Backend kiểm tra thông tin khách hàng trong Database.
     + Nếu thông tin không hợp lệ, trả về lỗi cho UI và yêu cầu khách hàng chỉnh sửa.
     + Nếu hợp lệ, tiếp tục bước kiểm tra tồn kho.
  3. Backend kiểm tra tồn kho trong Database.
     + Nếu đủ hàng, Backend lập phiếu xuất kho.
     + Nếu không đủ hàng, trả về thông báo lỗi cho UI (thông báo khách hàng hoặc quản lý để xử lý nhập thêm hàng).
  4. Backend gửi yêu cầu thanh toán đến Payment Service (Momo, VNPay, tiền mặt).
     + Payment Service xử lý thanh toán và trả về kết quả cho Backend.
     + Nếu thanh toán thành công, Backend cập nhật trạng thái đơn hàng trong Database thành "hoàn tất", gửi thông báo hoàn tất đến UI, và gửi thông báo trạng thái đơn hàng đến Email/SMS Service để gửi cho khách hàng qua email hoặc SMS.
     + Nếu thanh toán thất bại, Backend yêu cầu UI hiển thị thông báo và yêu cầu khách hàng thử lại hoặc chọn phương thức thanh toán khác.
  5. UI hiển thị thông báo hoàn tất hoặc yêu cầu sửa thông tin thanh toán cho khách hàng.
* **Luồng phụ:** 
  1. Thông tin khách hàng không hợp lệ: UI hiển thị thông báo yêu cầu khách hàng chỉnh sửa thông tin.
  2. Tồn kho không đủ: Backend thông báo cho UI, UI hiển thị thông báo cho khách hàng (hoặc thông báo quản lý để nhập thêm hàng).
  3. Thanh toán thất bại: UI yêu cầu khách hàng thử lại tối đa 3 lần hoặc chọn phương thức thanh toán khác.
* **Yêu cầu liên quan:** 
  1. [REQ-007]: Hỗ trợ nhiều phương thức thanh toán (Momo, VNPay, tiền mặt).
  2. [REQ-034]: Cập nhật trạng thái đơn hàng sau thanh toán.
  3. [REQ-035]: Gửi thông báo trạng thái đơn hàng qua email/SMS.
  4. [REQ-036]: Kiểm tra tồn kho thời gian thực trước khi xác nhận đơn hàng.

## 3.4. Biểu đồ class

**Biểu đồ Class:**

****

* **Lớp:** 
  + **LOAI\_HANG:** 
    - Thuộc tính: MaLoai (PK): varchar, TenLoai: varchar.
    - Phương thức: addCategory(), updateCategory(), deleteCategory().
    - Vai trò: Quản lý danh mục sản phẩm.
    - Gói: Quản lý sản phẩm.
  + **SAN\_PHAM:** 
    - Thuộc tính: MaSP (PK): varchar, TenSP: varchar, Gia: float, SoLuong: int, MauSac: varchar, KichCo: varchar, MaLoai (FK): varchar.
    - Phương thức: addProduct(), updateProduct(), deleteProduct(), checkStock().
    - Vai trò: Quản lý chi tiết sản phẩm.
    - Gói: Quản lý sản phẩm.
  + **KHACH\_HANG:** 
    - Thuộc tính: MaKH (PK): int, TenKH: varchar, DiaChi: varchar, SDT: varchar, LoaiKH: varchar, DiemTichLuy: int.
    - Phương thức: addCustomer(), updateCustomer(), updateLoyaltyPoints().
    - Vai trò: Quản lý thông tin khách hàng và chương trình khách hàng thân thiết.
    - Gói: Quản lý khách hàng.
  + **NHAN\_VIEN:** 
    - Thuộc tính: MaNV (PK): int, TenNV: varchar, VaiTro: varchar, Email: varchar, MatKhau: varchar.
    - Phương thức: addEmployee(), updateEmployee(), deleteEmployee().
    - Vai trò: Quản lý thông tin nhân viên.
    - Gói: Quản lý hệ thống.
  + **HOA\_DON:** 
    - Thuộc tính: MaHD (PK): int, MaKH (FK): int, MaNV (FK): int, NgayLap: date, TongTien: float, PhuongThucTT: varchar, TrangThai: varchar.
    - Phương thức: createInvoice(), getInvoiceDetails().
    - Vai trò: Quản lý hóa đơn cho đơn hàng.
    - Gói: Quản lý thanh toán.
  + **CHITIET\_HOA\_DON:** 
    - Thuộc tính: MaHD (FK): int, MaSP (FK): int, SoLuong: int, DonGia: float.
    - Phương thức: addItemToInvoice(), updateItem().
    - Vai trò: Liên kết sản phẩm với hóa đơn.
    - Gói: Quản lý thanh toán.
  + **PHIEU\_XUAT\_KHO:** 
    - Thuộc tính: MaPXK (PK): int, MaHD (FK): int, MaSP (FK): int, SoLuong: int, NgayXuat: date.
    - Phương thức: createExportSlip(), updateExportSlip().
    - Vai trò: Quản lý phiếu xuất kho.
    - Gói: Quản lý kho.
  + **PHIEU\_BAO\_HANH:** 
    - Thuộc tính: MaPBH (PK): int, MaHD (FK): int, MaSP (FK): int, NgayBatDau: date, ThoiHan: int, TrangThai: varchar.
    - Phương thức: createWarranty(), updateWarranty().
    - Vai trò: Quản lý phiếu bảo hành.
    - Gói: Quản lý bảo hành.
* **Mối quan hệ:** 
  + LOAI\_HANG ↔ SAN\_PHAM: Một-đến-Nhiều (một danh mục chứa nhiều sản phẩm, MaLoai là khóa ngoại).
  + KHACH\_HANG ↔ HOA\_DON: Một-đến-Nhiều (một khách hàng có nhiều hóa đơn, MaKH là khóa ngoại).
  + NHAN\_VIEN ↔ HOA\_DON: Một-đến-Nhiều (một nhân viên lập nhiều hóa đơn, MaNV là khóa ngoại).
  + HOA\_DON ↔ CHITIET\_HOA\_DON: Một-đến-Nhiều (một hóa đơn chứa nhiều chi tiết, MaHD là khóa ngoại).
  + SAN\_PHAM ↔ CHITIET\_HOA\_DON: Nhiều-đến-Nhiều (sản phẩm liên kết với hóa đơn qua bảng trung gian, MaSP là khóa ngoại).
  + HOA\_DON ↔ PHIEU\_XUAT\_KHO: Một-đến-Nhiều (một hóa đơn liên kết với nhiều phiếu xuất kho, MaHD là khóa ngoại).
  + SAN\_PHAM ↔ PHIEU\_XUAT\_KHO: Nhiều-đến-Nhiều (sản phẩm liên kết với phiếu xuất kho qua bảng trung gian, MaSP là khóa ngoại).
  + HOA\_DON ↔ PHIEU\_BAO\_HANH: Một-đến-Nhiều (một hóa đơn có thể tạo nhiều phiếu bảo hành, MaHD là khóa ngoại).
  + SAN\_PHAM ↔ PHIEU\_BAO\_HANH: Nhiều-đến-Nhiều (sản phẩm liên kết với phiếu bảo hành qua bảng trung gian, MaSP là khóa ngoại).

## 3.5. Biểu đồ component

**Biểu đồ Component:**

**A screenshot of a computer

AI-generated content may be incorrect.**

**Thành phần:**

* + **Client (Thiết bị người dùng):**
  + Gồm có: Browser và Mobile App.
  + Người dùng tương tác với hệ thống thông qua giao diện:

+ UI Web trên trình duyệt.

+ UI Mobile trên thiết bị di động.

* + **FrontEnd:**
  + Cung cấp giao diện người dùng (ReactJS hoặc framework tương đương).
  + Tương tác với người dùng và gửi yêu cầu tới hệ thống backend thông qua API Gateway.
  + **API Gateway:**
  + Là điểm vào duy nhất của hệ thống backend.
  + Chịu trách nhiệm định tuyến các yêu cầu đến các dịch vụ nghiệp vụ tương ứng.
  + **BackEnd Services (Microservices):**
  + Các dịch vụ nghiệp vụ độc lập xử lý logic và dữ liệu:

+ Order Service: Xử lý đơn hàng.

+ Inventory Service: Quản lý tồn kho.

+ Payment Service: Xử lý thanh toán.

+ Notification Service: Gửi thông báo cho người dùng.

+ Report Service: Tạo báo cáo, thống kê.

* + **Cơ sở dữ liệu (Database):**
  + Hệ quản trị cơ sở dữ liệu sử dụng: MySQL.
  + Dữ liệu được lưu trữ tập trung: sản phẩm, đơn hàng, người dùng, tồn kho,…
  + **Dịch vụ bên ngoài (External Services):**
  + External Payment API: Tích hợp các cổng thanh toán như Momo, VNPay để xử lý giao dịch.
  + Email/SMS Service: Gửi thông báo qua email hoặc tin nhắn SMS, có thể tích hợp với SendGrid,…

**Luồng Tương Tác:**

* + Khách hàng hoặc Quản trị viên (Admin) sử dụng UI Web hoặc UI Mobile để gửi yêu cầu (qua giao diện frontend).
  + Yêu cầu được gửi đến API Gateway.
  + API Gateway định tuyến yêu cầu đến dịch vụ backend thích hợp:
  + Ví dụ: Đặt hàng → Order Service; thanh toán → Payment Service.
  + Các dịch vụ backend:
  + Tương tác với Database để lấy hoặc cập nhật dữ liệu.
  + Nếu cần, gọi đến External Services để xử lý thanh toán hoặc gửi thông báo.
  + Kết quả được trả lại theo chiều ngược lại qua API Gateway → FrontEnd → Giao diện người dùng.

## 3.6. Biểu đồ deployment

**Biểu đồ Deployment:**

**A diagram of a computer flowchart

AI-generated content may be incorrect.**

* **Nút:** 
  + Client Device: Chạy trình duyệt web (Chrome, Firefox, Edge) hoặc ứng dụng di động (iOS, Android) với ReactJS.
  + Web Server: Máy chủ chạy Nginx trên hệ điều hành Linux (Ubuntu 20.04), lưu trữ giao diện người dùng.
  + Application Server: Máy chủ chạy Node.js/Express trên Linux (Ubuntu 20.04), xử lý logic nghiệp vụ.
  + Database Server: Máy chủ chạy MySQL trên Linux (Ubuntu 20.04), lưu trữ dữ liệu.
  + External Services:
    - Payment Gateway: Dịch vụ bên ngoài (Momo, VNPay).
    - Email/SMS Service: Dịch vụ gửi thông báo (SendGrid, Twilio).
* **Kết nối:** 
  + Client Device → Web Server: Sử dụng giao thức HTTPS.
  + Web Server → Application Server: Gọi API qua HTTP.
  + Application Server → Database Server: Truy vấn SQL qua JDBC.
  + Application Server → External Services: Tích hợp qua API (HTTPS).
* **Yêu cầu liên quan:** 
  + [REQ-060]: Mã hóa dữ liệu bằng HTTPS.
  + [REQ-063]: Sử dụng ReactJS, Node.js/Express, MySQL.
  + [REQ-069]: Hỗ trợ đa nền tảng (PC, di động) và trình duyệt phổ biến.

# 4. THIẾT KẾ DỮ LIỆU

## 4.1. Mô tả dữ liệu

Dựa trên biểu đồ class và luồng usecase, các thực thể bao gồm:

* **KHACH\_HANG**: Lưu thông tin khách hàng (ID, tên, địa chỉ, điện thoại, loại khách hàng, điểm tích lũy).
* **NHAN\_VIEN**: Lưu thông tin nhân viên (ID, tên, vai trò, email, mật khẩu).
* **HOA\_DON**: Lưu chi tiết hóa đơn (ID, ngày tạo, tổng tiền, phương thức thanh toán, trạng thái, ID khách hàng, ID nhân viên).
* **LOAI\_HANG**: Quản lý danh mục sản phẩm (ID, tên).
* **SAN\_PHAM**: Lưu chi tiết sản phẩm (ID, tên, giá, số lượng, màu sắc, kích cỡ, ID danh mục).
* **PHIEU\_BAO\_HANH**: Theo dõi bảo hành (ID, ngày bắt đầu, thời hạn, trạng thái, ID hóa đơn, ID sản phẩm).
* **PHIEU\_XUAT\_KHO**: Theo dõi xuất kho (ID, ngày xuất, số lượng, ID hóa đơn, ID sản phẩm).
* **CHITIET\_HOA\_DON**: Liên kết hóa đơn với sản phẩm (ID sản phẩm, ID hóa đơn, số lượng, đơn giá).

## 4.2. Thiết kế ER

* **Thực thể và Thuộc tính**:
  + **KHACH\_HANG** (MaKH: int PK, TenKH: varchar, DiaChi: varchar, SDT: varchar, LoaiKH: varchar, DiemTichLuy: int)
  + **NHAN\_VIEN** (MaNV: int PK, TenNV: varchar, VaiTro: varchar, Email: varchar, MatKhau: varchar)
  + **HOA\_DON** (MaHD: int PK, MaKH: int FK, MaNV: int FK, NgayLap: date, TongTien: float, PhuongThucTT: varchar, TrangThai: varchar)
  + **LOAI\_HANG** (MaLoai: varchar PK, TenLoai: varchar)
  + **SAN\_PHAM** (MaSP: varchar PK, TenSP: varchar, Gia: float, SoLuong: int, MauSac: varchar, KichCo: varchar, MaLoai: varchar FK)
  + **PHIEU\_BAO\_HANH** (MaPBH: int PK, MaHD: int FK, MaSP: varchar FK, NgayBatDau: date, ThoiHan: int, TrangThai: varchar)
  + **PHIEU\_XUAT\_KHO** (MaPXK: int PK, MaHD: int FK, MaSP: varchar FK, SoLuong: int, NgayXuat: date)
  + **CHITIET\_HOA\_DON** (MaHD: int FK, MaSP: varchar FK, SoLuong: int, DonGia: float)
* **Mối quan hệ**:
  + **KHACH\_HANG → HOA\_DON** (1:N): Một khách hàng có nhiều hóa đơn.
  + **NHAN\_VIEN → HOA\_DON** (1:N): Một nhân viên lập nhiều hóa đơn.
  + **HOA\_DON → CHITIET\_HOA\_DON** (1:N): Một hóa đơn chứa nhiều chi tiết.
  + **SAN\_PHAM ↔ CHITIET\_HOA\_DON** (N:M): Sản phẩm liên kết với hóa đơn.
  + **LOAI\_HANG → SAN\_PHAM** (1:N): Một danh mục chứa nhiều sản phẩm.
  + **HOA\_DON → PHIEU\_XUAT\_KHO** (1:N): Một hóa đơn liên kết với nhiều phiếu xuất kho.
  + **SAN\_PHAM ↔ PHIEU\_XUAT\_KHO** (N:M): Sản phẩm liên kết với phiếu xuất kho.
  + **HOA\_DON → PHIEU\_BAO\_HANH** (1:N): Một hóa đơn có thể tạo nhiều phiếu bảo hành.
  + **SAN\_PHAM ↔ PHIEU\_BAO\_HANH** (N:M): Sản phẩm liên kết với phiếu bảo hành.

## 4.3. Thiết kế dữ liệu

**Quy trình chuyển ER sang Lược đồ Quan hệ**:

1. **Ánh xạ thực thể sang bảng**: Mỗi thực thể (KHACH\_HANG, SAN\_PHAM, v.v.) thành một bảng với các thuộc tính làm cột.
2. **Gán khóa chính**: Sử dụng ID thực thể (MaKH, MaSP, v.v.) làm khóa chính.
3. **Ánh xạ mối quan hệ**:
   * Quan hệ 1:N: Thêm khóa ngoại vào bảng "nhiều" (MaKH, MaNV trong HOA\_DON).
   * Quan hệ N:M: Tạo bảng trung gian (CHITIET\_HOA\_DON, PHIEU\_XUAT\_KHO, PHIEU\_BAO\_HANH).
4. **Chuẩn hóa**: Đảm bảo bảng ở dạng chuẩn 3NF để tránh dư thừa và bất thường.
5. **Xác định kiểu dữ liệu**: Gán kiểu phù hợp (varchar cho tên, float cho giá, int cho số lượng).
6. **Thêm ràng buộc**: Áp dụng khóa chính, khóa ngoại và ràng buộc duy nhất.

**Lược đồ Quan hệ**:

* **LOAI\_HANG** (MaLoai: VARCHAR(10) PK, TenLoai: VARCHAR(50))
* **KHACH\_HANG** (MaKH: INT PK, TenKH: VARCHAR(50), DiaChi: VARCHAR(100), SDT: VARCHAR(15), LoaiKH: VARCHAR(20), DiemTichLuy: INT)
* **SAN\_PHAM** (MaSP: VARCHAR(10) PK, TenSP: VARCHAR(50), Gia: FLOAT, SoLuong: INT, MauSac: VARCHAR(20), KichCo: VARCHAR(10), MaLoai: VARCHAR(10) FK)
* **HOA\_DON** (MaHD: INT PK, MaKH: INT FK, MaNV: INT FK, NgayLap: DATE, TongTien: FLOAT, PhuongThucTT: VARCHAR(20), TrangThai: VARCHAR(20))
* **CHITIET\_HOA\_DON** (MaHD: INT FK, MaSP: VARCHAR(10) FK, SoLuong: INT, DonGia: FLOAT)
* **PHIEU\_BAO\_HANH** (MaPBH: INT PK, MaHD: INT FK, MaSP: VARCHAR(10) FK, NgayBatDau: DATE, ThoiHan: INT, TrangThai: VARCHAR(20))
* **PHIEU\_XUAT\_KHO** (MaPXK: INT PK, MaHD: INT FK, MaSP: VARCHAR(10) FK, SoLuong: INT, NgayXuat: DATE)
* **NHAN\_VIEN** (MaNV: INT PK, TenNV: VARCHAR(50), VaiTro: VARCHAR(20), Email: VARCHAR(50), MatKhau: VARCHAR(50))

**Mối quan hệ giữa các bảng**:

* Khóa ngoại liên kết các bảng (MaKH, MaNV trong HOA\_DON tham chiếu KHACH\_HANG.MaKH, NHAN\_VIEN.MaNV).
* Bảng trung gian (CHITIET\_HOA\_DON, PHIEU\_XUAT\_KHO, PHIEU\_BAO\_HANH) quản lý quan hệ nhiều-đến-nhiều.

**Biểu đồ ERD :**

A diagram of a computer

AI-generated content may be incorrect.

# 5. THIẾT KẾ GIAO DIỆN

**Mẫu Wireframe:**

* Màn hình Đăng nhập: Biểu mẫu với trường email/mật khẩu, nút Google Sign-In, liên kết đến đăng ký/quên mật khẩu.
* Màn hình Quản lý Sản phẩm: Bảng danh sách sản phẩm với bộ lọc (kích cỡ, màu, giá) và nút thêm/sửa/xóa.
* Màn hình Quản lý Đơn hàng: Bảng đơn hàng với bộ lọc trạng thái và tùy chọn cập nhật.
* Màn hình Theo dõi Trạng thái Giao hàng: Hiển thị trạng thái đơn hàng (Chờ xử lý, Đang giao, Hoàn tất, Hủy) với biểu tượng và màu sắc riêng biệt cho từng trạng thái, kèm nút "Xem chi tiết".
* Giao diện Khách hàng: Tìm kiếm sản phẩm, bộ lọc, giỏ hàng và theo dõi đơn hàng.
* Màn hình Báo cáo: Biểu đồ và bảng cho doanh thu, tồn kho, chi phí với tùy chọn xuất dữ liệu.

**Nguyên tắc UX/UI:**

* Đơn giản: Bố cục rõ ràng, điều hướng dễ hiểu [REQ-053].
* Nhất quán: Sử dụng font Times New Roman, cỡ 12, bảng màu trung tính (trắng, xám, xanh) để dễ tiếp cận.
* Đáp ứng: Hỗ trợ đa thiết bị (PC, di động) [REQ-069].
* Phản hồi: Hiển thị thông báo lỗi (màu đỏ) và xác nhận thành công (màu xanh lá) rõ ràng [REQ-016, REQ-025].
* Trực quan: Hiển thị trạng thái đơn hàng bằng biểu tượng hoặc màu sắc khác nhau (Chờ xử lý: vàng, Đang giao: xanh dương, Hoàn tất: xanh lá, Hủy: đỏ) [REQ-034].
* Đa ngôn ngữ: Hỗ trợ tiếng Việt và tiếng Anh, với nút chuyển đổi ngôn ngữ ở góc trên bên phải [REQ-070].

Hướng dẫn Màu sắc và Font:

* Font: Times New Roman, cỡ 12 cho văn bản; cỡ 14 cho tiêu đề.
* Màu sắc: Chính (#007BFF), Phụ (#6C757D), Nền (#FFFFFF), Lỗi (#DC3545), Thành công (#28A745), Chờ xử lý (#FFC107), Đang giao (#17A2B8), Hủy (#DC3545).
* Khoảng cách: Lề 16px, đệm 8px cho các phần tử.

# 6. CÁC MỤC NÂNG CAO KHÔNG BẮT BUỘC

## 6.1. Đánh giá rủi ro

* **Rủi ro kỹ thuật**:
  + **Tích hợp thanh toán thất bại**: Lỗi API Momo/VNPay [30% xác suất, tác động cao].
  + **Mất dữ liệu**: Do lỗi sao lưu hoặc tấn công mạng [20% xác suất, tác động cao].
  + **Hiệu suất thấp**: Phản hồi chậm khi lưu lượng truy cập cao [25% xác suất, tác động trung bình].
* **Rủi ro triển khai**:
  + **Lỗi nhân viên**: Nhập sai dữ liệu [40% xác suất, tác động trung bình].
  + **Lỗi thông báo**: Gián đoạn dịch vụ email/SMS [35% xác suất, tác động trung bình].
* **Kế hoạch giảm thiểu**:
  + Kiểm thử tích hợp kỹ lưỡng cho API thanh toán [REQ-076].
  + Sao lưu hàng ngày và mã hóa HTTPS [REQ-060].
  + Tối ưu truy vấn CSDL và dùng CDN để cải thiện hiệu suất [REQ-055].
  + Đào tạo nhân viên (<30 phút) và xác thực dữ liệu đầu vào [REQ-053, REQ-061].
  + Sử dụng nhiều nhà cung cấp dịch vụ thông báo dự phòng [REQ-066].

## 6.2. Kế hoạch kiểm thử

* **Chiến lược kiểm thử**:
  + **Kiểm thử đơn vị**: Kiểm tra hàm riêng lẻ (kiểm tra mã sản phẩm) bằng Jest [REQ-076].
  + **Kiểm thử tích hợp**: Kiểm tra tích hợp API (thanh toán, thông báo) bằng Postman/Cypress [REQ-076].
  + **Kiểm thử hiệu suất**: Đảm bảo phản hồi tìm kiếm <2s và xử lý 100 giao dịch đồng thời bằng JMeter [REQ-055, REQ-056].
  + **Kiểm thử bảo mật**: Ngăn chặn SQL Injection và truy cập trái phép bằng OWASP ZAP [REQ-078].
  + **Kiểm thử khả dụng**: Xác minh thời gian học <30 phút cho nhân viên [REQ-053].
* **Kịch bản kiểm thử**:
  + Đăng nhập với thông tin hợp lệ/không hợp lệ [REQ-001, REQ-016].
  + Thêm sản phẩm với mã trùng lặp [REQ-004].
  + Xử lý thanh toán qua Momo [REQ-007].
  + Tạo báo cáo doanh thu cho tháng cụ thể [REQ-011].

## 6.3. Lộ trình triển khai

* **Mốc thời gian**:
  + Tháng 1: Hoàn thiện thiết kế hệ thống và lược đồ CSDL.
  + Tháng 2: Phát triển frontend (ReactJS) và backend (Node.js/Express).
  + Tháng 3: Tích hợp dịch vụ thanh toán và thông báo.
  + Tháng 4: Kiểm thử và đào tạo nhân viên.
  + Tháng 5: Triển khai hệ thống và giám sát hiệu suất ban đầu.
* **Phân công nhiệm vụ**:
  + Frontend: 2 lập trình viên (triển khai ReactJS).
  + Backend: 2 lập trình viên (Node.js/Express, thiết lập MySQL).
  + Kiểm thử: 1 kỹ sư QA (kiểm thử đơn vị, tích hợp, hiệu suất).
  + Đào tạo: 1 huấn luyện viên cho nhân viên.