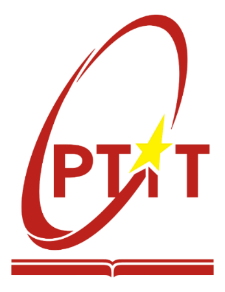
**BỘ THÔNG TIN VÀ TRUYỀN THÔNG**

TP.  
HCM

2023

NGUYỄN NGỌC MINH TÚ MSSV: N18DCCN192 Xây dưng ứng dụng web để quản lý và bán thực phẩm hữu cơ Lớp: D18CQCP02-N 2018-2023

**HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG**

------------------------------

**BÁO CÁO THỰC TẬP TỐT NGHIỆP**

**Đề tài: “Xây dựng ứng dụng web để quản lý và bán thực phẩm hữu cơ”**

**Người hướng dẫn : ThS. NGUYỄN ANH HÀO**

**Sinh viên thực hiện : NGUYỄN NGỌC MINH TÚ**

**Mã số sinh viên : N18DCCN192**

**Lớp : D18CQCP02-N**

**Khóa : 2018-2023**

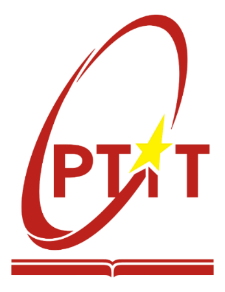
**Hệ : ĐẠI HỌC CHÍNH QUY**

**TP.HCM, tháng 03/2023**

**BỘ THÔNG TIN VÀ TRUYỀN THÔNG**

**HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG**

**------------------------------**



**BÁO CÁO THỰC TẬP TỐT NGHIỆP**

**Đề tài: “Xây dựng ứng dụng web để quản lý và bán thực phẩm hữu cơ”**

**Người hướng dẫn : ThS. NGUYỄN ANH HÀO**

**Sinh viên thực hiện : NGUYỄN NGỌC MINH TÚ**

**Mã số sinh viên : N18DCCN192**

**Lớp : D18CQCP02-N**

**Khóa : 2018-2023**

**Hệ : ĐẠI HỌC CHÍNH QUY**

**TP.HCM, tháng 03/2023**

TP.  
HCM

2023

**LỜI CẢM ƠN**

Em xin gửi lời cảm ơnchân thành đến các thầy cô của Khoa Công nghệ Thông tin 2 vì đã tạo điều kiện thuận lợi cho em thực hiện bài báo cáo thực tập này. Bên canh đó, em cũng muốn gửi lời cảm ơn đến thầy hướng dẫn đã luôn hỗ trợ và giúp đỡ em hoàn thành báo cáo tốt nhất có thể.

Đồng thời, em cũng xin gửi lời cảm ơn chân thành đến đơn vị nơi em thực tập – công ty ZIGVY, nơi làm việc mà em đã học được nhiều kiến thức hữu ích cả về chuyên môn và kĩ năng mềm, tại nơi ấy em được chào đón và hỗ trợ nhiệt tình từ quý anh chị, bạn bè và đồng nghiệp.

Em nhận thức được rằng bài báo cáo của mình có thể còn nhiều thiếu sót, vì vậy em mong muốn nhận được sự tiếp tục hỗ trợ, đóng góp ý kiến, phản hồi và hướng dẫn từ các thầy cô trong khoa và đơn vị thực tập để bài báo cáo của em được hoàn thiện hơn.

TP.HCM, tháng 03 năm 2023

Sinh viên thực hiện

Nguyễn Ngọc Minh Tú

**MỤC LỤC**

[Chương 1: GIỚI THIỆU ĐỀ TÀI 10](#_Toc131116068)

[1.1 Tên đề tài 10](#_Toc131116069)

[1.2 Loại đề tài 10](#_Toc131116070)

[1.3 Mục đích và mục tiêu của đề tài 10](#_Toc131116071)

[1.3.1 Mục đích 10](#_Toc131116072)

[1.3.2 Mục tiêu 10](#_Toc131116073)

[Chương 2: CƠ SỞ LÝ THUYẾT ĐỀ TÀI 11](#_Toc131116074)

[2.1 Công cụ phát triển 11](#_Toc131116075)

[2.1.1 Trình soạn thảo code - Visual Studio Code 11](#_Toc131116076)

[2.1.2 Cơ sở dữ liệu - MongoDB 11](#_Toc131116077)

[2.2 Thư viện / Framework nền tảng – ReactJS / NextJS 12](#_Toc131116078)

[2.2.1 Giới thiệu về ReactJS 12](#_Toc131116079)

[2.2.2 Giới thiệu về NextJS 12](#_Toc131116080)

[2.3 Một số thư viện nổi bật khác 12](#_Toc131116081)

[Chương 3: PHÂN TÍCH HỆ THỐNG 14](#_Toc131116082)

[3.1 Phân tích ngữ cảnh của hệ thống 14](#_Toc131116083)

[3.2 Lược đồ Use Case sơ lược khi phân tích 15](#_Toc131116084)

[3.3 Các yêu cầu và ràng buộc đối với hệ thống 17](#_Toc131116085)

[3.3.1 Từ môi trường nghiệp vụ 17](#_Toc131116086)

[3.3.2 Từ môi trường vận hành 18](#_Toc131116087)

[3.3.3 Từ môi trường phát triển 19](#_Toc131116088)

[Chương 4: THIẾT KẾ HỆ THỐNG 21](#_Toc131116089)

[4.1 Lược đồ Use Case 21](#_Toc131116090)

[4.1.1 Use Case tổng quát cho toàn bộ hệ thống 21](#_Toc131116091)

[4.1.2 Giải thích ý nghĩa mỗi Use Case 21](#_Toc131116092)

[4.2 Kiến trúc Webservice 30](#_Toc131116093)

[4.3 Thiết lập mô hình ERD (Entity Relationship Diagram) 30](#_Toc131116094)

[4.3.1 Xác định các thực thể 30](#_Toc131116095)

[4.3.2 Mô hình thực thể kết hợp ERD (Entity Relationship Diagram) 32](#_Toc131116096)

[4.3.3 Từ điển dữ liệu 34](#_Toc131116097)

[Chương 5: CHƯƠNG TRÌNH 44](#_Toc131116098)

[5.1 Giao diện Client 44](#_Toc131116099)

[5.2 Giao diện Admin 44](#_Toc131116100)

[Chương 6: KẾT LUẬN 45](#_Toc131116101)

[6.1 Kết quả đạt được và hạn chế 45](#_Toc131116102)

[6.2 Hạn chế và hướng phát triển 45](#_Toc131116103)

[Chương 7: PHỤ LỤC 46](#_Toc131116104)

[7.1 Danh mục tài liệu tham khảo 46](#_Toc131116105)

**KÍ HIỆU CÁC CỤM TỪ VIẾT TẮT**

|  |  |  |
| --- | --- | --- |
| **Từ viết tắt** | **Từ đầy đủ** | **Giải thích / Dịch** |
| API | Application Programming Interface | Giao diện lập trình ứng dụng |
| App | Application | Ứng dụng |
| VS Code | Visual Studio Code | Một ứng dụng soạn thảo code phổ biến |
| Dev | Developer | Người phát triển ứng dụng phần mềm |
| DOM | Document Object Model | Giao diện lập trình ứng dụng cho HTML |
| DFD | Data Flow Diagram | Mô hình luồng dữ liệu |
| ERD | Entity Relationship Diagram | Mô hình thực thể kết hợp |
| HTTP | HyperText Transfer Protocol | Giao thức truyền tải siêu văn bản |
| SQL | Structured Query Language | Ngôn ngữ truy vấn cấu trúc dữ liệu |
| RDBMS | Relational database management system | Hệ quản trị cơ sở dữ liệu quan hệ |
| REST | Representational State Transfer | Một tiêu chuẩn dùng trong việc thiết kế API cho các ứng dụng web |
| SPA | Single Page Application | Ứng dụng một trang |
| Component | Component | Thành phần của trang web |
| SSR | Server-side Rendering | Server gửi trực tiếp nội dung web (HTML/CSS) cho client |
| SSG | Static Site Generation | Nói về việc chuẩn bị sẵn trang web tĩnh cho client để tối ưu về tốc độ, thường sử dụng chung với SSR |
| SEO | Search Engine Optimization | Tối ưu hóa công cụ tìm kiếm, giúp trang web dễ dàng được tiếp cận bởi Google, Bing,… |
| UI | User Interface | Giao diện người dùng |
| UX | User experience | Trải nghiệm người dùng |
| KH | Khách hàng | Khách hàng, người đi mua hàng thông qua ứng dụng |
| QL | Quản lý | Nhân viên quản lý cửa hàng |
| HT | Hệ thống | Hệ thống / ứng dụng, phần xử lý các nghiệp vụ, logic,… |

**CHÚ GIẢI HÌNH**

[Hình 3.1 Use Case tổng quát – Giai đoạn phân tích 15](#_Toc131116021)

[Hình 4.1 Use Case tổng quát của toàn bộ hệ thống – Giai đoạn thiết kế 21](#_Toc131116022)

[Hình 4.2 Mô tả sơ lược kiến trúc Rest API 30](#_Toc131116023)

[Hình 4.3 Bảng chú thích các kí hiệu trong ERD 32](#_Toc131116024)

[Hình 4.4 ERD Tổng quát 33](#_Toc131116025)

[Hình 4.5 ERD Phân rã nghiệp vụ xuất nhập, quản lý kho 34](#_Toc131116026)

[Hình 4.6 ERD Phân rã nghiệp vụ bán hàng 34](#_Toc131116027)

**CHÚ GIẢI BẢNG**

[Bảng 3.1 Use Case Tra cứu sản phẩm 15](#_Toc131116106)

[Bảng 3.2 Use Case Cập nhật giỏ hàng 16](#_Toc131116107)

[Bảng 3.3 Use Case Đặt hàng 17](#_Toc131116108)

[Bảng 3.4 Use Case Huỷ đơn hàng 17](#_Toc131116109)

[Bảng 3.5 Yêu cầu và ràng buộc từ môi trường nghiệp vụ 18](#_Toc131116110)

[Bảng 3.6 Yêu cầu chức năng và ràng buộc từ môi trường vận hành 18](#_Toc131116111)

[Bảng 3.7 Yêu cầu phi chức năng và ràng buộc từ môi trường vận hành 19](#_Toc131116112)

[Bảng 3.8 Yêu cầu và ràng buộc từ môi trường phát triển 20](#_Toc131116113)

[Bảng 4.1 Use Case Đăng ký 22](#_Toc131116114)

[Bảng 4.2 Use Case Cập nhật thông tin tài khoản 23](#_Toc131116115)

[Bảng 4.3 Use Case Đăng xuất 23](#_Toc131116116)

[Bảng 4.4 Use Case Đổi mật khẩu 24](#_Toc131116117)

[Bảng 4.5 Use Case Đăng nhập 24](#_Toc131116118)

[Bảng 4.6 Use Case Khôi phục mật khẩu 25](#_Toc131116119)

[Bảng 4.7 Use Case Cập nhật thông tin sản phẩm 26](#_Toc131116120)

[Bảng 4.8 Use Case Cập nhật ảnh sản phẩm 26](#_Toc131116121)

[Bảng 4.9 Use Case Cập nhật thông tin nhà cung cấp 27](#_Toc131116122)

[Bảng 4.10 Use Case Xem thông tin các tài khoản trong hệ thống 27](#_Toc131116123)

[Bảng 4.11 Use Case Cập nhật đơn hàng 28](#_Toc131116124)

[Bảng 4.12 Use Case Cập nhật giờ làm việc 28](#_Toc131116125)

[Bảng 4.13 Use Case Cập nhật thông tin liên lạc 29](#_Toc131116126)

[Bảng 4.14 Use Case Xem thống kê 29](#_Toc131116127)

[Bảng 4.15 Use Case Xem thống kê 30](#_Toc131116128)

[Bảng 4.16 Mô tả ProductCategory 35](#_Toc131116129)

[Bảng 4.17 Mô tả OrderStatus 35](#_Toc131116130)

[Bảng 4.18 Mô tả Role 35](#_Toc131116131)

[Bảng 4.19 Mô tả Supplier 35](#_Toc131116132)

[Bảng 4.20 Mô tả Product 36](#_Toc131116133)

[Bảng 4.21 ProductImage 36](#_Toc131116134)

[Bảng 4.22 Mô tả ProductOrder 37](#_Toc131116135)

[Bảng 4.23 Mô tả Expiration 38](#_Toc131116136)

[Bảng 4.24 Mô tả ConsumptionHistory 38](#_Toc131116137)

[Bảng 4.25 Mô tả PasswordReset 39](#_Toc131116138)

[Bảng 4.26 Mô tả Account 39](#_Toc131116139)

[Bảng 4.27 Mô tả AccountAddress 40](#_Toc131116140)

[Bảng 4.28 Mô tả CustomerOrder 41](#_Toc131116141)

[Bảng 4.29 Mô tả CustomerOrderItem 42](#_Toc131116142)

[Bảng 4.30 Mô tả CartItem 42](#_Toc131116143)

[Bảng 4.31 Mô tả Store 43](#_Toc131116144)

[Bảng 4.32 Mô tả StoreHours 43](#_Toc131116145)

# GIỚI THIỆU ĐỀ TÀI

## Tên đề tài

Xây dựng ứng dụng web để quản lý và bán thực phẩm hữu cơ

## Loại đề tài

Xây dựng một phần mềm ứng dụng.

## Mục đích và mục tiêu của đề tài

### Mục đích

Cung cấp cho khách hàng và quản lý cửa hàng một giải pháp đơn giản và tiện lợi để giải quyết và tự động hoá các nghiệp vụ về đặt thực phẩm trực tuyến, qua đó giúp các quy trình như đặt hàng, quản lý thông tin đơn hàng, xuất nhập kho,… trở nên dễ dàng và hiệu quả hơn. Khách hàng có thể dễ dàng đặt và tra cứu thông tin sản phẩm, đặt hàng và quản lý đơn hàng. Quản lý cửa hàng có một công cụ giúp kiểm soát các thông tin liên quan, hỗ trợ các logic khi xuất nhập nguyên liệu, giảm thiểu tình trạng nhầm lẫn và phát sinh lỗi khi tính toán thủ công.

### Mục tiêu

Hướng đến một ứng dụng với giao diện đơn giản, dễ dùng nhưng không kém phần thông minh và hiệu quả. Để làm được điều đó, về mặt nghiệp vụ, cần phải tìm hiểu rõ các nghiệp vụ, logic về bán hàng và xuất nhập hàng để biết được các yêu cầu và tính năng mà phần mềm cần phải có. Về mặt kĩ thuật, cần áp dụng các công nghệ tiên tiến về web vào phần mềm để giúp tối ưu hoá hiệu năng và trải nghiệm người dùng.

# CƠ SỞ LÝ THUYẾT ĐỀ TÀI

## Công cụ phát triển

### Trình soạn thảo code - Visual Studio Code

Visual Studio Code (VS Code) là một trong những trình soạn thảo code miễn phí, mã nguồn mở rất được ưa chuộng bởi cộng đồng lập trình viên. Với các ưu điểm nổi bật là sự nhanh chóng, nhẹ, hỗ trợ đa nền tảng cùng nhiều tính năng thông qua một kho ứng dụng mở rộng (extension) do chính cộng đồng đóng góp phát triển, Intellisense chuyên nghiệp, giao diện thân thiện. Nhờ các ưu điểm nổi bật đó đã làm cho VS Code trở thành trình soạn thảo code phổ biến nhất hiện nay và vẫn đang không ngừng mở rộng về số lượng người dùng.

### Cơ sở dữ liệu - MongoDB

MongoDB là một hệ quản trị cơ sở dữ liệu phi quan hệ (No SQL) được phát triển bởi MongoDB Inc. MongoDB cho phép lưu trữ và truy xuất dữ liệu theo cấu trúc tài liệu, thay vì lưu trữ trong các bảng như trong các hệ quản trị cơ sở dữ liệu quan hệ (RDBMS) truyền thống. Điều này giúp cho MongoDB dễ dàng mở rộng và phù hợp với các dự án có tính mở rộng cao.

Với các tính năng mở rộng đa dạng và khả năng lưu trữ tài liệu có cấu trúc linh hoạt, MongoDB đã trở thành một trong những hệ quản trị cơ sở dữ liệu NoSQL phổ biến nhất hiện nay. MongoDB được sử dụng rộng rãi trong các ứng dụng web và mobile, và được đánh giá cao vì tính đơn giản và hiệu suất của nó.

Một số lợi thế của MongoDB so với RDBMS:

1. Ít Schema hơn: MongoDB là một cơ sở dữ liệu dựa trên Document, trong đó một Collection giữ các Document khác nhau. Số trường, nội dung và kích cỡ của Document này có thể khác với Document khác.
2. Cấu trúc của một đối tượng là rõ ràng.
3. Không có các Join phức tạp.
4. Khả năng truy vấn sâu hơn. MongoDB hỗ trợ các truy vấn động trên các Document bởi sử dụng một ngôn ngữ truy vấn dựa trên Document mà mạnh mẽ như SQL.
5. MongoDB dễ dàng để mở rộng.
6. Việc chuyển đổi/ánh xạ của các đối tượng ứng dụng đến các đối tượng cơ sở dữ liệu là không cần thiết.
7. Sử dụng bộ nhớ nội tại để lưu giữ phần công việc, giúp truy cập dữ liệu nhanh hơn.

## Thư viện / Framework nền tảng – ReactJS / NextJS

### Giới thiệu về ReactJS

ReactJS là một thư viện mã nguồn mở dùng để xây dựng UI, vô cùng phổ biến với đông đảo người dùng, được phát triển bởi Facebook. Với đặc thù là hỗ trợ xây dựng những thành phần (components) UI có tính tương tác cao, có trạng thái và có thể sử dụng lại được. ReactJS cho phép người dùng tạo các ứng dụng web một trang (Single page application - SPA) một cách dễ dàng.

SPA là một loại ứng dụng web giúp đem lại cho người dùng những trải nghiệm mượt mà như trên một ứng dụng native (Mobile app, desktop app). Tại đó, người dùng sẽ thực hiện tất cả mọi thao tác trên một trang duy nhất, mọi cấu trúc trang sẽ chỉ tải một lần và không tải lại khi chuyển trang.

### Giới thiệu về NextJS

NextJS là một framework mã nguồn mở của React được phát triển bởi Vercel. Điểm nổi bật của NextJS là khả năng hỗ trợ tính năng server-side rendering, giúp cải thiện tốc độ tải trang. Với tính năng này, các trang web được tạo ra từ phía máy chủ, giúp tăng tốc độ hiển thị và khả năng tương tác của trang web. NextJS còn hỗ trợ tối ưu hóa cho các công cụ tìm kiếm (SEO - Search Engine Optimization, đồng nghĩa với việc người dùng có thể dễ dàng tìm thấy trang web khi sử dụng Google, Bing,…

Dù là framework của một thư viện về phát triển giao diện, NextJS cũng tích hợp **khả năng phát triển web API (Server-side code) ngay trong cùng một dự án**. Qua đó, tăng khả năng tương tác giữa giao diện và logic, giúp ứng dụng trở nên chặt chẽ và mở rộng dễ dàng.

## Một số thư viện nổi bật khác

**Mongoose** là một thư viện mã nguồn mở được sử dụng để kết nối ứng dụng Node.js với MongoDB. Nó cung cấp cho các nhà phát triển một cách tiếp cận dễ dàng và trực quan để tương tác với cơ sở dữ liệu MongoDB, đồng thời giảm thiểu khả năng xảy ra lỗi và giảm thiểu thời gian phát triển.

**Redux** là một thư viện quản lý trạng thái phổ biến trong React, cho phép người dùng quản lý trạng thái của ứng dụng trong một nơi duy nhất. Với Redux, người dùng có thể truy cập trạng thái của ứng dụng một cách dễ dàng và hiệu quả, đồng thời cũng giúp tăng tính mở rộng của ứng dụng. Redux có thể sử dụng để quản lý cả các trạng thái của ứng dụng (Client-state) và các dữ liệu từ Server gửi về (Server-state), trong ứng dụng này, ta chỉ sử dụng Redux để quản lý Client-state, về phía Server-state, ta sẽ sử dụng React Query.

**React Query** là một thư viện cải tiến cho việc quản lý dữ liệu nhận được khi gọi API từ back-end trong ReactJS, giúp tối ưu hóa các yêu cầu mạng và xử lý dữ liệu. React Query cung cấp một bộ công cụ toàn diện giúp Developer kiểm soát các trạng thái khi gọi API, nhận và xử lý dữ liệu, xử lý lỗi, cache/refresh dữ liệu từ back-end một cách vô cùng đơn giản.

**Next Auth** là một thư viện cho phép người dùng xác thực người dùng và quản lý phiên đăng nhập trong ứng dụng NextJS mà không cần phải trực tiếp quản lý các Token hay Session, qua đó, giúp Developer có thể tạo ra các chức năng đăng nhập, đăng ký và đăng xuất trong ứng dụng của mình một cách dễ dàng và hiệu quả. Next Auth cũng hỗ trợ các nhà cung cấp xác thực bên ngoài như Google, Facebook, Twitter,…

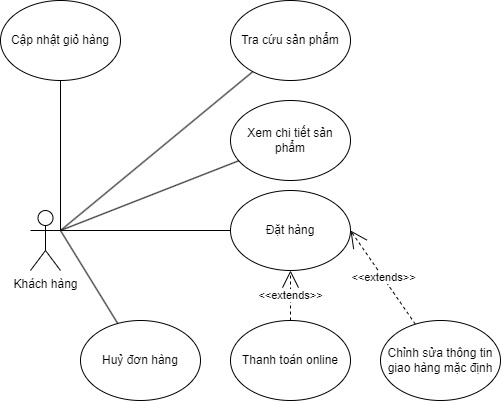
# PHÂN TÍCH HỆ THỐNG

## Phân tích ngữ cảnh của hệ thống

Nhìn từ bên ngoài vào, công ty/tổ chức quản lý việc vận hành đặt thực phẩm trực tuyến trước khi áp dụng phần mềm vào hệ thống có thể sẽ gặp một số vấn đề như sau:

* **Chi phí:** Trước khi ứng dụng phần mềm hỗ trợ toàn diện, công ty/tổ chức có thể phải chi tiêu nhiều tiền để duy trì và nâng cấp các công cụ, phần mềm khác nhau để hỗ trợ cho việc xử lý các nghiệp vụ, như Microsoft Office để quản lý thông tin, nhưng các phần việc như quản lý xuất nhập, ràng buộc dữ liệu, đa phần vẫn phải tính toán thủ công ở bên ngoài hoặc từ các phần mềm khác trước khi có thể đưa Microsoft Office. Điều này sẽ làm tăng chi phí cho việc mua bản quyền của nhiều phần mềm, do không có một phần mềm chuyên dụng, qua đó làm giảm lợi nhuận của công ty/tổ chức.
* **Thời gian:** Trước khi ứng dụng phần mềm hỗ trợ toàn diện, công ty/tổ chức có thể sẽ tiếp nhận đơn hàng qua điện thoại và quản lý thông tin qua các công cụ phần mềm khác nhau, họ có thể mất nhiều thời gian để xử lý đơn hàng và quản lý dữ liệu. Điều này có thể làm giảm hiệu quả hoạt động, tăng thời gian cần thiết để xử lý đơn hàng, các nghiệp vụ nhập xuất kho của công ty/tổ chức.
* **Hiệu quả công việc và chất lượng dịch vụ:** Không những tốn thời gian và chi phí, việc không có phần mềm chuyên dụng cũng làm tăng rủi ro dẫn đến sai sót trong quá trình tính toán thủ công, hoặc tính toán bằng nhiều phần mềm, tính toán qua nhiều bước, làm ảnh hưởng đến chất lượng dịch vụ và hiệu quả công việc

## Lược đồ Use Case sơ lược khi phân tích



Hình . Use Case tổng quát – Giai đoạn phân tích

Mô tả các use case chính:

* **Use Case Tra cứu sản phẩm**

|  |  |
| --- | --- |
| **ID** | UC\_TCSP |
| **Tên** | Tra cứu sản phẩm |
| **Tác nhân chính** | Khách hàng |
| **Ý nghĩa** | Khi khách hàng muốn tìm kiếm sản phẩm theo tên |
| **Điều kiện kích hoạt** | Khách hàng nhập từ khoá vào khung tìm kiếm |
| **Điều kiện trước** |  |
| **Chuỗi sự kiện chính**   1. Khách hàng nhập từ khoá vào khung tìm kiếm 2. Hệ thống tiến hành tra cứu và trả về các sản phẩm thoả yêu cầu | |
| **Ngoại lệ**  2.1 Không có sản phẩm nào thoả yêu cầu  2.1.1 Hệ thống thông báo với khách hàng không có sản phẩm nào được tìm thấy  2.1.2 Quay lại bước 1 | |

Bảng . Use Case Tra cứu sản phẩm

* **Use Case Xem chi tiết sản phẩm**

|  |  |
| --- | --- |
| **ID** | UC\_CTSP |
| **Tên** | Xem chi tiết sản phẩm |
| **Tác nhân chính** | Khách hàng |
| **Ý nghĩa** | Khách hàng muốn xem chi tiết sản một sản phẩm nào đó |
| **Điều kiện kích hoạt** | Khách hàng nhấp vào một sản phẩm bất kỳ |
| **Điều kiện trước** |  |
| **Chuỗi sự kiện chính**   1. Khách hàng nhấp vào một sản phẩm bất kỳ 2. Hệ thống tiến hành tra cứu và trả về dữ liệu chi tiết của sản phẩm đó | |

* **Use Case Cập nhật giỏ hàng**

|  |  |
| --- | --- |
| **ID** | UC\_GH |
| **Tên** | Cập nhật giỏ hàng |
| **Tác nhân chính** | Khách hàng |
| **Ý nghĩa** | Khách hàng muốn thêm / bớt sản phẩm trong giỏ hàng |
| **Điều kiện kích hoạt** | Khách hàng bấm nút thêm / bớt ở một sản phẩm |
| **Điều kiện trước** | Khách hàng đã đăng nhập thành công |
| **Chuỗi sự kiện chính:**   1. Khách hàng tìm kiếm và xem thông tin sản phẩm muốn mua 2. Khách hàng bấm nút thêm / bớt số lượng của một sản phẩm nào đó tại giỏ hàng 3. Hệ thống hiệu chỉnh giỏ hàng theo yêu cầu và hiện phản hồi khi tác vụ xử lý hoàn tất | |
| **Ngoại lệ**  3.1 Số lượng tồn kho đạt giới hạn  3.1.1 Hệ thống vô hiệu hoá nút thêm và hiện lý do cho người dùng biết  3.1.2 Quay lại bước 2  3.2 Tồn tại sản phẩm đã có trong giỏ hàng mà tồn kho không còn đủ số lượng  3.2.1 Hệ thống vô hiệu hoá chức năng thanh toán, yêu cầu người dùng giảm số lượng về đến mức còn đủ trong kho | |

Bảng . Use Case Cập nhật giỏ hàng

* **Use Case Đặt hàng**

|  |  |
| --- | --- |
| **ID** | UC\_DH |
| **Tên** | Đặt hàng |
| **Tác nhân chính** | Khách hàng |
| **Ý nghĩa** | Khách hàng muốn thanh toán các sản phẩm trong giỏ hàng |
| **Điều kiện kích hoạt** | Khách hàng bấm nút thanh toán |
| **Điều kiện trước** | Khách hàng đã đăng nhập thành công  Giỏ hàng có sản phẩm |
| **Chuỗi sự kiện chính**   1. Khách hàng chọn nút thanh toán 2. Hệ thống hiện chi tiết giỏ hàng và form điền thông tin đơn hàng với các giá trị mặc định lấy từ tài khoản 3. Khách hàng điều chỉnh thông tin **(nếu có)** và bấm tiếp tục 4. Hệ thống tính toán và hiển thị thông tin giao hàng (quãng đường, thời gian ước tính) để khách hàng xác 5. Khách hàng bấm đồng ý xác nhận và qua bước tiếp theo 6. Hệ thống hiện các tuỳ chọn phương thức thanh toán 7. Khách hàng chọn một phương thức thanh toán, tiến hành thanh toán online **(nếu có)** 8. Hệ thống tiến hành đặt đơn hàng, giảm số lượng nguyên liệu và làm các nghiệp vụ cần thiết khác | |
| **Ngoại lệ:**  4.1 Cửa hàng không thể giao do khoảng cách quá xa, hoặc thời gian giao hàng quá lâu do tình trạng giao thông  4.1.1 Hệ thống hiện thông báo cho khách hàng biết nguyên nhân  4.1.2 Quay lại bước 3 | |

Bảng . Use Case Đặt hàng

* **Use Case Huỷ đơn hàng**

|  |  |
| --- | --- |
| **ID** | UC\_HDH |
| **Tên** | Huỷ đơn hàng |
| **Tác nhân chính** | Khách hàng |
| **Ý nghĩa** | Khách hàng muốn huỷ một đơn hàng đã đặt |
| **Điều kiện kích hoạt** | Khách hàng bấm nút huỷ đơn tại một đơn hàng có thể huỷ |
| **Điều kiện trước** | Tồn tại đơn hàng có thể huỷ |
| **Chuỗi sự kiện chính**   1. Khách hàng chọn một đơn hàng có thể huỷ 2. Khách hàng chọn nút huỷ đơn hàng 3. Hệ thống hiện thông báo xác nhận 4. Khách hàng chọn đồng ý huỷ đơn 5. Hệ thống tiến hành huỷ đơn, hoàn trả số lượng tồn kho và gửi thông báo cho người dùng khi hoàn tất | |

Bảng . Use Case Huỷ đơn hàng

## Các yêu cầu và ràng buộc đối với hệ thống

### Từ môi trường nghiệp vụ

|  |  |  |  |
| --- | --- | --- | --- |
| **Use Case** | **Req-ID** | **Nội dung yêu cầu** | **Stack-Holder** |
| UC\_DH | F\_DH\_1 | **Cửa hàng cần có cam kết về chính sách giao hàng**   * Bán kính giao hàng tối đa: 10km * Thời gian giao hàng tối đa: 45 phút * Chỉ giao trong giờ làm việc, mỗi ngày có thể có giờ làm việc khác nhau, thông tin này được nêu rõ ở website | Khách hàng |
| F\_DH\_2 | **Cần có cơ chế trừ nguyên liệu trong kho khi đặt hàng hợp lý và hiệu quả**  Khi đặt hàng thành công, hệ thống phải trừ số lượng của các sản phẩm tồn kho có ngày hết hạn gần nhất, để tránh lãng phí nguyên liệu | Quản lý cửa hàng |
| UC\_HDH | F\_HDH\_1 | **Cửa hàng cần có cam kết về chính sách huỷ đơn**  Đơn hàng chỉ có thể huỷ khi đang ở trạng thái chờ xác nhận hoặc đang xử lý. Nếu đơn hàng đã qua trạng thái đang giao hoặc đã nhận hàng thì không thể huỷ | Khách hàng |

Bảng . Yêu cầu và ràng buộc từ môi trường nghiệp vụ

### Từ môi trường vận hành

Các yêu cầu chức năng:

|  |  |  |  |
| --- | --- | --- | --- |
| **Use Case** | **Req-ID** | **Nội dung yêu cầu** | **Stack-Holder** |
| UC\_GH | F\_GH\_1 | **Hệ thống cần phải có cơ chế theo dõi số lượng tồn kho linh hoạt và thường xuyên**  Cần cập nhật số lượng tồn kho thường xuyên và báo ngay cho người dùng khi có sản phẩm trong giỏ hàng không còn số lượng để giao, tránh để đến lúc thanh toán mới kiểm tra rồi báo thiếu, làm ảnh hưởng trải nghiệm người dùng | Khách hàng |

Bảng . Yêu cầu chức năng và ràng buộc từ môi trường vận hành

Các yêu cầu phi chức năng:

|  |  |  |  |
| --- | --- | --- | --- |
| **Use Case** | **Req-ID** | **Nội dung yêu cầu** | **Stack-Holder** |
| All | NF\_ALL\_1 | **Giao diện của ứng dụng phải tuỳ biến linh hoạt cho nhiều loại thiết bị (Responsive)**  Ứng dụng phải có giao diện thân thiện cho cả màn hình lớn (desktop) và màn hình nhỏ (mobile) | Quản lý cửa hàng, Khách hàng |
| NF\_ALL\_2 | **Ứng dụng phải thân thiện, tối ưu cho các công cụ tìm kiếm**  Ứng dụng cần dễ dàng được tìm thấy bởi các công cụ tìm kiếm phổ biến hiện nay như Google, Bing,… để có thể tiếp cận được với tập khách hàng người dùng lớn |
| NF\_ALL\_3 | **Authentication/Authorization**  Cần có các cơ chế bảo mật, phân quyền, chỉ những tài khoản có đủ quyền mới có thể truy cập một số tài nguyên được bảo vệ của hệ thống, các yêu cầu không đủ quyền hạn sẽ bị chặn hoặc chuyển hướng. |
| NF\_ALL\_4 | **Performance**   * Đối với các tác vụ nặng, cần xử lý dữ liệu ở nhiều nơi (Đặt hàng, nhập nguyên liệu,…): không quá 2 giây * Đối với các tác vụ nhẹ và nhiều (Hiệu chỉnh giỏ hàng: không quá 1 giây |
| NF\_ALL\_5 | **Maintainability**   * Tính tái sử dụng: Có * Khả năng sửa đổi / mở rộng: Có * Khả năng kiểm thử: Có | Developer |

Bảng . Yêu cầu phi chức năng và ràng buộc từ môi trường vận hành

### Từ môi trường phát triển

|  |  |  |  |
| --- | --- | --- | --- |
| **Use Case** | **Đối tượng** | **Nội dung yêu cầu** | **Stack-Holder** |
| TD01 | Software | * Viết code trên Visual Studio Code * Cơ sở dữ liệu sử dụng MongoDB version 6, phải đáp ứng được các ràng buộc toàn vẹn về khoá ngoại * Ứng dụng được phát triển dựa trên nền tảng ReactJS version 18, NextJS version 13. | Developer |
| TA02 | Software | Ứng dụng phải chạy ổn định trên các trình duyệt phổ biến như:   * Chrome: version 88+ * Edge: version 88+ * Safari: version 16+ * Firefox: version 89+ | Khách hàng, Quản lý cửa hàng |

Bảng . Yêu cầu và ràng buộc từ môi trường phát triển

# THIẾT KẾ HỆ THỐNG

## Lược đồ Use Case

### Use Case tổng quát cho toàn bộ hệ thống



Hình . Use Case tổng quát của toàn bộ hệ thống – Giai đoạn thiết kế

### Giải thích ý nghĩa mỗi Use Case

**Người dùng**:

* **Use Case Đăng ký**

|  |  |
| --- | --- |
| **ID** | UC\_DK |
| **Tên** | Đăng ký |
| **Tác nhân chính** | Người dùng |
| **Ý nghĩa** | Người dùng muốn đăng ký tài khoản để trở thành khách hàng |
| **Điều kiện kích hoạt** | Người dùng chọn lệnh đăng ký |
| **Điều kiện trước** |  |
| **Chuỗi sự kiện chính:**   1. Người dùng truy cập và chọn lệnh đăng ký 2. Hệ thống hiện form đăng ký 3. Người dùng nhập các thông tin cần thiết và bấm đăng ký 4. Hệ thống kiểm tra thông tin, nếu hợp lệ sẽ tạo tài khoản, gửi mail xác nhận, hiện thông báo yêu cầu người dùng kiểm tra mail và điều hướng người dùng về màn hình đăng nhập | |
| **Ngoại lệ**  4.1 Thông tin người dùng cung cấp không hợp lệ (Email đã tồn tại, xác nhận mật khẩu không đúng,…)  4.1.1 Hệ thống báo lỗi và yêu cầu khách hàng nhập lại  4.1.2 Quay lại bước 3 | |

Bảng . Use Case Đăng ký

* **Use Case Tra cứu sản phẩm**

Đã trình bày ở mục [**3.2**](#_Lược_đồ_Use)

* **Use Case Xem chi tiết sản phẩm**

Đã trình bày ở mục [**3.2**](#_Lược_đồ_Use)

**Khách hàng:**

* **Use Case Cập nhật giỏ hàng**

Đã trình bày ở mục [**3.2**](#_Lược_đồ_Use)

* **Use Case Đặt hàng**

Đã trình bày ở mục [**3.2**](#_Lược_đồ_Use)

* **Use Case Cập nhật thông tin tài khoản**

|  |  |
| --- | --- |
| **ID** | UC\_CNTK |
| **Tên** | Cập nhật thông tin tài khoản |
| **Tác nhân chính** | Khách hàng |
| **Ý nghĩa** | Người dùng muốn thay đổi một số thông tin trên tài khoản (Ảnh đại diện, họ tên,…) |
| **Điều kiện kích hoạt** | Khách hàng chọn lệnh cập nhật thông tin tài khoản |
| **Điều kiện trước** | Khách hàng đã đăng nhập thành công |
| **Chuỗi sự kiện chính:**   1. Khách hàng chọn lệnh cập nhật thông tin tài khoản 2. Hệ thống hiện form chứa các thông tin có thể thay đổi, các thông tin không thể thay đổi sẽ được tô xám báo hiệu 3. Khách hàng thay đổi các thông tin theo nhu cầu và chọn lưu 4. Hệ thống kiểm tra tính hợp lệ của thông tin, nếu đúng sẽ lưu thay đổi và hiện thông báo cập nhật thành công | |
| **Ngoại lệ**  4.1 Thông tin người dùng nhập không hợp lệ  4.1.1 Hệ thống báo lỗi và yêu cầu khách hàng nhập lại  4.1.2 Quay lại bước 3 | |

Bảng . Use Case Cập nhật thông tin tài khoản

* **Use Case Đăng xuất**

|  |  |
| --- | --- |
| **ID** | UC\_DX |
| **Tên** | Đăng xuất |
| **Tác nhân chính** | Khách hàng |
| **Ý nghĩa** | Khách hàng muốn đăng xuất khỏi hệ thống |
| **Điều kiện kích hoạt** | Khách hàng chọn lệnh đăng xuất |
| **Điều kiện trước** | Khách hàng đã đăng nhập thành công |
| **Chuỗi sự kiện chính:**   1. Khách hàng chọn lệnh đăng xuất 2. Hệ thống hiện thông báo yêu cầu xác nhận 3. Khách hàng xác nhận hành động 4. Hệ thống đăng xuất tài khoản và quay về màn hình chính | |
| **Ngoại lệ**  3.1 Khách hàng huỷ hành động  3.1.1 Thông báo xác nhận tắt, khách hàng không bị đăng xuất | |

Bảng . Use Case Đăng xuất

* **Use Case Đổi mật khẩu**

|  |  |
| --- | --- |
| **ID** | UC\_DMK |
| **Tên** | Đổi mật khẩu |
| **Tác nhân chính** | Khách hàng |
| **Ý nghĩa** | Khách hàng muốn đổi mật khẩu của tài khoản |
| **Điều kiện kích hoạt** | Khách hàng chọn lệnh đổi mật khẩu |
| **Điều kiện trước** | Khách hàng đã đăng nhập thành công |
| **Chuỗi sự kiện chính:**   1. Khách hàng chọn lệnh đổi mật khẩu 2. Hệ thống hiện form, yêu cầu nhập mật khẩu cũ, mật khẩu mới và xác nhận mật khẩu mới 3. Khách hàng nhập thông tin và chọn cập nhật 4. Hệ thống kiểm tra tính hợp lệ của của thông tin (Mật khẩu cũ đúng hay chưa, mật khẩu mới xác nhận mật khẩu mới có khớp nhau hay không) nếu hợp lệ sẽ tiến hành cập nhật | |
| **Ngoại lệ**  4.1 Mật khẩu cũ không chính xác hoặc mật khẩu mới và xác nhận mật khẩu không trùng nhau  4.1.1 Hệ thống báo lỗi và yêu cầu khách hàng nhập lại  4.1.2 Quay lại bước 3 | |

Bảng . Use Case Đổi mật khẩu

* **Use Case Đăng nhập**

|  |  |
| --- | --- |
| **ID** | UC\_DN |
| **Tên** | Đăng nhập |
| **Tác nhân chính** | Khách hàng |
| **Ý nghĩa** | Khách hàng muốn đăng nhập vào hệ thống để tiến hành mua hàng |
| **Điều kiện kích hoạt** | Khách hàng chọn lệnh đăng nhập |
| **Điều kiện trước** | Khách hàng đã có tài khoản |
| **Chuỗi sự kiện chính:**   1. Khách hàng chọn lệnh đăng nhập 2. Hệ thống hiện dialog chứa form đăng nhập 3. Người dùng điền đầy đủ thông tin (email, mật khẩu) và bấm xác nhận 4. Hệ thống tiến hành kiểm tra thông tin tài khoản người dùng nhập vào, nếu hợp lệ sẽ điều hướng về màn hình chính | |
| **Ngoại lệ**  4.1 Thông tin tài khoản người dùng nhập vào không tồn tại  4.1.1 Hệ thống báo lỗi yêu cầu người dùng kiểm tra lại thông tin  4.1.2 Quay lại bước 3 | |

Bảng . Use Case Đăng nhập

* **Use Case Khôi phục mật khẩu**

|  |  |
| --- | --- |
| **ID** | UC\_KPMK |
| **Tên** | Khôi phục mật khẩu |
| **Tác nhân chính** | Khách hàng |
| **Ý nghĩa** | Khách hàng đã có tài khoản, nhưng quên mật khẩu để đăng nhập |
| **Điều kiện kích hoạt** | Khách hàng chọn lệnh quên mật khẩu ở màn hình đăng nhập |
| **Điều kiện trước** | Khách hàng đã có tài khoản |
| **Chuỗi sự kiện chính:**   1. Khách hàng chọn lệnh khôi phục mật khẩu tại giao diện đăng nhập 2. Hệ thống hiển thoại hộp thoại yêu cầu khách hàng nhập email đã đăng ký trước đó 3. Khách hàng nhập email và chọn khôi phục mật khẩu 4. Hệ thống kiểm tra email tồn tại không, nếu có thì gửi thông báo KH cần xem mail đã gửi tới email KH và làm theo hướng dẫn 5. Khách hàng chọn đồng ý để xác nhận đã xem thông báo 6. Hệ thống quay lại màn hình chính | |
| **Ngoại lệ**  4.1 Email nhập vào không hợp lệ hoặc không tồn tại  4.1.1 Hệ thống phát thông báo email không đúng  4.1.2 Quay lại bước 3 | |

Bảng . Use Case Khôi phục mật khẩu

**Quản lý:**

* **Use Case Cập nhật thông tin sản phẩm**

|  |  |
| --- | --- |
| **ID** | UC\_CNTTSP |
| **Tên** | Cập nhật thông tin sản phẩm |
| **Tác nhân chính** | Quản lý |
| **Ý nghĩa** | Quản lý muốn cập nhật (thêm, sửa) thông tin của một sản phẩm nào đó |
| **Điều kiện kích hoạt** | Quản lý chọn lệnh thêm sản phẩm, hoặc chọn lệnh sửa sản phẩm ở một sản phẩm muốn sửa |
| **Điều kiện trước** | Quản lý đã đăng nhập |
| **Chuỗi sự kiện chính:**   1. Quản lý chọn lệnh thêm sản phẩm, hoặc chọn lệnh sửa sản phẩm ở một sản phẩm muốn sửa 2. Hệ thống hiện form để chứa các thông tin của sản phẩm 3. Quản lý điền / sửa các thông tin cần thiết 4. Hệ thống kiểm tra tính hợp lệ của thông tin, nếu đúng sẽ tiến hành lưu vào CSDL và thông báo thành công | |
| **Ngoại lệ**  4.1 Thông tin nhập vào không hợp lệ (Thiếu dữ liệu, dữ liệu bị trùng,…)  4.1.1 Hệ thống báo lỗi cho quản lý biết  4.1.2 Quay lại bước 3 | |

Bảng . Use Case Cập nhật thông tin sản phẩm

* **Use Case Cập nhật ảnh sản phẩm**

|  |  |
| --- | --- |
| **ID** | UC\_CNASP |
| **Tên** | Cập nhật ảnh sản phẩm |
| **Tác nhân chính** | Quản lý |
| **Ý nghĩa** | Quản lý muốn cập nhật (thêm, xoá) ảnh của một sản phẩm mong muốn, use case này là bắt buộc đối với việc thêm sản phẩm mới |
| **Điều kiện kích hoạt** | Quản lý chọn một sản phẩm để tiến hành cập nhật hoặc khi thêm sản phẩm mới |
| **Điều kiện trước** | Quản lý đã đăng nhập |
| **Chuỗi sự kiện chính:**   1. Quản lý chọn cập nhật ảnh sản phẩm hoặc thêm sản phẩm mới 2. Hệ thống hiện danh sách các ảnh hiện có của sản phẩm (nếu như là cập nhật sản phẩm hiện có) và nút thêm / xoá để người dùng tiến hành cập nhật 3. Người dùng tiến hành thao tác thêm / xoá theo nhu cầu 4. Hệ thống tiến hành lưu vào CSDL sau mỗi thao tác của người dùng | |
| **Ngoại lệ**  3.1 Số lượng ảnh sản phẩm chỉ còn 1, và người dùng chọn xoá nó  3.1.1 Hệ thống báo lỗi cho quản lý biết  3.1.2 Quay lại bước 2 | |

Bảng . Use Case Cập nhật ảnh sản phẩm

* **Use Case Cập nhật thông tin nhà cung cấp**

|  |  |
| --- | --- |
| **ID** | UC\_CNNCC |
| **Tên** | Cập nhật thông tin nhà cung cấp |
| **Tác nhân chính** | Quản lý |
| **Ý nghĩa** | Quản lý muốn cập nhật (thêm / sửa) một nhà cung cấp |
| **Điều kiện kích hoạt** | Quản lý chọn lệnh cập nhật thông tin nhà cung cấp |
| **Điều kiện trước** | Quản lý đã đăng nhập |
| **Chuỗi sự kiện chính:**   1. Quản lý chọn lệnh cập nhật thông tin nhà cung cấp 2. Hệ thống hiển thị form với các trường thông tin của nhà cung cấp 3. Quản lý tiến hành điền / sửa các thông tin cần thiết và bấm lưu 4. Hệ thống tiến hành kiểm tra tính hợp lệ, nếu đúng sẽ lưu vào CSDL | |
| **Ngoại lệ**  4.1 Thông tin nhập vào không hợp lệ  4.1.1 Hệ thống báo lỗi  4.1.2 Quay lại bước 3 | |

Bảng . Use Case Cập nhật thông tin nhà cung cấp

* **Use Case Xem thông tin các tài khoản trong hệ thống**

|  |  |
| --- | --- |
| **ID** | UC\_XTTTK |
| **Tên** | Xem thông tin các tài khoản trong hệ thống |
| **Tác nhân chính** | Quản lý |
| **Ý nghĩa** | Quản lý muốn xem thông tin các tài khoản trong hệ thống |
| **Điều kiện kích hoạt** | Quản lý chọn lệnh xem thông tin tài khoản |
| **Điều kiện trước** | Quản lý đã đăng nhập |
| **Chuỗi sự kiện chính:**   1. Quản lý chọn lệnh xem thông tin tài khoản 2. Hệ thống hiện danh sách thông tin các tài khoản hiện có trong hệ thống 3. Người dùng chọn một tài khoản theo mong muốn để xem chi tiết (nếu có) | |

Bảng . Use Case Xem thông tin các tài khoản trong hệ thống

* **Use Case Cập nhật đơn hàng**

|  |  |
| --- | --- |
| **ID** | UC\_CNDH |
| **Tên** | Cập nhật đơn hàng |
| **Tác nhân chính** | Quản lý |
| **Ý nghĩa** | Quản lý muốn cập nhật trạng thái của một đơn hàng |
| **Điều kiện kích hoạt** | Quản lý chọn lệnh cập nhật trạng thái ở một đơn hàng |
| **Điều kiện trước** | Quản lý đã đăng nhập  Tồn tại đơn hàng có thể cập nhật |
| **Chuỗi sự kiện chính:**   1. Quản lý chọn vào mục đơn hàng ở menu chính 2. Hệ thống hiển thị danh sách các đơn hàng đang có trong hệ thống 3. Quản lý nhập mã vào khung tìm kiếm (nếu cần thiết) để tìm đơn hàng 4. Quản lý chọn đơn hàng mong muốn để xem thông tin 5. Hệ thống trả ra thông tin chi tiết của đơn hàng cùng với nút cập nhật 6. Quản lý bấm vào nút cập nhật đơn hàng 7. Hệ thống hiện thông báo yêu cầu xác nhận hành động 8. Quản lý bấm xác nhận hành động 9. Hệ thống cập nhật cấp trạng thái đơn hàng lên trạng thái tiếp theo | |
| **Ngoại lệ**  7.1 Quản lý huỷ hành động cập nhật  7.1.1 Hệ thống không cập nhật đơn hàng | |

Bảng . Use Case Cập nhật đơn hàng

* **Use Case Cập nhật giờ làm việc**

|  |  |
| --- | --- |
| **ID** | UC\_CNGLV |
| **Tên** | Cập nhật giờ làm việc |
| **Tác nhân chính** | Quản lý |
| **Ý nghĩa** | Quản lý muốn cập nhật giờ làm việc để thay đổi giờ giao hàng và hiển thị thông tin này cho khách hàng biết |
| **Điều kiện kích hoạt** | Quản lý chọn lệnh cập nhật thông tin cửa hàng, phần cập nhật giờ làm việc |
| **Điều kiện trước** | Quản lý đã đăng nhập |
| **Chuỗi sự kiện chính:**   1. Quản lý chọn lệnh cập nhật thông tin cửa hàng, phần cập nhật giờ làm việc 2. Hệ thống hiển thị form gồm 7 ngày trong tuần và các trường giờ bắt đầu và kết thúc 3. Quản lý tiến hành sửa đổi giờ làm việc theo nhu cầu và bấm lưu 4. Hệ thống kiểm tra tính hợp lệ của thông tin, nếu đúng sẽ lưu vào CSDL | |
| **Ngoại lệ**  4.1 Thông tin giờ làm việc không hợp lệ (Giờ kết thúc sớm hơn giờ bắt đầu)  4.1.1 Hệ thống báo lỗi cho quản lý biết  4.1.2 Quay lại bước 3 | |

Bảng . Use Case Cập nhật giờ làm việc

* **Use Case Cập nhật thông tin liên lạc**

|  |  |
| --- | --- |
| **ID** | UC\_CNTTLL |
| **Tên** | Cập nhật thông tin liên lạc |
| **Tác nhân chính** | Quản lý |
| **Ý nghĩa** | Quản lý muốn cập nhật thông tin liên lạc và hiển thị thông tin này cho khách hàng biết |
| **Điều kiện kích hoạt** | Quản lý chọn lệnh cập nhật thông tin cửa hàng, phần cập nhật thông tin liên lạc |
| **Điều kiện trước** | Quản lý đã đăng nhập |
| **Chuỗi sự kiện chính:**   1. Quản lý chọn lệnh cập nhật thông tin cửa hàng, phần cập nhật thông tin liên lạc 2. Hệ thống hiện form gồm các trường email, số điện thoại, địa chỉ,… 3. Quản lý tiến hành sửa đổi thông tin liên lạc theo nhu cầu và bấm lưu 4. Hệ thống kiểm tra tính hợp lệ của thông tin, nếu đúng sẽ lưu vào CSDL | |
| **Ngoại lệ**  4.1 Thông tin giờ làm việc không hợp lệ (Email, số điện thoại không đúng định dạng,…)  4.1.1 Hệ thống báo lỗi cho quản lý biết  4.1.2 Quay lại bước 3 | |

Bảng . Use Case Cập nhật thông tin liên lạc

* **Use Case Xem thống kê**

|  |  |
| --- | --- |
| **ID** | UC\_XTK |
| **Tên** | Xem thống kê |
| **Tác nhân chính** | Quản lý |
| **Ý nghĩa** | Quản lý muốn xem các thông tin thống kê liên quan đến tình trạng kinh doanh của cửa hàng |
| **Điều kiện kích hoạt** | Quản lý chọn lệnh xem thống kê |
| **Điều kiện trước** | Quản lý đã đăng nhập |
| **Chuỗi sự kiện chính:**   1. Quản lý chọn lệnh xem thống kê 2. Hệ thống tổng hợp các thông tin kinh doanh từ CSDL và sinh ra thống kê hiện lên màn hình | |

Bảng . Use Case Xem thống kê

* **Use Case Cập nhật thông tin liên lạc**

|  |  |
| --- | --- |
| **ID** | UC\_NSP |
| **Tên** | Nhập sản phẩm |
| **Tác nhân chính** | Quản lý |
| **Ý nghĩa** | Quản lý muốn nhập thêm lô hàng cho một sản phẩm theo nhu cầu |
| **Điều kiện kích hoạt** | Quản lý chọn lệnh nhập sản phẩm tại một sản phẩm theo nhu cầu |
| **Điều kiện trước** | Quản lý đã đăng nhập |
| **Chuỗi sự kiện chính:**   1. Quản lý chọn lệnh nhập sản phẩm tại một sản phẩm theo nhu cầu Hệ thống hiện form gồm các trường email, số điện thoại, địa chỉ,… 2. Hệ thống hiện dialog hiện các trường dữ liệu cần thiết: Ngày nhập, nhà cung cấp, số lượng nhập, với các giá mặc định ở ngày nhập và nhà cung cấp 3. Quản lý chỉnh sửa và điền thêm thông tin cần thiết theo yêu cầu và bấm xác nhận nhập 4. Hệ thống kiểm tra tính hợp lệ của thông tin, nếu đúng sẽ lưu vào CSDL | |
| **Ngoại lệ**  4.1 Thông tin giờ làm việc không hợp lệ (Thiếu số lượng, ngày nhập quá trễ / sớm,…)  4.1.1 Hệ thống báo lỗi cho quản lý biết  4.1.2 Quay lại bước 3 | |

Bảng . Use Case Xem thống kê

## Kiến trúc Webservice



Hình . Mô tả sơ lược kiến trúc Rest API

REST là viết tắt của Representational State Transfer. RESTful API là một tiêu chuẩn dùng trong việc thiết kế các API cho các ứng dụng web để quản lý các resource.

RESTful là một trong những kiểu thiết kế API được sử dụng phổ biến ngày nay để cho các ứng dụng (web, mobile…) khác nhau giao tiếp với nhau.

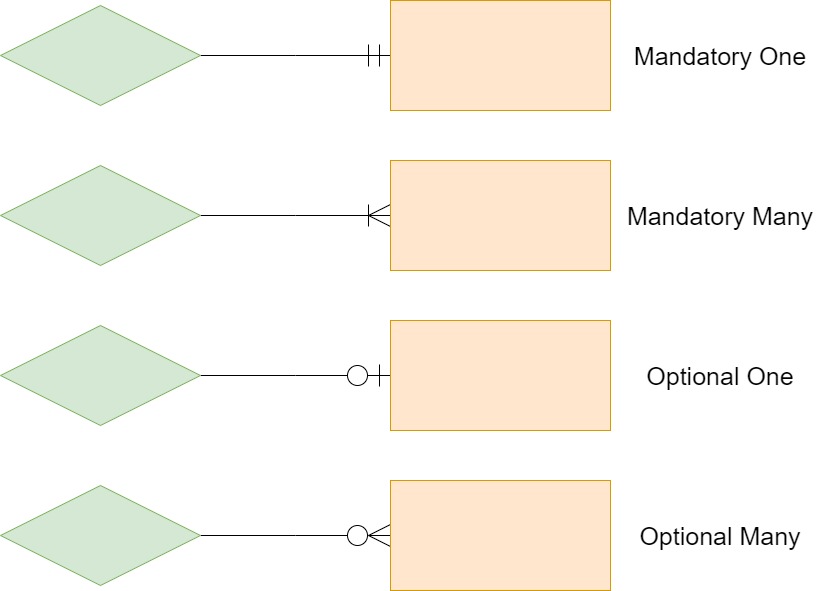
Chức năng quan trọng nhất của REST là quy định cách sử dụng các HTTP method (như GET, POST, PUT, DELETE…) và cách định dạng các URL cho ứng dụng web để quản các resource. RESTful không quy định logic code ứng dụng và không giới hạn bởi ngôn ngữ lập trình ứng dụng, bất kỳ ngôn ngữ hoặc framework nào cũng có thể sử dụng để thiết kế một RESTful API

## Thiết lập mô hình ERD (Entity Relationship Diagram)

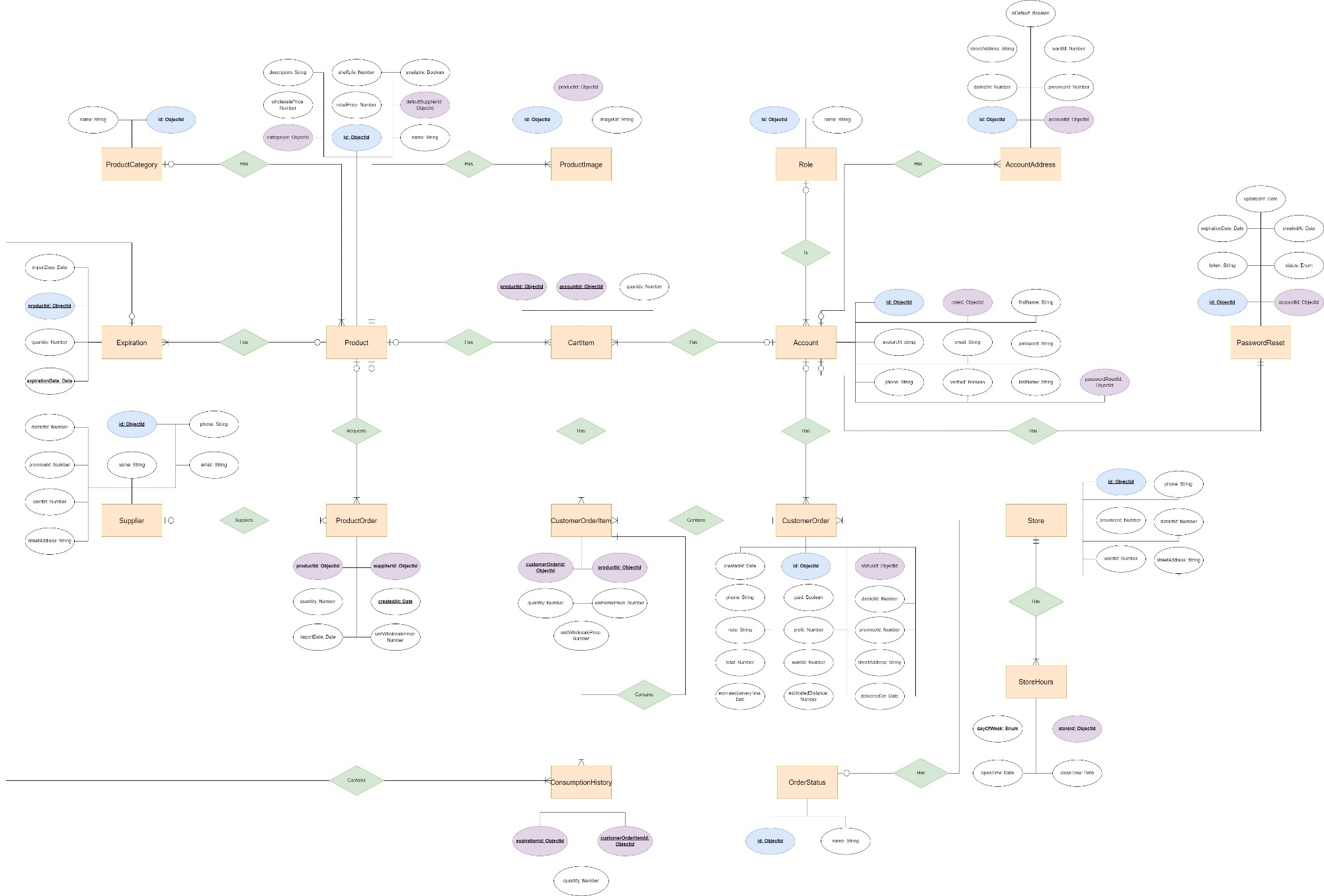
### Xác định các thực thể

1. ProductCategory (**id,** name)
2. OrderStatus (**id**, name)
3. Role (**id,** name)
4. Supplier (**id,** name, email, phone, districtId, provinceId, wardId, streetAddress)
5. Product(**id, categoryId**, name, wholesalePrice, retailPrice, description, available, shelfLife, **defaultSupplier**)
6. ProductImage(**id**, **productId,** imageUrl)
7. ProductOrder (**id**, **productId**, **supplierId**, quantity, createdAt, unitWholesalePrice, importDate)
8. Expiration (**id**, **productId**, expirationDate, quantity, importDate)
9. ConsumptionHistory (**id**, **expirationId**, **customerOrderItemId**, quantity)
10. PasswordReset (**id**, **accountId**, token, status, expirationDate, createdAt, updatedAt)
11. Account (**id,** **passwordResetId, roleId,** firstName, lastName, avatarUrl, email, phone, verified, password)
12. AccountAddress(**id**, **accountId**, districtId, provinceId, wardId, streetAddress, isDefault)
13. CustomerOrder (**id, orderStatusId**, createdAt, phone, provinceId, districtId, wardId, streetAddress, paid, note, total, estimatedDeliveryTime, estimatedDistance, deliveredOn )
14. CustomerOrderItem (**id**, **customerOrderId, productId**, quantity, unitWholesalePrice, unitRetailPrice,)
15. CartItem (**id**, **productId, accountId,** quantity)
16. Store (**id**, phone, districtId, provinceId, wardId, streetAddress)
17. StoreHours (**id**, **storeId**, dayOfWeek, openTime, closeTime**)**

### Mô hình thực thể kết hợp ERD (Entity Relationship Diagram)



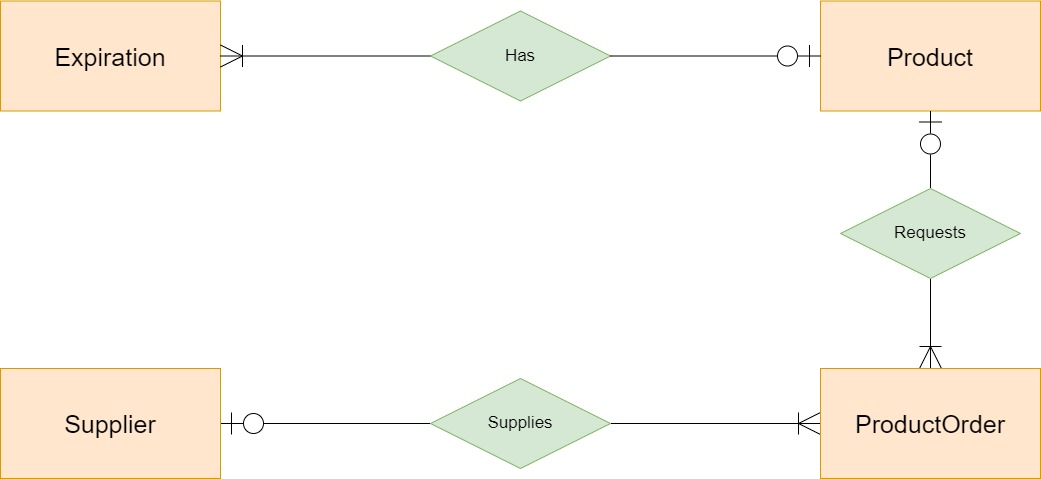
Hình . Bảng chú thích các kí hiệu trong ERD



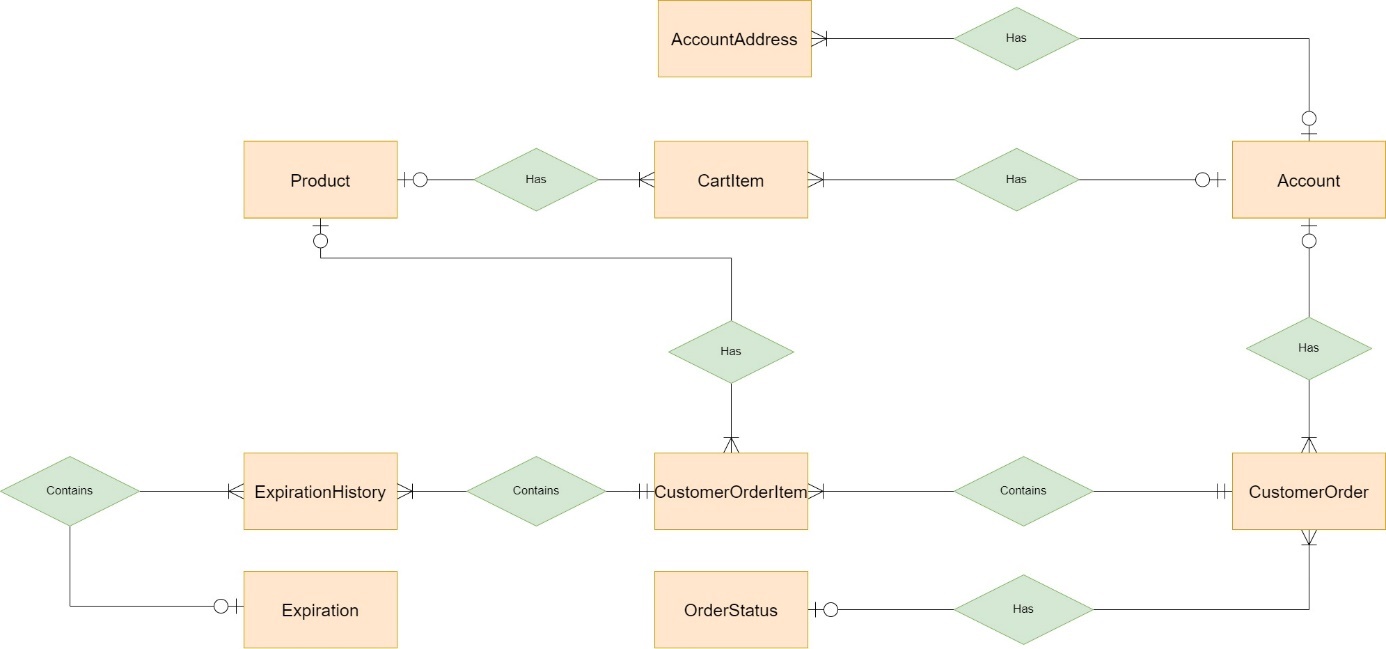
Hình . ERD Tổng quát

[*https://drive.google.com/file/d/1PpJy3ZAeMl\_LBsJ\_TF7ypeJr3A712uiN/view*](https://drive.google.com/file/d/1PpJy3ZAeMl_LBsJ_TF7ypeJr3A712uiN/view)

*ShortURL:* [*https://shorturl.at/cmnyC*](https://shorturl.at/cmnyC)



Hình . ERD Phân rã nghiệp vụ xuất nhập, quản lý kho



Hình . ERD Phân rã nghiệp vụ bán hàng

### Từ điển dữ liệu

**Lưu ý:**

* Do database triển khai bằng MongoDB trên nền của thư viện Mongoose (Để tăng tính chặt chẽ cho dữ liệu) nên kiểu của dữ liệu sẽ khác với kiểu thường thấy ở các database sử dụng SQL.
* **provinceId, districtId, wardId** là các mã do Nhà nước định nghĩa (VD: Hà Nội có mã là 1, TP HCM có mã là 79,…) và được cung cấp bởi API bên ngoài (<https://provinces.open-api.vn/>) để tránh phải lưu dữ liệu về địa chỉ vào trong database, vì chúng rất nặng và phải tự cập nhật thủ công nếu như có thay đổi (VD: Quận Thủ Đức ⇒ Thành phố Thủ Đức)

1. ProductCategory (**id,** name)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tên** | **Kiểu** | **Mô tả** | **Khoá** | **Ràng buộc** |
| id | ObjectId | Mã danh mục | * Khoá chính | * Không null * Độc nhất |
| name | String | Tên danh mục |  | * Độc nhất |

Bảng . Mô tả ProductCategory

1. OrderStatus (**id**, name)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tên** | **Kiểu** | **Mô tả** | **Khoá** | **Ràng buộc** |
| id | ObjectId | Mã trạng thái | * Khoá chính | * Không null * Độc nhất |
| name | String | Tên trạng thái |  | * Không null * Độc nhất |

Bảng . Mô tả OrderStatus

1. Role (**id,** name)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tên** | **Kiểu** | **Mô tả** | **Khoá** | **Ràng buộc** |
| id | ObjectId | Mã quyền | * Khoá chính | * Không null * Độc nhất |
| name | String | Tên quyền |  | * Không null * Độc nhất |

Bảng . Mô tả Role

1. Supplier (**id,** name, email, phone, districtId, provinceId, wardId, streetAddress)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tên** | **Kiểu** | **Mô tả** | **Khoá** | **Ràng buộc** |
| id | ObjectId | Mã nhà cung cấp | * Khoá chính | * Không null * Độc nhất |
| name | String | Tên nhà cung cấp |  | * Không null * Độc nhất |
| email | String | Email nhà cung cấp |  | * Không null * Độc nhất |
| phone | String | SDT nhà cung cấp |  | * Không null * Độc nhất |
| provinceId | Number | Mã Thành phố / Tỉnh |  | * Không null |
| districtId | Number | Mã Quận / Huyện |  | * Không null |
| wardId | Number | Mã Phường / Xã |  | * Không null |

Bảng . Mô tả Supplier

1. Product(**id**, **categoryId**, name, wholesalePrice, retailPrice, description, available, shelfLife, **defaultSupplier**)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tên** | **Kiểu** | **Mô tả** | **Khoá** | **Ràng buộc** |
| id | ObjectId | Mã sản phẩm | * Khoá chính | * Không null * Độc nhất |
| categoryId | ObjectId | Danh mục mà sản phẩm thuộc về | * Khoá ngoại | * Không null |
| name | String | Tên sản phẩm |  | * Không null * Độc nhất |
| wholeSalePrice | Number | Giá nhập sản phẩm (Giá sỉ) |  | * Không null * Lớn hơn 0 |
| retailPrice | Number | Giá bán sản phẩm (Giá lẻ) |  | * Không null * Lớn hơn giá nhập |
| description | String | Mô tả sản phẩm |  | * Tối đa 500 ký tự |
| shelfLife | Number | Số ngày có thể sử dụng của sản phẩm, khoảng cách giữa importDate và expirationDate |  | * Không null * Lớn hơn 0 |
| available | Boolean | Sản phẩm còn kinh doanh hay không (Khác với sản phẩm còn hàng hay không) |  | * Default: true |
| defaultSupplier | ObjectId | Nhà cung cấp sản phẩm này ở lần gần nhất, dùng làm giá trị mặc định khi nhập sản phẩm nếu không có thay đổi từ user |  |  |

Bảng . Mô tả Product

1. ProductImage(**id, productId,** imageUrl)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tên** | **Kiểu** | **Mô tả** | **Khoá** | **Ràng buộc** |
| id | ObjectId | Mã ảnh sản phẩm | * Khoá chính | * Không null * Độc nhất |
| productId | ObjectId | Sản phẩm mà ảnh này thuộc về | * Khoá ngoại | * Không null |
| imageUrl | String | Đường dẫn tới ảnh sản phẩm |  | * Không null * Độc nhất |

Bảng . ProductImage

1. ProductOrder (**id**, **productId**, **supplierId**, quantity, createdAt, unitWholesalePrice, importDate)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tên** | **Kiểu** | **Mô tả** | **Khoá** | **Ràng buộc** |
| id | ObjectId | Mã đơn nhập sản phẩm | * Khoá chính | * Không null * Độc nhất |
| productId | ObjectId | Mã sản phẩm nhập | * Khoá ngoại | * Không null |
| supplierId | ObjectId | Mã nhà cung cấp sản phẩm tại đơn nhập này | * Khoá ngoại | * Không null |
| createdAt | Date | Ngày tạo đơn nhập |  | * Không null |
| unitWholesalePrice | Number | Giá nhập ngay tại thời điểm nhập, trường này tồn tại để tránh sự thay đổi của giá nhập trong tương lại ảnh hưởng đến lịch sử |  | * Không null * Lớn hơn 0 |
| importDate | Number | Ngày nhập sản phẩm |  | * Không null * Không lớn hơn ngày hiện tại * Không nhỏ hơn thời điểm tại hiện quá 3 ngày |

Bảng . Mô tả ProductOrder

1. Expiration (**id**, **productId**, expirationDate, quantity, importDate)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tên** | **Kiểu** | **Mô tả** | **Khoá** | **Ràng buộc** |
| id | ObjectId | Mã tồn kho | * Khoá chính | * Không null * Độc nhất |
| productId | ObjectId | Mã sản phẩm tồn kho | * Khoá ngoại | * Không null |
| expirationDate | Date | Ngày hết hạn của sản phẩm tồn kho này |  | * Không null |
| quantity | Number | Số lượng tồn kho hiện thời của lô sản phẩm này |  | * Không null * Lớn hơn 0 |
| importDate | Date | Ngày nhập lô sản phẩm này, có giá trị tương đương với importDate ở ProductOrder, được lưu ở 2 nơi để tạo liên kết, làm các nghiệp vụ về nhập kho |  | * Không null |

Bảng . Mô tả Expiration

1. ConsumptionHistory (**id**, **expirationId**, **customerOrderItemId**, quantity)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tên** | **Kiểu** | **Mô tả** | **Khoá** | **Ràng buộc** |
| id | ObjectId | Mã lịch sử xuất hàng | * Khoá chính | * Không null * Độc nhất |
| expirationId | ObjectId | Mã lô hàng xuất sản phẩm | * Khoá ngoại | * Không null |
| customerOrderItemId | ObjectId | Mã chi tiết đơn hàng | * Khoá Ngoại | * Không null |
| quantity | Number | Số lượng sản phẩm xuất của lô hàng xuất |  | * Không null * Lớn hơn 0 |

Bảng . Mô tả ConsumptionHistory

1. PasswordReset (**id**, **accountId**, token, status, expirationDate, createdAt, updatedAt)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tên** | **Kiểu** | **Mô tả** | **Khoá** | **Ràng buộc** |
| id | ObjectId | Mã PasswordReset | * Khoá chính | * Không null * Độc nhất |
| accountId | ObjectId | Mã tài khoản thuộc PasswordReset này | * Khoá ngoại | * Không null * Độc nhất |
| token | String | Token để xác thực việc khôi phục mật khẩu |  | * Không null * Độc nhất |
| status | Enum | Trạng thái của việc khôi phục mật khẩu |  | * Không null * Thuộc 1 trong 3 giá trị: PENDING, RESOLVED, EXPIRED |
| expirationDate | Date | Thời điểm hết hạn của token |  | * Không null |
| createdAt | Date | Thời điểm gửi yêu cầu khôi phục mật khẩu lần đầu tiên |  | * Không null |
| updatedAt | Date | Thời điểm gửi yêu cầu khôi phục mật khẩu gần nhất |  | * Không null |

Bảng . Mô tả PasswordReset

1. Account (**id,** **passwordResetId, roleId,** firstName, lastName, avatarUrl, email, phone, verified, password)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tên** | **Kiểu** | **Mô tả** | **Khoá** | **Ràng buộc** |
| id | ObjectId | Mã tài khoản | * Khoá chính | * Không null * Độc nhất |
| passwordResetId | ObjectId | Mã PasswordReset | * Khoá ngoại | * Không null * Độc nhất |
| roleId | ObjectId | Mã quyền | * Khoá ngoại | * Không null |
| firstName | String | Tên chủ tài khoản |  | * Không null |
| lastName | String | Họ và tên lót chủ tài khoản |  | * Không null |
| avatarUrl | String | Đường dẫn ảnh đại diện tài khoản |  | * Default: ảnh từ ui-avatars.com |
| email | String | Email tài khoản, dùng để đăng nhập |  | * Không null * Độc nhất |
| phone | String | Số điện thoại, dùng làm giá trị mặc định khi đặt đơn hàng |  | * Không null * Độc nhất |
| verified | Boolean | Tài khoản đã được xác thực hay chưa |  | * Default: false |
| password | String | Mật khẩu đã được mã hoá của tài khoản |  | * Không null |

Bảng . Mô tả Account

1. AccountAddress (**id**, **accountId**, districtId, provinceId, wardId, streetAddress, isDefault)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tên** | **Kiểu** | **Mô tả** | **Khoá** | **Ràng buộc** |
| id | ObjectId | Mã địa chỉ tài khoản | * Khoá chính | * Không null * Độc nhất |
| accountId | ObjectId | Mã tài khoản mà địa chỉ thuộc về | * Khoá ngoại | * Không null |
| provinceId | Number | Mã Thành Phố / Tỉnh |  | * Không null |
| districtId | Number | Mã Quận / Huyện |  | * Không null |
| wardId | Number | Mã Phường / Xã |  | * Không null |
| streetAddress | String | Địa chị cụ thể (Tên đường, số nhà, tầng,…) |  | * Không null |
| isDefault | Boolean | Có phải địa chỉ mặc định dùng để điền vào thông tin hàng hay không |  | * Default: true nếu là địa chỉ đầu tiên, false với các trường hợp còn lại |

Bảng . Mô tả AccountAddress

1. CustomerOrder (**id**, **orderStatusId**, createdAt, phone, provinceId, districtId, wardId, streetAddress, paid, note, total, estimatedDeliveryTime, estimatedDistance, deliveredOn )

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tên** | **Kiểu** | **Mô tả** | **Khoá** | **Ràng buộc** |
| id | ObjectId | Mã đơn hàng | * Khoá chính | * Không null * Độc nhất |
| orderStatusId | ObjectId | Mã trạng thái đơn | * Khoá ngoại | * Không null |
| createdAt | Date | Ngày đặt hàng |  | * Không null |
| phone | String | Số điện thoại liên lạc khi giao hàng, giá trị mặc định lấy từ Account |  | * Không null |
| provinceId | Number | Mã Thành phố / Tỉnh |  | * Không null |
| districtId | Number | Mã Quận / Huyện |  | * Không null |
| wardId | Number | Mã Phường / Xã |  | * Không null |
| streetAddress | String | Địa chị cụ thể (Tên đường, số nhà, tầng,…) |  | * Không null |
| paid | Boolean | Đơn hàng đã được thanh toán online hay chưa |  | * Không null |
| note | String | Ghi chú đơn hàng |  | * Tối đa 500 ký tự |
| total | Number | Tổng tiền đơn hàng, tính bằng tổng của các retailPrice x quantity trong từng CustomerOrderItem |  | * Không null |
| estimatedDeliveryTime | Date | Thời gian giao hàng dự kiến, sử dụng Google Maps API để tính toán |  | * Không null |
| estimatedDistance | Number | Quãng đường giao hàng ngắn nhất dự kiến, đơn vị km, sử dụng Google Maps API để tính toán |  | * Không null |
| deliveredOn | Date | Thời điểm đã giao hàng thành công |  |  |

Bảng . Mô tả CustomerOrder

1. CustomerOrderItem (**id**, **customerOrderId, productId**, quantity, unitWholesalePrice , unitRetailPrice)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tên** | **Kiểu** | **Mô tả** | **Khoá** | **Ràng buộc** |
| id | ObjectId | Mã chi tiết đơn hàng | * Khoá chính | * Không null * Độc nhất |
| customerOrderId | ObjectId | Mã đơn hàng mà chi tiết này thuộc về | * Khoá ngoại | * Không null |
| productId | ObjectId | Mã sản phẩm của chi tiết đơn hàng này | * Khoá ngoại | * Không null |
| unitWholesalePrice | Number | Giá nhập (Giá sỉ) dùng chung với unitRetailPrice để tính lợi nhuận (Profit). Trường này tồn tại để tránh sự thay đổi của giá nhập trong tương lại ảnh hưởng đến lịch sử |  | * Không null * Lớn hơn 0 |
| unitRetailPrice | Number | Giá bán (Giá lẻ) dùng chung với unitWholesalePrice để tính lợi nhuận (Profit). Trường này tồn tại để tránh sự thay đổi của giá nhập trong tương lại ảnh hưởng đến lịch sử |  | * Không null * Lớn hơn giá nhập |

Bảng . Mô tả CustomerOrderItem

1. CartItem (**id**, **productId, accountId,** quantity)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tên** | **Kiểu** | **Mô tả** | **Khoá** | **Ràng buộc** |
| id | ObjectId | Mã chi tiết giỏ hàng | * Khoá chính | * Không null * Độc nhất |
| productId | ObjectId | Mã sản phẩm của chi tiết giỏ hàng này | * Khoá ngoại | * Không null |
| accountId | ObjectId | Mã tài khoản sở hữu chi tiết giỏ hàng này | * Khoá ngoại | * Không null |
| quantity | Number | Số lượng sản phẩm của chi tiết giỏ hàng |  | * Không null * Lớn hơn 0 |

Bảng . Mô tả CartItem

1. Store (**id**, phone, provinceId, districtId, wardId, streetAddress)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tên** | **Kiểu** | **Mô tả** | **Khoá** | **Ràng buộc** |
| id | ObjectId | Mã cửa hàng | * Khoá chính | * Không null * Độc nhất |
| phone | String | Số điện thoại cửa hàng |  | * Không null * Độc nhất |
| provinceId | Number | Mã Thành phố / Tỉnh |  | * Không null |
| districtId | Number | Mã Quận / Huyện |  | * Không null |
| wardId | Number | Mã Phường / Xã |  | * Không null |
| streetAddress | String | Địa chị cụ thể (Tên đường, số nhà, tầng,…) |  | * Không null |

Bảng . Mô tả Store

1. StoreHours (**id**, **storeId,** dayOfWeek, openTime, closeTime)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tên** | **Kiểu** | **Mô tả** | **Khoá** | **Ràng buộc** |
| id | ObjectId | Mã giờ làm việc | * Khoá chính | * Không null * Độc nhất |
| storeId | Object | Mã cửa hàng mà giở làm việc này thuộc về |  | * Không null |
| dayOfWeek | Enum | Cho biết giờ làm việc này thuộc về thứ nào trong tuần |  | * Không null * Thuộc 1 trong 7 giá trị: MONDAY, TUESDAY, WEDNESDAY, THURSDAY, FRIDAY, SATURDAY, SUNDAY |
| openTime | Date | Giờ mở cửa |  | * Không null |
| closeTime | Date | Giờ đóng cửa |  | * Không null * Trễ hơn giờ mở cửa |

Bảng . Mô tả StoreHours

# CHƯƠNG TRÌNH

## Giao diện Client

## Giao diện Admin

# KẾT LUẬN

## Kết quả đạt được và hạn chế

## Hạn chế và hướng phát triển

# PHỤ LỤC

## Danh mục tài liệu tham khảo

1. [Visual Studio Code là gì? Tính năng nổi bật của Visual Studio Code - Technology Diver](https://cuongquach.com/visual-studio-code-la-gi.html)
2. [Visual Studio Code là gì? Tính năng của Visual Studio Code - Fptshop.com.vn](https://fptshop.com.vn/tin-tuc/danh-gia/visual-studio-code-la-gi-cac-tinh-nang-noi-bat-cua-visual-studio-code-146213)
3. [Tổng quan về MongoDB - viblo.asia](https://viblo.asia/p/tong-quan-ve-mongodb-EoDkQoxqGbV)
4. [Single page application là gì? Lưu ý khi triển khai ứng dụng web với SPA | bizfly.vn](https://bizfly.vn/techblog/single-page-application-la-gi.html)
5. [Giới thiệu về ReactJS - Phần I (Các khái niệm cơ bản) - viblo.asia](https://viblo.asia/p/gioi-thieu-ve-reactjs-phan-i-cac-khai-niem-co-ban-V3m5WzjblO7)
6. [RESTful API là gì? Cách thiết kế RESTful API | TopDev](https://topdev.vn/blog/restful-api-la-gi/)