## **Mục tiêu**

Tìm hiểu các công nghệ: ASP.Net, SQL Server

Tìm hiểu cấu trúc chung của website bán hàng

Tìm hiểu các chức năng hệ thống quản trị website bán hàng

Đưa ra bản phân tích, thiết kế hệ thống website bán cafe.

Xây dựng được website bán cafe, hệ thống quản trị bằng cách sử dụng các công nghệ lập trình web và cơ sở dữ liệu: ASP.Net, SQL Server

## **Giới hạn và phạm vi của đồ án**

Trong phạm vi đề tài này em sẽ nghiên cứu các vấn đề:

Nghiên cứu, khảo sát, thu thập dữ liệu về quy trình mua bán các sản phẩm của một cửa hàng bán cafe, đi sâu vào nghiên cứu và phân tích một hệ thống thông tin để xây dựng website.

Hệ thống được xây dựng bằng sử dụng các công nghệ lập trình web và cơ sở dữ liệu: ASP.Net, SQL Server.

Kiểm thử và phát triển hệ thống.

## **Kết quả dự kiến đạt được**

Hệ thống website bán cafe khi hoàn thành dự kiến đạt được các kết quả sau:

Cấu trúc website bán hàng.

Phân tích, thiết kế chức năng của hệ thống như: đăng nhập, đăng ký tài khoản, xem thông tin sản phẩm, tìm kiếm sản phẩm, đặt hàng...

Thiết kế cơ sở dữ liệu.

Xây dựng một số module demo kết quả.

Cho phép quản trị thông tin danh mục, sản phẩm, đơn đặt hàng…

Cho phép người dùng tìm kiếm sản phẩm; xem chi tiết sản phẩm, đăng ký tài khoản, lựa chọn mua hàng...

Cơ sở dữ liệu hàng hóa được quản lý trên SQL Server.

Cho phép thiết lập cấu trúc website động.

Xây dựng kịch bản và thực hiện kiểm thử, báo cáo kiểm thử.

Có báo cáo tồn kho (sản phẩm, loại sản phẩm), báo cáo bán hàng (theo doanh thu, ngày, sản phẩm, loại sản phẩm).

Người sử dụng hệ thống: Chủ cửa hàng, Nhân viên bán hàng, Quản trị hệ thống, khách hàng.

Hoàn thành báo cáo chi tiết đồ án tốt nghiệp.

## **Công nghệ sử dụng**

### **ASP.NET**

ASP.Net là một dạng framework, nói một cách dễ hiểu chính là một **bộ khung ứng dụng we**b. ASP Net được hoàn thiện bởi Microsoft vào 2012 và là một giải pháp dùng để hỗ trợ quá trình sản xuất các loại website và nội dung động.

Mẫu kiến trúc Model – View – Controller được sử dụng nhằm chi ứng dụng thành ba thành phần chính: model, view và controller. Nền tảng ASP.NET MVC giúp cho chúng ta có thể tạo được các ứng dụng web áp dụng mô hình MVC thay vì tạo ứng dụng theo mẫu ASP.NET Web Forsm. Nền tảng ASP.NET MVC có đặc điểm nổi bật là nhẹ, dễ kiểm thử phần giao diện (so với ứng dụng Web Forms), tích hợp các tính năng có sẵn của ASP.NET. Nền tảng ASP.NET MVC được định nghĩa trong namespace System.Web.Mvc và là một phần của name space System.Web.

MVC là một mẫu thiết kế (design pattern) chuẩn mà nhiều lập trình viên đã quen thuộc. Một số loại ứng dụng web sẽ thích hợp với kiến trúc MVC. Một số khác vẫn thích hợp với ASP.NET Web Forms và cơ chế postbacks. Đôi khi có những ứng dụng kết hợp cả hai kiến trúc trên.

Models: Các đối tượng Models là một phần của ứng dụng, các đối tượng này thiết lập logic của phần dữ liệu của ứng dụng. Thông thường, các đối tượng model lấy và lưu trạng thái của model trong CSDL.

Views: Views là các thành phần dùng để hiển thị giao diện người dùng (UI). Thông thường, view được tạo dựa vào thông tin dữ liệu model.

Controllers: Controller là các thành phần dùng để quản lý tương tác người dùng, làm việc với model và chọn view để hiển thị giao diện người dùng. Trong một ứng dụng MVC, view chỉ được dùng để hiển thị thông tin, controller chịu trách nhiệm quản lý và đáp trả nội dung người dùng nhập và tương tác với người dùng.

### **SQL server**

SQL server chính là cụm từ viết tắt của Structure Query Language được sử dụng nhiều trong các lĩnh vực bởi chức năng quản lý dữ liệu. Các ngôn ngữ cấp cao như: [Visual C](https://www.visualstudio.com/vs/cplusplus/), [Oracle](https://www.oracle.com/database/index.html), Visual Basic,… đều có trình hỗ trợ là SQL. Những ứng dụng khi chạy phải sử dụng SQL khi người dùng truy cập tới cơ sở dữ liệu thì không cần sử dụng trực tiếp SQL.

SQL Server chính là một **hệ quản trị dữ liệu** quan hệ sử dụng câu lệnh SQL để trao đổi dữ liệu giữa máy cài SQL Server và máy Client. Một Relational Database Management System – RDBMS gồm có: databases, datase engine và các chương trình ứng dụng dùng để quản lý các bộ phận trong RDBMS và những dữ liệu khác.

SQL được sử dụng trong các dịch vụ [thiết kế web đẹp](https://mona.media/thiet-ke-website-tai-hcm/) với chức năng giao tiếp với người dùng với các vai trò sau:

* SQL là một ngôn ngữ đòi hỏi có tính tương tác cao: Người dùng có thể dễ dàng trao đổi với các tiện ích thông qua các câu lệnh của SQL đến cơ sở dữ liệu và nhận kết quả từ cơ sở dữ liệu.
* SQL là một ngôn ngữ lập trình cơ sở dữ liệu: Các lập trình viên có thể xây dựng các chương trình ứng dụng giao tiếp với cơ sở dữ liệu bằng cách nhúng các câu lệnh SQL vào trong ngôn ngữ lập trình.
* SQL là một ngôn ngữ lập trình quản trị cơ sở dữ liệu: Người quản trị cơ sở dữ liệu có thề quản lý, định nghĩa và điều khiển truy cập cơ sở dữ liệu thông qua SQL.
* SQL là một ngôn ngữ lập trình cho các hệ thống chủ khách: SQL được sử dụng như là một công cụ giao tiếp với các trình ứng dụng trong hệ thống cơ sở dữ liệu khách chủ.
* SQL là ngôn ngữ truy cập dữ liệu trên Internet: SQL được sử dụng với vai trò tương tác với dữ liệu trong hầu hết các máy chủ web và máy chủ Internet.
* SQL là ngôn ngữ cơ sở dữ liệu phân tán: Với vai trò giao tiếp với các hệ thống trên mạng, gửi và nhận các yêu cầu truy xuất dữ liệu với nhau.

Nhìn chung SQL Server là một hệ quản trị cơ sở dữ liệu quan hệ được sử dụng trong các chức năng tương tác giữa người dùng và dữ liệu. Chính vì thế, nó được sử dụng trong các dịch vụ thiết kế web rẻ và chuyên nghiệp.

### **Ajax**

* Ajax là chữ viết tắt của cụm từ Asynchronous Javascript and XML. Ajax là phương thức trao đổi dữ liệu với máy chủ và cập nhật một hay nhiều phần của trang web, hoàn toàn không reload lại toàn bộ trang.
* Ajax được viết bằng Javascript chạy trên client, tức là mỗi browser sẽ chạy độc lập hoàn toàn không ảnh hưởng lẫn nhau. Về mặt kỹ thuật, nó đề cập đến việc sử dụng các đối tượng XmlHttpRequest để tương tác với một máy chủ web thông qua Javascript.

Vai trò của Ajax:

* AJAX là một trong những công cụ giúp chúng ta đem lại cho người dùng trải nghiệm tốt hơn. Khi cần một thay đổi nhỏ thì sẽ không cần load lại cả trang web, làm trang web phải tải lại nhiều thứ không cần thiết.
* Ajax được sử dụng để thực hiện một callback. Được dùng để thực hiện việc truy xuất dữ liệu hoặc lưu trữ dữ liệu mà không cần phải reload lại toàn bộ trang web. Với những server nhỏ thì việc này cũng tiết kiệm được băng thông cho chúng ta hơn.
* Cần gì thì chỉ gửi dữ liệu phần đó, load lại 1 phần nhỏ để cập nhật thông tin chứ không load cả trang. Bằng cách này thì có thể giảm thiểu được tốc độ tải trang giúp người dùng có trải nghiệm tốt hơn.
* Trang web bạn tạo ra cũng sẽ đa dạng và động hơn.

## **Yêu cầu đề tài**

Website bán cafe cần đáp ứng được những yêu cầu sau:

* Giao diện phải dễ sử dụng và có tính thẩm mỹ cao.
* Cho phép khách hàng đăng kí tài khoản và phải đảm bảo bảo mật thông
* tin.
* Cho phép khách hàng tìm kiếm nhanh chóng, chính xác.
* Khách hàng có thể thêm sản phẩm vào giỏ hàng và tiến hành đặt hàng.
* Khách hàng có thể thay đổi thông tin cá nhân, thông tin giao hàng khi đã

đăng nhập.

* Người quản lý có thể theo dõi các công việc sau:
  + Thêm mới, cập thông tin chi tiết cho sản phẩm, có thể xoá sản phẩm, ẩn sản phẩm khỏi người dùng nếu nó không còn được bán.
  + Có thể tạo tài khoản cho nhân viên, cấp một số quyền xác định cho nhân viên.
  + Hiển thị thông tin chi tiết sản phẩm mà khách đã đặt hàng.
  + Quản lý các đơn đặt hàng: có thể cập nhật đơn hàng.
  + Hiển thị chi tiết đơn đặt hàng của khách hàng.

## **Hoạt động của hệ thống**

### **Hoạt động bán hàng**

* Khách hàng có thể xem, tìm kiếm các sản phẩm có trong giỏ hàng.
* Người dùng có thể tạo đăng ký, đăng nhập và sửa thông tin tài khoản.
* Khách hàng có thể thêm sản phẩm vào giỏ hàng và tiến hành đặt hàng. Khi khách hàng đặt mua hàng, nhân viên xác nhận đơn hàng. Sau đó chuyển hàng cho bên vận chuyển. Khi khách hàng nhận được đơn hàng sẽ xác nhận đã nhận hàng và có thể đánh giá sản phẩm.

### **Báo cáo, thống kê**

Khi chủ cửa hàng yêu cầu thống kê doanh số bán hàng cuối ngày hoặc cuối tháng thì hệ thống sẽ tổng hợp các hóa đơn của ngày hoặc tháng và đưa ra kết quả tổng hợp.

### **Cập nhật thông tin hệ thống**

Nhân viên có thể thực hiện:

* Thêm, sửa, xóa, thông tin:
  + Các danh mục sản phẩm.
  + Các sản phẩm.
  + Các tin tức
  + Quảng cáo.
  + Thông tin liên hệ.
* Quản lý đơn hàng: cập nhật đơn hàng.

Người quản lí có thể theo dõi các hoạt động mà nhân viên thực hiện. Đồng thời còn có thể thực hiện:

* Tạo tài khoản nhân viên, chủ cửa hàng, cấp một số quyền xác định.
* Sửa, xóa thông tin các nhân viên, chủ cửa hàng.

## **Các tác nhân của hệ thống**

* Tác nhân khách hàng: là những người khách hàng ghé thăm, tìm kiếm và đặt hàng.
* Tác nhân nhân viên của cửa hàng : là người tham gia quản lý hệ thống với các hoạt động liên quan đến bán hàng.
* Tác nhân chủ cửa hàng: là người có thể xem các báo cáo thống kê bán hàng, báo cáo kho, quản lý nhân viên.

## **Các yêu cầu chức năng**

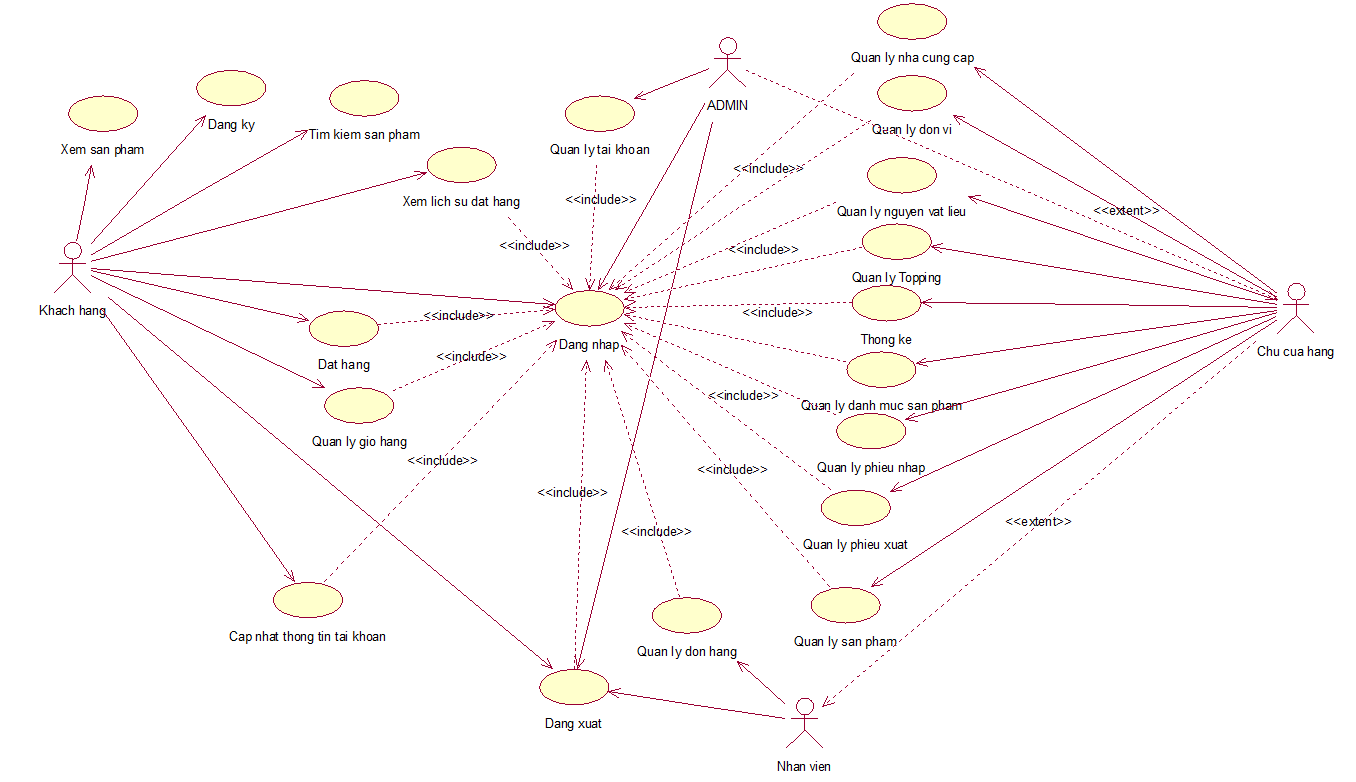
* Xem sản phẩm
* Tìm kiếm sản phẩm
* Đăng ký
* Đăng nhập
* Xem lịch sử đơn hàng
* Cập nhật thông tin tài khoản
* Đặt hàng
* Quản lý giỏ hàng
* Quản lý đơn hàng
* Quản lý danh mục sản phẩm
* Quản lý sản phẩm
* Quản lý tài khoản
* Quản lý đơn vị
* Quản lý nguyên vật liệu
* Quản lý nhà cung cấp
* Quản lý bài viết
* Quản lý quảng cáo
* Thống kê

## **Yêu cầu phi chức năng**

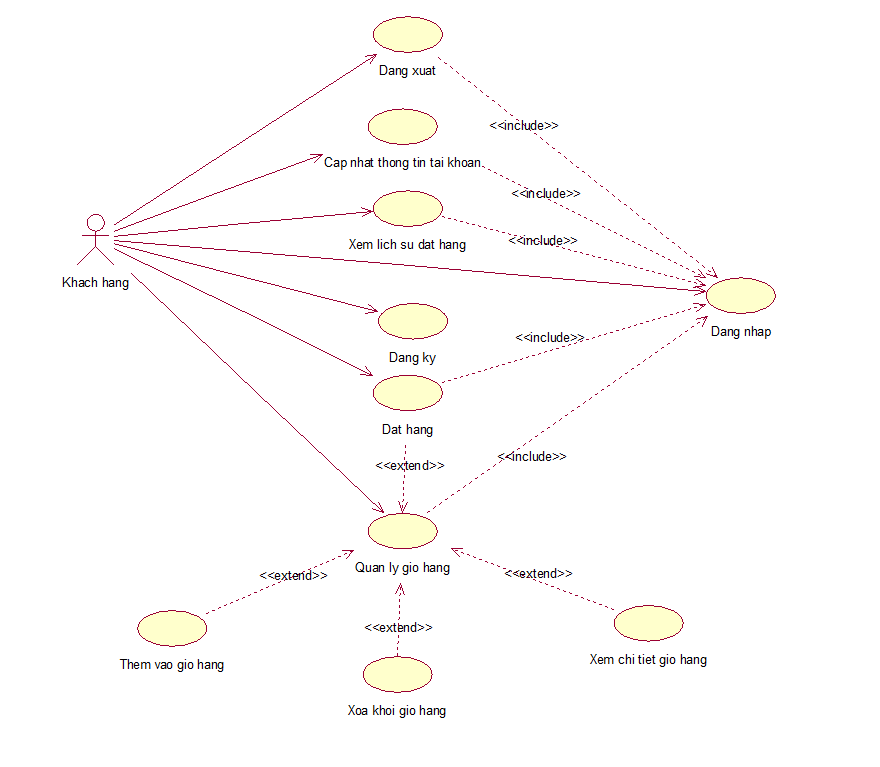
* Hệ thống phải đảm bảo uptime tối thiểu 99.9%.
* Giao diện phải dễ sử dụng và tối ưu cho trải nghiệm người dùng.
* Dữ liệu khách hàng phải được mã hóa và bảo mật cao.

## **Biểu đồ use case**

### **Biểu đồ use case tổng quát**



### **Biểu đồ phân rã use case**



Biểu đồ use case của khách hàng

## Mô tả chi tiết các use case

### 2.7.1. Mô tả use case Đăng ký

|  |
| --- |
| 1.  Tên Use Case  - Đăng ký  2.  Mô tả vắn tắt  - Use case này cho phép người dùng đăng đăng ký tài khoản.  3.  Luồng các sự kiện  3.1.  Luồng cơ bản  1.    Use case bắt đầu khi người dùng click vào nút “Đăng ký” trên thanh menu. Hệ thống sẽ hiển thị trang đăng ký.  2.   Người dùng nhập Họ tên, Email, Mật khẩu, Xác nhận mật khẩu sau đó click vào nút “Đăng ký”. Hệ thống kiểm tra địa chỉ email và mật khẩu ở bảng Acccount trong cơ sở dữ liệu sau đó hiển thị giao diện trang chủ. Use case kết thúc.  3.2. Các luồng rẽ nhánh  1.  Sau khi người dùng click vào nút “Sign in”, hệ thống kiểm tra địa chỉ email trong bảng Account của cơ sở dữ liệu.  2 .  Nếu địa chỉ email đã tồn tại trong cơ sở dữ liệu, hệ thống sẽ hiển thị thông báo lỗi: "Tài khoản đã tồn tại".Use case kết thúc.  4.  Các yêu cầu đặc biệt       - Không có   5.  Tiền điều kiện   * Không có    6.  Hậu điều kiện   * Không có    7.  Điều kiện mở rộng   * Không có |

### 2.7.2. Mô tả use case Đăng nhập

|  |
| --- |
| 1.  Tên Use Case  - Đăng nhập  2.  Mô tả vắn tắt  - Use case này cho phép người dùng đăng nhập vào hệ thống.  3.  Luồng các sự kiện  3.1.  Luồng cơ bản  1.    Use case bắt đầu khi người dùng click vào nút “Đăng nhập” trên thanh menu. Hệ thống sẽ hiển thị form đăng nhập.  2.   Người dùng nhập Email, Mật khẩu sau đó click vào nút “Đăng nhập”. Hệ thống kiểm tra địa chỉ email và mật khẩu ở bảng Acccount trong cơ sở dữ liệu sau đó hiển thị giao diện trang chủ. Use case kết thúc.  3.2.  Các luồng rẽ nhánh     1.  Sau khi người dùng click vào nút “Đăng nhập”, hệ thống kiểm tra địa chỉ email và mật khẩu trong bảng Account của cơ sở dữ liệu.     2 .  Nếu địa chỉ email hoặc mật khẩu không khớp với dữ liệu trong cơ sở dữ liệu, hệ thống sẽ hiển thị thông báo lỗi: "Thông tin đăng nhập không hợp lệ".Use case kết thúc.   4.  Các yêu cầu đặc biệt     - Không có   5.  Tiền điều kiện  - Đã có tài khoản trước đó   6.  Hậu điều kiện  - Không có   7.  Điều kiện mở rộng    - Không có |

### 2.7.3. Mô tả use case Quản lý Nguyên vật liệu

|  |
| --- |
| 1.    Tên use case  - Quản lý nguyên liệu.  2.    Mô tả vắn tắt  - Use case này cho phép người quản trị thực hiện xem, thêm, sửa, xóa nguyên liệu trong kho.  3.    Luồng sự kiện  3.1. Luồng cơ bản  1) Xem danh sách nguyên liệu:         Use case này bắt đầu khi người quản trị kích vào nút “Quản lý nguyên liệu” trên menu quản trị. Hệ thống sẽ lấy thông tin chi tiết về danh sách nguyên liệu bao gồm thông tin về nguyên liệu: name, quantity, minquantity từ bảng Material; thông tin về đơn vị: name từ bảng Unit; thông tin về nhóm nguyên liệu: name từ bảng MaterialGroup và hiển thị giao diện quản lý nguyên liệu.  2) Thêm nguyên liệu:   1. Người quản trị kích nút “Thêm nguyên liệu”. Hệ thống sẽ hiển thị một phiếu thêm thông tin cho nguyên liệu. 2. Người quản trị nhập thông tin cho nguyên liệu bao gồm: name, quantity, minquantity từ bảng Material;  name từ bảng Unit; name từ bảng MaterialGroup và kích nút “Thêm”. Hệ thống sẽ lấy thông tin và lưu vào cơ sở dữ liệu   3) Sửa nguyên liệu:   1. Người quản trị chọn nguyên liệu cần sửa và kích sửa. Hệ thống sẽ lấy thông tin bao gồm:  name, quantity, minquantity từ bảng Material;  name từ bảng Unit; name từ bảng MaterialGroup về nguyên liệu được chọn rồi hiển thị thông tin về nguyên liệu lên màn hình. 2. Người quản trị nhập thông tin cần sửa rồi bấm “Update”. Hệ thống sẽ lấy các thông tin từ màn hình hiển thị rồi lưu vào trong cơ sở dữ liệu.   4)  Xóa nguyên liệu:   1. Người quản trị kích vào nút “Xóa” ở màn hình quản lý nguyên liệu. Hệ thống sẽ hiển thị thông báo xác nhận xóa nguyên liệu. 2. Người quản trị kích nút “Yes”. Hệ thống sẽ xóa nguyên liệu mà người quản trị đã chọn và cập nhật lại danh sách phòng lên màn hình quản lý.   3.2. Luồng rẽ nhánh  1)        Tại bước 2b hoặc 3b hoặc 4b trong luồng cơ bản nếu người quản trị kích “Hủy”. Hệ thống sẽ bỏ qua thao tác và quay trở lại màn hình hiển thị danh sách các nguyên liệu.  2)        Tại bất cứ thời điểm nào trong quá trình thực hiện Use Case nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiển thị một thông báo lỗi và Use Case kết thúc.  4.    Các yêu cầu đặc biệt  - Use case này chỉ cho phép một số vai trò như người quản trị, người chủ hệ thống thực hiện.  5.    Tiền điều kiện  - Người quản trị cần đăng nhập với vai trò quản trị hệ thống trước khi có thể thực hiện use case.  6.    Hậu điều kiện  - Nếu use case kết thúc thành công thì thông tin về nguyên liệu được cập nhật trong cơ sở dữ liệu .  7.    Điểm mở rộng  - Không có. |

### 2.7.4. Mô tả use case Quản lý sản phẩm

|  |
| --- |
| 1.    Tên use case  - Quản lý sản phẩm  2.    Mô tả vắn tắt  - Use case này cho phép người quản trị thực hiện xem, thêm, sửa, xóa sản phẩm.  3.    Luồng sự kiện  3.1. Luồng cơ bản  1) Xem danh sách sản phẩm:  a) Use case này bắt đầu khi người quản trị kích vào nút “Quản lý sản phẩm” trên menu quản trị. Hệ thống sẽ hiển thị giao diện quản lý sản phẩm và lấy thông tin chi tiết danh sách các sản phẩm gồm có code, name, isActive, description từ bảng Product; imageUrl từ bảng ProductImage; name từ bảng ProductType; name từ bảng ProductSize; price từ bảng ProductDetails; name từ bảng Unit; trong cơ sở dữ liệu và hiển thị lên màn hình thông tin của các sản phẩm.  2) Thêm sản phẩm:  a) Người quản trị kích nút “Thêm sản phẩm”. Hệ thống sẽ hiển thị màn hình thêm sản phẩm.  b) Người quản trị nhập thông tin cần thêm: code, name, isActive, description, imageUrl, type, size, price và kích nút “Thêm”. Hệ thống sẽ thêm sản phẩm mới với các thông tin vừa nhập vào cơ sở dữ liệu và cập nhật lại danh sách các sản phẩm ở màn hình quản lý sản phẩm.  3) Sửa sản phẩm:  a) Người quản trị chọn một sản phẩm cần sửa và kích vào nút “Sửa sản phẩm” ở màn hình quản lý sản phẩm. Hệ thống sẽ lấy thông tin về sản phẩm đó bao gồm: code, name, isActive, description từ bảng Product; imageUrl từ bảng ProductImage; name từ bảng ProductType; name từ bảng ProductSize; price từ bảng ProductDetails trong cơ sở dữ liệu và hiển thị lên màn hình thông tin của sản phẩm.  b) Người quản trị nhập thông tin cho sản phẩm cần sửa rồi bấm “Sửa”. Hệ thống cập nhật thông tin tương ứng của sản phẩm  được chọn vào cơ sở dữ liệu và cập nhật lại danh sách các sản phẩm lên màn hình quản lý sản phẩm  4) Xóa sản phẩm  a) Người quản trị chọn một sản phẩm cần xóa và bấm nút “Xóa”. Hệ thống sẽ hiển thị thông báo xác nhận xóa sản phẩm này.  b) Người quản trị bấm nút “Yes”. Hệ thống sẽ xóa sản phẩm mà người quản trị đã chọn  và cập nhật lại danh sách các sản phẩm lên màn hình quản lý sản phẩm.  Use case kết thúc.  3.2. Luồng rẽ nhánh  1)        Tại bước 2b hoặc 3b hoặc 4b trong luồng cơ bản nếu người quản trị kích “Hủy”. Hệ thống sẽ bỏ qua thao tác và quay trở lại màn hình hiển thị danh sách sản phẩm.  2)        Tại bất cứ thời điểm nào trong quá trình thực hiện Use Case nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiển thị một thông báo lỗi và Use Case kết thúc.  4.    Các yêu cầu đặc biệt  - Use case này chỉ cho phép một số vai trò như người quản trị, người chủ hệ thống thực hiện.  5.    Tiền điều kiện  - Người quản trị cần đăng nhập với vai trò quản trị hệ thống trước khi có thể thực hiện use case.  6.    Hậu điều kiện  - Nếu use case kết thúc thành công thì thông tin về sản phẩm sẽ được cập nhật trong cơ sở dữ liệu.  7.    Điểm mở rộng  - Không có. |

### 2.7.5. Mô tả use case Quản lý tài khoản

|  |
| --- |
| 1.    Tên use case  - Quản lý tài khoản  2.    Mô tả vắn tắt  - Use case này cho phép người quản trị thực hiện xem, thêm, sửa, xóa tài khoản.  3.    Luồng sự kiện  3.1. Luồng cơ bản  1) Xem danh sách tài khoản:  a) Use case này bắt đầu khi người quản trị kích vào nút “Quản lý tài khoản” trên menu quản trị. Hệ thống sẽ hiển thị giao diện quản lý tài khoản và lấy thông tin chi tiết danh sách các tài khoản gồm có email, password, role, fullName, gender, phone number, address, avatarUrl từ bảng ACCOUNT trong cơ sở dữ liệu và hiển thị lên màn hình thông tin của các tài khoản.  2) Thêm tài khoản:  a) Người quản trị kích nút “Thêm”. Hệ thống sẽ hiển thị màn hình thêm tài khoản.  b) Người quản trị nhập thông tin cho tài khoản bao gồm: email, password, role, fullName, gender, phone number, address, avatarUrl từ bảng ACCOUNT và kích nút “Thêm”. Hệ thống sẽ thêm tài khoản mới với role tương ứng và cập nhật lại danh sách các tài khoản ở màn hình quản lý tài khoản  3) Sửa tài khoản:  a) Người quản trị chọn một tài khoản và kích vào nút “Sửa” ở màn hình quản lý tài khoản. Hệ thống sẽ lấy thông tin tài khoản đó bao gồm.email, password, role, fullName, gender, phone number, address, avatarUrl từ bảng ACCOUNT và hiển thị lên màn hình thông tin tài khoản.  b) Người quản trị nhập thông tin cần sửa rồi bấm “Update”. Hệ thống cập nhật thông tin tương ứng của tài khoản được chọn vào bảng Account và cập nhật lại danh sách các tài khoản lên màn hình quản lý tài khoản.  4) Xóa tài khoản  a) Người quản trị chọn một tài khoản và bấm nút “Xóa”. Hệ thống sẽ hiển thị thông báo xác nhận xóa tài khoản này.   b) Người quản trị bấm nút “Yes”. Hệ thống sẽ xóa tài khoản mà người quản trị đã chọn trong bảng Account và cập nhật lại danh sách các tài khoản lên màn hình quản lý tài khoản.  Use case kết thúc.  3.2. Luồng rẽ nhánh  1) Tại bước 2b hoặc 3b hoặc 4b trong luồng cơ bản nếu người quản trị kích “Hủy”. Hệ thống sẽ bỏ qua thao tác và quay trở lại màn hình hiển thị danh sách tài khoản.  2) Tại bất cứ thời điểm nào trong quá trình thực hiện Use Case nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiển thị một thông báo lỗi và Use Case kết thúc.  4.    Các yêu cầu đặc biệt  - Use case này chỉ cho phép một số vai trò như người quản trị, người chủ hệ thống thực hiện.  5.    Tiền điều kiện  - Người quản trị cần đăng nhập với vai trò quản trị hệ thống trước khi có thể thực hiện use case.  6.    Hậu điều kiện  - Nếu use case kết thúc thành công thì thông tin về tài khoản sẽ được cập nhật trong cơ sở dữ liệu .  7.    Điểm mở rộng  - Không có. |

### **2.7.6. Mô tả chi tiết Use Case Xem thông tin đơn hàng**

|  |
| --- |
| 1.  Tên Use Case  Xem thông tin đơn hàng  2.  Mô tả vắn tắt  Use case này cho phép người dùng xem thông tin các đơn hàng đã đặt.  3.  Luồng các sự kiện  3.1.  Luồng cơ bản  1.    Use case bắt đầu khi người dùng click vào nút “Account” trên thanh menu. Hệ thống sẽ menu của người dùng bao gồm : Profile, My Order, Log Out.  2. Người dùng click vào nút “My Order”  ở thanh menu người dùng. Hệ thống sẽ hiển thị danh sách các đơn hàng đã đặt lên màn hình.  3.   Người dùng kích vào 1 đơn hàng. Hệ thống hiển thị thông tin chi tiết bao gồm: CustomerName, PhoneNumber, Address, Status, CreatedDate từ bảng Order; Quantity, TotalMoney từ bảng OrderDetails; Price từ bảng ProductDetails; Name từ bảng ProductSize; Name từ bảng Product; ImageUrl từ bảng ProductImage trong cơ sở dữ liệu lên màn hình. Use case kết thúc.  3.2.      Các luồng rẽ nhánh     1.  Tại bước 2 trong luồng cơ bản, nếu khách hàng chưa có đơn hàng nào thì hệ thống hiển thị “You have no Order”.  2. Tại bất cứ thời điểm nào trong quá trình thực hiện Use Case nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiển thị một thông báo lỗi.  Use Case kết thúc   4.  Các yêu cầu đặc biệt        Không có   5.  Tiền điều kiện        Khách hàng phải đăng nhập vào hệ thống.   6.  Hậu điều kiện         Không có   7.  Điều kiện mở rộng     Không có |

## **2.8. Thiết kế cơ sở dữ liệu**

**- Account**: Lưu thông tin về các tài khoản sử dụng trong cửa hàng. Chi tiết về các tài khoản bao gồm mã tài khoản, email, mật khẩu, quyền trong hệ thống, họ tên, giới tính, số điện thoại, địa chỉ của người sử dụng.

**- Cart**: Lưu thông tin về giỏ hàng của tài khoản. Thông tin trên giỏ hàng được quan tâm là mã giỏ hàng, mã tài khoản, số lượng sản phẩm trong giỏ hàng.

**- Order:** Lưu thông tin về đơn hàng đã đặt của khách hàng. Thông tin chi tiết bao gồm mã đơn hàng, họ tên khách hàng, số điện thoại, địa chỉ, trạng thái đơn hàng, ngày tạo đơn hàng, người tạo đơn hàng và mã tài khoản.

**- OrderDetails:** Lưu thông tin về chi tiết đơn hàng đã đặt của khách hàng. Thông tin chi tiết bao gồm mã chi tiết đơn hàng, số lượng đặt hàng, tổng tiền, mã đơn hàng và mã chi tiết sản phẩm, mã tài khoản, mã sản phẩm, mã loại, mã giỏ hàng và mã size sản phẩm.

**- Product:** Lưu thông tin về các sản phẩm của cửa hàng. Thông tin chi tiết bao gồm mã sản phẩm, tên sản phẩm, trạng thái, mô tả và mã loại sản phẩm.

**- ProductDetails:** Lưu thông tin về chi tiết của sản phẩm trong cửa hàng. Thông tin chi tiết bao gồm mã chi tiết sản phẩm, mã sản phẩm, mã size sản phẩm, mã loại, mã giỏ hàng, mã tài khoản và giá bán.

**- ProductType:** Lưu thông tin về các loại sản phẩm của cửa hàng. Thông tin chi tiết bao gồm mã loại sản phẩm, tên loại sản phẩm, nhóm loại và mô tả chi tiết.

**- ProductSize:** Lưu thông tin về các size của sản phẩm trong cửa hàng. Thông tin chi tiết bao gồm mã size sản phẩm, tên size sản phẩm và mô tả chi tiết.

**- ProductImage:** Lưu thông tin về các hình ảnh của sản phẩm trong cửa hàng. Thông tin chi tiết bao gồm mã hình ảnh sản phẩm, mã sản phẩm, đường dẫn ảnh và tên file ảnh sản phẩm.

**- GoodsReceipt:** Lưu thông tin về phiếu nhập hàng trong cửa hàng. Thông tin chi tiết bao gồm mã phiếu nhập, mã nhà cung cấp, tên nhà cung cấp, mô tả phiếu nhập, ngày nhập.

**- GoodsReceiptDetails:** Lưu thông tin chi tiết phiếu nhập hàng. Thông tin bao gồm: mã phiếu nhập, mã nguyên vật liệu, số lượng, đơn giá.

**- GoodsIssue:** Lưu thông tin phiếu xuất hàng trong cửa hàng. Thông tin bao gồm mã phiếu xuất, ngày xuất, mô tả.

**- GoodsIssueDetails:** Lưu thông tin chi tiết phiếu xuất hàng. Thông tin bao gồm mã phiếu xuất, mã nguyên vật liệu, số lượng.

**- Meterial:** Lưu thông tin nguyên vật liệu trong cửa hàng. Lưu thông tin bao gồm mã sản phẩm, tên sản phẩm, số lượng, số lượng thấp nhất, mô tả sản phẩm.

**- Unit:** Lưu thông tin đơn vị của các sản phẩm trong cửa hàng. Thông tin bao gồm mã đơn vị, tên đơn vị.

- Mỗi sản phẩm có 0, 1 hoặc nhiều chi tiết sản phẩm. Mỗi chi tiết sản phẩm phải thuộc 1 sản phẩm.

- Mỗi tài khoản có 1 giỏ hàng. Mỗi giỏ hàng chỉ có trong 1 tài khoản.

- Mỗi giỏ hàng có thể có 0, 1 hoặc nhiều sản phẩm. Mỗi sản phẩm có thể có trong 0,1 hoặc nhiều giỏ hàng.

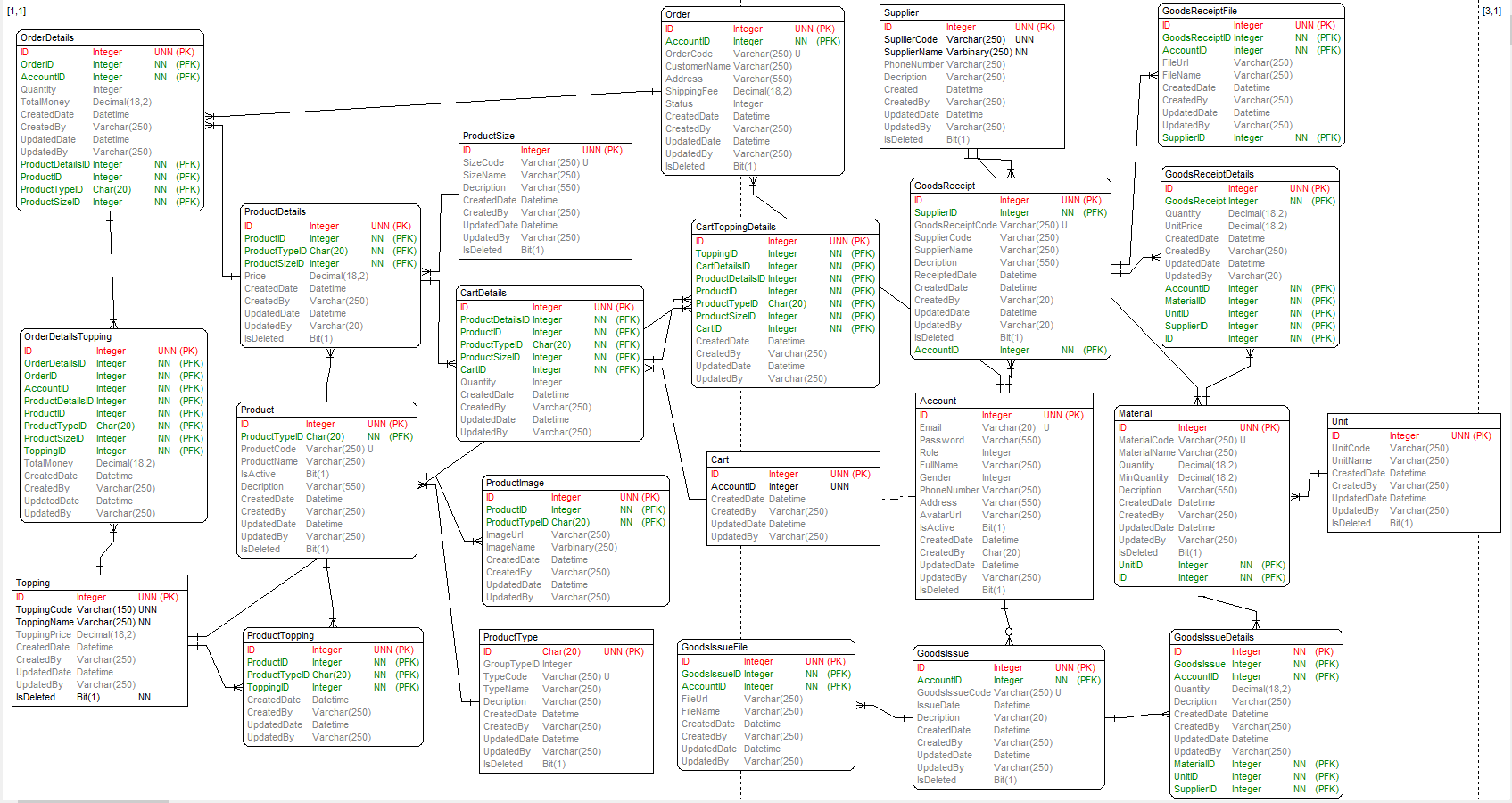
- Mỗi tài khoản có thể đặt 0, 1 hoặc nhiều đơn hàng. Mỗi đơn hàng phải được đặt bởi 1 tài khoản.

- Mỗi loại sản phẩm có thể có 0, 1 hoặc nhiều sản phẩm. Mỗi sản phẩm phải thuộc 1 loại sản phẩm.

- Mỗi size sản phẩm có thể thuộc 0, 1 hoặc nhiều sản phẩm. Mỗi sản phẩm có thể có 0,1 hoặc nhiều size sản phẩm.

- Mỗi sản phẩm có thể có 0, 1 hoặc nhiều hình ảnh. Mỗi hình ảnh phải thuộc 1 sản phẩm.

### **2.8.1. Biểu đồ thực thể liên kết**

****

Biểu đồ thực thể liên kết

### **2.8.2. Quan hệ các bảng**

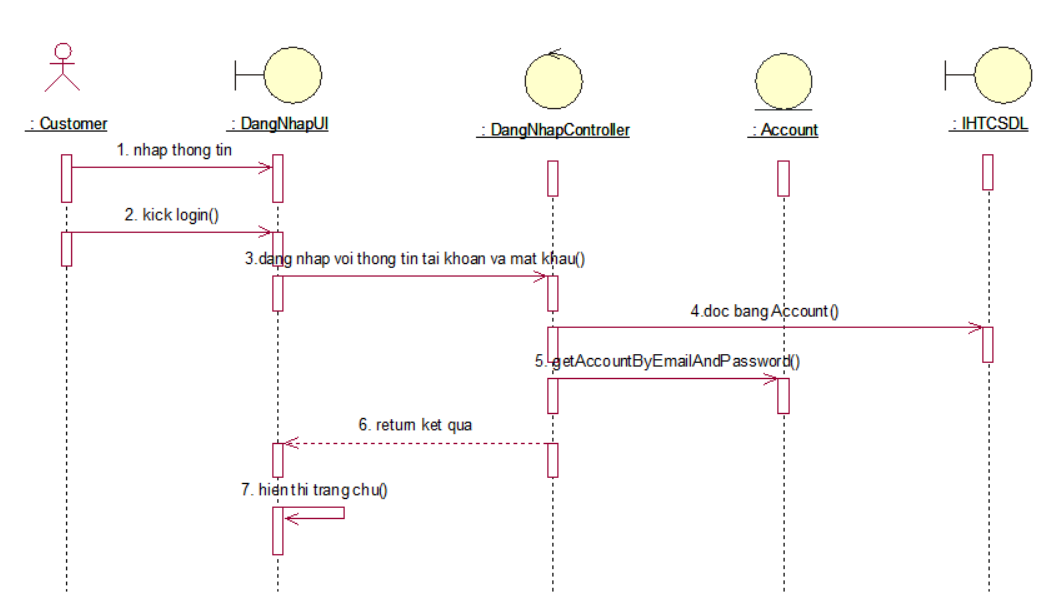


Quan hệ giữa các bảng

## **Phân tích các use case**

### **Phân tích Use case Đăng nhập**

1. **Biểu đồ trình tự**



Biểu đồ trình tự UC Đăng nhập

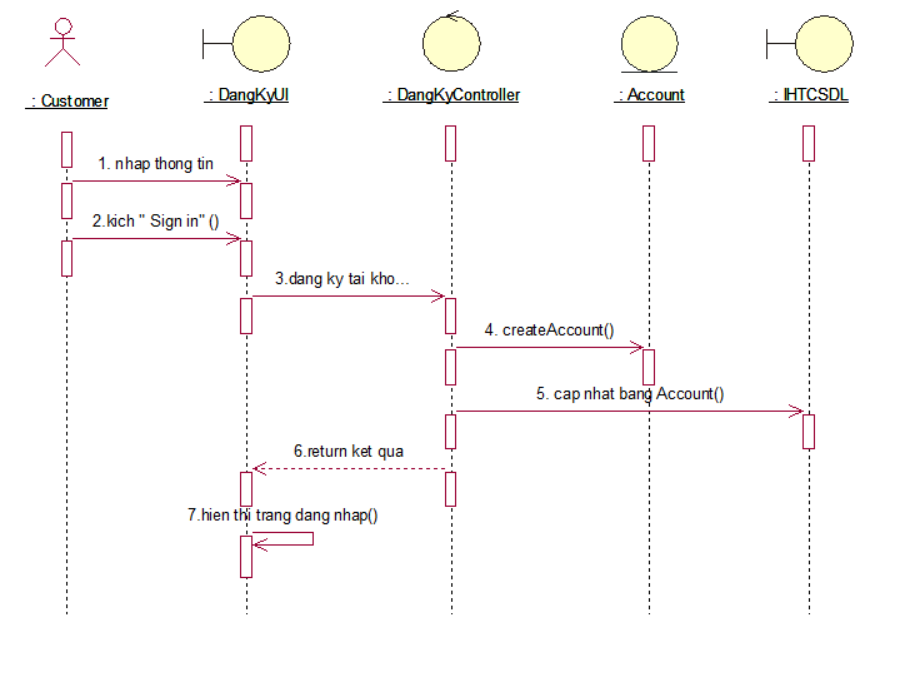
1. **Biểu đồ lớp VOPC (Phân luồng cơ bản)**

****

Biểu đồ lớp UC Đăng nhập

### **2.8.2. Phân tích Use case Đăng ký**

1. **Biểu đồ trình tự**



Hình 2.8 Biểu đồ trình tự UC Đăng ký

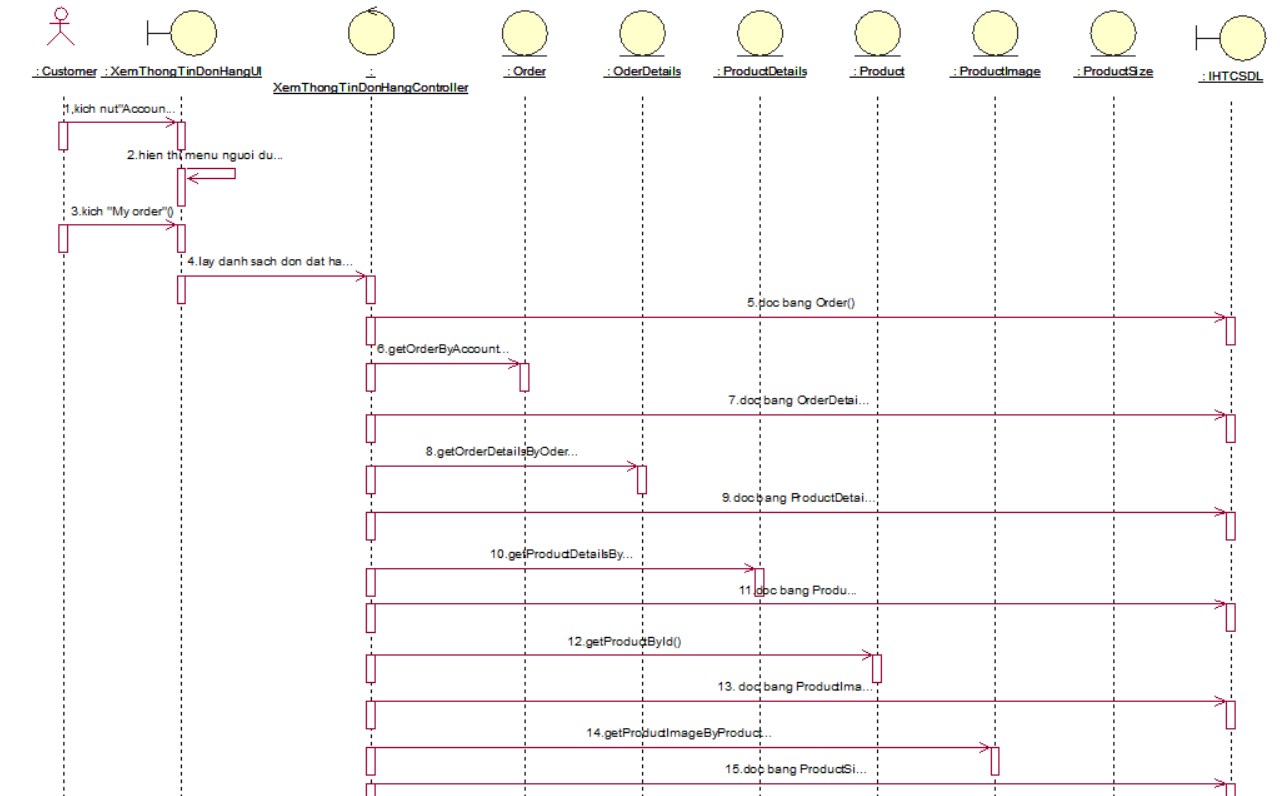
1. **Biểu đồ lớp VOPC (Phân luồng cơ bản)**

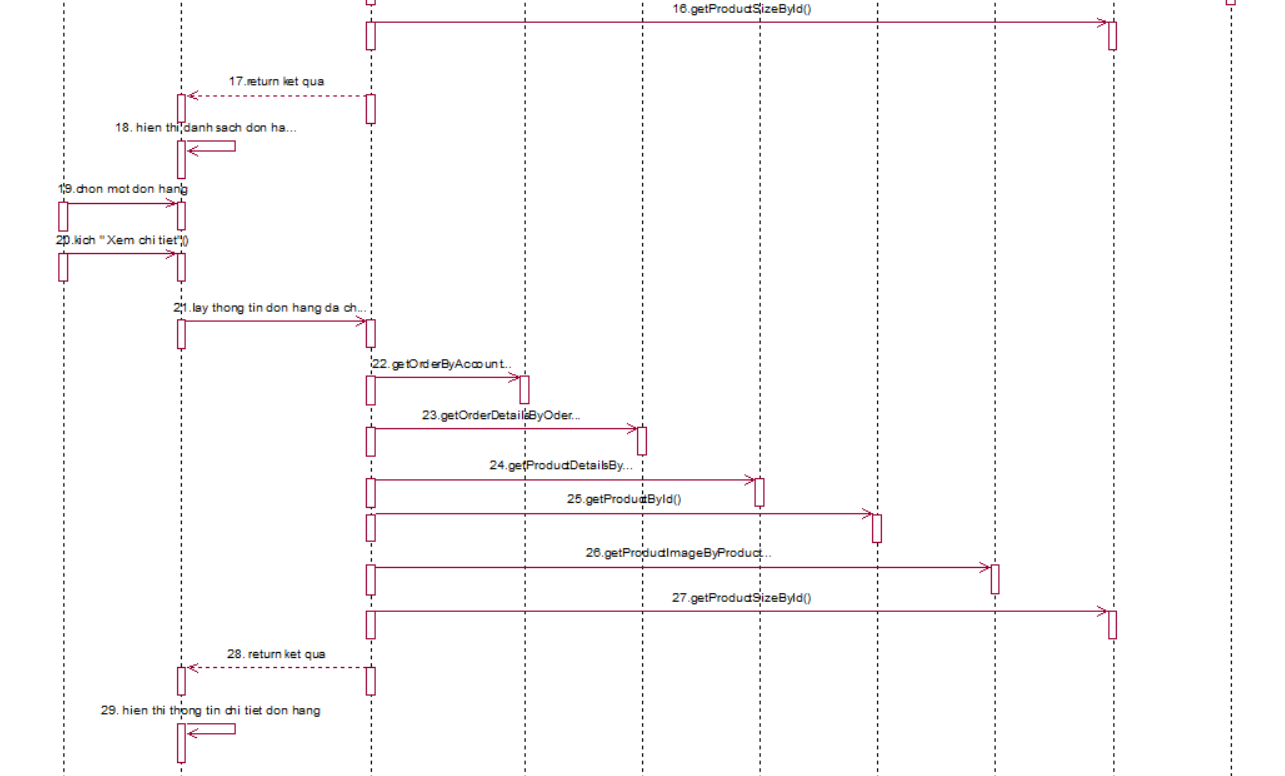


Biểu đồ lớp UC Đăng ký

### **2.8.3. Phân tích Use case Xem thông tin đơn hàng**

1. **Biểu đồ trình tự**





Biểu đồ trình tự UC Xem thông tin đơn hàng

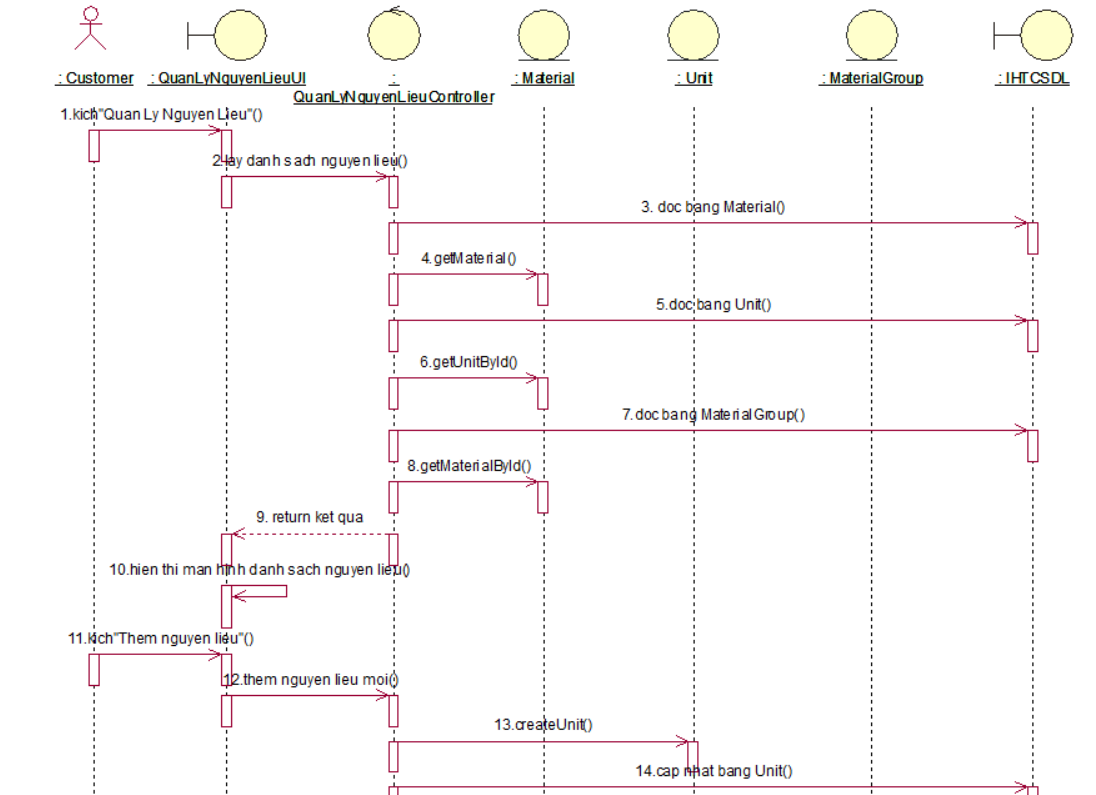
1. **Biểu đồ lớp VOPC (Phân luồng cơ bản)**

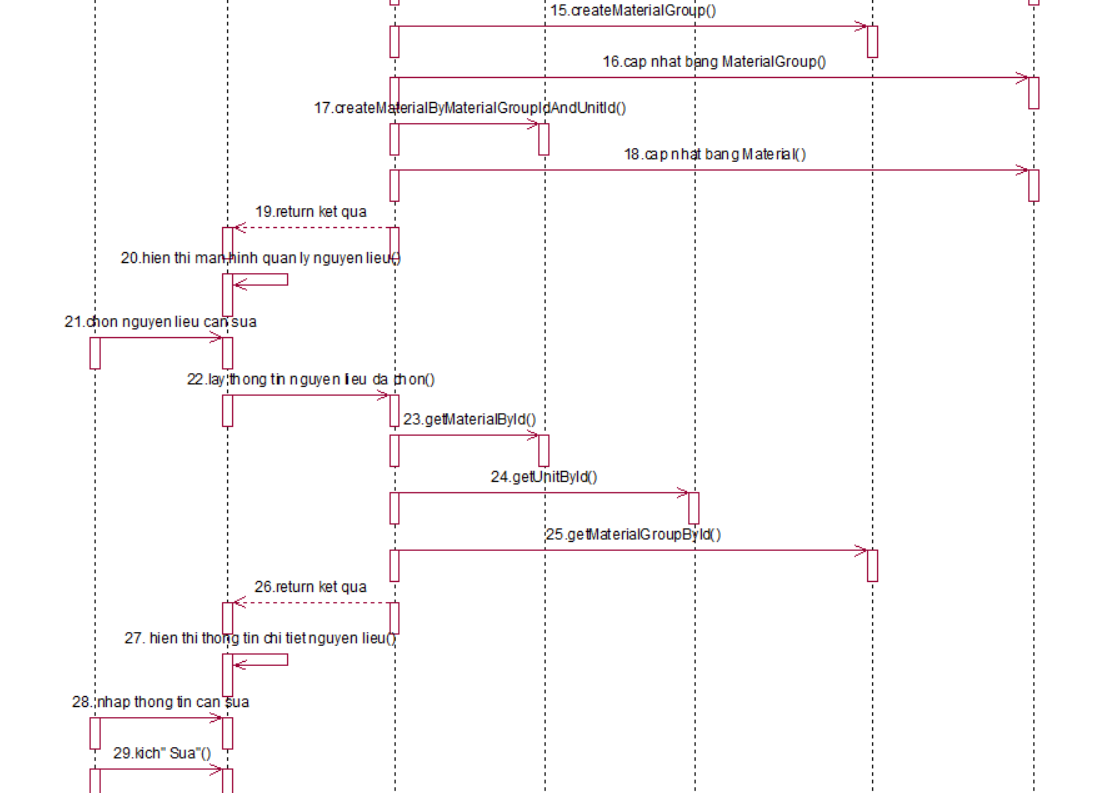
****

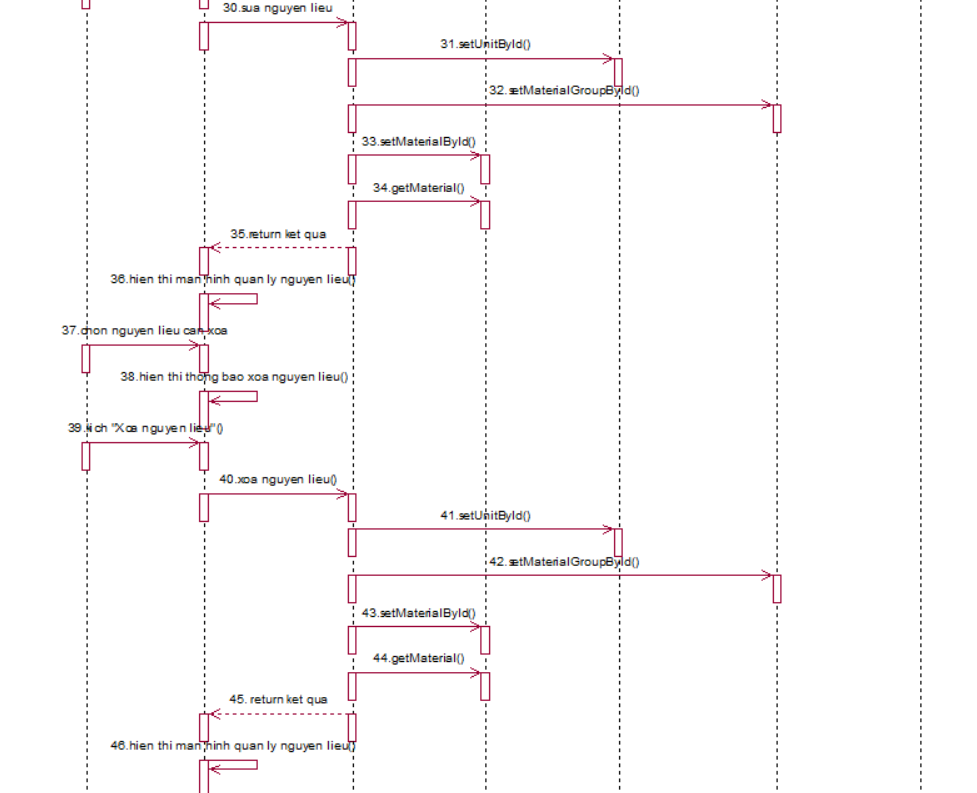
Biểu đồ lớp UC Xem thông tin đơn hàng

### **2.8.4. Phân tích Use case Quản lý Nguyên vật liệu**

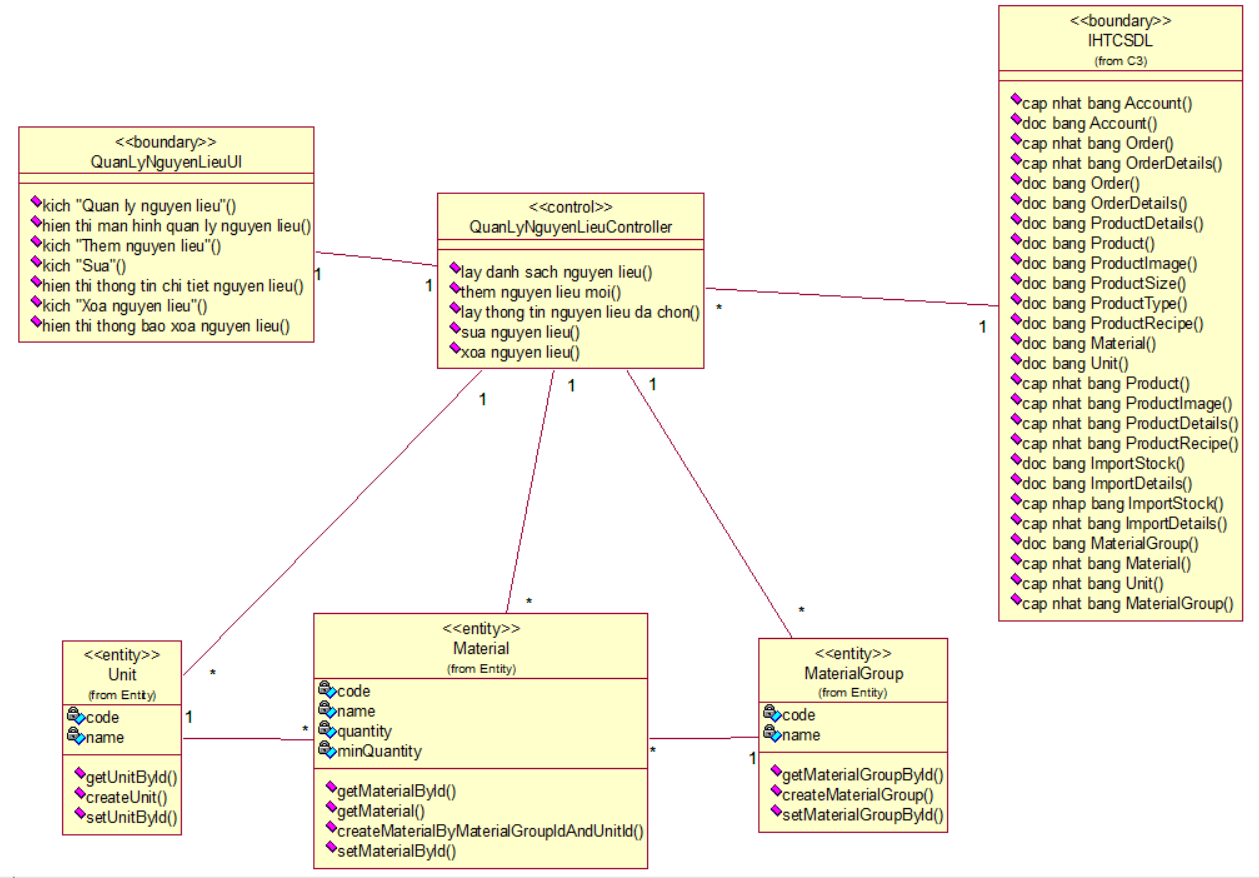
1. **Biểu đồ trình tự**

****

****

****

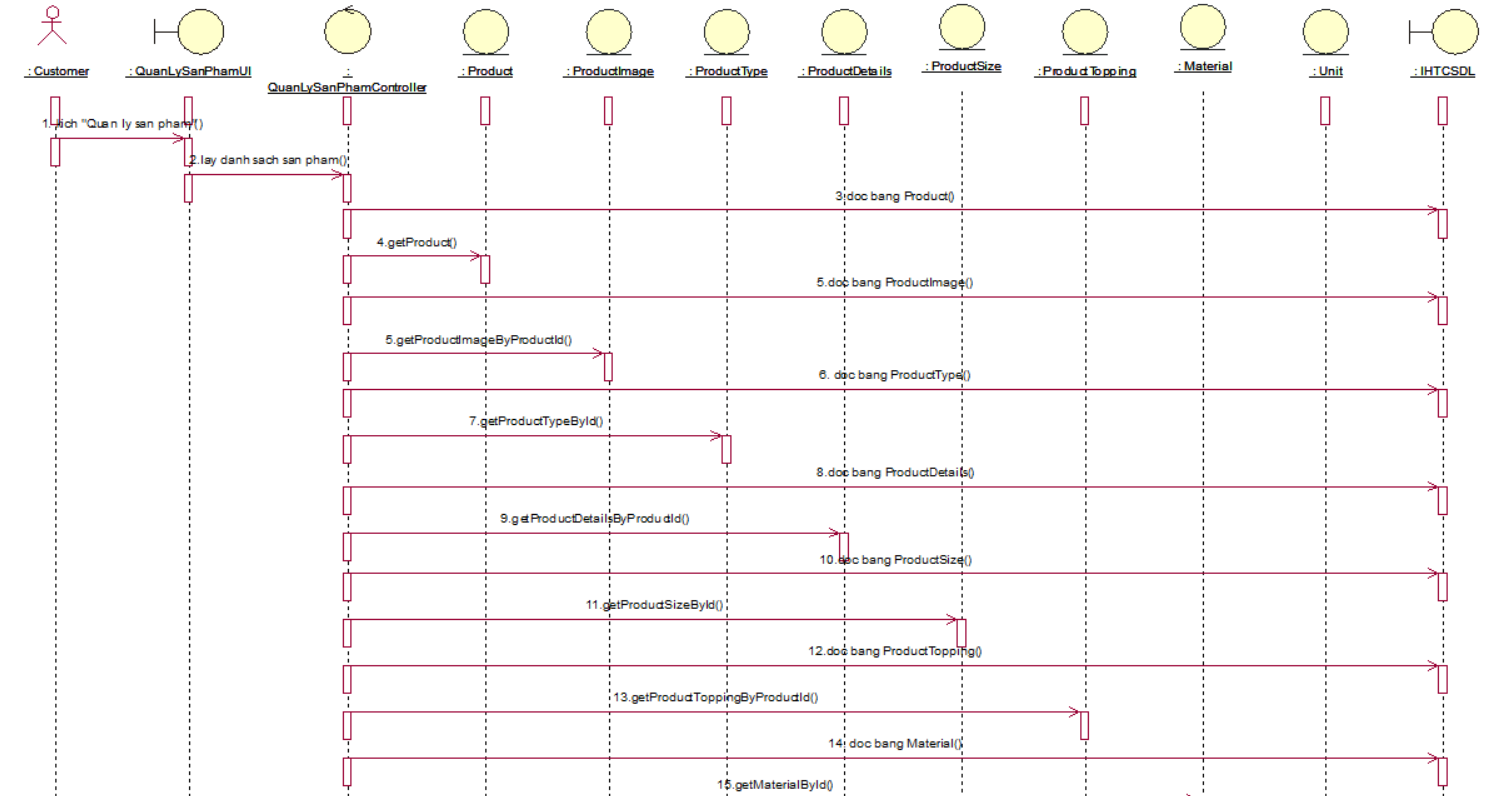
1. **Biểu đồ lớp VOPC (Phân luồng cơ bản)**

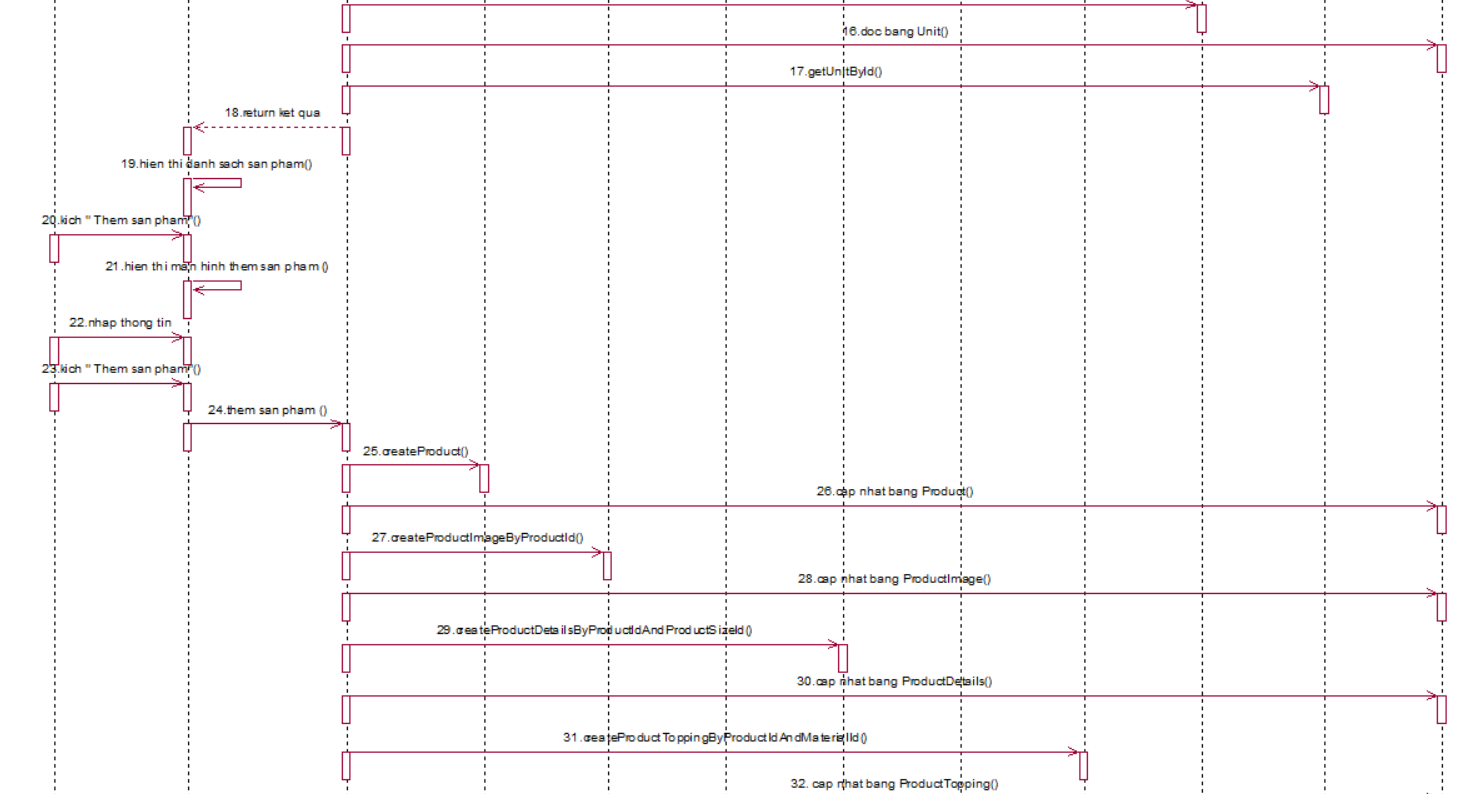


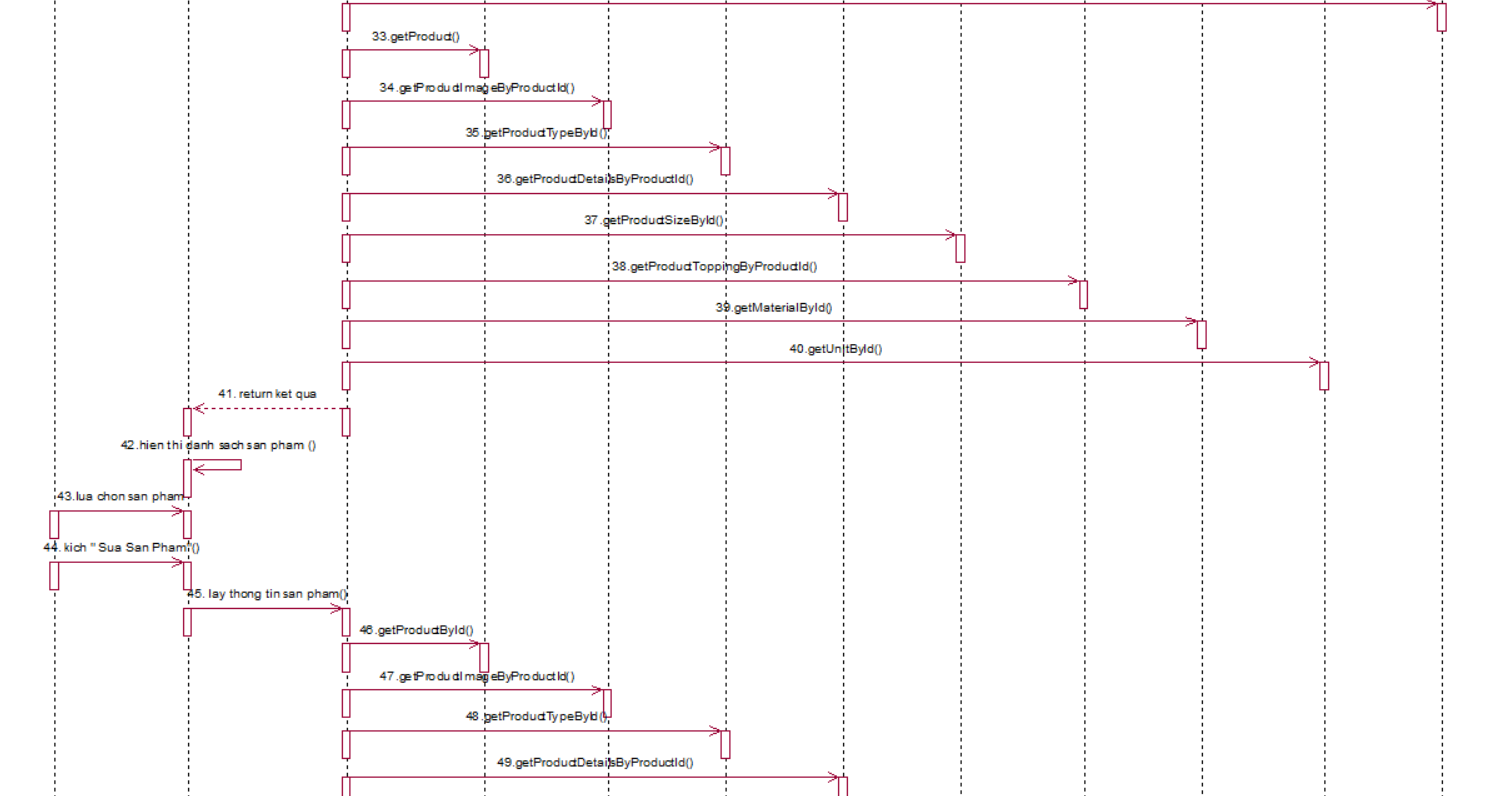
Biểu đồ lớp UC Quản lý nguyên liệu

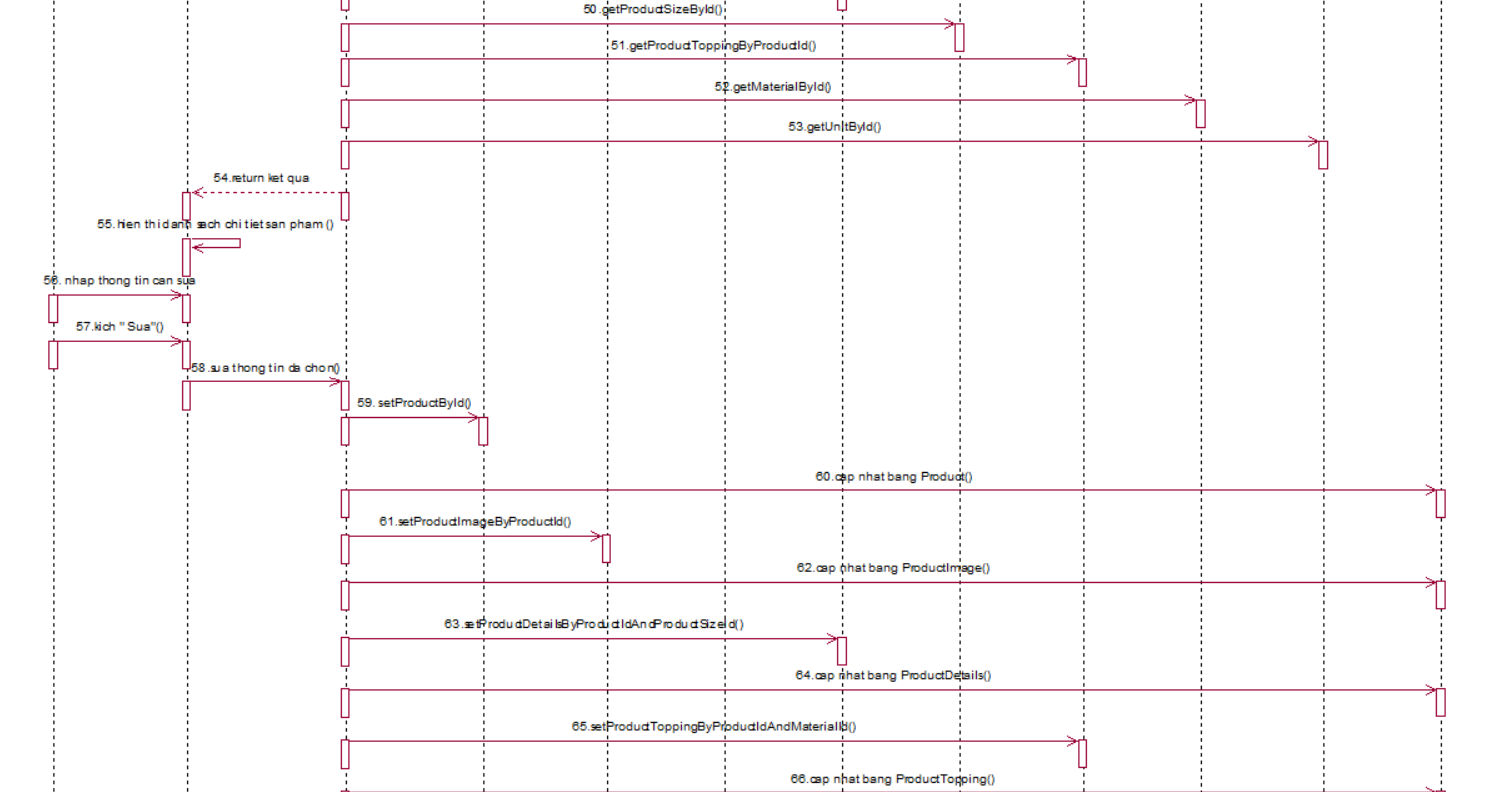
### **2.8.5. Phân tích Use case Quản lý sản phẩm**

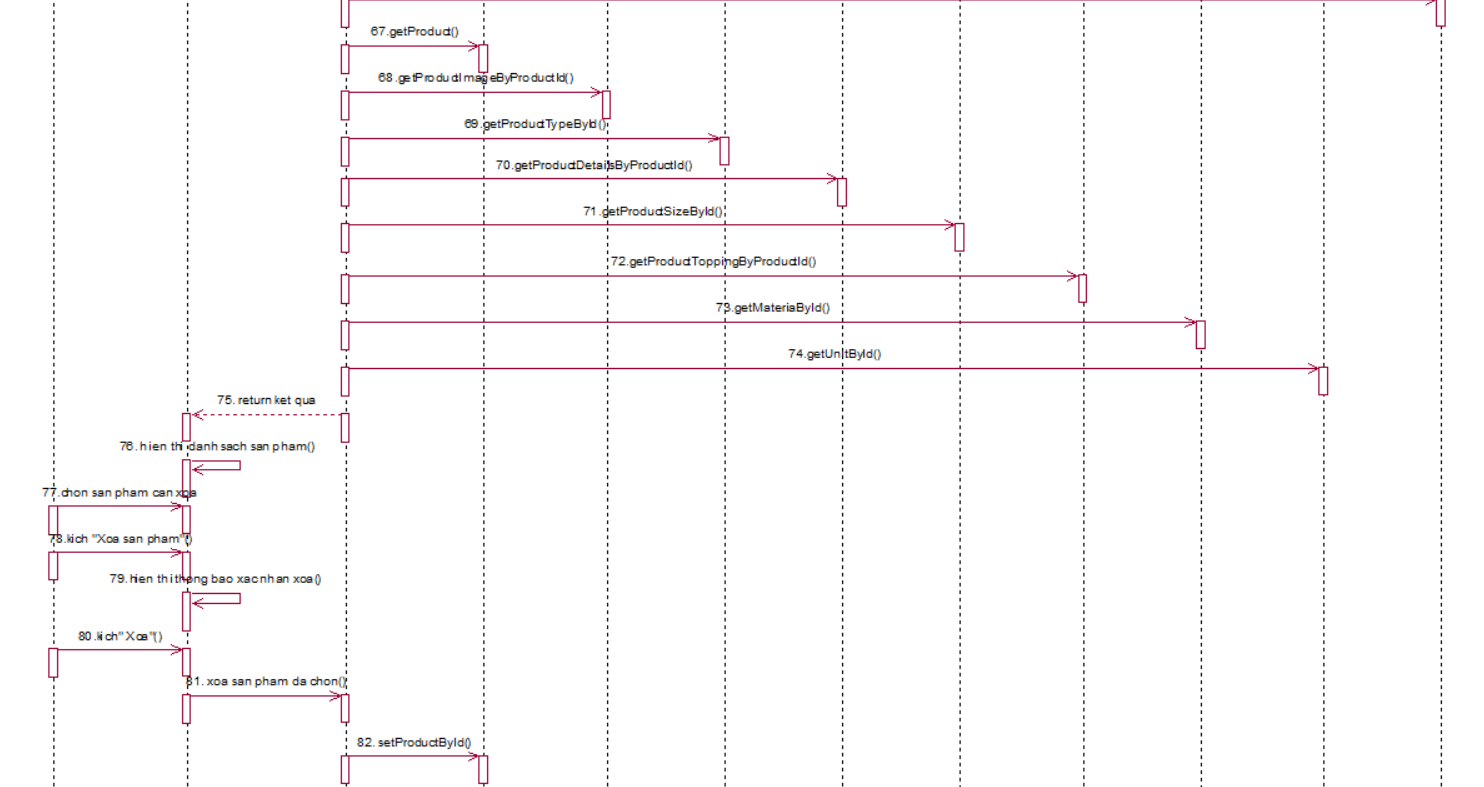
1. **Biểu đồ trình tự**

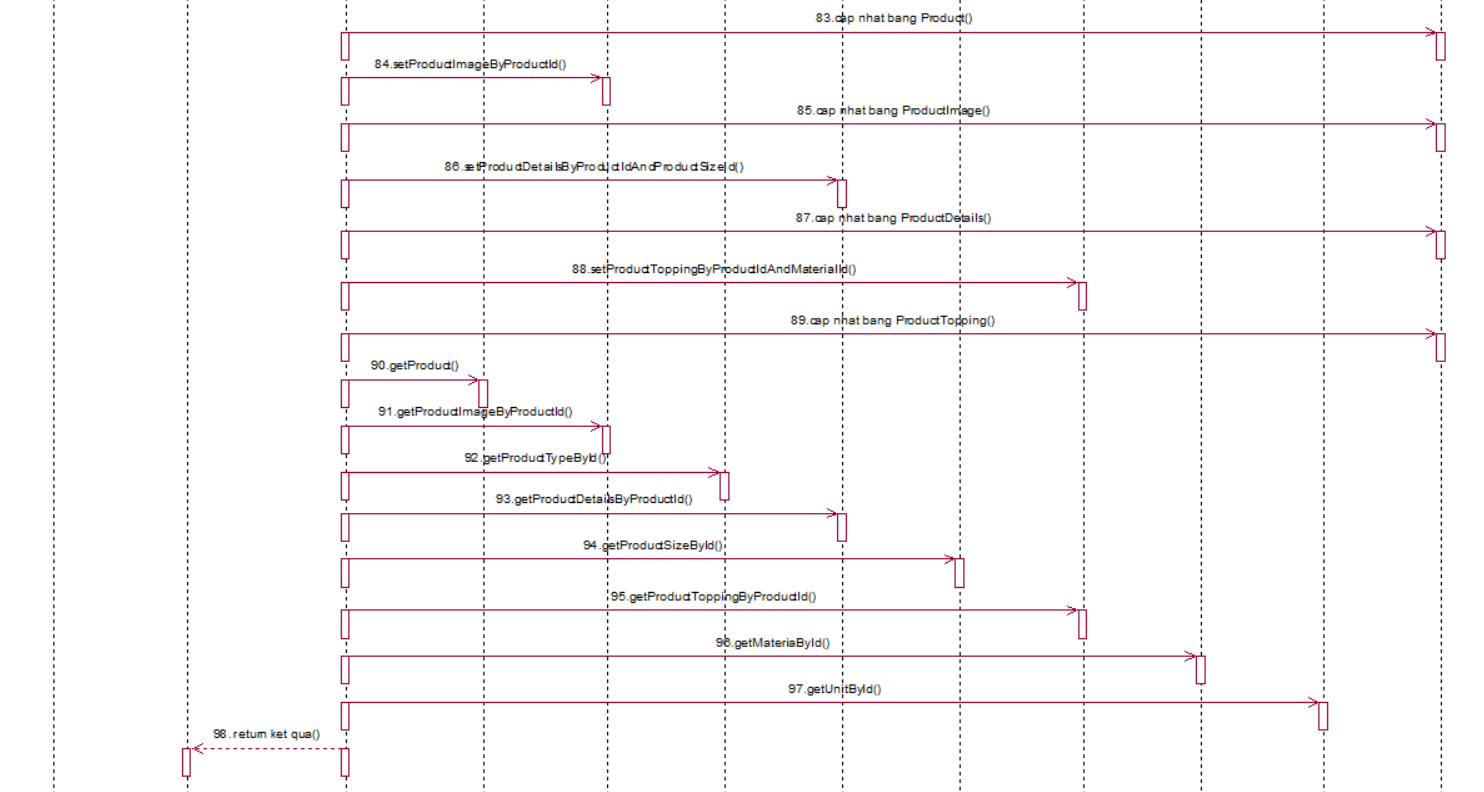






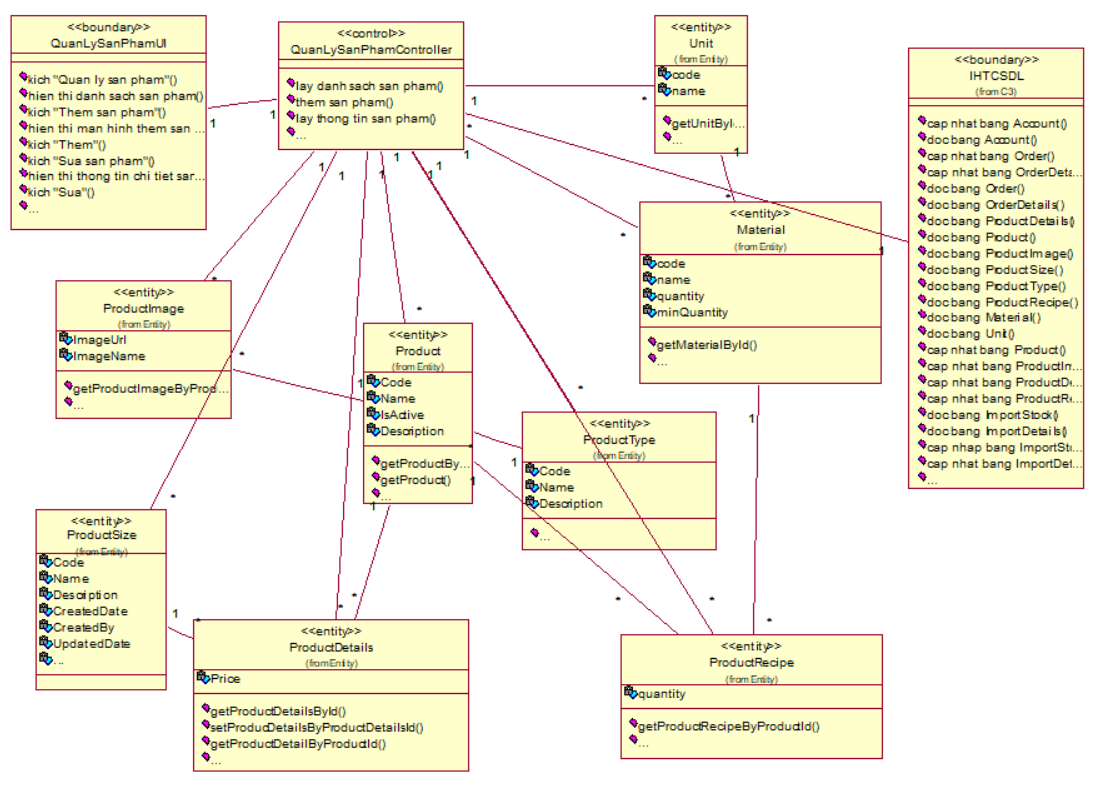






Biểu đồ trình tự UC Quản lý nguyên liệu

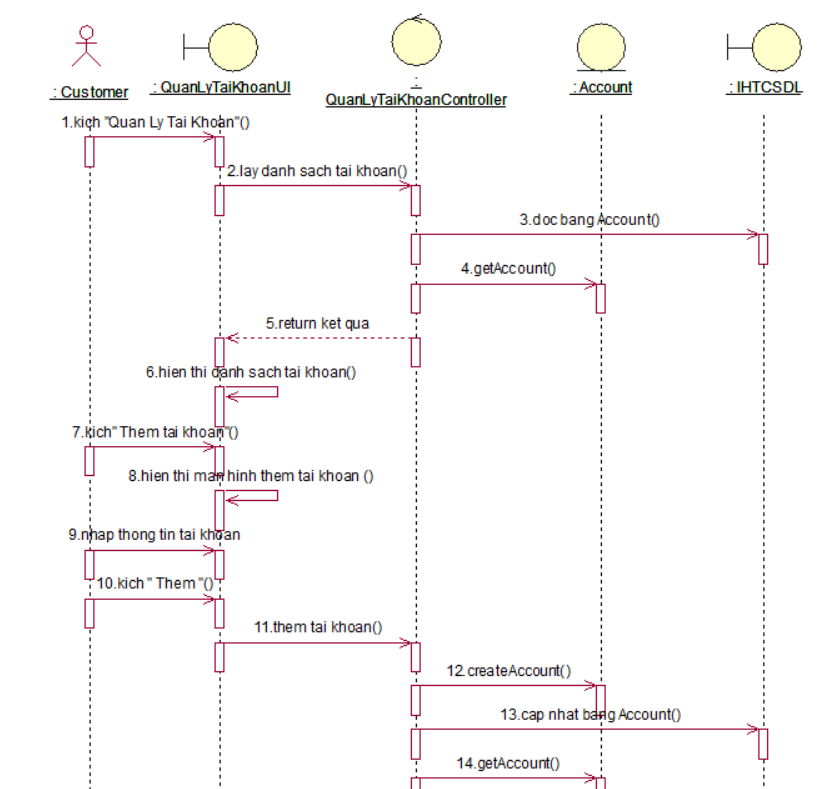
1. **Biểu đồ lớp VOPC (Phân luồng cơ bản)**

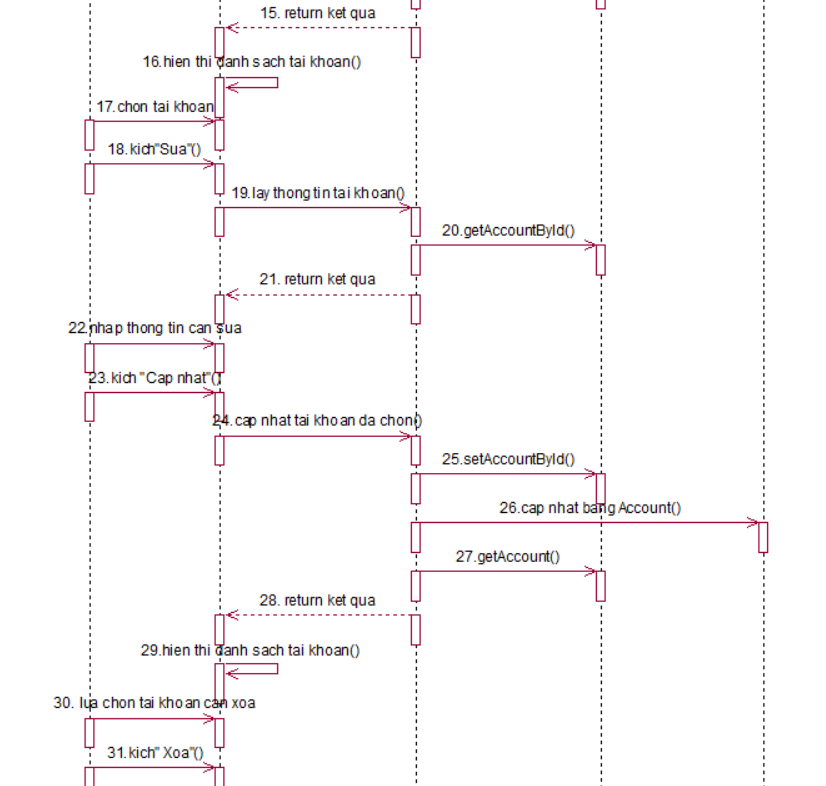


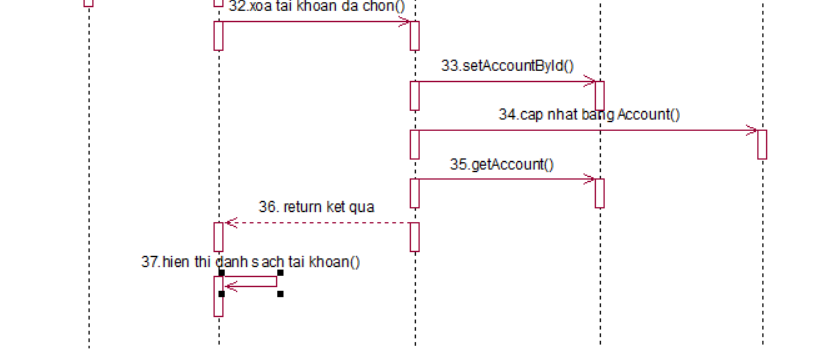
Biểu đồ lớp UC Quản lý sản phẩm

### **2.8.6. Phân tích Use case Quản lý tài khoản**

1. **Biểu đồ trình tự**

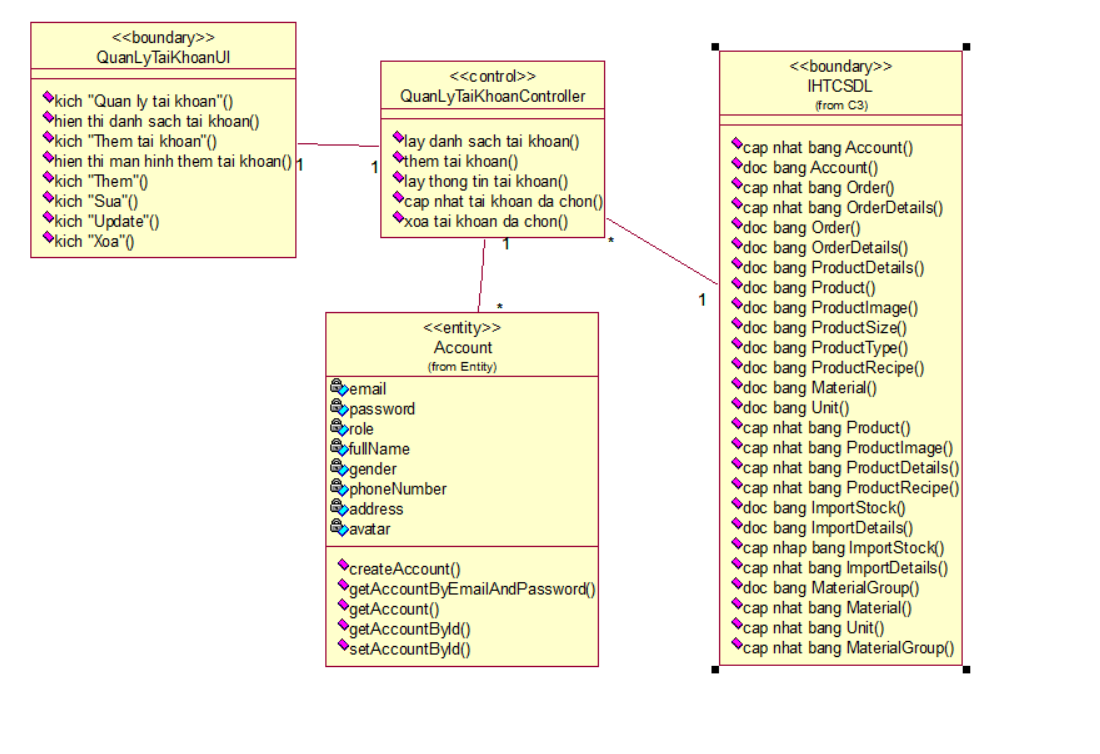
****

****

****

Biểu đồ trình tự UC Quản lý tài khoản

1. **Biểu đồ lớp VOPC (Phân luồng cơ bản)**

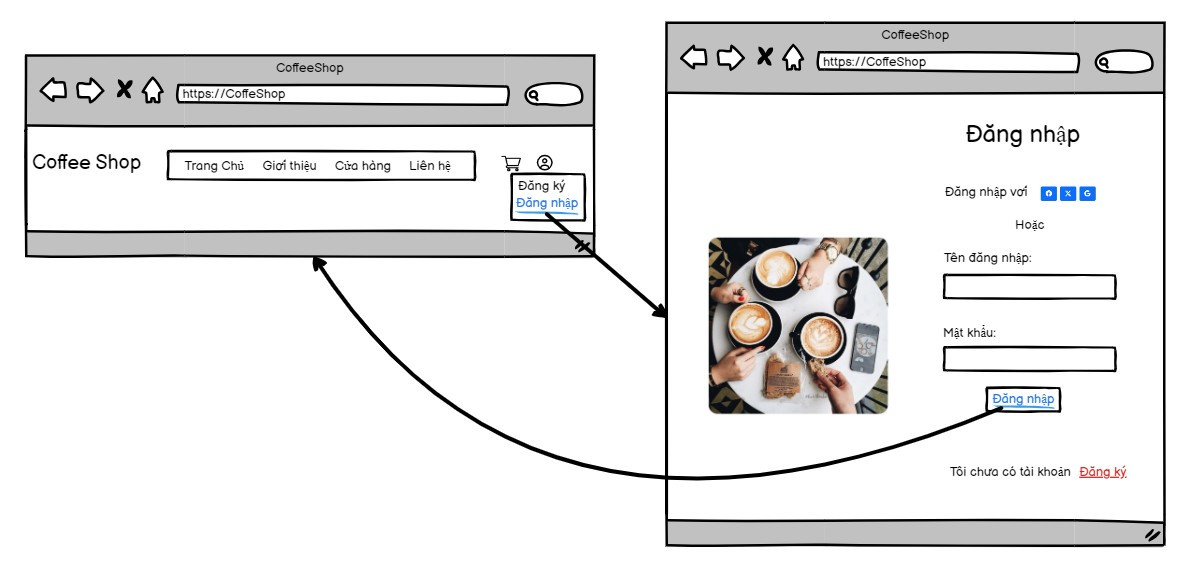


Biểu đồ lớp UC Quản lý tài khoản

## **Thiết kế giao diện**

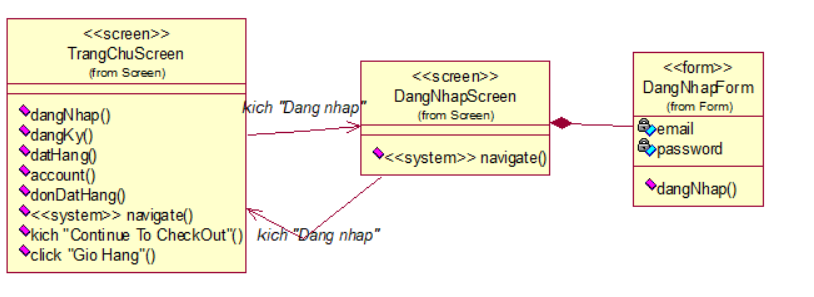
### **2.10.1. Giao diện Use Case Đăng nhập**

1. **Hình dung màn hình**



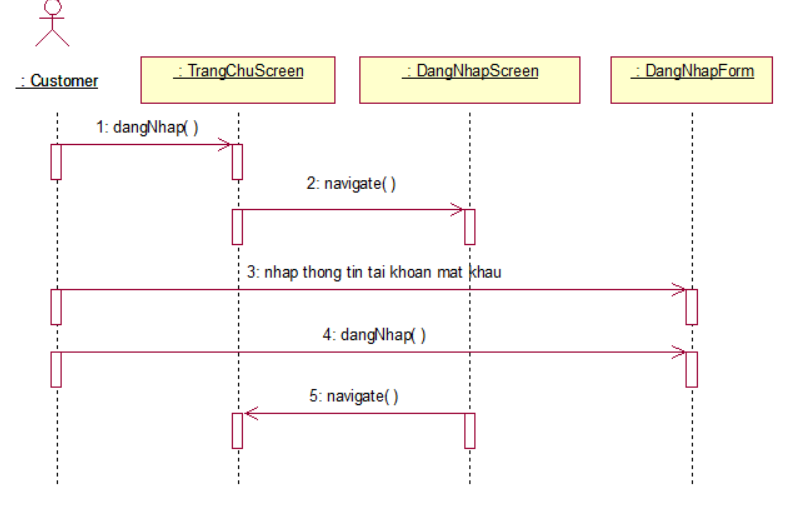
Hình dung màn hình UC Đăng nhập

1. **Biểu đồ lớp màn hình**

****

Biểu đồ lớp màn hình UC Đăng Nhập

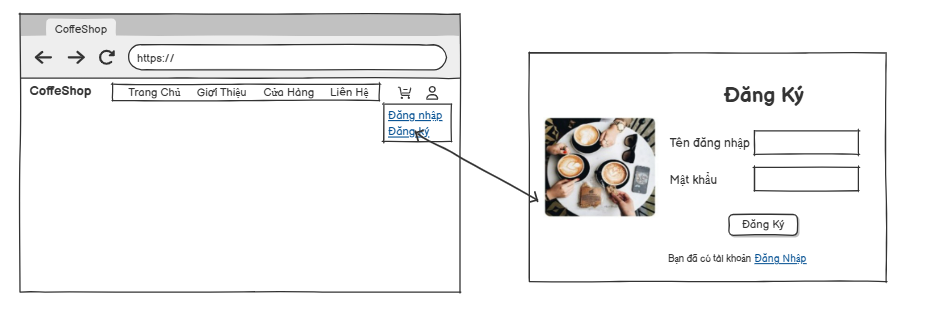
1. **Biểu đồ cộng tác màn hình**

****

Biểu đồ Cộng tác màn hình UC Đăng Nhập

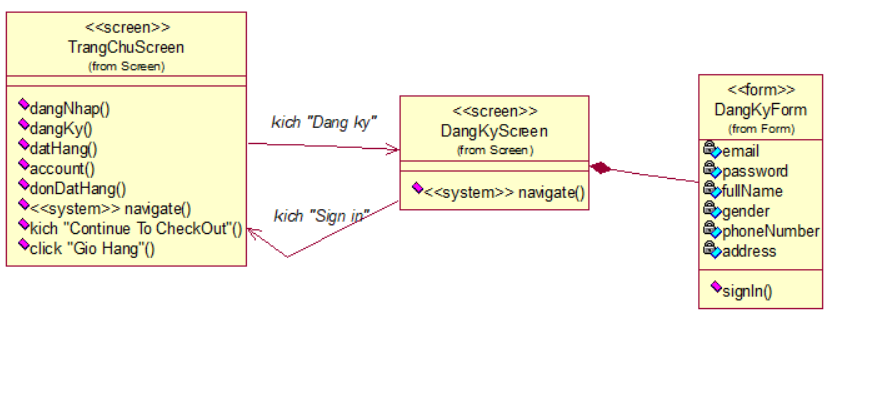
### **2.10.2. Giao diện Use Case Đăng ký**

1. **Hình dung màn hình**



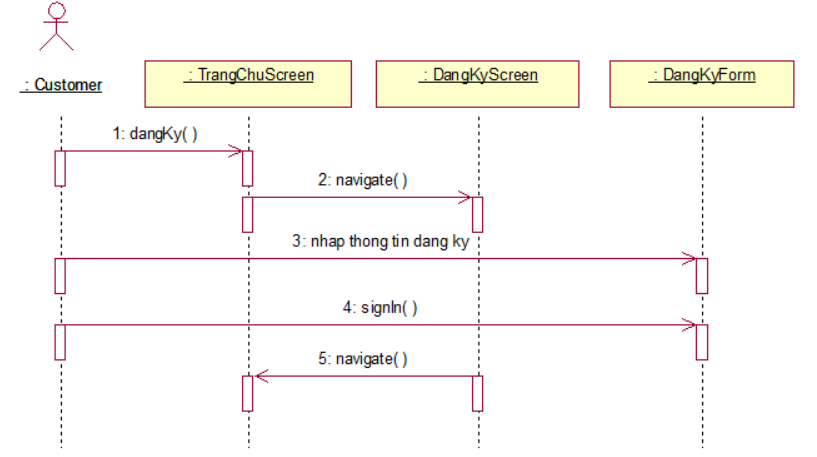
Hình dung màn hình UC Đăng ký

1. **Biểu đồ lớp màn hình**

****

Biểu đồ lớp màn hình UC Đăng Ký

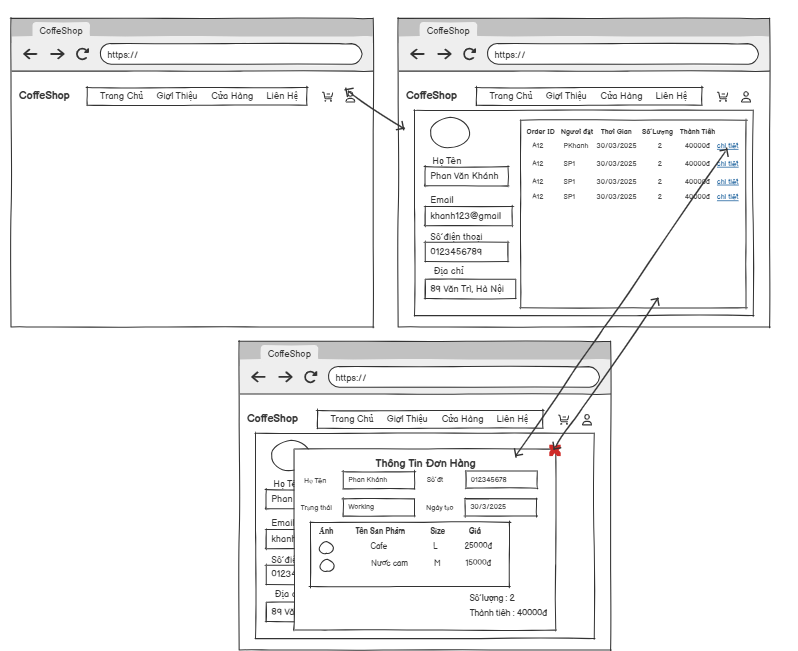
1. **Biểu đồ cộng tác màn hình**

****

Biểu đồng cộng tác màn hình UC Đăng ký

### **2.10.3. Giao diện Use Case Xem thông tin đơn hàng**

1. **Hình dung màn hình**

****

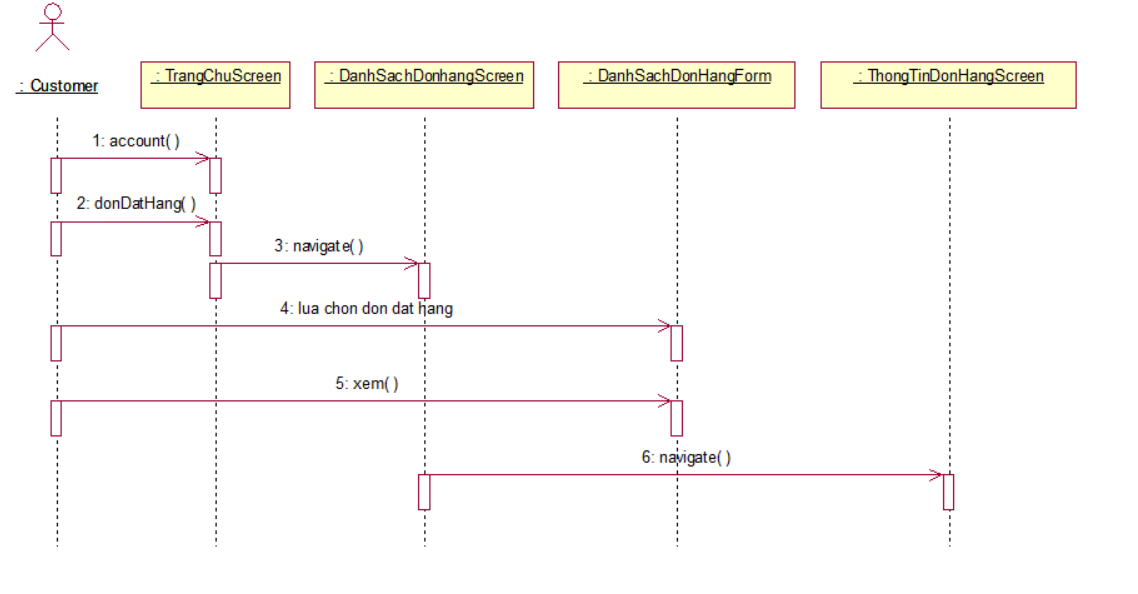
Hình dung màn hình UC Xem thông tin đơn hàng

1. **Biểu đồ lớp màn hình**

****

Biểu đồ lớp màn hình UC Xem thông tin đơn hàng

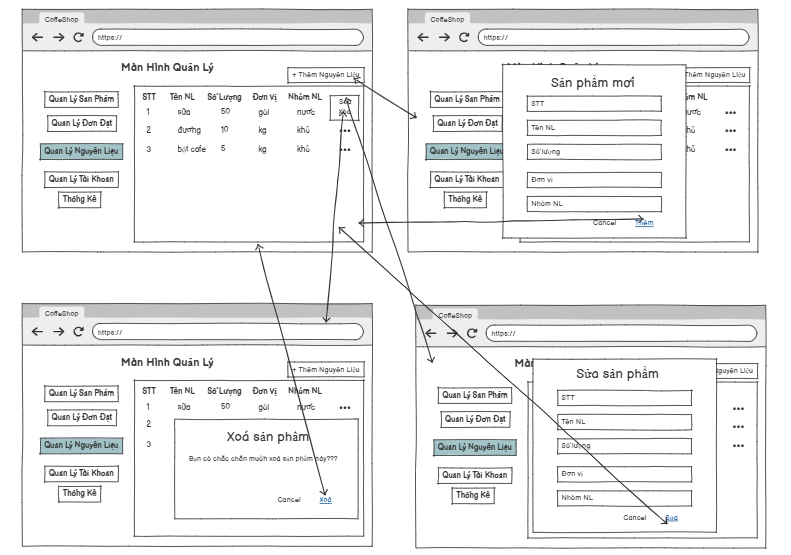
1. **Biểu đồ cộng tác màn hình**

****

Biểu đồ cộng tác màn hình UC Xem thông tin đơn hàng

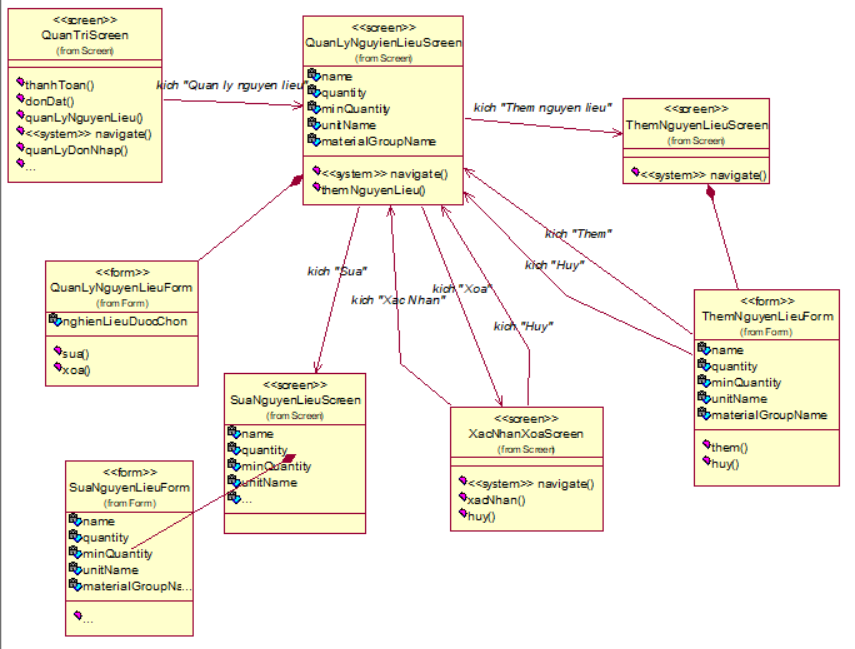
### **2.10.4. Giao diện Use Case Quản lý nguyên liệu**

1. **Hình dung màn hình**



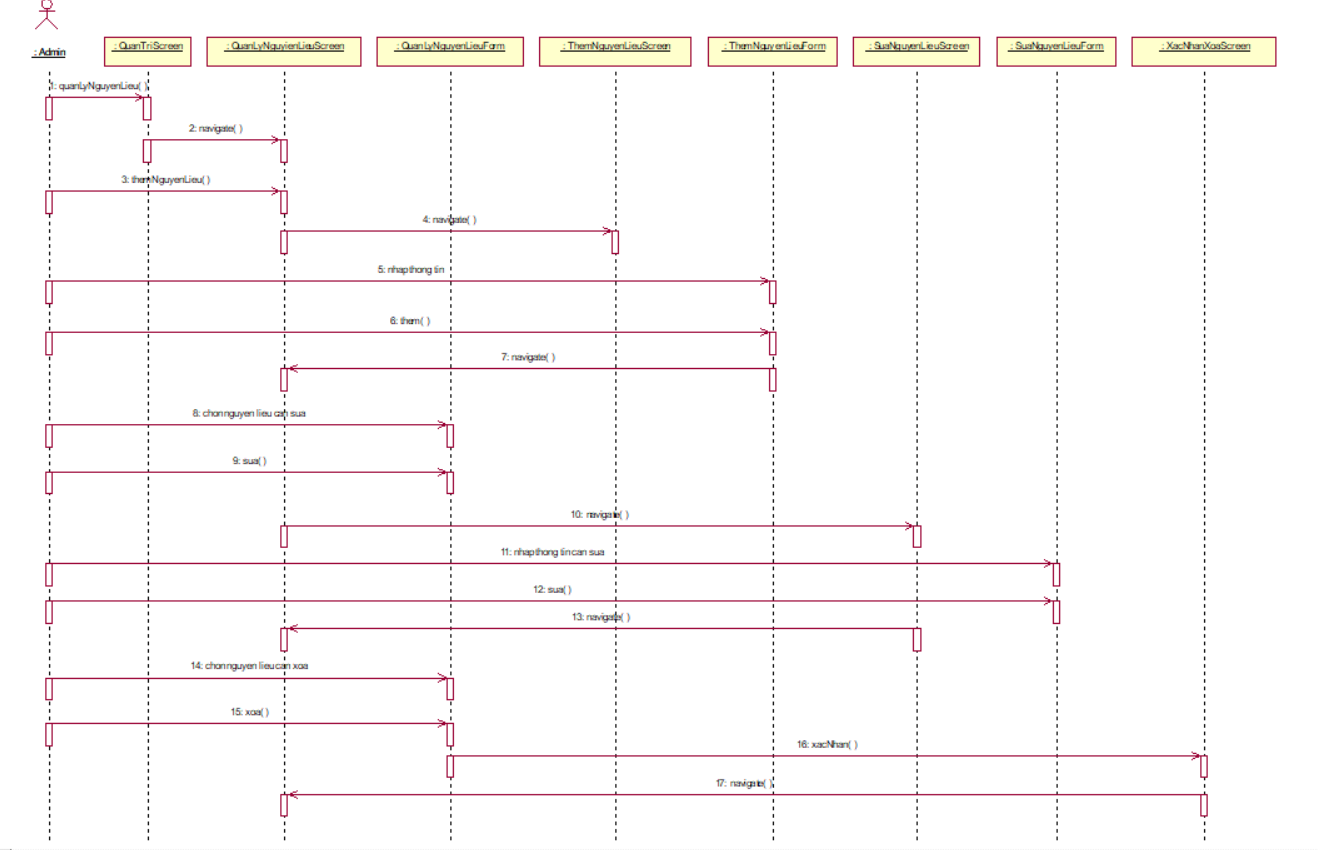
Hình dung màn hình UC Quản lý nguyên liệu

1. **Biểu đồ lớp màn hình**

****

Biểu đồ lớp màn hình UC Quản lý nguyên liệu

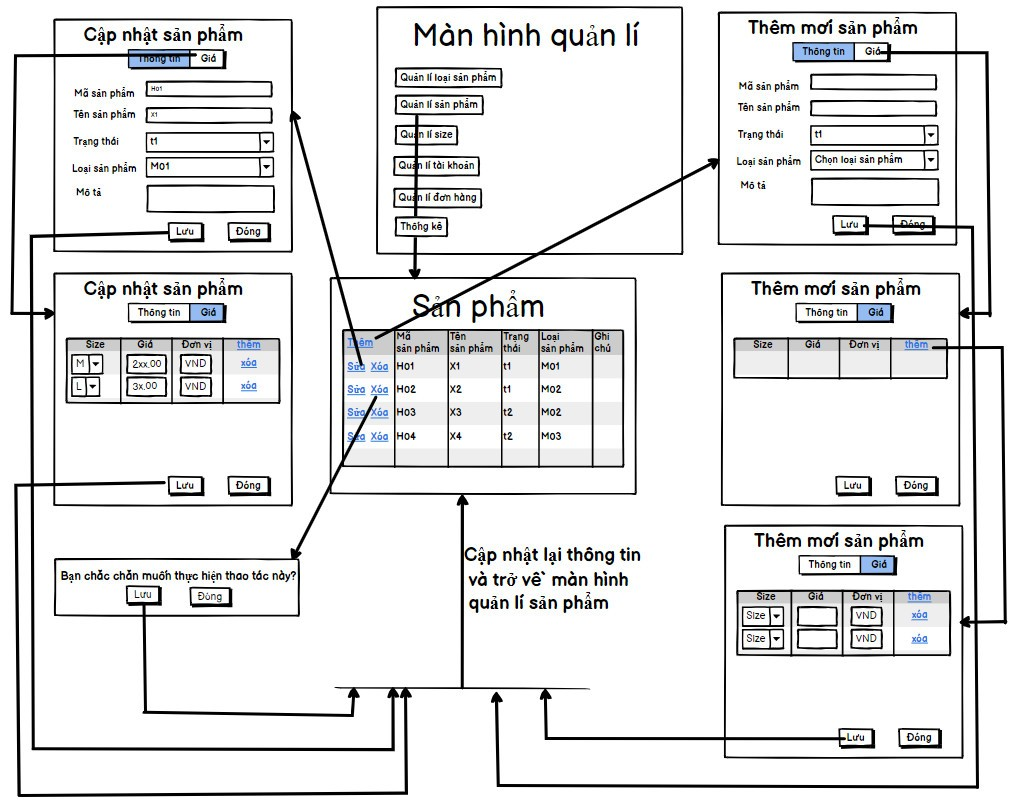
1. **Biểu đồ cộng tác màn hình**

****

Biểu đồ cộng tác màn hình UC Quản lý nguyên liệu

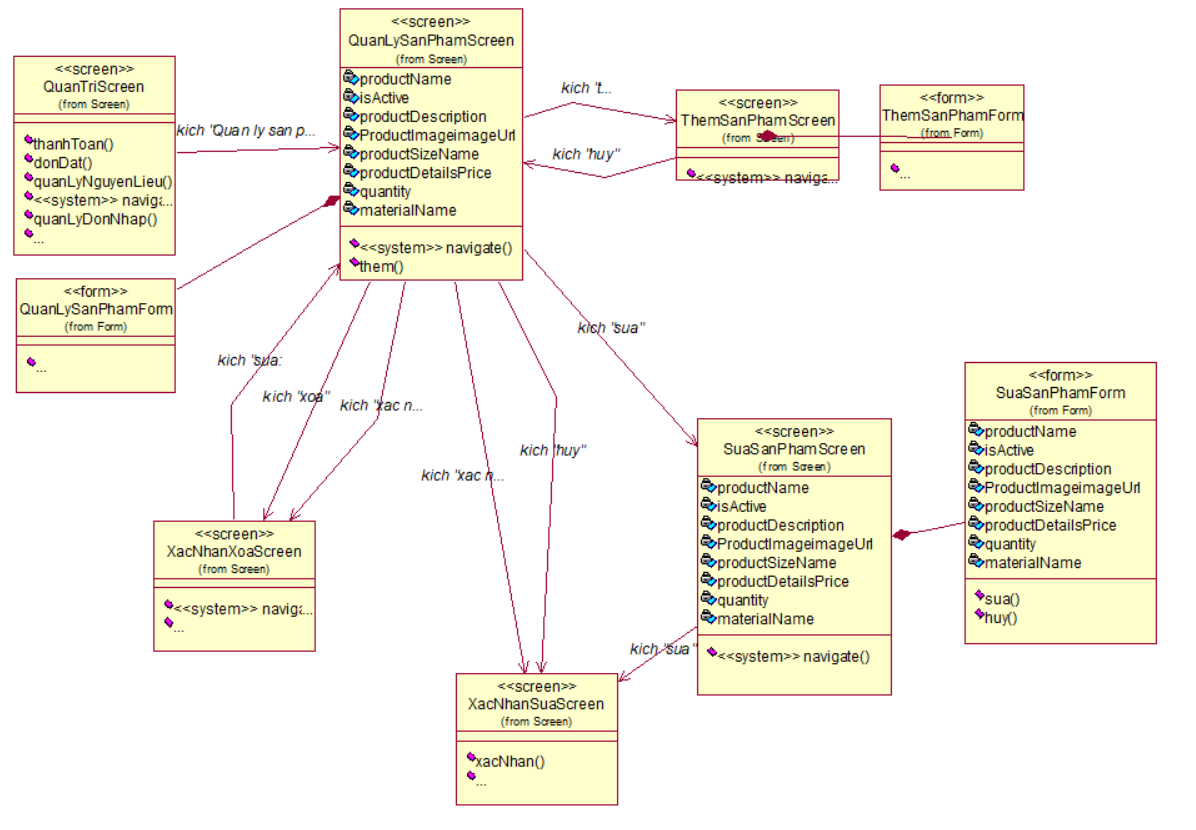
### **2.10.5. Giao diện Use Case Quản lý sản phẩm**

1. **Hình dung màn hình**



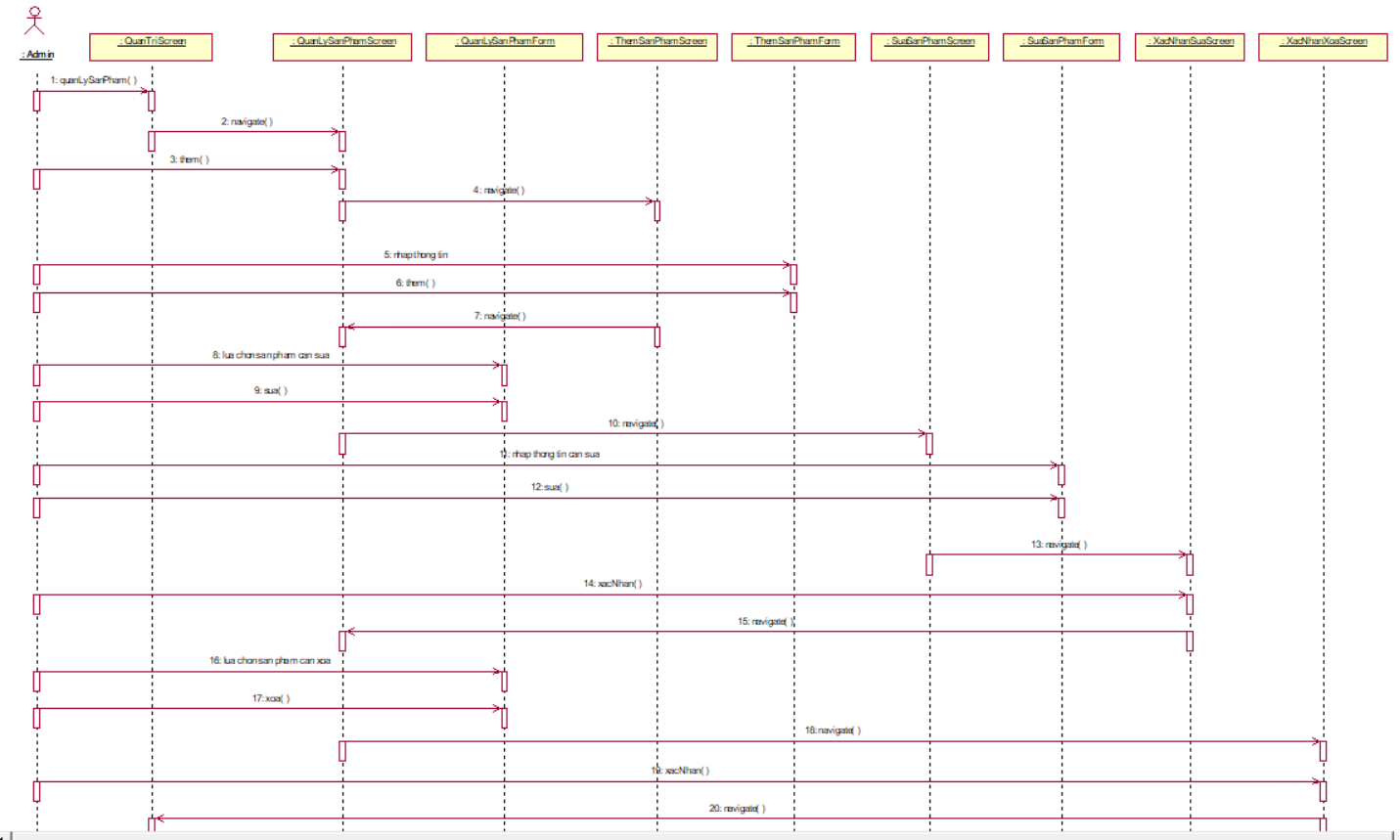
Hình dung màn hình UC Quản lý sản phẩm

1. **Biểu đồ lớp màn hình**

****

Biểu đồ lớp màn hình UC Quản lý sản phẩm

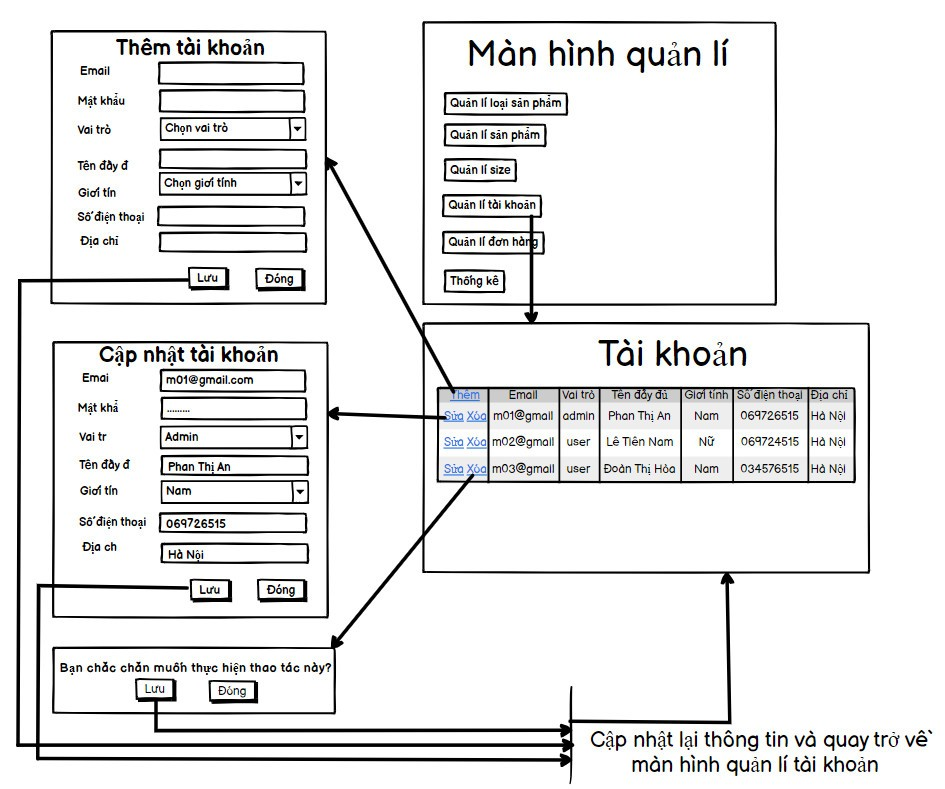
1. **Biểu đồ cộng tác màn hình**

****

Biểu đồ cộng tác màn hình UC Quản lý sản phẩm

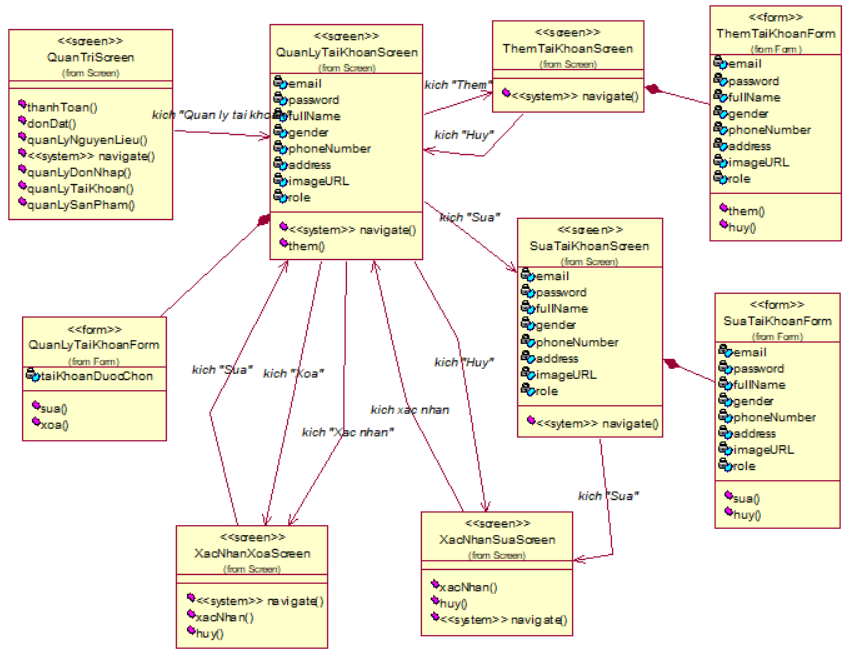
### **2.10.6. Giao diện Use Case Quản lý tài khoản**

1. **Hình dung màn hình**



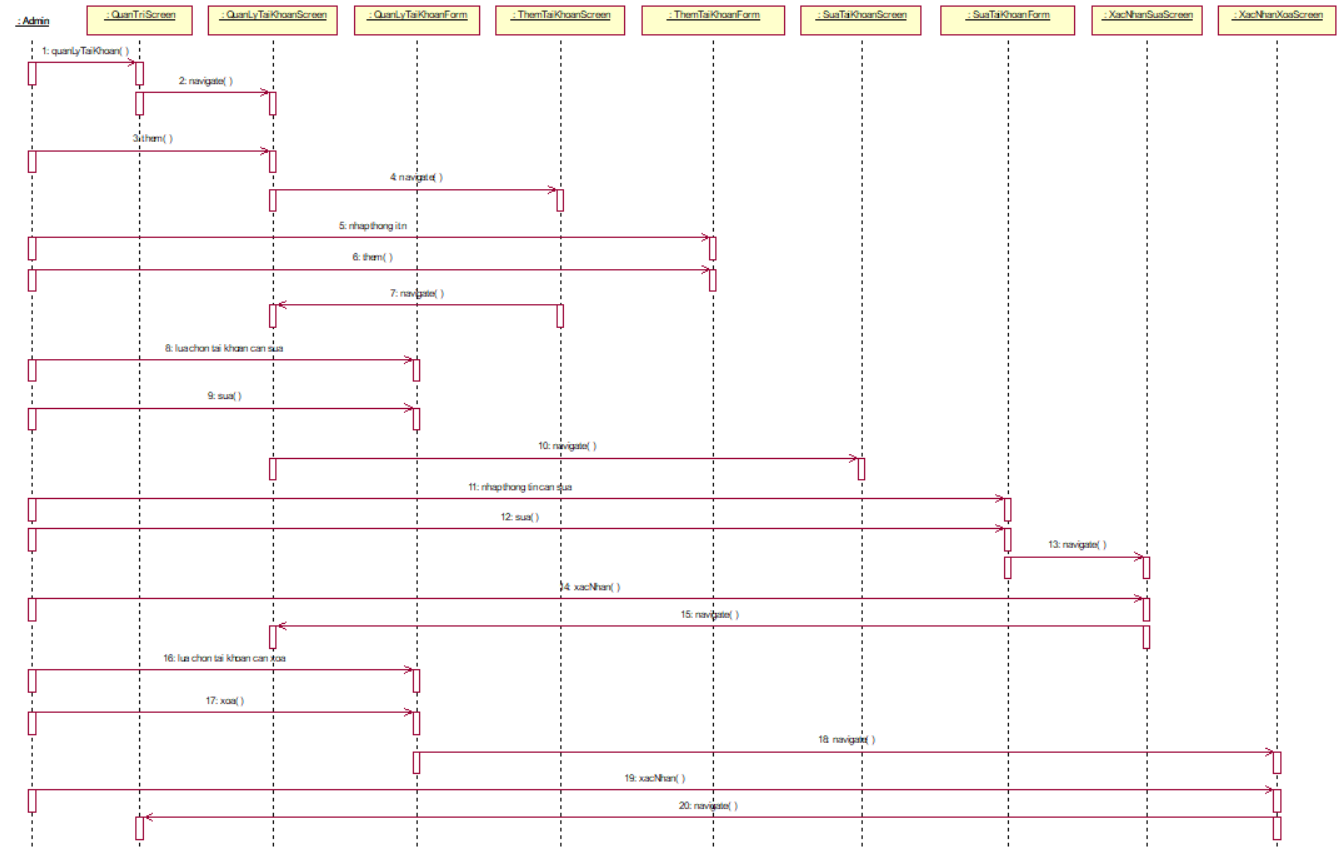
Hình dung màn hình UC Quản lý tài khoản

1. **Biểu đồ lớp màn hình**

****

Biểu đồ lớp màn hình UC Quản lý tài khoản

1. **Biểu đồ cộng tác màn hình**

****

Biểu đồ cộng tác màn hình UC Quản lý tài khoản