TRƯỜNG ĐẠI HỌC SÀI GÒN

Khoa Công nghệ thông tin

BÁO CÁO ĐỒ ÁN – LẦN 1
KIỂM THỬ PHẦN MỀM

ĐỀ TÀI: PHÂN TÍCH YÊU CẦU NGHIỆP VỤ - WEBSITE QUẢN LÝ BÁN QUẦN ÁO MQTT

GV hướng dẫn: **ThS. Đỗ Như Tài**

Lớp: **DCT122C3 – Mã học phần: 841408**

Thành viên: **Lê Song Nhật Quyền   – 3122411174**

Nguyễn Lê Nhật Minh   – 3122411125

Trần Minh Trí   – 3122411222

Bùi Văn Tiến   – 3122411206

Thành phố Hồ Chí Minh, tháng 9 năm 2025

BẢNG PHÂN CÔNG CỦA NHÓM

Họ tên các thành viên thực hiện	Nội dung công việc	Tiến độ công việc
Lê Song Nhật Quyền (Trưởng nhóm)	Làm phần **Introduction**	100%
	Làm phần **Business Context**	100%
	Làm phần **Conceptual Model**	100%
	Làm phần vẽ các **Use Case**	100%
	Làm phần **User Story**	100%
Trần Minh Trí		0%
Nguyễn Lê Nhật Minh		0%
Bùi Văn Tiến		0%

LỜI CAM ĐOAN

Tôi tên là Lê Song Nhật Quyền, xin đại diện nhóm chịu trách nhiệm và cam đoan rằng:

Những kết quả nghiên cứu được trình bày trong bài tiểu luận là công trình của riêng chúng tôi dưới sự hướng dẫn của giảng viên ThS. Đỗ Như Tài

Chúng tôi đã không sao chép bất kỳ thông tin nào từ các nguồn khác mà không được ghi nhận. Chúng tôi cam đoan không vi phạm bất kỳ quyền sở hữu trí tuệ hoặc quyền tác giả của bất kỳ ai hoặc bất kỳ tổ chức nào.

Tôi cam đoan rằng những kết quả và nhận định đưa ra trong bài báo cáo là sự hiểu biết và đánh giá của chúng tôi dựa trên nghiên cứu tài liệu và kiến thức về … . Chúng tôi đã cố gắng hết sức để cung cấp thông tin đầy đủ về các kiến thức của học phần kiểm thử phần mềm được đề cập trong bài. Chúng tôi cam đoan rằng bài tiểu luận này được thực hiện một cách độc lập và khách quan.

Xin chân thành cảm ơn
Sinh viên thực hiện

Lê Song Nhật Quyền
Nguyễn Lê Nhật Minh
Trần Minh Trí
Bùi Văn Tiến

DANH MỤC HÌNH

MỤC LỤC

INTRODUCTION

MQTTStore Website is a modern, containerised polyglot microservices application designed for the **Java ecosystem**. The platform powers a **clothing e-commerce website**, combining **Spring Boot**, **JSP/Servlets**, and a wide range of Java libraries, including utilities for handling **date, time, and calendar operations**.

At its core, the system leverages **JDBC with MySQL** for reliable database connectivity, and employs **Hibernate/JPA** to simplify persistence, reduce boilerplate, and ensure clean integration with relational data. For API development and testing, **Postman** is used extensively for calling API requests, while the **frontend is built with ReactJS**, delivering a dynamic and responsive shopping experience for users.

The entire application runs on **Docker**, enabling consistent deployments, scalability, and smooth integration across services. This architecture showcases how multiple Java-based microservices can be wired together into a larger, cloud-native application that embraces **modern microservice design principles** while remaining deeply rooted in the proven reliability of the **Java community's libraries and toolkits**.

BUSINESS CONTEXT

MTTQStore Website provides a **comprehensive set of business scenarios** that cover every stage of the fashion clothing industry and online shopping experiences, from browsing products to managing post-purchase activities.

● **Product Catalog**

Buyers can explore a rich catalog of clothing products with powerful **filtering and sorting** options by name, price, category, and even size or color. Each product detail page provides essential information such as product name, description, available quantity in stock, store location of the inventory, whether the product is marked as **Hot/Trending**, and its **customer rating**.

● **Shopping Cart**

Buyers can seamlessly add products to their shopping cart directly from either the product listing or the detail page. The cart dynamically updates with every change such as adding or removing products while showing real-time values such as subtotal, promotions, discounts, shipping costs,

and the final order total. Buyers can proceed to checkout with a single click, while SysAdmins maintain the ability to monitor and disable any invalid or fraudulent carts.

- **Payment Process**

When the Buyer clicks the **Checkout** button, the system validates product availability, securely processes payments, and automatically sends an **email notification** confirming the purchase. This ensures transparency and builds customer trust.

- **Inventory Management**

SysAdmins can manage multiple warehouses or store inventories, tracking stock levels down to each product's size and color variation. Low-stock alerts help prevent shortages, and bulk update tools allow easy import/export of inventory data. Additionally, the system provides visibility into **inventory turnover**, ensuring stock is efficiently managed and buyers always see accurate availability.

- **Rating System**

Buyers can provide feedback by rating purchased products from **1 to 5 stars** and leaving optional comments. These ratings not only help future customers make informed choices but also provide SysAdmins with valuable insights into product quality, customer satisfaction, and potential improvements. Ratings can be aggregated into analytics dashboards to highlight **best sellers** or identify products with poor performance.

- **Promotion & Voucher Management**

The system allows SysAdmins to create and manage **seasonal promotions, discount campaigns, and special vouchers** for loyal buyers. Buyers can apply vouchers during checkout and instantly see their savings reflected in the shopping cart summary.

- **Loyalty Points Management**

To encourage repeat purchases, Buyers earn **loyalty points** for each successful transaction. Points can later be redeemed for discounts, free shipping, or exclusive rewards, strengthening long-term customer engagement and brand loyalty.

- **Employee Management**

The platform also includes an **Employee Management module**, where SysAdmins can add, update, or remove employees, assign roles (e.g., cashier, warehouse staff, manager), and control their access permissions. This ensures secure and structured internal operations.

● **Access Control**

Both Buyers and SysAdmins authenticate securely through the **Access Control module**. Buyers are redirected to the product catalog upon login, while SysAdmins access an **administrative dashboard** with advanced tools for system monitoring, product assignments, and employee oversight.

● **Initial Setup**

When the MQTTStore website is first launched, the system seeds default data, including:

- One SysAdmin account,

- Two sample Buyer accounts,

- Preloaded sample data for products, inventories, ratings, promotions, and vouchers.

CONCEPTUAL MODEL
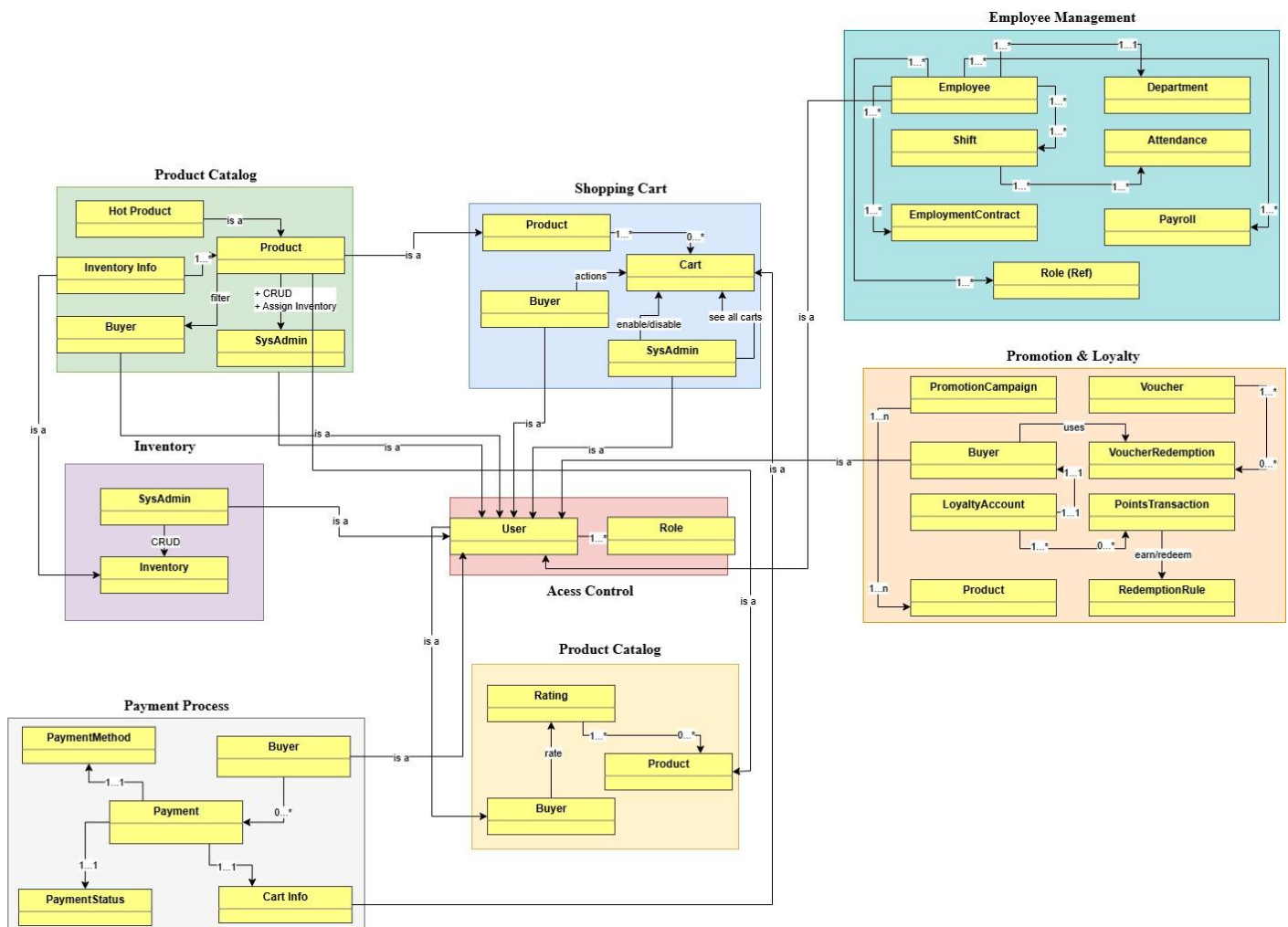
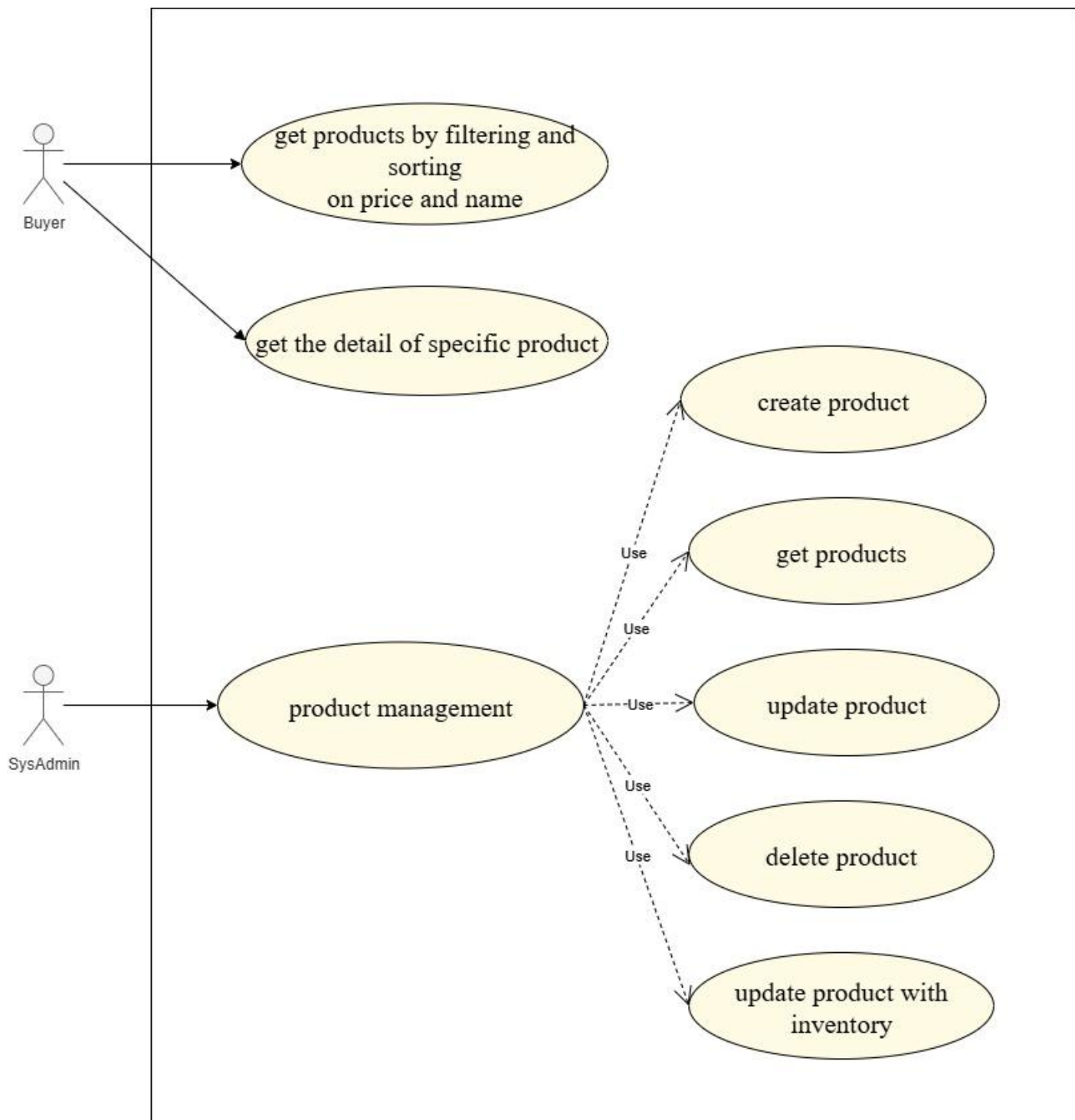*Hình 1. Simulate the website design through Conceptual Model*
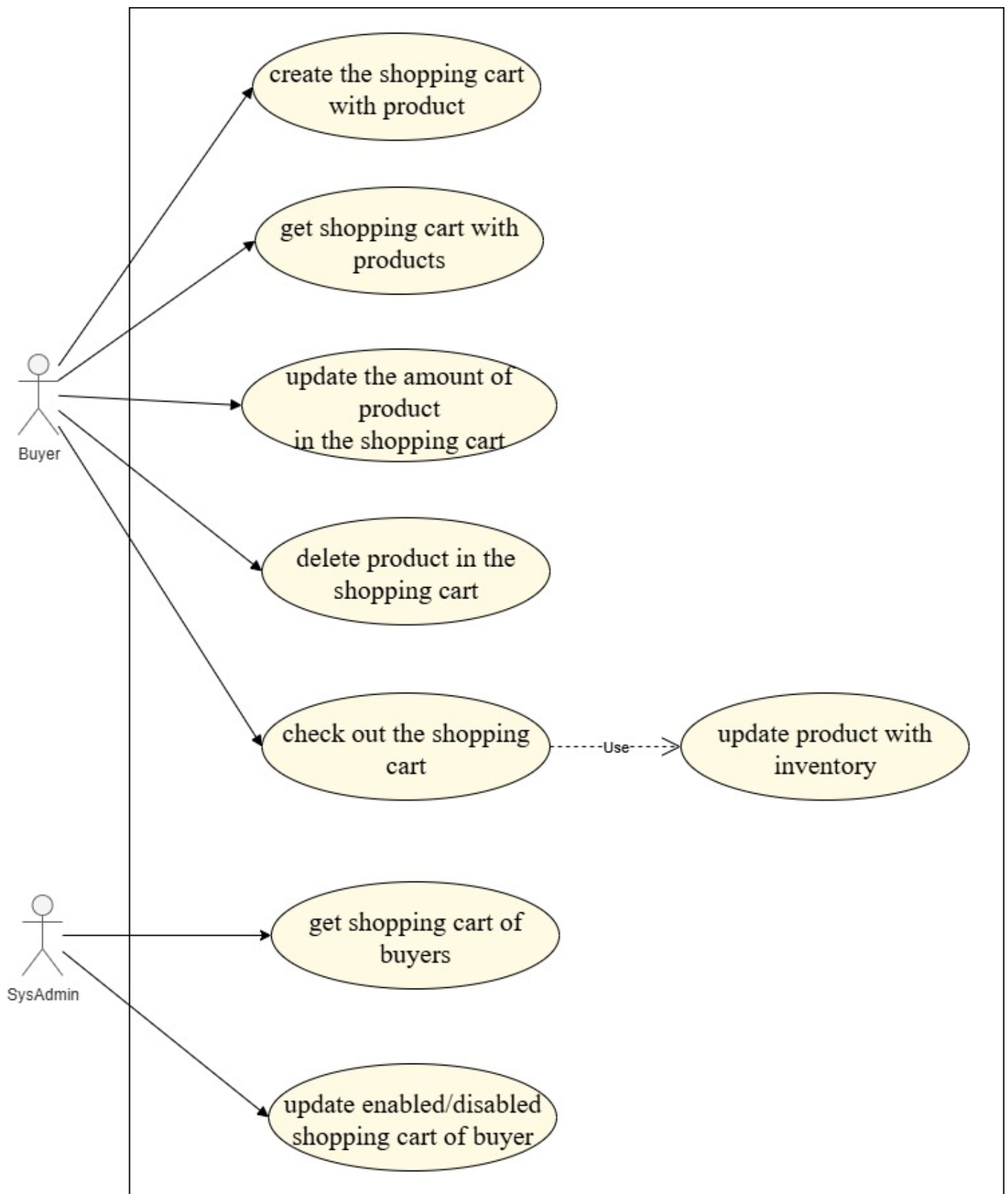
USE CASE

Use Case Summary

Hình 2. *Draw the usecase of all sub-functions*

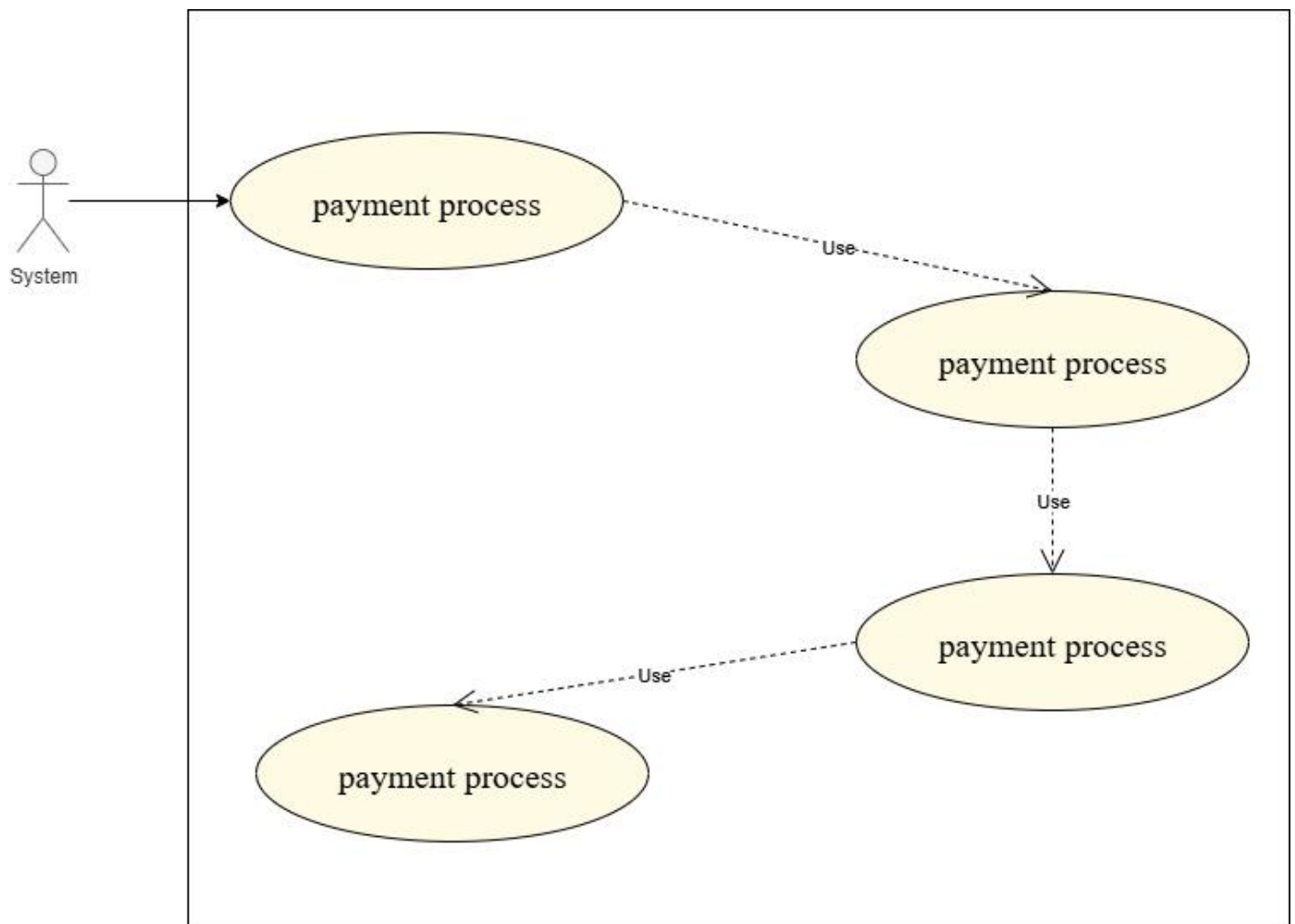   **Use Case: Product Catalog**

Hình 3. Draw the usecase of Product Catalog

Use Case: Shopping Cart

Hình 4. *Draw the usecase of Shopping Cart*

Use Case: Payment Process

Hình 5. *Draw the usecase of Payment Process*

Use Case: Inventory Management
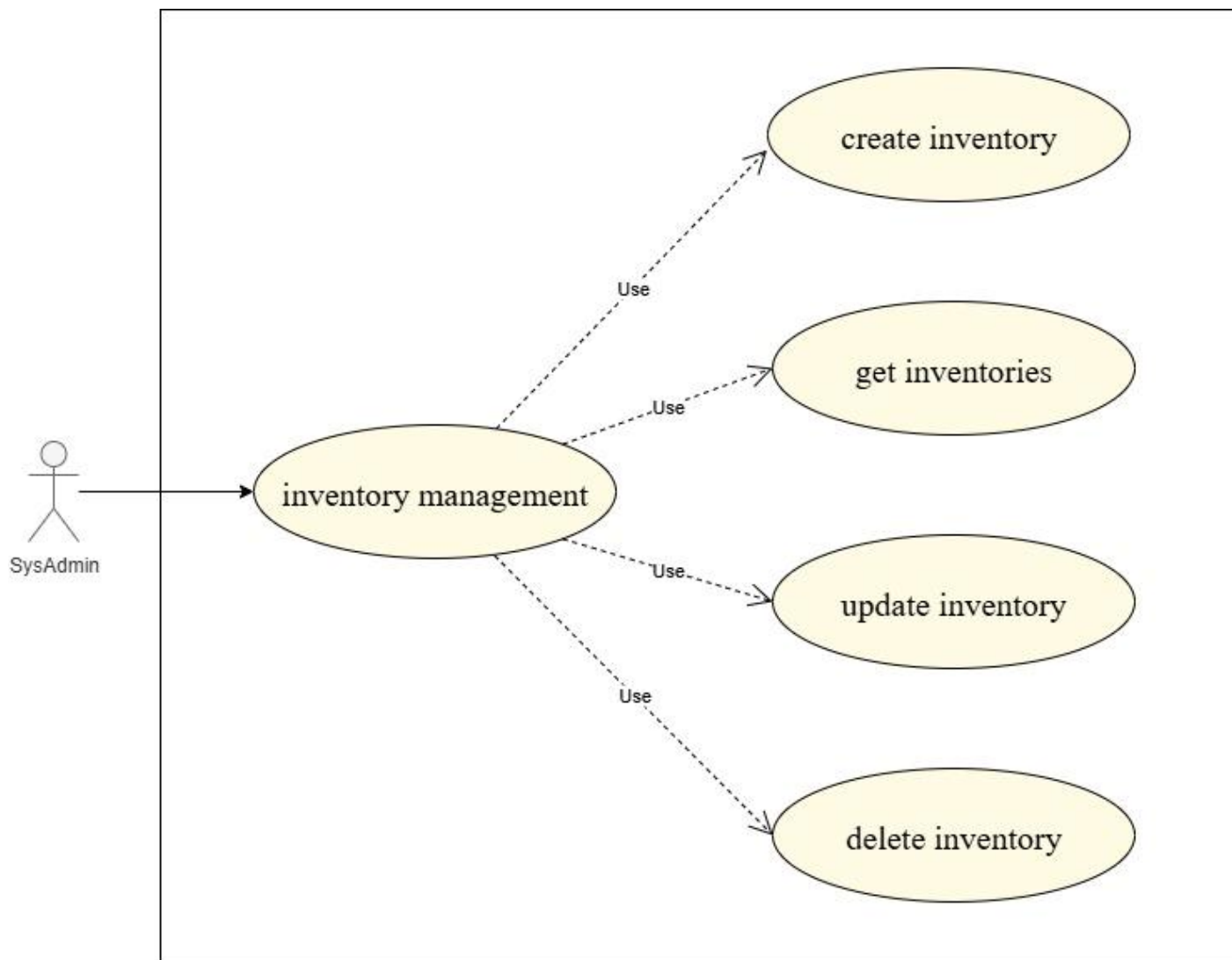
Hình 6. *Draw the usecase of Inventory*

Use Case: Rating System
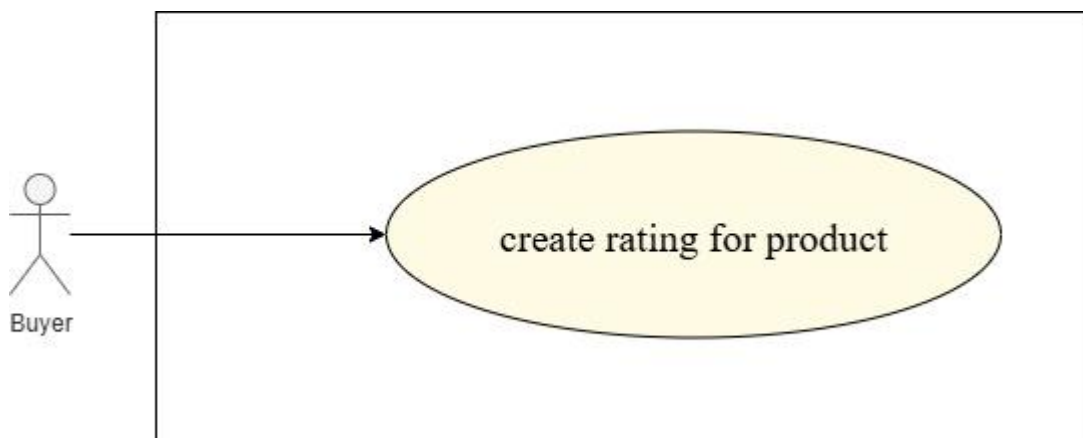
Hình 7. *Draw the usecase of Rating*

Use Case: Acess Controll

Hình 8. *Draw the usecase of Acess Controll*

Use Case: Promotion/Voucher/Loyalty Points Management

Hình 9. *Draw the usecase of Promotion/Voucher/Loyalty Points Management*

Use Case: Employee/Staff Management

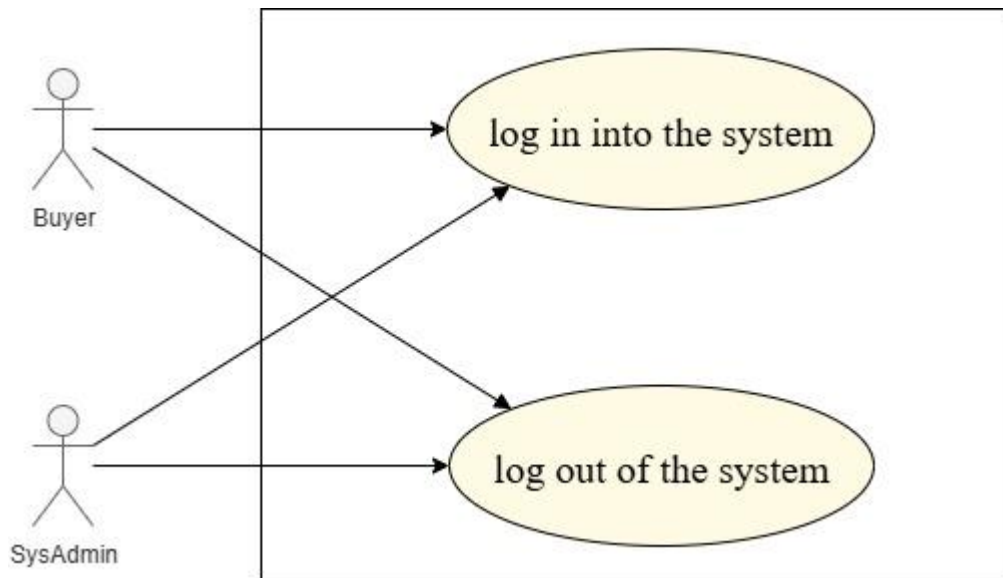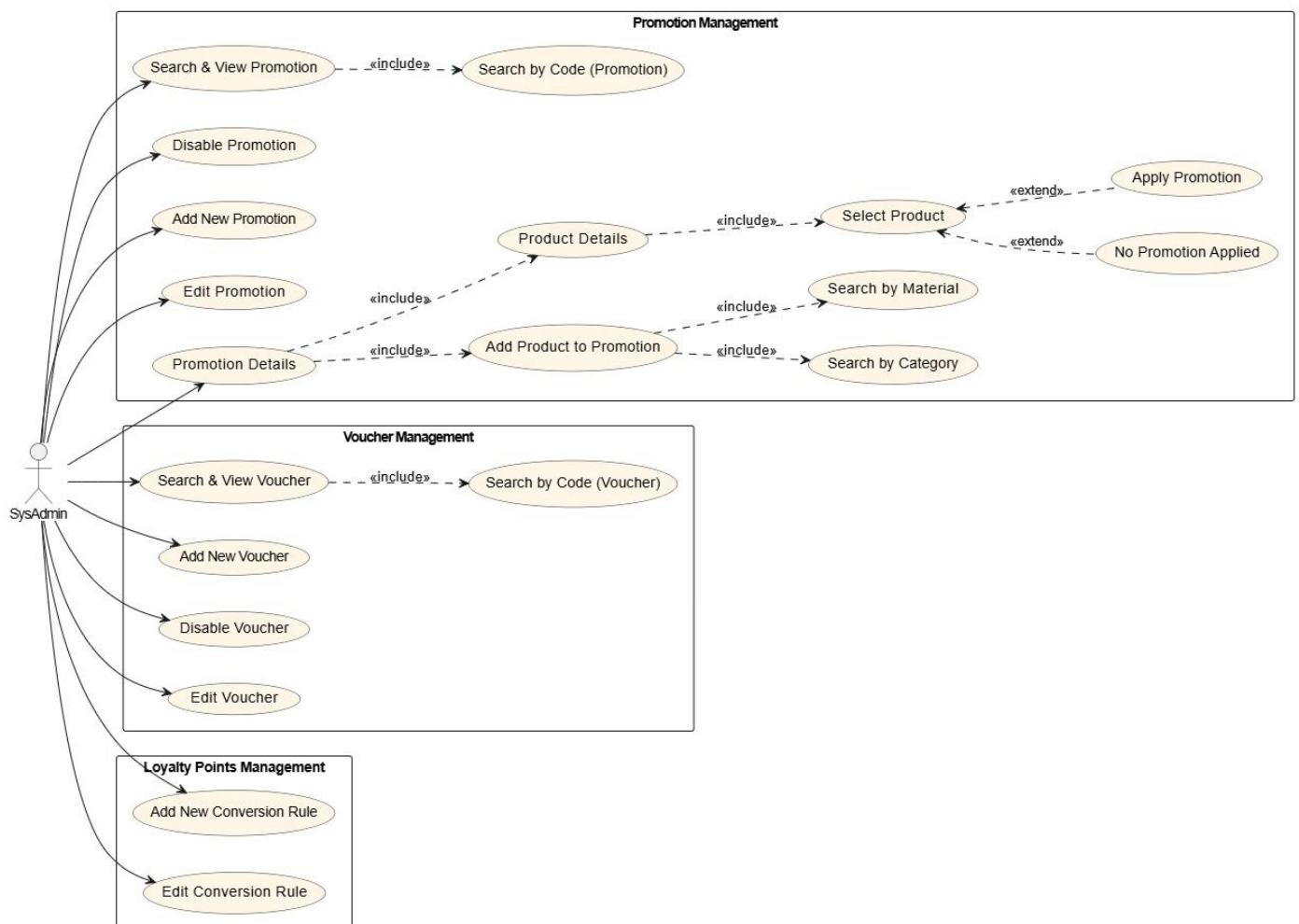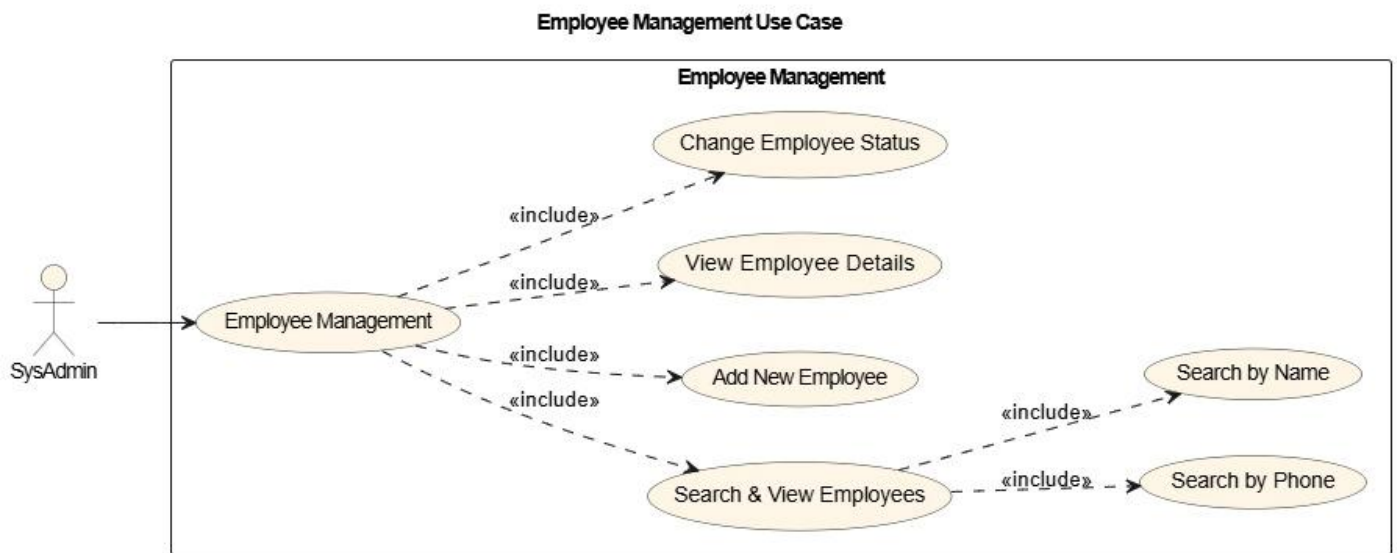***Hình 10.*** *Draw the usecase of Employee/Staff Management*

USER STORY

User Stories – Product Catalog

As a Buyer

- I want to view the **list of available clothing products** on the home page, with key attributes including **name, photo, short description, rating, and a "Hot Product" badge** (indicating popular or frequently purchased items).

- I want to **filter** the product list by criteria such as **price range** and **product name**, so that I can quickly narrow down to items that match my preferences.

- I want to **sort** products by **name or price** in both ascending and descending order, so that I can easily find what I'm looking for.

- When **both filtering and sorting** are applied, I want the product list to reflect both conditions consistently.

- I want to **navigate into the detail page** of a selected product, where I can see essential attributes such as:

    - Product name and description,
    - Availability in inventory (quantity in stock),
    - Store or warehouse location of the product,

- Hot Product badge (if applicable),
- Customer rating.

As a SysAdmin

- I want to **manage products through CRUD operations** (Create, Read, Update, Delete) to ensure the catalog is always up-to-date.
- I want to **assign products to an existing inventory**, so that stock levels and availability remain accurate across the system.

User Stories – Shopping Cart

As a Buyer

- I want to **add products to my shopping cart** directly from the product catalog page (by default, one unit will be added when I click "Buy").
- I want to **add a product from its detail page** if I like it, with the same default behavior of adding one unit to the cart.
- I want to **view all the products currently in my shopping cart**, along with a **summary panel** that shows:
  - Cart total cost,
  - Promotion item saving cost,
  - Subtotal cost,
  - Shipping cost,
  - Promotion shipping savings cost,
  - Final order amount.
- I want to **update the quantity** of any product in my cart, and whenever I make this change, the **summary panel should update automatically** to reflect the new totals.
- I want to **remove any product** from my shopping cart that I no longer wish to purchase, and the **summary panel should update immediately** after the deletion.
- I want to **proceed to checkout** when I am satisfied with my cart.
  - If my shopping cart is empty (no products), then the **checkout process should not be allowed**.
  - Once the checkout process is confirmed, the **payment workflow begins** automatically.

As a SysAdmin

- I want to **view the shopping carts of all buyers**, including detailed cost information such as cart total, promotion savings, subtotal, shipping cost, shipping promotion savings, and final order amount.

- I want to **enable or disable any shopping cart** in the system to ensure invalid or fraudulent carts are handled properly.

User Stories - Payment Process

As a Buyer

- I want to complete the **payment process** after checking out my shopping cart, so that I can finalize my order.

- During the payment process, the system must **validate product information** (availability, stock quantity, price, and promotions) before processing payment.

- If all information is valid, the system will **process the payment** and send me an **email confirmation** with the details of my purchase.

- If any product information is **invalid or inconsistent**, the system should **cancel the payment**, mark the transaction as failed, and send me an **email notification** explaining the issue.

- After successful payment, the system marks the cart with a **"Payment Processed" status** and sends me a **final email receipt**.

- (For demo purposes, the system simulates the payment process instead of integrating with a real payment gateway.)

User Stories - Inventory Management

As a SysAdmin

- I want to **manage inventories** with full **CRUD operations (Create, Read, Update, Delete)**, so that stock data is always accurate and reliable.

- I want to **track product quantities** across multiple warehouses or store locations, ensuring Buyers always see **real-time availability**.

- I want to configure **low-stock alerts** so I can proactively restock popular clothing items.

- I want to be able to **import/export inventory data in bulk** (e.g., via CSV/Excel upload), saving time when managing a large clothing catalog.

- I want to **assign products to specific inventories** (e.g., a particular warehouse or branch), ensuring location-based fulfillment and accurate delivery.
- I want to view **inventory analytics** such as stock turnover rate, most frequently restocked products, and slow-moving items, to optimize inventory management.

User Stories - Rating System

As a Buyer

- I want to **rate products I have purchased** on a scale of **1 to 5 stars**, so that I can share my feedback and help other customers make informed decisions.
- I want to optionally **write a short review** along with my rating, describing what I liked or disliked about the product (e.g., quality, size fit, material).
- I want to see the **average rating and total number of reviews** for each product, so I can evaluate its popularity and trustworthiness.
- I want to be able to **update or delete my rating/review**, in case I change my mind after further use of the product.
- **As a SysAdmin**, I want to **moderate product ratings and reviews**, so that inappropriate or fraudulent feedback can be removed.
- Ratings should contribute to **analytics dashboards**, identifying **best-selling and top-rated clothing items**, as well as **products with consistently poor feedback** that may need quality improvement.

User Stories - Access Control

- Each Buyer and SysAdmin is a **registered User** in the system.
- **As a Buyer or SysAdmin**, I want to **log in securely** with my credentials.
- If I log in as a **Buyer**, I should be redirected to the **Product Catalog page**, where I can start browsing and shopping.
- If I log in as a **SysAdmin**, I should be redirected to the **Administration Dashboard**, where I can manage products, inventories, employees, and system settings.
- **As a Buyer or SysAdmin**, I want to **log out at any time**, ensuring my session is securely terminated.

- The system should enforce **role-based access control (RBAC)**, ensuring that Buyers cannot access administrative features and SysAdmins have the necessary permissions to manage the system.

User Stories - Promotion, Voucher, and Loyalty Points Management

Promotion Management

- **As a SysAdmin**, I want to **create, update, and delete promotions** (e.g., seasonal sales, buy-one-get-one, holiday discounts), so that customers can enjoy special offers and sales campaigns.
- **As a Buyer**, I want to **see which products are under promotion** directly on the catalog and product detail pages, so I can take advantage of current discounts.
- **As a SysAdmin**, I want to **schedule promotions** to start and end on specific dates, ensuring campaigns run automatically without manual intervention.

Voucher Management

- **As a SysAdmin**, I want to **generate and distribute vouchers** (e.g., discount codes, gift cards), so that Buyers can use them during checkout.
- **As a Buyer**, I want to **apply a voucher code** in the shopping cart or checkout page, so I can see my savings immediately reflected in the order total.
- **As a SysAdmin**, I want to **track voucher usage** to ensure each code is valid, not expired, and not misused.

Loyalty Points Management

- **As a Buyer**, I want to **earn loyalty points** for each purchase I make, so I feel rewarded for shopping more frequently.
- **As a Buyer**, I want to **redeem loyalty points** for discounts, free shipping, or special rewards during checkout.
- **As a SysAdmin**, I want to **configure the loyalty point rules** (e.g., how many points per $1 spent, expiration dates, redemption policies), ensuring flexibility in customer rewards.
- **As a SysAdmin**, I want to **monitor customer loyalty balances**, so I can identify high-value customers and tailor promotions accordingly.

User Stories - Employee / Staff Management

Employee Records

- **As a SysAdmin**, I want to **add, update, and remove employee profiles** (name, role, contact info, assigned department), so the staff directory stays accurate.

- **As a SysAdmin**, I want to **assign roles** (e.g., cashier, inventory manager, marketing staff, system admin), so each employee has clear responsibilities and system permissions.

- **As a SysAdmin**, I want to **suspend or reactivate employees**, ensuring that only active staff members have access to the system.

Access & Permissions

- **As a SysAdmin**, I want to **set access control levels** for employees, so they can only use the features necessary for their role (e.g., inventory staff cannot manage promotions).

- **As an Employee**, I want to **log in with my staff account**, so I can perform my assigned tasks securely.

Employee Monitoring

- **As a SysAdmin**, I want to **view logs of employee activities** (e.g., product updates, inventory changes, voucher creations), so I can track accountability and system usage.

- **As a SysAdmin**, I want to **generate reports** on employee performance (e.g., number of orders processed, inventory adjustments made), to evaluate efficiency and contributions.

TÀI LIỆU THAM KHẢO

[1] Tài liệu bài giảng của giảng viên Ths. Đỗ Như Tài cung cấp trên lớp