



TRƯỜNG ĐẠI HỌC SÀI GÒN

Khoa Công nghệ thông tin

BÁO CÁO BÀI TẬP Lab 2

KIỂM THỬ PHẦN MỀM

ĐỀ BÀI: PHÂN TÍCH THIẾT KẾ HỆ THỐNG

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

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)	Bài 3. Vẽ lại business use case	100%
	Bài 5. Vẽ lược đồ lớp cho bài toán Quản lý thư viện.	100%
	Câu 1, 3, 9 phần Bài tập ứng dụng (Website CoolStore)	100%
Trần Minh Trí	Bài 2. Vẽ mô hình khái niệm sau	100%
	Bài 4. Vẽ sơ đồ hoạt động và tương tác sau và tóm tắt ý nghĩa sơ đồ	100%
	Câu 2, 5, 7 phần Bài tập ứng dụng (Website CoolStore)	100%
Nguyễn Lê Nhật Minh	Bài 1. Vẽ lại qui trình nghiệp vụ sau (Độc giả, NV Tiếp tân, Thành viên)	100%
	Bài 6. Vẽ lược đồ sau:	100%
	Câu 4, 6, 8 phần Bài tập ứng dụng (Website CoolStore)	90%
Lê Song Nhật Quyền Nguyễn Lê Nhật Minh Trần Minh Trí	Truy vấn SQL cho câu 2 Bài 7: Phân tích dữ liệu cho ABC Bank	100%
	Viết báo cáo, hoàn thiện báo cáo	100%
	Xây dựng dữ liệu và chạy thử	100%

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

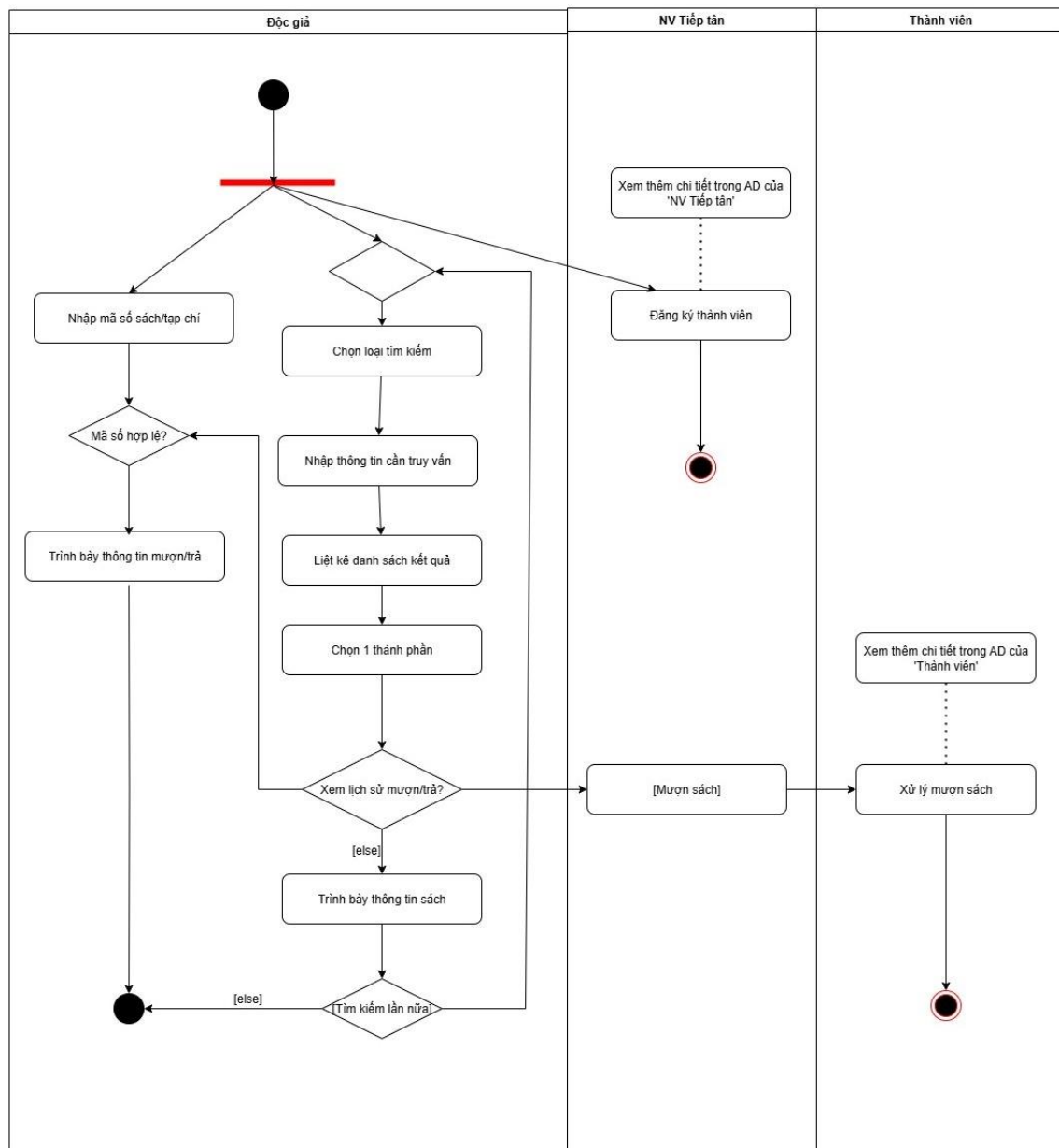
DANH MỤC HÌNH

HÌNH 1. BT VỀ LẠI CHO QUY TRÌNH NGHIỆP VỤ TRONG MƯỢN SÁCH HOẶC TẠP CHÍ	6
HÌNH 2. BT VỀ LẠI CHO MÔ HÌNH DỮ LIỆU Ở MỨC KHÁI NIỆM TRONG THUÊ XE CON	7
HÌNH 3. BT VỀ LẠI CHO BUSINESS USECASE TRONG ĐĂNG KÝ HỌC PHẦN.....	8
HÌNH 4. BT VỀ LẠI CHO ACTIVITY DIAGRAM TRONG QUY TRÌNH THANH TOÁN TIỀN CHO SẢN PHẨM.....	9
HÌNH 5. BT VỀ LẠI CHO SEQUENCE DIAGRAM TRONG QUY TRÌNH THANH TOÁN TIỀN CHO SẢN PHẨM.....	10
HÌNH 6. BT VỀ LẠI CHO CLASS DIAGRAM TRONG QUẢN LÝ THƯ VIỆN.....	11
HÌNH 7. BT VỀ LẠI CHO ACTIVITY DIAGRAM TRONG QUY TRÌNH NẠP/RÚT TIỀN TẠI ATM.....	12
HÌNH 8. BT VỀ LẠI CHO ACTIVITY DIAGRAM TRONG QUY TRÌNH KIỂM TRA TÍNH HỢP LỆ CỦA THẺ TẠI ATM	13
HÌNH 9. BIỂU DIỄN MÔ HÌNH DỮ LIỆU Ở MỨC KHÁI NIỆM	14
HÌNH 10. BIỂU DIỄN MÔ HÌNH DỮ LIỆU Ở MỨC LUẬN LÝ (LOGICAL).....	15
HÌNH 11. BIỂU DIỄN MÔ HÌNH DỮ LIỆU Ở MỨC VẬT LÝ (PHYSICAL)	16
HÌNH 12. HÌNH VẼ QUY TRÌNH NGHIỆP VỤ TẠI WEBSITE BÁN QUẦN ÁO THỜI TRANG	26
HÌNH 13. HÌNH VẼ USECASE DIAGRAM TỔNG QUAN CỦA WEBSITE COOLSTORE	27
HÌNH 14. HÌNH VẼ USECASE CHO XỬ LÝ CÁC DANH MỤC SẢN PHẨM.....	28
HÌNH 15. HÌNH VẼ USECASE CHO PHẦN XỬ LÝ GIỎ HÀNG	29
HÌNH 16. HÌNH VẼ USECASE CHO QUY TRÌNH THANH TOÁN TRỰC TUYẾN	30
HÌNH 17. HÌNH VẼ USECASE CHO QUY TRÌNH XỬ LÝ HÀNG TỒN KHO	31
HÌNH 18. HÌNH VẼ CHO USECASE PHẦN XỬ LÝ ĐÁNH GIÁ CỦA NGƯỜI MUA.....	31
HÌNH 19. HÌNH VẼ USECASE CHO XỬ LÝ ĐĂNG NHẬP/ĐĂNG XUẤT TRÊN HỆ THỐNG CỦA CỬA HÀNG	32
HÌNH 20. HÌNH VẼ ACTIVITY DIAGRAM CHO PHẦN XỬ LÝ MUA HÀNG.....	37
HÌNH 21. HÌNH VẼ CLASS DIAGRAM CHO PHẦN XỬ LÝ GIỎ HÀNG.....	38
HÌNH 22. HÌNH VẼ THIẾT KẾ PHẦN GIAO DIỆN CHO CHỨC NĂNG QUẢN LÝ GIỎ HÀNG	39

MỤC LỤC

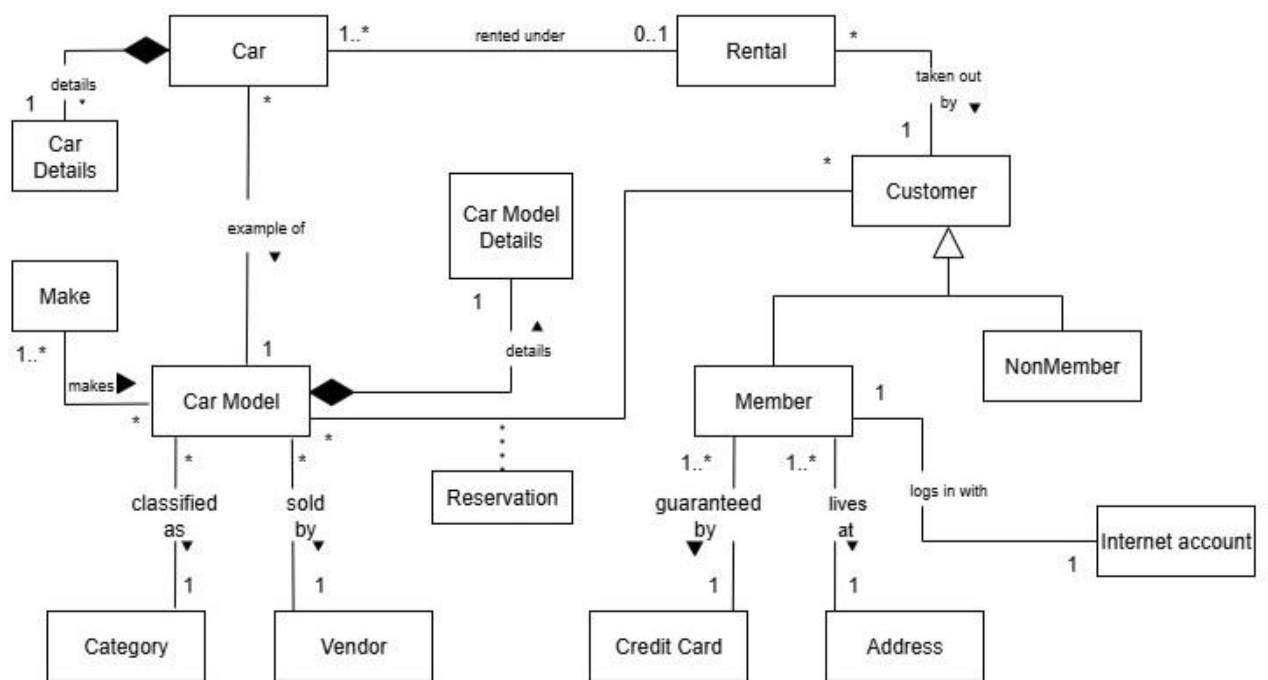
BÀI 1. VẼ LẠI QUI TRÌNH NGHIỆP VỤ SAU (ĐỘC GIẢ, NV TIẾP TÂN, THÀNH VIÊN). 6	
BÀI 2. VẼ MÔ HÌNH KHÁI NIỆM SAU..... 7	
BÀI 3. VẼ LẠI BUSINESS USE CASE..... 8	
BÀI 4. VẼ SƠ ĐỒ HOẠT ĐỘNG VÀ TƯƠNG TÁC SAU VÀ TÓM TẮT Ý NGHĨA SƠ ĐỒ.. 9	
BÀI 5. VẼ LƯỢC ĐỒ LỚP CHO BÀI TOÁN QUẢN LÝ THƯ VIỆN. 11	
BÀI 6. VẼ LƯỢC ĐỒ SAU:..... 12	
BÀI 7. PHÂN TÍCH DỮ LIỆU CHO ABC BANK..... 14	
1. THIẾT KẾ CƠ SỞ DỮ LIỆU Ở MỨC KHÁI NIỆM, LUẬN LÝ VÀ VẬT LÝ..... 14	
2. QUERY: INSERT, UPDATE, DELETE, SELECT 17	
BÀI TẬP ỨNG DỤNG 21	
1. HÃY LIỆT KÊ CÁC YÊU CẦU CHỨC NĂNG VÀ PHI CHỨC NĂNG CỦA HỆ THỐNG COOLSTORE DỰA TRÊN MÔ TẢ KỊCH BẢN NGHIỆP VỤ. 21	
2. VẼ QUI TRÌNH NGHIỆP VỤ CHO WEBSITE COOLSTORE..... 25	
3. XÂY DỰNG USE CASE DIAGRAM CHO WEBSITE COOLSTORE..... 27	
4. MÔ TẢ CHI TIẾT MỘT USE CASE "MUA SẢN PHẨM TỪ TRANG CHI TIẾT SẢN PHẨM". TRÌNH BÀY THEO MẪU CHUẨN GỒM: TÊN USE CASE, TÁC NHÂN CHÍNH, MỤC TIÊU, TIỀN ĐIỀU KIỆN, LƯỒNG CHÍNH, LƯỒNG THAY THẾ, HẬU ĐIỀU KIỆN, GHI CHÚ (NẾU CÓ). 32	
5. SỬ DỤNG SƠ ĐỒ ACTIVITY DIAGRAM ĐỂ MÔ TẢ QUY TRÌNH MUA HÀNG TỪ KHI NGƯỜI DÙNG NHẤN “MUA HÀNG” CHO ĐẾN KHI HOÀN TẤT THANH TOÁN. 36	
6. THIẾT KẾ SƠ ĐỒ CLASS DIAGRAM CHO MODULE GIỎ HÀNG. 37	
7. THIẾT KẾ GIAO DIỆN MÀN HÌNH CHO CHỨC NĂNG “QUẢN LÝ GIỎ HÀNG” (CÓ THỂ THIẾT KẾ MỘT HOẶC NHIỀU MÀN HÌNH) GỒM CÁC THÔNG TIN: 38	
8. ĐỀ XUẤT CÁCH TỔ CHỨC KIẾN TRÚC PHẦN MỀM CHO HỆ THỐNG COOLSTORE THEO MÔ HÌNH KIẾN TRÚC BA LỚP (3-TIER ARCHITECTURE)..... 40	
9. VIẾT ÍT NHẤT 3 CA KIỂM THỬ (TEST CASES) CHO CHỨC NĂNG “THÊM SẢN PHẨM VÀO GIỎ HÀNG” – BAO GỒM DỮ LIỆU ĐẦU VÀO, BƯỚC THỰC HIỆN VÀ KẾT QUẢ MONG ĐỢI. 42	
TÀI LIỆU THAM KHẢO 45	

Bài 1. Vẽ lại quy trình nghiệp vụ sau (Độc giả, NV Tiếp tân, Thành viên)



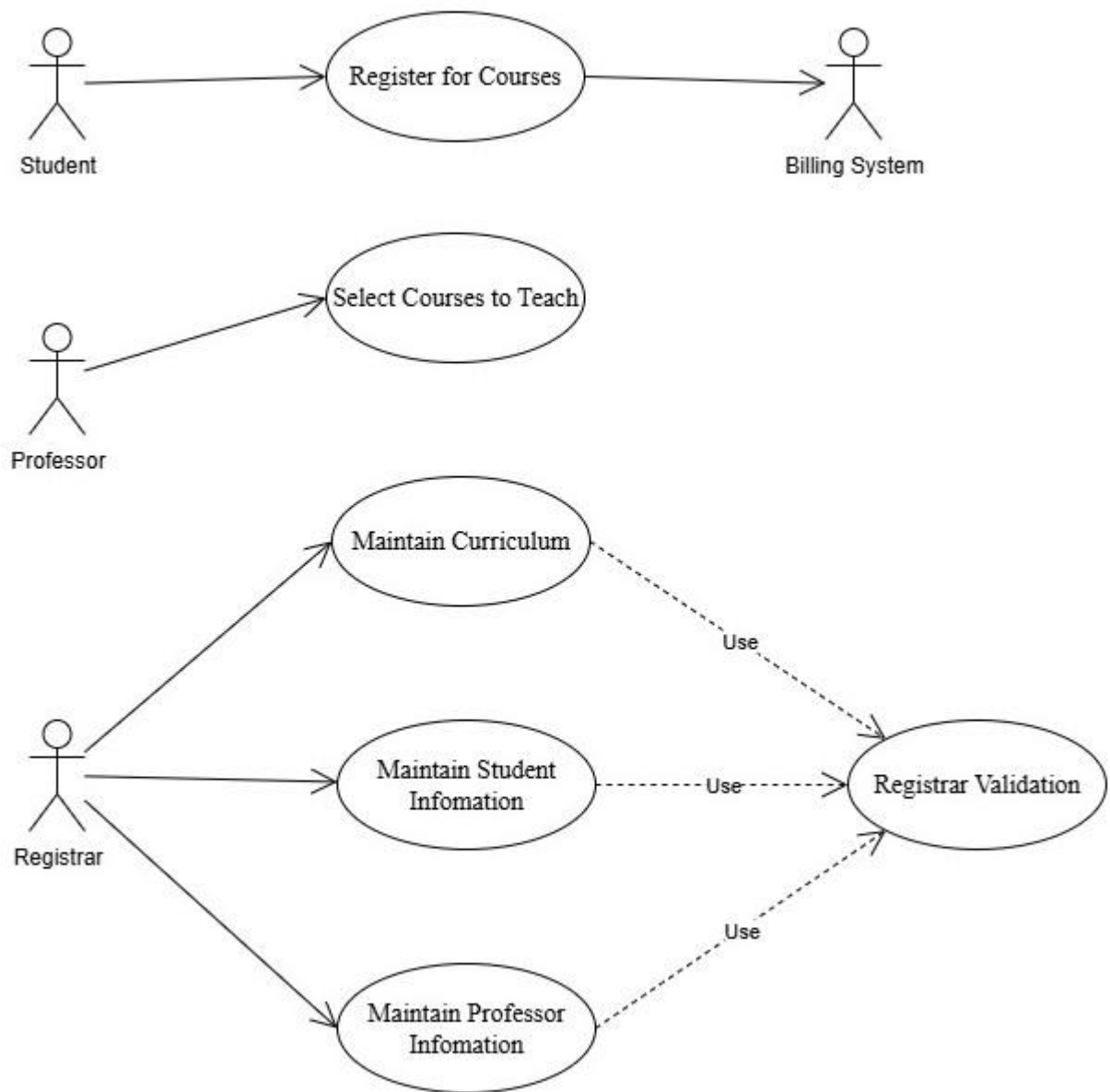
Hình 1. BT vẽ lại cho Quy trình nghiệp vụ trong mượn sách hoặc tạp chí

Bài 2. Vẽ mô hình khái niệm sau



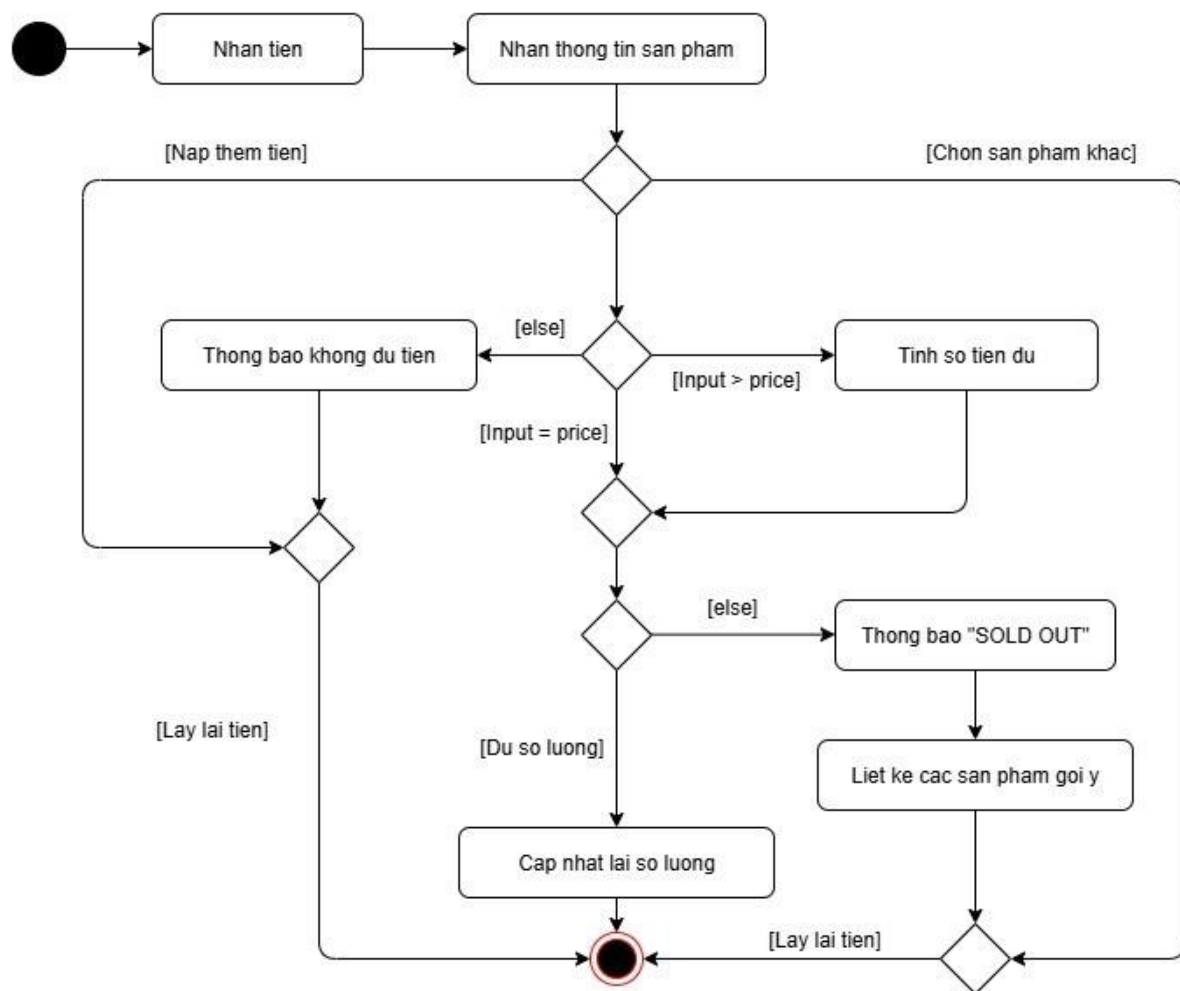
Hình 2. BT vẽ lại cho Mô hình dữ liệu ở mức khái niệm trong thuê xe con

Bài 3. Vẽ lại business use case

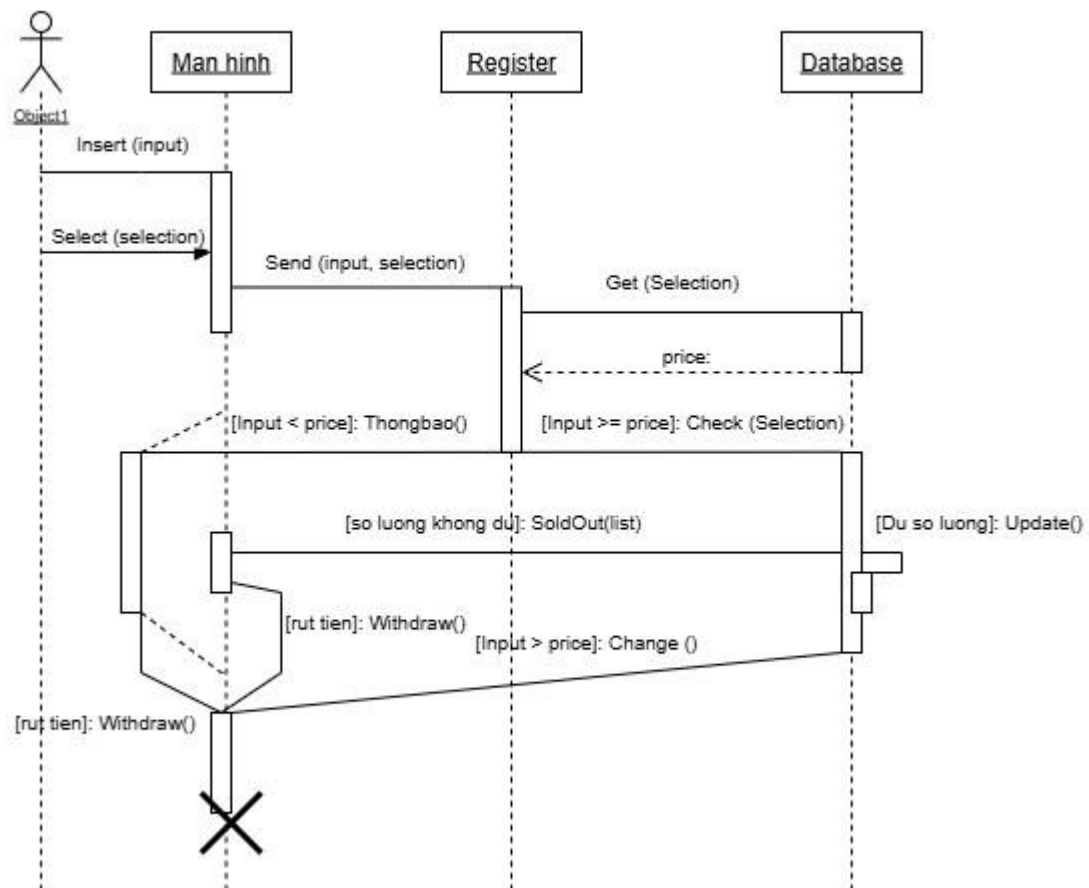


Hình 3. BT vẽ lại cho Business UseCase trong đăntg ký học phần

Bài 4. Vẽ sơ đồ hoạt động và tương tác sau và tóm tắt ý nghĩa sơ đồ

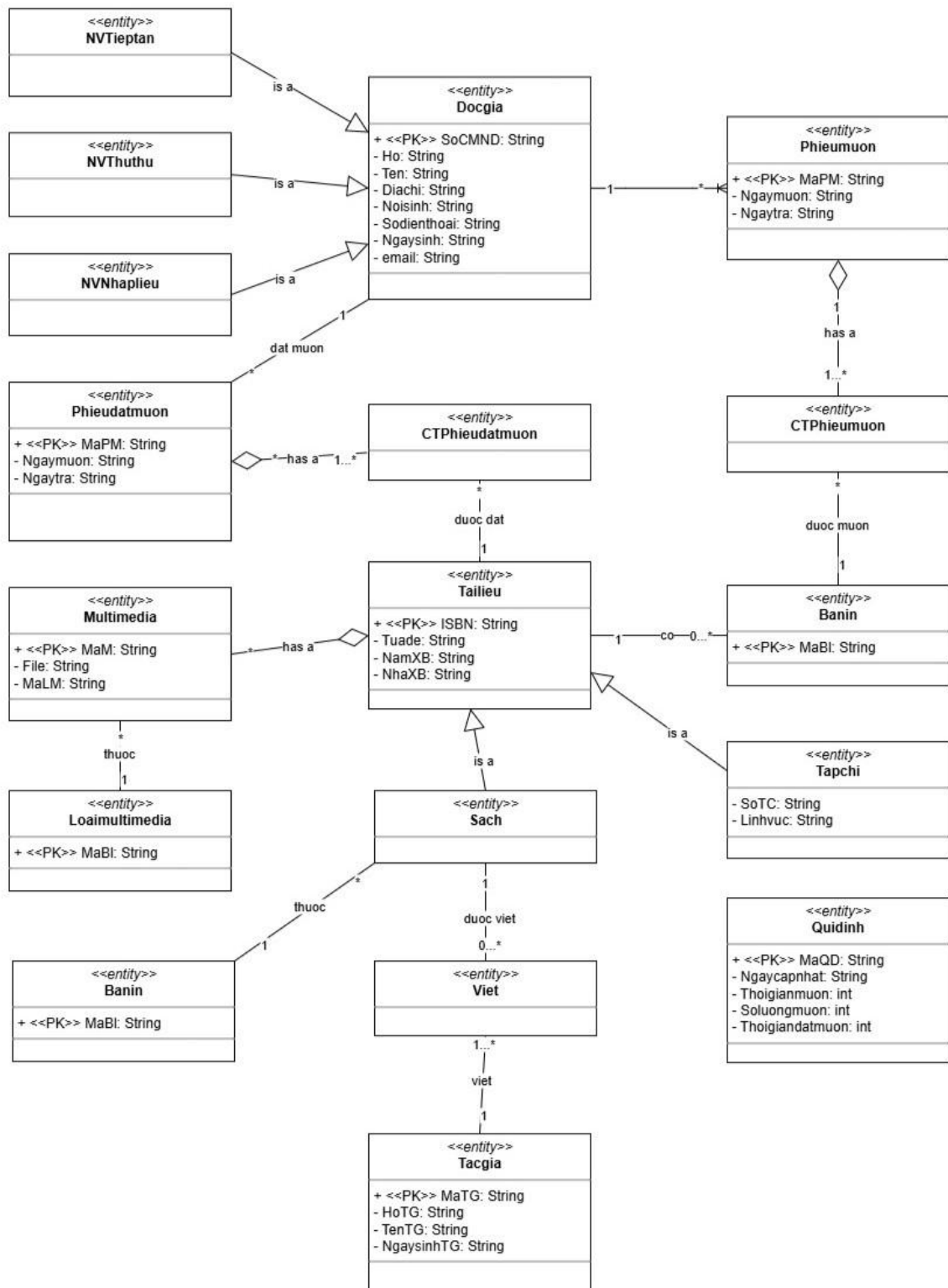


Hình 4. BT vẽ lại cho Activity Diagram trong quy trình thanh toán tiền cho sản phẩm



Hình 5. BT vẽ lại cho Sequence Diagram trong quy trình thanh toán tiền cho sản phẩm

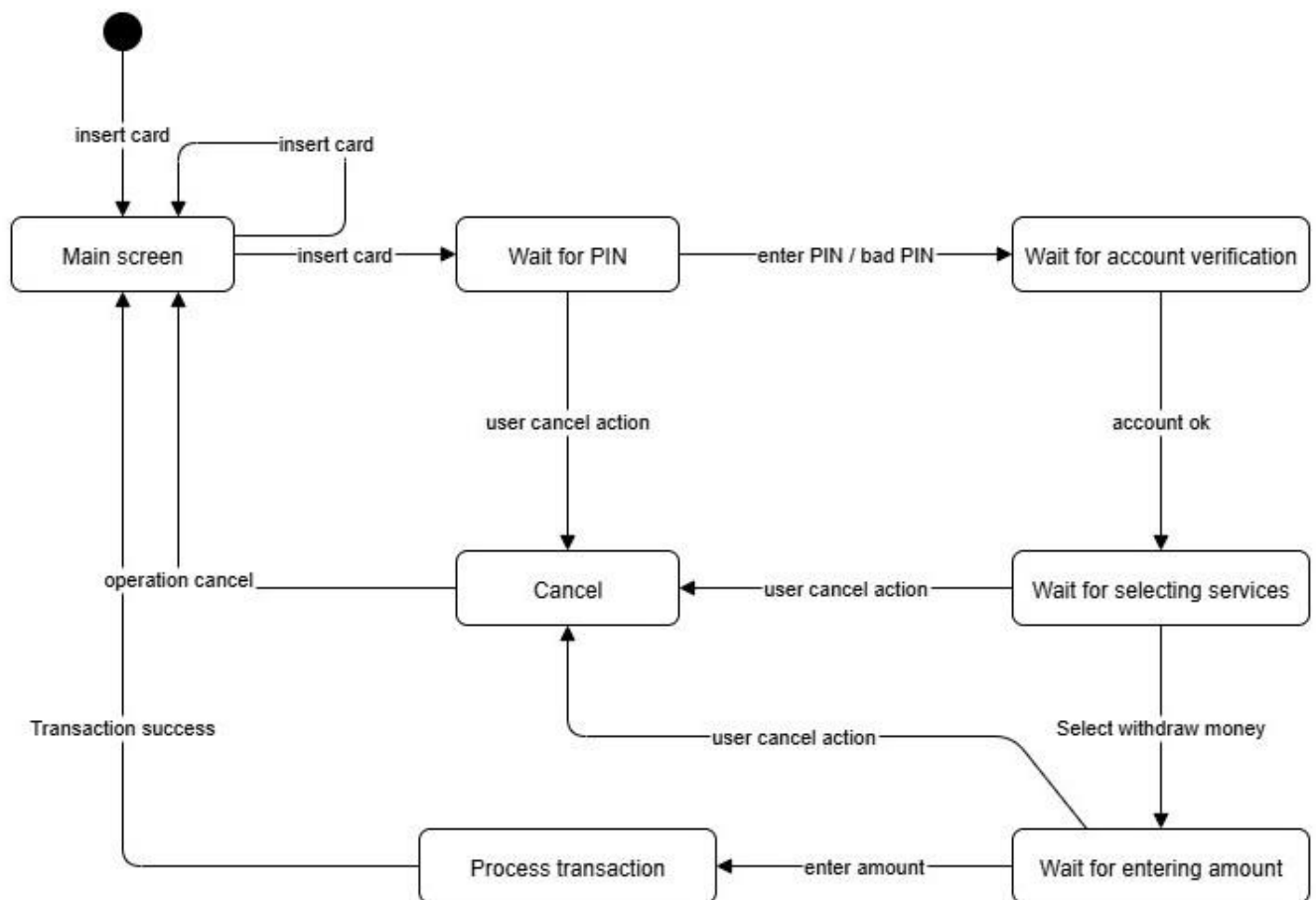
Bài 5. Vẽ lược đồ lớp cho bài toán Quản lý thư viện.



Hình 6. BT vẽ lại cho Class Diagram trong quản lý thư viện

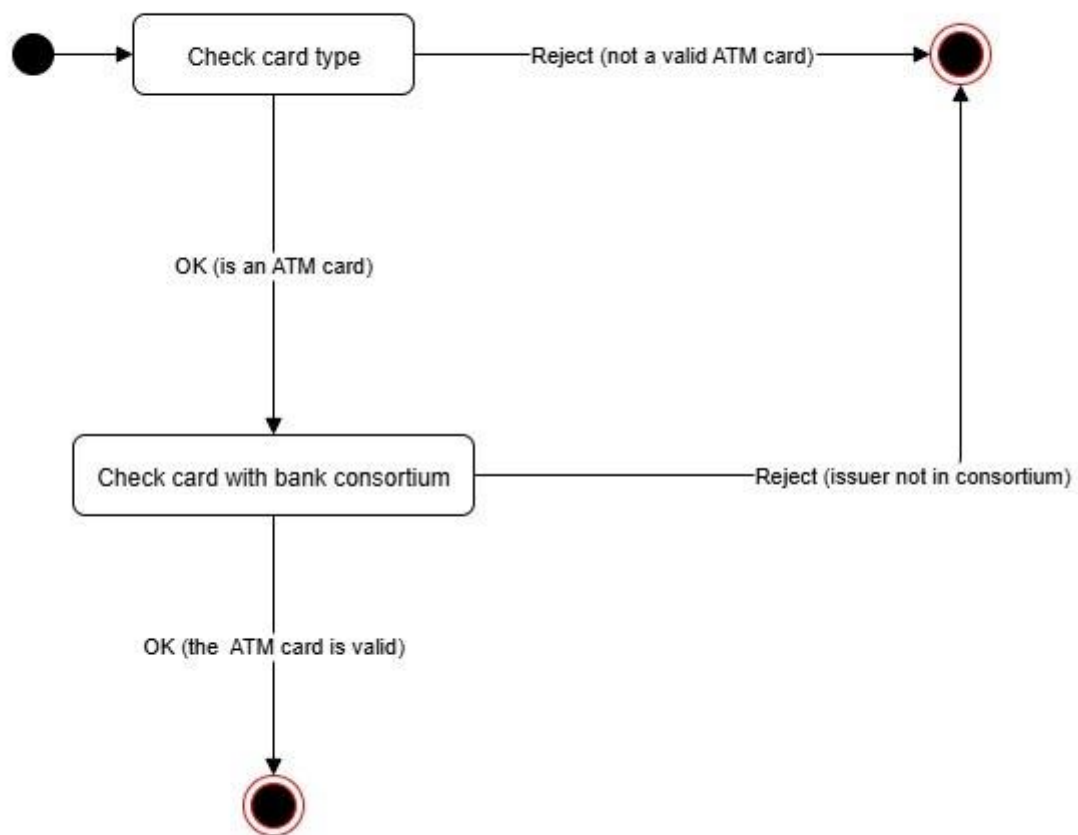
Bài 6. Vẽ lược đồ sau:

ATM



Hình 7. BT vẽ lại cho Activity Diagram trong quy trình nạp/rút tiền tại ATM

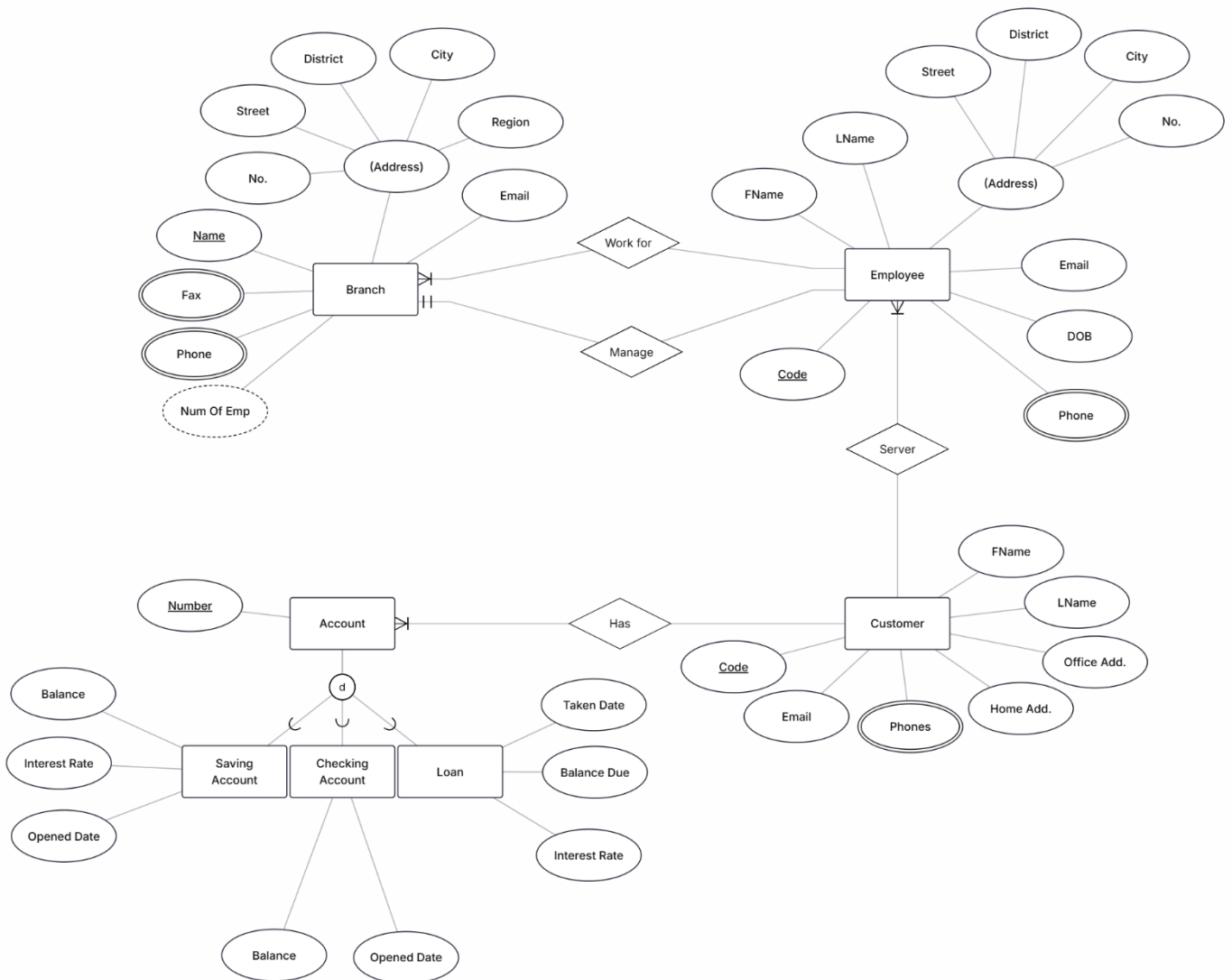
Card controller



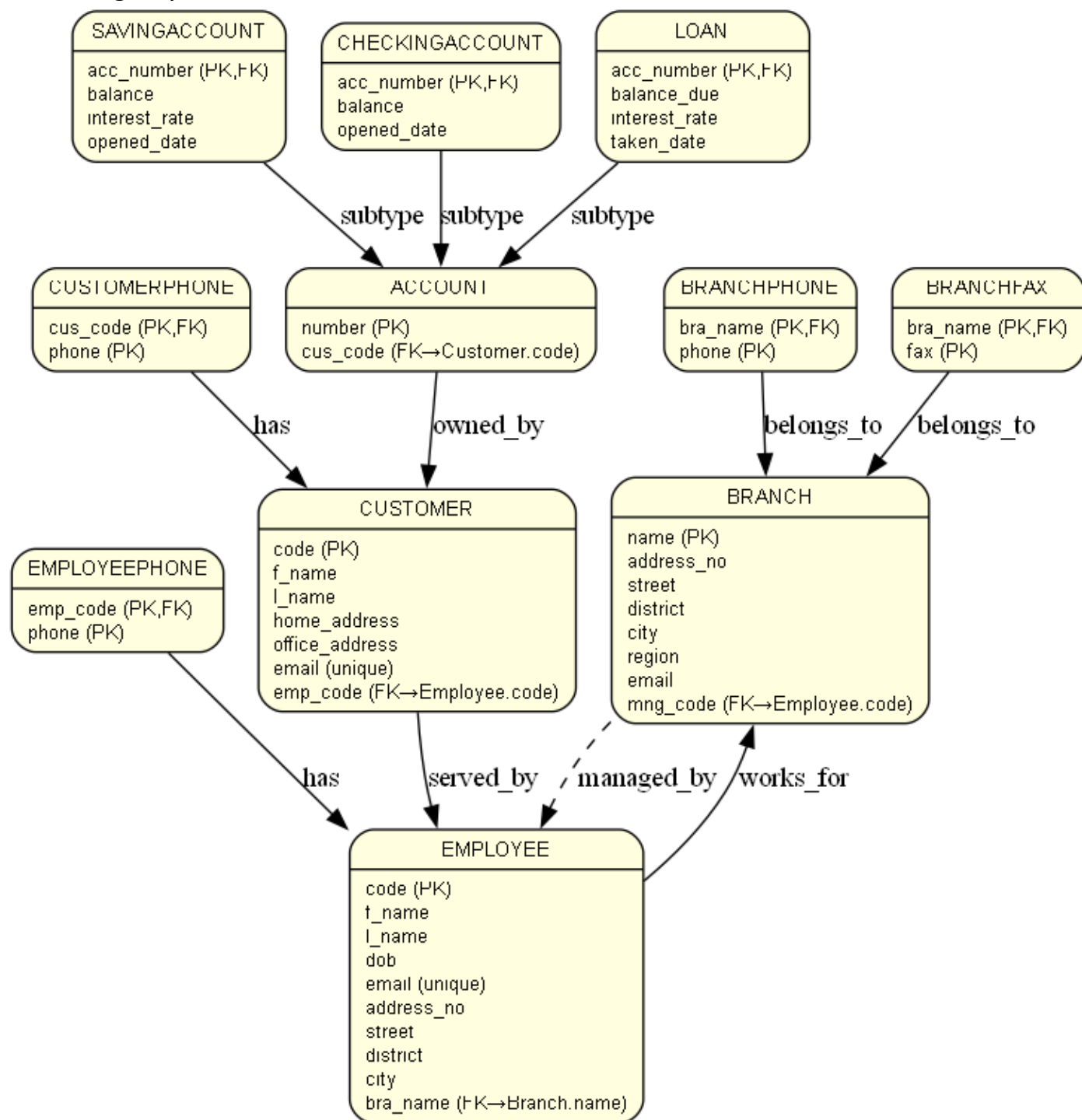
Hình 8. BT vẽ lại cho Activity Diagram trong quy trình kiểm tra tính hợp lệ của thẻ tại ATM

Bài 7. Phân tích dữ liệu cho ABC Bank

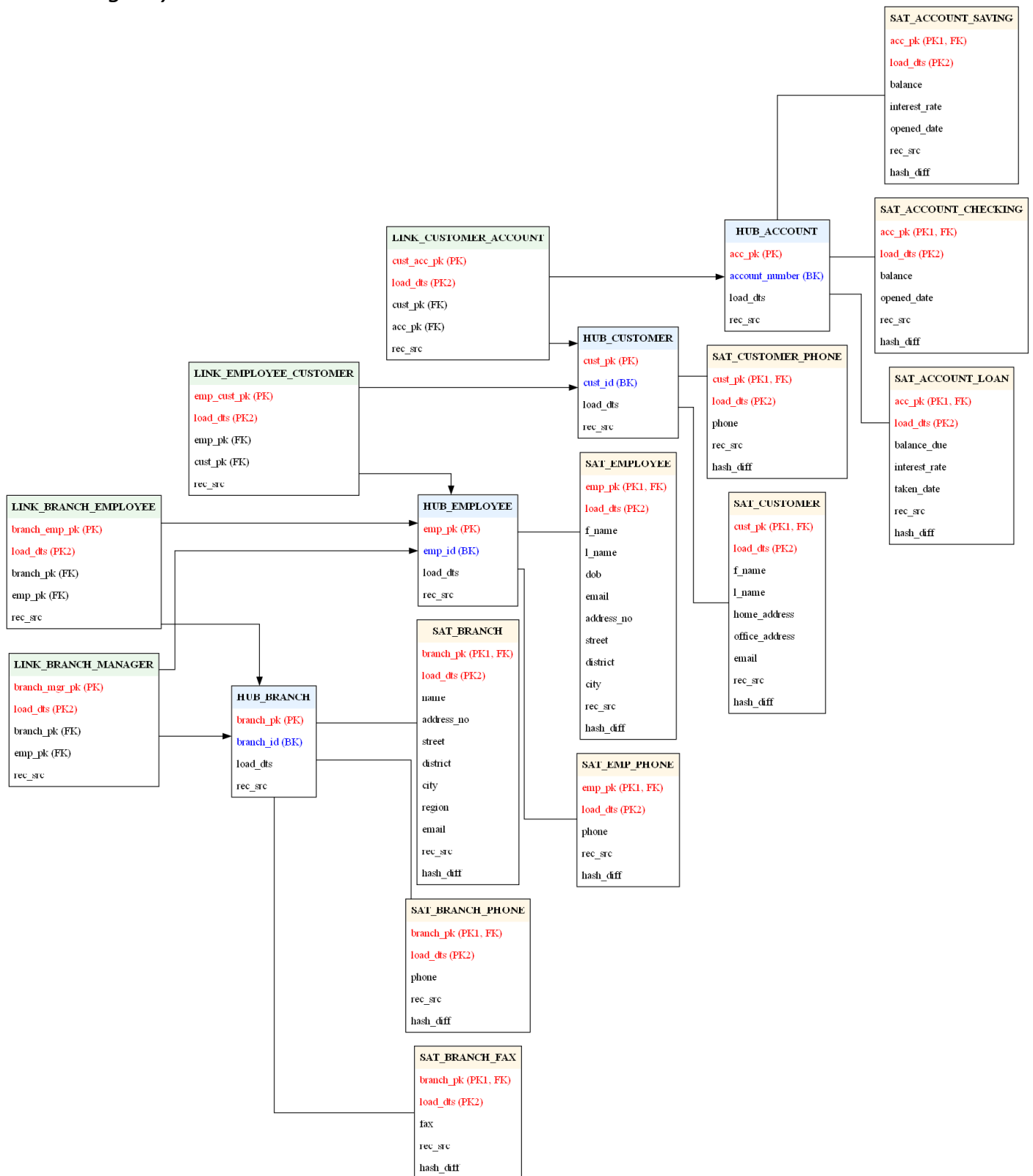
1. Thiết kế Cơ sở dữ liệu ở mức Khái niệm, Luận lý và Vật lý



Hình 9. Biểu diễn mô hình dữ liệu ở mức khái niệm



Hình 10. Biểu diễn mô hình dữ liệu ở mức luận lý (Logical)



Hình 11. Biểu diễn mô hình dữ liệu ở mức vật lý (Physical)

2. Query: Insert, Update, Delete, Select

a. You are required to insert valid and meaningful data into the database. Each table has at least 4 rows.

-- 1) BRANCH:

```
INSERT INTO BRANCH (Name, AddressNo, Street, District, City, Region, Email, MngCode)
VALUES
('Hanoi Central', '12', 'Trang Tien', 'Hoan Kiem', 'Ha Noi', 'North',
'hanoi.central@abcbank.com', NULL),
('Saigon District 1', '88', 'Nguyen Hue', 'District 1', 'Ho Chi Minh', 'South', 'sg.d1@abcbank.com',
NULL),
('Da Nang Riverside', '25', 'Bach Dang', 'Hai Chau', 'Da Nang',
'Central', 'dn.riverside@abcbank.com', NULL),
('Can Tho Mekong', '5A', 'Hoa Binh', 'Ninh Kieu', 'Can Tho',
'Mekong', 'ct.mekong@abcbank.com', NULL);
```

-- 2) BRANCHPHONE

```
INSERT INTO BRANCHPHONE (BraName, Phone) VALUES
('Hanoi Central', '+84-24-3938-1001'),
('Hanoi Central', '+84-24-3938-1002'),
('Saigon District 1', '+84-28-3821-9001'),
('Da Nang Riverside', '+84-236-382-5001'),
('Can Tho Mekong', '+84-292-381-7001');
```

-- 3) BRANCHFAX

```
INSERT INTO BRANCHFAX (BraName, Fax) VALUES
('Hanoi Central', '+84-24-3938-1999'),
('Saigon District 1', '+84-28-3821-9999'),
('Da Nang Riverside', '+84-236-382-5999'),
('Can Tho Mekong', '+84-292-381-7999');
```

-- 4) EMPLOYEE

```
INSERT INTO EMPLOYEE
```

Fabric Agency Database – Nhóm 5 – DCT122C3

```
(Code, FName, LName, AddressNo, Street, District, City, DOB, Email, BraName) VALUES
('E001','Minh','Nguyen','101','Kim Ma','Ba Dinh','Ha Noi','1985-04-
12','minh.nguyen@abcbank.com','Hanoi Central'),
('E002','Lan','Tran','22','Le Duan','Hoan Kiem','Ha Noi','1990-09-23','lan.tran@abcbank.com',
'Hanoi Central'),
('E003','Khoa','Le','77','Dong Khoi','District 1','Ho Chi Minh','1983-02-
18','khoa.le@abcbank.com','Saigon District 1'),
('E004','My','Vo','9','Ly Tu Trong','District 1','Ho Chi Minh','1992-11-
05','my.vo@abcbank.com','Saigon District 1'),
('E005','Huy','Pham','48','Tran Phu','Hai Chau','Da Nang','1986-07-
30','huy.pham@abcbank.com','Da Nang Riverside'),
('E006','Quynh','Bui','12','Nguyen Van Linh','Hai Chau','Da Nang','1993-03-
14','quynh.bui@abcbank.com','Da Nang Riverside'),
('E007','Tuan','Do','5','3/2','Ninh Kieu','Can Tho','1988-12-21','tuan.do@abcbank.com','Can Tho
Mekong'),
('E008','Trang','Ho','18','Nguyen Trai','Ninh Kieu','Can Tho','1995-06-
10','trang.ho@abcbank.com','Can Tho Mekong');
```

-- 5) Gán Manager cho mỗi chi nhánh

```
UPDATE BRANCH SET MngCode = 'E001' WHERE Name = 'Hanoi Central';
UPDATE BRANCH SET MngCode = 'E003' WHERE Name = 'Saigon District 1';
UPDATE BRANCH SET MngCode = 'E005' WHERE Name = 'Da Nang Riverside';
UPDATE BRANCH SET MngCode = 'E007' WHERE Name = 'Can Tho Mekong';
```

-- 6) EMPLOYEEPHONE

```
INSERT INTO EMPLOYEEPHONE (EmpCode, Phone) VALUES
('E001','+84-24-3555-1001'),
('E002','+84-24-3555-1002'),
('E003','+84-28-3999-3001'),
('E004','+84-28-3999-3002'),
('E005','+84-236-3777-2001'),
('E006','+84-236-3777-2002'),
```

Fabric Agency Database – Nhóm 5 – DCT122C3

```
('E007','+84-292-3666-4001'),  
('E008','+84-292-3666-4002');
```

-- 7) CUSTOMER

INSERT INTO CUSTOMER

```
(Code, FName, LName, OfficeAddress, HomeAddress, Email, EmpCode) VALUES  
( 'C001','An','Pham', 'Floor 8, 88 Nguyen Hue, D1, HCM', '12/5 Le Loi, D1, HCM',  
'an.pham@custmail.com','E003'),  
( 'C002','Binh','Vo', '101 Trang Tien, Hoan Kiem, Ha Noi', '20 Hang Bai, Hoan  
Kiem','binh.vo@custmail.com','E001'),  
( 'C003','Chi','Nguyen','25 Bach Dang, Hai Chau, Da Nang', '54 Hoang Dieu, Da  
Nang','chi.nguyen@custmail.com','E005'),  
( 'C004','Dung','Le', '5A Hoa Binh, Ninh Kieu, Can Tho', '22 Nguyen Trai, Can  
Tho','dung.le@custmail.com','E007'),  
( 'C005','Em','Tran', '88 Nguyen Hue, D1, HCM', '45 Tran Hung Dao, D5',  
'em.tran@custmail.com','E004'),  
( 'C006','Giang','Ho', '101 Kim Ma, Ba Dinh, Ha Noi', '7 Nguyen Thai Hoc, HN',  
'giang.ho@custmail.com','E002');
```

-- 8) CUSTOMERPHONE

INSERT INTO CUSTOMERPHONE (CusCode, Phone) VALUES

```
('C001','0909-111-222'),  
('C002','0912-333-444'),  
('C003','0935-555-666'),  
('C004','0922-777-888'),  
('C005','0903-999-000'),  
('C006','0988-123-456');
```

-- 9) ACCOUNT:

INSERT INTO ACCOUNT (Number, CusCode) VALUES

```
('A1001','C001'),('A1002','C002'),('A1003','C003'),('A1004','C004'),  
('A1005','C001'),('A1006','C002'),('A1007','C005'),('A1008','C006'),
```

Fabric Agency Database – Nhóm 5 – DCT122C3

```
('A1009','C003'),('A1010','C004'),('A1011','C005'),('A1012','C006');
```

-- 10) SAVINGACCOUNT

```
INSERT INTO SAVINGACCOUNT (AccNumber, Balance, InterestRate, OpenedDate) VALUES
```

```
('A1001', 850000000.00, 4.20, '2023-04-10'),
```

```
('A1002', 1200000000.00, 4.50, '2022-11-02'),
```

```
('A1003', 300000000.00, 4.00, '2024-06-15'),
```

```
('A1004', 450000000.00, 4.10, '2023-12-01');
```

-- 11) CHECKINGACCOUNT

```
INSERT INTO CHECKINGACCOUNT (AccNumber, Balance, OpenedDate) VALUES
```

```
('A1005', 150000000.00, '2024-01-05'),
```

```
('A1006', 52000000.00, '2023-08-20'),
```

```
('A1007', 90000000.00, '2024-03-01'),
```

```
('A1008', 25000000.00, '2024-04-12');
```

-- 12) LOAN

```
INSERT INTO LOAN (AccNumber, BalanceDue, InterestRate, TakenDate) VALUES
```

```
('A1009', 2200000000.00, 9.50, '2022-05-10'),
```

```
('A1010', 800000000.00, 8.90, '2023-07-22'),
```

```
('A1011', 1500000000.00, 10.20, '2024-02-10'),
```

```
('A1012', 600000000.00, 9.10, '2024-05-30');
```

b. Change last name of an employee whose code is “N1080” to “Brown”.

```
UPDATE EMPLOYEE
```

```
SET LName = 'Brown'
```

```
WHERE Code = 'N1080';
```

c. Delete the employee having code “N1080”. Explain what will happen to the customers who are severed by this employee.

```
DELETE FROM EMPLOYEE
```

```
WHERE Code = 'N1080';
```

d. Retrieve all account numbers of a customer whose name is Peter Johnson.

```
SELECT a.Number  
FROM CUSTOMER AS c  
JOIN ACCOUNT AS a ON a.CusCode = c.Code  
WHERE c.FName = 'Peter'  
AND c.LName = 'Johnson';
```

e. Find all employees who have their birthdays in March 2010.

```
SELECT Code, FName, LName, DOB  
FROM EMPLOYEE  
WHERE MONTH(DOB) = 3 AND YEAR(DOB) = 2010;
```

f. Retrieve the total balance of all saving accounts owned by a customer named “Peter Johnson”
(Supposing that he has many saving accounts)

```
SELECT COALESCE(SUM(s.Balance), 0) AS total_savings  
FROM CUSTOMER c  
JOIN ACCOUNT a ON a.CusCode = c.Code  
JOIN SAVINGACCOUNT s ON s.AccNumber = a.Number  
WHERE c.FName = 'Peter'  
AND c.LName = 'Johnson';
```

BÀI TẬP ỨNG DỤNG

Câu hỏi

1. Hãy liệt kê các yêu cầu chức năng và phi chức năng của hệ thống CoolStore dựa trên mô tả kịch bản nghiệp vụ.

Các yêu cầu chức năng (Functional Requirements):

1.1. Kiểm soát truy cập & người dùng

- FR-AUTH-01 (Must): Người dùng có thể đăng ký/đăng nhập/đăng xuất.
- FR-AUTH-02 (Must): Phân quyền: Buyer đăng nhập được chuyển tới trang Danh mục; SysAdmin tới trang Quản trị.
- FR-AUTH-03 (Must): Admin có thể kích hoạt/vô hiệu hóa tài khoản người dùng (tùy chính sách).

1.2. Danh mục sản phẩm (Product Catalog)

- FR-CAT-01 (Must): Buyer duyệt danh sách sản phẩm (phân trang).
- FR-CAT-02 (Must): Lọc theo tên, lọc/giới hạn theo giá.
- FR-CAT-03 (Must): Sắp xếp theo tên hoặc giá (tăng/giảm).
- FR-CAT-04 (Must): Xem chi tiết sản phẩm ngay trên trang danh sách bằng cách nhấp vào sản phẩm.
- FR-CAT-05 (Must): Trang chi tiết hiển thị: tên, mô tả, trạng thái “nổi bật”, tồn kho, thông tin kho (địa chỉ, v.v.), đánh giá (điểm trung bình/số lượng).
- FR-CAT-06 (Should): Đánh dấu sản phẩm nổi bật để ưu tiên hiển thị.
- FR-CAT-07 (Must – Admin): Admin CRUD sản phẩm.
- FR-CAT-08 (Must – Admin): Admin gán sản phẩm vào kho hiện có.

1.3. Giỏ hàng (Shopping Cart)

- FR-CART-01 (Must): Buyer có thể thêm sản phẩm vào giỏ từ danh sách hoặc trang chi tiết.
- FR-CART-02 (Must): Xem giỏ hàng: liệt kê các mặt hàng, số lượng, đơn giá, thành tiền.
- FR-CART-03 (Must): Cập nhật số lượng hoặc xóa sản phẩm khỏi giỏ.
- FR-CART-04 (Must): Bảng tổng hợp tự động cập nhật: tổng tiền hàng, khuyến mãi/tiết kiệm, phí vận chuyển, tổng sau khuyến mãi, tổng đơn hàng.
- FR-CART-05 (Must): Một giỏ “active” duy nhất cho mỗi Buyer; khi checkout xong giỏ được đóng/dọn.
- FR-CART-06 (Must – Admin): Admin có thể xem giỏ của bất kỳ user nào và vô hiệu hóa giỏ không hợp lệ.

1.4. Quy trình thanh toán (Payment Process)

- FR-PAY-01 (Must): Khi Buyer nhấn Thanh toán, hệ thống xác thực thông tin giỏ & sản phẩm (giá, tồn kho, tính lại tổng).
- FR-PAY-02 (Must): Xử lý thanh toán qua cổng (mock/giả lập trong bài).
- FR-PAY-03 (Must): Gửi email thông báo cho Buyer về trạng thái thanh toán/đơn hàng.

Fabric Agency Database – Nhóm 5 – DCT122C3

- FR-PAY-04 (Must): Giữ chỗ tồn kho trong lúc thanh toán; chỉ trừ tồn khi thanh toán thành công; hủy giữ chỗ nếu thất bại.
- FR-PAY-05 (Should): Tạo Đơn hàng và mục đơn hàng sau khi thanh toán thành công; lưu mã tham chiếu thanh toán.

1.5. Tồn kho/Kho (Inventory/Warehouse)

- FR-INV-01 (Must – Admin): Admin CRUD kho (địa chỉ, liên hệ).
- FR-INV-02 (Must – Admin): Quản lý tồn kho theo từng kho; có thể gán/điều chỉnh số lượng tồn cho (sản phẩm, kho).
- FR-INV-03 (Must): Buyer xem được tồn kho & thông tin kho trong trang chi tiết sản phẩm.

1.6. Đánh giá (Rating)

- FR-RAT-01 (Must): Buyer có thể đánh giá sản phẩm (1–5 sao), kèm nhận xét (tùy chọn).
- FR-RAT-02 (Must): Mỗi user chỉ được một đánh giá/sản phẩm (có thể cập nhật).
- FR-RAT-03 (Should): Hiển thị điểm trung bình và số lượng đánh giá trên trang chi tiết sản phẩm.
- FR-RAT-04 (Could – Admin): Admin duyệt/ẩn đánh giá vi phạm (moderation).

1.7. Thiết lập dữ liệu ban đầu (Seeding)

- FR-SEED-01 (Must): Khi khởi động, hệ thống tạo 1 admin, 2 buyer, dữ liệu mẫu cho sản phẩm, kho, và đánh giá cho một vài sản phẩm.

Các yêu cầu phi chức năng (Non-Functional Requirements)

2.1. Hiệu năng & khả năng mở rộng

- NFR-PERF-01: Thời gian phản hồi các API liệt kê/chi tiết sản phẩm ≤ 500 ms ở tải bình thường (≤ 50 req/s).
- NFR-PERF-02: Trang danh mục hỗ trợ phân trang và truy vấn có chỉ mục (tên, giá) để đảm bảo tốc độ ổn định khi số lượng sản phẩm lớn.
- NFR-SCAL-01: Hệ thống có thể scale ngang tầng API; dùng cache (VD: danh mục, xếp hạng trung bình) khi cần.

2.2. Tính sẵn sàng & độ tin cậy

- NFR-REL-01: Mục tiêu uptime $\geq 99\%$ cho môi trường triển khai bài tập/mô phỏng.
- NFR-REL-02: Giao dịch checkout đảm bảo tính nguyên tử (atomic): thanh toán thành công thì đơn hàng + trừ tồn được commit; thất bại thì rollback toàn bộ/giải phóng giữ chỗ.
- NFR-REL-03: Có log sự kiện (audit) cho hành động quản trị (CRUD sản phẩm/kho, vô hiệu giỏ).

2.3. Bảo mật & tuân thủ

- NFR-SEC-01: Mật khẩu băm (bcrypt/argon2), không lưu plain text.
- NFR-SEC-02: JWT/session bảo vệ API; bắt buộc HTTPS trong môi trường triển khai thực tế.
- NFR-SEC-03: RBAC: chỉ Admin mới truy cập API quản trị; Buyer chỉ được thao tác giỏ/đánh giá của chính mình.
- NFR-SEC-04: Chống Injection/XSS/CSRF ở các điểm nhập liệu; validate server-side.
- NFR-SEC-05: Rate-limit hoặc cơ chế chống lạm dụng với các thao tác nhạy cảm (login, rating).

2.4. Tính toàn vẹn dữ liệu

- NFR-DATA-01: Ràng buộc duy nhất cho (user, product) ở bảng đánh giá; không âm cho tồn kho.
- NFR-DATA-02: Một giỏ active duy nhất cho mỗi Buyer tại một thời điểm.
- NFR-DATA-03: Đồng bộ tổng giỏ (subtotal/discount/shipping/total) sau mọi thao tác thêm/xóa/cập nhật.

2.5. Khả dụng & khôi phục

- NFR-BKP-01: Chính sách sao lưu DB định kỳ (ví dụ ngày/tuần) trong môi trường lab/bài tập.
- NFR-BKP-02: Có script seed để khởi tạo nhanh dữ liệu khi reset môi trường.

2.6. Khả năng bảo trì & mở rộng

- NFR-MAINT-01: Kiến trúc tách lớp (FE/BE/DB), API RESTful, versioning cho API khi cần mở rộng.

Fabric Agency Database – Nhóm 5 – DCT122C3

- NFR-MAINT-02: Log/trace đủ để debug (mã lỗi, tương tác thanh toán/email).
- NFR-MAINT-03: Test tự động mức API/service cho các luồng chính (filter, cart, checkout, rating).

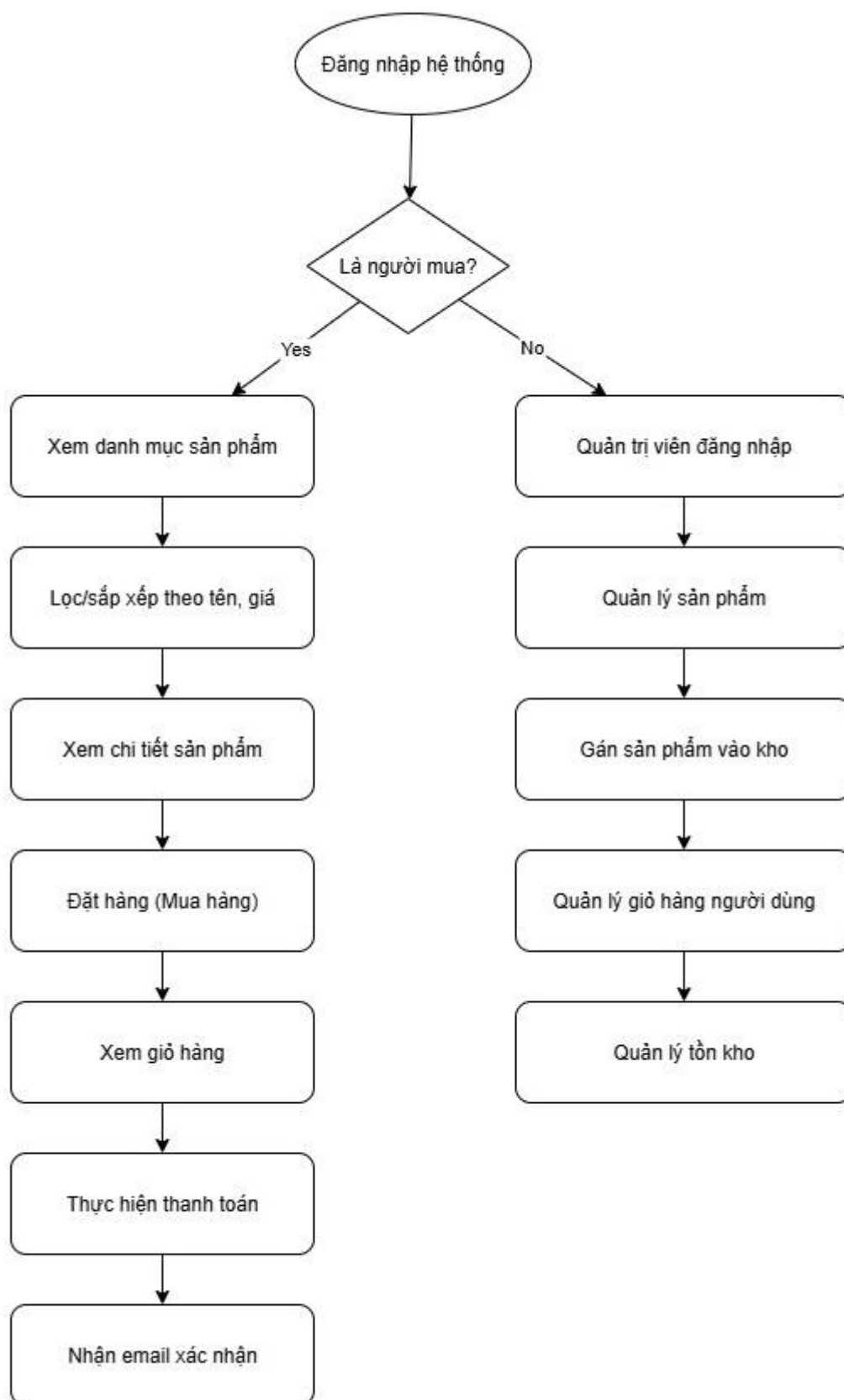
2.7. Khả dụng sử dụng (Usability) & UX

- NFR-UX-01: Giao diện phản hồi tức thời khi thêm/xóa/cập nhật giỏ (tổng tiền cập nhật ngay).
- NFR-UX-02: Thông báo rõ ràng trong các tình huống hết hàng, thanh toán lỗi, không đủ quyền.
- NFR-UX-03: Đáp ứng (responsive) trên màn hình desktop/laptop; (mobile là điểm cộng).

2.8. Tương thích & triển khai

- NFR-COMP-01: Hoạt động trên trình duyệt phổ biến (Chrome/Edge) phiên bản mới nhất gần đây.
- NFR-DEP-01: Có thể triển khai cục bộ (XAMPP/PHPMyAdmin hoặc ReactJS/Springboot + MySQL) phục vụ mục tiêu học tập.

2. Vẽ qui trình nghiệp vụ cho website CoolStore



Hình 12. Hình vẽ quy trình nghiệp vụ tại webiste bán quần áo thời trang

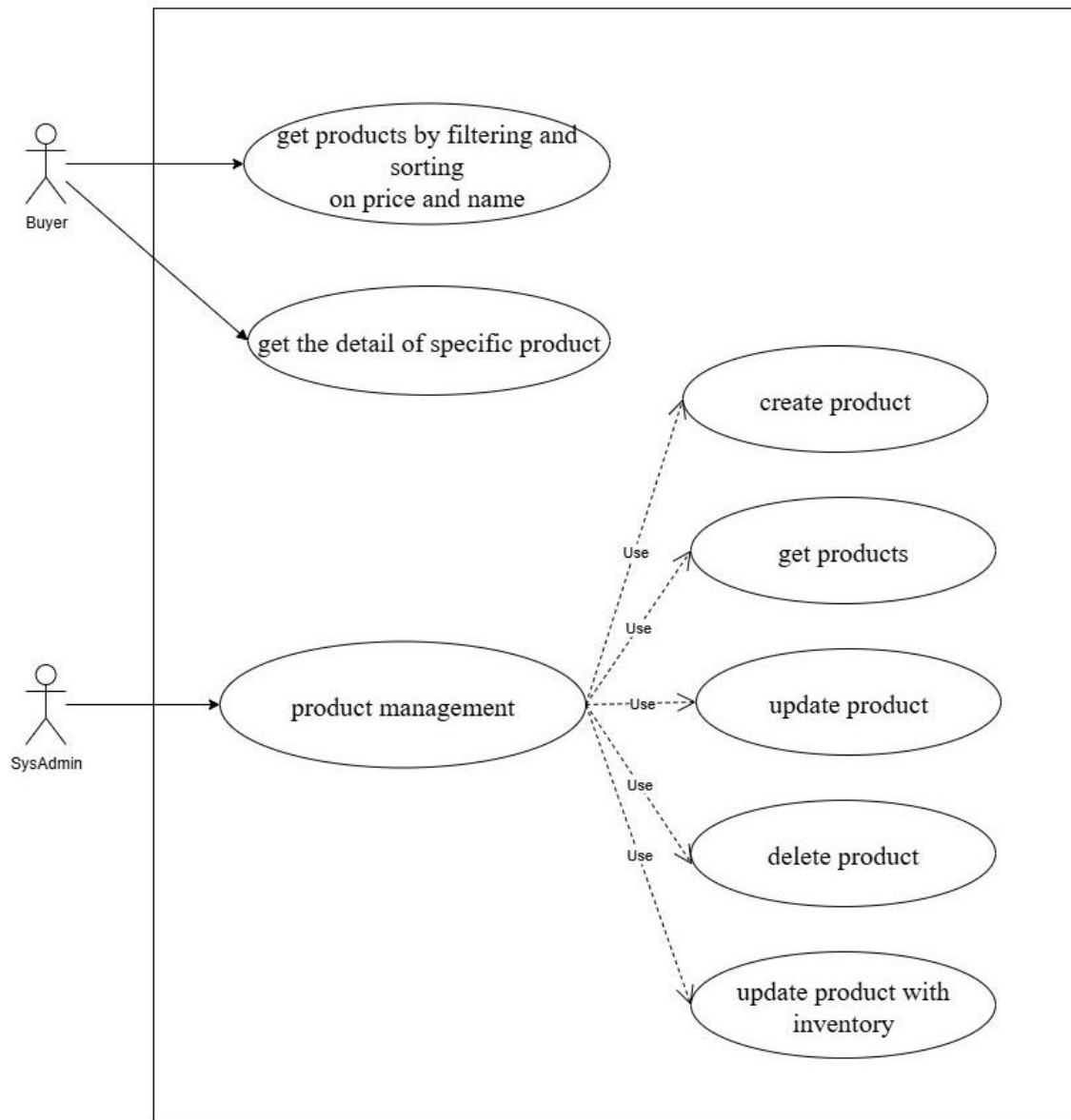
3. Xây dựng use case diagram cho website CoolStore.

- Use Case Diagram tổng quát



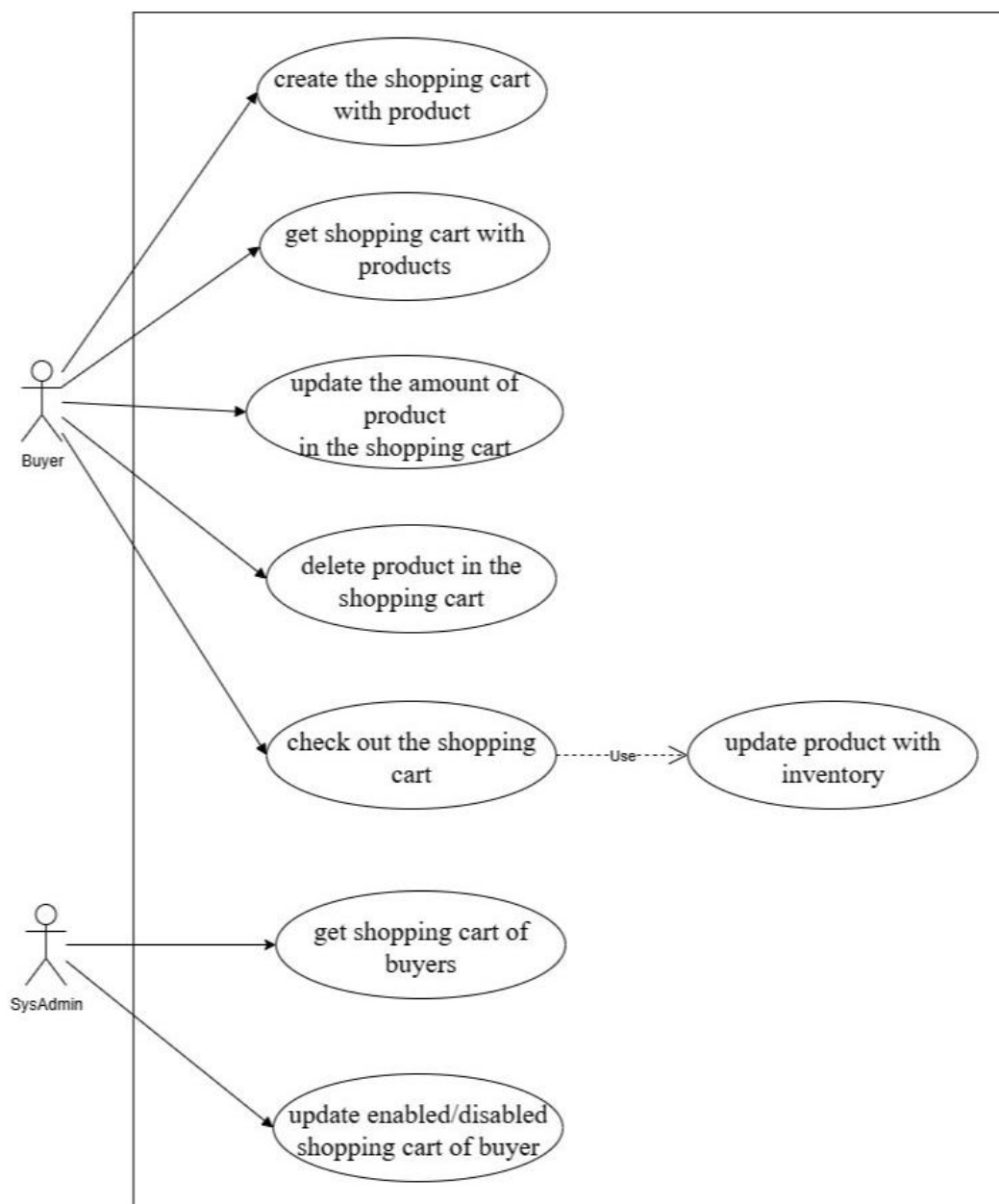
Hình 13. Hình vẽ UseCase Diagram tổng quan của website Coolstore

- Product Catalog



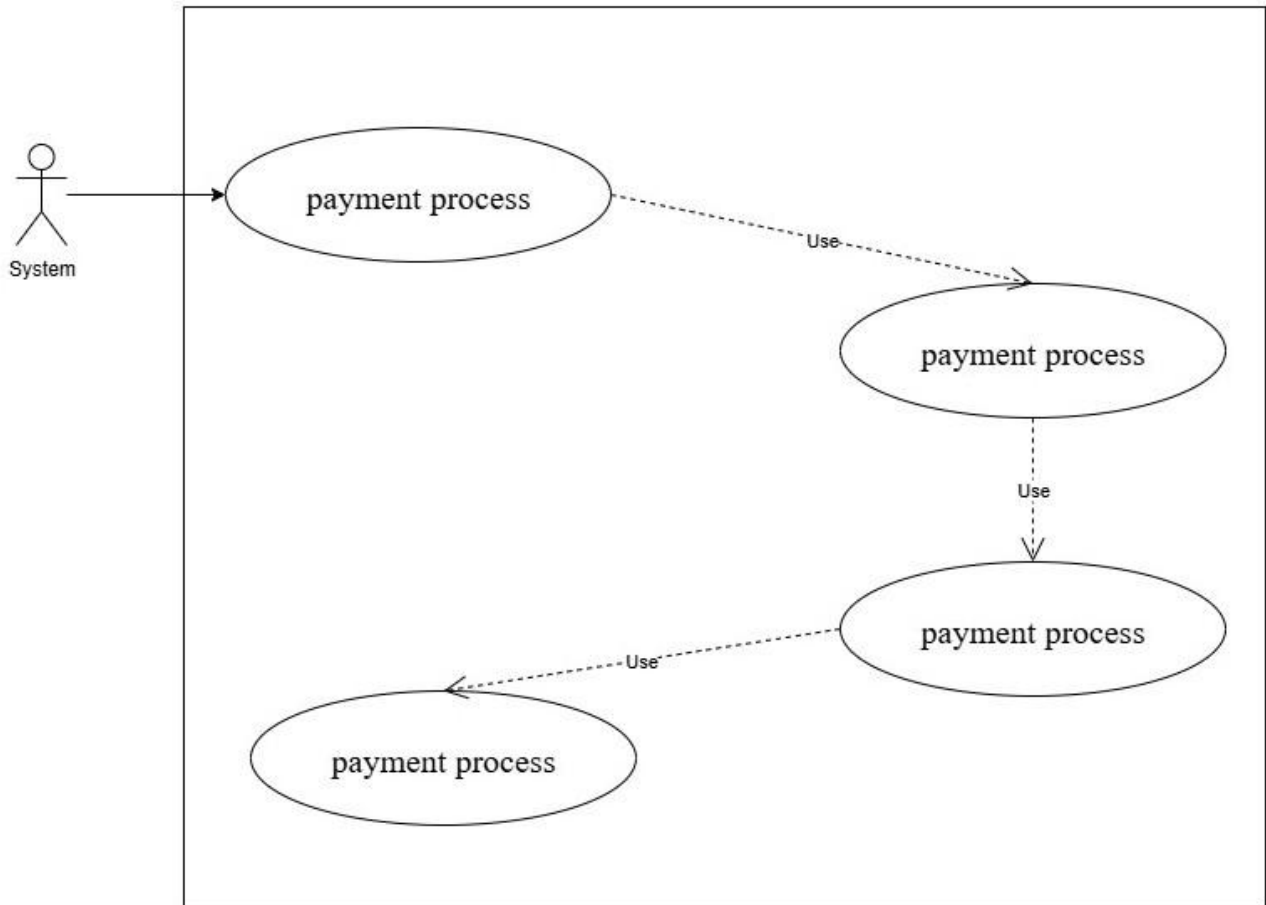
Hình 14. Hình vẽ UseCase cho xử lý các danh mục sản phẩm

- **Shopping Cart**



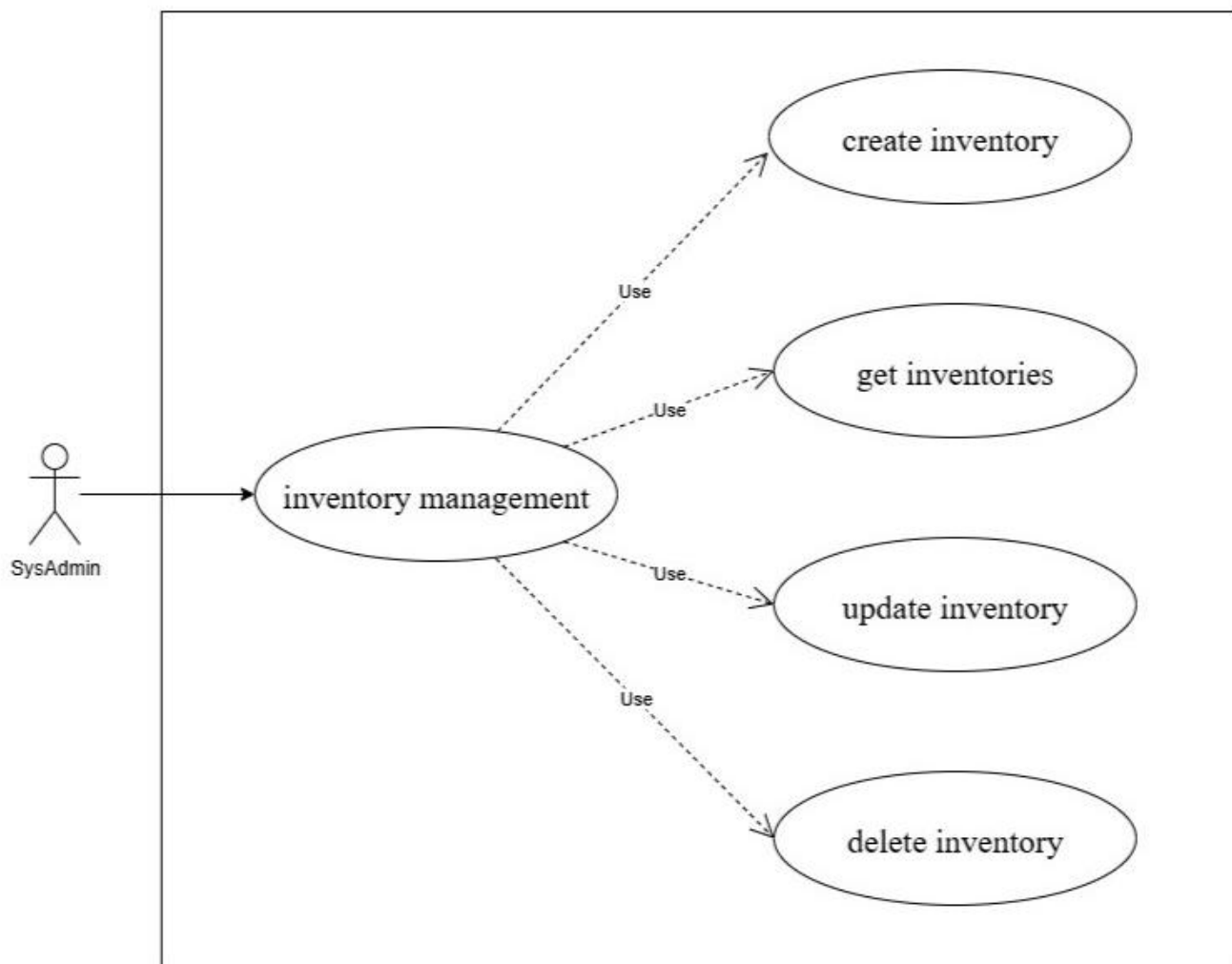
Hình 15. Hình vẽ UseCase cho phân xử lý giỏ hàng

- **Payment Process**



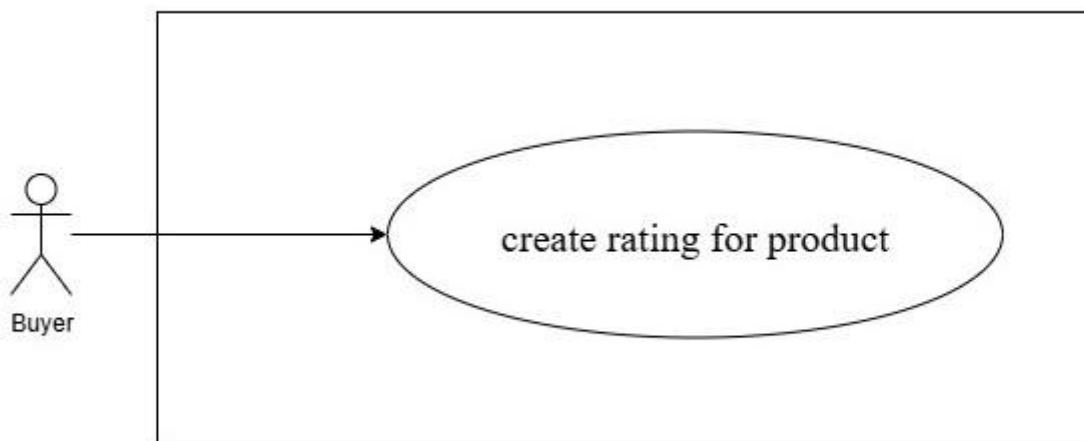
Hình 16. Hình vẽ UseCase cho quy trình thanh toán trực tuyến

- **Inventory**



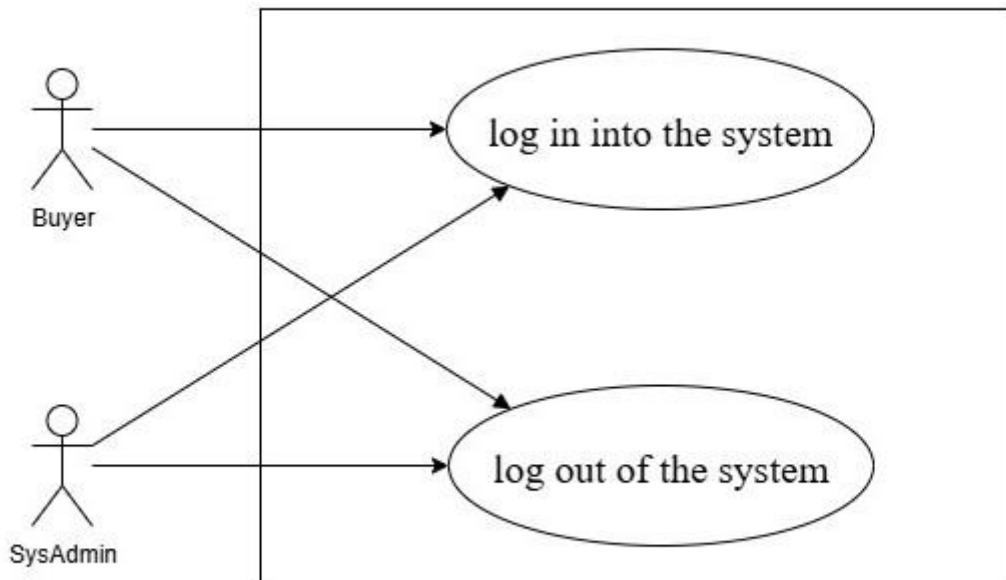
Hình 17. Hình vẽ UseCase cho quy trình xử lý hàng tồn kho

- **Rating**



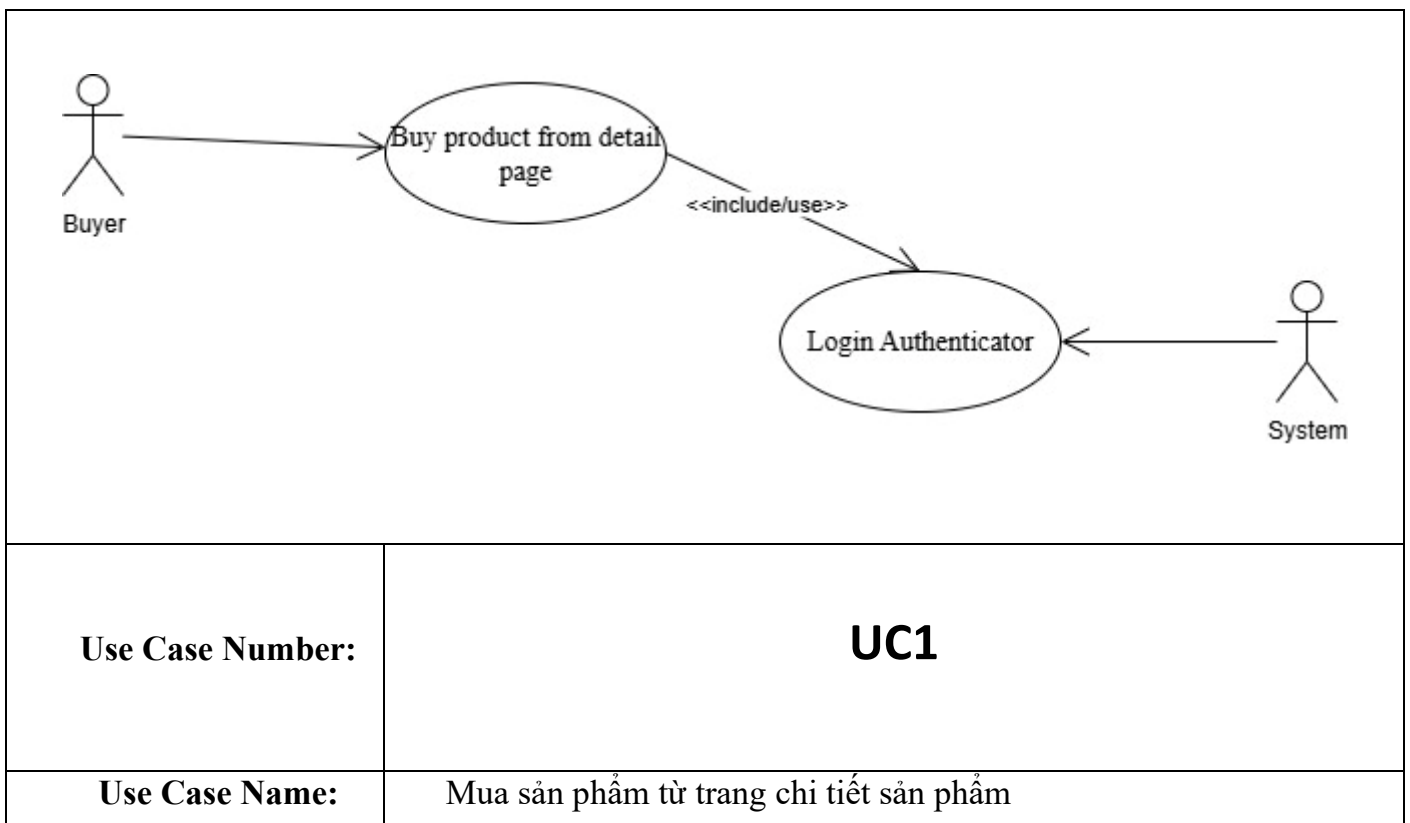
Hình 18. Hình vẽ cho UseCase phân xử lý đánh giá của người mua

- **Access Control**



Hình 19. Hình vẽ UseCase cho xử lý đăng nhập/dăng xuất trên hệ thống của cửa hàng

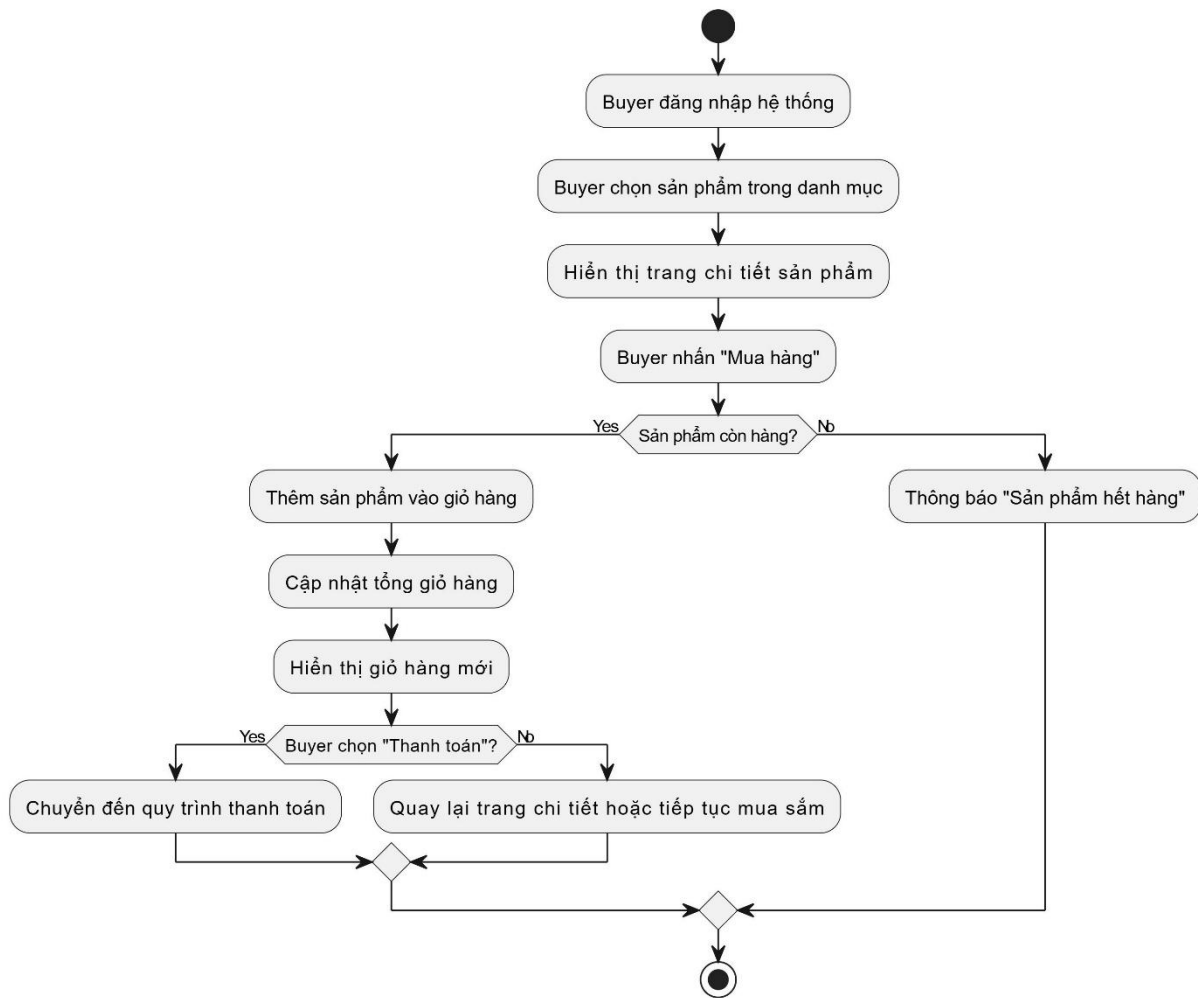
4. Mô tả chi tiết một Use Case "Mua sản phẩm từ trang chi tiết sản phẩm". Trình bày theo mẫu chuẩn gồm: Tên use case, Tác nhân chính, Mục tiêu, Tiền điều kiện, Luồng chính, Luồng thay thế, Hậu điều kiện, Ghi chú (nếu có).



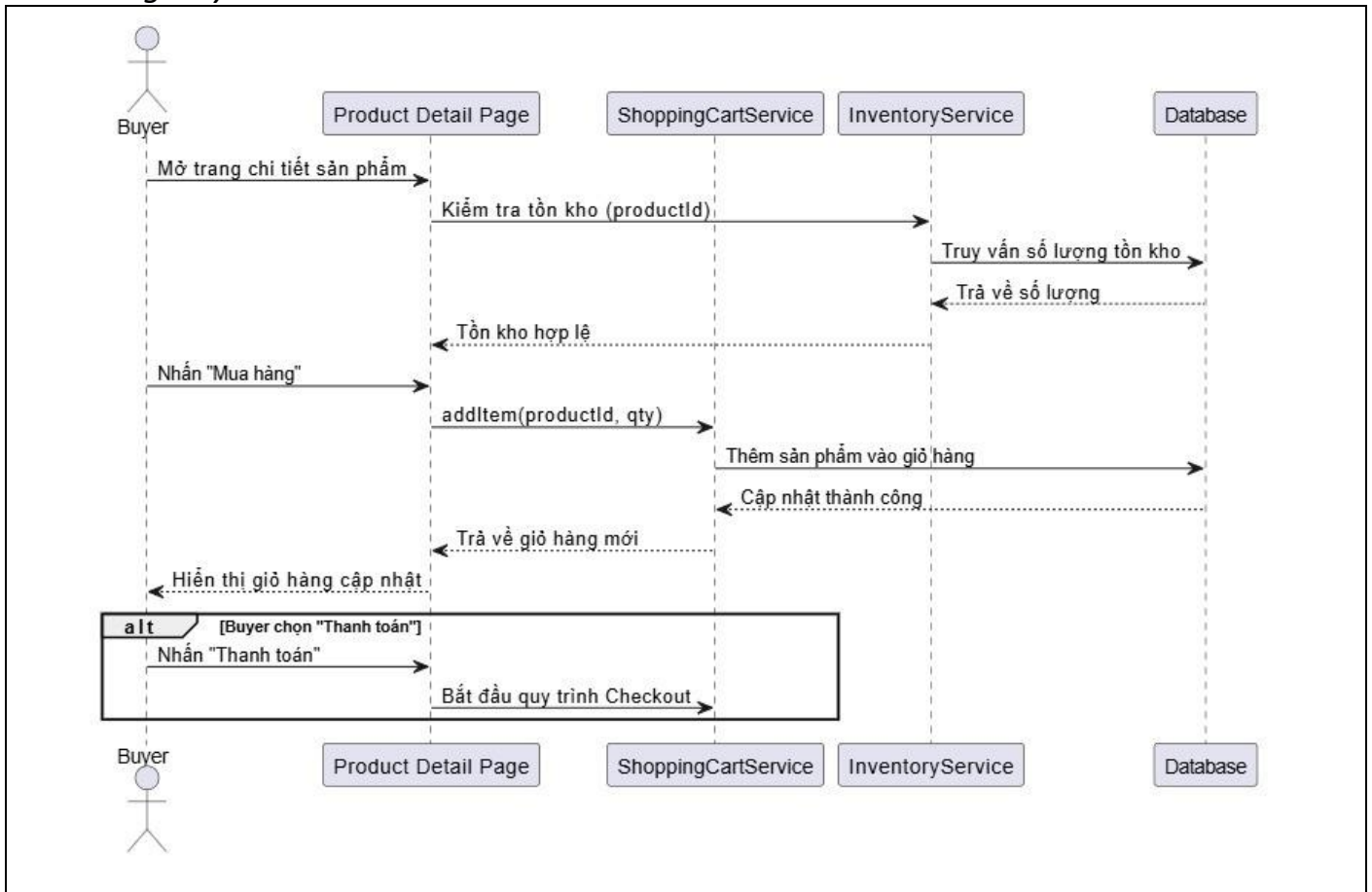
	Actor Action	System Response
Basic Course of Events:	1. Buyer đã đăng nhập và chọn sản phẩm từ danh sách. A1	2. Hệ thống hiển thị trang chi tiết sản phẩm (tên, mô tả, giá, tồn kho, đánh giá).
	3. Buyer nhấn nút “ Mua hàng ”.	4. Hệ thống kiểm tra số lượng tồn kho. A2, E2
		5. Nếu còn hàng, hệ thống thêm sản phẩm vào giỏ hàng.
	6. Buyer chọn “ Xem giỏ hàng ” hoặc tiếp tục mua sắm. A3	7. Hệ thống cập nhật giỏ hàng và hiển thị tổng chi phí, khuyến mãi, phí vận chuyển, tổng thanh toán. E1
	8. Buyer có thể chọn “ Thanh toán ” để bắt đầu quy trình thanh toán.	9. Hệ thống chuyển sang quy trình thanh toán.
Alternative Paths:	A1: Nếu Buyer chưa đăng nhập → Hệ thống chuyển hướng sang trang đăng nhập.	
	A2: Nếu sản phẩm hết hàng → Hệ thống thông báo lỗi “Sản phẩm tạm hết hàng” và quay lại trang chi tiết.	
	A3: Buyer có thể hủy thao tác mua và quay lại trang chi tiết.	
Exception Paths:	E1: Lỗi hệ thống khi cập nhật giỏ hàng → Thông báo lỗi và yêu cầu thử lại.	
	E2: Lỗi kết nối CSDL tồn kho → Hệ thống hiển thị thông báo “Không thể kiểm tra tồn kho lúc này”.	
Actor (s):	<i>Buyer (Người mua), System</i>	
Maturity:	Focused	
Summary:	<i>Người mua chọn sản phẩm tại trang chi tiết và thực hiện mua sản phẩm, hệ thống sẽ kiểm tra tồn kho, thêm sản phẩm vào giỏ hàng và cập nhật thông tin.</i>	
Exception Paths:	<ul style="list-style-type: none"> E1: Lỗi hệ thống khi cập nhật giỏ hàng → Thông báo lỗi và yêu cầu thử lại. 	

Fabric Agency Database – Nhóm 5 – DCT122C3

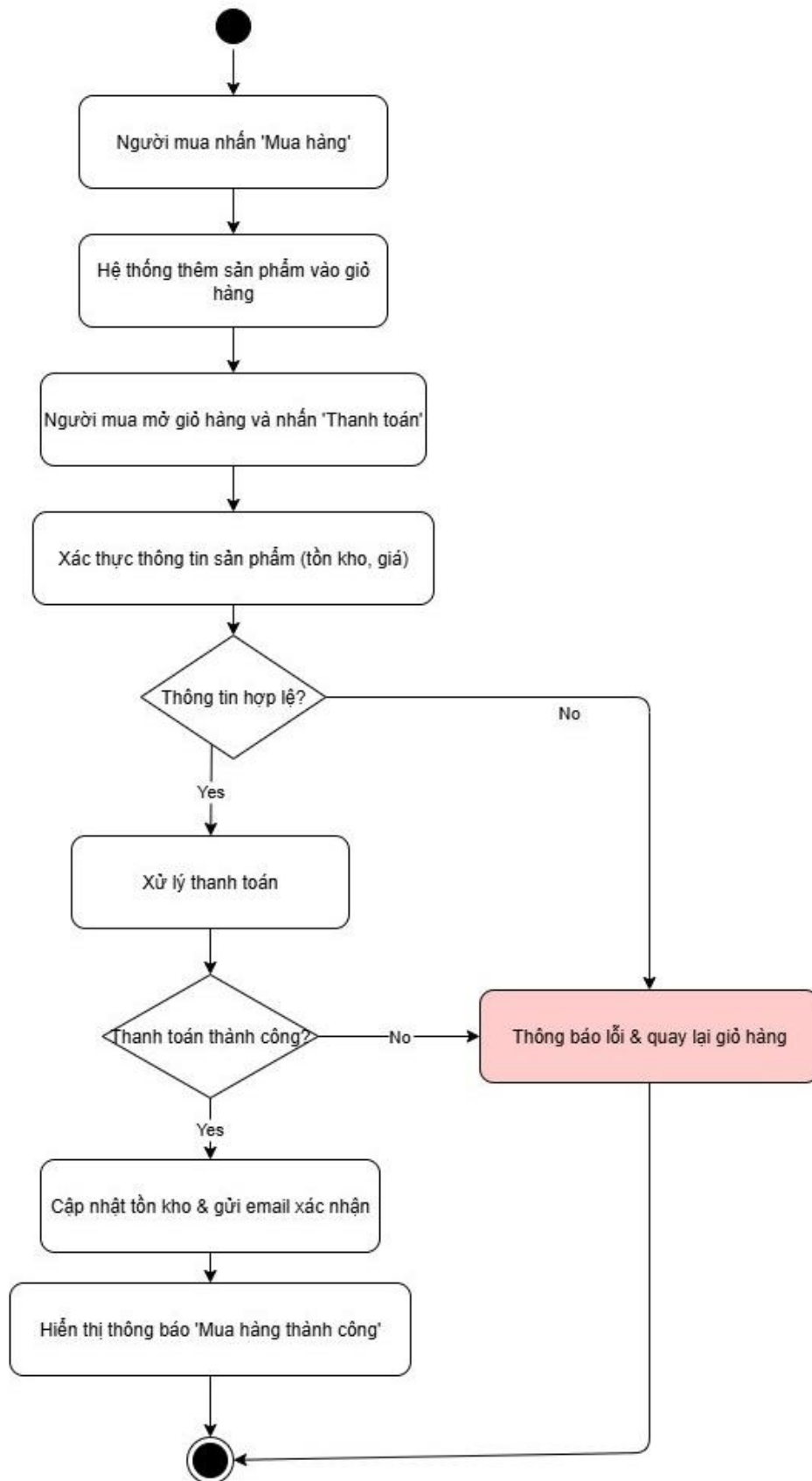
	<ul style="list-style-type: none"> • E2: Lỗi kết nối CSDL tồn kho → Hệ thống hiển thị thông báo “Không thể kiểm tra tồn kho lúc này”.
Extension Points:	Thanh toán (Checkout Process) → Khi Buyer nhấn nút Thanh toán, hệ thống chuyển sang quy trình thanh toán.
Triggers:	Buyer nhấn nút “Mua hàng” từ trang chi tiết sản phẩm.
Assumptions:	Buyer chỉ mua được sản phẩm còn trong kho.
Preconditions:	<ul style="list-style-type: none"> • Buyer đã đăng nhập vào hệ thống. • Trang chi tiết sản phẩm đã được load thành công.
Post Conditions:	<ul style="list-style-type: none"> • Sản phẩm được thêm thành công vào giỏ hàng. • Giỏ hàng của Buyer được cập nhật đúng trạng thái.
Reference: Business Rules	BR1 (tồn kho), BR2 (giỏ hàng), BR3 (thanh toán).
Author(s):	XXXXX
Date:	17-09-25
Activity Diagram:	



Sequence Diagram: Nextpage

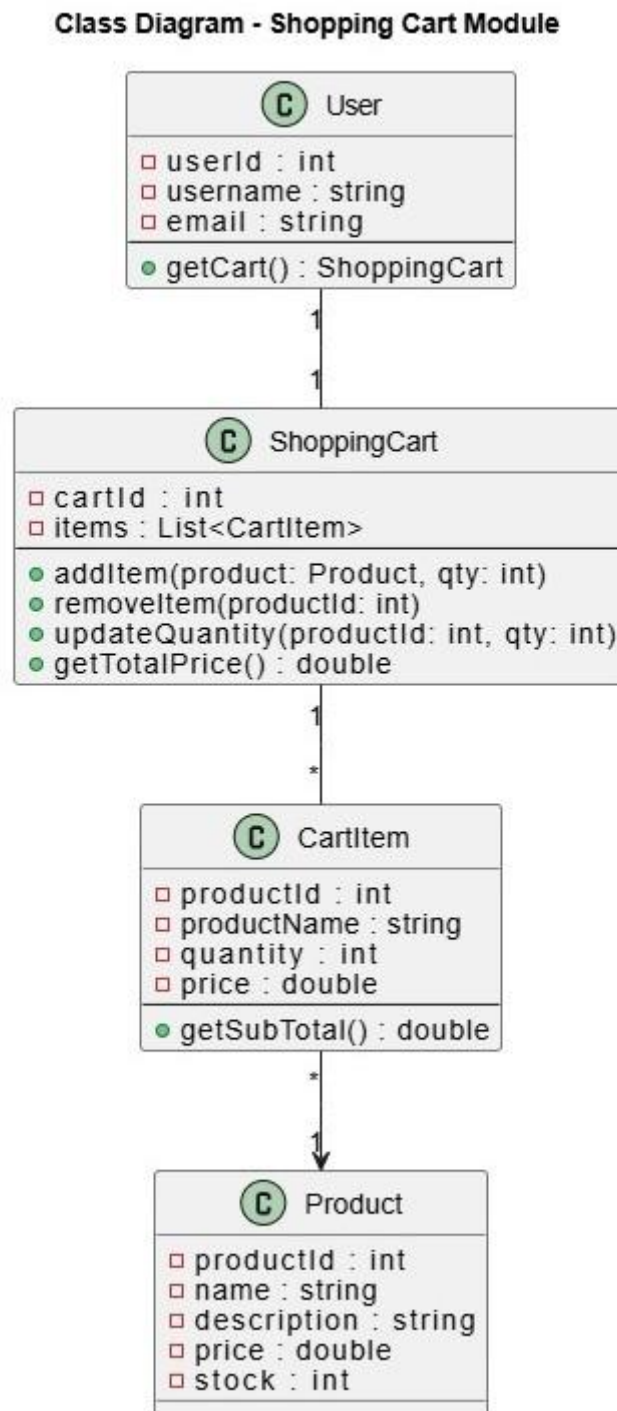


5. Sử dụng sơ đồ Activity Diagram để mô tả quy trình mua hàng từ khi người dùng nhấn “Mua hàng” cho đến khi hoàn tất thanh toán.



Hình 20. Hình vẽ Activity Diagram cho phần xử lý mua hàng

6. Thiết kế sơ đồ Class Diagram cho module Giỏ hàng.



Hình 21. Hình vẽ Class Diagram cho phân xử lý giỏ hàng

7. Thiết kế giao diện màn hình cho chức năng “Quản lý giỏ hàng” (có thể thiết kế một hoặc nhiều màn hình) gồm các thông tin:

- Danh sách sản phẩm trong giỏ hàng: tên sản phẩm, hình ảnh, số lượng, giá từng món, tổng phụ.

Fabric Agency Database – Nhóm 5 – DCT122C3

- Chức năng thao tác nhanh: Tăng/giảm số lượng sản phẩm; Xóa sản phẩm khỏi giỏ; Lưu sản phẩm để mua sau.
- Bảng tổng hợp đơn hàng: Tổng chi phí giỏ hàng; Khuyến mãi tiết kiệm được; Phí vận chuyển; Tổng thanh toán cuối cùng.
- Hành động chính: Nút “Thanh toán”; Gợi ý sản phẩm liên quan để thêm vào giỏ.

Giỏ hàng (2)

Tìm sản phẩm :

->

<input type="radio"/> Sản phẩm	Ảnh	Số lượng	Giá ▼	Tùy chọn	Thành tiền	Thao tác
<input checked="" type="radio"/> SP1	<div>ảnh thông tin #1</div>	<div><div></div>5<div></div></div>	100,000 đ	<div>Xanh lam</div> ▼	500,000 đ	<div>Xóa</div> <div>Lưu</div>
<input type="radio"/> SP2	<div>ảnh thông tin #2</div>	<div><div></div>2<div></div></div>	50,000 đ	<div>Đỏ tươi</div> ▼	100,000 đ	<div>Xóa</div> <div>Lưu</div>
Bạn có thể quan tâm:		SP3	<div>ảnh thông tin #3</div>	<div>Thêm</div> <div>Lưu</div>	<div>X</div>	

33,000 đ

Tổng tiền (5 sp)

500,000 đ

Voucher của shop:

Chọn hoặc nhập mã >

Phương thức thanh toán:

☒ Nhanh 28.700 đ

☐ Siêu tốc 140.000 đ

Phương thức thanh toán:

Xem tất cả >

Quay lại

Thanh toán

Chi tiết thanh toán:

Tổng tiền hàng:

500.000đ

Phí vận chuyển:

28.700đ

Tổng thanh toán:

528.700đ

Hình 22. Hình vẽ thiết kế phân giao diện cho chức năng quản lý giỏ hàng

8. Đề xuất cách tổ chức kiến trúc phần mềm cho hệ thống CoolStore theo mô hình kiến trúc ba lớp (3-tier architecture).

Kiến trúc phần mềm theo mô hình 3 lớp (3-tier architecture)

A. Presentation Layer

Thành phần

- Single Page App (React) / Mobile (iOS/Android).
- CDN (CloudFront, Akamai) để phục vụ ảnh, JS/CSS.
- Edge cache & HTTP caching (Cache-Control).
- Web Server / Static hosting (S3 + CloudFront).

Chức năng và cách thức hoạt động

- Hiển thị product list/detail, cart UI, checkout flow, login/logout.
- Gọi REST/GraphQL API qua HTTPS tới API Gateway.
- Lưu session client-side (localStorage) hoặc session cookie (HTTPOnly).

Bảo mật

- TLS everywhere.
- Không lưu dữ liệu nhạy cảm (thẻ) trên client.
- CSRF protection (cookie flows), CORS cấu hình đúng.

B. Application Layer (Middle Tier)

Kiến trúc

- **API Gateway** (nginx/kong/traefik) làm entry point: routing, rate-limit, auth (JWT validation), TLS termination, caching edge.
- **Auth Service / Identity Provider:** OAuth2/OpenID Connect (Keycloak/Auth0) — issue JWT + refresh tokens.
- **Domain Microservices:**
 - ProductService (catalog, search index integration)
 - CartService (persistent cart per user)

Fabric Agency Database – Nhóm 5 – DCT122C3

- CheckoutService / PaymentService (orchestrator + payment gateway adapter)
- InventoryService (stock counts, reservations)
- RatingService (reviews)
- AdminService (product management, warehouse assignment)
- **Supporting services:** Email worker (send order confirmations), NotificationService, Monitoring/Logging agents.

Giao tiếp

- Sync: REST/GraphQL or gRPC cho low-latency internal calls.
- Async: Message broker (Kafka/RabbitMQ) cho event-driven (order placed → inventory reserved → shipping) và để tách rời bounded contexts.

Patterns

- **Idempotency keys** cho checkout/payment requests.
- **Circuit breaker + retry** (Hystrix / Resilience4j).
- **Saga pattern** cho business transactions phân tán (reserve inventory, charge payment, confirm order) — dùng choreography (events) hoặc orchestration (CheckoutService điều phối).
- **Cache** (Redis) cho product read-heavy data và session/cart hot data.

Scale & HA

- Các service stateless (hộp lý) → horizontal scaling.
- Stateful components: DB, Redis cluster, Kafka clusters → scale theo tải.

C. Data Layer

Đề xuất lưu trữ theo service

- ProductService: PostgreSQL + Elasticsearch (search & filter full-text).
- CartService: Redis (fast ephemeral carts) + persistent store (Postgres / DynamoDB) for long-term carts.
- InventoryService: Relational DB (Postgres/MySQL) with strong consistency; hoặc use NewSQL if cần.
- RatingService: MongoDB / Cassandra (document storage) — nếu cần scale nhiều đọc/ghi.

Fabric Agency Database – Nhóm 5 – DCT122C3

- **PaymentService:** không lưu CC — dùng payment gateway tokenization (Stripe, Adyen).
- **Shared infra:** Redis cache, CDN, Message broker (Kafka/RabbitMQ), Blob storage (S3) cho hình ảnh.

Backup / DR: Daily snapshot, point-in-time recovery (PITR), cross-region replica nếu cần.

Operational concerns (đề xuất triển khai)

- **Containerization:** Docker + Kubernetes (K8s) cho orchestrating, autoscaling, rolling update.
- **Ingress & Load Balancer:** nginx/ALB + API Gateway.
- **Config & Secret management:** Vault / Kubernetes Secrets.
- **CI/CD:** Pipeline (GitHub Actions / GitLab CI / Jenkins) với unit → integration → contract tests → canary deploy.
- **Monitoring & Observability:** Prometheus + Grafana, ELK/EFK (Elasticsearch/Fluentd/Kibana) logs, Jaeger/Zipkin tracing.
- **Health checks & Liveness/Readiness probes** cho K8s.
- **Rate limiting / WAF** cho public API.

9. Viết ít nhất 3 ca kiểm thử (test cases) cho chức năng “Thêm sản phẩm vào giỏ hàng”
– bao gồm dữ liệu đầu vào, bước thực hiện và kết quả mong đợi.

Các test cases có thể có cho chức năng “Add to Cart”:

TC-01 – Thêm sản phẩm lần đầu (tạo giỏ mới)

- **Tiền điều kiện:** User (buyer) đã đăng nhập; chưa có giỏ active.
- **Dữ liệu đầu vào:** productId=PID_001, qty=2. Sản phẩm đang is_active=1, stock_qty ≥ 2, unit_price=2.290.000.
- **Bước thực hiện:** Gọi POST /carts/items {productId: PID_001, qty: 2} (hoặc click “Mua hàng” trên UI).
- **Kết quả mong đợi:**

Fabric Agency Database – Nhóm 5 – DCT122C3

- Hệ thống tạo giỏ active mới cho user (nếu chưa có).
- Thêm cart_item với quantity=2, unit_price lấy theo products.price, line_total=4.580.000.
- Tự động tính subtotal/discount/shipping/total đúng quy tắc (ví dụ shipping 30.000 nếu subtotal \leq 10.000.000).
- API trả 200/201 và payload có cart summary cập nhật đúng.

TC-02 – Thêm lại cùng sản phẩm (gộp dòng, tăng số lượng)

- Tiền điều kiện: Có giỏ active với PID_001 đã tồn tại quantity=2; stock_qty hiện có ≥ 3 (đủ để tăng).
- Dữ liệu đầu vào: productId=PID_001, qty=1.
- Bước thực hiện: POST /carts/items {productId: PID_001, qty: 1}.
- Kết quả mong đợi:
 - Không tạo dòng mới, update dòng hiện tại: quantity=3.
 - 'line_total' của dòng = unit_price * 3.
 - Subtotal/discount/shipping/total recalculated.
 - API trả 200; tổng tiền tăng đúng bằng giá * 1 so với trước.

TC-03 – Thêm vượt tồn kho (fail & gợi ý)

- Tiền điều kiện: Có giỏ active hoặc chưa có cũng được; PID_002 còn stock_qty=1.
- Dữ liệu đầu vào: productId=PID_002, qty=3.
- Bước thực hiện: POST /carts/items {productId: PID_002, qty: 3}.
- Kết quả mong đợi:
 - Server từ chối thêm, không thay đổi giỏ.
 - HTTP 409 Conflict (hoặc 422) với thông điệp “Không đủ tồn kho”, kèm gợi ý qty khả dụng = 1.
 - Subtotal/total không đổi.

TC-04 – Thêm sản phẩm không tồn tại/không hoạt động

- Tiền điều kiện: PID_XYZ không tồn tại (hoặc is_active=0).
- Dữ liệu đầu vào: productId=PID_XYZ, qty=1.
- Bước thực hiện: POST /carts/items {productId: PID_XYZ, qty: 1}.
- Kết quả mong đợi:

Fabric Agency Database – Nhóm 5 – DCT122C3

- HTTP 400 Bad Request hoặc 404 Not Found (tùy thiết kế).
- Không tạo/đổi giỏ; không thêm cart_item.
- Thông báo lỗi “Sản phẩm không hợp lệ/đã ngừng kinh doanh”.

TC-05 – Giỏ active bị vô hiệu hóa → tạo giỏ mới & thêm thành công

- Tiền điều kiện: User có cart status = disabled (bị Admin vô hiệu); không có giỏ active nào khác; PID_003 còn đủ tồn.
- Dữ liệu đầu vào: productId=PID_003, qty=2.
- Bước thực hiện: POST /carts/items {productId: PID_003, qty: 2}.
- Kết quả mong đợi:
 - Hệ thống không thêm vào giỏ disabled.
 - Tự tạo giỏ active mới, thêm PID_003 với quantity=2.
 - Tính lại summary đúng; trả 201/200 kèm cart mới.

TÀI LIỆU THAM KHẢO

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